CAMRAD II

COMPREHENSIVE ANALYTICAL MODEL OF ROTORCRAFT AERODYNAMICS AND DYNAMICS

Demonstration of Core Input

Wayne Johnson
Johnson Aeronautics
Palo Alto, California

Distributed by
Analytical Methods, Inc.
Redmond, Washington
This document summarizes the tasks involved in analyzing a tiltrotor using CAMRAD II. In particular, the construction of an elastic cantilever wing using core components is demonstrated, replacing the normal modes representation of the rotor support that is used by the rotorcraft shell.

1) Baseline Input

A model constructed by the rotorcraft shell is required as a starting point. The sample case analyzing a tiltrotor on a cantilever wing in a wind tunnel was used here: namelist input files TILTROTOR1.LIST and TILTROTORW.LIST; and job SAMPLE10. This case has a single rotor at shaft angle \( \text{ASHAFT} = -90 \) deg, on a constrained airframe with three elastic modes. The rotor radius is 15 ft. The rotor model was used without change. The file TILTROTORW.LIST was modified to replace the normal modes representation of the airframe by a cantilever wing constructed of beam and rigid body components.

With normal modes, the actual position of the rotor hub is not required. Now the wing root is at the origin of the airframe axes; and the rotor hub is at the end of the pylon. A wing semispan of 18 ft and mast length of 6 ft are used. Hence the baseline case was modified by setting \( \text{FSRTR} = -6 \) and \( \text{BLRTR} = 18 \) (AIRFRAME STRUCTURE input). Airframe aerodynamics were suppressed as well. The structural dynamic and aerodynamic components were drawn in order to check the baseline geometry; the INPUT program extracts the geometry required.

A CAMRAD II job was needed to check the work. Starting with SAMPLE10, trim sensors of the blade position were turned on (parameters MPSEN=1, OPPOS=4, NRPOS=2). These sensors include the position of the hub relative the airframe frame. The job was simplified for faster execution during this development work, by using only one blade mode in the flutter analysis; omitting the drive train model; using only two elastic beam elements per blade; and using only 10 aerodynamic panels per blade. This job was run for the baseline input.
2) Existing Pieces

Next the system pieces of the baseline model must be examined. For this purpose the baseline job was run with only initialization (TMTASK=2 and FLTASK=2), and without rotor aerodynamics. With rotor aerodynamics, the model would have many system pieces that are not of interest here. The job was run with NPRNTD=1 to list all the system pieces (the INPUT program could have been used instead). The following existing pieces of interest were identified:

<table>
<thead>
<tr>
<th>components:</th>
<th>AIRFRAME</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AIRFRAME ROTOR 1</td>
</tr>
<tr>
<td></td>
<td>AIRFRAME SWASHPLATE 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SD interfaces:</th>
<th>AIRFRAME/AIRFRAME ROTOR 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AIRFRAME/AIRFRAME SWASHPLATE 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>modes:</th>
<th>Rotor 1 Blade 1 Flutter</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AIRFRAME Flutter</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>rigid response:</th>
<th>FRM AIRFRAME</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DOF AIRFRAME ROTOR 1 RIGID BODY</td>
</tr>
<tr>
<td></td>
<td>DIRFRAME SWASHPLATE 1 RIGID BODY</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>variable response:</th>
<th>DOF AIRFRAME ELASTIC MODES</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>INT AIRFRAME/AIRFRAME ROTOR 1</td>
</tr>
<tr>
<td></td>
<td>I AIRFRAME/AIRFRAME SWASHPLATE 1</td>
</tr>
</tbody>
</table>

These pieces are involved in the airframe model; they are described in Volume III (Rotorcraft Shell) of the CAMRAD II documentation. Next the job was run with NPRNTC=n and PCLASS, PTYPE, PNAME as required to print the input for these system pieces. These system pieces must be studied to understand the baseline model, and as examples for construction of the new model.

Figure 1 shows the geometry of the model constructed by the rotorcraft shell. This information was obtained from the input of the system pieces, and from Volume III (Rotorcraft Shell).

3) Geometry of Modified Model

Now the geometry of the modified model must be defined. Figure 2 shows the geometry. Information from Volume II (Components Theory) and Volume V (Components Input) was used, particularly for the geometry of the beam components.
The wing and pylon are constructed of one elastic beam element each. The wing root is attached to the "AIRFRAME" component, which will be revised to be just a massless point at the origin of the airframe frame. The "AIRFRAME" component now just has one location, one connection, and one structural dynamic interface, all at the frame origin. The "AIRFRAME" component is retained since it defines the airframe frame. At the wing tip, there is a two-degree of freedom joint: rotation about the beam axis (pitch) and then rotation about the vertical axis (yaw). The pitch motion has a large spring rate; its offset allows the pylon angle to be set to an arbitrary value. The yaw motion has a moderate spring rate.

The pylon is represented as a beam extending from the pivot to the hub; and a rigid body at the pivot location. The beam is very stiff. Interfaces between the "AIRFRAME" component and the rotor hub and swashplate are deleted. New interfaces are created between the beam representing the pylon, and the rotor hub and swashplate. All interfaces and degrees of freedom require response definitions.

A modal analysis of the airframe system is needed, to allow an assessment of the model and to minimize the number of degrees of freedom used in the flutter analysis. A modal analysis for trim might be needed as well. The rotorcraft shell was used to create the airframe mode set, to which the new wing components and interfaces are assigned. Modal sensors are defined at the hub and pylon positions. Note that the massless rigid component "AIRFRAME SWASHPLATE 1", between the airframe and swashplate, is created by the shell only when airframe modes are used. (This component is required because the partition of the flutter equations into parts does not allow parts to be coupled by modal degrees of freedom.)

4) Modified Model

Next the core input required to construct this model is added to the namelist file. The input is defined using information from the listing of the pieces constructed by the shell; from Volume III (Rotorcraft Shell; weights for convergence and perturbation); and from Volume IV (Input) and Volume V (Components Input). The following core pieces are defined:

revised:

components: AIRFRAME
AIRFRAME ROTOR 1

deleted:

SD interfaces: AIRFRAME/AIRFRAME ROTOR 1
AIRFRAME/AIRFRAME SWASHPLATE 1

output: AIRFRAME MODE ROTOR 1 HUB

variable response: INT AIRFRAME/AIRFRAME ROTOR 1
1 AIRFRAME/AIRFRAME SWASHPLATE 1
OUT AIRFRAME MODE ROTOR 1 HUB
new:

components: AIRFRAME BEAM 1
AIRFRAME BEAM 2
AIRFRAME PYLON

SD interfaces: AIRFRAME / AIRFRAME BEAM 1
AIRFRAME BEAM 1 / BEAM 2
AIRFRAME BEAM 2 / AF ROTOR 1
AIRFRAME BEAM 2 / PYLON
AIRFRAME BEAM 2 / SWASHPLATE

rigid response: DOF AIRFRAME BEAM 1 RIGID BODY
DOF AIRFRAME BEAM 2 RIGID BODY
DOF AIRFRAME PYLON RIGID BODY

variable response: DOF AIRFRAME BEAM 1 ELASTIC
DOF AIRFRAME BEAM 1 JOINT
DOF AIRFRAME BEAM 2 ELASTIC
INT AIRFRAME / AIRFRAME BEAM 1
INT AIRFRAME BEAM 1 / BEAM 2
INT AIRFRAME BEAM 2 / AF ROTOR 1
INT AIRFRAME BEAM 2 / PYLON
INT AIRFRAME BEAM 2 / SWASHPLATE

output: AIRFRAME MODE SENSOR
variable response: OUT AIRFRAME MODE SENSOR

Note that the rotorcraft shell input is also modified, as follows. The parameter
MASSR is set to zero, since it is part of the airframe normal modes definition. The
number of airframe modes in the shell model is set to zero as well. All of the modal
information in the AIRFRAME STRUCTURE input could be deleted as well, since it is
no longer being used. Then the core input is defined. The job that creates the
unformatted input file requires NLISTC = 2 in order for this core input to be used.

5) Check of Modified Model

The input was checked by running the CAMRAD II job with just initialization. The
rest positions of the components were checked by drawing the system; the INPUT
program provides the required geometry data. Figure 3 shows the final result.

Finally, the CAMRAD II trim and flutter analysis were run to check the model. The
following can be examined in the output. Under "ROTOR 1 PERFORMANCE," the
advance ratio and shaft angle can be checked. Under "BLADE POSITION SENSOR," the
sensor at radial station 0.0R gives the hub position relative the origin of the
airframe frame (at the wing root). The drawing shows the rest positions; this
information from the job output shows the equilibrium solution. The results
verified that the equilibrium solution gave the expected position and orientation of the rotor hub.

The equilibrium solution was checked for pylon angles of 0, 10, 45, and 90 degrees. The pylon angle was set using the pitch joint offset OFFSET(2); and also using the frame pitch angle PITCH (in TRIM input).

Details of the airframe structural model were checked by examining the modal solution, found under "MODE SET = AIRFRAME FLUTTER" in the output.

The modal solution was examined to check the structural dynamic properties. In the present case, the stiffnesses were adjusted in order to obtain reasonable modes. More typically, the results would be compared with expected results, or with the solution from a finite-element analysis. In order to assess the mode shapes, output sensors were placed at the hub and pivot.

6) Analysis Input

Copies of the input files are included with the CAMRAD II sample cases. The namelist files are:

    YTRINPUT.COM, YTRINPUT1.LIST, YTRINPUTW.LIST

and

    YTILTROTOR.COM

is the job file.
AF/AR1 interface in S
AF/R1SP interface in S
AIRFRAME \( C^{SR} = C^{SF} \)

\[
CSF = \begin{bmatrix}
0 & 0 & -1 \\
0 & 1 & 0 \\
1 & 0 & 0
\end{bmatrix}
\]

AIRFRAME Rotor 1
nominal = \( C^{SF} \) and \( \begin{pmatrix} 6 \\ 18 \\ 0 \end{pmatrix} \)
\[ M = N M_b - M_k \]

AIRFRAME SWASHPLATE 1
nominal = \( C^{SF} \) and \( \begin{pmatrix} 4.5 \\ 18 \\ 0 \end{pmatrix} \)

AIRFRAME
linear normal modes
(constrained)

operating condition
yaw, pitch, roll
between F and I frames
Figure 2

\( C^{CJ} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \)

\( C^{BF} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \)

B2/AFR1 interface in S
B2/SP interface in S
BEAM 2 \( C^{CJ} = C^{SF} C^{FB} \)

AIRFRAME BEAM 2
nominal = \( C^{BF} \) and \( \begin{pmatrix} 0 \\ 18 \\ 0 \end{pmatrix} \)

AIRFRAME PYLON
nominal = \( C^{BF} \) and \( \begin{pmatrix} 0 \\ 18 \\ 0 \end{pmatrix} \)

joint dof
pitch then yaw

B2/P interface in B2

B1/B2 interface in F
BEAM 1 \( C^{CJ} = C^{FB} \)
BEAM 2 \( C^{CJ} = C^{FB} \)

AIRFRAME BEAM 1
nominal = \( C^{BF} \) and \( \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \)

\( C^{BF} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \)

AF/B1 interface in F
BEAM 1 \( C^{CJ} = C^{FB} \)

AIRFRAME (constrained)

F frame
This document summarizes the tasks involved in analyzing a bearingless rotor using CAMRAD II. In particular, the use of core components to construct a snubber-damper with axes following the blade pitch is demonstrated, replacing the snubber with axes fixed to the hub that is used by the rotorcraft shell.

1) Baseline Input

A model constructed by the rotorcraft shell is required as a starting point. The sample case analyzing a bearingless rotor in a wind tunnel was used here: namelist input file BEARINGLESS.LIST; and job SAMPLE8. This case has a single five-bladed rotor. The rotor radius is 20 ft. Some changes were made to the namelist input file BEARINGLESS.LIST. Then this file was modified to replace the snubber with one that has axes following the blade pitch.

A CAMRAD II job was needed to check the work. Starting with SAMPLE8, trim sensors of the blade position were turned on (parameters MPSEN=1). This job was run for the baseline input. This job has hover trim and stability of a single blade, so it is not very long. The job could be simplified, to be faster for the development process. For example, fewer aerodynamic panels could be used, or even no rotor aerodynamics at all.

2) Existing Pieces

Next the system pieces of the baseline model must be examined. For this purpose the baseline job was run with only initialization (TMTASK=2 and FLTASK=2), and without rotor aerodynamics. With rotor aerodynamics, the model would have many system pieces that are not of interest here. The job was run with NPRNTD=1 to list all the system pieces (the INPUT program could have been used instead). The following existing pieces of interest were identified:
components:  
ROTOR 1 BLADE 1 ROOT  
ROTOR 1 BLADE 1 ELEMENT 1  
ROTOR 1 FLEXBEAM 1 ELEMENT 1  

SD interfaces:  
ROTOR 1 BLADE 1 ROOT/BLADE  
ROTOR 1 BLADE 1 ROOT/FLEXBEAM  

output:  
ROTOR 1 BLADE 1 POS SJOINT  
ROTOR 1 BLADE 1 MODE SJOINT  

variable response:  
DOF ROTOR 1 BLADE 1 SNUBBER  
OUT ROTOR 1 BLADE 1 POS SJOINT  
OUT ROTOR 1 BLADE 1 MODE SJOINT  
INT ROTOR 1 BLADE 1 ROOT/BLADE  
INTROTOR 1 BLADE 1 ROOT/FLEXBEAM  

These pieces are involved in the hub and snubber model; they are described in Volume III (Rotorcraft Shell) of the CAMRAD II documentation. Next the job was run with NPRNTC=12 and PCLASS, PTYPE, PNAME as required to print the input for these system pieces. These system pieces must be studied to understand the baseline model, and as examples for construction of the new model.

The attached figure shows the geometry of the model constructed by the rotorcraft shell. This information was obtained from the input of the system pieces, and from Volume III (Rotorcraft Shell).

Note that the core input constructed by the shell can be changed by shell parameters. In particular, changing the number of sensors or output selected, the aerodynamic panels, or the number of nodes could change the locations, connections, interfaces, and sensors constructed for these components.

3) Geometry of Modified Model

The bearingless rotor constructed by the rotorcraft shell models the snubber as a 3-variable linear joint on the "ROTOR 1 BLADE 1 ROOT" component. Hence the axes of this snubber are fixed relative the hub. There is a pinned interface (connecting displacement but not rotation) between this snubber joint and the root of the blade (or torque tube). The location of the snubber joint on the blade root component is offset from the origin of the blade frame, as defined by the input parameters XSNUB, YSNUB, ZSNUB (in ROTOR FLEXBEAM input). The interface axes on the blade root are rotated from the beam axes to the hub axes ($C_{CJ} = C_{FB}$, which includes the 2 deg of precone), but this actually has no effect with the pinned interface.

The modified model has the snubber axes following the blade, hence changing with the blade pitch. The snubber axes are tilted forward by angle $\theta_s$ relative the hub vertical (at zero control input). The attached figure shows the two modifications.
considered. Information from Volume V (Components Input) was used, particularly for the geometry of the beam components.

Modification A assumes that the structure is rigid between the hub and the snubber attachment point. Hence it is not actually necessary to connect the blade root to the flexbeam; and the interface can remain between the blade and the root component. The modifications include the following:

a) The snubber joint is still on the "ROTOR 1 BLADE 1 ROOT" component, but is changed to a 3-variable angular motion (with no spring or damping) followed by a 3-variable linear motion.

b) The interface is changed from pinned to cantilever. Hence the snubber axes will follow the blade pitch motion.

c) The snubber axes are oriented by defining the connection axes of the interface on the "ROTOR 1 BLADE 1 ELEMENT 1" beam component:

\[ C^{CJ} = Z_{-90} \times X_{-\theta_s} \]

where \( Z \) and \( X \) are matrices implementing rotations about the z and x axes respectively (see the introduction chapter in the CAMRAD II documentation).

All modification must be made for each of the five blades of the rotor. The response of blades 1 to 4 are children of blade 5, for the degrees of freedom and interface.

Modification B assumes that the structure is elastic between the hub and the snubber attachment point. Hence it is necessary to connect the blade root to the flexbeam. The modifications include the following.

a) The snubber and interface on the "ROTOR 1 BLADE 1 ROOT" component are deleted.

b) The snubber interface is added to the appropriate flexbeam element ("ROTOR 1 FLEXBEAM 1 ELEMENT 1" here): a new location is defined at \( x = E\text{ROOT} \times \text{RADIUS} / \text{LENGTH} \) on the beam axis; a new connection at this location; and a new structural dynamic interface at this connection.

c) The snubber joint is added to the first blade element, component "ROTOR 1 BLADE 1 ELEMENT 1". The joint is placed at the existing location #3; the location axes are the component body axes, which are preconed, with the y-axis in the hub plane. The snubber axes are oriented by defining the joint axes at this location:

\[ C^{JE} = X_{-\theta_s} \]
where $X$ is the matrix implementing rotation about the $x$ axis (see the introduction chapter in the CAMRAD II documentation). The joint has 3-variable linear motion. The existing connection #3 is revised to be at this joint. The appropriate springs and sensors are defined. The original snubber joint defined by the rotorcraft shell in the "ROTOR 1 BLADE 1 ROOT" component is used as a guide for these modifications.

d) The snubber interface, sensors, and response system pieces are modified to identify the appropriate components.

Note that the degree of freedom, interface, sensor, and output names are not changed. Hence it is only necessary to modify existing system pieces, rather than delete old and create new pieces. All modification must be made for each of the five blades of the rotor. The response of blades 1 to 4 are children of blade 5, for the degrees of freedom and interface.

4) Modified Model

Next the core input required to construct this model is added to the namelist file. The input is defined using information from the listing of the pieces constructed by the shell; and from Volume IV (Input) and Volume V (Components Input). The following core pieces are changed for modification A:

components:       ROTOR 1 BLADE 1 ROOT
                  ROTOR 1 BLADE 1 ELEMENT 1
SD interfaces:    ROTOR 1 BLADE 1 ROOT/BLADE
variable response: INT ROTOR 1 BLADE 1 ROOT/BLADE

The following core pieces are changed for modification B:

components:       ROTOR 1 BLADE 1 ROOT
                  ROTOR 1 BLADE 1 ELEMENT 1
                  ROTOR 1 FLEXBEAM 1 ELEMENT 1
SD interfaces:    ROTOR 1 BLADE 1 ROOT/BLADE
variable response: DOF ROTOR 1 BLADE 1 SNUBBER
output:           ROTOR 1 BLADE 1 POS  SJOINT
                  ROTOR 1 BLADE 1 MODE  SJOINT

All modification must be made for each of the five blades of the rotor.
5) Check of Modified Model

The rest positions of the components can be checked by drawing the system; the INPUT program provides the required geometry data. The axes of the snubber joint and interfaces can be drawn. Alternatively, sensors could be defined on the appropriate connections, so the trim solution would show the equilibrium position of the snubber axes.

The CAMRAD II trim and flutter analyses were run to check the model, comparing the two modifications with the baseline. Quantities checked included the following: the trim blade $C_T/\sigma$ and pitch; the modal frequencies in the flutter analysis; the lag damping in the flutter analysis; and the equilibrium solution from the trim analysis.

6) Analysis Input

Copies of the input files are included with the CAMRAD II sample cases. The namelist files are:

    YBINPUT.COM, YBINPUT.LIST, YBINPUTMODA.LIST, YBINPUTMODB.LIST

and

    YBEARINGLESS.COM

is the job file.
HUB CONSTRUCTED BY SHELL:
snubber between hub and blade root, snubber axes fixed to hub

MODIFICATION A:
snubber axes fixed to blade, flexbeam rigid from hub to snubber

MODIFICATION B:
snubber axes fixed to blade, flexbeam elastic from hub to snubber
This document summarizes the tasks involved in analyzing a closed-loop higher-harmonic control system using CAMRAD II. In particular, the completion of the regulator loop using core input is demonstrated, including construction of filters to obtain harmonic loads for the feedback system.

1) Control Systems

The following systems are considered.

a) Helicopter flapping controller: minimize rotor tip-path plane tilt relative to the shaft, using steady rotor cyclic control. This system is implemented using the regulator loop.

b) Helicopter vibratory hub load controller: minimize N/rev hub force using N/rev swashplate control. This system is implemented using the regulator loop. Filters are constructed to obtained the harmonics of the hub force.

c) Tiltrotor flapping controller: minimize lateral tip-path plane tilt relative to the shaft, using steady lateral cyclic control. This system is implemented using the trim loop.

The flapping controller is demonstrated using both the trim loop and the regulator loop. In general, a regulator loop is used in order to solve two small problems instead of one large problem; use a cost function to define the controller; handle cases where the targets (such as zero vibration) are not achievable; or accommodate systems with more measurements than controls. However, the regulator loop requires more input to define, and usually increases the computation time.
2) Baseline Input

A model constructed by the rotorcraft shell is required as a starting point. For a rotor in a wind tunnel, the sample namelist input file WINDTUNNEL.LIST and the jobs SAMPLE3 and SAMPLE4 were used as a starting point. The rotor has four blades. A job at an advance ratio of 0.2 was defined, with blade position and hub loads output.

For a tiltrotor model, the sample namelist input file TILTROTOR.LIST was used, with job SAMPLE11 as a starting point.

The regulator loop (and parts that solve the sensors for potential use in the regulator) is created by setting OPSTR=1 in the trim input. The N/rev harmonics of the swashplate control are created by setting MHHCN=1 in the trim rotor input.

3) Existing Pieces

Next the system pieces of the baseline model must be examined. The core input created for specific system pieces can be generated by running the baseline job with NPRNTC=n and PCLASS, PTYPE, PNAME as required (and only initialization, TMTASK=2). The initialized system can be examined by running the input program interactively (reading the shell input file, and then a namelist file that includes OPSTR=1 and MHHCN=1). In particular, the names of the input and interfaces involved in the regulator loop are needed.

input names (vector and element):

<table>
<thead>
<tr>
<th>ROTOR 1 INPUT</th>
<th>COLLECTIVE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LATERAL CYCLIC</td>
</tr>
<tr>
<td></td>
<td>LONGITUDINAL CYCLIC</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROTOR 1 HHC COLLECTIVE</th>
<th>MEAN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>COS 1*NBLADE</td>
</tr>
<tr>
<td></td>
<td>SIN 1*NBLADE</td>
</tr>
</tbody>
</table>

interface names (vector and element):

<table>
<thead>
<tr>
<th>ROTOR 1 NONROTATING HUB FORCE</th>
<th>x,y,z components</th>
</tr>
</thead>
<tbody>
<tr>
<td>solved by part = ROTOR 1 REGULATOR SENSOR</td>
<td></td>
</tr>
<tr>
<td>at end of circulation loop</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROTOR 1 TRIM SENSOR</th>
<th>LONGITUDINAL TIP PLANE TILT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LATERAL TIP PLANE TILT</td>
</tr>
<tr>
<td>solved by part = ROTOR 1 PERFORMANCE</td>
<td></td>
</tr>
<tr>
<td>at end of circulation loop</td>
<td></td>
</tr>
</tbody>
</table>
For the flapping controller the interface "ROTOR 1 TIP PATH PLANE" might have been used instead; however the interface "ROTOR 1 TRIM SENSOR" is used by the trim loop, and contains the flap angles in degrees. The input manuals provide details on the parameters required to define the regulator loop, filter components, and interfaces.

4) Helicopter Flapping Controller

A job was constructed that minimizes rotor tip-path plane tilt relative the shaft, using steady rotor cyclic control. This system is implemented using the regulator loop. The regulator parameters are defined by reference to the values of the trim loop. TRACE=3 and NCPRNT=2 are needed to obtain output regarding the regulator convergence.

5) Helicopter Vibratory Hub Load Controller

A job was constructed that minimizes 4/rev vertical hub force using 4/rev collective pitch. This system is implemented using the regulator loop. Filters are constructed to obtain the harmonics of the hub force.

existing interface:

ROTOR 1 NONROTATING HUB FORCE

solved by part = ROTOR 1 REGULATOR SENSOR
at end of circulation loop;
must be connected to new filter components

to new FILTER components:

ROTOR 1 NONROT HUB FORCE COS4
ROTOR 1 NONROT HUB FORCE SIN4

The new part that solves for the harmonics is placed in the circulation loop, after the existing parts; the existing parts that solve for the means are used as a guide. In general, the interface variables require response system pieces (the response defined by the rotorcraft shell for the hub force could be used as a guide). Here the default
response is used, which gives a weight of 1.0 for convergence tests. Then the regulator tolerance has units of lb.

Convergence of the regulator loop is helped by a small relaxation factor. However, the tolerance on the circulation loop is most important for good performance of the regulator loop. With the default value of TOLERC, the hub force calculation is erratic, and the regulator loop does not converge well (and the job takes excessive computation time) or diverges. With TOLERC=0.2 and TOLERR=0.5, the hub force calculation is consistent and the regulator behaves well.

6) Tiltrotor Flapping Controller

A job was constructed that minimizes tiltrotor lateral tip-path plane tilt relative the shaft, using steady lateral cyclic control. This system is implemented using the trim loop. Lateral flapping and lateral cyclic control of the two rotors are added to the trim loop. Core input must be used since the rotorcraft shell does not implement these particular measurements and controls in the trim loop.

6) Trimming Rotor Hub Moments

Jobs were also constructed that illustrate the modifications required to trim to a specified hub moment. The interface vector "ROTOR 1 MEAN FORCE (RTR AXES)" contains the aerodynamic pitch and roll moments in the rotor frame. These are the quantities in the rotor performance output. The interface vector "ROTOR 1 NONROTATING HUB MOMENT" contains the structural moments on the hub node. (By creating the regulator loop with OPSTR=1, this quantity is available as an interface, solved in the part "ROTOR 1 REGULATOR SENSOR"). However, these are time-varying loads, so a filter component must be constructed to generate the mean hub moments.

7) Analysis Input

Copies of the input files are included with the CAMRAD II sample cases. The namelist files are:

YHHCINPUT.COM, YHHCINPUT.LIST

and

YHHC.COM, YFLAPCON.COM, YHUBCON.COM

are the job files.
This document summarizes the tasks involved in analyzing a rotor with a pendulum absorber using CAMRAD II. In particular, the use of core components to construct a flapwise pendulum absorber is demonstrated.

1) Baseline Input

A model constructed by the rotorcraft shell is required as a starting point. The sample case analyzing a bearingless rotor in a wind tunnel was used here: namelist input file BEARINGLESS.LIST; and job SAMPLE8. This case has a single five-bladed rotor; the rotor radius is 20 ft. Some changes were made to the namelist input file BEARINGLESS.LIST, including changing the number of blades to four. Then this file was modified add a flapwise pendulum absorber.

A CAMRAD II job was needed to check the work. Starting with SAMPLE8, trim sensors of the blade hub loads and the blade position were turned on (parameters MHSEN=1 and MPSEN=1). The rotor was analyzed in forward flight (advance ratio $\mu = 0.4$) so the pendulum absorber would be active. The flutter analysis was used for a single blade without aerodynamics, to obtain the blade frequencies. This job was run for the baseline input.

2) Existing Pieces

Next the system pieces of the baseline model must be examined. For this purpose the baseline job was run with only initialization (TMTASK=2 and FLTASK=2), and with NPRNTC, PCLASS, PTYPE, PNAME as required to print the input for the system pieces of interest (and perhaps without aerodynamics, to reduce the number of pieces printed). First the flexbeam elements were printed: "ROTOR 1 FLEXBEAM m ELEMENT 1" (with and without aerodynamics, to establish that aerodynamics does not change the flexbeam element definition). The flexbeam element was observed to have 7 locations and connections, 2 structural dynamic interfaces (at the beam ends),
and 5 sensors (for mode and position output). Then the following existing pieces of interest were identified:

- **components:** ROTOR 1 FLEXBEAM 1 ELEMENT 1
- **SD interfaces:** ROTOR 1 BLADE 1 ROOT/FLEXBEAM
- **output:** ROTOR 1 FLEXBEAM 1 POS 0.2500R
  ROTOR 1 FLEXBEAM 1 MODE AT 0.0R
- **rigid response:** DOTOR 1 FLEXBEAM 1 ELEMENT 1 RGD
- **variable response:** DOF ROTOR 1 FLEXBEAM 1 ELEMENT 1
  INTROTOR 1 BLADE 1 ROOT/FLEXBEAM
  O ROTOR 1 FLEXBEAM 1 POS 0.2500R
  OROTOR 1 FLEXBEAM 1 MODE AT 0.0R

These pieces are involved in the blade model; they are described in Volume III (Rotorcraft Shell) of the CAMRAD II documentation. These system pieces must be studied to understand the baseline model, and as examples for construction of the new model.

Note that the core input constructed by the shell can be changed by shell parameters. In particular, changing the number of sensors or output selected, the aerodynamic panels, or the number of nodes could change the locations, connections, interfaces, and sensors constructed for these components.

### 3) Geometry of Modified Model

The bearingless rotor constructed by the rotorcraft shell has a flexbeam extending to \( r/R = 0.25 \). The modified model adds a flapwise pendulum absorber. One approach would be to use the rotorcraft shell input to define a point mass, and then change the pinned interface to a joint plus cantilever interface, and change the inertial properties of the point mass. However, the rotorcraft shell does not put point masses on the hub or on the flexbeam, so this approach was not used. Instead, a location/joint/connection was added to the flexbeam component, and a new component and a new interface were defined for the pendulum.

For a flapwise pendulum absorber, a flapwise joint is created on the flexbeam, at radial station \( r/R = 0.20 \). The beam axes have the x-axis spanwise and the y-axis chordwise; hence the flapwise joint axis is the y-axis. The connection axes are parallel to the blade frame axes (y-axis spanwise, x-axis chordwise), so the interface axes as usual are the blade frame axes. The pendulum is modelled as a rigid body, with one interface to the joint on the flexbeam. The pendulum body axes are parallel the blade frame axes, so the pendulum cg offset is in the y-axis direction, and the pendulum moment of inertia about the x-axis is nonzero. A pendulum mass of
1 slug is used. For a point-mass pendulum with length \( L \) and hinge at radial station \( e \), the natural frequency is

\[
\omega^2 = 1 + e/L
\]

So with \( e = 4 \) ft, \( L = 0.266667 \) ft gives a natural frequency of \( 4/\text{rev} \) (neglecting coupling with the blade motion). Hence the pendulum has a cg offset of \( Z_{CG(2)} = 0.2666667 \) ft and moment of inertial of \( I_{xx} = 0.711111 \) slug-ft\(^2\).

All modification must be made for each of the four blades of the rotor. The response of blades 1 to 3 are children of blade 4, for the degrees of freedom and interface.

4) Modified Model

Next the core input required to construct this model was added to the namelist file. The input was defined using information from the listing of the pieces constructed by the shell; and from Volume IV (Input) and Volume V (Components Input). The following core pieces were defined:

revised:
components: ROTOR 1 FLEXBEAM 1 ELEMENT 1

new:
components: ROTOR 1 PENDULUM 1
SD interfaces: ROTOR 1 BLADE 1 PEND/FLEXBEAM
output: ROTOR 1 BLADE 1 POS  PENDULUM
                   ROTOR 1 BLADE 1 MODE  PENDULUM
rigid response: DOF ROTOR 1 PENDULUM 1 RGD
variable response: DOTOR 1 FLEXBEAM 1 ELEMENT 1 JNT
                   INTROTOR 1 BLADE 1 PEND/FLEXBEAM
                   OROTOR 1 BLADE 1 POS  PENDULUM
                   OOTOR 1 BLADE 1 MODE  PENDULUM

A new location, joint, connection, and interface were added to the existing flexbeam component; new position and mode sensors for the pendulum motion were also defined. All modification must be made for each of the four blades of the rotor.
5) Check of Modified Model

The rest positions of the components can be checked by drawing the system; the INPUT program provides the required geometry data. To check the input, the INPUT program was run; and the analysis program was run with only initialization (TMTASK=2 and FLTASK=2). Then the analysis was run without aerodynamics. The pendulum damping (CLIN=CEQUIV=3.) was selected to provide an appropriate damping of the pendulum mode. Finally, the CAMRAD II trim and flutter analysis was run to check the model, comparing the solution with the baseline. Hub load, pendulum motion, and the blade frequencies were examined.

6) Analysis Input

Copies of the input files are included with the CAMRAD II sample cases. The namelist files are:

YPENINPUT.COM, YPENINPUT.LIST

and

YPENDULUM.COM

is the job file.