Differential Dynamic Programming: An Optimization Technique for Nonlinear Systems

Differential Dynamic Programming:
An Optimization Technique
for Nonlinear Systems

By

Nobuyuki Sato, B.Eng.

A Thesis
Submitted to the Faculty of Graduate Studies
in Partial Fulfilment of the Requirements
for the Degree
Master of Engineering

McMaster University
April 1969

MASTER OF ENGINEERING (Electrical Engineering)

McMASTER UNIVERSITY
Hamilton, Ontario

TITLE: Differential Dynamic Programming: An Optimization Technique for Nonlinear Systems

AUTHOR: Nobuyuki Sato, B.Eng.
(Nagoya Institute of Technology)

SUPERVISOR: Dr. N. K. Sinha

NUMBER OF PAGES: V, 64

SCOPE AND CONTENTS:

Differential Dynamic Programming is a new method, based on Bellman's principle of optimality, for determining optimal control strategies for nonlinear systems. It has originally been developed by D.H.Jacobson.

In this thesis a result is presented for a problem with saturation characteristics in nonlinearity solved by the Jacobson's approach. In the differential dynamic programming the principle of optimality is applied to the differential change in non-optimal cost due to small changes in state variables instead of the cost itself. This results in modest memory requirements for its defining parameters and rapid convergence.

ACKNOWLEDGEMENTS

The author wishes to acknowledge with sincere gratitude the invaluable guidance and encouragement of Dr. N. K. Sinha in preparation of this thesis.

The author is also thankful to the Department of Electrical Engineering for the scholarship and teaching assistantship.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE PAGE

CHAPTER I

Introduction

1-1: Historical Background

There have been many approaches to optimization, which involves finding the best solution among several feasible alternatives. The term "best solution" is used because there may be more than one optimal solution.

Having constructed an appropriate mathematical model for the problem concerned, we must choose an optimization technique to solve the problem. The way we determine an optimal solution depends, of course, on the form of the objective function and constraints, the nature and number of variables, the kind of computational facilities available, taste and experience.

In 1940's there was a reawakening and change of direction in the study of optimization theory. This renaissance was stimulated by the war effort. Two parallel but interrelated occurrences are especially significant: the work of scientists and mathematicians on problems in operations research generated by modern warfare; and the invention and development of the elctronic digital computer.

In 1950's following studies in the area of multistage decision processes, Bellman[1,2,3] invented the rather undescriptive but alluring name for the approach —— Dynamic Programming. A more representative but less glamourous name would be "recursive optimization". Dynamic Programming appears as having been practiced long before it was named. Undoubtedly, however, R.Bellman is the father of Dynamic Programming. It has been applied to problems in numerous fields, e.g. theory of inventory and production, [7,22] purchasing and investment problems, distribution of drugs in the body, satellite interception and rendezvous problems, [21] design of chemical plants, statistical communication theory and control systems. [16,18,19] Bellman's own output has been prolific. The relevance of Dynamic Programming to control theory is unquestionable but there are limitations to its practical applications such as enormous computing time and storage requirements.

Several attempts have been made to overcome the limitations, and Differential Dynamic Programming is one of them.

1-2: Statement of the problem in a general form

In recent years much interest has centred on the problem of determining optimal control for dynamic systems described by nonlinear, ordinary differential equations of the following form:

$$\dot{\underline{x}} = f(\underline{x}, \underline{u}; t) ; \quad \underline{x}(t_0) = \underline{x}_0 \qquad (1\text{-}1)$$

The criterion of optimality is to minimize the cost functional:

$$V(\underline{x}_0 ; t_0) = \int_{t_0}^{t_f} L(\underline{x}, \underline{u}; t)dt + F(\underline{x}(t_f)) \qquad (1\text{-}2)$$

where $\underline{x}(t)$ is an n-dimensional vector function of time describing the state of the dynamic system at any time $t \in [t_0 , t_f]$, (The symbol $\underline{x}$ is also taken to mean $\underline{x}(t)$.); $\underline{u}(t)$ —— or $\underline{u}$ —— is an m-dimensional vector function of time representing the control variables available for manipulation at any time $t \in [t_0 , t_f]$; L and F are nonlinear functions of their arguments; and f is an n-dimensional vector of its arguments. (1-1) describes the dynamic structure of the system. The final time t_f is assumed to be given explicitly.

$f(\underline{x},\underline{u};t)$ is a function of $\underline{x}$, $\underline{u}$ and time t. At a particular time $t_1 \in [t_0 , t_f]$, $f(\underline{x},\underline{u};t_1)$ is a function of $\underline{x}$ and $\underline{u}$. The semi-colon is used to separate t from the other arguments. Similar remarks apply to V, F and L.

All functions are assumed to be continuously differentiable in each argument up to any order required.

The purpose of the problem is to find $\underline{u}(t)$; $t \in [t_0 , t_f]$

such that V, given by (1-2) is minimized.

Many second-variation type algorithms have been developed for successively improving a nominal, guessed control function. A very restrictive requirement of those second-order methods is that H_{uu}^{-1} -- the inverse matrix of the second partial derivatives of the Hamiltonian (see Eq.(3-17)) with respect to u -- should exist and be positive definite along non-optimal trajectories for $t \in [t_0, t_f]$. This restriction is very severe since it implies that H must be strictly convex, globally, with respect to u. In many non-linear problems one finds that H is strictly convex only in the neighbourhood of its minimum with respect to u.

The differential dynamic programming requires only that H_{uu}^{-1} exists and is positive definite in the neighbourhood of the minimum of H with respect to u. This is a much less severe requirement, therefore we may use the Fletcher and Powell routine[8] (see Appendix I) to minimize H with respect to u. The function and its first derivatives are calculated by this procedure and the inverse of the matrix of second derivatives is estimated from this data as the procedure progresses towards the minimum of the function. Thus when the minimum is reached, an estimate of the inverse of the matrix of second derivatives evaluated at the minimum is available. Since the procedure does not evaluate the matrix of second derivatives directly it is not necessary that its inverse be positive definite.

CHAPTER II

Dynamic Programming

Often before performing the optimization, it is desirable to make some changes of variables and transformations. In contrast to simplifying the model, these preparatory operations preserve the properties of the model completely. The transformed model has the same optimal solution as the original, but is of a form that can be optimized more easily.

Basically Dynamic Programming is such a transformation. It takes a sequential or multistage decision process containing many interdependent variables, and converts it into a series of single stage problems, each containing only a few variables. The transformation is invariant in that the number of feasible solutions, and the value of the objective function associated with each feasible solution is preserved.

The transformation is based on the principle of optimality, due to Bellman, which states that
" An optimal set of decisions has the property that whatever the first decision is, the remaining decisions must be optimal with respect to the outcome which results from the first decision."

Treating the control problem as a multistage decision process we may use Dynamic Programming to write

$$V(\underline{x};t) = \min_{t\leq\tau\leq t+\Delta}\left[\int_t^{t+\Delta} L(\underline{x},\underline{u};\tau)d\tau + V(\underline{x}+\underline{\Delta x};t+\Delta)\right] \qquad (2\text{-}1)$$

where $V(\underline{x};t)$ is the optimal cost to go from state $\underline{x}$ at time t. If we let Δ be a small interval of time then

$$V(\underline{x};t) = \min_{u(t)}\left[L\cdot\Delta + V(\underline{x}+\dot{\underline{x}}\Delta;t+\Delta)\right] \qquad (2\text{-}2)$$

Now we make the fundamental assumption that $V(\underline{x};t)$ may be expanded in a Taylor series about the point $\underline{x};t$,

$$V(\underline{x}+\underline{\Delta x};t+\Delta) = V(\underline{x};t) + \left\langle \frac{\partial V}{\partial x}, \Delta x\right\rangle + \frac{\partial V}{\partial t}\cdot\Delta$$

+ higher-order terms,

$$= V(\underline{x};t) + \langle V_x, f\rangle\Delta + \frac{\partial V}{\partial t}\cdot\Delta + (o)\Delta^2 \qquad (2\text{-}3)$$

where $V_x = \frac{\partial V}{\partial x}$, and $\langle V_x, f\rangle$ is the scalar product of the vectors V_x and f. Then we obtain from (2-2) and (2-3),

$$V(\underline{x};t) = \min_u\Big[L\cdot\Delta + V(\underline{x};t) + \langle V_x,f\rangle\cdot\Delta$$

$$+ \frac{\partial V}{\partial t}\cdot\Delta + (o)\cdot\Delta^2\Big]. \qquad (2\text{-}4)$$

Since $V(\underline{x};t)$ and $\frac{\partial V}{\partial t}$ are independent of u, then

$$-\frac{\partial V}{\partial t}\cdot\Delta = \min_{u}\left[L\cdot\Delta + \langle V_x, f\rangle\Delta + (0)\Delta^2\right] \qquad (2\text{-}5)$$

Dividing by Δ and allowing $\Delta \to 0$, we obtain

$$-\frac{\partial V(\underline{x};t)}{\partial t} = \min_{u}\left[L(x,\underline{u};t) + \langle V_x(\underline{x};t), f(x,\underline{u};t)\rangle\right] \qquad (2\text{-}6)$$

Equation (2-6) is the Bellman's partial differential equation in n+1 variables $\underline{x};t$ and is in general unsolvable analytically.

The difficulty of numerically solving (2-6) is in general, enormous primarily due to the high dimensionality of the equation which means that storage and computing time requirements are tremendous. To avoid this difficulty several attempts have been made and Differential Dynamic Programming is one of them.

CHAPTER III

Differential Dynamic Programming

3-1: The Basis of Differential Dynamic Programming

Differential Dynamic Programming is a technique, based on Dynamic Programming, for determining the optimal control function of a nonlinear system. Unlike conventional Dynamic Programming where the optimal cost function is considered globally, Differential Dynamic Programming applies the principle of optimality in the neighbourhood of a nominal, possibly non-optimal, trajectory.

Mayne [15] introduced the notation of "Differential Dynamic Programming" and Jacobson [10,11,12] developed it for his Ph.D. thesis.

It is well known that the optimal cost $V^\circ(x;t)$ satisfies the Bellman's partial differential equation(2-6) as shown in the previous chapter. We rewrite it for convenience,

$$-\frac{\partial V^\circ(\underline{x};t)}{\partial t} = \min_{u}\left[L(\underline{x},\underline{u};t) + \left\langle V_x^\circ(\underline{x};t), f(\underline{x},\underline{u};t)\right\rangle\right] \qquad (3\text{-}1)$$

in n+1 variables $\underline{x};t$ for the optimal cost V°. (In the above equation, one should realize that $\underline{x}$ is an optimal quantity and so should be written with superscript °. However, for simplicity, the superscript ° appears on 'V quantity' only.)

It is assumed that $V^{\circ}(\underline{x};t)$ is sufficiently smooth in $\underline{x}$ and t to allow the derivation of (3-1) which requires that the second partial derivatives of V° with respect to $\underline{x};t$ exist. [6]

In the problem of Section 1-2, we assume that the optimal control $u^{\circ}(t)$; $t \in [t_0, t_f]$ is unknown but that a nominal control $\bar{u}(t)$; $t \in [t_0, t_f]$ is available.

On application of the nominal control, a nominal state trajectory $\bar{x}(t)$; $t \in [t_0, t_f]$ is produced by (1-1), i.e.

$$\dot{\bar{x}} = f(\bar{x}, \bar{u}; t) \quad ; \quad \bar{x}(t_0) = x(t_0) = x_0 \qquad (3\text{-}2)$$

The nominal cost $\bar{V}(x_0;t_0)$ is calculated using (1-2), i.e.

$$\bar{V}(x_0; t_0) = \int_{t_0}^{t_f} L(\bar{x}, \bar{u}; t)\,dt + F(x(t_f)) \qquad (3\text{-}3)$$

Equations (1-1),(1-2), and (3-1) may be written in terms of the nominal trajectory by setting:

$$\begin{aligned} x &= \bar{x} + \delta x \\ u &= \bar{u} + \delta u \end{aligned} \qquad (3\text{-}4)$$

δx and δu are variations in the state and control vectors, respectively, measured with respect to the nominal quantities $\bar{x}$, $\bar{u}$; t they are not necessarily small quantities at this stage.

Equations (1-1), (1-2), and (3-1) become:

$$\frac{d(\bar{x}+\delta x)}{dt} = f(\bar{x}+\delta x,\ \bar{u}+\delta u\ ;\ t)\ ; \tag{3-5}$$

$$\bar{x}(t_0) + \delta x(t_0) = x_0 \ (\because \text{Set} \ \ \delta x(t_0) = 0)$$

$$V(x_0\ ;\ t_0) = \int_{t_0}^{t_f} L(\bar{x}+\delta x,\ \bar{u}+\delta u\ ;\ t)\,dt + F(\bar{x}(t_f) + \delta x(t_f)\ ;\ t_f) \tag{3-6}$$

$$-\frac{\partial V^{\circ}(\bar{x}+\delta x;t)}{\partial t} = \min_{\delta u}\Big[\ L(\bar{x}+\delta x,\ \bar{u}+\delta u\ ;\ t) + \big\langle V^{\circ}_{x}(\bar{x}+\delta x;t), f(\bar{x}+\delta x, \bar{u}+\delta u\ ;\ t)\big\rangle\Big] \tag{3-7}$$

respectively.

Equations (3-5),(3-6), and (3-7) are exactly equivalent to (1-1),(1-2), and (3-1) since no approximations have been introduced: the nominal trajectory has been made into a reference trajectory.

Now let us assume that the optimal cost is smooth enough to allow for a power series expansion in δx about $\bar{x}$.

$$V^{\circ}(\bar{x}+\delta x;t) = V^{\circ}(\bar{x};t) + \langle V^{\circ}_{x}, \delta x \rangle + \tfrac{1}{2}\langle \delta x, V^{\circ}_{xx}\,\delta x \rangle + \text{higher-order terms} \tag{3-8}$$

The optimal cost

$$V^{\circ}(\bar{x};t) = \bar{V}(\bar{x};t) + a^{\circ}(\bar{x};t) \tag{3-9}$$

where $a^{\circ}(\bar{x};t)$ is defined as the difference between the optimal cost $V^{\circ}(\bar{x};t)$ obtained by using the optimal controls

$$u^{\circ}(\tau) = \bar{u}(\tau) + \delta u(\tau) \; ; \quad \tau \in [t, t_f]$$

and the nominal cost $\bar{V}(\bar{x};t)$ obtained by using the nominal controls

$$\bar{u}(\tau) \; ; \quad \tau \in [t, t_f].$$

Using (3-9), Equation (3-8) becomes:

$$V^{\circ}(\bar{x}+\delta x;t) = \bar{V}(\bar{x};t) + a^{\circ} + \langle V^{\circ}_{x}, \delta x \rangle + \tfrac{1}{2}\langle \delta x, V^{\circ}_{xx}\,\delta x \rangle + \text{higher-order terms in } \delta x. \tag{3-10}$$

Substituting (3-10) into (3-7), we obtain:

$$-\frac{\partial\bar{V}}{\partial t}-\frac{\partial a^{\circ}}{\partial t}-\left\langle\frac{\partial V^{\circ}_{x}}{\partial t},\delta x\right\rangle-\frac{1}{2}\left\langle\delta x,\frac{\partial V^{\circ}_{xx}}{\partial t}\delta x\right\rangle$$

$$+\text{ higher-order terms}$$

$$=\min_{\delta u}\Big[L(\bar{x}+\delta x,\bar{u}+\delta u;t) \tag{3-11}$$

$$+\Big\langle V^{\circ}_{x}+V^{\circ}_{xx}\delta x+\text{ higher-order terms},$$

$$f(\bar{x}+\delta x,\bar{u}+\delta u;t)\Big\rangle\Big]$$

Equation (3-11) is very difficult to solve owing to the very large computing time, and storage requirements for the parameters of the power series expansion. However, we can truncate the power series provided that the truncated terms are negligible. In order to do this, the size of δx must, somehow, be restricted. That is, the trajectory $\bar{x}(t)+\delta x(t)$; $t\in[t_{o},t_{f}]$ must be kept in the neighbourhood of the nominal trajectory.

Let us assume that we keep the δx's small. (Methods ensuring this are discussed later, in Section 3-3. It is sufficient at the moment to note that, because $\delta x(t_{o})=0$, the δx's produced in the interval $[t_{o},t_{f}]$ are caused only by δu acting through Equation (3-5).) Assume, further, that the δx's produced are small enough such that an expansion up to quadratic terms, in δx, is sufficient to represent V° in the neighbourhood of the nominal trajectory. Equation (3-11) becomes:

$$\begin{aligned} -\frac{\partial \bar{V}}{\partial t} - \frac{\partial a}{\partial t} - \left\langle \frac{\partial V_x}{\partial t}, \delta x \right\rangle - \frac{1}{2}\left\langle \delta x, \frac{\partial V_{xx}}{\partial t}\,\delta x \right\rangle & \\ = \min_{\delta u}\Big[L(\bar{x}+\delta x, \bar{u}+\delta u; t) & \qquad (3\text{-}12) \\ + \left\langle V_x + V_{xx}\delta x, f(\bar{x}+\delta x, \bar{u}+\delta u; t)\right\rangle\Big] & \end{aligned}$$

where

$$\begin{aligned} V(\bar{x}+\delta x; t) &= \bar{V}(\bar{x}; t) + a + \langle V_x, \delta x \rangle \\ &\quad + \frac{1}{2}\langle \delta x, V_{xx}\,\delta x \rangle \end{aligned}$$

(3-13)

and

$$V_x(\bar{x}+\delta x; t) = V_x(\bar{x}; t) + V_{xx}\,\delta x$$

One should realize that expanding up to second-order in δx makes, on differentiating, an expression for $V_x(\bar{x}+\delta x; t)$ accurate only to first-order in δx.

In subsequent sections, an algorithm is considered 'second-order' if, given the a priori expansion (3-13) for V and V_x, all second-order terms arising on the right hand side of (3-12) during the derivations, are accounted for.

The superscript ° on V in (3-12) and (3-13) has been dropped for the following reason:

Modelling the cost surface, locally, by a second-order expansion is made possible by keeping δx small. So the cost described by the truncated series (3-13) is optimal, provided that the δu's are chosen in such a way that δx's remain small. It is, therefore, not the truly optimal cost given that any size of δx is allowed.

V, given by (3-13), is the optimal cost V^o when starting in the state $\bar{x}+\delta x$ at time t if the nominal trajectory is sufficiently close to the optimal one, i.e. if $\bar{u}(\tau)$ is close to $u^o(\tau); \tau \in [t, t_f]$ then the minimizing δu's will be small and, from (3-5), the resulting δx's will be small so that the size of the δx's need not be restrained artificially. The expansion of V up to second-order will thus be adequate to describe the true optimal cost V^o in the neighbourhood of the nominal trajectory. V^o_x will also be described adequately.

Equation (3-12) can be used to develop methods for determining the optimal control $u^o(t); t \in [t_o, t_f]$ by successively improving the current nominal trajectory $\bar{u}(t); t \in [t_o, t_f]$.

3-2: A Second-order Algorithm

To overcome the restriction that H^{-1}_{uu} should exist and be positive definite along non-optimal trajectories for $t \in [t_o, t_f]$, $H(\bar{x}, u, V_x; t)$ may be actually minimized (see eq.(3-17)) with respect to u using the Fletcher and Powell method.

This u, obtained by minimizing H, is denoted by u*. All quantities, including $H_{uu}^{-1}(\bar{x},u,V_x;t)$, are evaluated at u*. Because u* minimizes H, then the Fletcher and Powell routine requires only that $H_{uu}^{-1}(\bar{x},u^*,V_x;t)$ be positive definite. It is less restrictive than requiring H to be globally, strictly convex in u. Further, $H_u(\bar{x},u^*,V_x;t)=0$, which is a well known condition of optimality.

Variations δx about $\bar{x}$ are then introduced and a linear relationship found between δu and δx, which maintains the necessary condition of optimality,

$$H_u(\bar{x}+\delta x,\ u^*+\delta u,\ V_x+V_{xx}\delta x;t) = 0$$

for δx sufficiently small.

With the above two points in mind, the derivation of the second-order algorithm will be understood readily.

The following is the full derivation of the algorithm:

At any time $t\in[t_o,t_f]$, Equation (3-12) is valid locally with respect to δx, but globally with respect to δu. Consider Equation (3-12) at time t with $\delta x(t)$ set equal to zero; its right-hand-side becomes:

$$\min_{\delta u}\left[L(\bar{x},\bar{u}+\delta u;t) + \langle V_x, f(\bar{x},\bar{u}+\delta u;t)\rangle\right] \qquad (3\text{-}14)$$

Instead of using a second-order prediction of the minimizing δu, let us completely minimize the contents of the square brackets in Expression(3-14) with respect to δu; this may be done analytically or, if necessary, numerically.

Let us denote the minimizing control

$$u^* = \bar{u} + \delta u$$

Expression(3-14) becomes

$$L(\bar{x}, u^*; t) + \langle V_x, f(\bar{x}, u^*; t)\rangle \tag{3-15}$$

Now we consider variations δx about $\bar{x}$, i.e. reintroduce δx.

In order to keep the minimality of the right-hand-side of (3-12) the $\overset{\min}{\delta u}$ must be reintroduced; it should be noted that δu is now measured with reference to u^*.

$$\begin{aligned} -\frac{\partial \bar{V}}{\partial t} - \frac{\partial a}{\partial t} - \left\langle \frac{\partial V_x}{\partial t}, \delta x\right\rangle - \frac{1}{2}\left\langle \delta x, \frac{\partial V_{xx}}{\partial t}\delta x\right\rangle \\ = \overset{\min}{\delta u}\left[L(\bar{x}+\delta x, u^*+\delta u; t) + \langle V_x + V_{xx}\delta x, f(\bar{x}+\delta x, u^*+\delta u; t)\rangle\right] \end{aligned} \tag{3-16}$$

Of course, by allowing these large (global) changes in control, large δx's will be introduced through Equation(3-5). The δx's must, in some way, be restrained in size in order that the second-order expansion for V be valid. This will be discussed later.

Define:

$$H(x, u, V_x; t) = L(x, u; t) + \langle V_x, f(x, u; t)\rangle \qquad (3\text{-}17)$$

The right-hand-side of (3-16) becomes:

$$\min_{\delta u} H(\bar{x} + \delta x, u^* + \delta u, V_x + V_{xx}\delta x; t) \qquad (3\text{-}18)$$

Since u* minimizes $H(\bar{x}, u, V_x; t)$, the following necessary condition holds:

$$H_u(\bar{x}, u^*, V_x; t) = 0 \qquad (3\text{-}19)$$

Expanding (3-18) about $\bar{x}$, u* the following expression is obtained:

$$\begin{aligned}\min_{\delta u}\Big[H &+ \langle H_u, \delta u\rangle + \langle H_x, \delta x\rangle + \langle V_{xx} f, \delta x\rangle \\ &+ \langle \delta u, (H_{ux} + f_u^T V_{xx})\delta x\rangle + \tfrac{1}{2}\langle \delta u, H_{uu}\delta u\rangle \qquad (3\text{-}20) \\ &+ \tfrac{1}{2}\langle \delta x, (H_{xx} + f_x^T V_{xx} + V_{xx} f_x)\delta x\rangle + \text{(higher-order terms)}\Big]\end{aligned}$$

All quantities in (3-20) are evaluated at $\bar{x}$,u*;t.

From Equation (3-19), $H_u = 0$, so the terms involving δu in (3-20) are:

$$\langle \delta u, (H_{ux} + f_u^T V_{xx})\delta x \rangle + \frac{1}{2}\langle \delta u, H_{uu}\delta u \rangle$$
$$+(\text{higher-order terms}) \qquad (3\text{-}21)$$

If δu is of the same order as δx then these terms are quadratic in δx. There is, therefore, no point in finding a relationship between δu and δx which is of order higher than linear, since terms higher than second-order in δx are neglected. (Because the left-hand-side of (3-16) is expanded only to second-order in δx.) A relationship of the following form is, therefore, required:

$$\delta u = \beta \delta x \qquad (3\text{-}22)$$

where β is chosen to minimize the contents of the square brackets in Expression (3-20).

A necessary condition for minimality is obtained by differentiating (3-20) with respect to δu and equating to zero.

$$H_u + H_{uu}\delta u + (H_{ux} + f_u^T V_{xx})\delta x$$
$$+ (\text{higher-order terms}) = 0 \qquad (3\text{-}23)$$

Substituting (3-22) into (3-23):

$$H_u + H_{uu}\beta\,\delta x + (H_{ux} + f_u^T V_{xx})\,\delta x$$
$$+ \text{(higher-order terms)} = 0 \qquad (3\text{-}24)$$

From Equation (3-19), $H_u = 0$. Keeping δx sufficiently small in (3-24), coefficients of the first-order terms may be equated to zero, then

$$\beta = -H_{uu}^{-1}(H_{ux} + f_u^T V_{xx}) \qquad (3\text{-}25)$$

Quantities in (3-25) are evaluated at $x, u^*; t$. This β is the optimal linear feedback controller which maintains the necessary condition of optimality

$$H_u(\bar{x} + \delta x,\ u^* + \delta u,\ V_x + V_{xx}\,\delta x;\ t) = 0$$

for δx sufficiently small.

Substituting (3-22) into (3-20) and neglecting terms of order higher than the second, we obtain:

$$H + \langle H_x + V_{xx} f + \beta^T H_u,\ \delta x \rangle$$
$$+ \tfrac{1}{2}\langle \delta x, (H_{xx} + f_x^T V_{xx} + V_{xx} f_x - \beta^T H_{uu}\beta)\,\delta x \rangle \qquad (3\text{-}26)$$

Expression (3-26) equals to the left-hand-side of Equation

(3-16). Since equality holds for arbitrary δx we can equate coefficients:

$$\begin{aligned} -\frac{\partial \bar{V}}{\partial t} - \frac{\partial a}{\partial t} &= H \\ -\frac{\partial V_x}{\partial t} &= H_x + V_{xx} f + \beta^T H_u \\ -\frac{\partial V_{xx}}{\partial t} &= H_{xx} + f_x^T V_{xx} + V_{xx} f_x \\ &\qquad - (H_{ux} + f_u^T V_{xx})^T H_{uu}^{-1} (H_{ux} + f_u^T V_{xx}) \end{aligned} \tag{3-27}$$

where all quantities are evaluated at $\bar{x}, u^*; t$.

$\bar{V}$, a, V_x and V_{xx} are all functions of x and t along the nominal $\bar{x}$ trajectory, thus

$$\frac{d}{dt}(\bar{V}(\bar{x};t) + a(\bar{x};t)) = \frac{\partial(\bar{V}+a)}{\partial t} + \langle V_x, f(\bar{x},\bar{u};t)\rangle$$

also

$$\begin{aligned} \dot{V}_x &= \frac{\partial V_x}{\partial t} + V_{xx} f(\bar{x},\bar{u};t) \\ \dot{V}_{xx} &= \frac{\partial V_{xx}}{\partial t} \end{aligned} \tag{3-28}$$

since higher-order terms of V have been truncated. Substituting Equations (3-28) into (3-27) and noting that

$$-\dot{\bar{V}}(\bar{x};t) = L(\bar{x},\bar{u};t) :$$

$$
\begin{aligned}
-\dot{a} &= H - H(\bar{x}, \bar{u}, V_x; t) \\
-\dot{V}_x &= H_x + V_{xx}(f - f(\bar{x}, \bar{u}; t)) + \beta^T H_u^{=0} \\
-\dot{V}_{xx} &= H_{xx} + f_x^T V_{xx} + V_{xx} f_x \\
&\quad - (H_{ux} + f_u^T V_{xx})^T H_{uu}^{-1} (H_{ux} + f_u^T V_{xx})
\end{aligned}
\qquad (3\text{-}29)
$$

Unless otherwise specified, all quantities are evaluated at $\bar{x}, u^*; t$. ($H_u^{=0}$ indicates that $H_u = 0$.)

At $t = t_f$, $V(\bar{x}; t_f) = F(\bar{x}(t_f); t_f)$

whence

$$
\begin{aligned}
a(t_f) &= 0 \\
V_x(t_f) &= F_x(\bar{x}(t_f); t_f) \\
V_{xx}(t_f) &= F_{xx}(\bar{x}(t_f); t_f)
\end{aligned}
\qquad (3\text{-}30)
$$

Equations (3-30) are boundary conditions for the differential equations (3-29).

The new control applied to the system is

$$
u = \bar{u} + \delta u^* + \beta \delta x = u^* + \beta \delta x. \qquad (3\text{-}31)
$$

The above theory assumes that the δx's generated by (3-31) will be sufficiently small to justify the second-order expansions used earlier.

3-3: Step Size Adjustment Method

Substituting (3-31) into (3-5), we obtain the following equation:

$$\frac{d(\bar{x}+\delta x)}{dt} = f(\bar{x}+\delta x,\ u^{*}+\beta\delta x\,;\,t) \qquad (3\text{-}32)$$

with $\bar{x}(t_0) + \delta x(t_0) = x_0$.

Since $\delta x(t_0) = 0$, the δx's produced by (3-32) are due to the driving action of $\delta u^{*} = u^{*} - \bar{u}$.

A way in which the size of the δx's can be restrained is by adjusting the time interval over which Equation (3-32) is integrated.

Consider the time interval $[t_1, t_f]$ where $t_0 \leq t_1 < t_f$. Assume that one runs along the nominal trajectory $\bar{x}$ from t_0 to t_1. At time $t = t_1$, $x(t_1) = \bar{x}(t_1)$ since the path of the nominal trajectory has been followed from t_0 to t_1. (i.e. $\delta x(t)$; $t \in [t_0, t_1]$ is zero). Now let us consider integrating (3-32) over the time interval $[t_1, t_f]$. If $t_1 < t_f$ and $[t_1, t_f]$ is small, then the δx's produced by (3-32) in this interval will be small, even for large δu^{*}, since there is very little time over which to integrate the differential equation:

$$\frac{d(\bar{x}+\delta x)}{dt} = f(\bar{x}+\delta x,\ u^{*}+\beta\delta x\,;\,t) \qquad (3\text{-}33)$$

$$x(t_1) + \delta x(t_1) = \bar{x}(t_1)$$

Thus, by making t_1 near t_f one can force the δx's to be as small as one pleases.

The above description is summarized in the following statement:

There exists a time t_1, sufficiently close to t_f, in the range $t_o \leq t_1 < t_f$, such that if the nominal trajectory is followed from t_o to t_1 and then (3-33) is integrated from t_1 to t_f, the δx's produced by (3-33) in the interval $[t_1, t_f]$ will be small enough for the second-order expansions of V, L and f to be valid.

The following questions may arise:

1) How does one decide if the δx's are sufficiently small?

2) How does one choose a t_1 such that δx's produced are small enough?

The answers are as follows:

1)

$$|a(\bar{x}; t_1)| = \left| \int_{t_f}^{t_1} [H - H(\bar{x}, \bar{u}, V_x; t)]\, dt \right|$$

is the predicted improvement in cost when starting at the point $\bar{x}(t_1); t_1$ and using

$$u(\tau) = u^*(\tau) + \beta(\tau)\delta x(\tau); \quad \tau \in [t_1, t_f].$$

Let us assume for the moment that $t_1 = t_o$. (i.e.

consider the whole time interval $[t_0, t_f]$.) Integrating (3-33) and calculating the cost V, the actual improvement in cost is

$$\Delta V = \bar{V}(\bar{x};t_0) - V(\bar{x};t_0) \qquad (3\text{-}34)$$

If this actual improvement in cost is 'near' the predicted value

$$|a(\bar{x};t_1)|$$

then the δx's produced by the new control, acting through Equation (3-33), are considered 'small enough'.

It may be convenient, in practice, to define 'near' in the following way:

If the following inequality is satisfied, ΔV is considered to be 'near' $|a(\bar{x};t_1)|$

$$\frac{\Delta V}{|a(\bar{x};t_1)|} > C \quad ; \quad C \geq 0 \qquad (3\text{-}35)$$

C is practically set as 0.5. There are no hard and fast rules for setting C. Certainly it should be equal to or greater than zero since a negative ΔV is inadmissible. C should not be greater than unity since one cannot expect improvements in cost greater than predicted, if the expansions for V, L and f are valid. Moreover, C should be somewhat less than unity so that any decision based on (3-35) is not influenced by round off errors in the computations.

2) If the inequality (3-35) is satisfied with $t_1 = t_0$ all is well, and the next iteration of the main algorithms may be begun with the knowledge that a reasonable reduction in cost of ΔV has been made. If (3-35) is not satisfied then we use the step size adjustment method.

Set

$$t_1 = \frac{t_f - t_0}{2} + t_0 = t_{01} \qquad (3\text{-}36)$$

The above procedure is repeated with this t_1 and (3-35) is checked again (with the new ΔV). If it is satisfied then the next iteration may be begun. If not, then set

$$t_1 = \frac{t_f - t_{01}}{2} + t_{01} \qquad (3\text{-}37)$$

and repeat again.

Subdividing $[t_0, t_f]$ in this way, there will come a time t_1 when the inequality (3-35) is satisfied.

In general,

$$t_1 = \frac{t_f - t_{0r}}{2} + t_{0r} = t_{0r+1} \quad , \qquad (3\text{-}38)$$

where $r = 0, 1, 2, \cdots\cdots\cdots$ and $t_{00} = 2t_0 - t_f$.

The new nominal trajectory may sometimes have a corner at t_1 since $\bar{u}(t_1)$ may be different from $u^*(t_1)$. This introduces no difficulty provided that the numerical integration routine used is capable of handling differential equations with discontinuous right-hand-sides.

It may happen that the nominal trajectory $\bar{x}(t)$ is optimal on an interval $[t_2, t_f]$; $t_2 \in [t_0, t_f]$, but is non-optimal on the interval $[t_0, t_f]$. If t_1 is being found in the manner outlined above, then a trial t_1 may fall in the interval $[t_2, t_f]$. The δx's generated in the interval $[t_1, t_f]$ would then be zero ——— because $u^*(t) = \bar{u}(t)$; $t \in [t_2, t_f]$, ——— and no reduction in cost would occur, even though the whole trajectory $\bar{x}(t)$; $t \in [t_0, t_f]$ is non-optimal in $[t_2, t_f]$. One must ensure, therefore, that t_1 will never fall in $[t_2, t_f]$. This condition is ensured easily in the following way:

At $t = t_f$, $a(\bar{x}; t) = 0$. When integrating the backward equations, monitor $|a(\bar{x}; t)|$. Record the time t_{eff} when $|a(\bar{x}; t)|$ becomes different from zero. (Or in practice, when it becomes greater than a small, positive quantity, η). The trajectory between t_{eff} and t_f satisfies a necessary condition of optimality,

$$a(\bar{x}; t) = 0 \quad ; \quad t \in [t_{eff}, t_f] \qquad (3\text{-}39)$$

If on the forward run, a time t_1 ($\neq$ t_0) is required to be determined then the time interval $[t_0, t_{eff}]$, instead of $[t_0, t_f]$, is subdivided as described earlier. As the overall trajectory becomes more and more optimal from iteration to iteration, the time t_{eff} tends to t_0. Finally, on an optimal trajectory, $|a(\bar{x};t)| < \eta_1$; $t \in [t_0, t_f]$ and $t_{eff} = t_0$ and the computation is stopped.

When programming algorithms on a digital computer it is generally necessary to use a numerical iteration routine to integrate the differential equations. This means that the interval $[t_0, t_f]$ is divided into N - 1 time steps. (i.e. t going from 1 to N).

The subdivision of $[t_0, t_{eff}]$ used for determining t_1 must be done with respect to this discrete scale, i.e. a time $N1 \in [1, N_{eff}]$ must be sought such that

$$N1 = \frac{N_{eff} - N_{or}}{2} + N_{or} = N_{or+1} \qquad (3\text{-}40)$$

where r = 0, 1, ... and $N_{oo} = 2 - N_{eff}$. In the quantity $\frac{N_{eff} - N_{or}}{2}$ of Equation (3-40), only the integer part of the result of the division is to be used. r is increased until $N1 = N_{eff} - 1$. If $N_{eff} = 1$ then only r = 0 is used.

It should be appreciated that since there are a finite number N - 1 of discrete time steps, this subdivision can only be done a finite number of times. The smallest possible non-zero time interval is

$$\frac{t_f - t_o}{N - 1}$$

It is clear that N must be large enough such that the δx's produced during this basic time interval are 'small enough'. This restriction is a practical one and brought about by the discrete time routine of the digital computation.

When ΔV and $|a(\bar{x};N1)|$ are small, but greater than η_1, the criterion (3-35) may be too severe with C = 0.5, owing to round off errors, i.e. there may come a stage where (3-35) remains unsatisfied even when $N1 = N_{eff} - 1$. If this happens, set C = 0.0 and repeat the procedure for determining N1. C = 0 is a much less stringent test, because it asks only that $\Delta V > 0$. If once again (3-35) is unsatisfied, even when $N1 = N_{eff} - 1$, then we should stop the computation since no further reduction in cost is possible. This implies that either optimality has been attained (in which case $|a(\bar{x};t_o)| < \eta_1$ and so $N_{eff} < 1$) or N is not large enough and hence $\frac{t_f - t_o}{N - 1}$ is too large for a basic time interval.

3-4: The Computational Procedure

The computational procedure is given in the flow chart and the FORTRAN IV program at the end of this thesis. The minimization of H with respect to u may be done either analytically or numerically by using the Fletcher and Powell method.

If the δx's produced by the new control are too large, as measured by criterion (3-35), then the step size adjustment routine must be used. If the problem is very nonlinear the routine will have to be used a number of times in order to determine t_1, which will be close to t_f. However, as t_1 tends to t_f, Equation (3-33) is integrated over the decreasing time intervals $[t_1, t_f]$. So this method will use less computing time in determining t_1 than existing methods in determining ε [14,15], because we do not have to integrate (3-33) over the whole interval $[t_o, t_f]$.

In the algorithm the new control is computed using

$$u(t) = u^*(t) + \beta(t)\,\delta x(t) \qquad (3\text{-}41)$$

It can happen, in non-linear problems, that $\beta(t)\delta x(t)$ becomes too large and so invalidates the local expansions in δu. However, δx might still be small enough for

$$V_x(\bar{x} + \delta x\,;\, t) = V_x + V_{xx}\,\delta x \qquad (3\text{-}42)$$

to be valid. The following alternative way can be used for computing $u(t)$:

Instead of storing $u^*(t)$ and $\beta(t)$, store $V_x(t)$ and $V_{xx}(t)$. Compute $u(t)$ directly, by minimizing $H(\bar{x}+\delta x, u, V_x+V_{xx}\delta x; t)$ with respect to u either analytically or using the Fletcher and Powell routine. In this way the radius of convergence of the algorithm may be increased.

In order that the cost decreases at each iteration, for δx sufficiently small, $a(\bar{x};t)$ must be less than zero.

$$a(\bar{x};t) = \int_{t_f}^{t_1} [H(\bar{x}, u^*, V_x; t) - H(\bar{x}, \bar{u}, V_x; t)]dt \qquad (3\text{-}43)$$

For $a(\bar{x};t_1) < 0$ it is clearly sufficient that:

$$H(\bar{x}, u^*, V_x; t) < H(\bar{x}, \bar{u}, V_x; t) \; ; \; u^* \neq \bar{u} \qquad (3\text{-}44)$$

u^* is the control that minimizes H, so (3-44) is true for $u^* \neq \bar{u}$ if

$$H_u(\bar{x}, u^*, V_x; t) = 0$$

and

$$H_{uu}^{-1}(\bar{x}, u^*, V_x; t) \quad \text{is positive definite.}$$

The quantities manipulated above must be bounded in magnitude for these conditions to be valid, so it is required that the solutions of Equations (3-29) be bounded.

CHAPTER IV

Application to A System

Using the preceding algorithms Jacobson [10,11,12] successfully solved a linear first-order problem. It seems, however, that no attempt has been made as yet to solve a higher order nonlinear problem.

In this chapter the result is presented for the solution of a second-order nonlinear problem with saturation characteristics using the differential dynamic programming approach.

Let us consider a system which is shown in Fig.1, and is described by the following equations:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -10x_1 - 2x_2 + \tanh x_1 + u$$

$$x_1(0) = 3.20 \quad \text{and} \quad x_2(0) = -2.20 .$$

The problem is to find the optimal control variables u(t) such that

$$V(\underline{x};t) = \int_0^6 (x_1^2 + x_2^2 + 0.5u^2)\,dt$$

is minimized.

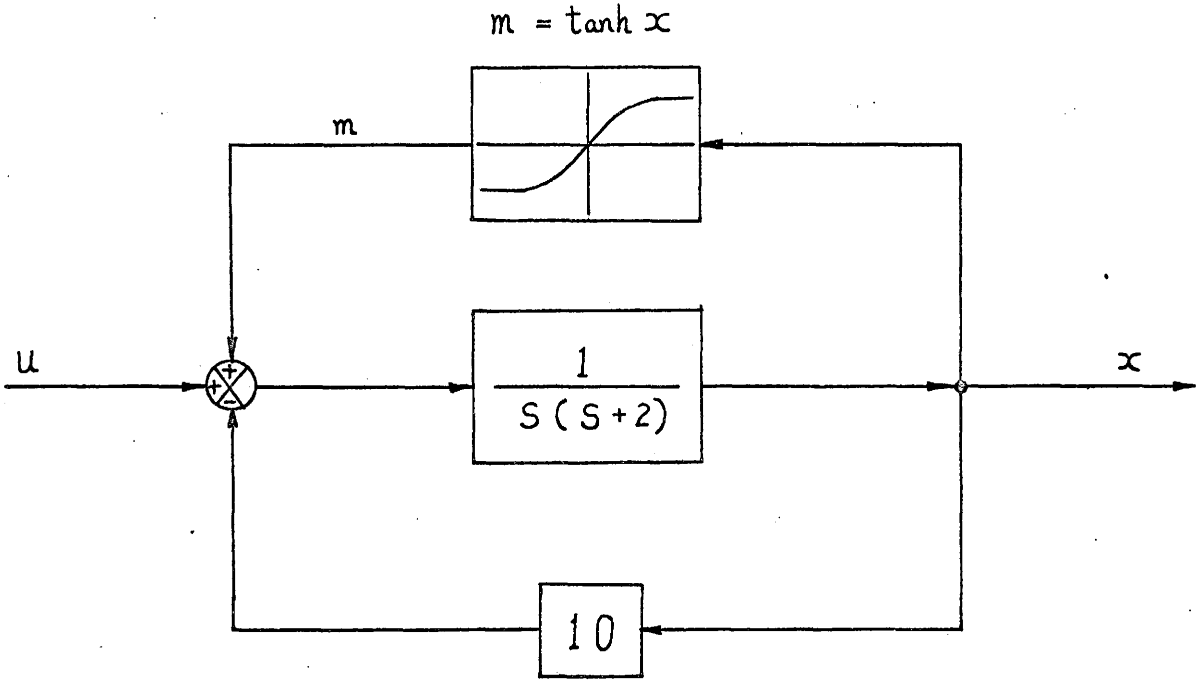


Fig. 1. A Second-order Nonlinear System under consideration

The Hamiltonian, in this case, is given by

$$H = x_1^2 + x_2^2 + 0.5\,u^2 + V_{x_1} x_2 + V_{x_2}(-10\,x_1 - 2x_2 + \tanh x_1 + u)$$

The necessary condition of optimality for the above problem is

$$H_u = u + V_{x_2} = 0$$

$$\therefore \quad u^* = -V_{x_2}$$

and the second derivative of H with respect to u is

$$H_{uu} = 1 > 0$$

Other factors of Equation (3-29) for this particular problem are

$$H_x = \begin{bmatrix} 2x_1 - V_{x_2}(9 + \tanh^2 x_1) \\ 2x_2 + V_{x_1} - 2V_{x_2} \end{bmatrix}$$

$$H_{ux} = 0$$

$$H_{xx} = \begin{bmatrix} 2 - 2V_{x_2}\tanh x_1(1 - \tanh^2 x_1) & 0 \\ 0 & 2 \end{bmatrix}.$$

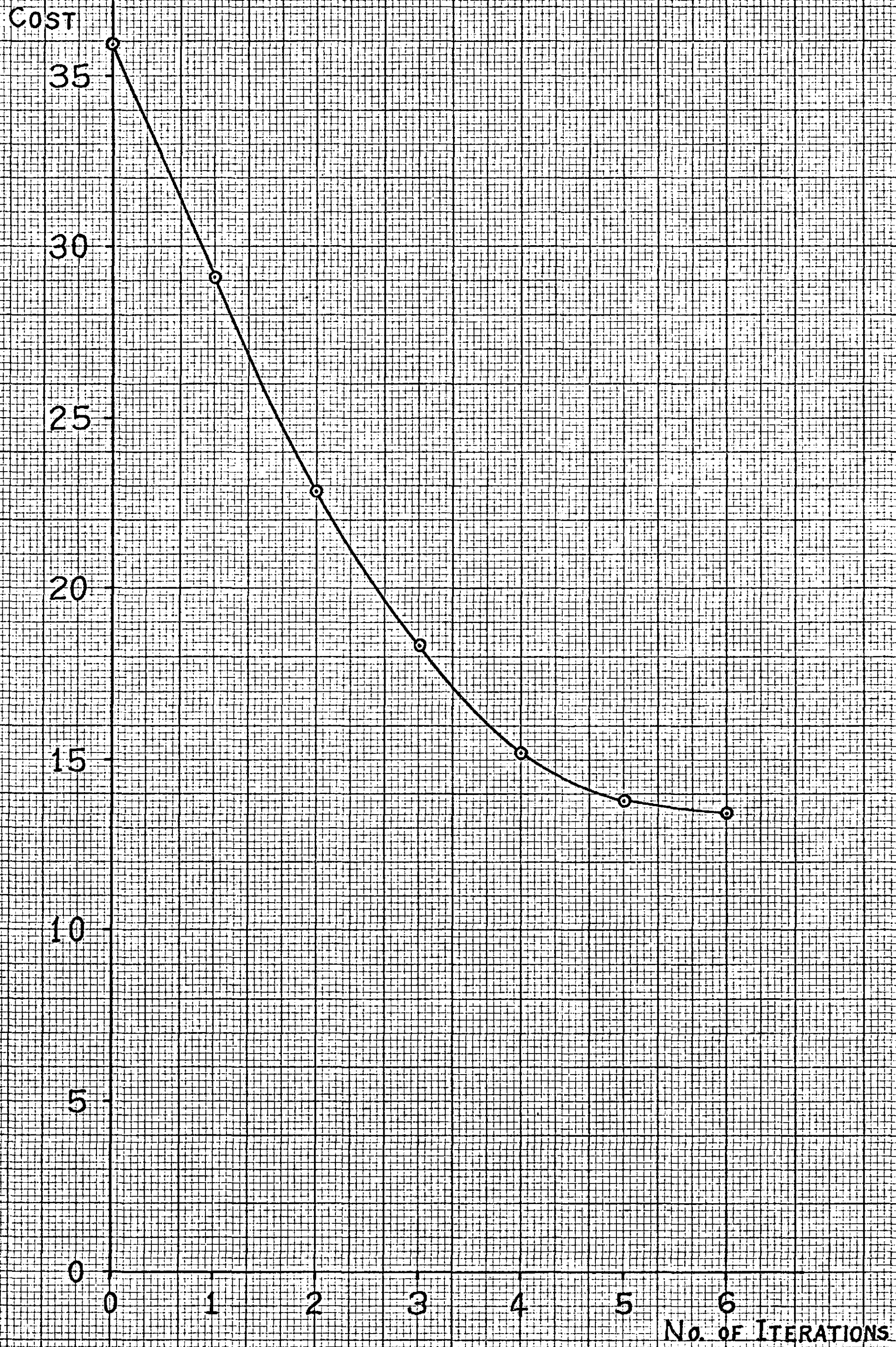


Fig.2 Cost as a function of number of iterations

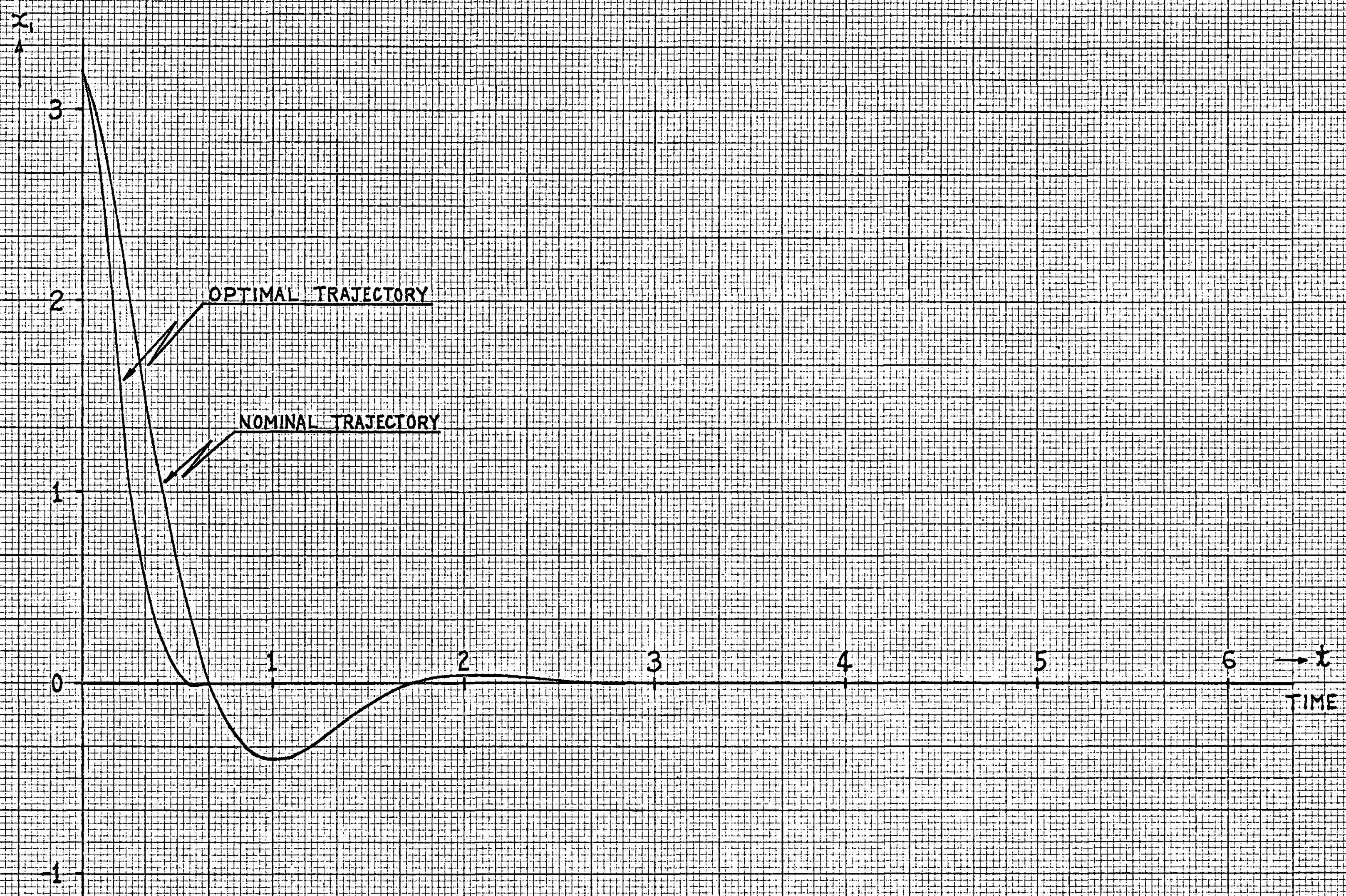


Fig.3 Unit step response

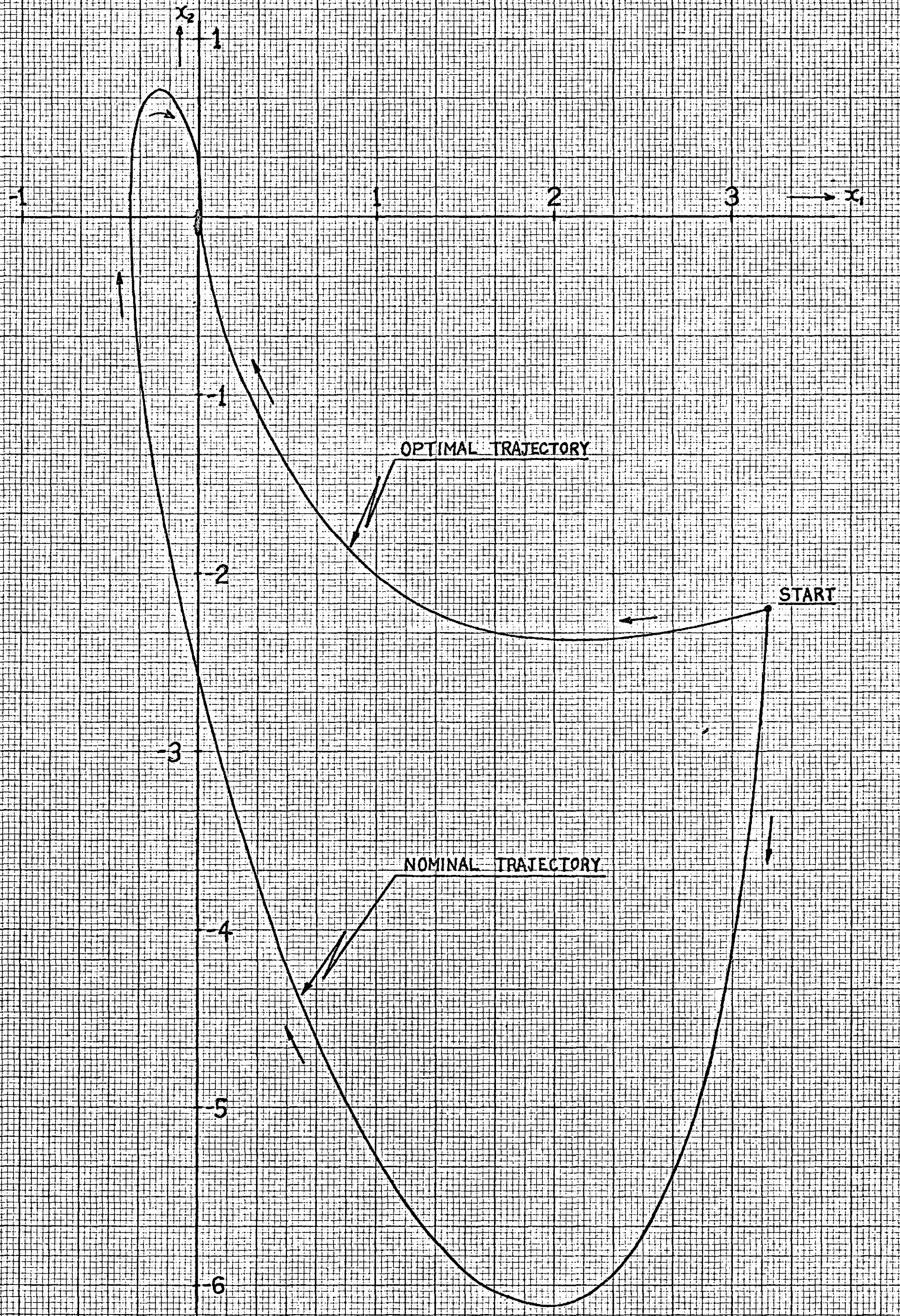


Fig.4 Phase plane diagram

An arbitrary nominal control of

$$\bar{u}(t) = -3.3\,x_1 - 2.2\,x_2 \;; \qquad t \in [\,0,\,6\,]$$

was chosen.

The DEQ subroutine (The Gill variation of the Runge-Kutta method) was used for the forward integrations. Three hundreds integration steps were used. Six iterations were performed to reach the optimal cost and it took 21 seconds for computation by CDC 6400. The cost was reduced from the nominal value of 35.9 to the optimal value of 13.4, which is shown in Fig.2. It follows that more iterations will not reduce the cost appreciably. The trajectory was considered optimal when $|a(x_o;t_o)|$, the predicted reduction in cost, was less than 0.01 . Observing Fig.2, one can find that convergence is rapid when the cost is far from the optimal value, but getting slower near the optimum.

The state x_1 versus time is shown in Fig.3. While the nominal trajectory oscillates initially, the optimal one reaches down to the final state almost immediately, without any overshoot.

Fig. 4 illustrates the phase plane diagram for both of the nominal and optimal trajectories. The superiority of the optimal trajectory over the nominal one is more obvious in Fig.4 than in Fig.3.

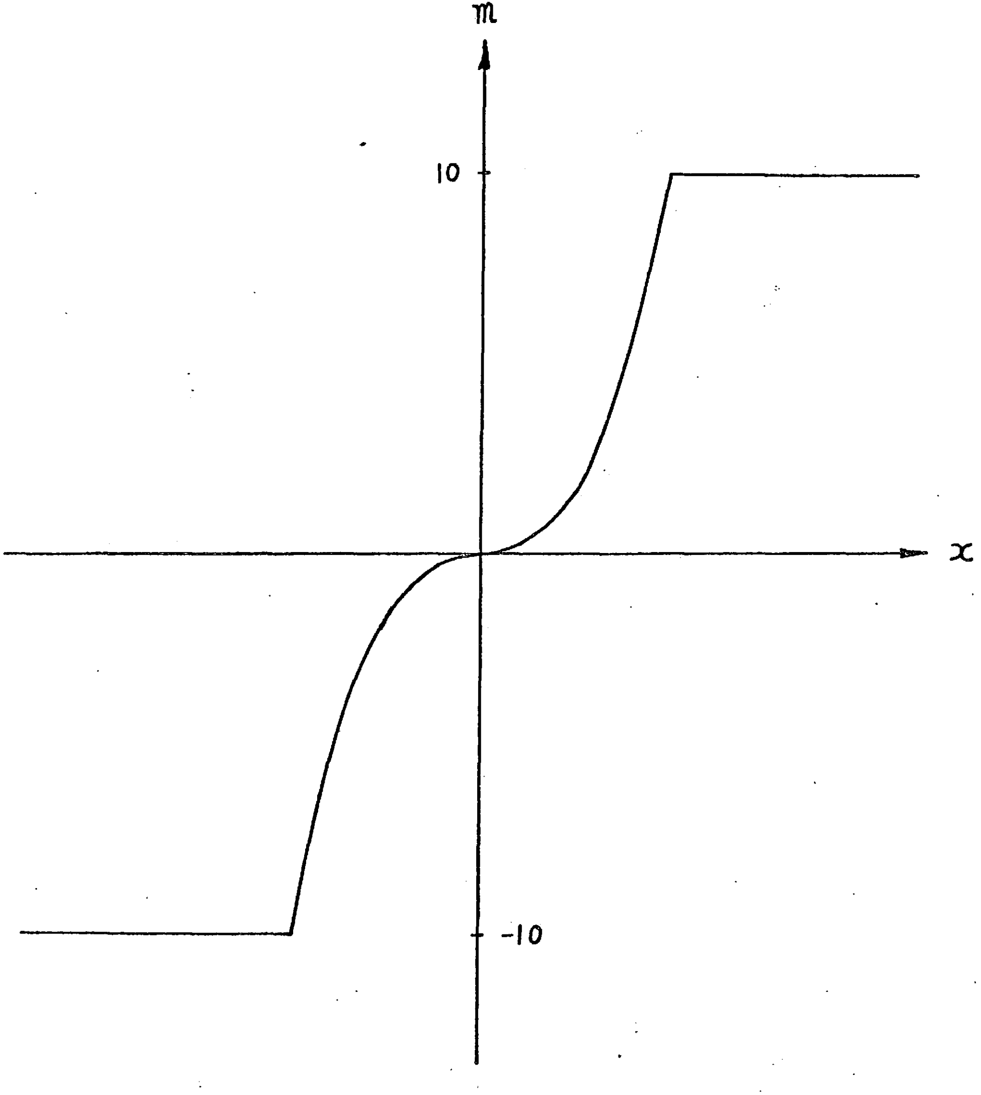


Fig. 5 A Characteristic of Nonlinearity with discontinuity in the first derivative.

As another example the differential dynamic programming algorithm was tried on a system similar to the one described by Fig.1, but for which

$$\begin{aligned} m &= x^3 && \text{for} \quad |x| \leq \sqrt[3]{10} \\ &= 10 && \text{for} \quad x > \sqrt[3]{10} \\ &= -10 && \text{for} \quad x < -\sqrt[3]{10} \end{aligned}$$

The characteristic of the nonlinearity is shown in Fig.5.

The author experienced difficulty in applying the differential dynamic programming algorithm to this case, as the final state did not converge to the desired value in various iterations. It is believed that the discontinuity in the first derivative of the nonlinear characteristic was the reason for this difficulty.

CHAPTER V

Conclusion

This thesis presents an extension of the original work of Jacobson. It is believed that Differential Dynamic Programming is a significant contribution to the field of "Numerical Techniques for Solving Optimal Control Problems", since it may handle a larger class of problems than the existing second variation methods.

The advantages of the method described in this thesis are:

(1) the principle of optimality is applied only in the neighbourhood of a nominal, possibly non-optimal, trajectory;

(2) $H_{uu}^{-1}(\bar{x},\bar{u},V_x;t)$ is not required to be positive definite along non-optimal nominal trajectories. In the algorithm $H(\bar{x},u,V_x;t)$ is minimized with respect to u and so it is required only that $H_{uu}^{-1}(\bar{x},u,V_x;t)$ be positive definite at the minimizing $u = u^*$, i.e. $H_{uu}^{-1}(\bar{x},u,V_x;t)$ must be strictly convex only in the neighbourhood of u^*;

(3) in some problems, the solutions of backward integration becomes unbounded along some nominal trajectories though along optimal trajectories it always has bounded solutions. Differential Dynamic Programming is able to compute optimal

control for these problems whereas the existing methods do not.

In this thesis the differential dynamic programming approach is applied to a nonlinear system whose nonlinearity is continuously differentiable in the whole time interval concerned.

It may also be possible to apply this approach to the following type of problems:

(1) 'bang-bang' type of control problems;

(2) systems with piecewise continuous nonlinearities over a given interval of time.

However, there is a difficulty in the direct application of this algorithm to such a system as mentioned in (2). The difficulty is that the final state is not attainable depending on values of the initial state. The author experienced the difficulty when the nonlinearity was as given in Fig.5. It is believed that this difficulty was due to the discontinuity in the first derivative of the nonlinear characteristic. Perhaps this may be overcome by introducing a suitable modification in the computational procedure.

REFERENCES

1. Bellman, R.: Dynamic Programming, Princeton University Press, 1957.

2. Bellman, R. and S. Dreyfus: Applied Dynamic Programming, Princeton University Press, 1962.

3. Bellman, R. and R . Kalaba: Dynamic Programming and Modern Control Theory, Academic Press, 1965.

4. Box, M. J.: A New Method of Constrained Optimization and A Comparison with Other Methods, Computer Journal, Vol.8, 1965.

5. Davidon, W. C.: Variable Metric Method for Minimization, AEC Research and Development Report, ANL-5990 Rev., 1959.

6. Dreyfus, S.: Dynamic Programming and the Calculus of Variations, Academic Press, 1965.

7. Droretzky, A., J. Kiefer and J. Wolfowitz: The Inventory Problem I. and II., Econometrica, 20, 1952.

8. Fletcher, R. and M.J.D. Powell: A Rapidly Convergent Descent Method for Minimization, Computer Journal, Vol.6, 1963.

9. Fletcher, R.: Function Minimization without evaluating derivatives --- a review, Computer Journal, Vol.8, 1965

10. Jacobson, D. H.: Differential Dynamic Programming Methods for Determining Optimal Control of Nonlinear Systems, Control Systems Research Report, Centre for Computing and Automation, Imperial College, London, Nov. 1966.

11. Jacobson, D. H.: Second-order and Second-variation Methods for Determining Optimal Control: A Comparative Study using Differential Dynamic Programming, International Journal of Control, Vol.7, No.2, 1968.

12. Jacobson, D. H.: New Second-order and First-order Algorithms for Determining Optimal Control, A Differential Dynamic Programming Approach, Office of Naval Research, Technical Report No.551, Division of Engineering and Applied Physics, Harvard University, Cambridge, Massachusetts, 1968.

13. Kalman, R. E.: The theory of Optimal Control and the Calculus of Variations, Mathematical Optimization Techniques, editedy by R. Bellman,1963.

14. Liebelt, P. B.: An Introduction to Optimal Estimation, Addison Wesley, 1967.

15. Mayne, D. Q.: A Second-order Gradient Method of Optimizing Nonlinear Discrete-Time Systems, International Journal of Control, Vol.3, No.1,1966.

16. Merriam, C. W.: Optimization Theory and the Design of Feedback Controls, McGraw Hill, 1964.

17. Nemhauser, G. L.: Introduction to Dynamic Programming, John Wiley, 1966.

18. Noton, A. R. M.: Introduction to Variational Methods in Control Engineering, Pergamon Press,1965.

19. Noton, A. R. M., P. Dyer and C. A. Markland: Numerical Computation of Optimal Control, IEEE trans., Vol. AC-12, No.1, 1967.

20. Powell, M. J. D.: An Efficient Method for Finding The Minimum of A Function of Several Variables Without Calculating Derivatives, Computer Journal, Vol.7, 1964.

21. Smith, F. T.: The Application of Dynamic Programming to Satellite Intersept and Rendezvous Problems, Advances in Control Systems, 1965.

22. Wald, A.: Statistical Decision Functions, John Wiley, 1950.

23. Zangwill, W. I.: Minimizing a Function Without Calculating Derivatives, Computer Journal, Vol.10, No.3, 1967.

APPENDIX I.

The Fletcher and Powell Method

General

The main disadvantage of second-order successive approximation mehtods is that the inverse of the second derivative of the function 'f' is required to be positive definite along non-optimal trajectories. This is a very restrictive requirement for nonlinear problems as f is unlikely to be quadratic in $\underline{X}$ when $\underline{X}$ is far from the minimum value.

In the neighbourhood of the minimum of f, the inverse of the matrix of second derivatives is often positive definite since many functions are essentially quadratic in the neighbourhood of their minimum.

Fletcher and Powell have developed a method to avoid the restrictive requirement, which is based on the Davidon's[5] method. In this method the function to be minimized and its first derivatives are calculated by the procedure and the inverse matrix of second derivatives is estimated from these data as the procedure progresses towards the minimum of the function. Thus when the minimum is reached, an estimate of the inverse of the matrix of second derivatives evaluated at the minimum is available.

Since the procedure does not evaluate the second derivatives directly it does not require that its inverse be positive definite.

Fletcher and Powell have shown that the procedure has quadratic convergence and for a region in which the function depends quadratically on the variables, no more than N iterations are required, where N is the number of variables.

Notation

x_μ^i : $\mu = 1,2,\cdots\cdots,N$ the set of N independent variables.

$f(\underline{x}^i)$: the value of the function to be minimized evaluated at the point $\underline{x}^i$.

$g_\mu^i(x)$: the derivatives of $f(\underline{x}^i)$ with respect to x_μ^i evaluated at $\underline{x}^i$; $g_\mu^i(x) = \frac{\partial f(\underline{x}^i)}{\partial x_\mu^i}$.

$h_{\mu\nu}^i$: a non-negative definite symmetric matrix which will be used as a matrix in the space of the variables.

ε : two times absolute accuracy to which the function $f(\underline{x}^i)$ is to be minimized.

Geometrical Interpretation

It is convenient to use geometrical concepts to describe the minimization procedure. We do so by considering the variables x_{μ}^{i} to be coordinates of a point in an N – dimensional linear space. As shown in Fig.6 -(a), the set of $\underline{x}$ for which $f(\underline{x})$ is constant forms an N - 1 dimensional surface in this space. One of this family of surfaces passes through each $\underline{x}$, and the surface about a point is characterized by the gradient of the function at that point:

$$g_{\mu}^{i}(\underline{x}) = \frac{\partial f(\underline{x})}{\partial x_{\mu}^{i}}$$

These N components of the gradient can in turn be considered as the coordinates of a point in a different space, as shown in Fig.6 -(b). As long as $f(\underline{x})$ is differentiable at all points, there is a unique point $\underline{g}$ in the gradient space associated with each point $\underline{x}$ in the position space, though there may be more than one $\underline{x}$ with the same $\underline{g}$.

In the neighbourhood of any one point A, the second derivatives of $f(\underline{x})$ specify a linear mapping of changes in position, $d\underline{x}$, onto changes in gradient $d\underline{g}$, in accordance with the equation

$$d g_{\mu} = \frac{\partial^2 f}{\partial x_{\mu} \partial x_{\nu}} d x_{\nu}$$

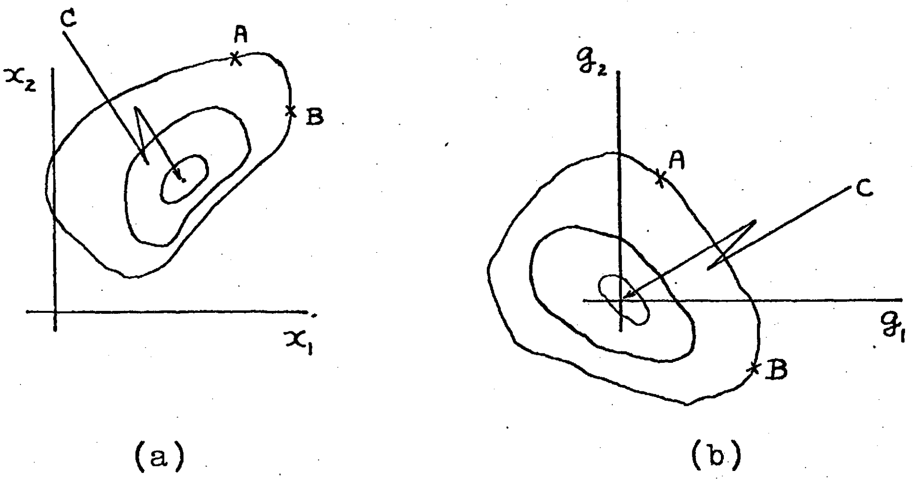


Fig. 6 Geometrical interpretation of x^i_μ and $g^i_\mu(\underline{x})$

The vectors $d\underline{x}$ and $d\underline{g}$ will be in the same direction only if $d\underline{x}$ is an eigenvector of the matrix

$$\left\| \frac{\partial^2 f}{\partial x_\mu \, \partial x_\nu} \right\|$$

If the ratios among the corresponding eigenvalues are large, then for most $d\underline{x}$ there will be considerable difference in the directions of these two vectors.

<u>The Fletcher and Powell Method</u>

In this method $\left\| \frac{\partial^2 f}{\partial x_\mu \partial x_\nu} \right\|^{-1}$ is not evaluated directly; instead an initial trial value is assumed for it. This matrix, denoted by $h_{\mu\nu}$, specifies a linear mapping of all changes in the gradient onto changes in position. It may initially be chosen to be any positive definite symmetric matrix.

After making a change in the variable $\underline{x}$, this trial value is improved on the basis of actual relation between the changes in $\underline{g}$ and $\underline{x}$, i.e. this matrix is modified after the i^{th} iteration using the information gained by moving down the direction

$$\underline{s}^i = - h_{\mu\nu}^i \underline{g}^i$$

in accordance with

$$x_{min} - \underline{x} = - \left\| \frac{\partial^2 f}{\partial x_\mu \partial x_\nu} \right\|^{-1} \underline{g}$$

which is the displacement between the point $\underline{x}$ and the minimum $\underline{x}_{min}$.

The modification is such that $\underline{\sigma}^i$, the step to the minimum down the line

$$\underline{x} = \underline{x}^i + \lambda \underline{s}^i$$

is effectively an eigenvector of the matrix

$$h_{\mu\nu}^{i+1} \cdot \left\| \frac{\partial^2 f}{\partial x_\mu \partial x_\nu} \right\|$$

This ensures that as the procedure converges $h_{\mu\nu}$ tends to $\left\| \frac{\partial^2 f}{\partial x_\mu \partial x_\nu} \right\|^{-1}$ evaluated at the minimum.

It is convenient to take the unit matrix initially for $h_{\mu\nu}$ so that the first direction is down the line of steepest descent.

Let the current point be $\underline{x}^i$ with gradient $\underline{g}^i$ and matrix $h_{\mu\nu}^i$. The iteration can then be stated as follows.

The direction of the first step is chosen by using the matrix $h_{\mu\nu}$ in the relation

$$\underline{s}^i = - h^i_{\mu\nu}\, \underline{g}^i$$

The component of the gradient in this direction is evaluated through the relation

$$g^i_s = \underline{s}^i \underline{g}^i$$

$-g^i_s$ is the squared length of $\underline{g}^i$, and hence the improvement to be expected in the function is $-\frac{1}{2} g^i_s$. The positive definiteness of $h_{\mu\nu}$ ensures that g_s is negative, so that the step is in a direction which (at least initially) decreases the function. If the reduction is within the accuracy desired, then the minimum has been determined. If not we continue with the procedure.

In a first effort to find a region containing the minimum, we take a step which is twice the size that would locate the minimum if the trial $h_{\mu\nu}$ were $\left\| \frac{\partial^2 f}{\partial x_\mu \partial x_\nu} \right\|^{-1}$. However, in order to prevent this step from being unreasonably large when the trial $h_{\mu\nu}$ is a poor estimate, we restrict the step to a length such that $(\lambda \underline{s}^i)\underline{g}^i$, the decrease in the function if it continues to decrease linearly, is not greater than some preassigned maximum, 2f. We then change $\underline{x}^i$ by

$$\underline{x}^+ = \underline{x}^i + \lambda \underline{s}^i$$

and calculate the new value of the function and its gradient at $\underline{x}^+$. If the projection $\underline{s}^i\underline{g}^+ = g_s^+$ of the new gradient in the direction of the step is positive, or if the new value of the function f^+ is greater than the original f^i, then there is a relative minimum along the direction $\underline{s}^i$ between $\underline{x}^i$ and $\underline{x}^+$, and we proceed to the next process where we will interpolate its position. However, if neither of these conditions is fulfilled, the function has decreased and is decreasing at the point $\underline{x}^+$, and we infer that the step taken was too small.

If the step had been taken on the basis of $h^i_{\mu\nu}$, i.e. $\lambda = 1$, we modify $h^i_{\mu\nu}$ so as to double the squared length of $\underline{s}^i$, leaving the length of all perpendicular vectors unchanged. This is accomplished by making

$$h^i_{\mu\nu} = h^i_{\mu\nu} + \frac{1}{l} s^i_\mu s^i_\nu$$

where l is the squared length of $\underline{s}^i$. This doubles the determinant of $h^i_{\mu\nu}$. The process is then repeated, starting from the new position.

Now we proceed to estimate the location of the relative minimum within the interval selected by the preceding processes.

The values of f^i and f^+ of the function at points $\underline{x}^i$ and $\underline{x}^+$ are known, and so are its slopes, g_s^i and g_s^+, at these two points. We interpolate for the location of the minimum by choosing the "smoothest" curve satisfying

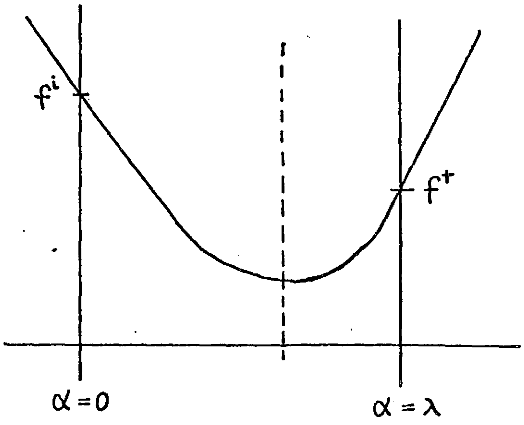


Fig. 7 Plot of f($\underline{x}$) along a one-dimensional interval

the boundary conditions at $\underline{x}^i$ and $\underline{x}^+$, namely, the curve defined as the one which minimizes

$$\int_0^\lambda \left(\frac{d^2 f}{d\alpha^2}\right) d\alpha$$

over the curve. This is the curve formed by a flat spring fitted to the known ordinates and slopes at the end points. The resulting curve is a cubic, and its slope at any α $(0 \leq \alpha \leq \lambda)$ is given by

$$g_s(\alpha) = g_s^i - \frac{2\alpha}{\lambda}(g_s^i + Z) + \frac{\alpha^2}{\lambda^2}(g_s^i + g_s^+ + 2Z)$$

where

$$Z = \frac{3(f^i - f^+)}{\lambda} + g_s^i + g_s^+$$

The root of $g_s(\alpha)$ that corresponds to a minimum lies between 0 and λ by virtue of the fact that $g_s^i < 0$ and either $g_s^+ > 0$ or $Z < g_s^i + g_s^+$. It can be expressed as

$$\alpha_{min} = \lambda(1 - a)$$

where

$$a = \frac{g_s^+ - W - Z}{g_s^+ - g_s^i + 2W}$$

and

$$W = (Z^2 - g_s^i g_s^+)^{\frac{1}{2}}$$

Now the step $\underline{\sigma}^i$ can be obtained as

$$\underline{\sigma}^i = \alpha \underline{s}^i$$

then we change $\underline{x}^i$ by

$$\underline{t} = \underline{x}^i + \underline{\sigma}^i$$

and calculate the new value of the function $\bar{f}$ and its gradient $\underline{\bar{g}}$ at $\underline{t}$. If the new value of the function $\bar{f}$ is greater than f^i or f^+ by a significant amount ε, the interpolation is not considered satisfactory and a new one is made within that part of the original interval for which f^i at the end point is smaller. Then $\bar{f}$ and $\underline{\bar{g}}$ are used as f^{i+1} and $\underline{g}^{i+1}$ respectively, for the next iteration.

The matrix $h_{\mu\nu}$ is modified by adding to it two terms A^i and B^i, which are given by Fletcher and Powell

$$h_{\mu\nu}^{i+1} = h_{\mu\nu}^i + A^i + B^i$$

where

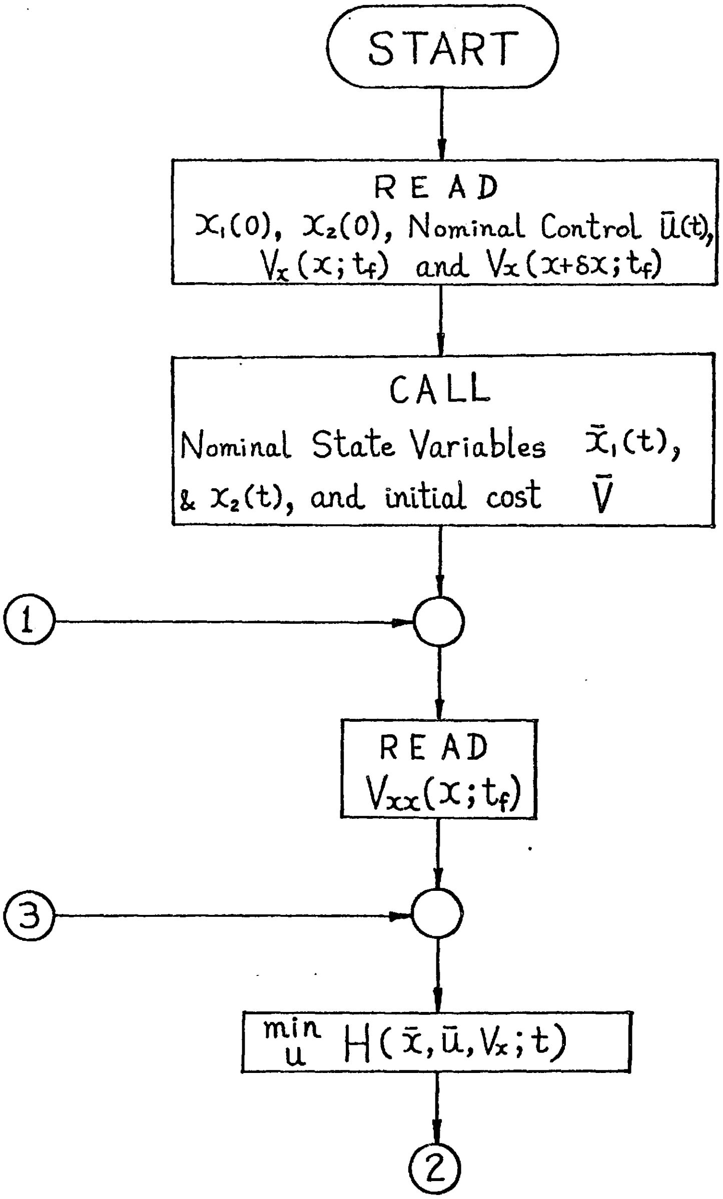
$$A^{i} = \frac{\underline{s}^{i}\,(\underline{s}^{i})^{T}}{(\underline{s}^{i})^{T}\,\underline{y}^{i}}$$

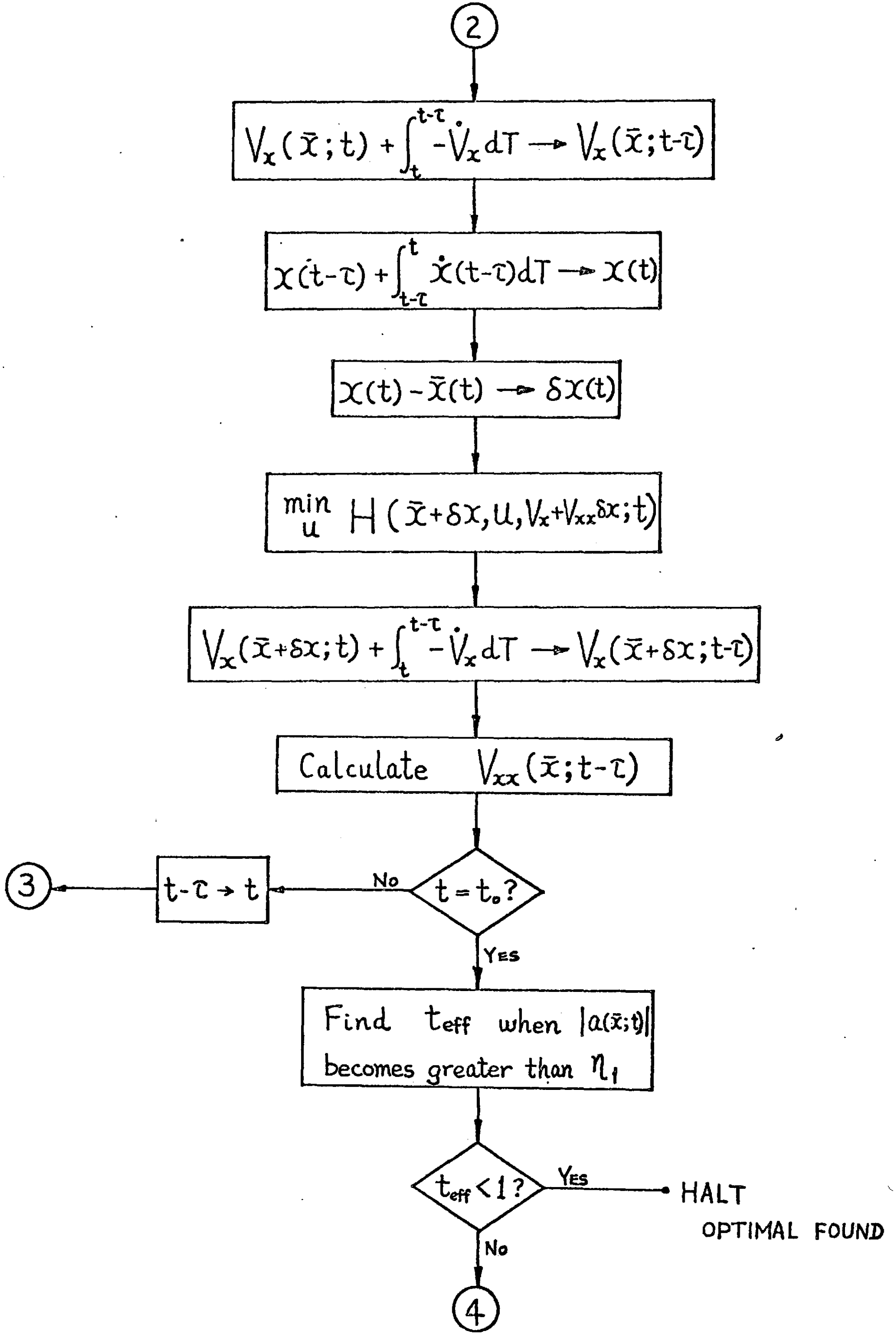
$$B^{i} = \frac{-\,h_{\mu\nu}\,\underline{y}^{i}\,(\underline{y}^{i})^{T}\,h_{\mu\nu}}{(\underline{y}^{i})^{T}\,h_{\mu\nu}\,\underline{y}^{i}}$$

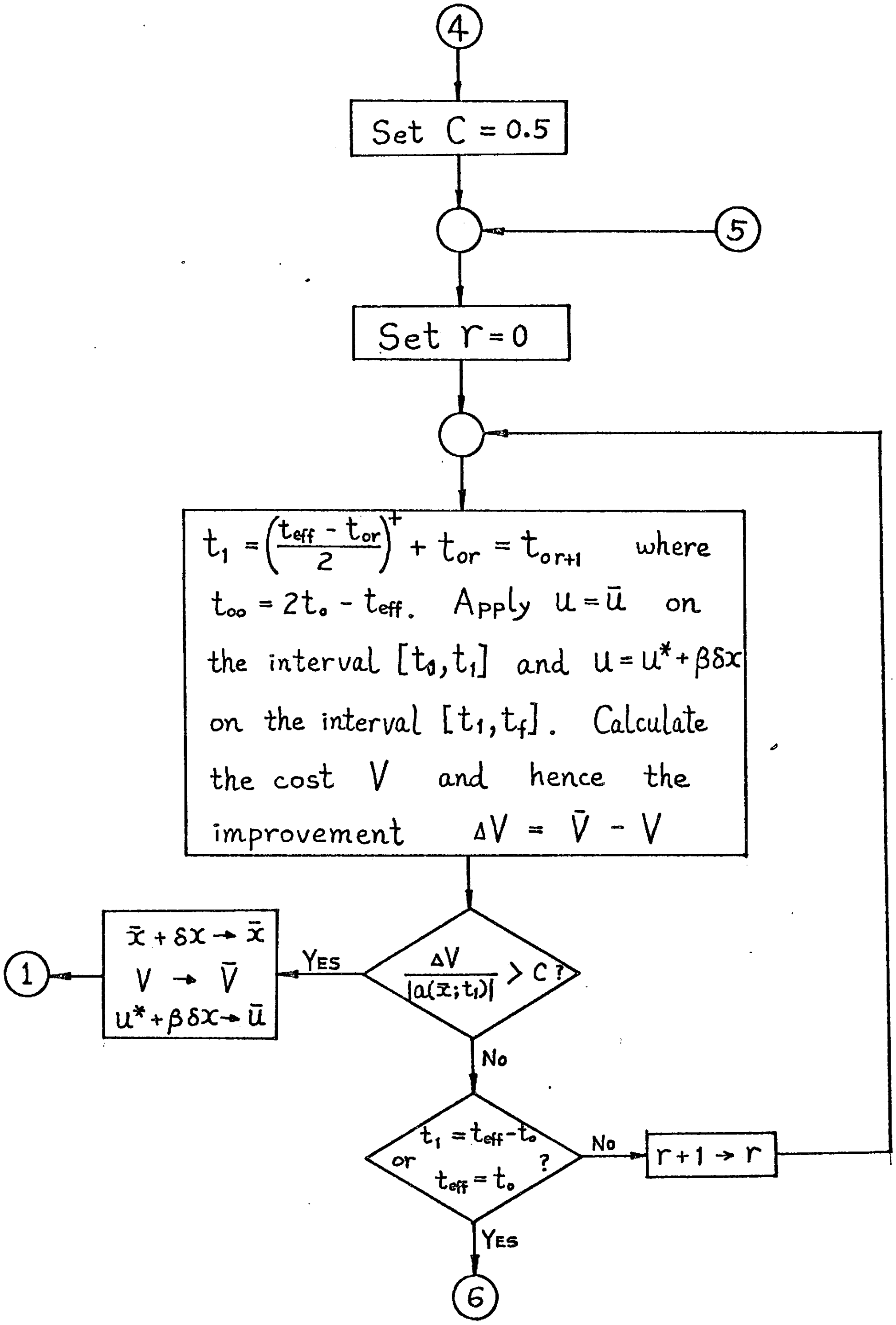
$$\underline{y}^{i} = \underline{g}^{i+1} - \underline{g}^{i}$$

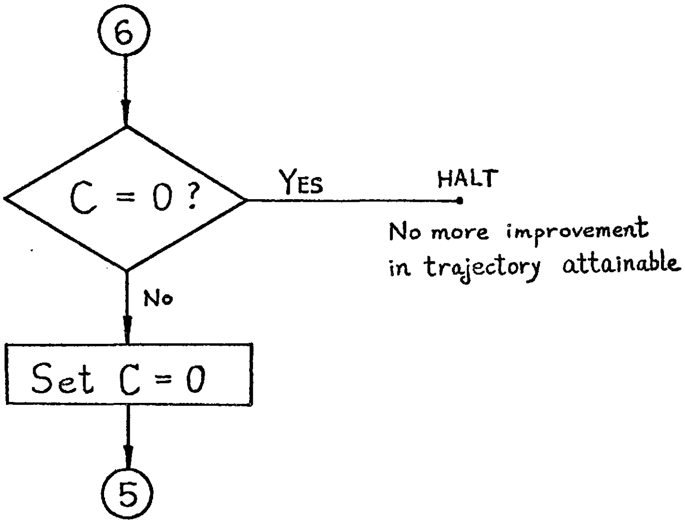
APPENDIX II

Flow Chart of Differential Dynamic Programming


START
READ
$x_1(0)$, $x_2(0)$, Nominal Control $\bar{u}(t)$,
$V_x(x;t_f)$ and $V_x(x+\delta x;t_f)$
CALL
Nominal State Variables $\bar{x}_1(t)$,
& $x_2(t)$, and initial cost $\bar{V}$
1
READ
$V_{xx}(x;t_f)$
3
$\min_u H(\bar{x},\bar{u},V_x;t)$
2


2
$V_x(\bar{x};t) + \int_t^{t-\tau} -\dot{V}_x dT \rightarrow V_x(\bar{x};t-\tau)$
$x(t-\tau) + \int_{t-\tau}^{t} \dot{x}(t-\tau)dT \rightarrow x(t)$
$x(t) - \bar{x}(t) \rightarrow \delta x(t)$
$\min_u H(\bar{x}+\delta x, u, V_x+V_{xx}\delta x; t)$
$V_x(\bar{x}+\delta x;t) + \int_t^{t-\tau} -\dot{V}_x dT \rightarrow V_x(\bar{x}+\delta x;t-\tau)$
Calculate $V_{xx}(\bar{x};t-\tau)$
$t = t_o$?
No
$t-\tau \rightarrow t$
3
Yes
Find t_{eff} when $|a(\bar{x};t)|$ becomes greater than η_1
$t_{eff} < 1$?
Yes
HALT
OPTIMAL FOUND
No
4


4
Set $C = 0.5$
5
Set $r = 0$
$t_1 = \left(\frac{t_{eff} - t_{or}}{2}\right)^+ + t_{or} = t_{or+1}$ where $t_{oo} = 2t_o - t_{eff}$. Apply $u = \bar{u}$ on the interval $[t_0, t_1]$ and $u = u^* + \beta\delta x$ on the interval $[t_1, t_f]$. Calculate the cost V and hence the improvement $\Delta V = \bar{V} - V$
$\frac{\Delta V}{|a(\bar{x};t_1)|} > C$?
YES
$\bar{x} + \delta x \rightarrow \bar{x}$
$V \rightarrow \bar{V}$
$u^* + \beta\delta x \rightarrow \bar{u}$
1
No
$t_1 = t_{eff} - t_o$ or $t_{eff} = t_o$?
No
$r + 1 \rightarrow r$
YES
6


6
C = 0 ?
YES
HALT
No more improvement
in trajectory attainable
No
Set C = 0
5

APPENDIX III

```
      DIMENSION X1(300), X2(300), U(300), VX1(300), VX2(300)
     1        ,.XM1(300), XM2(300), DX1(300), DX2(300),
     2          UOP2(300), ACHD(300), NO(300)
      EXTERNAL TAMAKI, CEIL
      COMMON UC,XL,YL
      READ(5,999) A1, A2
  999 FORMAT(2F15.2)
   31 FORMAT(I5, E15.5)
   32 FORMAT(4X1HM, 5X2HX1, 9X3HXM1, 8X2HX2, 9X3HXM2, 9X1HU,
     1       9X4HUOP1, 7X4HUOP2, MX5HVSX11, 6X5HVSX21,
     2       6X5HVSX22, 6X3HVX2 /)
   33 FORMAT(1XI4, 11E11.2)
  331 FORMAT( )
  332 FORMAT(//)
  333 FORMAT(///)
  334 FORMAT(1H1)
  335 FORMAT(/ 20X5HNEFF=, I5, 10X3HN1=, I5)
      DT = 0.02
      NT = 1
      TX = TY = 0.
      X = A1
      Y = A2
      VX1(300) = VX2(300) = -1.0E-8
C
C    INITIAL TRAJECTORY AND COST
C
      NASA = 0
      VOLD = 0.
      U(1) = -3.3*A1 - 2.2*A2
      UC = U(1)
      DO 20 IA = 1,300
      XL = X
      YL = Y
      UC = U(IA)
      CALL DEQ(TX, DT, NT, X, DX, WORK, TAMAKI)
      CALL DEQ
      CALL DEQSET
      X1(IA) = X
      CALL DEQ(TY, DT, NT, Y, DY, WORK, CEIL)
      CALL DEQ
      CALL DEQSET
      X2(IA) = Y
      IF(IA.EQ. 300) GO TO 20
      U(IA+1) = -3.3*X1(IA) - 2.2*X2(IA)
   20 VOLD = VOLD + (X1(IA)**2 + X2(IA)**2 + 0.5*U(IA)**2)*DT
      WRITE(6,31) NASA, VOLD
C
C    INITIAL SMALL CHANGES IN X
C
      DO 211 L = 1,300
      DX1(L) = -0.05*X1(L)
  211 DX2(L) = -0.05*X2(L)
  100 CONTINUE
      KPAGE = 300
      KLINE = 0
      NB = 1
      M = 300
      VSX11 = VSX12 = VSX21 = VSX22 = -1.0E-9
  101 CONTINUE
      IF(KLINE.NE.5) GO TO 1500
      WRITE(6,331)
      KLINE = 0
 1500 KLINE = KLINE + 1
      IF(M.NE.300) GO TO 1000
      WRITE(6,332)
      WRITE(6,32)
 1000 CONTINUE
      IF(KPAGE.NE.255) GO TO 1200
      WRITE(6,333)
      WRITE(6,32)
      KPAGE = 300
 1200 CONTINUE
      UOP1 = -VX2(M)
      H = X1(M)**2 + X2(M)**2 + 0.5*U(M)**2 + VX1(M)*X2(M) +
     1    VX2(M)*(-10.*X1(M) - 2.*X2(M) + TANH(X1(M)) + U(M))
      CH= X1(M)**2 + X2(M)**2 + 0.5*U(M)**2 + VX1(M)*X2(M) +
     1    VX2(M)*(-10*X1(M) - 2.*X2(M) + TANH(X1(M)) + UOP1)
      ACHD(M) = DT*ABS(CH - H)
      DVX1 = 2.*X1(M) - (9. + TANH(X1(M))**2)*VX2(M) +
     1       VSX12*(UOP1 - U(M))
      DVX2 = 2.*X2(M) + VX1(M) - 2.*VX2(M) +
     1       VSX22*(UOP1 - U(M))
      IF(M.EQ.1) GO TO 633
      VX1(M-1) = VX1(M) - DT*DVX1
      VX2(M-1) = VX2(M) - DT*DVX2
      TM = M - 1
      TX = TY = TM*DT
      PX = X1(M-1) + DX1(M-1)
      PY = X2(M-1) + DX2(M-1)
      GO TO 634
  633 PX = A1 * 0.999
      PY = A2
      TX = TY = 0.
  634 UC = U(M)
      XL = PX
      YL = PY
      CALL DEQ(TX, DT, NT, PX, DX, WORK, TAMAKI)
      CALL DEQ
      CALL DEQSET
      XM1(M) = PX
      CALL DEQ(TY, DT, NT, PY, DY, WORK, CEIL)
      CALL DEQ
      CALL DEQSET
      XM2(M) = PY
C
C     NEW SMALL CHANGES IN STATE VARIABLES
C
      IF(M.EQ.1) GO TO 777
      DX1(M) = XM1(M) - X1(M)
      DX2(M) = XM2(M) - X2(M)
      GO TO 666
  777 DX1(1) = -0.05*X1(1)
      DX2(1) = -0.05*X2(1)
C
  666 UOP2(M) = -(VX2(M) + VSX21*DX1(M) + VSX22*DX2(M))
      WRITE(6,33) M, X1(M), XM1(M), X2(M), XM2(M), U(M),
     1       UOP1, UOP2(M), VSX11, VSX21, VSX22, VX2(M)
      IF(M.EQ.1) GO TO 310
      D11XV = 2. - 2.*VX2(M)*TANH(X1(M))*(1. - TANH(X1(M))**2)
     1     -(9.+TANH(X1(M))**2)*(VSX21+VSX12) - VSX21**2
      D12XV = -(9.+TANH(X1(M))**2)*VSX22 + VSX11 -
     1        2.*VSX12 - VSX21*VSX22
      D21XV = VSX11 -2.*VSX21 - (9.+TANH(X1(M))**2)*VSX22 -
     1        VSX22*VSX21
      D22XV = 2. + VSX12 + VSX21 - 4.*VSX22 - VSX22**2
      VSX11 = VSX11 - DT*D11XV
      VSX12 = VSX12 - DT*D12XV
      VSX21 = VSX21 - DT*D21XV
      VSX22 = VSX22 - DT*D22XV
      AVX = ABS(VX2(M-1))
      IF(AVX.GT. 1.0E+02) GO TO 310
      M = M-1
      KPAGE = KPAGE - 1
      GO TO 101
  310 NB = M
  111 ALON = 0.
      JA = 300
  112 ALON = ALON + ACHD(JA)
      IF(ALON .GT. 0.01) GO TO 113
      JA = JA - 1
      IF(JA.EQ.0) GO TO 230
      GO TO 112
  113 NEFF = JA
      IF(NEFF. LE. NB) GO TO 103
      GO TO 104
  103 NB = NEFF -1
  104 C = 0.5
  114 K = 1
  115 NO(1) = 2*NB - NEFF
      N1 = NO(K+1) = (NEFF - NO(K))/2 + NO(K)
C
C    CALCULATION OF COST
C
      VC = 0.
      IF(N1.LE.1) GO TO 321
      LK = N1 -1
      DO 21 IB=1,LK
      VC = VC + (X1(IB)**2 + X2(IB)**2 + 0.5*U(IB)**2)*DT
   21 CONTINUE
  321 VD = 0.
      DO 22 IC = N1, 300
      VD = VD + (X1(IC)**2 + X2(IC)**2 + 0.5*UOP2(IC)**2)*DT
   22 CONTINUE
      VNEW = VC + VD
      DV = VOLD - VNEW
      CRI = 0.
      DO 23 ID = N1, 300
      CRI = CRI + ACHD(ID)
   23 CONTINUE
      RATIO = DV/CRI
      IF(RATIO.GT.C) GO TO 108
      IF(N1.EQ.NEFF-1) GO TO 71
      IF(NEFF.EQ.1) GO TO 71
      K = K+1
      GO TO 115
   71 CONTINUE
      IF(C.EQ.0.) GO TO 155
      C = 0.
      GO TO 114
  108 CONTINUE
      DO 24 IE = N1, 300
      U(IE) = UOP2(IE)
      X1(IE) = XM1(IE)
      X2(IE) = XM2(IE)
   24 CONTINUE
      VOLD = VNEW
      NASA = NASA + 1
      WRITE(6,335) NEFF, N1
      WRITE(6,334)
      WRITE(6,31) NASA, VOLD
      GO TO 100
  155 WRITE(6,35)
   35 FORMAT(10X 20H NO MORE IMPROVEMENT /)
      GO TO 225
  230 WRITE(6,36)
   36 FORMAT(10X 14H OPTIMAL FOUND /)
  225 CONTINUE
      STOP
      END
C
      SUBROUTINE TAMAKI(X, DX, T)
      COMMON UC, XL, YL
C
      DX = YL
C
      RETURN
      END
C
      SUBROUTINE CEIL(Y, DY, T)
      COMMON UC, XL, YL
C
      DY = -10.*XL - 2.*Y + TANH(XL) + UC
C
      RETURN
      END
```