TWENTY-FIFTH EUROPEAN ROTORCRAFT FORUM

Paper no. G24

THE USE OF DISCRETE OBSERVER THEORY TO TRIM AND STABILIZE PERIODIC COEFFICIENT DYNAMIC SYSTEMS

BY

MICHAEL H. PETERS
Graduate Research Assistant

PHILIP V. BAYLY
Associate Professor

DAVID A. PETERS
McDonnell Douglas Professor
and Chairman

DEPARTMENT OF MECHANICAL ENGINEERING
WASHINGTON UNIVERSITY
ST. LOUIS, MO 63130 USA

SEPTEMBER 14-16, 1999
ROME
ITALY

ASSOCIAZIONE INDUSTRIE PER L'AEROSPAZIO, I SISTEMI E LA DIFESA
ASSOCIAZIONE ITALIANA DI AERONAUTICA ED ASTRONAUTICA

The Use of Discrete Observer Theory to Trim and Stabilize Periodic Coefficient Systems

by

Michael H. Peters
Graduate Research Assistant

Philip V. Bayly
Associate Professor

and

David A. Peters
McDonnell Douglas Professor and Chairman
Department of Mechanical Engineering
Washington University – Campus Box 1185
St. Louis, MO 63130

Abstract

The nonlinear differential equations for rotorcraft simulation (in this case, flap-lag-pitch and ground resonance equations) are sampled once a period (or once a blade passage) and the resultant sampled errors in periodicity and trim are considered a discrete system. The algebraic Riccati Equation is then used to design a controller to trim this discrete system. The resultant controller is applied to the original nonlinear simulation in that the errors at each blade passage are fed back to give discrete control changes. When all states cannot be measured, a discrete observer is used to estimate them. The resultant algorithm is shown to be a robust tool both for trimming and for stabilizing a nonlinear set of rotorcraft equations.

Notation

a	lift curve slope, /radian
[A]	matrix multiplying states in discrete system
$[A]_t$	modified matrix = $A - BG$
b	blade semi-chord, m
[B]	matrix multiplying controls in discrete system
c_d	blade profile drag coefficient
C_T	thrust coefficient, $T/\rho(\Omega R)^2 \pi R^2$
$[C_z]$	observer matrix
$[C(\psi)]$	damping matrix, period T
$d_k, \hat{d}$	desired constraint values
dpl	lag damper setting, percent critical damping
f_j	state-derivative functions
$f_\beta, f_\theta, f_\zeta$	generalized loads
$J_\beta, J_\theta, J_\zeta$	blade principal moment of inertia in flap, torsion, lag, $kg-m^2$
g_k	constraints
G_k	constraint integrand
$[J]$	Jacobian matrix
K_β	flap stiffness, N-m/rad
K_ζ	lag stiffness, N-m/rad
K_θ	pitch stiffness, N-m/rad
$[K(\psi)]$	stiffness matrix, period T
$[M]$	mass matrix
P	stiffness parameter for blade

flap $1+K_\beta/\Omega J_\beta$

Q	number of blades
R	radius of rotor, m
T	period of system
V_i	induced flow, normal to disk
V_x	inplane flow of disk
V_z	normal flow to disk
x_j	state variables
Z_n	discrete states
β	flap angle positive flap and angle positive up, rad
β_0	coning angle, rad
β_s	lateral cyclic flap angle, rad
γ	Lock number $\rho acR^4/J_\beta$
$\Delta()$	difference of ( )
θ_b	pilot pitch control $\theta_b=\theta_0+\theta_s\sin\psi+\theta\cos\psi$
θ_0	collective pitch angle
θ_s	longitudinal cyclic pitch angle
θ_c	lateral cyclic pitch angle
Θ_m	vector of controls
λ	inflow ratio, $(V_i+V_z)/\Omega R$
$[\Lambda]$	square matrix with eigenvalues on the diagonal
μ	advance ratio, $V_x/\Omega R$
ρ	density of air, kg/m^3
σ	rotor solidity, $Qc/\pi R$
ψ	rotor azimuth angle = Ωt, rad
Ω	rotor angular velocity, rad/sec
ω_ζ	dimensionless nonrotating lead-lag frequencies at $\theta=0,\sqrt{K_\zeta/\Omega^2 J_\beta}$
ω_θ	dimensionless nonrotating elastic pitch frequencies at $\theta=0,\sqrt{K_\theta/\Omega^2 J_\beta}$
$(\ )_n$	n-th iteration of vector

Introduction

The analysis of a rotary-wing system consists of three steps. First, one must assemble a set of ordinary differential equations that describes the physics of the system. Unlike modeling in many other technical disciplines, the equations so assembled are generally nonlinear, have periodic coefficients, and contain certain unknown parameters called "trim settings". It is the job of the analyst to solve these equations subject to a number of side constraints on the solution equal in number to the number of unknown parameters. Thus, the second step in a rotary-wing analysis is to find simultaneously the unknown parameters and the corresponding solution that satisfies these constraints. (This is called the "trim" solution.) The third step in the analysis is to analyze the stability of the solution about this periodic orbit. Many times, the solution strategy used to find the trim solution holds the key to the stability analysis.

There are several reasons that the trim solution is difficult. First, although there may be only a few trim variables (usually 3-7), there may be thousands of dynamic states. Thus, an iterative solution of the state-space equations for a periodic solution may be prohibitive. Second, there are often hidden states that occur due to the various mathematical algorithms that are used to evolve the system equations through time. Thus, the states may not be in the form to be interrogated numerically. Third, the periodic trim solution is often unstable (due to aeroelastic effects or due to flight mechanics modes) so that it is impossible simply to time-march until transients decay.

Recent work has found that, when all states are known and limited to a hundred or so, a very effective trim strategy is the discrete auto-pilot, Ref. [1]. Such an auto-pilot makes adjustments to the trim settings every blade passage based on an optimized gain and controller. Such a controller can fly even an unstable system to equilibrium and satisfy the trim constraints. Since a discrete auto-pilot does not change the poles of the system, it is perfect for a subsequent stability analysis, Ref. [2].

Despite the successes of such a system, it is still subject to the limitation that all states must be known and that they must be limited to about one hundred. In this paper, we offer an alternative strategy designed for the case in which states are hidden. In particular, a discrete-time observer is created that gives an estimate of the hidden states that can be used in the discrete auto-pilot. This paper will describe how such an observer can be created and will give some simple examples for helicopter flap-lag-torsion stability.

The use of observers is not new to rotorcraft analyses, Refs. [3] and [4]; but this paper uses them in the sense of discrete systems and for mathematical trim and stabilization rather than real-time vibration control.

Mathematical Background

The nonlinear differential equations for a rotorcraft following general form:

$$\dot{x}_j = f_j(x_k, t, \Theta_m) \qquad (1)$$

where the x_j are the state variables, Θ_m are the trim parameters, and *t* is time. For every unknown trim parameter, there is a trim constraint of the form:

$$g_k \equiv \int_0^T G_k(x_i, t, \Theta_m)dt = d_k \qquad (2)$$

In general, states can be comprised of positions (displacement or rotations), rates, pressures, inflows, engine temperatures or pressures, damper states, etc. Trim variable can be collective and cyclic pitch position, throttle position, prescribed fuselage attitudes, RPM, or any unknown parameter in the equations. The trim constraints can be force or moment equilibrium, time averages or space averages of vehicle positions or velocities, etc. In trimming, one usually considers the equations to be linearized about some periodic equilibrium, $\bar{x}$ during each part of the design process. Thus, for simplicity, we consider the linearization of Eq. (1) about $\bar{x}_j$,

$$\{\dot{x}_j\} = [A(t)]\{x_j\} + [b(t)]\{\Theta_m\} \qquad (3)$$

where x_j are the new states which are perturbation states; and it is understood that $A(t)$ and $b(t)$ depend on $\bar{x}_j$. Equation (3) is never actually used in our trim algorithm, but it is an important conceptual tool.

The trim problem is to find the initial conditions $x_j(0)$ and trim parameters, Θ_m, that will result in a periodic solution to either Eq. (1) or Eq. (3) that satisfies the trim constraints. One way which has been tried in the past, Ref. [1], is to design an auto-pilot for the linear system in Eq. (3) that produces a Θ_m proportional to the current errors in the trim constraints. Three disadvantages of this approach are: 1) it produces unsteady $\Theta_m(t)$ when they should be constants, 2) it is hard to find a suitable auto-pilot since this requires solution of a periodic-coefficient Riccati Equation, and 3) the method cannot work if the system is unstable. References [1] and [2] have shown that this can be overcome with a discrete auto-pilot. In particular, if the system is sampled once a period, the errors in periodicity, $x(T) - x(0)$, and the errors in the trim constraints in Eq. (2), $e_k = g_k - d_k$, can be considered as a discrete system vector, Z_n

$$Z_{n+1} = A_n Z_n + B_n \Theta_n \qquad (4)$$

where n is an iteration index, not a vector subscript. Again, Eq. (4) is not actually used in the discrete auto-pilot, but it is a conceptual step that is used to understand how to design such an auto-pilot.

Naturally, it is important to know how the discrete matrices, A_n and B_n, relate to the continuous matrices, $A(t)$ and $B(t)$, or to perturbations of the original nonlinear system in Eq. (1). In particular, we consider how A_n and B_n would be related to the Jacobian matrix that would be formed by making perturbations to an initial guess for the $x(0)$ and Θ_m.

The Z_n vector is defined as follows:

$$Z_n = \begin{Bmatrix} x_n - x_{n-1} \\ g_n - \hat{d} \end{Bmatrix} \equiv \begin{Bmatrix} \Delta x_n \\ e_n \end{Bmatrix} \qquad (5)$$

where g_n is the nth iteration on constraints, and $\hat{d}$ is the vector of desired constraint values. Next, one defines the following Jacobian matrix (J) for the linear system in Eq. (3). This Jacobian can be computed by time-marching with perturbations to x_j and Θ_m.

$$\begin{Bmatrix} x_{n+1} - x_n \\ g_{n+1} - \hat{d} \end{Bmatrix} = [J] \begin{Bmatrix} x_n \\ \Theta_n \end{Bmatrix} \qquad (6)$$

Here is exactly how A and B are computed from a given J. In general:

1) partition J according to x and Θ.

$$[J] = \begin{bmatrix} J_{xx} & J_{x\Theta} \\ J_{\Theta x} & J_{\Theta\Theta} \end{bmatrix} \qquad (7)$$

2) Rearrange the top partition of equation (6) to be in standard form

$$\begin{aligned} \{x_{n+1}\} &= [J_{xx} + I]\{x_n\} + [J_{x\Theta}]\{\Theta_n\} \\ \{x_n\} &= [J_{xx} + I]\{x_{n-1}\} + [J_{x\Theta}]\{\Theta_{n-1}\} \end{aligned} \qquad (8)$$

3) Subtract to obtain equations for errors in periodicity

$$\{\Delta x_{n+1}\} = [J_{xx} + I]\{\Delta x_n\} + [J_{x\Theta}]\{\Delta\Theta_n\} \quad (9)$$

4) Similarly, consider that the constraint errors from equation (6) are

$$\begin{aligned} \{e_{n+1}\} &\equiv \{g_{n+1} - \hat{d}\} = [J_{\Theta x}]\{x_n\} + [J_{\Theta\Theta}]\{\Theta_n\} \\ \{\Delta e_{n+1}\} &= [J_{\Theta x}]\{\Delta x_n\} + [J_{\Theta\Theta}]\{\Delta\Theta_n\} \end{aligned} \quad (10)$$

5) Recombine Eq. (9) into the form of Eqs. (4) and (5)

$$\begin{Bmatrix} \Delta x_{n+1} \\ e_{n+1} \end{Bmatrix} = \begin{bmatrix} J_{xx} & 0 \\ J_{\Theta x} & I \end{bmatrix} \begin{Bmatrix} \Delta x_n \\ e_n \end{Bmatrix} + \begin{bmatrix} J_{x\Theta} \\ J_{\Theta\Theta} \end{bmatrix} \Delta\Theta_n \quad (11)$$

It is now clear that the partitioned matrices in Eq. (11) are the A_n and B_n of the discrete time model. Thus, one can find A_n and B_n from a numerical Jacobian of the original, non-linear time domain system.

Discrete Controller

What we would now like to do is to design a controller that produces changes to the controls at the end of every period (or blade passage) based on the errors in periodicity and trim constraints (i.e., the Z_n). In general, we would like a feedback law that would give

$$\Delta\Theta_n = -GZ_n \qquad (12)$$

If one can find an appropriate *gain* matrix G, then Eq. (4) would become:

$$Z_{n+1} = (A - BG)Z_n \qquad (13)$$

where $(A - BG)$ is designated A_t.

Although we plan to use $\Delta\Theta_n$ on the original continuous nonlinear equations to perturb the controls after every simulation of blade passage, we will design the controller G based on the *discrete* form in Eq. (4). We desire a G that places all of the eigenvalues of A_t less than unity and as close to zero as possible. Such eigenvalues will drive a difference equation to equilibrium. For linear systems, such a G can be found from an algebraic Riccati equation, Ref. [6]. Once A, B, and G are found, Eq. (13) is a feedback system the states of which converge to an equilibrium; thus, the thought experiment using the linear discrete system is over.

As the simulation runs, we sample the continuous system to obtain e_n and x_n and compare the sample with the previous sample to form a Δx_n. These form Z_n by Eq. (5). Equation (12) is then used to adjust the controls. It is important to keep in mind that x_n must be completely measurable in order for Eq. (12) to be used.

We now come to the consideration of what happens if some states cannot be measured or the case in which we can measure only certain combinations of the states. This can happen for either of two reasons. Either there are too many states to measure, or some states are truly hidden due to the nature of time-marching algorithm. For example, a pure time delay in a dynamic stall model or in a drive-train model has an infinite number of states, and they are all hidden. Another example would be when a structures code is linked to a CFD or free-wake code in which states are hidden in the "black box" that is the aerodynamic algorithm for generating induced flows. The idea here is to use an observer to estimate states that cannot be measured.

If we only know some of Z_k, say,

$$Y_k = C_z Z_k \qquad (14)$$

we will make a new system in $\hat{Z}_k$ which will be an estimate of Z_k. The $\hat{Z}_k$ dynamic system should be a discrete, dynamic feedback system having a forcing function dependent on Y_k and that we can control such that $\hat{Z}_k = Z_k$ a time increases. In other words, we want to build a dynamic observer that will estimate $\hat{Z}_k$ given the measurements that are known, Y_k

$$\hat{Z}_k = A_c \hat{Z}_{k-1} + LY_k \qquad (15)$$

for some A_c and L. Thus, we have $\hat{Z}_k$ dependent on $\hat{Z}_{k-1}$ and Y_k.

Now replace Z_k in Eq. (14) above with $\hat{Z}_k$ such that all terms in Eq. (15) are $k-1$. This yields

$$\hat{Z}_k = A_c \hat{Z}_{k-t} + LC_z A_t Z_{k-1} \qquad (16)$$

Now define $E_k = Z_k - \hat{Z}_k$ by Eqs. (4) and (16). After collecting Z_{k-1} terms, and setting:

$$A_c = A_t - LC_z A_t \qquad (17)$$

we have

$$E_k = \left(A - LC_z A\right) E_{k-1} \qquad (18)$$

All that is needed is to find an L which will minimize the eigenvalues of $A - LC_z A$. This can be done by eigenvalue placing or by a Riccati solution.

Now, the $\hat{Z}$ term can be used entirely in place of the Z term which could not be measured. Specifically, we use instead of Eq. (12)

$$\Delta\Theta_k = -G\hat{Z}_k \qquad (19)$$

where $\hat{Z}_k$ is computed from Eq. (15).

For the case in which we want to stabilize the system (rather than trim it), the mathematics is identical. There are no trim constraints, and the θ_k become the controls used to stabilize. As the controller drives the solution toward a periodic solution, it will automatically stabilize it.

Mathematical Model

We are now ready to apply this theory to an actual case. In this study, we will use the rigid-blade, flap-lag-torsion equations for a rotor in forward light which implies 6 states. We assume quasi-steady aerodynamics and neglect reversed flow. For trim variables, we include collective pitch and the two cyclic pitch values. The corresponding trim constraints are the time-averaged coning minus the desired value, and the time-averaged tip-path-plane tilts. Equations of motion for this system are based on Ref. [7], but with some corrections, and appear in the Appendix of Ref. [8].

To proceed with our methodology, an initial guess on controls and initial conditions is used to time march through one period for a baseline. Next, a Jacobian of perturbations is computed during which all states are measurable. Later, we will assume that, even if all states are not measurable for use in the feedback law, that the Jacobian is still available by some other method. $[A]$ and $[B]$ are computed from the partitioned $[J]$ as in Eq. (11). To find G , we tried two kinds of software. First, we tried using software which simply placed the eigenvalues of A_r. This resulted in an ill-conditioned G because such software works by first asking the user to input the exact desired placement of the eigenvalues of A_r, $[\Lambda]$, and then iteratively (by brute force) finding a G which satisfies the characteristic equation for the given $[\Lambda]$. This iterative algorithm in MATLAB, called "PLACE", created a matrix of very large numbers (order 1,000 or higher). Such an ill-conditioned G, when used in Eq. (12) tends to give huge $\Delta\Theta_n$.

An alternative approach is to do an optimal control in which the squares of controls and states are minimized over the iterations. The weighting functions of the errors and controls are set equal to unity for our work. The fact that states are in the objective function keeps the system stable. The fact that controls are in the objective function keeps the gains from becoming too large, as they do in eigenvalue placement. The result is a Riccati equation for G. The G computed by this Riccati solver is used as the feedback law in the discrete controller. The results work well in this application.

Next, we turn to creation of the observers. L must be computed by the same type Riccati method used to compute G for Eq. (12). The MATLAB command used here is called "DLQR."

Note that a Riccati solver may require that a matrix be in the form $A-BG$. Equation (18) is not in this form, however. Instead, we have the form $A-GB$. The equivalent system is $A^T - B^T G^T$ which can be "plugged into" the Riccati solver to yield a gain matrix which is the transpose of the G needed for $A-GB$. Also note that the MATLAB command "DLQR" does not work very well in this case because it does not accept non-square A's and B's. We therefore made a copy of the source code for DLQR in a new file, named it ndlqr (for New-DLQR), and changed it to work with non-square matrices. The math is the same as long as the dimensions agree, because the Riccati equation does not care if its arguments are square. The alternative is to put the non-square matrices from Eq. (20) into large square matrices the rest of which are just zeros.

Next, the simulation is started; and, after the n^{th} simulated blade revolution, a $\Delta\Theta_n$ is computed by Eq. (12) or Eq. (19). Then $\Delta\Theta_n$ is applied to the controls. This process is repeated until the error in the trim constraint is within certain parameters at which point the current controls are said to be approximately the "trim" control setting for the given system.

The above is similar to the development in Ref. [8] except that we have improved some of the derivations to give a better observer theory.

Results and Discussions

The equations of motion for all numerical experiments are given in Refs. [8] and [9]. The system of equations used for the numerical experiments were first run with the following constant parameters.

b/R	=	0.10
c_d/a	=	0.0016
dlp	=	0.10

P	=	$(1.05)^2$
σ	=	0.1
γ	=	5.0
μ	=	0.3
ω_ζ	=	0.49
ω_θ	=	14.1
C_T	=	0.003
J_ζ/J_β	=	1.01
J_Θ/J_β	=	.01

The above parameters yield an inherently stable system which simulates a helicopter in forward flight. In other words, even if there is not a closed-loop feedback control, the states will converge to a periodic flight path, although this equilibrium will probably not fulfill the trim constraints.

We begin with a discrete controller when all states can be measured. Figure 1 gives the error in trim constraints and the control settings as a function of iteration number. The first guess is identically zero. One can see that, after about 20 iterations, the system is trimmed; and the controls have reached their final values. Although not shown, the periodicity errors have also gone to zero. (Since these errors are in terms of radians of flapping angle, the final error of .001 is only .05 degrees.) This is typical of the trim convergence of a discrete auto-pilot with unity weighting matrices. In general terms, the control system is over-damped and could be improved by other weighting factors.

Next, we consider the same system but assume that we can measure only the trim-constraint errors. No states are assumed to be measurable. Thus, the C_z matrix is assumed to be a 9x3 matrix of all zeroes except for an identity at the far right. This makes the computation of the L for the observer very straightforward. The result is indistinguishable from that of Fig. 1. In other words, the direct measurement of states is not that important to the trim auto-pilot.

In the next case, we consider a system that is unstable. We switch to a stiff-inplane rotor $\left(\omega_\zeta = 0.70\right)$ at a higher advance ratio $(\mu = 0.4)$. We remove the lag damper (dpl = 0.10) and replace it with 1% negative damping (dpl = -.01). This gives a powerful flap-lag instability. We assume that only the three trim constraints can be measured. Thus, this is a case with an observer. Figure 2 shows the errors in trim constraints and the developing control settings. Also shown on the lower half of Fig. 2 are the evolving controls when all states are measured. Although the lack of state measurements causes some initial oscillations, the ultimate convergence is the same as when all states are measured. The convergence is also about the same as it was for the stable system, Fig. 1.

These results are much better than those previously published in Ref. [8] due to an improvement in the observer equations. The observer equations here are slightly different than those in Ref. [8].

In the next set of results, we apply the control to a 4-bladed ground resonance model. Thus, these are no trim side constraints; and the control to a periodic equilibrium is actually a stabilizing control to eliminate the ground resonance instability. Equations for the system can be found in Ref. [9]. The control is assumed to be an applied roll moment accomplished by force actuators at the base. It is assumed that only 2 of the twelve states can be measured, roll angle and pitch angle. Thus, an observer is used.

Figure 3 shows the resulting roll and pitch angles as well as the necessary control torque to achieve stabilization. Note that torque is discretely changed once per revolution. Figure 4 shows the motion (with the same step input in roll) when the controller is not used One can see the powerful ground resonance instability.

Summary and Conclusions

The method of discrete control with observer theory is applied to the problem of numerical trim of a rotorcraft simulation. First, the general problem of rotorcraft trim is reformulated in a discrete-time framework. Second, we show how to do numerical perturbations of simulation code in order to find a Jacobian. Third, we show how to use the partitioned Jacobian to form the discrete-time system matrices. These can then be sent to standard Riccati tools to find the controller and the observer.

The methodology is applied to nonlinear, flap-lag-torsion equations for a rotor blade in forward flight and to ground resonance. The results shown that the above scheme can successfully trim both stable and unstable cases in the absence of all measurements. For stable cases, only the trim constraints need to be measured. For unstable cases, the method stabilizes the system.

Acknowledgment

This work was sponsored by the United States Army Research Office AASERT Grant No. DAAH04-96-1-011, Dr. Gary Anderson, Technical Monitor.

References

1. Li, Si-Hao and Peters, David A., "A Combined Periodic-Shooting, Auto-Pilot Technique for Rotorcraft Analysis," International Conference on Computational Engineering Science, Mauna Lani, Hawaii, July 30-August 5, 1995.
2. Schmitt, John M., Bayly, Philip V., and Peters, David A., "Stabilization of Periodic Flap-Lag Dynamics in Rotor Blades," AHS 2nd International Aeromechanics Specialists' Conference, Bridgeport, October 11-13, 1995.
3. Davis, Mark, "Development and Evaluation of a Generic Active Helicopter Vibration Controller," Proceedings of the 40th Annual Forum of the AHS, May 16-18, 1994.
4. Sopher, Robert and Hallock, Daniel, "Time-History Analysis for Rotorcraft Dynamics Based on a Component Approach," 2nd Decennial Specialists' Meeting on Rotorcraft Dynamics NASA Ames Research Center, Moffett Field, CA, November 1984.
5. Peters, David A., and Barwey, Dinesh, "A General Theory of Rotorcraft Trim," *Mathematical Problems in Engineering*, Vol. 2, 1996, pp. 1-34.
6. Ramirez, W. Fred, *Process Control and Identification*, Academic Press, Boston, 1994, Chapter 4.
7. Li, Si-Hao, *A Hybrid Periodic-Shooting, Auto-Pilot Method for Rotorcraft Trim Analysis*, Doctoral Dissertation, Washington university in St. Louis, May 1997.
8. Peters, Michael H. and Peters, David A., "Discrete Control Theory and Dynamic Observers Applied to Rotorcraft Stability and Trim," Proceedings of the 54th Annual Forum of the American Helicopter Society, Washington, D.C., May 20-22, 1998.
9. Powell, et. al, *A Modern Course in Aeroelasticity*, Kluwer Academic Publishers, Dordecht, The Netherlands, Third Revised Edition, 1995, pp. 400-411.

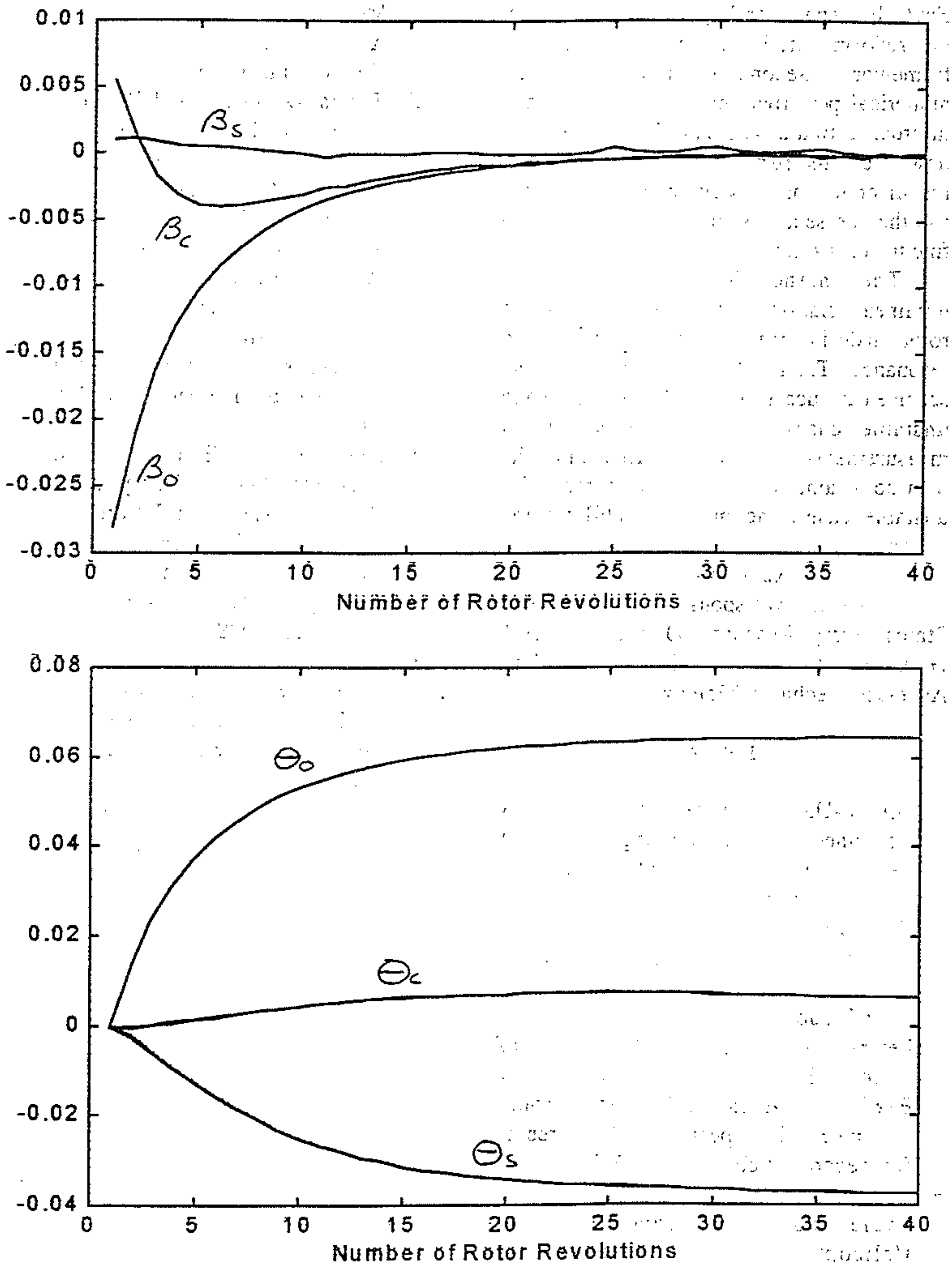

Figure 1. Trim Errors (top) and Controls (bottom) for Flap-Lag-Torsion.

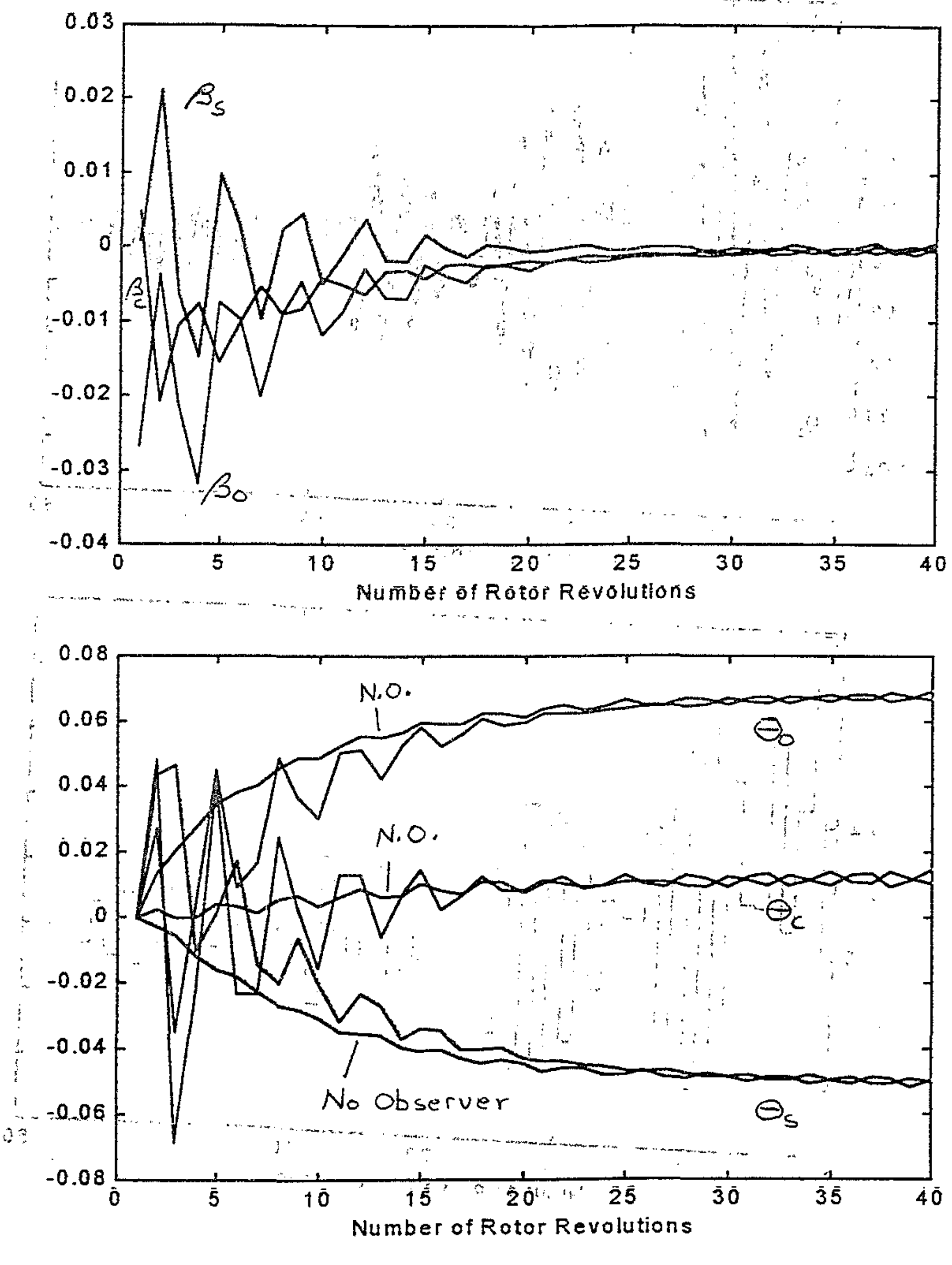

Figure 2. Trim Errors and Controls for Unstable F-L-T with Observer.
[Smoother curves on bottom are with all states measured.]

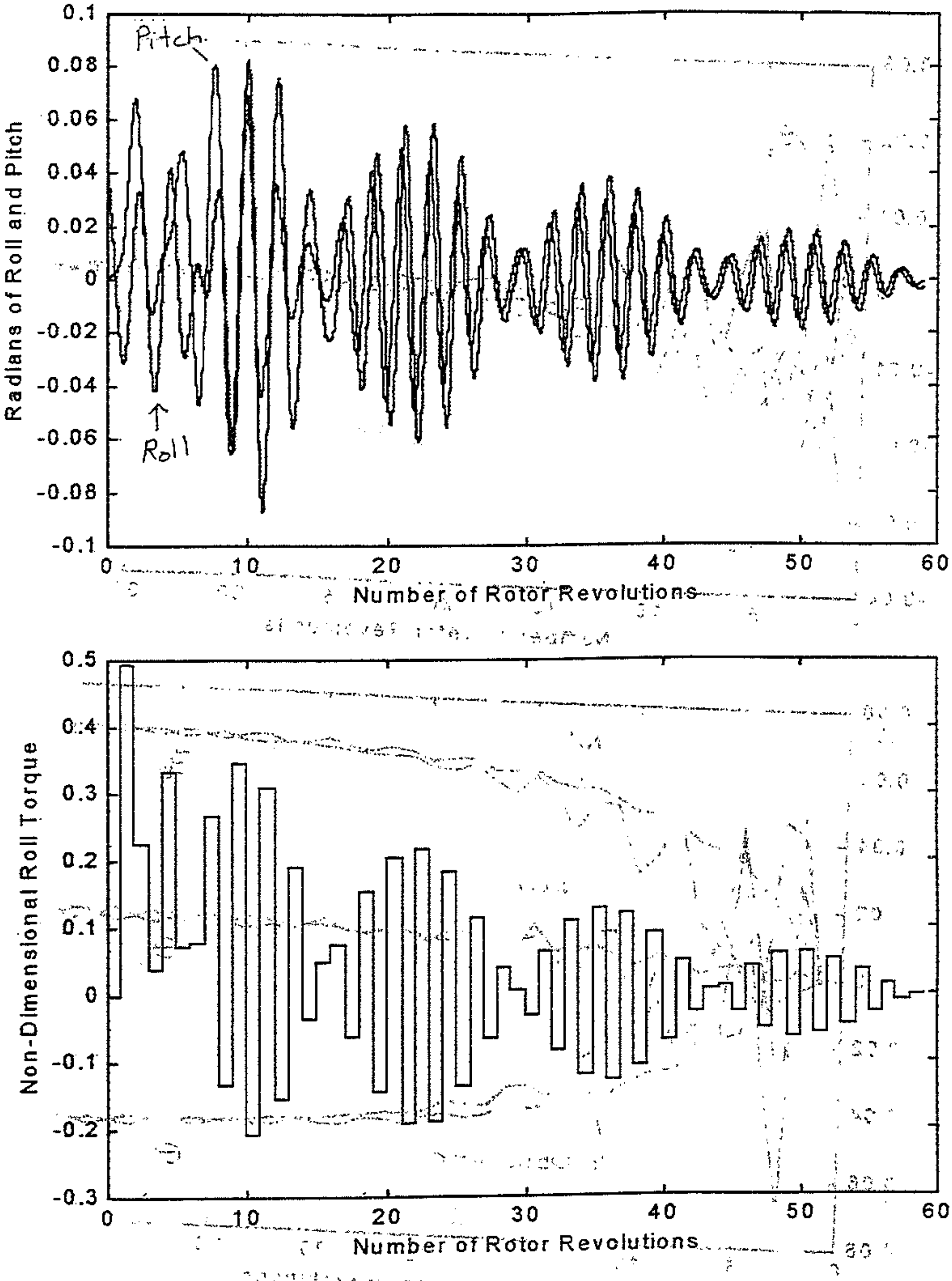

Figure 3. Roll and Pitch Angles and Control for Stabilized Ground Resonance.

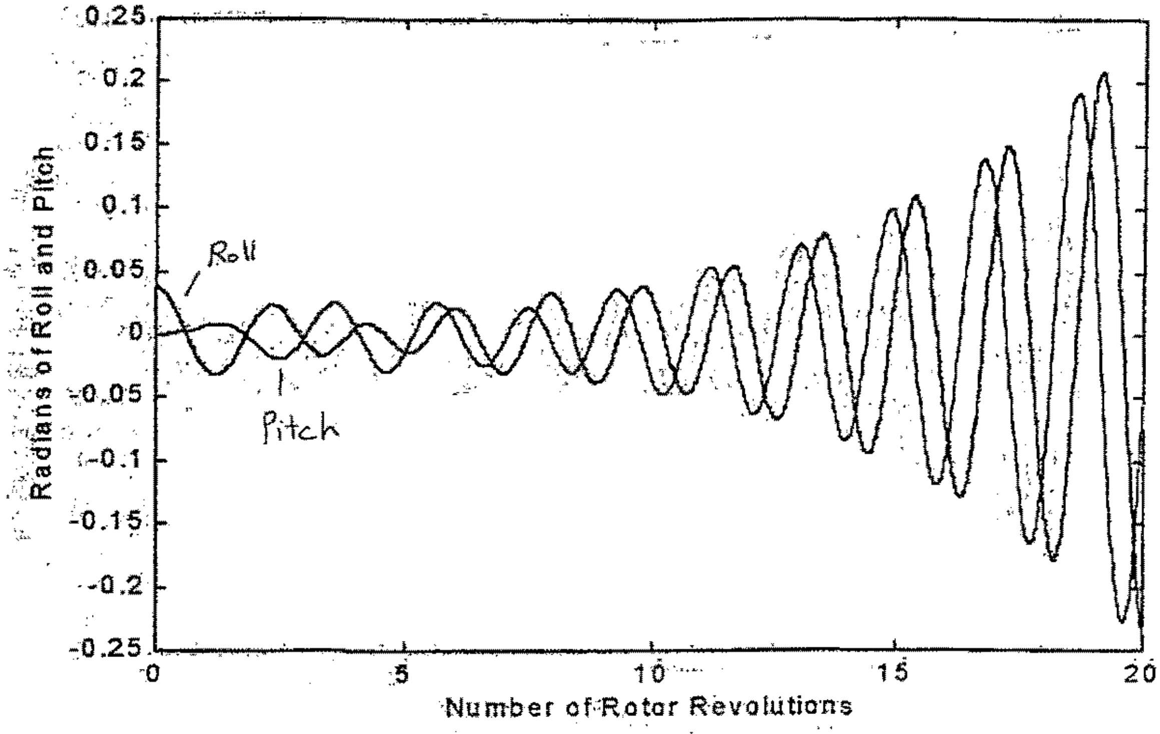

Figure 4. Roll and Pitch Angles for Ground Resonance Case with No Control.