$IN-08-CR$
71421
$P-64$

Phase 2

Semi-Annual Report on

NONLINEAR STABILITY AND CONTROL STUDY

OF HIGHLY MANEUVERABLE

HIGH PERFORMANCE AIRCRAFT

R.R. Mohler, Principal Investigator

Oregon State University
Department of Electrical and Computer Engineering
Corvallis, Oregon 97331-3211

(503) 737-3470/3617

Graduate Research Assistants: S. Cho, C. Koo, R. Zakrzewski
Undergraduate Participants (NSF Support): D. Aaberg, J. Young
Visiting Researchers: J. Dory[*], J. Kurek[**], A. Yagen[*]

Phase 2

Semi-Annual Report on

NONLINEAR STABILITY AND CONTROL STUDY

OF HIGHLY MANEUVERABLE

HIGH PERFORMANCE AIRCRAFT

Date: February 14, 1992

R.R. Mohler, Principal Investigator

Oregon State University
Department of Electrical and Computer Engineering
Corvallis, Oregon 97331-3211

(503) 737-3470/3617

Graduate Research Assistants: S. Cho, C. Koo, R. Zakrzewski
Undergraduate Participants (NSF Support): D. Aaberg, J. Young
Visiting Researchers: J. Dory[*], J. Kurek[**], A. Yagen[*]

TABLE OF CONTENTS

APPENDICES

A.   List of Project Publications

B.   On Nonlinear Model Algorithm Controller Design

C.   Analysis of Nonlinear Stability Using Robust Stability Analysis for Linear Systems

1. OVERVIEW

This research should lead to the development of new nonlinear methodologies for the adaptive control and stability analysis of high angle-of-attack aircraft such as the F18 (HARV). The present progress report reviews project research over the first half of the second year.

The emphasis has been on nonlinear adaptive control, but associated model development, system identification, stability analysis and simulation is performed in some detail as well. Table 1 summarizes various models under investigation for different purposes.

Models and simulations for the longitudinal dynamics have been developed for all types except 6 in Table 1. A very preliminary analysis has been made on type 6 (neural net models) for adaptive control thus far. It has been shown that dynamic accuracy roughly increases with ascending order of model type from 1 to 7, except that perhaps 3 (Volterra series) and 6 (neural nets) should be interchanged. However, such comparisons depend on how the models are utilized. Here, the focus is on adaptive control, generated by model-reference types 1 to 6, of a complex nonlinear aircraft motion represented by 7 (nonlinear ordinary differential equations). Preliminary analyses use a nonlinear second-order approximation [1] which we found useful for changes in angle of attack (α) by about 10°. A fifth-order nonlinear longitudinal model with the traditional stability derivatives generated as functions primarily of α, for a given altitude and mach number, successfully mimicked F18 flight trajectories [2], and is being utilized for our nonlinear adaptive-control studies at the present time. These models are discussed in the project's first annual report [3].

Briefly, studies completed indicate that nonlinear adaptive control can outperform linear adaptive control for rapid maneuvers with large changes in α. Figures 1 and 2 compare the transient responses where the desired α varies from 5° to 60° to 30° and back to 5° all in about 16 sec. Here, the horizontal stabilator is the only control used with an assumed first-order linear actuator with a 1/30 sec time constant. Unfortunately, an additional rate constraint significantly reduces the system performance for both the nonlinear and linear adaptive control as shown in Figures 3 to 5 and analyzed in the next

Table 1. Aircraft Models

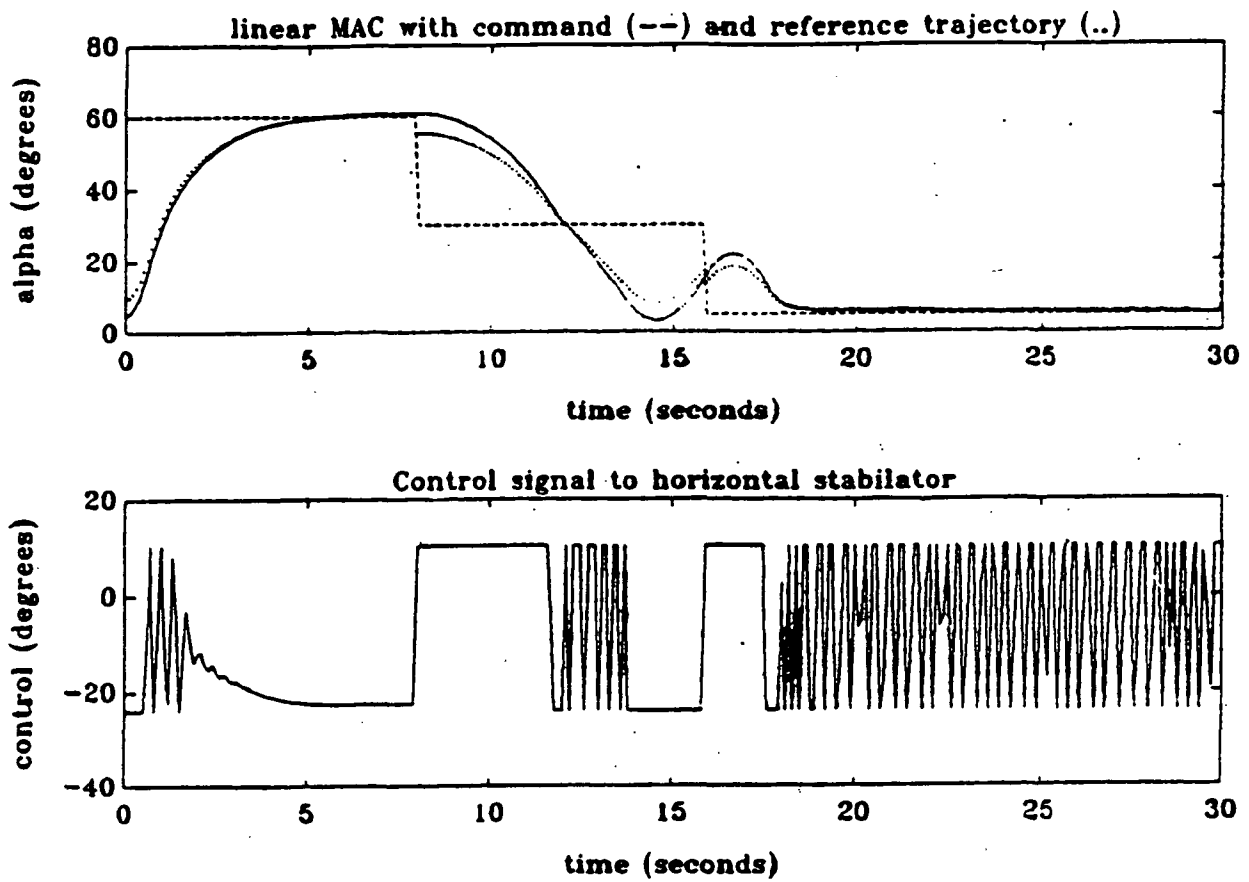|	Type	Purpose	Remarks/Limitations
1.	Linear perturbations at $\alpha = 5°$, $15°$, $35°$, $60°$	Local control, check of nonlinear system, application of well developed linear control methodologies Local stability	Only valid for small maneuvers Special case of types 2-5
2.	Gain scheduled (nonlinear function of α) from 1	Gain-scheduled adaptive control based on well developed methodologies Simplified description of complex system Approximate stability	May have stability problems with small number of reference states and/or large fast maneuvers
3.	Volterra series a) at reference states b) general case	Nonlinear adaptive control via cross-correlation and/or á priori dynamic structure stability approximation Simplified dynamic description of complex system	Non-orthogonal series approximation Sufficiency of 2 or 3 kernels Large computation time for adaptation
4. 5.	Bilinear system a) continuous b) BARMA Polynomial time series	Nonlinear adaptive control via model reference identification (NLMRAC) Stability approximation Simplified dynamic description	Large computation time Bilinearizing controllers may be more practical than linearizing ones Polynomial approximation may be more accurate but more time consuming than linear or bilinear approximation
6.	Neural network	Potential application to adaptive control	Probably less accurate than 4 or 5 for a given data set but accuracy may be more robust outside the available data set
7.	Nonlinear ordinary differential model	Accurate approximation to fast large maneuvers for "final" design and simulation Stability	Neglects flexible modes and other complications

linear MAC with command (– –) and reference trajectory (..)

Control signal to horizontal stabilator

Figure 1. Response for Linear MAC

response using nonlinear MAC (with command and reference curves)
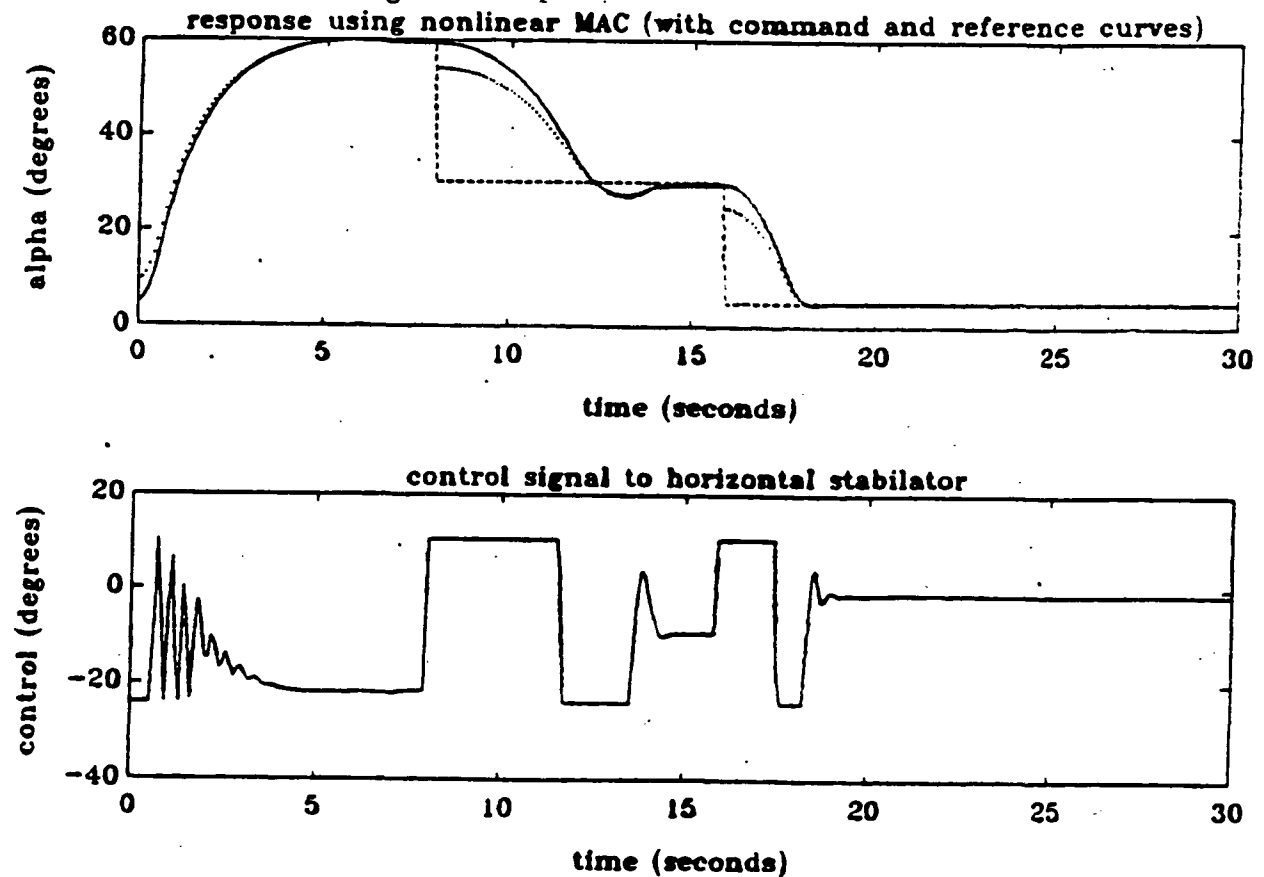

control signal to horizontal stabilator

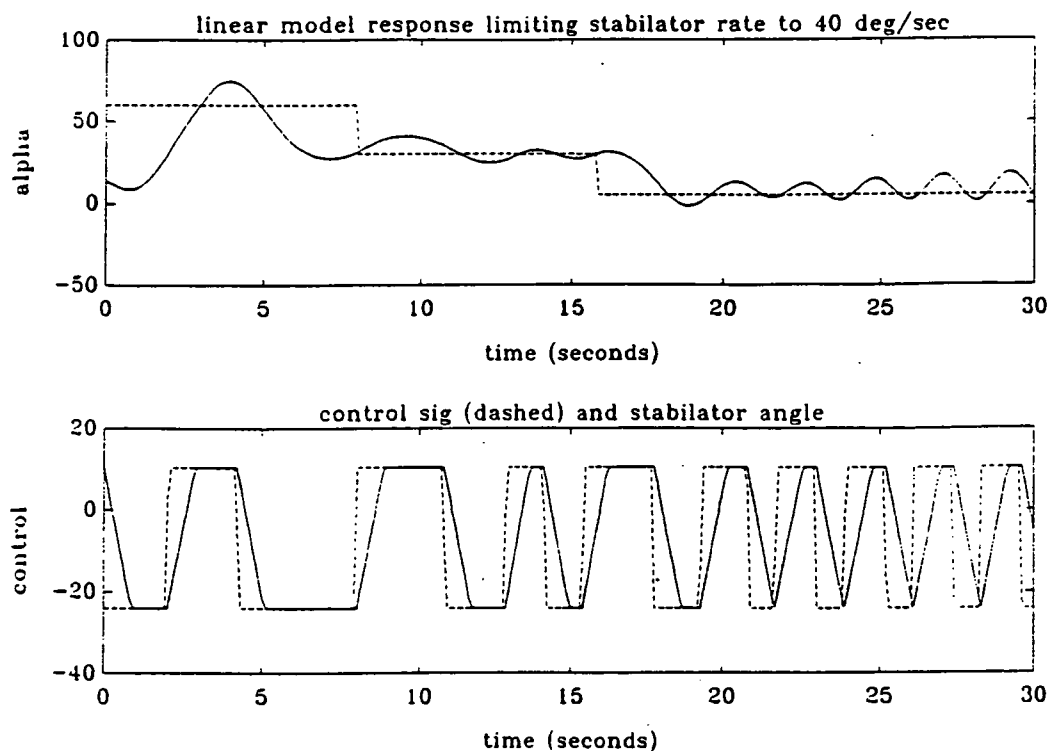Figure 2. Response for Nonlinear MAC

3

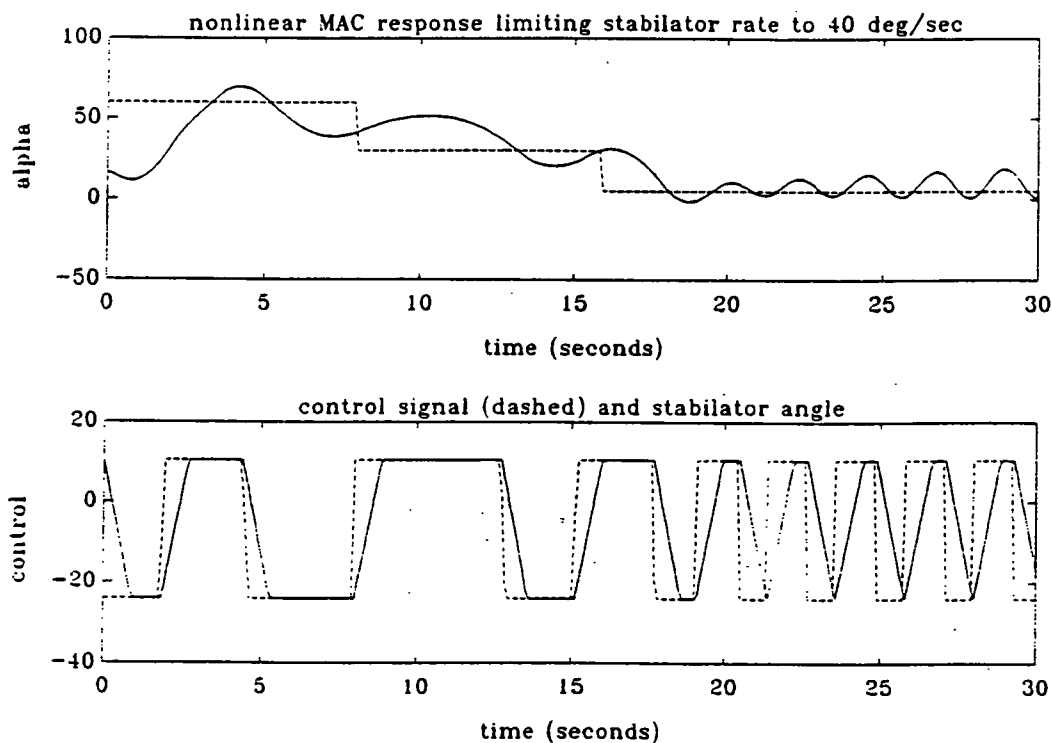Figure 3. Response of Linear MAC with Stabilator Limit of 40° per Second

Figure 4. Response of Nonlinear MAC with Stabilator Limit of 40° per Second
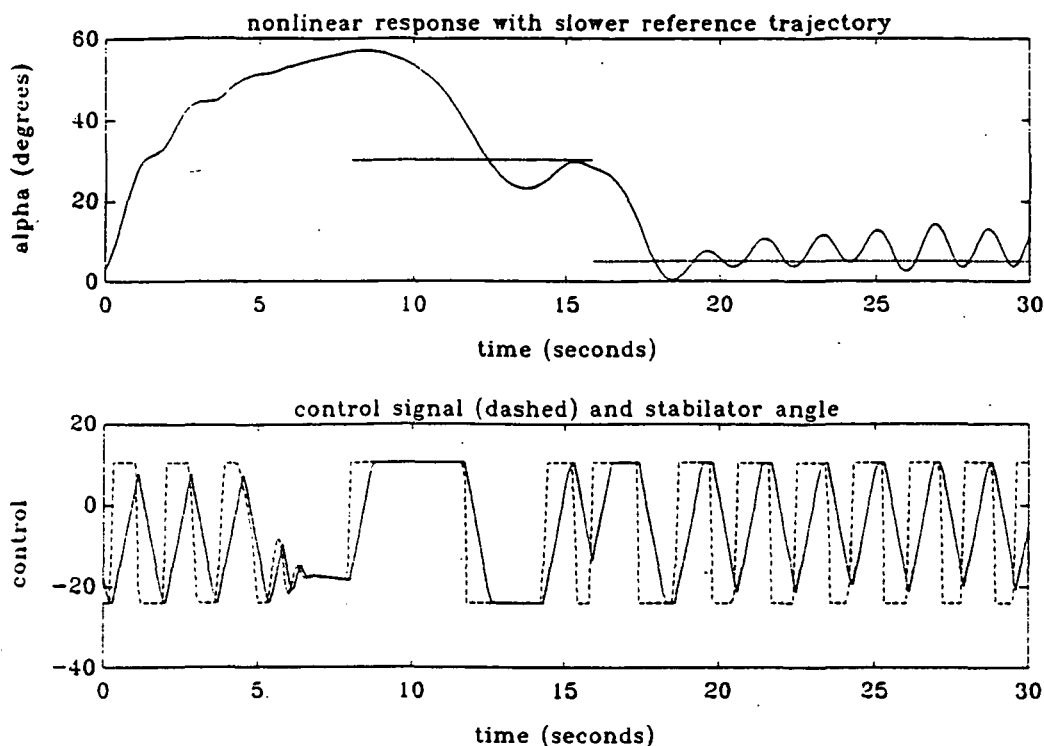
4

Figure 5. Response for Nonlinear MAC with Stabilator Limit of 40° per Second and Slower Reference Trajectory

section. Such lack of controllability can be improved, of course, by introducing thrust vectoring as used in [2]. Appropriate thrust-vector control to supplement the traditional pitch-motion stabilators is underway for the nonlinear adaptive controllers, and preliminary results are encouraging.

A preliminary analysis of time-optimal control of α is studied in Section 3. Here, a new algorithm is derived from the switching-time variational method [4,5] and then applied successfully to the simplified second-order nonlinear model [1]. The method is presently being adapted to the more complex nonlinear fifth-order model. This study should provide a "yard stick" by which to evaluate controller performance as well as provide a base for more effective controller designs. As a byproduct of this analysis the complicated Jacobian of the longitudinal dynamics will be computed as a function of α and other variables. While it is used here to compute bang-bang controller switching times, it may have other uses

5

for approximate dynamic-system identification beyond the usual time-invariant linearized models at trim states.

Finally, linear stability arguments are developed in Appendix C which tend to at least an approximation of the admissible range of model parameters as applied to the nonlinear second-order approximation [1].

2. NONLINEAR MAC ALGORITHM STATUS

Model algorithmic control (MAC), described in [3], starts with

$$\alpha_{ref}(k+1) = \alpha_{mod}(k+1) + \left(\alpha(k) - \alpha_{mod}(k)\right) \tag{1}$$

where

$$\alpha_{mod}(k+1) = p_a^T \phi(k) \tag{2}$$

$$\phi(k) = \left[\alpha, \alpha^2, \alpha^3, q, q\alpha, q a^2, q\alpha^3, u, u\alpha, u\alpha^2, u\alpha^3, 1\right]^T(k) \tag{3}$$

As the control at the moment k must be already computed at moment k the values of $\alpha(k)$ and $q(k)$ are not available for its computation so their estimates must be used instead. The correction term is taken to be the prediction error from the moment k-1 and the equation becomes

$$\alpha_{ref}(k+1) = \hat{\alpha}_{mod}(k+1) + \left(\alpha(k-1) - \alpha_{mod}(k-1)\right) \tag{4}$$

with

$$\hat{\alpha}_{mod}(k+1) = p_\alpha^T \hat{\phi}(k)$$

$$\hat{\phi}(k) = \left[\hat{\alpha}, \hat{\alpha}^2, \hat{\alpha}^3, q, q\hat{\alpha}, q\hat{\alpha}^2, q\hat{\alpha}^3, u, u\hat{\alpha}, u\hat{\alpha}^2, u\hat{\alpha}^3, 1\right]^T(k)$$

$$\hat{\alpha}(k) = p_\alpha^T \phi(k-1) + \left(\alpha(k-1) - \alpha_{mod}(k-1)\right)$$

$$\hat{q}(k) = p_q^T \phi(k-1) + \left(q(k-1) - q_{mod}(k-1)\right)$$

6

The controller is assumed to know the values of angle of attack and of pitch rate at the moment k-1. Then it estimates their current values $\alpha(k)$ and $q(k)$ taking into consideration previous prediction errors, and based on them it calculates the control required to achieve α_{ref} at the moment k+1. The value of control is found as:

$$u(k) = \frac{\tilde{\alpha}_r - p_{1\alpha}\hat{\alpha} - p_{2\alpha}\hat{\alpha}^2 - p_{3\alpha}\hat{\alpha}^3 - p_{4\alpha}\hat{q} - p_{5\alpha}\hat{q}\hat{\alpha} - p_{6\alpha}\hat{q}\hat{\alpha}^2 - p_{7\alpha}\hat{q}\hat{\alpha}^3 - p_{12\alpha}}{p_{8\alpha} + p_{9\alpha}\hat{\alpha} \, p_{10\alpha}\hat{\alpha}^2 + p_{11\alpha}\hat{\alpha}^3} \qquad (5)$$

where

$$\tilde{\alpha}_r = \alpha_{ref}(k+1) - \left(\alpha(k-1) - \alpha_{mod}(k-1)\right) \qquad (6)$$

and $\hat{\alpha} = \hat{\alpha}(k)$, $\hat{q} = \hat{q}(k)$ as described above.

This algorithm was made to be adaptive, or self-tuning, by incorporating on-line identification of the parameters. A recursive least squares (RLS) algorithm was implemented in the form taken from

$$p(k) = \frac{Q(k-2)\,\phi(k-1)}{\lambda(k-1) + \phi(k-1)^T Q(k-2)\,\phi(k-1)} \qquad (7)$$

$$Q(k-1) = \frac{1}{\lambda(k-1)} \left(Q(k-2) - \frac{Q(k-2)\,\phi(k-1)\,\phi(k-1)^T Q(k-2)}{\lambda(k-1) + \phi(k-1)^T Q(k-2)\,\phi(k-1)}\right) \qquad (8)$$

$$e(k-1) = y(k) - p^T \phi(k-1) \qquad (9)$$

where y may denote α or q and p may stand for p_α or p_q, respectively. The forgetting factor λ was introduced to enable the algorithm to change the estimates of parameters with the change of operating conditions. To avoid the unlimited growth of covariance matrix Q at the steady state when the input is not persistently exciting, the variable forgetting factor policy was implemented:

$$\lambda(k) = 1 - \epsilon \frac{e(k)^2}{\bar{e}(k)^2} \qquad (10)$$

where e(k) is the current prediction error, $\bar{e}(k)$ is the average prediction error from last 10 samples, and ϵ is equal to 0.01. As an additional precaution, the trace of the covariance matrix Q was monitored and Q was reset to diagonal matrix whenever the threshold value was exceeded.

To further damp the response, the controller is designed to minimize the one step ahead cost function:

$$J = (y_{mod}(k+1) - y_r(k_1))^2 + \rho(u(k) - u(k-1))^2 \tag{11}$$

with y_{mod}, y_r as before. Minimization of (11) with respect to u(k) yields

$$u(k) = \frac{(\bar{y}_r - a)b + \rho u(k-1)}{b^2 + \rho} \tag{12}$$

where

$$a = p_{1\alpha}\alpha + p_{2\alpha}\alpha^2 + p_{3\alpha}\alpha^3 + p_{4\alpha}q + p_{5\alpha}q\alpha + p_{6\alpha}q\alpha^2 + p_{7\alpha}q\alpha^3 + p_{12\alpha}$$

$$b = p_{8\alpha} + p_{9\alpha}\alpha + p_{10\alpha}\alpha^2 + p_{11\alpha}\alpha^3$$

Obviously, for $\rho = 0$ (12) reduces to (5) while for $\rho = \infty$ we have u(k) = u(k-1) = const.

This controller is used in Figures 2, 4, and 5 with only the linear portion of a,b used in Figures 1 and 3. The algorithm will be generalized to include thrust vector control and variable-horizon cost.

3. TIME-OPTIMAL CONTROL

3.1 Introduction

Various control strategies have been developed by the team and to find their merits it seems useful to have an idea of what are the best output and state trajectories theoretically possible, given the existing constraints on the control variables. For substantially nonlinear systems the problem of synthesis of the optimal feedback control law is usually untractable. On the other hand, there exist numerical techniques that allow us to calculate "open loop controls" - i.e., the specific control signals necessary to achieve the minimum performance index. Aware of the difficulties connected with the controller synthesis problem

we do not seek its exact solution; at this time, we merely want to find the limit for the performance of a controller assuming perfect knowledge of plant dynamics and absence of any unforeseen disturbances.

This report is concerned with the problem of time-optimal control in which we are interested in transferring the system's state from an initial value to some prescribed terminal set in minimal time. In the aircraft problem this might mean changing the flight's pitch angle, path angle, or angle of attack from an initial equilibrium value to some other terminal value, preferably also with all other states moving to the equilibrium. The control value (stabilator or elevator angle) is naturally bounded from below and from above. For some systems it turns out that in case of such simple cube-type constraints on control variables, the time-optimal control is of bang-bang type. However, for quite a large class of systems that are affine in control, we may approximate any measurable control signal with a bang-bang signal with arbitrary accuracy in the sense that corresponding state trajectories are arbitrary close to each other in L^1 metric. Hence, also time-optimal control, if it exists, may be approximated by a bang-bang control, even if it contains singular arcs. Therefore, the approach presented here is to find the bang-bang control that will minimize the transition time. The computational algorithm used here is the switching-time-variation method developed in [4,5]. Since the algorithm gives as an output a control signal with finite number of switchings, it is tacitly assumed that with large enough finite number of switchings, we are able to achieve good enough approximation of optimal control. This, unfortunately, does not follow from the theory I am aware of, since the above mentioned approximation result holds only for bang-bang signals with possibly infinite or even uncountable number of switchings. This delicate question is left aside for the time being to be clarified later. Another point worth indicating here is that resulting control, in an attempt to approximate a continuous "singular" control, may have inter-switching times very small, thus precluding any practicality of the approximation. This, however, is of no concern to us since, as mentioned before, we are interested only in finding the best possible output, or state, trajectories - not the actual control signals corresponding to them at this time.

9

A computer program has been developed for numerical solution of the problem. The program, due to its modular construction, easily allows various plant models to be plugged into it. The switching-time-variation method is used in it for fixed terminal time with the quality function being the weighted distance of the target set. Then the smallest such time is found that allows it to hit the target exactly, and finally the optimal number of switchings is iteratively found that gives minimal transition time.

In what follows the switching-time-variation method is briefly characterized in Section 3.2. Section 3.3 discusses briefly the approximation theorem for bang-bang controls in systems affine in control. The computer implementation of the algorithm is discussed in Section 3.4. Section 3.5 contains the test results of the program for a second-order model of longitudinal dynamics of an aircraft. The concluding remarks discuss the possibilities of application of the computer package to solutions of more complex and problems more close to reality.

3.2 Switching-Time-Variation Method

The switching-time-variation method used here was taken from [4], and the original thesis [5] was also consulted for the details. The method is designed for the computation of optimal control in the class of bang-bang control signals with finite number of switchings. The quality criterion is assumed to be

$$J = \int_{t_0}^{t_f} \left(f_0(x) + g_0(x) u(t) \right) dt \tag{13}$$

for the system of the form

$$\frac{dx}{dt} = f(x) + g(x) u(t) \tag{14}$$

where $x \in \mathbf{R}^n$, $u \in \mathbf{R}^1$, $t \in [t_0, t_f]$. To ensure the existence and uniqueness of solutions of (14), f and g are assumed to be continuously differentiable with respect to x. The control values are constrained by

$$-1 \leq u(t) \leq 1 \tag{15}$$

10

Of course, any control constraints of the cube-like type $u_{min} \leq u(t) \leq u_{max}$ may be transformed to form (15) for system affine in control. The control objective is to minimize the quality criterion (13) for given initial state x_0 with possible penalty term connected with final state already included in f_0 and g_0 by standard transformations, assuming that admissible controls are bang-bang with finite number of switchings. The version of the algorithm described in [4] was developed for systems with scalar controls and the computer program described here is also designed for this special case. However, it is not of any particular difficulty to generalize the algorithm to the case of $u \in \mathbf{R}^m$. If the need arises, the computer program may also be modified to accommodate this possibility. Here the scalar version will be presented because of its notational simplicity.

The method is an iterative one - in each step the gradient of the quality criterion with respect to switching times is computed. The switching vector is defined as:

$$\tau = (\tau_1, ..., \tau_N) \tag{16}$$

where N is the number of switchings, with constraints:

$$t_0 \leq \tau_1 \leq ... \leq \tau_N \leq t_f \tag{17}$$

The control value on the interval $[\tau_i, \tau_{i+1})$ is then equal $(-1)^i$. The augmented system is defined as:

$$\frac{d\tilde{x}}{dt} = \tilde{f}(\tilde{x}) + \tilde{g}(\tilde{x}) u(t) \tag{18}$$

where $\tilde{f}^T = (f_0, f^T)$, $\tilde{t}^T = (g_0, g^T)$, $\tilde{x}_0^T = (0, x_0^T)$, and the adjoint system equation is

$$\frac{d\lambda}{dt} = -\left(\frac{\partial \tilde{f}}{\partial \tilde{x}}(t) + \frac{\partial \tilde{g}}{\partial \tilde{x}}(t) u(t) \right)^T \lambda \tag{19}$$

with terminal conditions $\lambda_i(t_f) = (\partial J/\partial \tilde{x}_i)(t_f)$. Then the gradient of the quality criterion with respect to the switching vector may be calculated by means of the formula

$$\frac{\partial J}{\partial \tau_i} = (-1)^{i-1} \phi(\tau_i) \qquad (20)$$

with function ϕ defined by

$$\phi(t) = 2\langle \bar{g}(\bar{x}(t)), \lambda \rangle \qquad (21)$$

with gradient calculated the method consists of iterative descent steps

$$\tau_i(k+1) = \tau_i(k) + k_i \frac{\partial J}{\partial \tau_i} \qquad (22)$$

where k_i are such that constraints (17) are satisfied and sufficiently small to ensure that $J(k+1) < J(k)$. The algorithm is terminated if either the gradient is zero or no feasible (i.e., descent) step may be executed.

On top of the algorithm of finding the optimal switchings with their number given there is an outer loop modifying this number. If the optimal control results in a constraints $\tau_i \le \tau_{i+1}$ active than the switchings i and i+1 should be removed. On the other hand, if there are two zeros of $\phi(t)$ not coinciding with any of the switching times than two switchings should be added between these zeros. After the modifications of the dimensionality of the switching vector the inner loop of optimization is again performed and the process is terminated when no more changes of the number of switches are necessary.

It is worth noticing that the above algorithm of finding the optimal bang-bang control may be also generalized for broader class of systems dx/dt = f(x(t),u(t)). The main difference would be the formula for function $\phi(t0)$ which would become

$$\phi(t) = \langle \left(f\big(x(t), u_{max}\big) - f\big(x(t), u_{min}\big) \right), \lambda \rangle \qquad (23)$$

Of course, the technical assumptions ensuring the existence of solutions should be satisfied.

3.3 Approximation for Systems Affine in Control

The algorithm described above calculates the optimal control within the class of bang-bang control. However, for systems affine in control a result is available stating that we may approximate an arbitrary admissible control with a bang-bang control such that corresponding trajectories are arbitrary close.

The theorem, stated and proven in [6], assumes that we have a system of the form (14) with constraints (15). Functions f and g are continuously differentiable, and a Lipshitz type condition $< f(x) + g(x)u, x > \leq K(1 + \| x \|^2)$ preventing finite escape-time is also assumed to be satisfied for all x in the region of interest. Then an arbitrary measurable control signal u(t), $t \in [t_0, t_f]$ satisfying (15) is considered with corresponding state trajectory x(t). Then the theorem states that given any $\epsilon > 0$ is always possible to find a bang-bang control $u^*(t)$ satisfying $|u^*(t)| \leq 1$, such that the corresponding state trajectory $x^*(t)$ approximates x(t) uniformly on $[t_0, t_f]$ with accuracy less than ϵ, i.e., $|x(t) - x^*(t)| \leq \epsilon$ for all $t \in [t_0, t_f]$.

Although the theorem stated above considers a bang-bang control with not necessarily finite or even countable number of switchings, it gives some justification to using the switching-time-variation method for systems with singular optimal controls. Intuitively for reasonably smooth systems there should be some kind of continuity enabling in turn approximating the bang-bang control u^* with a sequence of bang-bang signals u with finite number of switchings. However, I am not aware of any such result, and in monograph [7] from 1990 the aforementioned result is cited after [6] as the only available. It still seems feasible to come up with some, maybe more restrictive, assumptions which would justify using finite number of switchings.

3.4 Computer Implementation

The algorithm discussed in Section 3.2 was implemented as a quite general software package. It finds the time-optimal control for the case when the terminal set is a single point y. The time-optimal

13

problem with fixed terminal set is replaced with a sequence of fixed time and free terminal state problems with quality index

$$J = \Sigma \, \rho_i \big(x_i(t_f) - y_i\big)^2 \tag{24}$$

Switching-time-variation method is used to solve this problem, and the desired final time is decreased if the resulting quality is zero or is increased in the opposite case. This iteration is repeated until we get to the limit time t_f^* below which the quality is always positive, i.e., it is not possible to find a bang-bang control transferring the system from x_0 to y.

The optimization method described in Section 3.2 was modified somewhat in details of the gradient minimization routine. Instead of performing single step in the direction, a directional search is performed with constrained step size. A combination of two-point gradient parabolic approximation and three-point non-gradient parabolic approximation is used to find the minimum in the direction. The generation of the descent direction is also somewhat different. First, if any of the constraints (15) are active and the gradient points outside the feasible region, the gradient is projected on the proper constraining hyperplane. The special structure of constraints causes the projection to consist solely of putting the appropriate coordinates of the gradient to zero. Then the direction is tangent to the constraining hyperplane, and we get an optimization problem of reduced dimensionality. This problem is solved using a conjugate gradient method in the version proposed in [6]. The conjugate gradient is restarted not only every N iterations, where N is the current dimensionality of the problem, but also whenever the set of active constraints changes - i.e., when the algorithm hits or leaves a constraining hyperplane. The termination of the procedure occurs when the projected gradient is zero - i.e., no feasible descent step is possible, or equivalently when the dimension of the current optimization hyperplane becomes zero.

The calculation of the quality criterion and of its gradient involves numerical integration of Eqs (18) and (19). This is done using a fourth-order Runge-Kutta integration method. To integrate the

14

adjoint equation (19) the whole state trajectory resulting from integrating (18) must be stored, but for calculation only a small number of points from the costate trajectory is needed.

The program is written in the fashion enabling easy substitutions of different plant models and different optimization tasks. To use another model one has simply to provide the routines calculating the right-hand sides of Eqs. (18) and (19). The problem is defined in a straightforward fashion by setting the values of initial state, terminal state, initial estimate of the final time, etc., in the main routine. The whole program is written in C programming language, and although compiled and run on an IBM PC, it may be easily ported to any machine with C language compiler. The only difficulty that may occur with more complex systems is the rather severe storage requirements - whole state trajectory has to be stored with sufficiently small discretization step in order to calculate the gradient. And, of course, there will always be the problem with the speed of calculations for higher dimensional systems.

3.5 Simulations

The program described above was tested on a model previously used (in our NASA project), i.e., a simplified longitudinal aircraft model of second order (so called "Stalford model") described in [6].

For the aircraft model the problem solved was to increase or decrease the value of angle of attack with the requirement that the maneuver should move the system from the equilibrium corresponding to starting value of the angle of attack to the equilibrium corresponding to its final value. The control signal, the elevator angle, was assumed to be between 0 and -20 degrees. The series of maneuvers simulated was labeled in the following way:

maneuver A: from 0 to 15 degrees; A': from 15 to 0 degrees

maneuver B: from 0 to 18 degrees; B': from 18 to 0 degrees

maneuver C: from 0 to 20 degrees; C': from 20 to 0 degrees

maneuver D: from 5 to 15 degrees, D': from 15 to 5 degrees

maneuver E: from 5 to 18 degrees, E': from 18 to 5 degrees

maneuver F: from 5 to 20 degrees, F': from 20 to 5 degrees

maneuver G: from 10 to 15 degrees, G': from 15 to 10 degrees

maneuver H: from 10 to 18 degrees, H': from 18 to 10 degrees

maneuver I: from 10 to 20 degrees, I': from 20 to 10 degrees

maneuver J: from 15 to 18 degrees, J': from 18 to 15 degrees

maneuver K: from 15 to 20 degrees, K': from 20 to 15 degrees

The results of the optimization for each of these maneuvers is depicted in Figures 6-16. It may be observed that for all of them the time-optimal control had only one switch. In all cases the time-optimal trajectory for α had a substantial overshoot and consisted of an almost linear first portion with high slope before the switch and of slowly decreasing second portion.

3.6 Conclusions

The computer program presented here is suitable for calculating the time-optimal controls for arbitrary finite-dimensional systems which are affine in control. The simulation results discussed here have mainly testing value showing that the program is in operation. The next step should be to use the program on some more complex models such as the fourth-order longitudinal-aircraft model, which together with a linear actuator is definitely affine in control. The resulting time-optimal trajectories for different maneuvers could be used as benchmark tests for other controllers or as reference trajectories for time-series-based, adaptive, one-step-ahead (or many-steps-ahead) control. The work on this is underway.

16
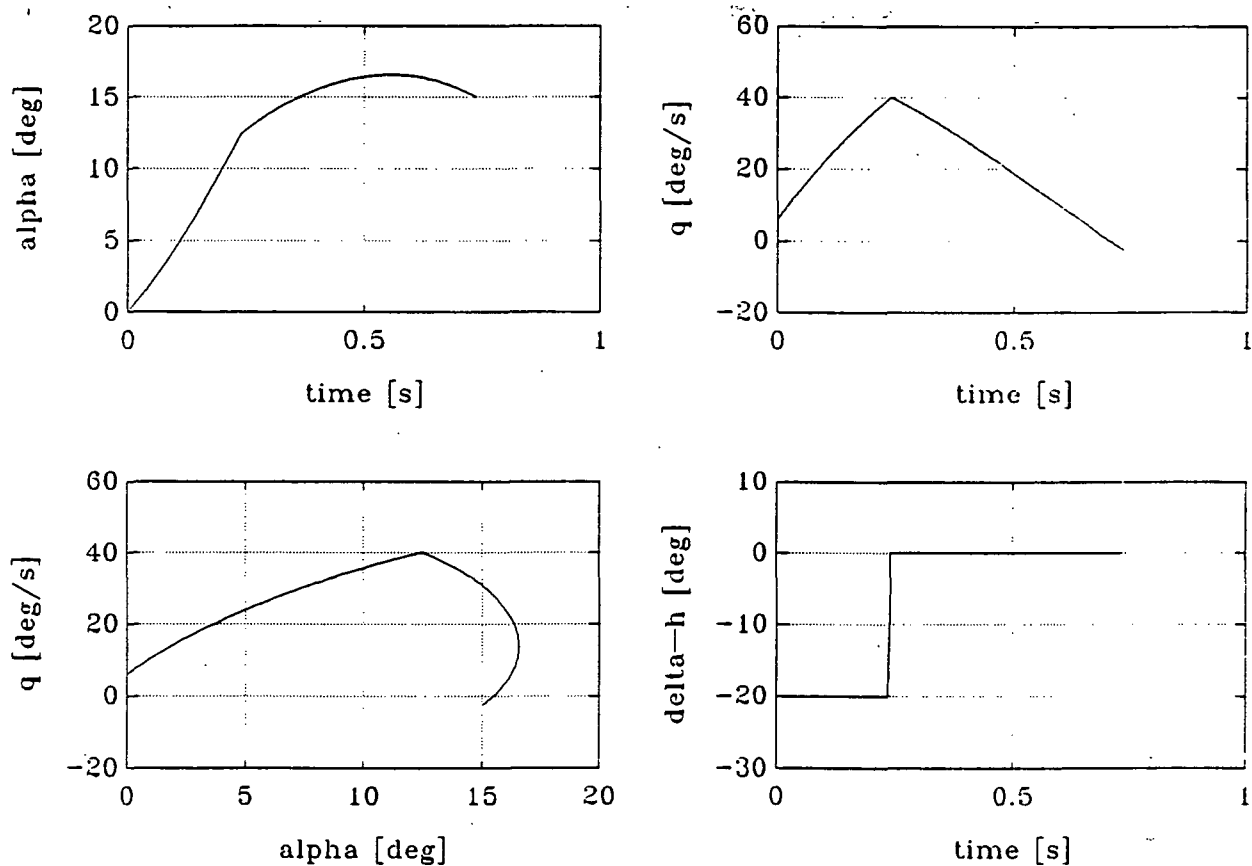
Figure 6a. Maneuver A (from 0 to 15 degrees)
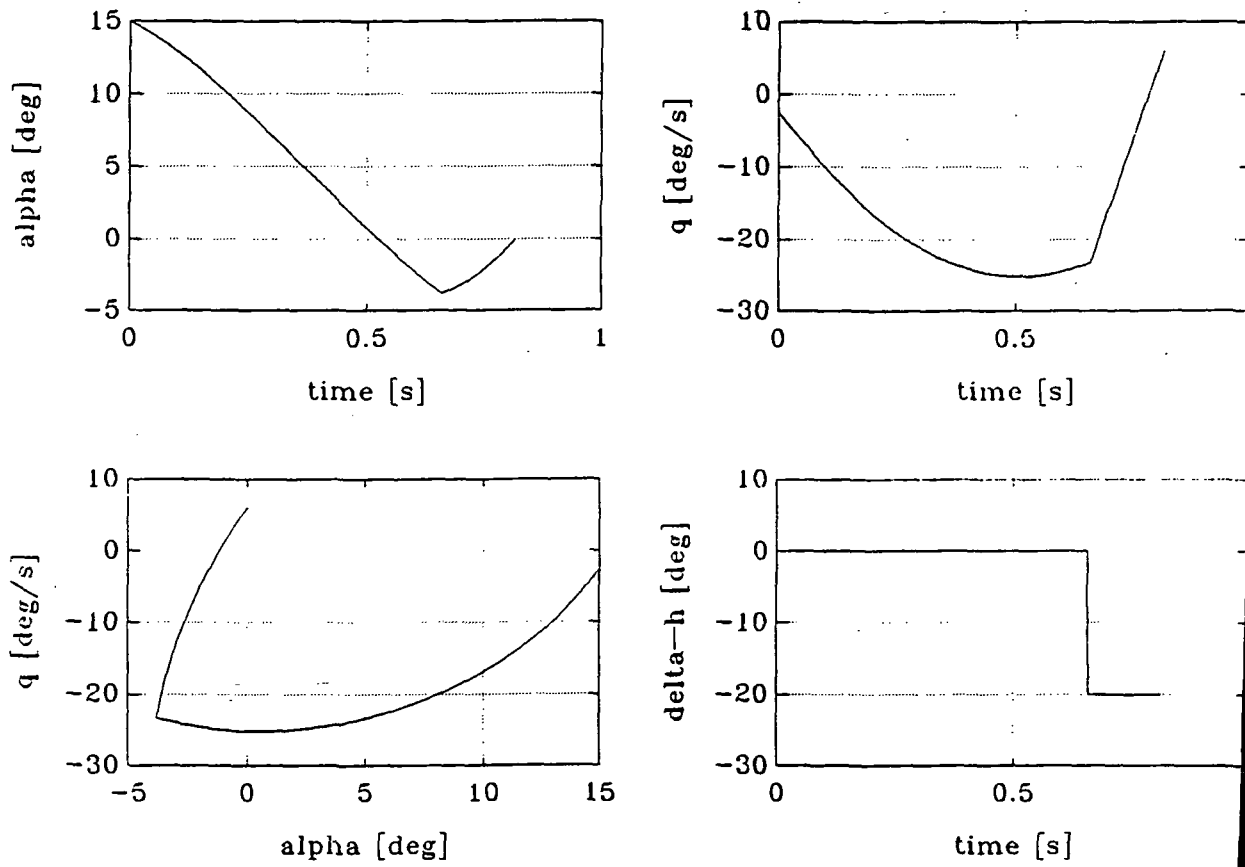
Figure 6b. Maneuver A' (from 15 to 0 degrees)
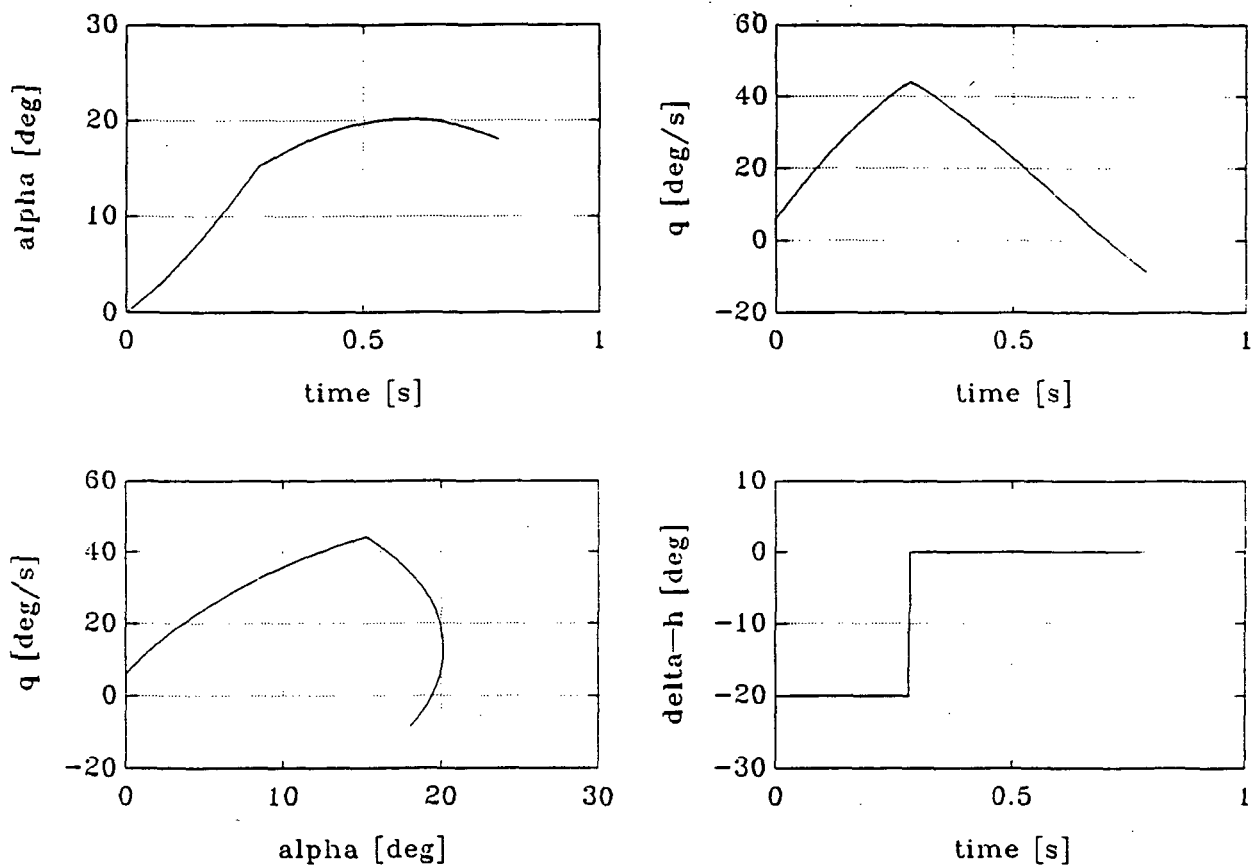
Figure 7a.   Maneuver B (from 0 to 18 degrees)
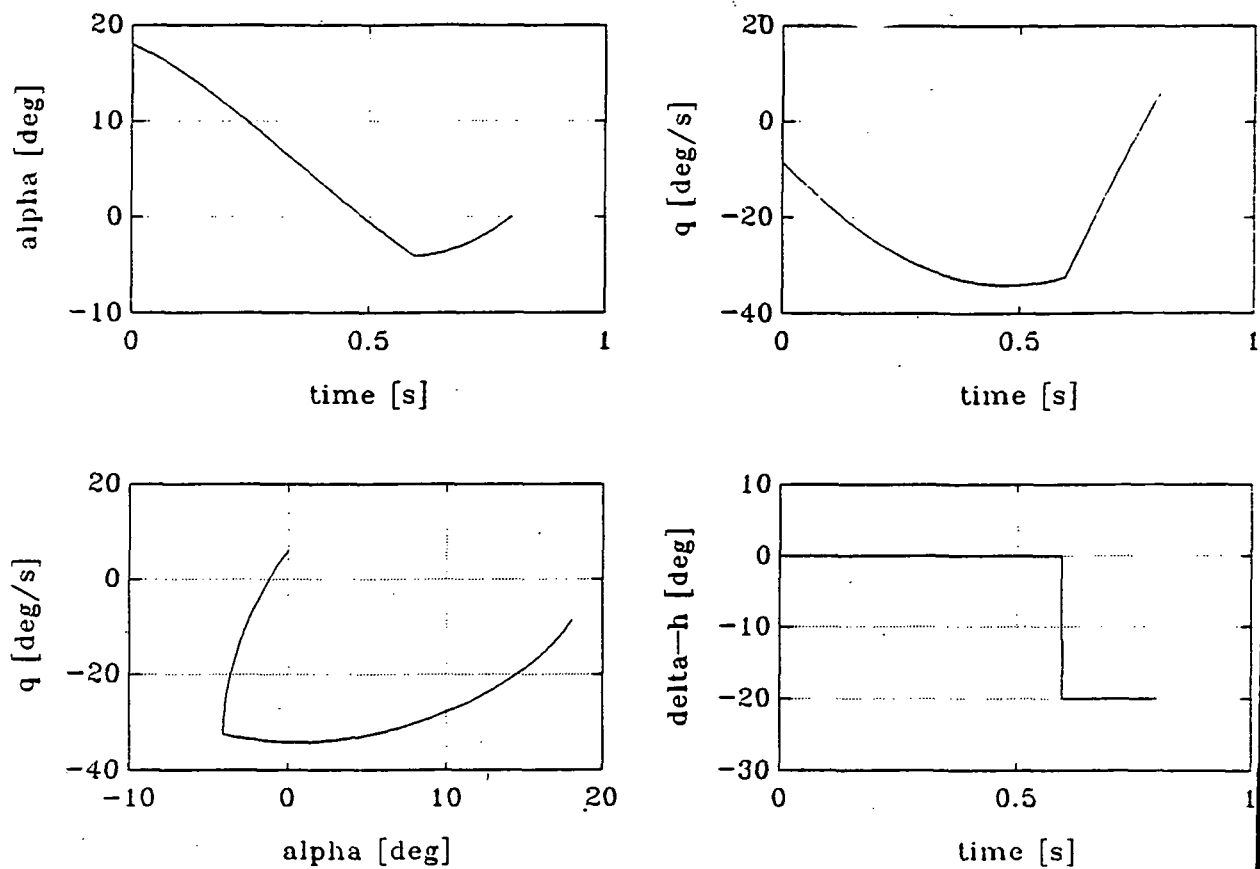
Figure 7b. Maneuver B' (from 18 to 0 degrees)
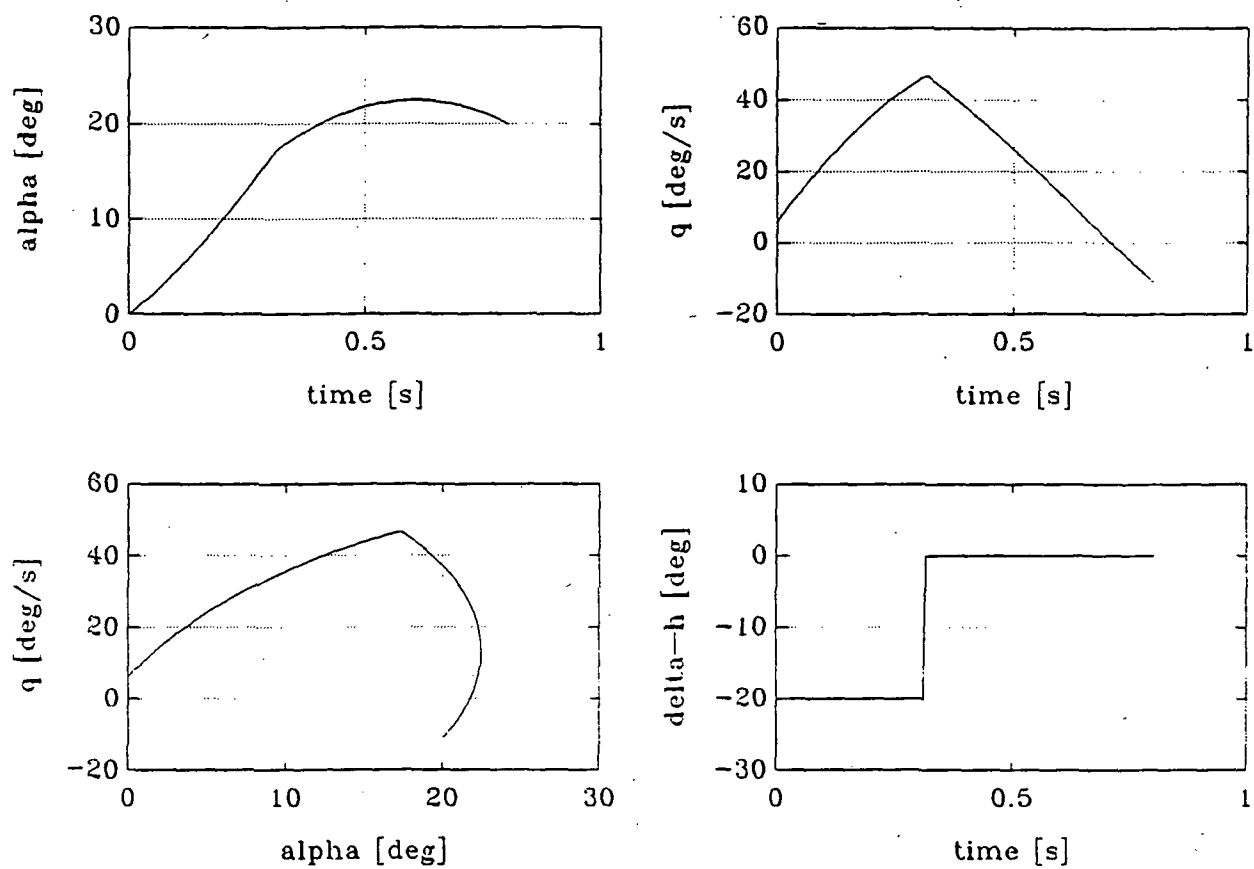
Figure 8a. Maneuver C (from 0 to 20 degrees)
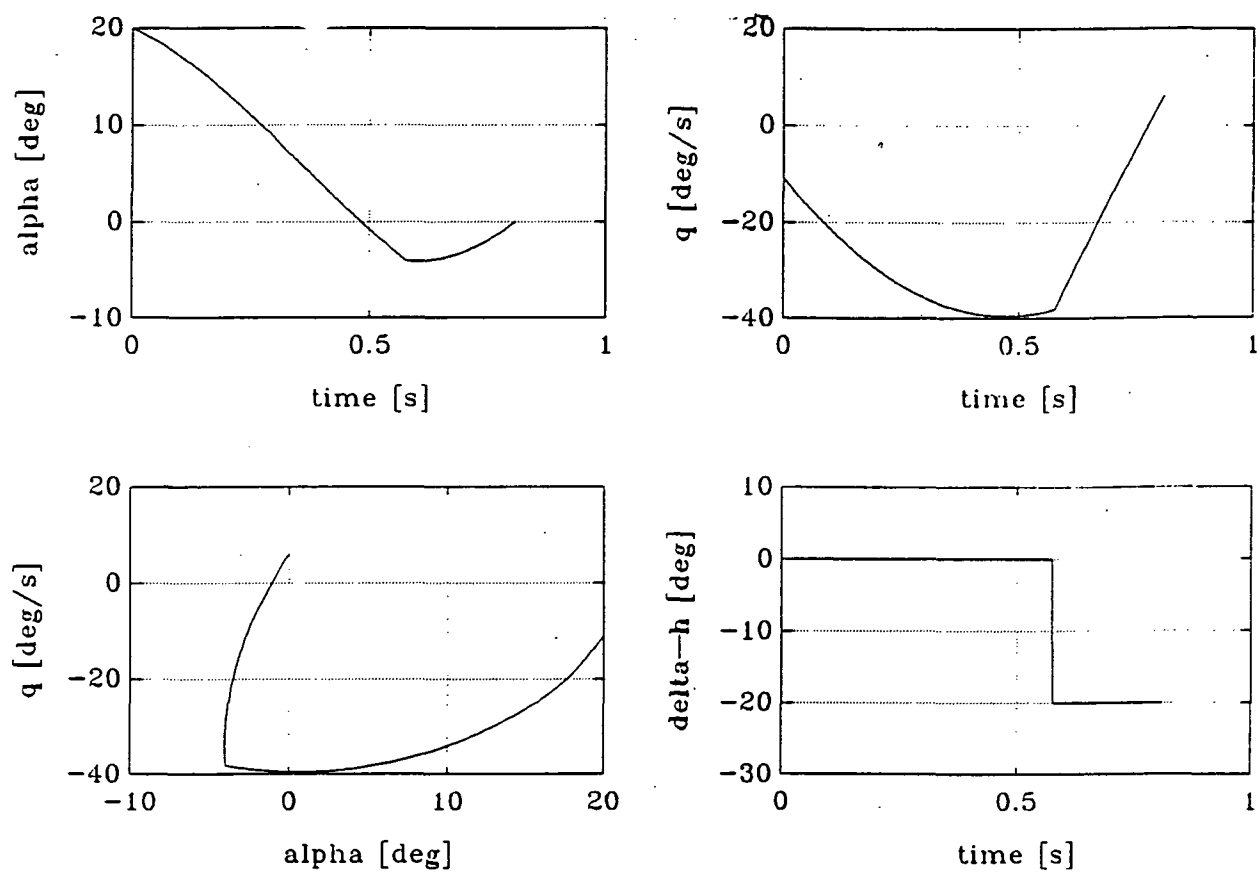
Figure 8b. Maneuver C' (from 20 to 0 degrees)
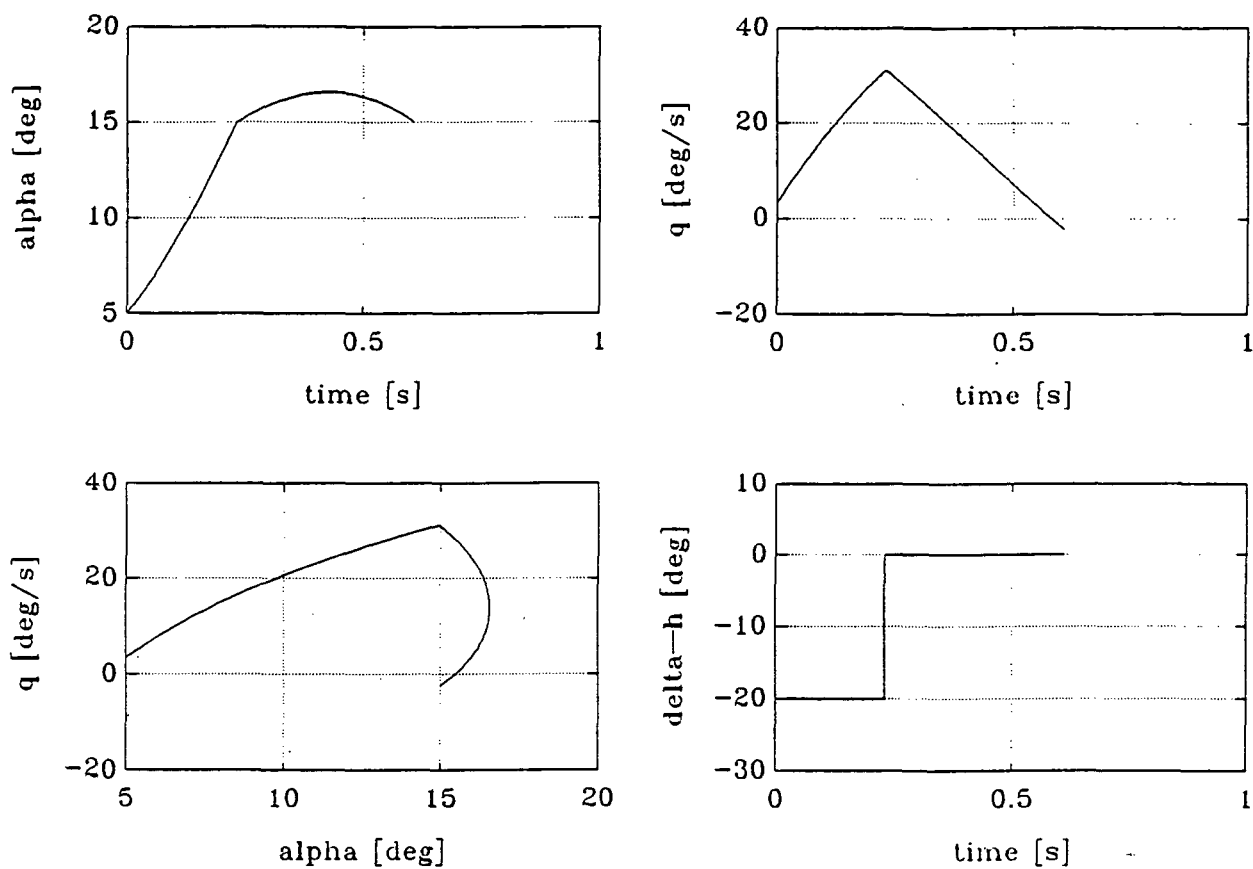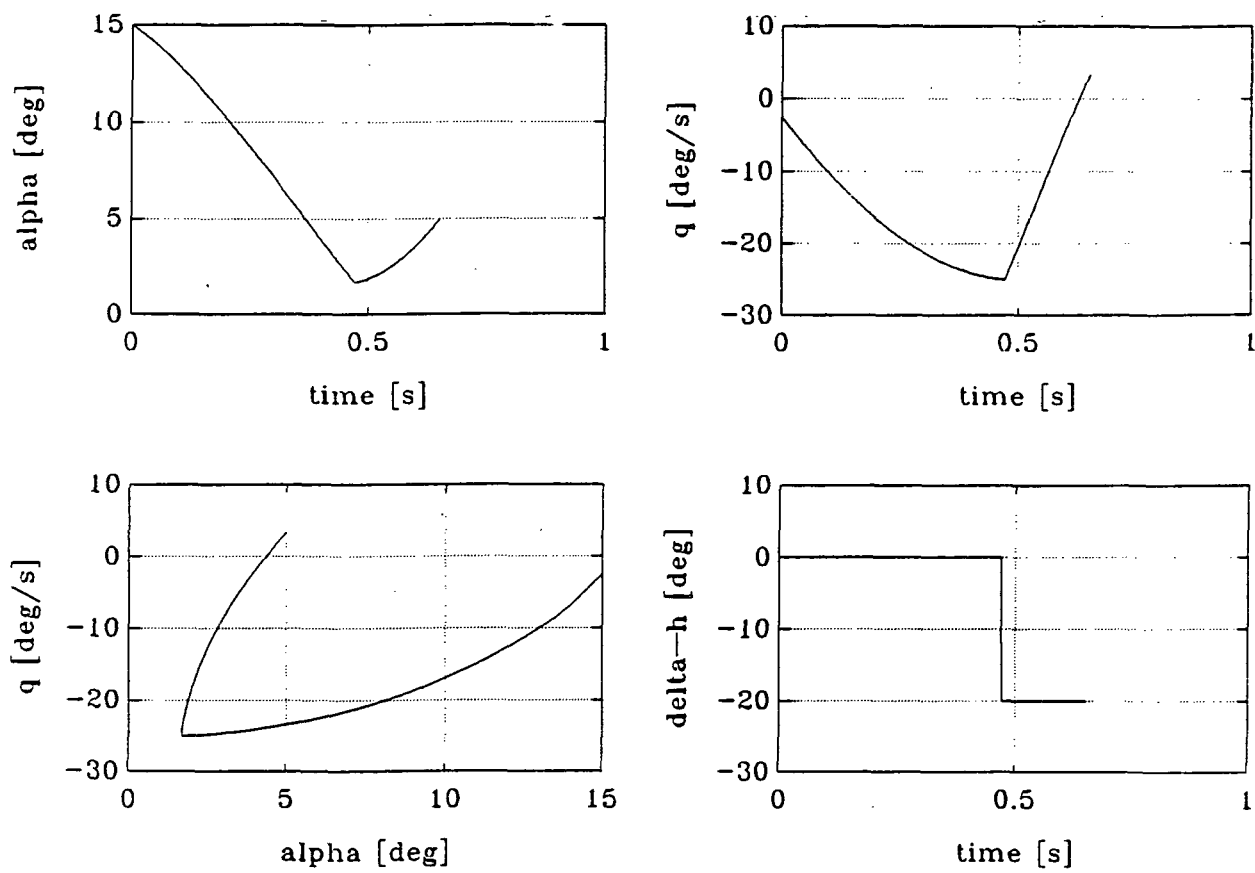
Figure 9a. Maneuver D (from 5 to 15 degrees)

Figure 9b. Maneuver D' (from 15 to 5 degrees)

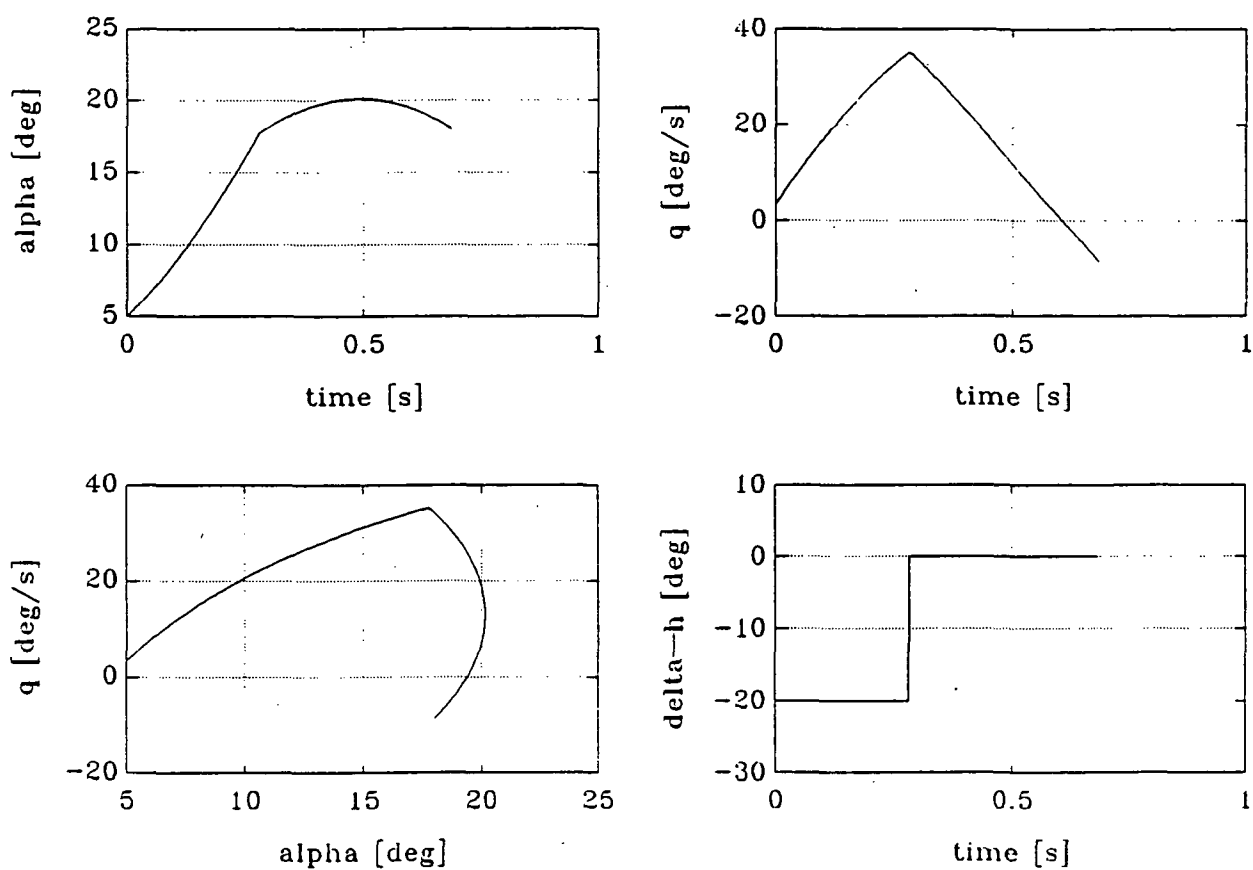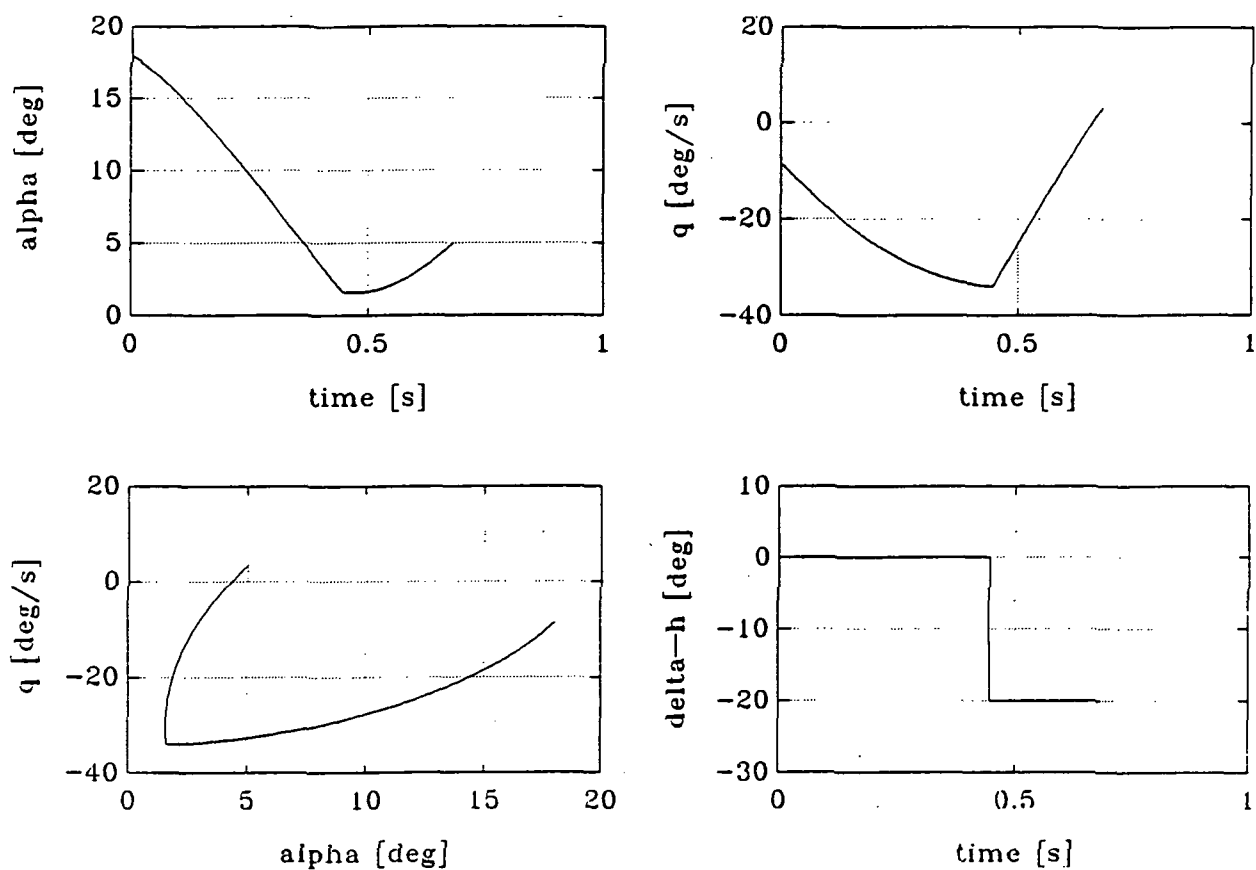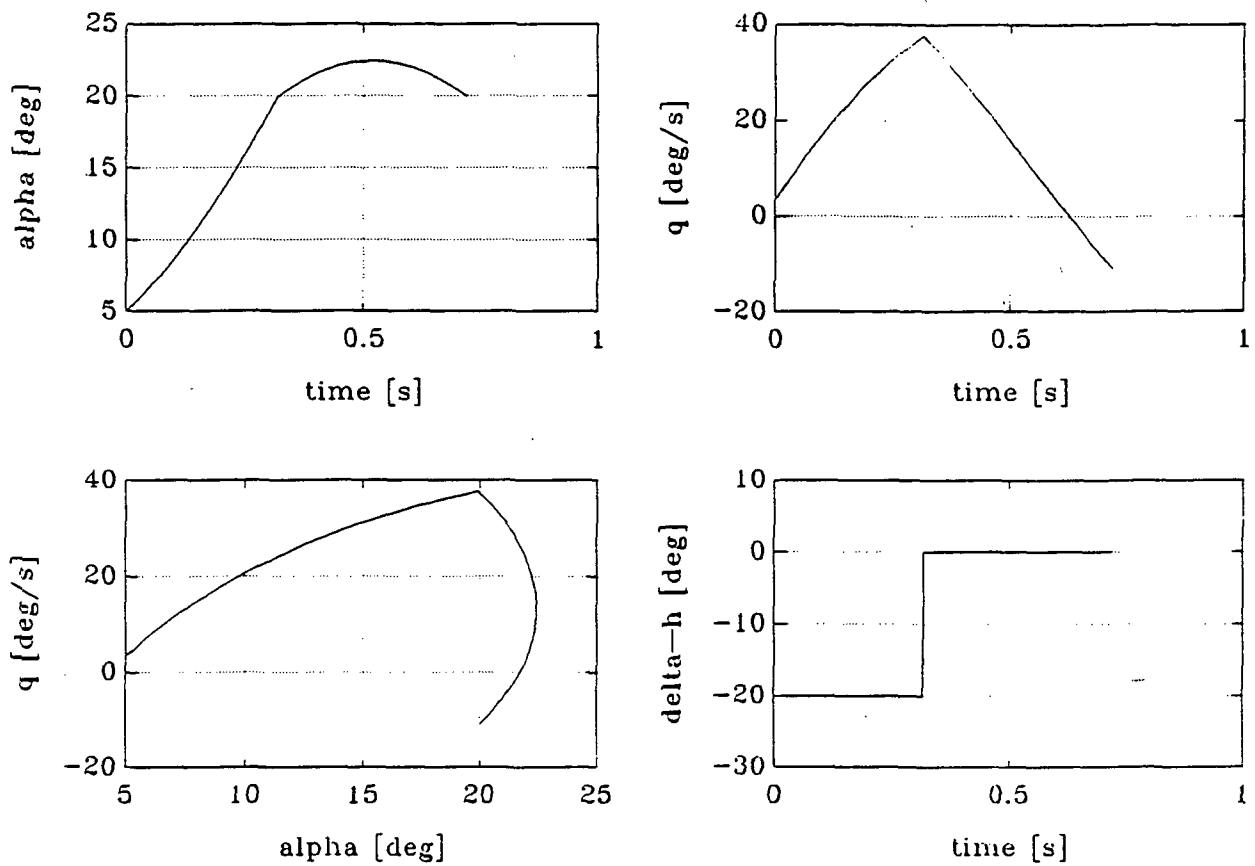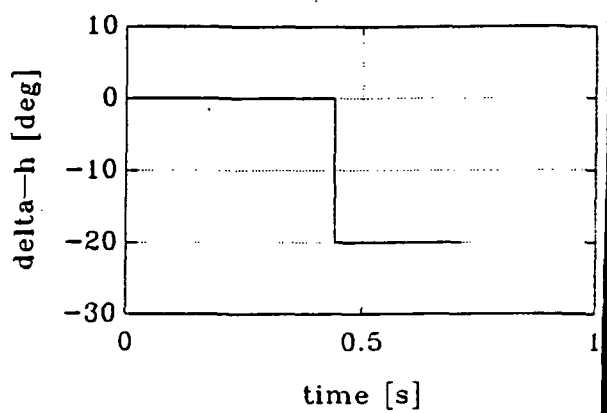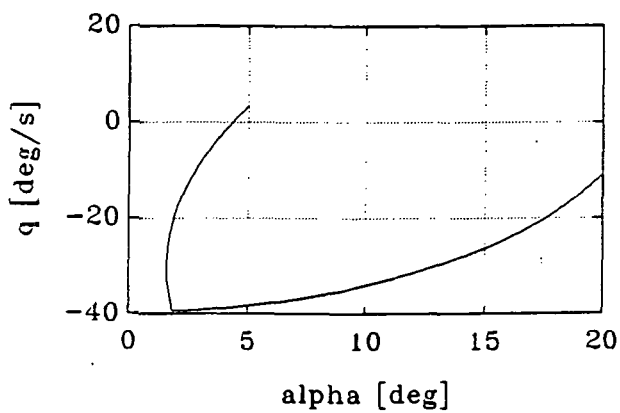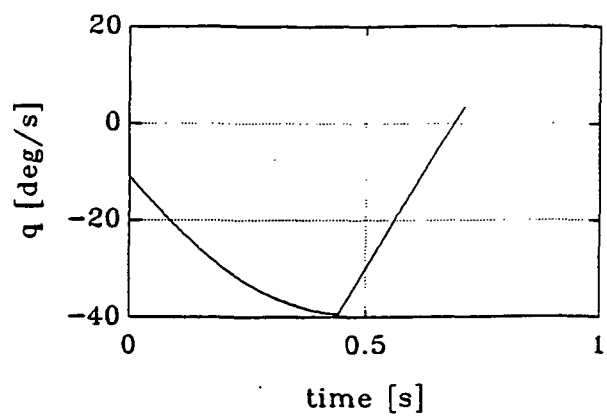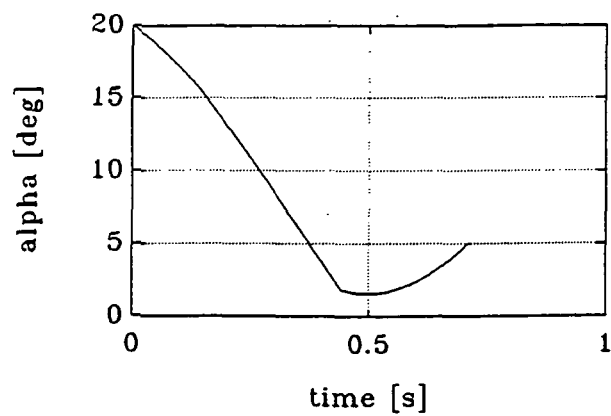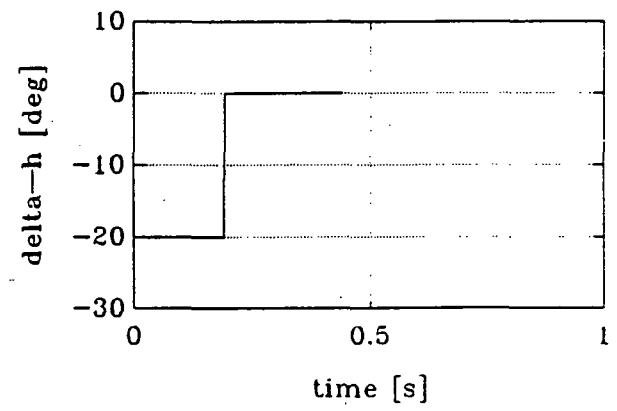Figure 10a. Maneuver E (from 5 to 18 degrees)

Figure 10b. Maneuver E' (from 18 to 5 degrees)

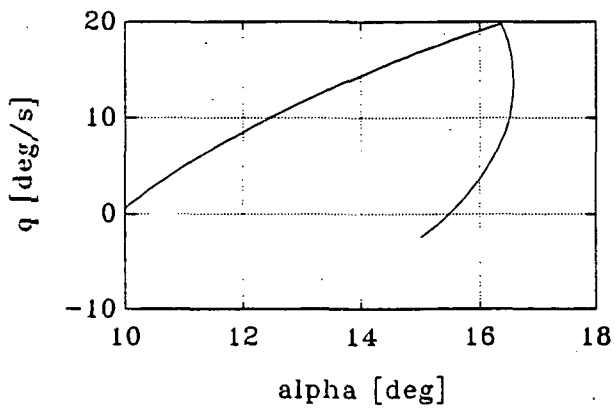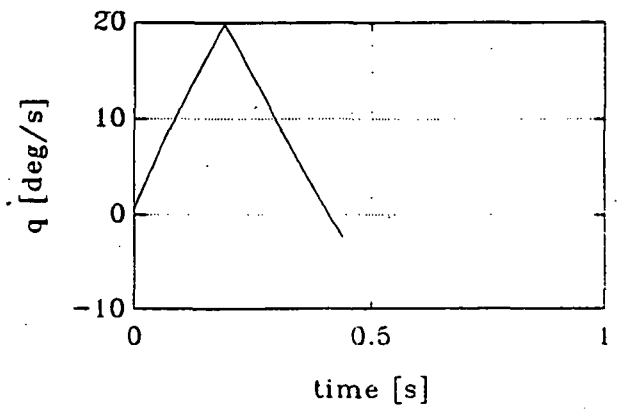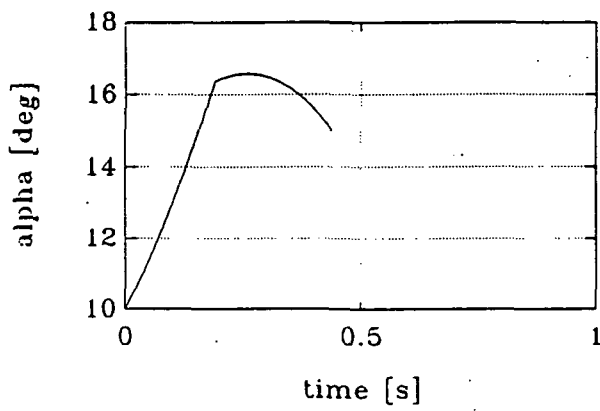Figure 11a. Maneuver F (from 5 to 20 degrees)

Figure 11b. Maneuver F' (from 20 to 5 degrees)
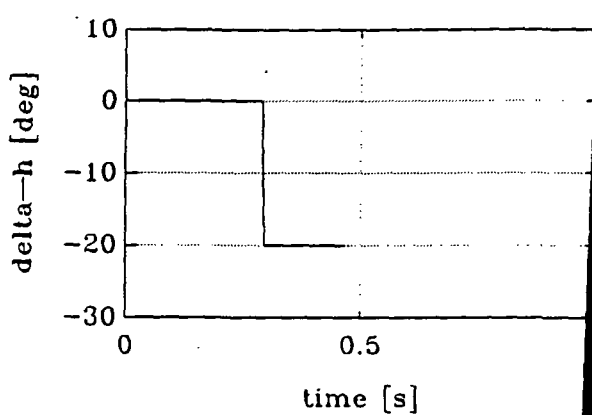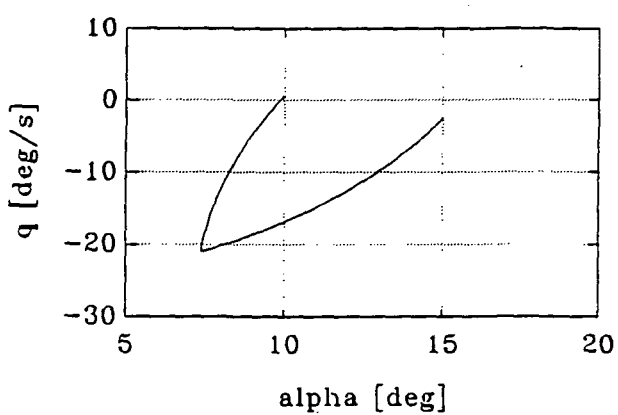
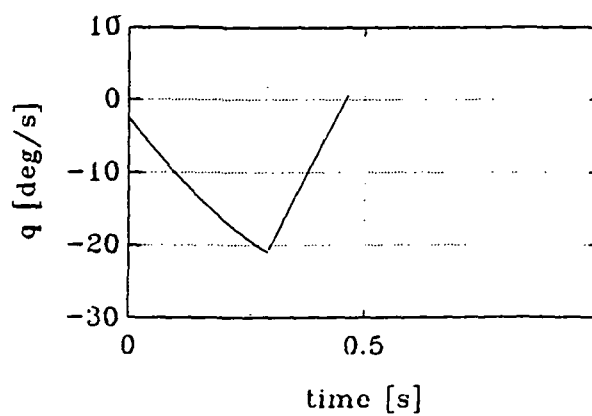Figure 12a. Maneuver G (from 10 to 15 degrees)

Figure 12b. Maneuver G' (from 15 to 10 degrees)

Figure 13a. Maneuver H (from 10 to 18 degrees)

Figure 13b. Maneuver H' (from 18 to 10 degrees)

32

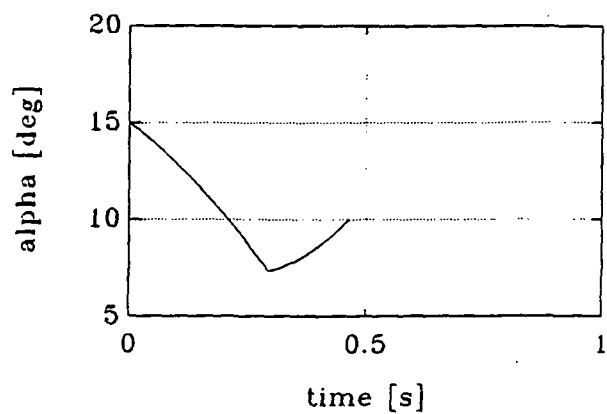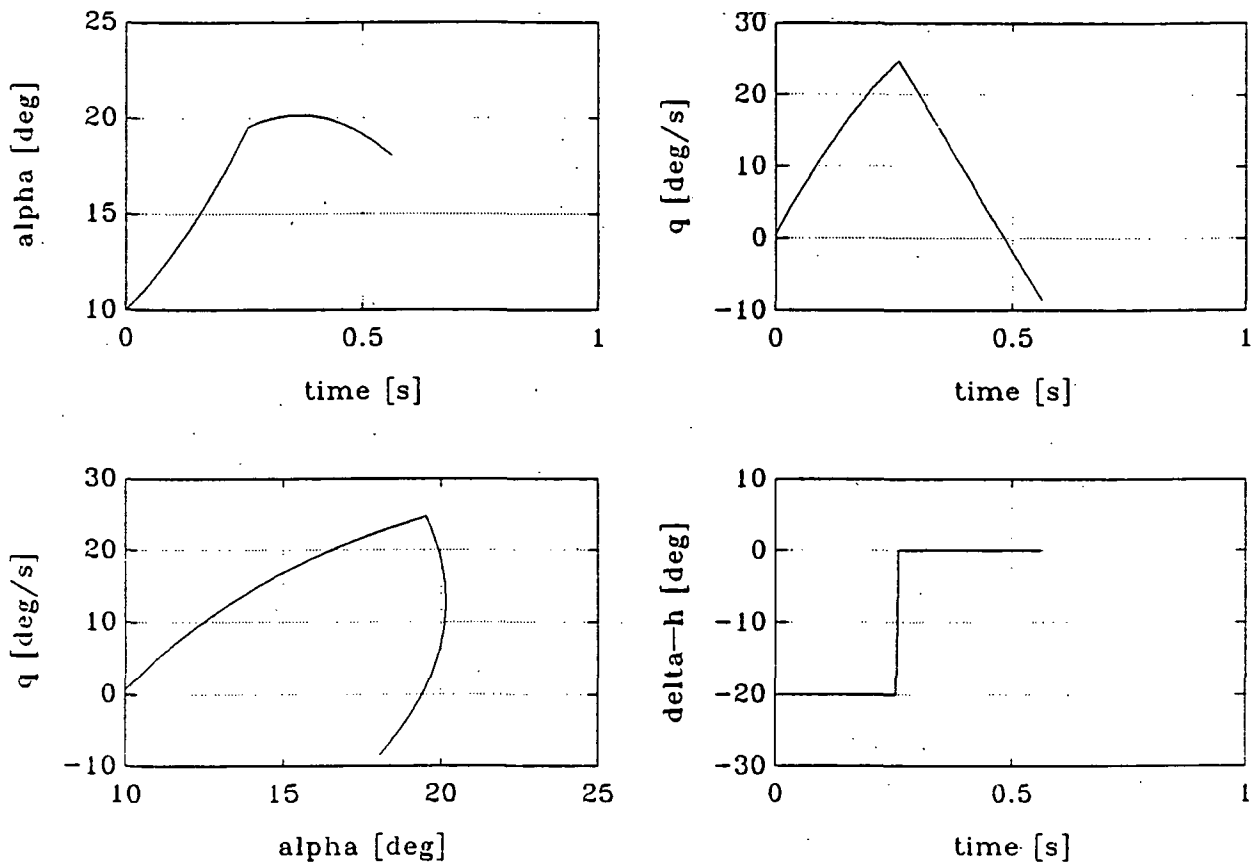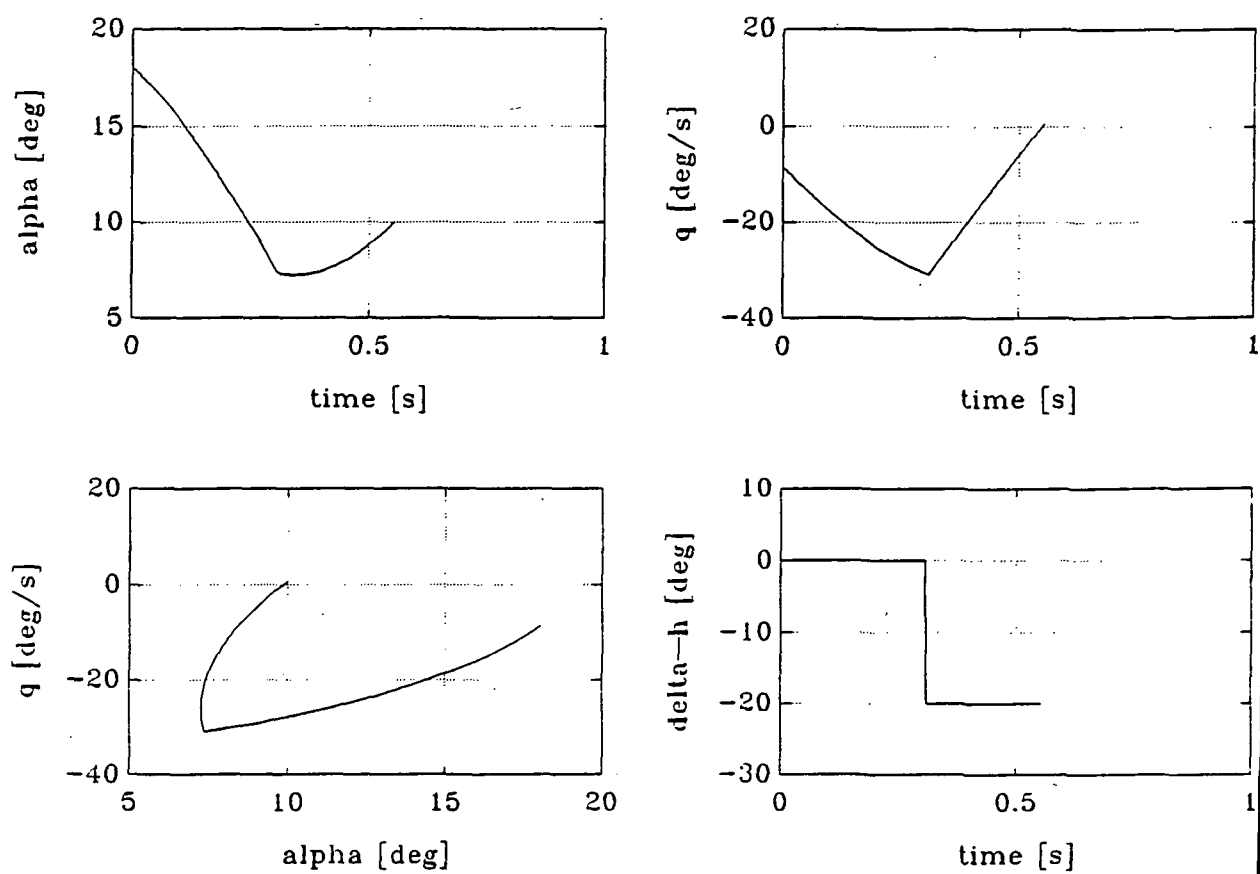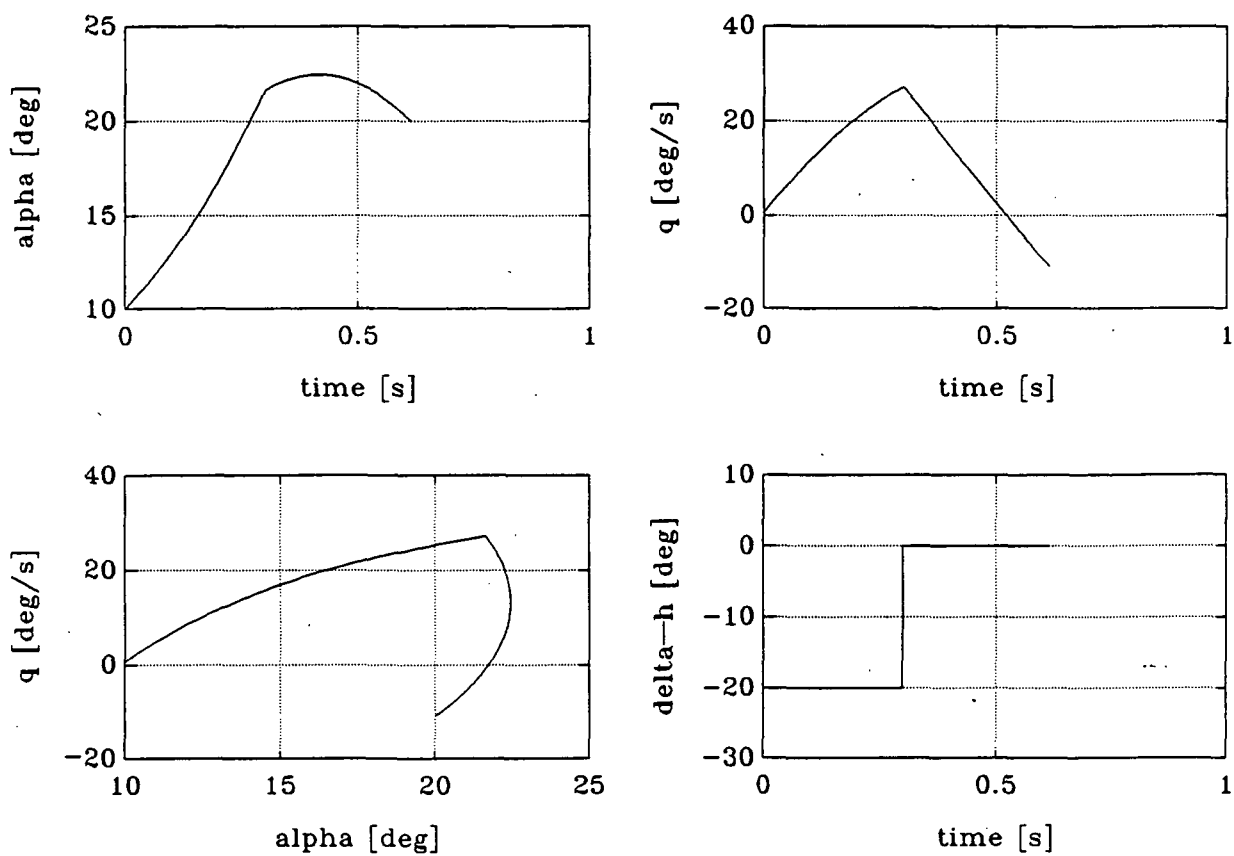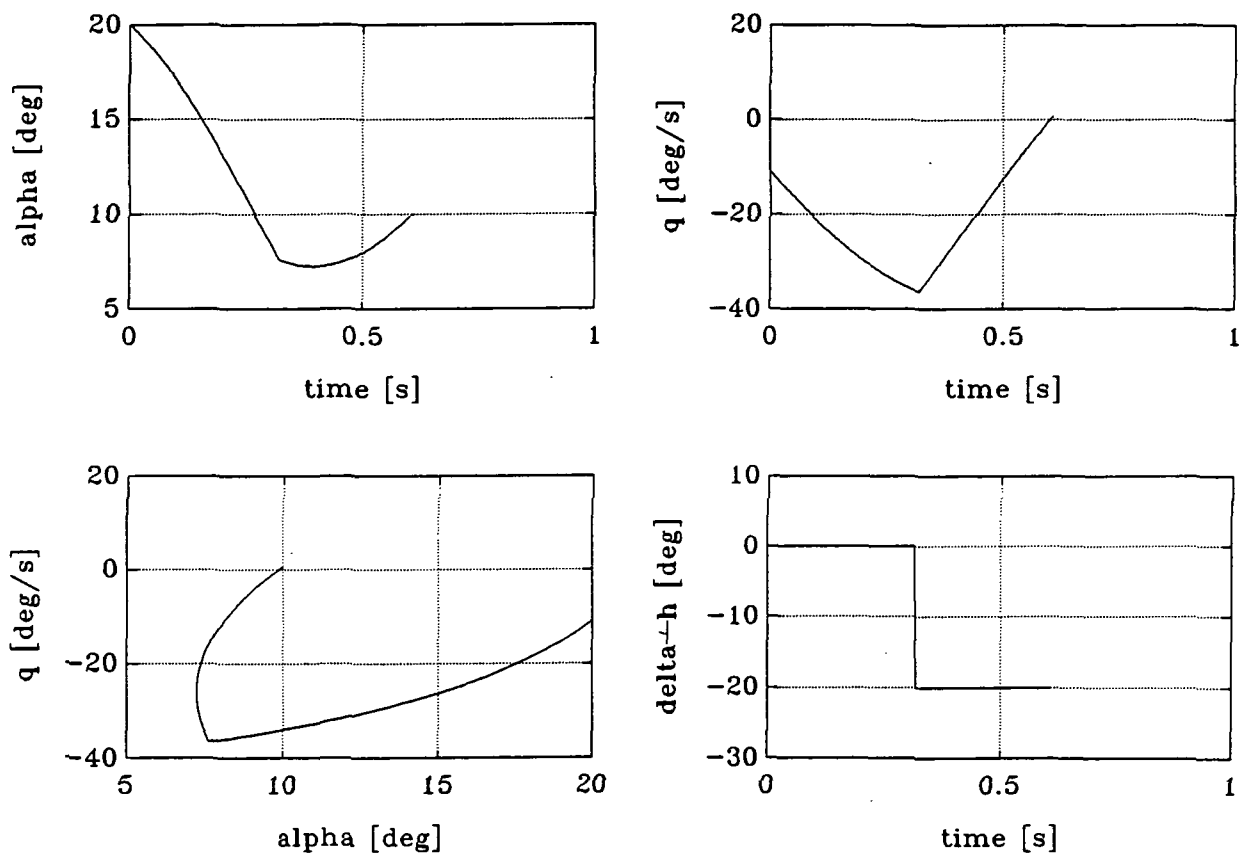Figure 14a. Maneuver I (from 10 to 20 degrees)

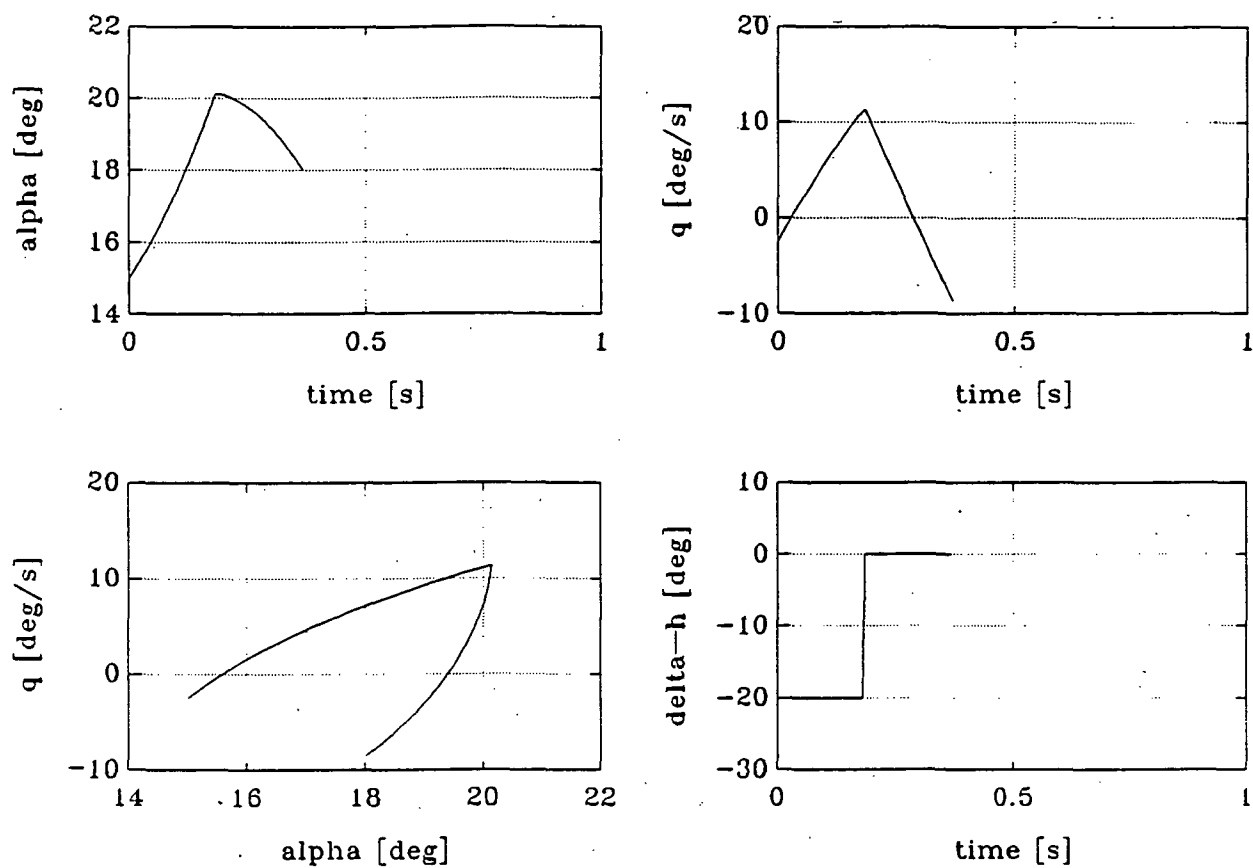Figure 14b. Maneuver I⁻ (from 20 to 10 degrees)

Figure 15a. Maneuver J (from 15 to 18 degrees)

35
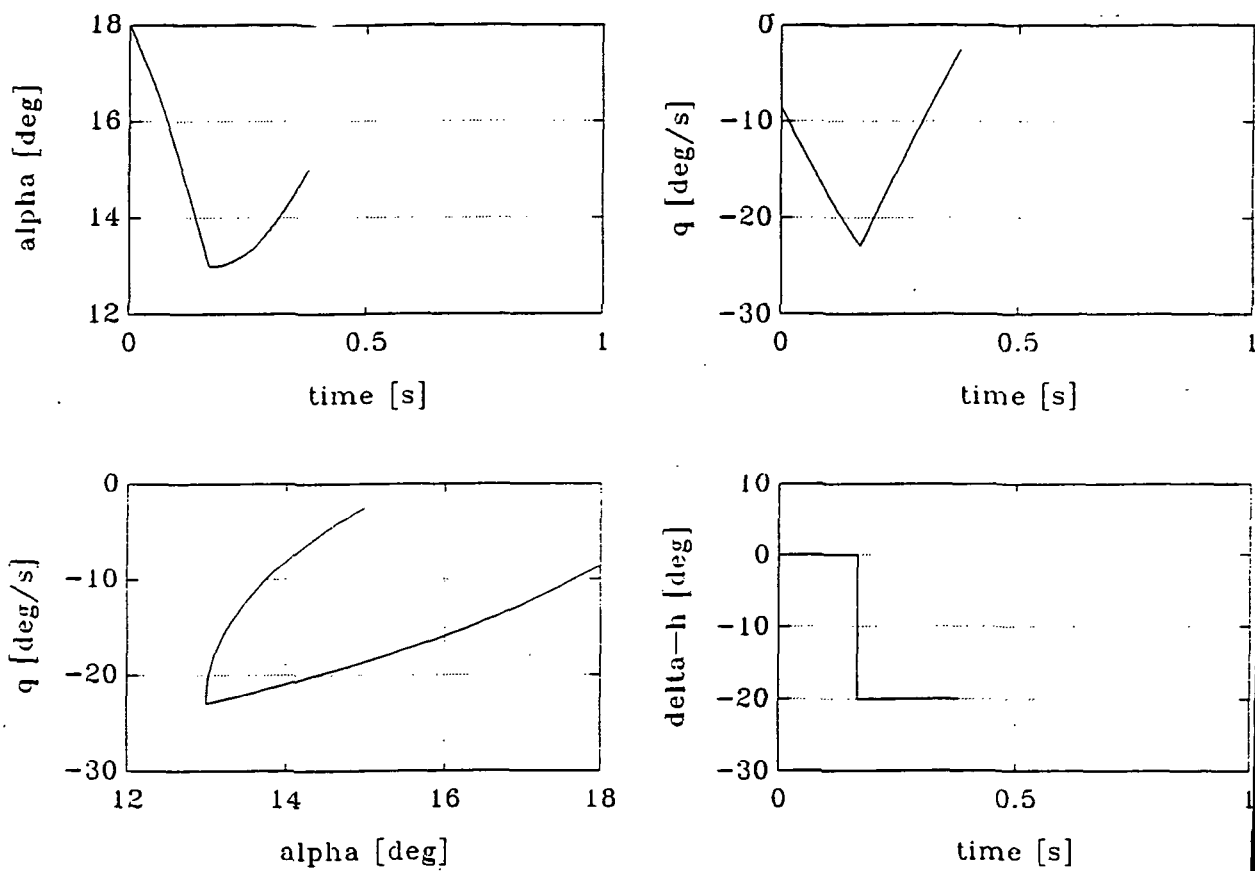
Figure 15b. Maneuver J' (from 18 to 15 degrees)
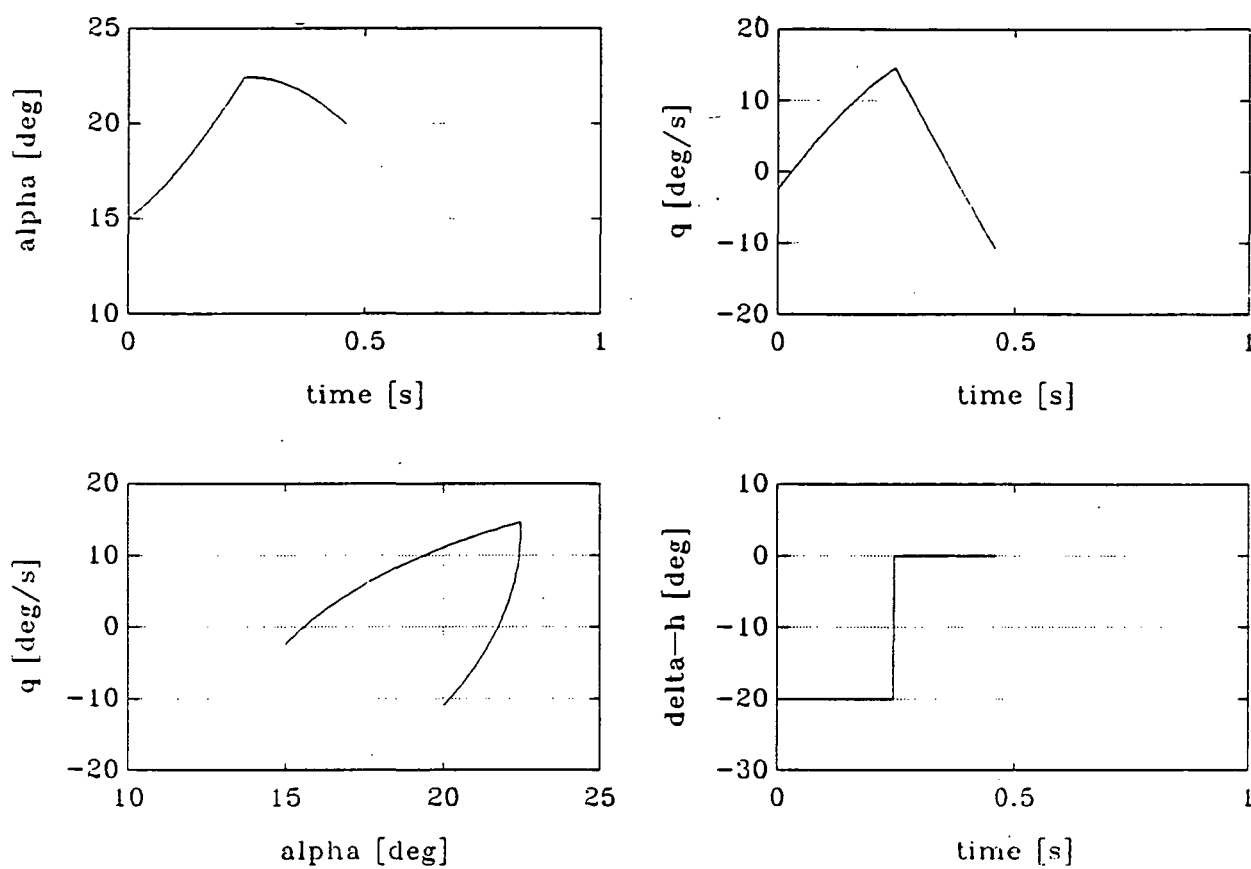
Figure 16a. Maneuver K (from 15 to 20 degrees)
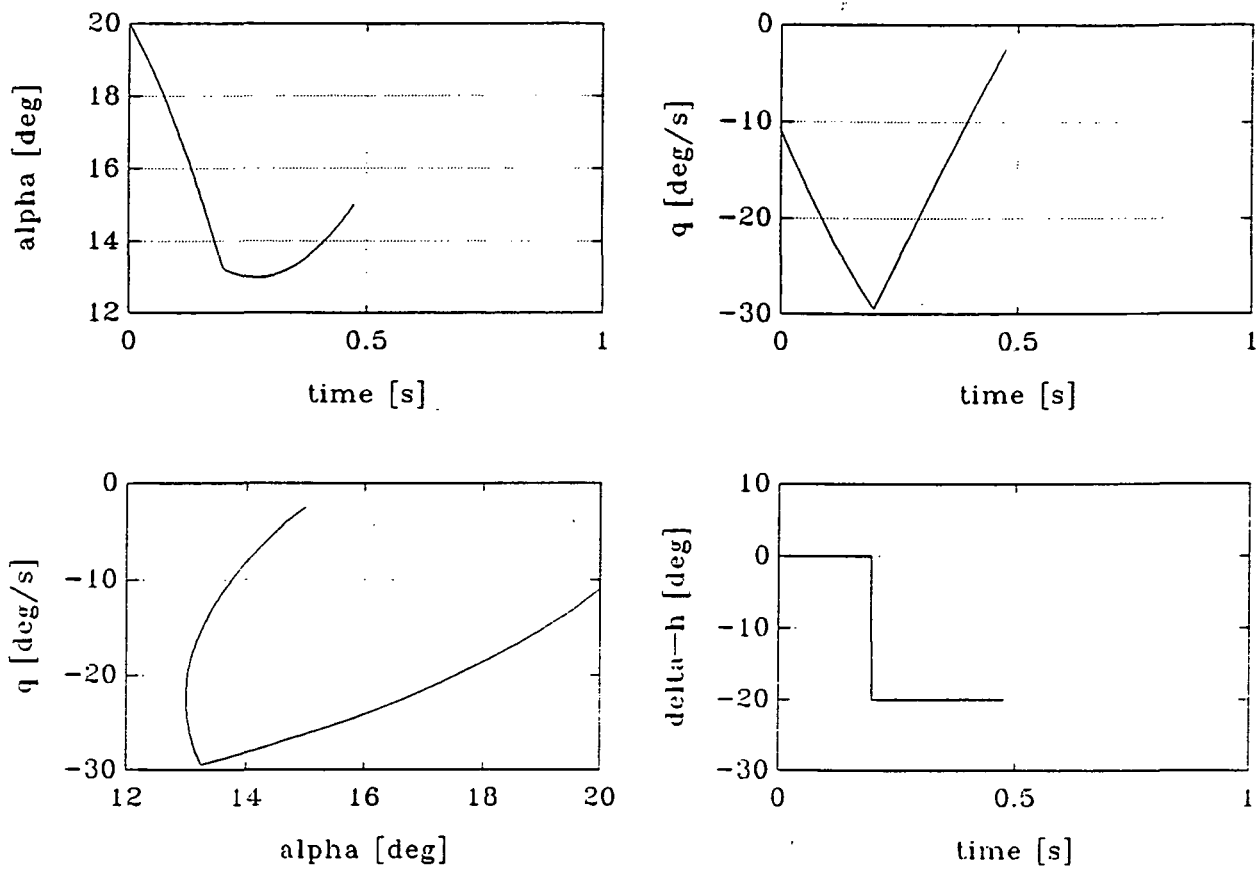
Figure 16b. Maneuver K' (from 20 to 15 degrees)

38

4. REFERENCES

[1] H. Stalford, W.T. Bauman, F.E. Garret, and T.L. Herdman, "Accurate modeling of nonlinear systems using Volterra series submodels," Proc. American Control Conf., Minneapolis, pp. 886-891, 1987.

[2] A.J. Ostroff, "Application of variable-gain output feedback for high-alpha control," AIAA Guidance, Nav. & Control Conf. (Paper No. 89-3576), Boston, 1989.

[3] R.R. Mohler, et al., "Nonlinear stability and control study of highly maneuverable high performance aircraft," OSU ECE Report NASA 9101, August 1991.

[4] R.R. Mohler, *Nonlinear Systems: V. 2 Applications to Bilinear Control*, Prentice Hall, 1991.

[5] S.F. Moon, "Optimal control of bilinear systems and systems linear in control," Ph.D. dissertation, The University of New Mexico, 1969.

[6] H. Hermes and J.P. Lasalle, *Functional Analysis and Time Optimal Control*, Academic Press, 1969.

[7] W. Hejo, *Theory of Time Optimal Control and Its Applications*, PWN, Warsaw, 1990 (in Polish).

APPENDIX A

Project Publications

PROJECT PUBLICATIONS

(Supported Wholly or in Part by NASA Grant)

1. R.R. Zakrzewski and R.R. Mohler, "On Nonlinear Model Algorithm Controller Design," *Proceedings*, IFIP Conf. Sys. Modeling and Optimiz., Zurich, 1991. (See Appendix B)

2. R.R. Mohler, V. Rajkumar, and R.R. Zakrzewski, "Nonlinear Time-Series Based Adaptive Control Applications," *Proceedings*, IEEE Conf. Decision & Control, Brighton, 1991.

3. R.R. Mohler, *Nonlinear Systems: Vol. 2 Applications to Bilinear Control*, Prentice Hall, Englewood Cliffs, NJ, 1991.

4. R.R. Mohler, V. Rajkumar, and R.R. Zakrzewski, "On Discrete Nonlinear Self-Tuning Control," *Proceedings*, Korean Control Conf., Seoul, 1991.

5. R.R. Mohler, R. Zakrzewski, S. Cho, and C. Koo, "New Results on Nonlinear Adaptive High Alpha Control," Comcon 3, Victoria, 1991.

6. J.E. Kurek, "Analysis of Nonlinear Stability Using Robust Stability Analysis for Linear Systems," submitted, 1992.

APPENDIX B

On Nonlinear Model Algorithmic Controller Design

ON NONLINEAR MODEL ALGORITHMIC CONTROLLER DESIGN

R.R. Zakrzewski and R.R. Mohler
Oregon State University
Department of Electrical and Computer Engineering
Corvallis, OR 97331
USA

1. INTRODUCTION

Two nonlinear algorithmic controllers, MAC, are studied here. One uses a block-canceling Volterra approximation, and the other MAC consists of solving an approximating polynomial time series instate and control. Both methods synthesize discrete control sequences and are applied successfully to the control of a simple nonlinear longitudinal aircraft model for large variations in angle of attack.

The Volterra-series approach used here was introduced by Modyaev and Averina [1], and a form of inverse generating control according to an assumed structure is presented by Harris [2]. This work formed the basis for the methods used here. The high angle-of-attack aircraft model derived by Stalford, et al. [3] was the plant simulated for the MAC application. In many traditional design studies, a sequence of linearized perturbation models are derived for different equilibrium flight conditions with linear controllers appropriately derived. Linear adaptive control can be derived according to nonlinear gain scheduling of the control law. A highly successful version of such control, which includes proportional plus integral plus filter (PIF) terms, is presented by Ostroff [4,5]. However, such designs usually require a large number of set-point design computations, and may have stability problems for large fast changes in angle of attack and/or mach number.

For generation of the nonlinear control, a nonlinear time-series based model reference is used. In order to identify such model, experimental data was collected for angle of attack (α) and pitch rate (q) subject to random steps of control

(stabilator, δ). To capture such phenomena as limit cycles in the data the steps were rather long (40 s). There were 64 such steps with time discretization of 0.1 s resulting in 25,600 points in a state plane for 64 values of control.

For a least-squares simulated data fit, the following approximation was surprisingly accurate:

$$
\begin{aligned}
\alpha(k+1) = & \ p_{1\alpha}\alpha(k) + p_{2\alpha}\alpha^2(k) + p_{3\alpha}\alpha^3(k) + \\
& p_{4\alpha}q(k) + p_{5\alpha}q(k)\alpha(k) + p_{6\alpha}q(k)\alpha^2(k) + p_{7\alpha}q(k)\alpha^3(k) + \\
& p_{8\alpha}u(k) + p_{9\alpha}u(k)\alpha(k) + p_{10\alpha}u(k)\alpha^2(k) + p_{11\alpha}u(k)\alpha^3(k) + p_{12\alpha} \\
q(k+1) = & \ p_{1q}\alpha(k) + p_{2q}\alpha^2(k) + p_{3q}\alpha^3(k) + \\
& p_{4q}q(k) + p_{5q}q(k)\alpha(k) + p_{6q}q(k)\alpha^2(k) + p_{7q}q(k)\alpha^3(k) + \\
& p_{8q}qu(k) + p_{9q}u(k)\alpha(k) + p_{10q}u(k)\alpha^2(k) + p_{11q}u(k)\alpha^3(k) + p_{12q}
\end{aligned}
\tag{1}
$$

Even limit cycles are accurately rendered by this model, as well as the stable zone behavior, although large discrepancies occur when the control values are close to the stable/unstable zones border.

2. ADAPTIVE CONTROL APPROACHES

2.1 Nonlinear Volterra-Based Control

Here, as in [6], the Volterra series serves as a conceptual starting point for a nonlinear time series base control. Continuous time controllers based on Volterra series were systematically developed in [7] with formulae for the controller's kernels given those of the plant and of the desired feedback system. In particular, the problem of so-called exact feedback linearization was solved here. However, those formulae may be of limited practical value because of the properties of Volterra series under feedback. The problem is that even finite (e.g., second order) Volterra series of the open loop results in infinite Volterra series of the closed loop. This makes it necessary for the controller to include theoretically an infinite number of

compensating terms even for a quadratic system. The same problem for the discrete time systems was treated in [1] with multidimensional Z transforms to derive the set of formulae equivalent to those for so-called exact feedback linearization [8]. However, they also provided a very elegant transformation of which results in a controller requiring only as many Volterra terms as there are in the assumed plant.

One attractive feature of this controller is that its structure makes it possible to utilize it not only with models represented in the form of Volterra series, but in fact with any model with easily divided linear and nonlinear parts of the dynamic equations such as (2) above.

The following algorithm results:

a) according to the linear part of the plant, calculate the linear control $u_L(k)$

b) calculate the predicted value of the output at the moment k

$$\hat{y}(k) = L(y(k-1),...,y(k-M),u(k-1),...,u(k-M))$$
$$N(y(k-1),...,y(k-M),u(k-1),...,u(k-M)$$

c) solve the "linearizing" control equation for $x(k)$ such that

$$N(\hat{y}(k),y(k-1),...,y(k-M+1),u_L(k)-x(k),u(k-1),...,u(k-M+1)) =$$
$$= L(x(k),x(k-1),...,x(k-M+1),\hat{y}(k),y(k-1),...,y(k-M+1))$$

3) calculate the control by

$$u(k) = U_L(k) - x(k)$$

This algorithm becomes a sort of prediction controller which tries to estimate the effects of the previous controls knowing the previous values of outputs and then to adjust the current value of control so that the nonlinear part of predicted output is canceled.

This discrete time nonlinear α control algorithm is generated according to an off-line identification of model (1) with a nonlinear aircraft simulation based on [3]. Also, a linear controller was designed according to the linear parts of (1)-(3).

The design was performed to obtain the closed loop model reference behavior of the form

$$G(z) = 0.05/(z^2 - 1.6z + 0.65)$$

In order not to cancel the zero of the plant, the observer polynomial (z-0.7) was introduced. The algorithm for the control value u(k) is as follows. First the estimate of the output at moment k is calculated from (1) with k replaced by k-1.

Then it can be shown that the control becomes

$$u(k) + \frac{p_{8\alpha}u_L(k) - \left(p_{2\alpha}\hat{\alpha}^2 + p_{3\alpha}\hat{\alpha}^3 + p_{5\alpha}\hat{\alpha}q + p_{6\alpha}\hat{q}\hat{\alpha}^2 + p_{7\alpha}\hat{q}\hat{\alpha}^3 + p_{12\alpha}\right)}{\left(p_{8\alpha} + p_{9\alpha}\hat{\alpha} + p_{10\alpha}\hat{\alpha}^2 + p_{11\alpha}\hat{\alpha}^3\right)} \quad (4)$$

with $\hat{\alpha}(k)$ and $\hat{q}(k)$ designating estimates taken from (1). It is seen that if there are no nonlinearities in the model the control reduces to a regular linear controller $u = u_L$.

Simulations were run to test the controller performance especially in the unstable range of angle of attack. The system is successfully stabilized and the transients are very smooth and without significant overshoots for the nonlinear control as demonstrated by Figure 1a. By different choice of the reference model it is possible to obtain much faster, but at the same time much more "nervous" transients. The elevator control is also relatively smooth and within the range corresponding to the terminal equilibria. As can be seen from Figure 1b, the similar linear control is unstable.

2.2 On-Line Adaptive MAC Algorithm

Model algorithmic control (MAC), described for example in [2], consists of solving the model equation for the value of control necessary to obtain required value of output. Usually this desired output trajectory is generated form the setpoint by means of a reference model. In case this model is linear, the algorithm in essence becomes a linearizing one.

Here, the controlled output is assumed to be the angle of attack such that the reference equation becomes:

$$\alpha_{ref}(k+1) = \hat{\alpha}_{mod}(k+1) + \left(\alpha(k-1) - \alpha_{mod}(k-1)\right) \tag{5}$$

with

$$\hat{\alpha}_{mod}(k+1) = p_\alpha^T \hat{\phi}(k)$$

$$\hat{\phi}(k) = [\hat{\alpha}, \hat{\alpha}^2, \hat{\alpha}^3, q, q\hat{\alpha}, q\hat{\alpha}^2, q\hat{\alpha}^3, u, u\hat{\alpha}, u\hat{\alpha}^2, u\hat{\alpha}^3, 1]^T(k)$$

$$\hat{\alpha}(k) = p_\alpha^T \phi(k-1) + \left(\alpha(k-1) - \alpha_{mod}(k-1)\right)$$

$$\hat{q}(k) = p_q^T \phi(k-1) + \left(q(k-1) - q_{mod}(k-1)\right)$$

The controller is assumed to know the values of angle of attack and of pitch rate at the moment k-1. Then it estimates their current values $\alpha(k)$ and $q(k)$ taking into consideration previous prediction errors and based on them it calculates the control required to achieve α_{ref} at the moment k+1. The value of control is found as:

$$u(k) = \frac{\bar{\alpha}_r - p_{1\alpha}\hat{\alpha} - p_{2\alpha}\hat{\alpha}^2 - p_{3\alpha}\hat{\alpha}^3 \; p_{4\alpha}\hat{q} - p_{5\alpha}\hat{q}\hat{\alpha} - p_{6\alpha}\hat{q}\hat{\alpha}^2 - p_{7\alpha}\hat{q}\hat{\alpha}^3 - p_{12\alpha}}{p_{8\alpha} + p_{9\alpha}\hat{\alpha} + p_{10\alpha}\hat{\alpha}^2 + p_{11\alpha}\hat{\alpha}^3} \tag{6}$$

where

$$\bar{\alpha}_r = \alpha_{ref}(k+1) - \left(\alpha(k-1) - \alpha_{mod}(k-1)\right)$$

and $\hat{\alpha} = \hat{\alpha}(k)$, $\hat{q} = \hat{q}(k)$ as described above.

The results of the simulations are seen in Figures 2a,b. The reference trajectory was chosen to be $1/z^2-1.6z+0.65)$. The actual output of the plant is seen to follow the reference very closely, even though the region of operation was that of the most severe nonlinearities. The control action is also remarkably smooth.

The discrete time nonlinear state space model (1) describes the behavior of the complex nonlinear plant quite accurately in the entire region of operation. In practice, however, such a global model is rather difficult to fit, and consequently one

should look for local approximations, depending on the current operating conditions. In such a situation, on-line adaptive control seems to offer an ideal solution.

The algorithm discussed in the previous section can be made adaptive, or self-tuning, by incorporating on-line identification of the parameters. A recursive least squares (RLS) algorithm was implemented in the following form taken from [8]:

$$p(k) = \frac{Q(k-2)\,\phi(k-1)}{\lambda(k-1) + \phi(k-1)^T Q(k-2)\,\phi(k-2)\,\phi(k-1)}\, e(k) \qquad (7)$$

$$Q(k-1) = \frac{1}{\lambda(k-1)}\left(Q(k-2) - \frac{Q(k-2)\,\phi(k-1)\,\phi(k-1)^T Q(k-2)}{\lambda(k-1) + \phi(k-1)^T Q(k-2)\,\phi(k-1)}\right) \qquad (8)$$

$$e(k-1) = y(k) - p^T\phi(k-1) \qquad (9)$$

where y may denote α or q and p may stand for p_α or p_q, respectively. The forgetting factor λ was introduced to enable the algorithm to change the estimates of parameters with the change of operating conditions. To avoid the unlimited growth of covariance matrix Q at the steady state when the input is not persistently exciting the variable forgetting factor policy was implemented:

$$\lambda(k) = 1 - e\,\frac{e(k)^2}{\bar{e}(k)^2} \qquad (10)$$

where e(k) is the current prediction error, $\bar{e}(k)$ is the average prediction error form last 10 samples and e is equal to 0.01. As an additional precaution the trace of the covariance matrix Q was monitored and Q was reset to diagonal matrix whenever the threshold value was exceeded. Starting values of parameters were taken to be as in (1).

Figure 2 displays the simulation results for a reference model specified as $1/(z^2-1.8z+0.82)$. Remarkably exact following of the reference trajectory may be observed, although, surprisingly enough, the performance is slightly worse than in the nonadaptive case. Most probably this is due to the fact that prediction error now changes much more quickly because of the ongoing identification process. Thus, approximating the term $(y(k+1)-y_{mod}(k+1))$ by $(y(k-1)-y_{mod}(k-1))$ may worsen the

behavior of the system as two values of y_{mod} no longer correspond to the same parameter vector. Since the on-line identification process assures (at least in principle) that the prediction error should asymptotically converge to zero it is possible that the correction terms in $\hat{\alpha}(k)$, $\hat{q}(k)$, and in control equation (5) ought to be omitted.

The performance of the adaptive nonlinear MAC controller was compared to the linear one, which uses the same control strategy but with a strictly linear model being identified and used for the calculation of the control action. Clear difference between the performance of linear and nonlinear controller can be seen from Figure 3, particularly in control action at the setpoint $\alpha = 15°$. The linear identifier has obvious difficulties with fitting the parameters of a linear model to the behavior of the plant which is highly nonlinear in this region. As a result, the control starts oscillating for a while. Also, it was seen that the nonlinear algorithm results in control plots that are more smooth, although they still contain one-pulse spikes. To eliminate these spikes weighting of the increments of control can be introduced into the algorithm with little performance deformation.

4. CONCLUSIONS

The nonlinear control applications to high angle-of-attack aircraft, as reported here, is of a preliminary nature. However, the analysis does suggest that nonlinear adaptive control can be quite effective to stabilize large rapid maneuvers in angle of attack. Of the comparisons made, the on-linear, nonlinear-time-series and adaptation performed the best and was quite superior to a similar linear MAC.

5. ACKNOWLEDGEMENT

REFERENCES

[1] A.D. Modyaev, A.D. Averina, "Analysis and synthesis of discrete control systems based on multidimensional z transforms," in Philosophy of Nonlinear Systems (B. Naumov, ed.), Mir Publishers/CRC Press, 1990.

[2] K. Harris, "Properties of nonlinear model algorithmic control," Proceedings of 24th Conference on Decision and Control, Ft. Lauderdale, 1985, vol. 1, pp. 663-665.

[3] H. Stalford, W.T. Baumann, F.E. Garrett, T.L. Herdman, "Accurate modeling of nonlinear systems using Volterra series submodels," Proceedings of the 1987 American Control Conference, Minneapolis, 1987, Vol. 2, pp. 886-891.

[4] A. Ostroff, "Application of variable-gain output feedback for high-alpha control," AIAA Guidance, Nav. & Control Conf., Boston, 1989.

[5] A. Ostroff, "Superagility application of a variable-gain output feedback control design methodology," NASA High Angle of Attack Tech. Conf., Hampton, VA, 1990.

[6] H. Wakamatsu, "Model reference nonlinear adaptive control system using nonlinear autoregressive moving average model derived from Volterra series and its application to control of respiration," Proceedings of IFAC 10th Triennial World Congress, Munich, 1987, Vol. 10, pp. 191-196.

[7] S.A. Al-Baiyat, "Nonlinear feedback synthesis: a Volterra approach," Ph.D. dissertation, Department of Electrical Engineering, University of Notre Dame, 1986.

[8] G.C. Goodwin, K.S. Sin, Adaptive Filtering, Prediction and Control, Prentice Hall, 1984.
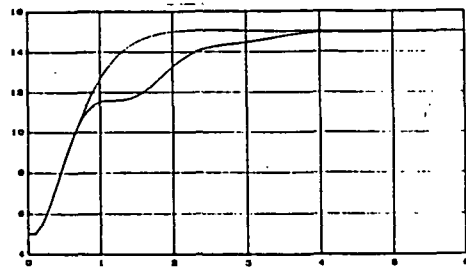
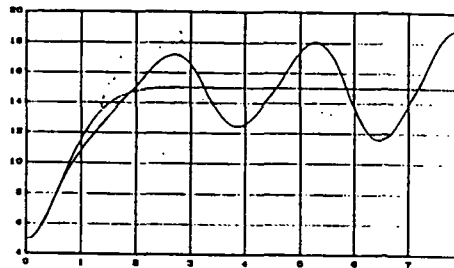Figure 1a: Step response with non-linear controller vs. nominal response

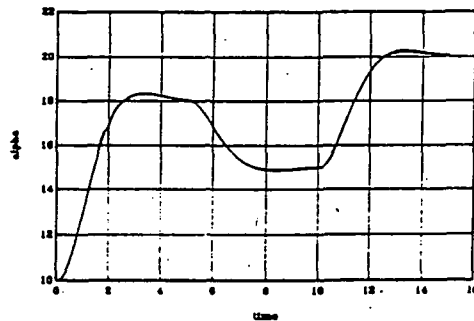Figure 1b: Step response with linear controller

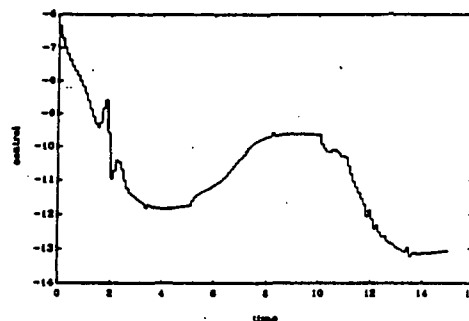Figure 2a: Nonlinear adaptive MAC (with reference trajectory)
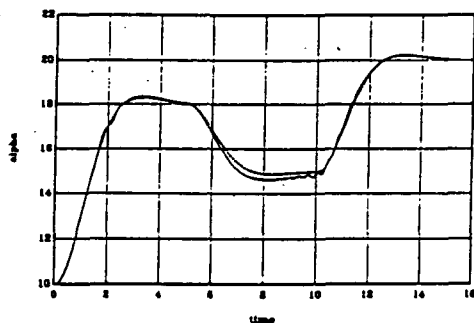

Figure 2b: Nonlinear adaptive MAC

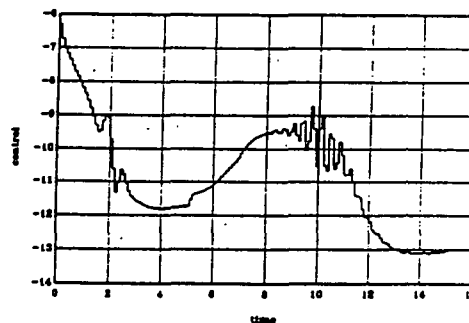Figure 3a: Linear adaptive MAC (with reference trajectory)

Figure 3b: Linear adaptive MAC

APPENDIX C

Analysis of Nonlinear System Stability Using

Robust Stability Analysis for Linear Systems

ANALYSIS OF NONLINEAR SYSTEM STABILITY USING ROBUST STABILITY ANALYSIS APPROACH FOR LINEAR SYSTEMS

Jerzy E. Kurek

Instytut Automatyki Przemyslowej, Politechnika Warszawska
(Inst. of Industrial Automatics, Warsaw University of Technology)
ul. Chodkiewicza 8, 02-525 Warszawa, Poland
Fax (22) 29-2962, Tel (22) 49-9616, Tlx 813307 pwpl

During the research author was with
Department of Electrical and Computer Engineering
Oregon State University, Corvallis, Oregon 97331-3211, U.S.A.

Summary
The stability analysis of an airplane using its nonlinear model is presented. The analysis is based on the robust stability analysis approach for linear systems. Then, based on analysis, a small static feedback gain is designed such that the robustness of the closed-loop nonlinear system stability is significantly improved.

1

1. INTRODUCTION

The stability is one of the most important issues in the cont
system design. Recently there has been observed a great inter
in the methodology of robust stability analysis and design
robust control systems for linear dynamic systems [6].
objective of this paper is to investigate the applicability
this approach for nonlinear dynamic system such as an aircr
flight in high angle of attack/sideslip flight. The unsta
control system can result in the plane crash.

There is considered stability of nonlinear, simplified howev
model of the airplane. The organization of the paper is
follows. In section 2, the model of the plane is present
Stability of the aircraft is considered in section 3. Final
concluding remarks are given.

2. THE AIRCRAFT MODEL.

Model of an airplane is highly nonlinear, [4,5]. There
usually, however, used simplified models for control sys
design, e.g. [1,3,9]. In this paper we consider very simple mo
given in [8]:

$$\dot{x}=A(x)x+Bu+D$$

where $x=\begin{bmatrix} \alpha \\ q \end{bmatrix}$ is a state vector, α is the angle of attack
degrees, q is the pitch ratio in degrees per second and u is
elevator control in degrees,

$$A=\begin{bmatrix} 9.168c_z(\alpha) & 1 \\ -5.73 & 0 \end{bmatrix}, \quad B=\begin{bmatrix} -1.8336 \\ -8.5950 \end{bmatrix}, \quad D=\begin{bmatrix} -5.473296 \\ 2.865000 \end{bmatrix}$$

and $c_z(\alpha)$ is a nonlinear function. This function can, however,
approximated as follows:

$$c_z(\alpha) = \begin{cases} -0.072815870 & \text{for} & 0° \le \alpha \le 14.74° \\ 0.088470922 - 2.3774/\alpha & \text{for} & 14.74° < \alpha \le 17.40° \\ 0.033099050 - 1.4068/\alpha & \text{for} & 17.40° < \alpha \le 18.87° \\ -0.016633734 - 0.4743/\alpha & \text{for} & 18.87° < \alpha \le 28.00° \end{cases} \quad (2)$$

It is easy to find that

$-0.072818087 < c_z(\alpha) \le -0.048161261$ for $14.74° < \alpha \le 17.40°$

$-0.047751524 < c_z(\alpha) \le -0.041453149$ for $17.40° < \alpha \le 18.87°$

$-0.041768869 < c_z(\alpha) \le -0.033573019$ for $18.87° < \alpha \le 28.00°$

This model approximates model taken from measured wind tunnel values of the T-2C airplane [7]. It is known that numerical values of $c_z(\alpha)$ and b_2 are uncertain.

Our purpose is to consider stability of system (1) in the range of angle of attack $0° \le \alpha \le 28°$, and to find a static feedback which can, eventually, improve the stability of the plane in this range.

3. STABILITY ANALYSIS.

Consider linear time-invariant system

$$\dot{x} = Ax \quad (4)$$

where $x \in R^n$ is a state vector. Then, assuming that the system is asymptotically stable one can define the following notions, [2].

Definition 1.
A connected set Ω_I in the system parameters-space (parameters of matrix A) is a robust time invariant stable (RTIS) set for system (4) iff $A \in \Omega_I$ and every time-invariant system

$$\dot{x} = \mathcal{A}x \quad (5)$$

is asymptotically stable for $\mathcal{A} \in \Omega_I$.

□

<u>Definition 2.</u>

A connected set Ω_V in the system parameters-space is a robust t‍
varying stable (RTVS) set for system (4) <u>iff</u> $A \in \Omega_V$ and every *ti‍*
varying system (5) is asymptotically stable for $A \in \Omega_V$.

Then, consider four linear models instead of (2), respectively:

$$c_z(\alpha) = \begin{cases} -0.072815870 & \text{for} & 0° \le \alpha \le 14.74° \\ -0.048161261 & \text{for} & 14.74° < \alpha \le 17.40° \\ -0.041453149 & \text{for} & 17.40° < \alpha \le 18.87° \\ -0.016633734 & \text{for} & 18.87° < \alpha \le 28.00° \end{cases}$$

It is easy to find that all models are asymptotically stable.
are, however, interested in the set of (k_1, k_2) such that all
linear closed loop systems will be stable with the follow‍
feedback

$$u = Kx, \quad K = [k_1 \ k_2]$$

An appropriate region Ω_I can be easily calculated based
algorithm 2 proposed in [2]. This is, however, only the sec‍
order system and one can simply obtain analytical formulas for
RTIS region in this case. The characteristic polynomial for
2nd order system has the following form

$$s^2 + as + b = 0$$

It is known that all roots of this polynomial are in the left h‍
plane, i.e. a system is asymptotically stable stable, <u>iff</u> a>0
b>0. Based on this, the RTIS region Ω_I for 'stable' feedback ga‍
was calculated. The region is presented on fig. 1, a dashed l‍
represents RTIS region for model P1, $0° \le \alpha \le 14.74°$, a dotted mo‍
P2 for $14.74° < \alpha \le 17.4°$, a dash-dotted model P3 for $17.40° < \alpha \le 18.8$‍
and a continuous line model P4 for $18.87° < \alpha \le 28°$. It is easy to s‍
that the system without feedback, i.e. $k_1 = k_2 = 0$, sign + on
plane (k_1, k_2), is very close to the stability region boundary.
can improve stability assuming appropriate K from $\Omega_I(k_1, k_2)$.

Next, RTVS sets Ω_V were calculated for these models, according to the algorithm given in [2], for uncertain parameters a_{11} and a_{21} in A. They are presented on fig. 2. All four models are inside the RTVS region calculated for the model P4. Moreover, since all time varying (nonlinear) $a_{11}=9.168c_z(\alpha)$ is smaller than nominal values used in linear models it means that the whole nonlinear system (1) is asymptotically stable for $0°\leq\alpha\leq28°$. However, there is a very small upper bound for a_{11} in this model, namely

$$+\Delta a_{11} < 0.0227$$

This can cause that with small system uncertainty the system can be unstable. The vertexes of RTVS quadrilateral Ω_V on the plane $(\Delta a_{11}, \Delta a_{21})$ are as follows:

$$V_{V0} = \{ \ (-176.5,0), \ (0.0226,0), \ (0,-0.2274) \ (0,0.2944) \ \}$$

In order to improve system stability feedback gain matrix K was chosen from Ω_I. Intuitively, it seems that a good gain is a small one - a high gain can result in a lack of system controllability because of saturation of the control input, and such that the stability margin with respect to K will be rather large.

Thus, the good choice seems to be $K_1=[0 \quad 0.2]$. For this gain one obtains significant improvement of RTVS set. This set is shown on fig. 3. In this case also all models are inside the Ω_{I1} calculated for the model P4. However, an upper bound for a_{11} is more than 8 times greater:

$$+\Delta a_{11} < 0.1835$$

Also range for uncertain parameter a_{21} is almost 6 times larger. Indeed, the vertexes of RTVS quadrilateral in this case on the plane $(\Delta a_{11}, \Delta a_{21})$ are as follows

$$V_{V1} = \{ \ (-25049,0), \ (0.1835,0), \ (0,-3.547), \ (0,3.211) \ \}$$

Then, it was considered feedback gain $K_2 = [0.2 \quad 0]$. This ga
however, seems to be worse situated in the RTIS set Ω_I than
considering the stability region with respect to K. Neverthele
also in this case one obtains improvement of robust stability
the closed loop system. The appropriate RTVS set Ω_V is shown
fig. 4. In this case an upper bound for a_{11} is as follows

$$+\Delta a_{11} < 0.0613$$

Similarly, range for perturbation in a_{21} is larger than for K
The vertexes of RTVS set Ω_{V2} are as follows

$$V_{V2} = \{ \ (-65.15,0), \ (0.0613,0), \ (0,-0.5770), \ (0,1.3641) \ \}$$

It should be noted that all RTVS sets were calculated un
assumption Q=I in algorithm 3 [2].

From the above analysis follows that relatively small sta
linear feedback gain $K = [0 \quad 0.2]$ significantly improves stabil
of the system. It should be emphsized that every nonlinear/ti
varying system (1) with a_{11} and a_{21} from the obtained RTVS set
will be asymptotically stable. This way we have designed a robu
stable nonlinear closed-loop system.

4. CONCLUDING REMARKS.

A robust-stable nonlinear controlsystem has been designed. It
shown that small linear static feedback gain can significan
improve stability of the airplane. The feedback gain seems to
so small that it should not constrain control signal during pl
maneuvering. This should also results in better a controllabil
of the plane.

The stability analysis and feedback gain synthesis were done us
methods designed for linear systems [2]. This, approach can
also used for more complicated nonlinear systems. For instan

6

assuming as a base model for the airplane, the linear 9th order model given in [1,9]. This model is unstable, but, as it was shown in [2], one can deal also with unstable models using the same approach.

Presented results also show the power of the approach proposed in [2].

REFERENCES.

[1] S.Grag, D.L.Mattern and R.E.Bullard, "Integrated flight/propulsion control system design on a centralized approach", J. Guidance Control and Dynamics, vol. 14, 1991.
[2] J.E.Kurek, "Robust stability region for linear systems - part I: continuous-time systems", IEEE Trans. Automat. Control, submitted.
[3] R.R.Mohler et. al., Nonlinear stability and control study of highly maneuverable high performance aircraft, OSU-ECE report NASA 91-01, Oregon State University, 1991.
[4] R.R.Mohler, Nonlinear systems v. 1: Dynamics and control, Prentice-Hall, Englewood Cliffs, 1991.
[5] A.J.Ostroff, "Application of variable-gain output feedback for high-alpha control", AIAA Guidance Navigation and Control Conf., Boston, paper No. 89-3576, 1989.
[6] D.D.Siljak, "Parameter space methods for robust control design: a guided tour", IEEE Trans. Automat. Control, vol. AC-34, pp. 674-688, 1989.
[7] H.Stalford, "Application of the estimation-before-modeling (EBM) system identification method to the high angle of attack/sideslip flight of the T-2C jet trainer aircraft", Naval Air development Center Report NADC-76097-30, vol. III, 1979.
[8] H.Stalford, W.T.Baumann, F.E.Garret, T.L.Herdman, "Accurate modeling of nonlinear systems using Volterra series submodels", Proc. 1987 American Control Conference, Minneapolis, vol. 2, pp. 886-891, 1987.
[9] T.Troudet, S.Garg, W.C.Merrill, "Neural network application to aircraft control system design", Proc. AIAA Guidance, Navigation and Control Conference, New Orleans, Loiusiana, vol. 3, pp. 993-1009, 1991.

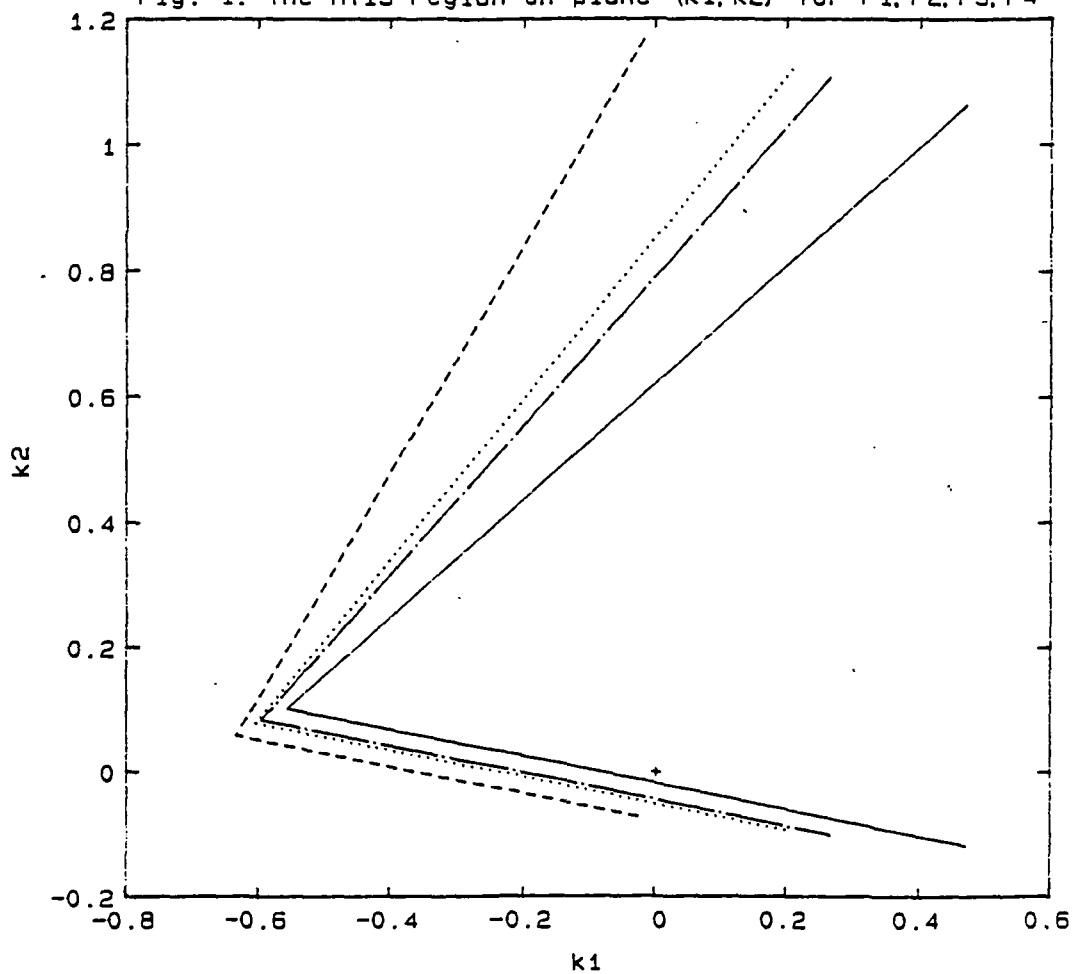Fig. 1. The ATIS region on plane (k1, k2) for P1, P2, P3, P4

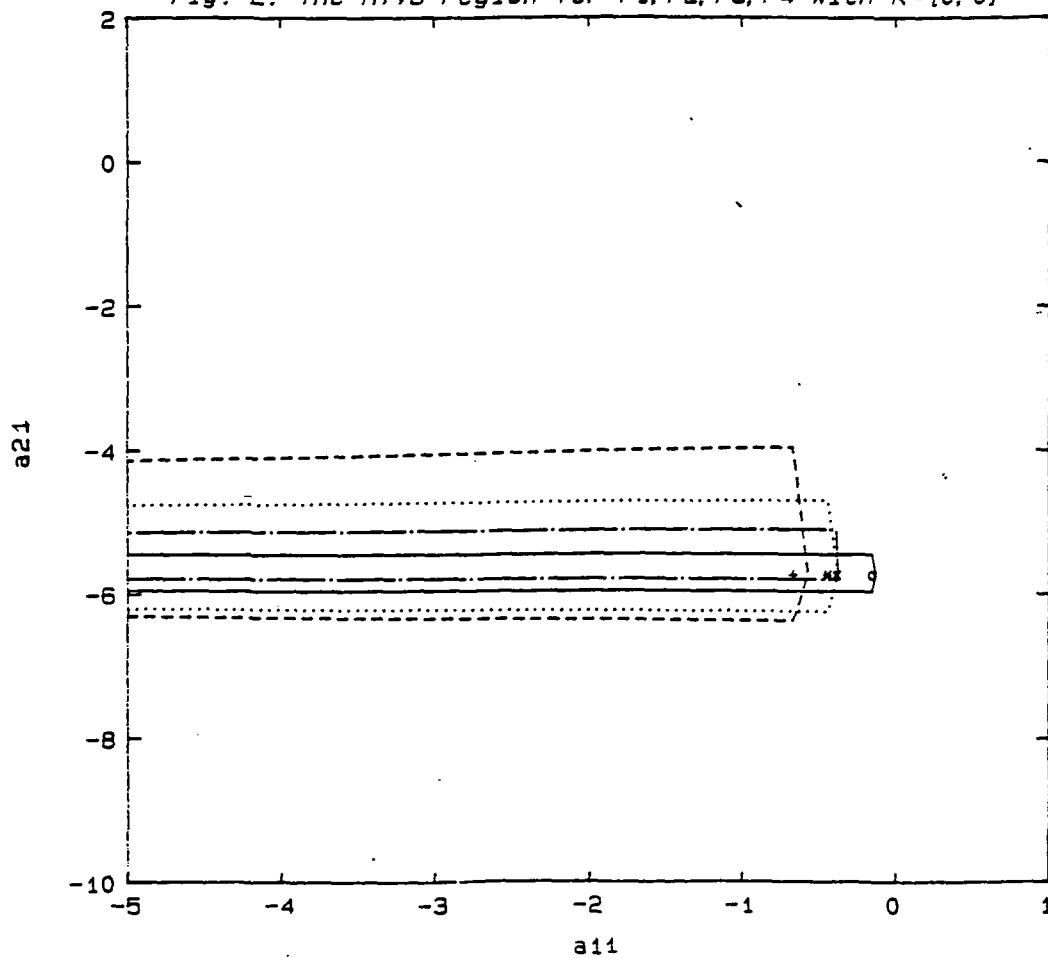Fig. 2. The ATVS region for P1, P2, P3, P4 with K=(0, 0)
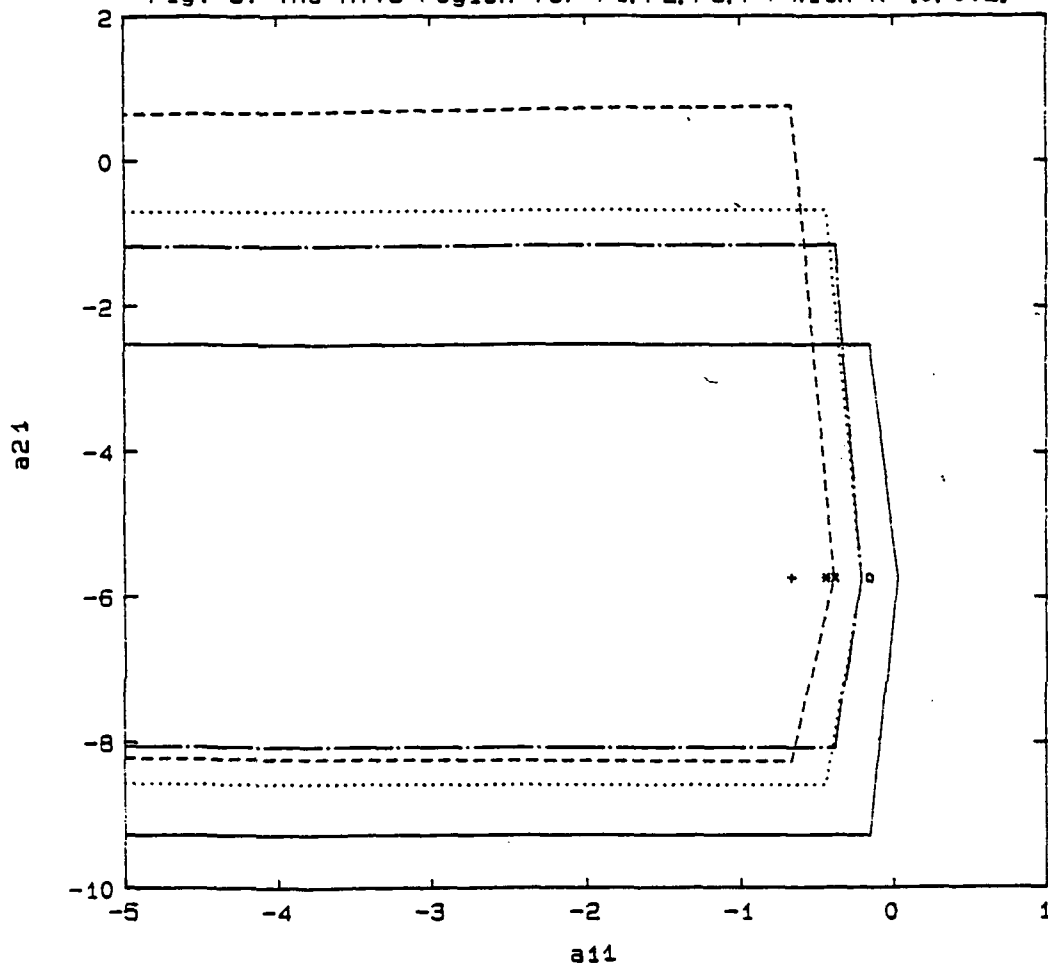
Fig. 3. The RTVS region for P1, P2, P3, P4 with K=[0, 0.2]

Fig. 4. The RTVS region for P1, P2, P3, P4 with K=[0.2, 0]