September 1978

LIDS-P-849

Submitted to the IEEE
Trans. on Autom. Control

# AN OPTIMAL CONTROL APPROACH TO DYNAMIC ROUTING IN DATA COMMUNICATION NETWORKS

## PART I: PRINCIPLES

Franklin H. Moss

and

Adrian Segall

This research was supported by the Advanced Research Project Agency of the Department of Defense (monitored by ONR) under contract No. N00014-75-C-1183 and by the Technion Research and Development Foundation Ltd., Research No. 050-383.

The authors were with the Department of Electrical Engineering and Computer Science, M.I.T., Cambridge, Mass. Franklin Moss is now with the IBM Israel Scientific Center, Technion City, Haifa, Israel, and Adrian Segall is with the Department of Electrical Engineering, Technion - Israel Institute of Technology, Haifa, Israel.

ABSTRACT

A continuous state space model for the problem of dynamic routing in data communication networks has been recently proposed. In this paper we present an algorithm for finding the feedback solution to the associated linear optimal control problem with linear state and control variable inequality constraints when the inputs are assumed to be constant in time. The Constructive Dynamic Programming Algorithm, as it is called, employs a combination of necessary conditions, dynamic programming and linear programming to construct a set of convex polyhedral cones which cover the admissible state space with optimal controls. Due to several complicating features which appear in the general case the algorithm is presented in a conceptual form which may serve as a framework for the development of numerical schemes for special situations. In this vain the authors present in a forthcoming paper the case of single destination network problems with all equal weightings in the cost functional.

TABLE OF CONTENTS

I. Introduction........................................................ 1

II. The Model for Dynamic Routing in Data Communication Networks..... 8

III. Feedback Solution Fundamentals..................................13

IV. Geometrical Characterization of the Feedback Space for Constant Inputs..................................................27

V. Examples of the Backward Construction of the Feedback Space......29

VI. The Constructive Dynamic Programming Algorithm...................49

VII. Conclusions.....................................................63

Appendix A...........................................................64

Appendix B...........................................................69

References...........................................................72

## I. INTRODUCTION

A data communication network is a facility which interconnects a number of data devices, such as computers and terminals, by communication channels for the purpose of transmission of data between them. Each device can use the network to access some or all of the resources available throughout the network. These resources consist primarily of computational power, memory capacity, data bases and specialized hardware and software. With the rapidly expanding role being played by data processing in today's society it is clear that the sharing of costly computer resources is an eventual, if not current, desirability. In recognition of this fact, research in data communication networks began in the early 1960's and has blossomed into a sizeable effort in the 1970's. A variety of data networks have been designed, constructed and implemented with encouraging success.

We begin our discussion with a brief description of the basic components of a data communication network and their respective functions. For more detail, refer to [1]. Fundamentally, what is known as the *communication subnetwork* consists of a collection of nodes which exchange data with each other through a set of connective links. Each node essentially consists of a minicomputer and associated devices which may possess data storage capability and which serve the function of directing data which passes through the node. The links are data transmission channels of a given rate capacity. The data devices which utilize the service of the communication subnetwork, known as *users*, insert data into and receive data from the subnetwork through the nodes.

The data traveling through the network is organized into messages, which are collections of bits which convey some information. In this paper we shall be concerned with the class of networks which contain message storage capability at the nodes, known as *store-and-forward networks*. The method by which messages are sent through the network from node of origin to node of destination is according to the technique known as *message switching*, in which only one link at a time is used for the transmission of a given message. Starting at the source node, the message is stored at the node until its time comes to be transmitted on an outgoing link to a neighboring node. Having arrived at that node it is once again stored in its entirity until being transmitted to the next node. The message continues in this fashion to traverse links and wait at nodes until it finally reaches its destination node. At that point it leaves the communication subnetwork by being immediately transmitted to the appropriate user.

Frequent use is made of a special type of message switching known as *packet switching*. This is fundamentally the same as message switching, except that a message is decomposed into smaller pieces of maximum length called packets. These packets are properly identified and work their way through the network in the fashion of message switching. Once all of the packets belonging to a given message arrive at the destination node, the message is reassembled and delivered to the appropriate user. Henceforth, any mention of message or message switching will apply equally as well to packets or packet switching.

The problems of routing messages through the network from their nodes of origin to their nodes of destination is one of the fundamental issues involved in the operation of networks. As such, it has received considerable attention in the data communication network literature. It is clear that the efficiency with which messages are sent to their destinations determines to a great extent the desirability of networking data devices. The subjective term "efficient" may be interpreted mathematically in many ways, depending on the specific goals of the networks for which the routing procedure is being designed. For example, one may wish to minimize total message delay, maximize message throughput, etc. In this paper we shall restrict attention to the *minimum delay message routing problem.*

In order to arrive at a routing procedure for a data-communication network one must begin with some representation of the system in the form of a mathematical model. As is always the case, there are a number of important considerations which enter into the choice of an appropriate model. Firstly, one wishes the model to resemble the nature of the actual system as closely as possible — for instance, if the system is dynamic the model should be capable of simulating its motions. Secondly, the model should describe the system's behavior directly at the level in which one is interested — not too specific or not too general. Finally, the model should be of some use in analyzing or controlling the ultimate behavior of the system.

These issues pose challenging problems in the formulation of models which are to be used as a basis for the design of message routing procedures for data communications networks. The basic problem is that there is no natural model which describes the phenomenon of data flow in such a network since the nature of this flow is largely dpendent upon the character of the routing procedure to be developed.

In this paper we do not confront the question of modelling message flow but rather base our analysis on a model proposed by A. Segall in [2]. This model, which is a continuous dynamical state space description of message flow, was formulated in order to overcome some basic deficiencies in previous models which are based upon *queueing theory*. The fundamental advantages of this model with respect to previous models are discussed in detail in [2] and are presented here briefly. Firstly, the model may accommodate *completely dynamic* strategies (continual changing of routes as a function of time) whereas previous techniques have been addressed primarily to *static* strategies (fixed routes in time) and *quasi-static* strategies (routes changing with intervals of time that are long compared to the time constants of the system). Next, the model can handle *closed-loop* strategies, where the routes are a function of the message congestion in the network, in contrast to the *open-loop* strategies of static procedures, in which the routes are functions only of the various parameters of the system. Finally, the *independence assumption* regarding message statistics, which is required in order to derive routing procedures based upon queueing theory, is not required to derive procedures based upon the model under consideration. On the other hand, dynamic and closed-loop procedures possess several drawbacks, such as difficulty in computation of the routing algorithm and implementation in the network.

In [2], the minimum delay dynamic routing problem is expressed as a *linear optimal control problem with linear state and control variable inequality constraints*. The inputs are assumed to be deterministic functions of time and a feedback solution is sought which drives all of the state variables to zero at the final time.

Little thoeretical or computational attention has been paid to the class of control problems with state variable inequality constraints and the control appearing linearly in the dynamics and performance index. In this case, the control is of the bang-bang variety and the costates may be characterized by a high degree of nonuniqueness. In [3] the necessary conditions associated with this problem are examined when the control and state constraints are both scalars, and an interesting analogy is presented between the junction conditions associated with state boundary arcs and those associated with singular control arcs. However, no computational algorithm is presented.

Perhaps the most intersting computational approach presented for the all linear problem is the mathematical programming oriented cutting plane algorithm presented in [4]. The basic algorithm consists of solving a sequence of succeedingly higher dimensional optimal control problems without state space constraints. The drawbacks to this approach are that the dimension of the augmented problem may grow unreasonably large and that even unconstrained state linear optimal control problems may be difficult to solve efficiently. In the same paper, an alternative technique is suggested whereby the problem is formulated as a large linear program via time discretization of the dynamics and the constraints. However, this technique also encounters the problem of high dimensionality when the time discretization is sufficiently fine to assure a good approximation to the continuous problem. Besides, neither of the above techniques provide explicitly for feedback solutions.

In [2] an approach is suggested, by way of a simple example, for constructing the feedback solution to the linear optimal control problem associated with message routing when all the inputs to the network are assumed to be zero. The purpose of this paper is to elaborate upon this approach by extending it to the general class of network problems with inputs which are constant in time. An algorithm is presented for the construction of the feedback solution which exploits the special structure of the problem.

We begin by presenting in Section II the model of [2] and the associated optimal control problem for closed-loop minimum delay dynamic routing.

The necessary conditions of optimality for general deterministic inputs are developed in Section III and shown to be sufficient. It is immediately seen that the costate variables may experience jumps when the associated state variables are on their boundaries and that the costates are possibly nonunique. Also, the optimal control is of the bang-bang variety and may also exhibit nonuniqueness. We subsequently restrict consideration to the case in which the *inputs are constant in time* and present a controllability condition for this situation. A special property regarding the final value of the costates is also presented.

In Section IV we define special subsets of the state space known as *feedback control regions*. Associated with each such region, in principle, is a set of controls which are optimal for all the states of the given region. Feedback control regions are shown to be convex polyhedral cones, and the goal is to construct enough of these regions to fill up the entire admissible state space. We demonstrate in Section V how this may be achieved for two simple examples, and generalize the notion in Section VI into the *constructive dynamic programming concept*. The basic idea is to utilize a certain comprehensive set of optimal trajectories fashioned backward in time from the necessary conditions in order to construct the feedback control regions. An algorithm is then presented, in conceptual form, for the realization of the constructive dynamic programming concept. Several of the basic computational techniques associated with the algorithm are presented in Appendices A and B. Discussion and conclusions are found in Section VII.

Several complicating features of the algorithm render it too difficult to compute numerically for general network problems. In [5] and a forthcoming paper by the authors it is shown that for a class of problems involving single destination networks these complicating features disappear. For this case it is possible to formulate the algorithm in a form suitable for numerical computation.

## II. THE MODEL FOR DYNAMIC ROUTING IN DATA COMMUNICATION NETWORKS

We now describe the model presented in [2]. For a network of N nodes let $N$ denote the set of nodes and $L$ the set of links. All links are taken to be simplex and $(i,k)$ denotes the link connecting node i to node k with capacity $C_{ik}$ (in units of traffic/unit time). Attention is restricted to the case in which all the inputs to the network are deterministic functions of time. The message flow dynamics are given by:

$$\dot{x}_i^j(t) = a_i^j(t) - \sum_{k \in E(i)} u_{ik}^j(t) + \sum_{\substack{\ell \in I(i) \\ \ell \neq j}} u_{\ell i}^j(t) \qquad i,j \in N,\ j \neq i \tag{1}$$

where

$x_i^j(t)$ = continuous state variable which approximates the amount of data traffic (measured in messages, packets, bits, etc.) at node i at time t whose destination is node j, $i \neq j$.

$a_i^j(t)$ = instantaneous rate of traffic input at node i at time t with destination j.

$u_{ik}^j(t)$ = control variable which represents that portion of $C_{ik}$ used at time t for messages with destination j.

$E(i)$ = collection of nodes k such that $(i,k) \in L$.

$I(i)$ = collection of nodes $\ell$ such that $(\ell,i) \in L$.

We have the positivity constraints

$$x_i^j(t) \geqslant 0 \tag{2}$$

$$u_{ik}^j(t) \geqslant 0 \tag{3}$$

and the link capacity constraints are

$$\sum_{j \neq i} u_{ik}^{j}(t) \leq C_{ik}, \qquad (i,k) \in L, \quad j \in N \; . \tag{4}$$

The goal is to empty the network of its current message storage in the presence of inputs in such a fashion as to minimize the total delay experienced by all the messages traveling through the network. Consider the cost functional

$$J = \int_{t_o}^{t_f} [\sum_{\substack{i,j \in N \\ j \neq i}} \alpha_i^j x_i^j(t)]dt \tag{5}$$

where $t_f$ is such that

$$x_i^j(t_f) = 0 \qquad i,j \in N, \; j \neq i \; . \tag{6}$$

It is demonstrated in [2] that when $\alpha_i^j = 1 \quad \forall i,j \in N, \; j \neq i$, expression (5) is exactly equal to the total delay. Priorities may be incorporated by taking the weightings $\alpha_i^j$ to be unequal.

For convenience we define the column vectors $\underline{x}$, $\underline{u}$, $\underline{a}$, $\underline{C}$ and $\underline{\alpha}$ to be consistently ordered concatenations of the state variables, control variables, inputs, link capacities and weightings respectively. In this paper we shall not be concerned with the particular ordering. Denote $n = \dim(\underline{x}) = \dim(\underline{a}) = \dim(\underline{\alpha})$, $m = \dim(\underline{u})$ and $r = \dim(\underline{C})$. Equation (1)-(6) may then be expressed in the vector form:

*Dynamics* $$\underline{\dot{x}}(t) = \underline{B}\ \underline{u}(t) + \underline{a}(t) \tag{7}$$

*Boundary Conditions* $$\underline{x}(t_o) = \underline{x}_o \ ; \ \underline{x}(t_f) = \underline{0} \tag{8}$$

*State Constraints* $$\underline{x}(t) \geqslant 0 \qquad \forall t \in [t_o, t_f] \tag{9}$$

*Control Constraints* $$U \begin{cases} \underline{D}\ \underline{u}(t) \leqslant \underline{C} \\ \\ \underline{u}(t) \geqslant \underline{0} \end{cases} \qquad \forall t \in [t_o, t_f] \tag{10}$$

*Cost Functional* $$J = \int_{t_o}^{t_f} \underline{\alpha}^T \underline{x}(t)\,dt \tag{11}$$

In (7) $\underline{B}$ is the n×m incidence matrix composed of 0's, +1's and -1's associated with the flow equations (1) and $\underline{D}$ is the r×m matrix composed of 0's and 1's corresponding to (4). We now express the linear optimal control problem with linear state and control variable inequality constraints which represents the data communication network closed-loop dynamic routing problem:

*Optimal Control Problem*

*Find the set of controls* $\underline{u}$ *as a function of time and state*

$$\underline{u}(t) \triangleq \underline{u}(t,\underline{x}) \qquad t \in [t_o, t_f] \tag{12}$$

*that brings any initial condition* $\underline{x}(t_o) = \underline{x}_o$ *to the final condition* $\underline{x}(t_f) = \underline{0}$ *and minimizes the cost functional (11) subject* *to the dynamics (7) and the state and control variable inequality constraints (9)-(10).*

Several assumptions have been made in order to facilitate the modelling and solution. These are now discussed briefly.

(i) *Continuous state variables.* Strictly speaking, the state variables are discrete with quantization level being the unit of traffic selected. The assumption is justified by recognizing that any single message contributes little to the overall behavior of the network; therefore, it is unnecessary to look individually at each of the messages and its length.

(ii) *Deterministic inputs.* Computer networks almost always operate in a stochastic user demand environment. It is suggested in [2] that the deterministic approach may take stochastic inputs into account approximately by utilizing the ensemble average rates of the inputs to generate nominal trajectories. Also, valuable insight into the stochastic situation may be gained by solving the more tractable deterministic problem.

(iii) *Centralized Controller.* This is implied by the form of the control law $\underline{u}(t) \triangleq \underline{u}(t,\underline{x})$. This assumption may be valid in the case of small networks. Also, obtaining the optimal strategy under this assumption could prove extremely useful in determining the suboptimality of certain decentralized schemes.

(iv) *Infinite capacity buffers.* Message buffers are of course of finite capacity. This may be taken into account by imposing upper bounds on the state variables, but this is not done in the current analysis.

(v) *All state variables go to zero at* $t_f$. During normal network operation the message backlogs at the nodes almost never go to zero. Our assumption may correspond to the situation in which one wishes to dispose of message backlogs for the purpose of temporarily relieving congestion locally in time.

## III. FEEDBACK SOLUTION FUNDAMENTALS

We begin by presenting the necessary conditions of optimality for the general deterministic inputs case.

Theorem 1 (Necessary Conditions)

Let the scalar functional h be defined as follows:

$$h(\underline{u}(t),\ \underline{\lambda}(t)) \triangleq \underline{\lambda}^T(t)\dot{\underline{x}}(t) = \underline{\lambda}^T(t)[\underline{B}\ \underline{u}(t) + \underline{a}(t)]. \tag{13}$$

A necessary condition for the control law $\underline{u}^*(\cdot) \in \mathcal{U}$ to be optimal for problem (7) - (12) is that it minimize h pointwise in time, namely

$$\underline{\lambda}^T(t)\underline{B}\ \underline{u}^*(t) \leqslant \underline{\lambda}^T(t)\underline{B}\ \underline{u}(t) \tag{14}$$

$$\forall \underline{u}(t) \in \mathcal{U} \qquad \forall t \in [t_o, t_f].$$

The costate $\underline{\lambda}(t)$ is possibly a discontinuous function which satisfies the following differential equation

$$-d\underline{\lambda}(t) = \underline{\alpha}dt + d\underline{\eta}(t), \quad t \in [t_o, t_f] \tag{15}$$

where componentwise $d\underline{\eta}(\tau)$ satisfies the following complementary slackness

$$\left.\begin{aligned} x_i^j(t)\,d\eta_i^j(t) &= 0 \\ d\eta_i^j(t) &\leqslant 0 \end{aligned}\right\} \quad \begin{aligned} &\forall t \in [t_o, t_f] &(16)\\ &i,j \in N, \quad j \neq i. &(17)\end{aligned}$$

The terminal boundary condition for the costate differential equation is

$$\underline{\lambda}(t_f) = \underline{\nu} \qquad \text{free} \tag{18}$$

and the transversality condition is

$$\underline{\lambda}^T(t_f)\dot{\underline{x}}(t_f) = \underline{0}. \tag{19}$$

Finally, the function $h$ is everywhere continuous, i.e.

$$h(\underline{u}(t^-), \underline{\lambda}(t^-)) = h(\underline{u}(t^+), \underline{\lambda}(t^+)) \qquad \forall t \in [t_o, t_f]. \tag{20}$$

Proof: In [6] a generalized Kuhn-Tucker theorem in a Banach space for the minimization of a differentiable function subject to inequality constraints is presented. For our problem, it calls for the formation of the Lagrangian

$$J = \int_{t_o}^{t_f} \underline{\alpha}^T \underline{x}(\tau) d\tau + \int_{t_o}^{t_f} \underline{\lambda}^T(\tau) \; [\underline{B} \; \underline{u}(\tau) + \underline{a}(\tau) - \dot{\underline{x}}(\tau)] d\tau$$

$$+ \int_{t_o}^{t_f} d\underline{\eta}^T(\tau) \underline{x}(\tau) + \underline{\nu}^T \underline{x}(t_f) \tag{21}$$

where $\underline{\eta}$ is an $n\times1$ vector adjoining the state constraints which satisfies the complementary slackness condition at optimality:

$$\int_{t_o}^{t_f} d\underline{\eta}^T(\tau) \underline{x}(\tau) = 0 \tag{22}$$

$$d\underline{\eta}(\tau) \leqslant 0 \qquad \forall \tau \in [t_o, t_f]. \tag{23}$$

The vector $\underline{\nu}$ which adjoins the final condition is an $n\times1$ vector of arbitrary constants.

For $\underline{u}^*(\cdot)$ to be optimal $\bar{J}$ must be minimized at $\underline{u}^*(\cdot)$, where $\underline{x}(\cdot)$, $\underline{x}(t_f)$ and $t_f$ are *unconstrained* and $\underline{u} \in U$. Taking the differential of $\bar{J}$ with respect to arbitrary variations of $\underline{x}(\cdot)$, $\underline{x}(t_f)$ and $t_f$ we obtain

$$d\bar{J} = \int_{t_o}^{t_f} \underline{\alpha}^T \delta\underline{x}(\tau) d\tau + \underline{\alpha}^T \underline{x}(t_f) dt_f$$

$$- \int_{t_o}^{t_f} \underline{\lambda}^T(\tau) \delta\dot{\underline{x}}(\tau) d\tau + \int_{t_o}^{t_f} d\underline{\eta}^T(\tau) \delta\underline{x}(\tau) + \underline{\nu}^T d\underline{x}(t_f) \tag{24}$$

where $\delta\underline{x}$ is the variation in $\underline{x}$ for time held fixed and

$$d\underline{x}(t_f) = \delta\underline{x}(t_f) + \dot{\underline{x}}(t_f)dt_f \tag{25}$$

is the total differential of $\underline{x}(t_f)$. We next integrate the third term of (24) by parts, substitute for $d\underline{x}(t_f)$ from (25) and take into account that $\delta\underline{x}(t_o) = \underline{0}$ to obtain

$$d\bar{J} = \int_{t_o}^{t_f} \delta\underline{x}^T(\tau)[\underline{\alpha}d\tau + d\underline{\eta}(\tau) + d\underline{\lambda}(\tau)] + [\underline{\nu}^T - \underline{\lambda}^T(t_f)]\delta\underline{x}(t_f) \tag{26}$$

$$+ [\underline{\alpha}^T\underline{x}(t_f) + \underline{\nu}^T\dot{\underline{x}}(t_f)]dt_f.$$

Now, in order for $\bar{J}$ to be stationary with respect to the free variations $\delta\underline{x}(\tau)$, $\delta\underline{x}(t_f)$ and $dt_f$ we must have

$$\underline{\alpha}d\tau + d\underline{\eta}(\tau) + d\underline{\lambda}(\tau) = \underline{0} \tag{27}$$

$$\underline{\lambda}(t_f) = \underline{\nu} \qquad \underline{\nu} \text{ free} \tag{28}$$

$$\underline{\nu}^T\dot{\underline{x}}(t_f) = \underline{\lambda}^T(t_f)\dot{\underline{x}}(t_f) = -\underline{\alpha}^T\underline{x}(t_f). \tag{29}$$

Equations (22) and (23) together with the constraints $\underline{x} > 0$ imply

$$d\eta_i^j(t)x_i^j(t) = 0 \qquad \forall t \in [t_o,t_f] \qquad i,j \in N,\ j \neq i. \tag{30}$$

If we integrate the term $\int_{f_o}^{t_f} \underline{\lambda}^T(\tau)\dot{\underline{x}}(\tau)d\tau$ by parts in equation (21) and substitute equations (22) and (27)-(29) into (21) we obtain

$$\overline{J} = \int_{t_o}^{t_f} \underline{\lambda}^T(\tau)[\underline{B}\ \underline{u}(\tau) + \underline{a}(\tau)]d\tau. \tag{31}$$

In order for $\overline{J}$ to be minimized with respect to $\underline{u}(\cdot) \in U$, the term $\underline{\lambda}^T(\tau)\ \underline{B}\ \underline{u}(\tau)$ must clearly be minimized pointwise in time, that is

$$\underline{\lambda}^T(\tau)\underline{B}\ \underline{u}^*(\tau) \leqslant \underline{\lambda}^T(\tau)\underline{B}\ \underline{u}(\tau) \qquad \forall \underline{u}(\tau) \in U,\ t \in [t_o,t_f]. \tag{32}$$

Thus, we have accounted for Equation (14), leaving only (20) to be proven.

To this end, let us assume that we have an optimal state trajectory $\underline{x}^*(t)$ and associated costate trajectory $\underline{\lambda}(t)$, $t \in [t_o,t_f]$. Then by the principle of optimality, for any fixed $\tau \leqslant t_f$, the functions $\underline{x}^*(t)$ and $\underline{\lambda}(t)$, $t \in [t_o, \tau]$, are optimal state and costate trajectories which carry the state from $\underline{x}_o$ to $\underline{x}(\tau)$. Hence, all of our previous conditions apply on $[t_o, \tau]$ with $\underline{x}(t_f) = x(\tau)$. Applying the transversality condition (29 ) at $t_f = \tau$, we obtain

$$\underline{\lambda}^T(\tau)\dot{\underline{x}}(\tau) = -\underline{\alpha}^T\underline{x}(\tau). \tag{33}$$

Since Equation (33) holds for all $\tau \in [t_o,t_f]$ and $x(\tau)$ is everywhere continuous, then $\underline{\lambda}^T(\tau)\dot{\underline{x}}(\tau)$ must be everywhere continuous. This proves Equation (20).

□

We shall now describe the behavior of the costate variables as functions of the corresponding state variables. We distinguish between the case when $x_i^j > 0$ ($x_i^j$ is said to be on an interior arc) and when $x_i^j = 0$ ($x_i^j$ is said to be on a boundary arc). When $x_i^j$ is on an *interior arc* Equation (16) implies $d\eta_i^j = 0$ and Equation (15) can be differential with respect to time to obtain

$$-\dot{\lambda}_i^j(t) = \alpha_i^j \ . \tag{34}$$

When $x_i^j$ is on a *boundary arc* its costate is possibly discontinuous, depending upon the nature of $\eta_i^j$. At points for which $\eta_i^j$ is absolutely continuous we define $\mu_i^j(t) \triangleq d\eta_i^j(t)/dt$. Differentiating (15) with respect to time and taking into account (16) and (17) we obtain:

$$-\dot{\lambda}_i^j(t) = \alpha_i^j + \mu_i^j(t) \qquad\qquad \mu_i^j(t) \leqslant 0. \tag{35}$$

On the other hand, at times when $\eta_i^j$ experiences a jump of magnitude $\Delta\eta_i^j$ we have from Equations (15)-(17) that $\lambda_i^j$ experiences the jump

$$\Delta\lambda_i^j = -\Delta\eta_i^j \geqslant 0. \tag{36}$$

It is not difficult to see that the costate vector may be non-unique for a given optimal trajectory — this is a fundamental characteristic of the state constrained problem. Previous works such as [6] have found this nonuniqueness to occur in costates corresponding to state variables which are on boundary arcs. However, due to the fact that in our case the pointwise minimization is a linear program, this nonuniqueness may also be exhibited by costates corresponding to state variables which are on interior arcs. This behavior is demonstrated in Example 3.5 of [5], pages 186-189.

In general, any trajectory which satisfies a set of necessary conditions is an extremal, and as such is merely a candidate for an optimal trajectory. Fortunately, in our problem it turns out that any such extremal trajectory is actually optimal, as is shown in the following theorem.

Theorem 2 The necessary conditions of Theorem 1 are sufficient.

Proof. Let $\underline{x}^*(t)$, $\underline{u}^*(t)$, $\underline{\lambda}(t)$ and $\eta(t)$ satisfy (7)-(10) and the necessary conditions of Theorem 1. Also let, $\underline{x}(t)$ and $\underline{u}(t)$ be *any* state and control trajectory satisfying (7)-(10). If we consider $\delta J \equiv J(\underline{x}) - J(\underline{x}^*)$ we have

$$\delta J = \int_{t_o}^{t_f} \underline{\alpha}^T(\tau)(\underline{x}(\tau) - \underline{x}^*(\tau))d\tau. \tag{37}$$

Substituting from (15) and expanding we obtain

$$\delta J = \int_{t_o}^{t_f} (-\underline{x}^T(\tau)d\underline{\lambda}(\tau) - \underline{x}^T(\tau)d\underline{\eta}(\tau) + \underline{x}^{*T}(\tau)d\underline{\lambda}(\tau) + \underline{x}^{*T}(\tau)d\underline{\eta}(\tau)). \tag{38}$$

From Equation (22)

$$\int_{t_o}^{t_f} \underline{x}^{*T}(\tau)d\underline{\eta}(\tau) = 0. \tag{39}$$

We now integrate the first and third terms on the right side of (38) by parts, take into account $\underline{x}(t_o) - \underline{x}^*(t_o) = \underline{x}(t_f) = \underline{x}^*(t_f) = \underline{0}$ and finally substitute from (7) to to obtain

$$\delta J = \int_{t_o}^{t_f} \underline{\lambda}^T(\tau)\underline{B}(\underline{u}(\tau) - \underline{u}^*(\tau))d\tau - \int_{t_o}^{t_f} \underline{x}^T(\tau)d\underline{\eta}(\tau). \tag{40}$$

But by (14)

$$\int_{t_o}^{t_f} \underline{\lambda}^T(\tau)\underline{B}(\underline{u}(\tau) - \underline{u}^*(\tau))d\tau \geqslant 0 \tag{41}$$

and since $\underline{x}(\tau) \geqslant 0$ and $d\underline{\eta}(\tau) \leqslant 0$ we have

$$\int_{t_o}^{t_f} \underline{x}^T(\tau)d\underline{\eta}(\tau) \leqslant 0. \tag{42}$$

Therefore, $\delta J \geqslant 0 \quad \forall \underline{u}(\cdot) \in \mathcal{U},\ \underline{x}(\cdot) \geqslant 0$

□

From inequality (14) of the necessary conditions we see that the optimal control function $\underline{u}^*(\cdot)$ is given at every time $\tau \in [t_o,t_f]$ by the solution to the following linear program with decision vector $\underline{u}(\tau)$:

$$\underline{u}^*(\tau) = \text{ARG} \min_{\underline{u}(\tau)\in\mathcal{U}} [\underline{\lambda}^T(\tau)\underline{B}\ \underline{u}(\tau)] \quad . \tag{43}$$

This is a fortuitous situation, since much is known about characterizing and finding solutions of linear programs in general. We know, for instance, that optimal solutions always lie on the boundary of the convex polyhedral constraint region $\mathcal{U}$. However, for the special forms of the matrices $\underline{B}$ and $\underline{D}$ which correspond to our network problem we may proceed immediately to represent explicitly the solution of the pointwise (in time) linear program. The minimization can actually be performed by considering *one link at a time*. Consider the link (i,k) and a possible set of associated controls:

$$u_{ik}^{1}, u_{ik}^{2}, \ldots, u_{ik}^{i-1}, u_{ik}^{i+1}, \ldots, u_{ik}^{N}$$

a given control variable may appear in one of the two following ways:

1) $u_{ik}^{j}$ enters into exactly two state equations:

$$\begin{aligned} \dot{x}_i^j(t) &= -u_{ik}^j(t) + \ldots + a_i^j \\ \dot{x}_k^j(t) &= +u_{ik}^j(t) + \ldots + a_k^j \end{aligned} \tag{44}$$

2) $u_{ik}^{k}$ enters into exactly one state equation:

$$\dot{x}_i^k(t) = -u_{ik}^k + \ldots + a_i^k \ . \tag{45}$$

Hence, all controls on link $(i,k)$ contribute the following terms to $\underline{\lambda}^T \underline{B}\ \underline{u}$ :

$$\sum_{j \neq i} (\lambda_k^j - \lambda_i^j) u_{ik}^j(t) \tag{46}$$

where $\lambda_k^k(t) = 0$.

The quantities which determine the optimal controls are the coefficients of the form $(\lambda_k^j(t) - \lambda_i^j(t))$ which multiply the control $u_{ik}^j$. The only situation is which it is optimal to have $u_{ik}^j$ strictly positive is if $(\lambda_k^j(t) - \lambda_i^j(t)) \leq 0$. In terms of the network, this says that it is optimal to send messages with destination $j$ from node $i$ to node $k$ at time $t$ only if the costate associated with $x_i^j$ at time $t$ is greater than or equal to that associated with $x_k^j$ at time $t$. This suggests an analogy between the frictionless flow of fluid in a network of pipes in which flow occurs from areas of higher pressure to areas of lower pressure, and the optimal flow of messages in a data communication network, in which flow occurs from nodes of "higher costate" to nodes of "lower costate". By way of analogy to pressure difference we refer to $(\lambda_k^j(t) - \lambda_i^j(t))$ as the *costate difference* which exists at time $t$ on link $(i,k)$ and is associated with destination $j$. Therefore, it is *optimal to send messages of a given destination only in the direction of a negative (or zero) costate difference.*

If the costate difference on link $(i,k)$ associated with destination $j$ is strictly negative and less then all the other costate differences on this link, then the optimal control is $u_{ik}^j = C_{ik}$ and all other controls are zero. However, when two or more costate differences on the same link are non-positive and equal the associated optimal control will not be uniquely determined. In such a situation the optimal solution set is in fact infinitely large. The actual computation of the optimal control at time $t$ requires knowledge of $\underline{\lambda}(t)$, which in turn requires knowledge of the optimal state trajectory for time greater than or equal to $t$. This is the central dilemma in the application of necessary conditions in the determination of a feedback solution. In order to overcome this difficulty, we shall *subsequently be considering only the situation in which all the inputs* $(a_i^j(t) \quad \forall i,j \in N,\ j \neq i)$ *are constant functions of time over the interval of interest* $t \in [t_0, t_f]$. From the network operation point of view, one can conceive of situations in which the inputs are regulated at constant values, such as the backlog emptying procedure mentioned in Section II. From the optimal control viewpoint, constant inputs appear to provide us with the minimum amount of structure required to characterize and construct the feedback solution with reasonable effort.

We begin the feedback solution for the constant inputs case by presenting a simple theorem which characterizes all those inputs which allow the state to be driven to zero under given link rate capacity constraints.

<u>Theorem 3</u> (Controllability to zero, constant inputs).

All initial conditions of the system (7)-(10) are controllable to zero under constant inputs if and only if

$$\underline{a} \in \mathrm{Int}(\dot{X}) \qquad (\underline{a} \in R^n,\ \dot{X} \subset R^n)$$

where

$$\dot{X} \triangleq \{\dot{\underline{x}} \mid -\dot{\underline{x}} = \underline{B}\ \underline{u} \ \text{ and } \ \underline{u} \in U\} \subset R^n$$

is the *set of feasible flows* attainable through the available controls.

Proof. See [5], pages 69-72.

We shall assume from herein that the controllability to zero condition of Theorem 3 is satisfied. The following is an easy consequence of Theorem 1 and therefore the proof is omitted.

Corollary 1 (constant inputs) There always exists an optimal solution for which the controls are piecewise constant in time and the state trajectories have piecewise constant slopes.

The solution to the constant input problem is of the *bang-bang* variety in that the optimal control switches intermittently among boundary points of $\mathcal{U}$. Also, in situations where one or more costate differences are zero or several are negative and equal, the control is termed *singular*. Under such circumstances, the optimal control is not determined uniquely. In the solution technique to be presented, this non-uniqueness will play a major role.

Owing to the bang-bang nature of the control, every optimal trajectory may be characterized by a finite number of parameters. We now present a compact set of notation for specifying these parameters:

Definition 1.

$$U(\underline{x}) \triangleq \{\underline{u}_o, \underline{u}_1, \underline{u}_2, \ldots, \underline{u}_{f-1}\}$$

and

$$T(\underline{x}) \triangleq \{t_o, t_1, \ldots, t_f\}$$

are a *sequence of optimal controls* and associated *control switch time sequence* which bring the state $\underline{x}$ optimally to $\underline{0}$ on $t \in [t_o, t_f]$, where $\underline{u}_p$ is the optimal control on $t \in [t_p, t_{p+1}]$, $p \in [0,1,\ldots,f-1]$.

An additional property of a given optimal trajectory that shall be of interest is which state variables travel on boundary arcs and over what periods of time. This information is summarized in the following definitions:

Definition 2:

$$\mathcal{B}_p \triangleq \{x_i^j \mid x_i^j(\tau) = 0 \quad , \; \tau \in [t_p, t_{p+1})\}$$

is the set of state variables traveling on boundary arcs during the application of $\underline{u}_p$ .

Definition 3:

$$\mathcal{B}(x) = \{\mathcal{B}_o, \mathcal{B}_1, \ldots, \mathcal{B}_{f-1}\}$$

is the sequence of sets $\mathcal{B}_p$ corresponding to the application of $U(\underline{x})$ on $T(\underline{x})$. $\mathcal{B}(\underline{x})$ is referred to as the *boundary sequence.*

In preparation for the development of the feedback solution we present the following corollary to Theorem 1 which narrows down the freedom of the costates at the final time indicated by necessary condition (18).

Corollary 2 (constant inputs)

If any state variable, say $x_i^k$, is strictly positive on the time interval $[t_{f-1},t_f]$ of an optimal trajectory, then $\lambda_i^k(t_f) = 0$.

Proof. Consider a specific state variable $x_i^k$ satisfying the hypothesis. By Corollary 1 we have $\dot{x}_i^k(t_f) < 0$ since $\dot{x}_i^k(\tau)$ is constant for $\tau \in [t_{f-1},t_f]$. Therefore, there must exist a directed chain of links from node $i$ to node $k$ (arbitrarily denote them by $\{(i,i+1),(i+1,i+2),\ldots(k-1,k)\}$) carrying some messages with destination $k$, that is

$$u_{i,i+1}^k(t_f) > 0, \quad u_{i+1,i+2}^k(t_f) > 0, \; \ldots \; , \; u_{k-1,k}^k(t_f) > 0.$$

We now recall that messages may only flow optimally in the direction of a non-positive costate difference. The sequence of costate values $\{\lambda_i^k(t_f), \lambda_{i+1}^k(t_f), \ldots , \lambda_{k-1}^k(t_f)\}$ must therefore be non-increasing from $i$ to $k-1$ and since $\lambda_k^k(t_f) = 0$ we must have $\lambda_{k-1}^k(t_f) \geqslant 0$. Consequently, all members of the above costate sequence are non-negative.

We now proceed to show by contradiction that $\lambda_i^k(t_f) = 0$. Suppose $\lambda_i^k(t_f) > 0$. Then the transversality condition $\sum_{i,j} \underline{\lambda}^T(t_f)\underline{\dot{x}}(t_f) = 0$ implies that there must be at least one $\dot{x}_i^\ell(t_f) < 0$ such that $\lambda_i^\ell(t_f) < 0$. But the above reasoning applied to $x_i^\ell$ implies that $\lambda_i^\ell(t_f) \geqslant 0$. Hence , a contradiction.

$\square$

## IV. GEOMETRICAL CHARACTERIZATION OF THE FEEDBACK SPACE FOR CONSTANT INPUTS

Our solution to the feedback control problem shall be based upon the *construction* of regions in the admissible state space to each of which we associate a feasible control (controls) which is optimal within that region. The set of such regions to be constructed will cover the entire admissible state space, and therefore the set of associated optimal controls will comprise the feedback solution. In order to assist in the systematic construction of these regions, we focus attention on regions with the following property: when we consider every point of a particular region to be an initial condition of the optimal control problem, a common optimal control sequence and a common associated boundary sequence apply to all points within that region. Formally, we define the following subset of $\mathbb{R}^n$:

Definition 4: A set $\mathcal{R}, \mathcal{R} \subset \mathbb{R}^n$, is said to be a *feedback control region* with control set $\Omega, \Omega \subset \mathcal{U}$, if the following properties hold:

(i) Consider any two points $\underline{x}_1$, $\underline{x}_2 \in \mathrm{Int}(\mathcal{R})$. Suppose $U(\underline{x}_1) = U$ with associated switch time set $T(\underline{x}_1)$. Then $U(\underline{x}_2) = U$ for some switch time set $T(\underline{x}_2)$.

(ii) $\mathcal{B}(\underline{x}_1) = \mathcal{B}(\underline{x}_2)$ .

(iii) Any control $\underline{u} \in \Omega$ that keeps the state inside $\mathcal{R}$ for a non-zero interval of time is an optimal control and there exists at least one such control.

A fundamental geometrical characterization of feedback control regions may be deduced directly from the necessary conditions. This interesting characterization, which shall subsequently be shown to be very useful, is given by the following theorem.

Theorem 4: The feedback control regions of Definition 4 are convex polyhedral cones in $\mathbb{R}^n$.

Proof. See [5], page 114.

Note that Theorem 4 applies for arbitrary matrices $\underline{B}$ and $\underline{D}$, not only those special to our network model.

## V. EXAMPLES OF THE BACKWARD CONSTRUCTION OF THE FEEDBACK SPACE

A basic observation with regard to feedback control regions is that they are functions of the entire future sequence of controls which carry any member state optimally to zero. This general dependence of the current policy upon the future is the basic dilemma in computing optimal controls. This problem is often accommodated by the application of the principle of dynamic programming, which seeks to determine the optimal control as a function of the state by working backward from the final time. The algorithm to be developed employs the spirit of dynamic programming to enable construction of feedback control regions from an appropriate set of optimal trajectories run backward in time. These trajectories are fashioned to satisfy the necessary and sufficient conditions of Theorem 1, as well as the costate boundary condition at $t_f$ given in Corollary 2.

We motivate the backward construction technique with several two dimensional examples which introduce the basic principles involved.

Example 1

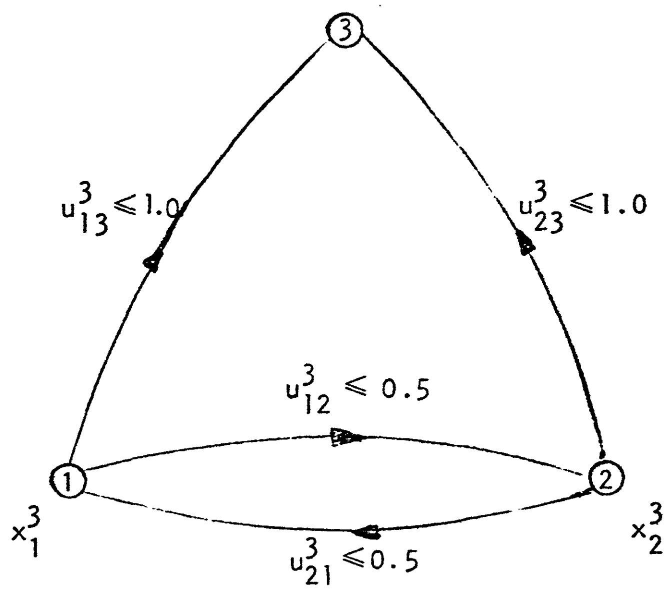


Figure 1. Simple Single Destination Network

The network as pictured in Figure 1 has a single destination, node 3; hence, we can omit the destination superscript "3" from the state and control variables without confusion. For simplicity, we assume that the inputs to the network are zero, so that the dynamics are:

$$\begin{aligned} \dot{x}_1(t) &= -u_{13}(t) - u_{12}(t) + u_{21}(t) \\ \dot{x}_2(t) &= -u_{23}(t) + u_{12}(t) - u_{21}(t) \end{aligned} \tag{47}$$

with control constraints as indicated in Figure 1. The cost function is the total delay

$$D = \int_{t_o}^{t_f} \{x_1(t) + x_2(t)\}dt. \tag{48}$$

Let the vector notation be

$$\underline{x} \triangleq \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \qquad \underline{u} \triangleq \begin{pmatrix} u_{12} \\ u_{21} \\ u_{13} \\ u_{23} \end{pmatrix}.$$

We wish to find the optimal control which drives any state $\underline{x}(t_o) \geq \underline{0}$ to $\underline{x}(t_f) = \underline{0}$ while minimizing $D$.

As our intent is to work backward from the final time, we consider all possible situations which may occur over the final time interval $[t_{f-1},t_f]$ with respect to the state variables $x_1$ and $x_2$ :

(i) $\dot{x}_1(\tau) < 0, \quad x_2(\tau) = \dot{x}_2(\tau) = 0, \quad \tau \in [t_{f-1},t_f].$

This situation is depicted in Figure 2. We begin by considering the time period $[t_{f-1}, t_f]$ in a general sense without actually fixing the switching time $t_{f-1}$. This is simply the time period corresponding to the final bang-bang optimal control which brings the state to zero with $\dot{x}_1(t_f) < 0$ and $x_2(t_f) = \dot{x}_2(t_f) = 0$. We now set out to find if there is a costate satisfying the necessary conditions for which this situation is optimal; and if so, to find the value of the optimal control. The linear program to be solved on $\tau \in [t_{f-1}, t_f]$ is

$$\underline{u}^*(\tau) = \underset{\underline{u} \in U}{\text{ARG MIN}} \, [\lambda_1(\tau)\dot{x}_1(\tau) + \lambda_2(\tau)\dot{x}_2(\tau)]$$

$$= \underset{\underline{u} \in U}{\text{ARG MIN}} \, [\, (\lambda_2(\tau) - \lambda_1(\tau))u_{12}(\tau) + (\lambda_1(\tau) - \lambda_2(\tau))u_{21}(\ )$$

$$- \lambda_1(\tau)u_{13}(\tau) - \lambda_2(\tau)u_{23}(\tau)]. \tag{49}$$

Now, the stipulation $\dot{x}_1 < 0$ tells us from Corollary 2 that

$$\lambda_1(t_f) = 0 \tag{50}$$

and since $x_1$ is on an interior arc, Equation (34) gives

$$\dot{\lambda}_1(\tau) = -1 \qquad \tau \in [t_{f-1}, t_f] \,. \tag{51}$$

This is shown in Figure 2. Now, since we specify $x_2 = 0$ on this interval, its costate equation is

$$-d\lambda_2(\tau) = 1 \, d\tau + d\eta_2(\tau) \tag{52}$$

$$d\eta_2(\tau) \leqslant 0$$

$$\lambda_2(t_f) = \nu_2 \quad \text{free} \qquad \tau \in [t_{f-1}, t_f] \;,$$

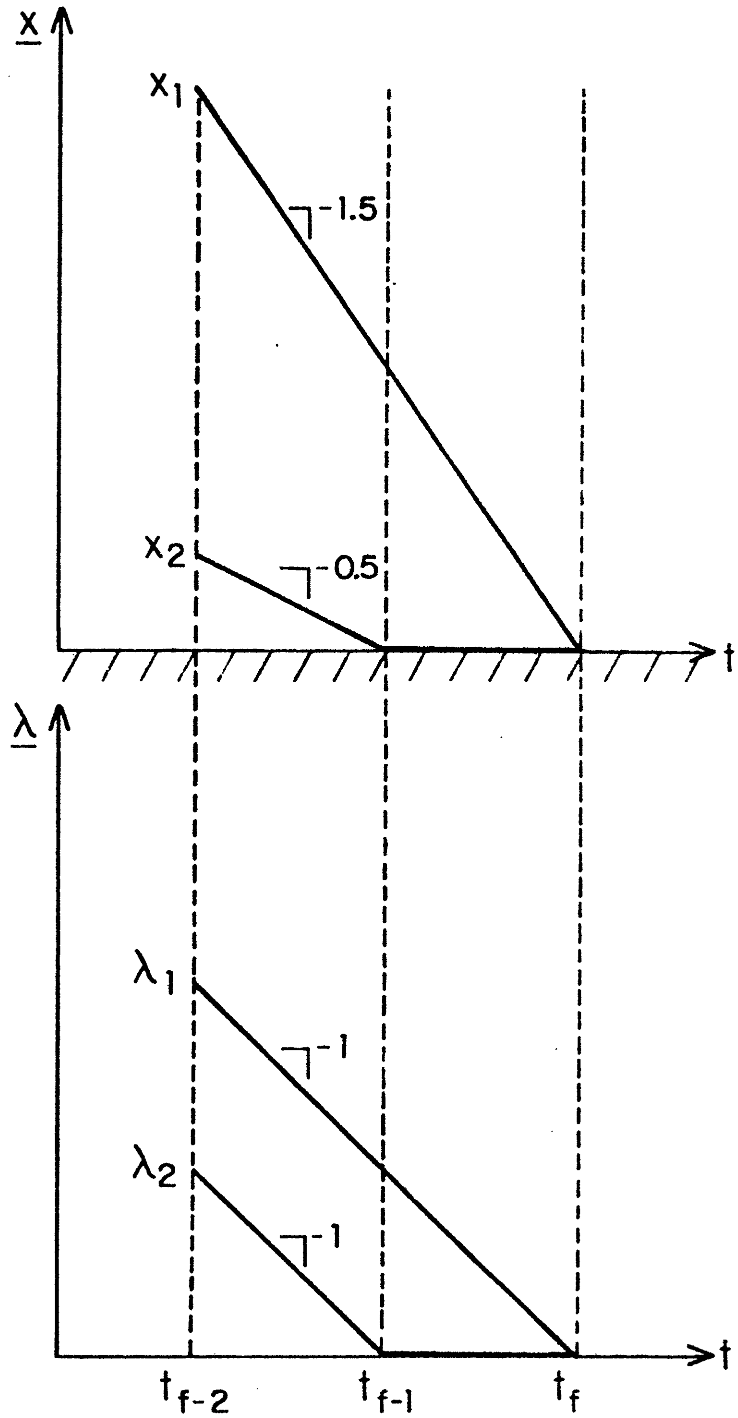


Figure 2 : State - costate Trajectory Pair for Example 1 , Case (i)

where $\eta_2$ is a possibly discontinuous function. We now submit that the costate value $\lambda_2(\tau) = \dot{\lambda}_2(\tau) = 0$, $\tau \in [t_{f-1}, t_f]$, satisfies the necessary conditions and is such that there exists an optimal solution for which $\dot{x}_1 < 0$ and $\dot{x}_2 = 0$. Firstly, the final condition $\lambda_2(t_f) = 0$ is acceptable since the necessary conditions leave $\lambda_2(t_f)$ entirely free; also, the choice of $d\eta_2(\tau) = -d\tau$ gives $\dot{\lambda}_2(\tau) = 0$ through Equation (52). Now, the reader may readily verify that $\lambda_2(\tau) = 0$, $\tau \in [t_{f-1}, t_f]$ is the only possible value which allows $\dot{x}_2(\tau) = 0$ optimally since $\lambda_2(\tau) > 0$ and $\lambda_2(\tau) < 0$ necessarily imply that $\dot{x}_2(\tau) < 0$ and $\dot{x}_2(\tau) > 0$ respectively. With the costates so determined, one solution to (49) is

$$\underline{u}(\tau) = (0.5, 0, 1.0, 0.5)^T \tag{53}$$

$$\dot{x}_1(\tau) = -1.5 \quad ; \quad \dot{x}_2(\tau) = 0$$

$$\tau \in [t_{f-1}, t_f].$$

We emphasize that the above solution is only one among an infinite set of solutions to (49). However, it is the solution which we are seeking. We now make an important observation regarding this solution. Since $\dot{\lambda}_1(\tau) = -1$ and $\dot{\lambda}_2(\tau) = 0$ for $\tau \in [t_{f-1}, t_f]$, the control (53) remains optimal on $\tau \in (-\infty, t_f]$. But as $t_{f-1} \to -\infty$, $x_1(t_{f-1}) \to \infty$. Thinking now in forward time, this implies that any initial condition on the $x_1$- axis can be brought to zero optimally with the control specified in (53). Therefore, the $\underline{x}_1$ axis is a feedback control region in the sense of Definition 4 for which we have:

$$R = \{\underline{x} \mid x_2 = 0\}$$

where

$$U = \{(0.5,\ 0,\ 1.0,\ 0.5)^T\} \tag{54}$$

$$B = \{\{x_2\}\}$$

$$\Omega = (0.5,\ 0,\ 1.0,\ 0.5)^T.$$

*We have therefore determined the optimal feedback control for all points on the* $x_1$*-axis.* This is indicated in Figure 3.

Suppose now that we wish to consider a more general class of trajectories associated with the end condition under discussion. What we may do is to temporarily fix $t_{f-1}$ and stipulate that the control on $[t_{f-2},t_{f-1})$ has $\dot{x}_2$ negative; that is, insist that $x_2$ "leave the boundary" backward in time. As before, the initial time $t_{f-2}$ of the segment $[t_{f-2},t_{f-1})$ is left free. The program to be solved is (49) with $\tau \in [t_{f-2},t_{f-1})$. Now, since $x_1$ is on an interior arc across $t_{f-1}$, by (34) its costate must be continuous across $t_{f-1}$, that is

$$\lambda_1(t_{f-1}^-) = \lambda_1(t_{f-1}^+) = t_f - t_{f-1}. \tag{55}$$

Since (52) allows for only positive jumps of $\lambda_2$ forward in time, we have

$$\lambda_2(t_{f-1}^-) = \lambda_2(t_{f-1}^+) = 0\ . \tag{56}$$

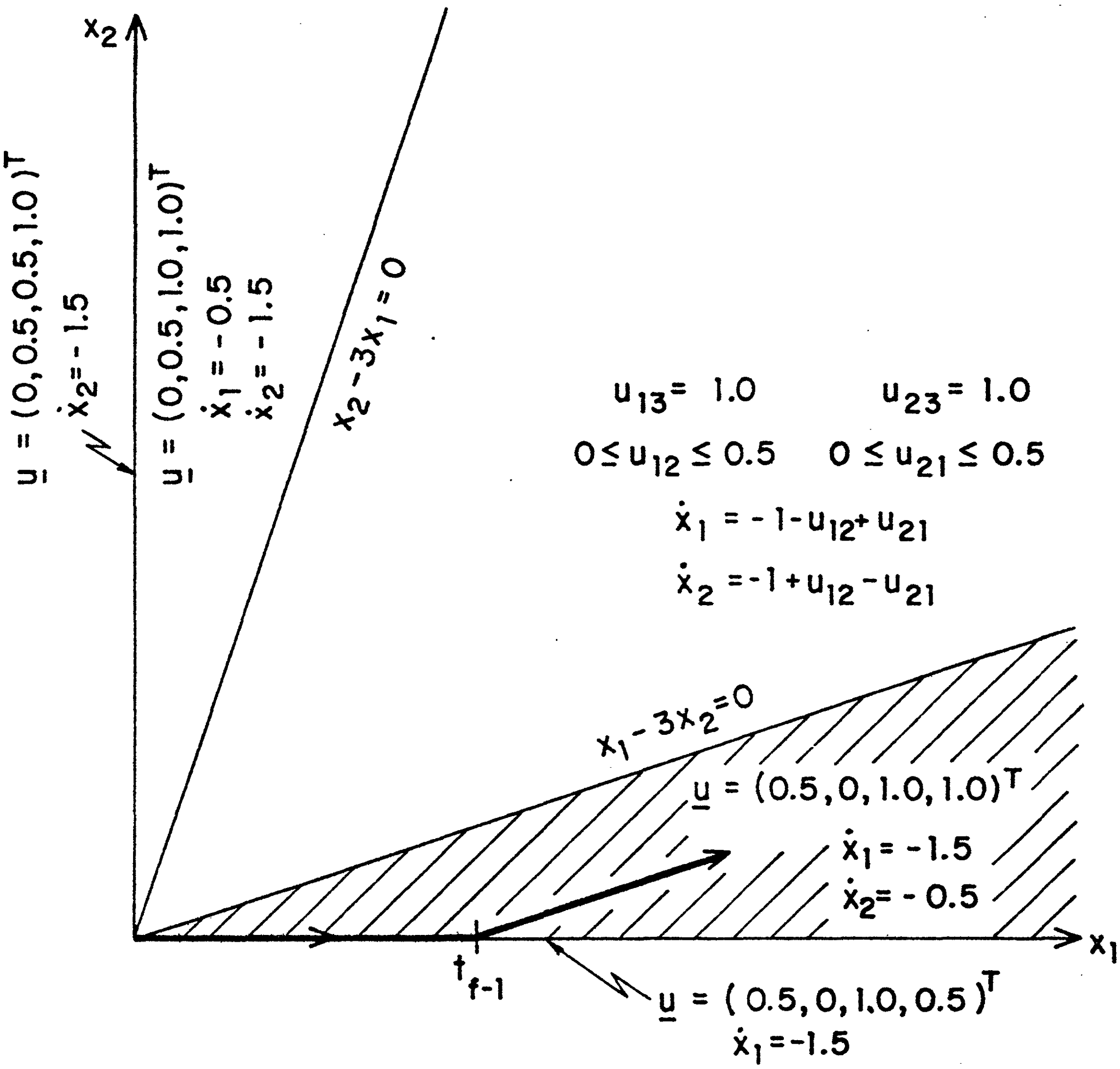


Figure 3 : Feedback Solution for Example 1

Also, since both $x_1$ and $x_2$ are on interior arcs on $[t_{f-2}, t_{f-1})$, Equation (34) gives

$$\left.\begin{aligned} \dot{\lambda}_1(\tau) &= -1 \\ \dot{\lambda}_2(\tau) &= -1 \end{aligned}\right\} \quad \tau \in [t_{f-2}, t_{f-1}) \ . \tag{57}$$

The resultant costate trajectory is depicted in Figure 2. We now perform the minimization (49) for $\tau \in [t_{f-2}, t_{f-1})$. Since $\lambda_1(\tau) > \lambda_2(\tau) > 0$, $\tau \in [t_{f-2}, t_{f-1})$, the solution is

$$\underline{u}(\tau) = (0.5, 0, 1.0, 1.0)^T \tag{58}$$

so that

$$\dot{x}_1(\tau) = -1.5 \ ; \qquad \dot{x}_2(\tau) = -0.5. \tag{59}$$

Therefore, the optimal control gives $\dot{x}_2(\tau) < 0$, which is the situation which we desire. Once again, we see that the control is optimal for $\tau \in (-\infty, t_{f-1}]$. Since $\dot{x}_1/\dot{x}_2 = 3$, upon leaving the $x_1$ axis backward in time the state travels parallel to the line $x_1 - 3x_2 = 0$ forever. Now, recall that $t_{f-1}$ is essentially free. Therefore, from anywhere on the $x_1$ axis the state leaves parallel to $x_1 - 3x_2 = 0$ with underlying optimal control (58). Thinking now in forward time, this implies that any initial condition lying in the region between the line $x_1 - 3x_2 = 0$ and the $x_1$-axis (not including the $x_1$-axis) may be brought optimally to the $x_1$-axis with the control (58). See Figure 3. Once the state reaches the $x_1$-axis, the optimal control which subsequently takes the state to zero is given by (53).

Based upon this logic we may now readily construct the following feedback control region:

$$\mathcal{R} = \{\underline{x} \mid 0 < x_2 \leqslant \frac{x_1}{3} \}$$

where

$$U = \{(0.5,\ 0,\ 1.0,\ 1.0)^T,\ (0.5,\ 0,\ 1.0,\ 0.5)^T\}$$

$$\mathcal{B} = \{\{\emptyset\},\ \{x_2\}\}$$

$$\Omega = (0.5,\ 0,\ 1.0,\ 1.0)^T \tag{60}$$

With the two feedback control regions just constructed we have managed to fill out the region $\{\underline{x} \mid 0 \leqslant x_2 \leqslant \frac{x_1}{3}\}$ with optimal controls.

(ii) $\dot{x}_2(\tau) < 0,\quad x_1(\tau) = \dot{x}_1(\tau) = 0, \qquad \tau \in [t_{f-1}, t_f).$

This situation is the same as (i) with the roles of $x_1$ and $x_2$ simply reversed. If we let $x_2$ leave the boundary first backward in time, we may construct a feedback control region consisting of the $x_2$-axis in a fashion analogous to that of (i). If we subsequently allow $x_1$ to leave the boundary backward in time, we may construct the feedback control region $\{\underline{x} \mid 0 < x_1 \leqslant \frac{x_2}{3} \}$. These regions and associated optimal controls are illustrated in Figure 3.

(iii) $\dot{x}_1(\tau) < 0,\quad \dot{x}_2(\tau) < 0, \qquad \tau \in [t_{f-1}, t_f].$

We are considering the situation in which both states go to zero at $t_f$. Since $x_1$ and $x_2$ are on interior arcs over this time interval, Corollary 2 gives

$$\lambda_1(t_f) = \lambda_2(t_f) = 0 \tag{61}$$

and from Equation (34)

$$\dot{\lambda}_1(\tau) = \dot{\lambda}_2(\tau) = -1 \qquad \tau \in [t_{f-1}, t_f]. \tag{62}$$

Hence, the costates are always equal over this time interval. The solution to the linear program (49) on $[t_{f-1}, t_f]$ is:

$$u_{13}(\tau) = 1.0 \qquad u_{23}(\tau) = 1.0 \tag{63}$$

$$0 \leqslant u_{12}(\tau) \leqslant 0.5 \qquad 0 \leqslant u_{21}(\tau) \leqslant 0.5$$

so that

$$\dot{x}_1(\tau) = -1.0 - u_{12}(\tau) + u_{21}(\tau) \tag{64}$$

$$\dot{x}_2(\tau) = -1.0 + u_{12}(\tau) - u_{21}(\tau) \; .$$

In this situation we have encountered non-uniqueness of the optimal control which we seek. The optimal values of $u_{12}$ and $u_{21}$ are completely arbitrary within their constraints. The optimal directions with which the state leaves the origin backward in time at $t_f$ lie between $\dot{x}_1/\dot{x}_2 = 3$ and $\dot{x}_2/\dot{x}_1 = 3$, that is, between the lines $x_1 - 3x_2 = 0$ and $x_2 - 3x_1 = 0$. Moreover, for any $\tau \in (\infty, t_f]$ the entire set of controls and associated directions in the state space remain optimal. As before, we now translate this information to forward time and recognize that for any point lying between the lines $x_1 - 3x_2 = 0$ and $x_2 - 3x_1 = 0$ (not including these lines) the complete set of controls (63) is optimal. Therefore, we may construct the following feedback control region ($u_i = [a,b]$ means that any value of $u_i$ between a and b is optimal):

$$R = \{\underline{x} \mid \frac{x_2}{3} \leq x_1 \leq 3x_2 \} \tag{65}$$

where

$$U = \{([0,0.5], [0,0.5], 1.0, 1.0)^T\}$$

$$B = \{\{x_1, x_2\}\}$$

$$\Omega = \{([0,0.5],[0,0.5], 1.0, 1.0)^T\}$$

This region is illustrated in Figure 3.

Having completed all three cases in this fashion we have filled up the entire state space with feedback control regions. The specification of the optimal feedback control is therefore complete.

☐ Example 1.

<u>Example 2.</u> The network is the same as for Example 1, but the cost functional is taken as the weighted delay

$$J = \int_{t_o}^{t_f} \{2x_1(t) + x_2(t)\}dt. \tag{66}$$

As in Example 1, we take the approach of working backward from the final time, beginning with the three possible situations which may occur at that time.

$$\text{(i)} \qquad \dot{x}_1(\tau) < 0, \quad x_2(\tau) = \dot{x}_2(\tau) = 0, \qquad \tau \in [t_{f-1}, t_f] \ .$$

The linear program to be solved over the final time interval $\tau \in [t_{f-1}, t_f]$ is (49) with $\lambda_1(\tau)$ and $\lambda_2(\tau)$ appropriately determined. The final condition (50) applies, but since the weighting on $x_1$ is $\alpha_1 = 2$, the appropriate differential equation for $\lambda_1$ is

$$\dot{\lambda}_1(\tau) = -2 \qquad \tau \in [t_{f-1}, t_f] \ . \tag{67}$$

Now, $\lambda_2(\tau)$ is determined in the same fashion as in case (i) of Example 1. That is, the value

$$\lambda_2(\tau) = \dot{\lambda}_2(\tau) = 0 \qquad \tau \in [t_{f-1}, t_f] \tag{68}$$

allows the solution to (49) to be such that $\dot{x}_2(\tau) = 0$, $\tau \in [t_{f-1}, t_f]$. Consequently, the optimal control (54) applies here. The feedback control region on the $x_1$-axis is therefore the same as (54). See Figure 4.

Let us now allow $x_2$ to leave the boundary backward in time at some time $t_{f-1}$. In this case we have

$$\lambda_1(t_{f-1}^-) = \lambda_1(t_{f-1}^+) = 2(t_f - t_{f-1}) \tag{69}$$

$$\lambda_2(t_{f-1}^-) = \lambda_2(t_{f-1}^+) = 0.$$

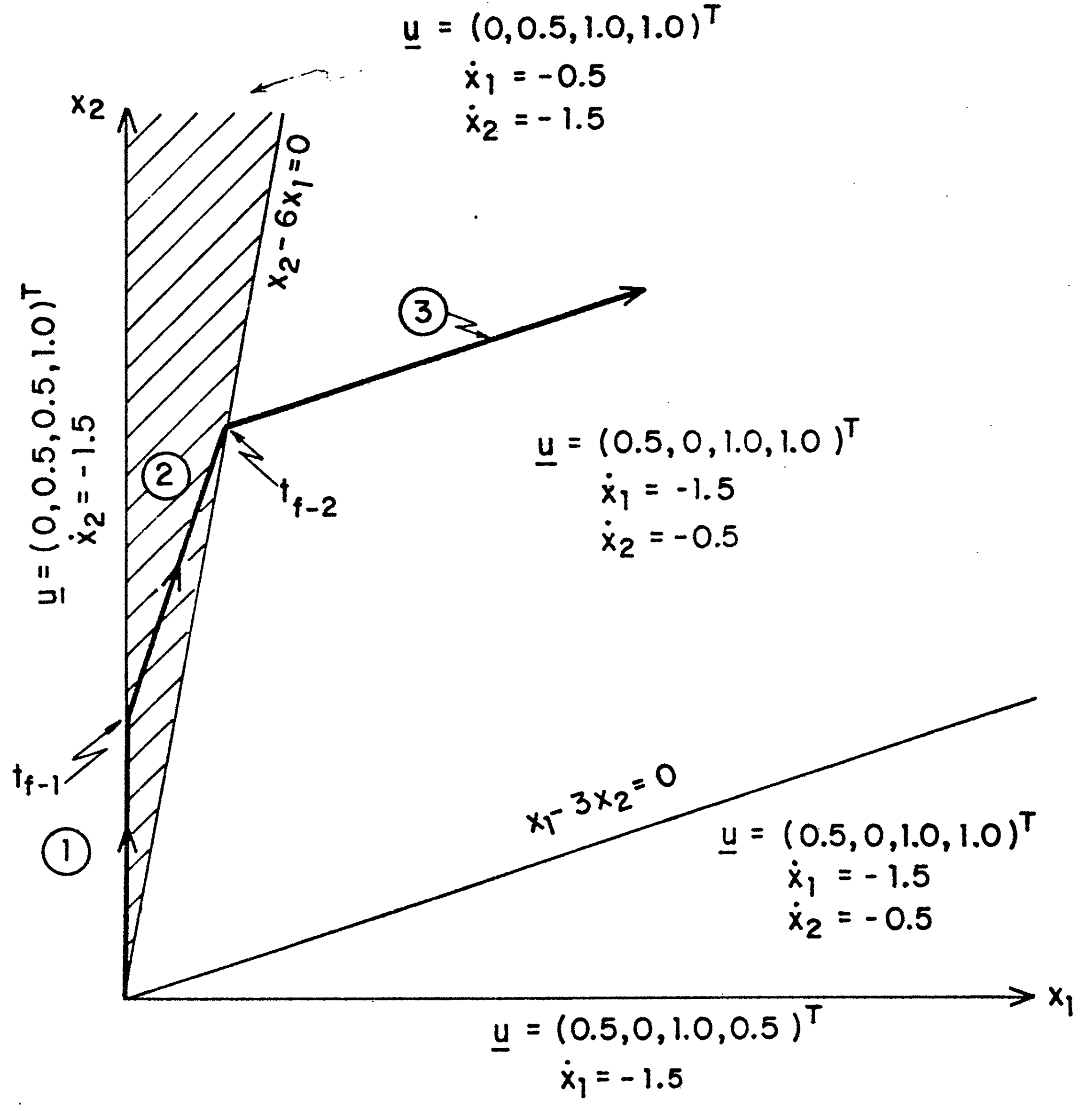


Figure 4 : Feedback Solution for Example 2

Since both $x_1$ and $x_2$ are on interior arcs over this interval, their differential equations are

$$\left.\begin{aligned} \dot{\lambda}_1(\tau) &= -2 \\ \dot{\lambda}_2(\tau) &= -1 \end{aligned}\right\} \quad \tau \in [t_{f-2}, t_{f-1}) \ . \tag{70}$$

Also, as before, all that matters in the solution of the linear program is that $\lambda_1(\tau) > \lambda_2(\tau) > 0$, $\tau \in [t_{f-2}, t_{f-1})$. Therefore, the solution is given by (58) and the feedback control region $\{\underline{x} \mid 0 < x_2 \leq \frac{x_1}{3}\}$ is as specified in (60). See Figure 4.

(ii) $\dot{x}_2(\tau) < 0$, $x_1(\tau) = \dot{x}_1(\tau) = 0$, $\tau \in [t_{f-1}, t_f]$.

The details of this situation are depicted in Figure 5.

We know from Corollary 2 that

$$\lambda_2(t_f) = 0 \tag{71}$$

and from (34) that

$$\dot{\lambda}_2(\tau) = -1 \qquad \tau \in [t_{f-1}, t_f]. \tag{72}$$

We now may find by the process of elimination that the only value of $\lambda_1(\tau)$, $\tau \in [t_{f-1}, t_f]$ for which $\dot{x}_1 = 0$ is optimal is:

$$\lambda_1(\tau) = \dot{\lambda}_1(\tau) = 0 \qquad \tau \in [t_{f-1}, t_f]. \tag{73}$$

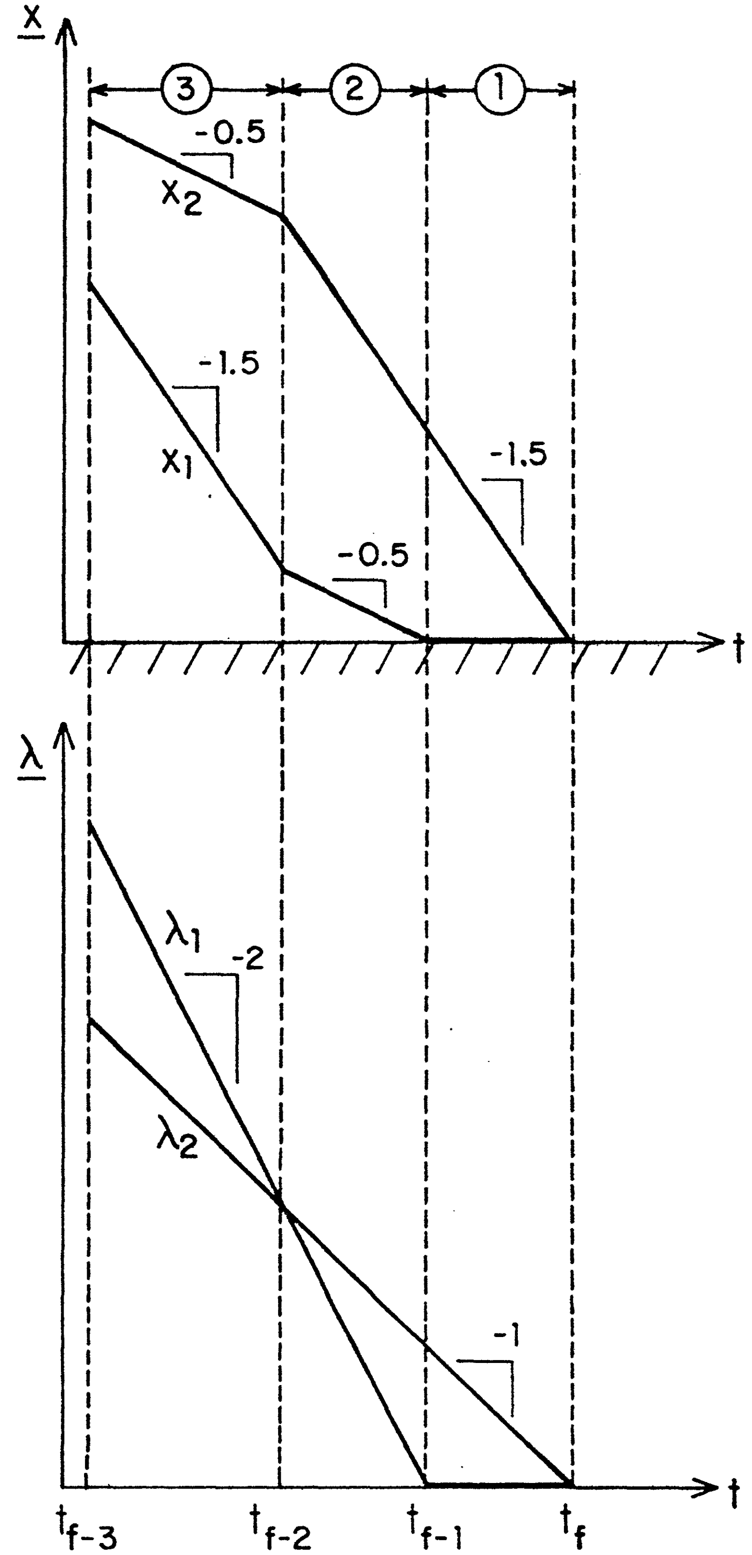


Figure 5: State - Costate Trajectory Pair for Example 2, Case (ii)

It is easily shown that $\lambda_1(\tau)$ as given in (73) satisfies the necessary conditions. Therefore, the solution to (49) is the same as in Example 1, case (ii), and the feedback control region on the $x_2$-axis is assigned in identical fashion. See Figure 4.

As the next step, we now stipulate that $x_1$ leaves the boundary backward in time at $t_{f-1}$. Since $x_2(t_{f-1}) > 0$

$$\lambda_2(t_{f-1}^-) = \lambda_2(t_{f-1}^+) = t_f - t_{f-1}. \tag{74}$$

Since costate jumps can only be positive in forward time, we must have

$$\lambda_1(t_{f-1}^-) = 0. \tag{75}$$

Also, since $x_1(\tau) > 0$, $x_2(\tau) > 0$, $\tau \in [t_{f-2}, t_{f-1})$,

$$\left.\begin{aligned} \dot{\lambda}_1(\tau) &= -2 \\ \dot{\lambda}_2(\tau) &= -1 \end{aligned}\right\} \quad \tau \in [t_{f-2}, t_{f-1}). \tag{76}$$

See Figure 5. We now notice a fundamental difference between this and the previous situations. *At some time before* $t_{f-1}$ *the sign of* $(\lambda_1(\tau) - \lambda_2(\tau))$ *changes, which imples that the solution to the linear program changes at that time.* Therefore, $t_{f-2}$ is not allowed to run to $-\infty$, but is actually the time at which the costates cross and the control switches. The optimal controls and state velocities on either side of the switch are:

$\tau \in [t_{f-2}, t_{f-1})$:

$$\underline{u} = (0,\ 0.5,\ 1.0,\ 1.0)^T \tag{77}$$

$$\dot{x}_1 = -0.5 \quad ; \quad \dot{x}_2 = -1.5. \tag{78}$$

$\tau \in [t_{f-3}, t_{f-2})$ :

$$\underline{u} = (0.5,\ 0,\ 1.0,\ 1.0)^T \tag{79}$$

$$\dot{x}_1 = -1.5 \quad ; \quad \dot{x}_2 = -0.5. \tag{80}$$

The relationship between the states $x_1$ and $x_2$ at $t_{f-2}$ may be calculated as follows:

$$\begin{aligned} \lambda_1(t_{f-2}) &= \lambda_1(t_{f-1}) + 2(t_{f-2} - t_{f-1}) \\ \lambda_2(t_{f-2}) &= \lambda_2(t_{f-1}) + (t_{f-2} - t_{f-1}) \end{aligned} \tag{81}$$

but

$$\begin{aligned} \lambda_1(t_{f-1}) &= 0 \\ \lambda_2(t_{f-1}) &= (t_{f-1} - t_f) \ . \end{aligned} \tag{82}$$

The crossing condition $\lambda_1(t_{f-2}) = \lambda_2(t_{f-2})$ implies from (81) and (82) that

$$t_{f-2} - t_{f-1} = t_{f-1} - t_f \ . \tag{83}$$

Now

$$x_1(t_{f-2}) = x_1(t_{f-1}) + 0.5(t_{f-2} - t_{f-1})$$
$$x_2(t_{f-2}) = x_2(t_{f-1}) + 1.5(t_{f-2} - t_{f-1}) \tag{84}$$

but

$$x_1(t_{f-1}) = 0.0$$
$$x_2(t_{f-1}) = 1.5(t_{f-1} - t_f). \tag{85}$$

Finally, (83) and (84) give

$$x_2(t_{f-2}) - 6x_1(t_{f-2}) = 0. \tag{86}$$

That is, the switch of control corresponding to the time $t_{f-2}$ always occurs when the state reaches the line (86). Therefore, backward in time the state leaves from anywhere on the $x_2$ axis with optimal control (77) and associated rate (78). The direction of travel is actually parallel to the line $x_2 - 3x_1 = 0$. Upon reaching the line $x_2 - 6x_1 = 0$, the optimal control switches to (79) and the state travels parallel to the line $x_1 - 3x_2 = 0$ forever. This sequence is illustrated for a sampled trajectory whose portions are labeled 1 , 2 , 3 in Figures 4 and 5.

From these observations, the following may be inferred by thinking in forward time: *The control (77) is optimal anywhere within the region* *bounded by the* $x_2$*-axis and the line* $x_2 - 6x_1 = 0$*, not including the* $x_2$*-axis (shaded in Figure 4). The control (79) is optimal anywhere within the region bounded by the lines* $x_2 - 6x_1 = 0$ *and* $x_1 - 3x_2 = 0$ *not including the former line.* This region is also indicated in Figure 4. Therefore, we can construct the following two feedback control regions:

$$R = \{\underline{x} \mid 0 < x_1 < \frac{x_2}{6}\} \tag{87}$$

where

$$U = \{(0, 0.5, 1.0, 1.0)^T, (0, 0.5, 0.5, 1.0)^T\}$$

$$B = \{\{\emptyset\}, \{x_2\}\}$$

$$\Omega = (0, 0.5, 1.0, 1.0)^T$$

and

$$R = \{\underline{x} \mid \frac{x_2}{6} < x_1 < 3x_2\} \tag{88}$$

where

$$U = \{(0.5, 0, 1.0, 1.0)^T, (0, 0.5, 1.0, 1.0)^T, (0, 0.5, 0.5, 1.0)^T\}$$

$$B = \{\{\emptyset\}, \{\emptyset\}, \{x_2\}\}$$

$$\Omega = (0.5, 0, 1.0, 1.0)^T \ .$$

Since the entire state space has now been filled up with feedback control regions, the specification of the feedback solution is now complete.

□ Example 2.

We now summarize the contents of the preceding examples. By starting at the final time $t_f$ we have allowed state variables to leave the boundary $\underline{x} = 0$ backward in time and have computed the corresponding optimal trajectories as time runs to minus infinity. In the instances when the optimal control did not switch, we were able to construct one feedback control region. When the optimal control did switch, as in case (ii) of Example 2, two adjacent feecback control regions were constructed. By considering enough cases we were able to fill up the entire state space with feedback control regions, thus providing the feedback solution.

Note that all we need for the final specification of the feedback solution are the geometrical descriptions of the feedback control region ($\mathcal{R}$'s) and their associated optimal control sets ($\Omega$'s). The sequences of optimal controls (U's) and the boundary sequences ($\mathcal{B}$'s) are involved in an intermediate fashion.

## VI. THE CONSTRUCTIVE DYNAMIC PROGRAMMING ALGORITHM

The examples of the previous section suggest an approach by which the feedback solution to the constant inputs problem may be synthesized in general:

*The Constructive Dynamic Programming Concept.*

*Construct a set of backward optimal trajectories, each starting at the final time* $t_f$ *and running to* $t = -\infty$*, among which all possible sequences of state variables leaving the boundary backward in time, both singly and in combination, are represented. Each segment of every optimal trajectory (where a segment is that portion which occurs on the time interval between two successive switch times* $t_p$ *and* $t_{p+1}$*, not including* $t_{p+1}$*) is utilized in the construction of a feedback control region with associated optimal control set. These feedback control regions are convex polyhedral cones, and the union of all such regions is the entire admissible state space.*

The conceptual structure of an algorithm which realizes the constructive dynamic programming concept is now presented. Due to several complicating features the algorithm as it is presented here is not in a form suitable for numerical computation. Instead, it serves as a framework for the development of numerical schemes for special simplifying situations. First, we present some shorthand notation:

Definition 5: $\mathcal{I}_p \triangleq \{x_i^j \mid x_i^j(\tau) > 0,\ \tau \in [t_p, t_{p+1})\}$ is the set of state variables traveling on interior arcs on $[t_p, t_{p+1})$.

Definition 6: $\mathcal{L}_p \triangleq \{x_i^j \mid x_i^j \in \mathcal{B}_p$ and $x_i^j$ is designated to leave the boundary backward in time at $t_p\}$.

Definition 7: $\sigma_p \triangleq$ cardinality of $\mathcal{I}_p$.
$\rho_p \triangleq$ cardinality of $\mathcal{L}_p$.

Definition 8: $R_p$ = the feedback control region constructed from the optimal trajectories on the segment $[t_p, t_{p+1})$.

The algorithm is characterized by the recursive execution of a basic *step* in which one or more feedback control regions are constructed from a previously constructed feedback control region of lower dimension. To describe a single recursive step of the algorithm we begin with the feedback control region $R_p$ which has been constructed in a previous step. On the current backward optimal trajectories the state variables of $\mathcal{I}_p$ are on interior arcs and those of $\mathcal{B}_p$ are on boundary arcs. Hence, $R_p \subset \mathbb{R}^{\sigma_p}$, where we assume that $\sigma_p < n$. The basic action of each step of the algorithm is to allow a subset $\mathcal{L}_p$ of state variables in $\mathcal{B}_p$ to leave the boundary backward in time simultaneously; that is, allow the state trajectory to leave $R_p \subset \mathbb{R}^{\sigma_p}$ and travel directly into $\mathbb{R}^{\sigma_p+\rho_p}$. The set of state variables which are subsequently on interior arcs is $\mathcal{I}_{p-1}$.

In order to formulate the algorithm we must make the following assumption: *it is optimal for all of the state variables in* $I_{p-1}$ *to remain off of the boundary as time runs to minus infinity.* This is equivalent to assuming that once a state variable reaches the boundary in forward time it is always optimal for it to remain on the boundary. This assumption is certainly not always valid, and a counter-example is presented in Example 3.7 of [5], p.197. The most general class of problems for which this assumption holds is not currently known. However, in [5], p.263, it is shown to be valid for the specific class of single destination network problems with all unity weightings in the cost functional. We now provide the rule which stipulates the complete set of steps which is to be executed with respect to $R_p$:

*Consider all of the subsets of* $B_p$ *which are combinations of its elements taken* $1, 2, \ldots, n-\sigma_p$ *at a time. Steps are to be executed for* $\mathcal{L}_p$ *equal to each one of the subsets so determined, or a total of* $2^{n-\sigma_p} - 1$ *steps.*

We now describe a single step of the algorithm by choosing a particular $\mathcal{L}_p \subset B_p$. Figure 6 is used to illustrate this description.

STEP OF THE ALGORITHM

- <u>Operation 1</u> Partition $R_p$ into *subregions* with respect to $\mathcal{L}_p$. The definition of subregion is deferred until Operation 3 since notions are required which are developed in the interim. Subregions, like feedback

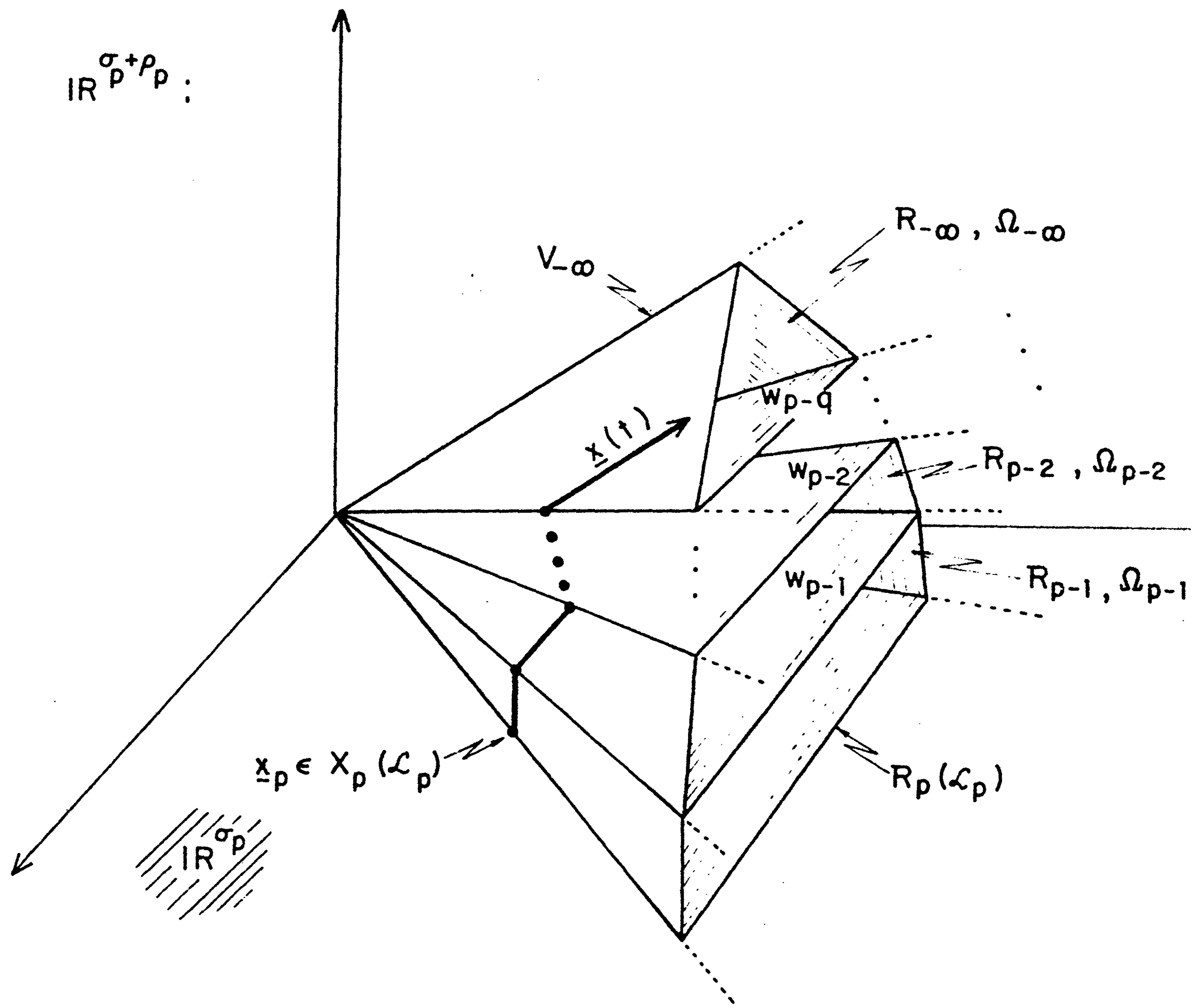


Figure 6 : Construction of Successive Feedback Control Regions from Subregion $R_p(\mathcal{L}_p)$

control regions, are convex polyhedral cones and the method by which the partition may be performed is presented in [5], p.165. For the present, let us assume that $R_p$ has been partitioned into s subregions and denote them by $R_p^1(\mathcal{L}_p), R_p^2(\mathcal{L}_p), \ldots, R_p^s(\mathcal{L}_p)$, where the dependence of the partition on the set $\mathcal{L}_p$ is indicated in parenthesis. We now perform the subsequent operations of the step for each of the s subregions taken one at a time.

- Operation 2 Consider the typical subregion $R_p(\mathcal{L}_p)$. We now call for the state variables in $\mathcal{L}_p$ to leave backward in time from each of a finite set of points of $R_p(\mathcal{L}_p)$ taken one at a time. This set of points is denoted by $X_p(\mathcal{L}_p)$ and as in the case of subregions the definition is deferred until Operation 3. Let us now focus attention on a typical such point $\underline{x}_p \in X_p(\mathcal{L}_p)$. We assume that $\underline{x}_p$ has been reached through a backward optimal trajectory constructed from a sequence of previous steps, and that the time at which $\underline{x}_p$ is reached along this trajectory is $t_p$. Associated with $\underline{x}_p$ at $t_p$ is some possibly nonunique set of costate vectors. We are interested in only those costate vectors which allow for the optimal departure of the state variables in $\mathcal{L}_p$ from the boundary backward in time at $t_p$, known appropriately as *leave-the-boundary costates*. This set may also be nonunique, in which case it will in fact be infinite. It is shown in [5], p.189, that we need only consider a certain finite subset of the total leave-the-boundary costate set and a method for determining this particular set of costate vectors, or for showing that no such costate vectors exist, is presented. We assume now that this set has been found and denote it by $\Lambda_p$.

- <u>Operation 3</u> Consider the typical leave-the-boundary costate $\underline{\lambda}_p \in \Lambda_p$. We now consider the situation in which the state variables in $\mathcal{L}_p$ leave the subregion $R_p(\mathcal{L}_p)$ backward in time from the point $\underline{x}_p$. We note that the set of state variables which are traveling on boundary arcs backward in time subsequent to the departure of $\mathcal{L}_p$ is $B_{p-1} = B_p \mid \mathcal{L}_p$ and the set on interior arcs is $I_{p-1} = I_p \cup \mathcal{L}_p$. We must now solve the following problem:

*Given the state* $\underline{x}_p$ *and the costate* $\underline{\lambda}_p$ *at time* $t_p$, *find* <u>*all*</u> *optimal trajectories backward in time on* $\tau \in (-\infty, t_p)$ *for which* $\dot{x}_i^j(\tau) = 0$ *for all* $x_i^j \in B_{p-1}$ <u>*or*</u> *determine that no such point trajectory exists.*

According to assumption stated earlier in this section it is optimal for all of the state variables of $I_{p-1}$ to remain off of the boundary for the entire time interval $\tau \in (-\infty, t_p)$. Therefore, by the necessary conditions (which are also sufficient) we know that any (and all) trajectories which solve the above problem must have a control which satisfies the following, henceforth referred to as the *global optimization problem:*

Find all

$$\underline{u}^*(\tau) = \text{ARG} \min_{\underline{u}(\tau) \in U} \underline{\lambda}^T(\tau)\dot{\underline{x}}(\tau) = \text{ARG} \min_{\underline{u}(\tau) \in U} \underline{\lambda}^T(\tau)\underline{B}\ \underline{u}(\tau) \tag{89}$$

where

$$\underline{\lambda}(t_p) = \underline{\lambda}_p \tag{90}$$

$$\dot{\lambda}_i^j(\tau) = -\alpha_i^j \qquad \forall x_i^j \in \mathcal{I}_{p-1} \tag{91}$$

$$\left.\begin{aligned} -d\lambda_i^j(\tau) &= \alpha_i^j d\tau + d\eta_i^j(\tau) \\ d\eta_i^j(\tau) &\leq 0 \end{aligned}\right\} \quad \forall x_i^j \in \mathcal{B}_{p-1} \tag{92}$$

$$\forall \tau \in (-\infty, t_p).$$

Our task is therefore to find all solutions to the global optimization problem which satisfy the constraints $x_i^j(\tau) = 0$ for all $x_i^j \in \mathcal{B}_{p-1}$ and all $\tau \in (-\infty, t_p)$ or show that no such solution exists. To find solutions requires producing values of $\eta_i^j(\tau)$ such that $\dot{x}_i^j(\tau) = 0$ is optimal for all $x_i^j \in \mathcal{B}_{p-1}$ and all $\tau \in (-\infty, t_p)$. A method for solving this problem is presented in Appendix A.

If it is shown that no solution exists we immediately terminate this step. On the other hand assume that using the technique of Appendix A we have arrived at a sequence of optimal switching times and optimal control sets on $\tau \in (-\infty, t_p)$. Suppose that $q$ switches occur in the optimal control over this interval and denote the times at which the switches occur by $t_{p-q}, \ldots, t_{p-2}, t_{p-1}$, where the control remians unchanged from time $t_{p-q}$ to minus infinity. All these switching times the backward optimal trajectory intersects the hypersurfaces of various dimensions which separate adjacent feedback control regions. The points of intersection are referred to as *breakpoints* and the hypersurfaces, which are convex polyhedral cones of dimension $\sigma_p + \rho_p - 1$, are referred to as *breakwalls*. We denote by $w_{p-s}$ the breakwall which is encountered at the s-th switch time $t_{p-s}$ and denote the entire set of breakwalls encountered on $\tau \in (-\infty, t_p)$ by

$$W \triangleq \{w_{p-q}, \ldots, w_{p-2}, w_{p-1}\}.$$

We shall show how to construct $W$ later on in this operation. Define $\Omega_{p-s}$ to be the *complete* set of optimal controls on $\tau \in [t_{p-s}, t_{p-s+1})$ which satisfy the constraints $\dot{x}_i^j(\tau) = 0$ for all $x_i^j \in \mathcal{B}_{p-1}$, or formally

$$\Omega_{p-s} \triangleq \{\underline{u}^* \mid \underline{u}^* = \text{ARG} \min_{\left\{\begin{array}{l}\underline{u}\in U \\ \dot{x}_i^j(\tau)=0 \ \forall x_i^j \in \mathcal{B}_{p-1}\end{array}\right.} \underline{\lambda}^T(\tau)\underline{B}\ \underline{u}(\tau),$$

$\tau \in [t_{p-s}, t_{p-s+1})$ where $\underline{\lambda}(\tau)$ is determined by (90)-(92) and $\underline{\lambda}_p$ ranges over all members of $\Lambda_p\}$.

Accordingly, the collection of optimal control sets on $\tau \in (-\infty, t_p)$ is denoted

$$\Omega \triangleq \{\Omega_{-\infty}, \Omega_{p-q}, \ \ldots \ , \Omega_{p-2}, \Omega_{p-1}\}$$

where $\Omega_{-\infty}$ is the solution set which applies from time $t_{p-q}$ to minus infinity. We are now able to provide various details which have been left unspecified until now. First, the definitions of subregion and the set of points $X_p(\mathcal{L}_p) \subset R_p(\mathcal{L}_p)$ mentioned in Operations 1 and 2.

<u>Definition 9</u>: Suppose the set of state variables $\mathcal{L}_p$ is designated to leave the feedback control region $R_p$ backward in time. Then a *subregion* $R_p(\mathcal{L}_p)$ of $R_p$ is the set of all those points in $R_p$ which have taken as the point of departure of $\mathcal{L}_p$ result in a common $\Omega$ and a common W.

<u>Definition 10</u>: If no control switches occur on $\tau \in (-\infty, t_p)$ then $X_p(\mathcal{L}_p)$ consists of exactly one point, and this may be *any* point of $R_p(\mathcal{L}_p)$. If one or more control switches occur (i.e., one or more breakwalls are encountered) then $X_p(\mathcal{L}_p)$ consists of exactly one point from each edge of $R_p(\mathcal{L}_p)$, where we may choose *any* point of a given edge.

Therefore, if no control switches occur we have exhausted $X_p(\mathcal{L}_p)$ by the consideration of the single point $\underline{x}_p$. On the other hand, if one or more control switches occur then we must repeat Operations 2 and 3 for all of the remaining points of $X_p(\mathcal{L}_p)$.

By the definition of subregion we shall obtain the same collection of optimal control sets $\Omega$ and encounter the same set of breakwalls W for every point in $X_p(\mathcal{L}_p)$. However, the breakpoint corresponding to a given breakwall will in general be different for optimal trajectories emanating from different points of $X_p(\mathcal{L}_p)$ or for different optimal trajectories emanating from the same point. We may now specify how to construct the breakwalls from the breakpoints: *Find the complete set of breakpoints occuring at the s-th switch time which correspond to extreme point solutions of* $\Omega_{p-s}$, *where we consider trajectories emanating from every point of* $X_p(\mathcal{L}_p)$. *Form the set of rays in the state space which pass through these breakpoints. Then* $w_{p-s}$ *is the convex hull of all the rays.*

- Operation 4 The sets $\Omega$ and $W$ obtained in the previous operation are now utilized to construct feedback control regions. We consider the two cases:

(i) $q = 0$

In this case $\Omega = \{\Omega_{-\infty}\}$ and $W = \{\emptyset\}$. Consider the linear transformation $\underline{y} = -\dot{\underline{x}} = -\underline{B}\ \underline{u} - \underline{a}$ and the convex polyhedral set $Y_{-\infty} = \{\underline{y} \mid \underline{u} \in \Omega_{-\infty}\}$. For every extreme point of $Y_{-\infty}$ form the ray in $\mathbb{R}^{\sigma_p+\rho_p}$ which passes through that extreme point. If there are $\omega$ extreme points then denote the set of rays by $V_{-\infty} = \{v_1, v_2, \ldots, v_\omega\}$. It is readily seen that each of these rays represents an extreme direction of travel of the optimal trajectory in the state space. Now, let $Co(\cdot)$ denote the convex hull function and from the convex polyhedral cone

$$R_{-\infty} = Co(R_p(\mathcal{L}_p) \cup V_{-\infty})/R_p(\mathcal{L}_p) .$$

It is proven in Appendix B that $R_{-\infty}$ is an feedback control region with associated optimal control set $\Omega_{-\infty}$ in the sense of Definition 4. We refer to $R_{-\infty}$ as a *non-break feedback control region.*

(ii) $q > 0$

In this case $\Omega = \{\Omega_{-\infty}, \Omega_{p-q}, \ldots, \Omega_{p-2}, \Omega_{p-1}\}$ and $W = \{w_{p-q}, \ldots, w_{p-2}, w_{p-1}\}$. Form the sequence of $q+1$ adjacent convex polyhedral cones

$$R_{p-1} = Co(R_p(\mathcal{L}_p) \cup w_{p-1})/R_p(\mathcal{L}_p)$$

$$R_{p-2} = Co(w_{p-1} \cup w_{p-2})/w_{p-1}$$

$$\vdots$$

$$R_{p-q} = Co(w_{p-q+1} \cup w_{p-q})/w_{p-q+1}$$

$$R_{-\infty} = Co(w_{p-q} \cup V_{-\infty})/w_{p-q} \ .$$

It is proven in [5], p.176, that $R_{p-1}, R_{p-2}, \ldots, R_{p-q}$ are feedback control regions with associated optimal control sets $\Omega_{p-1}, \Omega_{p-2}, \ldots, \Omega_{p-q}$ respectively. These are referred to as *break feedback control regions.* Here $R_{-\infty}$ is the *non-break feedback control region* with associated optimal control set $\Omega_{-\infty}$. See Appendix B for proof.

● Step of Algorithm

Note that upon the completion of a single step $q+1$ feedback control regions have been constructed: exactly one non-break feedback control region and $q$ break feedback control regions, $0 \leqslant q < \infty$. We may refer back to Example 2 to find simple examples of both type of feedback control regions: the region specified in (87) is a break feedback control region and that of (88) is of the non-break variety.

Having detailed a single step we now discuss how the overall algorithm operates. The procedure is initiated at $t_f$ with the first feedback control region $\mathcal{R}_f$ being the origin. Here the set $\mathcal{B}_f$ is composed of all the state variables of the problem. We allow $\mathcal{L}_f$ to range over all possible $2^n-1$ non-empty subsets of $\mathcal{B}_f$ and perform a step of the algorithm for each. To this end we know by Corollary 2 that the values of the costates at $t_f$ corresponding to those state variables leaving the boundary at $t_f$ are zero. The constrained optimization of Appendix B may then be solved since only those costates are required which correspond to state variables off the boundary. For each set of state variable leaving the boundary which is found to have globally optimal trajectories, feedback control regions are constructed which range from one dimensional (axes of $\mathbb{R}^n$) to n-dimensional subsets of $\mathbb{R}^n$. At each step we propogate backward in time an appropriate set of state and costate trajectories and save the information which is required to execute subsequent steps. Each region of the set thus constructed is used as the starting point for the sequence of steps which builds new higher dimensional regions. This process continues until all the feedback control regions which are constructed are n-dimensional. Note that the complete set of backward state and costate trajectories which is constructed during the execution of the algorithm will not in general be unique due to the arbitrariness in the selection of the set $X_p(\mathcal{L}_p)$ at each step.

We point out that the feedback control regions constructed during a particular step may have been constructed previously. In essence, we are being conservative in insisting that $\mathcal{L}_p$ be set equal successively to all possible non-empty subsets of $\mathcal{B}_p$, but no method is currently known for the a priori elminination of those subsets which will produce previously constructed regions. However, our thoroughness allows us to state the following:

Theorem 5. Complete execution of the constructive dynamic programming algorithm will result in the specification of the optimal feedback control over the entire admissible state space.

Proof: Feedback control regions are constructed for *every* conceivable type of optimal trajectory in terms of sequences of state variables on and off boundary arcs. Moreover, we are finding the *largest* such regions since we are taking into account all optimal controls corresponding to each sequence. Therefore, the feedback control regions constructed must cover the entire admissible state space.

□

Summarizing, the following questions which have been left unresolved in the current discussion:

1) The validity of the assumption that it is optimal for all the state variables in $I_{p-1}$ to remain off the boundary as time runs to minus infinity.

2) Partitioning $\mathcal{R}_p$ into subregions (Operation 1)

3) Determining the leave-the-boundary costate values (Operation 2).

4) Determination of global optimality (Operation 3 — part (b) of Appendix A).

As the algorithm is presented here in principle only we shall not enter into details regarding off-line calculation or on-line implementation. However, two points are worthy of mention. First, the number of steps to be performed and the number of feedback control regions constructed will be very large for reasonable size networks. In constructing a numerical version of the algorithm we must therefore be concerned with the efficiency of the various operations. Secondly, a large amount of computer storage will be required to implement the solution in real time. The feedback control regions must be specified by a set of linear inequalities which in general may be very large, and the optimal controls within these regions must also be specified. This situation illustrates the tradeoff which occurs between the storage which is required for the on-line implementation of feedback solutions calculated off-line and the amount of calculation involved in the repeated on-line calculation of open-loop solutions.

## VII. CONCLUSIONS

We have considered the linear optimal control problem with linear state and control variable inequality constraints proposed in [2] as a method of analyzing dynamic routing in data communication networks. The conceptual structure of the Constructive Dynamic Programming Algorithm has been presented for finding the feedback solution to this problem when all the inputs to the network are assumed to be constant in time. Several required tasks of the algorithm pose complex questions in themselves and are therefore left unresolved here. These questions are confronted in detail in [5] and a forthcoming paper by the authors, where in the case of single destination networks with all unity weightings in the cost functional simplifications arise which permit a numerical formulation of the algorithm.

## APPENDIX A - COMPUTING BACKWARD OPTIMAL TRAJECTORIES

Consider the following *constrained optimization problem* (i.e. constrained in state) in which the $\eta_i^j$ do not appear:

Find all

$$\underline{u}^*(\tau) = \text{ARG} \min_{\underline{u}(\tau) \in U} \sum_{x_i^j \in I_{p-1}} \lambda_i^j(\tau) \dot{x}_i^j(\tau) \tag{A.1}$$

subject to

$$\dot{x}_i^j(\tau) = 0 \qquad \forall x_i^j \in B_{p-1} \tag{A.2}$$

where

$$\left.\begin{aligned} \lambda_i^j(t_p) &= \text{appropriate component of } \underline{\lambda}_p \quad &\text{(A.3)} \\ \dot{\lambda}_i^j(\tau) &= -\alpha_i^j &\text{(A.4)} \end{aligned}\right\} \forall x_i^j \in I_{p-1}$$

$$\forall \tau \in (-\infty, t_p).$$

The following is presented without the proof, which is trivial:

Theorem A.1 Any solution to the global optimization problem which satisfies $\dot{x}_i^j(\tau) = 0$ for all $x_i^j \in B_{p-1}$ is also a solution to the constrained optimization problem.

We are able to solve the constrained optimization problem immediately since we know all of the coefficients of (A.1) and the values of $\lambda_i^j$ for $x_i^j \in B_{p-1}$ are not required. However, solutions to the constrained optimization problem may not be solutions to the global optimization problem. These observations suggest the following *two part approach to* *finding all solutions to the global optimization problem which satisfy* $\dot{x}_i^j = 0$ *for all* $x_i^j \in B_{p-1}$.

(a) Find all solutions to the constrained optimization problem.

(b) Produce values of $\lambda_i^j(\tau)$, $\tau \in (-\infty, t_p)$, for all $x_i^j \in B_{p-1}$ which satisfy the necessary conditions and such that all solutions to part (a) are also solutions to the global optimization problem *or* show that no such values exist.

The above tasks were performed in a simple fashion for the examples of Section V, where due to the small dimensionality of the problems we were able to solve part (b) by inspection. Of course, this is rarely possible, and a general method for solving part (b), referred to as the *determination of global optimality*, is presented in [5], p.163.

We now turn our attention to the solution of part (a). Taking into account the dynamics (7) and integrating (A.4) backward in time from $t_p$ we may re-write (A.1) - (A.4) in terms of the underlying decision vector $\underline{u}$ as follows:

$$\underline{u}^*(\tau) = \text{ARG} \min_{\underline{u}(\tau) \in U'} (\underline{c}_o + \tau \underline{c}_1)\underline{u}(\tau) \tag{A.5}$$

$$U' = \begin{cases} \underline{D}\,\underline{u}(\tau) \leqslant \underline{C} & \text{(A.6)} \\ \underline{u}(\tau) \geqslant \underline{0} & \text{(A.7)} \\ \underline{b}_i^j \underline{u}(\tau) = -a_i^j \qquad \forall x_i^j \in B_{p-1} & \text{(A.8)} \end{cases}$$

where $\underline{b}_i^j$ = row of $\underline{B}$ corresponding to $x_i^j$

$$\underline{c}_o = \sum_{x_i^j \in I_{p-1}} \lambda_i^j(t_p)\underline{b}_i^j$$

$$\underline{c}_1 = \sum_{x_i^j \in I_{p-1}} \alpha_i^j \underline{b}_i^j$$

and $\tau$ is time running backward from $t_p$ to minus infinity.

The presence of the constraints (A.8) prevents us from immediately specifying the optimal solution at a given time in terms of the costates as is possible in the absence of these constraints. However, since for fixed $\tau$ (A.5) - (A.8) is a linear program the Simplex technique may be applied to find a solution. Moreover, the cost function of (A.5) is a linear function of the single independent parameter $\tau$, while the constraints are not a function of $\tau$ since $\underline{a}$ is constant. This is precisely the form which can be accommodated by parametric linear programming with respect to the cost coefficients. The solution proceeds as follows:

Set $\tau = \delta$, where $\delta$ is some small positive number which serves to perturb all costate values by $\alpha_i^j\delta$. We wish to start our solution at time $t_p - \delta$ since we may have $\lambda_i^j(t_p) = 0$ for some $x_i^j \in B_{p-1}$, so that the solution exactly at $t_p$ may not correspond to $x_i^j$ leaving the boundary. The number $\delta$ must be such that $0 < \delta < t_{p-1}$, where $t_{p-1}$ is the first break time to be encountered backward in time.

We now use the Simplex technique to solve the program at $\tau = \delta$. There are many linear programming computer packages which may be enlisted for this task which utilize efficient algorithmic forms of the Simplex technique to arrive at a single optimal extremum solution. Given this starting solution which we call $\underset{\sim}{u}_{p-1}$, most packages are also equipped to employ parametric linear programming to find the value of $\tau$ for which the current solution ceases to be optimal as well as a new optimal solution. These are the break time $t_{p-1}$ and the optimal control $\underset{\sim}{u}_{p-2}$ respectively. We continue in this fashion to find controls and break times until the solution remains the same for $\tau$ arbitrarily large. This final solution is the control $\underset{\sim}{u}_{-\infty}$ .

The linearity of the pointwise minimization associated with the necessary conditions has enabled us to find a sequence of optimal controls on the time interval $(-\infty,t_p)$ by the efficient technique of parametric linear programming. However, in the description of Operation 3, we call for *all* optimal solutions on every time segment. Since we are dealing with a linear program, the specification of all optimal solutions is equivalent to the specification of all optimal extremum of the solution set. Unfortunately, it turns out that the problem of finding all the optimal extremum solutions to a linear program is an extremely difficult one. It is easily shown that given an initial optimal extremum solution this problem is equivalent to finding all the vertices of a convex polyhedral set defined by a system of linear equality and inequality constraints. Discussion of this problem has appeared intermittently in the linear programming literature since the early 1950's, where several algorithms based upon different approaches have been presented. However, none of these methods has proven computationally efficient for a reasonably large variety of problems. The fundamental difficulty which appears to foil many algorithms, no matter what their underlying approach, is degeneracy in the original linear program. As our problem is characterized by a high degree of degeneracy, one would expect poor performance from any of these algorithms. Hence, it appears at this time that the development of an efficient algorithm for the solution of this problem is contingent upon the discovery of methods for resolving degeneracy in linear programming. As degeneracy is a frequent nuisance in most linear programming procedures, this problem is the subject of much ongoing research.

## APPENDIX B - CONSTRUCTING NON-BREAK FEEDBACK CONTROL REGIONS

Exactly one non-break feedback control $R_{-\infty}$ is constructed in either of the cases $q = 0$ or $q \geqslant 1$. If $q = 0$ then the state variables in $\mathcal{L}_p$ leave $R_p(\mathcal{L}_p)$ backward in time with optimal control set $\Omega_{-\infty}$ and $R_{-\infty}$ is constructed adjacent to $R_p(\mathcal{L}_p)$. Similarly if $q \geqslant 1$ then the state variables in $\mathcal{L}_p$ leave the breakwall $w_{p-q}$ backward in time with optimal control set $\Omega_{-\infty}$ and $R_{-\infty}$ is constructed adjacent to $w_{p-q}$. In this discussion it is unnecessary to distinguish between these cases; we therefore let $\hat{R}_p$ represent either the subregion $R_p(\mathcal{L}_p)$ or the breakwall $w_{p-q}$ depending upon whether $q = 0$ or $q \geqslant 1$ respectively.

Theorem B.1  Suppose $\Omega_{-\infty}$ is the set of optimal controls with which the state variables $\mathcal{L}_p$ leave $\hat{R}_p$ backward in time. Then

$$R_{-\infty} = \mathrm{Co}(\hat{R}_p \cup V_{-\infty})/\hat{R}_p$$

is the non-break feedback control region with associated control set $\Omega_{-\infty}$ in the sense of Definition 4.

Proof. We must show that items (i)-(iii) of Definition 4 apply to $R_{-\infty}$ and $\Omega_{-\infty}$. The situation is depicted in Figure B.1.

We prove item (iii) first. Consider $\underline{x} \in \hat{R}_p$. Translate each ray in $V$ by placing its origin at $\underline{x}$ and call the translated set $V'_{-\infty} = \{v'_1, v'_2, \ldots, v'_\omega\}$. Next form the conical region $\kappa(\underline{x}) = \mathrm{Co}(\underline{x} \cup V'_{-\infty})/\underline{x}$. See Figure B.1. If $\underline{x}_1 \in \kappa(\underline{x})$, then there exists a direction which is some convex combination of the members of $V'_{-\infty}$ which takes $\underline{x}_1$ to $\underline{x}$. Hence, for any $\underline{x}_1 \in \mathcal{R}(x)$ there exists a $\underline{u} \in \Omega_{-\infty}$ which takes $\underline{x}_1$ to $\underline{x}$. Now, $\mathcal{R}_{-\infty} = \text{Co}(\{\mathcal{R}(\underline{x}) \mid \underline{x} \in \hat{\mathcal{R}}_p\})$ since the smallest convex set containing $\{\mathcal{R}(\underline{x}) \mid \underline{x} \in \hat{\mathcal{R}}_p\}$ is clearly $\mathcal{R}_{-\infty}$. Therefore, for any $\underline{x}_1 \in \mathcal{R}_{-\infty}$ there exists some direction which is a convex combination of members of $V_{-\infty}$ which carries $\underline{x}_1$ to some point $\underline{x} \in \mathcal{R}_p$. This is equivalent to saying that for any $\underline{x}_1 \in \mathcal{R}_{-\infty}$, there exists a $\underline{u} \in \Omega_{-\infty}$ such that $\dot{\underline{x}} = \underline{B}\,\underline{u} + \underline{a}$ carries $\underline{x}_1$ to some point $\underline{x} \in \hat{\mathcal{R}}_p$. Also, the trajectory remains within $\mathcal{R}_{-\infty}$ until it strikes $\hat{\mathcal{R}}_p$.

Now, let us select some $\underline{x}_1 \in \mathcal{R}_{-\infty}$ and apply any control $\underline{u}_1 \in \Omega_{-\infty}$ which helps the state within $\mathcal{R}_{-\infty}$ for a non-zero period of time $\Delta t$. Clearly there exists such a control by the above argument. Denote by $\underline{x}_2$ the state which results after applying $\underline{u}_1$ for the time $\Delta t$. Then also by the above argument there exists some control $\underline{u}_2 \in \Omega_{-\infty}$ which takes $x_{-2}$ to some point $\underline{x}_3 \in \hat{\mathcal{R}}_p$. See Figure B.1. The control $\underline{u}_2$ is optimal since $\Omega_{-\infty}$ is constructed such that any $\underline{u} \in \Omega_{-\infty}$ is optimal to move the state off of $\hat{\mathcal{R}}_p$ backward in time. Finally, $\underline{u}_1$ is optimal since $\underline{u}_1 \in \Omega_{-\infty}$ and the trajectory segment $\underline{x}_2 \to \underline{x}_1$ in part of the trajectory $\underline{x}_3 \to \underline{x}_2 \to \underline{x}_1$ which leaves from $\hat{\mathcal{R}}_p$. We have therefore shown that item (iii) of Definition 4 is satisfied.

Items (i) and (ii) follow easily from the fact that $\hat{\mathcal{R}}_p$ is itself part of a feedback control region.

□ Theorem B.1

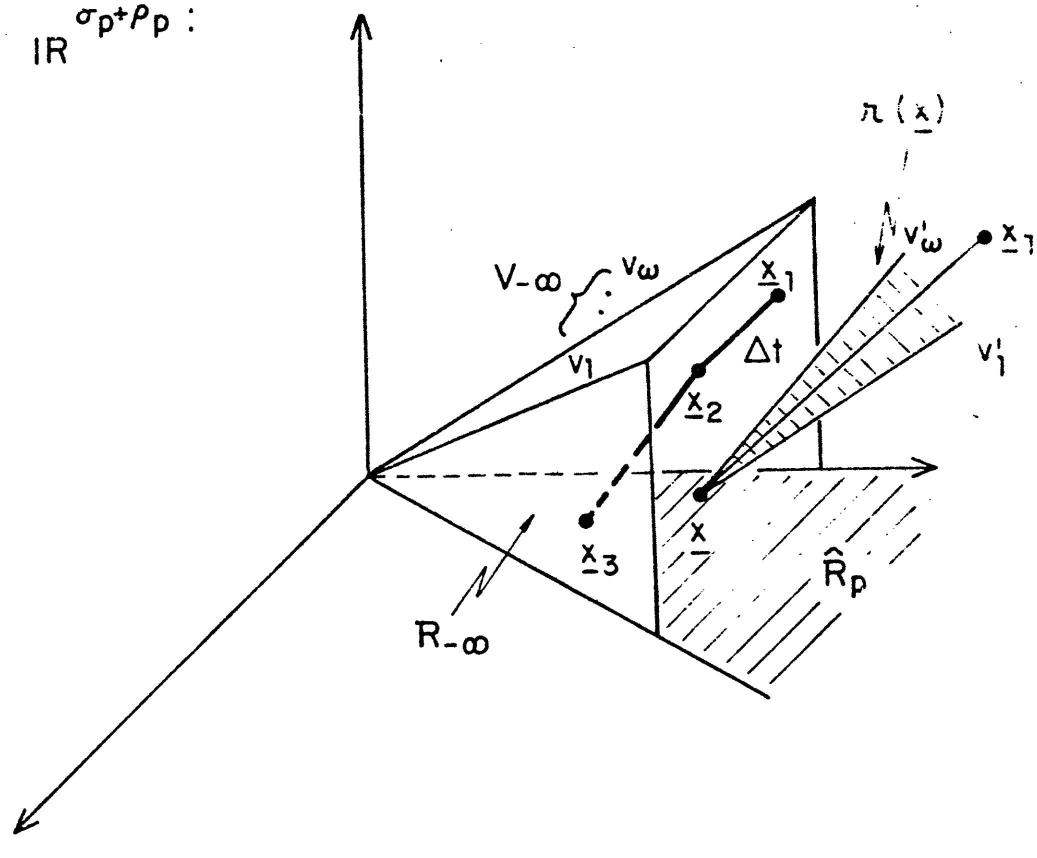


Figure B.1 ; Geometry for Proof of Theorem B.1

REFERENCES


[1] Kleinrock, L., Queueing Systems, Volume 2: Computer Applications, John Wiley & Sons, New York, 1976.

[2] Segall, A., "The Modelling of Adaptive Routing in Data-Communication Networks", IEEE Trans. Comm., Vol. COM-25, No.1, pp. 85-95, January 1977.

[3] Maurer, H., "On Optimal Control Problems with Bounded State Variables and Control Appearing Linearly", SIAM J. on Cont. and Opt. Vol. 15, No.3, May 1977.

[4] Kapur, K.C. and Van Slyke, R.M., "Cutting Plane Algorithms and State Space Constrained Linear Optimal Control Problems", J. of Computer and Systems Sciences, 4, pp. 570-605, 1970

[5] Moss, F.H., "The Application of Optimal Control Theory to Dynamic Routing in Data Communication Networks", Ph.D. dissertation, Electronic Systems Laboratory Report ESL-R-721, February 1977.

[6] Jacobson, D.H., Lele, M.M. and Speyer, J.L., "New Necessary Conditions of Optimality for Control Problems with State-Variable Inequality Constraints", J. of Math. Anal. and Appl., Vol.35, No.2, pp.255-284, August 1971.