AD 748347

AIR FORCE INSTITUTE OF TECHNOLOGY

AIR UNIVERSITY
UNITED STATES AIR FORCE

SCHOOL OF ENGINEERING

WRIGHT-PATTERSON AIR FORCE BASE, OHIO

71

A METHOD FOR GENERATING CLOSED-LOOP
SOLUTIONS TO DIFFERENTIAL GAMES

THESIS

GA/MC/72-4 Percy J. Gros, Jr.
Captain USAF

1a

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Air Force Institute of Technology (AFIT-EN) Wright-Patterson AFB, Ohio 45433	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED
	2b GROUP

3 REPORT TITLE

A Method for Generating Closed-Loop Solutions to Differential Games

4 DESCRIPTIVE NOTES (Type of report and inclusive dates)

AFIT Thesis

5 AUTHOR(S) (First name, middle initial, last name)

Percy J. Gros, Jr.
Captain, USAF

6 REPORT DATE	7a TOTAL NO OF PAGES	7b NO OF REFS
June 1972	68	2

8a CONTRACT OR GRANT NO	9a ORIGINATOR'S REPORT NUMBER(S) GA/MC/72-4
b PROJECT NO N/A	
c	9b OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10 DISTRIBUTION STATEMENT

This document has been approved for public release; distribution unlimited.

Approved for public release; IAW AFR 190-17 Keith A. Williams, 1st Lt., USAF Acting Director of Information	12 SPONSORING MILITARY ACTIVITY

13 ABSTRACT

The zero-sum, perfect information pursuit-evasion differential game is reviewed. The purpose of ~~this~~ thesis is to formulate a method for generating near-optimal closed-loop solutions to these problems. The method is then applied to a number of example problems in order to check its validity. This method deals with solutions in the small and is based on updating the two-point boundary-value problem by use of the neighboring extremal path concept.

The two differential game problems examined are a simple motion problem and a rocket problem. Two separate cases were studied for each problem. One was the fixed final time problem and the other was the free final time with a terminal constraint.

Analysis of the results obtained, supports the feasibility of this method to provide near-optimal closed-loop solutions to differential game problems.

DD FORM 1 NOV 65 1473

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Differential Games						
Pursuit-Evasion						
Closed-Loop Solutions						

A METHOD FOR GENERATING CLOSED-LOOP
SOLUTIONS TO DIFFERENTIAL GAMES

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology

Air University

in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

Percy J. Gros, Jr.
Captain USAF

Graduate Aerospace-Mechanical Engineering

June 1972

1d

Preface

This work is the result of my attempt to apply a neighboring extremal path approach to obtain closed-loop solutions to pursuit-evasion differential games. I am indebted to Professor Gerald M. Anderson of the Air Force Institute of Technology for his guidance and encouragement. I also want to thank my wife for her assistance in the preparation of this thesis.

Percy J. Gros, Jr.

Table of Contents

Table of Contents (Contd)

List of Figures

List of Tables

Abstract

The zero-sum, perfect information pursuit-evasion differential game is reviewed. The purpose of this thesis is to formulate a method for generating near-optimal closed-loop solutions to these problems. The method is then applied to a number of example problems in order to check its validity. This method deals with solutions in the small and is based on updating the two-point boundary-value problem by use of the neighboring extremal path concept.

The two differential game problems examined are a simple motion problem and a rocket problem. Two separate cases were studied for each problem. One was the fixed final time problem and the other was the free final time with a terminal constraint.

Analysis of the results obtained, supports the feasibility of this method to provide near-optimal closed-loop solutions to differential game problems.

A METHOD FOR GENERATING CLOSED-LOOP SOLUTIONS TO DIFFERENTIAL GAMES

I. Introduction

The original formulation of differential games was presented by Isaacs (Ref 1) less than twenty-five years ago. It was developed from the theory of games and deals with games in which two opposing players are "confronted with lengthy sequences--be they continuous or discrete--of decisions which are knit together logically so that a perceptible and calculable pattern prevails throughout" (Ref 1:3). Thus it soon found applications in economics, in the development of control systems and in analyzing warfare problems. The latter will be the area of consideration in this thesis.

We will consider games involving two players, a pursuer and an evader, each with conflicting interests and each with complete information about his opponent's state and available strategies. Ideally the evader wants to select his strategy based on the present state of the game that will maximize a certain quantity called the "cost" or "payoff". The cost may be any number of things such as the time to capture, or the distance between players, or the fuel required by the pursuer, etc... . The pursuer at the same time wants to select his strategy based on the present state of the game that will minimize the cost. The strategies supply instructions as to how to set the controls for each set of data measured. If the control variables are functions of the state variables and time, we have a closed-loop solution to the problem. In this case, if either player plays non-optimally, the other player if playing optimally will immediately take advantage of this and gain what the first player loses. This is known as the zero-sum feature of differential games.

Although it is evident that the closed-loop solution is the desired solution in all differential games, often there is no practical means of obtaining it. In most problems the costate or adjoint differential equations which must be integrated to provide the link between the controls and state variables are nonlinear, nonhomogeneous equations that can not be solved other than numerically. Therefore we are left with optimal control strategies which are functions of time and the initial conditions of the problem. These are called open-loop control laws. Basically we have a two-point boundary-value problem (TPBVP) with the initial states and final costates given. If solved, it provides an optimal open-loop trajectory which for zero-sum games is the same as the closed-loop trajectory if both players play optimally. This is not true if either player deviates from his optimal strategy.

The purpose of this thesis is to devise a method for obtaining near optimal closed-loop solutions to differential games and to apply the method to a number of problems to demonstrate its validity. The method is developed and an algorithm presented in Chapter II. This method is based on the assumption that the TPBVP can be solved to provide a needed reference optimal open-loop trajectory. Also, the solutions considered in this thesis are solutions in the small. That is, they refer to the smooth parts of the solution found between the singular surfaces that separate the number of parts of the playing space.

Fixed final time problems are examined in Chapters III and IV, while free final time problems with terminal constraint are examined in Chapters V and VI.

II. Statement of the Problem

Differential Game Problem

The zero-sum perfect information pursuit-evasion differential game setup will be represented by the following dynamic system (Ref 2:277)

$$\dot{x} = f(x,u,v,t), \quad x(t_o) = x_o \tag{2-1}$$

where x is an n-dimensional state vector, u is an m-dimensional decision (or control) vector for the pursuer, v is a p-dimensional control vector for the evader and t_o represents the initial time. The controls may or may not be subject to constraints depending on the problem being considered. The terminal constraints (conditions which must be satisfied when the game is over) are

$$\psi[x(t_f),t_f] = 0 \tag{2-2}$$

where ψ is a q vector, and the performance criterion (cost or payoff) is

$$J = \phi[x(t_f),t_f] + \int_{t_o}^{t_f} L(x,u,v,t)\,dt \tag{2-3}$$

The object is to find u* and v* such that

$$J(u^*,v) \leq J(u^*,v^*) \leq J(u,v^*) \tag{2-4}$$

If u* and v* can be found, the pair (u*,v*) is called a saddle point of the game and J(u*,v*) is called the value of the game.

Necessary Conditions. In order to apply the necessary conditions for a saddle point solution in the small, the Hamiltonian (H) is defined as

$$H(t,x,\lambda,u,v) = \lambda^T f + L \tag{2-5}$$

where λ is an n-dimensional costate vector. This scalar function H must be minimized over the set of admissible u and maximized over the set of admissible v in order to have a saddle point solution of the above differential game, that is

$$H^* = \underset{v}{\text{Max}}\ \underset{u}{\text{Min}}\ H = \underset{u}{\text{Min}}\ \underset{v}{\text{Max}}\ H \tag{2-6}$$

and the second-order necessary conditions are that

$$H^*_{uu} \geq 0 \quad , H^*_{vv} \leq 0 \tag{2-7}$$

As mentioned in Chapter I, singular arcs will not be considered in this thesis. In some optimization problems, extremal arcs ($H_u = 0$) occur on which the matrix H_{uu} is singular. Such arcs are called singular arcs.

The costate differential equations are

$$\dot{\lambda}^t = -\partial H/\partial x = -H_x \tag{2-8}$$

and the transversality conditions are given by

$$H(t_f) = -\Phi_t(t_f);\ \lambda^T(t_f) = \Phi_x(t_f) \tag{2-9}$$

where $\Phi[x(t_f),t_f] = \phi + \nu^T\psi$ and ν^T is a constant Lagrange multiplier.

Open Vs. Closed-Loop Controls. As previously stated, the optimal solution to the differential game problem is the pair of controls (u^*,v^*) which provides a saddle point of J. Attention must be given to the interpretation of u^* and v^* in Eq (2-4) as open-loop or closed-loop strategies. If the pair (u^*,v^*) is a function of time and the initial conditions (i.e. $u^*(t,x_o,t_o)$ and $v^*(t,x_o,t_o)$), one speaks of an open-loop solution. If the controls are expressed as functions of the instantaneous state and time

$$\begin{aligned} u^* &= k_u(x,t) \\ v^* &= k_v(x,t) \end{aligned} \tag{2-10}$$

one has what is known as a feedback or closed-loop control law. The closed-loop control law is a much more strigent type of optimality. It means that the evader must play optimally against an opponent whose control is produced in a feedback fashion; that is, the pursuer can immediately take advantage of any nonoptimal play made by the evader.

If both players play their optimal strategy, the open-loop and closed-loop solutions are the same for zero-sum games. But if either player deviates from his optimal play, the open-loop solution will differ from the actual or closed-loop solution and this difference could result in a complete change in the outcome of the game. Therefore we would like to generate closed-loop control laws. This can be done if the solution to the two-point boundary-value problem (TPBVP) can be continuously updated based on current states. In order to accomplish this, we must find the effect on the costates due to small changes in the states; that is, we must find $\delta\lambda(t)$ as a function of $\delta x(t)$ which can be obtained from a neighboring extremal path approach.

Neighboring Extremal Paths

This will be an extension of Bryson and Ho's development (Ref 2:177) to differential games As was mentioned before, if both players play their optimal strategy, the open and closed-loop solutions are the same for zero-sum games. Let us suppose that we have determined a reference trajectory by solving the TPBVP in the small and have been given the initial conditions for the states.

First, for problems in which the final time is specified, if we consider small perturbations from this reference extremal path produced by small perturbations in the initial state $\delta x(t_o)$ and in the terminal conditions $\delta\psi$, we expect that such perturbations will give rise to perturbations $\delta x(t)$, $\delta\lambda(t)$, $\delta u(t)$, $d\nu$ governed by linearizing Eqs (2-1), (2-2), (2-6), (2-8) and (2-9) around the extremal path. We therefore can obtain the following equations:

$$\delta\dot{x} = A(t)\delta x - B(t)\delta\lambda, \ \delta x(t_o) \text{ specified} \tag{2-11}$$

$$\delta\dot{\lambda} = -C(t)\delta x - A^T(t)\delta\lambda \tag{2-12}$$

$$\partial H/\partial u = 0, \ \partial H/\partial v = 0 \tag{2-13}$$

$$\delta\lambda(t_f) = [(\phi_{xx}+\nu^T{}_{xx})\delta x+\psi_x^T d\nu]_{t = t_f} \tag{2-14}$$

$$\delta\psi = [\psi\delta x]_{t = t_f} \tag{2-15}$$

where

$$A(t) = f_x - f_u H_{uu}^{-1} H_{ux}$$
$$B(t) = f_u H_{uu}^{-1} f_u^T$$
$$C(t) = H_{xx} - H_{xu} H_{uu}^{-1} H_{ux}$$

which are (nxn) matrices. These equations represent a linear two-point boundary-value problem since the coefficients are evaluated on the extremal path.

By using the backward sweep method (Ref 1:179) for determining the neighboring extremal path we arrive at the following matrix differential equations

$$\dot{S} = -SA - A^TS + SBS - C \qquad (2\text{-}16)$$
$$\dot{R} = -(A^T - SB)R \qquad (2\text{-}17)$$
$$\dot{Q} = R^TBR \qquad (2\text{-}18)$$

and boundary conditions

$$S(t_f) = [\phi_{xx} + \nu^T\psi_{xx}]_{t = t_f} \qquad (2\text{-}19)$$
$$R(t_f) = [\psi_x^T]_{t = t_f} \qquad (2\text{-}20)$$
$$Q(t_f) = 0 \qquad (2\text{-}20)$$

where

$$\delta\lambda(t) = S(t)\delta x(t) + R(t)d\nu \qquad (2\text{-}22)$$
$$\delta\psi = R^T(t)\delta x(t) + Q(t)d\nu \qquad (2\text{-}23)$$

If these matrix differential equations are integrated backwards from $t = t_f$, the relations (2-21) and (2-22) represent boundary conditions equivalent to the terminal boundary conditions (2-13) and (2-14) at earlier times; thus, we are "sweeping" the terminal boundary conditions backward to earlier times. This allows us to eventually determine δx and $\delta\lambda$ at this earlier time.

Now, if we consider the case where the final time is unspecified, the nominal optimum solution must satisfy the additional necessary condition

$$\Omega(x,u,v,t)\Big|_{t = t_f} \equiv \left(\frac{d\Phi}{dt} + L\right)_{t = t_f} \tag{2-24}$$

where

$$\Phi = \phi(x,t)+\nu^T\psi(x,t), \quad \frac{d\Phi}{dt} = \frac{\partial\Phi}{\partial t} + \frac{\partial\Phi}{\partial x}\dot{x} \tag{2-25}$$

This scalar equation determines the additional unknown parameter, t_f.

Perturbation of the necessary conditions (2-2), (2-9), and (2-23) must take into account the perturbations in the final time, dt_f. Finally this leads to

$$\begin{bmatrix} \delta\lambda(t_f) \\ d\nu \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{\partial^2\Phi}{\partial x^2}, & \left(\frac{\partial\psi}{\partial x}\right)^T, & \left(\frac{\partial\Omega}{\partial x}\right)^T \\ \frac{\partial\psi}{\partial x}, & 0, & \frac{d\psi}{dt} \\ \frac{\partial\Omega}{\partial x}, & \left(\frac{d\psi}{dt}\right)^T, & \frac{d\Omega}{dt} \end{bmatrix}_{t = t_f} \begin{bmatrix} \delta x(t_f) \\ d\nu \\ dt_f \end{bmatrix} \quad \begin{matrix} (2\text{-}26) \\ (2\text{-}27) \\ (2\text{-}28) \end{matrix}$$

where

$$\frac{d\Omega}{dt} = \frac{\partial\Omega}{\partial t} + \frac{\partial\Omega}{\partial x}f, \quad \frac{d\psi}{dt} = \frac{\partial\psi}{\partial t} + \frac{\partial\psi}{\partial x}f \tag{2-29}$$

Equations (2-11) through (2-13) plus (2-26) through (2-28) represent a linear two-point boundary-value problem for a neighboring extremal with small changes in initial conditions, $\delta x(t_o)$, and/or small changes in the terminal conditions, $d\psi$. These changes will, in general, produce small changes $\delta x(t_f)$, $d\nu$ and dt_f.

As in the previous case, where the final time was fixed, we may extend the backward sweep method to solve the unspecified final time problem. By using the substitution

$$\begin{bmatrix} \delta\lambda(t) \\ d\psi \\ d\Omega \end{bmatrix} = \begin{bmatrix} S(t) & , R(t) & , m(t) \\ R^T(t) & , Q(t) & , n(t) \\ m^T(t) & , n^T(t) & , \alpha(t) \end{bmatrix} \begin{bmatrix} \delta x(t) \\ d\nu \\ dt_f \end{bmatrix} \qquad \begin{matrix} (2\text{-}30) \\ (2\text{-}31) \\ (2\text{-}32) \end{matrix}$$

and after differentiating them and making use of the perturbation equations, we obtain the following differential equations and boundary conditions

$$\dot{S} = -SA-A^TS+SBS-C \quad , \quad S(t_f) = \left(\frac{\partial^2\phi}{\partial x^2}\right)_{t\,=\,t_f} \qquad (2\text{-}33)$$

$$\dot{R} = -(A^T-SB)R \quad , \quad R(t_f) = \left(\frac{\partial\psi}{\partial x}\right)^T_{t\,=\,t_f} \qquad (2\text{-}34)$$

$$\dot{Q} = R^TBA \quad , \quad Q(t_f) = 0 \qquad (2\text{-}35)$$

$$\dot{m} = -(A^T-SB)m \quad , \quad m(t_f) = \left(\frac{\partial\Omega}{\partial x}\right)^T_{t\,=\,t_f} \qquad (2\text{-}36)$$

$$\dot{n} = R^TBm \quad , \quad n(t_f) = \left(\frac{d\psi}{dt}\right)_{t\,=\,t_f} \qquad (2\text{-}37)$$

$$\dot{\alpha} = m^TBm \quad , \quad \alpha(t_f) = \left(\frac{d\Omega}{dt}\right)_{t\,=\,t_f} \qquad (2\text{-}38)$$

Equations (2-16) and (2-33) are identical and they are known as the matrix Riccati equation. If equations (2-33) through (2-38) are integrated backwards from t_f to t_1, we can use (2-31) and (2-32), evaluated at $t = t_1$, to determine $d\nu$ and dt_f in terms of $\delta x(t_1)$ and $d\psi$, as follows

$$d\nu = [\overline{Q}^{-1}(d\psi - \overline{R}^T \delta x)]_{t = t_f} \qquad (2\text{-}39)$$

$$dt_f = -[(\frac{m^T}{\alpha} - \frac{n^T}{\alpha}\overline{Q}^{-1}\overline{R}^T)\delta x + \frac{n^T}{\alpha}\overline{Q}^{-1}d\psi]_{t = t_f} \qquad (2\text{-}40)$$

where

$$\overline{Q} \equiv Q - \frac{nn^T}{\alpha} \qquad (2\text{-}41)$$

$$\overline{R} \equiv R - \frac{mn^T}{\alpha} \qquad (2\text{-}42)$$

Since we have $d\nu$ and dt_f from (2-39) and (2-40), $\delta\lambda(t_1)$ can be determined from (2-30):

$$\delta\lambda(t_1) = [(\overline{S} - \overline{R}\,\overline{Q}^{-1}\overline{R}^T)\delta x + \overline{R}\,\overline{Q}^{-1}d\psi]_{t = t_1} \qquad (2\text{-}43)$$

where

$$\overline{S} = S - \frac{mm^T}{\alpha} \qquad (2\text{-}44)$$

<u>Neighboring Extremal Algorithm</u>

The method used to solve the above problem is depicted in Fig 1, page 12, and explained below.

(a) Given the initial conditions, we assume we can solve the TPBVP. If both players play their optimal strategy, integrate the "reference" state and costate differential equations forward from t_0 to t_f and get the reference optimal open-loop trajectory. This is represented by the curve

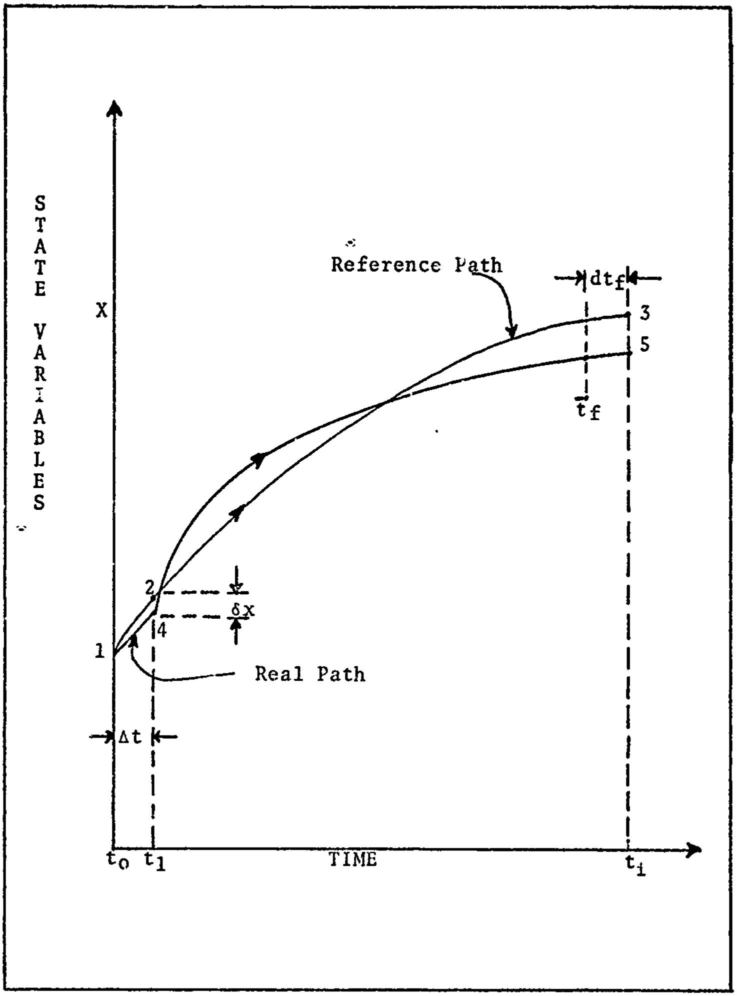

Figure 1. Neighboring Extremal Path

between points 1 and 3. But if the evader plays a non-optimal strategy, the "real" trajectory (from point 1 to 4) can be obtained by substituting this strategy into the state and costate equations and integrating forward from t_o. Stop the integration at t_1 after an elapsed time of Δt.

(b) Having determined the final reference states at t_f, enforce the transversality conditions (Eq 2-9) and boundary conditions (for fixed final time problems Eqs 2-19 through 2-21; for unspecified final time problems Eqs 2-33 through 2-38) for the matrix Riccati equations.

(c) Using the boundary conditions from step (b), integrate the matrix Riccati differential equations backwards along the reference trajectory from t_f to t_1 (point 2). This gives the values of $S(t_1)$, $R(t_1)$, $Q(t_1)$, ... which will be needed later to compute $\delta\lambda(t_1)$.

(d) Compute the difference between the real (point 4) and reference (point 2) states at t_1.

$$\delta x(t_1) = \underset{\text{Real}}{x(t_1)} - \underset{\text{Ref.}}{x(t_1)} \tag{2-45}$$

(e) Using Eq (2-19) for the fixed time problem (or Eq 2-30 for the unspecified final time problem) and the results of step (c), compute $\delta\lambda(t_1)$ and update the costates at t_1 by

$$\lambda(t_1)_{new} = \lambda(t_1)_{old} + \delta\lambda(t_1) \tag{2-46}$$

(f) Now at time t_1, you have a "new" or updated TPBVP with the initial conditions for the states being the real values at point 4 and the initial conditions for the co-states being the values computed in step (e).

(g) Repeat steps (a) through (f) until you reach t_f for the fixed final time problems or until the terminal constraint is satisfied for the unspecified final time problems. We are assuming that the linearized differential equations of the problems are valid for small perturbations from the optimal extremal path. If this were not a valid assumption the solution would be expected to diverge. If it does not diverge, the assumption should be valid.

III. Pursuit-Evasion Differential Game-Simple Motion Problem, Fixed Final Time

Statement of the Problem

This is a two-dimensional problem as depicted in Fig 2, page 16. The pursuer and evader have no restrictions on their direction or motion, that is they can change direction instantaneously over the complete 360° circle around them. There are no terminal constraints on the problem. We will start at some time t_o and run until a fixed time which will be called t_f. The magnitude of the velocity of each player is constant, but the pursuer has a speed advantage. The cost or payoff will be one-half the square of the distance between the players at t_f. In other words, we want to determine the saddle point of

$$J(t_f) = \frac{1}{2}(x^2+y^2)\Big|_{t = t_f} \tag{3-1}$$

subject to the following differential equations of motion in a relative coordinate system with the origin at the pursuer.

$$\dot{x} = V_e \cos v - V_p \cos u \tag{3-2}$$

$$\dot{y} = V_e \sin v - V_p \sin u \tag{3-3}$$

The subscripts p and e refer to the pursuer and the evader respectively. The pursuer's control is u and the evader's control is v, which is their respective direction of motion.

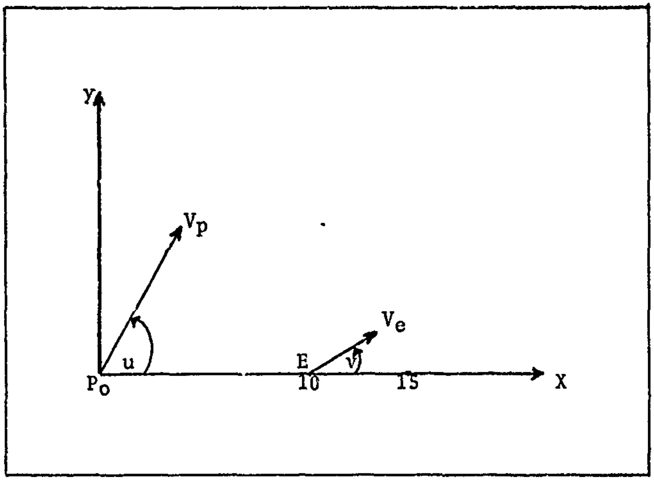

Figure 2. Simple Motion Two-Dimensional Problem

Necessary Conditions

Applying the necessary conditions for a saddle point solution, as explained in Chapter II, the Hamiltonian H is given by

$$H = \lambda_x[V_e\cos v - V_p\cos u] + \lambda_y[V_e\sin v - V_p\sin u] \quad (3\text{-}4)$$

The Hamiltonian is to be minimized with respect to the pursuer's control, and maximized with respect to the evader's control. To do this, it is necessary that

$$\frac{\partial H}{\partial u} = 0 \quad , \quad \frac{\partial H}{\partial v} = 0 \quad (3\text{-}5)$$

and

$$\frac{\partial^2 H}{\partial u^2} \geq 0 \quad , \quad \frac{\partial^2 H}{\partial v^2} \leq 0 \quad (3\text{-}6)$$

From Eq (3-4) we get

$$\partial H/\partial u = -V_p[-\lambda_x \sin u + \lambda_y \cos u] \qquad (3\text{-}7)$$

In order to satisfy the first-half of Eq (3-5) and at the same time minimize H we must have

$$\cos u^* = \lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2}, \ \sin u^* = \lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} \qquad (3\text{-}8)$$

which gives

$$\partial H/\partial u = -V_p[-\lambda_x\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} + \lambda_x\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2}] = 0 \qquad (3\text{-}9)$$

The first-half of Eq (3-6) becomes

$$\partial^2 H/\partial u^2 = +V_p[\lambda_x \cos u^* + \lambda_y \sin u^*]$$
$$= V_p[\lambda_x^2/(\lambda_x^2+\lambda_y^2)^{1/2}+\lambda_y^2/(\lambda_x^2+\lambda_y^2)^{1/2}]>0 \qquad (3\text{-}10)$$

In a similar manner it can be shown that in order to maximize H with respect to the evader's control v we must have

$$\cos v^* = \lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2}; \ \sin v^* = \lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} \qquad (3\text{-}11)$$

This will satisfy the second-half of Eqs (3-5) and (3-6) as shown here

$$\partial H/\partial v = V_e[-\lambda_x \sin v^* + \lambda_y \cos v^*]$$
$$= V_e[-\lambda_x\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2}+\lambda_x\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2}] = 0 \qquad (3\text{-}12)$$

$$\partial^2 H/\partial v^2 = V_e[-\lambda_x \cos v^* - \lambda_y \sin v^*]$$

$$= -V_e[\lambda_x^2/(\lambda_x^2+\lambda_y^2)^{1/2}+\lambda_y^2/(\lambda_x^2+\lambda_y^2)^{1/2}] < 0 \quad (3\text{-}13)$$

Also by using the controls in Eqs (3-8) and (3-11) we see that

$$H^* = \underset{v}{\text{Max}}\ \underset{u}{\text{Min}}\ H = \underset{u}{\text{Min}}\ \underset{v}{\text{Max}}\ H \quad (3\text{-}14)$$

which must be satisfied. Thus Eqs (3-8) and (3-11) represent the optimal open-loop control laws for this problem.

The costate equations are

$$\dot\lambda_x = -\partial H/\partial x = 0 \quad (3\text{-}15)$$

$$\dot\lambda_y = -\partial H/\partial y = 0 \quad (3\text{-}16)$$

The transversality conditions give

$$\lambda_x(t_f) = x(t_f) \quad (3\text{-}17)$$

$$\lambda_y(t_f) = y(t_f) \quad (3\text{-}18)$$

Open-Loop Solution. Now, if we substitute the controls (u^*,v^*) into the equation of motion, we get the reference differential equations for the open-loop solution to the problem. The state equation and boundary conditions are

$$\dot x = (V_e-V_p)\lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2},\ x(o) \text{ given} \quad (3\text{-}19)$$

$$\dot y = (V_e-V_p)\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2},\ y(o) \text{ given} \quad (3\text{-}20)$$

and the costate equations and transversality conditions are

$$\dot{\lambda}_x = \dot{\lambda}_y = 0 \qquad (3\text{-}21)$$

$$\lambda_x(t_f) = x(t_f) \qquad (3\text{-}22)$$

$$\lambda_y(t_f) = y(t_f) \qquad (3\text{-}23)$$

By making use of the neighboring extremal path development of Chapter II, we find that Eq (2-22) reduces to

$$\delta\lambda(t) = S(t)\ \delta x(t) \qquad (3\text{-}24)$$

with the matrix Riccati equation (2-16) becoming

$$\dot{S} = S\ B\ S,\ S(t_f) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad (3\text{-}25)$$

where

$$B(t) = -(V_c - V_p/(\lambda_x^2+\lambda_y^2)^{1/2} \begin{bmatrix} \lambda_y^2 & -\lambda_x\lambda_y \\ -\lambda_x\lambda_y & \lambda_x^2 \end{bmatrix} \qquad (3\text{-}26)$$

<u>Closed-Loop Solution</u>. For this problem, the costate differential equations can easily be integrated and the closed-loop solution found. Equations (3-21) through (3-23) give

$$\lambda_x = \text{constant} = x(t_f) \qquad (3\text{-}27)$$

$$\lambda_y = \text{constant} = y(t_f) \qquad (3\text{-}28)$$

which when substituted in Eqs (3-19) and (3-20) implies that

$$x(t)/y(t) = x(t_f)/y(t_f)$$

Therefore the optimal closed-loop control laws may be written as

$$\cos u^* = \cos v^* = x(t)/(x^2(t)+y^2(t))^{1/2}$$
$$\sin u^* = \sin v^* = y(t)/(x^2(t)+y^2(t))^{1/2} \qquad (3\text{-}29)$$

Substituting (3-29) into (3-2) and (3-3) gives the optimal closed-loop differential equations of motion. But if the evader decided to play non-optimally while the pursuer played his optimal strategy the closed-loop state equations of motion would be

$$\dot{x} = V_e \cos v - V_p x/(x^2+y^2)^{1/2} \qquad (3\text{-}30)$$

$$\dot{y} = V_e \sin v - V_p y/(x^2+y^2)^{1/2} \qquad (3\text{-}31)$$

These equations, when integrated forward from t_o to t_f, gave the actual trajectory which served as a basis with which to compare the results obtained from the proposed method for generating the closed-loop solutions for this problem.

<u>Program Algorithm</u>

Based on the above development of the open-loop solution, a computer program was written using the neighboring extremal path approach in an effort to arrive at the closed-loop solution for this problem. The program followed the algorithm outlined in Chapter II and depicted in Fig 1. All integrations were done using a variable step fourth order Runge-Kutta method.

(1) To carry out step (a) of the neighboring extremal algorithm, the initial conditions and input data for this problem were

$$x(t_o) = 10. \qquad V_p = \text{constant} = 2.$$
$$y(t_o) = 0. \qquad V_e = \text{constant} = 1.$$
$$t_o = 0. \qquad t_f = 5.$$

If both players play their optimal strategy, the reference open-loop solution to the TPBVP gives

$$x(t_f) = x(5.) = 5.$$
$$y(t_f) = y(5.) = 0.$$

which according to Eqs (3-21) through (3-23) gives the following costates

$$\lambda_x(t) = \text{constant} = x(t_f) = 5.$$
$$\lambda_y(t) = \text{constant} = y(t_f) = 0.$$

In our aim of determining control laws based on the current state and time, we will assume that the evader decides to play a non-optimal constant strategy of $v = 90°$ as opposed to the optimal strategy which for this problem is $v^* = 0°$ according to Eq (3-11). Therefore the "real" equations of motion are determined from Eqs (3-2) and (3-3) to be

$$\begin{aligned} \dot{x} &= -V_p \ \lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} \\ \dot{y} &= V_e - V_p\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} \end{aligned} \qquad (3\text{-}32)$$

Integrating these equations forward from t_o to t_1 gives the curve from point 1 to 4 in Fig 1.

(2) Now follow the procedures laid out in steps (b) through (g) of Chapter II. Thus we generate a closed-loop trajectory for a continuously updated TPBVP. The smaller the size of the sampling interval and integration step the closer we should be to the actual closed-loop trajectory.

Results and Analysis

The results from this problem are presented in Table I, page 24, and Fig 3, page 25. Two runs were made using the actual closed-loop solution with different integration step sizes. The resulting costs were the same for both runs. It is approximately 6% lower than the costs obtained from the near-optimal solution for the same sampling step size of 0.5.

From the data, we see that for a specific sampling step size there is no change in the final cost as a result of different integration step sizes being used for the reference, matrix Riccati and real differential equations. This indicates that one can decrease the integration time by selecting a relatively coarse step size of .01 and yet not change the cost.

The slope of the curve for the near-optimal solution seems to indicate that as the sampling step size approaches zero, a limiting minimum cost is approached. This is in agreement with one's intuition regarding the sampling interval and cost.

These runs were made without enforcing the transversality conditions. It would be interesting to compare this data with that obtained by enforcing the transversality conditions. This feature was examined quite extensively in Chapter IV.

Table I

Results of Simple Motion, Fixed Final Time Problem

Run No.	Closed-Loop Solution	Sampling Step Size Δt	Integration[a] Step Size: Ref.	Integration[a] Step Size: M.R.[b]	Integration[a] Step Size: Real	Final Cost $J(t_f)=1/2(x^2+y^2)_{t_f}$ $=5.$ (Input Data)	% Error Fm. Actual Solution
1	Actual	.5	—	—	.1	1.58086	—
2	Actual	.5	—	—	.01	1.58086	—
3	Near-Optimal	.1	.01	.1	.01	1.59805	1.08
4		.1	.1	.1	.005	1.59805	
5		.1	.01			1.59805	
6		.5	.01			1.68317	6.47
7		.5	.001			1.68327	
8		.5	.005	.1	.005	1.68327	
9		.5	.1	.1	.005	1.68327	
10		.5	.02	.02	.02	1.68361	6.49
11		1.0	.01			1.8332	15.9

a: A variable step fourth order Runge-Kutta method was used to perform all integrations.

b: M.R. stands for the matrix Riccati equations.

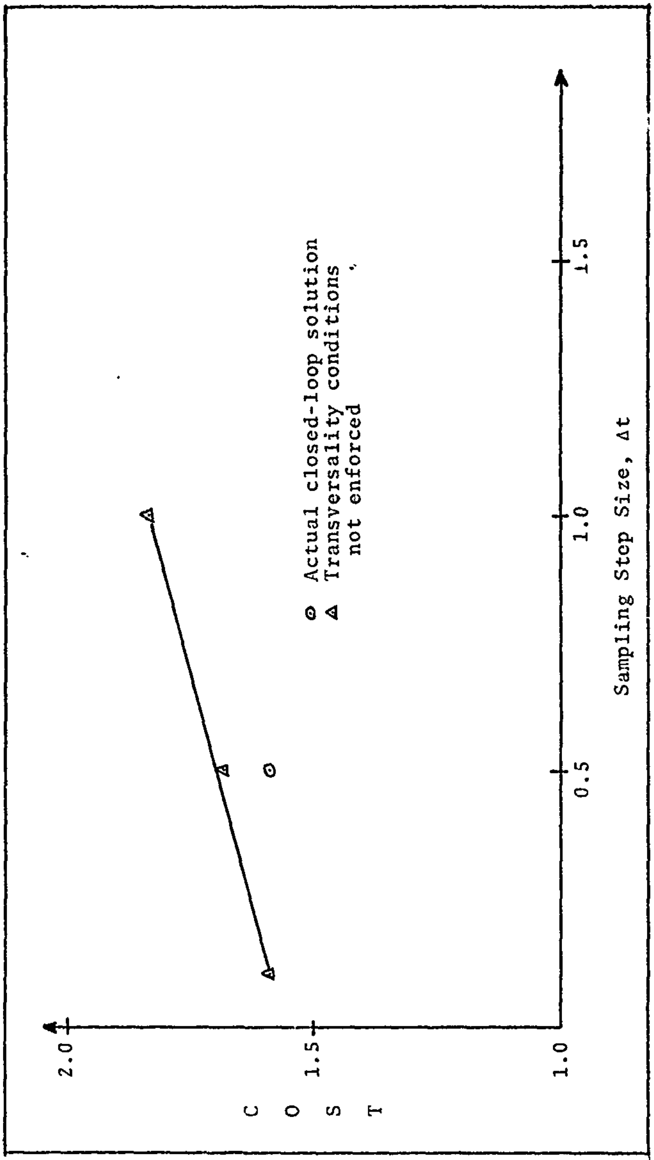

Figure 3. Fixed Final Time Simple Motion Problem, Cost Vs. Sampling Step Size

IV. Pursuit-Evasion Differential Game-Rocket Problem, Fixed Final Time

Statement of the Problem

The formulation of this problem is similar to that of Isaacs (Ref 1 105). But in order to have a slightly more non-linear problem, the drag will vary as a function of the velocity squared, whereas Isaacs represents it as a linear function of the velocity. The pursuer is driven by a fixed thrust magnitude F, but the direction is controlled by ϕ. The evader has simple motion with fixed speed, W. The action takes place in a plane and the payoff is one-half the square of the distance between the players at the end of a fixed final time.

The pursuer will be burdened with a friction drag proportional to the negative of his velocity squared. Without the drag there is no bound on the pursuer's speed. If the friction force is -k times the speed squared, there is a natural limit to the latter equal to F/k. It is the square of the speed the pursuer would come to asymptotically if his thrust propelled him along a straight line.

We will use a moving relative coordinate system centered on the pursuer; see Fig 4, page 34. The object is to find a saddle point of

$$J(t_f) = 1/2(x^2+y^2)\big|_{t\,=\,t_f} \qquad (4\text{-}1)$$

subject to

$$\dot{x} = \dot{x}_e - \dot{x}_p = W \sin\phi - u \tag{4-2}$$

$$\dot{y} = \dot{y}_e - \dot{y}_p = W \cos\theta - v \tag{4-3}$$

$$\dot{u} = F \sin\phi - ku^2 \tag{4-4}$$

$$\dot{v} = F \cos\phi - kv^2 \tag{4-5}$$

where u and v are the respective x and y components of the pursuer's velocity. In this problem, as in the one in Chapter III, there are no terminal constraints or control constraints. The pursuer's control is ϕ and the evader's control is θ.

Necessary Conditions

Applying the necessary conditions for a saddle point solution, the Hamiltonian H is given by

$$H = \lambda_x[W \sin\theta - u] + \lambda_y[W \cos\theta - v] + \lambda_u[F \sin\phi - ku^2] + \lambda_v[F \cos\phi - kv^2] \tag{4-6}$$

The Hamiltonian must be minimized with respect to the pursuer's control ϕ and maximized with respect to the evader's control θ. Therefore it is necessary that

$$\partial H/\partial\phi = 0 \quad , \partial H/\partial\theta = 0 \tag{4-7}$$

$$\partial^2 H/\partial\phi^2 \geq 0 \quad , \partial^2 H/\partial\theta^2 \leq 0 \tag{4-8}$$

Applying these conditions we get

$$\partial H/\partial\phi = F[\lambda_u \cos\phi - \lambda_v \sin\phi] \tag{4-9}$$

In order to satisfy (4-7) and at the same time minimize H we must therefore have

$$\sin\phi^* = -\lambda_u/(\lambda_u^2+\lambda_v^2)^{1/2}, \quad \cos\phi^* = -\lambda_v/(\lambda_u^2+\lambda_v^2)^{1/2} \qquad (4\text{-}10)$$

This also satisfies the first-half of Eq (4-8).

In a similar manner it can be shown that in order to maximize H with respect to the evader's control θ we must have

$$\sin\theta^* = \lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2}, \quad \cos\theta^* = \lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} \qquad (4\text{-}11)$$

This will satisfy the second-half of Eqs (4-7) and (4-8). These optimal open-loop control laws (Eqs (4-10) and (4-11)) also satisfy Eq (2-6).

The costate equations are

$$\dot{\lambda}_x = -\partial H/\partial x = 0 \qquad \lambda_x = \text{constant} \qquad (4\text{-}12)$$

$$\dot{\lambda}_y = -\partial H/\partial y = 0 \qquad \lambda_y = \text{constant} \qquad (4\text{-}13)$$

$$\dot{\lambda}_u = -\partial H/\partial u = \lambda_x + 2k\lambda_u u \qquad (4\text{-}14)$$

$$\dot{\lambda}_v = -\partial H/\partial v = \lambda_y + 2k\lambda_v v \qquad (4\text{-}15)$$

The transversality conditions give

$$\lambda_x(t_f) = x(t_f) \qquad (4\text{-}16)$$

$$\lambda_y(t_f) = y(t_f) \qquad (4\text{-}17)$$

$$\lambda_u(t_f) = \lambda_v(t_f) = 0 \qquad (4\text{-}18)$$

Substituting the controls (ϕ^*,θ^*) into the equations of motion (4-2 through 4-5), we get the following reference differential equations and boundary conditions for the open-loop solution to the problem.

$$\dot{x} = W\lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} - u \quad , \ x(o) \text{ given} \qquad (4\text{-}19)$$

$$\dot{y} = W\lambda_y/(\lambda_x^2+\lambda_x^2)^{1/2} - v \quad , \ y(o) \text{ given} \qquad (4\text{-}20)$$

$$\dot{u} = -F\lambda_u/(\lambda_u^2+\lambda_v^2)^{1/2} - ku^2 \ , \ u(o) \text{ given} \qquad (4\text{-}21)$$

$$\dot{v} = -F\lambda_v/(\lambda_u^2+\lambda_v^2)^{1/2} - kv^2 \ , \ v(o) \text{ given} \qquad (4\text{-}22)$$

From the neighboring extremal path developments of Chapter II, we find that the matrix Riccati equation for this problem is

$$\dot{S} = -SA - A^TS + SBS - C \qquad (4\text{-}23)$$

and

$$\dot{R} = -(A^T - SB)R \qquad (4\text{-}24)$$

$$\dot{Q} = R^TBR \qquad (4\text{-}25)$$

the boundary conditions determined from Eqs (2-19) through (2-21) give

$$S(t_f) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (4\text{-}26)$$

$$R(t_f) = \{0\} \qquad (4\text{-}27)$$

since there are no terminal constraints on the problem, and

$$Q(t_f) = \{0\} \qquad (4\text{-}28)$$

The last two boundary conditions force Eqs (4-24) and (4-25) to be equal to zero, therefore R(t) and Q(t) are constants, and equal to zero at all times. The (4x4) coefficient matrices in Eq (4-23) are from the general linearized state and costate equations (2-11) and (2-12). For this problem

$$A(t) = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & -2ku & 0 \\ 0 & 0 & 0 & -2kv \end{bmatrix} \tag{4-29}$$

$$B(t) = -\begin{bmatrix} \frac{W\lambda_y^2}{(\lambda_x^2+\lambda_y^2)^{3/2}} & \frac{-W\lambda_x\lambda_y}{(\lambda_x^2+\lambda_y^2)^{3/2}} & 0 & 0 \\ \frac{-W\lambda_x\lambda_y}{(\lambda_x^2+\lambda_y^2)^{3/2}} & \frac{W\lambda_x^2}{(\lambda_x^2+\lambda_y^2)^{3/2}} & & \\ 0 & 0 & \frac{-F\lambda_v^2}{(\lambda_u^2+\lambda_v^2)^{3/2}} & \frac{F\lambda_u\lambda_v}{(\lambda_u^2+\lambda_v^2)^{3/2}} \\ 0 & 0 & \frac{F\lambda_u\lambda_v}{(\lambda_u^2+\lambda_v^2)^{3/2}} & \frac{-F\lambda_u^2}{(\lambda_u^2+\lambda_v^2)^{3/2}} \end{bmatrix} \tag{4-30}$$

$$C(t) = -\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2k\lambda_u & 0 \\ 0 & 0 & 0 & 2k\lambda_v \end{bmatrix} \tag{4-31}$$

Equation (2-22) reduces to

$$\delta\lambda(t) = S(t)\ \delta x(t) \tag{4-32}$$

which reflects the effect on the costates due to changes in the state variables.

Program Algorithm

Just as in the Simple-Motion problem of Chapter III, we want to start with the given initial conditions. Then solve the TPBVP to obtain a reference trajectory. The availability of this reference trajectory is a necessary feature of this method for generating a closed-loop solution to differential games. However, this algorithm bypassed solving the TPBVP and instead used a backward integration as a means of generating an optimal open-loop solution.

(1) As before, a fixed final time of $t_f = 5.0$ was assumed along with the following input data

$$x(t_f) = 1. \qquad W = 1.$$
$$y(t_f) = 2. \qquad F = 2.$$
$$u(t_f) = 2. \qquad k = .1$$
$$v(t_f) = 4. \qquad t_0 = 0.$$

Basically we are starting the program from step (b) of the algorithm in Chapter II. Therefore, the transversality conditions Eqs (4-16) to (4-18) may be written as

$$\lambda_x(t_f) = 1. \qquad \lambda_u(t_f) = 0.$$
$$\lambda_y(t_f) = 2. \qquad \lambda_v(t_f) = 0.$$

Having both $\lambda_u(t_f)$ and $\lambda_v(t_f)$ equal to zero presents a problem when evaluating $\dot{u}(t_f)$ and $\dot{v}(t_f)$ from Eqs (4-21) and (4-22). To avoid dividing by zero, we apply L'Hospital's rule to the terms that would have zero in the denominator and we obtain the following expressions for λ_u and λ_v as functions of λ_x and λ_y.

$$\lambda_u = \lambda_x \delta t$$

$$\lambda_v = \lambda_t \delta d$$

where $\delta t \equiv t_f - t_f$

Therefore

$$\lambda_u/(\lambda_u^2+\lambda_v^2)^{1/2} = \lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2}$$

and

$$\lambda_v/(\lambda_u^2+\lambda_v^2)^{1/2} = \lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2}$$

(2) Now integrate the reference state, costate and matrix Riccati equations backwards from t_f to t_o. This gives us the reference open-loop trajectory represented in Fig 1 by the curve from point 3 to 1. We also have the reference values of the states and S(t) at the sampling time t_1 which will be needed later to compute $\delta\lambda(t_1)$.

(3) With the initial conditions for the problem now specified, compute the "real" trajectory (from point 1 to 4) by substituting the actual strategies of the players into the differential equations of motion (Eqs 4-2 through 4-5) and integrate forward from t_o to t_1 the sampling time. For this problem we assumed the pursuer played his optimal strategy (Eq 4-10) while the evader played the constant

non-optimal control of $\theta = 0°$. Using (ϕ^*, θ) we get the following "real" equations of motion

$$\dot{x} = -u \tag{4-33}$$

$$\dot{y} = W-v \tag{4-34}$$

$$\dot{u} = -F\lambda_u/(\lambda_u^2+\lambda_v^2)^{1/2} - ku^2 \tag{4-35}$$

$$\dot{v} = -F\lambda_v/(\lambda_u^2+\lambda_v^2)^{1/2} - kv^2 \tag{4-36}$$

(4) Now follow the same procedures laid out in steps (d) through (g) of Chapter II. The only thing that is different is the number of differential equations involved. In this problem there are twenty-four differential equations (4 state, 4 costate, and 16 matrix Riccati) which are integrated backwards in step 2 as opposed to four for the Simple-Motion problem.

Results and Analysis

Based on the input data, if both players played optimally, the cost $J(t_f)$ would be equal to 2.5. But, we assumed the evader did not play optimally. Therefore for a zero-sum differential game the pursuer should gain what the evader loses. We would expect the cost at t_f to be less than 2.5. The results, as shown in Table II, page 37, and Fig 5, page 38, do not completely bear this out. We see that for values of Δt less than .2 the cost shoots up instead of approaching some minimum optimal cost. This is due to the fact that both players are actually playing nonoptimally. The algorithm provides the pursuer with a "near" optimal strategy as opposed

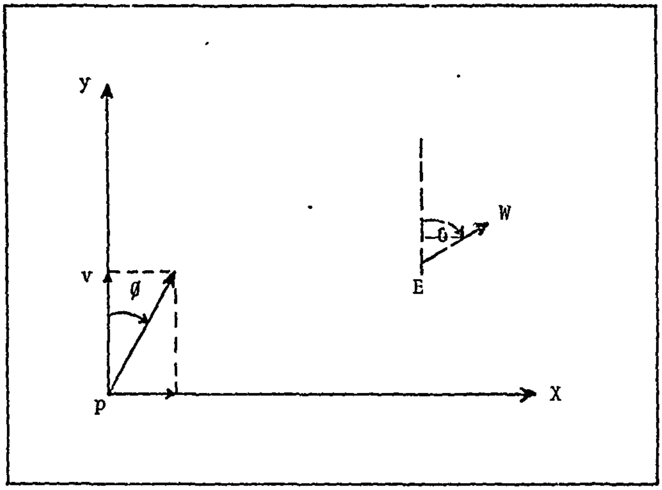

Figure 4. Two-Dimensional Rocket Problem

to the optimal strategy. In other runs it was found that integration step sizes equal to or greater than .01 provided too coarse of an integration to produce useful results.

In runs number nine and ten, the results obtained by using a simple predict-correct integration method were compared with that from the variable step fourth order Runge-Kutta method. It was felt that the resulting costs were close enough to justify using the method which was easiest to program although it provided a much finer integration than was necessary and caused the program to take longer to run. Therefore, the Runge-Kutta integration method was used for all subsequent runs.

A set of runs was used to study the effect of enforcing the transversality conditions. The first runs were made without enforcing the transversality conditions. They are the runs in Table II with no asterisk attached to the final cost. For these runs, the final values of the costates obtained from the forward integration in step a were used in the next cycle as starting values in step c.

For the next set of runs (those with a single asterisk on the final cost in Table II) the transversality conditions were enforced. That is in step b .

$$\lambda_x(t_f) = x(t_f)$$
$$\lambda_y(t_f) = y(t_f)$$

For the final set of runs, the following substitutions were made in step b to obtain a modified enforced transversality condition which would be used in step c of the next cycle.

$$\lambda_x(t_f) = 1/2[x(t_f)+\lambda_x(t_f)]$$
$$\lambda_y(t_f) = 1/2[y(t_f)+\lambda_y(t_f)]$$

As shown in Table II and Fig 5, the results from all three sets of runs were very close. For this problem, enforcing the transversality conditions appears to have had negligible influence on the final cost. This may have been due to the relatively small value of t_f. Had it been assumed to be much greater than 5.0, a difference in the costs may have been detected.

According to Fig 5, there is a relatively large range of sampling step size which provides a final cost very near the minimum cost. Therefore, by choosing a sampling step size Δt in the range of 1.0 we can get a very near optimal closed-loop solution.

Table II

Results of Rocket, Fixed Final Time Problem

Run No.	Sampling Step Size Δt	Integration Step Size	Final Cost $J(t_f) = 1/2(x^2+y^2)_{t_f}$ $t_f = 2.5$ (Input Data)
1	.05	.001	2.8370
2	.05	.001	2.6732**
3	.1	.001	2.2752
4	.1	.001	2.2336**
5	.2	.0005	1.9933
6	.2	.001	2.0796
7	.2	.001	2.0663*
8	.2	.001	2.0681**
9[a]	.5	.0005	1.9459
10	.5	.0005	1.9621
11	.5	.001	1.9958
12	.5	.001	1.9917*
13	.5	.001	1.9944**
14	1.0	.0005	1.9748
15	1.0	.001	1.9940
16	1.0	.001	2.0025*
17	1.0	.001	2.0023**
18	1.25	.001	2.0057
19	1.25	.001	2.0257*
20	1.25	.001	2.0193**
21	2.5	.001	2.1463

a: For this run a simple predict-correct integration scheme was used. All other runs used a fourth order Runge-Kutta integration method.

*: Transversality conditions enforced.
**: Modified transversality conditions enforced.

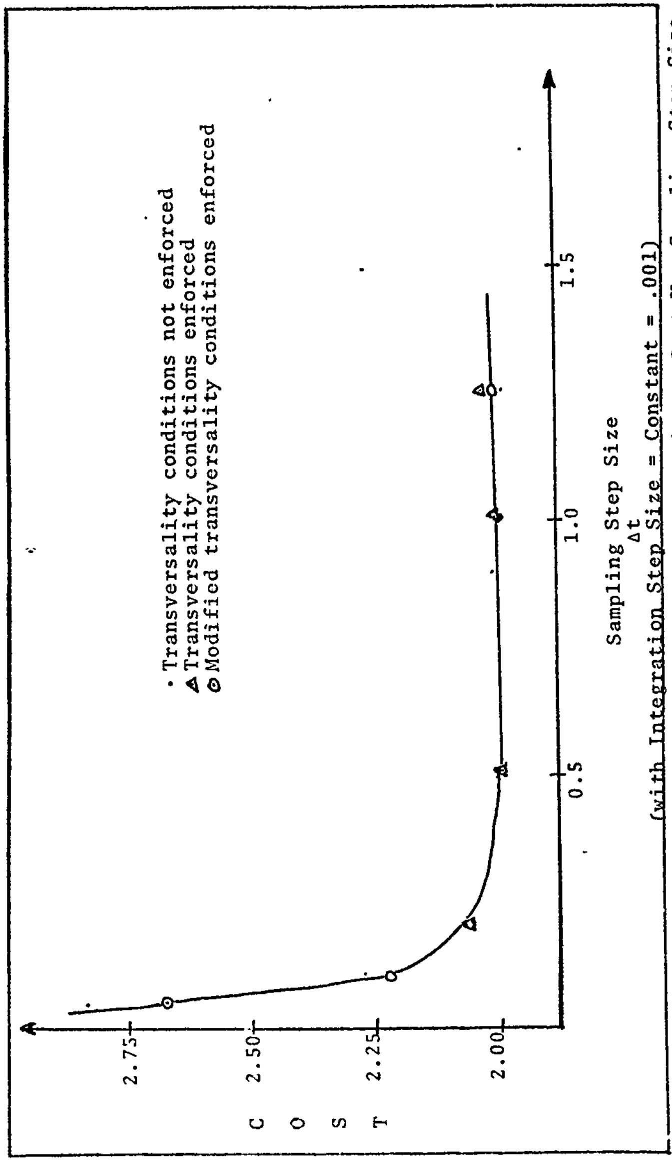

Figure 5. Results of Fixed Final Time Rocket Problem, Cost Vs. Sampling Step Size

V. Pursuit-Evasion Differential Game-Simple Motion Problem, Free Final Time with Terminal Constraint

Statement of the Problem

The basic problem is the same as that of Chapter III, the pursuer and evader have constant velocity, unrestricted simple plainer motion, with the pursuer having the speed advantage. The game will be over not at a fixed final time but when the terminal constraint is satisfied. That is when

$$\psi[x(t_f)] = 1/2[x^2(t_f) + y^2(t_f)]-1/2 = 0 \qquad (5\text{-}1)$$

This may be pictured as a capture circle with the center at the pursuer's position. The terminal constraint ψ will be satisfied if the evader is forced inside this unit circle. The object of the game for the pursuer is to accomplish this in the minimum time possible. The evader, when playing his optimal strategy, aims to prevent capture or at least delay it as long as possible. In other words, we want to determine the saddle point of

$$J(t_f) = \int_{t_o}^{t_f} dt = t_f - t_o \qquad (5\text{-}2)$$

which means we have the minimum time to capture. The game is subject to the same relative differential equations of motion

$$\dot{x} = V_e \cos v - V_p \cos u \qquad (5\text{-}3)$$

$$\dot{y} = V_e \sin v - V_p \sin u \qquad (5\text{-}4)$$

Necessary Conditions

Applying the necessary conditions for a saddle point solution, the Hamiltonian H may be written from Eq (2-6) as

$$H = 1+\lambda_x[V_e \cos v - V_p \cos u]+\lambda_y[V_e \sin v - V_p \sin u] \quad (5\text{-}5)$$

It can be seen, that as far as the controls (u,v) are concerned, H for this problem is the same as Eq (3-4), therefore the same (u*,v*) will provide a saddle point solution for both problems. These controls are

$$\sin u^* = \lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2}, \quad \cos u^* = \lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} \quad (5\text{-}6)$$

$$\sin v^* = \lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2}, \quad \cos v^* = \lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} \quad (5\text{-}7)$$

The costate equations are

$$\dot{\lambda}_x = -\partial H/\partial x = 0 \quad (5\text{-}8)$$

$$\dot{\lambda}_y = -\partial H/\partial y = 0 \quad (5\text{-}9)$$

and the transversality conditions (from Eq 2-9) give

$$\lambda_x(t_f) = \nu x(t_f) \quad (5\text{-}10)$$

$$\lambda_y(t_f) = \nu y(t_f) \quad (5\text{-}11)$$

where ν is a constant Lagrange multiplier.

Now substituting (u*,v*) in Eqs (5-3) and (5-4) we get the reference differential equations for the open-loop solution to the problem.

$$\dot{x} = (V_e-V_p)\lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} \quad (5\text{-}12)$$

$$\dot{y} = (V_e-V_p)\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} \quad (5\text{-}13)$$

We see from the extremal path development in Chapter II, that for the case of unspecified final time the changes in the costates $\delta\lambda(t)$ are a function of $\delta x(t)$, $d\nu$ and dt_f. Equation (2-30) applies here

$$\delta\lambda(t) = S(t)\,\delta x(t) + R(t)\,d\nu + m(t)dt_f \tag{5-14}$$

The additional necessary condition of Eq (2-24) must also be satisfied. For this problem we see that

$$\Phi = \nu[1/2(x^2+y^2)-1/2] \tag{5-15}$$

$$d\Phi/dt = \frac{\partial\Phi}{\partial t} + \frac{\partial\Phi}{\partial x}\dot{x} + \frac{\partial\Phi}{\partial y}y = \nu(x\dot{x} + y\dot{y}) \tag{5-16}$$

this gives

$$\Omega(t_f) = \nu(V_e-V_p)[(x\lambda_x+y\lambda_y)/(\lambda_x^2+\lambda_y^2)^{1/2}]_{t=t_f} + 1 = 0 \tag{5-17}$$

Substitute Eqs (5-10) and (5-11) into (5-17) and we get the value of the arbitrary constant

$$\nu = \{(V_p-V_e)[x^2(t_f) - y^2(t_f)]^{1/2}\}^{-1} \tag{5-18}$$

We will now determine the terms in Eqs (2-26) to (2-28). From Eq (5-15) we get

$$[\partial^2\phi/\partial x^2]_{t_f} = \begin{bmatrix} \frac{\partial^2\phi}{\partial} & \frac{\partial^2\phi}{\partial x\partial y} \\ \frac{\partial^2\phi}{\partial x\partial y} & \frac{\partial^2\phi}{\partial y^2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}_{t = t_f} \tag{5-19}$$

Using Eq (5-17) we get

$$\left[\frac{\partial\Omega}{\partial x}\right]^T_{t_f} = \begin{bmatrix} \frac{\partial\Omega}{\partial x} \\ \\ \frac{\partial\Omega}{\partial y} \end{bmatrix} = \nu(V_e - V_p)/(\lambda_x^2 + \lambda_y^2)^{1/2} \begin{bmatrix} \lambda_x \\ \\ \lambda_y \end{bmatrix}_{t=t_f} \quad (5\text{-}20)$$

and

$$d\Omega/dt = \partial\Omega/\partial t + [\partial\Omega/\partial x, \partial\Omega/\partial y] \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \nu(V_e - V_p)^2 \Big|_{t=t_f} \quad (5\text{-}21)$$

Equation (5-1) gives

$$\left[\frac{\partial\Omega}{\partial x}\right]^T_{t_f} = \begin{bmatrix} \frac{\partial\psi}{\partial x} \\ \\ \frac{\partial\psi}{\partial y} \end{bmatrix} = \nu^{-1} \begin{bmatrix} \lambda_x \\ \\ \lambda_y \end{bmatrix}_{t = t_f} \quad (5\text{-}22)$$

and

$$d\psi/dt\big|_{t_f} = \partial\psi/\partial t + [\partial\psi/\partial x, \partial\psi/\partial y] \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix}$$

$$= (V_e - V_p)/\nu[\lambda_x^2(t_f) + \lambda_y^2(t_f)]^{1/2} \quad (5\text{-}23)$$

Using the above development and substituting into Eqs (2-33) to (2-38) we get the following differential equations and boundary conditions for this problem.

$$\dot{S} = SBS \ , \ S(t_f) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (5\text{-}24)$$

$$\dot{R} = SBR \ , \ R(t_f) = \begin{bmatrix} x(t_f) \\ y(t_f) \end{bmatrix} \quad (5\text{-}25)$$

$$\dot{Q} = R^T BR, \ Q(t_f) = 0 \quad (5\text{-}26)$$

$$\dot{m} = SBm \ , \ m(t_f) = \left(\frac{\partial\Omega}{\partial x}\right)^T_{t_f} \quad (5\text{-}27)$$

$$\dot{n} = R^T Bm, \ n(t_f) = \left(\frac{d\psi}{dt}\right)_{t_f} \quad (5\text{-}28)$$

$$\dot{\alpha} = m^T Bm, \ \alpha(t_f) = \left(\frac{d\Omega}{dt}\right)_{t_f} \quad (5\text{-}29)$$

where

$$B(t) = -(V_e - V_p)/(\lambda_x^2 + \lambda_y^2)^{1/2} \begin{bmatrix} \lambda_y^2 & -\lambda_x\lambda_y \\ -\lambda_x\lambda_y & \lambda_x^2 \end{bmatrix} \quad (5\text{-}30)$$

Problem Algorithm

As mentioned in previous chapters, all real problems start with some given initial conditions. The TPBVP must then be solved to provide the needed reference open-loop trajectory to be used in this method for generating closed-loop solutions. However, as in Chapter III, rather than solve the TPBVP given specific initial conditions, this algorithm uses a backward integration as a means of generating a reference optimal open-loop solution.

(1) To terminate the game at some minimum time t_f, the terminal constraint must be satisfied. Therefore we will assume the following input data.

$$t_f = 5.0 \qquad t_o = 0.$$
$$x(t_f) = 1.0 \qquad V_e = 1.0$$
$$y(t_f) = 0 \qquad V_p = 2.0$$

This allows us to solve for the terminal conditions of the costate, matrix Riccati and auxiliary differential equations.

(2) The program then integrates the reference states, costates, matrix Riccati and auxiliary differential equations backwards from t_f to t_1, at which time the reference states $x(t_1)$, $S(t_1)$, $R(t_1)$, $Q(t_1)$, $m(t_1)$, $n(t_1)$ and

$\alpha(t_1)$ are stored. The backward integration is then continued to t_o in order to determine the initial state and costate values.

(3) With this accomplished, integrate the "real" equations of motion forward from t_o to t_1. In this problem we assume the same real equations of motion as used in Chapter III, that is

$$\dot{x} = -V_p\lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} \tag{5-31}$$

$$\dot{y} = V_e - V_p\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} \tag{5-32}$$

(4) Now compute the difference between the "real" and "reference" states at t_1.

$$\delta x(t_1) = x_{Real}(t_1) - x_{Ref.}(t_1) \tag{5-33}$$

$$\delta y(t_1) = y_{Real}(t_1) - y_{Ref.}(t_1) \tag{5-34}$$

(5) There is now enough information to use Eqs (2-39) and (2-40) to compute $d\nu$ and dt_f, which then allows us to solve for $\delta\lambda(t_1)$ using (2-43)

(6) Next, compute the new costates and t_f by using

$$\lambda(t_1)_{new} = \lambda(t_1)_{old} + \delta\lambda(t_1) \tag{5-35}$$

$$t_f = t_{f\ old} + dt_f \tag{5-36}$$

(7) In this step, integrate the reference state and costate equations forward to determine the states at some new final time. Here we have a choice between two

approaches. We could stop the forward integration at the computed t_f, or we could stop whenever the terminal constraint is satisfied. that is whenever $\psi = 0$. The latter approach was used for this problem.

(8) Now having determined the new final states, enforce the transversality conditions by recomputing the terminal conditions on the costate, matrix Riccati and auxiliary difference equations. Then go back to step 2 and repeat the cycle.

Results and Analysis

The results of this problem are presented in Table III, page 47, and Fig 6, page 48. The terminal constraint for this free final time problem was assumed to be a circle around the pursuer of radius equal to one. This was sufficient for capture to occur in all cases. It has not been determined just how small this circle could be and still assure capture. This would be good to know in a dogfight situation, where the minimum radius of capture may represent the minimum firing range for the weapons the pursuer has on the aircraft. To close inside this minimum range would be a mistake.

From the data we see that there is a broad range of integration step sizes for a specific sampling interval which will provide a fairly uniform final cost. Therefore we could use the larger step size 0.1 to decrease integration time and the cost would change by less than 5% of the average value for the case where the sampling interval is equal to 0.5.

Three runs were made to determine the effect of enforcing the transversality conditions. Figure 5 shows that for sampling step sizes of 0.2 and 0.5 there appears to be very little effect. But for a sampling interval of 1.0 there is definitely a reduction in final cost due to enforcing the transversality conditions.

Here again, as in Chapter III, the slope of the curve seems to indicate that as the sampling step size approaches zero, a limiting minimum cost is approached.

Table III

Results of Simple Motion, Free Final Time Problem

Run No.	Sampling Step Size Δt	Integration Step Size	Computed Final Time Cost, $J(t_f)$ = 5. (Input Data)
1	.2	.001	3.1590
1*	.2	.001	3.1880*
2	.2	.005	3.1650
3	.2	.01	3.1700
4	.2	.1	3.300
5	.5	.001	3.3370
5*	.5	.001	3.3490*
6	.5	.005	3.3400
7	.5	.01	3.3500
8	.5	.1	3.5000
9	1.0	.001	3.7580
9*	1.0	.001	3.5940*
10	1.0	.005	3.7600
11	1.0	.01	3.7700
12	1.0	.1	3.9000

*: Transversality conditions enforced.

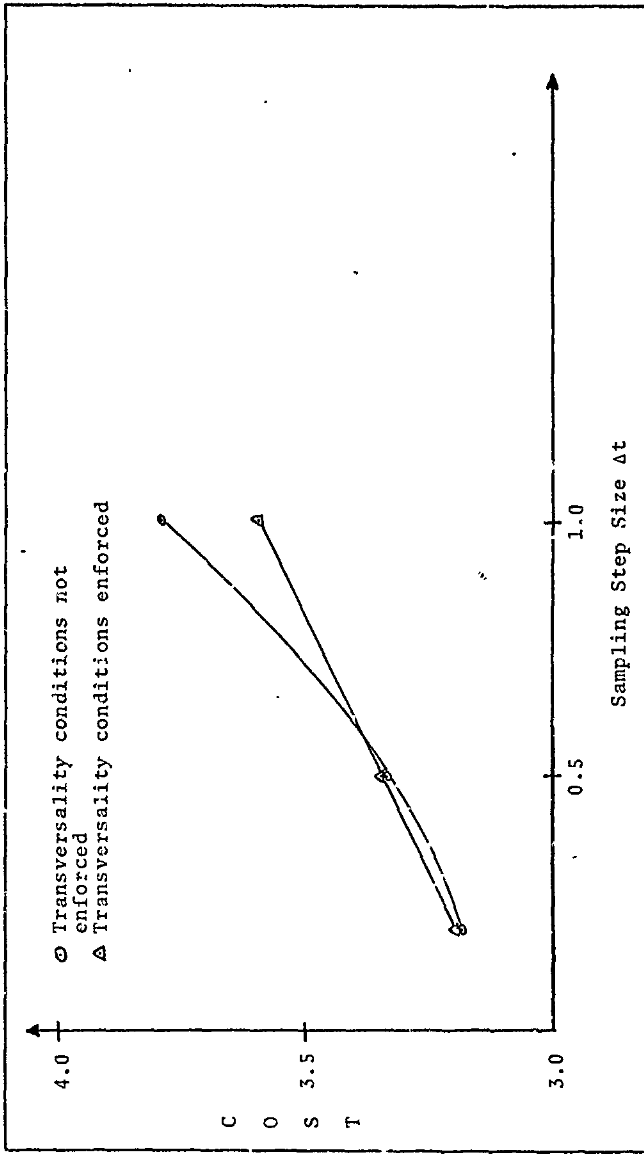

Figure 6. Free Final Time Simple Motion Problem, Cost Vs. Sampling Step Size

VI. Pursuit-Evasion Differential Game-Rocket Problem, Free Final Time with Terminal Constraint

Statement of the Problem

The problem is basically the same as that of Chapter IV, except that the game will terminate not at a fixed final time but when the terminal constraint, $\psi(t_f)$ is satisfied. The terminal constraint will be

$$\psi[x(t_f)] = 1/2[x^2(t_f) + y^2(t_f)] - 1/2 = 0 \qquad (6\text{-}1)$$

This represents a unit circle, centered on the pursuer, in a relative coordinate system. Therefore, the game will be over when the evader is forced inside this unit circle or in other words when capture occurs. The object will be to capture in minimum time, which means we must determine the saddle point solution of

$$J(t_f) = \int_{t_o}^{t_f} dt = t_f - t_o \qquad (6\text{-}2)$$

subject to

$$\dot{x} = W \sin\theta - u \qquad (6\text{-}3)$$

$$\dot{y} = W \cos\theta - v \qquad (6\text{-}4)$$

$$\dot{u} = F \sin\phi - ku^2 \qquad (6\text{-}5)$$

$$\dot{v} = F \cos\phi - kv^2 \qquad (6\text{-}6)$$

where, as in Chapter IV, u and v are the respective x and y components of the pursuer's velocity and ϕ and θ are the respective pursuer's and evader's controls.

Necessary Conditions

From Eq (2-6) the Hamiltonian may be written as

$$H = 1+\lambda_x[W \sin\theta - u] + \lambda_y[W \cos\theta - v]$$

$$+ \lambda_x[F \sin\phi - ku^2] + \lambda_y[F \cos\phi - kv^2] \qquad (6\text{-}7)$$

It can be seen, that as far as the controls (ϕ,θ) are concerned, H for this problem is the same as Eq (4-6), therefore the same (ϕ^*,θ^*) will provide a saddle point solution for both problems. These controls are

$$\sin\phi^* = -\lambda_u/(\lambda_u^2+\lambda_v^2)^{1/2} \ , \ \cos\phi^* = -\lambda_v/(\lambda_u^2+\lambda_v^2)^{1/2} \qquad (6\text{-}8)$$

$$\sin\theta^* = \lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} \ , \ \cos\theta^* = \lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} \qquad (6\text{-}9)$$

The costate equations are the same as for Chapter IV,

$$\dot{\lambda}_x = -\partial H/\partial x = 0 \qquad (6\text{-}10)$$

$$\dot{\lambda}_y = -\partial H/\partial y = 0 \qquad (6\text{-}11)$$

$$\dot{\lambda}_u = -\partial H/\partial u = \lambda_x + 2k\lambda_u u \qquad (6\text{-}12)$$

$$\dot{\lambda}_v = -\partial H/\partial v = \lambda_y + 2k\lambda_v v \qquad (6\text{-}13)$$

and using Eq (2-9), the transversality conditions are

$$\lambda_x(t_f) = \nu x(t_f) \qquad (6\text{-}14)$$

$$\lambda_y(t_f) = \nu y(t_f) \qquad (6\text{-}15)$$

$$\lambda_u(t_f) = \lambda_v(t_f) = 0 \qquad (6\text{-}16)$$

where ν is a constant Lagrange multiplier.

Using the optimal controls (ϕ^*,θ^*) in the equations of motion, we get the same reference differential equations for the open-loop solution as in Chapter IV.

$$\dot{x} = W\lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} - u \tag{6-17}$$

$$\dot{y} = W\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} - v \tag{6-18}$$

$$\dot{u} = -F\lambda_u/(\lambda_u^2+\lambda_v^2)^{1/2} - ku^2 \tag{6-19}$$

$$\dot{v} = -F\lambda_v/(\lambda_u^2+\lambda_v^2)^{1/2} - kv^2 \tag{6-20}$$

Making use of the neighboring extremal path development of Chapter II, we see that for the case of unspecified final time, the changes in the costates $\delta\lambda(t)$ are given by Eq (2-30) as a function of $\delta x(t)$, $d\nu$ and dt_f.

$$\delta\lambda(t) = S(t)\ \delta x(t) + R(t)d\nu + m(t)dt_f \tag{6-21}$$

In order to satisfy the additional necessary condition (Eq 2-24) which must be satisfied, we find that

$$\Phi = \nu[1/2(x^2+y^2)-1/2] \tag{6-22}$$

$$\begin{aligned} d\Phi/dt &= \partial\Phi/\partial t+(\partial\Phi/\partial x)\dot{x}+(\partial\Phi/\partial y)\dot{y}+(\partial\Phi/\partial u)\dot{u}+(\partial\Phi/\partial v)\dot{v} \\ &= [x\dot{x} + y\dot{y}] \end{aligned} \tag{6-23}$$

This gives

$$\begin{aligned} \Omega(t_f) = \nu\{x[W\lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} - u] \\ + y[W\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} - v]\}_{t=t_f} + 1 = 0 \end{aligned} \tag{6-24}$$

Substitute Eqs (6-14) and (6-15) into (6-24) and we get the value of the arbitrary constant

$$\nu = -\{W(x^2+y^2)^{1/2} - (ux+vy)\}^{-1}_{t = t_f} \qquad (6\text{-}25)$$

We will now determine the terms in Eqs (2-26) to (2-28). From Eq (6-22) we get

$$\partial^2\phi/\partial x^2 = \nu \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad (6\text{-}26)$$

Using Eq (6-24) we get

$$[\partial\Omega/\partial x]^T_{t_f} = \nu \begin{bmatrix} W\lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} - u \\ W\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} - v \\ -\lambda_x/\nu \\ -\lambda_y/\nu \end{bmatrix} \qquad (6\text{-}27)$$

and

$$d\Omega/dt|_{t_f} = \nu\{[W\lambda_x/(\lambda_x^2+\lambda_y^2)^{1/2} - u]^2+[W\lambda_y/(\lambda_x^2+\lambda_y^2)^{1/2} - v]^2$$
$$+ x[F\lambda_u/(\lambda_u^2+\lambda_v^2)^{1/2} + ku^2]+y[F\lambda_v/(\lambda_x^2+\lambda_y^2)^{1/2} + kv^2]\}|_{t=t_f} \qquad (6\text{-}28)$$

Equation (6-1) gives

$$[d\psi^T/dt]^T_{t_f} = \nu^{-1} \begin{bmatrix} \lambda_x \\ \lambda_y \\ 0 \\ 0 \end{bmatrix}_{t = t_f} \qquad (6\text{-}29)$$

and

$$d\psi/dt|_{t_f} = \nu^{-1}[W(\lambda_x^2+\lambda_y^2)^{1/2}-(u\lambda_x+v\lambda_y)]|_{t=t_f} \qquad (6\text{-}30)$$

Using these equations along with Eqs (2-33) to (2-38) we get the following matrix Riccati and auxiliary differential equations for this problem.

$$\dot{S} = -SA - A^TS + SBS - C \quad ,$$

$$S(t_f) = \nu \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}_{t_f} \qquad (6\text{-}31)$$

$$\dot{R} = -(A^T-SB)R, \; R(t_f) = \nu^{-1} \begin{bmatrix} \lambda_x \\ \lambda_y \\ 0 \\ 0 \end{bmatrix}_{t_f} \qquad (6\text{-}32)$$

$$\dot{Q} = R^TBR, \; Q(tf) = \{0\} \qquad (6\text{-}33)$$

$$\dot{m} = -(A^T-SB)m, \; m(t_f) = (\partial\Omega/\partial x)^T_{t_f} \qquad (6\text{-}34)$$

$$\dot{n} = R^TBm, \; n(tf) = (d\psi/dt)_{t_f} \qquad (6\text{-}35)$$

$$\dot{\alpha} = m^TBm, \; \alpha(tf) = (d\Omega/dt)_{t_f} \qquad (6\text{-}36)$$

The coefficient matrices of Eqs (2-11) and (2-12) are

$$A(t) = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & -2ku & 0 \\ 0 & 0 & 0 & -2kv \end{bmatrix} \tag{6-37}$$

$$B(t) = -\begin{bmatrix} \frac{W\lambda_x^2}{(\lambda_x^2+\lambda_y^2)^{3/2}} & \frac{-W\lambda_x\lambda_y}{(\lambda_x^2+\lambda_y^2)^{3/2}} & 0 & 0 \\ \frac{-W\lambda_x\lambda_y}{(\lambda_x^2+\lambda_y^2)^{3/2}} & \frac{W\lambda_y^2}{(\lambda_x^2+\lambda_y^2)^{3/2}} & 0 & 0 \\ 0 & 0 & \frac{-F\lambda_v^2}{(\lambda_u^2+\lambda_v^2)^{3/2}} & \frac{F\lambda_u\lambda_v}{(\lambda_u^2+\lambda_v^2)^{3/2}} \\ 0 & 0 & \frac{F\lambda_u\lambda_v}{(\lambda_u^2+\lambda_v^2)^{3/2}} & \frac{-F\lambda_u^2}{(\lambda_u^2+\lambda_v^2)^{3/2}} \end{bmatrix} \tag{6-38}$$

$$C(t) = -\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2k\,u & 0 \\ 0 & 0 & 0 & 2k\,v \end{bmatrix} \tag{6-39}$$

Problem Algorithm

The computer program for this unspecified final time Rocket problem follows the same steps outlined in Chapter V. The program input data was

$$t_f = 5.0 \qquad x(t_f) = 1.0$$
$$W = 1.0 \qquad y(t_f) = 0.0$$
$$F = 2.0 \qquad u(t_f) = 4.0$$
$$k = 1 \qquad v(t_f) = 0.0$$

and the "real" equations of motion were assumed to be the same as those used in Chapter IV for the fixed final time Rocket problem. They were assumed to be

$$\dot{x} = -u \tag{6-40}$$

$$\dot{y} = W-v \tag{6-41}$$

$$\dot{u} = -F\lambda_u/(\lambda_u^2+\lambda_v^2)^{1/2} - ku^2 \tag{6-42}$$

$$\dot{v} = -F\lambda_v/(\lambda_u^2+\lambda_v^2)^{1/2} - kv^2 \tag{6-43}$$

Results and Analysis

The results from this free final time problem are presented in Table IV, page 56, and Fig 7, page 57. For this data, the capture circle radius R equals $\sqrt{2}$. It was found that capture would not occur if R = 1. Therefore the minimum radius that assures capture is someplace between the two values but it was not specifically determined.

The computed final time t_f is the sum of t_f the actual final time from the previous iteration and the computed value of dt_f at the final sampling time. The actual capture (or final) time is determined by integrating the "real" equations of motion forward from the last sampling time until the terminal constraint is satisfied ($\psi = 0$). Except for the cases where the sampling step size equals 0.2, the computed final time appears to provide an optimistic final cost as compared to the actual final cost. This is another example of the computational errors introduced by the larger sampling step sizes. We see that as the sampling step size decreases the resulting final cost also decreases.

From Fig 7, we can see that by enforcing the transversality conditions we achieve a definite reduction in

Table IV

Results of Rocket, Free Final Time Problem

Run No.	Sampling Step Size Δt	Integration Step Size	Computed Final Time $t_f = t_f + dt_f$	Actual Final Time ($\psi = 0$)
1	.2	.001	4.0934	4.0830
1*	.2	.001	4.1118	4.1110
2	.5	.001	4.1898	4.2020
2*	.5	.001	4.1683	4.1750
3	1.0	.001	4.2575	4.5430
3*	1.0	.001	4.2074	4.2840

*: Transversality conditions enforced

final cost, especially for the larger sampling step sizes. We are in essence starting with an updated two-point boundary-value problem each time we enforce the transversality conditions.

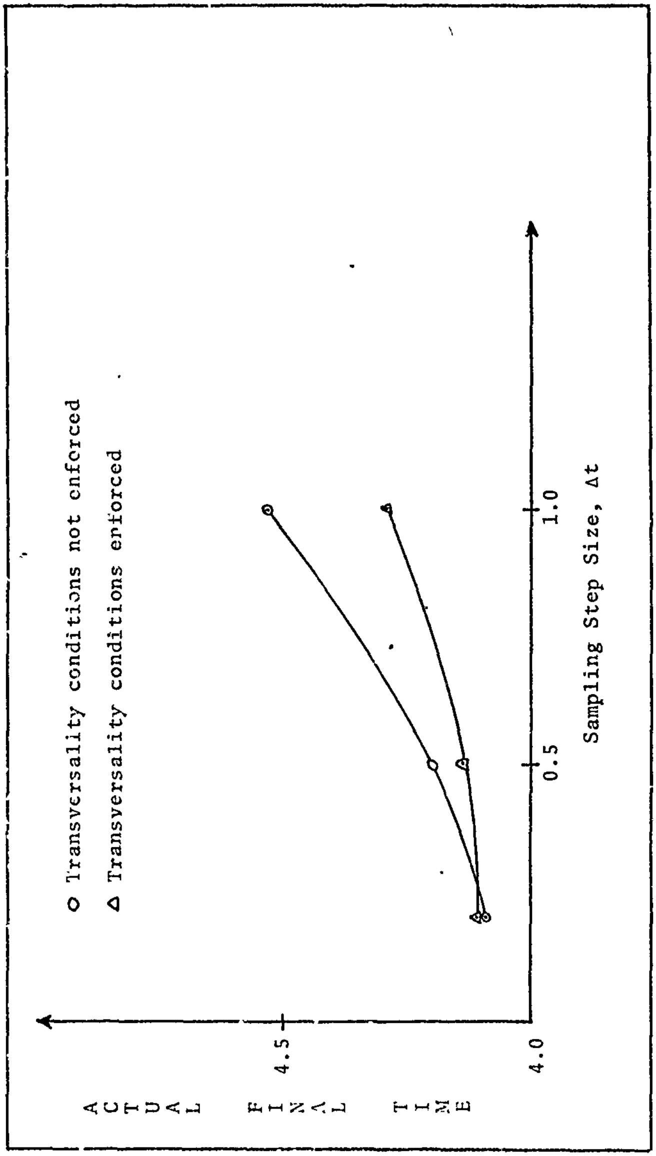

Figure 7. Free Final Time Rocket Problem

VII. Conclusions and Recommendations

There are many questions left unanswered by this brief attempt at applying the proposed method of generating near-optimal closed-loop solutions to a few example differential game problems. But some general observations can be made. There are a number of factors that influence the final cost in differential games. Among them are the sampling step size, the size of the capture circle in the unspecified final time problems and enforcing the transversality conditions. It was found that generally the final cost varied directly with the sampling step size and inversely with the size of the capture circle. Enforcing the transversality conditions resulted in decreased final cost. In some problems it appears as though the integration step sizes from 0.1 to 0.001 had very little effect on the final cost. It appears as though one can use a coarse integration and yet not affect the final cost significantly. Therefore it may be possible by using a hybrid computer to approach "real time" closed-loop solutions. The analog computer would provide the coarse but rapid integration of the differential equations. In any practical application of this method, we would want to update the TPBVP as often as possible. But the minimum sampling step size Δt is limited by the time required to perform the numerical calculations needed to update the solution. During the updating interval, the players must base their strategies on the "best" available information (the state of the game at the beginning of the last updated interval) as opposed to the perfect or complete information based on present states. This should still be adequate provided the states of the game do not change too rapidly.

Although this method for generating near-optimal closed-loop solutions is most applicable to differential game problems, it would also be applicable in many optimal control problems. One of the main limitations of the method is that we must have the solution to the TPBVP. At times, solving the TPBVP could be quite an accomplishment in itself. Also all previous discussion was limited to solutions in the small. Even then, we did not begin to examine the many problems available through various possible combinations of final time, control constraints and terminal constraints. To say the least, there is a lot more work to be done in this area.

Bibliography

1. Isaacs, R. Differential Games. New York: John Wiley and Sons, Inc., 1965.

2. Bryson, A. E. and Y. C. Ho. Applied Optimal Control. Waltham, Massachusetts: Blaisdell Publishing Company, 1969.

Vita

Percy J. Gros, Jr. was born on 12 May 1939 in Thibodaux, Louisiana. He graduated from high school in 1957 and attended Francis T. Nicholls State College in Thibodaux for two years. In 1961 he received his degree in Mechanical Engineering from Louisiana State University and his USAF Commission through the OTS Program. He was assigned to the Aeromedical Research Laboratory, Holloman AFB, New Mexico, for two years prior to entering pilot training at Vance AFB, Oklahoma. He received his wings in 1965 and then flew F4's in the Tactical Air Command and in Southeast Asia until 1970. He is presently enrolled in the Graduate Astronautics Course at the Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio.

Permanent address: 407 Green Street
Thibodaux, Louisiana 70301

This thesis was typed by Mrs. Anna L. Lloyd.