

LA GRANT
IN 61-CR
319017
P31

Optimal Guidance Law Development for an Advanced Launch System

INTERIM PROGRESS REPORT

1 June 1990 – 30 November 1990

December 1990

Research Supported by NASA Langley Research Center

NASA Grant No. NAG-1-939

Principal Investigators: Anthony J. Calise & Dewey H. Hodges

Research Assistants: Martin S. Leung & Robert R. Bless

NASA Grant Monitor: Dr. Daniel D. Moerder

**Georgia Institute of Technology
School of Aerospace Engineering
Atlanta, GA 30332-0150**

(NASA-CR-187652) OPTIMAL GUIDANCE LAW
DEVELOPMENT FOR AN ADVANCED LAUNCH SYSTEM
Interim Progress Report, 1 Jun. - 30 Nov.
1990 (Georgia Inst. of Tech.) 31 pCSCL 09B

N91-15749

Unclas
G3/61 0319017

1. Summary

1.1 Regular Perturbation Analysis

During this reporting period research was directed at evaluating the regular perturbation method described in details in [1]. Closed-loop simulations were performed with a first order correction including all of the atmospheric terms. In addition, a method was developed for independently checking the accuracy of the analysis and the rather extensive programming required to implement the complete first order correction with all of the aerodynamic effects included. This amounted to developing an equivalent Hamiltonian for the first order analysis and evaluating it by quadrature. The result was compared to the Hamiltonian computed from the first order analysis. A second order correction was also completed for the neglected spherical Earth and back-pressure effects. Finally, an analysis was begun on a method for dealing with control inequality constraints.

To date, the results on including higher order corrections do show some improvement for this application, however we do not know at this stage if significant improvement will result when the aerodynamic forces are included. If the result is negative, then our recommendation is that the method of Matched Asymptotic Expansions (MAE) be explored as the next major step in this research effort. The results from a parallel research effort on aeroassisted orbit transfer trajectories indicate that the regular perturbation analysis under current investigation actually plays the role of the inner expansion in a MAE analysis. The outer solution in a MAE analysis provides a correction currently not available from a regular expansion. We would like to explore if a similar situation holds for the dynamics associated with launch vehicle trajectories.

1.2 Finite Element Analysis

The weak formulation for solving optimal control problems has now been extended in order to account for state inequality constraints. The formulation has been tested on three example problems and numerical results have been compared to the exact solutions. Development of a general-purpose computational environment for the solution of a large class of optimal control problems is now well underway. An example, along with the necessary input and the output, is given.

2. Research Accomplishments

2.1 Regular Perturbation Analysis

Closed-loop Simulation

Figures 1 and 2 compare the performance of the closed-loop control solutions generated by two different methods, with the open loop optimal solution generated using a multiple shooting method. A first order correction was made in each case for the neglected spherical Earth & engine back-pressure effects. The simulation results are for lift and drag set to zero. In Method 1, the control update interval was 1 second and within each interval the control was held constant. The control was determined by repeatedly calculating a new zero order solution and performing a quadrature at every update. Method 2 was based on a pre-calculated quadrature for a fixed zero order solution corresponding to the conditions near launch. See [1] for details on these two Methods. Nearly continuous control updating was used for Method 2 because the computational effort is trivial. It amounts to solving a set of 4 linear equations to generate the on-line control. A mid-point extrapolation scheme (accuracy equivalent to a Runge-Kutta 7/8) was used in both methods for the simulation. Table 1 gives a comparison of the terminal conditions and the performance index.

Table 1: Terminal Values Comparison

	Method 1	Method 2	Optimal
h_f	148160m	148147m	148160m
V_f	7857.58m/s	7864.99m/s	7858.2m/s
γ_f	0.001deg	0.035deg	0
t_f	355.612s	355.744s	355.591s

The results show a dramatic improvement in comparison to the open loop solutions reported in [1] for these two methods. In [1] the trajectories were obtained from a single calculation at launch, and the trajectories were constructed by simply summing the zero order solution and integrated first order dynamics. For the results shown here, the solutions were obtained by integration of the complete dynamics, with the control computed from the perturbation analysis.

Figures 3 and 4 show the closed loop simulation results for Method 1 including the aerodynamic forces in the first order correction. Since the zero order solution gives an unrealistically high angle-of-attack (approximately -45 deg.) at launch, the simulation was started at an altitude of 10525 meters, so that the zero order solution for alpha was still within the range of the tabulated aerodynamic data. Figure 3 clearly indicates the onset of an instability in alpha at this altitude. The slight increase in alpha near the end is due to a numerical problem that can be removed at a later date.

It was not known at this point if this instability was due to an analysis and/or programming error, or due to the inability of the regular perturbation analysis to account for aerodynamic effects using a first order correction. It could also be that a second-order correction would not significantly improve matters, since at best we are forming an asymptotic series solution to the problem. Thus we decided to develop an independent check on our results before proceeding to a second order analysis, which is described in the next section

Checking the First-order Analysis

Checking was performed by monitoring a Hamiltonian function which corresponds to the first order necessary conditions when viewed as being derived from an equivalent optimization problem. This Hamiltonian is different from the first order expansion of the original Hamiltonian for the full nonlinear dynamics. The first order Taylor's series expansion of the original Hamiltonian does not correspond to the costate equations and the optimality condition of the first-order dynamics [2]. So a new Hamiltonian (H) was derived which has the following form:

$$\begin{aligned}
 H = & \{f_x^{T0}x_1 + f_u^{T0}u_1 + \frac{T_1}{T_0}[f^0 + (t - t_0)f_t^0] + g^0\}^T \lambda_1 \\
 & + \frac{1}{2} \begin{bmatrix} x_1 & u_1 \end{bmatrix} \begin{bmatrix} \frac{\partial(f_x^T \lambda)_0}{\partial x} & \frac{\partial(f_x^T \lambda)_0}{\partial u} \\ \frac{\partial(f_u^T \lambda)_{T_0}}{\partial u} & \frac{\partial(f_u^T \lambda)_0}{\partial u} \end{bmatrix} \begin{bmatrix} x_1 \\ u_1 \end{bmatrix} \\
 & + \left\{ \frac{T_1}{T_0} [(f_x^T \lambda)^0 + (t - t_0)(f_x^T \lambda)_t^0] + (g_x^T \lambda)_t^0 \right\}^T x_1 + \left\{ \frac{T_1}{T_0} (t - t_0)(f_x^T \lambda)_t^0 + (g_u^T \lambda)_t^0 \right\}^T u_1 \quad (2.1)
 \end{aligned}$$

Since the first order system is time-varying, the Hamiltonian is not constant. The first order analysis is checked by realizing the following two expressions:

$$\frac{dH}{dt} = \frac{\partial H(\mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}^*)}{\partial t} \quad (2.2)$$

$$H = \boldsymbol{\lambda}^T \mathbf{f}(\mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}^*) \quad (2.3)$$

where for the right hand side of (2.2) we mean the partial derivative of the expression in (2.1). The Hamiltonian was computed in two ways. First by numerical integration of (2.2) along the trajectory with an arbitrary initial condition. Second, by direct substitution of the state, costate and control values from the first order solution into (2.3). If the analysis and programming is correct, the *difference* between the two ways of computing H should be stagewise constant. This was verified by the results in Figures 5 and 6. In this setting, both the zero and first-order optimality conditions and their costate equations were verified because H also depends on the zero order solution. The difference in the two calculations is zero to within 4 significant digits.

Second-order Correction

A second order analysis was also carried out to determine if any improvement results in comparison to the first order solution. At this stage, the second order analysis including the aerodynamic forces is not completed. However, a second order correction for the spherical Earth and back-pressure effects was evaluated and the results are depicted in Figures 7-11. These are the open-loop histories obtained by summing the forward integration results for each corrected term. In integrating the second order dynamics, the first order state and costate histories are required. This is done by a forward integration of the first order dynamics using the known initial values for $\mathbf{x}_1(t_0)$ and the calculated initial costate correction $\boldsymbol{\lambda}_1(t_0)$. The histories are stored for a sufficient number of sample points, and retrieved using a piecewise linear interpolation for the integration of the second order dynamics.

In examining the results of Figures 7-11 it should be noted that the pattern throughout is that the first-order correction over-corrects the zeroth-order solution, and that the second-order correction over-corrects the first-order solution. Unfortunately, the error is not significantly decreased by the second-order correction in most of the results, with the exception of the λ_u profile which shows a dramatic improvement. The estimates for the initial values of the costates and the final time (performance index) are compared in Table 2.

Table 2. Performance Comparison Of The Open-loop Results

	$\lambda_v(0)/s^2m^{-1}$	$\lambda_u(0)/s^2m^{-1}$	$\lambda_r(0)/sm^{-1}$	t_f/s
Zeroth-order	0.20156e-1	0.19334e-1	0.54560e-4	360.047
First-order	0.39188e-1	0.22036e-1	0.56468e-3	354.335
Second-order	0.35344e-1	0.20899e-1	0.57143e-3	355.254
Optimal	0.37352e-1	0.20868e-1	0.60304e-3	355.606

Control Inequality Constraint

Preliminary work on addressing control inequality constraints ($C(\mathbf{x}, \mathbf{u}) \leq 0$) is under investigation. This approach makes use of a slack variable (σ) to transform to a strict equality constraint [3]. The necessary conditions are as follows:

$$\dot{\mathbf{x}} = \mathbf{f} + \varepsilon \mathbf{g}; \quad t \in [t_0, t_f] \quad (2.4)$$

$$\dot{\boldsymbol{\lambda}} = -\mathbf{f}_x^T \boldsymbol{\lambda} - C_x \boldsymbol{\mu} - \varepsilon \mathbf{g}_x^T \boldsymbol{\lambda} \quad (2.5)$$

$$0 = \mathbf{f}_u^T \boldsymbol{\lambda} + C_u \boldsymbol{\mu} + \varepsilon \mathbf{g}_u^T \boldsymbol{\lambda} \quad (2.6)$$

$$0 = \alpha \boldsymbol{\mu} \quad (2.7)$$

$$0 = C + \frac{1}{2} \alpha^2 \quad (2.8)$$

Equations (2.4-2.6) are derived from the necessary conditions on the augmented Hamiltonian,

$$H = \mathbf{f}^T \boldsymbol{\lambda} + \varepsilon \mathbf{g}^T \boldsymbol{\lambda} + (C + \frac{1}{2} \alpha^2) \boldsymbol{\mu} \quad (2.9)$$

When the trajectory is on the constraint, $\alpha = 0$. When it is off the constraint, $\boldsymbol{\mu} = 0$. Note that the product is zero at every instant. Alternatively, (2.7) can be derived if we realize that the slack variable can be treated as a control variable and then use the optimality condition $H_\alpha = 0$. Equations (2.7) and (2.8) provides the additional information needed to determine α and $\boldsymbol{\mu}$.

To obtain the zeroth- and higher-order formulations, we simply need to carry out the expansions including:

$$\boldsymbol{\mu} = \boldsymbol{\mu}_0 + \varepsilon \boldsymbol{\mu}_1 + \varepsilon^2 \boldsymbol{\mu}_2 + \dots \quad (2.10)$$

$$\alpha = \alpha_0 + \varepsilon \alpha_1 + \varepsilon^2 \alpha_2 + \dots \quad (2.11)$$

Substituting these expansions and equating like powers in ε , the algebraic equations (2.6-2.8) can be grouped as (note that for simplicity, we consider a scalar u case):

$$\begin{bmatrix} (\mathbf{f}_u^T \boldsymbol{\lambda})_u^0 + C_{uu}^0 \boldsymbol{\mu}_0 & C_u^0 & 0 \\ 0 & \alpha_0 & \boldsymbol{\mu}_0 \\ C_u^0 & 0 & \alpha_0 \end{bmatrix} \begin{bmatrix} u_j \\ \boldsymbol{\mu}_j \\ \alpha_j \end{bmatrix} = - \begin{bmatrix} (\mathbf{f}_u^T \boldsymbol{\lambda})_H^0 + C_{uH}^0 \boldsymbol{\mu}_0 \\ 0 \\ C_H^0 \end{bmatrix} \mathbf{H}_j - \begin{bmatrix} \mathbf{f}_u^{0T} \\ 0 \\ 0 \end{bmatrix} \boldsymbol{\lambda}_j - \begin{bmatrix} \Omega(\mathbf{x}_0, \dots, \mathbf{x}_{j-1}, u_0, \dots, u_{j-1}, \boldsymbol{\lambda}_0, \dots, \boldsymbol{\lambda}_{j-1}) \\ 0 \\ 0 \end{bmatrix} \quad (2.12)$$

where $j = 1, 2, \dots$

Solving the control constraint problem requires a guess of the switching structure. This is true of all indirect methods. In this case, it is the switching structure on the zero order solution that matters. The method requires that the zero order solution captures the true switching structure because it affects the matrix on the left-hand-side of (2.12). It is this matrix which subsequently produces the control correction that leads to a better approximation. If the matrix is singular the method and the expansion technique will fail. One case that does lead to a singular matrix is the touch-point switching structure, where

$$\alpha_0 = \boldsymbol{\mu}_0 = 0$$

$$\det \begin{pmatrix} (\mathbf{f}_u^T \lambda)_u^0 + C_{uu}^0 \mu_0 & C_u^0 & 0 \\ 0 & \alpha_0 & \mu_0 \\ C_u^0 & 0 & \alpha_0 \end{pmatrix} = (\mathbf{f}_u^T \lambda)_u^0 \alpha_0^2 + C_u^{02} \mu_0 = 0 \quad (2.13)$$

For some simple cases, it may be possible to incorporate the control constraint in the zero order problem, thus capturing the true switching structure. However, for the launch vehicle problem, where the dynamics are nonlinear and time varying, and incorporation of any form of control constraint will make the derivation of an analytic solution difficult. Further analysis is required to see whether any simplification is possible.

2.2 Finite Element Analysis

2.2.1 Extension of the Analysis. The method based on the weak Hamiltonian formulation derived in [4] and [5] has now been extended to handle problems with state inequality constraints. An outline of the derivation and a simple example problem are given in Appendix A. (Even more details of the derivation can be found in [6], a copy of which will be sent to the Technical Monitor as soon as it is complete.)

The derivation proceeds in the following manner. It is desired to develop a solution strategy for optimal control problems with state inequality constraints based on finite elements in time. In an attempt to make the solution scheme as general as possible, all strong boundary conditions are transformed into natural boundary conditions. This is done so that the shape functions can be chosen from a less restrictive class of functions, which enables one to choose the same shape functions for every optimal control problem.

The idea of transforming the strong boundary conditions to natural boundary conditions [7] revolves around adjoining a constraint equation to the performance index with an unknown Lagrange multiplier. The variation of the performance index is then taken in a straightforward manner. Through appropriate integration by parts, it is possible to show that the Euler-Lagrange equations are identical to those derived in classical textbooks [8] and that the boundary conditions are the same, only stated weakly instead of strongly.

2.2.2 Development of a General Code. The weak formulation is capable of solving optimal control problems that have continuous states, costates, and controls, and problems with discontinuities arising from staging (i.e., discontinuities in the system equations), control inequality constraints and state inequality constraints. The algebraic equations which come from the weak formulation may be derived prior to specifying the problem to be solved. It is this feature in particular that allows for a general problem-solving environment to be created.

The main goal of the general code is to reliably solve a large class of optimal control problems with a *minimum* of user interaction. Specifically, it is desired to create an environment where the user does not have to write subroutines. To this end, the general code is being developed on a SUN 3/260 workstation and requires a FORTRAN 77 compiler, MACSYMA [9], and the Harwell subroutine library [10]. The general procedure can be broken into three parts that must interface together. The first part is the FORTRAN code. This code contains all the subroutines necessary to solve any of the optimal control problems described above. However, if certain problems require table look-up routines (such as aerodynamic data for a rocket model), then these subroutines must be given by the user and interfaced to the rest of the general code. Thus, there may be a need for some user programming for certain problems. The second part of the general procedure is the use of MACSYMA. The user must supply an input file specifying the problem. This input file is in symbolic form and will be loaded into MACSYMA. MACSYMA will then evaluate all the necessary expressions and automatically generate the FORTRAN code. This code is spliced into a

template file and becomes one of the subroutines. The third and final part of the general procedure will consist of subroutines to generate initial guesses that will reliably converge. Homotopy methods are the prime candidates for this. A very simple type of homotopy method described in [11] is being used. This method converts the algebraic equations to initial-value ordinary differential equations. A second-order Runge-Kutta method is used to integrate the equations and obtain initial guesses for a Newton-Raphson method. This method has worked on all the problems tested to date.

The general code is still being developed at this time. Currently, the code can handle problems with continuous states, costates, and controls, problems with control inequality constraints, and problems with state inequality constraints that only touch the constraint boundary. The general code is now functional (but not complete) for a large class of optimal control problems. An example problem demonstrating the use of the code is given in Appendix B.

3. Future Research

In the perturbation analysis area we plan to complete the second order analysis, and to perform both open loop and closed loop comparisons to the first order results and to the optimal solution. However, we are skeptical at this point that second order correction will remove the instability observed in the first order results when aerodynamic forces are included. Along this line we plan to spend some time investigating the potential that Matched Asymptotic Expansions has for improving the solutions that we have obtained to date. We will also continue investigating the control inequality constraint formulation. Results will first be developed for several simpler problems to evaluate its potential for application to launch vehicle guidance problems.

In the finite element analysis area we plan to complete the development of the general code so as to make it applicable to all types of optimal control problems encountered so far (i.e., up through state inequality constraints). We further plan to document the methodology through the completion of one paper (which we are now revising in response to reviewers) on the application of the method to launch vehicle trajectory analysis, two technical notes on control and state inequality constraints, one paper on the general code, and a user's manual for the code. We have received several calls from parties interested in application of the methodology in industry and, although there is nothing concrete established as yet, hope to somehow transfer the technology to an industry application in the future.

4. References

- [1]. Leung, S. K., Calise A. J., "An Approach To Real-time Guidance Law For An Advanced Launch System," ACC 1990 Proc., San Diego.
- [2]. Lietmann, G., *Optimization Techniques*, Academic Press, 1962.
- [3]. Bensoussan A., *Perturbation Methods In Optimal Control*, John Wiley & Sons, 1982
- [4]. Hodges, Dewey H., and Bless, Robert R., "Weak Hamiltonian Finite Element Method for Optimal Control Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 1, January-February, 1991, to appear.
- [5]. Hodges, Dewey H., Bless, Robert R., Calise, Anthony J., and Leung, Martin, "Finite Element Method for Optimal Guidance of an Advanced Launch Vehicle with Inequality Constraints," *Journal of Guidance, Control, and Dynamics*, submitted for publication, 1990.

- [6]. Bless, Robert R., and Hodges, Dewey H., "Finite Element Solution of Optimal Control Problems with State Inequality Constraints," in preparation.
- [7]. Wu, J. J., and Simkins, T. E., "A Numerical Comparison Between Two Unconstrained Variational Formulations," *Journal of Sound and Vibration*, Vol. 72, 1980, pp. 491 – 505.
- [8]. Bryson, Arthur E. Jr., and Ho, Yu-Chi, *Applied Optimal Control*, Blaisdell Publishing Company, Waltham, Massachusetts, 1969, Chapter 2.
- [9]. MACSYMA Reference Manual, Symbolics, Inc., Burlington, Massachusetts, 1988.
- [10]. Duff, I. S., Harwell Subroutine Library, Computer Science and Systems Division, Harwell Laboratory, Oxfordshire, England, February 1988, Chapt. M.
- [11]. Kane, Thomas R., and Levinson, David A., *Dynamics: Theory and Applications*, McGraw-Hill Book Company, New York, 1985, Chapter 7.
- [12]. Jacobson, D. H., Lele, M. M., and Speyer, J. L., "New Necessary Conditions of Optimality for Control Problems with State-Variable Inequality Constraints," *Journal of Mathematical Analysis and Applications*, Vol. 35, 1971, pp. 255 – 284.

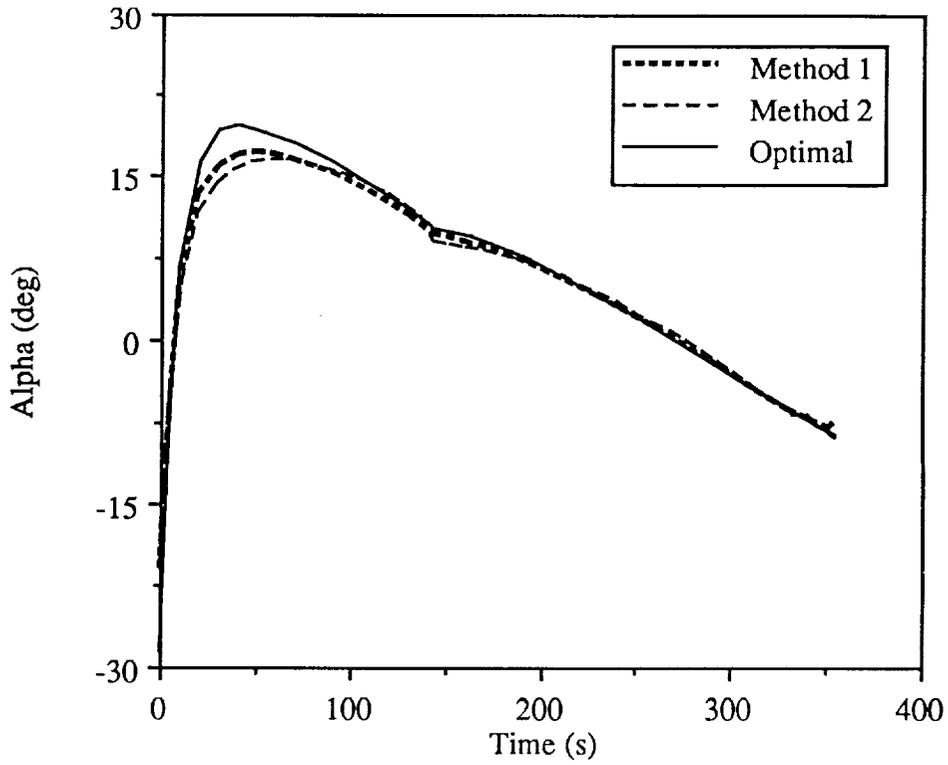


Figure 1: Angle-of-attack Closed-loop Simulation With 1st-order Correction For Spherical Earth & Back-pressure Effects.

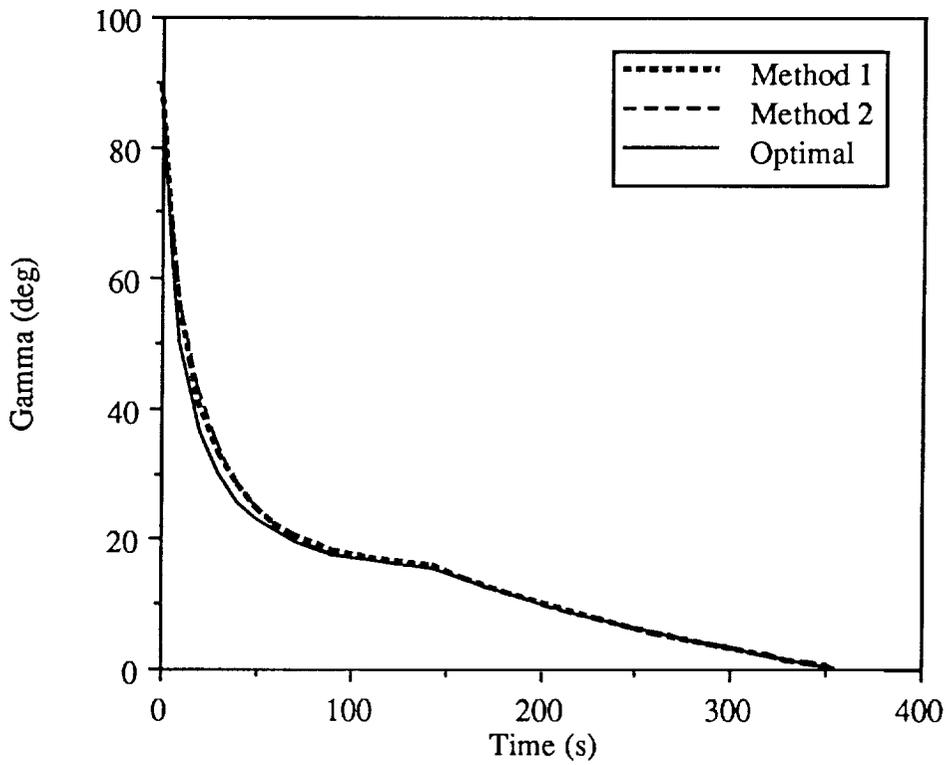


Figure 2: Flight Path Angle Closed-loop Simulation With 1st-order Correction For Spherical Earth & Back-pressure Effects.

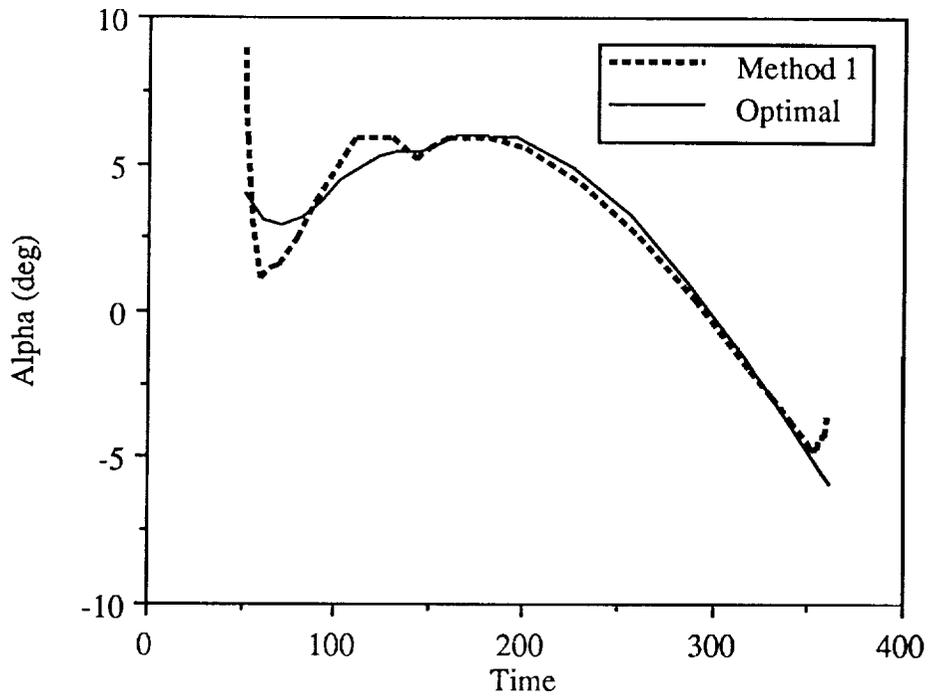


Figure 3. Angle-of-attack Closed-loop Simulation
With 1st-order Correction For Spherical Earth,
Back-pressure & Aerodynamic Effects

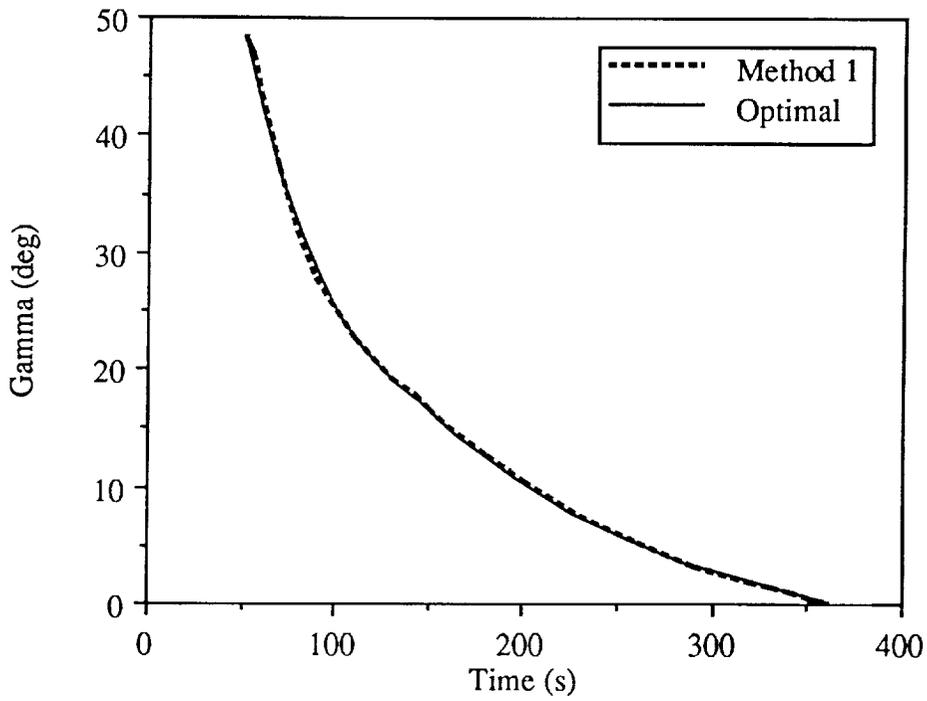


Figure 4. Flight Path Angle Closed-loop Simulation Result

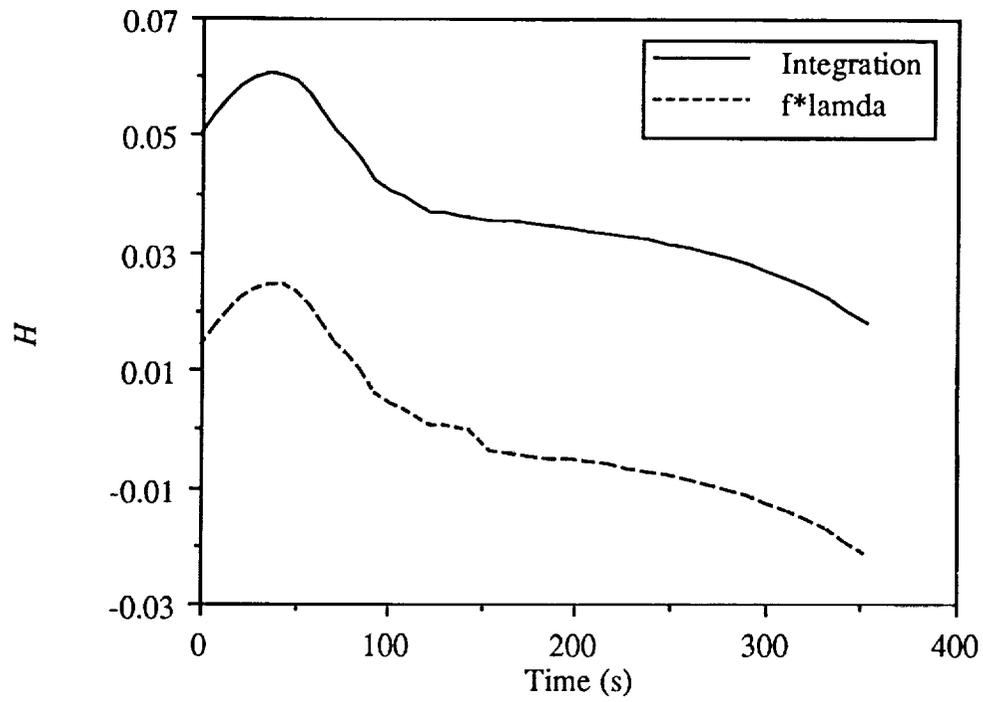


Figure 5. 1st-order Formulation (Spherical Earth & Back-pressure)
Checking Using The Hamiltonian H

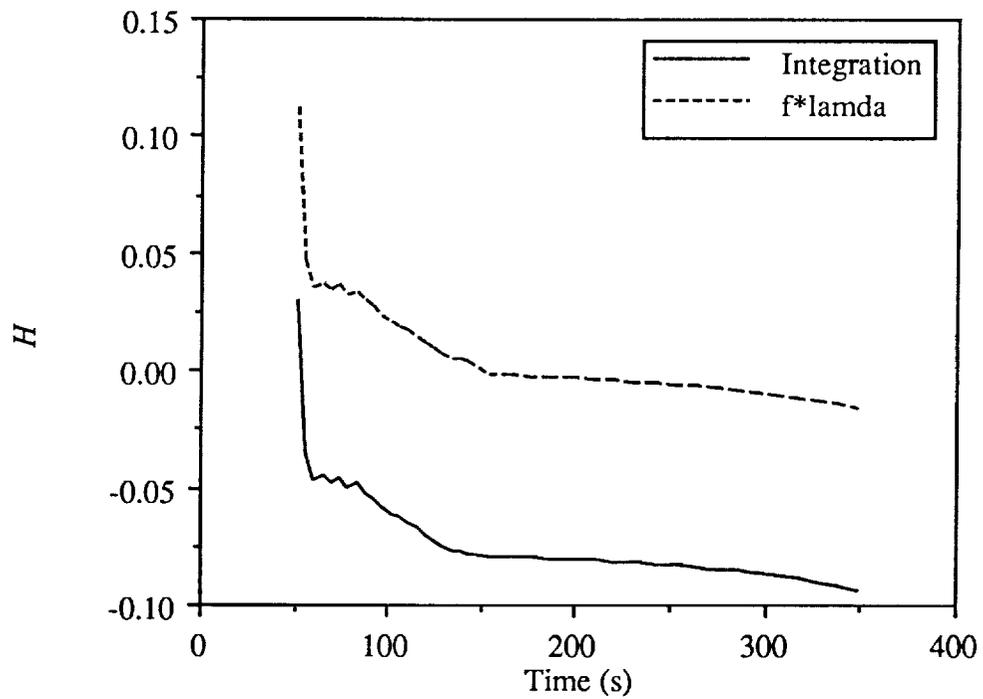


Figure 6. 1st-order Formulation (Spherical Earth, Back-pressure & Aerodynamic Effects) Checking Using H

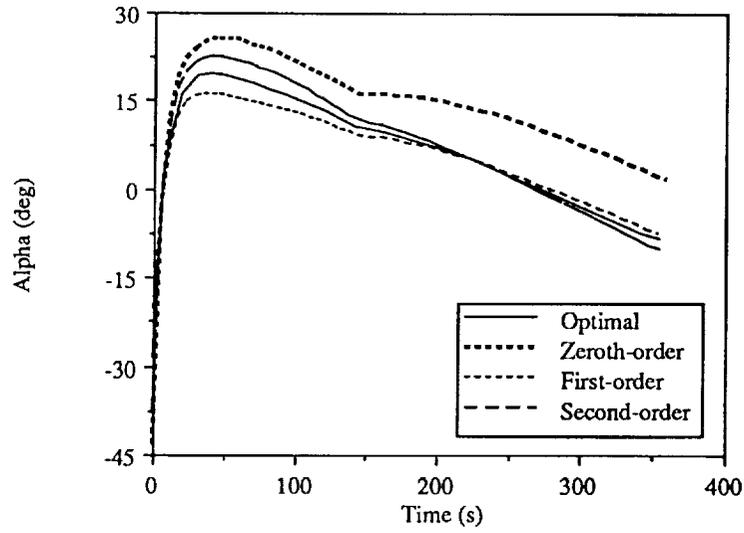


Figure 7. Angle-of-attack Open-loop Solution Comparison For Spherical Earth & Back-pressure Effects

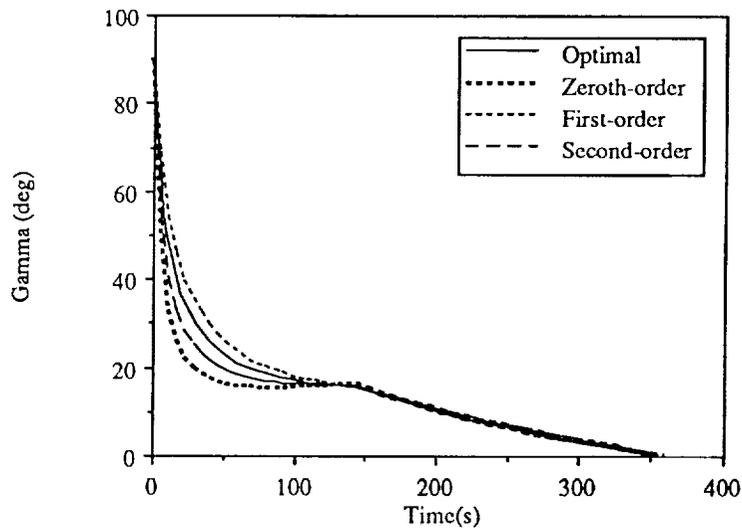


Figure 8. Flight Path Angle Open-loop Solution Comparison

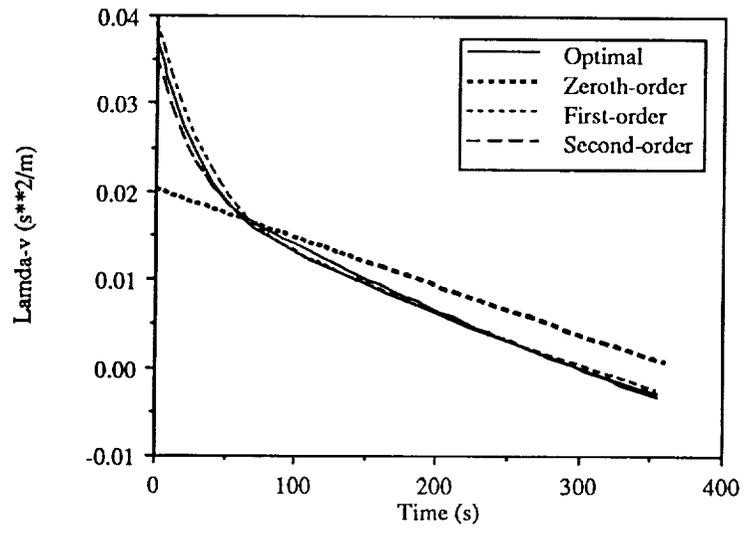


Figure 9. Lamda-v Open-loop Solution Comparison

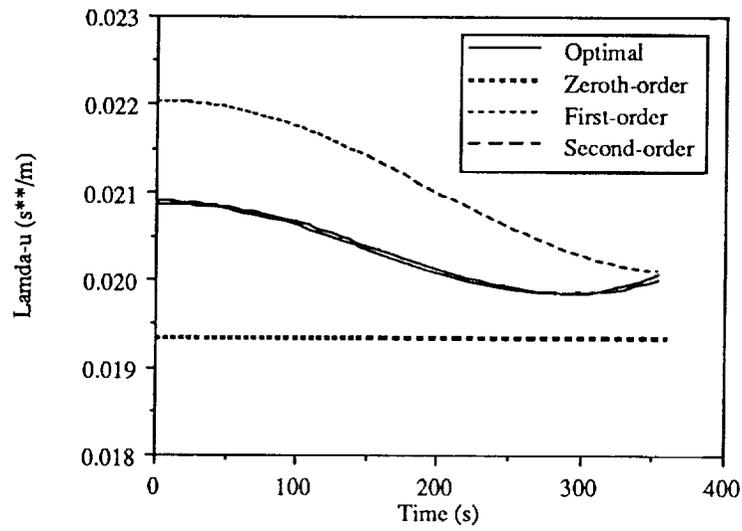


Figure 10. Lamda-u Open-loop Solution Comparison

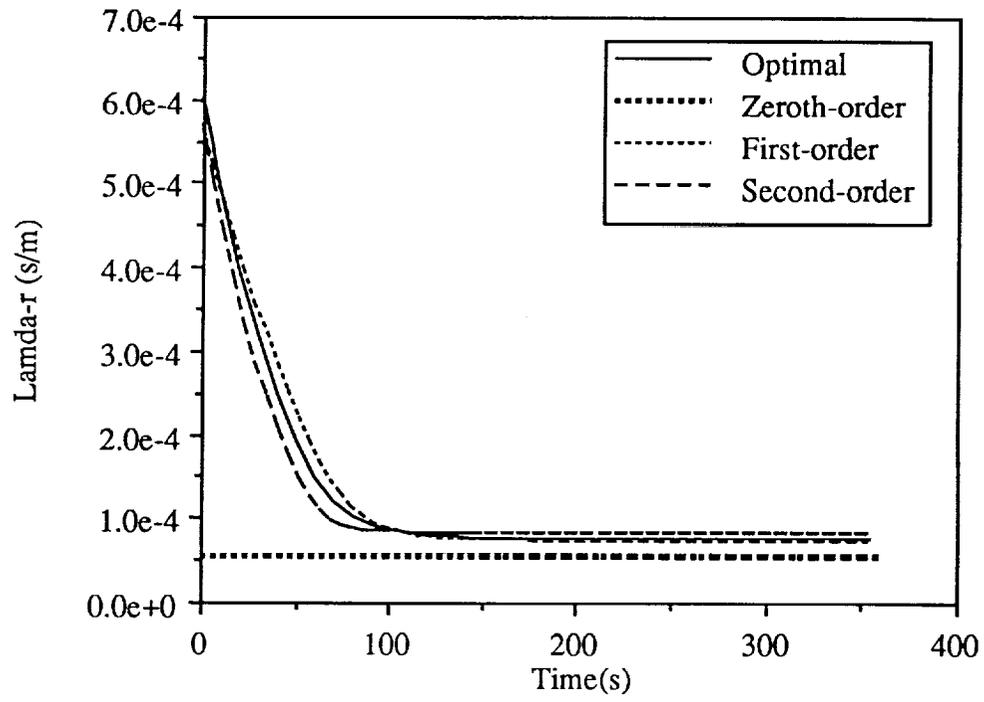


Figure 11. Lamda-r Open-loop solution Comparison

Appendix A: State Inequality Constraints

Consider a system defined by a set of n states x and a set of m controls u . Let the system be governed by a set of state equations of the form $\dot{x} = f(x, u, t)$. The class of problems to be considered is limited to the case where x is continuous, but there may be discontinuities in the costates λ . These discontinuities may be the result of state constraints present in the problem. Elements of the performance index, J_0 , may be denoted by an integrand $L(x, u, t)$ and discrete functions of the states and time $\phi[x(t), t]$ at the initial and final times t_0 and t_f . In addition, any constraints imposed on the states and time at the initial and final times may be placed in sets of functions $\psi[x(t), t]$. These constraints may be adjoined to the performance index by discrete Lagrange multipliers ν defined at t_0 and t_f . Similarly, the state equations may be adjoined to the performance index with a set of Lagrange multiplier functions $\lambda(t)$ which will be referred to as costates.

Now, suppose that there is a scalar constraint on the states and time defined by $S(x, t) \leq 0$. The constraint is said to be of p^{th} -order if the p^{th} total time derivative of S is the first to contain the control u explicitly. The first attempt to solve problems with state inequality constraints was to use the necessary conditions presented in [12]. These necessary conditions lead to successful and accurate solution strategies for states that only touch (*i.e.*, do not ride) the constraint boundary. As is derived in [12], for constraints of odd order greater than one, the solution can at most only touch the constraint boundary. However, for cases where the states ride the constraint boundaries for a nonzero length of time, the algebraic equations developed by the weak form are singular. Private discussions with Jason Speyer and Dan Moerder indicate that the cause is related to a reduced-dimensional manifold; however, we have not been able to develop a nonsingular weak form as of now.

Fortunately, the necessary conditions presented in [8] are accurate for first and second order constraints where the solution often rides the constraint boundary. Thus a weak formulation is also developed using these necessary conditions for constraints where $p = 1$ or $p = 2$. Therefore, below are presented two very similar weak formulations which are accurate for up to a third order constraint and odd-ordered constraints beyond that. Most practical applications will be third-order or less.

General Development

The weak formulation is now derived for touch-point cases. Without loss of generality, assume that there is only one touch-point over the time interval of interest. In this case, the state constraint is nothing more than an interior boundary point which creates a jump in the costate.

The performance index J_0 now takes the form:

$$J_0 = \int_{t_0}^{t_1} [L(x, u, t) + \lambda^T(f - \dot{x})] dt + \int_{t_2}^{t_f} [L(x, u, t) + \lambda^T(f - \dot{x})] dt + \nu_1 S|_{t_1} + \Phi|_{t_0}^{t_f} \quad (1)$$

where $\Phi = \phi[x(t), t] + \nu^T \psi[x(t), t]$. The constraints to be adjoined to J_0 above to transform the strong boundary conditions to weak boundary conditions are that the states be continuous at the initial and final times. Introducing

$$x|_{t_0} \triangleq \lim_{t \rightarrow t_0^+} x(t) \quad \text{and} \quad x|_{t_f} \triangleq \lim_{t \rightarrow t_f^-} x(t) \quad (2)$$

and

$$\hat{x}_0 = \hat{x}|_{t_0} \triangleq x(t_0) \quad \text{and} \quad \hat{x}_f = \hat{x}|_{t_f} \triangleq x(t_f) \quad (3)$$

one can weakly enforce continuity by adjoining $\alpha^T(x - \hat{x})|_{t_0}^{t_f}$ to J_0 where α is a set of discrete unknown Lagrange multipliers defined only at t_0 and t_f . The new performance index is

$$J = J_0 + \alpha^T(x - \hat{x})|_{t_0}^{t_f} \quad (4)$$

To derive the weak principle, it is necessary to take the first variation of J and set it equal to zero. For notational convenience, the following variables are introduced.

$$\hat{\lambda}_0 = \left. \frac{\partial \Phi}{\partial x} \right|_{t_0} \quad \text{and} \quad \hat{\lambda}_f = \left. \frac{\partial \Phi}{\partial x} \right|_{t_f} \quad (5)$$

Also, as is shown in [4], the Lagrange multiplier α can be chosen so that $\delta\alpha = \delta\lambda$. The final form of the weak principle is obtained after integrating by parts so that no derivatives of the states or costates appear. After defining the Hamiltonian $H = L + \lambda^T f$ and denoting the variations of the variables at the initial, touch-point, and final times with subscripts 0, 1, and f respectively, then the resulting equation is

$$\begin{aligned} & \int_{t_0}^{t_1} \left\{ -\delta \dot{x}^T \lambda + \delta \lambda^T f + \delta \dot{\lambda}^T x + \delta x^T \left[\left(\frac{\partial L}{\partial x} \right)^T + \left(\frac{\partial f}{\partial x} \right)^T \lambda \right] \right. \\ & + \delta u^T \left[\left(\frac{\partial L}{\partial u} \right)^T + \left(\frac{\partial f}{\partial u} \right)^T \lambda \right] \left. \right\} dt + \int_{t_1}^{t_f} \left\{ -\delta \dot{x}^T \lambda + \delta \lambda^T f + \delta \dot{\lambda}^T x \right. \\ & + \delta x^T \left[\left(\frac{\partial L}{\partial x} \right)^T + \left(\frac{\partial f}{\partial x} \right)^T \lambda \right] + \delta u^T \left[\left(\frac{\partial L}{\partial u} \right)^T + \left(\frac{\partial f}{\partial u} \right)^T \lambda \right] \left. \right\} dt \quad (6) \\ & + \delta \nu_1^T S|_{t_1} + \delta \nu^T \psi|_{t_f} + \delta x_1^T \left(\frac{\partial S}{\partial x} \right)^T \nu_1 + \delta x_f^T \hat{\lambda}_f - \delta x_0^T \hat{\lambda}_0 - \delta \lambda_f^T \hat{x}_f + \delta \lambda_0^T \hat{x}_0 \\ & + \delta t_1 \left[H(t_1^-) - H(t_1^+) + \nu_1 \frac{\partial S}{\partial t} \right] + \delta t_f H(t_f) = 0 \end{aligned}$$

This is the governing equation for the weak Hamiltonian method for problems with touch-point state inequality constraints. It is easily shown by integrating the $\delta\dot{x}$ and $\delta\dot{\lambda}$ terms by parts in Eq. (6) that all the Euler-Lagrange equations are the same as in [12] and that all boundary conditions are now of the natural type.

One simplification may be made to Eq. (6). If the control is continuous across t_1 (as is often the case), then it is possible to simplify the δt_1 equation since then $f(t_1^-) = f(t_1^+) = f(t_1)$ and $L(t_1^-) = L(t_1^+) = L(t_1)$. From the necessary conditions that are found in [12] or from the ones that could be found from Eq. (6), it is seen that

$$\lambda^T(t_1^-) - \lambda^T(t_1^+) = \nu_1 \frac{\partial S}{\partial x} \quad (7)$$

Now, rewriting the coefficient of δt_1 as

$$\begin{aligned} H(t_1^-) - H(t_1^+) + \nu_1 \frac{\partial S}{\partial t} &= [\lambda^T(t_1^-) - \lambda^T(t_1^+)] f(t_1) + \nu_1 \frac{\partial S}{\partial t} \\ &= \nu_1 \frac{\partial S}{\partial x} \dot{x} + \nu_1 \frac{\partial S}{\partial t} = \nu_1 \frac{dS}{dt} \end{aligned} \quad (8)$$

we see that the condition for continuity of the Hamiltonian reduces to the condition that the first total time derivative of the constraint be zero at t_1 if the control is continuous.

For cases where there is a boundary arc (*i.e.*, the solution rides the constraint boundary for a nonzero length of time), then the weak formulation must be modified. For simplicity and without loss of generality, consider the case where the solution has an unconstrained arc followed by a constrained arc and then another unconstrained arc. Introducing a new Lagrange multiplier function η to adjoin the p^{th} derivative of the constraint S to the performance index, then J_0 becomes

$$\begin{aligned} J_0 &= \int_{t_0}^{t_1} [L(x, u, t) + \lambda^T(f - \dot{x})] dt + \int_{t_1}^{t_2} \left[L(x, u, t) + \lambda^T(f - \dot{x}) + \eta \frac{d^p S}{dt^p} \right] dt \\ &\quad + \int_{t_2}^{t_f} [L(x, u, t) + \lambda^T(f - \dot{x})] dt + \nu_1 N|_{t_1} + \Phi|_{t_0} \end{aligned} \quad (9)$$

where N is a column matrix defined as

$$N^T = \left[S \quad \frac{dS}{dt} \quad \dots \quad \frac{d^{p-1}S}{dt^{p-1}} \right] \quad (10)$$

Again, we define

$$J = J_0 + \alpha^T(x - \hat{x})|_{t_0}^{t_f} \quad (11)$$

Analogous steps to those described above lead to a weak formulation for state constraint problems which ride the constraint boundary. These details are presented in [6].

Example

This example is taken from section 3.11 of [8]. The problem is to minimize

$$J = \frac{1}{2} \int_0^1 u^2 dt \quad (12)$$

The state equations are

$$\begin{aligned} \dot{x}_1 &= u \\ \dot{x}_2 &= x_1 \end{aligned} \quad (13)$$

The state inequality constraint $S(x, t) = x_2 - \theta \leq 0$ is to be imposed. For certain values of θ , the solution only touches the boundary, whereas for other values of θ the solution rides the boundary.

The algebraic equations were solved using a Newton-Raphson method and a FORTRAN code written on a SUN 3/260. The sparse, linearized equations are solved using subroutine MA28 from the Harwell subroutine library [10]. This subroutine takes advantage of sparsity which leads to great computational savings.

The state x_2 is shown in Fig. 12 for the single touch-point case. Results for 2, 4, and 8 elements on either side of the touch-point (denoted by 2:2, etc.) are compared to the exact solution. Note that even the 2:2 element case lies essentially on the exact solution. In Fig. 13, the state x_2 is shown for an example case where the state rides the boundary. Here, there are three time intervals and the number of elements in each interval is denoted by 2:2:2 etc. Again we see that the 2:2:2 case has essentially converged on the exact solution. One drawback of the weak formulation is that two separate codes had to be written to solve this problem. Also, one must determine in advance if the solution will ride or just touch the constraint. However, with the general code described in Appendix B, these are simple and quick things to do.

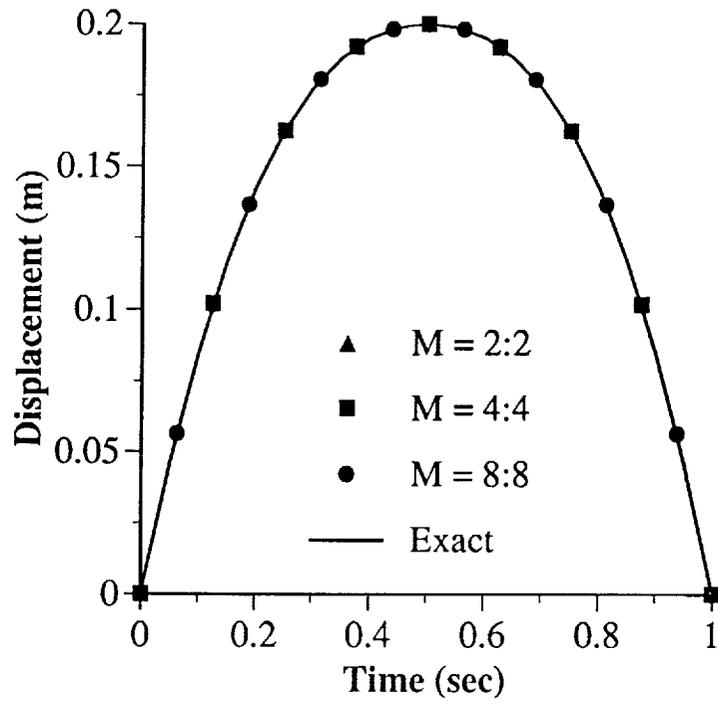


Figure 12: Displacement vs. Time for a touch-point case of $\theta=0.2$

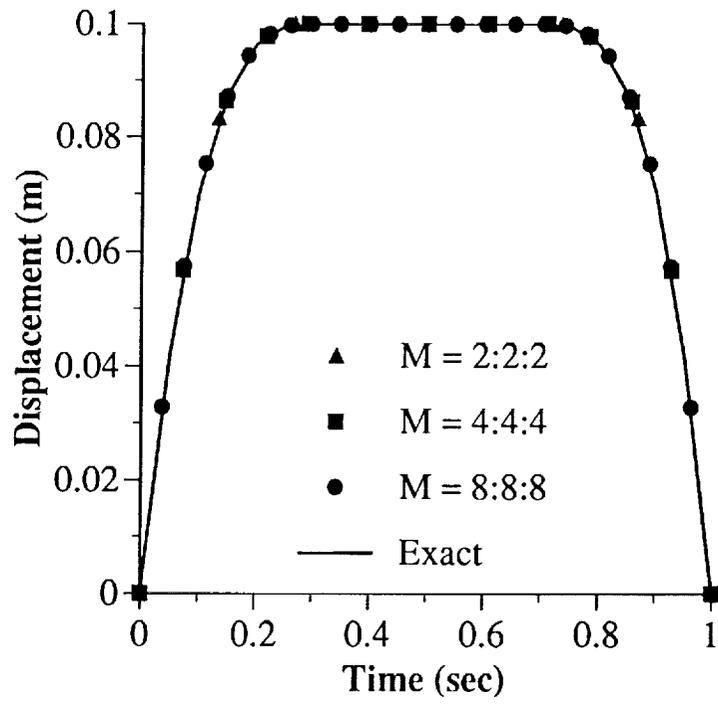


Figure 13: Displacement vs. Time for a boundary arc case of $\theta=0.1$

Appendix B: General Code Usage

As an example of how the general code is used, consider the following model of a single-stage, four-state rocket. The four states are m (mass), h (height), V (velocity), and γ (flight-path angle). The control u will be the angle-of-attack. Letting T_{vac} be the thrust in a vacuum, D be the drag, L be the lift, g be the acceleration due to gravity, I_{sp} be the specific impulse, μ be the earth's gravitational constant, and R_e be the radius of the earth, then the following equations of motion may be used.

$$\begin{aligned}
 \dot{m} &= \frac{-T_{vac}}{gI_{sp}} \\
 \dot{h} &= V \sin \gamma \\
 \dot{V} &= \frac{T - D}{m} - \frac{\mu \sin \gamma}{(R_e + h)^2} \\
 \dot{\gamma} &= \frac{(T + L)u}{mV} + \left(\frac{V}{R_e + h} - \frac{\mu}{(R_e + h)^2 V} \right) \cos \gamma
 \end{aligned} \tag{14}$$

For simplicity in this example, the atmospheric pressure has been neglected and the drag and lift coefficients have been made constants. Note that this is *not* necessary in general. Thus,

$$\begin{aligned}
 T &= T_{vac} = 8155800 \text{ N} \\
 \rho &= 1.225 \exp(-h/6700) \\
 q &= \frac{1}{2} \rho V^2 \\
 D &= qS(C_{D_o} + C_{N_\alpha} u^2) \\
 L &= qS C_{L_\alpha} \\
 S &= 33.468 \text{ m}^2 \\
 C_{D_o} &= 0.02 \\
 C_{N_\alpha} &= 6.0 \\
 C_{L_\alpha} &= 5.98
 \end{aligned} \tag{15}$$

The physical constants used in the above model are $\mu = 3.9906 \times 10^{14} \text{ m}^3 \text{ s}^{-2}$, $R_e = 6378000 \text{ m}$, $g = 9.81 \text{ ms}^{-2}$, and $I_{sp} = 263.4 \text{ s}$. The performance index is the final mass. The known initial conditions are $m(0) = 520000 \text{ kg}$, $h(0) = 1800 \text{ m}$, $V(0) = 300 \text{ m/s}$, and $\gamma(0) = 1.5 \text{ rad}$. The final conditions are $h(t_f) = 50000 \text{ m}$, $V(t_f) = 4000 \text{ m/s}$, and $\gamma(t_f) = 0.0 \text{ rad}$.

Below the input file used to solve this problem is given. The user is required to supply the number of states NS, the number of control constraints NP (zero in this example), the number of controls M, and the number of constraints on the states at the final time Q. The next series of lines from TVAC to F[4] define the system equations as given in Eqs. (14) and (15) above. (The lines from TVAC to DRAG are not required but are used

to simplify the actual expressions for F[1] thru F[4].) After the equations are formed, the user supplies the performance index L and PHI. Then the Q constraints are given in PSI and the initial conditions are given in IC. Next the user supplies the final time TF and a guess at the value of the final time TFGUES. Since the final time is unknown, TF is set to zero and the user gives a guess at the final time. Also, guesses for the states at the midpoint of the trajectory and the final point are given in XGUES. These guesses may be *very* crude and can even be zero for many problems. Since the final value of three of the states were known for this problem, crude guesses were easily and obviously obtained. Finally, the number of elements to be run is given in NE.

Regardless of the value of NE, the code automatically starts with the two element case and uses the continuation method of [11] and the Newton-Raphson method to solve the problem. The code then interpolates the solution to this case and runs a four element case using only the Newton-Raphson method. The code continues in this manner until NE is met. If the Newton-Raphson fails to converge for the four or higher element case (which is rare) then the program will start that case over and try the continuation method to solve the four element case.

The output of the example is given after the input file and consists of the solutions for the states, costates, controls, and Hamiltonian for 2, 4, and 8 elements. At the top of each page is the total elapsed computer time from the start of the program. On the two element case sheets is 15.74 secs. This is the time the code took to run the continuation method and the Newton-Raphson method for this case. This is a rather small number given the complexity of the problem and the fact that an accurate second-order Runge-Kutta method was used to solve the problem. The time at the top of the four element case is 18.84 which tells us that only $18.84 - 15.74 = 3.1$ seconds was required to run the four-element case given the solution to the two element case. Finally, the desired eight element case solution was obtained in a total of 23.46 secs and only 4.62 secs from the four element case. Note that this time includes the extraction of nodal values and the production of the data files. This is a nonnegligible part of the total time.

In summary, a complicated rocket trajectory optimization problem which originally took several weeks to program and solve is now solved in about 10 or 15 minutes. The simple input file is typed in a few minutes and a few minutes are required by MACSYMA to create the FORTRAN subroutines. After that, the program runs in a matter of seconds.

```

NS:4;
NP:0;
M:1;
Q:3;
TVAC:8155800.0;
ISP:263.4;
GRAV:9.81;
MU:3.9906E14;
RE:6378000.0;
H(X):= RE+X(2);
RHOSEA:1.225;
S:33.468;
CNA:6.0;
CAT:0.02;
CLA:5.98;
RHO(X):=RHOSEA*EXP(-X(2)/6700.0);
DP(X):=0.5*RHO(X)*X(3)^2;
LIFT(X):=DP(X)*S*CLA;
DRAG(X,U):=DP(X)*S*(CAT+CNA*U(1)^2);
F[1]:-TVAC/(GRAV*ISP);
F[2]:X(3)*SIN(X(4));
F[3]:(TVAC-DRAG(X,U))/X(1) - MU*SIN(X(4))/H(X)^2;
F[4]:(TVAC+LIFT(X))*U(1)/(X(1)*X(3)) + (X(3)/H(X)-MU/(X(3)*H(X)**2))*COS(X(4));
L:0.0;
PHI:X(1);
PSI[1]:X(2) - 50000.0;
PSI[2]:X(3)-4000.0;
PSI[3]:X(4);
IC[1]:520000.0;
IC[2]:1800.0;
IC[3]:300.0;
IC[4]:1.5;
TF:0.0;
TFGUES:100.0;
XGUES[1,1]:260000.0;
XGUES[1,2]:1000.0;
XGUES[2,1]:25000.0;
XGUES[2,2]:50000.0;
XGUES[3,1]:2000.0;
XGUES[3,2]:4000.0;
XGUES[4,1]:0.5;
XGUES[4,2]:.0;
NE:8;

```

NODAL VALUES FOR THE STATES

NUMBER OF ELEMENTS = 2 TOTAL ELAPSED TIME = 15.74

X1	X2	X3	X4	TIME
0.52000E+06	0.18000E+04	0.30000E+03	0.15000E+01	0.00000E+00
0.30215E+06	0.37957E+05	0.11357E+04	0.13600E+00	0.69021E+02
0.84297E+05	0.50000E+05	0.40000E+04	-.27756E-16	0.13804E+03

NODAL VALUES FOR THE STATES

NUMBER OF ELEMENTS = 4 TOTAL ELAPSED TIME = 18.84

X1	X2	X3	X4	TIME
0.52000E+06	0.18000E+04	0.30000E+03	0.15000E+01	0.00000E+00
0.41393E+06	0.14334E+05	0.57867E+03	0.52797E+00	0.33606E+02
0.30786E+06	0.26293E+05	0.12035E+04	0.29367E+00	0.67212E+02
0.20179E+06	0.40027E+05	0.21967E+04	0.19185E+00	0.10082E+03
0.95716E+05	0.50000E+05	0.40000E+04	-.83267E-16	0.13442E+03

NODAL VALUES FOR THE STATES

NUMBER OF ELEMENTS = 8 TOTAL ELAPSED TIME = 23.46

X1	X2	X3	X4	TIME
0.52000E+06	0.18000E+04	0.30000E+03	0.15000E+01	0.00000E+00
0.46737E+06	0.71167E+04	0.39732E+03	0.80861E+00	0.16674E+02
0.41474E+06	0.12224E+05	0.60461E+03	0.50716E+00	0.33349E+02
0.36211E+06	0.17557E+05	0.88158E+03	0.38264E+00	0.50023E+02
0.30948E+06	0.23577E+05	0.12259E+04	0.31682E+00	0.66697E+02
0.25685E+06	0.30505E+05	0.16554E+04	0.26830E+00	0.83371E+02
0.20422E+06	0.38270E+05	0.22047E+04	0.21910E+00	0.10005E+03
0.15159E+06	0.45939E+05	0.29391E+04	0.14049E+00	0.11672E+03
0.98964E+05	0.50000E+05	0.40000E+04	-.24980E-15	0.13339E+03

ALL VALUES FOR CONTROL AND HAMILTONIAN

NUMBER OF ELEMENTS = 2 TOTAL ELAPSED TIME = 15.74

U1	U2	U3	HAMIL	TIME
-.92485E+00	0.00000E+00	0.00000E+00	-.46274E+03	0.00000E+00
-.27303E+00	0.00000E+00	0.00000E+00	-.82633E+03	0.34510E+02
0.42760E+00	0.00000E+00	0.00000E+00	-.71640E+03	0.69021E+02
0.74199E-01	0.00000E+00	0.00000E+00	-.98340E+03	0.10353E+03
0.33028E-01	0.00000E+00	0.00000E+00	-.10640E-08	0.13804E+03

ALL VALUES FOR CONTROL AND HAMILTONIAN

NUMBER OF ELEMENTS = 4 TOTAL ELAPSED TIME = 18.84

U1	U2	U3	HAMIL	TIME
-.61026E+00	0.00000E+00	0.00000E+00	-.17105E+03	0.00000E+00
-.23120E+00	0.00000E+00	0.00000E+00	-.28890E+03	0.16803E+02
-.12248E-01	0.00000E+00	0.00000E+00	-.23618E+03	0.33606E+02
0.73313E-01	0.00000E+00	0.00000E+00	-.24606E+03	0.50409E+02
0.14683E+00	0.00000E+00	0.00000E+00	-.23657E+03	0.67212E+02
0.91624E-01	0.00000E+00	0.00000E+00	-.26451E+03	0.84015E+02
0.56181E-01	0.00000E+00	0.00000E+00	-.21651E+03	0.10082E+03
-.14828E+00	0.00000E+00	0.00000E+00	-.38816E+03	0.11762E+03
-.32205E+00	0.00000E+00	0.00000E+00	0.24727E-11	0.13442E+03

ALL VALUES FOR CONTROL AND HAMILTONIAN

NUMBER OF ELEMENTS = 8 TOTAL ELAPSED TIME = 23.46

U1	U2	U3	HAMIL	TIME
-.55186E+00	0.00000E+00	0.00000E+00	-.51935E+02	0.00000E+00
-.32779E+00	0.00000E+00	0.00000E+00	-.95367E+02	0.83371E+01
-.16595E+00	0.00000E+00	0.00000E+00	-.62856E+02	0.16674E+02
-.38584E-01	0.00000E+00	0.00000E+00	-.76463E+02	0.25011E+02
0.44739E-01	0.00000E+00	0.00000E+00	-.69769E+02	0.33349E+02
0.79906E-01	0.00000E+00	0.00000E+00	-.71521E+02	0.41686E+02
0.10973E+00	0.00000E+00	0.00000E+00	-.70243E+02	0.50023E+02
0.11215E+00	0.00000E+00	0.00000E+00	-.72219E+02	0.58360E+02
0.12021E+00	0.00000E+00	0.00000E+00	-.69462E+02	0.66697E+02
0.10770E+00	0.00000E+00	0.00000E+00	-.74437E+02	0.75034E+02
0.10186E+00	0.00000E+00	0.00000E+00	-.67925E+02	0.83371E+02
0.67127E-01	0.00000E+00	0.00000E+00	-.79036E+02	0.91709E+02
0.29691E-01	0.00000E+00	0.00000E+00	-.64306E+02	0.10005E+03
-.66484E-01	0.00000E+00	0.00000E+00	-.92023E+02	0.10838E+03
-.18969E+00	0.00000E+00	0.00000E+00	-.52356E+02	0.11672E+03
-.28902E+00	0.00000E+00	0.00000E+00	-.14072E+03	0.12506E+03
-.37039E+00	0.00000E+00	0.00000E+00	0.16485E-10	0.13339E+03

NODAL VALUES FOR THE COSTATES

NUMBER OF ELEMENTS = 2 TOTAL ELAPSED TIME = 15.74

L1	L2	L3	L4	TIME
0.38777E+00	0.23895E+00	0.33772E+02	-.95589E+04	0.00000E+00
0.50266E+00	0.13755E+00	0.33298E+02	0.20419E+04	0.69021E+02
0.10000E+01	0.12191E+00	0.32653E+02	0.10502E+04	0.13804E+03

NODAL VALUES FOR THE COSTATES

NUMBER OF ELEMENTS = 4 TOTAL ELAPSED TIME = 18.84

L1	L2	L3	L4	TIME
0.24671E+00	0.31034E+00	0.41713E+02	-.77904E+04	0.00000E+00
0.30216E+00	0.28438E+00	0.42814E+02	-.22655E+03	0.33606E+02
0.38809E+00	0.26974E+00	0.38756E+02	0.41327E+04	0.67212E+02
0.54729E+00	0.25414E+00	0.36560E+02	0.14102E+04	0.10082E+03
0.10000E+01	0.22100E+00	0.36298E+02	-.11383E+05	0.13442E+03

NODAL VALUES FOR THE COSTATES

NUMBER OF ELEMENTS = 8 TOTAL ELAPSED TIME = 23.46

L1	L2	L3	L4	TIME
0.19901E+00	0.34293E+00	0.42300E+02	-.71441E+04	0.00000E+00
0.22653E+00	0.32947E+00	0.47525E+02	-.28336E+04	0.16674E+02
0.25878E+00	0.32159E+00	0.44891E+02	0.11449E+04	0.33349E+02
0.29819E+00	0.31294E+00	0.42327E+02	0.37755E+04	0.50023E+02
0.34844E+00	0.30244E+00	0.40402E+02	0.47900E+04	0.66697E+02
0.41599E+00	0.29121E+00	0.38903E+02	0.39834E+04	0.83371E+02
0.51405E+00	0.28441E+00	0.37713E+02	0.96373E+03	0.10005E+03
0.67453E+00	0.27907E+00	0.36970E+02	-.49726E+04	0.11672E+03
0.10000E+01	0.22278E+00	0.37315E+02	-.13459E+05	0.13339E+03