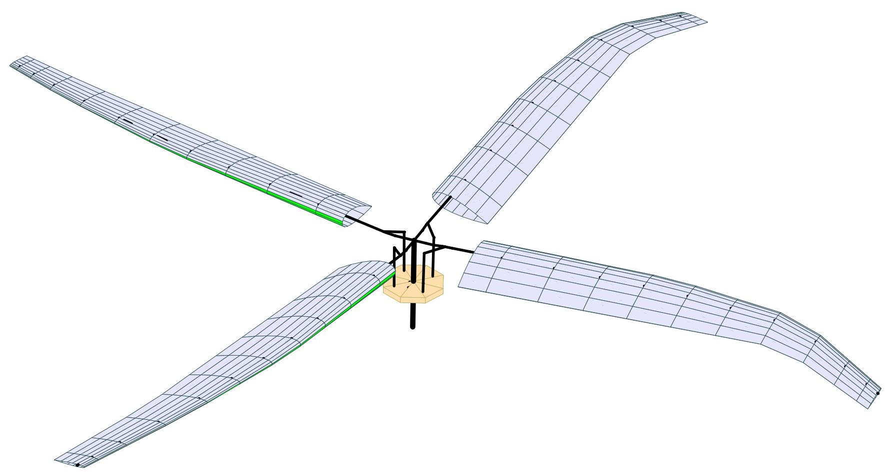
CAMRAD II

COMPREHENSIVE ANALYTICAL MODEL OF ROTORCRAFT AERODYNAMICS AND DYNAMICS

Volume VII: Rotorcraft Training



Wayne Johnson

Johnson Aeronautics
Palo Alto, California

Distributed by

AMI Aero, LLC
Redmond, Washington

CAMRAD II

COMPREHENSIVE ANALYTICAL MODEL OF ROTORCRAFT AERODYNAMICS AND DYNAMICS

Volume VII: Rotorcraft Training

Release 5.0

Wayne Johnson

Johnson Aeronautics

Published by Johnson Aeronautics, Palo Alto, California

CONTENTS

Chapter 1

INTRODUCTION

CAMRAD II IS AN AEROMECHANICAL ANALYSIS OF HELICOPTERS AND ROTORCRAFT

INCORPORATING ADVANCED TECHNOLOGY
- multibody dynamics
- nonlinear finite elements
- rotorcraft aerodynamics and wakes

CAMRAD II IS A COMPREHENSIVE ANALYSIS

for design, testing, evaluation of rotors and rotorcraft

at all stages, including research, conceptual design, detailed design, and development

calculates performance, loads, vibration, response, stability

with a consistent, balanced, yet high level of technology in a single computer program

applicable to wide range of problems, wide class of rotorcraft

SUCH CAPABILITY IS ESSENTIAL FOR HELICOPTER PROBLEMS, WHICH ARE INHERENTLY COMPLEX AND MULTIDISCIPLINARY

CAMRAD II USES A BUILDING BLOCK APPROACH TO ACHIEVE FLEXIBILITY IN MODEL OF DYNAMIC AND AERODYNAMIC CONFIGURATION

- CAN MODEL TRUE GEOMETRY OF ROTORCRAFT
- MULTIPLE LOAD PATHS
  - swashplate and control system; lag dampers; tension/torsion straps; bearingless rotors
- VIBRATION CONTROL DEVICES
  - such as pendulum absorbers or active control
- ARBITRARY HINGE ORDER
- DROOPED AND SWEPT TIPS
- ARBITRARY ELASTIC AXIS AND QUARTER CHORD
- DISSIMILAR BLADES

CAMRAD II PROVIDES A POWERFUL ANALYSIS CAPABILITY

- ADVANCED ROTOR AERODYNAMICS
- RIGOROUS KINEMATICS AND DYNAMICS
  - consistent structural loads and dynamic response; general interfaces between aerodynamic and structural dynamic components
- GENERAL TRANSIENTS

CAMRAD II IS A PRACTICAL ENGINEERING TOOL

- CORE FOR FLEXIBILITY, ROTORCRAFT SHELL FOR EASE OF USE
  - shell builds typical rotorcraft and rotor models; core input always gives capability to change model
- RANGE OF COMPONENTS AND MODELLING OPTIONS
  - can balance efficiency and accuracy for a particular problem

CAMRAD II OFFERS A DESIGN FOR GROWTH

- appropriate platform for future developments, for continuing access to new technology

1–1 Background

CAMRAD II INCORPORATES COMBINATION OF ADVANCED TECHNOLOGY

MULTIBODY DYNAMICS
rigid body components, frames, and joints;
nonlinear kinematics; joint operations

but necessary to deal with elastic motion

STRUCTURAL DYNAMICS
substructure coupling and static residuals;
modal analysis and truncation;
and elimination of constraints

but necessary to deal with nonlinear systems

NONLINEAR FINITE ELEMENTS
nonlinear elements, numerical integration, and beam components

but necessary to deal with large motion (so do not use nodal motion as degrees of freedom)

rotorcraft problems involve more than just structural dynamics (so no system Lagrangian; and use explicit constraint forces)

ROTORCRAFT AEROMECHANICS
aircraft dynamics, rotating systems, and aerodynamics;
and define tasks and results required of analysis

GENERAL REQUIREMENTS FOR MODERN COMPREHENSIVE ANALYSIS

MUST CALCULATE PERFORMANCE, LOADS, VIBRATION, RESPONSE, STABILITY

so comprehensive analysis needs technology for rotor wake, drag and stall, nonlinear dynamics, entire aircraft

and must perform trim, transient, and flutter tasks

ANALYSIS MUST HANDLE COMPLEX CONFIGURATIONS

unusual load paths and interactions, and many subsystems

structural, aerodynamic, and kinematic nonlinearities

arbitrary large motion, including rigid body motions and rotation of components relative to each other

and not just equations and interfaces of structural dynamics

ANALYSIS MUST BE FLEXIBLE

separate specification of configuration, aeromechanical models, and solution procedures

ANALYSIS MUST BE EASY TO USE

especially for normal configurations

ANALYSIS MUST BE PRACTICAL

efficient, accurate, and reliable

1–2 Approach

CAMRAD II PERFORMS NONLINEAR DYNAMIC/STATIC ANALYSIS OF AEROMECHANICAL SYSTEM

CAMRAD II TASKS

- SOLVE DIFFERENTIAL, INTEGRAL, STATIC, AND IMPLICIT EQUATIONS FOR MOTION OF THE SYSTEM
  - AND EVALUATE REQUIRED OUTPUT QUANTITIES FROM RESPONSE
- TRIM, TRANSIENT, FLUTTER TASKS (figure 1)

TRIM TASK

- EQUILIBRIUM SOLUTION FOR STEADY STATE OPERATING CONDITION
  - free flight: level flight, steady climb or descent, steady turns
  - constrained: rotor in wind tunnel; typically trim thrust and flapping to target values
  - identify control positions and aircraft orientation required to achieve specified operating condition
- STEADY OR PERIODIC RESPONSE OF SYSTEM
- UNIFORM INFLOW, NONUNIFORM INFLOW WITH PRESCRIBED WAKE GEOMETRY, OR NONUNIFORM INFLOW WITH FREE WAKE GEOMETRY

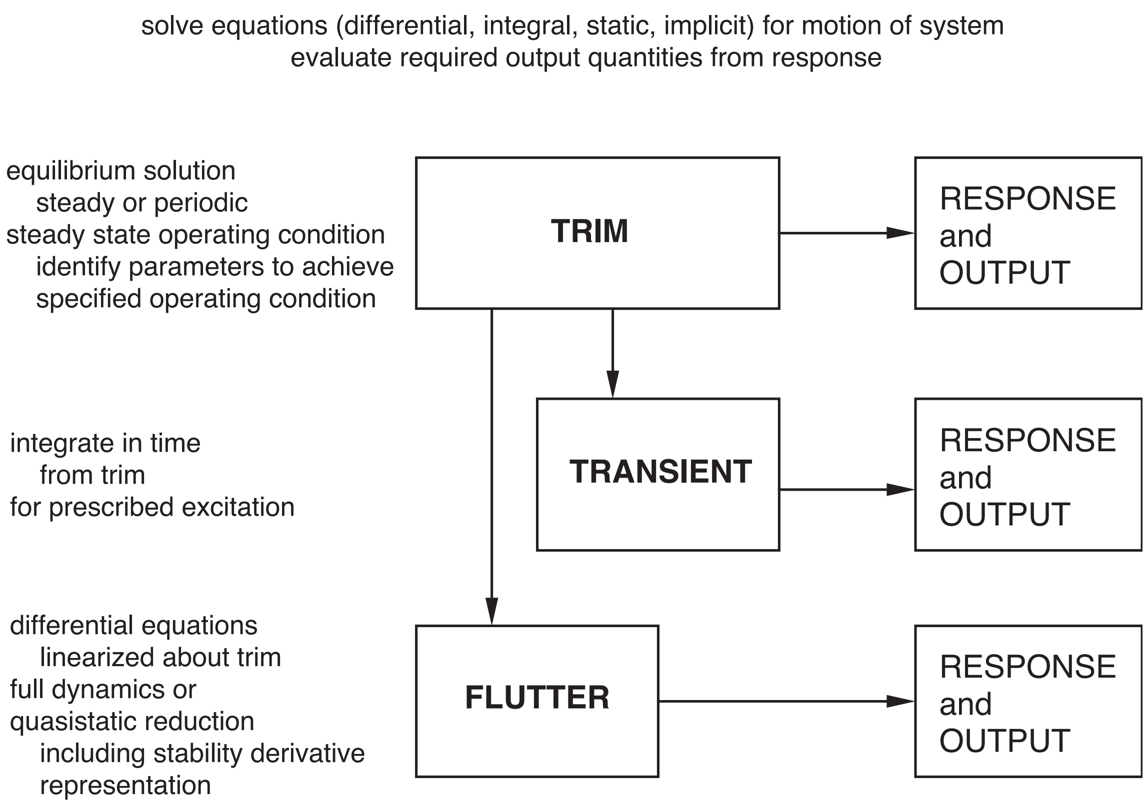


Figure 1-1 CAMRAD II tasks.

TRANSIENT TASK

NUMERICALLY INTEGRATE IN TIME (FROM TRIM SOLUTION) FOR PRESCRIBED EXCITATION

FLUTTER TASK

DIFFERENTIAL EQUATIONS, LINEARIZED ABOUT TRIM (PROBABLY NUMERICAL PERTURBATION)

full dynamics, or quasistatic reduction of selected variables (including stability derivative representation)

RESPONSE OF TIME-INVARIANT (PERHAPS AVERAGED) OR PERIODIC EQUATIONS

FLEXIBILITY OF CONFIGURATION MODEL

MODERN COMPREHENSIVE ANALYSIS MUST BE ABLE TO ANALYZE ARBITRARY CONFIGURATION — WHATEVER DESIGNERS CAN INVENT

OLD APPROACH (CAMRAD/JA):

- FIXED GEOMETRY AND DYNAMIC MODELS, FIXED AERODYNAMIC MODELS
  - new rotor or helicopter configuration requires new development of equations
  - structural dynamic and aerodynamic models mixed; can not change one without considering other

REQUIREMENT:

- CONFIGURATION DEFINED AND CHANGED BY INPUT TO ANALYSIS
- DO NOT NEED NEW CODE AS LONG AS REQUIRED PHYSICS AVAILABLE
  - separate physical and mathematical models from definition of configuration
  - separate structural dynamic and aerodynamic models

CAMRAD II:

- USES BUILDING-BLOCK APPROACH TO ACHIEVE FLEXIBILITY
- flexibility and generality of system configuration
  - obtained by assembling standard components with standard interfaces
  - and solving system using standard procedures

CAMRAD II SYSTEM DESCRIPTION

FOR CONFIGURATION GENERALITY, CAMRAD II SPLITS SYSTEM INTO PIECES, WITH CONNECTIONS BETWEEN

PIECES AVAILABLE TO CONSTRUCT SYSTEM:

environmental	physical	logical
case	component	loop
wind	frame	part
operating condition	interface	transform
period	output	modes
	input	response
		weights

environmental pieces: describe system operation
physical pieces: produce system equations
logical pieces: solve system equations

STANDARD DESCRIPTIONS OF COMPONENTS, INTERFACES, AND SOLUTION PROCEDURES REQUIRED

structural dynamic systems are important subset of problems

aerodynamic models also important

CONVENTIONS AND METHODOLOGY MUST ENCOMPASS ALL TYPES OF COMPONENTS, INTERFACES, AND SOLUTION PROCEDURES

THESE SYSTEM PIECES CONSTITUTE THE CORE ANALYSIS, PROVIDING A FLEXIBLE, BUILDING-BLOCK ORIENTED MODELLING CAPABILITY

CAMRAD II COMPONENTS

COMPONENTS PERFORM ALL COMPUTATIONS ASSOCIATED WITH PHYSICS OF A MODEL OF SYSTEM

development of improved model requires constructing new component, which will fit into existing analysis framework

COMPONENTS AVAILABLE TO CONSTRUCT SYSTEM:

rigid body	Fourier series	wing wake geometry
linear normal modes	prescribed control	rotor wake geometry
finite element beam	gust	
rod/cable		wing performance
	rigid airframe aerodynamics	rotor performance
transmission	airframe flow field	rotorcraft performance
reference frame		
	lifting line wing	helicopter tail boom
filter	rigid wing	computational fluid dynamics
reference plane		
	wing inflow	
differential equation	rotor inflow	
programmable	rotor dynamic wake	
transfer function	wing wake	plugin

structural dynamic, aerodynamic, differential equation, and performance models

REQUIRE STANDARD FORM FOR COMPONENTS

must accommodate all kinds of components

must also handle and use special characteristics of structural dynamic components

STRUCTURAL DYNAMIC COMPONENTS HAVE COMMON CHARACTERISTICS

common implementation for structural dynamic components — differ in model of elastic motion

OTHER KINDS OF COMPONENTS AS WELL, INCLUDING AERODYNAMICS

can not assume have symmetric, second order, differential equations

can not assume constraint equations are obtained from displacements

can not base component model on system Lagrangian

FLEXIBILITY OF SOLUTION PROCEDURE

DEFINITION OF SOLUTION PROCEDURE MUST BE JUST AS FLEXIBLE AS DEFINITION OF CONFIGURATION

OLD APPROACH (CAMRAD/JA):

- ONLY ONE SOLUTION METHOD; LITTLE SYSTEMATIC DEVELOPMENT OF SOLUTION PROCEDURES
  - solution procedure and physical models mixed — can not change one without considering other

REQUIREMENT:

- SOLUTION PROCEDURE DEFINED AND CHANGED BY INPUT TO ANALYSIS
- DO NOT NEED NEW CODE AS LONG AS REQUIRED PROCEDURES AVAILABLE
  - separate solution procedure from aeromechanical model and configuration

CAMRAD II:

- USES BUILDING-BLOCK APPROACH TO ACHIEVE FLEXIBILITY

EXPANDABILITY

- BUILDING-BLOCK APPROACH ESSENTIAL FOR EXPANDABILITY

- MUST SEPARATE SPECIFICATION OF CONFIGURATION, AEROMECHANICAL MODEL, AND SOLUTION PROCEDURE
  - OTHERWISE SMALLEST CHANGE INVOLVES ENTIRE ANALYSIS
  - growth becomes increasingly harder as each new feature added

- DEVELOPMENTS SINCE INITIAL RELEASE HAVE DEMONSTRATED EXPANDABILITY OF CAMRAD II

- EVEN MAJOR EXTENSIONS TYPICALLY REQUIRE ONLY ONE OR TWO WEEKS
  - including documentation and testing
  - NOT INCLUDING DEVELOPMENT OF THEORY

MATHEMATICAL MODEL

- MINIMIZE APPROXIMATIONS AT TOP LEVEL OF ANALYSIS
  - ESSENTIAL FOR EXPANDABILITY — TOP LEVEL APPROXIMATIONS AND ASSUMPTIONS ARE VERY DIFFICULT TO CHANGE

- CAMRAD II:
  - finite-dimension description of system
    - components and interfaces introduce spatial discretization
    - solution procedures introduce time discretization
  - structural dynamic interfaces are holonomic and independent

- NO FURTHER APPROXIMATIONS AT TOP LEVEL
  - coupling and solution procedures can handle arbitrary and exact models, with nonlinear and time-varying equations

- APPROXIMATIONS ARE IN COMPONENTS
  - required for practical solution of most problems
  - new technology and more accurate models introduced by constructing new components, without changing framework of analysis

MATHEMATICAL MODEL

BUILDING BLOCK APPROACH LEADS TO MORE GENERAL, MORE RIGOROUS MODELS

separate physical and logical pieces
separate structural dynamic and aerodynamic models

each piece capable of general analysis

OLD APPROACH (CAMRAD/JA)

solution procedure: mixes part/loop solution for rotor and airframe motion; does not identify all iteration variables

wing model: assumes geometry is that of specific structural model; first order velocity and displacement

CAMRAD II

solution procedure: sound mathematical basis; automatically finds iteration variables

wing model: general line geometry; droop and sweep no different than flap, lag, and bending, and all can be large; can connect to any structural model, with exact kinematics of velocity, displacement, force interface

MATHEMATICAL MODEL

- EXACT KINEMATICS OF RIGID BODY MOTION AND STRUCTURAL DYNAMIC INTERFACES
  - large displacement, exact kinematics of large rotations
  - ASSEMBLY OF SYSTEM NOT IMPLEMENTED BY IDENTIFYING DEGREES OF FREEDOM OR VARIABLES OF TWO COMPONENTS AS EQUAL
    - must solve nonlinear, time-varying constraint equations for some variables (degrees of freedom or forces)

- REQUIRED TO MODEL ARBITRARY CONFIGURATIONS, ARBITRARY MOTION

- ESSENTIAL FOR CAPABILITY TO MODEL ROTOR HUB CONFIGURATIONS
  - otherwise rotating-to-nonrotating frame connection is special (large rotation)
  - OLD APPROACH (CAMRAD/JA): one rotating-to-nonrotating frame connection, at hub node; pitch bearing with spring/actuator
    - control system load path model wrong without swashplate node
  - CAMRAD II: arbitrary, general hub and swashplate model

TRANSPORTABILITY

CAMRAD II MUST BE EASILY INSTALLED ON MANY DIFFERENT COMPUTERS

installation on more than 10 different machines so far

FOR TRANSPORTABILITY, WRITTEN IN STANDARD FORTRAN

BUT FORTRAN DOES NOT HAVE DATA STRUCTURES NEEDED TO IMPLEMENT FLEXIBLE ANALYSIS

need records and lists; arbitrary numbers and arbitrary sizes

CAMRAD II WRITTEN USING SOFTWARE TOOL

TOOL DESIGNED FOR DEVELOPMENT, MODIFICATION, AND MAINTENANCE OF LARGE FORTRAN PROGRAM

emphasizes definition, access, and documentation of data structures

dictionary of data structures; prologue defines subroutine communication through data structures; variables uses almost like standard Fortran variables; translator produces compilable source

EASE OF USE

COMPREHENSIVE ANALYSIS MUST BE EASY TO USE

- especially for normal or typical configurations
  - not same as saying helicopters should be easy to analyze

FLEXIBLE ANALYSIS (CORE SYSTEM) REQUIRES LARGE AMOUNT OF DETAILED INPUT INFORMATION

- helicopter is complex system — 1000's of system pieces

CAMRAD II HAS ROTORCRAFT SHELL TO CONSTRUCT CORE INPUT FOR TYPICAL CONFIGURATIONS AND TYPICAL PROBLEMS

- builds multi-rotor aircraft, in free flight or wind tunnel
- N-bladed rotor with general hub, perhaps with swashplate control
- constructs solution procedure as well
  - figure 2 is simplified description of model
    figure 3 shows example of rotor constructed

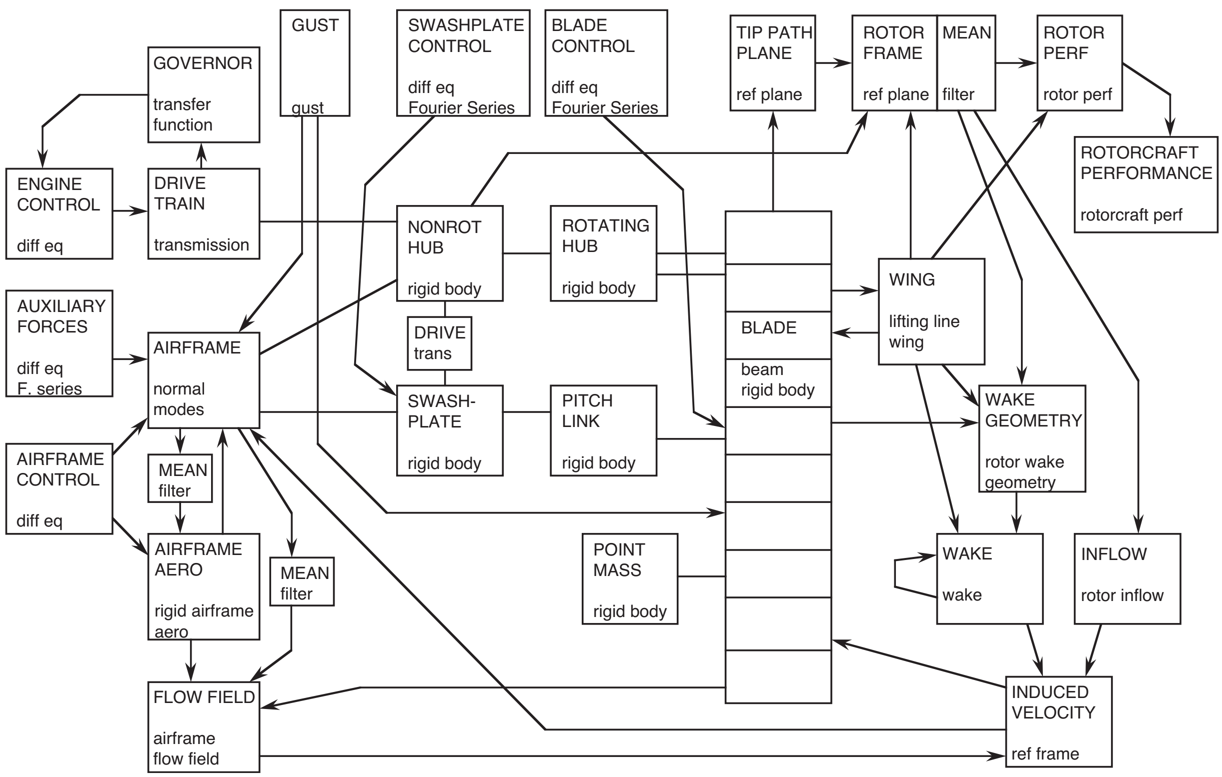


Figure 1-2 Rotorcraft model (simplified).

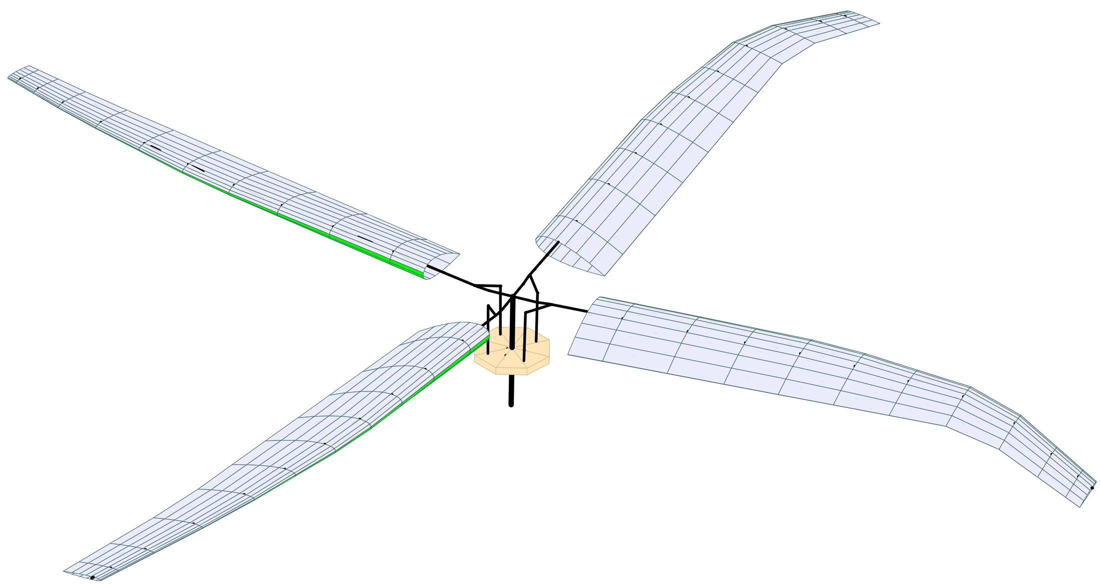


Figure 1-3 Example of rotor constructed.

SHELL ISOLATES USER FROM DETAILS OF SYSTEM DEFINITION

ACCOMPLISH 2-3 ORDERS OF MAGNITUDE REDUCTION IN AMOUNT OF INPUT INFORMATION

SHELL DOES NOT HAVE FLEXIBILITY OF CORE INPUT, AND MAY NOT BE ABLE TO MODEL EXACTLY EVERY CONFIGURATION

- SHELL CONSTRUCTS MOST OF SYSTEM
  - minimizing need to deal directly with core input
  - what shell constructs will provide guidance for use of core input
- CAN USE CORE INPUT TO CHANGE MODEL CONSTRUCTED BY SHELL

1–3 Sources of Information

CAMRAD II DOCUMENTATION

VOLUME I:	THEORY
VOLUME II:	COMPONENTS THEORY
VOLUME III:	ROTORCRAFT THEORY
VOLUME IV:	INPUT
VOLUME V:	COMPONENTS INPUT
VOLUME VI:	ROTORCRAFT INPUT
VOLUME VII:	ROTORCRAFT TRAINING
VOLUME VIII:	PROGRAM
VOLUME IX:	PROGRAM TRAINING

SAMPLE JOBS
EXTRAS
NEWSLETTER
SOFTWARE SUPPORT

WEB SITE: www.johnson-aeronautics.com or www.camrad.com

documentation, technical papers, sample jobs, and extras can be downloaded by customers

1–4 General Warnings and Limitations

NOT YET POSSIBLE TO ACCURATELY CALCULATE OR PREDICT AEROMECHANICAL SYSTEM BEHAVIOR IN ALL CIRCUMSTANCES

always possible to get bad answer from bad input or misuse of analysis

also possible that best results of analysis may not be accurate

computer program is a tool, not substituting for but rather to be used with judgement, experience, and much testing of the actual system

ANALYSIS CONSTRUCTS SYSTEM BY DIVIDING IT INTO COMPONENTS, WITH EACH COMPONENT A MODEL OF THE ACTUAL DEVICE

rarely are complete equations being solved for exact response

empiricism and approximations are required in order to make the analysis practical

engineering judgement required to use and interpret analysis, based on experience with code, particularly projects using code to correlate with measured results

Chapter 2

RUNNING CAMRAD II PROGRAMS

GENERAL REFERENCE FOR CHAPTER: CAMRAD II Documentation, Volume VI, Rotorcraft Input; Chapter “Running CAMRAD II Programs”

2–1 Main Programs

CAMRAD II MAIN PROGRAMS

ROTORCRAFT ANALYSIS (CAMRADII)
INPUT AND TABLE FILE PREPARATION (INPUT)
PLOT FILE EXAMINATION (OUTPUT)

programs communicate by means of files

figure 1 shows relation between programs for typical application

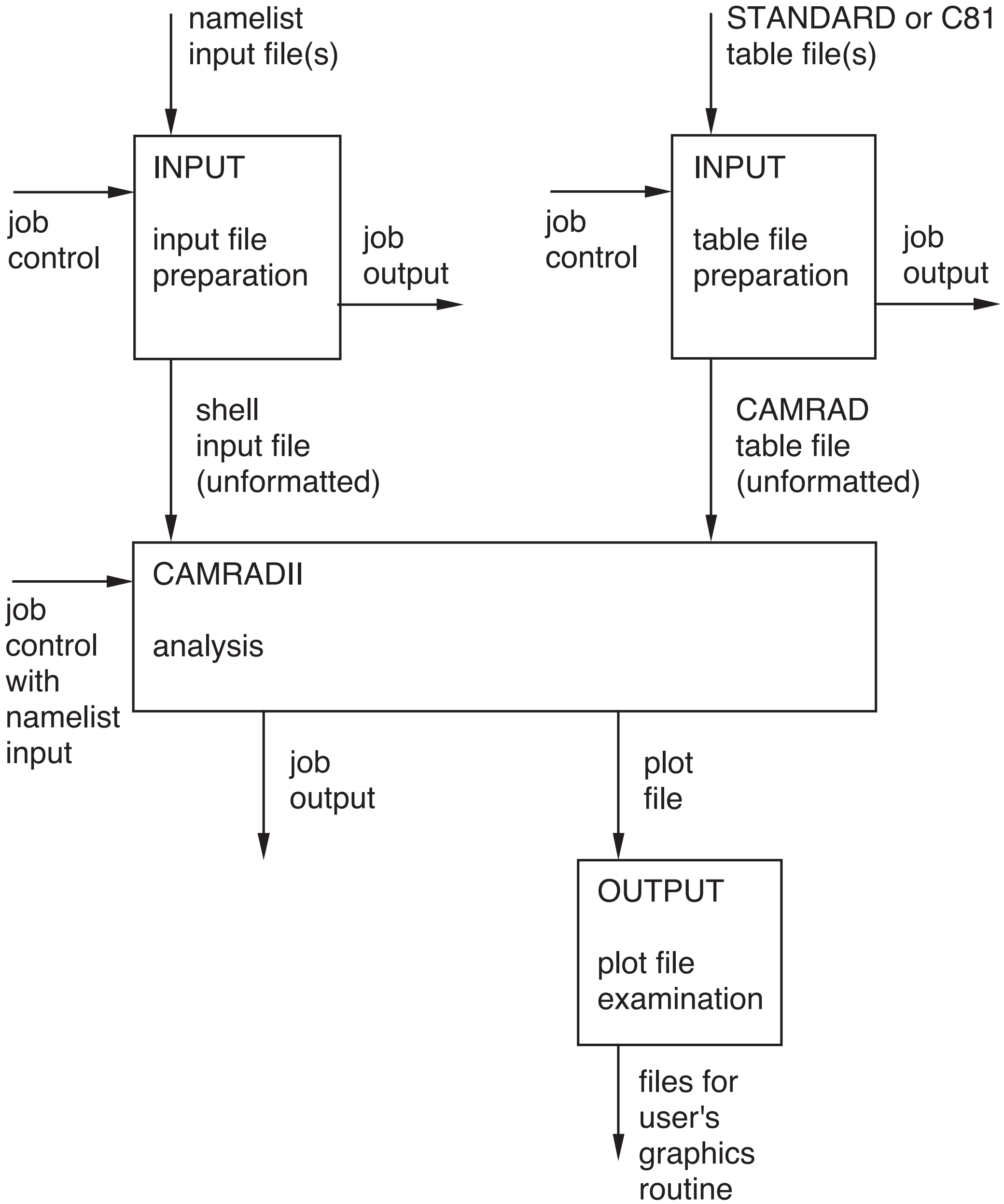


Figure 2-1 Relation between CAMRAD II programs.

TYPICAL STEPS TO RUN CAMRAD II

CREATE AIRFOIL TABLE FILE

program INPUT

creates `CAMRAD` (unformatted) airfoil table file for entire blade

from `STANDARD` (free form) or `C81` (fixed form) airfoil table file(s)

CREATE SHELL INPUT FILE

program INPUT

creates shell input file (unformatted)

from namelist file(s) of baseline parameters (shell input blocks)

RUN ROTORCRAFT ANALYSIS

program CAMRADII

reads airfoil table file and shell input file

one or more cases per job
namelist input for each case (shell input blocks)
create job output and plot file

EXAMINE PLOT FILE

program OUTPUT

creates files for user's graphics routine

from plot file of rotorcraft analysis

FILE NAME CONVENTIONS REQUIRED TO ORGANIZE FILES

TYPICAL FILE NAME EXTENSIONS:

`.STD`	`STANDARD` airfoil table file
`.C81`	`C81` airfoil table file
`.TAB`	`CAMRAD` airfoil table file
`.LIST`	namelist input file
`.DAT`	shell input file
`.COM`	command file
`.OUT`	output file
`.PLOT`	plot file

GENERAL USE OF INPUT AND OUTPUT PROGRAMS

INPUT FILE PREPARATION

PREPARE UNFORMATTED FILES FROM NAMELIST DATA; PRINT INPUT DATA
batch or interactive

EXTRACT DATA TO PLOT OR DRAW INPUT; EXAMINE INITIALIZED SYSTEM DATA
interactive (or script file)

TABLE FILE PREPARATION

PREPARE FORMATTED OR UNFORMATTED FILES; PRINT TABLE DATA
for each table class/type
batch or interactive

EXTRACT DATA TO PLOT TABLE
interactive (or script file)

PLOT FILE EXAMINATION

READ AND EXTRACT DATA FROM PLOT FILE; PRINT LINEAR DIFFERENTIAL EQUATIONS TO LOG FILE
interactive (or script file)

2–2 Input

2-2.1 Shell and Core Input

CORE INPUT PROVIDES A FLEXIBLE, COMPONENT-ORIENTED ANALYSIS

- core input defines tables required

SHELL INPUT FACILITATES APPLICATIONS TO SPECIFIC PROBLEMS

SHELL INPUT CONSISTS OF BOTH SHELL AND CORE NAMELIST

- figure 2 shows typical CAMRADII input
- ANALYSIS USES SHELL INPUT TO PRODUCE CORE DATA, WHICH USER CAN THEN CHANGE USING CORE INPUT
  - core namelist input stored as changes to core data produced by shell, so these changes can be implemented for subsequent cases in job (without repeating core namelist for each case)
- INPUT CONSISTS OF SET OF SECTIONS
  - identified by class, type, and name
  - each section corresponds to a system piece
- A SHELL OR CORE SECTION IS READ FROM NAMELIST
- SHELL OR CORE INPUT CAN BE STORED AS UNFORMATTED FILE
  - this file can be read before namelists, to initialize all sections

```
setenv BLADEAIRFOIL1 advtecheq.tab
setenv SHELLINPUT windtunnel.dat
setenv PLOTFILE sample.plot
/camrad/camradii > sample.out << ’endofinput’
 &NLJOB PLFILE=1,&END
 !========================================================
 &NLDEF class=’CASE’,&END
 &NLVAL FLTASK=1,&END
 !========================================================
 &NLDEF class=’TRIM’,&END
 &NLVAL WVEL=.4,COLL=10.,PITCH=-5.,LEVEL=2,NWPRNT=1,&END
 &NLDEF class=’TRIM ROTOR’,name=’ROTOR 1’,&END
 &NLVAL MASEN=1,MBSEN=1,MHSEN=1,MPSEN=1,&END
 !========================================================
 &NLDEF class=’FLUTTER’,&END
 &NLVAL DOFM=5*0,&END
 !========================================================
 &NLDEF class=’ROTOR’,type=’WAKE’,name=’ROTOR 1’,&END
 &NLVAL OPFW=2,&END
 !========================================================
 &NLDEF class=’AIRFRAME’,type=’STRUCTURE’,&END
 &NLVAL OPAERO=1,&END
 &NLDEF class=’AIRFRAME’,type=’AERODYNAMICS’,&END
 &NLVAL VISRC=1,&END
 !========================================================
 &NLDEF action=’end of shell’,&END
 &NLDEF action=’end of core’,&END
’endofinput’
```

Figure 2-2 Typical CAMRADII input.

INPUT SECTIONS AND TABLES ARE UNIQUELY IDENTIFIED BY CLASS, TYPE, AND NAME

ANALYSIS DEFINES ALLOWABLE CLASSES, AND TYPES FOR EACH CLASS

USER DEFINES THE NAMES

within a class, names must be unique, even if the type is different

store and search are based on class and name
type defines form of data within a class

EACH CLASS/TYPE CORRESPONDS TO SET OF COMMONS

POSSIBLY THERE IS NO TYPE FOR A CLASS

POSSIBLY THERE IS ONLY ONE BLOCK FOR A CLASS/TYPE
name is ignored

2-2.2 Table Input

TABLES CORRESPOND TO FILES

TABLES PROVIDE GENERAL FILE INPUT CAPABILITY
plot file used for general file output

TABLES PROVIDE MECHANISM FOR INPUT OF LARGE ARRAYS OF DATA

table data may be function of continuous variables (interpolated from the data vector)

or table data may be function of discrete variables (moved to record for use)

CORE INPUT DATA DEFINES WHICH TABLES ARE REQUIRED BY ANALYSIS

CORE SECTION CAN DEFINE MORE THAN ONE TABLE REQUIRED

SAME TABLE (SAME CLASS/TYPE/NAME) CAN BE REQUIRED BY MORE THAN ONE CORE SECTION

unit number and logical name of first requirement encountered are used to read table

USER CAN DELETE TABLES NOT USED FOR CASE, OR ALL TABLES AT END OF CASE

to minimize storage in table data vector

A TABLE SECTION IS READ FROM A FILE

FORMATTED OR UNFORMATTED
ONE FILE PER TABLE

TABLE SPECIFICATION CONSISTS OF CLASS, TYPE, AND NAME

PLUS FILE UNIT NUMBER
AND FILE NAME OR LOGICAL NAME

FILE NAME OR LOGICAL NAME CAN BE SAME AS “NAME” IN CLASS/TYPE/NAME SPECIFICATION

this is the default

FOR MOST INSTALLATIONS, CAN IGNORE UNIT NUMBERS

THERE IS A SPECIFIC FILE FORMAT FOR EACH TABLE CLASS/TYPE

2-2.3 System Definition and Names

NAMES (32-CHARACTER STRINGS)

FOR EACH CLASS THERE IS A COLLECTION OF SYSTEM PIECES, IDENTIFIED BY UNIQUE NAMES

USER IDENTIFIES THINGS IN THE INPUT BY NAME

THERE ARE SHORT LABELS (8-CHARACTERS) CORRESPONDING TO EACH NAME, FOR USE BY OUTPUT ROUTINES

for core input, a blank name or label produces the default (if a default exists)

EACH SYSTEM PIECE HAS A NAME

SAME AS NAME OF INPUT DATA SECTION FOR THE PIECE

WITHIN A CLASS, NAMES MUST BE UNIQUE, EVEN IF TYPE IS DIFFERENT

class and name used to identify system piece, hence to store and search the data

type serves to define optional structure of data within a class

SUBNAMES MAY BE NEEDED TO IDENTIFY A QUANTITY

for example, specifying a degree of freedom requires component name, vector name, and element name

TOP LEVEL NAME (FROM INPUT SECTION) MUST BE DEFINED BY USER

OTHER NAMES CAN BE DEFINED BY USER OR COMPUTER (IF DEFAULT EXISTS)

user can generally override any default name

there are defaults for all labels

WHEN USER MUST IDENTIFY SOMETHING BY NAME, OPTIONALLY IT CAN BE IDENTIFIED BY SEQUENCE NUMBER (AS A CHARACTER STRING) INSTEAD

USEFUL FOR SUBNAMES (VECTORS AND ELEMENTS), WHERE SEQUENCE NUMBER IS OBVIOUS

such as rigid body motion, force, or moment

when analysis searches for specified name and does not find it, then tries to translate name as valid sequence number

OPTIONALLY, THERE MAY BE NO ELEMENT NAMES FOR A QUANTITY

USEFUL WHEN VECTOR IS VERY LONG

BUT AN INPUT VECTOR OR A COMPONENT DEGREE-OF-FREEDOM VECTOR MUST HAVE ELEMENT NAMES

otherwise analysis can not label or select the element

2-2.4 Units of Input Parameters

DIMENSIONAL INPUT PARAMETERS

ENGLISH OR METRIC (SI) UNITS, USING CONSISTENT LENGTH-MASS-TIME SYSTEM

foot-slug-second or meter-kilogram-second

selected by parameter OPUNIT in class = CASE

ANGLES ARE INPUT IN DEGREES

EXCEPTIONS TO CONVENTION IN ROTORCRAFT SHELL:

AIRCRAFT GROSS WEIGHT IS IN LB OR KG

WIND SPEED AND AIRCRAFT SPEED CAN BE IN KNOTS

2–3 Shell and Core Namelist

SHELL AND CORE NAMELIST

- OCCUR IN JOB COMMAND STREAM OF PROGRAM CAMRADII
- OCCUR IN A FILE FOR PROGRAM INPUT

INPUT FOR ANALYSIS PROGRAM CAMRADII BEGINS WITH NAMELIST NLJOB (WHICH ALWAYS HAS DEFAULTS)

THEN THERE IS NAMELIST INPUT FOR EACH CASE OF JOB

- FIGURE 3 SHOWS FORM OF NAMELIST INPUT FOR A CASE (OR IN A FILE FOR PROGRAM INPUT)
  - consists of set (perhaps none) of definition and value (NLDEF–NLVAL) namelist pairs
    - arbitrary order
    - more than one namelist pair can be used for any input section (class/type/name)
  - case input must always conclude with a definition namelist (NLDEF) containing “action = end” (with no NLVAL namelist)
  - core input (perhaps just the concluding NLDEF) is required even when the shell is used

- THE END OF THE INPUT FOR ONE CASE IS IDENTIFIED BY THE PAIR OF “action = end” LINES:

```
$NLDEF action='END OF SHELL',$END
$NLDEF action='END OF CORE',$END
```

shell and core input:

```
$NLDEF class='xxxx',type='xxxx',name='xxxx',action='xxxx',$END
$NLVAL data,.....................................,data,$END
....
....
$NLDEF class='xxxx',type='xxxx',name='xxxx',action='xxxx',$END
$NLVAL data,.....................................,data,$END
$NLDEF action='END OF SHELL',$END
$NLDEF class='xxxx',type='xxxx',name='xxxx',action='xxxx',$END
$NLVAL data,.....................................,data,$END
....
....
$NLDEF class='xxxx',type='xxxx',name='xxxx',action='xxxx',$END
$NLVAL data,.....................................,data,$END
$NLDEF action='END OF CORE',$END
```

Figure 2-3 Program namelist input.

DEFINITION NAMELIST (NLDEF)

ONLY FOUR PARAMETERS: CLASS, TYPE, NAME, ACTION

all character strings, length 32

CLASS, TYPE, AND NAME IDENTIFY SHELL OR CORE INPUT BLOCK

possibly there is no type

possibly the name is ignored (only one block of this class/type)

lower case and some abbreviations and alternatives are accepted in CLASS and TYPE

VALUE NAMELIST (NLVAL)

CONTENTS DEPEND ON SECTION CLASS AND TYPE

EACH NLVAL NAMELIST CONTAINS CHARACTER VARIABLE C (or COMMENT, IF C IS ALREADY AN INPUT VARIABLE)

can be used to insert comments among the input data, if installation does not allow comments identified by exclamation point

PARAMETER ACTION IN NLDEF NAMELIST

VALUES: `zero`, `initialize`, `change`, `delete`, or `end`

DEFAULT IS ACTION = `'change'`

only first character in ACTION is used
both upper case and lower case are accepted

ACTION	function
`zero`	set all parameters in block to zero before read namelist
`initialize`	set all parameters in block to initial values before read namelist
`change`	get current values of parameters before read namelist; same as initialize for new block
`delete`	delete section
`end`	end input category

USUALLY CAN IGNORE PARAMETER ACTION (USE DEFAULT VALUE)

except to specify end of shell input and end of core input

THERE ARE BUILT-IN PARAMETER VALUES FOR EACH CORE AND SHELL BLOCK

- IMPLEMENTED BY INITIALIZATION AND SCENARIOS
- INITIALIZATION IS CONTROLLED BY PARAMETER ACTION IN NLDEF NAMELIST
  - gives option to set all parameters to default values, before reading NLVAL namelist
- SCENARIOS ARE CONTROLLED BY PARAMETER OPSCEN IN SOME NLVAL NAMELISTS
  - scenario is set of values for selected parameters, implemented immediately after reading NLVAL namelist
  - figure 4 shows typical use of initialization and scenarios

```
!============================================================
!==== initialization
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
&NLVAL &END
!============================================================
!==== forward flight wake scenario
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL OPSCEN=1,&END
!============================================================
!==== change wake extent
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL OPSCEN=0,RFW=4.,&END
!============================================================
```

Figure 2-4 Typical use of initialization and scenarios.

INPUT SCENARIOS

IMPLEMENTED IMMEDIATELY AFTER EACH READ OF NAMELIST NLVAL

PROCEDURE:

move existing section from general data storage to commons (or initialize the commons)

use namelist to read selected parameters into commons

implement scenarios

move section from commons to general data storage (in compressed form)

CONTROLLED BY PARAMETER OPSCEN IN EACH BLOCK (IF AVAILABLE)

SET OPSCEN = 0 FOR NO SCENARIO

a scenario provides set of values for other parameters in the namelist, superseding any other input value

so in order to change a parameter that is set by a scenario:

must use separate, subsequent NLDEF–NLVAL pair, containing OPSCEN = 0 and new parameter values

IF SCENARIO TURNED ON AFTER PARAMETERS HAVE ALREADY BEEN READ FOR A BLOCK

best to use ACTION='init' to reset all parameters to default values

EACH SHELL INPUT BLOCK HAS AN ELEMENTARY SCENARIO

CORRESPONDING TO SIMPLIFIED CONFIGURATION

these scenarios are used with initial values defined for shell input blocks

2–4 Summary of Input Process

FIGURE 5 SUMMARIZES THE INPUT PROCESS

- SHELL INPUT CONSISTS OF BLOCKS CORRESPONDING TO INPUT CATEGORIES
  - usually baseline parameters are obtained from a file
  - then namelist input is used to change the shell for this job and case
  - shell then constructs core input for typical configurations
- CORE INPUT CONSISTS OF BLOCKS CORRESPONDING TO SYSTEM PIECES
  - namelist input is used to change core constructed by shell
  - core input defines tables required
- TABLE INPUT CONSISTS OF BLOCKS CORRESPONDING TO TABLE FILES

EACH INPUT BLOCK IS IDENTIFIED BY CLASS, TYPE, AND NAME

- plus file name and unit number for tables

FIGURE 6 LISTS CORE, TABLE, AND SHELL INPUT BLOCKS

IN SHELL INPUT, ALL BLOCKS NEEDED MUST BE DEFINED BY USER

- even if only default parameter values are required
  - there is no automatic definition of shell blocks

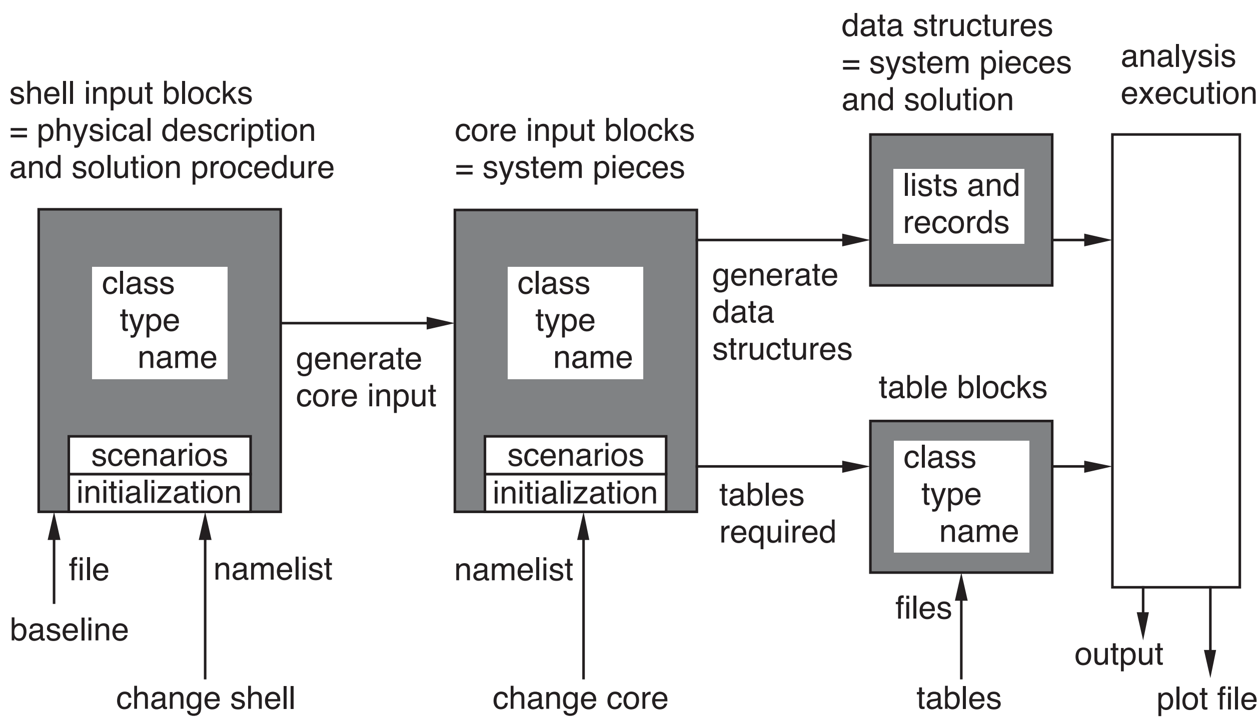


Figure 2-5 Summary of shell and core input process.

CLASS	TYPE	NAME
CASE	none	ignore
COMPONENT	type	
FRAME	none	
INTERFACE	STRUCTURAL DYNAMIC	
INTERFACE	INPUT/OUTPUT	
OUTPUT	none	
INPUT	none	
WIND	none	ignore
OPERATING CONDITION	none	
PERIOD	none	
TRIM LOOP	NO SOLUTION	
TRIM LOOP	SUCCESSIVE SUBSTITUTION	
TRIM LOOP	NEWTON RAPHSON	
TRIM LOOP	REGULATOR	
TRIM PART	NO SOLUTION	
TRIM PART	IMPLICIT	
TRIM PART	STATIC	
TRIM PART	HARMONIC	
TRIM PART	TIME FINITE ELEMENT	
TRANSIENT	none	ignore
TRANSIENT LOOP	NO SOLUTION	
TRANSIENT LOOP	SUCCESSIVE SUBSTITUTION	
TRANSIENT PART	NO SOLUTION	
TRANSIENT PART	TRIM SOLUTION	
TRANSIENT PART	IMPLICIT	
TRANSIENT PART	INTEGRATION	
FLUTTER	none	ignore
FLUTTER LOOP	none	
FLUTTER PART	NO SOLUTION	
FLUTTER PART	INTERFACE	
FLUTTER PART	DIFFERENTIAL EQUATIONS	
TRANSFORM	none	
MODES	none	
RESPONSE	RIGID	
RESPONSE	VARIABLE	
WEIGHTS	none	

If there is only one block, the name is ignored. Lower case is accepted for class and type. Unique initial letters of class or type are accepted. Specific abbreviations and alternatives:

- class = TRIM LOOP:
  - type = NO SOLUTION = NULL
  - type = NEWTON RAPHSON = NR
- class = TRIM PART, TRANSIENT LOOP, TRANSIENT PART:
  - type = NO SOLUTION = NULL
- class = FLUTTER PART:
  - type = NO SOLUTION = NULL
  - type = DIFFERENTIAL EQUATIONS = DE = EQUATIONS

Figure 2-6a Core input blocks.

CLASS	TYPE	NAME
COMPONENT	RIGID BODY	
COMPONENT	LINEAR NORMAL MODES	
COMPONENT	FINITE ELEMENT BEAM	
COMPONENT	ROD/CABLE	
COMPONENT	TRANSMISSION	
COMPONENT	REFERENCE FRAME	
COMPONENT	FILTER	
COMPONENT	REFERENCE PLANE	
COMPONENT	DIFFERENTIAL EQUATION	
COMPONENT	PROGRAMMABLE	
COMPONENT	TRANSFER FUNCTION	
COMPONENT	FOURIER SERIES	
COMPONENT	PRESCRIBED CONTROL	
COMPONENT	GUST	
COMPONENT	RIGID AIRFRAME AERODYNAMICS	
COMPONENT	AIRFRAME FLOW FIELD	
COMPONENT	LIFTING LINE WING	
COMPONENT	WING INFLOW	
COMPONENT	ROTOR INFLOW	
COMPONENT	ROTOR DYNAMIC WAKE	
COMPONENT	WING WAKE	
COMPONENT	WING WAKE GEOMETRY	
COMPONENT	ROTOR WAKE GEOMETRY	
COMPONENT	WING PERFORMANCE	
COMPONENT	ROTOR PERFORMANCE	
COMPONENT	ROTORCRAFT PERFORMANCE	
COMPONENT	RIGID WING	
COMPONENT	HELICOPTER TAIL BOOM	
COMPONENT	COMPUTATIONAL FLUID DYNAMICS	
COMPONENT	PLUGIN	

Lower case is accepted for class and type. Unique initial letters of class or type are accepted. Specific abbreviations and alternatives:

class = COMPONENT:

- type = LINEAR NORMAL MODES = NORMAL MODES = MODES
- type = FINITE ELEMENT BEAM = BEAM
- type = ROD/CABLE = ROD = CABLE
- type = FILTER = LOW PASS FILTER = RESOLVER = DIFFERENTIATOR
- type = DIFFERENTIAL EQUATION = DE = STATIC EQUATION = ADDER
- type = AIRFRAME FLOW FIELD = AIRFRAME INTERFERENCE VELOCITY
- type = ROTOR INFLOW = UNIFORM INFLOW
- type = WING WAKE = NONUNIFORM INFLOW
- type = COMPUTATIONAL FLUID DYNAMICS = CFD

Figure 2-6b Core component input blocks.

CLASS	TYPE	NAME
TWO DIMENSIONAL	STANDARD	
THREE DIMENSIONAL	STANDARD	
FOUR DIMENSIONAL	STANDARD	
AIRFOIL	STANDARD	
AIRFOIL	C81	
AIRFOIL	CAMRAD	
MATRIX	STANDARD	
UNSTRUCTURED	REAL	
UNSTRUCTURED	INTEGER	

Lower case is accepted for class and type. Unique initial letters of class or type are accepted. Specific abbreviations and alternatives:

class = `TWO DIMENSIONAL` = `2D`
class = `THREE DIMENSIONAL` = `3D`
class = `FOUR DIMENSIONAL` = `4D`

class = `AIRFOIL`:
  type = `C81` = `FIXED FORMAT`
  type = `CAMRAD` = `WING`

class = `MATRIX` = `ARRAY`

Figure 2-6c Table blocks.

CLASS	TYPE	NAME
CHANGE CORE	none	ignore
CASE	none	ignore
TRIM	none	ignore
TRIM ROTOR	none	“Rotor n”
TRANSIENT	none	ignore
TRANSIENT ROTOR	none	“Rotor n”
FLUTTER	none	ignore
FLUTTER ROTOR	none	“Rotor n”
AIRFRAME	STRUCTURE	ignore
AIRFRAME	AERODYNAMICS	ignore
AIRFRAME	CONTROL	ignore
AIRFRAME	DRIVE TRAIN	ignore
ROTOR	STRUCTURE	“Rotor n”
ROTOR	FLEXBEAM	“Rotor n”
ROTOR	AERODYNAMICS	“Rotor n”
ROTOR	INFLOW	“Rotor n”
ROTOR	WAKE	“Rotor n”
TABLES	none	ignore

The “Rotor n” name is defined by the parameter ROTOR in the class = AIRFRAME, type = STRUCTURE input (default = 'ROTOR 1', 'ROTOR 2', ..., 'ROTORkk').

If there is only one block, the name is ignored. Lower case is accepted for class and type. Unique initial letters of class or type are accepted. Specific abbreviations and alternatives:

class = ROTOR:

type = INFLOW = UNIFORM INFLOW

type = WAKE = NONUNIFORM INFLOW

Figure 2-6d Shell input blocks.

2–5 Programs

CAMRAD II MAIN PROGRAMS

ROTORCRAFT ANALYSIS (CAMRADII)
INPUT AND TABLE FILE PREPARATION (INPUT)
PLOT FILE EXAMINATION (OUTPUT)

INPUT DATA FOR CAMRADII PROGRAM

INPUT FILE, JOB NAMELISTS, TABLE FILES

INPUT FILE OPTIONAL, BUT RECOMMENDED

separates baseline and job input

provides configuration control of baseline input parameters

ROTORCRAFT SHELL USUALLY REQUIRES AIRFOIL TABLE FILE(S)

other tables may be needed as well

TYPICAL STEPS TO RUN CAMRAD II

CREATE AIRFOIL TABLE FILE

CREATE SHELL INPUT FILE

RUN ROTORCRAFT ANALYSIS

EXAMINE PLOT FILE

2–6 Input and Table File Preparation: Program INPUT

PROGRAM INPUT

INPUT FILE PREPARATION

PRODUCES UNFORMATTED SHELL INPUT FILE, FROM NAMELIST FILE(S)

can also extract data to plot or draw input, and examine initialized system data

TABLE FILE PREPARATION

PRODUCES UNFORMATTED OR FORMATTED FILES, FOR EACH TABLE CLASS AND TYPE

SPECIFICALLY, CREATES AIRFOIL TABLE FILE FOR ROTORCRAFT SHELL

unformatted `CAMRAD` airfoil file for entire blade from `STANDARD` (free form) or `C81` (fixed form) airfoil file(s)

can also extract data to plot table

BATCH OR INTERACTIVE OPERATION

FIRST LINE OF JOB INPUT STREAM MUST BE “BATCH” OR “INTERACTIVE”

default is interactive

FOR INTERACTIVE MODE, PRINTED OUTPUT CAN BE DIRECTED TO LOG FILE (OUTPUTLOG)

THEN PROGRAM READS NAMELIST NLJOB, CONTAINING PARAMETERS DEFINING THE JOB

NAMELIST NLJOB

AFTER SPECIFYING BATCH OPERATION, PROGRAM READS NAMELIST NLJOB, CONTAINING PARAMETERS DEFINING THE JOB

all parameters have defaults, which are set (and may be changed) in main program

TABLE FILE PREPARATION REQUIRES ADDITIONAL NAMELISTS

2-6.1 Shell Input File Preparation

REFERENCE: CAMRAD II Documentation, Volume VI, Rotorcraft Input; Chapter “Shell Input File Preparation”

figure 7 outlines process
figure 8 shows form and example of batch job

SAMPLE JOBS PROVIDE EXAMPLES

FILES THAT MAY BE REQUIRED

file	name or logical name	control parameter
input file	INPUTFILE	NFILEI
namelist file	INPUTLISTn	NLISTI
input file	OUTPUTFILE	NFILEO
namelist file	OUTPUTLIST	NLISTO

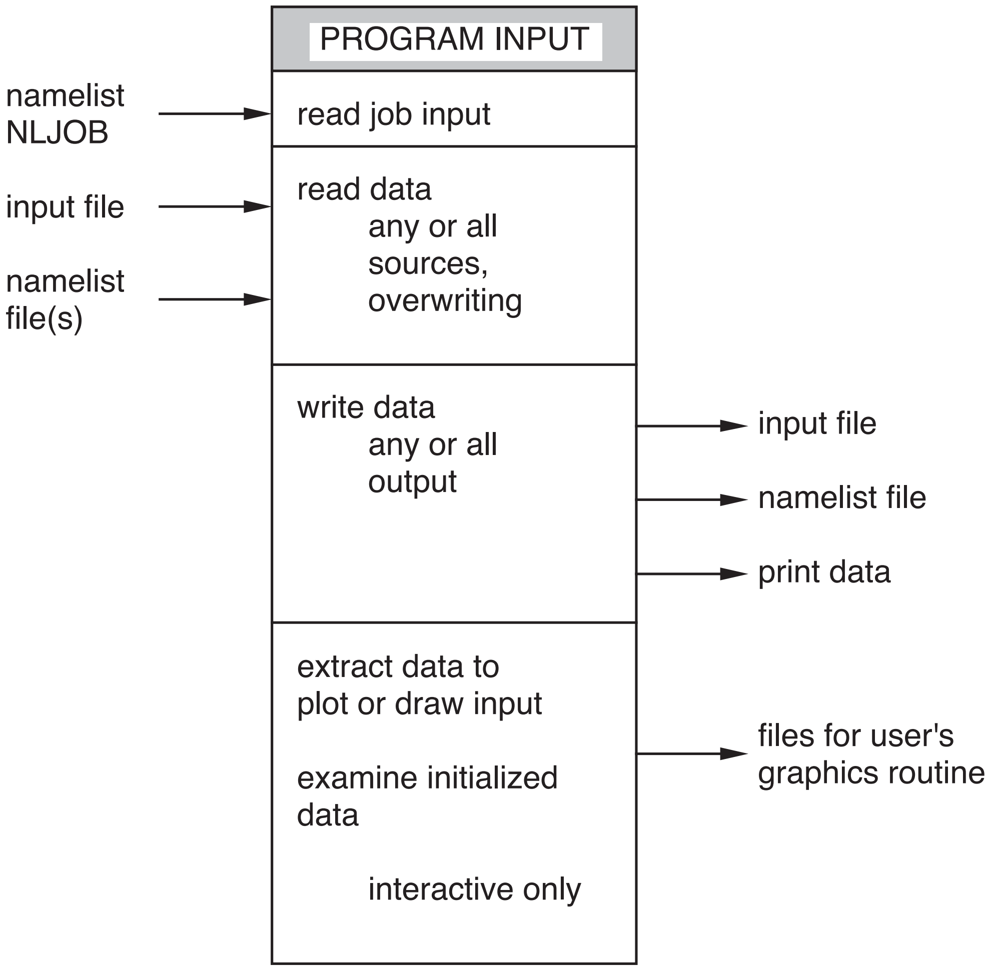


Figure 2-7 Outline of INPUT process — input file preparation.

BATCH JOB

define files required by job
run INPUT
BATCH
namelist NLJOB (contains parameters defining job)

EXAMPLE (unix)

```
setenv INPUTLIST1 helicopter.list
setenv OUTPUTFILE helicopter.dat
/camrad/input > helicopter.out << 'endofinput'
 BATCH
 &NLJOB NLISTI=1,NLISTC=0,&END
'endofinput'
```

Figure 2-8 Form and example of INPUT job — input file preparation.

TYPICAL JOB TO PREPARE SHELL INPUT FILE

CREATE UNFORMATTED SHELL INPUT FILE, FROM NAMELIST FILE(S) OF BASELINE PARAMETERS; AND PRINT INPUT PARAMETERS

FILES REQUIRED BY JOB:

INPUTLIST1 to INPUTLISTn, and OUTPUTFILE

file for printed output

NAMELIST NLJOB PARAMETERS:

NLISTI $= n$, number of namelist files

NLISTC = first file containing core changes to shell input

with no core changes, use NLISTC = 0 (default)

with core changes in only one file, make it the last file and use NLISTC = NLISTI (so the shell needs to create the core input only once)

default values generally used for other parameters

INTERACTIVE MODE OPERATION

after reading input data, INPUT program can extract the data and write it to new files, which can then be read by user's graphics routines

EXTRACT DATA IN ORDER TO PLOT OR DRAW SHELL INPUT

blade parameters and geometry (structure and aerodynamic information that depends on blade radius)

PARAMETERS ARE INPUT VARIABLES, IDENTIFIED BY NAME AS DEFINED IN INPUT MANUAL

structural parameters: at input radial stations or at Gaussian integration points

aerodynamic parameters: at input radial stations, at aerodynamic panel edges, or at panel mid points

Gaussian points and panel mid points are where input data are actually used

GEOMETRY CONSISTS OF CHORDWISE AND NORMAL OFFSETS, USED TO DRAW PLANFORM

offsets relative blade span axis, reference line, or element beam axis; normally use offset relative blade span axis (total offset)

normally twist suppressed, otherwise twist makes blade appear to be tapered

structural geometry includes elastic axis at nodes, can be compared with input elastic axis data

aerodynamic geometry includes quarter chord, leading edge, and trailing edge

script `zplanform.com` available to run INPUT program to extract all information needed to draw planform

EXTRACT DATA IN ORDER TO PLOT OR DRAW CORE INPUT, OR EXAMINE INITIALIZATION

before creating core input, may override shell parameters

flutter task may be required to create all wings (normal trim task solves for motion of just one blade, so aerodynamic components created only for that blade unless flutter or transient task is executed or initialized)

PLOT INPUT DATA

plot input as function of radius for wings and for beams, similar to plot function for shell input

EXAMINE INITIALIZED DATA

for specified system pieces; this information can also be produced by analysis program (NPRNTD)

DRAW GEOMETRY

three-dimensional geometry extracted, typically for “all structural dynamic components” and “lifting line wings”

default for structural dynamic components: draw position as line from a reference line (such as the beam axis), for all structural dynamic interfaces

can also draw orientation, as axis system at interface

for most structural dynamic components, if there are more than two locations then user must identify two locations as end points of reference line

for wings, default option is recommended

2-6.2 Table File Preparation

REFERENCE: CAMRAD II Documentation, Volume VI, Rotorcraft Input; Chapter “Table File Preparation”

figure 9 outlines process
figure 10 shows form and example of batch job

SAMPLE JOBS PROVIDE EXAMPLES

FILES THAT MAY BE REQUIRED

file	name or logical name	control parameter
source file	INPUTDECK (1)	OPSRC = 1
table file	INPUTTABLE	OPSRC = 3
table file	OUTPUTTABLE	NFILEO

(1) INPUTDECKn or INPUTDECKnFLAPm for CAMRAD airfoil table

INPUT PROGRAM CAN PREPARE CLASS = AIRFOIL FROM SOURCE FILES OR INTERNALLY

for other table classes, table file preparation does not transform the table data

ADDITIONAL NAMELISTS, AFTER NLJOB

NAMELIST NLTABL
FOR TABLE CLASS = AIRFOIL
contains table parameters

NAMELIST NLEQN
FOR TABLE CLASS = AIRFOIL,
WHEN TABLE CALCULATED INTERNALLY
contains table calculation variables

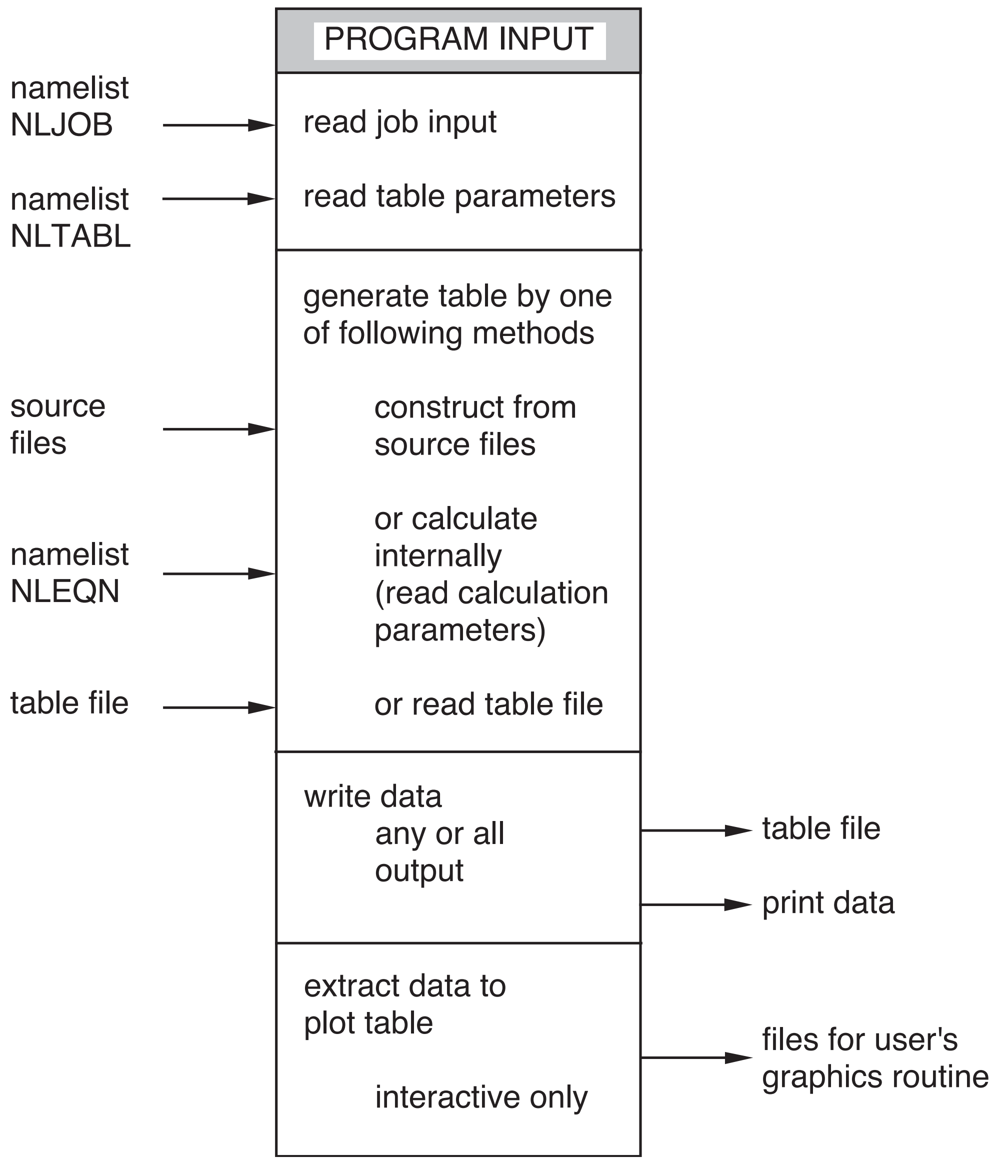


Figure 2-9 Outline of INPUT process — table file preparation.

BATCH JOB

define files required by job
run INPUT
BATCH
namelist NLJOB (contains parameters defining job)
namelist NLTABL (contains table parameters)
namelist NLzz (table calculation parameters)

EXAMPLE (unix)

```
setenv INPUTDECK1 airfoil.c81
setenv OUTPUTFILE airfoil.tab
/camrad/input > airfoil.out << 'endofinput'
 BATCH
 &NLJOB OPFILE=7,OPSRC=1,&END
 &NLTABL OPFORM=2,RNTRP=1,TITLE='AIRFOIL',&END
'endofinput'
```

Figure 2-10 Form and example of INPUT job — table file preparation.

TYPICAL JOB TO PREPARE AIRFOIL TABLE FILE, FROM AIRFOIL DECKS

create `CAMRAD` airfoil table file (unformatted file for entire blade),
from `STANDARD` (free form) or
`C81` (fixed form) airfoil table file(s)

and print table parameters (not table data)

FILES REQUIRED BY JOB:

INPUTDECK1 (root) to INPUTDECKn (tip), and OUTPUTTABLE

file for printed output

NAMELIST NLJOB PARAMETERS:

OPFILE = 7, to produce `CAMRAD` airfoil table
OPSRC = 1, from source files (airfoil decks)

airfoil decks should include Reynolds number

default values generally used for other parameters

NAMELIST NLTABL PARAMETERS:

OPFORM = 1, from airfoil deck in STANDARD form
OPFORM = 2, from airfoil deck in C81 form

RNTRP = 0, spanwise interpolation of airfoil decks
RNTRP = 1, spanwise search of airfoil decks

only required if NRB > 1; spanwise interpolation must be used with care

TITLE = title for wing data

NRB $= n$, number of span stations
R = span station values (root to tip)

only required if NRB > 1; NRB is number of airfoil decks; for spanwise search R is inboard edge of airfoil deck range

OPFLAP = trailing edge flap
ND(NRB) = number of flap angles at span station

default values generally used for other parameters

INTERACTIVE MODE OPERATION

after producing the table, INPUT program can extract the data and write it to new files, which can then be read by user's graphics routines

TWO-DIMENSIONAL, THREE-DIMENSIONAL, FOUR-DIMENSIONAL TABLES

data extracted as function of one independent variable, for selected values of second independent variable and one value of remaining variables

AIRFOIL TABLES

EXTRACT DATA IN ORDER TO PLOT TABLE, PERHAPS WITH STALL DELAY FACTOR

$c_\ell(\alpha)$, $c_d(\alpha)$, $c_m(\alpha)$ for selected M
$c_\ell(M)$, $c_d(M)$, $c_m(M)$ for selected α
$c_d(c_\ell)$, $c_m(c_\ell)$ for one M

IDENTIFY DYNAMIC STALL PARAMETERS OR AIRFOIL CHARACTERISTICS FROM AIRFOIL TABLE DATA

dynamic stall parameters define trailing edge separation point function, for Leishman-Beddoes or ONERA BH model

airfoil characteristics: zero lift angle α_z; drag and moment at zero lift, $c_{dz} = c_d(\alpha_z)$ and $c_{mz} = c_m(\alpha_z)$; lift-curve slope and moment slope, $c_{\ell\alpha}$ and $c_{m\alpha}$; aerodynamic center shift $c_{m\alpha}/c_{\ell\alpha}$; drag recovery factor η

IDENTIFY SEPARATION POINT PARAMETERS AND AIRFOIL CHARACTERISTICS

at selected Mach number

and produce plot of lift coefficient and separation point function as function of angle of attack, from table data and from identified parameters

IDENTIFY CRITICAL LIFT COEFFICIENT (at all Mach numbers)

PRODUCE TWO-DIMENSIONAL TABLE FILE

dynamic stall parameters as function of Mach number and span station, to be pasted into dynamic stall file

PLOT AIRFOIL CHARACTERISTICS

one characteristic or parameter, as function of Mach number

CHANGE IDENTIFICATION OPTIONS

2–7 Analysis: Program CAMRADII

PROGRAM CAMRADII ANALYZES THE AEROMECHANICAL SYSTEM

REFERENCE: CAMRAD II Documentation, Volume VI, Rotorcraft Input; Chapter “CAMRAD II Analysis”

figure 11 outlines process
figure 12 shows form and example of batch job

SAMPLE JOBS PROVIDE EXAMPLES

FILES THAT MAY BE REQUIRED

file	name or logical name	control param
input data file	SHELLINPUT	OPSHLL
airfoil table files	BLADEAIRFOILn (3)	OPAERO
other table files	(2)	shell input
plot data file	PLOTFILE	PLFILE
trim initialization input	TRIMSOLUTIONINm (1)	OPITJR
trim initialization output	TRIMSOLUTIONOUTm (1)	OPITJW
transient restart input	TRANRESTARTIN	OPRNJR
transient restart output	TRANRESTARTOUT	OPRNJW

(1) m is case number (1 if only one file is read)
(2) table names are defined in shell input class = TABLES
(3) n is the rotor number

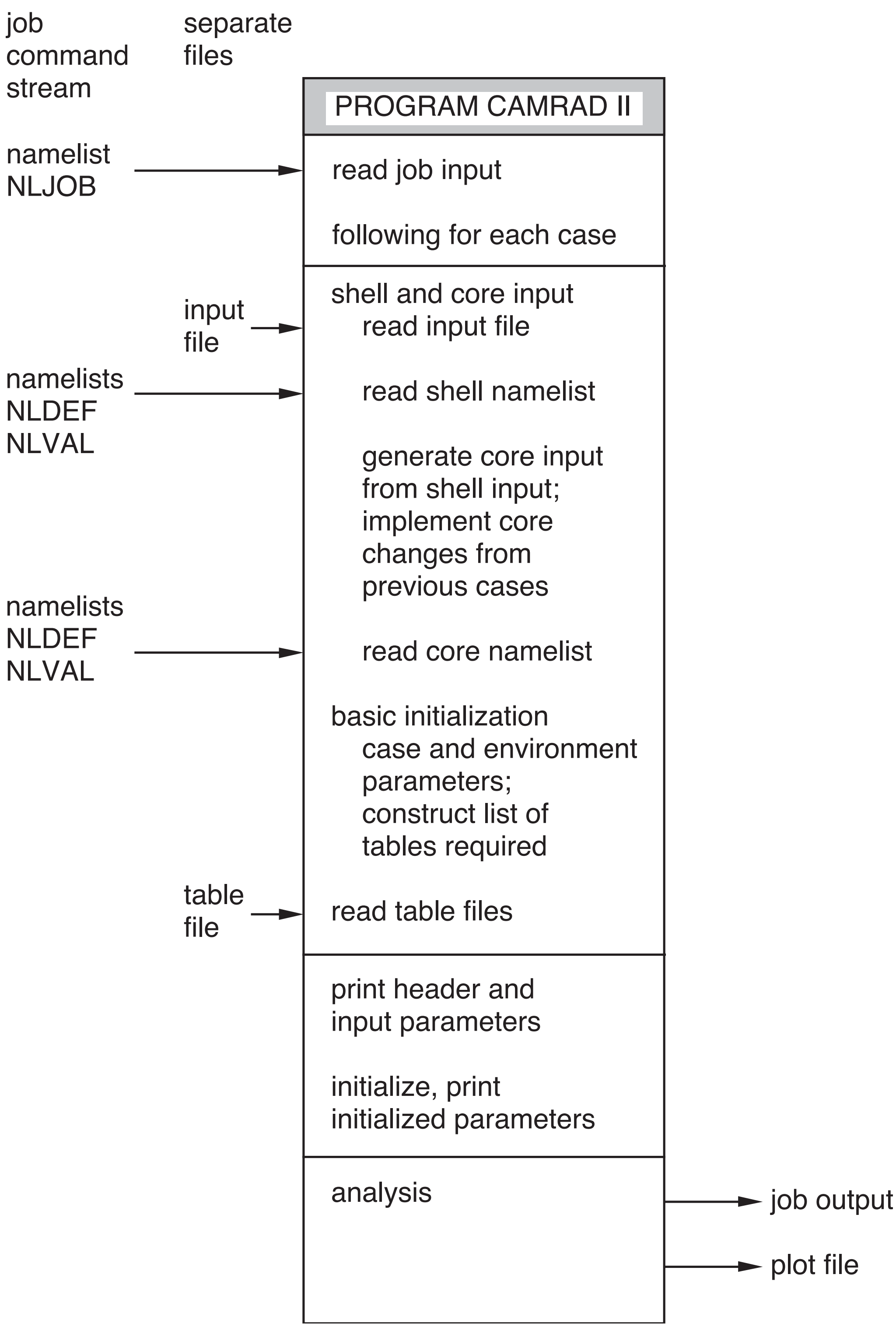


Figure 2-11 Outline of CAMRADII process.

BATCH JOB

- define input files required by job
  - shell input file SHELLINPUT (contains baseline parameters)
  - airfoil file(s)
  - other table files
- define output files required by job
  - plot data file PLOTFILE
- run CAMRADII
- namelist NLJOB (contains parameters defining job)
- shell and core namelist for each case (NLDEF and NLVAL)

EXAMPLE (unix)

```
setenv SHELLINPUT helicopter.dat
setenv BLADEAIRFOIL1 airfoil.tab
setenv PLOTFILE sample.plot
/camrad/camradii > sample.out << 'endofinput'
 &NLJOB PLFILE=1,&END
 &NLDEF class='TRIM',&END
 &NLVAL VKTS=100.,&END
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

Figure 2-12 Form and example of CAMRADII job.

MULTIPLE CASES

SHELL INPUT FILE TYPICALLY READ ONLY FOR FIRST CASE

SO FOLLOWING CASES INHERIT ALL PREVIOUS INPUT

including core namelist input that changes core data constructed by shell

SECOND AND SUBSEQUENT CASES CAN BE INITIALIZED FROM TRIM SOLUTION OF PREVIOUS CASE

USING THE TRIM LOOP VARIABLES (OPINIT = 1)

derivative matrices, control, wind, operating condition, and period parameters involved in any loop of trim task

variables VNAME identified by Newton-Raphson trim loop

OR THE TRIM PART TOTAL SOLUTION (OPINIT = 6)

solution of rotor, airframe, and drive train parts, plus all interfaces

OR BOTH (OPINIT = 7)

CASE CAN ALSO BE INITIALIZED BY READING FILE OF THIS TRIM SOLUTION, WRITTEN BY A PREVIOUS JOB

figure 13 illustrates initialization of trim solution

if the system configuration changes from case to case, then names of system pieces must be appropriately changed as well; otherwise this initialization procedure will not be able to correctly match response records of two cases

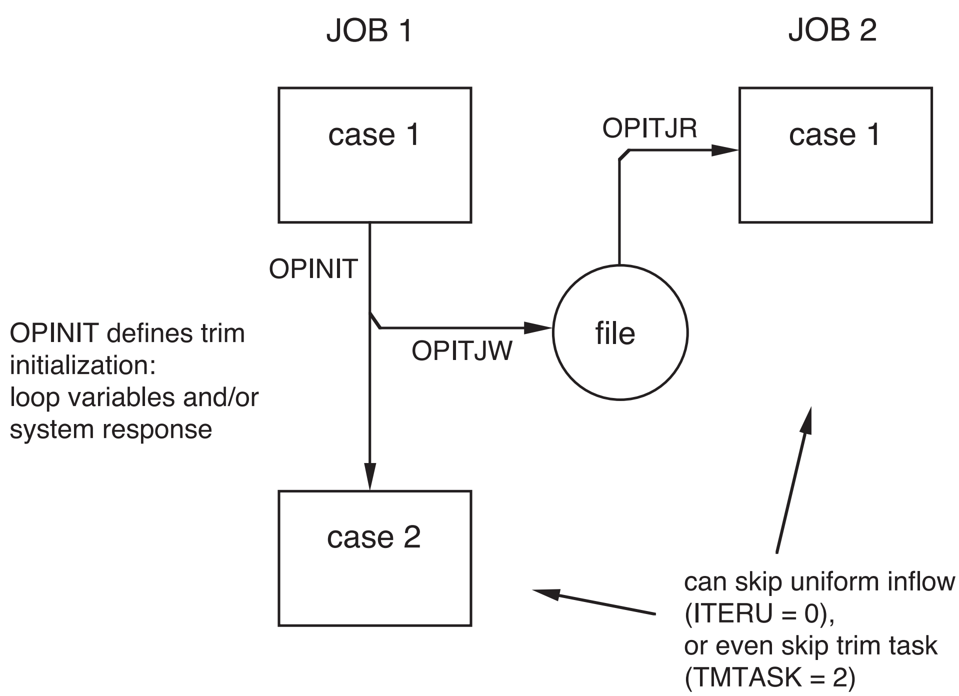


Figure 2-13 Initialization of trim solution from previous case or job.

TYPICAL JOB TO ANALYZE ROTORCRAFT

READ SHELL INPUT FILE AND AIRFOIL TABLE FILES;
READ NAMELIST INPUT FOR EACH CASE;
ANALYZE ROTORCRAFT;
PRINT JOB OUTPUT AND CREATE PLOT FILE

FILES REQUIRED BY JOB

SHELLINPUT, BLADEAIRFOIL1, BLADEAIRFOIL2, and PLOTFILE

airfoil tables are not required if rotor blade structure specifies no aerodynamics, OPAERO = 0

number of airfoil tables is specified in shell input class = TABLES

AFTABL = 0, same table for all rotors (default)

AFTABL = 1, separate tables (number of tables = number of rotors)

file for printed output

PROGRAM FIRST READS NAMELIST NLJOB, CONTAINING PARAMETERS DEFINING THE JOB

all parameters have defaults, which are set (and may be changed) in main program

PROGRAM THEN READS SHELL AND CORE NAMELIST FOR EACH CASE

NLDEF and NLVAL pairs

NAMELIST NLJOB PARAMETERS:

NCASES = number of cases
PLFILE = 1 to write plot file for job
OPSHLL = shell input file read

defaults: one case, no plot file written, shell input file read for first case only (OPSHLL = 1)

OPINIT = initialization from previous case

trim solution can be initialized from previous case using:

the trim loop variables (OPINIT = 1) (derivative matrices, controls, and other variables used by Newton-Raphson trim loop)

the trim part total solution (OPINIT = 6) (response of all degrees of freedom and interfaces)

or both (OPINIT = 7)

default is no initialization

default values generally used for other parameters

2–8 Plot File Examination: Program OUTPUT

SELECTED OUTPUT CAN BE DIRECTED TO PLOT DATA FILE

- PLOT FILE INCLUDES ONLY ENOUGH HEADER INFORMATION AND TITLES TO IDENTIFY ITS DATA
  - reference must be made to printed output for complete description of quantities
- PLOT FILE CAN BE READ USING OUTPUT PROGRAM

PROGRAM OUTPUT

- READS AND EXTRACTS DATA FROM PLOT FILE
  - create files for user's graphics routine
- PRINT (TO A LOG FILE) LINEAR DIFFERENTIAL EQUATIONS OF FLUTTER ANALYSIS
- figure 14 outlines process

GRAPHICS DATA REQUIRED TO PRODUCE ANIMATION OF TRIM OR TRANSIENT SOLUTION IS SENT TO PLOT FILE, AND READ BY OUTPUT PROGRAM

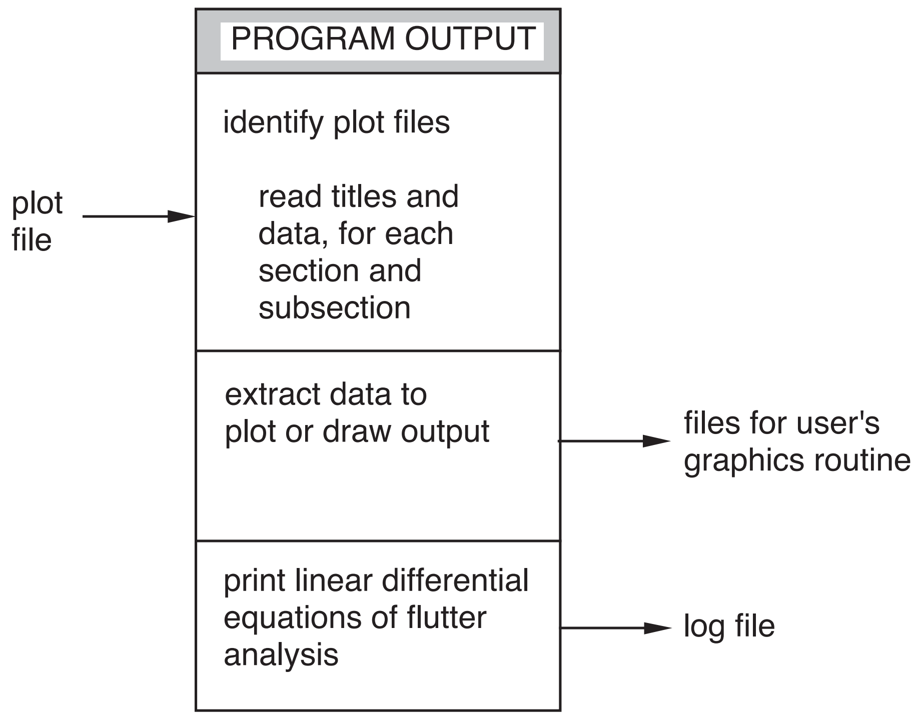


Figure 2-14 Outline of `OUTPUT` process (interactive only).

INTERACTIVE OPERATION, NOT BATCH

often run using script files

plot file data considered by section, with following operations

LIST THE SECTIONS

move down or up sections while displaying titles

perhaps jump down to next status line, next case, or next graphics section

default (carriage return) is to move down to next section

IDENTIFY A SECTION (from a compact list)

FIND A TITLE (search for a string in the title)

FIND A TITLE AGAIN

CHANGE LIST OPTIONS

EXAMINE THE CURRENT SECTION

extract data to file for graphics routines

quantities that can be extracted:
time history data ($F(\psi)$ or $F(t)$);
wing sensors ($F(r,\psi)$ for selected r or selected ψ)
wake geometry data (three-dimensional geometry)

UTILITY SUBROUTINES FILEXP, FILEXA, AND FILEXD WRITE EXTRACTED DATA TO FILE

USED BY INPUT AND OUTPUT PROGRAMS

CAN BE MODIFIED AS APPROPRIATE FOR USER'S GRAPHICS ROUTINES

UTILITY FILEXP HANDLES PLOTABLE-DATA
DATA CONSISTS OF $y_1(x)$, $\ldots$, $y_n(x)$ FOR SET OF POINTS x_i

figure 15 shows format of file produced

maximum 40 characters in code
maximum 12 characters in labels
blanks removed from labels

UTILITY FILEXD HANDLES DRAWABLE-DATA
DATA CONSISTS OF (x, y, z) COORDINATES FOR SET OF POINTS

figure 16 shows format of file produced

additional legends can be present in data section
maximum 40 characters in code
maximum 80 characters in legend

EXPORT FORMATS
CAMRAD standard, space delimiter
CAMRAD standard, tab delimiter
CAMRAD standard, comma delimiter
Tecplot
DXF (draw only)

DEFAULT EXPORT FORMAT CAN BE CHANGED BY REVISING MAIN PROGRAMS

```
header lines
number of dependent variables = N, number of points = NNNN
CODE
code
LABELS
xlabel    y1label    . . .    ynlabel
DATA
xvalue    y1value    . . .    ynvalue
. . . .
. . . .
xvalue    y1value    . . .    ynvalue
```

Figure 2-15 Format of file produced by FILEXP.

```
header lines
CODE
code
LEGEND
legend
DATA
xvalue   yvalue   zvalue   penaction
. . . .
. . . .
xvalue   yvalue   zvalue   penaction
```

pen action (fourth field) can have the following values:

`NEW` = start new line (move to point with pen up)

`NEWn` = start new line, and change to line type n = 1–4

blank = continue old line (move to point with pen down)

Figure 2-16 Format of file produced by FILEXD.

2–9 Graphics and Animation

A GRAPHICAL IMAGE OF THE TRIM OR TRANSIENT SOLUTION, EITHER A STILL PICTURE OR AN ANIMATION, CAN BE PRODUCED

GRAPHICS DATA REQUIRED CAN BE SENT TO PLOT FILE, AND READ BY OUTPUT PROGRAM

OUTPUT PROGRAM GENERATES TWO INPUT FILES FOR MATHEMATICA (COMMAND FILE AND GEOMETRY FILE) OR
ONE INPUT FILE FOR TECPLOT OR
ONE INPUT FILE FOR MATLAB

MATHEMATICA OR TECPLOT OR MATLAB PRODUCES THE ANIMATION

typically OUTPUT program run using script file, to facilitate changes

INPUT program used to extract geometry of system rest position, not the solution

OBJECTS THAT CAN BE DRAWN INCLUDE STRUCTURE GEOMETRY, STRUCTURAL SENSORS, WINGS, WING SENSORS, AND WAKE GEOMETRY

structure geometry drawn as lines, or with appended shapes

wings drawn as surfaces, or with defined airfoil shapes

shapes of structure and wings are defined when OUTPUT program is run, not part of the CAMRAD II input

structural sensors drawn as arrows representing force or moment acting at a point

wing sensors display section aerodynamic quantities along wing span

MATHEMATICA INPUT CONSISTS OF A COMMAND FILE AND A GEOMETRY FILE

user should read comments in command file (either using Mathematica, or reading it as a text file)

ANIMATION PRODUCED BY OPENING MATHEMATICA NOTEBOOK

execute commands to set current working directory to location of the two input files, read the geometry file, and read the command file

then cells are executed by selecting them in order they appear (contents of most of cells are hidden), and pressing “enter”

PALETTE CREATED CONTAINING BUTTONS THAT CAN BE USED TO CONTROL DISPLAY

change orientation, size, and viewpoint

IMAGE FRAMES ARE CREATED FOR NUMBER OF AZIMUTH OR TIME STEPS, AND FROM THEM AN ANIMATION CAN BE CREATED

MATLAB INPUT FILE CONTAINS COMMAND AND GEOMETRY DATA

user should read comments in file

ANIMATION PRODUCED BY OPENING MATLAB AND LOADING FILE (type *.m)

select its folder, and execute this file to create the images and movie (mp4)

adjust the image, revise the position and orientation values, and execute the file again

IMAGE FRAMES ARE CREATED FOR NUMBER OF AZIMUTH OR TIME STEPS, AND FROM THEM THE ANIMATION IS CREATED

TECPLOT INPUT FILE CONTAINS GEOMETRIC DATA FOR A SINGLE TIME STEP

open Tecplot and load this data file using “3D Cartesian” plot type

activate the “Contour” zone layer to display surfaces

the “Mesh” zone layer displays lines and polygon edges

color specified directly for lines

for surfaces (polygons), color defined in terms of contour values

ANIMATION PRODUCED BY WRITING SCRIPT TO RUN OUTPUT PROGRAM AND PRODUCE TECPLOT INPUT FILE FOR EACH TIME STEP

then Tecplot macro can be used to load the files and create the animation

2–10 Data Vectors and Memory

A DATA STRUCTURE IS A SET OF REAL, INTEGER, AND CHARACTER VARIABLES

STRUCTURE TYPES:

- DATA VECTORS
- RECORDS
- LISTS
- COMMONS

DATA VECTORS

PROGRAM STORES MOST GLOBAL DATA IN DATA-VECTOR-TYPE DATA STRUCTURE (A “HEAP”)

DATA VECTORS OF CAMRAD II:

- core input
- table input
- shell input
- principal data vector of analysis

DATA VECTORS IMPLEMENTED AS COMMONS

- separate commons used for real, integer, and character variables

SO MEMORY AVAILABLE TO ANALYSIS IS FIXED WHEN PROGRAM EXECUTABLE IS CREATED

STORAGE IN DATA VECTOR

- DATA VECTOR IS PARTITIONED INTO SECTIONS
  - analysis creates sections as required
  - sections stored in data vector starting from beginning
- FREE SPACE
  - free space exists in data vector from last section to end
  - there may also be free space between sections, if sections have been moved
  - usage of space in data vectors is printed at end of each case, and this information can be saved in statistics file
- NEW SECTIONS
  - added after last section in data vector
  - if there is not enough free space after last section, vector is packed
    - if after the pack there still is not enough free space, then analysis halts
    - with the following error message: "SPACE REQUIRED GREATER THAN FREE SPACE"

WHEN ANALYSIS RUNS OUT OF FREE SPACE IN A DATA VECTOR:

- EITHER CHANGE PARAMETERS TO CREATE SMALLER PROBLEM
- OR INCREASE SIZE OF DATA VECTOR
  - revise definition of data vector size in main programs, following instructions given with error message
- OR USE DYNAMIC MEMORY EXECUTABLE
  - assign data vector sizes at run time

2–11 Sample Jobs

PURPOSE

CHECK INSTALLATION
DEMONSTRATE USE OF ANALYSIS
PROVIDE STARTING POINT FOR PROJECTS

SHELL INPUT FILE PREPARATION

NAMELIST FILES
COMMAND FILES
RESULTING OUTPUT
elementary rotor, rotor in wind tunnel, teetering rotor, helicopter, bearingless rotor, tandem helicopter, tiltrotor in wind tunnel, tiltrotor aircraft

AIRFOIL TABLE FILE PREPARATION

AIRFOIL DECK
COMMAND FILES
RESULTING OUTPUT
simulated advanced technology airfoils (from airfoil equations)
NACA 0012 airfoil (from standard C81 table)

SAMPLE DYNAMIC STALL TABLE

ROTORCRAFT ANALYSIS

SAMPLE JOBS

following table summarizes configuration, operating condition, aerodynamic model, dynamic model

SEE LISTINGS OF SAMPLE JOBS, FOR SPECIFIC INSTALLATIONS

case	configuration	characteristics
1	elementary rotor in wind tunnel	hover, no trim
		dynamic inflow
		rigid blade
		eigenvalues, frequency response
		flutter and transient time history
2	teetering rotor in wind tunnel	hover
		hover wake (free geometry)
		rigid blade
		periodic eigenvalues (Floquet theory)
3	rotor in wind tunnel	forward flight
		low speed wake (free geometry)
		rigid blade
4	swept tip rotor in wind tunnel	forward flight, no trim
		forward flight wake (rigid geometry)
		airframe-to-rotor aerodynamic interference
		elastic blade, swashplate, elastic airframe
		dynamic stall; eigenvalues
5	swept tip rotor in wind tunnel	hover, no trim
		uniform inflow
		elastic blade, swashplate, elastic airframe
		eigenvalues
6	main-rotor and tail-rotor helicopter	hover
		uniform inflow
		rigid blade, drive train
		flight dynamics
		flutter and transient time history
7	main-rotor and tail-rotor helicopter	forward flight
		high speed wake (rigid geometry)
		elastic blade, swashplate
		vibration
8	bearingless rotor in wind tunnel	hover, no trim
		uniform inflow
		elastic blade, swashplate
		eigenvalues

case	configuration	characteristics
9	tandem helicopter	forward flight forward flight wake (rigid geometry) rotor-to-rotor aerodynamic interference rigid blade
10	tiltrotor in wind tunnel	axial flow uniform inflow elastic blade, swashplate, gimballed hub elastic airframe, drive train eigenvalues
11	tiltrotor aircraft	forward flight uniform inflow airframe-to-rotor aerodynamic interference rigid blade, swashplate, gimballed hub elastic airframe, drive train vibration, eigenvalues

OTHER RESOURCES

FILES AVAILABLE TO HELP PREPARE INPUT (CAMRAD II EXTRAS FOLDER)

TYPICAL INPUT

- conversion of CAMRAD/JA input (`ztemplate.list`)
- hover analysis (`zhover.list`)
- simplified models (`zsimple.list`)
- windtunnel case (`zwindtunnel.list`)
- blade frequency calculation (`zfrequency.list`)
- script to draw planform (`zplanform.com`)

EXAMPLES

- external aeroacoustic analysis, using post-trim
- demonstration of graphics generation
- generic programs to extract data from output file
- nonrotating lifting-line wing
- frozen wake geometry for second case or job
- generic wind turbine
- individual blade control
- interblade damper
- dual load path lag damper
- demonstration of trailing-edge flap
- elastic beams for flap structure
- tip-path plane sensors
- autopilot for transient task
- prescribed airframe motion for transient task
- revised trim variables or matrix

DOCUMENTS DEMONSTRATING USE OF CORE INPUT

- construction of elastic cantilever wing for tiltrotor
- bearingless rotor analysis
- closed-loop higher-harmonic control
- pendulum absorber

sample jobs and extras folder can be downloaded by customers from CAMRAD II web site

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'

#$!  simulated advanced technology airfoils for sample jobs
#$!
setenv OUTPUTTABLE $airfoil/advtecheq.tab
$camrad/input > $airfoil/advtecheq.out << 'endofinput'
 BATCH
 &NLJOB OPFILE=7,OPSRC=2,&END
 &NLTABL RNTRP=1,
     TITLE='SIMULATED ADVANCED TECHNOLOGY AIRFOILS (CONSTRUCTED DATA)',
     NRB=2,R=0.,.85,
 &END
 &NLEQN
 !  11 percent thick airfoil for inboard sections
     CLMAX=1.55,ALPHAD=15.5,MCRIT=.800,MDIV=.675,DEL0=.0070,CMAC=-.005,
     RETBL1=5.E6,
 &END
 &NLEQN
 !   8 percent thick airfoil for tip
     CLMAX=1.20,ALPHAD=12.0,MCRIT=.835,MDIV=.710,DEL0=.0065,CMAC=-.005,
     RETBL1=5.E6,
 &END
'endofinput'


#$!  NACA 0012 airfoil, standard C81 table
#$!
setenv INPUTDECK1 $airfoil/naca0012.c81
setenv OUTPUTTABLE $airfoil/naca0012.tab
$camrad/input > $airfoil/naca0012.out << 'endofinput'
 BATCH
 &NLJOB OPFILE=7,OPSRC=1,&END
 &NLTABL OPFORM=2,RNTRP=1,
     TITLE='NACA 0012 AIRFOIL (STANDARD C81 TABLE)',
 &END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'

#$!  sample jobs:  elementary rotor
#$!
setenv INPUTLIST1 $input/elementary.list
setenv OUTPUTFILE $input/elementary.bin
$camrad/input > $input/elementary.out << 'endofinput'
 BATCH
 &NLJOB NLISTI=1,NLISTC=0,&END
'endofinput'


#$!  sample jobs:  rotor in wind tunnel
#$!
setenv INPUTLIST1 $input/windtunnel.list
setenv OUTPUTFILE $input/windtunnel.bin
$camrad/input > $input/windtunnel.out << 'endofinput'
 BATCH
 &NLJOB NLISTI=1,NLISTC=1,&END
'endofinput'


#$!  sample jobs:  teetering rotor
#$!
setenv INPUTLIST1 $input/teeter.list
setenv OUTPUTFILE $input/teeter.bin
$camrad/input > $input/teeter.out << 'endofinput'
 BATCH
 &NLJOB NLISTI=1,NLISTC=0,&END
'endofinput'


#$!  sample jobs:  single main rotor and tail rotor helicopter
#$!
setenv INPUTLIST1 $input/helicopter.list
setenv OUTPUTFILE $input/helicopter.bin
$camrad/input > $input/helicopter.out << 'endofinput'
 BATCH
 &NLJOB NLISTI=1,NLISTC=0,&END
'endofinput'


#$!  sample jobs:  bearingless rotor
#$!
setenv INPUTLIST1 $input/bearingless.list
setenv OUTPUTFILE $input/bearingless.bin
$camrad/input > $input/bearingless.out << 'endofinput'
 BATCH
 &NLJOB NLISTI=1,NLISTC=0,&END
'endofinput'
```

```
#$!  sample jobs:  tandem helicopter
#$!
setenv INPUTLIST1 $input/tandema.list
setenv INPUTLIST2 $input/tandem1.list
setenv INPUTLIST3 $input/tandem2.list
setenv OUTPUTFILE $input/tandem.bin
$camrad/input > $input/tandem.out << 'endofinput'
 BATCH
 &NLJOB NLISTI=3,NLISTC=0,&END
'endofinput'
```

```
#$!  sample jobs:  tiltrotor in wind tunnel
#$!
setenv INPUTLIST1 $input/tiltrotorw.list
setenv INPUTLIST2 $input/tiltrotor1.list
setenv OUTPUTFILE $input/tiltrotorwt.bin
$camrad/input > $input/tiltrotorwt.out << 'endofinput'
 BATCH
 &NLJOB NLISTI=2,NLISTC=0,&END
'endofinput'
```

```
#$!  sample jobs:  tiltrotor aircraft
#$!
setenv INPUTLIST1 $input/tiltrotora.list
setenv INPUTLIST2 $input/tiltrotor1.list
setenv INPUTLIST3 $input/tiltrotor2.list
setenv OUTPUTFILE $input/tiltrotor.bin
$camrad/input > $input/tiltrotor.out << 'endofinput'
 BATCH
 &NLJOB NLISTI=3,NLISTC=0,&END
'endofinput'
```

```
!  Elementary Rotor
!==============================================================================
&NLDEF class='CASE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  ELEMENTARY ROTOR MODEL',        ! description
&END
!==============================================================================
&NLDEF class='TRIM',&END
&NLVAL OPSCEN=1,
    WINDIN=2,VTIPIN=1,VTIP=650.,                         ! operating condition
    LEVEL=1,                                             ! wake loop
    OPTRIM=0,COLL=10.,                                   ! no trim
    MHARMR=1,                                            ! harmonics
&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL OPSCEN=1,&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL OPSCEN=0,MHSEN=0,MPSEN=0,&END                     ! output
!==============================================================================
&NLDEF class='TRANSIENT',&END
&NLVAL OPSCEN=1,&END
&NLDEF class='TRANSIENT ROTOR',name='ROTOR 1',&END
&NLVAL OPSCEN=1,&END
!==============================================================================
&NLDEF class='FLUTTER',&END
&NLVAL OPSCEN=1,
    DOFA=6*0,                                            ! degrees of freedom
&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL OPSCEN=1,
    OPWAKE=3,DOFL=3*1,                                   ! dynamic inflow
&END
!==============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL OPSCEN=1,
    TITLE='SAMPLE JOBS:  WIND TUNNEL SUPPORT',           ! description
    CONFIG=0,OPFREE=0,OPAERO=0,                          ! wind tunnel
    MASSR=9.584,                                         ! inertia
&END
&NLDEF class='AIRFRAME',type='CONTROL',&END
&NLVAL OPSCEN=1,&END
!==============================================================================
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
&NLVAL OPSCEN=1,
    TITLE='SAMPLE JOBS:  ELEMENTARY ROTOR MODEL',        ! description
    RADIUS=20.,NBLADE=4,ROTATE=1,SIGMA=.075,VTIPN=650.,
    EFLAP=0.,EPITCH=0.,TWISTL=-8.,
    KFLAP=108823.325,MASS=.1198,                         ! nu=1.15,gamma=8.
    NRPOS=0,                                             ! sensors
&END
!==============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
&NLVAL OPSCEN=1,
    CHORD=1.1781,
    BTIP=1.,USMODL=0,OPCOMP=0,                           ! ideal aerodynamics
&END
!==============================================================================
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
&NLVAL OPSCEN=1,
    KHLMDA=1.,KFLMDA=1.,OPFFLI=0,                        ! ideal aerodynamics
&END
!==============================================================================
&NLDEF class='TABLES',&END
&NLVAL &END
!==============================================================================
```

```
! turn scenarios off
&NLDEF class='TRIM',&END
&NLVAL OPSCEN=0,&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL OPSCEN=0,&END
&NLDEF class='TRANSIENT',&END
&NLVAL OPSCEN=0,&END
&NLDEF class='TRANSIENT ROTOR',name='ROTOR 1',&END
&NLVAL OPSCEN=0,&END
&NLDEF class='FLUTTER',&END
&NLVAL OPSCEN=0,&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL OPSCEN=0,&END
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL OPSCEN=0,&END
&NLDEF class='AIRFRAME',type='CONTROL',&END
&NLVAL OPSCEN=0,&END
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
&NLVAL OPSCEN=0,&END
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
&NLVAL OPSCEN=0,&END
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
&NLVAL OPSCEN=0,&END
!==============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
!  Rotor in Wind Tunnel
!===============================================================================
&NLDEF class='CASE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  ROTOR IN WIND TUNNEL',          ! description
    OPUNIT=1,OPDENS=3,DENSE=.00237689,TEMP=59.,          ! environment
&END
!===============================================================================
&NLDEF class='TRIM',&END
&NLVAL
    VELIN=2,WINDIN=2,VTIPIN=1,VTIP=650.,                 ! operating condition
    LEVEL=1,                                             ! wake loop
    COLL=10.,CTTRIM=.08,MTRIM=3,                         ! wind tunnel trim
    MNAME='CT/S    ','BETAS   ','BETAC   ',
    VNAME='COLL    ','LATCYC  ','LNGCYC  ',
    MHARMR=10,MHARMA=10,                                 ! harmonics
    DOFA=6*0,DOFM=5*0,                                   ! degrees of freedom
&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL OPMODE=0,DOFB=12*1,&END                           ! degrees of freedom
!===============================================================================
&NLDEF class='FLUTTER',&END
&NLVAL DOFA=6*0,DOFM=5*1,DOFD=8*0,&END                   ! degrees of freedom
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL
    OPWAKE=4,OPVATR=2,OPVRTA=2,                          ! trim inflow
    OPMODE=1,DOFM=2*1,38*0,DOFL=3*1,NMPRNT=1,            ! degrees of freedom
&END
!===============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  WIND TUNNEL SUPPORT'            ! description
    CONFIG=0,OPFREE=0,OPAERO=0,                          ! wind tunnel
    MASSR=9.6,                                           ! inertia
    HSP=2.,OPSPM=0,                                      ! control system
    NLOC=6,                                              ! locations
    FSLOC(5)=5.,-5.,WLLOC(5)=-7.,-7.,                    !   5=center,6=nose
    NMODE=5,                                             ! test module
    QMASS=5*100.,QFREQ=2.1,2.6,20.,22.,23.,QDAMP=2*.02,3*.2,
    LSHAPE(1,1,1) =   1.,   0., 0.,
    LSHAPE(1,1,2) =   0.,   1., 0.,
    LSHAPE(1,1,3) =   1.,   0., 0.,
    LSHAPE(1,1,4) =   0.,   1., 0.,
    LSHAPE(1,1,5) =   0.,   0., 1.,
    ASHAPE(1,1,1) =   0., -1.5, 0.,
    ASHAPE(1,1,2) =  1.5,   0., 0.,
    ASHAPE(1,1,3) =   0.,   0., 0.,
    ASHAPE(1,1,4) =   0.,   0., 0.,
    ASHAPE(1,1,5) =   0.,   0., 0.,
&END
&NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
&NLVAL
    VISRC=0,                                             ! test module
    NBODY=1,LOCBC=5,LOCBN=6,
    LENGTH=20.,THICK=.5,SHAPE=3,
&END
&NLDEF class='AIRFRAME',type='CONTROL',&END
&NLVAL &END
&NLDEF class='AIRFRAME',type='DRIVE TRAIN',&END
&NLVAL
    CONFIG=0,OPGOV=0,
    EGEAR=7.,KRS=10000.,IRS=.1,
    KES=10000.,IENG=50.,DENG=1.,
&END
!===============================================================================
```

```
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
&NLVAL                                                    ! description
    TITLE='SAMPLE JOBS:  HELICOPTER MAIN ROTOR, SWEPT/TAPERED TIP',
    RADIUS=20.,NBLADE=4,ROTATE=1,SIGMA=.075,
    VTIPN=650.,
    HINGE=6,EFLAP=.04,EPITCH=.065,                        ! blade root
    KFLAP=1000.,KLAG=1000.,DLAG=2000.,
    CONTRL=2,PITCH=1,KPITCH=0.,LOCKP=1,                   ! control system
       XSP=-.023,YSP=.045,ZSP=-.1,
       XPH=-.023,YPH=.045,ZPH=0.,
       EPH=.07,KPL=80000.,LOCKPL=1,LOCKSP=0,
    OPBEAM=2,DRELST=.05,KNODE=3,RNODE=.94,.35,.64,        ! elastic blade
    NSEN=3,QUANT=3*4,RLOAD=.4,.6,.8,                      ! loads
    TWIN=1,TWISTL=-10.,                                   ! linear twist
    ETPP=.94,                                             ! tpp sensor
    NPROP=3,RPROP=0.,.94,1.,                              ! section properties
    ZEA=3*0.,ZQC=3*0.,ZC=3*0.,ZI=3*0.,
    XEA=3*0.,XQC=3*0.,XC=3*0.,XI=3*0.,
    XEA=2*0.,.03464,                                      ! 30 deg sweep
    XQC=2*0.,.03464,
    KP=3*.011,KT=3*.011,EA=3*20000000.,
    EIFLAP=3*68000.,EILAG=3*800000.,GJ=3*90000.,
    MASS=3*.12,ITHETA=3*.0056,IPOLAR=3*.0056,
&END
!===============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
&NLVAL
    NPANEL=20,                                            ! aero panels
    REDGE=.12,.20,.28,.35,.42,.48,.54,.59,.64,.69,.73,
          .77,.81,.84,.87,.90,.92,.94,.96,.98,1.0,
    NPROP=4,RPROP=0.,.94,.9401,1.,                        ! aero properties
    CHORD=3*1.1781,.70686,ASWEEP=2*0.,2*30.,              ! 60% taper
    NSEN=6,OPREF=6*4,                                     ! aerodynamic sensors
    QUANT= 5,6,25,35,82,82,                               !   lambda,lambda-int
    IDENT= 1,1, 0, 0, 0, 0,                               !   alpha,theta
    AXIS=  3,3, 0, 0, 1, 3,                               !   Fx,Fz
    OPSCL= 2,2, 1, 1, 2, 2,                               !
    NAPLOT=1,1, 4, 1, 0, 0,                               !
&END
!===============================================================================
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
&NLVAL &END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL
    OPSCEN=1,                                             ! forward flight wake
    TWIST=-10.,RICWG=.15,                                 ! hover wake
&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL OPSCEN=0,&END
!===============================================================================
&NLDEF class='TABLES',&END
&NLVAL &END
!===============================================================================
&NLDEF action='end of shell',&END
!===============================================================================
! hydraulic lag damper (core input)
&NLDEF class='COMPONENT',type='BEAM',name='ROTOR 1 BLADE 1 ELEMENT 1',&END
&NLVAL
    CTYPE(2)=3,CLIN(2)=0.,
    NCHYDA(2)=0,CHYDA(1,2)=1000.,
    NCHYDB(2)=2,CHYDB(1,2)=0.,0.,100000.,
&END
&NLDEF class='COMPONENT',type='BEAM',name='ROTOR 1 BLADE 2 ELEMENT 1',&END
&NLVAL
    CTYPE(2)=3,CLIN(2)=0.,
```

```
    NCHYDA(2)=0,CHYDA(1,2)=1000.,
    NCHYDB(2)=2,CHYDB(1,2)=0.,0.,100000.,
&END
&NLDEF class='COMPONENT',type='BEAM',name='ROTOR 1 BLADE 3 ELEMENT 1',&END
&NLVAL
    CTYPE(2)=3,CLIN(2)=0.,
    NCHYDA(2)=0,CHYDA(1,2)=1000.,
    NCHYDB(2)=2,CHYDB(1,2)=0.,0.,100000.,
&END
&NLDEF class='COMPONENT',type='BEAM',name='ROTOR 1 BLADE 4 ELEMENT 1',&END
&NLVAL
    CTYPE(2)=3,CLIN(2)=0.,
    NCHYDA(2)=0,CHYDA(1,2)=1000.,
    NCHYDB(2)=2,CHYDB(1,2)=0.,0.,100000.,
&END
!===============================================================================
&NLDEF action='end of core',&END
```

```
!  Teetering Rotor in Wind Tunnel
!===============================================================================
&NLDEF class='CASE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  TEETERING ROTOR IN WIND TUNNEL',! description
    OPUNIT=1,OPDENS=3,DENSE=.00237689,TEMP=59.,          ! environment
&END
!===============================================================================
&NLDEF class='TRIM',&END
&NLVAL
    VELIN=2,WINDIN=2,VTIPIN=1,VTIP=650.,                 ! operating condition
    LEVEL=2*1,                                           ! wake loop
    COLL=10.,CTTRIM=.08,MTRIM=1,                         ! wind tunnel trim
    MNAME='CT/S    ',VNAME='COLL    ',
    RELAXT=.5,                                           ! trim loop
    OPPART=3,                                            ! teetering rotor
    MHARMR=0,MHARMA=0,MPSIAV=1,                          ! harmonics
    DOFA=6*0,DOFM=2*0,                                   ! degrees of freedom
&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL
    NAPRNT=1,NAFILE=1,MATIME=1,                          ! output
    OPMODE=0,DOFG=1,DOFB=1,0,1,9*0,                      ! degrees of freedom
&END
!===============================================================================
&NLDEF class='FLUTTER',&END
&NLVAL
    MPSIAV=1,                                            ! no average
    DOFA=6*0,DOFM=2*1,DOFD=8*0,                          ! degrees of freedom
&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL
    OPWAKE=4,OPVATR=2,OPVRTA=2,                          ! trim inflow
    OPMODE=0,DOFG=1,DOFB=1,0,1,9*0,DOFL=3*0,             ! degrees of freedom
&END
!===============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  WIND TUNNEL SUPPORT'            ! description
    CONFIG=0,OPFREE=0,OPAERO=0,                          ! wind tunnel
    MASSR=8.4,                                           ! inertia
    NMODE=2,                                             ! test module
    QMASS=2*1000.,QFREQ=2.,4.,QDAMP=2*.1,
    LSHAPE(1,1,1) =   1.,   0., 0.,
    LSHAPE(1,1,2) =   0.,   1., 0.,
    ASHAPE(1,1,1) =   0., -1.5, 0.,
    ASHAPE(1,1,2) =  1.5,   0., 0.,
&END
&NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
&NLVAL &END
&NLDEF class='AIRFRAME',type='CONTROL',&END
&NLVAL &END
&NLDEF class='AIRFRAME',type='DRIVE TRAIN',&END
&NLVAL &END
!===============================================================================
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
&NLVAL
    TITLE='SAMPLE JOBS:  TEETERING MAIN ROTOR',          ! description
    RADIUS=20.,NBLADE=2,ROTATE=1,SIGMA=.05,
    VTIPN=650.,
    GIMBAL=1,CONTRL=0,HINGE=0,PITCH=2,                   ! blade root
    CONE=2.,ZUS=-.01,EPITCH=.05,LOCKP=0,
    OPBEAM=0,KNODE=0,                                    ! rigid blade
    TWIN=1,TWISTL=-10.,                                  ! linear twist
    NPROP=2,RPROP=0.,1.,                                 ! section properties
```

```
    ZEA=2*0.,ZQC=2*0.,ZC=2*0.,ZI=2*0.,
    XEA=2*0.,XQC=2*0.,XC=2*0.,XI=2*0.,
    KP=2*.015,KT=2*.015,EA=2*60000000.,
    EIFLAP=2*500000.,EILAG=2*5000000.,GJ=2*93000.,
    MASS=2*.21,ITHETA=2*.018,IPOLAR=2*.018,
&END
!===============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
&NLVAL
    NPANEL=20,                                          ! aero panels
    REDGE=.12,.20,.28,.35,.42,.48,.54,.59,.64,.69,.73,
          .77,.81,.84,.87,.90,.92,.94,.96,.98,1.0,
    NPROP=2,RPROP=0.,1.,                                ! aero properties
    CHORD=2*1.5708,
    NSEN=5,OPREF=5*4,                                   ! aerodynamic sensors
    QUANT= 5,25,35,82,82,                               !   lambda
    IDENT= 1, 0, 0, 0, 0,                               !   alpha,theta
    AXIS=  3, 0, 0, 1, 3,                               !   Fx,Fz
    OPSCL= 2, 1, 1, 2, 2,                               !
    NAPLOT=1, 4, 1, 0, 0,                               !
&END
!===============================================================================
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
&NLVAL &END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL
    OPSCEN=1,                                           ! forward flight wake
    TWIST=-10.,RICWG=.20,                               ! hover wake
&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL OPSCEN=0,&END
!===============================================================================
&NLDEF class='TABLES',&END
&NLVAL &END
!===============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
!  Single Main Rotor and Tail Rotor Helicopter
!===============================================================================
&NLDEF class='CASE',&END
&NLVAL                                                    ! description
    TITLE='SAMPLE JOBS:  SINGLE MAIN ROTOR AND TAIL ROTOR HELICOPTER',
    OPUNIT=1,OPDENS=3,DENSE=.00237689,TEMP=59.,           ! environment
&END
!===============================================================================
&NLDEF class='TRIM',&END
&NLVAL
    VELIN=1,WINDIN=1,VTIPIN=1,VTIP=650.,                  ! operating condition
    LEVEL=2*1,                                            ! wake loop
    COLL=10.,MTRIM=6,                                     ! free flight trim
    MNAME='FORCE X ','FORCE Y ','FORCE Z ',
          'MOMENT X','MOMENT Y','MOMENT Z',
    VNAME='COLL    ','LATCYC  ','LNGCYC  ',
          'PEDAL   ','PITCH   ','ROLL    ',
    MHARMR=10,1,MHARMA=10,1,MHARMD=10,1,                  ! harmonics
    DOFA=6*0,DOFD=8*0,                                    ! degrees of freedom
&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL OPMODE=0,DOFB=1,0,1,9*0,&END                       ! degrees of freedom
&NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
&NLVAL OPMODE=0,DOFB=1,0,1,9*0,&END                       ! degrees of freedom
!===============================================================================
&NLDEF class='TRANSIENT',&END
&NLVAL DOFA=6*1,DOFD=8*0,&END                             ! degrees of freedom
&NLDEF class='TRANSIENT ROTOR',name='ROTOR 1',&END
&NLVAL
    OPWAKE=4,OPVATR=2,OPVRTA=2,                           ! trim inflow
    OPMODE=0,DOFB=1,0,1,9*0,DOFMBC=6*1,                   ! degrees of freedom
&END
&NLDEF class='TRANSIENT ROTOR',name='ROTOR 2',&END
&NLVAL
    OPWAKE=4,OPVATR=2,OPVRTA=2,                           ! trim inflow
    OPMODE=0,DOFB=1,0,1,9*0,DOFMBC=6*1,                   ! degrees of freedom
&END
!===============================================================================
&NLDEF class='FLUTTER',&END
&NLVAL DOFA=6*1,&END                                      ! degrees of freedom
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL
    OPWAKE=4,OPVATR=2,OPVRTA=2,                           ! trim inflow
    OPMODE=0,DOFB=1,0,1,9*0,DOFL=3*0,                     ! degrees of freedom
&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR 2',&END
&NLVAL
    OPWAKE=4,OPVATR=2,OPVRTA=2,                           ! trim inflow
    OPMODE=0,DOFB=1,0,1,9*0,DOFL=3*0,                     ! degrees of freedom
&END
!===============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  HELICOPTER AIRFRAME',            ! description
    CONFIG=1,RGEAR(2)=4.,                                 ! single mr/tr
    WEIGHT=8000.,IXX=2600.,IYY=12000.,IZZ=10000.,         ! inertia
    MASSR=9.6,.48,
    HSP=2.,                                               ! control
    FSCG=0.,WLCG=0.,                                      ! geometry
    FSRTR=0.,24.,WLRTR=6.,0.,BLRTR=0.,0.,
    ASHAFT=-4.,0.,
    FSWB=0.,WLWB=0.,FSHT=20.,WLHT=0.,FSVT=24.,WLVT=0.,
    NLOC=8,                                               ! locations
    FSLOC(5)=0.,-4., 2*-9.,                               !   5-6 for sensor
    WLLOC(5)=0.,0.,  2*3.,                                !   7-8 for inflow
```

```
    BLLOC(5)=0.,2.,  0.,5.,
    NSEN=3,LOCSEN=5,5,6,QUANT=4,5,17,OPSCL=2*1,0,        ! airframe sensors
&END
&NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
&NLVAL
    LFTAW=50.,DRG0W=12.,DRGVW=0.,AMAXW=20.,MOMAW=400.,
    SIDEB=-150.,YAWB=-1500.,
    LFTAH=50.,AMAXH=20.,IHTL=5.,LFTAV=40.,AMAXV=20.,
&END
&NLDEF class='AIRFRAME',type='CONTROL',&END
&NLVAL &END
&NLDEF class='AIRFRAME',type='DRIVE TRAIN',&END
&NLVAL
    CONFIG=1,OPGOV=0,IGEAR=6.,EGEAR=12.,GAINE=100.,
    KRS=50000.,1000.,IRS=.1,.1,
    KIS=10000.,KES=5000.,IENG=.02,
&END
!===============================================================================
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
&NLVAL
    TITLE='SAMPLE JOBS:  HELICOPTER MAIN ROTOR',         ! description
    RADIUS=20.,NBLADE=4,ROTATE=1,SIGMA=.075,
    VTIPN=650.,
    HINGE=3,                                             ! blade root
    EFLAP=.04,ELAG=.06,EPITCH=.065,DLAG=2000.,
    CONTRL=2,PITCH=1,KPITCH=0.,LOCKP=1,                  ! control system
       XSP=-.03,YSP=.035,ZSP=-.1,
       XPH=-.03,YPH=.035,ZPH=0.,
       EPH=.07,KPL=80000.,LOCKPL=1,LOCKSP=0,
    OPBEAM=2,DRELST=.1,KNODE=2,RNODE=.35,.64,            ! elastic blade
    NSEN=3,QUANT=3*4,RLOAD=.4,.6,.8,                     ! loads
    TWIN=1,TWISTL=-10.,                                  ! linear twist
    NPROP=2,RPROP=0.,1.,                                 ! section properties
    ZEA=2*0.,ZQC=2*0.,ZC=2*0.,ZI=2*0.,
    XEA=2*0.,XQC=2*0.,XC=2*0.,XI=2*0.,
    KP=2*.011,KT=2*.011,EA=2*20000000.,
    EIFLAP=2*68000.,EILAG=2*800000.,GJ=2*90000.,
    MASS=2*.12,ITHETA=2*.0056,IPOLAR=2*.0056,
&END
!===============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
&NLVAL
    NPANEL=20,                                           ! aero panels
    REDGE=.12,.20,.28,.35,.42,.48,.54,.59,.64,.69,.73,
          .77,.81,.84,.87,.90,.92,.94,.96,.98,1.0,
    NPROP=2,RPROP=0.,1.,                                 ! aero properties
    CHORD=2*1.1781,
    NSEN=5,OPREF=5*4,                                    ! aerodynamic sensors
    QUANT= 5,25,35,82,82,                                !   lambda
    IDENT= 1, 0, 0, 0, 0,                                !   alpha,theta
    AXIS=  3, 0, 0, 1, 3,                                !   Fx,Fz
    OPSCL= 2, 1, 1, 2, 2,                                !
    NAPLOT=1, 4, 1, 0, 0,                                !
&END
!===============================================================================
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
&NLVAL KINTFW=1.5,KINTFH=1.8,&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL
    OPSCEN=1,                                            ! forward flight wake
    TWIST=-10.,RICWG=.15,                                ! hover wake
    NPOFF=2,LOCOFF=7,8,                                  ! inflow off rotor
&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL OPSCEN=0,&END
```

```
!==============================================================================
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 2',&END
&NLVAL
    TITLE='SAMPLE JOBS:  HELICOPTER TAIL ROTOR',          ! description
    RADIUS=5.,NBLADE=4,ROTATE=1,SIGMA=.15,
    VTIPN=650.,
    CONTRL=0,HINGE=3,PITCH=2,                             ! blade root
    EFLAP=0.,ELAG=.1,EPITCH=.15,
    AFLAP=30.,LOCKP=0,DLAG=10.,
    OPBEAM=0,KNODE=0,                                     ! rigid blade
    TWIN=1,TWISTL=0.,                                     ! linear twist
    NPROP=2,RPROP=0.,1.,                                  ! section properties
    ZEA=2*0.,ZQC=2*0.,ZC=2*0.,ZI=2*0.,
    XEA=2*0.,XQC=2*0.,XC=2*0.,XI=2*0.,
    KP=2*.022,KT=2*.022,EA=2*2000000.,
    EIFLAP=2*850.,EILAG=2*25000.,GJ=2*1500.,
    MASS=2*.024,ITHETA=2*.00028,IPOLAR=2*.00028,
&END
!==============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 2',&END
&NLVAL
    NPANEL=8,REDGE=.2,.3,.4,.5,.6,.7,.8,.9,1.,
    NPROP=2,RPROP=0.,1.,                                  ! aero properties
    CHORD=2*.58905,BTIP=.95,
&END
!==============================================================================
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 2',&END
&NLVAL KINTHV=.7,KINTFV=.7,&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 2',&END
&NLVAL &END
!==============================================================================
&NLDEF class='TABLES',&END
&NLVAL AFTABL=1,&END                                      ! separate tables
!==============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
!  Bearingless Rotor
!===============================================================================
&NLDEF class='CASE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  BEARINGLESS ROTOR',              ! description
    OPUNIT=1,OPDENS=3,DENSE=.00237689,TEMP=59.,           ! environment
&END
!===============================================================================
&NLDEF class='TRIM',&END
&NLVAL
    VELIN=1,WINDIN=1,WKTS=0.,VTIPIN=1,VTIP=650.,          ! operating condition
    LEVEL=2*1,                                            ! wake loop
    OPTRIM=0,COLL=10.,                                    ! no trim
    MHARMR=0,MHARMA=0,DOFA=6*0,MPSIAV=1,MPSI=4,           ! hover
&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL OPMODE=0,DOFB=12*1,&END                            ! no blade modes
!===============================================================================
&NLDEF class='FLUTTER',&END
&NLVAL
    MPSIAV=1,                                             ! no average
    DOFA=6*0,                                             ! degrees of freedom
&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL
    OPWAKE=4,OPVATR=2,OPVRTA=2,                           ! trim inflow
    OPMODE=1,DOFM=4*1,36*0,DOFL=3*0,                      ! degrees of freedom
    GDAMPM=40*.01,NMPRNT=1,
&END
!===============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  WIND TUNNEL SUPPORT',            ! description
    CONFIG=0,OPFREE=0,OPAERO=0,                           ! wind tunnel
    MASSR=12.26,                                          ! inertia
    HSP=2.,                                               ! control
&END
&NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
&NLVAL &END
&NLDEF class='AIRFRAME',type='CONTROL',&END
&NLVAL &END
&NLDEF class='AIRFRAME',type='DRIVE TRAIN',&END
&NLVAL &END
!===============================================================================
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
&NLVAL
    TITLE='SAMPLE JOBS:  BEARINGLESS ROTOR',              ! description
    RADIUS=20.,NBLADE=5,ROTATE=1,SIGMA=.075,
    VTIPN=650.,
    CONFIG=2,CONTRL=2,LOCKSP=0,HINGE=0,PITCH=0,           ! blade root
       XSP=.04,YSP=.05,ZSP=-.1,                           !   te pitch link
       XPH=.04,YPH=.05,ZPH=0.,
       EPH=.05,LOCKPL=1,KPL=100000.,
    CONE=2.,
    OPBEAM=2,KNODE=3,RNODE=.28,.48,.73,                   ! elastic blade
    DRELST=.04,KRIGID=1,RRGDB=0.,RRGDE=.28,
    TWIN=1,TWISTL=-10.,                                   ! linear twist
    NPROP=4,RPROP=0.,.2499,.2501,1.,                      ! section properties
    ZEA=4*0.,ZQC=4*0.,ZC=4*0.,ZI=4*0.,
    XEA=4*0.,XQC=4*0.,XC=4*0.,XI=4*0.,
    KP=2*.020,2*.011,KT=2*.020,2*.011,EA=4*40000000.,
    EIFLAP=2*600000.,2*60000.,EILAG=4*1650000.,GJ=2*900000.,2*90000.,
    MASS=4*.12,ITHETA=2*.02,2*.0056,IPOLAR=2*.001,2*.0056,
&END
!===============================================================================
```

```
&NLDEF class='ROTOR',type='FLEXBEAM',name='ROTOR 1',&END
&NLVAL
    EBLADE=.02,EFB=.25,SNUB=1,                                 ! blade root
       XSNUB=0.,YSNUB=.025,ZSNUB=.005873,                      !   snubber
       XROOT=0.,EROOT=.025,ZROOT=.005000,
       KXSNUB=0.,KYSNUB=0.,KZSNUB=100000.,
       DXSNUB=10.,DYSNUB=0.,DZSNUB=0.,
    OPBEAM=2,KNODE=2,RNODE=.03,.23,                            ! elastic flexbeam
    DRELST=.04,KRIGID=2,RRGDB=0.,.23,RRGDE=.03,.25,
    NPROP=2,RPROP=0.,1.,                                       ! section properties
    ZEA=2*0.,ZC=2*0.,ZI=2*0.,
    XEA=2*0.,XC=2*0.,XI=2*0.,
    KP=2*.011,KT=2*.011,EA=2*40000000.,
    EIFLAP=2*36000.,EILAG=2*440000.,GJ=2*5000.,
    MASS=2*.02,ITHETA=2*.001,IPOLAR=2*.001,
&END
!===============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
&NLVAL
    NPANEL=20,                                                 ! aero panels
    REDGE=.12,.20,.28,.35,.42,.48,.54,.59,.64,.69,.73,
          .77,.81,.84,.87,.90,.92,.94,.96,.98,1.0,
    NPROP=2,RPROP=0.,1.,                                       ! aero properties
    CHORD=2*.94248,
    NSEN=5,OPREF=5*4,                                          ! aerodynamic sensors
    QUANT= 5,25,35,82,82,                                      !   lambda
    IDENT= 1, 0, 0, 0, 0,                                      !   alpha,theta
    AXIS=  3, 0, 0, 1, 3,                                      !   Fx,Fz
    OPSCL= 2, 1, 1, 2, 2,                                      !
    NAPLOT=1, 4, 1, 0, 0,                                      !
&END
!===============================================================================
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
&NLVAL &END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL &END
!===============================================================================
&NLDEF class='TABLES',&END
&NLVAL &END
!===============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
!  Tandem Helicopter
!==============================================================================
&NLDEF class='CASE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  TANDEM HELICOPTER',              ! description
    OPUNIT=1,DENSE=.00237689,TEMP=59.,                    ! environment
&END
!==============================================================================
&NLDEF class='TRIM',&END
&NLVAL
    VELIN=1,WINDIN=1,VTIPIN=1,VTIP=650.,                  ! operating condition
    LEVEL=2*2,                                            ! wake loop
    COLL=10.,MTRIM=6,                                     ! free flight trim
    MNAME='FORCE X ','FORCE Y ','FORCE Z ',
          'MOMENT X','MOMENT Y','MOMENT Z',
    VNAME='COLL    ','LATCYC  ','LNGCYC  ',
          'PEDAL   ','PITCH   ','ROLL    ',
    MHARMR=2*1,MHARMA=2*1,                                ! harmonics
    DOFA=6*0,                                             ! degrees of freedom
    CNTRLF=0.,                                            ! auxiliary propulsion
&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL OPMODE=0,DOFB=1,0,1,9*0,&END                       ! degrees of freedom
&NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
&NLVAL OPMODE=0,DOFB=1,0,1,9*0,&END                       ! degrees of freedom
!==============================================================================
&NLDEF class='FLUTTER',&END
&NLVAL DOFA=6*1,&END                                      ! degrees of freedom
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL OPMODE=0,DOFB=3*1,9*0,DOFL=1,2*0,&END              ! degrees of freedom
&NLDEF class='FLUTTER ROTOR',name='ROTOR 2',&END
&NLVAL OPMODE=0,DOFB=3*1,9*0,DOFL=1,2*0,&END              ! degrees of freedom
!==============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  TANDEM HELICOPTER AIRFRAME',     ! description
    CONFIG=2,                                             ! tandem
    WEIGHT=16000.,IXX=6000.,IYY=60000.,IZZ=55000.,        ! inertia
    FSRTR=-15.,14.,WLRTR=6.,10.,ASHAFT=-5.,-2.,           ! geometry
    NFRC=1,LOCFRC=5,AXFRC=1,NLOC=5,                       ! auxiliary propulsion
&END
&NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
&NLVAL
    LFTAW=300.,IWBM=-5.,MOMAW=1200.,
    DRG0W=40.,DRGVW=40.,DRGIW=.01,AMAXW=20.,
    SIDEB=-300.,ROLLB=-1200.,YAWB=-700.,
&END
&NLDEF class='AIRFRAME',type='CONTROL',&END
&NLVAL KF0=1.1,KR0=.9,&END
&NLDEF class='AIRFRAME',type='DRIVE TRAIN',&END
&NLVAL &END
!==============================================================================
&NLDEF class='TABLES',&END
&NLVAL &END
!==============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
!  Tandem Helicopter Front Rotor
!==============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL MASSR(1)=9.6,&END
!==============================================================================
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
&NLVAL
    TITLE='SAMPLE JOBS:  TANDEM HELICOPTER FRONT ROTOR', ! description
    RADIUS=20.,NBLADE=4,ROTATE=-1,SIGMA=.075,
    VTIPN=650.,
    CONTRL=0,HINGE=3,PITCH=2,                           ! blade root
    EFLAP=.02,ELAG=.06,EPITCH=.05,
    DLAG=1000.,LOCKP=0,
    OPBEAM=0,KNODE=0,                                   ! rigid blade
    TWIN=1,TWISTL=-10.,                                 ! linear twist
    NPROP=2,RPROP=0.,1.,                                ! section properties
    ZEA=2*0.,ZQC=2*0.,ZC=2*0.,ZI=2*0.,
    XEA=2*0.,XQC=2*0.,XC=2*0.,XI=2*0.,
    KP=2*.011,KT=2*.011,
    EIFLAP=2*68000.,EILAG=2*800000.,GJ=2*90000.,EA=2*20000000.,
    MASS=2*.12,ITHETA=2*.0056,IPOLAR=2*.0056,
&END
!==============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
&NLVAL
    NPANEL=20,                                          ! aero panels
    REDGE=.12,.20,.28,.35,.42,.48,.54,.59,.64,.69,.73,
          .77,.81,.84,.87,.90,.92,.94,.96,.98,1.0,
    NPROP=2,RPROP=0.,1.,                                ! aero properties
    CHORD=2*1.1781,
    NSEN=6,OPREF=6*4,                                   ! aerodynamic sensors
    QUANT= 5,6,25,35,82,82,                             !   lambda,lambda-int
    IDENT= 1,1, 0, 0, 0, 0,                             !   alpha,theta
    AXIS=  3,3, 0, 0, 1, 3,                             !   Fx,Fz
    OPSCL= 2,2, 1, 1, 2, 2,                             !
    NAPLOT=1,1, 4, 1, 0, 0,                             !
&END
!==============================================================================
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
&NLVAL
    OPFFLI=2,FMLMDA=2*0.,                               ! uniform inflow
    KINTFR(2)=1.8,KINTHR(2)=.06,                        ! interference inflow
&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL OPSCEN=1,&END                                    ! forward flight wake
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL OPSCEN=0,
    CORE=.5,COREWG=.5,                                  ! vortex core
    OPFWG=4,OPDISP=2*0,RFW=3.,MFWG=3,                   ! wake geometry
&END
!==============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
!  Tandem Helicopter Rear Rotor
!===============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL MASSR(2)=9.6,&END
!===============================================================================
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 2',&END
&NLVAL
    TITLE='SAMPLE JOBS:  TANDEM HELICOPTER REAR ROTOR',  ! description
    RADIUS=20.,NBLADE=4,ROTATE=1,SIGMA=.075,
    VTIPN=650.,
    CONTRL=0,HINGE=3,PITCH=2,                            ! blade root
    EFLAP=.02,ELAG=.06,EPITCH=.05,
    DLAG=1000.,LOCKP=0,
    OPBEAM=0,KNODE=0,                                    ! rigid blade
    TWIN=1,TWISTL=-10.,                                  ! linear twist
    NPROP=2,RPROP=0.,1.,                                 ! section properties
    ZEA=2*0.,ZQC=2*0.,ZC=2*0.,ZI=2*0.,
    XEA=2*0.,XQC=2*0.,XC=2*0.,XI=2*0.,
    KP=2*.011,KT=2*.011,
    EIFLAP=2*68000.,EILAG=2*800000.,GJ=2*90000.,EA=2*20000000.,
    MASS=2*.12,ITHETA=2*.0056,IPOLAR=2*.0056,
&END
!===============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 2',&END
&NLVAL
    NPANEL=20,                                           ! aero panels
    REDGE=.12,.20,.28,.35,.42,.48,.54,.59,.64,.69,.73,
          .77,.81,.84,.87,.90,.92,.94,.96,.98,1.0,
    NPROP=2,RPROP=0.,1.,                                 ! aero properties
    CHORD=2*1.1781,
    NSEN=6,OPREF=6*4,                                    ! aerodynamic sensors
    QUANT= 5,6,25,35,82,82,                              !   lambda,lambda-int
    IDENT= 1,1, 0, 0, 0, 0,                              !   alpha,theta
    AXIS=  3,3, 0, 0, 1, 3,                              !   Fx,Fz
    OPSCL= 2,2, 1, 1, 2, 2,                              !
    NAPLOT=1,1, 4, 1, 0, 0,                              !
&END
!===============================================================================
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 2',&END
&NLVAL
    OPFFLI=2,FMLMDA=2*0.,                                ! uniform inflow
    KINTFR(1)=0.,KINTHR(1)=.06,                          ! interference
&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 2',&END
&NLVAL OPSCEN=1,&END                                     ! forward flight wake
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 2',&END
&NLVAL OPSCEN=0,
    CORE=.5,COREWG=.5,                                   ! vortex core
    OPFWG=4,OPDISP=2*0,RFW=2.,MFWG=2,                    ! wake geometry
&END
!===============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
!  Tiltrotor in Wind Tunnel
!===============================================================================
&NLDEF class='CASE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  TILTROTOR IN WIND TUNNEL',      ! description
    OPUNIT=1,OPDENS=3,DENSE=.00237689,TEMP=59.,          ! environment
&END
!===============================================================================
&NLDEF class='TRIM',&END
&NLVAL
    VELIN=1,WINDIN=1,VTIPIN=1,VTIP=600.,                 ! operating condition
    LEVEL=1,                                             ! wake loop
    COLL=10.,CTTRIM=.08,MTRIM=1,                         ! wind tunnel trim
    MNAME='CT/S    ',VNAME='COLL    ',
    OPPART=3,                                            ! gimballed rotor
    MHARMR=0,MHARMA=0,MHARMD=0,MPSIAV=1,                 ! harmonics
    DOFA=6*0,DOFM=3*0,DOFD=8*0,                          ! degrees of freedom
&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL OPMODE=0,DOFG=1,DOFB=12*1,&END                    ! degrees of freedom
!===============================================================================
&NLDEF class='FLUTTER',&END
&NLVAL
    MPSIAV=1,                                            ! no average
    DOFA=6*0,DOFM=3*1,DOFD=1,7*0,DOFORD=5*1,             ! degrees of freedom
&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL
    OPWAKE=4,OPVATR=2,OPVRTA=2,                          ! trim inflow
    OPMODE=1,DOFG=1,DOFM=4*1,36*0,DOFL=2,2*0,            ! degrees of freedom
    GDAMPM=40*.06,NMPRNT=1,
&END
!===============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  CANTILEVER WING (ATILT=0)',     ! description
    CONFIG=0,OPFREE=0,                                   ! wind tunnel
    OPAERO=1,                                            ! wing aerodynamics
    OPTRAN=1,                                            ! drive train
    ASHAFT=-90.,                                         ! geometry
    NMODE=3,                                             ! elastic modes
    MENAME='BEAM','CHORD','TORSION',
    MELABL='BEAM','CHORD','TORSION',
    QMASS=80.,80.,1.2,QFREQ=3.,5.,11.,
    QDAMP=3*.04,QAEROD=100.,1.,3.,
    LSHAPE(1,1,1)=  0., 0.,-1.,
    LSHAPE(1,1,2)= -1.,.45, 0.,
    LSHAPE(1,1,3)=  0., 0.,-.2,
    ASHAPE(1,1,1)= -.1,.04, 0.,
    ASHAPE(1,1,2)=  0., 0., .1,
    ASHAPE(1,1,3)= .02,.06, 0.,
&END
&NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
&NLVAL &END
&NLDEF class='AIRFRAME',type='CONTROL',&END
&NLVAL &END
&NLDEF class='AIRFRAME',type='DRIVE TRAIN',&END
&NLVAL
    CONFIG=0,OPGOV=0,
    EGEAR=1.,KRS=1000000.,KES=10000.,
&END
!===============================================================================
&NLDEF class='TABLES',&END
&NLVAL &END
!===============================================================================
```

```
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
!  Tilting Proprotor Aircraft
!===============================================================================
&NLDEF class='CASE',&END
&NLVAL
    TITLE='SAMPLE JOBS:  TILTING PROPROTOR AIRCRAFT',    ! description
    OPUNIT=1,DENSE=.00237689,TEMP=59.,                   ! environment
&END
!===============================================================================
&NLDEF class='TRIM',&END
&NLVAL
    VELIN=1,WINDIN=1,VTIPIN=1,VTIP=600.,                 ! operating condition
    LEVEL=2*1,                                           ! wake loop
    GOV=10.,MTRIM=3,                                     ! free flight trim
    MNAME='FORCE X ','FORCE Z ','MOMENT Y',
    VNAME='GOV     ','LNGCYC  ','PITCH   ',
    OPPART=2*3,                                          ! gimballed rotors
    MHARMR=2*10,MHARMA=2*10,MHARMD=2*10,                 ! harmonics
    DOFA=6*1,DOFM=10*1,DOFD=8*1,                         ! degrees of freedom
&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL
    CNTRLR=25.,2*0.,
    OPMODE=0,DOFG=1,DOFB=12*1,                           ! degrees of freedom
&END
&NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
&NLVAL
    CNTRLR=25.,2*0.,
    OPMODE=0,DOFG=1,DOFB=12*1,                           ! degrees of freedom
&END
!===============================================================================
&NLDEF class='FLUTTER',&END
&NLVAL
    DOFA=6*1,DOFM=8*1,2*0,DOFD=8*1,                      ! degrees of freedom
    OPEQN=4*0,2*1,3*0,                                   ! sym/antisym eqns
&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL
    OPWAKE=3,OPVATR=2,OPVRTA=2,                          ! dynamic inflow
    OPMODE=1,DOFG=1,DOFM=4*1,36*0,DOFL=2,2*0,            ! degrees of freedom
    GDAMPM=40*.06,
&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR 2',&END
&NLVAL
    OPWAKE=3,OPVATR=2,OPVRTA=2,                          ! dynamic inflow
    OPMODE=1,DOFG=1,DOFM=4*1,36*0,DOFL=2,2*0,            ! degrees of freedom
    GDAMPM=40*.06,
&END
!===============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL                                                   ! description
    TITLE='SAMPLE JOBS:  TILTING PROPROTOR AIRCRAFT AIRFRAME',
    CONFIG=4,ATILT=0.,OPTRAN=1,                          ! tiltrotor
    WEIGHT=15000.,                                       ! inertia
    IXX=50000.,IYY=15000.,IZZ=60000,
    FSCG=-1.,WLCG=-2.,                                   ! geometry
    FSRTR=0.,BLRTR=20.,WLRTR=5.,ASHAFT=0.,ACANT=1.,
    FSPIV=0.,BLPIV=20.,WLPIV=0.,ASPIV=0.,ADPIV=-1.,
    FSWB=0.,WLWB=0.,FSHT=25.,WLHT=0.,FSVT=25.,WLVT=0.,
    NLOC=17,                                             ! locations
    FSLOC(5)=-7.,WLLOC(5)=-4.,                           !   5=sensor
    FSLOC(6)=  0.,2., 0., 25.,25.,25.,                   !   6-11=wings
    BLLOC(6)=-20.,0.,20., -7., 0., 7.,
    WLLOC(6)=  0.,0., 0.,  0., 0., 0.,
    FSLOC(12)=-3.,-15.5,  0.,-8.,  0.,-8.,               !   12-17=bodies
    BLLOC(12)=2*0.,       2*-20.,  2*20.,
```

```
WLLOC(12)=2*-2.,      2*0.,    2*0.,
NSEN=1,LOCSEN=5,QUANT=3,OPSCL=1,                         ! sensor
NMODE=10,                                                ! elastic modes
QMASS=10*12.,QDAMP=3*.03,.2,3*.03,3*.2,
QFREQ=3.,6.,8.,18.,6.,8.,7.,20.,15.,22.,
QSYM=4*1,4*-1,2*1,
QAEROD=10.,2.,20.,12.,24.,80.,14.,16.,0.,0.,
QAEROC=   15., -10., 2*0.,
           0.,   8., 2*0.,
          -4.,  35., 2*0.,
         -32., -15., 2*0.,
         2*0.,-16.,  -12.,
         2*0.,  5.,  -40.,
         2*0.,  0.,   15.,
         2*0.,-25.,   11.,
           0.,  0.,  2*0.,
           0.,  0.,  2*0.,
! location1=rotor1(right)
LSHAPE(1,1,1)=   .0,  .0, -.2,
LSHAPE(1,1,2)=  -.2,  .2,  .2,
LSHAPE(1,1,3)=  -.1,  .1, -.3,
LSHAPE(1,1,4)=   .0,  .4,  .0,
LSHAPE(1,1,5)=   .0,  .1, -.2,
LSHAPE(1,1,6)=  -.1,  .1, -.1,
LSHAPE(1,1,7)=   .0,  .0, -.3,
LSHAPE(1,1,8)=   .0,  .3,  .0,
LSHAPE(1,1,9)=   .0,  .1,  .0,
LSHAPE(1,1,10)=  .0,  .2,  .1,
ASHAPE(1,1,1)=  -.03, .0,  .0,
ASHAPE(1,1,2)=   .0, -.06, .05,
ASHAPE(1,1,3)=   .0,  .1,  .04,
ASHAPE(1,1,4)=  -.06, .0,  .2,
ASHAPE(1,1,5)=  -.04, .04, .0,
ASHAPE(1,1,6)=   .02, .07, .07,
ASHAPE(1,1,7)=   .0,  .1,  .0,
ASHAPE(1,1,8)=  -.1,  .0,  .2,
ASHAPE(1,1,9)=   .0,  .0,  .03,
ASHAPE(1,1,10)=  .0, -.1,  .1,
! location2=rotor2(left)
LSHAPE(1,2,1)=   .0, -.0, -.2,
LSHAPE(1,2,2)=  -.2, -.2,  .2,
LSHAPE(1,2,3)=  -.1, -.1, -.3,
LSHAPE(1,2,4)=   .0, -.4,  .0,
LSHAPE(1,2,5)=   .0, -.1, -.2,
LSHAPE(1,2,6)=  -.1, -.1, -.1,
LSHAPE(1,2,7)=   .0, -.0, -.3,
LSHAPE(1,2,8)=   .0, -.3,  .0,
LSHAPE(1,2,9)=   .0, -.1,  .0,
LSHAPE(1,2,10)=  .0, -.2,  .1,
ASHAPE(1,2,1)=   .03, .0, -.0,
ASHAPE(1,2,2)=  -.0, -.06,-.05,
ASHAPE(1,2,3)=  -.0,  .1, -.04,
ASHAPE(1,2,4)=   .06, .0, -.2,
ASHAPE(1,2,5)=   .04, .04,-.0,
ASHAPE(1,2,6)=  -.02, .07,-.07,
ASHAPE(1,2,7)=  -.0,  .1, -.0,
ASHAPE(1,2,8)=   .1,  .0, -.2,
ASHAPE(1,2,9)=   .0,  .0, -.03,
ASHAPE(1,2,10)=  .0, -.1, -.1,
! location5=sensor
LSHAPE(1,5,1)=   .0,  .0,  .1,
LSHAPE(1,5,2)=   .1,  .0,  .03,
LSHAPE(1,5,3)=   .04, .0,  .08,
LSHAPE(1,5,4)=  -.04, .0, -.06,
LSHAPE(1,5,5)=   .0, -.16, .0,
```

```
    LSHAPE(1,5,6)=   .0, -.16, .0,
    LSHAPE(1,5,7)=   .0,  .1,  .0,
    LSHAPE(1,5,8)=   .0,  .04, .0,
    LSHAPE(1,5,9)=  -.03, .0,  1.,
    LSHAPE(1,5,10)= -.02, .0, -.02,
&END
&NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
&NLVAL
    LFTAW=1000.,IWBL=4.,IWBD=4.,IWBM=-2.,LFTDW=200.,LFTFW=300.,
    DRG0W=12.,DRGVW=12.,DRGIW=.0005,DRGDW=12.,DRGFW=5.,
    AMAXW=20.,MOMAW=1400.,MOMDW=-200.,MOMFW=-300.,
    SIDEB=-100.,ROLLB=200.,ROLLP=-90000.,ROLLR=9000.,ROLLDA=-3500.,
    YAWB=-1500.,YAWP=-2000.,YAWR=-2000.,YAWDA=70.,
    LFTAH=250.,LFTEH=150.,AMAXH=15.,LFTAV=180.,LFTRV=70.,AMAXV=20.,
    EHTAIL=.0005,LHTAIL=25.,HVTAIL=3.,
!
    VISRC=2*0,
    NWING=2,LOCWL=6,9,LOCWM=7,10,LOCWR=8,11,
    CIRC=2*0.,FCIRCW=1.,0.,FCIRCH=0.,1.,FCIRCV=0.,0.,AXS=5.,2.,
    XCIRC=.19,.25,XTHICK=.27,.375,SPAN=40.,14.,CHORD=6.,4.,
    NBODY=3,LOCBC=12,14,16,LOCBN=13,15,17,
    LENGTH=25.,2*16.,THICK=.25,2*.18,SHAPE=3*1,
&END
&NLDEF class='AIRFRAME',type='CONTROL',&END
&NLVAL
    K0=0.,KC=0.,KS=0.,KP=0.,
    KT=4.,KA=6.,KE=2.,KR=5.,
&END
&NLDEF class='AIRFRAME',type='DRIVE TRAIN',&END
&NLVAL
    CONFIG=3,OPGOV=1,
    IGEAR=16.,EGEAR=35.,GAINE=12.,
    KRS=2*750000.,IRS=2*2.,KIS=5000.,IIS=2.,
    KES=12000.,IENG=.1,DENG=.2,
    KIGOV=.02,KPGOV=0.,KRGOV=2*1.,KEGOV=0.,
    WGOV=6.2832,ZGOV=.7,
&END
!===============================================================================
&NLDEF class='TABLES',&END
&NLVAL &END
!===============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
!  Tilting Proprotor Aircraft Right Rotor
!==============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    MASSR(1)=10.8,                                          ! inertia
    HSP(1)=1.5,OPSPM=0,                                     ! control system
&END
!==============================================================================
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
&NLVAL                                                      ! description
    TITLE='SAMPLE JOBS:  TILTING PROPROTOR AIRCRAFT RIGHT ROTOR',
    RADIUS=15.,NBLADE=3,ROTATE=1,SIGMA=.10,
    VTIPN=600.,
    GIMBAL=1,HINGE=0,                                       ! blade root
    CONE=1.5,EPITCH=.09,KGMBL=15000.,
    CONTRL=2,PITCH=1,KPITCH=0.,LOCKP=1,                     ! control system
    XSP=.06,YSP=.016,ZSP=-.1,
    XPH=.06,YPH=.016,ZPH=.03,
    EPH=.11,KPL=25000.,LOCKPL=1,LOCKSP=0,
    OPBEAM=2,DRELST=.04,KNODE=2,RNODE=.35,.64,              ! elastic blade
    NSEN=2,QUANT=2*4,RLOAD=.05,.35,
    NPROP=3,RPROP=0.,.5,1.,                                 ! section properties
    ZEA=3*0.,ZQC=3*0.,ZC=3*0.,ZI=3*0.,
    XEA=3*0.,XQC=3*0.,XC=3*0.,XI=3*0.,
    TWISTA=34.5,7.,-7.,THETAC=34.5,7.,-7.,THETAI=34.5,7.,-7.,
    KP=3*.019,KT=3*.019,EA=3*250000000.,
    EIFLAP=3*1500000.,EILAG=3*20000000.,GJ=3*200000.,
    MASS=3*.24,ITHETA=3*.02,IPOLAR=3*.02,
&END
!==============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
&NLVAL
    NPANEL=20,                                              ! aero panels
    REDGE=.12,.20,.28,.35,.42,.48,.54,.59,.64,.69,.73,
          .77,.81,.84,.87,.90,.92,.94,.96,.98,1.0,
    NPROP=11,RPROP=0.,.1,.2,.3,.4,.5,.6,.7,.8,.9,1.,        ! aero properties
    CHORD=11*1.5708,
    KSDL=3*.81,.50,.33,.23,.16,.10,.07,.04,.01,
    KSDD=3*.54,.34,.20,.12,.06,.02,.00,.00,.00,
    NSEN=9,OPREF=9*4,                                       ! aerodynamic sensors
    QUANT= 5,25,31,32,33,35,71,82,82,                       !   lambda,alpha
    IDENT= 1, 0, 0, 0, 0, 0, 0, 0, 0,                       !   uT,uR,uP,theta
    AXIS=  3, 0, 0, 0, 0, 0, 0, 1, 3,                       !   L,Fx,Fz
    OPSCL= 2, 1, 2, 2, 2, 1, 2, 2, 2,                       !
    NAPLOT=1, 4, 1, 1, 1, 1, 1, 0, 0,                       !
&END
!==============================================================================
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 1',&END
&NLVAL
    KHLMDA=1.085,KFLMDA=2.,FMLMDA=2*0.,
    KINTFR(2)=-.085,KINTHW=1.5,KINTFH=1.8,
&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL OPSCEN=2,TWIST=-28.,RICWG=.3,&END                   ! hover wake
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
&NLVAL OPSCEN=0,RNW=.25,WKMODL=8*2,&END
!==============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
!  Tilting Proprotor Aircraft Left Rotor
!===============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    MASSR(2)=10.8,                                        ! inertia
    HSP(2)=1.5,OPSPM=0,                                   ! control system
&END
!===============================================================================
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 2',&END
&NLVAL                                                    ! description
    TITLE='SAMPLE JOBS:  TILTING PROPROTOR AIRCRAFT LEFT ROTOR',
    RADIUS=15.,NBLADE=3,ROTATE=-1,SIGMA=.10,
    VTIPN=600.,
    GIMBAL=1,HINGE=0,                                     ! blade root
    CONE=1.5,EPITCH=.09,KGMBL=15000.,
    CONTRL=2,PITCH=1,KPITCH=0.,LOCKP=1,                   ! control system
    XSP=.06,YSP=.016,ZSP=-.1,
    XPH=.06,YPH=.016,ZPH=.03,
    EPH=.11,KPL=25000.,LOCKPL=1,LOCKSP=0,
    OPBEAM=2,DRELST=.04,KNODE=2,RNODE=.35,.64,            ! elastic blade
    NSEN=2,QUANT=2*4,RLOAD=.05,.35,
    NPROP=3,RPROP=0.,.5,1.,                               ! section properties
    ZEA=3*0.,ZQC=3*0.,ZC=3*0.,ZI=3*0.,
    XEA=3*0.,XQC=3*0.,XC=3*0.,XI=3*0.,
    TWISTA=34.5,7.,-7.,THETAC=34.5,7.,-7.,THETAI=34.5,7.,-7.,
    KP=3*.019,KT=3*.019,EA=3*250000000.,
    EIFLAP=3*1500000.,EILAG=3*20000000.,GJ=3*200000.,
    MASS=3*.24,ITHETA=3*.02,IPOLAR=3*.02,
&END
!===============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 2',&END
&NLVAL
    NPANEL=20,                                            ! aero panels
    REDGE=.12,.20,.28,.35,.42,.48,.54,.59,.64,.69,.73,
          .77,.81,.84,.87,.90,.92,.94,.96,.98,1.0,
    NPROP=11,RPROP=0.,.1,.2,.3,.4,.5,.6,.7,.8,.9,1.,      ! aero properties
    CHORD=11*1.5708,
    KSDL=3*.81,.50,.33,.23,.16,.10,.07,.04,.01,
    KSDD=3*.54,.34,.20,.12,.06,.02,.00,.00,.00,
    NSEN=9,OPREF=9*4,                                     ! aerodynamic sensors
    QUANT= 5,25,31,32,33,35,71,82,82,                     !   lambda,alpha
    IDENT= 1, 0, 0, 0, 0, 0, 0, 0, 0,                     !   uT,uR,uP,theta
    AXIS=  3, 0, 0, 0, 0, 0, 0, 1, 3,                     !   L,Fx,Fz
    OPSCL= 2, 1, 2, 2, 2, 1, 2, 2, 2,                     !
    NAPLOT=1, 4, 1, 1, 1, 1, 1, 0, 0,                     !
&END
!===============================================================================
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR 2',&END
&NLVAL
    KHLMDA=1.085,KFLMDA=2.,FMLMDA=2*0.,
    KINTFR(1)=-.085,KINTHW=1.5,KINTFH=1.8,
&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 2',&END
&NLVAL OPSCEN=2,TWIST=-28.,RICWG=.3,&END                 ! hover wake
&NLDEF class='ROTOR',type='WAKE',name='ROTOR 2',&END
&NLVAL OPSCEN=0,RNW=.25,WKMODL=8*2,&END
!===============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!  ELEMENTARY ROTOR IN WIND TUNNEL
#$!  hover, no trim; rotor stability and response
#$!  flutter time history; transient time history
#$!  flap motion, rigid blade and rigid wing; dynamic inflow
#$!  elementary scenarios
#$!
setenv BLADEAIRFOIL1 $airfoil/naca0012.tab
setenv SHELLINPUT $input/elementary.bin
setenv PLOTFILE $jobs/sample1.plot
$camrad/camradii > $jobs/sample1.out << 'endofinput'
 &NLJOB NCASES=3,PLFILE=1,OPSHLL=2,&END
 !==============================================================================
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=1,CODE='RESPONSE',&END                        ! flutter; description
 &NLDEF class='TRIM',&END
 &NLVAL
     WVEL=0.,COLL=5.,                                        ! operating condition
     MHARMR=0,                                               ! hover
 &END
 !==============================================================================
 &NLDEF class='FLUTTER',&END
 &NLVAL
     MPSIAV=1,                                               ! no average
     TASK=1,0,1,0,MFREQ=1,                                   ! frequency response
 &END
 &NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
 &NLVAL
     CONR=3*1,                                               ! rotor controls
     MPSEN=1,                                                ! output
     OPWAKE=4,DOFL=3*0,                                      ! trim inflow
     OPWAKE=1,DOFL=3*0,                                      ! uniform inflow
     OPWAKE=3,DOFL=3*2,                                      ! quasistatic inflow
     OPWAKE=3,DOFL=3*1,                                      ! dynamic inflow
 &END
 !==============================================================================
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !  reduced output
 &NLDEF class='FLUTTER',&END
 &NLVAL OPSTEP=0,KPRNT(3)=0,OPSCL(3)=5,&END                  ! output
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !==============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
 !##############################################################################
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=1,CODE='TIME HISTORY',&END                    ! flutter; description
 &NLDEF class='TRIM',&END
 &NLVAL
     WVEL=0.,COLL=5.,                                        ! operating condition
     MHARMR=0,                                               ! hover
 &END
 !==============================================================================
 &NLDEF class='FLUTTER',&END
 &NLVAL
     MPSIAV=1,                                               ! no average
     GUST=2*0,1                                              ! vertical gust
     TASK=1,1,0,0,KPRNT(2)=1,                                ! time history
     OPHIST=7,TPER=.2,TIMEB=.1,TSTEP=.02,TMAX=.4,TOUT=.02,MHIST=0,
     NLPRNT=0,NMPRNT=0,                                      ! matrices not printed
```

```
&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL
    CONR=3*0,
    MHSEN=1,MPSEN=1,MASEN=1,                             ! output
    DOFMBC=1,5*0,SENMBC=1,5*0,                           ! collective motion
    OPWAKE=4,DOFL=3*0,                                   ! trim inflow
    OPWAKE=1,DOFL=3*0,                                   ! uniform inflow
    OPWAKE=3,DOFL=2,2*0,                                 ! quasistatic inflow
    OPWAKE=3,DOFL=1,2*0,                                 ! dynamic inflow
&END
!==============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
&NLVAL
    NSEN=2,OPREF=2*4,                                    ! aerodynamic sensors
    QUANT =5,25,                                         !   induced velocity
    IDENT =0, 0,                                         !   angle of attack
    AXIS  =3, 0,
    OPSCL =2, 1,
    NAPLOT=1, 1,
&END
!==============================================================================
!%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
!  reduced output
&NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
&NLVAL MHSEN=0,MASEN=0,&END                              ! output
!%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
!==============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
!##############################################################################
&NLDEF class='CASE',&END
&NLVAL TNTASK=1,CODE='TIME HISTORY',&END                 ! transient; desc
&NLDEF class='TRIM',&END
&NLVAL
    WVEL=0.,COLL=5.,                                     ! operating condition
    MHARMR=0,                                            ! hover
&END
&NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
&NLVAL
    MHSEN=1,NHPRNT=1,NHFILE=1,MHTIME=1,                  ! output
    MPSEN=1,NPPRNT=1,NPFILE=1,MPTIME=1,                  ! output
    MASEN=1,NAPRNT=1,NAFILE=1,MATIME=1,                  ! output
&END
!==============================================================================
&NLDEF class='TRANSIENT',&END
&NLVAL
    WGUST=20.,                                           ! vertical gust
    TIMEB=0.,TIMEE=.5,OPHIST=7,TPER=.2,TIMEZ=.1,         ! time history
    TRACEP=2,
&END
&NLDEF class='TRANSIENT ROTOR',name='ROTOR 1',&END
&NLVAL
    MHSEN=1,MPSEN=1,MASEN=1,                             ! output
    DOFMBC=6*1,                                          ! degrees of freedom
    OPVATR=2,OPVRTA=2,                                   ! no rotor/body int
    OPWAKE=1,                                            ! uniform inflow
    OPWAKE=4,                                            ! trim inflow
    OPWAKE=3,                                            ! dynamic inflow
&END
!==============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
&NLVAL
    NSEN=2,OPREF=2*4,                                    ! aerodynamic sensors
    QUANT =5,25,                                         !   induced velocity
```

```
     IDENT =0, 0,                                          !   angle of attack
     AXIS  =3, 0,
     OPSCL =2, 1,
     NAPLOT=1, 1,
 &END
 !===============================================================================
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !  reduced output
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL MHSEN=0,MASEN=0,&END                               ! output
 &NLDEF class='TRANSIENT ROTOR',name='ROTOR 1',&END
 &NLVAL MHSEN=0,MASEN=0,&END                               ! output
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !===============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!  TEETERING ROTOR IN WIND TUNNEL
#$!  hover, axisymmetric trim; hover wake, free geometry
#$!  periodic coefficient system stability
#$!  teetering rotor; rigid blade and rigid wing
#$!
setenv BLADEAIRFOIL1 $airfoil/naca0012.tab
setenv SHELLINPUT $input/teeter.bin
setenv PLOTFILE $jobs/sample2.plot
$camrad/camradii > $jobs/sample2.out << 'endofinput'
 &NLJOB PLFILE=1,&END
 !===============================================================================
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=1,CODE='WAKE AND STABILITY',&END           ! flutter; description
 &NLDEF class='TRIM',&END
 &NLVAL
     WVEL=0.,                                              ! operating condition
     LEVEL=3,NWPRNT=1,                                     ! wake loop
     COLL=8.3,CTTRIM=.08,                                  ! wind tunnel trim
     TOLERC=.2,ITERF=4,RELAXF=.5,                          ! hover free wake
 &END
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL MASEN=1,MWSEN=1,&END                               ! output
 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
 &NLVAL NAPLOT=5*2,MSPAN=25,&END                           ! output
 !===============================================================================
 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',action='init',&END
 &NLVAL OPSCEN=2,TWIST=-10.,RICWG=.20,&END                 ! hover wake
 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
 &NLVAL OPSCEN=0,&END
 !===============================================================================
 &NLDEF class='FLUTTER',&END
 &NLVAL OPFLUT=1,OPMEAN=0,MSTEP=100,&END                   ! periodic system
 !===============================================================================
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !  reduced output
 &NLDEF class='TRIM',&END
 &NLVAL NWPRNT=0,&END                                      ! wake loop
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL MWSEN=0,&END                                       ! output
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !===============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!   ROTOR IN WIND TUNNEL
#$!   forward flight trim; low speed wake
#$!   rigid blade and rigid wing (from elastic blade model)
#$!   wind tunnel operation (from free flight model)
#$!
setenv BLADEAIRFOIL1 $airfoil/advtecheq.tab
setenv SHELLINPUT $input/helicopter.bin
setenv PLOTFILE $jobs/sample3.plot
$camrad/camradii > $jobs/sample3.out << 'endofinput'
 &NLJOB PLFILE=1,&END
 !============================================================================
 &NLDEF class='CASE',&END
 &NLVAL CODE='LOW SPEED WAKE',&END                        ! description
 &NLDEF class='TRIM',&END
 &NLVAL
     WINDIN=2,WVEL=.15,                                   ! operating condition
     LEVEL=3,NWPRNT=1,                                    ! wake loop
     CTTRIM=.08,XTRIM=-12.,MTRIM=4,                       ! wind tunnel trim
     MNAME='CT/S    ','BETAS   ','BETAC   ','CX/S    ',
     VNAME='COLL    ','LATCYC  ','LNGCYC  ','PITCH   ',
     COLL=7.06,LATCYC=-3.65,LNGCYC=3.02,PITCH=-1.16,
     RELAXT=.25,                                          ! trim loop
     MHARMR=2*1,MHARMA=2*1,                               ! harmonics
     RELAXC(2)=2*.05,                                     ! convergence
 &END
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL
     OPMODE=0,DOFB=1,0,1,9*0,                             ! degrees of freedom
     MASEN=1,MWSEN=1,                                     ! output
 &END
 !============================================================================
 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
 &NLVAL RFW=3.,MFWG=3,NPOFF=0,&END                        ! wake extent
 !============================================================================
 &NLDEF class='AIRFRAME',type='STRUCTURE',&END
 &NLVAL
     TITLE='WIND TUNNEL SUPPORT',                         ! description
     CONFIG=0,OPFREE=0,OPAERO=0,                          ! wind tunnel
     ASHAFT=0.,ACANT=0.,                                  ! geometry
 &END
 !============================================================================
 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
 &NLVAL OPBEAM=0,KNODE=0,LOCKPL=0,&END                    ! rigid blade
 !============================================================================
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !  reduced output
 &NLDEF class='TRIM',&END
 &NLVAL NWPRNT=0,&END                                     ! wake loop
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!   ROTOR IN WIND TUNNEL
#$!   forward flight trim; high speed wake; rotor stability
#$!   dynamic stall; loads; airframe-to-rotor aerodynamic interference
#$!   swashplate, elastic blade; swept tip
#$!
setenv BLADEAIRFOIL1 $airfoil/advtecheq.tab
setenv BLADESTALL1 $airfoil/dynamicstall.2std
setenv SHELLINPUT $input/windtunnel.bin
setenv PLOTFILE $jobs/sample4.plot
$camrad/camradii > $jobs/sample4.out << 'endofinput'
 &NLJOB PLFILE=1,&END
 !================================================================================
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=1,CODE='HIGH SPEED WAKE',&END              ! flutter; description
 &NLDEF class='TRIM',&END
 &NLVAL
     WVEL=.4,PITCH=-5.,                                   ! operating condition
     LEVEL=2,NWPRNT=1,                                    ! wake loop
     OPTRIM=0,COLL=10.,LATCYC=-3.,LNGCYC=5.,              ! no trim
     RELAXR=.5,                                           ! convergence
 &END
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL MASEN=1,MBSEN=1,MCSEN=1,MHSEN=1,MPSEN=1,&END      ! output
 !================================================================================
 &NLDEF class='FLUTTER',&END
 &NLVAL DOFM=5*0,&END                                     ! degrees of freedom
 !================================================================================
 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
 &NLVAL OPSTLL=2,&END                                     ! dynamic stall
 &NLDEF class='ROTOR',type='WAKE',name='ROTOR 1',&END
 &NLVAL OPFW=2,OPNW=0,&END                                ! dual peak model
 !================================================================================
 &NLDEF class='AIRFRAME',type='STRUCTURE',&END            ! aerodynamic int
 &NLVAL OPAERO=1,&END
 &NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
 &NLVAL VISRC=1,&END
 !================================================================================
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !  reduced output
 &NLDEF class='TRIM',&END
 &NLVAL NWPRNT=0,&END                                     ! wake loop
 &NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
 &NLVAL NMFILE=0,&END                                     ! output
 !  faster
 &NLDEF class='FLUTTER',&END
 &NLVAL MPSIAV=4,&END                                     ! averaged equations
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !================================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!   ROTOR IN WIND TUNNEL
#$!   hover, no trim; ground resonance
#$!   swashplate, elastic blade; swept tip
#$!
setenv BLADEAIRFOIL1 $airfoil/advtecheq.tab
setenv SHELLINPUT $input/windtunnel.bin
setenv PLOTFILE $jobs/sample5.plot
$camrad/camradii > $jobs/sample5.out << 'endofinput'
 &NLJOB PLFILE=1,&END
 !===============================================================================
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=1,CODE='GROUND RESONANCE',&END             ! flutter, description
 &NLDEF class='TRIM',&END
 &NLVAL
     WVEL=0.,VTIPIN=2,RPM=200.,                           ! operating condition
     LEVEL=1,                                             ! wake loop
     COLL=4.,OPTRIM=0,                                    ! no trim
     MHARMR=0,MHARMA=0,MPSIAV=1,MPSI=4,                   ! hover
 &END
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL DOFB=3*1,9*0,&END                                 ! degrees of freedom
 !===============================================================================
 &NLDEF class='FLUTTER',&END
 &NLVAL
     MPSIAV=1,                                            ! no average
 &END
 &NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
 &NLVAL
     DOFMBC=0,2*1,3*0,OPMODE=0,DOFB=0,1,10*0,             ! lag only (cyclic)
     DOFMBC=0,2*1,3*0,OPMODE=1,DOFM=4*1,36*0,             ! elastic (cyclic)
 &END
 !===============================================================================
 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
 &NLVAL
     OPAERO=0,                                            ! no aerodynamics
     OPAERO=1,                                            ! aerodynamics
     DLAG=500.,                                           ! lag damping
 &END
 !===============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!  SINGLE MAIN ROTOR AND TAIL ROTOR HELICOPTER
#$!  hover, free flight trim; flight dynamics (stability)
#$!  flutter time history; transient time history
#$!  rigid blade and rigid wing (from elastic model); drive train
#$!
setenv BLADEAIRFOIL1 $airfoil/advtecheq.tab
setenv BLADEAIRFOIL2 $airfoil/naca0012.tab
setenv SHELLINPUT $input/helicopter.bin
setenv PLOTFILE $jobs/sample6.plot
$camrad/camradii > $jobs/sample6.out << 'endofinput'
 &NLJOB NCASES=3,OPINIT=7,PLFILE=1,&END
 !==============================================================================
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=1,CODE='FLIGHT DYNAMICS',&END              ! flutter, description
 &NLDEF class='TRIM',&END
 &NLVAL
     VKTS=0.,                                             ! operating condition
     LEVEL=1,                                             ! wake loop
     COLL=9.81,LATCYC=.51,LNGCYC=-2.42,PEDAL=-8.68,       ! trim
     PITCH=1.57,ROLL=-4.07,THROTL=10.54,
     RELAXT=.25,                                          ! trim loop
 &END
 &NLDEF class='FLUTTER',&END
 &NLVAL
     OPEQN=3*0,6*1,                                       ! equation sets
     DOFORD=5*1,                                          ! first order states
     !#not used# OPSTAB=1,                                ! flight dynamics
     !#not used# NRPRNT=2*2,NAPRNT=2,NDPRNT=2,            ! stability deriv
 &END
 !==============================================================================
 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
 &NLVAL
     OPBEAM=0,KNODE=0,                                    ! rigid blade
     CONTRL=0,PITCH=2,LOCKP=0,                            ! no swashplate
 &END
 !==============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
 !##############################################################################
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=1,CODE='TIME HISTORY',&END                 ! flutter, description
 &NLDEF class='TRIM',&END
 &NLVAL
     VKTS=0.,                                             ! operating condition
     LEVEL=1,                                             ! wake loop
     COLL=9.81,LATCYC=.51,LNGCYC=-2.42,PEDAL=-8.68,       ! trim
     PITCH=1.57,ROLL=-4.07,THROTL=10.54,
     RELAXT=.25,                                          ! trim loop
 &END
 !==============================================================================
 &NLDEF class='FLUTTER',&END
 &NLVAL
     OPEQN=3*0,1,5*0,                                     ! equation sets
     MPSIAV=4,                                            ! loop solution
     CONP=1,4*0,                                          ! collective
     NOUT=1,KINDO=3,NAMEV='PILOT INPUT',NAMEE='COLLECTIVE',
     TASK=1,1,0,0,MHIST=0,                                ! time history
     OPHIST=3,TPER=.6,NCYCLE=1,TIMEB=.1,TOUT=.02,TMAX=1.0,
     MSSEN=1,                                             ! output
```

```
 &END
 &NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
 &NLVAL MHSEN=1,MPSEN=1,&END                              ! output
 &NLDEF class='FLUTTER ROTOR',name='ROTOR 2',&END
 &NLVAL DOFMBC=2,5*0,&END                                 ! quasistatic tail rtr
 !==============================================================================
 &NLDEF class='TRIM',&END                                 ! drive train
 &NLVAL MTRIM=7,MNAME(7)='TORQUE',VNAME(7)='THROTL',&END
 &NLDEF class='FLUTTER',&END
 &NLVAL
     DOFD=1,7*0,
     NOUT=2,KINDO(2)=1,NAMEC(2)='DRIVE TRAIN ROTOR 1 SHAFT',
     NAMEV(2)='DRIVE TRAIN AZIMUTH',NAMEE(2)='1',
 &END
 &NLDEF class='AIRFRAME',type='STRUCTURE',&END
 &NLVAL OPTRAN=1,&END
 !==============================================================================
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !  reduced output
 &NLDEF class='FLUTTER ROTOR',name='ROTOR 1',&END
 &NLVAL MHSEN=0,MPSEN=0,&END                              ! output
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !==============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
 !##############################################################################
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=0,TNTASK=1,CODE='TIME HISTORY',&END        ! transient, desc
 &NLDEF class='TRIM',&END
 &NLVAL
     VKTS=0.,                                             ! operating condition
     LEVEL=1,                                             ! wake loop
     COLL=9.81,LATCYC=.51,LNGCYC=-2.42,PEDAL=-8.68,       ! trim
     PITCH=1.57,ROLL=-4.07,THROTL=10.54,
     RELAXT=.25,                                          ! trim loop
     MSSEN=1,                                             ! output
 &END
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL MHSEN=1,MPSEN=1,&END                              ! output
 !==============================================================================
 &NLDEF class='TRANSIENT',&END
 &NLVAL
     WGUST=20.,                                           ! vertical gust
     TIMEB=0.,TIMEE=1.1,                                  ! time history
     OPHIST=3,TPER=.6,NCYCLE=1,TIMEZ=.1,TSTEPS=.01,
     MSSEN=1,                                             ! output
     TRACEP=2,
 &END
 &NLDEF class='TRANSIENT ROTOR',name='ROTOR 1',&END
 &NLVAL MHSEN=1,MPSEN=1,&END                              ! output
 !==============================================================================
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !  reduced output
 &NLDEF class='TRIM',&END
 &NLVAL NSPRNT=1,NSFILE=1,MSTIME=4,&END                   ! output
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL MHSEN=0,MPSEN=0,&END                              ! output
 &NLDEF class='TRANSIENT ROTOR',name='ROTOR 1',&END
 &NLVAL MHSEN=0,MPSEN=0,&END                              ! output
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !==============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!  SINGLE MAIN ROTOR AND TAIL ROTOR HELICOPTER
#$!  forward flight, free flight trim; forward flight wake
#$!  loads; vibration; inflow off rotor
#$!  swashplate, elastic blade
#$!
setenv BLADEAIRFOIL1 $airfoil/advtecheq.tab
setenv BLADEAIRFOIL2 $airfoil/naca0012.tab
setenv SHELLINPUT $input/helicopter.bin
setenv PLOTFILE $jobs/sample7.plot
$camrad/camradii > $jobs/sample7.out << 'endofinput'
 &NLJOB PLFILE=1,&END
 !===============================================================================
 &NLDEF class='CASE',&END
 &NLVAL CODE='LOADS AND VIBRATION',&END                     ! description
 &NLDEF class='TRIM',&END
 &NLVAL
     VKTS=100.,                                             ! operating condition
     LEVEL=2,                                               ! wake loop
     COLL=8.6,LATCYC=-2.1,LNGCYC=3.5,PEDAL=-3.8,            ! trim
     PITCH=-.5,ROLL=-2.2,
     RELAXT=.25,                                            ! trim loop
     DOFA=6*1,                                              ! degrees of freedom
     RELAXR=.5,                                             ! convergence
     MSSEN=1,                                               ! output
 &END
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL
     OPMODE=1,DOFM=6*1,34*0,                                ! blade modes
     MASEN=1,MBSEN=1,MCSEN=1,MHSEN=1,MVSEN=1,               ! output
 &END
 !===============================================================================
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !  faster
 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
 &NLVAL NPANEL=10,REDGE=.12,.28,.42,.54,.64,.73,.81,.87,.92,.96,1.,&END
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !===============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!  BEARINGLESS ROTOR IN WIND TUNNEL
#$!  hover, no trim; isolated blade stability
#$!  bearingless hub with snubber, swashplate
#$!
setenv BLADEAIRFOIL1 $airfoil/advtecheq.tab
setenv SHELLINPUT $input/bearingless.bin
setenv PLOTFILE $jobs/sample8.plot
$camrad/camradii > $jobs/sample8.out << 'endofinput'
 &NLJOB PLFILE=1,&END
 !===============================================================================
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=1,CODE='STABILITY',&END                        ! flutter; description
 &NLDEF class='TRIM',&END
 &NLVAL WKTS=0.,COLL=10.,&END                                 ! operating condition
 !===============================================================================
 &NLDEF class='FLUTTER',&END
 &NLVAL OPBLD=1,&END                                          ! independent blade
 !===============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!  TANDEM HELICOPTER
#$!  forward flight, free flight trim; nonuniform inflow
#$!  rotor-rotor interference
#$!  rigid blade and rigid wing; auxiliary force
#$!
setenv BLADEAIRFOIL1 $airfoil/advtecheq.tab
setenv SHELLINPUT $input/tandem.bin
setenv PLOTFILE $jobs/sample9.plot
$camrad/camradii > $jobs/sample9.out << 'endofinput'
 &NLJOB PLFILE=1,&END
 !===============================================================================
 &NLDEF class='CASE',&END
 &NLVAL CODE='TANDEM',&END                                ! description
 !===============================================================================
 &NLDEF class='TRIM',&END
 &NLVAL
     VKTS=100.,                                           ! operating condition
     LEVEL=2*3,NWPRNT=0,                                  ! wake loop
     COLL=9.4,LATCYC=-0.6,LNGCYC=2.2,PEDAL=0.3,           ! uniform inflow trim
     PITCH=-2.4,ROLL=0.3,
     COLL=8.1,LATCYC=-0.1,LNGCYC=1.2,PEDAL=0.7,           ! free wake trim
     PITCH=-1.9,ROLL=0.6,
     RELAXT=.25,OPPID=2,                                  ! trim loop
     ITERC=100,
     CNTRLF=1500.,                                        ! auxiliary propulsion
 &END
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL MASEN=1,MWSEN=1,&END                              ! output
 &NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
 &NLVAL MASEN=1,MWSEN=1,&END                              ! output
 !===============================================================================
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !  reduced output
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL MWSEN=0,&END                                      ! output
 &NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
 &NLVAL MWSEN=0,&END                                      ! output
 !  faster
 &NLDEF class='TRIM',&END
 &NLVAL OPTRIM=0,&END                                     ! no trim
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !===============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!  TILTROTOR IN WIND TUNNEL
#$!  axial flow, axisymmetric trim; tiltrotor stability
#$!  gimballed hub, swashplate, elastic blade; drive train
#$!
setenv BLADEAIRFOIL1 $airfoil/advtecheq.tab
setenv SHELLINPUT $input/tiltrotorwt.bin
setenv PLOTFILE $jobs/sample10.plot
$camrad/camradii > $jobs/sample10.out << 'endofinput'
 &NLJOB PLFILE=1,&END
 !===============================================================================
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=1,CODE='STABILITY',&END                    ! flutter; description
 &NLDEF class='TRIM',&END
 &NLVAL
     WKTS=300.,                                           ! operating condition
     LEVEL=1,                                             ! wake loop
     CPTRIM=0.,COLL=53.77,                                ! wind tunnel trim
     MTRIM=1,MNAME='CP/S    ',VNAME='COLL    ',
     RELAXT=.25,                                          ! trim loop
     TOLERC=.05,                                          ! circulation loop
     RELAXR=.5,                                           ! convergence
     MPSI=4,                                              ! uniform inflow only
 &END
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL MBSEN=1,&END                                      ! output
 !===============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

```
#!/bin/csh
set camrad='/home/camrad/'
set airfoil='/home/camrad/sample/airfoil'
set input='/home/camrad/sample/input'
set jobs='/home/camrad/sample/jobs'
#
#$!  TILTING PROPROTOR AIRCRAFT
#$!  airplane mode, symmetric trim; tiltrotor stability
#$!  airframe-to-rotor aerodynamic interference
#$!  vibration; higher harmonic control
#$!  gimballed hub, swashplate, rigid blade; drive train
#$!
#$!
setenv BLADEAIRFOIL1 $airfoil/advtecheq.tab
setenv SHELLINPUT $input/tiltrotor.bin
setenv PLOTFILE $jobs/sample11.plot
$camrad/camradii > $jobs/sample11.out << 'endofinput'
 &NLJOB PLFILE=1,&END
 !===============================================================================
 &NLDEF class='CASE',&END
 &NLVAL FLTASK=1,CODE='VIBRATION/STABILITY',&END          ! flutter; description
 &NLDEF class='TRIM',&END
 &NLVAL
     VKTS=250.,                                           ! operating condition
     LEVEL=2*1,                                           ! wake loop
     GOV=24.32,LNGCYC=.66,PITCH=.26,                      ! trim
     RELAXT=.25,                                          ! trim loop
     RELAXR=2*.5,                                         ! convergence
     MSSEN=1,                                             ! output
 &END
 !===============================================================================
 &NLDEF class='TRIM ROTOR',name='ROTOR 1',&END
 &NLVAL
     MHSEN=1,MASEN=1,MPSEN=1,                             ! output
     MHHCN=1,TCHHC=-.2,TCHHS=-.3,TSHHC=.3,TSHHS=-.1,      ! higher harmonic
 &END
 &NLDEF class='TRIM ROTOR',name='ROTOR 2',&END
 &NLVAL
     MHHCN=1,TCHHC=-.2,TCHHS=-.3,TSHHC=.3,TSHHS=-.1,      ! higher harmonic
 &END
 !===============================================================================
 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 1',&END
 &NLVAL OPBEAM=0,KNODE=0,LOCKPL=0,&END                    ! rigid blade and wing
 &NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR 2',&END
 &NLVAL OPBEAM=0,KNODE=0,LOCKPL=0,&END                    ! rigid blade and wing
 !===============================================================================
 &NLDEF class='AIRFRAME',type='AERODYNAMICS',&END
 &NLVAL VISRC=2*1,&END                                    ! aerodynamic int
 !===============================================================================
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !  faster
 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 1',&END
 &NLVAL NPANEL=2,REDGE=.3,.7,1.,&END                      ! two wing panels
 &NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR 2',&END
 &NLVAL NPANEL=2,REDGE=.3,.7,1.,&END                      ! two wing panels
 !%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 !===============================================================================
 &NLDEF action='end of shell',&END
 &NLDEF action='end of core',&END
'endofinput'
```

2–12 CAMRAD II Jobs For Training

RUN SAMPLE JOB

OBJECTIVES:
- establish access to program executables
- establish approach to handling files and directories
- become familiar with basic use of programs

COPY SAMPLES/EXTRAS TO USER'S DIRECTORY

CHOOSE A SAMPLE JOB
- SAMPLE8 is fast

RUN INPUT PROGRAM TO GENERATE AIRFOIL FILE
- one of jobs in `AIRFOIL.UNIX` (revise directories)
- interactively run INPUT program to examine or plot airfoil file data

RUN INPUT PROGRAM TO GENERATE SHELL INPUT FILE
- one of jobs in `AINPUT.UNIX` (revise directories)
- interactively run INPUT program to examine, plot, or draw input file data

RUN CAMRADII PROGRAM
- one of jobs in `JOBS.UNIX` (revise directories)
- examine output file

INTERACTIVELY RUN OUTPUT PROGRAM TO PLOT OR DRAW PLOTFILE DATA
- can use existing plot file from sample jobs (SAMPLE3 has free wake geometry)

IDENTIFY ROTORCRAFT FOR INITIAL CALCULATIONS

OBJECTIVES:

simple enough for efficient progress during training,
representative of actual near-term calculations

typically single main rotor and tail rotor configuration;
rotor only (wind tunnel) and complete helicopter

PREPARE INPUT

GENERATE INPUT DATA

namelist files for main rotor, tail rotor, airframe, wind tunnel

input file for wind tunnel (from wind tunnel, main rotor namelists)

input file for helicopter (from airframe, main rotor, tail rotor namelists)

RESOURCES: `ztemplate.list`, `helicopter.list`, `windtunnel.list` (and other sample job input)

CREATE AIRFOIL FILES

for main rotor and tail rotor

RESOURCES: sample job input

CHECK INPUT DATA

run INPUT program to plot (2D) and draw (3D) input file and airfoil file data

RESOURCES: `zplanform.com` (script file for 2D plot of planform)

RUN CALCULATIONS

BLADE FREQUENCIES

frequencies in vacuum, for varying rotor speed

RESOURCES: `zfrequency.list`

STABILITY

rotor only, hover
rotor only, forward flight
rotor and airframe (ground resonance)
rotor and airframe (free flight)

RESOURCES: `sample6`, `sample5`

HOVER PERFORMANCE

rotor only (axisymmetric)
helicopter (not axisymmetric)

RESOURCES: `zhover.list`, `sample2`

FORWARD FLIGHT PERFORMANCE, LOADS, VIBRATION

rotor only (wind tunnel)
- trim rotor lift, propulsive force, flapping; or
- fixed collective and shaft angle, trim flapping

helicopter
- trim aircraft force and moment
- trim aircraft force and moment, specify
  - climb or descent angle
- trim aircraft force and moment and power

RESOURCES: `sample4`, `sample7`

MAXIMUM THRUST

rotor only (wind tunnel)
collective sweep with zero shaft angle, trim flapping

helicopter
turn rate sweep, trim aircraft force and moment
(and maybe power)

GUST RESPONSE OR MANEUVER

vertical gust input or longitudinal cyclic input
flutter task (linear response) and transient task (nonlinear)
helicopter, forward flight

RESOURCES: `sample6`

Chapter 3

TABLE FORMATS

GENERAL REFERENCES FOR CHAPTER:

CAMRAD II Documentation, Volume VI, Rotorcraft input; Chapter “Table Formats”

CAMRAD II Documentation, Volume I, Theory; Chapter “Tables”

3–1 Table Class and Type

TABLES CORRESPOND TO FILES

PROVIDING GENERAL FILE INPUT CAPABILITY

EACH TABLE CLASS/TYPE HAS SPECIFIC FILE FORMAT AND DATA ORGANIZATION

CLASS = MATRIX
table data function of discrete variables, not interpolated;
initialization moves table data to record array for use

CLASS = UNSTRUCTURED
table data unstructured set of numbers, not interpolated;
use of table data by analysis depends on component

OTHER CLASSES
table data function of continuous variables, interpolated from data vector

COMPONENT IDENTIFIES TABLE BY CLASS, TYPE, AND NAME

USER (OR ROTORCRAFT SHELL) DEFINES NAME

names must be unique with class, even if type is different

TABLE SPECIFICATION ALSO INCLUDES FILE UNIT NUMBER, AND FILE NAME OR LOGICAL NAME

file name or logical name can be same as name in class/type/name specification (this is the default)

for most installations, can ignore unit numbers

FIGURE 1 LISTS THE TABLE CLASSES AND TYPES IMPLEMENTED

INPUT PROGRAM CAN PREPARE CLASS = AIRFOIL FROM SOURCE FILES OR INTERNALLY

for other table classes, table file preparation does not transform the table data

CLASS	TYPE	NAME
TWO DIMENSIONAL	STANDARD	
THREE DIMENSIONAL	STANDARD	
FOUR DIMENSIONAL	STANDARD	
AIRFOIL	STANDARD	
AIRFOIL	C81	
AIRFOIL	CAMRAD	
MATRIX	STANDARD	
UNSTRUCTURED	REAL	
UNSTRUCTURED	INTEGER	

Lower case is accepted for class and type. Unique initial letters of class or type are accepted. Specific abbreviations and alternatives:

class = `TWO DIMENSIONAL` = `2D`
class = `THREE DIMENSIONAL` = `3D`
class = `FOUR DIMENSIONAL` = `4D`

class = `AIRFOIL`:
  type = `C81` = `FIXED FORMAT`
  type = `CAMRAD` = `WING`

class = `MATRIX` = `ARRAY`

Figure 3-1 Table blocks.

3–2 Table File Formats

TABLE FILE CAN BE FORMATTED OR UNFORMATTED

FORMATTED FILE CAN BE FIXED FORM OR FREE FORM

FREE FORM:
each line consists of number of fields, separated by blanks or commas or tabs

maximum 132 characters in line
maximum 20 fields in line
maximum 40 characters in each field

ignore any equals sign and part of field to its left

so format "name = value" can be used

comments (exclamation mark and what follows on line) are ignored, except for title and identification lines

FIXED FORM:
data found in fields of specific width and position

GENERAL FORMAT CONVENTIONS

EACH TABLE HAS TITLE, AND FILE HAS IDENTIFICATION, WHICH SHOULD BE UNIQUE

INDEPENDENT AND DEPENDENT VARIABLES MAY HAVE LABELS

maximum 80 characters in title
maximum 20 characters in file identification
maximum 8 characters in labels

blank lines typically interpreted as zeros and blank labels

END OF EACH TABLE HAS VALUES OF REFERENCE QUANTITIES

3–3 Evaluation of Dependent Variables

TABLE QUANTITIES ARE FUNCTIONS OF CONTINUOUS VARIABLES, ACCESSED BY INTERPOLATION

EXCEPTION: CLASS = MATRIX or UNSTRUCTURED

LINEAR INTERPOLATION, WITH NO EXTRAPOLATION BEYOND RANGE OF TABLE

OPTIONAL CYCLICAL OR ANGLE INTERPOLATION FOR INDEPENDENT VARIABLE α

α is angle in degrees

table range typically 0 to 360 degrees, or -180 to 180 degrees

multiple of 360 degrees added to or subtracted from α, until value within table range

then dependent variable interpolated

with no extrapolation, in case table range does not extend full 360 degrees

FOR THREE-DIMENSIONAL TABLE, LAST VARIABLE γ CAN BE SEARCHED OR INTERPOLATED

for interpolation, γ_k defines value for k-th two-dimensional array

for search, γ_k defines beginning of range of k-th two-dimensional array

INDEPENDENT VARIABLES CAN HAVE UNIFORM INCREMENTS

increment determined from first two values of independent variable (but all values must still be present in table)

3–4 Summary of Tables

CLASS = TWO DIMENSIONAL, TYPE = STANDARD

SET OF $C_\gamma(\alpha, \beta)$

one or more 2D coefficients, identified by label γ
separate rectangular (α, β) array for each
α and β can be cyclical, and uniform

FORMATTED, FREE FORM

CLASS = THREE DIMENSIONAL, TYPE = STANDARD

$C(\alpha, \beta, \gamma)$

3D coefficient, arranged as separate rectangular (α, β) arrays for set of γ values

α and β and γ can be cyclical, and uniform
γ can be searched or interpolated

FORMATTED, FREE FORM

CLASS = FOUR DIMENSIONAL, TYPE = STANDARD

$C(\alpha, \beta, \gamma, \delta)$

4D coefficient, arranged as separate rectangular (α, β) arrays for set of (γ, δ) values

same (α, β) values for all (γ, δ), same γ values for all δ
α and β and γ and δ can be cyclical, and uniform

FORMATTED, FREE FORM

CLASS = MATRIX, TYPE = STANDARD

SET OF $C_M(i, j)$
one or more 2D matrices or arrays, identified by label or number M

i and j can be uniform
no interpolation

FORMATTED, FREE FORM

CLASS = UNSTRUCTURED, TYPE = REAL or INTEGER

SET OF $T(i)$
real or integer numbers, counted as read, stored in record
no interpolation

FORMATTED, FREE FORM

CLASS = AIRFOIL, TYPE = STANDARD

$c_\ell(\alpha, M)$, $c_d(\alpha, M)$, $c_m(\alpha, M)$;
$c_{\ell f}(\alpha, M)$, $c_{df}(\alpha, M)$, $c_{mf}(\alpha, M)$
lift, drag, moment coefficients (any order)
and perhaps flap coefficients, for one flap angle

function of angle of attack α, and Mach number M

separate (α, M) array for each coefficient
α can be row or column, α cyclical

FORMATTED, FREE FORM

CLASS = AIRFOIL, TYPE = C81

$c_\ell(\alpha, M)$, $c_d(\alpha, M)$, $c_m(\alpha, M)$;
$c_{\ell f}(\alpha, M)$, $c_{df}(\alpha, M)$, $c_{mf}(\alpha, M)$

lift, drag, moment coefficients (in this order)
and perhaps flap coefficients, for one flap angle

function of angle of attack α, and Mach number M

separate (α, M) array for each coefficient
α must be row, α cyclical

FORMATTED, FIXED FORM

CLASS = AIRFOIL, TYPE = CAMRAD

$c_\ell(\alpha, M, \phi, r)$, $c_d(\alpha, M, \phi, r)$, $c_m(\alpha, M, \phi, r)$;
$c_{\ell f}(\alpha, M, \phi, r)$, $c_{df}(\alpha, M, \phi, r)$, $c_{mf}(\alpha, M, \phi, r)$

lift, drag, moment coefficients
and perhaps flap coefficients

function of angle of attack α, Mach number M,
flap angle ϕ, and span station r

special (α, M) array for efficient interpolation
α cyclical, r can be searched or interpolated

CONSTRUCTED FROM SET OF AIRFOIL TABLES (STANDARD OR C81)

UNFORMATTED

3–5 Class = Airfoil

AIRFOIL TABLES CONTAIN LIFT COEFFICIENT, DRAG COEFFICIENT, AND MOMENT COEFFICIENT DATA

AS FUNCTION OF ANGLE OF ATTACK α AND MACH NUMBER M

two-dimensional airfoil data: $c_\ell(\alpha, M)$, $c_d(\alpha, M)$, $c_m(\alpha, M)$

angle of attack is in range -180 to 180 degrees
angle of attack is cyclical

LOADS AND ANGLE OF ATTACK DEFINED AS IN TWO-DIMENSIONAL AIRFOIL TEST

α varied by pitching airfoil
lift positive upward
drag positive in direction of free stream velocity
moment positive nose up

figure 2 shows the conventions

figure 3 illustrates expected behavior of airfoil characteristics

moments measured about reference axis
usually this axis is at 25% chord
α measured to reference chord line in airfoil
usually not zero-lift line

component that uses airfoil data must have consistent definition of reference chord line and reference axis

consistent definition of blade twist required

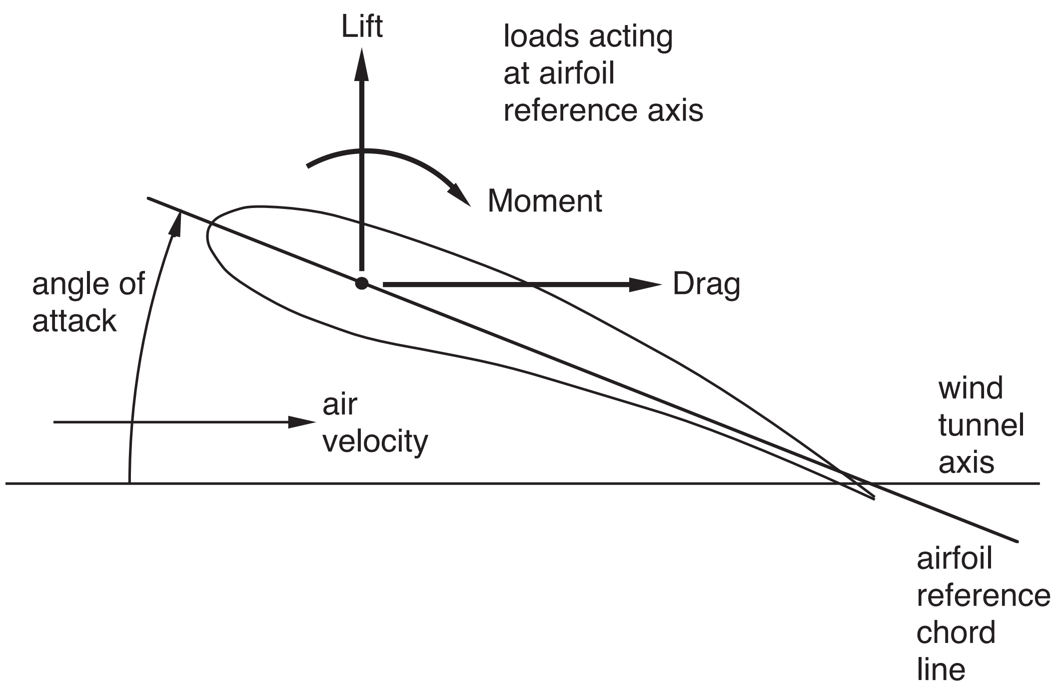


Figure 3-2 Conventions for airfoil loads and angle of attack.

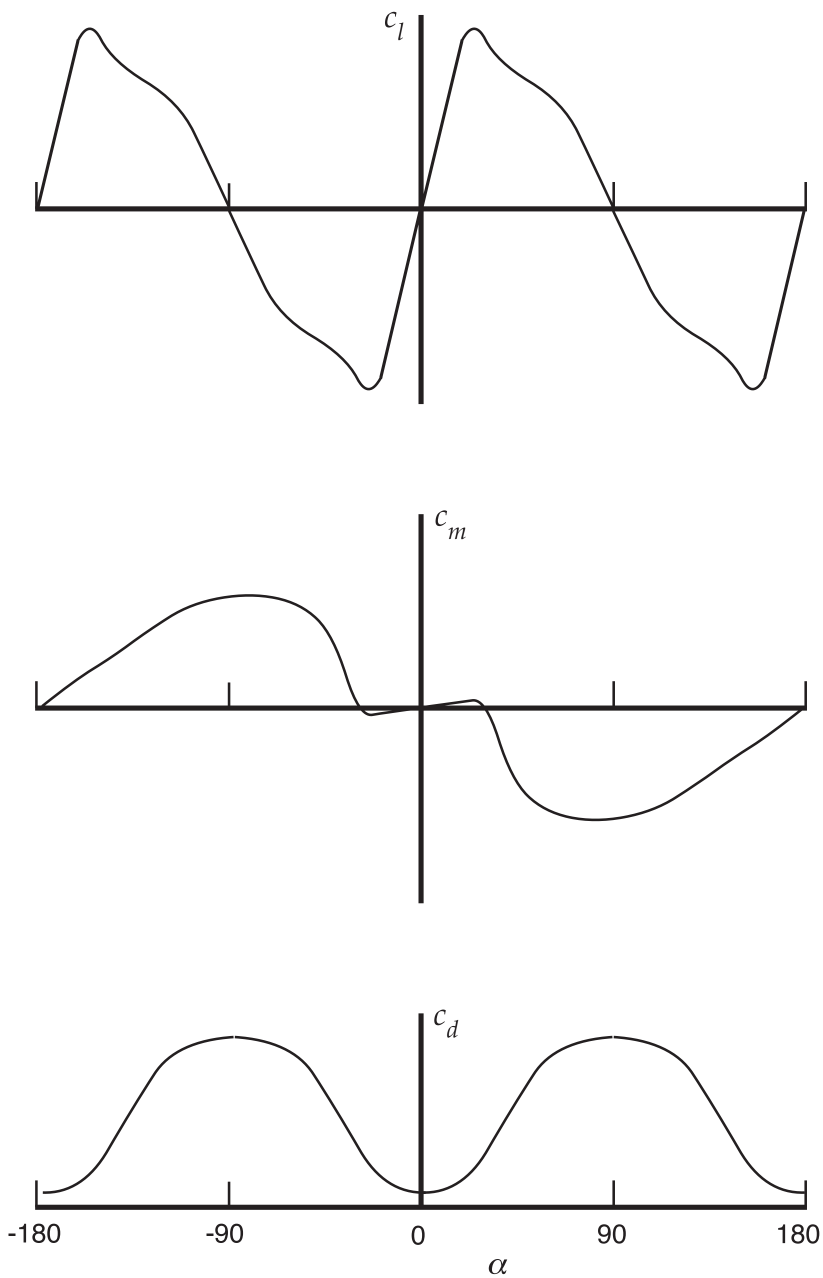


Figure 3-3 Sketch of two-dimensional airfoil characteristics.

FIRST REFERENCE VALUE IS REYNOLDS NUMBER Re_1 (DEFAULT VALUE OF ZERO)

this is Reynolds number of table corresponding to Mach number $M = 1$

Reynolds number at arbitrary Mach number will then be calculated as $Re = M\,Re_1$

figure 4 illustrates the approximation

this linear relation is usually an approximation to true relation between Reynolds number and Mach number for airfoil data

SYNTHESIZED AIRFOIL DATA

WHEN DATA ARE NOT AVAILABLE FOR REQUIRED AIRFOIL

CHARACTERISTICS CAN BE SYNTHESIZED FROM EQUATIONS THAT REPRESENT TYPICAL AIRFOIL BEHAVIOR

equations can also be used to generate simplified airfoil characteristics

PARAMETERS OF THESE EQUATIONS ARE INPUT THROUGH NAMELIST NLEQN

REFERENCE FOR AIRFOIL EQUATIONS: CAMRAD II Documentation, Volume I, Theory; Chapter “Tables”

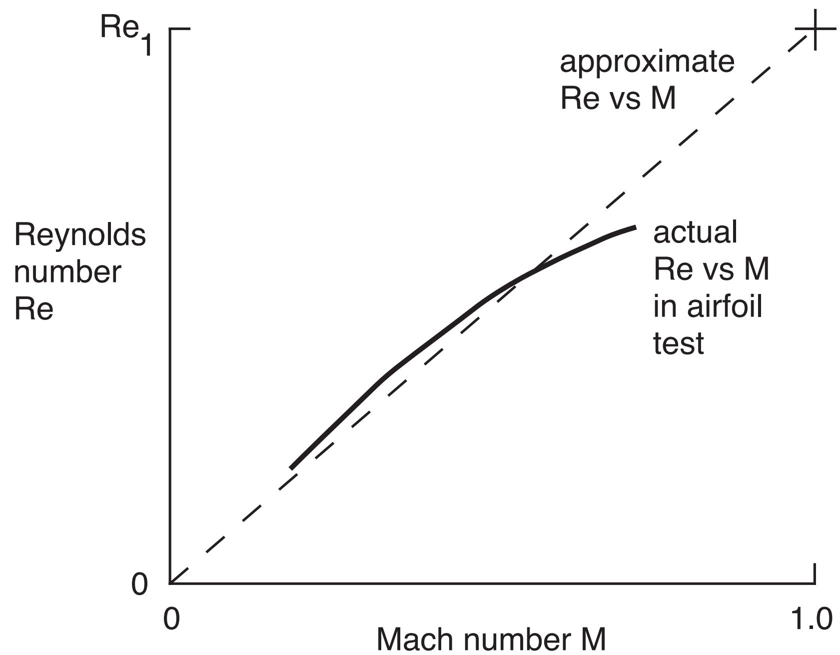


Figure 3-4 Relation between Reynolds number and Mach number.

AIRFOIL CAN HAVE TRAILING EDGE FLAP

coefficients are functions of flap angle ϕ, as well as angle of attack and Mach number

tables include coefficients of lift, drag, and moment on flap: $c_{\ell f}$, c_{df}, c_{mf}

c_ℓ, c_d, c_m are total loads on airfoil, including flap loads

second reference value is the flap angle ϕ

TYPE = STANDARD OR C81:

table is for one flap angle ϕ

six two-dimensional coefficients in table

TYPE = CAMRAD:

wing can have trailing edge flap at any span station

six four-dimensional coefficients in table

3–6 Class = Airfoil, Type = Standard

TABLE CONTAINS THREE OR SIX TWO-DIMENSIONAL COEFFICIENTS

$c_\ell(\alpha, M)$, $c_d(\alpha, M)$, $c_m(\alpha, M)$;
$c_{\ell f}(\alpha, M)$, $c_{df}(\alpha, M)$, $c_{mf}(\alpha, M)$ with trailing edge flap

separate, rectangular (α, M) array for each:

α_i for $i = 1$ to N_α
M_j for $j = 1$ to N_M

dependent variable data consists of values $c_\ell(i, j)$, $c_d(i, j)$, $c_m(i, j)$; $c_{\ell f}(i, j)$, $c_{df}(i, j)$, $c_{mf}(i, j)$

coefficients can be in any order
angle of attack can be row or column
angle of attack α is cyclical
table for one flap angle ϕ

FILE FORMAT: FORMATTED, FREE FORM

figure 5 defines the file format

three or six coefficients, each function of A and B: set of C(A,B)

labels LABELA and LABELB identify independent variables A and B as
    angle of attack: label = ALPHA
    Mach number: label = MACH
either order

table is for airfoil with trailing edge flap if keyword “FLAP” is present

second reference value is the flap angle

labels LABELGk identify the dependent variables as
lift coefficient: label = CL
drag coefficient: label = CD
moment coefficient: label = CM
flap lift coefficient: label = CLF
flap drag coefficient: label = CDF
flap moment coefficient: label = CMF
any order

limitations:

parameter	maximum	minimum
TITLE	80 characters	
IDENT	20 characters	
LABELA, LABELB, LABELG	8 characters	
NAk, NBk	400 α, 100 M	2
number of reference values	2	

values of A and B must be unique and sequential

more than one line can be used for each table row; new table row must start on new line.

nothing is read after “END OF FILE”

general format

```
TITLE
IDENT
LABELA   LABELB   FLAP

NAk   NBk   LABELGk                              block repeated 3 or 6 times
           B(1,k)     ... ... B(NBk,k)     k = 1 to 3 or 6
A(1,k)     C(1,1,k)   ... ... C(1,NBk,k)
.          .                  .
.          .                  .
.          .                  .
A(NAk,k)   C(NAk,1,k) ... ... C(NAk,NBk,k)

REF1  REF2
END OF FILE
```

typical format

```
TITLE
IDENT
ALPHA   MACH
NAL   NML   CL
     MMM  MMM  ...  MMM
AAA  CCC  CCC  ...  CCC
AAA  CCC  CCC  ...  CCC
...  ...  ...  ...  ...
...  ...  ...  ...  ...
AAA  CCC  CCC  ...  CCC
AAA  CCC  CCC  ...  CCC
NAD   NMD   CD
     MMM  MMM  ...  MMM
AAA  CCC  CCC  ...  CCC
AAA  CCC  CCC  ...  CCC
...  ...  ...  ...  ...
...  ...  ...  ...  ...
AAA  CCC  CCC  ...  CCC
AAA  CCC  CCC  ...  CCC
NAM   NMM   CM
     MMM  MMM  ...  MMM
AAA  CCC  CCC  ...  CCC
AAA  CCC  CCC  ...  CCC
...  ...  ...  ...  ...
...  ...  ...  ...  ...
AAA  CCC  CCC  ...  CCC
AAA  CCC  CCC  ...  CCC
RETBL1
END OF FILE
```

Figure 3-5 Table File Format: Class = Airfoil, Type = Standard.

3–7 Class = Airfoil, Type = C81

TABLE CONTAINS THREE OR SIX TWO-DIMENSIONAL COEFFICIENTS

$c_\ell(\alpha, M)$, $c_d(\alpha, M)$, $c_m(\alpha, M)$;
$c_{\ell f}(\alpha, M)$, $c_{df}(\alpha, M)$, $c_{mf}(\alpha, M)$ with trailing edge flap

separate, rectangular (α, M) array for each:

$$\begin{array}{rl} \alpha_i & \text{for } i = 1 \text{ to } N_\alpha \\ M_j & \text{for } j = 1 \text{ to } N_M \end{array}$$

dependent variable data consists of values $c_\ell(i, j)$, $c_d(i, j)$, $c_m(i, j)$; $c_{\ell f}(i, j)$, $c_{df}(i, j)$, $c_{mf}(i, j)$

coefficients must be in fixed order
angle of attack must be row
angle of attack α is cyclical
table for one flap angle ϕ

FILE FORMAT: FORMATTED, FIXED FORM

figure 6 defines the file format

three or six coefficients, each a function of A and M: set of C(A,M)

independent variable A is angle of attack, and independent variable M is Mach number

table is for airfoil with trailing edge flap if NA and NM for flap are present in header line

second reference value is the flap angle dependent variables are lift, drag, and moment coefficients; and flap lift, drag, and moment coefficients (fixed order)

limitations:

parameter	maximum	minimum
HEADER	30 characters	
NML, NAL, NMLF, NALF	99	2
NMD, NAD, NMDF, NADF	99	2
NMM, NAM, NMMF, NAMF	99	2
number of reference values	2	

values of A and M must be unique and sequential
header is used as title

basic read format is 10 columns, each 7 characters wide; first column only used for A values

if NM greater than 9, more than one line used for each table row; new table row must start on new line

with trailing edge flap, header line includes NMLF, NALF, NMDF, NADF, NMMF, NAMF; and CLF, CDF, CMF arrays follow CM array

without trailing edge flap, there must be two blanks after NAM in the header line

nothing is read after reference values (or one blank line)

EXTENSIONS OF ORIGINAL C81 FILE FORMAT

reference values added to end of table

can include coefficients for a trailing edge flap

without trailing edge flap, there must be two blanks after NAM in the header line general format

```
                                                                  read format
HEADER   NML,NAL,NMD,NAD,NMM,NAM                                  A30,6I2,6I2
            M(1,k)      ... ... M(NML,k)                          7X,9F7.0
A(1,k)      CL(1,1,k)   ... ... CL(1,NML,k)                       10F7.0/(7X,9F7.0)
.           .                   .                                 .
.           .                   .                                 .
A(NAL,k)    CL(NAL,1,k) ... ... CL(NAL,NML,k)                     10F7.0/(7X,9F7.0)
            M(1,k)      ... ... M(NMD,k)                          7X,9F7.0
A(1,k)      CD(1,1,k)   ... ... CD(1,NMD,k)                       10F7.0/(7X,9F7.0)
.           .                   .                                 .
.           .                   .                                 .
A(NAD,k)    CD(NAD,1,k) ... ... CD(NAD,NMD,k)                     10F7.0/(7X,9F7.0)
            M(1,k)      ... ... M(NMM,k)                          7X,9F7.0
A(1,k)      CM(1,1,k)   ... ... CM(1,NMM,k)                       10F7.0/(7X,9F7.0)
.           .                   .                                 .
.           .                   .                                 .
A(NAM,k)    CM(NAM,1,k) ... ... CM(NAM,NMM,k)                     10F7.0/(7X,9F7.0)
REF1  REF2                                                        2F14.0
```

typical format

```
HEADERHEADERHEADERHEADERHEADER N N N N N N
        MMMMMM MMMMMM MMMMMM
AAAAAA  CLIFT  CLIFT  CLIFT
AAAAAA  CLIFT  CLIFT  CLIFT
AAAAAA  CLIFT  CLIFT  CLIFT
        MMMMMM MMMMMM MMMMMM
AAAAAA  CDRAG  CDRAG  CDRAG
AAAAAA  CDRAG  CDRAG  CDRAG
AAAAAA  CDRAG  CDRAG  CDRAG
        MMMMMM MMMMMM MMMMMM
AAAAAA   CMOM   CMOM   CMOM
AAAAAA   CMOM   CMOM   CMOM
AAAAAA   CMOM   CMOM   CMOM
RETBL1
```

Figure 3-6 Table File Format: Class = Airfoil, Type = C81.

USE OF TABLE DATA (STANDARD AND C81 TYPES)

LINEAR INTERPOLATION, WITH NO EXTRAPOLATION BEYOND RANGE OF TABLE

figure 7 illustrates use of table data

ANGLE OF ATTACK α

to avoid errors (since there is no extrapolation), range should always extend from -180 to 180 deg

data at large $|\alpha|$ likely from generic test (usually NACA 0012 data), not from test of this airfoil

careful joining two sets of data

MACH NUMBER M

data from $M_{\rm min}$ to $M_{\rm max}$

evaluation of coefficients (with no extrapolation) for $M < M_{\rm min}$ good, as long as $M_{\rm min}$ is below about 0.3

evaluation of coefficients (with no extrapolation) for $M > M_{\rm max}$ can produce significant errors

NO ERROR MESSAGE PRODUCED IF TABLE DATA ACCESSED OUTSIDE AVAILABLE RANGE

when analyze rotors operating at high advancing tip Mach number and/or high thrust, be sure within range of valid airfoil data

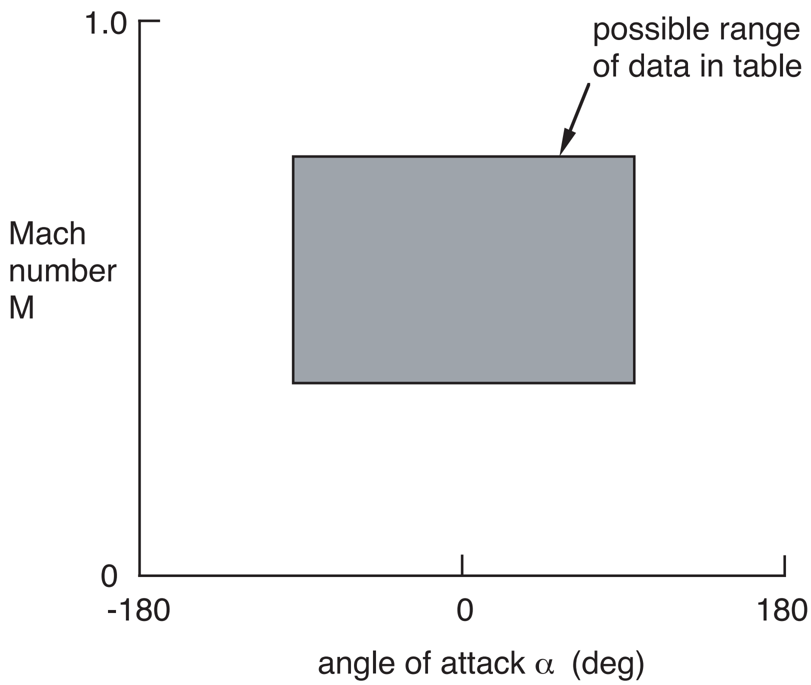


Figure 3-7 Use of airfoil table data (Standard or C81 type).

3–8 Class = Airfoil, Type = CAMRAD

TABLE CONTAINS THREE THREE-DIMENSIONAL COEFFICIENTS

$c_\ell(\alpha, M, r)$, $c_d(\alpha, M, r)$, $c_m(\alpha, M, r)$

last independent variable is wing span station r

if wing has trailing edge flap at any span station:
$c_{\ell f}(\alpha, M, \phi, r)$, $c_{df}(\alpha, M, \phi, r)$, $c_{mf}(\alpha, M, \phi, r)$ as well

common, rectangular (α, M) array:

α_i for $i = 1$ to N_α
M_j for $j = 1$ to N_M

for set of values ϕ_l ($l = 1$ to N_ϕ) and r_k ($k = 1$ to N_r)

special convention used for (α, M) array, designed for efficient interpolation

dependent variable data consists of values $c_\ell(i, j, l, k)$, $c_d(i, j, l, k)$, $c_m(i, j, l, k)$; $c_{\ell f}(i, j, l, k)$, $c_{df}(i, j, l, k)$, $c_{mf}(i, j, l, k)$

angle of attack α is cyclical

span station r can be searched or interpolated

for interpolation, r_k defines value for k-th airfoil

for search, r_k defines beginning of range of k-th airfoil

FILE FORMAT: UNFORMATTED

TABLE IS CONSTRUCTED BY PROGRAM INPUT FROM SET OF AIRFOIL TABLES

source table is STANDARD or C81 type

one source table for each span station and each flap angle

STANDARD and C81 tables are for single airfoil section and flap angle

CAMRAD table is for entire wing (one or more airfoil sections)

DATA IN SOURCE TABLE INTERPOLATED TO VALUES OF α and M REQUIRED FOR CAMRAD TABLE

resolution of (α, M) in CAMRAD table normally small enough that behavior of data in source table is accurately captured

USE OF TABLE DATA

LINEAR INTERPOLATION, WITH NO EXTRAPOLATION BEYOND RANGE OF TABLE

figure 8 illustrates use of table data

FOR EVALUATION OF CAMRAD TABLE FROM SOURCE TABLE

be sure range in CAMRAD table is larger than range in STANDARD or C81 table, so no information is lost

primary concern is maximum Mach number

FOR EVALUATION OF COEFFICIENTS FROM CAMRAD TABLE, IN AERODYNAMIC ANALYSIS

same considerations as for STANDARD and C81 tables

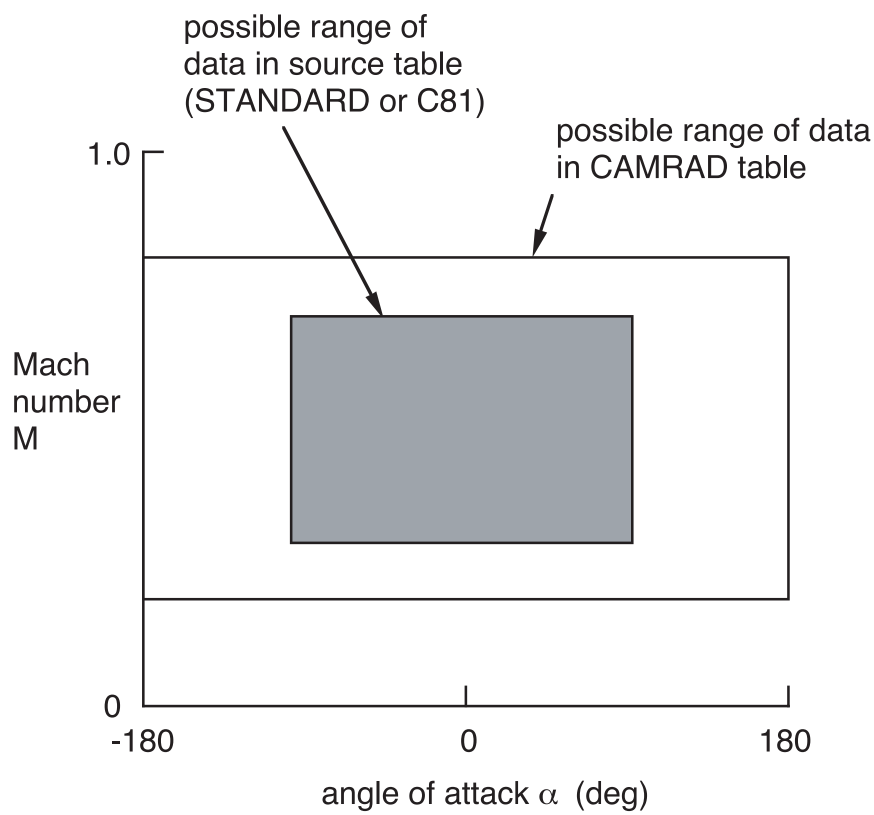


Figure 3-8 Use of airfoil table data (CAMRAD type).

SPECIAL CONVENTION USED FOR (α, M) ARRAY, DESIGNED FOR EFFICIENT INTERPOLATION

COEFFICIENT DATA DEFINED AT SET OF ANGLE OF ATTACK POINTS

THESE POINTS CONSIST OF SEVERAL GROUPS

WITH SAME ANGLE OF ATTACK INCREMENT WITHIN EACH GROUP

SET OF ANGLE OF ATTACK POINTS COMPLETELY SPECIFIED BY α VALUES AT BOUNDARIES BETWEEN GROUPS, AND INDICES OF THESE POINTS

figure 9 illustrates the convention

figure 10 shows typical boundaries (defaults)

there are N_a angle of attack boundaries ($N_a - 1$ groups)

index n counts actual angle of attack values, with $n = n_k$ at boundaries ($k = 1$ to N_a)

$n_1 = 1$ always, and n_{N_a} is total number of angle of attack values in table

corresponding angles of attack at the boundaries are α_k

no extrapolation, so angle of attack range should be -180 to 180 degrees

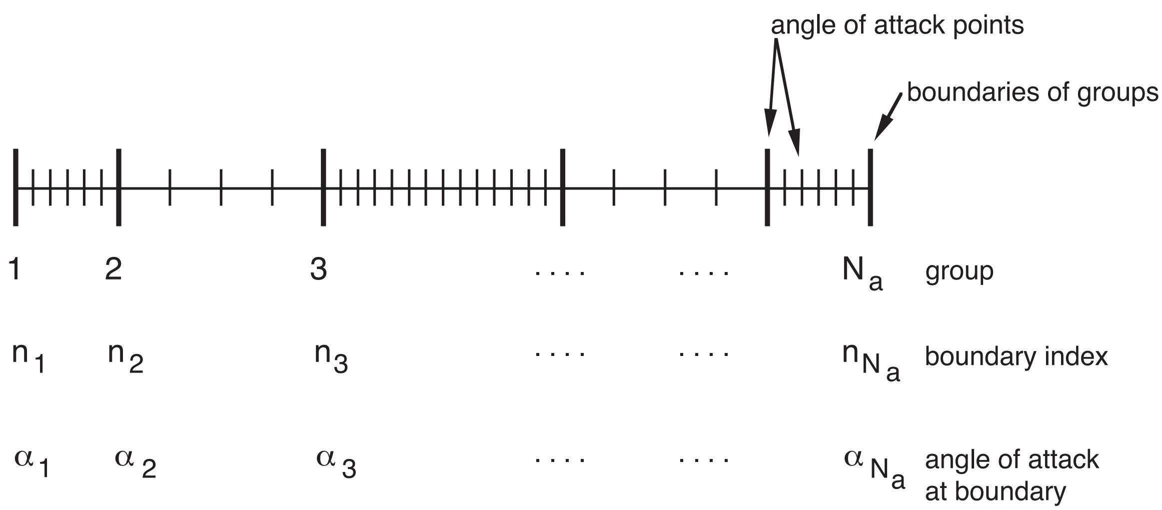


Figure 3-9 Convention for angle of attack points and boundaries.

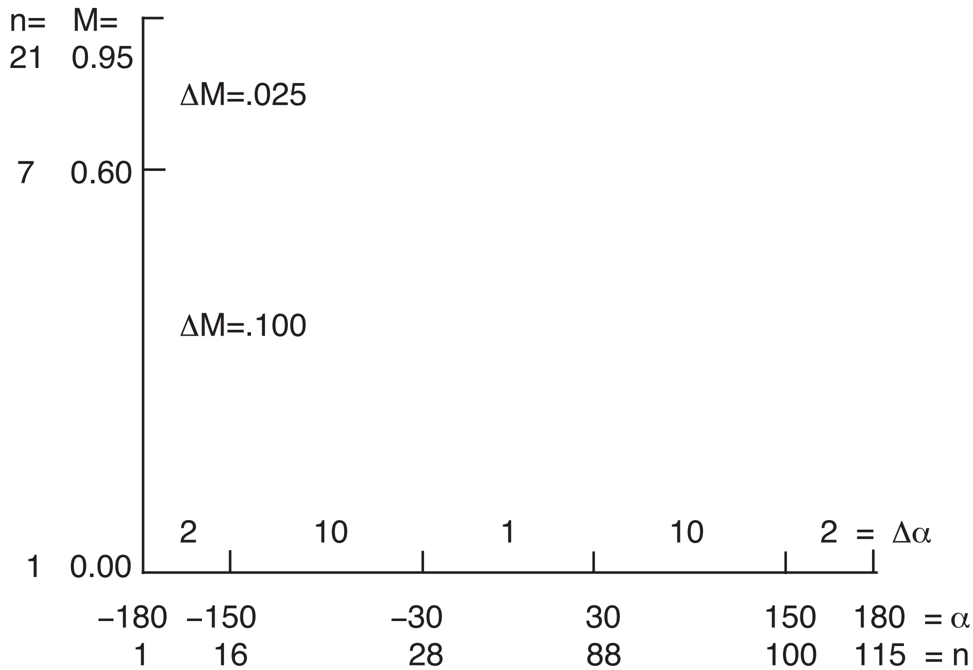


Figure 3-10 Typical angle of attack and Mach number boundaries.

PURPOSE OF THIS ORGANIZATION IS TO MINIMIZE THE SEARCH REQUIRED TO INTERPOLATE DATA

- ONLY NECESSARY TO SEARCH TABLE IN TERMS OF GROUPS
- INTERPOLATION WITHIN GROUP REQUIRES ONLY NUMERICAL OPERATIONS
  - reduction in search operations will be significant if large number of points can be divided into small number of groups
- ADDITIONALLY, SEARCH CAN BE STARTED WITHIN SPECIFIED GROUP, RATHER THAN AT BEGINNING OF RANGE

SIMILAR CONVENTION USED FOR VARIATION WITH MACH NUMBER

TABLE DATA DESCRIBED BY FOLLOWING QUANTITIES:

BOUNDARIES

angle of attack:	n_k and α_k, for $k = 1$ to N_a
Mach number:	n_k and M_k, for $k = 1$ to N_m
flap angle:	ϕ_l, for $l = 1$ to N_ϕ
span station:	r_k, for $k = 1$ to N_r

COEFFICIENT VALUES

$c(n_{N_a}, n_{N_m}, N_\phi, N_r)$ for lift, drag, and moment (total and flap)

SAME BOUNDARIES ARE USED FOR ALL COEFFICIENTS

DEFAULT BOUNDARIES

DEFAULT ANGLE OF ATTACK INCREMENTS:

1 deg from $\alpha = -30$ to 30,
2 deg from $\alpha = 150$ to -150,
10 deg otherwise

DEFAULT MACH NUMBER INCREMENTS:

$.100$ from $M = 0.$ to $.6$,
$.025$ from $M = .6$ to $.95$

IF WING HAS TRAILING EDGE FLAP

AT EACH SPAN STATION k, $N_{\phi k}$ SETS OF COEFFICIENTS, FOR FLAP ANGLES ϕ_l, $l = 1$ to $N_{\phi k}$

interpolate over α and M, then ϕ, then r

SPAN STATION r CAN BE SEARCHED OR INTERPOLATED

N_r SETS OF COEFFICIENTS, CORRESPONDING TO SPAN STATIONS r_k, $k = 1$ to N_r

for interpolation, r_k defines value for k-th airfoil

for search, r_k defines beginning of range of k-th airfoil

k-th airfoil used over range $r = r_k$ to $r = r_{k+1}$

in order to use spanwise interpolation on wing with trailing edge flaps, there should be tables at the span stations of the flap edges

WING GEOMETRY

CROSS-SECTION GEOMETRY DEFINED BY SPECIFIC AIRFOIL SECTIONS AT CERTAIN SPAN STATIONS

typically linear variation of geometry between these specified sections

figure 11 illustrates the geometry

CAMRAD TABLE FOR WING — CAN INTERPOLATE SPANWISE

TABLE SECTIONS CORRESPOND TO SPAN STATIONS WHERE WING GEOMETRY SPECIFIED

CAREFUL USING SPANWISE INTERPOLATION

with nonlinear aerodynamics, linear variation of geometry does not imply linear variation of loads

interpolation can produce problems with stall behavior, compressible drag rise, or zero lift angle

check behavior by plotting data from CAMRAD table at intermediate span stations

CAMRAD TABLE FOR WING — CAN SEARCH SPANWISE

SPAN STATIONS IN TABLE CORRESPOND TO BOUNDARIES BETWEEN SECTIONS

typically these boundaries correspond to edges of aerodynamic panels

IF TRANSITION REGION BETWEEN TWO SECTIONS IS LONG, MAY NEED TO INCLUDE SEPARATE TRANSITION SECTION

such transition sections are not usually tested, so the data for this section must be generated by the user (perhaps by linear interpolation)

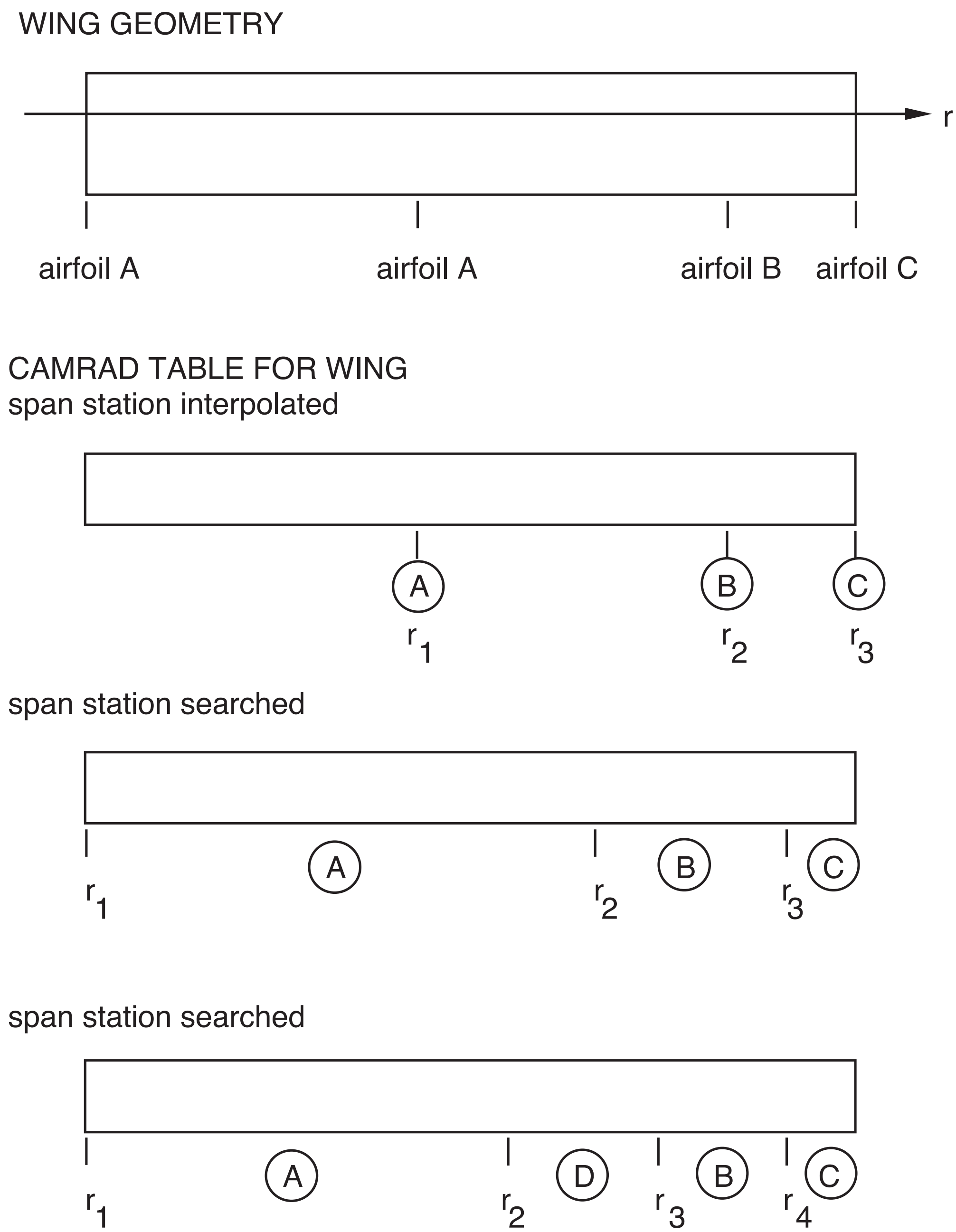


Figure 3-11 Spanwise specification of airfoil tables.

Chapter 4

THEORY

GENERAL REFERENCE FOR CHAPTER: CAMRAD II Documentation, Volume I, Theory

4–1 System Pieces

FOR CONFIGURATION GENERALITY, SPLIT SYSTEM INTO PIECES, WITH CONNECTIONS BETWEEN

- ENVIRONMENTAL PIECES
- PHYSICAL PIECES
- LOGICAL PIECES (SOLUTION PROCEDURE)

PIECES AVAILABLE TO CONSTRUCT SYSTEM:

environmental	physical	logical
case	component	loop
wind	frame	part
operating condition	interface	transform
period	output	modes
	input	response
		weights

ENVIRONMENTAL PIECES

provide standard description of system operation

PHYSICAL PIECES

correspond to physical description of system

produce system equations

LOGICAL PIECES

define procedure for solving system equations

transform and modes provide standard means to introduce various transformations of system variables and equations

response provides standard means to define characteristics of all system variables

weights provides standard perturbation and convergence weights for response

4–2 System Equations

PHYSICAL PIECES DEFINE EQUATIONS THAT DESCRIBE SYSTEM

figure 1 illustrates relations between physical pieces

ASSOCIATED WITH EACH PIECE ARE CERTAIN VARIABLES:

system piece	variables
component n	degree of freedom ξ_n
interface l	constraint variables f_l
input m	input u_m
output q	output y_q

ALL THESE VARIABLES ARE VECTORS; EACH VECTOR HAS ONE OR MORE ELEMENTS

4-2.1 Components

A COMPONENT CONTAINS THE PHYSICS OF THE MODEL, AND PERFORMS ALL CALCULATIONS FOR THE SYSTEM

COMPONENT n CAN HAVE DEGREES OF FREEDOM ξ_{ni}

dependence on degrees of freedom means that equations of motion exist, and so usually a differential equation formulation is possible

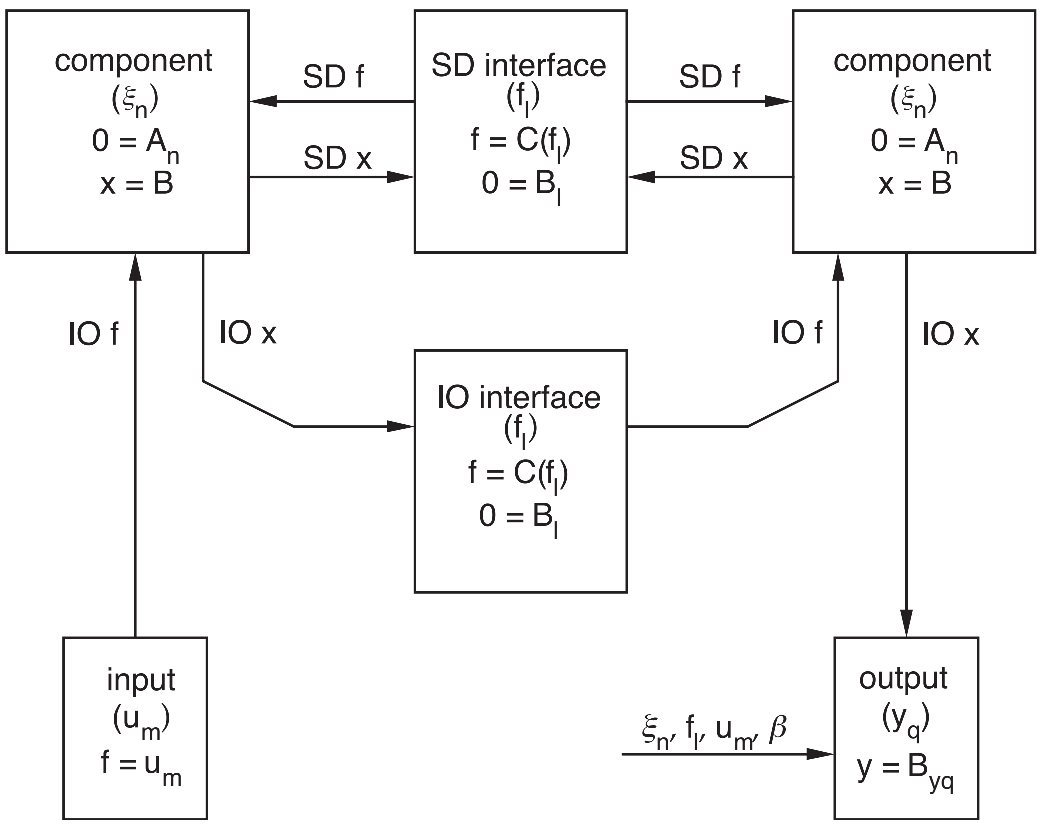


Figure 4-1 System equations.

COMPONENT CAN DEPEND ON FRAME MOTION β

FRAME MOTION CAN BE PRESCRIBED

OR FRAME MOTION CAN EQUAL RIGID BODY MOTION OF THIS OR OTHER COMPONENT (CALLED "FRAME DEGREES OF FREEDOM")

COMPONENT HAS INPUT f_{ni} AND OUTPUT x_{nj}

COMPONENT INPUT IS OF STRUCTURAL DYNAMIC KIND OR INPUT/OUTPUT KIND

can be connected to an interface or to a system input piece

COMPONENT OUTPUT IS OF STRUCTURAL DYNAMIC KIND OR INPUT/OUTPUT KIND

can be connected to an interface or to a system output piece

FOR STRUCTURAL DYNAMIC INTERFACE, INPUT AND OUTPUT OCCUR IN PAIRS

input is vector of force and moment at the connection

output is motion of axes at connection

HENCE COMPONENT PRODUCES MOTION AND OUTPUT EQUATIONS, DEPENDING ON DEGREES OF FREEDOM, FRAME MOTION, AND INPUT:

$$\begin{aligned} 0 &= A_{nj}(\xi_{ni}, \beta, f_{ni}) \\ x_{nj} &= B_{nj}(\xi_{ni}, \beta, f_{ni}) \end{aligned}$$

in general, nonlinear and time varying

if component does not have degrees of freedom, motion equations do not exist

4-2.2 Interfaces

AN INTERFACE CONNECTS TWO (OR MORE) COMPONENTS

INTERFACE HAS CONSTRAINT VARIABLES f_l

INTERFACE PRODUCES CONSTRAINT EQUATIONS
to be solved for constraint variables or degrees of freedom

AND CONSTRAINT FORCES
equations to evaluate component input f in terms of constraint variables

THESE EQUATIONS IN GENERAL DEPEND ON COMPONENT INPUT x AND CONSTRAINT VARIABLES:

$$\begin{aligned} 0 &= B_l(x, f_l) \\ f &= C_l(f_l) \end{aligned}$$

INTERFACE TYPE CAN BE STRUCTURAL DYNAMIC OR INPUT/OUTPUT

INPUT/OUTPUT INTERFACE

OUTPUT OF COMPONENT A SENT TO INPUT OF COMPONENT B
or to input of more than one component

CONSTRAINT VARIABLES f_l ARE COMPONENT OUTPUT

EQUATIONS:

$$\begin{aligned} 0 = B_l &= x_A - f_l \\ f_B = C_l &= f_l \end{aligned}$$

component output x_A need not be displacement, so can depend on derivatives of component A degrees of freedom

STRUCTURAL DYNAMIC INTERFACE

TWO COMPONENTS CONNECTED (COMPONENTS A AND B)

CONSTRAINT VARIABLES f_l ARE REACTION FORCES OF CONNECTION

component input is force and moment at connection (acting on the component)

component output is linear and angular motion of axes at connection

EQUATIONS:

$$\begin{aligned} 0 = B_l &= x_A - x_B \\ f_A = C_l &= +f_l \\ f_B = C_l &= -f_l \end{aligned}$$

EQUATIONS ARE ACTUALLY MORE COMPLICATED THAN THIS

constraint equation obtained from exact kinematics of interface

derivatives of B_l required

4-2.3 Output

AN OUTPUT PROVIDES AN EXTERNAL CONNECTION TO THE SYSTEM

OUTPUT HAS VARIABLES y_q

VARIABLES MAY BE CONNECTED TO OUTPUT x OF COMPONENT

this component output must be of input/output kind

OR OUTPUT VECTOR MAY BE IDENTIFIED AS SOME DEGREE OF FREEDOM, INTERFACE, INPUT, OR FRAME VECTOR (ξ_n, f_l, u_m, OR β)

EQUATIONS:

$$y_q = B_{yq} = \begin{cases} x \\ \xi_n \\ f_l \\ u_m \\ \beta \end{cases}$$

4-2.4 Input

AN INPUT PROVIDES AN EXTERNAL CONNECTION TO THE SYSTEM

INPUT HAS VARIABLES u_m

VARIABLES ARE CONNECTED TO INPUT OF ONE OR MORE COMPONENTS

EQUATIONS:

$$f = u_m$$

component input must be of input/output kind

4-2.5 System

COMPLETE SYSTEM COMBINES ALL SYSTEM PIECES

SYSTEM VARIABLES INCLUDE DEGREES OF FREEDOM ξ, CONSTRAINT VARIABLES f, INPUT u, AND OUTPUT y.

frame degrees of freedom are included in ξ

all system variables are partitioned into vectors, the vectors associated with individual system pieces

SYSTEM HAS MOTION EQUATIONS, CONSTRAINT EQUATIONS, AND OUTPUT EQUATIONS:

$$
\begin{aligned}
0 &= A(\xi, f, u) \\
0 &= B(\xi, f, u) \\
y &= B_y(\xi, f, u)
\end{aligned}
$$

THESE EQUATIONS ARE ASSOCIATED WITH DEGREES OF FREEDOM, CONSTRAINT VARIABLES, AND OUTPUT QUANTITIES RESPECTIVELY

usually for structural dynamic interfaces, derivatives of constraint equation are also required

IN GENERAL, EQUATIONS ARE TIME VARYING AND NONLINEAR

CAN BE DIFFERENTIAL EQUATIONS (IF SYSTEM HAS DEGREES OF FREEDOM)

OR IMPLICIT EQUATIONS (EVALUATING OUTPUT FROM INPUT)

4–3 Solution Procedure

4-3.1 Loops and Parts

TO MAINTAIN A FLEXIBLE ABILITY TO SOLVE AEROMECHANICAL PROBLEMS, A BUILDING-BLOCK APPROACH IS REQUIRED FOR LOGICAL PIECES OF SYSTEM, JUST AS FOR PHYSICAL PIECES

IN GENERAL IT WILL BE NECESSARY TO IMPLEMENT ITERATIVE SOLUTION METHODS, SINCE EQUATIONS ARE OFTEN LARGE AND NONLINEAR

- APPROACH USED IS TO DEFINE SOLUTION PROCEDURE IN TERMS OF LOOPS AND PARTS
  - figure 2 illustrates approach

A PART SOLVES SET OF EQUATIONS FOR RESPONSE

- PHYSICAL SYSTEM (COMPONENTS, INTERFACES, AND OUTPUT) IS DIVIDED INTO PARTS
  - so each part has subset of system motion, constraint, and output equations
- USING METHOD THAT DEPENDS ON PART TYPE, PART SOLVES EQUATIONS FOR CORRESPONDING DEGREE OF FREEDOM, CONSTRAINT, AND OUTPUT VARIABLES

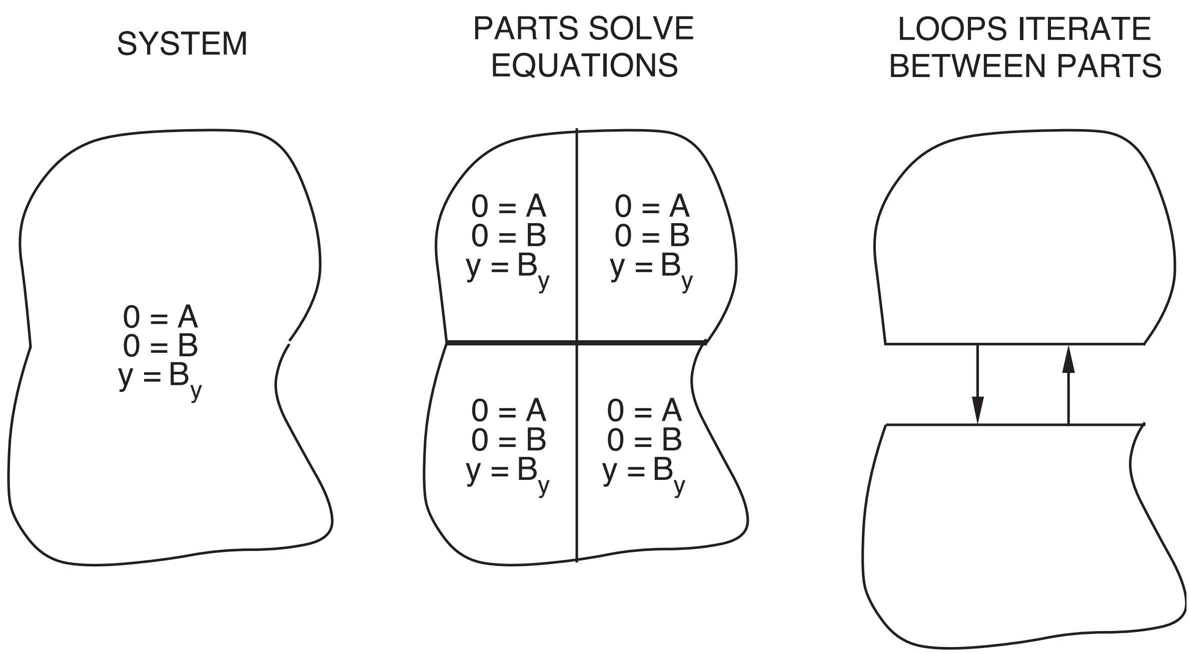


Figure 4-2 Solution procedure.

A LOOP ITERATES BETWEEN PART SOLUTIONS

USING METHOD THAT DEPENDS ON LOOP TYPE, THE LOOP ITERATES UNTIL CONVERGED SYSTEM SOLUTION IS OBTAINED

LOOPS CAN BE HIERARCHICAL, EXECUTING CHILD LOOPS AS WELL AS SPECIFIED PART SOLUTIONS

LOOPS ALSO CONTROL WRITE OF OUTPUT AND OTHER INFORMATION, BY EXECUTING SPECIFIED WRITE MODULES

FACTORS THAT CAN MAKE IT ADVANTAGEOUS OR ESSENTIAL TO SOLVE THE EQUATIONS BY PARTS (SUBSETS OF THE SYSTEM EQUATIONS)

a) iterative solution of smaller problems can be more efficient than direct solution of large problem

b) different kinds of equations (implicit, static, differential, integral) may require different solution procedures

c) different subsystems may require or allow different parameters in solution procedure

d) certain approximations can be implemented only with partitioned solution (such as using averaged equations in solution procedure; or handling system with more than one period)

e) subsystem motion or constraint equations can be simplified when the effects of variables in other parts are entirely contained in the right-hand-side force vectors

ANALYSIS CAN SOLVE ALL EQUATIONS TOGETHER (ONE PART, NO-SOLUTION TYPE LOOP), IF APPROPRIATE

4-3.2 Iterative Methods

AN ITERATIVE LOOP OR PART SOLUTION METHOD IS OFTEN REQUIRED, BECAUSE OF EITHER NONLINEARITY OR SIZE OF PROBLEM

ITERATIVE METHODS GENERALLY HAVE RELAXATION FACTOR (AND OTHER PARAMETERS) TO IMPROVE CONVERGENCE

AND TOLERANCE TO MEASURE CONVERGENCE

convergence is tested in terms of an error, which is typically some norm of difference between two iterations

BASIC APPROACH IS TO FIND LARGEST VALUE OF RELAXATION FACTOR FOR WHICH ITERATION WILL CONVERGE

HAVING ACHIEVED CONVERGENCE, LARGEST VALUE OF TOLERANCE THAT PRODUCES ACCURATE SOLUTION IS FOUND

using smaller relaxation factor or smaller tolerance can be too inefficient

ASSESSING TOLERANCE USUALLY DONE BY LOOKING AT OUTPUT QUANTITIES RELEVANT TO SPECIFIC PROBLEM

- tolerance is decreased until user decides that change in these output quantities is not important

WHEN CORRECT SOLUTION IS NOT KNOWN, CONVERGENCE MUST BE TESTED BY COMPARING THE VALUES OF TWO SUCCESSIVE ITERATIONS

- effect of relaxation factor is to reduce difference between iterations
  - SO REDUCTION OF RELAXATION FACTOR MUST OFTEN BE ACCOMPANIED BY REDUCTION IN TOLERANCE, TO MAINTAIN SAME ACCURACY

4-3.3 Order Reduction

EFFICIENCY CAN OFTEN BE IMPROVED BY MINIMIZING NUMBER OF VARIABLES INVOLVED IN SOLUTION

- SUPPRESSING EFFECT OF SELECTED VARIABLES CAN ALSO BE CONCEPTUALLY USEFUL

SO ORDER REDUCTION AVAILABLE:

- SET RESPONSE TO ZERO (PERHAPS OMIT VARIABLE AND CORRESPONDING EQUATION ENTIRELY)
- QUASISTATIC REDUCTION OF DEGREES OF FREEDOM

order reduction specified in response piece for each element of each vector (superseded for any mode or transform)

- specify zero, dynamic, or quasistatic
- for trim, transient, and flutter tasks

MODAL TRANSFORMATION OF STRUCTURAL DYNAMIC EQUATIONS

ALLOWS ORDER REDUCTION ON BASIS OF FREQUENCY CONTENT OF EXCITATION, COMPARED TO NATURAL FREQUENCIES OF MODES

equations of motion and output

$$m\ddot{\xi} + k\xi = \phi_e^T f$$
$$x = \phi_d \xi$$

partitioned into kept (ξ_k) and neglected (ξ_l)

$$m_k\ddot{\xi}_k + k_k\xi_k = \phi_{ek}^T f$$
$$m_l\ddot{\xi}_l + k_l\xi_l = \phi_{el}^T f$$
$$x = \phi_{dk}\xi_k + \phi_{dl}\xi_l$$

for modes with large natural frequency, can neglect inertia relative to spring — quasistatic reduction

$$(m_l s^2 + k_l)\xi_l = m_l(s^2 + \omega_l^2)\xi_l \cong k_l\xi_l$$

good approximation if excitation at high frequencies is small

perhaps can just solve dynamic equation for ξ_k, so

$$x \cong \phi_{dk}\xi_k$$

but spring of high frequency mode can be small, leading to significant static response

most accurate to retain complete static response, in terms of static residual from truncated modes (equivalent to a mode-acceleration method)

$$\xi_l = k_l^{-1}\phi_{el}^T f$$
$$x = \phi_{dk}\xi_k + (\phi_{dl}k_l^{-1}\phi_{el}^T)f = \phi_{dk}\xi_k + G_l f$$

when subsystem has rigid degrees of freedom, must also consider effect of inertial relief on residual

4–4 Physical System Pieces

CORRESPOND TO PHYSICAL DESCRIPTION OF SYSTEM

PIECES AVAILABLE TO CONSTRUCT SYSTEM (BY CLASS):

- COMPONENT
  - type = available components
- FRAME
- INTERFACE
  - type = structural dynamic
  - type = input/output
- OUTPUT
- INPUT

THESE PIECES PRODUCE THE SYSTEM EQUATIONS

4–5 System Pieces: Components

COMPONENTS PERFORM ALL COMPUTATIONS ASSOCIATED WITH THE PHYSICS OF THE MODEL OF A SYSTEM

SO COMPONENTS ARE FOCUS FOR MODELLING ISSUES

including empiricism and approximations needed for practical model of many physical systems

DEVELOPING AN IMPROVED MODEL REQUIRES DEVELOPING A NEW COMPONENT, WHICH WILL FIT INTO EXISTING ANALYSIS FRAMEWORK

A COMPONENT CAN BE CONSIDERED AN OPERATOR THAT EVALUATES A VECTOR OR MATRIX

figure 3 illustrates functionality

at time t component evaluates one of following vectors:

motion equation: A_{nj}

component output for input/output interface: $x_{nj} = B_{nj}$

component output for structural dynamic interface: axes motion x_{nj}

from degrees of freedom ξ_{ni} (including frame motion) and from component input f_{ni}

COMPONENT CAN ALSO PERTURB THIS VECTOR (ANALYTICALLY OR NUMERICALLY), TO CONSTRUCT MATRIX COLUMN

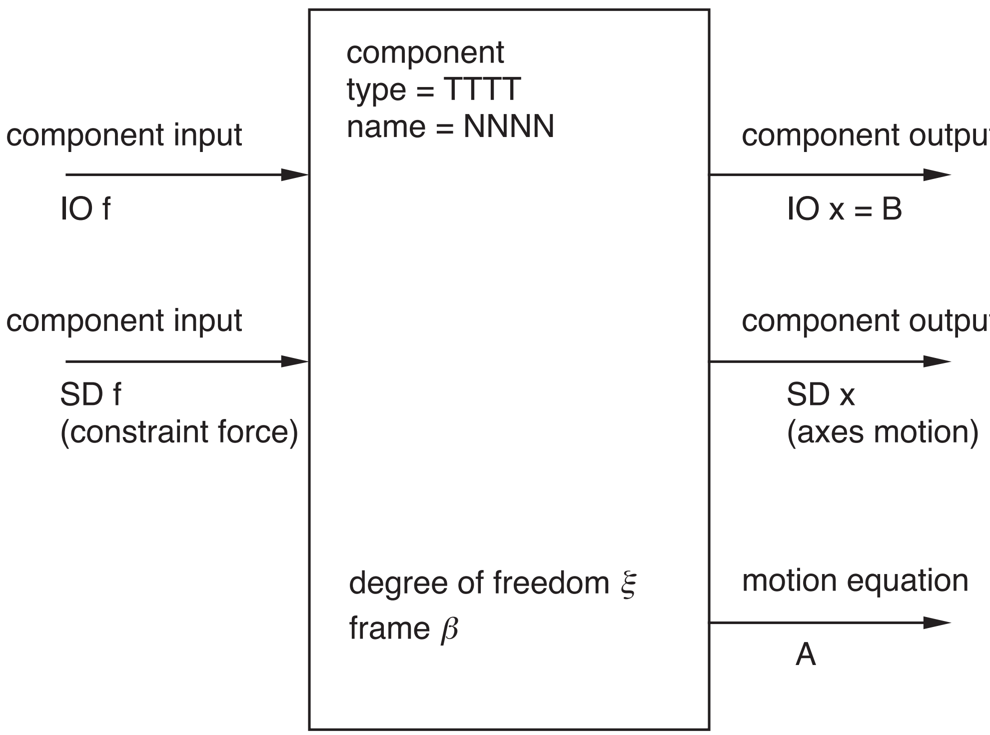


Figure 4-3 Component functionality.

IN GENERAL, EQUATIONS ARE TIME VARYING AND NONLINEAR

- CAN BE DIFFERENTIAL EQUATIONS (IF COMPONENT HAS DEGREES OF FREEDOM)
  - differential equation can be zero, first, or second order, as defined by dependence of A_{nj} and B_{nj} on derivatives of ξ
  - for structural dynamic no-residual interface, x_{nj} can not depend on the velocity or acceleration of ξ
  - may not be assumed to be symmetric
- OR IMPLICIT EQUATIONS (EVALUATING OUTPUT FROM INPUT)

COMPONENT EQUATIONS CAN ALSO DEPEND ON FRAME MOTION β

- frame motion is hierarchical, and each frame can be degrees of freedom (identified as rigid body motion of some component)
- so frame dependence translates into dependence on system degrees of freedom: component rigid degrees of freedom that correspond to frame and all its parent frames

INTERFACES BETWEEN COMPONENTS CAN BE STRUCTURAL DYNAMIC OR INPUT/OUTPUT KIND

COMPONENT DEFINITION INCLUDES COMPONENT INPUT AND COMPONENT OUTPUT

for a structural dynamic connection:
component input = force and moment at connection
component output = motion of axes at connection

component defines all input and output available

to be used, must be connected to another system piece (interface, input, or output)

STRUCTURAL DYNAMIC INPUT AND OUTPUT PAIR

can be connected to only one other component

but more than one connection can be defined at same point on component

COMPONENT OUTPUT OF INPUT/OUTPUT KIND

can be connected to one or more interfaces and one or more output pieces

COMPONENT INPUT OF INPUT/OUTPUT KIND

can be connected to only one interface or one input piece

but interface variables and input variables can be combined either inside component, or by using a differential equation component

figure 4 shows use of component input and component output by interface, input, and output system pieces

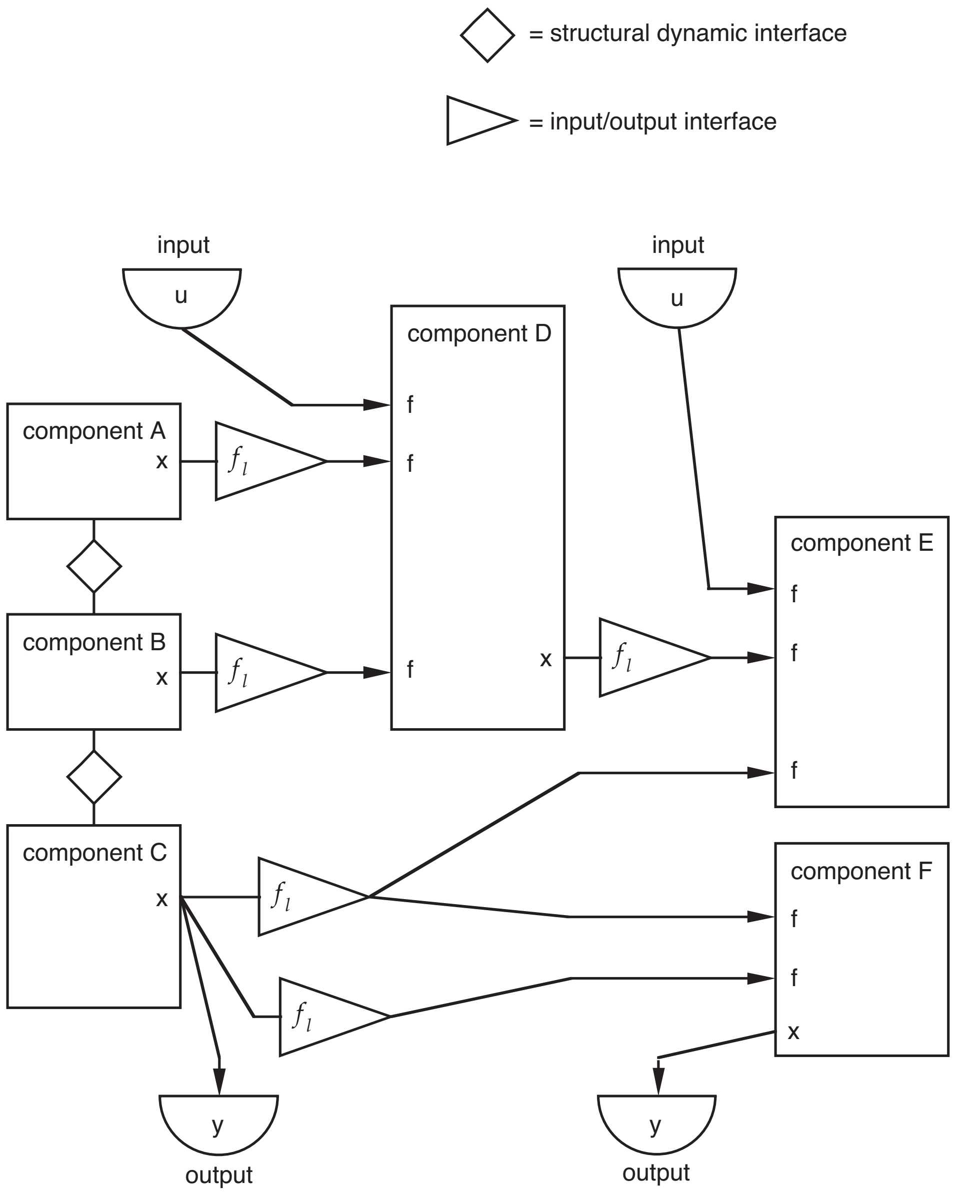


Figure 4-4 Use of component input and component output.

4-5.1 Structural Dynamic Components

STRUCTURAL DYNAMIC COMPONENTS CHARACTERIZED BY

rigid body motion and frames
mass, hence inertial and gravitational forces
structural dynamic interfaces

standard interface (component input and component output) defined for connections with aerodynamic components

STRUCTURAL DYNAMIC COMPONENTS DIFFER PRINCIPALLY IN MATTERS ASSOCIATED WITH ELASTIC MOTION

A FRAME (PERHAPS INERTIAL) MUST BE IDENTIFIED

rigid motion of component measured relative to that frame

STRUCTURAL DYNAMIC COMPONENT HAS RIGID BODY MOTION

“CONSTRAINED COMPONENT”: connected to frame, so no rigid body degrees of freedom

“FRAME COMPONENT”: rigid body degrees of freedom are the frame motion

OTHER CASES: rigid body degrees of freedom exist, and represent motion relative to the frame

figure 5 summarizes options for rigid body motion

ONLY A STRUCTURAL DYNAMIC COMPONENT CAN HAVE STRUCTURAL DYNAMIC INPUT AND OUTPUT

for connection to another component through a structural dynamic interface

structural dynamic connection made in common parent frame of two components

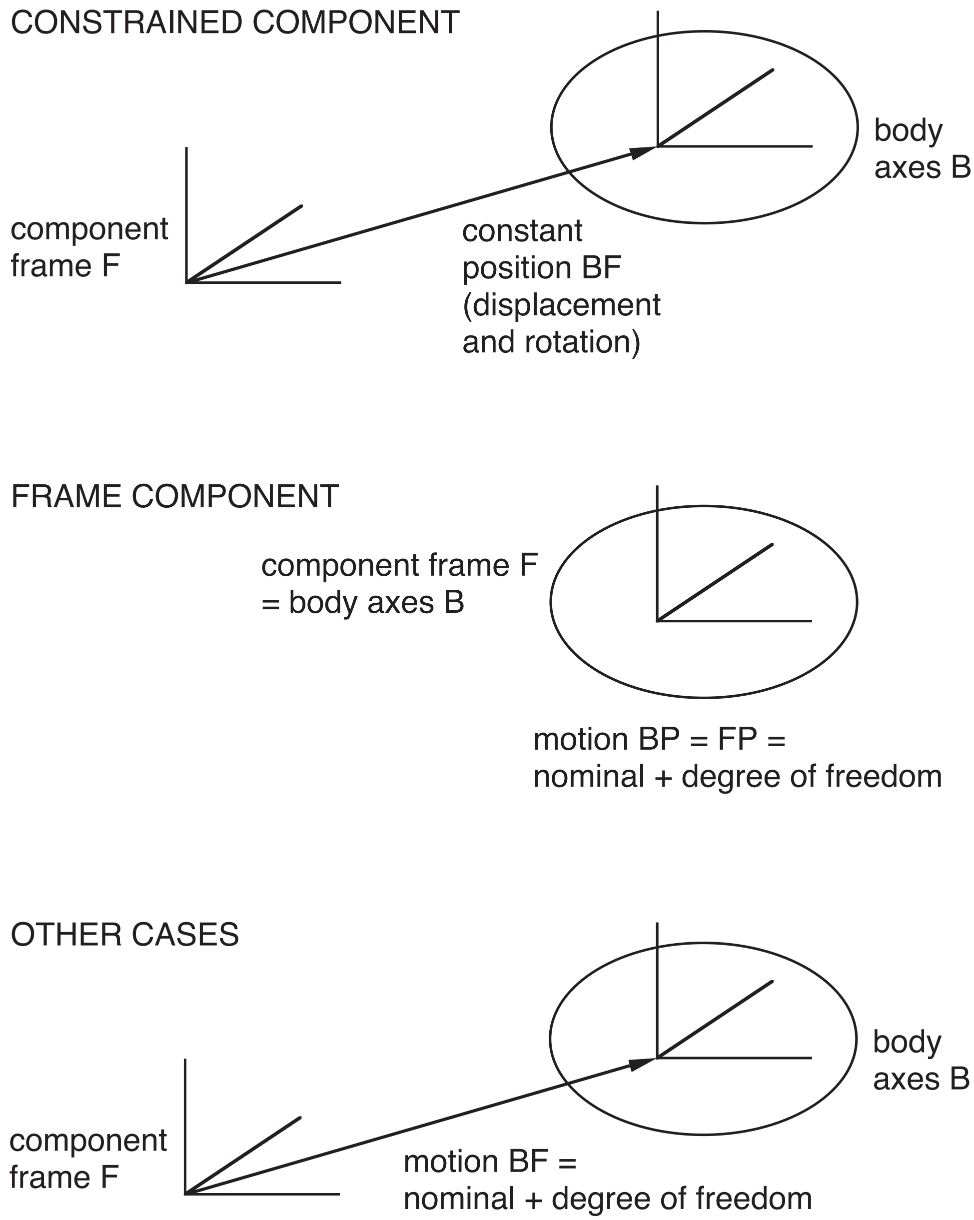


Figure 4-5 Rigid body motion of structural dynamic component.

4-5.2 Aerodynamic Components

AERODYNAMIC COMPONENTS INCLUDE WINGS AND WAKES

WING OR BODY IS SURFACE MOVING THROUGH AIR

- wing is thin surface, described by spanwise reference line (not necessarily straight), and two or more chordwise points at each span station

INTERFACES BETWEEN STRUCTURE AND AIR OCCUR AT SURFACE OF WING OR BODY

- interface is discretized (set of collocation points)
- these collocation points must be defined as connection points on structural dynamic components
  - typically interface involves velocity and force at connection points

WAKE COMPONENT SOLVES FOR MOTION OF AIR

- wing and wake components can be separate or combined, depending on model and solution procedure
- mutual influence between aerodynamic components typically accounted for by wakes

4–6 System Pieces: Frames

FRAMES HELP DERIVATION AND ANALYSIS OF PROBLEM

BY PROVIDING DESCRIPTION OF MOTION APPROPRIATE TO SPECIFIC CONFIGURATION

PARTICULARLY FOR STRUCTURAL DYNAMIC COMPONENTS

frame identified for a component,
all motion measured relative that frame

TO BE USEFUL, FRAME MOTION MUST BE PHYSICALLY RELEVANT

frame can be equated to rigid body motion of component (frame degrees of freedom)

component can be connected to frame (constrained component)

component motion relative frame can be small

components are coupled relative to their common frame

only motion relative that frame enters constraint equation;
so can use body-axis velocity for system rigid motion

FRAME MOTION CAN BE PRESCRIBED, OR DEGREES OF FREEDOM

FRAME DEGREES OF FREEDOM

IMPLEMENTED BY IDENTIFYING FRAME AS RIGID BODY MOTION OF PARTICULAR COMPONENT (“FRAME COMPONENT”)

so not necessary to introduce additional equations that tie frame to structure

FRAMES ARE HIERARCHICAL

frame motion measured relative parent frame (perhaps inertial)

MOTION (DISPLACEMENT AND ROTATION) OF PHYSICAL POINT (PROBABLY ON SOME COMPONENT):

$$\text{motion} = \text{frame} + \text{frame} + \ldots + \text{variable}$$

OPERATION "$+$" IS ADDITION OF RIGID MOTION

first frame ("base frame") consists of motion relative to inertial space

base frame can be associated with an operating condition

second frame consists of motion relative to first frame, and so on

final variable measured relative to its frame

final variable will probably be combination of several motions

figure 6 illustrates use of frames to describe motion

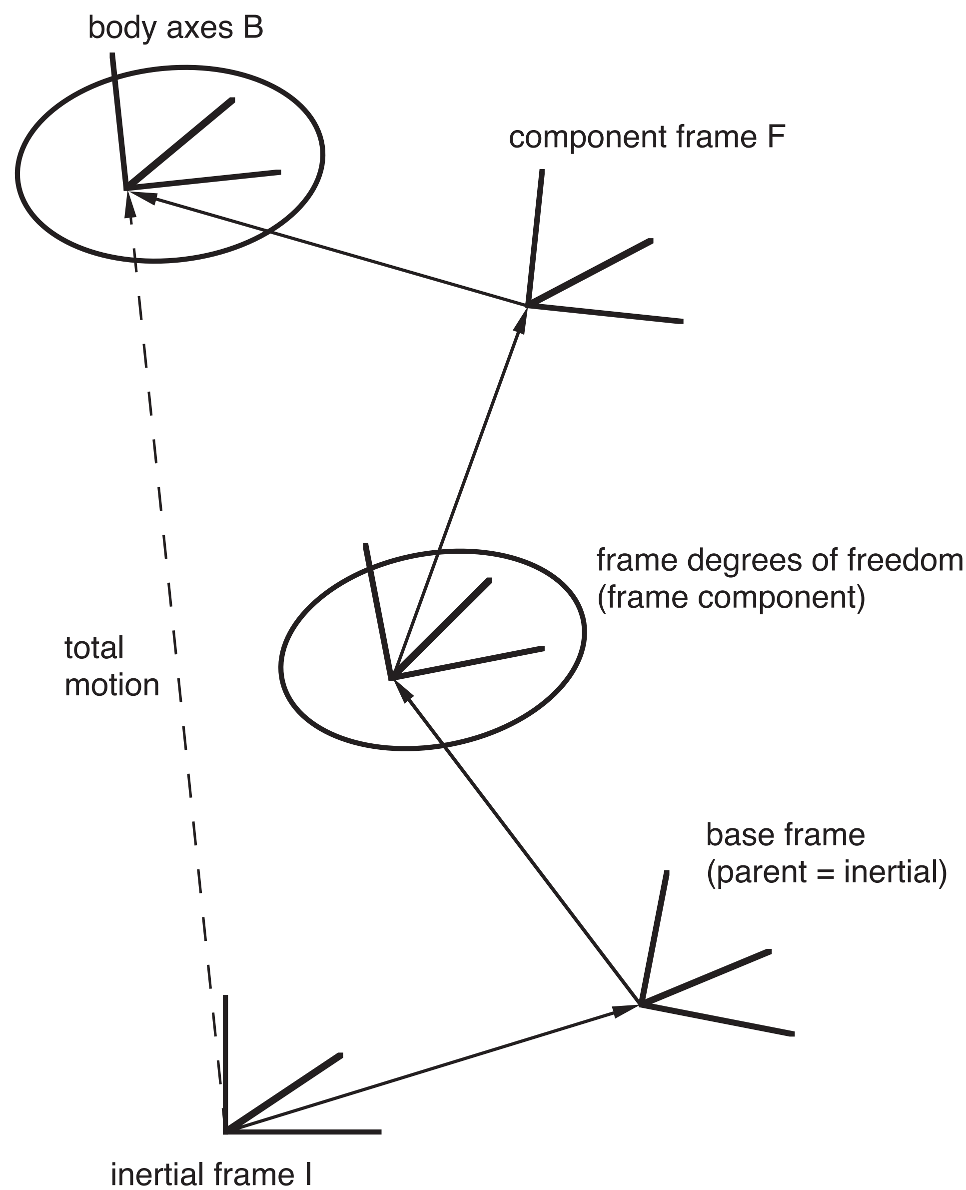


Figure 4-6 Use of frames to describe motion.

INERTIAL FRAME IS REFERENCE FOR ENTIRE SYSTEM

- EARTH AXES ARE FIXED RELATIVE TO INERTIAL AXES
    - orientation of inertial frame relative to earth defined by direction of gravity
    - ground plane defined by downward normal, and height origin of inertial frame above ground level

BASE FRAME (FREE FLIGHT)

- THE BASE FRAME IS FIRST FRAME OF SYSTEM: ITS PARENT IS THE INERTIAL FRAME
- "BASE FRAME" CAN BE ASSOCIATED WITH OPERATING CONDITION
    - linear motion must be represented by body-axes velocity or inertial axes displacement
    - angular motion must be represented by aircraft Euler angles
- BASE FRAME CAN BE USED AS RIGID MOTION OF SYSTEM (FRAME DEGREES OF FREEDOM)
    - with elastic motion of system measured relative rigid motion of base frame
    - so only base frame requires operating condition

MEAN AXES

FOR A FREE SYSTEM, OFTEN DESIRABLE TO DESCRIBE SYSTEM RIGID MOTION USING MOTION OF MEAN AXES

- mean axes are defined such that relative to them the entire system has zero linear and angular momentum
  - zero linear momentum: origin of mean axes at system center-of-mass
  - for linear system, degrees of freedom of mean axes are zero frequency (rigid body) modes of free vibration modes
- so mean axes and elastic modes are decoupled

TYPICAL AIRCRAFT CONVENTIONS

- inertial axes have z-axis down
- system rigid motion is frame degrees of freedom (center-of-mass, mean axes if possible)
  - body-axes velocity and aircraft Euler angle representation of motion
- rest position has x-axis in positive velocity direction (forward)
- yaw, pitch, and roll angles of aircraft Euler angle representation rotate the system from inertial axes to base frame

SYSTEM RIGID MOTION FOR AIRCRAFT DYNAMICS

figure 7 illustrates frames and axes of system rigid motion

CONVENTIONAL TO REPRESENT RIGID MOTION OF SYSTEM BY CENTER-OF-MASS, MEAN AXES

problem: not motion of physical point

solution: for linear normal modes component, rigid body motion is mean axes of component; usually can define system inertia so also mean axes of system

SYSTEM RIGID MOTION IN TRIM DEFINED BY OPERATING CONDITION

problem: complicated to use operating condition to describe trim motion of all components

solution: use operating condition to describe motion of base frame, and define all other motion relative to that frame

CONVENTIONAL TO REPRESENT SYSTEM RIGID MOTION BY BODY-AXIS VELOCITY COORDINATES

problem: complicated to obtain position relative inertial space when using body-axis velocity coordinates

solution: use body-axis velocity to describe motion of base frame, and define all other motion relative to that frame

(coupling components requires only their motion relative common frame, so position relative inertial space not needed)

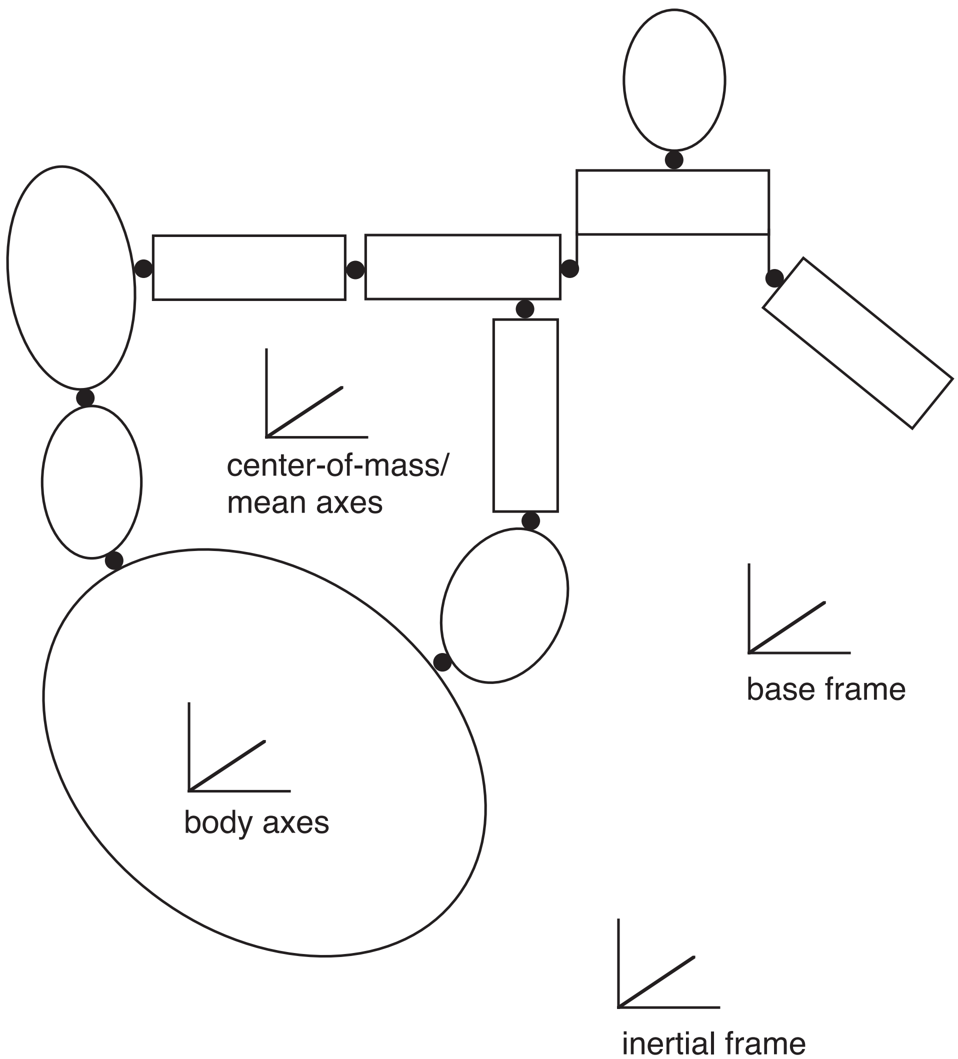


Figure 4-7 System rigid motion.

FRAMES FOR ROTORCRAFT SHELL

AIRFRAME FRAME (free flight)

aircraft convention for axes (z down, x forward)
base frame (parent = inertial frame)
frame degrees of freedom
center-of-mass, mean axes of component

conventions for definition of system inertia such that usually center-of-mass, mean axes of system

body-axes velocity and aircraft Euler angle representation of motion

or inertial axes displacement

parent frame of rest of system

so operating condition only required for frame reference motion

and structural dynamic interfaces (relative common frame) do not require position relative inertial space

and can use body-axes velocity representation

ROTOR FRAME (nonrotating hub, parent = airframe frame)

ROTATING FRAME (rotating hub, parent = rotor frame)

BLADE FRAME (rotating blade, parent = rotating frame)

frame degrees of freedom

4–7 System Pieces: Interfaces

AN INTERFACE IS A CONNECTION BETWEEN TWO OR MORE COMPONENTS

THERE ARE TWO TYPES OF INTERFACES: STRUCTURAL DYNAMIC AND INPUT/OUTPUT

EACH INTERFACE DEFINES CONSTRAINT EQUATIONS, WHICH ARE SOLVED TO ELIMINATE SOME SYSTEM VARIABLES

STRUCTURAL DYNAMIC INTERFACE: ELIMINATE SYSTEM DEGREES OF FREEDOM ξ_b

INPUT/OUTPUT INTERFACE: ELIMINATE INTERFACE VARIABLES f_l

IN GENERAL, EQUATIONS ARE TIME VARYING AND NONLINEAR

possibly not symmetric even for structural dynamic systems

input/output interface equation can depend on derivatives of system degrees of freedom

INTERFACE CONNECTS INPUT f AND OUTPUT x OF TWO OR MORE COMPONENTS

connection operation B_l produces constraint equation, from x and constraint variables f_l

connection operation C_l produces constraint force f, from constraint variables f_l

4-7.1 Input/Output Interface

INPUT/OUTPUT CONNECTION DEFINED BY IDENTIFYING OUTPUT x OF COMPONENT “A”, AND INPUT f OF ONE OR MORE COMPONENTS “B”

ALL COMPONENT INPUT AND OUTPUT INVOLVED MUST BE INPUT/OUTPUT KIND

“A” AND “B” CAN BE SAME COMPONENT

CONSTRAINT EQUATION IS SOLVED FOR CONSTRAINT VARIABLES f_l

$$0 = B_l = x_A - f_l$$

causes component to evaluate x_A

THEN COMPONENT INPUT f_B CAN BE EVALUATED FROM CONSTRAINT VARIABLES

$$f_B = C_l = f_l$$

figure 8 illustrates use of input/output interfaces

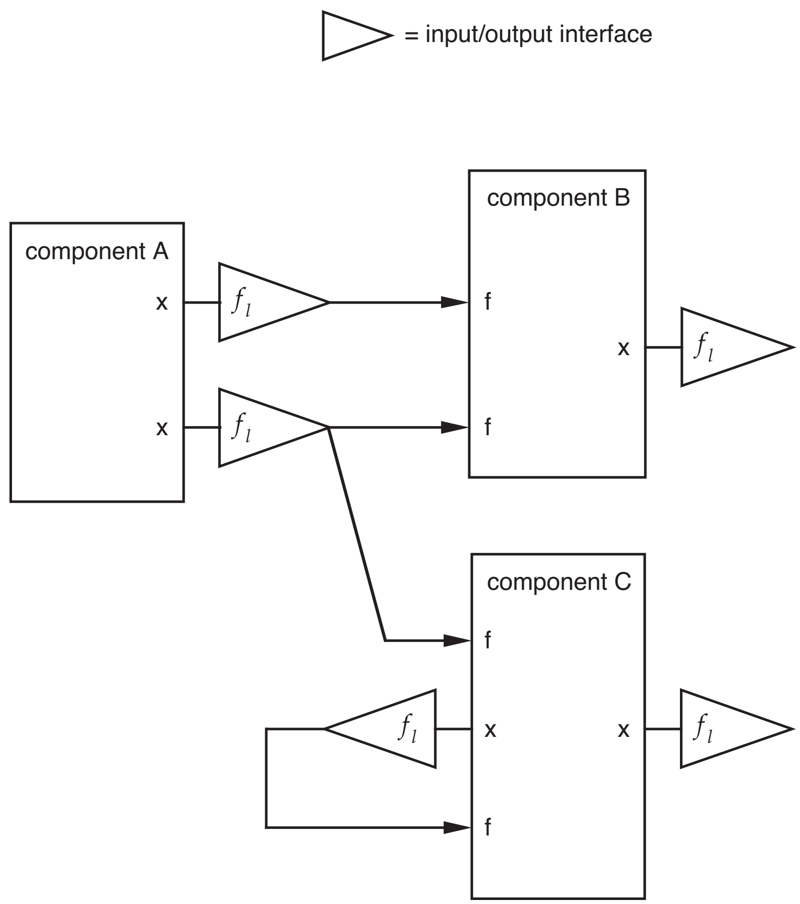


Figure 4-8 Use of input/output interfaces.

4-7.2 Structural Dynamic Interface

STRUCTURAL DYNAMIC INTERFACE ASSUMED HOLONOMIC AND INDEPENDENT

HOLONOMIC MEANS THAT THE CONSTRAINT EQUATION CAN BE DEFINED IN TERMS OF DISPLACEMENTS (RATHER THAN VELOCITIES OR VIRTUAL DISPLACEMENTS)

for nonholonomic interface, solution of constraint equation is path dependent (for example, rolling of a wheel on a surface)

INDEPENDENT MEANS THAT THE CONSTRAINT EQUATIONS ARE FULL RANK

dependent equations imply either redundant or inconsistent constraints

STRUCTURAL DYNAMIC CONNECTION DEFINED BY IDENTIFYING STRUCTURAL DYNAMIC INTERFACE (CONSISTING OF INPUT f AND OUTPUT x AT SAME POINT) OF TWO COMPONENTS, "A" AND "B"

component can not be connected to itself

INTERFACE VARIABLE f_l IS FORCE ACTING ON "A" SIDE OF INTERFACE

figure 9 illustrates use of structural dynamic interfaces

CONSTRAINT EQUATION SOLVED FOR SOME DEGREES OF FREEDOM ξ_b

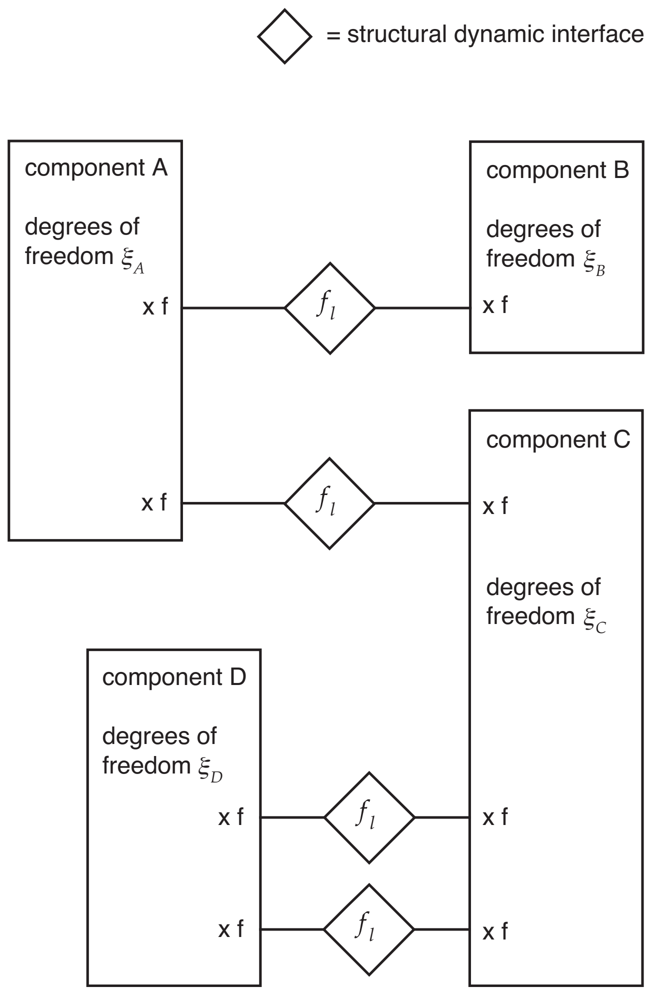


Figure 4-9 Use of structural dynamic interfaces.

CONSTRAINT EQUATIONS

$$0 = B_l = x_A - x_B$$

(equations for angular motion constraint are exact, so B_l is actually more complicated)

THEN COMPONENT INPUT f (FORCE AND MOMENT) CAN BE EVALUATED FROM CONSTRAINT FORCES f_l

$$f_A = C_l = +f_l$$
$$f_B = C_l = -f_l$$

(f_l has only force if angular motion not constrained)

CONNECTION KIND IS COMPLETE OR TORQUE

complete interface is true physical connection, torque interface is approximation

TORQUE INTERFACE

SPECIAL (APPROXIMATE) STRUCTURAL DYNAMIC CONNECTION

CONNECTS ONLY ROTATIONAL MOTION AND TORQUES

can act only on appropriate joints of component

produces no net moment on true structural dynamic component; only involved in joint equations

COMPLETE INTERFACE

INTERFACE CAN BE JOINT BETWEEN TWO COMPONENTS

general spring/damper/actuator joints implemented as part of all structural dynamic components

so only two joints implemented in interface: cantilever and pinned

CONSTRAINED COMPONENT (NOT SINGLE-POINT CONSTRAINT) USED TO ELIMINATE SYSTEM RIGID MOTION

so all structural dynamic interfaces are two-sided

POSITION AND ORIENTATION OF POINTS CONNECTED BY INTERFACE ARE DEFINED BY COMPONENT GEOMETRY

CANTILEVER

connection with no relative motion; equates linear and angular motion of points on two components

PINNED

ball joint connection; equates only linear motion of points on two components

figure 10 shows cantilever and pinned interfaces

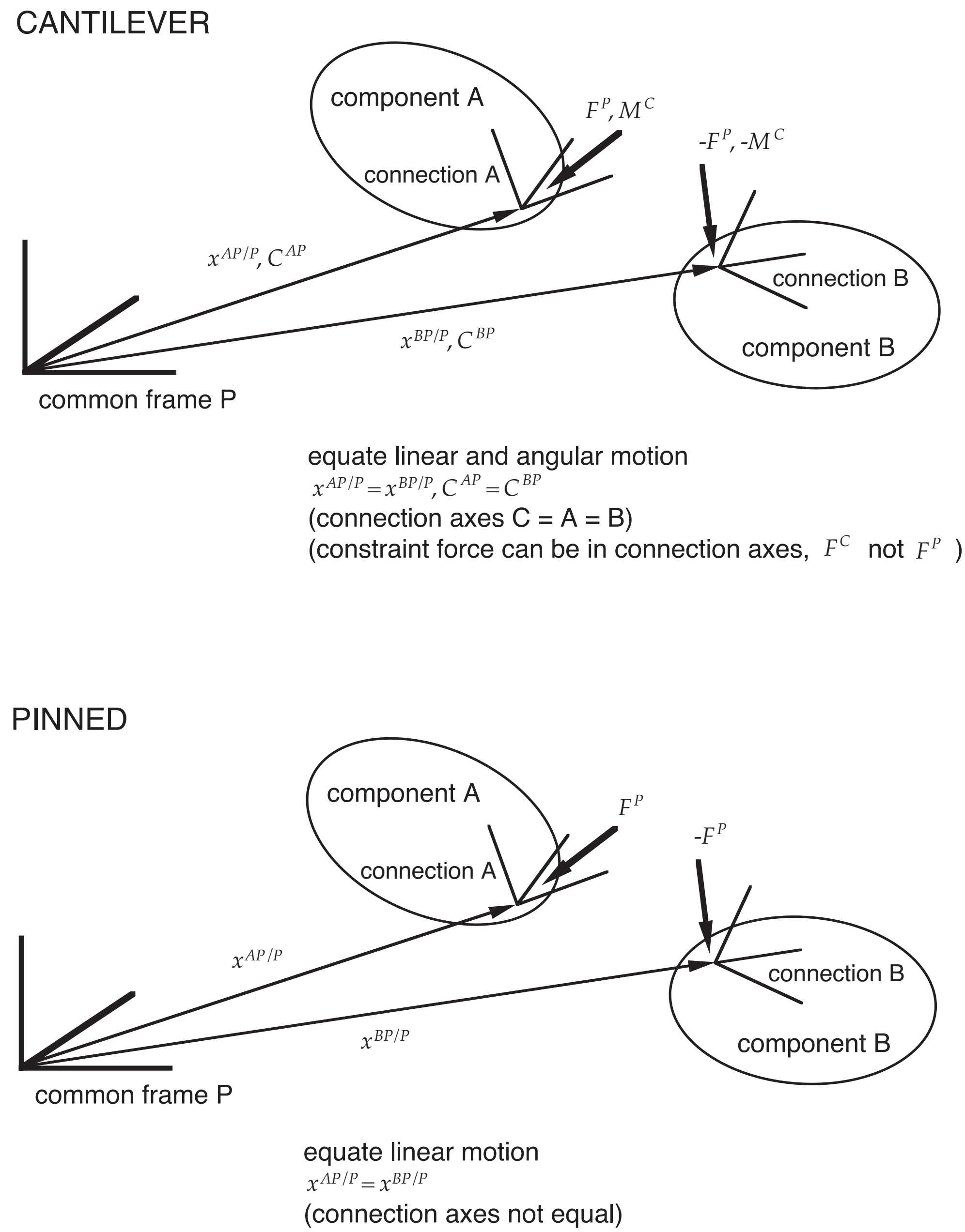


Figure 4-10 Cantilever and pinned structural dynamic interfaces.

USER MUST IDENTIFY SYSTEM DEGREES OF FREEDOM TO BE ELIMINATED BY CONSTRAINT EQUATIONS OF A STRUCTURAL DYNAMIC INTERFACE

PROGRAM CAN ELIMINATE RIGID DEGREES OF FREEDOM OF EITHER COMPONENT “A” OR COMPONENT “B”

so tree structure is simple: each interface should eliminate rigid motion of component on branch side of structure

with branch side identified as component “B”, interface forces are those acting on root side

USER NEED ONLY BE CONCERNED WITH IDENTIFYING THOSE DEGREES OF FREEDOM ASSOCIATED WITH MULTIPLE LOAD PATHS

simple if elastic joint put at connection that creates multiple load path: eliminate joint degrees of freedom

examples from rotorcraft shell:

pitch link — eliminate pitch bearing rotation, or some elastic torsion degree of freedom

snubber of bearingless hub — eliminate snubber joint degrees of freedom

figure 11 illustrates use of structural dynamic interfaces

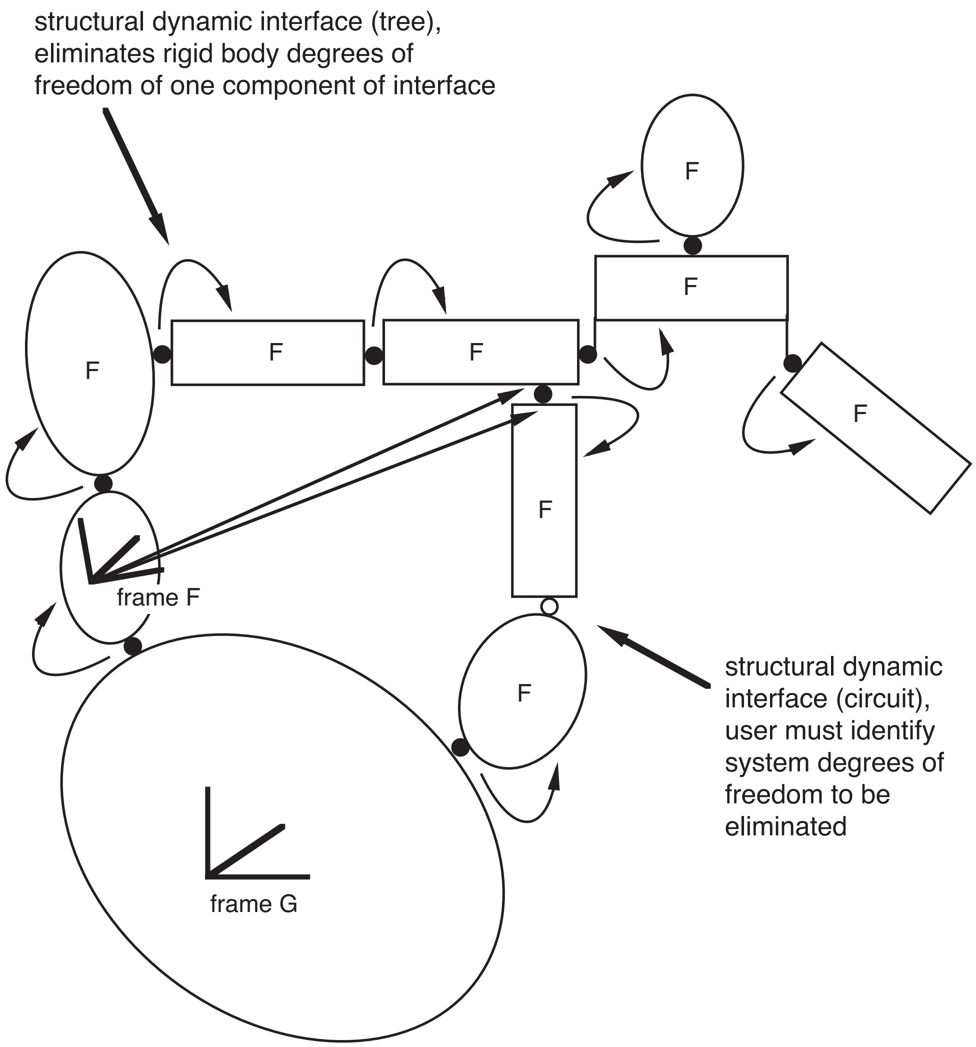


Figure 4-11 Use of structural dynamic interfaces.

4–8 System Pieces: Output

AN OUTPUT PROVIDES AN EXTERNAL CONNECTION TO THE SYSTEM

OUTPUT KIND:

- component degree of freedom, ξ_n
- interface variable, f_l
- input variable, u_m
- frame degree of freedom, β
- component output, x (input/output kind only)

FOR ALL QUANTITIES EXCEPT COMPONENT OUTPUT x:

ALREADY AVAILABLE IN SOLUTION, SO OUTPUT PIECE JUST CONTROLS WRITE PROCESS

separate solution for output is not necessary

IDENTIFYING y AS COMPONENT OUTPUT x PRODUCES AN OUTPUT EQUATION

this equation must be solved in some part

only this output kind is available as output variable in flutter equations

OUTPUT PIECE CONTROLS WRITE PROCESS

DEFINES QUANTITIES THAT ARE WRITTEN
such as time history or harmonics
AND THEIR DESTINATION
print, plot file, or printer-plot

A TIME HISTORY IS EVALUATED FROM RESPONSE SOLUTION

whether that solution is in time domain or frequency domain

AT NUMBER OF TIME STEPS SPECIFIED BY OUTPUT PIECE

FOR PERIODIC DATA, A COSINE/SINE FOURIER SERIES CAN BE EVALUATED

from preceding time history (even if solution is in frequency domain)

FOR NUMBER OF HARMONICS SPECIFIED BY OUTPUT PIECE

figure 12 illustrates the process

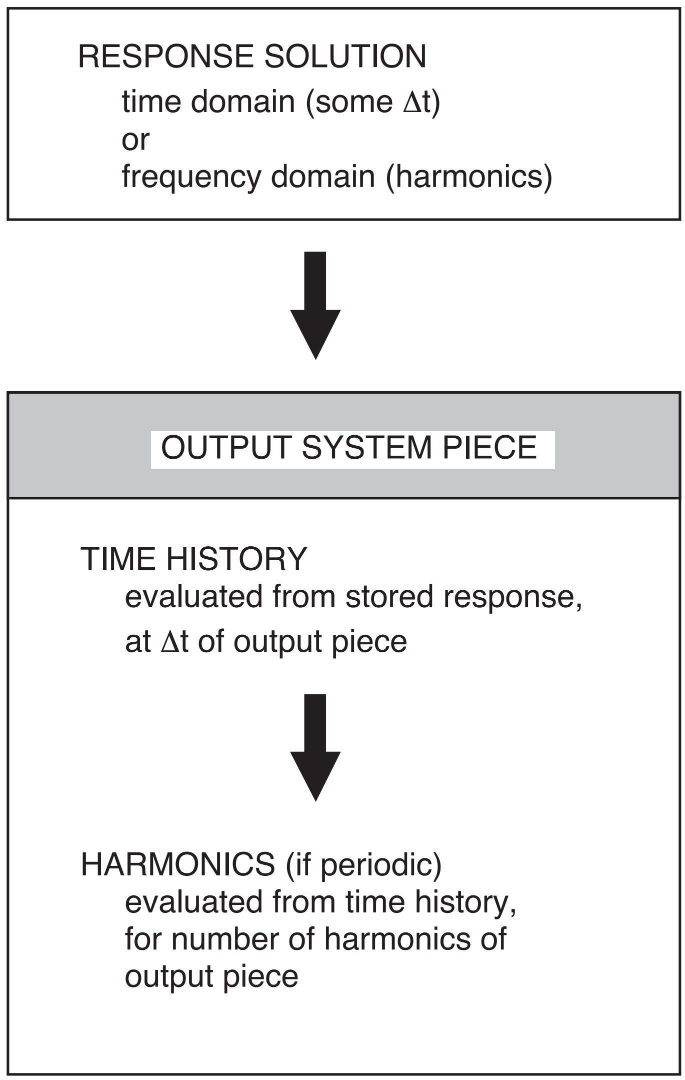


Figure 4-12 Calculation process for output piece.

HARMONIC ANALYSIS

PRODUCES COSINE/SINE FOURIER SERIES FROM DISCRETIZED DATA IN TIME DOMAIN

INTERPOLATION OPTIONS

Fourier interpolation

Fourier series representation (all harmonics) gives discrete data points exactly, but behavior uncontrolled in between

linear interpolation: harmonics of a linear interpolation between data at discrete times

Fourier series representation (infinite number of harmonics) gives linear interpolation exactly

finite number of harmonics gives smoothed curve, close to discrete data points

figure 13 illustrates interpolation options

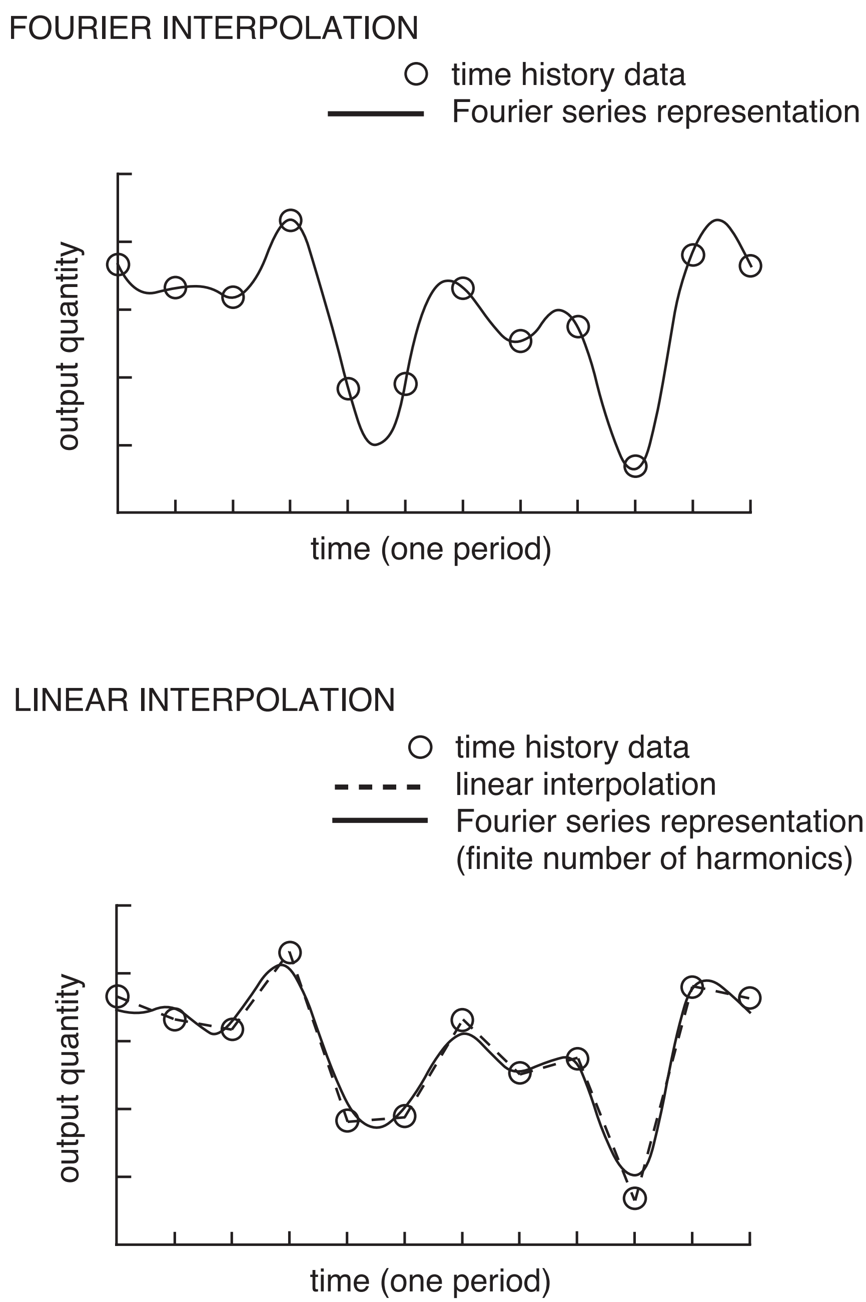


Figure 4-13 Interpolation options of harmonic analysis.

4–9 System Pieces: Input

AN INPUT PROVIDES AN EXTERNAL CONNECTION TO THE SYSTEM

required for trim and flutter tasks

a trim loop can change value of input (constant)

flutter can perturb input for linear equations

transient task needs components to generate prescribed control (input has constant value from trim solution)

DEFINED BY LENGTH OF INPUT VECTOR AND NAMES OF ELEMENTS IN INPUT VECTOR

CONNECTIONS OF INPUT TO COMPONENT INPUT f ARE IDENTIFIED

component input must be input/output kind

each input of component can be connected to only one source (interface variable f_l or input vector u_m)

TO OBTAIN COMPONENT INPUT FROM COMBINATION OF SEVERAL u_m AND/OR f_l:

combine within component

or use differential equation component (in static mode) to combine; result evaluated as input/output interface and sent to component that requires it

4–10 Environmental System Pieces

PROVIDE STANDARD DESCRIPTION OF SYSTEM ENVIRONMENT

PIECES AVAILABLE TO CONSTRUCT SYSTEM (BY CLASS):

- CASE
- WIND
- OPERATING CONDITION
- PERIOD

WIND, OPERATING CONDITION, AND PERIOD SYSTEM PIECES

- parameters available to all components
- trim loop solution methods can change these parameters

4–11 Environment

EARTH

ASSUME FLAT, NONROTATING EARTH

earth axes fixed relative to inertial axes
center-of-mass and center-of-gravity coincident
no gravity-gradient forces

ACCELERATION PRODUCED BY GRAVITY

vector g, constant in inertial axes

direction of g defines orientation of inertial frame on earth

GROUND PLANE

defined by downward normal, and height origin of inertial frame above ground level

needed for rotor inflow and wake models when in-ground-effect

AERODYNAMIC

SPEED OF SOUND c_s, DENSITY ρ, KINEMATIC VISCOSITY $\nu = \mu/\rho$ OF AIR (OR OTHER FLUID)

calculated from altitude for standard day

or from pressure altitude and temperature

or from density and temperature

or from density, sound speed, and viscosity

4–12 System Pieces: Case

SYSTEM MUST HAVE ONE (AND ONLY ONE) CASE PIECE DEFINED

INFORMATION

- case description
  - selection of analysis tasks
  - control print of input data
- timer and debug parameters
- earth and aerodynamic environment parameters

4–13 System Pieces: Wind

SYSTEM MUST HAVE ONE (AND ONLY ONE) WIND PIECE DEFINED

PARAMETERS

wind speed

orientation of the wind/gust axes (G) relative inertial axes, defined by yaw and pitch angles

these parameters are used to calculate velocity of air relative inertial frame

trim loop solution methods can change these parameters

WIND SPEED

wind speed positive for air from $+x$ axis direction (x is forward)

WIND/GUST AXES

gust axes and wind axes are same

wind/gust axes can be

velocity axes (from a specified free-flight operating condition)

inertial axes

or input orientation

gust components use wind/gust axes to define change of air velocity: u_G, v_G, and w_G components in G axes

OPTIONAL GROUND BOUNDARY LAYER

air velocity depends on height above ground

calculated using log law, modified log law, or power law

wind speed is air velocity at reference height

effect of ground boundary layer included in:

velocity relative air (aerodynamic interface) calculated by structural dynamic components

wake geometry distortion calculated by wing wake geometry component

4–14 System Pieces: Operating Condition

SYSTEM MUST HAVE AT LEAST ONE OPERATING CONDITION PIECE DEFINED

PARAMETERS

operating condition can be constrained or free

flight speed

orientation of the body frame (F) relative inertial axes, defined by yaw, pitch and roll angles

turn rate

orientation of the velocity frame (V) relative inertial axes, defined by yaw and pitch angles

these parameters are used to define motion of “base frame”

trim loop solution methods can change these parameters

FREE FLIGHT OPERATING CONDITION

AIRCRAFT CONVENTIONS:

flight speed positive for body moving in $+x$ axis direction (x is forward)

body axes or stability axes can be used to describe the motion

orientation defined by Euler angles, using yaw-pitch-roll sequence

TYPICALLY FREE SYSTEM WILL NOT USE WIND SPEED, AND GUST AXES ARE VELOCITY AXES

CONSTRAINED OPERATING CONDITION

ONLY ORIENTATION OF BODY FRAME RELATIVE INERTIAL AXES USED

yaw, pitch, and roll angles

no flight speed or turn rate

A VELOCITY OF AIR RELATIVE TO BODY OBTAINED USING THE WIND

4–15 System Pieces: Period

PARAMETER

rotational speed (constant)

PERIOD KIND CAN BE BASE OR CHILD

BASE PERIOD

rotational speed defined

trim loop solution methods can change this parameter

CHILD PERIOD

parent period and gear ratio r defined

(child rotational speed) = $r \times$ (parent rotational speed)

not hierarchical (parent must be base period)

ANY SYSTEM PIECE CAN IDENTIFY A PERIOD

SO CAN GET CORRESPONDING ROTATIONAL SPEED Ω DURING THE SOLUTION PROCESS

STANDARD RELATION BETWEEN AZIMUTH AND TIME IS USED:

$$\psi = \Omega t + \psi_0$$

REFERENCE AZIMUTH ψ_0 MUST BE DEFINED WHENEVER A PERIOD IS IDENTIFIED

4–16 Logical System Pieces

DEFINE PROCEDURE FOR SOLVING SYSTEM EQUATIONS

PART SOLVES SET OF EQUATIONS FOR RESPONSE,
LOOP ITERATES BETWEEN PART SOLUTIONS

type selects solution method

PIECES AVAILABLE TO CONSTRUCT TRIM SOLUTION (BY CLASS):

TRIM LOOP
- type = no solution
- type = successive substitution
- type = Newton Raphson
- type = regulator

TRIM PART
- type = no solution
- type = implicit
- type = static
- type = harmonic
- type = time finite element

TRANSFORM

MODES

RESPONSE
- type = rigid
- type = variable

WEIGHTS

PIECES AVAILABLE TO CONSTRUCT TRANSIENT SOLUTION (BY CLASS):

TRANSIENT

TRANSIENT LOOP
- type = no solution
- type = successive substitution

TRANSIENT PART
- type = no solution
- type = trim solution
- type = implicit
- type = integration

PIECES AVAILABLE TO CONSTRUCT FLUTTER SOLUTION (BY CLASS):

FLUTTER

FLUTTER LOOP

FLUTTER PART
- type = no solution
- type = interface
- type = differential equations

4–17 Trim Task

EQUILIBRIUM SOLUTION FOR STEADY STATE OPERATING CONDITION

ASSUME:

SYSTEM ENVIRONMENT AND INPUT ARE CONSTANT OR PERIODIC

EQUILIBRIUM SOLUTION EXISTS

SOLUTION IS CONSTANT OR PERIODIC

possible to define systems for which these assumptions are not true

USUALLY IDENTIFY PARAMETER VALUES REQUIRED TO ACHIEVE A SPECIFIED OPERATING CONDITION (AN INVERSE PROBLEM)

PARAMETERS THAT CAN BE IDENTIFIED:
inputs, and operating condition variables (system mean rigid motion)

IDENTIFICATION REQUIRES:
output quantities achieve a target value,
or aircraft force and moment be zero (mean equations of system rigid motion)

INVERSE SOLUTION IS IMPLEMENTED BY A TRIM LOOP

OR EQUATIONS CAN BE SOLVED FOR FIXED VALUES OF INPUTS AND OPERATING CONDITION

then free body will not be in equilibrium

EXAMPLES FOR ROTORCRAFT

FREE FLIGHT

find inputs (pilot collective, cyclic, pedal controls) and system mean rigid motion (pitch and roll angles)

so mean equations of system rigid motion are zero (net mean force and moment on aircraft equal zero)

WIND TUNNEL

find inputs (collective and cyclic controls)

so output quantities achieve target values (thrust and flapping)

FIXED CONTROLS

fixed input (collective and cyclic controls), in wind tunnel

4-17.1 Periodic Solution

WHEN EQUILIBRIUM SOLUTION IS PERIODIC, ASSUMED THAT PERIOD IS KNOWN

- TYPICALLY OBTAINED FROM ROTATIONAL SPEED OF SOME STRUCTURAL DYNAMIC SUBSYSTEM, $T_{\mathrm{ref}} = 2\pi/\Omega$

THERE CAN BE SEVERAL PERIODS T_i SIMULTANEOUSLY INVOLVED IN THE SYSTEM BEHAVIOR

- EXAMPLE: SINGLE MAIN-ROTOR AND TAIL-ROTOR HELICOPTER
- SYSTEM PERIOD T, EQUAL TO COMMON INTEGER MULTIPLE OF ALL T_i, MIGHT STILL EXIST
  - with several rotating subsystems connected through gear train, the ratio of periods will be a rational number, and there will be a system period
- BUT SUCH A PERIOD MAY BE TOO LARGE FOR A PRACTICAL TRIM ANALYSIS
  - TRANSIENT TASK CAN ANALYZE SUCH CASES

APPROXIMATE SOLUTION WITH SEVERAL PERIODS PRESENT:

SOLVE FOR RESPONSE AT EACH PERIOD SEPARATELY, IGNORING NON-HARMONIC VIBRATORY COUPLING BETWEEN PARTS

CONSIDER PERIODIC AND TIME-INVARIANT PARTS

PERIODIC PART

example: rotor

INHERENTLY INVOLVES PARTICULAR PERIOD T_I

so can be solved only for that period

assumption of periodicity requires that any input from other parts be at period T_I, or else only the mean value can be used

TIME-INVARIANT PART

example: airframe

HAS NO INHERENT PERIOD

so can respond to input at all periods T_i

solving at each period separately is approximation if part equations are nonlinear

4-17.2 Rotorcraft Trim Task

PARTITIONED, ITERATIVE SOLUTION PROCEDURE
CONSTRUCTED BY ROTORCRAFT SHELL

figure 14a shows the loops and principal interface variables

WAKE LOOP

computationally intensive task of obtaining wake influence coefficients (which depend on wake geometry) moved outside all other loops

TRIM LOOP

inverse problem requires solution of algebraic equations for controls and other trim variables

CIRCULATION LOOP

induced velocity calculated from integral equations, while rotor and airframe motion calculated from differential equations; large number of circulation variables

MOTION LOOP

separate solution for rotor and airframe motion required to avoid interharmonic coupling (from rotating-to-nonrotating interface), and to handle case of two periods for main-rotor and tail-rotor configuration

WAKE LOOP HAS UP TO THREE LEVELS

uniform inflow,
nonuniform inflow with prescribed wake geometry,
nonuniform inflow with free wake geometry

no test for convergence

figure 14b shows levels of wake loop

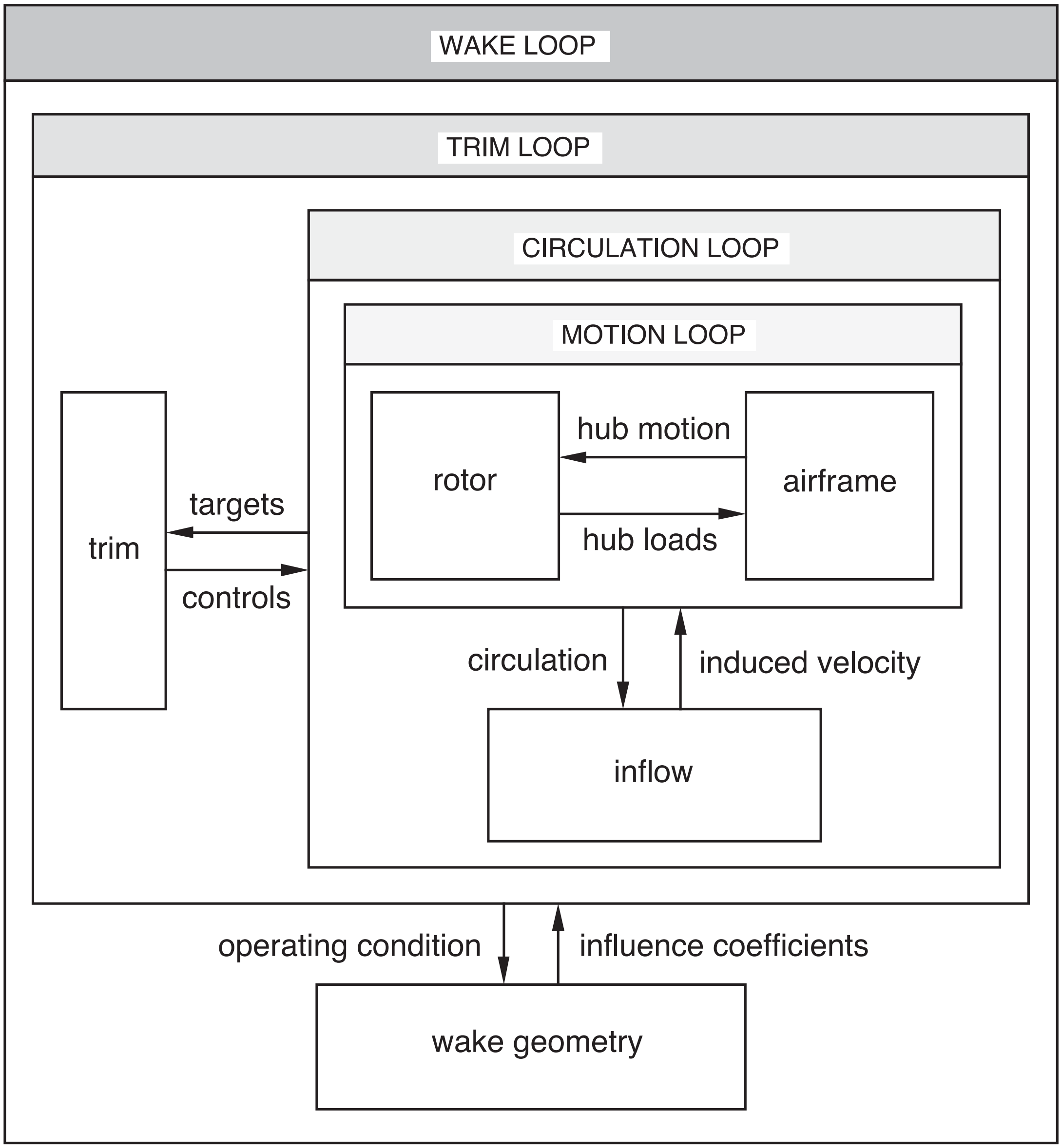


Figure 4-14a Basic loops of trim analysis (optional regulator loop not shown).

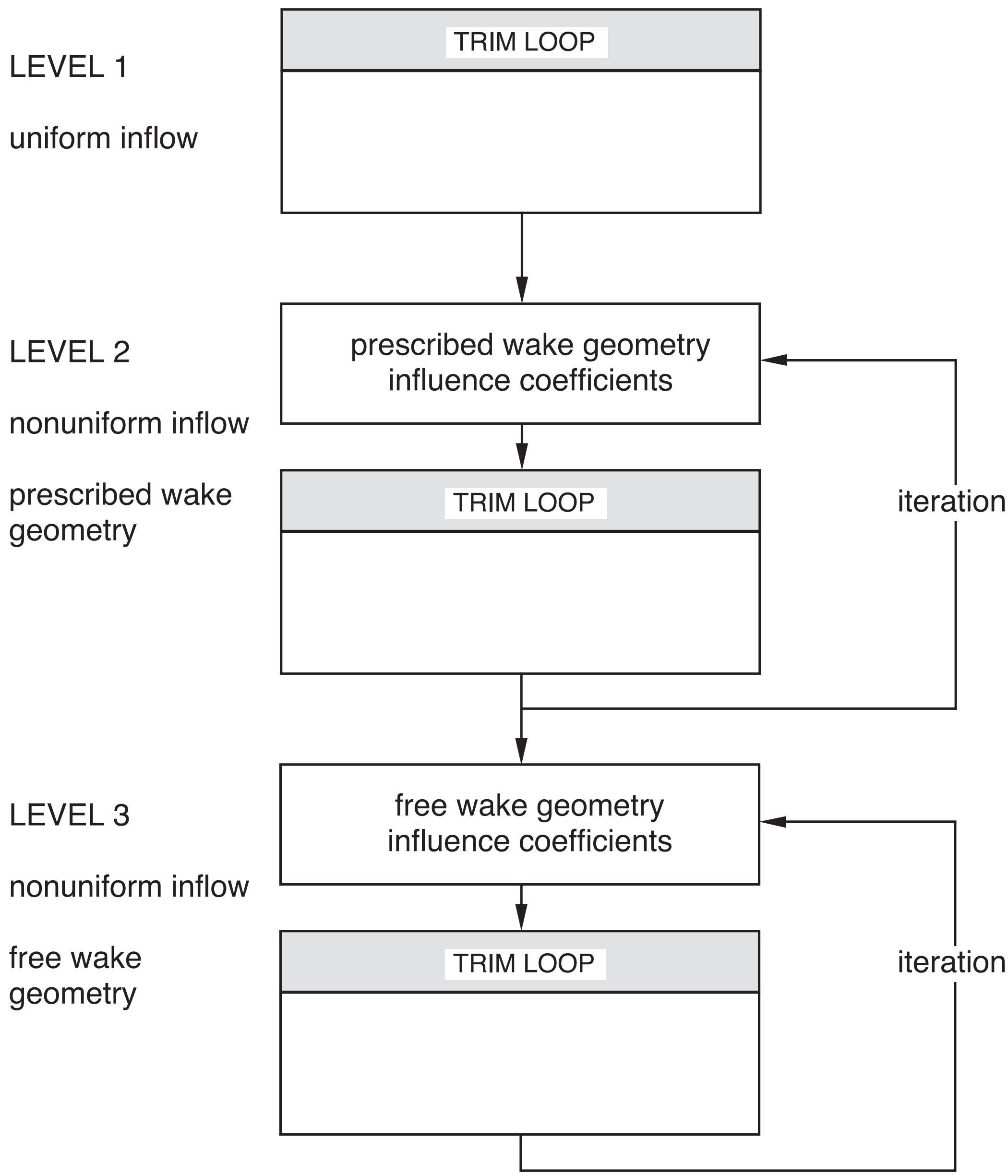


Figure 4-14b Levels of wake loop.

LOOPS AND THEIR SOLUTION METHODS:

loop name	solution method
`WAKE`	successive substitution
`TRIM`	Newton-Raphson
`REGULATOR`	regulator
`CIRCULATION`	successive substitution
`MOTION`	successive substitution
`AIRFRAME`	no solution

PART SOLUTION METHOD IS “HARMONIC” OR “TIME FINITE ELEMENT” FOR FOLLOWING:

```
ROTOR n HUB
ROTOR n BLADE m
AIRFRAME
DRIVE TRAIN
```

“IMPLICIT” METHOD USED FOR OTHER PARTS

4–18 System Pieces: Trim Loop

A LOOP CONTROLS EXECUTION OF PART SOLUTION AND WRITE MODULES, AND SOLVES NONLINEAR ALGEBRAIC EQUATIONS

LOOP IMPLEMENTS GENERAL ITERATION BETWEEN PART SOLUTIONS

ITERATION BETWEEN:
parts solved in loops
and all inner loops

ITERATION CONTROL AND CONVERGENCE TEST OPERATIONS CONSTITUTE LOOP ALGORITHM

figure 15 outlines loop solution process

LOOP DESCRIBED BY IDENTIFYING PARTS, CHILD LOOP, AND WRITE MODULES

PARTS SOLVED IN LOOP (ITERATED WITH CHILD LOOPS), OR AT END OF LOOP
parts are solved in order defined

LOOPS ARE HIERARCHICAL
loop can have at most one child and one parent;
trim task can have more than one set of nested loops

WRITE MODULES PRODUCE OUTPUT, FOR:

SHELL
CONVERGENCE
LOOP SOLUTION
PART SOLUTION
MODES
OUTPUT OF PART
OUTPUT
GRAPHICS

loop iteration
- iteration control
- solve parts in loop
- child loop
- test loop convergence

solve parts at end of loop
execute write modules at end of loop

Figure 4-15 Outline of loop solution process.

METHODS IMPLEMENTED FOR TRIM LOOPS:

NO SOLUTION
SUCCESSIVE SUBSTITUTION (with stages)
NEWTON RAPHSON (with identification)
REGULATOR (with identification)

NO SOLUTION METHOD

NO ITERATION CONTROL OR CONVERGENCE TEST OPERATIONS

LOOP JUST ORGANIZES PART SOLUTIONS

4-18.1 Theory

LOOP SOLVES NONLINEAR ALGEBRAIC EQUATIONS

CAN WRITE NONLINEAR ALGEBRAIC EQUATIONS IN TWO FORMS:

FIXED POINT, $x = G(x)$

ZERO POINT, $f(x) = 0$

with vectors x, G, and f

function G or f consists of part and child loop operations

REQUIRE EFFICIENT AND CONVERGENT METHODS TO FIND SOLUTION $x = \alpha$

FOR NONLINEAR PROBLEMS, METHOD WILL BE ITERATIVE:

$x_{n+1} = F(x_n)$

operation F depends on solution method

iteration will converge if F is not too sensitive to errors in x: converge if $|F'(\alpha)| < 1$

convergence is linear for $F' \neq 0$, quadratic (at least) for $F' = 0$

ITERATIVE METHODS WILL HAVE RELAXATION FACTOR (AND OTHER PARAMETERS) TO IMPROVE CONVERGENCE, AND TOLERANCE TO MEASURE CONVERGENCE

SUCCESSIVE SUBSTITUTION IS FIXED POINT METHOD

NEWTON RAPHSON IS ZERO POINT METHOD

4-18.2 Successive Substitution Method

SUCCESSIVE SUBSTITUTION LOOP EXECUTES GENERAL ITERATION BETWEEN PART SOLUTIONS

used for wake, circulation, and motion loops of rotorcraft shell

figure 16 outlines the method

SUCCESSIVE SUBSTITUTION ALGORITHM SOLVES EQUATIONS OF FORM $x = G(x)$ (FIXED POINT)

let $y = G_1(x)$ represent result of parts solved in loop
and $x = G_2(y)$ result of child loop solution

then $G = G_2(G_1)$, and x is defined by how system equations are split between this loop and its child loops

x is all quantities calculated by child loops, and used by parts solved in this loop

analysis finds x from functionality of part equations

ALGORITHM INCLUDES STAGES OR LEVELS: CYCLES OUTSIDE BASIC SUCCESSIVE SUBSTITUTION ITERATION

A PART SOLUTION CAN CHANGE ITS MODEL DEPENDING ON THE LOOP LEVEL

SO AN INITIALIZATION OR STARTUP PROCEDURE CAN BE IMPLEMENTED

used for wake loop

level cycle
- save: $x_{\text{old}} = x$
- successive substitution iteration
  - relax: $x = \lambda x + (1-\lambda)x_{\text{old}}$
  - save: $x_{\text{old}} = x$
  - evaluate x (solve parts in loop, child loops)
  - test convergence: $\text{error} = \|x - x_{\text{old}}\| \le \text{tolerance} \times \text{weight}$

Figure 4-16 Outline of successive substitution method.

THEORY

ITERATION

direct iteration is simply $x_{n+1} = G(x_n)$, but $|G'| > 1$ for many practical problems

relaxed iteration uses $F = (1-\lambda)x + \lambda G$:

$$x_{n+1} = (1-\lambda)x_n + \lambda G(x_n)$$

with relaxation factor λ; then convergence criterion is

$$|1 - \lambda + \lambda G'| < 1$$

so λ can be found to ensure convergence for any finite G'

CONVERGENCE TEST

correct solution $x = \alpha$ is not known, so convergence tested by comparing two successive iterations:

$$\text{error} = \|x_{n+1} - x_n\| \leq \text{tolerance}$$

typically error is absolute value of difference between iterations

effect of relaxation factor is to reduce difference between iterations, so reduction of λ must be accompanied by reduction in tolerance, to maintain same convergence accuracy

PARAMETERS CONTROLLING CONVERGENCE AND ACCURACY OF METHOD:

MAXIMUM NUMBER OF ITERATIONS
TOLERANCE
RELAXATION FACTOR

METHOD FAILS IF $G'(\alpha) = \infty$

typically then iteration oscillates about correct solution, magnitude of oscillation decreasing as λ approaches zero

but at $\lambda = 0$ the iteration is turned off, so correct solution can never be found

to analyze the system, necessary to change definition of problem:

change order that parts are solved in loop, thereby changing definition of x and G

or change physical model that is source of sensitivity of G to x

example: circulation loop, for hover near zero thrust

4-18.3 Newton-Raphson Method

NEWTON-RAPHSON LOOP EXECUTES GENERAL SOLUTION OF NONLINEAR ALGEBRAIC EQUATIONS

used for trim loop of rotorcraft shell

AVAILABLE TO IMPLEMENT SOLUTION OF INVERSE TRIM PROBLEM, IDENTIFYING PARAMETER VALUES REQUIRED TO ACHIEVE SPECIFIED OPERATING CONDITION

figure 17 outlines the method

NEWTON-RAPHSON ALGORITHM SOLVES EQUATIONS OF FORM $M(v) - M_{\text{target}} = 0$ FOR VARIABLES v (ZERO POINT)

quantities M can be

an input/output interface variable f_l

some mean system rigid force or moment F

force F corresponds to system rigid degrees of freedom; target is probably zero

variables v can be

some input variable u

a parameter in a wind, operating condition, or period system piece

typically used to define rigid motion of system

initialize
- evaluate M (solve parts in loop, child loops)
- test convergence: $\mathrm{error} = |M_j - M_{\mathrm{target}j}| \leq \mathrm{tolerance} \times \mathrm{weight}_j$
- initialize derivative matrix D to input matrix; $P = 0$
- calculate gain matrix: $C = \lambda D^{-1}$

iteration
- identify derivative matrix
  - optional perturbation identification
    - perturb each element of v: $\delta v_i = \Delta \times \mathrm{weight}_i$
    - evaluate M (solve parts in loop, child loops)
    - calculate D, $P = 1/\delta v_i^2$
    - calculate gain matrix: $C = \lambda D^{-1}$
  - optional recursive identification
    - update D, P
    - calculate gain matrix: $C = \lambda D^{-1}$
- increment controls: $\delta v = -C(M - M_{\mathrm{target}})$
- evaluate M (solve parts in loop, child loops)
- test convergence: $\mathrm{error} = |M_j - M_{\mathrm{target}j}| \leq \mathrm{tolerance} \times \mathrm{weight}_j$

Figure 4-17 Outline of Newton Raphson method.

THEORY

ITERATION

Taylor series expansion of $f(x) = M - M_{\text{target}} = 0$ produces iteration operator $F = x - f/f'$:

$$x_{n+1} = x_n - \left[f'(x_n)\right]^{-1} f(x_n)$$

which gives quadratic convergence; modified Newton-Raphson iteration is $F = x - Cf$:

$$x_{n+1} = x_n - Cf(x_n) = x_n - \lambda D^{-1} f(x_n)$$

where λ is relaxation factor, and derivative matrix D is estimate of f'

behavior of iteration depends on accuracy of f'; here analysis can evaluate directly f, but not f'; necessary to evaluate f' by numerical perturbation of f, and for efficiency derivatives may not be evaluated for each x_n

these approximations compromise convergence of method, so relaxation factor is introduced to compensate

convergence criterion is

$$|1 - Cf'| = |1 - \lambda D^{-1} f'| < 1$$

so iteration converges if $\lambda < 2D/f'$

INITIAL VALUES

Newton-Raphson method has good convergence when x is sufficiently close to the solution, but frequently has difficulty converging elsewhere

so initial estimate x_0 that starts iteration is important parameter affecting convergence

CONVERGENCE TEST

convergence of x tested in terms of required value (zero) for f:

$$\text{error} = \|f\| \leq \text{tolerance}$$

typically error is absolute value of f

DERIVATIVE MATRIX IDENTIFICATION

derivative matrix D obtained by identification process, either perturbation or recursive

perturbation identification can be performed at beginning of the loop, or at the beginning and again every M_{PID} iterations

or never, using an input derivative matrix

recursive identification can be performed never, or at each iteration (except when the perturbation identification is performed)

perturbation identification restarts the recursive identification

PERTURBATION IDENTIFICATION

D calculated from one-step or two-step finite-difference expression; each element x_i of x is perturbed, one at a time, giving the i-th column of D:

$$D = \left[\cdots \quad \frac{\partial f}{\partial x_i} \quad \cdots\right] = \left[\cdots \quad \frac{f(x_i + \delta x_i) - f(x_i)}{\delta x_i} \quad \cdots\right]$$

accuracy of D (hence convergence) affected by magnitude and sign of perturbation

RECURSIVE IDENTIFICATION

iteration makes change Δx, which then produces change Δf; these increments contain new information about $f'(x)$, which can be used to update D

weighted least-squares solution for D can be used

with equal weight on changes from each iteration, identification eventually turns itself off, since latest change is worth less and less compared to complete set of changes

latest information should be most important

so exponential window method used: weight on change from iteration k steps in past is α^k, where $0 < \alpha < 1$

adding one measurement (with weight $1/\alpha$) to current estimate of D produces recursive identification algorithm

algorithm uses error variance matrix P; if calculation of P diverges, perturbation identification can be performed more often, to reinitialize P

or Broyden's formula can be used

updated derivative matrix satisfies $D_n \Delta x_n = \Delta f_n$

PARAMETERS CONTROLLING CONVERGENCE AND ACCURACY OF METHOD:

- MAXIMUM NUMBER OF ITERATIONS
- TOLERANCE
- RELAXATION FACTOR
- INITIAL VALUES

- PERTURBATION IDENTIFICATION
  - how often
  - difference order
  - magnitude and sign of control step

- RECURSIVE IDENTIFICATION
  - method
  - exponential window weight

4-18.4 Regulator Method

REGULATOR LOOP EXECUTES OPTIMIZATION OF SYSTEM SOLUTION

algorithm similar to (and developed from) equations of automatic control system, but as a trim loop it functions as optimizer rather than modelling actual controller

controller can be part of system, defined using available components

such a controller must be defined in order to analyze its behavior in transient or flutter task, and it could be defined also for trim task

REGULATOR ALGORITHM USES A SELF-TUNING REGULATOR SOLUTION METHOD TO FIND THE CONTROL θ TO MINIMIZE COST FUNCTION $J(z,\theta)$, CALCULATED FROM MEASUREMENTS z:

self-tuning regulator is system that combines recursive parameter estimation with linear feedback; cost function is

$$J = (z_n - z_{\text{target}})^T W_z (z_n - z_{\text{target}}) + \Delta\theta_n^T W_{\Delta\theta} \Delta\theta_n$$

where W_z and $W_{\Delta\theta}$ are diagonal weighting matrices

weighting matrices must also account for differences in units among the elements of z and θ

last term in J limits rate of change of θ, improving stability without changing converged value

measurements z can be an input/output interface variable f_l

control θ can be some input variable u

PROCESS, THEORY, PARAMETERS SIMILAR TO NEWTON-RAPHSON METHOD

change in θ required to minimize J found assuming a local linearization of $z(\theta)$:

$$z_n = z_{n-1} + T(\theta_n - \theta_{n-1})$$

number of measurements z can be greater than number of controls θ

4–19 System Pieces: Trim Part

METHODS IMPLEMENTED FOR TRIM PARTS:

- NO SOLUTION
- IMPLICIT
  - equations solved in passes
  - single value, periodic, not periodic, multiperiod
- STATIC
  - solve differential equations
- HARMONIC, TIME FINITE ELEMENT
  - solve differential equations
  - one or more periods; option filter response

PART DESCRIBED BY IDENTIFYING ITS MOTION, CONSTRAINT, AND OUTPUT EQUATIONS

- EACH EQUATION CAN BE IN ONLY ONE PART (POSSIBLY NO-SOLUTION METHOD)
- EACH PART CAN BE SOLVED IN ONE OR MORE LOOPS

POSSIBLE FOR COMPONENT TO EXECUTE SOLUTION PROCEDURE WHEN IT EVALUATES COMPONENT OUTPUT

- IMPLICIT SOLUTION FOR SINGLE VALUE CAN BE USED

NO SOLUTION METHOD

- EQUATIONS ASSIGNED TO PART ARE NOT SOLVED
  - corresponding variables will be zero
- SO SUPPRESSES EFFECT OF THESE SYSTEM VARIABLES DURING TRIM TASK

4-19.1 Filtered Solution

SYSTEM CAN HAVE SYMMETRIES THAT IN CERTAIN OPERATING CONDITIONS IMPLY RESTRICTIONS ON FORM OF SOLUTION

CERTAIN OF THESE RESTRICTIONS ARE ENFORCED WHEN RESPONSE IS EVALUATED BY PART SOLUTION

at end of time step or end of iteration;
if permitted by response definition

FOR ROTATING, AXISYMMETRIC SYSTEM (TYPICALLY N IDENTICAL, EQUALLY SPACED SUBSYSTEMS):

CERTAIN HARMONICS OF SOLUTION SHOULD BE ZERO

so trim filter can be defined for harmonic or time finite element solution method

OTHER SYMMETRIES

SYMMETRY ABOUT A PLANE (SUCH AS LATERAL SYMMETRY)

typically asymmetric motions should be zero;
implement using standard order reduction

RELATIONS BETWEEN SOLUTIONS FOR VARIOUS SUBSYSTEMS

enforce by defining trim response of one subsystem as child of another subsystem (specified in response system piece)

TRIM CHILD SOLUTION

DEGREES OF FREEDOM (INCLUDING MODES) CAN ALSO BE DESIGNATED CHILD DEGREES OF FREEDOM FOR THE TRIM PART SOLUTION

after the equations are solved, child solution replaced in trim part by parent (or averaged parent) solution

GIMBAL OR TEETER DEGREES OF FREEDOM

certain harmonics of solution set to zero as well

THIS OPTION IS AVAILABLE FOR SYSTEMS THAT MUST BE SOLVED AS A WHOLE, EVEN THOUGH SYMMETRIES IMPLY IDENTICAL (WITH PHASE SHIFTS) MOTION OF SUBSYSTEMS

4-19.2 Implicit Solution Method

IMPLICIT METHOD DIRECTLY SOLVES CONSTRAINT AND OUTPUT EQUATIONS

PART HAS COMPONENT EVALUATE REQUIRED COMPONENT OUTPUT VECTOR, AT SET OF TIME VALUES

component will use component input at current time, and perhaps at past times as well

set of time values can be specified as single value, periodic, not periodic, or multiperiod

figure 18 outlines the method

ORDER IN WHICH EQUATIONS ARE SOLVED:

SPECIFIED BY LOCATION OF PART IN LOOPS, AND WITHIN PART BY AN ASSIGNED SOLUTION PASS

since each interface variable will affect other constraint equations, important that equations be solved in proper order

```
for each time value
    constraint equations
        for each solution pass
        for each vector
            evaluate component output: total f_l = x
            translate total to difference
    output equations
        for each vector
            evaluate component output: total y_q = x
            translate total to difference
implement order reduction: f_l, y_q
```

Figure 4-18 Outline of implicit solution method.

TIME VALUES

SINGLE VALUE

solved at one time; often solution independent of time

PERIODIC

solved at J times in revolution; time defined in terms of azimuth $\psi_j = \Omega t_j + \psi_0$, with frequency Ω obtained from some period of system

NOT PERIODIC

solved at $K + 1$ times, from t_B to t_E

MULTIPERIOD

solved at $K + 1$ times, from t_B to t_E; time defined in terms of azimuth $\psi_j = \Omega t_j + \psi_0$, with frequency Ω obtained from some period of system; and solved at J azimuth steps per revolution

4-19.3 Differential Equations

HARMONIC, TIME FINITE ELEMENT, AND STATIC SOLUTION METHODS SOLVE DIFFERENTIAL EQUATIONS

MATRICES OF DIFFERENTIAL EQUATIONS

response = nominal/reference + difference

nominal/reference specified by input
reference can be updated from difference
part solves for difference

MOTION AND CONSTRAINT EQUATIONS LINEARIZED ABOUT NOMINAL/REFERENCE OF RESPONSE

FOR TRIM, MATRICES ARE CONSTANT

AVERAGED OVER PERIOD,
OR EVALUATED AT ONE TIME

OPTIONALLY REFERENCE CAN BE UPDATED (FROM CURRENT SOLUTION FOR DIFFERENCE) AND MATRICES RECALCULATED DURING SOLUTION PROCESS

an update requires additional computation, but may improve convergence

best update strategy, minimizing computation while ensuring convergence, will be problem dependent

OPTIONS FOR MATRIX UPDATE STRATEGY:

iterations in part solution are counted from beginning of system solution, or from beginning of part

always references are updated and matrices calculated for first iteration (system or part count)

they may be updated for second iteration

thereafter they may be updated never, or every N iterations

where update interval N may be constant,
increased by specified number after each update,
or increased by factor of 2 after each update

any modes used by part may or may not be recalculated at each update

MOST EFFICIENT TO CALCULATE MATRICES JUST ONCE

UPDATE FOR SECOND ITERATION MAY BE NECESSARY TO GET EFFECT OF INTERFACE FORCES, SUCH AS CENTRIFUGAL FORCE OF ROTOR BLADE

THEORY

PART HAS COMPONENT EVALUATE NONLINEAR MOTION AND CONSTRAINT EQUATIONS

$$0 = \widetilde{A}$$

PRODUCT OF MATRICES AND RESPONSE DIFFERENCE ADDED TO BOTH SIDES OF NONLINEAR EQUATIONS

$$H\xi = (\widetilde{A} + H\xi) = A$$

no approximation yet — equations are still exact

linearized part of equations needed to solve nonlinear equations

inaccuracies in matrices will affect convergence

part solution method may introduce approximation (such as quasistatic reduction), or treat two sides differently

DIFFERENTIAL EQUATIONS

$$\begin{bmatrix} H & \phi_{f1}^T & \phi_{f0}^T \\ \phi_{x1} & G_1 & 0 \\ \phi_{x0} & 0 & 0 \end{bmatrix} \begin{pmatrix} \xi \\ f_1 \\ f_0 \end{pmatrix} = \begin{pmatrix} A \\ B_1 \\ B_0 \end{pmatrix}$$

$$y = B_y$$

constraint equations and interface variables:

f_0 = structural dynamic interfaces

f_1 = input/output interfaces

H is differential operator (mass, damping, spring)

CONSTRAINT EQUATIONS SOLVED TO ELIMINATE PART VARIABLES

- structural dynamic constraint equations solved to eliminate degrees of freedom ξ_b
  - assume interfaces independent, and ξ_b properly identified, so constraint equations are full rank
  - so number of degrees of freedom is reduced (and solution for interface forces is decoupled from equations of motion)
- input/output constraint equations solved to eliminate interface variables f_1
  - does not reduce number of degrees of freedom
- degrees of freedom ξ partitioned into ξ_b and ξ_c
  - $\eta_1 = \xi_c$ ARE DEGREES OF FREEDOM REMAINING AFTER CONSTRAINT EQUATIONS ARE SOLVED
    - typically ξ_b are component rigid body degrees of freedom, plus elastic or joint degrees of freedom for multiple load paths
    - so η_1 are remaining elastic and joint degrees of freedom, plus system rigid degrees of freedom

MODAL TRANSFORM

PART CAN HAVE ONE OR MORE MODE SETS

modal transform obtained from mass and spring of structural dynamic subset of equations of part

constant modal matrix Φ, transforming degrees of freedom η_1

Φ defines mode shapes in terms of degrees of freedom η_1, which probably do not measure total displacement of structure

modal frequency, mass, and output are calculated and printed (to identify modes), but only Φ is used in solution

TRANSFORM $\eta_1 = \Phi\theta$, AND MULTIPLY EQUATIONS FOR η_1 BY Φ^T

θ are modal degrees of freedom ($\theta = \eta_1$ if there is no modal transform)

ADD MODAL STRUCTURAL DAMPING

diagonal viscous damping matrix D with elements calculated from modal mass and spring, and input structural damping factor g ($d = g\sqrt{mk}$, so g is twice critical damping ratio)

MODAL TRANSFORM (WITHOUT TRUNCATION) DOES NOT INTRODUCE APPROXIMATION, AS LONG AS Φ IS COMPLETE MATRIX

MODAL TRANSFORM USED BECAUSE ALLOWS MODAL TRUNCATION AND/OR ADDITION OF MODAL DAMPING

ORDER REDUCTION

order reduction specified for elements of degrees of freedom ξ, or for modal degrees of freedom

so partition θ:

dynamic θ_k	solve complete equations
quasistatic θ_l	truncate dynamic terms
zero θ_z	truncate dynamic and static terms

zero θ_z handled by omitting or ignoring equations, and setting response to zero

quasistatic reduction implemented by neglecting mass and damping of θ_l equations, then introducing transformation that decouples θ_l and θ_k in the spring matrix

REDUCED EQUATIONS

part differential equations take form

$$h_{kk}\theta_k = \widetilde{A}_k \qquad \text{(dynamic)}$$
$$K_{ll}\widetilde{\theta}_l = A_{l\mathrm{QS}} \qquad \text{(static)}$$

forces required are obtained from A, B_1, and B_0

these equations solved for degrees of freedom θ_k and $\widetilde{\theta}_l$ then response evaluated for ξ, f_0, and f_1

PART SOLUTION METHOD MUST SOLVE THESE REDUCED EQUATIONS

4-19.4 Harmonic Solution Method

HARMONIC SOLUTION METHOD SOLVES DIFFERENTIAL EQUATIONS FOR PERIODIC RESPONSE, IN TERMS OF A FOURIER SERIES

- PERIOD SYSTEM PIECE IDENTIFIED TO SPECIFY FUNDAMENTAL FREQUENCY
- FOR TIME INVARIANT PART, SOLUTION CAN BE OBTAINED FOR MORE THAN ONE PERIOD
  - figure 19 outlines the method

HARMONIC SOLUTION METHOD SOLVES CONSTANT-COEFFICIENT DIFFERENTIAL EQUATIONS FOR RESPONSE DIFFERENCE

- ASSUMED THAT SOLUTION IS PERIODIC, SO RESPONSE CAN BE REPRESENTED BY FOURIER SERIES
  - frequency domain enforces periodicity of solution
  - and allows use of large time step, if solution adequately described by small number of harmonics
- IF ACTUAL EQUATIONS ARE TIME-VARYING (PERIODIC), MATRICES MUST BE AVERAGED
  - by assuming time-invariant matrices, equations can be solved separately for each harmonic
  - all interharmonic coupling is in forces acting on equations

initialize: $F_{\text{old}} = 0$
    evaluate implicit constraint equations
iteration
    save response: $x_{\text{old}} = x$
    calculate matrices
    for each time value (over period)
        evaluate equations: F
        relax equations: $F = \lambda F + (1 - \lambda) F_{\text{old}}$
        save equations: $F_{\text{old}} = F$
        calculate and harmonically analyze forces of reduced eqns
        solve equations for degrees of freedom θ
        calculate and filter response: x
    test convergence: $\text{error} = \|x - x_{\text{old}}\| \le \text{tolerance} \times \text{weight}$
output
    for each time value (over period)
        evaluate equations: B_y
        harmonically analyze equations
        calculate and filter response: y
implement order reduction: ξ, f_0, f_1, y, f_l

Figure 4-19 Outline of harmonic solution method.

TRIM FILTER OPERATIONS

RESPONSE CAN BE FILTERED WHEN EVALUATED AT END OF TIME STEP

OPTIONS

suppress all but pN harmonics, where N is specified fundamental harmonic, and p an integer; that is, suppress all harmonics not a multiple of N

for rotorcraft airframe, N = number of blades

suppress all but $pN \pm 1$ harmonics

for gimbal or teeter motion of hub (in rotating frame), N = number of blades

used to enforce true symmetries of system

CAN ALSO FILTER DEGREES OF FREEDOM WHEN EQUATIONS ARE SOLVED

used for approximate solution, such as simulated gimbal or teeter hinge model

SET HARMONICS TO ZERO AND IGNORE CORRESPONDING EQUATIONS WHILE EQUATIONS ARE BEING SOLVED

THEORY

REDUCED EQUATIONS OF MOTION

differential equations for θ_k, and static equations for θ_l:

$$h_{kk}\theta_k = \widetilde{A}_k \qquad \text{(dynamic)}$$
$$K_{ll}\widetilde{\theta}_l = A_{l\mathrm{QS}} \qquad \text{(static)}$$

equations have form $Hu = R$

$H = H(d/dt) =$ differential operator (spring, damping, and mass) for θ_k

$H = K =$ static operator (spring) for θ_l

in general, R depends on time and on u and its derivatives, and problem is nonlinear

FOURIER SERIES

azimuth and time related by $\psi = \Omega t + \psi_0$, with frequency Ω from part solution period

assume cosine-sine Fourier series representation of solution for u:

$$u = u_0 + \sum_{n=1}^{N} u_{nc} \cos n\psi + u_{ns} \sin n\psi$$

all solution variables of part (ξ, f_1, f_0, and y) are represented by Fourier series

FORCE

force $R_j = R(\psi_j)$ evaluated at J azimuth steps per revolution

harmonics R_n obtained from R_j over one period

interpolation options (factor K_n in harmonic analysis)

Fourier interpolation

linear interpolation (default): harmonics of linear interpolation between force at discrete times

DAMPING IN MATRICES

matrices of equations of motion might not include all important damping sources

to improve convergence, estimate of damping can be added to both sides of equation for θ_k

diagonal damping matrix D, calculated from diagonal mass and spring of h_{kk}: $d = 2\zeta\sqrt{mk}$, where ζ is input critical damping coefficient

no approximation (not numerical damping), since algorithm treats both sides of equation same

will affect convergence but not final result

equation for harmonics (complex form):

$$H_n u_n = [H(in) + inK_n D]\, u_n = (R + D\dot{u})_n = R_n$$

damping term added to R before harmonic analysis performed

factor K_n (from interpolation option) included so added damping treated exactly same on both sides

ITERATION

by assuming Fourier series representation of solution, differential equations are converted to algebraic equations, to be solved for harmonics

equations take form $x = G(x)$, where x is force R_j over a period

so successive substitution method applicable

each iteration of algorithm finds solution over period, hence has a time loop

at each time step over period, evaluate force R_j

with relaxation factor for convergence of nonlinear problems

$$R_j = \lambda R_j + (1 - \lambda) R_{j\mathrm{old}}$$

old force is from previous iteration; R_{old} initialized to zero at beginning of system solution, so with $\lambda \neq 1$ the part solution has asymptotic convergence for constant force R

then damping term added to R_j, and harmonics R_n evaluated

solution for degrees of freedom is $u_n = H_n^{-1} R_n$

at end of each iteration (end of each period), convergence is tested

CONVERGENCE TEST

correct solution for u is not known, so convergence tested by comparing two successive iterations:

$$\text{error} = \|u - u_{\text{old}}\| \leq \text{tolerance}$$

typically error is rms of difference between iterations

PARAMETERS CONTROLLING CONVERGENCE AND ACCURACY OF METHOD:

MAXIMUM NUMBER OF ITERATIONS
TOLERANCE
RELAXATION FACTOR
DAMPING

NUMBER OF AZIMUTH STEPS PER REVOLUTION

NUMBER OF HARMONICS

MATRIX CALCULATION
- perturbation order
- magnitude and sign of perturbation

MATRIX UPDATE STRATEGY

4-19.5 Time Finite Element Solution Method

TIME FINITE ELEMENT SOLUTION METHOD SOLVES DIFFERENTIAL EQUATIONS FOR PERIODIC RESPONSE, IN TERMS OF A FOURIER SERIES (AS SHAPE FUNCTIONS)

- harmonics of equations of motion solved for harmonics of response, using Newton-Raphson algorithm
- PERIOD SYSTEM PIECE IDENTIFIED TO SPECIFY FUNDAMENTAL FREQUENCY
- FOR TIME INVARIANT PART, SOLUTION CAN BE OBTAINED FOR MORE THAN ONE PERIOD
  - figure 20 outlines the method

PROCESS SIMILAR TO HARMONIC METHOD IN MANY RESPECTS

- WITH NEWTON-RAPHSON ITERATION

TIME FINITE ELEMENT SOLUTION METHOD SOLVES CONSTANT-COEFFICIENT DIFFERENTIAL EQUATIONS FOR RESPONSE DIFFERENCE

- ASSUMED THAT SOLUTION IS PERIODIC, SO RESPONSE CAN BE REPRESENTED BY FOURIER SERIES (AS SHAPE FUNCTIONS)
- IF ACTUAL EQUATIONS ARE TIME-VARYING (PERIODIC), MATRICES MUST BE AVERAGED

initialize: $F_{\text{old}} = 0$; evaluate $\widehat{R}$; C from H, $P = 0$
    evaluate implicit constraint equations
iteration
    save response: $x_{\text{old}} = x$
    calculate matrices
    identify derivative matrix
        optional perturbation identification
            perturb each element of u: $\delta u_i = \Delta \times \text{weight}_i$
            evaluate $\widehat{R}$ (without relaxation of F)
            calculate D, $P = 1/\delta u_i^2$
            calculate gain matrix: $C = \lambda_N D^{-1}$
        optional recursive identification
            update P, C
    increment degrees of freedom: $\delta u = -C\widehat{R}$
    evaluate $\widehat{R}$
        calculate and filter response: x
        for each time value (over period)
            evaluate equations: F
            relax equations: $F = \lambda F + (1-\lambda) F_{\text{old}}$
            save equations: $F_{\text{old}} = F$
        calculate and harmonically analyze forces of reduced eqns
        calculate equation residual $\widehat{R}$
    test convergence: $\text{error} = \|x - x_{\text{old}}\| \leq \text{tolerance} \times \text{weight}$
output
    for each time value (over period)
        evaluate equations: B_y
        harmonically analyze equations
        calculate and filter response: y
implement order reduction: ξ, f_0, f_1, y, f_l

Figure 4-20 Outline of time finite element solution method.

THEORY

REDUCED EQUATIONS OF MOTION

differential equations for θ_k, and static equations for θ_l:

$$h_{kk}\theta_k = \widetilde{A}_k \qquad \text{(dynamic)}$$
$$K_{ll}\widetilde{\theta}_l = A_{l\mathrm{QS}} \qquad \text{(static)}$$

equations have form $Hu = R$ or $\widehat{R} = R - Hu = 0$

$H = H(d/dt) =$ differential operator (spring, damping, and mass) for θ_k

$H = K =$ static operator (spring) for θ_l

TIME FINITE ELEMENTS

Hamilton's equation for periodic structural dynamic system

$$0 = \delta \int_0^T L\,dt = \int_0^T \delta u^T \widehat{R}\,dt$$

finite element method expands response as $u(t) = h(t)^T q$

h = shape functions in time
q = finite element variables

equations to be solved for q are then

$$0 = \int_0^T h^T \widehat{R}\,dt$$

can divide period into N_t elements, with polynomial shape functions over each element

disadvantage: many q variables required even for motion that only contains few harmonics

so here use harmonics for shape functions

q = coefficients in Fourier series for u

solve harmonics of equations of motion for q, using Newton-Raphson method

FOURIER SERIES

azimuth and time related by $\psi = \Omega t + \psi_0$, with frequency Ω from part solution period

assume cosine-sine Fourier series representation of solution for u:

$$u = u_0 + \sum_{n=1}^{N} u_{nc} \cos n\psi + u_{ns} \sin n\psi$$

all solution variables of part (ξ, f_1, f_0, and y) are represented by Fourier series

FORCE

force $R_j = R(\psi_j)$ evaluated at J azimuth steps per revolution

harmonics R_n obtained from R_j over one period

interpolation options (factor K_n in harmonic analysis)

Fourier interpolation

linear interpolation (default): harmonics of linear interpolation between force at discrete times

equation residuals (complex form):

$$\widehat{R}_n = R_n - H_n u_n$$

NEWTON-RAPHSON ITERATION

by introducing time finite elements, differential equations are converted to algebraic equations, to be solved for harmonics

equations take form $f(x) = 0$, where x are harmonics of motion and f are harmonics of residual equations

so Newton-Raphson method applicable

Newton-Raphson algorithm

$$x_{m+1} = x_m - Cf(x_m) = x_m - \lambda_N D^{-1} f(x_m)$$

with relaxation factor λ_N (defined separately from relaxation factor for forces)

and derivative matrix D (estimate of f')

derivative matrix D

initialize using matrix H_n

for cases without perturbation identification; include damping estimate in order to avoid singularities at resonant harmonics; gives iteration similar to harmonic method

optional perturbation identification

at beginning of part or system solution; optionally repeated every M_{PID} iterations

optional recursive identification

exponential window or Broyden

ITERATION

each iteration of algorithm finds solution over period, hence has a time loop

at each time step over period, evaluate force R_j

with relaxation factor for convergence of nonlinear problems

$$R_j = \lambda R_j + (1 - \lambda) R_{j\text{old}}$$

then harmonics R_n are evaluated, and residual equations $\widehat{R}_n$ calculated

solution for degrees of freedom updated using $\delta u = -C\widehat{R}$

at end of each iteration (end of each period), convergence is tested

CONVERGENCE TEST

correct solution for u is not known, so convergence tested by comparing two successive iterations:

$$\text{error} = \|u - u_{\text{old}}\| \leq \text{tolerance}$$

typically error is rms of difference between iterations

PARAMETERS CONTROLLING CONVERGENCE AND ACCURACY OF METHOD:

MAXIMUM NUMBER OF ITERATIONS
TOLERANCE
RELAXATION FACTORS
DAMPING (FOR INITIAL DERIVATIVE MATRIX)

PERTURBATION IDENTIFICATION
- how often
- difference order
- mag and sign of degree of freedom step

RECURSIVE IDENTIFICATION
- method
- exponential window weight

NUMBER OF AZIMUTH STEPS PER REVOLUTION

NUMBER OF HARMONICS

MATRIX CALCULATION
- perturbation order
- magnitude and sign of perturbation

MATRIX UPDATE STRATEGY

4–20 System Pieces: Transform

PROVIDES STANDARD MEANS TO DEFINE TRANSFORMATIONS OF SYSTEM VARIABLES AND EQUATIONS

- DEGREES OF FREEDOM, MODES, OUTPUT VARIABLES, OR INPUT VARIABLES
  - for each transformed variable, can specify order reduction (zero, dynamic, or quasistatic)

IN TRIM AND TRANSIENT AND FLUTTER TASKS

TRANSFORMS IMPLEMENTED:

- MULTIBLADE COORDINATES
- ROTATING-TO-NONROTATING
- SYMMETRIC/ANTISYMMETRIC

MULTIBLADE COORDINATE TRANSFORM

INTENDED FOR ROTATING SYSTEMS WITH AXISYMMETRIC GEOMETRY BUT ARBITRARY MOTION

most useful for configurations with N equally-spaced rotating subsystems

used for rotor blades

can be applied to configuration with blades that are not equally-spaced or not identical; if blade-to-blade differences are small, still expect to obtain benefits of transform

ELIMINATES PERIODIC COEFFICIENTS ($N \geq 3$)

FROM COUPLING OF ROTATING AND NONROTATING SYSTEM

for coupled rotor and airframe motions

BY AVERAGING WEAKLY PERIODIC SYSTEM

for aerodynamics of edgewise-moving rotor

SO HAVE TIME-INVARIANT EQUATIONS FOR FLUTTER AND TRANSIENT ANALYSES

ELIMINATES 1/REV BLADE MOTION IN FLIGHT DYNAMICS PROBLEMS

transformed (nonrotating) degrees of freedom vary more slowly than rotating degrees of freedom

SO CAN PERHAPS USE LARGER TIME STEP IN TRANSIENT ANALYSIS

THEORY

rotating system with N equally spaced, identical blades

m-th blade at azimuth $\psi_m = \psi + m\,\Delta\psi$, $\Delta\psi = 2\pi/N$ for $m = 1$ to N

$\psi = \Omega t + \psi_0$ is prescribed, reference azimuth

vector of rotating frame variables (length N, one for each blade): $x_{\rm rot} = (x_m)$

vector of corresponding nonrotating variables or multiblade coordinates (also length N):

$$x_{\rm non} = \begin{pmatrix} x_0 \\ x_{nc} \\ x_{ns} \\ x_{N/2} \end{pmatrix}$$

with $n = 1$ to $(N-1)/2$ for N odd;
or $n = 1$ to $(N-2)/2$ for N even

x_0 is collective coordinate;
x_{1c} and x_{1s} are cyclic coordinates

transformation: $x_{\rm rot} = T x_{\rm non}$

m-th row of T is

$$t_m = \begin{pmatrix} 1 & \cos k\psi_m & \sin k\psi_m & -1^m \end{pmatrix}$$

so

$$x_m = x_0 + \textstyle\sum_n (x_{nc}\cos k\psi_m + x_{ns}\sin k\psi_m) + x_{N/2}(-1)^m$$

ROTATING-TO-NONROTATING TRANSFORM

INTENDED FOR VARIABLES SUCH AS DEGREES OF FREEDOM OF ROTATING-FRAME GIMBAL OR UNIVERSAL JOINT

THEORY

rotating frame variables $x_{\mathrm{rot}} = (x_A \quad x_B)^T$

nonrotating frame variables $x_{\mathrm{non}} = (x_c \quad x_s)^T$

typically motion or force variables about two perpendicular axes in plane of rotation

$\psi = \Omega t + \psi_0$ is prescribed, reference azimuth

transformation: $x_{\mathrm{rot}} = T x_{\mathrm{non}}$

$$\begin{pmatrix} x_A \\ x_B \end{pmatrix} = \begin{bmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{bmatrix} \begin{pmatrix} x_c \\ x_s \end{pmatrix}$$

SYMMETRIC/ANTISYMMETRIC TRANSFORM

INTENDED FOR SYSTEMS WITH PLANE OF SYMMETRY

ONLY USED IN FLUTTER TASK

analysis of flutter equations can be performed separately for symmetric and antisymmetric subsets of the system

used for tilting proprotor aircraft

THEORY

right and left variables (lateral symmetry): $x = (x_R \quad x_L)^T$

symmetric and antisymmetric variables: $y = (y_S \quad y_A)^T$

transformation: $x = Ty$

$$\begin{pmatrix} x_R \\ x_L \end{pmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{pmatrix} y_S \\ y_A \end{pmatrix}$$

4–21 System Pieces: Modes

PROVIDES STANDARD MEANS TO DEFINE MODAL TRANSFORMATION OF STRUCTURAL DYNAMIC DEGREES OF FREEDOM

- FOR USE BY PART SOLUTION PROCEDURE
  - constant modal matrix Φ to transform reduced degrees of freedom and equations
  - can specify order reduction for each modal variable (zero, dynamic, or quasistatic)
- IN TRIM, TRANSIENT, AND FLUTTER TASKS

MODAL TRANSFORMATION

- MODAL TRANSFORMATION FOLLOWED BY MODAL TRUNCATION CAN MAKE ANALYSIS MORE EFFICIENT (IF ACCURACY IS RETAINED)
  - BY REDUCING NUMBER OF DEGREES OF FREEDOM INVOLVED IN DYNAMICS
  - modal transformation also allows introduction of modal damping
- FOR LINEARIZED PROBLEM (FLUTTER TASK)
  - modal truncation reduces size of system, and helps interpretation of results

FOR NONLINEAR PROBLEM (TRIM AND TRANSIENT TASKS)

still must evaluate all equations and forces of components and interfaces, so size of problem not reduced

transform does restrict states to just those excited by environment

MODAL TRANSFORMATION HELPS CONVERGENCE OF TRIM TASK WITH ELASTIC ROTOR BLADE MODEL

FOR EACH NEW PROBLEM, MUST DETERMINE NUMBER OF MODES REQUIRED FOR ACCURATE SOLUTION

if truncation does not improve efficiency or convergence, and do not need modal damping, no reason to use modes

ANALYSIS PRINTS MODAL FREQUENCIES AND MASSES, AND MODE SHAPES FROM OUTPUT VARIABLES

used to identify modes

none of printed information is used in part solution

part solution only uses modal matrix Φ (mode shapes for part degrees of freedom, not for output variables)

figure 21 illustrates part degrees of freedom and output variables in mode set

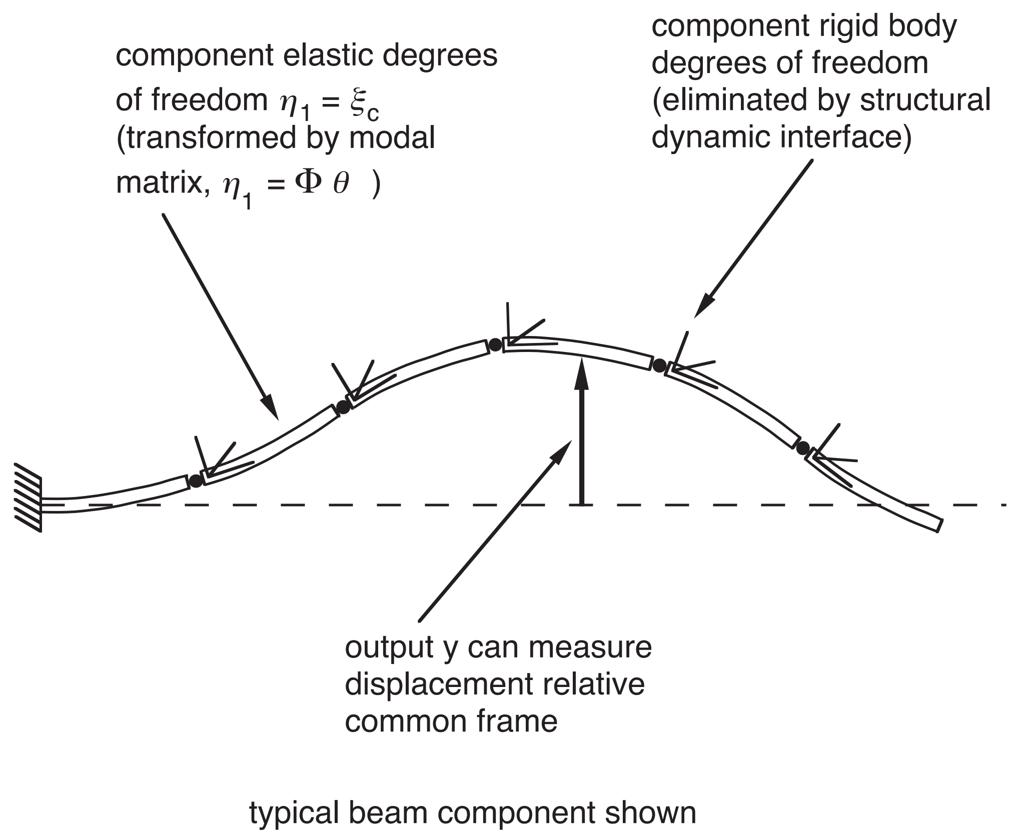


Figure 4-21 Part degrees of freedom and output variables in mode set.

MODE SETS

TRIM, TRANSIENT, OR FLUTTER PART CAN USE ONE OR MORE MODE SETS

IDENTIFY MOTION, CONSTRAINT, AND OUTPUT EQUATIONS

motion and constraint equations from only structural dynamic components and interfaces, subset of part equations

linearized about current reference solution

output equations only used to normalize and print mode shapes

AS IMPLEMENTED, MODES MUST BE FOR CONSTRAINED SUBSYSTEM

so mode set must include constraint equations that eliminate any rigid degrees of freedom of subsystem

NORMALIZATION

MODES ARE ORDERED BY FREQUENCY

OUTPUT VARIABLES y_q OF MODE SET USED TO NORMALIZE AND PRINT MODES SHAPE

typically y_q includes displacement at key points of subsystem

4–22 System Pieces: Response

PROVIDES STANDARD MEANS TO DEFINE CHARACTERISTICS OF SYSTEM VARIABLES

RESPONSE REQUIRED FOR

COMPONENT DEGREES OF FREEDOM ξ
FRAMES β
INTERFACE VARIABLES f_l
INPUT VARIABLES u_m
OUTPUT VARIABLES y_q

defined by vector

RESPONSE SPECIFICATION INCLUDES:

NOMINAL, REFERENCE, AND INITIAL DIFFERENCE
including rest position, rotating frame, operating condition
REPRESENTATION OF RIGID MOTION
CHILD RESPONSE IN TRIM
WEIGHTS
convergence test, perturbation, mode normalization
ORDER REDUCTION
zero, dynamic, or quasistatic
for trim, transient, and flutter tasks
(superseded by mode or transform specification)
SYMMETRY (FOR FLUTTER TASK)
symmetric, antisymmetric, or both

RESPONSE TYPES: RIGID AND VARIABLE

CONVENTIONS FOR DESCRIPTION OF MOTION

4-22.1 Response Representation

FORM OF REPRESENTATION OF EACH SYSTEM VARIABLE:

total $=$ reference $+$ difference

CONSIDERING RIGID MOTION VARIABLES, ALSO INTRODUCE NOMINAL:

total $=$ nominal $+$ (reference $+$ difference)

sum of reference and difference always scalar

sum of nominal and degree of freedom may involve kinematics of rigid motion

RIGID RESPONSE REPRESENTATION REQUIRED FOR RIGID BODY MOTION OF COMPONENTS, AND FOR FRAMES

RIGID MOTION IS MEASURED RELATIVE TO FRAME, AND FRAMES ARE HIERARCHICAL

COMPLETE REPRESENTATION OF MOTION RELATIVE SOME PARENT FRAME (ULTIMATELY, INERTIAL SPACE):

motion $=$ frame $+$ frame $+$ . . . $+$ total

each frame and total has separate rigid response definition

NOMINAL, REFERENCE, AND DIFFERENCE

- SOLUTION PROCEDURES SOLVE FOR DIFFERENCE
  - FIND MOTION RELATIVE TO SOME PRESCRIBED NOMINAL AND REFERENCE
- CHOOSE NOMINAL AND REFERENCE TO HELP SOLUTION PROCESS
- USES OF NOMINAL AND REFERENCE:
  - to keep difference small, by accounting for rest position and any large rotation or displacement
    - analysis does not assume that difference is small, but many solution procedures work better if it is
  - to keep difference periodic, so solution procedure need be applied only over one revolution
  - to avoid singularities in representation of rotations
  - to introduce specified operating condition
- DISTINGUISH BETWEEN NOMINAL AND REFERENCE BECAUSE CAN BE DIFFERENT FORMS OF MOTION
  - reference required since it can be updated with latest solution for difference (because sum is scalar)
  - nominal can be needed to account for large rigid motion, or steady state velocity or rotation, or so the difference will be constant or periodic

USER (OR ROTORCRAFT SHELL) SELECTS NOMINAL AND REFERENCE KIND

available conventions for nominal and reference must be appropriate for problems to be handled by analysis

DIFFERENCE KIND AND OTHER PROPERTIES OF RESPONSE FOLLOW FROM CHOICE OF PART SOLUTION PROCEDURE

from trim part solution, difference can be

single value, periodic, not periodic, or multiperiod
time domain or harmonics
one period, or primary and secondary periods
with part filter active or not

from transient part solution, difference can be

time domain
structured for implicit or integration solution
possibly just trim response

UPDATED REFERENCE

PART SOLUTIONS MAY LINEARIZE SYSTEM EQUATIONS ABOUT NOMINAL AND REFERENCE

CONVERGENCE MAY BE IMPROVED IF REFERENCE IS UPDATED FROM CURRENT SOLUTION FOR DIFFERENCE, AND MATRICES RECALCULATED (MODES PROBABLY RECALCULATED ALSO)

UPDATE CONVENTIONS IMPLEMENTED:

for trim task:

trim reference updated as required by part solution procedure, by adding mean part of current difference value

reference of all response variables updated at end of trim task

for transient task:

updated trim reference is reference in transient task

equations can be linearized about current transient solution

for flutter task:

equations are usually linearized about total trim solution (nominal, reference, and difference)

4-22.2 Response Evaluation

TRIM RESPONSE MUST ACCOMMODATE SYSTEM WITH MORE THAN ONE PERIOD

TRIM PARTS (SUCH AS HARMONIC SOLUTION METHOD) CAN SOLVE EQUATIONS ASSUMING THAT SYSTEM HAS KNOWN PERIOD, SO EQUILIBRIUM SOLUTION IS KNOWN TO BE PERIODIC

IF DIFFERENT SUBSYSTEMS HAVE DIFFERENT PERIODS, THEN APPROXIMATIONS ARE REQUIRED TO ENFORCE ASSUMPTION OF PERIODICITY WITHIN PART SOLUTION

produces approximate solution, for reasonable effort

can use transient task to get exact solution

SO DIFFERENTIAL EQUATIONS OF TRIM PART IDENTIFIED AS TIME INVARIANT OR PERIODIC

PERIODIC PART (ROTOR)

CAN BE SOLVED ONLY FOR RESPONSE AT ITS OWN PERIOD

uses component input = mean value + vibratory motion at same period

TIME INVARIANT PART (AIRFRAME)

CAN BE SOLVED FOR RESPONSE AT SEVERAL PERIODS

separate solution for each period acting on part (primary period, and optionally one or more secondary periods)

uses component input = mean value (primary period only) + vibratory motion at each period

4-22.3 Child Response in Trim

SYSTEM CAN HAVE SYMMETRIES THAT IN CERTAIN OPERATING CONDITIONS IMPLY RELATION BETWEEN SOLUTIONS FOR VARIOUS SUBSYSTEMS

- CAN BE ENFORCED BY DEFINING RESPONSE OF ONE SUBSYSTEM AS CHILD OF ANOTHER SUBSYSTEM
  - ONLY NECESSARY TO SOLVE FOR PARENT RESPONSE, SO CAN SIGNIFICANTLY REDUCE COMPUTATION REQUIRED
  - child response used for blades of rotor
- SYMMETRIES CAN ALSO IMPLY RESTRICTIONS ON FORM OF PART SOLUTION
  - can be enforced at end of part solution, by order reduction and filter operations

SYMMETRIES

- AXISYMMETRIC SYSTEM (TYPICALLY N IDENTICAL, EQUALLY SPACED SUBSYSTEMS, PERHAPS ROTATING)
  - motion can be identical except for phase shift between child and parent
    - typically also have certain harmonics of solution equal zero (trim filter)
- SYMMETRY ABOUT PLANE (SUCH AS LATERAL SYMMETRY)
  - motion can be mirrored across plane, hence identical except for sign changes between child and parent
    - also antisymmetric motions should be zero (order reduction)

TRIM RESPONSE CAN BE IDENTIFIED AS CHILD OF SOME OTHER RESPONSE

NO TRIM SOLUTION FOR CHILD
equations in no-solution type part, or not in any loop

IDENTIFY PARENT
not hierarchical

SPECIFY PHASE SHIFT
parent can not have secondary periods

SPECIFY SIGN CHANGE

RESPONSE EVALUATION

CHILD RESPONSE AT TIME t EVALUATED FROM PARENT RESPONSE (DIFFERENCE, REFERENCE, AND NOMINAL) AT TIME t_e

with phase shift, $t_e = t + \Delta t$

$\Delta t = \Delta\psi/\Omega$ for periodic or multiperiod response

PERHAPS WITH SIGN CHANGES

for rigid response type: sign of angular motion about x, y, or z-axis is changed; and signs of linear motion about other two axes are changed

4-22.4 Response Specification

RESPONSE TYPES: RIGID AND VARIABLE

REFERENCE FOR RESPONSE CONVENTIONS: CAMRAD II Documentation, Volume I, Theory; Chapter “Response”

VARIABLE RESPONSE TYPE

CONVENTIONS FOR NOMINAL AND REFERENCE:

ROTATING VARIABLE (depends on a period)
REST POSITION
NONE

ROTATING VARIABLE

NOMINAL MOTION IS CONSTANT ANGULAR VELOCITY FOR SCALAR QUANTITY (NOT RIGID BODY ROTATION)

REST POSITION

NOMINAL MOTION IS CONSTANT DISPLACEMENT

NONE

NO NOMINAL OR REFERENCE; DIFFERENCE IS TOTAL RESPONSE

intended to provide efficient storage and access, for quantities that do not require nominal or reference

RIGID RESPONSE TYPE

CONVENTIONS FOR NOMINAL AND REFERENCE:

BASE FRAME (depends on an operating condition)
ROTATING FRAME (depends on a period)
REST POSITION

BASE FRAME NOMINAL/REFERENCE

FOR RESPONSE OF A “BASE FRAME” (PARENT FRAME IS INERTIAL)

defined in terms of operating condition (which can be changed by solution procedures), and can provide constant-velocity motion

motion of system body axes F, relative to inertial axes I

operating condition defines nominal and reference, which is mean motion of F relative I

difference contains remainder of motion, found by the part solution (with zero mean for trim task)

BODY OR STABILITY AXES, OR CONSTRAINED (CONSISTENT WITH OPERATING CONDITION)

OPTIONS:

free, body axes: no nominal, so degrees of freedom represent total motion of F relative I

body-axes velocity or inertial-axes displacement

and aircraft Euler angles

free, stability axes: degrees of freedom represent motion of velocity axes relative inertial axes (V relative I); so nominal is constant orientation of F relative V

body-axes velocity or inertial-axes displacement

and aircraft Euler angles

constrained: no nominal, so degrees of freedom represent total motion of F relative I

constant reference linear motion

and aircraft Euler angles, Rodrigues parameters, or arbitrary Euler angles

ROTATING FRAME

NOMINAL MOTION IS ROTATION BY ANGLE $\psi = \Omega t + \psi_0$ ABOUT DESIGNATED AXIS

REST POSITION

NOMINAL MOTION IS CONSTANT DISPLACEMENT AND ROTATION

REPRESENTATION OF RIGID MOTION

FOR RIGID RESPONSE TYPE, DEGREE OF FREEDOM IS RIGID MOTION:

linear motion $x^{BA/A}$ (B relative A, in A axes), and derivatives

angular motion C^{BA}, and derivatives

linear motion only, or both linear and angular motion

constrained component has no rigid motion;
point mass component needs only linear motion;
other components will have both linear and angular motion

NEED SCALAR, THREE-PARAMETER REPRESENTATION OF LINEAR MOTION q AND ANGULAR MOTION p

REPRESENTATION OF LINEAR MOTION

inertial axes displacement

$q = x^{BA/A} =$ displacement measured in A axes
inertial acceleration $= \dot{v} + \widetilde{\omega} v = C^{BA}\ddot{q}$

body axes displacement

$q = x^{BA/B} =$ displacement measured in B axes
inertial acceleration $= \dot{v} + \widetilde{\omega} v = \ddot{q} + 2\widetilde{\omega}\dot{q} + \dot{\widetilde{\omega}} q + \widetilde{\omega}\widetilde{\omega} q$

body axes velocity

$\dot{q} = v^{BA/B} =$ velocity measured in B axes
inertial acceleration $= \dot{v} + \widetilde{\omega} v = \ddot{q} + \widetilde{\omega}\dot{q}$

BODY AXES VELOCITY REPRESENTATION OF LINEAR MOTION

CAN ONLY BE USED WITH BASE FRAME NOMINAL KIND

used for airframe frame

difficulties arise because displacement x not directly obtained from q, but rather by integral of $\dot{q}$ (which will be path dependent)

$\dot{q} = v^{BA/B} = C^{BA}\dot{x}^{BA/A}$

so $x^{BA/A} = \int^t C^{AB}\dot{q}\,dt$

COMPLETE EVALUATION OF DISPLACEMENT IS NOT IMPLEMENTED

do not integrate total $\dot{q}$ to obtain x

for trim or transient, contribution of reference calculated, but difference neglected

for flutter, approximate perturbation for small trim velocity

SO BODY AXES VELOCITY REPRESENTATION OF LINEAR MOTION CAN NOT BE USED IF DISPLACEMENT RELATIVE INERTIAL FRAME IS IMPORTANT

use inertial axes displacement instead

correct, but unconventional representation for aircraft rigid body motion

REPRESENTATION OF ANGULAR MOTION

aircraft Euler angles (Tait-Bryan)

$p = (\phi\,\theta\,\psi)^T =$ rotation angles about x, y, and z axes

rotation $= C^{BA} = X_\phi Y_\theta Z_\psi$

Rodrigues parameters (Euler-Rodrigues)

$p = 2u\tan(\psi/2) =$ rotation by angle ψ about axis u

rotation $= C^{BA} = I - R\tilde{p}$

arbitrary Euler angles

$p = (\phi\,\theta\,\psi)^T =$ rotation angles about u, v, and w axes

u, v, and w = positive or negative x, y, or z axis

rotation $= C^{BA} = U_\phi V_\theta W_\psi$

4-22.5 Convergence Test

ITERATIVE SOLUTION METHOD FOR PART OR LOOP MAY REQUIRE THAT CONVERGENCE BE TESTED

BY COMPARING CURRENT VALUE x OF VARIABLE WITH OLD VALUE x_{old} FROM PREVIOUS ITERATION

GENERAL FORM OF CONVERGENCE TEST:

$$\mathrm{error} = \|x - x_{\mathrm{old}}\| \le \mathrm{tolerance} \times \mathrm{weight}$$

tolerance is single number, from part or loop data

weighting factor (“weight”) is defined for each element of vector, as part of response data

error is norm of difference between current and old solution

for single value: $\mathrm{error} = |x - x_{\mathrm{old}}|$

for other difference kind: $\mathrm{error} = \mathrm{rms}(x - x_{\mathrm{old}})$

RELATIVE VALUES OF CONVERGENCE WEIGHTS AFFECT WHAT VARIABLE SEEMS TO DRIVE CONVERGENCE OF LOOP OR PART

hence affect interpretation of tolerance

value of tolerance required for accurate solution is based on physical results

4-22.6 Perturbation

PART OR LOOP SOLUTION METHOD MAY REQUIRE THAT VARIABLE x BE PERTURBED

TYPICALLY IN ORDER TO LINEARIZE SOME SUBSET OF SYSTEM

GENERAL FORM OF PERTURBATION:

$$\delta x = \Delta \times \text{weight}$$

perturbation amplitude Δ is from part or loop data

weighting factor ("weight") is defined for each element of vector, as part of response data

separate values of Δ are used for displacement, velocity, and acceleration perturbations

VARIABLE PERTURBATION AFFECTS ACCURACY OF MATRICES CALCULATED BY FINITE DIFFERENCE APPROXIMATION

relative values of perturbation weights must be chosen so all perturbations have similar behavior (all too small or all too large) with a common perturbation amplitude

4–23 System Pieces: Weights

PROVIDES STANDARD PERTURBATION AND CONVERGENCE WEIGHTS FOR RESPONSE

STANDARD WEIGHTS CALCULATED FROM REFERENCE QUANTITIES:

length L, angular velocity Ω, linear velocity V, force F

ITERATIVE PART OR LOOP SOLUTION METHODS:

MAY REQUIRE THAT CONVERGENCE BE TESTED
MAY REQUIRE THAT VARIABLE BE PERTURBED

analysis automatically identifies which variables a loop or part must perturb and test

FOR EACH RESPONSE SYSTEM PIECE, NEED TO SPECIFY WEIGHTS THAT CAN BE USED TO PERTURB THE VARIABLE AND TO TEST ITS CONVERGENCE

OTHER SYSTEM PIECES CAN ALSO REQUIRE WEIGHTS

WEIGHT USED IS PRODUCT OF STANDARD WEIGHT AND INPUT FACTOR

typically factor is 1., but available to handle special cases

IF WEIGHTS SYSTEM PIECE IS NOT IDENTIFIED FOR A VARIABLE, THEN INPUT VALUE OF WEIGHT IS USED

VARIABLE KIND MUST BE IDENTIFIED IN ORDER TO OBTAIN APPROPRIATE STANDARD WEIGHT

variable kind	standard weight
basic	
unit	1
angular displacement (deg)	1
angular displacement (rad)	$1/57.3$
angular velocity	$\Omega/100$
angular acceleration	$\Omega^2/100$
linear displacement	$L/100$
linear velocity	$V/100$
linear acceleration	$\Omega V/100$
force	$F/100$
moment	$LF/1000$
power	$VF/1000$
circulation	$LV/10000$
dynamic pressure	$(50F/L^2)/100$
aerodynamic interfaces of structural dynamic components	
velocity (10 elements)	v, q, dv/dt, ω
position (6 elements)	r, dr/dt
force (6 elements)	force, moment
aerodynamic	
circulation peaks	magnitude and position

4–24 Transient Task

NUMERICALLY INTEGRATE SYSTEM EQUATIONS, FROM TRIM SOLUTION, FOR PRESCRIBED EXCITATION

TRANSIENT SOLUTION BEGINS AT TIME t_B, AND ENDS AT TIME t_E

- EACH PART SOLVES ITS EQUATIONS OVER t_B to t_E, USING RESPONSE TIME STEP Δt
  - solution at this time step saved, and thus available to other parts
  - part solution procedure may use smaller time increment δt, because of accuracy or convergence considerations
    - save response at step size that reflects content of solution
- RESPONSE HAS TRIM VALUE AT t_B
  - and at earlier times, if required

RESTART CAPABILITY FOR TRANSIENT TASK

TRANSIENT SOLUTION CAN BE INTERRUPTED

at time t_X in numerical integration part

or at specified iteration and level in successive substitution loop

AT THE INTERRUPTION:

complete solution saved to restart file;
transient output produced, using solution obtained so far;
transient task is terminated

RESTARTED JOB:

jump to interrupt point in specified part or loop;
restore solution by reading restart file;
transient task continues

results same as if transient task completed in single job

possible to interrupt again a restarted transient task

PURPOSE OF RESTART CAPABILITY IS TO ALLOW PROGRESS OF TRANSIENT TASK TO BE ASSESSED BY EXAMINATION AT INTERMEDIATE POINT IN SOLUTION

none of the analysis input can be changed for restarted job (except specification of next interrupt)

transient analysis is completely specified by initial job

4-24.1 Rotorcraft Transient Task

TRANSIENT SOLUTION PROCEDURE CONSTRUCTED BY ROTORCRAFT SHELL WITH NO LOOPS

solve transient problem all in one part

no solution loop `ROTORCRAFT`
integration part `ROTORCRAFT`

4–25 System Pieces: Transient Loop

METHODS IMPLEMENTED FOR TRANSIENT LOOPS:

- NO SOLUTION
- SUCCESSIVE SUBSTITUTION (with stages)

LOOP DESCRIBED BY IDENTIFYING PARTS, CHILD LOOP, AND WRITE MODULES

- PARTS SOLVED IN LOOP (ITERATED WITH CHILD LOOPS), OR AT END OF LOOP
  - parts are solved in order defined
- LOOPS ARE HIERARCHICAL
  - loop can have at most one child and one parent
  - transient task can have more than one set of nested loops
- WRITE MODULES PRODUCE OUTPUT, FOR:
  - SHELL
  - CONVERGENCE
  - LOOP SOLUTION
  - PART SOLUTION
  - MODES
  - OUTPUT OF PART
  - OUTPUT
  - GRAPHICS

LOOP ALGORITHM, THEORY, PARAMETERS SIMILAR TO TRIM LOOPS

4–26 System Pieces: Transient Part

METHODS IMPLEMENTED FOR TRANSIENT PARTS:

NO SOLUTION
TRIM SOLUTION
IMPLICIT
- equations solved in passes

INTEGRATION
- solve differential equations
- numerical integration algorithms

PART DESCRIBED BY IDENTIFYING ITS MOTION, CONSTRAINT, AND OUTPUT EQUATIONS

EACH EQUATION CAN BE IN ONLY ONE PART (POSSIBLY NO-SOLUTION METHOD)

EACH PART CAN BE SOLVED IN ONE OR MORE LOOPS

EACH PART SOLVES ITS EQUATIONS OVER TRANSIENT TIME RANGE t_B to t_E, USING RESPONSE TIME STEP Δt

solution at this time step saved, and thus available to other parts

different parts can use different time steps

PART THAT PERFORMS NUMERICAL INTEGRATION USES TIME INCREMENT δt

ACCURACY OR CONVERGENCE CONSIDERATIONS DETERMINE SIZE OF δt

WHILE CONTENT OF SOLUTION DETERMINES SIZE OF Δt

- step Δt must be multiple of δt

NO SOLUTION METHOD

- EQUATIONS ASSIGNED TO PART ARE NOT SOLVED
  - corresponding variables will be zero
- SO SUPPRESSES EFFECT OF THESE SYSTEM VARIABLES DURING TRANSIENT TASK

TRIM SOLUTION METHOD

- EQUATIONS ASSIGNED TO PART ARE NOT SOLVED
  - corresponding variables will have value from trim solution

4-26.1 Implicit Solution Method

IMPLICIT METHOD DIRECTLY SOLVES CONSTRAINT AND OUTPUT EQUATIONS

PROCESS AND THEORY SIMILAR TO TRIM PART

TIME VALUES

response time steps

solved at K times, from $t_B + \Delta t$ to t_E

4-26.2 Differential Equations

INTEGRATION SOLUTION METHOD SOLVES DIFFERENTIAL EQUATIONS

- PROCESS AND THEORY SIMILAR TO TRIM PART

TRANSFORM AND REDUCTION

- CAN USE MODAL TRANSFORM FOR STRUCTURAL DYNAMIC DEGREES OF FREEDOM OF PART
- DEGREES OF FREEDOM OR MODES CAN BE TRANSFORMED
  - multiblade coordinates
  - rotating-to-nonrotating
- SOLUTION FOR SELECTED VARIABLES CAN BE QUASISTATIC, WITH OR WITHOUT A RESIDUAL
- POSSIBLE TO OMIT SELECTED VARIABLES FROM SOLUTION

MATRICES OF DIFFERENTIAL EQUATIONS

MOTION AND CONSTRAINT EQUATIONS LINEARIZED ABOUT CURRENT TRANSIENT SOLUTION

EQUATIONS CAN BE TIME INVARIANT OR TIME-VARYING

optionally, matrices can be averaged over period

OPTIONALLY REFERENCE CAN BE UPDATED (FROM CURRENT SOLUTION FOR DIFFERENCE) AND MATRICES RECALCULATED DURING SOLUTION PROCESS

matrices that depend on time-varying transform (multiblade coordinates or rotating-to-nonrotating) are always updated at each integration step

IF EQUATIONS ARE NONLINEAR, MIGHT RECALCULATE MATRICES WHENEVER REFERENCE CHANGES

IF EQUATIONS ARE TIME-VARYING, MIGHT RECALCULATE MATRICES EACH TIME STEP

GENERALLY MORE EFFICIENT TO EXECUTE UPDATE LESS OFTEN

4-26.3 Integration Solution Method

INTEGRATION METHOD SOLVES DIFFERENTIAL EQUATIONS FOR TRANSIENT RESPONSE, BY NUMERICAL INTEGRATION

INTEGRATION ALGORITHMS:

linear Newmark	nonlinear Newmark
linear Wilson	nonlinear Wilson
linear HHT	nonlinear HHT

"linear" and "nonlinear" refer to form of algorithm, each can be applied to linear or nonlinear problem

LINEAR NEWMARK ALGORITHM IS GOOD CHOICE

figure 22 outlines the method

AT EACH TIME STEP AN ITERATION IS REQUIRED FOR NONLINEAR PROBLEMS

THEORY

REDUCED EQUATIONS OF MOTION

differential equations for θ_k, and static equations for θ_l:

$$h_{kk}\theta_k = \widetilde{A}_k$$
$$K_{ll}\widetilde{\theta}_l = A_{l\mathrm{QS}}$$

equations for θ_l directly inverted at each time step

equations to be integrated take form $Hu = R$

$H = H(d/dt) =$ differential operator (spring, damper, and mass)

in general, R depends on time and on u and its derivatives, and problem is nonlinear

initialize: $F_{\text{old}} = 0$, $\theta_{\text{old}} = \text{trim}$
time step k (response Δt)
initialize response: $x_k = x_{k-1}$
time increment i (integration δt)
    calculate matrices
    initialize degrees of freedom θ
    iteration
        save response: $x_{\text{old}} = x$
        evaluate equations: F
        relax equations: $F = \lambda F + (1-\lambda)F_{\text{old}}$
        save equations: $F_{\text{old}} = F$
        calculate forces of reduced equations
        integrate equations for degrees of freedom θ
        calculate response: x
        test convergence: $\text{error} = \|x - x_{\text{old}}\| \leq \text{tolerance} \times \text{weight}$
    save degrees of freedom θ
output
    for each time value (response Δt)
        evaluate equations: $y = B_y$
implement order reduction: ξ, f_0, f_1, y, f_l

Figure 4-22 Outline of integration solution method.

FINITE-DIFFERENCES

introducing finite-difference approximation for time derivatives converts differential equations at time t^i into algebraic equations

$$\widetilde{K}^i u^i = \widetilde{R}^i$$

solved for displacement u^i, and then derivatives $\dot{u}^i$ and $\ddot{u}^i$ evaluated using finite-difference expressions

for nonlinear problem, solve $\widehat{R} = R - Hu = 0$

finite-difference approximation for derivatives converts to algebraic equations:

$$0 = \widehat{R}(u^i) = (R - Hu)^i = \widetilde{R}^i - \widetilde{K}^i u^i$$

solved for displacement u^i at time t^i

Newton-Raphson solution (with relaxation) is

$$\widehat{K}^i_n \Delta u^i = \lambda \widehat{R}^i_n$$

where $\widehat{K}$ is derivative matrix (partial derivative of $-\widehat{R}$ with respect to u)

with approximation $\widehat{K} \cong \widetilde{K}$, equation for nonlinear case becomes:

$$\widetilde{K}^i_n \Delta u^i = \lambda \widehat{R}^i_n$$

solved for change Δu from iteration n to iteration $n+1$, all at time t^i

same algorithm obtained considering relaxed, successive substitution solution of $\widetilde{K}u = \widetilde{R}$

for generality, relaxation applied to force R rather than u

ITERATION

- algorithm has time loop, over transient time range
- at each time increment, there is an iteration
  - at beginning of iteration, solution for u^i initialized using u^{i-1}
  - evaluate force R^i
  - with relaxation factor for convergence of nonlinear problems

    $$R^i = \lambda R^i + (1 - \lambda) R^i_{\rm old}$$

    old force is from previous iteration (initialized to zero at beginning of part solution)
  - reduced equations integrated, solving for degrees of freedom u^i and derivatives
- at end of each iteration, convergence is tested
- when iteration converged, u and R saved for use during next time increment

RELAXATION CAN ALSO BE APPLIED TO SELECTED IMPLICIT CONSTRAINT EQUATIONS

- used to obtain convergence of circulation with vortex wake model

CONVERGENCE TEST

correct solution for u^i is not known, so convergence tested by comparing two successive iterations:

$$\text{error} = \|u - u_{\rm old}\| \leq \text{tolerance}$$

typically error is absolute value of difference between iterations

INITIALIZATION OF ITERATION AT t^i

at start of iteration, solution u^i must be initialized, using u^{i-1}

options:

- zero
- constant
- constant displacement
- constant acceleration
- zero acceleration

choice will affect only convergence of iteration

in practice, this initialization option is not usually important

INTEGRATION ALGORITHMS

ALGORITHM MUST SOLVE FOR u AND ITS DERIVATIVES FROM H (M, C, and K) AND R, ALL AT TIME t^i

EXPLICIT METHODS NOT CONSIDERED (ONLY CONDITIONALLY STABLE)

LINEAR NEWMARK

one-step, one-leg method formulated for linear problems

Newmark constant average acceleration method (trapezoidal rule)

unconditionally stable, and has no attenuation of high frequency modes

LINEAR WILSON

one-step, two-leg method formulated for linear problems

linear acceleration method (Wilson-θ), with $\theta = 1.4$

unconditionally stable for $\theta > 1.37$, and has attenuation of high frequency modes

LINEAR HILBER-HUGHES-TAYLOR

one-step, two-leg method formulated for linear problems

HHT α-method (extension of Newmark method); numerical dissipation to damp out spurious participation of high frequency modes

unconditionally stable, and has attenuation of high frequency modes

typically $\alpha = -0.05$ or -0.10

NONLINEAR NEWMARK, WILSON, OR HILBER-HUGHES-TAYLOR

methods formulated for nonlinear problems

PARAMETERS CONTROLLING CONVERGENCE AND ACCURACY OF METHOD:

MAXIMUM NUMBER OF ITERATION
TOLERANCE
RELAXATION FACTORS
DAMPING

NUMERICAL INTEGRATION METHOD
HHT NUMERICAL DISSIPATION
INITIALIZATION METHOD

SIZE OF INTEGRATION TIME STEP

MATRIX CALCULATION
- perturbation order
- magnitude and sign of perturbation
- averaging

MATRIX UPDATE STRATEGY

DISTURBANCES AT BEGINNING OF TRANSIENT, t_B

derivatives of u are evaluated using solution at previous time step (finite-differences)

at beginning of transient ($t_B + \delta t$), need $u = \theta_k$ at t_B

obtained from trim solution

trim and transient solution can use different degrees of freedom, different modes, different transforms

such solution and model differences can produce initial disturbance in transient solution

4–27 Flutter Task

DIFFERENTIAL EQUATIONS LINEARIZED ABOUT TRIM

- LINEARIZE SYSTEM EQUATIONS, ABOUT TRIM SOLUTION
  - linearization implemented using loops and parts
- ANALYZE RESULTING DIFFERENTIAL EQUATIONS

FLUTTER EQUATIONS CAN BE TIME INVARIANT OR PERIODIC

- TIME INVARIANT EQUATIONS OBTAINED FOR TRULY TIME INVARIANT SYSTEM
- OR BY AVERAGING EQUATIONS OF PERIODIC SYSTEM
  - first (outermost) loop can average system equations over period, producing constant-coefficient approximation
    - child loops must average equations for subsystem that has period not equal system period
  - averaged equations are approximation
    - if system is strongly periodic, constant-coefficient approximation may not well represent the true behavior
    - averaging can also suppress interactions between subsystems coupled by loops

- PERIODIC EQUATIONS ARE ANALYZED USING FLOQUET-LYAPONOV THEORY
  - time invariant equations can be analyzed as if they are periodic
    - way to check accuracy of integration required for periodic equations

EXAMPLES FOR ROTORCRAFT

- periodic coefficients from aerodynamics of edgewise flow
  - with multiblade coordinates and three or more blades, averaged coefficients good enough up to advance ratio of about $\mu = 0.5$
- coupled dynamics of two-bladed rotor and shaft motion
  - strong 1/rev periodicity, so Floquet theory is required

EQUATIONS CAN BE WRITTEN TO OUTPUT FILE OR PLOT FILE, FOR USE BY EXTERNAL ANALYSIS

4-27.1 Analysis of Flutter Equations

FLUTTER LOOPS AND PARTS PRODUCE LINEAR DIFFERENTIAL EQUATIONS DESCRIBING SYSTEM RESPONSE

$$A_2\ddot{x} + A_1\dot{x} + A_0x = B_0v$$
$$y = C_2\ddot{x} + C_1\dot{x} + C_0x + D_0v$$

x = vector of degrees of freedom
v = vector of controls
y = vector of sensors

subset of x, $\dot{x}$, $\ddot{x}$, and v can be designated as output quantities

coefficient matrices can be constant, or periodic in time

figure 23 outlines analysis process

ANALYSIS TASKS

FOR TIME INVARIANT EQUATIONS (TRUE OR AVERAGED)

eigenanalysis
time history response
frequency response
rms gust response

FOR PERIODIC EQUATIONS

eigenanalysis
time history response

- time-invariant equations
  - evaluate equations (loops and parts)
  - analyze equations
    - eigenanalysis
    - time history response
    - frequency response
    - rms gust response
- periodic equations
  - time loop (one period)
    - evaluate equations (loops and parts)
    - store equations
  - eigenanalysis
    - time loop (one period)
      - interpolate equations
      - analyze equations
  - time history response
    - time loop
      - interpolate equations
      - analyze equations

Figure 4-23 Outline of flutter task.

REDUCTION OF EQUATIONS

DEGREES OF FREEDOM, OUTPUT VARIABLES, OR INPUT VARIABLES CAN BE ELIMINATED FROM EQUATIONS

options:

- complete equations (eliminate only designated variables)
- symmetric equations
- antisymmetric equations

QUASISTATIC REDUCTION OF DEGREES OF FREEDOM

options:

- full dynamics (no quasistatic reduction)
- eliminate designated degrees of freedom
- eliminate all but system rigid degrees of freedom

neglect mass and damping of quasistatic degrees of freedom x_l; then introduce transformation that decouples spring matrix

SO NINE SUBSETS OF FLUTTER EQUATIONS CAN BE ANALYZED

STATE VARIABLE FORM

EQUATIONS TRANSFORMED TO STANDARD FIRST ORDER FORM

degree of freedom that has both mass and spring terms is second order, so states are displacement and velocity

remaining degrees of freedom are first order, with single state each (displacement or velocity)

STATE VECTOR

velocity of all degrees of freedom, and displacement of second order degrees of freedom

$$x = \begin{pmatrix} \dot{x}_2 \\ \dot{x}_1 \\ x_2 \end{pmatrix}$$

x_1 = first order degrees of freedom
x_2 = second order degrees of freedom

although for first order degrees of freedom with zero mass, state is actually displacement

EQUATIONS OF MOTION

$$\dot{x} = Ax + Bv$$
$$y = Cx + Dv$$

TIME INVARIANT EQUATIONS

EIGENANALYSIS

eigenvalues (poles)
- frequency, damping, time constant

state eigenvectors

output eigenvectors

zeros for each input/output pair
- residues and static response

TIME HISTORY RESPONSE

for each input/output pair (unit amplitude of input)

input time history options:
- impulse, step, ramp
- cosine impulse, sine doublet
- square impulse, square doublet
- triangular impulse, triangular doublet
- general piecewise linear

time history calculation:

- from modes (eigenvalues and eigenvectors): independent of time step

- from numerical integration: time step affects accuracy

  - modified trapezoidal method
  - 4-th order Runge-Kutta method

FREQUENCY RESPONSE

for each input/output pair (unit amplitude of input)

transfer function calculation:

from matrices: probably most accurate

from modes (eigenvalues and eigenvectors): probably fastest

from poles and zeros: probably least accurate

frequencies:

static response from matrices (zero frequency)

calculated range (linear or log scale)

input list

RMS GUST RESPONSE

input = vector of gust components

rms response not properly defined for system with unstable modes

gust spectra: Dryden and von Kármán

rms gust magnitude
gust correlation length or time constant

rms response calculation:

by stochastic method (from modes), with Markov process gust model (Dryden spectrum only)

by integrating transfer function (matrices, modes, or poles and zeros)

PERIODIC EQUATIONS

EVALUATE EQUATIONS

evaluated over one period, stored for analysis

time step based on harmonic content of coefficients

interpolated to times required by numerical integration (for state transition matrix or time history)

time step based on dynamics of system

interpolation methods:

linear
Fourier (default)
Fourier-linear

EIGENANALYSIS

eigenvalues
z-plane (state transition matrix)
s-plane

eigenvectors of state transition matrix

state transition matrix ϕ calculated by integrating $\dot{\phi} = A\phi$ over one period

numerical integration: time step affects accuracy

modified trapezoidal method
4-th order Runge-Kutta method

TIME HISTORY RESPONSE

for each input/output pair (unit amplitude of input)

input time history options:

- impulse, step, ramp
- cosine impulse, sine doublet
- square impulse, square doublet
- triangular impulse, triangular doublet
- general piecewise linear

time history calculation:

- from numerical integration: time step affects accuracy
    - modified trapezoidal method
    - 4-th order Runge-Kutta method

4-27.2 Rotorcraft Flutter Task

PARTITIONED PROCEDURE CONSTRUCTED BY ROTORCRAFT SHELL TO GENERATE EQUATIONS

FOR ENTIRE ROTORCRAFT
OR JUST FOR INDEPENDENT BLADE

SEPARATE LOOP FOR EACH SUBSYSTEM:

```
ROTORCRAFT
    ROTOR n
        ROTOR n INFLOW
        ROTOR n HUB
        ROTOR n BLADE m
    AIRFRAME
    DRIVE TRAIN
```

partitioning system makes it possible to examine equations of each subsystem

if rotors turn at different speeds, then always average equations for second (and other) rotor, airframe, and drive train

“DIFFERENTIAL EQUATIONS” PART METHOD FOR FOLLOWING:

```
ROTOR n INFLOW
ROTOR n HUB
ROTOR n BLADE m
AIRFRAME
DRIVE TRAIN
```

“interface” parts used for most input/output constraint equations, to minimize size of equations handled at each step

4–28 System Pieces: Flutter Loop

FLUTTER LOOP OPERATIONS:

COMBINE EQUATIONS OF PARTS AND CHILD LOOPS

TRANSFORM DEGREES OF FREEDOM, OUTPUT, AND INPUT
multiblade coordinate
rotating-to-nonrotating
symmetric/antisymmetric

REDUCED EQUATIONS
eliminate degrees of freedom, output, input

AVERAGE EQUATIONS OVER PERIOD

QUASISTATIC REDUCTION OF EQUATIONS
degrees of freedom; with or without residual

any variables identified for reduction (zero or quasistatic) that remain in final loop equations will be handled by analysis of flutter equations

4–29 System Pieces: Flutter Part

METHODS IMPLEMENTED FOR FLUTTER PARTS:

- NO SOLUTION
- INTERFACE
  - produce linearized equations
- DIFFERENTIAL EQUATIONS
  - produce linearized equations

PART DESCRIBED BY IDENTIFYING ITS MOTION, CONSTRAINT, AND OUTPUT EQUATIONS

- EACH EQUATION CAN BE IN ONLY ONE PART (POSSIBLY NO-SOLUTION METHOD)
- EACH PART CAN BE SOLVED IN ONLY ONE LOOP (OR NO LOOP)

MOTION, CONSTRAINT, AND OUTPUT EQUATIONS LINEARIZED ABOUT TRIM RESPONSE

- assume trim solution produces equilibrium solution, which therefore cancels from flutter equations

CONSTRAINT EQUATIONS ARE SOLVED TO ELIMINATE SYSTEM VARIABLES

- when system is divided into parts, necessary to make additional assumptions about the form of equations
  - so constraint equations of parts can be solved independently, and part equations then coupled by loops

NO SOLUTION METHOD

EQUATIONS ASSIGNED TO PART ARE NOT LINEARIZED

corresponding variables will then have zero perturbation

SO SUPPRESSES EFFECT OF THESE SYSTEM VARIABLES DURING FLUTTER TASK

PARAMETERS CONTROLLING ACCURACY OF LINEARIZATION:

MATRIX CALCULATION

perturbation order

magnitude and sign of perturbation

Chapter 5

COMPONENTS THEORY

GENERAL REFERENCE FOR CHAPTER: CAMRAD II Documentation, Volume II, Components Theory

5–1 Components

FOR CONFIGURATION GENERALITY, SPLIT SYSTEM INTO PIECES, WITH CONNECTIONS BETWEEN

> COMPONENTS ARE PIECES THAT PERFORM ALL COMPUTATIONS ASSOCIATED WITH THE PHYSICS OF THE MODEL OF A SYSTEM

COMPONENTS AVAILABLE TO CONSTRUCT SYSTEM:

rigid body	Fourier series	wing wake geometry
linear normal modes	prescribed control	rotor wake geometry
finite element beam	gust	
rod/cable		wing performance
	rigid airframe aerodynamics	rotor performance
transmission	airframe flow field	rotorcraft performance
reference frame		
	lifting line wing	helicopter tail boom
filter	rigid wing	computational fluid dynamics
reference plane		
	wing inflow	
differential equation	rotor inflow	
programmable	rotor dynamic wake	
transfer function	wing wake	plugin

> figure 1 shows use of components to construct rotorcraft

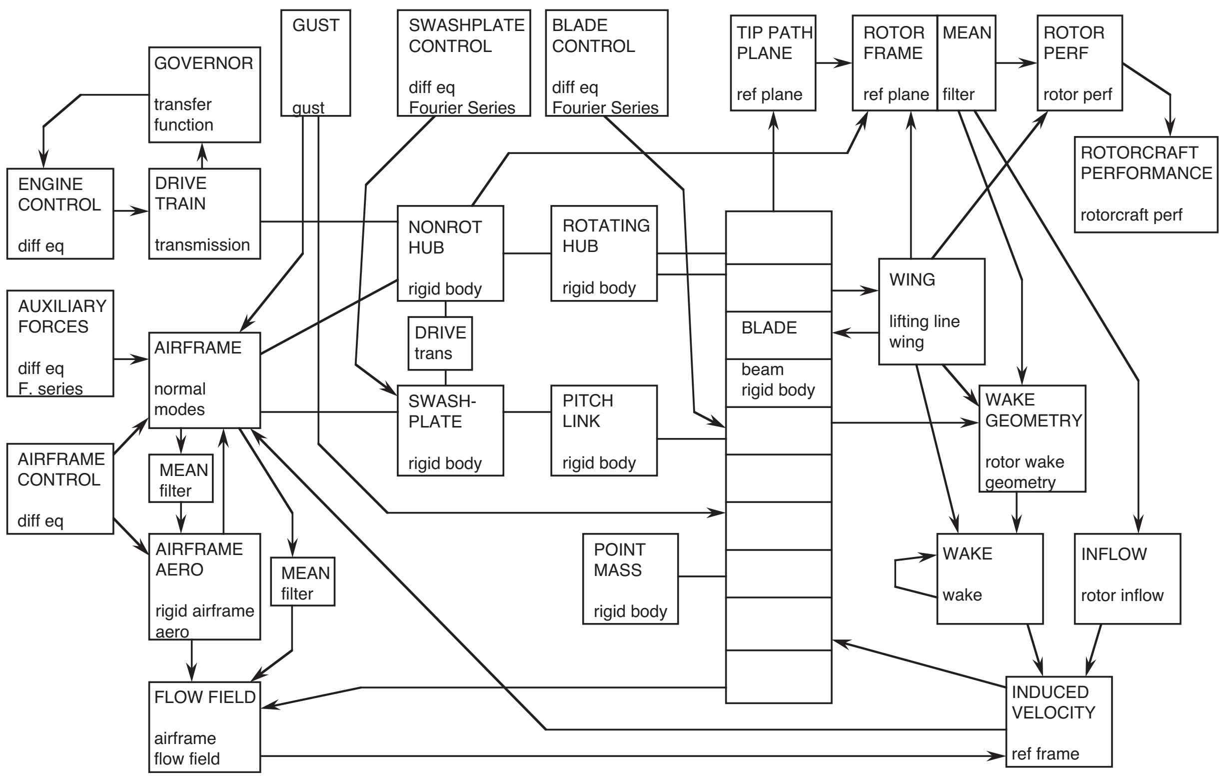


Figure 5-1 Rotorcraft model (simplified).

COMPONENT CAN HAVE DEGREES OF FREEDOM ξ_i,

DEPENDENCE ON DEGREES OF FREEDOM MEANS THAT EQUATIONS OF MOTION EXIST, AND USUALLY A DIFFERENTIAL EQUATION FORMULATION IS POSSIBLE

COMPONENT CAN DEPEND ON FRAME MOTION β

COMPONENT HAS INPUT f_i AND OUTPUT x_j

COMPONENT INPUT IS OF STRUCTURAL DYNAMIC KIND OR INPUT/OUTPUT KIND

can be connected to an interface or to a system input piece

COMPONENT OUTPUT IS OF STRUCTURAL DYNAMIC KIND OR INPUT/OUTPUT KIND

can be connected to an interface or to a system output piece

FOR STRUCTURAL DYNAMIC INTERFACE, INPUT AND OUTPUT OCCUR IN PAIRS

input is vector of force and moment at the connection

output is motion of axes at the connection

HENCE THE COMPONENT PRODUCES MOTION AND OUTPUT EQUATIONS, DEPENDING ON DEGREES OF FREEDOM, FRAME MOTION, AND INPUT:

$$\begin{aligned} 0 &= A_j(\xi_i, \beta, f_i) \\ x_j &= B_j(\xi_i, \beta, f_i) \end{aligned}$$

in general, nonlinear and time varying

if component does not have degrees of freedom, motion equations do not exist

A COMPONENT CAN BE CONSIDERED AN OPERATOR THAT EVALUATES A VECTOR OR MATRIX

figure 2 illustrates functionality

at time t component evaluates one of following vectors:

motion equation: A_j

component output for input/output interface: $x_j = B_j$

component output for structural dynamic interface: axes motion x_j

from degrees of freedom ξ_i (including frame motion) and from component input f_i

COMPONENT CAN ALSO PERTURB THIS VECTOR (ANALYTICALLY OR NUMERICALLY), TO CONSTRUCT MATRIX COLUMN

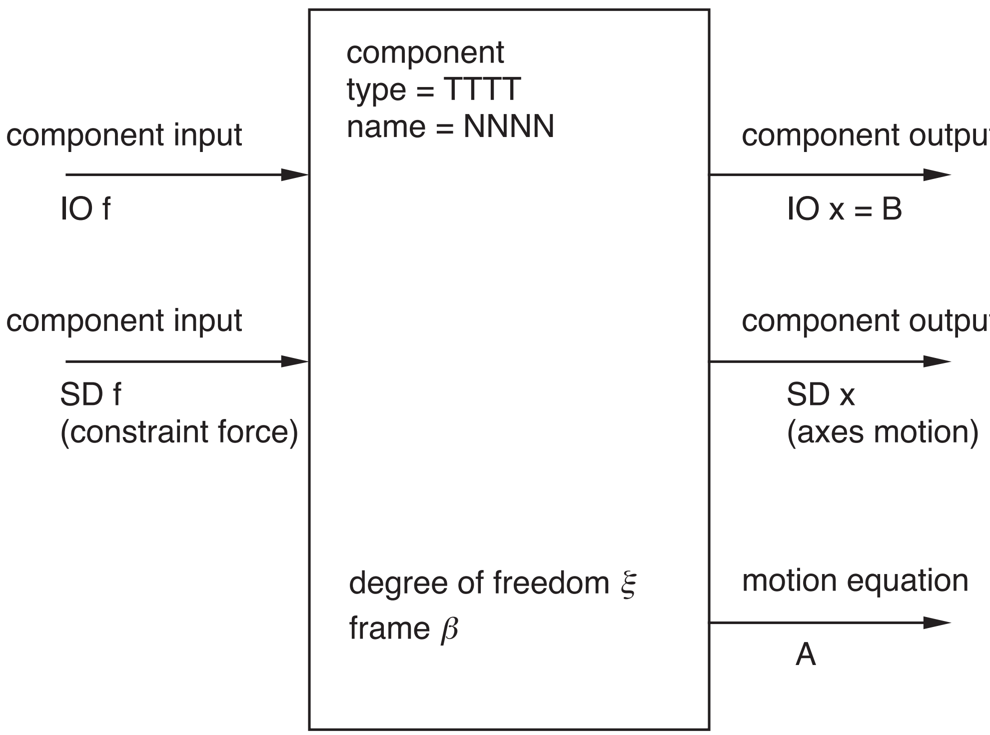


Figure 5-2 Component functionality.

5–2 Structural Dynamic Components

ALL STRUCTURAL DYNAMIC COMPONENTS SHARE COMMON CHARACTERISTICS, INCLUDING:

- rigid body motion and frame
- mass, hence inertial and gravitational forces
- joints
- structural dynamic interfaces
- aerodynamic interfaces
- applied load interfaces
- controls and sensors

  figure 3 illustrates standard functionality of structural dynamic components

COMPONENT MOTION DESCRIBED BY

- RIGID BODY MOTION, WHICH CAN ALWAYS BE LARGE
- PLUS ELASTIC MOTION, WHICH IS MEASURED RELATIVE RIGID MOTION

  STRUCTURAL DYNAMIC COMPONENTS DIFFER IN MODELLING OF ELASTIC MOTION

STANDARD RELATIONS USE EXACT KINEMATICS

- each component has large rigid body motion
- exact kinematics for all structural dynamic interfaces

STRUCTURAL DYNAMIC COMPONENT TYPICALLY MAKES APPROXIMATIONS IN REPRESENTATION OF ELASTIC MOTION

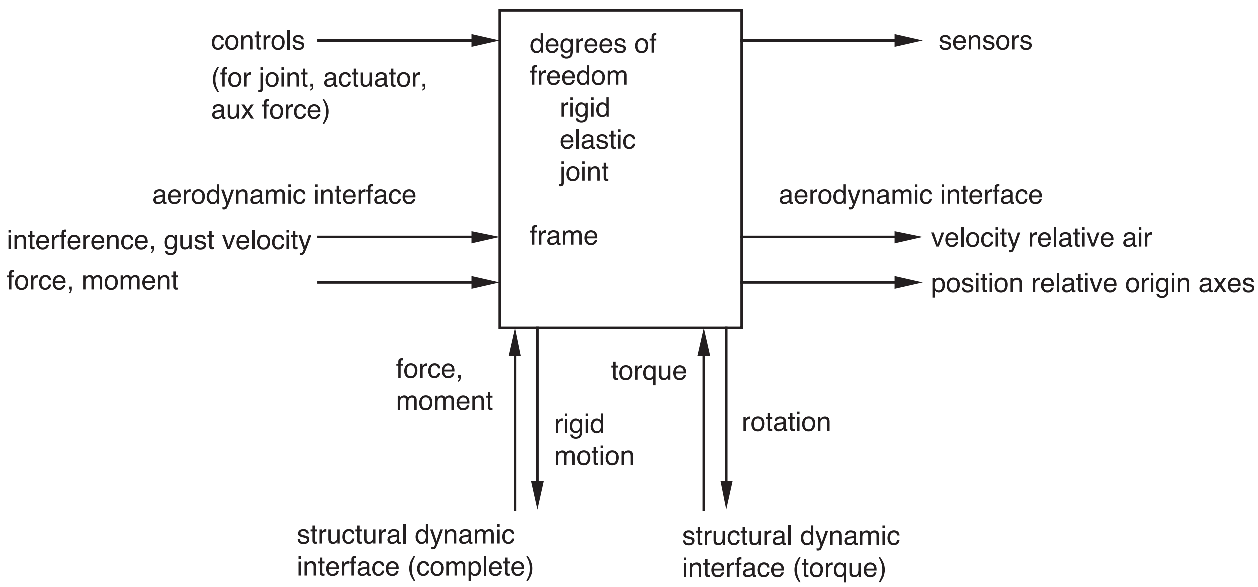


Figure 5-3 Functionality of structural dynamic components.

5-2.1 Kinematics

FRAME F (PERHAPS INERTIAL) IDENTIFIED FOR STRUCTURAL DYNAMIC COMPONENT

ALL MOTION OF COMPONENT MEASURED RELATIVE THAT FRAME

COMPONENT HAS RIGID BODY MOTION, DESCRIBED BY MOTION OF BODY AXES B

“CONSTRAINED COMPONENT”

this motion connected to the frame; rigid body motion is not degrees of freedom

“FRAME COMPONENT”

rigid body degrees of freedom are the frame motion

OTHER CASES

rigid body degrees of freedom exist, and represent motion relative to the frame

figure 4 summarizes options for rigid body motion

RIGID BODY MOTION CAN BE JUST LINEAR DEGREES OF FREEDOM

ignoring angular motion appropriate for point mass (no moments of inertia or center-of-gravity offset) with only pinned interfaces

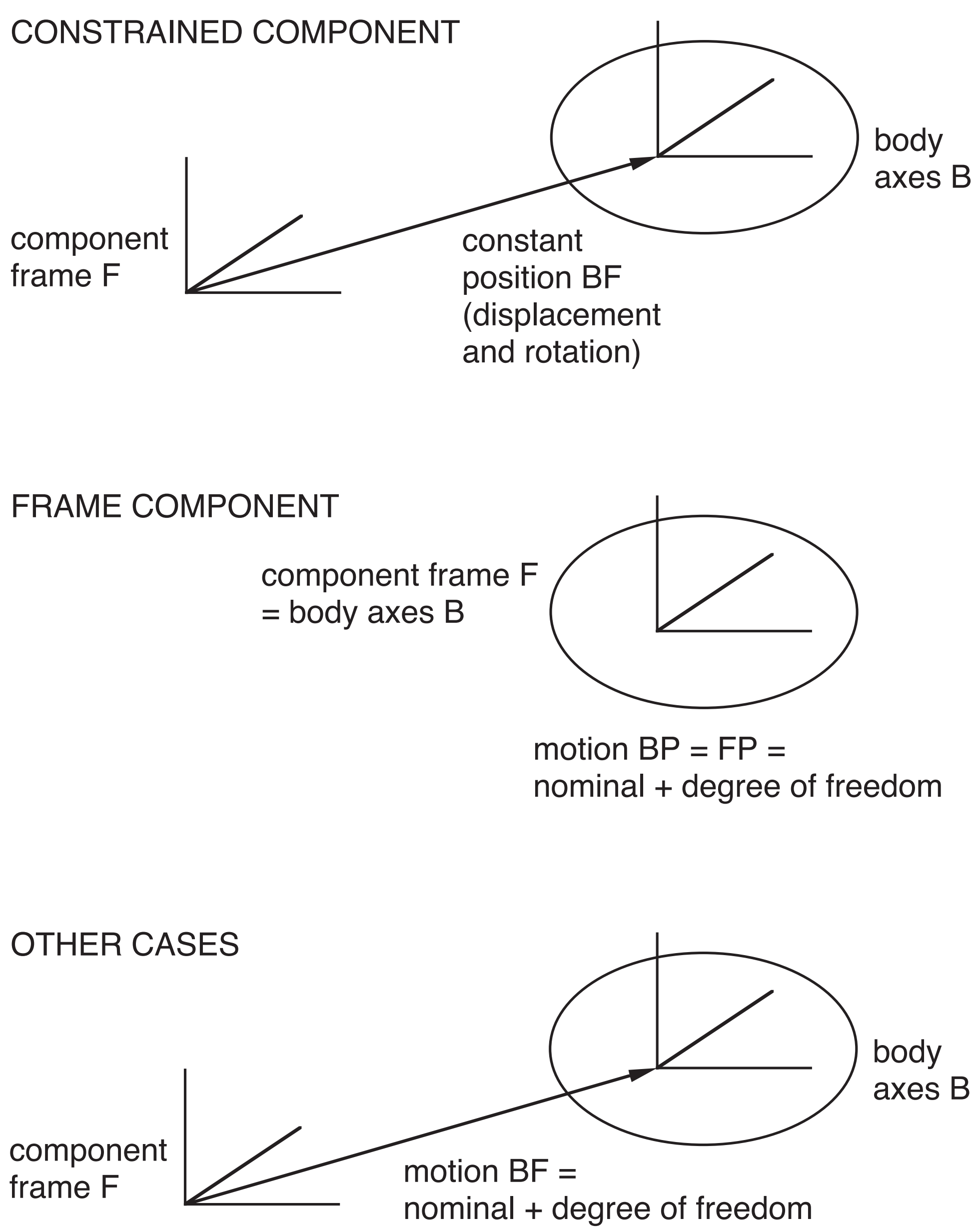


Figure 5-4 Rigid body motion of structural dynamic component.

CONVENTION TO DESCRIBE GEOMETRY AND KINEMATICS

- INTERFACES OCCUR AT CONNECTIONS (C) ON STRUCTURE
  - so all loads act at connections
- GEOMETRY SPECIFIED IN TERMS OF LOCATIONS (E)

- figure 5 summarizes construction of motion at a connection

- POSITION OF CONNECTION DESCRIBED IN STEPS:
  - I $\rightarrow$ P $\rightarrow$ F $\rightarrow$ B $\rightarrow$ E $\rightarrow$ J $\rightarrow$ C
  - position of each step consists of orientation and displacement (and derivatives if required) of an axis system, measured relative axes of preceding step
  - addition of steps involves addition of axes rigid motion, not scalar summation
  - each step may itself consist of several motions
    - any subset of motion can be calculated, as required by component analysis

FOR EACH COMPONENT, CAN DEFINE:

- any number of locations
- any number of joints at a location
- any number of connections at a location, or at a joint
- any number of uses of a connection, including interfaces and sensors
  - with some restrictions (for example, aerodynamic interfaces can not be at a joint)

C $=$ connection

J $=$ joint

E $=$ location

B $=$ body motion

F $=$ component frame

P $=$ parent frame

I $=$ inertial frame

$$\text{CI} = \text{CJ} + \text{JE} + \text{EB} + \text{BF} + \text{FP} + \text{PI}$$

PI $= \sum \mathrm{P}_i\mathrm{P}_{i-1} = \sum$ (nominal $+$ dof)

FP $=$ (nominal $+$ dof); or (dof $+$ nominal)

BF $=$ (nominal $+$ dof); or (dof $+$ nominal);

or 0 (frame degrees of freedom);

or constant BF (constrained component)

EB $=$ EL $+$ LB; or constant EB (no elastic)

JE $=$ JD $+$ DE; or 0 (no joint)

CJ $=$ constant CJ; or constant CE (no joint)

Figure 5-5a Summary of motion at a connection.

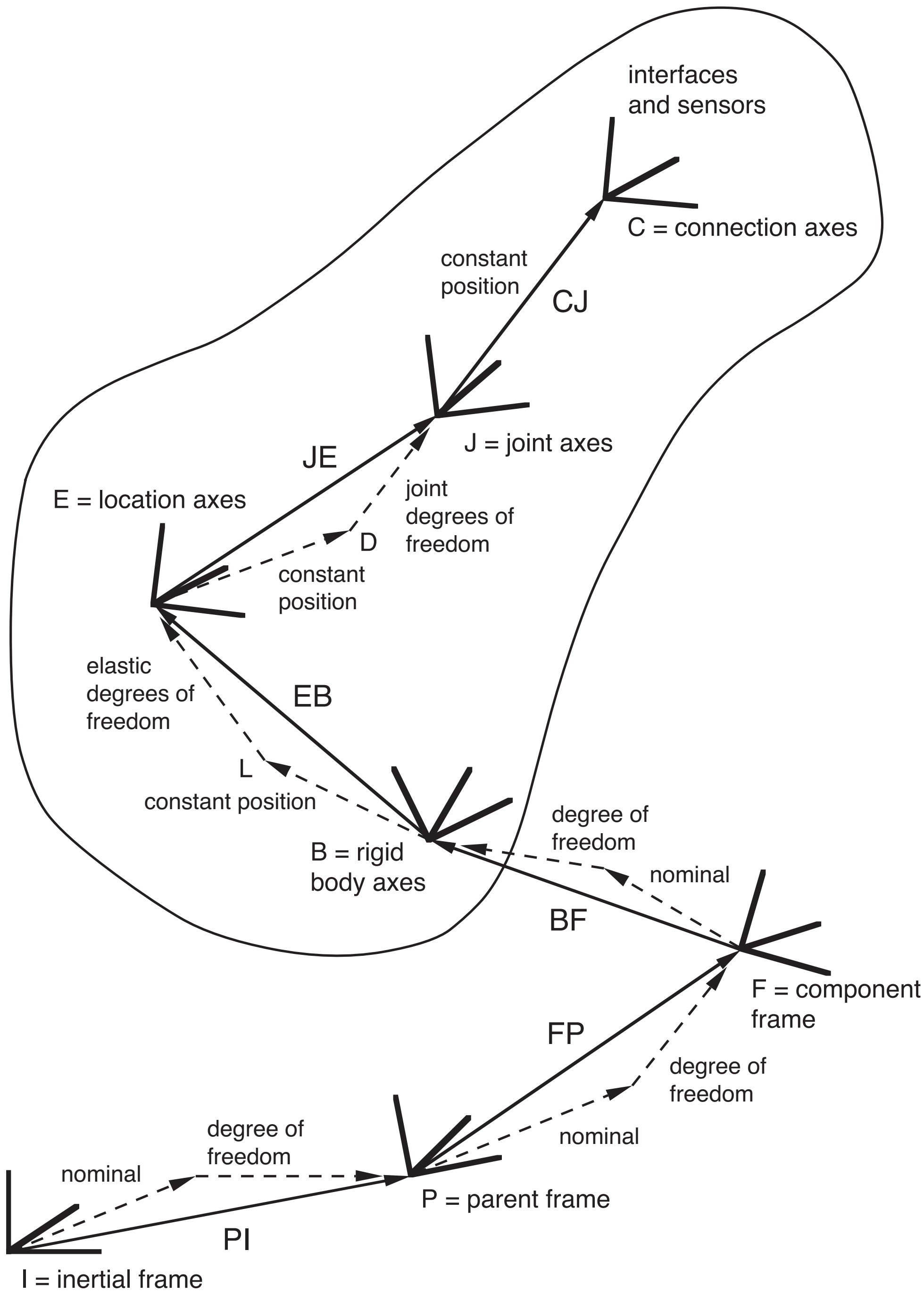


Figure 5-5b Summary of motion at a connection.

STEPS IN DESCRIPTION OF POSITION ON STRUCTURE

INERTIAL FRAME I

general reference for all motion

PARENT FRAME P

some parent frame of the component

PI is sum of all frame motions between P and inertial frame

each parent frame might be frame degrees of freedom (rigid body degrees of freedom of some other component)

COMPONENT FRAME F

designated frame of component

FP is motion of frame relative its parent

FP is sum of nominal and degree-of-freedom motions (in either order, standard rigid-type response)

BODY MOTION B

rigid body motion of component

BF is motion of component body axes relative to component frame

BF is sum of nominal and degree-of-freedom motions (in either order, standard rigid-type response)

frame component: body axes of this component are the frame (B $=$ F); so frame motion FP provides rigid body degrees of freedom of component, while motion BF is zero

constrained component: body axes of this component are fixed relative to the frame; constant position of body is specified by $z^{BF/F}$ and C^{BF} (B relative F, in F axes)

ELASTIC MOTION E

geometric location and elastic motion

EB is position of location relative body axes, $z^{EB/B}$ and C^{EB} (E relative B, in B axes)

EB is sum of elastic degrees of freedom and constant position (EB $=$ EL $+$ LB)

analysis can suppress effects of elastic motion or elastic force at location (an approximation)

JOINT MOTION J

joint motion of component (only present if joint exists at the location)

JE is position of joint relative location axes, $z^{JE/E}$ and C^{JE} (J relative E, in E axes)

JE is sum of joint degrees of freedom and constant position (JE $=$ JD $+$ DE)

CONNECTION C

position of connection on component

CJ is constant position of connection relative joint axes, $z^{CJ/J}$ and C^{CJ} (C relative J, in J axes)

if connection not at joint, then CJ is actually position relative location axes (C relative E, in E axes)

5-2.2 Joints

CONVENTION TO DEFINE JOINTS

JOINT MOTION OCCURS AFTER RIGID BODY AND ELASTIC MOTION, WITH NO MASS ON CONNECTION SIDE OF JOINT

so joint equations of motion obtained from just balance of forces acting on the joints

there must be at least one interface producing forces on the joint

MORE COMPLEX MOTIONS MODELLED BY USING SEVERAL COMPONENTS

JOINT KINEMATICS DESCRIBE POSITION OF JOINT AXES RELATIVE TO A LOCATION (JE)

JE is sum of joint degrees of freedom and constant position (JE $=$ JD $+$ DE)

constant position of joint (for zero joint variables) relative location axes is specified by $z^{DE/E}$ and C^{DE} (D relative E, in E axes)

then joint motion occurs, described by $x^{JD/D}$ and C^{JD} (and derivatives)

JOINT CAN CONSIST OF

just linear motion
just angular motion
linear followed by angular motion
angular followed by linear motion

CONFIGURATIONS FOR LINEAR MOTION

1-linear (slide, prism, linear hinge)
2-linear (plane)
3-linear (space)

described by one to three variables (with gain factor)

axes can be arbitrarily specified

CONFIGURATION FOR ANGULAR MOTION

hinge (hinge, revolute)
2-hinge (universal)
3-hinge (hinge plus universal)

hinge (Rodrigues parameters)
gimbal (2-Rodrigues)
rod end (3-Rodrigues, ball, spherical, pinned)

hinge (third variable) then gimbal
gimbal then hinge (first variable)

described by one to three variables (with gain factor; first variable is last rotation)

axes can be arbitrarily specified

Rodrigues parameters are convenient way to generate joint with all rotations treated same

LINEAR OR ANGULAR DISPLACEMENT IN JOINT EQUALS JOINT VARIABLE θ TIMES GAIN FACTOR

CONVENTIONS WILL HANDLE COMBINED MOTIONS

such as screw, rack and pinion, or transmission

EACH JOINT VARIABLE θ CAN BE DEGREE OF FREEDOM, PRESCRIBED, OR CONTROLLED:

DEGREE OF FREEDOM

joint degree of freedom can be shared by more than one variable, of one or more than one joint

each joint degree of freedom (involving one or more joint variables) has equation of motion obtained by equilibrium of loads on the joints:

$f_{\text{spring}} = f_{\text{interfaces}}$

reaction f_{spring} obtained from spring/damper/ actuator model

load $f_{\text{interfaces}}$ obtained by summing effects of all interface loads acting on all joints involved in this degree of freedom

PRESCRIBED

joint motion can be constant velocity, rotating variable, or constant position

CONTROLLED

- joint motion can be obtained from a control vector (component input), implementing a displacement actuator
- if motion is time varying, this control vector should include joint velocity and acceleration, as well as joint displacement
  - force or offset actuator can be implemented using spring/damper/actuator convention for joint degree of freedom
  - controlled joint can not be used in flutter task
    - joint degree of freedom with offset actuator can be used instead (and the degree of freedom can be quasistatic)
  - controlled joint can be used in trim and transient tasks, as long as control variable is solved in separate part

STRUCTURAL DYNAMIC TORQUE INTERFACE ONLY ALLOWED ON CERTAIN JOINTS

JOINTS WHERE LAST MOTION (FIRST VARIABLE) IS ANGULAR ROTATION ABOUT AN AXIS

5-2.3 Spring/Damper/Actuator

STANDARD SPRING/DAMPER/ACTUATOR MODEL USED TO PROVIDE REACTION FORCE AS REQUIRED BY COMPONENT

USED FOR JOINTS, ROD/CABLES, OR TRANSMISSION BRANCHES

component can use reaction force in only one context, depending on component type

FOR EACH DEGREE OF FREEDOM θ (SCALAR ELEMENT OF DEGREE-OF-FREEDOM VECTOR):

SCALAR REACTION FORCE f IS GENERATED

$$f_{\text{spring}} = f(\theta, \dot{\theta}, c)$$

c = optional control (element of control vector) for an actuator model

FORCE EVALUATED AS SUM OF SPRING, DAMPER, TABLE, AND BIAS TERMS

$$f_{\text{spring}} = K(\theta - \theta_0) + C(\dot{\theta}, \theta - \theta_0) + T(\dot{\theta}, \theta - \theta_0, s) - c_0$$

with offset θ_0 and bias c_0

offset θ_0 usually value of θ for zero load;
bias c_0 gives load for zero displacement

s is optional switch in table

optionally can support only compression, or only tension

ACTUATOR ADDED BY INTRODUCING CONTROL c

force actuator, c_0 replaced by $(c_0 + c)$
offset actuator, θ_0 replaced by $(\theta_0 + c)$

for very large stiffness, offset actuator should produce $\theta = \theta_0 + c$, which is a displacement actuator

joint has separate implementation for displacement actuator

figure 6 summarizes the actuator models

FOR LINEAR SPRING AND LINEAR DAMPER:

$f_{\text{spring}} = K\theta + C\dot{\theta} - (K\theta_0 + c_0)$	no actuator
$f_{\text{spring}} = K\theta + C\dot{\theta} - (K\theta_0 + c_0 + Kc)$	offset actuator
$f_{\text{spring}} = K\theta + C\dot{\theta} - (K\theta_0 + c_0 + c)$	force actuator

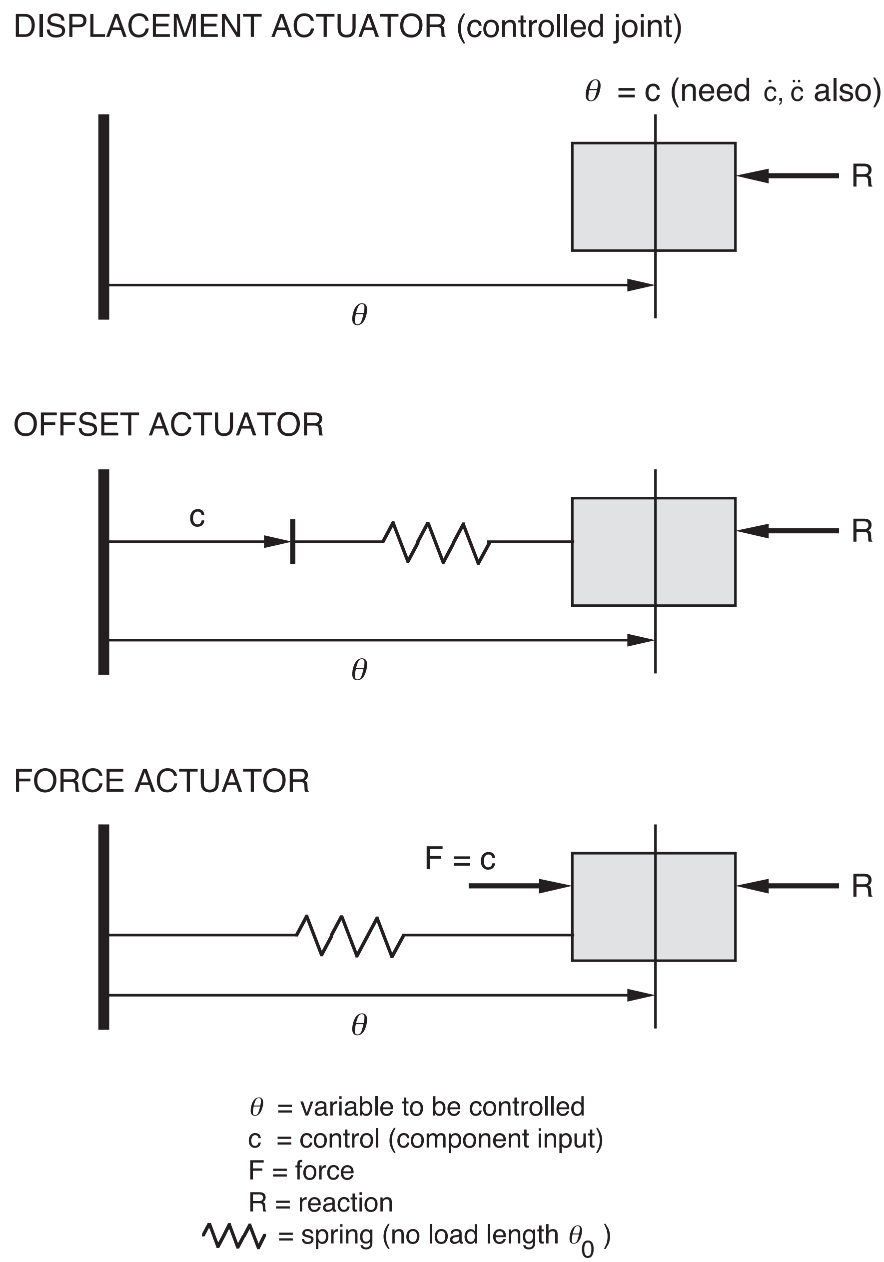


Figure 5-6 Summary of actuator models.

SPRING HAS LINEAR AND ELASTOMERIC TERMS
DAMPER HAS LINEAR, ELASTOMERIC, AND HYDRAULIC TERMS

$$K(\theta) = K_{\text{lin}}\theta + K_{\text{elast}}$$
$$C(\dot{\theta}, \theta) = C_{\text{lin}}\dot{\theta} + C_{\text{elast}} + C_{\text{hyd}}$$

used with an offset θ_0

linear terms have constant spring and damping coefficients

ELASTOMERIC SPRING/DAMPER DESCRIBED BY POLYNOMIALS IN DISPLACEMENT θ AND RATE $\dot{\theta}$:

$$K_{\text{elast}} = \text{sign}(\theta)(P_{ekd}(|\theta|) + P_{ekr}(|\dot{\theta}|))$$
$$C_{\text{elast}} = \text{sign}(\dot{\theta})(P_{ecd}(|\theta|) + P_{ecr}(|\dot{\theta}|))$$

each polynomial is fifth-order (six terms)

P_{ekd} term is nonlinear spring
P_{ecr} term is nonlinear viscous damping
P_{ecd} term is friction damping

necessary to identify coefficients of this time domain model from measured force and motion data (usually for harmonic motion at one or two frequencies)

HYDRAULIC DAMPER DESCRIBED BY POLYNOMIALS IN RATE $\dot{\theta}$:

$$C_{\text{hyd}} = \text{sign}(\dot{\theta}) \min(P_{ha}(|\dot{\theta}|),\ P_{hb}(|\dot{\theta}|))$$

each polynomial is fifth-order (six terms)

hydraulic damper typically has

$$P_{ha} = c_a\dot{\theta}^2 = f_{\max}(\dot{\theta}/\dot{\theta}_{\max})^2$$
$$P_{hb} = c_b \quad = f_{\max}$$

MORE GENERAL BEHAVIOR CAN BE IMPLEMENTED BY OBTAINING FORCE $T(\dot{\theta}, \theta)$ FROM A TWO-DIMENSIONAL TABLE

or $T(\dot{\theta}, \theta, s)$ from a three-dimensional table, including switch parameter s (perhaps $s =$ time)

NONLINEAR SPRING/DAMPER CAN BE DESCRIBED IN TERMS OF EQUIVALENT LINEAR SPRING AND LINEAR DAMPER

VALUES OF EQUIVALENT K AND C DEPEND ON AMPLITUDE AND FREQUENCY OF MOTION

such behavior depends on finite amplitude of motion, so can not be obtained by local linearization of force

SO WHEN PERTURB SPRING/DAMPER/ACTUATOR, CAN OPTIONALLY USE

$$\begin{aligned} K(\theta) &= K_{\text{equiv}}\theta \\ C(\dot{\theta}, \theta) &= C_{\text{equiv}}\dot{\theta} \\ T(\dot{\theta}, \theta) &= 0 \end{aligned}$$

for the trim, transient, or flutter task

5-2.4 Interfaces

STRUCTURAL DYNAMIC INTERFACE

ONLY STRUCTURAL DYNAMIC COMPONENT CAN HAVE STRUCTURAL DYNAMIC INPUT AND OUTPUT

FOR CONNECTION TO ANOTHER COMPONENT THROUGH STRUCTURAL DYNAMIC INTERFACE

interface occurs at connection (C)

INTERFACE KIND CAN BE COMPLETE OR TORQUE

COMPLETE INTERFACE: TRUE PHYSICAL CONNECTION

CONNECTS TWO COMPONENTS RELATIVE THEIR COMMON PARENT FRAME (P)

component input is force and moment acting on the connection

component output is motion of axes at the connection, relative common frame axes ($x^{CP/P}$ and C^{CP}, and their derivatives)

TORQUE INTERFACE

DEALS WITH ONLY ROTATIONAL MOTION AND TORQUES (APPROXIMATION)

can be used only on appropriate angular joint

produces no net moment on true structural dynamic component; only involved in joint equations

AERODYNAMIC INTERFACE

STANDARD INPUT/OUTPUT INTERFACE (COMPONENT INPUT AND/OR COMPONENT OUTPUT)

FOR INTERFACES WITH AERODYNAMIC COMPONENTS

interface occurs at connection on structure (which can not be at a joint)

AERODYNAMIC COMPONENT TYPICALLY INVOLVES SET OF COLLOCATION POINTS

task of calculating geometry of aerodynamic collocation points belongs to structural dynamic component, which knows about physical configuration

calculating velocity of collocation point relative to air also belongs in structural dynamic component, since requires kinematics of connection

connection that is collocation point typically has aerodynamic interfaces for:

interference and gust velocities (input)
velocity relative air (output)
position (output)
resulting aerodynamic load (input)

POSSIBLE COMPONENT INPUT

force (F)

force and moment (F and M)

interference velocity (v_A)

gust velocity (v_G)

gust angular velocity (ω_G)

at connection; in body, frame, or parent frame axes

POSSIBLE COMPONENT OUTPUT

velocity relative the air (v, q, $\dot{v}$, ω)

position relative origin of axes (r and $\dot{r}$)

or subset of these quantities

at connection; in connection, body, frame, or parent frame axes

velocity relative the air calculated using all interference and gust velocities (input) at this connection; and wind velocity, including ground boundary layer

time derivatives of interference and gust velocities are not considered

figure 7 illustrates calculation of velocity relative the air

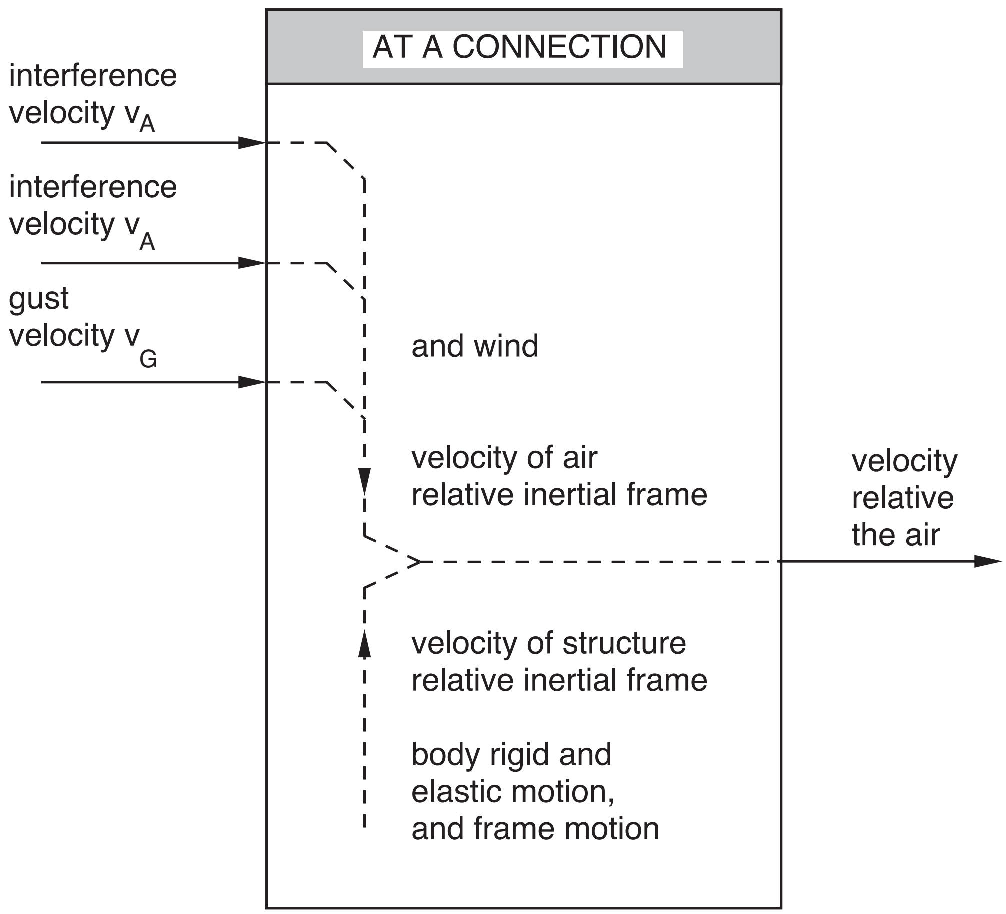


Figure 5-7 Aerodynamic interface to calculate velocity relative the air.

APPLIED LOAD INTERFACE

APPLIED FORCE CAN ACT ON COMPONENT

interface occurs at connection on structure

FORCE ACTS AT CONNECTION

SPECIFY AXIS OF FORCE IN C FRAME

u = x, y, or z-axis direction

MAGNITUDE OF FORCE IS GAIN a TIMES VALUE f OF SOME ELEMENT OF CONTROL VECTOR (COMPONENT INPUT)

SO APPLIED FORCE IS $F^C = uaf$

OR APPLIED MOMENT IS $M^C = uaf$

5-2.5 Controls and Sensors

CONTROLS

SET OF CONTROL VECTORS AVAILABLE TO COMPONENT

COMPONENT INPUT, FOR CONNECTION TO SYSTEM INPUT PIECE OR TO INPUT/OUTPUT INTERFACE

a particular control vector and element can be used once, more than once, or not at all

STANDARD FEATURES OF STRUCTURAL DYNAMIC COMPONENT CAN USE CONTROLS FOR:

JOINTS
ACTUATORS
APPLIED LOAD INTERFACES

COMPONENT-SPECIFIC USES ARE PERMITTED

SENSORS

SET OF SENSOR VECTORS CAN BE DEFINED FOR COMPONENT

COMPONENT OUTPUT, FOR CONNECTION TO SYSTEM OUTPUT PIECE OR TO INPUT/OUTPUT INTERFACE

STANDARD SENSORS MEASURE:

- motion
  - motion of axes at a point
  - velocity relative air
  - aerodynamic interface motion
  - degrees of freedom
  - motion of torque interface
- structural dynamic interface constraint
  - force and moment
  - torque
- reaction
  - total load at point
  - joint degree of freedom
  - weight and load factor
- power
  - joint
  - joint degree of freedom

VALUE OF ANY SENSOR CAN BE MULTIPLIED BY SCALE FACTOR

- automatic scaling for value in g's, degrees, Mach number, or horsepower

THERE MAY ALSO BE COMPONENT-SPECIFIC SENSORS AVAILABLE

5-2.6 Structural Dynamic Components

ALL STRUCTURAL DYNAMIC COMPONENTS HAVE STANDARD FEATURES

differ in modelling of elastic motion

COMPONENT MOTION DESCRIBED BY

rigid body motion, which can always be large (exact kinematics)

plus elastic motion, which is measured relative rigid motion

exact kinematics for all structural dynamic interfaces

STRUCTURAL DYNAMIC COMPONENTS IMPLEMENTED

RIGID BODY COMPONENT
no elastic motion

LINEAR NORMAL MODES COMPONENT
elastic motion represented by free normal modes

FINITE ELEMENT BEAM COMPONENT
elastic motion represented by beam deflection (axial, bending, torsion)

ROD/CABLE COMPONENT
elastic motion represented by flexible rod or cable between two points

5–3 Linear Normal Modes Component

STRUCTURAL DYNAMIC COMPONENT

- ELASTIC MOTION DESCRIBED BY FREE-VIBRATION NORMAL MODES
  - typically modes obtained from large finite-element analysis
  - if component is constrained, then constrained modes can be used

RIGID MOTION DESCRIBES CENTER-OF-MASS, MEAN-AXES MOTION OF BODY

- SO RIGID BODY DEGREES OF FREEDOM ARE ORTHOGONAL TO FREE-VIBRATION MODES
  - elastic motion is small (consistent with using linear modes), measured relative to rigid motion

AERODYNAMIC SPRING, DAMPING, AND CONTROL TERMS INCLUDED

- approximation to effects of aerodynamics on elastic motion

MODAL SENSORS INCLUDED

- to calculate quantities proportional to modal deflection (loads, stress, strain)

LINEAR NORMAL MODES

MODES CALCULATED FOR FREE BODY, INCLUDING RIGID MOTION

small motion is assumed, consistent with linear modes

modes are orthogonal

no effect of frame motion on elastic modes

no effect of elastic motion on inertial properties of rigid modes

rigid modes represent motion of center-of-mass, mean-axes of body

MODEL EXTENDED BY REPLACING RIGID MODES BY LARGE RIGID BODY MOTION OF BODY AXES B

origin of body axes must be at center-of-mass

but not necessary to use principal axes

elastic modes represent motion relative to the body axes

elastic and rigid equations are still fully decoupled

MODE DESCRIBED BY:

- modal mass, spring, structural damping
- mode shapes at connections where forces and moments are applied
  - 3 linear and 3 angular displacements
    - defined at locations (in body or undeflected location axes)
  - input using namelist arrays, or table file
- aerodynamic spring and damping, and modal force of control
  - scaled with air density and velocity relative air

FOR REFERENCE IN PREPARING INPUT PARAMETERS, MODAL EQUATION OF MOTION HAS FOLLOWING FORM (DIMENSIONAL):

$$M\left[\ddot{q} + g\omega\dot{q} + \omega^2 q\right] = \xi^T F + \gamma^T M + {}^1\!/\!{}_2\rho V^2\left[F_{q\delta}\delta - F_{q\dot{q}}(\dot{q}/V) - F_{qq}q\right]$$

5–4 Finite Element Beam Component

STRUCTURAL DYNAMIC COMPONENT

ELASTIC MOTION DESCRIBED BY AXIAL, BENDING, AND TORSION DEFLECTION OF A BEAM

RIGID MOTION IS MOTION OF ONE END OF BEAM

DO NOT USE NODAL COORDINATES AS DEGREES OF FREEDOM

elastic motion measured relative to rigid motion

elastic motion is moderate for second-order model; rigid motion can always be large

GENERAL GEOMETRY OF BEAM

component has straight beam axis

any number of locations (on beam axis)

any number of joints at a location

any number of connections at a location or at a joint

joints and connections are typically in a beam section: in a plane normal to bent beam axis

any number of interfaces and sensors at a connection

figure 8 illustrates use of beam component

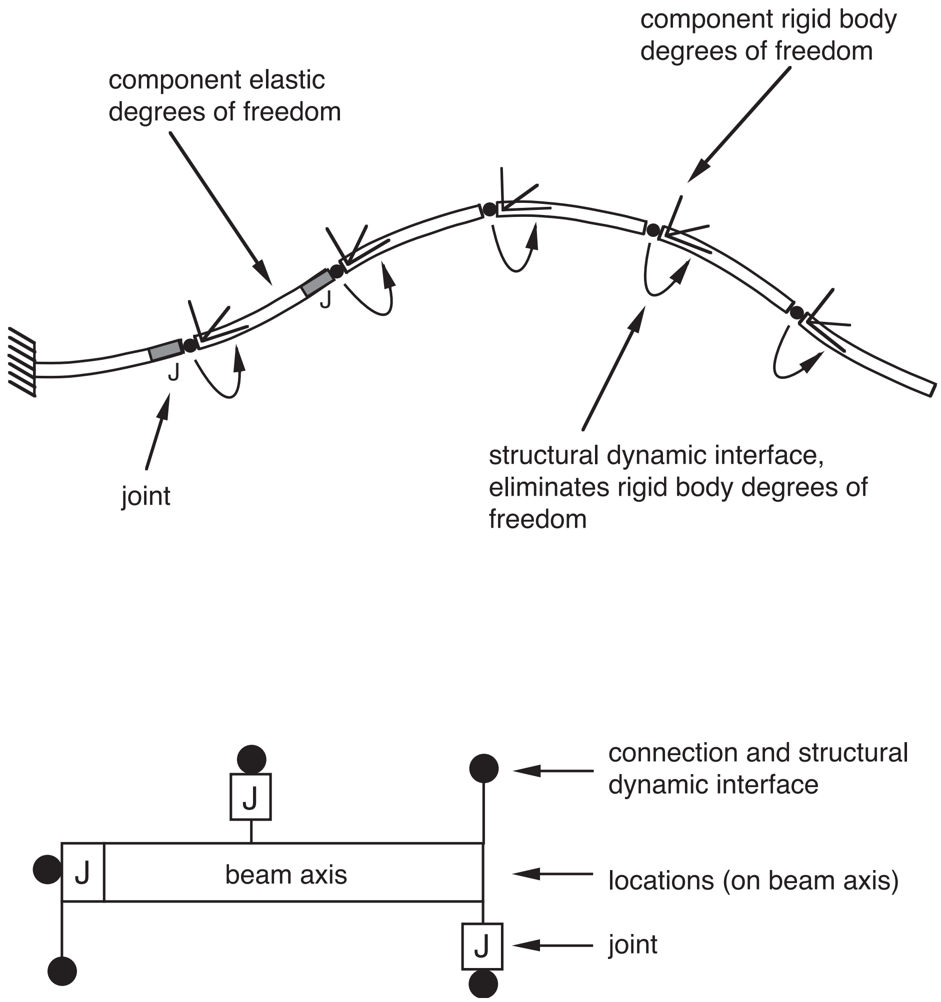
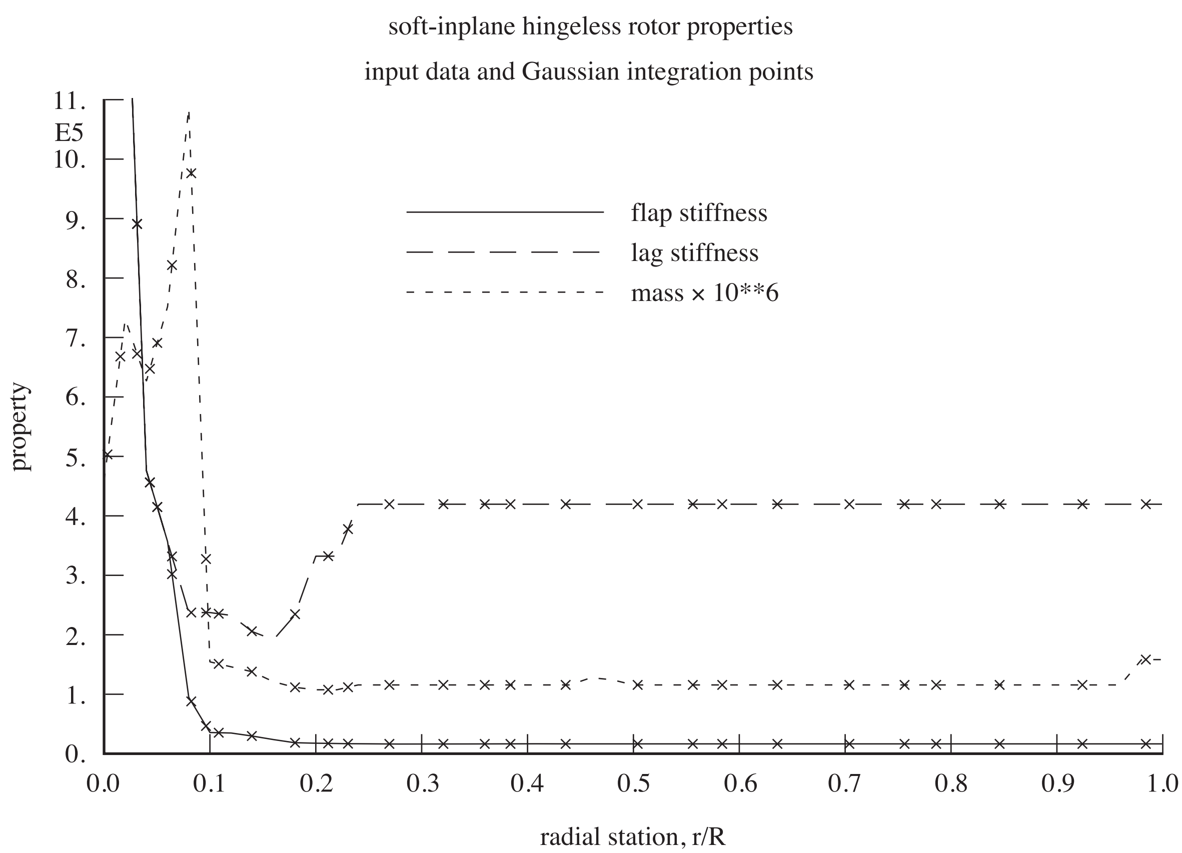


Figure 5-8 Use of beam component.

MODEL ROTOR BLADE BY BREAKING STRUCTURE INTO SEVERAL BEAM SEGMENTS

- “nodes” are points where segments are joined
  - in CAMRAD II terminology: connections at joint/location on each structural dynamic component, with structural dynamic interface between
- insert node to handle beam axis (perhaps the elastic axis) with curvature, kinks, or jumps
  - beam segment has straight beam axis
- insert node to handle properties that vary rapidly along blade span (major jumps in properties at nodes)
  - Gaussian integration along beam segment (implying polynomial approximation to variation of properties, accurate only if variation is sufficiently smooth)
  - continuous shape functions for elastic deflection within beam segment (can not accurately represent large changes in curvature or slope)
- if very short beam segments are required to accommodate properties, then beam theory is probably not applicable


soft-inplane hingeless rotor properties
input data and Gaussian integration points
flap stiffness
lag stiffness
mass × 10**6
property
E5
11.
10.
9.
8.
7.
6.
5.
4.
3.
2.
1.
0.
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
radial station, r/R

BEAM MODEL

MATERIAL

Euler-Bernoulli beam theory for isotropic material, with elastic axis; or

beam theory for anisotropic or composite materials

- effects of cross-section warping included in section properties
- for anisotropic model, effects of transverse shear deformation included in section properties
- structural model assumes strain is small

GEOMETRY

kinematics of elastic motion are exact, almost-exact, or second-order

- almost-exact model still uses second-order approximation for extension and torsion produced by bending
- second-order model retains only second-order effects of elastic motion in potential and kinetic energies (hence moderate deflection)

cross-section rotation produced by bending must be less than 90 deg within beam component

- so very large elastic motion must be modelled using several beam components

STRAIGHT BEAM AXIS (UNDEFLECTED), LENGTH ℓ

BEAM AXIS IS ELASTIC AXIS FOR ISOTROPIC MODEL

beam axis is positive x-axis of B frame, beam extending from $x = 0$ to $x = \ell$

motion BF describes orientation and displacement of beam end, relative the frame

nominal part of BF typically is rest position of beam relative frame

figure 9 illustrates the configuration

LOCATIONS, JOINTS, AND CONNECTIONS

locations are on beam axis at axial station x

undeflected location axes parallel body axes

elastic motion produces additional displacement and rotation of location axes

so E axes are bent and twisted section axes, with origin on beam axis

connection has constant position (offset and rotation) relative location axes

or joint has constant position relative location axes, and connection has constant position relative joint axes

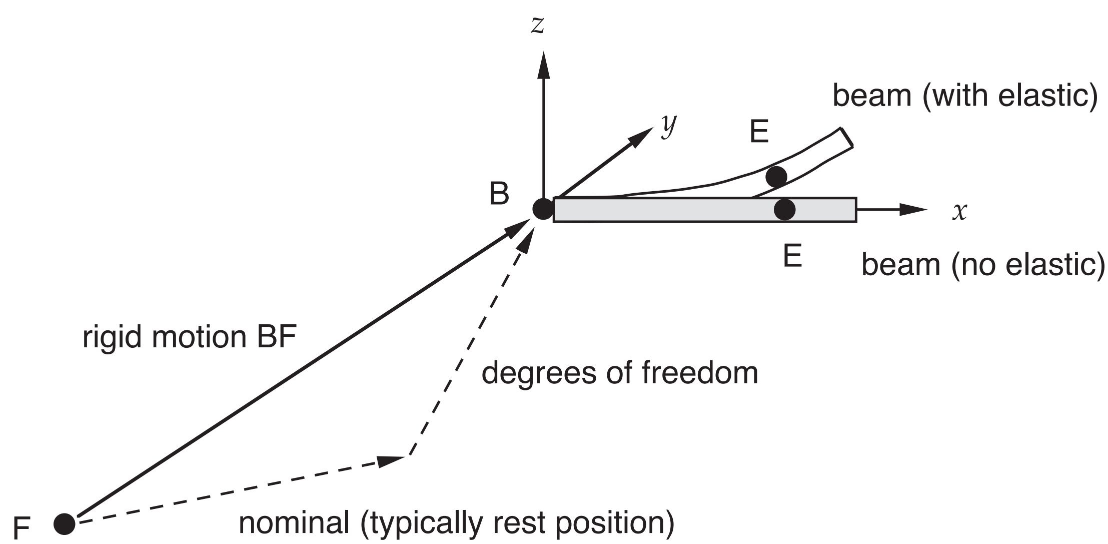


Figure 5-9 Beam configuration.

BEAM PROPERTIES

ARBITRARY LOCUS OF CENTER-OF-GRAVITY
ARBITRARY LOCUS OF TENSION CENTER

properties defined relative beam axes

- displacements y and z measured from beam axis (the x-axis)
- pitch angles are measured from x–y plane, positive for rotation about x-axis

structural properties

- θ_C: pitch of structural principal axes, relative x–y plane; pitch angle can be large
- y_C and z_C: offset of tension center (modulus-weighted centroid) from beam axis, relative principal axes (at θ_C)
- k_P: modulus-weighted radius of gyration, about the beam axis
- EA, EI_{yy}, EI_{zz}, GJ, k_T (isotropic)
- S_{uu}, $S_{\phi u}$, S_{wu}, S_{vu}, $S_{\phi\phi}$, $S_{w\phi}$, $S_{v\phi}$, S_{ww}, S_{vw}, S_{vv} (anisotropic)

inertial properties

- θ_I: pitch of inertial principal axes, relative x–y plane; pitch angle can be large
- y_I and z_I: offset of center of gravity (mass-weighted centroid) from beam axis, relative principal axes (at θ_I)
- m, I_θ, I_P

Gaussian integration of properties along beam length

input data use piecewise linear definition of properties

BEAM DEFLECTION

- axial deflection (u), then bending deflection (v then w), then torsion (ϕ)
  - axial and torsion degrees of freedom exclude motion produced by bending kinematics
    - so zero axial degree of freedom is solution for large EA
    - and zero torsion degree of freedom is solution for large GJ
- elastic deflections represented by polynomial shape functions
  - maximum 3 shape functions for bending
  - maximum 4 shape functions for axial or torsion
    - typically use $uvw\phi = 3222$ for rotor blades (total 15 degrees of freedom, 6 rigid and 9 elastic)
  - unit deflection (1 ft or m, 1 radian) at end of beam
- coefficients of shape functions are elastic degrees of freedom
- include model of structural damping for each elastic degree of freedom

SENSORS

SECTION LOAD

component specific sensor

torsion and bending moments, axial tension, and shear forces at a location (axial station x_L); acting at tension center, in structural principal axes

calculated from deflection, or by force balance

DEFLECTION METHOD

section load from stiffness and elastic displacement at x_L

bending moment = EI times curvature
torsion moment = GJ times slope

accuracy depends on accuracy of representation of curvature or slope (product of degrees of freedom and shape functions)

at step in stiffness should be corresponding step in curvature or slope, so load remains continuous

with small number of shape functions not possible to simulate such a step well

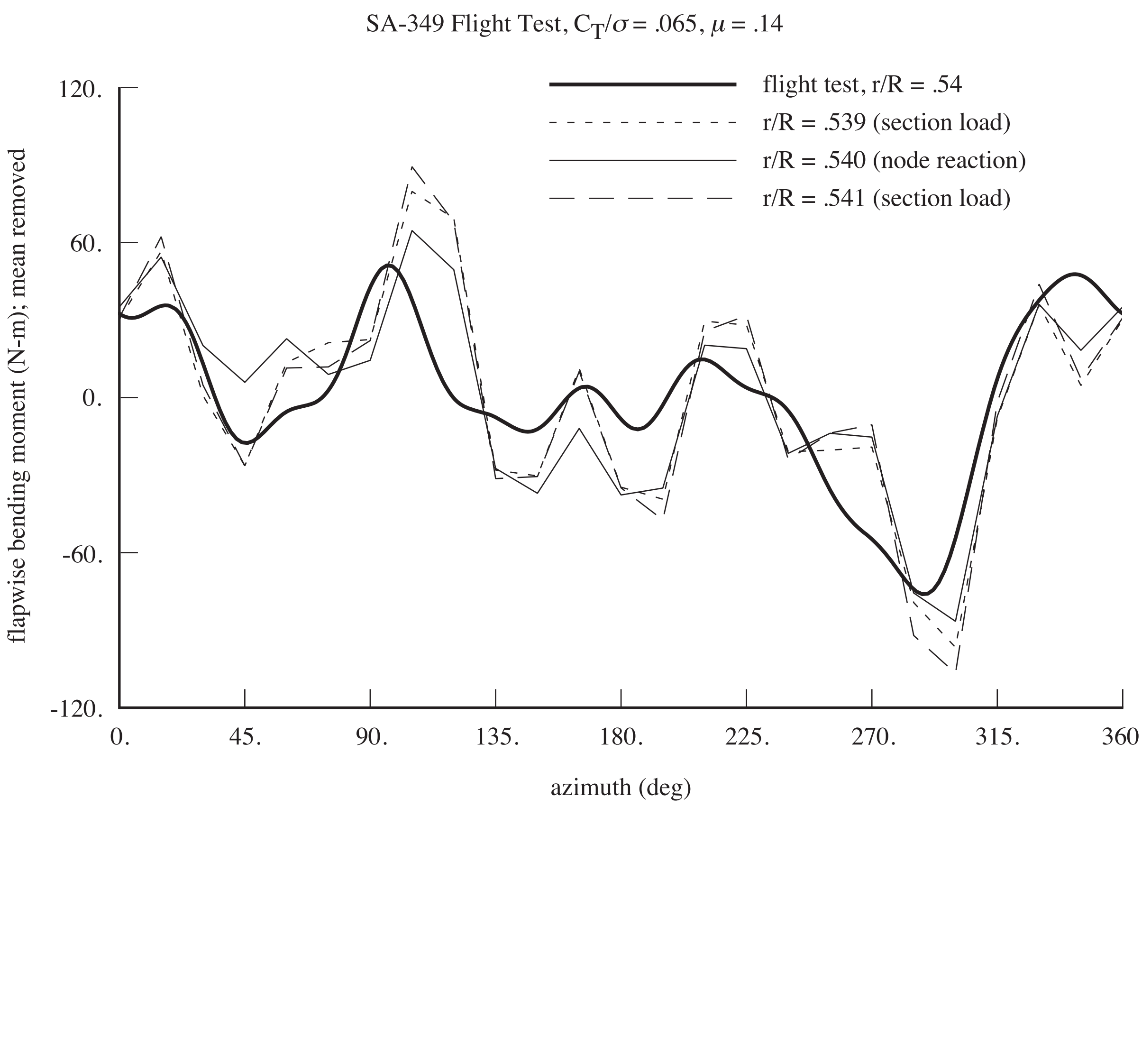
theory does not imply continuity of curvature on two sides of node

FORCE BALANCE METHOD

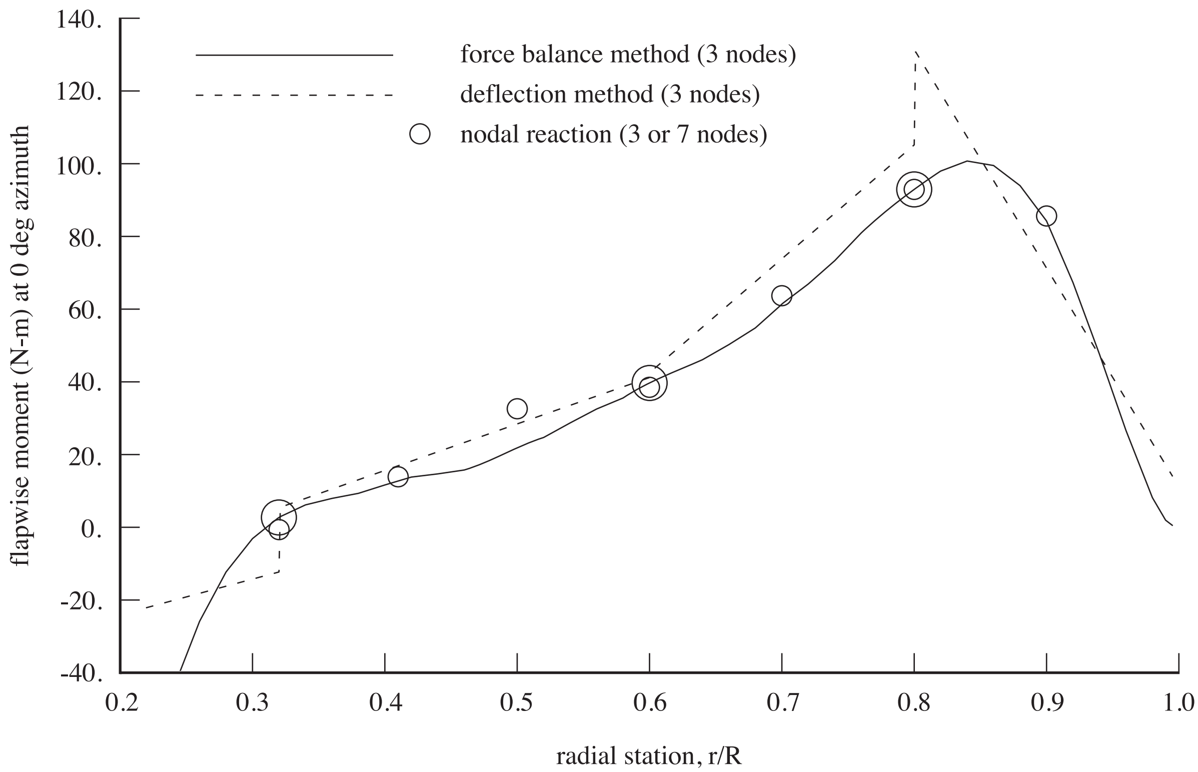
- section load from difference between applied forces and inertial forces acting on beam segment to one side of x_L
- sensor gives at beam ends same result as nodal reaction
  - can capture steps in section load produced by discrete forces and moments on beam
  - aerodynamic loads integrated as distributed forces and moments on beam

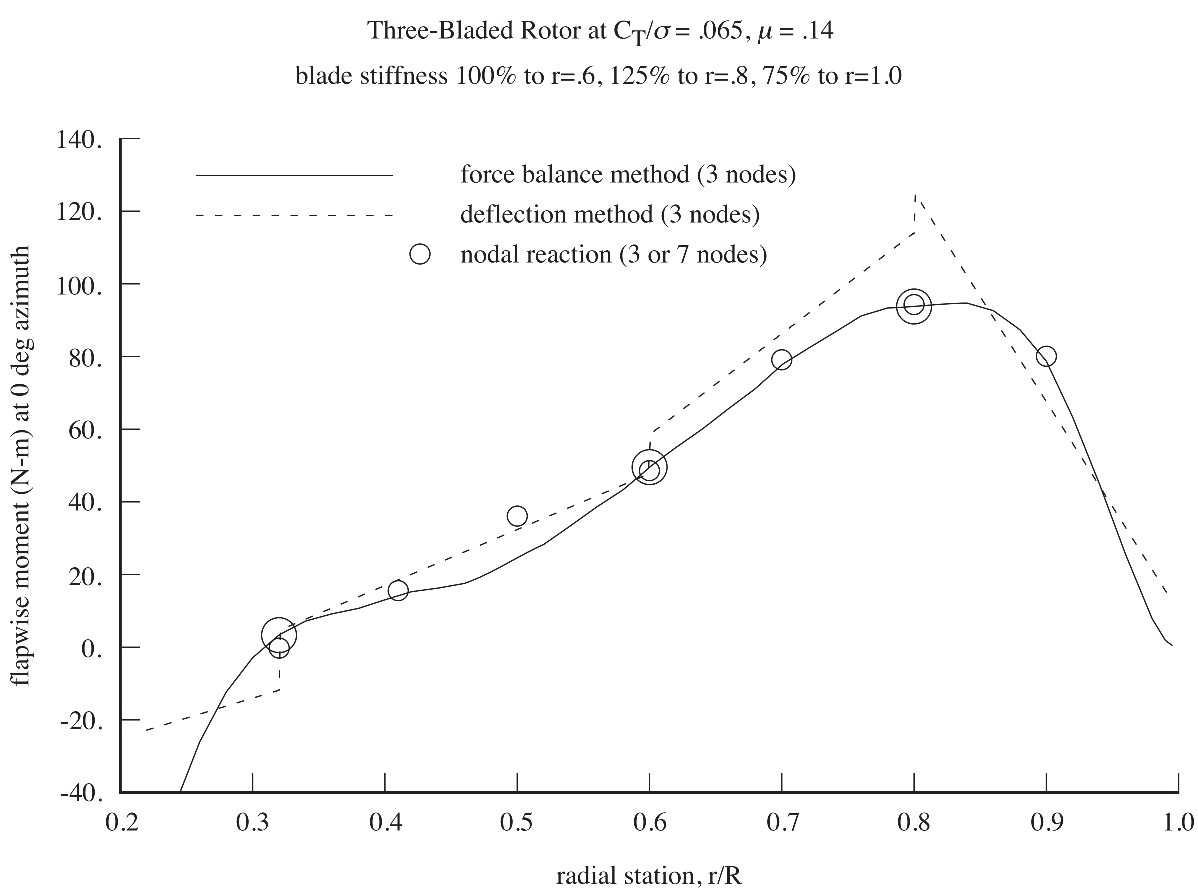
NODAL REACTION

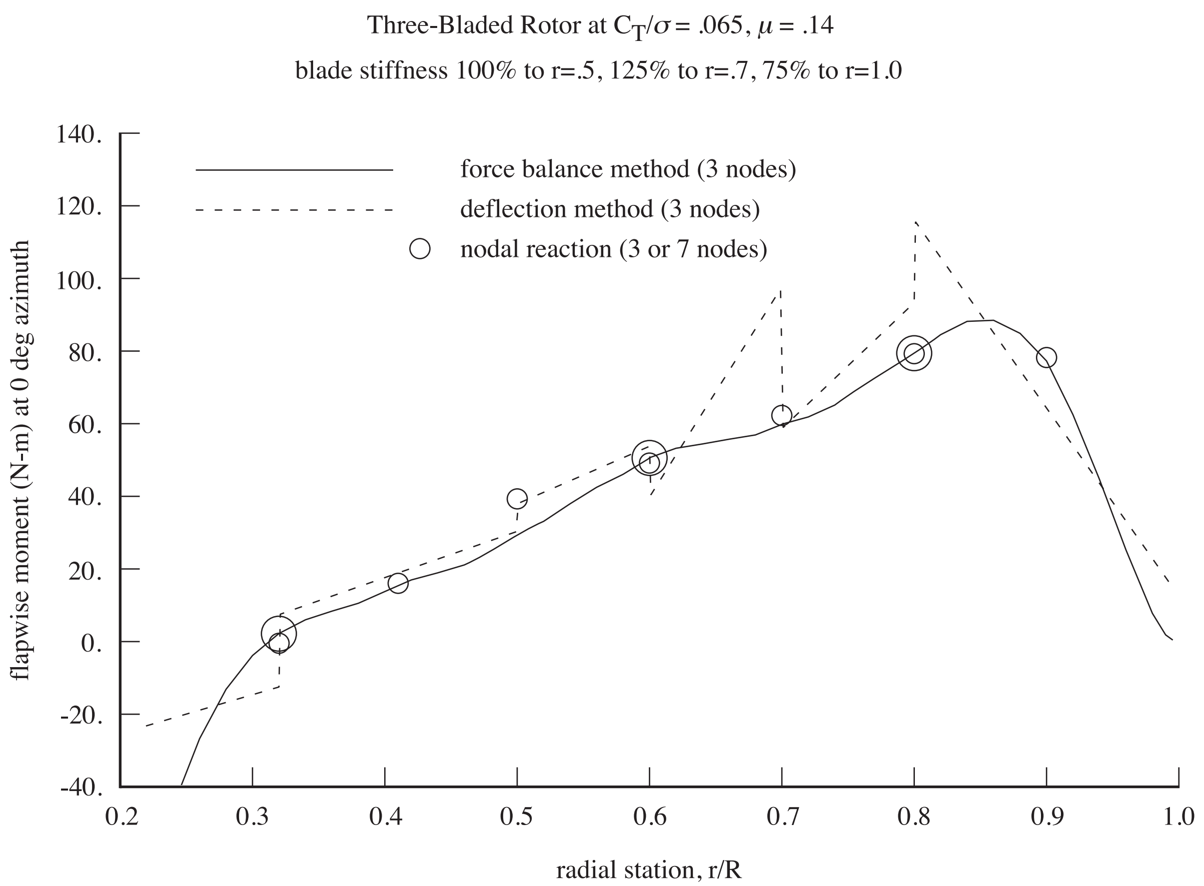
- can also calculate beam reaction using standard sensor for structural dynamic load
  - about point at tension center, in structural principal axes
- accuracy only depends on tolerance in solution for equilibrium of beam
- but need node (structural dynamic interface) at sensor point


SA-349 Flight Test, C_T/σ = .065, μ = .14
flight test, r/R = .54
r/R = .539 (section load)
r/R = .540 (node reaction)
r/R = .541 (section load)
120.
60.
0.
-60.
-120.
flapwise bending moment (N-m); mean removed
0.
45.
90.
135.
180.
225.
270.
315.
360.
azimuth (deg)

Three-Bladed Rotor at $C_T/\sigma = .065$, $\mu = .14$
uniform blade properties




Three-Bladed Rotor at C_T/σ = .065, μ = .14
blade stiffness 100% to r=.6, 125% to r=.8, 75% to r=1.0
force balance method (3 nodes)
deflection method (3 nodes)
nodal reaction (3 or 7 nodes)
flapwise moment (N-m) at 0 deg azimuth
140.
120.
100.
80.
60.
40.
20.
0.
-20.
-40.
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
radial station, r/R


Three-Bladed Rotor at C_T/σ = .065, μ = .14
blade stiffness 100% to r=.5, 125% to r=.7, 75% to r=1.0
force balance method (3 nodes)
deflection method (3 nodes)
nodal reaction (3 or 7 nodes)
flapwise moment (N-m) at 0 deg azimuth
140.
120.
100.
80.
60.
40.
20.
0.
-20.
-40.
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
radial station, r/R

5–5 Rod/Cable Component

STRUCTURAL DYNAMIC COMPONENT

- ELASTIC MOTION DESCRIBED BY SET OF FLEXIBLE RODS OR CABLES, EACH CONNECTING TWO POINTS
  - figure 10 illustrates configuration

RIGID MOTION IS MOTION OF CENTRAL BODY, WHICH CONTAINS ALL THE MASS

- ONE END OF ROD/CABLE CONNECTED TO LOCATION ON CENTRAL BODY
- ELASTIC DEGREES OF FREEDOM ARE POSITION OF OTHER END OF ROD/CABLE
  - measured relative to body location (hence relative to rigid motion)

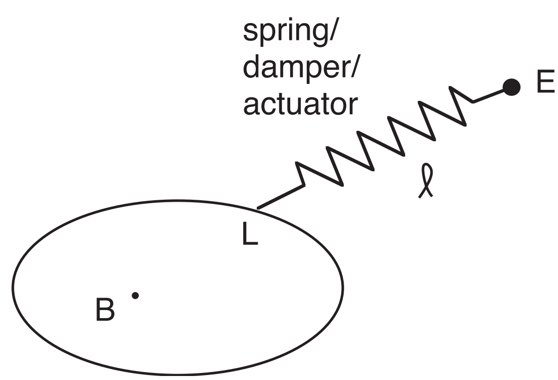


Figure 5-10 Rod/cable configuration.

ROD/CABLE DOES NOT SUPPORT MOMENTS

ENDS ARE PINNED; SO AT ROD/CABLE LOCATION:

no cantilever structural dynamic interfaces

no interface can produce moment

no joints

SPRING/DAMPER/ACTUATOR CONNECTS ROD/CABLE ENDS, TRANSMITTING FORCE THAT IS ALWAYS DIRECTED BETWEEN TWO POINTS

CABLE MODELLED BY ASSUMING THAT SPRING CAN NOT SUPPORT COMPRESSION LOAD

bending dynamics of cable are neglected

COMPONENT SPECIFIC SENSORS

to measure motion, length, reaction, power (of rod/cable)

5–6 Transmission Component

TRANSMISSION COMPONENT MODELS SUBSYSTEM THAT TRANSMITS ROTATIONAL MOTION AND TORQUES

APPROXIMATION TO ACTUAL DEVICE

ROOT AND ONE OR MORE BRANCHES (OR NO BRANCHES), CONNECTED THROUGH GEAR TRAIN

figure 11 shows the configuration

figure 12 illustrates functionality

NOT TRUE STRUCTURAL DYNAMIC COMPONENT, LACKING RIGID BODY MOTION AND FRAME

SHARES SOME STANDARD FEATURES OF STRUCTURAL DYNAMIC COMPONENT

inertia (rotational only)
structural dynamic interfaces (torque only)
applied load interfaces (torque only)
controls
sensors (component-specific only)

BUT EXCLUDING JOINTS, GRAVITATIONAL FORCES, AND AERODYNAMIC INTERFACES

STANDARD SPRING/DAMPER/ACTUATOR MODEL USED FOR ELASTIC BRANCHES

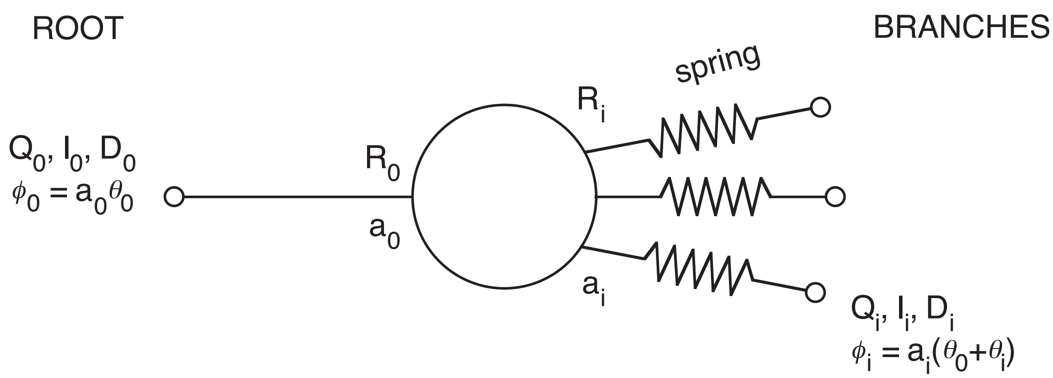


Figure 5-11 Configuration of transmission component.

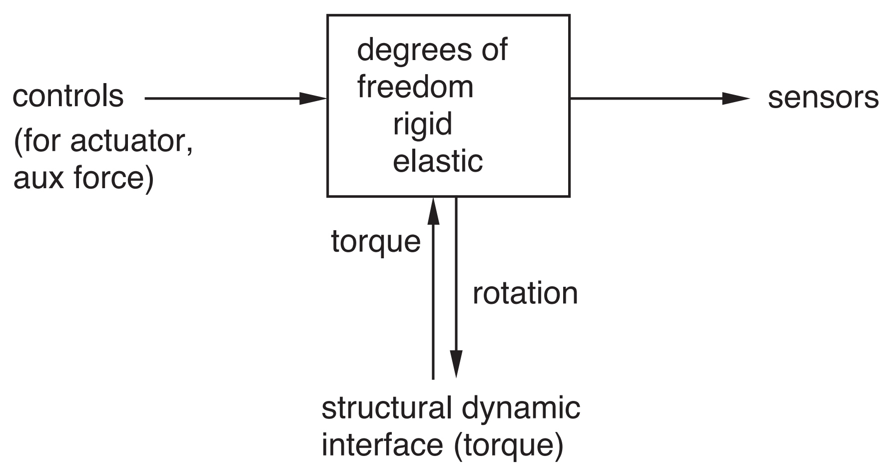


Figure 5-12 Functionality of transmission component.

COMPONENT MOTION DESCRIBED BY

- RIGID ROTATION θ_0
  - describes rotation of entire transmission, without elastic restraint
  - root motion is only rigid
- BRANCH MOTION CAN INCLUDE ELASTIC ROTATION θ_i, MEASURED RELATIVE RIGID ROTATION
- GEAR RATIOS RELATE ROTATION OF ROOT AND BRANCH TO DEGREES OF FREEDOM

ROOT AND BRANCHES

- AT END OF ROOT AND END OF EACH BRANCH, THERE IS:
  - rotation ϕ
  - torques produced by rotational inertia I and rotational damping D
  - interface torque Q (structural dynamic interface or control torque)
- VARIABLE θ_0 DESCRIBING RIGID MOTION CAN BE
  - degree of freedom
  - prescribed: constant velocity, rotating variable, or constant position
  - controlled: motion from control vector, implementing displacement actuator
    - (as for joint variables)

MOTION OF EACH BRANCH CAN BE RIGID, OR ELASTIC WITH DEGREE OF FREEDOM θ_i

equation of motion for θ_i from equilibrium of loads on branch: $f_{\text{spring}} = f_{\text{interfaces}}$

reaction f_{spring} from spring/damper/actuator model

load $f_{\text{interfaces}} = Q$ from interface load acting on branch

SENSORS

SET OF SENSOR VECTORS, TO MEASURE

motion (degrees of freedom, torque interface)
structural dynamic interface constraint (torque)
reaction (branch)
power (branch)
governor

GOVERNOR MEASURES RIGID ROTATION ERROR

θ_0 relative reference rotation

proportional (rotational speed error) and integral (rotation angle error) terms, with constant gains

5–7 Reference Frame Component

REFERENCE FRAME COMPONENT PROVIDES ACCESS TO FRAME MOTION

ALSO CAN SUM AND TRANSFORM VECTOR QUANTITIES, TYPICALLY LOADS OR MOTION

related to structural dynamic components, but does not model physical entity

figure 13 illustrates functionality

FRAMES:

COMPONENT FRAME IS REFERENCE FRAME F

parent frame P can be specified

axes T can be defined, by rotation from frame F (TF can be constant or variable)

COMPONENT FUNCTIONS:

PROVIDE ACCESS TO FRAME MOTION

motion of frame axes
velocity relative air
position and orientation relative ground
unit vectors of frame

SUM AND TRANSFORM LOAD VECTORS (FORCE AND MOMENT)

from frame F to parent frame P; or to axes T

SUM AND TRANSFORM MOTION VECTORS

sum, average, or weighted sum

from frame F to parent frame P, or to axes T
from parent frame P to frame F, or to axes T

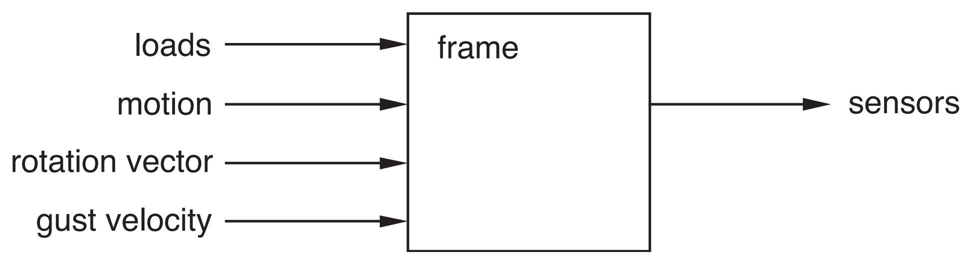


Figure 5-13 Functionality of reference frame component.

5–8 Filter Component

FILTER COMPONENT IMPLEMENTS FILTER OPERATION ON INPUT

CAN FUNCTION AS LOW-PASS FILTER, RESOLVER, OR DIFFERENTIATOR

FREQUENTLY REQUIRED TO OBTAIN MEAN VALUE OF QUANTITY

figure 14 illustrates functionality

FILTER OPERATIONS:

mean
min/max: minimum, maximum, mid value, half peak-to-peak
square: mean square, rms, amplitude, cyclic product
harmonic: cosine, sine, magnitude, phase
derivative: velocity, acceleration

mean, cosine, and sine operations are linear

all other operations are nonlinear

FILTER MODEL

OPERATION CAN BE IMPLEMENTED BY IMPLICIT MODEL, DIFFERENTIAL EQUATION MODEL, OR DIRECT MODEL

different model kinds can be used for trim, transient, and flutter tasks

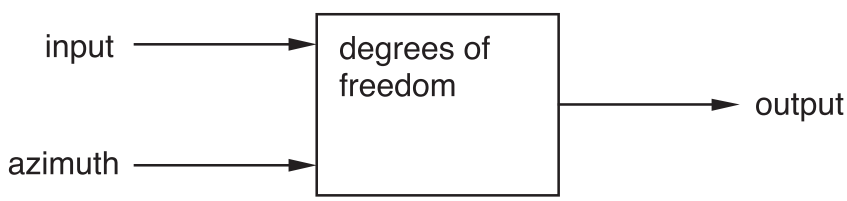


Figure 5-14 Functionality of filter component.

IMPLICIT

operates on response solution within a window

uses standard capability of analysis to apply filter when evaluating component input

can not be used for flutter task

DIFFERENTIAL EQUATION

uses first-order lag to filter the input

only available for the mean, cosine, and sine operations

DIRECT

performs no operation on the input

available for cases where filter is not required or not appropriate for all three tasks

MODEL OPTIONS DEPEND ON OPERATION AND TASK:

task	linear operations	nonlinear operations
trim	all models	implicit
transient	all models	implicit
flutter	differential equation or direct	direct

for nonlinear operations, filter operation is not applied in flutter task (and results will probably not be meaningful)

5–9 Reference Plane Component

REFERENCE PLANE COMPONENT IDENTIFIES ORIENTATION OF A PLANE

ORIENTATION DEFINED BY SMALL ROTATION FROM SOME NOMINAL PLANE

figure 15 illustrates functionality

PLANES:

NOMINAL PLANE (x–y PLANE IN T AXES) HAS FIXED ORIENTATION RELATIVE COMPONENT FRAME (F AXES)

orientation is small rotation from nominal plane (large angles would require nonlinear identification method)

IDENTIFICATION METHODS:

PLANE THROUGH SET OF POSITION VECTORS

OR HARMONICS OF NORMAL DISPLACEMENT OF POSITION VECTORS

OR HARMONICS OF SENSOR(S) THAT MEASURE NORMAL DISPLACEMENT

identified orientation can be filtered, to produce mean position

FILTER MODEL

OPERATION CAN BE IMPLEMENTED BY IMPLICIT MODEL, DIFFERENTIAL EQUATION MODEL, OR DIRECT MODEL

as for filter component

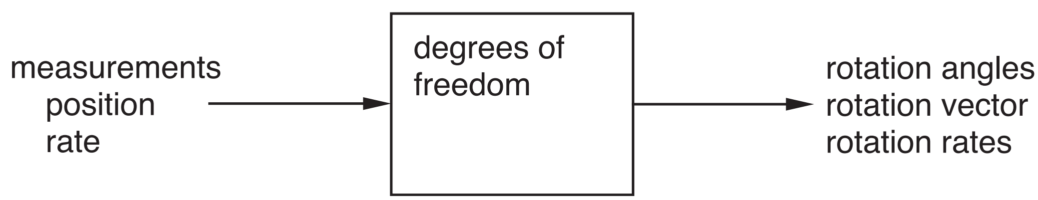


Figure 5-15 Functionality of reference plane component.

5–10 Differential Equation Component

DIFFERENTIAL EQUATION COMPONENT IMPLEMENTS STATIC, FIRST ORDER, OR SECOND ORDER DIFFERENTIAL EQUATIONS

COMPONENT CAN ALSO FUNCTION TO SIMPLY ADD SCALAR QUANTITIES

figure 16 illustrates functionality

EQUATIONS

VARIABLES

ξ = degree of freedom vector
x = component output vector
v = differential equation input vector

v is combination of component input:

can concatenate elements of input vectors

or sum, average, or weighted sum of vectors

STATIC EQUATIONS

$$x = D_s v$$

input form static, first order, or second order

optionally $D_s = I$

FIRST ORDER DIFFERENTIAL EQUATIONS

$$\dot{\xi} = A\xi + Bv$$
$$x = C\xi + Dv$$

input form first order or second order

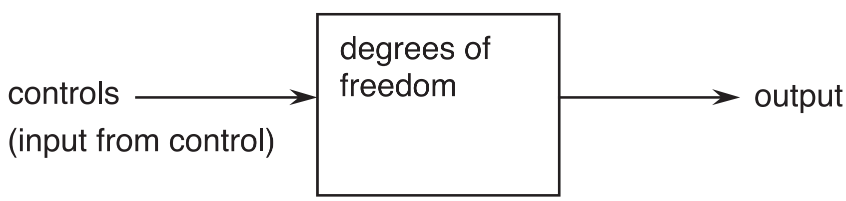


Figure 5-16 Functionality of differential equation component.

SECOND ORDER DIFFERENTIAL EQUATIONS

$$A_2\ddot{\xi} + A_1\dot{\xi} + A_0\xi = B_0 v$$
$$x = C_2\ddot{\xi} + C_1\dot{\xi} + C_0\xi + D_0 v$$

input form must be second order

MATRICES INPUT USING NAMELIST ARRAYS, OR TABLE FILE

STATIC EQUATION

STATIC EQUATION $x = D_s v$ CAN BE USED TO ADD COMPONENT INPUT, IN GENERAL WITH MATRIX OF WEIGHTS

implements scalar addition of quantities

reference frame component used for vector addition of loads and motion

DEFINING v AS CONCATENATED CONTROL VECTORS, AND D_s AS CORRESPONDING CONCATENATED MATRICES, GIVES

$$x = D_s v = \begin{bmatrix} \cdots & D_i & \cdots \end{bmatrix} \begin{pmatrix} \vdots \\ f_i \\ \vdots \end{pmatrix}$$

common application of this component

OUTPUT TRANSFORMATION

OPTIONALLY A NONLINEAR TRANSFORMATION OF THE OUTPUT VARIABLES CAN BE IMPLEMENTED

$x = F(x_{\text{lin}})$, where x_{lin} is calculated from the linear equations

transformation evaluated by linear interpolation (with no extrapolation) from set of values F_n at x_n

5–11 Programmable Component

PROGRAMMABLE COMPONENT IMPLEMENTS USER-DEFINED CALCULATIONS

component equations can be static, first order, or second order

BASIC VERSION EQUIVALENT TO DIFFERENTIAL EQUATION COMPONENT

OTHER VERSIONS CONSTRUCTED BY MODIFYING SUBROUTINES CALLED BY PROGRAMMABLE COMPONENT

figure 17 illustrates functionality

EQUATIONS

VARIABLES

ξ = degree of freedom vector
x = component output vectors
f = component input vectors

v = differential equation input vector
v is combination of component input f:

can concatenate elements of input vectors;
or sum, average, or weighted sum of vectors

STATIC EQUATIONS

$$F = D_s v$$

$$x = U_B(F, v, f, t)$$

U_B is user-defined function; $U_B = F$ for basic version

input form of matrix static, first order, or second order;
or matrices (hence F) can be ignored

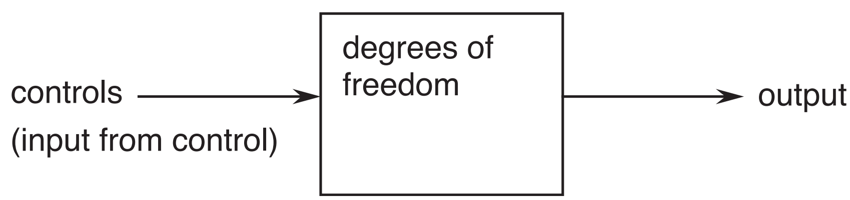


Figure 5-17 Functionality of programmable component.

FIRST ORDER EQUATIONS

$$E = A\xi + Bv - \dot{\xi}$$
$$F = C\xi + Dv$$

$$0 = U_A(E, \dot{\xi}, \xi, v, f, t)$$
$$x = U_B(F, \dot{\xi}, \xi, v, f, t)$$

U_A and U_B are user-defined function; $U_A = E$ and $U_B = F$ for basic version

input form of matrices first order or second order;
or matrices (hence E and F) can be ignored

SECOND ORDER EQUATIONS

$$E = B_0 v - (A_2\ddot{\xi} + A_1\dot{\xi} + A_0\xi)$$
$$F = C_2\ddot{\xi} + C_1\dot{\xi} + C_0\xi + D_0 v$$

$$0 = U_A(E, \ddot{\xi}, \dot{\xi}, \xi, v, f, t)$$
$$x = U_B(F, \ddot{\xi}, \dot{\xi}, \xi, v, f, t)$$

U_A and U_B are user-defined function; $U_A = E$ and $U_B = F$ for basic version

input form of matrices second order;
or matrices (hence E and F) can be ignored

MORE THAN ONE OUTPUT VECTOR CAN BE DEFINED

basic version of component gives $x = F$ for first output vector, and $x = 0$ for all others

MATRICES INPUT USING NAMELIST ARRAYS, OR TABLE FILE

if matrices are ignored, the equations of motion and output equations are entirely user-defined

USER-DEFINED CALCULATIONS

USER-DEFINED CALCULATIONS IMPLEMENTED BY MODIFYING SUBROUTINES CALLED BY PROGRAMMABLE COMPONENT

subroutine	operation
UPGMRD	read parameters
UPGMWT	write parameters
UPGMTB	tables required
UPGMIN	initialize
UPGMB0	function U_B (static)
UPGMB1	function U_B (first order)
UPGMB2	function U_B (second order)
UPGMA1	function U_A (first order)
UPGMA2	function U_A (second order)

SUBROUTINE NAMES AND COMMON NAMES BEGINNING WITH LETTER “U” ARE RESERVED FOR USER

exception: common UNITCM not available to user

INPUT PARAMETER SPECIFIES KIND OF USER-DEFINED CALCULATIONS TO BE PERFORMED

5–12 Transfer Function Component

TRANSFER FUNCTION COMPONENT IMPLEMENTS FIRST ORDER DIFFERENTIAL EQUATIONS

DEFINED BY SINGLE (SCALAR) TRANSFER FUNCTION H: $w = Hv$

figure 18 illustrates functionality

EQUATIONS

VARIABLES

transfer function input v (scalar) = weighted sum elements of component input vectors

component output vector = gain times transfer function output x (scalar)

TRANSFER FUNCTION

H = product of poles, zeros, gains, and lags

time lag modelled as one pole and one zero

poles must be unique

number of poles $\geq$ number of zeros $\geq$ 0

CONSTRUCT EQUIVALENT FIRST ORDER DIFFERENTIAL EQUATION

analysis is in time domain, not frequency domain

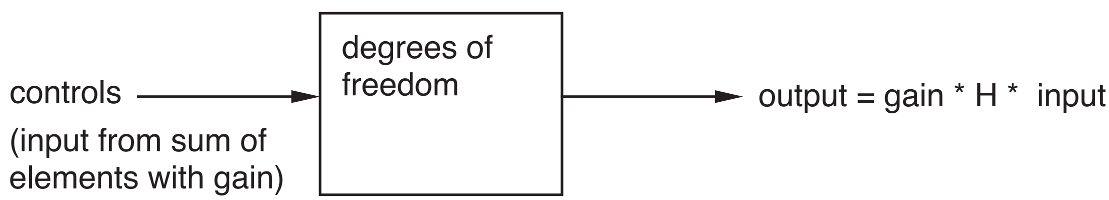


Figure 5-18 Functionality of transfer function component.

5–13 Fourier Series Component

FOURIER SERIES COMPONENT GENERATES TIME HISTORY FROM HARMONICS

figure 19 illustrates functionality

FOURIER SERIES

COMPONENT EVALUATES c IN TIME DOMAIN, FROM HARMONICS:

$$c = G\left[f_0 + \sum_{n=1}^{M} (f_{nc}\cos n\psi + f_{ns}\sin n\psi)\right]$$

G = constant gain value

c can be calculated using subset of harmonics available

COMPONENT INPUT

component input f is vector of harmonics:

$$f^T = (\, f_0 \quad \cdots \quad f_{nc} \quad f_{ns} \quad \cdots \,)$$

derivatives of f may also be available

COMPONENT OUTPUT

c is scalar, but component output can include time derivatives

AZIMUTH ANGLE ψ CAN BE A VARIABLE OR PRESCRIBED

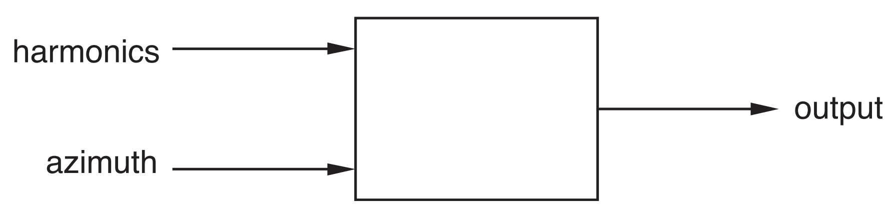


Figure 5-19 Functionality of Fourier series component.

5–14 Prescribed Control Component

PRESCRIBED CONTROL COMPONENT GENERATES TRANSIENT TIME HISTORY

FOR USE IN TRANSIENT TASK

figure 20 illustrates functionality

TIME HISTORY

COMPONENT EVALUATES c FROM PRESCRIBED FUNCTION OF TIME:

$$c = GF(t - t_B)$$

G = constant gain value

t_B = time at which transient begins

TIME HISTORY OPTIONS (FUNCTION $F(\tau)$)

step, ramp
cosine impulse, sine doublet
square impulse, square doublet
triangular impulse, triangular doublet
general piecewise linear
random (uniform, normal, or filtered)

COMPONENT OUTPUT

component output can be c (scalar), perhaps including derivatives

or component output can be vector of displacements (c times constant scale factor for each vector element)

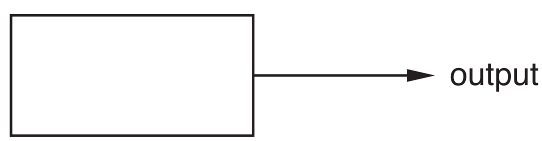


Figure 5-20 Functionality of prescribed control component.

5–15 Gust Component

GUST COMPONENT GENERATES AERODYNAMIC GUST VELOCITY

GUST VELOCITY CALCULATED AT COLLOCATION POINTS ON SYSTEM

figure 21 illustrates functionality

AERODYNAMIC GUST

GUST MODELS:

elementary (uniform, angular, or quadratic);
prescribed (function of one position coordinate);
tabular (function of 2, 3, or 4 of the 4 space/time coordinates)

COMPONENT OUTPUT

gust velocity v^I, and perhaps angular velocity ω^I (perturbation velocities of air, in inertial axes)

GUST AMPLITUDE (COMPONENT INPUT)

gust amplitude v^G (and perhaps amplitude of angular velocity or gradients)

perturbation velocities of air, in negative wind/gust axes

figure 22 shows definition and sign conventions

with aircraft conventions for wind/gust axes: u_G positive aft, v_G positive from right, w_G positive up

gust amplitude used by all models; multiplied by shape factor for prescribed or tabular model

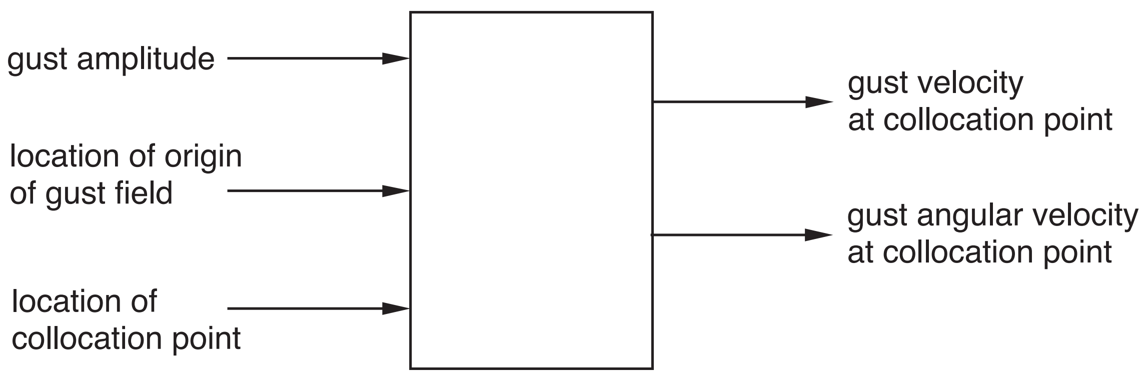


Figure 5-21 Functionality of gust component.

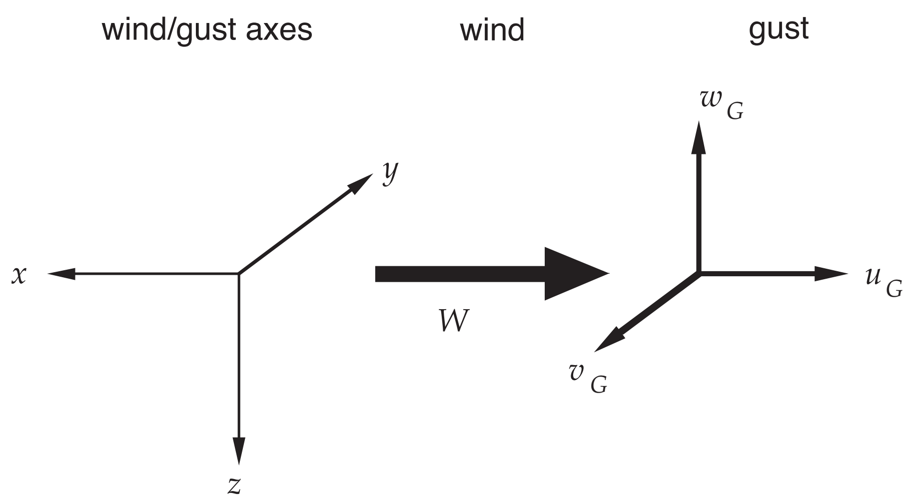


Figure 5-22 Velocity of air produced by wind and gust.

GUST FIELD

in general, velocity in gust field depends on collocation point position in wind/gust axes, and on time

measured from some origin: typically origin inertial frame, perhaps origin of system base frame (moving with system)

optionally convected with wind velocity W

CONVECTED GUST

free body moves through gust field with flight velocity V; gust field can be convected by wind velocity W

hence spatial variation of gust velocity produces time variation at system

figure 23 illustrates geometry

UNIFORM GUST MODEL

VELOCITY FROM INPUT GUST AMPLITUDE

transformed from wind/gust axes to inertial axes

gust velocity varies with time, but uniform throughout space

velocity does not depend on collocation point position

ANGULAR GUST MODEL

VELOCITY AND ANGULAR VELOCITY AT COLLOCATION POINT, FROM INPUT GUST AMPLITUDE

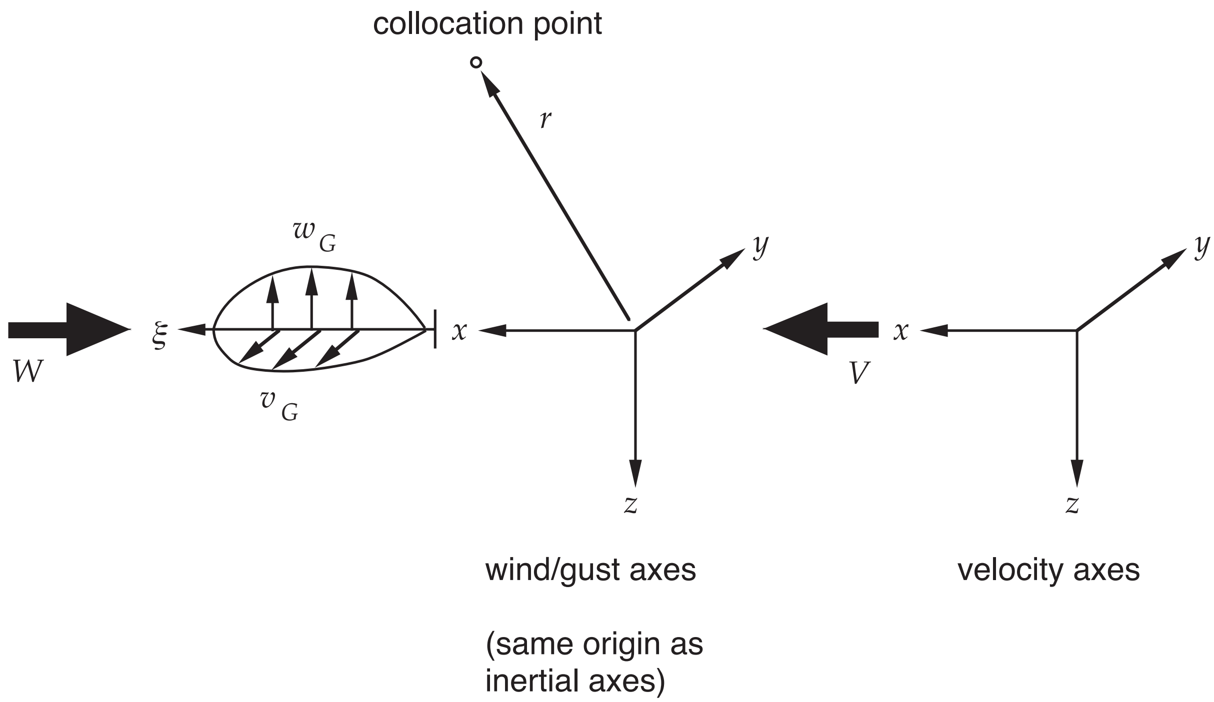


Figure 5-23 Geometry and sign conventions for convected gust.

GRADIENT GUST MODEL

VELOCITY FROM INPUT GRADIENT GUST AMPLITUDE TERMS; INCLUDING UNIFORM, LINEAR, AND QUADRATIC VARIATION IN SPACE

PRESCRIBED GUST MODEL

VELOCITY AT COLLOCATION POINT, FROM GUST AMPLITUDE TIMES SPATIAL SHAPE FACTOR

shape factor function of distance along one axis (x, y, or z-axis of wind/gust axes)

shape function options:

step, ramp
cosine impulse, sine doublet
square impulse, square doublet
triangular impulse, triangular doublet
general piecewise linear

TABULAR GUST MODEL

VELOCITY AT COLLOCATION POINT, FROM GUST AMPLITUDE TIMES SHAPE FACTOR

separate shape factor for each of three gust velocity components

shape factors depend on two, three, or four of the four space/time coordinates in wind/gust axes (x, y, z, t)

obtained from 2D or 3D or 4D table

if gust amplitude has value of 1, then table gives velocity of air relative inertial frame, in negative wind/gust axes

5–16 Aerodynamic Components

AERODYNAMIC COMPONENTS INCLUDE WINGS AND WAKES

- WING AND WAKE COMPONENTS CAN BE SEPARATE OR COMBINED, DEPENDING ON THE MODEL AND SOLUTION PROCEDURE
  - WAKE GEOMETRY AND WAKE-INDUCED VELOCITY CAN BE CALCULATED BY SEPARATE COMPONENTS
  - figure 24 illustrates typical relation between aerodynamic components

WINGS AND BODIES

- WING OR BODY IS SURFACE MOVING THROUGH AIR (WING IS THIN SURFACE)
- INTERFACES BETWEEN STRUCTURE AND AIR OCCUR AT SURFACE
  - INTERFACE IS DISCRETIZED: SET OF COLLOCATION POINTS
    - collocation points must be defined as connection points on structural dynamic components
- GENERALLY INTERFACE BETWEEN STRUCTURAL DYNAMIC AND AERODYNAMIC COMPONENTS IS IN TERMS OF VELOCITY AND FORCE (DISCRETIZED) AT COLLOCATION POINTS
  - follows from boundary conditions of aerodynamic problem

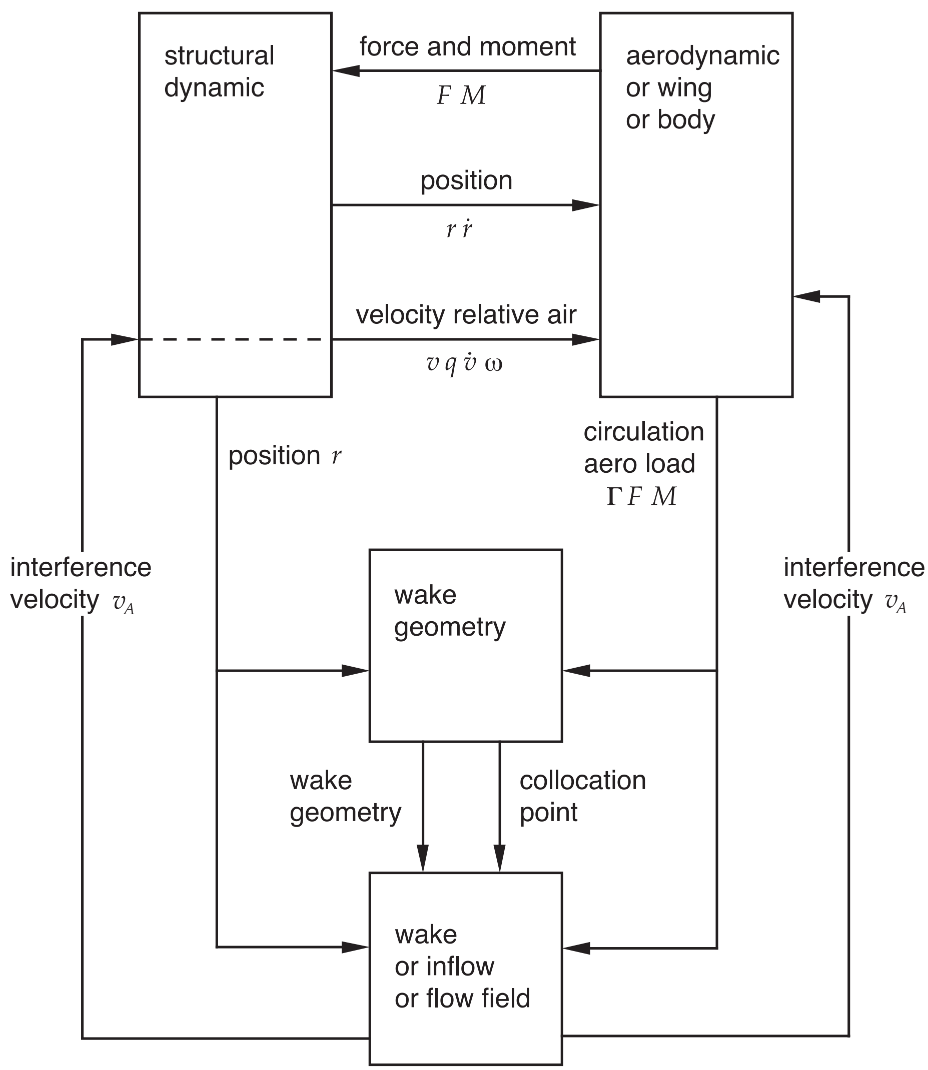


Figure 5-24 Typical relation between aerodynamic components.

INFLOW AND WAKES

WAKE COMPONENT SOLVES FOR MOTION OF AIR

in principle, there is complete mutual interaction between wings and bodies of system (calculated from wakes)

in practice, some paths of influence may be neglected

ONE WAKE PER WING

COMPONENT MAY DEAL WITH ALL WAKES IN WING SET

by convention, mutual interference between wings in a set is completely calculated

- one or more rotors is a wing set
- momentum theory deals with entire rotor

calculation of mutual interference between aerodynamic components depends on how interfaces (in terms of interference velocity) are constructed

INFLOW OR WAKE COMPONENT CALCULATES WAKE-INDUCED VELOCITY AT A COLLOCATION POINT

collocation points can be on wing generating the wake, on another aerodynamic component, or at any other point in flow field

- structural dynamic components refer to this velocity as interference velocity v_A, in the standard aerodynamic interface

in general, inertial frame is only common parent frame for wake geometry and all collocation points, so inertial axes used in wake analysis

MODEL CAN BE UNIFORM INFLOW (“INFLOW” COMPONENTS) OR NONUNIFORM INFLOW (“WAKE” COMPONENTS)

vortex wake model typically used to calculate nonuniform induced velocity

or approximate wake model based on ideal-wing theory can be used

approximate model called “uniform inflow”, although induced velocity might still have simple variation over surface of rotor disk

UNIFORM INFLOW SOLUTION GENERALLY USED TO INITIALIZE NONUNIFORM INFLOW SOLUTION, FOR BETTER CONVERGENCE

NONUNIFORM INFLOW COMPONENTS (“WAKE”)

CALCULATE INDUCED VELOCITY FROM INTEGRAL EQUATION

BASED ON DISCRETIZED VORTEX WAKE MODEL

integral equation arises from integration over the wake age, so velocity depends on loading at past times

TRIM AND TRANSIENT SOLUTION METHODS CAN EVALUATE INDUCED VELOCITY BY INTEGRATING OVER WAKE

but integral equation can not be linearized for flutter task

for flutter task, need set of differential equations equivalent to integral equation

UNIFORM INFLOW COMPONENTS (“INFLOW”)

- GLOBAL, QUASISTATIC REPRESENTATIONS OF WAKE FOR ROTOR, USING EMPIRICAL MODELS BASED ON MOMENTUM OR VORTEX THEORY
  - approximations usually require additional corrections in wing component (such as tip loss factor)
- QUASISTATIC MODEL IS DESIGNED TO BE USED WITH MEAN (FILTERED) VALUES OF ROTOR LOADING AND VELOCITY
  - so model can be used in all solution tasks, by using appropriate filters for trim, transient, and flutter models
- COMPONENT CAN ALSO INCLUDE DYNAMIC INFLOW MODEL
  - global, low frequency representation of wake
    - finite state model of unsteady aerodynamic effects of wake, relating parameters defining induced velocity and aerodynamic loading distributions on rotor, by means of ordinary differential equations
    - typically low order models (in time and space) for perturbations from trim solution, based on simplified representations of rotor aerodynamics
  - quasistatic version of dynamic inflow can be used, but frequently time lags are important

WAKE GEOMETRY

- WAKE GEOMETRY DESCRIBES POSITION OF WAKE VORTICITY IN SPACE
  - undistorted geometry obtained from motion of wing
    - wake element is convected by wind, from position in air at which it was created
  - this geometry is distorted by self-induced velocity of all wakes in system, and by wings and bodies
- DISTORTED WAKE GEOMETRY CALCULATED FOR SET OF WINGS
  - not necessarily same set as for calculation of induced velocity
    - specifying set (one or more rotors) defines what mutual interaction between wakes is accounted for in wake geometry
- WAKE GEOMETRY MODEL CAN BE RIGID, PRESCRIBED, OR FREE
  - rigid model calculates wake geometry distortion from convection by mean induced and interference velocity at wing
  - prescribed wake geometry obtained from empirical model, based on measurements
  - free wake geometry obtained by calculation
    - wake geometry model may depend solution status (wake loop level of trim task)

AERODYNAMIC SOLUTION

PARTITIONED SOLUTION PROCEDURE

solution often implemented using separate wing, wake, and wake geometry components

and calculations of single component can be performed in separate steps

so different wing sets can be used, depending on models available and mutual interaction accounted for

and different solution procedures and time steps can be used

TYPICALLY:

WAKE GEOMETRY COMPONENTS OBTAIN WAKE POSITION FROM MOTION AND LOADING

THEN WAKE COMPONENTS CALCULATE INFLUENCE COEFFICIENTS FROM WAKE GEOMETRY, AND THEN INDUCED VELOCITY FROM INFLUENCE COEFFICIENTS AND LOADING

THEN WING COMPONENTS OBTAIN LOADING FROM INDUCED VELOCITY

TRIM TASK CAN USE PARTITIONED SOLUTION PROCEDURE

parts solve equations and loops iterate between part solutions

typically have wake geometry loop and circulation loop

WAKE GEOMETRY LOOP

FOR EFFICIENCY, MOVE COMPUTATIONALLY INTENSIVE CALCULATIONS OUTSIDE INNER LOOPS (IF ALLOWED BY WEAK COUPLING)

CALCULATION OF WAKE GEOMETRY AND INFLUENCE COEFFICIENTS CAN BE MOVED OUTSIDE THE TRIM ITERATION

possible when coupling between wake geometry and rest of solution is relatively weak

hence when rotor is trimmed to specified speed, orientation, and lift

WAKE GEOMETRY LOOP DIVIDES SYSTEM INTO CALCULATION OF WAKE GEOMETRY AND INFLUENCE COEFFICIENTS, AND SOLUTION OF REST OF EQUATIONS

successive substitution iteration, with stages or levels:

level 1 is uniform inflow

level 2 is nonuniform inflow with rigid or prescribed wake geometry

level 3 is nonuniform inflow with free wake geometry

for better convergence, uniform inflow used to initialize nonuniform inflow, and rigid wake geometry used to initialize free wake geometry

whether results of level 3 or level 2 are required depends on importance of free wake geometry and nonuniform inflow to the problem

AT EACH LEVEL, ITERATION IS NECESSARY IF ANYTHING IS BEING CALCULATED BY INNER LOOPS THAT WILL CHANGE WAKE GEOMETRY

- such as speed; mean inflow; rotor lift or circulation; or inboard circulation peak locations
  - with trim solution inside wake geometry loop, may not be necessary to iterate on wake geometry at each level
- if rotor is at fixed collective (rather than trimmed to specified lift), then overall geometry of wake not known in advance, only after rotor loading calculated
  - so if wake geometry significantly influences rotor loading, iteration required in wake geometry loop
    - and need relaxation factor on wake geometry

CIRCULATION LOOP

ITERATION BETWEEN WAKE OR INFLOW COMPONENTS (WHICH CALCULATE INDUCED VELOCITY FROM LOADING)

AND COMPONENTS OF REST OF SYSTEM (WHICH CALCULATE THE LOADING FOR FIXED INDUCED VELOCITY)

so can solve wake components (integral equations) and structural dynamic components (differential equations) by appropriate methods

and to reduce number of variables being solved in part

SUCCESSIVE SUBSTITUTION ITERATION

relaxation factor of $\lambda = 0.10$ or 0.05 often required for nonuniform inflow models

FOR HOVERING ROTOR NEAR ZERO THRUST, CIRCULATION ITERATION DOES NOT CONVERGE

modified inflow calculation required to achieve convergence

5–17 Rigid Airframe Aerodynamics Component

RIGID AIRFRAME AERODYNAMICS COMPONENT CALCULATES AERODYNAMIC FORCES AND MOMENTS ACTING ON A RIGID BODY

- CONSISTING OF WING-BODY, HORIZONTAL TAIL, AND VERTICAL TAIL
- THE "RIGID BODY" IS TYPICALLY A SINGLE STRUCTURAL DYNAMIC COMPONENT
  - figure 25 illustrates functionality

AERODYNAMIC MODEL

- QUASISTATIC MODEL, USING SMALL NUMBER OF COLLOCATION POINTS
  - intended for static and low frequency motion of airframe
  - designed to be used with mean (filtered) values of component input

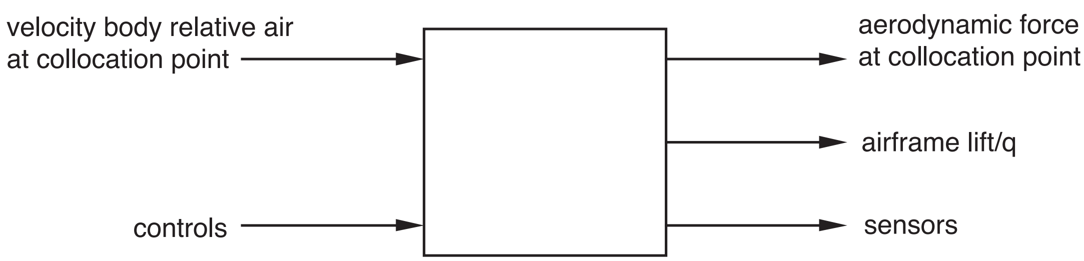


Figure 5-25 Functionality of rigid airframe aerodynamics component.

ELASTIC MOTION INVOLVES LOCALIZED, HIGH FREQUENCY OSCILLATIONS

wake model and noncirculatory loads needed for high frequency aerodynamics are not included

component model not consistent with elastic motion

so velocity at collocation points should be calculated without elastic motion, and aerodynamic loads should not be included in elastic equations of motion of airframe

THREE OR FOUR COLLOCATION POINTS ON THE AIRFRAME

wing-body, horizontal tail, and vertical tail
(and perhaps a collocation point for stability derivatives)

velocity relative air required at collocation points

collocation point is point-of-action of corresponding force and moment

velocities, angular motion, and loads are in body axes of airframe

AERODYNAMIC PROPERTIES

- aerodynamic properties of airframe defined in velocity axes
  - airframe angle of attack α and sideslip angle β define orientation of velocity axes relative body axes
    - figure 26 shows conventions for velocity axis forces
  - angle of attack $+$ for linear velocity in z-axis direction
  - sideslip $+$ for linear velocity in y-axis direction
  - yaw $+$ for rotation about z-axis
- horizontal tail can have cant angle ϕ_{HT} ($+$ to left, so at 90 degrees equivalent to vertical tail)
- vertical tail can have cant angle ϕ_{VT} ($+$ to right, so at 90 degrees equivalent to horizontal tail)

AIRFRAME AERODYNAMIC CONTROLS

- flaperon δ_f (positive for down deflection, increasing wing-body lift)
- elevator δ_e (positive for down deflection, increasing horizontal tail lift)
- aileron δ_a (positive for down deflection on right wing, causing a negative roll moment)
- rudder δ_r (positive for down deflection, increasing vertical tail lift)

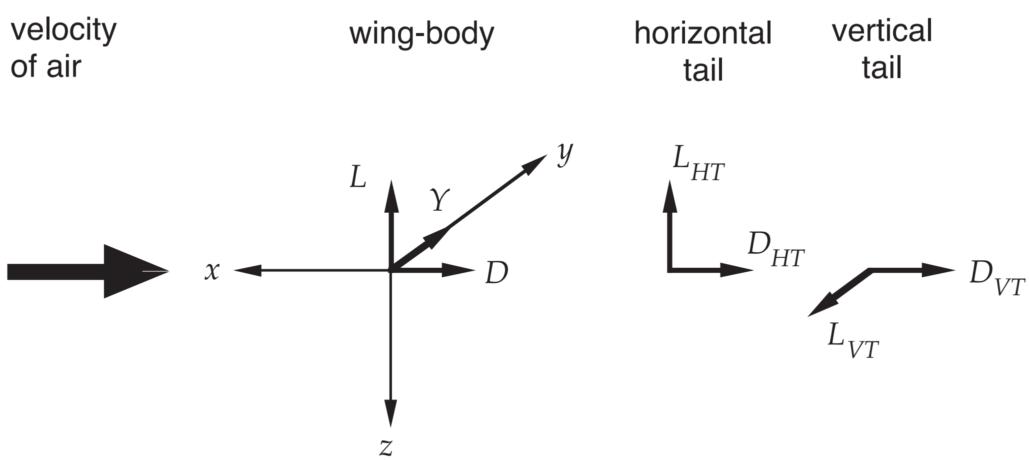


Figure 5-26 Velocity axis loads.

AERODYNAMIC LOAD CAN BE CALCULATED USING NONLINEAR MODEL OR STABILITY DERIVATIVES (LINEARIZED) MODEL

TRIM TASK USES NONLINEAR MODEL

- nonlinear loads calculated from simple equations and/or tables
  - can also be used in transient and flutter tasks
- separate properties can be used for wing-body, horizontal tail, and vertical tail
  - so can have separate aerodynamic interference effects at three collocation points
- or tail loads can be included in wing-body properties

LINEARIZED MODEL EVALUATES LOADS FROM STABILITY DERIVATIVES

- can only be used in transient and flutter tasks
- perturbation (total minus trim) loads calculated from perturbation velocity
- conventional aircraft model, useful for tiltrotor configuration

NONLINEAR MODEL

VELOCITY AXIS LOADS ACTING ON WING-BODY (WB), HORIZONTAL TAIL (HT), AND VERTICAL TAIL (VT)

wing-body drag, side, and lift forces: D, Y, and L (along x, y, and z axes respectively, acting at wing-body collocation point)

wing-body roll, pitch, and yaw moments: M_x, M_y, and M_z (about x, y, and z axes respectively, with origin at wing-body collocation point)

horizontal tail drag and lift forces: D and L

vertical tail drag and lift forces: D and L

loads defined in terms of force or moment divided by dynamic pressure q

TEN LOADS (6 WB, 2 HT, 2 VT), EACH OBTAINED FROM ONE OF FOLLOWING SOURCES

equations

two-dimensional table

function of α or β, and Mach number M

three-dimensional or four-dimensional tables

function of α, β, M, flaperon, elevator, aileron, rudder, or flap (any combination of three or four independent variables)

MODEL INCLUDES EFFECTS OF AERODYNAMIC INTERFERENCE FROM WING-BODY ON THE TAIL

tail angle of attack change ϵ from wing body lift, including time lag from $\dot{\alpha}_{WB}$

tail sideslip change σ from wing-body roll rate

EQUATIONS

FOR WING BODY:

$$\begin{pmatrix} M_y/q \\ D/q \\ L/q \end{pmatrix} = \begin{pmatrix} M_0/q + (M_\alpha/q)(\alpha_{WB} + i_{WB_M}) \\ f_{WB} + (g_{WB} - f_{WB})\sin^2(\alpha_{WB} + i_{WB_D}) + D_I(L/q)^2 \\ L_0/q + (L_\alpha/q)(\alpha_{WB} + i_{WB_L}) \end{pmatrix} + \begin{pmatrix} M_{\delta_e}/q \\ D_{\delta_e}/q \\ L_{\delta_e}/q \end{pmatrix}\delta_e + \begin{bmatrix} M_{\delta_f}/q & M_{\delta_F}/q \\ D_{\delta_f}/q & D_{\delta_F}/q \\ L_{\delta_f}/q & L_{\delta_F}/q \end{bmatrix}\begin{pmatrix} \delta_f \\ \delta_F \end{pmatrix}$$

$$\begin{pmatrix} M_x/q \\ M_z/q \\ Y/q \end{pmatrix} = \begin{pmatrix} N_{x0}/q + (N_{x_\beta}/q)\beta_{WB} \\ N_{z0}/q + (N_{z_\beta}/q)\beta_{WB} \\ Y_0/q + (Y_\beta/q)\beta_{WB} \end{pmatrix} + \begin{pmatrix} N_{x_{\delta_r}}/q \\ N_{z_{\delta_r}}/q \\ Y_{\delta_r}/q \end{pmatrix}\delta_r + \begin{bmatrix} VN_{x_p}/q & VN_{x_r}/q & N_{x_{\delta_a}}/q \\ VN_{z_p}/q & VN_{z_r}/q & N_{z_{\delta_a}}/q \\ VY_p/q & VY_r/q & Y_{\delta_a}/q \end{bmatrix}\begin{pmatrix} p/V \\ r/V \\ \delta_a \end{pmatrix}$$

elevator and rudder terms in wing-body loads should only be used if tail loads are not present

last two terms in D/q both give α^2 dependence

second term represents stall drag

third term is induced drag

EXPECTED SIGNS OF DERIVATIVES: Y_β, N_{z_β}, $N_{x_{\delta_a}}$ negative

FOR HORIZONTAL TAIL:

$$\begin{pmatrix} D/q \\ L/q \end{pmatrix} = \begin{pmatrix} f_{HT} + (g_{HT} - f_{HT}) \sin^2(\alpha_{HT} + i_{HT_D}) \\ L_0/q + (L_\alpha/q)(\alpha_{HT} + i_{HT_L}) \end{pmatrix} + \begin{pmatrix} D_{\delta_e}/q \\ L_{\delta_e}/q \end{pmatrix} \delta_e$$

FOR VERTICAL TAIL:

$$\begin{pmatrix} D/q \\ L/q \end{pmatrix} = \begin{pmatrix} f_{VT} + (g_{VT} - f_{VT}) \sin^2(\alpha_{VT} + i_{VT_D}) \\ L_0/q + (L_\alpha/q)(\alpha_{VT} + i_{VT_L}) \end{pmatrix} + \begin{pmatrix} D_{\delta_r}/q \\ L_{\delta_r}/q \end{pmatrix} \delta_r$$

incidence angles i measured relative body axes, such that zero load obtained for $\alpha = -i$

lift and moment stall modelled by using truncated angle of attack

$$\alpha_e = \text{sign}(\alpha)\ \min(|\alpha|, \alpha_{\max})$$

TWO-DIMENSIONAL TABLE

equations still needed for rate, flaperon, aileron, and flap terms; and for elevator and rudder terms

FOR WING-BODY:

$$\begin{pmatrix} M_y/q \\ D/q \\ L/q \end{pmatrix} = \begin{pmatrix} ScC_M \\ SC_D \\ SC_L \end{pmatrix} + \begin{pmatrix} M_{\delta_e}/q \\ D_{\delta_e}/q \\ L_{\delta_e}/q \end{pmatrix} \delta_e + \begin{bmatrix} M_{\delta_f}/q & M_{\delta_F}/q \\ D_{\delta_f}/q & D_{\delta_F}/q \\ L_{\delta_f}/q & L_{\delta_F}/q \end{bmatrix} \begin{pmatrix} \delta_f \\ \delta_F \end{pmatrix}$$

$$\begin{pmatrix} M_x/q \\ M_z/q \\ Y/q \end{pmatrix} = \begin{pmatrix} SbC_\ell \\ SbC_n \\ SC_y \end{pmatrix} + \begin{pmatrix} N_{x\delta_r}/q \\ N_{z\delta_r}/q \\ Y_{\delta_r}/q \end{pmatrix} \delta_r + \begin{bmatrix} VN_{x_p}/q & VN_{x_r}/q & N_{x\delta_a}/q \\ VN_{z_p}/q & VN_{z_r}/q & N_{z\delta_a}/q \\ VY_p/q & VY_r/q & Y_{\delta_a}/q \end{bmatrix} \begin{pmatrix} p/V \\ r/V \\ \delta_a \end{pmatrix}$$

elevator and rudder terms in wing-body loads should only be used if tail loads are not present

C_M, C_D, and C_L function of α_{WB} and Mach number

C_ℓ, C_n, and C_y function of β_{WB} and Mach number

coefficients based on wing-body area S, chord c, span b

FOR HORIZONTAL TAIL:

$$\begin{pmatrix} D/q \\ L/q \end{pmatrix} = \begin{pmatrix} SC_D \\ SC_L \end{pmatrix} + \begin{pmatrix} D_{\delta_e}/q \\ L_{\delta_e}/q \end{pmatrix} \delta_e$$

FOR VERTICAL TAIL:

$$\begin{pmatrix} D/q \\ L/q \end{pmatrix} = \begin{pmatrix} SC_D \\ SC_L \end{pmatrix} + \begin{pmatrix} D_{\delta_r}/q \\ L_{\delta_r}/q \end{pmatrix} \delta_r$$

C_D and C_L function of α_{HT} or α_{VT}, and Mach number

coefficients based horizontal or vertical tail area S

THREE-DIMENSIONAL OR FOUR-DIMENSIONAL TABLES

coefficients function of α, β, M, δ_f, δ_e, δ_a, δ_r, δ_F (any combination of three or four independent variables)

equations still needed for rate terms, and perhaps for the control and flap terms

> control effects may be in equations or tables, but should not be in both

loads are calculated using same equations as for two-dimensional table

FOR WING-BODY:

> elevator and rudder terms should only be used if tail loads are not present

FOR HORIZONTAL AND VERTICAL TAILS:

> sideslip angle, flaperon deflection, aileron deflection, and flap angle ignored

two-dimensional table: coefficients must all be functions of same independent variables (angle and Mach number)

three-dimensional or four-dimensional tables: different independent variables can be defined for each coefficient

> if only one value in table for third independent variable, 3D table is effectively 2D

5–18 Airframe Flow Field Component

AIRFRAME FLOW FIELD COMPONENT CALCULATES PERTURBATION AERODYNAMIC INTERFERENCE VELOCITY PRODUCED BY AIRFRAME (OR OTHER OBJECT)

- figure 27 illustrates functionality

AERODYNAMIC MODEL

- QUASISTATIC MODEL
  - designed to be used with mean (filtered) values of component input
    - shed wake and noncirculatory loads needed for high frequency aerodynamics are not included
  - by using appropriate frame for component, location and velocity of wings and bodies can be filtered to eliminate high frequency oscillations
  - then flow field is quasistatic relative to component frame
    - location of collocation point relative wings and bodies might still be time varying, so should not be filtered
- INTERFERENCE SOURCES:
  - velocity calculated
  - velocity from table file

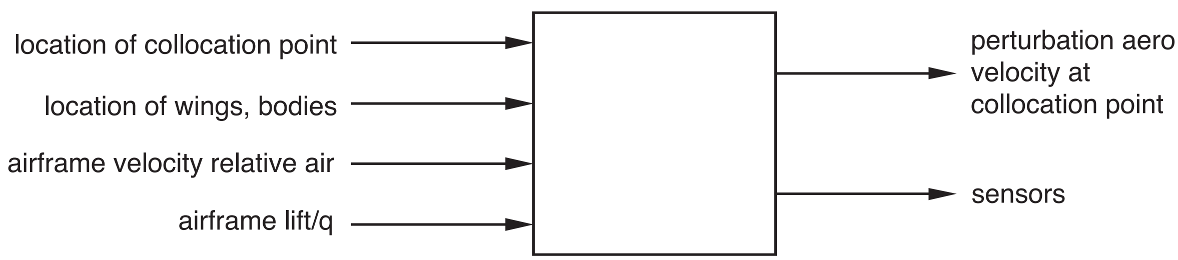


Figure 5-27 Functionality of airframe flow field component.

CALCULATED INTERFERENCE VELOCITY

PRODUCED BY SIMPLE REPRESENTATION OF AIRFRAME: SET OF WINGS AND BODIES

wing modelled as constant strength horseshoe vortex (lift) and constant strength doublet line (thickness)

body modelled by potential flow (no separation) about nonlifting body-of-revolution at arbitrary angle of attack

velocities calculated at actual locations of collocation points relative airframe

approximate model, requiring calibration for reasonable results near the wings and bodies

figure 28 shows the geometry

WINGS

wing defined by quarter-chord, in two straight segments

input = locations of left tip, middle, and right tip (at quarter chord)

wing not necessarily horizontal or symmetric

right-hand rule for vector from “left” tip to “right” tip defines positive direction of bound circulation (hence wing upper surface)

circulation and thickness lines placed distance xc_w behind leading edge (to allow calibration of velocity field near wing)

for far field velocity, circulation line should be at 1/4 chord, and thickness line should be at 3/8 chord

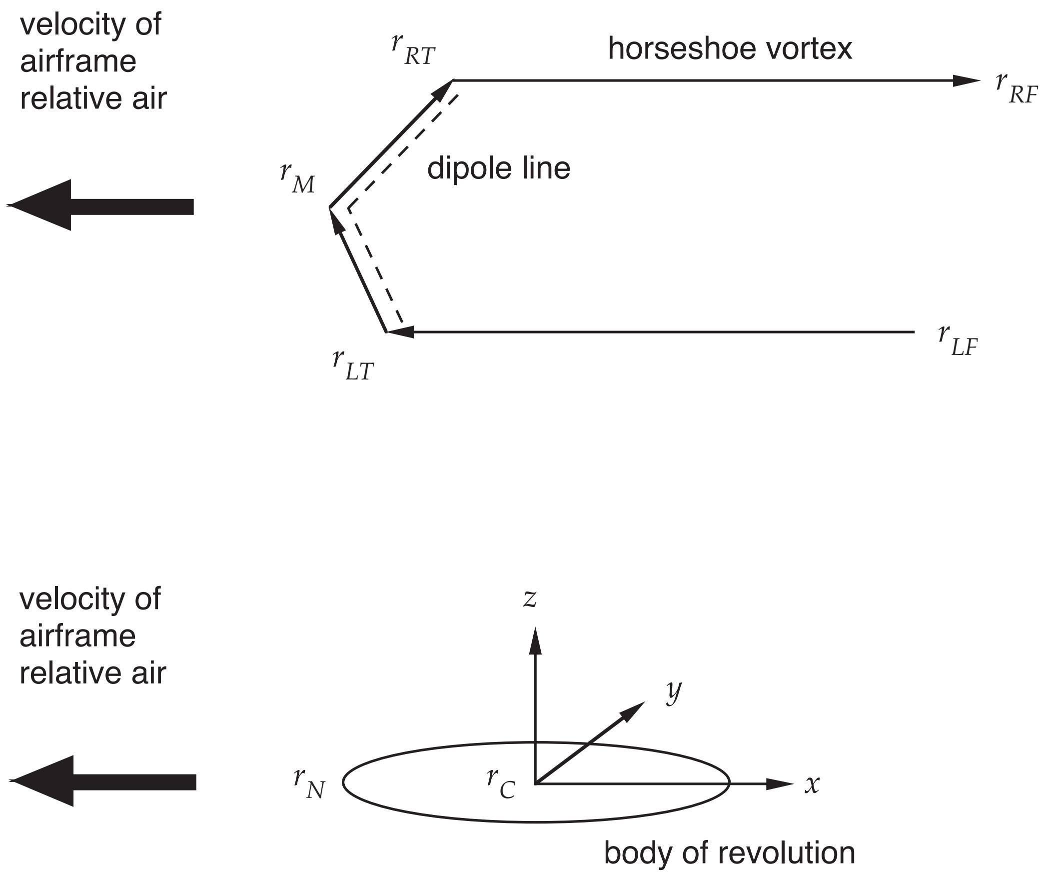


Figure 5-28 Geometry of wings and bodies.

wing circulation strength is:

$$\frac{\Gamma}{V} = \frac{L/q}{2b_w} = \frac{c_w C_L}{2}$$

evaluated from combination of input value, and the wing-body, horizontal tail, and vertical tail lifts:

$$\frac{\Gamma}{V} = \frac{\Gamma_0}{V} + \frac{1}{2b_w}\left[f_W(L/q)_{WB} + f_H(L/q)_{HT} + f_V(L/q)_{VT}\right]$$

with factors f_W, f_H, f_V to calibrate model

dipole strength obtained from wing airfoil cross section area A_{XS} (approximately $0.68\tau_w c_w^2$)

BODIES

body defined by axis, length, and thickness ratio

input = locations center and nose (on axis)

figure 28 shows the geometry

shapes:

ovary (thin) ellipsoid, created by revolving ellipse about its major axis (exact solution)

planetary (flat) ellipsoid, created by revolving ellipse about its minor axis (exact solution)

sphere (exact solution)

airfoil-shaped body, created by revolving NACA 4-digit airfoil thickness distribution about chord line (modified slender-body theory solution)

VELOCITY TABLE FILE

PROVIDES ACCESS TO EXTERNALLY CALCULATED INTERFERENCE

typically obtained using panel method with detailed representation of airframe geometry

velocities calculated for specified locations of collocation points relative airframe

TWO-DIMENSIONAL TABLE

dependent variables are three components of interference velocity $\Delta v^F/V$, in component frame F

first independent variable = span station value

- interpolated to span station of collocation points

second independent variable = time or azimuth

- interpolated (cyclically for azimuth) to current time value
- with only one time value, table is function of span station only

5–19 Lifting Line Wing Component

LIFTING LINE WING COMPONENT CALCULATES AERODYNAMIC FORCES ON A WING

- AERODYNAMIC MODEL IS LIFTING-LINE THEORY, USING STEADY TWO-DIMENSIONAL AIRFOIL CHARACTERISTICS AND A VORTEX WAKE
- WAKE MODEL IMPLEMENTED IN SEPARATE COMPONENT
  - wing component has as input the wake-induced interference velocity (and any other interference terms in velocity of air)
    - figure 29 illustrates functionality

AERODYNAMIC MODEL

- LIFTING-LINE THEORY ASSUMES THAT WING HAS HIGH ASPECT RATIO
  - more generally, assume small spanwise variations of aerodynamic environment
    - assumption allows problem to be split into separate wing and wake models
    - solved individually and combined
  - assumptions of lifting-line theory generally well satisfied for rotor blades

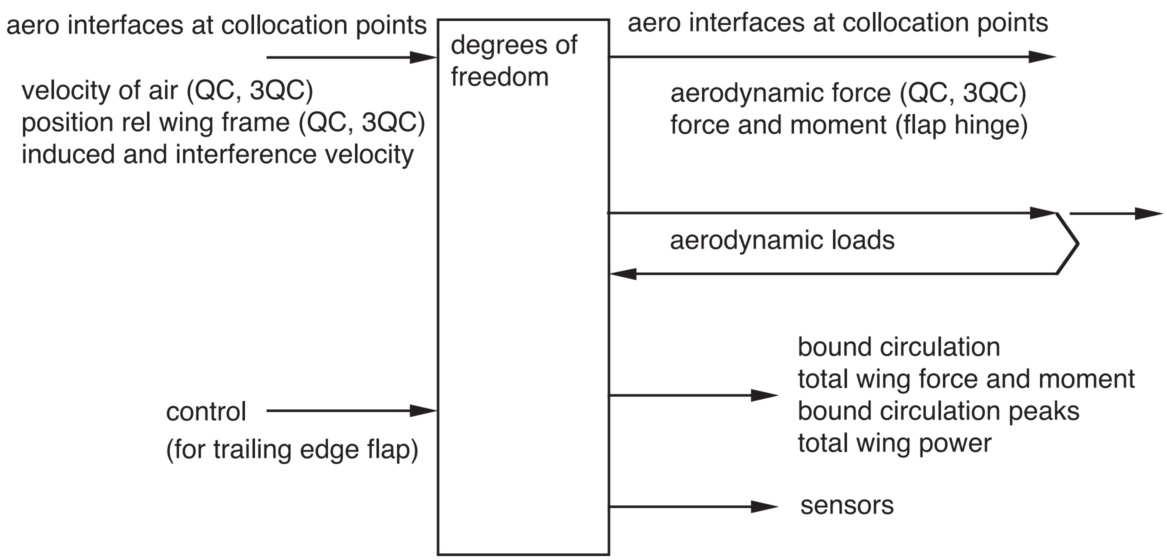


Figure 5-29 Functionality of lifting line wing component.

this component solves wing problem of lifting-line theory

- viscous and compressibility effects included by using experimental data for two-dimensional airfoil characteristics

other components solve wake problem of lifting-line theory

- vortex wake model used to calculate nonuniform induced velocity on wing
- or approximate wake model based on ideal-wing theory can be used (“uniform inflow”)
  - with uniform inflow, three-dimensional flow effects at wing tips are lost, requiring additional corrections in wing model (tip loss factor)

MODEL FEATURES:

- stall models (static, empirical dynamic stall model, none)
- unsteady lift and moment for attached flow
- yawed and swept flow corrections, and spanwise drag
- drag and moment increments, zero lift angle increment
- static stall delay
- Reynolds number corrections
- trailing edge flaps
- prescribed lift, drag, and moment increments from an external aeroacoustic analysis

WING IS SURFACE, DESCRIBED BY SET OF QUARTER CHORD AND THREE-QUARTER CHORD COLLOCATION POINTS ALONG SPAN

- lifting-line theory deals with section aerodynamic environment
  - required section geometry (bent-chord axes) obtained from geometry of collocation points
  - arbitrary geometry, including droop and sweep of quarter-chord line, and twist
- geometry actually calculated on structural dynamic side of interfaces with wing component
  - component interfaces are velocity and force normal to wing surface (discretized)
- component analyzes the entire wing

figure 30 shows relation between geometry of wing and geometry of structure

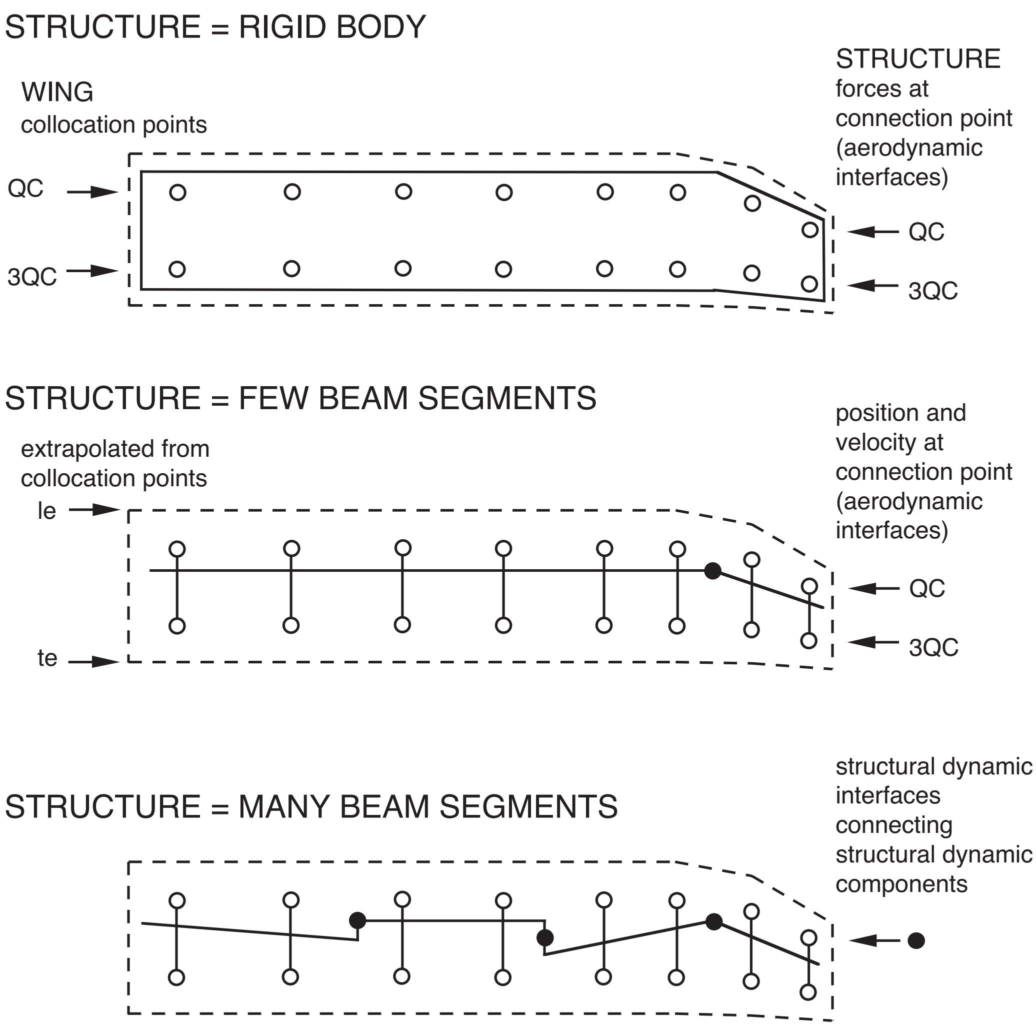


Figure 5-30 Relation between wing geometry and structure geometry.

BASIC LIFTING-LINE THEORY

- THREE-DIMENSIONAL WING AERODYNAMIC PROBLEM IS UNSTEADY, COMPRESSIBLE, VISCOUS
- ASSUMPTION OF HIGH ASPECT-RATIO SPLITS IT INTO INNER AND OUTER PROBLEMS — INTO WING AND WAKE MODELS
    - figure 31 illustrates approach
- OUTER PROBLEM IS WAKE
    - trailed and shed vorticity behind lifting-line (bound vortex)
- INNER PROBLEM IS TWO-DIMENSIONAL AIRFOIL
    - infinite wing in uniform, yawed free stream
- CONNECTED THROUGH THE WAKE-INDUCED VELOCITY AND THE BOUND CIRCULATION
    - outer problem calculates induced velocity at wing, from wake with strength determined by bound circulation
        - induced velocity not needed at arbitrary point, just at lifting-line
    - inner problem calculates bound circulation from aerodynamic environment, with wake-induced velocity included in free stream
        - pressure on wing not needed, just bound circulation (and section lift, drag, and moment in order to calculate performance and couple with structural dynamics)
- UNIFORM INFLOW FROM IDEAL-WING (MOMENTUM OR VORTEX) THEORY IS APPROXIMATION FOR SOLUTION OF OUTER PROBLEM (WAKE)

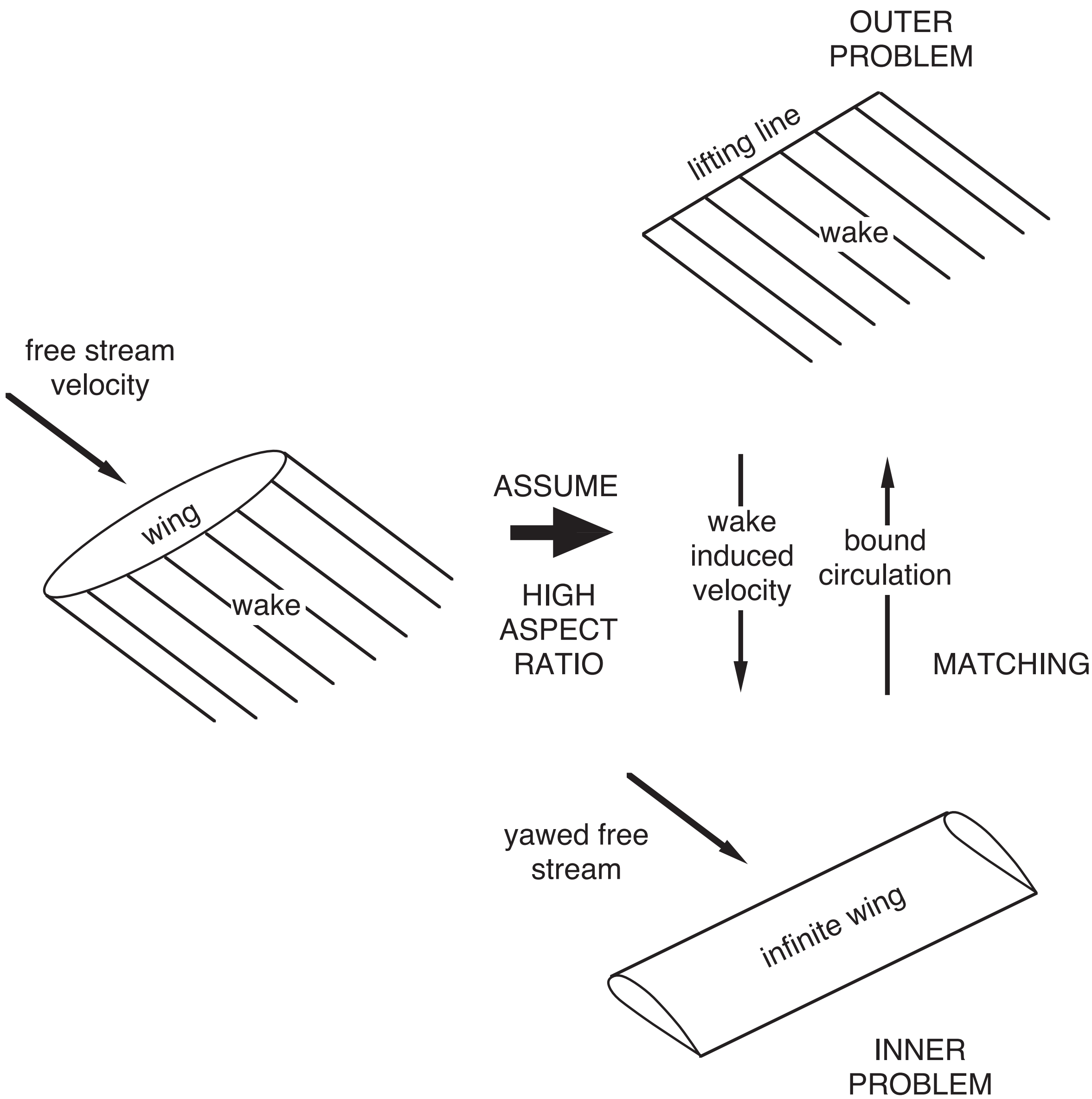


Figure 5-31 Lifting line theory.

PERTURBATION THEORY

FORMAL LIFTING-LINE THEORY IS SOLUTION FOR THREE-DIMENSIONAL WING LOADING USING METHOD OF MATCHED ASYMPTOTIC EXPANSIONS

lowest-order fixed wing solution is Prandtl's theory (steady, no sweep)

developments found in literature include higher orders, unsteady, transonic, and swept flow

higher order theories from Weissinger (intuitive) and van Dyke (perturbation theory)

these theories generally obtain analytical solutions for both inner and outer problems, and are in quadrature rather than integral form

often inner solution is inviscid, or even a thin airfoil

PERTURBATION THEORY PROVIDES GUIDE TO DEVELOPMENT OF PRACTICAL APPROACH

for rotary wing, must include stall (high angles of attack) in inner solution, and include distorted and rolled-up wake geometry in outer solution

WANT LIFTING-LINE THEORY WITH

numerical solutions for inner and outer problems,
iterative method based on matching procedure

must retain two-dimensional airfoil tables in inner solution, for viscous effects

accept whatever approximations are required to retain airfoil tables

SECOND-ORDER LIFTING-LINE THEORY

- second-order lifting-line theory gives nearly same results as lifting-surface theory, including lift produced in close wing-vortex interactions
    - second-order theory also improves loads calculations for swept tips, yawed flow, and low aspect-ratio wings
- second-order outer solution
    - wing is dipole line + quadrupole line, equivalent to dipole at quarter chord
    - dipole solution is wake of vortex sheets
- second-order inner solution
    - boundary condition is wake-induced velocity varying linearly in space
    - get same lift with uniform induced velocity, by using value at three-quarter chord
        - moment error, since linear induced-velocity variation over chord produces moment about quarter chord, but uniform induced-velocity does not
        - moment still first order
- RETAINING AIRFOIL TABLES, SECOND-ORDER THEORY ONLY MEANS PLACING LIFTING-LINE AT QUARTER CHORD AND COLLOCATION POINT AT THREE-QUARTER CHORD
    - first-order theory has collocation point at quarter chord

PRACTICAL IMPLEMENTATION OF LIFTING-LINE THEORY

OUTER PROBLEM IS INCOMPRESSIBLE VORTEX WAKE BEHIND LIFTING-LINE, WITH DISTORTED GEOMETRY AND ROLLUP

- lifting-line (bound vortex) at quarter chord
  - approximation for quadrupole line of second-order loading
- trailed wake begins at bound vortex
- shed wake created quarter chord aft of collocation point on the wing
  - lifting-line approximation for unsteady loading
- three components of wake-induced velocity evaluated at collocation points, excluding contributions of bound vortex
- collocation points at three-quarter chord (in direction of local flow)
  - approximation for linearly varying induced velocity introduced by second-order wake
- induced velocity at three-quarter chord used only to calculate angle of attack for loading solution
  - local section aerodynamic environment, including orientation of lift and drag and hence magnitude of induced power, still obtained from induced velocity at quarter chord

INNER PROBLEM IS UNSTEADY, COMPRESSIBLE, VISCOUS FLOW ABOUT INFINITE ASPECT-RATIO WING, IN UNIFORM FLOW CONSISTING OF YAWED FREE STREAM AND THREE COMPONENTS OF WAKE-INDUCED VELOCITY

- split into parts
  - two-dimensional, steady, compressible, viscous flow = airfoil tables
  - plus corrections
- corrections account for
  - swept and yawed flow
  - tip flow
  - Reynolds number
  - unsteady flow: small angle-of-attack noncirculatory loads, without any shed wake
  - dynamic stall: empirical model

GENERALLY SECOND-ORDER ACCURATE FOR LIFT

- including effects of sweep and yaw
- with typical blade-vortex separations, as accurate as lifting-surface theory for vortex-induced lift calculations

BUT LESS ACCURATE FOR SECTION MOMENTS

YAWED AND SWEPT FLOW CORRECTIONS

INNER PROBLEM IS INFINITE WING WITH YAW AND SWEEP

planform defined relative to wing reference line, so spanwise velocity produces yawed flow, while sweep obtained from locus of quarter chord relative to wing reference line

figure 32 shows geometry

SECTION LOADS OBTAINED FROM AIRFOIL TABLES (SOLUTION FOR WING WITH NO YAW OR SWEEP), USING EQUIVALENCE ASSUMPTION FOR SWEPT WINGS

must derive equivalent angle of attack and Mach number for evaluating coefficients, and corrections for coefficients from tables

and correctly account for chord and wing area when multiplying coefficients by chord, dynamic pressure, and panel width to obtain section loads

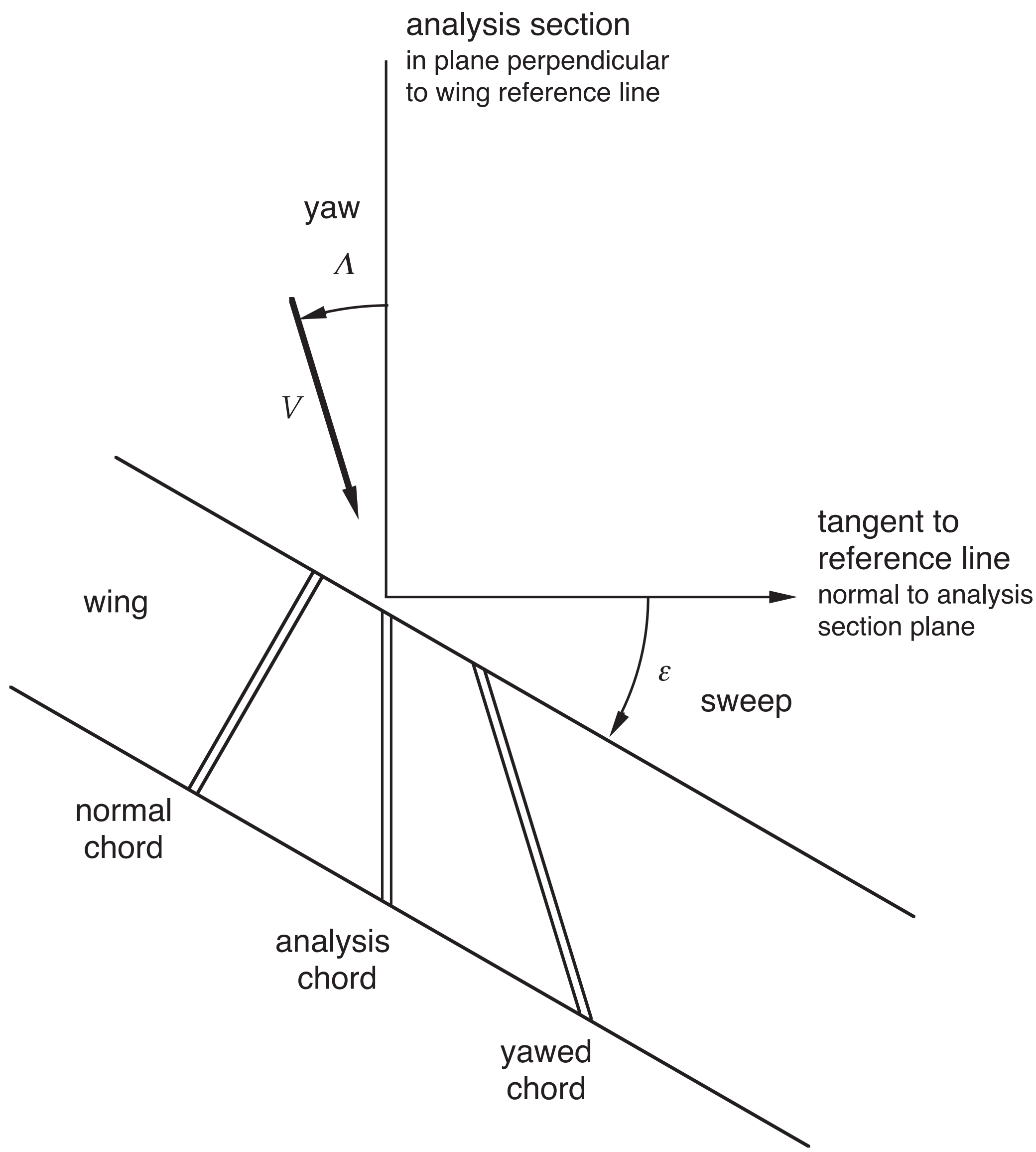


Figure 5-32 Swept and yawed wing aerodynamics.

PRINCIPAL ASSUMPTIONS

- lift-curve slope of normal sections not affected by spanwise flow
- Mach number normal to (swept) quarter chord defines compressibility effects
- angle of attack and chord in local flow direction define drag and stall
- total viscous drag force (vector addition of spanwise and chordwise components) has same direction as local yawed flow, so spanwise drag component can be obtained from section drag coefficient
    - skin friction drag not pressure drag that has spanwise component; really in direction of velocity at bottom not top of boundary layer
    - analysis can calculate spanwise drag from chordwise drag coefficient, or from drag coefficient at zero angle of attack

NO CORRECTIONS IMPLEMENTED FOR EFFECTIVE SHAPE AND THICKNESS OF AIRFOIL IN YAWED FLOW

all information about shape and thickness is in airfoil tables

probably airfoil tables should correspond to shape of cross-section in local flow direction (or mean direction for time-varying flow)

so for rotor blade, shape perpendicular to wing reference line (straight or drooped) through center of rotation is appropriate

which may or may not correspond to how the section geometry was defined

DISCUSSING HERE HOW TO ANALYZE WING IN SWEPT/YAWED FLOW

how to manufacture wing is separate issue

can define airfoil sections relative straight reference axis, or relative swept quarter chord

how to describe wing geometry is separate issue

here describe planform by chord measured perpendicular to reference line (usually straight reference line, so wing area is easily calculated)

STATIC STALL DELAY

THERE IS EVIDENCE THAT ROTATIONAL EFFECTS ON BOUNDARY LAYER PRODUCE A DELAY OF SEPARATION, PARTICULARLY FOR INBOARD SECTIONS OF ROTATING WINGS

three-dimensional aerodynamic phenomenon

modelled using input stall delay factors to modify the lift, drag, and moment coefficients obtained from tables

TIP FLOW CORRECTIONS

COMPRESSIBLE TIP RELIEF

small reduction in effective Mach number

implemented using input factor for Mach number

TIP LOSS FACTOR

needed when induced velocity is obtained from uniform inflow model (an approximation for wake solution)

wing has no lift outboard of span station B

typically tip loss factor B represents about 3% of wing span

TIP FLOW CORRECTIONS

SPAN STATION OF ROLLED UP TIP VORTEX

- tip loss factor not used for nonuniform inflow, but must consider span station of rolled-up tip vortex when it reaches trailing edge of wing, r_{TV}
  - wing has no circulation outboard of r_{TV}
  - figure 33 illustrates formation of vortex on wing tip
- implementation similar to tip loss factor
- typically tip vortex forms 1–2% span inboard of wing tip (rectangular planform)
  - little effect on loading; should suppress effect to avoid convergence problems
- for highly tapered tip, vortex can form 5–6% span inboard of wing tip
  - significant effect on loading
- if all or most of an aerodynamic panel is outboard of r_{TV}, wake induces large upwash at it, perhaps stalling wing section
  - for highly tapered tip this is representation of physical effect
  - for rectangular tip it is just source of convergence problems

TIP VORTEX INITIAL SPAN STATION IN UNDISTORTED WAKE GEOMETRY CAN BE INBOARD OF WING TIP

- feature of wake geometry model, in addition to this affect on loading

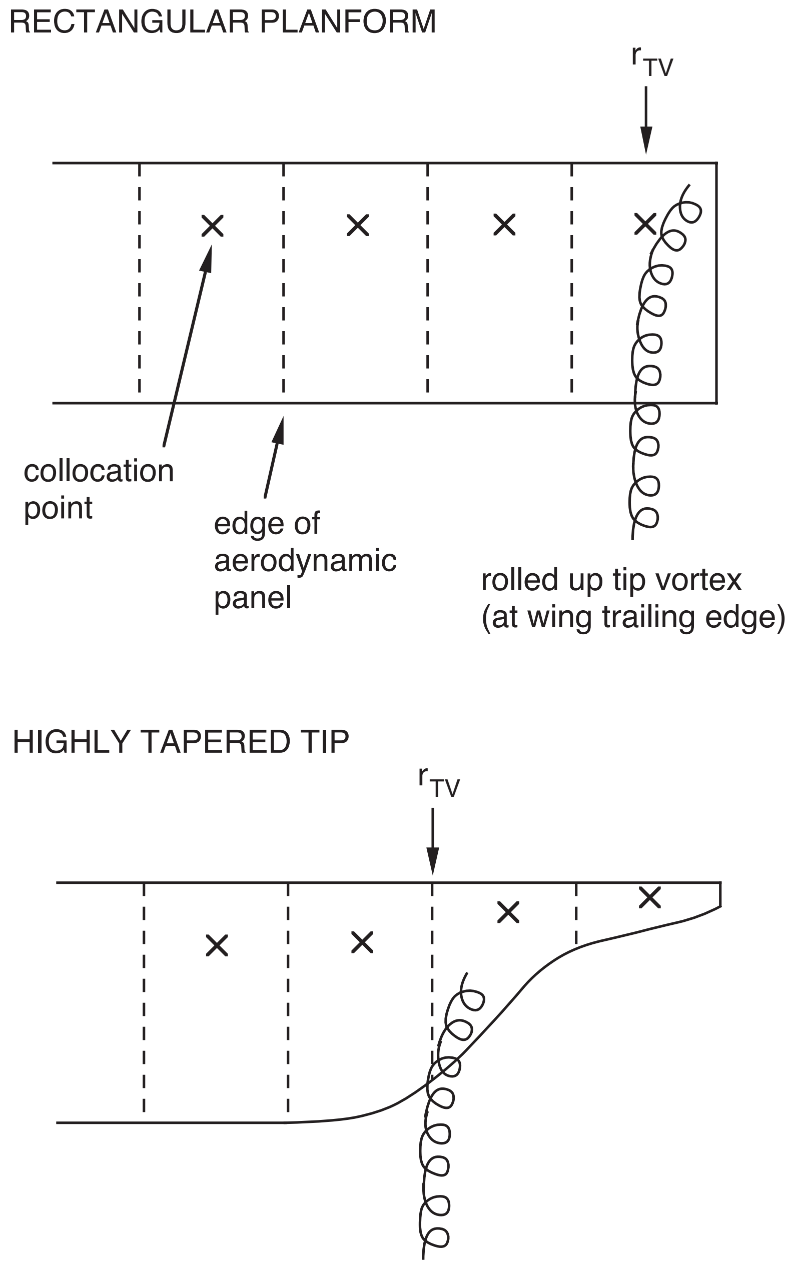
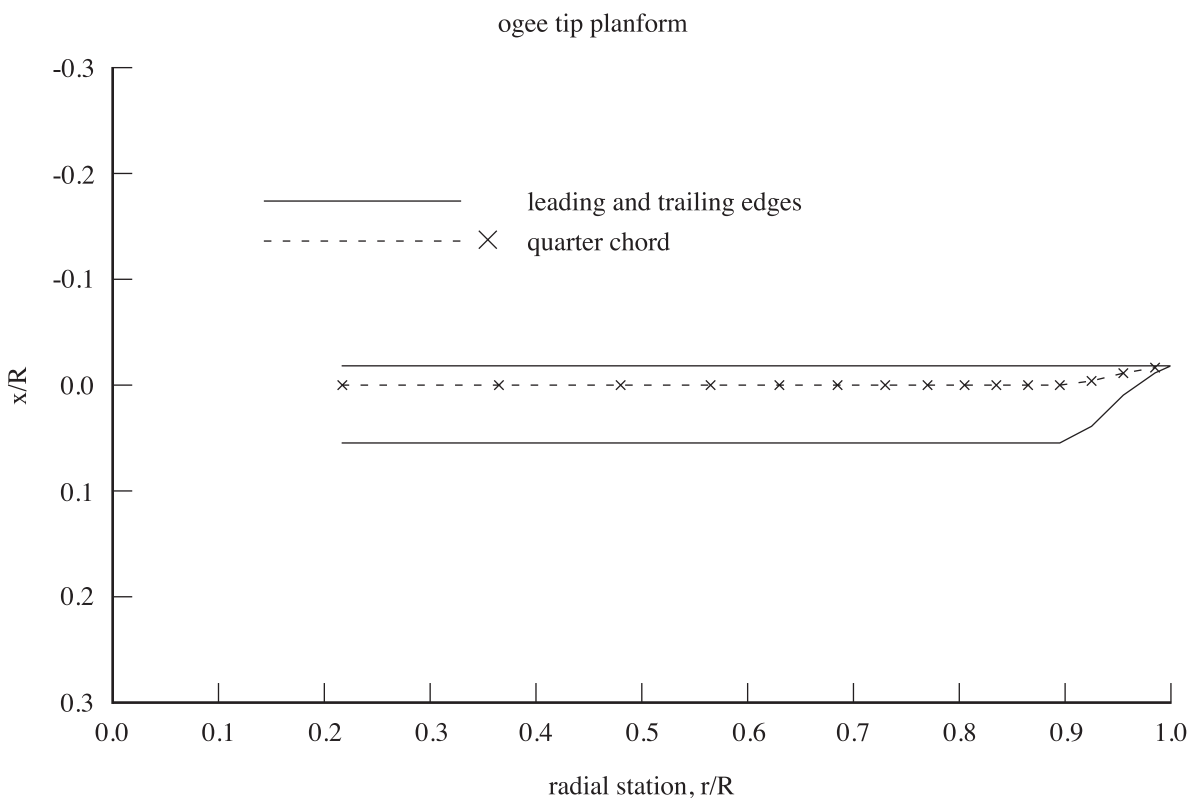
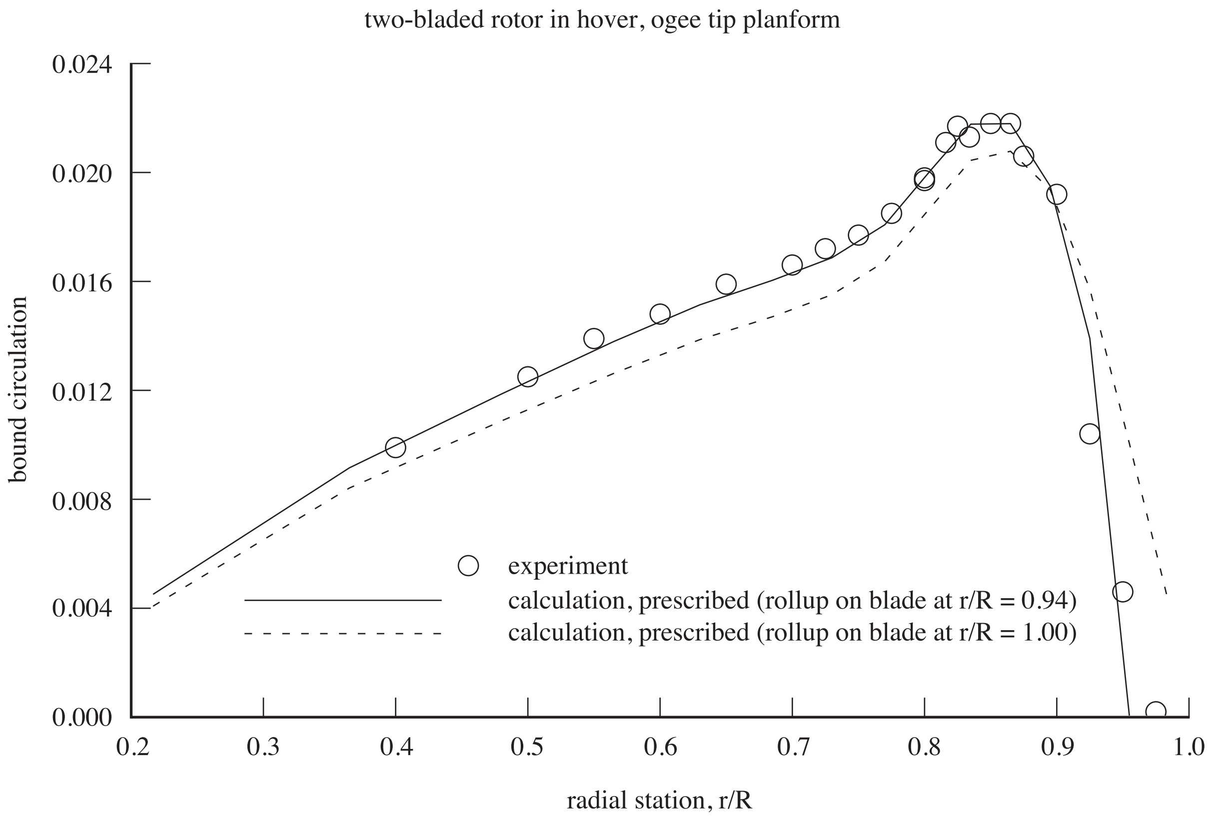


Figure 5-33 Formation of vortex on wing tip.


ogee tip planform
-0.3
-0.2
-0.1
0.0
0.1
0.2
0.3
x/R
leading and trailing edges
quarter chord
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
radial station, r/R



two-bladed rotor in hover, ogee tip planform
bound circulation
0.024
0.020
0.016
0.012
0.008
0.004
0.000
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
radial station, r/R
experiment
calculation, prescribed (rollup on blade at r/R = 0.94)
calculation, prescribed (rollup on blade at r/R = 1.00)

REYNOLDS NUMBER CORRECTIONS

IF THE REYNOLDS NUMBER Re OF ROTOR SECTION IS NOT EQUAL REYNOLDS NUMBER Re_t OF AIRFOIL TABLE, DRAG AND LIFT COEFFICIENTS CAN BE CORRECTED

for airfoil Reynolds number lower than table, drag increased and maximum lift decreased

correction uses factor $K = (Re/Re_t)^n$, $n = 0.125$ to 0.2

TABLE REYNOLDS NUMBER CALCULATED FROM REYNOLDS NUMBER CORRESPONDING TO MACH NUMBER $M = 1$: $Re_t = M_e Re_1$

approximation for Re vs M of table

this correction is the only use analysis makes of Reynolds number information in airfoil table

ADJUSTMENTS TO AIRFOIL TABLE DATA (FUNCTION OF SPAN STATION)

LIFT, DRAG, MOMENT COEFFICIENT INCREMENTS; AERODYNAMIC CENTER INCREMENT

provide means to modify aerodynamic characteristics without changing airfoil table

examples: drag increment for Reynolds number effect; aerodynamic center increment for three-dimensional effects at wing tip

TRAILING-EDGE FLAP DERIVATIVES;
MACH NUMBER, MAXIMUM LIFT, DRAG CORRECTIONS;
STATIC STALL DELAY;
AIRFOIL TABLE α AND MOMENT REFERENCES

UNSTEADY LOADS IN ATTACHED FLOW

NEED THE NONCIRCULATORY LOADS (INDEPENDENT OF SHED WAKE)

static loads obtained from airfoil tables, and usually shed wake effects accounted for through wake-induced velocity

noncirculatory terms essential for aerodynamic pitch damping, and sometimes unsteady lift and even virtual mass terms are important

MODELS (THIN-AIRFOIL THEORY):

INCOMPRESSIBLE
ONERA EDLIN
LEISHMAN-BEDDOES

including effects of time-varying free stream and reverse flow, and trailing-edge flaps

options for lift-curve slope, aerodynamic center, and drag recovery factor

SHED WAKE CAN BE IN WING COMPONENT OR IN WAKE COMPONENT

if shed wake is in wing component, omit near shed vorticity of wake component

with near shed vorticity in wake component, or with dynamic inflow, should not include shed wake in wing component

however, Leishman-Beddoes “circulatory” terms include more than shed wake effects

DYNAMIC STALL

DYNAMIC STALL CHARACTERIZED BY

delay in occurrence of separated flow produced by wing motion

high transient loads induced by vortex shed from leading edge when stall does occur

EMPIRICAL DYNAMIC STALL MODELS IMPLEMENTED:

JOHNSON
BOEING
LEISHMAN-BEDDOES
ONERA EDLIN
ONERA BH

still use airfoil table for steady characteristics, evaluated at effective angle of attack that includes the dynamic stall delay

require 2D table of dynamic stall parameters (function of Mach number and span station)

sample cases include table for NACA 0012 airfoil

UNSTEADY LOADS AND DYNAMIC STALL MODELS MAY INTRODUCE STATES

usually implicit (solved internal to component)

or the component can have degrees of freedom and equations of motion

GEOMETRY AND AXES

figure 34 shows geometry of wing model

ANALYSIS IS PERFORMED IN AXES OF WING FRAME

WING REFERENCE LINE USED TO DEFINE GEOMETRY

- arbitrary curvature and torsion of reference line
- straight reference line simplest, and frequently sufficient
- rotorcraft shell uses straight line through center of rotation for wing reference line

WING CONSISTS OF SET OF AERODYNAMIC PANELS

- quarter chord and three-quarter chord collocation point at each panel
- component input at each collocation point:
  - motion relative air
    - velocity including interference terms
    - dynamic pressure
    - rate of change of velocity
  - location of wing, measured from the origin of the wing frame
    - position
    - rate of change of position
- component output at each collocation point:
  - aerodynamic force

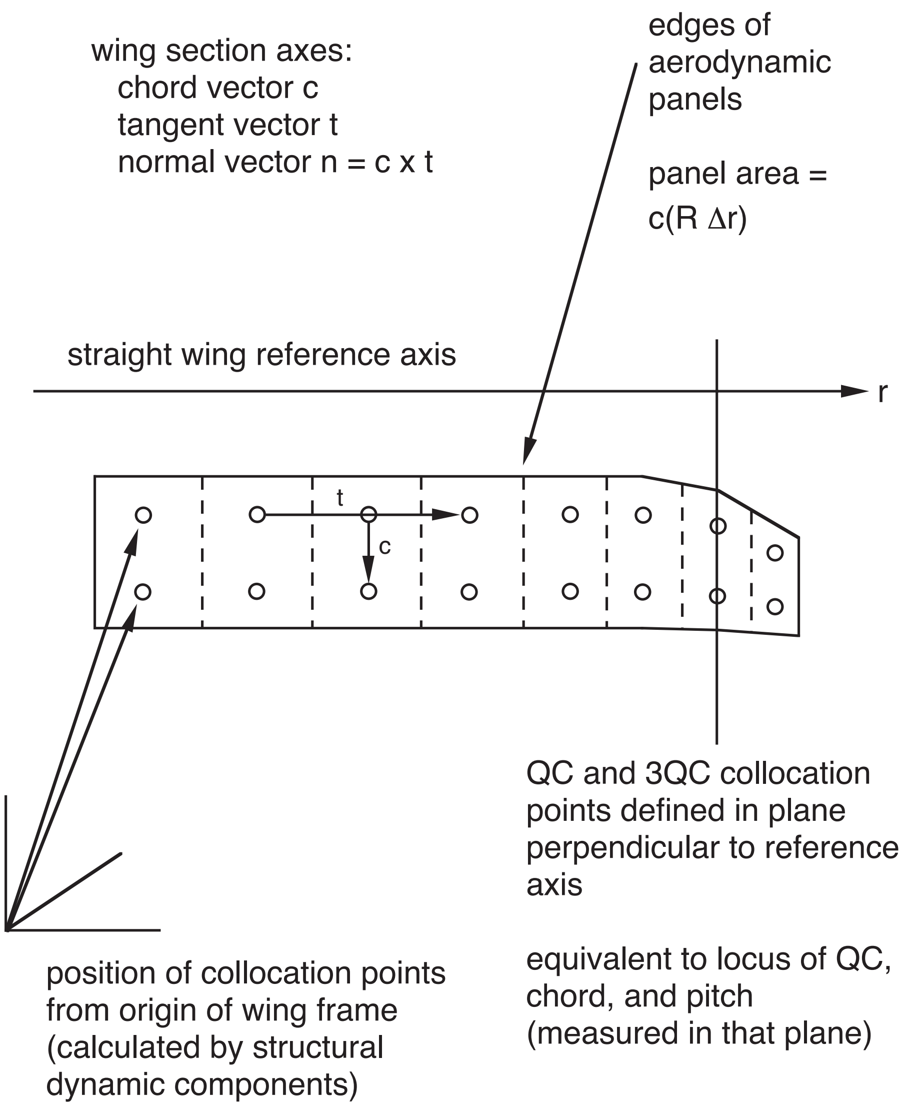


Figure 5-34 Geometry for definition and analysis of wing.

WING DESCRIPTION

- geometry defined relative to wing reference line
  - section properties measured in planes perpendicular to wing reference line
    - chord, quarter chord position, pitch
  - arbitrary sweep and droop (offsets of quarter chord from wing reference line)
  - arbitrary twist (from positions of three-quarter chord and quarter chord points in section plane)
- span station r identifies positions on wing reference line
  - positive r runs from root to tip of rotor blade
    - for both counter-clockwise and clockwise rotation
- span scale factor R such that Rr is true measure of distance along reference line
  - for rotor blade with straight reference line and no droop, r = dimensionless radial variable and R = blade radius
    - with large droop, will need wing reference line that includes the droop
    - need not change definition of span station r
- panel width (including R) times chord gives wing area

INPUT DATA

aerodynamic panels defined by span stations at edges

wing tips are first and last edges

collocation points (QC and 3QC) are at panel midpoints

aerodynamic properties defined at arbitrary span stations, with linear variation between

ANALYSIS

aerodynamic analysis in section axes

chord, normal, tangent vectors calculated from positions of collocation points

loads in sections axes are chord and normal forces

conventional output available in wing plane axes (rotated from section axes by pitch angle θ, so x-axis is in hub plane)

figure 35 shows section aerodynamic environment

calculate section forces and pitch moment at point cx_{AF} aft of airfoil leading edge

x_{AF} is position of moment reference axis of airfoil table moment data

θ_{AF} accounts for difference between twist definition and chord line of airfoil table data

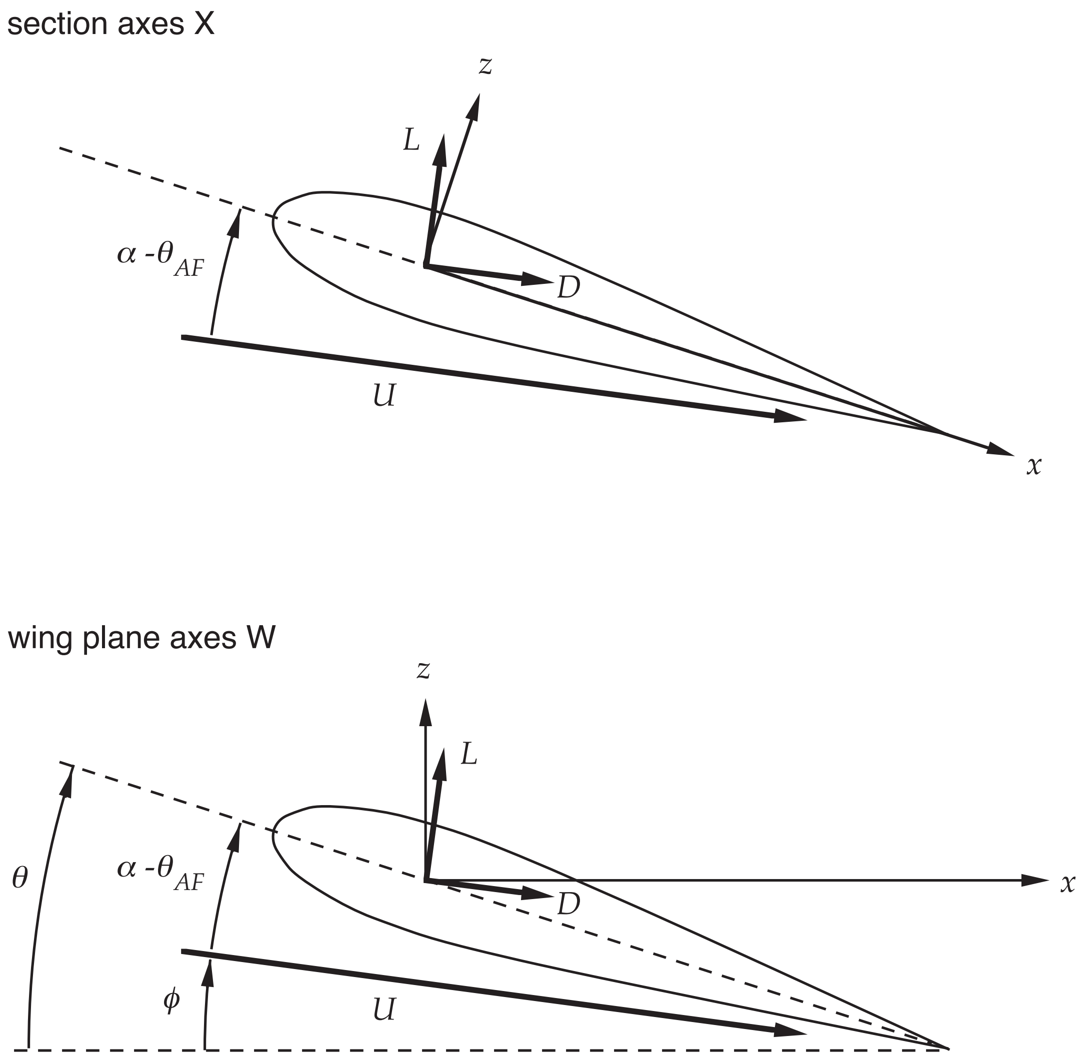


Figure 5-35 Section aerodynamic environment (nose up pitch moment and spanwise drag not shown; lift and drag act a distance cx_{AF} aft of the leading edge).

TRAILING-EDGE FLAPS

WING CAN HAVE ONE OR MORE TRAILING-EDGE FLAPS

each flap extends over one or more aerodynamic panels

flap deflection angle ϕ and its derivatives identified as elements of some control vector (component input)

AIRFOIL TABLE FILE INCLUDES EFFECTS OF FLAPS

total coefficients and flap coefficients,
function of α, M, ϕ, span station

in addition, flap deflection can produce increments in the coefficients through input derivatives

UNSTEADY LOADS INCLUDE EFFECTS OF FLAPS

with aerodynamically balanced flap (flap leading edge forward of flap hinge), unsteady loads depend on whether gap at flap leading edge is sealed or open

EXTERNAL AEROACOUSTIC ANALYSIS

CAMRAD II CAN BE USED WITH EXTERNAL AEROACOUSTIC ANALYSIS

external analysis typically employs methods of computational fluid dynamics

possibly with limited computational domain that does not encompass entire wing wake

external analysis may not calculate effects of structural dynamic motion on aerodynamic boundary conditions

figure 36 illustrates communcation between CAMRAD II and external aeroacoustic analysis

PROCEDURE KNOWN AS LOOSE COUPLING

COMMUNICATION BASED ON WING MOTION (AERODYNAMIC OR STRUCTURAL DYNAMIC SENSORS), AND PRESCRIBED INCREMENTS IN SECTION COEFFICIENTS

IF CFD ANALYSIS ACCOUNTS FOR ENTIRE FLOW FIELD, THEN IT REQUIRES JUST STRUCTURAL MOTION

otherwise, may use partial angle-of-attack α_P for boundary condition

so wake not used twice, wake-induced velocity used to evaluate α_P is calculated excluding wake vorticity in computational domain of external analysis

α_P includes structural motion

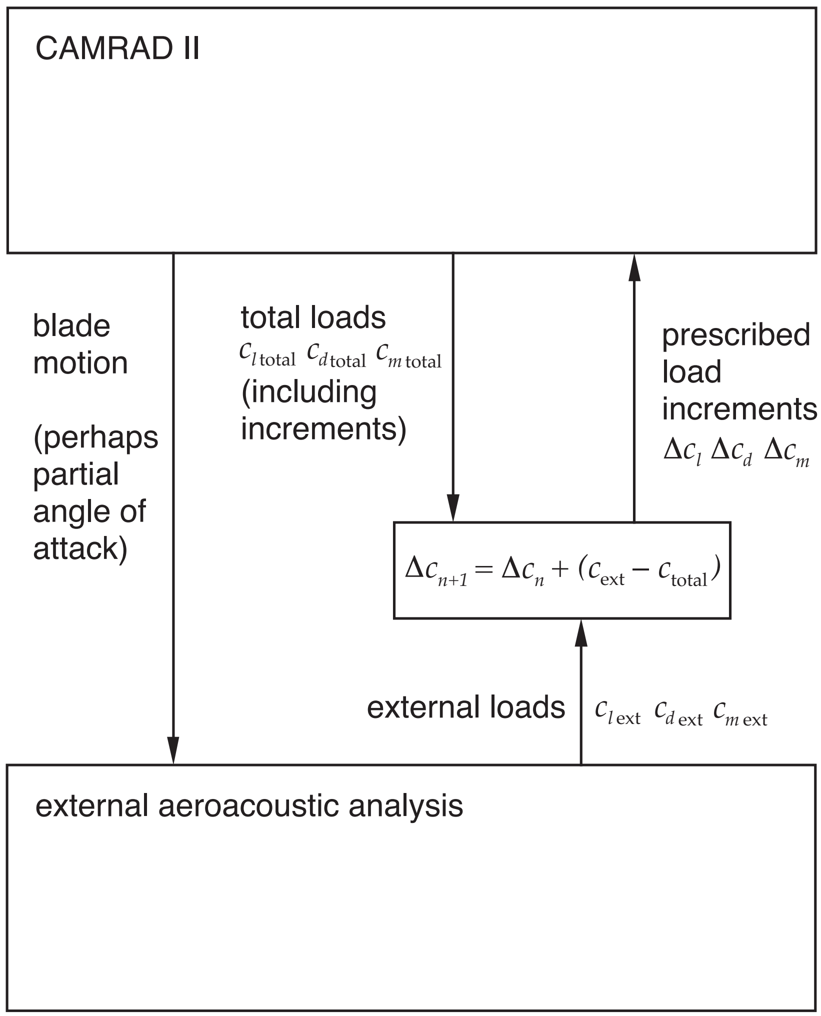


Figure 5-36 Communication with an external aeroacoustic analysis.

SETTING LIFT COEFFICIENT TO VALUE FROM EXTERNAL ANALYSIS DOES NOT ACCOUNT FOR CHANGES IN ANGLE-OF-ATTACK AS WING MOTION AND WAKE EFFECTS ARE UPDATED

approach:

$$\begin{aligned} c_{\ell\text{total}}(\alpha) &= c_{\ell\text{ext}}(\alpha_{\text{old}}) + c_{\ell_\alpha}(\alpha - \alpha_{\text{old}}) \\ &= c_{\ell\text{ext}}(\alpha_{\text{old}}) + c_{\ell\text{int}}(\alpha) - c_{\ell\text{old}}(\alpha_{\text{old}}) \\ &= c_{\ell\text{int}}(\alpha) + (c_{\ell\text{ext}}(\alpha_{\text{old}}) - c_{\ell\text{old}}(\alpha_{\text{old}})) \\ &= c_{\ell\text{int}} + \Delta c_\ell \end{aligned}$$

so update of increment is:

$$\begin{aligned} (\Delta c_\ell)_{n+1} &= (c_{\ell\text{ext}})_n - (c_{\ell\text{old}})_n \\ &= (\Delta c_\ell)_n + (c_{\ell\text{ext}} - c_{\ell\text{total}})_n \end{aligned}$$

aerodynamic or structural dynamic sensors also provide quantities required for boundary conditions in the external analysis

PRESCRIBED COEFFICIENT INCREMENTS OBTAINED FROM TWO-DIMENSIONAL TABLE

coefficient Δc_ℓ, Δc_d, Δc_n, Δc_x, Δc_m, Δc_r, Δc_g
$\Delta c_{\ell f}$, Δc_{df}, Δc_{nf}, Δc_{xf}, Δc_{mf}, Δc_{rf}

or $\Delta(M^2 c)$ form

or load ΔF_{QC}, ΔF_{3QC}, ΔM_{QC}, ΔF_f, ΔM_f

function of span station and time,
or span station and azimuth

PARTIAL ANGLE-OF-ATTACK

external analysis may account for — and wake component must exclude — when calculating induced velocity:

wake vorticity directly behind wing

and perhaps tip vortices from other wings

or all vorticity inside domain

wing component calculates α_P from this induced velocity, including structural motion

computational domain approximated by box

domain boundary must be correct where it intersects wake; α_P probably not sensitive to small errors in that intersection

OFTEN EXTERNAL ANALYSIS REQUIRES HIGH RESOLUTION IN SPAN, TIME, AND WAKE AGE

so calculate α_P and other quantities in separate loop in trim task

using fixed wake geometry and structural motion, after converged solution solution for structural dynamic response obtained

5–20 Rotor Inflow Component

ROTOR INFLOW COMPONENT CALCULATES WAKE INDUCED VELOCITY OF A ROTOR

- “UNIFORM INFLOW” COMPONENT (ALTHOUGH VELOCITY CAN VARY OVER ROTOR DISK)
- THERE MAY BE CORRESPONDING NONUNIFORM INFLOW COMPONENT
  - which component gives nonzero velocity depends on specified model, and perhaps on wake loop level
    - figure 37 illustrates functionality

INDUCED VELOCITY MODEL

- GLOBAL, QUASISTATIC MODEL FOR ROTOR
  - designed to be used with mean (filtered) values of component input
- EMPIRICAL MODEL BASED ON IDEAL-WING THEORY
  - momentum theory
  - induced velocity at rotor = mean term + linear variation over rotor disk
- OPTIONAL DUCTED FAN MODEL
- OPTIONAL DIFFERENTIAL MOMENTUM THEORY

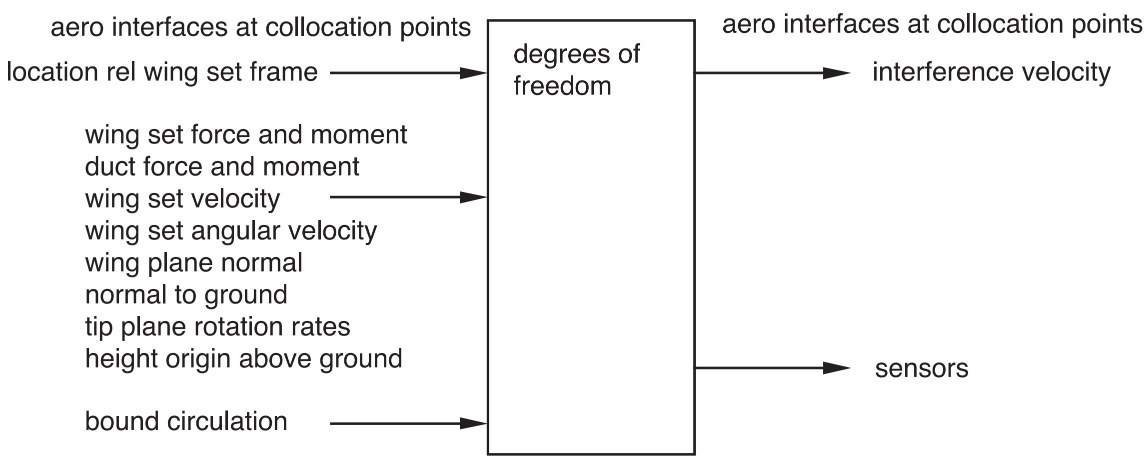


Figure 5-37 Functionality of rotor inflow component.

MOMENTUM THEORY

$$\lambda_i = \kappa \, \frac{C_T}{2\sqrt{\lambda^2 + \mu^2}}$$

with corrections for

nonideal induced losses in hover and forward flight (factors κ_h for axial flow and κ_f for edgewise flow)

singularity of momentum theory at ideal autorotation of axial flow

vortex ring state

circulation loop divergence for hovering rotor near zero thrust

ground effect

linear inflow variation over rotor disk, from edgewise flow and hub moments and angular velocity

assume that induced velocity is in direction of z-axis of rotor axes (shaft axes or tip-path plane axes)

can calculate interference velocity at points off this rotor

induced velocity = (input interference factor) times (mean inflow at rotor)

input interference factor accounts for fraction of system at collocation point affected by wake, and for fraction of fully developed wake achieved at collocation point

DYNAMIC INFLOW

FOR INDUCED VELOCITY IN TRANSIENT OR FLUTTER TASK (PERTURBATIONS FROM TRIM INFLOW)

GLOBAL, LOW FREQUENCY MODEL

MEAN AND LINEAR INDUCED VELOCITY COMPONENTS RELATED TO NET AERODYNAMIC THRUST AND HUB MOMENTS ON ROTOR

inflow degrees of freedom

$$\delta\lambda = \begin{pmatrix} \delta\lambda_u \\ \delta\lambda_x \\ \delta\lambda_y \end{pmatrix}$$

and loading

$$\delta L = \begin{pmatrix} \delta C_T \\ \delta C_{My} \\ \delta C_{Mx} \end{pmatrix}$$

first order differential equations (linearized model, with time lags)

$$\tau\,\delta\dot{\lambda} + \delta\lambda = \left(\frac{\partial\lambda}{\partial L}\right)\,\delta L$$

must also include induced velocity perturbations produced by wing speed changes

STATIC DERIVATIVE MATRIX $(\partial\lambda/\partial L)$ FROM

differential momentum theory (perturbation of uniform inflow model), which gives good results hover

or from unsteady actuator disk theory (Pitt and Peters), which is needed for good results in forward flight

DIFFERENTIAL MOMENTUM THEORY

INDUCED VELOCITY CALCULATED FROM BOUND CIRCULATION AT SPAN STATION OF COLLOCATION POINT

hence λ_i varies over disk

intended for axial flow, but can be used in forward flight as well

INCLUDING TIP LOSS CORRECTION

DUCTED FAN

COMPONENT INPUT = ROTOR FORCE AND DUCT FORCE

duct force calculated by some aerodynamic component

MOMENTUM THEORY FOR DUCTED FAN

depends on ratio rotor thrust to total thrust

which determines wake velocity ratio and area ratio (ratio of far wake value and value at rotor disk)

SIMPLE DUCT FORCE MODEL: SPECIFY RATIO ROTOR LOAD TO TOTAL LOAD

apply duct force to structure, at rotor hub node

HOVERING ROTOR NEAR ZERO THRUST

CIRCULATION ITERATION

consider inflow solution depending on rotor lift L; relaxation can be applied to L:

$$\overline{L}_n = \lambda L_n + (1-\lambda)\overline{L}_{n-1}$$

circulation iteration can be represented as operation

$$L_n = A(\overline{L}_{n-1})$$

successive substitution iteration will converge if

$$|1-\lambda+\lambda A'| < 1$$

for any finite A', relaxation factor can be found that produces convergence

but method fails if $A' = \infty$

to analyze system in such case, necessary to change definition of problem: either change order that parts are solved in loop; or change physical model that is source of sensitivity of A to L

CIRCULATION CONVERGENCE IS PROBLEM FOR HOVERING ROTOR AT LOW THRUST

uniform inflow gives for hover

$$C_T = c_1\theta - c_2\lambda = c_1\theta - c_3 C_T^{1/2} = A(C_T)$$

$$A'(C_T) = -\tfrac{c_3}{2}\, C_T^{-1/2}$$

so $A' = \infty$ at zero thrust

MODIFIED INFLOW CALCULATION REQUIRED TO ACHIEVE CONVERGENCE OF CIRCULATION ITERATION

AVOID PROBLEM BY FIXING WAKE GEOMETRY

uniform inflow

for hover, calculate induced velocity from

$$\lambda_i = \frac{\kappa_h^2 C_T}{2\lambda_0}$$

where $\lambda_0 = \kappa_h\sqrt{C_{T0}/2}$, and C_{T0} is fixed, nominal value of rotor thrust

then $A' = -c_4/\lambda_0$ is finite (and independent of C_T), and circulation iteration will converge for small enough nonzero relaxation factor

nonuniform inflow

fix wake geometry directly by using input values of constants K_1, K_2, K_3, and K_4 (instead of calculating them from rotor thrust)

RESULTING INDUCED VELOCITY IS NOT ACCURATE

but acceptable for cases of small thrust, as long as reasonable value of C_{T0} is used

5–21 Rotor Dynamic Wake Component

ROTOR DYNAMIC WAKE COMPONENT CALCULATES WAKE INDUCED VELOCITY OF A ROTOR

- figure 38 illustrates functionality

INDUCED VELOCITY CALCULATED USING GENERALIZED DYNAMIC WAKE THEORY DEVELOPED BY PETERS AND HE

- UNSTEADY WAKE THEORY FOR LIFTING ROTORS
  - based on acceleration potential for actuator disk
  - induced inflow at rotor disk is expressed as Fourier series azimuthally and polynomials radially

- RESULT IS SYSTEM OF FIRST-ORDER, ORDINARY DIFFERENTIAL EQUATIONS FOR INFLOW STATES
  - in trim task, obtain static solution for inflow variables (no degrees of freedom)
  - in transient and flutter tasks, solve ordinary differential equations

- FINITE-STATE INDUCED FLOW MODEL PARTICULARLY SUITABLE FOR ROTORCRAFT AEROELASTIC ANALYSIS

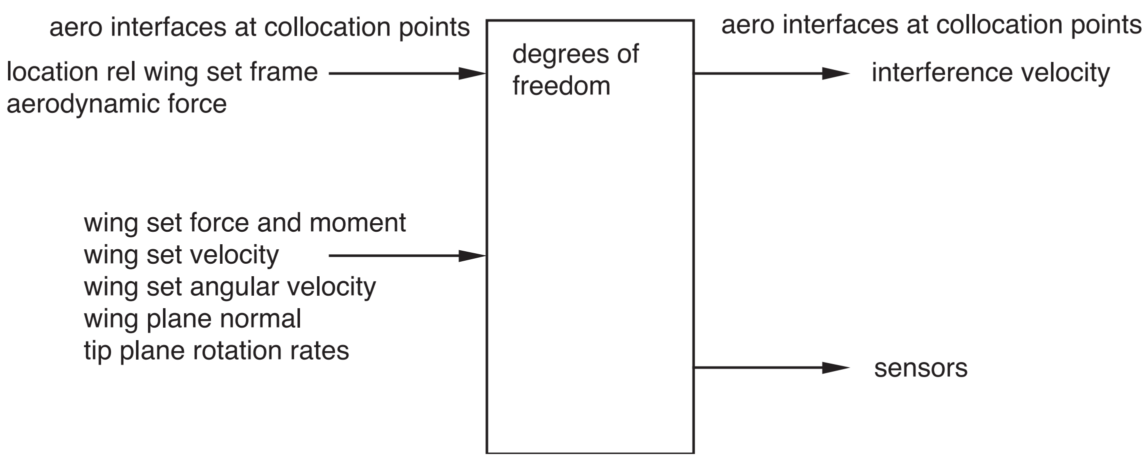


Figure 5-38 Functionality of rotor dynamic wake component.

DYNAMIC WAKE THEORY

INFLOW VELOCITY DEFINED BY DEGREE OF FREEDOM VECTOR α

ROTOR LOADING REPRESENTED BY GENERALIZED FORCE VECTOR τ

RELATED BY FIRST ORDER DIFFERENTIAL EQUATION

$$LM\dot{\alpha} + \alpha = L\tau$$

DERIVATIVE MATRIX $L = \partial\alpha/\partial\tau$ DEPENDS ON WAKE SKEW ANGLE $\chi = tan^{-1}|\mu/\lambda|$

WAKE-INDUCED VELOCITY CALCULATED FROM α

assume that induced velocity is in direction of z-axis of rotor axes (shaft axes or tip-path plane axes)

can calculate interference velocity at points off this rotor

induced velocity = (input interference factor) times (mean inflow at rotor)

input interference factor accounts for fraction of system at collocation point affected by wake, and for fraction of fully developed wake achieved at collocation point

NONLINEAR FORMULATION USED

uniform induced velocity term produced by the rotor thrust is the usual momentum theory result

including corrections for

nonideal induced losses in hover and forward flight (factors κ_h for axial flow and κ_f for edgewise flow)

singularity of momentum theory at ideal autorotation of axial flow (correction covers vortex ring state and turbulent wake state)

wake skew angle $\chi = tan^{-1}|\mu/\lambda|$ evaluated using total normal velocity λ, not just μ_z

rotor velocity replaced by $V = 1/2(\partial C_T/\partial\lambda)$ to account for influence of induced flow on total steady flow through disk

5–22 Wing Inflow Component

WING INFLOW COMPONENT CALCULATES WAKE INDUCED VELOCITY OF ONE NON-ROTATING WING

figure 39 illustrates functionality

INDUCED VELOCITY MODEL

GLOBAL, QUASISTATIC MODEL FOR WING

EMPIRICAL MODEL BASED ON IDEAL-WING THEORY

induced velocity at wing = mean term

from solution for minimum induced-drag of wing ($AR = b^2/S$ is wing aspect-ratio)

$$\lambda_i = \frac{v_i}{|V_x|} = \frac{C_L}{\pi e AR}$$

with correction for non-ideal losses (factor e)

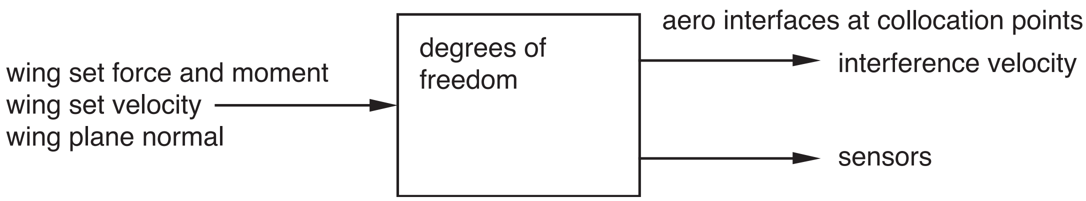


Figure 5-39 Functionality of wing inflow component.

5–23 Wing Wake Component

WING WAKE COMPONENT CALCULATES WAKE INDUCED VELOCITY OF A WING SET

ONE OR MORE WINGS IN SET; ONE WAKE PER WING

"NONUNIFORM INFLOW" COMPONENT

THERE MAY BE CORRESPONDING UNIFORM INFLOW COMPONENT

which component gives nonzero velocity depends on specified model, and perhaps on wake loop level

figure 40 illustrates functionality

VORTEX WAKE OF ROTOR IS FACTOR IN MOST PROBLEMS OF HELICOPTER

PERFORMANCE, BLADE LOADS, VIBRATION, NOISE

accurate calculation of wake-induced nonuniform inflow and resulting airloads needed in order to predict rotor behavior

need free wake geometry below advance ratio of about $\mu = 0.20$

AIRLOADS BEHAVIOR OF ROTARY WINGS

LOW SPEED
impulsive loading on advancing and retreating blades, caused by encounters of blade with tip vortices

HIGH SPEED
blade-vortex interaction loading still evident;
negative loading on advancing tip common,
consequence of flap moment balance, with stall-limited loads on retreating side

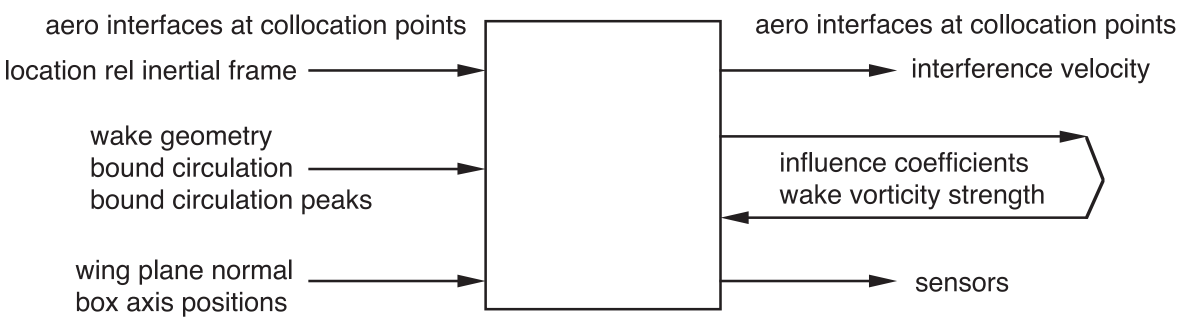
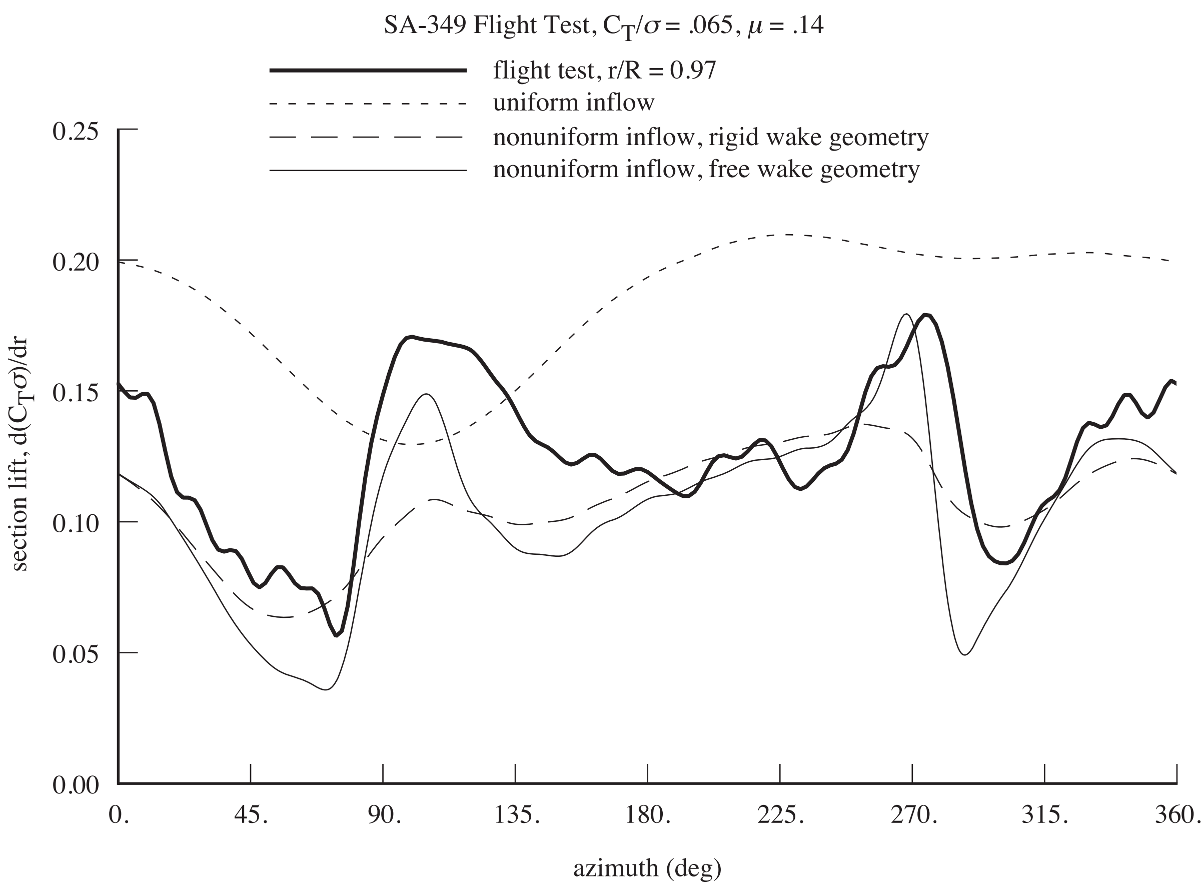
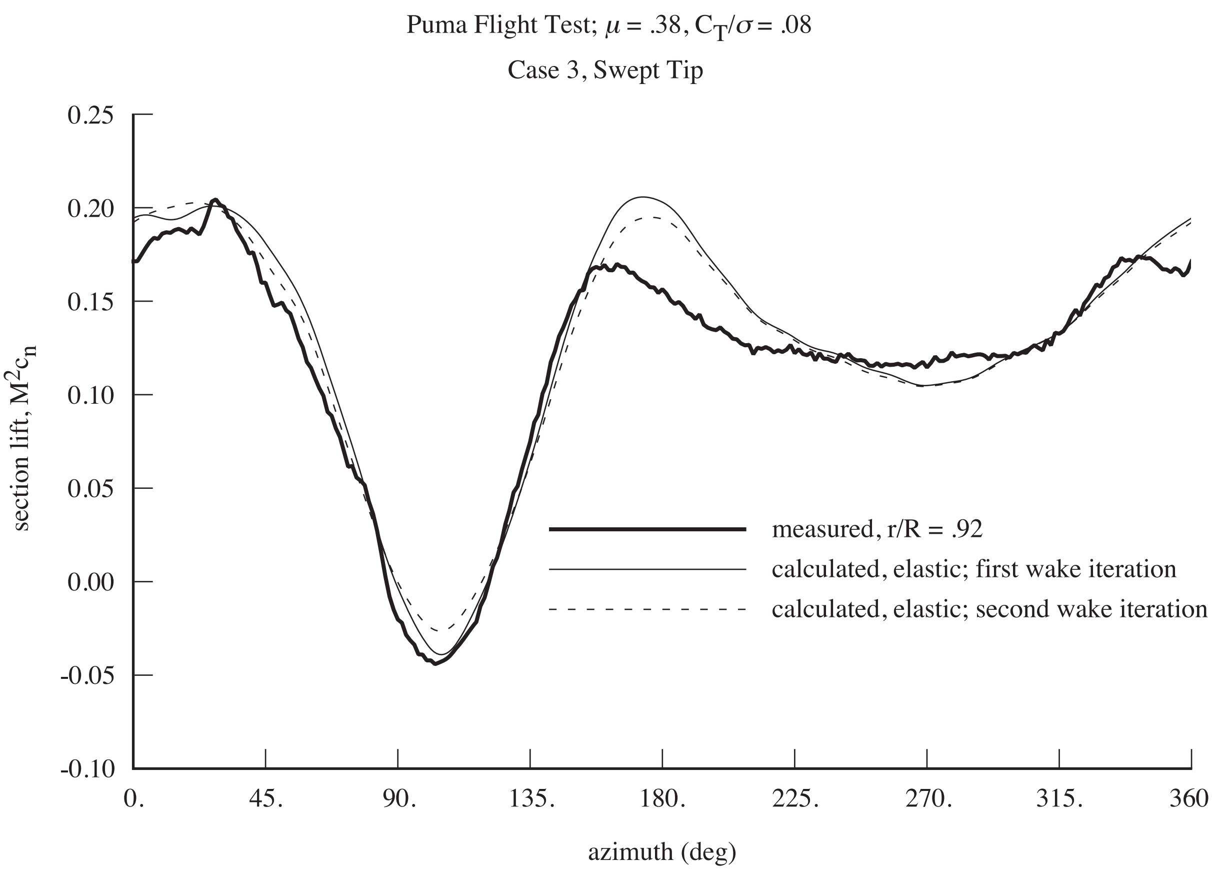


Figure 5-40 Functionality of wing wake component.



SA-349 Flight Test, $C_T/\sigma = .065$, $\mu = .14$
flight test, r/R = 0.97
uniform inflow
nonuniform inflow, rigid wake geometry
nonuniform inflow, free wake geometry
section lift, $d(C_T\sigma)/dr$
0.00
0.05
0.10
0.15
0.20
0.25
0.
45.
90.
135.
180.
225.
270.
315.
360.
azimuth (deg)


Puma Flight Test; μ = .38, C_T/σ = .08
Case 3, Swept Tip
section lift, M^2c_n
0.25
0.20
0.15
0.10
0.05
0.00
-0.05
-0.10
0.
45.
90.
135.
180.
225.
270.
315.
360.
azimuth (deg)
measured, r/R = .92
calculated, elastic; first wake iteration
calculated, elastic; second wake iteration

AERODYNAMIC MODEL

VELOCITY CALCULATED AT COLLOCATION POINTS

“induced velocity” at collocation points on this wing set (on wing generating wake, or on other wings)

“interference velocity” at other points (on other aerodynamic components, or arbitrary point in flow field)

VORTEX WAKE MODEL, WITH DISCRETIZED VORTEX ELEMENTS

incompressible flow in wake

integral equation (inflow depends on past circulation), which requires implicit solution

usually vortex lattice (straight-line segments) approximation

small viscous core radius for tip vortices

vortex sheet elements can be used for inboard wake, but usually sufficient (and more efficient) to approximate sheets by line segments, with large core radius to eliminate large velocities

model of wake rollup process

eventually tip vortex has strength of wing peak bound circulation at time that the wake elements were created

tip vortex core radius is input parameter

MODEL FEATURES

FOR LIFTING-LINE THEORY, COLLOCATION POINTS ON THIS WING SET CAN BE AT QUARTER CHORD OR THREE-QUARTER CHORD

- lifting line can be at wing quarter chord, or approximated by straight line

LINE AND SHEET WAKE ELEMENTS

- constant or linear strength lines
- nonplanar-quadrilateral sheets (or planar-rectangular; or approximate using lines)

MODEL OF WAKE ROLLUP PROCESS

- tip vortex rollup at one or both wing tips, or neither
- possibly two bound circulation peaks (inboard and outboard peaks of opposite sign): single-peak or dual-peak model
- entrainment and stretching in rollup process can be defined
- far wake trailed vorticity can be divided into several spanwise panels, to provide more detailed structure for inboard vorticity
    - perhaps with consolidation of trailed lines in the wake geometry

GROUND EFFECT

influence of ground can be included in wake-induced velocity calculation, through use of image elements in wake model

wake geometry components can also include ground plane influence

EXTERNAL AEROACOUSTIC ANALYSIS

induced velocity can be calculated excluding vortex elements inside computational domain of external analysis

BLADE-VORTEX INTERACTION LOADING

calculated using second-order lifting-line theory (three-quarter chord collocation point)

> or first-order theory (quarter chord collocation point) with a larger vortex core size

vortex core radius can be constant, or grow with wake age; and can include term that scales with trailed vorticity moment

can suppress blade-vortex interaction on inboard part of blade (as observed in measurements), by using large core radius when calculate velocity induced there

5-23.1 Wake Structure and Rollup

THREE-DIMENSIONAL WING TRAILS BOUND CIRCULATION Γ INTO A WAKE

figure 41 illustrates wake of rotor blade

SPANWISE VARIATION OF Γ PRODUCES TRAILED VORTICITY (ORIENTED IN LOCAL FREE STREAM DIRECTION)

TIME VARIATION OF Γ PRODUCES SHED VORTICITY (ORIENTED SPANWISE)

strength of trailed and shed vorticity determined by span and time derivatives of Γ at time wake element created

BOUND CIRCULATION TYPICALLY DROPS QUICKLY TO ZERO AT WING TIPS

so trailed sheet has high strength at wake outer edges

quickly rolls up to form concentrated tip vortices

ROLLUP PROCESS (ALSO INFLUENCED BY TIP GEOMETRY) PRODUCES LINE VORTEX WITH SMALL CORE RADIUS, HENCE LARGE VELOCITIES

ROTOR WAKE ROLLUP

BOUND CIRCULATION PEAK OCCURS NEAR BLADE TIP, GRADIENT OF BOUND CIRCULATION LOW AT BLADE ROOT

so blade tip vortex is strong and concentrated, while root vortex weaker and more diffuse

BASIS FOR DEVELOPING MODEL OF ROTOR WAKE, ALTHOUGH ROTOR WAKES ARE NOT ALWAYS SO SIMPLE

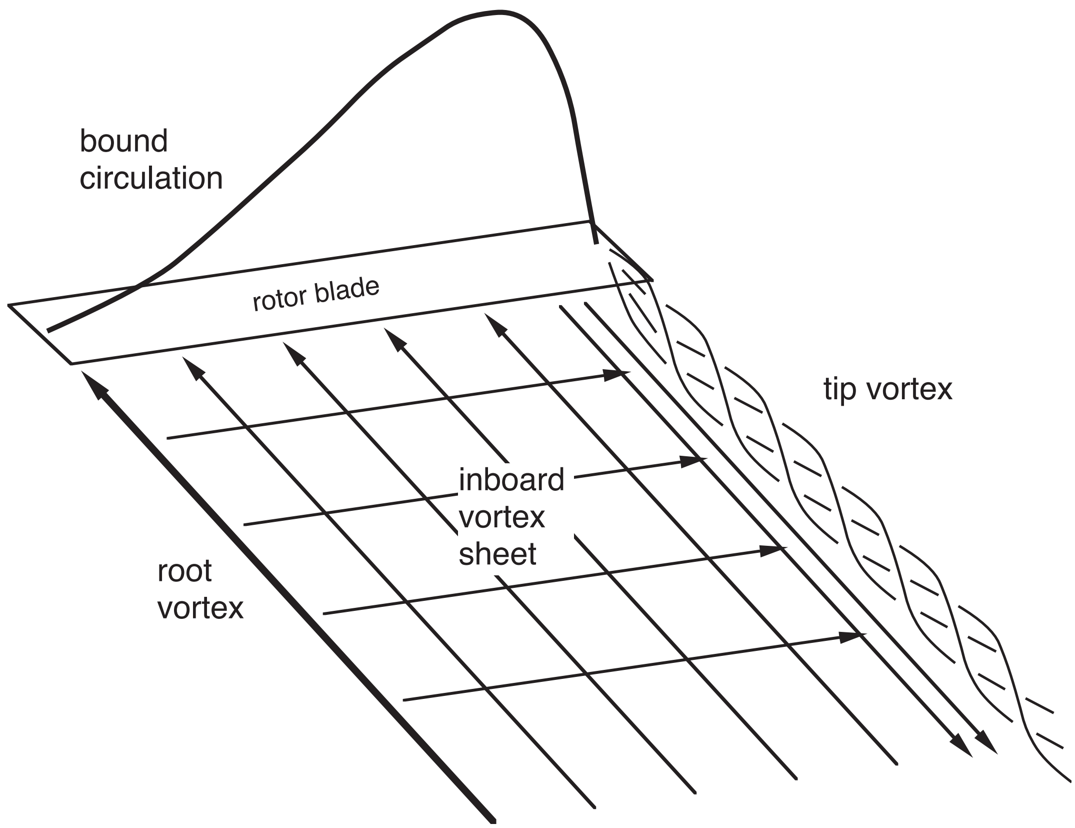


Figure 5-41 Trailed and shed vorticity in rotor wake.

WAKE GEOMETRY

WAKE GEOMETRY DESCRIBES POSITION OF WAKE VORTICITY IN SPACE

UNDISTORTED GEOMETRY OBTAINED FROM MOTION OF WING

- wake element is convected by wind, from position in air at which it was created

THIS GEOMETRY IS DISTORTED BY SELF-INDUCED VELOCITY OF ALL WAKES IN SYSTEM, AND BY WINGS AND BODIES

ROTOR WAKE GEOMETRY

WAKE CONVECTED DOWNWARD RELATIVE ROTOR BY MEAN INDUCED VELOCITY AND FREE STREAM, AFT IN FORWARD FLIGHT BY FREE STREAM

- in hover, tip vortices trailed in helix
- in forward flight, convected rearward as well as downward
  - with substantial distortion from self-induced velocities

ROTOR WAKE = CONCENTRATED TIP VORTICES, TRAILED IN DISTORTED, INTERLOCKING HELICES; ONE BEHIND EACH BLADE, SKEWED AFT IN FORWARD FLIGHT

- figure 42 shows typical wake geometry

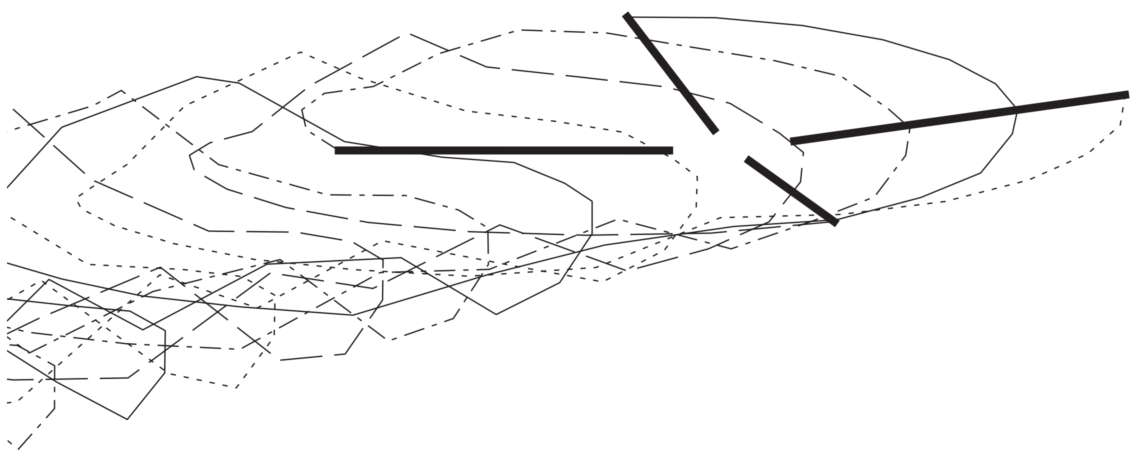


Figure 5-42 Calculated rotor free wake geometry ($C_T/\sigma = 0.087$ and $\mu = 0.18$).

TIP VORTICES

- VORTICITY IN TIP VORTEX DISTRIBUTED OVER SMALL BUT FINITE REGION, BECAUSE OF VISCOSITY
- VORTEX CORE IS IMPORTANT FACTOR IN INDUCED VELOCITY CHARACTER SINCE IT DEFINES AND LIMITS MAXIMUM VELOCITIES NEAR TIP VORTICES
- STRONG, CONCENTRATED TIP VORTICES ARE DOMINANT FEATURE OF HELICOPTER ROTOR WAKE
    - produce highly nonuniform flow field
    - so any aerodynamic component encountering this flow field will see large vortex-induced loadings
    - because of rotation, rotor blade encounters tip vortices from preceding blades
        - vortex-induced loading is principal source of higher harmonic loads on blades

5-23.2 Blade-Vortex Interaction

AIRLOADS PRODUCED BY BLADE-VORTEX INTERACTION DEPEND ON NUMEROUS PHYSICAL EFFECTS

extent of tip vortex rollup
tip vortex strength
size of viscous core
distorted wake geometry
lifting-surface effects on induced loading

and possibly even vortex bursting,
or vortex-induced stall on the wing,
or wing-induced geometry changes

VORTICITY IN TIP VORTEX DISTRIBUTED OVER SMALL BUT FINITE REGION, BECAUSE OF VISCOSITY

VISCOUS CORE DETERMINES VELOCITY NEAR LINE VORTEX

CONSIDER TANGENTIAL OR CIRCUMFERENTIAL VELOCITY v ABOUT LINE VORTEX, AT DISTANCE r FROM LINE

core radius r_c = distance r where have maximum v

figure 43 shows influence of core radius on peak velocity

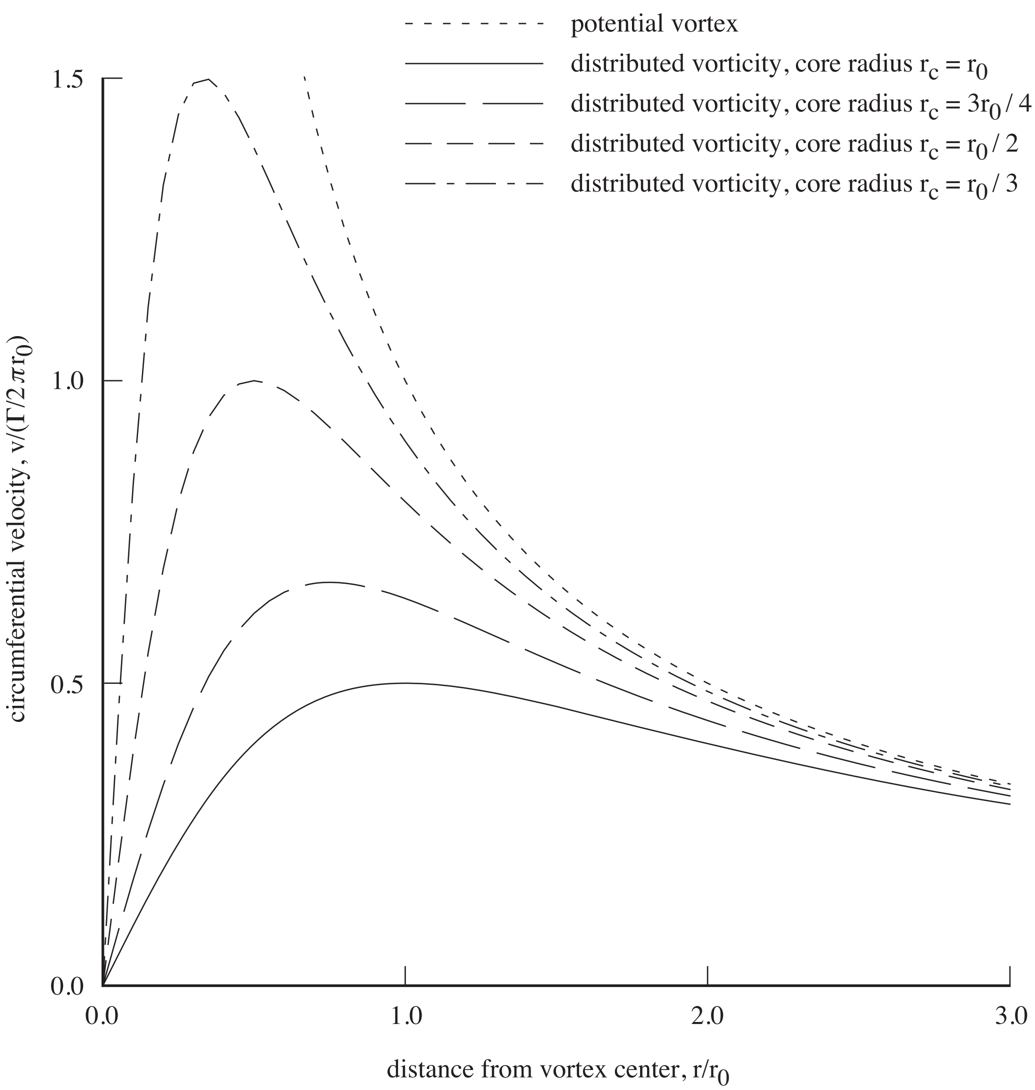


Figure 5-43 Tip vortex core radius and peak velocity.

FOR POTENTIAL LINE VORTEX (NO CORE):

$v = \Gamma/2\pi r$, where strength Γ is some fraction of maximum bound circulation (determined by rollup process)

at small r, viscosity reduces magnitude of v, by spreading vorticity over nonzero domain

FOR RANKINE VORTEX CORE

solid body rotation of fluid inside r_c

uniform vorticity distribution, concentrated entirely within core radius

maximum tangential velocity is $v_{\max} = \Gamma/2\pi r_c$

FOR DISTRIBUTED VORTICITY CORE

Scully model (a simple possibility) has circulation proportional to $r^2/(r^2 + r_c^2)$, so half of vorticity is outside core radius

maximum tangential velocity is $v_{\max} = \Gamma/4\pi r_c$

figure 44 shows influence of core type on velocity

DISTRIBUTED CORE MODEL IS MORE REALISTIC

measurements show that maximum tangential velocity is much less than $\Gamma/2\pi r_c$, so substantial fraction of vorticity is outside core radius

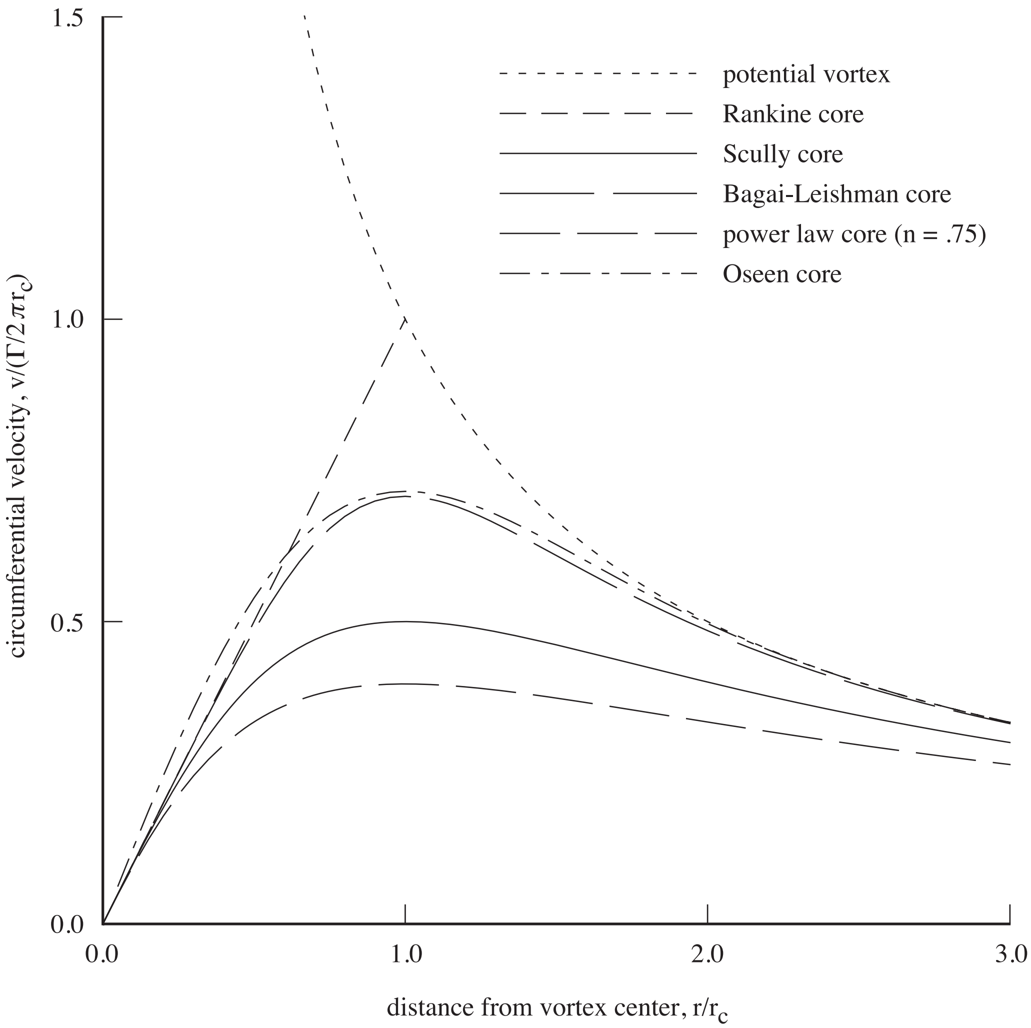


Figure 5-44 Tip vortex core types.

CORE RADIUS

- ONLY LIMITED EXPERIMENTAL DATA ON CORE SIZE, PARTICULARLY FOR FULL SCALE ROTOR BLADES
- ANALYSIS REQUIRES CORE RADIUS r_c OF ABOUT 20–25% CHORD
  - based on correlation with measured blade-vortex interaction loads on rotors in low speed flight
    - using distributed core model, and second-order lifting-line theory (three-quarter chord collocation point)
- LIMITED DATA SUGGEST THAT THE ACTUAL VISCOUS CORE RADIUS IS ABOUT 10% CHORD (FULL SCALE)
- AIRLOADS CORRELATION SUGGESTS THAT CORE SIZE VARIES WITH AZIMUTH
  - or rollup varies with azimuth, so less strength (compared to peak bound circulation) on retreating side

- CORE SIZE r_c IS CONVENIENT PARAMETER TO CONTROL AMPLITUDE OF CALCULATED BLADE-VORTEX INTERACTION LOADS
  - determines maximum tangential velocity about vortex (inversely proportional to r_c)
  - use value of r_c that accounts for actual viscous core radius, and also for all phenomena of interaction that are not otherwise modelled (or are inaccurately modelled)

BLADE-VORTEX INTERACTION ON HELICOPTER ROTORS IN LOW-SPEED FORWARD FLIGHT

when vortex-induced loads are calculated using core size that gives good correlation at blade tip, find that strength of blade-vortex interaction is significantly overpredicted for inboard radial stations (compared to flight test measurements)

measured vortex-induced loading is high when blade first encounters vortex, on advancing side;
decreases inboard as blade sweeps over vortex, on front of disk;
and then recovers again on retreating side

EVIDENTLY THERE IS SOME PHENOMENON LIMITING THE LOADS

MECHANISM REMAINS SPECULATIVE; POSSIBILITIES:

local distortion of the vortex geometry

bursting of vortex core, induced by blade

interaction of vortex with trailed wake it induces behind blade, with effect of diffusing and reducing circulation in vortex

local flow separation produced by high radial pressure gradients on blade

viscous effects on interaction, because blade-vortex separation is much smaller on front of disk than on sides

CORE SIZE CAN BE INCREASED FOR COLLOCATION POINTS ON INBOARD PART OF BLADE IN ORDER INCLUDE THIS EFFECT IN CALCULATIONS

simple way to model effects on airloads,
but no explanation of the physics is intended

5-23.3 Nonuniform Inflow Calculation

NONUNIFORM INFLOW CALCULATED BY INTEGRATING BIOT-SAVART LAW OVER WAKE

induced velocity produced by one wing at a collocation point:

$$v(t) = \int\int \mathbf{C}(t, \tau; r)\Gamma(t - \tau; r)\, dr\, d\tau$$

Γ = wing bound circulation

C = influence coefficients (differential operator, calculated from wake geometry and modelling assumptions)

at time t, wake element and its position in wake identified by wake age τ

$t - \tau$ = time wake element was created

SO INDUCED VELOCITIES DEPEND ON STRENGTH AND GEOMETRY OF WAKE VORTICITY

strength defined by span and time variation of bound circulation

geometry can be obtained from rigid, prescribed, or free models

INTEGRATION OVER WAKE

can not evaluate integrals analytically, even with no distortion

because of helical wake geometry of rotor

direct numerical integration not satisfactory either

large variations in integrand (produced by blade-vortex interactions) require small step sizes for accuracy

so wake modelled by set of discrete elements

Biot-Savart law integrated analytically for each element

total velocity obtained by summing contributions from all elements

accurate and efficient

DISCRETIZED WAKE

bound circulation and hence wake discretized spanwise

wake age discretized, using geometry and strength only at fixed wake age increment $\Delta\tau$

integral equation for the wake-induced velocity becomes:

$$v_A^I(t) = \sum_{\text{wings}} \Big[\sum_{\text{panels}} \sum_{k=0}^{K_F} \big(C_L(t,\tau_k)\Gamma_L(t-\tau_k) + C_R(t,\tau_k)\Gamma_R(t-\tau_k) \big) + \sum_{k=0}^{K_N} \sum_{i=1}^{M} C_N(t,\tau_k;r_{Ai})\Gamma(t-\tau_k;r_{Ai}) \Big]$$

DISCRETIZED WAKE MODEL

TIP VORTICES MODELLED AS CONNECTED SERIES OF STRAIGHT LINE SEGMENTS, WITH SOME KIND OF CORE REPRESENTATION

good model for most important part of wake

INBOARD TRAILED AND SHED VORTICITY TYPICALLY MODELLED AS VORTEX MESH OR LATTICE

with lattice (line segment) model of inboard wake, a large core size is needed

not representation of physical effect, but produces approximation for sheet element, by eliminating singularities in velocity for close passage of following wings

find line segments give about same results as sheet elements

model for inboard wake is not as good as tip vortex representation, but inboard wake is not as important

APPROXIMATIONS INVOLVED USING SET OF DISCRETE VORTEX ELEMENTS

replacing curvilinear geometry by series of straight line or planar elements; simplified distribution of strength over individual element (constant or linear); sheets replaced by lines, or planar-rectangle approximation for nonplanar-quadrilateral

PRACTICAL MODEL BALANCES ACCURACY AND EFFICIENCY OF SUCH APPROXIMATIONS

DEVELOPMENT OF DISCRETIZED MODEL

- ASSUME WING BOUND CIRCULATION KNOWN AT DISCRETE POINTS ALONG SPAN AND IN PAST
  - bound circulation $\Gamma(r,t)$ calculated at aerodynamic span stations r_{Ai}, and at wake ages $\tau_k = k\,\Delta\tau$, $k=0$ to K
  - linear variation of Γ between these points means wing generates wake of sheet elements (figure 45)

- STRENGTH OF TRAILED AND SHED VORTICITY IN AN ELEMENT OBTAINED FROM Γ AT TIME THAT VORTICITY WAS CREATED
  - spanwise change in Γ produces trailed vorticity
  - time change in Γ produces shed vorticity
  - wake geometry provides locations at four corners of element

ECONOMICAL APPROXIMATION FOR SHEET ELEMENT IS LINE SEGMENTS, WITH LARGE CORE TO AVOID VELOCITY SINGULARITIES NEAR LINE

- so vortex lattice model obtained by collapsing all sheet elements to finite strength line segments (figure 46)
- line segments are in center of sheet element, and points at which induced velocity are calculated (collocation points) are at midpoints of vortex lattice grid
  - locating collocation points midway between trailers is standard practice, to avoid singularities at lines
  - locating collocation points midway in time is required to correctly obtain shed wake effects

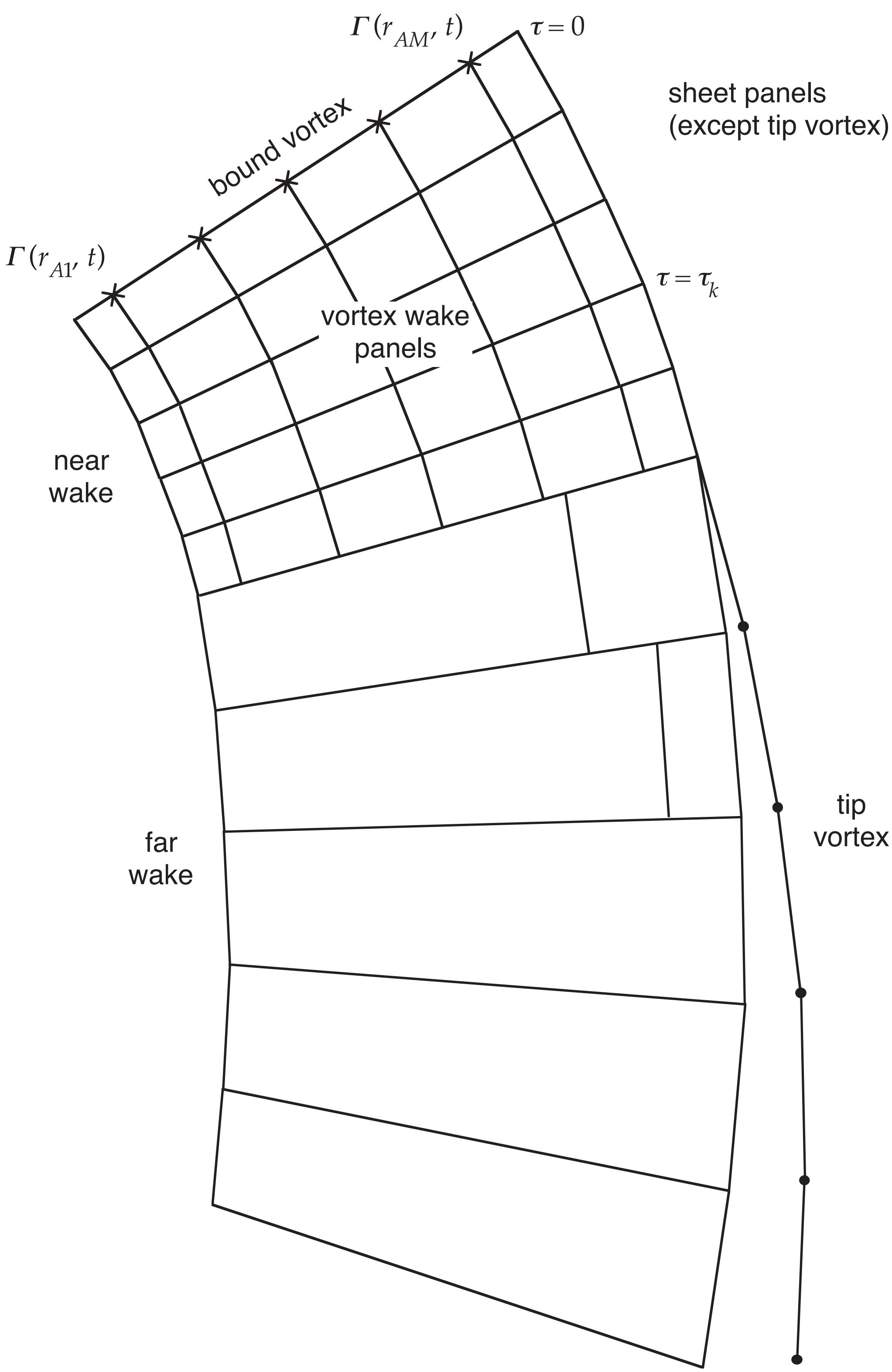


Figure 5-45 Wake model constructed from sheet panels (right tip vortex rollup).

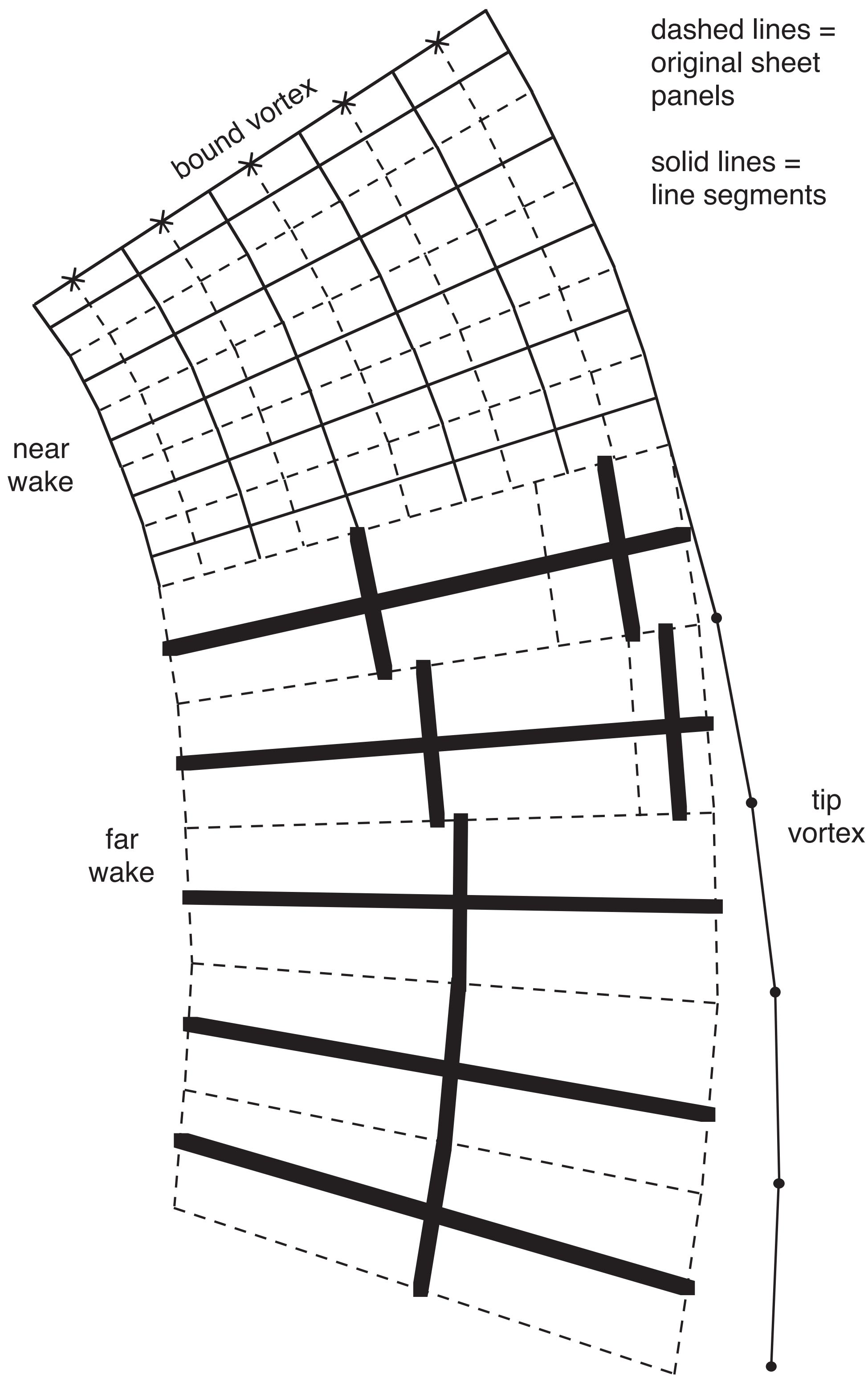


Figure 5-46a Wake model with line segments replacing sheet panels (right tip vortex rollup).

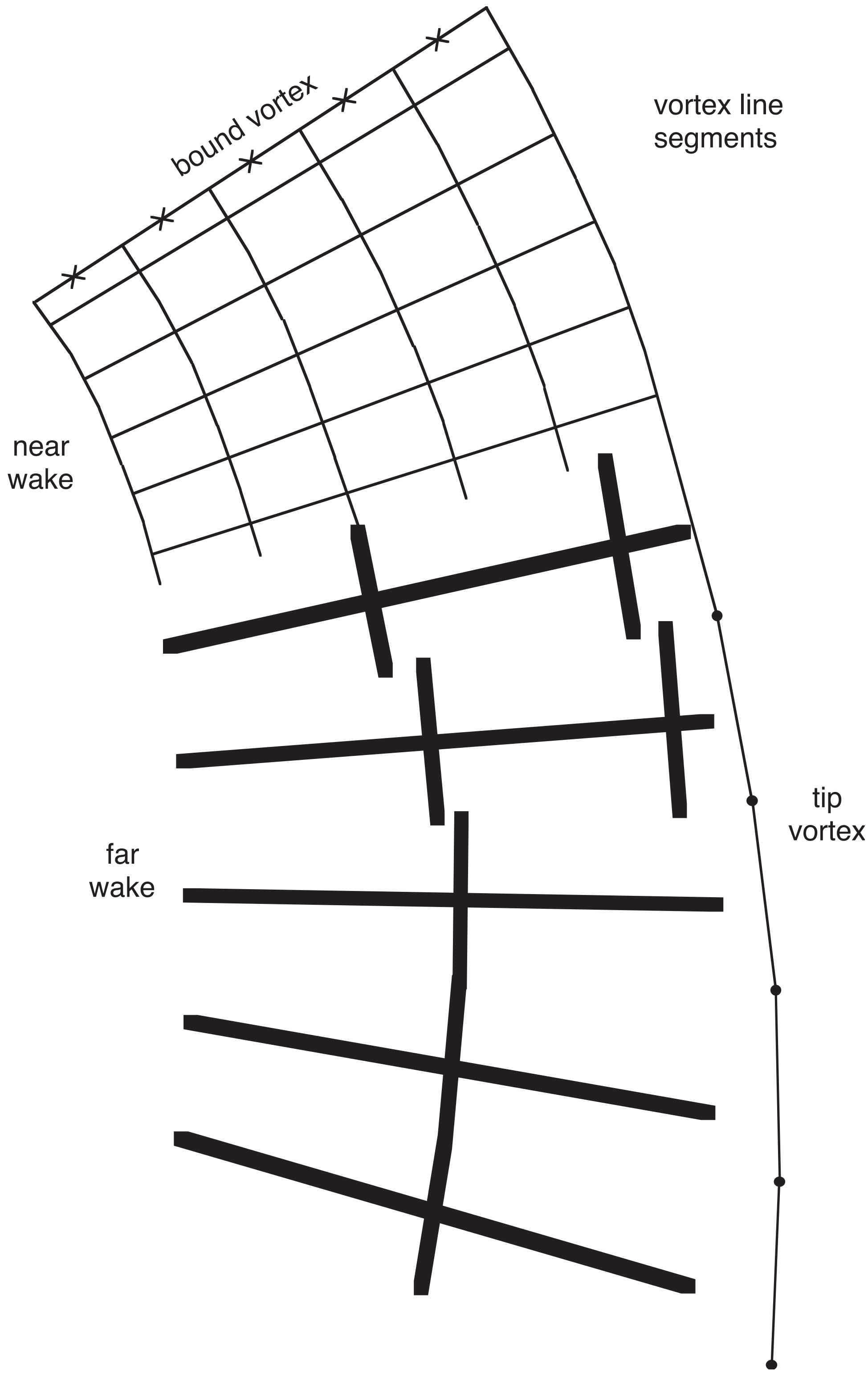


Figure 5-46b Wake model constructed from line segments (right tip vortex rollup).

WAKE MODEL FOR COLLOCATION POINTS ON WING THAT GENERATED THE WAKE

- WAKE DIRECTLY BEHIND THIS WING IS CALLED THE “NEAR WAKE”
- IMPLEMENTATION OF LIFTING-LINE THEORY INVOLVES PRIMARILY THIS REGION OF WAKE
    - most important requirement is to model detailed variation of wake strength, both spanwise and in time
        - to accurately get classical three-dimensional (Prandtl) and unsteady (Theodorsen) effects of wake on wing loading
    - rollup process is less important for near wake
- VORTEX LATTICE IS ALWAYS USED FOR NEAR WAKE (NOT SHEET ELEMENTS)
    - discretization of wake is better behaved with line segments than with sheet elements

- ROLLUP PROCESS NOT CONSIDERED IN MODELLING THE NEAR WAKE
    - except for effect of rollup on tip loading of generating wing
        - modelled by prescribing spanwise location of tip vortex, forcing wing to trail remaining bound circulation into wake at that span station
        - implemented in wing component (r_{TV})

MODEL OF ROLLUP PROCESS

- CONSIDER WAKE WHEN IT REACHES COLLOCATION POINTS ON A FOLLOWING WING OR OTHER AERODYNAMIC COMPONENT
- THIS WAKE REGION CALLED THE “FAR WAKE”
  - most important to model rollup process
- WAKE QUICKLY ROLLS UP AT OUTER EDGES TO FORM CONCENTRATED TIP VORTICES
  - so tip vortices dominate flow field, and must be modelled well
  - inboard vorticity less important, so more approximations are acceptable
- BOUND CIRCULATION CALCULATED AT MANY SPANWISE POINTS ON WING
  - but because of entrainment and stretching during rollup process, such detailed knowledge of vorticity strength not available in far wake
  - so far wake model constructed using bound circulation at small number of points — typically just circulation peaks

5-23.4 Wake Rollup Process

WAKE ROLLUP PROCESS

CALCULATION OF WAKE ROLLUP FROM FIRST PRINCIPLES IS NEEDED, BUT IT IS A DIFFICULT PROBLEM

vortex core largely formed at wing trailing edge, so problem is not inviscid rollup of vortex sheet

discretization of wake for rollup calculation is difficult (may not be well-posed even for inviscid, two-dimensional problem)

so rollup problem involves three-dimensional, unsteady, viscous fluid dynamics; compressible for rotors

beginning to be attacked using computational fluid dynamics methods

HERE WAKE ROLLUP PROCESS IS MODELLED NOT CALCULATED

structure and properties of rolled up wake are determined from assumptions and input parameters

and from spanwise distribution of bound circulation where wake was created

MODEL MUST ACCOUNT FOR INFLUENCE OF

strength of tip vortex when it encounters following wing

core radius when vortex is fully rolled up

ASSUMED ROLLUP PROCESS FOR WAKE MODEL

LIFT CONCENTRATED AT TIP OF WING

so trailed vorticity strength high at edge of wake, and vortex sheet quickly rolls up into concentrated tip vortex

tip vortex may form at both tips (for nonrotating wing), or at just one tip (for rotor blade)

FORMATION OF VORTEX INFLUENCED BY TIP GEOMETRY

with core largely formed by time vortex leaves trailing edge

ROLLED UP TIP VORTEX QUICKLY ATTAINS STRENGTH NEARLY EQUAL TO MAXIMUM BOUND CIRCULATION

TIP VORTEX HAS SMALL CORE RADIUS, DEPENDING ON WING GEOMETRY AND LOADING

SUCH CONCEPTS ARE BASIS FOR DEVELOPING MODEL OF ROTOR WAKE

HELICOPTER BLADE LOADING

wing loading at given time need not be all positive or all negative

helicopter rotor in high speed forward flight:

typically has negative lift on advancing tip, in second quadrant

for flapping rotor, net pitch and roll moment on hub must be small (zero with no flap hinge offset)

in forward flight, lift capability on retreating side limited by combination of low dynamic pressure and stall of airfoil

so lift on advancing side must also be small, to maintain roll balance

at high speed, lift on advancing tip can become negative

large twist of blade (built-in or elastic) increases negative loading

vortex-induced loading on following blades shows that this negative loading produces substantial negative trailed vorticity in wake

over much of rotor disk, Γ still positive all along span

range of azimuth on advancing side with negative peak near tip and positive peak inboard

figure 47 illustrates wake structure

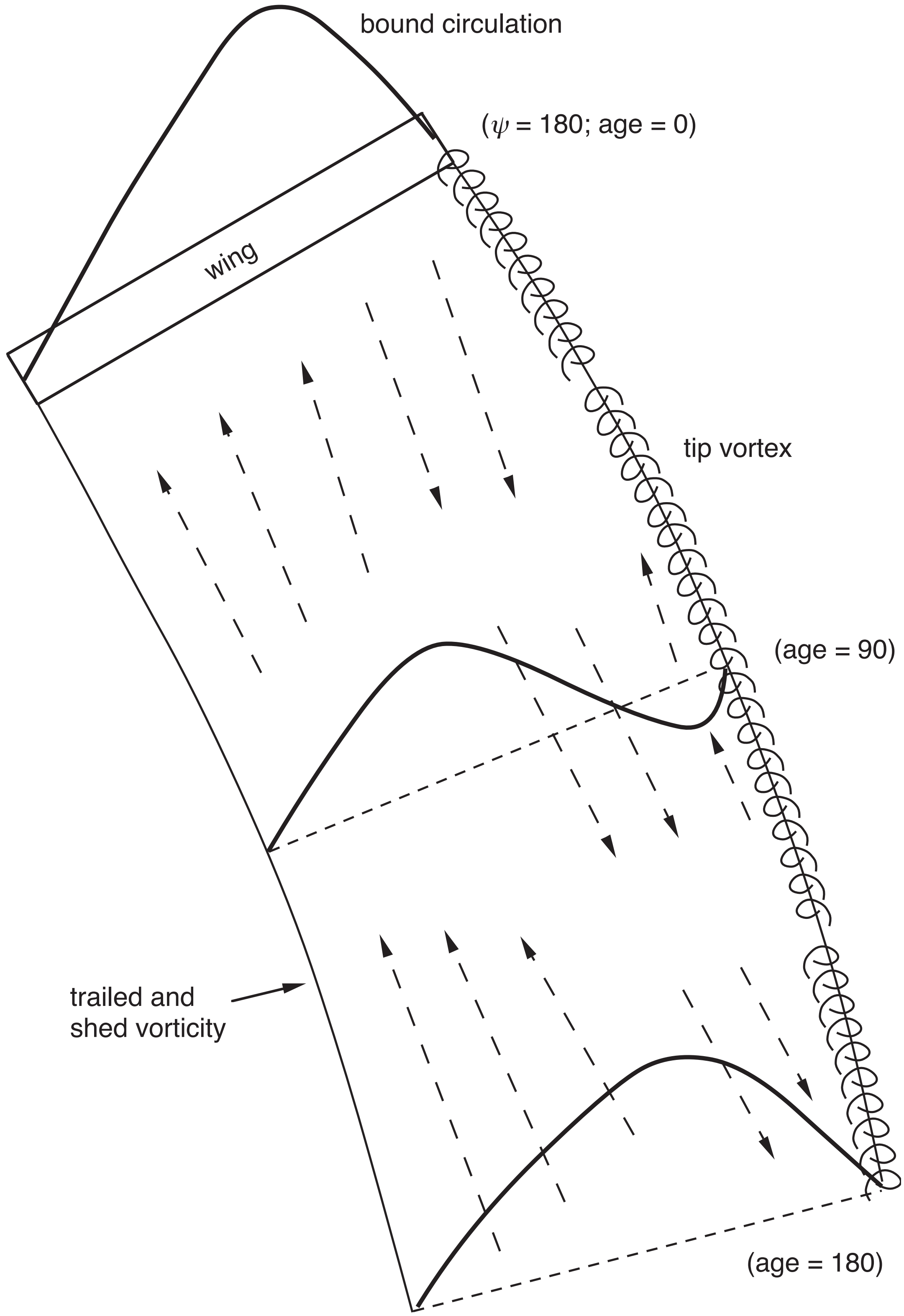


Figure 5-47 Rotor wake rollup structure.

SINGLE-PEAK AND DUAL-PEAK ROLLUP

SINGLE-PEAK CASE

often bound circulation has same sign along entire wing span at a given time

then rolled up wake is constructed solely from magnitude and position of peak bound circulation

spanwise maximum of bound circulation is $\Gamma_{\max}$

in far wake (where rollup process is complete) have rolled-up tip vortices with strength equal to value of $\Gamma_{\max}$ at time that wake element was created

corresponding negative trailed vorticity with total strength $-\Gamma_{\max}$ in inboard sheet

tip vortex modelled as line segment with this strength, and small core radius (input parameter)

any error in assumed strength (because vortex is partially or over rolled-up), is compensated for by value of core radius

detailed structure of inboard vortex sheet not known (no calculation of stretch and rollup of vorticity)

so use single sheet element with trailed and shed vorticity

models of tip vortex entrainment (partial rollup) and stretching of inboard wake implemented

but seldom have enough information to select input parameters

DUAL-PEAK CASE

bound circulation can be positive over part of wing span, and negative over remainder

then rolled up wake constructed from magnitude and position of left and right bound circulation peaks

two peaks of opposite sign (left peak Γ_L and right peak Γ_R) in spanwise distribution of bound circulation

for rotor blade, with only right tip vortex rolled up:

> tip vortex consists of trailed vorticity outboard of right peak, hence strength Γ_R
>
> inboard sheet has both positive and negative trailed vorticity, divided at span station corresponding to left peak

trailed vorticity between two peaks may partially roll up as well

> modelled by using line segment with physically-meaningful core radius

IF SINGLE-PEAK MODEL APPLIED TO DUAL-PEAK CIRCULATION DISTRIBUTION, CAN RESULT IN WRONG SIGN OF TIP VORTEX STRENGTH

wake structure (single-peak or dual-peak) used when influence coefficients are calculated may not correspond to wing loading distribution

> either because solution has not converged or because wrong wake model was specified

figure 48 illustrates possible rotor wake models with dual-peak loading

single-peak model based on Γ_R rather than $\Gamma_{\max}$ at least gets correct sign and magnitude of tip vortex strength

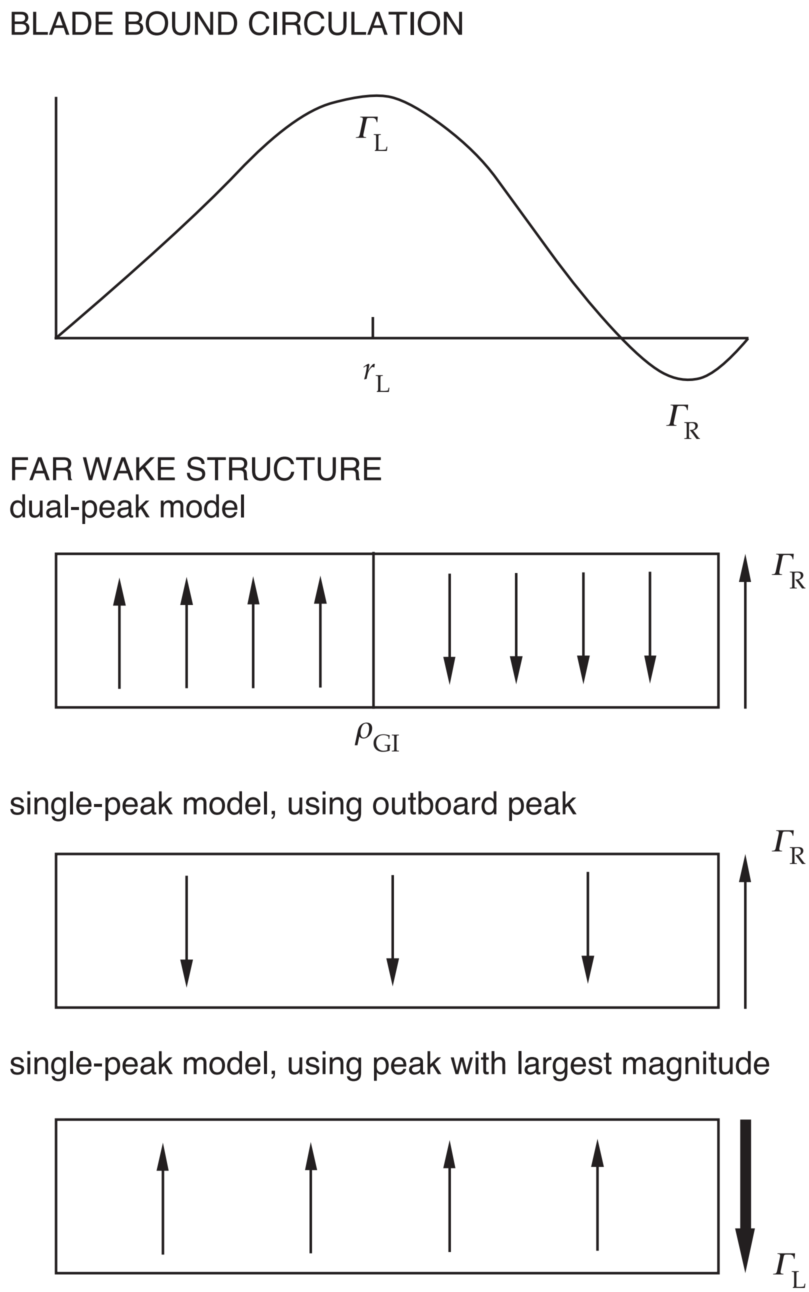


Figure 5-48 Wake models with dual-peak loading.

MULTIPLE-TRAILER WAKE MODEL

FAR WAKE TRAILED VORTICITY

far wake can be divided into several spanwise panels, to provide more detailed structure of inboard vorticity

rollup can occur at boundaries between panels

THERE MAY BE FEATURES OF INBOARD WAKE STRUCTURE THAT MUST BE MODELLED

for example, trailing-edge flap or rapid change in chord can produce step in bound circulation

which might produce significant rollup of inboard vorticity

figure 49 illustrates the wake configuration

CAN SPECIFY SPAN STATIONS OF SUCH INBOARD ROLLUP, BASED ON GEOMETRY OF WING

far wake trailed vorticity then divided into several spanwise panels

single-peak or dual-peak model applied to each panel

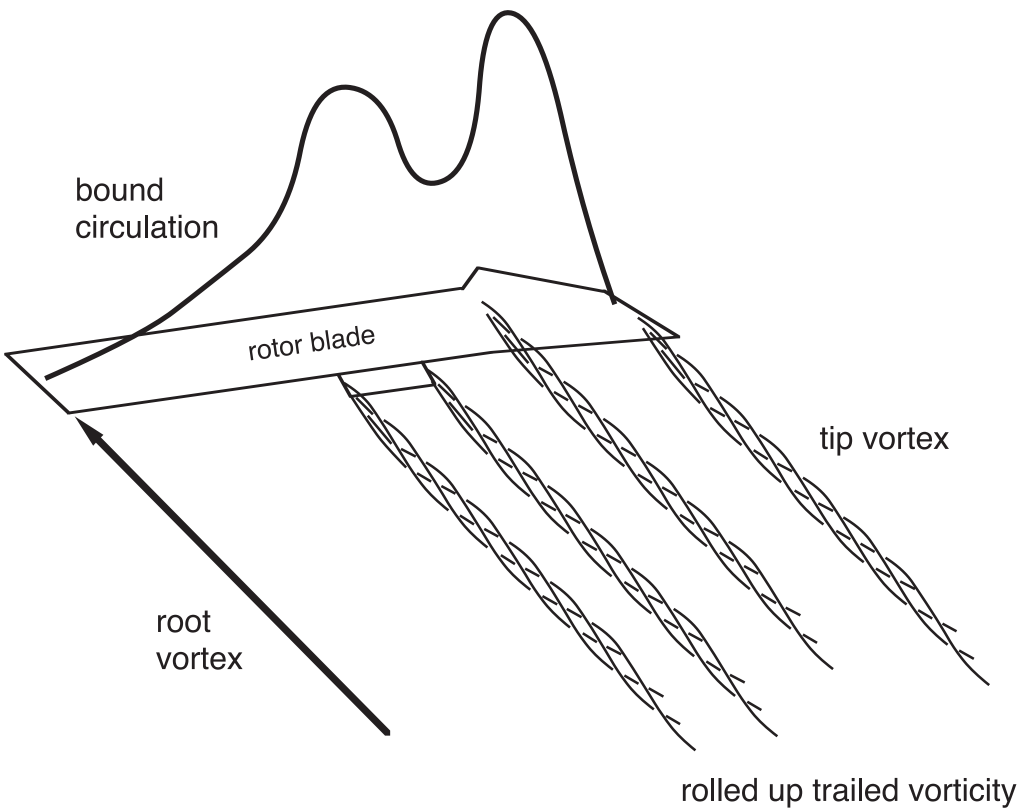


Figure 5-49 Rotor wake with multiple rolled-up trailed vorticity.

ALTERNATIVELY, CAN HAVE TRAILED VORTEX LINE EMANATING FROM EACH AERODYNAMIC PANEL EDGE

rollup is not well calculated even with many trailed vortex lines, because of the coarse discretization and neglect of viscosity

trailed lines can be consolidated into single rolled-up line, using trailed vorticity moment to scale the rate of rollup

figure 50 illustrates the wake model options

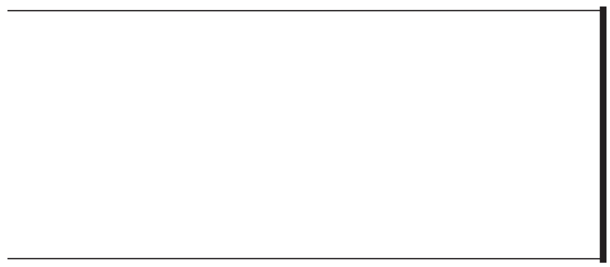
CONSOLIDATION IS SIMULATION OF TIP VORTEX FORMATION PROCESS

implemented in wake geometry calculation,
requires use of wing wake geometry component

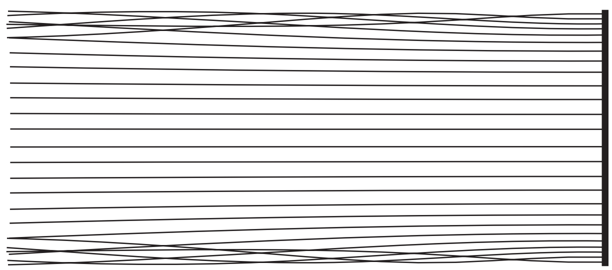
trailed vorticity partitioned into sets of adjacent lines that have same sign (bound circulation increasing or decreasing)

assume that all vorticity in a set eventually rolls up into a single vortex, located at centroid of original vorticity distribution

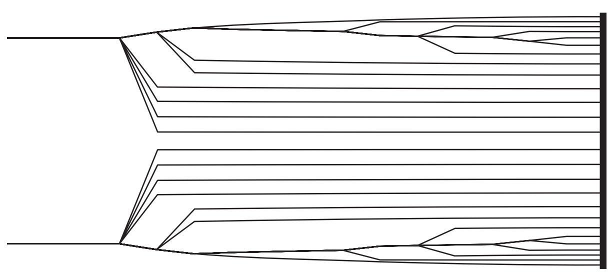
using the total strength and trailed vorticity moment of the set (Γ/r_G^2) to scale the rate of rollup



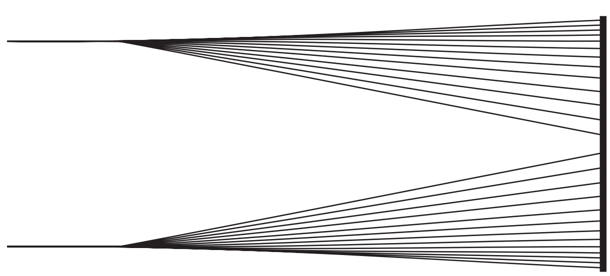
a) rolled-up wake model



b) multiple-trailer wake model



c) multiple-trailer wake model, with consolidation (entrainment form)



d) multiple-trailer wake model, with consolidation (compression form)

Figure 5-50 Illustration of wake models.

5-23.5 Summary of Wake Model

WAKE REGIONS

wake behind a wing divided into near wake and far wake regions

near wake only used for collocation points on wing that generated the wake

elements in near wake always modelled using line segments

extent of near wake should be at least one-half span, or 30 deg azimuth for rotor

far wake used for rest of wake behind each wing

FAR WAKE

tip vortices modelled using line segments

inboard wake modelled using line segments or sheet elements

extent of far wake can be different for collocation points on and off the wing set

for rotor, collocation points on wing set need at least two revolutions of wake, more at low speed

roll up of both tip vortices, or just left or right tip vortex

partial entrainment and partial stretching can be included in rollup process

consists of one or more spanwise panels

single-peak or dual-peak model for each panel

inboard rolled-up trailed vortex structures can be defined based on wing geometry

can use multiple-trailer model with consolidation

ONLY RIGHT TIP VORTEX ROLLED UP

- model for rotor, with counter-clockwise rotation
- for clockwise rotation, span stations still ordered from root to tip

DUAL-PEAK MODEL AND ONLY RIGHT TIP VORTEX ROLLED UP

- inboard (left) peak breaks sheet into two elements at span station r_L (calculated location of left circulation peak)
- trailed vorticity between two peaks may partially roll up
  - modelled by using line segment with physically-meaningful core radius
- wake geometry iteration must produce converged r_L
  - inboard peak position r_{GI} is part of wake geometry description, needed to calculate influence coefficients
  - these influence coefficients used to evaluate wing loading, hence new value of location of left peak, r_L
  - next wake geometry iteration uses $\rho_{GI} = r_L$ to calculate influence coefficients

5–24 Wing Wake Geometry Component

WING WAKE GEOMETRY COMPONENT CALCULATES WAKE GEOMETRY FOR A WING SET

- figure 51 illustrates functionality

WAKE GEOMETRY MODEL

- RIGID, PRESCRIBED, AND FREE WAKE GEOMETRY MODELS
  - rigid geometry
    - calculates distortion from mean interference velocity at wing set
  - prescribed geometry
    - obtained from empirical model for hovering rotor (same as rotor wake geometry component)
      - both rigid and prescribed models considered “rigid wake geometry” for trim wake loop (LEVEL 2)
  - free geometry
    - obtained by calculating distortion simultaneously for all wings in the wing set
    - using wake model of the wing wake component
    - free wake geometry available for trim and transient tasks

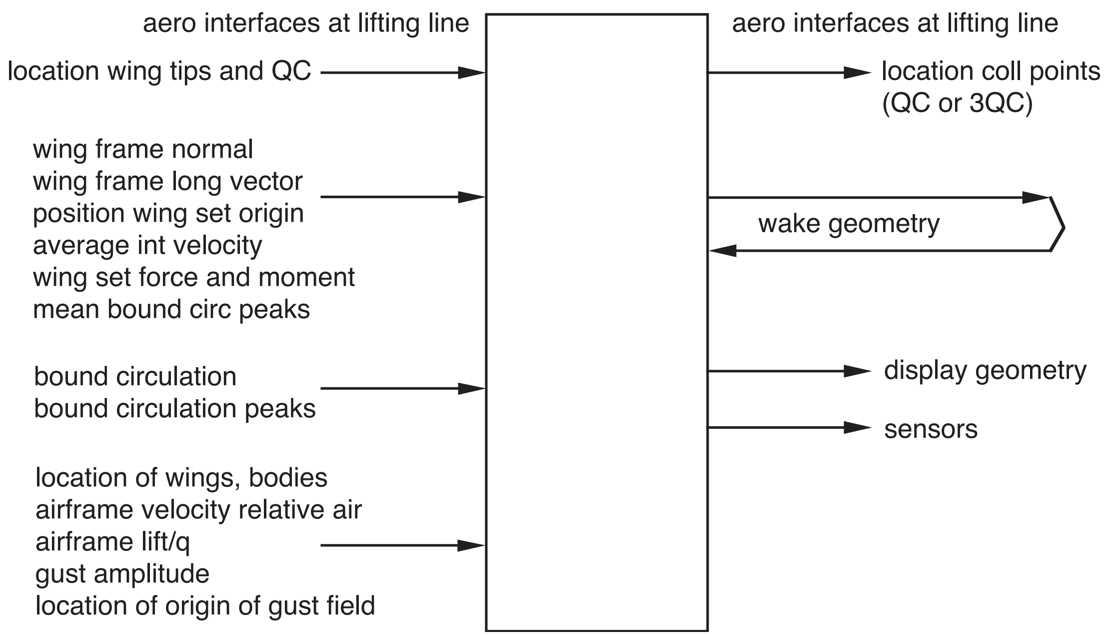


Figure 5-51 Functionality of wing wake geometry component.

WAKE GEOMETRY DESCRIBES POSITION OF WAKE VORTICITY IN SPACE

- undistorted geometry obtained from motion of wing
    - wake element convected by wind, from position in air at which it was created
- this geometry is distorted by wake self-induced velocity

NONUNIFORM INDUCED FLOW AT WAKE SURFACE PRODUCES SIGNIFICANT DISTORTION OF GEOMETRY

- distortion may be unimportant, or adequately represented by convection with mean induced velocity
- wake geometry (including distortion) important when wing or body (perhaps wing that generated wake) passes close to wake
    - wake-induced loading then very sensitive to separation between wings and tip vortices

CHARACTER OF ROTOR WAKE GEOMETRY

ROTOR WAKE GEOMETRY CONSISTS OF DISTORTED, INTERLOCKING HELICES, ONE BEHIND EACH BLADE, SKEWED AFT IN FORWARD FLIGHT

typically only vortex from blade tip rolls up significantly

relative rotor disk, wake is convected downward (normal to disk plane) by mean induced velocity and free stream

and convected aft in forward flight by inplane component of free stream

self-induced velocity produces substantial distortion of vortex filaments as they are convected

DISTORTED WAKE GEOMETRY EXHIBITS AN OVERALL PATTERN IN WHICH EDGES OF WAKE ARISING FROM ROTOR DISK ROLL UP TO FORM VORTICES, AS BEHIND CIRCULAR WING

actually this wake consists of helical tip vortices from individual blades

consequence of this pattern near rotor disk:

tip vortices tend to move upward on sides of disk

tend to move downward in middle of disk

so self-induced distortion moves tip vortices closer to blades on advancing and retreating sides (compared to rigid geometry)

increasing blade-vortex interaction loads

FREE WAKE GEOMETRY HAS LARGE INFLUENCE ON BLADE AIRLOADING AT LOW SPEED

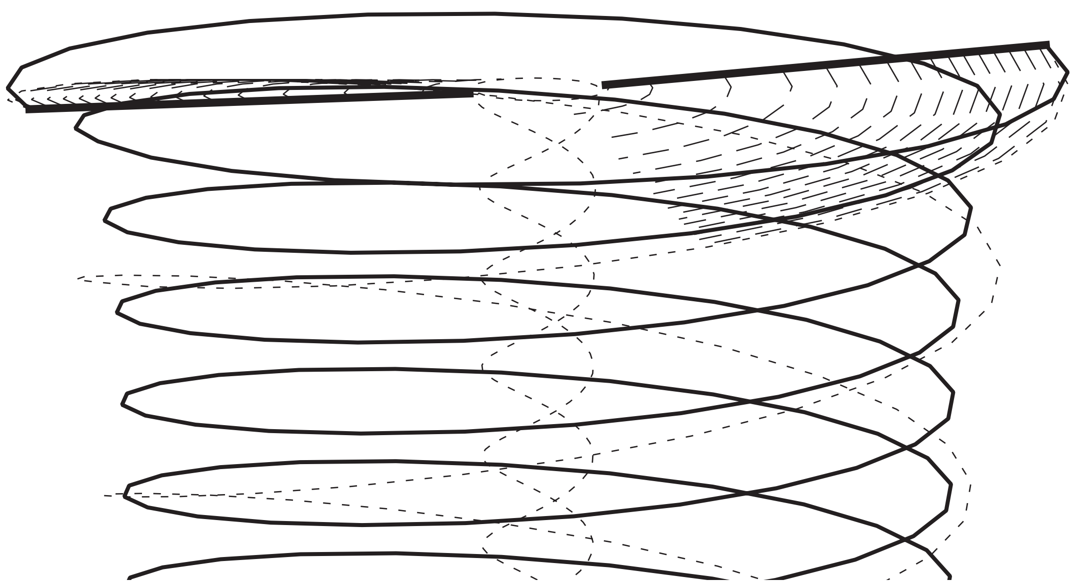
need free wake geometry in analysis for advance ratios below about $\mu = 0.20$ to 0.25

at higher speeds, wake convected quickly downstream and helicopter rotor has large tip-path-plane angle of attack to provide propulsive force

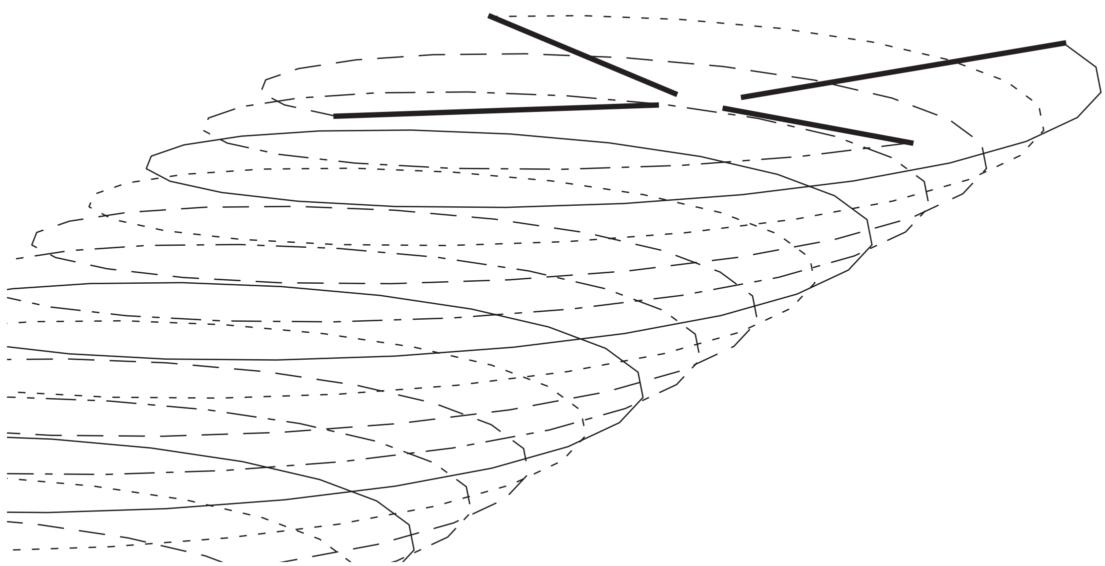
so wake is convected away from disk by normal component of free stream, and distorted geometry is less important

nonuniform inflow at rotor disk still important hover wake geometry, $C_T/\sigma = 0.099$

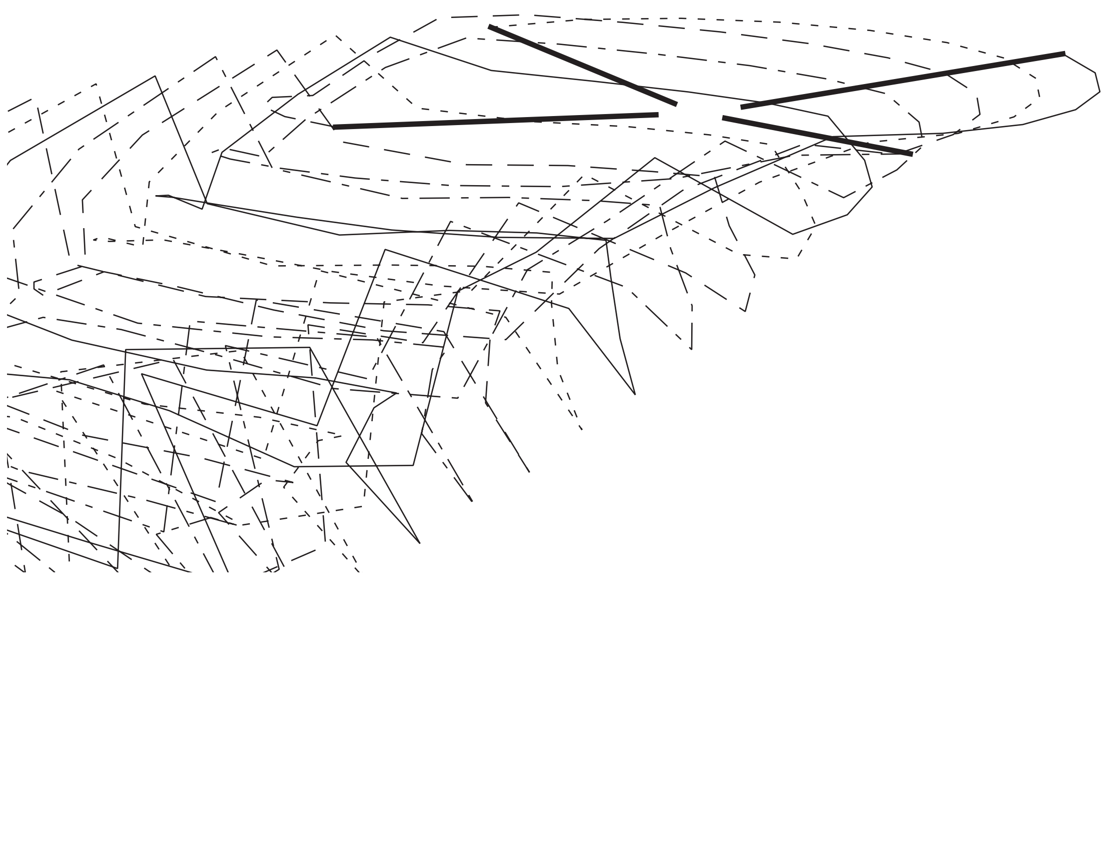
(view: azimuth = 20, elevation = 10)



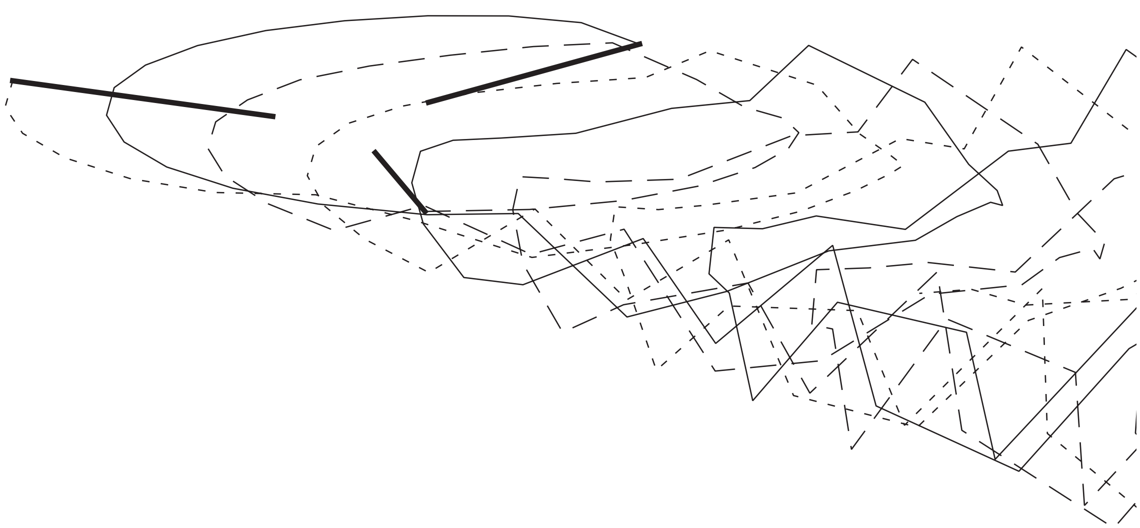
$\mu = 0.1, C_T/\sigma = 0.08$



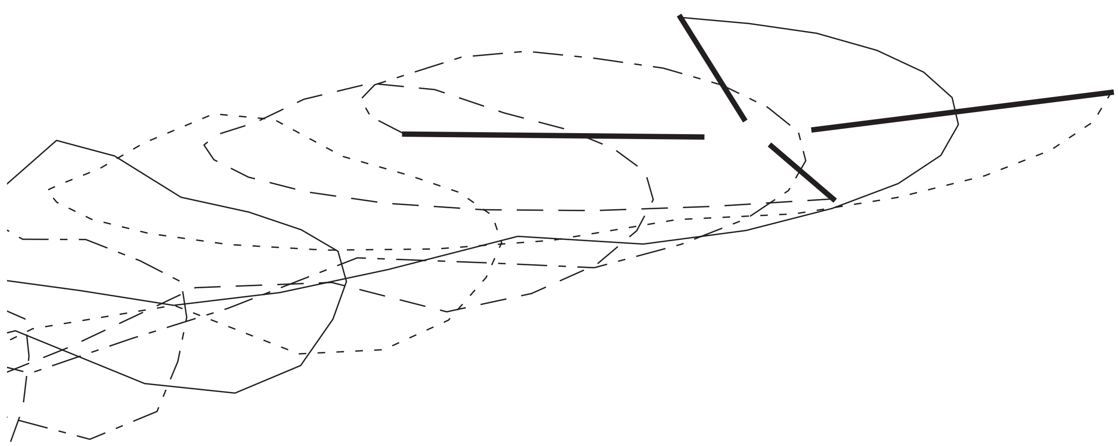
$\mu = 0.1, C_T/\sigma = 0.08$



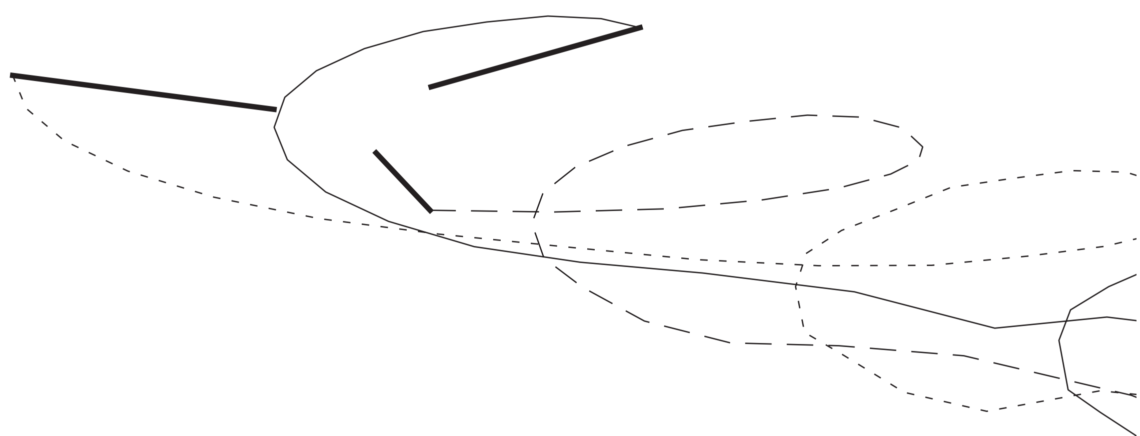
SA-349 Flight Test, $C_T/\sigma = .065$, $\mu = .14$

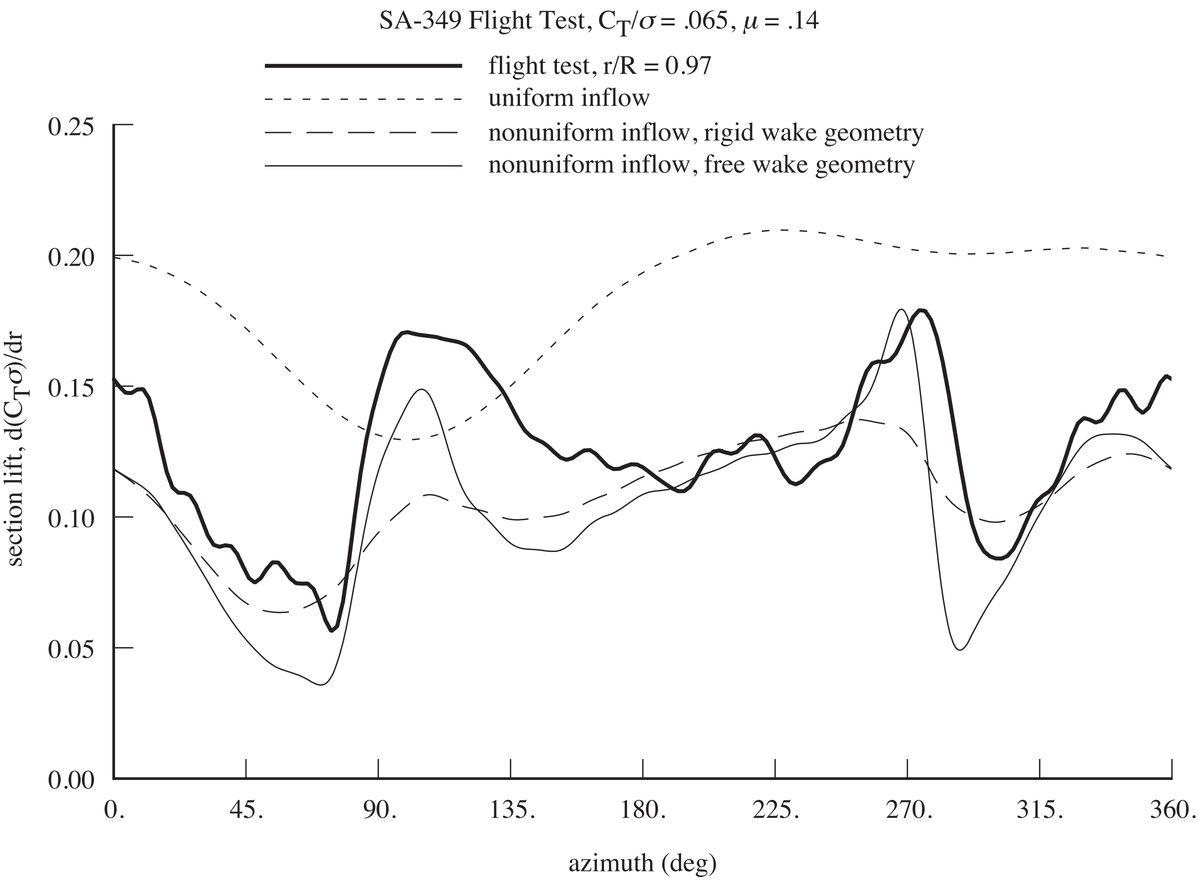


H-34 Flight Test, $C_T/\sigma = .091$, $\mu = .29$

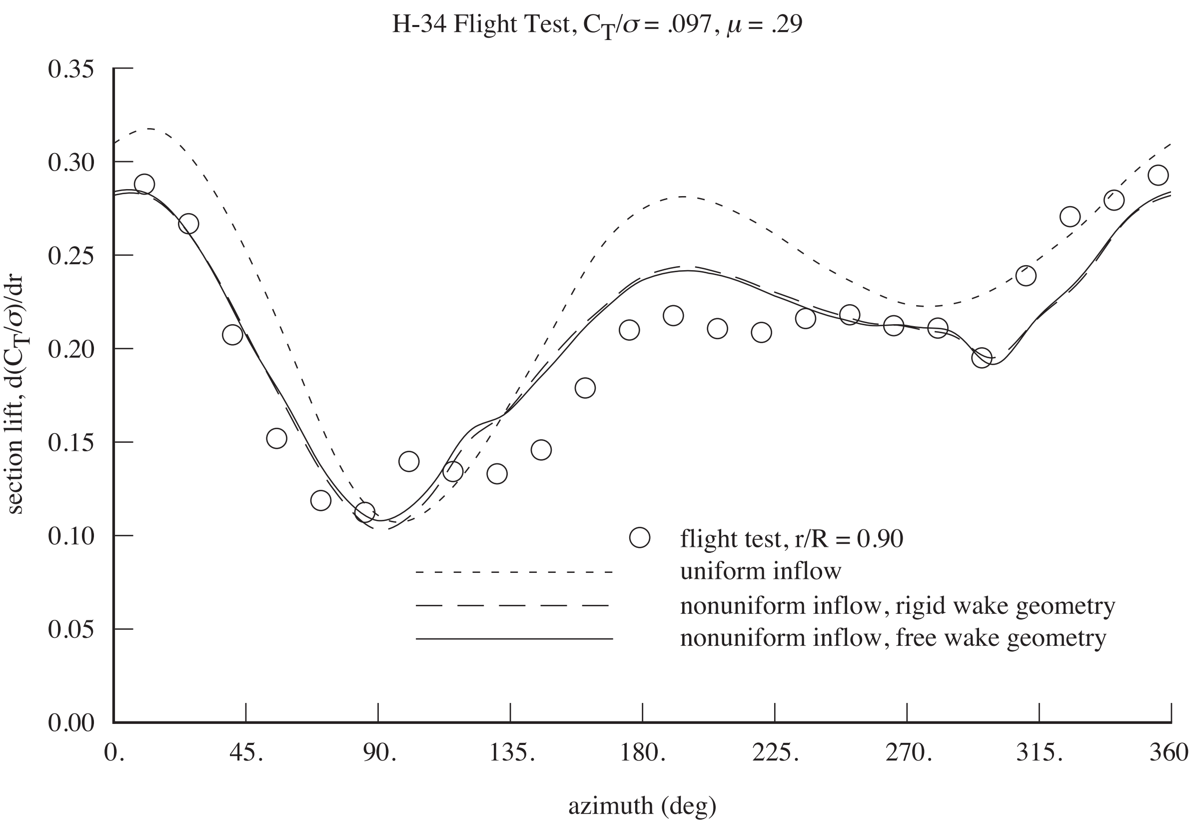


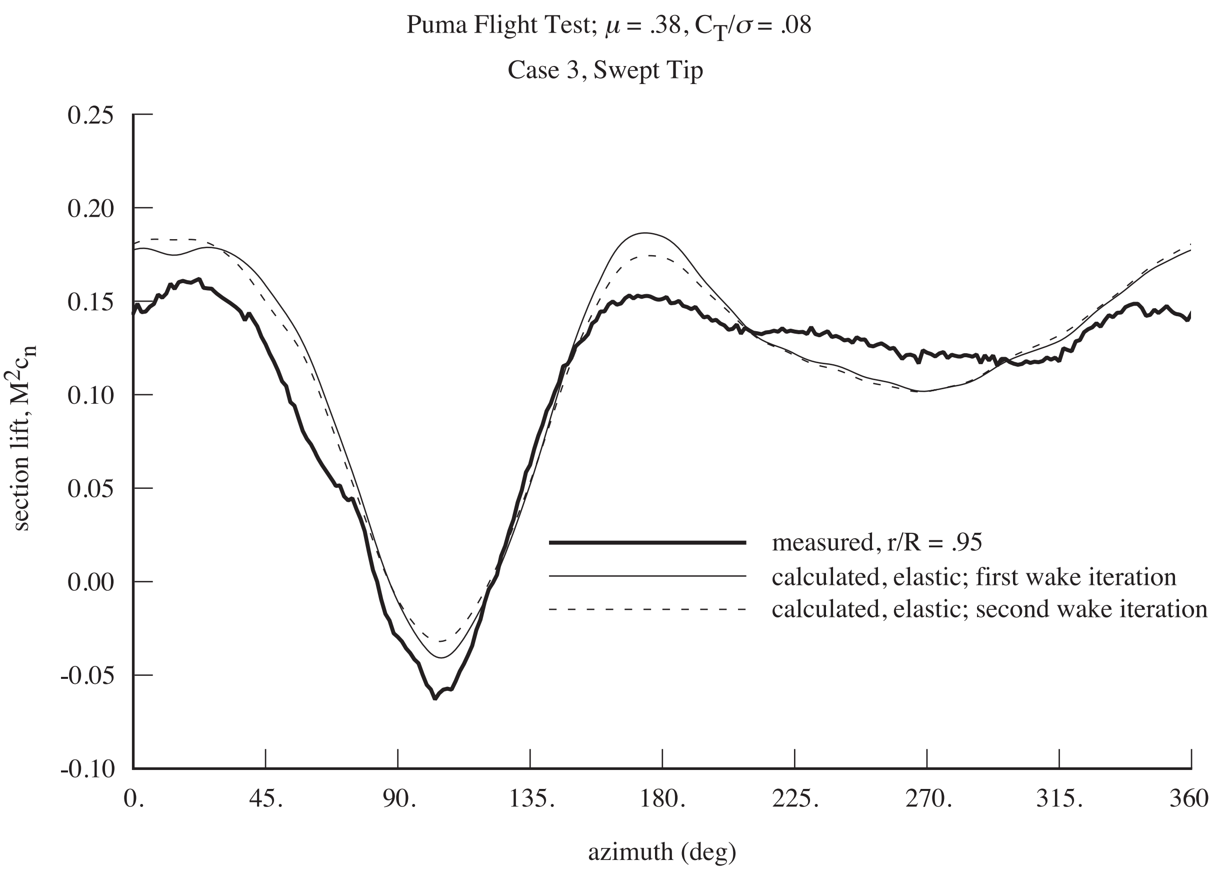
SA-349 Flight Test, $C_T/\sigma = .071$, $\mu = .36$





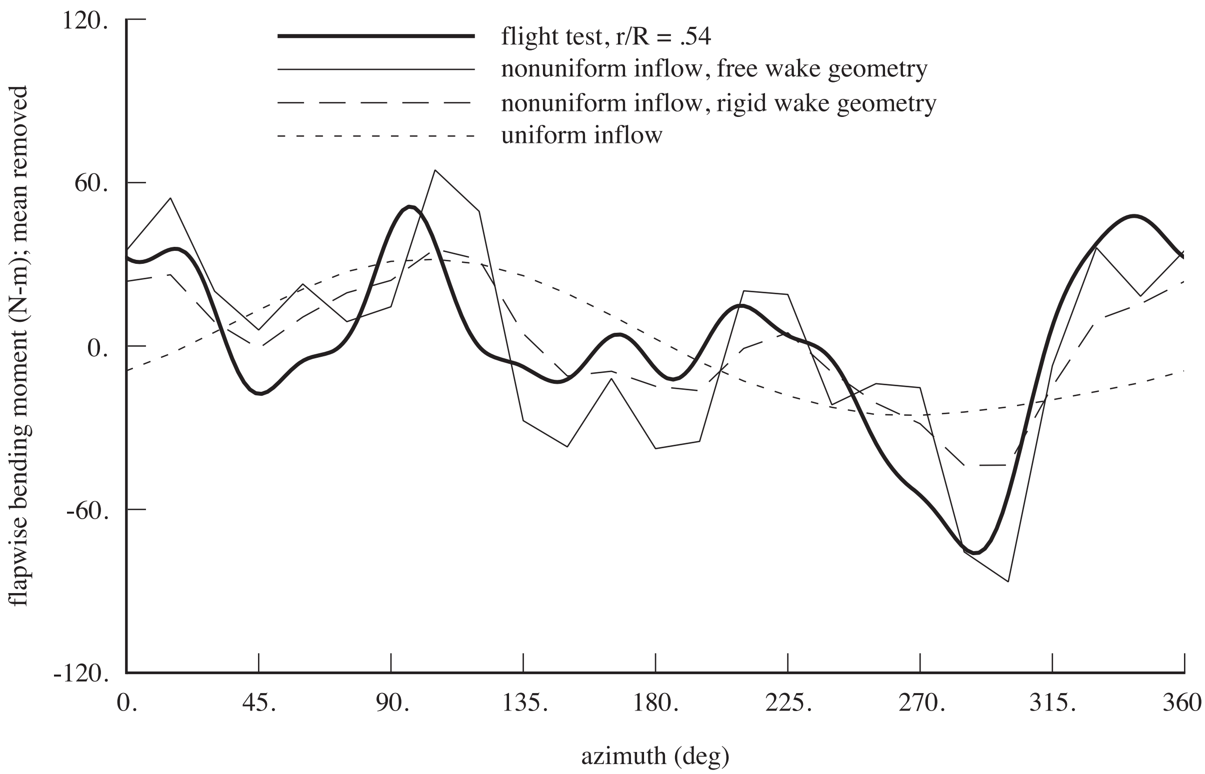
SA-349 Flight Test, C_T/σ = .065, μ = .14
flight test, r/R = 0.97
uniform inflow
nonuniform inflow, rigid wake geometry
nonuniform inflow, free wake geometry
section lift, $d(C_T\sigma)/dr$
0.25
0.20
0.15
0.10
0.05
0.00
0.
45.
90.
135.
180.
225.
270.
315.
360.
azimuth (deg)

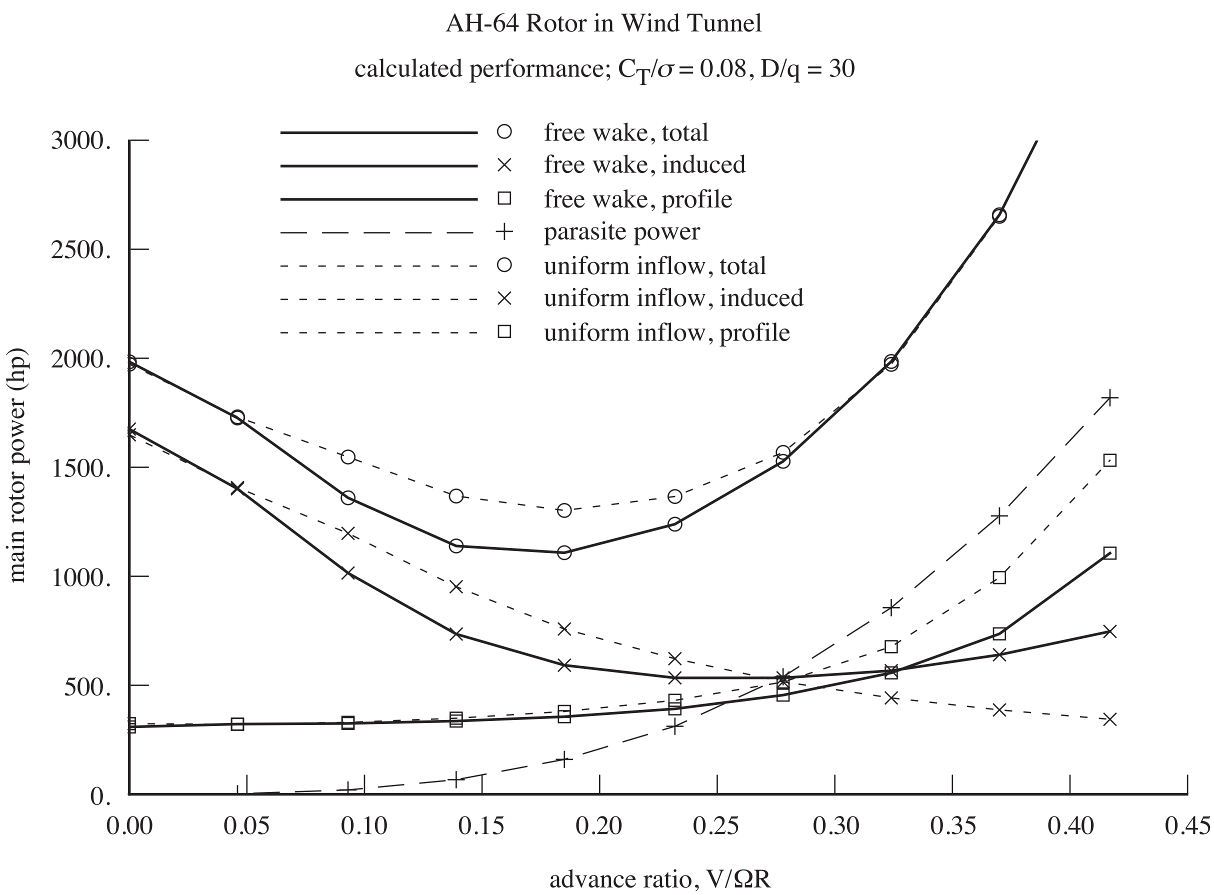

H-34 Flight Test, C_T/σ = .097, μ = .29
section lift, $d(C_T/\sigma)/dr$
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0.
45.
90.
135.
180.
225.
270.
315.
360.
azimuth (deg)
flight test, r/R = 0.90
uniform inflow
nonuniform inflow, rigid wake geometry
nonuniform inflow, free wake geometry



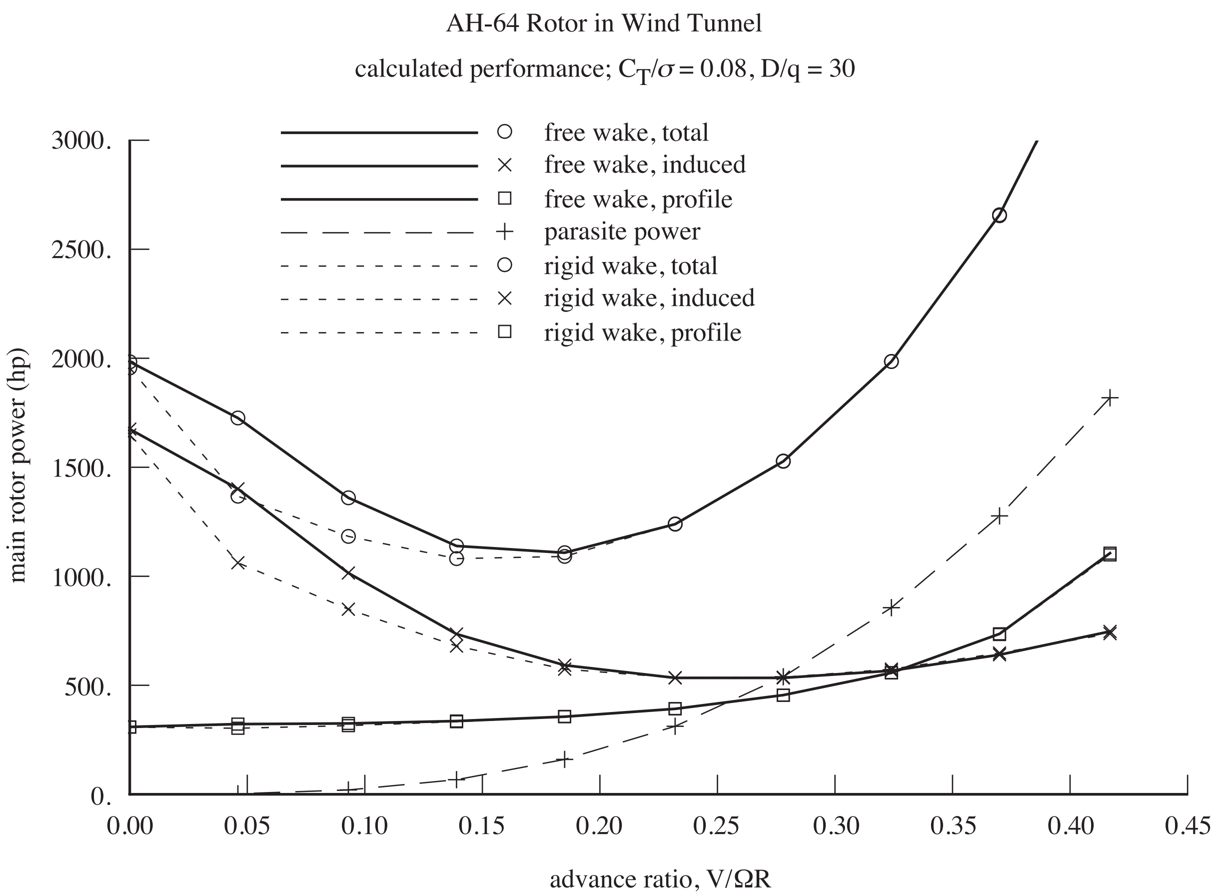
Puma Flight Test; μ = .38, C_T/σ = .08
Case 3, Swept Tip
section lift, M^2c_n
0.25
0.20
0.15
0.10
0.05
0.00
-0.05
-0.10
0.
45.
90.
135.
180.
225.
270.
315.
360.
azimuth (deg)
measured, r/R = .95
calculated, elastic; first wake iteration
calculated, elastic; second wake iteration

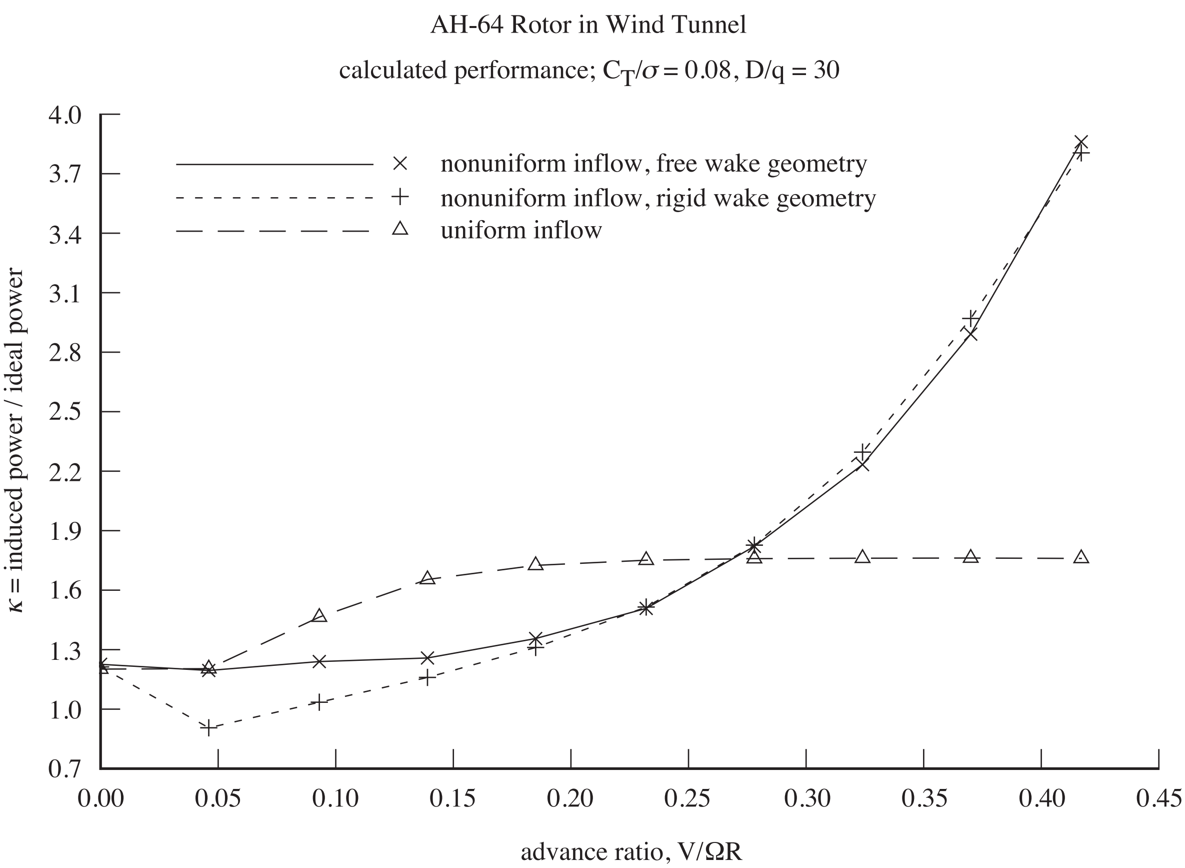
SA-349 Flight Test, $C_T/\sigma = .065$, $\mu = .14$

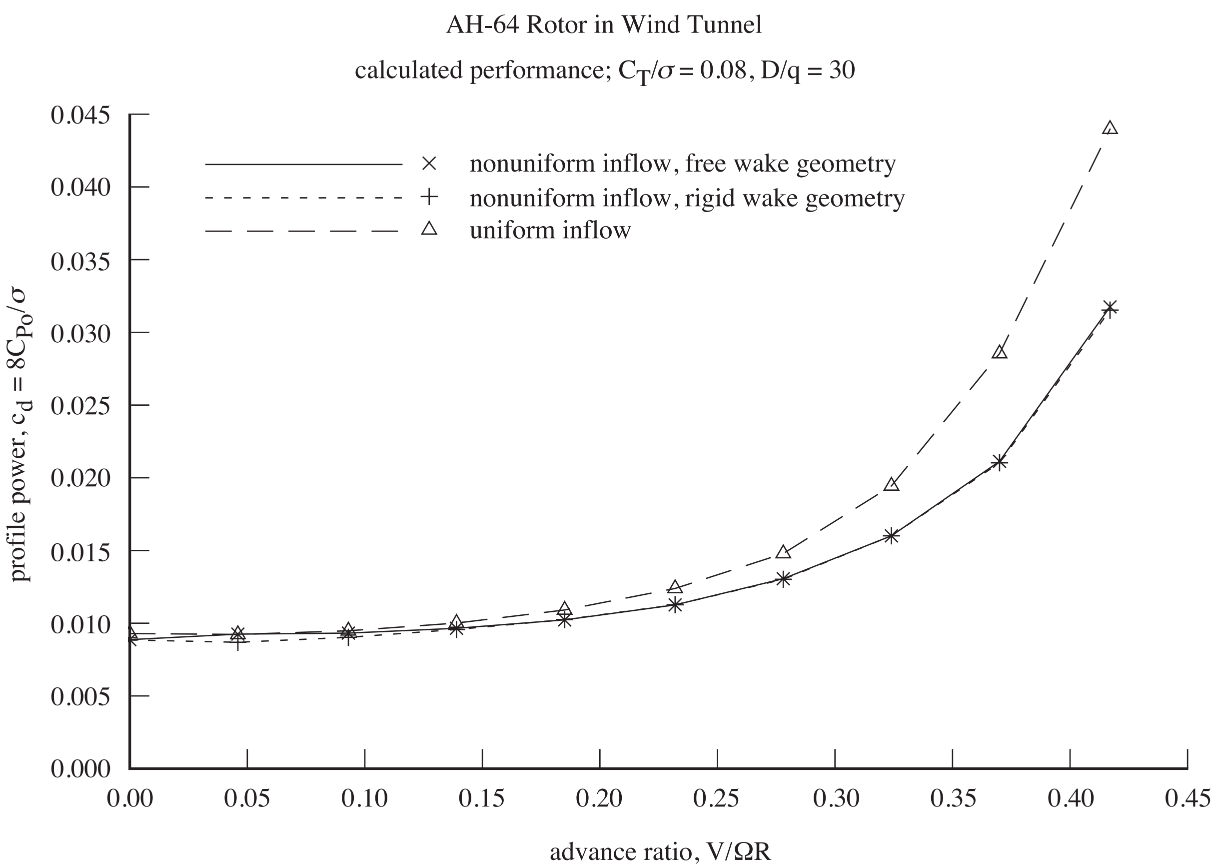




AH-64 Rotor in Wind Tunnel
calculated performance; $C_T/\sigma = 0.08$, D/q = 30
free wake, total
free wake, induced
free wake, profile
parasite power
uniform inflow, total
uniform inflow, induced
uniform inflow, profile
main rotor power (hp)
3000.
2500.
2000.
1500.
1000.
500.
0.
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0.45
advance ratio, V/ΩR


AH-64 Rotor in Wind Tunnel
calculated performance; $C_T/\sigma = 0.08$, D/q = 30
free wake, total
free wake, induced
free wake, profile
parasite power
rigid wake, total
rigid wake, induced
rigid wake, profile
main rotor power (hp)
3000.
2500.
2000.
1500.
1000.
500.
0.
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0.45
advance ratio, V/ΩR


AH-64 Rotor in Wind Tunnel
calculated performance; $C_T/\sigma = 0.08$, D/q = 30
nonuniform inflow, free wake geometry
nonuniform inflow, rigid wake geometry
uniform inflow
κ = induced power / ideal power
advance ratio, V/ΩR
4.0
3.7
3.4
3.1
2.8
2.5
2.2
1.9
1.6
1.3
1.0
0.7
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0.45


AH-64 Rotor in Wind Tunnel
calculated performance; $C_T/\sigma = 0.08$, D/q = 30
nonuniform inflow, free wake geometry
nonuniform inflow, rigid wake geometry
uniform inflow
profile power, $c_d = 8C_{P_0}/\sigma$
advance ratio, V/ΩR
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
0.040
0.045
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0.45

TIP VORTEX FORMATION

EFFECTS ON WAKE GEOMETRY ARE MODELLED

important for hover free wake geometry

INITIAL SPAN STATION

tip vortex can form at span station r_{TV} inboard of wing tip

input parameter

or Betz rollup (calculated assuming centroid of rolled up trailed vorticity is conserved)

INITIAL CONVECTION

with three-dimensional wing, Kutta condition requires that wake leave trailing-edge tangent to wing surface

extent (in wake age) of this initial convection is input parameter

WAKE GEOMETRY DISTORTION

RIGID GEOMETRY

wake elements all convected by average interference velocity

with empirical corrections factors for normal and inplane convection

PRESCRIBED GEOMETRY

based on empirical model for hovering rotor

effects of turn rate or transient motion of rotor included in undistorted geometry, but not in prescribed distortion

interference from other aerodynamic components is not considered

distortion is described by two-stage vertical convection D_z and exponential spanwise contraction D_r:

$$D_z = \begin{cases} K_1\phi & \phi < \phi_1 \\ K_1\phi_1 + K_2(\phi - \phi_1) & \phi > \phi_1 \end{cases}$$

$$D_r = (1 - e^{-K_3\phi})(1 - K_4)$$

$\phi = \Omega\tau$ is dimensionless wake age

$\phi_1 = 2\pi/N$ is age at encounter with following blade (N = number of blades)

PRESCRIBED MODELS IMPLEMENTED

RIGID: convection by normal interference velocity

$K_1 = f_1\lambda$ and $K_2 = f_2\lambda$, with input values of contraction constants K_3 and K_4

EMPIRICAL MODEL FOR HOVERING ROTOR: equations for K_1, K_2, K_3, K_4

Landgrebe model, or Kocurek and Tangler model

from thrust coefficient C_T, or from peak bound circulation

equations depend on N, σ, θ_{tw}

INPUT K_1, K_2, K_3, K_4

FREE GEOMETRY

distortion calculated simultaneously for all wings in the wing set

wing set can include wings of several rotors, and nonrotating wings

using wake model of the wing wake component

including models for distributed vorticity in vortex core, and core radius growth

only far wake region of wing wake component considered for wake geometry calculation

far wake trailed vorticity can be divided into several spanwise panels

wake can roll up at boundaries between panels

wake geometry can be calculated optionally for tip vortices; for all rolled up trailed vortices; and for inboard sheet edges as well

rigid distortion used where free distortion not calculated

with multiple-trailer wake model, trailed lines can be consolidated into a single rolled-up line using the trailed vorticity moment (Γ/r_G^2) to scale the rate of rollup free wake geometry available for trim and transient tasks

free distortion calculated for entire period at first time step of solution for wake geometry in trim task

or at next time step in transient task

distortion calculated using time and wake age increment of wake geometry component

distortion is interpolated or extrapolated as required to times of solution procedure, and wake ages of wake component

WIND AND GUST

variation of wind velocity in ground boundary layer, and gust velocities, can be included in convection term of distortion

GROUND EFFECT

influence of ground can be included in free wake geometry calculation, through use of image elements in wake model

AIRFRAME FLOW FIELD INFLUENCE

calculated using simple model of airframe flow field component

set of wings and bodies, producing velocity perturbations that distort the wake geometry

mutual interference between wake elements and airframe flow field elements not included

5–25 Rotor Wake Geometry Component

ROTOR WAKE GEOMETRY COMPONENT CALCULATES WAKE GEOMETRY FOR A ROTOR

- figure 52 illustrates functionality

WAKE GEOMETRY MODEL

- RIGID, PRESCRIBED, AND FREE WAKE GEOMETRY MODELS
  - rigid geometry
    - calculates distortion from mean interference velocity at wing set
  - prescribed geometry
    - obtained from empirical model for hovering rotor
      - both rigid and prescribed models considered “rigid wake geometry” for trim wake loop (LEVEL 2)
  - free geometry
    - obtained by calculation for helicopter rotor
      - using Scully’s method or Johnson’s method
    - free wake geometry generated only for tip vortex of single rotor
    - simplified wake model
    - free wake geometry only available for trim task
- JOHNSON METHOD IS SPECIALIZED VERSION OF WING WAKE GEOMETRY COMPONENT

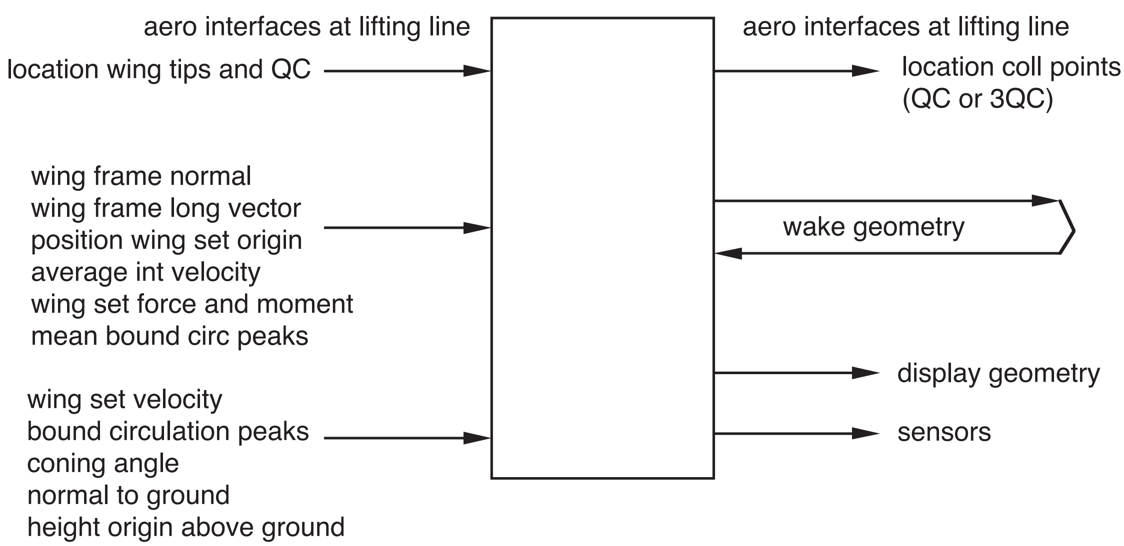


Figure 5-52 Functionality of rotor wake geometry component.

WAKE GEOMETRY DISTORTION

FREE GEOMETRY

Scully method or Johnson method

distortion calculated only for tip vortex of single rotor

- rigid or prescribed geometry model used for distortion of inboard sheet
- constant tip vortex core radius, with Scully vorticity distribution
- effects of turn rate included in undistorted geometry, but not in free distortion
- interference from other aerodynamic components is not considered
- distortion calculation assumes periodicity of wake geometry, hence model can only be used in trim task

free distortion calculated for entire rotor period at first time step of solution for wake geometry

distortion calculated using internally generated time and wake age increment (typically 15 degrees in azimuth)

- distortion is interpolated or extrapolated as required to times of solution procedure, and wake ages of component

FREE WAKE GEOMETRY MODELS AT VERY LOW SPEED

Scully method applicable only in forward flight

- good performance and airloads results at $\mu = 0.125$ and above
- results unrealistic at lower μ

Johnson method can be used in both forward flight and hover

- simplified version of wing wake geometry component
- baseline parameters give good performance and airloads results at $\mu = 0.125$ and above
- good performance results at lower advance ratios using appropriate parameters
- can include influence of ground

5–26 Performance Components

WING PERFORMANCE COMPONENT CALCULATES
PERFORMANCE QUANTITIES FOR ONE NONROTATING WING

figure 53 illustrates functionality

ROTOR PERFORMANCE COMPONENT CALCULATES
PERFORMANCE QUANTITIES FOR A ROTOR

figure 54 illustrates functionality

ROTORCRAFT PERFORMANCE COMPONENT CALCULATES
PERFORMANCE QUANTITIES FOR A ROTORCRAFT

figure 55 illustrates functionality

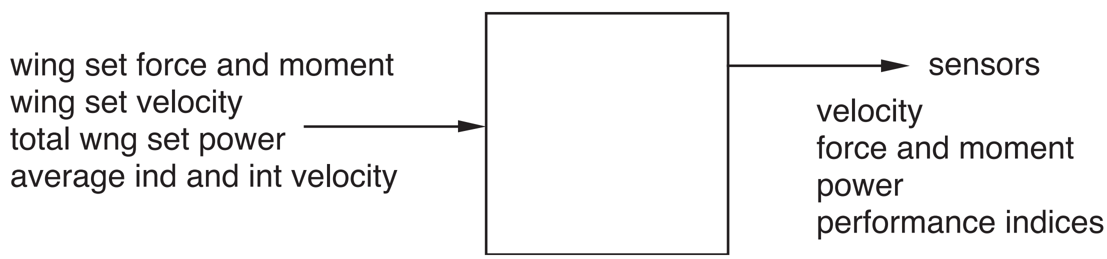


Figure 5-53 Functionality of wing performance component.

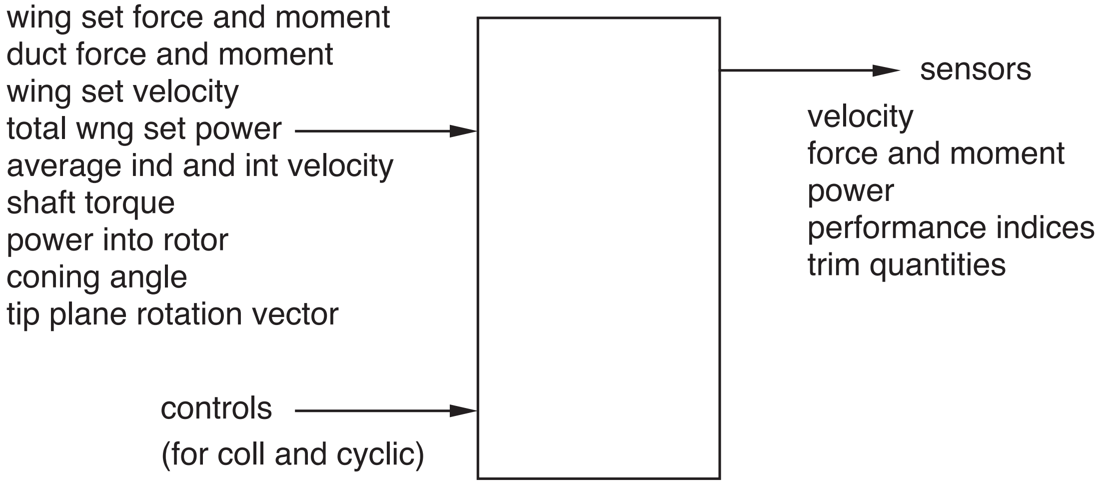


Figure 5-54 Functionality of rotor performance component.

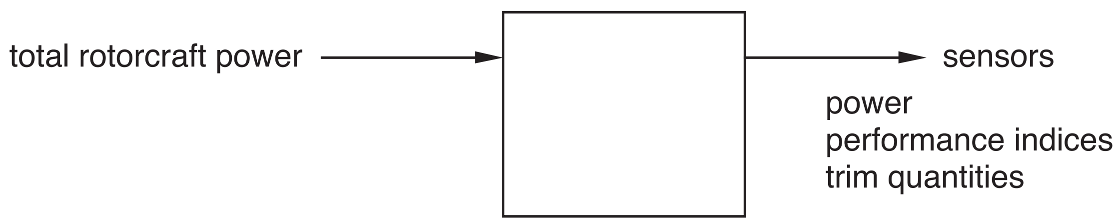


Figure 5-55 Functionality of rotorcraft performance component.

5–27 Rigid Wing Component

RIGID WING COMPONENT COMBINES RIGID BODY COMPONENT AND LIFTING LINE WING COMPONENT

STRUCTURAL DYNAMIC COMPONENT, WITH INTERNAL AERODYNAMIC MODEL

no elastic motion

internal calculation of velocity, position, and force on wing (not implemented as interfaces)

COMPONENT TRADES CONFIGURATION FLEXIBILITY FOR COMPUTATIONAL EFFICIENCY

figure 56 illustrates the combination

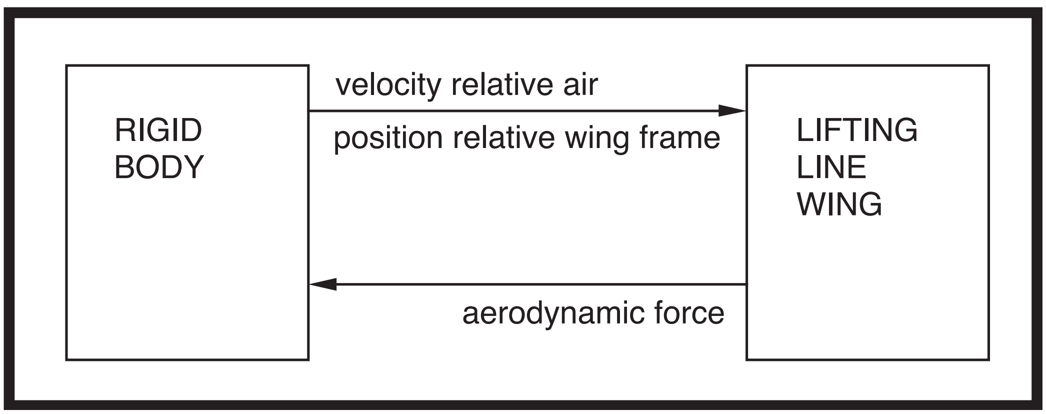


Figure 5-56 Rigid wing as combination of rigid body and lifting line wing.

5–28 Helicopter Tail Boom Component

HELICOPTER TAIL BOOM COMPONENT CALCULATES AERODYNAMIC FORCES ACTING ON CIRCULATION-CONTROL BOOM AND REACTION JET

- CIRCULATION-CONTROL BOOM OPERATES IN WAKE OF MAIN ROTOR
  - intended to provide torque reaction and yaw control for single main-rotor helicopter
    - figure 57 illustrates functionality

- COMPONENT IMPLEMENTS USER-DEFINED CALCULATIONS
  - basic version is simple model of tail boom
  - other versions constructed by modifying subroutines called by helicopter tail boom component

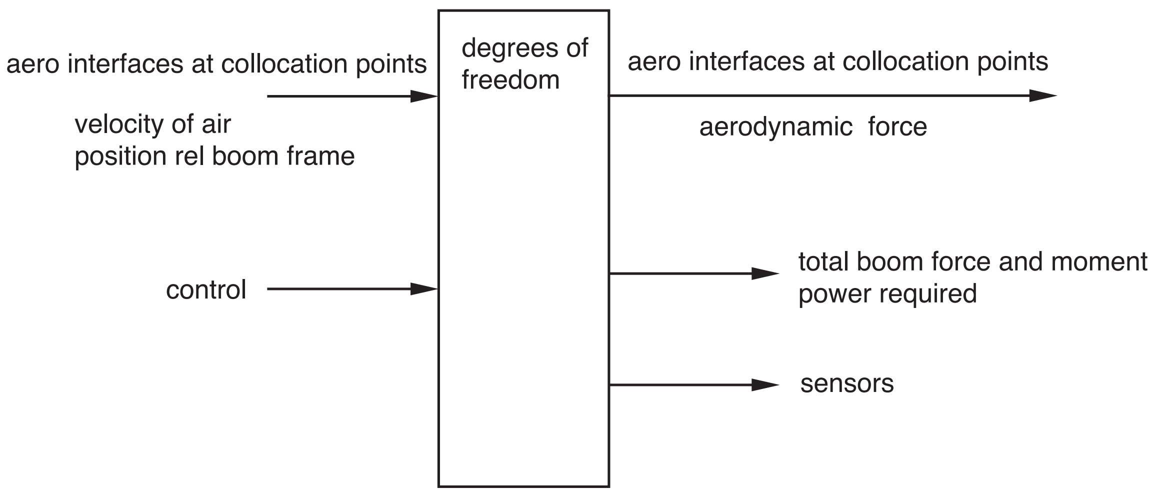


Figure 5-57 Functionality of helicopter tail boom component.

MODEL

AERODYNAMIC MODEL OF BOOM IS LIFTING-LINE THEORY

includes approximate induced angle-of-attack calculation based on ideal-wing theory for boom

component typically obtains aerodynamic coefficient data from tables

simple model of the basic version uses a two dimensional table: lift and drag coefficient as function of angle of attack and blowing coefficient

user-defined calculations can make use of any table class and type

REACTION JET MODELLED AS FORCE WITH SPECIFIED DIRECTION AND MAGNITUDE

COMPONENT DEGREES OF FREEDOM MAY BE NEEDED TO ACCOUNT FOR BEHAVIOR OF INTERNAL AERODYNAMICS

component equations can be static, or first-order differential equations

5–29 Computational Fluid Dynamics Component

COMPUTATIONAL FLUID DYNAMICS COMPONENT CALCULATES AERODYNAMICS OF A SET OF OBJECTS

- USER PROVIDES CFD ANALYSIS, BY MODIFYING SUBROUTINES CALLED BY COMPONENT
  - CFD analysis can be viscous or inviscid
  - general interface with CFD analysis implemented
    - figure 58 illustrates functionality
    - figure 59 shows the operation of the interface routines

AERODYNAMIC SYSTEM CONSISTS OF WINGS AND OTHER OBJECTS

BASIC VERSION OF COMPONENT

- can be used to test component, and used if component is being perturbed
- for basic version, wings modelled as in lifting line wing component (without states, dynamic stall, or prescribed coefficient increments)

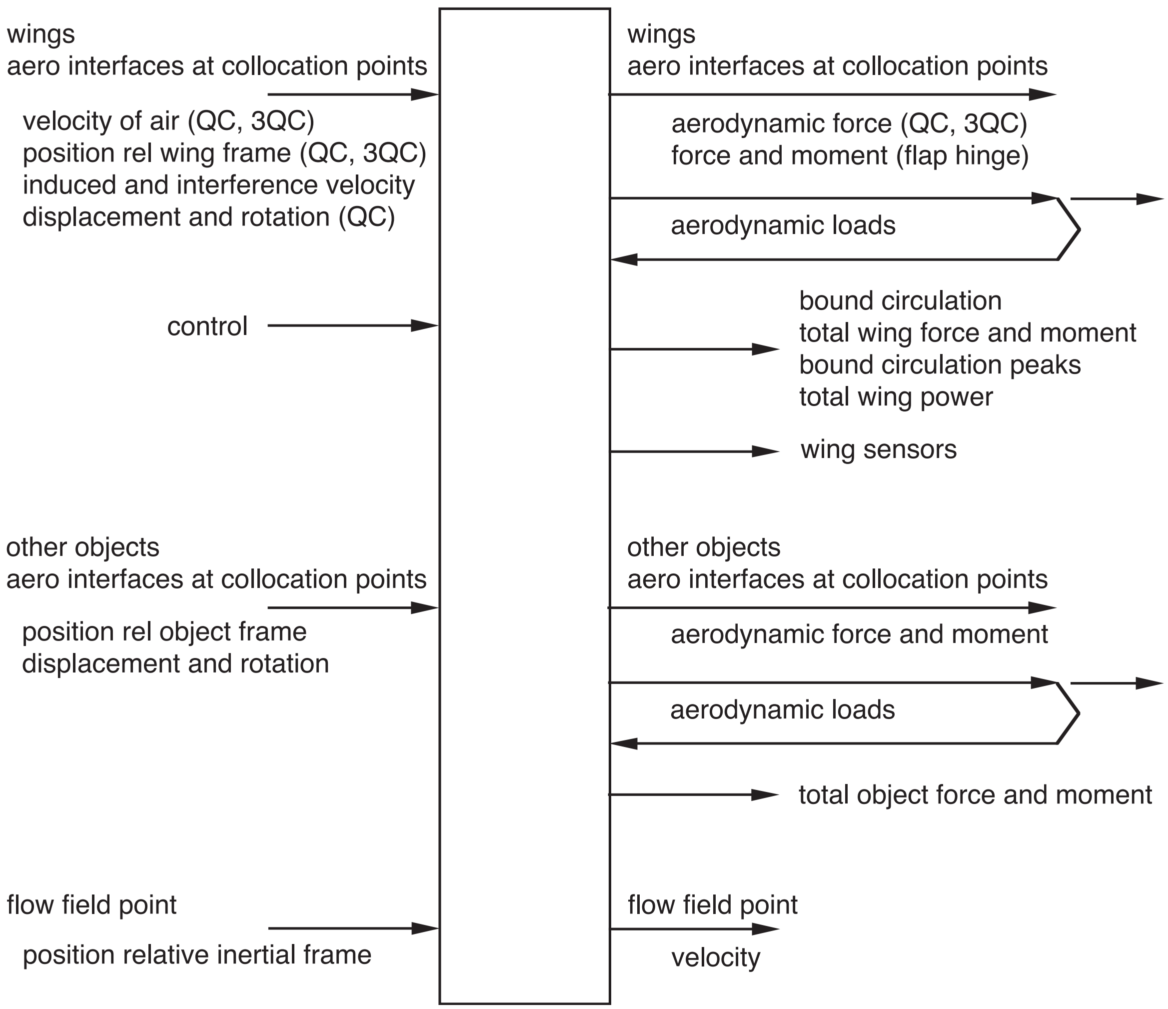


Figure 5-58 Functionality of computational fluid dynamics component.

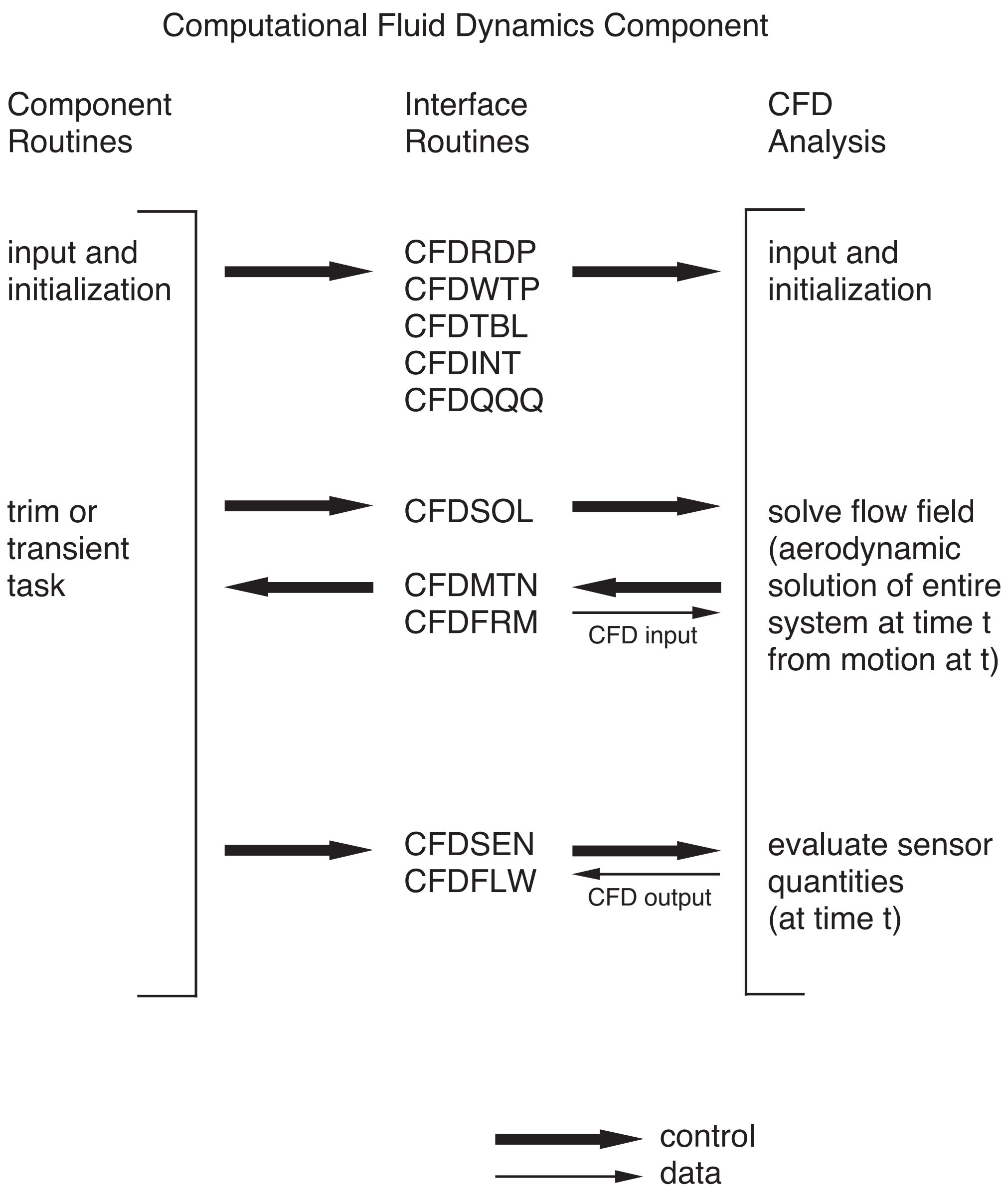


Figure 5-59 Operation of interface routines.

5–30 Plugin Component

THE PLUGIN COMPONENT ALLOWS SOFTWARE DEVELOPERS AND USERS TO ADD CONTENT TO CAMRAD II

PLUGINS CAN ADD TO FUNCTIONALITY OF BOTH SHELL AND COMPONENTS

- SHELL PLUGINS CAN BE USED INDEPENDENTLY OF PLUGIN COMPONENTS
  - extend capability of shell to construct core input
- PLUGIN COMPONENTS CAN BE USED WITH CORE INPUT ENTIRELY
  - but corresponding shell plugin is recommended, to make it easy to use the new component technology

PLUGIN COMPONENT CAN NOT BE A STRUCTURAL DYNAMICS COMPONENT

Chapter 6

ROTORCRAFT THEORY

GENERAL REFERENCE FOR CHAPTER: CAMRAD II Documentation, Volume III, Rotorcraft Theory

6–1 Rotorcraft Shell

CONFIGURATION CONSTRUCTED BY SHELL

ROTORCRAFT: AIRCRAFT WITH ONE OR TWO OR MORE ROTORS

configurations: one rotor, single main-rotor and tail-rotor helicopter, tandem helicopter, coaxial helicopter, tilting proprotor aircraft, multirotor aircraft

in free flight or in wind tunnel

figure 1 is simplified description of model

ROTOR: ARBITRARY N-BLADED ROTOR

hub and blade root configurations: articulated, hingeless, teetering, gimballed, and bearingless

control configurations: swashplate mechanism (with pitch horns and pitch links), or blade root pitch

optionally with higher harmonic control

aerodynamic model includes a sophisticated wake analysis to calculate the rotor nonuniform induced-velocities; using rigid, prescribed, or free wake geometry

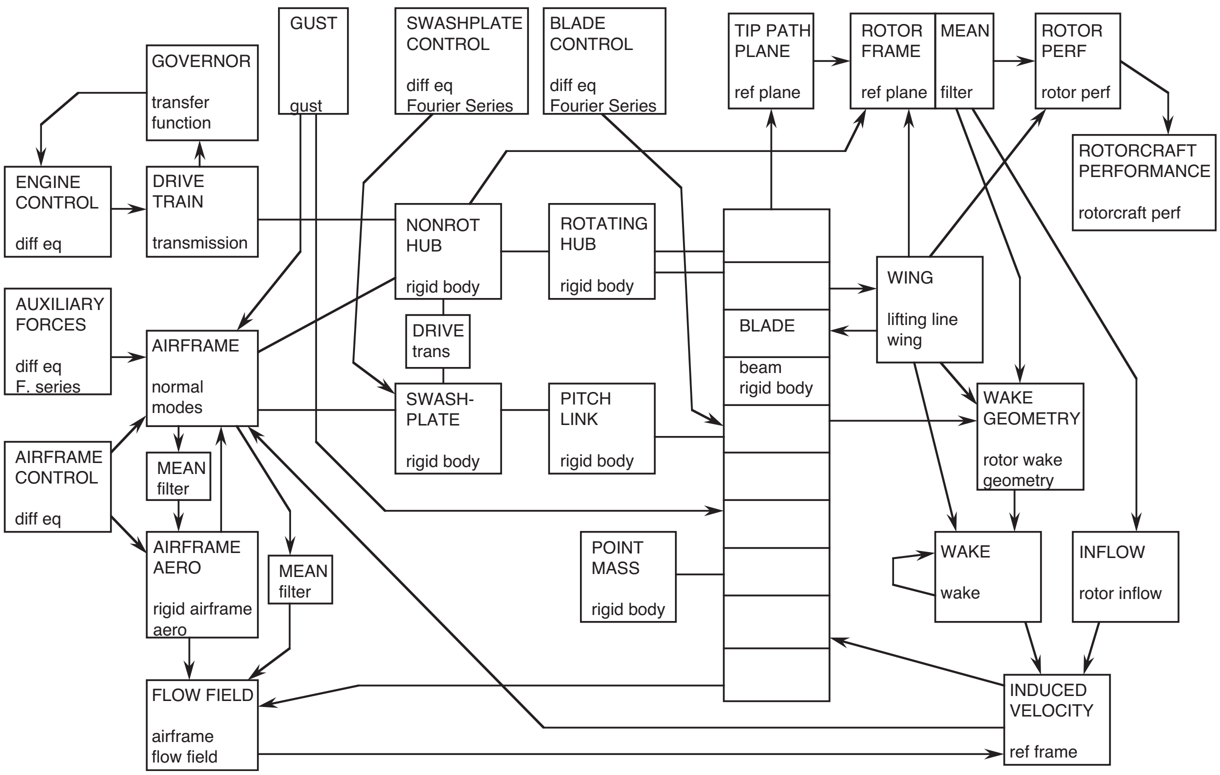


Figure 6-1 Rotorcraft model (simplified).

MODEL CONSTRUCTED BY SHELL

ROTOR STRUCTURAL DYNAMICS

beam theory, with large pitch and twist; isotropic material with elastic axis, or anisotropic material; arbitrary beam axis (straight within beam segments); arbitrary hinge order; exact kinematics of hinges and bearings; sweep and droop of tip; point masses on blade

ROTOR AERODYNAMICS

lifting line theory (steady two-dimensional airfoil characteristics plus vortex wake); empirical dynamic stall models; unsteady aerodynamic forces from thin airfoil theory; yawed-flow and swept-blade corrections; sweep and droop of tip

induced velocity from momentum theory or vortex wake model

momentum theory includes mean and linear variation over rotor disk (produced by forward flight and hub moment); rotor-on-rotor and rotor-on-airframe interference models; and dynamic inflow model (perturbation momentum theory or actuator disk theory) or dynamic wake model

ROTOR WAKE

vortex lattice, with small viscous core for tip vortices; nonplanar, quadrilateral sheet elements available for inboard wake; second-order lifting line theory (three-quarter chord collocation point)

wake rollup process modelled, including possibility of two bound circulation peaks (inboard and outboard peaks of opposite sign); and simulation of rollup process in wake geometry calculation (consolidation) rotor-on-rotor interference; mean induced-velocity at airframe; induced-velocity at arbitrary point in space

wake geometry models: undistorted (rigid); hover prescribed; calculated free wake

AIRFRAME

elastic airframe modes; drive train model; auxiliary forces (constant and higher harmonic terms)

airframe aerodynamic forces from nonlinear or linearized (stability derivative) model, from equations or tables; airframe-to-rotor interference velocity, from simple model or table

SOLUTION PROCEDURE CONSTRUCTED BY SHELL

ITERATIVE TRIM SOLUTION

wake loop, with levels: uniform inflow; nonuniform inflow with prescribed wake geometry; nonuniform inflow with free wake geometry

trim loop (modified Newton-Raphson solution); circulation and motion loops (successive substitution solution)

part solutions: arbitrary time step; harmonic solution method for rotor, airframe, and drive train parts (differential equations), optionally with modal transform

CONSTRUCT IDENTICAL, EQUALLY-SPACED BLADES

for efficiency, trim analysis can assume that all blades have same motion, so only necessary to analyze one blade

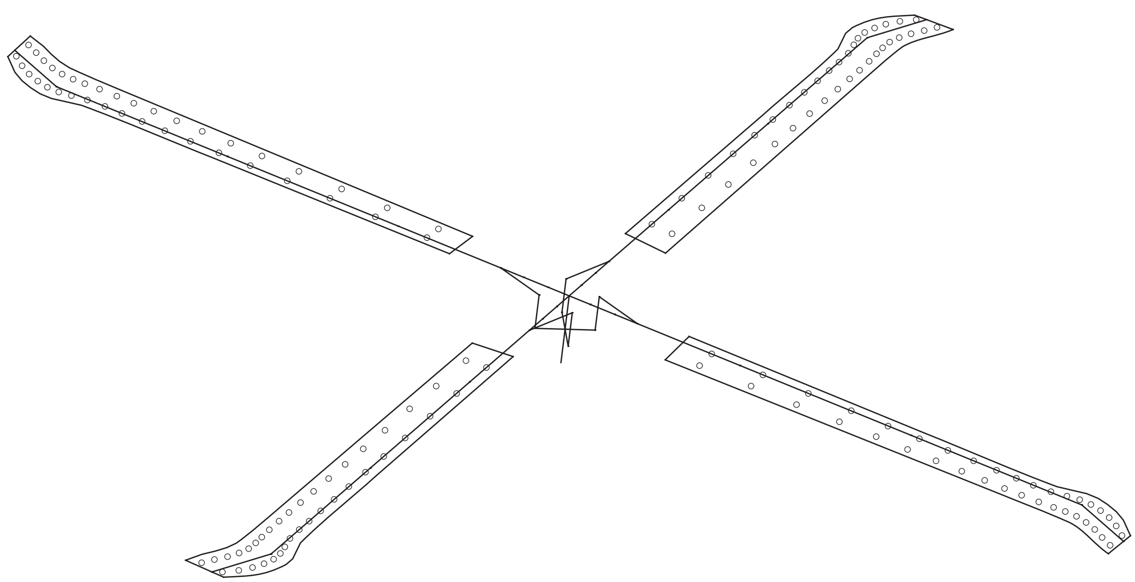
but all blades are constructed, and assumption that motion is same need not be made rotorcraft shell can also construct a rotor with unequal blade spacing

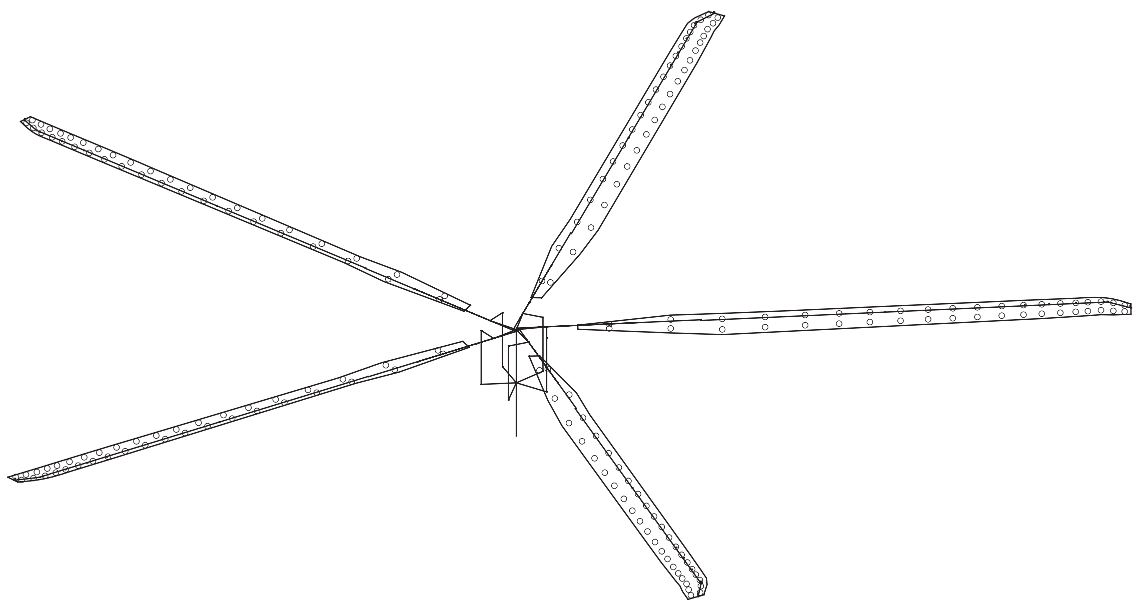
more general problems, including dissimilar blades, can be implemented using core input

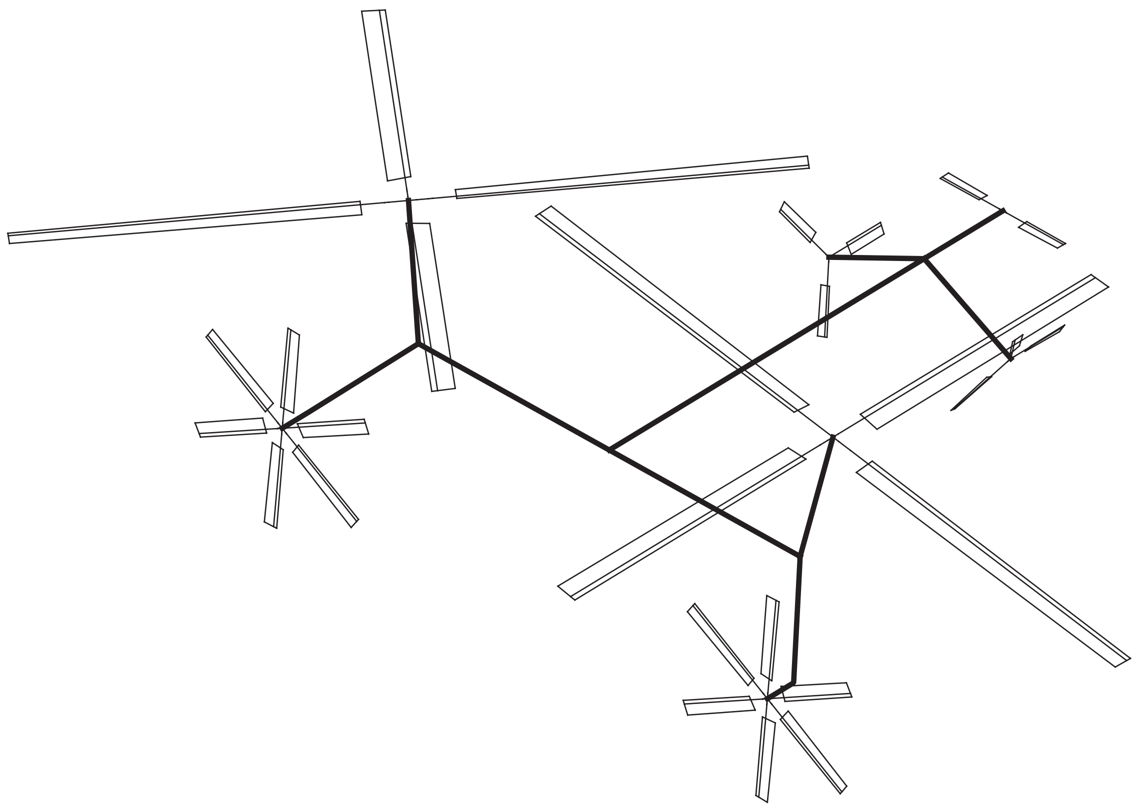
USUALLY ASSUME SYSTEM MOTION IS PERIODIC FOR TRIM ANALYSIS

so necessary to exclude vibratory dynamic and aerodynamic interaction between rotors operating at different rotational speeds (as for main rotor and tail rotor configuration)

can also analyze rotors and airframe using common period, with full interaction







6–2 Rotorcraft System

FIGURES 2 to 28 SHOW SYSTEM CONSTRUCTED BY ROTORCRAFT SHELL

conventions in figures:

components = rectangle; name in upper case, type in lower case

input/output interfaces = triangle

structural dynamic interfaces = diamond

input and output = semicircle

input/output interfaces and structural dynamic components (shaded) connect the figures

FOR NAMES OF SYSTEM PIECES AND VARIABLES APPEARING IN FIGURES 2 to 28:

CAMRAD II Documentation, Volume III, Rotorcraft Theory; Chapter “Rotorcraft System”

PART AND LOOP NAMES ARE LISTED IN OUTLINES OF TRIM, TRANSIENT, AND FLUTTER SOLUTION PROCEDURES

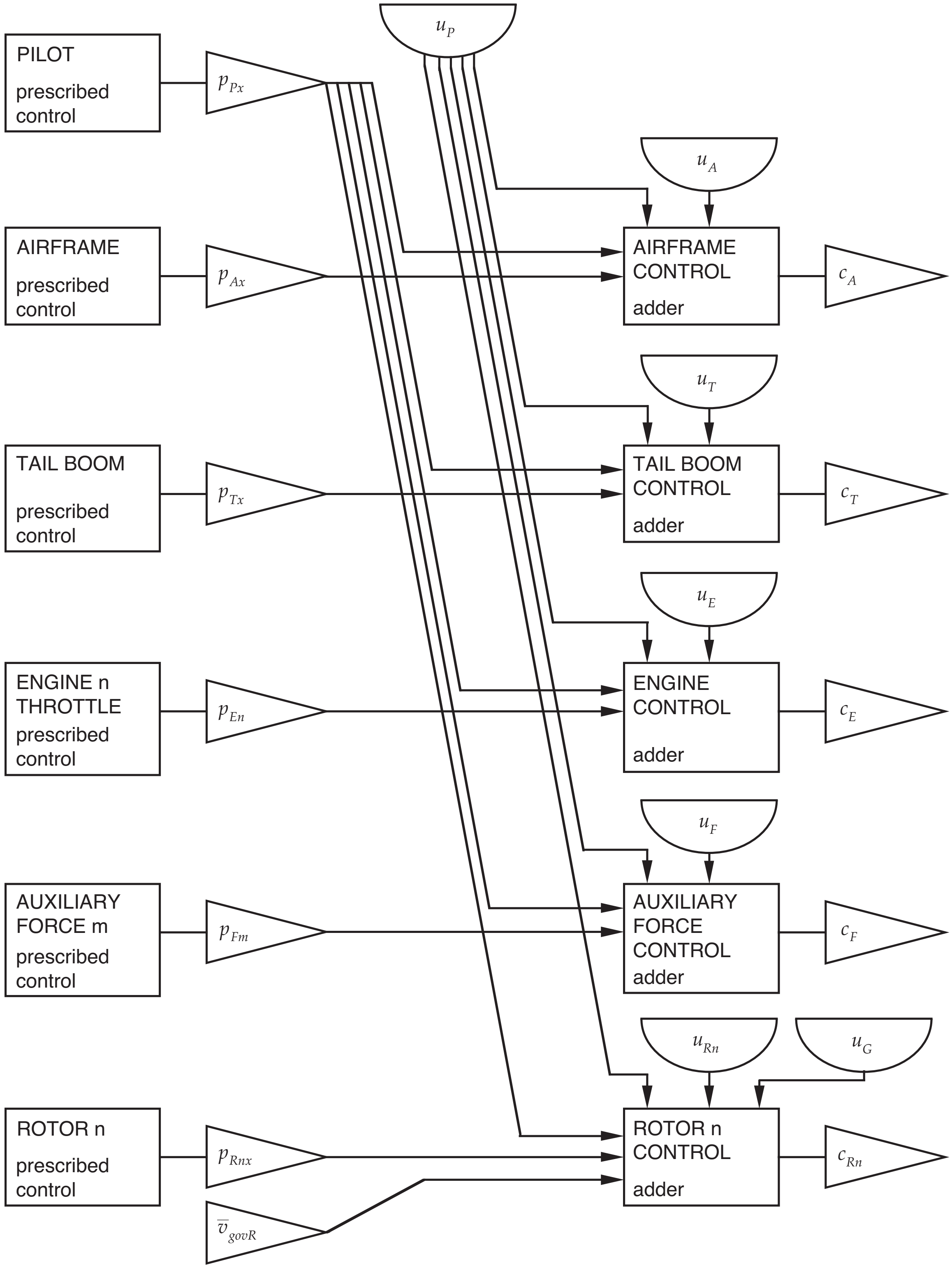


Figure 6-2 Rotorcraft control.

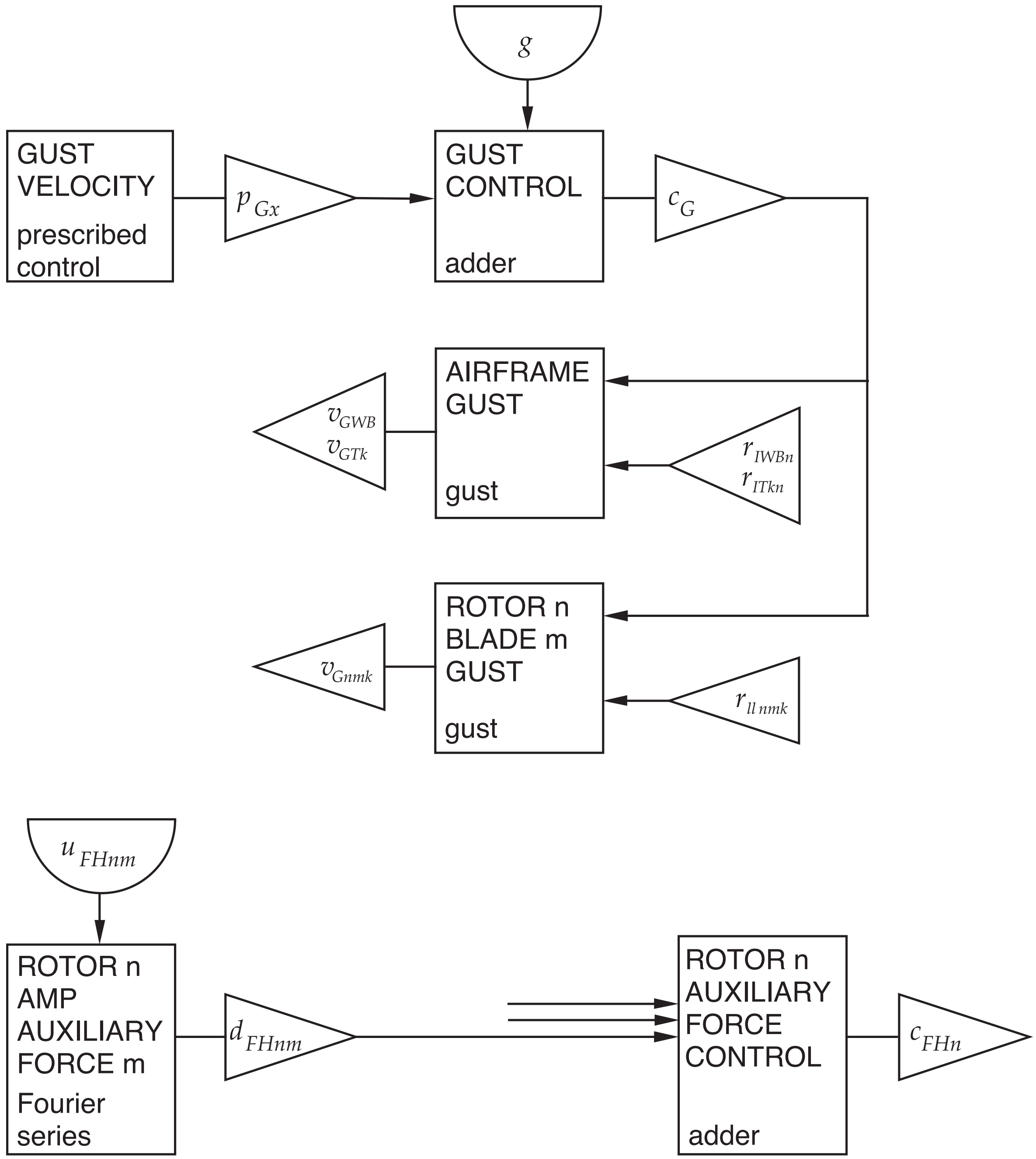


Figure 6-3 Gust and auxiliary force control.

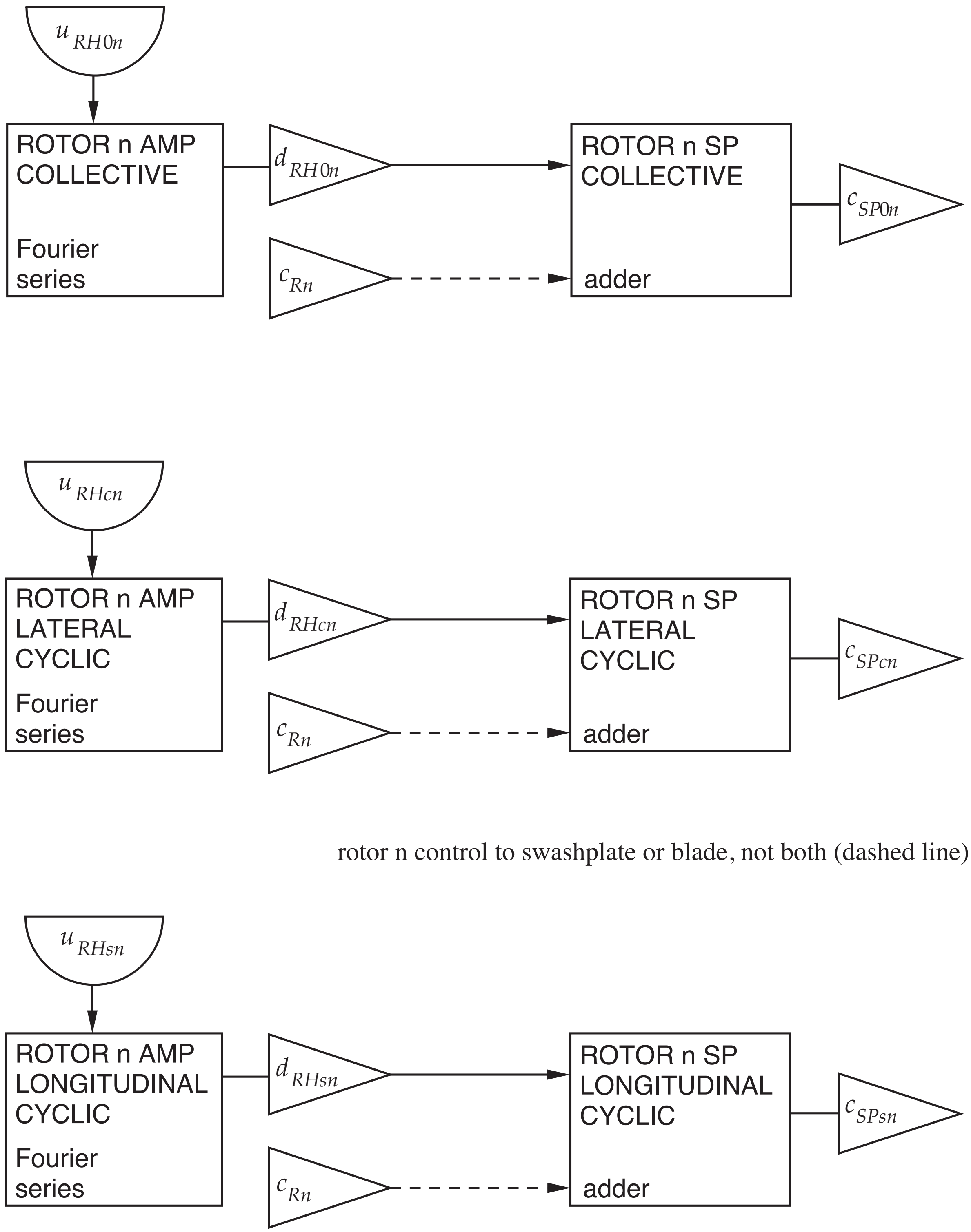


Figure 6-4 Rotor swashplate control.

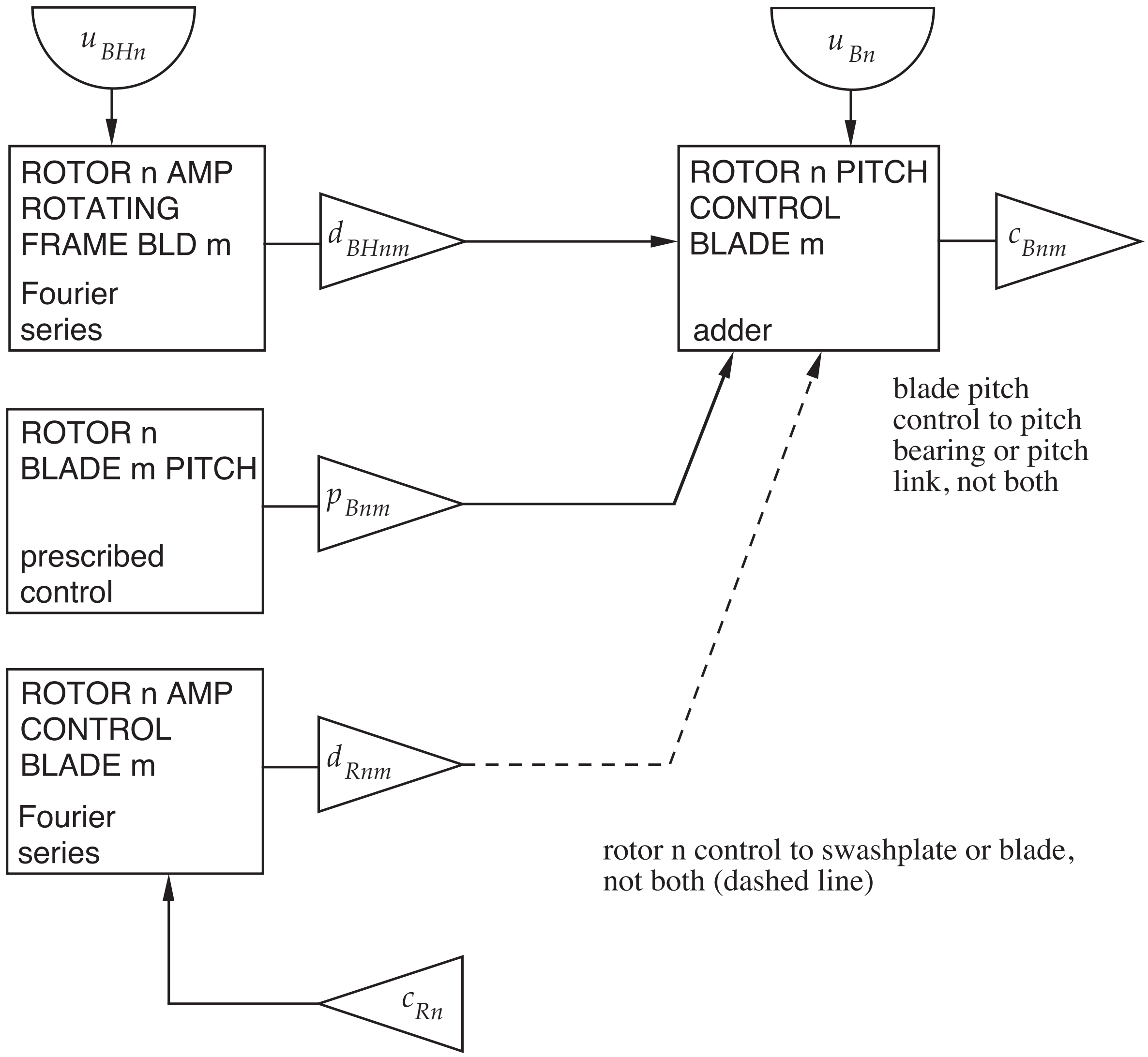


Figure 6-5 Rotor blade control.

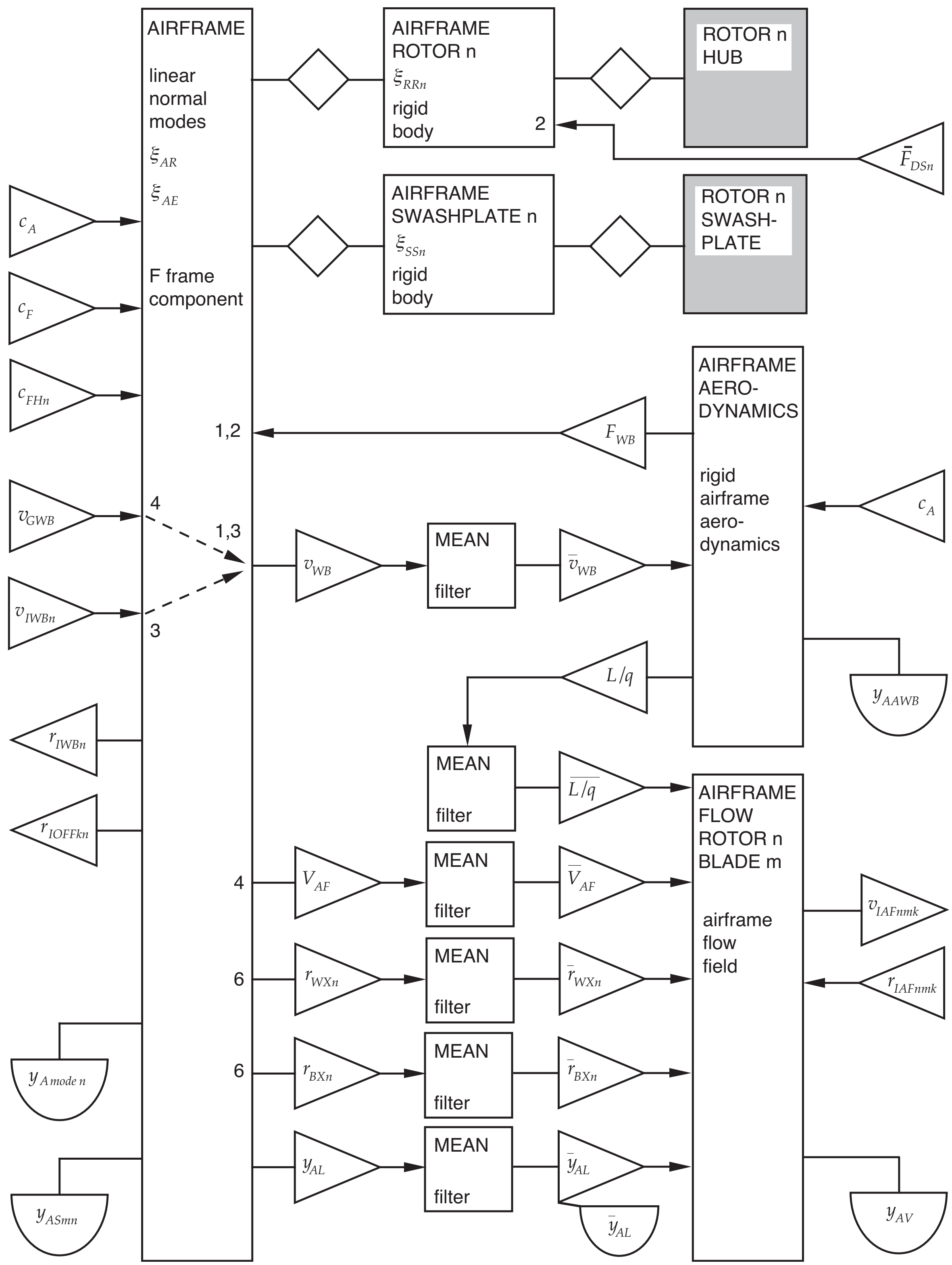


Figure 6-6 Airframe.

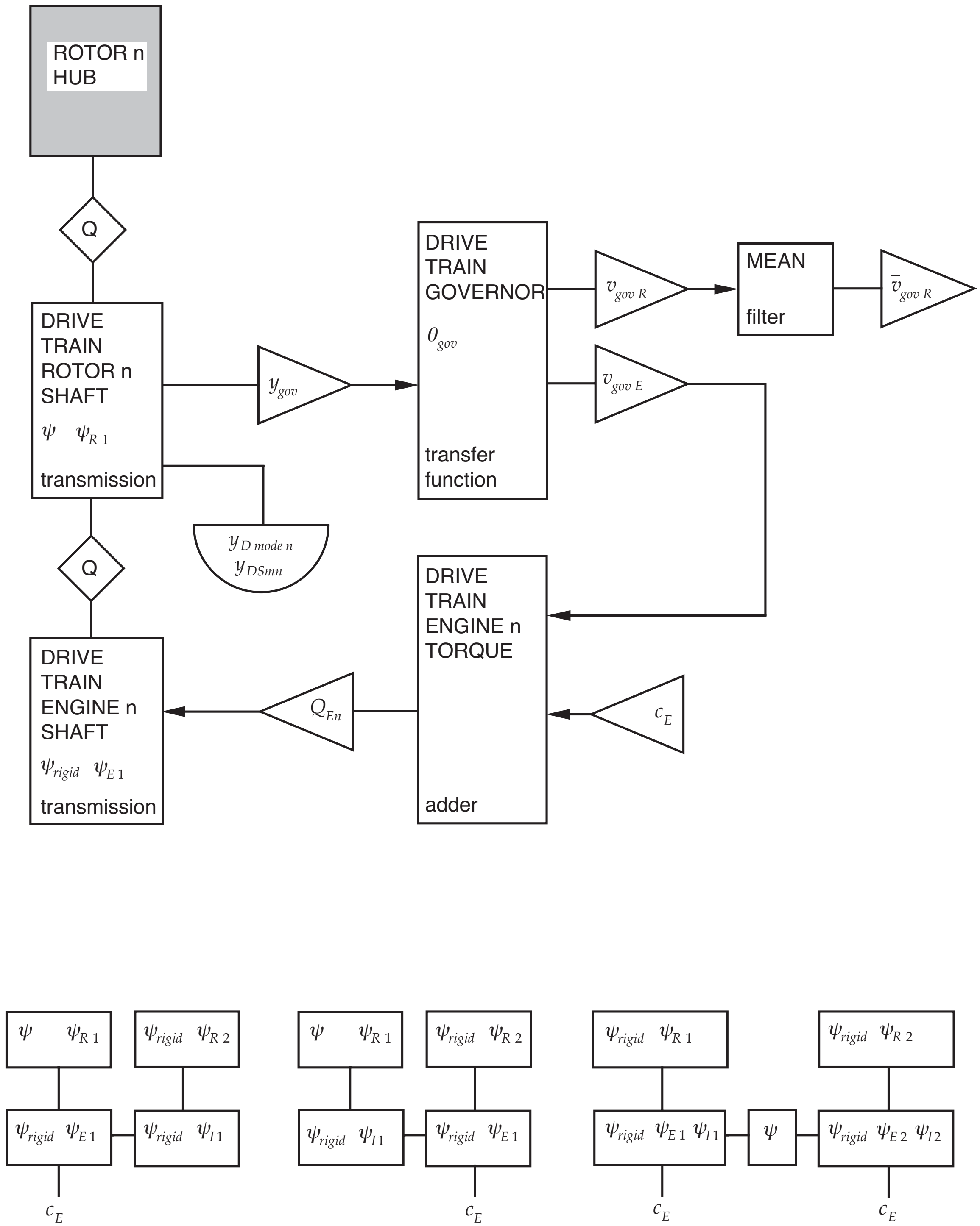


Figure 6-7 Drive train.

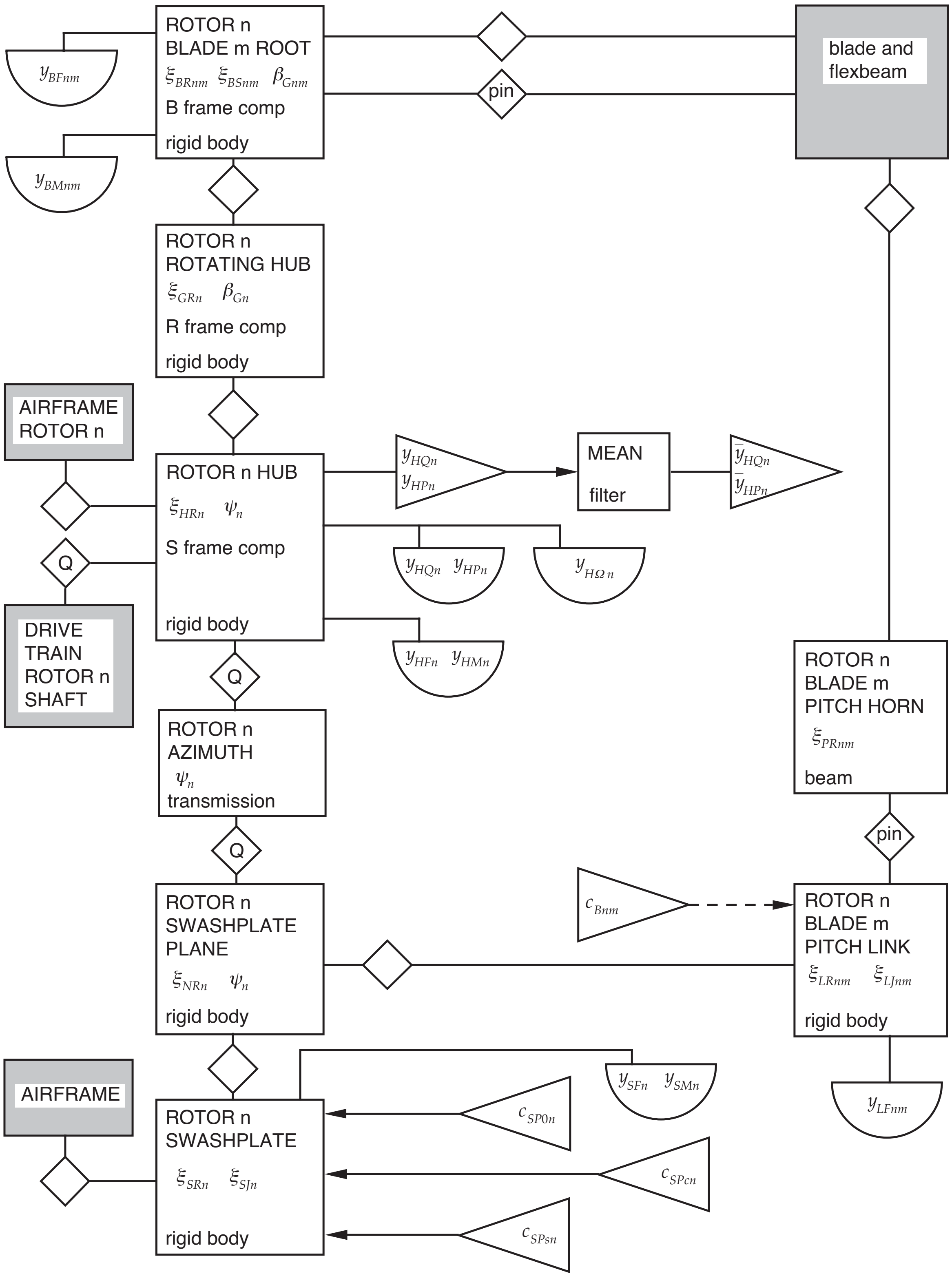


Figure 6-8 Rotor hub and control system.

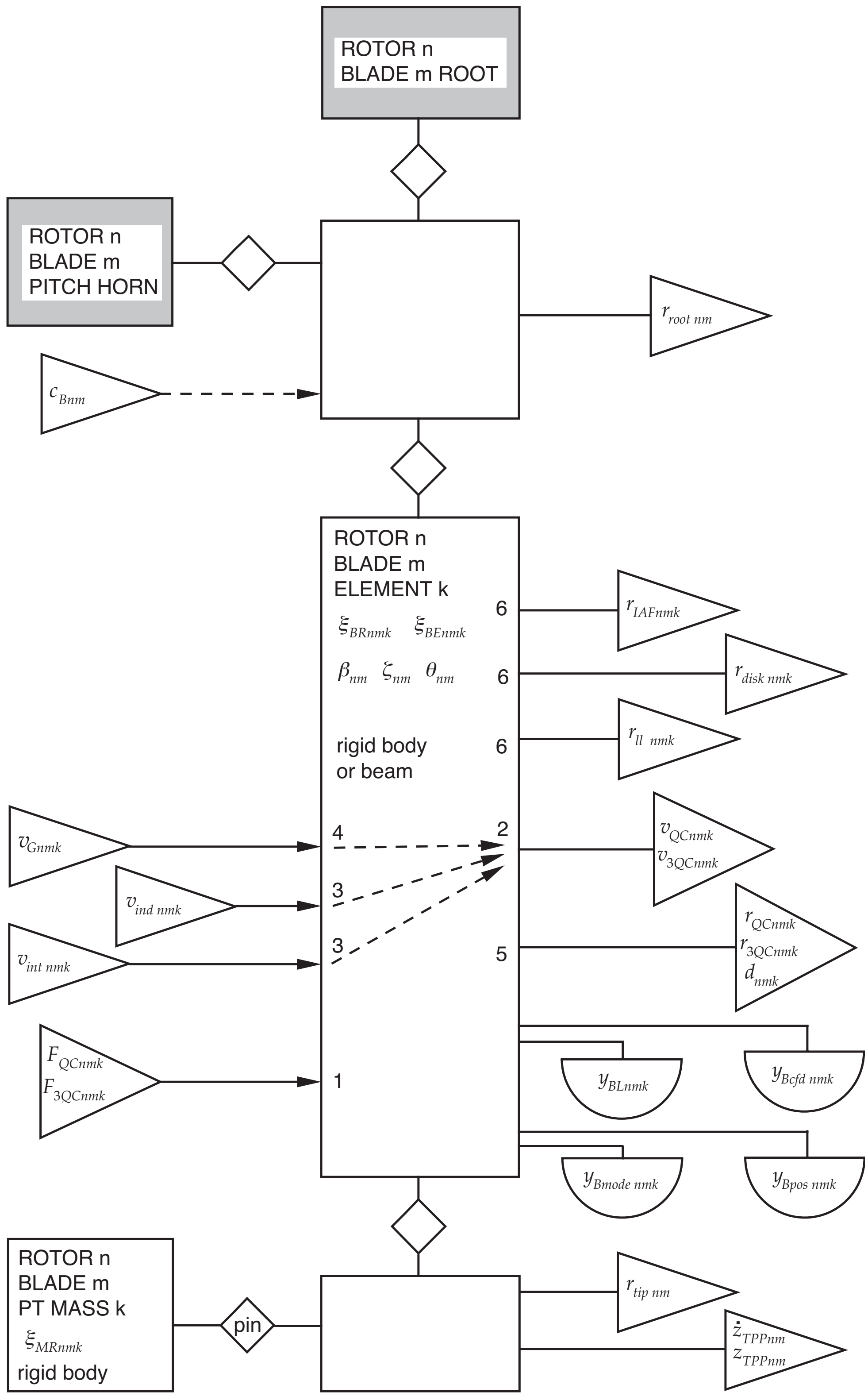


Figure 6-9 Rotor blade structure (single load path).

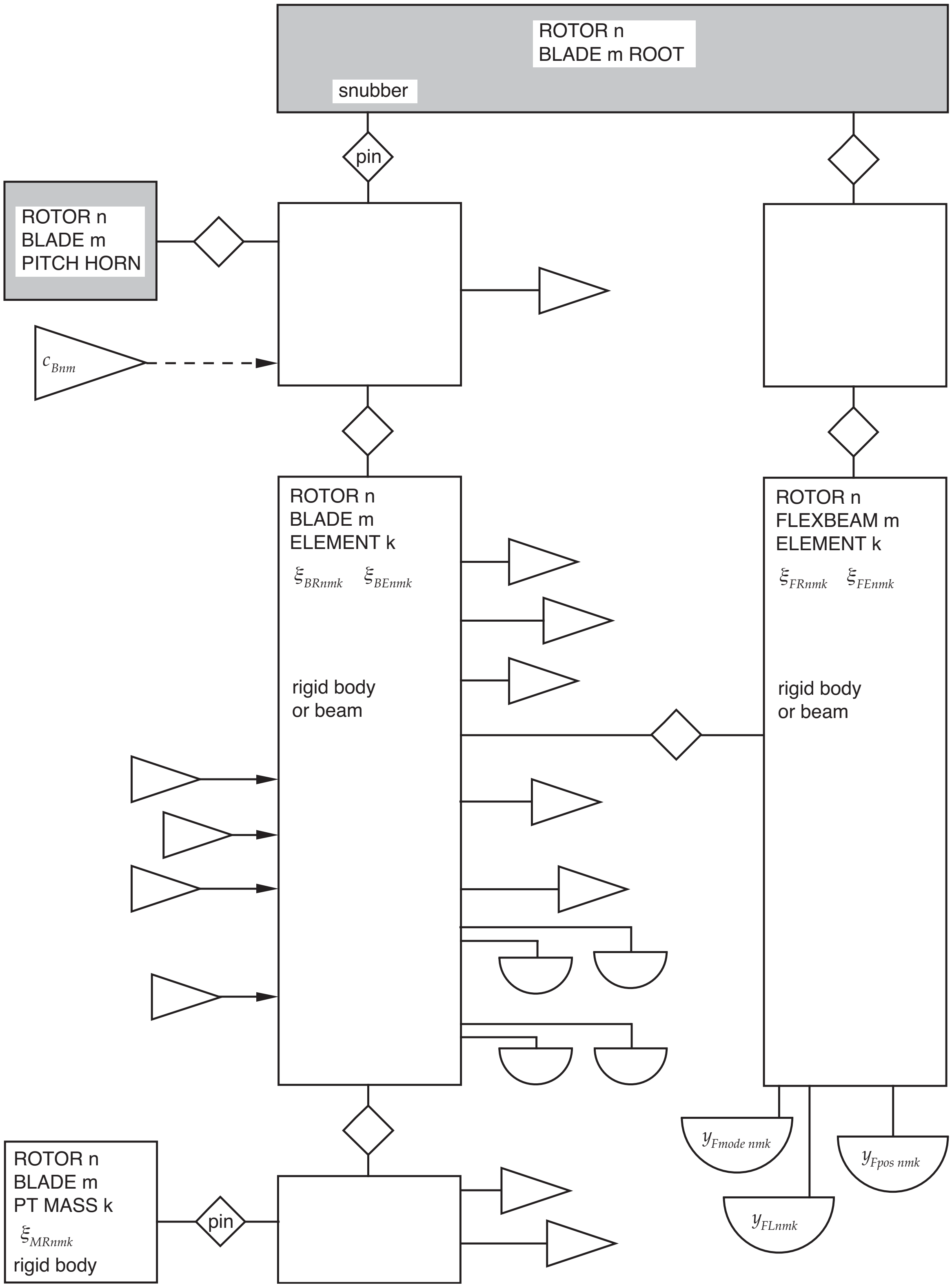


Figure 6-10 Rotor blade structure (bearingless).

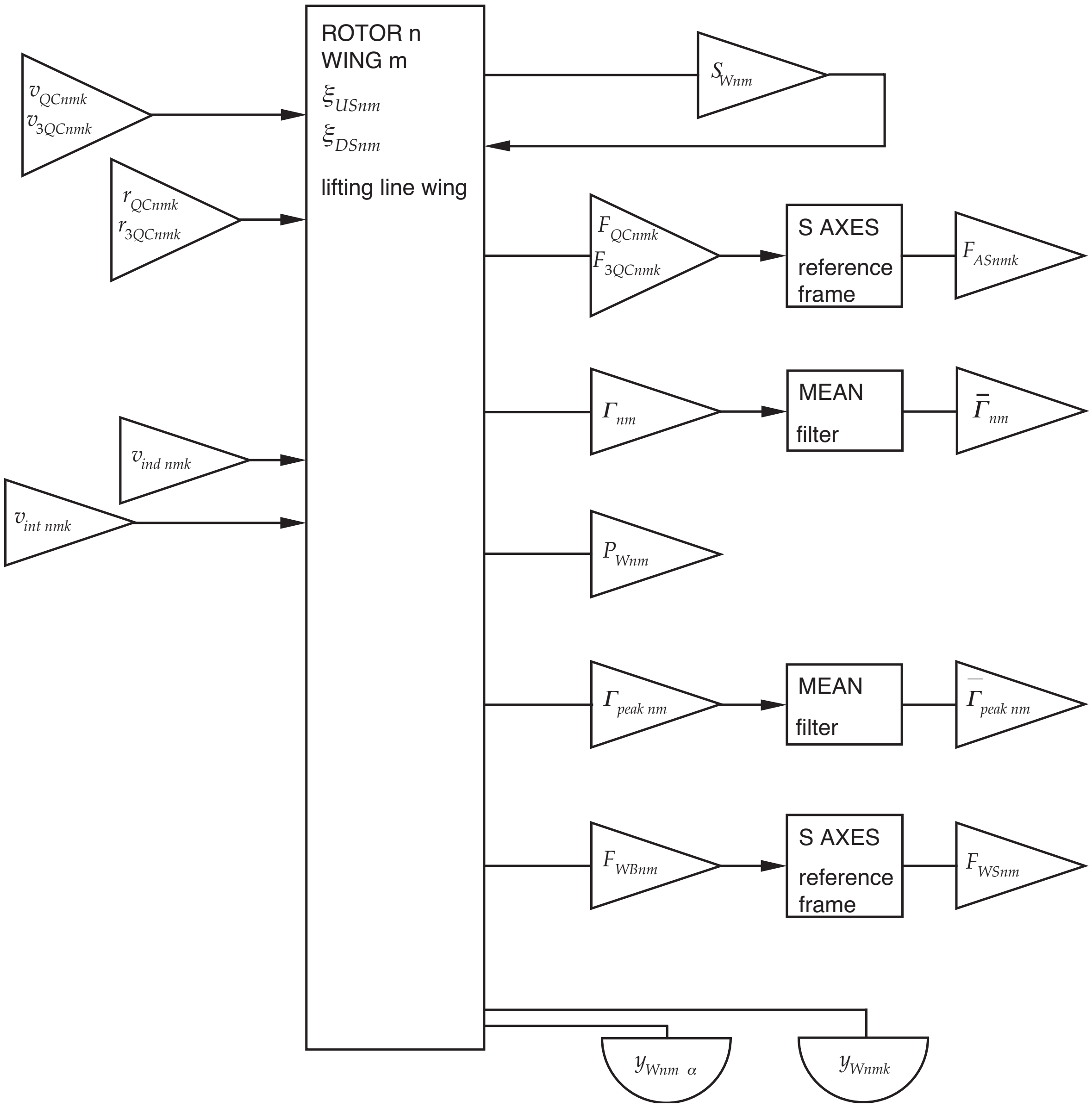


Figure 6-11 Rotor blade aerodynamics.

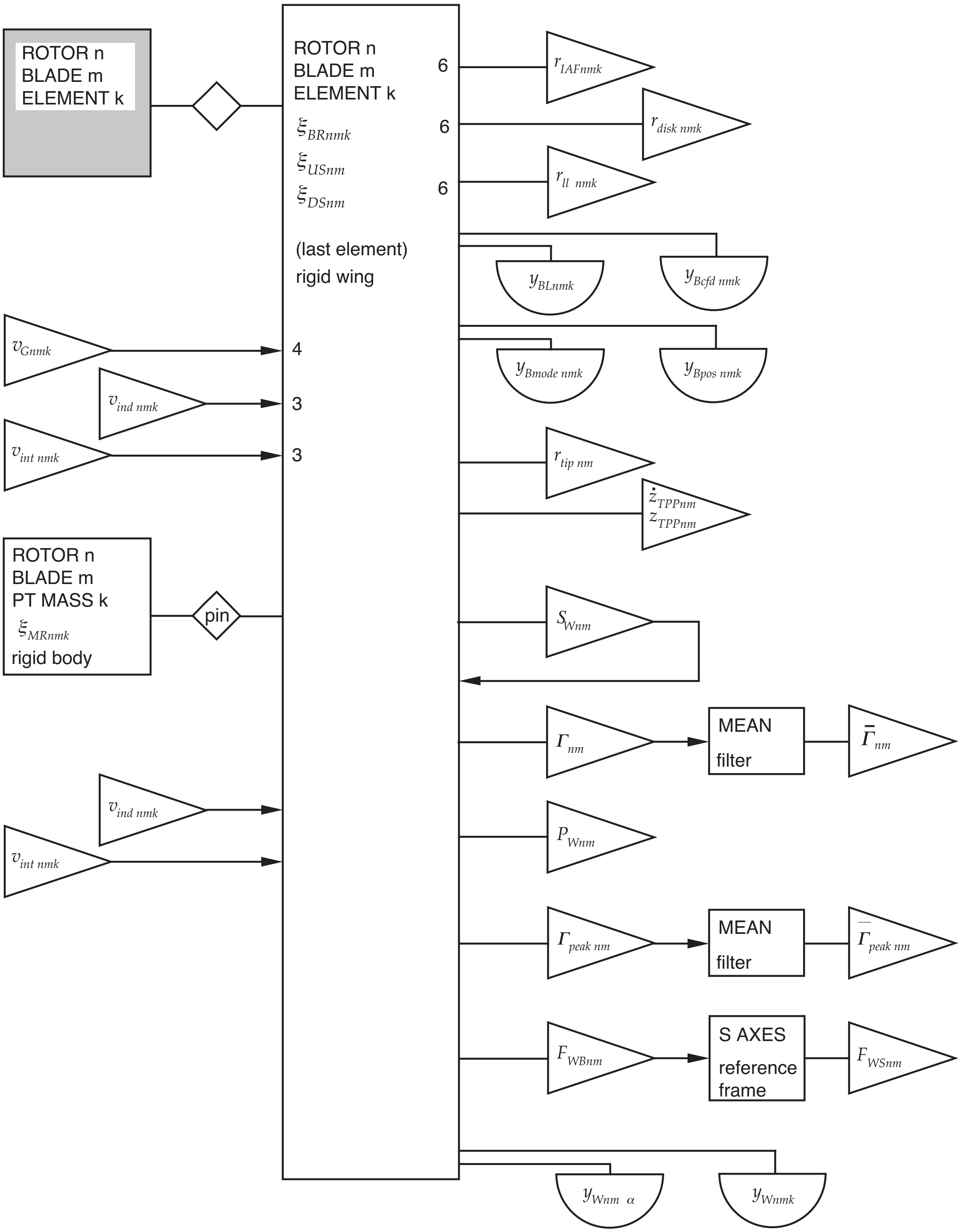


Figure 6-12 Rotor blade structure and aerodynamics (rigid wing).

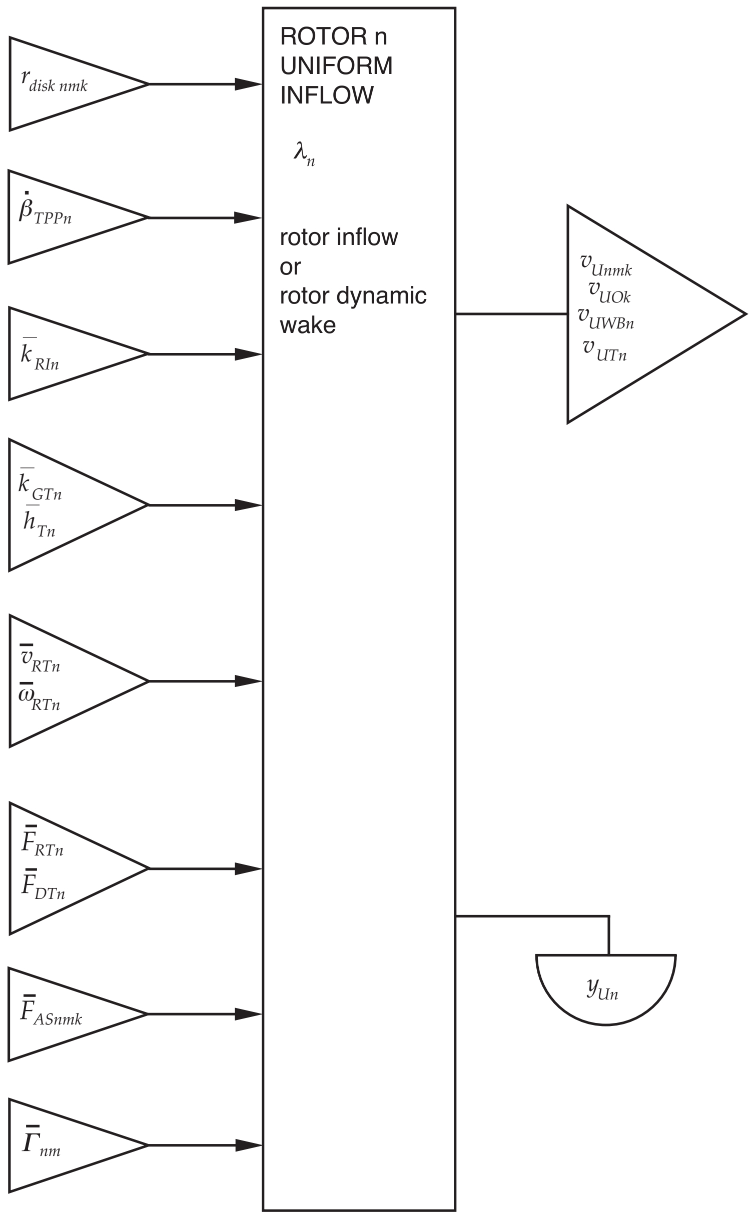


Figure 6-13 Rotor uniform inflow.

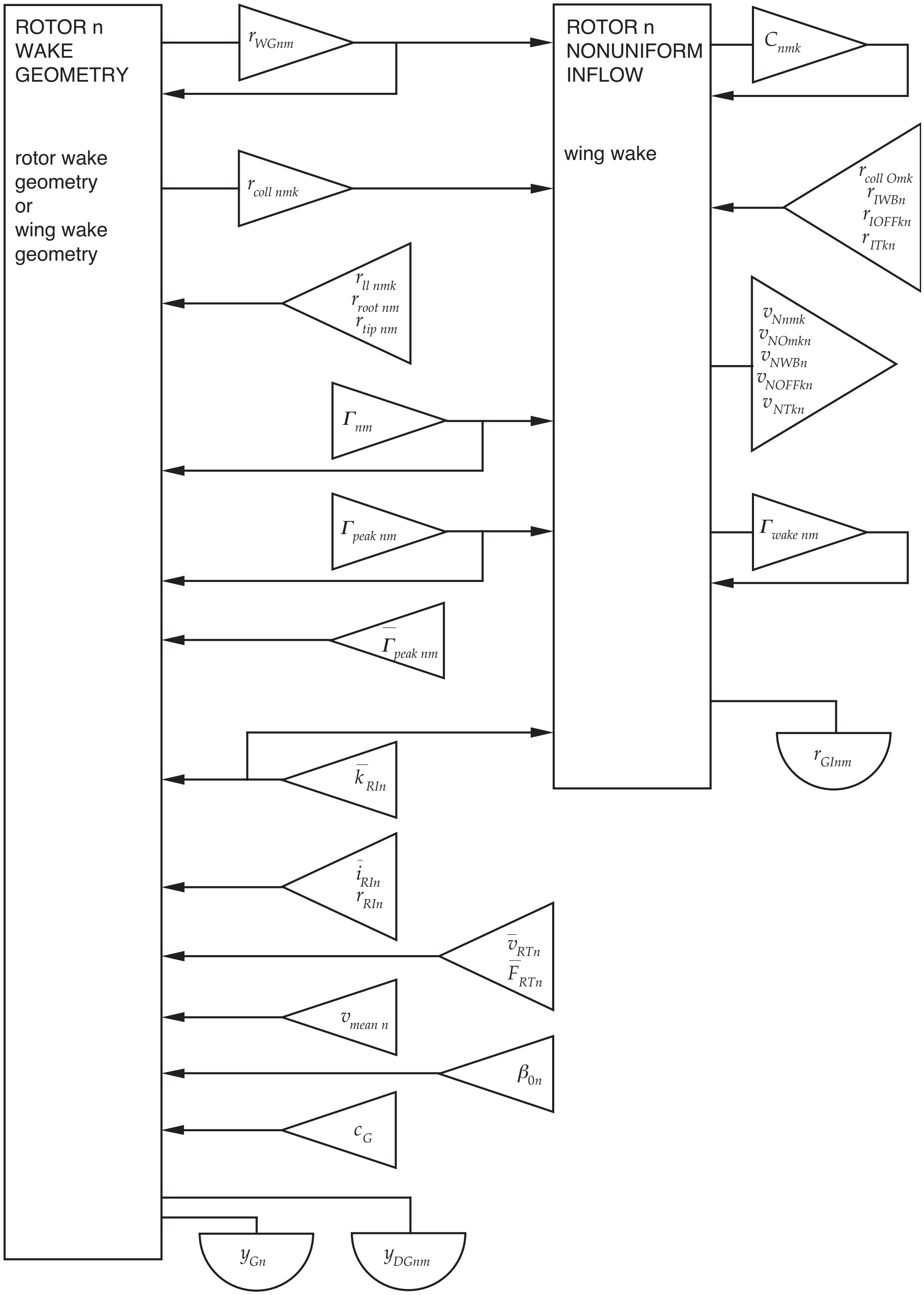


Figure 6-14 Rotor nonuniform inflow.

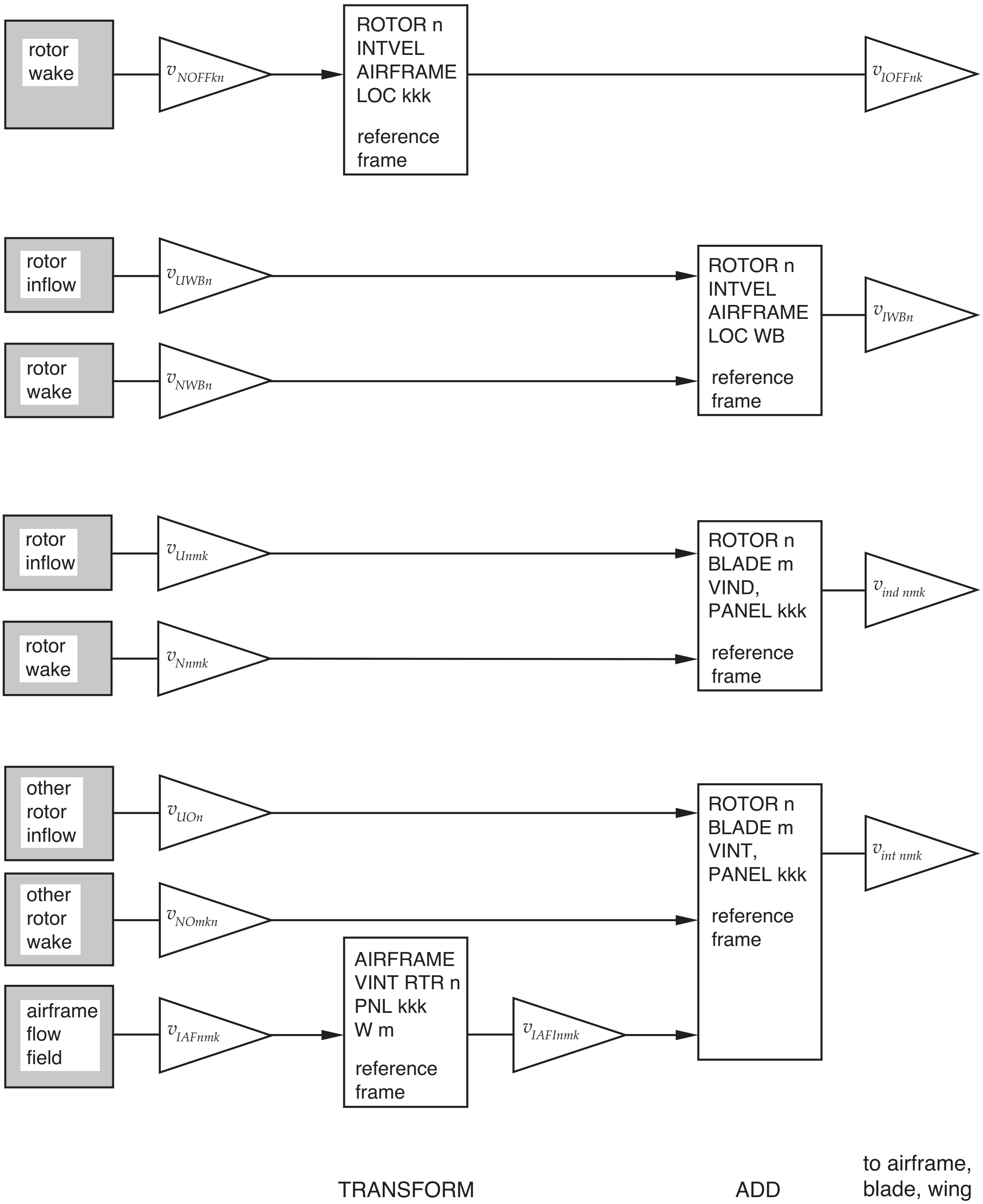


Figure 6-15a Induced and interference velocities.

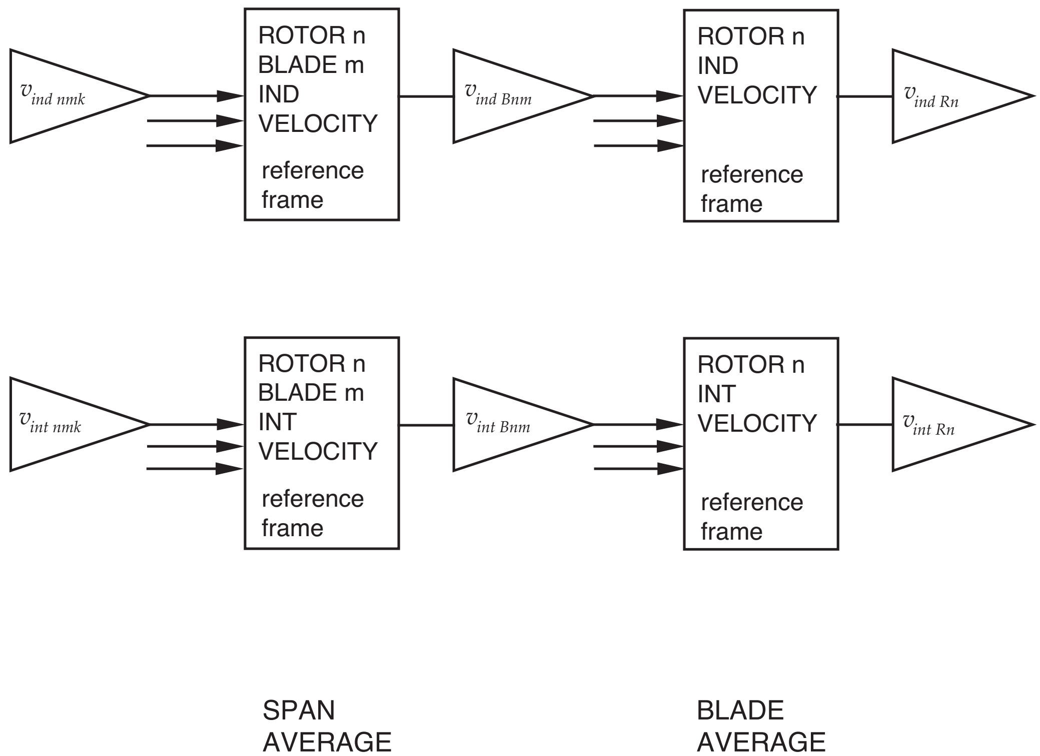


Figure 6-15b Induced and interference velocities.

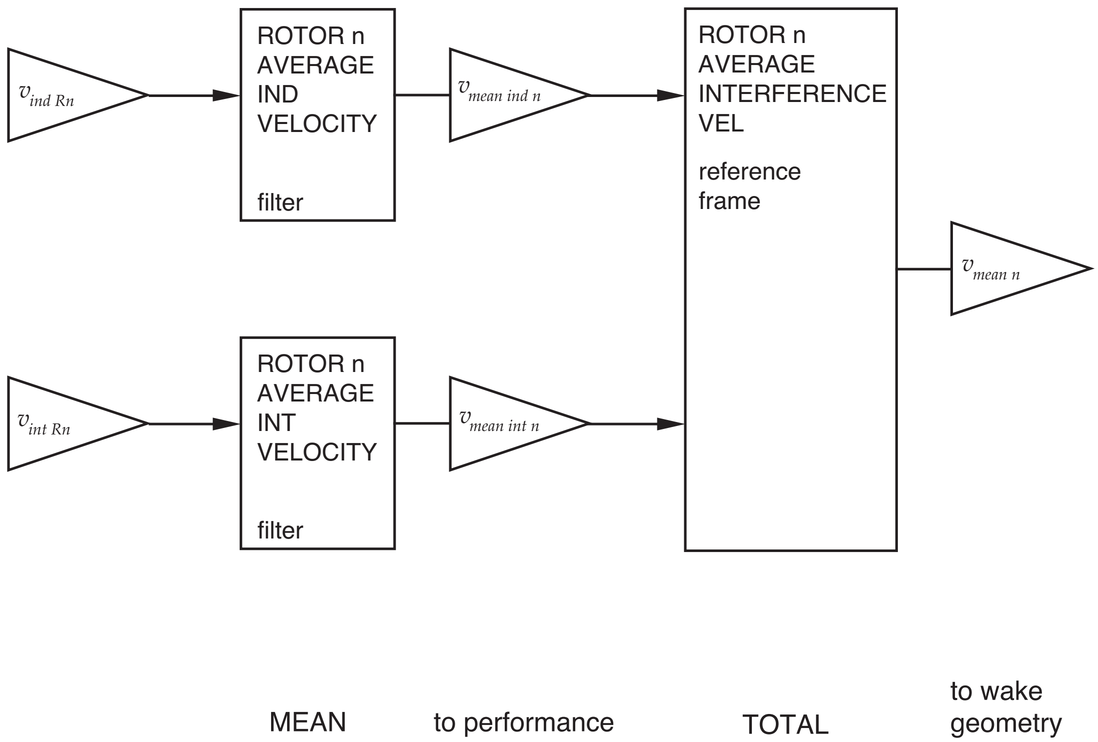


Figure 6-15c Induced and interference velocities.

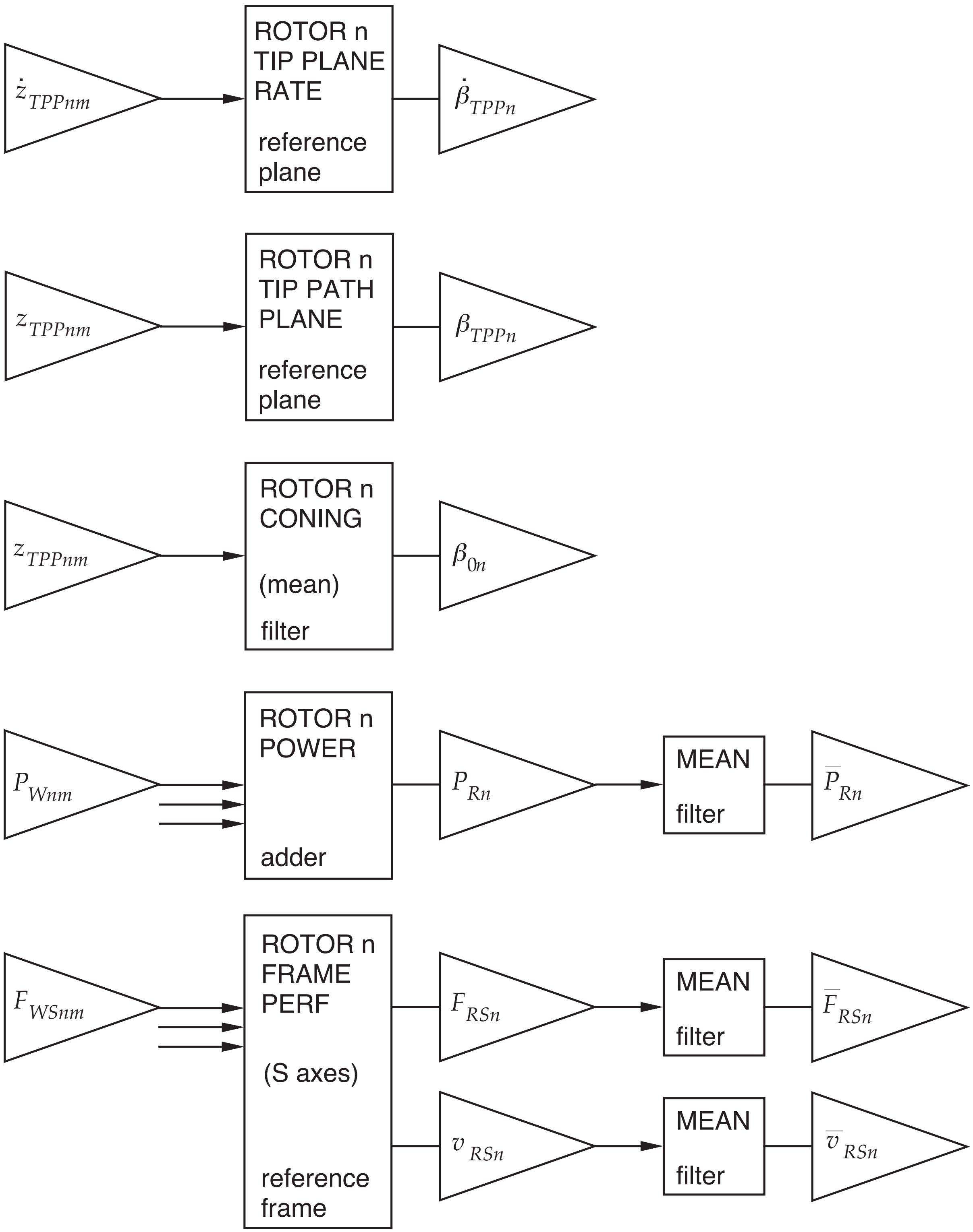


Figure 6-16 Rotor tip-path-plane tilt, coning, power, and frame (performance).

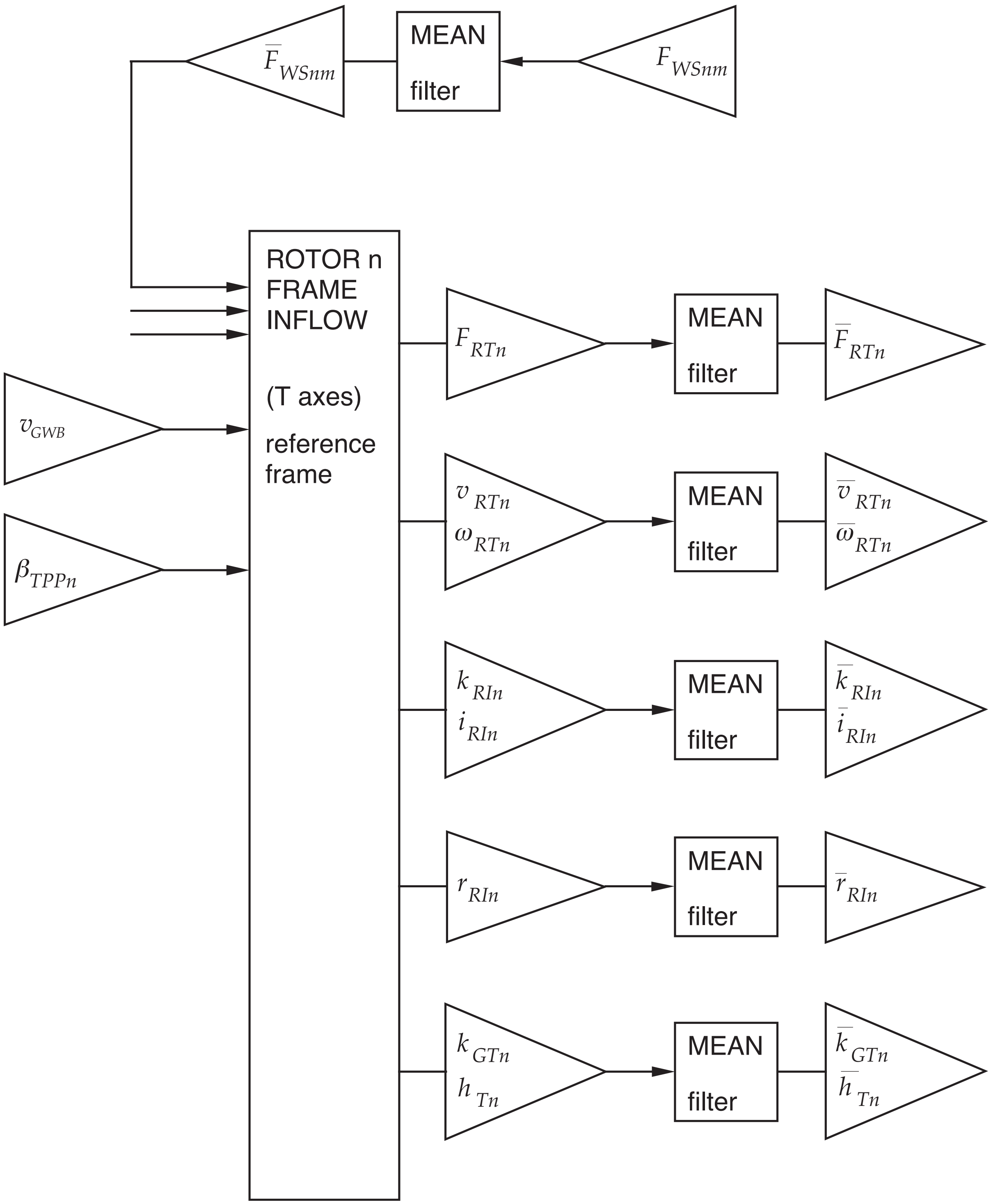


Figure 6-17 Rotor frame (inflow).

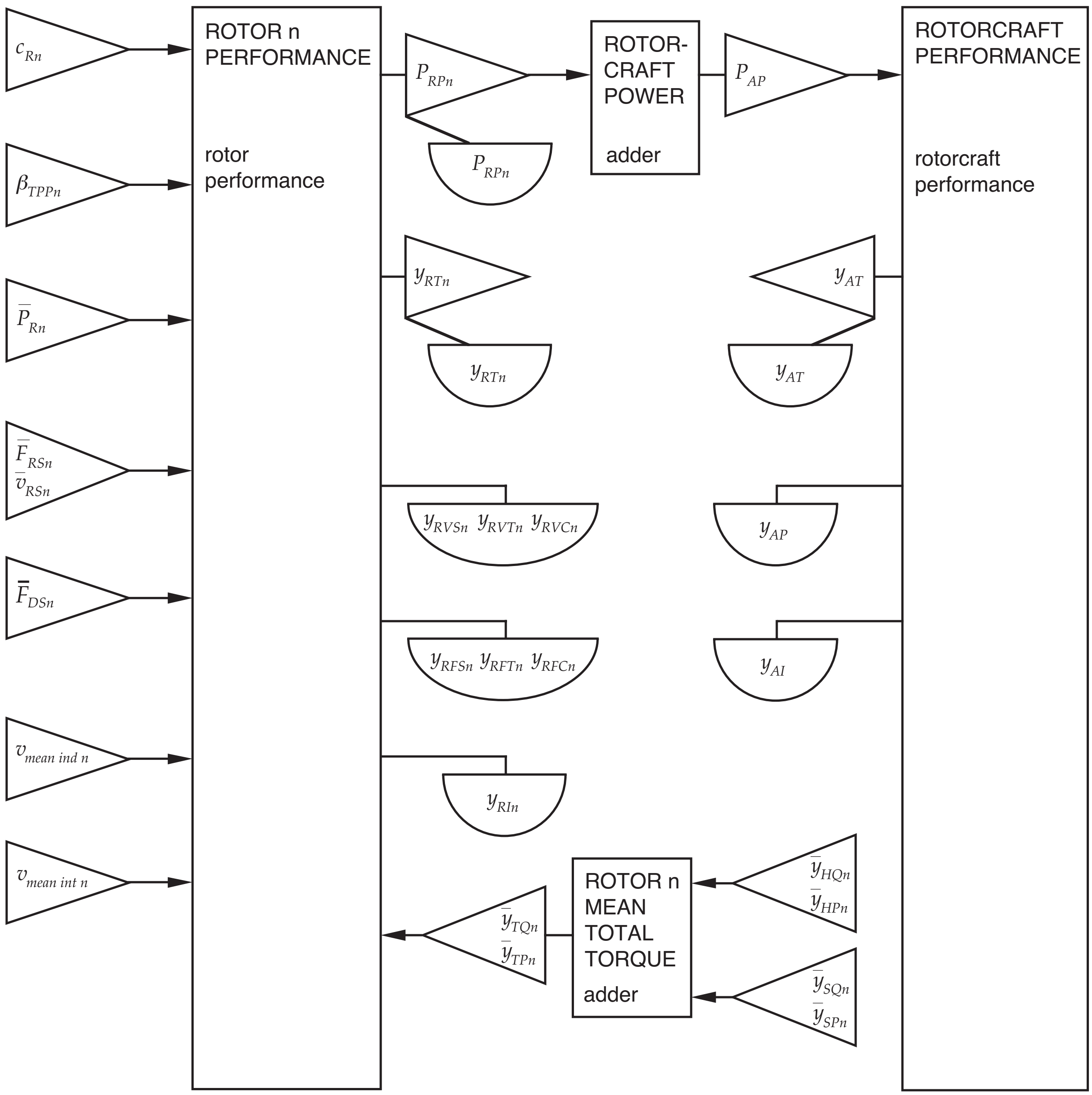


Figure 6-18 Rotor and rotorcraft performance.

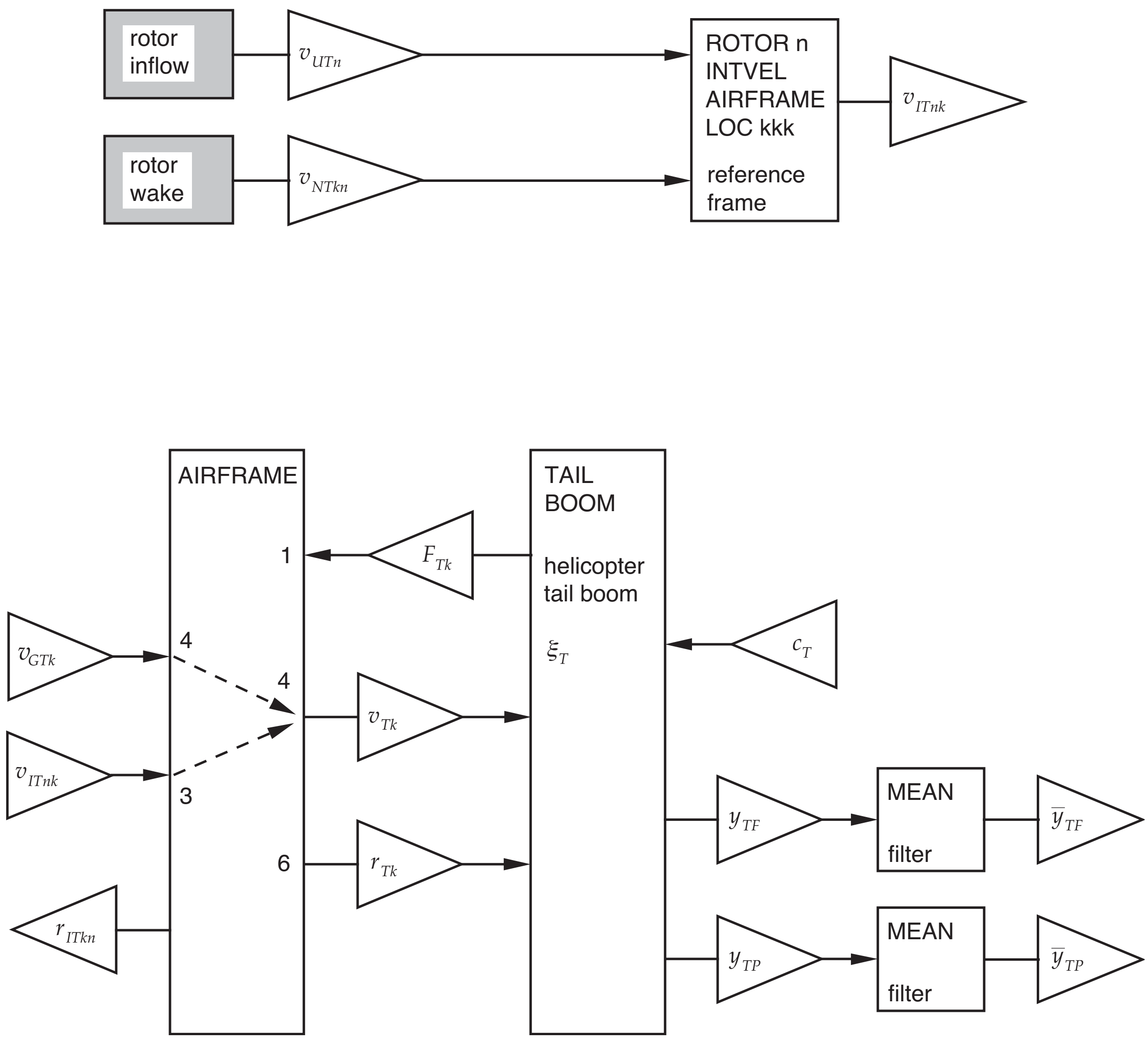


Figure 6-19 Tail boom.

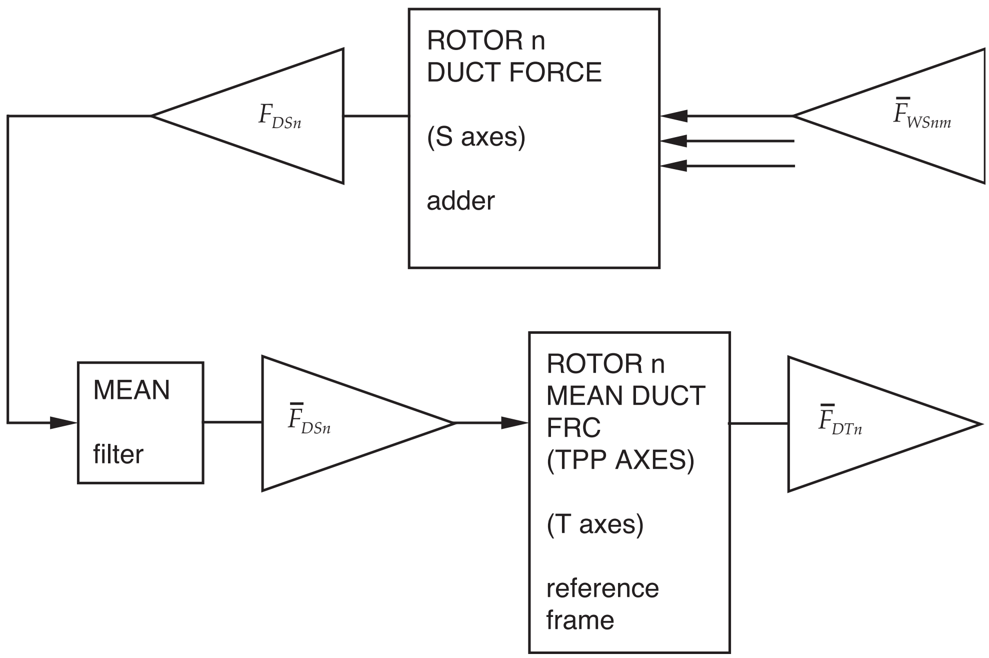


Figure 6-20 Rotor duct force.

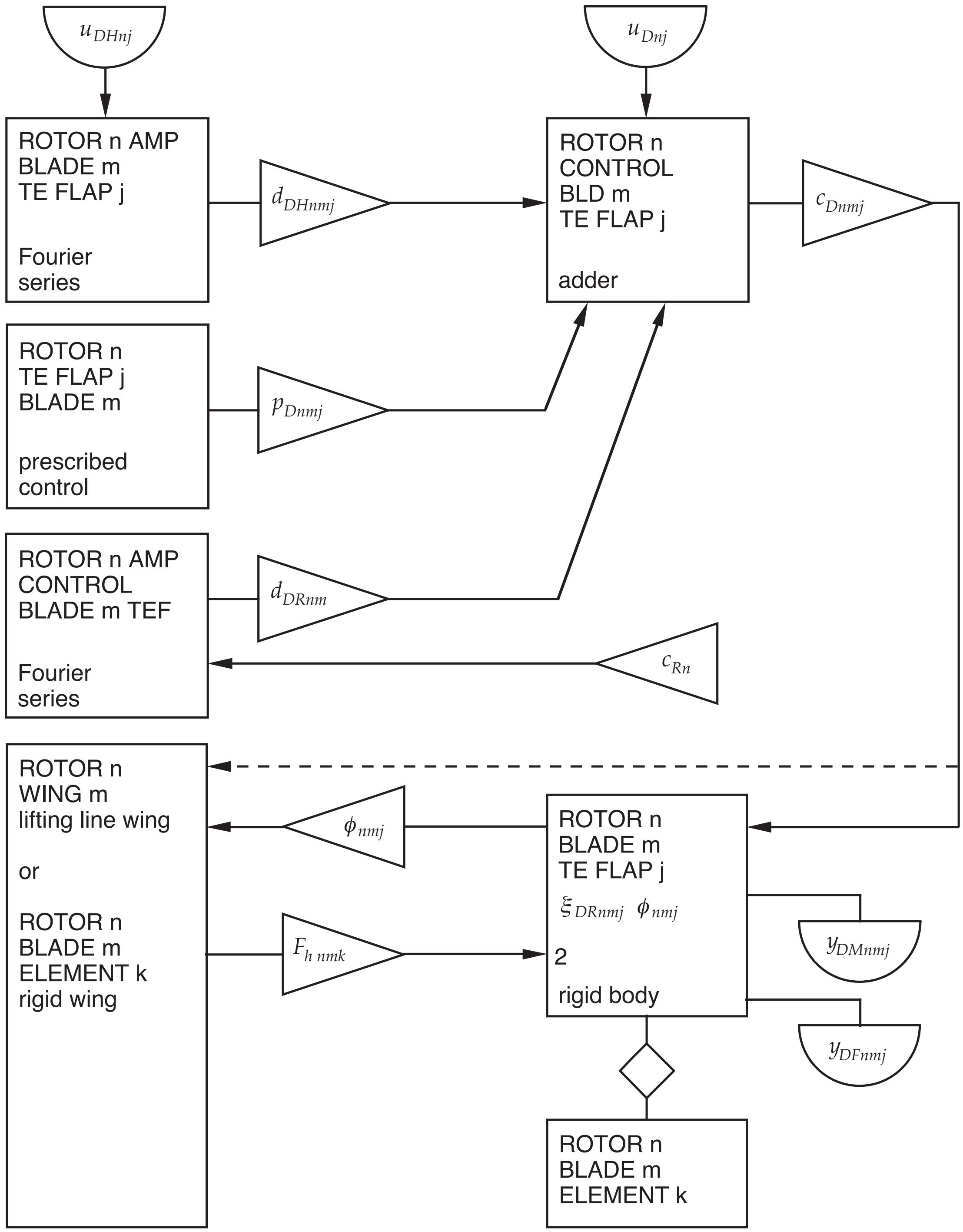


Figure 6-21 Rotor blade trailing edge flap control.

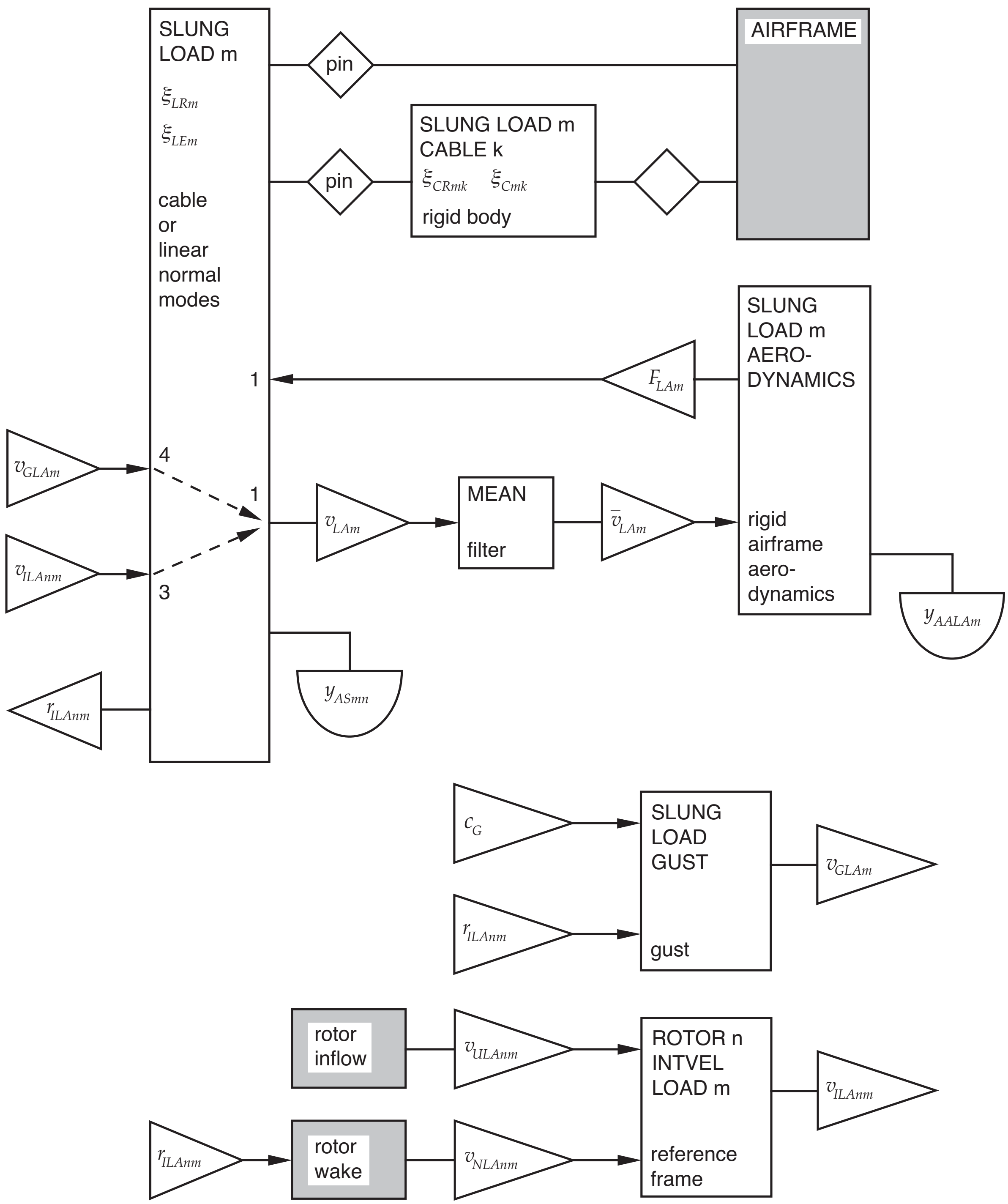


Figure 6-22 Slung load.

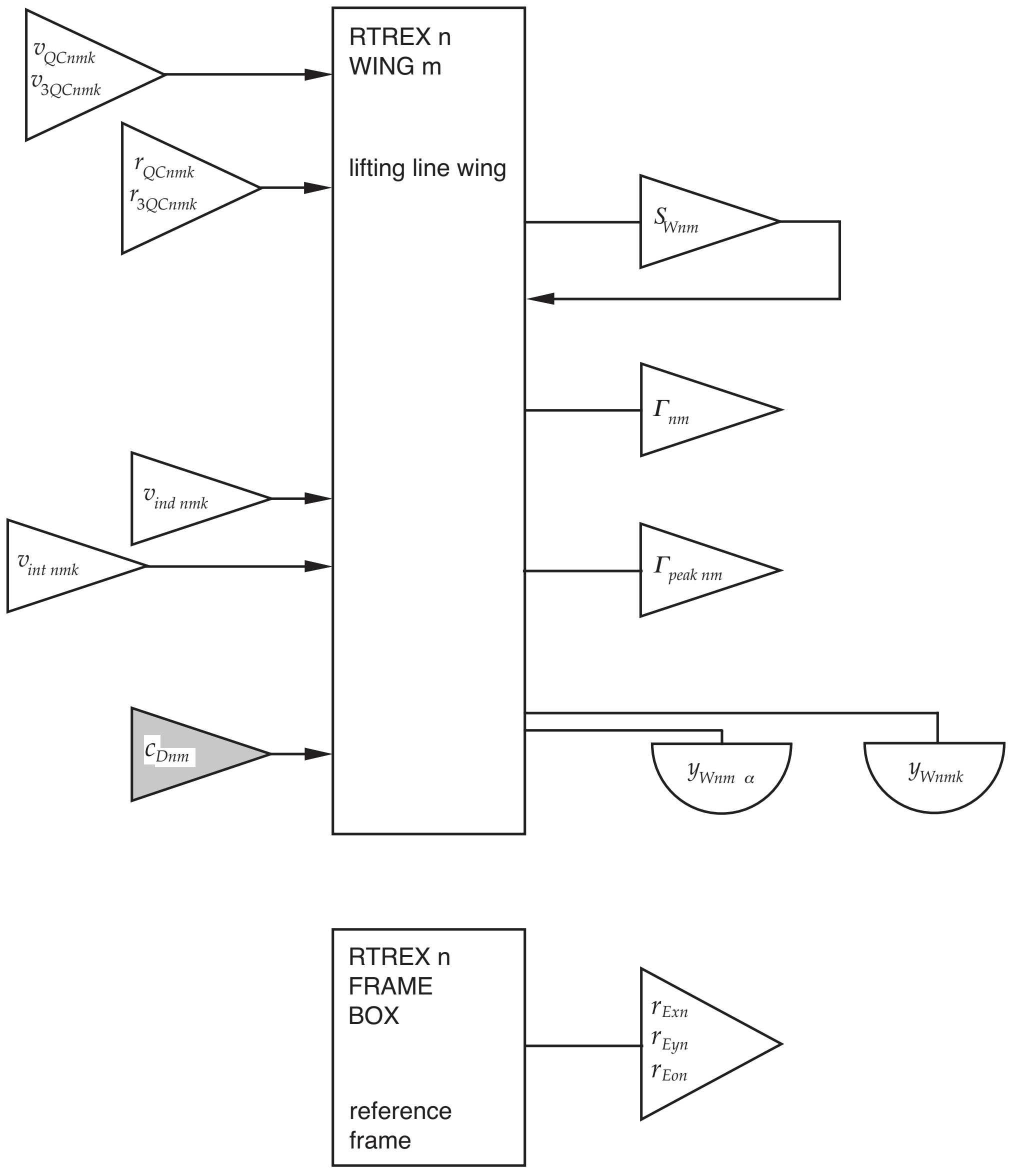


Figure 6-23a Post-trim rotor blade aerodynamics.

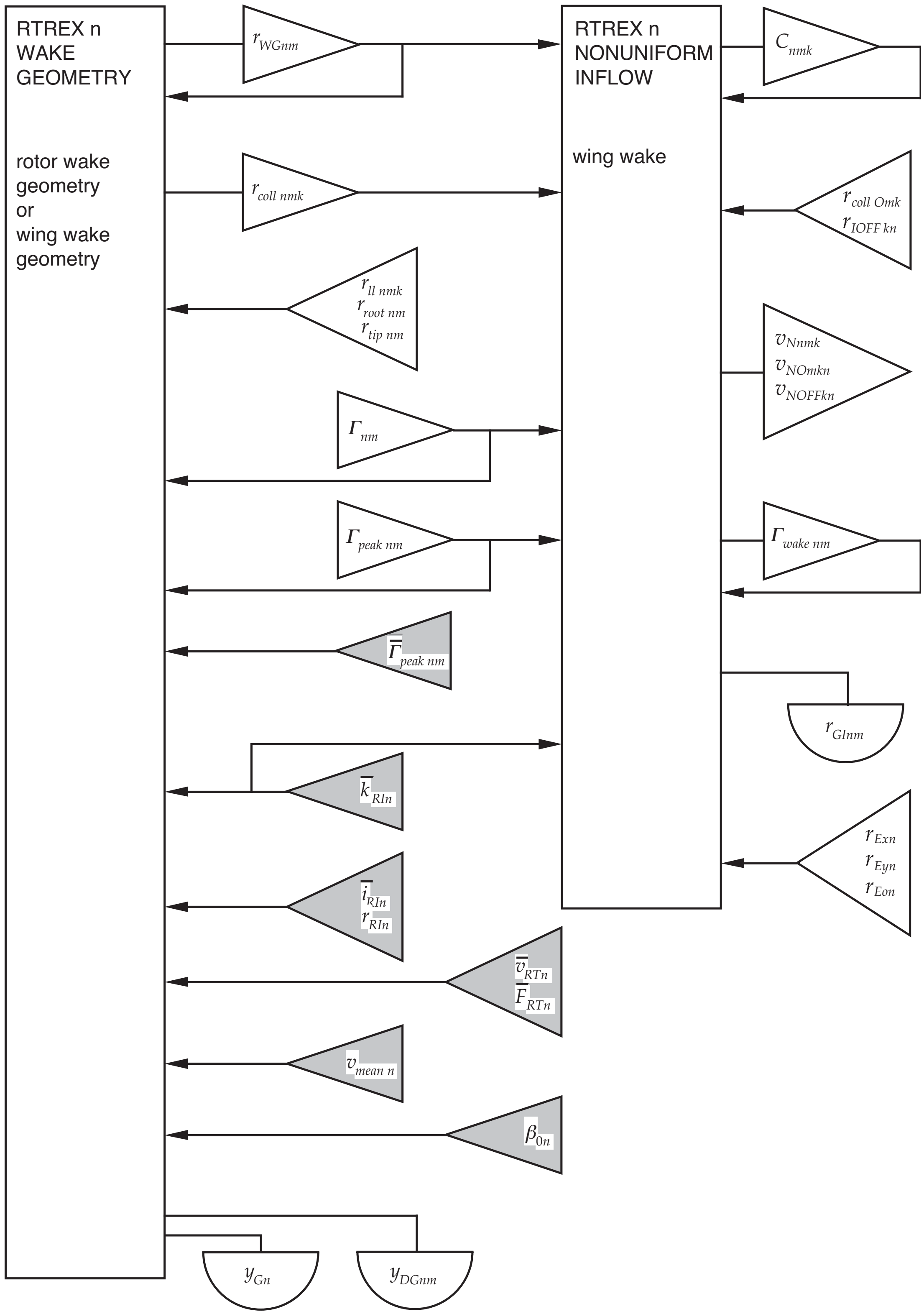


Figure 6-23b Post-trim rotor blade aerodynamics.

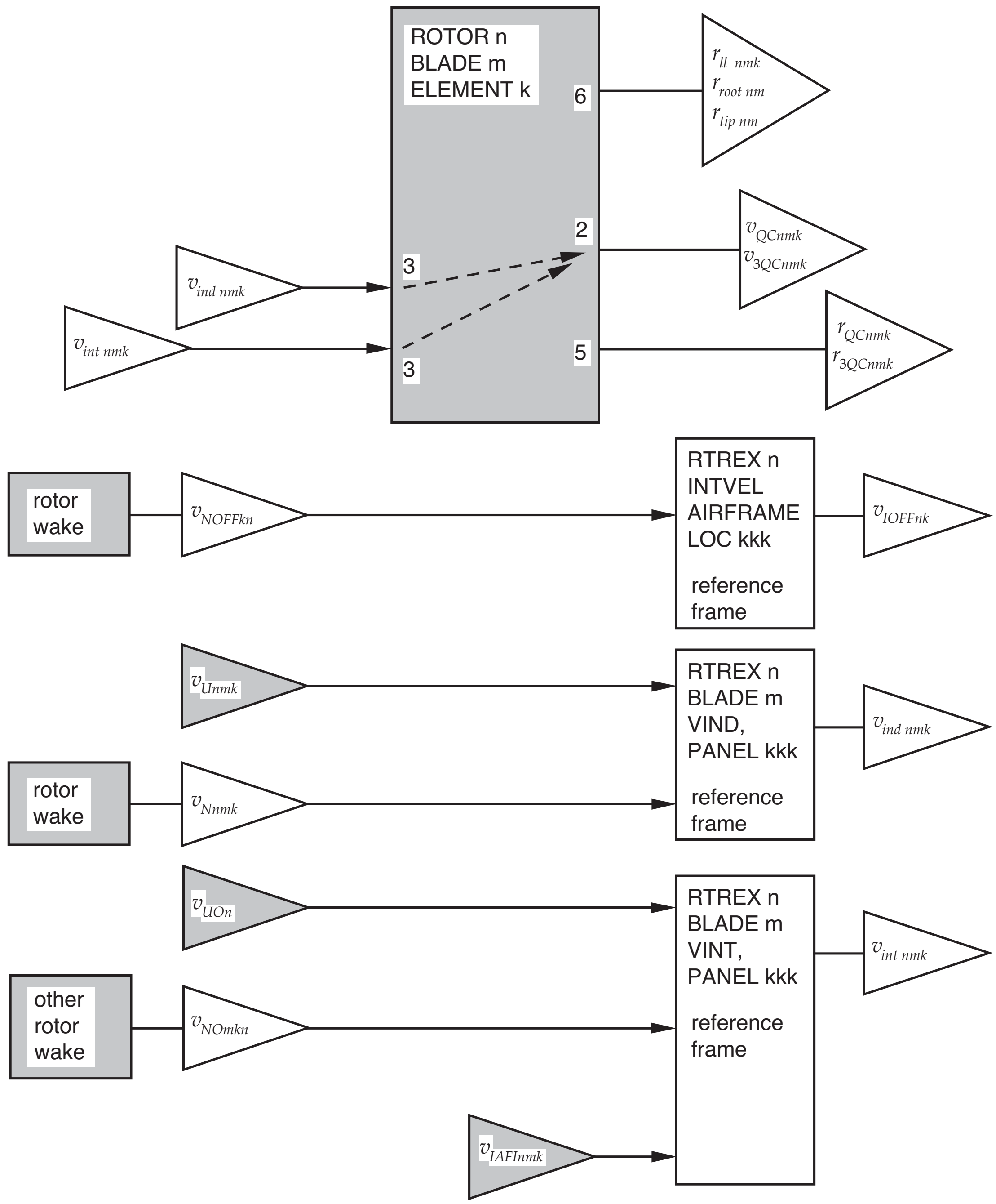


Figure 6-23c Post-trim rotor blade aerodynamics.

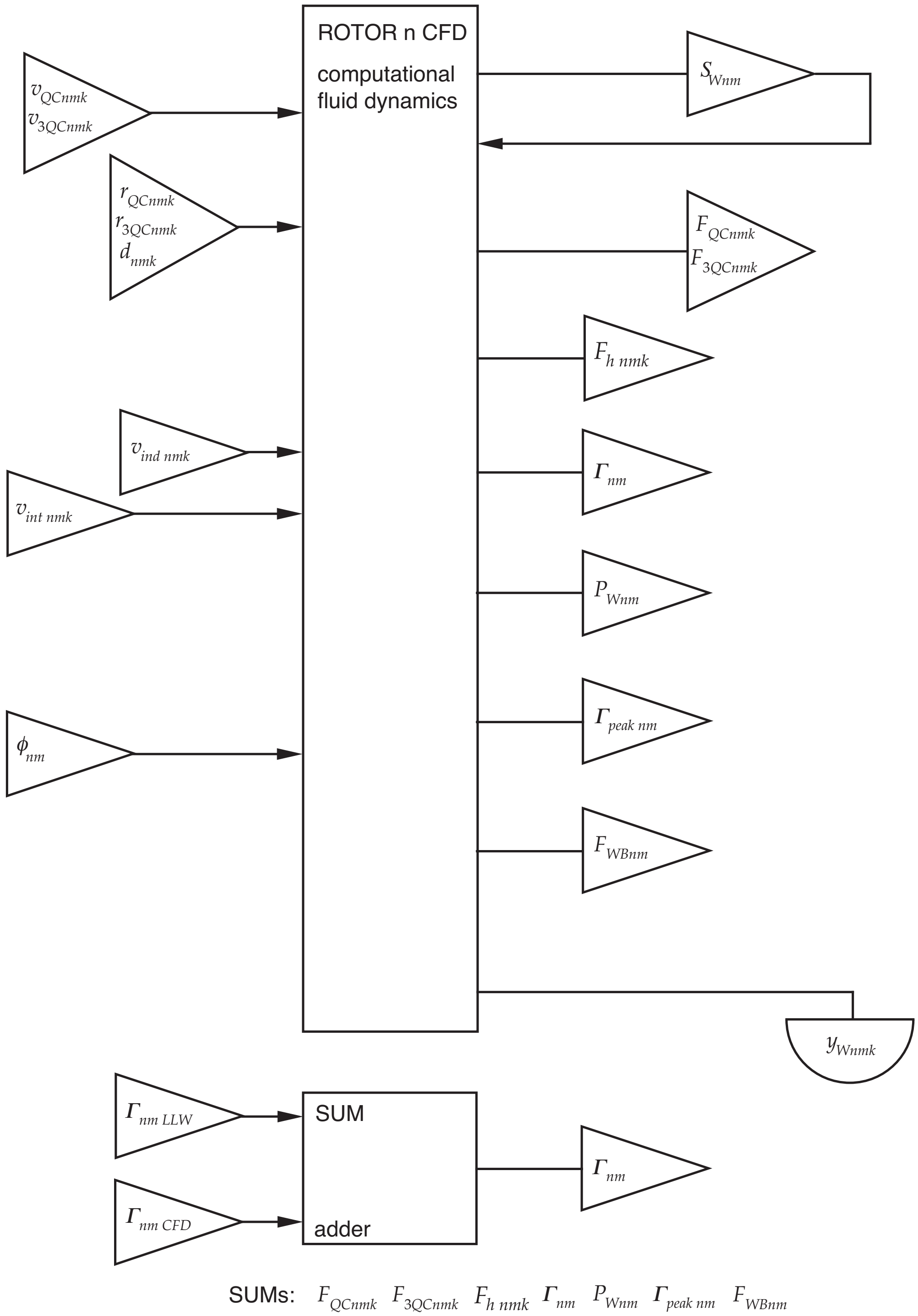


Figure 6-24 Computational fluid dynamics analysis.

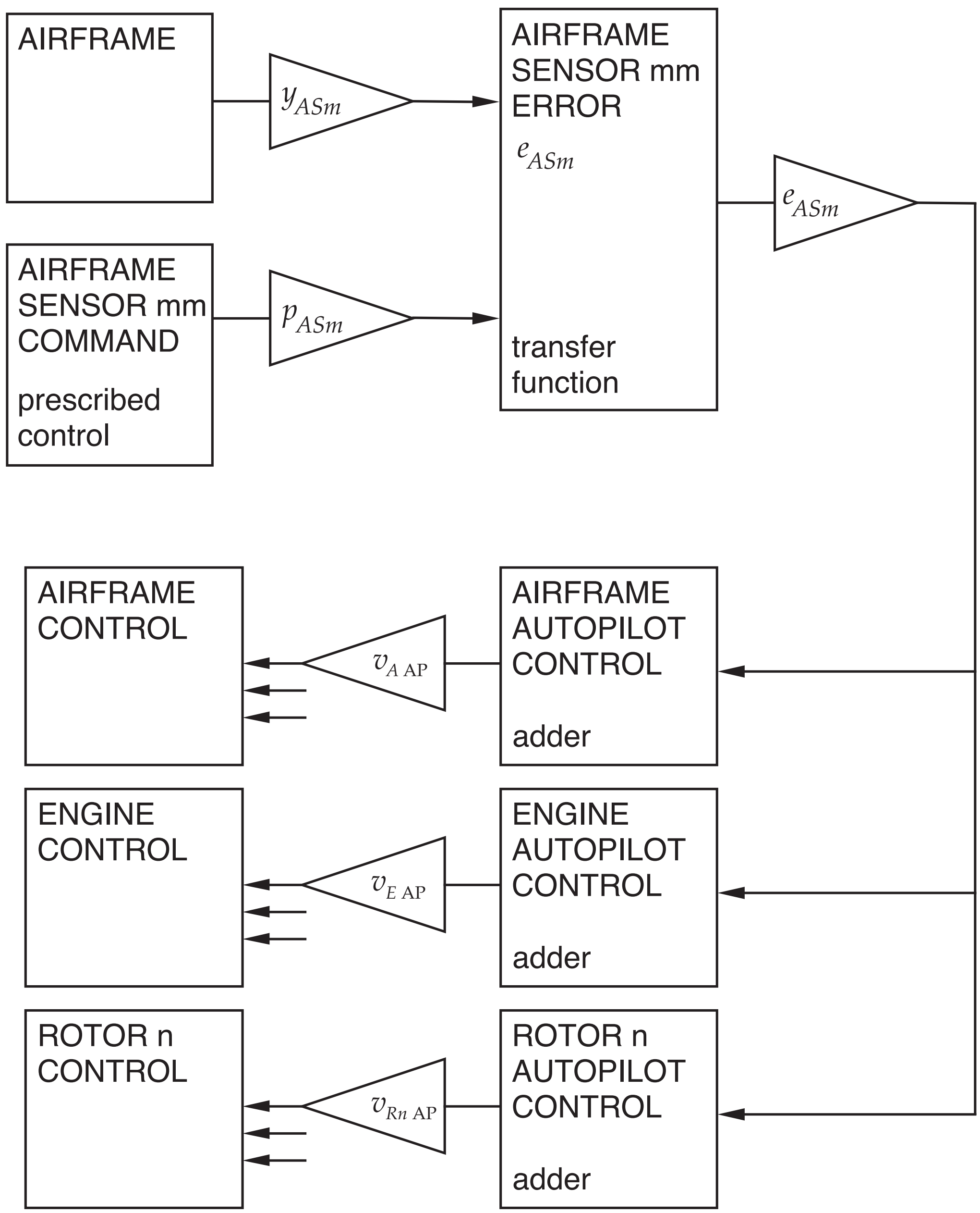


Figure 6-25 Autopilot.

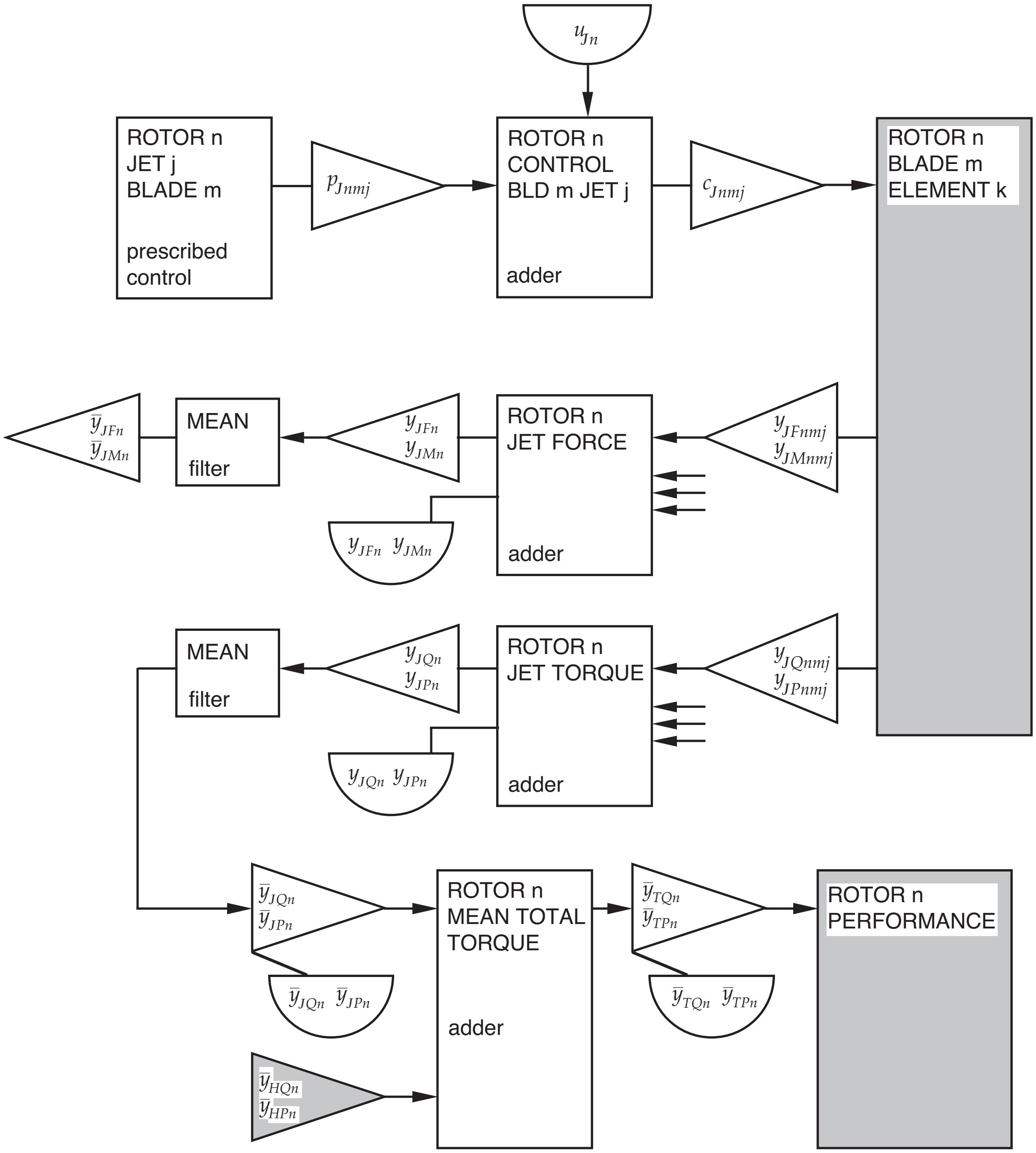


Figure 6-26 Rotor blade reaction jet control.

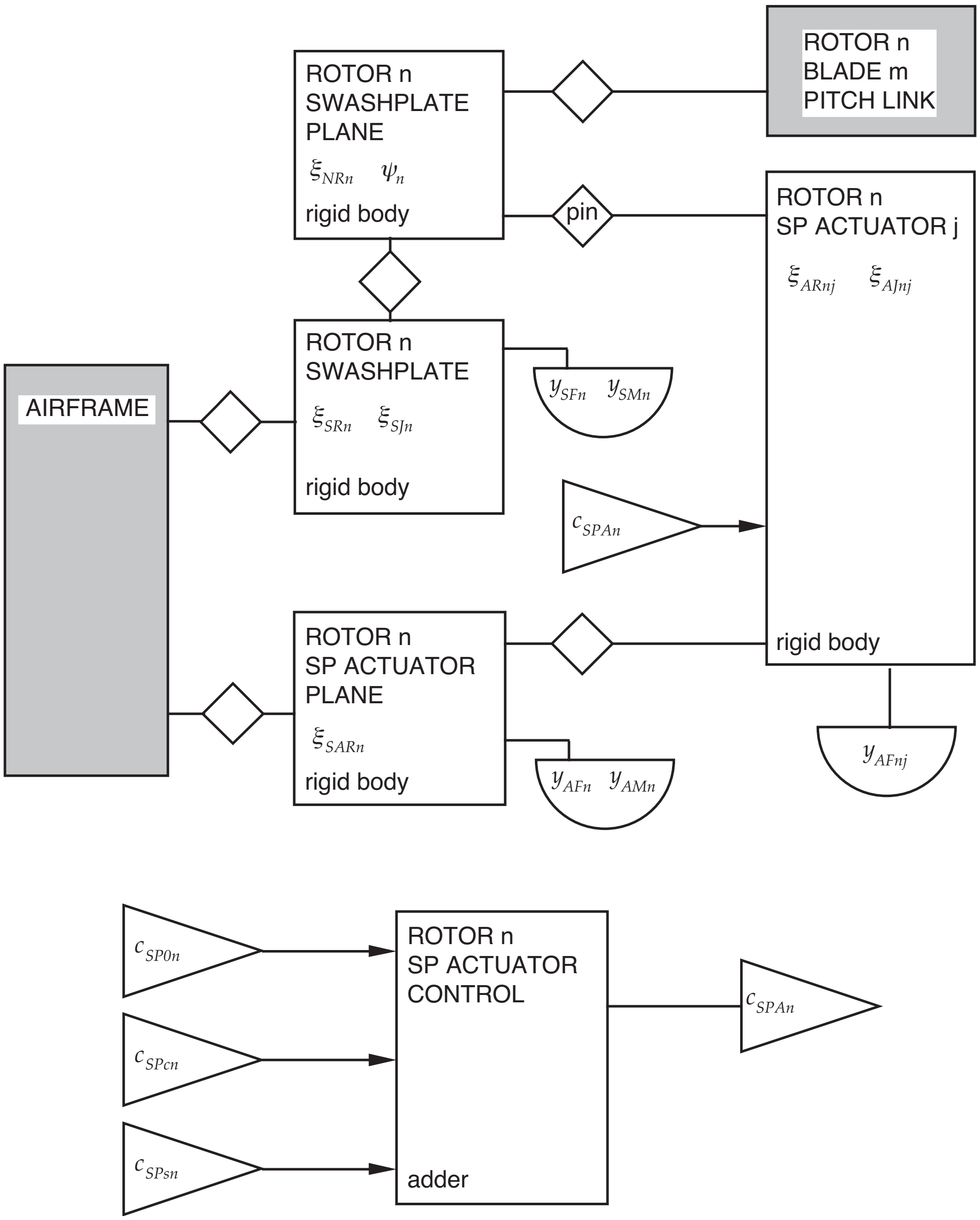


Figure 6-27 Rotor control system with nonrotating actuators.

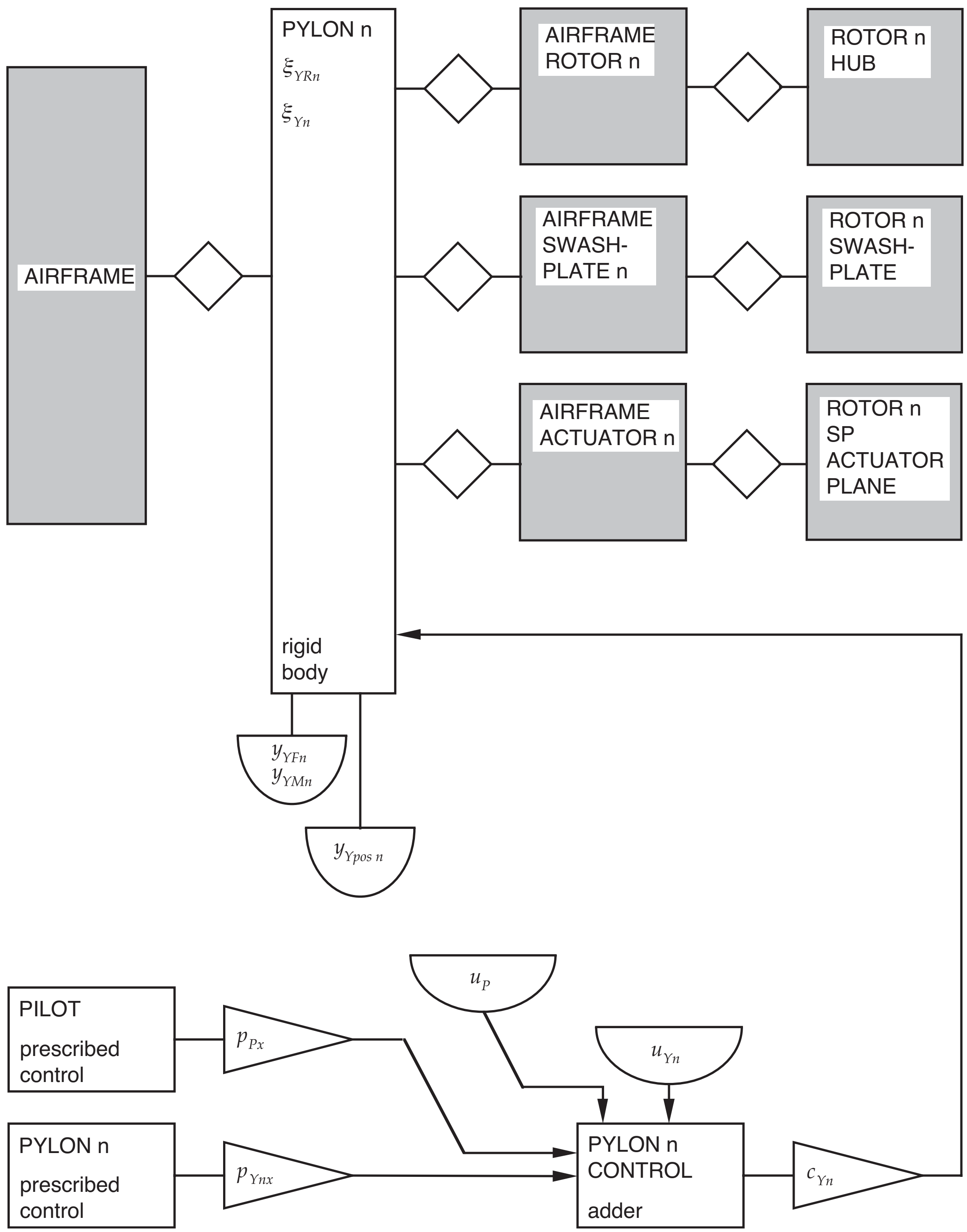


Figure 6-28 Rotor pylon.

6–3 Frames, Periods, and Weights

FRAMES

FRAMES ARE ESTABLISHED FOR AIRFRAME (F), ROTORS NONROTATING (S) AND ROTATING (R), AND BLADES (B)

frame degrees of freedom used for each

corresponding frame components are airframe, rotor hub, rotor rotating hub, and rotor blade root

figure 29 illustrates the conventions

INERTIAL FRAME

gravity acts in $+z$ direction; so z-axis positive downward

FRAME = `AIRFRAME`

F FRAME IS BASE FRAME FOR SYSTEM

parent is inertial frame

angular motion of F measured from orientation of inertial axes

FREE FLIGHT

airframe is frame component

airframe constructed so normally F axes are center-of-mass, mean axes of system, with origin at rotorcraft center of gravity

x-axis forward, with aircraft velocity positive in x direction

y-axis to right; z-axis down

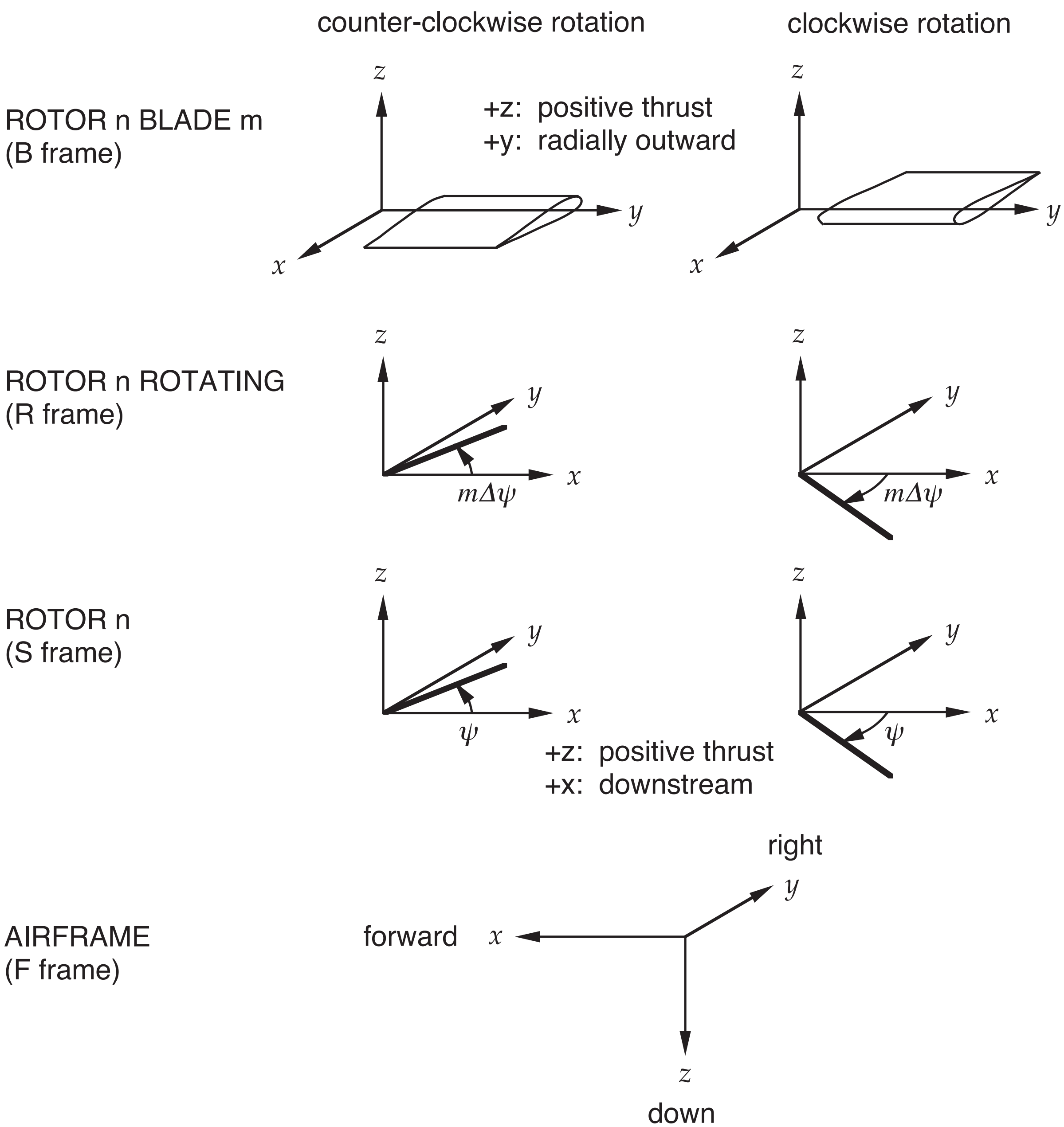


Figure 6-29 Rotorcraft frames.

degrees of freedom are typically body axis velocity and Euler angles (aircraft convention)

yaw about z-axis, then pitch about y-axis, then roll about x-axis

body axis velocity representation does not produce complete calculation of frame displacement

can use inertial axis displacement for linear degrees of freedom, if rotorcraft position relative inertial frame is important (such as for ground effect, or for wake geometry in transients)

WIND TUNNEL

airframe is constrained component

F axes are axes of wind tunnel test module, origin at center of test module

x-axis forward, with wind positive from x direction

y-axis to right; z-axis down

fixed orientation of F frame relative inertial frame, specified by Euler angles

yaw about z-axis, then pitch about y-axis, then roll about x-axis

velocity of air given by wind speed (flight speed not used)

FRAME = `ROTOR n`

S FRAME IS NONROTATING HUB FRAME (SHAFT AXES) OF ROTOR

parent is airframe frame

z-axis is shaft axis, positive in positive thrust direction

rotor rotates about z-axis, clockwise or counter-clockwise

x-axis positive downstream

blade azimuth angle measured from x-axis, in direction of rotor rotation

y-axis is on advancing side for counter-clockwise rotation, on retreating side for clockwise rotation

x–y plane is rotor hub plane; origin of frame is at center of rotation

S AXES ARE WING SET AXES (REFERENCE AXES OF ROTOR AERODYNAMICS)

MOTION REPRESENTATION

if airframe is rigid body, then S frame has constant position and orientation relative F frame

if airframe has elastic motion, then S frame motion includes effects of airframe dynamics

FRAME = `ROTOR n ROTATING`

R FRAME IS ROTATING HUB FRAME (SHAFT AXES) OF ROTOR

parent is rotor frame

z-axis is shaft axis, positive in positive thrust direction

origin of frame is at center of rotation

MOTION REPRESENTATION

nonrotating hub component has no elastic motion, and joint at origin to produce rotor rotation

origin of rotating hub component attached to joint

so R frame has same origin as S frame, and is rotated from S frame by angle ψ about shaft

ψ = azimuth of reference blade ($m = N$)

without drive train model, ψ is prescribed rotation of joint, at constant angular speed Ω

with drive train model, ψ is degree of freedom

FRAME = `ROTOR n BLADE m`

B FRAME IS ROTATING FRAME OF m-th BLADE OF ROTOR ($m = 1$ TO N)

separate frame for each blade, parent is rotor rotating frame

z-axis is shaft axis, positive in positive thrust direction
y-axis is spanwise, positive radially outward
x-axis is chordwise

x-axis positive towards blade trailing edge for counter-clockwise rotation

x-axis positive towards blade leading edge for clockwise rotation

origin of frame is at center of rotation

B AXES ARE WING AXES (REFERENCE AXES OF BLADE AERODYNAMICS)

MOTION REPRESENTATION

rotating hub component has no elastic motion

origin of blade root component attached to rotating hub component, possibly through gimbal joint

so B frame has same origin as R frame, and is rotated from R frame by angle $(m\Delta\psi - 90)$ about shaft

$\Delta\psi = 2\pi/N$ = interblade spacing

blade azimuth angle is $\psi_m = \psi + m\Delta\psi$

for rotor with unequal blade spacing, specified blade azimuth angle $\Delta\psi_m$ replaces $m\Delta\psi$

AXES OF BLADE ELEMENT

axes of rigid body motion (origin at inboard end of element), following conventions of beam component

x-axis is beam axis of element, approximately spanwise, positive radially outward

z-axis is approximately parallel to shaft axis, positive in positive thrust direction

y-axis is in x–y plane of B frame, approximately chordwise

- y-axis positive towards blade leading edge for counter-clockwise rotation

- y-axis positive towards blade trailing edge for clockwise rotation

PERIODS

ROTATIONAL SPEED OF FIRST ROTOR (Ω_1) IS BASE PERIOD FOR ROTORCRAFT

all other periods are children of Ω_1

PERIODS REQUIRED

period name	frequency
`ROTOR 1`	Ω_1
`ROTOR n`	$\Omega_n = r_n\Omega_1$
`ROTOR n BLADE PASSAGE`	$N_n\Omega_n$
`DRIVE TRAIN INTERCONNECT SHAFT`	$r_I\Omega_1$
`ROTORCRAFT FLUTTER`	$r_f\Omega_1$
`ROTOR n FLUTTER`	$r_f\Omega_n$
`ROTOR n TRIM`	$\Omega_n/K_n = (r_n/K_n)\Omega_1$
`AIRFRAME TRIM`	$(1/K_A)\Omega_1$
`DRIVE TRAIN TRIM`	$(1/K_D)\Omega_1$

r_n = gear ratio for n-th rotor shaft
r_I = gear ratio for interconnect shaft
N_n = number of blades for n-th rotor
r_f = flutter gear ratio (used in flutter analysis)
K = number of periods (used in trim part solution)

WEIGHTS

WEIGHTS SYSTEM PIECE DEFINED FOR EACH ROTOR

name = `ROTOR n`

required to provide standard weights for perturbation and convergence of variables

6–4 Control System

ROTORCRAFT CONTROL SYSTEM

SET OF CONTROL VARIABLES c
CALCULATED FROM SYSTEM INPUT VECTORS u
AND PRESCRIBED CONTROL VARIABLES p

control c (interface variables) used by rest of system as required

generally units = degrees for input vectors, and radians for control vectors

except for tail boom and throttle variables,
and lb or N for forces

GUST VELOCITIES CALCULATED FROM INPUT VECTOR g

gust generally not used for trim task, but available in trim to simulate aerodynamic interference

figures 2 to 5, 21, 25 to 28 describe configuration constructed

ROTOR MODEL CONSTRUCTED CAN HAVE SWASHPLATE MECHANISM

WITHOUT SWASHPLATE:
rotor collective and cyclic control produce rotating frame pitch control (at blade pitch joint)

IF SWASHPLATE EXISTS:
swashplate used to generate rotor control, perhaps with individual-blade-control

ROTOR BLADE AERODYNAMIC MODEL CAN HAVE ONE OR MORE TRAILING-EDGE FLAPS

perhaps connected to rotor primary control

ROTOR BLADE CAN HAVE ONE OR MORE REACTION JETS FOR CONTROL

modelled as applied forces acting on beam elements

PRESCRIBED CONTROL ONLY USED FOR TRANSIENT TASK

CONSTRUCT PRESCRIBED CONTROL p CORRESPONDING TO ALL INPUT VECTORS u AND g

except for higher harmonic pitch and force

SEPARATE PRESCRIBED CONTROL COMPONENT CONSTRUCTED FOR EACH ELEMENT OF u AND g

shell defines different amplitude but same time history shape for these prescribed control components

core input can be used to define different time history shapes for individual elements

AUTOPILOT CAN BE CONSTRUCTED FOR TRANSIENT TASK

using designated airframe sensors

core input required to complete definition: time constants and gains; perhaps more states (poles and zeros in transfer function), different sensors, nonlinear output

INPUT VECTORS

VALUES OF INPUT VARIABLES REQUIRED FOR TRIM, TRANSIENT, AND FLUTTER TASKS

INPUT VARIABLES CONSTRUCTED (VECTORS)

pilot stick: $u_P = (\delta_0, \delta_c, \delta_s, \delta_p, \delta_t)^T$

airframe aerodynamic controls: $u_A = (\delta_f, \delta_e, \delta_a, \delta_r)^T$

pylon controls: $u_{Yn} = (\delta_{Yc}, \delta_{Yp}, \delta_{Yy})^T$

tail boom controls: $u_T = (\delta_b, \delta_j)^T$

engine throttle: $u_E = (\theta_{tn})$

auxiliary force: $u_F = (F_m)$

collective governor: $u_G = (\delta_g)$

rotor primary controls: $u_{Rn} = (\theta_0, \theta_{1c}, \theta_{1s})^T$

blade pitch: $u_{Bn} = (\theta_m)$

blade trailing-edge flap: $u_{Dnj} = (\phi_m)$

blade reaction jet: $u_{Jn} = (J)$

rotating frame HHC pitch:

$$u_{BHn} = (0, 0, 0, \theta_{2c}, \theta_{2s}, \ldots, \theta_{kc}, \theta_{ks})^T$$

rotating frame higher harmonic trailing-edge flap:

$$u_{DHnj} = (0, \phi_{1c}, \phi_{1s}, \phi_{2c}, \phi_{2s}, \ldots, \phi_{kc}, \phi_{ks})^T$$

nonrotating frame HHC swashplate control:

$$u_{RH0n} = (0, (\theta_{pNc})_0, (\theta_{pNs})_0)^T$$
$$u_{RHcn} = (0, (\theta_{pNc})_{1c}, (\theta_{pNs})_{1c})^T$$
$$u_{RHsn} = (0, (\theta_{pNc})_{1s}, (\theta_{pNs})_{1s})^T$$

higher harmonic auxiliary forces:

$$u_{FHnm} = (0, (F_{pNc})_m, (F_{pNs})_m)^T$$

gust velocities: $g = (u_G, v_G, w_G)^T$

SIGN CONVENTIONS

PILOT'S CONTROLS

collective stick, δ_0	positive up
lateral cyclic stick, δ_c	positive right
longitudinal cyclic stick, δ_s	positive forward
pedal, δ_p	positive yaw right
throttle, δ_t	positive increase power

typically then positive pilot's longitudinal and lateral cyclic stick produce respectively $(-\sin\psi)$ and $(-\cos\psi)$ variation of blade pitch

GUST VARIABLES ARE UNIFORM VELOCITY PERTURBATIONS, IN NEGATIVE WIND/GUST AXES

longitudinal gust velocity, u_G	positive from forward
lateral gust velocity, v_G	positive from right
vertical gust velocity, w_G	positive from below

CALCULATION OF CONTROL VECTORS

CONTROL VECTORS c (INPUT/OUTPUT INTERFACE VARIABLES) CALCULATED FROM INPUT VECTORS u AND g, AND PRESCRIBED CONTROL VARIABLES p

OPERATIONS INCLUDE:

- connection of pilot's controls to airframe, pylon, tail boom, engine, auxiliary force, and rotor controls
    - connection defined in terms of control matrix T_C
- conversion of units from degrees to radians
- evaluation of control in time domain from harmonics
- incorporation of feedback variables
    - collective governor = feedback from drive train mean governor (throttle governor is part of drive train model)

CONTROL MATRICES

CONTROL SYSTEM MODEL CONNECTS PILOT'S CONTROLS TO AIRFRAME, PYLON, TAIL BOOM, ENGINE, AUXILIARY FORCE, AND ROTOR CONTROLS

connection defined in terms of control matrices T_C

linear relation, typically of form:

$$c_X = \frac{1}{57.3}\left(T_{CX} u_P + u_X\right)$$

u_X = input vector specifying control position with all sticks centered ($u_P = 0$)

T_{CX} = control matrix

FOR MODERN ROTORCRAFT, COUPLING BETWEEN PILOT'S CONTROLS AND ROTOR/AIRCRAFT CONTROLS CAN BE QUITE COMPLEX

involving nonlinear functions and gain scheduling with flight condition

FOR TRIM TASK, ONLY NECESSARY TO CORRECTLY ACCOUNT FOR COUPLING BETWEEN REDUNDANT ROTOR AND AIRCRAFT CONTROLS

then trim solution will produce correct positions of individual controls

and true position of pilot's sticks (if required) can be determined from the actual control system geometry

CONTROL MATRICES T_C

INPUT DIRECTLY

OR USE STANDARD FORMS, CALCULATED FROM GAIN FACTORS AND SWASHPLATE PHASE ANGLES

STANDARD FORMS

CAMRAD II Documentation, Volume III, Rotorcraft Theory; Chapter “Control System”

for airframe, tail boom, and engine

for auxiliary forces

- first auxiliary force for propulsion (connected to throttle)
- second auxiliary force for anti-torque (connected to pedal)

for rotor controls, depending on configuration:

configuration	first rotor
one rotor	
single main-rotor and tail-rotor	main rotor
tandem main-rotor	front rotor
coaxial main-rotor	lower rotor
tilting proprotor aircraft	right rotor

6–5 Airframe

ROTORCRAFT AIRFRAME CONSISTS OF AN ELASTIC BODY WITH ONE OR MORE ROTORS ATTACHED

- FREE FLIGHT OR WIND TUNNEL
  - for free flight: rigid and elastic airframe motion
  - in wind tunnel: rotor support can be flexible; no rigid body motion of system

- ELASTIC MOTION DESCRIBED BY LINEAR MODES
  - linear normal modes component
  - modes often obtained from large finite-element analysis of structure (free-vibration modes for free flight; constrained modes for wind tunnel)

- QUASISTATIC MODEL FOR AERODYNAMIC FORCES AND MOMENTS
  - rigid airframe aerodynamics component
    - nonlinear model for trim task
    - nonlinear or stability derivatives model for flutter and transient tasks
  - typically based on wind tunnel measurements of aerodynamic characteristics

AIRFRAME-INDUCED INTERFERENCE VELOCITY AT ROTORS

calculated using simplified representation of airframe

or interference velocities can be obtained from large panel-method analysis

TAIL BOOM

optional circulation-control boom and reaction jet

torque reaction and yaw control for single main-rotor helicopter configuration

programmable component, so model can be user-defined; interfaces constructed should still be appropriate

ROTOR PYLON

airframe model can include pylon for each rotor

attached to airframe at a pivot with spring, damper, and control

hub, swashplate, and swashplate actuators attached to pylon

SLUNG LOADS

one or more slung loads can be attached to airframe

load suspension is one or more cables from each load to airframe

figures 6, 19, 22, and 28 describe configuration constructed

FRAMES AND INERTIA

FIGURES 30 AND 31 SUMMARIZE FRAMES, AND CONVENTIONS FOR DESCRIBING INERTIA

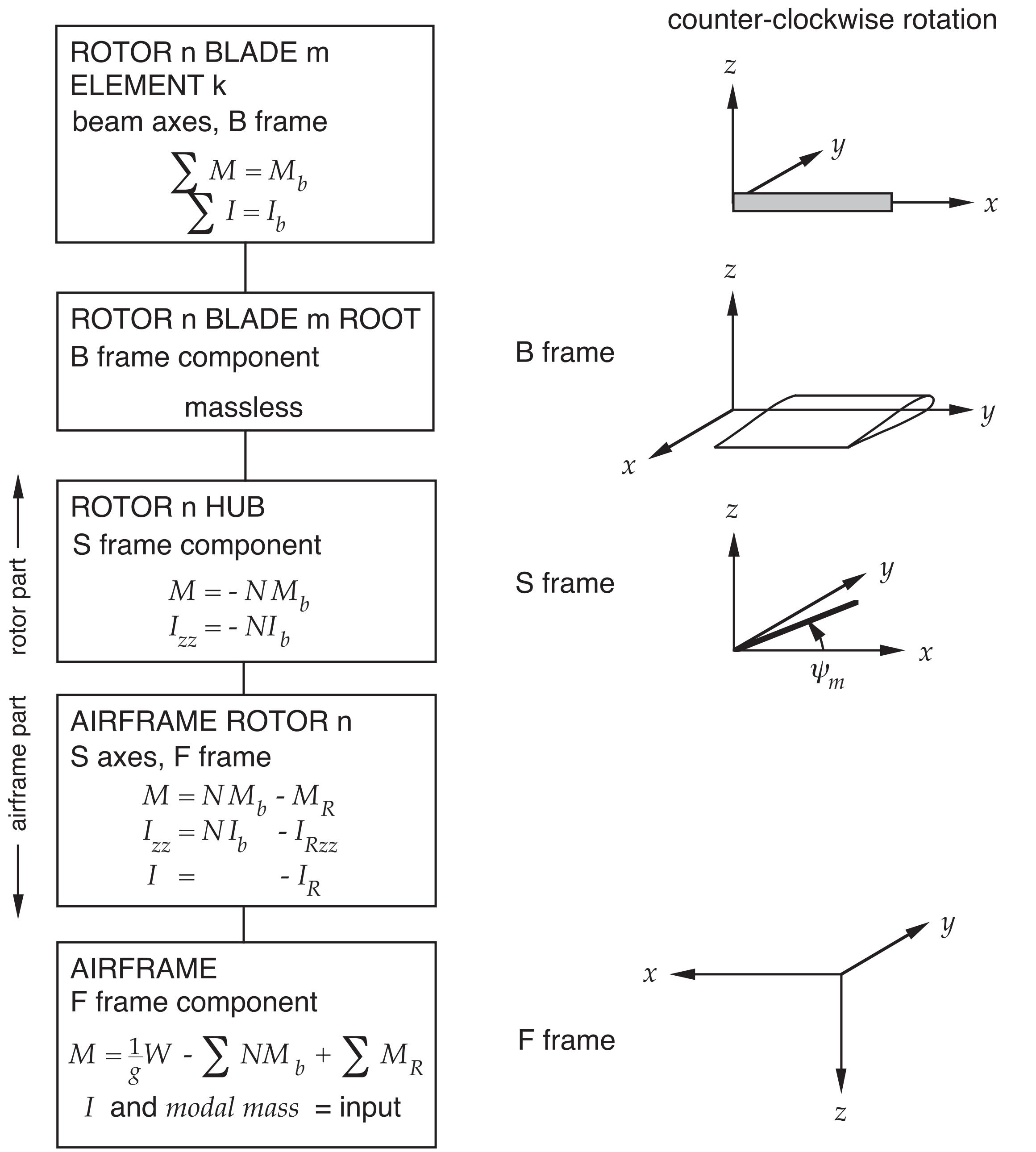
- orientation of blade and element axes depends on rotor direction of rotation

FOR FREE FLIGHT, AIRFRAME IS CONSTRUCTED SO F AXES ARE NORMALLY CENTER-OF-MASS, MEAN AXES OF SYSTEM, WITH ORIGIN AT ROTORCRAFT CENTER OF GRAVITY

- by using the linear normal modes component, the rigid motion describes center-of-mass, mean axes motion of airframe component
- inertial properties of components defined so this property can hold for entire system

AIRFRAME ELASTIC MODES DESCRIBE ENTIRE SYSTEM, EXCEPT FOR EFFECTS OF ROTATING BLADES

- obtained by calculation using finite-element analysis, or by shake test of actual airframe
- various conventions in use by industry to simulate effects of rotor inertia in these modes
  - rotorcraft shell designed to accommodate these conventions

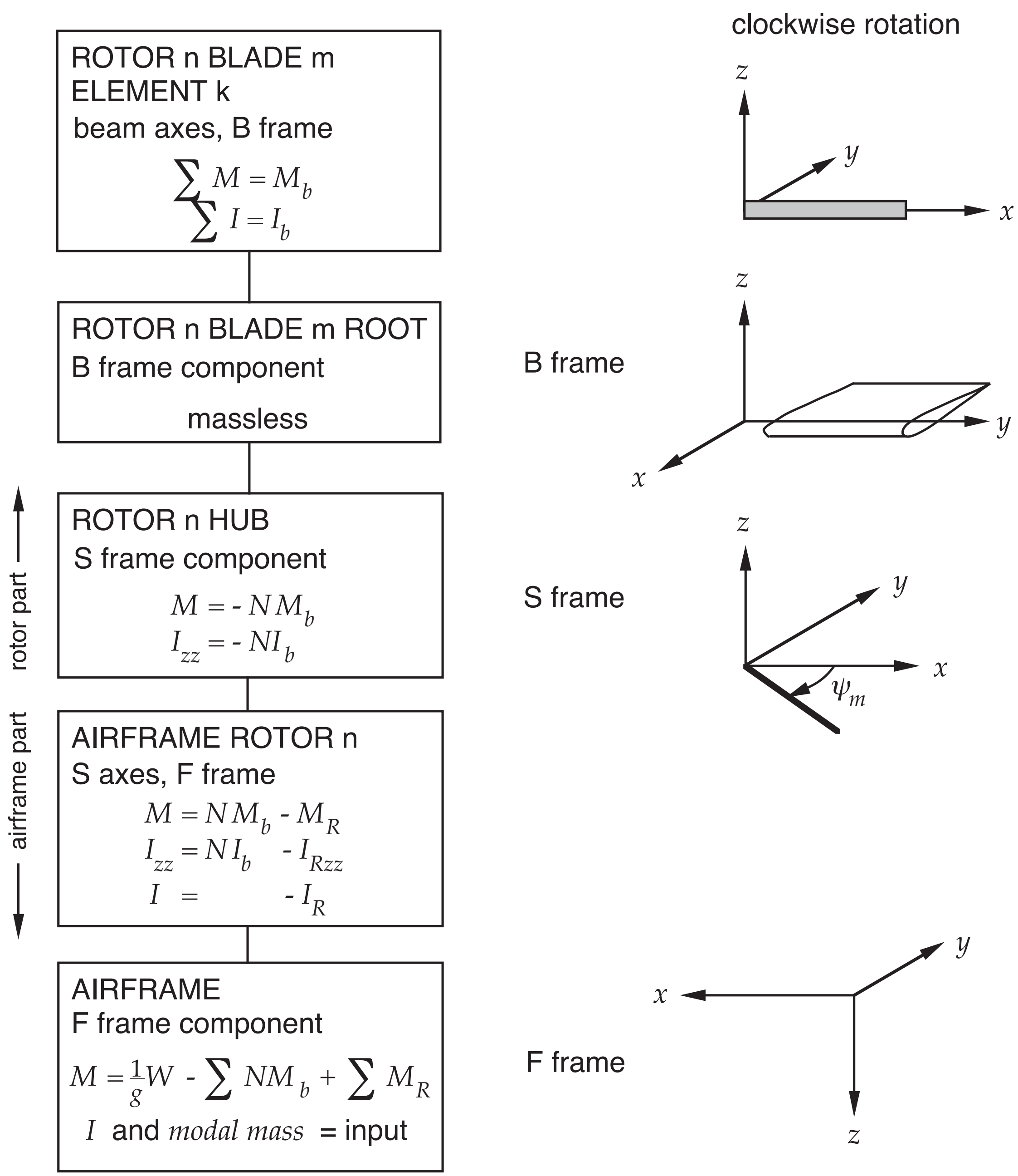


gross weight W and CG position input for entire system

inertia I and *modal mass* input without rotor inertia (M_b and I_b),
plus M_R and I_R at hub node (canceled by terms in **AIRFRAME ROTOR n**)

solution procedure requires rotor inertia terms (M_b and I_b) in
ROTOR n HUB and **AIRFRAME ROTOR n**

Figure 6-30 Rotorcraft frames and inertia (counter-clockwise rotation).



gross weight W and CG position input for entire system

inertia I and *modal mass* input without rotor inertia (M_b and I_b), plus M_R and I_R at hub node (canceled by terms in **AIRFRAME ROTOR n**)

solution procedure requires rotor inertia terms (M_b and I_b) in **ROTOR n HUB** and **AIRFRAME ROTOR n**

Figure 6-31 Rotorcraft frames and inertia (clockwise rotation).

INPUT DESCRIPTION OF SYSTEM INERTIAL PROPERTIES

definition of rotor structure (blades and flexbeams, perhaps with point masses and flaps) includes inertia

all in rotating frame, outboard of rotor blade root component

M_b = total mass of a blade
I_b = corresponding rotational moment of inertia

input of rotorcraft gross weight W and center of gravity:

for entire system, including rotors

input of rigid body moments of inertia I and elastic mode generalized masses:

for the system without rotating frame inertia (same inertia that gives M_b and I_b)

plus equivalent hub inertia consisting of mass M_R and moments of inertia I_R

equivalent hub inertia corresponds to configuration analyzed or tested to obtain the modal properties

typically $M_R = NM_b$

in practice many approaches are used to represent rotational inertia of rotor, and sometimes total system mass is not matched

figure 32 illustrates determination of equivalent hub mass M_R (similar for moments of inertia)

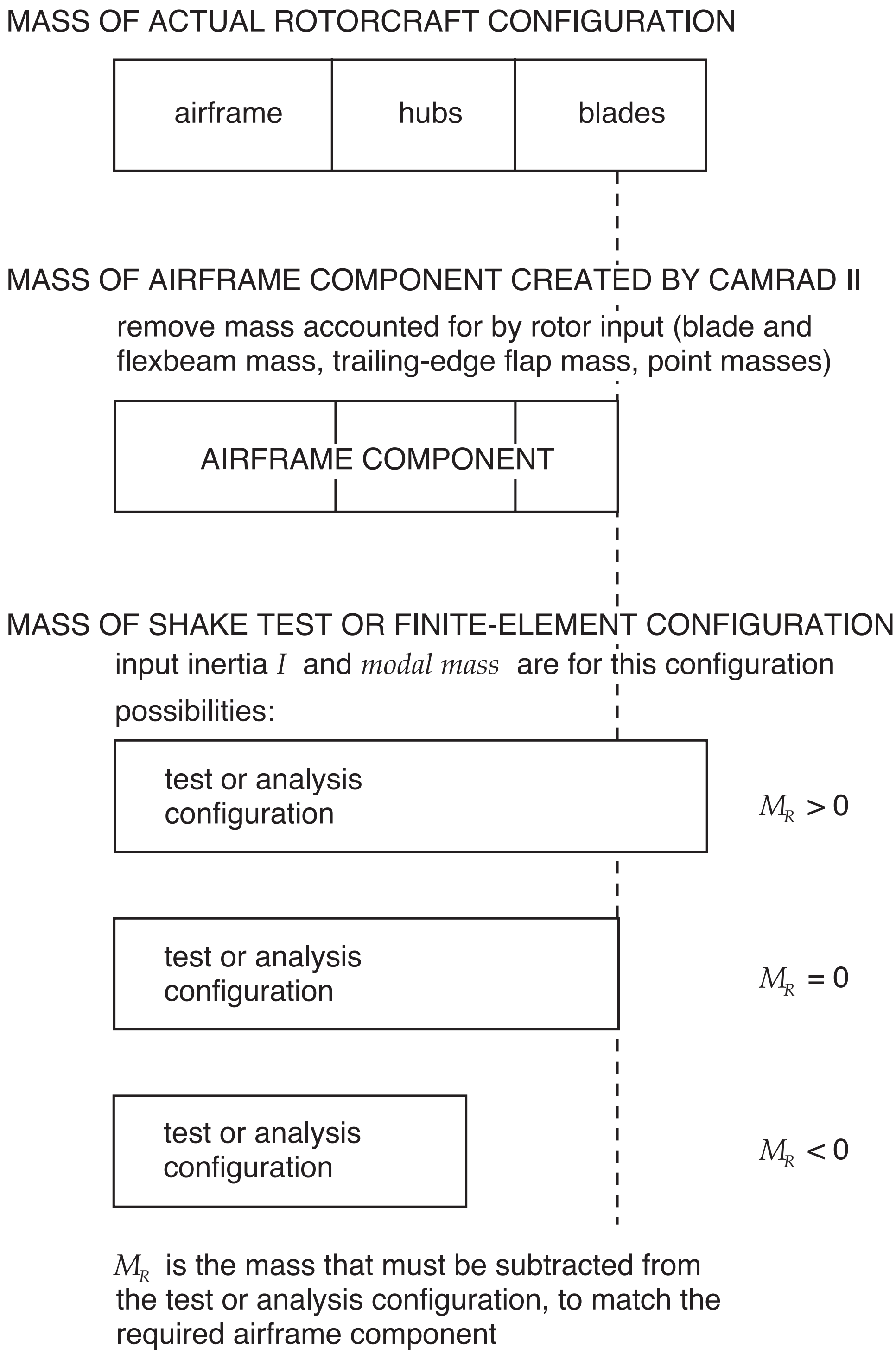


Figure 6-32 Determination of equivalent hub inertia.

INERTIAL PROPERTIES OF COMPONENTS ARE CALCULATED FROM INPUT DATA

rigid body component is placed at rotor hub node, to subtract equivalent hub inertia (M_R and I_R)

mass and center-of-gravity position of airframe component calculated

if $M_R = NM_b$, then F axes are motion of system center-of-gravity

for convergence of motion loop (iteration between rotor and airframe solutions), inertial reactions of rotor mass to hub motion should be on airframe side

so rotor mass NM_b and rotational inertia NI_b added to rigid body component at hub node (on airframe side)

and subtracted from rotor hub component (on rotor side)

as shown in figures 30 and 31

GEOMETRY

AIRFRAME GEOMETRY DESCRIBED IN TERMS OF LOCATIONS OF STRUCTURAL DYNAMIC COMPONENT

LOCATIONS USED (IN ORDER DEFINED):

rotor hub: for each rotor

rotor swashplate: for each rotor

general locations: for swashplate nonrotating actuators, auxiliary forces, tail boom, sensors, airframe wings and bodies, slung loads, and rotor-induced velocity calculation

airframe aerodynamic collocation points: wing-body, horizontal tail, vertical tail, and stability derivative

POSITION OF LOCATION SPECIFIED IN TERMS OF REFERENCE AXES

reference axes are parallel to F axes, but have arbitrary origin and different directions (figure 33)

fuselage station (FS)	positive aft
buttline (BL)	positive right
waterline (WL)	positive up

GENERAL LOCATIONS

ORIENTATION AT GENERAL LOCATION SPECIFIED IN TERMS OF AZIMUTH ANGLE AND ELEVATION ANGLE, RELATIVE F AXES

azimuth angle ψ = rotation about positive z-axis

elevation angle θ = rotation about resulting y-axis, measured from x–y plane of F axes

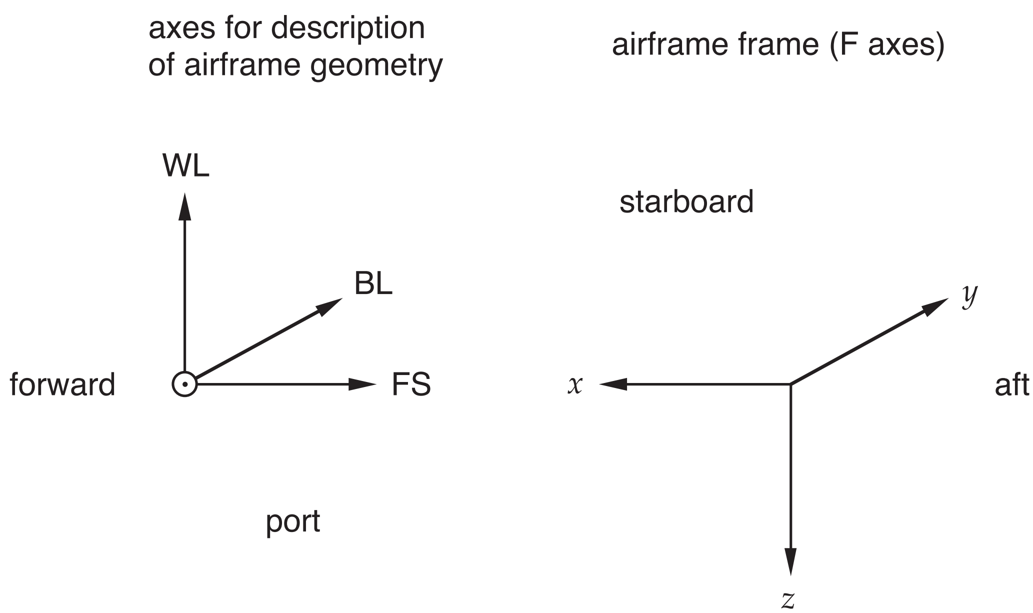


Figure 6-33 Definition of aircraft geometry.

ROTOR LOCATIONS

ORIENTATION OF LOCATION AXES AT ROTOR HUB = ORIENTATION OF UNDEFLECTED NONROTATING HUB FRAME (S AXES) RELATIVE AIRFRAME FRAME (F AXES)

position is center of rotation

z-axis is shaft axis, positive in positive thrust direction

x-axis is positive downstream, azimuth measured from x-axis

SPECIFICATION (RELATIVE BASIC ORIENTATION WITH S AXES PARALLEL F AXES)

for main rotor

first shaft roll ϕ_R (positive right)

then shaft angle of attack θ_R (positive rearward)

then azimuth reference angle ψ_R about final z-axis (thrust axis)

for tail rotor

first shaft cant ϕ_R (positive up)

then shaft angle of attack θ_R (positive rearward)

then azimuth reference angle ψ_R about final z-axis (thrust axis)

SWASHPLATE LOCATIONS

axes at swashplate location are parallel to rotor S axes

on shaft (z axis), distance h_{SP} below hub

TILTING PROPROTOR AIRCRAFT GEOMETRY DEFINED IN HELICOPTER MODE, FOR RIGHT ROTOR

nacelle tilt angle α is zero for airplane mode, and 90 degrees for helicopter mode

specification of rotor orientation (at $\alpha = 90$)

first shaft roll ϕ_R (positive right)

then shaft angle of attack θ_R (positive rearward)

then azimuth reference angle ψ_R about final z-axis (thrust axis)

nacelle pivots about axis with sweep ψ_P (positive aft) and dihedral ϕ_P (positive up) relative the y-axis of F frame

position of pivot and right rotor hub are specified for helicopter mode ($\alpha = 90$)

usually geometry at left rotor calculated assuming that aircraft is symmetric about $\mathrm{BL} = 0$

or geometry can be separately specified for left rotor, to model configuration that is not symmetric

with more than two rotors, first two rotors always tilt, other rotors may or may not tilt

if all rotors tilt, aircraft may or may not be symmetric

MODE SHAPES OF ELASTIC MOTION

MODES SHAPES DESCRIBED BY LINEAR COMPONENTS ξ AND ANGULAR COMPONENTS γ

at each airframe location

input in F axes, or in undeflected location axes

ASSUMING THAT GENERALIZED COORDINATE q_k HAS UNITS OF FEET OR METERS, MODAL PROPERTIES HAVE FOLLOWING UNITS:

generalized mass M_k: slug or kg
linear mode shape ξ_k: ft/ft or m/m (dimensionless)
angular mode shape γ_k: rad/ft or rad/m

ELASTIC MODES CAN BE ARBITRARILY SCALED

if ξ_k and γ_k multiplied by factor S (at all locations)

then M_k should be multiplied by S^2

and solution for q_k will be divided by S

AIRFRAME FINITE-ELEMENT ANALYSIS MIGHT USE INCHES FOR LENGTH UNITS

to convert to feet: angular modes shapes γ_k and generalized mass M_k multiplied by 12

COORDINATE SYSTEM OF FINITE-ELEMENT ANALYSIS CAN HAVE x-AXIS POSITIVE AFT, y-AXIS POSITIVE TO RIGHT, AND z-AXIS POSITIVE UPWARD

to convert to F axes, change signs of x and z-components of linear and angular mode shapes (ξ_k and γ_k)

EFFECTS OF ELASTIC MOTION AND FORCES ARE SUPPRESSED AT AIRFRAME AERODYNAMIC COLLOCATION POINTS

- so mode shapes not required at these locations

OPTIONALLY, CAN CALCULATE MODE SHAPES AT SWASHPLATE LOCATIONS FROM MODE SHAPES AT ROTOR HUB LOCATIONS

- assuming rigid motion of shaft
- so there will be no pitch/mast-bending coupling

TRANSMISSION COMPONENTS USED TO MODEL DRIVE TRAIN

- so actual load paths between rotor, airframe, and drive train are not modelled
- and rotor torque always reacted at hub node of airframe (and perhaps at swashplate node), with or without drive train model

- because true load path through transmission is not modelled, may be appropriate to suppress effect of rotor torque on airframe elastic modes
  - accomplish by setting to zero the shaft-axis component of angular mode shape at hub (and swashplate)

AUXILIARY FORCES

AUXILIARY FORCES CAN BE APPLIED TO AIRFRAME, USING AN APPLIED LOAD INTERFACE

location and axis at that location are identified for each force

forces are connected to control vectors

CONVENTIONS FOR CONTROL MATRIX

first auxiliary force for propulsion (connected to throttle)

second auxiliary force for anti-torque (connected to pedal)

FOLLOWING SOURCES CONSTRUCTED:

constant forces, from input vector u_F

- can represent fixed auxiliary lift or propulsive force

forces proportional to the pilot control, from the input vector u_P

- can be used to trim rotorcraft
- both u_F and u_P terms can be used to perturb system in transient or flutter

higher-harmonic forces, from the input vector u_{FHnm}

- can represent effects such as fuselage loads produced by rotor wake impingement

prescribed forces p_{Fm}, generated for the transient task

6–6 Drive Train

ROTORCRAFT DRIVE TRAIN CONSISTS OF ONE OR MORE ROTORS, AND ONE OR TWO ENGINES, CONNECTED THROUGH FLEXIBLE TRANSMISSION

- DRIVE TRAIN LOAD PATHS CONSIDER TORQUE BALANCE ONLY
  - rotor torque acting on airframe is reacted at the hub component, with or without a drive train model

- ROTOR SPEED GOVERNOR IS INCLUDED, USING COLLECTIVE OR THROTTLE FEEDBACK OF ROTOR SPEED ERROR
  - governor dynamics represented by second-order transfer function

- DRIVE TRAIN CONNECTED TO JOINT ON HUB COMPONENT, AND THUS TO THE ROTOR
  - when swashplate is constructed, there is also a transmission component connecting hub joint and rotating swashplate
    - so swashplate azimuth and rotor azimuth remain consistent

ROTORCRAFT CAN BE CONSTRUCTED WITHOUT DRIVE TRAIN SUBSYSTEM

> then rotor and swashplate rotate at constant speed, determined by prescribed motion of hub and swashplate joints

ROTOR CAN HAVE FREE-ROTATION HUB

> hub joint is degree of freedom
>
> connected to swashplate (if it exists) but not connected to drive train

HELICOPTER HAS OVERRUNNING CLUTCH THAT DISCONNECTS ROTORS FROM ENGINE (BUT NOT FROM EACH OTHER) AT ZERO OR NEGATIVE TORQUE

> so for autorotation operation, input parameters should be set to eliminate engine and engine shaft, throttle input, and governor
>
> for operation with engines out (but still connected to rotors), input parameters should be set to eliminate engine damping, throttle input, and governor

figure 7 describes configuration constructed

DRIVE TRAIN INERTIAL PROPERTIES

FIGURE 34 SUMMARIZES DRIVE TRAIN CONNECTIONS, AND CONVENTIONS FOR DESCRIBING INERTIA

rotational inertia is specified for rotor and engine shafts

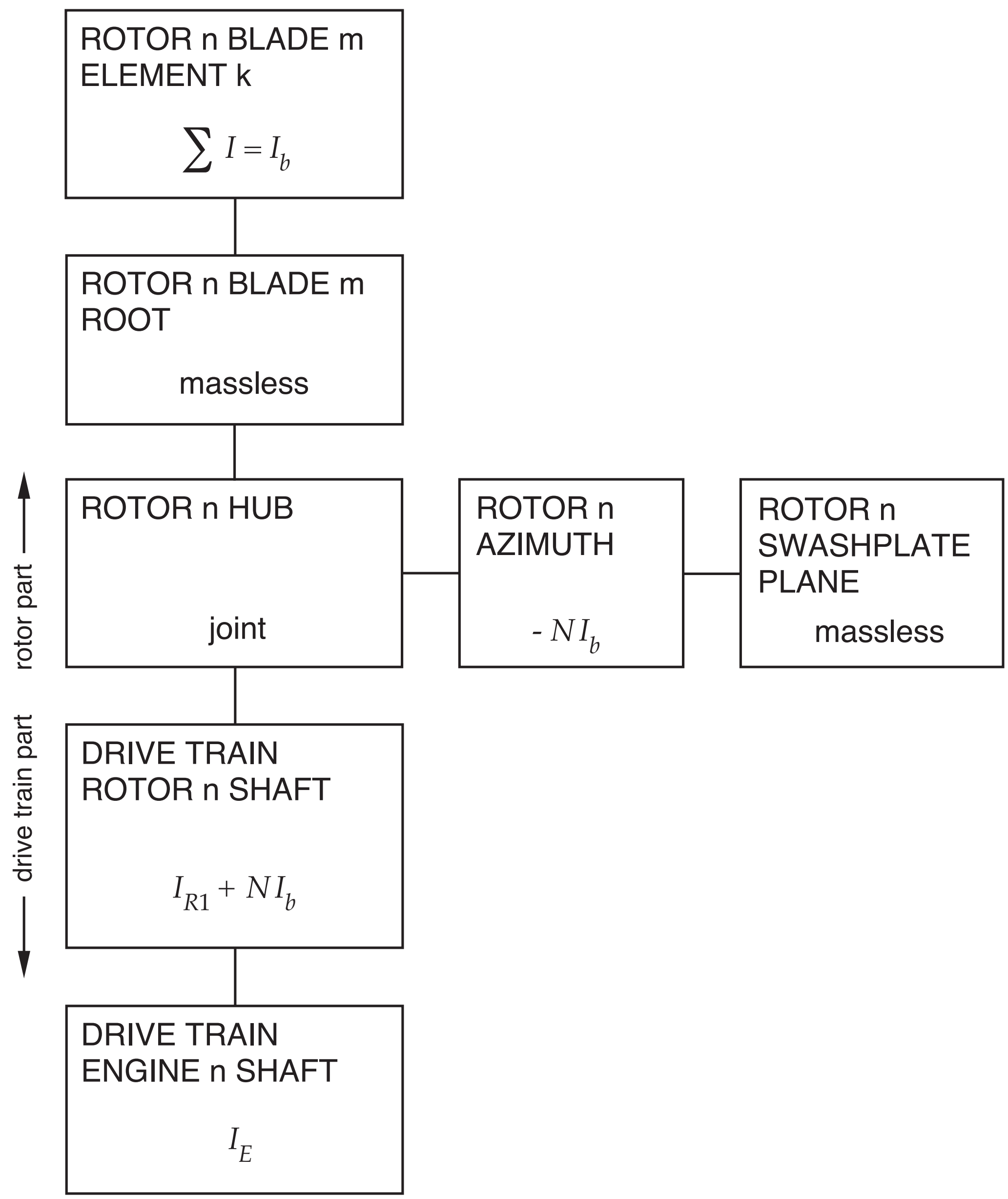
I_b = total rotational inertia of a blade

INERTIAL PROPERTIES OF COMPONENTS ARE CALCULATED FROM INPUT DATA

for convergence of motion loop (iteration between rotor and drive train solutions), inertial reactions of rotor mass to hub motion should be on drive train side

so rotor rotational inertia NI_b added to rotor shaft (on drive train side)

and subtracted from rotor azimuth component (on rotor side)



solution procedure requires rotor inertia terms (I_b) in
ROTOR n AZIMUTH and **DRIVE TRAIN ROTOR n SHAFT**

Figure 6-34 Drive train connections and inertia.

TRANSMISSION CONFIGURATIONS AND MOTION

CONFIGURATIONS ILLUSTRATED IN FIGURE 35

conventions

- solid dots define structural dynamic interfaces (torque kind) between components
- circles indicate gear ratios for branches of transmission
- squares indicate rotational inertia of root or branch
  - control torque Q_E is applied load on engine branch
  - also a damping term for engine branch
- zig-zag structures indicate rotational springs of branches
  - also a damper for each branch
- there is an inertia corresponding to each degree of freedom (solution procedure requires that there be no massless elastic degrees of freedom)
- elastic motion of any branch can be suppressed

TRANSMISSION CONFIGURATION MUST BE CONSISTENT WITH ROTORCRAFT CONFIGURATION

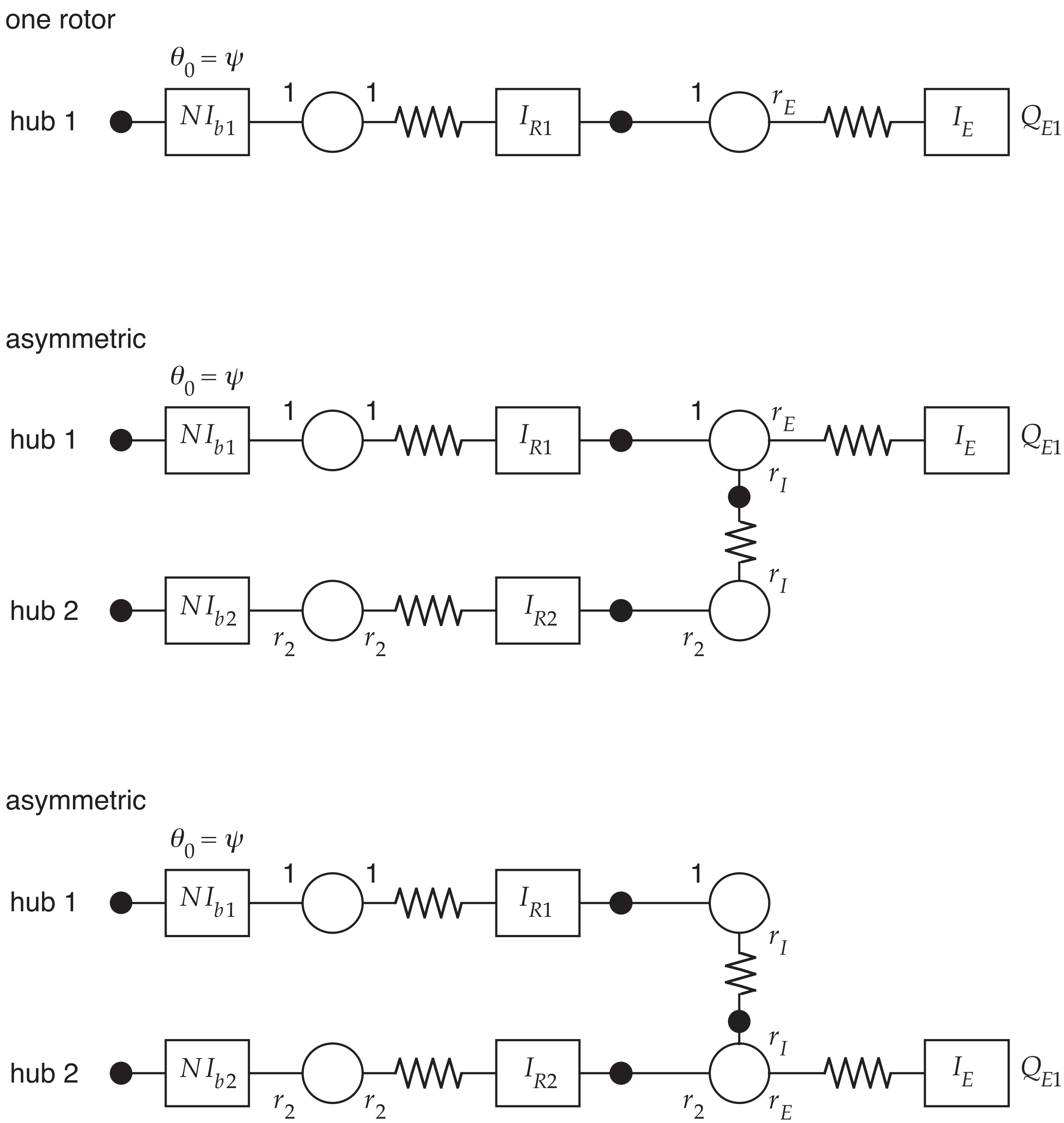


Figure 6-35a Drive train configurations.

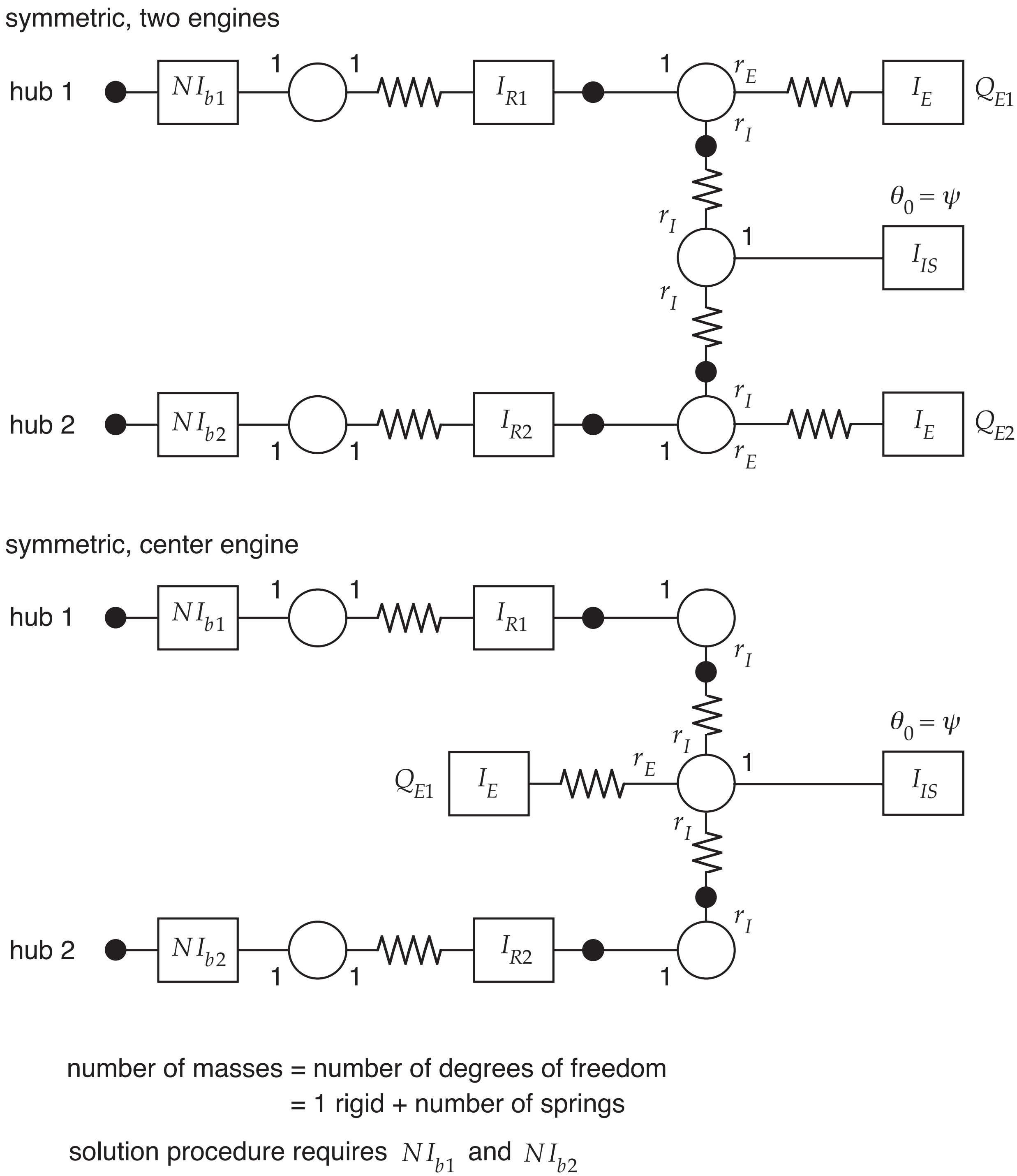


Figure 6-35b Drive train configurations.

CONFIGURATIONS AND ASSOCIATED SYSTEM DEGREES OF FREEDOM

one rotor: ψ, ψ_{R1}, ψ_{E1}

asymmetric, engine by first rotor: ψ, ψ_{R1}, ψ_{R2}, ψ_{E1}, ψ_{I1}

asymmetric, engine by second rotor: ψ, ψ_{R1}, ψ_{R2}, ψ_{E1}, ψ_{I1}

symmetric, two engines: ψ, ψ_{R1}, ψ_{R2}, ψ_{E1}, ψ_{E2}, ψ_{I1}, ψ_{I2}

symmetric, center engine: ψ, ψ_{R1}, ψ_{R2}, ψ_{E1}, ψ_{I1}, ψ_{I2}

ψ = azimuth
ψ_{Rn} = rotor shaft
ψ_{En} = engine shaft
ψ_{In} = interconnect shaft

each transmission component has rigid degree of freedom, all but one of which is eliminated by structural dynamic interfaces

other degrees of freedom are elastic motion of transmission shaft (can be suppressed)

WITH MORE THAN TWO ROTORS, ADD BRANCH FOR n-th ROTOR ($n \geq 3$) AS FOR SECOND ROTOR

RIGID DEGREE OF FREEDOM OF DRIVE TRAIN SYSTEM (INCLUDING THE ROTORS) IS AZIMUTH ψ

ψ = rigid motion of appropriate transmission component (first rotor shaft, except for symmetric case)

response nominal for ψ is $\psi_{\text{nom}} = \Omega_1 t + \psi_{01}$

Ω_1 = rotational speed of first rotor
ψ_{01} = reference azimuth of first rotor

AZIMUTH ψ IS SYSTEM RIGID DEGREE OF FREEDOM

steady state motion is specified rotational speed of rotor

corresponding system rigid force is net torque on drive train

trim loop can solve for throttle (or rotor speed or other variable) that makes this net torque zero

GOVERNOR AND ENGINE CONTROL

DRIVE TRAIN AZIMUTH/SPEED ERROR y_{gov} OBTAINED FROM TRANSMISSION COMPONENT WITH AZIMUTH ψ

component calculates errors $\dot{\psi}_e$ and ψ_e relative reference rotation at prescribed rotor speed Ω

sensor has proportional (rotational speed error) and integral (rotation angle error) terms, with constant gains:

$$y_{\text{gov}} = \theta_{g\ \text{static}} = K_P \dot{\psi}_e + K_I \psi_e$$

GOVERNOR DYNAMICS DESCRIBED BY SECOND-ORDER DIFFERENTIAL EQUATION

in terms of transfer function for governor degree of freedom θ_g

$$\frac{\theta_g}{\theta_{g\ \mathrm{static}}} = \frac{\omega_n^2}{s^2 + 2\omega_n \zeta s + \omega_n^2}$$

GOVERNOR FEEDBACK

control to rotor collective and engine throttle

$$\theta_{\mathrm{gov}R} = K_{R\mathrm{gov}}\theta_g$$
$$\theta_{\mathrm{gov}E} = K_{E\mathrm{gov}}\theta_g$$

feedback to engine throttle is part of drive train subsystem (directly coupled to transmission dynamics)

feedback to rotor collective implemented through filter, so mean value can be included in steady collective pitch for trim

ENGINE CONTROL MODELLED BY APPLIED LOAD (TORQUE) ON TRANSMISSION BRANCH REPRESENTING THE ENGINE

control torque produced by engine control c_E, and governor feedback term

gain G_E converts total control from throttle units to torque

6–7 Rotor Structure

ROTOR STRUCTURE CONSISTS OF ONE OR MORE IDENTICAL, EQUALLY-SPACED BLADES

HUB AND BLADE ROOT CONFIGURATIONS INCLUDE ARTICULATED, HINGELESS, TEETERING, GIMBALLED, AND BEARINGLESS

STRUCTURAL MODEL BASED ON BEAM THEORY FOR COUPLED FLAP-LAG-TORSION MOTION OF BLADE, INCLUDING LARGE PITCH AND LARGE TWIST

assumption that structural elements have high aspect-ratio is normally well satisfied for rotor blades

although beam theory may be less accurate modelling root of hingeless or bearingless blade

structural model: isotropic with elastic axis; or anisotropic/composite

geometric model for elastic motion: second-order, almost-exact, or exact

CONTROL CONFIGURATIONS INCLUDE SWASHPLATE MODEL (WITH PITCH HORNS AND PITCH LINKS), OR BLADE ROOT PITCH

ROTOR WITH UNEQUAL BLADE SPACING CAN BE CONSTRUCTED

ROTOR MODEL CAN BE USED TO CONSTRUCT A NONROTATING WING

ROTOR BLADE AND FLEXBEAM ARE DIVIDED INTO ELEMENTS

modelled by beam components, with joints for required hinges and bearings

beam component has straight beam axis

and only use blade properties at Gaussian integration points

so blade having properties that vary rapidly along its length must be modelled by breaking it into many elements, with major jumps in properties at the nodes

optionally, elements can have only rigid motion, or be modelled by rigid body components

rigid wing component can be used for last element of blade

combines rigid body component and lifting line wing component, for computational efficiency

ANALYSIS CAN USE MODAL TRANSFORMATION AND TRUNCATION IN SOLVING FOR ROTOR MOTION

in trim, transient, or flutter tasks

modes encompass any control system components, but not rotating hub component or swashplate component (which couple structural dynamics of blades)

optionally modes can exclude joints, all elements and point masses inboard of joints, pitch horn and pitch link, and flexbeam

figures 8 to 10 describe rotor hub, blade, and control system configuration constructed

ROTOR INERTIAL PROPERTIES

DEFINITION OF ROTOR STRUCTURE (BLADES AND FLEXBEAMS, PERHAPS WITH POINT MASSES AND FLAPS) INCLUDES INERTIA

- this inertia is all in rotating frame, outboard of rotor blade root component
- used to calculate blade mass M_b and moment of inertia I_b (for system inertia)

INPUT VALUE OF BLADE AND FLEXBEAM MASS IS PER UNIT SPAN STATION (slug/ft or kg/m)

- so integral over span station gives the correct blade mass
- when elements representing blade are constructed, input is converted to mass per length along beam axis

MAJOR CONFIGURATION OPTIONS

ROOT CONFIGURATION IS SINGLE LOAD PATH, OR BEARINGLESS

for articulated and hingeless rotors, blade structure has single load path from root to tip

for bearingless rotors, structure has dual load path, consisting of blade and flexbeam

GIMBAL OR TEETER JOINT MAY EXIST BETWEEN ROTATING HUB AND ROOT COMPONENTS

teeter hinge for $N \leq 2$, gimbal for $N \geq 3$

DRIVE TRAIN MODEL MAY EXIST

including connection between rotating hub and rotating swashplate

SWASHPLATE MECHANISM CAN BE CONSTRUCTED

including pitch link and pitch horn components, perhaps with individual-blade-control

optionally can use nonrotating actuators

without swashplate mechanism, rotor control introduced at blade pitch joint

POINT MASSES MAY BE ATTACHED TO BLADE AND FLEXBEAM

BLADE CAN HAVE ONE OR MORE TRAILING-EDGE FLAPS

figure 36 shows components, connections, and joints used to implement these options

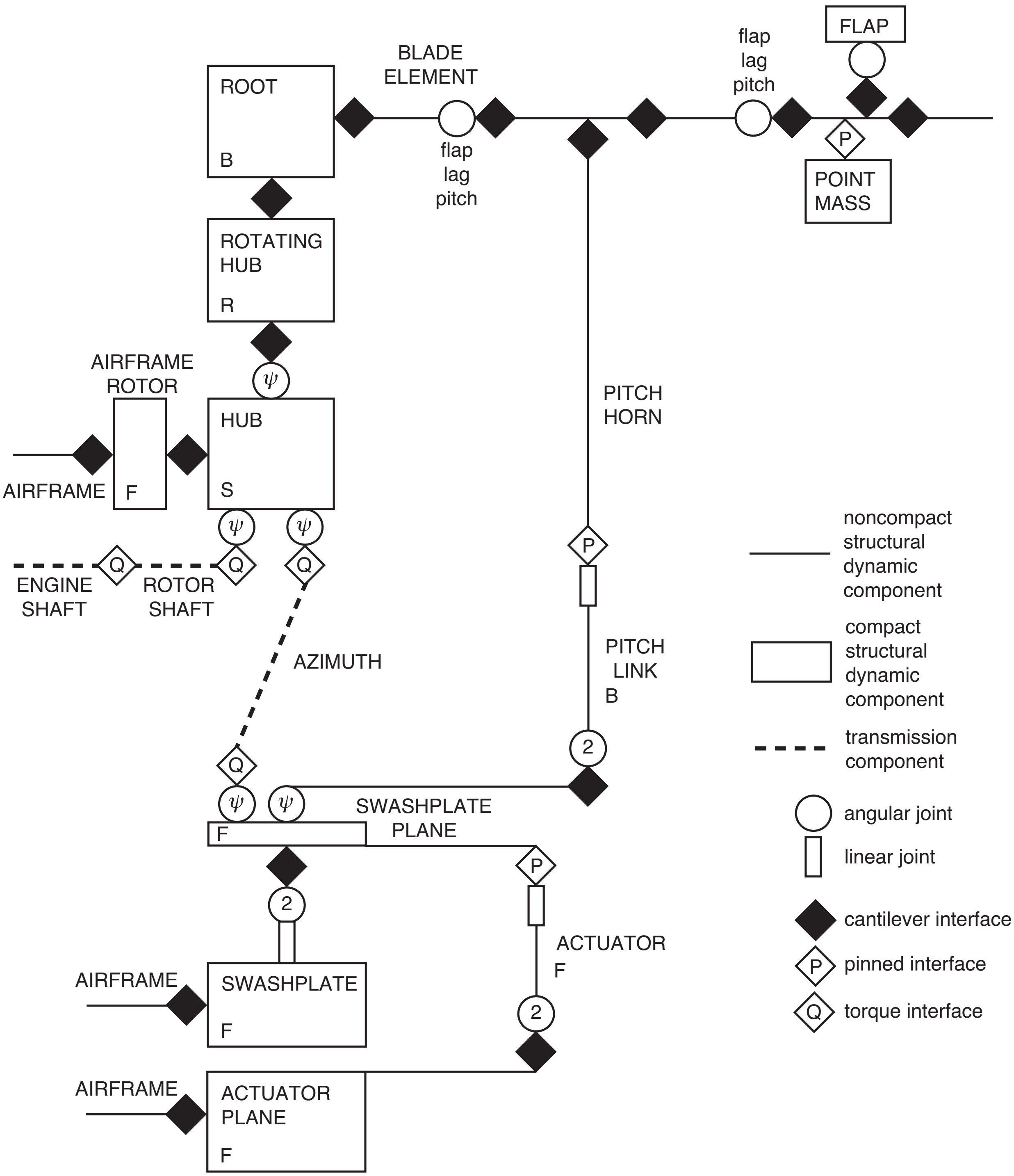


Figure 6-36a Construction of rotor hub, blade, and control system.

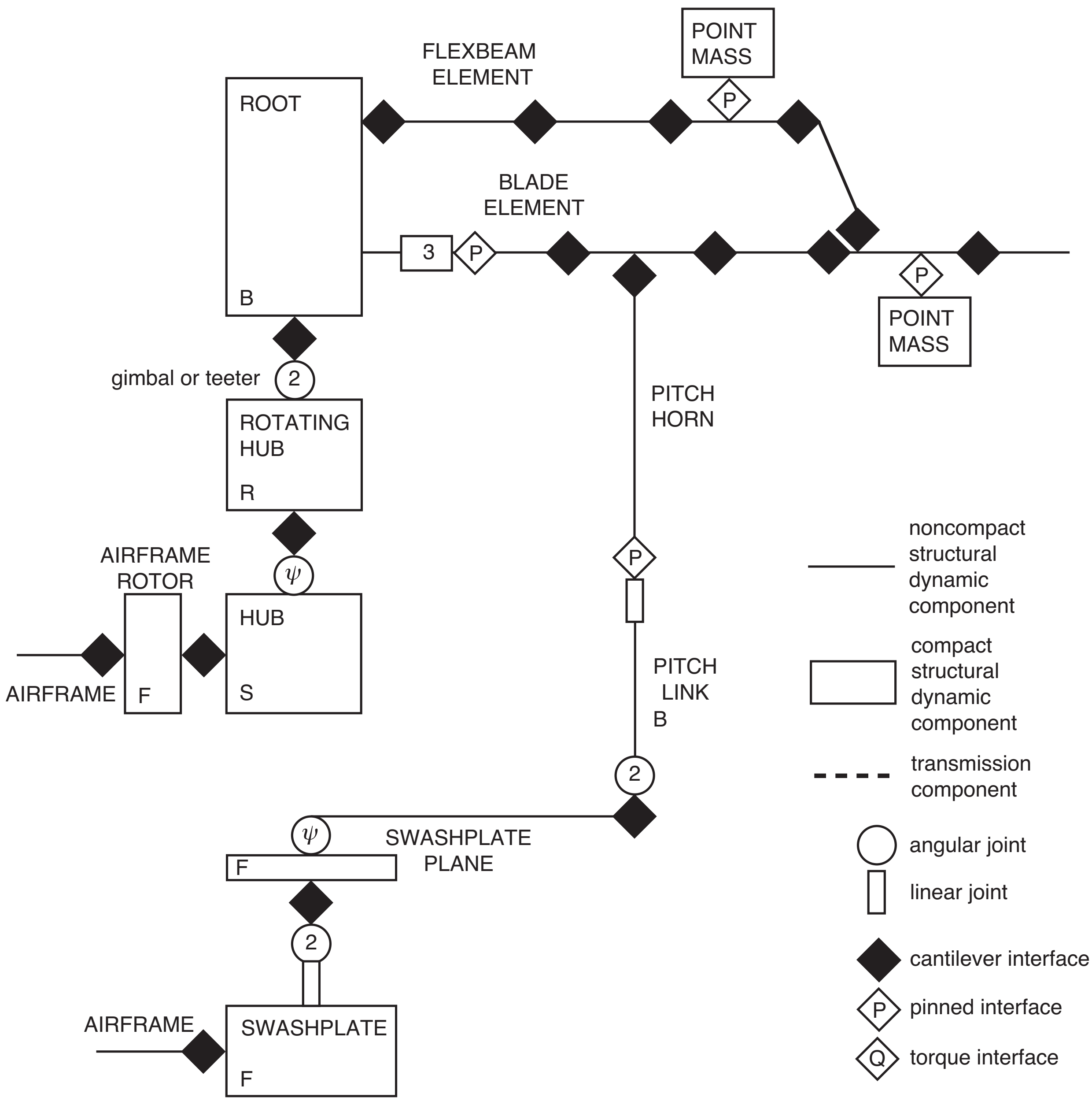


Figure 6-36b Construction of rotor hub, blade, and control system.

HUB AND BLADES

ROTOR HUB COMPONENT CONNECTED TO AIRFRAME

through a rigid body component, in order to handle description of rotor inertia

hub has joint for rotor rotation

rotating hub component attached to that joint

rotor blade root components of all blades attached to connections on rotating hub component

rotating hub component can have gimbal or teeter joint

gimbal joint is Rodrigues hinge, not universal joint

teeter joint can include pitch-flap coupling (δ_3)

can have rotating shaft component between hub and rotating hub components, with rotational spring and inertia

optionally the gimbal or teeter hinge can be simulated by a flap hinge at the center of rotation on each blade

trim solution filtered so only $pN \pm 1$ harmonics nonzero

flutter degrees of freedom just tip-path plane tilt (β_{1c}, β_{1s}) or teeter (β_1)

approximate model, but simpler

can solve for just reference blade,
can even use pitch bearing control instead of swashplate

hub, rotating hub, rotating shaft, and root components are compact:

all locations, joints, and connections are at hub node (except the snubber connection)

so hub node is origin of S, R, and B frames, center of rotation, and location of gimbal or teeter hinges

hub node also point at which hub and blade root loads are measured

FIRST BLADE ELEMENT CONNECTED TO ROTOR BLADE ROOT COMPONENT

blade elements form single load path from root to tip (except for bearingless configuration)

each element can have joint at its outboard end

flap, lag, or pitch hinge, or damper

control can be introduced at pitch bearing

outboard of all joints is droop-sweep node

where sweep and droop of blade reference line introduced

with a swashplate mechanism, pitch horn component connected to blade at specified radial station

point mass component can be connected to any element

FOR BEARINGLESS CONFIGURATION, ELEMENTS FORM TWO LOAD PATHS: BLADE AND FLEXBEAM

flexbeam: no joints

first element connected to rotor blade root component

last element connected to blade at flexbeam node (inboard of droop-sweep node)

point mass component can be connected to any element

blade: constructed as for single load-path configuration

with point masses, but no joints

no pitch bearing, so need swashplate for control

all aerodynamic interfaces on blade, not flexbeam

inboard end of blade can be free, or connected to snubber joint on hub or flexbeam

snubber: connects blade to hub (structure rigid from hub node to snubber attachment) or to flexbeam

for snubber axes fixed to hub or flexbeam:

snubber joint has three linear degrees of freedom, with pinned interface to blade

if snubber axes rotate with blade:

snubber joint has three angular and then three linear degrees of freedom, then cantilever interface to blade

CONTROL SYSTEM

SWASHPLATE MECHANISM

- if swashplate mechanism exists, rotor swashplate component connected to airframe, at swashplate node on shaft axis, below the hub
- swashplate component has joint for swashplate control
  - one linear motion for collective, then two angular motions for cyclic
    - cyclic joint is Rodrigues gimbal (not universal joint)
    - joint axes defined so longitudinal and lateral cyclic control properly phased
    - swashplate gains calculated from nominal geometry, so control can still be interpreted as blade pitch angle
  - joint motion = degrees of freedom with spring and offset actuator
  - rotor swashplate plane component attached to connection at that joint
- optionally swashplate motion can be produced by 3 nonrotating actuators
  - each actuator connected to swashplate plane component and to actuator plane component
  - each actuator has linear joint with spring and offset actuator

swashplate plane component has joint for rotation of rotating swashplate

- rotor blade pitch link components of all blades attached to connections at that joint

- these connections are at appropriate location in swashplate plane, not at center of rotation

except for collective joint motion and pitch link connections, swashplate and swashplate plane components are compact

- with all locations, joints, and connections at swashplate node

rotor blade pitch horn component connected to blade, at specified radial station (inboard of node or joint)

rotor blade pitch link component connects pitch horn and swashplate plane

- at connection to pitch horn, pitch link component has linear joint, with spring for control system flexibility

- optionally with actuator for individual-blade-control

- at connection to swashplate plane, pitch link component has gimbal joint (and cantilever interface)

    - pinned interface (and no joint) here would leave pitch link axial rotation unconstrained; a cantilever interface is also needed so that the interface force can be kept in rotating frame

CONTROL SYSTEM MODELS

can construct:

swashplate mechanism with nonrotating actuators
swashplate mechanism with swashplate control
swashplate mechanism
no swashplate mechanism

swashplate mechanism introduces additional load paths and additional rotating-to-nonrotating frame interfaces for system

not restricted to small motions of linkages

with swashplate control, there must be pitch flexibility inboard of pitch horn

pitch bearing for single load-path configuration

flexbeam torsion flexibility for bearingless configuration

swashplate collective and cyclic joints have degrees of freedom, with offset actuators to introduce control

swashplate control =
rotor primary control (collective and cyclic)
+ nonrotating frame higher-harmonic control

control system flexibility represented by springs in pitch link and swashplate joints

locked swashplate obtained using quasistatic solution, with a large spring

SINGLE LOAD-PATH CONFIGURATION CAN INCLUDE PITCH BEARING

can construct:

pitch bearing with pitch control
pitch bearing
no pitch bearing

pitch bearing with control not available if swashplate mechanism used

pitch joint has degree of freedom, with offset actuator to introduce control

control =
rotor primary control
+ blade pitch control
+ rotating frame higher-harmonic control

with a swashplate and individual-blade-control actuators, this control goes to pitch link joint, not to root pitch joint

control system flexibility can be represented by spring in pitch bearing (in absence of swashplate mechanism, this is only source of flexibility)

locked pitch bearing obtained using quasistatic solution, with a large spring

DRIVE TRAIN

WITHOUT DRIVE TRAIN MODEL

rotation of rotor and swashplate prescribed, at constant angular velocity

WITH DRIVE TRAIN MODEL

this rotation is a degree of freedom, with nominal motion equal to the prescribed motion

COMPONENTS

hub component has two additional rotational joints, with same joint variable as rotor rotation

swashplate plane component has one additional rotational joint, with same joint variable as rotating swashplate

transmission component (rotor azimuth) connects rotating swashplate joint to hub rotation joint

so control system has same azimuth as rotor

drive train rotor shaft component connected to remaining joint on hub

ROTOR CAN HAVE FREE-ROTATION HUB

hub joint is degree of freedom; connected to swashplate (if it exists) but not connected to drive train

TRANSMISSION COMPONENTS USED TO MODEL DRIVE TRAIN

so actual load paths between rotor, airframe, and drive train are not modelled

and rotor torque always reacted at hub node of airframe (and perhaps at swashplate node), with or without a drive train model

GENERAL CHARACTERISTICS

ROTOR COMPONENTS

finite element beam components: elements of blade and flexbeam, and pitch horns

elastic motion can be suppressed; or can be replaced by rigid body component

pitch horn constructed without elastic degrees of freedom

rigid body components: all other

last element of blade can be replaced by rigid wing component

INERTIA

only elements of blade and flexbeam, point masses, and flaps have inertia

all other components of rotor model are massless

GENERALLY BLADES HAVE INDEPENDENT STRUCTURAL DYNAMICS

coupled through airframe, drive train, and aerodynamics

so may be possible to solve for motion of just reference blade

gimbal or teeter hinge, or flexibility in swashplate joints, will couple blade structural dynamics

SENSORS

NONROTATING FRAME HUB FORCE AND MOMENT
MEASURED ON ROTOR HUB COMPONENT

sum of loads applied at location of rotor rotation joint

loads at this location include inertial reactions and weight of rotors

SHAFT TORQUE AND POWER, ROTATIONAL SPEED
MEASURED ON ROTOR HUB COMPONENT

torque is shaft-axis component of hub moment

power sensor is power of rotor rotation joint

rotational speed is angular velocity of rotor rotation joint

SWASHPLATE FORCE AND MOMENT
MEASURED ON SWASHPLATE COMPONENT

ROTATING FRAME BLADE ROOT FORCE AND MOMENT
MEASURED ON ROTOR BLADE ROOT COMPONENT

sum of loads acting on hub

PITCH LINK LOAD
MEASURED ON ROTOR BLADE PITCH LINK COMPONENT

axial force, at location at top of pitch link

BLADE SECTION FORCE AND MOMENT EVALUATED AT NODE, OR ON FINITE ELEMENT BEAM COMPONENT

node reaction measured as reaction on connection for structural dynamic interface at the node

in structural principal axes relative the tension center

on either side of node (since there might be jump in tension center or structural axes at node)

section load evaluated from deflection, or by force balance

BLADE POSITION SENSORS MEASURE MOTION OF BLADE

flap, lag, pitch, and axial motion at span stations on blade and flexbeam

displacement measured relative
blade frame, rotating hub frame,
nonrotating hub frame,
airframe frame, or inertial frame

pitch rotation always measured relative
rotating blade frame

and motion of all joints

flap, lag, pitch, damper joints
pitch link joint
snubber joint
gimbal or teeter joint
swashplate joint

CFD POSITION SENSORS MEASURE DISPLACEMENT AND EULER ANGLES OF BLADE

typically quarter chord position relative blade frame

TIP-PATH PLANE SENSOR

AERODYNAMIC AND PERFORMANCE CALCULATIONS, AND TRIM OPTIONS, REQUIRE ROTOR TIP-PATH PLANE ORIENTATION

calculated by reference plane component from sensor z_{TPPnm}

SENSOR CAN BE THE MOTION OF THE BLADE TIP (OR SOME OTHER SPAN STATION)

structural dynamic sensor that measures z-axis displacement of blade tip relative rotating hub frame (not blade frame, since blade frame is outboard of gimbal or teeter hinge)

displacement divided by rotor radius R, so sensor gives tilt of line from hub to blade tip, relative the hub plane

in general, can put sensor at span station e_{TPP}, on blade beam axis; scaled to tip value:

$$z_{TPPnm} = \frac{z}{e_{TPP}R}$$

need inboard span station for blade with swept tip, to avoid pitch motion contributing to vertical displacement

for gimballed or teetering rotor, sensor span station near hub will give just gimbal or teeter hinge motion

OR SENSOR CAN BE ROTATION OF FLAP HINGE (IF JOINT WITH FLAP MOTION EXISTS)

OR ROTATION OF GIMBAL OR TEETER HINGE

BLADE GEOMETRY

FIGURE 37 SHOWS GEOMETRY OF BLADE AND CONTROL SYSTEM

specified in blade axes (B coordinates)

radial y-axis = span station variable

INPUT CONVENTIONS

shell input data dimensionless, scaled with rotor radius R

directions of B axes relative blade are different for counter-clockwise and clockwise rotation

- but shell input data same for both directions of rotation
- input X-quantities always are positive towards trailing edge

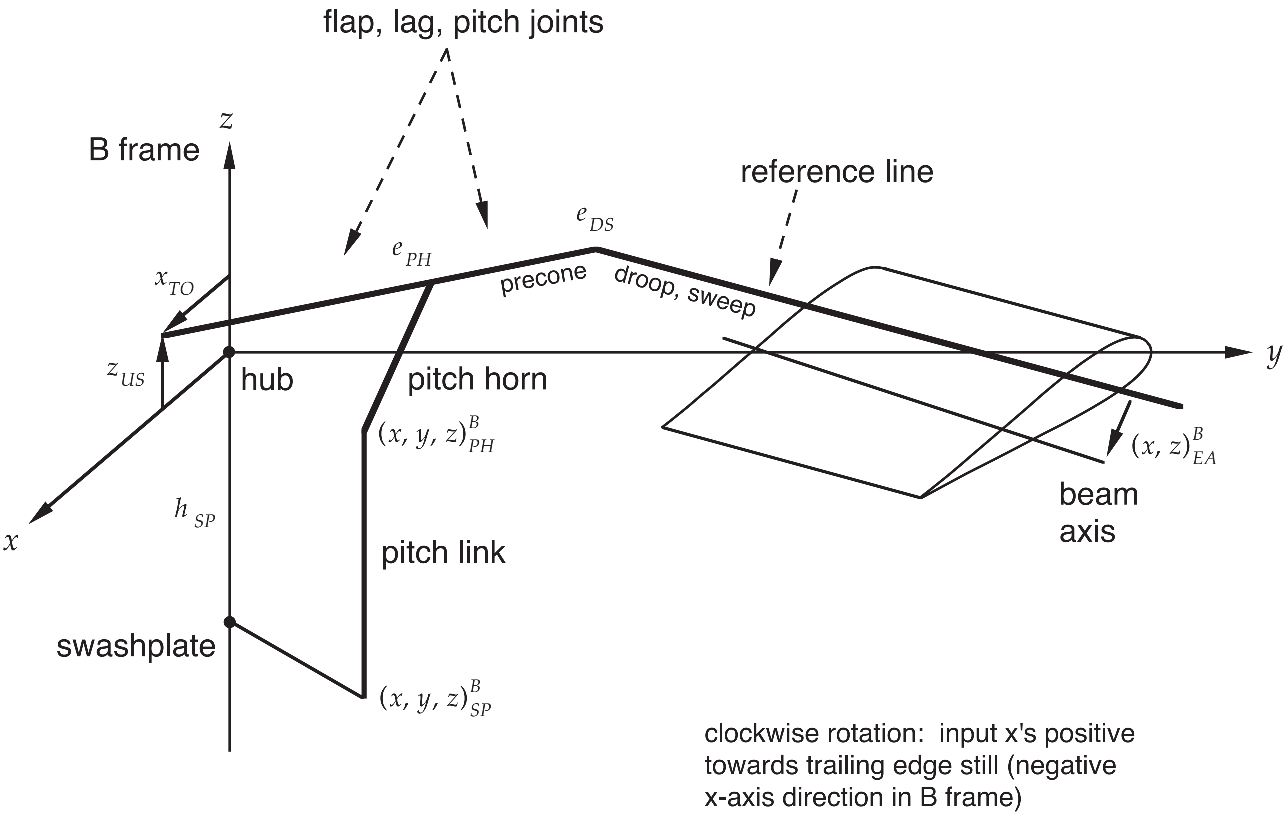
blade properties are input as piecewise linear functions of span station

BLADE TIP AND BLADE RADIUS

blade tip radial station is e_{tip} (dimensionless)

usually R is actual rotor disk radius, and $e_{\text{tip}} = 1$

if $e_{\text{tip}} \neq 1$, then R is not the physical radius,
rather R is only used for normalization of the input parameters and the output



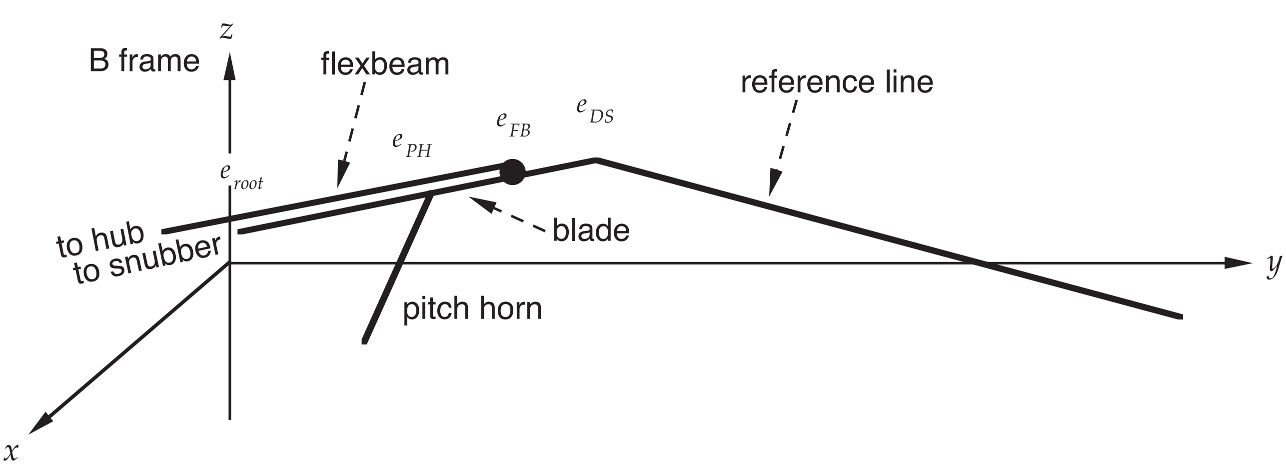


Figure 6-37 Geometry of blade and control system.

REFERENCE LINE

define reference line, extending from root to tip:

- torque offset x_{TO} and gimbal/teeter undersling z_{US} at root (probably negative)
- precone angle β_p at root (relative y-axis, positive up)
- droop angle β_d and sweep angle ζ_s (relative preconed line, positive down and aft) at droop-sweep node at span station e_{DS}
  - so reference line consists of two straight segments, divided at droop-sweep node (which may be absent)

all joints are on first segment of reference line, at span stations inboard of droop-sweep node

- so pitch bearing is aligned with preconed reference line

this reference line is used to describe blade geometry

- wing reference line is y-axis of rotating blade frame (B axes)

GEOMETRY RELATIVE REFERENCE LINE

blade geometry specified in B coordinates, measured from the reference line:

locus of beam axis
locus of quarter chord
damper attachments
locations of point masses

aerodynamic twist measured about wing reference line (y-axis), from x–y plane of B frame

BEAM AXIS

beam axis is elastic axis for isotropic structural model

beam axis must be straight within each element (between nodes)

such a beam axis constructed from input data, as straight line between specified beam axis positions at 10% and 90% element length (so jumps at the nodes are possible)

all structural dynamic interfaces between elements of blade and flexbeam are at mean of beam axis positions of ends of elements being connected

axes of interface are B frame axes

BLADE PROPERTIES

blade properties are specified relative beam axis

positions of center of gravity and tension center: measured from straight beam axis of element, in section principal axes

pitch angle of principle axes: measured about beam axis, from element x–y plane

section mass specified per unit length along span axis, not along element axis (so integration over span station gives correct total blade mass)

CONTROL SYSTEM

geometry of control system:

position of swashplate node, on shaft axis h_{SP} below hub (positive)

position of connection between swashplate and pitch link, in B axes (with $z \cong -h_{SP}$)

position of connection between pitch link and pitch horn, in B axes (with x positive for trailing pitch horn)

span station e_{PH} of pitch horn connection to blade

geometry defined for zero collective and cyclic control and reference blade pitch (typically corresponding to zero pitch at 75% radius, depending on definition of aerodynamic twist)

BEARINGLESS CONFIGURATION

inboard end of blade is span station e_{blade}

flexbeam connected to hub at span station $e = 0$

snubber joint can connect blade root component and inboard end of blade

location of snubber on blade root component specified in B axes (as for pitch link ends)

connection of snubber to blade specified in B coordinates, measured from the reference line (as for beam axis), at radial station e_{root}

or snubber joint can connect flexbeam element and inboard end of blade

location of snubber on flexbeam element specified in B coordinates, measured from the reference line (as for beam axis), at radial station y_{snub}

connection of snubber to blade specified in B coordinates, measured from the reference line (as for beam axis), at radial station e_{root}

snubber joint has three linear degrees of freedom, each with spring and damper

snubber joint axes rotated by precone angle β_p, and then by snubber pitch angle θ_s

ELEMENTS, NODES, AND JOINTS

BLADE STRUCTURE DIVIDED INTO SET OF ELEMENTS (COMPONENTS) BY NODES (STRUCTURAL DYNAMIC INTERFACES) POSITIONED ON REFERENCE LINE

nodes identified by span station e

NODES

automatic nodes

connection of first element to rotor blade root component is first node, at $e = 0$

blade tip is last node, at $e = e_{\text{tip}}$

droop-sweep node required at e_{DS} if reference line has nonzero droop or sweep angle

node at each hinge or bearing

additional nodes can be specified as required

to account for large changes in beam axis definition, since beam axis is straight within each element

to increase number of elastic degrees of freedom

figure 38 illustrates connection between specification of nodes and construction of beam axis

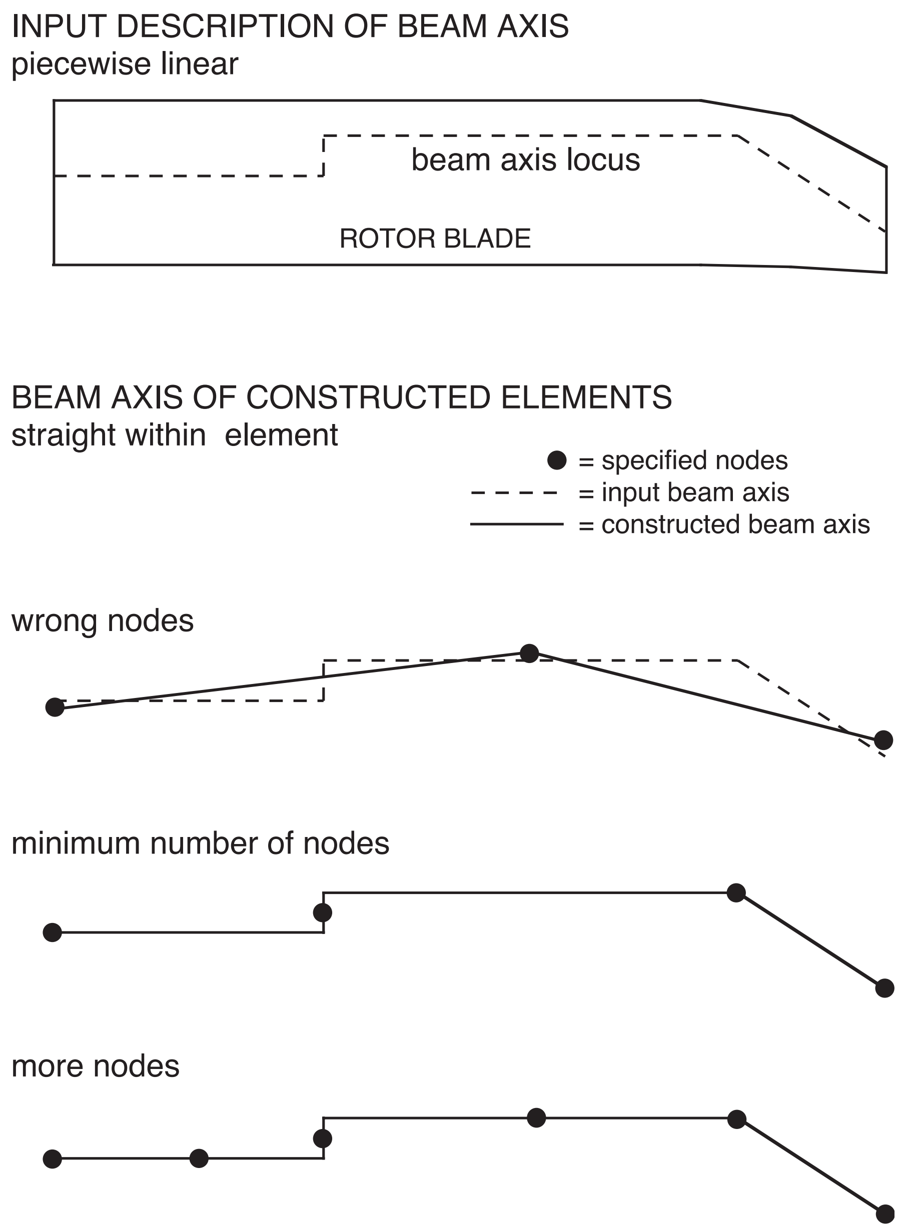


Figure 6-38 Blade beam axis constructed by rotorcraft shell.

ELEMENTS

blade element constructed between each pair of nodes

options for component type:

- finite element beam component

  - without elastic motion if length is less than a specified minimum, or if element entirely within a specified range

- finite element beam component, with no elastic motion

- rigid body component (with inertial properties evaluated as by beam component)

analysis may create zero-length elements to accommodate specified joints

- element with zero length is massless and rigid

JOINTS

outboard end of an element can have a joint; all joints occur before droop-sweep node

possible joints: flap and lag hinges, pitch bearing, damper

FLAP AND LAG HINGES, AT e_F AND e_L

options:
- none
- flap hinge only
- lag hinge only
- both hinges
- flap then lag universal joint
- lag then flap universal joint
- flap-lag gimbal (Rodrigues hinges)

each hinge can have spring and damper

hinges placed on reference line, with joint axes parallel B axes

- flap hinge can have pitch-flap coupling (δ_3)
- lag hinge can have pitch-lag coupling (α_1)
  - hinge skew frequently used on tail rotor
  - but for main rotor, coupling with pitch more often produced by kinematics of control system
- flap and lag hinges can be rotated by fixed pitch angles relative the B axes

optionally joints can be locked: joint exists, but motion is prescribed (to zero) rather than degree of freedom

PITCH BEARING AT e_P

options:
- none
- pitch bearing
- pitch bearing with control

joint can have spring and damper

with swashplate mechanism, pitch bearing typically has zero spring and damping

without swashplate mechanism, spring at pitch bearing represents control system flexibility

PITCH BEARING PLACED ON REFERENCE LINE, ALIGNED WITH PRECONED DIRECTION

for controlled pitch, joint has offset actuator

locked pitch bearing obtained by using quasistatic solution for joint motion, with large spring

BEARINGLESS CONFIGURATION

two load paths constructed: blade and flexbeam

blade extends from e_{blade} to tip

blade may be attached to root component or flexbeam through snubber joint

flexbeam extends from hub ($e = 0$) to connection with blade at e_{FB}

flexbeam attached to rotor blade root component by cantilever structural dynamic interface

$e_{FB} \leq e_{DS}$, so reference line for flexbeam has only one segment, at precone angle

6–8 Rotor Aerodynamics

ROTOR CONSISTS OF ONE OR MORE IDENTICAL, EQUALLY-SPACED WINGS (BLADES)

- BLADE CONFIGURATION INCLUDES SWEPT AND DROOPED TIPS (ARBITRARY QUARTER CHORD LOCUS)
- ROTOR WITH UNEQUAL BLADE SPACING CAN BE CONSTRUCTED

- WING MODEL IS BASED ON LIFTING-LINE THEORY
  - using steady two-dimensional airfoil characteristics and vortex wake; plus corrections for unsteady and yawed-flow effects
  - wing component includes: second order lifting line theory; empirical dynamic stall models; unsteady loads from thin-airfoil theory; yawed and swept flow corrections; spanwise drag
  - blade can have one or more trailing-edge flaps

INDUCED VELOCITY OBTAINED FROM:

UNIFORM INFLOW MODEL, BASED ON MOMENTUM THEORY

mean inflow plus linear variation over rotor disk (from edgewise flow and hub moments)

dynamic inflow can be used in transient and flutter tasks (low frequency, global model of wake influence in unsteady aerodynamics of rotor)

or dynamic wake model (unsteady wake theory based on acceleration potential for actuator disk)

OR NONUNIFORM MODEL, USING VORTEX WAKE REPRESENTATION

wake geometry can be rigid, prescribed (for hover), or free

UNIFORM INFLOW MODEL CAN BE FOR DUCTED FAN

duct aerodynamic loads calculated from rotor loads, and specified ratio of rotor load to total load

figures 11 to 18, 20, 23, and 24 describe configuration constructed

INPUT CONVENTIONS

chord input is dimensional (ft or m)

wing properties are input as piecewise linear functions of span station r

WING GEOMETRY

WING REFERENCE LINE IS y-AXIS OF ROTATING BLADE FRAME (B AXES)

straight line, with origin at center of rotation

span station variable is dimensionless radial distance $r = y/R$

span scale factor R = blade radius

with large droop, will need wing reference line that includes the droop

for moderate droop, just use core input to define span scale factor R as a function of r, such that $(R\,\Delta r\,c)$ is wing panel area

WING PLANFORM DEFINED BY QUARTER-CHORD LINE, AERODYNAMIC TWIST, AND CHORD

rotor structure input gives quarter-chord line and twist

wing sections are defined in plane perpendicular to wing reference line

input blade chord measured in this section, so blade area properly calculated

pitch measured in this section, from x-axis of B frame

quarter chord locus defined in this section, but measured from reference line of blade structural definition

6–9 Trim Task

TRIM TASK OBTAINS EQUILIBRIUM SOLUTION OF SYSTEM EQUATIONS, FOR STEADY STATE OPERATING CONDITION

ASSUME:

system environment and input are constant or periodic,
equilibrium solution exists,
solution is constant or periodic

assumption that system is periodic requires exclusion of any vibratory dynamic and aerodynamic interaction between rotors operating different rotational speeds (such as for main rotor and tail rotor configuration)

rotors and airframe can be also analyzed using a common period, with full interaction

USUALLY IDENTIFY PARAMETERS REQUIRED TO ACHIEVE A SPECIFIED OPERATING CONDITION (AN INVERSE PROBLEM)

TRIM TASK OBTAINS SOLUTION FOR PERIODIC MOTION AND FORCES, AND STEADY TRIM VARIABLES

then performance, structural loads, vibration, and other output quantities can be evaluated

FREE FLIGHT AND WIND TUNNEL OPERATING CONDITIONS CONSIDERED

FREE FLIGHT

rotorcraft trimmed to force and moment equilibrium for specified flight condition

can also trim to specified rotor power

steady state implies unaccelerated flight path (or turn)

turns and climb or descent, as well as level flight

IN WIND TUNNEL

rotor trimmed to specified values of designated parameters, such as rotor thrust and flapping

CAN ALSO OMIT TRIM ITERATION

solve system equations for fixed controls

for free flight case, rotorcraft will probably not be in equilibrium

A REGULATOR CAN BE INTRODUCED, USING ANY CONTROLS NOT REQUIRED TO ACHIEVE EQUILIBRIUM SOLUTION

TRIM LOOPS

PARTITIONED, ITERATIVE PROCEDURE CONSTRUCTED

system is divided into parts, which solve subset of equations for periodic or constant response

loops iterate between part solutions, until converged system solution is obtained

figure 39 describes trim loops and principal interface variables, and shows levels of wake loop

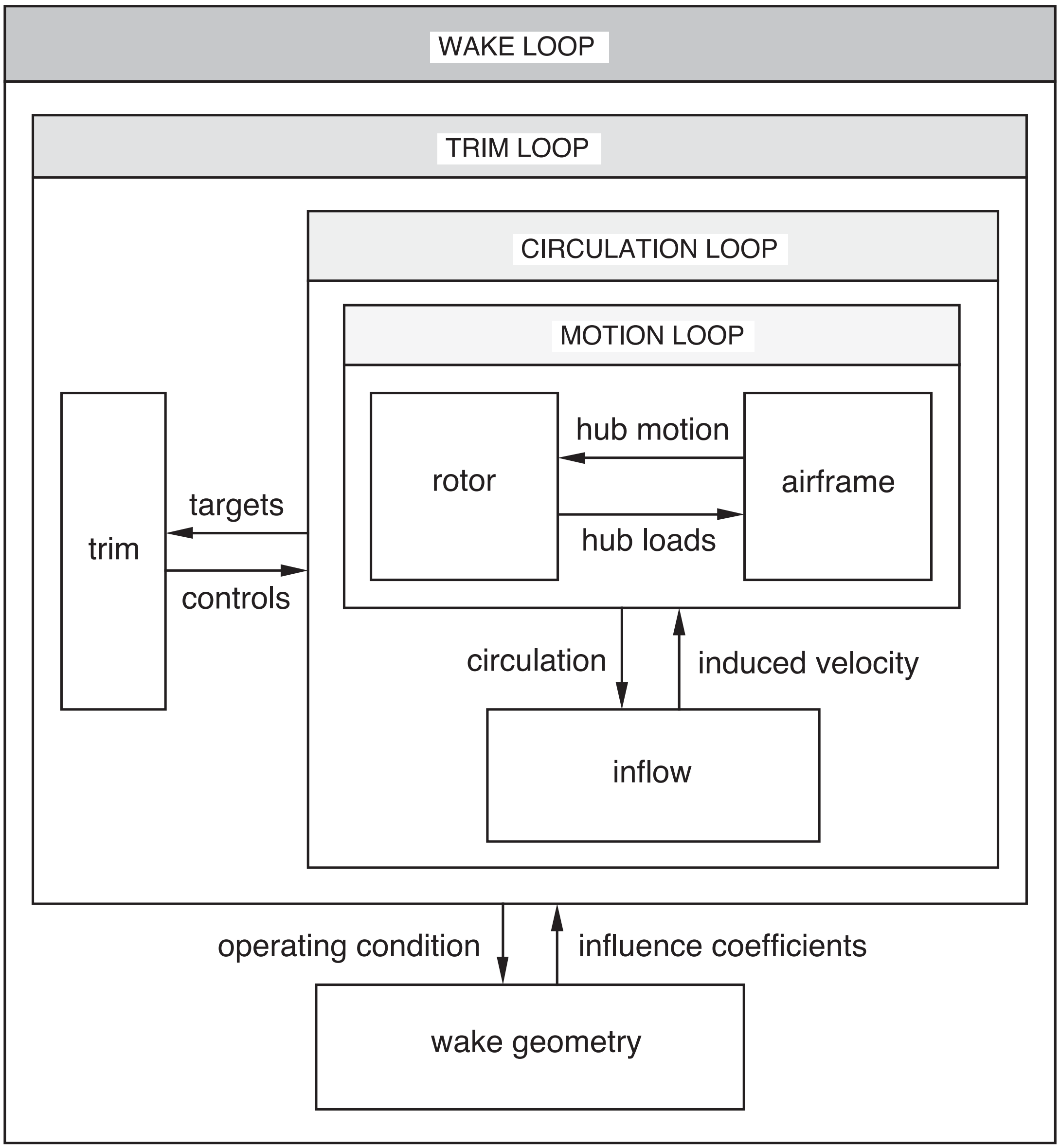


Figure 6-39a Basic loops of trim analysis (optional regulator loop not shown).

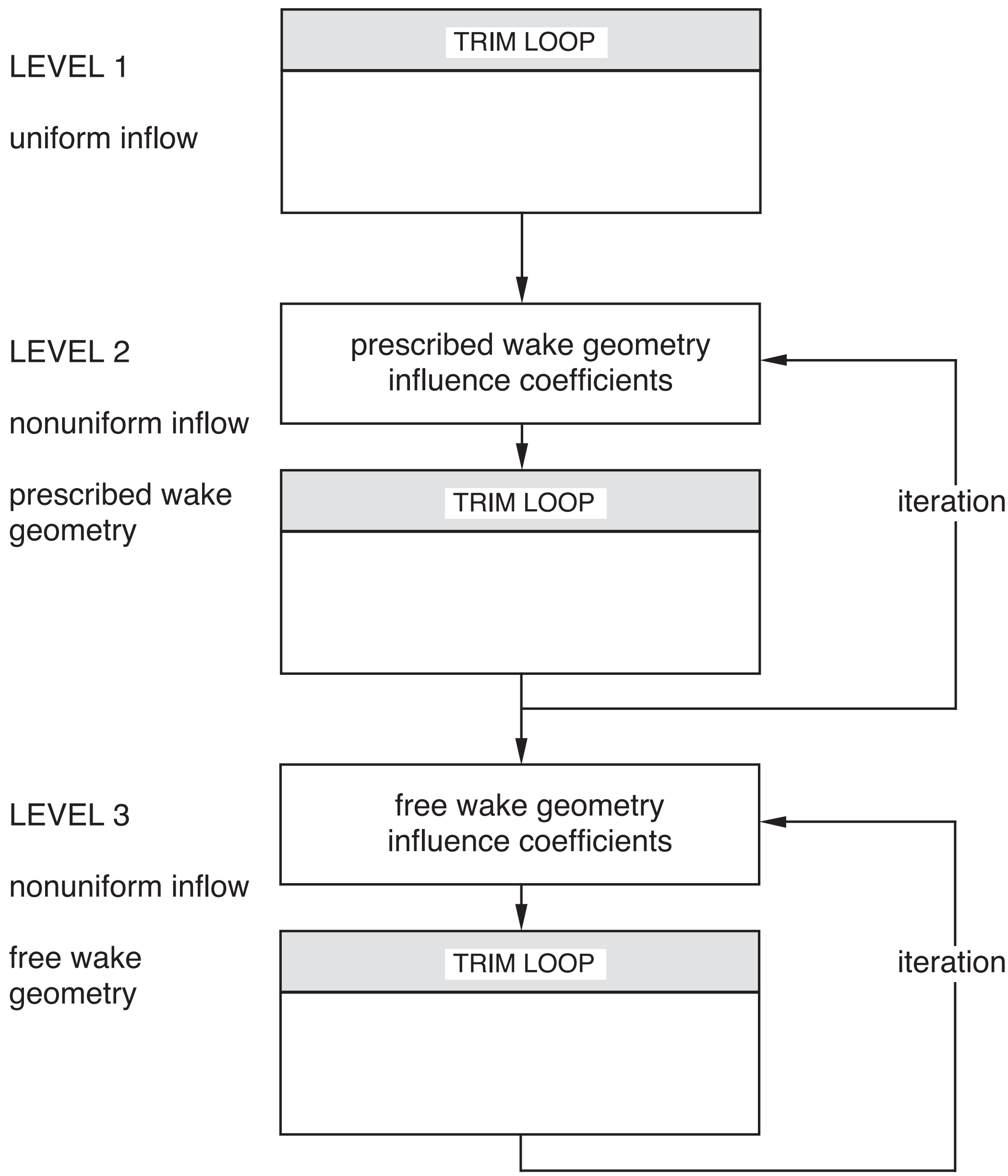


Figure 6-39b Levels of wake loop.

POST-TRIM AERODYNAMIC CALCULATIONS

FOR HIGH AZIMUTHAL RESOLUTION, OR TO OBTAIN PARTIAL ANGLE-OF-ATTACK FOR EXTERNAL AEROACOUSTIC ANALYSIS

POST-TRIM CALCULATIONS IMPLEMENTED IN SEPARATE LOOPS OF TRIM TASK

- fixed wake geometry and structural motion
- executed after converged solution for structural dynamic response obtained

DEFINE DUPLICATE WING, WAKE, AND WAKE GEOMETRY COMPONENTS

- structural dynamic components not duplicated, but have additional aerodynamic interfaces

CIRCULATION CAN BE CALCULATED, TO BE CONSISTENT WITH WAKE ANALYSIS

OR CIRCULATION STRENGTH CAN BE FIXED DURING POST-TRIM SOLUTION

- circulation obtained from coupled aerodynamic and structural solution
- no iteration required in circulation loop
- must be used when partial angle-of-attack evaluated, otherwise calculated circulation reflects the computational domain

WAKE LOOP

ROTOR WAKE-INDUCED VELOCITY CAN BE OBTAINED FROM UNIFORM INFLOW OR NONUNIFORM INFLOW

nonuniform inflow refers to calculation of wake-induced velocity using vortex model

or approximate wake model based on momentum theory can be used (called “uniform inflow”, although induced velocity can have linear variation over rotor disk)

NONUNIFORM INFLOW REQUIRES WAKE GEOMETRY MODEL, WHICH CAN BE RIGID OR PRESCRIBED, OR FREE

rigid model calculates wake distortion from mean induced and interference velocity at rotor

prescribed geometry is obtained from empirical model, based on measurements

free wake geometry obtained by calculation

SUCCESSIVE SUBSTITUTION ITERATION, WITH UP TO THREE LEVELS:

level 1 is uniform inflow

level 2 is nonuniform inflow with rigid or prescribed wake geometry

level 3 is nonuniform inflow with free wake geometry

for better convergence, uniform inflow used to initialize nonuniform inflow, and rigid wake geometry used to initialize free wake geometry

whether results of level 3 or level 2 are required depends on importance of free wake geometry and nonuniform inflow to the problem

WAKE LOOP IS ITERATION ON WAKE GEOMETRY

- wake geometry and influence coefficients calculated for given operating condition and loading
  - then trimmed rotorcraft solution obtained, with induced velocity evaluated from latest bound circulation and fixed influence coefficients
- iteration is required if trimmed rotorcraft solution significantly changes any quantities that influence the wake geometry
- with trim solution inside wake geometry loop, may not be necessary to iterate on wake geometry at each level
- some problems need iteration in wake geometry loop (and relaxation factor on wake geometry)
  - if rotor is at fixed collective pitch (rather than trimmed to specified thrust), then overall geometry of wake is not known in advance, only after thrust calculated
  - or in hover and at low speeds, where rotor loading is very sensitive to wake geometry

WAKE LOOP ITERATION EXECUTED SPECIFIED NUMBER OF TIMES, WITH NO TEST FOR CONVERGENCE

- outer loop iteration considered too expensive to implement numerical test for convergence
- experience shows that often only one or two iterations are required at last stage of loop

TRIM LOOP

FREE FLIGHT

STEADY STATE OPERATING CONDITION IMPLIES UNACCELERATED MOTION

so net force and moment on entire system must be zero

requirement that six components of mean force and moment be zero provides algebraic equations that must be satisfied in trim state

specified rotor power can also be included

UNACCELERATED MOTION ALSO IMPLIES

that system linear velocity is constant

and angular velocity is zero (except for yaw rate in steady turn)

SO OPERATING CONDITION DESCRIBED BY FOLLOWING TRIM VARIABLES:

magnitude and orientation of system velocity (flight path) and wind speed

orientation of system relative inertial frame

control settings

turn rate
rotor speed

some of these trim variables define operating condition, some can have arbitrary values

solution procedure must determine values of remaining variables

IN WIND TUNNEL

ROTOR TRIMMED TO SPECIFIED VALUES OF DESIGNATED PARAMETERS

requirement that these parameters equal target values provides algebraic equations that must be satisfied in trim state

these trim quantities can be rotor forces, flapping, or power

AVAILABLE TRIM VARIABLES

magnitude and orientation of wind speed

orientation of system relative inertial frame

control settings

rotor speed

TRIM LOOP IS ITERATION THAT ADJUSTS TRIM VARIABLES UNTIL TRIM CRITERIA ARE SATISFIED

values for trim variables are calculated by loop

then differential and integral equations of motion are solved for periodic response of rotor and airframe, for fixed values of trim variables

finally quantities required by trim criteria are evaluated

TRIM LOOP IS NEWTON-RAPHSON ITERATION

CIRCULATION LOOP

CIRCULATION LOOP IS ITERATION ON WAKE-INDUCED VELOCITY AND ROTOR BOUND CIRCULATION

induced velocity calculated from rotor loading, using uniform or nonuniform inflow model

then differential equations of motion are solved for periodic response of rotor and airframe, for fixed induced velocity

iteration is required, until induced velocity and loading are consistent

CIRCULATION LOOP IS SUCCESSIVE SUBSTITUTION ITERATION

relaxation factor of $\lambda = 0.10$ or 0.05 is often required with nonuniform inflow

MOTION LOOP

MOTION LOOP IS ITERATION ON ROTOR AND AIRFRAME MOTION

differential equations of motion are solved for periodic response of rotor, including effects of hub motion generated by airframe

then differential equations of motion are solved for periodic response of airframe, including effects of rotor hub forces

iteration is required, until hub motion and hub forces are consistent

if airframe does not produce hub motion, no iteration is required

MOTION LOOP IS SUCCESSIVE SUBSTITUTION ITERATION

ROTORCRAFT HAS ROTATING AND NONROTATING SUBSYSTEMS

equations of motion for rotor are in rotating frame

equations of motion for airframe and drive train in nonrotating frame

so must have separate part solutions for rotor and airframe

FOR MAIN ROTOR AND TAIL ROTOR CONFIGURATION

must analyze airframe as responding separately to different frequencies of two rotors

REGULATOR LOOP

A SELF-TUNING REGULATOR CAN BE USED TO AUTOMATICALLY ADJUST SELECTED CONTROLS TO MINIMIZE A COST FUNCTION

- COST FUNCTION = WEIGHTED SUM OF SQUARES OF SELECTED MEASUREMENTS

A SELF-TUNING REGULATOR COMBINES RECURSIVE PARAMETER ESTIMATION WITH LINEAR FEEDBACK

- control system characterized by
  - linear, quasistatic, frequency-domain model of rotorcraft response to control
  - identification of rotorcraft model by least-squared-error or Kalman-filter method
  - minimum variance or quadratic performance-function controller
- in trim task, regulator functions as an optimizer
  - does not necessarily correspond to an actual control system

TYPICAL APPLICATIONS FOR ROTORCRAFT:

- multicyclic control to minimize vibration or loads, or improve performance
- cyclic pitch feedback to control rotor flapping (for cyclic controls not used by trim loop)

ROTORCRAFT SHELL CAN CONSTRUCT A REGULATOR LOOP, BETWEEN TRIM AND CIRCULATION LOOPS

measurements can include

hub, root, blade, or control loads
airframe vibration and other sensors
rotor flapping, rotor power

sensors created specifically for regulator

any controls not used by trim loop can be used by regulator

rotor and airframe primary controls

higher harmonic rotor pitch in the rotating or nonrotating frame

mean or higher harmonic auxiliary forces

should print standard output for measurements, in order to get descriptions of quantities

CORE INPUT IS REQUIRED TO COMPLETE LOOP DEFINITION

identify controls and measurements, and set loop parameters

in order to use means, harmonics, or other derived quantities as measurements, appropriate filter components must be defined and solved

AS CREATED BY SHELL, REGULATOR LOOP DOES NOTHING

number of controls and number of measurements set to zero

SHELL CONSTRUCTS INPUT/OUTPUT INTERFACES FOR ANY TRIM OUTPUT QUANTITIES REQUESTED

solved in implicit part `"ROTOR n REGULATOR SENSOR"`, at end of circulation loop

airframe sensors
blade loads, pitch link loads
blade root force and moment
hub force and moment

without regulator loop, only solved as output

with regulator loop present, solved as interfaces as well, hence available to regulator as measurements

also then available to rest of system, including trim loop (perhaps with regulator loop doing nothing)

ROTOR AND AIRFRAME PARTS

ROTORCRAFT EQUATIONS OF MOTION AND GENERALIZED FORCES USUALLY PERIODIC

so steady state response is periodic

response of rotor subsystem has fundamental frequency Ω in rotating frame (Ω = rotational speed)

ROTORCRAFT SHELL USUALLY CONSTRUCTS ROTORS WITH N IDENTICAL, EQUALLY SPACED BLADES

assuming that these blades have identical trim motion, follows that response of airframe subsystem has fundamental frequency $N\Omega$

analysis can filter solution for airframe, suppressing all harmonics not multiple of fundamental frequency

- for airframe and drive train response, pass only pN harmonics ($p =$ integer)
- for gimbal/teeter response, pass only $pN \pm 1$ harmonics ($p =$ integer)

if rotor subsystem is not axisymmetric, then fundamental frequency of airframe will also be Ω

- axisymmetry can be lost if hub has universal-joint type gimbal
- or if some change is introduced to one blade

ROTOR AND AIRFRAME PARTS SOLVED USING HARMONIC METHOD OR TIME FINITE ELEMENT METHOD

using Fourier series representation of response, so time step can usually be kept large

time step determined by frequency content of generalized forces, not frequency spectrum of equations of motion

EQUATIONS ARE GENERALLY NONLINEAR, SO ITERATIVE PROCEDURE REQUIRED

one iteration produces solution over period (one revolution)

convergence can be controlled using relaxation of generalized forces, and an estimate of aerodynamic damping

AIRFRAME AND DRIVE TRAIN PARTS IDENTIFIED AS TIME-INVARIANT

if two rotors have different rotational speeds, then part is solved for two periods:

primary = first rotor period
secondary = second rotor period

similarly with more than two rotors, part solved for each unique period

if all rotors have same period, then single solution is obtained

ROTOR PARTS

ROTORCRAFT SHELL CONSTRUCTS ROTORS WITH IDENTICAL, EQUALLY SPACED BLADES

SO TRIM SOLUTION WILL BE IDENTICAL MOTION OF BLADES

N-th blade (at azimuth ψ) is reference blade

N = number of blades
$\Delta\psi = 2\pi/N$ = interblade spacing

for rotor with unequal blade spacing, specified blade azimuth angle $\Delta\psi_m$ replaces $m\Delta\psi$

WHEN ROTOR HAS IDENTICAL BLADES UNDERGOING IDENTICAL MOTION, MOST EFFICIENT TO SOLVE FOR ONLY RESPONSE OF ONE BLADE

methods implemented:

solve for response of only reference blade
solve for response of each blade separately
solve for response of all blades together

must solve for response of complete rotor:

if system changed so motion of blades no longer identical

or if blades are structurally coupled, even if blades still have identical motion

if rotor has a gimbal or teeter joint;
perhaps if rotor has flexible swashplate

SOLVE FOR RESPONSE OF REFERENCE BLADE

differential equation parts defined for rotor hub and N-th blade

structural components and interfaces still created for all blades

trim response of reference blade is parent for trim response of other blades (with phase shift)

wing created only for reference blade (unless flutter or transient task initialized or executed)

SOLVE FOR RESPONSE OF EACH BLADE SEPARATELY

differential equation parts defined for rotor hub, and all N blades

SOLVE FOR RESPONSE OF ALL BLADES TOGETHER

differential equation part defined for entire rotor, including hub and all N blades

in cases such as gimballed or teetering hub, identical blades can still have identical motion, even though necessary to solve rotor system as a whole

enforced by replacing harmonic solution for blade motion (with phase shift) by its parent blade motion

parent motion is average (with phase shifts) of solution for all blades

for gimbal and teeter degrees of freedom, also only $pN \pm 1$ harmonics are nonzero

TRIM SOLUTION PROCEDURE

FIGURE 40 DEFINES SOLUTION PROCEDURE CONSTRUCTED FOR TRIM TASK

analysis prints actual solution procedure used

LOOPS AND THEIR SOLUTION METHODS

loop name	solution method	levels	test convergence
`WAKE`	successive substitution	3	no
`TRIM`	Newton-Raphson		yes
`REGULATOR`	regulator		yes
`CIRCULATION`	successive substitution	1	yes
`MOTION`	successive substitution	1	yes
`AIRFRAME`	no solution		
`POST TRIM WAKE`	successive substitution	1	no
`POST TRIM CIRCULATION`	successive substitution	1	yes
`POST TRIM MOTION`	no solution		

wake loop has up to three levels (four with CFD component implemented), and no test for convergence; a specified number of iterations are executed at each level

optionally, output can be obtained for each iteration of wake loop

post-trim circulation loop uses no-solution method for fixed circulation option

```
First Loop = WAKE
    Parts Solved in Loop
    Part = AIRFRAME VELOCITY
    Part = AIRFRAME MEAN VELOCITY
    Part = AIRFRAME LOCATION ROTOR n
    Part = ROTOR n LIFTING LINE POSITION
    Part = ROTOR n COLLOCATION POINTS
    Part = ROTOR n DISK POSITION
    Part = ROTOR n AIRFRAME POSITION
    Part = ROTOR n WING MEAN FRAME
    Part = ROTOR n DUCT FORCE
    Part = ROTOR n MEAN DUCT FORCE
    Part = ROTOR n FRAME
    Part = ROTOR n MEAN FRAME
    Part = ROTOR n WING LOAD
    Part = ROTOR n WING MEAN LOAD
    Part = ROTOR n WAKE GEOMETRY
    Part = ROTOR n WAKE GEOMETRY SENSOR
    Part = ROTOR n WAKE GEOMETRY DISPLAY
    Part = ROTOR n WAKE INFLUENCE COEFF
    Part = ROTOR n WAKE INFL COEFF SENSOR
        Child Loop = TRIM
            Parts Solved at End of Loop
            Part = AIRFRAME SENSOR ROTOR n
            Part = ROTOR n HUB LOAD SENSOR
            Part = ROTOR n CONTROL LOAD SENSOR
            Part = ROTOR n BLADE LOAD SENSOR
            Part = ROTOR n BLADE POSITION SENSOR
            Part = ROTOR n WING SENSOR
                Modules Written at End of Loop
                Module Convergence = WAKE
                Module Shell = ROTORCRAFT 11
                Module Shell = ROTOR n
                Module Shell = FRAMES 10
                Module Part = ROTOR n HUB
                Module Part = ROTOR n BLADE N
                Module Part = AIRFRAME
                Module Part = DRIVE TRAIN
                Module Output of Part = ROTOR n INTVEL TRANSFORM
                Module Output of Part = AIRFRAME SENSOR ROTOR n
                Module Output of Part = ROTOR n HUB LOAD SENSOR
                Module Output of Part = ROTOR n CONTROL LOAD SENSOR
                Module Output of Part = ROTOR n BLADE LOAD SENSOR
                Module Output of Part = ROTOR n BLADE POSITION SENSOR
                Module Output of Part = ROTOR n WING SENSOR
                Module Output of Part = ROTOR n WAKE GEOMETRY DISPLAY
                Module Graphics, Part = ROTOR n GRAPHICS
```

Figure 6-40a Trim solution procedure.

```
Loop = TRIM
      Parts Solved in Loop
      Part = CONTROL
      Part = ROTOR n CONTROL
         Pass 1 = ROTOR n HARMONICS
         Pass 2 = ROTOR n CONTROL
                  Child Loop = CIRCULATION

Loop = CIRCULATION
      Parts Solved in Loop
      Part = ROTOR n UNIFORM INFLOW
      Part = ROTOR n UNIFORM INFLOW SENSOR
      Part = ROTOR n WAKE VORTICITY
      Part = ROTOR n NONUNIFORM INFLOW
      Part = AIRFRAME FLOW FIELD ROTOR n
      Part = ROTOR n INTVEL TRANSFORM
      Part = ROTOR n INTVEL ADDITION
      Part = ROTOR n INTVEL SPAN AVERAGE
      Part = ROTOR n INTVEL BLADE AVERAGE
      Part = ROTOR n INTVEL MEAN
      Part = ROTOR n INTVEL TOTAL
            Child Loop = MOTION
                  Parts Solved at End of Loop
                  Part = ROTOR n TORQUE
                  Part = ROTOR n MEAN TORQUE
                  Part = ROTOR n PERFORMANCE
                     Pass 1 = ROTOR n POWER
                     Pass 2 = ROTOR n PERFORMANCE
                  Part = ROTORCRAFT PERFORMANCE
```

Figure 6-40b Trim solution procedure.

```
Loop = MOTION
      Parts Solved in Loop
      Part = ROTOR n HUB
      Part = ROTOR n BLADE N
         Pass 1 = ROTOR n AERODYNAMIC INTERFACE (V)
         Pass 2 = ROTOR n AERODYNAMIC INTERFACE (R)
         Pass 3 = ROTOR n WING AERODYNAMICS
         Pass 4 = ROTOR n AERODYNAMIC INTERFACE (F)
      Part = ROTOR n HUB
            Child Loop = AIRFRAME
                  Parts Solved at End of Loop
                  Part = ROTOR n DISK POSITION
                  Part = ROTOR n AIRFRAME POSITION
                  Part = ROTOR n TIP PATH PLANE
                  Part = ROTOR n MEAN TIP PATH PLANE
                  Part = ROTOR n WING LOAD
                  Part = ROTOR n WING MEAN LOAD
                  Part = ROTOR n WING FRAME
                  Part = ROTOR n WING MEAN FRAME
                  Part = ROTOR n DUCT FORCE
                  Part = ROTOR n MEAN DUCT FORCE
                  Part = ROTOR n FRAME
                  Part = ROTOR n MEAN FRAME

Last Loop = AIRFRAME
      Parts Solved in Loop
      Part = AIRFRAME AERODYNAMICS
      Part = AIRFRAME
      Part = DRIVE TRAIN
                  Parts Solved at End of Loop
                  Part = AIRFRAME VELOCITY
                  Part = AIRFRAME MEAN VELOCITY
                  Part = AIRFRAME LOCATION ROTOR n
                  Part = DRIVE TRAIN MEAN GOVERNOR
                  Part = AIRFRAME LOAD
                  Part = AIRFRAME MEAN LOAD
```

Figure 6-40c Trim solution procedure.

```
First Loop = POST TRIM WAKE
      Parts Solved in Loop
      Part = RTREX n LIFTING LINE POSITION
      Part = RTREX n COLLOCATION POINTS
      Part = RTREX n MOTION
      Part = RTREX n WING LOAD
      Part = RTREX n WAKE GEOMETRY
      Part = RTREX n WAKE GEOMETRY SENSOR
      Part = RTREX n WAKE GEOMETRY DISPLAY
      Part = RTREX n WAKE INFLUENCE COEFF
      Part = RTREX n WAKE INFL COEFF SENSOR
            Child Loop = POST TRIM CIRCULATION
                  Parts Solved at End of Loop
                  Part = RTREX n WING SENSOR
                        Modules Written at End of Loop
                        Module Convergence = POST TRIM WAKE
                        Module Output of Part = RTREX n INTVEL TRANSFORM
                        Module Output of Part = RTREX n WING SENSOR
                        Module Output of Part = RTREX n WAKE GEOMETRY DISPLAY
                        Module Graphics, Part = RTREX n GRAPHICS

Loop = POST TRIM CIRCULATION
      Parts Solved in Loop
      Part = RTREX n WAKE VORTICITY
      Part = RTREX n NONUNIFORM INFLOW
      Part = RTREX n INTVEL TRANSFORM
      Part = RTREX n INTVEL ADDITION
            Child Loop = POST TRIM MOTION

Loop = POST TRIM MOTION
      Parts Solved in Loop
      Part = RTREX n MOTION
      Part = RTREX n WING LOAD
```

Figure 6-40d Trim solution procedure.

6–10 Transient Task

TRANSIENT TASK NUMERICALLY INTEGRATES SYSTEM EQUATIONS, FROM TRIM SOLUTION, FOR PRESCRIBED EXCITATION

ROTORCRAFT SHELL CONSTRUCTS TRANSIENT SOLUTION PROCEDURE WITH NO LOOPS (NO ITERATION):

> one loop (no solution) to organize process,
> entire system solved in one part (integration method)
>
> > relaxation applied to bound circulation, for convergence when nonuniform inflow model used

TRANSIENT TASK WITH VORTEX WAKE MODEL

> NUMERICAL INTEGRATION SHOULD BE ITERATED AT EACH TIME STEP TO ACHIEVE CONSISTENT, CONVERGED SOLUTION
>
> > with nonuniform inflow wake model, relaxation of circulation required for this iteration to converge
>
> MORE CHAOTIC BEHAVIOR OF FREE WAKE GEOMETRY CAN BE EXPECTED IN TRANSIENT TASK
>
> > compared to trim task, where periodicity enforced in wake distortion solution
> >
> > might use growth of core radius to control this chaotic behavior

6–11 Flutter Task

FLUTTER TASK LINEARIZES SYSTEM EQUATIONS, ABOUT TRIM SOLUTION; THEN ANALYZES RESULTING DIFFERENTIAL EQUATIONS

SHELL CONSTRUCT FLUTTER EQUATIONS

for entire system,
or for independent blade (of N-bladed rotor)

FLUTTER EQUATIONS CAN BE TIME INVARIANT OR PERIODIC

time invariant equations obtained for truly time invariant system, or by averaging equations of periodic system (in first loop)

FLUTTER ANALYSIS PERIOD T

for analysis of entire system: $2\pi/T = N\Omega$ (N odd) or $N\Omega/2$ (N even)

N = number of blades
Ω = rotational speed of first rotor

for independent blade analysis, or without multiblade coordinates: $2\pi/T = \Omega$

can specify T = period of first rotor, for problems such as non-identical blades

in more general cases, core input can be used to redefine flutter period

if second (or other) rotor has period not equal to system period, then its equations must be averaged by child loop

FLUTTER LOOPS

PARTITIONED PROCEDURE CONSTRUCTED TO GENERATE EQUATIONS

system is divided into parts, which obtain subset of differential equations

loops combine and operate on part solutions, to obtain system equations

PRINCIPAL LOOPS ARE FOR ROTOR, AIRFRAME, AND DRIVE TRAIN SUBSYSTEMS

rotor loop has child loops for blade, hub, and inflow

each loop has corresponding differential equations part

SEPARATE LOOP REQUIRED FOR EACH ROTOR IN ORDER TO IMPLEMENT DIFFERENT OPERATIONS

if rotors turn at different speeds, then always average equations for second (and other) rotor, airframe, and drive train

partitioning system also makes it possible to examine equations of each subsystem

FLUTTER EQUATIONS DO NOT LINEARIZE FOLLOWING:

prescribed or higher-harmonic control
collocation point positions
rotor tip-path plane tilt or coning
nonuniform inflow or wake geometry
bound circulation or circulation peaks
performance variables

Chapter 7

USING CAMRAD II ANALYSIS

7–1 Output Format

LISTING OF OUTPUT FOR TYPICAL CAMRAD II ANALYSIS

header and job parameters
case input, rotor elements constructed
shell header
shell input data
core and table input data
trim solution procedure, flutter solution procedure
trim loop iteration and part solution
trim convergence
trim solution: rotorcraft performance
trim solution: rotor performance
trim solution: analysis frames
trim solution: output of part
flutter solution procedure
flutter loop iteration and part solution
flutter solution: mode set
flutter solution: description of variables
flutter solution: analysis
computation times, data vector statistics
end of case

```
                    ******************************************************************************
                    *                                                                            *
                    *                                 CAMRAD II                                  *
                    *   COMPREHENSIVE ANALYTICAL MODEL OF ROTORCRAFT AERODYNAMICS AND DYNAMICS   *
                    *                                RELEASE 5.0                                 *
                    *                            JOHNSON AERONAUTICS                             *
                    *                     COPYRIGHT 1992-2020 WAYNE JOHNSON                      *
                    *                                                                            *
                    ******************************************************************************


PLUGINS INSTALLED:
NONE


READING NAMELIST NLJOB


JOB DESCRIPTION
---------------

    NUMBER OF CASES,  NCASES =    1
    PLOT FILE WRITE (0 FOR NEVER, 1 FOR ONE FILE PER JOB),  PLFILE =   1

    INPUT CATEGORY (0 FOR CORE, 1 FOR SHELL),                                      INFILE =   1
    SHELL INPUT FILE READ (0 FOR NEVER, 1 FOR FIRST CASE ONLY, 2 FOR EACH CASE),      OPSHLL =   1
    EMPTY INPUT DATA VECTORS FOR NEXT CASE (0 NEVER, 1 SHELL/CORE, 2 TABLES, 3 ALL),  OPREIN =   0
    DELETE TABLES NOT USED BY THIS CASE (0 FOR NEVER),                             OPTABL =   1
    INITIALIZE TRIM RESPONSE FROM PREVIOUS CASE (0 NONE, 1 LOOP VARIABLES, 6 PART SOLUTION, 7 BOTH),         OPINIT =   0
    INITIALIZE TRIM SOLUTION FROM PREVIOUS JOB, READ FILE (0 NEVER, 1 ONLY FIRST CASE, 2 SAME, 3 SEPARATE),  OPITJR =   0
    INITIALIZE TRIM SOLUTION FROM PREVIOUS JOB, WRITE FILE (0 NEVER, 1 EACH CASE),                           OPITJW =   0
    RESTART TRANSIENT SOLUTION FROM PREVIOUS JOB, READ FILE (0 NEVER),                                       OPRNJR =   0
    RESTART TRANSIENT SOLUTION FROM PREVIOUS JOB, WRITE FILE (0 NEVER),                                      OPRNJW =   0
    TRACE OF OPERATION (0 TO 4; 0 NONE, 4 HIGH LEVEL PRINT); FOR JOB START (DATA VECTOR UTILITIES),  JOBTRC =   0
    TRACE OF OPERATION (0 TO 4; 0 NONE, 4 HIGH LEVEL PRINT); FOR CASE INPUT,                         CSITRC =   0
    TRACE OF OPERATION (0 TO 4; 0 NONE, 4 HIGH LEVEL PRINT); FOR TIMER AND DEBUG COUNTER,            TIMTRC =   0

INPUT/OUTPUT UNIT NUMBERS

    NAMELIST INPUT,                                          NUIN   =   5
    PRINTER OUTPUT,                                          NUOUT  =   6
    SHELL INPUT DATA FILE,                                   NFSHLL =  41
    CORE INPUT DATA FILE,                                    NFCORE =  42
    PLOT DATA FILE,                                          NFPLOT =  43
    TRIM INITIALIZATION INPUT FILE,                          NFITJR =  44
    TRIM INITIALIZATION OUTPUT FILE,                         NFITJW =  45
    TRANSIENT RESTART INPUT FILE,                            NFRNJR =  46
    TRANSIENT RESTART OUTPUT FILE,                           NFRNJW =  47
    DATA VECTOR STATISTICS FILE,                             NFSTAT =  48
    TABLE FILES DEFAULT (NFTABL FOR ALL),                    NFTABL =  49

SYSTEM PARAMETERS

    FILE OPEN UTILITY (0 FOR NO ACTION, 1 TO OPEN FILE),                                  OPNFIL =         1
    FILE CLOSE UTILITY (0 FOR NO ACTION, 1 TO CLOSE FILE, 2 TO REWIND FILE),              CLSFIL =         1
    MAXIMUM FILE UNIT NUMBER (0 FOR NO MAXIMUM, -1 FOR CONSTANT UNIT NUMBER),             NFUMAX =        -1
```

```
    FILE DEFINITION FOR INTERACTIVE MODE (1 NAME, 2 UNIT NUMBER, 3 BOTH),                     DEFFIL =         1
    BLOCK COMPRESSION IN DATA VECTOR (0 FOR NONE, 1 TO COMPRESS),                             BLKCMP =         1
    DATA VECTOR STATISTICS FILE (0 FOR NONE, 1 TO APPEND TO OLD FILE, 2 TO CREATE NEW FILE),  DVSFIL =         0
    ACTION ON ERROR (0 FOR NONE, 1 TO EXIT, 2-3 FOR TRACEBACK),                               ERRACT =         1
    ANALYSIS EXIT (0 FOR NO ACTION, 1 TO STOP ON ABNORMAL EXIT, 2 TO STOP),                   OPEXIT =         1
    NUMERICAL PRECISION IN MATHEMATICS UTILITIES (1 SINGLE, 2 DOUBLE),                        OPMATH =         2
    EIGENANALYSIS METHOD (1 FOR FIRST, 2 FOR SECOND),                                         OPEIGN =         2


CREATING PLOT FILE FOR JOB
OPEN FILE 43, NAME:  PLOTFILE = [CAMRADII]ROTOR.PLOT


SHELL INPUT, CASE NUMBER   1
OPEN FILE 41, NAME:  SHELLINPUT = [CAMRADII.INPUT]ROTORINPUT.DAT
READING DATA VECTOR "SHLLCM" (TYPE = SHELL ) FROM FILE 41, FILE IDENTIFICATION = HH:MM:SS   DD-MMM-YY
CLOSE FILE 41
SHELL NAMELIST INPUT
READING NAMELIST NLDEF, SHELL INPUT
   ACTION = CHANGE OLD SECTION (INITIALIZE IF NEW); CLASS = CASE
READING NAMELIST NLVAL
READING NAMELIST NLDEF, SHELL INPUT
   ACTION = CHANGE OLD SECTION (INITIALIZE IF NEW); CLASS = TRIM
READING NAMELIST NLVAL
READING NAMELIST NLDEF, SHELL INPUT
   ACTION = CHANGE OLD SECTION (INITIALIZE IF NEW); CLASS = TRIM ROTOR
   NAME = ROTOR 1
READING NAMELIST NLVAL
READING NAMELIST NLDEF, SHELL INPUT
   ACTION = CHANGE OLD SECTION (INITIALIZE IF NEW); CLASS = FLUTTER
READING NAMELIST NLVAL
READING NAMELIST NLDEF, SHELL INPUT
   ACTION = CHANGE OLD SECTION (INITIALIZE IF NEW); CLASS = FLUTTER ROTOR
   NAME = ROTOR 1
READING NAMELIST NLVAL
READING NAMELIST NLDEF, SHELL INPUT
   ACTION = END OF INPUT
GENERATING CORE INPUT FROM SHELL INPUT

         GENERATING CORE INPUT FOR SHELL CASE DATA
         GENERATING CORE INPUT FOR SHELL ROTORCRAFT DATA
         GENERATING CORE INPUT FOR SHELL AIRFRAME DATA

                             AIRFRAME, COMPONENT MASS = 0.00000E+00
                             COMPONENT CG, FUSELAGE STATION =  0.00000E+00, BUTTLINE =  0.00000E+00, WATERLINE =  0.00000E+00
                             ROTOR 1, BLADE MASS =  7.8314    , FIRST MOMENT =  72.062    , ROTATIONAL INERTIA =  1207.9
                               TOTAL ROTOR MASS = N*MB+MSP =  31.326    , BLADE CENTER OF GRAVITY RADIAL STATION =  0.37218
                               BLADE MEAN GEOMETRIC CHORD =  1.8205    , THRUST-WEIGHTED CHORD =  1.8814
                               ROTOR GEOMETRIC SOLIDITY =  0.09375, THRUST-WEIGHTED SOLIDITY =  0.09689
                             AXIS    HUB POSITION          RELATIVE COMPONENT CG    HUB ORIENTATION, ROTATION MATRIX
                               X     FS =  0.00000E+00     0.00000E+00             -1.00000    0.00000    0.00000
                               Y     BL =  0.00000E+00     0.00000E+00              0.00000    1.00000    0.00000
                               Z     WL =  0.00000E+00     0.00000E+00              0.00000    0.00000   -1.00000
                             CONSTRUCTING COMPONENT (LINEAR NORMAL MODES),  NAME = AIRFRAME
                             CONSTRUCTING COMPONENT (RIGID BODY),           NAME = AIRFRAME ROTOR 1
```

```
GENERATING CORE INPUT FOR SHELL ROTOR 1 DATA

          CONSTRUCTING COMPONENT (RIGID BODY),           NAME = ROTOR 1 HUB
          CONSTRUCTING COMPONENT (RIGID BODY),           NAME = ROTOR 1 ROTATING HUB
          CONSTRUCTING COMPONENT (RIGID BODY),           NAME = ROTOR 1 BLADE 1 ROOT
          ROTOR 1 BLADE MODEL (SINGLE LOAD PATH), NUMBER OF NODES =   7
                NODE   SPAN       INBOARD BEAM AXIS         OUTBOARD BEAM AXIS        STRUCTURAL DYN INTERFACE
          NODE  KIND  STATION        X            Z            X            Z            X            Z
            1     0   0.00000                              0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
            2     3   0.03710    0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
            3     6   0.08000    0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
            4     0   0.30000    0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
            5     0   0.60000    0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
            6     0   0.90000    0.00000E+00  0.00000E+00  0.31085E-06  0.00000E+00  0.15543E-06  0.00000E+00
            7     0   1.00000    -1.1373      0.00000E+00
          NODE KIND:  0(1234567)89=OTHER,(JOINT=FLAP,LAG,FLAP-LAG,LAG-FLAP,FLGIMBAL,PITCH,DAMPER),FLEXBEAM,DS
          X/Z = POSITION RELATIVE REFERENCE LINE
          DROOP/SWEEP NODE =   8; PITCH NODE =   3

          CONSTRUCTING BLADE COMPONENTS
          ELEMENT  SPAN STATIONS      JOINT AEROINT    MASS       KIND  UVWT   NAME
            1      0.00000 TO 0.03710   3      0    0.11188E+01   BEAM  0000   ROTOR 1 BLADE 1 ELEMENT 1
            2      0.03710 TO 0.08000   6      0    0.12741E+01   BEAM  0000   ROTOR 1 BLADE 1 ELEMENT 2
            3      0.08000 TO 0.30000   0      2    0.17116E+01   BEAM  3222   ROTOR 1 BLADE 1 ELEMENT 3
            4      0.30000 TO 0.60000   0      4    0.13832E+01   BEAM  3222   ROTOR 1 BLADE 1 ELEMENT 4
            5      0.60000 TO 0.90000   0      9    0.14416E+01   BEAM  3222   ROTOR 1 BLADE 1 ELEMENT 5
            6      0.90000 TO 1.00000   0      6    0.90015E+00   BEAM  3222   ROTOR 1 BLADE 1 ELEMENT 6

GENERATING CORE INPUT FOR SHELL SOLUTION DATA

CORE NAMELIST INPUT
READING NAMELIST NLDEF, CORE INPUT
   ACTION = END OF INPUT

TABLE FILE INPUT, CASE NUMBER   1
REQUIRED TABLE NUMBER   1; CLASS = AIRFOIL, TYPE = CAMRAD, NAME = BLADEAIRFOIL1
READING TABLE FILE; CLASS = AIRFOIL, TYPE = CAMRAD
OPEN FILE 61, NAME:  BLADEAIRFOIL1 = [CAMRADII.AIRFOIL]ROTORAIRFOIL.TAB
       TITLE = ROTOR AIRFOIL (C81 TABLE)
       FILE IDENTIFICATION = HH:MM:SS   DD-MMM-YY
CLOSE FILE 61
REQUIRED TABLE NUMBER   2; CLASS = AIRFOIL, TYPE = CAMRAD, NAME = BLADEAIRFOIL1
TABLE ALREADY PRESENT
REQUIRED TABLE NUMBER   3; CLASS = AIRFOIL, TYPE = CAMRAD, NAME = BLADEAIRFOIL1
TABLE ALREADY PRESENT
REQUIRED TABLE NUMBER   4; CLASS = AIRFOIL, TYPE = CAMRAD, NAME = BLADEAIRFOIL1
TABLE ALREADY PRESENT
```

```
CAMRAD II, COMPREHENSIVE ANALYTICAL MODEL OF ROTORCRAFT AERODYNAMICS AND DYNAMICS                                       RELEASE 5.0
************************************************************************************************************************************
CASE NUMBER   1,  IDENTIFICATION = HH:MM:SS   DD-MMM-YY,  CODE = PERFORMANCE                                 ENGLISH UNITS (FT, SLUG, SEC)

       TITLE    = HELICOPTER
       AIRFRAME = WIND TUNNEL SUPPORT
       ROTOR 1  = HELICOPTER MAIN ROTOR


ANALYSIS TASKS:  EXECUTE TRIM, NO TRANSIENT, EXECUTE FLUTTER
INPUT FILE IDENTIFICATION = HH:MM:SS   DD-MMM-YY

   CONFIGURATION
       WIND TUNNEL
       ONE ROTOR

       AIRFRAME
         GROSS WEIGHT (LB)  =      0.00        CG FUSELAGE STATION (FT)    =      0.00        NUMBER OF ELASTIC MODES    =    0
         CW/SIGMA           =   0.00000        CG BUTTLINE (FT)            =      0.00        NUMBER OF AUXILIARY FORCES =    0
         ROTOR 2 GEAR RATIO =   1.00000        CG WATERLINE (FT)           =      0.00
       NO DRIVE TRAIN MODEL (CONSTANT ROTATIONAL SPEED)
       NO AIRFRAME AERODYNAMICS

       ROTOR 1
         RADIUS (FT)        =    24.724        LOCK NUMBER                 =    7.4096        TIP SPEED (FT/SEC)     =    673.16
         NUMBER OF BLADES   =      4           BLADE MASS (SLUG)           =    7.8314        TIP MACH NUMBER        =    0.6030
         SOLIDITY           =   0.09100        SECOND MOMENT (SLUG-FT**2)  =  1207.854        ROTATIONAL SPEED (RPM) =    260.00
         MEAN CHORD/RADIUS  =   0.07147        REYNOLDS NO (CMEAN*VTIP/NU) = 0.757E+07        OMEGA (RAD/SEC)        =    27.227
         MEAN CHORD (FT)    =    1.7671                                                       BLADE PASS FREQ (HZ)   =    17.333
       CLOCKWISE ROTATION DIRECTION
       SINGLE LOAD PATH; NO SWASHPLATE MECHANISM
       FLAP THEN LAG UNIVERSAL JOINT; PITCH BEARING (WITH CONTROL)


   OPERATING CONDITION
       WIND SPEED *                                AIRFRAME ORIENTATION                          ROTOR SPEED *
         VELOCITY (KNOTS)     =    150.00            YAW ANGLE (DEG)        =      0.00 *          TIP SPEED (FT/SEC)     =    673.16
         V/(OMEGA*R)          =    0.3761            PITCH ANGLE (DEG)      =     -7.00 *          TIP MACH NUMBER        =    0.6030
         MACH NUMBER          =    0.2268            ROLL ANGLE (DEG)       =      0.00 *          ROTATIONAL SPEED (RPM) =    260.00
         VELOCITY (FT/SEC)    =    253.17                                                          OMEGA (RAD/SEC)        =    27.227
         DYN PRESS (LB/FT**2) =     76.21                                                          REYNOLDS NO (/FT)      = 0.428E+07
         REYNOLDS NO (/FT)    = 0.161E+07          HEIGHT ABOVE GROUND (FT) =      0.00

       ENVIRONMENT (FROM DENSITY AND TEMPERATURE)
         DENSITY (SLUG/FT**3) =  0.002378          TEMPERATURE (DEG F)      =     59.00          ALTITUDE MSL (FT)        =       0.0
         DENSITY RATIO        =   1.00000          SOUND SPEED (FT/SEC)     =   1116.45          PRESSURE ALTITUDE (FT)   =       0.0
         VISCOSITY            = 0.374E-06          KINEMATIC VISCOSITY      = 0.157E-03          DENSITY ALTITUDE (FT)    =       0.0

                                                                                     * INITIAL VALUE IF VARIED BY TRIM ITERATION

   CONTROL SETTINGS
       ROTORCRAFT
         PILOT CONTROL (DEG)                     PILOT CONTROL                    GUST (FT/SEC)
           COLLECTIVE         =      9.00 *        PEDAL (DEG) =      0.00 *        LONGITUDINAL      =      0.00
           LATERAL CYCLIC     =      3.00 *        THROTTLE    =      0.00 *        LATERAL           =      0.00
           LONGITUDINAL CYCLIC =     7.00 *      GOVERNOR (DEG)=      0.00 *        VERTICAL          =      0.00
```

```
    AIRFRAME
      AERODYNAMIC CONTROLS (DEG)              AERODYNAMIC CONTROLS (DEG)       ENGINE CONTROL
        FLAPERON           =      0.00          AILERON     =      0.00          THROTTLE          =      0.00
        ELEVATOR           =      0.00          RUDDER      =      0.00
    ROTOR 1
      PRIMARY CONTROLS (DEG)                  BLADE CONTROL (DEG)              HIGHER HARMONIC CONTROL (NUMBER OF HARMONICS)
        COLLECTIVE         =      0.00          PITCH       =      0.00          ROTATING FRAME    =   0      TE FLAP =   0
        LATERAL CYCLIC     =      0.00          TE FLAPS    =      0.00          NONROTATING FRAME =   0
        LONGITUDINAL CYCLIC =     0.00          JETS        =   0.00000E+00      AUXILIARY FORCES  =   0
    AIRFRAME AND ROTOR CONTROLS ARE VALUES WITH ZERO PILOT AND GOVERNOR CONTROLS


TRIM SOLUTION PROCEDURE
    WAKE LOOP:  NONUNIFORM INFLOW WITH PRESCRIBED WAKE GEOMETRY
      NUMBER OF ITERATIONS =   1 (UNIFORM),   1 (PRESCRIBED),   1 (FREE)
      RELAXATION FACTOR = 1.00000 (UNIFORM), 1.00000 (PRESCRIBED), 1.00000 (FREE)
    TRIM LOOP:  NUMBER OF VARIABLES =   3, NUMBER OF ITERATIONS =  40, RELAXATION FACTOR = 0.20000, TOLERANCE =   5.00000
      TRIM VARIABLE                      TRIM QUANTITY                        TARGET
      COLL                               CT/S                                 CTTRIM =  0.080000
      LATCYC                             BETAS                                BCTRIM =  0.000000
      LNGCYC                             BETAC                                BSTRIM =  0.000000
    CIRCULATION LOOP:  NUMBER OF ITERATIONS =  200, RELAXATION FACTOR = 0.05000, TOLERANCE =   1.00000
    MOTION LOOP:       NUMBER OF ITERATIONS =   40, RELAXATION FACTOR = 1.00000, TOLERANCE =   2.00000
    PART SOLUTION:  NUMBER OF AZIMUTH STEPS PER REVOLUTION =   24, AZIMUTH INCREMENT (DEG) =   15.00

    ROTOR 1 PART SOLUTION
      NONUNIFORM INFLOW WITH PRESCRIBED WAKE GEOMETRY
      SINGLE ROTOR PART FOR REFERENCE BLADE; HARMONIC SOLUTION METHOD
      ROTOR:        10 HARMONICS,  40 ITERATIONS, RELAXATION FACTOR = 0.50000, TOLERANCE =   2.00000, DAMPING = 1.00000
      AIRFRAME:     10 HARMONICS,  40 ITERATIONS, RELAXATION FACTOR = 1.00000, TOLERANCE =   2.00000, DAMPING = 0.50000


    DEGREES OF FREEDOM IN TRIM SOLUTION (0 ZERO, 1 DYNAMIC, 2 QUASISTATIC)
      AIRFRAME
      ROTOR 1
         BLADE MODES,                      DOFM = 1111111111   1122222222   2222222222   2222222222


FLUTTER EQUATIONS
    TIME INVARIANT SYSTEM
    CONSTANT COEFFICIENT APPROXIMATION; INDEPENDENT BLADE ANALYSIS
    ROTOR 1:  NO ROTOR AERODYNAMICS
    ANALYSIS TASKS:  EIGENANALYSIS; NO TIME HISTORY RESPONSE; NO FREQUENCY RESPONSE; NO RMS GUST RESPONSE


FLUTTER SOLUTION PROCEDURE
    NUMBER OF STEPS PER REVOLUTION IN AVERAGE OF MATRICES =   12
    REDUCTION OF EQUATIONS:  ZERO IN LOOPS; QUASISTATIC IN LOOPS

    TIME HISTORY RESPONSE:  FROM MODES
    FREQUENCY RESPONSE:  CALCULATED FROM MODES
    RMS GUST RESPONSE:  STOCHASTIC (FROM MODES)


    VARIABLES IN FLUTTER SOLUTION
    DEGREES OF FREEDOM (0 ZERO, 1 DYNAMIC, 2 QUASISTATIC)
```

```
  AIRFRAME
  ROTOR 1
     BLADE MODES,                       DOFM = 1111111111   2222222222   2222222222   2222222222
     DYNAMIC INFLOW,                    DOFL = 111
     MULTIBLADE COORDINATES,          DOFMBC = 1 11 11 1
SPECIFICATION OF ORDER (0 DETERMINE FROM EQUATIONS, 1 FIRST ORDER, 2 SECOND ORDER
  AIRFRAME DEGREES OF FREEDOM,        DOFORD = 0000  0

          ORDER DOFA = UNSTEADY LOADS, DYNAMIC STALL
          ORDER DOFL = UNIFORM, LONGITUDINAL, LATERAL
          ORDER DOFMBC = (0, 1C,1S, NC,NS, N/2)
          ORDER DOFORD = AIRFRAME RIGID BODY (X,Y,Z,PSI) + DRIVE TRAIN RIGID (PSI)

CONTROLS (0 ZERO, 1 USED)
  ROTORCRAFT
     PILOT CONTROL,                     CONP = 00000
     GUST,                              GUST = 000
  AIRFRAME
     AERODYNAMIC CONTROLS,              CONA = 0000
     ENGINE CONTROL,                    CONE = 0
  ROTOR 1
     PRIMARY CONTROLS,                  CONR = 000
     BLADE PITCH,                       CONB = 0
     BLADE TE FLAP,                     COND = 0
     BLADE REACTION JET,                CONJ = 0
     MULTIBLADE COORDINATES,          CONMBC = 1 11 11 1

          ORDER CONP = COLLECTIVE, LATERAL CYCLIC, LONGITUDINAL CYCLIC, PEDAL, THROTTLE
          ORDER GUST = LONGITUDINAL, LATERAL, VERTICAL
          ORDER CONA = FLAPERON, ELEVATOR, AILERON, RUDDER
          ORDER CONR = COLLECTIVE, LATERAL CYCLIC, LONGITUDINAL CYCLIC
          ORDER CONMBC = (0, 1C,1S, NC,NS, N/2)

OUTPUT (0 ZERO, 1 USED)
  AIRFRAME
     AIRFRAME SENSORS,                 MSSEN = 0
  ROTOR 1
     HUB LOAD SENSORS,                 MHSEN = 0
     CONTROL LOAD SENSORS,             MCSEN = 0
     BLADE LOAD SENSORS,               MBSEN = 0
     BLADE POSITION SENSORS,           MPSEN = 0
     BLADE AERODYNAMICS,               MASEN = 0
     MULTIBLADE COORDINATES,          SENMBC = 1 11 11 1

          ORDER SENMBC = (0, 1C,1S, NC,NS, N/2)
```

```
****************
SHELL INPUT DATA
****************

-------------------------------------------------------------------------------------------------------------------
CLASS = CASE                                                                                 NUMBER OF BLOCKS =      1
-------------------------------------------------------------------------------------------------------------------
CLASS = TRIM                                                                                 NUMBER OF BLOCKS =      1
-------------------------------------------------------------------------------------------------------------------
CLASS = TRIM ROTOR                                                                           NUMBER OF BLOCKS =      1
-------------------------------------------------------------------------------------------------------------------
CLASS = FLUTTER                                                                              NUMBER OF BLOCKS =      1
-------------------------------------------------------------------------------------------------------------------
CLASS = FLUTTER ROTOR                                                                        NUMBER OF BLOCKS =      1
-------------------------------------------------------------------------------------------------------------------
CLASS = AIRFRAME                          TYPE = STRUCTURE                                   NUMBER OF BLOCKS =      1
CLASS = AIRFRAME                          TYPE = AERODYNAMICS                                NUMBER OF BLOCKS =      1
CLASS = AIRFRAME                          TYPE = CONTROL                                     NUMBER OF BLOCKS =      1
CLASS = AIRFRAME                          TYPE = DRIVE TRAIN                                 NUMBER OF BLOCKS =      1
-------------------------------------------------------------------------------------------------------------------
CLASS = ROTOR                             TYPE = STRUCTURE                                   NUMBER OF BLOCKS =      1
CLASS = ROTOR                             TYPE = AERODYNAMICS                                NUMBER OF BLOCKS =      1
CLASS = ROTOR                             TYPE = INFLOW                                      NUMBER OF BLOCKS =      1
CLASS = ROTOR                             TYPE = WAKE                                        NUMBER OF BLOCKS =      1
-------------------------------------------------------------------------------------------------------------------
CLASS = TABLES                                                                               NUMBER OF BLOCKS =      1
-------------------------------------------------------------------------------------------------------------------


UNITS FOR INPUT PARAMETERS
         DIMENSIONAL INPUT PARAMETERS CAN BE IN ENGLISH OR METRIC (SI) UNITS,
         USING A CONSISTENT LENGTH-MASS-TIME SYSTEM
              ENGLISH (OPUNIT = 1)     FOOT-SLUG-SECOND
              METRIC  (OPUNIT = 2)     METER-KILOGRAM-SECOND
         WITH THE FOLLOWING EXCEPTIONS
              ROTORCRAFT GROSS WEIGHT (LB OR KG); AIRCRAFT SPEED AND WIND SPEED (KNOTS)
         ANGLES ARE INPUT IN DEGREES


---------------------------------------------------------------------------------------------------------------------------------
CLASS = xxxxxxxxxxxxxxxxxxxx              TYPE = xxxxxxxxxxxxxxxxxxxx              NAME = xxxxxxxxxxxxxxxxxxxx
---------------------------------------------------------------------------------------------------------------------------------
ORIGIN = FILE (IDENT = HH:MM:SS   DD-MMM-YY), MODIFICATION BY NAMELIST
....
....
....
```

```
***************
CORE INPUT DATA
***************

-----------------------------------------------------------------------------------------------------------------------
CLASS = CASE                                                                                   NUMBER OF BLOCKS =     1
-----------------------------------------------------------------------------------------------------------------------
CLASS = COMPONENT                          TYPE = RIGID BODY                                   NUMBER OF BLOCKS =     7
CLASS = COMPONENT                          TYPE = LINEAR NORMAL MODES                          NUMBER OF BLOCKS =     1
CLASS = COMPONENT                          TYPE = FINITE ELEMENT BEAM                          NUMBER OF BLOCKS =    24
CLASS = COMPONENT                          TYPE = REFERENCE FRAME                              NUMBER OF BLOCKS =   245
CLASS = COMPONENT                          TYPE = FILTER                                       NUMBER OF BLOCKS =    29
CLASS = COMPONENT                          TYPE = REFERENCE PLANE                              NUMBER OF BLOCKS =     1
CLASS = COMPONENT                          TYPE = DIFFERENTIAL EQUATION                        NUMBER OF BLOCKS =     9
CLASS = COMPONENT                          TYPE = FOURIER SERIES                               NUMBER OF BLOCKS =     4
CLASS = COMPONENT                          TYPE = GUST                                         NUMBER OF BLOCKS =     5
CLASS = COMPONENT                          TYPE = LIFTING LINE WING                            NUMBER OF BLOCKS =     4
CLASS = COMPONENT                          TYPE = ROTOR INFLOW                                 NUMBER OF BLOCKS =     1
CLASS = COMPONENT                          TYPE = WING WAKE                                    NUMBER OF BLOCKS =     1
CLASS = COMPONENT                          TYPE = ROTOR WAKE GEOMETRY                          NUMBER OF BLOCKS =     1
CLASS = COMPONENT                          TYPE = ROTOR PERFORMANCE                            NUMBER OF BLOCKS =     1
CLASS = COMPONENT                          TYPE = ROTORCRAFT PERFORMANCE                       NUMBER OF BLOCKS =     1
-----------------------------------------------------------------------------------------------------------------------
CLASS = FRAME                                                                                  NUMBER OF BLOCKS =     7
-----------------------------------------------------------------------------------------------------------------------
CLASS = INTERFACE                          TYPE = STRUCTURAL DYNAMIC                           NUMBER OF BLOCKS =    31
CLASS = INTERFACE                          TYPE = INPUT/OUTPUT                                 NUMBER OF BLOCKS =  1563
-----------------------------------------------------------------------------------------------------------------------
CLASS = OUTPUT                                                                                 NUMBER OF BLOCKS =   120
-----------------------------------------------------------------------------------------------------------------------
CLASS = INPUT                                                                                  NUMBER OF BLOCKS =     6
-----------------------------------------------------------------------------------------------------------------------
CLASS = WIND                                                                                   NUMBER OF BLOCKS =     1
-----------------------------------------------------------------------------------------------------------------------
CLASS = OPERATING CONDITION                                                                    NUMBER OF BLOCKS =     1
-----------------------------------------------------------------------------------------------------------------------
CLASS = PERIOD                                                                                 NUMBER OF BLOCKS =     3
-----------------------------------------------------------------------------------------------------------------------
CLASS = TRIM LOOP                          TYPE = NO SOLUTION                                  NUMBER OF BLOCKS =     1
CLASS = TRIM LOOP                          TYPE = SUCCESSIVE SUBSTITUTION                      NUMBER OF BLOCKS =     3
CLASS = TRIM LOOP                          TYPE = NEWTON RAPHSON                               NUMBER OF BLOCKS =     1
-----------------------------------------------------------------------------------------------------------------------
CLASS = TRIM PART                          TYPE = NO SOLUTION                                  NUMBER OF BLOCKS =     5
CLASS = TRIM PART                          TYPE = IMPLICIT                                     NUMBER OF BLOCKS =    41
CLASS = TRIM PART                          TYPE = HARMONIC                                     NUMBER OF BLOCKS =     3
-----------------------------------------------------------------------------------------------------------------------
CLASS = FLUTTER                                                                                NUMBER OF BLOCKS =     1
-----------------------------------------------------------------------------------------------------------------------
CLASS = FLUTTER LOOP                                                                           NUMBER OF BLOCKS =     3
-----------------------------------------------------------------------------------------------------------------------
CLASS = FLUTTER PART                       TYPE = NO SOLUTION                                  NUMBER OF BLOCKS =   861
CLASS = FLUTTER PART                       TYPE = INTERFACE                                    NUMBER OF BLOCKS =     2
CLASS = FLUTTER PART                       TYPE = DIFFERENTIAL EQUATIONS                       NUMBER OF BLOCKS =     1
-----------------------------------------------------------------------------------------------------------------------
CLASS = MODES                                                                                  NUMBER OF BLOCKS =     5
-----------------------------------------------------------------------------------------------------------------------
```

```
CLASS = RESPONSE                          TYPE = RIGID                                     NUMBER OF BLOCKS =     32
CLASS = RESPONSE                          TYPE = VARIABLE                                  NUMBER OF BLOCKS =   1734
----------------------------------------------------------------------------------------------------------------------
CLASS = WEIGHTS                                                                            NUMBER OF BLOCKS =      1
----------------------------------------------------------------------------------------------------------------------
```

```
****************
TABLE INPUT DATA
****************

---------------------------------------------------------------------------------------------------------------------
CLASS = AIRFOIL                           TYPE = CAMRAD                                NUMBER OF TABLES =      1
---------------------------------------------------------------------------------------------------------------------

    ==========================================================================================================================
    INDEX   USAGE   CLASS                                TYPE                                 NAME
    --------------------------------------------------------------------------------------------------------------------------
        1   YES     AIRFOIL                              CAMRAD                               BLADEAIRFOIL1
    ==========================================================================================================================
    INDEX           FILE NAME                            FILE IDENTIFICATION                  FILE UNIT NUMBER
    --------------------------------------------------------------------------------------------------------------------------
        1           BLADEAIRFOIL1                        HH:MM:SS   DD-MMM-YY                 61
    ==========================================================================================================================
    INDEX           TITLE
    --------------------------------------------------------------------------------------------------------------------------
        1           ROTOR AIRFOIL (C81 TABLE)
    ==========================================================================================================================

INITIALIZATION
GENERATE RECORDS FROM CORE BLOCKS, FIRST PASS
GENERATE RECORDS FROM CORE BLOCKS, SECOND PASS
GENERATE RECORDS FROM CORE BLOCKS, THIRD PASS
INITIALIZE RECORDS, FIRST PASS
INITIALIZE RECORDS, SECOND PASS
INITIALIZE RECORDS, THIRD PASS
INITIALIZE RECORDS, FOURTH PASS
INITIALIZE RESPONSE, FIRST PASS
INITIALIZE RESPONSE, SECOND PASS
INITIALIZE RESPONSE, THIRD PASS
GENERATE MATRICES AND VECTORS
INITIALIZE RESPONSE SUBVECTORS
```

```
***********************
SYSTEM AND FUNCTIONALITY
***********************

-----------------------
TRIM SOLUTION PROCEDURE
-----------------------

     ==========================================================================================
     FIRST LOOP = WAKE
              PARTS SOLVED IN LOOP
              PART = ROTOR 1 LIFTING LINE POSITION
              PART = ROTOR 1 COLLOCATION POINTS
              PART = ROTOR 1 DISK POSITION
              PART = ROTOR 1 WING MEAN FRAME
              PART = ROTOR 1 FRAME
              PART = ROTOR 1 MEAN FRAME
              PART = ROTOR 1 WING LOAD
              PART = ROTOR 1 WING MEAN LOAD
              PART = ROTOR 1 WAKE GEOMETRY
              PART = ROTOR 1 WAKE GEOMETRY SENSOR
              PART = ROTOR 1 WAKE INFLUENCE COEFF
              PART = ROTOR 1 WAKE INFL COEFF SENSOR
                       CHILD LOOP = TRIM
                                PARTS SOLVED AT END OF LOOP
                                PART = ROTOR 1 BLADE POSITION SENSOR
                                PART = ROTOR 1 WING SENSOR
                                         MODULES WRITTEN AT END OF LOOP
                                         MODULE = CONVERGENCE, LOOP = WAKE
                                         MODULE = SHELL  1000
                                         MODULE = SHELL     1
                                         MODULE = SHELL  2000
                                         MODULE = OUTPUT OF PART = ROTOR 1 BLADE POSITION SENSOR
                                         MODULE = OUTPUT OF PART = ROTOR 1 WING SENSOR
     ------------------------------------------------------------------------------------------
     LOOP = TRIM
              PARTS SOLVED IN LOOP
              PART = CONTROL
              PART = ROTOR 1 CONTROL
                       CHILD LOOP = CIRCULATION
     ------------------------------------------------------------------------------------------
     LOOP = CIRCULATION
              PARTS SOLVED IN LOOP
              PART = ROTOR 1 UNIFORM INFLOW
              PART = ROTOR 1 UNIFORM INFLOW SENSOR
              PART = ROTOR 1 WAKE VORTICITY
              PART = ROTOR 1 NONUNIFORM INFLOW
              PART = ROTOR 1 INTVEL TRANSFORM
              PART = ROTOR 1 INTVEL ADDITION
              PART = ROTOR 1 INTVEL SPAN AVERAGE
              PART = ROTOR 1 INTVEL BLADE AVERAGE
              PART = ROTOR 1 INTVEL MEAN
              PART = ROTOR 1 INTVEL TOTAL
                       CHILD LOOP = MOTION
                                PARTS SOLVED AT END OF LOOP
                                PART = ROTOR 1 TORQUE
```

```
                           PART = ROTOR 1 MEAN TORQUE
                           PART = ROTOR 1 PERFORMANCE
                           PART = ROTORCRAFT PERFORMANCE
     ------------------------------------------------------------------------------------------
     LOOP = MOTION
              PARTS SOLVED IN LOOP
              PART = ROTOR 1 HUB
              PART = ROTOR 1 BLADE 4
              PART = ROTOR 1 HUB
                       CHILD LOOP = AIRFRAME
                           PARTS SOLVED AT END OF LOOP
                           PART = ROTOR 1 DISK POSITION
                           PART = ROTOR 1 TIP PATH PLANE
                           PART = ROTOR 1 MEAN TIP PATH PLANE
                           PART = ROTOR 1 WING LOAD
                           PART = ROTOR 1 WING MEAN LOAD
                           PART = ROTOR 1 WING FRAME
                           PART = ROTOR 1 WING MEAN FRAME
                           PART = ROTOR 1 FRAME
                           PART = ROTOR 1 MEAN FRAME
     ------------------------------------------------------------------------------------------
     LAST LOOP = AIRFRAME
              PARTS SOLVED IN LOOP
              PART = AIRFRAME
                           PARTS SOLVED AT END OF LOOP
                           PART = AIRFRAME LOAD
                           PART = AIRFRAME MEAN LOAD
     ==========================================================================================


---------------------
TRIM SYSTEM EQUATIONS
---------------------

     TOTAL NUMBER OF VECTORS
         MOTION EQUATIONS AND DEGREES OF FREEDOM =     60
         CONSTRAINT EQUATIONS AND INTERFACE VARIABLES =   1594
         OUTPUT EQUATIONS =    122
         INPUT VARIABLES =      6
     NUMBER OF LOOPS =      5
     NUMBER OF PARTS =     49



INITIALIZATION FOR TRIM
GENERATE SUBMATRICES AND SUBVECTORS FOR TRIM PART RECORDS
GENERATE SUBMATRICES FOR TRIM MODES RECORDS
INITIALIZE SCRATCH DATA STRUCTURES FOR TRIM
```

```
-------------------------------------
TRIM LOOP ITERATION AND PART SOLUTION
-------------------------------------


SUCCESSIVE SUBSTITUTION ITERATION, LOOP = WAKE
LEVEL   1 = UNIFORM INFLOW
                                                                                                                                LL
    NEWTON RAPHSON ITERATION, LOOP = TRIM                                                                                       LL
    VARIABLES                                                 QUANTITIES                                                        LL
        COLL          LATCYC        LNGCYC                        CT/S          BETAS         BETAC                             LL
    TARGETS                                                        0.80000E-01   0.00000E+00   0.00000E+00                      LL
    BEGINNING OF ITERATION, LOOP = TRIM                                                                                         LL
          9.0000        3.0000        7.0000                       0.85116E-01  -0.15321      -0.17324                          LL
                                                                                                                                LL
    PERTURBATION IDENTIFICATION, COUNT =    1, LOOP = TRIM                                                                      LL
          8.5000        3.0000        7.0000                       0.70264E-01   0.69957       0.56290                          LL
                                                                                                                                LL
    PERTURBATION IDENTIFICATION, COUNT =    2, LOOP = TRIM                                                                      LL
          9.0000        2.0000        7.0000                       0.85132E-01   -1.2907      -0.16100E-01                      LL
                                                                                                                                LL
    PERTURBATION IDENTIFICATION, COUNT =    3, LOOP = TRIM                                                                      LL
          9.0000        3.0000        8.0000                       0.72022E-01   0.59471        1.3653                          LL
                                                                                                                                LL
                                                                                                                                LL
         PERTURBATION IDENTIFICATION                                                                                            LL
         DERIVATIVE MATRIX D                                                                                                    LL
                         VARIABLE = COLL          LATCYC        LNGCYC                                                          LL
            QUANTITY = CT/S          0.29704E-01  -0.16049E-04  -0.13094E-01                                                    LL
            QUANTITY = BETAS         -1.7056        1.1375       0.74792                                                        LL
            QUANTITY = BETAC         -1.4723      -0.15714        1.5385                                                        LL
         GAIN MATRIX = (RELAXATION FACTOR) * D**-1                                                                              LL
                         QUANTITY = CT/S          BETAS         BETAC                                                           LL
            VARIABLE = COLL           12.445       0.13875E-01   0.99172E-01                                                    LL
            VARIABLE = LATCYC         10.148       0.17607       0.77625E-03                                                    LL
            VARIABLE = LNGCYC         12.945       0.31260E-01   0.22497                                                        LL
                                                                                                                                LL
                                                                                                                                LL
    ITERATION, COUNT =    1, LOOP = TRIM                                                                                        LL
          8.9556        2.9752        6.9775                       0.85254E-01  -0.27723      -0.17706                          LL
                                                                                                                                LL
    ITERATION, COUNT =    2, LOOP = TRIM                                                                                        LL
          8.9117        2.9708        6.9580                       0.84040E-01  -0.27016      -0.14817                          LL
                                                                                                                                LL
    ITERATION, COUNT =    3, LOOP = TRIM                                                                                        LL
          8.8798        2.9775        6.9475                       0.83886E-01  -0.25342      -0.14216                          LL
                                                                                                                                LL
    ITERATION, COUNT =    4, LOOP = TRIM                                                                                        LL
          8.8491        2.9828        6.9371                       0.83472E-01  -0.23580      -0.13311                          LL
                                                                                                                                LL
    ITERATION, COUNT =    5, LOOP = TRIM                                                                                        LL
          8.8223        2.9892        6.9295                       0.82971E-01  -0.21302      -0.11850                          LL
                                                                                                                                LL
    ITERATION, COUNT =    6, LOOP = TRIM                                                                                        LL
          8.8001        2.9966        6.9243                       0.82429E-01  -0.18291      -0.98623E-01                      LL
```

```
                                                                                                                       LL
    ITERATION, COUNT =    7, LOOP = TRIM                                                                               LL
          8.7822        3.0043        6.9208                      0.81910E-01  -0.14614      -0.76324E-01              LL
                                                                                                                       LL
    ITERATION, COUNT =    8, LOOP = TRIM                                                                               LL
          8.7680        3.0107        6.9178                      0.81438E-01  -0.10594      -0.54869E-01              LL
                                                                                                                       LL
    ITERATION, COUNT =    9, LOOP = TRIM                                                                               LL
          8.7570        3.0148        6.9148                      0.81026E-01  -0.66360E-01  -0.36649E-01              LL
                                                                                                                        L
    SUCCESSIVE SUBSTITUTION ITERATION, NO VARIABLES, NUMBER OF LEVELS =   2, LOOP = WAKE                                L
    ITERATION, COUNT =    1, FOR LEVEL   1 = UNIFORM INFLOW                 ; LOOP = WAKE                               L


SUCCESSIVE SUBSTITUTION ITERATION, LOOP = WAKE
LEVEL   2 = NONUNIFORM INFLOW PRSC WAKE GEOM

                                                                                                                       LL
    NEWTON RAPHSON ITERATION, LOOP = TRIM                                                                              LL
    VARIABLES                                                  QUANTITIES                                              LL
        COLL           LATCYC        LNGCYC                        CT/S          BETAS          BETAC                  LL
    TARGETS                                                       0.80000E-01   0.00000E+00   0.00000E+00              LL
    BEGINNING OF ITERATION, LOOP = TRIM                                                                                LL
          8.7570        3.0148        6.9148                      0.60368E-01    1.3825      -0.11420                  LL
                                                                                                                       LL
    PERTURBATION IDENTIFICATION, COUNT =    1, LOOP = TRIM                                                             LL
          8.2570        3.0148        6.9148                      0.45722E-01    2.0701       0.87886                  LL
                                                                                                                       LL
    PERTURBATION IDENTIFICATION, COUNT =    2, LOOP = TRIM                                                             LL
          8.7570        2.0148        6.9148                      0.59877E-01   0.35883       0.28163E-01              LL
                                                                                                                       LL
    PERTURBATION IDENTIFICATION, COUNT =    3, LOOP = TRIM                                                             LL
          8.7570        3.0148        7.9148                      0.46821E-01    1.9948        1.6175                  LL
                                                                                                                       LL
                                                                                                                       LL
         PERTURBATION IDENTIFICATION                                                                                   LL
         DERIVATIVE MATRIX D                                                                                           LL
                         VARIABLE = COLL          LATCYC        LNGCYC                                                 LL
            QUANTITY = CT/S          0.29292E-01   0.49026E-03  -0.13547E-01                                           LL
            QUANTITY = BETAS         -1.3751        1.0237       0.61229                                               LL
            QUANTITY = BETAC         -1.9861      -0.14237        1.7317                                               LL
         GAIN MATRIX = (RELAXATION FACTOR) * D**-1                                                                     LL
                         QUANTITY = CT/S          BETAS         BETAC                                                  LL
            VARIABLE = COLL           14.965       0.86869E-02   0.11400                                               LL
            VARIABLE = LATCYC         9.3750       0.19165       0.55746E-02                                           LL
            VARIABLE = LNGCYC         17.935       0.25720E-01   0.24671                                               LL
                                                                                                                       LL
                                                                                                                       LL
    ITERATION, COUNT =    1, LOOP = TRIM                                                                               LL
          9.0518        2.9345        7.2596                      0.64271E-01    1.0811      -0.50174E-01              LL
                                                                                                                       LL
    ITERATION, COUNT =    2, LOOP = TRIM                                                                               LL
          9.2835        2.8750        7.5262                      0.67142E-01   0.86982       0.13545E-01              LL
                                                                                                                       LL
    ITERATION, COUNT =    3, LOOP = TRIM                                                                               LL
          9.4669        2.8288        7.7311                      0.69923E-01   0.66389       0.24469E-01              LL
```

```
                                                                                                                           LL
ITERATION, COUNT =    4, LOOP = TRIM                                                                                       LL
      9.6091        2.7959        7.8888                      0.71883E-01   0.50373       0.45100E-01                      LL
                                                                                                                           LL
ITERATION, COUNT =    5, LOOP = TRIM                                                                                       LL
      9.7211        2.7752        8.0103                      0.73591E-01   0.37796       0.38564E-01                      LL
                                                                                                                           LL
ITERATION, COUNT =    6, LOOP = TRIM                                                                                       LL
      9.8093        2.7626        8.1060                      0.73425E-01   0.37165       0.42224E-01                      LL
                                                                                                                           LL
ITERATION, COUNT =    7, LOOP = TRIM                                                                                       LL
      9.8997        2.7528        8.2039                      0.76389E-01   0.16084       0.23948E-01                      LL
                                                                                                                           LL
ITERATION, COUNT =    8, LOOP = TRIM                                                                                       LL
      9.9496        2.7557        8.2586                      0.76184E-01   0.15632       0.25904E-01                      LL
                                                                                                                           LL
ITERATION, COUNT =    9, LOOP = TRIM                                                                                       LL
      10.002        2.7614        8.3167                      0.77136E-01   0.13893       0.28563E-01                      LL
                                                                                                                           LL
ITERATION, COUNT =   10, LOOP = TRIM                                                                                       LL
      10.041        2.7614        8.3574                      0.78965E-01   0.15718E-01  -0.25938E-01                      LL
                                                                                                                            L
SUCCESSIVE SUBSTITUTION ITERATION, NO VARIABLES, NUMBER OF LEVELS =   2, LOOP = WAKE                                        L
ITERATION, COUNT =    1, FOR LEVEL   2 = NONUNIFORM INFLOW PRSC WAKE GEOM; LOOP = WAKE                                      L
```

```
****************
TRIM CONVERGENCE
****************

SUCCESSIVE SUBSTITUTION ITERATION, LOOP = WAKE
ITERATION  1, LEVEL   2 = NONUNIFORM INFLOW PRSC WAKE GEOM


    --------------------------------
    STATUS OF LOOP AND PART SOLUTIONS
    --------------------------------
                                                                                                                TOTAL NUMBER
                                                                                                               OF ITERATIONS
         LOOP = WAKE                              SUCCESSIVE SUBSTITUTION, NO VARIABLES                                    2
         LOOP = TRIM                              NEWTON RAPHSON, CONVERGED                                               27
         LOOP = CIRCULATION                       SUCCESSIVE SUBSTITUTION, CONVERGED                                     390
         LOOP = MOTION                            SUCCESSIVE SUBSTITUTION, CONVERGED                                     390
         LOOP = AIRFRAME                          NO SOLUTION

              PART = ROTOR 1 HUB                       HARMONIC SOLUTION, CONVERGED                                      830
              PART = ROTOR 1 BLADE 4                   HARMONIC SOLUTION, CONVERGED                                      472
              PART = ROTOR 1 HUB                       HARMONIC SOLUTION, CONVERGED
              PART = AIRFRAME                          HARMONIC SOLUTION, CONVERGED                                      421


    -------------------------------------
    CONVERGENCE OF LOOP AND PART SOLUTIONS
    -------------------------------------
         ------------------------------------------------------------------------------------------------------------------------
         LOOP = WAKE                              SUCCESSIVE SUBSTITUTION, NO VARIABLES
         NUMBER OF ITERATIONS =    1 (MAXIMUM    1), TOLERANCE =  0.000000E+00 FOR LEVEL   1 = UNIFORM INFLOW
         NUMBER OF ITERATIONS =    1 (MAXIMUM    1), TOLERANCE =  0.000000E+00 FOR LEVEL   2 = NONUNIFORM INFLOW PRSC WAKE GEOM
         ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
         ------------------------------------------------------------------------------------------------------------------------
         LOOP = TRIM                              NEWTON RAPHSON, CONVERGED
         NUMBER OF ITERATIONS =   10 (MAXIMUM   40), TOLERANCE =   5.00000
         ERROR = ABS(TRIMMED-TARGET); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)

              VARIABLE                                  QUANTITY
                         TRIMMED        INITIAL                     TRIMMED        TARGET         WEIGHT         ERROR RATIO
              COLL        10.0408        8.75701          CT/S       0.789654E-01   0.800000E-01   0.800000E-03   0.258662
              LATCYC      2.76145        3.01478          BETAS      0.157181E-01   0.000000E+00   0.200000E-01   0.157181
              LNGCYC      8.35741        6.91484          BETAC     -0.259375E-01   0.000000E+00   0.200000E-01   0.259375
         ------------------------------------------------------------------------------------------------------------------------
         LOOP = CIRCULATION                       SUCCESSIVE SUBSTITUTION, CONVERGED
         NUMBER OF ITERATIONS =   17 (MAXIMUM  200), TOLERANCE =   1.00000
         ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
         ------------------------------------------------------------------------------------------------------------------------
         LOOP = MOTION                            SUCCESSIVE SUBSTITUTION, CONVERGED
         NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000
         ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
         ------------------------------------------------------------------------------------------------------------------------


         ------------------------------------------------------------------------------------------------------------------------
         PART = ROTOR 1 HUB                       HARMONIC SOLUTION, CONVERGED
         NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000     FOR PERIOD   1 = ROTOR 1
```

```
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
------------------------------------------------------------------------------------------------------------------------
PART = ROTOR 1 BLADE 4                     HARMONIC SOLUTION, CONVERGED
NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000     FOR PERIOD   1 = ROTOR 1
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
------------------------------------------------------------------------------------------------------------------------
PART = ROTOR 1 HUB                         HARMONIC SOLUTION, CONVERGED
NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000     FOR PERIOD   1 = ROTOR 1
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
------------------------------------------------------------------------------------------------------------------------
PART = AIRFRAME                            HARMONIC SOLUTION, CONVERGED
NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000     FOR PERIOD   1 = ROTOR 1
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
------------------------------------------------------------------------------------------------------------------------
```

```
*************
TRIM SOLUTION
*************

----------------------
ROTORCRAFT PERFORMANCE
----------------------

   CONFIGURATION
       WIND TUNNEL
       ONE ROTOR

   OPERATING CONDITION
       WIND SPEED                               AIRFRAME ORIENTATION                        ROTOR SPEED
         VELOCITY (KNOTS)     =    150.00         YAW ANGLE (DEG)        =      0.00          TIP SPEED (FT/SEC)     =    673.16
         V/(OMEGA*R)          =    0.3761         PITCH ANGLE (DEG)      =     -7.00          TIP MACH NUMBER        =    0.6030
         MACH NUMBER          =    0.2268         ROLL ANGLE (DEG)       =      0.00          ROTATIONAL SPEED (RPM) =    260.00
         VELOCITY (FT/SEC)    =    253.17                                                     OMEGA (RAD/SEC)        =    27.227
         DYN PRESS (LB/FT**2) =     76.21                                                     REYNOLDS NO (/FT)      = 0.428E+07
         REYNOLDS NO (/FT)    = 0.161E+07

   CONTROL SETTINGS
       ROTORCRAFT
         PILOT CONTROL (DEG)                     PILOT CONTROL                    GUST (FT/SEC)
           COLLECTIVE          =     10.04         PEDAL (DEG) =      0.00          LONGITUDINAL      =      0.00
           LATERAL CYCLIC      =      2.76         THROTTLE    =      0.00          LATERAL           =      0.00
           LONGITUDINAL CYCLIC =      8.36       GOVERNOR (DEG)=      0.00          VERTICAL          =      0.00
       AIRFRAME
         AERODYNAMIC CONTROLS (DEG)              AERODYNAMIC CONTROLS (DEG)       ENGINE CONTROL
           FLAPERON            =      0.00         AILERON     =      0.00          THROTTLE          =      0.00
           ELEVATOR            =      0.00         RUDDER      =      0.00
       ROTOR 1
         PRIMARY CONTROLS (DEG)                  BLADE CONTROL (DEG)              HIGHER HARMONIC CONTROL (NUMBER OF HARMONICS)
           COLLECTIVE          =      0.00         PITCH       =      0.00          ROTATING FRAME    =   0      TE FLAP =   0
           LATERAL CYCLIC      =      0.00         TE FLAP 1   =      0.00          NONROTATING FRAME =   0
           LONGITUDINAL CYCLIC =      0.00         JETS        =   0.00000E+00      AUXILIARY FORCES  =   0
       AIRFRAME AND ROTOR CONTROLS ARE VALUES WITH ZERO PILOT AND GOVERNOR CONTROLS


   ROTORCRAFT POWER
       TOTAL                     CP=  0.00071896          CP/S=  0.0079007          P=    1820.972  HP  =   1001534.75  LB-FT/SEC
       CLIMB + PARASITE      CPC+CPP=  0.00033592    CPC/S+CPP/S=  0.0036914      PC+PP=     850.818  HP  =    467949.75  LB-FT/SEC
       PROFILE + INDUCED     CPO+CPI=  0.00038304    CPO/S+CPI/S=  0.0042092      PO+PI=     970.155  HP  =    533585.00  LB-FT/SEC

       TOTAL                     CP=  0.00071896          CP/S=  0.0079007          P=    1820.972  HP  =   1001534.75  LB-FT/SEC
       INDUCED                CPIND=  0.00019416       CPIND/S=  0.0021336       PIND=     491.758  HP  =    270467.09  LB-FT/SEC
       INTERFERENCE           CPINT=  0.00000000       CPINT/S=  0.0000000       PINT=       0.000  HP  =         0.00  LB-FT/SEC
       PROFILE                  CPO=  0.00018888         CPO/S=  0.0020756         PO=     478.396  HP  =    263117.91  LB-FT/SEC
       PARASITE                 CPP=  0.00033592         CPP/S=  0.0036914         PP=     850.818  HP  =    467949.75  LB-FT/SEC
       CLIMB                    CPC=  0.00000000         CPC/S=  0.0000000         PC=       0.000  HP  =         0.00  LB-FT/SEC

       PROFILE (SECTION)        CPO=  0.00018999         CPO/S=  0.0020878         PO=     481.211  HP  =    264666.03  LB-FT/SEC
       MINIMUM INDUCED          CPM=  0.00006841         CPM/S=  0.0007518         PM=     173.274  HP  =     95300.67  LB-FT/SEC

       TOTAL                     CP=  0.00071896          CP/S=  0.0079007          P=    1820.972  HP  =   1001534.75  LB-FT/SEC
       IDEAL                CPIDEAL=  0.00040433     CPIDEAL/S=  0.0044432     PIDEAL=    1024.092  HP  =    563250.44  LB-FT/SEC
```

```
    NONIDEAL                 CPN=  0.00031463         CPN/S=  0.0034574        PN=     796.881  HP  =    438284.31  LB-FT/SEC

PERFORMANCE INDICES
    EQUIVALENT DRAG            D    =   (PI+PO)/VREF       =     2107.6030  LB
    DRAG AREA                  D/Q  =   D/.5*RHO*VREF**2   =       27.6552  FT**2
    ROTOR LIFT-TO-DRAG RATIO   L/DR =   WEIGHT/D           =     0.0000
    FIGURE OF MERIT            M    =   PM/(PO+PI)         =     0.1786
    FIGURE OF MERIT            M    =   PIDEAL/P           =     0.5624
    AIRCRAFT LIFT-TO-DRAG      L/D  =   WEIGHT*VREF/P      =     0.0000

          ROTORCRAFT POWER = SUM OF POWER FOR ALL ROTORS; TOTAL POWER = SHAFT AND REACTION JET POWER ABSORBED BY ROTORS
          INDUCED POWER FROM SELF-INDUCED VELOCITY OF ROTORS; INTERFERENCE POWER FROM ALL OTHER SOURCES; PI = PIND + PINT
          CLIMB POWER PC = VCLIMB*WEIGHT; PARASITE POWER PP = (PP+PC) - PC
          PROFILE + INDUCED POWER PO+PI = P - (PP+PC); PROFILE POWER PO = (PO+PI) - PIND - PINT
          PROFILE POWER (SECTION) FROM INTEGRATED SECTION DRAG; MINIMUM INDUCED POWER FROM MOMENTUM THEORY INDUCED VELOCITY
          IDEAL POWER PIDEAL = MINIMUM INDUCED + PARASITE + CLIMB
          NONIDEAL POWER PN = P - PIDEAL = PROFILE + EXCESS INDUCED + INTERFERENCE

          -----------------------------------------------------------------------------------------------------------------------
          POWER COEFFICIENT CP = P / RHO*A*VTIP**3
              AIR DENSITY RHO = 0.002378 SLUG/FT**3, REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTOR AREA A = PI*R**2
              REFERENCE ROTATIONAL SPEED OMEGA =  260.000 RPM, TIP SPEED VTIP = OMEGA*R, SOLIDITY RATIO S =  0.09100
          GROSS WEIGHT =      0.000 LB, CLIMB SPEED =     0.000 FT/SEC =      0.00 FT/MIN
          REFERENCE SPEED VREF =   253.171 FT/SEC
          -----------------------------------------------------------------------------------------------------------------------

LOADS APPLIED TO AIRFRAME COMPONENT
    FORCE (LB) AND MOMENT (FT-LB); IN AIRFRAME AXES, MOMENTS ABOUT ORIGIN OF AIRFRAME AXES
    APPLIED LOAD = F FOR AUXILIARY FORCE, T FOR TAIL BOOM, L FOR SLUNG LOAD; A FOR NONROTATING SWASHPLATE ACTUATOR
    AIRFRAME COMPONENT MASS = 0.00000E+00 SLUG = 0.00000E+00 LB; (WEIGHT+ACCELERATION)/(AIRFRAME WEIGHT) =   1.000 G
                   LOCATION   LOAD       FORCE X         FORCE Y         FORCE Z          MOMENT X       MOMENT Y       MOMENT Z
    -----------------------------------------------------------------------------------------------------------------------
    ROTOR 1 HUB           1              54.603        221.781     -14875.857           579.97          393.83      -36784.44
    -----------------------------------------------------------------------------------------------------------------------
    TOTAL                                54.603        221.781     -14875.857           579.97          393.83      -36784.44

AVAILABLE TRIM QUANTITIES
    ROTORCRAFT PERFORMANCE, TRIM SENSOR
      TOTAL POWER CP/S =  0.0079007
    ROTOR 1 PERFORMANCE, TRIM SENSOR
      SHAFT AXES                 TIP-PATH PLANE AXES            WIND AXES                   TIP-PATH PLANE TILT (DEG)
        THRUST CT/S     =  0.078965    THRUST CT/S     =  0.078966    LIFT CL/S =  0.078362      LONGITUDINAL, BETAC =   -0.026
        DRAG FORCE CH/S = -0.000193    DRAG FORCE CH/S = -0.000229    DRAG CX/S = -0.009815      LATERAL, BETAS      =    0.016
        SIDE FORCE CY/S =  0.001204    SIDE FORCE CY/S =    0.0012
      POWER CP/S =  0.0079007

          -----------------------------------------------------------------------------------------------------------------------
          ROTORCRAFT COEFFICIENT/SOLIDITY FORM:  POWER CP/S = P / RHO*A*VTIP**3*S (BASED ON ROTOR 1 PARAMETERS)
              AIR DENSITY RHO = 0.002378 SLUG/FT**3, REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTOR AREA A = PI*R**2
              REFERENCE ROTATIONAL SPEED OMEGA =  260.000 RPM, TIP SPEED VTIP = OMEGA*R, SOLIDITY RATIO S =  0.09100
          -----------------------------------------------------------------------------------------------------------------------
          ROTOR 1 COEFFICIENT/SOLIDITY FORM:  FORCE CF/S = F / RHO*A*VTIP**2*S; POWER CP/S = P / RHO*A*VTIP**3*S
              AIR DENSITY RHO = 0.002378 SLUG/FT**3, REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTOR AREA A = PI*R**2
```

REFERENCE ROTATIONAL SPEED OMEGA = 260.000 RPM, TIP SPEED VTIP = OMEGA*R, SOLIDITY RATIO S = 0.09100

```
*************
TRIM SOLUTION
*************

-------------------
ROTOR 1 PERFORMANCE
-------------------

   CONFIGURATION
       ONE ROTOR

   OPERATING CONDITION
                               ADVANCE RATIO            ANGLES (DEG)     TIP-PATH PLANE TILT (DEG)     CYCLIC CONTROL (DEG)
                           MUX      MUY      MUZ        ALPHA    PSI        BC (LONG)  BS (LAT)       T1S (LONG)  T1C (LAT)
       HUB PLANE         0.3733   0.0000   0.0458       -7.00    0.00         -0.03      0.02            -8.36       2.76
       TIP-PATH PLANE    0.3733   0.0000   0.0457       -6.97    0.00          0.00      0.00            -8.38       2.75
       CONTROL PLANE     0.3627  -0.0035   0.0994      -15.33   -0.55         -8.38     -2.75             0.00       0.00

       ROTOR VELOCITY                BLADE TIP SPEED                            ANGLES (DEG)
         ADVANCE RATIO = 0.3761        HOVER TIP MACH NUMBER       = 0.6030        CONING ANGLE       =   4.215
         MACH NUMBER   = 0.2268        MAXIMUM SECTION MACH NUMBER = 0.8285        COLLECTIVE CONTROL =  10.041


             VELOCITY OF AIR RELATIVE ROTOR MU = (-MUX,MUY,MUZ); MUX + AFT, MUY + FROM RIGHT, MUZ + DOWN THROUGH ROTOR DISK
                 ALPHA = ANGLE OF ATTACK OF ROTOR PLANE RELATIVE VELOCITY, + AFT; ALFHP = ALFCP - T1S = ALFTPP + BC
                 PSI = YAW ANGLE OF VELOCITY IN ROTOR PLANE, + FROM FORWARD/RIGHT QUADRANT
             LONGITUDINAL TIP-PATH PLANE TILT BC + FORWARD, LATERAL TILT BS + TOWARD RETREATING SIDE
             TIP-PATH PLANE TILT (BC,BS) AND CYCLIC CONTROL (T1C,T1S) ARE COEFFICIENTS OF POSITIVE FOURIER SERIES IN BLADE AZIMUTH
                 (BC + T1S)HP = BCCP = T1STPP, (BS - T1C)HP = BSCP = - T1CTPP

             -------------------------------------------------------------------------------------------------------------------
             ADVANCE RATIO = VELOCITY / VTIP
                 REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTATIONAL SPEED OMEGA =  260.000 RPM; TIP SPEED VTIP = OMEGA*R
                 SOUND SPEED = 1116.45 FT/SEC
             -------------------------------------------------------------------------------------------------------------------

             HUB PLANE (NONROTATING SHAFT AXES):  Z IS SHAFT AXIS (POSITIVE THRUST DIRECTION)
                 X-AXIS DOWNSTREAM; AZIMUTH ANGLE MEASURED FROM X-AXIS, IN DIRECTION OF ROTOR ROTATION
                 ROTOR 1 ROTATES ABOUT Z-AXIS CLOCKWISE; Y-AXIS TOWARDS RETREATING SIDE
             TIP-PATH PLANE TILTED RELATIVE HUB PLANE, FORWARD FOR + BC AND TOWARD RETREATING SIDE FOR + BS
             CONTROL PLANE TILTED RELATIVE HUB PLANE, AFT FOR + T1S AND TOWARD RETREATING SIDE FOR + T1C


   ROTOR FORCES AND MOMENTS
       SHAFT AXES
         THRUST            CT =  0.00718585        CT/S =  0.0789653        T =       14870.263  LB
         DRAG FORCE        CH = -0.00001759        CH/S = -0.0001933        H =         -36.394  LB           ANGLE =  -0.14 DEG
         SIDE FORCE        CY =  0.00010953        CY/S =  0.0012037        Y =         226.666  LB           ANGLE =   0.87 DEG
         ROLL MOMENT       CMX= -0.00001341        CMX/S= -0.0001474        MX=        -686.146  FT-LB        OFFSET =  -0.002 R
         PITCH MOMENT      CMY=  0.00000985        CMY/S=  0.0001082        MY=         503.994  FT-LB        OFFSET =   0.001 R
         YAW MOMENT        CMZ=  0.00071986        CMZ/S=  0.0079105        MZ=       36830.453  FT-LB
       TIP-PATH PLANE AXES
         THRUST            CT =  0.00718587        CT/S =  0.0789656        T =       14870.307  LB
         DRAG FORCE        CH = -0.00002084        CH/S = -0.0002290        H =         -43.126  LB           ANGLE =  -0.17 DEG
         SIDE FORCE        CY =  0.00010756        CY/S =  0.0011820        Y =         222.586  LB           ANGLE =   0.86 DEG
         ROLL MOMENT       CMX= -0.00001374        CMX/S= -0.0001510        MX=        -702.819  FT-LB        OFFSET =  -0.002 R
```

```
  PITCH MOMENT       CMY=  0.00000965       CMY/S=  0.0001061       MY=        493.890  FT-LB        OFFSET =   0.001 R
  YAW MOMENT         CMZ=  0.00071986       CMZ/S=  0.0079105       MZ=      36830.277  FT-LB
WIND AXES
  LIFT               CL =  0.00713098       CL/S =  0.0783625       L =      14756.729  LB
  DRAG               CX = -0.00089319       CX/S = -0.0098153       X =      -1848.351  LB           ANGLE =  -7.14 DEG
TORQUE
  SHAFT TORQUE       CQ = -0.00071896       CQ/S = -0.0079007       Q =     -36784.430  FT-LB

      AERODYNAMIC ROTOR FORCE = (H,Y,T) AND MOMENT = (MX,MY,MZ) ACTING ON HUB
      FORCE ANGLES = PITCH AND ROLL ANGLES OF TOTAL FORCE VECTOR RELATIVE ROTOR PLANE
      MOMENT OFFSETS = PITCH AND ROLL MOMENTS DIVIDED BY THRUST TIMES RADIUS
      WIND AXES:  DRAG IS COMPONENT OF TOTAL FORCE IN VELOCITY DIRECTION (NEGATIVE FOR PROPULSIVE FORCE)
      WIND AXES:  LIFT IS COMPONENT OF TOTAL FORCE PERPENDICULAR TO VELOCITY (ALWAYS POSITIVE)
      SWASHPLATE TORQUE = SHAFT-AXIS COMPONENT OF MOMENT AT HUB NODE

      ----------------------------------------------------------------------------------------------------------------------
      ROTOR COEFFICIENT FORM:  FORCE CF = F / RHO*A*VTIP**2; MOMENT CM = M / RHO*A*VTIP**2*R
          AIR DENSITY RHO = 0.002378 SLUG/FT**3, REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTOR AREA A = PI*R**2
          REFERENCE ROTATIONAL SPEED OMEGA =  260.000 RPM, TIP SPEED VTIP = OMEGA*R, SOLIDITY RATIO S =  0.09100
      ----------------------------------------------------------------------------------------------------------------------


ROTOR POWER
    TOTAL                      CP=  0.00071896          CP/S=  0.0079007         P=    1820.972  HP  =   1001534.75  LB-FT/SEC
    CLIMB + PARASITE      CPC+CPP=  0.00033592    CPC/S+CPP/S=  0.0036914     PC+PP=     850.818  HP  =    467949.75  LB-FT/SEC
    PROFILE + INDUCED     CPO+CPI=  0.00038304    CPO/S+CPI/S=  0.0042092     PO+PI=     970.155  HP  =    533585.00  LB-FT/SEC

    INDUCED                 CPIND=  0.00019416        CPIND/S=  0.0021336      PIND=     491.758  HP  =    270467.09  LB-FT/SEC
    INTERFERENCE            CPINT=  0.00000000        CPINT/S=  0.0000000      PINT=       0.000  HP  =         0.00  LB-FT/SEC
    PROFILE                   CPO=  0.00018888          CPO/S=  0.0020756        PO=     478.396  HP  =    263117.91  LB-FT/SEC
    PROFILE (SECTION)         CPO=  0.00018999          CPO/S=  0.0020878        PO=     481.211  HP  =    264666.03  LB-FT/SEC

    MINIMUM INDUCED           CPM=  0.00006841          CPM/S=  0.0007518        PM=     173.274  HP  =     95300.67  LB-FT/SEC
    IDEAL                 CPIDEAL=  0.00040433      CPIDEAL/S=  0.0044432    PIDEAL=    1024.092  HP  =    563250.44  LB-FT/SEC
    NONIDEAL                  CPN=  0.00031463          CPN/S=  0.0034574        PN=     796.881  HP  =    438284.31  LB-FT/SEC

      TOTAL POWER = SHAFT, SWASHPLATE, AND REACTION JET POWER ABSORBED BY ROTOR
      SWASHPLATE POWER PS = MEAN POWER AT JOINT (TILTED BY CYCLIC)
      CLIMB + PARASITE POWER PC+PP = - VELOCITY*DRAG; PROFILE + INDUCED POWER PO+PI = P - (PC+PP)
      INDUCED POWER FROM SELF-INDUCED VELOCITY OF ROTOR; INTERFERENCE POWER FROM ALL OTHER SOURCES; PI = PIND + PINT
      PROFILE POWER PO = (PO+PI) - PIND - PINT
      PROFILE POWER (SECTION) FROM INTEGRATED SECTION DRAG; MINIMUM INDUCED POWER FROM MOMENTUM THEORY INDUCED VELOCITY
      IDEAL POWER PIDEAL = MINIMUM INDUCED + INTERFERENCE + PARASITE + CLIMB
      NONIDEAL POWER PN = P - PIDEAL = PROFILE + EXCESS INDUCED

      ----------------------------------------------------------------------------------------------------------------------
      POWER COEFFICIENT CP = P / RHO*A*VTIP**3
          AIR DENSITY RHO = 0.002378 SLUG/FT**3, REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTOR AREA A = PI*R**2
          REFERENCE ROTATIONAL SPEED OMEGA =  260.000 RPM, TIP SPEED VTIP = OMEGA*R, SOLIDITY RATIO S =  0.09100
      ----------------------------------------------------------------------------------------------------------------------


PERFORMANCE INDICES
                  MEAN INFLOW RATIO    EQUIVALENT INFLOW    EXCESS INDUCED LOSSES              MEAN INFLOW RATIO
```

	MAGNITUDE LAMBDA	CP/CT	KAPPA = P/PM	LAMBDAX	LAMBDAY	LAMBDAZ
INDUCED	0.017977	0.027019	2.8380	0.004519	0.000974	-0.017372
INTERFERENCE	0.000000	0.000000	0.0000	0.000000	0.000000	0.000000
MINIMUM INDUCED	0.009520	0.009520	1.0000			

EQUIVALENT SECTION DRAG COEFFICIENT, PROFILE POWER CDO = (8*CPO/S)/F(MU) = 0.00991
EQUIVALENT SECTION DRAG COEFFICIENT, NONIDEAL POWER CDN = (8*CPN/S)/F(MU) = 0.01650

FIGURE OF MERIT, M = (PM+PINT)/(PO+PI) =	0.1786	POWER COEFFICIENT, CP = -P/QAV =	-0.0270
FIGURE OF MERIT, M = PIDEAL/P =	0.5624	WIND TURBINE EFFICIENCY, M = (27/16)CP =	-0.0456
PROPULSIVE EFFICIENCY, ETA = (PC+PP)/P =	0.4672	WING INDUCED EFFICIENCY, E =	1.3928
ROTOR LIFT-TO-DRAG RATIO, L/D =	7.0017	WING LIFT-TO-DRAG RATIO, L/X =	-7.9837

		FORCE (LB)	FORCE/Q (FT**2)	COEFFICIENT (F/QA)	WING COEFFICIENT (F/QS)
ROTOR EQUIV DRAG	D = P/V+X =	2107.6030	D/Q = 27.6552	CD = 0.01440	CDW = 0.15825
PROFILE DRAG	DO = PO/V =	1039.2872	DO/Q = 13.6372	CDO = 0.00710	CDWO = 0.07804
INDUCED DRAG	DIND = PIND/V =	1068.3158	DIND/Q = 14.0181	CDIND = 0.00730	CDWIND = 0.08022
INTERFERENCE DRAG	DINT = PINT/V =	0.0000	DINT/Q = 0.0000	CDINT = 0.00000	CDWINT = 0.00000
ROTOR DRAG	X =	-1848.3508	X/Q = -24.2534	CX = -0.01263	CXW = -0.13879
ROTOR LIFT	L =	14756.7285	L/Q = 193.6326	CL = 0.10083	CLW = 1.10803

```
MEAN INFLOW RATIO:  VELOCITY MAGNITUDE AND COMPONENTS IN SHAFT AXES (AVERAGE OVER ROTOR DISK)
FIGURE OF MERIT IS MEASURE OF ROTOR EFFICIENCY APPROPRIATE FOR AXIAL FLOW
    BOTH DEFINITIONS INCLUDE CONVENTIONAL HOVER FIGURE OF MERIT, M = T*VIDEAL/P = T*SQRT(T/2*RHO*A)/P
    M = (PM+PINT)/(PO+PI) = T*(VIDEAL+VINT)/(PO+PI)
    M = PIDEAL/P = 1 - PN/P = T*(V+VIDEAL)/P FOR AXIAL FLOW  (NOT VALID FOR ZERO POWER)
    SECOND DEFINITION NEARLY PROPULSIVE EFFICIENCY ETA = TV/P FOR HIGH INFLOW
PROPULSIVE EFFICIENCY IS MEASURE APPROPRIATE FOR HIGH SPEED PROPULSION
    ETA = -X*V/P = (PC+PP)/P  (NOT VALID FOR ZERO VELOCITY OR ZERO POWER)
ROTOR LIFT-TO-DRAG RATIO IS MEASURE OF ROTOR LIFTING EFFICIENCY, APPROPRIATE FOR EDGEWISE FLOW
    L/D = L*V/(PI+PO)  (NOT USEFUL FOR ZERO VELOCITY); ROTOR EQUIVALENT DRAG D = P/V+X = (PI+PO)/V
WING LIFT-TO-DRAG RATIO AND INDUCED EFFICIENCY E ARE MEASURES APPROPRIATE FOR FIXED WING
    DI/Q = (L/Q)**2/(E*PI*B**2)

----------------------------------------------------------------------------------------------------------------------
INFLOW RATIO, INFLOW = VELOCITY / VTIP
    REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTATIONAL SPEED OMEGA =  260.000 RPM; TIP SPEED VTIP = OMEGA*R
COEFFICIENT CF = F/QA, WING COEFFICIENT CFW = F/QS
    ROTOR AREA A = PI*R**2; BLADE AREA S = A*SIGMA; WING SPAN B = R; DYNAMIC PRESSURE Q=.5*RHO*V**2
    RADIUS R =   24.7240 FT, SOLIDITY RATIO SIGMA =  0.09100, AIR DENSITY RHO = 0.002378 SLUG/FT**3
----------------------------------------------------------------------------------------------------------------------
```

BLADE SECTION ANGLE OF ATTACK (DEG)

RADIAL STATION =	0.560	0.615	0.665	0.710	0.750	0.785	0.815	0.845	0.870	0.890	0.910	0.930	0.950	0.970	0.990
AZIMUTH (DEG)	ALPHA														
PSI = 0.0	8.1	7.9	8.0	8.1	8.1	8.1	7.9	7.2	6.3	5.9	6.2	6.7	6.6	6.9	6.0
PSI = 15.0	6.1	6.6	6.8	6.8	6.8	6.7	6.5	5.9	5.1	4.8	5.1	5.4	5.4	5.7	5.0
PSI = 30.0	4.9	5.0	5.2	5.3	5.3	5.3	5.1	4.7	4.1	3.8	3.9	4.2	4.2	4.4	3.9
PSI = 45.0	3.8	3.9	4.0	4.0	4.0	3.9	3.8	3.5	3.1	2.8	2.9	3.1	3.0	3.2	2.8
PSI = 60.0	2.6	2.7	2.8	2.8	2.8	2.7	2.6	2.4	2.1	1.9	1.9	2.0	2.0	2.1	1.8
PSI = 75.0	1.8	1.8	1.8	1.8	1.7	1.6	1.6	1.4	1.2	1.1	1.1	1.1	1.1	1.1	1.0
PSI = 90.0	1.3	1.2	1.1	1.0	0.9	0.8	0.7	0.6	0.5	0.4	0.4	0.4	0.3	0.3	0.2
PSI = 105.0	1.1	1.0	0.8	0.6	0.5	0.3	0.2	0.1	0.0	-0.1	-0.1	-0.1	-0.2	-0.2	-0.2
PSI = 120.0	1.3	1.0	0.7	0.4	0.3	0.2	0.1	0.0	-0.1	-0.1	-0.1	-0.2	-0.2	-0.3	-0.3
PSI = 135.0	1.4	1.1	1.0	0.9	0.7	0.6	0.5	0.4	0.3	0.1	0.1	0.0	0.0	-0.1	-0.1

PSI = 150.0	2.5	2.4	2.3	2.2	2.0	1.8	1.7	1.6	1.4	1.0	0.8	0.7	0.6	0.5	0.4
PSI = 165.0	4.4	4.3	4.2	3.9	3.7	3.4	3.3	3.2	3.0	2.3	1.8	1.6	1.4	1.3	1.0
PSI = 180.0	6.7	6.5	6.2	5.8	5.5	5.2	5.0	4.9	4.7	3.7	3.0	2.6	2.3	2.1	1.7
PSI = 195.0	8.8	8.4	8.0	7.6	7.2	6.8	6.5	6.4	6.1	4.9	3.9	3.5	3.1	2.8	2.2
PSI = 210.0	10.5	9.9	9.4	8.8	8.4	7.9	7.6	7.4	7.1	5.7	4.6	4.1	3.6	3.2	2.5
PSI = 225.0	11.9	11.1	10.4	9.8	9.3	8.8	8.5	8.3	7.8	6.4	5.2	4.7	4.2	3.7	2.9
PSI = 240.0	12.8	11.9	11.2	10.6	10.1	9.6	9.2	9.0	8.5	7.0	6.0	5.4	4.9	4.4	3.5
PSI = 255.0	12.6	12.0	11.5	11.0	10.6	10.2	9.9	9.7	9.0	7.6	6.7	6.3	5.7	5.3	4.3
PSI = 270.0	11.7	11.4	11.3	11.1	10.8	10.6	10.4	10.1	9.3	8.1	7.5	7.2	6.7	6.3	5.2
PSI = 285.0	10.5	10.6	10.8	10.9	10.9	10.9	10.9	10.4	9.4	8.5	8.2	8.1	7.6	7.4	6.2
PSI = 300.0	9.4	10.0	10.4	10.7	10.9	11.0	11.0	10.4	9.4	8.6	8.6	8.8	8.4	8.3	7.0
PSI = 315.0	8.5	9.2	9.7	10.0	10.2	10.2	10.1	9.4	8.3	7.8	8.0	8.4	8.1	8.3	7.1
PSI = 330.0	7.4	8.5	9.1	9.4	9.6	9.7	9.5	8.8	7.8	7.3	7.6	8.0	7.8	8.0	7.0
PSI = 345.0	6.8	7.9	8.6	8.9	9.1	9.2	9.1	8.3	7.2	6.8	7.2	7.7	7.5	7.7	6.8
PSI = 360.0	8.1	7.9	8.0	8.1	8.1	8.1	7.9	7.2	6.3	5.9	6.2	6.7	6.6	6.9	6.0

RADIAL STATION =	0.250	0.340	0.420	0.495
AZIMUTH (DEG)	ALPHA			
PSI = 0.0	6.5	-5.2	6.1	8.5
PSI = 15.0	0.9	6.1	3.7	4.9
PSI = 30.0	2.9	3.0	4.5	4.7
PSI = 45.0	1.9	3.0	3.3	3.6
PSI = 60.0	1.7	1.6	2.1	2.4
PSI = 75.0	0.9	1.2	1.5	1.7
PSI = 90.0	1.2	1.3	1.4	1.3
PSI = 105.0	1.2	1.4	1.3	1.2
PSI = 120.0	1.5	1.8	1.8	1.6
PSI = 135.0	2.2	2.4	2.1	1.7
PSI = 150.0	3.1	3.2	3.0	2.7
PSI = 165.0	3.8	4.2	4.3	4.4
PSI = 180.0	4.8	5.7	6.2	6.6
PSI = 195.0	6.4	7.9	8.8	9.0
PSI = 210.0	9.9	10.9	11.5	11.1
PSI = 225.0	26.7	17.3	14.3	12.9
PSI = 240.0	90.1	26.8	16.9	14.0
PSI = 255.0	-155.8	41.8	19.0	14.1
PSI = 270.0	-123.1	6.5	15.9	12.7
PSI = 285.0	-130.8	-89.7	16.7	10.7
PSI = 300.0	-134.6	-28.0	5.9	8.6
PSI = 315.0	-147.7	-18.2	-0.7	7.4
PSI = 330.0	-135.5	4.5	-4.7	3.8
PSI = 345.0	-50.4	16.5	10.7	5.8
PSI = 360.0	6.5	-5.2	6.1	8.5

BOUND CIRCULATION PEAKS (ENTIRE BLADE)

	BOUND CIRCULATION			RADIAL STATION			WAKE GEOMETRY
AZIMUTH (DEG)	GMAX	GO	GI	RMAX	RO	RI	RGI
PSI = 0.0	0.02614	0.02614	-0.00595	0.87000	0.87000	0.34000	0.25000
PSI = 15.0	0.02496	0.02496	0.00000	0.87000	0.87000	-0.60000	0.25000
PSI = 30.0	0.02294	0.02294	0.00000	0.87000	0.87000	-0.60000	-0.60000
PSI = 45.0	0.01916	0.01916	0.00000	0.87000	0.87000	-0.60000	-0.60000
PSI = 60.0	0.01445	0.01445	0.00000	0.87000	0.87000	-0.60000	-0.60000
PSI = 75.0	0.00913	0.00913	0.00000	0.81500	0.81500	-0.60000	-0.60000
PSI = 90.0	0.00540	0.00540	0.00000	0.61500	0.61500	-0.60000	0.56000
PSI = 105.0	0.00501	-0.00190	0.00501	0.56000	0.97000	0.56000	0.25000
PSI = 120.0	0.00678	-0.00190	0.00678	0.42000	0.97000	0.42000	0.25000
PSI = 135.0	0.00782	0.00782	0.00000	0.42000	0.42000	-0.60000	0.34000
PSI = 150.0	0.01200	0.01200	0.00000	0.71000	0.71000	-0.60000	-0.60000
PSI = 165.0	0.01810	0.01810	0.00000	0.75000	0.75000	-0.60000	-0.60000
PSI = 180.0	0.02258	0.02258	0.00000	0.78500	0.78500	-0.60000	-0.60000
PSI = 195.0	0.02401	0.02401	0.00000	0.89000	0.89000	-0.60000	-0.60000
PSI = 210.0	0.02321	0.02321	0.00000	0.89000	0.89000	-0.60000	0.25000
PSI = 225.0	0.02237	0.02237	0.00000	0.91000	0.91000	-0.60000	0.25000
PSI = 240.0	0.02187	0.02187	0.00000	0.91000	0.91000	-0.60000	0.34000
PSI = 255.0	0.02179	0.02179	0.00000	0.91000	0.91000	-0.60000	0.34000
PSI = 270.0	0.02188	0.02188	0.00000	0.91000	0.91000	-0.60000	0.49500
PSI = 285.0	0.02240	0.02240	0.00000	0.89000	0.89000	-0.60000	0.49500
PSI = 300.0	0.02318	0.02318	-0.00682	0.89000	0.89000	0.34000	0.42000
PSI = 315.0	0.02236	0.02236	-0.00591	0.89000	0.89000	0.34000	0.34000
PSI = 330.0	0.02376	0.02376	-0.00342	0.89000	0.89000	0.42000	0.25000
PSI = 345.0	0.02578	0.02578	-0.00919	0.87000	0.87000	0.25000	0.25000
PSI = 360.0	0.02614	0.02614	-0.00595	0.87000	0.87000	0.34000	0.25000

GO = OUTBOARD PEAK, GI = INBOARD PEAK, GMAX = PEAK WITH MAXIMUM MAGNITUDE
CORRESPONDING RADIAL STATIONS OF PEAKS ARE RO,RI,RMAX; VALUE OUTSIDE BLADE IF PEAK NOT FOUND
RGI = SPAN STATION OF INBOARD CIRCULATION PEAK
FOR DUAL PEAK WAKE MODEL: CONVERGENCE OF WAKE GEOMETRY ITERATION REQUIRES RGI = RI

GAMMA = BOUND CIRCULATION / OMEGA*R**2
REFERENCE ROTOR RADIUS R = 24.7240 FT, ROTATIONAL SPEED OMEGA = 260.000 RPM; TIP SPEED VTIP = OMEGA*R

UNIFORM INFLOW

ROTOR VELOCITY	ROTOR LOADS	INDUCED VELOCITY	INFLOW FACTOR	TIP PATH PLANE NORMAL
MUX = 0.3733	CT = 0.0071050	LAMBDAI = 0.01641		UX = 0.1217
MUY = -0.0001	CMY = 0.0000143	LAMBDAX = 0.02282	KX = 1.39483	UY = 0.0016
MUZ = 0.0458	CMX = -0.0000142	LAMBDAY = 0.01218	KY = 0.74687	UZ = -0.9926

COSINE ANGLE BETWEEN WAKE AND GROUND NORMAL, COSE = 0.0000 HEIGHT ROTOR ABOVE GROUND LEVEL, ZG/R = 0.0000
WAKE VELOCITY RATIO, W(FAR)/V(DISK) = 2.0000 WAKE AREA RATIO, A(DISK)/A(FAR) = 1.0080
INFLOW STATE
MUZ/LH = 0.768 LAMBDAI/LH = 0.275 (MUZ+ LAMBDAI)/LH = 1.044 LH = 0.05960
MU /LH = 6.263 (MUZ+2*LAMBDAI)/LH = 1.319

ROTOR VELOCITY AND ROTOR LOADS IN TIP-PATH PLANE AXES; TIP PATH PLANE NORMAL IN INERTIAL AXES
ROTOR LOADS: THRUST COEFFICIENT CT, PITCH MOMENT COEFFICIENT CMY, ROLL MOMENT COEFFICIENT CMX

```
INDUCED VELOCITY:  MEAN LAMBDAI, LONGITUDINAL GRADIENT LAMBDAX=KX*LAMBDAI, LATERAL GRADIENT LAMBDAY=KY*LAMBDAI
INFLOW FACTOR:  LONGITUDINAL KX, LATERAL KY
INFLOW STATE:  LH = SQRT(CT/2), MU = SQRT(MUX**2+MUY**2)

---------------------------------------------------------------------------------------------------------------------
ADVANCE RATIO MU, INFLOW RATIO LAMBDA = VELOCITY / VTIP
    REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTATIONAL SPEED OMEGA =  260.000 RPM; TIP SPEED VTIP = OMEGA*R
---------------------------------------------------------------------------------------------------------------------


RIGID WAKE GEOMETRY
    WAKE CONVECTION VELOCITY, TIP-PATH PLANE AXES      DMUX =  0.0000    DMUY =  0.0000    DMUZ = -0.0171
    WAKE CONVECTION VELOCITY, INERTIAL AXES            DMUX = -0.0021    DMUY =  0.0000    DMUZ =  0.0170

---------------------------------------------------------------------------------------------------------------------
CONVECTION ADVANCE RATIO (WITHOUT WIND) DMU = VELOCITY / VTIP
    REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTATIONAL SPEED OMEGA =  260.000 RPM; TIP SPEED VTIP = OMEGA*R
---------------------------------------------------------------------------------------------------------------------
```

```
*************
TRIM SOLUTION
*************

---------------
ANALYSIS FRAMES
---------------

         CONVENTIONS
             ------------------------------------------------------------------------------------------------------------------------
             FRAME = INERTIAL             GRAVITY IN +Z DIRECTION (+Z AXIS OF INERTIAL FRAME DOWNWARD)
             ------------------------------------------------------------------------------------------------------------------------
             FRAME = AIRFRAME             AXES OF TEST MODULE (CONSTRAINED)
                                          X-AXIS FORWARD (WIND POSITIVE FROM X DIRECTION); ORIGIN AT CENTER OF TEST MODULE
             ------------------------------------------------------------------------------------------------------------------------
             FRAME = ROTOR N              NONROTATING AXES OF ROTOR
                                          Z IS SHAFT AXIS (POSITIVE THRUST DIRECTION); ORIGIN AT CENTER OF ROTATION (HUB)
                                          X-AXIS DOWNSTREAM; AZIMUTH ANGLE MEASURED FROM X-AXIS, IN DIRECTION OF ROTOR ROTATION
                                          ROTOR 1 ROTATES ABOUT Z-AXIS CLOCKWISE
             ------------------------------------------------------------------------------------------------------------------------
             FRAME = ROTOR N ROTATING     ROTATING AXES OF ROTOR
                                          Z IS SHAFT AXIS (POSITIVE THRUST DIRECTION); ORIGIN AT CENTER OF ROTATION (HUB)
                                          X-AXIS DOWNSTREAM FOR ZERO AZIMUTH
             ------------------------------------------------------------------------------------------------------------------------
             FRAME = ROTOR N BLADE M      ROTATING AXES OF BLADE
                                          Z IS SHAFT AXIS (POSITIVE THRUST DIRECTION); ORIGIN AT CENTER OF ROTATION (HUB)
                                          Y-AXIS SPANWISE (POSITIVE OUTWARD); X-AXIS CHORDWISE
                                          ROTOR 1 X-AXIS POSITIVE TOWARD LEADING EDGE
             ------------------------------------------------------------------------------------------------------------------------
             AXES OF BLADE ELEMENT        Z-AXIS IS PARALLEL TO SHAFT AXIS; ORIGIN AT INBOARD END OF ELEMENT
                                          X-AXIS SPANWISE (POSITIVE OUTWARD); Y-AXIS CHORDWISE
                                          ROTOR 1 Y-AXIS POSITIVE TOWARD TRAILING EDGE
                                          (APPROXIMATE; ACTUALLY X IS BEAM AXIS OF ELEMENT, Y-AXIS IS IN HUB PLANE)
             ------------------------------------------------------------------------------------------------------------------------
             AXES OF TRAILING EDGE FLAP   X-AXIS IS FLAP HINGE AXIS (SPANWISE, POSITIVE OUTWARD); ORIGIN AT FLAP HINGE
                                          Y-AXIS IN HUB PLANE (CHORDWISE); Z-AXIS APPROXIMATELY PARALLEL TO SHAFT AXIS
                                          ROTOR 1 Y-AXIS POSITIVE TOWARD TRAILING EDGE
             ------------------------------------------------------------------------------------------------------------------------
```

```
*************
TRIM SOLUTION
*************

OUTPUT OF PART = ROTOR 1 BLADE POSITION SENSOR


OUTPUT = ROTOR 1 BLADE 4 POS FLJOINT

         OUTPUT KIND = COMPONENT OUTPUT                                                                    OUTPUT SEQUENCE =    55
         RESPONSE KIND = OUTPUT                                                                          RESPONSE SEQUENCE =   728
             COMPONENT = ROTOR 1 BLADE 4 ELEMENT 1           OUTPUT = ROTOR 1 BLADE 4 POS FLJOINT

             ------------------------------------------------------------------------------------------------------------
             STRUCTURAL DYNAMIC COMPONENT, SENSOR    4; BLADE POSITION SENSOR, AT JOINT
             UNITS:  DISPLACEMENT = X / RADIUS (BLADE RADIUS R =  24.724 FT), ANGLE = DEG
             ELEMENT DEFINITION:  FLAP AND LAG HINGE ROTATION
             ------------------------------------------------------------------------------------------------------------

         SOLUTION KIND = TIME DOMAIN PERIODIC; NUMBER OF PERIODS =     1
             PERIOD   1 (PRIMARY) = ROTOR 1
             AZIMUTH FOR ZERO TIME =   0.000 DEG
             FREQUENCY =    27.227     RAD/SEC   =    4.3333     HZ   =    260.00     RPM
             NUMBER OF TIME STEPS IN SOLUTION =   24
             DIMENSIONLESS TIME PSI:  AZIMUTH = PSI, TIME = (AZIMUTH - (AZIMUTH FOR ZERO TIME))/OMEGA
         VARIABLE RESPONSE TYPE, LENGTH =    2; NOMINAL KIND = REST POSITION
         MOTION = NOMINAL + DOF
                   ELEMENT    NAME                                LABEL         NOMINAL
                   1          RTR 1 BLD 4 POS   JOINT    FLAP     FLAP          0.000000E+00
                   2          RTR 1 BLD 4 POS   JOINT    LAG      LAG           0.000000E+00


         MOTION VALUE           FLAP            LAG
         TIME HISTORY
           MEAN                  4.21460         11.1450
           MAXIMUM               5.41003         11.4647
           MINIMUM               2.85712         10.8017
           1/2 PEAK-TO-PEAK      1.27646        0.331498
         TIME HISTORY (AZIMUTH IN DEG)
           PSI =   0.000         3.54013         10.8481
           PSI =  15.000         2.99181         10.8047
           .............         .......         .......
           .............         .......         .......
           PSI = 345.000         4.19055         10.8888
           PSI = 360.000         3.54013         10.8481
         HARMONICS (FROM   24 TIME STEPS)
           MEAN                  4.21460         11.1450
           COSINE   1          -0.259376E-01   -0.307642
           SINE     1           0.157176E-01    0.168354
           .............         .......         .......
           .............         .......         .......
           COSINE  10           0.455621E-02    0.139678E-02
           SINE    10          -0.683559E-02   -0.521005E-03
                                                                                                    ====== WRITING TO PLOT DATA FILE ======
....
....
```

```
*************
TRIM SOLUTION
*************

OUTPUT OF PART = ROTOR 1 WING SENSOR
....
....
....

OUTPUT = ROTOR 1 WING 4 SENSOR   6

         OUTPUT KIND = COMPONENT OUTPUT                                                                   OUTPUT SEQUENCE =   102
         RESPONSE KIND = OUTPUT                                                                         RESPONSE SEQUENCE =  1010
             COMPONENT = ROTOR 1 WING 4                      OUTPUT = ROTOR 1 WING 4 SENSOR   6

             -------------------------------------------------------------------------------------------------------------------
             LIFTING LINE WING COMPONENT, AERODYNAMIC SENSOR    7
             QUANTITY DEFINITION:  SECTION FORCE; Z COMPONENT, WING SECTION AXES
             UNITS:  M**2 * SECTION COEFFICIENT FORM  --  FORCE/LENGTH M**2*CF = F / .5*RHO*CS**2*C
                 VELOCITY SCALE = SOUND SPEED CS = 1116.45 FT/SEC; LENGTH SCALE = SECTION CHORD C; TIME SCALE TREF = CS/U
                 AIR DENSITY RHO = 0.002378 SLUG/FT**3; WING SPAN BREF =   19.7792 FT; SPAN INTEGRAL SCALED WITH WING SPAN BREF
             ELEMENTS:  QUANTITY AT  19 SPAN STATIONS, AND SPAN INTEGRAL
             -------------------------------------------------------------------------------------------------------------------

         SOLUTION KIND = TIME DOMAIN PERIODIC; NUMBER OF PERIODS =     1
             PERIOD   1 (PRIMARY) = ROTOR 1
             AZIMUTH FOR ZERO TIME =   0.000 DEG
             FREQUENCY =    27.227     RAD/SEC    =    4.3333     HZ    =    260.00     RPM
             NUMBER OF TIME STEPS IN SOLUTION =   24
             DIMENSIONLESS TIME PSI:  AZIMUTH = PSI, TIME = (AZIMUTH - (AZIMUTH FOR ZERO TIME))/OMEGA


         NUMBER OF SPAN STATIONS =  19, RAMIN =   0.20000, RAMAX =   1.00000
         SPAN STATION            0.25000         0.34000         0.42000         0.49500         0.56000         0.61500
         TIME HISTORY (AZIMUTH IN DEG)
           PSI =   0.000        0.131863E-01   -0.136295E-01    0.270109E-01    0.570006E-01    0.732155E-01    0.899542E-01
           PSI =  15.000        0.246759E-02    0.311239E-01    0.261462E-01    0.489125E-01    0.794093E-01    0.105650
           .............        ............    ............    ............    ............    ............    ............
           .............        ............    ............    ............    ............    ............    ............
           PSI = 345.000       -0.785984E-05    0.235235E-01    0.283153E-01    0.246407E-01    0.403295E-01    0.614545E-01
           PSI = 360.000        0.131863E-01   -0.136295E-01    0.270109E-01    0.570006E-01    0.732155E-01    0.899542E-01

           ....
           ....

         SPAN STATION            0.99000         INTEGRAL
         TIME HISTORY (AZIMUTH IN DEG)
           PSI =   0.000        0.103863        0.874264E-01
           PSI =  15.000        0.105340        0.981071E-01
           .............        ............    ............
           .............        ............    ............
           PSI = 345.000        0.963893E-01    0.725137E-01
           PSI = 360.000        0.103863        0.874264E-01
```

```
NUMBER OF SPAN STATIONS =  19, RAMIN =   0.20000, RAMAX =   1.00000
  ....
  ....
           SPAN STATION =   (*)  0.93000        (+)  0.95000        (0)  0.97000
        -3 X 10** -1                                         0.0                                           3 X 10** -1
           I.........I.........I.........I.........I.........I.........I.........I.........I.........I.........I
PSI =   0.0                                                  I                     0  + *
PSI =  15.0                                                  I                     0   +  *
PSI =  30.0                                                  I                     0   +  *
PSI =  45.0                                                  I                   0  +  *
PSI =  60.0                                                  I              0  +  *
PSI =  75.0                                                  I        0 + *
PSI =  90.0                                                  I 0+*
PSI = 105.0                                              0+  I
PSI = 120.0                                              0+  I
PSI = 135.0                                                  I 0+*
PSI = 150.0                                                  I           0 +*
PSI = 165.0                                                  I                    0  +*
PSI = 180.0                                                  I                         0 + *
PSI = 195.0                                                  I                        0  +*
PSI = 210.0                                                  I                     0 +*
PSI = 225.0                                                  I                  0 +*
PSI = 240.0                                                  I                0 +
PSI = 255.0                                                  I               0+*
PSI = 270.0                                                  I              0+*
PSI = 285.0                                                  I              0+*
PSI = 300.0                                                  I               0+*
PSI = 315.0                                                  I              0 +*
PSI = 330.0                                                  I                0 + *
PSI = 345.0                                                  I                   0  +*
PSI = 360.0                                                  I                     0  + *
           I.........I.........I.........I.........I.........I.........I.........I.........I.........I.........I
  ....
  ....
                                                                          ====== WRITING TO PLOT DATA FILE ======
....
....

TRIM REFERENCE UPDATE
```

```
***********************
SYSTEM AND FUNCTIONALITY
***********************

-------------------------
FLUTTER SOLUTION PROCEDURE
-------------------------

     ==========================================================================================
     FIRST LOOP = ROTORCRAFT
              PARTS SOLVED BEFORE CHILD LOOPS
              PART = GUST
                       CHILD LOOP = ROTOR 1
                                         MODULES WRITTEN AT END OF LOOP
                                         MODULE = MODES = ROTOR 1 BLADE 4 FLUTTER
     ------------------------------------------------------------------------------------------
     LOOP = ROTOR 1
                       CHILD LOOP = ROTOR 1 BLADE 4
     ------------------------------------------------------------------------------------------
     LOOP = ROTOR 1 BLADE 4
              PARTS SOLVED BEFORE CHILD LOOPS
              PART = ROTOR 1 BLADE 4 CONTROL
              DE PART = ROTOR 1 BLADE 4
     ==========================================================================================


INITIALIZATION FOR FLUTTER
GENERATE SUBMATRICES FOR FLUTTER PART RECORDS
GENERATE SUBMATRICES FOR FLUTTER MODES RECORDS


----------------------------------------
FLUTTER LOOP ITERATION AND PART SOLUTION
----------------------------------------
```

****

FLUTTER SOLUTION

****

MODE SET = ROTOR 1 BLADE 4 FLUTTER

MODAL TRANSFORMATION FOR PART = ROTOR 1 BLADE 4

ONLY MODE SHAPES ARE USED IN SOLUTION PROCEDURE; FREQUENCY AND MODAL MASS ARE PRESENTED FOR INFORMATION

MODE SHAPE DISPLACEMENT (FLAP, LAG, AXIAL, LINEAR JOINT) SCALED WITH ROTOR RADIUS; ROTATION (PITCH, ANGULAR JOINT) IN RADIANS

ROTOR ROTATIONAL SPEED = 27.227 RAD/SEC = 4.3333 HZ = 260.000 RPM

MODE			MODAL FREQUENCY			MODAL	MODAL DAMPING	MODE SHAPE AT TIP			
NUMBER	LABEL	UTILIZATION	PER REV	RAD/SEC	HZ	MASS	G=2*CRITICAL	FLAP	LAG	PITCH	AXIAL
1	M5E1	DYNAMIC	0.2215	6.032	0.9600	1238.8	0.0000	0.000	1.000	0.047	-0.246
2	M5E2	DYNAMIC	1.0218	27.822	4.4279	1189.1	0.0000	1.000	-0.016	0.093	-0.061
3	M5E3	DYNAMIC	2.8566	77.777	12.3785	100.78	0.0000	-0.339	-0.077	1.000	0.040
4	M5E4	DYNAMIC	4.7114	128.279	20.4163	57.449	0.0000	-0.119	0.187	1.000	-0.077
5	M5E5	DYNAMIC	5.1700	140.764	22.4033	0.64482	0.0000	-0.049	-0.013	1.000	0.007
6	M5E6	DYNAMIC	5.9562	162.170	25.8101	1.2431	0.0000	-0.013	-0.001	1.000	0.001
7	M5E7	DYNAMIC	10.0124	272.609	43.3871	12.790	0.0000	0.082	0.008	1.000	-0.002
8	M5E8	DYNAMIC	13.9513	379.855	60.4558	225.49	0.0000	-0.100	0.363	1.000	-0.239
9	M5E9	DYNAMIC	15.3666	418.388	66.5886	5.9904	0.0000	-0.102	-0.018	1.000	0.009
10	M5E10	DYNAMIC	19.1520	521.455	82.9922	0.52240	0.0000	-0.022	-0.005	1.000	0.003
11	M5E11	STATIC	25.9464	706.445	112.4343	6.9881	0.0000	0.058	0.007	1.000	-0.001
12	M5E12	STATIC	28.2659	769.599	122.4855	119.49	0.0000	-0.078	0.205	1.000	-0.368
13	M5E13	STATIC	29.8656	813.156	129.4177	2733.3	0.0000	-0.144	1.000	0.742	0.438
14	M5E14	STATIC	34.3678	935.738	148.9273	0.57636	0.0000	-0.031	-0.005	1.000	0.003
15	M5E15	STATIC	46.3886	1263.029	201.0173	6.2591	0.0000	-0.120	0.013	1.000	-0.023
16	M5E16	STATIC	46.6911	1271.266	202.3282	37.387	0.0000	-0.098	-0.166	1.000	0.157
17	M5E17	STATIC	57.2403	1558.488	248.0411	2.2187	0.0000	-0.003	-0.002	1.000	0.002
18	M5E18	STATIC	62.4466	1700.241	270.6017	0.64587	0.0000	0.027	0.004	1.000	-0.002
19	M5E19	STATIC	75.9098	2066.807	328.9425	7.3313	0.0000	0.016	-0.068	1.000	0.100
20	M5E20	STATIC	79.0744	2152.970	342.6559	0.14960	0.0000	-0.002	0.002	1.000	-0.003
....		......	....	....	....	......	......	.....	.....	.....	.....
....		......	....	....	....	......	......	.....	.....	.....	.....
38	M5E38	STATIC	897.9336	24448.162	3891.0457	45.895	0.0000	0.018	-0.312	1.000	-0.277
39	M5E39	STATIC	1840.1981	50103.328	7974.1919	3.0502	0.0000	-0.030	0.251	1.000	0.305

```
OUT-OF-PLANE COMPONENT OF MODE SHAPE
==============================================================================================================================
MODE LABEL                                   M5E1           M5E2           M5E3           M5E4           M5E5           M5E6
------------------------------------------------------------------------------------------------------------------------------
MODE NUMBER                                     1              2              3              4              5              6
UTILIZATION                                DYNAMIC        DYNAMIC        DYNAMIC        DYNAMIC        DYNAMIC        DYNAMIC
MODAL FREQUENCY (PER REV)                   0.2215         1.0218         2.8566         4.7114         5.1700         5.9562
------------------------------------------------------------------------------------------------------------------------------
MODE SHAPE
RTR 1 BLD 4 MODE   JOINT   FLAP            0.01505        1.05102        0.84344        0.04091       -0.05924        0.15019
RTR 1 BLD 4 MODE 0.1000R   FLAP           -0.00002        0.06474        0.05176        0.00315       -0.00365        0.00922
RTR 1 BLD 4 MODE 0.2000R   FLAP           -0.00003        0.16791        0.13115        0.00803       -0.00891        0.02173
RTR 1 BLD 4 MODE 0.3000R   FLAP            0.00019        0.27157        0.19505        0.01613       -0.01142        0.02465
RTR 1 BLD 4 MODE 0.4000R   FLAP            0.00050        0.37583        0.23144        0.02937       -0.00916        0.01403
RTR 1 BLD 4 MODE 0.5000R   FLAP            0.00089        0.48051        0.23426        0.04254       -0.00325       -0.00317
RTR 1 BLD 4 MODE 0.6000R   FLAP            0.00174        0.58555        0.19808        0.04937        0.00416       -0.01894
RTR 1 BLD 4 MODE 0.7000R   FLAP            0.00263        0.69088        0.11865        0.04218        0.00996       -0.02330
RTR 1 BLD 4 MODE 0.8000R   FLAP            0.00307        0.79649        0.00205        0.01796        0.01135       -0.01350
RTR 1 BLD 4 MODE 0.9000R   FLAP            0.00335        0.90225       -0.14225       -0.02282        0.00638        0.00739
RTR 1 BLD 4 MODE 1.0000R   FLAP           -0.00041        1.00000       -0.33899       -0.11946       -0.04883       -0.01292
==============================================================================================================================
....
....
                                                                                     ====== WRITING TO PLOT DATA FILE ======


****************
FLUTTER SOLUTION
****************

------------------------
DESCRIPTION OF VARIABLES
------------------------

    DEGREES OF FREEDOM
         VARIABLE  LABEL      TRANSFORMS      COMPONENT    VECTOR                              ELEMENT
         ------------------------------------------------------------------------------------------------------------------------
           1       R1MB4                           MODE    ROTOR 1 BLADE 4 FLUTTER             ROTOR 1 BLADE 4 MODE   1
           2       R2MB4                           MODE    ROTOR 1 BLADE 4 FLUTTER             ROTOR 1 BLADE 4 MODE   2
           3       R3MB4                           MODE    ROTOR 1 BLADE 4 FLUTTER             ROTOR 1 BLADE 4 MODE   3
           4       R4MB4                           MODE    ROTOR 1 BLADE 4 FLUTTER             ROTOR 1 BLADE 4 MODE   4
           5       R5MB4                           MODE    ROTOR 1 BLADE 4 FLUTTER             ROTOR 1 BLADE 4 MODE   5
           6       R6MB4                           MODE    ROTOR 1 BLADE 4 FLUTTER             ROTOR 1 BLADE 4 MODE   6
           7       R7MB4                           MODE    ROTOR 1 BLADE 4 FLUTTER             ROTOR 1 BLADE 4 MODE   7
           8       R8MB4                           MODE    ROTOR 1 BLADE 4 FLUTTER             ROTOR 1 BLADE 4 MODE   8
           9       R9MB4                           MODE    ROTOR 1 BLADE 4 FLUTTER             ROTOR 1 BLADE 4 MODE   9
          10       R10MB4                          MODE    ROTOR 1 BLADE 4 FLUTTER             ROTOR 1 BLADE 4 MODE  10
         ------------------------------------------------------------------------------------------------------------------------
         TRANSFORMS:  MULTIBLADE COORDINATES (0,1C,1S,...,NC,NS,N/2), ROTATING TO NONROTATING (COS,SIN), SYMMETRIC/ANTISYMMETRIC
```

```
**********************************************************************
ANALYSIS OF SYSTEM OF CONSTANT COEFFICIENT LINEAR DIFFERENTIAL EQUATIONS
**********************************************************************
FLUTTER SOLUTION


    SECOND ORDER FORM
         NUMBER OF DEGREES OF FREEDOM, MX =   10
         NUMBER OF CONTROLS,           MV =    0
         NUMBER OF OUTPUT,             MY =    0
    FREQUENCY SCALE FACTOR (RAD/SEC) =      27.2271
    QUASISTATIC REDUCTION PERFORMED WITH OUTPUT RESIDUAL
    MATRIX INPUT FORM:  DIMENSIONAL TIME
    MATRIX OUTPUT FORM:  DIMENSIONAL TIME
    EIGENVALUES WRITTEN TO PLOT FILE
    NUMBER OF EQUATION SETS ANALYZED = 1


    DEGREES OF FREEDOM
         NAME          UTILIZATION      SYMMETRY           RIGID      ORDER              DEGREE OF FREEDOM AS OUTPUT
         ------------------------------------------------------------------------------------------------------------------------
         R1MB4         DYNAMIC          BOTH               NO         FROM EQUATIONS
         R2MB4         DYNAMIC          BOTH               NO         FROM EQUATIONS
         R3MB4         DYNAMIC          BOTH               NO         FROM EQUATIONS
         R4MB4         DYNAMIC          BOTH               NO         FROM EQUATIONS
         R5MB4         DYNAMIC          BOTH               NO         FROM EQUATIONS
         R6MB4         DYNAMIC          BOTH               NO         FROM EQUATIONS
         R7MB4         DYNAMIC          BOTH               NO         FROM EQUATIONS
         R8MB4         DYNAMIC          BOTH               NO         FROM EQUATIONS
         R9MB4         DYNAMIC          BOTH               NO         FROM EQUATIONS
         R10MB4        DYNAMIC          BOTH               NO         FROM EQUATIONS
         ------------------------------------------------------------------------------------------------------------------------
```

```
------------------------------------------------------------------------
ANALYSIS OF SYSTEM OF CONSTANT COEFFICIENT LINEAR DIFFERENTIAL EQUATIONS
------------------------------------------------------------------------
EQUATION SET NUMBER 4:  COMPLETE EQUATIONS,      DESIGNATED QUASISTATIC
FLUTTER SOLUTION
                                                                                               ====== WRITING TO PLOT DATA FILE ======


    FIRST ORDER FORM
         NUMBER OF STATES,   MXX =   20
         NUMBER OF CONTROLS, MVV =    0
         NUMBER OF OUTPUT,   MYY =    0
         NUMBER OF QUASISTATIC DEGREES OF FREEDOM =    0
         NUMBER OF FIRST ORDER STATES =    0


    --------------------------------------------------------------------------------------------------------------------
    NAMES OF STATES:      D(R1MB4   )/DT      D(R2MB4   )/DT      D(R3MB4   )/DT      D(R4MB4   )/DT      D(R5MB4   )/DT
                          D(R6MB4   )/DT      D(R7MB4   )/DT      D(R8MB4   )/DT      D(R9MB4   )/DT      D(R10MB4  )/DT
                          R1MB4               R2MB4               R3MB4               R4MB4               R5MB4
                          R6MB4               R7MB4               R8MB4               R9MB4               R10MB4
    --------------------------------------------------------------------------------------------------------------------
```

EIGENVALUES

LAMBDA = REAL + I*IMAG, FREQUENCY = IMAG, NATURAL FREQUENCY = MAG(LAMBDA)
DAMPING RATIO = - REAL(LAMBDA) / MAG(LAMBDA)
PERIOD = 1 / FREQUENCY(HZ), TIME CONSTANT = - 1 / REAL(LAMBDA)
TIME TO HALF (DOUBLE) AMPLITUDE = 0.6931472 * TIME CONSTANT
FREQUENCY SCALE FACTOR = 27.2271 RAD/SEC

LAMBDA REAL PER/REV	LAMBDA IMAG PER/REV	NATURAL FREQUENCY PER/REV	LAMBDA REAL 1/SEC	LAMBDA IMAG RAD/SEC	DAMPING RATIO	FREQUENCY HZ	PERIOD SEC	TIME CONSTANT SEC	TIME HALF AMPLITUDE SEC
-0.08655	17.24194	17.24216	-2.3566	469.4488	0.0050	74.715	0.0134	0.4243	0.2941
-0.08655	-17.24194	17.24216	-2.3566	-469.4488	0.0050	74.715	0.0134	0.4243	0.2941
-0.19559	14.36533	14.36666	-5.3253	391.1267	0.0136	62.250	0.0161	0.1878	0.1302
-0.19559	-14.36533	14.36666	-5.3253	-391.1267	0.0136	62.250	0.0161	0.1878	0.1302
-1.96705	11.05222	11.22590	-53.5572	300.9203	0.1752	47.893	0.0209	0.0187	0.0129
-1.96705	-11.05222	11.22590	-53.5572	-300.9203	0.1752	47.893	0.0209	0.0187	0.0129
-0.17269	8.70923	8.71094	-4.7018	237.1274	0.0198	37.740	0.0265	0.2127	0.1474
-0.17269	-8.70923	8.71094	-4.7018	-237.1274	0.0198	37.740	0.0265	0.2127	0.1474
-0.00692	5.69705	5.69705	-0.1883	155.1142	0.0012	24.687	0.0405	5.3113	3.6815
-0.00692	-5.69705	5.69705	-0.1883	-155.1142	0.0012	24.687	0.0405	5.3113	3.6815
-0.00605	4.65397	4.65398	-0.1647	126.7144	0.0013	20.167	0.0496	6.0701	4.2075
-0.00605	-4.65397	4.65398	-0.1647	-126.7144	0.0013	20.167	0.0496	6.0701	4.2075
-1.26894	4.39969	4.57903	-34.5495	119.7911	0.2771	19.065	0.0525	0.0289	0.0201
-1.26894	-4.39969	4.57903	-34.5495	-119.7911	0.2771	19.065	0.0525	0.0289	0.0201
-0.04942	2.73792	2.73837	-1.3456	74.5458	0.0180	11.864	0.0843	0.7432	0.5151
-0.04942	-2.73792	2.73837	-1.3456	-74.5458	0.0180	11.864	0.0843	0.7432	0.5151
-0.00241	1.03347	1.03347	-0.0655	28.1383	0.0023	4.478	0.2233	15.2683	10.5832
-0.00241	-1.03347	1.03347	-0.0655	-28.1383	0.0023	4.478	0.2233	15.2683	10.5832
-0.10787	0.19612	0.22383	-2.9370	5.3399	0.4819	0.850	1.1767	0.3405	0.2360
-0.10787	-0.19612	0.22383	-2.9370	-5.3399	0.4819	0.850	1.1767	0.3405	0.2360

COMPUTATION TIMES

	CPU TIME (SEC)	PERCENT OF CASE	NUMBER OF CALLS	TIME PER CALL (SEC)
CASE	xxx.xxxx	100.00	1	xxx.xxxxx
INPUT	xxx.xxxx	0.74	1	xxx.xxxxx
INITIALIZE	xxx.xxxx	0.55	3	xxx.xxxxx
TRIM	xxx.xxxx	97.15	1	xxx.xxxxx
FLUTTER	xxx.xxxx	1.56	1	xxx.xxxxx
TRIM PART SOLUTION	xxx.xxxx	97.08	9938	xxx.xxxxx
TRIM DIFFERENTIAL EQUATIONS	xxx.xxxx	0.78	5	xxx.xxxxx
TRIM OUTPUT	xxx.xxxx	0.02	6	xxx.xxxxx
FLUTTER PART SOLUTION	xxx.xxxx	1.55	36	xxx.xxxxx
FLUTTER DIFFERENTIAL EQUATIONS	xxx.xxxx	1.50	12	xxx.xxxxx
FLUTTER OUTPUT	xxx.xxxx	0.00	2	xxx.xxxxx
MODES	xxx.xxxx	0.59	3	xxx.xxxxx

DATA VECTOR STATISTICS

DATA VECTOR	NUMBER OF SECTIONS		DIRECTORY SPACE	MAXIMUM USED SPACE	MINIMUM FREE SPACE	TOTAL LENGTH	NUMBER OF PACKS
DTVCCM	35972	REAL		1930981 (19%)	6899298 (69%)	10000000	1
DTVCCM	35972	INTEGER	431695	1129210 (11%)	8347586 (83%)	10000000	1
DTVCCM	35972	CHARACTER		258996 (26%)	737684 (74%)	1000000	1
CORECM	4763	REAL		64751 (6%)	935249 (94%)	1000000	0
CORECM	4763	INTEGER	57187	160820 (16%)	839180 (84%)	1000000	0
CORECM	4763	CHARACTER		402336 (20%)	1519331 (76%)	2000000	0
TABLCM	1	REAL		7267 (1%)	992733 (99%)	1000000	0
TABLCM	1	INTEGER	43	259 (0%)	99741 (99%)	100000	0
TABLCM	1	CHARACTER		106 (0%)	99894 (99%)	100000	0
SHLLCM	14	REAL		31863 (8%)	368137 (92%)	400000	0
SHLLCM	14	INTEGER	199	9595 (10%)	90405 (90%)	100000	0
SHLLCM	14	CHARACTER		11591 (12%)	88409 (88%)	100000	0

END OF CASE NUMBER 1

****

CLOSE FILE 43

ANALYSIS EXIT (END OF PROGRAM)

7–2 Trim Convergence

REFERENCE: CAMRAD II Documentation, Volume VI, Rotorcraft Input; Chapter “Convergence and Efficiency”

PARTITIONED, ITERATIVE PROCEDURE CONSTRUCTED FOR TRIM TASK

system divided into parts, which solve subset of equations for periodic or constant response

loops iterate between part solutions, until converged system solution obtained

figure 1 shows basic loops and principal interface variables

TRIM LOOPS:

WAKE, TRIM, CIRCULATION, AND MOTION

regulator loop can be introduced, between trim and circulation loops

TRIM PARTS USING HARMONIC OR TIME FINITE ELEMENT SOLUTION METHOD:

ROTOR, AIRFRAME, AND DRIVE TRAIN

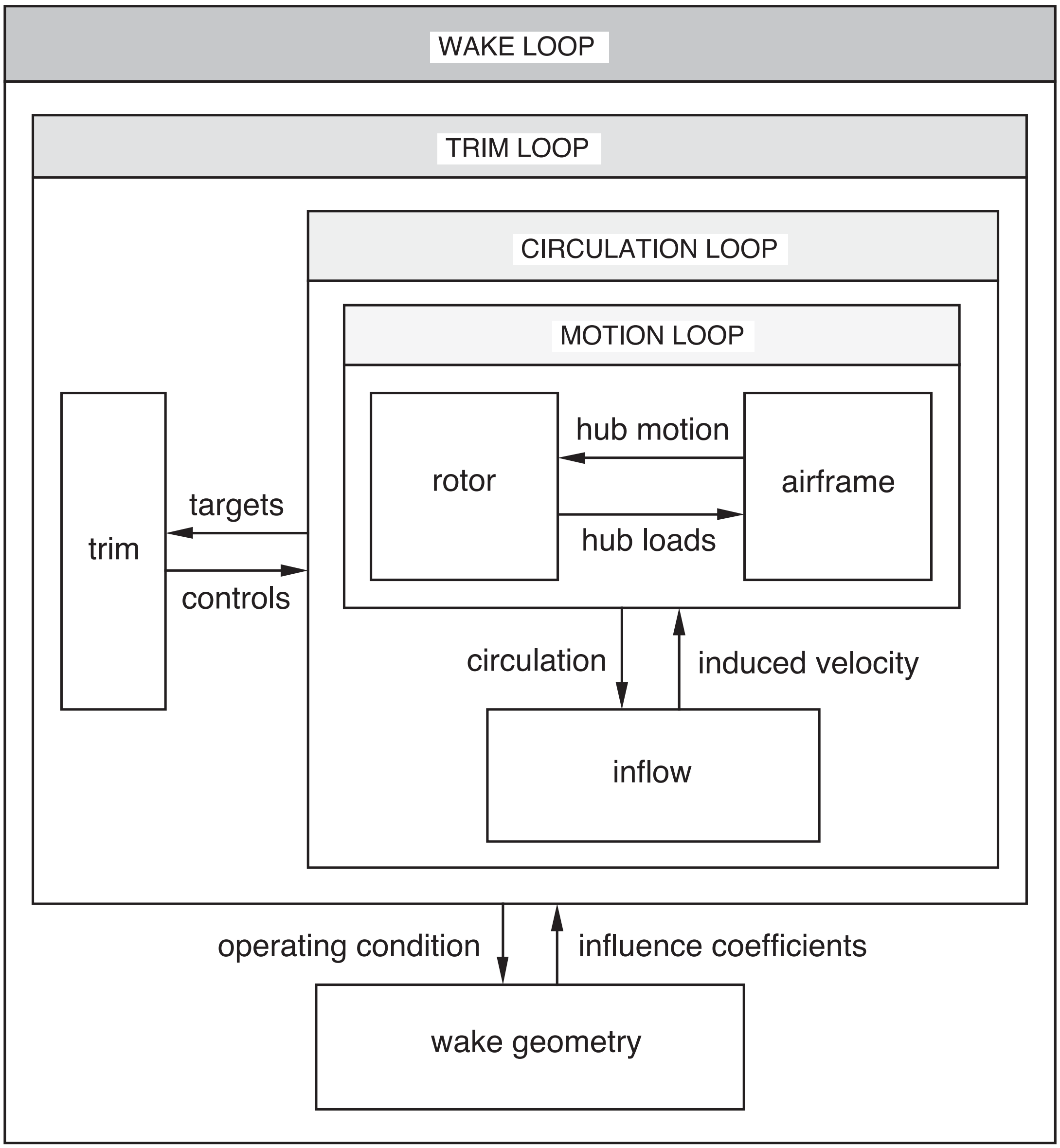


Figure 7-1a Basic loops of trim analysis (optional regulator loop not shown).

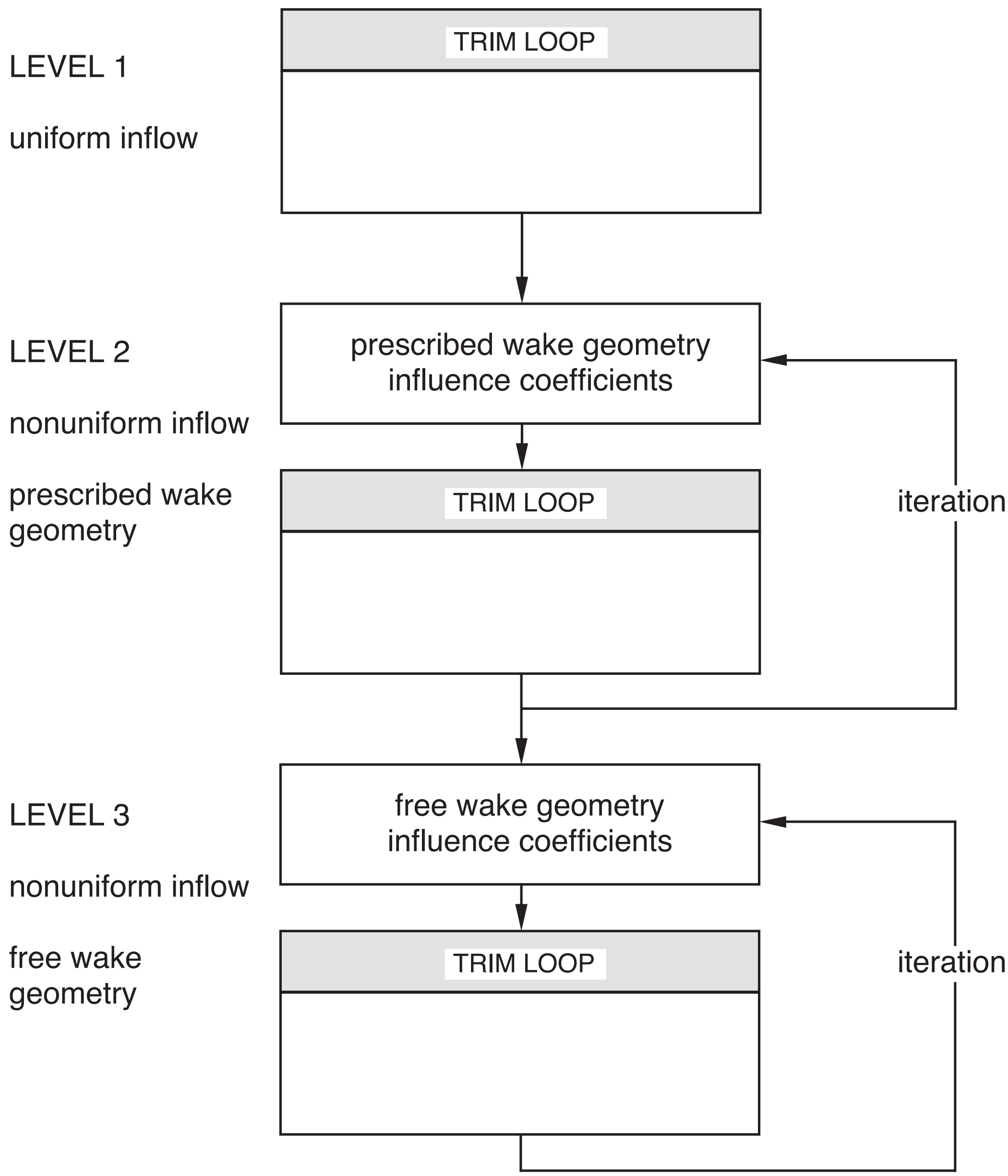


Figure 7-1b Levels of wake loop.

EACH LOOP AND EACH PART MUST BE CONVERGED

RESULTS ARE UNRELIABLE IF ANY ITERATION IS NOT CONVERGED

warnings are produced if iteration does not converge

trace during loop or part solution (input parameters TRACEL and TRACEP) can be used to obtain more information about convergence

SEVERE DIVERGENCE CAN PRODUCE FLOATING POINT OVERFLOW

ITERATIVE LOOP AND PART SOLUTIONS

FOR EACH ITERATION, CONVERGENCE CONTROLLED BY

RELAXATION FACTOR
TOLERANCE
NUMBER OF ITERATIONS

trim loop also depends on initial conditions, and derivative matrix identification

FIRST ACTION TO IMPROVE OR ACHIEVE CONVERGENCE IS TO REDUCE RELAXATION FACTOR

for efficiency, relaxation factor should be as large as possible

good practice to establish required value for any new class of problems

if relaxation factor is reduced significantly, it may also be necessary to:

increase number of iterations allowed

reduce tolerance

IF USUAL RANGE OF RELAXATION FACTOR VALUES DOES NOT PRODUCE CONVERGENCE, THEN OTHER PARAMETERS CAN BE CONSIDERED

TOLERANCE IN TEST FOR CONVERGENCE CONTROLS ACCURACY OF SOLUTION, AFTER CONVERGENCE HAS BEEN ACHIEVED

- interpretation of tolerance value depends on weights in convergence test
- for efficiency, tolerance should be as large as possible
  - establish required value for any new class of problems
  - by reducing tolerance until answer of interest is no longer affected, based on appropriate engineering judgement
- tolerance required for accurate solution can be affected by other parameters
  - relaxation factor functions by reducing difference between successive iterations
  - so if convergence test is applied to that difference, may need to reduce tolerance whenever relaxation factor is reduced
    - otherwise convergence test satisfied and iteration stops simply because small relaxation factor slows rate of convergence

CONVERGENCE OF LOOP CAN DEPEND ON ACCURACY OF SOLUTION OF ITS CHILD LOOPS AND PARTS

so may be necessary to use small tolerance in order to achieve convergence of some parent loop

IF SOLUTION DOES NOT CONVERGE BUT DOES NOT EXHIBIT OSCILLATORY OR DIVERGENT BEHAVIOR

may be sufficient to just increase maximum number of iterations allowed

CONVERGENCE PROBLEMS CAN ALSO REFLECT PHYSICAL LIMITATIONS OF SYSTEM DEFINED

POSSIBLE TO DEFINE PROBLEM THAT DOES NOT HAVE A SOLUTION

must change definition of system, or change definition of solution task

WAKE LOOP

LEVELS OR STAGES

LEVEL 1: UNIFORM INFLOW

LEVEL 2: NONUNIFORM INFLOW WITH PRESCRIBED/RIGID WAKE GEOMETRY

LEVEL 3: NONUNIFORM INFLOW WITH FREE WAKE GEOMETRY

NO TEST FOR CONVERGENCE

execute specified number of iterations for each stage

required number of stages and number of iterations should be established for any new class of problems

increase wake level and number of iterations until answer of interest is no longer affected, based on appropriate engineering judgement

results of all wake levels can be examined in single job

ANALYSIS LEVEL

increase analysis level to improve accuracy

nonuniform inflow often affects results

free wake geometry generally important at low speed (below advance ratio of about $\mu = 0.25$)

WAKE GEOMETRY ITERATIONS

often just one iteration required for each stage

increase number of iterations of last stage to improve convergence

- increasing number of iterations of earlier stages is less effective

- more than one iteration for uniform inflow has no effect

iteration is required if anything is being calculated by inner loops that will change wake geometry

- usually not necessary to iterate on wake geometry if trim loop specifies rotor speed, thrust, and tip-path plane angle-of-attack

  - so in practice, single iteration of each required stage often sufficient

some problems need iteration in wake geometry loop (and relaxation factor on wake geometry)

if rotor is at fixed collective pitch (rather than trimmed to specified thrust), then overall geometry of wake is not known in advance, only after thrust calculated

or in hover and at low speeds, where rotor loading is very sensitive to wake geometry

- then 4–6 iterations often required

- momentum or vortex theory suggests value of 0.5 for relaxation factor or with dual-peak wake model

- span station of inboard peak r_{GI} used to calculate influence coefficients
- r_{GI} is wake geometry feature, potentially requiring additional wake loop iterations for convergence
- typically one more iteration of last stage required with dual-peak model
    - but effect of r_{GI} is not large, and other wake geometry features updated at same time
- convergence of r_{GI} assessed by comparing value used to calculate influence coefficients with value found in current circulation solution

TRIM LOOP

RELAXATION FACTOR

reduce relaxation factor to achieve convergence

use value just small enough to prevent oscillation of Newton-Raphson solution

for simple problem of trimming thrust in hover, relaxation factor of 0.7 can be satisfactory

in extreme operating conditions, value of 0.25 or less may be required

PERTURBATION IDENTIFICATION

improve perturbation identification of derivative matrix in order to achieve convergence

using second-order difference often very effective

or change sign or magnitude of perturbation; or recalculate matrix more often

RECURSIVE IDENTIFICATION

use recursive identification of derivative matrix in order to improve convergence

or change weight parameter

if parameter variance matrix of recursive update diverges, then perturbation identification should be performed more often, to re-initialize recursive identification

TRIM ITERATIONS

increase maximum number of iterations in order to achieve convergence

INITIAL CONDITIONS

use better initial conditions for trim variables to achieve convergence

trim loop unlikely to converge if started too far from the solution

these initial conditions are for uniform inflow stage, even if nonuniform inflow is being used (uniform inflow solution initializes nonuniform inflow stage)

first task in project dealing with new problem is to establish set of initial conditions for operating conditions of interest

initial conditions can be obtained from nearby operating condition

can perform operating condition sweep of several cases in a single job

starting from some case that easily converges, such as hover

with analysis using trim solution from previous case as initial conditions

can also use solution obtained by turning off stall in rotor model as initial conditions for full solution

CONVERGENCE OF INNER LOOPS

may need to decrease tolerance on child loops and parts in order to achieve convergence of trim loop

> if inner loop is not sufficiently converged, trim iteration may become erratic or even diverge

inner loops may not be converged during perturbation process to identify derivative matrix

> resulting derivative matrix can be bad, so trim iteration diverges even if inner loops finally converge
>
> > then must achieve better convergence of inner loops during perturbation process

PHYSICAL LIMITATIONS

CONVERGENCE PROBLEMS CAN ALSO REFLECT PHYSICAL LIMITATIONS OF ROTORCRAFT DEFINED; FOR EXAMPLE:

operating condition specified may be beyond capabilities of rotor to produce thrust or propulsive force

> in free flight at high gross weight, high turn rate, or high speed
>
> or in wind tunnel at high thrust
>
> > trim problem should be changed: use specified power (free flight) or fixed collective pitch (wind tunnel)

operating condition specified may involve high main rotor power, and hence be beyond capabilities of tail rotor to produce thrust to balance main rotor torque

> tail rotor thrust limit can be increased, perhaps by suppressing stall
>
> or tail rotor can be replaced by auxiliary force

connection of pilot's controls to rotor controls may not be consistent with specified trim problem

> for example: tiltrotor aircraft in wind tunnel, not possible to trim lateral flapping with lateral stick since pilot's controls are not connected to rotor lateral cyclic
>
> > control matrix should be changed

for trim with yaw angle in free flight, be sure to use velocity yaw angle or sideslip (SIDE), not yaw of body axes relative inertial frame (YAW)

REGULATOR LOOP

REGULATOR ALGORITHM IS SIMILAR TO NEWTON-RAPHSON METHOD USED FOR TRIM LOOP

SO INFLUENCES OF PARAMETERS ON CONVERGENCE AND ACCURACY ARE SIMILAR

PHYSICAL LIMITATIONS

CONVERGENCE PROBLEMS CAN ALSO REFLECT PHYSICAL LIMITATIONS OF ROTORCRAFT DEFINED; FOR EXAMPLE:

controls might be used by both trim loop and regulator loop

cost function objective defined by tolerance may not be physically achievable

CIRCULATION LOOP

RELAXATION FACTOR

reduce relaxation factor to achieve convergence

relaxation factor of $\lambda = 0.10$ or 0.05 often required for nonuniform inflow models

smaller relaxation factor required with three-quarter chord collocation points than with quarter chord points

CIRCULATION ITERATIONS

increase maximum number of iterations in order to achieve convergence

AERODYNAMIC SPAN STATIONS

revise aerodynamic span stations to achieve convergence

spanwise oscillation in sign of angle-of-attack can be produced if small panel is next to large panel

panel widths should vary smoothly along blade span

in nonuniform inflow model, can have tip vortex rollup occur inboard of blade tip, at span station r_{TV}

any aerodynamic panels outboard of this rollup location will see large upwash, with significant influence on loads and perhaps convergence problems as well

generally $r_{TV} = e_{\text{tip}}$ should be used, unless inboard rollup is physically reasonable (as with highly tapered tip)

NEAR WAKE CORE SIZE

- with highly distorted geometry, some close encounters between collocation points and vortex lines of near wake can still occur
  - especially near a reverse flow boundary
- can increase core size of near wake line segments in order to avoid singularities and achieve convergence
  - but this core size has no physical significance
  - its value must not influence solution for loading

HOVERING ROTOR AT LOW THRUST

- circulation convergence problems will be encountered for hovering rotor at low thrust
  - circulation iteration then only asymptotically convergent with normal inflow and wake models
- avoid problem by fixing wake geometry in calculation of induced velocity at low thrust
  - for uniform inflow stage: change induced velocity model, using special inflow equation
  - for nonuniform inflow stage: wake geometry specified directly, rather than using prescribed geometry (which depends on thrust)
    - resulting inflow is not correct, but is small at low thrust

CIRCULATION CONVERGENCE IN NONUNIFORM INFLOW CALCULATION

- if use relatively small value for maximum number of circulation iterations (for efficiency), circulation may not be converged for uniform inflow stage, or in early trim iterations of nonuniform inflow stage
  - acceptable as long as trim converges, and circulation is converged at end of nonuniform inflow stage
  - uniform inflow gives approximate solution, which may be reasonable even with circulation not converged
    - check ratio of induced power to ideal power

- uniform inflow result initializes prescribed wake geometry stage
  - mean induced velocity determines prescribed/rigid wake geometry
  - if uniform inflow not sufficiently converged, may need more wake iterations in nonuniform inflow stage
    - if wake geometry is important, will need free wake geometry stage in any case

- RESULTS ARE UNRELIABLE IF CIRCULATION LOOP NOT CONVERGED AT END OF ANY ITERATION OF NONUNIFORM INFLOW STAGE IN WAKE LOOP

MOTION LOOP

RELAXATION FACTOR

reduce relaxation factor to achieve convergence

relaxation factor $\lambda < 1$ may be required when airframe vibration is being calculated

if airframe does not produce vibratory hub motion, no iteration is required, and relaxation factor should be 1.0

MOTION ITERATIONS

increase maximum number of iterations in order to achieve convergence

ROTOR, AIRFRAME, AND DRIVE TRAIN PARTS

RELAXATION FACTOR

reduce relaxation factor to achieve convergence

small relaxation factor may be required for rotor with significant elastic motion, or operating in dynamic stall

DAMPING FACTOR

estimate of aerodynamic damping can be added to both sides of differential equations, in order to improve convergence

harmonic solution procedure treats this damping same on both sides of equations, so damping value does not affect solution

typical value is 0.5 (fraction critical)

PART SOLUTIONS

increase maximum number of iterations of part solution in order to achieve convergence

convergence can also be affected by strategy for update of reference and matrices for differential equation solutions

7–3 Trim Efficiency

REFERENCE: CAMRAD II Documentation, Volume VI, Rotorcraft Input; Chapter “Convergence and Efficiency”

KEY TO EFFICIENT UTILIZATION OF ANALYSIS:

- USE MINIMUM COMPLEXITY AND SIZE REQUIRED FOR PARTICULAR CONFIGURATION AND PROBLEM
  - first phase of an investigation is best devoted to establishing minimum model required
- RECOMMENDATIONS REGARDING MODELLING REQUIREMENTS ARE BASED ON PAST EXPERIENCE WITH ANALYSIS
  - the more a new configuration or new problem differs from that experience, the more important it is to check these recommendations

PARAMETERS CONTROLLING SOLUTION PROCEDURE CAN INFLUENCE EFFICIENCY

- relaxation factor should be as large as possible
- tolerance should be as large as possible
  - in order to achieve converged and accurate solution with minimum number of iterations

MODEL SIZE

ROTORCRAFT SHELL PROVIDES NUMBER OF PARAMETERS THAT CONTROL MODEL SIZE

AZIMUTH STEP

azimuth step of 15 degrees typically used in solution procedures (24 azimuth steps per revolution)

NUMBER OF HARMONICS

typically 7 to 10 harmonics of motion required to calculate rotor structural loads

airframe vibration occurs at harmonics pN (N = number of blades

- typically $p = 1$ or 2 required to calculate vibration

generally 10 per-rev is limit of harmonic content required for performance and loads problems, consistent with:

- frequency content of excitation
- bandwidth of solution (one-half number of azimuth steps
- frequency range of rotor modes used

fewer harmonics, perhaps just 1/rev, may be sufficient for some problems

zero harmonics (just the mean motion) appropriate and efficient for ideal hover

- but rotorcraft in hover can still have tip-path plane tilt relative shaft, so at least 1/rev needed and 1/rev solution in rotating frame required to obtain mean nonrotating frame forces and moments acting on airframe

NUMBER OF BLADE MODES AND ELEMENTS

when blade modes are used in solution procedure:

typically 4 or 5 bending modes required to calculate structural loads, and 1 or 2 torsion modes

so typically 3 to 5 elastic elements needed to model blade (whether modes used or not)

generally modes with natural frequency below 10 per-rev may be important

but selecting modes based on frequency alone gives many flap modes, less lag modes, and few torsion modes

fewer modes and elements, perhaps just rigid flapping, may be sufficient for some problems

AIRFRAME ELASTIC MODES

airframe elastic modes required to calculate vibration

typically need those modes with natural frequency beyond pN/rev vibration of interest

MODEL COMPLEXITY

ROTORCRAFT SHELL PROVIDES NUMBER OF PARAMETERS THAT CONTROL MODEL COMPLEXITY

low complexity	high complexity
airframe structure	
one rotor in wind tunnel	rotorcraft in free flight
constant rotational speed	drive train model
no elastic modes	airframe elastic modes
airframe aerodynamics	
no aerodynamics	airframe aerodynamics
no interference velocity	airframe flow field at rotor
rotor structure	
blade pitch control	swashplate mechanism
no control flexibility	pitch link flexibility
pitch link flexibility	swashplate flexibility
flap hinge	flap and lag hinges, pitch bearing
rigid blade elements	elastic blade elements
few elastic elements	many elastic blade elements
few degrees of freedom	many elastic degrees of freedom
rigid wing	general wing
rotor aerodynamics	
no aerodynamics	rotor blade aerodynamics
few blade panels	many aerodynamic panels
static stall	dynamic stall
rotor wake	
uniform inflow	nonuniform inflow
rigid or prescribed geometry	free wake geometry
single-peak wake model	dual-peak wake model
few wake spirals	many wake spirals

7–4 Trim Parameters

REFERENCE: CAMRAD II Documentation, Volume VI, Rotorcraft Input; Chapters "Class = TRIM" and "Class = TRIM ROTOR"

TABLE SUMMARIZES PARAMETERS CONTROLLING CONVERGENCE AND ACCURACY OF TRIM SOLUTION

and default values in analysis

loop	iterations	tolerance	relaxation	perturbation	parameters
wake	ITERP=1		RELAXP=1.		LEVEL=2*1
	ITERF=1		RELAXF=1.		
trim	ITERT=40	TOLERT=1.	RELAXT=.5	DELTA=1.	OPPID=1
					MPID=40
					OPRID=0
					ALPHA=.5
					initial conditions
circulation	ITERC=60	TOLERC=1.	RELAXC=.1		
	ITERH=60	TOLERH=1.	RELAXH=.1		
motion	ITERM=40	TOLERM=2.	RELAXM=1.		

part	iterations	tolerance	relaxation	perturbation	parameters
all					MPSI=MPSIH=24
					MPSIAV=4
					OPHRMP=2
rotor	MREVR=40	TOLERR=2.	RELAXR=1.	DELTAR=.01	MHARMR=10
					DAMPR=.5
					OPUPDT=1101
					NUPDTR=4
					OPPART=1
					METHOD=1
					OPMODE=0
airframe	MREVA=40	TOLERA=2.	RELAXA=1.	DELTAA=.01	MHARMA=10
					DAMPA=.5
drive train	MREVD=40	TOLERD=2.	RELAXD=1.	DELTAD=.01	MHARMD=10
					DAMPD=.5

7–5 Transient Parameters

REFERENCE: CAMRAD II Documentation, Volume VI, Rotorcraft Input; Chapters “Class = TRANSIENT” and “Class = TRANSIENT ROTOR”

TABLE SUMMARIZES PARAMETERS CONTROLLING CONVERGENCE AND ACCURACY OF TRANSIENT SOLUTION

and default values in analysis

part	iterations	tolerance	relaxation	perturbation	parameters
integration					TRESP=.02
					OPUPDT=101
					MPSIAV=4
					METHOD=1
					ALPHA=-.05
					OPINIT=1
rotorcraft	MITERS=100	TOLERS=2.	RELAXS=1.,.5	DELTAS=.01	TSTEPS=.01
					DAMPS=0.
					OPAVS=1
					NUPDTS=4

7–6 Flutter Parameters

REFERENCE: CAMRAD II Documentation, Volume VI, Rotorcraft Input; Chapters “Class = FLUTTER” and “Class = FLUTTER ROTOR”

TABLE SUMMARIZES PARAMETERS CONTROLLING CONVERGENCE AND ACCURACY OF FLUTTER SOLUTION

and default values in analysis

loop or part	perturbation	parameters
loops		MPSIAV=12
rotor	DELTAR=.01	OPMODE=0
airframe	DELTAA=.01	
drive train	DELTAD=.01	

7–7 CAMRAD II Experience

7-7.1 Convergence and Accuracy of Trim Loops

GENERAL

if analysis converges:

circulation loop and parts require many iterations first time they are executed, but few iterations during convergence of outer loops

if analysis does not converge:

inner loops always run to maximum number of iterations

so symptom of divergence is job taking much more computational time than expected

TRIM

relaxation factor RELAXT

hover, trim thrust with collective: 0.7 usually good

otherwise 0.25 may be needed

tolerance TOLERT

sometimes need smaller value than default (1.), in order to obtain accurate propulsive force and parasite power for trim convergence, may need one or more of following parameter changes:

OPPID = 2
DELTA = -1.
OPRID = 1 and ALPHA = 0.9
MPID = 5
operating condition sweep in multicase job

second-order difference (OPPID = 2) often very effective

initial conditions

good initial conditions may be required for convergence (especially in forward flight, or with elastic blade)

and more efficient (fewer trim iterations needed)

needed for uniform inflow solution (which then initializes the nonuniform inflow level)

can obtain by running uniform inflow cases first

multicase jobs with OPINIT = 1 or 7

then run nonuniform inflow, with good initial conditions for uniform inflow level with new configuration, must obtain control setting for uniform inflow trim

use simplified model, to get approximate results with minimum computational time

some of simplifications to consider:

uniform inflow trim

trim input: LEVEL = 1, RELAXC = 0.5

few azimuth stations, few harmonics

trim input: MPSI = 8, MHARMR = 1, MHARMA = 0

two blades modes or no blade elasticity

trim rotor input: DOFM = 2*1,38*2, DOFB = 3*1,9*0

simple blade structure

rotor structure input: OPBEAM = 0, KNODE = 0, LOCKPL = 0

few aerodynamic panels

rotor aerodynamics input: NPANEL = 10, REDGE = .12,.28,.42,.54,.64,.73,.81,.87,.92,.96,1.

or 5 panels, or even 2 panels for fast (but approximate) result

resulting control settings can be used to start trim calculations without these simplifications

CIRCULATION

- relaxation factor RELAXC
  - uniform inflow: 0.5 usually good
  - nonuniform inflow: need 0.10 (default) or 0.05
- number of iterations ITERC
  - 20 will be faster (if trim converges, and circulation is converged at end of solution)
- tolerance TOLERC (default value 1.0)
  - windmilling rotor: 0.05 needed (uniform inflow)
  - vibratory hub loads calculations: may need smaller tolerance to improve consistency and repeatability of results (nonuniform inflow)
    - more sensitive than performance calculations to convergence of aerodynamics
    - for example, TOLERC = 0.2 and ITERC = 200, instead of default TOLERC = 1.0
  - ducted fan: 0.01 needed (and RELAXC = 0.1)
- if large number of iterations (say several hundred) are needed, it may be symptom of bad wake model
- circulation convergence can be essential to convergence of trim iteration
- erratic trim iteration can be symptom of insufficient circulation convergence

MOTION

rotor part or motion loop convergence can be more difficult with airframe degrees of freedom in the trim solution (for vibration calculation)

relaxation factor RELAXM: may need value less than 1.0

or run multicase job with OPINIT = 7

vary parameters that control vibration level

for example, vary airframe moments of inertia and elastic mode generalized masses, from very large down to actual values

CONVERGENCE WITH ELASTIC BLADE IN FORWARD FLIGHT

may need one or more of following parameter changes for convergence of trim loops and parts

	parameter	default value	for convergence
trim	RELAXT	0.5	0.2
circulation	RELAXC	0.1	0.05
	ITERC	60	200
rotor	RELAXR	1.0	0.5–0.3
	DAMPR	0.5	1.0
	blade modes	no	yes

WHEN CONVERGENCE PROBLEMS ARE ENCOUNTERED, ALWAYS TRY SMALLER RELAXATION FACTORS FIRST

use of trim blade modes and RELAXR = 0.5–0.3 often required for convergence with elastic blade in forward flight

blade modes help by reducing order (neglect very high frequency modes, quasistatic solution for moderate frequency modes)

and by allowing introduction of modal damping GDAMPM

if eventually converges, faster with smaller ITERC (say 20)

default for uniform inflow (RELAXC = 0.5) often satisfactory (RELAXC = 0.1 is default for nonuniform inflow level)

smaller tolerances (TOLERC and TOLERR) may be needed to improve convergence of outer loop convergence can also be affected by strategy for update of reference and matrices for differential equation solutions

> for example, OPUPDT(3) = 1 and NUPDTR = 10 might help a case where the rotor part solution diverges after many iterations of the circulation loop

rotor part solution usually require linear interpolation to evaluate harmonics

> OPHRMP = 2 (default), not Fourier interpolation

can use time domain finite element solution method for rotor part, instead of harmonic solution method

> METHOD = 2, with blade modes and RELAXR = 0.5–0.2; and perhaps smaller RELAXN (set by core input)

initialization of response by simpler previous case (using multicase job with OPINIT = 6 or 7) often effective for very difficult convergence problems

WITH ELASTIC BLADE IN HOVER, TYPICALLY CAN USE DEFAULT VALUES

7-7.2 Structural Dynamics of Rotor Blade

NUMERICAL PROBLEMS CAN BE ENCOUNTERED WHEN USING BLADE MODES WITH QUASISTATIC ELIMINATION OF HIGH FREQUENCY MODES

GENERAL SYMPTOMS:

negative eigenvalues of modal mass or spring matrix ;
perhaps singular spring matrix of quasistatic modes (KLL)

SYMPTOMS IN TRIM TASK: divergence

SYMPTOMS IN FLUTTER TASK: erratic or incorrect eigenvalues

MODAL ANALYSIS SET UP SUCH THAT LOW FREQUENCY MODES ARE MOST ACCURATE

theoretical approach allows good results to be obtained from high frequency modes, as long as they are not so inaccurate that information is lost in the modal transformation

CAN AVOID THESE NUMERICAL PROBLEMS BY NEGLECTING THE HIGH FREQUENCY MODES

use DOFM = 0 instead of DOFM = 2

in trim rotor, transient rotor, and flutter rotor input as required

may still need quasistatic response for moderate frequency modes, in order to retain control response

TRIM SOLUTION MAY CONVERGE WITHOUT USING BLADE MODES

avoid these issues, as well as any question of accuracy with modal truncation; but can not use modal damping without modes

7-7.3 Selection of Modes for Elastic Rotor Blade

WHEN ANALYZING ELASTIC BLADE, NUMBER OF BLADE MODES REQUIRED FOR ACCURACY SHOULD BE DETERMINED FOR ANY NEW PROBLEM

often blade modes needed to improve convergence

then DOFM = 1 (dynamic), 2 (quasistatic), or 0 (neglect) must be specified

USE OF QUASISTATIC MODES CAN AFFECT TRIM AND PERFORMANCE SOLUTION

since the blade is rotating, quasistatic approximation can neglect inertial and Coriolis loads that would cancel spring and centrifugal loads in rotor total mean loads

locked pitch bearing (LOCKP = 0) means a quasistatic pitch degree of freedom, if blade modes are not used

FREE FLIGHT TRIM:

quasistatic approximation can lead to hub force $\neq$ rotor aerodynamic propulsive force (plus blade weight term)

then trim loop convergence (hub force $=$ airframe drag) does not give rotor propulsive force $=$ airframe drag

WIND TUNNEL TRIM:

even with rotor aerodynamic thrust and/or propulsive force trimmed to specified values, rotor power (from torque at hub node) can be wrong

then profile power ($P_o = P - P_i - P_p$) also wrong, and very different from P_o calculated from blade section forces

PROBLEM ELIMINATED BY USING MORE DYNAMIC MODES

WITH CONTROLLED PITCH JOINT, PITCH MODE MAY BE AT HIGH FREQUENCY

- especially with locked pitch joint (rotorcraft shell creates large pitch spring)

- if neglect pitch mode, control loads will be large; and beam elements inboard of pitch bearing may twist, in place of pitch joint motion

  - check pitch motion in mode shapes used (flutter task)

  - check trim pitch joint motion using position sensor

- NEED ENOUGH MODES (MAYBE QUASISTATIC) TO CAPTURE CONTROL INPUT

  - with elastic blade model, can use realistic pitch spring and unlocked pitch joint

ACCURATE BLADE LOADS CALCULATION MAY REQUIRE MANY MODES

- neglecting modes may make structure appear too stiff, and hence blade loads too high

- particularly with multiple load path

- example: stiff pitch case of bearingless rotor

- PROBLEM ELIMINATED BY USING MORE MODES (MAYBE QUASISTATIC)

7-7.4 Bearingless Rotors

STRUCTURAL DYNAMIC CHARACTERISTICS OF BEARINGLESS ROTOR BLADE

MANY ELASTIC VARIABLES WITH SMALL INERTIA (PARTICULARLY IN FLEXBEAM)

mass matrix is poorly-conditioned or nearly singular

so numerical problems may be encountered in calculating eigenvalues for flutter analysis

STATIC REDUCTION APPROPRIATE

can use modal transform and modal truncation (zero or quasistatic) to implement static reduction

calculating modal transform itself requires eigenanalysis

FREE VIBRATION CALCULATIONS

use to check physical and numerical aspects of structural dynamics

PARAMETERS THAT MAY IMPROVE NUMERICAL CHARACTERISTICS

neglect high frequency modes (DOFM = 0 not 2)

NDOFU = 1 for flexbeam; consistent with constant tension force along element length, which nearly true with small flexbeam mass

flexbeam MASS and ITHETA not too small

NINTEG large enough for accurate integration

CONSTRAINT PROVIDED BY PITCH LINK ELIMINATES A TORSION DEGREE OF FREEDOM

structural dynamics poorly-conditioned if choose wrong torsion degree of freedom to eliminate

shell identifies torsion degree of freedom of softest flexbeam element (largest $\int_0^\ell dx/GJ$)

user should define flexbeam nodes so shell finds correct element (or use core input to specify correct element)

TYPICAL MODEL FOR STRUCTURAL DYNAMIC CALCULATIONS

BLADE ELEMENTS

1 element for cuff or torque tube (inboard of EFB)

4 elements for blade (outboard of EFB)

standard elastic degrees of freedom (3222)

FLEXBEAM ELEMENTS

2 elastic elements

plus rigid elements at hub and blade attachment

standard elastic degrees of freedom (3222 or 1222)

FEWER BLADE AND FLEXBEAM ELASTIC ELEMENTS MAY GIVE ACCURATE RESULTS FOR FUNDAMENTAL FLAP AND LAG MODES

7-7.5 Rotor Wake Model

CONSIDER DUAL-PEAK MODELS OF FAR WAKE ROLLUP (OPFW = 1 OR 2)

- CAN HAVE LOADING DISTRIBUTION WHERE BLADE LIFT IS NEGATIVE JUST ON LAST AERODYNAMIC PANEL
  - depending on blade twist and operating condition
  - then not appropriate to assume that tip vortex strength is determined by outboard peak
  - and calculations can be sensitive to very small changes in tip loading
- CAN HAVE LOCALIZED NEGATIVE LOADING PRODUCED BY STRONG BLADE-VORTEX INTERACTIONS

DUAL-PEAK MODEL NOT APPROPRIATE IN SUCH CASES

- can produce convergence problems for circulation loop
- can produce poor performance calculations

CONTROL IDENTIFICATION OF DUAL-PEAK LOADING BY LIMITING SEARCH FOR CIRCULATION PEAKS

- use RGMAX = 0.95 or so

BEST FOR USER TO CHOOSE WAKE MODEL

- make initial runs with OPFW = 0 (single-peak model using maximum circulation)
- examine each job for extensive negative tip loading, and rerun with OPFW = 2 (dual-peak model) as required

7-7.6 Geometry Definition for Structural Dynamic Components

CAN CREATE NEW COMPONENTS AND INTERFACES USING CORE INPUT

MAY ENCOUNTER ERROR MESSAGE:

matrix singular
part differential equations
inverting PHIX0B

MAY BE CAUSED BY 180 DEGREE MISMATCH OF CONNECTION AXES (REST POSITION) ON TWO SIDES OF STRUCTURAL DYNAMIC INTERFACE

constraint equation singular since constructed using Rodrigues parameters

USE INPUT PROGRAM TO DRAW AXES AT STRUCTURAL DYNAMIC INTERFACES

shows rest position of components

7–8 CAMRAD II Projects

TYPICAL CAMRAD II PROJECT

DEFINE BASELINE INPUT FOR A ROTORCRAFT

SERIES OF CAMRAD II JOBS, VARYING:

operating condition
analysis parameters
rotorcraft parameters

BEGIN WITH CORRELATION: CALCULATIONS FOR SOMETHING THAT HAS ALREADY BEEN TESTED

determine how to use analysis to perform required tasks

define empirical models, establish analysis complexity

THEN PREDICTION: CALCULATIONS FOR NEW DESIGN

must be aware of limits

analysis limitations determined by correlation

new designs may go beyond correlation

near limits: use analysis to establish sensitivity

past limits: need new tests (for correlation), perhaps new analysis methods

THEN VERIFICATION: CHECK PREDICTIONS AFTER TEST NEW ROTORCRAFT

usually develops into new correlation activity

FUNDAMENTAL APPROACH TO DEVELOPING CAMRAD II CALCULATION:

START SIMPLE, SMALL, FAST

BUILD UP TO COMPLEX, LARGE, SLOW

try new task or new configuration as soon as identify requirement

so have time to work on problems encounter

ASSISTANCE AVAILABLE FROM MAINTENANCE/SUPPORT CONTRACTOR

ORGANIZE THE PROJECT

SET UP SYSTEM TO ORGANIZE ALL INPUT, JOB, OUTPUT FILES

need conventions for directories, file names, file extensions

airfoil and input files created by INPUT program have unique identification (time/date)

AIDS TO ORGANIZATION

USE NAMELIST FILES OF BASELINE PARAMETERS FOR EACH ROTOR OR ROTORCRAFT

- maintain standard files for organization
- use these namelist files to generate shell input file, for configuration control of input data and to avoid long job namelists
- keep namelist files readable
  - several scalars per line; matrices and arrays in appropriate block format
- annotate namelist files
- document changes (leave superseded data in file)

USE CODE PARAMETER TO IDENTIFY CASES

- every case has unique identification (time/date)

KEEP INPUT AND JOBS FOR OLD PROJECTS

- computer files (backed up) and paper documentation, sufficient to duplicate work several years later

STEPS IN DEVELOPING CAMRAD II CALCULATION

(1) PREPARE INPUT

template available for preparation of input data
(file `ztemplate.list`)

previous work or sample jobs provide starting point

CHECK INPUT

(2) DRAW GEOMETRY

script available to extract data needed to draw planform
(file `zplanform.com`, runs INPUT program)

draw 3D geometry

CHECK GEOMETRY

(3) FREE VIBRATION CALCULATIONS

flutter analysis, no aerodynamics

- first independent blade
  then entire rotor, drive train, etc

typical job for blade frequency calculation available
(file `zfrequency.list`)

CHECK PHYSICAL AND NUMERICAL ASPECTS OF STRUCTURAL DYNAMICS

- check frequencies
- establish CAMRAD II model for accurate blade frequencies (with fewest possible elastic elements)

(4) RUN BASIC CASE

start with no trim, uniform inflow

perhaps simplified model, or wind tunnel case
(files `zsimple.list` and `zwindtunnel.list`)

hover then forward flight (file `zhover.list`)

(5) DETERMINE REQUIRED MODEL

FOR TYPICAL CASE, TO ESTABLISH PARAMETERS VALUES FOR CONVERGENCE AND ACCURACY

investigate input parameters for which value or importance is not clear

turn things on to check assumptions about what is important

such as nonuniform inflow, free wake, degrees of freedom

turn things off for efficiency

use minimum model possible

vary tolerances, wake loop level and number of wake iterations to establish accuracy

based on engineering judgement regarding answer of interest

BUILD UP COMPLEXITY OF JOB

add trim, nonuniform inflow, elastic motion as appropriate

establish parameters required to achieve convergence

(6) UNIFORM INFLOW FOR REQUIRED OPERATING CONDITIONS

establish initial conditions for trim loop

(7) RUN JOBS OF PROJECT

multicase jobs, varying:

operating condition
analysis parameters
rotorcraft parameters

CONVERGENCE PROBLEMS ARE OFTEN ENCOUNTERED

for new problem or new configuration, analysis may not work at first

engineer must help the solution procedure

ASSISTANCE AVAILABLE FROM MAINTENANCE/SUPPORT CONTRACTOR

```
!===========================================================================================================================
! This file provides a template for preparation of the input for CAMRAD II.
! In particular, it can be used to convert CAMRAD/JA input data.
!===========================================================================================================================
&NLDEF class='CASE',&END
&NLVAL
    TITLE='xxx',CODE='xxx',                                 ! description
    OPUNIT=1,DENSE=.00237689,TEMP=59.,                      ! environment (English units)
    OPUNIT=2,DENSE=1.225,TEMP=15.,                          ! environment (SI units)
    OPDENS=3,                                               ! OPDENS=3 for density and temperature
&END
!===========================================================================================================================
&NLDEF class='TRIM',&END
&NLVAL
    VELIN=1,WINDIN=1,                                       ! operating condition (1 knots, 2 V/Omega*R)
    VTIPIN=2,VTIP=#,RPM=#,                                  ! operating condition (1 tip speed, 2 rpm)
    LEVEL=2*1,                                              ! wake loop (uniform inflow)
    LEVEL=2*2,                                              ! wake loop (prescribed geometry; ITERP=2 for dual peak)
    LEVEL=2*3,                                              ! wake loop (free geometry; ITERF=2 for dual peak)
    OPTRIM=0,COLL=10.,                                      ! no trim
    COLL=10.,MTRIM=6,                                       ! free flight trim
    MNAME='FORCE X ','FORCE Y ','FORCE Z ',
          'MOMENT X','MOMENT Y','MOMENT Z',
    VNAME='COLL    ','LATCYC  ','LNGCYC  ',
          'PEDAL   ','PITCH   ','ROLL    ',
    COLL=10.,CTTRIM=.08,MTRIM=3,                            ! wind tunnel trim
    MNAME='CT/S    ','BETAS   ','BETAC   ',                 !   CAMRAD/JA:  261 = T,bs,bc with coll,lat,lng;  244 = T with coll
    VNAME='COLL    ','LATCYC  ','LNGCYC  ',
    MHARMR=2*#,MHARMA=2*#,MHARMD=2*#,                       ! harmonics
    DOFA=6*#,DOFM=#,DOFD=8*#,                               ! degrees of freedom
    MSSEN=1,                                                ! airframe sensors
&END
&NLDEF class='TRIM ROTOR',name='ROTOR n',&END
&NLVAL
    OPMODE=0,DOFB=12*#,                                     ! no blade modes
    OPMODE=1,DOFM=40*#,GDAMPM=40*#,                         ! with blade modes
    MHSEN=1,MCSEN=1,MBSEN=1,MPSEN=1,MASEN=1,MWSEN=1,        ! rotor output
&END
!===========================================================================================================================
&NLDEF class='TRANSIENT',&END
&NLVAL
    TIMEB=#,TIMEE=#,                                        ! time range
    OPHIST=#,TIMEZ=#,                                       ! prescribed control
    METHOD=1,OPINIT=1,                                      ! integration method
    TRESP=#,TSTEPS=#,                                       ! time step
    DOFA=6*#,DOFM=#,DOFD=8*#,                               ! degrees of freedom
    MSSEN=1,                                                ! airframe sensors
&END
&NLDEF class='TRANSIENT ROTOR',name='ROTOR n',&END
&NLVAL
    OPWAKE=2,OPGEOM=2,OPVATR=1,OPVRTA=1,DOFL=3*1,           ! nonuniform inflow
    OPWAKE=3,OPGEOM=2,OPVATR=1,OPVRTA=1,DOFL=3*1,           ! dynamic inflow
    OPWAKE=4,OPGEOM=2,OPVATR=2,OPVRTA=2,DOFL=3*1,           ! trim inflow
    OPMODE=0,DOFB=12*#,                                     ! no blade modes
    OPMODE=1,DOFM=40*#,GDAMPM=40*#,                         ! with blade modes
    MHSEN=1,MCSEN=1,MBSEN=1,MPSEN=1,MASEN=1,MWSEN=1,        ! rotor output
&END
```

```
!==============================================================================================================================
&NLDEF class='FLUTTER',&END
&NLVAL
    OPFLUT=1,OPMEAN=0,                                       ! Floquet theory, periodic coefficients
    OPFLUT=0,OPMEAN=1,                                       ! time invariant, averaged coefficients
    OPSTAB=1,OPMEAN=1,NRPRNT=2*1,NAPRNT=1,NDPRNT=1,          ! flight dynamics
    DOFA=6*#,DOFM=#,DOFD=8*#,                                ! degrees of freedom
    CONP=5*#,GUST=3*#,                                       ! controls and gust
    MSSEN=1,                                                 ! airframe sensors
    NOUT=#,KINDO=#,NAMEC='xxx',NAMEV='xxx',NAMEE='xxx',      ! variables as output
&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR n',&END
&NLVAL
    OPWAKE=3,OPVATR=1,OPVRTA=1,DOFL=3*1,                     ! dynamic inflow
    OPWAKE=4,OPVATR=2,OPVRTA=2,DOFL=3*1,                     ! trim inflow
    OPMODE=0,DOFB=12*#,                                      ! no blade modes
    OPMODE=1,DOFM=40*#,GDAMPM=40*#,                          ! with blade modes
    MHSEN=1,MCSEN=1,MBSEN=1,MPSEN=1,MASEN=1,                 ! rotor sensors
&END
!==============================================================================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    TITLE='xxx',                                             ! description
    CONFIG=0,OPFREE=0,OPAERO=0,                              ! wind tunnel
    CONFIG=1,RGEAR(2)=#,                                     ! single mr/tr
    CONFIG=2,                                                ! tandem
    CONFIG=3,                                                ! coaxial
    CONFIG=4,ATILT=#,                                        ! tiltrotor
    WEIGHT=#,IXX=#,IYY=#,IZZ=#,IXY=#,IXZ=#,IYZ=#,            ! inertia
    MASSR=2*#,
    FSCG=#,BLCG=#,WLCG=#,                                    ! geometry
    FSRTR=2*#,BLRTR=2*#,WLRTR=2*#,ASHAFT=2*#,ACANT=2*#,
    FSWB=#,BLWB=#,WLWB=#,
    FSHT=#,BLHT=#,WLHT=#,FSVT=#,BLVT=#,WLVT=#,
    FSPIV=#,BLPIV=#,WLPIV=#,ASPIV=#,ADPIV=#,                 ! tiltrotor
&END
&NLDEF class='AIRFRAME',type='AERODYNAMICS',&END             ! CAMRAD/JA:  IWB=>IWBL, IWBM=IWB+57.3*MOM0W/MOMAW
&NLVAL                                                       !    SIDEA=>SIDEDA,ROLLA=>ROLLDA,YAWA=>YAWDA,
&END                                                         !    IHT=>IHTL; IVT=>IVTL; FETAIL=>1/EHTAIL
&NLDEF class='AIRFRAME',type='CONTROL',&END
&NLVAL &END
&NLDEF class='AIRFRAME',type='DRIVE TRAIN',&END
&NLVAL &END
!==============================================================================================================================
```

```
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR n',&END
&NLVAL
    TITLE='xxx',                                         ! description
    RADIUS=#,NBLADE=#,ROTATE=#,SIGMA=#,
    VTIPN=#,
    CONTRL=0,HINGE=3,PITCH=2,                            ! blade root (flap and lag hinges)
    CONTRL=0,HINGE=0,PITCH=2,                            ! blade root (hingeless)
    CONE=#,EFLAP=#,ELAG=#,EPITCH=#,
    DROOP=#,SWEEP=#,EDS=#,
    DLAG=#,KPITCH=#,LOCKP=#,                             ! with PITCH=2, use LOCKP=0 or KPITCH=# (not DOFB=0)
    OPBEAM=2,DRELST=.04,KNODE=2,RNODE=.28,.59,           ! elastic blade analysis (align RNODE with REDGE)
    OPBEAM=0,KNODE=0,OPWING=0,                           ! rigid blade analysis
    GDAMPU=.01,GDAMPV=.01,GDAMPW=.01,GDAMPT=.01,
    TWIN=1,TWISTL=#,                                     ! linear twist
    NPROP=51,                                            ! section properties
    RPROP=0.,.02,.04,.06,.08,.10,.12,.14,.16,.18,.20,
             .22,.24,.26,.28,.30,.32,.34,.36,.38,.40,
             .42,.44,.46,.48,.50,.52,.54,.56,.58,.60,
             .62,.64,.66,.68,.70,.72,.74,.76,.78,.80,
             .82,.84,.86,.88,.90,.92,.94,.96,.98,1.0,
    ZEA=51*0.,ZQC=51*0.,ZC=51*0.,ZI=51*0.,
    XEA=51*0.,XQC=51*0.,XC=51*0.,XI=51*0.,
    TWISTA=51*#,THETAC=51*#,THETAI=51*#,                 ! TWISTA at inertial stations, not aerodynamic stations
    KP=51*#,KT=51*#,                                     ! sqrt(Itheta/mR**2);  CAMRAD/JA:  KP=KT=sqrt(KP2)
    EA=51*#,                                             ! EI(m/Itheta) = EI/(R*KP)**2
    EIFLAP=51*#,EILAG=51*#,GJ=51*#,
    MASS=51*#,ITHETA=51*#,IPOLAR=51*#,
&END
!==================================================================================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR n',&END
&NLVAL
    NPANEL=20,                                           ! aerodynamic panels
    REDGE=.12,.20,.28,.35,.42,.48,.54,.59,.64,.69,
          .73,.77,.81,.84,.87,.90,.92,.94,.96,.98,1.,
    NPROP=#,                                             ! aero properties (0, panel midpoints, 1)
    RPROP=0.,#,1.,
    CHORD=#,ASWEEP=#,                                    ! chord in ft or m
    NSEN=5,OPREF=5*4,                                    ! aerodynamic sensors
    QUANT= 5,25,35,82,82,                                !   lambda
    IDENT= 1, 0, 0, 0, 0,                                !   alpha,theta
    AXIS=  3, 0, 0, 1, 3,                                !   Fx,Fz
    OPSCL= 2, 1, 1, 2, 2,                                !
    NAPLOT=1, 4, 1, 1, 1,                                !
&END
!==================================================================================================================================
```

```
&NLDEF class='ROTOR',type='INFLOW',name='ROTOR n',&END
&NLVAL &END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR n',&END
&NLVAL
    OPSCEN=1,                                             ! forward flight wake
    OPSCEN=2,TWIST=#,RICWG=#,                             ! hover wake
&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR n',&END
&NLVAL
    OPSCEN=0,                                             ! turn off scenario
    OPFW=2,                                               ! dual peak wake model
    RFW=3.,MFWG=3,                                        ! wake extent
&END
!==================================================================================================================================
&NLDEF class='TABLES',&END
&NLVAL
    AFTABL=0,                                             ! same table all rotors
    AFTABL=1,                                             ! separate tables
&END
!==================================================================================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
!
!
!==================================================================================================================================
! swashplate control and kinematics (shell input)
! CAMRAD/JA: parameters xph,rph,phiph,phipl
!       XSP =  XPH + HSP/R tan(phipl)        XPH = XTO + xph cos(phiph)
!       YSP =  YPH                           YPH = rph
!       ZSP = -HSP/R                         ZPH = ZUS + rph tan(CONE) - xph sin(phiph)
!       KPL = KPITCH/(xph * R)**2
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    HSP=#,OPSPM=0,                                        ! swashplate node
&END
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR n',&END
&NLVAL
    CONTRL=2,PITCH=1,KPITCH=0.,LOCKP=1,                   ! control system
    XSP=#,YSP=#,ZSP=#,XPH=#,YPH=#,ZPH=#,EPH=#,
    KPL=#,LOCKPL=#,LOCKSP=0,
&END
!==================================================================================================================================
!
!
!==================================================================================================================================
! hydraulic lag damper (core input; see Training Manual, Volume VII, Chapter 9)
! identify component (n,m,k), and spring/damper for lag joint (j)
&NLDEF class='COMPONENT',type='BEAM',name='ROTOR n BLADE m ELEMENT k',&END
&NLVAL
    CTYPE(j)=3,CLIN(j)=0.,                                ! linear + hydraulic damper
    NCHYDA(j)=0,CHYDA(1,j)=x,                             ! lag moment M = min(x,y*r**2), r = lag rate
    NCHYDB(j)=2,CHYDB(1,j)=0.,0.,y,                       ! CAMRAD/JA:  x=LDAMPM, y=LDAMPM/LDAMPR**2
&END
!==================================================================================================================================
```

```
file: zfrequency.list
!===============================================================================
! blade frequency calculation
!===============================================================================
&NLJOB &END
!===============================================================================
&NLDEF class='CASE',&END
&NLVAL FLTASK=1,CODE='FREQUENCY',&END                       ! flutter task
&NLDEF class='TRIM',&END
&NLVAL
    VTIPIN=1,VTIP=#,COLL=#,                                 ! operating condition
    VTIPIN=2,RPM=#,COLL=#,                                  ! operating condition
    LEVEL=2*1,OPTRIM=0,                                     ! no trim, no wake
    MHARMR=2*0,MHARMA=2*0,MHARMD=2*0,                       ! harmonics
    MPSI=4,MPSIAV=1,                                        ! axisymmetric
&END
&NLDEF class='TRIM ROTOR',name='ROTOR n',&END
&NLVAL
    OPMODE=0,DOFB=12*1,                                     ! no blade modes
    OPMODE=1,DOFM=40*#,GDAMPM=40*#,                         ! with blade modes
    MPSEN=1,                                                ! blade position
&END
!===============================================================================
&NLDEF class='FLUTTER',&END
&NLVAL
    OPBLD=1,                                                ! independent blade
    MPSIAV=1,                                               ! no average
&END
&NLDEF class='FLUTTER ROTOR',name='ROTOR n',&END
&NLVAL
    OPMODE=1,DOFM=10*1,30*0,GDAMPM=40*#,                    ! blade modes
&END
!===============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL OPAERO=0,&END                                        ! no aerodynamics
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR n',&END
&NLVAL OPAERO=0,&END                                        ! no aerodynamics
!===============================================================================
&NLDEF action='end of shell',&END
&NLDEF action='end of core',&END
```

```
file: zhover.list
!==============================================================================
! hover analysis
!==============================================================================
&NLDEF class='TRIM',&END
&NLVAL
    LEVEL=2*1,                                              ! uniform inflow
    LEVEL=2*2,                                              ! nonuniform inflow
    CTTRIM=.08,MTRIM=1,                                     ! wind tunnel trim
       MNAME='CT/S    ',VNAME='COLL    ',
       COLL=10.,LNGCYC=0.,LATCYC=0.,
    TOLERT=1.,RELAXT=.7,                                    ! trim loop
    MHARMR=2*0,MHARMA=2*0,MHARMD=2*0,MPSIAV=1,              ! harmonics
    MPSI=4,                                                 ! uniform inflow only
    NxPRNT=1,NxFILE=1,MxTIME=1,                             ! output
&END
&NLDEF class='TRIM ROTOR',name='ROTOR n',&END
&NLVAL
    OPMODE=1,DOFM=40*#,GDAMPM=40*#,                         ! with blade modes
    OPMODE=0,DOFB=12*#,                                     ! no blade modes
    NxPRNT=1,NxFILE=1,MxTIME=1,                             ! output
&END
!==============================================================================
&NLDEF class='FLUTTER',&END
&NLVAL MPSIAV=1,&END                                        ! no average of eqns
!==============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL OPAERO=0,&END                                        ! no aerodynamics
!==============================================================================
!==============================================================================
! hover free wake
&NLDEF class='TRIM',&END
&NLVAL
    LEVEL=3,NWPRNT=1,                                       ! free wake geometry
    TOLERC=.2,ITERF=4,RELAXF=.5,                            ! wake convergence
    MPSI=24,                                                ! nonuniform inflow
&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR n',action='init',&END
&NLVAL
    OPSCEN=2,TWIST=#,RICWG=#,                               ! hover scenario
&END
&NLDEF class='ROTOR',type='WAKE',name='ROTOR n',&END
&NLVAL
    OPSCEN=0,                                               ! turn off scenario
    OPFWG=3,                                                ! general method
  ! OPMCRC=0,                                               ! mean circulation
&END
!==============================================================================
```

```
file: zsimple.list
!==============================================================================
! simplified models
!==============================================================================
&NLDEF class='TRIM',&END
&NLVAL
    LEVEL=1,RELAXC=.5,                                        ! uniform inflow
    OPTRIM=0,                                                 ! no trim
    MPSI=8,                                                   ! few azimuth stations
    MHARMR=1,MHARMA=1,MHARMD=1,                               ! few harmonics
&END
!==============================================================================
&NLDEF class='TRIM ROTOR',name='ROTOR n',&END
&NLVAL
    DOFM=2*1,38*0,DOFB=3*1,9*0,                               ! degrees of freedom
&END
!==============================================================================
&NLDEF class='ROTOR',type='STRUCTURE',name='ROTOR n',&END
&NLVAL
    OPBEAM=0,KNODE=0,                                         ! rigid blade
    OPWING=0,                                                 ! rigid wing
    LOCKPL=0,                                                 ! locked pitch link
&END
!==============================================================================
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR n',&END
&NLVAL NPANEL=10,                                             ! ten wing panels
    REDGE=.12,.28,.42,.54,.64,.73,.81,.87,.92,.96,1.,
&END
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR n',&END
&NLVAL NPANEL=5,                                              ! five wing panels
    REDGE=.12,.42,.64,.81,.92,1.,
&END
&NLDEF class='ROTOR',type='AERODYNAMICS',name='ROTOR n',&END
&NLVAL NPANEL=2,                                              ! two wing panels
    REDGE=.3,.7,1.,
&END
!==============================================================================
```

```
file: zwindtunnel.list
!==============================================================================
! conversion of free flight case to single rotor in wind tunnel
!==============================================================================
&NLDEF class='TRIM',&END
&NLVAL
    WINDIN=1,WKTS=#,                                          ! operating condition
    COLL=#,CTTRIM=#,MTRIM=3,                                  ! wind tunnel trim
    MNAME='CT/S    ','BETAS   ','BETAC   ',
    VNAME='COLL    ','LATCYC  ','LNGCYC  ',
&END
!==============================================================================
&NLDEF class='AIRFRAME',type='STRUCTURE',&END
&NLVAL
    TITLE='WIND TUNNEL SUPPORT',                              ! description
    CONFIG=0,OPFREE=0,OPAERO=0,                               ! wind tunnel
    OPTRAN=0,                                                 ! no drive train
    ASHAFT=0.,ACANT=0.,                                       ! geometry
&END
!==============================================================================
```

WHEN PROBLEMS ARE ENCOUNTERED:

FOLLOW STEPS RECOMMENDED ABOVE FOR DEVELOPING CAMRAD II CALCULATION

REVIEW GUIDELINES REGARDING CONVERGENCE

TWO WAYS TO ATTACK A PROBLEM:

consider physics and solution procedure, to identify possible source of problem, and hence revised parameters

try changing whatever parameters are available

SIMPLIFY JOB: TASKS, ANALYSIS PARAMETERS, SYSTEM CONFIGURATION

find simplest case such that switching one parameter controls problem

in order to understand what is happening, and hence identify what modifications to make to analysis or input

and to reduce computation time while investigate problem

CAN USE TRACEP and TRACEL PARAMETERS OF CASE INPUT TO GET MORE INFORMATION ABOUT CONVERGENCE

KEEP TRYING

ASSISTANCE AVAILABLE FROM MAINTENANCE/SUPPORT CONTRACTOR

7–9 Checking CAMRAD II Results

DEVELOP HABIT OF CHECKING RESULTS

- NEVER ACCEPT RESULTS OF COMPUTER PROGRAM WITHOUT SCRUTINY
  - usually possible to establish that results are satisfactory for intended purpose
  - must question code in order to recognize cases when results are unreliable
    - in process, develop better understanding of computer program
- CONSIDER LIMITATIONS OF ANALYSIS
  - job can always be done better
  - recognize when <u>must</u> be done better

CHECK AIRFOIL TABLE FILE

PLOT TABLE DATA

graphical display is best way to assess table data

- typographical errors
- interpolation and extrapolation problems
- inadequate resolution

use INPUT program to extract data from unformatted CAMRAD airfoil file

- not necessarily same information as in source files

POSSIBLE PROBLEMS

angle-of-attack and Mach number resolution too small

bad patches between real airfoil data (low angle-of-attack) and generic airfoil data (high angle-of-attack)

attempt to go beyond data in source files (no extrapolation)

spanwise interpolation can give bad characteristics

- stall behavior, and effect of zero lift angle

ANALYSIS MAY USE AIRFOIL TABLE BEYOND RANGE OF GOOD DATA

data at high angle-of-attack often not available for true airfoil (so tables contain generic data for high angle-of-attack, typically for NACA 0012)

no extrapolation beyond maximum Mach number in table (or in original source file)

CHECK OTHER TABLE FILES

CHECK SHELL INPUT FILE

WHEN THINK INPUT IS COMPLETE, RE-EXAMINE ALL NUMBERS

checking entire set of data, and looking for things forgotten

use listing of input parameters (one line per parameter), refer to input manual as required

for each parameter, consider job and ask:

is variable important?
is value correct?

product of this work is changes,
and list of parameters not sure of

for parameter uncertainties:

refer to training and input manuals for recommendations

run CAMRAD II to determine proper values

PLOT INPUT DATA: RADIAL DISTRIBUTION OF BLADE PROPERTIES

graphical display is best way to assess blade data

compare blade properties with original information

use INPUT program to extract data from namelist files or from shell input file

DRAW GEOMETRY

- graphical display is best way to assess geometry of system constructed by rotorcraft shell
- use INPUT program to extract data from namelist files or from shell input file
  - analysis job can be read as a namelist input file, after read shell input file
- if use core input to change system, essential to check definition of geometry
  - including axes at structural dynamic interfaces

FREE VIBRATION CALCULATIONS

- checks physical and numerical aspects of structural dynamics
  - frequencies represent integrated inertia and stiffness
  - often know what calculated frequencies should be

CHECK CAMRAD II ANALYSIS

CHECK INPUT PARAMETERS OF JOB

project requires series of jobs, varying parameters

define reference job, save its output file

print shell input at beginning of each case

so difference between current output and reference job output will list changes in shell input parameters

make sure that input intended to change was in fact changed

THINK ABOUT PHYSICS OF PROBLEM

formulate expectations, and check them

CORRELATE CALCULATED RESULTS WITH EXPERIMENT

perhaps with other analyses

BE SURE OF DEFINITIONS

units, sign conventions, normalization, frames

documented in output

CHECK CONVERGENCE OF EVERY CASE

RESULTS ARE UNRELIABLE IF ANY ITERATION IS NOT CONVERGED

usually each wake level must be converged, since it initializes the next level

search for “`TRIM CONVERGENCE`”
search for “`BOUND CIRCULATION PEAKS`”

CHECK RESULTS FOR REASONABLE ROTOR BEHAVIOR

controls, flapping, rotor angles

search for “`ROTOR n PERFORMANCE`”

performance indices, angle of attack

search for “`PERFORMANCE INDICES`”

induced power factor $\kappa = P/P_m$ will show problems with inflow or wake model

equivalent section drag $c_{do} = 8C_{Po}/\sigma$ will show problems with aerodynamics

angle of attack will show problems with aerodynamic model

radial oscillations can occur with nonuniform inflow

large α at tip station possible with bad RTVTX

figure of merit, propulsive efficiency, rotor L/D will show problems with total performance

plot aerodynamic sensors

polar plot of angle of attack
span plot of bound circulation
time history of section thrust
time history of induced velocity

printer-plot may be adequate to identify problems

wake geometry

```
-------------------------------------
TRIM LOOP ITERATION AND PART SOLUTION
-------------------------------------


SUCCESSIVE SUBSTITUTION ITERATION, LOOP = WAKE
LEVEL   1 = UNIFORM INFLOW

                                                                                                                               LL
    NEWTON RAPHSON ITERATION, LOOP = TRIM                                                                                      LL
    VARIABLES                                                 QUANTITIES                                                       LL
        COLL          LATCYC        LNGCYC                        CT/S          BETAS        BETAC                             LL
    TARGETS                                                        0.80000E-01   0.00000E+00   0.00000E+00                     LL
    BEGINNING OF ITERATION, LOOP = TRIM                                                                                        LL
          9.0000        3.0000        8.0000                       0.39784E-01    1.4387        2.5608                         LL
                                                                                                                               LL
    PERTURBATION IDENTIFICATION, COUNT =    1, LOOP = TRIM                                                                     LL
          8.5000        3.0000        8.0000                       0.30490E-01    1.8366        3.2195                         LL
                                                                                                                               LL
    PERTURBATION IDENTIFICATION, COUNT =    2, LOOP = TRIM                                                                     LL
          9.0000        2.0000        8.0000                       0.39607E-01   0.39735        2.5518                         LL
                                                                                                                               LL
    PERTURBATION IDENTIFICATION, COUNT =    3, LOOP = TRIM                                                                     LL
          9.0000        3.0000        9.0000                       0.31105E-01    1.7844        3.9966                         LL
                                                                                                                               LL
                                                                                                                               LL
         PERTURBATION IDENTIFICATION                                                                                           LL
         DERIVATIVE MATRIX D                                                                                                   LL
                         VARIABLE = COLL          LATCYC        LNGCYC                                                         LL
            QUANTITY = CT/S          0.18589E-01   0.17744E-03  -0.86790E-02                                                   LL
            QUANTITY = BETAS        -0.79570        1.0414       0.34567                                                       LL
            QUANTITY = BETAC         -1.3175       0.89340E-02    1.4358                                                       LL
         GAIN MATRIX = (RELAXATION FACTOR) * D**-1                                                                             LL
                         QUANTITY = CT/S          BETAS         BETAC                                                          LL
            VARIABLE = COLL           18.635      -0.41501E-02   0.11364                                                       LL
            VARIABLE = LATCYC         8.5809       0.19054       0.59961E-02                                                   LL
            VARIABLE = LNGCYC         17.045      -0.49935E-02   0.24352                                                       LL
                                                                                                                               LL
                                                                                                                               LL
    ITERATION, COUNT =    1, LOOP = TRIM                                                                                       LL
          9.4644        3.0556        8.0691                       0.47929E-01    1.1517        2.0447                         LL
                                                                                                                               LL
    ITERATION, COUNT =    2, LOOP = TRIM                                                                                       LL
          9.8344        3.0991        8.1235                       0.54253E-01   0.92381        1.6364                         LL
                                                                                                                               LL
    ITERATION, COUNT =    3, LOOP = TRIM                                                                                       LL
          10.132        3.1342        8.1685                       0.59358E-01   0.74143        1.3061                         LL
                                                                                                                               LL
    ITERATION, COUNT =    4, LOOP = TRIM                                                                                       LL
          10.371        3.1622        8.2059                       0.63437E-01   0.59577        1.0424                         LL
                                                                                                                               LL
    ITERATION, COUNT =    5, LOOP = TRIM                                                                                       LL
          10.564        3.1846        8.2374                       0.66674E-01   0.47913       0.83002                         LL
                                                                                                                               LL
    ITERATION, COUNT =    6, LOOP = TRIM                                                                                       LL
          10.720        3.2026        8.2648                       0.69280E-01   0.38591       0.66153                         LL
```

```
                                                                                                                        LL
   ITERATION, COUNT =    7, LOOP = TRIM                                                                                 LL
         10.846        3.2171        8.2884                    0.71354E-01   0.31072       0.52580                      LL
                                                                                                                        LL
   ITERATION, COUNT =    8, LOOP = TRIM                                                                                 LL
         10.949        3.2290        8.3092                    0.73016E-01   0.25047       0.41753                      LL
                                                                                                                        LL
   ITERATION, COUNT =    9, LOOP = TRIM                                                                                 LL
         11.033        3.2387        8.3278                    0.74375E-01   0.20306       0.33292                      LL
                                                                                                                        LL
   ITERATION, COUNT =   10, LOOP = TRIM                                                                                 LL
         11.100        3.2462        8.3437                    0.75439E-01   0.16376       0.26355                      LL
                                                                                                                        LL
   ITERATION, COUNT =   11, LOOP = TRIM                                                                                 LL
         11.156        3.2526        8.3581                    0.76301E-01   0.13216       0.20835                      LL
                                                                                                                        LL
   ITERATION, COUNT =   12, LOOP = TRIM                                                                                 LL
         11.202        3.2579        8.3710                    0.76999E-01   0.10668       0.16454                      LL
                                                                                                                        LL
   ITERATION, COUNT =   13, LOOP = TRIM                                                                                 LL
         11.240        3.2624        8.3827                    0.77562E-01   0.86092E-01   0.12974                      LL
                                                                                                                        LL
   ITERATION, COUNT =   14, LOOP = TRIM                                                                                 LL
         11.271        3.2661        8.3930                    0.78029E-01   0.68738E-01   0.10099                      LL
                                                                                                                        LL
   ITERATION, COUNT =   15, LOOP = TRIM                                                                                 LL
         11.296        3.2693        8.4024                    0.78396E-01   0.55531E-01   0.79517E-01                  LL
```

```
****************
TRIM CONVERGENCE
****************

SUCCESSIVE SUBSTITUTION ITERATION, LOOP = WAKE
ITERATION  1, LEVEL   1 = UNIFORM INFLOW


    ---------------------------------
    STATUS OF LOOP AND PART SOLUTIONS
    ---------------------------------                                                                        TOTAL NUMBER
                                                                                                            OF ITERATIONS
         LOOP = TRIM                       NEWTON RAPHSON, CONVERGED                                               19
         LOOP = CIRCULATION                SUCCESSIVE SUBSTITUTION, CONVERGED                                      80
         LOOP = MOTION                     SUCCESSIVE SUBSTITUTION, CONVERGED                                      80
         LOOP = AIRFRAME                   NO SOLUTION

              PART = ROTOR 1 HUB                HARMONIC SOLUTION, CONVERGED                                      184
              PART = ROTOR 1 BLADE 4            HARMONIC SOLUTION, CONVERGED                                      103
              PART = ROTOR 1 HUB                HARMONIC SOLUTION, CONVERGED
              PART = AIRFRAME                   HARMONIC SOLUTION, CONVERGED                                       94


    --------------------------------------
    CONVERGENCE OF LOOP AND PART SOLUTIONS
    --------------------------------------
         ------------------------------------------------------------------------------------------------------------------------
         LOOP = TRIM                       NEWTON RAPHSON, CONVERGED
         NUMBER OF ITERATIONS =   15 (MAXIMUM   40), TOLERANCE =   5.00000
         ERROR = ABS(TRIMMED-TARGET); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)

              VARIABLE                          QUANTITY
                         TRIMMED        INITIAL                     TRIMMED        TARGET         WEIGHT         ERROR RATIO
              COLL        11.2963        9.00000        CT/S        0.783959E-01   0.800000E-01   0.800000E-03   0.401022
              LATCYC      3.26931        3.00000        BETAS       0.555307E-01   0.000000E+00   0.200000E-01   0.555307
              LNGCYC      8.40239        8.00000        BETAC       0.795175E-01   0.000000E+00   0.200000E-01   0.795175
         ------------------------------------------------------------------------------------------------------------------------
         LOOP = CIRCULATION                SUCCESSIVE SUBSTITUTION, CONVERGED
         NUMBER OF ITERATIONS =    2 (MAXIMUM  200), TOLERANCE =   1.00000
         ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
         ------------------------------------------------------------------------------------------------------------------------
         LOOP = MOTION                     SUCCESSIVE SUBSTITUTION, CONVERGED
         NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000
         ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
         ------------------------------------------------------------------------------------------------------------------------


         ------------------------------------------------------------------------------------------------------------------------
         PART = ROTOR 1 HUB                HARMONIC SOLUTION, CONVERGED
         NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000     FOR PERIOD   1 = ROTOR 1
         ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
         ------------------------------------------------------------------------------------------------------------------------
         PART = ROTOR 1 BLADE 4            HARMONIC SOLUTION, CONVERGED
         NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000     FOR PERIOD   1 = ROTOR 1
         ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
         ------------------------------------------------------------------------------------------------------------------------
```

```
PART = ROTOR 1 HUB                         HARMONIC SOLUTION, CONVERGED
NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000     FOR PERIOD   1 = ROTOR 1
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
------------------------------------------------------------------------------------------------------------------------
PART = AIRFRAME                            HARMONIC SOLUTION, CONVERGED
NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000     FOR PERIOD   1 = ROTOR 1
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
------------------------------------------------------------------------------------------------------------------------
```

```
------------------------------------
TRIM LOOP ITERATION AND PART SOLUTION
------------------------------------


SUCCESSIVE SUBSTITUTION ITERATION, LOOP = WAKE
LEVEL   1 = UNIFORM INFLOW

                                                                                                                                          LL
    NEWTON RAPHSON ITERATION, LOOP = TRIM                                                                                                 LL
    VARIABLES                                                QUANTITIES                                                                   LL
        COLL          LATCYC        LNGCYC                       CT/S          BETAS          BETAC                                       LL
    TARGETS                                                          0.80000E-01   0.00000E+00   0.00000E+00                              LL
    BEGINNING OF ITERATION, LOOP = TRIM                                                                                                   LL
          9.0000        3.0000        8.0000                         0.45121E-01    2.0143        1.7348                                  LL
                   *** WARNING, NOT CONVERGED *** LOOP = CIRCULATION                                                                      LL
                   *** WARNING, NOT CONVERGED *** PART = ROTOR 1 BLADE 4                                                                  LL
                                                                                                                                          LL
    PERTURBATION IDENTIFICATION, COUNT =    1, LOOP = TRIM                                                                                LL
          8.5000        3.0000        8.0000                         0.28859E-01    1.8063        3.2437                                  LL
                   *** WARNING, NOT CONVERGED *** LOOP = CIRCULATION                                                                      LL
                                                                                                                                          LL
    PERTURBATION IDENTIFICATION, COUNT =    2, LOOP = TRIM                                                                                LL
          9.0000        2.0000        8.0000                         0.40068E-01   0.54300        2.6426                                  LL
                   *** WARNING, NOT CONVERGED *** LOOP = CIRCULATION                                                                      LL
                   *** WARNING, NOT CONVERGED *** PART = ROTOR 1 BLADE 4                                                                  LL
                                                                                                                                          LL
    PERTURBATION IDENTIFICATION, COUNT =    3, LOOP = TRIM                                                                                LL
          9.0000        3.0000        9.0000                         0.31349E-01    1.6425        3.8198                                  LL
                                                                                                                                          LL
                                                                                                                                          LL
         PERTURBATION IDENTIFICATION                                                                                                      LL
         DERIVATIVE MATRIX D                                                                                                              LL
                         VARIABLE = COLL          LATCYC        LNGCYC                                                                    LL
            QUANTITY = CT/S         0.32522E-01   0.50521E-02  -0.13772E-01                                                               LL
            QUANTITY = BETAS        0.41610        1.4713      -0.37180                                                                   LL
            QUANTITY = BETAC       -3.0177       -0.90774        2.0849                                                                   LL
         GAIN MATRIX = (RELAXATION FACTOR) * D**-1                                                                                        LL
                         QUANTITY = CT/S          BETAS         BETAC                                                                     LL
            VARIABLE = COLL          15.998       0.11532E-01   0.10773                                                                   LL
            VARIABLE = LATCYC        1.4911       0.15381       0.37279E-01                                                               LL
            VARIABLE = LNGCYC        23.805       0.83659E-01   0.26809                                                                   LL
                                                                                                                                          LL
                   *** WARNING, NOT CONVERGED *** LOOP = CIRCULATION                                                                      LL
                   *** WARNING, NOT CONVERGED *** PART = ROTOR 1 BLADE 4                                                                  LL
                                                                                                                                          LL
    ITERATION, COUNT =    1, LOOP = TRIM                                                                                                  LL
          9.3479        2.6775        8.1967                         0.46030E-01   0.91203        2.3460                                  LL
                   *** WARNING, NOT CONVERGED *** LOOP = CIRCULATION                                                                      LL
                                                                                                                                          LL
    ITERATION, COUNT =    2, LOOP = TRIM                                                                                                  LL
          9.6281        2.5004        8.3001                         0.48863E-01   0.55325        2.1836                                  LL
                   *** WARNING, NOT CONVERGED *** LOOP = CIRCULATION                                                                      LL
                   *** WARNING, NOT CONVERGED *** PART = ROTOR 1 BLADE 4                                                                  LL
                   *** WARNING, NOT CONVERGED *** LOOP = TRIM                                                                             LL
```

```
****************
TRIM CONVERGENCE
****************

SUCCESSIVE SUBSTITUTION ITERATION, LOOP = WAKE
ITERATION  1, LEVEL   1 = UNIFORM INFLOW


    ---------------------------------
    STATUS OF LOOP AND PART SOLUTIONS
    ---------------------------------                                                                  TOTAL NUMBER
                                                                                                       OF ITERATIONS
         LOOP = TRIM                        NEWTON RAPHSON, NOT CONVERGED                                          6
         LOOP = CIRCULATION                 SUCCESSIVE SUBSTITUTION, NOT CONVERGED                                12
         LOOP = MOTION                      SUCCESSIVE SUBSTITUTION, CONVERGED                                    12
         LOOP = AIRFRAME                    NO SOLUTION

              PART = ROTOR 1 HUB                 HARMONIC SOLUTION, NOT CONVERGED                                 35
              PART = ROTOR 1 BLADE 4             HARMONIC SOLUTION, NOT CONVERGED                                 21
              PART = ROTOR 1 HUB                 HARMONIC SOLUTION, NOT CONVERGED
              PART = AIRFRAME                    HARMONIC SOLUTION, CONVERGED                                     22

         **************************************
         **************************************
         **                                  **
         **              WARNING             **
         **        LOOP NOT CONVERGED        **
         **        PART NOT CONVERGED        **
         **        RESULTS UNRELIABLE        **
         **                                  **
         **************************************
         **************************************



    --------------------------------------
    CONVERGENCE OF LOOP AND PART SOLUTIONS
    --------------------------------------
         ------------------------------------------------------------------------------------------------------------------------
         LOOP = TRIM                        NEWTON RAPHSON, NOT CONVERGED
         NUMBER OF ITERATIONS =    2 (MAXIMUM    2), TOLERANCE =   5.00000
         ERROR = ABS(TRIMMED-TARGET); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)

              VARIABLE                           QUANTITY
                         TRIMMED        INITIAL                     TRIMMED        TARGET         WEIGHT         ERROR RATIO
              COLL        9.62809        9.00000         CT/S       0.488627E-01   0.800000E-01   0.800000E-03    7.78432
              LATCYC      2.50043        3.00000         BETAS      0.553251       0.000000E+00   0.200000E-01    5.53251
              LNGCYC      8.30013        8.00000         BETAC       2.18360       0.000000E+00   0.200000E-01    21.8360
         ------------------------------------------------------------------------------------------------------------------------
         LOOP = CIRCULATION                 SUCCESSIVE SUBSTITUTION, NOT CONVERGED
         NUMBER OF ITERATIONS =    2 (MAXIMUM    2), TOLERANCE =   1.00000
         ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)

              VARIABLE                        QUANTITY  ELEMENT      ERROR                         WEIGHT          ERROR RATIO
```

```
     ROTOR 1 MEAN FORCE (TPP AXES)    INTERFACE      4     ABS(X-XOLD) =  2973.4         467.07          6.3660
     ROTOR 1 MEAN FORCE (TPP AXES)    INTERFACE      5     ABS(X-XOLD) =  1995.4         467.07          4.2722
     ROTOR 1 MEAN FORCE (TPP AXES)    INTERFACE      6     ABS(X-XOLD) =  526.72         467.07          1.1277
------------------------------------------------------------------------------------------------------------------------
LOOP = MOTION                                SUCCESSIVE SUBSTITUTION, CONVERGED
NUMBER OF ITERATIONS =    1 (MAXIMUM   40), TOLERANCE =   2.00000
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
------------------------------------------------------------------------------------------------------------------------


------------------------------------------------------------------------------------------------------------------------
PART = ROTOR 1 HUB                           HARMONIC SOLUTION, NOT CONVERGED
NUMBER OF ITERATIONS =    2 (MAXIMUM    2), TOLERANCE =   2.00000    FOR PERIOD   1 = ROTOR 1
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
------------------------------------------------------------------------------------------------------------------------
PART = ROTOR 1 BLADE 4                       HARMONIC SOLUTION, NOT CONVERGED
NUMBER OF ITERATIONS =    2 (MAXIMUM    2), TOLERANCE =   2.00000    FOR PERIOD   1 = ROTOR 1
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)

     VARIABLE                         QUANTITY  ELEMENT     ERROR                         WEIGHT          ERROR RATIO
     ROTOR 1 ROT HUB/BLADE 4 ROOT     INTERFACE      4     RMS(X-XOLD) =  1491.3         467.07          1.5964
     ROTOR 1 BLADE 4 ROOT/BLADE       INTERFACE      4     RMS(X-XOLD) =  1491.3         467.07          1.5964
     ROTOR 1 BLADE 4 EL 1/EL 2        INTERFACE      4     RMS(X-XOLD) =  1408.2         467.07          1.5075
     ROTOR 1 BLADE 4 EL 2/EL 3        INTERFACE      4     RMS(X-XOLD) =  1378.8         467.07          1.4760
------------------------------------------------------------------------------------------------------------------------
PART = ROTOR 1 HUB                           HARMONIC SOLUTION, NOT CONVERGED
NUMBER OF ITERATIONS =    2 (MAXIMUM    2), TOLERANCE =   2.00000    FOR PERIOD   1 = ROTOR 1
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
------------------------------------------------------------------------------------------------------------------------
PART = AIRFRAME                              HARMONIC SOLUTION, CONVERGED
NUMBER OF ITERATIONS =    2 (MAXIMUM   40), TOLERANCE =   2.00000    FOR PERIOD   1 = ROTOR 1
ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)
------------------------------------------------------------------------------------------------------------------------
```

```
Converged Calculation:  RELAXC = .05 and RTVTX = 1.00
=====================================================


PERFORMANCE INDICES
                   MEAN INFLOW RATIO    EQUIVALENT INFLOW    EXCESS INDUCED LOSSES            MEAN INFLOW RATIO
                    MAGNITUDE LAMBDA          CP/CT              KAPPA = P/PM           LAMBDAX    LAMBDAY    LAMBDAZ
    INDUCED            0.018372             0.026004               2.6798               0.004577   0.001313  -0.017744
    INTERFERENCE       0.000000             0.000000               0.0000               0.000000   0.000000   0.000000
    MINIMUM INDUCED    0.009704             0.009704               1.0000
    EQUIVALENT SECTION DRAG COEFFICIENT, PROFILE POWER     CDO = (8*CPO/S)/F(MU) =   0.01032
    EQUIVALENT SECTION DRAG COEFFICIENT, NONIDEAL POWER    CDN = (8*CPN/S)/F(MU) =   0.01659

    FIGURE OF MERIT, M = (PM+PINT)/(PO+PI) =   0.1835       POWER COEFFICIENT, CP = -P/QAV          =  -0.0270
    FIGURE OF MERIT, M = PIDEAL/P          =   0.5596       WIND TURBINE EFFICIENCY, M = (27/16)CP =  -0.0456
    PROPULSIVE EFFICIENCY, ETA = (PC+PP)/P =   0.4606       WING INDUCED EFFICIENCY, E              =   1.4777
    ROTOR LIFT-TO-DRAG RATIO, L/D          =   7.0651       WING LIFT-TO-DRAG RATIO, L/X            =  -8.2727

                      FORCE (LB)                     FORCE/Q (FT**2)         COEFFICIENT (F/QA)    WING COEFFICIENT (F/QS)
    ROTOR EQUIV DRAG   D    = P/V+X  =     1718.6964     D/Q    =    27.9609     CD    =  0.01456     CDW    =   0.16000
    PROFILE DRAG       DO   = PO/V   =      873.3690     DO/Q   =    14.2085     CDO   =  0.00740     CDWO   =   0.08131
    INDUCED DRAG       DIND = PIND/V =      845.3275     DIND/Q =    13.7523     CDIND =  0.00716     CDWIND =   0.07870
    INTERFERENCE DRAG  DINT = PINT/V =        0.0000     DINT/Q =     0.0000     CDINT =  0.00000     CDWINT =   0.00000
    ROTOR DRAG         X             =    -1467.8143     X/Q    =   -23.8794     CX    = -0.01243     CXW    =  -0.13665
    ROTOR LIFT         L             =    12142.7373     L/Q    =   197.5460     CL    =  0.10287     CLW    =   1.13042


          MEAN INFLOW RATIO:  VELOCITY MAGNITUDE AND COMPONENTS IN SHAFT AXES (AVERAGE OVER ROTOR DISK)
          FIGURE OF MERIT IS MEASURE OF ROTOR EFFICIENCY APPROPRIATE FOR AXIAL FLOW
              BOTH DEFINITIONS INCLUDE CONVENTIONAL HOVER FIGURE OF MERIT, M = T*VIDEAL/P = T*SQRT(T/2*RHO*A)/P
              M = (PM+PINT)/(PO+PI) = T*(VIDEAL+VINT)/(PO+PI)
              M = PIDEAL/P = 1 - PN/P = T*(V+VIDEAL)/P FOR AXIAL FLOW  (NOT VALID FOR ZERO POWER)
              SECOND DEFINITION NEARLY PROPULSIVE EFFICIENCY ETA = TV/P FOR HIGH INFLOW
          PROPULSIVE EFFICIENCY IS MEASURE APPROPRIATE FOR HIGH SPEED PROPULSION
              ETA = -X*V/P = (PC+PP)/P  (NOT VALID FOR ZERO VELOCITY OR ZERO POWER)
          ROTOR LIFT-TO-DRAG RATIO IS MEASURE OF ROTOR LIFTING EFFICIENCY, APPROPRIATE FOR EDGEWISE FLOW
              L/D = L*V/(PI+PO)  (NOT USEFUL FOR ZERO VELOCITY); ROTOR EQUIVALENT DRAG D = P/V+X = (PI+PO)/V
          WING LIFT-TO-DRAG RATIO AND INDUCED EFFICIENCY E ARE MEASURES APPROPRIATE FOR FIXED WING
              DI/Q = (L/Q)**2/(E*PI*B**2)

          ----------------------------------------------------------------------------------------------------------------------
          INFLOW RATIO, INFLOW = VELOCITY / VTIP
              REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTATIONAL SPEED OMEGA =  260.000 RPM; TIP SPEED VTIP = OMEGA*R
          COEFFICIENT CF = F/QA, WING COEFFICIENT CFW = F/QS
              ROTOR AREA A = PI*R**2; BLADE AREA S = A*SIGMA; WING SPAN B = R; DYNAMIC PRESSURE Q=.5*RHO*V**2
              RADIUS R =   24.7240 FT, SOLIDITY RATIO SIGMA =  0.09100, AIR DENSITY RHO = 0.001918 SLUG/FT**3
          ----------------------------------------------------------------------------------------------------------------------
```

Converged Calculation: RELAXC = .05 and RTVTX = 1.00
===

BLADE SECTION ANGLE OF ATTACK (DEG)

RADIAL STATION =	0.560	0.615	0.665	0.710	0.750	0.785	0.815	0.845	0.870	0.890	0.910	0.930	0.950	0.970	0.990
AZIMUTH (DEG)	ALPHA														
PSI = 0.0	7.1	7.5	7.5	7.4	7.2	7.0	6.8	6.1	5.1	4.7	5.0	5.4	5.3	5.6	4.9
PSI = 15.0	5.6	5.6	5.6	5.5	5.4	5.2	4.9	4.3	3.7	3.3	3.5	3.8	3.7	3.9	3.5
PSI = 30.0	4.2	4.0	3.9	3.7	3.5	3.3	3.1	2.7	2.3	2.0	2.1	2.3	2.2	2.4	2.1
PSI = 45.0	2.6	2.4	2.2	2.0	1.9	1.7	1.5	1.3	1.0	0.9	0.9	1.0	1.0	1.1	0.9
PSI = 60.0	1.7	1.6	1.4	1.2	1.0	0.9	0.7	0.5	0.4	0.3	0.2	0.2	0.2	0.2	0.2
PSI = 75.0	1.2	1.0	0.8	0.6	0.5	0.3	0.2	0.1	0.0	0.0	-0.1	-0.1	-0.1	-0.1	-0.1
PSI = 90.0	1.1	0.9	0.8	0.6	0.5	0.3	0.2	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
PSI = 105.0	1.5	1.3	1.1	0.9	0.8	0.7	0.6	0.6	0.5	0.4	0.3	0.3	0.2	0.2	0.1
PSI = 120.0	2.0	1.8	1.7	1.6	1.5	1.4	1.3	1.2	1.1	0.8	0.6	0.5	0.5	0.4	0.3
PSI = 135.0	2.9	2.9	2.8	2.6	2.4	2.3	2.1	2.0	1.8	1.4	1.1	0.9	0.8	0.8	0.6
PSI = 150.0	4.2	4.2	4.0	3.7	3.5	3.3	3.1	2.9	2.7	2.2	1.7	1.5	1.3	1.2	1.0
PSI = 165.0	5.9	5.6	5.4	5.0	4.7	4.4	4.2	4.1	3.9	3.1	2.5	2.2	1.9	1.8	1.4
PSI = 180.0	7.5	7.2	6.8	6.4	6.1	5.8	5.5	5.4	5.2	4.2	3.4	3.0	2.7	2.4	1.9
PSI = 195.0	9.1	8.7	8.3	7.9	7.6	7.2	6.9	6.9	6.7	5.4	4.4	3.9	3.5	3.2	2.5
PSI = 210.0	10.7	10.2	9.7	9.3	9.0	8.6	8.4	8.3	8.0	6.6	5.5	4.9	4.4	4.0	3.2
PSI = 225.0	12.2	11.6	11.1	10.7	10.3	9.9	9.7	9.6	9.2	7.7	6.6	6.0	5.4	4.9	4.0
PSI = 240.0	13.2	12.7	12.3	11.8	11.4	11.1	10.9	10.7	10.1	8.6	7.6	7.1	6.5	6.0	4.9
PSI = 255.0	13.8	13.4	13.1	12.8	12.5	12.2	11.9	11.5	10.6	9.3	8.5	8.1	7.5	7.0	5.8
PSI = 270.0	18.4	16.4	15.1	12.7	12.7	11.3	21.1	10.5	10.1	9.3	9.0	8.9	8.3	8.0	6.6
PSI = 285.0	12.8	12.9	12.8	12.1	11.8	10.7	22.8	10.2	9.8	9.2	9.2	9.3	8.9	8.7	7.4
PSI = 300.0	11.5	11.7	11.8	11.9	11.7	11.7	12.4	10.6	9.3	8.5	8.7	9.0	8.7	8.8	7.5
PSI = 315.0	10.1	10.6	10.8	10.8	10.8	10.7	10.3	9.6	8.4	7.8	8.1	8.4	8.2	8.4	7.2
PSI = 330.0	8.1	9.3	9.7	9.9	9.9	9.9	9.7	8.9	7.6	7.0	7.5	8.0	7.8	8.1	7.1
PSI = 345.0	8.7	8.6	8.7	8.8	8.8	8.7	8.5	7.6	6.5	6.0	6.4	6.9	6.8	7.1	6.2
PSI = 360.0	7.1	7.5	7.5	7.4	7.2	7.0	6.8	6.1	5.1	4.7	5.0	5.4	5.3	5.6	4.9

```
Circulation Not Converged:  RELAXC = .10
========================================

      ------------------------------------------------------------------------------------------------------------------------
      LOOP = CIRCULATION                          SUCCESSIVE SUBSTITUTION, NOT CONVERGED
      NUMBER OF ITERATIONS =  200 (MAXIMUM  200), TOLERANCE =   1.00000
      ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)

           VARIABLE                         QUANTITY  ELEMENT      ERROR                           WEIGHT          ERROR RATIO
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      13     RMS(X-XOLD) =  78.396             30.309           2.5865
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      14     RMS(X-XOLD) =  167.99             30.309           5.5424
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      15     RMS(X-XOLD) =  205.85             30.309           6.7917
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      16     RMS(X-XOLD) =  175.90             30.309           5.8036
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      17     RMS(X-XOLD) =  104.72             30.309           3.4550
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      18     RMS(X-XOLD) =  31.256             30.309           1.0312
      ------------------------------------------------------------------------------------------------------------------------

PERFORMANCE INDICES
                   MEAN INFLOW RATIO    EQUIVALENT INFLOW    EXCESS INDUCED LOSSES            MEAN INFLOW RATIO
                    MAGNITUDE LAMBDA          CP/CT              KAPPA = P/PM             LAMBDAX    LAMBDAY    LAMBDAZ
    INDUCED            0.018208             0.022389               2.2764                 0.003993   0.001193  -0.017725
    INTERFERENCE       0.000000             0.000000               0.0000                 0.000000   0.000000   0.000000
    MINIMUM INDUCED    0.009836             0.009836               1.0000
    EQUIVALENT SECTION DRAG COEFFICIENT, PROFILE POWER     CDO = (8*CPO/S)/F(MU) =   0.01114
    EQUIVALENT SECTION DRAG COEFFICIENT, NONIDEAL POWER    CDN = (8*CPN/S)/F(MU) =   0.01603

    FIGURE OF MERIT, M = (PM+PINT)/(PO+PI) =   0.1929       POWER COEFFICIENT, CP = -P/QAV          =  -0.0265
    FIGURE OF MERIT, M = PIDEAL/P          =   0.5665       WIND TURBINE EFFICIENCY, M = (27/16)CP =  -0.0447
    PROPULSIVE EFFICIENCY, ETA = (PC+PP)/P =   0.4629       WING INDUCED EFFICIENCY, E              =   1.7412
    ROTOR LIFT-TO-DRAG RATIO, L/D          =   7.3294       WING LIFT-TO-DRAG RATIO, L/X            =  -8.5046

                     FORCE (LB)                    FORCE/Q (FT**2)         COEFFICIENT (F/QA)     WING COEFFICIENT (F/QS)
    ROTOR EQUIV DRAG  D    = P/V+X  =    1680.0514    D/Q    =    27.3322    CD    =  0.01423     CDW    =   0.15640
    PROFILE DRAG      DO   = PO/V   =     942.3221    DO/Q   =    15.3303    CDO   =  0.00798     CDWO   =   0.08772
    INDUCED DRAG      DIND = PIND/V =     737.7292    DIND/Q =    12.0019    CDIND =  0.00625     CDWIND =   0.06868
    INTERFERENCE DRAG DINT = PINT/V =       0.0000    DINT/Q =     0.0000    CDINT =  0.00000     CDWINT =   0.00000
    ROTOR DRAG        X             =   -1447.8981    X/Q    =   -23.5554    CX    = -0.01227     CXW    =  -0.13479
    ROTOR LIFT        L             =   12313.8506    L/Q    =   200.3298    CL    =  0.10432     CLW    =   1.14635

        MEAN INFLOW RATIO:  VELOCITY MAGNITUDE AND COMPONENTS IN SHAFT AXES (AVERAGE OVER ROTOR DISK)
        FIGURE OF MERIT IS MEASURE OF ROTOR EFFICIENCY APPROPRIATE FOR AXIAL FLOW
            BOTH DEFINITIONS INCLUDE CONVENTIONAL HOVER FIGURE OF MERIT, M = T*VIDEAL/P = T*SQRT(T/2*RHO*A)/P
            M = (PM+PINT)/(PO+PI) = T*(VIDEAL+VINT)/(PO+PI)
            M = PIDEAL/P = 1 - PN/P = T*(V+VIDEAL)/P FOR AXIAL FLOW  (NOT VALID FOR ZERO POWER)
            SECOND DEFINITION NEARLY PROPULSIVE EFFICIENCY ETA = TV/P FOR HIGH INFLOW
        PROPULSIVE EFFICIENCY IS MEASURE APPROPRIATE FOR HIGH SPEED PROPULSION
            ETA = -X*V/P = (PC+PP)/P  (NOT VALID FOR ZERO VELOCITY OR ZERO POWER)
        ROTOR LIFT-TO-DRAG RATIO IS MEASURE OF ROTOR LIFTING EFFICIENCY, APPROPRIATE FOR EDGEWISE FLOW
            L/D = L*V/(PI+PO)  (NOT USEFUL FOR ZERO VELOCITY); ROTOR EQUIVALENT DRAG D = P/V+X = (PI+PO)/V
        WING LIFT-TO-DRAG RATIO AND INDUCED EFFICIENCY E ARE MEASURES APPROPRIATE FOR FIXED WING
            DI/Q = (L/Q)**2/(E*PI*B**2)
```

Circulation Not Converged: RELAXC = .10
==

BLADE SECTION ANGLE OF ATTACK (DEG)

RADIAL STATION =	0.560	0.615	0.665	0.710	0.750	0.785	0.815	0.845	0.870	0.890	0.910	0.930	0.950	0.970	0.990
AZIMUTH (DEG)	ALPHA														
PSI = 0.0	7.2	7.5	7.5	7.4	7.2	7.0	6.7	6.0	5.1	4.6	4.9	5.3	5.3	5.6	4.9
PSI = 15.0	5.1	5.3	5.3	5.3	5.2	5.0	4.8	4.3	3.6	3.3	3.5	3.7	3.7	3.9	3.5
PSI = 30.0	4.0	4.0	3.8	3.7	3.5	3.4	3.2	2.8	2.3	2.1	2.2	2.3	2.3	2.5	2.2
PSI = 45.0	2.7	2.5	2.4	2.2	2.0	1.8	1.7	1.4	1.1	1.0	1.0	1.1	1.1	1.2	1.0
PSI = 60.0	1.8	1.8	1.7	1.6	1.5	1.3	1.2	1.0	0.7	0.6	0.5	0.5	0.5	0.5	0.4
PSI = 75.0	1.4	1.1	0.9	0.6	0.5	0.3	0.2	0.1	-0.1	0.0	-0.3	-0.2	-0.3	-0.3	-0.3
PSI = 90.0	1.8	1.7	1.6	1.4	1.3	1.1	0.9	0.8	-0.5	2.2	-1.7	1.1	-0.4	-0.1	-0.2
PSI = 105.0	1.4	1.2	0.9	0.7	0.6	0.4	0.3	0.4	-1.5	2.8	-3.1	2.2	-1.2	0.1	-0.2
PSI = 120.0	2.0	1.8	1.6	1.5	1.4	1.2	1.1	1.2	-0.8	3.8	-2.7	3.4	-1.4	0.8	0.1
PSI = 135.0	2.9	2.9	2.8	2.6	2.4	2.2	2.0	2.1	0.4	4.2	-2.0	4.1	-1.0	1.4	0.5
PSI = 150.0	4.0	4.1	4.0	3.8	3.6	3.4	3.2	3.2	1.8	4.4	-0.6	4.0	0.0	1.8	0.9
PSI = 165.0	6.0	5.9	5.6	5.3	5.0	4.7	4.4	4.4	3.4	4.8	1.0	3.9	1.2	2.2	1.4
PSI = 180.0	7.9	7.6	7.2	6.8	6.4	6.1	5.9	5.8	5.1	5.4	2.5	4.2	2.4	2.8	2.0
PSI = 195.0	9.5	9.1	8.7	8.3	7.9	7.5	7.3	7.3	6.7	6.2	4.0	4.7	3.4	3.4	2.6
PSI = 210.0	11.1	10.6	10.1	9.7	9.3	9.0	8.8	8.7	8.2	7.2	5.4	5.4	4.5	4.2	3.3
PSI = 225.0	12.5	11.9	11.4	11.0	10.6	10.3	10.1	10.0	9.5	8.2	6.7	6.4	5.6	5.1	4.1
PSI = 240.0	13.6	13.0	12.6	12.2	11.8	11.5	11.3	11.1	10.5	9.0	7.8	7.4	6.7	6.2	5.0
PSI = 255.0	14.3	13.8	13.6	13.3	13.0	12.7	12.4	12.0	11.0	9.7	8.8	8.4	7.8	7.3	6.0
PSI = 270.0	19.5	16.3	19.2	12.2	12.7	11.6	23.8	10.8	10.4	9.6	9.3	9.1	8.6	8.2	6.8
PSI = 285.0	13.1	12.4	14.0	12.2	11.9	11.1	25.9	10.6	10.1	9.5	9.4	9.5	9.1	8.9	7.5
PSI = 300.0	11.5	12.2	11.7	12.2	12.1	12.1	12.6	11.0	9.6	8.8	8.9	9.2	8.9	9.0	7.7
PSI = 315.0	10.4	10.7	11.1	11.0	11.0	10.9	10.5	9.8	8.6	7.9	8.3	8.6	8.4	8.6	7.4
PSI = 330.0	8.1	9.3	9.7	10.0	10.0	10.0	9.8	9.0	7.7	7.1	7.6	8.1	7.9	8.2	7.2
PSI = 345.0	8.7	8.6	8.7	8.8	8.8	8.7	8.5	7.6	6.5	6.0	6.4	7.0	6.9	7.1	6.3
PSI = 360.0	7.2	7.5	7.5	7.4	7.2	7.0	6.7	6.0	5.1	4.6	4.9	5.3	5.3	5.6	4.9

```
Circulation Not Converged:  RELAXC = .15
========================================


      ------------------------------------------------------------------------------------------------------------------------
      LOOP = CIRCULATION                         SUCCESSIVE SUBSTITUTION, NOT CONVERGED
      NUMBER OF ITERATIONS =  200 (MAXIMUM  200), TOLERANCE =   1.00000
      ERROR = ABS(X-XOLD) OR RMS(X-XOLD); ERROR RATIO = ERROR/(TOLER*WEIGHT) (LE 1 FOR CONVERGENCE)

           VARIABLE                         QUANTITY  ELEMENT      ERROR                           WEIGHT           ERROR RATIO
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      12      RMS(X-XOLD) =  146.06            30.309            4.8191
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      13      RMS(X-XOLD) =  392.04            30.309            12.935
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      14      RMS(X-XOLD) =  478.17            30.309            15.776
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      15      RMS(X-XOLD) =  491.73            30.309            16.224
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      16      RMS(X-XOLD) =  480.85            30.309            15.865
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      17      RMS(X-XOLD) =  467.51            30.309            15.424
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      18      RMS(X-XOLD) =  321.40            30.309            10.604
           ROTOR 1 WING 4 BOUND CIRCULATION INTERFACE      19      RMS(X-XOLD) =  116.89            30.309            3.8564
      ------------------------------------------------------------------------------------------------------------------------


PERFORMANCE INDICES
                   MEAN INFLOW RATIO    EQUIVALENT INFLOW    EXCESS INDUCED LOSSES            MEAN INFLOW RATIO
                    MAGNITUDE LAMBDA          CP/CT              KAPPA = P/PM             LAMBDAX    LAMBDAY    LAMBDAZ
    INDUCED            0.012623             -0.034309              -3.1554                0.001870  -0.001472  -0.012396
    INTERFERENCE       0.000000              0.000000               0.0000                0.000000   0.000000   0.000000
    MINIMUM INDUCED    0.010873              0.010873               1.0000
    EQUIVALENT SECTION DRAG COEFFICIENT, PROFILE POWER     CDO = (8*CPO/S)/F(MU) =   0.02964
    EQUIVALENT SECTION DRAG COEFFICIENT, NONIDEAL POWER    CDN = (8*CPN/S)/F(MU) =   0.01018

    FIGURE OF MERIT, M = (PM+PINT)/(PO+PI) =   0.3151       POWER COEFFICIENT, CP = -P/QAV          =  -0.0232
    FIGURE OF MERIT, M = PIDEAL/P          =   0.6861       WIND TURBINE EFFICIENCY, M = (27/16)CP =  -0.0392
    PROPULSIVE EFFICIENCY, ETA = (PC+PP)/P =   0.5418       WING INDUCED EFFICIENCY, E              =  -1.2602
    ROTOR LIFT-TO-DRAG RATIO, L/D          =  10.8450       WING LIFT-TO-DRAG RATIO, L/X            =  -9.1730

                     FORCE (LB)                           FORCE/Q (FT**2)         COEFFICIENT (F/QA)    WING COEFFICIENT (F/QS)
    ROTOR EQUIV DRAG   D    = P/V+X   =     1257.4633     D/Q     =    20.4572     CD    =   0.01065     CDW    =   0.11706
    PROFILE DRAG       DO   = PO/V    =     2507.6746     DO/Q    =    40.7965     CDO   =   0.02124     CDWO   =   0.23345
    INDUCED DRAG       DIND = PIND/V  =    -1250.2112     DIND/Q =   -20.3393      CDIND =  -0.01059     CDWIND =  -0.11639
    INTERFERENCE DRAG  DINT = PINT/V  =        0.0000     DINT/Q =     0.0000      CDINT =   0.00000     CDWINT =   0.00000
    ROTOR DRAG         X              =    -1486.6710     X/Q     =   -24.1861     CX    =  -0.01259     CXW    =  -0.13840
    ROTOR LIFT         L              =    13637.2422     L/Q     =   221.8596     CL    =   0.11553     CLW    =   1.26955

        MEAN INFLOW RATIO:  VELOCITY MAGNITUDE AND COMPONENTS IN SHAFT AXES (AVERAGE OVER ROTOR DISK)
        FIGURE OF MERIT IS MEASURE OF ROTOR EFFICIENCY APPROPRIATE FOR AXIAL FLOW
            BOTH DEFINITIONS INCLUDE CONVENTIONAL HOVER FIGURE OF MERIT, M = T*VIDEAL/P = T*SQRT(T/2*RHO*A)/P
            M = (PM+PINT)/(PO+PI) = T*(VIDEAL+VINT)/(PO+PI)
            M = PIDEAL/P = 1 - PN/P = T*(V+VIDEAL)/P FOR AXIAL FLOW  (NOT VALID FOR ZERO POWER)
            SECOND DEFINITION NEARLY PROPULSIVE EFFICIENCY ETA = TV/P FOR HIGH INFLOW
        PROPULSIVE EFFICIENCY IS MEASURE APPROPRIATE FOR HIGH SPEED PROPULSION
            ETA = -X*V/P = (PC+PP)/P  (NOT VALID FOR ZERO VELOCITY OR ZERO POWER)
        ROTOR LIFT-TO-DRAG RATIO IS MEASURE OF ROTOR LIFTING EFFICIENCY, APPROPRIATE FOR EDGEWISE FLOW
            L/D = L*V/(PI+PO)  (NOT USEFUL FOR ZERO VELOCITY); ROTOR EQUIVALENT DRAG D = P/V+X = (PI+PO)/V
        WING LIFT-TO-DRAG RATIO AND INDUCED EFFICIENCY E ARE MEASURES APPROPRIATE FOR FIXED WING
            DI/Q = (L/Q)**2/(E*PI*B**2)
```

Circulation Not Converged: RELAXC = .15
==

BLADE SECTION ANGLE OF ATTACK (DEG)

RADIAL STATION =	0.560	0.615	0.665	0.710	0.750	0.785	0.815	0.845	0.870	0.890	0.910	0.930	0.950	0.970	0.990
AZIMUTH (DEG)	ALPHA														
PSI = 0.0	8.0	8.2	8.1	7.9	7.6	7.3	7.0	6.5	3.9	8.2	1.9	8.2	4.1	6.0	4.9
PSI = 15.0	6.9	6.8	6.7	6.5	6.3	6.1	5.8	5.3	3.6	5.5	2.5	5.5	3.9	4.7	4.0
PSI = 30.0	5.4	5.2	5.0	4.6	4.3	3.9	3.6	2.3	6.1	-1.6	8.8	-2.8	5.5	0.9	2.1
PSI = 45.0	2.8	2.4	2.1	1.9	1.7	1.5	1.7	-0.3	7.6	-5.1	10.7	-6.8	7.7	-2.2	1.4
PSI = 60.0	2.1	1.4	0.9	0.6	0.3	0.0	0.8	-2.0	7.1	-7.5	9.1	-8.6	7.7	-3.8	1.0
PSI = 75.0	2.3	2.0	1.6	1.2	0.9	0.2	1.2	-2.2	7.0	-7.9	7.9	-8.8	6.7	-4.5	0.8
PSI = 90.0	1.6	1.0	0.5	0.1	-0.1	-0.7	0.6	-2.9	5.7	-9.2	6.4	-9.3	6.0	-4.9	1.2
PSI = 105.0	3.1	2.7	2.3	1.8	1.4	0.8	0.9	0.1	-2.0	3.6	-3.9	-0.8	4.0	-4.1	1.0
PSI = 120.0	3.7	3.2	2.6	2.0	1.5	1.1	0.4	2.1	-5.9	8.9	-7.2	6.7	-4.8	2.7	-0.9
PSI = 135.0	4.6	3.9	3.2	2.7	2.3	2.1	1.0	3.4	-5.4	11.7	-5.9	9.9	-5.8	5.9	-1.8
PSI = 150.0	5.6	4.9	4.4	3.9	3.5	3.4	2.3	4.5	-3.8	13.5	-4.7	11.4	-4.4	7.0	-1.6
PSI = 165.0	7.0	6.4	6.0	5.5	5.1	4.9	3.9	5.9	-2.1	14.1	-3.0	11.9	-3.1	7.5	-0.9
PSI = 180.0	8.9	8.3	7.8	7.3	6.8	6.5	5.7	7.5	-0.3	14.4	-1.2	11.7	-1.6	7.6	0.1
PSI = 195.0	10.6	10.1	9.7	9.2	8.7	8.4	7.6	9.3	1.6	15.1	0.5	11.4	0.0	7.7	1.1
PSI = 210.0	12.6	12.0	11.4	10.9	10.4	10.0	9.3	10.9	3.5	15.7	2.1	11.5	1.4	7.9	2.3
PSI = 225.0	14.4	13.6	12.9	12.3	11.8	11.5	10.9	12.1	5.8	15.6	3.8	11.6	3.0	8.1	3.5
PSI = 240.0	15.8	14.8	14.2	13.6	13.1	12.7	12.2	12.8	8.3	14.6	5.8	11.7	4.6	8.5	4.8
PSI = 255.0	15.6	15.0	14.7	14.3	14.1	13.7	13.5	13.1	10.1	13.0	7.4	11.4	6.2	8.8	5.9
PSI = 270.0	21.6	21.1	12.6	23.2	12.2	11.8	24.9	11.5	10.2	11.6	8.4	11.2	7.7	9.3	6.9
PSI = 285.0	13.2	14.0	10.7	23.9	11.0	10.9	26.4	11.0	10.0	10.7	8.7	11.1	8.6	9.7	7.7
PSI = 300.0	11.4	11.2	12.4	12.9	11.8	11.8	12.4	10.9	9.2	9.6	8.0	10.2	8.4	9.3	7.7
PSI = 315.0	10.0	10.5	10.4	10.4	10.7	10.6	10.2	9.6	7.9	8.6	7.1	9.2	7.8	8.5	7.2
PSI = 330.0	7.6	8.7	9.3	9.5	9.5	9.6	9.4	8.7	6.9	8.0	6.3	8.8	7.3	8.2	7.0
PSI = 345.0	8.6	8.4	8.4	8.5	8.5	8.4	8.1	7.4	5.5	7.8	4.3	8.2	6.0	7.1	6.1
PSI = 360.0	8.0	8.2	8.1	7.9	7.6	7.3	7.0	6.5	3.9	8.2	1.9	8.2	4.1	6.0	4.9

Tip Vortex Rollup: RTVTX = 0.98 and RELAXC = .05
===

```
PERFORMANCE INDICES
                   MEAN INFLOW RATIO    EQUIVALENT INFLOW    EXCESS INDUCED LOSSES             MEAN INFLOW RATIO
                    MAGNITUDE LAMBDA          CP/CT              KAPPA = P/PM            LAMBDAX    LAMBDAY    LAMBDAZ
    INDUCED            0.012507             0.026857               2.9481                0.004561   0.001347  -0.011568
    INTERFERENCE       0.000000             0.000000               0.0000                0.000000   0.000000   0.000000
    MINIMUM INDUCED    0.009110             0.009110               1.0000
    EQUIVALENT SECTION DRAG COEFFICIENT, PROFILE POWER     CDO = (8*CPO/S)/F(MU) =   0.01911
    EQUIVALENT SECTION DRAG COEFFICIENT, NONIDEAL POWER    CDN = (8*CPN/S)/F(MU) =   0.02551

    FIGURE OF MERIT, M = (PM+PINT)/(PO+PI) =   0.1141       POWER COEFFICIENT, CP = -P/QAV          =  -0.0325
    FIGURE OF MERIT, M = PIDEAL/P          =   0.4380       WIND TURBINE EFFICIENCY, M = (27/16)CP =  -0.0549
    PROPULSIVE EFFICIENCY, ETA = (PC+PP)/P =   0.3656       WING INDUCED EFFICIENCY, E              =   1.3429
    ROTOR LIFT-TO-DRAG RATIO, L/D          =   4.6785       WING LIFT-TO-DRAG RATIO, L/X            =  -8.1178

                     FORCE (LB)                     FORCE/Q (FT**2)        COEFFICIENT (F/QA)    WING COEFFICIENT (F/QS)
    ROTOR EQUIV DRAG   D    = P/V+X  =     2436.0659     D/Q    =    39.6315    CD    =  0.02064    CDW    =   0.22678
    PROFILE DRAG       DO   = PO/V   =     1616.6117     DO/Q   =    26.3001    CDO   =  0.01370    CDWO   =   0.15050
    INDUCED DRAG       DIND = PIND/V =      819.4544     DIND/Q =    13.3314    CDIND =  0.00694    CDWIND =   0.07629
    INTERFERENCE DRAG  DINT = PINT/V =        0.0000     DINT/Q =     0.0000    CDINT =  0.00000    CDWINT =   0.00000
    ROTOR DRAG         X             =    -1403.9565     X/Q    =   -22.8405    CX    = -0.01189    CXW    =  -0.13070
    ROTOR LIFT         L             =    11397.0957     L/Q    =   185.4155    CL    =  0.09655    CLW    =   1.06100

          MEAN INFLOW RATIO:  VELOCITY MAGNITUDE AND COMPONENTS IN SHAFT AXES (AVERAGE OVER ROTOR DISK)
          FIGURE OF MERIT IS MEASURE OF ROTOR EFFICIENCY APPROPRIATE FOR AXIAL FLOW
              BOTH DEFINITIONS INCLUDE CONVENTIONAL HOVER FIGURE OF MERIT, M = T*VIDEAL/P = T*SQRT(T/2*RHO*A)/P
              M = (PM+PINT)/(PO+PI) = T*(VIDEAL+VINT)/(PO+PI)
              M = PIDEAL/P = 1 - PN/P = T*(V+VIDEAL)/P FOR AXIAL FLOW  (NOT VALID FOR ZERO POWER)
              SECOND DEFINITION NEARLY PROPULSIVE EFFICIENCY ETA = TV/P FOR HIGH INFLOW
          PROPULSIVE EFFICIENCY IS MEASURE APPROPRIATE FOR HIGH SPEED PROPULSION
              ETA = -X*V/P = (PC+PP)/P  (NOT VALID FOR ZERO VELOCITY OR ZERO POWER)
          ROTOR LIFT-TO-DRAG RATIO IS MEASURE OF ROTOR LIFTING EFFICIENCY, APPROPRIATE FOR EDGEWISE FLOW
              L/D = L*V/(PI+PO)  (NOT USEFUL FOR ZERO VELOCITY); ROTOR EQUIVALENT DRAG D = P/V+X = (PI+PO)/V
          WING LIFT-TO-DRAG RATIO AND INDUCED EFFICIENCY E ARE MEASURES APPROPRIATE FOR FIXED WING
              DI/Q = (L/Q)**2/(E*PI*B**2)

          ----------------------------------------------------------------------------------------------------------------------
          INFLOW RATIO, INFLOW = VELOCITY / VTIP
              REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTATIONAL SPEED OMEGA =  260.000 RPM; TIP SPEED VTIP = OMEGA*R
          COEFFICIENT CF = F/QA, WING COEFFICIENT CFW = F/QS
              ROTOR AREA A = PI*R**2; BLADE AREA S = A*SIGMA; WING SPAN B = R; DYNAMIC PRESSURE Q=.5*RHO*V**2
              RADIUS R =   24.7240 FT, SOLIDITY RATIO SIGMA =  0.09100, AIR DENSITY RHO = 0.001918 SLUG/FT**3
          ----------------------------------------------------------------------------------------------------------------------
```

Tip Vortex Rollup: RTVTX = 0.98 and RELAXC = .05
==

BLADE SECTION ANGLE OF ATTACK (DEG)

AZIMUTH (DEG) \ RADIAL STATION =	0.560	0.615	0.665	0.710	0.750	0.785	0.815	0.845	0.870	0.890	0.910	0.930	0.950	0.970	0.990
	ALPHA														
PSI = 0.0	7.1	7.5	7.4	7.3	7.1	6.8	6.5	5.8	4.8	4.3	4.5	4.7	4.3	3.8	24.7
PSI = 15.0	5.4	5.5	5.5	5.4	5.2	5.0	4.7	4.1	3.5	3.1	3.2	3.3	3.0	2.7	17.8
PSI = 30.0	4.1	4.0	3.8	3.6	3.4	3.2	3.0	2.6	2.1	1.8	1.9	2.0	1.8	1.6	11.1
PSI = 45.0	2.6	2.4	2.2	2.0	1.8	1.6	1.4	1.2	0.9	0.8	0.8	0.8	0.8	0.7	5.4
PSI = 60.0	1.7	1.5	1.3	1.1	1.0	0.8	0.6	0.5	0.3	0.2	0.2	0.2	0.2	0.1	1.4
PSI = 75.0	1.2	1.0	0.8	0.6	0.4	0.3	0.2	0.1	0.0	-0.1	-0.1	-0.1	-0.1	-0.1	-0.5
PSI = 90.0	1.0	0.9	0.7	0.6	0.4	0.3	0.2	0.1	0.0	0.0	0.0	0.0	-0.1	-0.1	-0.1
PSI = 105.0	1.4	1.2	1.0	0.9	0.7	0.7	0.6	0.5	0.5	0.3	0.3	0.2	0.1	0.1	0.9
PSI = 120.0	2.0	1.8	1.7	1.6	1.5	1.4	1.3	1.1	1.0	0.7	0.6	0.5	0.4	0.3	2.5
PSI = 135.0	2.9	2.8	2.7	2.6	2.4	2.2	2.0	1.9	1.7	1.3	1.0	0.8	0.7	0.5	4.9
PSI = 150.0	4.2	4.2	4.0	3.7	3.5	3.2	3.0	2.9	2.6	2.0	1.6	1.3	1.1	0.8	7.7
PSI = 165.0	5.9	5.6	5.3	5.0	4.7	4.4	4.1	4.0	3.8	3.0	2.3	2.0	1.6	1.2	11.0
PSI = 180.0	7.5	7.2	6.8	6.4	6.0	5.7	5.4	5.3	5.0	4.0	3.2	2.7	2.3	1.7	14.5
PSI = 195.0	9.1	8.7	8.3	7.9	7.5	7.1	6.8	6.7	6.4	5.2	4.1	3.5	3.0	2.2	18.3
PSI = 210.0	10.6	10.1	9.7	9.3	8.9	8.5	8.2	8.1	7.7	6.3	5.1	4.4	3.7	2.8	22.0
PSI = 225.0	12.0	11.5	11.0	10.6	10.2	9.8	9.5	9.3	8.8	7.3	6.1	5.4	4.6	3.5	25.6
PSI = 240.0	13.0	12.5	12.1	11.7	11.3	10.9	10.6	10.3	9.7	8.2	7.1	6.4	5.5	4.2	28.8
PSI = 255.0	13.6	13.2	12.9	12.6	12.3	11.9	11.6	11.1	10.2	8.8	7.9	7.3	6.3	5.0	31.7
PSI = 270.0	18.1	16.5	13.4	13.0	13.0	12.8	12.5	11.6	10.3	9.1	8.5	8.0	7.1	5.6	34.2
PSI = 285.0	12.8	13.3	12.3	12.6	12.7	12.6	12.4	11.5	10.1	9.1	8.8	8.5	7.5	6.1	36.5
PSI = 300.0	11.5	11.6	12.0	11.8	11.7	11.5	11.2	10.3	9.0	8.2	8.1	8.1	7.3	6.0	37.1
PSI = 315.0	10.2	10.6	10.7	10.7	10.6	10.5	10.2	9.3	8.0	7.3	7.4	7.5	6.7	5.7	35.2
PSI = 330.0	8.1	9.2	9.6	9.8	9.8	9.7	9.4	8.5	7.2	6.6	6.9	7.1	6.4	5.4	34.1
PSI = 345.0	8.7	8.5	8.6	8.6	8.6	8.5	8.2	7.2	6.1	5.6	5.8	6.1	5.5	4.7	30.7
PSI = 360.0	7.1	7.5	7.4	7.3	7.1	6.8	6.5	5.8	4.8	4.3	4.5	4.7	4.3	3.8	24.7

SUCCESSIVE SUBSTITUTION ITERATION, NO VARIABLES, NUMBER OF LEVELS = 2, LOOP = WAKE
ITERATION, COUNT = 1, FOR LEVEL 1 = UNIFORM INFLOW ; LOOP = WAKE

BOUND CIRCULATION PEAKS (ENTIRE BLADE)

	BOUND CIRCULATION			RADIAL STATION			WAKE GEOMETRY
AZIMUTH (DEG)	GMAX	GO	GI	RMAX	RO	RI	RGI
PSI = 0.0	0.03448	0.03448	-0.00058	0.89000	0.89000	0.25000	-0.60000
PSI = 15.0	0.02837	0.02837	0.00000	0.89000	0.89000	-0.60000	-0.60000
PSI = 30.0	0.01968	0.01968	0.00000	0.89000	0.89000	-0.60000	-0.60000
PSI = 45.0	0.01032	0.01032	0.00000	0.89000	0.89000	-0.60000	-0.60000
PSI = 60.0	0.00715	-0.00005	0.00715	0.56000	0.97000	0.56000	-0.60000
PSI = 75.0	0.00638	-0.00315	0.00638	0.49500	0.95000	0.49500	-0.60000
PSI = 90.0	0.00694	-0.00328	0.00694	0.49500	0.97000	0.49500	-0.60000
PSI = 105.0	0.00845	-0.00103	0.00845	0.56000	0.97000	0.56000	-0.60000
PSI = 120.0	0.01064	0.01064	0.00000	0.66500	0.66500	-0.60000	-0.60000
PSI = 135.0	0.01564	0.01564	0.00000	0.91000	0.91000	-0.60000	-0.60000
PSI = 150.0	0.02050	0.02050	0.00000	0.91000	0.91000	-0.60000	-0.60000
PSI = 165.0	0.02478	0.02478	0.00000	0.93000	0.93000	-0.60000	-0.60000
PSI = 180.0	0.02877	0.02877	0.00000	0.93000	0.93000	-0.60000	-0.60000
PSI = 195.0	0.03117	0.03117	0.00000	0.95000	0.95000	-0.60000	-0.60000
PSI = 210.0	0.03228	0.03228	-0.00475	0.95000	0.95000	0.25000	-0.60000
PSI = 225.0	0.03178	0.03178	-0.00438	0.95000	0.95000	0.34000	0.25000
PSI = 240.0	0.03040	0.03040	-0.01064	0.95000	0.95000	0.34000	0.25000
PSI = 255.0	0.02904	0.02904	-0.00728	0.95000	0.95000	0.42000	0.25000
PSI = 270.0	0.02802	0.02802	-0.00334	0.95000	0.95000	0.49500	0.25000
PSI = 285.0	0.02842	0.02842	-0.00813	0.91000	0.91000	0.42000	0.25000
PSI = 300.0	0.03042	0.03042	-0.01571	0.89000	0.89000	0.34000	0.25000
PSI = 315.0	0.03322	0.03322	-0.00968	0.89000	0.89000	0.34000	0.25000
PSI = 330.0	0.03514	0.03514	-0.01169	0.89000	0.89000	0.25000	-0.60000
PSI = 345.0	0.03591	0.03591	-0.00541	0.89000	0.89000	0.25000	-0.60000
PSI = 360.0	0.03448	0.03448	-0.00058	0.89000	0.89000	0.25000	-0.60000

GO = OUTBOARD PEAK, GI = INBOARD PEAK, GMAX = PEAK WITH MAXIMUM MAGNITUDE
CORRESPONDING RADIAL STATIONS OF PEAKS ARE RO,RI,RMAX; VALUE OUTSIDE BLADE IF PEAK NOT FOUND
RGI = SPAN STATION OF INBOARD CIRCULATION PEAK
FOR DUAL PEAK WAKE MODEL: CONVERGENCE OF WAKE GEOMETRY ITERATION REQUIRES RGI = RI

GAMMA = BOUND CIRCULATION / OMEGA*R**2
REFERENCE ROTOR RADIUS R = 24.7240 FT, ROTATIONAL SPEED OMEGA = 260.000 RPM; TIP SPEED VTIP = OMEGA*R

SUCCESSIVE SUBSTITUTION ITERATION, NO VARIABLES, NUMBER OF LEVELS = 2, LOOP = WAKE
ITERATION, COUNT = 1, FOR LEVEL 2 = NONUNIFORM INFLOW PRSC WAKE GEOM; LOOP = WAKE
==

BOUND CIRCULATION PEAKS (ENTIRE BLADE)

	BOUND CIRCULATION			RADIAL STATION			WAKE GEOMETRY
AZIMUTH (DEG)	GMAX	GO	GI	RMAX	RO	RI	RGI
PSI = 0.0	0.02641	0.02641	0.00000	0.89000	0.89000	-0.60000	0.25000
PSI = 15.0	0.02287	0.02287	0.00000	0.87000	0.87000	-0.60000	-0.60000
PSI = 30.0	0.01707	0.01707	0.00000	0.89000	0.89000	-0.60000	-0.60000
PSI = 45.0	0.01002	0.01002	0.00000	0.71000	0.71000	-0.60000	-0.60000
PSI = 60.0	0.00758	0.00758	0.00000	0.61500	0.61500	-0.60000	0.56000
PSI = 75.0	0.00631	-0.00073	0.00631	0.42000	0.99000	0.42000	0.49500
PSI = 90.0	0.00588	0.00588	0.00000	0.49500	0.49500	-0.60000	0.49500
PSI = 105.0	0.00756	0.00756	0.00000	0.49500	0.49500	-0.60000	0.56000
PSI = 120.0	0.00967	0.00967	0.00000	0.75000	0.75000	-0.60000	-0.60000
PSI = 135.0	0.01329	0.01329	0.00000	0.71000	0.71000	-0.60000	-0.60000
PSI = 150.0	0.01639	0.01639	0.00000	0.75000	0.75000	-0.60000	-0.60000
PSI = 165.0	0.01872	0.01872	0.00000	0.75000	0.75000	-0.60000	-0.60000
PSI = 180.0	0.02038	0.02038	0.00000	0.91000	0.91000	-0.60000	-0.60000
PSI = 195.0	0.02195	0.02195	0.00000	0.93000	0.93000	-0.60000	-0.60000
PSI = 210.0	0.02276	0.02276	0.00000	0.93000	0.93000	-0.60000	0.25000
PSI = 225.0	0.02302	0.02302	0.00000	0.93000	0.93000	-0.60000	0.34000
PSI = 240.0	0.02266	0.02266	0.00000	0.93000	0.93000	-0.60000	0.34000
PSI = 255.0	0.02252	0.02252	0.00000	0.93000	0.93000	-0.60000	0.42000
PSI = 270.0	0.02346	0.02346	-0.00151	0.87000	0.87000	0.34000	0.49500
PSI = 285.0	0.02414	0.02414	-0.00347	0.89000	0.89000	0.34000	0.42000
PSI = 300.0	0.02338	0.02338	-0.00442	0.89000	0.89000	0.34000	0.34000
PSI = 315.0	0.02486	0.02486	-0.00218	0.87000	0.87000	0.42000	0.34000
PSI = 330.0	0.02662	0.02662	0.00000	0.89000	0.89000	-0.60000	0.25000
PSI = 345.0	0.02893	0.02893	-0.00433	0.87000	0.87000	0.34000	0.25000
PSI = 360.0	0.02641	0.02641	0.00000	0.89000	0.89000	-0.60000	0.25000

GO = OUTBOARD PEAK, GI = INBOARD PEAK, GMAX = PEAK WITH MAXIMUM MAGNITUDE
CORRESPONDING RADIAL STATIONS OF PEAKS ARE RO,RI,RMAX; VALUE OUTSIDE BLADE IF PEAK NOT FOUND
RGI = SPAN STATION OF INBOARD CIRCULATION PEAK
FOR DUAL PEAK WAKE MODEL: CONVERGENCE OF WAKE GEOMETRY ITERATION REQUIRES RGI = RI

GAMMA = BOUND CIRCULATION / OMEGA*R**2
REFERENCE ROTOR RADIUS R = 24.7240 FT, ROTATIONAL SPEED OMEGA = 260.000 RPM; TIP SPEED VTIP = OMEGA*R

SUCCESSIVE SUBSTITUTION ITERATION, NO VARIABLES, NUMBER OF LEVELS =   2, LOOP = WAKE
ITERATION, COUNT =   2, FOR LEVEL   2 = NONUNIFORM INFLOW PRSC WAKE GEOM; LOOP = WAKE

BOUND CIRCULATION PEAKS (ENTIRE BLADE)

	BOUND CIRCULATION			RADIAL STATION			WAKE GEOMETRY
AZIMUTH (DEG)	GMAX	GO	GI	RMAX	RO	RI	RGI
PSI = 0.0	0.02538	0.02538	0.00000	0.87000	0.87000	-0.60000	-0.60000
PSI = 15.0	0.02162	0.02162	0.00000	0.84500	0.84500	-0.60000	-0.60000
PSI = 30.0	0.01593	0.01593	0.00000	0.78500	0.78500	-0.60000	-0.60000
PSI = 45.0	0.01017	0.01017	0.00000	0.71000	0.71000	-0.60000	-0.60000
PSI = 60.0	0.00761	0.00761	0.00000	0.56000	0.56000	-0.60000	-0.60000
PSI = 75.0	0.00617	-0.00019	0.00617	0.42000	0.99000	0.42000	0.42000
PSI = 90.0	0.00569	0.00569	0.00000	0.49500	0.49500	-0.60000	-0.60000
PSI = 105.0	0.00750	0.00750	0.00000	0.49500	0.49500	-0.60000	-0.60000
PSI = 120.0	0.00962	0.00962	0.00000	0.75000	0.75000	-0.60000	-0.60000
PSI = 135.0	0.01329	0.01329	0.00000	0.71000	0.71000	-0.60000	-0.60000
PSI = 150.0	0.01630	0.01630	0.00000	0.71000	0.71000	-0.60000	-0.60000
PSI = 165.0	0.01863	0.01863	0.00000	0.78500	0.78500	-0.60000	-0.60000
PSI = 180.0	0.02040	0.02040	0.00000	0.89000	0.89000	-0.60000	-0.60000
PSI = 195.0	0.02203	0.02203	0.00000	0.91000	0.91000	-0.60000	-0.60000
PSI = 210.0	0.02294	0.02294	0.00000	0.91000	0.91000	-0.60000	-0.60000
PSI = 225.0	0.02334	0.02334	0.00000	0.91000	0.91000	-0.60000	-0.60000
PSI = 240.0	0.02338	0.02338	0.00000	0.91000	0.91000	-0.60000	-0.60000
PSI = 255.0	0.02353	0.02353	0.00000	0.91000	0.91000	-0.60000	-0.60000
PSI = 270.0	0.02380	0.02380	-0.00235	0.91000	0.91000	0.34000	0.34000
PSI = 285.0	0.02444	0.02444	-0.00358	0.89000	0.89000	0.34000	0.34000
PSI = 300.0	0.02391	0.02391	-0.00451	0.89000	0.89000	0.34000	0.34000
PSI = 315.0	0.02433	0.02433	-0.00200	0.89000	0.89000	0.42000	0.42000
PSI = 330.0	0.02622	0.02622	0.00000	0.87000	0.87000	-0.60000	-0.60000
PSI = 345.0	0.02680	0.02680	-0.00317	0.87000	0.87000	0.25000	0.34000
PSI = 360.0	0.02538	0.02538	0.00000	0.87000	0.87000	-0.60000	-0.60000

GO = OUTBOARD PEAK, GI = INBOARD PEAK, GMAX = PEAK WITH MAXIMUM MAGNITUDE
CORRESPONDING RADIAL STATIONS OF PEAKS ARE RO,RI,RMAX; VALUE OUTSIDE BLADE IF PEAK NOT FOUND
RGI = SPAN STATION OF INBOARD CIRCULATION PEAK
FOR DUAL PEAK WAKE MODEL: CONVERGENCE OF WAKE GEOMETRY ITERATION REQUIRES RGI = RI

GAMMA = BOUND CIRCULATION / OMEGA*R**2
REFERENCE ROTOR RADIUS R =   24.7240 FT, ROTATIONAL SPEED OMEGA = 260.000 RPM; TIP SPEED VTIP = OMEGA*R

Chapter 8

ROTORCRAFT INPUT

GENERAL REFERENCE FOR CHAPTER: CAMRAD II Documentation, Volume VI, Rotorcraft Input

ROTORCRAFT SHELL INPUT PARAMETERS

- COMPLETE DESCRIPTION IN ROTORCRAFT INPUT MANUAL

INPUT PARAMETERS ORGANIZED BY SHELL CLASS AND TYPE

- ALL SHELL INPUT BLOCKS NEEDED BY JOB MUST BE DEFINED BY USER
  - even if only default parameter values are required
  - usually include all blocks that might be needed in shell input file, at least with default values

USER SHOULD KNOW WHAT OPTIONS ARE AVAILABLE THROUGH INPUT PARAMETERS

- rely on input manual for details

JAVA PROGRAM AVAILABLE TO CREATE AND EDIT ROTORCRAFT SHELL INPUT

UNITS OF INPUT PARAMETERS

CONVENTION:

english or metric (SI) units, using consistent length-mass-time system

foot-slug-second or meter-kilogram-second

angles are input in degrees

EXCEPTIONS TO THIS CONVENTION IN ROTORCRAFT SHELL:

aircraft gross weight is in lb or kg

wind speed and aircraft speed can be in knots

8–1 CAMRAD II Analysis

NAMELIST NLJOB PARAMETERS:

NCASES = number of cases
PLFILE = 1 to write plot file for job
OPSHLL = shell input file read

defaults: one case, no plot file written, shell input file read for first case only (OPSHLL = 1)

OPINIT = initialization from previous case

trim solution can be initialized from previous case using the trim loop variables (OPINIT = 1); or using the trim part total solution (OPINIT = 6); or both (OPINIT = 7); default is no initialization

default values generally used for other parameters

8–2 Class = CASE

CASE DESCRIPTION

OPUNIT = identify units (English or metric)

TMTASK = select trim task
TNTASK = select transient task
FLTASK = select flutter task

skip, execute, or initialize

CONTROL PRINT OF INPUT DATA

usually do not print all core input or derived data (too much information)

PLUGINS

PLUGIN = run installed shell plugins

TIMER AND DEBUG PARAMETERS

ENVIRONMENT PARAMETERS

AERODYNAMIC

EARTH (GENERALLY USE DEFAULTS)

8–3 Class = TRIM

OPERATING CONDITION

initial values required for variables of trim loop; remaining parameters specify operating condition

FLIGHT SPEED AND WIND SPEED

four input formats; other three input values ignored

wind speed usually zero for free flight operation, but can be used

only wind speed used for wind tunnel operation, flight speed ignored

ROTOR ROTATIONAL SPEED

four input formats; other input values ignored

CONTROL SETTINGS

SOLUTION PROCEDURE

WAKE LOOP

LEVEL = wake analysis level

uniform inflow
nonuniform inflow, prescribed wake geometry
nonuniform inflow, free wake geometry

any level except last executed can be skipped

typically skip levels when trim solution initialized from previous job (OPITJR $>$ 0) or from previous case

TRIM LOOP

OPTRIM = 0 for no trim, 1 for trim iteration

MTRIM, MNAME, VNAME define trim problem

trim variables VNAME: convenient to identify using name of corresponding initial value parameter

can also use descriptive name, or name of symbol

trim quantities MNAME: can identify using descriptive name, name of symbol, or name of corresponding target value parameter

typical free flight trim:

```
MTRIM = 6,
VNAME = 'COLL','LATCYC','LNGCYC',
'PEDAL','PITCH','ROLL',
MNAME = 'FORCE X','FORCE Y','FORCE Z',
'MOMENT X','MOMENT Y','MOMENT Z',
```

typical wind tunnel trim:

```
MTRIM = 3,
VNAME = 'COLL','LATCYC','LNGCYC',
MNAME = 'CT/S','BETAS','BETAC',
```

can select flight speed, wind speed, or rotor rotational speed as trim variable

with drive train model, should trim torque with throttle (rotor speed perturbation is additional system rigid degree of freedom)

PART SOLUTIONS

MPSI = number of azimuth steps per revolution

shell generally uses same $\Delta\psi$ throughout solution

motion solution can use MPSIR < MPSI
output solution can use MxTIME

OPPER = common period and full interaction

default is to suppress vibratory interaction at inconsistant period

OPPART = rotor part definition

solve only reference blade (N-th blade): assuming identical motion

solve each blade separately: needed if system changed (using core input) so motion of blades no longer identical

solve all blades together: needed if blades are structurally coupled

analysis requires this option for rotor with gimbal or teeter joint

perhaps needed with flexible swashplate

consistently set filter of response, using OPFLTx

OPCHLD = identical motion of all blades

option if solve all blades (OPPART = 3)

harmonic solution for blade degree of freedom replaced by parent, with phase shift; parent is average (with phase shifts) of solution for all blades; for gimbal and teeter degree of freedom, also only $pN \pm 1$ harmonics nonzero

METHOD = rotor part solution method

harmonic or time finite element

EXTERNAL AEROACOUSTIC ANALYSIS

OPPOST = post-trim solution

for high resolution, or to obtain partial angle-of-attack

OPEXT of rotor aerodynamics for prescribed coefficient increments

OPEXT of rotor wake for partial angle-of-attack

MPSIH = resolution in time and wake age

DEGREES OF FREEDOM

DOFx = zero, dynamic, or quasistatic

OUTPUT

TRACEL = 2 and TRACEP = 3 to follow convergence

and use TIMERP of class = CASE to see execution of all parts

always get trace of trim loop

NAPRNT, NDPRNT, NRPRNT: control print of part solution (degree of freedom and constraint variables)

generally not used (blade position sensor gives physically meaningful motion)

NWPRNT = 1 for output from each wake iteration

SENSORS

response calculated:

airframe sensors (MSSEN)
frame motion

output selection: NxPRNT, NxFILE, NxPLOT
number of azimuth steps: MxTIME
number of harmonics: MxHARM

harmonics calculated from time history at MxTIME azimuth steps

ELEMENTARY SCENARIO

using initial values

azimuth step $\Delta\psi = 15$ deg
solve reference blade only

no airframe or drive train degrees of freedom

8–4 Class = TRIM ROTOR

CONTROL SETTINGS

COMPONENT CONTROLS
HIGHER HARMONIC CONTROL
HIGHER HARMONIC AUXILIARY FORCES

SOLUTION PROCEDURE

DEGREES OF FREEDOM

DOFx = zero, dynamic, or quasistatic

use DOFB and/or DOFM, depending on OPMODE

should not use
DOFS = 0 with swashplate not locked; or
DOFD = 0 with trailing-edge flap not locked; or
DOFB(3) = 0 with pitch bearing not locked:

get zero motion, so zero control

instead use quasistatic motion (DOFx = 2)
or locked structure (LOCKPx = 0)

similar issue when use blade modes:
should not use DOFM = 0 for blade pitch mode

option to use multiblade coordinates

use of transformed degree of freedom determined
by product of appropriate DOFx and DOFMBC

OUTPUT

HUB AND BLADE SENSORS

response calculated:

hub load sensors (MHSEN)
control load sensors (MCSEN)
blade load sensors (MBSEN)
blade position sensors (MPSEN)
blade aerodynamics (MASEN)
wake velocity off rotor (MVSEN)
wake geometry (MWSEN)

output selection: NxPRNT, NxFILE, NxPLOT
number of azimuth steps: MxTIME
number of harmonics: MxHARM

harmonics calculated from time history at MxTIME azimuth steps

ELEMENTARY SCENARIO

using initial values

only flap degree of freedom

8–5 Class = TRANSIENT

TRANSIENT CONDITION

TIMEB to TIMEE = time range of transient solution

SEPARATE PRESCRIBED CONTROL COMPONENT CONSTRUCTED FOR EACH ELEMENT OF INPUT

constructed with same time history kind, as specified by shell input

amplitudes specified by control settings in shell input

use core input for different time history kind for particular element

CONTROL SETTINGS

SOLUTION PROCEDURE

parameters controlling convergence and accuracy of transient solution have already been discussed

PART SOLUTIONS

METHOD and OPINIT define integration method

TRESP = response time step, Δt

to solve implicit parts, and save integration solution

TSTEPS = numerical integration time increment, δt

determines accuracy of integration

DEGREES OF FREEDOM

DOFx = zero, dynamic, or quasistatic

OUTPUT

TRACEP = 2 to follow progress of transient integration and assess convergence

NPPRNT, NAPRNT, NRPRNT: control print of part solution (degree of freedom and constraint variables)

generally not used (blade position sensor gives physically meaningful motion)

SENSORS

response calculated:

airframe sensors (MSSEN)

output selection: NxPRNT, NxFILE, NxPLOT
time step: TxOUT

should calculate output for trim also

ELEMENTARY SCENARIO

using initial values

implicit filters

no airframe or drive train degrees of freedom

8–6 Class = TRANSIENT ROTOR

CONTROL SETTINGS

AERODYNAMIC MODEL

OPWAKE = rotor wake model

OPGEOM = wake geometry model

SOLUTION PROCEDURE

DEGREES OF FREEDOM

DOFx = zero, dynamic, or quasistatic

use DOFB and/or DOFM, depending on OPMODE

option to use multiblade coordinates

use of transformed degree of freedom determined by product of appropriate DOFx and DOFMBC

OUTPUT

HUB AND BLADE SENSORS

response calculated:

hub load sensors (MHSEN)
control load sensors (MCSEN)
blade load sensors (MBSEN)
blade position sensors (MPSEN)
blade aerodynamics (MASEN)
wake velocity off rotor (MVSEN)
wake geometry (MWSEN)

output selection: NxPRNT, NxFILE, NxPLOT
time step: TxOUT

should calculate output for trim also

ELEMENTARY SCENARIO

using initial values

dynamic inflow, with aerodynamic interference

only flap degree of freedom
only collective and cyclic multiblade coordinates

8–7 Class = FLUTTER

TASK DEFINITION

OPFLUT = time invariant or periodic

OPMEAN = constant coefficient approximation (averaged)

generally analyze time invariant equations, averaged in forward flight

MPSIAV = number of steps per rev in average

OPBLD = independent blade analysis

typically used for free vibration calculations

OPSTAB = flight dynamics analysis (quasistatic reduction of all states except aircraft rigid body motion)

should print stability derivatives as well (by printing loop equations)

AERODYNAMIC MODEL

OPAERO = airframe aerodynamics

can trim in air, but obtain flutter equations without aerodynamics (free vibration)

SOLUTION PROCEDURE

parameters controlling accuracy of flutter solution have already been discussed

MPSIAV = number of steps per rev in average

FLUTTER VARIABLES

DOFx = degrees of freedom (zero, dynamic, or quasistatic)

CONx = controls (zero, used)
GUST = gust (zero, used)

MxSEN = output (zero, used)

DOFORD can be used to override internal determination of order of degree of freedom (based on columns of mass and spring matrices)

option to use symmetric/antisymmetric transform (for tiltrotor)

use of transformed degree of freedom determined by product of appropriate DOFx and DOFSYM, etc

OUTPUT

NxPRNT: control print of loop solution for linearized equations

allows examination of subsystem equations

flutter analysis can also print final system equations

ANALYSIS OF FLUTTER EQUATIONS

TASK = analysis tasks

OPEQN = equation sets analyzed

eigenanalysis, time history response, frequency response, rms gust response

for nine subsets of flutter equations

OUTPUT SELECTION

can print flutter equations, or send to plot file for use outside CAMRAD II

VARIABLES AS OUTPUT

can select degrees of freedom and controls as output variables in flutter analysis

must identify by name: run analysis to get list of names in flutter task "DESCRIPTION OF VARIABLES"

PERIODIC EQUATIONS

evaluate and interpolate equations: MPSI, OPNTRP, MHARM

integrate equations: MSTEP, TSTEP

EIGENANALYSIS

eigenvalues are most common product of flutter analysis

with experience, simplest to identify modes from eigenvalues (frequency and damping)

eigenvectors of states are influenced by normalization and units of degrees of freedom

eigenvectors of output are more useful

TIME HISTORY RESPONSE

more efficient than transient task

FREQUENCY RESPONSE

RMS GUST RESPONSE

rms response not properly defined for system with unstable modes

ELEMENTARY SCENARIO

using initial values

six rigid body degrees of freedom (free flight)
quasistatic filters
analysis of equations: eigenvalues only

no drive train degrees of freedom
print rotorcraft equations and matrices of flutter equations

8–8 Class = FLUTTER ROTOR

AERODYNAMIC MODEL

OPAERO = rotor aerodynamics

can trim in air, but obtain flutter equations without aerodynamics (free vibration)

OPWAKE = rotor wake model

trim inflow: inflow not perturbed at all

dynamic inflow: DOFL controls inflow states

uniform inflow: perturbed trim model, similar to quasistatic dynamic inflow

lagged inflow: perurbed trim model with time lag, DOFL controls inflow state

SOLUTION PROCEDURE

FLUTTER VARIABLES

DOFx = degrees of freedom (zero, dynamic, or quasistatic)

CONx = controls (zero, used)

MxSEN = output (zero, used)

use DOFB and/or DOFM, depending on OPMODE

option to use multiblade coordinates

use of transformed degree of freedom determined by product of appropriate DOFx and DOFMBC, etc

rotor in forward flight has periodic equations from aerodynamics of edgewise flow; with multiblade coordinates, constant coefficient approximation (giving time invariant equations) is adequate up to advance ratio of about $\mu = 0.5$

ELEMENTARY SCENARIO

using initial values

dynamic inflow (but DOFL = 0), with aerodynamic interference

only flap degree of freedom

8–9 Class = AIRFRAME, Type = STRUCTURE

ROTORCRAFT DESCRIPTION

CONFIG = configuration
OPFREE = wind tunnel (constrained) or free flight
OPTRAN = drive train model
OPAERO = airframe aerodynamics
OPFIX = fixed wing

ROTORCRAFT INERTIA

EQUIVALENT HUB INERTIA

gross weight and center of gravity

input for entire system, including rotors

moments of inertia and elastic mode generalized masses

input for rotorcraft system with:

all mass defined by rotor input parameters (blade and flexbeam mass, point masses) removed

“equivalent hub inertia” added

often MASSR = total rotor inertia (blade and flexbeam mass and point masses, for all blades)

analysis prints calculated blade mass M_b (can use to define MASSR = NM_b)

LOCATION OF ROTORCRAFT COMPONENTS

GENERAL LOCATIONS

first NROTOR locations: rotor hubs
next NROTOR locations: rotor swashplates

only need OPAXES for these locations

swashplate locations created even if rotor model does not have swashplate

location number $>$ 2NROTOR: define as required for swashplate nonrotating actuators, auxiliary forces, tail boom, sensors, slung loads, airframe wings and bodies, and rotor-induced velocity calculation

ROTOR PYLON

ELASTIC AIRFRAME

SOURCE = mode properties from namelist arrays or table file

mode shapes, LSHAPE and ASHAPE

namelist input: use one line (x, y, and z components) for each location and each mode; group by location; see sample jobs for example

with more than 40 locations or more than 40 modes, must use table file

aerodynamic control coefficient QAEROC

AIRFRAME AUXILIARY FORCES

TAIL BOOM

OPTAIL = configuration (circulation-control boom, reaction jet)

definition of tail boom parameters completed using core input

SLUNG LOAD

OPSLNG = model (rigid load, elastic load)

core input may be required to complete slung load model

AIRFRAME SENSORS

OUTPUT SELECTED SEPARATELY FOR TRIM, TRANSIENT, AND FLUTTER TASKS

ELEMENTARY SCENARIO

using initial values

constant rotational speed
no airframe elastic modes
no auxiliary forces
no sensors

wing-body center of action at rotorcraft center of gravity

8–10 Class = AIRFRAME, Type = AERODYNAMICS

RIGID AIRFRAME AERODYNAMICS

TNMODL and FLMODL = model for transient and flutter tasks

NONLINEAR AERODYNAMIC MODEL

NLSCR = from namelist variables (equations) or table file

table kind (2D or 3D or 4D) identified by NLTABL in class = TABLES input

some derivatives from namelist variables are used with tables

STABILITY DERIVATIVE AERODYNAMIC MODEL

SDSRC = from namelist variables or table file

AIRFRAME FLOW FIELD

VISRC = interference from calculation or table file (0 for no interference)

ELEMENTARY SCENARIO

using initial values

nonlinear aerodynamic model, from namelist variables
no airframe flow field

minimum input:

DRG0W = wing-body drag, $f = D_0/q$ (ft^2 or m^2)
DRGVW = vertical drag, $g = D_V/q$ (ft^2 or m^2)

required for correct rotor propulsive force

LFTAH, or MOMAW including tail effect; required for correct angle-of-attack stability in flight dynamics

8–11 Class = AIRFRAME, Type = CONTROL

CONTROL MATRIX INPUT FORMAT

TCxIN = calculated or input T_C matrix

typically use calculated matrix, unless not adequate to represent connection of redundant controls to pilot's controls

CONTROL MATRIX

CONTROL MATRIX CALCULATION

SWASHPLATE GAINS AND PHASE ANGLES

matrices defined in theory manual

ELEMENTARY SCENARIO

using initial values

calculated control matrix, from unit swashplate gains

8–12 Class = AIRFRAME, Type = DRIVE TRAIN

TRANSMISSION AND ENGINE

CONFIG = configuration
OPGOV = rotor speed governor

existence of drive train model determined by OPTRAN airframe structure input

GOVERNOR

FEEDBACK TO ROTOR COLLECTIVE OR ENGINE THROTTLE

careful with units of throttle

ELEMENTARY SCENARIO

using initial values

airframe structure scenario selects constant rotational speed (no drive train model)

8–13 Class = ROTOR, Type = STRUCTURE

ROTOR DESCRIPTION

GEOMETRY AND STRUCTURE

CONFIG = blade root configuration
OPAERO = blade aerodynamics
OPAZ = blade spacing

SHAFT = rotor shaft spring or free rotation
GIMBAL = gimbal/teeter joint
CONTRL = control system configuration
HINGE = flap and lag hinge configuration
PITCH = pitch bearing configuration
OPTEF = trailing-edge flaps

TIP PATH PLANE SENSOR

normal displacement at ETPP (should be inboard of swept tip), or flap hinge rotation, or gimbal/teeter hinge rotation

can use core input to replace with blade sensor

REFERENCE LINE

two straight segments, divided at droop-sweep node (which may be absent)

defined by torque offset, undersling, precone, droop, sweep, and span station of droop-sweep node

figure 1 shows geometry of blade and control system

all joints on first segment of reference line, inboard of droop-sweep node

PITCH BEARING ALIGNED WITH PRECONED REFERENCE LINE

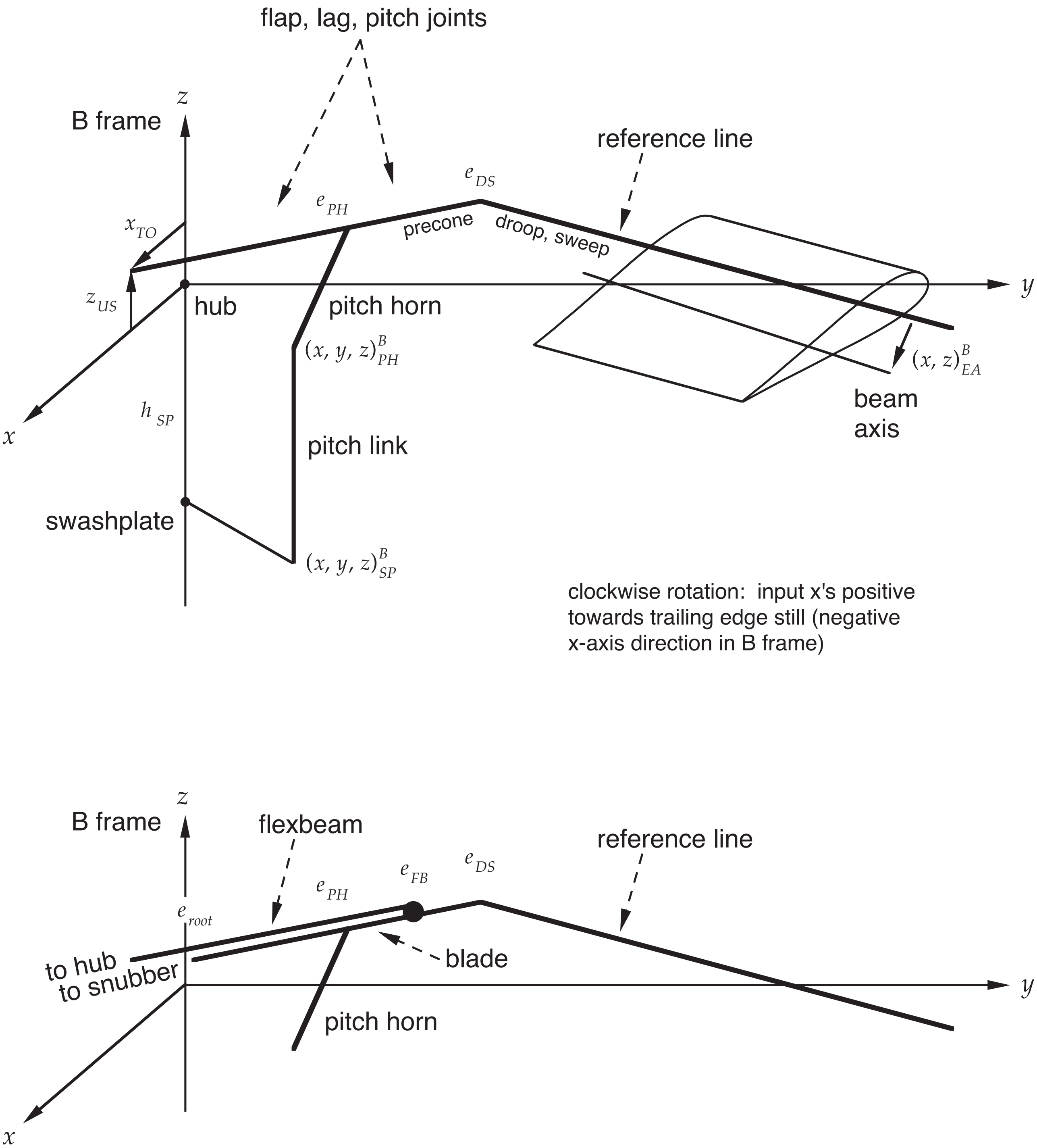


Figure 8-1a Geometry of blade and control system.

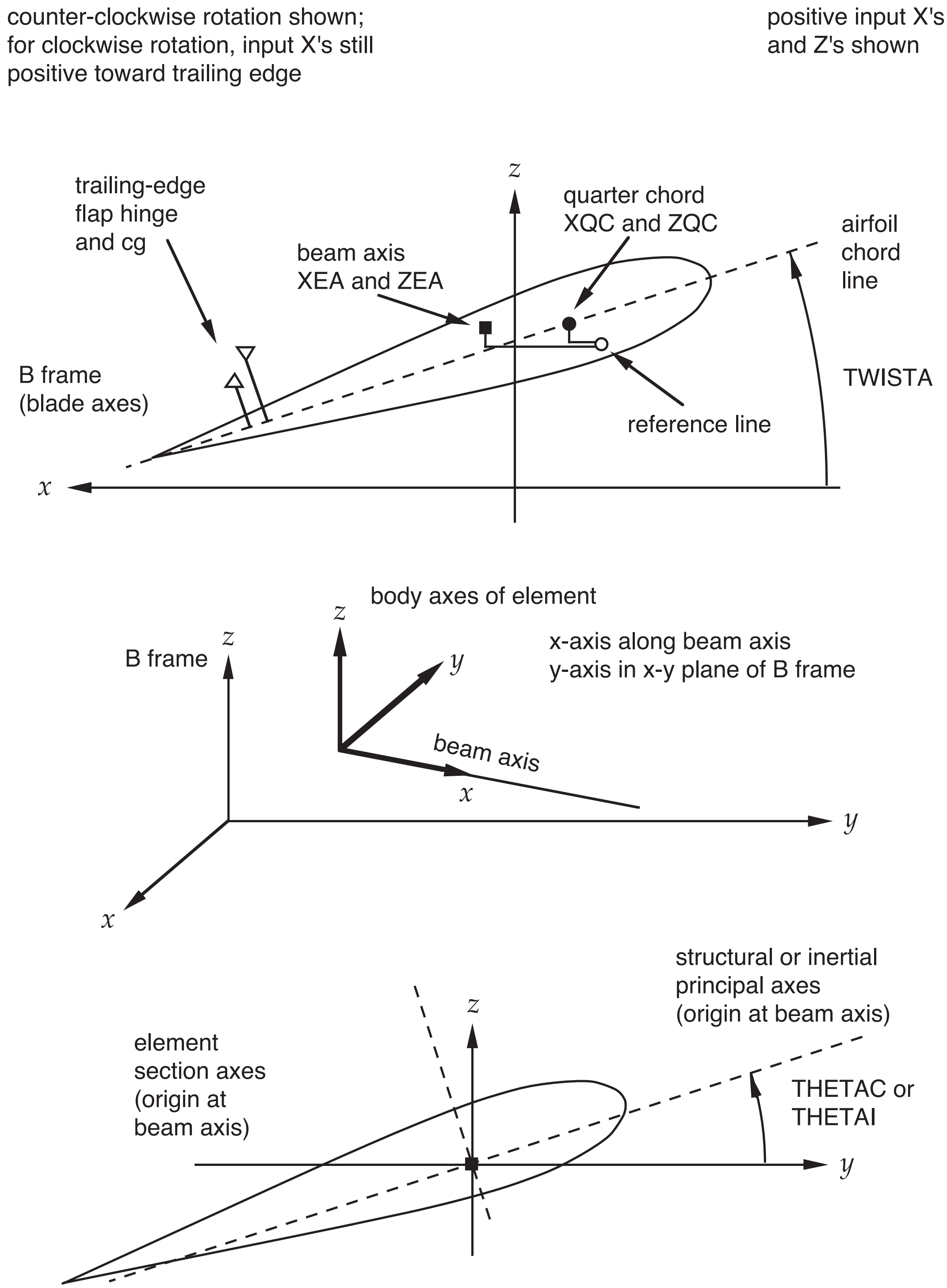
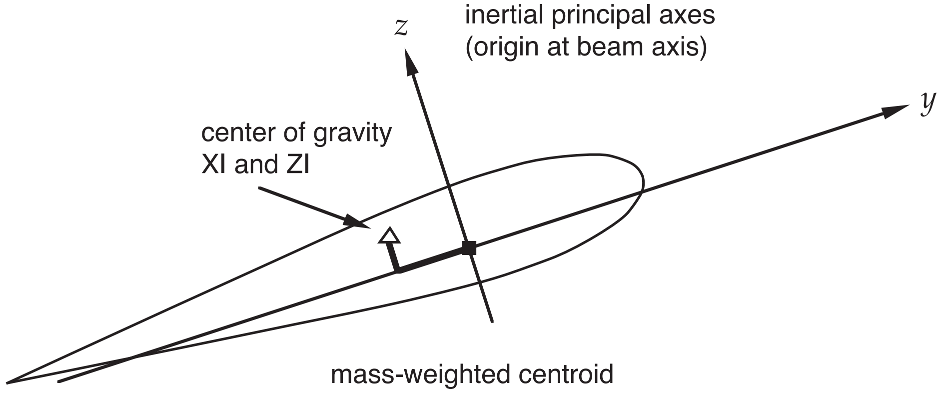


Figure 8-1b Geometry of blade and control system.

counter-clockwise rotation shown;
for clockwise rotation, input X's still
positive toward trailing edge

positive input X's
and Z's shown

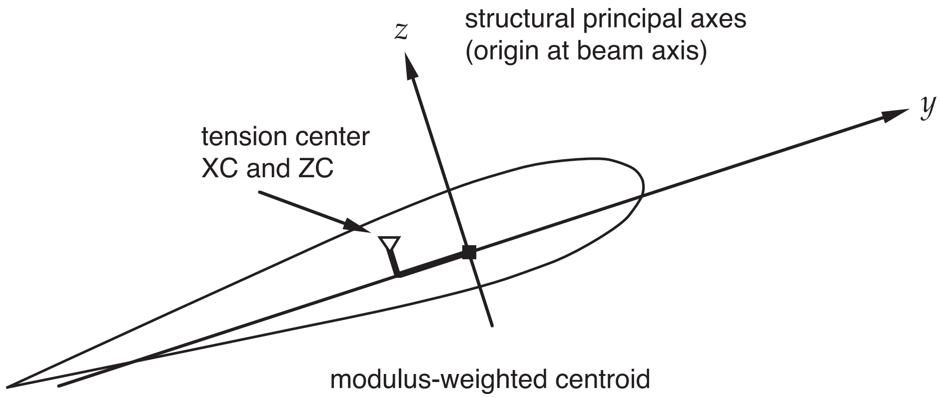


mass-weighted centroid

ITHETA $I_\theta = \int (y^2 + z^2)\, dm$

IPOLAR $I_P = \int (y^2 - z^2)\, dm$

about beam axis



modulus-weighted centroid

KP $EAk_P^2 = \int E\,(y^2 + z^2)\, dA$ about beam axis

EIFLAP $EI_{zz} = \int E\,(z - z_C)^2\, dA$

EILAG $EI_{yy} = \int E\,(y - y_C)^2\, dA$

about principal axes, relative tension center

Figure 8-1c Geometry of blade and control system.

BLADE SECTION PROPERTIES

INPUT AS PIECEWISE LINEAR FUNCTIONS OF SPAN STATION

	chordwise offset	normal offset	twist or pitch
beam axis	XEA	ZEA	
quarter chord	XQC	ZQC	TWISTA
tension center	XC	ZC	THETAC
center of gravity	XI	ZI	THETAI
structure	KP		
isotropic stiffness	EIFLAP, EILAG, EA, GJ, KT		
anisotropic stiffness	SUU, STU, SWU, SVU, STT		
anisotropic stiffness	SWT, SVT, SWW, SVW, SVV		
inertia	MASS, ITHETA, IPOLAR		

INPUT GEOMETRY

RADIUS in ft or m, all other lengths input as fraction radius

blade radial stations Exxx measured along y-axis of rotating blade frame (not along reference line)

input in axes of rotating blade frame: B axes

X chordwise ($+$ aft, towards trailing edge)
Y spanwise
Z normal ($+$ up)

frame origin is hub node (center of rotation)

y-axis is span station variable (and wing reference line) quantities measured in blade B axes, from hub node:

pitch link attachment to swashplate xSP
pitch link attachment to pitch horn xPH
torque offset XTO and undersling ZUS

quantities measured in blade B axes, from reference line:

locus of beam axis, XEA and ZEA
locus of quarter chord, XQC and ZQC
damper attachments, XDAMP and ZDAMP
locations of point masses, XPM and ZPM

aerodynamic twist TWISTA measured about wing reference line, from x–y plane of B frame

quarter chord locus and aerodynamic twist only required if blade aerodynamics used

chord measured in plane perpendicular to wing reference line (in x–z plane of B axes)

quantities measured in section principal axes, from (straight) beam axis of element:

tension center, XC and ZC
center of gravity, XI and ZI

pitch angle of principle axes (THETAC and THETAI) measured about beam axis, from element x–y plane

quantities measured in twisted aerodynamic axes, from quarter chord:

trailing-edge flap hinge (XHTEF and ZHTEF)
and center of gravity (XCGTEF and ZCGTEF)

quantities measured in hub S axes:

actuator attachment to swashplate (xACTSP)
actuator attachment to airframe (xACTAF)

CONVENTIONS

beam axis must be straight within each element (between nodes)

such a beam axis is constructed using input beam axis positions at 10% and 90% element length, so jumps at nodes are possible

most important that relative positions of center of gravity, quarter chord, and feathering axis are correct

section mass specified per unit length along span axis (not along element axis)

structural and inertial properties within elastic beam element only used at Gaussian integration points (and at sensor or interface location)

inertial properties of rigid element also calculated using Gaussian integration

POINT MASSES

REACTION JETS

BLADE ANALYSIS

OPBEAM = rigid body, rigid beam, or elastic beam elements

short elements usually rigid (DRELST sets criterion)

OPMAT = isotropic with elastic axis, or anisotropic/composite material

OPGEOM = second order, almost exact, or exact geometry

NDOFx = number of elastic degrees of freedom that exist

order reduction can be specified separately for trim, transient, and flutter tasks

KNODE and RNODE = specification of elements in blade structural model

OPWING = separate structural and wing components, or rigid wing component for last element (not recommended)

RIGID BLADE IS SIMPLEST MODEL:

OPBEAM = 0, KNODE = 0

BLADE POSITION SENSORS
BLADE CFD POSITION SENSORS
BLADE MODE SENSORS
BLADE LOADS SENSORS

OUTPUT SELECTED SEPARATELY FOR TRIM, TRANSIENT, AND FLUTTER TASKS

ELEMENTARY SCENARIO

RIGID, FLAPPING BLADE CONSTRUCTED

using initial values

single load path configuration
no swashplate mechanism, locked pitch bearing
flap hinge, no lag hinge
uniform properties, no point masses, linear twist
rigid body elements, no additional nodes
rigid wing model

structural dynamic properties estimated from section mass, blade radius, number of blades, normal tip speed, and solidity

required parameters:

TITLE, RADIUS, NBLADE, ROTATE, SIGMA, VTIPN
OPAERO (default 1)
EFLAP, KFLAP, DFLAP, AFLAP (defaults all 0.)
TWISTL
MASS (only first value used)

8–14 Class = ROTOR, Type = FLEXBEAM

BLADE GEOMETRY AND STRUCTURE

SNUB = snubber configuration

blade connected to hub or to flexbeam
snubber axes fixed to hub/flexbeam, or rotate with blade

CONFIGURATION

blade extends from EBLADE to tip (torque tube is inboard part of blade)

connected to flexbeam at EFB (inboard of droop-sweep node)

blade root (inboard end of torque tube) may be attached to hub or flexbeam through snubber, at radial station EROOT

flexbeam extends from hub to blade

swashplate required for pitch link (connected through pitch horn to torque tube) and control

FLEXBEAM SECTION PROPERTIES

INPUT GEOMETRY

quantities measured in blade B axes, from hub node:
snubber attachment to hub xSNUB

quantities measured in blade B axes, from reference line:
snubber attachment to flexbeam, xSNUB
snubber attachment to blade, xROOT

POINT MASSES

FLEXBEAM ANALYSIS

BLADE POSITION SENSORS
BLADE MODE SENSORS
BLADE LOADS SENSORS

OUTPUT SELECTED SEPARATELY FOR TRIM, TRANSIENT, AND FLUTTER TASKS

ELEMENTARY SCENARIO

using initial values

blade structure scenario selects single load patch configuration

8–15 Class = ROTOR, Type = AERODYNAMICS

AERODYNAMIC PANELS

DEFINED BY EDGES OF PANELS

at least two panels needed, in order to generate wing surface

aerodynamic collocation points are at midpoints of panels

REDGE(1) = blade root, REDGE(NPANEL+1) = blade tip

GEOMETRY

WING REFERENCE LINE

y-axis of rotating blade frame is wing reference line

not structural reference line (which is defined by precone, droop, sweep)

$r = y/R$ is span station variable

SPAN SCALE FACTOR

RSCALE = blade radius R (constant)

with large droop, use core input to define correct RSCALE, such that (RSCALE $\times$ Δr $\times$ CHORD) is panel area

AERODYNAMIC PROPERTIES

INPUT AS PIECEWISE LINEAR FUNCTIONS OF SPAN STATION

used at collocation points, wing tips, and other points as required

CHORD = wing chord (ft or m)

chord measured in plane perpendicular to wing reference line (in x–z plane of B axes)

ASWEEP = sweep angle of quarter chord

angle between tangent to quarter chord locus and tangent to wing reference line

required for swept/yawed flow model (OPYAW); in particular, controls use of Mach number normal to swept quarter chord

TAF, XAF, XAFF = airfoil table references

positions airfoil section relative quarter chord and aerodynamic twist, consistent with definition of zero angle of attack and moment axis in airfoil table

AERODYNAMIC MODEL

RTVTX = tip vortex rollup location

unless blade has highly tapered tip, use RTVTX = ETIP (default is 1.0) to avoid problems

NTRAIL = number of panels for far wake trailed vorticity

specify panel boundaries TEDGE

defines structure and rollup of inboard vorticity in wake model

USMODL, OPUS = unsteady aerodynamic loads (attached flow)

DSMODL, OPDS = dynamic stall model

EXTERNAL AEROACOUSTIC ANALYSIS

OPEXT = prescribed coefficient increment from table

coefficient Δc_ℓ, Δc_d, Δc_n, Δc_x, Δc_m, Δc_r, Δc_g
$\Delta c_{\ell f}$, Δc_{df}, Δc_{nf}, Δc_{xf}, Δc_{mf}, Δc_{rf}

or $\Delta(M^2 c)$ form

or load ΔF_{QC}, ΔF_{3QC}, ΔM_{QC}, ΔF_f, ΔM_f

COMPUTATIONAL FLUID DYNAMICS ANALYSIS

AERODYNAMIC SENSORS

OUTPUT SELECTED SEPARATELY FOR TRIM, TRANSIENT, AND FLUTTER TASKS

ELEMENTARY SCENARIO

using initial values

uniform properties, rectangular planform
static stall

minimum input: CHORD (only first value)

8–16 Class = ROTOR, Type = INFLOW

ROTOR INFLOW

KINTFx = forward flight interference factor
KINTHx = hover interference factor

INFLOW PARAMETERS

KHLMDA and KFLMDA = momentum theory correction factors

OPFFLI = linear inflow variation in forward flight
FMLMDA = linear inflow variation from hub moments

OPHZT = wake model for hover near zero thrust

OPDMT = momentum theory model (global or differential)

OPDUCT = ducted fan model

OPDW = dynamic wake model

ELEMENTARY SCENARIO

using initial values

minimum input:

KHLMDA (default = 1.1)
KFLMDA (default = 1.8)

8–17 Class = ROTOR, Type = WAKE

ROTOR WAKE

NONUNIFORM INFLOW CALCULATION OFF ROTORS

at locations defined in airframe axes

NONUNIFORM INFLOW PARAMETERS

OPFW = far wake rollup
OPNW = lifting line theory order

generally use single-peak model, unless observe dual-peak character of calculated loading

second-order lifting-line theory most accurate

increase vortex core radius by 15% chord if use first-order (quarter chord collocation point)

far wake extent should be at least two revolutions, more at low speed

OPRTV = tip vortex initial span station

prescribed RTVTX (separate specification for wake geometry and loading) or Betz rollup

EXTERNAL AEROACOUSTIC ANALYSIS

OPEXT = exclude vortex elements inside computational domain

to calculate partial angle-of-attack in post-trim solution; must use fixed circulation strength

WAKE GEOMETRY

OPRWG = rigid or prescribed wake geometry model

prescribed geometry calculated using input TWIST value (specified separate from aerodynamic twist input)

OPFWG = free wake geometry model

tip vortex core size COREWG usually same as value CORE(1) used to calculate induced velocity

extent of calculated distorted geometry MFWG can be less than far wake extent RFW or RDW (calculated distortion extrapolated as necessary)

SUMMARY OF WAKE MODELS, OPTIONS, AND SCENARIOS

Rotor Wake Scenarios
Rolled-Up Wake Model
Multiple-Trailer Wake Model
Multiple-Trailer Wake Model, With Consolidation
Tip Vortex Formation
Vortex Core Radius
Display Geometry

empirical models introduce many input parameters, most of which are used with default values or ignored

SCENARIOS

using initial values

trim input:

LEVEL specifies rigid/prescribed or free wake geometry
default azimuth step = 15 deg

ROTOR IN FORWARD FLIGHT

single peak model
two revolutions of far wake
second-order lifting-line theory, swept lifting line
near wake extent = 60 deg
tip vortex core radius = 20% chord
core type: Scully model, constant core radius
wake vorticity elements: line segment with linear strength
suppress inboard blade-vortex interaction
rigid wake geometry model (not prescribed)
free wake geometry model (Johnson method)

minimum input:

OPFW = far wake rollup model (for dual peak)
RFW = far wake extent

ROTOR IN HOVER

single peak model
five revolutions of far wake
three-quarter chord collocation point, swept lifting line
near wake extent = 60 deg
tip vortex core radius = 20% chord
core type: Scully model, constant core radius
wake vorticity elements: line segment with linear strength
no shed wake vorticity
prescribed wake geometry model (Kocurek and Tangler)
free wake geometry model (Johnson method)

minimum input:
TWIST = blade twist
RICWG = initial convection

ROTOR IN LOW SPEED FLIGHT

single peak model
four revolutions of far wake
second-order lifting-line theory, swept lifting line
near wake extent = 60 deg
tip vortex core radius = 20% chord
core type: Scully model, constant core radius
wake vorticity elements: line segment with linear strength
suppress inboard blade-vortex interaction
rigid wake geometry model (not prescribed)
free wake geometry model (general method)

minimum input:
OPFW = far wake rollup model (for dual peak)
RFW = far wake extent

8–18 Class = TABLES

NAME, FILE UNIT NUMBER, AND FILE NAME OR LOGICAL NAME FOR FOR ALL TABLES REQUIRED BY ANALYSIS

ROTOR AERODYNAMICS

AIRFOIL TABLE

AFTABL = same table for all rotors, or separate tables

default file name or logical name: BLADEAIRFOILn

AIRFRAME STRUCTURE

MODE PROPERTIES INPUT

TAIL BOOM COEFFICIENT INPUT

AIRFRAME AERODYNAMICS

NONLINEAR AERODYNAMIC MODEL INPUT

NLTABL = 2D or 3D or 4D table

STABILITY DERIVATIVE AERODYNAMIC MODEL INPUT

INTERFERENCE VELOCITY

Chapter 9

INPUT AND COMPONENTS INPUT

GENERAL REFERENCE FOR CHAPTER: CAMRAD II Documentation, Volume IV, Input; and Volume V, Components Input

FOLLOWING SECTIONS ARE EXAMPLES OF USING CORE INPUT TO MODIFY ROTORCRAFT CONSTRUCTED BY SHELL

input parameters of core system pieces and components

SEE ALSO DOCUMENTS DEMONSTRATING USE OF CAMRAD II CORE INPUT:

TILTROTOR ANALYSIS
construction of elastic cantilever wing, replacing normal modes representation of rotor support

BEARINGLESS ROTOR ANALYSIS
construction of snubber-damper with axes following blade pitch, replacing snubber with axes fixed to hub

CLOSED-LOOP HIGHER-HARMONIC CONTROL ANALYSIS
completion of regulator loop, including construction of filters to obtain harmonic loads for feedback

PENDULUM ABSORBER ANALYSIS
construction of flapwise absorber

SEVERAL EXAMPLES OF CORE INPUT ARE INCLUDED IN CAMRAD II EXTRAS FOLDER

frozen wake geometry for second case or job (solution procedure change)

individual blade control

interblade damper (extensive core input)

elastic beams for trailing-edge flap structure

tip-path-plane sensors for hingeless and bearingless rotors

autopilot for transient task

prescribed airframe motion for transient task

revised trim variables or input trim matrix

9–1 Lag Damper

ROTORCRAFT SHELL DEFINES LINEAR SPRING AND DAMPING VALUE FOR EACH JOINT CONSTRUCTED

CORE INPUT NEEDED FOR NONLINEAR SPRING OR DAMPING

example: hydraulic lag damper

IDENTIFY NAME OF COMPONENT WITH LAG DAMPER

ELEMENTS CONSTRUCTED ARE LISTED AT BEGINNING OF JOB

name: `'ROTOR n BLADE m ELEMENT k'`

IDENTIFY SPRING/DAMPER (j) FOR LAG JOINT

PRINT CORE INPUT FOR COMPONENT

using class = CASE parameters:

TMTASK = 2
NPRNTC = 1
PCLASS = `'COMPONENT'`
PTYPE = `'RIGID BODY'` or `'BEAM'`
PNAME = name

OBTAIN DAMPER CHARACTERISTICS

HYDRAULIC DAMPER MOMENT M, AS FUNCTION OF LAG RATE $\dot{\zeta}$:

$$M = \min(M_{\max}, M_{\max}(\dot{\zeta}/\dot{\zeta}_{\text{ref}})^2) = \min(x, y\dot{\zeta}^2)$$

PREPARE CORE INPUT

IDENTIFY INPUT BY CLASS, TYPE, AND NAME

usually for all blades

LINEAR + HYDRAULIC DAMPER: CTYPE = 3

NO LINEAR DAMPING: CLIN = 0.

keep CEQUIV, for flutter and solution procedures

HYDRAULIC DAMPING: polynomial order and coefficients

figure 1 illustrates input required

```
....
....
....
&NLDEF action='end of shell',&END
! hydraulic lag damper (core input)
!=========================================================================
&NLDEF class='COMPONENT',type='BEAM',name='ROTOR n BLADE m ELEMENT k',&END
&NLVAL
   CTYPE(j)=3,CLIN(j)=0.,
   NCHYDA(j)=0,CHYDA(1,j)=x,
   NCHYDB(j)=2,CHYDB(1,j)=0.,0.,y,
&END
!=========================================================================
&NLDEF action='end of core',&END
```

Figure 9-1 Lag damper.

9–2 Additional Output

ROTORCRAFT SHELL CONSTRUCTS OUTPUT FOR VARIOUS STRUCTURAL AND AERODYNAMIC SENSORS

CORE INPUT NEEDED FOR OUTPUT OF SYSTEM VARIABLES

example: output for input/output interface variable

IDENTIFY NAME OF INPUT/OUTPUT INTERFACE

USE INPUT PROGRAM TO EXAMINE INTERFACES CONSTRUCTED BY SHELL

these variables are already being solved, so only need to add output to system

can also examine trim loops, to determine where to put write module

name: `'interfacename'`

EXAMPLES OF NAMES

```
'ROTOR n PITCH CONTROL BLADE m'
'AIRFRAME AERODYNAMIC FORCE WB'
'ROTOR n FORCE (RTR AXES)'
'ROTOR n WAKE   V, PNL kkk W mm'
```

see CAMRAD II Documentation, Volume III, Rotorcraft Theory; Chapter “Rotorcraft System”

PREPARE CORE INPUT

IDENTIFY INPUT BY CLASS, TYPE, AND NAME

CREATE OUTPUT PIECE FOR INTERFACE VARIABLE

specify number of time steps and number of harmonics desired

name of output piece must be unique

ADD WRITE MODULE TO WAKE LOOP

here “20” is intended to be larger than current number of write modules in loop

if add write module to some other loop, should consider where the analysis solves for the variable

figure 2 illustrates input required

```
....
....
....
&NLDEF action='end of shell',&END
!======================================================================
&NLDEF class='OUTPUT',name='NEWOUTPUT',&END
&NLVAL KINDY=2,YNAMEV='interfacename',MTIME=n,MHARM=m,&END
&NLDEF class='TRIM LOOP',type='SUCCESSIVE SUBSTITUTION',name='WAKE',&END
&NLVAL NWRT=20,KINDW(20)=7,WRT(20)='NEWOUTPUT',&END
!======================================================================
&NLDEF action='end of core',&END
```

Figure 9-2 Additional output.

9–3 General Approach

ROTORCRAFT SHELL CONSTRUCTS TYPICAL SYSTEM

CORE INPUT NEEDED TO MODIFY MODEL

often just modify structural model,
perhaps just change geometry

ASSISTANCE AVAILABLE FROM MAINTENANCE/SUPPORT CONTRACTOR

STEPS IN DEVELOPING CORE INPUT TO MODIFY ROTORCRAFT SHELL MODEL:

1) PREPARE SHELL MODEL

SHELL INPUT AND JOBS

get as close as possible to desired system

usually shell creates all of aerodynamic model

2) EXAMINE SYSTEM CONSTRUCTED BY SHELL

CAN USE INPUT PROGRAM TO IDENTIFY SYSTEM PIECES

or analysis with TMTASK = 2, NPRNTD = 1

PRINT CORE INPUT FOR SPECIFIC SYSTEM PIECES OR ENTIRE MODEL

using class = CASE parameters:

TMTASK=2 (perhaps TNTASK=2,FLTASK=2) and
NPRNTC=n,PCLASS= . . . ,PTYPE= . . . ,PNAME= . . .

to print core input for entire system, use

```
NPRNTC=1,PCLASS='    '
```

USE CORE INPUT CREATED BY SHELL AS MODEL AND GUIDE, AND AS INPUT MANUAL

follow pattern for joints, degrees of freedom, interfaces, sensors

IDENTIFY (BY NAME) PIECES TO BE MODIFIED

TYPICALLY INTERESTED IN STRUCTURE

structural dynamic components
structural dynamic interfaces
modes
response for degrees of freedom
response for constraint forces

rigid degrees of freedom require rigid response piece

other quantities (elastic degrees of freedom, joint degrees of freedom, constraint forces) require variable response pieces

MAY NOT NEED TO CHANGE SOLUTION PROCEDURE

ASSIGN NEW COMPONENTS AND INTERFACES TO EXISTING PARTS AND MODES

following pattern constructed by shell

3) SKETCH SYSTEM CONSTRUCTED BY SHELL

GEOMETRY AND LOGICAL RELATION, FOR COMPONENTS AND INTERFACES

see CAMRAD II Documentation, Volume III, Rotorcraft Theory; Volume IV, Input; Volume V, Components Input

STRUCTURAL MODEL REQUIRES GEOMETRY

summary of geometry in rotorcraft shell: Table 7.6 of Volume III

identify body axes of all elements
airframe: z-axis down, x-axis forward
hub: z-axis in $+T$ direction, x-axis downstream
beams: x-axis axial

and axes of structural dynamic interfaces
locations, joints, connections

4) DEFINE MODIFIED SYSTEM

GEOMETRY AND LOGICAL RELATION, FOR COMPONENTS AND INTERFACES

sketch geometry of each body and beam in isolation, showing body axes, location axes, joint axes, connection axes (ignore frames) connection axes define position and orientation of interfaces

typically interface axes are rotated from body axes back to frame; then rotation matrix of location or connection is transpose of rotation matrix of nominal response

convenient approach, but in general not necessary to involve frame in description of component geometry

conventions for describing rotations:
Euler angles, $C = XYZ$ (most useful)
arbitrary rotations, $C = UVW$
Unit, $C = I$; or input matrix

system assembled by gluing connection axes together

create sensors (with output) for all new joints and degrees of freedom

perhaps create sensors for loads as well

use blade position sensor solution procedures (MPSEN = 1)

identify frames for description of motion (new frames probably not needed)

frames and nominal (rest position or rotating) are required to describe motion of structural dynamic components

motion of body axes is measured relative frame of component

nominal position and orientation defined in rigid response piece

sketch each input/output interface, and connection to components

5) PREPARE CORE INPUT

DELETE EXISTING SYSTEM PIECES AS REQUIRED

IDENTIFY INPUT FOR NEW OR MODIFIED SYSTEM PIECES BY CLASS, TYPE, AND NAME

set up all NLDEF namelists

define parameters in NLVAL namelists (guided by input manuals, and values constructed by shell)

SET UP INPUT FOR ONE BLADE, THEN DUPLICATE FOR OTHER BLADES

reference (last) and other blades are different

reference blade has sensors and aerodynamics interfaces, other blades defined as child response

follow pattern created by shell

RUN ANALYSIS TO FIND ERRORS

6) CHECK MODIFICATIONS

USE INPUT PROGRAM TO DRAW GEOMETRY OF SYSTEM WITH NEW COMPONENTS AND INTERFACES

draws rest position, as specified by nominal in response pieces

rest position should be close to where component will be when the system is assembled

can draw axes at structural dynamic interfaces (change drawing option to get orientation, in separate file)

CHECK STRUCTURAL DYNAMICS AND GEOMETRY (WITHOUT AERODYNAMICS)

trim solution (NRPRNT = 7)

if geometry of connections and rest positions is correct, trim solution for rigid motion of structural dynamic components should be small

units of rigid motion: ft or m for linear, radians for angular

trim position sensors (MPSEN), and sensors for new joints and new degrees of freedom

free vibration calculation of frequencies (flutter task)

solution for new input/output interfaces

7) RUN TEST CASES

CORE INPUT CONSTRUCTED BY SHELL CAN BE CHANGED BY SHELL PARAMETERS

- IN EXISTING STRUCTURAL DYNAMIC COMPONENTS, NUMBER OF LOCATIONS, JOINTS, CONNECTIONS, INTERFACES, AND SENSORS CAN BE CHANGED
  - for example, different models are constructed with and without aerodynamics, or modes, or sensors
- CARE REQUIRED SO CORE CHANGES REMAIN CONSISTENT WITH MODEL CREATED BY SHELL
- LOCATIONS, JOINTS, CONNECTIONS, INTERFACES, SENSORS ON A STRUCTURAL DYNAMIC COMPONENT CAN BE IGNORED
  - default is to ignore location, joint, connection, interface, or sensor
  - so core input can add items without knowing exactly the current number
    - add locations, joints, connections, interfaces, sensors beyond the maximum possible created by the shell

GEOMETRY DEFINITION FOR STRUCTURAL DYNAMIC COMPONENTS

CAN CREATE NEW COMPONENTS AND INTERFACES USING CORE INPUT

MAY ENCOUNTER ERROR MESSAGE:

matrix singular
part differential equations
inverting PHIX0B

OR FIND VERY LARGE ANGLES (1 or 2 radians) IN TRIM SOLUTION (printed using NRPRNT = 7)

MAY BE CAUSED BY 180 DEGREE MISMATCH OF CONNECTION AXES (REST POSITION) ON TWO SIDES OF STRUCTURAL DYNAMIC INTERFACE

produces singular constraint equations

CAN USE INPUT PROGRAM TO DRAW AXES AT STRUCTURAL DYNAMIC INTERFACES

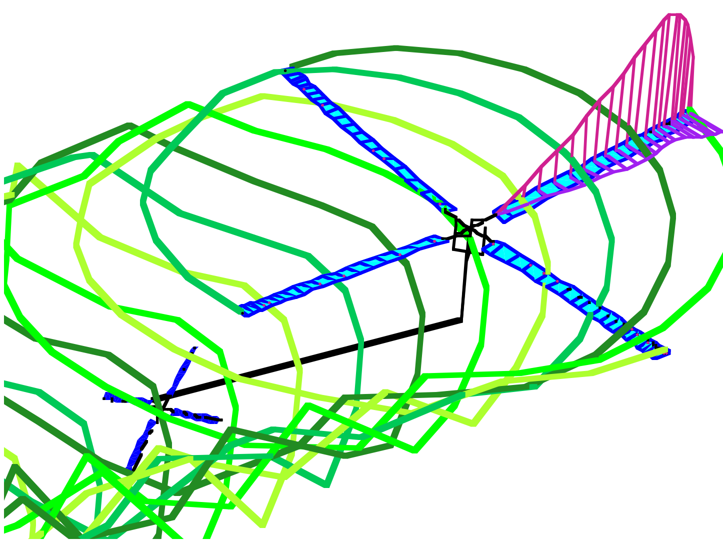
Chapter 10

CONCLUSION

IT IS DIFFICULT TO ANALYZE ROTORCRAFT

- COMPLEX, MULTIDISCIPLINARY SYSTEM
- STILL A LOT WE DO NOT KNOW ABOUT AERODYNAMICS, DYNAMICS, AND STRUCTURES OF ROTORCRAFT
- STILL NEED ENGINEERING JUDGEMENT, EXPERIENCE, AND MUCH TESTING OF THE ACTUAL SYSTEM

IT IS IMPOSSIBLE TO ANALYZE ROTORCRAFT EFFECTIVELY WITHOUT THE PROPER TOOLS



CAMRAD II IS A SOPHISTICATED AND MATURE AEROMECHANICAL ANALYSIS OF HELICOPTERS AND ROTORCRAFT