A Comprehensive Analytical Model of Rotorcraft Aerodynamics and Dynamics
Part II: User's Manual

Wayne Johnson

NASA Technical Memorandum 81183


July 1980

National Aeronautics and Space Administration

United States Army Aviation Research and Development Command
CONTENTS

1. Program Summary 1
2. Subprogram Functions 3
3. Namelist, File, and Common Block Labels 7
4. Program Skeleton 9
5. Job Structure 24
6. Input Description 28
   Namelist NLCASE 29
   Namelist NLIHRIM 30
   Namelist NLHRTR 38
   Namelist NLWAKE 45
   Namelist NLBODY 50
   Namelist NLLOAD 58
   Namelist NLFLUT 63
   Namelist NLSTAB 70
   Namelist NLTRAN 76
   Namelist Inputs for Old Job (Restart) 79
7. Notes on Printed Output 80
8. Units 85
9. Airfoil Table Preparation 86
A COMPREHENSIVE ANALYTICAL MODEL OF
ROTORCRAFT AERODYNAMICS AND DYNAMICS

Part II: User's Manual

Wayne Johnson
Ames Research Center
and
Aeromechanics Laboratory
AVRADCOM Research and Technology Laboratories

SUMMARY

The use of a comprehensive analytical model of rotorcraft aerodynamics and dynamics is described. This analysis is designed to calculate rotor performance, loads, and noise; the helicopter vibration and gust response; the flight dynamics and handling qualities; and the system aeroelastic stability. The analysis is a combination of structural, inertial, and aerodynamic models, that is applicable to a wide range of problems and a wide class of vehicles. The analysis is intended for use in the design, testing, and evaluation of rotors and rotorcraft, and to be a basis for further development of rotary wing theories. This report describes the use of the computer program that implements the analysis.
1. PROGRAM SUMMARY

The computer program calculates the loads and motion of helicopter rotors and airframe. First the trim solution is obtained; then the flutter, flight dynamics, and/or transient behavior can be calculated. Either a new job can be initiated, or further calculations can be performed for an old job.

For a new job, the input consists of block data or an input file (the program can create the input file from the block data), and airfoil files. Then namelists are read for additional data, particularly case-specific inputs. One or more cases can be run for a new job.

For an old job, the input consists of a restart file (written during the execution of a previous job), and namelists. Only one case can be run for an old job. The job can be resumed either at the point where the trim solution was completed, or it can be resumed in one of the subsequent tasks. For a trim restart, any or all of the other tasks can be initiated. For flutter, flight dynamics, or transient restarts, only that task can be done.

For both new and old jobs, a scratch file is usually needed; and the job may write data on the restart file. In the flutter and flight dynamics tasks, eigenvalue data may be written on a file.

For both new and old jobs, a case namelist is always read to define the job, and a trim namelist is read to define the flight condition and analysis tasks. Component and task namelists may be read as required.

The loads and motion solution is obtained by an iterative process. The inner-most loop consists of the rotor and airframe motion calculation, for prescribed control positions, induced velocity distribution, and mean shaft motion. Convergence of the motion solution is determined by comparing the calculated harmonics every few revolutions. The next loop consists of
the uniform or nonuniform rotor-induced velocity calculation, followed by the motion solution. Convergence is determined by comparing the rotor thrust or circulation used to calculate the induced velocity with that resulting after the motion has been re-calculated. Before beginning the circulation and motion iterations, the blade bending and torsion modes are calculated. If the rotor nonuniform induced velocity is used, there is an additional cutar loop, consisting of calculation of the rotor wake influence coefficients followed by the circulation and motion iterations.

To calculate the influence coefficients, the prescribed or free wake geometry must be evaluated. Having completed the motion solution, the performance, loads, vibration, and noise can be evaluated as required.

The trim analysis proceeds in stages. In the first stage the trim solution is obtained for uniform inflow; in the second and third stages the trim solution is obtained for nonuniform inflow, with prescribed or free wake geometry respectively. The analysis can stop at any of these stages. Within each stage, the aircraft controls and orientation are incremented until the equilibrium of forces required for the specified trim state is achieved.

In the flutter analysis, the matrices are constructed that describe the linear differential equations of motion, and the equations are analyzed. Optionally the equations are reduced to just the aircraft rigid body degrees of freedom (by a quasistatic reduction), and the equations are analyzed as for the flight dynamics task.

In the flight dynamics analysis, the stability derivatives are calculated and the matrices are constructed that describe the linear differential equations of motion. These equations are analyzed (optionally including a numerical integration as for the transient analysis).

In the transient analysis, the rigid body equations of motion are numerically integrated, for a prescribed transient gust or control input.
2. SUBPROGRAM FUNCTIONS

The following pages list the subprograms that constitute the analysis, and state the primary function of each subprogram. Only the subprograms for rotor #1 are listed; the subprograms for rotor #2 have identical functions.

Subprogram
Name

MAIN Primary job and analysis control
TIMER Program timer
INPTN Input for new job
INPTO Input for old job
INPTA1 Read airfoil table file
INPTR1 Read rotor namelist
INPTW1 Read wake namelist
INPTB Read body namelist
INPTL1 Read loads namelist
INPTF Read flutter namelist for new job
INPTS Read flight dynamics namelist for new job
INPTT Read transient namelist for new job
INPTG Read flutter namelist for old job
INPTU Read flight dynamics namelist for old job
INPTV Read transient namelist for old job
FILEI Read or write input file
FILEJ Read or write trim data file
FILER Read or write restart file
FILEF Read or write flutter restart file
FILEE Read or write flight dynamics restart file
FILET Read or write transient restart file
FILEE Write eigenvalue file
INIT Initialization
INITA Initialize environment parameters
INITC Initialize case parameters
INITR1 Initialize rotor parameters
INITB Initialize airframe parameters
INITD Initialize drive train parameters
CHEKR1 Check for fatal errors
<table>
<thead>
<tr>
<th>Subprogram</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FRNTJ</td>
<td>Print job input data</td>
</tr>
<tr>
<td>FRNTC</td>
<td>Print case input data</td>
</tr>
<tr>
<td>FRNT</td>
<td>Print trim input data</td>
</tr>
<tr>
<td>FRNTR1</td>
<td>Print rotor input data</td>
</tr>
<tr>
<td>FRNTW1</td>
<td>Print wake input data</td>
</tr>
<tr>
<td>FRNTB</td>
<td>Print body input data</td>
</tr>
<tr>
<td>FRNTF</td>
<td>Print flutter input data</td>
</tr>
<tr>
<td>FRNTS</td>
<td>Print flight dynamics input data</td>
</tr>
<tr>
<td>FRNTT</td>
<td>Print transient input data</td>
</tr>
<tr>
<td>FRNTG</td>
<td>Print transient gust and control input data</td>
</tr>
<tr>
<td>TRIM</td>
<td>Trim</td>
</tr>
<tr>
<td>TRIMI</td>
<td>Calculate trim solution by iteration</td>
</tr>
<tr>
<td>TRIMP</td>
<td>Print trim solution</td>
</tr>
<tr>
<td>FLUT</td>
<td>Flutter</td>
</tr>
<tr>
<td>FLUTM</td>
<td>Calculate flutter matrices</td>
</tr>
<tr>
<td>FLUTB</td>
<td>Calculate flutter aircraft matrices</td>
</tr>
<tr>
<td>FLUTR1</td>
<td>Calculate flutter rotor matrices</td>
</tr>
<tr>
<td>FLUTI1</td>
<td>Calculate flutter inertia coefficients</td>
</tr>
<tr>
<td>FLUTA1</td>
<td>Calculate flutter aerodynamic coefficients</td>
</tr>
<tr>
<td>FLUTL</td>
<td>Analyze flutter constant coefficient linear equations</td>
</tr>
<tr>
<td>STAB</td>
<td>Flight dynamics</td>
</tr>
<tr>
<td>STABM</td>
<td>Calculate flight dynamics stability derivatives and matrices</td>
</tr>
<tr>
<td>STABD</td>
<td>Print stability derivatives</td>
</tr>
<tr>
<td>STABE</td>
<td>Calculate flight dynamics equations</td>
</tr>
<tr>
<td>STABL</td>
<td>Analyze flight dynamics linear equations</td>
</tr>
<tr>
<td>STABP</td>
<td>Print flight dynamics transient solution</td>
</tr>
<tr>
<td>TRAN</td>
<td>Transient</td>
</tr>
<tr>
<td>TRANI</td>
<td>Calculate transient acceleration for numerical integration</td>
</tr>
<tr>
<td>TRANP</td>
<td>Print transient solution</td>
</tr>
<tr>
<td>TRANC</td>
<td>Calculate transient gust and control</td>
</tr>
<tr>
<td>CONTRL</td>
<td>Calculate transient control time history</td>
</tr>
<tr>
<td>GUSTU</td>
<td>Calculate uniform gust time history</td>
</tr>
<tr>
<td>GUSTC</td>
<td>Calculate convected gust wave shape</td>
</tr>
<tr>
<td>PERF</td>
<td>Performance</td>
</tr>
<tr>
<td>PERFR1</td>
<td>Calculate and print rotor performance</td>
</tr>
<tr>
<td>Subprogram</td>
<td>Name</td>
</tr>
<tr>
<td>------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>LOAD</td>
<td>Loads, vibration, and noise</td>
</tr>
<tr>
<td>LOADR1</td>
<td>Calculate and print rotor loads</td>
</tr>
<tr>
<td>LOADH1</td>
<td>Calculate and print hub and control loads</td>
</tr>
<tr>
<td>LOADS1</td>
<td>Calculate and print blade section loads</td>
</tr>
<tr>
<td>LOADI1</td>
<td>Calculate inertia coefficients for section loads</td>
</tr>
<tr>
<td>LOADF</td>
<td>Calculate fatigue damage</td>
</tr>
<tr>
<td>LOADM</td>
<td>Calculate mean and half peak-to-peak</td>
</tr>
<tr>
<td>GEOMPL</td>
<td>Printer-plot of wake geometry</td>
</tr>
<tr>
<td>POLRPP</td>
<td>Printer-plot of polar plot</td>
</tr>
<tr>
<td>HISTPP</td>
<td>Printer-plot of azimuthal time history</td>
</tr>
<tr>
<td>NOISR1</td>
<td>Calculate and print far field rotational noise</td>
</tr>
<tr>
<td>BESSKL</td>
<td>Calculate J Bessel function</td>
</tr>
<tr>
<td>RAMF</td>
<td>Calculate rotor/airframe periodic motion and forces</td>
</tr>
<tr>
<td>MODE1</td>
<td>Blade modes</td>
</tr>
<tr>
<td>MODEC1</td>
<td>Initialize blade mode parameters</td>
</tr>
<tr>
<td>MODEB1</td>
<td>Calculate blade bending modes</td>
</tr>
<tr>
<td>MODEG</td>
<td>Calculate Galerkin functions for bending modes</td>
</tr>
<tr>
<td>MODEA1</td>
<td>Calculate articulated blade flap and lag modes</td>
</tr>
<tr>
<td>MODET1</td>
<td>Calculate blade torsion modes</td>
</tr>
<tr>
<td>MODEK1</td>
<td>Calculate kinematic pitch-bending coupling</td>
</tr>
<tr>
<td>MODED1</td>
<td>Calculate blade root geometry</td>
</tr>
<tr>
<td>INRTC1</td>
<td>Calculate blade inertia coefficients</td>
</tr>
<tr>
<td>MODEP1</td>
<td>Print blade modes</td>
</tr>
<tr>
<td>BODYC</td>
<td>Initialize airframe parameters at trim</td>
</tr>
<tr>
<td>ENMNC1</td>
<td>Initialize drive train parameters at trim</td>
</tr>
<tr>
<td>MOTNC1</td>
<td>Initialize rotor parameters at trim</td>
</tr>
<tr>
<td>BODYM1</td>
<td>Calculate airframe transfer function matrix</td>
</tr>
<tr>
<td>ENMNM1</td>
<td>Calculate drive train transfer function matrix</td>
</tr>
<tr>
<td>WAKEU1</td>
<td>Calculate uniform wake-induced velocity</td>
</tr>
<tr>
<td>WAKEN1</td>
<td>Calculate nonuniform wake-induced velocity</td>
</tr>
<tr>
<td>INRTM1</td>
<td>Calculate rotor transfer function matrix</td>
</tr>
<tr>
<td>INRTI</td>
<td>Calculate inverse of transfer function matrix</td>
</tr>
<tr>
<td>MOTNH1</td>
<td>Calculate harmonics of hub motion</td>
</tr>
<tr>
<td>MOTNR1</td>
<td>Calculate harmonics of rotor motion</td>
</tr>
<tr>
<td>MOTNB1</td>
<td>Calculate blade and hub motion</td>
</tr>
<tr>
<td>AEROF1</td>
<td>Calculate blade aerodynamic forces</td>
</tr>
<tr>
<td>AEROS1</td>
<td>Calculate blade section aerodynamic coefficients</td>
</tr>
<tr>
<td>AERT1</td>
<td>Interpolate airfoil tables</td>
</tr>
<tr>
<td>BODYV1</td>
<td>Calculate harmonics of airframe motion</td>
</tr>
<tr>
<td>ENGNV1</td>
<td>Calculate harmonics of drive train motion</td>
</tr>
<tr>
<td>MOTNF1</td>
<td>Calculate rotor generalized forces</td>
</tr>
<tr>
<td>MOTNS</td>
<td>Calculate static elastic motion</td>
</tr>
<tr>
<td>BODIF</td>
<td>Calculate airframe generalized forces</td>
</tr>
<tr>
<td>BODYA</td>
<td>Calculate body aerodynamic forces</td>
</tr>
<tr>
<td>Subprogram</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>WAKECl</td>
<td>Calculate influence coefficients for nonuniform inflow</td>
</tr>
<tr>
<td>WAKEB1</td>
<td>Calculate blade position</td>
</tr>
<tr>
<td>VTXL</td>
<td>Calculate vortex line segment induced velocity</td>
</tr>
<tr>
<td>VTXS</td>
<td>Calculate vortex sheet segment induced velocity</td>
</tr>
<tr>
<td>GEOMEl</td>
<td>Evaluate wake geometry</td>
</tr>
<tr>
<td>GEOMFl</td>
<td>Calculate wake geometry distortion</td>
</tr>
<tr>
<td>GEOMF1</td>
<td>Calculate free wake geometry distortion</td>
</tr>
<tr>
<td>MINV</td>
<td>Calculate inverse of matrix</td>
</tr>
<tr>
<td>MINVC</td>
<td>Calculate inverse of complex matrix</td>
</tr>
<tr>
<td>EIGENJ</td>
<td>Calculate eigenvalues and eigenvectors of matrix</td>
</tr>
<tr>
<td>DERED</td>
<td>Eliminate equations and variables from system of differential equations</td>
</tr>
<tr>
<td>QSTRAN</td>
<td>Quasistatic reduction of system of linear differential equations</td>
</tr>
<tr>
<td>CSYSAN</td>
<td>Analyze system of constant coefficient linear differential equations</td>
</tr>
<tr>
<td>DETRAN</td>
<td>Transform equations to state variable form</td>
</tr>
<tr>
<td>SINE</td>
<td>Calculate frequency response from matrices</td>
</tr>
<tr>
<td>STATIC</td>
<td>Calculate static response from matrices</td>
</tr>
<tr>
<td>ZERO</td>
<td>Calculate zeros</td>
</tr>
<tr>
<td>ZETRAN</td>
<td>Transform matrix for zero calculation</td>
</tr>
<tr>
<td>BODE</td>
<td>Calculate and printer-plot transfer function (Bode plot)</td>
</tr>
<tr>
<td>BODEPFP</td>
<td>Printer-plot transfer function magnitude and phase</td>
</tr>
<tr>
<td>TRACKS</td>
<td>Calculate and printer-plot time history of time-invariant system response</td>
</tr>
<tr>
<td>TRCKPP</td>
<td>Printer-plot time history</td>
</tr>
<tr>
<td>GUSTS</td>
<td>Calculate and print rms gust response</td>
</tr>
<tr>
<td>PSYSAN</td>
<td>Analyze system of periodic coefficient linear differential equations</td>
</tr>
<tr>
<td>DEPRAN</td>
<td>Transform equations to state variable form</td>
</tr>
</tbody>
</table>
3. NAMELIST, FILE, AND COMMON BLOCK LABELS

The list below gives the namelist labels used by the program, and the type of input data read in each. The corresponding common block labels are given in the right-hand column.

<table>
<thead>
<tr>
<th>Namelist Label</th>
<th>Common Block Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLCASE</td>
<td>Job data</td>
</tr>
<tr>
<td>NLRTR</td>
<td>Rotor data</td>
</tr>
<tr>
<td>NLTRIM</td>
<td>Trim data</td>
</tr>
<tr>
<td>NLWAKE</td>
<td>Wake data</td>
</tr>
<tr>
<td>NLLOAD</td>
<td>Loads data</td>
</tr>
<tr>
<td>NLFLUT</td>
<td>Flutter data</td>
</tr>
<tr>
<td>NLSTAB</td>
<td>Flight dynamics data</td>
</tr>
<tr>
<td>NLTRAN</td>
<td>Transient data</td>
</tr>
<tr>
<td>NLBODY</td>
<td>Airframe and drive train data</td>
</tr>
<tr>
<td>NLOAD</td>
<td>Loads data</td>
</tr>
<tr>
<td>TMDATA</td>
<td>RDATA, FDATA</td>
</tr>
<tr>
<td>G1DATA,W1DATA</td>
<td></td>
</tr>
<tr>
<td>BDATA,BNDATA,ENDATA</td>
<td></td>
</tr>
<tr>
<td>LDATA,L1DATA</td>
<td></td>
</tr>
<tr>
<td>FLDATA</td>
<td></td>
</tr>
<tr>
<td>STDATA,GCDATA</td>
<td></td>
</tr>
<tr>
<td>TNDATA,GCDATA</td>
<td></td>
</tr>
</tbody>
</table>

The list below gives the files used by the program. The left-hand column gives the input parameter that defines the file unit number.

<table>
<thead>
<tr>
<th>Unit Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NFDAT</td>
<td>Input data</td>
</tr>
<tr>
<td>NFAF1</td>
<td>Rotor #1 airfoil data</td>
</tr>
<tr>
<td>NFAF2</td>
<td>Rotor #2 airfoil data</td>
</tr>
<tr>
<td>NFSR</td>
<td>Restart data</td>
</tr>
<tr>
<td>NFEIG</td>
<td>Eigenvalue data</td>
</tr>
<tr>
<td>NFSR</td>
<td>Scratch data</td>
</tr>
</tbody>
</table>

The list below gives the labels of the common blocks used by the program, and states the type of data contained in each. Only the common blocks for rotor #1 are listed; the common blocks for rotor #2 have identical functions.
Common Block

Label

TMDATA  Input trim data
R1DATA  Input rotor data
W1DATA  Input wake data
G1DATA  Input free wake geometry data
BDATA  Input airframe data
BADATA  Input airframe aerodynamics data
BNDATA  Input drive train data
L1DATA  Input rotor loads data
L2DATA  Input airframe loads data
GCDATA  Input gust and control data
TNDATA  Input transient data
STDATA  Input flight dynamics data
FDTDATA  Input flutter data
A1TABL  Rotor airfoil tables
UNITNO  Input/output unit numbers
CASECM  Job description

TRIMCM  Calculated trim data
RTR1CM  Calculated rotor data
RH1CM  Transfer function matrices
BOD1CM  Calculated airframe data
EN1CM  Calculated drive train data
GUSTCM  Gust and transient control
CONT1CM  Aircraft controls and motion
CONVC1M  Circulation and motion convergence
ND1CM  Blisie modes
IN1CM  Rotor inertial coefficients
WKV1CM  Induced velocity
MN1CM  Hub motion
AB1CM  Blade section aerodynamics
MN1CM  Rotor motion and airframe vibration
NS1CM  Static elastic motion
ABF1CM  Rotor forces
QF1CM  Rotor generalized forces
QF2CM  Airframe generalized forces
WG1CM  Wake geometry
WKF1CM  Wake influence coefficients

AB1CM  Calculated motion data
LDM1CM  Calculated loads data
F1M1CM  Flutter matrices
F2M1CM  Flutter rotor matrices
F3M1CM  Flutter airframe matrices
F4M1CM  Flutter inertial coefficients
F5M1CM  Flutter aerodynamic coefficients
STD1CM  Flight dynamics stability derivatives
STM1CM  Flight dynamics matrices
TRANCM  Calculated transient data
4. PROGRAM SKELETON

The following pages present a schematic of the program, showing the basic flow of control and the major loops, options, and decisions. The appearance of a subprogram name (always in capital letters) means that the subprogram is called at that point in the analysis. The contents of a subprogram are shown by means of a three-sided box appearing below the subprogram name. The schematic defines the input and output structure of the program. Timer calls and trace-debug prints are also shown.
read namelist NLCase
if new job and BLKDAT > 0
  DATE (for FILEID)
  TIME (for FILEID)
  FILEI (input file write)

if new job
  JCASE = 1 to NCASES
  TIMER (initialize)
  DATE (for IDENT)
  TIME (for IDENT)
  if new job
    INPTN
    INIT
      \begin{tabular}{l}
        INITA \\
        INITC \\
        INITR1 \\
        INITR2 \\
        INITB \\
        INITE \\
        CHEKR1 \\
        CHEKR2
      \end{tabular}
    if old job
      INPTO
      PRNTC
      if new job or trim restart
        TRIM
        FILEJ (trim data scratch file write)
      if ANTYPE(1) \neq 0 or flutter restart
        FLUT
        FILEJ (trim data scratch file read)
      if ANTYPE(2) \neq 0 or flight dynamics restart
        STAB
        FILEJ (trim data scratch file read)
      if ANTYPE(3) \neq 0 or transient restart
        TRAN
        TIMER
      TIMER (print)
FILE1 (input file read)
read namelist NLTRIM
if OPREAD(1) ≠ 0
  INPTR1
    read namelist NLTRIM
  if OPREAD(2) ≠ 0
    INPTR1
    read namelist NLWAKE
  if OPREAD(3) ≠ 0
    INPTR2
    read namelist NLTRIM
  if OPREAD(4) ≠ 0
    INPTR2
    read namelist NLWAKE
  if OPREAD(5) ≠ 0
    INPTB
    read namelist NLLOAD
  if OPREAD(6) ≠ 0
    INPTL1
    read namelist NLLOAD
  if OPREAD(7) ≠ 0
    INPTL2
    read namelist NLLOAD
  if OPREAD(8) ≠ 0
    INPTF
    read namelist NLFLUT
  if OPREAD(9) ≠ 0
    INPTS
    read namelist NLSTAB
  if OPREAD(10) ≠ 0
    INPTT
    read namelist NLTRAN
if first case
  INPTA1
    read airfoil #1 file
  INPTA2
    read airfoil #2 file
FILEI (restart file read)
FILEJ
FILEF
FILES
FILET
read namelist NLTRIM
if OPREAD(6) ≠ 0
INPTL1
read namelist NLLOAD
if OPREAD(7) ≠ 0
INPTL2
read namelist NLLOAD
if OPREAD(8) ≠ 0
INPTF
read namelist NLFLUT
INPTG
read namelist NLFLUT
if OPREAD(9) ≠ 0
INPTS
read namelist NLSTAB
INPTU
read namelist NLSTAB
if OPREAD(10) ≠ 0
INPTT
read namelist NLTRAN
INPTV
read namelist NLTRAN
flutter restart
flight dynamics restart
transient restart
TRIM

TIMER
if trim restart, go to restart entry point
uniform inflow
if ITERU ≠ 0
    TRIMI
    if NPRNTT = 1
        PERF
        LOAD
    nonuniform inflow, prescribed wake geometry
    for IT = 1 to ITERR
        WAKECl
        WAKEC2
        TRIMI
        if IT = multiple NPRNTT
            PERF
            LOAD
    nonuniform inflow, free wake geometry
    for IT = 1 to ITERF
        WAKECl
        WAKEC2
        TRIMI
        if IT = multiple NPRNTT
            PERF
            LOAD
    FILE (restart file write)
    FILEI
    FILEJ
trim restart entry point
PRINT
    PRNTC
    if NPRNTI ≠ 0
        PRNTR1
        PRINTW1
        PRNTR2
        PRINTW2
        PRNTB
MODEP1
MODEP2
TRIMP
PERF
LOAD
TIMER
TRIM

RAMP
if MTRIM ≤ 0 or no trim iteration, return
if DEBUG(4) ≥ 1, print trim iteration
for COUNTT = 1 to MTRIM
  if COUNTT-1 = multiple MTRIM, construct D⁻¹
    for I = 1 to WT
      increment controls
      RAMP
    MINV
      increment controls
      RAMP
    if DEBUG(4) ≥ 1, print trim iteration
    test trim convergence
  CPTRIM
EPTRIM, CPTRIM

PERF

TIMER
PERF1
PERF2
TIMER
LOAD

TRANS

LOAD1

" MALOAD # 0
GEOME1
HISTPP
GEOMP1
POLRFP
HISTPP

if MHLOAD # 0
LOADM
LOADF
HISTPP

for IR = 1 to MHLOAD
LOADS1

LOADF1
LO ' A
LOADF
HISTPP

for IN = 1 to MNOISE
MUISH1

BESSEL

NPLot(68-71)

NPLot(72-75)
FLUT

for OFFLOW ≤ 0 (constant coefficients)
  if flutter restart, go to restart entry point
    FLUTM
    FILEF (restart file write)
    flutter restart entry point
    PRNTF
    MODEP1
    MODEP2
    FLUTL

  if OFFLAN ≠ 0
    STABD
    STABE

for OFFLOW > 0 (periodic coefficients)
  for NT = 0 to MFSIPC
    FLUTM
      if NT = MFSIPC
        PRNTF
        MODEP1
        MODEP2
        PSYSAN
      if NT = MFSIPC
        FILEE (eigenvalue file write)
<table>
<thead>
<tr>
<th>FLUTRM</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODE1</td>
</tr>
<tr>
<td>MODE2</td>
</tr>
<tr>
<td>FLUTR1</td>
</tr>
<tr>
<td>FLUTR2</td>
</tr>
<tr>
<td>FLUTB</td>
</tr>
</tbody>
</table>

| BODYF |

<table>
<thead>
<tr>
<th>FLUTRI</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB = NBLADE if OFFLOW &gt; 0, 1 if OFFLOW = 0, MPSICC if OFFLOW &lt; 0</td>
</tr>
<tr>
<td>for JPSI = 1 to NB</td>
</tr>
<tr>
<td>FLUTI1</td>
</tr>
<tr>
<td>FLUTA1</td>
</tr>
<tr>
<td>for IR = 1 to MRA</td>
</tr>
<tr>
<td>AEROS1</td>
</tr>
</tbody>
</table>
STAB

TIMER
PRINTS
if flight dynamics restart, go to restart entry point
STABM

for ID = 1 to 21
  increment controls or motion
  for IT = 1 to ITERS
    WAKEC1
    WAKEC2
    RAMF
    PERF
    LOAD

FILES (restart file write)
flight dynamics restart entry point
STABD
STABE
TIMER

for IEQ = 1 to 12
  DERED
  STABL

  TIMER
  CSYSAN
  FILEE (eigenvalue file write)
  BODE
  TRACKS
  GUSTS
  numerical integration
  MINV
  STABP

  PRNTP
  for IT = 1 to TMAX/TSTEP
  TRANC
    CONTU
    GUSTU
    GUSTC
  if IT = multiple NPRNTT
    STABP

  TRCKPP

  TIMER

STABE
TRAN

TIMER
PRMTT
PRNTG

if transient restart, go to restart entry point

MINV

TRANP
for IT = 1 to TMAX/TSTEP

TRANC

for IT = 1 to ITERT

CONTRCL
GUSTU
GUSTC

TRANI

if IT = multiple NFRNTT

TRANP
PERP
LOAD

if IT = multiple NRSTRT

FILET (restart file write)

transient restart entry point

-20-
for COUNTER = 1 to ITERC (circulation iteration)
  WAKE1
  WAKEN1
  WAKE2
  WAKEN2
  for COUNTER = 1 to ITERM (motion iteration)
    INRTM1
    INRTI
    INRTM2
    INRTI
    ENGM1
    INRTI
    ENGM2
    INRTI
  for JPSI = 0 to MREV * MPSI by MPSIR (Ψ loop)
    MOTNH1
    MOTNR1
    MOTNH2
    MOTNR2
    BODIV1
    ENGNV1
    MOTNF1
    BODIV2
    ENGNV2
    MOTNF2
    MOTNS

  test motion convergence
  test circulation convergence

  EPMOTN
  EPCIRC
MODE1

TIMER
MODEC1
if \( \Delta \theta > \theta_{\text{P}0} \)

\[
\text{MODEB1} \\
\text{MODEC} \\
\text{MINV} \\
\text{EIGENJ} \\
\text{MODEA1} \\
\text{MODEK1} \\
\text{MODED1}
\]

HINGE \( \neq 2 \)

MOTNR1

TIMER
for \( JP = JPSI + 1 \) to \( JPSI + \text{MPSIR} \) (\( \Psi \) step)

\[
\text{MOTNB1} \\
\text{AEROF1} \\
\text{for IR = 1 to MRA} \\
\text{AEROS1} \\
\text{AEROT1}
\]

HINGE = 2

TIMER
WAKEC1

GEOMR1

<table>
<thead>
<tr>
<th>TIMER</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEOMF1</td>
</tr>
<tr>
<td>TIMER</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td>WAKEB1</td>
</tr>
<tr>
<td>GEOME1</td>
</tr>
</tbody>
</table>

for I = 1 to MPSI (Ψ loop)
  WAKEB1
  WAKEB2
  for M = 1 to NBLADE (blade loop)
    GEOME1
    VTXL
    for K = 1 to KFW or KDW (φ loop)
      GEOME1
      VTXL
      VTXS

TIMER

LEVEL = 2

DBV ≥ 0.

DBV ≥ 0.

IN: LOW(3) = 3

-23-
5. JOB STRUCTURE

In this section the structure of a job to run the program is defined. The basic structure consists of the following steps:

1) File definition as required for job
2) Block data load for airframe and each rotor
3) Main program call
4) Namelist &NLCASE
5) Namelist &NLTRIM (for each case)
6) Component and task namelists as required

File definition parameters:

a) RET = T  Erase file at logoff
b) DISP = NEW  New file to be created
c) DISP = OLD  Existing file

Sample jobs are presented below,

New job, 2 cases: trim analysis; block data input, basic namelist input, same airfoil table for both rotors

DDEF PT50F001,SCRATCH,DISP=NEW,RET=T
DDEF PT41F001,AIRFOIL,DISP=OLD
LOAD HELA; LOAD HEL1; LOAD HEL2
CALL MAINPROG
&NLCASE JOB=0,NCASE=2,RSWRT=0,BLKDAT=-1,
NFAF1=41,NFAF2=41,NFSCR=50,NFRS=-1,NFEIG=-1,
&END
&NLTRIM VKTS=x, JLL=x, LATCYC=x, LNGCYC=x, PEDAL=x, APITCH=x, AROLL=x,
ANTYPE=3*0, OPREAD=10*0,
&END
&NLTRIM data for second case,&END
%END
New job, 1 case; trim, flutter, flight dynamics, and transient analysis; block data input, all namelist inputs, different airfoil table for each rotor; write eigenvalue file

```
DDEF FT50F001,,SCRATCH,DISP=NEW,RET=T
DDEF FT41F001,,AIRFOIL1,DISP=OLD
DDEF FT42F001,,AIRFOIL2,DISP=OLD
DDEF FT45F001,,EIGEN,DISP=NEW
LOAD HELA; LOAD HELR1; LOAD HELR2
CALL MAINPROG
&NLCASE JOB=0,NCASES=1,RSWRT=0,BLKDAT=1,
   NPAF1=41,NPAF2=42,NFSCR=50,NFRS=-1,NFEIG=45,
&END
&NLTRM data,&END
&NLTRM data,&END
&NLTRM data,&END
&NLTRM data,&END
&NLTRM data,&END
&NLTRM data,&END
&NLTRM data,&END
&NLTRAN data,&END
&END
```

```
New job, 1 case; trim analysis; block data input and write input file

```
DDEF FT50F001,,SCRATCH,DISP=NEW,RET=T
DDEF FT41F001,,AIRFOIL,DISP=OLD
DDEF FT40F001,,INPUT,DISP=NEW
LOAD HELA; LOAD HELR1; LOAD HELR2
CALL MAINPROG
&NLCASE JOB=0,NCASES=1,RSWRT=0,BLKDAT=1,
   NPAF1=41,NPAF2=41,NFSCR=50,NFRS=-1,NFEIG=-1,NFDAT=40,
&END
&NLTRAN data,&END
&END
```
New job, 1 case; trim analysis; read input file

```plaintext
DDEF FT50FOO1,SCRATCH,DISP=NEW,RET=T
DDEF FT41FOO1,AIRFOIL,DISP=OLD
DDEF FT40FOO1,INPUT,DISP=OLD
CALL MAINPROG
&NLCASE JOB=0,NCASE=1,RSWRT=0,BLKDAT=0,RDFILE=1,
     NFAF1=41,NFAF2=41,NFSCR=50,NFRS=-1,NFEIG=-1,NFDET=40,
     &END
&NLTRIM data,&END
%END
```

New job, 2 cases; trim and flutter analysis; write restart file

```plaintext
DDEF FT50FOO1,SCRATCH,DISP=NEW,RET=T
DDEF FT41FOO1,AIRFOIL,DISP=OLD
DDEF FT40FOO1,RESTART1,DISP=NEW
DDEF FT40FOO2,RESTART2,DISP=NEW
LOAD HELA; LOAD HELR1; LOAD HELR2
CALL MAINPROG
&NLCASE JOB=0,NCASE=2,RSWRT=1,BLKDAT=1,
     NFAF1=41,NFAF2=41,NFSCR=50,NFRS=44,
     &END
&NLTRIM data for first case,
     ANTYPE=1,0,0,OPREAD(8)=1,
     &END
&NLFLUT data,&END
&NLTRIM data for second case,&END
&NLFLUT data,&END
%END
```

Old job; trim restart with flutter analysis

```plaintext
DDEF FT50FOO1,SCRATCH,DISP=NEW,RET=T
DDEF FT44FOO1,RESTART,DISP=OLD
CALL MAINPROG
&NLCASE JOB=1,RSWRT=1,START=1,
     NFSCR=50,NFEIG=-1,NFRS=44,
     &END
&NLTRIM ANTYPE=1,0,0,OPREAD(8)=1,
     &END
&NLFLUT data,&END
%END
```
Old job; flutter restart

DDEF FT5OF001,,SCRATCH,DISP=NEW,RET=T
DDEF FT44F001,,RESTART,DISP=OLD
CALL MAINPROG
  &NLCASE JOB=1,RSWRT=0,START=2,
  NPSCE=50,NFREI=-1,NFRC=44,
  &END
  &NLTRIM OREAD(8)=1,
  &END
  &NLFLTUT data,&END
&END
6. INPUT DESCRIPTION

In this section the input variables for the program are defined. The variables are categorized according to the namelist that reads them. The program namelist labels are listed in the table below.

<table>
<thead>
<tr>
<th>Namelist</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLCAKE</td>
<td>Job data</td>
</tr>
<tr>
<td>NLTRIM</td>
<td>Trim data</td>
</tr>
<tr>
<td>NLRTR</td>
<td>Rotor data</td>
</tr>
<tr>
<td>NLWAKE</td>
<td>Wake data</td>
</tr>
<tr>
<td>NLBODY</td>
<td>Airframe and drive train data</td>
</tr>
<tr>
<td>NLLOAD</td>
<td>Loads data</td>
</tr>
<tr>
<td>NLFLUT</td>
<td>Flutter data</td>
</tr>
<tr>
<td>NLSTAB</td>
<td>Flight dynamics data</td>
</tr>
<tr>
<td>NLTRAN</td>
<td>Transient data</td>
</tr>
</tbody>
</table>

The corresponding common block labels, for the block data form of input, may be obtained from Section 3. In the description of the input parameters for the rotor, the variables NBH and NTM are used:

a) NBH is the index of the highest-frequency blade bending mode used in the analysis;

b) NTM is the index of the highest-frequency blade torsion mode used in the analysis.
**NameList NLCASE**

**JOB**
- integer parameter defining job: EQ 0 for new job (default); NE 0 for old job or restart (one case only)

**RSWRT**
- integer parameter controlling restart file write: 0 to suppress write (default)

**NCASES**
- number of cases (default = 1)

**BLKDA1**
- integer parameter defining input source:
  - EQ 0 read input file (default)
  - GT 0 use loaded block data and write input file
  - LT 0 use loaded block data

**RDFILE**
- integer parameter controlling input file read:
  - EQ 0 read file for first case only
  - NE 0 read file for every case (default)

**START**
- integer parameter defining task:
  - 1 for trim restart (default)
  - 2 for flutter restart
  - 3 for flight dynamics restart
  - 4 for transient restart
- trim restart can be followed by any or all of the other tasks (as defined by ANTYPE); for flutter, flight dynamics, or transient restart, only that task can be done

**Input/output unit numbers**

**NFDAT**
- input data file (new job only); default = 40

**NFAF1**
- rotor #1 airfoil file (new job only); default = 41

**NFAF2**
- rotor #2 airfoil file (new job only; only if have two rotors); default = 42

**NFRS**
- restart file (no file write if LE 0); default = 44

**NFEIG**
- eigenvalue file (no file write if LE 0); default = 45

**NFSCR**
- scratch file; default = 50

**NUIN**
- namelist input; default = 5

**NUOUT**
- printer (and debug level 1); default = 6

**NUDB**
- debug output (levels 2 and 3); default = 6

**NUPP**
- printer-plots; default = 6

**NULIN**
- linear system analysis; default = 6
**Namelist NLTRIM**

**OPREAD(10)**

integer vector defining namelist read structure: $\text{EQ } 0$
to suppress read:

components (new job only)

1. NLRTR, rotor #1
2. NLWAKE, rotor #1
3. NLRTR, rotor #2
4. NLWAKE, rotor #2
5. NLBODY
6. NLLCAD, rotor #1
7. NLLOAD, rotor #2
8. NLFLUT
9. NLSTAB
10. NLTRAN

**NFRNTI**

integer parameter controlling input data print: $\text{EQ } 0$
for short form print only

**ANTYPE(3)**

integer vector defining tasks for new job or trim restart; $\text{EQ } 0$
to suppress:

1. flutter
2. flight dynamics
3. transient

**TITLE(20)**

title for job and case (80 characters)

**CODE**

alphanumeric code for job and case identification;
4 characters

**OPUNIT**

integer parameter designating unit system: 1 for
English units (ft-slug-sec); 2 for metric units (m-kg-sec)

**NROTOR**

number of rotors
DEBUG(25)  integer vector controlling debug print:

0  no debug print
1  trace print
2  low level print
3  high level print

(1) time (sec) at which debug print enabled
(2) input, 2-3 (INPTx)
(3) initialization, 2 (INITC, INITR, INITB, INITE)
(4) trim iteration, 1-2 (TRIMI)
(5) loads, 2 (LOADI)
(6) flutter matrices, 2-3 (FLUTM)
(7) flutter coefficients, 2-3 (FLUTI, FLUTA)
(8) flight dynamics, 2-3 (STABM, STABE)
(9) transient, 2 (TRANI)
(10) rotor/airframe motion and forces, 2-3 (RAMP)
(11) blade modes, 2 (MODE, MODEX)
(12) inertia coefficients, 2 (INRTC)
(13) airframe constants and matrices, 2 (BODYC, ENGNC, MOTNC, BODYM, ENGNM)
(14) induced velocity, 2 (WAKEU, WAKEN)
(15) rotor matrices, 2-3 (INRTM)
(16) hub/airframe motion and generalized forces, 2 (MOTNH, BODYV, ENGNV, MOTNF, MOTNS)
(17) rotor motion, 2-3 (MOTNR)
(18) rotor aerodynamics, 2-3 (AEROF)
(19) blade section aerodynamics, 3 (AERCS)
(20) body forces and aerodynamics, 2 (BODYF)
(21) wake influence coefficients, 2 (WAKEC)
(22) vortex line and sheet, 3 (VTXL, VTXS)
(23) prescribed wake geometry, 2-3 (GEOMR)
(24) free wake geometry, 1-3 (GEOMF)
(25) timer, 1 (TIMER)

-32-
VKTS  aircraft speed \( V \) (knots)

VEL   velocity ratio \( V/N \)

input either VEL or VKTS by namelist; if neither parameter is defined, \( V = 0 \) is used

VTIP  rotor #1 tip speed \( \Omega R \) (ft/sec or m/sec)

RPM   rotor #1 rotational speed (rpm)

input either VTIP or RPM by namelist; if neither parameter is defined, the normal tip speed VTIP is used; rotor #2 speed is calculated from the gear ratio TRATIO

OPDENS integer parameter defining specification of aerodynamic environment: if 1, given altitude and standard day; if 2, given altitude and temperature; if 3, given density and temperature

ALTMSL altitude above mean sea level (ft or m), for OPDENS = 1 or 2

TEMP  air temperature (°F or °C), for OPDENS = 2 or 3

DENSE air density (slug/ft\(^3\) or kg/m\(^3\)), for OPDENS = 3

OPGRND integer parameter controlling ground effect analysis: EQ 0 for out of ground effect, NE 0 for in ground effect

HAGL  altitude helicopter center of gravity above ground for ground effect analysis (ft or m)

OPENGN integer parameter specifying engine state: 1 for autorotation (engine inertia, engine damping, and throttle control torque zero; no engine speed degree of freedom); 2 for engine out (engine damping and throttle control torque zero); 0 for normal operation

AFLAP wing flap angle \( \delta_F \) (deg)

RTURN for free flight, trim turn rate \( \phi_F \) (deg/sec), positive to right
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COll</td>
<td>collective stick displacement $\delta_0$ or $\Delta \Theta_{gov}$ (deg), positive up</td>
</tr>
<tr>
<td>LATCYC</td>
<td>lateral cyclic stick displacement $\delta_c$ (deg), positive left</td>
</tr>
<tr>
<td>LNGCYC</td>
<td>longitudinal cyclic stick displacement $\delta_s$ (deg), positive aft</td>
</tr>
<tr>
<td>PEDAL</td>
<td>pedal displacement $\delta_p$ (deg), positive to right</td>
</tr>
<tr>
<td>APITCH</td>
<td>for free flight, aircraft pitch angle $\Theta_{FP}$ (deg), positive nose up; for wind tunnel, rotor shaft angle of attack $\Theta_T$ (deg), positive nose up</td>
</tr>
<tr>
<td>AROLL</td>
<td>for free flight, aircraft roll angle $\phi_{FP}$ (deg), positive to right</td>
</tr>
<tr>
<td>ACLIMB</td>
<td>for free flight, aircraft climb angle $\Theta_{FP}$ (deg), positive up</td>
</tr>
<tr>
<td>AYAW</td>
<td>for free flight, aircraft yaw angle $\Psi_{FP}$ (deg), positive to right; for wind tunnel, test module yaw angle $\Psi_T$ (deg), positive to right</td>
</tr>
<tr>
<td>MPSI</td>
<td>number of azimuth steps per revolution in motion and loads analysis, maximum 36; for nonuniform inflow must be multiple of number of blades; for free wake geometry, maximum 24</td>
</tr>
<tr>
<td>MPSIR</td>
<td>in harmonic motion solution, number of azimuth steps between update of airframe vibration and rotor matrices</td>
</tr>
<tr>
<td>MREV</td>
<td>in harmonic motion solution, number of revolutions between tests for motion convergence</td>
</tr>
<tr>
<td>ITERM</td>
<td>maximum number of motion iterations</td>
</tr>
<tr>
<td>EPMOTN</td>
<td>tolerance for motion convergence (deg)</td>
</tr>
<tr>
<td>ITERC</td>
<td>maximum number of circulation iterations</td>
</tr>
<tr>
<td>EPCIRC</td>
<td>tolerance for circulation convergence ($\Delta C_T/\psi$)</td>
</tr>
</tbody>
</table>
integer vector defining degrees of freedom used in vibratory motion solution, 0 if not used; order:

\[
\begin{align*}
\text{rotor } #1 & \quad q_1 \ldots q_{10} \quad p_0 \ldots p_4 \quad \beta_G \\
\text{rotor } #2 & \quad q_1 \ldots q_{10} \quad p_0 \ldots p_4 \quad \beta_G \\
\text{(bending, max 10)} & \quad \text{(torsion, max 5)} \quad \text{(gimbal/teeter)}
\end{align*}
\]

airframe \( \phi_F \quad \theta_F \quad \psi_F \quad x_F \quad y_F \quad z_F \quad q_{57} \ldots q_{16} \) 
(rigid body) 
(flexible body, max 10)

drive train \( \psi_s \quad \psi_i \quad \psi_e \quad \Delta \theta_t \quad \Delta \theta_{govr1} \quad \Delta \theta_{govr2} \) 
(rotor/engine speed) 
(governor)

integer vector defining blade bending degrees of freedom used for mean deflection (subset of DOF), 0 if not used; order:

\[
\begin{align*}
\text{rotor } #1 & \quad q_1 \quad q_2 \quad q_3 \quad q_4 \\
\text{rotor } #2 & \quad q_1 \quad q_2 \quad q_3 \quad q_4 \\
\text{(bending, max 4)}
\end{align*}
\]

number of harmonics in rotor motion analysis; maximum 20; EQ 0 for mean only

\[
\begin{align*}
(1) & \quad \text{rotor } #1 \\
(2) & \quad \text{rotor } #2
\end{align*}
\]

number of harmonics in airframe vibration analysis (harmonics of N/rev); maximum 10; EQ 0 for static elastic only; suggest LE MHARM/NBLADE, and the same value for both rotors if coupled hub vibration used (see CPHVIB)

\[
\begin{align*}
(1) & \quad \text{rotor } #1 \\
(2) & \quad \text{rotor } #2
\end{align*}
\]

integer parameter specifying rotor wake analysis level:
0 for uniform inflow, 1 for nonuniform inflow with prescribed wake geometry, 2 for nonuniform inflow with free wake geometry (must be consistent with INFLOW)

\[
\begin{align*}
(1) & \quad \text{rotor } #1 \\
(2) & \quad \text{rotor } #2
\end{align*}
\]
number of wake and trim iterations

- **ITERU**: at uniform inflow level; EQ 0 to skip
- **ITERR**: at non-uniform inflow/prescribed wake geometry level; EQ 0 to skip
- **ITERF**: at non-uniform inflow/free wake geometry level

**NPRNTT**: integer parameter $n_i$; trim/performance/load print every $n$-th iteration; LE 0 to suppress

**NPRNTP**: integer parameter controlling performance print; LE 0 to suppress

**NPRNTL**: integer parameter controlling loads print; LE 0 to suppress

**MTRIM**: maximum number of iterations on controls to achieve trim

**MTRIMD**: number of trim iterations between update of trim derivative matrix

**DELTA**: control step in trim derivative calculation (stick displacement, deg)

**FACTOR**: factor reducing control increment in order to improve trim convergence (typically 0.5)

**EPTRIM**: tolerance on trim convergence

**OPGOVT**: integer parameter specifying governor trim

- 0: trim collective stick $\delta_0$
- 1: trim rotor #1 governor
- 2: trim rotor #2 governor
- 3: trim both rotor governors

**CXTRIM** $C_x/\text{ft}$

**XTRIM** $X/q$ (ft or m)

**CTTRIM** $C_T/\text{ft}$ or $C_L/\text{ft}$

**CPTRIM** $C_P/\text{ft}$

**CYTRIM** $C_Y/\text{ft}$

**BCTRIM** $\delta_e$ (deg)

**BSTRIM** $\delta_s$ (deg)
integer parameter specifying trim option

free flight cases

OPTRIM = 0  no trim

1 trim forces and moments with

$\delta_o \delta_c \delta_s \delta_P \theta_{FT} \phi_{FT}$

2 trim forces and moments with

$\delta_o \delta_c \delta_s \delta_P \theta_{FT} \psi_{FP}$

3 trim forces, moments, and power with

$\delta_o \delta_c \delta_s \delta_P \theta_{FT} \phi_{FP}$

4 trim forces, moments, and power with

$\delta_o \delta_c \delta_s \delta_P \theta_{FT} \psi_{FP} \theta_{FP}$

5 trim symmetric forces and moments with

$\delta_o \delta_s \theta_{FT}$

6 trim symmetric forces, moments, and power with

$\delta_o \delta_s \theta_{FT} \theta_{FP}$

wind tunnel cases

OPTRIM = 10  no trim

11 trim $C_T$ with $\delta_o$

12 trim $C_\alpha$ with $\theta_T$

13 trim $C_L$ with $\delta_o$

14 trim $\beta_c$ with $\delta_c \delta_s$

15 trim $C_T \beta_c \beta_s$ with $\delta_o \delta_c \delta_s$

16 trim $C_L C_\alpha$ with $\delta_o \delta_c \delta_s$

17 trim $C_L C_\alpha \beta_c \beta_s$ with $\delta_o \delta_c \delta_s \theta_T$

18 trim $C_L X/q$ with $\delta_o \delta_c \delta_s$

19 trim $C_L X/q \beta_c \beta_s$ with $\delta_o \delta_c \delta_s \theta_T$

20 trim $C_L X/q \theta_T$ with $\delta_o \delta_c \delta_s$

21 trim $C_L X/q \theta_T$ with $\delta_o \delta_c \delta_s \theta_T$

22 trim $C_\alpha$ with $\delta_s$

23 trim $C_\alpha \beta_c$ with $\delta_o \delta_s$

24 trim $C_\alpha \beta_s$ with $\delta_o \delta_s$

25 trim $C_\alpha \beta_s$ with $\delta_o \theta_T$

26 trim $C_\alpha \beta_c \beta_s$ with $\delta_o \delta_s \theta_T$

27 trim $C_\alpha X/q$ with $\delta_o \delta_s$

28 trim $C_\alpha X/q \theta_T$ with $\delta_o \theta_T$

29 trim $C_\alpha X/q \theta_T$ with $\delta_o \delta_s \theta_T$
see namelist NLBCUDY
**Namelist NLRTR**

**TITLE**(20)  
- title for rotor and wake data (80 characters)

**TYPE**  
- rotor identification (4 characters); suggest MAIN, FRNT, or RGHT for rotor #1; and TAIL, HRR, or LEFT for rotor #2

**VTIPN**  
- normal tip speed \( \Omega_R \) (ft/sec or m/sec)

**RADIUS**  
- blade radius \( R \) (ft or m)

**SIGMA**  
- solidity ratio \( \sigma = \frac{N_c}{\pi R} \) (based on mean chord)

**GAMMA**  
- blade Lock number \( \gamma = \frac{\rho a c R^4}{I_b} \) (based on standard density, \( a = 5.7 \), and mean chord)

\((\gamma \text{ and } \sigma \text{ are only used to calculate the normalization parameters } c_m \text{ and } I_b)\)

**NBLADE**  
- number of blades

**TDAMPO**  
- control system damping (ft-lb/rad/sec or m-N/rad/sec)

**TDAMPC**  
- cyclic

**TDAMFR**  
- rotating

**NUGC**  
- longitudinal gimbal natural frequency \( \omega_G \) or teeter natural frequency \( \omega_T \) (per rev at normal tip speed \( \Omega_R \))

**NUGS**  
- lateral gimbal natural frequency \( \omega_G \) (per rev at normal tip speed \( \Omega_R \))

**GDAMPC**  
- longitudinal gimbal damping \( C_G \) or teeter damping \( C_T \)

\((ft-lb/rad/sec or m-N/rad/sec)\)

**GDAMPS**  
- lateral g: bal damping \( C_G \)

\((ft-lb/rad/sec or m-N/rad/sec)\)

**LDAMPC**  
- linear lag damper coefficient \( C_L \)

\((ft-lb/rad/sec or m-N/rad/sec); \text{ estimated damping if a nonlinear damper is used (LDAMPM GT 0.)}; \text{ the lag mode has structural damping also (GSB)}\)

**LDAMPM**  
- maximum moment of nonlinear lag damper, \( M_{LD} \)

\((ft-lb or m-N); \text{ linear lag damper used if LDAMPM EQ 0.})\)

**LDAMPR**  
- lag velocity \( \xi_L \) where maximum moment of lag damper occurs

\((rad/sec); \text{ hydraulic damping below } \xi_{LD} \) and friction damping above

**GSB**  
- bending mode structural damping \( g_a \)

**GST**  
- torsion mode structural damping \( g_a \)

**ROTATE**  
- integer parameter specifying rotor rotation direction:

\(1 \text{ for counter-clockwise, } -1 \text{ for clockwise (viewed from above)}\)
OPHIV(3) integer parameter controlling hub vibration contributions; gravity and static velocity terms always retained; 0 to suppress:

1. vibration due to this rotor
2. vibration due to other rotor (must suppress if \( \Omega_2/\Omega_1 \neq 1 \))
3. static elastic motion

RTIP tip loss parameter B

OPTIP integer parameter specifying tip loss type: 1 for tip loss factor, 2 for Prandtl function

LNTW integer parameter specifying twist type: 0 for nonlinear twist, 1 for linear twist

TWISTL linear twist rate \( \Theta_{tw} \) (deg); used to calculate TWISTA and TWISTL if LNTW = 0

OPUSLD integer parameter controlling use of unsteady lift, moment, and circulation terms: if 0, suppress; if 1, include; if 2, zero for stall (15° < \( |\alpha| < 165° \))

OPCOMP integer parameter controlling aerodynamic model, 0 for incompressible loads

Inflow model

INFLOW(6) integer vector defining induced velocity calculation (must be consistent with LEVEL)

1. at this rotor: 0 for uniform, 1 for nonuniform
2. at other rotor: 0 for zero, 1 for empirical, 2 for average at hub, 3 for nonuniform (only if \( \Omega_2/\Omega_1 = 1 \))
3. at wing-body: 0 for zero, 1 for empirical, 2 for nonuniform
4. at horizontal tail: 0 for zero, 1 for empirical, 2 for nonuniform
5. at vertical tail: 0 for zero, 1 for empirical, 2 for nonuniform
6. at point off rotor disk: 0 for zero, 1 for nonuniform

RRCOT root vortex position for wake model, \( r_{root}/R \)

RCMAX \( r_{max}/R \) (induced velocity calculated using maximum bound circulation magnitude outboard of \( r_{max} \))
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRA</td>
<td>number of aerodynamic segments; maximum 30</td>
</tr>
<tr>
<td>RAE(MRA+1)</td>
<td>radial stations r/R at edges of aerodynamic segments; sequential, from root to tip</td>
</tr>
<tr>
<td>CHORD(MRA)</td>
<td>blade chord, c/R</td>
</tr>
<tr>
<td>XA(MRA)</td>
<td>offset of aerodynamic center aft of elastic axis, xₐ/R; xₐ is the point about which the moment data in the airfoil tables is given</td>
</tr>
<tr>
<td>THET2L(MRA)</td>
<td>incremental pitch of zero lift line, Θ₂L (deg); can be included in TWISTA; Θ₂L is the pitch of the axis corresponding to zero angle of attack in the airfoil tables, relative to the twist angle (TWISTA)</td>
</tr>
<tr>
<td>TWISTA(MRA)</td>
<td>blade twist relative .75R, Θₜw (deg)</td>
</tr>
<tr>
<td>XAC(MRA)</td>
<td>offset of aerodynamic center (for unsteady aerodynamics) of elastic axis, XₐC/R</td>
</tr>
<tr>
<td>MCORRL(MRA)</td>
<td>Mach number correction factor fₘ = M_eff/M for lift</td>
</tr>
<tr>
<td>MCORRD(MRA)</td>
<td>Mach number correction factor fₘ = M_eff/M for drag</td>
</tr>
<tr>
<td>MCORRM(MRA)</td>
<td>Mach number correction factor fₘ = M_eff/M for moment</td>
</tr>
</tbody>
</table>

| MRI  | number of radial stations where characteristics defined; maximum 51         |
| RI(MRI) | radial stations r/R; sequential, from root to tip, RI(1) = 0, and RI(MRI) = 1. |
| MASS(MRI) | section mass, m (slug/ft or kg/m)                                     |
| E1XX(MRI) | chordwise bending stiffness (lb-ft² or N-m²)                            |
| E1ZZ(MRI) | flapwise bending stiffness (lb-ft² or N-m²)                             |
| XI(MRI) | offset of center of gravity aft of elastic axis, x₁/R                  |
| XC(MRI) | offset of tension center aft of elastic axis, x₂C/R (at the tip, XC should be set nearly equal XI) |
| KP2(MRI) | polar radius of gyration about elastic axis, k²P/R²                   |
| ITHETA(MRI) | section moment of inertia about elastic axis, I₀ (slug-ft or kg-m)  |
| GJ(MRI) | torsional stiffness, GJ (lb-ft² or N-m²)                               |
| TWISTI(MRI) | blade twist relative .75R, Θₜw (deg)                                    |
Stall model

**OPSTLL**
integer parameter defining stall model
0 no stall
1 static stall
2 McCroskey stall delay
3 McCroskey stall delay with dynamic stall vortex loads
4 Boeing stall delay
5 Boeing stall delay with dynamic stall vortex loads
(the stall delay can be suppressed by setting TAU=0.)

**OPYAW**
integer parameter defining yawed flow corrections
0 both yawed flow and radial drag included
1 no yawed flow (cos $\Lambda = 1$.)
2 no radial drag ($F_r = 0$.)
3 neither yawed flow nor radial drag included

**TAU(3)**
stall delay time constants for lift, drag, and moment:
$\tau_L$, $\tau_D$, $\tau_M$ (calculated if LT 0.)

**ADELAY**
maximum angle of attack increment due to stall delay,
$\alpha_{\text{max} \text{ delay}}$ (deg)

**AMAXNS**
angle of attack in linear range for no stall model, $\alpha_{\text{max}}$ (deg)

**FSIDS(3)**
dynamic stall vortex load rise and fall time (azimuth increment)
for lift, drag, and moment: $\Delta \Psi_{ds}$ (deg)

**ALFDS(3)**
dynamic stall angle of attack for lift, drag, and moment:
$\alpha_{ds}$ (deg)

**ALFRB(3)**
stall recovery angle of attack for lift, drag, and moment:
$\alpha_{\text{re}}$ (deg)

**CLDSP**
maximum peak dynamic stall vortex induced lift coefficient:
$\Delta c_L ds$

**CDDSP**
maximum peak dynamic stall vortex induced drag coefficient:
$\Delta c_D ds$

**CMDSP**
maximum peak dynamic stall vortex induced moment coefficient:
$\Delta c_M ds$
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KFLMDA</td>
<td>factor $k_f$ for forward flight induced velocity (typically 1.2)</td>
</tr>
<tr>
<td>FXLMDA</td>
<td>factor $f_x$ for linear inflow variation in forward flight (typically 1.5)</td>
</tr>
<tr>
<td>FYLMDA</td>
<td>factor $f_y$ for linear inflow variation in forward flight (typically 1.)</td>
</tr>
<tr>
<td>FMLMDA</td>
<td>factor $f_m$ on linear inflow variation due to hub moment (typically 1.)</td>
</tr>
<tr>
<td>FACTUW</td>
<td>factor introducing lag in $C_p$, $C_{M_x}$, and $C_{M_y}$ used to calculate induced velocity (typically .5)</td>
</tr>
<tr>
<td>KINTH</td>
<td>factor for hover interference velocity at other rotor ($K_{21}$ or $K_{12}$)</td>
</tr>
<tr>
<td>KINTF</td>
<td>factor for forward flight interference velocity at other rotor ($K_{21}$ or $K_{12}$) (linear variation between KINTH at $\alpha = 0.05$ and KINTF at $\alpha = 0.10$ is used)</td>
</tr>
<tr>
<td>KINTWB</td>
<td>factor for rotor-induced interference velocity at wing-body, $K_w$</td>
</tr>
<tr>
<td>KINTHT</td>
<td>factor for rotor-induced interference velocity at horizontal tail, $K_H$</td>
</tr>
<tr>
<td>KINTVT</td>
<td>factor for rotor-induced interference velocity at vertical tail, $K_V$ ($K_w$, $K_H$, $K_V$ equal fraction of fully-developed wake times maximum fraction surface in wake)</td>
</tr>
<tr>
<td>HINGE</td>
<td>integer parameter specifying blade mode type</td>
</tr>
<tr>
<td></td>
<td>0 hinged</td>
</tr>
<tr>
<td></td>
<td>1 cantilever</td>
</tr>
<tr>
<td></td>
<td>2 articulated (flap and lag modes only)</td>
</tr>
<tr>
<td>NCOOLB</td>
<td>number of collocation functions for bending mode calculations (total flap and lag, alternating); maximum 20</td>
</tr>
<tr>
<td>NCCLT</td>
<td>number of collocation functions for torsion mode calculations; maximum 10</td>
</tr>
<tr>
<td>NONROT</td>
<td>integer parameter: NE 0 to calculate nonrotating bending frequencies</td>
</tr>
<tr>
<td>EPMODE</td>
<td>criterion on change of collective pitch to update blade modes, $\Delta \Theta = 75$ (deg)</td>
</tr>
</tbody>
</table>
MASST  tip mass (slug or kg); the tip mass can also be included directly in the section mass distribution
XIT    offset of tip mass center of gravity aft of elastic axis, $x_T/R$
MBLADE blade mass (slug or kg); if $L6 = 0.$, integral of section mass used (with mass included at $r = 0.$ to account for the hub mass)
KFLAP flap hinge offset $e_f/R$ (extent of rigid hub for cantilever blade)
KLAG  lag hinge offset $e_l/R$ (extent of rigid hub for cantilever blade)
KFLAP flap hinge spring (ft-lb/rad or m-N/rad)
KLAG  lag hinge spring (ft-lb/rad or m-N/rad)
RCPLS hinge spring parameter, $\Theta_S$
TSPRNG hinge spring parameter, $\Theta_{SO}$
(RCPL structural coupling parameter $\Theta$ (effective pitch angle $\Theta$ used to calculate blade bending modes; normally $\Theta = 1.$)
NCPB integer parameter specifying twist inboard of $r_{FA}$: EQ 1 for no pitch bearing
WTIN integer parameter defining control system stiffness input:
  1 for $K_{\Theta}$, 2 for $\omega_{\Theta}$
  control system frequency $\omega_{\Theta}$ (per rev, at normal tip speed VTIPN)
FTO  collective
FTC  cyclic
FTR  reactionless
   control system stiffness $K_{\Theta}$ (ft-lb/rad or m-N/rad)
KTO  collective
KTC  cyclic
KTR  reactionless
KPIN integer parameter defining pitch/bending coupling input:
  1 for input, 2 for calculated (negative to suppress cosine factors in $K_{P\ell}$ and $K_{PC}$)
PHIPH pitch horn cant angle, $\Phi_{PH}$ (deg)
PHIPL pitch link cant angle, $\Phi_{PL}$ (deg)
RPB  pitch bearing radial location, $r_{PB}/R$
RPH  pitch horn radial location, $r_{PH}/R$
XPH  pitch horn length, $x_{PH}/R$
pitch/bending coupling \( \tan^{-1} K_{p1} \) (deg), for pitch horn level \((KPIN = 1 \text{ or } -1)\)

pitch/gimbal coupling \( \tan^{-1} K_{pG} \) (deg), for pitch horn level

feathering axis radial location, \( r_{FA/R} \)

gimbal undersling, \( z_{FA/R} \)

torque offset, \( x_{FA/R} \)

precone angle \( \delta_{FA1} \) (deg), positive up

droop angle \( \delta_{FA2} \) (deg) at \( \Theta_{75} = 0 \), positive down from precone

sweep angle \( \delta_{FA3} \) (deg) at \( \Theta_{75} = 0 \), positive aft

feathering axis droop angle \( \delta_{FA4} \) (deg), positive down from precone

feathering axis sweep angle \( \delta_{FA5} \) (deg), positive aft
Namelist NLWAKE

FACTWN factor introducing lag in bound circulation used to calculate induced velocity

OPVXVY integer parameter: EQ 0 to suppress x and y components of induced velocity calculated at the rotors

KNW extent of near wake, $K_{NW}$

KRW extent of rolling up wake, $K_{RW}$

KFW extent of far wake and tip vortices, $K_{FW}$

KDW extent of far wake and tip vortices for points off rotor disk, $K_{DW}$

(initial $\phi = K_\Delta \Psi$; all $K \geq 1$)

RRU initial radial station of wake rollup, $r_{RU}/R$

FRU initial tip vortex fraction of $\Gamma_{max}$ for rollup, $f_{RU}$

FRU extent of rollup in wake age, $\phi_{RU}$ (deg)

FNW tip vortex fraction of $\Gamma_{NW}$ for near wake, $f_{NW}$

DVS sheet edge test parameter $d_{vs}$; LT 0. to suppress test

DLS lifting surface correction parameter $d_{Ls}$; LT 0. to suppress correction

CORE(5) vortex core radii $r/R$

(1) tip vortices

(2) burst tip vortices

(3) tip vortices in far wake off rotor

(4) trailed lines (LT 0. for default = s/2)

(5) shed lines (LT 0. for default = t/2)

OPCORE(2) integer parameter specifying vortex core type: 0 for distributed vorticity, 1 for concentrated vorticity

(1) tip vortices

(2) inboard wake

OPNWS(2) integer parameter controlling action when inflow and circulation points coincide in near wake ($\phi = 0$) and sheets are being used: 0 to use two sheets, 1 to use lines, 2 to use single sheet

(1) shed wake

(2) trailed wake

LHW number of spirals of far wake for axisymmetric case, $L_{HW}$

OPHW integer parameter: EQ 0 for axisymmetric wake geometry

OPRTS integer parameter: NE 0 to include rotation matrices ($R_{TS}$, etc.) in influence coefficients
integer parameter defining wake model: 0 to omit element, 1 for line segment with stepped circulation distribution, 2 for line segment with linear circulation distribution, 3 for vortex sheet element

1. tip vortices (stepped line or linear line)
2. near wake shed vorticity
3. near wake trailed vorticity
4. rolling up wake shed vorticity
5. rolling up wake trailed vorticity
6. far wake shed vorticity
7. far wake trailed vorticity
8. far wake (off rotor) shed vorticity
9. far wake (off rotor) trailed vorticity
10. bound vortices (no sheet model)
11. axisymmetrical wake axial vorticity (no line model)
12. axisymmetrical wake shed vorticity (no line model)
13. axisymmetrical wake ring vorticity (no line model)

MRG
number of circulation points for near wake; LE MRA

NG(MRG)
circulation points, identified by aerodynamic segment number: \( n_{Q_i} \) for \( i = 1 \) to MRG (corresponding \( r_i \) must be between \( r_{\text{root}}/R \) and 1.)

MRL
number of inflow points; LE MRA

NL(MRL)
points at which the induced velocity is calculated, identified by aerodynamic segment number: \( n_{L_i} \) for \( i = 1 \) to MRL

CFWKBP(3)integer parameter controlling blade position model for wake analysis

1. \( \text{EQ } 0 \) to suppress inplane motion
2. \( \text{EQ } 0 \) to suppress all harmonics except mean
3. \( \text{EQ } 0 \) for linear from \( r = r_{\text{root}}/R \) to \( r = 1. \)
VELB  core burst propagation rate, $v_b = \frac{\Delta \phi}{\Delta \psi}$

DPHIB  core burst age increment, $\Delta \phi_b$ (deg)

LBV  core burst test parameter $d_{bv}$, LT 0. to suppress bursting

QDEBUG  velocity criterion for debug print; print if $|V_{v} \cdot k / \eta| > \text{QDEBUG}$

**Prescribed wake geometry**

**KRWG**  extent of prescribed wake geometry, $K_{RWG}$ (age $\phi = K \Delta \psi$); maximum $144$

**OPRWG**  integer parameter defining prescribed wake geometry model
1 from $K_1 = f_1 \lambda$, $K_2 = f_2 \lambda$, input $K_3$, input $K_4$
2 option #1, without interference velocity in $\lambda$
3 from input $K_1$, $K_2$, $K_3$, $K_4$

Landgrebe prescribed wake geometry
4 from $C_T$
5 from $\eta_{max}$
6 from $\lambda$
7 from $\lambda$ without interference

Kocurek and Tangler prescribed wake geometry
8 from $C_T$
9 from $\eta_{max}$
10 from $\lambda$
11 from $\lambda$ without interference

**Factors $f_1$ and $f_2$ for prescribed wake geometry**

**FWGT(4)**  tip vortex

**FWGSI(2)**  inside sheet edge

**FWGSO(2)**  outside sheet edge

**KWGT(4)**  tip vortex

**KWGSI(4)**  inside sheet edge

**KWGSO(4)**  outside sheet edge

-48-
Free wake geometry

KFWG
extent of free wake geometry distortion calculation, \( k_{FWG} \)
(age \( \Phi = K_{FWG} \); suggest \( .4/\mu \)) MPSI; maximum 96, multiple MPSI

OPFWG
integer parameter defining free wake geometry model
1 Scully free wake geometry
2 option #1, without interference velocity

ITERWG
number of wake geometry iterations; suggest 2 or 3

FACTWG
factor introducing lag in distortion calculation to improve convergence; suggest 0.5

RTWG(2)
radial station \( r/R \) of trailed vorticity
   (1) inside sheet edge
   (2) outside sheet edge, or trailed line (suggest .4)

WGMODL(2)
integer parameter defining wake model: 0 to omit, 1 for line segment, 2 for sheet element
   (1) inboard trailed wake elements
   (2) shed wake elements

CCREWG(4)
vortex core radii \( r/R \)
   (1) tip vortices
   (2) burst tip vortices (LE 0. for default = unburst value)
   (3) inboard trailed lines (LE 0. for default = \( \frac{3}{2}(RTWG(2) - RTWG(1)) \))
   (4) shed lines (LE 0. for default = 0.4 \( \Delta \Psi \))

MRVBWG
number of wake revolutions used below point where induced velocity is being calculated; suggest 2

LDMWG
integer parameter \( \Delta \Psi \): general update every \( \Delta \Psi \) increment in boundary age; suggest 180°/\( \Delta \Psi \)

NDMWG(MPSI)
integer parameter \( n_{DM}(\Psi_j) \): boundary update every \( n_{DM} \) increment in age, function of \( \Psi_j = \Delta \Psi_j \), \( j = 1 \) to MPSI; suggest 90°/\( \Delta \Psi \) fore and aft, and 45°/\( \Delta \Psi \) on sides

DQWG(2)
incremental velocity criteria; suggest 0.04 \( \Delta \Psi \) to 0.08 \( \Delta \Psi \)
   (1) near wake elements defined by \( |\Delta \Psi| > DQWG(1) \)
   (2) integrate bound vortex line in time over if \( |\Delta \Psi| > DQWG(2) \)

-49 ORIGIANL PAGE  IS
OF POOR QUALITY
IPWGI\(\text{B}(2)\) integer parameters controlling debug level 3 print of wake geometry distortion

1. IPR: print distortion before general update every \(\text{IPR} \times \Delta \Phi\); EQ 0 to suppress
2. INPS: print distortion after each iteration every \(\text{INPS} \times \Delta \Phi\); EQ 0 to suppress; last iteration printed in full

QWGI\(\text{B}\) parameter controlling debug level 3 print: induced velocity contribution of wake element printed if \(|\Delta \Phi| > \text{QWGI}\(\text{B}\)\); suggest 0.5\(\lambda_1\) to 1.0\(\lambda_1\)
<table>
<thead>
<tr>
<th>Namelist</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TITLE(20)</strong></td>
<td>title for airframe and drive train data (80 characters)</td>
</tr>
<tr>
<td><strong>WEIGHT</strong></td>
<td>aircraft gross weight including rotors (lb or kg)</td>
</tr>
<tr>
<td><strong>IXX</strong></td>
<td>aircraft moments of inertia including rotors (slug-ft² or kg-m²)</td>
</tr>
<tr>
<td><strong>IYY</strong></td>
<td></td>
</tr>
<tr>
<td><strong>IZZ</strong></td>
<td></td>
</tr>
<tr>
<td><strong>IXY</strong></td>
<td></td>
</tr>
<tr>
<td><strong>IXZ</strong></td>
<td></td>
</tr>
<tr>
<td><strong>IYZ</strong></td>
<td></td>
</tr>
<tr>
<td><strong>TRATIO</strong></td>
<td>ratio of rotor #2 rotational speed t rotor #1 rotational speed, Ω₂/Ω₁ (transmission gear ratio r₁/r₂)</td>
</tr>
<tr>
<td><strong>CONFIG</strong></td>
<td>integer parameter specifying helicopter configuration</td>
</tr>
<tr>
<td>0</td>
<td>for one rotor</td>
</tr>
<tr>
<td>1</td>
<td>for single main rotor and tail rotor (rotor #2 is the tail rotor)</td>
</tr>
<tr>
<td>2</td>
<td>for tandem main rotors (rotor #2 is the rear rotor)</td>
</tr>
<tr>
<td>3</td>
<td>for tilting proprotor aircraft (rotor #2 is the left rotor)</td>
</tr>
<tr>
<td><strong>ASHAFT(2)</strong></td>
<td>shaft angle of attack Θᵣ (deg), positive rearward</td>
</tr>
<tr>
<td>1</td>
<td>rotor #1</td>
</tr>
<tr>
<td>2</td>
<td>rotor #2</td>
</tr>
<tr>
<td><strong>ACANT(2)</strong></td>
<td>shaft cant angle φᵣ (deg); positive to right for main rotor; positive upward for tail rotor; positive inward in helicopter mode for tilt rotor</td>
</tr>
<tr>
<td>1</td>
<td>rotor #1</td>
</tr>
<tr>
<td>2</td>
<td>rotor #2</td>
</tr>
<tr>
<td><strong>ATILT</strong></td>
<td>nacelle tilt angle θᵣ (deg), for tilting proprotor configuration only; 0° for airplane mode, 90° for helicopter mode</td>
</tr>
<tr>
<td><strong>HMAST</strong></td>
<td>rotor mast length from pivot to hub (ft or m), for tilting proprotor configuration only</td>
</tr>
<tr>
<td><strong>DPSI21</strong></td>
<td>ΔΨ₂₁ (deg); rotor #2 azimuth angle Ψ₂ when rotor #1 azimuth angle Ψ₁ = 0°; must be 0° if Ω₂/Ω₁ ≠ 1.</td>
</tr>
<tr>
<td><strong>CANTHT</strong></td>
<td>horizontal tail cant angle φₜ (deg), positive to left</td>
</tr>
<tr>
<td><strong>CANTVT</strong></td>
<td>vertical tail cant angle φᵥ (deg), positive to right</td>
</tr>
</tbody>
</table>
location (fuselage station, butt line, and waterline) of aircraft components relative to a body fixed reference system having an arbitrary orientation and origin; fuselage station (FS) positive aft, butt line (BL) positive to right, and waterline (WL) positive up (ft or m)

FSCG  aircraft center of gravity location
BLCG
WLCG
FSR1  rotor #1 hub location (right nacelle pivot location for tilting proprotor configuration)
BLR1
WLR1
FSR2  rotor #2 hub location
BLR2
WLR2
FSWB  wing-body center of action
BLWB
WLBW
FSHT  horizontal tail center of action
BLHT
WLHT
FSVT  vertical tail center of action
BLVT
WLVT
FSOFF  -int off rotor disk (for induced velocity calculation)
BLOFF
WLOFF

$CNTRL2(11)$  control inputs (deg) for all sticks centered ($\hat{\gamma}_p = 0$):

\[ \hat{\varphi}_0 = (\Theta_0 \Theta_c \Theta_{is} \Theta_0 \Theta_c \Theta_{is} \zeta_f \zeta_e \zeta_a \zeta_r \Theta_t)^T \]

rotor #1  rotor #2  aircraft
description of control system (for \( T_{\text{CFE}} \)); \( K \) parameters are gains (deg per stick deflection); \( \Delta \psi \) parameters are swashplate azimuth lead angles (deg)

one rotor, single main rotor and tail rotor, tilting proprotor configurations

KOCFE
- \( K_0 \), collective stick to collective pitch

KCCFE
- \( K_c \), lateral cyclic stick to cyclic or differential collective pitch

KSCFE
- \( K_s \), longitudinal cyclic stick to cyclic pitch

KPCFE
- \( K_p \), pedal to tail rotor collective or differential cyclic pitch

PCCFE
- \( \Delta \psi_c \), lateral cyclic stick to cyclic pitch (one rotor, or single main rotor and tail rotor configurations)

PSCFE
- \( \Delta \psi_s \), longitudinal cyclic stick to cyclic pitch

PPCFE
- \( \Delta \psi_p \), pedal to differential cyclic pitch (tilting proprotor configuration only)

tandem main rotor configuration

KFCFE
- \( K_{FO} \), collective stick to front collective pitch

KRCFE
- \( K_{RO} \), collective stick to rear collective pitch

KFCCFE
- \( K_{FC} \), lateral cyclic stick to front cyclic pitch

KRCFE
- \( K_{RC} \), lateral cyclic stick to rear cyclic pitch

KSCFE
- \( K_{FS} \), longitudinal cyclic stick to front collective pitch

KRSCE
- \( K_{RS} \), longitudinal cyclic stick to rear collective pitch

KFPCE
- \( K_{FP} \), pedal to front cyclic pitch

KRPCFE
- \( K_{RP} \), pedal to rear cyclic pitch

PFCCFE
- \( \Delta \psi_{FC} \), lateral cyclic stick to front cyclic pitch

PRCCE
- \( \Delta \psi_{RC} \), lateral cyclic stick to rear cyclic pitch

FPCFE
- \( \Delta \psi_{FP} \), pedal to front cyclic pitch

FRPCFE
- \( \Delta \psi_{RP} \), pedal to rear cyclic pitch

aircraft controls (all configurations)

KFCFE
- \( K_f \), collective stick to flaperon

KTOFE
- \( K_t \), collective stick to throttle

KACFE
- \( K_a \), lateral cyclic stick to ailerons

KECFE
- \( K_e \), longitudinal cyclic stick to elevator

KRCFE
- \( K_r \), pedal to rudder
NEM number of airframe modes for which data supplied; maximum 10

QMASS(NEM) generalized mass $M_k$ including rotors (slug or kg)

QFREQ(NEM) generalized frequency $\omega_k$ (Hz)

QDAMP(NEM) structural damping $\xi_k$

QDAMPA(NEM) aerodynamic damping $F_{qk}A_k = \delta\left(\frac{Q_k}{\frac{1}{2}v^2}\right)/\delta(\dot{\alpha}_{sk}/v)$ (ft$^2$ or m$^2$)

QCNTRL(4,NEM) control derivatives $F_{qk}A_k = \delta\left(\frac{Q_k}{\frac{1}{2}v^2}\right)/\delta \delta_\phi, \delta_{\omega}, \delta_{\epsilon_\phi}$ (ft$/\text{rad}$ or m$/\text{rad}$)

DOFSYMP(NEM) integer vector designating type of mode: GT 0 for symmetric, LT 0 for antisymmetric; only required for flutter analysis with OFSYM NE 0

ZETAR1(3,NEM) linear mode shape $\tilde{f}_k$ at rotor #1 hub (ft/ft or m/m)

ZETAR2(3,NEM) linear mode shape $\tilde{f}_k$ at rotor #2 hub (ft/ft or m/m)

GAMAR1(3,NEM) angular mode shape $\tilde{\psi}_k$ at rotor #1 hub (rad/ft or rad/m)

GAMAR2(3,NEM) angular mode shape $\tilde{\psi}_k$ at rotor #2 hub (rad/ft or rad/m)

KPCMC1(NEM) pitch/mast-bending coupling (rad/ft or rad/m)

\[ K_{MC_k} = -\frac{\partial \theta_{1c}}{\partial q_{sk}} \text{ for rotor #1} \]

KPCM2(NEM) \[ K_{MC_k} = -\frac{\partial \theta_{1c}}{\partial q_{sk}} \text{ for rotor #2} \]

KPCMS1(NEM) \[ K_{MS_k} = -\frac{\partial \theta_{1s}}{\partial q_{sk}} \text{ for rotor #1} \]

KPCMS2(NEM) \[ K_{MS_k} = -\frac{\partial \theta_{1s}}{\partial q_{sk}} \text{ for rotor #2} \]
## Aircraft aerodynamic characteristics

### Wing-body

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Expression</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>LFTA W</td>
<td>$L_{\alpha}/q$</td>
<td>(ft²/ rad or m²/ rad)</td>
</tr>
<tr>
<td>LFTFW</td>
<td>$L_{\delta_{f}}/q$</td>
<td>(ft²/ rad or m²/ rad)</td>
</tr>
<tr>
<td>LFTIW</td>
<td>$L_{\delta_{p}}/q$</td>
<td>(ft²/ rad or m²/ rad)</td>
</tr>
<tr>
<td>AMAXW</td>
<td>$\alpha_{\text{max}}$</td>
<td>(deg)</td>
</tr>
<tr>
<td>IWB</td>
<td>$\theta_{\text{WB}}$</td>
<td>(deg)</td>
</tr>
<tr>
<td>DRCGW</td>
<td>$f_{\text{WB}} = D_{0}/q$</td>
<td>(ft² or m²)</td>
</tr>
<tr>
<td>DRCGW</td>
<td>$f_{\text{vert}}$</td>
<td>(ft² or m²)</td>
</tr>
<tr>
<td>DRCGW</td>
<td>$\tau_{\text{e}<em>W} = \left( \Delta(D</em>{1}/q)/ \Delta(1/q) \right)^{-1}$</td>
<td>(ft² or m²)</td>
</tr>
<tr>
<td>DRCGW</td>
<td>$D_{0g}/q$</td>
<td>(ft²/ rad or m²/ rad)</td>
</tr>
<tr>
<td>DRCGW</td>
<td>$D_{0g_{r}}/q$</td>
<td>(ft²/ rad or m²/ rad)</td>
</tr>
<tr>
<td>MOMOW</td>
<td>$M_{0}/q$</td>
<td>(ft³ or m³)</td>
</tr>
<tr>
<td>MOMOW</td>
<td>$M_{\alpha}/q$</td>
<td>(ft³/ rad or m³/ rad)</td>
</tr>
<tr>
<td>MOMOW</td>
<td>$M_{\delta_{f}}/q$</td>
<td>(ft³/ rad or m³/ rad)</td>
</tr>
<tr>
<td>MOMOW</td>
<td>$M_{\delta_{p}}/q$</td>
<td>(ft³/ rad or m³/ rad)</td>
</tr>
<tr>
<td>SIDEB</td>
<td>$Y_{\epsilon}/q$</td>
<td>(ft²/ rad or m²/ rad)</td>
</tr>
<tr>
<td>SIDEW</td>
<td>$V_{\epsilon}/q$</td>
<td>(ft³/ rad or m³/ rad)</td>
</tr>
<tr>
<td>SIDEP</td>
<td>$V_{\epsilon}/q$</td>
<td>(ft³/ rad or m³/ rad)</td>
</tr>
<tr>
<td>ROLLB</td>
<td>$N_{x_{b}}/q$</td>
<td>(ft³/ rad or m³/ rad)</td>
</tr>
<tr>
<td>ROLLP</td>
<td>$V_{N_{x_{b}}}/q$</td>
<td>(ft⁴/ rad or m⁴/ rad)</td>
</tr>
<tr>
<td>ROLLR</td>
<td>$V_{N_{x_{b}}}/q$</td>
<td>(ft⁴/ rad or m⁴/ rad)</td>
</tr>
<tr>
<td>ROLLA</td>
<td>$N_{x_{w_{a}}}/q$</td>
<td>(ft³/ rad or m³/ rad)</td>
</tr>
<tr>
<td>YAWB</td>
<td>$N_{z_{w_{a}}}/q$</td>
<td>(ft³/ rad or m³/ rad)</td>
</tr>
<tr>
<td>YAWF</td>
<td>$V_{N_{z_{w_{a}}}}/q$</td>
<td>(ft⁴/ rad or m⁴/ rad)</td>
</tr>
<tr>
<td>YAWR</td>
<td>$V_{N_{z_{w_{a}}}}/q$</td>
<td>(ft⁴/ rad or m⁴/ rad)</td>
</tr>
<tr>
<td>YAWA</td>
<td>$N_{z_{w_{a}}}/q$</td>
<td>(ft³/ rad or m³/ rad)</td>
</tr>
</tbody>
</table>

### Horizontal tail

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Expression</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>LFTA W</td>
<td>$L_{\alpha}/q$</td>
<td>(ft²/ rad or m²/ rad)</td>
</tr>
<tr>
<td>LFTFW</td>
<td>$L_{\delta_{f}}/q$</td>
<td>(ft²/ rad or m²/ rad)</td>
</tr>
<tr>
<td>AMAXH</td>
<td>$\alpha_{\text{max}}$</td>
<td>(deg)</td>
</tr>
<tr>
<td>IHT</td>
<td>$\theta_{\text{HT}}$</td>
<td>(deg)</td>
</tr>
</tbody>
</table>
### Vertical Tail

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>LFTAV</td>
<td>$L_{\omega}/q$</td>
<td>$(ft^2/\text{rad} \text{ or } m^2/\text{rad})$</td>
</tr>
<tr>
<td>LFRV</td>
<td>$L_{\delta_x}/q$</td>
<td>$(ft^2/\text{rad} \text{ or } m^2/\text{rad})$</td>
</tr>
<tr>
<td>AMAXV</td>
<td>$\alpha_{\text{max}}$</td>
<td>$(\text{deg})$</td>
</tr>
<tr>
<td>IVT</td>
<td>$1_{VT}$</td>
<td>$(\text{deg})$</td>
</tr>
</tbody>
</table>

### Airframe interference

- **FETAIL**: $\ell_{\epsilon} = \left( \frac{\partial \epsilon}{\partial (L/q)} \right)^{-1}$ | $(ft^2 \text{ or } m^2)$ |
- **LHTAIL**: horizontal tail length $\ell_{HT}$ for $\epsilon$ | $(ft \text{ or } m)$ |
- **HVTAIL**: vertical tail height $h_{VT}$ for $\varphi$, positive up | $(ft \text{ or } m)$ |
- **OPTINT**: integer parameter controlling airframe/tail aerodynamic interference. EQ 0 to suppress $(\epsilon = 0 \text{ and } \varphi = 0)$
Engine and drive train parameters

**ENGPOS**
- integer parameter specifying drive train configuration:
  0: one rotor
  1: asymmetric, engine by rotor #1
  2: asymmetric, engine by rotor #2
  3: symmetric

**IENG**
- engine rotational inertia \( r^2_E \) for both engines if symmetric configuration (slug-ft² or kg-m²)
- drive train spring constants (ft-lb/rad or m-N/rad)

**KMAST1**
- rotor #1 shaft, \( K_{M1} \) or \( K_{M} \)

**KMAST2**
- rotor #2 shaft, \( K_{M2} \)

**KICS**
- interconnect shaft, \( r^2_{I2}K_I \) or \( r^2_IK_I \)

**KENG**
- engine shaft, \( r^2_EK_E \)

**GSE**
- engine shaft structural damping \( g_s \) (\( \Psi_e \) degree of freedom)

**GSI**
- interconnect shaft structural damping \( g_s \) (\( \Psi_i \) degree of freedom)

**KEDAMP**
- engine damping factor \( K_e \); typically 1.0 for turboshaft engines, or 10.0 for induction electric motors

**THRTL C**
- \( \Delta P_e/\Delta \Theta_t \) (dimensional), for both engines if symmetric configuration; if the throttle variable \( \Theta_t \) is only used for the governor, just the products
  \[
  K_p \frac{\Delta P_e}{\Delta \Theta_t} = -\frac{\Delta P}{\Delta \Psi_s} \\
  K_i \frac{\Delta P_e}{\Delta \Theta_t} = -\frac{\Delta P}{\Delta \Psi_s}
  \]
  must be correct (\( P = \Omega Q, \Psi_s = \Omega_EQ_E \))

**KPGOVE**
- governor proportional feedback gains (sec)
  - to throttle, \( K_p = -\frac{\partial \Theta_t}{\partial \Psi_s} \)
  - to rotor #1 collective, \( K_p = \frac{\partial \Theta}{\partial \Psi_s} \)
  - to rotor #2 collective, \( K_p = \frac{\partial \Theta}{\partial \Psi_s} \)

**KPGOV1**
- governor integral feedback gains
  - to throttle, \( K_i = -\frac{\partial \Theta_t}{\partial \Psi_s} \)
  - to rotor #1 collective, \( K_i = \frac{\partial \Theta}{\partial \Psi_s} \)
  - to rotor #2 collective, \( K_i = \frac{\partial \Theta}{\partial \Psi_s} \)
governor time lag $\tau_1 = 2 \gamma / \omega_n \text{ (sec)}$

$T_{GOV}$

throttle

$T_{GOV1}$

rotor #1

$T_{GOV2}$

rotor #2

governor time lag $\tau_2 = 1 / \omega_n^2 \text{ (sec}^2)\text{)}$

$T_{GOV}$

throttle

$T_{GOV1}$

rotor #1

$T_{GOV2}$

rotor #2
**Namelist NLLOAD**

Airframe vibration

**MVIB**
number of stations for airframe vibration calculation and print; maximum 10; LE 0 to suppress

**FSVIB(MVIB)**
airframe location for vibration calculation (ft or m)

**BLVIB(MVIB)**
butt line

**WLVIB(MVIB)**
waterline

**ZETAV(3, NEM, MVIB)**
linear mode shape at airframe vibration stations (ft/ft or m/m)

**MALOAD**
integer parameter controlling print of motion and aerodynamics: EQ 0 to suppress; LT 0 for only plots

**MHLOAD**
integer parameter controlling print of hub and control loads: EQ 0 to suppress

**MRLOAD**
number of radial stations for blade section load calculation and print; maximum 20; LE 0 to suppress

**RLOAD(MRLOAD)**
blade radial stations r/R for section loads

**MHARML**
number of harmonics in loads analysis; maximum 30; LT 0 for no harmonic analysis; suggest about MPSI/3

**NPOLAR**
integer parameter n for polar plots: symbol printed every n-th step

**MWKGM(4)**
integer parameter controlling wake geometry printer plot; EQ 0 to suppress

1. top view
2. side view
3. back view
4. axial convection

**MWKGM**
number of azimuth stations at which wake geometry plotted; maximum 8; LE 0 for no plots

**JWKGM(MWKGM)**
azimuth stations at which wake geometry plotted (ψ = jΔψ)
**NLOAD**

**NPUIT(75)**  
integer parameter controlling printer-plots of motion and aerodynamics: 0 for no plot, 1 for time history plot, 2 for polar plot, 3 for both (only time history available for 1-4 and 68-75)

1. bending motion
2. torsion motion
3. maximum circulation
4. \( \Lambda \) off rotor
5. \( \alpha \)
6. \( M \)
7. \( \Lambda \)
8. \( c_R \)
9. \( c_D \)
10. \( c_m \)
11. \( c_{d_{\text{radial}}} \)
12. \( \gamma \)
13. \( u_p \)
14. \( u_T \)
15. \( u_R \)
16. \( U \)
17. \( \Theta \)
18. \( \phi \)
19. \( \phi_{\text{lag}} \)
20. \( \phi_{\text{flap}} \)
21. \( \alpha_{\text{eff}} \), lift
22. \( \alpha_{\text{eff}} \), drag
23. \( M_{\text{eff}} \), moment
24. \( M_{\text{eff}} \), lift
25. \( M_{\text{eff}} \), drag
26. \( M_{\text{eff}} \), moment
27. \( \lambda_{\text{x}} \)
28. \( \lambda_{\text{y}} \)
29. \( \lambda_{\text{z}} \)
30. interference \( \lambda_{\text{x}} \)
31. \( \lambda_{\text{y}} \)
32. \( \lambda_{\text{z}} \)
33. \( u_C \)
34. \( v_C \)
35. \( w_C \)
36. \( \mathcal{L}/c \)
37. \( D/c \)
38. \( M/c \)
39. \( D_p/c \)
40. \( F_{x}/c \)
41. \( F_{y}/c \)
42. \( F_{z}/c = C_T/\sqrt{c} \)
43. \( M_{a}/c \)
44. \( F_{r}/c \)
not used
not used
not used
\( C_p \)
\( C_p \)
\( C_p \)
\( C_{p, \text{int}} \)
\( L \)
\( D \)
\( M \)
\( D_p \)
\( F_x \)
\( F_T \)
\( F_z = T \)
\( M^a \)
\( F_T \)
not used
not used
not used
\( P \)
\( P_1 \)
\( F_{\text{int}} \)
\( F_0 \)
rotating frame root loads
nonrotating frame hub loads
rotating frame root loads
nonrotating frame hub loads
section loads, shaft axes
section loads, principal axes
section loads, shaft axes
section loads, principal axes

*dimensional quantities

For polar plots, last digit of integer part of (value/increment) is printed, if it is a multiple of NPLAR; the plot increment is defined as follows:

- .01 plots 27-35
- .1 plots 6, 8-16, 24-26, 36-51
- 1. plots 5, 7, 17-23, 52-61
- 10. plots 62-69
parameter $K$ in fatigue damage calculation; suggest 3 or 4

endurance limit $S_E$ (dimensional force or moment)

material constant $C$

material exponent $M$

rotating frame root loads

1. inplane shear $f_x$
2. axial shear $f_T$
3. vertical shear $f_Z$
4. flap moment $m_x$
5. lag moment $m_y$
6. control moment $m_z$

nonrotating frame hub loads

7. drag force $H$
8. side force $Y$
9. thrust $T$
10. roll moment $m_x$
11. pitch moment $m_y$
12. torque $Q$

section loads (principal axes)

13. chord shear $f_x$
14. axial shear $f_T$
15. normal shear $f_Z$
16. flatwise moment $m_x$
17. edgewise moment $m_y$
18. torsion moment $m_t$

the S-N curve is approximated by $N = C / (S/S_E - 1)^M$

use $S_E$ LT 0. or $C$ LT 0. to suppress damage fraction calculation; use $M$ EQ 0. to suppress equivalent peak-to-peak load calculation as well.
Far field rotational noise

**MNOISE**
- number of microphones: maximum 10; LE 0 for no noise analysis

**RANGE(MNOISE)**
- microphone range relative hub (ft or m)

**ELVATN(MNOISE)**
- microphone elevation relative hub (deg), positive above rotor disk

**AZMUTH(MNOISE)**
- microphone azimuth relative hub (deg), defined as for rotor azimuth

**MHARMN(3)**
- number of harmonics
  1. in noise calculation; maximum 500
  2. in aerodynamic load harmonic analysis (suggest MPSI/3)
  3. in print of noise (LE 0 for no print)

**MTIMEN(3)**
- number of time steps (LE 0 to suppress)
  1. in period of noise calculation; maximum 500
  2. increment in noise print
  3. increment in noise plot

**AXS(MRA)**
- blade cross section area \( A_{x,y}/c^2 \) at aerodynamic segments, for thickness noise calculation (typically 0.685 times thickness ratio)

**OPNOIS(4)**
- integer parameter controlling noise calculation:
  0 to suppress, 1 for impulsive chordwise loading,
  2 for distributed chordwise loading
  1. lift noise
  2. drag noise
  3. radial force noise
  4. thickness noise
**Namelist NLFLUT**

**OPFLOW**
integer parameter specifying analysis type: LT 0 for constant coefficient approximation; EQ 0 for axial flow; GT 0 for periodic coefficients

**OPSYM**
integer parameter: NE 0 for symmetric and antisymmetric analyses (tilting proprotor configuration only)

**OPFDAN**
integer parameter: EQ 0 to suppress flight dynamics analysis

**NBLJFL**
integer parameter: EQ 1 for independent rotor blade analysis

**MPSIFC**
number of azimuth steps in period for nonaxial flow, periodic coefficient analysis (OPFLOW GT 0); \( \Delta \psi = 360/(N_{bld} M) \) for odd number of blades, \( \Delta \psi = 720/(N_{bld} M) \) for even number of blades

**NINTPC**
integer parameter specifying numerical integration option for periodic coefficient analysis (OPFLOW GT 0): 1 for modified trapezoidal method, 2 for Runge-Kutta method

**MPSICC**
number of azimuth stations (per revolution) in evaluation of average coefficients for constant coefficient approximation (OPFLOW LT 0); \( \Delta \psi = 360^\circ/\psi \)

**DALPHA**
angle of attack increment \( \Delta \alpha \) (deg) for calculation of \( c_x, c_d, \) and \( c_m \) derivatives in aerodynamic coefficients

**DMACH**
Mach number increment \( \Delta M/M \) for calculation of \( c_x, c_d, \) and \( c_m \) derivatives in aerodynamic coefficients

**OPUSLD**
integer parameter controlling use of unsteady lift and moment in flutter analysis: 0 to suppress; 1 to include; 2 for zero in stall \( (15^\circ < |\omega| < 165^\circ) \)

**DELTA**
control and motion increment for aircraft stability derivative calculation (dimensionless)

**CPRINT**
integer parameter: EQ 0 to suppress rotor/body aerodynamic interference in flutter analysis

**OPGRND**
integer parameter controlling ground effect analysis: EQ 0 for out of ground effect, NE 0 for in ground effect

**KASGE**
factor for antisymmetric ground effect: 0. to suppress, 1.0 for unstable roll moment due to ground effect (tilting proprotor configuration only)

**OPSAS**
integer parameter controlling use of SAS: EQ 0 to suppress

**KCSAS**
lateral SAS gain \( K_c = -\Delta \delta_c / \Delta \phi_p \) (deg/deg)

**KSSAS**
longitudinal SAS gain \( K_s = -\Delta \delta_s / \Delta \Theta_f \) (deg/deg)

**TCSAS**
lateral SAS lead time \( T_c \) (sec)

**TSSAS**
longitudinal SAS lead time \( T_s \) (sec)
OPTORS(2)  integer parameter: EQ 0 for rigid pitch model (infinite control system stiffness, no \( p_0 \) degree of freedom)

(1)  rotor #1
(2)  rotor #2

DOF(80)  integer vector defining degrees of freedom for flutter analysis; 0 if not used, 1 if used, 2 if quasistatic variable; order:

- Rotor #1:
  - \( \beta_0, \beta_{1c}, \beta_{1s}, \ldots, \beta_{N/2} \)
  - \( \phi_0, \phi_{1c}, \phi_{1s}, \ldots, \phi_{N/2} \)
  - \( \phi_{g0}, \phi_{g1c}, \phi_{g1s}, \ldots, \phi_{gN/2} \)
- Rotor #2:
  - \( \beta_0, \beta_{1c}, \beta_{1s}, \ldots, \beta_{N/2} \)
  - \( \phi_0, \phi_{1c}, \phi_{1s}, \ldots, \phi_{N/2} \)
  - \( \phi_{g0}, \phi_{g1c}, \phi_{g1s}, \ldots, \phi_{gN/2} \)

- Bending
- Pitch/torsion
- Gimbal rotor
- Inflow
- Teeter speed

airframe:
- \( \phi_{t0}, \phi_{t1c}, \phi_{t1s}, \ldots, \phi_{tN/2} \)
- \( \phi_{c0}, \phi_{c1c}, \phi_{c1s}, \phi_{c2} \)
- \( \phi_{w0}, \phi_{w1c}, \phi_{w1s}, \phi_{w2} \)
- \( \phi_{d0}, \phi_{d1c}, \phi_{d1s}, \phi_{d2} \)
- \( \phi_{e0}, \phi_{e1c}, \phi_{e1s}, \phi_{e2} \)
- \( \phi_{t0}, \phi_{t1c}, \phi_{t1s}, \phi_{t2} \)

rigid body: flexible engine governor body

speed

CON(26)  integer vector defining control variables, 0 if not used; order:

- Rotor #1:
  - \( \theta_0, \theta_{1c}, \theta_{1s}, \ldots, \theta_{N/2} \)
- Rotor #2:
  - \( \theta_0, \theta_{1c}, \theta_{1s}, \ldots, \theta_{N/2} \)
  - Pitch (8)

- Airframe:
  - \( \delta_y, \delta_c, \delta_s, \delta_r, \delta_t \)
- Pilot:
  - \( \delta_0, \delta_c, \delta_s, \delta_r, \delta_t \)

GUS(3)  integer vector defining gust components, 0 if not used; order: \( u_G, v_G, w_G \)

for a two-bladed rotor, \( \beta_{GC} \) is replaced by \( \beta_T \)

there are \( N \) rotor pitch control variables; except for a two-bladed rotor, which has the 4 variables \( \theta_0, \theta_{1c}, \theta_{1s}, \theta_1 \)
ANTYPE(4) integer parameter specifying tasks in linear system analysis, SQ 0 to suppress
   (1) eigenanalysis
   (2) transfer function printer-plot
   (3) time history printer-plot
   (4) rms gust response

Eigenanalysis

NSYSAN calculation control: 0 for eigenvalues, 1 for eigenvalues and eigenvectors; 10 or 11 for zeros as well

NSTEP static response calculated if NE 0

NFREQ number of frequencies for which frequency response calculated; LE 0 to suppress; maximum 100

FREQ(NFREQ) vector of frequencies (per rev)

Transfer function printer-plot

NBPLT calculation method: if 1, from matrices; if 2, from poles and zeros

NXPLT number of degrees of freedom to be plotted; maximum 80

NVPLT number of controls to be plotted; maximum 29

NAMEXP(NXPLT) vector of variable names to be plotted (inconsistent names ignored)

NAMEVP(NVPLT) vector of control names to be plotted (inconsistent names ignored)

NDPLT frequency steps per decade

NFOPLT exponent (base 10) of beginning frequency

NF1PLT exponent (base 10) of end frequency

(maximum NF = (NF1PLT - NFOPLT) * NDPLT + 1 = 151)

MSPLT magnitude plot scale: if 1, plot relative maximum value; if 2, plot relative 10**K; if 3, plot relative 10.
NLFLPT

Time history printer-plot

NTPLT  control input type: 1 for step, 2 for impulse, 3 for cosine impulse, 4 for sine doublet, 5 for square impulse, 6 for square doublet

PERPLT period T for impulse or doublet (sec)

DTPLT time step (sec)

TMXPLT maximum time (sec); maximum NXPLT*NVPLT*TMXPLT/DTPLT = 7200

NXPLT number of degrees of freedom to be plotted; maximum 80

NVPLT number of controls to be plotted; maximum 29

NAMEXP(NXPLT)  vector of variable names to be plotted (inconsistent names ignored)

NAMEVP(NVPLT)  vector of control names to be plotted (inconsistent names ignored)

Rms gust response

LGUST(MG)  real vector of gust correlation lengths: $CT \sigma_g$, dimensional length $L (\sigma_g = L/2\nu)$; $EQ 0.$, set $L = 400.$; $LT 0.$, magnitude is dimensionless correlation time $\tau_G$ (frequency $\nu = \omega/2\pi$)

MGUST(MG)  real vector of gust component relative magnitudes

NAMEXA(MACC)  vector of names of degrees of freedom for which acceleration calculated; last 3 equal ACCB for body axis acceleration (all 3 or none) (inconsistent names ignored)

FREQA(MACC)  vector of acceleration break frequencies (Hz); 2/rev used if $LT 0.$; in same order as NAMEXA

MACC number of accelerometers; $LE 0.$ for none; maximum 83

location of point at which body axis acceleration calculated (ft or m)

FSACC fuselage station

BLACC butt line

WLACC waterline

ZETACC(3,NEM)  linear mode shape $\hat{y}_k$ at point where body axis acceleration calculated

NAMEXR(3)  names of $\beta_{ie}, \xi_{ie},$ and $\Theta_{ie}$ in state vector; assumed that $\beta_{ie}, \xi_{ie},$ and $\Theta_{ie}$ follow immediately (inconsistent names ignored)
Variable names for linear system analysis

<table>
<thead>
<tr>
<th>Degrees of freedom</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1B1 $\beta_{11}^{(1)} \beta_{12}^{(1)} ... \beta_{1n}^{(1)}$</td>
<td>bending rotor #1</td>
</tr>
<tr>
<td>1B15</td>
<td></td>
</tr>
<tr>
<td>1T1 $\theta_{11}^{(1)} \theta_{12}^{(1)} ... \theta_{1n}^{(1)}$</td>
<td>pitch/torsion</td>
</tr>
<tr>
<td>1T9</td>
<td></td>
</tr>
<tr>
<td>1BGCC $\beta_{GC}$</td>
<td>gimbal/teeter</td>
</tr>
<tr>
<td>1BGCS $\beta_{GS}$</td>
<td></td>
</tr>
<tr>
<td>PSIS $\psi_s$</td>
<td>rotor speed</td>
</tr>
<tr>
<td>1LU $\lambda_u$</td>
<td>inflow</td>
</tr>
<tr>
<td>1LX $\lambda_x$</td>
<td></td>
</tr>
<tr>
<td>1LY $\lambda_y$</td>
<td></td>
</tr>
<tr>
<td>2B1 $\beta_{21}^{(1)} \beta_{22}^{(1)} ... \beta_{2n}^{(1)}$</td>
<td>bending rotor #2</td>
</tr>
<tr>
<td>2B15</td>
<td></td>
</tr>
<tr>
<td>2T1 $\theta_{21}^{(1)} \theta_{22}^{(1)} ... \theta_{2n}^{(1)}$</td>
<td>pitch/torsion</td>
</tr>
<tr>
<td>2T9</td>
<td></td>
</tr>
<tr>
<td>2BGCC $\beta_{GC}$</td>
<td>gimbal/teeter</td>
</tr>
<tr>
<td>2BGCS $\beta_{GS}$</td>
<td></td>
</tr>
<tr>
<td>PSII $\psi_x$</td>
<td>rotor speed</td>
</tr>
<tr>
<td>2LU $\lambda_u$</td>
<td>inflow</td>
</tr>
<tr>
<td>2LX $\lambda_x$</td>
<td></td>
</tr>
<tr>
<td>2LY $\lambda_y$</td>
<td></td>
</tr>
<tr>
<td>PHIF $\phi_F$</td>
<td>rigid body</td>
</tr>
<tr>
<td>THTF $\theta_F$</td>
<td>airframe</td>
</tr>
<tr>
<td>PSIF $\psi_F$</td>
<td></td>
</tr>
<tr>
<td>XF $\xi_F$</td>
<td></td>
</tr>
<tr>
<td>YF $\eta_F$</td>
<td></td>
</tr>
<tr>
<td>ZF $\zeta_F$</td>
<td></td>
</tr>
</tbody>
</table>
Control variables

\[ q_{nk} \quad (k \geq 7) \]

\[ \psi_e \]

\[ \Delta \theta_g \]

\[ \Delta \theta_{gov} \]

\[ \Delta \theta_{gov2} \]

Flexible body

Airframe

Engine speed

Governor

Rotor #1

Rotor #2

Airframe

Pilot
Gust components

UG \ u_G
VG \ v_G
WG \ w_G

For the rotor names, the leading character (1 or 2) is replaced as follows, depending on the helicopter configuration

<table>
<thead>
<tr>
<th>CONFIG</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>blank (left justified)</td>
</tr>
<tr>
<td>1</td>
<td>M or T</td>
</tr>
<tr>
<td>2</td>
<td>F or R</td>
</tr>
<tr>
<td>3</td>
<td>R or L (OPSYM = 0)</td>
</tr>
<tr>
<td>3</td>
<td>S or A (OPSYM \neq 0)</td>
</tr>
</tbody>
</table>

For a two bladed rotor, BGC is replaced by BT

For first order degrees of freedom, the only state is the velocity, hence it is the velocity that will be plotted
**Namelist NLSTAB**

- **NFRNTP**: integer parameter controlling performance print during stability derivative calculation; LE 0 to suppress
- **NFRNTL**: integer parameter controlling loads print during stability derivative calculation; LE 0 to suppress
- **ITERS**: number of wake influence coefficient/motion and forces iterations
- **OFLMDA**: integer parameter controlling induced velocity calculation:
  - If 0, update influence coefficients and inflow; if 1, suppress influence coefficient update; if 2, suppress inflow update (and influence coefficient update)
- **DELTA**: control and motion increment for stability derivative calculation (dimensionless)
- **DF(?)**: integer vector defining degrees of freedom, 0 if not used; order: \( \phi_F, \Theta_F, \psi_F, x_F, y_F, z_F, \psi_S \)
- **CON(16)**: integer vector defining control variables, 0 if not used; order:
  -Rotor \#1: \( \Theta_0, \Theta_1, \Theta_5, \Theta_6 \)
  -Rotor \#2: \( \Theta_0, \Theta_1, \Theta_5, \Theta_6 \)
  -Airframe: \( \delta_b, \delta_c, \delta_a, \delta_r, \Theta_c \)
  -Pilot: \( \delta_b, \delta_c, \delta_a, \delta_r, \Theta_c \)
- **GUS(3)**: integer vector defining gust components, 0 if not used; order: \( \nu_G, \nu_G', \nu_G'' \)
- **CPPRNT(+)**: integer parameters controlling stability derivative print, EQ 0 to suppress:
  1. Rotor coefficient form, dimensionless
  2. Rotor coefficient form, dimensional
  3. Stability derivative form, dimensionless
  4. Stability derivative form, dimensional
- **KCSAS**: lateral SAS gain, \( K_c = -\Delta \psi_c / \Delta \phi_p \) (deg/deg)
- **KSSAS**: longitudinal SAS gain, \( K_s = -\Delta x_s / \Delta \Theta_p \) (deg/deg)
- **TCSAS**: lateral SAS lead time \( \tau_c \) (sec)
- **TSSAS**: longitudinal SAS lead time \( \tau_s \) (sec)
EQTYPE(12) integer parameter specifying equations to be analyzed, EQ 0 to suppress
with \( \psi \), with SAS
(1) complete
(2) symmetric
(3) antisymmetric
with \( \psi \), without SAS
(4) complete
(5) symmetric
(6) antisymmetric
without \( \psi \), with SAS
(7) complete
(8) symmetric
(9) antisymmetric
without \( \psi \), without SAS
(10) complete
(11) symmetric
(12) antisymmetric

ANTYPE(5) integer parameter specifying tasks in linear system analysis, EQ 0 to suppress
(1) eigenanalysis
(2) transfer function printer-plot
(3) time history printer-plot
(4) rms gust response
(5) numerical integration of transient

Eigenanalysis

NSYSAN calculation control: 0 for eigenvalues, 1 for eigenvalues and eigenvectors; 10 or 11 for zeros as well

NSTEP static response calculated if NE 0

NFREQ number of frequencies for which frequency response calculated; LE 0 to suppress; maximum 100

FREQ(NFREQ) vector of frequencies (per rev)
Transfer function printer-plot

**NBPLT** calculation method: if 1, from matrices; if 2, from poles and zeros

**NAMEXP(NXPLT)** vector of variable names to be plotted (inconsistent names ignored)

**NAMEVP(NVPLT)** vector of control names to be plotted (inconsistent names ignored)

**NXPLT** number of degrees of freedom to be plotted; maximum 7

**NVPLT** number of controls to be plotted; maximum 19

**NDPLT** frequency steps per decade

**NFOPLT** exponent (base 10) of beginning frequency

**NFIPLT** exponent (base 10) of end frequency

\[ \text{(maximum } NF = (NFIPLT - NFOPLT) \times NDPLT + 1 = 151) \]

**MSPLT** magnitude plot scale: if 1, plot relative maximum value; if 2, plot relative \(10^{M}K\); if 3, plot relative 10.

Time history printer-plot

**NTPLT** control input type: 1 for step, 2 for impulse, 3 for cosine impulse, 4 for sine doublet, 5 for square impulse, 6 for square doublet

**PERPLT** period \(T\) for impulse or doublet (sec)

**DTPLT** time step (sec)

**TMXPLT** maximum time (sec); maximum \(NXPLT \times NVPLT \times TMXPLT / DTPLT = 7200\)

**NXPLT** number of degrees of freedom to be plotted; maximum 7

**NVPLT** number of controls to be plotted; maximum 19

**NAMEXP(NXPLT)** vector of variable names to be plotted (inconsistent names ignored)

**NAMEVP(NVPLT)** vector of control names to be plotted (inconsistent names ignored)

Rms gust response

**LGUST(MG)** real vector of gust correlation lengths: \(GT 0.\), dimensional length \(L (\tau_c = L/2V)\); \(EQ 0.\), set \(L = 400.\); \(LT 0.\), magnitude is dimensionless correlation time \(\tau_c\) (frequency \(\omega = \Omega / \tau_c\))

**MGUST(MG)** real vector of gust component relative magnitudes

\[ MG = \text{number of gust components, maximum } 3 \]
**NLSTAB**

- **NAMEXA(MACC)**: vector of names of degrees of freedom for which acceleration calculated; last 3 equal ACCB for body axis acceleration (all 3 or none) (inconsistent names ignored)
- **FREQA(MACC)**: vector of acceleration break frequencies (Hz); 2/rev used if LT 0.; same order as NAMEXA
- **MACC**: number of accelerometers; LE 0 for none; maximum 10
- **FSACC**: fuselage station
- **BLACC**: butt line
- **WLACC**: waterline

Numerical integration of transient
- **TSTEP**: time step in numerical integration (sec)
- **TMAX**: maximum time in numerical integration (sec)
- **NFRNTT**: integer parameter n; transient print every n-th integration step; LE 0 to suppress
- **OPPLOT**: integer parameter controlling printer plot of body motion; EQ 0 to suppress
- **DOFPLT(21)**: integer vector designating variables to be plotted, EQ 0 if not plotted; order:
  - $\phi, \theta, \psi, x, y, z, \dot{\phi}, \dot{\theta}, \dot{\psi}, \ddot{x}, \ddot{y}, \ddot{z}, \dddot{x}, \dddot{y}, \dddot{z}$
- **OPTRAN**: see namelist NLTRAN
- **CTIME**
- **CMAG(5)**
- **GTIME**
- **GMAG(3)**
- **GDIST(2)**
- **VELG**
- **FSIG**
- **OPGUST(3)**
Variable names for linear system analysis

Degrees of freedom

PHIF $\phi_F$ rigid body
THTF $\theta_F$
PSIF $\nu_F$
XF $x_F$
YF $y_F$
ZF $z_F$
PSIS $\psi_S$ rotor speed

Control variables

1C0 $\phi_0$ rotor #1
1C1C $\phi_{1c}$
1C1S $\phi_{1s}$
2C0 $\phi_0$ rotor #2
2C1C $\phi_{1c}$
2C1S $\phi_{1s}$
DELF $\delta_F$ aircraft
DELE $\delta_E$
DELA $\delta_A$
DELR $\delta_R$
CT $\theta_T$
DELO $\delta_0$ pilot
DELC $\delta_C$
DELS $\delta_S$
DELP $\delta_P$
DELT $\delta_T$

Gust components

UG $u_G$
VG $v_G$
WG $w_G$
For the rotor control names, the leading character (1 or 2) is replaced as follows, depending on the helicopter configuration:

<table>
<thead>
<tr>
<th>CONFIG</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>blank (left justified)</td>
</tr>
<tr>
<td>1</td>
<td>M or T</td>
</tr>
<tr>
<td>2</td>
<td>F or R</td>
</tr>
<tr>
<td>3</td>
<td>R or L</td>
</tr>
</tbody>
</table>

For first order degrees of freedom the only state is the velocity; hence it is the velocity that will be plotted.
Namelist NLTRAN

NPRNTT integer parameter n: transient/performance/loads print every n-th integration step; LE 0 to suppress

NPRNTP integer parameter controlling performance print; LE 0 to suppress

NPRNTL integer parameter controlling loads print: LE 0 to suppress

NRSTRT integer parameter n: restart file written only every n-th integration step; LE 0 to suppress

TSTEP time step in numerical integration (sec)

TMAX maximum time in numerical integration (sec)

ITERT number of wake influence coefficients/motion and forces iterations

CPLMDE integer parameter controlling induced velocity calculation:
   if 0, update influence coefficients and inflow; if 1, suppress influence coefficient update; if 2, suppress inflow update (and influence coefficient update)

DOP(7) integer vector defining degrees of freedom in numerical integration; E= 0 to suppress acceleration;
   order: \( \Phi_F, \Theta_F, \Psi_F, x_F, y_F, z_F, \Psi_S \)

CFSAS integer parameter controlling use of SAS; E= 0 to suppress

KCSAS lateral SAS gain, \( K_C = -\Delta \Psi_C / \Delta \Phi_F \) (deg/deg)

KSSAS longitudinal SAS gain, \( K_S = \Delta \Psi_S / \Delta \Theta_F \) (deg/deg)

TCSAS lateral SAS lead time \( \tau_C \) (sec)

TSSAS longitudinal SAS lead time \( \tau_S \) (sec)

OPPLOT integer parameter controlling printer plot of body motion; E= 0 to suppress

DOPFLT(21) integer vector designating variables to be plotted; E= 0 for not plotted; order:

\( \Phi_F, \Theta_F, x_F, y_F, z_F, \Phi_F, \Theta_F, x_F, y_F, z_F, \Phi_F, \Theta_F, x_F, y_F, z_F, \Phi_F, \Theta_F, x_F, y_F, z_F \)
Transient gust and control

**OPTRAN**  integer parameter specifying transient option; 1 for control; 2 for uniform gust; 3 for convected gust

**CTIME**  period T for control (sec)

**CMAG(5)**  control magnitude \( \mathbf{v}_{\text{PC}} = (\delta_0, \delta_c, \delta_p, \delta_v)^T \) (deg)

defines cosine control transient with period T and magnitude \( \mathbf{v}_{\text{PC}} \)

**CTIME**  period T for uniform gust (sec)

**CMAG(3)**  gust magnitude \( \mathbf{g}_o = (u_G, v_G, w_G)^T \) (ft/sec or m/sec)

defines cosine uniform gust transient with period T and magnitude \( \mathbf{g}_o \)

**GDIST(2)**  lengths for convected gust (ft or m)

1. wavelength L
2. starting position \( L_o \)

**VELG**  gust convection velocity \( v_G \) (ft/sec or m/sec)

**PSIG**  azimuth angle of convected gust wave front \( \psi_g \) (deg)

**OPGUST(3)**  integer parameters defining convected gust model

1. EQ 0 to not use \( v_a \)
2. rotor #1: 0 for gust at hub, 1 for over disk
3. rotor #2: 0 for gust at hub, 1 for over disk

defines cosine convected gust transient with wavelength L and magnitude \( \mathbf{g}_o \); for \( L_o = R \) the wave starts at edge of rotor disk, for \( L_o = 0 \), the wave starts at hub -- assuming the aircraft center of gravity is directly below the hub; convected at rate \( v_a \) relative to moving aircraft if \( v_a \) is not used, \( \mathbf{g} \) at rate \( v_a \) relative to fixed frame if \( v_a \) is used
Transient gust and control subroutines

The subroutine CONTRL calculates the transient control time history, \( C(t) \). The subroutine GUSTU calculates the uniform gust time history, \( G(t) \). The subroutine GUSTC calculates the convected gust wave shape, \( G(x) \). The subroutines presently calculate a cosine-impulse gust:

\[
\begin{align*}
\text{CONTRL} & \quad C(t) = \frac{1}{2}(1 - \cos \frac{2\pi t}{T}) \\
\text{GUSTU} & \quad G(t) = \frac{1}{2}(1 - \cos \frac{2\pi t}{T}) \\
\text{GUSTC} & \quad G(x_g) = \frac{1}{2}(1 - \cos \frac{2\pi(x_g - L_o)}{L})
\end{align*}
\]

Other transients may be used by replacing these subroutines as required.
Namelist Inputs for Old Job (Restart)

Namelist NLTRIM

ANTYPE(3)
OPREAD(10)
DEBUG(25)
NPRNTI

Namelist NLFLUT

ANTYPE(4)
NSYSAN
:
NAMEXR(3)

Namelist NLSTAB

CPFRNT(4)
KCSAS
KSSAS
TCSAS
TSSAS
EQTYPE(12)
ANTYPE(5)
NSYSAN
:
PIGUST(3)

Namelist NLTRAN

NFRNTT
NFRNTP
NFRNTL
NRSTRT
TMAX
7. NOTES ON PRINTED OUTPUT

This section presents notes on the printed output of the program, particularly regarding the units of the variables appearing in the output.

Print of Performance (Program PERF)

Operating condition:

a) motion: 1st number dimensionless, 2nd number dimensional
   1) velocity = ft/sec or m/sec
   2) dynamic pressure, q = lb/ft$^2$ or N/m$^2$
   3) weight, $C_w/\pi = lb$ or N
   4) body motion = deg/sec, ft/sec or m/sec
   5) $\dot{z} = ft/sec^2$ or m/sec$^2$
   6) $\dot{\psi}_S = rpm$

b) body orientation and controls in deg

Circulation convergence:

a) tolerance, CG/S in $C_w/\pi$ form
b) C/E = ratio error to tolerance ($\leq 1.0$, if converged)

Motion convergence:

a) tolerance, BETA (etc) in deg
b) BETA/E (etc) = ratio error to tolerance ($\leq 1.0$, if converged)

Airframe performance: section 4.2.6

a) aerodynamic loads: dimensional
b) components
   1) angles in deg
   2) loads, q dimensional
   3) induced velocity, total velocity dimensionless

Gust velocity: dimensionless

System power:

a) dimensional (HP); number in parentheses is percent total power
b) climb power = $V_c W$
System efficiency parameters:

a) gross weight, $W = 1$ or $N$

b) drag-rotor $= \frac{P_1 + P_0}{\nu}$; $D/\nu$-rotor $= D/\nu \sqrt{\frac{P_0}{\rho}}$;
$L/D$-rotor $= \frac{W}{D}$

c) drag-total $= D_{\text{total}}/\nu$; $D/\nu$-total $= D_{\text{total}}/\nu \sqrt{\frac{P_0}{\rho}}$;
$L/D$-total $= \frac{W}{D_{\text{total}}}$

d) figure of merit $= M = 1 - \frac{P_{\text{non-ideal}}}{P_{\text{total}}}$

Print of Rotor Loads (Program LCADR1)

Print aerodynamics (function $r$ and $\psi$)

a) dimensionless quantities generally, angles in degrees

b) induced velocity in nonrotating shaft axes ($\lambda_x$, $\lambda_y$, $\lambda_z$)

c) interference induced velocity is that due to other rotor

d) gust components in city axes

Force/c mean (dimensionless):

$L/C = \frac{1}{2}u^2 (c/c_{\text{mean}}) \rho = L/c_{\text{mean}}$

$D/C = \frac{1}{2}u^2 (c/c_{\text{mean}}) \sigma = D/c_{\text{mean}}$

$M/C = \frac{1}{2}u^2 (c^2/c_{\text{mean}}) \chi = M/c_{\text{mean}}$

$\Delta R/C = \frac{1}{2}u^2 (c/c_{\text{mean}}) \sigma_{\text{radial}} = D_{\text{radial}}/c_{\text{mean}}$

$F_2/C = CT/S = F_2/c_{\text{mean}} = d(c_T/\sigma)/dr$

$FX/C = F_x/c_{\text{mean}}$

$MA/C = M_a/c_{\text{mean}}$

$FR/C = F_r/c_{\text{mean}}$

$FMT/C = F_{\text{total}}/c_{\text{mean}}$
Forces (dimensional)

\[ L = \text{section lift} \quad (\text{lb/ft or N/m}) \]
\[ D = \text{section drag} \quad (\text{lb/ft or N/m}) \]
\[ M = \text{section pitch moment} \quad (\text{ft-lb/ft or m-N/m}) \]
\[ DR = \text{section radial drag} \quad (\text{lb/ft or N/m}) \]
\[ F_Z = \frac{dF}{dr} \quad (\text{lb/ft or N/m}) \]
\[ F_X = F_x \quad (\text{lb, ft or N/m}) \]
\[ MA = M_a \quad (\text{ft-lb/ft or m-N/m}) \]
\[ FR = F_r \quad (\text{lb/ft or N/m}) \]
\[ FRT = F_{rt} \quad (\text{ft-lb/ft or m-N/m}) \]

Blade section power: section 5.2.1

\[ CF/S = \frac{d(C_{p} \omega)}{dr} \]
\[ P = \text{section power (kW/ft or kW/m)} \]

Print During Stability Derivative Calculation (Program STABM)

a) increment: 1st number dimensionless, 2nd number dimensional

b) motion and controls: 1st number dimensionless, 2nd number dimensional

1) angular velocity = deg/sec
2) linear velocity, gust velocity = ft/sec or m/sec
3) \( \dot{\Psi}_s \) = rpm
4) \( \ddot{\alpha}_r \) = ft/sec^2 or m/sec^2
5) controls = deg

Print of Stability Derivatives (Program STABD)

Options:

a) rotor coefficient form, \( M^* = \text{dim} \times \omega \)
b) stability derivative form, \( \dot{x} \) (acceleration)
c) dimensionless or dimensional
Dimensions:

a) force or moment

<table>
<thead>
<tr>
<th>M^X form</th>
<th>forces</th>
<th>moments</th>
<th>torque</th>
</tr>
</thead>
<tbody>
<tr>
<td>X form</td>
<td>(\frac{1}{2}N_i\Omega^2/R)</td>
<td>(\frac{1}{2}N_i\Omega^2)</td>
<td>(N_i\Omega^2)</td>
</tr>
</tbody>
</table>

b) subscripts

- acceleration \((\ddot{z})\) = \(\Omega^2R\) (FA)
- angular velocity = \(\Omega\)
- linear velocity = \(\Omega R\) (FV)
- controls = 57.3
- gust velocity = \(\Omega R\) (FV)

Print During Flight Dynamics Numerical Integration (Program STABP)

a) controls in deg

b) gust velocity: 1st number dimensionless, 2nd number dimensional

c) aircraft motion: 1st number dimensionless, 2nd number dimensional

1) displacement = deg, ft or m
2) velocity = deg/sec, ft/sec or m/sec
3) acceleration = deg/sec^2, g
4) inertial axes = deg/sec, g

Print Transient Solution (Program TRANP)

a) controls in deg

b) gust velocity dimensional

c) aircraft motion: 1st number dimensionless, 2nd number dimensional

1) displacement = deg, ft or m
2) velocity = deg/sec, ft/sec or m/sec
3) acceleration = deg/sec^2, g
4) inertial axes = deg/sec, g
d) generalized forces: moments and forces in $\mathcal{C}/\mathcal{V}$ a form (rotor #1 parameters, body axes); torque in $-\mathcal{C}/\mathcal{V}$ a form (rotor #1 parameters)
8. UNITS

The program will work with English or metric (SI) units for input and output. Some of the input parameters and most of the internal program parameters are dimensionless (based on the rotor radius, the rotor rotational speed, and the air density). The units for input and output parameters are based on the consistent mass-length-time system (foot-slug-second or meter-kilogram-second), with the following exceptions:

a) The aircraft gross weight is input in pounds or kilograms.

b) The aircraft velocity is input in knots for both systems of units (alternatively the dimensionless speed can be input).

c) Power is output in horsepower for both systems of units.

The "dimensional" output for angles is in degrees; the "dimensionless" form for angles is in radians.
9. AIRFOIL TABLE PREPARATION

This section describes a program that constructs airfoil table files in the form required by the rotor analysis. The program will also print or printer-plot the airfoil data in the file being created or in an existing file. The airfoil tables are constructed using either analytical expressions or an airfoil table deck (in C81 format). The subprogram functions and namelist input labels are summarized below.

Subprogram Name

- MAINTB: Airfoil table preparation (main program)
- AERCT: Interpolate airfoil tables
- AERCPP: Printer-plot airfoil aerodynamic characteristics

Namelist Label

- NL'ABL: Table and print/plot data
- NLCHAR: Airfoil characteristics data

The structure of a job to run the airfoil table preparation program is defined below. The basic structure consists of the following steps:

1) Airfoil file definition
2) Main program call
3) Title card
4) Namelist NLTABL
5) For each radial station (CPREAD ≠ 0), either
   a) Namelist NLCHAR (CPREAD = 1)
   b) Airfoil table card deck (CPREAD = 2)

Sample jobs are presented below.

Create airfoil table using analytical expressions.

DDEF FT40F001,,AIRFOIL
CALL MAINPROG
title card
&NLTABL table data,NFAF=40,CPREAD=1,&END
&NLCHAR airfoil characteristics data,&END
&END
Create airfoil table using C81 format airfoil card deck

DDEF FT40F001,,AIRFOIL
CALL MAINPROG
title card
 &NLTABL table data,NFAF=40,CPREAD=2,&END
 : airfoil card deck
 : &END

Print and plot airfoil table data

DDEF FT40F001,,AIRFOIL
CALL MAINPROG
blank card
 &NLTABL output data,NFAF=40,CPREAD=0,&END
&END

The following pages described the input variables and data for the airfoil table preparation program.

**First Card**

**TITLE(20)** title (80 characters); blank card for CPREAD EQ 0

**Namelist NLTABL**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAB</td>
<td>number of boundaries, ( N_A ); maximum 20</td>
</tr>
<tr>
<td>NA(NAB)</td>
<td>indices at boundaries, ( n_k )</td>
</tr>
<tr>
<td>A(NAB)</td>
<td>( \alpha ) at boundaries (deg, (-180^\circ) to (180^\circ))</td>
</tr>
<tr>
<td>NMB</td>
<td>number of boundaries, ( N_r ); maximum 20</td>
</tr>
<tr>
<td>N2(NMB)</td>
<td>indices at boundaries, ( n_k )</td>
</tr>
<tr>
<td>M(NMB)</td>
<td>( \rho ) at boundaries (0. to 1.)</td>
</tr>
<tr>
<td>NRB</td>
<td>number of segments, ( N_r ); maximum 10</td>
</tr>
<tr>
<td>R(NRB+1)</td>
<td>boundaries of segments ( (R(1)=0. , R(NRB+1)=1.) )</td>
</tr>
</tbody>
</table>

maximum \( NAB*NMB*NRB = 5000 \)
OPPRNT(3) integer parameter controlling output; EQ 0 to suppress; default value is 1
   (1) interpolate and print
   (2) interpolate and plot
   (3) list tables

NMPRNT number of Mach number values for print and plot; maximum 10

MPRNT(NMPRNT) Mach number values for print and plot

NAPRNT number of angle of attack values for print; maximum 60

APRNT(NAPRNT) angle of attack values (deg)

NFAF unit number for airfoil table file (default 40)

OPREAD integer parameter: EQ 0 to read airfoil table and print data only; EQ 1 to create airfoil table using analytical expressions, write airfoil file, and print data (default); EQ 2 to create airfoil table using C81 format airfoil card deck, write airfoil file, and print data

Namelist NLCHAR (for each radial station; if CPREAD = 1)

CLA
   \[ a = \frac{c_{max}}{M} \] at M = 0 (per rad) (default 5.7)

MDIV lift divergence Mach number \( M_{div} \) (default 0.75)

\( \nu_{\text{max}} \) Mach number \( M_{\text{max}} \) at M = 0 (default 1.2)

FSTALL factor \( f_s \) for \( c_{\text{max}} \) (default 0.5)

MSTALL Mach number \( M_s \) for \( c_{\text{max}} \) (default 0.4)

GSTALL factor \( g_s \) for stall \( C_d \) (default 1.2)

HSTALL factor \( h_s \) for stall \( C_d \) (default 0.4)

CLF \( c_{\text{A}} \) for stall \( C_d \) (default 1.12)

CMAC \( c_{\text{mac}} \) (default 0.1)

CMS \( c_{\text{ms}} \) (default -0.07)

DELO \( \delta_0 \) (default 0.0084)

DELI \( \delta_1 \) (default -0.0102)

DEL2 \( \delta_2 \) (default 0.394)

DQDIM \( \Phi_{d}/M \) (default 0.65)

MCRT critical Mach number at \( \alpha = 0 \) (default 0.83)

ACRT critical Mach number zero at \( \alpha = \alpha_{\text{crit}} \) (default 33.)

ALFD drag stall angle (deg) (default 10.)

CIDF \( c_{d} \) for stall \( C_d \) (default 2.05)

-89-
Airfoil Card Deck (for each radial station; if OPREAD = 2)

I. Header
   a) Card 1, format (30A1.612)
      title, 30 alphanumeric characters
      NML, number of Mach number entries in c_a table
      NAL, number of angle of attack entries in c_a table
      NMD, number of Mach number entries in c_d table
      NAD, number of angle of attack entries in c_d table
      NMM, number of Mach number entries in c_m table
      NAM, number of angle of attack entries in c_m table

II. Lift Coefficient Table
   b) Card 2, format (7X,9F7.0); 2 or more cards if NML \geq 10
      Mach numbers M_1 to M^{NML}
   c) Card 3a, format (F7.0,9F7.0)
      angle of attack, \( \alpha \)
      lift coefficients c_a at M = M_1 to M^{NML} or M^9
   Card 3b, format (7X,9F7.0); 1 or more cards if NML \geq 10
      lift coefficients c_a at M = M_{10} to M^{NML}
   d) repeat card 3 for \( \alpha = \alpha_1 \) to \( \alpha^{NAL} \)

III. Drag Coefficient Table
   e-g) format as or lift coefficient table

IV. Moment Coefficient Table
   h-j) format as for lift coefficient table

V. Parameter Limits
   a) \( M_1 = 0 \); data extrapolated for M \geq M^{NM}; Mach numbers in
      sequential order
   b) \( \alpha_1 = -180^\circ, \alpha_{NA} = 180^\circ \); angles of attack in sequential order
   c) NM \geq 2, NA \geq 2 for lift, drag, and moment
   d) \((NM+1)(NA+1) \leq 501\) for lift, 1101 for drag, 576 for moment
For $\text{OPREAD} = 1$, the program calculates representative airfoil characteristics using the following expressions (refer to the accompanying figures).

A) Below stall

\[
c_\alpha = \begin{cases} 
\frac{a}{\sqrt{1-M^2}} & M < M_{\text{div}} \\
\frac{a(1-M)/((1-M_{\text{div}})\sqrt{1-M_{\text{div}}^2})}{1-M_{\text{div}}} & M_{\text{div}} < M < M_{\text{div}} + 1 \\
\frac{a[(1-M)/((1-M_{\text{div}})\sqrt{1-M_{\text{div}}^2}) + (M-M_{\text{div}}-1)/(1-M_{\text{div}}-1)]}{1-M_{\text{div}}} & M > M_{\text{div}} + 1
\end{cases}
\]

\[
c_\alpha' = a^2 \cos \alpha
\]

\[
c_m = c_{m_{\text{ac}}}
\]

\[
c_d = \delta_0 + \delta_1 \alpha + \delta_2 \alpha^2 + \Delta c_d
\]

\[
\Delta c_d = \max(0, \Delta c_d/\omega M (M-M_c))
\]

\[
M_c = \max(0, M_{\text{crit}} (1-\alpha/\alpha_{\text{crit}}))
\]

B) Stall angle

\[
c_{\alpha_s} = c_{\alpha_{\text{max}}} \min \left(1, \frac{(1-M) + f_s(M-M_s)}{1-M_s}\right)
\]

\[
\alpha_s = c_{\alpha_s}/c_{\alpha_{\text{max}}}
\]

C) Stalled lift ($|\alpha| > \alpha_s$)

\[
c_\alpha = \text{sign}(\alpha) \max \left[ \frac{(g_s \alpha_s - |\alpha|)c_{\alpha_s} + (|\alpha| - \alpha_s)h_s c_{\alpha} G_s}{c_{\alpha_s} - \alpha_s}, \max(h_s c_{\alpha_s}, c_{\alpha_s} \sin 2|\alpha|) \right]
\]

\[
c_{\alpha} = c_{\alpha_f} \sin 2 \alpha \quad \text{if } |\alpha| > 45^\circ
\]
D) Stalled moment \( |\alpha| > \alpha_s \)

\[
c_m = \begin{cases} 
\text{sign}(\alpha) \frac{60 - |\alpha| c_{ms}}{60 - \alpha_s} + (1 - \alpha_s c_{ms}) \cdot 75 c_{mf} & |\alpha| < 60^\circ \\
\text{sign}(\alpha) \frac{(90 - |\alpha|) c_{mf} + (1 - \alpha) - 60 c_{mf}}{30} & |\alpha| > 60^\circ 
\end{cases}
\]

\[c_{mf} = -\frac{1}{4} c_d(\alpha = 90) = -\frac{1}{4} (c_d(\alpha = \alpha_d) + c_{df})\]

E) Stalled drag \( |\alpha| > \alpha_d \)

\[c_d = c_d(\alpha = \alpha_d) + c_{df} \sin \left(\frac{|\alpha| - \alpha_d}{90 - \alpha_d} \cdot 90\right)\]

F) Reverse flow \( |\omega| > 90 \)

use effective angle of attack and account for moment axis shift

\[\alpha_e = \alpha - \frac{\pi}{2} \text{sign} \alpha\]

\[c_m = c_m + \left(\frac{1}{2} \cos \alpha_e\right) c_{\lambda} + \left(\frac{1}{2} \sin \alpha_e\right) c_d\]
a. Lift and drag information

Fig. 1.- Airfoil Characteristics

-93-
b. Moment and lift curve slope

Fig. 1.- Concluded