Description of a Computer Program and Numerical Technique for Developing Linear Perturbation Models From Nonlinear Systems Simulations

James E. Dieudonne

JULY 1978
Description of a Computer Program and Numerical Technique for Developing Linear Perturbation Models From Nonlinear Systems Simulations

James E. Dicudonne
Langley Research Center
Hampton, Virginia

NASA
National Aeronautics and Space Administration
Scientific and Technical Information Office
1978
A numerical technique has been developed at Langley Research Center which generates linear perturbation models from nonlinear aircraft vehicle simulations. The technique is very general and can be applied to simulations of any system that is described by nonlinear differential equations. The computer program used to generate these models is discussed with emphasis placed on generation of the Jacobian matrices, calculation of the coefficients needed for solving the perturbation model, and generation of the solution of the linear differential equations. Included in the paper is an example application of the technique to a nonlinear model of the NASA Terminal Configured Vehicle.

INTRODUCTION

Linearized models of physical systems, when used either in conjunction with nonlinear simulations of the physical systems or in an independent mode, offer the research engineer many insights to his problem. Obviously, a nonlinear simulation will be a more valid model over a much larger range of the system's state variables, but with the present state of the art of mathematics and systems design techniques, linear models offer many advantages.

A linear model used to represent the system over some limited region has a known analytical solution which can be programmed on a digital computer and does not require standard numerical integration techniques (ref. 1). This property can result in a savings in both computation time requirements and computer memory allocations for the simulation of the system. Many available computer algorithms have been written which will identify the eigenvalues and eigenvectors of a linear model (ref. 2). This gives the researcher a quick look at the characteristic modes of the system. Once these modes have been identified, steps can be taken to eliminate undesirable characteristics by adding a feedback control system. At present, many books and articles have been written on the subject of linear feedback controls, and many computer programs are available which will provide such features as root placement and the solution to both the time-varying and steady-state optimal regulator problems (refs. 3 to 7).

This report will describe a computer program which was designed to obtain linear models about a nominal state and control vector from nonlinear real-time aircraft simulations. The program is very general in design and may be applied to any system that is described by a set of nonlinear differential equations about any trajectory in state space. The program uses various Lagrange interpolation formulas to obtain both the state and control Jacobian matrices. Once they are obtained, the linear differential equations are integrated by using the local linearization technique described in reference 1. Eigenvectors and eigenvalues are calculated using standard computer routines that are available at Langley Research Center.
<table>
<thead>
<tr>
<th>SYMBOLS</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A(t)$</td>
<td>$n \times n$ dimensional state Jacobian matrix, $A(t) = \begin{vmatrix} \frac{\partial \ddot{\bar{x}}}{\partial \bar{x}} \end{vmatrix}_{\bar{x}=\bar{x}_0, \bar{u}=\bar{u}_0}$</td>
</tr>
<tr>
<td>$a_{ij}$</td>
<td>element of $A(t)$ located at intersection of $i$th row and $j$th column</td>
</tr>
<tr>
<td>$B(t)$</td>
<td>$n \times k$ dimensional control Jacobian matrix, $B(t) = \begin{vmatrix} \frac{\partial \ddot{\bar{F}}}{\partial \bar{u}} \end{vmatrix}_{\bar{x}=\bar{x}_0, \bar{u}=\bar{u}_0}$</td>
</tr>
<tr>
<td>$\bar{c}$</td>
<td>mean aerodynamic chord, meters</td>
</tr>
<tr>
<td>$\bar{F}$</td>
<td>vector of total aerodynamic forces</td>
</tr>
<tr>
<td>$\bar{f}(\bar{x},\bar{u})$</td>
<td>$n$ dimensional vector of general, nonlinear time-varying functions of state vector $\bar{x}$ and control vector $\bar{u}$</td>
</tr>
<tr>
<td>$f_i(\cdot)$</td>
<td>$i$th component of $\bar{f}$</td>
</tr>
<tr>
<td>$g$</td>
<td>acceleration due to gravity, meters per second</td>
</tr>
<tr>
<td>$h$</td>
<td>integration interval step size, seconds</td>
</tr>
<tr>
<td>$h(t)$</td>
<td>$n$ dimensional vector whose elements are residual higher order terms from Taylor series expansion</td>
</tr>
<tr>
<td>$I$</td>
<td>$n \times n$ identity matrix</td>
</tr>
<tr>
<td>$I_{XX}, I_{YY}, I_{ZZ}, I_{XZ}$</td>
<td>moments of inertia, kilograms-meters$^2$</td>
</tr>
<tr>
<td>$J_u$</td>
<td>$n \times k$ Jacobian matrix of $\bar{f}$ with respect to $\bar{u}$</td>
</tr>
<tr>
<td>$J_x$</td>
<td>$n \times n$ Jacobian matrix of $\bar{f}$ with respect to $\bar{x}$</td>
</tr>
<tr>
<td>$k$</td>
<td>constant equal to number of elements in control vector $\bar{u}$</td>
</tr>
<tr>
<td>$L/D$</td>
<td>lift-drag ratio</td>
</tr>
<tr>
<td>$\ell_k(\cdot)$</td>
<td>coefficients used in Lagrange interpolation formulas for approximation of $f_i$</td>
</tr>
<tr>
<td>$\ell'_k(\cdot)$</td>
<td>coefficients used in Lagrange interpolation formulas for approximation of $\frac{\partial f_i}{\partial x_j}$</td>
</tr>
<tr>
<td>$\bar{M}$</td>
<td>vector of total aerodynamic moments</td>
</tr>
<tr>
<td>$M_{\text{max}}$</td>
<td>maximum operating Mach number</td>
</tr>
</tbody>
</table>
constant equal to number of elements in state vector \( x \)

\( P \)  
\( n \times n \) dimensional matrix which is a Pade' approximation to \( A^{-1}(e^{Ah} - 1)B \)

\( P_{sec} \)  
period of characteristic mode, seconds

\( Pb \)  
roll rate, radians per second

\( Q \)  
n \( \times k \) matrix which is a Pade' approximation to \( A^{-2}(e^{Ah} - Ah - 1)B \)

\( Qb \)  
pitch rate, radians per second

\( rb \)  
yaw rate, radians per second

\( S \)  
similarity transformation matrix

\( T \)  
unit thrust, newtons (NGG in computer-generated tables)

\( t \)  
time, an independent variable, seconds

\( tf \)  
final time, seconds

\( t_i \)  
starting time, seconds

\( t_1/2 \)  
time to damp to one-half amplitude, seconds

\( t_2 \)  
time to double amplitude, seconds

\( \dot{u}_b \)  
longitudinal translation velocity, meters per second

\( u(t) \)  
k-dimensional vector whose elements are control variables of system

\( \dot{v}_c \)  
calibrated airspeed, knots

\( \dot{V}_{max} \)  
maximum operating airspeed, knots

\( \dot{v}_b \)  
lateral translation velocity, meters per second

\( \dot{w}_b \)  
vertical translation velocity, meters per second

\( x(t) \)  
n-dimensional vector whose elements are state variables of system

\( \dot{x}_j \)  
nominal state vector with \( j \)th element possibly different from its nominal value

\( \delta \)  
small perturbation of variable away from its nominal value (PI in computer-generated tables)

\( \dot{\alpha} \)  
sailor position, degrees

\( \dot{\delta}_e \)  
elevator position, degrees
\[ \delta_j \text{ constant which is amount jth element of nominal state or control vector was perturbed} \]

\[ \delta_r \text{ rudder position, degrees} \]

\[ \delta_s \text{ stabilator position, degrees} \]

\[ \delta_{sp,L} \text{ flight spoiler position, left side, degrees} \]

\[ \delta_{sp,R} \text{ flight spoiler position, right side, degrees} \]

\[ \theta \text{ pitch attitude, radians (\text{THETA} in computer-generated tables)} \]

\[ \xi_{DR} \text{ damping coefficient for Dutch roll mode of aircraft} \]

\[ \xi_P \text{ damping coefficient for phugoid mode of aircraft} \]

\[ \xi_{SP} \text{ damping coefficient for short period mode of aircraft} \]

\[ \tau \text{ variable of integration} \]

\[ T_{RS} \text{ time constant of roll subsidence mode} \]

\[ T_{SD} \text{ time constant of spiral divergence mode} \]

\[ \phi \text{ roll attitude, radians (\text{PHI} in computer-generated tables)} \]

\[ \psi \text{ yaw attitude, radians (\text{PSI} in computer-generated tables)} \]

Subscripts:

\[ o \text{ nominal values of variables} \]

\[ R \text{ rotor} \]

A dot over a variable indicates a time (\text{DOT} in computer-generated tables).

**PROBLEM DESCRIPTION**

Aircraft simulated on Langley's real-time simulation system are described by a set of nonlinear simultaneous differential equations of the form

\[ \dot{x}(t) = \vec{f}[\vec{x}(t),\vec{u}(t),t] \]  

(1)

where \( t \) represents time, \( \dot{x}(t) \) is an \( n \) dimensional time-varying state vector, \( \vec{u}(t) \) is a \( k \) dimensional time-varying control vector, and \( \vec{f} \) is an \( n \) dimensional vector of general nonlinear functions. As shown by reference 3, if \( \vec{u}_0(t) \) is a given input (control) to the system described by equation (1) and \( \vec{x}_0(t) \) is a known solution of the state differential equation, one can find approximations to neighboring solutions for small deviations from the initial
state and input vectors by using a linear state differential equation. Assume that \( \tilde{x}_0(t) \) satisfies
\[
\dot{\tilde{x}}_0(t) = \tilde{f}[\tilde{x}_0(t), u_0(t), t] \quad (t_i \leq t \leq t_f)
\]
and that the system is operated close to nominal conditions. Therefore, one can write
\[
\begin{align*}
\tilde{u}(t) &= \tilde{u}_0(t) + \delta \tilde{u}(t) \\
\tilde{x}(t) &= \tilde{x}_0(t) + \delta \tilde{x}(t)
\end{align*}
\]
(2)

Substituting equations (2) into the state differential equation (eq. (1)) and expanding in a Taylor series about \((\tilde{x}_0(t), \tilde{u}_0(t)) \) yields
\[
\dot{\tilde{x}}_0(t) + \delta \dot{x}(t) = \tilde{f}[\tilde{x}_0(t), \tilde{u}_0(t), t] + J_x[\tilde{x}_0(t), u_0(t), t] \delta \dot{x}(t) \\
+ J_u[\tilde{x}_0(t), \tilde{u}_0(t), t] \delta \tilde{u}(t) + \tilde{h}(t)
\]
(3)

where \( J_x \) and \( J_u \) are the Jacobian matrices of \( \tilde{f} \) with respect to \( \tilde{x} \) and \( \tilde{u} \), respectively. They are given by
\[
J_x = A = \begin{bmatrix} \frac{\partial \tilde{f}}{\partial \tilde{x}} \\ \frac{\partial \tilde{f}}{\partial \tilde{u}} \end{bmatrix} \quad \text{and} \quad J_u = B = \begin{bmatrix} \frac{\partial \tilde{f}}{\partial \tilde{x}} \\ \frac{\partial \tilde{f}}{\partial \tilde{u}} \end{bmatrix}
\]

The term \( \tilde{h}(t) \) is the sum of higher order terms from the Taylor expansion and should be "small" with respect to \( \delta \tilde{x} \) and \( \delta \tilde{u} \). Neglecting \( \tilde{h}(t) \), \( \delta \tilde{x} \) and \( \delta \tilde{u} \) approximately satisfy the "linear" equation
\[
\delta \dot{x}(t) = A(t) \delta \tilde{x}(t) + B(t) \delta \tilde{u}(t)
\]
(4)

which is called the linearized state equation. For the particular applications of interest, only the time invariant system is considered in which \( A(t) \) and \( B(t) \) are constant matrices and \( A \) and \( B \). Therefore, the linearized state equation can be written as
\[
\delta \dot{x} = A \delta x + B \delta u
\]
(5)

**NUMERICAL LINEARIZATION TECHNIQUE**

Computation of the \( A \) and \( B \) Matrices

Now by using equation (1) and by assuming that each component \( f_i(\tilde{x}, \tilde{u}, t) \) for \( i = 1, n \) is continuously differentiable \( m \) times and can be evaluated \( m \) times, the partials required for the \( A \) and \( B \) matrices can be approximated by using the Lagrange interpolation formulas (refs. 8 and 9). The components of \( \tilde{f}(\tilde{x}, \tilde{u}, t) \) are approximated by
Due to notation complexity, this formula is explained by an example that uses the three-point Lagrange formula. First, 
\[ \bar{x}_k^j = (x_{o1}, x_{o2}, \ldots, x_j, \ldots, x_{on}) \]
which is the nominal state vector with the jth element allowed to vary from its nominal value while all the elements remain fixed. For the three-point formula \((m = 3)\), these vectors are
\[ \bar{x}_1^j = (x_{o1}, x_{o2}, \ldots, x_{o;j} - \delta_j, \ldots, x_{on}) \]
\[ \bar{x}_2^j = (x_{o1}, x_{o2}, \ldots, x_{o;j}, \ldots, x_{on}) \]
\[ \bar{x}_3^j = (x_{o1}, x_{o2}, \ldots, x_{o;j} + \delta_j, \ldots, x_{on}) \]
and
\[ \ell_1(x_j) = \frac{(x_j - x_{o;j})[x_j - (x_{o;j} + \delta_j)]}{[x_{o;j} - \delta_j - x_{o;j}][x_{o;j} - \delta_j] - (x_{o;j} + \delta_j)} \]
\[ = \frac{(x_j - x_{o;j})(x_j - x_{o;j} - \delta_j)}{2\delta_j^2} \]
\[ \ell_2(x_j) = \frac{[x_j - (x_{o;j} - \delta_j)][x_j - (x_{o;j} + \delta_j)]}{[x_{o;j} - (x_{o;j} - \delta_j)][x_{o;j} - (x_{o;j} + \delta_j)]} \]
\[ = \frac{(x_j - x_{o;j} + \delta_j)(x_j - x_{o;j} - \delta_j)}{-\delta_j^2} \]
In order to compute \( \frac{\partial f_1}{\partial x_j} \) equation (6) must be differentiated with respect to \( x_j \) and the resulting equation evaluated at \((\bar{x}_o, \bar{u}_o)\) as follows:

\[
\ln_j(x_j) = \frac{\left[ x_j - (x_{o_j} - \delta_j) \right] (x_j - x_{o_j})}{\left[ (x_{o_j} + \delta_j) - (x_{o_j} - \delta_j) \right] \left[ (x_{o_j} + \delta_j) - x_{o_j} \right]}
\]

\[
= \frac{(x_j - x_{o_j}) (x_j - x_{o_j})}{2\delta_j^2}
\]

In order to compute \( \frac{\partial f_1}{\partial x_j} \), equation (6) must be differentiated with respect to \( x_j \) and the resulting equation evaluated at \((\bar{x}_o, \bar{u}_o)\) as follows:

\[
a_{1j} = \frac{\partial f_1}{\partial x_j} \bigg|_{\bar{x}=\bar{x}_o \atop \bar{u}=\bar{u}_o} = \sum_{k=1}^{m} \ln'_k(x_j) f'_i(\bar{x}_k, \bar{u}_o) \tag{7}
\]

Again using the three-point formula as an example yields

\[
\ln_1(x_j) = \frac{2(x_j - x_{o_j}) - \delta_j}{2\delta_j^2}
\]

\[
\ln_2(x_j) = \frac{-2(x_j - x_{o_j})}{\delta_j^2}
\]

\[
\ln_3(x_j) = \frac{2(x_j - x_{o_j}) + \delta_j}{2\delta_j^2}
\]

which, when evaluated at the three values of \( x_j \) and summed according to equation (7), results in

\[
\frac{\partial f_1}{\partial x_j} \bigg|_{\bar{x}=\bar{x}_o \atop \bar{u}=\bar{u}_o} = \frac{a_{11}}{2\delta_j} \left[ f'_i(\bar{x}_1, \bar{u}_o) + f'_i(\bar{x}_3, \bar{u}_o) \right] \tag{8}
\]
An equivalent result for the five-point differentiation formula is

\[
\frac{3f_i}{\delta x_j} = \frac{1}{12\delta_j} \left[ f_i(\bar{x}_1^j, \bar{u}_0) - 8f_i(\bar{x}_2^j, \bar{u}_0) + 8f_i(\bar{x}_4^j, \bar{u}_0) - f_i(\bar{x}_5^j, \bar{u}_0) \right]
\]

(9)

and that for the seven-point differentiation formula is

\[
\frac{\delta f_i}{\delta x_j} = \frac{1}{30\delta_j} \left[ -f_i(\bar{x}_1^j, \bar{u}_0) + 9f_i(\bar{x}_2^j, \bar{u}_0) - 45f_i(\bar{x}_3^j, \bar{u}_0) + 45f_i(\bar{x}_5^j, \bar{u}_0) - 9f_i(\bar{x}_6^j, \bar{u}_0) + f_i(\bar{x}_7^j, \bar{u}_0) \right]
\]

(10)

The computation of the B matrix is identical to that of the A matrix except that \( \bar{x} \) is held constant and \( \bar{u} \) is varied.

Use of the Perturbation Model

Once the A and B matrices have been determined, the perturbation model defined by equation (5) is ready for use by the researcher. The two most common uses of this model are to use it in place of a nonlinear simulation in studies that will be very limited in their area of operation and to determine the eigenvalues of the model about the defined state trajectory. These eigenvalues in aerodynamic problems identify the basic modes of the aircraft; these are Dutch roll, short period, phugoid, spiral divergence, and roll subsidence. In this program, the eigenvalues of the perturbation model (characteristic roots of the A matrix) are determined by a standard Langley library routine.

If the researcher desires to use the perturbation model in place of, or to compare with, his nonlinear model, it will be necessary to integrate equation (5). The solution to equation (5) is

\[
\delta \bar{x}(t) = e^{At} \delta \bar{x}(0) + \int_0^t e^{A(t-\tau)} B \delta \bar{u}(\tau) \, d\tau
\]

(11)

where the solution to the nonlinear system would be approximated by

\[
\bar{x}(t) = \bar{x}(0) + \delta \bar{x}(t)
\]

As shown by reference 1, a discrete approximation to equation (11) with local truncation error good to \( O(h^3) \) is given by

\[
\delta \bar{x}_{k+1} = e^{Ah} \delta \bar{x}_k + p \delta \bar{u}_k + Q \delta \bar{u}_k
\]

(12)
where
\[
P = A^{-1}(e^{Ah} - I)B
\] (13)

\[
Q = A^{-2}(e^{Ah} - Ah - I)B
\] (14)

\[
\delta u_k = \frac{\delta u - \delta u_{k-1}}{h}
\] (15)

and as before
\[
x_{k+1} = x_k + \delta x_{k+1}
\] (16)

Equations (12) to (16) are solved by the program's integration subroutine.

**PROGRA USAGE AND LIMITATIONS**

For normal application the following should be followed by the user (fig. 1):

1. Trim the nonlinear aircraft model about the desired trajectory to obtain the nominal state vector \(x_0\) and the nominal control vector \(u_0\). The trim algorithm used at Langley Research Center for most real-time simulations is described in reference 10.

2. Compute the \(A\) matrix by using subroutine JACMAT (appendix A) with \(XNOM = \bar{x}_0\) and \(M = N = n\) (number of states).

3. Reset the states to their trim values.

4. Compute the \(B\) matrix by using JACMAT with \(XNOM = u_0\), \(M = k\) (number of controls), and \(N = n\).

5. Reset controls to their trim values.

6. If eigenvalues are required, the user must call a subroutine which generates eigenvalues. For the applications presented, subroutine REQR, a part of the Langley computer mathematical library, was used.

7. If integration of the linear system is required, call subroutine COEFF (appendix A) with \(NDIMA = n\) and \(NCOLE = k\) for calculating the coefficients of \(\delta x_k\), \(\delta u_k\), and \(\delta u_k\).

8. Obtain response of the linear system to a predetermined input sequence \(u_k\) by calling subroutine INTEGRRT (appendix A).
When applying this technique to general nonlinear simulations, certain potential problem areas should be mentioned. First, all implicit loops in the nonlinear equations must be broken by substituting variables and by reformulating the equations. If this is not possible, an iterative technique may possibly be used to determine the approximate perturbed steady-state forces and moments. Second, the magnitudes of the perturbations used for the state and control variables need be chosen with care because if the perturbations are too small or too large, the derived linear model will not be a good approximation to the nonlinear analysis. The method used to choose perturbation magnitudes for the NASA Terminal Configured Vehicle (TCV) example is described in appendix B. And third, even though the linearization technique can be applied about any nominal state trajectory, the results are more meaningful when the vehicle is trimmed and the nominal trajectory is stable.

PROGRAM APPLICATION

As an example of the results obtained from a standard application, the TCV airplane, a Boeing 737-100, was chosen. The desired outputs of this application were (1) linearized models of the B-737 about various trim conditions, (2) identification of the basic modes of the aircraft (eigenvalues) at these trim conditions, and (3) time-history comparisons of the linear and nonlinear models for predetermined inputs.

The desired linearized models were of the form

$$\delta \ddot{x} = A\delta \dot{x} + B\delta u$$

where the state vector was chosen to be in the body-axis system. The elements of the body-axis state vector $\bar{x}_b$ are

$$\bar{x}_b = \begin{bmatrix} u_b \\ w_b \\ q_b \\ \theta \\ v_b \\ P_b \\ r_b \\ \phi \\ \psi \end{bmatrix}$$
and the elements of the control vector are

\[
\mathbf{\delta} = \begin{bmatrix}
T \\
\delta_s \\
\delta_r \\
\delta_e \\
\delta_a \\
\delta_{sp,L} \\
\delta_{sp,R}
\end{bmatrix}
\]

Tables I to V are example outputs and show the nominal state and control vectors, the body-axis A and B matrices, the eigenvalues, and the corresponding eigenvectors.

Linear models defined in other axis systems can be derived from this model by means of a similarity transformation \( S \) where

\[
\mathbf{\dot{x}}_b = S\mathbf{\dot{x}}_D
\]

and

\[
\mathbf{\dot{x}}_b = S\mathbf{\dot{x}}_D
\]

with \( \mathbf{\dot{x}}_D \) being the desired state vector defined in the new axis system and \( S \) being time invariant. Substituting into our linear model

\[
\delta \mathbf{\ddot{x}}_L = A\delta \mathbf{x}_b + B\delta \mathbf{u}
\]

yields

\[
S\delta \mathbf{\ddot{x}}_D = AS\delta \mathbf{x}_D + B\delta \mathbf{u}
\]

or

\[
\delta \mathbf{\ddot{x}}_D = S^{-1}AS\delta \mathbf{x}_D + S^{-1}B\delta \mathbf{u}
\]

as our linear model in the desired axis system.

To further show the usefulness of these linear models as a simulation verification and validation tool for the various flight conditions shown in table VI, a comparison of independent Boeing data (unpublished) and the linear models generated from the nonlinear simulation is shown in tables VII and VIII. A review of these tables will show that good agreement exists between the simulation models and the independent data in almost all cases, excluding the spiral divergence mode. However, major disagreements do exist in the short period mode of condition V with the aft center of gravity and in the phugoid mode of condi-
Additional insights into the system being simulated can be gained by comparing the linear models generated by each Lagrange interpolation formula. A general indication of the linearity of the simulation about the nominal trajectory is obtained, as well as an indication of sensitive modes and parameters (nonlinearities) of the simulation. For example, a comparison of the models obtained for the maximum speed case (table VI, condition V) showed that the lateral and the short period modes were approximately linear, but for the phugoid mode, the damping ratio varied by 36 percent and the natural frequency by 3 percent. This information implies that the linear models obtained would not be suitable for studies requiring precise knowledge of the phugoid mode.

The numerical linearization technique has also been successfully applied to nonlinear simulations of other aircraft. Linear models of a fighter aircraft, a general aviation aircraft, a standard rotorcraft, and the rotor systems research aircraft (RSRA) developed by NASA and the U.S. Army (ref. 11) have been obtained about various trim conditions. The standard procedure as described was used in all cases except that of the RSRA aircraft which required procedural modifications since the nonlinear simulation included a dynamic rotor model which was continuously rotating during the linearization. Figure 2 outlines the iterative technique used to obtain the linear models for this vehicle. Basically, the approach taken was to allow integration of the rotor dynamics. However, a steady-state condition had to be obtained after each perturbation of a state or control before numerically calculating the Jacobians. Averaging the forces and moments over a number of rotor revolutions and at various points during each revolution is also done to enhance the credibility of the linear model obtained.

CONCLUDING REMARKS

The numerical linearization technique described in this paper has been successfully applied to nonlinear simulations of various aircraft. At this writing, linear models of the NASA Terminal Configured Vehicle, a fighter aircraft, a general aviation aircraft, a standard rotorcraft, and the RSRA have been obtained about various trim conditions. Linear models of aircraft with stability augmentation systems have also been obtained by augmenting the state vector with the associated automatic control system states and by proceeding in the manner described in the paper.

A modification of the technique for application to simulations of rotorcraft with dynamic rotor models has also been developed and described.

Langley Research Center
National Aeronautics and Space Administration
Hampton, VA 23665
May 18, 1978
APPENDIX A

DESCRIPTION AND LISTINGS OF SUBROUTINES JACMAT, COEFF, AND INTEGR

The major portion of the linear analysis package consists of three subroutines, JACMAT, COEFF, and INTEGR.

Subroutine JACMAT

Purpose: Calculates the Jacobian of the nonlinear system about a nominal point in vector space. This corresponds to the A matrix when the input argument is the state vector and the B matrix when the input argument is the control vector.

Use: CALL JACMAT (XNOM, F, FVAL, JACB, DELTA, N, M, EOM, IPNTS, MAXROW, MAXCOL) where

XNOM An N dimensional input vector; this vector contains the nominal values of the independent variables about which the Jacobian is calculated

F An N dimensional output vector; during computation of the partial derivatives, it contains the values of the dependent variables

FVAL An N X M X IPNTS dimensional array used in calculating the partial derivatives; FVAL(I,J,K) is the kth component of F evaluated at the Kth change in the Jth component of XNOM

JACB An N X M dimensional output array which is the Jacobian matrix evaluated at XNOM; that is,

\[
\text{JACB}(i,j) = \left( \frac{\partial f_j}{\partial x_i} \right)_{x=XNOM}
\]

DELTA An N dimensional input vector to step sizes; DELTA(I) is the increment for XNOM(I)

N An integer input specifying the number of equations

M An integer input specifying the number of independent variables

EOM A user-supplied subroutine which calculates the values of F used in computing FVAL; EOM is a subroutine in the parameter list of JACMAT; the statement

EXTERNAL EOM

must be included in the calling program of JACMAT; the calling statement for EOM is
APPENDIX A

CALL ECM (N,M,XNOM,F)

where N, M, and XNOM are inputs and F is the output.

IPNTS  An integer input which specifies the interpolation formula to be used:
       IPNTS = 2, three-point formula
       IPNTS = 4, five-point formula
       IPNTS = 6, seven-point formula

MAXROW  An integer input specifying the maximum number of equations to be used.

MAXCOL An integer input specifying the maximum number of independent variables to be used.

The listing of subroutine JACMAT is as follows.
SUBROUTINE JACMAT(XNOM,F,FVAL,JACORN,DELTA,N,M,EOM,IPNTS,MAXROW,MAXCOL)

DIMENSION DELTA(MAXCOL),F(MAXROW),FVAL(MAXROW,MAXCOL,IPNTS),R(6)

S(6),XNOM(MAXCOL)
REAL JACORN(MAXROW,MAXCOL)
DATA R,5/12*0.1/

THIS SUBROUTINE USES AN (IPNTS+1)-POINT FORMULA EVALUATED AT
THE CENTRAL POINT TO APPROXIMATE THE PARTIALS. THE PARAMETER,
IPNTS, MAY TAKE ON THE VALUES 2, 4, OR 6.
DELTA IS AN ARRAY OF STEP SIZES WHICH ARE USED IN COMPUTING
THE PARTIALS.
JACORN IS THE JACOBIAN MATRIX.
XNOM IS A VECTOR OF INDEPENDENT VARIABLES AND IS THE POINT
AT WHICH THE PARTIALS ARE CALCULATED.
N IS THE NUMBER OF EQUATIONS.
M IS THE NUMBER OF unknowns.
F IS AN ARRAY USED TO STORE CALCULATED VALUES OF THE FUNCTIONS
FVAL IS AN ARRAY USED IN COMPUTING THE PARTIALS FOR JACOB.

IF((IPNTS.LT.1).OR.(IPNTS.GT.6)) GO TO 70
GO TO (70,2,70,4,70,6),IPNTS

2 CONTINUE

CONSTANTS USED IN THE 3-POINT FORMULA

DIV = 2
R(1) = -1
R(2) = 2
S(1) = -1
S(2) = 1

GO TO 9
4 CONTINUE
CO STANTS USED IN THE 5-POINT FORMULA

\[ \text{DIV} = 12. \]
\[ R(1) = 2. \]
\[ R(2) = R(4) = 1. \]
\[ R(3) = 2. \]
\[ S(1) = 1. \]
\[ S(2) = -R. \]
\[ S(3) = R. \]
\[ S(4) = -1. \]

GO TO 9
6 CONTINUE

CONSTANTS USED IN THE 7-POINT FORMULA

\[ \text{DIV} = 40. \]
\[ R(1) = 3. \]
\[ R(2) = R(3) = R(5) = R(6) = 1. \]
\[ R(4) = 2. \]
\[ S(1) = -1. \]
\[ S(2) = 9. \]
\[ S(3) = -45. \]
\[ S(4) = 45. \]
\[ S(5) = -9. \]
\[ S(6) = 1. \]

9 CONTINUE

HERE THE ARRAY FVAL IS COMPUTED. FVAL(I,J,K) IS THE ITH FUNCTION EVALUATED AT THE KTH CHANGE IN THE JTH VARIABLE.

DO 30 J=1,N
   XNMSAV = XNOM(J)
DO 20 K=1,IPNTS
   XNOM(J) = XNOM(J) + R(K)*DELTA(J)
   CALL EOM(N,N,XNOM,F)
DO 10 I=1,N
   FVAL(I,J,K) = F(I)/(DIV*DELTA(J))
10 CONTINUE
APPENDIX A

20 CONTINUE
30 CONTINUE

AT THIS POINT THE ARRAY JACORN IS COMPUTED.

DO 60 I = 1, N
DO 50 J = 1, M
DO 40 K = 1, IPNTS

JACORN(I, J) = JACORN(I, J) + S(K) * FVAL(I, J, K)

40 CONTINUE
50 CONTINUE
60 CONTINUE
70 RETURN
END
APPENDIX A

Subroutine COEFF

Purpose: Computes the coefficients (e^{Ah}, P, and Q) required for calculation of the discrete approximation to the solution of perturbation model.

Use: CALL COEFF (A,NDIMA,B,NCOLB,H,EAH,P,Q,W,MAXDIMA,MAXCOLB), where

- **A**: An NDIMA x NDIMA dimensional input array; this array is the A matrix of the perturbation model
- **NDIMA**: An integer input specifying the dimension of A
- **B**: An NDIMA x NCOLB dimensional input array; this array is the B matrix of the perturbation model
- **NCOLB**: An integer input specifying the number of columns of B
- **H**: Length of the integration interval
- **EAH**: An NDIMA x NDIMA dimensional output array which approximates e^{Ah}
- **P**: An NDIMA x NCOLB dimensional output array which approximates A^{-1}(e^{Ah} - I)B
- **Q**: An NDIMA x NCOLB dimensional output array which approximates A^{-2}(e^{Ah} - Ah - I)B
- **W**: An NDIMA x NDIMA dimensional working space array
- **MAXDIMA**: An integer input specifying the maximum dimension of A
- **MAXCOLB**: An integer input specifying the maximum number of columns of B

The listing of subroutine COEFF is as follows:
SURROUNTE COEFF(A,NDIMA,B,NCOLB,H,EAH,P,Q,W,MAXDIMA,MAXCOLB)

THIS SURROUNTE CALCULATES THE MATRIX COEFFICIENTS USED IN

APPROXIMATING THE SOLUTION OF XDOT = AX + RU. HERE EAH = E**((A*H)COEFF

AND EAH IS APPROXIMATED BY (I + A*H/2)**(-1) - (I - A*H/2)**(-1)). HERE COEFF

P = A**(-1)*((E**((A*H) - I)*B AND P IS APPROXIMATED BY COEFF

H*((I - A*H/2)**(-1))*B. HERE Q = A**(-2)*((E**((A*H) - A*H = I)*R COEFF

AND Q IS APPROXIMATED BY (H**3/2)/(I + A*H/3)*B. H IS THE LENGTH COEFF

OF THE INTERVAL.

DIMENSION A(MAXDIMA,MAXDIMA), B(MAXDIMA,MAXCOLB), COEFF

1 EAH(MAXDIMA,MAXDIMA), P(MAXDIMA,MAXCOLB), COEFF

Q(MAXDIMA,MAXCOLB), W(MAXDIMA,MAXDIMA), KARRAY(7) COEFF

SET UP CONSTANTS.

H2 = H**.5
H3 = H/3.
H22 = (H**2)*.5
H36 = (H**3)/6.

DO 10 I=1,NDIMA
DO 10 J=1,NDIMA

COMPUTE I*A*H/2 AND CALL THE HFSULT EAH.

EAH(I,J) = H2*A(I,J)

IF(I.EQ.J) EAH(I,J) = 1. + EAH(I,J)

COMPUTE I*A*H/2 AND CALL THE HFSULT W.

W(I,J) = H2*A(I,J)


10 CONTINUE
INVERT I-A*H/2 AND CALL THE RESULT W.

KARRAY(1) = 10
KARRAY(2) = KARRAY(3) = NDIMA
KARRAY(5) = MAXDIMA

KARRAY(4) = KARRAY(6) = KARRAY(7) = 0
CALL MATOPS(KARRAY,W,DET,DUM)

COMPUTE (I + A*H/2)*((1 - A*H/2)**(-1)) AND CALL THE RESULT EAH.

KARRAY(1) = 20
KARRAY(4) = NDIMA
KARRAY(6) = KARRAY(7) = MAXDIMA
CALL MATOPS(KARRAY,EAH,W,EAH)

COMPUTE H*((I - A*H/2)**(-1))*B AND CALL THE RESULT P.

KARRAY(4) = NCOLR
CALL MATOPS(KARRAY,W,R,P)
DO 20 I=1,NDIMA
DO 20 J=1,NCOLB
20 P(I,J) = H*P(I,J)

COMPUTE (H**2/2)*((I + A*H/3)*B AND CALL THE RESULT Q.

DO 30 I=1,NDIMA
DO 30 J=1,NDIMA
W(I,J) = H36*A(I,J)
30 CONTINUE
CALL MATOPS(KARRAY,W,R,Q)
RETURN
END
APPENDIX A

Subroutine INTEGRT

Purpose: Generates solutions to the linear differential equations obtained from the nonlinear simulation.

Use: CALL INTEGRT (N,X,XO,XOOTH,L,U,UO,UDOT,EAH,P,Q,W1,MAXN,MAXL), where

- **N**: An integer input specifying the number of states being used; 
  \( N \leq \text{MAXN} \)
- **X**: A MAXN-dimensional input/output vector which contains the values of the states in its first \( N \) locations; on input, \( X \) contains the past values of the states, and on output, it contains the current values of the states
- **XO**: A MAXN-dimensional input vector that contains the values of the states at which the \( A \) and \( B \) matrices were calculated in its first \( N \) locations; that is, \( XO \) contains \( \bar{X}_0 \)
- **XOOTH**: A MAXN-dimensional input vector which contains \( C^*H \) in its first \( N \) locations where \( C \) is the value of \( f(\bar{X}_0,\bar{U}_0) \) and \( H \) is the same as in COEFF
- **L**: An integer input specifying the number of controls being used; 
  \( L \leq \text{MAXL} \)
- **U**: A MAXL-dimensional input vector which contains the current values of the controls in its first \( L \) locations
- **UO**: A MAXL-dimensional input vector that contains the values of the controls at which the \( A \) and \( B \) matrices were calculated in its first \( L \) locations; that is, \( UO \) contains \( \bar{U}_0 \)
- **UDOT**: A MAXL-dimensional input vector which contains the time derivatives of the controls in its first \( L \) locations
- **EAH**: A MAXN \( \times \) MAXN-dimensional input array; this is the same EAH as in COEFF
- **P**: A MAXN \( \times \) MAXL-dimensional input array; this is the same \( P \) as in COEFF
- **Q**: A MAXN \( \times \) MAXL-dimensional input array; this is the same \( Q \) as in COEFF
- **W1**: A MAXN-dimensional vector used for working space
- **MAXN**: An integer input specifying the maximum number of states
- **MAXL**: An integer input specifying the maximum number of controls

The listing of subroutine INTEGRT is as follows:
SUBROUTINE INTEGR(N, X, X0, XDOT, L, U, U0, UDOT, EA, P, Q, W1, MAXN, MAXL)

THIS SUBROUTINE APPROXIMATES THE SOLUTION TO THE DIFFERENTIAL EQUATION XDOT = F(X, U), FIRST, THE DIFFERENTIAL EQUATION DELXDOT = A * DELX + B * DELU + F(X0, U0) IS SOLVED. THE SOLUTION TO THIS DIFFERENTIAL EQUATION IS GIVEN BY DELX(K+1) = EA * DELX(K) + P * DELU(K) + Q * UDOT(K) + XDOT*K, THEN, THE SOLUTION TO XDOT = F(X, U) IS GIVEN BY X(K+1) = DELX(K+1) + X0.

DIMENSION EA(MAXN, MAXN), P(MAXN, MAXL), Q(MAXN, MAXL), U(MAXL),
U0(MAXL), UDOT(MAXL), W1(MAXN), X(MAXN), X0(MAXN),
KARRAY(7), XDOT(XMAXN)

CALCULATE DELX = X - X0 AND CALL THE RESULT X.

DO 1 I=1, N
1 X(I) = X(I) - X0(I)

CALCULATE DELU = U - U0 AND CALL THE RESULT U.

DO 2 I=1, L
2 U(I) = U(I) - U0(I)

CALCULATE EA * DELX AND CALL THE RESULT X.

KARRAY(1) = 20
KARRAY(4) = 1
KARRAY(2) = KARRAY(3) = N
KARRAY(5) = KARRAY(6) = KARRAY(7) = MAXN
CALL MATOPS(KARRAY, EA, X, X)

CALCULATE P * DELU AND CALL THE RESULT W1.

KARRAY(3) = L
KARRAY(6) = MAXL
CALL MATOPS(KARRAY, P, U, W1)
DO 3 I=1,N
  X(I) = X(I) * W1(I)
CALCULATE Q*UDOT AND CALL THE RESULT W1.
CALL MATOPS(KARRAY,Q*UDOT,W1)

DO 4 I=1,N
  X(I) = X(I) * W1(I) * X0(I) * X0DOT(I)
DO 5 I=1,L
  U(I) = U(I) * U0(I)
RETURN
END
APPENDIX B

METHOD OF CHOOSING PERTURBATION MAGNITUDES

The magnitudes of the perturbations used for the state variables in the TCV example were chosen as a function of aircraft states in which the aeronautical engineer would have some intuitive feel as to their desired range of variation. For this example, the variables used were total velocity $V_T$, angle of attack $\alpha$, angle of sideslip $\beta$, roll attitude $\phi$, pitch attitude $\theta$, and yaw attitude $\psi$. The values of the variations in the body-axis state variables are given by

$$
\Delta u_b = \Delta V_T \frac{u_b}{V_T} - w_b \Delta \alpha - v_b \cos \alpha \Delta \beta
$$

$$
\Delta w_b = \Delta V_T \frac{w_b}{V_T} + u_b \Delta \alpha - v_b \sin \alpha \Delta \beta
$$

$$
\Delta q_b = \Delta \hat{\alpha} \sin \phi \cos \theta + r_b \Delta \phi + p_b \sin \phi \Delta \theta
$$

$$
\Delta v_b = \Delta V_T \frac{v_b}{V_T} + V_T \cos \beta \Delta \beta
$$

$$
\Delta p_b = -\Delta \hat{\alpha} \sin \theta - \hat{\alpha} \cos \theta \Delta \beta
$$

$$
\Delta r_b = \Delta \hat{\alpha} \cos \phi \cos \theta - q_b \Delta \phi + p_b \cos \phi \Delta \theta
$$

where

$$
\hat{\alpha} = \frac{g \tan \phi}{V_T}
$$

$$
\Delta \hat{\alpha} = \frac{g \Delta \phi}{V_T \cos^2 \phi} - \frac{a}{V_T} \Delta V_T
$$
APPENDIX B

\[ \Delta V_T = 0.01 V_T \]

\[ \Delta \alpha = 0.2/57.3 \text{ rad} \]

\[ \Delta \beta = 0.1/57.3 \text{ rad} \]

\[ \Delta \theta = 1./57.3 \text{ rad} \]

\[ \Delta \phi = 1./57.3 \text{ rad} \]

\[ \Delta \psi = 1./57.3 \text{ rad} \]

It should be noted that all variables except \( \phi \) are at these trim values. The magnitude of \( \phi \) was not allowed to be less than 2/57.3 rad so that a nonzero value for \( \Delta \phi \) would be calculated.

The variations in the control variables were 1 percent of the total range of each control variable.
REFERENCES


TABLE I.- VALUES OF THE STATES, CONTROLS, AND STATE DERIVATIVES

AT THE POINT AT WHICH THE PERTURBATION MODEL

WAS GENERATED

[Aircraft weight, 36 287.4 kg; altitude, 457.2 m; airspeed, 63.09 m/sec; flap deflection, 40°; landing gear down]

<table>
<thead>
<tr>
<th>STATE</th>
<th>STATE DERIVATIVE</th>
<th>CONTROL</th>
</tr>
</thead>
<tbody>
<tr>
<td>UB</td>
<td>= 63.04486</td>
<td>UBDOT</td>
</tr>
<tr>
<td>WB</td>
<td>= 2.8632</td>
<td>WBDOT</td>
</tr>
<tr>
<td>QB</td>
<td>= 0.</td>
<td>QBDOT</td>
</tr>
<tr>
<td>THETA</td>
<td>= -0.00698</td>
<td>THETADOT</td>
</tr>
<tr>
<td>VB</td>
<td>= 0.</td>
<td>VBDOT</td>
</tr>
<tr>
<td>PB</td>
<td>= 0.</td>
<td>PBDOT</td>
</tr>
<tr>
<td>KB</td>
<td>= 0.</td>
<td>RBDOT</td>
</tr>
<tr>
<td>PHI</td>
<td>= 0.</td>
<td>PIDOT</td>
</tr>
<tr>
<td>PSI</td>
<td>= 0.</td>
<td>PSIDOT</td>
</tr>
<tr>
<td>ENG</td>
<td>= 35339.07894</td>
<td>STAB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DELR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DELE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DELA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPR</td>
</tr>
</tbody>
</table>
### TABLE II - THE A MATRIX

| Aircraft weight, 36 287.4 kg; altitude, 457.2 m; airspeed, 63.09 m/sec; flap deflection, 40°; landing gear down |

<table>
<thead>
<tr>
<th></th>
<th>UB DOT</th>
<th>WB DOT</th>
<th>QB DOT</th>
<th>THETADOT</th>
<th>VBDOT</th>
<th>PB DOT</th>
<th>RB DOT</th>
<th>PHIDOT</th>
<th>PSIDOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>m/sec</td>
<td>-0.03781</td>
<td>-0.28511</td>
<td>-0.00055</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>m/sec/sec</td>
<td>0.11139</td>
<td>0.72225</td>
<td>-0.01971</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>m/sec/sec</td>
<td>-2.86186</td>
<td>63.01435</td>
<td>-0.50162</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>rad/sec</td>
<td>0.00000</td>
<td>0.00000</td>
<td>1.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>rad/sec/sec</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>rad/sec/sec</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>rad/sec/sec</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
</tbody>
</table>

UB, WB, QB, THETA, VB, PB, RB, PHI, PSI

m/sec  m/sec  rad/sec  rad  m/sec  rad/sec  rad/sec  rad  rad
## TABLE III.- THE B MATRIX

| | Aircraft weight, 36 287.4 kg; altitude, 457.2 m;airspeed, 63.09 m/sec; flap deflection, 40°; landing gear down |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | |
| UBDOT | w/sec/sec | 0.00003 | 0.00458 | 0. | 0.00220 | 0. | -0.00252 | -0.00252 | | | | |
| WBDOT | m/sec/sec | 0. | -0.10089 | 0. | -0.04851 | 0. | 0.02570 | 0.02570 | | | | |
| QBDOT | rad/sec/sec | 0.00001 | -0.04098 | 0. | -0.01972 | 0. | 0.00097 | 0.00097 | | | | |
| THETADOT | rad/sec | 0. | 0. | 0. | 0. | 0. | 0. | 0. | | | | |
| VBDOT | m/sec/sec | 0. | 0. | 0.04154 | 0. | 0.00026 | -0.00383 | 0.00383 | | | | |
| PBDOT | rad/sec/sec | 0. | 0. | 0.01084 | 0. | 0.01946 | 0.01156 | -0.01156 | | | | |
| RBDOT | rad/sec/sec | 0. | 0. | -0.01102 | 0. | 0.00163 | 0.00141 | -0.00141 | | | | |
| PHIDOT | rad/sec | 0. | 0. | 0. | 0. | 0. | 0. | 0. | | | | |
| PSIDOT | rad/sec | 0. | 0. | 0. | 0. | 0. | 0. | 0. | | | | |
| THRU | N | 0.00003 | 0.00458 | 0.00220 | 0.00252 | 0.00252 | | | | | | |
| STAB | deg | 0. | 0.00458 | 0.00220 | 0.00252 | 0.00252 | | | | | | |
| DELR | deg | 0. | 0.00220 | 0.00252 | 0.00252 | | | | | | | |
| DELE | deg | 0. | 0.00220 | 0.00252 | 0.00252 | | | | | | | |
| DELA | deg | 0. | 0.00220 | 0.00252 | 0.00252 | | | | | | | |
| SPR | deg | 0. | 0.00220 | 0.00252 | 0.00252 | | | | | | | |

29
TABLE IV.- EIGENVALUES OF SYSTEM

Aircraft weight, 36 287.4 kg; altitude, 457.2 m; airspeed, 63.09 m/sec; flap deflection, 40°; landing gear down

<table>
<thead>
<tr>
<th>EIGENVALUES</th>
<th>TIME CONSTANT</th>
<th>DAMPING RATIO</th>
<th>UNDAMPED NATURAL FREQUENCY</th>
<th>PERIOD</th>
<th>t1/2</th>
</tr>
</thead>
<tbody>
<tr>
<td>-.2016E+01 + 0.</td>
<td>*I .4960E+06</td>
<td></td>
<td></td>
<td></td>
<td>.3438E+00</td>
</tr>
<tr>
<td>-.5940E-02 + 0.</td>
<td>*I .1684E+03</td>
<td></td>
<td></td>
<td></td>
<td>.1167E+03</td>
</tr>
<tr>
<td>0.</td>
<td>*I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-.1635E-01 + .1778E+00*I</td>
<td>.9161E-01</td>
<td>.1785E+00</td>
<td></td>
<td>.3.35E+02</td>
<td>.4238E+02</td>
</tr>
<tr>
<td>-.1635E-01 + -.1778E+00*I</td>
<td>.9161E-01</td>
<td>.1785E+00</td>
<td></td>
<td>.3.35E+02</td>
<td>.4238E+02</td>
</tr>
<tr>
<td>-.7636E-01 + .1138E+01*I</td>
<td>.6694E-01</td>
<td>.1141E+01</td>
<td></td>
<td>.5520E+01</td>
<td>.9077E+01</td>
</tr>
<tr>
<td>-.7636E-01 + -.1138E+01*I</td>
<td>.6694E-01</td>
<td>.1141E+01</td>
<td></td>
<td>.5520E+01</td>
<td>.9077E+01</td>
</tr>
<tr>
<td>-.6145E+00 + .1110E+01*I</td>
<td>.4845E+00</td>
<td>.1268E+01</td>
<td></td>
<td>.5663E+01</td>
<td>.1128E+01</td>
</tr>
<tr>
<td>-.6145E+00 + -.1110E+01*I</td>
<td>.4845E+00</td>
<td>.1268E+01</td>
<td></td>
<td>.5663E+01</td>
<td>.1128E+01</td>
</tr>
</tbody>
</table>

ERROR RETURN FROM REQR = 0
Aircraft weight, 36287.4 kg; altitude, 457.2 m; airspeed, 63.09 m/sec; flap deflection, 40°; landing gear down

<table>
<thead>
<tr>
<th>Flap</th>
<th>Landing Gear</th>
</tr>
</thead>
<tbody>
<tr>
<td>40°</td>
<td>0°</td>
</tr>
</tbody>
</table>

TABLE V. THE A MATRIX EIGENVECTORS

<table>
<thead>
<tr>
<th>Aircraft weight, 36287.4 kg; altitude, 457.2 m; airspeed, 63.09 m/sec; flap deflection, 40°; landing gear down</th>
</tr>
</thead>
<tbody>
<tr>
<td>.92388E-13 - .77217E-11 0.</td>
</tr>
<tr>
<td>-.31828E-13 -.30459E-11 0.</td>
</tr>
<tr>
<td>.25839E-13 -.10793E-15 0.</td>
</tr>
<tr>
<td>-.12815E-13 -.18171E-13 0.</td>
</tr>
<tr>
<td>-.99308E+00 -.28896E+00 0.</td>
</tr>
<tr>
<td>-.10499E+00 -.18419E-03 0.</td>
</tr>
<tr>
<td>-.65566E-02 -.56815E-02 0.</td>
</tr>
<tr>
<td>-.52050E-01 -.37683E-01 0.</td>
</tr>
<tr>
<td>.32519E-02 -.95658E+00 1.00000E+01</td>
</tr>
</tbody>
</table>
### TABLE VI.- BASIC FLIGHT CONDITIONS

<table>
<thead>
<tr>
<th>Condition</th>
<th>Description</th>
<th>Weight, kg</th>
<th>Center of gravity</th>
<th>Flap deflection</th>
<th>Altitude, m</th>
<th>Mach number</th>
<th>( V_C ), knots</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Approach</td>
<td>( 40,816.3 )</td>
<td>0.10( \bar{c} )</td>
<td>0.31( \bar{c} )</td>
<td>40° Down</td>
<td>0</td>
<td>0.185</td>
</tr>
<tr>
<td>II</td>
<td>Holding(^\text{b})</td>
<td></td>
<td></td>
<td>Up</td>
<td>1 254</td>
<td>0.382</td>
<td>230</td>
</tr>
<tr>
<td>III</td>
<td>Maximum dynamic pressure</td>
<td></td>
<td></td>
<td>Up</td>
<td>3 962</td>
<td>0.831</td>
<td>440</td>
</tr>
<tr>
<td>IV</td>
<td>Climb(^\text{c})</td>
<td></td>
<td></td>
<td></td>
<td>3 048</td>
<td>0.621</td>
<td>340</td>
</tr>
<tr>
<td>V</td>
<td>Maximum speed and Mach number</td>
<td></td>
<td></td>
<td></td>
<td>6 096</td>
<td>0.900</td>
<td>403</td>
</tr>
<tr>
<td>VI</td>
<td>Cruise</td>
<td></td>
<td></td>
<td></td>
<td>6 096</td>
<td>0.735</td>
<td>330</td>
</tr>
<tr>
<td>VII</td>
<td>( V_{max}/M_{max} ) corner</td>
<td></td>
<td></td>
<td></td>
<td>6 102</td>
<td>0.840</td>
<td>350</td>
</tr>
<tr>
<td>VIII</td>
<td>Maximum altitude cruise</td>
<td></td>
<td></td>
<td></td>
<td>10 058</td>
<td>0.742</td>
<td>250</td>
</tr>
<tr>
<td>IX</td>
<td>Lightweight ( V_{max}/M_{max} ) corner</td>
<td>( 31,746.0 )</td>
<td></td>
<td>0.33( \bar{c} )</td>
<td>7 102</td>
<td>0.789</td>
<td>330</td>
</tr>
</tbody>
</table>

\(^a\) For weight of \( 40\,823 \) kg
\(^b\) For weight of \( 31\,751 \) kg
\(^\text{c}\) Chosen as being 10 percent above maximum L/D speed.
\(^\text{d}\) Minimum cost climb.
### Table VII. - Characteristic Modes from Boeing Data

<table>
<thead>
<tr>
<th>Condition</th>
<th>Center of Gravity</th>
<th>Short Period</th>
<th>Phugoid</th>
<th>Dutch Roll</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( L_{SP} )</td>
<td>( P_{SEC} )</td>
<td>( t_{1/2} )</td>
<td>( L_{P} )</td>
</tr>
<tr>
<td>I</td>
<td>0.1</td>
<td>0.418</td>
<td>5.39</td>
<td>1.29</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.583</td>
<td>8.58</td>
<td>1.32</td>
</tr>
<tr>
<td>II</td>
<td>0.1</td>
<td>0.361</td>
<td>2.86</td>
<td>0.82</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.490</td>
<td>4.22</td>
<td>0.3</td>
</tr>
<tr>
<td>III</td>
<td>0.1</td>
<td>0.426</td>
<td>1.96</td>
<td>0.46</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.743</td>
<td>4.67</td>
<td>0.46</td>
</tr>
<tr>
<td>IV</td>
<td>0.1</td>
<td>0.354</td>
<td>1.97</td>
<td>0.57</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.499</td>
<td>3.04</td>
<td>0.58</td>
</tr>
<tr>
<td>V</td>
<td>0.1</td>
<td>0.419</td>
<td>2.40</td>
<td>0.57</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.924</td>
<td>12.70</td>
<td>0.58</td>
</tr>
<tr>
<td>VI</td>
<td>0.1</td>
<td>0.333</td>
<td>1.86</td>
<td>0.58</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.484</td>
<td>2.89</td>
<td>0.59</td>
</tr>
<tr>
<td>VII</td>
<td>0.1</td>
<td>0.366</td>
<td>2.28</td>
<td>0.64</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.645</td>
<td>5.00</td>
<td>0.65</td>
</tr>
<tr>
<td>VIII</td>
<td>0.1</td>
<td>0.285</td>
<td>2.45</td>
<td>1.01</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.371</td>
<td>3.76</td>
<td>1.04</td>
</tr>
<tr>
<td>IX</td>
<td>0.1</td>
<td>0.401</td>
<td>2.18</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.739</td>
<td>5.54</td>
<td>0.56</td>
</tr>
</tbody>
</table>

\(^a\) Complex conjugate pair splits into two simple poles.
### TABLE VII.- Concluded

<table>
<thead>
<tr>
<th>Condition</th>
<th>Center of gravity</th>
<th>Spiral divergence</th>
<th>Roll subsidence</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Roc'</td>
<td>tSD</td>
<td>t1/2 or t2</td>
</tr>
<tr>
<td>I</td>
<td>0.1</td>
<td>---</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>---</td>
<td>24</td>
</tr>
<tr>
<td>II</td>
<td>0.1</td>
<td>---</td>
<td>124</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>---</td>
<td>182</td>
</tr>
<tr>
<td>III</td>
<td>0.1</td>
<td>---</td>
<td>131</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>---</td>
<td>131</td>
</tr>
<tr>
<td>IV</td>
<td>0.1</td>
<td>---</td>
<td>81</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>---</td>
<td>94</td>
</tr>
<tr>
<td>V</td>
<td>0.1</td>
<td>---</td>
<td>829</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>---</td>
<td>659</td>
</tr>
<tr>
<td>VI</td>
<td>0.1</td>
<td>---</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>---</td>
<td>20</td>
</tr>
<tr>
<td>VII</td>
<td>0.1</td>
<td>---</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>---</td>
<td>56</td>
</tr>
<tr>
<td>VIII</td>
<td>0.1</td>
<td>---</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>---</td>
<td>11</td>
</tr>
<tr>
<td>IX</td>
<td>0.1</td>
<td>---</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>---</td>
<td>95</td>
</tr>
</tbody>
</table>
### TABLE VIII.—CHARACTERISTIC MODES FROM LINEAR ANALYSIS OUTPUT

<table>
<thead>
<tr>
<th>Condition</th>
<th>Center of gravity</th>
<th>Short period</th>
<th>Phugoid</th>
<th>Dutch roll</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$\xi_{SP}$</td>
<td>$P_{sec}$</td>
<td>$t_{1/2}$</td>
</tr>
<tr>
<td>I</td>
<td>0.1</td>
<td>0.399</td>
<td>5.17</td>
<td>1.31</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.585</td>
<td>11.73</td>
<td>1.79</td>
</tr>
<tr>
<td>II</td>
<td>0.1</td>
<td>0.368</td>
<td>2.90</td>
<td>0.81</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.494</td>
<td>4.26</td>
<td>0.83</td>
</tr>
<tr>
<td>III</td>
<td>0.1</td>
<td>0.430</td>
<td>2.02</td>
<td>0.47</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.742</td>
<td>4.77</td>
<td>0.47</td>
</tr>
<tr>
<td>IV</td>
<td>0.1</td>
<td>0.357</td>
<td>2.00</td>
<td>0.58</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.495</td>
<td>3.04</td>
<td>0.59</td>
</tr>
<tr>
<td>V</td>
<td>0.1</td>
<td>0.419</td>
<td>2.41</td>
<td>0.57</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.695</td>
<td>4.74</td>
<td>0.54</td>
</tr>
<tr>
<td>VI</td>
<td>0.1</td>
<td>0.323</td>
<td>1.98</td>
<td>0.64</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.455</td>
<td>3.03</td>
<td>0.65</td>
</tr>
<tr>
<td>VII</td>
<td>0.1</td>
<td>0.361</td>
<td>2.28</td>
<td>0.65</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.597</td>
<td>4.21</td>
<td>0.67</td>
</tr>
<tr>
<td>VIII</td>
<td>0.1</td>
<td>0.264</td>
<td>2.51</td>
<td>1.01</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.374</td>
<td>1.76</td>
<td>1.03</td>
</tr>
<tr>
<td>IX</td>
<td>0.1</td>
<td>0.358</td>
<td>1.99</td>
<td>0.571</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.547</td>
<td>3.46</td>
<td>0.584</td>
</tr>
</tbody>
</table>

$^a$Complex conjugate pair splits into two simple poles.
TABLE VIII.- Concluded

<table>
<thead>
<tr>
<th>Condition</th>
<th>Center of gravity</th>
<th>Spiral divergence</th>
<th>Roll subsidence</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Root</td>
<td>$T_{SD}$</td>
<td>$t_{1/2}$ or $t_2$</td>
</tr>
<tr>
<td>I</td>
<td>0.1</td>
<td>-0.003</td>
<td>312.5</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.001</td>
<td>-1429.0</td>
</tr>
<tr>
<td>II</td>
<td>0.1</td>
<td>0.027</td>
<td>-37.0</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.026</td>
<td>-38.5</td>
</tr>
<tr>
<td>III</td>
<td>0.1</td>
<td>-0.003</td>
<td>322.6</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.009</td>
<td>-111.1</td>
</tr>
<tr>
<td>IV</td>
<td>0.1</td>
<td>0.019</td>
<td>-52.63</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.019</td>
<td>-52.63</td>
</tr>
<tr>
<td>V</td>
<td>0.1</td>
<td>0.003</td>
<td>-321.6</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>-0.002</td>
<td>572.3</td>
</tr>
<tr>
<td>VI</td>
<td>0.1</td>
<td>0.027</td>
<td>-36.77</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.027</td>
<td>-37.59</td>
</tr>
<tr>
<td>VII</td>
<td>0.1</td>
<td>0.014</td>
<td>-71.94</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.014</td>
<td>-73.53</td>
</tr>
<tr>
<td>VIII</td>
<td>0.1</td>
<td>0.037</td>
<td>-27.32</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.036</td>
<td>-28.09</td>
</tr>
<tr>
<td>IX</td>
<td>0.1</td>
<td>0.030</td>
<td>-33.11</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.088</td>
<td>-11.42</td>
</tr>
</tbody>
</table>
Figure 1.- Program usage flow chart.
Figure 1.- Concluded.
Figure 2.- Procedural specification for system with rotor dynamics.