NASA Technical Memorandum 86392

UNAERO: A Package of FORTRAN Subroutines for Approximating Unsteady Aerodynamics in the Time Domain

H. J. Dunn

OCTOBER 1985

Date for general release _____ October 31, 1987 _____

NASA

UNAERO: A Package of FORTRAN Subroutines for Approximating Unsteady Aerodynamics in the Time Domain

H. J. Dunn

Langley Research Center
Hampton, Virginia

Contents

Summary

Computer subroutines for approximating unsteady aerodynamic forces in the time domain and for obtaining a set of constant-coefficient first-order differential equations are described. Information is provided for using and maintaining the subroutines. The unsteady aerodynamic forces are approximated by means of either a weighted-matrix Padé method or a least-squares method. The subroutines are capable of applying a gust spectrum to the data prior to calculating the approximations. This technique can be used to enhance the accuracy of approximations to unsteady aerodynamic forces resulting from random excitation. The approximations may be applied over a truncated set of modes at a selected number of reduced frequencies. Once the approximations to the unsteady aerodynamic forces are calculated, the coefficients are combined with the structural matrices defining the dynamic response of the vehicle to formulate a set of four matrices. These four matrices, or the system quadruple, are the coefficients of a set of first-order differential equations and output equations that represent the dynamic response of the vehicle in the time domain. The user is provided with a method of defining output equations that are derived from a linear combination of the system states. Examples are presented to demonstrate the use of the subroutines.

1 Introduction

The computer subroutine package described herein, named UNAERO, was written to implement and verify the method reported in reference 1. The package contains a collection of FORTRAN IV subroutines specifically written for use on a CDC CYBER computer system. UNAERO is intended to aid the engineer in obtaining a set of linear constant-coefficient differential equations of motion that represent the dynamic behavior of a flexible airplane. This document serves to instruct and aid the user of UNAERO in installing the package, generating the input, understanding the output, and maintaining the package.

Unlike other computer programs, such as that described in reference 2, UNAERO was designed to be incorporated into the user's existing software. UNAERO returns to the calling program the coefficients of the differential equations of motion as a set of four matrices, or the system quadruple. The user's program can use the system quadruple to produce the desired data, such as the eigenvalues of the system matrix or the singular values of the return difference matrix.

The task of generating the equations of motion of the vehicle is divided into the following two parts: task I, the generation of the coefficients of the unsteady-aerodynamics approximation, and task II, the generation of the system quadruple. Within UNAERO, either task may be performed independently of the other. The standard way of using UNAERO is to perform task I and then to use those results in task II. However, externally generated task I data may be used in UNAERO for task II. Within task I, the user has the option of using either a least-squares approximation method or the weighted-matrix Padé method. In either method the user can weight the error between the approximations and the unsteady-aerodynamic-force data to achieve the desired accuracy of fit at a specified frequency. In addition, the aerodynamic-force data representing random inputs can be modified by applying a gust spectrum before the approximations are calculated. UNAERO has a feature which allows the user to reduce the number of structural modes or the data or both before the approximations are calculated. Within task II, sensor equations can be added to the system equations through the use of subroutines supplied by the user.

The following is a brief outline of the major sections of this paper. A description of the methodology of the program functions along with the options and guidelines is presented in section 3. Section 4 is a technical description of the equations that form the basis of the subroutines within UNAERO. In section 5, the command input file for the execution of the program is described. To define the vehicle being analyzed, UNAERO uses two data bases, one for task I and one for task II. These data bases are described in section 6. Program printed output and associated options are described in section 7. Section 8, a large section concerned with the structure of the program, follows section 7. The majority of this material need only be referred to by those readers who are installing and maintaining the program. Finally, examples of the program use are presented in section 9.

If the reader is using this document for instruction on the use of the program, sections 3, 4, 5, 7, and 9 should be read in their entirety, section 6 should be studied up to and including the description of the discrete-frequency data base, and the part of section 8 on program use should be read. The information required to bypass the aerodynamic-force approximation calculations and to formulate a set of system equations from data supplied by the user's own technique is contained in the description of the continuous-frequency data base in section 6, in the appropriate information from sections 4, 5, and 7,

and in the last part of example 1 in section 9. The information presented in appendix A describes the interface subroutines PRESEN and SENSOR for use in adding output equations to the generated system quadruple. Subroutine USER is described in appendix B. This subroutine is used to modify the value of the aerodynamic force before the approximation is calculated.

2 Symbols

$\mathbf{A}$	matrix of coefficients, a function of the reduced frequency k_l used in solution of aerodynamic-force approximation (see eq. (12))
$\mathbf{A}^\dagger$	pseudoinverse of $\mathbf{A}$
$\overline{\mathbf{A}}$	weighted $\mathbf{A}$
$\mathbf{A0}, \mathbf{A1}, \mathbf{A2}$	coefficient matrices of aerodynamic-force approximation of equation (6)
$\mathbf{b}$	vector of constant coefficients used in solution of aerodynamic approximation (see eq. (13))
$\overline{\mathbf{b}}$	weighted $\mathbf{b}$ (see eq. (15))
b_m	mth element of $\mathbf{b}$
$\bar{c}$	reference length
$\mathbf{D}$	generalized damping matrix
$\mathbf{D}_m$	coefficient matrices of aerodynamic-force approximation of equation (6)
$\mathbf{I}$	identity matrix
J_j	objective function (see eq. (9))
j	$= \sqrt{-1}$
K	$= (n)(N+2)$
$\mathbf{K}$	generalized stiffness matrix
k	reduced frequency, $\omega \bar{c}/2V$
k_l	specific reduced frequency
L	gust characteristic length
M	$= (n)(N_k)$
$\mathbf{M}$	generalized mass matrix
N	order of the approximation
N_k	number of reduced-frequency values
n	number of modes used to calculate the approximation
P	an element of $\mathbf{P}$

$\mathbf{P}, \mathbf{P}(j\omega)$	matrix of numerator polynomials of unsteady-aerodynamic-force approximation
$P_{ij}(jk_l)$	ith,jth element of $\mathbf{P}$ at the reduced frequency k_l
$\mathbf{P}_m$	matrix of coefficients of $\mathbf{P}(j\omega)$ (see eq. (5))
$\mathbf{Q}(j\omega)$	unsteady-aerodynamic-force matrix
$\widehat{\mathbf{Q}}(j\omega)$	approximation to the unsteady-aerodynamic-force matrix
$Q_{ij}(jk_l)$	ith,jth element of $\mathbf{Q}(j\omega)$ at the reduced frequency k_l
$\widehat{Q}_{ij}(jk_l)$	ith,jth element of $\widehat{\mathbf{Q}}(j\omega)$ evaluated at reduced frequency k_l
$\mathbf{q}(j\omega)$	vector of generalized coordinate
$\mathbf{R}, \mathbf{R}(j\omega)$	matrix of denominator polynomials of unsteady-aerodynamic-force approximation
$\overline{\mathbf{R}}(j\omega)$	matrix $\mathbf{R}(j\omega)$ with leading coefficient equal to $\mathbf{I}$
$R_j(jk_l)$	jth element of $\mathbf{R}(j\omega)$ evaluated at reduced frequency k_l
R_j	jth element of matrix $\mathbf{R}$
$\mathbf{R}_l$	matrix of coefficients of $\mathbf{R}(j\omega)$ (see eq. (4))
r	coefficient of the polynomial R
s	semispan
$\mathbf{u}$	control state vector (see eq. (27))
V	reference velocity
$\mathbf{W}$	weight matrix defined by equation (14)
$\overline{\mathbf{W}}$	real diagonal matrix (see eq. (14))
W_m	an element of $\mathbf{W}$
$\mathbf{X}$	system state vector (see eq. (25))
$\mathbf{x}$	solution vector used in calculation of unsteady-aerodynamic-force approximation (see eq. (13))
x_{ij}	an element of $\mathbf{x}$
$\mathbf{y}$	system output vector (see eq. (25))
$\mathbf{z}$	error between the approximation and the data value (see eq. (17))

$\bar{z}$	complex conjugate of z
$\frac{1}{2}\rho V^2$	dynamic pressure of the free stream
σ	root-mean-square gust intensity
ω	frequency of oscillation
$\| \ \|$	norm, square root of the scalar product

Subscripts:

af	state vector used to model unsteady-aerodynamic-force approximation
C	control-surface mode
G	gust mode
m	state vector used for generalized coordinate
S	structural mode
W	weighted norm
δ	S, G, or C

3 Discussion of the Programmed Options and Guidelines

This section is concerned with the description and rationale of the different options that can be invoked within UNAERO. As mentioned previously, UNAERO consists of two major parts, or tasks. The first task (task I) is to calculate the parameters of the unsteady-aerodynamic-force approximations, and the second task (task II) is to formulate the equations of motion of the vehicle from these calculated parameters. In performing these two tasks, there are several options that are available. Some of the options aid the user in a search for the smallest set of dynamic equations that adequately represents the aircraft's dynamic response. Other options are for the efficient use of computer resources. The options that control the calculation of the parameters in the approximations (task I) are described first. The options that control the formulation of the dynamic equations of motion (task II) are described subsequently.

3.1 Task I: Options and Guidelines

In the development of the methodology for calculating the parameters of the approximations to the unsteady aerodynamic forces, the assumption is usually made that the dynamics of the aircraft can be represented accurately by a finite number of aircraft modes. Furthermore, the generalized forces acting on the aircraft are a linear combination of forces arising from the motion of rigid-body or flexible modes, motion of control-surface modes, and motion of forces resulting from an external random disturbance. These three sources are hereinafter referred to as the structural modes, control modes, and gust modes, respectively. For the structural modes, UNAERO requires the generalized mass, damping, and frequency in the data base.

UNAERO makes no distinction between the forms of the unsteady-aerodynamic-force approximations used for any of the structural or control modes. The form of the approximation that is used for the structural and control modes is the following rational transfer function:

$$\frac{\sum\limits_{i=0}^{K} (jk)^i a_i}{\sum\limits_{l=0}^{N} (jk)^l b_l}$$

with the order of the numerator polynomial K being two greater than the order of the denominator polynomial N. The additional derivatives that are required for the structural mode approximations are obtained from the equations used to represent the motion of the generalized modes. These equations are generated with the generalized mass, damping, and frequency data. For the control modes, UNAERO expands the number of control inputs to be three times the original number of control modes. The additional inputs represent the first and second time derivatives of each control deflection. The form of the aerodynamic-force approximation used for the gust mode is identical to the form used for the structural mode except that it is restricted to be a causal function. This form of the approximation for a gust mode does not require additional inputs that represent the derivatives of the gust mode (i.e., the order of the numerator K does not exceed the order of the denominator polynomial N for the approximation).

For all unsteady-aerodynamic-force approximations that are dependent on the dynamics of a mode (i.e., all elements of a column of the generalized force matrix), the user has the option of using the least-squares method or the matrix Padé method to solve for the denominator coefficients of the approximation. The recommended procedure for calculating the approximation is to use the Padé method first and to adjust the parameters of the approximation (order of the approximation, number of modes, etc.) until the desired degree of accuracy is obtained. Then the results of the Padé method are used as the values of the denominator coefficients for the least-squares

method. This procedure can result in a substantial savings in the resources used for an analysis of a vehicle. The savings are obtained because once denominator coefficients are determined with the Padé method, they will be constant. Since these coefficients are constant and known, they can be used as input to the least-squares method to produce approximation functions with identical numerator coefficients but which require less resources to calculate.

A special case has been added to the least-squares method to allow a zero-order approximation to the aerodynamic forces. An approximation of zero order has no lag terms, and therefore no additional states are required to approximate the unsteady aerodynamic forces. In this case, the number of system states in the resulting equations of motion is two times the number of modes in the structural model. Although the approximations of zero order generate functions that produce large errors, the error at any value of reduced frequency can be minimized through the use of a weighted least-squares solution for the coefficients of the approximations. An example in section 9 illustrates this option. This technique of using weights to selectively reduce the error of the function can also be used with the Padé method and the least-squares method when the order of the approximation is greater than zero.

The data being used to generate the approximation functions can be adjusted so as to be a subset of the original data. If this option is used, a number of systems can be generated and evaluated as to which modes make a significant contribution to the dynamics of the problem being investigated. Also, the number of reduced frequencies where data are used to calculate the approximating functions can be reduced so that the resulting function is based on a frequency range that is significant for the mode being approximated. This could also be accomplished with a value of zero for the weight at the desired value of reduced frequency. But it would not be as efficient as removing the data value from the data base used to calculate the approximating function. Reducing the number of modes used to represent the dynamics of the vehicle or the number of data values used or both will reduce the amount of computer resources required to calculate the coefficients of the approximation.

Based on experience gained in using UNAERO, it has been observed that approximating gust modes accurately is a more difficult task than approximating other types of modes. In an effort to improve the quality of the gust mode representation, there is an option in UNAERO which allows the modification of the aerodynamic data (by a gust attenuation function) before the approximation coefficients are calculated. The attenuation can be based on the magnitude of a Von Karman or a Dryden gust spectrum or represented by a FORTRAN subroutine supplied by the user. The details needed for writing this subroutine are contained in appendix B. Because of the method by which this option is implemented, the data being modified are not limited to the gust mode data and can be used for any mode.

3.2 Task II: Options and Guidelines

When UNAERO is being used to formulate the equations of motion from data that have been previously approximated, there are two options that are available to the user. In the first option the reference condition at which the equations of motion are formulated must be specified. This reference condition, or set point, is defined and must be redefined each time the equations of motion are formulated. Of the many different ways that the reference condition could be specified, for UNAERO a characteristic length, velocity, and dynamic pressure are required. This set of parameters allows the equations of motion to be conveniently reformulated at another dynamic pressure. One of the first checks on the formulation of the equations of motion can be an inspection of the system matrix with the dynamic pressure set to zero. This ensures that the generalized mass, damping, and stiffness are entered in the data base correctly.

In the second option the user may change the coefficients of the output equations, thereby allowing, for example, an analysis of the effect of changing the location of a sensor. Since adjusting the coefficients of the output equations requires the interpolation of the modal displacements, the user must supply the subroutines PRESEN and SENSOR to use this option. The details of these two subroutines are given in appendix A.

4 Description of Programmed Equations

The equations within UNAERO are described in this section. This description is a supplement to the technical description of the methods in reference 1 and is not intended to provide the detailed technical development that was presented in that reference. The equations of motion that describe the airplane, written in the frequency domain, are described first. The unsteady-aerodynamic-force approximation functions are then described, followed by a description of the objective function and the equations used in the optimization procedures to minimize that objective function. The time domain equations that represent the dynamic response of the aircraft are then presented.

4.1 Task I Equations

In the development of the equations of motion for an elastic airplane, the motion is assumed to be a perturbation from straight and level flight. The equations of motion in the frequency domain, which were developed in equations (1) to (10) of reference 1, are summarized as follows:

$$\left[(j\omega)^2 \mathbf{M} + (j\omega)\mathbf{D} + \mathbf{K} + \frac{1}{2}\rho V^2 \mathbf{Q}(j\omega)\right] \mathbf{q}(j\omega) = 0 \tag{1}$$

where $\mathbf{M}$, $\mathbf{D}$, and $\mathbf{K}$ are the generalized mass, generalized damping, and generalized stiffness matrices, $\frac{1}{2}\rho V^2$ is the dynamic pressure of the free stream, $\mathbf{Q}(j\omega)$ is the unsteady-aerodynamic-force matrix, and $\mathbf{q}(j\omega)$ is the vector of generalized coordinates. To develop the equations that are used in UNAERO, equation (1) is rewritten with the aerodynamic forces separated to take into account the notation for the different modes that were described earlier. The modified equation (1) is

$$\left[(j\omega)^2 \mathbf{M} + (j\omega)\mathbf{D} + \mathbf{K} + \frac{1}{2}\rho V^2 \mathbf{Q}_S(j\omega)\right] \mathbf{q}_S(j\omega)$$
$$+ \frac{1}{2}\rho V^2 \mathbf{Q}_C(j\omega)\mathbf{q}_C(j\omega) = -\frac{1}{2}\rho V^2 \mathbf{Q}_G(j\omega)\frac{\mathbf{q}_G(j\omega)}{V} \tag{2}$$

The subscripts S, C, and G denote a generalized force or coordinate associated with a structural mode, a control mode, and a gust mode, respectively.

The equation used to represent the approximations of order N to the unsteady-aerodynamic forces due to motion of the structural modes or the control modes can be written as equation (3).

$$\widehat{\mathbf{Q}}_\delta(j\omega) = \mathbf{P}_\delta(j\omega)\mathbf{R}_\delta(j\omega)^{-1} \tag{3}$$

where δ represents either S or C,

$$\mathbf{R}_\delta(j\omega) = (j\omega)^N \left(\frac{\bar{c}}{2V}\right)^N \mathbf{I} + \sum_{l=0}^{N-1} (j\omega)^l \mathbf{R}_{l,\delta} \left(\frac{\bar{c}}{2V}\right)^l \tag{4}$$

and

$$\mathbf{P}_\delta(j\omega) = \sum_{m=0}^{N+2} (j\omega)^m \mathbf{P}_{m,\delta} \left(\frac{\bar{c}}{2V}\right)^m \tag{5}$$

The matrix $\widehat{\mathbf{Q}}(j\omega)$ is the approximation in the frequency domain of the unsteady aerodynamic forces, $\mathbf{P}(j\omega)$ is a matrix such that the ith,jth element represents the numerator polynomials of the unsteady-aerodynamic-force approximation, and $\mathbf{R}(j\omega)$ is a diagonal matrix such that the jth diagonal element is the denominator polynomial of the approximation

which best fits the jth column of the matrix $\mathbf{Q}(j\omega)$ at a number of values of reduced frequency. Each element of the matrix $\mathbf{P}(j\omega)$ is a polynomial of degree $N + 2$ in $j\omega$. Each diagonal element of the matrix $\mathbf{R}(j\omega)$ is a polynomial of degree N in $j\omega$. Because $\mathbf{R}(j\omega)$ is a diagonal matrix, each column of the approximation matrix $\widehat{\mathbf{Q}}(j\omega)$ will have the same denominator polynomial. Equation (4) represents the denominator polynomials and equation (5) represents the numerator polynomials of equation (3) rewritten so that all the numerator coefficients appear as constant matrices.

Equations (4) and (5) are the internal representation of the approximation for the UNAERO package. The elements of the denominator polynomial $\mathbf{R}(j\omega)$ are defined in table 1. The polynomial used is different for each order of the approximation because of the technique used to solve for the elements of the matrix $\mathbf{R}(j\omega)$. The form of equation (5) that appears the most in the literature and that is used in UNAERO during the task II phase when formulating the equations of motion is written as

$$\widehat{\mathbf{Q}}_\delta(j\omega) = \mathbf{A0}_\delta + (j\omega)\left(\frac{\bar{c}}{2V}\right)\mathbf{A1}_\delta + (j\omega)^2 \left(\frac{\bar{c}}{2V}\right)\mathbf{A2}_\delta$$
$$+ \left[\sum_{m=0}^{N-1} (j\omega)^m \left(\frac{\bar{c}}{2V}\right)^{m-N} \mathbf{D}_{m,\delta}\right] \overline{\mathbf{R}}_\delta(j\omega)^{-1} \tag{6}$$

where the matrix $\overline{\mathbf{R}}(j\omega)$ is defined as

$$\overline{\mathbf{R}}_\delta(j\omega) = \left(\frac{\bar{c}}{2V}\right)^{-N} \mathbf{R}_\delta(j\omega) \tag{7}$$

For forces resulting from the random or gust excitation, the approximation function is written as

$$\widehat{\mathbf{Q}}_G(j\omega) = \mathbf{A0}_G$$
$$+ \left[\sum_{m=0}^{N-1} (j\omega)^m \left(\frac{\bar{c}}{2V}\right)^{m-N} \mathbf{D}_{m,G}\right] \overline{\mathbf{R}}_G(j\omega)^{-1} \tag{8}$$

The matrices $\mathbf{A0}$, $\mathbf{A1}$, $\mathbf{A2}$, and $\mathbf{D}_m$ in equations 6 to 8 and the matrix $\mathbf{R}_l$ in equation (4) are the matrices computed by UNAERO. Note that equation (8) is equivalent in form to equation (6) with matrices $\mathbf{A1}$ and $\mathbf{A2}$ set to zero.

4.1.1 Least-squares approximation equations. The objective function, or the function that is minimized by calculating the coefficients of the approximations with either the least-squares method or the Padé

5

method for the jth column of the unsteady aerodynamic forces, is:

$$J_j = \frac{1}{2} \sum_{i=1}^{n} \sum_{l=1}^{N_k} \|Q_{jk}(jk_l) - \widehat{Q}_{ij}(jk_l)\|^2 \quad (9)$$

where n is the number of modes used in the approximation, N_k is the number of reduced-frequency values used to calculate the approximation, $Q_{ij}(jk_l)$ is the value of the ith,jth generalized aerodynamic force at the reduced frequency k_l, and $\widehat{Q}_{ij}(jk_l)$ is the value of the ith,jth approximation function evaluated at reduced frequency k_l.

For the least-squares approximation method, equation (9) is minimized by solving for the numerator coefficients with the method described in reference 1 and modified so that weights on the error between the approximation and the calculated force at specified frequencies can be applied. Note that when the denominator coefficients of the approximation are constant, which is the case for the least-squares method, then the solution to the minimization of the following equation will also be the solution that minimizes equation (9):

$$J_j = \frac{1}{2} \sum_{i=1}^{n} \sum_{l=1}^{N_k} \|Q_{ij}(jk_l)R_j(jk_l) - P_{ij}(jk_l)\|^2 \quad (10)$$

where P_{ij} is the numerator polynomial for the ith,jth element and R_j is the denominator polynomial for the jth column of the approximation; both are evaluated at the reduced frequency of k_l. If the elements of the summation over the indices l and i of equation (10) are grouped together to form a vector $\mathbf{b}$ of size $M = (n)(N_k)$, the mth element of $\mathbf{b}$ would be defined as

$$b_m = Q_{ij}(jk_l)R_j(jk_l) \qquad m = n(l-1)+i \quad (11)$$

The coefficients of the numerator polynomials for the jth column are also grouped into a vector $\mathbf{x}$, of length $K = (n)(N+2)$, such that when this vector is premultiplied by an $M \times K$ matrix $\mathbf{A}$, the mth element of the resulting vector is the ith numerator polynomial evaluated at the k_l value of reduced frequency, or

$$P_{ij}(jk_l) = (\mathbf{Ax})_{m\text{th element}} \quad (12)$$

The matrix $\mathbf{A}$ is a complex matrix formed from the discrete values of reduced frequency being used to calculate the approximation. Thus, equation (10)

can be rewritten as

$$J_j = \frac{1}{2} \|\mathbf{b} - \mathbf{Ax}\|^2 \quad (13)$$

Note that the summation of the scalar norm of equation (10) is equivalent to the vector norm of the algebraic quantity in equation (13). The real vector $\mathbf{x}$ contains all the coefficients of the numerator polynomials P_{ij} that are the solution of the least-squares optimization procedure.

To modify equation (13) for the weighted optimization procedure, define a weight matrix as

$$\mathbf{W} = \overline{\mathbf{W}}'\overline{\mathbf{W}} \quad (14)$$

where $\overline{\mathbf{W}}$ is a real $M \times M$ diagonal matrix of weights and the prime indicates the transpose of the matrix. Then, by defining

$$\left.\begin{array}{l} \overline{\mathbf{b}} = \overline{\mathbf{W}}\mathbf{b} \\ \overline{\mathbf{A}} = \overline{\mathbf{W}}\mathbf{A} \end{array}\right\} \quad (15)$$

equation (13) becomes

$$J_j = \frac{1}{2}\|\overline{\mathbf{b}} - \overline{\mathbf{A}}\mathbf{x}\|^2 = \frac{1}{2}\|\mathbf{b} - \mathbf{Ax}\|_{\mathbf{W}}^2 \quad (16)$$

The vector $\mathbf{x}$ is found such that it minimizes equation (16) by the least-squares method described in the appendix of reference 1.

4.1.2 Padé approximation equations.

In order to minimize the Padé objective function, a nonlinear optimization algorithm is used that requires analytical gradients. To obtain these gradients, equation (9) is rewritten as

$$J_j = \frac{1}{2}\mathbf{z}'\mathbf{W}\overline{\mathbf{z}} \quad (17)$$

where $\mathbf{z}$ is a complex vector of length $M = (n)(N_k)$ with an mth element of

$$z_m = Q_{ij}(jk_l) - \widehat{Q}_{ij}(jk_l) \quad (18)$$

where the subscript m is $i + n(l-1)$, $\overline{\mathbf{z}}$ is the complex conjugate of $\mathbf{z}$, and n is defined in equation (9). The mth element of $\mathbf{z}$ represents the error between the data and the approximation at the lth value of reduced frequency for the ith mode. The term $\mathbf{W}$ is a diagonal matrix of weights such that the mth element is the weight on the error for the ith mode at the lth value of reduced frequency, and N_k is the number of reduced-frequency values being used to calculate the

approximations. Equation (17) can be rewritten as

$$J_j = \frac{1}{2} \sum_{m=1}^{M} \left[\text{Re}(z_m)^2 + \text{Im}(z_m)^2 \right] W_m \qquad (19)$$

where $\text{Re}(z_m)$ and $\text{Im}(z_m)$ are the real and imaginary parts of the mth element of $\mathbf{z}$, W_m is the mth element of the matrix $\mathbf{W}$ defined in equation (17), and $M = (n)(N_k)$. Taking the partial derivative of J_j with respect to a parameter r of the denominator polynomial yields

$$\frac{\partial J_j}{\partial r} = \sum_{m=1}^{M} \left[\text{Re}(z_m)\text{Re}\left(\frac{\partial z_m}{\partial r}\right) + \text{Im}(z_m)\text{Im}\left(\frac{\partial z_m}{\partial r}\right) \right] W_m \qquad (20)$$

If R_j is an element of the matrix $\mathbf{R}$ in equation (4) (for which the coefficients are defined in table 1) and P is the jth column vector of the matrix $\mathbf{P}$ in equation (5), then the column vector that is the partial derivative of $\mathbf{z}$ with respect to the parameter r can be calculated from a relationship of the form

$$\frac{\partial z}{\partial r} = -\frac{\left[R_j \frac{\partial P}{\partial r} - P \frac{\partial R_j}{\partial r} \right]}{R_j^2} \qquad (21)$$

where the appropriate elements of the polynomials P and R_j must be used to be consistent with the element of $\mathbf{z}$ for which the derivative is being calculated. The relationship for $\partial P / \partial r$ is obtained from the solution of the least-squares minimization of equation (13) as

$$\frac{\partial P}{\partial r} = \mathbf{A}^\dagger \begin{bmatrix} \text{Re}\left(\mathbf{Q}\frac{\partial R_j}{\partial r}\right)\mathbf{W} \\ \text{Im}\left(\mathbf{Q}\frac{\partial R_j}{\partial r}\right)\mathbf{W} \end{bmatrix} \qquad (22)$$

Equations (20) to (22) and the denominator polynomials defined in table 1 are used to calculate the gradient of equation (17), which is then used in the optimization procedure to calculate the denominator coefficients of the approximation function.

4.1.3 Gust attenuation equations.
In order to modify the calculated unsteady-aerodynamics gust data with a gust attenuation function before the approximation function is calculated, the following equations are evaluated to obtain the attenuation to be applied.

Von Karman attenuation function:

$$\Phi_{\text{VK}}(k) = \sigma^2 L \frac{1 + (32/3)[1.339(L/\bar{c})k]^2}{\{1 + 4[1.339(L/\bar{c})k]^2\}^{11/6}} \qquad (23)$$

Dryden attenuation function:

$$\Phi_{\text{D}}(k) = \sigma^2 L \frac{1 + 24[(L/\bar{c})k]^2}{\{1 + 4[(L/\bar{c})k]^2\}^2} \qquad (24)$$

The quantity σ is the root-mean-square gust intensity and L is the characteristic length, or scale, of the gust excitation. Equations (23) and (24) are based on equations (6) and (8), respectively, of reference 3 and have been modified to account for the differences in the Fourier transform used in references 1 and 3.

4.2 Task II Equations

The time domain equations of motion, or system quadruple, are formulated in the following manner: (1) the approximations for the unsteady aerodynamic force (eqs. (3) to (6)) are substituted into the frequency domain formulation of the equations of motion (eq. (2)), and (2) the resulting equations are transformed into the time domain by performing an inverse Fourier transformation. (The details of those transformations are in ref. 1.) The time domain equations are then in the following form:

$$\left. \begin{array}{l} \dot{\mathbf{X}} = \mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{u} \\ \mathbf{y} = \mathbf{C}\mathbf{X} + \mathbf{D}\mathbf{u} \end{array} \right\} \qquad (25)$$

where A, B, C, and D form the system quadruple that defines the equations of motion, and $\mathbf{X}$ is a state vector composed of all the states to represent the dynamics of the vehicle and has the following definition:

$$\mathbf{X} = \left\{ \begin{array}{l} \mathbf{q}_m \\ \dot{\mathbf{q}}_m \\ \mathbf{q}_{\text{af}} \end{array} \right\} \qquad (26)$$

where $\mathbf{q}_m$ is a vector containing the generalized coordinate of any mode. The vector $\dot{\mathbf{q}}_m$ is the time derivative of the vector $\mathbf{q}_m$. The vector $\mathbf{q}_{\text{af}}$ is a vector formed by grouping the states used to approximate the unsteady aerodynamic forces.

The elements of the vector $\mathbf{q}_{\text{af}}$ are arranged such that the states that are used for an individual approximation are in sequence. For example, if the approximation of the unsteady aerodynamics due to motion of the first mode is third order and the approximation for the aerodynamics due to motion of the second mode is second order, then the first three elements of $\mathbf{q}_{\text{af}}$ will be the state variables used in the

approximation of the aerodynamics due to motion of the first mode and the next two will be used in the approximation for the aerodynamics due to motion of the second mode. Additional unsteady-aerodynamic-force states due to the motion of modes will be added in a similar manner. Note that the order of the states is different from the order of the states that is used in equation (16) in reference 1. The control vector $\mathbf{u}$ is expanded to provide the required input as follows:

$$\mathbf{u} = \left\{ \begin{array}{c} u_1 \\ \dot{u}_1 \\ \ddot{u}_1 \\ \vdots \\ u_m \\ \dot{u}_m \\ \ddot{u}_m \\ u_g \end{array} \right\} \qquad (27)$$

where, for this illustration, there are m aerodynamic controls u_1 to u_m and one gust input u_g. The output vector $\mathbf{y}$ is defined by the user and can be any linear combination of the state vector, the derivative of the state vector, and the control vector. (See appendix A.)

5 UNAERO Command Input

There are two distinct sets of program input. The first set, the command input, selects the different options of the program and is described here. (The second set, the data bases, defines the unsteady-aerodynamic-force model and is discussed in section 6.) Examples of the command input are given in figure 1 and are executed by the program in figure 2. The syntax of the command input file was designed to be self-documenting in that the characteristic parameters of the model being generated can be understood from examining the input data without the need of a reference manual. Also, an effort was made to allow the user an easy method for changing the mutually exclusive options. For example, to change from a Padé approximation to a least-squares approximation, the analysts need only change the phrase "PADÉ" to "LEAST SQUARES" on the appropriate data record.

5.1 Key Elements

Throughout this discussion, the term "record" refers to one card image. Each input record has space allocated for a description of the data that are contained on the record. A position-dependent portion of this description will be referred to as the "key element." The key element is used to differentiate the command input data by UNAERO. Exceptions to

this are data records that can have a variable number of items, such as data records that define the modal selection array or the frequency selection array. For these exceptions, the description and the key element are contained on the previous record. The key elements must appear in the correct position in the record relative to the start of the record. Figure 3 is a listing of all the legal command input records. The relative position of each key element can be found by comparing the column position of the key element with the first and second entries in the figure. For example, in figure 3 the key element of the first command input, "E LE," begins at the ninth relative position from the start of the record. If the key element does not agree with the expected key element or if there is an error while decoding the data, such as an illegal character in the data field, an error message is generated. The key elements of figure 1(a) are underlined in figure 3.

5.2 Syntax Diagrams

The command input is best described by the use of the syntax diagrams of figure 4. The diagrams are used to construct the input records for UNAERO. Starting at the upper left arrow of each diagram, the path through the diagram is indicated by the arrows. A path through the diagram defines a syntactically correct input into UNAERO. Each box within the diagram references by name another diagram, which is then used to define the expected input. For convenience, the figure number that this box refers to is in the lower right-hand corner. For example, the word "Block" in figure 4(a) refers to figure 4(b). The word "Comment" in figure 4(b) refers to figure 4(c). This process continues until a terminal symbol is reached. A terminal symbol represents actual data input into the program and is indicated in figure 4 by a rounded enclosure. For an example, see figure 4(d). In figure 4, the key element in each of the terminal symbols is underlined. Data formats are enclosed in parentheses and must be strictly adhered to.

5.3 Example of Using Syntax Diagrams

Consider for this example the following single command statement:

REFERENCE LENGTH (F10.0)

The character sequence "E LE" is the key element beginning at the ninth relative position from the beginning of the record. Also, there is a data field starting one space after "LENGTH" which is read by the program with a format of F10.0. The path for this example is as follows:

1. The command input starts with figure 4(a). The word "Block" refers the path to figure 4(b).

2. In figure 4(b), the path for this example continues to figure 4(d), "Statements for calculating approximation."

3. In figure 4(d), the path is referred to figure 4(f), "Define set point."

4. In figure 4(f), the terminal symbol is reached.

From figure 4(f), the program expects the next statement to be

REFERENCE VELOCITY (F10.0)

This input statement defines the reference velocity. The units must be consistent with the reference system that was used in developing the data base. Note that, according to figures 4(b) and 4(c), if the original record had been

USE APPROXIMATION ON VDATA FILE

instead of "REFERENCE LENGTH," the "REFERENCE VELOCITY" statement would have generated an error. Examples of syntactically correct input can be found in figure 1.

6 UNAERO Data Bases

This section describes the second set of input to UNAERO, the data bases. Since the input described here is considered to be a data base rather than command input, as described in section 5, the file structure is not self-documenting. Allowance is made for comments to describe the blocks of data. The data bases contain a description of the unsteady-aerodynamic-force model of the aircraft under investigation.

There are two data bases, one for each of the basic tasks. For task I the data base contains information describing the unsteady-aerodynamic-force model at a specific number of reduced frequencies, as would be supplied by an analytical oscillatory aerodynamic program. Henceforth this data base is referred to as the discrete-frequency data base, or the discrete data base. The second data base, used for task II, differs from the first in that the aerodynamics of the model have been approximated and the coefficients of the approximation have replaced the unsteady-aerodynamics data of the discrete-frequency data base. Only the selected modes which are specified in task I are retained in the discrete-frequency data base. This form of the data base will be referred to as the continuous-frequency data base, or the continuous data base. The continuous-frequency data base can either be generated internally from the discrete-frequency data base by execution of UNAERO or externally by some other method. If generated externally, the user is responsible for ensuring that the data format and structure conform to what UNAERO requires. This provides a convenient method for comparing different unsteady-aerodynamics approximation techniques.

6.1 Description of the Discrete-Frequency Data Base

The discrete-frequency data base has all the information necessary to calculate the approximation to the unsteady-aerodynamics model and to generate the continuous-frequency data base. Figure 5 contains a partial listing of an example of this data base. This data base is large even for a small number of modes and reduced frequencies. In order to reduce the amount of data presented, only the significant portions of the data base are listed. The data base contains an unsteady-aerodynamics model for the DAST ARW-1 wind tunnel model at a Mach number of 0.9. It will be useful, in the following discussion, to refer to figure 5.

The discrete-frequency data base contains mandatory comment records. The comment records are not optional and must be present even if they are blank. In figure 5, records containing alphanumeric information are the mandatory comment records and the numbers in parentheses on these records correspond to the numbered items in the following list. These numbers in parentheses have been placed to the left of the actual record and do not represent actual data in the data base. Each numbered item (1) to (8) describes a section of the data base.

(1) The first record of the data base is a comment record used to describe the contents of the data base. The second record describes the format for reading all real-type data arrays in the rest of the data base. This record must be a syntactically correct FORTRAN FORMAT statement and must be enclosed in parentheses. In the following sections, this format is stored in the array called IFORM.

(2) The second record beginning this section describes the system parameters that are on the next record. The next record following the comment record contains four integer-type data elements in the format of 4I5. These four system parameters are the number of structural modes NM, the number of control modes NC, the number of gust modes NG, and the number of reduced frequencies NRF for which there are aerodynamic data.

(3) The third section of the data base contains the generalized mass matrix GM. This is a square matrix and UNAERO uses the following FORTRAN statement for reading GM:

READ(JS1,IFORM)((GM(I,J),J=1,NM),I=1,NM)

9

(4) The generalized damping coefficients are specified in this section. UNAERO reads NM data elements with the IFORM format specification. The data elements are placed along the diagonal of the square generalized damping matrix.

(5) The natural frequencies (in hertz) for each of the structural modes are read in this section. UN-AERO reads NM data elements with the IFORM format and creates a square generalized frequency matrix.

(6) The reduced frequencies for which the oscillatory aerodynamic data were calculated are read in this section. UNAERO reads NRF data elements with the IFORM format.

(7) This section contains the real parts of the generalized aerodynamic forces for the first frequency. UNAERO reads these data with the following statement:

READ(JS1,IFORM)((DUM(I,J),I=1,M1), J=1,M2)

where DUM is the variable array where the data are initially stored, M1 is the sum of the number of structural modes and the number of controls NM+NC, and M2 is the sum of the number of structural modes, the number of controls, and the number of gust modes NM+NC+NG.

(8) This section contains the imaginary parts of the generalized aerodynamic forces for the first frequency. The data are read with a statement identical to that in item (7).

Items (7) and (8) are repeated for each reduced frequency that is contained in the data base. Optional data records can be appended to the end of the data base for use by the user-defined subroutine PRESEN. For a description of the method for adding sensors to the model definition, see appendix A.

6.2 Description of the Continuous-Frequency Data Base

The continuous-frequency data base consists of the calculated parameters for the unsteady-aerodynamic-force approximation plus all other data necessary to construct a system of first-order constant-coefficient differential equations at a given flight condition. Figure 6 contains an example of this data base. This data base was generated from the data base described in figure 5, but retains only the first, second, fourth, fifth, and sixth modes. The generalized aerodynamic forces were approximated with a first-order Padé approximation. As in the description of the discrete-frequency data base, only the significant portions of the data base are listed in order to conserve space. Each numbered element of the following list describes a section of the data base that has the number enclosed in parentheses to the left of the actual data. The comment on the description record is not required, but the record is not optional and must be present even if it is blank.

(1) The first record of the data base is a comment record that describes the contents of the data base. (If this data base was generated by UNAERO, the first record is the same as the first record for the discrete data base.)

(2) The second section of the data base begins with a comment record followed by a data record containing, in 3I5 format, the number of structural modes NM, the number of control modes NC, and the number of gust modes NG that are contained in the data base. The next record in this section must be a syntactically correct FORTRAN FORMAT statement that describes the format for reading the successive data. If the data base is generated by UNAERO, this format is "(6G12.5)." This format is stored in the variable IFORM.

(3) This comment record indicates that the coefficients of an aerodynamic approximation are to follow. Since the input data records are the same for a structural mode or for a control mode, both sets of input are described here. UNAERO uses the comment "MODE" to indicate that an approximation to a structural mode is to follow and the comment "CONTROL" to indicate data for a control mode is to follow. The data record following this comment record contains the order of the aerodynamics approximation NP in an I5 format. UNAERO reads the coefficients of the approximation from the next data record or records with the following statements:

IF(NP.NE.O)READ(JD,IFORM)(R(I),I=1,NP) READ(JD,IFORM)((P(J,K),K=1,NPP3), J=1,NM)

where NP and NM are defined above and NPP3 = NP + 3. If the order of the approximation is zero (NP=0), the R array is not present in the data base. The polynomial defined with the R coefficients of the R array will be different depending on the order of the approximation NP. These polynomials are defined in table 1. The coefficients of the P array are used to form NM polynomials of order NP + 2 with the coefficients of the polynomial being assigned in decreasing powers of reduced frequency.

(4) This section defines the approximation for the gust. Following a comment record, the first data record contains the number of coefficients in the denominator NP1 and in the numerator NP2 in a

10

format of 2I5. The quantity NP1 is the order of the denominator polynomial, whereas NP2 is the order of the numerator polynomial plus one. If NP1 is not zero, the coefficients of the polynomial are read by

READ(JD,IFORM)(R(I),I=1,NP1)

The coefficients of the denominator, in descending order, follow and are read by

READ(JD,IFORM)((P(J,K),K=1,NP2), J=1,NM)

For the example in figure 6, the unsteady-aerodynamics effects of the random gust were approximated by a constant. Therefore the values of the corresponding R vector are not present and there are five elements of the P matrix. (See eq. (4).)

(5) This section defines the generalized mass data. Following a comment record, UNAERO reads the generalized mass matrix with

READ(JD,IFORM)((GM(I,J),J=1,NM), I=1,NM)

(6) This section describes the generalized damping vector. Following a comment record, UNAERO reads NM damping coefficients from the data base with the format statement stored in the variable IFORM.

(7) The natural frequencies (in hertz) for each of the structural modes are the final items. Following a comment record, UNAERO reads NM data elements according to the format statement stored in the variable IFORM.

Additional data records can be appended to the data base as required by the user-defined subroutine SENSOR. If the data base is generated by UNAERO. rather than from an external source, these additional data records are read by the user-defined subroutine PRESEN. See appendix A for an example of using the subroutines PRESEN and SENSOR.

7 UNAERO Output

UNAERO prints selected output and returns the system quadruple to the calling program through the calling sequence. The user can direct UNAERO to list the command file, the calculated parameters of the aerodynamics approximation, the original and the estimate of the data, the error of the approximation, the value of the objective function, the coefficients of the system equations at the specified set point, and diagnostic information.

Figure 7 contains a sample of the printed output from UNAERO with IPRINT set equal to 4 by the calling program. Table 2 contains the relationship between the value of IPRINT and the resulting output. In the following list, each numbered item in the list corresponds to the circled number in figure 7. (If IPRINT had been set to 2 rather than to 4, then item 1 would be the only item printed.)

1. The command input file.

2. The calculated coefficients of the aerodynamic approximation. For the results printed in this example the approximation method is the Padé method. Therefore the P and R matrices are printed. If a least-squares method is used, then the R matrix is suppressed.

3. Indicates that a comparison between the original data and the calculated approximations for the indicated mode follows. The first number is the mode number in the continuous-frequency data base at which the calculated data are stored. The number in parentheses is the mode in the discrete-frequency data base from which the coefficients of the approximation are obtained, and it corresponds to a column of the generalized force matrix.

4. Indicates that the following information, items 5 to 10, are the data for the approximation of the listed mode number in the continuous-frequency data base. This corresponds to a row element of the generalized force matrix.

5. The frequencies at which calculations (listed to the right) were made.

6. The data from the discrete data base that were used in the approximation.

7. The calculated approximations at the indicated value of reduced frequency.

8. The absolute error between the original data and the approximations.

9. The relative error between the original data and the calculated approximations, expressed as a percentage of the original data.

10. The square root of the weight that was applied to the squared error in the objective function.

11. Items 4 to 10 represent the printed analysis of the approximations to the generalized aerodynamic forces that are due to motion of the mode indicated in item 3. Items 4 to 10 are repeated for each mode in the system being generated that the generalized force acts upon. In this example there are two sets printed, one for each of the modes in the system being generated.

12. The maximum absolute and relative errors for the approximation listed at the end of the analysis along with the value of the objective function, identified as "COST" in the sample printout. The

lower the objective function, the more accurate the fit.

13. The system matrices at the requested set point printed at the end of the comparison of the data and the approximation.

If this example were complete, UNAERO would repeat items 2 to 12 for each approximation function being generated. In this example there are three approximation denominator polynomials, one for each mode of the model and one for the control force. Only the data from the first approximation appear in figure 7.

For the convenience of the user, the printed output has been grouped into three sets chosen to coincide with the three intended uses of UNAERO: (1) initial calculation of the aerodynamics approximation, (2) model generation with or without printed results, and (3) program installation and checkout. When UNAERO is used to calculate an unsteady-aerodynamics model, the accuracy of the approximation needs to be monitored while the parameters of the approximation are adjusted accordingly. In this application, output set 1 should be chosen. After the user is satisfied with the accuracy of the approximation, the printed output of UNAERO can be reduced to a set that consists only of the command input file. This use of UNAERO can arise when systems are generated by varying parameters of the equations that do not affect the approximation accuracy, such as the dynamic pressure or the position of a sensor. An option of suppressing all the output exists, such as when UNAERO is an intermediate step of another algorithm in which the output of UNAERO would not be appropriate. The final output option (set 3) consists of two print levels that are designed for program installation and checkout. The choice of these options requires the user to be familiar with the program modules in order for the printed information to be of value. Therefore, set 3 should only be used by a user that is installing UNAERO and needs to check an intermediate calculation or result or both. The set to be printed is controlled by the value of the variable IPRINT and is written to the file whose input/output unit designator is contained in the variable IW. The file that is generated on this unit is intended to be routed to a printer with the capability of printing records that have a length of at least 130 characters.

A complete listing with IPRINT equal to 7 is present in the microfiche available from the author. The information on the microfiche is written in columns, with 15 pages per column. The first page of the first column is labeled A01 and the last page of the first column is labeled O01. The first page of the second column is labeled A02. The following list itemizes the information contained on the microfiche pages.

8 UNAERO Structure and Execution

In this section the interrelation and calling procedures for the subroutines used within UNAERO are discussed. For those users who only require the information necessary to use UNAERO, this information is presented first. For those users who will install and maintain UNAERO, each major subroutine of the software package is discussed in section 8.2. For each subroutine a brief description of the function or subroutine is given, the calling sequence is defined, and the argument list is described.

8.1 UNAERO Execution

Since UNAERO is a package of FORTRAN subroutines, the user must provide a program that initializes the input/output routines, initializes the variables of the labeled common block INOUT, and calls the subroutine UNAERO. The program listed in figure 2 is an example of a calling program which uses

the UNAERO package of subroutines. The subroutine UNAERO will expect to read the command input file and data bases from the unit designators stored in the variables IR and JD of the labeled common block INOUT. On return the system matrices that are in the calling sequence will contain the desired system as specified by the command input file.

There are two characteristics of the UNAERO subroutines that the user must be aware of in order to use the subroutines. These characteristics are (1) the manner in which the dynamic data storage is allocated and (2) the structure of the file for the data base containing the description of the vehicle that is being investigated.

The data storage allocation method is similar to the method described by Chung-Phillips and Rosen in reference 4. Before each subroutine is called, the dynamic data storage needed for that subroutine is calculated and checked against the amount of storage available, which is the length of the data stack. In figure 2, the dynamic data storage is obtained from the array named STACK and the length of the data stack is in the variable named SL. If there is enough space in the array STACK for the required storage, it is partitioned off for the subroutine and the length of the stack is adjusted accordingly and passed to the subroutine being called. If the subroutine being called also calls another subroutine that requires dynamic data storage, an identical procedure is followed for the subsequently called subroutine. On completion of the subroutine, the length of the stack is adjusted so the data storage can be used by other subroutines. Since UNAERO has a large number of options that affect the amount of dynamic data storage required, the storage requirements have never been explicitly calculated. The approach that has been adopted is to set the length of STACK to a value based on experience and run the program. If the problem being analyzed requires more than the available dynamic storage, the execution of the problem is terminated with the appropriate error message. The user then adjusts the length of STACK and reruns the program. Using the dynamic storage approach allows a large aeroelastic problem to be analyzed with a modest amount of space. For example, all the examples that are presented in this report were run with a stack length of 4000 memory locations.

The other characteristic of UNAERO the user must be aware of is the manner in which UNAERO handles the discrete and continuous data bases. These files are both stored in the file that has its unit designator stored in the variable JD. If the user is generating the continuous-frequency data base, UNAERO will destroy the discrete-frequency data base and replace it with the continuous data

base being generated. It is the responsibility of the user to make sure that the continuous-frequency data base is not destroyed when the user saves the discrete-frequency data base. On the CDC Network Operating System (NOS) this is accomplished by the following:

1. Use an indirect access file to store the continuous and discrete data bases.

2. After the program has executed and the user desires to retain the continuous-frequency data base, store the indirect access file containing the data base under a different name on the permanent storage facility.

The example in figure 2 represents the simplest calling sequence of the UNAERO subroutine that will calculate a model of the unsteady aerodynamic forces. UNAERO requires five data files that are designated on the PROGRAM card as INPUT, OUTPUT, VDATA, SCR1, and SCR2. The command input file is assigned to the INPUT file, the system output file is assigned to the OUTPUT file, and the data bases are assigned to the VDATA file. The files SCR1 and SCR2 are scratch files for temporary storage. The unit designators for these files are passed to UNAERO by the variables IR, IW, JD, JS, and JS1 in the labeled common block INOUT. The maximum dimensions of the system matrices with the variable names A, B, C, and D are passed to UNAERO in the variables NXM, NRM, and NUM. The print level and the dynamic storage size are passed to UNAERO with the variables IPRINT and SL. On completion of the calculations, UNAERO returns the coefficients of the model of the aeroelastic vehicle, as defined by equation (25), in the matrices A, B, C, and D. UNAERO returns the actual dimensions of the system quadruple as the number of state equations defining the model in NX, as the number of outputs in NR, and as the number of inputs in NU.

In the remainder of this section, the calling sequence and parameters of the subroutine UNAERO and the variables in the labeled common block INOUT are listed and their key characteristics are discussed.

Calling sequence:

CALL UNAERO(A,B,C,D,NXM,NRM,NUM,WK1,
　　　　　　　STACK,SL,NX,NR,NU)

Argument list:

A　　　　　System-matrix dimensioned (NXM ×
　　　　　　NXM) output (see eq. (25))

B　　　　　Control-input-matrix dimensioned
　　　　　　(NXM × NUM) output (see eq. (25))

C	Measurement-matrix dimensioned (NRM × NXM) output (see eq. (25))
D	Control-response-matrix dimensioned (NRM × NUM) output (see eq. (25))
NXM	Maximum number of states in the system quadruple (input); number of rows and columns of A as dimensioned in the calling program
NRM	Maximum number of outputs in the system quadruple (input); number of rows of C and D as dimensioned in the calling program
NUM	Maximum number of inputs in the system quadruple (input); number of columns of B and D as dimensioned in the calling program
WK1	Work array of length NXM
STACK	Work array of length SL (see discussion on dynamic storage)
SL	Length of array STACK (input) (see discussion on dynamic storage)
NX	Actual number of states in the system quadruple (output)
NR	Actual number of outputs in the system quadruple (output)
NU	Actual number of inputs in the system quadruple (output)

Labeled common block INOUT:

COMMON/INOUT IR,IW,JD,JS,IPRINT,JS1

Variable list:

IR	Unit designator for command input file
IW	Unit designator for print file
JD	Unit designator for data base file
JS	Unit designator for scratch file
IPRINT	Control variable for print file (see table 2 for options)
JS1	Unit designator for a second scratch file

8.2 Description of Subroutines and Functions

In this section a description of each subroutine is given. This section is intended primarily for those who are installing and maintaining UNAERO. The information in this section can be skipped if the user is interested only in executing the program. The structure and hierarchy of the subroutines used in UNAERO are given in figure 8. A horizontal line between two subroutine names indicates that the subroutines on the left call the subroutines on the right. In the sections that follow, each of the subroutines and functions in figure 8 is listed in alphabetical order with its usage, purpose, and a description of its argument list. Exceptions are subroutines PRESEN, SENSOR, USER, and CONMIN. Descriptions of PRESEN and SENSOR can be found in appendix A, and an example of subroutine USER can be found in appendix B. The subroutine CONMIN is described in detail in reference 5. The subroutine SNVDEC is not part of this development and is obtained from reference 6, but it is included in this section for completeness.

8.2.1 Description of global variables. To reduce the space used for listing the descriptions of the argument lists, the following variable list contains the description of the variables that are common to several subroutines. The entries for these variables have been omitted from the individual subroutine argument lists. A variable can be assumed to be an input variable unless otherwise noted.

APLUS	$A^\dagger$ matrix of equation (22) and the solution matrix of equation (16) (dimensioned NPPT × NM). APLUS is an output matrix for subroutines CALAP and SNVDEC and a work matrix for subroutines LSTSOL, PADSOL, and PRTDIF.
CBAR	Reference length used to convert frequency to nondimensional form
IFS	Array of length NRF defining the index to the array RNF. IFS is a work array of subroutine CALPR.
NF	Number of structural modes in the model
NM	= 2(NRF)
NP	Order of the approximation (i.e., the degree of the denominator polynomial of the unsteady-aerodynamics approximation)
NPPT	Order of the numerator polynomial in the approximation
NRF	Number of reduced frequency values

14

P   Vector of length NPPT containing the coefficients of the numerator polynomial approximation. P is an output matrix of subroutines CALPJ and LSTSOL.

Q   Complex matrix (dimensioned NF × NRF) containing the unsteady-aerodynamics data being approximated. Q is an output matrix of subroutine PRTDIF.

QHAT   Complex matrix (dimensioned NF × NRF) containing the estimate of the unsteady aerodynamic forces

QSNV   Work array of length NPPT

R   Vector of length NP containing the coefficients of the denominator polynomial. In the subroutine PADSOL this vector contains the initial value used for the denominator coefficients of the approximation. When subroutine PADSOL returns to the calling program, this vector will contain the calculated values of the Padé approximation. R is a work array for the subroutine CALPR.

RNF   Array of length NRF containing the values of reduced frequency. RNF is a scalar containing a value of reduced frequency in the functions EVALP, EVALR, EVALRR, PHID, and PHIVK.

SIGMA   Root-mean-square value of the gust (eqs. (23) and (24))

SCALE   Characteristic length of the gust (eqs. (23) and (24))

VON,DRY   Logical variables specifying which scaling routine is called for in the gust attenuation option. The routine called is defined by the following:

VON	DRY	Routine called
.T.	.T.	None
.T.	.F.	PHIVK
.F.	.T.	PHID
.F.	.F.	USER

VSNV   Work array (dimensioned NPPT × NPPT)

WK   Work array of length NF

WK1   Work array of length NPPT

WTF   Array of weights (dimensioned NRF)

Z   Complex matrix (dimensioned NF × NRF) containing the error between the aerodynamics data and approximation (see eq. (17)). Z is an output matrix of subroutine CALZFQ and a work matrix otherwise.

· **8.2.2 Subroutine CALAP.** Subroutine CALAP calculates the solution to the least-squares optimization problem described by equation (16). The subroutine forms the matrix A using the selected reduced frequencies and calls subroutine SNVDEC to calculate the pseudoinverse of A.

Calling sequence:

CALL CALAP(A,NM,NPPT,NRF,WTF,IFS,RNF,
          APLUS,QSNV,VSNV)

Argument list:

A   Work array (dimensioned NM × NPPT)

8.2.3 Subroutine CALGRA. Subroutine CALGRA calculates the gradient of the objective function with respect to the coefficients of the denominator polynomial. The ith element of the vector G is the gradient of the ith coefficient of the R polynomial in table 1 and is calculated with equations (20) to (22).

Calling sequence:

CALL CALGRA(G,Z,QHAT,APLUS,WK,WK1,R,
           IFS,RNF,Q,WTF,NP,NPPT,
           NRF,NM,NF)

Argument list:

G   Output vector of length NP containing the gradient of the objective function

8.2.4 Subroutine CALPJ. This subroutine calculates the numerator coefficients of the approximation by using a least-squares method. This is accomplished by multiplying the pseudoinverse of the matrix A (see eq. (16)) by the appropriate vector to obtain the solution of the minimization of equation (16). The solution is returned in the vector P.

Calling sequence:

CALL CALPJ(Q,WTF,WK,R,RNF,IFS,APLUS,P,J,
          NP,NF,NRF,NPPT,NM)

15

Argument list:

J   Integer indicating the structural mode number for which the approximation is being calculated

8.2.5 Subroutine CALPR. The subroutine CALPR calculates the parameters of the unsteady-aero-dynamics approximations for a column of the generalized aerodynamic-force matrix and selected values of the reduced frequency. These parameters are calculated with the least-squares method or the Padé method and are written to the VDATA file.

Calling sequence:

CALL CALPR(MS,CS,GS,NF,NC,NG,NFM,NCM,
        NGM,DUM,M1M,M2M,RNF,
        NRFM,IRNF,BUFF,BUF,
        R,WTF,IFS,Q,X,XMAX,CBAR)

Argument list:

MS   Structural mode selection array of length NF

CS   Control mode selection array of length NC

GS   Gust mode selection array of length NG

NC   Number of control modes in the model

NG   Number of gust modes in the model

NFM   Number of structural modes in the discrete data base

NCM   Number of control modes in the discrete data base

NGM   Number of gust modes in the discrete data base

DUM   Work array (dimensioned M1M × M2M)

M1M   = NFM + NCM

M2M   = NFM + NCM + NGM

NRFM   Number of reduced-frequency values in the discrete data base

IRNF   Work array of length NRFM

BUFF   Input data buffer of length 8

BUF   Input data buffer of length 20

X   Stack array

XMAX   Length of stack array

8.2.6 Subroutine CALZFQ. This subroutine calculates the estimate of the unsteady-aerodynamic-force approximation function ($\widehat{Q}$ in eq. (18)), the error vector between the data and the approximation (z_m in eq. (18)), and the value of the objective function that is being minimized (J_j in eq. (19)).

Calling sequence:

CALL CALZFQ(QHAT,Q,R,APLUS,WK,WK1,
        NF,NRF,NP,NPPT,NM,Z,F,
        IFS,RNF,WTF)

Argument list:

F   Output variable F containing the value of the objective function

8.2.7 Subroutine ERMON. This subroutine is called when an error in the program is detected. The subroutine prints the appropriate error message and returns if there is a valid way of continuing and generating useful data. Otherwise the subroutine terminates the program execution with a CDC NOS return code of 105.

Calling sequence:

CALL ERMON(ERRNO,PAGE,BUFF)

Argument list:

ERRNO   Error number being printed (input)

PAGE   Logical input variable that, if true, will print the error message at the top of the next page and terminate the program

BUFF   Input string of up to 80 characters used to pass a message to the subroutine for printing along with the error message

8.2.8 Subroutine ERR. Subroutine ERR is called by the CDC CYBER NOS monitor whenever an illegal-data-in-field or a data-overflow error is detected during execution. If the error number passed to ERR in the labeled common block ERROR is not zero, that error number is passed to the subroutine ERMON as the argument ERRNO, which generates the appropriate error message. This subroutine is used to trap illegal syntax usage in the command. For the cases where the command file input is not being decoded, the error flag is set to zero so that normal detection and processing of data overflow can occur.

8.2.9 Subroutine ERRMON. The subroutine ERRMON handles the special error conditions that

arise when the maximum system dimensions NXM, NRM, and NUM (i.e., the dimensions of the matrices with the variable names A, B, C, and D in the argument list of the subroutine UNAERO) are exceeded. When this subroutine is called, the requested system dimensions and the maximum system dimensions are printed and the program is terminated with a CYBER NOS return code of 101.

Calling sequence:

CALL ERRMON(NX,NC,NR,NXM,NCM,NRM)

Argument list:

NX	Number of system states being requested
NC	Number of control states being requested
NR	Number of output states being requested
NXM	Maximum number of system states
NCM	Maximum number of control states
NRM	Maximum number of output states

8.2.10 Function EVALP. The function EVALP evaluates the numerator polynomial of the unsteady-aerodynamics approximation at the reduced frequency RNF.

Calling sequence:

X=EVALP(P,RNF,NPPT)

8.2.11 Function EVALR. The function EVALR evaluates the denominator polynomial of the unsteady-aerodynamics approximation at the reduced frequency RNF.

Calling sequence:

X=EVALR(R,RNF,NP)

8.2.12 Function EVALRR. The function EVALRR calculates the value of the partial derivative of the denominator polynomial R with respect to the kth coefficient of the polynomial r (see eq. (21)) evaluated at a reduced-frequency value of RNF.

Calling sequence:

X=EVALRR(R,K,RNF,NP)

Argument list:

| K | Integer indicating the parameter that the partial derivative is to be taken to |

8.2.13 Subroutine FABCD. Subroutine FABCD forms the system quadruple, matrices A, B, C, and D, from the data stored in the continuous-frequency data base.

Calling sequence:

CALL FABCD(A,B,C,D,XN1,XN2,NXM,NRM,
NUM,CBAR,V,QBAR,NF,NC,
NG,NS,NGF,XN3,XN4,XN5,
NGFA,XN6,XN7,XN8,XN9,
XN10,XN11,XN12,XN13,
XN14,NX,NR,NU)

Argument list:

A	System matrix defined by equation (25) (dimensioned NXM × NXM)
B	System matrix defined by equation (25) (dimensioned NXM × NUM)
C	System matrix defined by equation (25) (dimensioned NRM × NXM)
D	System matrix defined by equation (25) (dimensioned NRM × NUM)
XN1,XN9, XN11	Work arrays (dimensioned NXM × NXM)
XN2	Work array (dimensioned NRM × NXM)
NXM	Maximum number of states that the system matrices can accommodate, as defined by the calling program
NRM	Maximum number of outputs that the system matrices can accommodate, as defined by the calling program
NUM	Maximum number of inputs that the system matrices can accommodate, as defined by the calling program
V	Reference velocity
QBAR	Reference dynamic pressure
NC	Number of control modes in the model

17

NG	Number of gust modes in the model
NS	Number of system outputs
NGF	Number of generalized forces to be retained
XN3	Work array of length 4
XN4	Work array (dimensioned NF × 7)
XN5	Work array (dimensioned NF × 8)
NGFA	Vector of length NGF containing an index as to which generalized forces are to be retained in the system being formed
XN6,XN7, XN8	Work arrays (dimensioned NF × NF)
XN10	Work array of length NXM
XN12,XN13, XN14	Work array (dimensioned NS × NF)
NX	Number of system states generated by this subroutine
NR	Number of system outputs generated by this subroutine
NU	Number of system inputs generated by this subroutine

8.2.14 Subroutine FDATA. The subroutine FDATA reads and modifies the data from the discrete-frequency data base and returns the data in the complex matrix Q of equation (18) according to the mode selection array and frequency selection array in the argument list. Before returning to the calling program the data are attenuated if the user has so specified. specified.

Calling sequence:

CALL FDATA(DUM,M1M,M2M,Q,NF,NRF,IFS,
          MS,IM,NRFM,IFORM1,SIGMA,
          SCALE,RNF,CBAR,VON,DRY)

Argument list:

DUM	Work array (dimensioned M1M × M2M)
M1M	= NF + NC, input as defined in subroutine CALPR
M2M	= NF + NC + NG, input as defined in subroutine CALPR
MS	Mode selection array
IM	Mode for which data are being requested

NRFM	Number of reduced-frequency values in the discrete-frequency data base
IFORM1	Format for reading data from the discrete-frequency data base (VDATA)

8.2.15 Subroutine LSTSOL. Subroutine LSTSOL calculates the least-squares solution for the numerator coefficients of the unsteady-aerodynamics approximation and writes the results in the continuous-frequency data base.

Calling sequence:

CALL LSTSOL(A,B,NM,NRF,Q,NF,WTF,IFS,
            RNF,R,NP,P,NPPT,QSNV,
            VSNV,APLUS,IFORM)

Argument list:

A	Work array (dimensioned NM × NPPT)
B	Work array of length NM
IFORM	Format for writing the approximation to the continuous-frequency data base

8.2.16 Subroutine PADSOL. The subroutine PADSOL calculates the Padé coefficients of the unsteady-aerodynamics approximation and writes the results in the continuous-frequency data base.

Calling sequence:

CALL PADSOL(R,G,VLB,VUB,SCAL,DF,AX,S,
            G1,G2,B,C,ISC,IC,MS1X,N1P,
            N2P,N3P,N4P,N5P,NP,Q,
            APLUS,Z,QHAT,WK,WK1,
            IFS,RNF,WTF,NPPT,NRF,NM,
            NF,IFORM,VSNV,A,AT)

Argument list:

G,VLB, VUB,SCAL, DF,S	Work arrays of length N1P used by the subroutine CONMIN
AX	Work array (dimensioned N1P × N3P) used by the subroutine CONMIN
G1,G2,ISC	Work arrays of length N2P used by the subroutine CONMIN
B	Work array (dimensioned N3P × N3P) used by the subroutine CONMIN
C	Work array of length N4P used by the subroutine CONMIN

IC	Work array of length N3P used by the subroutine CONMIN
MS1X	Work array of length N5P used by the subroutine CONMIN
N1P	$= NP + 2$
N2P	$= 2(NP)$
N3P	$= NP + 1$
N4P	$= NP + 1$
N5P	$= 2(NP + 1)$
IFORM	Format for writing the calculated approximation to the continuous-frequency data base
A	Work array (dimensioned NM × NPPT)
AT	Work array (dimensioned NF × NPPT); may be made equivalent to the matrix A in the calling program

8.2.17 Function PHID.
The function PHID returns the attenuation factor for a Dryden turbulence spectrum (eq. (24)) at the spectral frequency that is equivalent to the reduced frequency specified by the parameter RNF.

Calling sequence:

X=PHID(SIGMA,SCALE,RNF,CBAR)

8.2.18 Function PHIVK.
The function PHIVK returns the attenuation factor for a Von Karman turbulence spectrum (eq. (23)) at the spectral frequency that is equivalent to the reduced frequency specified by the parameter RNF.

Calling sequence:

X=PHIVK(SIGMA,SCALE,RNF,CBAR)

8.2.19 Subroutine PRNT.
The subroutine PRNT is a matrix utility that prints large matrices on the output device designated by the variable IW in the labeled common block INOUT. The subroutine divides the large matrix into submatrices that will fit on an 8.5 × 17-in. page. Each of the submatrices is then printed.

Calling sequence:

CALL PRNT(AR,ND,NR,NC)

Argument list:

AR	Input matrix to be printed (dimensioned ND × NC)
ND	First dimension of NR in the calling program
NR	Number of rows to be printed
NC	Number of columns to be printed

8.2.20 Subroutine PRTDIF.
The subroutine PRTDIF prints program output as described in section 7.

Calling sequence:

CALL PRTDIF(QHAT,Q,R,APLUS,WK,WK1,
NF,NRF,NP,NPPT,NM,Z,F,IFS,
RNF,WTF,IMX,IM)

Argument list:

F	On return, contains the value of the objective function for the given parameters
IMX	Mode number in the discrete-frequency data base
IM	Mode number in the continuous-frequency data base

8.2.21 Subroutine REFORM.
The subroutine REFORM converts the coefficients of an unsteady-aerodynamics approximation from the form that is used in the calculations (eqs. (3) and (4)) into the form used by the subroutine FABCD (eq. (5)).

Calling sequence:

CALL REFORM(P,R,A2,A1,A0,D0,D1,D2,D3,W,
NP,NF,CBAR,V)

Argument list:

A2,A1,A0, D0, D1, D2,D3	Work array of length NF
W	Work array of length NP
V	Reference velocity

8.2.22 Subroutine SNVDEC.
The subroutine SNVDEC calculates the pseudoinverse of a matrix. Additional information can be found in reference 6.

Calling sequence:

CALL SNVDEC(IOP,MD,ND,M,N,A,NOS,B,IAC,
ZTEST,Q,V,IRANK,APLUS,IERR)

Argument list:

IOP   Set equal to 4 for pseudoinverse option

MD   Input integer specifying the maximum row dimension for A

ND   Input integer specifying the maximum row dimension for V

M   Input integer specifying the number of rows in A

N   Input integer specifying the number of columns in A

A   Input matrix, destroyed by the program

NOS   Number of columns in B. The value of zero is used by the calling program.

B   Dummy argument

IAC   Input integer specifying the number of decimal digits of accuracy in the elements of the matrix A

ZTEST   Output variable used by SNVDEC to test for zero

Q   Vector of length N which will contain the singular values on return from the subroutine

V   Work array (dimensioned ND × N)

IRANK   Rank of the matrix A

IERR   Returned error indicator, as follows:

    = 0, no error

    = K, Kth singular value not found after 30 iterations

    = −1, this matrix is close to a matrix which is of lower rank than IRANK

8.2.23 Subroutine ZZLINE. The subroutine ZZLINE formats the printed output into pages of 44 lines each. If the number of lines N being added to the output file is greater than 44, a page eject command is issued and the line counter is set to N. If N is greater than 44, then the line counter is set to zero. The unit designator for the output is contained in the variable IW of the labeled common block INOUT.

20

Calling sequence:

CALL ZZLINE(N)

Argument list:

N   Number of lines to be added to the current page

8.2.24 Subroutine ZZPAGE. The subroutine ZZPAGE is used with the subroutine ZZLINE to return 44, the number of lines printed since the last page eject.

Calling sequence:

CALL ZZPAGE(N)

Argument list:

N   Number of lines left on the page

9 Examples of UNAERO Use

Several examples are presented which illustrate the options that are available in the program and which show the input used to invoke these options. The output listings that are produced by running the program with these input are omitted. This output is omitted in the interest of saving space and has the same form as described in section 7, but it has different data printed for the results. All the examples presented use the same task I discrete-frequency data base as listed in figure 5. This data base is described in the next paragraph followed by a discussion of all the examples.

The discrete-frequency data bases for the examples described are based on the aeroelastic semispan wind tunnel model (ARW-1) that was used for the example in reference 1. The wing geometry is presented in figure 9 and the generalized masses and frequencies which describe the aeroelastic model are presented in table 3. Structural damping is assumed to be negligible. The simple harmonic oscillatory aerodynamic forces are calculated with a doublet-lattice technique similar to that described in reference 7. In order to calculate the pressure distribution on an oscillating wing undergoing simple harmonic motion, the lifting surface is subdivided into an array of trapezoidal boxes arranged in strips parallel to the air stream, as shown in figure 10. The lifting surface is then represented by a lattice of doublets located along the quarter-chord of each box. The downwash condition is satisfied at the three-quarter-chord of each box by equating it to the downwash resulting from the slope and deflection rate of each structural mode. The semispan lifting surface is divided into 210 boxes arranged in 30 spanwise strips of 7 chordwise boxes.

The side-edge effects of the control surface (as described in ref. 7) are not applied. Oscillatory aerodynamic forces are calculated at eight reduced frequencies (0, 0.1, 0.3, 0.5, 0.7, 0.9, 1.3, and 1.8). The structural model consists of 10 structural modes, 1 control-surface mode, and 1 gust mode.

9.1 Example 1

The command input file for example 1 is presented in figure 1(a). In this example a first-order Padé approximation is used for 5 of the 10 structural modes that are stored in the discrete data base. The modes that are retained are 1, 2, 4, 5, and 6. (For the ARW-1 wind tunnel model, the third mode is mainly motion fore and aft in the plane of the wing and contributes little to the dynamic behavior of the wing under investigation.) Since the gust input mode is not needed for the flutter calculation, it is also deleted. This illustrates how to delete a set of modes from the data being used for the calculation of the approximation. Each approximation is based on data calculated at the first six values of reduced frequency. The last six lines of figure 1(a) represent a second call to subroutine UNAERO. In this call, UNAERO uses the stored data in the continuous-frequency data base and calculates a system quadruple (i.e., the matrices A, B, C, and D) at a new set point with the dynamic pressure increased by a factor of two.

9.2 Example 2

The command input file for example 2 is presented in figure 1(b). This example is identical to the first example except that now second-order Padé approximations are being calculated. The initial value of the denominator coefficients is based on the results from example 1. One initial value is set to a fraction of the result generated from example 1 and the other initial value is set to a multiple of that result. For this example the initial values are 1/2 and 2 times the result of example 1.

9.3 Example 3

Example 3, presented in figure 1(c), is also a derivative of example 1, except that here fourth-order least-squares approximations are used. The least-squares approximations are used in this case after the denominator coefficients are determined with the Padé approximation.

Using the least-squares approximations with the results from the Padé approximations effects a savings in computer resources while giving identical results. This is often done while parametric studies are performed that do not influence the aerodynamics approximation calculations.

9.4 Example 4

Figure 1(d) is the command input file which illustrates the generation of a system quadruple with a weighted least-squares approximation. The system generated is of "minimum order" in that the zero-order least-squares approximations induce no additional states into the system to model the effects of the unsteady aerodynamics. This approximation function results in the largest errors of any of the approximation functions that UNAERO can generate. With proper selection of the weights, however, useful results can be obtained. In fact, this example predicts the flutter dynamic pressure to the same degree of accuracy as does the second-order Padé approximation, since a priori knowledge was used in the selection of the weight so that the minimum error occurs at the reduced frequency of flutter. The weights are selected to be inversely proportional to the difference between the reduced frequency of the data and the reduced frequency of flutter. Note that the weights for the data at the seventh and eighth values of reduced frequency have been left blank, thereby resulting in a zero for the weight at those frequencies. Using a weight of zero requires more computer resources than does removing that value of reduced frequency from the data base before the approximation is calculated. In this example, the frequencies were retained, since it was desired to observe the calculated value of the approximation at these reduced frequencies.

9.5 Example 5

The command input file for this example is presented in figure 1(e). In this example, two structural modes (modes 1 and 2) and the gust input are retained whereas the control-surface mode is removed. The approximations to the unsteady aerodynamic forces due to motion in the first mode are fourth-order Padé approximation functions based on the first six frequencies in the discrete data base. The unsteady-aerodynamic-force approximations corresponding to the second mode are fourth-order least-squares approximations, but these approximations are based on all the frequencies in the discrete data base. The gust input is approximated by Padé approximations with second-order numerators and third-order denominators. These approximations are based on data at three values of reduced frequency (0, 0.3, and 0.9) and are attenuated with a Von Karman gust spectrum with a characteristic length of 1200 in. and a vertical velocity of 12 in/sec. Two sensors are to be added to the system equations with the user-supplied PRESEN and SENSOR subroutines. An explanation of these two subroutines is

given in appendix A. The subroutine SENSOR reads the data for the location of the sensors from the two cards after the "ADD 00002 SENSORS AT" card and provides UNAERO with the required information to calculate the sensor equations. The reader may refer to appendix A for any further discussion.

10 Concluding Remarks

UNAERO is a collection of FORTRAN IV subroutines for approximating the unsteady aerodynamic forces in the time domain and obtaining a set of constant-coefficient first-order differential equations. This report serves as an instruction and maintenance manual for UNAERO. These subroutines are intended to be run on CDC CYBER computer systems.

UNAERO has provisions for adjusting the number of modes used for calculating the approximations so that an accurate approximation is generated. The number of data points at different values of reduced frequency can also be varied to adjust the accuracy of the approximation over the reduced-frequency range. The denominator coefficients of the approximation may be calculated by means of a gradient method or may be entered into a least-squares approximation technique. Both approximation methods use weights on the residual error. UNAERO uses data bases to store the coefficients of the approximation. In doing so, a new set of system equations, at a different dynamic pressure, can be generated without the approximations being recalculated.

Appendix A

Sample Subroutines PRESEN and SENSOR

The mode shapes that are used to describe the dynamics of the vehicle are normally provided as a set of modal displacements at specific points on the vehicle. The generation of the sensor equations that define the output of a sensor (at an arbitrary location on the surface of the vehicle) requires that the value of the displacement for each mode be calculated at the location of the sensor. This calculation requires the use of an interpolation method for the modes and can depend on the structural model used in generating the modes. The inclusion of all known useful interpolation schemes was beyond the scope of this work. In order to have the ability to add sensor equations to the system equations and to meet the individual needs of the users, the section of the program which performs the modal interpolation is left for the user to provide.

UNAERO assumes that the output of a sensor can be modeled as a linear combination of the acceleration, the velocity, and the displacement of the generalized coordinates. Thus the sensor output is modeled by the following equation:

$$Y = AP * \ddot{q} + VP * \dot{q} + DP * q \qquad (A1)$$

where Y is a vector of the sensor outputs, q is a vector of the generalized coordinates, and AP, VP, and DP are matrices of the coefficients which relate the generalized coordinate's acceleration, velocity, and displacement to the sensor output. Dots over q denote a derivative with respect to time.

To interface the interpolation method UNAERO requires that the user provide two subroutines. One subroutine, PRESEN, is called during the execution of task I. This subroutine is required to edit and copy the structural modal data, which have been appended by the user to the end of the discrete-frequency data base, in the continuous-frequency data base. The other subroutine, SENSOR, is called during task II. It uses the copied data from the continuous-frequency data base to calculate the required coefficients of the sensor equation.

In order for the subroutine PRESEN to edit the discrete-frequency data base, an array of integers indicating the modes which are to be retained for the approximations is transferred to the subroutine in the variable array MS. Each element of MS is an integer that is a mode number to be retained. The numbers in MS are initialized by the "mode selection array" in figure 4(d). If the user does not reduce the number of modes used for the approximation, the MS array will contain the numbers 1 to NFM. If the user does omit some of the modes, the array will be of length NF and contain the "mode selection array" data that was read by the program from the command file.

The variables that contain the INPUT/OUTPUT unit designators for the continuous- and discrete-frequency data bases are in the labeled common block INOUT. The INPUT/OUTPUT unit designator for the discrete-frequency data base is contained in the variable JS1, and the INPUT/OUTPUT unit designator for the continuous-frequency data base is contained in the variable JD. On entry into the subroutine PRESEN, the discrete data base must be rewound and positioned at the beginning of the modal displacement data.

The calling sequence for the subroutine is

CALL PRESEN(NFM,NF,MS,IFORM,IFORM1)

where

NFM	Number of structural modes in the discrete-frequency data base
NF	Number of structural modes in the continuous-frequency data base
MS	Array of length NF. Each element of MS contains the number of the mode that is to be copied from the discrete data base to the continuous data base.
IFORM	Format for writing data to the continuous data base
IFORM1	Format for reading data from the discrete data base

A sample of the subroutine PRESEN is presented in figure A1.

Subroutine SENSOR is called during task II by UNAERO to calculate the elements of the AP, VP, and DP matrices in equation (A1). Each of these matrices has dimensions of NS × NF, where NS is the number of sensors (as specified in the "ADD SENSORS" command) and NF is the number of modes. For example, if a single accelerometer is added to a system with four modes, the matrices in equation (A1) would have dimensions of one by four in subroutine SENSOR. The subroutine sets each element of the AP matrix to the interpolated modal displacement for each mode. On entry to the subroutine SENSOR, the continuous-frequency data base is positioned to the first card of the augmented data that was written to the data base by the subroutine PRESEN. The INPUT/OUTPUT unit designator for the continuous-frequency data base is contained in the

variable JS in the labeled common block INOUT. The command input file, with the input/output unit designator contained in the variable JD of the labeled common block INOUT, is positioned to the next record after the "ADD SENSOR AT" request. The user can use input following this card to pass data and parameters, such as the sensor position, to the interpolation routine contained in the subroutine SENSOR.

The calling sequence for the subroutine SENSOR is

CALL SENSOR(AP,VP,DP,NS,NF,IFORM)

where

AP   Output matrix of coefficients that relates the output of the sensor to the acceleration of the generalized coordinate

VP   Output matrix of coefficients that relates the output of the sensor to the velocity of the generalized coordinate

DP   Output matrix of coefficients that relates the output of the sensor to the position of the generalized coordinate

NS   Number of sensors being added

NF   Number of generalized coordinates in the system

IFORM   Format for reading the discrete data base

The subroutines PRESEN and SENSOR are required only when the user requests that sensors be added to the equations being generated. If the user makes the request "ADD SENSORS AT" in the command input file without supplying the subroutines PRESEN and SENSOR, an error message is generated.

```
C         SAMPLE PRESEN SUBROUTINE
          SUBROUTINE PRESEN(NFM,NF,MS,IFORM,IFORM1)
          DIMENSION MS(NF),BUF(8),Z(102,10)
          COMMON /INOUT/ IR,IW,JD,JS,IPRINT,JS1
C         REWIND DISCRETE DATA FILE
          REWIND JS1
          HNODE=10HNODES
1         FORMAT(8A10)
2         FORMAT(I5)
3         FORMAT(25HZ - DISPLACEMENT FOR MODE ,I5)
C         SEARCH FOR BEGINNING OF MODEL DATA
100       CONTINUE
          READ(JS1,1) TEMP
          IF(TEMP.EQ.HNODE) GO TO 110
          GO TO 100
110       CONTINUE
C         READ NUMBER OF NODES
          READ(JS1,2) NN
C         READ AND WRITE COMMENT CARD FOR X-NODES
          READ(JS1,1) BUF
          WRITE(JD,1) BUF
C         READ AND WRITE X-NODES
          READ(JS1,IFORM1) (Z(I,1),I=1,102)
          WRITE(JD,IFORM) (Z(I,1),I=1,102)
C         READ AND WRITE COMMENT CARD FOR Y-NODES
          READ(JS1,1) BUF
          WRITE(JD,1) BUF
C         READ AND WRITE Y-NODES
          READ(JS1,IFORM1) (Z(I,1),I=1,102)
          WRITE(JD,IFORM) (Z(I,1),I=1,102)
C         LOOP FOR EACH MODE
          DO 200 I=1,10
C         SKIP HEADER AND READ Z-DISPLACEMENT FOR MODE I
          READ(JS1)
          READ(JS1,IFORM1) (Z(J,I),J=1,102)
200       CONTINUE
C         LOOP FOR EACH OUTPUT MODE
          DO 210 I=1,NF
C         WRITE HEADER
          WRITE(JD,3) I
C         WRITE Z-DISPLACEMENTS FOR SELECTED MODE
          WRITE(JD,IFORM) (Z(J,MS(I)),J=1,102)
210       CONTINUE
          RETURN
          END
```

Figure A1. Sample of PRESEN subroutine.

Appendix B

Subroutine User

It is recognized that the user may wish to apply modifications to the data before the unsteady-aerodynamics approximations are calculated. This subroutine allows the user to apply a change in the magnitude of the data as a function of the frequency. A possible use for this change is to attenuate the data for the gust mode with a gust spectrum which is neither a Von Karman nor a Dryden distribution. This might occur in an experimental test with the random disturbance being artificially generated. Subroutine USER is called instead of the function PHIVK or PHID whenever the "ATTENUATE DATA" card specifies "USER SPECTRUM." (See fig. 4(k).)

The calling sequence for the subroutine USER is

X=USER(SIGMA,SCALE,RNF,CBAR)

The parameters SIGMA, SCALE, and CBAR are read in from the command input file. For this option the parameters SIGMA and SCALE pass to the function USER any data the user may desire. The parameter RNF is the reduced frequency for which the calling program (subroutine FDATA) requests the attenuation factor. The parameter CBAR is the reference length from the input described in figure 4(f). The formula used to attenuate the data is

(NEW DATA) = (OLD DATA)*USER(SIGMA, SCALE,RNF,CBAR)

where (NEW DATA) is the data used to calculate the approximation and (OLD DATA) is the corresponding value in the discrete data base.

References

1. Dunn, H. J.: *An Analytical Technique for Approximating Unsteady Aerodynamics in the Time Domain.* NASA TP-1738, 1980.
2. Adams, William M., Jr.; Tiffany, Sherwood H.; Newsom, Jerry R.; and Peele, Ellwood L.: *STABCAR—A Program for Finding Characteristic Roots of Systems Having Transcendental Stability Matrices.* NASA TP-2165, 1984.
3. Houbolt, John C.; Steiner Roy; and Pratt, Kermit G.: *Dynamic Response of Airplanes to Atmospheric Turbulence Including Flight Data on Input and Response.* NASA TR R-199, 1964.
4. Chung-Phillips, Alice; and Rosen, R. W.: A Note on Dynamic Data Storage in Fortran IV. *Computer J.,* vol. 18, no. 4, Nov. 1975, pp. 342–343.
5. Vanderplaats, Garret N.: *CONMIN—A Fortran Program for Constrained Function Minimization. User's Manual.* NASA TM X-62,282, 1973.
6. Armstrong, Ernest S.: *ORACLS—A System for Linear-Quadratic-Gaussian Control Law Design.* NASA TP-1106, 1978.
7. Albano, Edward; and Rodden, William P.: A Doublet-Lattice Method for Calculating Lift Distributions on Oscillating Surfaces in Subsonic Flows. *AIAA J.,* vol. 7, no. 2, Feb. 1969, pp. 279–285; Errata, vol. 7, no. 11, Nov. 1969, p. 2192.

26

TABLE 1. PARAMETERS OF DENOMINATOR POLYNOMIAL $\mathbf{R}(j\omega)$

Order of approximation	Polynomial used in equation (4)
1	$(jk) + r_1$
2	$(jk)^2 + r_2(jk) + r_1$
3	$[(jk)^2 + r_2(jk) + r_1][(jk) + r_3]$
4	$[(jk)^2 + r_2(jk) + r_1][(jk)^2 + r_4(jk) + r_3]$

TABLE 2. PRINT OPTIONS

IPRINT	Resulting output
0	Print nothing.
2	Print command input file (item 1, fig. 7).
4	Print command input file, parameter of the approximation, original data, approximated data, and error and system matrices (items 1–13, fig. 7).
6	Print diagnostic information plus the information of IPRINT=4.
7	Set CONMIN to print level 6 plus the information of IPRINT=6. (See microfiche output.)

TABLE 3. FREQUENCIES AND GENERALIZED MASSES OF ARW-1 WIND TUNNEL MODEL

Mode	Natural frequency, Hz	Generalized mass, kg
1	5.233	3.678
2	19.129	7.769
3	20.906	7.044
4	25.769	2.970
5	46.110	4.714
6	61.234	4.758
7	79.682	5.156
8	86.030	11.297
9	98.087	7.558
10	118.150	5.501

```
UNSTEADY AERO DATA
REFERENCE LENGTH  25.4
REFERENCE VELOCITY 5400.
REFERENCE DYNAMIC PRESSURE   100.
USE 00005 STRUCTURAL MODES
     1    2    4    5    6
00000 GUST INPUTS
USE A 1 ORDER PADE' APPROXIMATION FOR MODE 1
1.02366
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 1 ORDER PADE' APPROXIMATION FOR MODE 2
1.2135
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 1 ORDER PADE' APPROXIMATION FOR MODE 3
1.1353
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 1 ORDER PADE' APPROXIMATION FOR MODE 4
1.1281
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 1 ORDER PADE' APPROXIMATION FOR MODE 5
1.09640
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 1 ORDER PADE' FOR CONTROL 1
1.3336
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
END OF UNSTEADY AERO DATA
UNSTEADY AERO DATA
USE APPROXIMATION ON VDATA FILE
REFERENCE LENGTH 25.4
REFERENCE VELOCITY 5400.
REFERENCE DYNAMIC PRESSURE 200.
END OF UNSTEADY AERO DATA
```

(a) Command input file for example 1. Key elements are underlined.

Figure 1. Examples of command input files.

```
UNSTEADY AERO DATA
REFERENCE LENGTH  25.4
REFERENCE VELOCITY  5400.
REFERENCE DYNAMIC PRESSURE 100.
USE 00005 STRUCTURAL MODES
    1    2    4    5    6
00000 GUST INPUTS
USE A 2 ORDER PADE' APPROXIMATION FOR MODE 1
.126136      .50226
BASE THE APPROXIMATION ON 00006 FREQUENCIES
    1    2    3    4    5    6
USE A 2 ORDER PADE' APPROXIMATION FOR MODE 2
.04251       .29160
BASE THE APPROXIMATION ON 00006 FREQUENCIES
    1    2    3    4    5    6
USE A 2 ORDER PADE' APPROXIMATION FOR MODE 3
.02042       .20211
BASE THE APPROXIMATION ON 00006 FREQUENCIES
    1    2    3    4    5    6
USE A 2 PADE' ORDER APPROXIMATION FOR MODE 4
.02613       .22861
BASE THE APPROXIMATION ON 00006 FREQUENCIES
    1    2    3    4    5    6
USE A 2 ORDER PADE' APPROXIMATION FOR MODE 5
.03369       .25956
BASE THE APPROXIMATION ON 00006 FREQUENCIES
    1    2    3    4    5    6
USE A 2 ORDER PADE' FOR CONTROL 1
.15572       .55806
BASE THE APPROXIMATION ON 00006 FREQUENCIES
    1    2    3    4    5    6
END OF UNSTEADY AERO DATA
```

(b) Command input file for example 2.

Figure 1. Continued.

```
UNSTEADY AERO DATA
REFERENCE LENGTH   25.4
REFERENCE VELOCITY   5400.
REFERENCE DYNAMIC PRESSURE   100.
USE 00005 STRUCTURAL MODES
     1    2    4    5    6
00000 GUST INPUTS
USE A 4 ORDER LEAST SQUARES APPROXIMATION FOR MODE 1
.23263        1.9581        .50516        .68992
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 4 ORDER LEAST SQUARES APPROXIMATION FOR MODE 2
.34664        3.2804        .15091        .74886
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 4 ORDER LEAST SQUARES APPROXIMATION FOR MODE 3
1.9397        3.1545        .033269       .40409
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 4 ORDER LEAST SQUARES APPROXIMATION FOR MODE 4
1.2788        2.2638        .052972       .48044
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 4 ORDER LEAST SQUARES APPROXIMATION FOR MODE 5
.57787        1.1957        .058564       .49398
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 4 ORDER LEAST SQUARES FOR CONTROL 1
.30149        3.61          1.853         3.61
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
END OF UNSTEADY AERO DATA
```

(c) Command input file for example 3.

Figure 1. Continued.

```
UNSTEADY AERO DATA
REFERENCE LENGTH 25.4
REFERENCE VELOCITY  5400.
REFERENCE DYNAMIC PRESSURE 100.
USE 00004 STRUCTURAL MODES
     1    2    4    5
00000 GUST INPUTS
USE A 0 ORDER LEAST SQUARES APPROXIMATION FOR MODE 1
WEIGHT EACH FREQUENCY WITH
69.44        2500.        ·30.86        6.93        2.97        1.64


USE A 0 ORDER LEAST SQUARES APPROXIMATION FOR MODE 2
WEIGHT EACH FREQUENCY WITH
69.44        2500.        30.86        6.93        2.97        1.64


USE A 0 ORDER LEAST SQUARES APPROXIMATION FOR MODE 3
WEIGHT EACH FREQUENCY WITH
69.44        2500.        30.86        6.93        2.97        1.64


USE A 0 ORDER LEAST SQUARES APPROXIMATION FOR MODE 4
WEIGHT EACH FREQUENCY WITH
69.44        2500.        30.86        6.93        2.97        1.64


USE A 0 ORDER LEAST SQUARES FOR CONTROL 1
WEIGHT EACH FREQUENCY WITH
69.44        2500.        30.86        6.93        2.97        1.64


END OF UNSTEADY DATA
```

(d) Command input file for example 4.

Figure 1. Continued.

```
UNSTEADY AERO DATA
REFERENCE LENGTH 25.4
REFERENCE VELOCITY  5400.
REFERENCE DYNAMIC PRESSURE 100.
USE 00002 STRUCTURAL MODES
     1    2
00000 AERODYNAMIC CONTROLS
USE A 4 ORDER PADE' APPROXIMATION FOR MODE 1
.08          .6            .48          1.6
BASE THE APPROXIMATION ON 00006 FREQUENCIES
     1    2    3    4    5    6
USE A 4 ORDER LEAST SQUARES APPROXIMATION FOR MODE 2
.34664       3.2804        .15091       .74886
USE A 2,3 ORDER PADE' APPROXIMATION FOR GUST 1
1.5          7.5           .1
BASE THE APPROXIMATION ON 00003 FREQUENCIES
     1    3    6
ATTENUATE DATA WITH VON KARMAN SPECTRUM
SCALE 1200.
RMS 12.
ADD 00002 SENSORS AT
     73.1        65.0
     47.8        43.5
END
```

(e) Command input file for example 5.

Figure 1. Concluded.

```
PROGRAM EXAMPLE(INPUT,OUTPUT,VDATA,SCR1,SCR2,
               TAPE5=INPUT,TAPE6=OUTPUT,TAPE10=VDATA,
               TAPE12=SCR1,TAPE13=SCR2)
INTEGER SL
DIMENSION A(32,32),B(32,5),C(7,32),D(7,5),
      STACK(4000),WK(32)
COMMON /INOUT/ IR,IW,JD,JS,IPRINT,JS1
IR=5
IW=6
JD=10
JS=12
JS1=13
NXM=32
NRM=7
NUM=5
IPRINT=4
SL=4000
CALL UNAERO(A,B,C,D,NXM,NRM,NUM,WK,STACK,SL,NX,NR,NU)
STOP
END
```

Figure 2. Sample calling program which uses UNAERO.


```
                    1         2         3         4         5
          1234567890123456789012345678901234567890123456789 0
REFERENCE LENGTH 10.0
REFERENCE VELOCITY 120.
REFERENCE DYNAMIC PRESSURE 100.
USE APPROXIMATION ON VDATA FILE
USE 00002 STRUCTURAL MODES
00001 AERODYNAMIC CONTROLS
00003 GUST INPUTS
USE A 1 ORDER LEAST SQUARES APPROXIMATION
USE A 2 ORDER PADE' APPROXIMATION
USE A 1,2 ORDER LEAST SQUARES APPROXIMATION
USE A 2,3 ORDER PADE' APPROXIMATION
BASE THE APPROXIMATION ON 00004 FREQUENCIES
WEIGHT EACH FREQUENCY WITH
RETAIN 00002 NORMALIZED GENERALIZED FORCES AS OUTPUTS
ADD 00003 SENSORS AT
ATTENUATE DATA WITH VON KARMAN SPECTRUM
ATTENUATE DATA WITH DRYDEN SPECTRUM
ATTENUATE DATA WITH USER SPECTRUM
SCALE 4.
RMS 100.
END
```

Figure 3. UNAERO legal command inputs. Key elements are underlined.
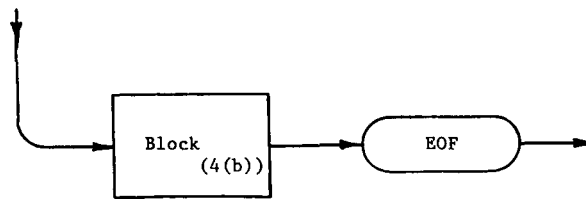
Note: EOF denotes the "END OF FILE".

(a) Data set.
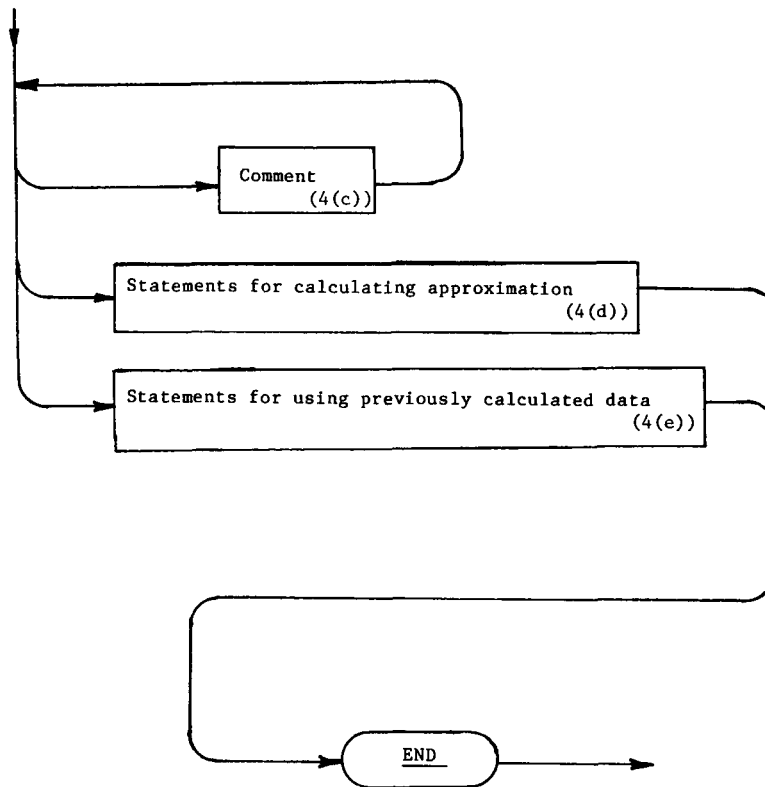
(b) Block.

Anything except:

REFERENCE LENGTH

or

USE APPROXIMATION ON VDATA FILE

(c) Comment.

Figure 4. Syntax diagrams for command input. Key elements are underlined.
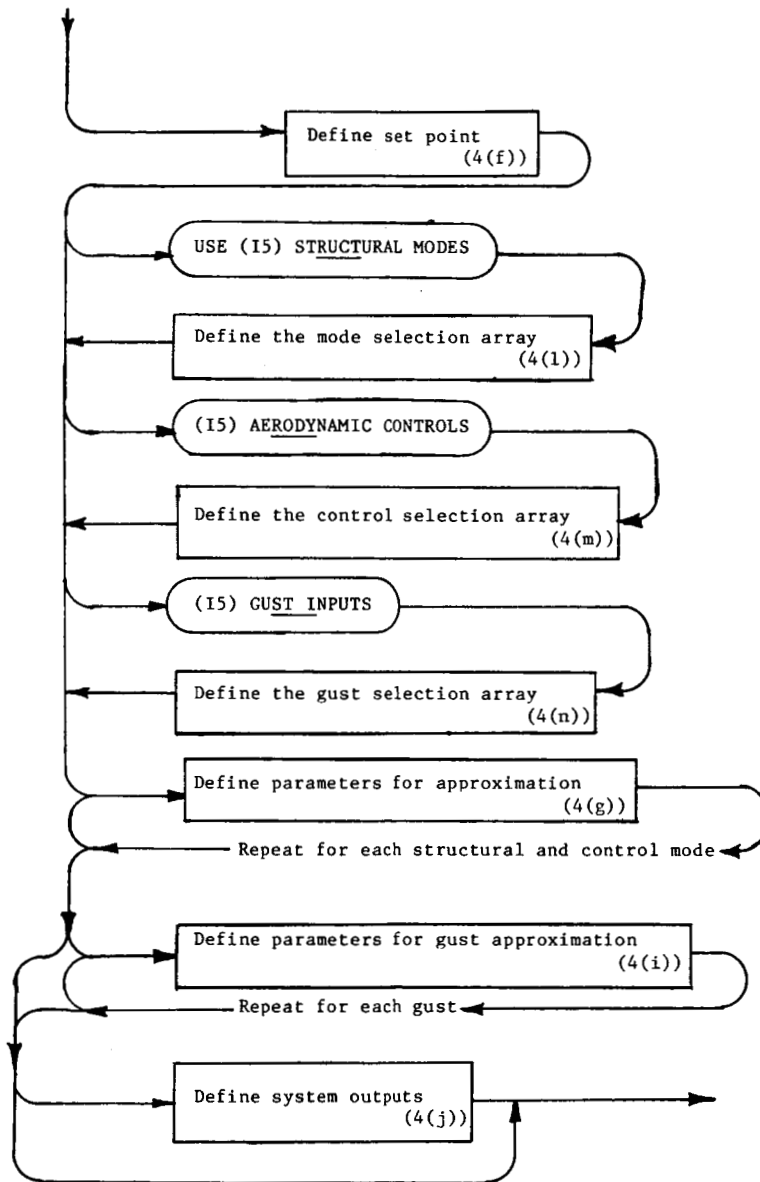
(d) Statements for calculating approximation.

Figure 4. Continued.
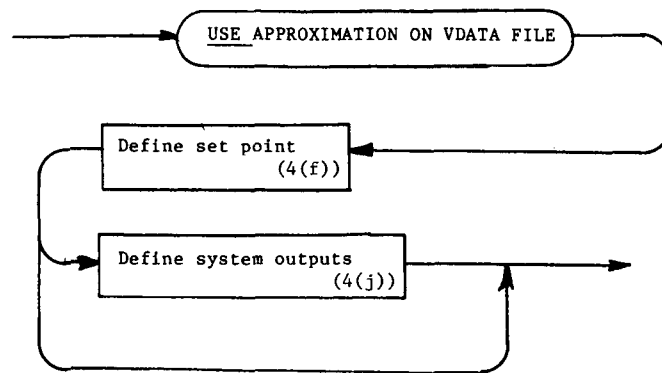
(e) Statements for using previously calculated data.
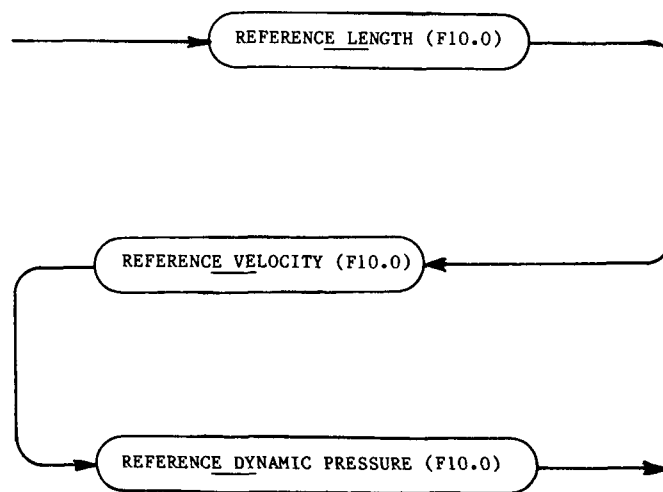
(f) Definition of set point.

Figure 4. Continued.
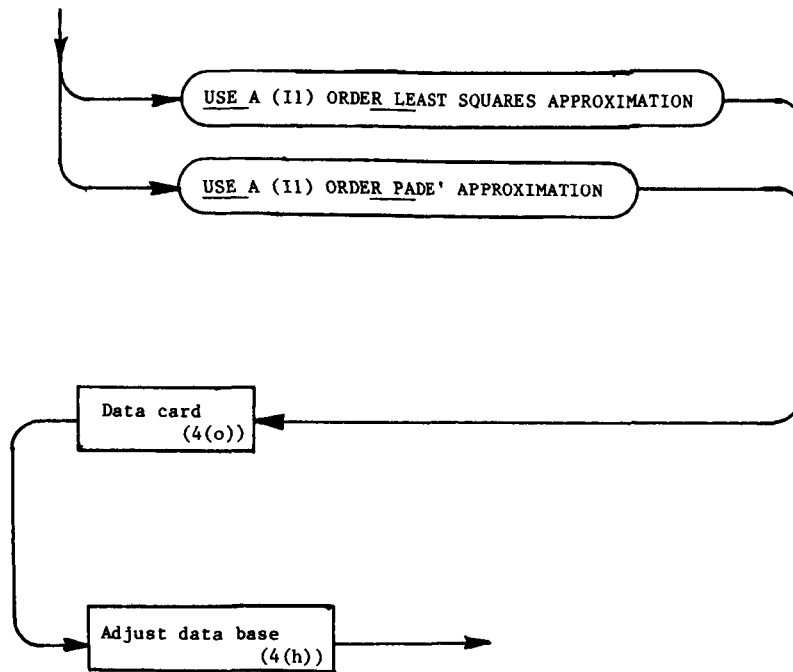
(g) Definition of parameters for approximation.
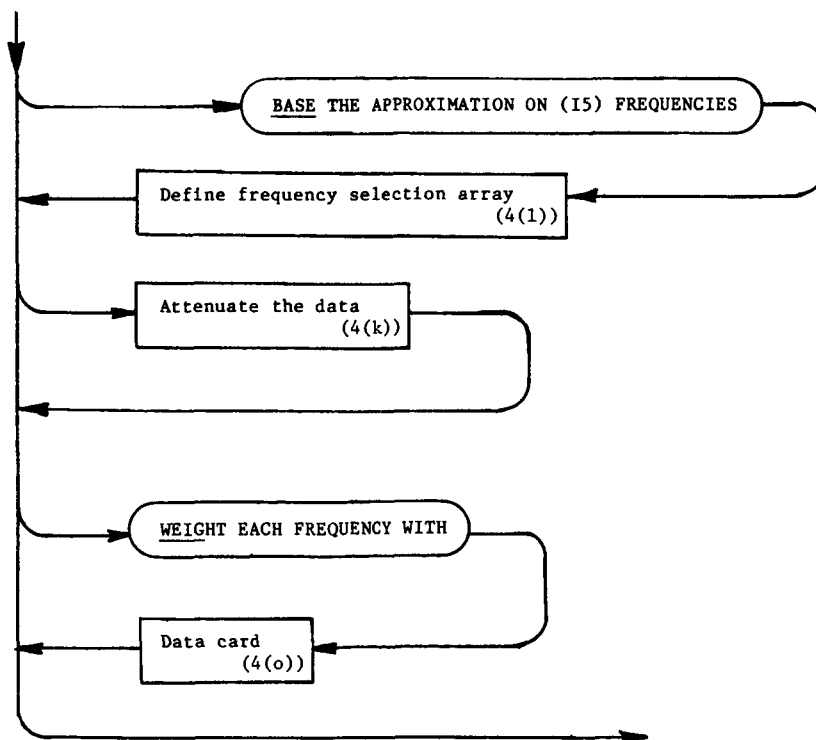
(h) Adjustment of data base.

Figure 4. Continued.
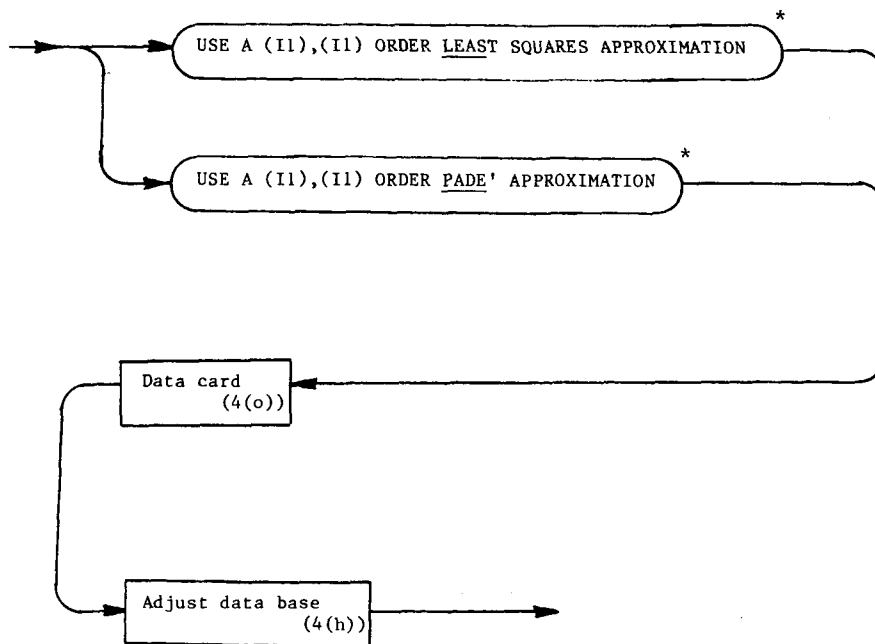
* First entry specifies order of numerator,
  second entry specifies order of denominator.

(i) Definition of parameters for gust approximation.
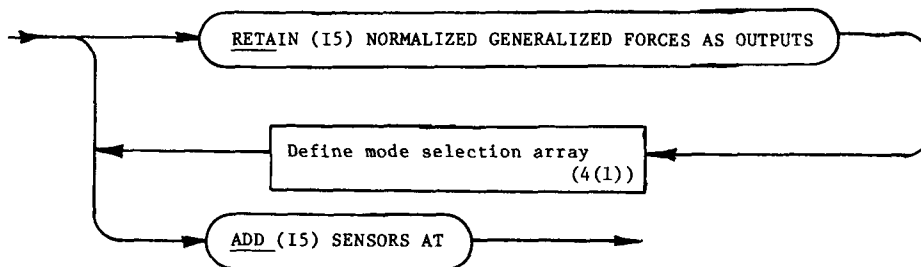
(j) Definition of system outputs.

Figure 4. Continued.
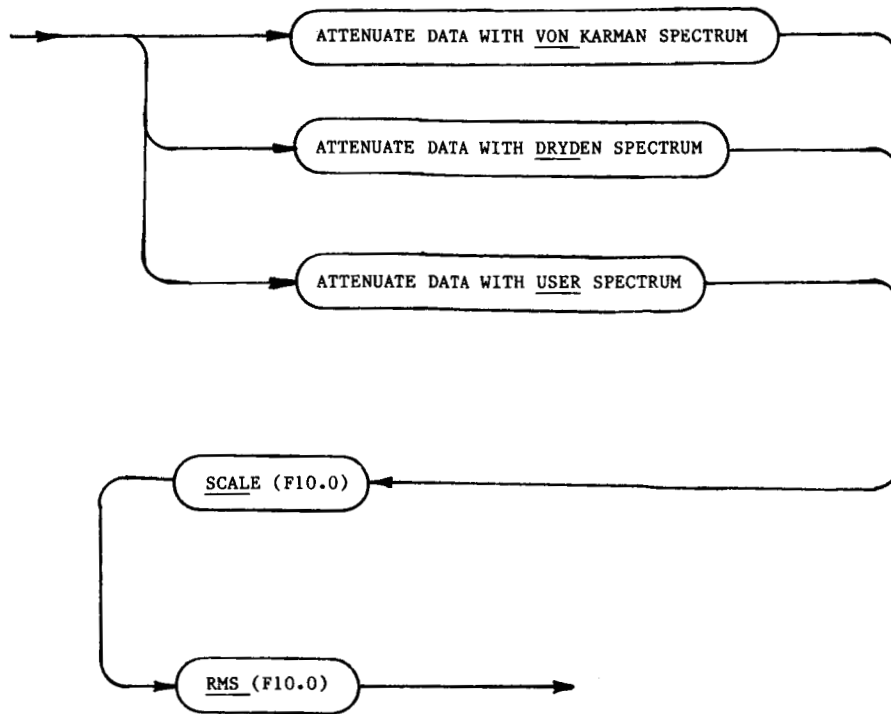
(k) Attenuation of the data.
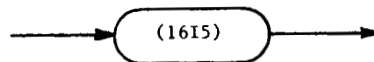
(l) Define mode selection array.

Use only if the number of controls is not zero
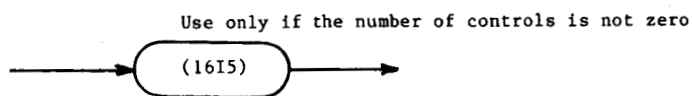
(m) Define control selection array.

Use only if the number of gusts is not zero
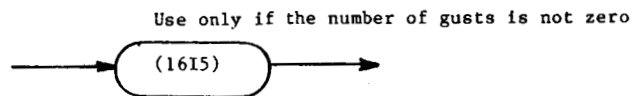
(n) Define gust selection array.
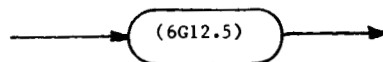
(o) Data card.

Figure 4. Concluded.

```
(1) DAST ARW-1 WIND TUNNEL MODEL, MACH=.9
    (6G12.5)
(2) MODES, CONTROLS, GUSTS, FREQUENCIES
        10     1     1     8
(3) GENERALIZED MASS
        .21000E-01 0.          0.          0.          0.          0.
        0.          0.          0.          0.          0.          .44360E-01
        0.          0.          0.          0.          0.          0.
        0.          0.          0.          0.          .40220E-01 0.
        0.          0.          0.          0.          0.          0.
        0.          0.          0.          .16960E-01 0.          0.
        0.          0.          0.          0.          0.          0.
        0.          0.          .26920E-01 0.          0.          0.
        0.          0.          0.          0.          0.          0.
        0.          .27170E-01 0.          0.          0.          0.
        0.          0.          0.          0.          0.          0.
        .29440E-01 0.          0.          0.          0.          0.
        0.          0.          0.          0.          0.          .64510E-01
        0.          0.          0.          0.          0.          0.
        0.          0.          0.          0.          .43160E-01 0.
        0.          0.          0.          0.          0.          0.
        0.          0.          0.          .31410E-01
(4) GENERALIZED DAMPING
        0.          0.          0.          0.          0.          0.
        0.          0.          0.          0.
(5) GENERALIZED FREQUENCY
        5.2330      19.130      20.910      25.770      46.110      61.230
        79.680      86.030      98.090      118.20
(6) REDUCED FREQUENCIES
        0.          .10000      .30000      .50000      .70000      .90000
        1.3000      1.8000
(7) GENERALIZED FORCE FOR THE,     1 FREQUENCY, REAL PART
        25.240      39.583      .18119E-01 -4.9160      13.585      3.0988
        4.1912     -.16004E-01 -2.9069      6.4767      .41256     -261.26
        -383.03    -.19168      31.395     -64.890     -31.222     -72.605
        .20863      25.085     -59.666     -2.7330     -.20968     -.32400
        -.15800E-03 .29615E-01 -.76497E-01 -.30163E-01 -.48386E-01 .15504E-03
        .21652E-01 -.53256E-01 -.18149E-02 -112.69     -218.92     -.97377E-01
        24.582     -109.38     -26.026     -3.9417      .52573E-01  22.905
        -44.271    -.20162     -90.554     -349.37     -.14447      41.350
        -332.36    -47.701     -10.520      .43263E-01  84.945     -42.929
        2.1818      4.8231      77.844      .28151E-01  22.286      208.21
        -53.838    -157.73      .23838      9.5637      78.125      4.1296
        -40.955    -104.09     -.61997E-01  28.705      88.825     -58.793
```

Figure 5. Example of discrete-frequency data base. Parenthetical numbers (1) to (8) refer to section 6.1 of the text.

```
     -231.19         .42353       -31.645        33.970        8.7387       .67100E-01
      .12163       .89145E-04  -.24503E-01    -.22023        .23544E-01    .34157
    -.71539E-03     .19546       .83258E-02  -.16810E-01   -15.612       -18.523
    -.14360E-01  -24.816        -.37068       174.85        131.42       -.28194E-01
    -344.77      -129.31         7.0539         9.3481       21.380        .86733E-02
      -8.0774      49.002       -9.1440         7.0522        .14844E-01   30.566
    -135.28         5.3168      186.44        -107.84       -.77173E-01  -23.090
     -45.771      -90.593       -80.140         .16757     -100.71        1.9255
      62.662     2136.4        4326.9           1.9373     -321.53      2905.7
    -243.87      -667.34        -.17575       663.29        647.10        28.739
```
(8) GENERALIZED FORCE FOR THE,   1 FREQUENCY, IMAGINARY PART
```
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
      0.          0.           0.            0.            0.            0.
```

Figure 5. Concluded.

```
(1) DAST ARW-1 WIND TUNNEL MODEL, MACH=.9
(2) NUMBER OF MODES, CONTROLS AND GUST IN THE MODEL
        5      1      1
    (6G12.5)
(3) MODE      1
        1
      .02366
      .9371       23.44        67.66       16.03       .4722       35.23
      33.03      -22.60       -.4374      -4.286      -14.62      -11.67
      .4216       7.995       -14.86      -5.188      -.05001      4.431
      .5359      -1.664
    MODE      2
        1
      .2135
     -53.96     -233.1       -106.2       43.75      -81.11      -193.2
      277.8      -63.34        5.117       61.05      107.7       -50.92
     -14.13       69.20       164.3       -58.83      -7.393      -32.96
     -12.50       -7.457
    MODE      3
        1
      .1353
     -15.22      -92.02       -77.42       20.30      -29.42      -145.3
      69.22       -9.049        3.537       42.34      152.2         4.483
     -15.10      -48.06        92.07       -6.066      -3.176      -45.76
     -53.61       -3.735
    MODE      4
        1
      .1281
     -10.81      -52.49       -75.14       44.23      -43.03      -246.0
      107.2        1.714        3.414       75.90      231.0       -84.57
     -45.12     -208.2        314.1       -86.71      -4.133      -98.91
     -100.5       21.23
    MODE      5
        1
      .0964
      .9128      -25.74       -3.885      -19.63       6.972       22.70
     -1.655      -31.81        1.707       13.91     -130.8        39.08
      19.39      164.1        -99.10       -7.791      -7.853        5.663
      264.2       -5.296
    CONTROL      1
        1
      .3336
      61.79      158.7        -7.124        9.744     -36.74      -179.4
     -30.73       15.45       -7.183      -29.25      -21.63        7.229
     -17.91      -30.03       13.81       -19.33      -31.18      -89.12
     -12.65       -7.938
(4) GUST      1
        0      1
      2136.4      4326.9      -321.53      2905.7     -243.87
(5) GENERALIZED MASS
      .21000E-01 0.         0.          0.          0.          0.
      .44360E-01 0.         0.          0.          0.          0.
      .16960E-01 0.         0.          0.          0.          0.
      .26920E-01 0.         0.          0.          0.          0.
      .27170E-01
(6) GENERALIZED DAMPING
      0.          0.         0.          0.          0.
(7) GENERALIZED FREQUENCY
      5.2330      19.130      25.770      46.110      61.230      79.680
```

Figure 6. Example of continuous-frequency data base. Parenthetical numbers (1) to (7) refer to section 6.2 of the text.

```
****  INPUT DATA CARDS  ****

$UNSTEADY AERO DATA
  LOW Q TEST
REFERENCE LENGTH 2.0
REFERENCE VELOCITY 37.67
REFERENCE DYNAMIC PRESSURE 1.
USE A 2 ORDER PADE' APPROXIMATION FOR MODE 1
 .01312   .361
WEIGHT EACH FREQUENCY WITH
3.33      4.       5.       6.67     10.      20.
50.       20.      10.      6.67     5.       4.
3.33      2.56     2.5      2.2      2.0      1.82
1.67      1.54     1.43
USE A 2 ORDER PADE' APPROXIMATION FOR MODE 2
 .01312   .361
WEIGHT EACH FREQUENCY WITH
3.33      4.       5.       6.67     10.      20.
50.       20.      10.      6.67     5.       4.
3.33      2.56     2.5      2.2      2.0      1.82
1.67      1.54     1.43
USE A 0 ORDER LEAST SQURES APPROXIMATION FOR CONTROL 1
END OF UNSTEADY DATA

****  END OF INPUT DATA CARDS  ****

P MATRIX
 COLUMN       1            2           3           4           5
ROW   1   -.31188      12.192      38.900      34.696      9.3565
ROW   2   .43660E-01   -1.7069     -.78899     7.7088      1.8317

R VECTOR
 COLUMN       1            2
ROW   1    1.4824       4.0000
```

Figure 7. Example of printed UNAERO output. Numbers circled refer to section 7 of the text.

COMPARISON OF Q 1 (1)
FOR MODE 1

MODE NUMBER 1

FREQUENCY	ORIG REAL	ORIG IMAG	RECOMP REAL	RECOMP IMAG	ABS ERROR	REL ERROR (%)	WEIGHT
0.00	0.	0.	-.210393	0.	.210393	0.	1.82483
.05	-.157080E-01	.571150	-.217181	.438351	.241303	42.2326	2.00000
.10	-.628320E-01	1.04540	-.239174	.869469	.249095	23.7848	2.23607
.15	-.141370	1.45670	-.280626	1.28784	.218871	14.9549	2.58263
.20	-.251330	1.82860	-.346962	1.69056	.167932	9.09809	3.16228
.25	-.392700	2.17570	-.443323	2.07710	.110840	5.01345	4.47214
.30	-.565490	2.50690	-.573711	2.44853	.589424E-01	2.29358	7.07107
.35	-.769690	2.82760	-.740774	2.80666	.357028E-01	1.21832	4.47214
.40	-1.00530	3.14150	-.946010	3.15332	.604575E-01	1.83292	3.16228
.45	-1.27230	3.45070	-1.19010	3.49011	.911573E-01	2.47860	2.58263
.50	-1.57080	3.75690	-1.47324	3.81821	.115223	2.82960	2.23607
.55	-1.90070	4.06110	-1.79937	4.13843	.130069	2.91419	2.00000
.60	-2.26190	4.36410	-2.15631	4.45124	.136900	2.78510	1.82483
.65	-2.65460	4.66620	-2.55591	4.75687	.134019	2.49641	1.60000
.70	-3.07880	4.96790	-2.99402	5.05537	.121811	2.08418	1.58114
.75	-3.53430	5.26940	-3.47060	5.34666	.100138	1.57824	1.48324
.80	-4.02120	5.57090	-3.98563	5.63058	.694798E-01	1.01126	1.41421
.85	-4.53960	5.87250	-4.53919	5.90692	.344253E-01	.463793	1.34907
.90	-5.08940	6.17430	-5.13141	6.17544	.420304E-01	.525281	1.29228
.95	-5.67060	6.47640	-5.76249	6.43590	.100419	1.16657	1.24097
1.00	-6.28320	6.77870	-6.43264	6.68803	.174795	1.89115	1.19583

MODE NUMBER 2

FREQUENCY	ORIG REAL	ORIG IMAG	RECOMP REAL	RECOMP IMAG	ABS ERROR	REL ERROR (%)	WEIGHT
0.00	0.	-.799610E-01	-.294535E-01	0.	.294535E-01	1.89115	1.82483
.05	-.565490E-02	-.146360	-.225499E-01	-.613694E-01	.337810E-01	42.1416	2.00000
.10	-.226190E-01	-.203940	-.206677E-02	-.121726	.348743E-01	23.5482	2.23607
.15	-.508940E-01	-.256010	-.314000E-01	-.180299	.306418E-01	14.5779	2.58263
.20	-.904780E-01	-.304600	-.770910E-01	-.236679	.235140E-01	8.65988	3.16228
.25	-.141370	-.350960	-.134286	-.290794	.159173E-01	4.62088	4.47214
.30	-.203580	-.395870	-.202426	-.342795	.824604E-02	2.03239	7.07107
.35	-.277090	-.439810	-.281139	-.392932	.500209E-02	1.03518	4.47214
.40	-.361910	-.483100	-.370215	-.441465	.846835E-02	1.48679	3.16228
.45	-.458040	-.525970	-.469559	-.488615	.127715E-01	1.91845	2.58263
.50	-.565490	-.568560	-.579145	-.534549	.161261E-01	2.08810	2.23607
.55	-.684240	-.610970	-.698980	-.579379	.182845E-01	2.05529	2.00000
.60	-.814300	-.653270	-.829089	-.623172	.191728E-01	1.88334	1.82483
.65	-.955670	-.695500	-.969494	-.665960	.187656E-01	1.62106	1.60000
.70	-1.10840	-.737710	-1.12021	-.707750	.170189E-01	1.30061	1.58114
.75	-1.27230	-.779920	-1.28126	-.748530	.140480E-01	.955188	1.48324
.80	-1.44760	-.822150	-1.45263	-.788279	.975429E-02	.593208	1.41421
.85	-1.63430	-.864400	-1.63431	-.826967	.481674E-02	.263290	1.34907
.90	-1.83220	-.906690	-1.82628	-.864560	.591736E-02	.292090	1.29228
.95	-2.04140	-.949020	-2.02853	-.901023	.140621E-01	.629543	1.24097
1.00	-2.26190		-2.24101	-.936322	.244441E-01	.996528	1.19583

MAX ABS ERROR =   .24909   AT MODE 1 , FREQ .10
MAX REL ERROR = 42.233 %   AT MODE 1 , FREQ .05
COST = .4547

Figure 7. Continued.

44

(13)

A MATRIX

	1	2	3	4	5	6	7
ROW 1	0.	0.	1.0000	0.	0.	0.	0.
ROW 2	0.	0.	0.	1.0000	0.	0.	0.
ROW 3	-1101.1	-540.95	.77362E-01	4.1034	77441.	4892.1	.30328E+07
ROW 4	86.208	176.72	-.94712E-01	-2.1901	-33076.	-2089.5	-.12955E+07
ROW 5	0.	0.	0.	0.	0.	1.0000	0.
ROW 6	1.0000	0.	0.	0.	-2103.6	-150.68	0.
ROW 7	0.	0.	0.	0.	0.	0.	-5676.1
ROW 8	0.	1.0000	0.	0.	0.	0.	0.

	8
ROW 1	0.
ROW 2	0.
ROW 3	55357.
ROW 4	-23646.
ROW 5	0.
ROW 6	1.0000
ROW 7	-150.68
ROW 8	

B MATRIX

	1	2	3
ROW 1	0.	0.	0.
ROW 2	.14920	0.	-.20545E-17
ROW 3	-2.6337	0.	.36266E-16
ROW 4	0.	0.	0.
ROW 5	0.	0.	0.
ROW 6	0.	0.	0.
ROW 7	0.	0.	0.
ROW 8	0.	0.	0.

C MATRIX

	1	2	3	4	5	6	7
ROW 1	31.537	536.68	-.75089E-01	-4.0748	-76647.	-4842.0	-.300177E+07

	8
ROW 1	-54789.

D MATRIX

	1	2	3
ROW 1	-.85993E-01	0.	.11841E-17

Figure 7. Concluded.

45

```
                                          ┌PHID
                         FDATA----┤PHIVK
                                          └USER

                         LSTSOL---┌CALAP------SNVDEC
                                          └CALPJ------EVALR

                                          ┌CALAP------SNVDEC

                                          ├CALPJ------EVALR

                                          │          ┌EVALR
                                          ├CALGRA----┤EVALP
                         PADSOL---┤          └EVALRR
             CALPR----┤
                                          │          ┌CALPJ----EVALR
                                          ├CALZFQ----┤EVALP
                                          │          └EVALR
                                          │
                                          └CONMIN

┌─────────────┐
│ USER        │
│ CALLING     │---UNAERO---┤
│ PROGRAM     │
└─────────────┘                                   ┌CALPJ----EVALR
                         PRTDIF----CALZFQ---┤EVALP
                                                     └EVALR

                         PRESEN

                                  ┌REFORM
             FABCD----┤SENSOR
                                  └SNVDEC
```

┌────────────────────┐
│ Common Routines   │
├────────────────────┤
│   PRNT   │
│   ERR   │
│   ERRMON   │
│   ERMON   │
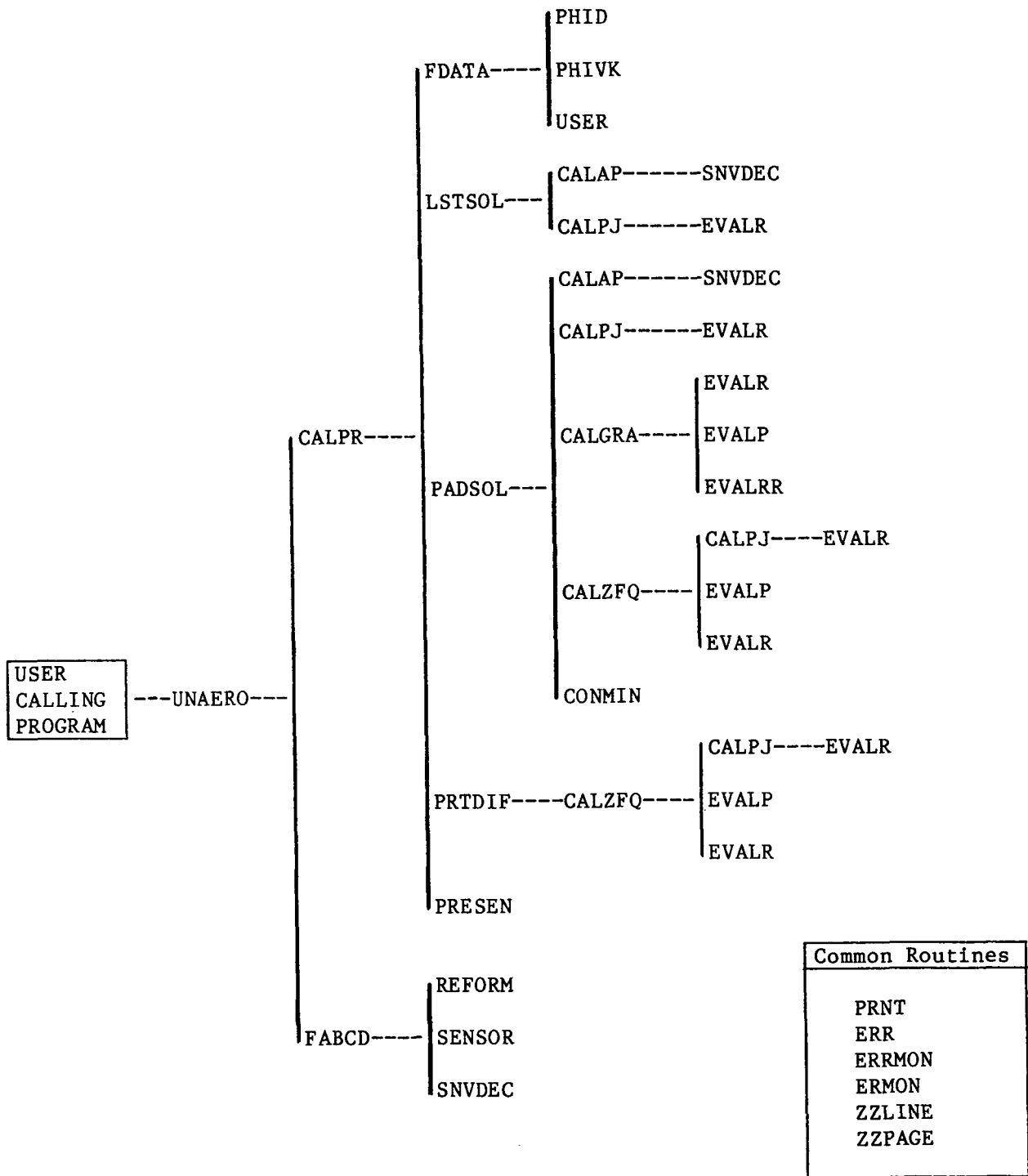│   ZZLINE   │
│   ZZPAGE   │
└────────────────────┘
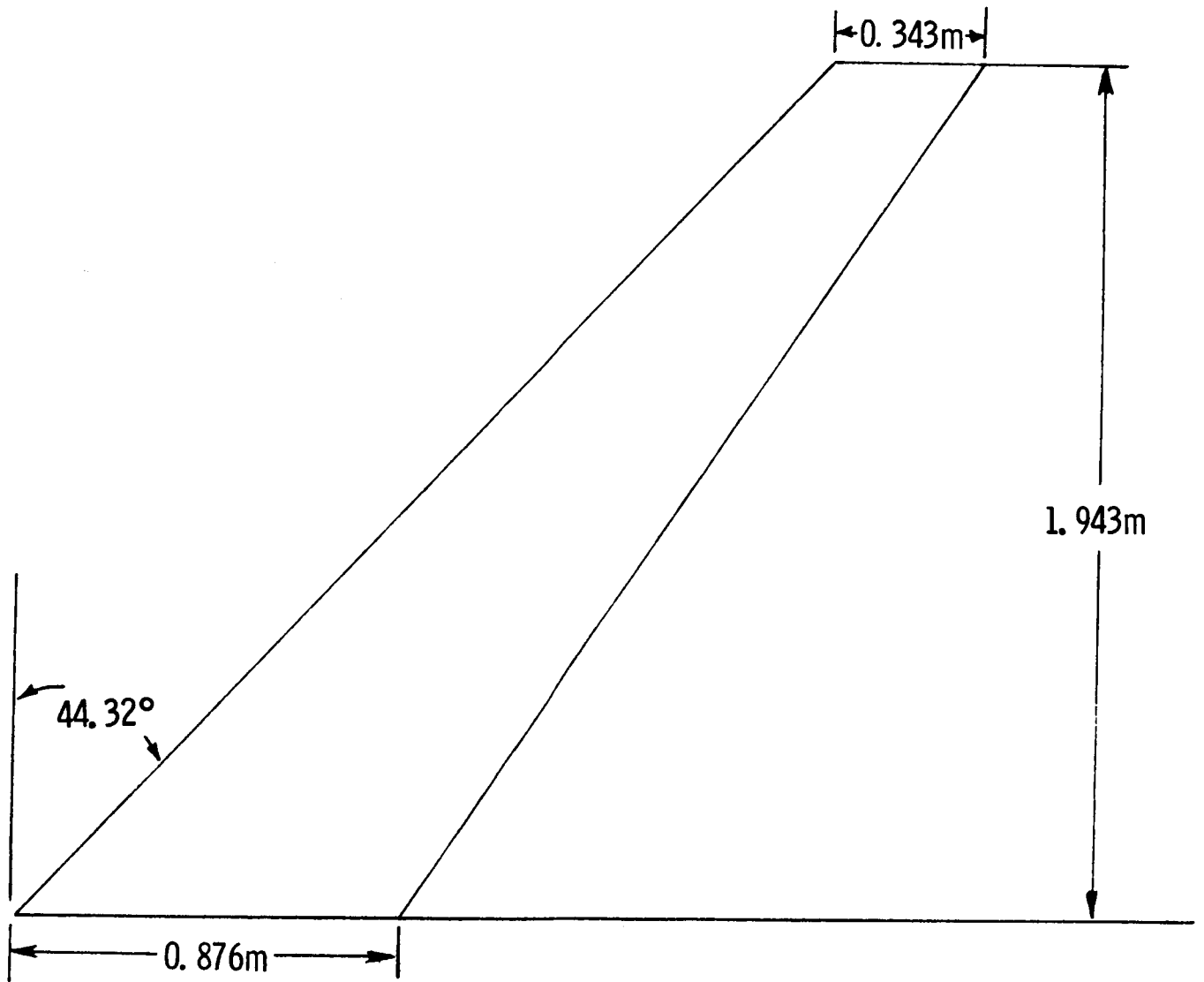
Figure 8. UNAERO structure and hierarchy.
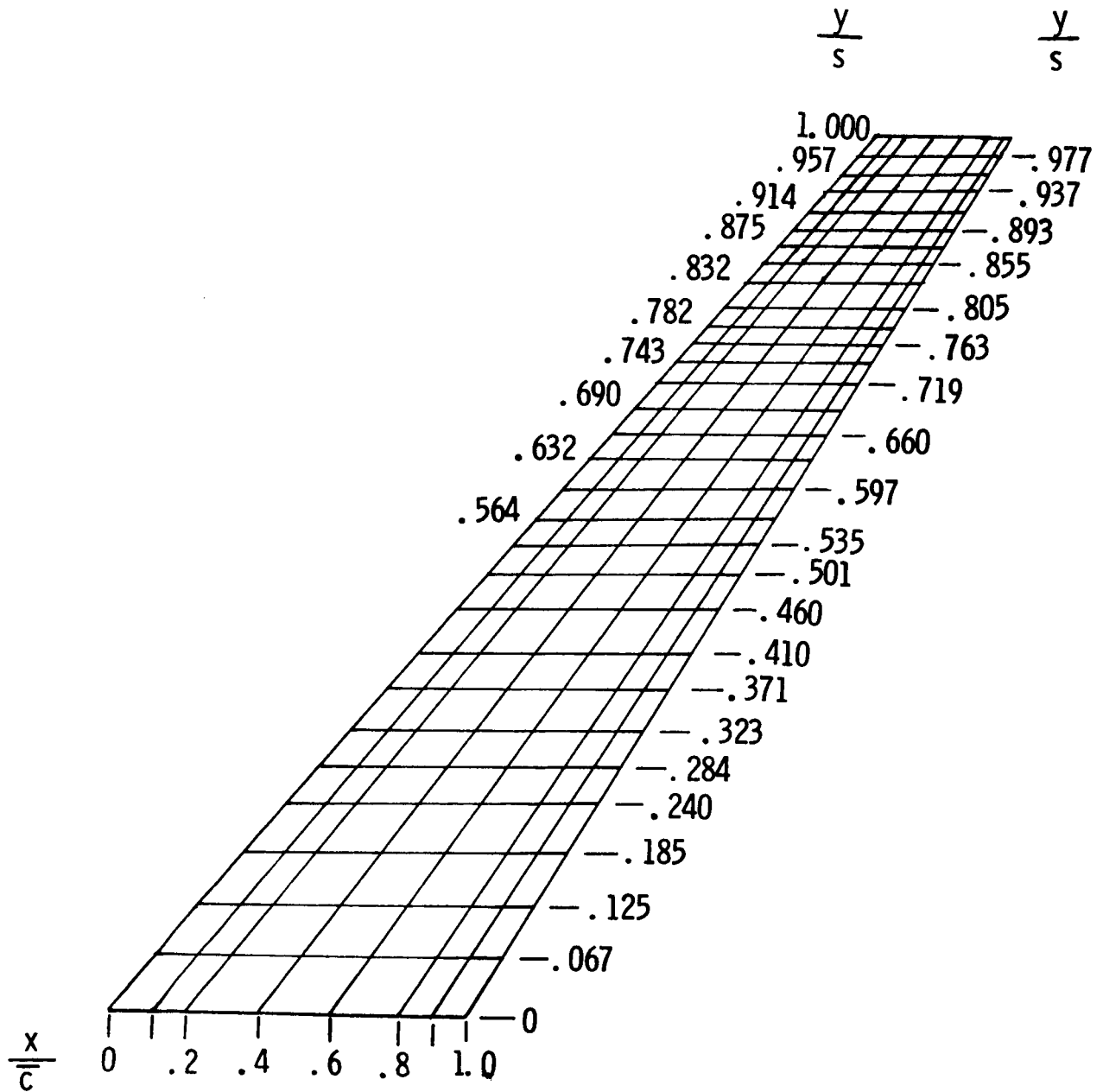
Figure 9. Model wing geometry.

Figure 10. Scheme for doublet-lattice aerodynamics.

Standard Bibliographic Page

1. Report No. NASA TM-86392	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle UNAERO: A Package of FORTRAN Subroutines for Approximating Unsteady Aerodynamics in the Time Domain		5. Report Date October 1985
		6. Performing Organization Code 505-33-43-08
7. Author(s) H. J. Dunn		8. Performing Organization Report No. L-15862
9. Performing Organization Name and Address NASA Langley Research Center Hampton, VA 23665-5225		10. Work Unit No.
		11. Contract or Grant No.
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001		13. Type of Report and Period Covered Technical Memorandum
		14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

This report serves as an instruction and maintenance manual for a collection of CDC CYBER FORTRAN IV subroutines for approximating the unsteady aerodynamic forces in the time domain. The result is a set of constant-coefficient first-order differential equations that approximate the dynamics of the vehicle. Provisions are included for adjusting the number of modes used for calculating the approximations so that an accurate approximation is generated. The number of data points at different values of reduced frequency can also be varied to adjust the accuracy of the approximation over the reduced-frequency range. The denominator coefficients of the approximation may be calculated by means of a gradient method or a least-squares approximation technique. Both the approximation methods use weights on the residual error. A new set of system equations, at a different dynamic pressure, can be generated without the approximations being recalculated.

17. Key Words (Suggested by Authors(s)) Aeroservoelasticity Unsteady aerodynamics Equations of motion Flexible vehicle	18. Distribution Statement Subject Category 61
19. Security Classif.(of this report) Unclassified	20. Security Classif.(of this page) Unclassified	21. No. of Pages 51	22. Price