ABSTRACT

COMPUTATION OF OPTIMAL CONTROLS FOR NONLINEAR SYSTEMS VIA GEOMETRIC SEARCH

By

Richard Blain Stratton

This thesis develops computational procedures for determining optimal controls for a rather general class of nonlinear systems. The procedures combine the general applicability of search techniques with the more rapid convergence of reachable set-oriented methods. Example problems and numerical results are given for the algorithms developed.

The minimum distance problem is principally considered although the time optimal control problem is also discussed. Extensions to other control problems are also possible. For the minimum distance problem, all optima lie on the boundary of the reachable set--the collection of attainable system states for a specified final time.

Reachable sets resulting from linear systems are convex, and the optimum final state is well-defined and in most cases unique. For nonlinear systems, however, the resulting reachable sets are, in general, nonconvex. As a result, there may be many boundary points on the reachable set which are optima in a local sense. The global optima or optimum, if unique, are found within this collection of local optima. Since the reachable set may not be convex, many of the pre-

viously developed reachable set techniques are not easily applied. Thus a relatively new approach is taken.

To evaluate each control which is considered, several error functions are developed which depend on the collinearity of the final adjoint and the final state at an optimum. In as much as an explicit expression for the boundary of the reachable set is not available, principles from differential geometry are used to define a path on the boundary of the reachable set. A sequence of final system states for which the error functions decrease monotonically to the optimum value characterizes the path.

Because of the fact that the reachable set is defined implicitly through the system differential equation, it is not possible to write an explicit equation for this path. However, an algorithm to determine an optimum final state is developed utilizing an approximation to the path. Because of the approximate nature of this path, several alternative decisions relating to the algorithm are considered and their relationship to the error functions are investigated.

Some special problems pertaining to reachable set characteristics are discussed and shown to be related to the global problem--that of find a global optimum. To treat the global problem, a random sequence of starting points are generated as a basis for each determination of a local optimum.

Richard Blain Stratton

Several example nonlinear systems are considered and algorithm alternatives are compared. Example computational results for a variety of applications are given as are example reachable sets and trajectories. A summary of the theoretical and computational results for the algorithms developed in this thesis is presented in the concluding section.

COMPUTATION OF OPTIMAL CONTROLS FOR NONLINEAR SYSTEMS VIA GEOMETRIC SEARCH

By

Richard Blain Stratton

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Electrical Engineering and Systems Science

1969

To My Wife And Our Parents

ACKNOWLEDGMENTS

The author wishes to express his appreciation to the members of his committee for their interest, encouragement and advice. In particular, the significant and thoughtful suggestions, guidance and criticisms of the committee chairman, Dr. Robert O. Barr, are gratefully acknowledged. Special gratitude is due Dr. Leroy M. Kelly for the informative and helpful discussions of differential geometry.

The author also deeply appreciates the assistance of his wife, Teddy, in typing the manuscript and her continuing support and encouragement.

TABLE OF CONTENTS

	Page
List of Tables	vii
List of Figures	viii
Chapter 1 Introduction	1
Chapter 2 Problem Formulation and Reachable Set Concepts	6
2.1 Notation and Terminology	6
2.2 System Definition	10
2.3 Optimal Control Problem Definition	14
2.4 Minimum Regulator Problem	15
2.5 Reachable Set Definition	15
2.6 Pontryagin's Maximum Principle	16
2.7 Normality	22
2.8 Properties of the Reachable Set	25
2.9 Problem Statements	27
Chapter 3 The Local Optimum Procedure	29
3.1 A Discussion of Iterative Methods	30
3.2 Initial Costate Iterations and Error Functions	32
3.3 Determination of a Local Optimum via Direct Search	39
3.4 Essential Concepts from Differential Geometry	45
3.5 Convergence to a Local Optimum via Lines of Curvature	50

Page

3.6 Effective Boundary Path Selection 56

3.6.1 Curvature Algorithm Alternatives--Final Time 59

3.6.2 Curvature Algorithm Alternatives--Initial Time 72

3.7 The Local Optimum Procedure--Composite Method 74

Chapter 4 The Global Optimum and Related Problems 80

4.1 Special Problem 1: Nonconvex regions on ∂R(T) 81

4.2 Special Problem 2: Flats and Corners 87

4.3 Special Problem 3: Extremum but Not Local Optimum 90

4.4 Special Problem 4: Internal Boundaries 91

4.5 Special Problem 5: False Boundary Points 93

4.6 Special Problem 6: Origin Interior to R(T) 95

4.7 The Global Optimum 97

4.8 Application of GOP to Time Optimal Control Problems 99

Chapter 5 Computational Results and Conclusions 103

5.1 Example Systems 104

5.2 An Introduction to Computational Examples 108

5.3 Computational Comparisons of Algorithm Alternatives 110

5.3.1 Study of Perturbation Relationships--Final Time 111

5.3.2 Comparison of Curvature Values 113

5.3.3 Effect of the Basic Curvature Formula Choice 123

5.3.4 Comparison of LOP-CM Subalgorithms 4a and 4b 124

5.3.5 Comparison of Perturbation Direction Alternatives 125

Page

5.3.6 Comparison of Perturbation Step Size Alternatives 126

5.3.7 Analog Error and the Integration Correction Routine 127

5.3.8 Comparison of Analog vs Digital Integration in LOP-CM 131

5.4 Comparison of LOP-CM with LOP-DS 134

5.5 Global Optimization Examples 134

5.5.1 Examples for ES-1: 2nd-Order System, Scalar Control 136

5.5.2 Examples for Higher Order Systems 138

5.6 Time Optimization Examples 138

5.7 Summary and Conclusions 147

Bibliography 156

Appendix A Analog Diagram for Example Problem 1 159

Appendix B Example Computer Programs 160

Appendix C Input Data Sets 174

Appendix D Possible Extensions and Future Investigations 175

LIST OF TABLES

Table		Page
5.1	Comparison of Various Analog Estimates of Curvature	116
5.2	Digital Estimates of Curvature for a 2nd-Order System	118
5.3	Digital Estimates of Curvature for a 3rd-Order System	120
5.4	Evaluation of the Significance of σ_n	122
5.5	Forward-Reverse Time Integration Error Evaluation	130
5.6	Application of GOP to ES-1 with $x_o = (-10,-5)^T$	137
5.7	Application of GOP to Higher Order Systems	140
5.8	Application of TOP to ES-1, $x_o = (-10,-5)^T$	143
5.9	Application of TOP to ES-1, $x_o = (5,-5)^T$	145

LIST OF FIGURES

Figure		Page
2.1	Outward Normals for Various Subsets and Sets	11
2.2	Endpoint Terminology and the Reachable Set R(T)	20
2.3	Singular Trajectories	24
3.1	Flow Chart for LOP-DS	43
3.2	Final Time Perturbation Decisions	60
3.3	Final State Perturbations: $\delta x=-cx$	65
3.4	Final State Perturbations near $x^+(T)$	66
3.5	Flow Chart for LOP-CM	76
3.6	Flow Chart for Subalgorithm 4.b	78
4.1	Optima on Concave Boundary Regions	82
4.2	Flats and Corners on Surfaces	87
4.3	Extremum but Not a Local Optimum for Cos γ	91
4.4	Development of an Internal Boundary	92
4.5	A Local Optimum on an Internal Boundary	93
4.6	Reachable Sets with Interior Origins	96
5.1	Example System #2--3rd Order, Nonlinear	106
5.2	Joint VS Single (x_T only) R(T) Perturbations	112
5.3	Example Curvature Values in an Iteration Sequence	114
5.4	Example Curvature Values for R(20), $x_0=(-10,-5)^T$	117
5.5	Comparison of Perturbation Step Size Alternatives	127
5.6	Analog Error Evaluation and Correction Flow Chart	129
5.7	Example Trajectories for a Second-Order System	132

Figure Page

5.8 LOP-CM Convergence for Analog or Digital Integration 133

5.9 Comparison of LOP-CM and LOP-DS Convergence 135

5.10 Reachable Set and Example Iterations for LOP-CM 139

5.11 Example Iterations of GOP for Example Problem 2 141

5.12 R(t) for ES-1, $x_0 = (-10,-5)^T$, Various t 144

5.13 R(t) for ES-1, $x_0 = (5,-5)^T$, Various t 146

CHAPTER 1

INTRODUCTION

The traditional approach to the investigation of control systems has evolved significantly in the past twenty years. The criteria previously used for evaluating systems have changed as have the approaches used to insure that a system has desirable performance characteristics. Concepts such as rise time and steady-state error are being supplanted with performance functional, target set, control constraints, etc. Bode, Nyquist and root-locus procedures are being supplemented by various other theoretical and computational methods. The newer concepts of modern optimal control theory are used in addition to the classical control principles.

Together with this evolution in concepts, terminology and methods, there has been a change in the computational methods utilized to solve problems. These are becoming much more computer-oriented because of the advent of computers which are faster and have greater capacity.

As a result of this increased use of computers in solving optimal control problems, many computational techniques have been presented [T1]. These methods, though related, have some significant differences. Dynamic programming, which was developed by Bellman [B3] and others is one approach. The methods of linear and nonlinear pro-

gramming [H1,Z1] are appropriate for certain static optimization problems. Another major class of procedures includes gradient methods [B7,N1] and search methods [H6] which attempt to "directly" minimize the performance functional.

In 1958 Soviet Union mathematicians led by Pontryagin [P2] presented the important maximum principle which is a necessary condition for optimality. Its impact on control theory has been great. Many computational methods involve the determination of a solution to the two-point boundary value problem generated by the maximum principle [D1,K1,P1]. Closely related to these approaches are those which utilize properties of the reachable system states and the related adjoint system vectors [B2,E1,G1,H5,N2]. These concepts are basic to the work of this thesis.

Initially, optimal control problems and the associated computational methods were applicable only to low order, linear systems for which time optimal or minimum error regulator solutions were desired. Recently, however, emphasis has been placed on more complex systems--systems of higher order and systems which are nonlinear or stochastic in nature. Moreover, an additional objective has been to increase the computation speed for determining an optimal control.

The subject area for this thesis is the computation of optimal controls for nonlinear systems. Within this general

area, several approaches have been suggested. By far the most obvious is a linearization [H7,L1] of the systems such that existing linear methods can be utilized. More subtle linearizations include the methods of successive approximations applied to the system, to the control or to the reachable set [H2,K2].

Other techniques for nonlinear systems include direct, random and pattern search. The method of quasilinearization in which time-independent nonlinear systems become time-dependent linear systems is sometimes employed [B4,B5]. The approach to be used in this thesis combines direct search methods with reachable set techniques. A feature of direct search procedures is general applicability to a wide class of problems. By utilizing the geometrically-oriented reachable set concepts, more efficient computational procedures can be obtained.

The application of reachable set techniques to the optimization of nonlinear systems is relatively new. The resulting reachable sets are usually not convex, hence existing techniques for convex sets must be significantly modified. Another difficulty is the lack of useful examples and computational data for comparison. Thus, one contribution of this thesis is providing data for example nonlinear problems and reachable sets. The minimum error regulator problem is primarily examined but, as has been shown [B1,F1], solution to more complex problems can be based upon the

successive solution of this basic problem.

Often thesis topics are generated by an attempt to find the solution to a specific physical problem or by restricting the system considered to a very specialized class. While such a restriction often yields a definite mathematical structure, it results in a method which is, as expected, restricted in the area of application. On the contrary, the approach of this thesis is general in the physical applications which can be treated and in the rather general form of the nonlinear equations. It should be noted that such a general approach does not preclude specific applications--as is evident from the examples which are included.

This dissertation may be outlined in the following manner. Chapter 2 includes comments relative to notation and basic definitions. The systems and problems to be considered are also defined. The important maximum principle is introduced as are the concepts of normality, extremality and optimality. Also in Chapter 2 the reachable set is defined and related properties are given. Finally, the modified (for nonconvex reachable sets) minimum distance problem (MP) and the associated local optimum problem (LP) are defined.

Reachable set computational methods are discussed in Chapter 3, particularly as they apply to problems involving nonlinear systems. Preparatory to the introduction of an algorithm to solve the local minimum distance problem, several error functions are developed. They emphasize the

collinearity of the optimum final state and adjoint vectors. A direct search algorithm is given which utilizes these error functions to evaluate convergence.

Also in Chapter 3 various concepts from differential geometry are presented and are used in the development of a geometric search procedure for computing optima. The previously defined error functions are shown to decrease monotonically along paths defined on the boundary of the reachable set. Various algorithm alternatives are discussed as they relate to effective boundary path selection. Concluding Chapter 3 is an algorithm based on geometric search, to determine the optimum in a locally convex subset of the reachable set.

In Chapter 4, the global optimization problem is considered. In this case there may be many local optima. Related to the global problem are several special cases which are also discussed. A global minimum distance procedure is then presented and its application to time optimal control problems is demonstrated.

In Chapter 5 example nonlinear problems are given. Reachable sets and typical trajectories are also presented. Within the algorithms, alternative choices are compared and general computational results are given for the local and the global problem. Finally, conclusions relating to the computational results and the developments of previous chapters are discussed.

CHAPTER 2

PROBLEM FORMULATION AND REACHABLE SET CONCEPTS

A discussion of systems and optimal control problems can be approached in any one of several ways. One approach is to start with a basic, simple system and later extend the discussion to more general systems. In this chapter, however, the more general formulation is first introduced and then specialized as needed. System and control assumptions necessary for future development are also introduced.

Knowledge of many common optimal control concepts, such as target set, performance functional, etc., is assumed and only discussed as deemed necessary or instructive. The reachable set is defined and important related results are summarized. In the last section of this chapter, an introduction is given to the basic problems to be solved.

2.1 Notation and Terminology

Let E^n denote n-dimensional Euclidean space. No special notation is used to distinguish between scalars and vectors. Most symbols in this thesis are vectors (for example, x, u and p); scalars are so designated as they are introduced. The components of any vector are denoted by subscripts, namely, x_i, $i = 1, \cdots, n$.

Let t be a scalar, time. Let $[t_o, T]$ denote a general time interval where t_o is initial time and T is final time. Where any vector, for example x, is a function, x(t), of

time, the following abbreviated forms are frequently used:

$$x(t) = x_t, \tag{2.1}$$

$$x(t_o) = x_o \tag{2.2}$$

and

$$x(T) = x_T. \tag{2.3}$$

Where both component subscripts and the time subscripts of Equations 2.1 through 2.3 are simultaneously used, the component subscripts are placed first and the time subscripts last (for example, x_{2T}). To represent the vector function $x(t)$ over an interval of time, $x(\cdot)$ is used. The time derivative of $x(t)$, dx/dt, is denoted $\dot{x}(t)$ and $\partial x/\partial t$ is used to denote the partial derivative.

Let $|a|$ denote the absolute value of the scalar a. Let $<x, y>$ denote the inner product of two vectors, x and y:

$$<x, y> = \sum_{i=1}^{n} x_i y_i. \tag{2.4}$$

Let $||x||$ denote the Euclidean norm of x:

$$||x|| = (<x, x>)^{\frac{1}{2}}, \tag{2.5}$$

and thus $||x||$ represents the length of the vector x. Since x may be viewed as either the vector x or the point x in E^n, $||x||$ also represents the distance of the point x to the origin. Similarly, $||x-y||$ denotes the distance between the points x and y.

Superscripts are used to denote iteration indices, namely x^i denotes the x vector for the i^{th} iteration. To

denote optima, superscripts are also used: x^+ denotes a local optimum and x^* denotes a global optimum. Optima are also indexed, if necessary, using pre-superscripts. For example, ${}^i x^+$ represents the i^{th} local optimum.

Standard set notation is used ($\bigcap$, $\subset$, $\in$, etc.) with one exception: brackets are used to define a set. For example,

$$Y = [y \in E^n : \|y\| < 1], \tag{2.6}$$

denotes the set of all y in E^n such that norm y is less than 1. The boundary of a set Y is denoted ∂Y, the complement is denoted Y^c and the closure is denoted $\overline{Y}$. A neighborhood, or open sphere, with center x and radius ϵ, is denoted $N(x;\epsilon)$:

$$N(x;\epsilon) = [y : \|y-x\| < \epsilon]. \tag{2.7}$$

A set K in E^n is <u>convex</u> if for any x_1 and any x_2 in K, the point $x_3 = \pi x_1 + (1-\pi)x_2$, $0 \leq \pi \leq 1$, x_3 belongs to K. A set K in E^n is <u>strictly convex</u> if for any x_1 and x_2 in K, the point $x_3 = \pi x_1 + (1-\pi)x_2$, $0 < \pi < 1$, is in K but not on ∂K. If $x \in R$ and if

$$R(x;\epsilon) = N(x;\epsilon) \bigcap R \tag{2.8}$$

is convex, then $R(x;\epsilon)$ is said to be a <u>convex subset of the set</u> R <u>at</u> x (R may be nonconvex).

The boundary ∂K of a convex set K is a <u>convex surface</u>. If $K = R(x;\epsilon)$, then

$$R_{vx}(x;\epsilon) = [r \in R(x;\epsilon) : r \in \partial R] \tag{2.9}$$

is called a <u>locally convex surface at</u> x. If $N(x;\epsilon) \bigcap \overline{R^c}$

is convex, then

$$R_{cv}(x;\varepsilon) = [r \in N(x;\varepsilon) \cap \overline{R^c} : r \in \partial R], \qquad (2.10)$$

is said to be a locally concave surface at x. If $N(x;\varepsilon) \cap \partial R$ is neither a locally convex surface nor a locally concave surface, then it is said to be a mixed surface at x.

The hyperplane (dimension n-1) through x with normal p is defined as

$$Q(x;p) = [y \in E^n : \langle y, p \rangle = \langle x, p \rangle], \; p \neq 0. \qquad (2.11)$$

The closed half-space bounded by Q(x;p) with outward normal p is defined as:

$$Q^-(x;p) = [y \in E^n : \langle y, p \rangle \leq \langle x, p \rangle], \; p \neq 0. \qquad (2.12)$$

Let K be a closed, convex set in E^n. A hyperplane such that $K \cap Q(x;p)$ is nonempty and $K \subset Q^-(x;p)$ is called a support hyperplane to K with outward normal p.

DEFINITION 2.1 Let $p \in E^n$, let $R = R(T) \subset E^n$ be a nonempty, compact, reachable set (defined in Section 2.5) and let $x \in E^n$ be such that $x \in \partial R$.

Case 1 (convex surface): If R(x;ε) for some $\varepsilon > 0$ is a convex set, then p is an outward normal to R at x, if p is the outward normal to a support hyperplane to the convex set R(x;ε) at x.

Case 2 (concave surface): If $N(x;\varepsilon) \cap \overline{R^c}$ for some $\varepsilon > 0$ is a convex set, then p is an outward normal to R at x if -p is the outward normal to a support hyperplane to the convex set $N(x;\varepsilon) \cap \overline{R^c}$ at x.

Case 3 (mixed surface): If $N(x;\epsilon) \cap \partial R$ is a mixed surface, then p is an outward normal to R at x if p = p(T) is a final adjoint corresponding to the extremal endpoint x(T) = x (extremal endpoints are subsequently defined in Definition 2.4).

Note that there may be many such p for one x or many x for one p. See Figure 2.1 for examples of these cases.

The scalar signum function, sgn, is defined by:

$$\text{sgn}(a) = 1 \qquad a > 0, \tag{2.13}$$

$$|\text{sgn}(a)| \leq 1 \qquad a = 0, \tag{2.14}$$

$$\text{sgn}(a) = -1 \qquad a < 0. \tag{2.15}$$

The vector signum function is denoted SGN and is defined as:

$$\text{SGN}(x) = \begin{bmatrix} \text{sgn}(x_1) \\ \vdots \\ \text{sgn}(x_n) \end{bmatrix}. \tag{2.16}$$

2.2 System Definition

Consider a system whose state at any time t is described by the solution x(t), $t_0 \leq t \leq T$, to the following nonhomogeneous, nonlinear, vector differential equation:

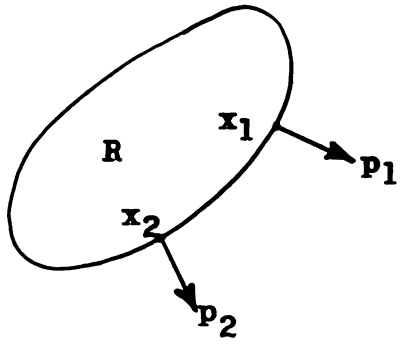
$$\dot{x}(t) = \hat{f}(t,x(t),v(t),w(t)), \quad x(t_0) = x_0, \tag{2.17}$$

where:

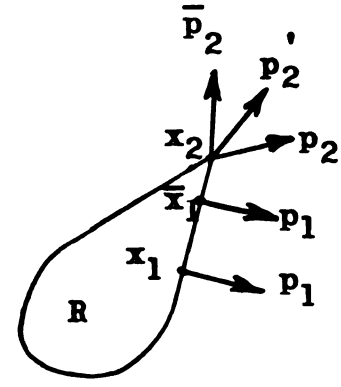
t represents the independent variable, time,

$x(t) \in E^n$ is the state vector,

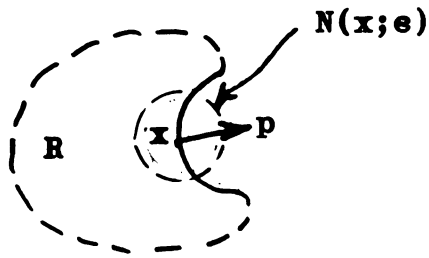
$\dot{x}(t)$ is its time derivative,



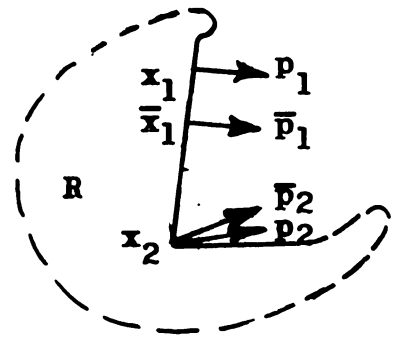
a. Case 1 - Strictly Convex



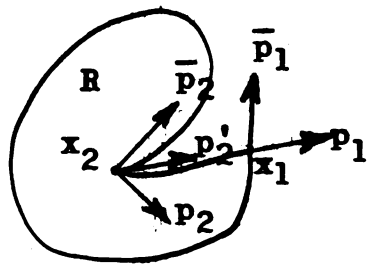
b. Case 1 - Convex but has "flats" and a "corner" (See Section 4.2)



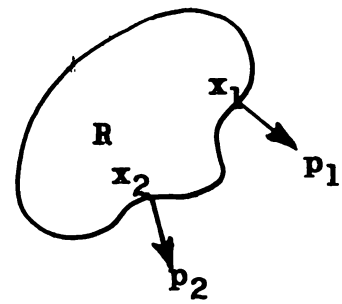
c. Case 2 - Concave



d. Case 2 - Concave with "flats" and a "corner"



e. Case 3 - Mixed with corners



f. Case 3 - Mixed

FIGURE 2.1 Outward Normals for Various Subsets and Sets

x_o is the initial state

$v(t) \in E^m$ is the control vector defined on a compact interval of E^1, namely, $I = [t_o, T]$,

$w(t) \in E^q$ is the parameter vector defined on I,

$\hat{f}(\cdot,\cdot,\cdot,\cdot)$ is an n-dimensional vector function defined on $I \times E^n \times E^m \times E^q$.

In the development to follow, v(t) and w(t) are treated as one composite control vector, u(t), i.e.,

$$u(t) = \begin{bmatrix} v(t) \\ w(t) \end{bmatrix} \tag{2.18}$$

where u(t) is an m + q = r-dimensional vector defined on I. Thus Equation 2.17 becomes:

$$\dot{x}(t) = f(t,x(t),u(t)),\ x(t_o) = x_o, \tag{2.19}$$

where $f(\cdot,\cdot,\cdot)$ is an n-dimensional vector function defined on $I \times E^n \times E^r$. Unless otherwise specified, the term "control" will hereafter refer to the composite vector, u(t). Let U be a nonempty compact set in E^r. A measurable function $u(\cdot)$, defined on I with range space U is said to be an admissible control and F is used to denote the family of admissible controls.

In order that the solution exists, is unique and continuous [A1] for all $u(\cdot)$ in F, additional assumptions are introduced:

H1) $f(t,x(t),u(t))$ is continuous on $I \times \theta \times U$, where θ is a nonempty open set in E^n,

and

H2) for any x in θ, u in U and t in I, $f(t,x,u) \in C^1$, (i.e., the first partial derivative with respect to x is also continuous).

For linear systems it can be shown that a unique, global solution exists, but for nonlinear systems only a local (unique) solution can be proven. Note that the assumption of measurability for $u(\cdot)$ is often replaced with the stronger assumption that $u(\cdot)$ is piecewise continuous. Note also that H2) is often replaced with the Lipschitz Condition:

H2') There exists an integrable function $K(\cdot)$ on I such that:

$$\| f(t,x,u)-f(t,y,u) \| \leq K(t) \| x-y \|, \qquad (2.20)$$

for any x and y in θ, u in U and t in I.

The stronger assumption H2) is used since it is necessary for proving convergence to a local optimum.

Additional assumptions are required for some of the results. In another section of this chapter the reachable set R(t) is defined. To guarantee that R(t) is compact and varies continuously with time (hence to guarantee the general existence of the optimal control) the following two conditions (boundedness and convexity) are necessary [L2]:

H3) $\| x(t) \| \leq B$ for any t in I, any $u(\cdot)$ in F (uniform bound).

H4) $V(t,x) = [f(t,x,u) : u \in U]$ is convex for each fixed x and t.

The assumptions listed above are not excessively restrictive. For instance, consider one of the most often used families of admissible controls, F_1, corresponding to

the range space U_1 in which each component of $u(\cdot)$ has absolute value less than or equal to 1 on I:

$$U_1 = [y \in E^r : |y_i| \leq 1, \ i = 1,\cdots,r]. \qquad (2.21)$$

Certainly this control satisfies the requirement that U be compact and simplies the fulfillment of H4). Further, if the control is a vector signum function of a continuous argument, then $u(\cdot)$ is certainly measurable.

2.3 Optimal Control Problem Definition

In addition to the system description and the class of available control functions, the optimal control problem also includes a prescribed set of conditions (final and sometimes initial) and a performance functional to be optimized. The initial conditions for time and state were previously given as t_o and $x(t_o) = x_o$, respectively. The final conditions are often determined by a target set G for the problem. For example, $x_T \in G(T)$ may be required where $G(\cdot)$ is a nonempty set in E^n for each $t \in I$. Thus the general optimal control problem is as follows:

PROBLEM 2.1: Given: the system (Equation 2.19), the class F of admissible controllers, and the performance functional

$$J(t_o,T,x,u) = K(T,x_T) + \int_{t_o}^{T} L(s,x,u)\, ds, \qquad (2.22)$$

where $K(\cdot,\cdot)$ is a continuous function from $E^1 \times \theta$ to E^1, and $L(\cdot,\cdot,\cdot)$ is a continuous function from $I \times \theta \times U$ to E^1.

Find: a control function, $u^*(\cdot)$ in F which optimizes (maximizes or minimizes) the performance functional while satisfying Equation 2.19 and the prescribed set of conditions. It should be emphasized that an optimal control $u^*(\cdot)$ which generates an optimal trajectory $x^*(\cdot)$ need not be unique.

2.4 Minimum Regulator Problem

In preparation for more complex problems, initial attention is given to the minimum regulator problem. For this problem, given a specific final time T, a control which drives the state x_T closest to the origin is an optimal control. Specifically, the problem is defined as follows:

PROBLEM 2.2: Given: the system (Equation 2.19), the class F of admissible control functions, final time T and the performance functional

$$J(t_0,T,x,u) = K(x_T) = \| x_T \|, \qquad (2.23)$$

Find: a control function $u^*(\cdot)$ in F which minimizes $K(x_T)$ while satisfying Equation 2.19.

2.5 Reachable Set Definition

In much of the discussion to follow, the concept of the reachable set is important. For example, many system characteristics are directly related to properties of the reachable set. In fact, the search for a solution to Problem 2.2 may be viewed as a search along the boundary of a reachable set.

For each $u(\cdot)$ belonging to F there corresponds a trajectory, $x_u(\cdot)$ (a solution to Equation 2.19), which originates at x_o and terminates at $x_u(T)$.

DEFINITION 2.2 The reachable (attainable or obtainable) set at time $t \in I$, denoted $R(t)$, is the set of all states which can be reached at time t utilizing admissible controls, i.e.,

$$R(t) = [x \in E^n : x = x_u(t),\ u(\cdot) \in F]. \qquad (2.24)$$

Let $R(\cdot)$ designate the reachable set as a function of time on the interval I. As previously indicated, $\partial R(t)$ represents the boundary of the reachable set at time t and let $\partial R(\cdot)$ designate the boundary of the reachable set on the time interval I. For most problems it is nearly impossible to give an explicit formula for $\partial R(t)$. Certain general properties of $R(t)$ are known and are described in Section 2.8.

2.6 Pontryagin's Maximum Principle

Consider now, Pontryagin's Maximum Principle [P2] and its relationship to the optimal control problems previously introduced. The statement of the maximum principle varies with the nature of the problem, specifically with the nature of the prescribed conditions and the performance functional. There are, however, several essential concepts in the description of the maximum principle regardless of the nature of the problem. These include the Hamiltonian function:

$$H(t,x(t),u(t),p(t)) = L(t,x(t),u(t)) + \langle p(t), f(t,x(t),u(t))\rangle \tag{2.25}$$

with the associated Hamiltonian differential system:

$$\dot{x}(t) = \frac{\partial H(t,x(t),u(t),p(t))}{\partial p} \tag{2.26}$$

$$\dot{p}(t) = -\frac{\partial H(t,x(t),u(t),p(t))}{\partial x}, \tag{2.27}$$

where p(t) is a nontrivial solution of the differential systems called the adjoint or the costate response. In the event that L(t,x(t),u(t)) is independent of x(t) (for instance, constant, as in the case of time optimal control problems), the adjoint equation (Equation 2.27) becomes:

$$\dot{p}(t) = -\frac{\partial f^T(t,x(t),u(t))}{\partial x}\, p(t) \tag{2.28}$$

because the partial derivative of L is zero. In fact, this is the same adjoint equation which one would obtain if L(t,x(t),u(t)) = 0, i.e. if the Hamiltonian function were "unaugmented":

$$H(t,x(t),u(t)) = \langle p(t), f(t,x(t),u(t))\rangle . \tag{2.29}$$

Utilizing this unaugmented Hamiltonian function and the adjoint equation (Equation 2.28) above, the following theorem results [L2]:

THEOREM 2.1 Consider the process given in Equation 2.19 with assumptions H1) through H3). Let u'(•) belong to F and have the response x'(•) with x'(T) on the boundary of the reachable set, R(T).

Then there exists a nontrivial adjoint response p'(•) of Equation 2.28 such that the maximum condition holds almost

everywhere:

$$H(t,x'(t),u'(t),p'(t)) = M(t,x'(t),p'(t)), \qquad (2.30)$$

where

$$M(t,x(t),p(t)) = \max_{y \in U} H(t,x(t),y,p(t)). \qquad (2.31)$$

This theorem is proved for autonomous systems in Lee and Markus [L2], page 254 and is extended to nonautonomous systems on page 318 and following pages.

Before discussing this theorem in relation to the optimal control, consider the following theorem which is a general existence theorem for optimal controllers [L2].

THEOREM 2.2 Consider Problem 2.1. Let the target set $G(t)$ in E^n be a nonempty, compact set which varies continuously for all t in I. Let the family of admissible controllers, F, be nonempty. Further, let Hypotheses H1) and H2) apply.
Then there exists an optimal control, $u^*(\cdot)$, in F, on I minimizing $J(t_o,T,x_u,u)$.

Before relating the results of the preceding two theorems, consider the following terminology.

DEFINITION 2.3 Controls which satisfy the maximum principle (Equation 2.30) are called maximal controls. The resulting trajectories are maximal trajectories and terminate at maximal endpoints.

DEFINTION 2.4 Controls which result in trajectories terminating on the boundary of the reachable set are called extremal controls and the corresponding trajectories are

extremal trajectories. The boundary of the reachable set thus consists of extremal endpoints.

DEFINITION 2.5 Controls which minimize $J(t_o,T,x_u,u)$, as stated in Theorem 2.2, are optimal controls. The corresponding trajectories are called optimal trajectories and terminate at optimal endpoints.

Theorem 2.1 asserts that state trajectories terminate on the boundary of the reachable set (extremal trajectories) only if the Hamiltonian is maximized. For nonlinear systems trajectories corresponding to controls which satisfy the maximum principle (maximal controls) do not necessarily terminate on the boundary of the reachable set. For linear systems, however, maximal controls are also extremal controls.

A third important and well-known theorem is the following which asserts that all optimal controls are extremal controls.

THEOREM 2.3 Let the hypotheses of Theorem 2.2 apply. Let $u^*(\cdot)$ be an admissible control with corresponding trajectory $x^*(\cdot)$ from x_o to $G(T)$. Then the control $u^*(\cdot)$ is optimal only if it is extremal.

This theorem is proven in Lee and Markus [L2,page 310] and in Athans and Falb [A1, page 305], among others.

Theorem 2.2 states the existence of an optimal control but does not guarantee that such a control is unique. Because of Theorem 2.3, Theorem 2.2 also indicates the

existence of at least one extremal control. In general, of course, there are many extremal controls. It is also possible that several distinct extremal controls could generate the same extremal endpoint. The terminology relating optimal, extremal and maximal endpoints is illustrated in Figure 2.2.

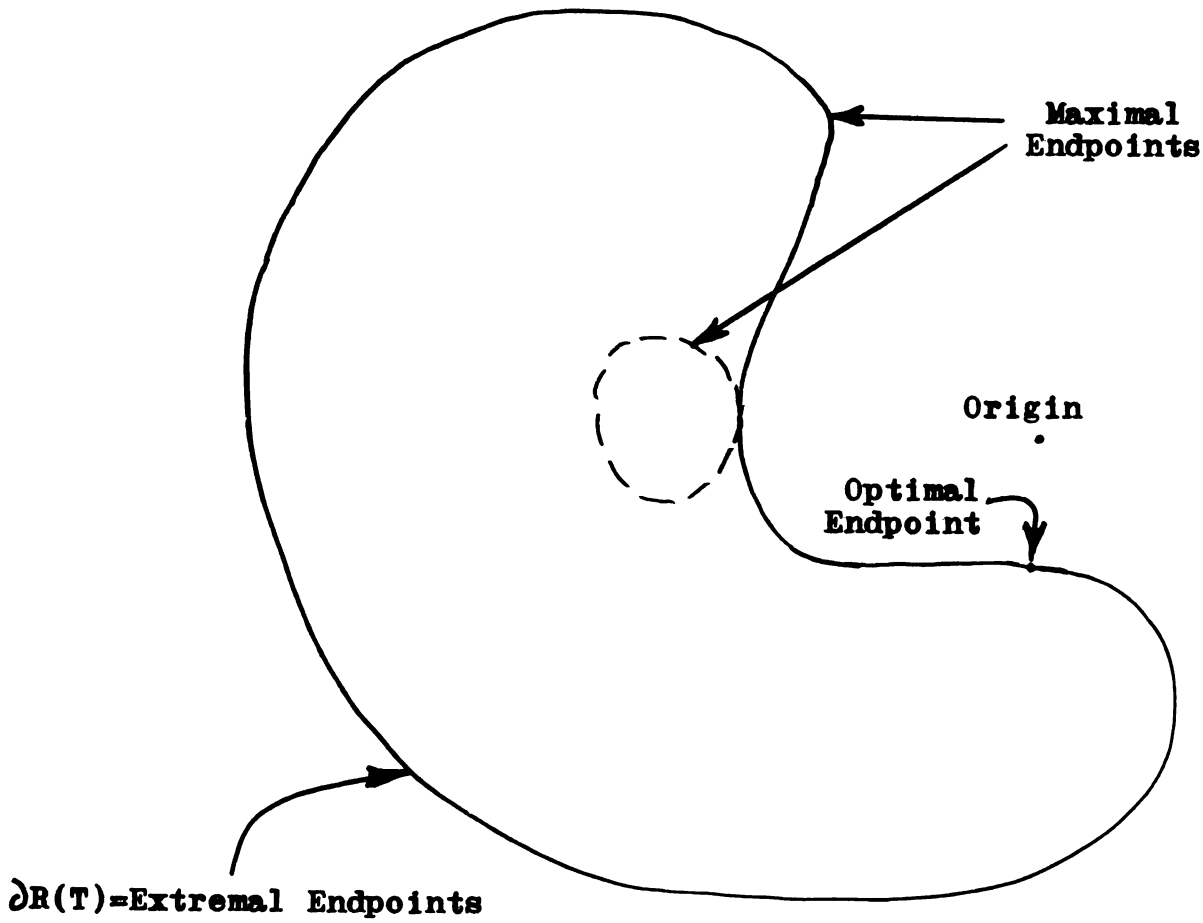


FIGURE 2.2 Endpoint Terminology and the Reachable Set R(T)

Since the reachable set represents all possible trajectory endpoints, one possible means of locating an optimal trajectory would be to examine the entire reachable set--a prohibitive procedure in the case of higher order systems. It should be noted, however, that it is possible to examine the boundary points of the reachable set by considering all maximal endpoints. Certainly this considerably reduces the computations necessary to determine an optimal control, but for higher order systems, such an examination would still include a prohibitive number of possible trajectory endpoints. It should also be repeated that interior points of R(T) might also exist as maximal endpoints (See Figure 2.2).

Finally, several important additional facts relating to extremal controls should be stated. The adjoint or costate variable p(T) is an outward normal to the reachable set R(T) at x(T) [L2]. This fact is important in the development of several error functions later to be considered. It is also equivalent to Equation 2.30. In addition it is related to the transversality condition which is an additional necessary condition in the event that the target set G(t) is a convex set. The transversality condition states that the final adjoint p(T) is normal (inward) to the target set at x(T).

2.7 Normality

The concept of normality is briefly discussed in this section due to its close relationship to the character of the reachable set and because of its effect upon the availability and difficulty in computing the optimal control. It has previously been stated that extremal controls belong to the more general class of maximal controls. Hence extremal controls must maximize the Hamiltonian. Such maximization would be straightforward if for each adjoint variable there exists exactly one corresponding control. Unfortunately this is not always the case. In fact, there may be several (say two: $\bar{u}_1(\cdot)$ and $u_1'(\cdot)$), corresponding to the same adjoint $p_1(\cdot)$, which maximize the Hamiltonian for an arbitrary time interval $I_s \subset I$.

Were the two controls equal almost everywhere:

$$\bar{u}_1(t) = u_1'(t) \quad \text{a.e.}, \tag{2.32}$$

no singularity would result. If, however,

$$\bar{u}_1(t) \neq u_1'(t), \quad t \in I_s, \tag{2.33}$$

where I_s is finite or countably infinite, then a singularity occurs.

DEFINITION 2.6 If for any solution to Equation 2.28, there are two or more controls ($u_1(t) \neq u_2(t)$, $t \in I_s$, I_s finite or countably infinite) which differ yet which maximize the Hamiltonian on I_s, then the problem is singular.

DEFINITION 2.7 If for each solution of the adjoint equation (Equation 2.28) there is one unique maximal

control then the system (problem) is normal.

If a problem is singular, optimal controls may exist, but may not be uniquely defined on some interval I_s. Such nonuniqueness may have a variety of effects on the reachable set. For linear systems, normality is equivalent to strict convexity of the reachable set, while singularity causes "flats" on the boundary of R(t). With a singularity, i.e. $u_1(t) \neq u_2(t)$, t in I_s, it is still possible that $x_1(T) = x_2(T)$ (that the same maximal point is attained).

For more insight into this problem, consider the following nonlinear state equation in which the control belongs to the control family F_1 and is separable:

$$\dot{x}(t) = A(t,x(t)) + B(t,x(t))\, u(t). \qquad (2.34)$$

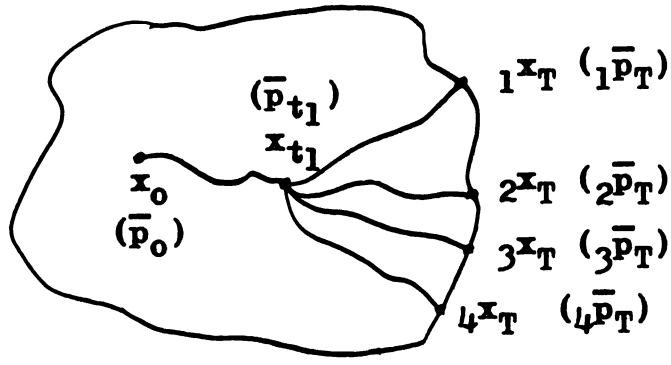
The corresponding unaugmented Hamiltonian function is:

$$H(t,x(t),u(t),p(t)) = \langle A(t,x(t)) , p(t) \rangle + \langle B(t,x(t))u(t) , p(t) \rangle. \qquad (2.35)$$

At any instant in time, the maximum with respect to u(t) is attainted if

$$u(t) = SGN[\, B^T(t,x(t))\, p(t)\,]. \qquad (2.36)$$

If, however, any component of $B^T(\cdot,x(\cdot))p(t)$ is zero for a finite interval of time, the corresponding control component is indeterminate, taking on any value or variation allowed for F_1. Each of these various controls may, in turn, lead to different maximal endpoints. This possibility is illustrated in Figure 2.3. Note that even though the adjoint trajectories all start at $\bar{p}_0$, the final adjoints may vary



( ) indicates the value of the costate corresponding to the state vector.

FIGURE 2.3 Singular Trajectories

greatly since the partial derivative in Equation 2.28 is control dependent.

According to Theorem 2.3, an optimal control is one of the extremal controls. Associated with each of the extremal controls is an extremal endpoint which lies on the boundary of the reachable set, and a final value for the adjoint variable which is normal to the boundary of the reachable set at the extremal endpoint. For such a control to be extremal, it must also be maximal. Note that once the initial state and initial adjoint have been specified, the final state and final adjoint are also specified (through Equation 2.30, the maximum condition, and integration of Equations 2.19 and 2.28) unless the problem is singular. In this case, a whole section of the boundary of the reachable set might correspond to the same initial

adjoint (See Figure 2.3), but the extremal controls differ over the singularity interval, I_s.

While such a "singularity gap" can occur and would complicate the determination of the optimal control, it would be readily recognizable. That is, any procedure yielding a series of extremal endpoints would encounter a "jump" between successive final states when the system singularity is encountered. Finally, it is possible that the singularity may not affect the determination of the optimal control--all extremal endpoints in a large region around the optimal endpoint are the result of normal controls. As a result of the above consideration, normality is not a requirement imposed upon the problems to be considered. Even if it were desired, proving normality for a general class of nonlinear systems would be extremely difficult, if possible at all.

2.8 Properties of the Reachable Set

The concept of the reachable set, R(t), was needed for earlier discussion, hence was defined in Section 2.5. It is the purpose of this section to list and discuss some of the important properties of R(t). Necessary assumptions for proving these properties have already been given. As expected, fewer results are available in nonlinear system study than in the study of linear systems. Reachable sets resulting from linear systems can be proved to be continuous, compact, convex and to have a computable contact

function [B1,L2]. For nonlinear systems, however, the reachable set is generally not convex (although there may be locally convex regions) and the computation of a contact function is complicated. For most nonlinear systems, however, compactness and continuity have been proven [L2,R1]:

THEOREM 2.4 Consider the process given in Equation 2.19 with assumptions H1) through H4). Let F be defined as in Section 2.2. Then R(t) is compact and varies continuously with time on I.

Both compactness and continuity are important in proving the existence of a unique optimal control. For instance, if R(t) is not continuous then a unique time optimal control is not guaranteed for time optimal control problems.

Another important observation is given in the following theorem [R1]:

THEOREM 2.5 Let R(t) be the reachable set for the process given in Equation 2.19 with assumptions H1) and H2). If $x_T \in \partial R(T)$ then $x_t \in \partial R(t)$ for any t in I. Stated differently, all points on an extremal trajectory will belong to the boundary of the reachable set of corresponding time, t.

Several other important properties of the boundary of the reachable set have already been given, including:

1) $x_T \in \partial R(T)$ (extremal trajectory) $\Longrightarrow$ x_T is a maximal endpoint.

2) x_T c R(T) $\Longrightarrow$ p_T (corresponding to x_T) is normal to the boundary of R(t) at x_T.

2.9 Problem Statements

Gilbert and Barr have shown that a useful approach to optimal control problems centers around the solution or repetitive solutions to the following basic problem (BP) [G1]:

PROBLEM 2.3 (BP) Given: K, a compact, convex set in E^n; Find: A point x^* e K such that $||x^*|| = \min_{x \in K} ||x||$.

In the case of linear systems, an obvious candidate for K is the reachable set R(T) and the problem essentially becomes a minimum error regulator problem. Assuming that the origin is external to R(T), the solution lies on the boundary of the reachable set. Since R(T) is not, in general, convex for nonlinear systems, the following modified problem (MP) must be solved:

PROBLEM 2.4 (MP) Given: R, a compact set in E^n; Find: a point r^* e R, such that $||r^*|| = \min_{r \in R} ||r||$.

As part of this modified problem, it is possible that several subproblems similar to Problem 2.3 (BP) but local in nature must be solved. Consider the local problem (LP):

PROBLEM 2.5 (LP) Given: ${}^iR$, a convex subset of R, a compact set in E^n; Find: a point ${}^ir^+$ e ${}^iR$ such that $||{}^ir^+|| = \min_{{}^ir \in {}^iR} ||{}^ir||$.

Since R and ${}^{i}R$ are defined to be compact, and since $\|r\|$ is a continuous function of r, solutions always exist to Problems 2.4 and 2.5. It may be emphasized that R is not necessarily convex. In addition, the following properties can be shown [B1,G1] for LP:

1) ${}^{i}r^{+}$ is unique,

2) $\|{}^{i}r^{+}\| = 0$ if and only if $0 \in {}^{i}R$.

3) For $\|{}^{i}r^{+}\| > 0$, ${}^{i}r^{+} \in \partial {}^{i}R$.

For Problem 2.4 (MP), properties 2) and 3) are true but r^{*} is not necessarily unique. In the next chapter, one additional important property is proven--the collinearity of r^{*} (MP) (or ${}^{i}r^{+}$ for LP) with the normal to ∂R (or $\partial {}^{i}R$) at r^{*} (${}^{i}r^{+}$).

CHAPTER 3

THE LOCAL OPTIMUM PROCEDURE

In this chapter various iterative methods and their limitations are discussed. Consideration is given to the important function which the initial adjoint has in the determination of the extremal endpoint (for a given x_o) and the associated normals to the boundary of the reachable set (final adjoints). It is shown that a straightforward solution to LP can be implemented by a direct search on the initial adjoints.

Utilizing properties of reachable sets and principles from differential geometry, a more sophisticated iterative procedure is developed for solution of LP. In as much as reachable sets for nonlinear problems are generally not explicitly defined, part of the development of this algorithm is based on geometrical considerations of the reachable set and perturbation analysis.

Special consideration is given to the choice of error functions which correctly indicate a solution to LP and which are based on significant properties of reachable sets. Finally, convergence is considered and the solution algorithm is given. Special problems arising as a result of nonlinearity will be deferred until Chapter 4.

3.1 A Discussion of Iterative Methods

The set R of the modified problem (MP) given in Chapter 2 is not convex, hence the global optimum (r^*, the point closest to the origin) is generally difficult to compute. Methods developed for obtaining such an optimum depend on the nature of the system, of the admissible controls and of the reachable set. For linear systems, hence convex reachable sets (assuming certain conditions on the control, etc.), the iterative procedures given by Neudstadt, Gilbert and Barr [N1,G1 and B1] are effective in solving the problem.

For nonlinear systems, however, convergence to the global optimum is not guaranteed. If this were the only handicap, such methods would still have direct application in determining local optima. Possibly a linearization of the system or a "convexization" of the reachable set could be used to implement these methods. In the event, however, that one desires to retain the nonlinear equations describing the system, difficulties are encountered in applying the above-mentioned methods. These methods require the determination of a contact point corresponding to each final costate selected [B2]. A contact point is any point in $\partial R(T)$ which maximizes the projection onto the final costate. In typical methods for linear systems, this determination is relatively easy (considering present-day computational equipment) to implement by the following steps:

1) Consider the desired outward normal which is in the direction of the final value of the costate.

2) Since Equation 2.19 reduces to:

$$\dot{x}(t) = A(t)x(t) + B(t)u(t), \tag{3.1}$$

where A(t) is an n x n matrix and B(t) is an n x r matrix, the costate differential equation is also linear and homogeneous:

$$\dot{p}(t) = -A^T(t)\, p(t). \tag{3.2}$$

Thus, given one boundary point, p_T, p(•) is defined on I.

3) Once p(•) is defined, u(•) (a maximal control) and x(•) are also defined by Equations 2.30 and 3.1. Hence x_T(the contact point) is computable.

Not only is the resulting set usually nonconvex for nonlinear systems, but also the above listed method to solve for contact points is not directly applicable. An iterative method to solve MP would thus have <u>two</u> levels of iteration instead of one, with the additional level resulting from the difficulty in obtaining contact points.

To demonstrate this, consider the adjoint equation for nonlinear systems (Equation 2.28):

$$\dot{p}(t) = -\frac{\partial f^T(t,x(t),u(t))}{\partial x}\, p(t). \tag{3.3}$$

Since contact points are on the boundary of R(T), maximal controls are employed. Certainly the determination of p(•), u(•) and hence x(•) is possible once p_0 is known; but it is not possible to solve for p(•) from the final adjoint (as suggested above) since the adjoint differential equation also depends on the yet unknown x(•). In summary, two levels of iteration would be necessary:

1) An iterative solution of the two-point boundary value problem to determine each (p_0, x_T) pair corresponding to each given (x_0, p_T) pair.

2) Some type of iteration (such as the Basic Iterative or Improved Iterative Procedure of Gibert and Barr, respectively [G1 and B1], on x_T and p_T to determine x^* as defined in LP.

3.2 Initial Costate Iterations and Error Functions

Consideration of the above discussion suggested to the author that an iterative method based on the initial costate would be simple, most direct, yet effective. Since x_0 is given and maximal controls are utilized, p_0 is sufficent to yield $x(\cdot)$ and $p(\cdot)$. Starting with an arbitrary initial adjoint, a sequence of initial adjoints can be determined such that the resulting sequence of final states converges to a local optimum. An evaluation of whether the local optimum is being approached or has been reached is based on error functions to be later discussed.

Both digital and hybrid computation methods are feasible; Each approach has advantages and disadvantages. Numerical integration methods (particularly for ill-behaved nonlinear systems) are sometimes slow, but consistent and accurate if sufficient computation time is available. Hybrid computation techniques have improved significantly in recent years and thus represent another effective approach. Since analog components are used to determine $x(\cdot)$ and $p(\cdot)$, hybrid computations are usually faster. On the other hand, less accuracy and consistency can be expected from hybrid

computers. Overall control of the iterations, of course, is a digital function for either approach, as is the evaluation of the error functions and selection of the next initial adjoint. In this thesis, both hybrid and complete digital computation are employed. Based on these computations, conclusions comparing their relative value are given in Chapter 5.

The choice of the error function is critical and determines the convergence and efficiency of the method. Consider the local problem (LP); one obvious error function would simply be:

$$E_1(p_o) = \| x_T \|. \tag{3.4}$$

If a small change in p_o results in a correspondingly small change in x_T (proof to be given later), a method using E_1 would generally converge if the step size of changes in p_o were reasonably chosen.

The error function E_1 is certainly not the only available test for optimality. The following development, based upon a theorem proven for convex sets provides an effective error function, E_2, which is later used in conjunction with E_1. Since the reachable set is not, in general, convex, for nonlinear problems, it is necessary to select a subset of the reachable set in the following manner.

Consider the compact, but not necessarily convex, set R(T) in E^n. Let x_T be on the boundary of R(T) and let R(T) be convex in a local region of x_T. That is, $R(x_T;e)$ defined

by

$$R(x_T;e) = N(x_T;e) \cap R(T), \tag{3.5}$$

is convex. Then

$$M(x_T,p) = [y \text{ in } E^n : \langle p , y \rangle = c], \tag{3.6}$$

where $c = \langle x_T , p \rangle$ is a scalar constant, can be called a local support hyperplane of R(T) at x_T. The following theorem can be applied in this case with $K = R(x_T;e)$.

THEOREM 3.1 Let K be a compact, convex set in E^n, $0 \notin K$; Then $x_T \in \partial K$ is the closest point of K to the origin:

$$\| x_T \| < \| x \|, \text{ for any } x \in K, x \neq x_T \tag{3.7}$$

if and only if there exists a support hyperplane $M(x_T,p_T)$ of K through the point x_T such that $-x_T$ is normal to $M(x_T,p_T)$, i.e.,

$$M(x_T,p_T) = M(x_T,-x_T). \tag{3.8}$$

Or, stated differently, x_T and p_T are collinear and oppositely directed.

Proof: This theorem has been proven by Gilbert [G1]; however, a different proof which provides additional insight is presented here. First sufficiency is shown.

If $M(x_T,p_T)$ is a support hyperplane to K, then either

$$\langle p_T , x \rangle \geq c \text{ for any } x \text{ in } K, \tag{3.9}$$

or

$$\langle p_T , x \rangle \leq c \text{ for any } x \text{ in } K. \tag{3.10}$$

Since $\langle p_T , x \rangle = c$ (x_T is on the hyperplane), then either

$$\langle p_T , (x-x_T) \rangle \geq 0 \text{ for any } x \text{ in } K, \tag{3.11}$$

or

$$\langle p_T , (x-x_T) \rangle \leq 0 \quad \text{for any } x \text{ in } K. \tag{3.12}$$

Thus if $-x_T$ is to be normal to the support hyperplane, then either

$$\langle -x_T , (x-x_T) \rangle \geq 0 \text{ for any } x \text{ in } R(x_T;e), \tag{3.13}$$

or

$$\langle -x_T , (x-x_T) \rangle \leq 0 \text{ for any } x \text{ in } R(x_T;e). \tag{3.14}$$

In fact, it will be shown that Equation 3.14 is true:

$$\langle x_T , (x-x_T) \rangle \geq 0 \quad \text{for any } x \text{ in } R(x_T;e). \tag{3.15}$$

Since x_T is the closest point to the origin,

$$||x||^2 - ||x_T||^2 \geq 0. \tag{3.16}$$

Since $R(x_T;e)$ is convex, it is known that for any x_1 and x_2 in $R(x_T;e)$, x_3 defined by:

$$x_3 = \pi x_1 + (1 - \pi)x_2, \quad 0 \leq \pi \leq 1 \tag{3.17}$$

is also in $R(x_T;e)$. In particular, let x_1 be an arbitrary $x \neq x_T$ and let x_2 be x_T. The resulting x_3 is in $R(x_T;e)$, but is not ($\pi \neq 0$) the closest point to the origin. Define, for $0 \leq \pi \leq 1$,

$$f(\pi) = ||\pi x + (1-\pi)x_T||^2 - ||x_T||^2, \tag{3.18}$$

or

$$f(\pi) = \langle (\pi x + (1-\pi x_T)) , (\pi x + (1-\pi x_T)) \rangle - ||x_T||^2. \tag{3.19}$$

Note that

$$f(\pi) \geq 0 \tag{3.20}$$

and that

$$f(\pi) = 0 \text{ if and only if } \pi = 0. \tag{3.21}$$

Therefore $f(\pi)$ has a minimum on the interval $[0,1]$ at the endpoint, $\pi = 0$, consequently:

$$\left.\frac{df(\pi)}{d\pi}\right|_{\pi = 0} \geq 0. \tag{3.22}$$

Differentiating Equation 3.19 yields

$$\frac{df(\pi)}{d\pi} = < (\pi x + (1-\pi)x_T) , (x-x_T) > + < (x-x_T) , (\pi x + (1-\pi)x_T) > , \tag{3.23}$$

or

$$\frac{df(\pi)}{d\pi} = 2< (\pi x + (1-\pi)x_T), (x-x_T) > , \tag{3.24}$$

and

$$\left.\frac{df(\pi)}{d\pi}\right|_{\pi = 0} = 2 < x_T , (x-x_T) >. \tag{3.25}$$

But by Equation 3.22

$$2 < x_T , (x-x_T) > \geq 0, \tag{3.26}$$

or

$$< x_T , (x-x_T) > \geq 0. \tag{3.27}$$

which is precisely Equation 3.15, thus the proof of sufficiency is complete.

Necessity is easily proved. Since $M(x_T,p_T)$ is a support hyperplane and since the origin is a point external to the hyperplane, there is exactly one line through the origin which is normal to the hyperplane [B6]. But this is the shortest path to the hyperplane, hence to the set K, since $M(x_T,p_T)$ is a support hyperplane. Thus the proof of Theorem 3.1 is complete.

Using the results given in Theorem 3.1, several error

functions can be developed. They are all based on the fact that at a local optimum (closest point to the origin in a convex subset of R(T)), x_T and p_T are collinear and oppositely directed. One obvious candidate for an error function is the following:

$$E_2(p_o) = \cos \gamma = \frac{\langle x_T , p_T \rangle}{\| x_T \| \, \| p_T \|} . \qquad (3.28)$$

It should be noted that at a local optimum, cos γ is -1 since x_T and p_T are collinear. Much attention is later given to this error function; in particular, it is shown that cos γ monotonically decreases to -1 along a (yet to be defined) path on the boundary of a locally convex subset of the reachable set.

It is also possible to consider other error functions. They are, however, just modifications and combinations of E_1 and E_2. Since both E_1 and E_2 are a minimum at the optimum, one possible combination would be the product of the two, thus compounding their convergence:

$$E_3(p_o) = \frac{\langle x_T , p_T \rangle}{\| p_T \|} , \qquad (3.29)$$

with the optimum occurring at the minimum <u>negative</u> number. While E_3 should compound the convergence rate, it is less desirable than E_2 from one standpoint--it does not approach a specific value at the minimum. Converting E_2 such that it approaches zero gives:

$$E_4(p_o) = \frac{\langle x_T , p_T \rangle}{\| x_T \| \| p_T \|} + 1, \tag{3.30}$$

and the corresponding compound error function would be:

$$E_5(p_o) = \frac{\langle x_T , p_T \rangle}{\| p_T \|} + \| x_T \|. \tag{3.31}$$

Multiplying by $\| p_T \|$, another version is:

$$E_6(p_o) = \langle x_T , p_T \rangle + \| x_T \| \| p_T \|. \tag{3.32}$$

Of course, many other combinations are possible, but those listed above represent the most convenient forms.

Examination of the error functions indicates that there is a possibility of erroneous minima if either x_T or p_T equals 0. If x_T is zero then E_2 and E_4 have the indeterminant 0/0 form. If p_T is zero then E_2 through E_5 all have the indeterminant 0/0 form and E_6 is zero even though x_T is not necessarily an optimum. Thus E_1 must be used in the event that p_T is 0. Of the compound error functions (E_3, E_5 and E_6), E_5 is preferable to E_3 because it equals zero at the optimum and to E_6 because E_5 is not dependent upon the magnitude of the final adjoint. In most of the computations described in Chapter 5, the error function E_5 is used.

In summary, several error functions have been introduced which can be used in an algorithm for solving LP. The suggested iterative procedure is:

1) Pick an initial p_o.

2) Integrate Equations 2.19 and 2.28 using a maximal control (Equation 2.30) until x_T and p_T are available.

3) Compute the value of the error function, E.

4) Change p_o (by some method yet to be determined) such that E is improved. Continue until the error function indicates (for example, $E_5 = 0$) that an optimum has been determined.

Now that error functions are available to differentiate between local optima and other points, it is important to consider the precise method for changing p_o (item 4 of the suggested method) such that E is improved. Two methods are considered. One is based upon a direct search on the initial adjoints, the other upon geometric considerations of the reachable set. These methods and their convergence are discussed in the following sections of this chapter.

3.3 Determination of a Local Optimum via Direct Search

Any direct (or pattern) search technique presupposes that the change in the controlled parameter (p_o) can be made sufficiently small such that the resulting change in the evaluation parameter (E) is correspondingly small. For linear systems it is evident that a sufficiently small change in E can be obtained. For nonlinear systems, however, it is not so evident.

Given a specified bound on the change in p_T and x_T, it must be shown that the change in p_o can be made sufficiently small to keep the change in p_T and x_T within this

bound. To prove this result for nonlinear systems, an embedding theorem by Hestenes and Guinn is utilized [H3]. The theorem is somewhat more general than the embedding theorem given in Hestenes' book [H4].

THEOREM 3.2 Let $x(\cdot)$, $p(\cdot)$ and $u(\cdot)$ be defined by Equations 2.19, 2.28 and 2.30 with corresponding initial adjoint p_0. Let $E(p_0)$ represent any of the error functions previously defined (Equations 3.4, 3.28-3.32). Then for any $\epsilon > 0$, there exists a $\delta > 0$ such that

$$| E(p_0) - E(p_0') | < \epsilon, \tag{3.33}$$

whenever

$$|| p_0 - p_0' || < \delta. \tag{3.34}$$

Proof: Let ϵ be given. Since each of the error functions being considered is continuous in x_T and p_T, there exists a δ_1 and a δ_2 such that

$$|| p_T - p_T' || < \delta_1 \tag{3.35}$$

$$|| x_T - x_T' || < \delta_2 \tag{3.36}$$

imply Inequality 3.33 is satisfied. Now consider Inequality 3.35. The relationship between p_0 and p_T can be written as

$$p_T = \Psi(T,t_0)\, p_0 \tag{3.37}$$

where $\Psi(T,t_0)$ is the fundamental matrix for the adjoint system (Equation 2.28) which satisfies:

$$\frac{d\Psi(t,t_0)}{dt} = -\frac{\partial f^T}{\partial x}\Psi(t,t_0). \tag{3.38}$$

Since $\Psi(t,t_0)$ is nonsingular, there exists a δ_3 such that

$$|| p_o - p_o' || < \delta_3 \tag{3.39}$$

implies

$$|| p_T - p_T' || < \delta_1. \tag{3.40}$$

Since hypotheses H1) through H3) are sufficient assumptions for the embedding theorem [H3], there exists a δ_4 such that

$$|| p_o - p_o' || < \delta_4 \tag{3.41}$$

implies

$$|| x_T - x_T' || < \delta_2. \tag{3.42}$$

Now let δ be the smaller of δ_3 and δ_4 and the theorem is proved. The fact that small changes in p_o result in small changes in x_T is necessary for the following direct search algorithm:

LOCAL OPTIMUM PROCEDURE--DIRECT SEARCH (LOP-DS)

A. $E(p_o^i)$ Evaluation: Whenever $E(p_o^i)$ is to be evaluated, then these steps are followed: Given p_o^i and x_o, initial conditions, integrate Equations 2.19 and 2.28 utilizing a maximal control. From the values obtained for x_T^i and p_T^i, evaluate $E(p_o^i)$ by means of Equation 3.31 $(E_5(p_o))$ unless otherwise indicated.

B. Initialization: Choose a step size h for the components of p_o, a final stopping tolerance E_t, an improvement factor $N_t > 1$ and a maximum number of allowed iterations, I_t. Set i = 0 and select an arbitrary p_o^0; then evaluate $E(p_o^0)$. If $E(p_o^0) \leq E_t$ then $x_T^0 = x_T^{@}$ is a suitable approximation to an optimum x_T^+.

C. Iterations: Define the vector δ_j whose j^{th} component is h, all other components being zero. Let j = 1.

1. Evaluate $E(p_o^i + \delta_j)$ and $E(p_o^i - \delta_j)$.

2. Let

$$p_o^i(j) = \begin{cases} p_o^i + \delta_j \text{ if } E(p_o^i + \delta_j) < E(p_o^i) \\ p_o^i - \delta_j \text{ if } E(p_o^i+\delta_j) \geq E(p_o^i) \text{ and } E(p_o^i-\delta_j) < E(p_o^i) \\ p_o^i \quad \text{otherwise.} \end{cases} \quad (3.43)$$

3. If $j < n$, let $j = j+1$ and repeat steps 1 and 2 with p_o^i replaced by $p_o^i(j-1)$. If $j = n$, go to step 4.

4. If $p_o^i(n) = p_o^i$, decrease h, set $j = 1$ and repeat steps 1 through 3. Otherwise go to step 5.

5. Test to determine if

$$E_5(p_o^i(n)) \leq E_t. \quad (3.44)$$

If so, $p_o^i(n)$ is an approximation to p_o^+ and the corresponding final state $x_T^i(n) = x_T^@$ is an approximation to a local optimum. If not, go to step 6.

6. If

$$E(p_o^i) - E(p_o^i(n)) \geq E_t/N_t \quad (3.45)$$

let $p_o^{i+1} = p_o^i(n)$, $i = i+1$, $j = 1$ and repeat steps 1 through 6. Otherwise go to step 7.

7. Evaluate $E_1(p_o^i(n))$. Let $p_o^{i+1} = p_o^i(n)$, $i = i+1$ and $j = 1$. Apply steps 1 through 6 using E_1 for E except in Equation 3.44. In the event that in step 6,

$$E(p_o^i) - E(p_o^i(n)) < E_t/N_t, \quad (3.46)$$

go to step 8.

8. If $i \geq I_t$, then terminate the procedure (in this case, a new arbitrary p_o^0 could be selected and the algorithm repeated). Otherwise, let $j = 1$, $p_o^{i+1} = p_o^i(n)$ and $i = i+1$. Increase h and go to step 1 (continue using $E = E_1$).

NOTE: A consistent pattern for increasing h is desirable. For example, at each decrease, h could be halved; at each increase, h could be multiplied by a factor such that a continually increasing h is attained in step 8. A flow chart for this algorithm is given in Figure 3.1.

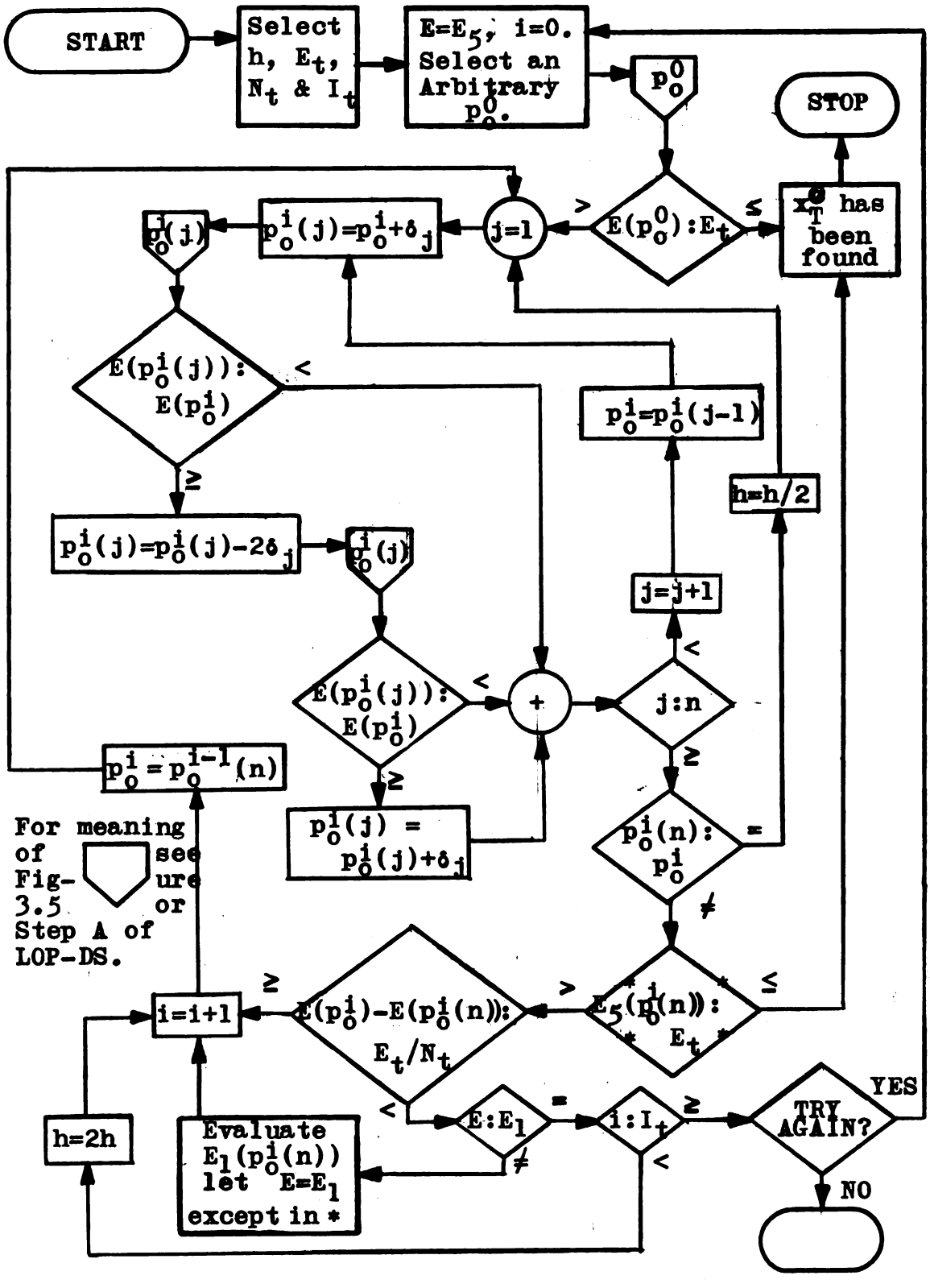


FIGURE 3.1 Flow Chart for LOP-DS

Examination of the preceding algorithm is instructive relative to the possible limit points of $E_5(p_o)$. If Equation 3.44 holds, then an approximation to a local optimum has been achieved. If step 7 is reached and subsequent usage of E_1 does achieve convergence to a local optimum, then a local minimum of E_5 (but not a local minimum of E_1) had previously been attained (see Section 4.3). In the event that step 8 is reached, a limit point which is neither a minimum of E_1 nor of E_5 is a possibility. Clearly the initial choice of h, E_t, N_t and I_t require careful consideration.

Since E_5 and E_1 have a lower bound and since a sequence of controls is possible for which these error functions monotonically decrease, convergence to a limit point is assured even though convergence to a local optimum cannot be guaranteed. In addition, as indicated in LOP-DS, it is possible to evaluate a limit point (using E_5) to determine whether or not it is a local optimum. Similarly, it is possible to determine if the limit point represents a minimum of E_5 but not of E_1. Certainly it should be pointed out that no limit points, other than local minima of E_1 and E_5 were encountered in any experimentation performed by the author. Thus in a practical sense, LOP-DS converges to x_T^{θ}, an approximation to x_T^{+}, while theoretically speaking, only convergence to limit points is guaranteed.

Finally, one other limitation must be noted in the

above algorithm. There exists the possibility, though remote, that x_T^{ℓ} does not belong to $\partial R(T)$, i.e., that a maximal but not an extremal solution has been found. Any effective, global iterative procedure must take into account the above limitations. These are again considered in Chapter 4.

Several limitations are apparent in the above method and convergence proof:

1) There is no guarantee that the initial adjoint thus determined represents an optimum (for instance, E_5 may be a minimum but not zero).

2) There exists the possibility, though remote, that x_T^{ℓ} does not belong to $\partial R(T)$, i.e., that a maximal but not an extremal solution has been found.

Any effective, global iterative procedure must take the above limitations into account. These are considered in Chapter 4.

3.4 Essential Concepts from Differential Geometry

In the previous section it was shown that a sequence of extremal endpoints, $x_T^0,\cdots,x_T^{\ell}$, can be determined starting with an arbitrary final state and terminating at a final state which is an approximation to a local optimum. If the step size is kept sufficiently small, this sequence essentially defines a curve, $L(x_T^0,x_T^{\ell})$, on the boundary of the reachable set. It should be noted, however, that few geometric characteristics of the reachable set were employed in determining this curve.

It is the purpose of this and future sections of this chapter to utilize geometric characteristics of the hypersurface $\partial R(T)$ in defining the curve $L(x_T^0, x_T^{\ell})$. Preparatory to this, the following definitions from differential geometry are given [G2,W1]. While these definitions apply to curves and hypersurfaces in general, specific application is made to the curve $L(x_T^0, x_T^{\ell})$ with position vector x_T on the hypersurface $\partial R(T)$. Incremental arc length is designated ds and the position vector is designated x(s).

DEFINITION 3.1 Let x(s) be a vector function describing a curve L. Let $\frac{dx(s)}{ds}\Big|_{s \neq q} \neq 0$ and let $\frac{dx}{ds}\Big|_q \cdots \frac{d^r x}{ds^r}\Big|_q$ exist and be linearly independent. Then the r-dimensional osculating space of the curve at the point q is the r-dimensional vector space spanned by the derivative vectors $\frac{dx}{ds}\Big|_q \cdots \frac{d^r x}{ds^r}\Big|_q$.

DEFINITION 3.2 The unit tangent vector to x(s) (that is, to the curve L) at a point q is defined as

$$t(s) = \frac{dx(s)}{ds}. \tag{3.47}$$

Note that the one-dimensional osculating space at point q is the tangent to x(s) at q.

DEFINITION 3.3 Let x(q) and x(q') be two neighboring points on L, then the osculating plane of L at x(q) is the limiting position (as x(q') approaches x(q)) of that plane containing t(q) and x(q'). Note that this is the two-dimensional osculating space.

DEFINITION 3.4 Let x(q) be a point on L with tangent t(q), then the normal hyperplane at x(q) is the hyperplane through x(q) which is orthogonal to t(q).

DEFINITION 3.5 The principal normal at x(q) is the line of intersection of the osculating plane and the normal hyperplane. The unit vector along the principal normal is denoted by n(q).

DEFINITION 3.6 The curvature (of the curve L), denoted k, is the arc rate at which the tangent changes direction along L:

$$\frac{dt(s)}{ds} = \frac{d^2x(s)}{ds^2} = kn(s). \tag{3.48}$$

Note that n belongs to the hyperplane normal to t(s), hence they are orthogonal:

$$< t(s) , n(s) > = 0. \tag{3.49}$$

The preceding definitions are given for curves in general; now consider curves on a hypersurface. The hypersurface itself has a tangent hyperplane (assuming smoothness) and an associated unit normal N to the hypersurface. The two normals (n, to the curve and N, to the hypersurface) do not necessarily coincide. In fact, two of the most important curves on a hypersurface are defined by the behavior of the normal to the hypersurface as related to the tangent and to the normal to a curve on the hypersurface.

DEFINITION 3.7 For any curve on a hypersurface the curvature vector is $d^2x(s)/ds^2$, which can further be

represented as a combination of the hypersurface normal and the vector w:

$$\frac{d^2x(s)}{ds^2} = kn(s) = k_n N + w \qquad (3.50)$$

where k_n is the normal curvature.

One important curve on a hypersurface is the geodesic. In differential geometry it is defined in the following manner:

DEFINITION 3.8 A geodesic is a curve on a hypersurface for which w = 0, i.e. for which the principal normal to the curve coincides with the normal to the hypersurface. Thus the curvature k and the normal curvature k_n are equal.

Generally speaking, geodesics (on hypersurfaces) are analogous to straight lines in Euclidean space and are curves of shortest distance. Since geodesics are curves of shortest distance, it was desired to seek a modification of LOP-DS so that the curve $L(x_T^0, x_T^\theta)$ would be a geodesic. Thus $L(x_T^0, x_T^\theta)$ would represent a shortest path from the arbitrary starting point x_T^0 on the boundary of the reachable set to x_T^θ. Attempts of the author to achieve such a modification have thus far been unsuccessful.

A second important class of curves on a hypersurface are the lines of curvature. Such a line is determined by considering the rate of change of the hypersurface normal. In general, one can write

$$\frac{dN}{ds} = -k \frac{dx}{ds} + \gamma_s v \qquad (3.51)$$

where v is a unit vector orthogonal to dx/ds and contained in the tangent hyperspace at the point under consideration. The factor γ_s is called the torsion of the hypersurface in the direction of the tangent (dx/ds) to the curve.

DEFINITION 3.9 Consider curves which have a direction such that $\gamma_s = 0$; such directions are called principal directions (of curvature) and the associated curvatures, k, are called principal curvatures.

DEFINITION 3.10 A curve on a hypersurface whose tangent at every point is along a principal direction is a line of curvature.

Lines of curvature thus have the property that the rate of change of hypersurface normal coincides (in direction) with the tangent to the curve on the hypersurface. This is expressed in Rodriques' formula:

$$dN + k\,dx = 0. \tag{3.52}$$

Further, concerning lines of curvature and principal directions, it has been shown [G2,W1] that:

1) A point where the principal directions are wholly indeterminate (all principal curvatures have the same value) is called an umbilical point. A hyperplane and a hypersphere (or portion thereof) are the only hypersurfaces whose points are all umbilical points.

2) Except for the two instances mentioned in item 1), the principal directions always exist and define an orthogonal system of directions.

3) The principal directions represent directions of extreme values of curvature.

3.5 Convergence to a Local Optimum via Lines of Curvature

In Section 3.3 a direct search technique on the initial adjoints was shown to result in a sequence of extremal endpoints (defining a curve, L, on the boundary of R(T)) which converges to an approximation to a minimum of $E(p_o)$. In this section, the definitions of the previous section are combined with the concepts of extremal endpoints and orthogonal final adjoints to formulate a different converging sequence. It is shown that for a convex set, there exists a well-defined curve, $L(x_T^0, x_T^+)$, consisting of lines of curvature, along which the error function monotonically decreases to the optimum. This is shown using $E_2(p_o)$ or $\cos\gamma$.

THEOREM 3.3 Let R in E^n be a strictly convex compact set with $0 \notin R$. Let x_T^0 be an arbitrary boundary point of R and let x_T^+ be such that

$$||x_T|| > ||x_T^+|| \text{ for any } x_T \neq x_T^+ \text{ in R.} \tag{3.53}$$

Then there exists a path $L(x_T^0, x_T^+)$ on the boundary of R, consisting entirely of lines of curvature, such that the error function

$$E_2(p_o) = \cos\gamma = \frac{\langle p_T , x_T \rangle}{||p_T||\,||x_T||} \tag{3.54}$$

monotonically decreases to -1 and such that $||x_T||$ monotonically decreases to its minimum value x_T^+.

Proof: Note first that R may represent the entire reachable set R(T) if the system being considered is linear, or it may represent a convex subset of R(T). The boundary

points in these cases would represent extremal endpoints and the corresponding outward normals, p_T, would represent final adjoints. As previously introduced, ds represents the incremental arc length along $L(x_T^0, x_T^+)$. To show the monotonicity of cos γ, it is sufficient to show that

$$\frac{d \cos \gamma}{ds} < 0 \tag{3.55}$$

along the specified path.

Since p_T is an outward normal to the surface at x_T, utilizing the terminology of the previous section yields

$$N = \frac{p_T}{\|p_T\|} . \tag{3.56}$$

As a notational simplification, the subscript T will be deleted. Thus Equation 3.54 becomes

$$E_2(p_o) = \cos \gamma = \frac{\langle x , N \rangle}{\|x\|} . \tag{3.57}$$

Taking the derivative of Equation 3.57 one obtains:

$$\frac{d \cos \gamma}{ds} = \left\langle \frac{dN}{ds} , \frac{x}{\|x\|} \right\rangle + \left\langle N , \frac{dx/ds}{\|x\|} \right\rangle - \left\langle N , \frac{x \, d\|x\|/ds}{\|x\|^2} \right\rangle . \tag{3.58}$$

Since dx/ds = t is the tangent

$$\left\langle N , \frac{dx/ds}{\|x\|} \right\rangle = 0 \tag{3.59}$$

and Equation 3.58 becomes

$$\frac{d \cos \gamma}{ds} = \left\langle \frac{dN}{ds} , \frac{x}{\|x\|} \right\rangle - \left\langle N , \frac{x \, d\|x\|/ds}{\|x\|^2} \right\rangle . \tag{3.60}$$

To this point, the path has not been identified more precisely than belonging to the boundary of R; thus it is now

assumed that L is composed entirely of segments of lines of curvature. Hence Equation 3.52 applies and Equation 3.60 becomes

$$\frac{d \cos \gamma}{ds} = \left\langle -k \frac{dx}{ds}, \frac{x}{\|x\|} \right\rangle - \left\langle N, \frac{x \, d\|x\|/ds}{\|x\|^2} \right\rangle. \tag{3.61}$$

Now consider

$$\langle x, x \rangle = \|x\|^2. \tag{3.62}$$

Thus

$$\frac{d}{ds}\langle x, x \rangle = 2\left\langle x, \frac{dx}{ds} \right\rangle = \frac{d}{ds}\|x\|^2 = 2\|x\| \frac{d\|x\|}{ds}. \tag{3.63}$$

Or

$$\frac{d\|x\|}{ds} = \frac{1}{\|x\|}\left\langle x, \frac{dx}{ds} \right\rangle. \tag{3.64}$$

Substituting Equation 3.64 into Equation 3.61 gives:

$$\frac{d \cos \gamma}{ds} = -k \frac{d\|x\|}{ds} - \frac{d\|x\|/ds}{\|x\|}\left\langle N, \frac{x}{\|x\|} \right\rangle. \tag{3.65}$$

Hence,

$$\frac{d \cos \gamma}{ds} = -\left(k + \frac{\cos \gamma}{\|x\|}\right) \frac{d\|x\|}{ds}. \tag{3.66}$$

Since the curve, L, is on the surface of a convex set, the curvature k is always negative. At or near the optimum on any reasonably behaved surface cos γ is negative; thus for the derivative of cos γ to always satisfy Equation 3.55, it is necessary that

$$\frac{d\|x\|}{ds} < 0. \tag{3.67}$$

Not only is this requirement necessary for the proof of the theorem, it is also desirable from an understanding of the problem since the optimum point is the point of minimum

$||x||$. It is now shown that Equation 3.67 can be satisfied while simultaneously remaining on lines of curvature.

The hypersurface ∂R is m-dimensional where $m < n-1$. Through each point x on ∂R there are m orthogonal directions defined by the tangents to the lines of curvature through x. Although these vectors are orthogonal, in the proof to follow, it is only necessary to require that they are linearly independent.

Consider the arbitrary starting point $x^0 \neq x^+$ and let $L(x^0,x^i)$ denote a path on ∂R composed of lines of curvature, along which $||x||$ decreases monotonically from $||x^0||$ to $||x^i||$. It is first shown that there must exist one such path, i.e. that it is possible to move from x^0 along a line of curvature such that $||x||$ is decreased.

Define level hypersurfaces, P(r), on ∂R as the intersection of ∂R with m-dimensional hyperspheres S(r) of radius r and centered at the origin. The resulting level hypersurfaces are of dimension m-1, thus at least one of the tangent vectors for any $x \neq x^+$ belonging to one of these level hypersurfaces must intersect that level hypersurface. Thus it is possible to move along the lines of curvature corresponding to this tangent such that $||x||$ is decreased. In the event that $r = ||x^+||$, the intersection of S(r) with ∂R yields a point, x^+.

Consider the collection of all paths, $L(x^0,x)$, formed of lines of curvature for which $||x||$ is decreased. If

$L(x^0,x^+)$ is in this collection, then the theorem is proved. If not, then there must be a lower bound greater than $||x^+||$ for the $||x||$ obtainable, i.e., a level hypersurface P(b) must bound all $L(x^0,x)$ from below. There may be many such paths which approach P(b).

For each x^i there is a corresponding orthogonal system through x^i consisting of lines of curvature. Along some of these lines, $||x||$ is decreased. Let $D(x^i)$ represent the point x^i together with those arcs of its lines of curvature, containing x^i, along which $||x||$ decreases and for which $||x|| < ||x^+||$.

Since b is the lower bound of the norms of points which are joinable to x^0 by admissible paths, there exists a sequence of points $\{x^i\}$ such that $\{||x^i||\}$ converges to b. Since ∂R is compact, $\{x^i\}$ contains a subsequence converging to a point $\bar{x}$, whose norm is b. For notational simplicity, $\{x^i\}$ is also used to denote the convergent subsequence. As previously mentioned for an arbitrary $x \neq x^+$, at least one member of the orthogonal system must penetrate the level hypersurface to which the point x belongs. Thus there exists an $\bar{x}' \in D(\bar{x})$ with $||\bar{x}'|| < b$.

From considerations of continuity, it follows that $\{D(x^i)\}$ approaches $D(\bar{x})$. Thus there exists a sequence $\{x'^i\}$, where $x'^i \in D(x^i)$, which converges to $\bar{x}'$. Also $\{||x'^i||\}$ approaches $||\bar{x}'|| < b$. Hence for some k, $||x'^k|| < b$. Consider the union of $L(x^0,x^k)$ with the arc $x^k \frown x'^k$ of

$D(x^k)$. This union joins x^0 to x'^k. But x'^k has a norm less than b, thus contradicting the definition of b as the lower bound. Hence the only lower bound is $||x^+||$ and it has been shown that x^0 is joinable to x^+ by a path, $L(x^0,x^+)$. Hence the proof of the theorem.

Although this theorem is proven for the error function $E_2 = \cos\gamma$, it is also possible to prove the monotonicity of the other error functions along the path $L(x_T^0,x_T^+)$. Of all error functions given in Section 3.2, E_1 and E_2 are most basic and are fundamental to all others. The final statement of Theorem 3.3 also demonstrates the monotonicity of E_1.

Of the compound error functions, E_5 is most direct and effective. Thus, consider Theorem 3.3 as it relates to E_5.

COROLLARY 3.3.1 Let the hypotheses of Theorem 3.3 apply. Then there exists a path $L(x_T^0,x_T^+)$ on the boundary of R, consisting entirely of lines of curvature, such that the error function

$$E_5(p_o) = ||x_T|| \ (1 + \cos\gamma) \qquad (3.68)$$

monotonically decreases to 0.

Proof: Since E_2 decreases monotonically to -1, $1 + \cos\gamma$ decreases monotonically to 0. But $||x_T||$ also decreases monotonically, hence the product decreases monotonically to 0. Hence the proof.

It is instructive to consider E_5 so as to develop an equation analogous to Equation 3.66. Consider

$$\frac{dE_5}{ds} = \frac{d}{ds} \left[\| x_T \| \ (1 + \cos \gamma) \right], \tag{3.69}$$

or

$$\frac{dE_5}{ds} = \frac{d \| x_T \|}{ds} (1 + \cos \gamma) + \| x_T \| \frac{d \cos \gamma}{ds} . \tag{3.70}$$

Substituting Equation 3.66, one obtains

$$\frac{dE_5}{ds} = \frac{d \| x_T \|}{ds} (1 + \cos \gamma) - (k \| x_T \| + \cos \gamma) \frac{d \| x_T \|}{ds} . \tag{3.71}$$

Thus

$$\frac{dE_5}{ds} = (1 - k \| x_T \|) \frac{d \| x_T \|}{ds} . \tag{3.72}$$

For E_5 to decrease monotonically along $L(x_T^0, x_T^+)$, the following inequality must hold:

$$1 - k \| x_T \| > 0, \tag{3.73}$$

or

$$k < \frac{1}{\| x_T \|} . \tag{3.74}$$

For the convex set of Theorem 3.3 this inequality certainly holds since k is negative. Questions relating to non-convex sets are considered in Chapter 4. Note that Equation 3.74 allows this possibility since dE_5/ds may be negative even though k is positive.

3.6 Effective Boundary Path Selection

For a convex region of R(T) which includes a local optimum, the results of the previous section show that at least one and probably many acceptable paths exist on the boundary of the reachable set. Each of these paths run

from the arbitrary starting point to the local optimum. These paths represent collections of extremal trajectory endpoints. Any method based upon these results would have as its goal the successive determination of these endpoints, hence of these trajectories. Certainly it is not desirable to identify all of the endpoints since the path consists of an infinite number, but it is desirable to identify a sufficient number to define the path and hence locate the optimum endpoint.

Were the boundary of the reachable set explicitly defined by a vector function, the path would likewise be precisely defined. If this were the case, however, determination of the path would be unnecessary since such an explicit definition of $\partial R(T)$ would easily lead to location of the local optimum either by direct calculation or through functional minimization. In any practical nonlinear problem, however, it is not possible to explicitly define the reachable set. Based on hypotheses and theorems given earlier in this thesis, only the following general observations are available:

1) R(t) is of dimension $r \leq n$, where n is the dimension of the state equations.

2) The boundary of R(t) consists of extremal trajectory endpoints resulting from extremal controls.

3) R(t) is compact and varies continuously with time.

4) R(t) is, in general, not globally convex, but will have regions of local convexity.

How then, does one, experimentally, determine the path from the arbitrary starting point to the local optimum? Because of the fact that so little is known of the shape of the reachable set, it was the author's original decision to implement some type of search method on $\partial R(T)$ (i.e. among the initial adjoints corresponding to these extremal end-points). One such method is LOP-DS given in Section 3.3. Theorem 3.3, however, yields some additional insight into the nature of an effective path to the optimum. Some of these results were previously apparent from the nature of the problem:

1) $d\|x_T\|/ds < 0$, i.e. $\|x_T\|$ must successively decrease.

2) The decrease in $\|x_T\|$ should be made as large as possible (i.e., $d\|x_T\|/ds$ should be minimized).

Others are provided as a result of Theorem 3.3:

3) Since the path consists of lines of curvature,

$$\frac{dN}{ds} + k\frac{dx_T}{ds} = 0. \tag{3.75}$$

4) If there is a choice of lines of curvature, as exists in most cases, the one for which k is minimum should be selected at each decision point along the path.

While arbitrary changes in the p_o's might be acceptable in lower order systems, higher order systems require more sophisticated methods; hence the insight provided through Theorem 3.3 is important and should be utilized. The basic structure of such a method would be as follows:

1) Try an arbitrary p_o^0, note $E(p_o^0)$, p_T^0 and x_T^0.

2) Employing insight into the relationship of x_T and/or p_T to x_T^+ and/or p_T^+ and the interrelationship between δx_T and δp_T (perturbations), change x_T^0 and p_T^0 yielding $\bar{x}_T^0$ and $\bar{p}_T^0$.

3) Run the system in reverse time from $\bar{x}_T^0$ and $\bar{p}_T^0$ using extremal switching. This yields an $\bar{x}_o^0$ and a $\bar{p}_o^0$. Note that $\bar{x}_o^0$ probably differs from x_o (the specified initial state).

4) Consider a new initial adjoint p_o^1 which is related to $\bar{p}_o^0$ and perhaps to $p_o^{0_o}$ and to the change in x_o $(\bar{x}_o^0 - x_o)$.

5) Assuming that p_o^1 results in a better value of the error function, $E(p_o^1)$, then the method is repeated. If the result is not better, some alternative approach must be taken.

Within these five steps, there are strategies which must be chosen on the basis of good judgement as well as mathematical development. Basically these alternatives can be classed into two subdivisions, according to the time at which they occur: in step 2), after forward integration of the system and in step 4), after reverse time integration.

3.6.1 Curvature Algorithm Alternatives--Final Time

Consider first the alternatives available at the final time, i.e. on the boundary of the reachable set. The following choices must be made:

1) Should just x_T or p_T be perturbed or both x_T and p_T?

2) What should be the size of the perturbation?

3) In what direction should the perturbation be made?

4) If both x_T and p_T are perturbed, are the perturbations performed dependently or independently?

5) If x_T and p_T are perturbed dependently, which one is perturbed independently and what is the relationship between the perturbations?

These decisions can be represented as shown in the decision flow chart given in Figure 3.2. It should be emphasized that these decisions must be made on the basis of little knowledge of the nature of the reachable set in the neighborhood of x_T--any additional information must be experimentally determined. The goal, of course, is a perturbation of x_T and p_T such that the error function is decreased and such that the perturbed final state $\bar{x}_T \in \partial R(T)$ and the perturbed final adjoint is normal to $\partial R(T)$ at $\bar{x}_T$.

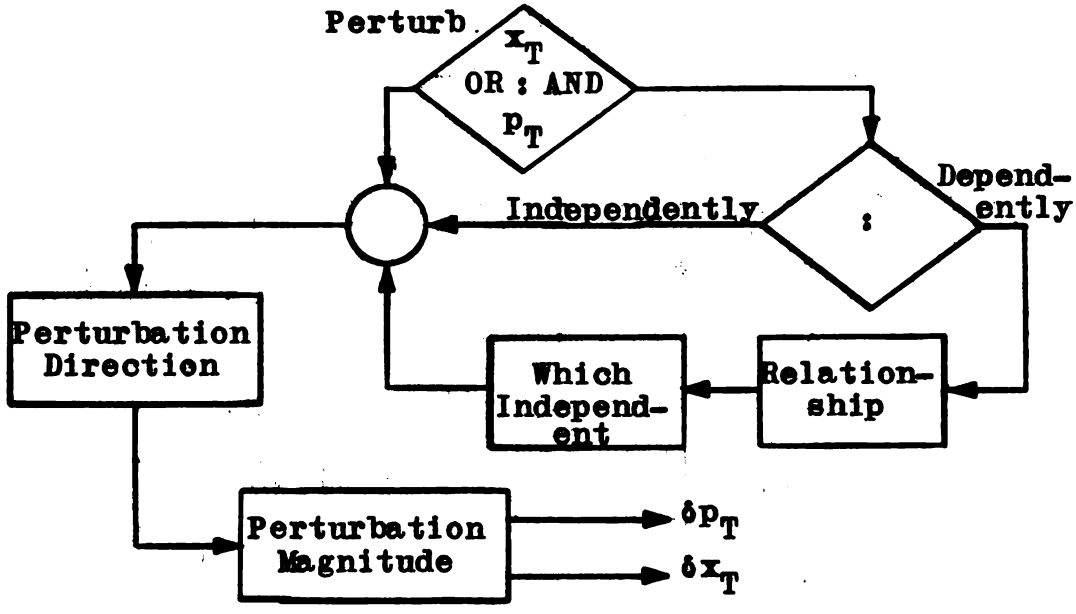


FIGURE 3.2 Final Time Perturbation Decisions

Consider some aspects relative to these perturbations and the above listed alternatives. In the discussion to follow, the perturbed final state and costate are designated $\bar{x}_T$ and $\bar{p}_T$:

$$\bar{x}_T = x_T + \delta x_T \tag{3.76}$$

$$\bar{p}_T = p_T + \delta p_T. \tag{3.77}$$

For any perturbation in x_T, there is a good possibility that $\bar{x}_T$ does not lie on the boundary of the reachable set. Of course, the goal of any perturbation is to minimize this possibility. In addition to this fact, however, other aspects must be considered.

If only x_T is perturbed, $\bar{p}_T = p_T$ probably does not represent a correct outward normal for the new boundary point. Similarly if only p_T is perturbed, $\bar{p}_T$ does not represent a correct outward normal for the unperturbed final state. If both x_T and p_T are perturbed independently or if an incorrect perturbation interrelationship is utilized, again the resulting perturbed normal may not be accurate for the resulting boundary point. Thus a judiciously chosen, related perturbation is most desirable. Even in this case, some departure from the reachable set and the proper outward normal can be expected, particularly if the step size is too large. With careful control of the step size, however, a joint, related perturbation may result in a new $\bar{x}_T$ close to $\partial R(T)$ yet allow δx_T to be large enough to represent a considerable improvement in the final

state.

The interrelationship between x_T and p_T is chosen to coincide with the theoretical developments for lines of curvature, namely, Equation 3.52 applies, or, in incremental steps:

$$\delta N + k \delta x_T = 0. \tag{3.78}$$

Obviously, k must experimentally be estimated and updated as the path moves towards the optimum. The final state, x_T, is chosen for the independent perturbation since the proof of Theorem 3.3 requires (and common sense dictates) that $\| x_T \|$ continuously decreases. Thus, choosing δx_T to be independent allows easy verification of this requirement.

The crucial question remaining unanswered is number 3)--in what direction should the perturbation be made? While this decision must be made without an explicit expression for the reachable set, some facts are available:

1) $\| x_T \|$ must decrease.
2) $E(p_o)$ must decrease.
3) The final perturbed state, $\bar{x}_T$, should remain on the boundary of the reachable set.

To give justification for the direction of perturbation to be selected, consider the following perturbation analyses of $E(p_o)$ which are presented as theorems. The first perturbation direction is based mostly on fact number 1)--decreasing $\| x_T \|$.

THEOREM 3.4 Consider the error function $E_2(p_o)$ (the final time subscript, T, has been deleted for notational simplicity):

$$E_2(p_o) = \cos\gamma = \frac{\langle x , p \rangle}{\|x\|\|p\|} = \frac{\langle x , N \rangle}{\|x\|} \tag{3.79}$$

where x belongs to a convex region of $\partial R(T)$ (thus the curvature, k, is negative) and N is normal to R(T) at x. Let δx be a perturbation in x with corresponding δN given by Equation 3.78. If δx is defined by

$$\delta x = -cx \tag{3.80}$$

where c is a positive constant, $0 < c < 1$, then

$$E(x+\delta x, p+\delta p) \leq E(x,p) = E_2(p_o) \tag{3.81}$$

with equality only at the optimum.

Proof: Let

$$\delta E = E(x,p) - E(x+\delta x, p+\delta p). \tag{3.82}$$

It must be shown that

$$\delta E > 0. \tag{3.83}$$

Consider

$$\delta E = \frac{\langle x , N \rangle}{\|x\|} - \frac{\langle x+\delta x , N+\delta N \rangle}{\|x+\delta x\|}, \tag{3.84}$$

or,

$$\delta E = \frac{\langle x , N \rangle}{\|x\|} - \frac{\langle x , N \rangle}{\|x+\delta x\|} - \frac{\langle \delta x , N \rangle}{\|x+\delta x\|} - \frac{\langle x , \delta N \rangle}{\|x+\delta x\|} - \frac{\langle \delta x , \delta N \rangle}{\|x+\delta x\|}. \tag{3.85}$$

Substituting Equation 3.78 into 3.85 gives:

$$\delta E = \langle x , N \rangle \left(\frac{1}{\|x\|} - \frac{1}{\|x+\delta x\|} \right) - \frac{\langle \delta x , N \rangle}{\|x+\delta x\|} + \frac{\langle x , k\delta x \rangle}{\|x+\delta x\|} + \frac{\langle \delta x , k\delta x \rangle}{\|x+\delta x\|}. \tag{3.86}$$

Note that

$$\|x+\delta x\| = \|x-cx\| = |1-c|\,\|x\| = (1-c)\,\|x\| \qquad (3.87)$$

because of Equation 3.80 and since $0<c<1$. Substituting Equations 3.87 and 3.80 into 3.86 results in:

$$\delta E = \cos\gamma\left(\frac{-c}{1-c}\right) + \frac{c}{1-c}\cos\gamma - \frac{ck}{1-c}\|x\| + \frac{c^2k}{1-c}\|x\|; \qquad (3.88)$$

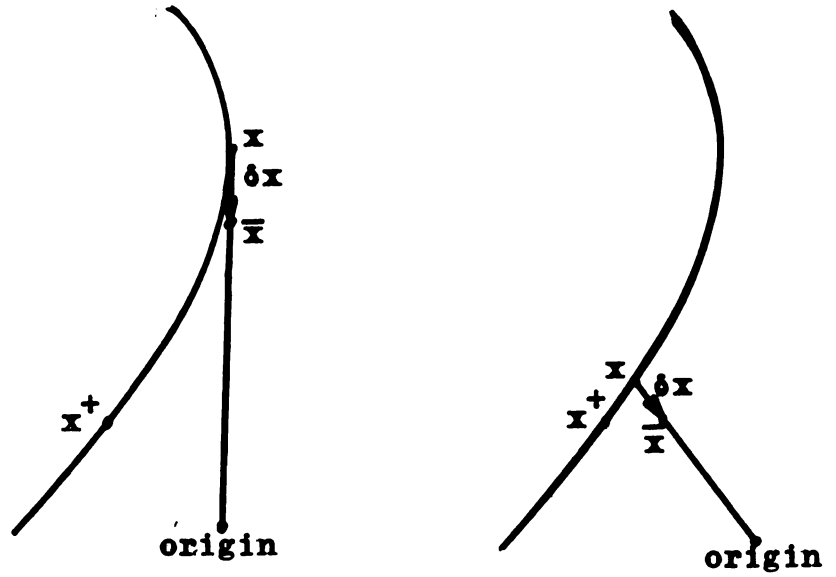
or,

$$\delta E = \frac{ck\|x\|}{1-c}(c-1) = -ck\|x\|. \qquad (3.89)$$

Since c is positive and k is negative, Inequality 3.83 is satisfied and the proof is complete. Note that maximizing $|k|$ will maximize δE.

It should be noted that Equation 3.80 disregards the fact that $x+\delta x$ should also lie on the boundary of the reachable set. If cos γ is near zero (i.e., far from the optimum), δx is approximately tangential and Equation 3.80 is a good approximation (See Figure 3.3a). On the other hand, consider an x near the optimum (Figure 3.3b). In this event, any δx defined by Equation 3.80 is nearly normal to $\partial R(T)$ and thus is a poor choice. Hence, some other choice must be made for δx based on the available information which includes past knowledge of $\partial R(T)$, $x(T)$ and $p(T)$ and current knowledge of $p(T)$ and $x(T)$.

When x_T is near x_T^+, δx is nearly orthogonal to x_T. For second order systems, this would be sufficient information for calculating δx, but for higher dimensional systems, this still does not adequately define δx. Consideration of p_T and x_T near an optimum shows that they are



a. Far from Optimum b. Near the Optimum

FIGURE 3.3 Final State Perturbations: $\delta x = -cx$.

nearly collinear, thus the vector

$$\delta x = -c' \left(\frac{x}{\|x\|} + \frac{p}{\|p\|} \right), \tag{3.90}$$

is nearly tangential near the optimum. This fact is illustrated in Figure 3.4, and leads to the next perturbation analysis as given in Theorem 3.5.

THEOREM 3.5 Let the error function be defined by Equation 3.79 and let Equation 3.78 apply. Let x and N be as previously defined. If δx is defined by Equation 3.90 with $0 < c' < \|x\|/2$, then Equation 3.81 is satisfied.

Proof: Consider δE as defined in Equation 3.82, and as expressed in Equation 3.86. Substituting Equation 3.90 into parts of Equation 3.86, one obtains:

$$\delta E = \langle x , N \rangle \left(\frac{1}{\|x\|} - \frac{1}{\|x+\delta x\|}\right) + \frac{\langle c'x , N \rangle}{\|x\|\|x+\delta x\|} + \frac{\langle c'N , N \rangle}{\|x+\delta x\|}$$

$$- \frac{\langle x , kc'x \rangle}{\|x+\delta x\|\|x\|} - \frac{\langle x , kc'N \rangle}{\|x+\delta x\|}$$

$$+ \frac{k(c')^2}{\|x+\delta x\|} \left(\left\| \frac{x}{\|x\|} \right\|^2 + 2 \left\langle \frac{x}{\|x\|} , N \right\rangle + \langle N , N \rangle \right). \quad (3.91)$$

Since N and $x/\|x\|$ are both unit vectors, Equation 3.91 becomes:

$$\delta E = \langle x , N \rangle \left(\frac{1}{\|x\|} - \frac{1}{\|x+\delta x\|}\right) + \frac{c' \cos \gamma}{\|x+\delta x\|} + \frac{c'}{\|x+\delta x\|} - \frac{kc'\|x\|}{\|x+\delta x\|}$$

$$- \frac{kc'\|x\|}{\|x+\delta x\|} \cos \gamma + \frac{2k(c')^2}{\|x+\delta x\|}(1 + \cos \gamma). \quad (3.92)$$

Rearranging Equation 3.92 and collecting terms yields:

$$\delta E = \cos \gamma \left(1 - \frac{\|x\|}{\|x+\delta x\|}\right) + \frac{c'(1 + \cos \gamma)}{\|x+\delta x\|}(1 - k\|x\| + 2kc'). \quad (3.93)$$

To relate the magnitudes of the δx's in this theorem and the previous theorem, define a new constant

$$c'' = \frac{c'}{\|x\|} . \quad (3.94)$$

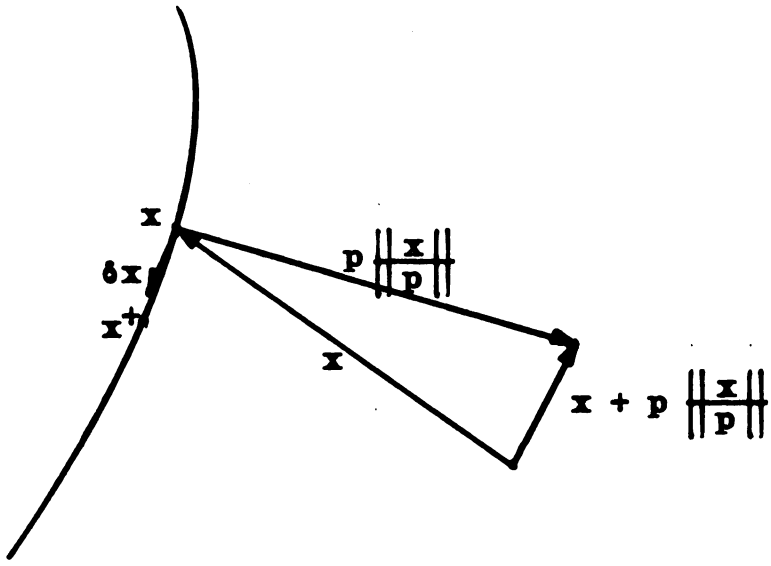


FIGURE 3.4 Final State Perturbations near $x^+(T)$

Note that the hypothesis for the theorem requires that $0 < c'' < \frac{1}{2}$. With substitution of c", Equation 3.93 becomes:

$$\delta E = \cos\gamma \left(1 - \frac{||x||}{||x+\delta x||}\right) + \frac{c''||x||(1+\cos\gamma)}{||x+\delta x||}(1 - k||x|| + 2kc''||x||). \quad (3.95)$$

Now it must be proven that δE is positive. Note that the second term is positive if

$$1 - k||x|| + 2kc''||x|| > 0, \quad (3.96)$$

since $\cos\gamma \geq -1$ with equality only at the optimum (hence the second term is zero at the optimum). Inequality 3.96 can be rearranged as follows:

$$2kc''||x|| > k||x|| - 1, \quad (3.97)$$

or since k is negative (convex surface),

$$c'' < \frac{k||x|| - 1}{2k||x||} = \frac{1}{2} - \frac{1}{2k||x||}. \quad (3.98)$$

Certainly if c" is less than $\frac{1}{2}$, Inequality 3.98 is satisfied since $-1/2k||x||$ is positive; hence the second term of Equation 3.95 is positive except at the optimum where it is zero. Now consider the first term of Equation 3.95, for which it must be shown that

$$\cos\gamma \left(1 - \frac{||x||}{||x+\delta x||}\right) \geq 0. \quad (3.99)$$

It has previously been stated that $\cos\gamma$ is negative in any well-behaved region near a local optimum; thus one must show that

$$\left(1 - \frac{||x||}{||x+\delta x||}\right) \leq 0; \quad (3.100)$$

or

$$||x+\delta x|| - ||x|| \leq 0, \tag{3.101}$$

which was already the objective of this method. With the introduction of c", Equation 3.90 has become

$$\delta x = -c''x - c''||x|| N. \tag{3.102}$$

Consider

$$||x+\delta x|| = ||(x - c''x - c''||x|| N)|| \tag{3.103}$$

or

$$||x+\delta x|| = ||(x(1-c'') - c''||x|| N)||. \tag{3.104}$$

Using the triangular inequality one obtains:

$$||x+\delta x|| \leq ||x(1-c'')|| + ||(c''||x||N)||; \tag{3.105}$$

or

$$||x+\delta x|| \leq (1-c'')||x|| + c''||x|| \tag{3.106}$$

since $c'' < \frac{1}{2}$ (thus 1-c" is positive) and since $||N||$ is 1. Thus Inequality 3.101 is satisfied and the proof of the theorem is complete.

Summarizing, two candidates have been selected for perturbing the final state, as given in Equations 3.80 and 3.90. Equation 3.80 is most effective at points where cos γ is near 0 (far from optimum) and Equation 3.90 is most effective for points on $\partial R(T)$ which are near the optimum (cos γ near -1). Letting

$$c = c'', \tag{3.107}$$

an obvious candidate for a <u>composite</u> choice for δx is

$$\delta x = -c(x - ||x|| N \cos \gamma), \tag{3.108}$$

because it reduces to Equation 3.80 when cos γ is 0 and to

Equation 3.90 when cos γ is -1. This perturbation choice is considered in the next theorem.

THEOREM 3.6 Let the error function be defined by Equation 3.79 and let Equation 3.78 apply. Let x and N be as previously defined. If δx is defined by Equation 3.108, $0 < c < 1$, then Equation 3.81 is satisfied.

Proof: Again consider δE as defined in Equation 3.82. By partial substitution of Equation 3.108 into Equation 3.86, one obtains:

$$\delta E = \langle x , N \rangle \left(\frac{1}{\|x\|} - \frac{1}{\|x+\delta x\|} \right) + \frac{c}{\|x+\delta x\|} \langle x , N \rangle - \frac{c}{\|x+\delta x\|} \langle \|x\| N \cos\gamma , N \rangle - \frac{ck}{\|x+\delta x\|} \langle x , x \rangle + \frac{ck}{\|x+\delta x\|} \langle x , \|x\| N \cos\gamma \rangle + \frac{k}{\|x+\delta x\|} \|\delta x\|^2 . \quad (3.109)$$

Since

$$\|x\| \cos\gamma = \langle x , N \rangle , \quad (3.110)$$

Equation 3.109 becomes

$$\delta E = \cos\gamma \left(1 - \frac{\|x\|}{\|x+\delta x\|} \right) - \frac{ck\|x\|^2}{\|x+\delta x\|} + \frac{ck\|x\|^2 \cos^2\gamma}{\|x+\delta x\|} + \frac{k}{\|x+\delta x\|} \|\delta x\|^2 . \quad (3.111)$$

Now consider

$$\|\delta x\|^2 = c^2 (\|x\|^2 - 2\|x\| \cos\gamma \langle x , N \rangle + \|x\|^2 \cos^2\gamma) \quad (3.112)$$

or

$$\|\delta x\|^2 = c^2 (\|x\|^2 - \|x\|^2 \cos^2\gamma) = c^2 \|x\|^2 (1 - \cos^2\gamma) . \quad (3.113)$$

Substituting this result into Equation 3.111 gives:

$$\delta E = \cos\gamma \left(1 - \frac{\|x\|}{\|x+\delta x\|} \right) - \frac{ck\|x\|^2}{\|x+\delta x\|} (1 - \cos^2\gamma) + \frac{kc^2\|x\|^2}{\|x+\delta x\|} (1 - \cos^2\gamma) ; \quad (3.114)$$

or

$$\delta E = \cos\gamma \left(1 - \frac{||x||}{||x+\delta x||}\right) + \frac{ck||x||^2}{||x+\delta x||}(1-\cos^2\gamma)(c-1). \quad (3.115)$$

While the first term of this equation is the same as derived in Equation 3.95, it must again be considered since δx differs. For that term to be positive (or zero) it must be shown that

$$||x|| \geq ||x+\delta x||. \quad (3.116)$$

From Equation 3.108 we can write,

$$||x+\delta x|| = ||(x-cx+c||x||N\cos\gamma)|| = ||(x(1-c)+c||x||N\cos\gamma)||. \quad (3.117)$$

Using the triangular inequality one obtains:

$$||x+\delta x|| \leq ||x(1-c)|| + ||(c||x||N\cos\gamma)||. \quad (3.118)$$

Since $|\cos\gamma| \leq 1$, c and 1-c are positive; and $||N|| = 1$; thus

$$||x+\delta x|| \leq (1-c)||x|| + c||x|| = ||x||. \quad (3.119)$$

Thus the first term in Equation 3.115 is positive or zero (at the optimum). Now consider the second term. The factor (c - 1) is negative as is k; all other factors are positive hence the term is positive except at the optimum where it is zero; hence the proof of the theorem.

Experimental results are given in Chapter 5 to compare the three possible methods of perturbing δx as given by Equations 3.80, 3.90 and 3.108. Once δx has been defined, then δN is determined through Equation 3.78. Once k is determined then δN is specifically defined. The determination of curvature, k, however, is not an easy task for higher order systems. Since R(T) is not explicitly

defined, k must be experimentally approximated.

Given two neighboring points on a line of curvature, (x,p) and (x',p'), several means are available for calculating an estimate of k. The first estimates are apparent from Equation 3.78 where n (dimension of the state) approximations are possible

$$\bar{k}_i = - \frac{N_i - N_i'}{x_i - x_i'} \quad , \; i = 1, \cdots, n. \qquad (3.120)$$

any of these estimates could be used or the average:

$$\bar{k}_{av} = \left(\sum_{i=1}^{n} k_i \right) / \, n. \qquad (3.121)$$

A second estimate of k is available from Equation 3.66 which, solving for k gives

$$k = - \frac{d \cos \gamma}{d \| x \|} - \frac{\cos \gamma}{\| x \|}. \qquad (3.122)$$

Or, in terms of perturbations

$$k \approx \bar{k} = - \frac{\cos \gamma - \cos \gamma'}{\| x \| - \| x' \|} - \frac{\cos \gamma'}{\| x \|}. \qquad (3.123)$$

Finally, a combination of $\bar{k}$ and $\bar{k}_{av}$ may be used:

$$\bar{k}_{av2} = \tfrac{1}{2} \left(\bar{k}_{av} + \bar{k} \right). \qquad (3.124)$$

For two final states x and x' close to each other (and on a line of curvature), all of the estimates of k (Equations 3.121, 3.123 and 3.124) are near the actual value.

For a second order system, with δx sufficiently small, k can easily be estimated since the boundary of the reachable set is <u>the</u> line of curvature. In higher order systems, however, a perturbation of (x,p) yielding

$$\bar{k}_i \approx \bar{k}_{i+1} \approx \bar{k}, \quad i = 1, \cdots, n-1 \qquad (3.125)$$

would indeed be fortuitous. In general one can expect that the $\bar{k}_i$'s and $\bar{k}$ have differing values and differing signs since the perturbation of (x,p) may not be along a line of curvature through x. For small perturbations in x, the variance of the values for $\bar{k}$ and the $\bar{k}_i$'s is certainly an indication of whether or not the perturbation is along a line of curvature. If the variance is large, several options are available in selecting k: an average may be chosen, one particular estimate formula may be relied upon or more perturbations may be taken until a better estimate of k results. The methods for estimating k are experimentally compared in the next chapter.

3.6.2 Curvature Algorithm Alternatives--Initial Time

As indicated in Section 3.6, after reverse time integration from $(\bar{x}_T^i, \bar{p}_T^i)$, there are several alternative means of choosing p_o^{i+1}, the initial adjoint for the next iteration. Since δx_T^i and δp_T^i are only estimates for an accurate perturbation on $\partial R(T)$, the resulting x_o^i found by reverse time integration of the state and costate equations using maximal switching often differs from the specified initial state, x_o. In addition, computational errors may develop which also contribute to the difference between x_o^i and x_o. Indeed, experimentation indicates that significant errors do develop and that a computated $(\hat{x}_T, \hat{p}_T)$ pair when used, <u>without</u> perturbations, as initial points in a reverse time

integration, may not produce x_o and $\hat{p}_o$. The origin of this computational error is discussed in Chapter 5. For the developments in this section, it is sufficient to observe that a computed x_o^i may not be the initial state because of:

1) Computational errors.

2) Failure of $\bar{x}_T^i$ to lie on $\partial R(T)$ and/or failure of $\bar{p}_T^i$ to take the correct direction.

Regardless of the source of this difference, when a new p_o^{i+1} is selected, based upon $\bar{p}_o^i$, the validity of $\bar{p}_o^i$ should be examined as well as the possible benefit of adjusting for

$$\delta x_o^i = \bar{x}_o^i - x_o. \tag{3.126}$$

Disregarding these differences, a new approximation for p_o is

$$p_o^{i+1} = \bar{p}_o^i. \tag{3.127}$$

Consider the computational errors in integration which might develop. One possible means of reducing their effect would be to consider the following error correction vectors:

$$\alpha^i = \hat{x}_o^i - x_o \tag{3.128}$$

$$\beta^i = \hat{p}_o^i - p_o^i, \tag{3.129}$$

where $\hat{x}_o^i$ and $\hat{p}_o^i$ are initial points obtained by reverse time integration with extremal switching from p_T^i and x_T^i (i.e., without perturbation). The differences α^i and β^i thus result entirely from computational errors. These correction factors may then be used to give new estimates of x_o and p_o:

$$\tilde{x}_o^i = \bar{x}_o^i - \alpha^i \tag{3.130}$$

and

$$\tilde{p}_o^i = \bar{p}_o^i - \beta^i. \tag{3.131}$$

These new, hopefully better estimates can then be used to determine p_o^{i+1}. The effect of these correction factors is evaluated through experimentation in the next chapter. It should be noted that their computation at each step requires an extra integration of the state/costate equation pair.

Once $\tilde{x}_o^i$ and $\tilde{p}_o^i$ (or $\bar{x}_o^i$ and $\bar{p}_o^i$) have been computed, then a new p_o^{i+1} must be calculated. As mentioned, a straightforward method is to pick

$$p_o^{i+1} = \tilde{p}_o^i \text{ (or } \bar{p}_o^i). \tag{3.132}$$

Other alternatives are available such as:

$$p_o^{i+1} = p_o^i + c(\tilde{p}_o^i - p_o^i),\ c > 0 \tag{3.133}$$

where $c < 1$ if δx_o^i is large and $c \geq 1$ if the estimates are good.

3.7 The Local Optimum Procedure--Composite Method

Incorporating the results obtained in the previous sections of this chapter, an algorithm can be given to solve Problem 2.5 (LP). Included in this algorithm are several alternatives. Some are given in the form of subalgorithms, others are evident by the choice between several alternative equations. In Figure 3.5 a flow chart is given for LOP-CM and in Figure 3.6 a flow chart is given for an example subalgorithm (4b).

LOCAL OPTIMUM PROCEDURE--COMPOSITE METHOD (LOP-CM)

1. Select an error function
2. Select an arbitrary p_o^0, $(i = 0)$.
3. Determine x_T^i, p_T^i and $E(p_o^i)$ corresponding to x_o, p_o^i and extremal switching.
4. Determine an estimate for k through Subalgorithm 4.a or 4.b.
5. Perturb x_T^i by using Equation 3.80, 3.90 or 3.108 yielding $\bar{x}_T^i$), where the magnitude of c (step size) may be dependent on the error of the i-1'st iteration.
6. Perturb p_T^i in a corresponding manner (through Equation 3.78), giving $\bar{p}_T^i$.
7. Determine $\bar{x}_o^i$ and $\bar{p}_o^i$ corresponding to extremal switching, $\bar{x}_T^i$, $\bar{p}_T^i$ and reverse time integration.
8. Evaluate $\bar{x}_o^i$ and $\bar{p}_e^i$ and determine (through one of the Subalgorithms 8.a through 8.c) a new p_o^{i+1}.
9. Determine x_T^{i+1}, p_T^{i+1} and $e(p_o^{i+1})$. Test to see if the error value is decreased. If so, test to determine if the error value is sufficiently close to the optimum value. If it is sufficiently close, then a local optimum has been found; if not, go to step 4 and repeat. If $E(p_o^{i+1})$ is not an improvement, and if the step size has not been reduced past its reduction limits, decrease c and go to step number 5. If the step size can no longer be reduced, go to LOP-DS for final determination of the optimum.

SUBALGORITHMS 4--CURVATURE DETERMINATION

NOTE: For either of the following, the value finally selected for curvature may be given by Equations 3.120, 3.121, 3.123 or 3.124.

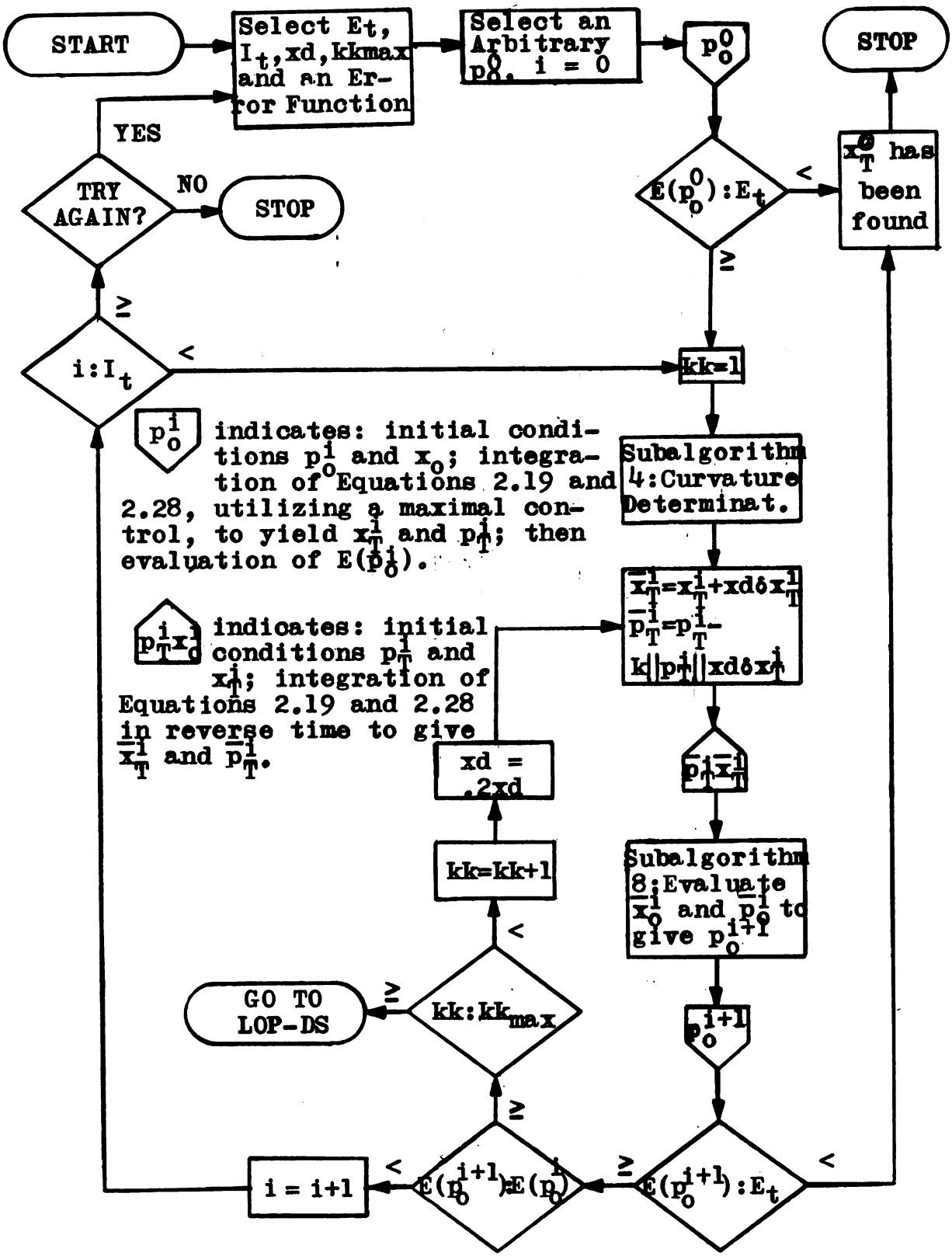


FIGURE 3.5 Flow Chart for LOP-CM

4.a.I. Perturb p_o^i resulting in $\tilde{p}_o^i$.

II. Determine $\tilde{x}_T^i$ and $\tilde{p}_T^i$ from x_o, $\tilde{p}_T^i$ and extremal switching.

III. If $\tilde{x}_T^i = x_T^i$, increase the perturbation size and go to step I. If not, go to step IV.

IV. Calculate an estimate of k from $\tilde{x}_T^i$, $\tilde{p}_T^i$, x_T^i and p_T^i.

4.b.I. Perturb p_o^i resulting in ${}_jp_o^i$, $j = 1$.

II. Determine ${}_jx_T^i$ and ${}_jp_T^i$ from x_o, ${}_jp_o^i$ and extremal switching.

III. If ${}_jx_T^i = x_T^i$, increase the perturbation size and go to step II. If not, go to step IV.

IV. Determine the standard deviation ${}_j\sigma_n = {}_j\sigma(\bar{k})/\bar{k}_a$ where ${}_j\sigma(\bar{k})$ is the standard deviation of the $\bar{k}_i$'s and $\bar{k}_a$ is the average of their absolute values. If ${}_j\sigma_n$ is less than σ_a (allowable limit for the standard deviation), then calculate an estimate of k. If ${}_j\sigma_n$ is greater than σ_a then go to step V.

V. Perturb p_o^i in a random manner from the previous perturbations, yielding ${}_{j+1}p_o^i$. Unless j+1 is greater than a preset limit on the number of allowed initial costate perturbations, go to step II and repeat steps II through V. In the event that j+1 exceeds the preset limit, use the value of p_o^i for which σ_n is a minimum to estimate k.

SUBALGORITHMS 8--NEW INITIAL COSTATE DETERMINATION

8.a. Utilize Equation 3.127, i.e. $p_o^{i+1} = \bar{p}_o^i$. do not correct for δx_o^i.

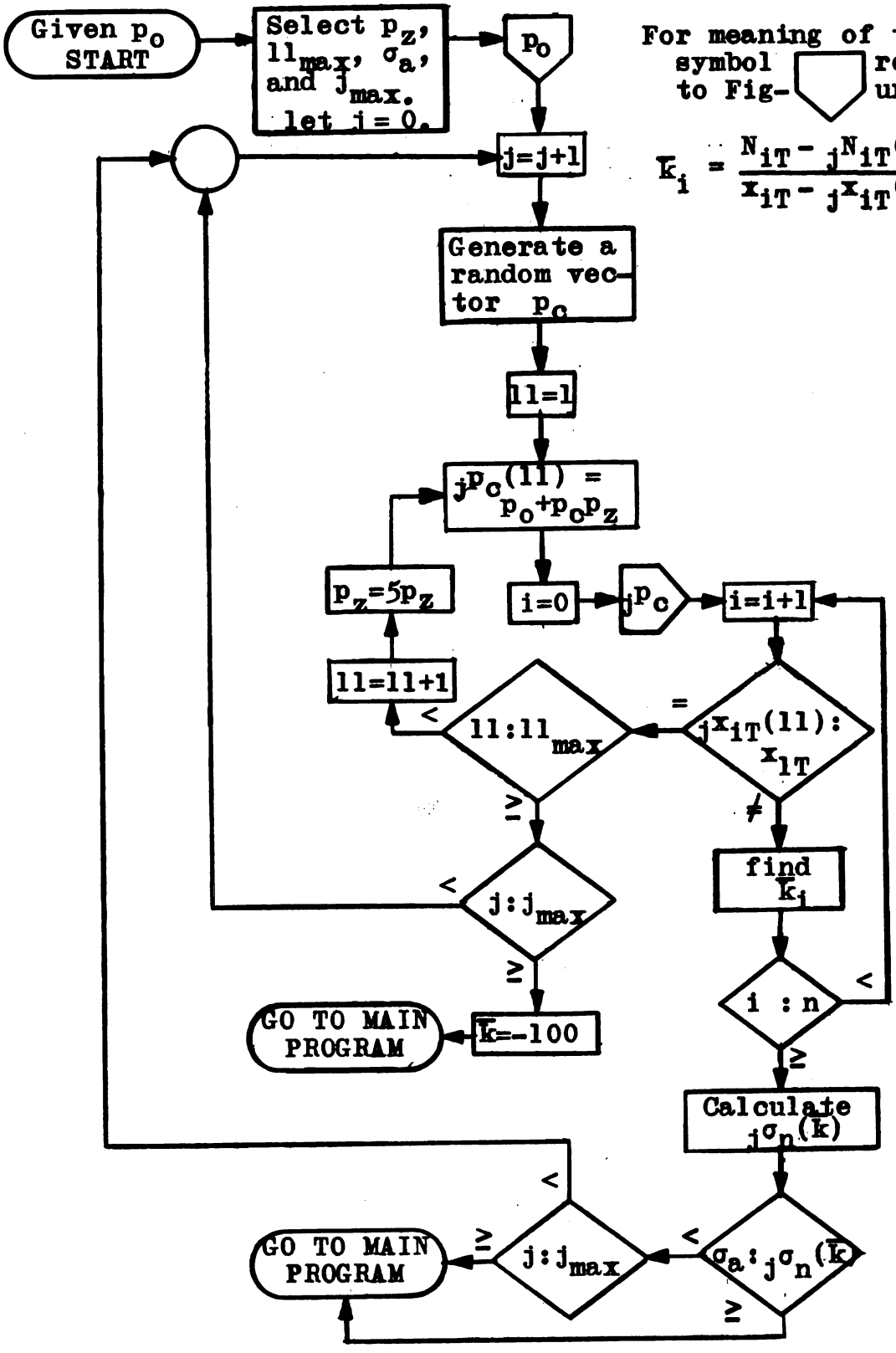


FIGURE 3.6 Flow Chart for Subalgorithm 4.b

8.b.I. Determine $\hat{x}_o^i$ and $\hat{p}_o^i$ corresponding to p_T^i and x_T^i (without perturbations) and extremal switching in reverse time integration.

II. Calculate the correction factors of Equations 3.128 and 3.129.

III. Correct $\bar{p}_o^i$ and $\bar{x}_o^i$ using Equations 3.130 and 3.131.

IV. Let $p_o^{i+1} = \hat{p}_o^i$. Ignore $\delta\hat{x}_o^i$.

8.c. Using either $\bar{x}_o^i$ and $\bar{p}_o^i$ or $\hat{x}_o^i$ and $\hat{p}_o^i$ (as determined in Subalgorithm 8.b), define a new p_o^{i+1} by using Equation 3.133.

Examination of the above algorithm and subalgorithms points out a number of possible alternatives within the basic algorithm. By way of summary these alternatives occur at:

1) Step 1. -- Error Function Selection
2) Step 4. -- Curvature Determination
3) Step 5. -- Final State Perturbation
4) Step 8. -- New p_o^{i+1} Estimation.

In addition to these, there are alternative step sizes to be chosen and also step 6 could be altered so that p_T^i would not be perturbed, or some of the other unrelated perturbations as discussed earlier (See Figure 3.2) could be selected. In Chapter 5 comparisons are made using these various alternatives.

CHAPTER 4

THE GLOBAL OPTIMUM AND RELATED PROBLEMS

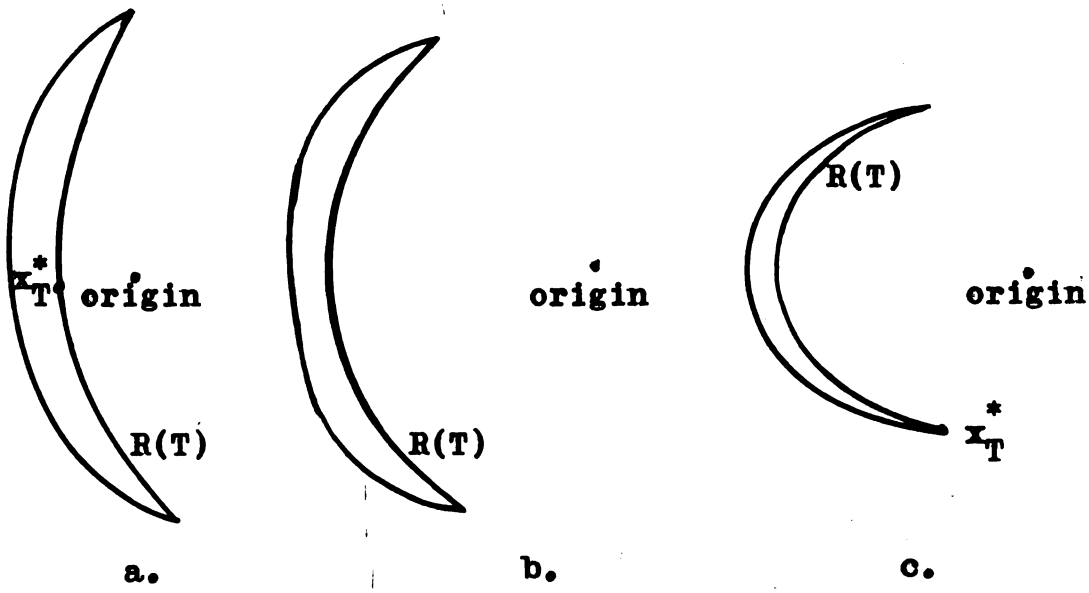
In the previous chapter, a method (including several options) was developed for determining the local optimum of a convex subset of R(T) (i.e. solving LP). Direct application of this method to an arbitrary R(T) might encounter one of several special cases for which the results of the previous chapter need to be reassessed. In this chapter, means of identifying and treating these are discussed. Most of these special cases are actually part of the more general problem of locating the global optimum; hence it is not necessary to implement specialized techniques for their solution.

The more general problem of determining the global optimum for an arbitrary R(T) which may have several or even numerous local optima is most difficult. In this thesis a random approach is taken for finding the local optima and thus identifying a global optimum. This global optimum procedure solves Problem 2.4 (MP). Using some of the concepts of Fadden and Barr [F1 and B1], other types of optimal control problems can also be solved. Of these, only the time optimal control problem is considered here. All experimental results and example problems are presented in the next chapter.

4.1 Special Problem 1: Nonconvex regions on $\partial R(T)$

The algorithm presented in Chapter 3 assumes that the region on $\partial R(T)$ being considered is convex. For an arbitrary reachable set there may be many local optima (i.e. locally closest points to the origin). Much of the surface may not be convex and, in fact, some of these local optima and even the global optimum may lie on a non-convex region. Such regions may be concave or may be "mixed" (saddle points, neither concave nor convex).

Upon first consideration it may seem that it is not possible for an optimum to lie on a concave region of $\partial R(T)$. This, however, is not the case. Consider, for example, the reachable set shown in Figure 4.1a, with the origin located as indicated. Since all of the surface (in this case a curve--R(T) is 2-dimensional) near the origin is concave, the optimum is obviously on a concave region, namely at x_T^*. A very special case is shown in Figure 4.1b, where the optimum is not unique; in fact, where there are an infinite number of minima, all equally close to the origin. As a final example, Figure 4.1c is given in which the optimum is located at a "corner". Note that the determining factor in all three examples is the relationship between the radius of curvature of the concave surface and the distance of that surface from the origin. This observation leads to the following theorem which is given after some preliminary definitions.



a. b. c.

FIGURE 4.1 Optima on Concave Boundary Regions

DEFINITION 4.1 Let $\partial R(T)$, $x(T)$ and k be as previously defined. Let the principal curvatures k_i, $i = 1,\cdots,n-1$, be nonzero, then the principal radii of curvature, ρ_i, at $x(T)$ are defined as

$$\rho_i = \frac{1}{k_i} . \tag{4.1}$$

DEFINITION 4.2 Let $\partial R(T)$, $x(T)$ and k_i be as previously defined, then the maximum convex curvature, K_{vx}, at $x(T)$ is defined as

$$K_{vx} = \min_i k_i \text{ , if any } k_i < 0, \tag{4.2}$$

or

$$K_{vx} = 0, \text{ if all } k_i \geq 0. \tag{4.3}$$

DEFINITION 4.3 Let $\partial R(T)$, $x(T)$ and k_i be as previously defined, then the maximum concave curvature, K_{cv}, at $x(T)$ is defined as

$$K_{cv} = \max_i k_i, \text{ if any } k_i > 0, \tag{4.4}$$

or

$$K_{cv} = 0, \text{ if all } k_i \leq 0. \tag{4.5}$$

THEOREM 4.1 Let R(T) be a compact set in E^n, $0 \notin R(T)$. Consider a point $x(T) \in \partial R(T)$ for which $\cos \gamma = -1$, where p(T) is defined as the outward normal to R(T) at x(T). Let $N_\partial(x,\epsilon)$ be a neighborhood of x(T) on $\partial R(T)$. Let k_i, $i = 1,\cdots,n-1$ be the principal curvatures at x(T).

a. If

$$K_{cv} > \frac{1}{\| x_T \|}, \tag{4.6}$$

then x(T) is not a local optimum; i.e., there exists a $y \in N_\partial(x,\epsilon)$ such that

$$\| y \| < \| x(T) \|. \tag{4.7}$$

b. If

$$K_{cv} = \frac{1}{\| x_T \|}, \tag{4.8}$$

then x(T) is not a unique optimum; i.e. there exist many $z \in N_\partial(x,\epsilon)$ such that

$$\| z \| = \| x(T) \|. \tag{4.9}$$

c. Finally, if

$$K_{cv} < \frac{1}{\| x_T \|}, \tag{4.10}$$

then x(T) is a local optimum; i.e. for any $w \in N_\partial(x,\epsilon)$,

$$\| w \| > \| x(T) \|. \tag{4.11}$$

Proof: Consider part a. first. There exists at least one principal curvature, namely

$$k_{cv} = K_{cv} \tag{4.12}$$

for which

$$k_{cv} > \frac{1}{\|x(T)\|} . \tag{4.13}$$

Consider the line of curvature, L_{cv}, through x(T) corresponding to k_{cv}. Since L_{cv} is a line of curvature, Equation 3.66 applies. Since cos γ at x(T) is -1, Equation 3.66 reduces to:

$$\frac{d \cos \gamma}{ds} = -(k_{cv} - \frac{1}{\|x(T)\|}) \frac{d \|x(T)\|}{ds} . \tag{4.14}$$

By Equation 4.13 this reduces to

$$\frac{d \cos \gamma}{ds} = - (\lambda) \frac{d\|x(T)\|}{ds} , \tag{4.15}$$

where λ is a positive number. If an infinitesimal change is made in x(T) along L_{cv},

$$\frac{d \cos \gamma}{ds} > 0 \tag{4.16}$$

since cos γ is a minimum at x(T). Therefore by Equation 4.15,

$$\frac{d \|x_T\|}{ds} < 0. \tag{4.17}$$

Let y be a point on L_{cv} infinitesimally close to x(T); thus by Equation 4.17, Equation 4.7 is true and part a. is proven.

Now consider part b. If for one curvature,

$$k_{cv} = K_{cv} = \frac{1}{\|x(T)\|} , \tag{4.18}$$

then at x(T),

$$\frac{d \cos \gamma}{ds} = 0, \tag{4.19}$$

over a segment of L_{cv}, hence the optimum is not unique.

Finally consider part c. In this case, the number λ of Inequality 4.15 is negative, regardless of the choice of principal curvature. Since the principal curvatures represent the extreme values which curvature can take, K_{cv} represents the maximum value for curvature on the surface in the neighborhood of x(T). Since Equation 4.16 holds and λ is negative,

$$\frac{d\|x(T)\|}{ds} > 0 \qquad (4.20)$$

and x(T) is the local minimum. Hence the proof of the theorem.

It is interesting to note that the important relationship between k and $1/\|x(T)\|$ for concave surfaces is confirmed by Equation 3.72 for the composite error function E_5. In this case the requirement on k for monotonicity of the error function is the same as that for the existence of a local optimum on a concave surface (part c of Theorem 4.1).

Since the object of this chapter is to eventially adapt LOP-CM to the global problem, consider the results of Theorem 4.1 as they apply to LOP-CM. First of all, it is evident that any minimum of cos γ must be verified if even one of the principal curvatures is positive. In this event, one of the alternatives suggested by Theorem 4.1 applies. Thus, consider the following decision scheme for computation based on these alternatives:

1) If Equation 4.6 applies, the apparent optimum must be a "false" optimum. Switching to $E_1(p_o) = \| x(T) \|$ effectively overcomes this difficulty. Note that this circumstance is improbable since the nature of LOP-CM, particularly the method of choosing δx, contradicts Equation 4.7. The chance of an arbitrary trajector, $x^0(\cdot)$ yielding this special case is likewise remote.

2) If Equation 4.8 applies, the optimum is not unique, but computationally x(T) is one of the (infinite number of) local optima.

3) If Equation 4.10 applies, then no computational change need be made since x(T) is the local optimum.

It is appropriate to make an observation at this point. Considering the fact that a convex reachable set results from a linear system and considering the very nature of the optimization problem with $0 \notin R(T)$, it is reasonable to expect that the global optimum usually lies on a convex, or at worst, a mixed (saddle-point) surface. Thus if an initial guess for p_o yields a boundary point where Equation 4.6 applies, a major change or jump in the initial costate might be most effective rather than a routine application of LOP-CM.

In the discussion given above and in Chapter 3 only convex or concave surfaces have been considered. The results, however, apply to mixed surfaces which are neither concave nor convex. For example, if Equation 4.10 holds (which allows the possibility of negative principal curvatures) then it is quite possible that the optimum lies on a mixed region. For such a region, examination of

Equations 3.66 and 3.72 indicates that if a choice of path is available, based on the value of curvature, then the principal curve corresponding to the minimum value for k should be chosen. Since the actual computational method of perturbation from iteration to iteration is based on the δx's, the value of k only enters in the computation of δp.

4.2 Special Problem 2: Flats and Corners

In the application of many reachable set-oriented computational methods, two surface characteristics are particularly troublesome. These are the flat, for which one outward normal corresponds to more than one adjacent boundary point, and the corner, for which a unique normal plane is not defined to the surface at that point. These are illustrated in Figure 4.2. The normal plane to a flat is said to be nonregular and the corner is termed a nonregular point.

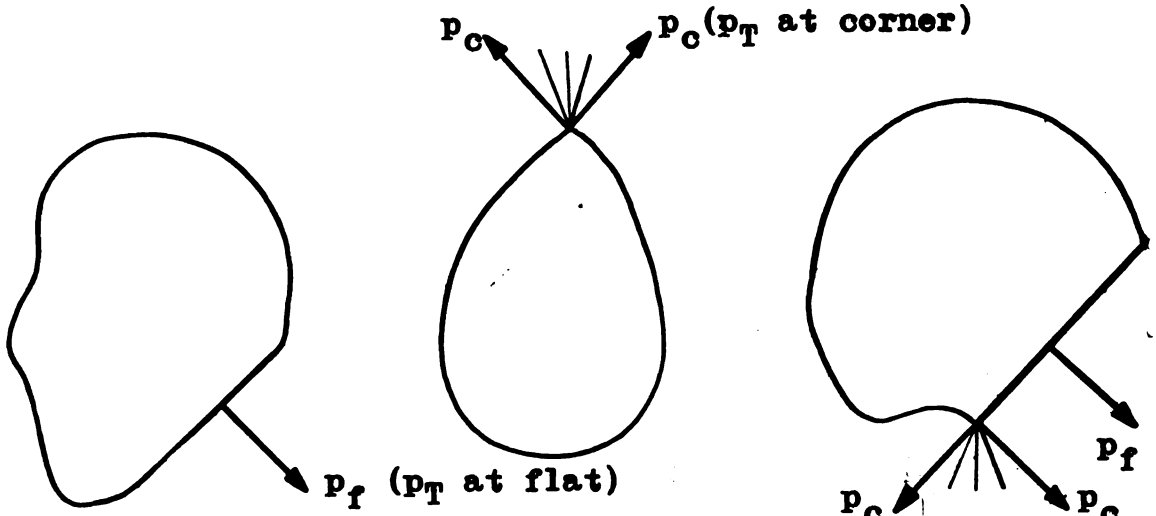


FIGURE 4.2 Flats and Corners on Surfaces

Consider first the flat. In many computational methods which determine contact points [B2] corresponding to a specific p(T), sets such as that shown in Figure 4.2a and 4.2c are especially difficult to handle since many boundary points correspond to the outward normal p_f. In Section 2.7 it was indicated that the flats result from singular controls in linear systems. For nonlinear systems, however, flats may occur even though the problem is nonsingular. If the flat is not the result of singular controls, then it does not present special problems for the computational method introduced in Section 3.7. Although the direction of p(T) is constant for a region on $\partial R(T)$ (i.e. k is zero), there is a change in the final state corresponding to a change in the initial adjoint, hence any of the previously introduced error functions differentiate between boundary points even though the outward normals are the same.

If the flat results from a singular control, then <u>one</u> initial adjoint corresponds to all of the points on the flat. Thus if only the maximal values of the control are assigned at singular points, only the boundary points of the flat are determined, resulting in the singularity gap mentioned in Section 2.7. This gap is only significant if the global optimum is on the flat. In the event that this optimum is the only optimum, then the failure of LOP-CM to converge would identify this special case. If, on the other hand, other optima exist, then a global technique

would incorrectly identify one of these optima as the global optimum. The possibility of such an occurrence is slight but must be considered.

Short of restricting LOP-CM and its global modification only to normal problems, there is an additional alternative which can be taken. Any minima (but not optima) located by a global procedure would have to be carefully examined to verify that they are not boundary points of a flat, and that $||x_T||$ for these points is not, in fact, less than $||x_T||$ for the supposedly global optimum. While this approach does not guarantee the identification of a global optimum on a flat, it does reduce the already small possibility that such an optimum is overlooked.

One final observation should be made. For linear systems the adjoint equation is fixed regardless of the state trajectory; thus to each p_0 there corresponds one p_T even though the state trajectories may vary (due to singularities). This, of course, results in a flat on $\partial R(T)$. For nonlinear systems, however, the adjoint equation is dependent on the state trajectory, thus an initial adjoint which results in singular controls usually would produce differing final adjoints since the state trajectories vary. While no conclusion can be drawn, this would seem to indicate that flats caused by singular controls are less likely for nonlinear systems than for linear systems.

The second special case to consider is the corner. Whereas the flat spot results from a boundary section for which p(T) is constant in direction, a corner results from a boundary point which is constant for a number of final adjoints (hence initial adjoints). In this event no particular computational difficulty is introduced for LOP-CM if any of the error functions including the cos γ factor are used. Since the final adjoints vary, the error function changes even though the final state does not change. Note that the effective curvature in this case is infinite, thus indicating a perturbation in p(T) but no corresponding perturbation in x(T).

4.3 Special Problem 3: Extremum but Not Local Optimum

If error function $E_1 = \| x(T) \|$ is used, any minimum of the error function must also indicate a local optimum. For the other error functions, however, a minimum may be obtained but may not be the optimum value. For instance, cos γ may reach a minimum but not the optimum value of -1. This possibility is illustrated in Figure 4.3. While LOP-DS or LOP-CM may converge to such a point, no particular computational problem would result since any global procedure excludes the choice of such an extremum as the global optimum. In the event that convergence to such an extremum is to be prevented, $E_1(p_o)$ is utilized.

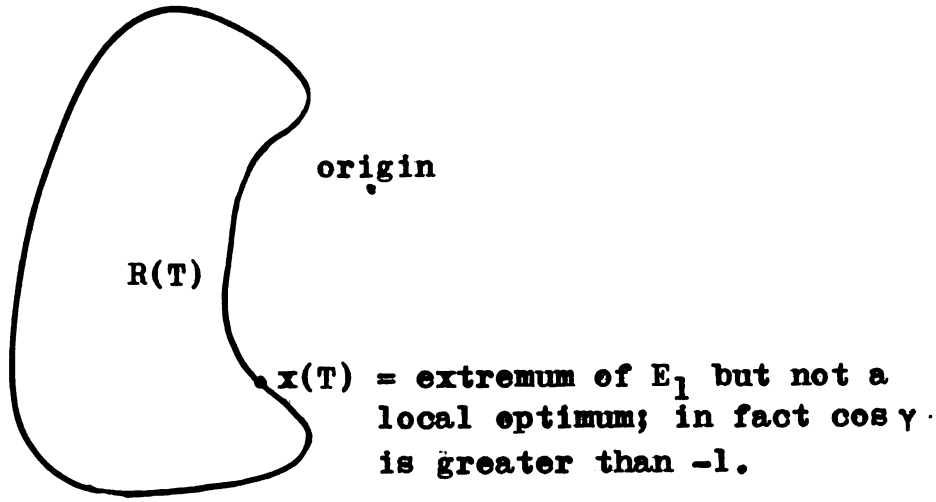


FIGURE 4.3 Extremum but Not a Local Optimum for Cos γ

4.4 Special Problem 4: Internal Boundaries

Consider the complement of R(T):

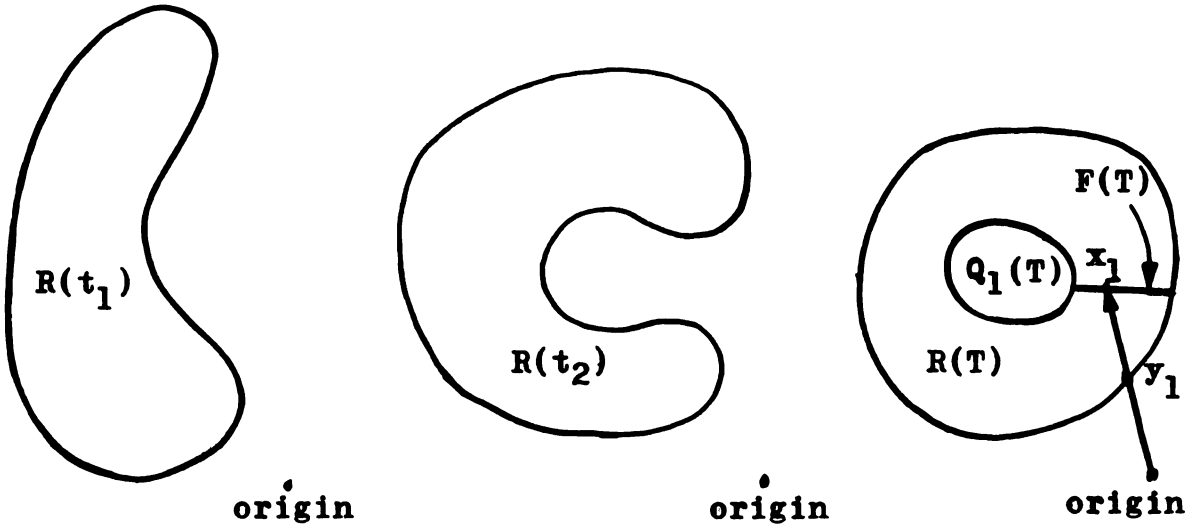
$$Q(T) = [\ x \in E^n : x \notin R(T)\ ]. \tag{4.21}$$

Usually Q(T) is connected; however, a situation as shown in Figure 4.4c is certainly possible. Designate the portion of Q(T) which contains the origin as $Q_0(T)$ and index the remaining disjoint subsets of Q(T) as $Q_i(T)$, i=1,···,k.

DEFINITION 4.4 The i^{th} internal boundary, $\partial_i R(T)$, of R(T) is the subset of $\partial R(T)$ defined by

$$\partial_i R(T) = \partial R(T) \cap \partial \bar{Q}_i(T). \tag{4.22}$$

Several observations can be made concerning Q_1 of Figure 4.4c. Because of Theorem 2.5, it is known that any $x \in \partial_1 R(T)$ was a boundary point for all reachable sets R(t), $t < T$; thus one can assume that the reachable set shown in Figure 4.4c evolved in a manner similar to that illustrated in Figures 4.4a and 4.4b.



a. $t = t_1 < t_2 < T$ b. $t = t_2 < T$ c. $t = T$

FIGURE 4.4 Development of an Internal Boundary

The global optimum (or optima, if not unique) must lie on $\partial_0 R(T)$:

THEOREM 4.2 Let R(T) be a compact set for which Q(T) is not connected. Let $Q_0(T)$ be the connected subset of Q(T) which contains the origin. Then any global optimum belongs to $\partial_0 R(T)$.

Proof: Let $x^* \in \partial R(T)$ be a global optimum. Assume that $x^* \in \partial_j R(T)$, $j \neq 0$. Thus $x^* \in \overline{Q}_j(T)$, $j \neq 0$. But the origin belongs to $Q_0(T)$, thus there is no path lying entirely in $\overline{Q}_j(T)$ from x^* to the origin. Consider the line from x^* to the origin. Since for any $i \neq j$, $Q_i(T)$ and $Q_j(T)$ are disjoint, part of the line must lie in R(T). But this contradicts the assumption that x^* is a global optimum, hence $x^* \notin \partial_j R(T)$, $j \neq 0$. The only portion of R(T) remaining is $\partial_0 R(T)$ and since $x^* \in \partial R(T)$, $x^* \in \partial_0 R(T)$; hence the proof.

Note that the theorem is proven for a global optimum, x^*. A local optimum may lie on an internal boundary (See Figure 4.5) and thus may be determined by LOP-CM. Although a means of identifying such an optimum is available (i.e., examination of the line segment from x^+ to the origin), such an approach is not necessary since a global technique cannot select such local optima as global optima. Thus, no specialized algorithm is given to differentiate these local optima from other local optima.

4.5 Special Problem 5: False Boundary Points

As has previously been indicated, all extremal controls are maximal controls but the converse does not hold. As a result, "false boundary points" (fbp) are generated.

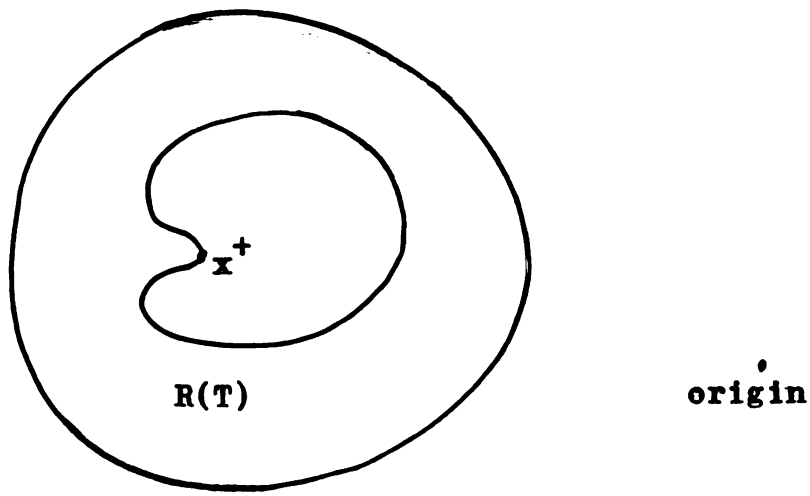


FIGURE 4.5 A Local Optimum on an Internal Boundary

DEFINITION 4.5 A false boundary point is a maximal endpoint which is not extremal (i.e., $x \notin \partial R(T)$).

It can be seen that the situation depicted in Figure 4.4 can result in false boundary points. The evolution of the reachable set is such that the points lying on the line F(T) in Figure 4.4c are false boundary points.

Since LOP-CM utilizes maximal controls with the anticipated result that $x(T) \in \partial R(T)$, false boundary points could cause computational difficulties. If LOP-CM were required to converge only to true (i.e., not fbp's) local optima, some specialized technique would be needed to identify and treat the case of false boundary points. Such a method could be based on the following facts:

1) Each fbp (e.g., $x \in F(T)$ in Figure 4.4c) previously existed as a boundary point for some $t_1 < T$.

2) At some time t_i, for each $x_i \in F(T)$, the state-costate pairs $(x_i(t_1),p_i(t_1))$ and $(x_i(t_1),-p_i(t_1))$ represent true boundary points.

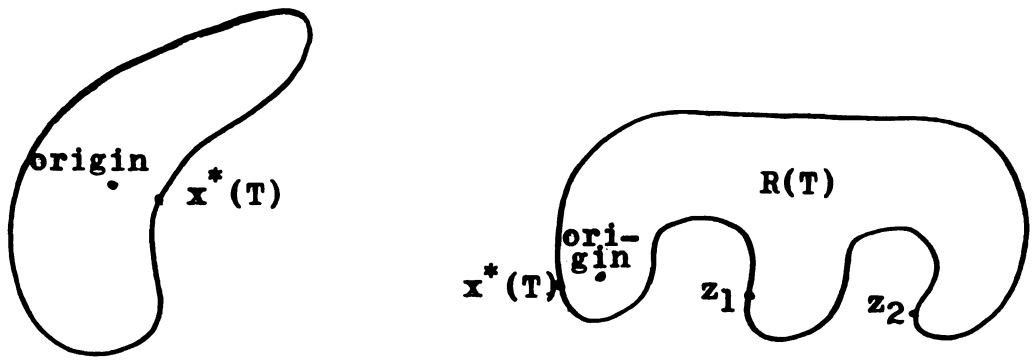
3) For each $x \in F(T)$ there exists a y, $y = cx$, $0<c<1$, which is a boundary point of R(T). An illustration of this is given in Figure 4.4c, points x_1 and y_1.

In as much as the ultimate objective is the determination of the global optimum, specialized techniques are not needed. Although LOP-CM might converge to a false boundary point, any global optimum procedure would not choose this false boundary point as an x^* since it is not a local optimum and certainly not a global optimum.

4.6 Special Problem 6: Origin Interior to R(T)

All of the discussion to this point has assumed that the origin is exterior to the reachable set. On the other hand, the system definition is general and since little is known, prior to computation, of the reachable set, it is quite possible that $0 \in R(T)$. Some effective means of testing this possibility is desirable. This is especially true in the case of origin-seeking, time optimal controls where the optimum is attained when the origin first belongs to the boundary of the reachable set.

For many reachable sets, an interior origin would readily be identified since boundary points satisfying minimum distance considerations would have +1 as the value for $\cos \gamma$ (See Figure 4.6a). Other possibilities, however, can occur, such as that illustrated in Figure 4.6b, in which local optima z_1 or z_2 would give no indication that the origin belongs to R(T). While specialized techniques can be developed to test each local optimum to see if $0 \in R(T)$, these are not necessary in a global technique. In fact, if the closest point to the origin has been determined, then it is simply necessary to test that point to ascertain if $\cos \gamma$ is positive or negative. If it is -1, the optimum has been determined, if not, the origin must belong to R(T). This result is presented in the following theorem.



a. b.

FIGURE 4.6 Reachable Sets with Interior Origins

THEOREM 4.3 Let R(T) be a compact set in E^n. Let $x^*(T)$ be a global boundary optimum, i.e.

$$\| x^*(T) \| \leq \| y(T) \| \text{ for any } y \in \partial R(T), \tag{4.23}$$

with corresponding outward normal $p^*(T)$. Then $x^*(T)$ is the optimum for MP if

$$< x^*(T) , p^*(T) > = -1. \tag{4.24}$$

If

$$< x^*(T) , p^*(T) > = +1, \tag{4.25}$$

then $x^*(T)$ is not the optimum for MP and $0 \in R(T)$.

Proof: Consider the first part of the theorem. Since $x^*(T)$ is the global boundary optimum, it must only be shown that $0 \notin R(T)$ to establish that $x^*(T)$ is the global optimum. Since Equation 4.24 holds, $x^*(T)$ and $p^*(T)$ must be collinear but oppositely directed. Let $L(x^*(T),0)$ denote the line segment from $x^*(T)$ to the origin. Since the final adjoint, $p^*(T)$, is directed outward from R(T) (i.e., from $x^*(T)$) along $L(x^*(T),0)$, points on L near $x^*(T)$ must lie exterior

to R(T). If this line were to intersect R(T) then it would include at least one point on $\partial R(T)$, closer to the origin than $x^*(T)$. Since this contradicts Equation 4.23, the origin must be external to R(T) and thus the global boundary optimum $x^*(T)$ is the optimum for MP.

Now consider the second part of the theorem. If Equation 4.25 holds true, then $x^*(T)$ and $p^*(T)$ must be collinear and similarly directed. Since $p^*(T)$ is outward to the reachable set, points on the line $L(x^*(T),0)$ near $x^*(T)$ must lie in R(T). Thus the origin must lie internal to R(T) or the line $L(x^*(T),0)$ would intersect $\partial R(T)$. If the latter were the case, then Equation 4.23 would be contradicted, thus the origin must belong to the reachable set. Hence there is a trajectory endpoint (namely $x(T) = 0$) which is closer to the origin than $x^*(T)$; hence $x^*(T)$ is not the optimum for MP.

4.7 The Global Optimum

In the previous sections, special problems have been suggested as possible difficulties in determining an optimum. Careful consideration, however, has demonstrated that these special problems are really part of the global problem. In as much as an explicit equation is not available for $\partial R(T)$, the global problem is very difficult, especially for high order, nonlinear systems.

Several possible approaches can be taken to extend LOP-CM to determination of global optima. Once a local

optimum, x^+, has been located, then the global optimum must be located within a hypersphere of radius $||x^+||$. One method of determining a global optimum would thus be to investigate all possible final states within this radius.

Since extremal endpoints, and thus global optima, are associated with initial adjoints (assuming normality), another approach is to consider the set of all possible initial adjoints. Since only the magnitude of the adjoint (hence initial adjoint--see Equation 3.37) is important:

$$\max_{u \in U} \langle cp , f \rangle = \max_{u \in U} \langle p , f \rangle, \quad 0 < c \in E^1, \tag{4.26}$$

a search on a hypersphere in E^n of arbitrary radius, would likewise locate global optima.

The second approach is chosen here since it involves a search <u>on</u> a hypersphere in E^n rather than a search <u>within</u> a hypersphere in E^n. Thus a sequence of <u>random</u> initial adjoints of constant norm is generated and LOP-CM is used to converge to a local optimum corresponding to each random initial adjoint. The resulting set of local optima with associated starting initial adjoints is then examined and the global optimum (or optima) is identified. Since it is not feasible to exhaustively search all possibilities, the degree of confidence in the solution to the global problem is directly related to the amount of time which one is willing to allocate to the computation.

The approach presented above is summarized in the following procedure:

GLOBAL OPTIMUM PROCEDURE (GOP)

1. Select a sequence of random starting initial adjoints, N terms in the sequence.
2. Utilize LOP-CM for each starting initial adjoint.
3. Compare all local optima thus obtained (${}^{i}x^{+}$, $i = 1,\cdots,N$. Let the global optimum (optima) be chosen such that

$$\| x^{*} \| = \min_{i} \| {}^{i}x^{+} \|. \tag{4.27}$$

Examples of the application of this procedure are given in the next chapter.

4.8 Application of GOP to Time Optimal Control Problems

In the previous section an algorithm is given to determine the global optimum for MP (i.e. solve the minimum distance problem). It is the purpose of this section to apply this method to the determination of the origin seeking time optimal control. This problem can be stated as follows:

PROBLEM 4.1 Given: the system (Equation 2.19), the class F of admissible control functions, and the performance functional

$$J(T) = T - t_{0}; \tag{4.28}$$

Find a control function $u^{*}(\cdot)$ in F which minimizes J(T) while satisfying Equation 2.19 and the final condition $x(T) = 0$.

Since GOP determines the closest point to the origin on $\partial R(T)$ for a specified time T, it is apparent that a

series of iterations, allowing t to incrementally increase at each iteration, would be one approach for solving the time optimal control problem [F1,B1]. An optimal control exists if there is a control which results in a trajectory terminating at the origin. If R(t) is compact and continuous (Theorem 2.4) the control is an extremal control and the optimal time t^* (time at which the origin first belongs to R(T)) can be determined and corresponds to the time at which the origin first belongs to $\partial R(T)$, i.e., $\partial R(t^*)$.

Thus consider the following time optimal control algorithm which includes several alternative choices:

TIME OPTIMUM PROCEDURE (TOP)

1. Let $T^0 = t_0$; choose an initial δT^0. Pick an arbitrary initial costate p_0^0. Let i = 1 and $T^1 = T^0 + \delta T^0$.

2. Make an initial iteration using LOP-DS, LOP-CM or GOP to determine an initial optimum ${}^1x(T^1)$. If ${}^1x(T^1) \approx 0$, the problem is solved. If $0 \in R(T^1)$, then decrease δT^0 and repeat. Otherwise proceed to step 3.

3. Increment time, $T^{i+1} = T^i + \delta T^i$, where δT^i may be constant or may be a variable dependent on $E(p_0)$.

4. Starting at an arbitrary initial costate, at the ith-optimum ${}^ip_0^*$ (optimum initial costate for the previous time iteration) or an initial costate determined in some other manner, use LOP-DS, LOP-CM or GOP to determine ${}^{i+1}x(T^{i+1})$. If ${}^{i+1}x(T^{i+1}) \approx 0$, the problem is solved. If $0 \in R(T^{i+1})$ then decrease δT^i and repeat. Otherwise, let i = i+1 and repeat steps 3 and 4.

Examination of the above algorithm demonstrates a number of possible alternatives within TOP:

1) Step 2--Choice of LOP-DS, LOP-CM or GOP to determine the initial optimum ${}^{1}x(T^{1})$.

2) Step 3--Method of choosing δT^{i}:
   a) $\delta T^{i} = h$, a constant, or
   b) δT^{i} = some function of E, the error function.

3) Step 4--Method of selecting the $i+1^{st}$ starting initial costate:
   a) Choose an arbitrary initial costate, or
   b) Choose the preceding optimal initial costate, ${}^{i}p_{0}^{*}$.

4) Step 4--Choice of LOP-DS, LOP-CM or GOP for determining the $i+1^{st}$ optima, $i \geq 0$.

Consider the alternatives given above. As was the case for GOP, there is a tradeoff between the reliability of the method and the amount of allowable computational time. If GOP is utilized at each time increment and if the time increments are made sufficiently small, then there is confidence in the choice of t^{*}. On the other hand, to be realistic, some compromise in computations must be made. For example, GOP may just be utilized at the initial iteration and as a check of the final result.

In as much as the step size alternatives (step 3) are not difficult to implement, the choice should be based entirely on the effectiveness of the two alternatives. For the alternative given in 3), it is logical to use prior information, rather than starting with an arbitrary initial

costate. In fact, if δT^i is kept reasonably small, the new optimum initial costate ${}^{i+1}p_o^*$ may be very close to ${}^{i}p_o^*$, indicating that LOP-DS is a good candidate for use in alternative 4).

Caution, however, must be exercised in attempting to reduce the computation time. Even though a T^j has been determined for which $0 \in \partial R(T^j)$ and even though $0 \notin R(T^{j-1})$, it is possible that there existed a $T^k < T^j$ for which $0 \in \partial R(T^k)$. If the step size were too large or if an incorrect, local optimum was considered the global optimum at each time increment, then this situation could occur.

One final observation should be made. The successive values of final states at each iteration, ${}^{i}x(T^i)$ do not necessarily approach the origin monotonically. It is quite possible that an optimum final state ${}^{j}x(T^j)$ has the property that

$$\| {}^{j}x(T^j) \| > \| {}^{j-1}x(T^{j-1}) \|, \qquad (4.29)$$

even though the sequence of ${}^{i}x(T^i)$'s converges to 0. Examples of this as well as other aspects of TOP are given in the following chapter.

CHAPTER 5

COMPUTATIONAL RESULTS AND CONCLUSIONS

In the preceding chapters, general problems, methods and alternative choices have been given which can effectively be evaluated utilizing computational results. It is the purpose of this chapter to provide these results and to form conclusions based on the data obtained.

Specifically, example systems are detailed and discussed in the first section. These systems are then used as examples in the remaining sections of the chapter. These computational examples are introduced with several purposes in mind:

1) To give insight into the nature of the problems.

2) To present the resulting reachable sets and give example extremal trajectories.

3) To use these examples in comparing the various computational alternatives previously introduced.

4) To demonstrate that the previously introduced algorithms are effective in computing optima.

In particular, the author feels, based on personal experience, that example nonlinear problems with resulting reachable sets should be given for the benefit of others wishing to consider this type of problem.

As the computational results are presented and analyzed, comparisons are made and conclusions are drawn. These conclusions are listed in the various sections of this chapter and are summarized in the final section.

5.1 Example Systems

In this section several example systems are introduced. For each of these systems, a differential equation is given, state and costate equations are introduced and maximal control is considered. For simplicity, these results are presented in outline form.

EXAMPLE SYSTEM #1 (ES-1)

a. Differential Equation

$$\frac{d^2x(t)}{dt^2} = -.01\,[x(t)\,|x(t)|\,] + .1[\frac{dx(t)}{dt}u(t)] \quad (5.1)$$

b. State Equations ($x = x_1$)

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}0 & .1\\ -.1|\,x_1\,| & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ .1x_2\end{bmatrix} u \quad (5.2)$$

c. Hamiltonian

$$H = .1p_1x_2 - .1p_2|\,x_1\,|x_1 + .1p_2x_2u \quad (5.3)$$

d. Adjoint System

$$\begin{bmatrix}\dot{p}_1\\ \dot{p}_2\end{bmatrix} = \begin{bmatrix}0 & .2|\,x_1\,|\\ -.1 & -.1u\end{bmatrix}\begin{bmatrix}p_1\\ p_2\end{bmatrix} \quad (5.4)$$

e. Control

i. Restraint Set: $|u| \leq 1$

ii. Maximal Switching:

$$u = \text{sgn}\,(.1x_2p_2) = \text{sgn}\,(x_2p_2) \quad (5.5)$$

f. An analog diagram for this system is given in Appendix A.

EXAMPLE SYSTEM #2 (ES-2)

a. Differential Equation:

$$\frac{d^3x}{dt^3} = -x\frac{d^2x}{dt^2} + \left(\frac{dx}{dt}\right)^2 + u_1 + x\,u_2 + \frac{du_2}{dt} \qquad (5.6)$$

This system equation was derived from a well-known equation of fluid dynamics known as Schlichting's Equation (Equation 5.6 becomes Schlichting's Equation if $u_1 = 1$ and $u_2 = 0$). In general, equations of this type represent fluid flow across surfaces. With the addition of u_1, flow across a wedge is indicated. [K3,S1].

b. State Equations ($x_1 = x$)

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix} = \begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ -x_3 & x_2 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + \begin{bmatrix}0 & 0\\ 0 & 1\\ 1 & 0\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix} \qquad (5.7)$$

c. Hamiltonian

$$H = p_1x_2 + p_2x_3 + p_2u_2 - p_3x_3x_1 + p_3x_2^2 + p_3u_1 \qquad (5.8)$$

d. Adjoint System

$$\begin{bmatrix}\dot{p}_1\\ \dot{p}_2\\ \dot{p}_3\end{bmatrix} = \begin{bmatrix}0 & 0 & x_3\\ -1 & 0 & -2x_2\\ 0 & -1 & x_1\end{bmatrix}\begin{bmatrix}p_1\\ p_2\\ p_3\end{bmatrix} \qquad (5.9)$$

e. Control

i. Restraint Set: $|u_i| \leq 1$, $i = 1,2$

ii. Maximal Control:

$$[u] = \begin{bmatrix}p_2\\ p_3\end{bmatrix} \qquad (5.10)$$

EXAMPLE SYSTEM #3 (ES-3) [K3, page 295]

a. Differential Equation (for the system shown in Figure 5.1)

$$\frac{d^3x}{dt^3} = -\frac{d^2x}{dt^2} - x^2\frac{dx}{dt} + u(t) \tag{5.11}$$

b. State Equations ($x = x_1$)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -x_1^2 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u \tag{5.12}$$

c. Hamiltonian

$$H = x_2p_1 + x_3p_2 - x_1^2x_2p_3 - x_3p_3 + up_3 \tag{5.13}$$

d. Adjoint System

$$\begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \\ \dot{p}_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 2x_1x_2 \\ -1 & 0 & x_1^2 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \tag{5.14}$$

e. Control

i. Restraint Set: $|u| \leq 1$

ii. Maximal Control:

$$u = \text{sgn}\,(p_3) \tag{5.15}$$

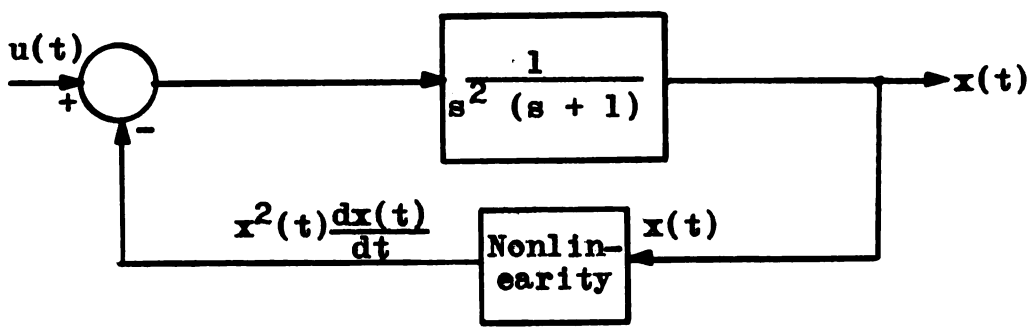


FIGURE 5.1 Example System #2--3rd Order, Nonlinear

EXAMPLE SYSTEM #4 (ES-4) [K3, page 296]

a. **Differential Equation**

$$\frac{d^4x(t)}{dt^4} = -2\frac{d^3x(t)}{dt^3} - g[x(t)] + u(t) \tag{5.16}$$

$$g[x(t)] = -\tfrac{1}{2}x(t) - \frac{2}{15}x(t)^2 + \frac{1}{750}x(t)^4 \tag{5.17}$$

b. **State Equations** ($x = x_1$)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{1}{2} + \frac{2x_1}{15} - \frac{x_1^3}{750} & 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ u_2 \\ u_1 \end{bmatrix} \tag{5.18}$$

c. **Hamiltonian**

$$H = p_1x_2 + p_2x_3 + p_3x_4 + \tfrac{1}{2}p_4x_1 + \frac{2}{15}p_4x_1^2 - \frac{1}{750}p_4x_1^4 - 2x_4p_4 + p_4u_1 + p_3u_2 \tag{5.19}$$

d. **Adjoint System**

$$\begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \\ \dot{p}_3 \\ \dot{p}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & -\frac{1}{2} - \frac{4}{15} + \frac{2x_1^3}{375} \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} \tag{5.20}$$

e. **Control**

i. **Restraint Set:** $|u_1| \le 1$, $|u_2| \le 1$

ii. **Maximal Control:**

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \operatorname{sgn} p_4 \\ \operatorname{sgn} p_3 \end{bmatrix} \tag{5.21}$$

5.2 An Introduction to Computational Examples

The computations summarized in the following sections were performed in the Hybrid Simulation and Control Laboratory at Michigan State University. In this lab an IBM-1800 digital computer and an AD-4 analog computer are linked together, thus allowing digital control of analog operation and also hybrid computation.

In all computational examples in this thesis, the digital computer was used to provide overall control (direct the optimization routines, implement alternative comparisons, etc.) and to input/output data. Depending on the example, either the analog or the digital computer was used to integrate the state and costate equations at each iteration. While the capability exists, the digital was not used in on-line integration linkage (i.e., true hybrid operation) with the analog computer.

The digital integrations were performed using a 4th order Runge-Kutta method. While this method is relatively slow on the IBM-1800, it is reasonably accurate. It is presumed that the most significant source of inaccuracies is introduced through the control, which in all example systems is only piecewise continuous (signum function control--signum switching). To partially overcome this difficulty, whenever a discontinuity in any component of the control occurs, the integration step size is reduced by a factor of ten for the interval containing the control discontinuity.

The analog computer, on the other hand, is generally faster for equivalent accuracy. Its accuracy, however, cannot readily be improved by reducing step size, as is the case for the digital computer. It also has the tendency to be less consistent. This is especially apparent in the forward-reverse time integrations as is indicated in Section 3.6.2.

Example computer programs and subroutines are given in Appendix B. It should be noted that these programs are more complex than the basic algorithms because capabilities (data and sense switch options) for quick alternative comparisons are included.

Most of the alternative method comparisons were made using ES-1 and ES-2 for various reachable sets. This is possible since the nature of the reachable set significantly changes, for a given system, with changes in initial state, initial time and final time. For ES-1 the analog-digital computer combination was utilized. Otherwise, total digital methods were employed.

The comparisons are given in two sections. The first contains comparisons for the various algorithm alternatives. The second presents example optimization problems and includes example trajectories, reachable sets and comparisons with a totally direct search method. As previously mentioned, conclusions are included within these sections and are clearly identified.

5.3 Computational Comparisons of Algorithm Alternatives

In Chapter 3, algorithms have been presented which include alternative choices and subalgorithms. Although theoretical considerations indicate, in most instances, which alternatives are best, it is the purpose of this section to substantiate (or refute) these choices on the basis of actual computational results.

For each comparison, only one change or alternative is evaluated. Taking a specified group of reachable sets, LOP-CM is started at an arbitrary (but fixed) p_0 and the number of iterations required for convergence is measured for the first of the two alternatives. Then the second alternative is incorporated into LOP-CM and again the specified reachable sets and p_0's are used. A comparison of the average number of iterations required for the two alternatives or of the average percent decrease in the error function per iteration for each method gives an estimate of their relative value.

All results are identified by the alternatives to be compared and the basis on which the comparisons are made. These results are presented in semi-outline form and are arranged in an order corresponding to the discussion of the alternatives in Chapter 3. It is only possible to present a summary of the computational work and example results. In most cases, data, other than that listed, also confirm the results presented.

5.3.1 Study of Perturbation Relationships--Final Time

In this section, comparisons are made of some of the possibilities introduced in Section 3.6.1 (See Figure 3.2). Specifically, the relative effectiveness of a joint perturbation (both x_T and p_T) versus a single perturbation (x_T and p_T) is considered.

a. Joint Perturbation (x_T and p_T via Equation 3.78 VS x_T perturbation only.

i. Comparison Basis: ES-1, Analog Integration, $t_o = 0$, T = 20 secs., Input Data Set 1 (See Appendix C).

ii. Illustrative Example: In Figure 5.2 the reachable set for $x_o = (5,-5)^T$ is given. On this set the sequence of final states generated for each approach of this comparison are given (joint-- □, x_T only-- o).

iii. Summary of Comparison Results: The average number of iterations required to attain a local optimum were compared for the joint perturbation and for the perturbation of x_T only. The joint perturbation approach required an average of 4.7 iterations whereas perturbing x_T only resulted in an average of 6.0 iterations.

iv. CONCLUSION: The joint perturbation approach is more effective than an x_T only perturbation approach.

b. Joint Perturbation VS p_T perturbation only.

i. Comparison Basis: ES-1, Analog Integration, $t_o = 0$, T = 20 secs., Input Data Set 1; ES-2, Digital Integration, $t_o = 0$, T = .5 sec., Input Data Set 3.

ii. Summary of Comparison Results: The average number of iterations required to attain a local optimum for the joint perturbation was 8.1, for perturbation of p_T only, 8.5. Another means of comparison is the average percent improvement for each curvature move (perturbation of x_T and p_T). On this basis, a joint perturbation yielded an average 56% improvement while the p_T only approach gave an average 51% improvement.

iii. CONCLUSION: The joint perturbation approach is slightly better than a p_T only approach.

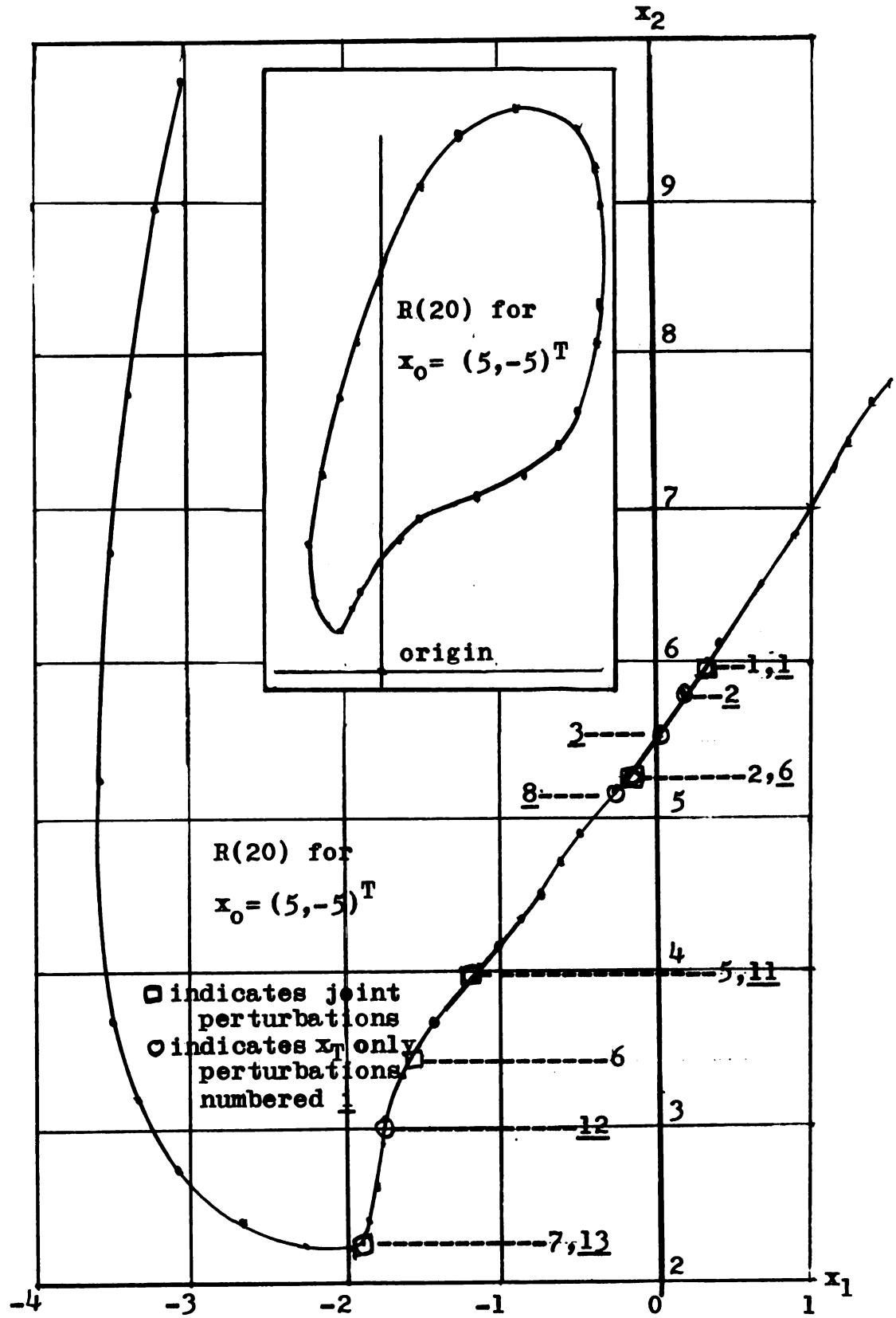


FIGURE 5.2 Joint VS Single (x_T only) R(T) Perturbations

c. GENERAL CONCLUSIONS: On the basis of the two comparisons made above, it can be concluded that the perturbation of the final adjoint has the most significant effect in improving the error function. The improvement due to perturbations in the final state are probably diminished due to the fact that slightly inaccurate perturbations in x_T result in perturbed points lying off the boundary of the reachable set. In either case the joint perturbation approach is the most effective.

5.3.2 Comparison of Curvature Values

In Chapter 3 several alternative means of estimating curvature are given. It is the purpose of this section to compare these means and the resulting curvature values. In as much as the estimates of the curvature are obtained by perturbing the final state-final costate pair, one can expect that small errors in the determination of the final state and final costate can introduce significant errors in curvature determination. Because of this, the analog and digital means of integration are analyzed separately since significant inconsistencies do result from analog computation.

a. Comparison of Curvature Values as Determined on the Analog Computer (using Equations 3.120, 3.121 and 3.123).

i. Comparison Basis: ES-1, $t_0 = 0$, $T = 20$ secs., Input Data Set 1 (See Appendix C).

ii. Illustrative Example: In Figure 5.3 the reachable set corresponding to $T = 20$ and $x_0 = (-5,0)^T$ is given. Curvature values for several extremal endpoints in an iteration sequence are given.

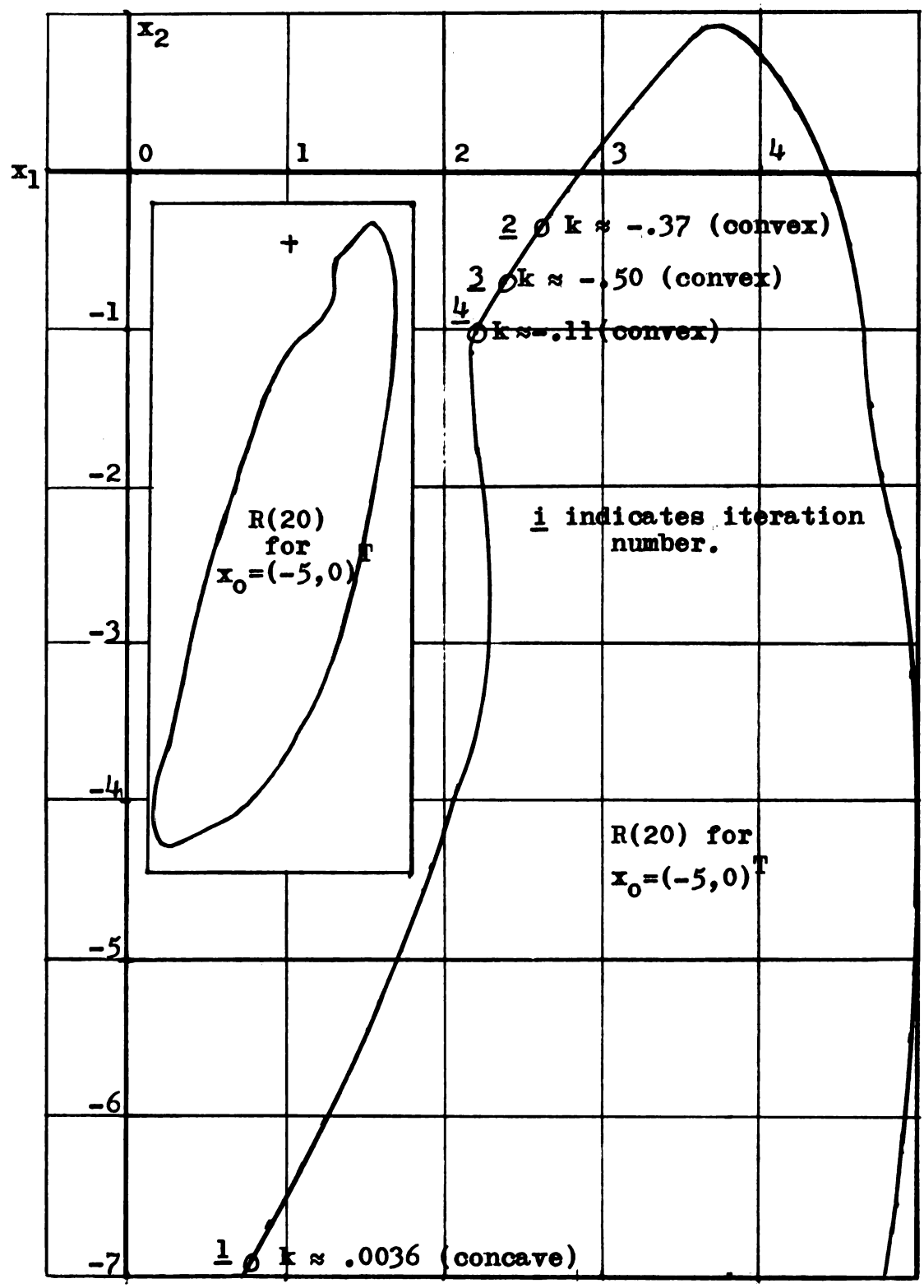


FIGURE 5.3 Example Curvature Values in an Iteration Sequence

iii. Summary of Comparison Results: In as much as ES-1 is a second order system, the boundary of the reachable set is a line of curvature. If the perturbations of x_T and p_T are sufficiently small and if no computational inaccuracies are encountered, all estimates of curvature as defined in Section 3.6.1 should coincide. This is not, however, the case for higher order systems as the perturbed final state may not be on a line of curvature through the original final state.

Comparison results are given in Table 5.1. There is reasonable correspondence between $\bar{k}$ and $\bar{k}_{av}$ except for input data numbers 2, 6, 10 and 11. Since ES-1 is a second order system, these inaccuracies must result from too large of perturbations or from integration inaccuracies in the analog computer. It is interesting to note that in these four cases, $\bar{k}_1$ and $\bar{k}_2$ have opposite signs, indicating that the estimate of k is not accurate. Several methods of correlating the various curvature estimates are available. One is the standard deviation of the $\bar{k}_i$'s (denoted $\sigma(\bar{k})$), another the standard deviation of $\bar{k}$ and $\bar{k}_{av}$ (denoted $\sigma(\bar{k},\bar{k}_{av})$ and a third, the ratio of $\sigma(\bar{k})$ and the average of $|\bar{k}_i|$ (denoted σ_n). The importance of these various comparisons are discussed further in comparison 5.3.2.d of this section.

iv. CONCLUSIONS: Experimental curvature estimates agree with those expected from reachable set geometry. Perhaps the best measure of the accuracy of the curvature determination (also the measure of whether or not the perturbation is on a line of curvature, for greater than second order systems) is $\sigma_n = \sigma(\bar{k})/|\bar{k}_i|_{av}$. If this value is too large, then the perturbation (for determining curvature) is too large, computational inaccuracies have resulted and/or the perturbation is not along a line of curvature.

b. Comparison of Curvature Values as Determined on the Digital Computer (using Equations 3.120, 3.121 and 3.123) for a Second Order System.

i. Comparison Basis: ES-1, $t_o = 0$, $T = 20$ sec., Input Data Set 1 (See Appendix C).

ii. Illustrative Example: In Figure 5.4, the reachable set corresponding to $x_o = (-10,-5)^T$ is given. Curvature values for several extremal endpoints are indicated on the boundary.

iii. Summary of Comparison Results: In Table 5.2 estimates of curvature are given. It is readily apparent that there is good agreement between these estimates.

TABLE 5.1 Comparison of Various Analog Estimates of Curvature

INPUT DATA #*	$\bar{k}_1$	$\bar{k}_2$	$\bar{k}_{av}$	$\bar{k}$	$\sigma(\bar{k})$	$\sigma(\bar{k},\bar{k}_{av})$	σ_n
1	.3630	.1217	..2423	.2175	.1206	.0124	.4977
2	.0809	-.1111	-.0157	1.2311	.0966	.6234	1.0000
4	-.0185	-.0159	-.0172	.0123	.0013	.0148	.0756
5	.1437	.1342	.1389	.1390	.0045	.0001	.0338
6	-.2101	.0870	-.0616	2.2419	.1486	1.1518	1.0000
7	.0799	.0975	.0887	.0821	.0088	.0033	.0992
8	-.0256	-.0552	-.0404	-.1670	.0148	.0633	.3663
9	.0563	.0520	.0542	.0455	.0022	.0044	.0406
10	.5915	-.6657	-.0371	1.0449	.6286	.5410	1.0000
11	-.0048	.0141	.0046	.1248	.0095	.0647	1.0000
12	.2186	.1984	.2085	.2096	.0101	.0005	.0489
13	.0650	.0538	.0594	.0921	.0056	.0163	.0943
14	-.0168	-.0159	-.0164	.0358	.0005	.0261	.0305
15	.0164	.0124	.0144	.0490	.0020	.0158	.1389

*determined for Data Set 1 (See Appendix C). No data was obtained for #3 since the initial guess of p_0 was approximately the optimum; hence no curvature estimates were calculated.

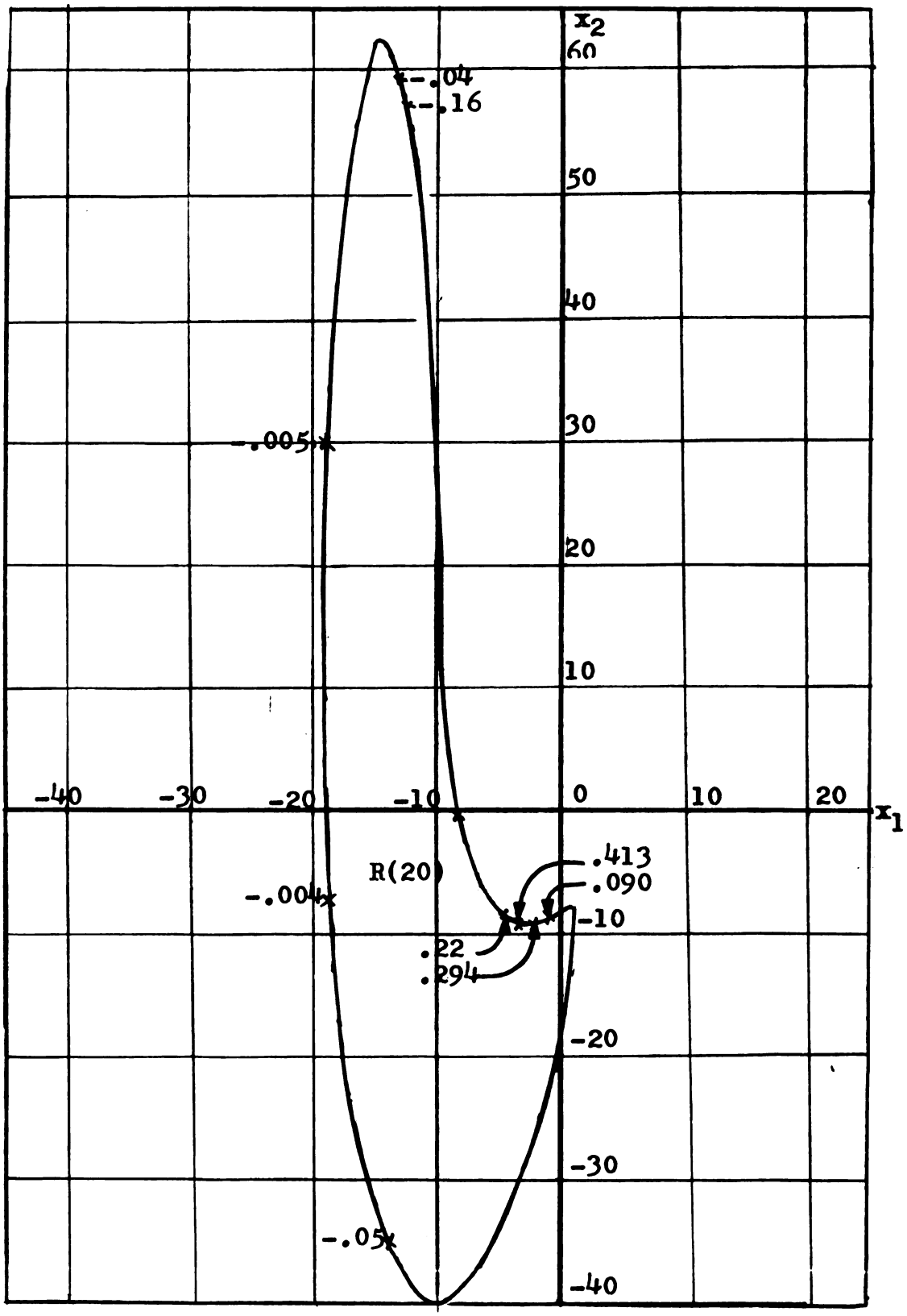


FIGURE 5.4. Example Curvature Values for R(20), $x_0 = (-10,-5)^T$

TABLE 5.2 Digital Estimates of Curvature for a 2nd-Order System

INPUT DATA #*	$\bar{k}_1$	$\bar{k}_2$	$\bar{k}_{av}$	$\bar{k}$	$\sigma(\bar{k})$	$\sigma(\bar{k},\bar{k}_{av})$	σ_n
1	.1544	.1534	.1539	.1541	.0005	.0001	.0033
2	-.7205	-.7126	-.7166	-.7117	.0039	.0025	.0055
4	.0002	.0002	.0002	.0041	.0000	.0020	.0020
5	.1559	.1562	.1561	.1557	.0002	.0003	.0010
7	.0195	.0195	.0195	.0214	.0000	.0010	.0009
8	.0685	.0687	.0686	.0670	.0001	.0008	.0016
9	.0868	.0865	.0866	.0846	.0001	.0010	.0018
10	-.0292	-.0293	-.0292	-.0301	.0001	.0005	.0020
11	-.2440	-.2414	-.2427	-.2462	.0013	.0018	.0055
12	.0697	.0695	.0696	.0690	.0001	.0003	.0013
14	.0043	.0043	.0043	.0086	.0000	.0022	.0009
15	-.0019	-.0019	-.0019	-.0023	.0000	.0002	.0051

*determined for Data Set 1 (See Appendix C). Data numbers 3, 6 and 13 are left off since the original guess was so near the optimum that LOP-DS was employed rather than LOP-CM; hence no estimates of curvature were calculated.

iv. CONCLUSIONS: For the second order system being considered, perturbations of the digitally integrated differential equations gave very good estimates of the curvature. As expected, all variances were low; thus verifying that the perturbation size was sufficiently small to give a good estimate of curvature.

c. Comparison of Curvature Values as Determined on the Digital Computer (using Equations 3.120, 3.121, and 3.123) for a Third Order System.

i. Comparison Basis: ES-2, $t_0 = 0$, T = .5 sec., Input Data Sets 3 and 4.

ii. Summary of Comparison Results: In Table 5.3 various estimates of curvature are given. Unlike the previous comparison, made for a second order system, most data points have rather large variance of curvature values. This is, of course, expected where the boundary of the reachable set is not implicitly a line of curvature. Error function values (E_2) are given at the final state before perturbation and as the result of the next iteration (which is based on the estimate of curvature value).

iii. CONCLUSIONS: Since the same methods were utilized in this example and in comparison 5.3.2.b, it is reasonable to expect that the large variances as shown in Table 5.3 result from perturbations which are not generally on lines of curvature. There is no reason to believe that these large variances result from perturbations which are too large or from computational inaccuracies.

d. GENERAL CONCLUSIONS: Examination of the three comparisons made in this section indicate several important facts pertaining to curvature determination as it relates to computational efficiency. First of all, comparisons 5.3.2.a and 5.3.2.b indicate the relative consistency and accuracy of the digital integration approach as compared to that of analog integration. Since a second order system represents a very special case, the important conclusions relative to curvature determination rests on third and higher order comparisons.

TABLE 5.3 Digital Estimates of Curvature for a 3rd-Order system

INPUT DATA #*	$\bar{k}_1$	$\bar{k}_2$	$\bar{k}_3$	$\bar{k}_{av}$	$\bar{k}$	σ_n	i^{th} cos γ	$i+1^{st}$ cos γ
3.1.1	-482.6	-9.232	-.059	-163.9	23.679	1.374	-.111	-.962
3.1.2	.375	-.253	-.026	.032	-.074	1.191	-.962	-.979
3.3.1	-.140	-1.288	-1.224	-.884	-1.513	.596	-.194	-.688
3.3.2	-3.292	-.370	-1.531	-1.731	-.254	.694	-.688	-.971
3.4.1	3.803	-2.000	-2.590	-.262	-2.590	1.031	-.074	-.896
3.4.2	-2.841	-5.475	-4.328	-4.215	-12.80	.256	-.896	-.999
3.5.1	1.455	-.461	-.702	.097	-.899	1.106	-.561	-.797
3.5.2	1.195	-.793	-1.486	-.361	-1.512	.981	-.797	-.918
3.6.3	-.259	-.462	-3.413	-1.378	-.490	1.046	-.918	-.975
3.7.1	-4.633	-.597	-31.23	-12.15	-4.867	1.118	-.410	-.515
3.7.3	-7.748	.352	-.996	-2.797	8.375	1.169	-.691	-.482
3.9.1	-4.716	-.709	.457	-1.656	-.905	1.130	-.299	-.304
3.9.2	-7.504	-1.057	.092	-2.823	-1.490	1.159	-.304	-.350
3.9.3	-7.001	-1.385	-.367	-2.920	-1.720	1.000	-.350	-.445
3.9.4	-5.656	-1.139	-.340	-2.378	-1.389	.984	-.445	-.629
3.9.5	-4.303	-.912	-.555	-1.923	-1.041	.878	-.629	-.887
3.10.1	22.413	-1.718	.837	7.177	.401	1.300	-.115	.186
4.1.1	-.389	.153	-.147	-.127	-.169	.966	-.769	-.901
4.1.2	-.555	-1.371	-.506	-.812	-.939	.491	-.901	-.976
4.2.1	-4.766	-.463	-.095	-1.771	-.860	1.195	-.222	-.176
4.4.1	10.097	-3.804	-.173	2.040	.810	1.255	-.666	-.043
4.5.1	-4.491	10.213	-.060	1.887	-.142	1.251	-.899	-.771
4.6.1	-3.153	-1.403	-.114	-1.557	-2.502	.800	-.061	-.973

*determined for Data Sets 3 and 4 (See Appendix C). The input data # indicates the data set, data point and the iteration number for that point. For example, 3.9.3 indicates Data Set 3, p_o^0 number 9 and LOP-CM iteration number 3.

Examination of comparison 5.3.2.c (third order) yields several important conclusions. Obviously, if it were necessary to accurately identify a line of curvature, it would be necessary to attempt several, perhaps even many, perturbations. In fact, of all the perturbations represented in Table 5.3, only several give definite indication of being near a line of curvature (data number 3.4.2, for example).

Perhaps the single most important number given for each data point is σ_n, the ratio of the standard deviation of the curvature estimates to the average of the absolute values of these estimates. To demonstrate the significance of this value and to indicate its use in LOP-CM, consider Table 5.4. In this table, the data is arranged according to increasing magnitude of σ_n. Except for one data point (3.1.1), there is a close correspondence between the value for σ_n and the percent decrease of the error function, denoted δ_γ. The data point 3.1.1 which does not follow this general observation is assumed to be an anomaly--resulting from the extreme values taken by the curvature estimates.

The value for σ_n which seems to represent the cut-off point (i.e., the point at which no improvement in cos γ is noted) is approximately at $\sigma_n = 1.2$. Stated differently, for any curvature estimate with a value of σ_n below 1.159, the error function is improved in the iteration based upon that estimate.

It would certainly be desirable to obtain a very good

TABLE 5.4 Evaluation of the Significance of σ_n

INPUT DATA #*	σ_n^*	i^{th} cos γ	i+1st cos γ	$-\delta\cos\gamma$	δ_γ^*	ave δ_γ
3.4.2	.256	-.896	-.999	.103	.990	
4.1.2	.491	-.901	-.976	.075	.758	.817
3.3.1	.596	-.194	-.688	.494	.613	
3.3.2	.694	-.688	-.971	.283	.908	
4.6.1	.800	-.061	-.973	.912	.971	
3.9.5	.878	-.629	-.887	.158	.427	.641
4.1.1	.966	-.769	-.901	.132	.572	
3.5.2	.981	-.797	-.918	.121	.596	
3.9.4	.984	-.445	-.629	.184	.332	
3.9.3	1.000	-.350	-.445	.095	.146	.516
3.4.1	1.031	-.074	-.896	.826	.892	
3.6.3	1.046	-.918	-.975	.057	.695	
3.5.1	1.106	-.561	-.797	.236	.537	
3.7.1	1.118	-.410	-.515	.105	.178	.197
3.9.1	1.130	-.299	-.304	.005	.007	
3.9.2	1.159	-.304	-.350	.046	.066	
3.7.3	1.169	-.691	-.482	-.209	-.678	
3.1.2	1.191	-.962	-.979	.017	.448	-.272
4.2.1	1.195	-.222	-.176	-.046	-.059	
4.5.1	1.251	-.899	-.771	-.128	-1.268	
4.4.1	1.255	-.666	-.043	-.623	-1.868	-1.104
4.10.1	1.300	-.115	-.186	-.301	-.340	
3.1.1	1.374	-.111	-.962	.857	.951	

* determined for Data Sets 3 and 4 (See Appendix C). The data number corresponds to those in Table 5.3.

* $\sigma_n = \dfrac{\sigma(\bar{k})}{|k_i|_{av}}$

* $\delta_\gamma = \dfrac{-\delta\cos\gamma}{1+\cos\gamma}$

estimate of curvature, but there is a tradeoff between the computational time required for obtaining that estimate and the time required for iterating to a final state nearer the optimum. For this reason, the factor σ_n is particularly useful in indicating when the curvature estimate is close enough to use in LOP-CM.

Another observation is apparent from Tables 5.3 and 5.4. As the optimum is approached, the curvature estimate generally improves. This is especially apparent in considering σ_n for data points 3.1, 3.4, 3.5 and 3.9.

In summary, the experimentation of this section indicates:

1) For second order systems (where the boundary is the line of curvature), the curvature estimates correspond closely to that expected from reachable set geometry.

2) Estimates of curvature obtained by digital integration are more consistent and more accurate than those determined by the analog computer.

3) The factor σ_n is a good measure of both the location of a final state perturbation relative to a line of curvature and the usefulness of the curvature estimate.

4) As the optimum is approached, the curvature estimate generally improves.

5.3.3 Effect of the Basic Curvature Formula Choice

In this section the effect of the basic curvature formula choice (Equations 3.120, 3.121, 3.123 and 3.124) on the rate of optimization convergence is considered. Comparisons were made for LOP-CM using both Subalgorithms 4 (i.e. with and without standard deviation evaluation of

the curvature).

a. Comparison Basis: ES-1, $t_o = 0$, T = 20 secs., Input Data Sets 1 and 2 (See Appendix C); ES-2, $t_o = 0$ T = .5 sec., Input Data Sets 3 and 4 (See Appendix C).

b. Summary of Comparison Results: The average number of iterations required to attain a local optimum were compared for each of the basic curvature formulas. The following results were obtained:

$\bar{k}$ (Equation 3.123)------7.7 iterations
$\bar{k}_{av2}$(Equation 3.124)----7.7 iterations
$\bar{k}_i$ (Equation 3.120)-----8.3 iterations
$\bar{k}_{av}$ (Equation 3.121)----9.3 iterations.

In as much as $\bar{k}_{av2}$ and $\bar{k}$ had the same average number of iterations, a further comparison of these two estimates was made. This comparison was based on ES-2, a third order system. In this case the percent improvement in the error function for each curvature move was considered. For $\bar{k}$ the percent improvement was 27.9 while it was 43.3 for $\bar{k}_{av2}$.

c. CONCLUSIONS: If the estimates of the curvature were accurate, the choice of basic curvature formula would have little effect on the algorithm efficiency. In the case of third and higher order systems, where only approximations to lines of curvature are achieved, the choice of basic curvature formula does alter the algorithm convergence. The data of this section indicates that $\bar{k}_{av2}$, the overall average of curvature estimates, is the best choice with $\bar{k}$ being the second choice.

5.3.4. Comparison of LOP-CM Subalgorithms 4a and 4b

Within the structure of LOP-CM, two potential subalgorithms are introduced to determine curvature. Subalgorithm 4.a provides an estimate of k without considering the accuracy of the estimate. On the other hand, Subalgorithm 4.b considers the normalized standard deviation σ_n and rejects any estimate for which σ_n is greater than a specified value. In the computations of this

section, the limit is specified as 1.2 corresponding to the results indicated in Section 5.3.2.

a. Comparison Basis: ES-2, $t_o = 0$, T = .5 sec., Input Data Set 3.

b. Summary of Comparison Results: In as much as there is tradeoff between computation time spent estimating curvature and time spent iterating, only a maximum of n (3 for this third order system) attempts at obtaining a suitable estimate of curvature are allowed. For this case, the average number of iterations to achieve a local optimum without considering σ_n was 11.0 whereas utilizing σ_n to evaluate the curvature estimate improved this average number to 9.3. Consideration of the average percent improvement in the error function per curvature move (LOP-CM iteration), substantiates these figures. If σ_n is not considered, this percent is 34.7 whereas utilization of the σ_n factor gives an improvement of 43.3%.

c. CONCLUSION: The data of this section indicates that rejection of poor curvature estimates (high σ_n) improves the convergence of LOP-CM. This effect would be even more marked were the allowable σ_n smaller and the allowed number of attempts at obtaining a good curvature estimate larger..

5.3.5 Comparison of Perturbation Direction Alternatives

In step 5 of LOP-CM, three alternative equations (3.80, 3.90 and 3.108) are listed for determining the direction of the perturbation of the state at the final time. It is the purpose of this section to present experimental results comparing these three choices.

a. Comparison Basis: ES-1, $t_o = 0$, T = 20 sec., Input Data Sets 1 and 2; ES-2, $t_o = 0$, T = .5 sec., Input Data Set 3.

b. Summary of Comparison Results: The average number of iterations required to attain a local optimum for these three alternatives were as follows:

i. Equation 3.80--------7.20 iterations

ii. Equation 3.108-------7.31 iterations

iii. Equation 3.90--------7.81 iterations.

Similar results were obtained for the average percent decrease in the error function for each LOP-CM iteration: i. 42.4%, ii. 40.7% and iii. 20.9%

c. CONCLUSIONS: Very little difference was noted between the effect on LOP-CM of using Equations 3.80 and 3.108. On the other hand, Equation 3.90 gives less satisfactory results. It should be recalled that Equation 3.80 was developed for regions far from an optimum, Equation 3.90 for regions near an optimum and Equation 3.108 for general applicability. In the computations of this section, LOP-CM was used until E_r reached .1 and then LOP-DS was employed. Had LOP-CM been used for greater convergence, the utility of Equation 3.108 would have been more readily apparent.

5.3.6 Comparison of Perturbation Step Size Alternatives

In the previous subsection, the direction of the final state perturbation was considered. In this section, its relative magnitude, c, is discussed. The important comparison to be made is that of a constant step size as compared to a step size dependent upon the error of the unperturbed boundary point. Specifically, the two candidates selected are $c = .2$ and $c = (1 + \cos \gamma)$.

a. Comparison Basis: ES-1, $t_o = 0$, T = 20 secs, Input Data Sets 1 and 2 (See Appendix C); ES-2, $t_o = 0$, T = 20 sec., Input Data Set 3.

b. Illustrative Example: In Figure 5.5 an example iteration sequence is given for both the constant and variable step factors.

c. Summary of Comparison Results: The average number of iterations required to attain a local optimum for constant step factor (c = .2) was 6.68, while the error dependent factor yielded a lower average number of iterations--5.92. The average percent decrease in the error for each LOP-CM iteration gave corresponding results: constant step factor--43.0%, error dependent step factor--49.1%.

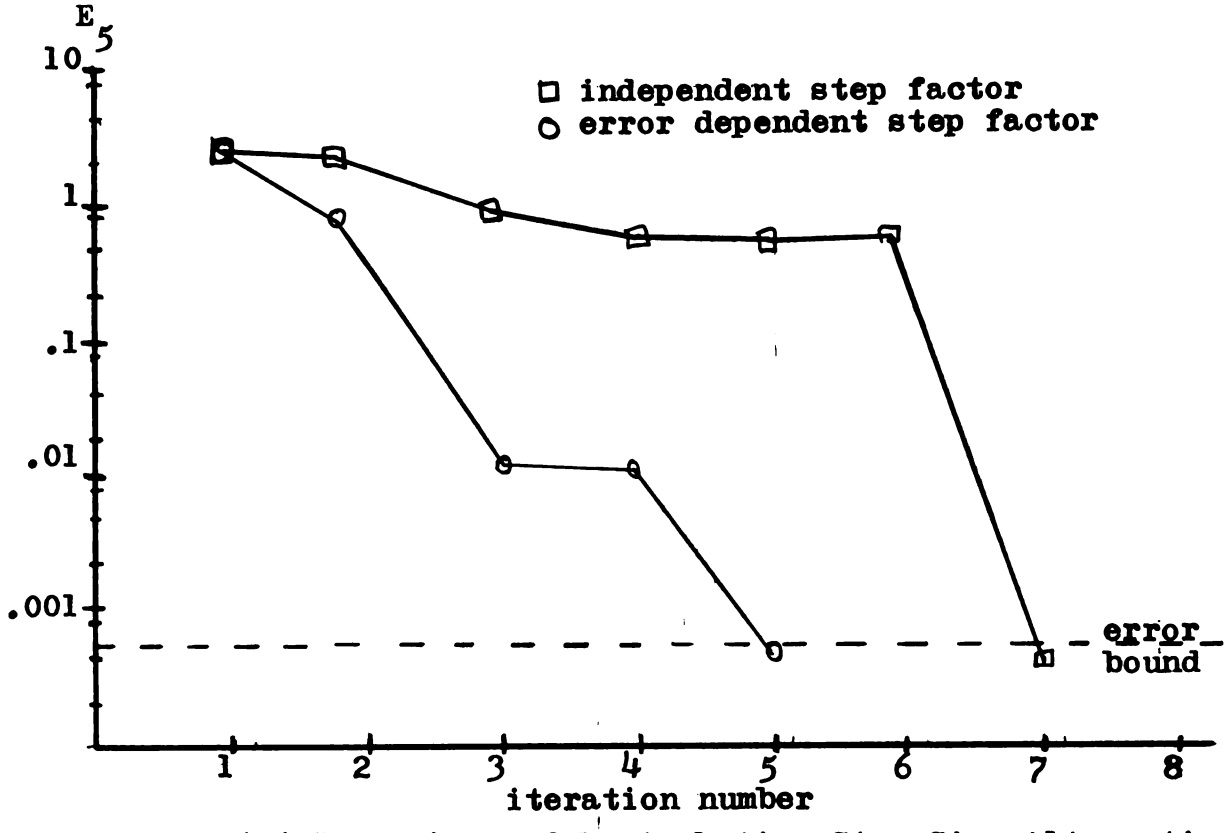


FIGURE 5.5 Comparison of Perturbation Step Size Alternatives

d. CONCLUSION: The error dependent step factor ($c = 1 + \cos \gamma$) is more satisfactory in achieving convergence than the constant step factor.

5.3.7 Analog Error and the Integration Correction Routine

Section 3.6.2 pointed out that significant errors are encountered in analog integration of the differential equations of Example Problem 1. As a consequence, correction factors α and β are introduced in Step 8 of LOP-CM. It is the purpose of this section to give examples of these errors, consider their origin and evaluate the effectiveness of the correction factors.

a. Comparison Basis: ES-1, $t_o = 0$, T = 20 secs., Input Data Set 1 (See Appendix C), Both Analog and Digital Integration.

b. Illustrative Examples: Figure 5.6 is given to relate the various state and adjoint vectors and correction factors. In Table 5.5, examples of these various vectors are given for both analog and digital integration. Values of the various vectors depicted in Figure 5.6 are presented as are error function values at the start and at the conclusion of the iteration.

c. Summary of Results: The computations of this section yield several important results. First of all, consideration of the values for α^i and β^i in Table 5.5 demonstrates that analog computation errors are indeed important. Furthermore, at points on R(T) far from an optimum, these errors appear to be larger than for points near an optimum. Two digital iterations are also presented and it is readily apparent that the digital integration routine does not produce these large errors.

Comparison of LOP-CM with and without the use of the error correction factors α^i and β^i demonstrates their usefulness. Without the error corrections, the average number of iterations required to determine a local optimum is 9.75 whereas the introduction of the correction factors lowered this to 7.00. Comparison of the relative efficiency of LOP-CM with analog integration and LOP-CM with digital integration is given in the next section.

d. CONCLUSIONS: It is, of course, obvious that analog computation errors do occur and that the correction factors α^i and β^i do help compensate for these difficulties. The significant item to determine is the cause of these errors. At first the author considered the possibility of incorrect patching on the analog computer or incorrect conversion to reversed time integration. After eliminating these as possibilities, analog-digital conversion and digital-analog conversion were considered. These too, do not appear to be significant since the error is only on the order of one-half of one percent.

Further consideration indicates that the errors encountered are a result of the interaction between several factors:

1) The nature of the differential equations and the large time interval involved in these comparisons (20 seconds).

2) The sensitivity of the trajectories to switching time.

3) The errors created by analog equipment--comparitors, integrators, multipliers, electronic switches, etc.

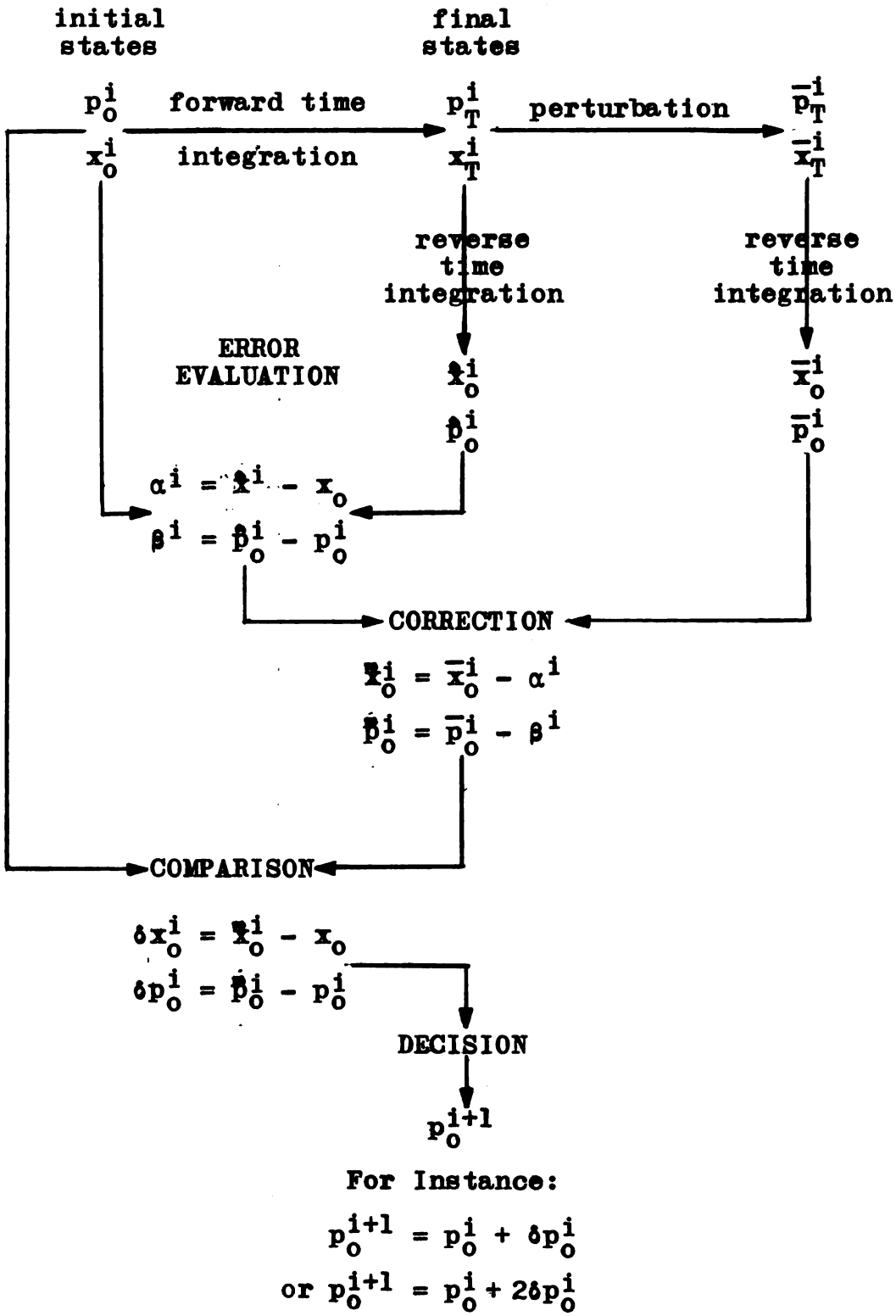


FIGURE 5.6 Analog Error Evaluation and Correction Flow Chart

TABLE 5.5 Forward-Reverse Time Integration Error Evaluation

DATA #*	x_o	p_o^i	error	α^i	β^i	new error
A1-1	-10.	0.000	1.087	.161	8.212	.600
	- 5.	10.000		4.292	-4.293	
A1-2	-10.	1.474	.600	.063	.021	4.619
	-5.	6.779		3.218	-.358	
D1-1	-10.	0.000	1.277	.001	-.029	1.219
	-5.	10.000		.002	-.000	
D1-2	-10.	2.166	1.219	.001	-.026	.689
	-5.	9.769		.002	.006	
A4-1	6.	10.000	3.962	-.360	-.139	1.052
	2.	-3.000		-.755	-.429	
A4-2	6.	-6.932	1.052	-.360	-.271	.393
	2.	-6.642		-1.487	.894	
A4-3	6.	-9.556	.393	-.360	-.115	.286
	2.	-2.743		-1.634	1.218	
A4-4	6.	-9.883	.286	-.409	-.060	.167
	2.	-1.400		-1.243	.927	
A15-1	5.	-5.000	1.842	-1.167	4.036	.430
	-5.	-5.000		.630	-2.457	
A15-2	5.	-9.389	.430	-.630	.156	.293
	-5.	-5.995		.386	-.514	
A15-3	5.	-9.862	.293	-.532	.002	.056
	-5.	-1.892		-.396	.780	
A15-4	5.	-9.946	.056	-.581	.378	.030
	-5.	0.271		-.200	1.567	

*Aj-i indicates the i^{th} iteration for the j^{th} data point using analog integration. Digital integration for the same data point is indicated by Dj-i.

The first two factors multiply the effect of the errors introduced by the analog equipment. Take, for example, the state and costate trajectories plotted in Figure 5.7. While the initial and final points are not far removed from one another, the trajectories traced out do by no means represent a shortest path from the initial to the final points. Also, four switching times occur for this extremal trajectory even though only a second order system is involved. Even slight differences in switching time due to comparitor hysteresis, slight multiplier inaccuracies, etc. can have a significant effect on trajectory behavior.

5.3.8 Comparison of Analog vs Digital Integration in LOP-CM

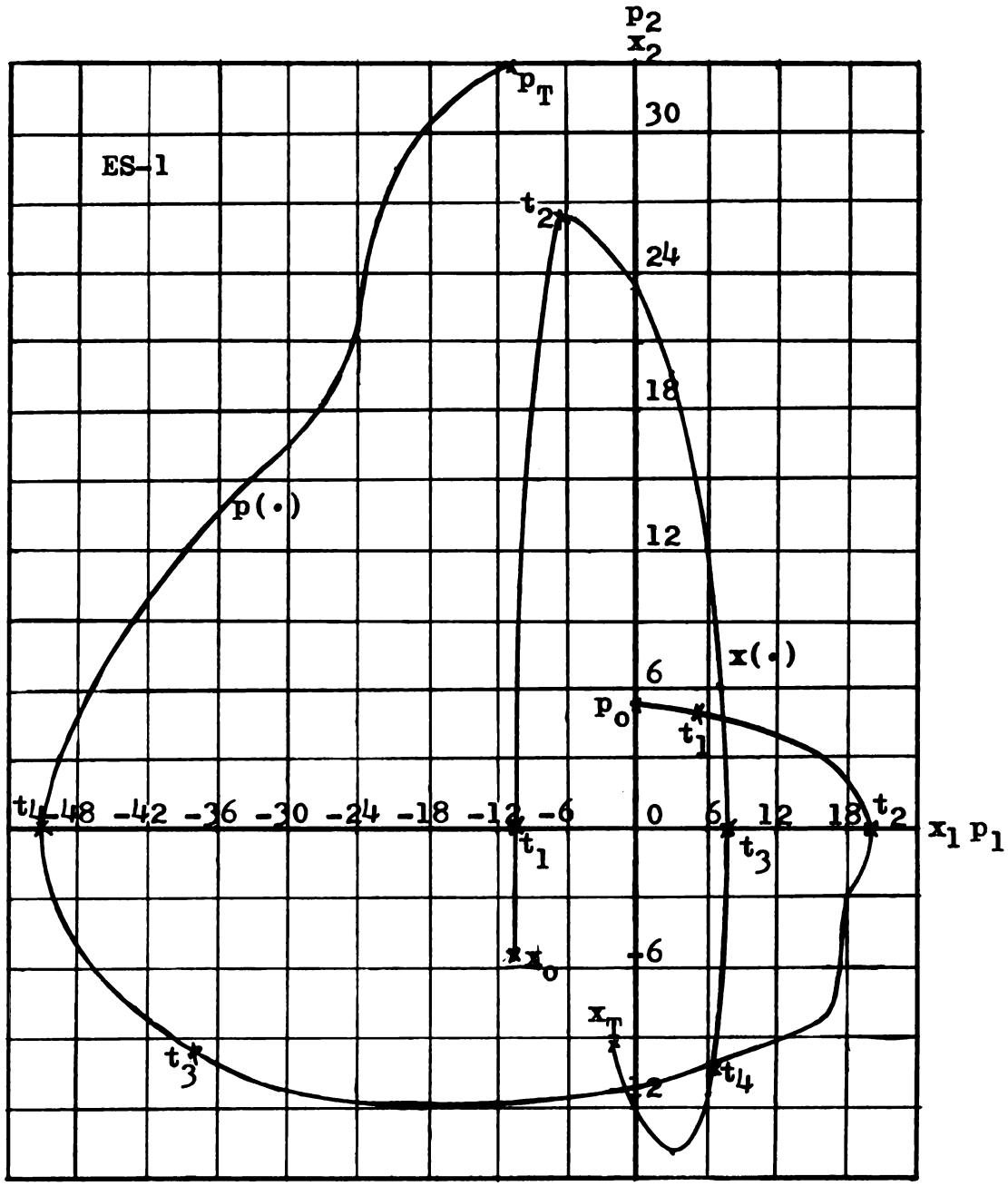
The previous subsection strongly indicates that the errors due to inaccurate analog computation have a significant effect on the algorithm convergence. The introduction of the correction factors α^i and β^i improve this convergence. It is the purpose of this section to determine if the resulting improved LOP-CM with analog integration is as effective as a completely digital LOP-CM.

a. Comparison Basis: ES-1, $t_o = 0$, T = 20 secs., Input Data Set 1.

b. Summary of Results: The average number of iterations required to attain a local optimum for LOP-CM with corrected analog integration was 6.25. With digital integration, the average number was only 5.33 for the same data. In addition, LOP-CM with analog integration failed to converge in two cases because of analog inconsistencies whereas convergence was always achieved using digital integration.

c. Illustrative Examples: Two typical examples of the convergence of LOP-CM with analog and LOP-CM with digital integration are shown in Figure 5.8.

d. CONCLUSION: Use of digital integration in LOP-CM is far more consistent and accurate. As a result, it converges in less iterations than LOP-CM with analog integration. It has the disadvantage of requiring more digital computer time. A judgment as to which approach is best would depend on the equipment available, the accuracy of



$x_0 = (-10,-5)^T$ $p_0 = (0,5)^T$ t_1-t_4 indicate switching times

$t_1 = .48$ sec. $t_2 = 3.35$ sec. $t_3 = 11.26$ sec. $t_4 = 13.08$ sec.

FIGURE 5.7 Example Trajectories for a Second-Order System

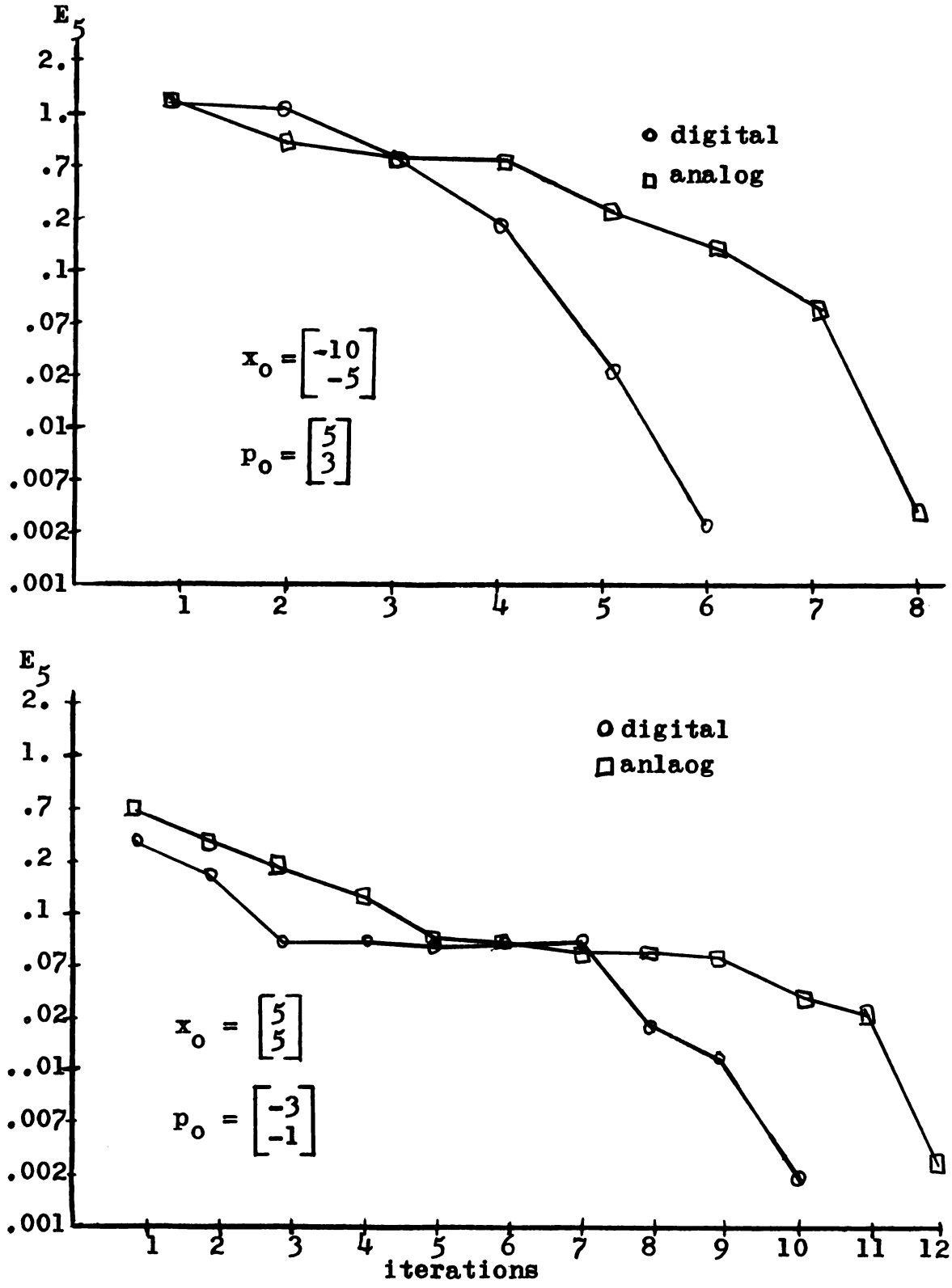


FIGURE 5.8 LOP-CM Convergence for Analog or Digital Integration

the desired results and the computation time available. It is the author's opinion that, in general, LOP-CM with digital integration would be the better choice.

5.4 Comparison of LOP-CM with LOP-DS

In the preceding section many LOP-CM alternatives were considered. Choice of the most suitable of these alternatives yields an effective optimization algorithm. It is the purpose of this section to summarize the comparisons made between the resulting LOP-CM and LOP-DS, a direct search algorithm.

The comparisons were made for Example Systems 1 and 2, with selected data points from Appendix C. The results of several sequences of comparisons showed an average of 4.17 iterations required to achieve each local optimum using LOP-CM. The direct search routine, LOP-DS, required an average of 7.12. Thus LOP-CM represents a definite improvement over a direct search routine. In Figure 5.9 an example of their relative convergence is given.

5.5 Global Optimization Examples

All of the preceding developments and comparisons were aimed at producing an efficient algorithm for solving the modified problem MP. As a result of the comparisons summarized thus far in this chapter, an efficient form of LOP-CM is identified. Using this "optimized" LOP-CM, the Global Optimum Procedure GOP given in Section 4.7 is now applied to example problems. This procedure is based on

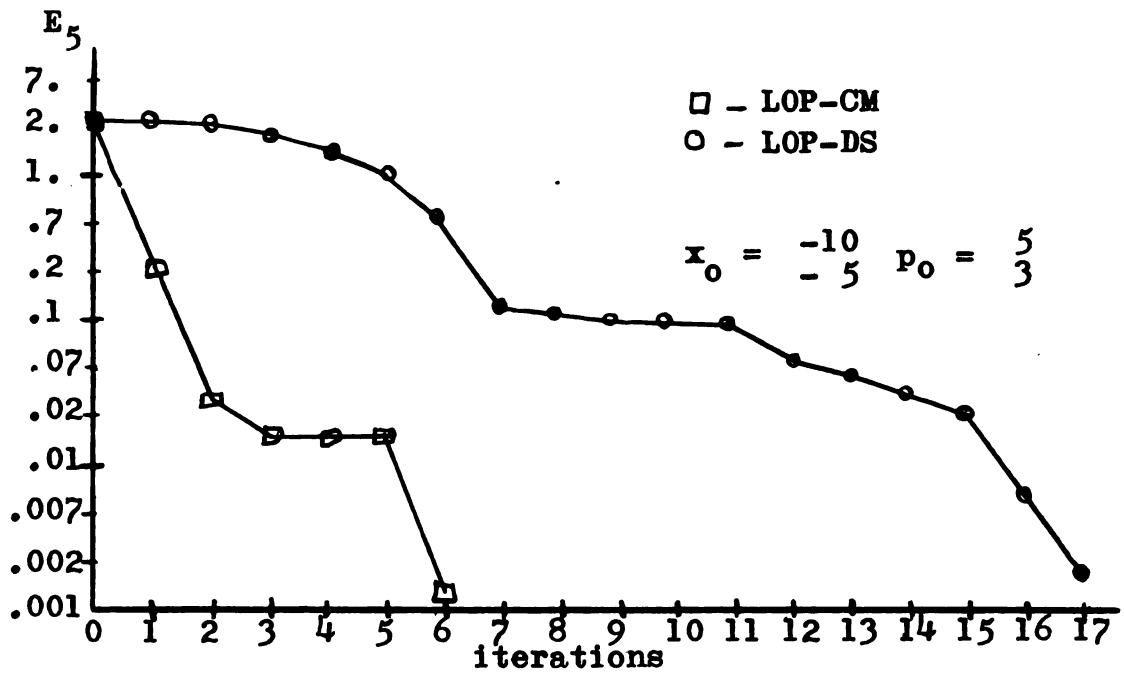


FIGURE 5.9 Comparison of LOP-CM and LOP-DS Convergence

the generation of N local minima of the error function by starting at N arbitrary initial adjoints and utilizing LOP-CM until a local minimum is obtained. Of course, the probability that a global optimum is included in this resulting group of local minima increases as the integer N is increased.

In this section, the Global Optimum Procedure is applied to the example problems of Section 5.1. The integer N is fixed at 10, thus ten arbitrary initial adjoints are specified and ten minima of the error function E_5 are located. Many or all of these minima may coincide. In fact, if only one minimum of E_5 exists, i.e. the global optimum is unique, then all ten must coincide.

Some of the special problems discussed in Chapter 4 are encountered in the computations of this section. They are discussed as they occur. In addition, typical iterations for LOP-CM are presented for some of the global problems considered.

5.5.1 Examples for ES-1: 2nd-Order System, Scalar Control

Example Problem 1 is particularly useful as an example of global optimization since it was specifically developed to produce significantly nonconvex reachable sets. The shape of these sets, of course, depends on the initial state selected and on the time interval of integration. The effect of increasing the time interval is demonstrated in the next section (time optimal control). In this section the global problem is considered for several reachable sets for ES-1 corresponding to different initial states. The time interval is fixed at T = 20 seconds.

The first initial state to be considered is $x_0 = (-10,-5)^T$. The reachable set has already been shown in Figure 5.4. The ten resulting minima of GOP are given in Table 5.6. Consideration of these data points shows that GOP selects three possibilities for the global optimum--data points 1, 3 and 8; data points 2, 4, 5, 7 and 9; and data points 6 and 10. Of these, E_1 is minimum for data number 1, 3 and 8. Hence they represent the global optimum as consideration of the reachable set demonstrates.

TABLE 5.6 Application of GOP to ES-1 with $x_0 = (-10,-5)^T$

Data #	x_{1T}	x_{2T}	$E_1=\|\|x_T\|\|$	E_2	E_5
1	.526	-8.268	8.284	-.9993	.0057
2	-4.104	-9.106	9.989	-.9999	.0013
3	.526	-8.268	8.284	-.9990	.0078
4	-4.248	-9.039	9.987	-.9995	.0052
5	-4.104	-9.108	9.989	-.9998	.0021
6	-8.151	-2.141	8.427	-.9999	.0009
7	-4.104	-9.108	9.989	-.9998	.0022
8	.586	-8.264	8.285	-.9997	.0027
9	-4.248	-9.039	9.987	-.9994	.0056
10	-8.039	-2.539	8.430	-.9997	.0023

Consideration of the other two minima of E_5 shows that one is a local optimum (Data points 6 and 10) and that the other is a false optimum as discussed in Section 4.1. Both of these minima occur on concave surfaces. Application of Theorem 4.1 to the first point (6 and 10) demonstrates that it is a true local optimum. The curvature at this point is .02; thus K_{CV} is .02 since the boundary is <u>the</u> line of curvature. But $\|x_T\|$ is 8.4 or $1/\|x_T\|$ is .119. Thus Equation 4.10 applies:

$$K_{CV} = .02 < .119 = 1/\|x_T\|. \qquad (5.22)$$

and thus x_T is a local optimum. For data points 2, 4, 5, 7 and 9, the curvature is .22; hence K_{CV} is also .22. But $\|x_T\|$ is 9.99 thus $1/\|x_T\|$ is approximately .1, thus Equation 4.6 applies:

$$K_{cv} = .22 > .1 = 1/||x_T||, \qquad (5.23)$$

or x_T in this case is not a local optimum.

The second initial state considered for ES-1 is $x_0 = (5,5)^T$. The resulting reachable set is shown in Figure 5.10. While this set is nonconvex, it does not produce as interesting results as the previous example. In this case, there is only one local optimum (located at $x_T = (-2.4,.5)^T$) and thus each iteration of GOP selected it. It, of course, represents the global optimum.

Also illustrated in Figure 5.10 is an example sequence of iterations. In this case, it took 5 iterations to converge to the optimum. The final state perturbations for the first four steps are shown.

5.5.2 Examples for Higher Order Systems

While application of GOP to ES-1 resulted in several minima of E_5 being achieved, application to the other example problems yielded unique optima. The resulting optimum final states are given in Table 5.7 as are the corresponding values of the error functions. Given in Figure 5.11 are several examples of the change of error with each iteration for ES-2 for various p_0^0.

5.6 Time Optimization Examples

In Section 4.8 the application of GOP to time optimal control problems was introduced. It is the purpose of this section to apply the resulting procedure to several example

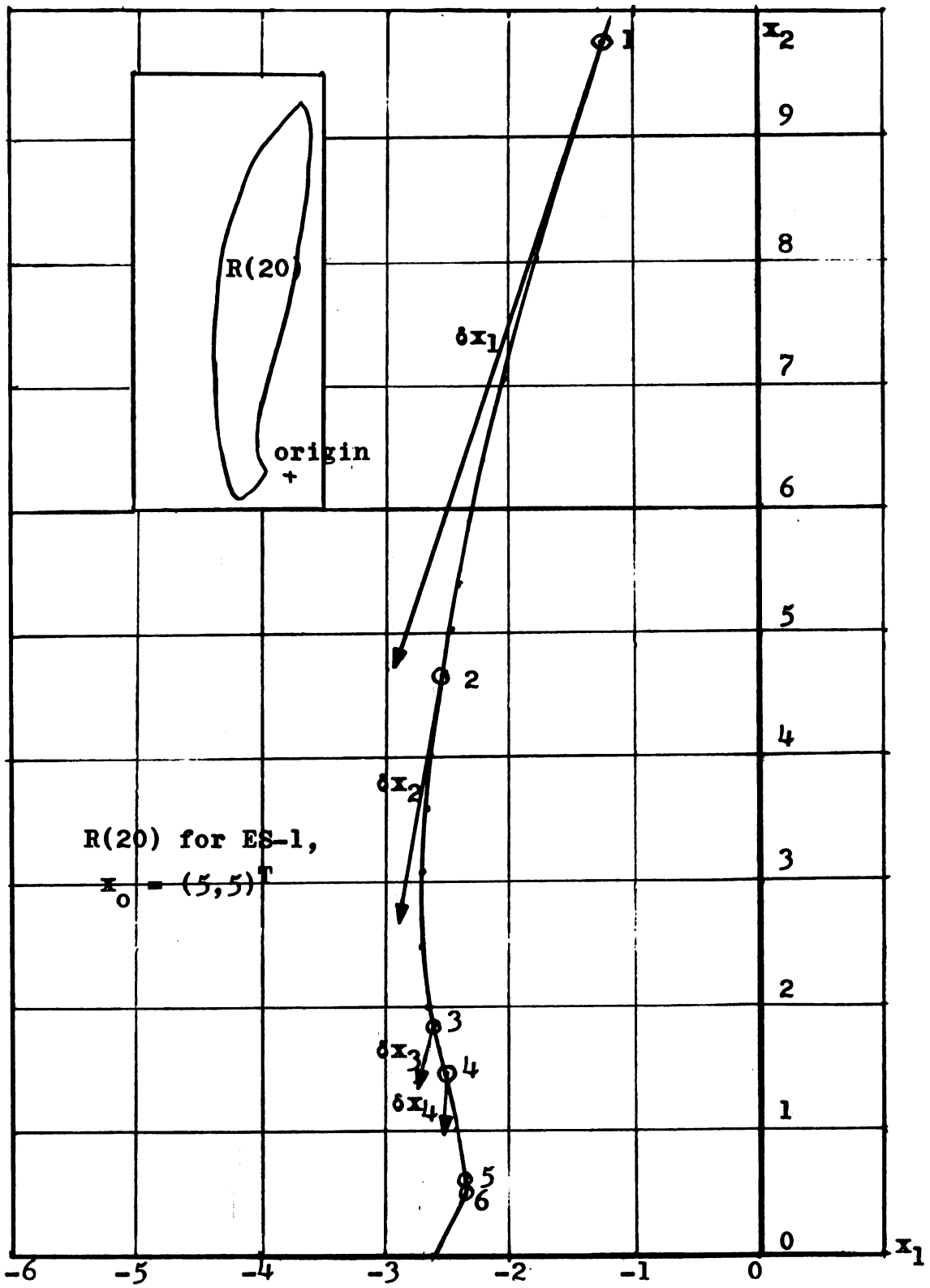


FIGURE 5.10 Reachable Set and Example Iterations for LOP-CM

TABLE 5.7 Application of GOP to Higher Order Systems

Example Problem	x_o	T	x_T^*	$\|\| x_T^* \|\|$	E_2	E_5
ES-2	-3		-4.745			
	-2	1	-1.271	4.931	-.9998	.001
	-1		.533			
	2		.695			
	-2	1	-.541	.947	-.990	.009
	0		.358			
	-3		-3.947			
	-2	.5	-1.676	4.300	-.9998	.001
	-1		.314			
	2		1.344			
	-2	.5	-.701	1.882	-.998	.004
	2		1.116			
	2		1.181			
	-2	.5	-1.227	1.831	-.998	.004
	0		.670			
	2		1.588			
	-1	.5	-.492	1.688	-.9996	.001
	1		.291			
ES-3	-1		-.037			
	2	.5	1.831	1.848	-.996	.007
	0		-.248			
	-5		-.483			
	-5	1	-1.182	13.762	-.9990	.007
	-5		-12.834			
ES-4	-2		-.197			
	2	1	1.463	1.610	-.962	.062
	0		-.632			
	0		-.111			

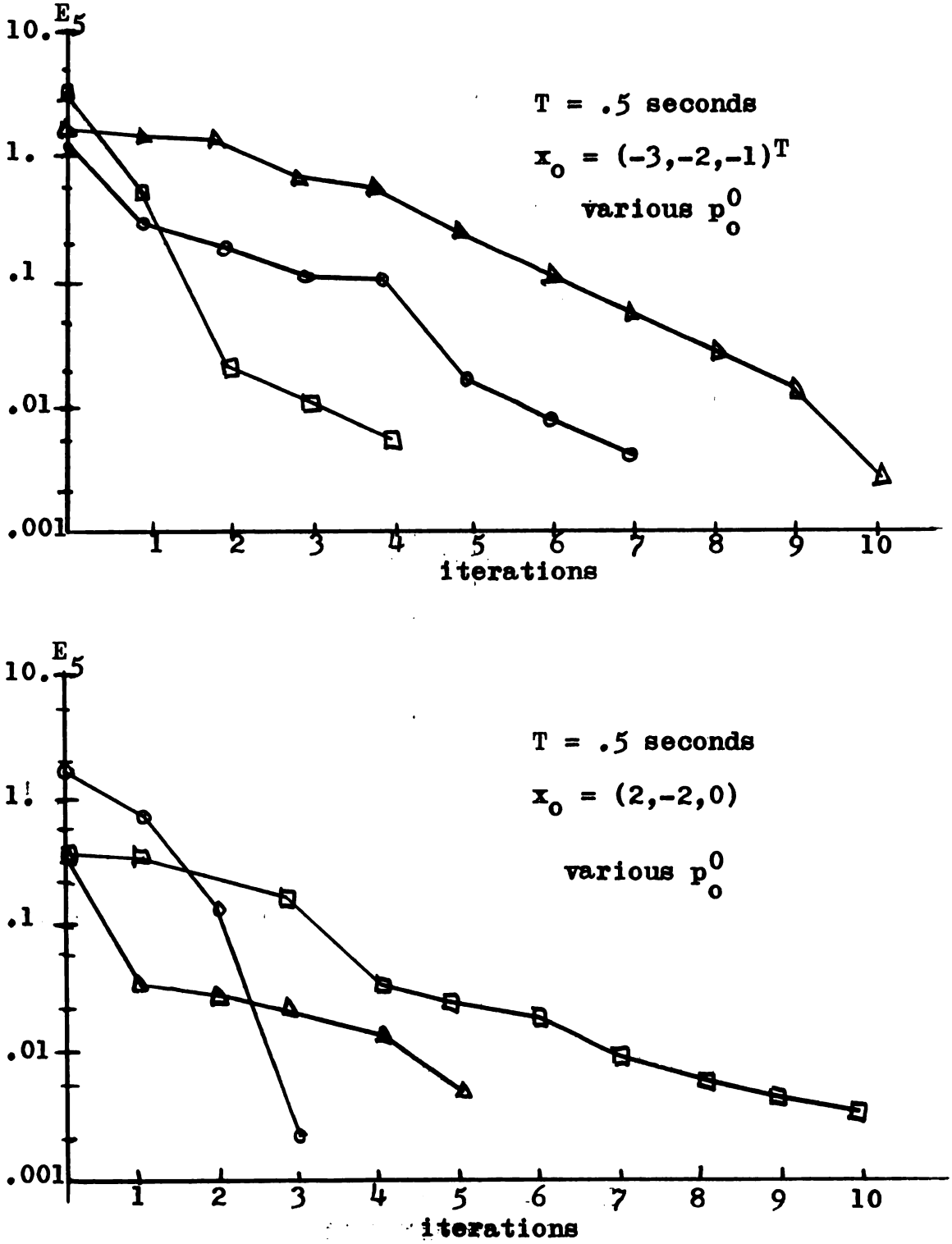


FIGURE 5.11 Example Iterations of GOP for Example Problem 2

reachable sets for ES-1. Example Problem 1 is chosen since the resulting reachable sets are particularly nonconvex and interesting in their behavior. Unfortunately, however, the sequence of optimum final states for these reachable sets only asymptotically approaches the origin as time is increased.

The first example considered is for the reachable set corresponding to the initial state $x_0 = (-10,-5)^T$. In Table 5.8 the results of some time increments are given. It should be observed that $||x_T||$ does approach the origin, but the convergence is asymptotic. Likewise, it should be noted that the sequence of $||x_T||$'s is not monotonic. These results are illustrated in the reachable sets corresponding to various times as given in Figure 5.12. From this figure it is easy to see the spiraling which R(t) does around the origin and its increasingly nonconvex nature. The locus of optimum final states (for the minimum distance problem) is also plotted in Figure 5.12.

The second example corresponds to the initial state $x_0 = (5,-5)^T$. In Table 5.9 the results for various final times are given. The reachable set at various times is plotted in Figure 5.13. Again, there is the spiraling effect noted in the previous example, with the reachable set becoming increasingly nonconvex as time increases.

Consideration of the initial adjoints listed in Tables 5.8 and 5.9 shows that there is a consistency in the manner

TABLE 5.8 Application of TOP to ES-1, $x_o = (-10,-5)^T$

Time	$\|\|x_T\|\|$	x_{1T}	x_{2T}	p_{1o}	p_{2o}
0.00	11.180	-10.000	-5.000	5.000	-10.000
5.00	20.980	-3.540	20.679	5.929	0.471
7.10	18.009	0.757	15.994	6.784	0.836
9.65	12.464	4.810	11.499	5.955	0.734
12.06	6.639	6.519	1.245	5.972	0.999
15.75	9.899	5.473	-8.250	5.998	0.110
19.17	9.792	1.001	-9.742	5.999	0.618
25.00	4.321	-4.077	-1.489	5.900	0.100
28.03	3.955	-3.881	0.756	5.882	-0.007
30.49	4.551	-3.662	2.710	6.000	0.000
33.53	4.309	-1.245	4.126	3.333	3.533
36.79	2.812	-0.170	2.807	4.117	4.364
38.47	2.044	0.268	2.026	3.817	3.470
40.04	1.809	0.659	1.684	3.942	4.322
41.49	1.659	0.903	1.391	4.043	4.443
42.86	1.624	1.098	1.196	4.043	4.433
44.26	1.519	1.098	1.049	4.984	3.504
45.60	1.442	1.196	0.805	5.449	3.249
46.92	1.391	1.098	0.854	6.377	1.072
48.22	1.334	1.098	0.756	6.038	0.674
49.51	1.343	1.196	0.610	5.962	0.666
50.84	1.358	1.293	0.415	5.962	0.666

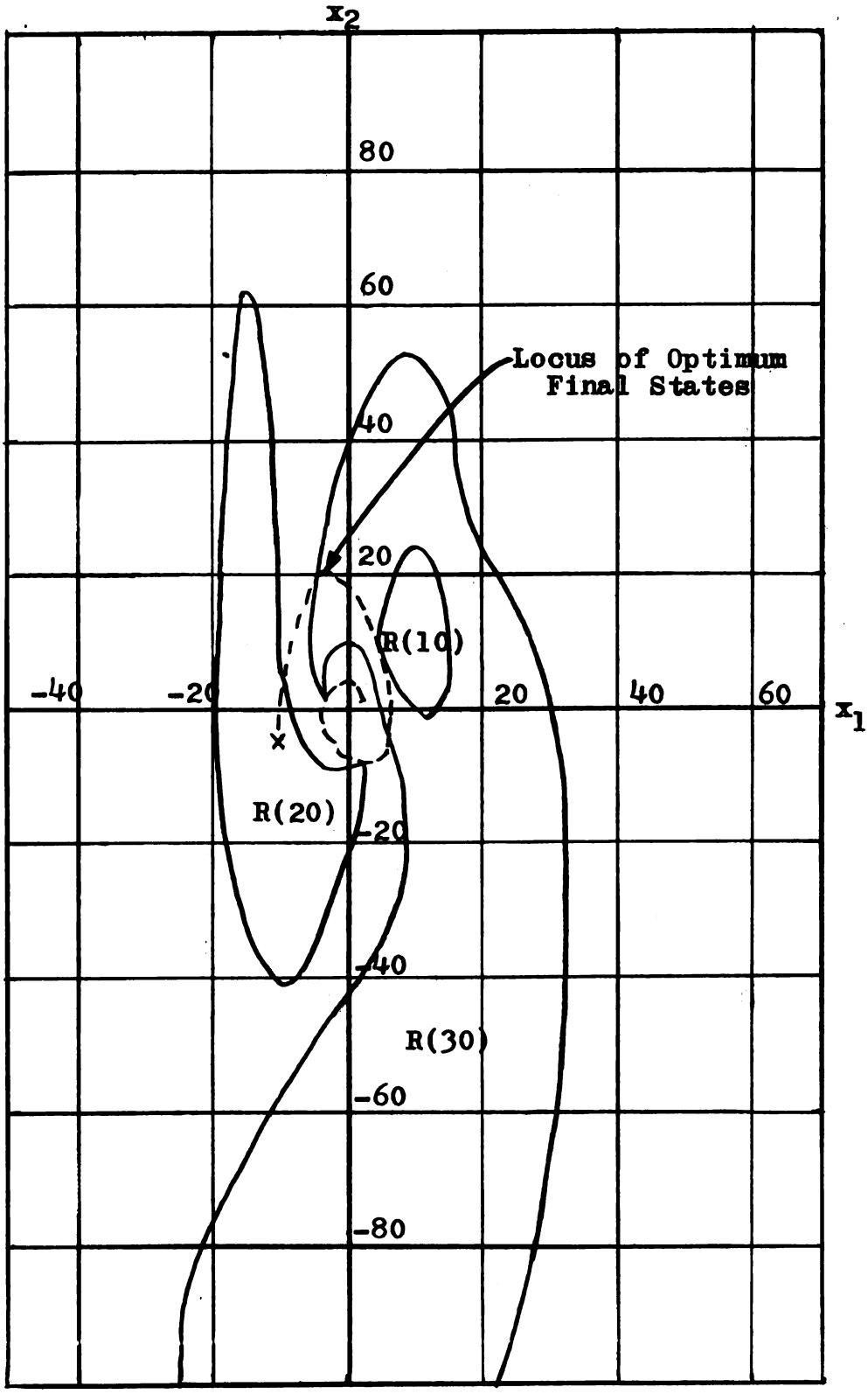


FIGURE 5.12 R(t) for ES-1, $x_0 = (-10,-5)^T$, Various t

TABLE 5.9 Application of TOP to ES-1, $x_0 = (5, -5)^T$

Time	$\|\| x_T \|\|$	x_{1T}	x_{2T}	p_{1o}	p_{2o}
0.00	7.071	5.000	-5.000	5.000	5.000
5.00	7.758	1.342	-7.641	-5.992	0.395
5.78	7.242	0.756	-7.202	-5.983	0.251
6.12	6.716	0.170	-6.714	-5.994	0.251
7.50	6.142	-0.415	-6.128	-5.994	0.251
8.42	5.679	-1.001	-5.590	-5.994	0.251
9.38	5.081	-1.489	-4.858	-5.988	0.188
10.33	4.422	-1.928	-3.979	-5.990	0.002
12.80	3.089	-2.758	-1.391	-5.035	-0.134
14.49	2.764	-2.758	-0.170	-6.342	1.556
15.29	2.741	-2.710	0.415	-5.349	2.211
16.13	2.843	-2.661	1.001	-5.577	2.211
17.05	2.964	-2.563	1.489	-5.577	2.211
19.16	3.004	-2.075	2.172	-6.453	2.011
21.50	2.702	-1.538	2.221	-6.214	1.081
23.74	2.107	-0.366	2.075	-5.955	-1.137
25.65	1.587	-0.024	1.586	-5.893	-1.125
27.23	1.348	0.122	1.342	-5.881	-0.995
28.66	1.202	0.122	1.196	-5.950	-0.766
30.04	1.143	0.317	1.098	-5.950	-0.766
32.02	1.038	0.415	0.952	-5.968	-0.612
34.12	0.911	0.317	0.854	-6.679	-0.036
35.37	0.929	0.463	0.805	-5.999	-0.032
36.03	0.954	0.512	0.805	-5.999	-0.032

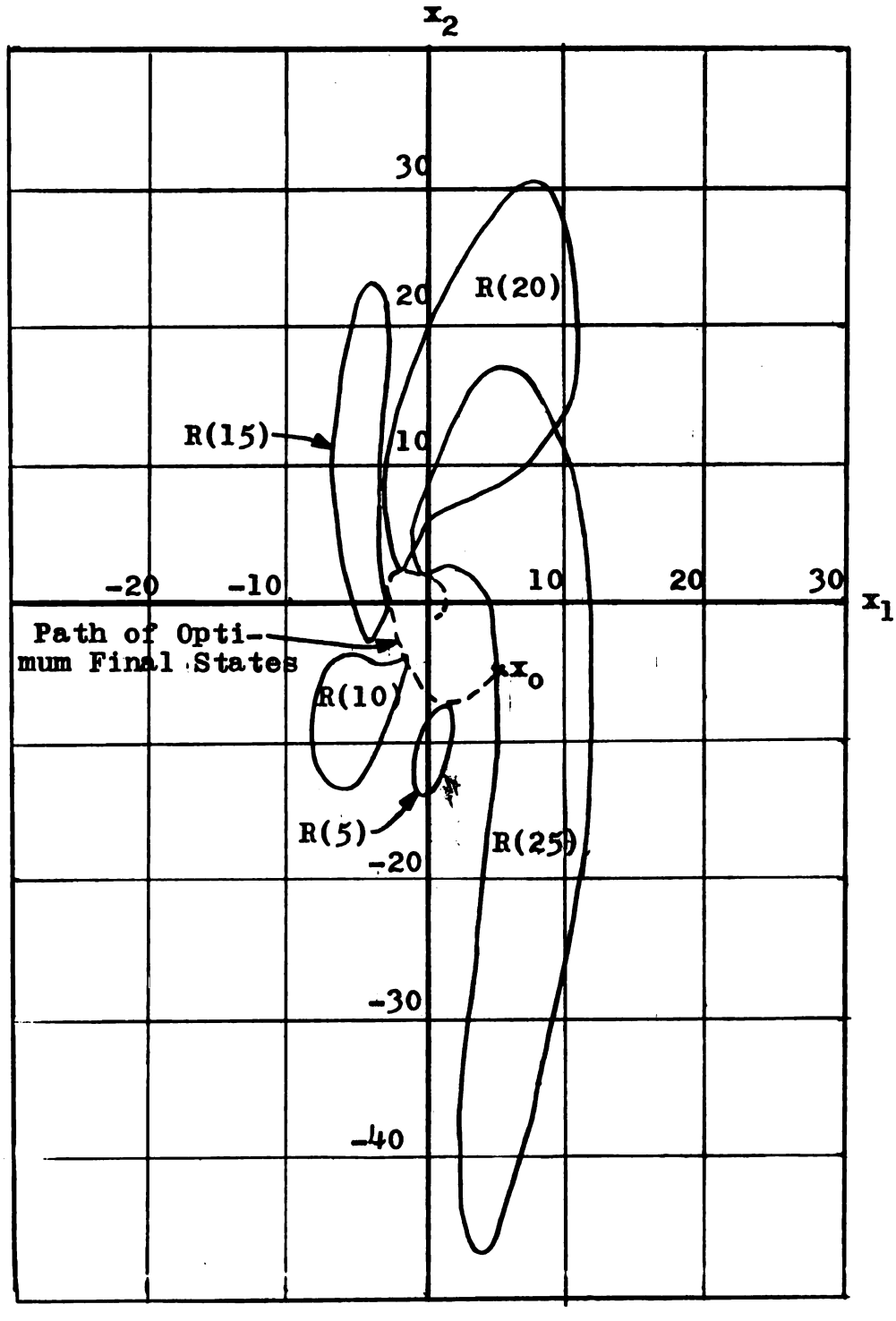


FIGURE 5.13 R(t) for ES-1, $x_o = (5,-5)^T$, Various t

in which the optimum initial costate changes as time is incremented. In the examples of this section, use is made of this fact. For the first local optimum (T = 5 seconds) LOP-CM is employed. For succeeding optima, the change in the initial adjoints is sufficiently small to optimize using LOP-DS. In fact, in several instances the i^{th} p_o^* is also the $^{i+1st}$ p_o^* (See Table 5.9, time = 6.12, 7.50 and 8.42).

5.7 Summary and Conclusions

It is the purpose of this section to present a general summary of the theoretical developments, computational results and comparisons of the previous sections of this thesis. Conclusions which are given in previous sections of this chapter are also briefly summarized in this section. Additional comments or conclusions which seem appropriate are also made.

As indicated by the title, the purpose of this thesis is to consider the computation of optimal controls for nonlinear systems. Rather general nonlinear systems are allowed. The restrictions placed on these systems are given in Section 2.2. The concept of the reachable set is introduced and investigated. A number of related definitions and results are given in Chapter 2.

The minimum-error regulator problem is principally treated and computational methods are developed to solve this problem. The problem is somewhat difficult when

nonlinearities are allowed in the state equations. The nonlinearities have a two-fold effect on the computation of an optimal control for the minimum distance problem. First of all, they introduce the possibility of the occurrence of several or even many local optima since the resulting reachable set is not convex. Secondly, the determination of a local optimum is made more difficult since the adjoint system becomes dependent on the state.

Although many reachable set-oriented methods have previously been utilized, their application to nonlinear systems has been limited. One of the most restrictive limitations is a result of the dependence of the adjoint system of equations on the state variables. As a result, the direct determination of an initial adjoint given a final adjoint is not possible without knowing the corresponding state trajectory.

To overcome this difficulty, the author decided to place the emphasis of his computational approach on the initial adjoint. Once the initial adjoint is specified, it is possible to compute the state response, a maximal trajectory, and to identify a boundary point on the reachable set.

An intrinsic part of any effective computational method is a means of evaluating each iteration and of determining when an optimum has been achieved. This is the purpose for the error function. One obvious error function

for the minimum distance problem is E_1, the norm of the final state. Other error functions are available and are introduced in Chapter 3. It is shown that for the minimum distance problem, the final state and the final adjoint are collinear at the optimum. This leads to a second error function:

$$E_2 = \cos \gamma = \frac{\langle x , p \rangle}{\|x\| \|p\|}, \tag{5.24}$$

where x is the final state and p is the final adjoint. This error function is more sensitive to changes in extremal trajectories than is $\|x\|$. Also, $\|x\|$ may just be approaching an unspecified minimum whereas E_2 approaches -1 at the optimum. For these reasons, E_2 is usually employed in combination with E_1:

$$E_5 = (1 + E_2) E_1. \tag{5.25}$$

One approach to the solution of the minimum error regulator problem is a direct search (related to gradient techniques) method based on varying the initial adjoints such that the error function is improved. A direct search procedure is given on pages 41 and 42 and a flow chart of this method is given in Figure 3.1.

Since the direct search technique is inefficient, other possible methods are considered. To develop another method, the determination of the optimal control can be viewed as the determination of a sequence of initial adjoints which, in turn, determines a sequence of extremal endpoints starting at an arbitrary final state and

terminating at an optimum. Thus the problem is one of defining a path on the boundary of the reachable set, specifically a path for which the error functions decrease monotonically.

To show that such a path exists and to give insight into the nature of this path, various principles and facts from differential geometry are utilized. Specifically, it is possible to prove that on a locally convex surface, a satisfactory path can be constructed from lines of curvature. Requiring that the path consist of lines of curvature gives a relationship between the perturbations of the final state and the final adjoint on the boundary of the reachable set:

$$k\,\delta x + \delta p = 0, \tag{5.26}$$

where k represents the curvature of the surface at the point x on the line of curvature. This aids in the implementation of a procedure using this geometric approach. Chapter 3 develops this procedure and considers the alternative choices which are encountered.

Several interesting facts develop as a result of these geometric considerations. To prove the monotonicity of the error function E_2, it is necessary to also prove the monotonicity of the error function E_1 along the desired path. As a result of this proof the importance of the curvature of the surface is demonstrated (see Equations 3.66 and 3.72). These facts lead to Theorem 4.1 which analyzes the

effect of concave surfaces on the procedure.

The alternative choices available in the procedure can generally be classified as those which are available on the boundary of the reachable set (perturbation of the final state and the final adjoint to yield a better estimate of the initial state, hence of an optimum) and those which are important at the initial time (perturbation of the initial adjoint).

The computations summarized in Section 5.3.1 indicate that the final time perturbation is most effective if both the final state and the final adjoint are perturbed as related through Equation 5.26. The perturbation of x_T is considered to be the independent perturbation, hence it is important to consider the direction and magnitude of this perturbation. Sections 5.3.5 and 5.3.6 consider these choices and indicate that the best means of perturbing the final state x_T is:

$$\delta x_T = -c(x_T - \| x_T \| \frac{p_T}{\| p_T \|} E_2) \qquad (5.27)$$

where c represents the step size. Consideration of the effect of the step size demonstrates that the method is most effective if c is made dependent on the error at each iteration.

The perturbation of the final adjoint is related to the perturbation of the final state through Equation 5.26, hence determination of the curvature k is important. Sections 5.3.2 through 5.3.4 consider the various methods

of estimating curvature and of evaluating the validity of the estimate. Examples of estimates of curvature are given in these sections and shown to correspond to the values expected from reachable set geometry.

The analog correction factors α and β introduced in Section 3.6.2 are shown to improve the convergence of LOP-CM using analog integration. While the experimentation of this thesis was not intended to evaluate the relative merits of analog or hybrid computation versus digital computation, the necessity of these correction factors introduces this subject. The advantages and disadvantages of both are apparent in the computations of this thesis. No decision is clearly obvious concerning their relative merits, but digital computation is more reliable.

While the hybrid computer has the advantage of speed of integrations, it has disadvantages also. Accuracy is limited due to the analog elements employed and to analog-digital and digital-analog conversion limits. Because of the nonlinear systems considered and the discontinuities introduced by the controls, the trajectories are very sensitive to hystersis of comparitors and switches, inaccuracies in multipliers, integrators, etc. Thus there is a large amount of sensitivity to the control switching times.

On the other hand, the digital computer is consistent and repeatable in its results. The accuracy of the integration can be controlled by controlling the step size.

However, the major drawback of the digital computer is its speed of integration of the differential equations. Thus, as previously indicated, the decision as to whether to use analog or digital computers for the differential equation integration must be based on the nature of the systems being considered, the accuracy desired, the computational equipment available and the speed of computation desired. For the equipment available to the author, total digital computation was preferred because of its accuracy.

Both local optimum procedures, LOP-CM and LOP-DS, were utilized and performed as expected--converging to local optima. The convergence of LOP-CM, as expected, was more rapid than that of LOP-DS. In none of the experimentation performed for this thesis were any limit points of LOP-CM or LOP-DS encountered. The only failures of these methods to converge can be traced to analog integration inconsistencies.

Some of the special problems introduced in Chapter 4 were encountered. Special Problem 1, relating to concave curvatures, was encountered in Example 5.5.1 and the results verify those predicted by Theorem 4.1. No flats or singularities were noted, nor were false boundary points and interior boundaries. Corners were produced in several examples, but caused no special problem. In most cases, in fact, the corner is the optimum.

Generating a sequence of local minima did locate the

global optimum in the examples considered (See Section 5.5). Only one optimum was computed for the reachable sets corresponding to Example Problems 2 through 4. While this is a disappointment from the standpoint of effectively testing the Global Optimum Procedure, it is encouraging in that it indicates that the reachable sets corresponding to many nonlinear problems are not extremely badly behaved.

In Section 5.6, the time optimal control problem is considered and example computation is done. Examination of this computation demonstrates the mobility and changing shape of the reachable sets with increasing time. It is important to note that for the examples considered, the initial adjoint (corresponding to the optimal control) did not significantly change as time was increased.

In summary it can be concluded that the theory of Chapters 2 through 4 provides the basis for an effective computational procedure. The consideration of the algorithm alternatives in Chapter 5 verifies the choices made in Chapter 3. The resulting LOP-CM is effective in the determination of local minima and, when employed as part of a global procedure, determines the global optimum.

Certainly there are many areas open for future investigation. Possibly other principles of differential geometry can be brought to bear on the optimization problem (utilization of geodesics, for instance). Other applications of LOP-CM, GOP and TOP could be considered as well

as comparisons with other existing methods. Possible future investigations and extensions are suggested in Appendix D.

BIBLIOGRAPHY

BIBLIOGRAPHY

A1 Athans, M. and P.L. Falb, "Optimal Control," McGraw-Hill Book Co., New York, (1966).

B1 Barr, R. O., "Computation of Optimal Controls by Quadratic Programming on Convex Reachable Sets," Ph.D. thesis, University of Michigan, 1966.

B2 Barr, R. O. and E. G. Gilbert, Some Efficient Algorithms for a Class of Abstract Optimization Problems Arising in Optimal Control, To Appear in IEEE Transactions on Automatic Control, 1969.

B3 Bellman, R. "Dynamic Programming," Princeton University Press, Princeton, N.J., (1957).

B4 Bellman, R. and R. Kalaba, Dynamic Programming, Invariant Imbedding and Quasilinearization: Comparisons and Interconnections, in "Computing Methods in Optimization Problems," Academic Press, New York, (1964), pp 135-145.

B5 Bellman, R. and R. E. Kalaba, "Quasilinearization and Boundary Value Problems," American Elsevier Publishing Co., New York, (1965).

B6 Benson, R., "Euclidean Geometry and Convexity," McGraw-Hill Book Company, New York, (1966).

B7 Bryson, A. E. and W. F. Denham, A Steepest-Ascent Method for Solving Optimum Programming Problems, J. Appl. Mech. Ser E. Vol 29, (1962), pp. 247-257.

D1 de Jong, J. L., "Application of Picard's and Newton's Methods to the Solution of Two-Point Boundary-Value Problems in Optimal Control Theory," Ph.D. Thesis, University of Michigan, 1967.

E1 Eaton, J. H., An Iterative Solution to Time-Optimal Control, J. Math. Anal and Appl., Vol 5, (1962), pp 329-344; Errata and Addenda, J. Math. Anal. and Appl., Vol. 9, (1964), pp 147-152.

F1 Fadden, E.J., "Computational Aspects of a Class of Optimal Control Problems," Ph.D. Thesis, University of Michigan, 1965.

G1 Gilbert, E. E., An Iterative Procedure for Computing the Minimum of a Quadratic Form on a Convex Set, SIAM J. on Control, Ser. A, Vol 4, No. 1, (1966), pp 61-80.

G2 Gerretsen, J. C. H., "Lectures on Tensor Calculus and Differential Geometry," P. Noordhoff, Groningen, The Netherlands, (1962).

H1 Hadley G., "Nonlinear and Dynamic Programming," Addison-Wesley, Reading Mass., (1964).

H2 Halkin, H., Method of Convex Ascent, in "Computing Methods in Optimization Problems," Academic Press, New York, (1964), pp. 211-239.

H3 Hestenes, M.R. and T. Guinn, An Embedding Theorem for Differential Equations, J. of Optimization Theory and Applications, Vol. 2, No. 2 (March 1969), pp 87-101.

H4 Hestenes, M. R., "Calculus of Variations and Optimal Control Theory," John Wiley and Sons, New York, (1967).

H5 Ho, Y. E. A Successive Approximation Technique for Optimal Control Systems Subject to Input Saturation, Trans. ASME, J. Basic Eng., Series D, V. 82, (1960), pp. 33-40.

H6 Hooke, R. and T. A. Jeeves, "Direct Search" Solution of Numerical and Statistical Problems, J. Assoc. Comp. Mach., Vol 8, No. 2, (April 1961), pp 212-229.

H7 Hsu, J.C. and A. U. Meyer, "Modern Control Principles and Its Applications," McGraw Hill Book Co., New York, (1968).

K1 Knudsen, H. K., An Iterative Procedure for Computing Optimal Controls, IEEE Trans. on Auto. Control, Vol. AC-9, No. 1, (1964), pp 23-30.

K2 Kopp, R. E. and R. McGill, Several Trajectory Optimization Techniques, in "Computing Methods in Optimization Problems," Academic Press, New York, (1964), pp. 65-89.

K3 Ku, Y. H., "Analysis and Control of Nonlinear Systems," The Ronald Press Co., New York, (1958).

L1 Lapidus, L. and R. Luus, "Optimal Control of Engineering Processes," Blaisdell Publishing Co., Waltham, Mass., (1967).

L2 Lee, E. B. and L. Markus, "Foundations of Optimal Theory," John Wiley and Sons, New York, (1967).

N1 Neustadt, L. W., Synthesizing Time Optimal Control Systems, J. Math. Anal. and Appl., Vol 1, (1960), pp. 484-492.

N2 Neustadt, L. W. and B. Paiewonsky, On Synthesing Optimal Controls, in "Proceedings of the Second Congress of the International Federation of Automatic Control (IFAC), Basel, 1963", Butterworth, London.

P1 Plant, J. B., "Some Iterative Solutions in Optimal Control," The M.I.T. Press, Cambridge, Mass., (1968).

P2 Pontryagin, L. S., V. G. Boltyanskii, R. V. Gamkrelidze and E. F. Mischenko, "The Mathematical Theory of Optimal Processes," John Wiley and Sons, Inc., New York, (1962).

R1 Roxin, E., A Geometric Interpretation of Pontryagin's Maximum Principle, in "Nonlinear Differential Equations and Nonlinear Mechanics," Academic Press, New York, (1963), pp.

S1 Schliching, H., "Boundary Layer Theory", 6th Ed., McGraw Hill Book Co., New York, (1968).

T1 Tapley, B.D. and J. M. Lewallen, Comparison of Several Numerical Optimization Methods, J. of Optimization Theory and Applications, Vol. 1, No. 1, (1967), pp 1-32.

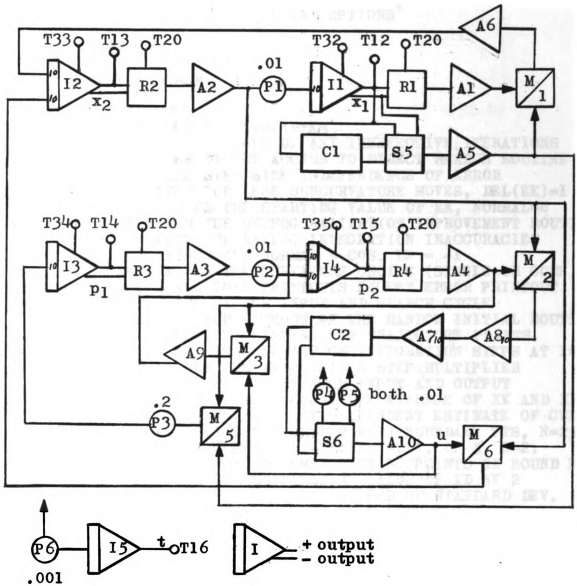
W1 Willmore, T. J., "An Introduction to Differential Geometry," The Clarendon Press, Oxford, England, (1959).

Z1 Zukhovitskiy, S. I. and L. I. Avdeyera, "Linear and Convex Programming," W. B. Saunders Co., Philadelphia, (1966).

APPENDICES

APPENDIX A

ANALOG DIAGRAM FOR EXAMPLE PROBLEM 1



I indicates an integrator; A indicates an amplifier;
R indicates a relay; P indicates a potentiometer;
M indicates a multiplier; C indicates a comparitor;
T indicates a trunk line (to the digital computer);
S indicates an electronic switch.

All amplifiers and integrators consisted of two operational amplifiers; hence both a positive and negative output were available.

Trunk 20 was used to switch the system for reverse time integration.

APPENDIX B

EXAMPLE COMPUTER PROGRAMS

1. GLOBAL OPTIMUM PROCEDURE AND OPTIONS*

```
// FOR GLOP2
*IOCS(CARD,1443 PRINTER)
*LIST SOURCE PROGRAM
*NONPROCESS PROGRAM
*ONE WORD INTEGERS
C DATSW 0  PROVIDES FOR XM=-XM*XM
C DATSW 1  PROVIDES KICKOUT OF ANY INEFFECTIVE ITERATIONS
C DATSW 2  PROVIDES DIRECT ACCESS TO DIRECT SEARCH ROUTINE
C DATSW 3  PROVIDES STEP SIZE INDEPENDENCE OF ERROR
C DATSW 4  PROVIDES FOR LESS SUBCURVATURE MOVES, DEL(KK)=1
C DATSW 5  DETERMINES THE STARTING VALUE OF KK, NORMAL=0
C DATSW 6  BYPASSES THE ORTHOGONALIZATION IMPROVEMENT ROUTINE
C DATSW 7  CORRECTS FOR ANALOG INTEGRATION INACCURACIES
C DATSW 8  DETERMINES XM, NORMAL = COS, UP = -1
C DATSW 9  DETERMINES THE CURVATURE CHOICE (SINGLE OR COMP.)
C DATSW 10  STOPS ALL GOAN PRINTOUTS EXCEPT ERROR PRINTOUT
C DATSW 11  SKIPS THE RANDOM INPUT AND SEARCH CYCLE
C DATSW 12  PROVIDES FOR KICKOUT OF THE RANDOM INITIAL ROUTINE
C DATSW 13  PROVIDES FOR PRINTING OUT TRAJECTORY POINTS
C DATSW 14  SETS THE BASIC NUMBER OF INTEGRATION STEPS AT 10
C DATSW 15  DETERMINES THE INTEGRATION STEP MULTIPLIER
C SSWTCH 0  PROVIDES FOR TYPEWRITER INPUT AND OUTPUT
C SSWTCH 1  PROVIDES FOR THE CURVATURE AVERAGE OF XK AND XKQ
C SSWTCH 2  PROVIDES FOR ST. DEV. DEPENDENT ESTIMATE OF CURV.
C SSWTCH 3  PROVIDES A BASIS OF 100 R(T) BOUNDARY PTS, N=25.
C SSWTCH 4  PROVIDES AN R(T) DATA POINT MULT. OF 5, N=2.
C SSWTCH 5  PROVIDES FOR RANDOMLY PICKING POINTS ON BOUND R(T)
C SSWTCH 6  PROVIDES FOR MULTIPLYING VALUE OF XD BY 2
C ITI  (FIRST READ) SPECIFIES THE NUMBER OF STANDARD DEV.
     POSSIBILITIES OR CORNER CURVATURE ATTEMPTS
C TOL1 (2ND READ)  SPECIFIES THE KICKOUT ERROR TO THE DIRECT
     SEARCH ROUTINE
C TOL2 (3RD READ)  SPECIFIES THE FINAL ERROR TOLERANCE
C TOL3 (4TH READ)  SPECIFIES THE STANDARD DEVIATION MAXIMUM
      DIMENSION X(4),P(4),XI(4),PI(4),XJ(4),PJ(4),Q(4),XL(4),
          XC(4),PC(4),XS(4,10),PS(4,10),XERS(11)
      IY=5349
      WRITE (1,2184)
2184  FORMAT ('()','(       )','(       )','(       )',
          '(       )')
      READ (6,2183) IT1,TOL1,TOL2,TOL3
2183  FORMAT (I2,3F10.4)
      CALL DIGO (20,-1)
      READ (2,2)T,N
2     FORMAT (F10.4,I1)
```

```
3     FORMAT (8F10.3)
4     READ (2,3) (XI(I),I=1,N),(PI(I),I=1,N)
      WRITE (3,401)
401   FORMAT (' THE FOLLOWING DATA SWITCHES WERE UP   ')
      DO 404 I=1,16
      IWW=I-1
      CALL DATSW(IWW,IW)
      GO TO (402,404),IW
402   WRITE (3,403) IWW
404   CONTINUE
403   FORMAT (I40)
      WRITE (3,406)
406   FORMAT ('  THE FOLLOWING SENSE SWITCHES WERE UP   ')
      DO 408 I=1,8
      IWW=I-1
      CALL SSWTCH (IWW,IW)
      GO TO (407,408),IW
407   WITE (3,403)IWW
408   CONTINUE
      KY=1771
      ITKM=0
      CALL SSWTCH (5,IR)
      GO TO (1000,1011),IR
1000  READ (6,1117) IRSY
1117  FORMAT (I4)
      CALL SSWTCH (3,II)
      GO TO (1001,1002),II
1001  INRS=100
      GO TO 1003
1002  INRS=25
1003  CALL SSWTCH(4,JJ)
      GO TO (1004,1005),JJ
1004  INRS=INRS*5
      GO TO 1006
1005  INRS=INRS*2
1006  DO 1010 I=1,INRS
      DO 1007 J=1,N
      IRSX=IRSY
      CALL RANDU (IRSX,IRSY,RSY)
1007  PI(J)=RSY-.5
      CALL SSWTCH (0,I00)
      GO TO (1020,1010),I00
1020  WRITE (1,1475)
1475  FORMAT ('(         )(         )(         )')
      READ (6,3) (PI(I),I=1,N)
1010  CALL GOAN(N,T,XI,PI,X,P,X2,P2,XP,ER)
1011  CALL DATSW (11,IRX)
      GO TO (490,491),IRX
490   IZA=10
      GO TO 492
491   IZA = 0
      IY=5349
8     DO 882 I=1,N
      IX=IY
```

```
      IX=IX
    8 DO 882 I=1,N
      IY=5*I+6
      IZA = 0
      GO TO 492
  402 IZA=10
      GO TO (490,401),IRX
 1011 CALL DATSW (II,IRX)
 1010 CALL GOAM(K,X1,Y1,I,X,Y,X2,Y2,LP,LEN)
      READ (6,3) (PI(I),I=1,N)
 1025 FORMAT (4(          )(          )(          )/)
      WRITE (3,1025)
      GO TO (1020,100),IRX
 1020 CALL SSWTCH (0,100)
 1002 PI(4)=RAT-.5
      CALL MAMUL (IRAX,IRAY,NAY)
      IRAX=IRAY
      DO 1002 J=1,N
 1000 DO 1010 I=1,IXRS
      GO TO 1006
 1004 IXRS=IXRS*2
      GO TO 1000
 1003 CALL SSWTCH(4,J1)
      GO TO (1004,1003),J1
 1002 IXRS=25
      GO TO 1003
 1001 IXRS=100
      GO TO (1001,1002),II
      CALL SSWTCH (3,II)
 1112 FORMAT (1H)
 1000 READ (6,1112) IRS1
      CALL SSWTCH (2,IR)
      GO TO (1000,1011),IR
      IZZM=0
      KZ=IZ*I
  404 CONTINUE
  402 WRITE (3,403))IMM
      GO TO (402,408),IM
      CALL SSWTCH (IMM,IM)
      IMM=I-1
      DO 408 I=1,8
  406 FORMAT (' THE FOLLOWING SENSE SWITCHES WERE UP ')
      WRITE (3,406)
  403 FORMAT (1H0)
  404 CONTINUE
  402 WRITE (3,403) IMM
      GO TO (402,408),IM
      CALL DATSW(IMM,IM)
      IMM=I-1
      DO 404 I=1,16
  401 FORMAT (' THE FOLLOWING DATA SWITCHES WERE UP ')
      WRITE (3,401)
    4 READ (5,3) (XI(I),I=1,N),(PI(I),I=1,N)
    3 FORMAT (8F10.3)
```

```
      CALL RANDU (IX,IY,Y)
      CALL DATSW (12,IRK)
      GO TO (9,882),IRK
882   PI(I)=Y-.5
492   WRITE (3,101)(XI(I),I=1,N)
101   FORMAT (///4F20.8)
11    CALL GOAN(N,T,XI,PI,X,P,X2,P2,XP,ER)
      ERS=ER
      IF(ER-TOL2)5,5,379
379   ERX=ER/X2)-1.
      IF (ERX) 4923,13,13
13    DO 14 I=1,N
14    PI(I)=-PI(I)
      CALL GOAN (N,T,XI,PI,X,P,X2,P2,XP,ER)
      ERS=ER
      ERX=(ER/X2)-1.
      IF(ERX) 4923,8,8
4923  CALL DATSW(2,IE)
      GO TO (1113,12),IE
1113  HH=SQRT(ERS)+.5
      DO 1322 I=1,N
1322  Q(I)=PI(I)
      ERD=ERS
      GO TO 1662
12    PZ=.05
      XPI=0.
      DO 555 I=1,N
555   XPI=XPI+PI(I)*PI(I)
      XPI=SQRT(XPI)
      DO 556 I=1,N
556   PI(I)=PI(I)*5./XPI
      PC0=0
      DO 141 I=1,N
      KX=KY
      CALL RANDU (KX,KY,YK)
      PC(I)=YK-.5
141   PC0=PC0+PC(I)*PC(I)
      PC0=SQRT(PC0)
      DO 1411 I=1,N
1411  PC(I)=PC(I)/PC0
1321  LL=0
      IF(ERS-TOL1)20,20,1212
1212  PZ=5.*PZ
      LL=LL+1
      IF(LL-4) 217,217,2181
2181  IF(ITKM-IT1)2182,2182,218
2182  ITKM=ITKM+1
      GO TO 12
218   DO 219 I=1,N
219   XL(I)=-100.
      XK=-100.
      XKQ=-100.
```

```
      GO TO 220
217   DO 216 I=1,N
216   Q(I)=PI(I)+PZ*PC(I)
      CALL SSWTCH (0,I00)
      GO TO (2160,2162),I00
2160  WRITE (1,1475)
      READ (6,3) (Q(I),I=1,N)
2162  CALL GOAN (N,T,XI,Q,XJ,PJ,X2D,P2D,XPD,ERD)
      IF(ERD-TOL2) 1,1,2161
2161  IF(ERD-TOL1) 1662,1662,2163
2163  CALL DATSW (6,I0)
      GO TO (160,184),I0
184   IF(.95-ERD/ERS)160,160,180
180   DO 181 I=1,N
      PI(I)=Q(I)
      X(I)=XJ(I)
181   P(I)=PJ(I)
      ER=ERD
      ERS=ER
      X2=X2D
      P2=P2D
      GO TO 217
160   XK=0.0
      DO 161 I=1,N
      IF (ABS(XJ(I)-X(I))-.00001)  1212,1212,162
162   XL(I)=((PJ(I)/P2D)-(P(I)/P2))/(X(I)-XJ(I))
161   XK=XK+XL(I)
      XK=XK/N
      WRITE (3,102) (XL(I),I=1,N),XK
      C1=XP/(P2*x2)
      C2=XPD/(P2D*X2D)
      XKQ=(C1-C2)/(X2D-X2)-C1/X2
      XKAV=(XK+XKN)/2.
220   WRITE (3,102) XKQ,XKAV
102   FORMAT (7F15.5)
      CALL SSWTCH (2,IV)
      GO TO (170,179),IV
170   AAK=0
      VAR=0
      DO 171 I=1,N
      VAR=XL(I)*XL(I)+VAR
171   AAK=AAK+ABS(XL(I))
      VAR=VAR/N-XK*XK
      STDV=SQRT(VAR)
      AAK=AAK/N
      TEST=STDV/AAK
      WRITE (3.102) VAR,STDV,AAK,TEST
      ITKM=ITKM+1
      IF(TEST-TOL3)179,179,1791
1791  IF (ITKM-IT1) 12,12,179
179   IF(ERD-ER) 30,30,20
20    DO 21 I=1,N
      Q(I)=PI(I)
```

```
      XJ(I)=X(I)
21    PJ(I)=P(I)
      ERD=ER
      X2D=X2
      ERS=ER
      P2D=P2
30    CALL DATSW (5,IMX)
      GO TO (3111,3112),IMX
3111  KK=-3
      GO TO 3113
3112  KK=0
3113  HH=SQRT(ERS)+.5
      IF (ERS-TOL2) 1,1,3211
3211  CALL DATSW (3,IS)
      GO TO (1201,1200),IS
1201  XD=.2
      GO TO 1202
1200  XD=ERS/X2D
1202  CALL SSWTCH (6,IDM)
      GO TO (1203,1204),IDM
1203  XD=XD*2.
1204  WRITE (3,102) XD
32    WRITE (3,102)(XL(I),I=1,N),XK
      IF(ERS-TOL1) 1662,1662,3291
3291  CALL DATSW (4,IKK)
      GO TO (3293,3292),IKK
3292  KK=KK+1
      GO TO 3294
3293  KK=KK+2
3294  IF(KK-2)1661,1661,1662
1663  HH=HH*.5
      IF(HH-.005)1,1664,1664
1664  WRITE (3,102) HH
      CALL DATSW (1,MM)
      GO TO (1,1662),MM
1662  CALL EXPLR(N,T,XI,Q,XJ,PJ,X2D,P2D,XPD,ERD,HH)
      IF (ERD-TOL2)1,1,1665
1665  IF (ERD-ERS)1666,1663,1663
1666  ERS=ERD
      GO TO 1664
1661  CALL DATSW (1,II)
      GO TO (1,166),II
166   CALL DATSW(8,IXM)
      GO TO (1671,1672),IXM
1671  XM=-1.
      GO TO 1673
1672  XM=(ERS/X2D)-1.
C     XM=0
1673  CALL DATSW (0,IYM)
      GO TO (1674,1675),IYM
1674  XM=-XM*XM
1675  WRITE (3,102) XM
      DO 35 I=1,N
```

```
      DX=-XD*(XJ(I)-(X2D*PJ(I)*XM)/P2D)
      CALL SSWTCH (1,IKAV)
      GO TO (1214,1213),IKAV
1214  P(I)=PJ(I)-P2D*DX*XKAV
      GO TO 35
1213  CALL DATSW (9,IK)
      GO TO (1210,1211),IK
C     P(I)=PJ(I)
C     P(I)=PJ(I)-P2D*DX*XK
1210  P(I)=PJ(I)-P2D*DX*XL(I)
      GO TO 35
1211  P(I)=PJ(I)-P2D*DX*XKQ
C     X(I)=XJ(I)
35    X(I)=XJ(I)+DX
      CALL DIGO (20,1)
      WRITE (3,102) (X(I),I=1,N)
      WRITE (3,102) (P(I),I=1,N)
      CALL GOAN (N,T,X,P,X,P,X2,P2,XP,ER)
      CALL DATSW (7,JL)
      GO TO (891,886),JL
891   CALL GOAN (N,T,XJ,PJ,XC,PC,X2C,P2C,XPC,ERC)
      DO 888 I=1,N
      XC(I)=XC(I)-XI(I)
888   PC(I)=PC(I)-Q(I)
      WRITE (3,102) (XC(I),I=1,N)
      WRITE (3,102) (PC(I),I=1,N)
      DO 889 I=1,N
      X(I)=X(I)-XC(I)
889   PI(I)=P(I)-PC(I)
      WRITE (3,102) (X(I),I=1,N)
      WRITE (3,102) (PI(I),I=1,N)
      DO 890 I=1,N
890   PI(I)=PI(I)
      GO TO 887
886   DO 885 I=1,N
885   PI(I)=P(I)
887   CALL DIGO (20,-1)
      CALL GOAN (N,T,XI,PI,X,P,X2,P2,XP,ER)
      IF (ER-ERS) 42,41,41
41    XD=XD*.2
412   WRITE (3,102)XD
      GO TO 32
42    ERS=ER
      WRITE (3,102) ERS,ERS,ERS,ERS
      ITKM=0
      IF (ERS-TOL2) 5,5,12
5     DO 6 I=1,N
      Q(I)=PI(I)
6     XJ(I)=X(I)
      X2D=X2
1     DO 7 I=1,N
      PS(I,IZA)=Q(I)
7     XS(I,IZA)=XJ(I)
      XERS(IZA)=X2D
      IZA=IZA+1
      IF(IZA-11)8,997,997
```

```
      DX=-XD*(XJ(I)-(X2D*PJ(I)*XM)/P2D)
      CALL SSWTCH (1,IKAV)
      GO TO (1214,1213),IKAV
1214  P(I)=PJ(I)-P2D*DX*XKAV
      GO TO 35
1213  CALL DATSW (9,IK)
      GO TO (1210,1211),IK
C     P(I)=PJ(I)
C     P(I)=PJ(I)-P2D*DX*XK
1210  P(I)=PJ(I)-P2D*DX*XL(I)
      GO TO 35
1211  P(I)=PJ(I)-P2D*DX*XKQ
C     X(I)=XJ(I)
35    X(I)=XJ(I)+DX
      CALL DIGO (20,1)
      WRITE (3,102) (X(I),I=1,N)
      WRITE (3,102) (P(I),I=1,N)
      CALL GOAN (N,T,X,P,X,P,X2,P2,XP,ER)
      CALL DATSW (7,JL)
      GO TO (891,886),JL
891   CALL GOAN (N,T,XJ,PJ,XC,PC,X2C,P2C,XPC,ERC)
      DO 888 I=1,N
      XC(I)=XC(I)-XI(I)
888   PC(I)=PC(I)-Q(I)
      WRITE (3,102) (XC(I),I=1,N)
      WRITE (3,102) (PC(I),I=1,N)
      DO 889 I=1,N
      X(I)=X(I)-XC(I)
889   PI(I)=P(I)-PC(I)
      WRITE (3,102) (X(I),I=1,N)
      WRITE (3,102) (PI(I),I=1,N)
      DO 890 I=1,N
890   PI(I)=PI(I)
      GO TO 887
886   DO 885 I=1,N
885   PI(I)=P(I)
887   CALL DIGO (20,-1)
      CALL GOAN (N,T,XI,PI,X,P,X2,P2,XP,ER)
      IF (ER-ERS) 42,41,41
41    XD=XD*.2
412   WRITE (3,102)XD
      GO TO 32
42    ERS=ER
      WRITE (3,102) ERS,ERS,ERS,ERS
      ITKM=0
      IF (ERS-TOL2) 5,5,12
5     DO 6 I=1,N
      Q(I)=PI(I)
6     XJ(I)=X(I)
      X2D=X2
1     DO 7 I=1,N
      PS(I,IZA)=Q(I)
7     XS(I,IZA)=XJ(I)
      XERS(IZA)=X2D
      IZA=IZA+1
      IF(IZA-11)8,997,997
```

```
997   CALL DATSW (11,IRX)
      GO TO (4,9), IRX
9     DO 999 J=1,10
999   WRITE (3,102)XERS(J),(XS(I,J),I=1,N),(PS(I,J),I=1,N)
      GO TO 4
99    CALL EXIT
      END
```

*As previously mentioned, several options or alternatives are provided as a result of the CALL DATSW and CALL SSWTCH subroutines. In addition, unnecessary printout routines are included.

2. DIGITAL INTEGRATION AND ERROR FUNCTION EVALUATION ROUTINE

```
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*NONPROCESS PROGRAM
      SUBROUTINE GOAN (N,T,XI,PI,X,P,X2,P2,XP,ER)
      EXTERNAL F,FO
      DIMENSION PI(4),X(4),P(4),XI(4),Y(9),DY(9),AUX(10,9),
           PRMT(5)
      COMMON ITRNK(8),OLD(9),XYZ(5)
      NAX=2*N
      DO 1 I=1,N
      Y(2*I-1)=XI(I)
1     Y(2*I)=PI(I)
      ITRNK(3)=0
      E=.5/N
      CALL DATSW(14,II)
      GO TO (591,592),II
591   HT=10.
      GO TO 593
592   HT=50.
593   CALL DATSW (15,JJ)
      GO TO (594,595),JJ
594   HT=HT*2.
      GO TO 596
595   HT=HT
596   IF(ITRNK(1)) 3,3,4
3     PRMT(1)=0
      TF=T
      XYZ(1)=TF
      PRMT(2)=T
      PRMT(3)=T/HT
      GO TO 5
4     PRMT (1)=T
      TF=0
      XYZ(1)=TF
      PRMT(2)=0
      PRMT(3)=-T/HT
```

```
5     DO 2 I=1,NAX
2     DY(I)=E
      ITRNK(2)=0
      CALL HPCG (PRMT,Y,DY,2*N,IHLF,F,FO,AUX)
      IF(PRMT(5)) 525,7,525
525   IF(ITRNK(3)) 7,528,7
528   DO 526 KJ=1,9
526   Y(KJ)=OLD(KJ)
      PRMT(1)=PRMT(1)-PRMT(3)
      PRMT(3)=PRMT(3)/10.
      PRMT (2)=PRMT(1)+10.*PRMT(3)
      DO 527 KJ=1,NAX
527   DY(KJ)=E
      ITRNK(2)=1
      CALL HPCG (PRMT,Y,DY,2*N,IHLF,F,FO,AUX)
      PRMT(1)=PRMT(2)
      PRMT3)=PRMT(3)*10.
      IF(ITRNK(3)) 7,77,7
77    PRMT(2)=TF
      GO TO 5
7     DO 6 I=1,N
      X(I)=Y(2*I-1)
6     P(I)=Y(2*I)
      P2=0.
      DO 30 I=1,N
30    P2=P2+P(I)*P(I)
      P2=SQRT(P2)
      DO 308 I=1,N
308   P(I)=P(I)*5./P2
      P2=5.
      CALL DATSW (10,IPX)
      GO TO (20,110),IPX
110   WRITE (3,112)T
112   FORMAT (F15.5)
111   FORMAT (' THE CURRENT ERROR IS ',F15.5,' NORM X = ',
      F15.5,'  XP=  ',F10.5)
      DO 18 L=1,N
18    WRITE (3,10) XI(L),PI(L),P(L),X(L)
10    FORMAT (6F20.5)
20    X2=0.0
      DO 31 I=1,N
31    X2=X2+X(I)*X(I)
      X2=SQRT(X2)
      XP=0.
      DO 32 I=1,N
32    XP=XP+X(I)*P(I)
      ER=(XP/P2)+X2
      XPN=ER/X2-1.
      WRITE (3,111) ER,X2,XPN
      CALL SSWTCH (0,I00)
      GO TO (38,40),I00
38    DO 39 JI=1,N
```

```
39    WRITE (1,10) XI(JI),PI(JI),P(JI),X(JI)
      WRITE (1,111) ER,X2,XPN
40    RETURN
      END
```

3. DIGITAL INTEGRATION FORWARD-REVERSE TIME SUBROUTINE*

```
// FOR
*LIST SOURCE PROGRAM
*NONPROCESS PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE DIGO (M,N)
      COMMON ITRNK(8),OLD(9)
      ITRNK (M-19)=N
      RETURN
      END
```

*This subroutine supplements the digital integration subroutine to replace a hybrid subroutine called in the main program.

4. HYBRID INTEGRATION AND ERROR FUNCTION EVALUATION ROUTINE

```
// FOR
*NONPROCESS PROGRAM
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE GOAN (N,T,XI,PI,X,P,X2,P2,XP,ER)
      DIMENSION PI(4),X(4),P(4),XI(4),JI(4),KI(4),KJ(4),JJ(4)
      CALL LOGEX(1)
      DO 5 J=1,N
4     JI(J)=XI(J)*3276.7
5     CALL ANOUT (31+J,JI(J))
      XPI=0.
      DO 555 I=1,N
555   XPI=XPI+PI(I)*PI(I)
      XPI=SQRT(XPI)
      DO 556 I=1,N
556   PI(I)=PI(I)*10./XPI
      DO 8 K=1,N
      JJ(K)=PI(K)*3276.7
8     CALL ANOUT (33+K,JJ(K))
      NT=T*40.+30.
      CALL LOAD
      CALL DELAY (8000)
      CALL RUN
      DO 742 I=1,10
742   CALL DELAY (NT)
      CALL STOPY
      CALL AINP (12,N,KI(1),KI(2))
      CALL AINP (14,N,KJ(1),KJ(2))
      DO 19 L=1,N
```

```
      X(L)=-KI(L)/327.67
19    P(L)=-KJ(L)/327.67
      CALL AINP (16,1,KL)
      TK=-KL/327.67
      P2=0.
      DO 30 I=1,N
30    P2=P2+P(I)*P(I)
      P2=SQRT(P2)
      DO 308 I=1,N
308   P(I)=P(I)*10./P2
      P2=10.
      CALL DATSW (10,IPX)
      GO TO (20,110),IPX
110   WRITE (3,112)TK
111   FORMAT (' THE CURRENT ERROR IS  ',F15.5)
112   FORMAT (F15.5)
      DO 18 L=1,N
18    WRITE (3,10) XI(L),PI(L),P(L),X(L)
10    FORMAT (6F20.5)
20    CALL LOAD
      X2=0.
      DO 31 I=1,N
31    X2=X2+X(I)*X(I)
      X2=SQRT(X2)
      XP=0.0
      DO 32 I=1,N
32    XP=XP+X(I)*P(I)
      ER=(XP/P2)+X2
      WRITE (3,111) ER
      RETURN
      END
```

5. DIRECT SEARCH SUBROUTINE

```
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*NONPROCESS PROGRAM
      SUBROUTINE EXPLR(N,T,XI,PI,X,P,X2,P2,XP,ER,HH)
      DIMENSION XI(4),PI(4),X(4),P(4)
      DO 260 I=1,N
      PI(I)=PI(I)-HH
      CALL GOAN(N,T,XI,PI,X,P,X2,P2,XP,ES)
      IF (ES-ER) 205,210,210
205   ET=ER
      ER=ES
      IF(.80-ER/ET)260,260,265
210   PI(I)=PI(I)+2.*HH
      CALL GOAN (N,T,XI,PI,X,P,X2,P2,XP,ES)
      IF(ES-ER) 215,220,220
215   ET=ER
      ER=ES
```

```
      IF(.95-ER/ET)260,260,265
220   PI(I)=PI(I)-HH
260   CONTINUE
265   WRITE (3,270) (PI(II),II=1,N)
270   FORMAT (9F10.5)
99    RETURN
      END
```

6. DIGITAL INTEGRATION OUTPUT AND CONTROL DISCONTINUITY SENSOR SUBROUTINE

```
// FOR
*NONPROCESS PROGRAM
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE FO(T,Y,DY,IHLF,NN,PRMT)
      DIMENSION Y(9),PRMT(5)
      COMMON ITRNK(8),OLD(9),XYZ(5)
      IF(T-PRMT(1)) 101,100,101
100   SIGN=Y(9)
      SIGNV=Y(8)
      GO TO 103
101   IF(ITRNK(2)) 111,111,103
111   IF(Y(9)*SIGN)104,113,113
113   IF(Y(8)*SIGNV(104,103,103)
104   PRMT(5)=1.
      PRMT(1)=T
103   CALL DATSW (13,KK)
      GO TO (1,2),KK
10    FORMAT(12F10.3)
1     WRITE (3,10) T,PRMT(4),(Y(I),I=1,NN,2),(Y(I),I=2,NN,2),
           Y(9),Y(8)
2     IF(PRMT(5))108,22,108
22    DO 109 I=1,9
109   OLD(I)=Y(I)
108   IF(ABS(XYZ(1)-T)-.0001)107,107,110
107   ITRNK(3)=1
      PRMT(5)=1.
110   RETURN
      END
```

7. RANDOM NUMBER GENERATOR SUBROUTINE

```
// FOR
*NONPROCESS PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE RANDU(IX,IY,Y)
      IY=IX*899
      IF(IY)5,6,6
5     IY=IY+32767+1
6     Y=IY
      Y=Y/32767
      RETURN
      END
```

8. BASIC DIGITAL INTEGRATION SUBROUTINE*

```
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE HPCG (PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
      DIMENSION PRMT(5),Y(200),DERY(200),AUX(10,200)
      IHLF=0
      X=PRMT(1)
      H=PRMT(3)
      PRMT(4)=0.
      PRMT(5)=0.
      DO 1 I=1,NDIM
      AUX(10,I)=0.
1     AUX(9,I)=DERY(I)
      IF (H*(PRMT(2)-X)) 3,2,5
2     IHLF=12
      GO TO 4
3     IHLF = 13
4     RETURN
5     DO 10 N=1,3
      CALL FCT (X,Y,DERY)
      IF (N-1) 11,11,12
11    CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
12    IF(PRMT(5)) 4,6,4
6     DO 9 I=1,NDIM
      AUX(N,I)=Y(I)
9     AUX(N+4,I)=DERY(I)
      DO 101 I=1,NDIM
101   Y(I)=AUX(N,I)+H*AUX(N+4,I)
      X=X+H
      CALL FCT (X,Y,DERY)
      DO 102 I=1,NDIM
102   Y(I)=AUX(N,I) +.5*H*(AUX(N+4,I)+DERY(I))
10    CONTINUE
21    N=1
      CALL FCT (X,Y,DERY)
      X=PRMT(1)
      DO 22 I=1,NDIM
      AUX(8,I)=DERY(I)
22    Y(I)=AUX(1,I)+H*(.375*AUX(5,I)+.791667*AUX(6,I)
           -.2083333*AUX(7,I)+.04166667*DERY(I))
23    X=X+H
      N=N+1
      CALL FCT(X,Y,DERY)
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5)) 4,24,4
24    IF(N-4)25,204,204
25    DO 26 I=1,NDIM
      AUX(N+4,I)=DERY(I)
      IF(N-3)27,29,204
27    DO 28 I=1,NDIM
      DELT=AUX(6,I)+AUX(6,I)
```

```
      DELT=DELT+DELT
28    Y(I)=AUX(1,I)+.3333333*H*(AUX(5,I)+DELT+AUX(7,I))
      GO TO 23
29    DO 30 I=1,NDIM
      DELT=AUX(6,I)+AUX(7,I)
      DELT =DELT+DELT+DELT
30    Y(I)=AUX(1,I)+.375*H*(AUX(5,I)+DELT+AUX(8,I))
      GO TO 23
200   DO 203 N=2,8
      DO 203 I=1,NDIM
203   AUX(N-1,I)=AUX(N,I)
204   DO 205 I=1,NDIM
      AUX(4,I)=Y(I)
205   AUX(8,I)=DERY(I)
      X=X+H
      DO 207 I=1,NDIM
      DELT=AUX(1,I)+1.333333*H*(AUX(8,I)+AUX(8,I)-AUX(7,I)
           +AUX(6,I)+AUX(6,I))
      Y(I)=DELT-.9256198*AUX(10,I)
207   AUX(10,I)=DELT
      CALL FCT(X,Y,DERY)
      DO 208 I=1,NDIM
      DELT=.125*(9.*AUX(4,I)-AUX(2,I)+3.*H*(DERY(I)+AUX(8,I)
           +AUX(8,I)-AUX(7,I)))
      AUX(10,I)=AUX(10,I)-DELT
208   Y(I)=DELT+.07438017*AUX(10,I)
      DELT=).
      DO 209 I=1,NDIM
209   DELT=DELT+AUX(9,I)*ABS(AUX(10,I))
      IF (PRMT(4)-DELT) 215,210,210
215   PRMT(4) = DELT
210   CALL FCT(X,Y,DERY)
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5)) 212,213,212
212   RETURN
213   IF (H*X-PRMT(2))) 214,212,212
214   IF (ABS(X-PRMT(2))-.1*ABS(H)) 212,200,200
      END
```

*This integration routine is part of the Scientific Subroutine Set for the IBM System/360.

9. FUNCTION EVALUATION SUBROUTINE FOR ES-2

```
// FOR
*LIST SOURCE PROGRAM
*NONPROCESS PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE F(T,Y,DY)
      DIMENSION Y(9),DY(8)
      IF(Y(6)) 2,1,2
1     U=1.
```

```
      GO TO 3
2     U=ABS(Y(6))/Y(6)
3     IF(Y(4)) 5,4,5
4     V=1.
      GO TO 6
5     V=ABS(Y(4))/Y(4)
6     Y(9)=U
      Y(8)=V
      DY(1)=Y(3)
      DY(3)=Y(5)+V
      DY(5)=-Y(5)*Y(1)+Y(3)*Y(3)+U
      DY(2)=Y(5)*Y(6)
      DY(4)=-Y(2)-2.*Y(3)*Y(6)
      DY(6)=-Y(4)+Y(1)*Y(6)
      RETURN
      END
```

APPENDIX C

INPUT DATA SETS

1. Input Data Set 1

Number	$(x_o)^T$	$(p_o)^T$	Number	$(x_o)^T$	$(p_o)^T$
1	(-10, -5)	(0, 5)	9	(-5,-5)	(5,-5)
2	(2,-10)	(-5, -3)	10	(-10,-5)	(5,-1)
3	(2, 6)	(5, 4)	11	(5, 5)	(5,-1)
4	(6, 2)	(10, -3)	12	(-5, 5)	(5, 5)
5	(-7, -2)	(-2, -2)	13	(-10,-5)	(5, 3)
6	(-1, 9)	(3, -4)	14	(5, 5)	(-3,-1)
7	(5, 0)	(5, -5)	15	(5,-5)	(5, 5)
8	(-5, 0)	(-3, 4)			

2. Input Data Set 2: $x_o = (-10,-5)^T$, p_o is randomly generated and normalized to 10.

Number	$(p_o)^T$	Number	$(p_o)^T$
1	(9.879, 1.551)	6	(-4.919, 8.706)
2	(9.997,-0.254)	7	(9.823,-1.871)
3	(3.671,-9.302)	8	(9.264,-3.766)
4	(-6.959,-7.181)	9	(9.970,-0.769)
5	(-8.171, 5.764)	10	(-0.235,-9.997)

3. Input Data Set 3

Number	$(x_o)^T$	$(p_o)^T$
1	(-3,-2,-1)	(1,-2,-3)
2	(2,-1, 1)	(-1, 0, 0)
3	(2,-2, 2)	(1, 1,-1)
4	(2,-2, 0)	(1,-1,-5)
5	(2,-2,-2)	(-2,-2, 4)
6	(2, 0, 2)	(-5,-4,-2)
7	(2, 0, 0)	(4,-3, 5)
8	(2, 0,-2)	(7,-7,-7)
9	(2, 2, 2)	(-3, 1, 2)
10	(2, 2, 0)	(5, 2, 5)

4. Input Data Set 4: $x_o = (-3,-2,-1)^T$, p_o is randomly generated:

Number	$(p_o)^T$	Number	$(p_o)^T$
1	(.2514, .0394, .4738)	4	(.1732,-.2497, .4420)
2	(-.0120, .1630,-.4131)	5	(.4003,-.0762, .4392)
3	(-.4465,-.4607,-.2456)	6	(-.1785, .4760,-.0366)

APPENDIX D

POSSIBLE EXTENSIONS AND FUTURE INVESTIGATIONS

1. Further examination of time optimal control problems.

2. Extension to other optimal control problems such as minimum fuel, convex function minimum error regulator, etc.

3. Determination of computational results for the special cases given in Chapter 4.

4. Another approach to the solution of MP based on LOP, other than simply the generation of random initial adjoints.

5. Development of higher order systems with nonconvex reachable sets and many local optima.

6. Application to parameter optimization problems.

7. Comparison of GOP and LOP with other nonlinear methods.

8. Further investigations of the nature of the reachable sets as they relate to the nonlinear problem involved:

   a. Classification of reachable sets in some manner.

   b. Transformation of reachable sets as a result of the transformation of the origin.

9. Effect of singularities and controllability on the general problems considered in this thesis.

10. Relationship of the path composed of lines of curvature to the path determined by gradient methods.

11. Use of geodesics instead of lines of curvature as the means of defining the path on the boundary of the reachable set.