Università degli studi di Trento
Dipartimento di Ingegneria Industriale

Relazione

Mechanical Vibrations

System identification of a 3 DOF system

Relatore
Prof. Daniele Bortoluzzi

Correlatore
Dott. Egidio Labbate

Candidato
Francesco Aregntieri

ID: 183892

Anno Accademico 2015/2016

Contents

Chapter 1

Introduzione

1.1 Contesto applicativo

Ciao ragazzi :D questo è un template che potete utilizzare per scrivere la vostra tesi in LaTeX(sì, il nome di questo linguaggio è tutto un programma...)

Cos'è LaTeX? Cercatevelo su Wikipedia[1]!

A parte gli scherzi... è un linguaggio che vi permette (in poche parole) di creare documenti accademici (e non) con uno stile molto professionale. Gran parte del lavoro sporco (creazione dei capitoli, delle sezioni, dell'indice, della bibliografia, gestione dei margini, ecc...) viene interamente gestito da LaTeX, le poche cose da configurare sono già state impostate in questo template... (quindi in poche parole avete già tutto pronto, stronzi!)

In questo pdf è spiegato un po' come far funzionare il tutto, ovvero:

- Dove scaricare l'IDE e come configurarlo
- Come scaricare il compilatore
- Come iniziare a personalizzare questo template

Cercherò di utilizzare più elementi LaTeXpossibile nello scrivere questa guida (tabelle, elenchi puntati, footnote, immagini...) così quando andrete a leggere il codice vi imparate pure qualcosa, caproni (<3)

1.2 Motivazioni e obiettivi

Il principale motivo che mi spinge a creare questo pdf è quello di risparmiarvi gran parte delle rotture che si trovano quando ci si avvicina al mondo LaTeX... insomma mi auguro

[1]http://it.wikipedia.org/wiki/LaTeX

che questa guida vi permetta di avere un buon punto d'inizio.

Come già spiegato nella sezione precedente, LaTeXoffre tantissimi servizi utili ed uno stile professionale unico, cose che su altri programmi (come Microsoft Word) potete anche sognarvi.

Insomma... con LaTeXpotete presentare una tesi fatta come si deve :)

1.3 Risultati raggiunti

Nella Figura 1.1 potete ammirare quanto LaTeXsia più figo di Microsoft Word, ooooooh...

Figure 1.1: Oooooooh che figo LaTeX, oooooooooh

1.4 Organizzazione della tesi

Vi spiego brevemente quali sono le parti di questo template:

Susanna.tex Questo è il file principale del template: contiene le impostazioni generali e la struttura della tesi. Ricordate di impostarlo come documento master ogni volta che iniziate a lavorare alla tesi (ovviamente potete rinominarlo, nabbazzi)

frontespizio.tex Sarebbe la copertina della tesi, nonchè la prima pagina. Contiene nome dell'università, del dipartimento, nome della tesi bla bla bla... io ho scelto un argomento di Fisica molto importante <3

dedica.tex Una pagina dove scrivete a chi dedicate la tesi (Susanna <3)

introduzione.tex Il file che contiene questo capitolo introduttivo (vi consiglio di creare appunto un file .tex per ogni capitolo). Le 4 sezioni di questo capitolo (Contesto applicativo, Motivazioni e obiettivi, Risultati raggiunti e Organizzazione della tesi) sono le 4 sezioni standard da inserire nell'introduzione di una tesi, quindi vi consiglio di lasciarle così

start.tex Contiene l'unico capitolo utile di questo documento: spiega come scaricare l'occorrente e come configurare il tutto per lavorare con LaTeX

conclusione.tex Contiene la conclusione (YOU DON'T SAY)

bibliografia.bib Contiene i dati relativi alle fonti che citerete nella tesi (ad esempio, ora sto citando un libro sugli algoritmi genetici [1], l'unico inserito nella bibliografia di questo template)

IEEEtran.bst È lo stile della bibliografia, non lo toccate

imgs Cartella contenente le immagini (YOU DON'T SAY AGAIN)

Chapter 2

Come fare le cose

In questo capitolo vediamo la roba smanettosa per iniziare a smanettare

2.1 Occorrente

Roba da scaricare e installare (Tabella 2.1).

Table 2.1: Tabella vergognosamente inutile

File	Piattaforma	Dimensioni
TexMaker	Windows	46.3 MB
TexMaker	Mac	40.7 MB
MiKTeX	Windows	154.1 MB
MacTex	Mac	2.3 GB
Template Susanna	Multiglobale-powa	4 MB (circa)

2.1.1 L'IDE

Allora... per prima cosa vi serve un IDE, ovvero un programma che vi funga da editor e compilatore (in realtà il compilatore si scarica a parte ma vabbè). Ce ne sono molti in giro ma io vi consiglio *TexMaker*[1] per due motivi:

1. È molto intuitivo è ben fatto
2. Esiste sia per Mac che per Windows

Non dovreste avere problemi con il download e l'installazione (vi state per laureare porca paletta, non devo spiegarvi pure questo).

2.1.2 Il compilatore

Per quanto riguarda il compilatore il discorso è un po' più complicato. Armatevi di pazienza e scaricate *MiKTeX*[2] se avete Windows oppure *MacTex*[3] se avete un Mac (mi dispiace ma non conosco un compilatore LaTeX per Linux... se lo trovato fatemelo sapere che aggiorno la guida). Entrambi questi programmi inglobano un ambiente di sviluppo LaTeXcostituito da diversi compilatori che il nostro IDE riconoscerà automaticamente.

P.S. Prima che cominciate a strapparvi i capelli, sì, *MacTex* occupa più di 2 GB... questo perchè comprende tutti i pacchetti necessari per LaTeX, mentre *MiKTeX* (che occupa solo 150 mb) li scarica volta per volta.

2.1.3 Il template

Trovate il sorgente di questo template ad un link dropbox non meglio specificato[4]

2.2 Configurazione dell'IDE

Oooh, ora che avete installato IDE e compilatore, lanciate l'IDE. Principalmente dovete fare tre cose una volta avviato:

1. Aprite il file Susanna.tex del template
2. Andate su Opzioni -> Definire il documento corrente come Master (questo serve per dire all'IDE che gli altri documenti sono inclusi in un documento master e che

[1] http://www.xm1math.net/texmaker/
[2] http://miktex.org/download
[3] http://mirror.ctan.org/systems/mac/mactex/MacTeX.pkg
[4] https://www.dropbox.com/sh/1f06sd7eprongvl/dKsfID1Kwc

quindi, al momento della compilazione, non devono essere trattati come documenti separati)

3. Andate nelle preferenze dell'IDE nella sezione Compilazione Rapida e personalizzate la compilazione tramite l'assistente-wizard. Essenzialmente dovete configurarla in modo da effettuare 3 compilazioni: PdfLatex, BibTex e di nuovo PdfLatex. Oltre a queste tre compilazioni aggiungete una quarta opzione ovvero la visualizzazione pdf.

Vi spiego meglio il punto 3... praticamente ci sono più compilatori diversi, che svolgono operazioni diverse... ma a noi interessano solo due compilatori, ovvero PdfLatex (che compila il codice LaTeXin un documento pdf) e BibTex (che compila la bibliografia). Le compilazioni sono 3 e in quel preciso ordine perchè altrimenti la bibliografia non viene compilata bene (non chiedetemi perchè). Per evitare di dover eseguire manualmente le diverse compilazioni, *TexMaker* vi dà la possibilità di utilizzare la Compilazione Rapida che esegue automaticamente queste operazioni con un click. Configuratela come vi ho spiegato e non avrete problemi.

2.3 Siamo pronti

Abbiamo configurato l'IDE ed il (i) compilatore(i). Ora premendo sulla freccina della Compilazione Rapida (oppure premendo F1) potrete compilare il vostro codice LaTeXin pdf. Fate una prova compilando il template (il pdf purtroppo, così come tutti gli scarti della compilazione, verranno generati nella stessa cartella del sorgente...).

E ora? Ora create i vostri capitoli copiando la struttura di *start.tex* e di *introduzione.tex* ed integrateli nel documento master :) se avete bisogno di ulteriori dettagli sulla sintassi LaTeXvi consiglio di farvi un giro nella sezione Tex di *Stack Exchange*[5]: è tipo Yahoo Answers ma focalizzato ovviamente su LaTeX:)

[5] http://tex.stackexchange.com/

Chapter 3

Eurocopter AS350

This chapter opens with an introduction to the helicopter named Eurocopter AS350, its development over the years and some technical features. In follow deals with the study of the tail in particular . . . , where the aeroelastic problem will be neglected and as a result all the aerodynamic components will be removed from the model.

3.1 Introduction

The **Eurocopter AS350 Écureuil** (*Squirrel*) is a single-engine light helicopter originally designed and manufactured in France by Aérospatiale, now Airbus Helicopters. In North America, the AS350 is marketed as the AStar. The AS355 Ecureuil 2 is a twin-engine variant, marketed in North America as the TwinStar. The Eurocopter EC130 is a derivative of the AS350 airframe and is considered by the manufacturer to be part of the Écureuil single-engine family. In the early 1970s, Aérospatiale decided to initiate a new development programme to produce a suitable replacement for the aging Aérospatiale Alouette II. While the Aérospatiale Gazelle, which had been developed in the 1960s and 1970s, had been met with numerous orders by military customers, commercial sales of the type had been less than anticipated, thus the need for a new civil-orientated development was identified. The development of the new rotorcraft, which was headed by Chief Engineer René Mouille, was focused on the production of an economic and cost-effective aerial vehicle, thus both Aérospatiale's Production and Procurement departments were heavily involved in the design process. One such measure was the use of a rolled sheet structure, a manufacturing technique adapted from the automotive industry; another innovation was the newly developed Starflex main rotor. It was also decided that both civil and military variants of the emergent helicopter would be developed to conform with established military requirements[2].

Figure 3.1: Helicopter takeoff
Di Fabien1309 - Own work

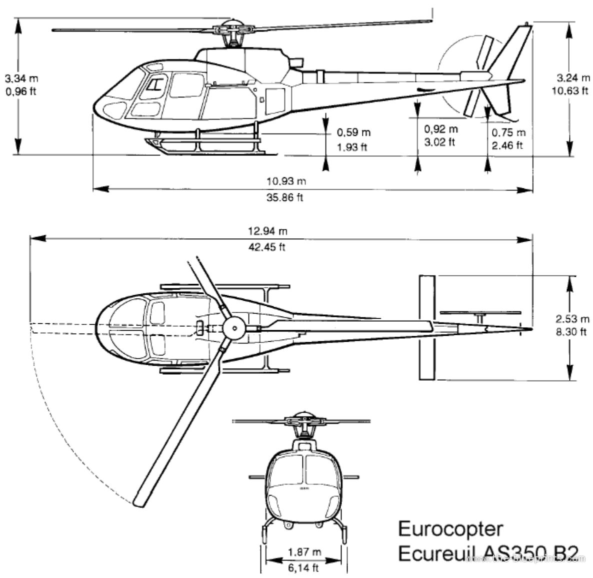


Figure 3.2: AS350 blueprints

General characteristics	
Length	12.94 m
Height	3.34 m
Main rotor diameter	10.69 m
Empty weight	1220 kg
Max takeoff weight	2250 kg
capability	2500 kg
Propulsion	
Powerplant	1 x Turbine Turbomeca Arriel 1D1
Power	546 kW
Performance	
Maximum speed	287 km/h
Range	476 km
service ceiling	6100 m

Table 3.1: Main characteristics AS350

3.2 Geometric model

The geometric model is shared among three different configuration where was enfatizated a different approach to evalute the repsonce of structural elemtents in different condition of approximation. Infact the first case is an evalution of the simple model where considering only tail like as a cantilever beam. While in the second case considering also the trasmission shaft rigidly linked at the tail and the presence of a lumped mass to simulate the persence of the block of rotor in the proximity the end of tail. Finally consider a third model where we considering the shaft's weight is distributed along the lenght of tail like as lumped mass, we consider as before another concentrated mass to represent the block of rotor at the tail's end. The model was obtained by the revolution of seven contiguous segments with respect to axis placed in the plane xz, thus obtaining a truncated cone having a radius $325\,mm$ at the base and a radius $50\,mm$ for the minor base, the whole extension is $5.2\,m$. The cone trunk was highlighted in twentyfour equal segments in order to obtain a bse for the reallization of components such as stringer, horizontal stabilizer attachment, stiffners and ribs. The result can be seen in the figure 3.3.

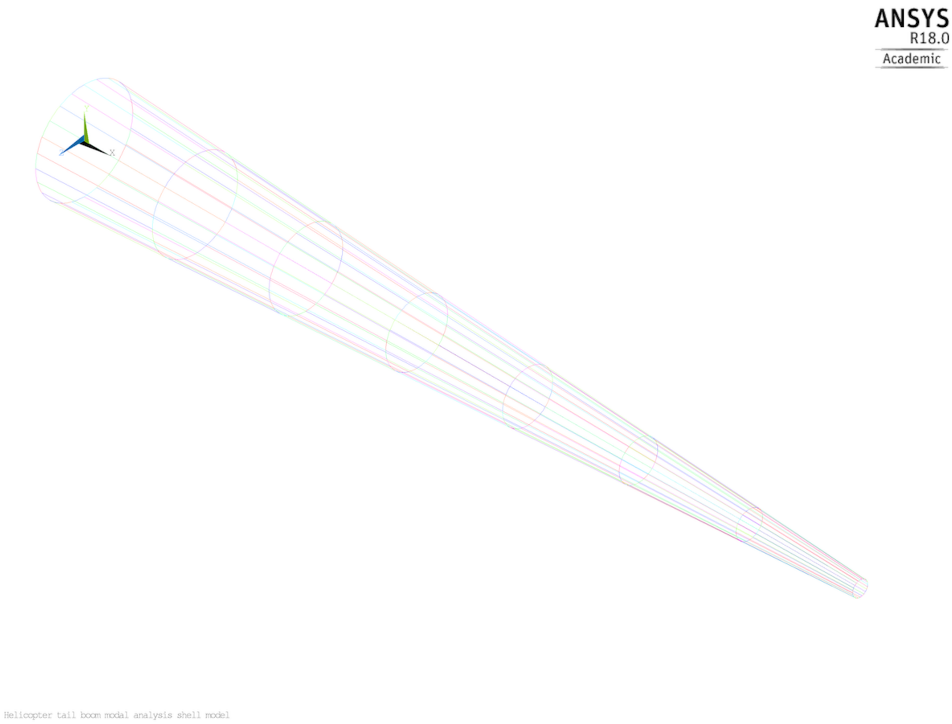


Figure 3.3: AS350 ansys model

3.3 Mesh model

After the geometric model has been realized, the finite element model has the same types of shared elements to ensure consistency between the models, as well as the texture of the matrices used to study the different models.

3.3.1 Simple model

For structural elements such as:

- stringer;
- horizontal stabilizer attachment;
- stiffners;
- ribs.

Type elements have been used: BEAM189, this type of element has been assigned a rectangular section and a square as shown in 3.4. Specifically, the rectangular section, fig. 3.4a, is intended to form the stinger and longerons components, instead the square section, fig.3.4b, perform the ribs, then the structure skeleton are represented in fig. 3.5.

The material used is steel with the following properties:

- Young's modulus, 205 $[GPa]$
- Poisson's ratio, 0.29
- Density, 7850 $[kg/m^3]$

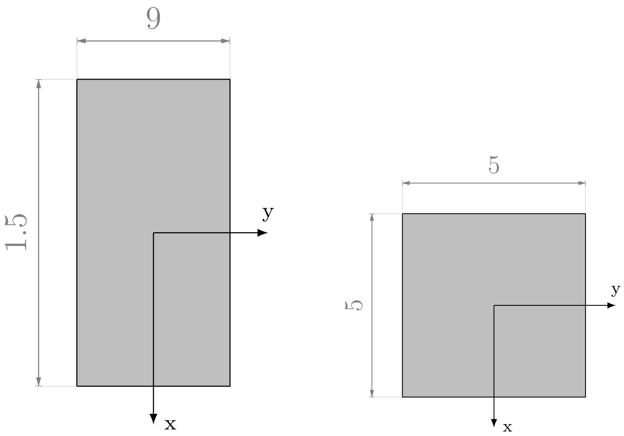
On the other hand, to realize the mantle covering the support structure was used type element: SHELL181 with aluminum used as material with the following properties:

- Young's modulus: 64 $[GPa]$;
- Poisson's ratio: 0.34;
- Density: 2700 $[kg/m^3]$.

Finally, you can see the full model in the figure 3.6.

3.3.2 Simple model with shaft

In this case, the geometric model has the same features as the previously described model, with the addition of a transmission shaft on top of the cone. the shaft has been modellated with type of elements BEAM189 with tube section, fig. 3.7, and realized in duralumin characterized by the following property:



(a) *Rectangular.* **(b)** *Square.*

Figure 3.4: Section for element BEAM189

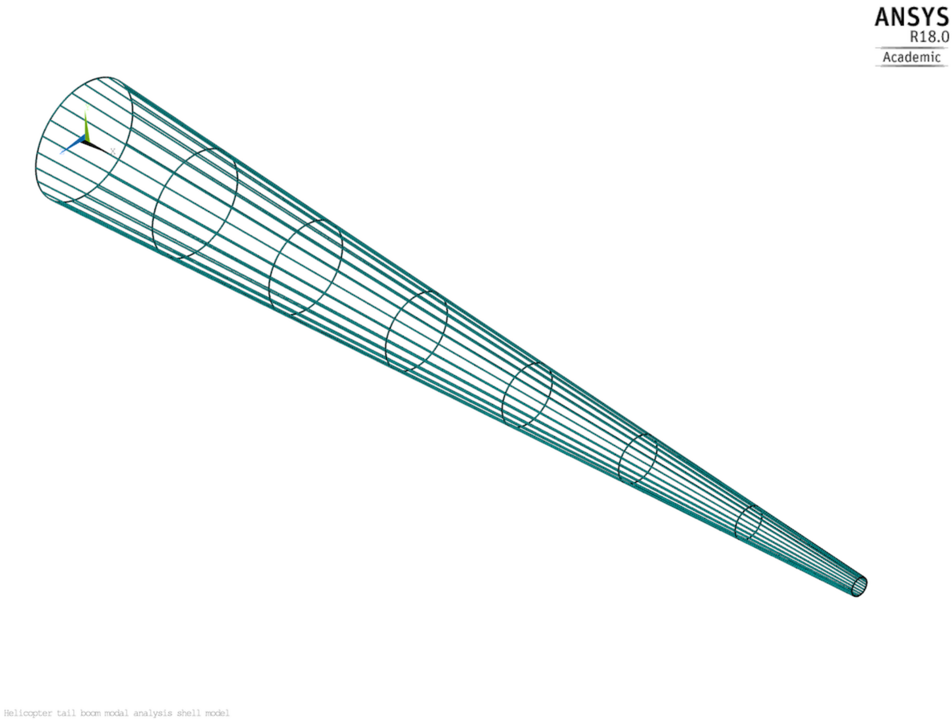


Figure 3.5: Skeleton mesh model

- Young's modulus: $72\,[GPa]$;
- Poisson's ratio: 0.33;
- Density: $2810\,[kg/m^3]$.

To make rigid connections, elements of type: MPC184, with key option: 1,used to block six degrees of freedom. Moreover, they have resistance properties equal to those of the extruded steel for the beam elements. Note that they were used without densities be-

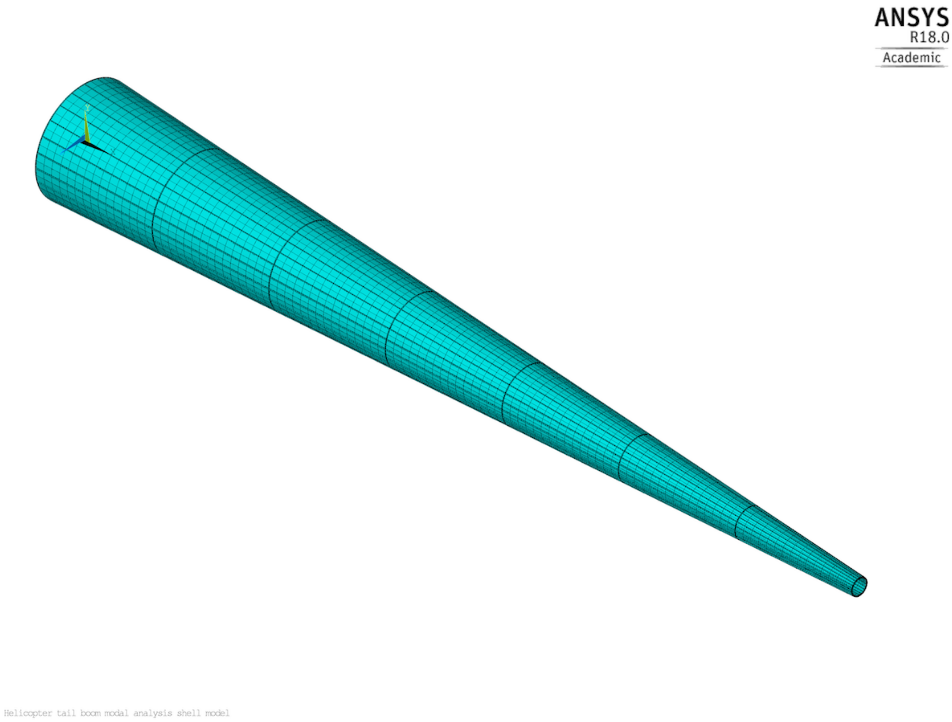


Figure 3.6: Complete mesh model

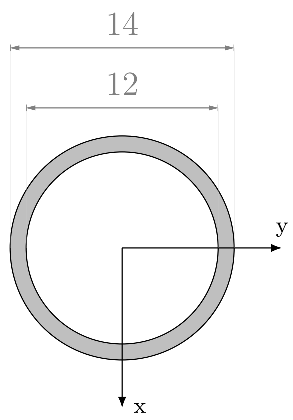
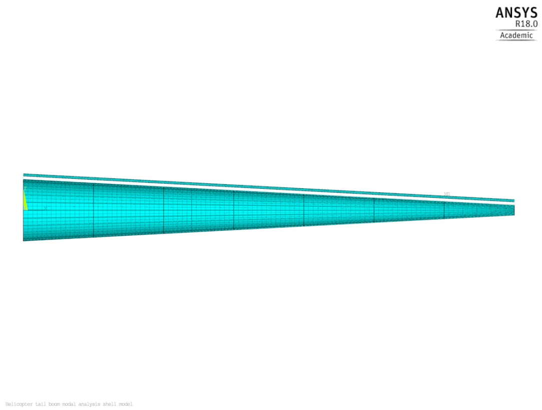


Figure 3.7: Shaft's tubolar section

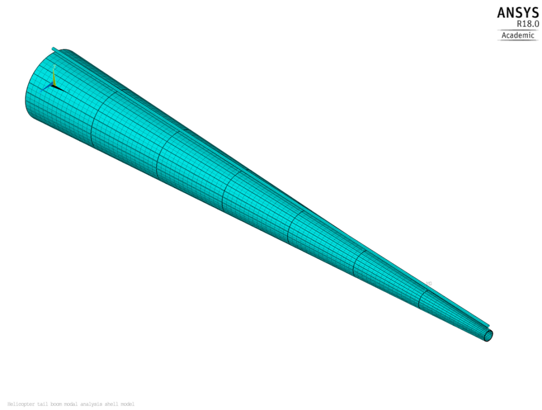
cause the default section in ANSYS is equal to 1 for default, this because they caused an overestimation in mass calculation and then in the derived natural frequencies of the modal analysis. Finally, we consider the mass concentrated near the end of the queue representing the tail rotor block, this simplification is static represented by elements of the type: MASS21, to which the following mass and inertia properties were assigned:

- concentrated mass: $30\,[kg]$;
- concentrated inertia: $1\,[kg * m^2]$.

The result is visible in fig. 3.8



(a) *side view.*

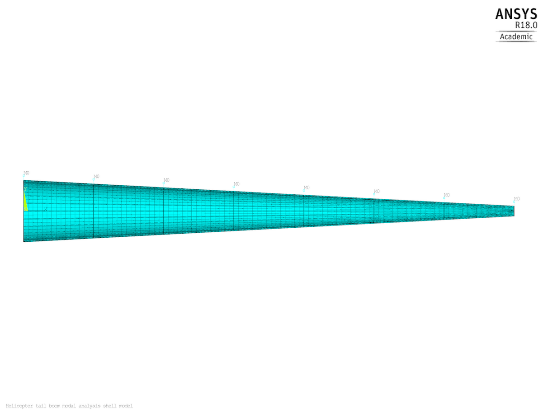


(b) *isometric view.*

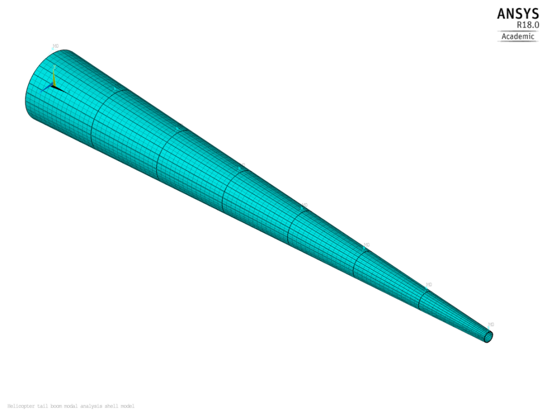
Figure 3.8: Shell model with shaft

3.3.3 Simple model with lumped mass

Starting with the simple model, in this case, the rigid rigid links are added as done to the model with the shaft, but in this case in the connecting terminal part are added the concave mass portions representing the distributed shaft weight. In fact, these are modeled with MASS21 element type and have the property of having the mass equal to a fraction of the shaft. Also in this case consider the rotor block as near the tail as modeled in the previous case, the result is visible in fig.3.9.



(a) *side view.*



(b) *isometric view.*

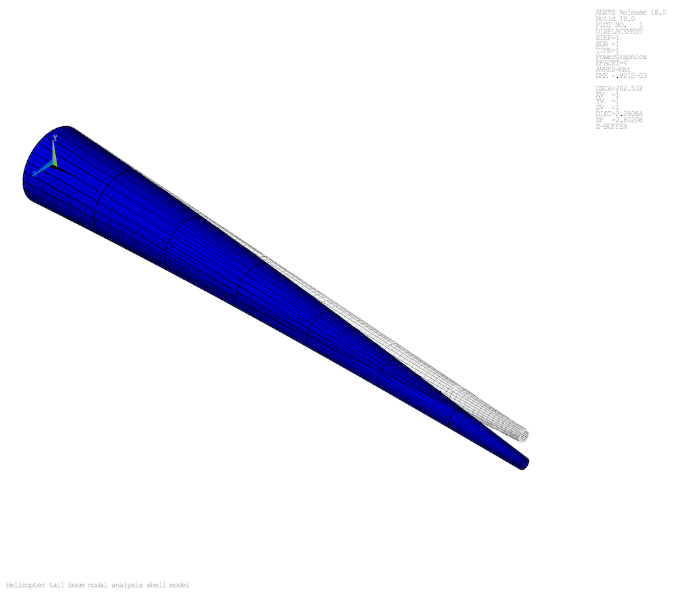
Figure 3.9: Simple model with lumped mass

3.4 Preliminary analysis

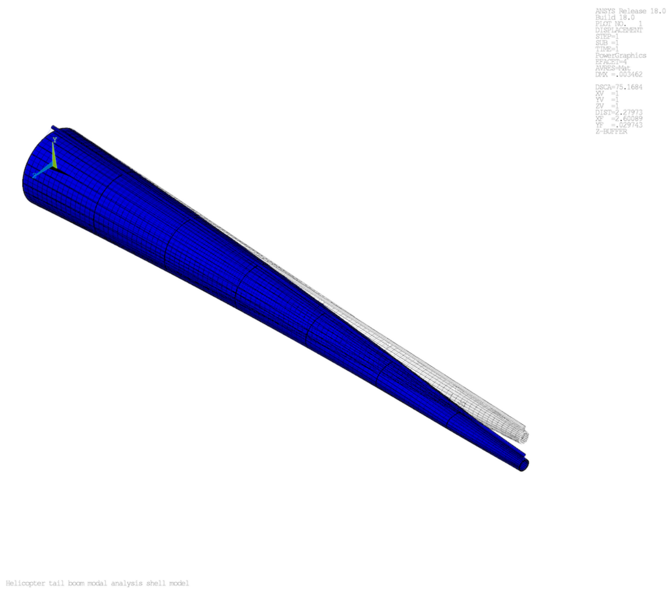
In the preliminary analysis we have investigated the deformations caused by the proper weight of the structure, in fact this is only subject to gravity acceleration, using an approximate value equal to 9.81 $[m/s^2]$. Each model has been studied by making it a constraint with a cone at the base of the cone, so it is possible to make a parallelism with a cantilever beam. In the following, table 3.2, shows deflection and stress values while you can observe the effects of flexural tension distribution for structures and the effect due to the presence or not of the shaft, osservable in figures 3.9–3.10.

Case	Displacement $[m]$
Simple model	0.00092
With shaft	0.00346
Lumped mass	0.00459

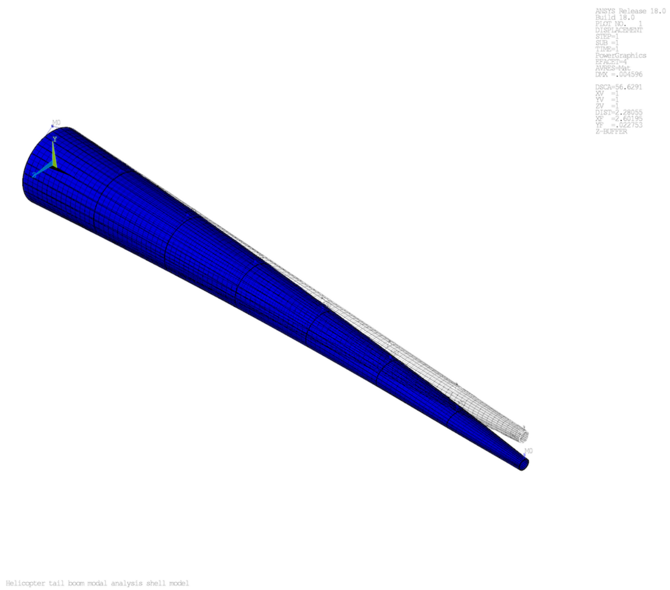
Table 3.2: Deformation values and stress of the models analyzed



(a) *Simple model.*

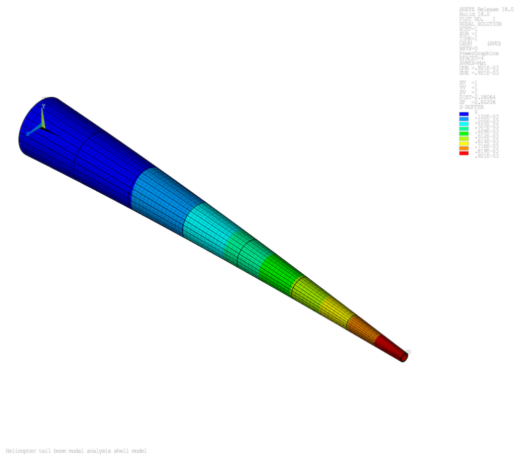


(b) *Model with Shaft.*

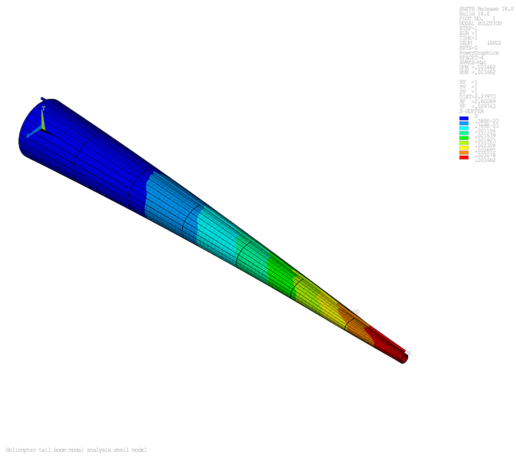


(c) *Model with lumped mass.*

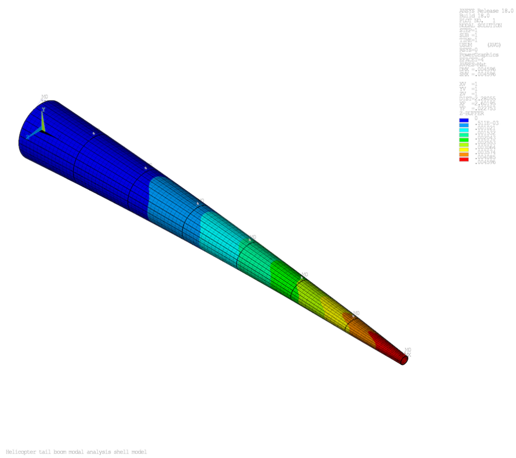
Figure 3.9: Maximum displacement



(a) *Simple model.*



(b) *Model with Shaft.*



(c) *Model with lumped mass.*

Figure 3.10: Nodal soluiton

3.5 Modal Analysis

Mode	Frequence [Hz]
1	21.14089
2	21.15116
3	40.79259
4	40.80173
5	60.04059
6	60.05571
7	74.74651
8	74.76411
9	79.03090
10	79.08638

Table 3.3: Natural frequencies for the simple model

Mode	Frequence [Hz]
1	9.52497
2	19.89946
3	31.80325
4	42.65110
5	42.72723
6	57.92407
7	63.11685
8	63.51130
9	74.61297
10	78.93013

Table 3.4: Natural frequencies for the model with shaft

Mode	Frequence [Hz]
1	8.28224
2	9.64315
3	17.48365
4	37.98169
5	40.80936
6	51.66412
7	59.71804
8	59.95684
9	64.32600
10	65.32928

Table 3.5: Natural frequencies for the model with lumped mass

Chapter 4

Tail rotor

In this chapter we analyze the tail rotor in condition decoupled from rest of the structure because in this way it's possible observe in detail the behaviour when is performed a modal analisys and a transient analisys. In the following the tail rotor will be attached at the rest of the structure thanks to the macro realized both for the truss model and for the shell model.

Chapter 5

Complete Model

Now describe the procedure to attach the tail rotor to two structure: the truss model and shell model, respectively. In both base model we add a part that realize the nodes that represent rigid link between the tail structure and tail rotor. After call the macro to realize the component, realized connection by rigid elements between these and component previous realized, thus you can obserse the result in figure.

Chapter 6

Conclusioni

Spero che la guida possa servirvi ragazzuoli!

Ringrazio di cuore Giordano Cardillo e Matteo Merola per avermi aiutato (e sopportato) a capire come funziona questo maledetto LaTeX! :P

Ringraziate Giordano anche per aver creato dei loghi vettoriali decenti da utilizzare :D (li trovate nella cartella delle immagini).

Vi voglio bene <3 (tranne a Rosangela u.u)

Guida scritta da Lorenzo Valente [1]

[1] http://facebook.com/lorenzo.valente

Chapter 7

Listing

Shell Model

Listing 7.1: `ShellModel.txt`

```
/COM,  ----------------------------------------------------------------------------------
/COM,   PROBLEM: MODAL ANALYSIS of the helicopter's TAIL BOOM
/COM,  ----------------------------------------------------------------------------------
/COM,   Natural frequencies and modal shapes of an helicopter's tail boom
/COM,  ----------------------------------------------------------------------------------

FINISH
/CLEAR, START, NEW
/FILNAM, Shellmodel
/TITLE, Helicopter tail boom modal analysis shell model
/UNIT, SI
/INQUIRE, StrJobname, JOBNAME
*USE, testperson.mac

! >>>>> MODEL PARAMETERS <<<<<
*SET, Pi, ACOS(-1)              !Pi constant
*AFUN, DEG                      !Specify units for angular measures [DEG], specify after
     function *AFUN
*SET, eps, 10e-3                !precision interval

! >>>>> MATERIAL PROPERTIES <<<<<
!*** steel
*SET, SteelEyounG, 205e9                        ![Pa] Young's modulus
*SET, SteelNi, 0.29                             !Poisson's ratio
*SET, SteelDensity, 7850                        ![Kg/m^3]

!*** Aluminium
*SET, AluminiumEyounG, 64e9                     ![Pa] Young's modulus
*SET, AluminiumNi, 0.34                         !Poisson's ratio
*SET, AluminiumDensity, 2700                    ![Kg/m^3]

!*** Element size
*SET, E_length_ct, 50e-3                        ![m]

```

```
!geometric parameter
*SET, Lt, 5.2                              ![m] total tail lenght
*SET, RadBaseTail, 0.65/2                  ![m]
*SET, RadEndTail, 0.10/2                   ![m]
*SET, alpha, atan((RadBaseTail-RadEndTail)/Lt)  ![deg]

!geometric parameters section stiffner
*SET, StiffnerBase, 5e-3
*SET, StiffnerHeight, 5e-3

!geometric parameters section Horizzontal Stabilizer
*SET, HorizStabBase, 0.009
*SET, HorizStabHeigth, 0.0015

!geometric parameter mantle
*SET, ThichknessMantle, 5e-4               ![m]

!other parameters
*SET, NumberDivisonSurface, 24  !division cone's area

/COM, ----------------------------------------------------
/COM,  MODEL DEFINITION
/COM, ----------------------------------------------------
! >>>> DEFINE GEOMETRY <<<<
! Reduced integration Timoshenko beam element, with quadratic shape functions
/PREP7
! Define axis rotation
K, 1, 0, 0, 0
K, 2, 5, 0, 0

! create keypoints
K, 3, 0, RadBaseTail, 0                                       !start keypoint tails
*GET, MaxKp, KP, 0, num, max                                  !extract max keypoint
*SET, StartKpTail, MaxKp                                      !store in variable
*DIM, SegmentRadius, ARRAY, 7, 1                              !store in array radius of segment distance between 0 - Y before rotating
*DO, i, 1, 7, 1
        *GET, MaxKp, KP, 0, NUM, MAX                          !extract max keypoint
        K, MaxKp+1, Lt/7*i, RadBaseTail-Lt/7*i*tan(alpha), 0  !create keypoint
        SegmentRadius(i) = RadBaseTail-Lt/7*i*tan(alpha)
*ENDDO
*GET, MaxKp, KP, 0, num, max                                  !extract max keypoint
*SET, EndKpTail, MaxKp                                        !store in variable end's tail

! define keypoint mantle'tail
K, Maxkp+1, 0, RadBaseTail, 0
*GET, StartKpMantle, KP, 0, num, max
K, MaxKp+2, Lt/7*i, RadEndTail, 0
*GET, EndKpMantle, KP, 0, num, max

*DO, j, StartKpTail, EndKpTail-1, 1
        L, j, j+1
*ENDDO
*GET, LenghtLineRference, LINE, 1, leng                               !extract lenght segment line's cone generator
```

```
! generate cone
*GET, NumberLine, line, 0, count                                            !count lines
*GET, StartNumberLine, line, 0, num, min                                    !extract first
    number line
*GET, NumberLine, line, 0, count                                            !count lines
*GET, StartNumberLine, line, 0, num, min                                    !extract first
    number line
LSEL, S, LOC, X, 0, Lt
AROTAT, ALL, , , , , , 1, 2, 360, NumberDivisonSurface                      !generate cone

!divide line in array straight line and curved line
!to manage in next step meshing of elements
LSEL, S, LENGTH, , LenghtLineRference, LenghtLineRference
*VGET, StraighLine, LINE, , LLIST, , , 0
ALLSEL, ALL
LSEL, U, LENGTH, , LenghtLineRference, LenghtLineRference
*VGET, AcrLine, LINE, , LLIST, , , 0

*GET, StartNumberLine, line, 0, num, min
LSEL, ALL

/VIEW, 1, 1, 1, 1
/ANG, 1
/REP, FAST
LPLOT
*USE, generateimages.mac

! >>>> MESH <<<<
/ESHAPE, 1
!set material properties and element
ET, 1, BEAM189
MP, EX, 1, SteelEyounG
MP, NUXY, 1, SteelNi
MP, DENS, 1, SteelDensity

!***Horizzontal Stabilizer
SECTYPE,  1, BEAM, RECT, HorizzontalStabilizer, 0
SECOFFSET, CENT
SECDATA, HorizStabBase, HorizStabHeigth

*GET, LenghtStraighLine, 'PARM', StraighLine, DIM, X
LSEL, S, LINE, , StraighLine(1), StraighLine(LenghtStraighLine)
LESIZE, all, E_length_ct
LATT, 1, , 1, , , , 1
LMESH, ALL
LSEL, NONE
/ESHAPE, 1
/REPLO

!***stiffners
SECTYPE,  2, BEAM, RECT, Stiffeners, 0
SECOFFSET, ORIG
SECDATA, StiffnerBase, StiffnerHeight

*GET, LenghtAcrLine, 'PARM', AcrLine, DIM, X
```

```
LSEL, S, LINE, , AcrLine(1), AcrLine(LenghtAcrLine)
LESIZE, ALL, , , , 3
LATT, 1, , 1, , , , 2
LMESH, ALL
LSEL, NONE
/REPLO

/VIEW, 1, 1, 1, 1
/ANG, 1
/REP, FAST
EPLOT
*USE, generateimages.mac

!***Mantle
!set material properties and element
ET, 2, SHELL181
MP, EX, 2, AluminiumEyounG
MP, NUXY, 2, AluminiumNi
MP, DENS, 2, AluminiumDensity
SECT, 3, SHELL, , mantle
SECDATA, ThichknessMantle, 2, 0.0, 3
SECOFFSET, top

TYPE, 2
SECNUM, 3
AESIZE, ALL, E_length_ct,
MSHAPE, 0, 2D
MSHKEY, 1
AMESH, ALL
SAVE

/VIEW, 1, 1, 1, 1
/ANG, 1
/REP, FAST
EPLOT
*USE, generateimages.mac

!***Boundary conditions at base of tail
ALLSEL, ALL
LSEL, S, LOC, Y, -(RadBaseTail*2)+eps, (RadBaseTail*2)+eps
LSEL, S, LOC, x, 0, 0
DL, ALL, , ALL, 0

*USE, calcmass.mac, StrJobname(1)
FINISH

/COM, ------------------------------------------------
/COM,  Preliminary static analysis - Prestress
/COM, ------------------------------------------------
! >>>> SOLUTION <<<<
/SOLU
ANTYPE, STATIC, NEW              !Preliminary static analysis
PSTRES, ON                       !Includes prestress effects
Acel, , 9.81                     !gravitational acceleration

SOLVE
```

```
FINISH

! >>>> POSTPROCESSING <<<<
/POST1
PLDISP, 1
*USE, generateimages.mac
PLNSOL, U, SUM, 0, 1.0
*USE, generateimages.mac
PLNSOL, S, EQV, 0, 1.0
*USE, generateimages.mac


/COM, ------------------------------------------------
/COM,  Modal analysis
/COM, ------------------------------------------------
! >>>> SOLUTION <<<<
/SOLU
ANTYPE, MODAL, NEW
PSTRES, ON                              ! Includes prestress effects
MODOPT, LANB, 10                        ! First 10 modes are extracted
LUMPM, OFF                              ! lumped/consistent mass matrix
MXPAND, 10, , , Yes                     ! First 10 modes are expanded, calculate element
     stress

SOLVE
FINISH

! >>>> POSTPROCESSING <<<<
/POST1
SET, LIST                       !List natural frequencies
SET, 1, 1                       !Loads first mode
PLDISP                          !Displays modal shape

*DIM,ModalFreq,ARRAY,10,1
*Do, i, 1, 10, 1
 *GET, omega, MODE, i, FREQ
        ModalFreq(i,1) = omega
        *SET, omega,
*ENDDO

*SET, omega,
*CFOPEN, 'ModalFreq-%StrJobname(1)%', 'txt'
 *VWRITE, 'Num', 'omega'
  (8x, A8, 4X, A8,)
  *VWRITE, sequ, ModalFreq(1,1)
   (F12.0, F12.5)
*CFCLOS

FINISH
/EXIT, ALL
```

Listing 7.2: ShellModelShaft.txt

```
/COM,  ------------------------------------------------------------------------------
/COM,   PROBLEM: MODAL ANALYSIS of the helicopter's TAIL BOOM
/COM,  ------------------------------------------------------------------------------
```

```
/COM,   Natural frequencies and modal shapes of an helicopter's tail boom
/COM,  ---------------------------------------------------------------------------------

FINISH
/CLEAR, START, NEW
/FILNAM, ShellmodelShaft
/TITLE, Helicopter tail boom modal analysis shell model
/UNIT, SI
/INQUIRE, StrJobname, JOBNAME
*USE, testperson.mac

! >>>>> MODEL PARAMETERS <<<<<
*SET, Pi, ACOS(-1)               !Pi constant
*AFUN, DEG                       !Specify units for angular measures [DEG], specify after
     function *AFUN
*SET, eps, 10e-3                 !precision interval

! >>>>> MATERIAL PROPERTIES <<<<<
!*** steel
*SET, SteelEyounG, 205e9                          ![Pa] Young's modulus
*SET, SteelNi, 0.29                               !Poisson's ratio
*SET, SteelDensity, 7850                          ![Kg/m^3]

!*** Aluminium
*SET, AluminiumEyounG, 64e9                       ![Pa] Young's modulus
*SET, AluminiumNi, 0.34                           !Poisson's ratio
*SET, AluminiumDensity, 2700                      ![Kg/m^3]

!*** Duraluminium
*SET, DurAluminiumEyounG, 72e9                    ![Pa] Young's modulus
*SET, DurAluminiumNi, 0.33                        !Poisson's ratio
*SET, DurAluminiumDensity, 2810                   ![Kg/m^3]

!*** Element size
*SET, E_length_ct, 50e-3                          ![m]

!geometric parameter
*SET, Lt, 5.2                                     ![m] total tail lenght
*SET, RadBaseTail, 0.65/2                         ![m]
*SET, RadEndTail, 0.10/2                          ![m]
*SET, alpha, atan((RadBaseTail-RadEndTail)/Lt)    ![deg]

!geometric parameters section stiffner
*SET, StiffnerBase, 5e-3
*SET, StiffnerHeight, 5e-3

!geometric parameters section Horizzontal Stabilizer
*SET, HorizStabBase, 0.009
*SET, HorizStabHeigth, 0.0015

!geometric parameter mantle
*SET, ThichknessMantle, 5e-4                      ![m]

!geometric parameter shaft
*SET, InternalRadius, 0.012
*SET, ExternalRadius, 0.014
```

```

!mass Lumped
*SET, MassRotor, 30                          ![kg] concentrated mass motorblock
*SET, J0, 1                                  ![kg*m^2] concentrated inertia motorblock

!other parameters
*SET, NumberDivisonSurface, 24  !division cone's area

/COM, ---------------------------------------------
/COM,  MODEL DEFINITION
/COM, ---------------------------------------------
! >>>> DEFINE GEOMETRY <<<<
! Reduced integration Timoshenko beam element, with quadratic shape functions
/PREP7
! Define axis rotation
K, 1, 0, 0, 0
K, 2, 5, 0, 0

! create keypoints
K, 3, 0, RadBaseTail, 0                                          !start keypoint tails
*GET, MaxKp, KP, 0, NUM, MAX                                     !extract max keypoint
*SET, StartKpTail, MaxKp                                         !store in variable
*DIM, SegmentRadius, ARRAY, 7, 1                                 !store in array radius of segment distance between 0 - Y before rotating
*DO, i, 1, 7, 1
        *GET, MaxKp, KP, 0, NUM, MAX                             !extract max keypoint
        K, MaxKp+1, Lt/7*i, RadBaseTail-Lt/7*i*tan(alpha), 0     !create keypoint
        SegmentRadius(i) = RadBaseTail-Lt/7*i*tan(alpha)
*ENDDO
*GET, MaxKp, KP, 0, NUM, MAX                                     !extract max keypoint
*SET, EndKpTail, MaxKp                                           !store in variable end's tail
SAVE

! define keypoint mantle'tail
K, Maxkp+1, 0, RadBaseTail, 0
*GET, StartKpMantle, KP, 0, NUM, MAX
K, MaxKp+2, Lt/7*i, RadEndTail, 0
*GET, EndKpMantle, KP, 0, NUM, MAX
SAVE
*DO, j, StartKpTail, EndKpTail-1, 1
        L, j, j+1
*ENDDO
*GET, LenghtLineRference, LINE, 1, LENG                          !extract lenght segment line's cone generator

! generate cone
*GET, NumberLine, line, 0, COUNT                                 !count lines
*GET, StartNumberLine, LINE, 0, NUM, MIN                         !extract first number line
LSEL, S, LOC, X, 0, Lt
AROTAT, ALL, , , , , , 1, 2, 360, NumberDivisonSurface           !generate cone
SAVE

!divide line in array straight line and curved line
```

```
!to manage in next step meshing of elements
*GET, NumberLine, LINE, 0, COUNT                                !count lines
*DIM, StraighLine, ARRAY, NumberLine, 2
*DIM, ArchLine, ARRAY, NumberLine, 2
*DO, n, StartNumberLine, NumberLine, 1
        *GET, LenghtLine, LINE, n, leng,
        *IF, LenghtLineRference, EQ, LenghtLine, THEN
                StraighLine(n, 1) = n
                StraighLine(n, 2) = LenghtLine
                *SET, LastStraightLine, n
        *ELSE
                ArchLine(n, 1)=n
                ArchLine(n, 2)=LenghtLine
                *SET, LastArchLine, n
        *ENDIF
*ENDDO

*DIM, ConnectionKp, ARRAY, 8, 2
KSEL, S, LOC, X, 0, RadBaseTail
KSEL, S, LOC, Y, RadBaseTail*cos(360/NumberDivisonSurface), RadBaseTail*cos(360
    /NumberDivisonSurface)
KSEL, R, LOC, Z, -RadBaseTail, RadBaseTail
*GET, tmp1, KP, 0, NUM, MIN
*GET, tmp2, KP, 0, NUM, MAX
ConnectionKp(1, 1) =tmp1
ConnectionKp(1, 2) =tmp2
*DO, i, 1, 7, 1
        KSEL, S, LOC, X, Lt/7*i
        KSEL, R, LOC, Y, SegmentRadius(i)*cos(360/NumberDivisonSurface), SegmentRadius(i
            )*cos(360/NumberDivisonSurface)
        KSEL, R, LOC, Z, -SegmentRadius(i)*sin(360/NumberDivisonSurface), SegmentRadius(
            i)*sin(360/NumberDivisonSurface)
        *GET, tmp1, KP, 0, NUM, MIN
        *GET, tmp2, KP, 0, NUM, MAX
        ConnectionKp(i+1, 1) =tmp1
        ConnectionKp(i+1, 2) =tmp2
        !clear temp variable
        *SET, tmp1,
        *SET, tmp2,
*ENDDO
SAVE

/VIEW,1,1,1,1
/ANG,1
/REP,FAST
LPLOT
*USE, generateimages.mac

! >>>> MESH <<<<
/ESHAPE, 1
! set material properties and element
ET, 1, BEAM189
MP, EX, 1, SteelEyounG
MP, NUXY, 1, SteelNi
MP, DENS, 1, SteelDensity

```

```
!***Horizzontal STabilizer
SECTYPE,   1, BEAM, RECT, HorizzontalStabilizer, 0
SECOFFSET, CENT
SECDATA, HorizStabBase, HorizStabHeigth

LSEL, S, LINE, , 1, LastStraightLine
LESIZE, all, E_length_ct
LATT, 1, , 1, , , , 1
LMESH, ALL
LSEL, NONE
/REPLO

!***stiffners
SECTYPE,   2, BEAM, RECT, Stiffeners, 0
SECOFFSET, ORIG
SECDATA, StiffnerBase, StiffnerHeight

LSEL, S, LINE, , LastStraightLine+1, LastArchLine
LESIZE, ALL, , , , 3
LATT, 1, , 1, , , , 2
LMESH, ALL
LSEL, NONE
/REPLO

/VIEW, 1, 1, 1, 1
/ANG, 1
/REP, FAST
EPLOT
*USE, generateimages.mac

!***Mantle
! set material properties and element
ET, 2, SHELL181
KEYOPT, 2, 3, 2
KEYOPT, 2, 8, 1

MP, EX, 2, AluminiumEyounG
MP, NUXY, 2, AluminiumNi
MP, DENS, 2, AluminiumDensity
SECT, 3, SHELL, , mantle
SECDATA, ThichknessMantle, 2, 0.0, 3
SECOFFSET, TOP

TYPE, 2
SECNUM, 3
AESIZE, ALL, E_length_ct,
MSHAPE, 0, 2D
MSHKEY, 1
AMESH, ALL
SAVE

/VIEW, 1, 1, 1, 1
/ANG, 1
/REP, FAST
EPLOT
*USE, generateimages.mac
```

```

/COM,  -------------------------------
/COM,   DEFINITION SHAFT AND SUPPORT
/COM,  -------------------------------
*DIM, KpShaft, ARRAY, 8, 1
*GET, LastNode, KP, , num, max
K, LastNode+1, 0, RadBaseTail+50e-3, 0
*GET, LastNode, KP, , num, max
Kpshaft(1)=LastNode
*DO, i, 1, 7, 1
        *GET, LastNode, KP, , num, max
        K, LastNode+1, Lt/7*i, SegmentRadius(i)+50e-3, 0
        *GET, LastNode, KP, , num, max
        Kpshaft(i+1)=LastNode
        *GET, LastLine, LINE, , num, max
        L, LastNode-1, LastNode
*ENDDO

!***Shaft
MP, EX, 3, DurAluminiumEyounG
MP, NUXY, 3, DurAluminiumNi
MP, DENS, 3, DurAluminiumDensity
SECTYPE,   4, BEAM, CTUBE, Shaft, 0
SECOFFSET,  CENT
SECDATA,InternalRadius,ExternalRadius
LESIZE, ALL, E_length_ct
LATT, 3, , 1, , , , 4
LMESH, ALL
SAVE

!***Support shaft - rigid connection
ET, 3, MPC184, 1
KEYOPT, 3, 1, 1
MP, EX, 4, SteelEyounG
MP, NUXY, 4, SteelNi

SECNUM, 4
MAT, 4
TYPE, 3
*DIM, ConnectionShaftSupportNode, ARRAY, 8, 3
*DO, i, 1, 8, 1
        KSEL, s, KP, , KpShaft(i)
        NSLK, s
        *GET, NodeShaft, node, , num, max
        ConnectionShaftSupportNode(i, 1) = NodeShaft
        *SET, NodeShaft,
        KSEL, ALL
        KSEL, s, KP, , ConnectionKp(i, 1)
        NSLK, s
        *GET, NodeSupport1, node, , num, max
        ConnectionShaftSupportNode(i, 2) = NodeSupport1
        *SET, NodeSupport1,
        KSEL, ALL
        KSEL, s, KP, , ConnectionKp(i, 2)
        NSLK, s
        *GET, NodeSupport2, node, , num, max
```

```
        ConnectionShaftSupportNode(i, 3) = NodeSupport2
        *SET, NodeSupport2,
        E, ConnectionShaftSupportNode(i, 1), ConnectionShaftSupportNode(i, 2)
        E, ConnectionShaftSupportNode(i, 1), ConnectionShaftSupportNode(i, 3)
*ENDDO

!***gearbox lumped
ET, 5, MASS21, , , 3        !2D mass with rotary inertia
R, 5, MassRotor, J0
TYPE, 5
REAL, 5
E, ConnectionShaftSupportNode(7, 1)

EPLOT
/VIEW,1,,,1
/ANG,1
/REP,FAST
*USE, generateimages.mac
/VIEW,1,1,1,1
/ANG,1
/REP,FAST
*USE, generateimages.mac

!***Boundary conditions at base of tail
ALLSEL, ALL
LSEL, S, LOC, Y, -(RadBaseTail*2)+eps, (RadBaseTail*2)+eps
LSEL, S, LOC, x, 0, 0
*GET, MinBaseTailLine, line, 0, num, min
*GET, MaxBaseTailLine, line, 0, num, max
*DO, i, MinBaseTailLine, MaxBaseTailLine, 8
        DL, i, , ALL, 0
*ENDDO

*USE, calcmass.mac, StrJobname(1)
FINISH
SAVE

/COM, ------------------------------------------------
/COM, Preliminary static analysis - Prestress
/COM, ------------------------------------------------
! >>>> SOLUTION <<<<
/SOLU
ANTYPE, STATIC, NEW      !Preliminary static analysis
PSTRES, ON               !Includes prestress effects
Acel, , 9.81             !gravitational acceleration

SOLVE
FINISH

! >>>> POSTPROCESSING <<<<
/POST1
PLDISP, 1
*use, generateimages.mac
PLNSOL, U, SUM, 0, 1.0
*use, generateimages.mac
PLNSOL, S, EQV, 0, 1.0
```

```
*use, generateimages.mac

/COM, ------------------------------------------------
/COM,  Modal analysis
/COM, ------------------------------------------------
! >>>> SOLUTION <<<<
/SOLU
ANTYPE, MODAL, NEW
PSTRES, ON                              ! Includes prestress effects
MODOPT, LANB, 10                        ! First 10 modes are extracted
LUMPM, OFF                              ! lumped/consistent mass matrix
MXPAND, 10, , , Yes                     ! First 10 modes are expanded, calculate element
     stress

SOLVE
FINISH

! >>>> POSTPROCESSING <<<<
/POST1
SET, LIST                       !List natural frequencies
SET, 1, 1                       !Loads first mode
PLDISP                          !Displays modal shape

*DIM,ModalFreq,ARRAY,10,1
*Do, i, 1, 10, 1
 *GET, omega, MODE, i, FREQ
        ModalFreq(i,1) = omega
        *SET, omega,
*ENDDO

*SET, omega,
*CFOPEN, 'ModalFreq-%StrJobname(1)%', 'txt'
 *VWRITE, 'Num', 'omega'
  (8x, A8, 4X, A8,)
  *VWRITE, sequ, ModalFreq(1,1)
   (F12.0, F12.5)
*CFCLOS

FINISH
/EXIT, ALL
```

Listing 7.3: ShellModelShaftLumped.txt

```
/COM,  --------------------------------------------------------------------------------
/COM,   PROBLEM: MODAL ANALYSIS of the helicopter's TAIL BOOM
/COM,  --------------------------------------------------------------------------------
/COM,   Natural frequencies and modal shapes of an helicopter's tail boom
/COM,  --------------------------------------------------------------------------------

FINISH
/CLEAR, START, NEW
/FILNAM, ShellmodelShaftLumped
/TITLE, Helicopter tail boom modal analysis shell model
/UNIT, SI
/INQUIRE, StrJobname, JOBNAME
*USE, testperson.mac
```

```
! >>>>> MODEL PARAMETERS <<<<<
*SET, Pi, ACOS(-1)              !Pi constant
*AFUN, DEG                      !Specify units for angular measures [DEG], specify after
     function *AFUN
*SET, eps, 10e-3                !precision interval

! >>>>> MATERIAL PROPERTIES <<<<<
!*** steel
*SET, SteelEyounG, 205e9                        ![Pa] Young's modulus
*SET, SteelNi, 0.29                             !Poisson's ratio
*SET, SteelDensity, 7850                        ![Kg/m^3]

!*** Aluminium
*SET, AluminiumEyounG, 64e9                     ![Pa] Young's modulus
*SET, AluminiumNi, 0.34                         !Poisson's ratio
*SET, AluminiumDensity, 2700                    ![Kg/m^3]

!*** Element size
*SET, E_length_ct, 50e-3                        ![m]

!geometric parameter
*SET, Lt, 5.2                                   ![m] total tail lenght
*SET, RadBaseTail, 0.65/2                       ![m]
*SET, RadEndTail, 0.10/2                        ![m]
*SET, alpha, atan((RadBaseTail-RadEndTail)/Lt)  ![deg]

!geometric parameters section stiffner
*SET, StiffnerBase, 5e-3
*SET, StiffnerHeight, 5e-3

!geometric parameters section Horizzontal Stabilizer
*SET, HorizStabBase, 0.009
*SET, HorizStabHeigth, 0.0015

!geometric parameter mantle
*SET, ThichknessMantle, 5e-4                    ![m]

!mass Lumped
*SET, MassShaft, 7                              ![kg] concentrated mass shaft
*SET, MassRotor, 30                             ![kg] concentrated mass motorblock
*SET, J0, 1                                     ![kg*m^2] concentrated inertia
    motorblock

!other parameters
*SET, NumberDivisonSurface, 24  !division cone's area

/COM, -----------------------------------------------
/COM,  MODEL DEFINITION
/COM, -----------------------------------------------
! >>>> DEFINE GEOMETRY <<<<
! Reduced integration Timoshenko beam element, with quadratic shape functions
/PREP7
! Define axis rotation
K, 1, 0, 0, 0
K, 2, 5, 0, 0
```

```

! create keypoints
K, 3, 0, RadBaseTail, 0                                             !start keypoint tails
*GET, MaxKp, KP, 0, num, max                                        !extract max keypoint
*SET, StartKpTail, MaxKp                                            !store in variable
*DIM, SegmentRadius, ARRAY, 7, 1                                    !store in array radius of segment distance between 0 - Y before rotating
*DO, i, 1, 7, 1
        *GET, MaxKp, KP, 0, num, max                                !extract max keypoint
        K, MaxKp+1, Lt/7*i, RadBaseTail-Lt/7*i*tan(alpha), 0        !create keypoint
        SegmentRadius(i) = RadBaseTail-Lt/7*i*tan(alpha)
*ENDDO
*GET, MaxKp, KP, 0, num, max                                        !extract max keypoint
*SET, EndKpTail, MaxKp                                              !store in variable end's tail
SAVE

! define keypoint mantle'tail
K, Maxkp+1, 0, RadBaseTail, 0
*GET, StartKpMantle, KP, 0, num, max
K, MaxKp+2, Lt/7*i, RadEndTail, 0
*GET, EndKpMantle, KP, 0, num, max
SAVE
*DO, j, StartKpTail, EndKpTail-1, 1
        L, j, j+1
*ENDDO
*GET, LenghtLineRference, LINE, 1, leng                             !extract lenght segment line's cone generator

! generate cone
*GET, NumberLine, line, 0, count                                    !count lines
*GET, StartNumberLine, line, 0, num, min                            !extract first number line
LSEL, S, LOC, X, 0, Lt
AROTAT, ALL, , , , , , 1, 2, 360, NumberDivisonSurface              !generate cone
SAVE

!divide line in array straight line and curved line
!to manage in next step meshing of elements
*GET, NumberLine, line, 0, count                                    !count lines
*DIM, StraighLine, ARRAY, NumberLine, 2
*DIM, ArchLine, ARRAY, NumberLine, 2
*DO, n, StartNumberLine, NumberLine, 1
        *GET, LenghtLine, LINE, n, leng,
        *IF, LenghtLineRference, EQ, LenghtLine, THEN
                StraighLine(n, 1) = n
                StraighLine(n, 2) = LenghtLine
                *SET, LastStraightLine, n
        *ELSE
                ArchLine(n, 1)=n
                ArchLine(n, 2)=LenghtLine
                *SET, LastArchLine, n
        *ENDIF
*ENDDO

*DIM, ConnectionKp, ARRAY, 8, 2
```

```
ksel, s, loc, x, 0, RadBaseTail
Ksel, s, loc, y, RadBaseTail*cos(360/NumberDivisonSurface), RadBaseTail*cos(360/NumberDivisonSurface)
Ksel, r, loc, z, -RadBaseTail, RadBaseTail
*GET, tmp1, KP, 0, num, min
*GET, tmp2, KP, 0, num, max
ConnectionKp(1, 1) =tmp1
ConnectionKp(1, 2) =tmp2
*DO, i, 1, 7, 1
        ksel, s, loc, x, Lt/7*i
        Ksel, r, loc, y, SegmentRadius(i)*cos(360/NumberDivisonSurface), SegmentRadius(i)*cos(360/NumberDivisonSurface)
        ksel, r, loc, z, -SegmentRadius(i)*sin(360/NumberDivisonSurface), SegmentRadius(i)*sin(360/NumberDivisonSurface)
        *GET, tmp1, KP, 0, num, min
        *GET, tmp2, KP, 0, num, max
        ConnectionKp(i+1, 1) =tmp1
        ConnectionKp(i+1, 2) =tmp2
        !clear temp variable
        *SET, tmp1,
        *SET, tmp2,
*ENDDO
SAVE

/VIEW,1,1,1,1
/ANG,1
/REP,FAST
LPLOT
*USE, generateimages.mac

! >>>> MESH <<<<
/ESHAPE, 1
! set material properties and element
ET, 1, BEAM189
MP, EX, 1, SteelEyounG
MP, NUXY, 1, SteelNi
MP, DENS, 1, SteelDensity

!***Horizzontal STabilizer
SECTYPE,   1, BEAM, RECT, HorizzontalStabilizer, 0
SECOFFSET, CENT
SECDATA, HorizStabBase, HorizStabHeigth

LSEL, S, LINE, , 1, LastStraightLine
LESIZE, all, E_length_ct
LATT, 1, , 1, , , , 1
LMESH, ALL
LSEL, NONE
/REPLO

!***stiffners
SECTYPE,   2, BEAM, RECT, Stiffeners, 0
SECOFFSET, ORIG
SECDATA, StiffnerBase, StiffnerHeight

LSEL, S, LINE, , LastStraightLine+1, LastArchLine
```

```
LESIZE, ALL, , , , 3
LATT, 1, , 1, , , , 2
LMESH, ALL
LSEL, NONE
/REPLO

/VIEW, 1, 1, 1, 1
/ANG, 1
/REP, FAST
EPLOT
*USE, generateimages.mac, StrJobname(1)

!***Mantle
! set material properties and element
ET, 2, SHELL181
KEYOPT, 2, 3, 2
KEYOPT, 2, 8, 1

MP, EX, 2, AluminiumEyounG
MP, NUXY, 2, AluminiumNi
MP, DENS, 2, AluminiumDensity
SECT, 3, SHELL, , mantle
SECDATA, ThichknessMantle, 2, 0.0, 3
SECOFFSET, TOP

TYPE, 2
SECNUM, 3
AESIZE, ALL, E_length_ct,
MSHAPE, 0, 2D
MSHKEY, 1
AMESH, ALL
SAVE

/VIEW, 1, 1, 1, 1
/ANG, 1
/REP, FAST
EPLOT
*USE, generateimages.mac

!define shaft and support
*DIM, KpShaft, ARRAY, 8, 1
*GET, LastNode, KP, , num, max
K, LastNode+1, 0, RadBaseTail+50e-3, 0
*GET, LastNode, KP, , num, max
Kpshaft(1)=LastNode
*DO, i, 1, 7, 1
        *GET, LastNode, KP, , num, max
        K, LastNode+1, Lt/7*i, SegmentRadius(i)+50e-3, 0
        *GET, LastNode, KP, , num, max
        Kpshaft(i+1)=LastNode
        *GET, LastLine, LINE, , num, max
        L, LastNode-1, LastNode
*ENDDO
SAVE

!***Shaft lumped mass
```

```
*DIM, ConnectionShaftSupportNode, ARRAY, 8, 3
*GET, KpShaftDimension,'PARM',KPSHAFT,DIM,X
ET, 4, MASS21, , ,2
R, 4, MassShaft/8
TYPE, 4
REAL, 4
*DO, i, 1, KpShaftDimension, 1
        *GET,LastNode,NODE,,NUM,MAX
        N, LastNode+1, KX(Kpshaft(i)), KY(Kpshaft(i)), KZ(Kpshaft(i))
        *GET,node_numb,NODE,,NUM,MAX
        ConnectionShaftSupportNode(i, 1) = node_numb
        E, node_numb
*ENDDO
SAVE

!***gearbox lumped
ET, 5, MASS21, , , 3        !2D mass with rotary inertia
R, 5, MassRotor, J0
TYPE, 5
REAL, 5
E, ConnectionShaftSupportNode(7, 1)

!***Support shaft - rigid connection
ET, 3, MPC184, 1
KEYOPT, 3, 1, 1
MP, EX, 3, SteelEyounG
MP, NUXY, 3, SteelNi

SECNUM, 4
MAT, 3
TYPE, 3
*DO, i, 1, 8, 1
        KSEL, s, KP, , KpShaft(i)
        NSLK, s
        *GET, NodeShaft, node, , num, max
        *SET, NodeShaft,
        KSEL, ALL
        KSEL, s, KP, , ConnectionKp(i, 1)
        NSLK, s
        *GET, NodeSupport1, node, , num, max
        ConnectionShaftSupportNode(i, 2) = NodeSupport1
        *SET, NodeSupport1,
        KSEL, ALL
        KSEL, s, KP, , ConnectionKp(i, 2)
        NSLK, s
        *GET, NodeSupport2, node, , num, max
        ConnectionShaftSupportNode(i, 3) = NodeSupport2
        *SET, NodeSupport2,
        E, ConnectionShaftSupportNode(i, 1), ConnectionShaftSupportNode(i, 2)
        E, ConnectionShaftSupportNode(i, 1), ConnectionShaftSupportNode(i, 3)
*ENDDO
SAVE

EPLOT
/VIEW, 1, , , 1
/ANG, 1
```

```
/REP, FAST
*USE, generateimages.mac
/VIEW, 1, 1, 1, 1
/ANG, 1
/REP, FAST
*USE, generateimages.mac

!***Boundary conditions at base of tail
ALLSEL, ALL
LSEL, S, LOC, Y, -(RadBaseTail*2)+eps, (RadBaseTail*2)+eps
LSEL, S, LOC, x, 0, 0
*GET, MinBaseTailLine, line, 0, num, min
*GET, MaxBaseTailLine, line, 0, num, max
*DO, i, MinBaseTailLine, MaxBaseTailLine, 8
        DL, i, , ALL, 0
*ENDDO

*USE, calcmass.mac, StrJobname(1)
FINISH

/COM, ------------------------------------------------
/COM,  Preliminary static analysis - Prestress
/COM, ------------------------------------------------
! >>>> SOLUTION <<<<
/SOLU
ANTYPE, STATIC, NEW     !Preliminary static analysis
PSTRES, ON              !Includes prestress effects
LUMPM, ON
Acel, , 9.81            !gravitational acceleration

SOLVE
FINISH

! >>>> POSTPROCESSING <<<<
/POST1
PLDISP, 1
*use, generateimages.mac
PLNSOL,  U, SUM,  0, 1.0
*use, generateimages.mac
PLNSOL,  S, EQV,  0, 1.0
*use, generateimages.mac

/COM, ------------------------------------------------
/COM,  Modal analysis
/COM, ------------------------------------------------
! >>>> SOLUTION <<<<
/SOLU
ANTYPE, MODAL, NEW
PSTRES, ON                         ! Includes prestress effects
MODOPT, LANB, 10                   ! First 10 modes are extracted
LUMPM, ON                          ! lumped/consistent mass matrix
MXPAND, 10, , , Yes                ! First 10 modes are expanded, calculate element
     stress

SOLVE
FINISH
```

```

! >>>> POSTPROCESSING <<<<
/POST1
SET, LIST                      !List natural frequencies
SET, 1, 1                      !Loads first mode
PLDISP                         !Displays modal shape

*DIM,ModalFreq,ARRAY,10,1
*Do, i, 1, 10, 1
 *GET, omega, MODE, i, FREQ
        ModalFreq(i,1) = omega
        *SET, omega,
*ENDDO

*SET, omega,
*CFOPEN, 'ModalFreq-%StrJobname(1)%', 'txt'
 *VWRITE, 'Num', 'omega'
  (8x, A8, 4X, A8,)
  *VWRITE, sequ, ModalFreq(1,1)
   (F12.0, F12.5)
*CFCLOS

FINISH
/EXIT, ALL
```

Listing 7.4: ShellModelRotor.txt

```
/COM, ---------------------------------------------------------------------------------
/COM,  PROBLEM: MODAL ANALYSIS of the helicopter's TAIL BOOM
/COM, ---------------------------------------------------------------------------------
/COM,  Natural frequencies and modal shapes of an helicopter's tail boom
/COM, ---------------------------------------------------------------------------------

FINISH
/CLEAR, START, NEW
/FILNAM, ShellmodelRotor
/TITLE, Helicopter tail boom modal analysis shell model
/UNIT, SI
/INQUIRE, StrJobname, JOBNAME
*USE, testperson.mac

! >>>>> MODEL PARAMETERS <<<<<
*SET, Pi, ACOS(-1)             !Pi constant
*AFUN, DEG                     !Specify units for angular measures [DEG], specify after
     function *AFUN
*SET, eps, 10e-3               !precision interval

! >>>>> MATERIAL PROPERTIES <<<<<
!*** steel
*SET, SteelEyounG, 205e9                        ![Pa] Young's modulus
*SET, SteelNi, 0.29                             !Poisson's ratio
*SET, SteelDensity, 7850                        ![Kg/m^3]

!*** Aluminium
*SET, AluminiumEyounG, 64e9                     ![Pa] Young's modulus
*SET, AluminiumNi, 0.34                         !Poisson's ratio
```

```
*SET, AluminiumDensity, 2700                    ![Kg/m^3]

!*** Element size
*SET, E_length_ct, 50e-3                        ![m]

!geometric parameter
*SET, Lt, 5.2                                   ![m] total tail lenght
*SET, RadBaseTail, 0.65/2                       ![m]
*SET, RadEndTail, 0.10/2                        ![m]
*SET, alpha, atan((RadBaseTail-RadEndTail)/Lt)  ![deg]

!geometric parameters section stiffner
*SET, StiffnerBase, 5e-3
*SET, StiffnerHeight, 5e-3

!geometric parameters section Horizzontal Stabilizer
*SET, HorizStabBase, 0.009
*SET, HorizStabHeigth, 0.0015

!geometric parameter mantle
*SET, ThichknessMantle, 5e-4                    ![m]

!other parameters
*SET, NumberDivisonSurface, 24  !division cone's area

!change view - isometric
/VIEW,1,1,1,1
/ANG,1
/REP,FAST

/COM, -----------------------------------------------
/COM,  MODEL DEFINITION
/COM, -----------------------------------------------
! >>>> DEFINE GEOMETRY <<<<
! Reduced integration Timoshenko beam element, with quadratic shape functions
/PREP7
! Define axis rotation
K, 1, 0, 0, 0
K, 2, 5, 0, 0

! create keypoints
K, 3, 0, RadBaseTail, 0                                         !start keypoint
    tails
*GET, MaxKp, KP, 0, num, max                                    !extract max
    keypoint
*SET, StartKpTail, MaxKp                                        !store in
    variable
*DIM, SegmentRadius, ARRAY, 7, 1                        !store in array radius
    of segment distance between 0 - Y before rotating
*DO, i, 1, 7, 1
        *GET, MaxKp, KP, 0, NUM, MAX                    !extract max keypoint
        K, MaxKp+1, Lt/7*i, RadBaseTail-Lt/7*i*tan(alpha), 0    !create keypoint
        SegmentRadius(i) = RadBaseTail-Lt/7*i*tan(alpha)
*ENDDO
*GET, MaxKp, KP, 0, num, max                                    !extract max
    keypoint
```

```
*SET, EndKpTail, MaxKp                                            !store in
    variable end's tail

!define keypoint mantle'tail
K, Maxkp+1, 0, RadBaseTail, 0
*GET, StartKpMantle, KP, 0, num, max
K, MaxKp+2, Lt/7*i, RadEndTail, 0
*GET, EndKpMantle, KP, 0, num, max

*DO, j, StartKpTail, EndKpTail-1, 1
        L, j, j+1
*ENDDO
*GET, LenghtLineRference, LINE, 1, leng                           !extract lenght
    segment line's cone generator

! generate cone
*GET, NumberLine, line, 0, count                                  !count lines
*GET, StartNumberLine, line, 0, num, min                          !extract first
    number line
*GET, NumberLine, line, 0, count                                  !count lines
*GET, StartNumberLine, line, 0, num, min                          !extract first
    number line
LSEL, S, LOC, X, 0, Lt
AROTAT, ALL, , , , , , 1, 2, 360, NumberDivisonSurface            !generate cone

!divide line in array straight line and curved line
!to manage in next step meshing of elements
LSEL, S, LENGTH, , LenghtLineRference, LenghtLineRference
*VGET, StraighLine, LINE, , LLIST, , , 0
ALLSEL, ALL
LSEL, U, LENGTH, , LenghtLineRference, LenghtLineRference
*VGET, AcrLine, LINE, , LLIST, , , 0

*GET, StartNumberLine, line, 0, num, min
LSEL, ALL

! >>>> MESH <<<<
/ESHAPE, 1
!set material properties and element
ET, 1, BEAM189
KEYOPT, 1, 1, 1
KEYOPT, 1, 15, 1
MP, EX, 1, SteelEyounG
MP, NUXY, 1, SteelNi
MP, DENS, 1, SteelDensity

!***Horizzontal Stabilizer
SECTYPE,  1, BEAM, RECT, HorizzontalStabilizer, 0
SECOFFSET, CENT
SECDATA, HorizStabBase, HorizStabHeigth

*GET, LenghtStraighLine, 'PARM', StraighLine, DIM, X
LSEL, S, LINE, , StraighLine(1), StraighLine(LenghtStraighLine)
LESIZE, all, E_length_ct
LATT, 1, , 1, , , , 1
LMESH, ALL
```

```
LSEL, NONE
/ESHAPE, 1
/REPLO

!***stiffners
SECTYPE,  2, BEAM, RECT, Stiffeners, 0
SECOFFSET, ORIG
SECDATA, StiffnerBase, StiffnerHeight

*GET, LenghtAcrLine, 'PARM', AcrLine, DIM, X
LSEL, S, LINE, , AcrLine(1), AcrLine(LenghtAcrLine)
LESIZE, ALL, , , , 3
LATT, 1, , 1, , , , 2
LMESH, ALL
LSEL, NONE
/REPLO

!***Mantle
!set material properties and element
ET, 2, SHELL181
KEYOPT, 2, 3, 2
KEYOPT, 2, 8, 2

MP, EX, 2, AluminiumEyounG
MP, NUXY, 2, AluminiumNi
MP, DENS, 2, AluminiumDensity
SECT, 3, SHELL, , mantle
SECDATA, ThichknessMantle, 2, 0.0, 3
SECOFFSET, top

TYPE, 2
SECNUM, 3
AESIZE, ALL, E_length_ct,
MSHAPE, 0, 2D
MSHKEY, 1
AMESH, ALL
SAVE

/COM, ------------------------------------------------
/COM,  COMPLETE MODEL
/COM, ------------------------------------------------

!Selct base keypoint and node
KSEL, S, LOC, X, 4.457143
KSEL, R, LOC, Y, 0.4464286E-01
KSEL, R, LOC, Z, -0.7732370E-01, 0.7732370E-01
*GET, BaseBearing1, KP, , NUM, MIN
*GET, BaseBearing2, KP, , NUM, MAX
ALLSEL,ALL

!create node for rigid link
*GET, LastNode, NODE, , NUM, MAX
N, LastNode + 1, kx(BaseBearing2), SegmentRadius(6)+50e-3, kz(BaseBearing2)
*GET, NodeBearing2, NODE, , NUM, MAX
*GET, LastNode, NODE, , NUM, MAX
N, LastNode + 1, kx(BaseBearing1), SegmentRadius(6)+50e-3, kz(BaseBearing1)
```

```
*GET, NodeBearing1, NODE, , NUM, MAX
*USE, distancenode.mac, NodeBearing1, NodeBearing2

!Set local systems
*USE, relativereferencesystem.mac, 1, 0, Lt/7*6, SegmentRadius(6)+50e-3, kz(BaseBearing1
    ), , , 90
*USE, relativereferencesystem.mac, 2, 0, Lt/7*6, SegmentRadius(6)+50e-3, kz(BaseBearing2
    ), , , 90

!Enter local system - z-positive
CSYS, LocalSystem_1

!create rotorblock
*USE, rotor.mac, NodeBearing1, NodeBearing2, DistancePoint, LocalSystem_1, LocalSystem_2

!return principal Reference system
CSYS, 0
SAVE

/COM, -------------------------------------------------
/COM,  RIGID LINK
/COM,  create revolute joint to connect shfat and tail
/COM, -------------------------------------------------
!define element rigid link
*GET, LastElementType, ETYP, , NUM, MAX
*SET, RigidLink, LastElementType + 1
ET, RigidLink, MPC184
!0 - rigid link block 3 d.o.f (x, y, z)
!1 - beam block 6 d.o.f (x, y, z, rx, ry, rz)
KEYOPT, RigidLink, 1, 1
KEYOPT, RigidLink, 2, 0

!material for rigid link
*GET, MaterialNum, MAT, , NUM, MAX
*SET, RigidLinkMaterial, MaterialNum + 1
!MP, EX, RigidLinkMaterial, SteelEyounG
!MP, NUXY, RigidLinkMaterial, SteelNi
!MAT, RigidLinkMaterial

!create element
TYPE, RigidLink
KSEL,S,KP,,BaseBearing1
NSLK,S
*GET, LastNode, NODE, , NUM, MAX
E, LastNode, NodeStart
ALLSEL,ALL
KSEL,S,KP,,BaseBearing2
NSLK,S
*GET, LastNode, NODE, , NUM, MAX
E, LastNode, NodeEnd

!clear parmaters
*SET, BaseBearing1,
*SET, BaseBearing2,
*SET, NodeBearing1,
*SET, NodeBearing2,
```

```
*SET, LastNode,
*SET, i,

!change view - isometric
EPLOT
/VIEW,1,1,1,1
/ANG,1
/REP,FAST

/COM, ------------------------------------------------
/COM,  BOUNDARY CONDITION
/COM, ------------------------------------------------
ALLSEL, ALL
LSEL, S, LOC, Y, -(RadBaseTail*2)+eps, (RadBaseTail*2)+eps
LSEL, S, LOC, x, 0, 0
*GET, MinBaseTailLine, line, 0, num, min
*GET, MaxBaseTailLine, line, 0, num, max
*DO, i, MinBaseTailLine, MaxBaseTailLine, 8
        DL, i, , ALL, 0
*ENDDO
SAVE

!calc mass
*USE, calcmass.mac, StrJobname(1)
FINISH
SAVE

/COM, ------------------------------------------------
/COM,  PRELIMINARY STATIC ANALYSIS - PRESTRESS
/COM, ------------------------------------------------
! >>>> SOLUTION <<<<
/SOLU
ANTYPE, STATIC, NEW     !Preliminary static analysis
PSTRES, ON              !Includes prestress effects
Acel, , 9.81            !gravitational acceleration

SOLVE
FINISH

/COM, ------------------------------------------------
/COM,  Modal analysis
/COM, ------------------------------------------------
! >>>> SOLUTION <<<<
/SOLU
ANTYPE, MODAL, NEW
PSTRES, ON                              ! Includes prestress effects
MODOPT, LANB, 10                        ! First 10 modes are extracted
LUMPM, OFF                              ! lumped/consistent mass matrix
MXPAND, 10, , , Yes                     ! First 10 modes are expanded, calculate element stress

SOLVE
FINISH

! >>>> POSTPROCESSING <<<<
/POST1
```

```
SET, LIST                        !List natural frequencies
SET, 1, 1                        !Loads first mode
PLDISP                           !Displays modal shape

*DIM,ModalFreq,ARRAY,10,1
*Do, i, 1, 10, 1
 *GET, omega, MODE, i, FREQ
        ModalFreq(i,1) = omega
        *SET, omega,
*ENDDO

*SET, omega,
*CFOPEN, 'ModalFreq-%StrJobname(1)%', 'txt'
 *VWRITE, 'Num', 'omega'
  (8x, A8, 4X, A8,)
  *VWRITE, sequ, ModalFreq(1,1)
   (F12.0, F12.5)
*CFCLOS

FINISH
/EXIT, ALL
```

Listing 7.5: `ShellModeldiskRun.txt`

```
/COM,  ---------------------------------------------------------------------------------
/COM,   PROBLEM: MODAL ANALYSIS of the helicopter's TAIL BOOM
/COM,  ---------------------------------------------------------------------------------
/COM,   Natural frequencies and modal shapes of an helicopter's tail boom
/COM,  ---------------------------------------------------------------------------------

FINISH
/CLEAR, START, NEW
/FILNAM, ShellmodelRotor
/TITLE, Helicopter tail boom modal analysis shell model
/UNIT, SI
/INQUIRE, StrJobname, JOBNAME
*USE, testperson.mac

! >>>>> MODEL PARAMETERS <<<<<
*SET, Pi, ACOS(-1)               !Pi constant
*AFUN, DEG                       !Specify units for angular measures [DEG], specify after
     function *AFUN
*SET, eps, 10e-3                 !precision interval

! >>>>> MATERIAL PROPERTIES <<<<<
!*** steel
*SET, SteelEyounG, 205e9                        ![Pa] Young's modulus
*SET, SteelNi, 0.29                             !Poisson's ratio
*SET, SteelDensity, 7850                        ![Kg/m^3]

!*** Aluminium
*SET, AluminiumEyounG, 64e9                     ![Pa] Young's modulus
*SET, AluminiumNi, 0.34                         !Poisson's ratio
*SET, AluminiumDensity, 2700                    ![Kg/m^3]

!*** Element size
```

```
*SET, E_length_ct, 50e-3                          ![m]

!geometric parameter
*SET, Lt, 5.2                                     ![m] total tail lenght
*SET, RadBaseTail, 0.65/2                         ![m]
*SET, RadEndTail, 0.10/2                          ![m]
*SET, alpha, atan((RadBaseTail-RadEndTail)/Lt)  ![deg]

!geometric parameters section stiffner
*SET, StiffnerBase, 5e-3
*SET, StiffnerHeight, 5e-3

!geometric parameters section Horizzontal Stabilizer
*SET, HorizStabBase, 0.009
*SET, HorizStabHeigth, 0.0015

!geometric parameter mantle
*SET, ThichknessMantle, 5e-4                      ![m]

!other parameters
*SET, NumberDivisonSurface, 24  !division cone's area

!change view - isometric
/VIEW,1,1,1,1
/ANG,1
/REP,FAST

/COM, ------------------------------------------------
/COM,  MODEL DEFINITION
/COM, ------------------------------------------------
! >>>> DEFINE GEOMETRY <<<<
! Reduced integration Timoshenko beam element, with quadratic shape functions
/PREP7
! Define axis rotation
K, 1, 0, 0, 0
K, 2, 5, 0, 0

! create keypoints
K, 3, 0, RadBaseTail, 0                                               !start keypoint tails
*GET, MaxKp, KP, 0, num, max                                          !extract max keypoint
*SET, StartKpTail, MaxKp                                              !store in variable
*DIM, SegmentRadius, ARRAY, 7, 1                              !store in array radius of segment distance between 0 - Y before rotating
*DO, i, 1, 7, 1
        *GET, MaxKp, KP, 0, NUM, MAX                          !extract max keypoint
        K, MaxKp+1, Lt/7*i, RadBaseTail-Lt/7*i*tan(alpha), 0     !create keypoint
        SegmentRadius(i) = RadBaseTail-Lt/7*i*tan(alpha)
*ENDDO
*GET, MaxKp, KP, 0, num, max                                          !extract max keypoint
*SET, EndKpTail, MaxKp                                                !store in variable end's tail

```

```
!define keypoint mantle'tail
K, Maxkp+1, 0, RadBaseTail, 0
*GET, StartKpMantle, KP, 0, num, max
K, MaxKp+2, Lt/7*i, RadEndTail, 0
*GET, EndKpMantle, KP, 0, num, max

*DO, j, StartKpTail, EndKpTail-1, 1
        L, j, j+1
*ENDDO
*GET, LenghtLineRference, LINE, 1, leng                                !extract lenght
    segment line's cone generator

! generate cone
*GET, NumberLine, line, 0, count                                       !count lines
*GET, StartNumberLine, line, 0, num, min                               !extract first
    number line
*GET, NumberLine, line, 0, count                                       !count lines
*GET, StartNumberLine, line, 0, num, min                               !extract first
    number line
LSEL, S, LOC, X, 0, Lt
AROTAT, ALL, , , , , , 1, 2, 360, NumberDivisonSurface                 !generate cone

!divide line in array straight line and curved line
!to manage in next step meshing of elements
LSEL, S, LENGTH, , LenghtLineRference, LenghtLineRference
*VGET, StraighLine, LINE, , LLIST, , , 0
ALLSEL, ALL
LSEL, U, LENGTH, , LenghtLineRference, LenghtLineRference
*VGET, AcrLine, LINE, , LLIST, , , 0

*GET, StartNumberLine, line, 0, num, min
LSEL, ALL

! >>>> MESH <<<<
/ESHAPE, 1
!set material properties and element
ET, 1, BEAM189
KEYOPT, 1, 1, 1
KEYOPT, 1, 15, 1
MP, EX, 1, SteelEyounG
MP, NUXY, 1, SteelNi
MP, DENS, 1, SteelDensity

!***Horizzontal Stabilizer
SECTYPE,  1, BEAM, RECT, HorizzontalStabilizer, 0
SECOFFSET, CENT
SECDATA, HorizStabBase, HorizStabHeigth

*GET, LenghtStraighLine, 'PARM', StraighLine, DIM, X
LSEL, S, LINE, , StraighLine(1), StraighLine(LenghtStraighLine)
LESIZE, all, E_length_ct
LATT, 1, , 1, , , , 1
LMESH, ALL
LSEL, NONE
/ESHAPE, 1
/REPLO
```

```
!***stiffners
SECTYPE,  2, BEAM, RECT, Stiffeners, 0
SECOFFSET, ORIG
SECDATA, StiffnerBase, StiffnerHeight

*GET, LenghtAcrLine, 'PARM', AcrLine, DIM, X
LSEL, S, LINE, , AcrLine(1), AcrLine(LenghtAcrLine)
LESIZE, ALL, , , , 3
LATT, 1, , 1, , , , 2
LMESH, ALL
LSEL, NONE
/REPLO

!***Mantle
!set material properties and element
ET, 2, SHELL181
KEYOPT, 2, 3, 2
KEYOPT, 2, 8, 2

MP, EX, 2, AluminiumEyounG
MP, NUXY, 2, AluminiumNi
MP, DENS, 2, AluminiumDensity
SECT, 3, SHELL, , mantle
SECDATA, ThichknessMantle, 2, 0.0, 3
SECOFFSET, top

TYPE, 2
SECNUM, 3
AESIZE, ALL, E_length_ct,
MSHAPE, 0, 2D
MSHKEY, 1
AMESH, ALL
SAVE

/COM, -------------------------------------------------
/COM,  COMPLETE MODEL
/COM, -------------------------------------------------

!Selct base keypoint and node
KSEL, S, LOC, X, 4.457143
KSEL, R, LOC, Y, 0.4464286E-01
KSEL, R, LOC, Z, -0.7732370E-01, 0.7732370E-01
*GET, BaseBearing1, KP, , NUM, MIN
*GET, BaseBearing2, KP, , NUM, MAX
ALLSEL,ALL

!create node for rigid link
*GET, LastNode, NODE, , NUM, MAX
N, LastNode + 1, kx(BaseBearing2), SegmentRadius(6)+50e-3, kz(BaseBearing2)
*GET, NodeBearing2, NODE, , NUM, MAX
*GET, LastNode, NODE, , NUM, MAX
N, LastNode + 1, kx(BaseBearing1), SegmentRadius(6)+50e-3, kz(BaseBearing1)
*GET, NodeBearing1, NODE, , NUM, MAX
*USE, distancenode.mac, NodeBearing1, NodeBearing2
```

```
!Set local systems
*USE, relativereferencesystem.mac, 1, 0, Lt/7*6, SegmentRadius(6)+50e-3, kz(BaseBearing1
    ), , , 90
*USE, relativereferencesystem.mac, 2, 0, Lt/7*6, SegmentRadius(6)+50e-3, kz(BaseBearing2
    ), , , 90

!Enter local system - z-positive
CSYS, LocalSystem_1

!create rotorblock
*USE, rotor.mac, NodeBearing1, NodeBearing2, DistancePoint, LocalSystem_1, LocalSystem_2

!return principal Reference system
CSYS, 0
SAVE

/COM, ------------------------------------------------
/COM,  RIGID LINK
/COM,  create revolute joint to connect shfat and tail
/COM, ------------------------------------------------
!define element rigid link
*GET, LastElementType, ETYP, , NUM, MAX
*SET, RigidLink, LastElementType + 1
ET, RigidLink, MPC184
!0 - rigid link block 3 d.o.f (x, y, z)
!1 - beam block 6 d.o.f (x, y, z, rx, ry, rz)
KEYOPT, RigidLink, 1, 1
KEYOPT, RigidLink, 2, 0

!material for rigid link
*GET, MaterialNum, MAT, , NUM, MAX
*SET, RigidLinkMaterial, MaterialNum + 1
!MP, EX, RigidLinkMaterial, SteelEyoungG
!MP, NUXY, RigidLinkMaterial, SteelNi
!MAT, RigidLinkMaterial

!create element
TYPE, RigidLink
KSEL,S,KP,,BaseBearing1
NSLK,S
*GET, LastNode, NODE, , NUM, MAX
E, LastNode, NodeStart
ALLSEL,ALL
KSEL,S,KP,,BaseBearing2
NSLK,S
*GET, LastNode, NODE, , NUM, MAX
E, LastNode, NodeEnd

!clear parmaters
*SET, BaseBearing1,
*SET, BaseBearing2,
*SET, NodeBearing1,
*SET, NodeBearing2,
*SET, LastNode,
*SET, i,

```

```
!change view - isometric
EPLOT
/VIEW,1,1,1,1
/ANG,1
/REP,FAST

/COM, ------------------------------------------------
/COM,  BOUNDARY CONDITION
/COM, ------------------------------------------------
ALLSEL, ALL
LSEL, S, LOC, Y, -(RadBaseTail*2)+eps, (RadBaseTail*2)+eps
LSEL, S, LOC, x, 0, 0
*GET, MinBaseTailLine, line, 0, num, min
*GET, MaxBaseTailLine, line, 0, num, max
*DO, i, MinBaseTailLine, MaxBaseTailLine, 8
        DL, i, , ALL, 0
*ENDDO
SAVE

!calc mass
*USE, calcmass.mac, StrJobname(1)
FINISH
SAVE

/COM, ------------------------------------------------
/COM,  PRELIMINARY STATIC ANALYSIS - PRESTRESS
/COM, ------------------------------------------------
! >>>> SOLUTION <<<<
/SOLU
ANTYPE, STATIC, NEW     !Preliminary static analysis
PSTRES, ON              !Includes prestress effects
Acel, , 9.81            !gravitational acceleration

SOLVE
FINISH

! >>>>> ANALISYS PARAMETRS <<<<<
*SET, nmd, 10           ! Number of modes
*SET, nmspn, 8          ! Number of speeds
*SET, mxspn, 4000       ! Max Omega (rpm)

!get element component about a user-defined rotational
!in this case there is only one component
*GET, NumberComponet, 'COMP', , NCOMP
*GET, RotorBlock, 'COMP', NumberComponet, NAME

!get value for axis X1, Y1, Z1 <-> X2, Y2, Z2
*USE, getextremeaxis.mac

/SOLU
!Setup modal run
ANTYPE, MODAL
CORIOLIS, ON, , , ON
MODOPT, QRDAMP, nmd, , , ON

*ABSET, "Loop On Speed - progress: 0%", Bar
```

```
! Loop on speeds
*DO, i, 1, nmspn
        *SET, progress, i/nmspn*100
        spn = (i-1)*(mxspn/(nmspn-1))
        *SET, OMEGAX, spn
        *SET, OMEGAY, 0
        *SET, OMEGAZ, 0
        !CMOMEGA, CM_NAME, OMEGAX, OMEGAY, OMEGAZ, X1, Y1, Z1, X2, Y2, Z2
        /COM,===============================================
        /COM, CMOMEGA, %CM_Name%, %OMEGAX%, %OMEGAY%, %OMEGAZ%,
        /COM, %Node1CoorX%, %Node1CoorY%, %Node1CoorZ%,
        /COM, %Node2CoorX%, %Node2CoorY%, %Node2CoorZ%
        /COM,===============================================
        CMOMEGA, %CM_Name%, OMEGAX, OMEGAY, OMEGAZ, Node1CoorX, Node1CoorY, Node1CoorZ,
            Node2CoorX, Node2CoorY, Node2CoorZ
        MXPAND, nmd, , , YES
        SOLVE
        *ABCHECK, "progress, Loop On Speed - progress: %progress% %"
*ENDDO
*ABFINI
FINISH

/post1
! Plot/List Campbell Diagrams
/gropts,view,2
/yrange,0,100
/xrange,0,4000
plcamp,,1,rpm
prcamp,,1,rpm

! Plot orbit
set,1,2
plorb
! Animate whirl
set,3,1
plnsol,u,sum
anharm
finish
```

Macro

Listing 7.6: testperson.mac

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!                                                  !
!         WINDOW PERSONAZATION STYLE MACRO         !
!           Author: Francesco Argentieri           !
!                                                  !
! NO PARAMETERS INPUT                              !
!                                                  !
!                                                  !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

```
WPSTYLE, , , , , , , , 0
/PLOPTS, INFO, 2
/PLOPTS, LEG1, 1
/PLOPTS, LEG2, 1
/PLOPTS, LEG3, 1
/PLOPTS, FRAME, 0
/PLOPTS, TITLE, 1
/PLOPTS, MINM, 1
/PLOPTS, FILE, 0
/PLOPTS, SPNO, 0
/PLOPTS, WINS, 1
/PLOPTS, WP, 0
/PLOPTS, DATE, 0
/TRIAD, ORIG
/REPLOT
```

Listing 7.7: `generateimages.mac`

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!                                                  !
!              IMAGE GENERATOR MACRO               !
!           Author: Francesco Argentieri           !
!                                                  !
! NO PARAMETERS INPUT                              !
!                                                  !
!                                                  !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

/EFACET, 4
/RGB, INDEX, 100, 100, 100, 0
/RGB, INDEX, 80, 80, 80, 13
/RGB, INDEX, 60, 60, 60, 14
/RGB, INDEX, 0, 0, 0, 15
/REPLOT
/SHOW, PNG, , 0
PNGR, COMP, 0, -1
PNGR, ORIENT, HORIZ
PNGR, COLOR, 2
PNGR, TMOD, 1
/GFILE, 2400,
!*
/REPLOT
/SHOW, CLOSE
/DEVICE, VECTOR, 0
!*
/RGB, INDEX, 0, 0, 0, 0
/RGB, INDEX, 60, 60, 60, 13
/RGB, INDEX, 80, 80, 80, 14
/RGB, INDEX, 100, 100, 100, 15
/REPLOT
/EOF
```

Listing 7.8: `rotor.mac`

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!                                                  !
```

```
!             TAIL ROTOR GENERATOR MACRO            !
!                                                   !
!                                                   !
!PARAMETERS INPUT:                                  !
!                ARG1: 1st node value               !
!                ARG2: 2nd node value               !
!                            ARG3: pass distance between              !
!                                  previous nodes                               !
!                ARG4: Local reference frame        !
!                ARG5: Local reference frame        !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

! >>>>> MODEL PARAMETERS SHAFT AND DISK <<<<<
*SET, shftlen, 0.50
*SET, shftdia, 0.03
*SET, dskoff, shftlen
*SET, brngoff, 0.12
*SET, dskdia, 1.912
*SET, d_i, 1.75
*SET, dskthk, 0.02
*SET, bstf, 378E+7      ! Brng Stiffness

/COM, -----------------------------------------------
/COM,  MODEL DEFINITION -SHAFT
/COM, -----------------------------------------------
! 1st part
*GET, NodeMax, NODE, , NUM, MAX
*SET, NodeStart, NodeMax + 1
N, NodeStart,
N, NodeStart + 8, ARG3 $ /COM, %NodeStart + 8%, %ARG3%
*GET, NodeEnd, NODE, , NUM, MAX
FILL, NodeStart, NodeEnd
! 2nd part
*GET, NodeMax, NODE, , NUM, MAX
N, NodeMax+19, dskoff
*GET, LastNode, NODE, , NUM, MAX
FILL, LastNode,  NodeEnd
N, LastNode, nx(LastNode) + 0.02
*GET, LastNode, NODE, , NUM, MAX
SAVE

/COM, -----------------------------------------------
/COM,  MESH - SHAFT
/COM, -----------------------------------------------

! >>>>> MATERIAL PROPERTIES <<<<<
*GET, MaterialNum, MAT, , NUM, MAX

! Aluminum 7075
*SET, Aluminum7075, MaterialNum + 1
MP, EX, Aluminum7075, 71.7e+9
MP, DENS, Aluminum7075, 2810
MP, NUXY, Aluminum7075, 0.33

/COM, DEFINE BEAM188 for SHAFT
*GET, LastElementType, ETYP, , NUM, MAX
```

```
*SET, BeamShaft, LastElementType + 1
! Use 188 elements, solid
ET, BeamShaft, 188
KEYOPT, BeamShaft, 1, 1
KEYOPT, BeamShaft, 3, 2
KEYOPT, BeamShaft,15, 0

*GET, LastElementType, ETYP, , NUM, MAX
/COM, BEAM188 for SHAFT have number %BeamShaft%
*GET, LastSectionType, SECP, ,NUM, MAX
*SET, SectionBeamShaft, LastSectionType + 1
SECTYPE, SectionBeamShaft, BEAM, CSOLID
SECDATA, shftdia/2
!cm, BeamOmega, elem

! Make elements
SECNUM, SectionBeamShaft
TYPE, BeamShaft
MAT, Aluminum7075
NSEL, S, NODE, , NodeStart, LastNode, 1
*GET, NodeNumShaft, NODE, , COUNT
*GET, NodeStartShaft, NODE, , NUM, MIN
*GET, NodeEndShaft, NODE, , NUM, MAX
*DO, i, NodeStartShaft, NodeEndShaft-1, 1
        E, i, i+1
*ENDDO
SAVE

/COM, -----------------------------------------------
/COM,  MODEL DEFINITION - DISK
/COM, -----------------------------------------------

! >>>> DEFINE GEOMETRY <<<<
WPCSYS, -1
WPOFFS, nx(LastNode), ny(LastNode), nz(LastNode)
WPROTA, , , 90
CYL4, 0, 0, d_i/2, 0, dskdia/2, 360

/COM, -----------------------------------------------
/COM,  MESH - DISK
/COM, -----------------------------------------------
/COM, DEFINE SHELL181 for DISK
*GET, LastElementType, ETYP, , NUM, MAX
*SET, ShellDisk, LastElementType + 1
ET, ShellDisk, SHELL181
KEYOPT, ShellDisk, 3, 2
KEYOPT, ShellDisk, 8, 2

*GET, LastElementType, ETYP, , NUM, MAX
/COM, SHELL181 for Disk have number %ShellDisk%
*GET, LastSectionType, SECP, ,NUM, MAX
*SET, SectionShellDisk, LastSectionType + 1
SECTYPE, SectionShellDisk, SHELL
SECD, dskthk
TYPE, ShellDisk
SECNUM, SectionShellDisk
```

```
MAT, Aluminum7075
!cm, DiscOmega, elem
MSHKEY, 0
!MSHAPE, 1, 2D
ESIZE, 0.02
AMESH, ALL
SAVE

!*** clamp between disk and spindle

/COM, *** CLAMP BETWEEN DISK AND SPINDLE
/COM, DEFINE MASS21 for CLAMP BETWEEN DISK AND SPINDLE
*GET, LastElementType, ETYP, , NUM, MAX
*SET, ClampDiskSplinde, LastElementType + 1
ET, ClampDiskSplinde, MASS21
KEYOPT, ClampDiskSplinde, 2, 0
*GET, ClampDiskSplinde, ETYP, , NUM, MAX
/COM, MASS21 for clamp between disk and spindle have number %ClampDiskSplinde%
R, ClampDiskSplinde, 10, 10, 10, 8.59e-2, 4.295e-2, 4.295e-2
TYPE, ClampDiskSplinde
REAL, ClampDiskSplinde
MAT, Aluminum7075
E, LastNode
SAVE

CERIG, LastNode, LastNode-1, ALL
ALLSEL, ALL
LSEL, S, LOC, X, nx(LastNode)
LSEL, R, RADIUS, , d_i/2
NSLL, S, 1
NSEL, A, NODE, , LastNode
CERIG, LastNode, ALL, ALL
/REP, FAST
SAVE

/COM, ------------------------------------------------
/COM,  MESH - BEARINGS
/COM, ------------------------------------------------
!*** define bearings
*GET, LastElementType, ETYP, , NUM, MAX
*SET, SpringDamper, LastElementType+1
ET, SpringDamper, COMBIN14
*GET, RealConstant, RCON, , NUM, MAX
*SET, SpringDumperRealConst, RealConstant + 1
R, SpringDumperRealConst, bstf

!1-D longitudinal spring-damper (UY degree of freedom)
TYPE, SpringDamper
KEYOPT, SpringDamper, 2, 2
REAL, SpringDumperRealConst
E, ARG1, NodeStart
E, ARG2, NodeEnd
/COM, 1-D longitudinal spring-damper (UY degree of freedom):
/COM,   Node1: %ARG1% (ARG1); Node2: %NodeStart% (NodeStart)
/COM,   Node1: %ARG2% (ARG2); Node2: %NodeEnd% (NodeEnd)

```

```
!***longitudinal spring-damper (UZ degree of freedom)
TYPE, SpringDamper
KEYOPT, SpringDamper, 2, 3
E, ARG1, NodeStart
E, ARG2, NodeEnd
/COM, 1-D longitudinal spring-damper (UZ degree of freedom):
/COM,   Node1: %ARG1% (ARG1); Node2: %NodeStart% (NodeStart)
/COM,   Node1: %ARG2% (ARG2); Node2: %NodeEnd% (NodeEnd)

/COM, -----------------------------------------------
/COM,  REVOLUTE JOINT
/COM,  create revolute joint to block d.o.f shaft
/COM, -----------------------------------------------
!MPC184 Revolute Joint Geometry shows the geometry and node locations for this element.
!Two nodes (I and J) define the element. The two nodes are expected to have identical
    spatial
!coordinates initially.
*GET, LastElementType, ETYP, , NUM, MAX
*SET, RevoluteJoint, LastElementType+1
ET, RevoluteJoint, MPC184
KEYOPT, RevoluteJoint, 1, 6

!If KEYOPT(4) = 0, then element is an x-axis revolute joint with the local e1 axis as
    the revolute axis.
!If KEYOPT(4) = 1, then element is a z-axis revolute joint with the local e3 axis as the
     revolute axis.
KEYOPT, RevoluteJoint, 4, 0

!***define revolute joint
TYPE, RevoluteJoint
SECTYPE, , JOINT, REVO, RevoJoint_1
SECJOINT, , ARG4, ARG4
*GET, LastSectionType, SECP, ,NUM, MAX
SECNUM, LastSectionType
E, ARG1, NodeStart
SECTYPE, , JOINT, REVO, RevoJoint_2
SECJOINT, , ARG5, ARG5
*GET, LastSectionType, SECP, ,NUM, MAX
SECNUM, LastSectionType
E, ARG2, NodeEnd
SAVE

!select element to
ESEL, S, TYPE, , BeamShaft
ESEL, A, TYPE, , ShellDisk
ESEL, A, TYPE, , ClampDiskSplinde
EPLOT

!create a component
CM, RotorBlock, ELEM

!in this case there is only one component
*GET, NumberComponet, 'COMP', , NCOMP
*GET, CM_Name, 'COMP', NumberComponet, NAME
*MSG, UI, NumberComponet, CM_Name
"Created: (%I) component, Named: (%S)"
```

```

!clear parameters
*SET, NodeMax,
*SET, LastElementType,
*SET, RevoluteJoint,
*SET, LastNode,
*SET, LastElementType,
*SET, SpringDamper
*SET, SpringDumperRealConst
*SET, ClampDiskSplinde,
*SET, ShellDisk,
*SET, NodeNumShaft,
*SET, NodeStartShaft,
*SET, NodeEndShaft,
*SET, BeamShaft,
!Exit Macro
/EOF
```

Listing 7.9: `relativereferencesystem.mac`

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!                                                                     !
!              CREATE NEW LOCAL REFERENCE SYSTEM                      !
!                                                                     !
!                                                                     !
!PARAMETERS INPUT:                                                    !
!               ARG1: Number new reference system RF                  !
!               ARG2: KCS - Coordinate system type                    !
!                                       ARG3: x - Location (in the active coordinate system)  !
!                                                   of the origin of the new coordinate system      !
!                                       ARG4: y - Location (in the active coordinate system)  !
!                     of the origin of the new coordinate system      !
!                                       ARG5: z - Location (in the active coordinate system)  !
!                     of the origin of the new coordinate system      !
!                                       ARG6: First rotation about local Z                       !
!                     (positive X toward Y)                           !
!               ARG7: Second rotation about local X                   !
!                     (positive Y toward Z)                           !
!               ARG8: Third rotation about local Y                    !
!                     (positive Z toward X)                           !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

!CLOCAL, KCN, KCS, XL, YL, ZL, THXY, THYZ, THZX
!Defines a local coordinate system relative to the active coordinate system.

!KCN
!Arbitrary reference number assigned to this coordinate system. Must be greater than 10.
!A coordinate system previously defined with this number will be redefined.

!KCS - Coordinate system type:
!0 or CART - Cartesian
```

```
!1 or CYLIN - Cylindrical (circular or elliptical)
!2 or SPHE - Spherical (or spheroidal)
!3 or TORO - Toroidal

!XL, YL, ZL - Location (in the active coordinate system) of the origin of the
!new coordinate system (R, θ, Z for cylindrical, R, θ,Φ for spherical or toroidal).

!THXY - First rotation about local Z (positive X toward Y)
!THYZ - Second rotation about local X (positive Y toward Z)
!THZX - Third rotation about local Y (positive Z toward X).

! define local variable
*SET, i, ARG1
*SET, KCS, ARG2
*SET, XL, ARG3
*SET, YL, ARG4
*SET, ZL, ARG5
*SET, THXY, ARG6
*SET, THYZ, ARG7
*SET, THZX, ARG8

!Set new local reference system
*GET, LastLocalSystem, CDSY, , NUM, MAX
*SET, LocalSystem_%i%, LastLocalSystem + 100
!CLOCAL, KCN, KCS, XL, YL, ZL, THXY, THYZ, THZX
CLOCAL, LocalSystem_%i%, KCS, XL, YL, ZL, THXY, THYZ, THZX
/COM, New local system generated, return: LocalSystem_%i%

!clear local variable
*SET, i,
*SET, KCS,
*SET, XL,
*SET, YL,
*SET, ZL,
*SET, THXY,
*SET, THYZ,
*SET, THZX,
*SET, LastLocalSystem

!reset to roginal system refence
CSYS, 0
/EOF
```

Listing 7.10: `distancenode.mac`

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!                                                 !
!            NODE DISTANCE CALCULATOR             !
!                                                 !
!PARAMETERS INPUT:                                !
!                 ARG1: 1st Node value            !
!                 ARG2: 2nd Node value            !
!RETURN:                                          !
!               DistancePoint                     !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

```
!define local variable
*GET, Node1CoorX, NODE, ARG1, LOC, X
*GET, Node1CoorY, NODE, ARG1, LOC, Y
*GET, Node1CoorZ, NODE, ARG1, LOC, Z
*GET, Node2CoorX, NODE, ARG2, LOC, X
*GET, Node2CoorY, NODE, ARG2, LOC, Y
*GET, Node2CoorZ, NODE, ARG2, LOC, Z

*SET, x_1, Node1CoorX
*SET, y_1, Node1CoorY
*SET, z_1, Node1CoorZ
*SET, x_2, Node2CoorX
*SET, y_2, Node2CoorY
*SET, z_2, Node2CoorZ

!calculates the distance between your nodes according to the formula
!dist^2=(x_1-x_2)^2+(y_1-y_2)^2+(z_1-z_2)^2
!Return value DistancePoint
*SET, DistancePoint, ((x_1-x_2)**2+(y_1-y_2)**2+(z_1-z_2)**2)**(1/2)

!clear local variable
*SET, Node1CoorX,
*SET, Node1CoorY,
*SET, Node1CoorZ,
*SET, Node2CoorX,
*SET, Node2CoorY,
*SET, Node2CoorZ,
*SET, x_1,
*SET, y_1,
*SET, z_1,
*SET, x_2,
*SET, y_2,
*SET, z_2,
/EOF
```

Listing 7.11: calcmass.mac

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!                                                   !
!          WINDOW PERSONAZATION STYLE MACRO         !
!            Author: Francesco Argentieri           !
!                                                   !
! PARAMETERS INPUT:                                 !
!                   ARG1: JOBNAME string            !
!                                                   !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

/SOLVE
OUTPR, BASIC, ALL
/OUTPUT, mass_output-%ARG1%, txt
PSOLVE, ELFORM
PSOLVE, ELPREP
/OUT
FINISH
/EOF
```

Listing 7.12: getextremeaxis.mac

```
!CMGRP, RotorBlockGrouped, BeamOmega, DiscOmega, HubOmega

!CMOMEGA, CM_NAME, OMEGAX, OMEGAY, OMEGAZ, X1, Y1, Z1, X2, Y2, Z2
!Specifies the rotational velocity of an element component about a user-defined rotational axis.
!SOLUTION: Inertia

!CM_NAME
!The name of the element component.

!OMEGAX, OMEGAY, OMEGAZ
!If the X2, Y2, Z2 fields are not defined, OMEGAX, OMEGAY, and OMEGAZ specify the components of
!the rotational velocity vector in the global Cartesian X, Y, Z directions.

!If the X2, Y2, Z2 fields are defined, only OMEGAX is required. OMEGAX specifies the scalar
!rotational velocity about the rotational axis. The rotational direction of OMEGAX is designated
!either positive or negative, and is determined by the right hand rule.

!X1, Y1, Z1
!If the X2, Y2, Z2 fields are defined,X1, Y1, and Z1 define the coordinates of the beginning
!point of the rotational axis vector. Otherwise, X1, Y1, and Z1 are the coordinates of a point
!through which the rotational axis passes.

!X2, Y2, Z2
!The coordinates of the end point of the rotational axis vector.

!Change RF
CSYS,  LocalSystem_1
WPCSYS, -1

!Select node
NSEL, S, LOC, Y, , 0, shftlen + dskthk
NSEL, R, LOC, z, , 0, 0
*GET, StartNode, NODE, , NUM, MIN
*GET, EndNode, NODE, , NUM, MAX
ALLSEL,ALL
CSYS, 0
WPCSYS, -1
!Plot nodes
NPLOT
*DO, j, StartNode, EndNode
        /COM, %j%
*ENDDO

!define local variable
*GET, Node1CoorX, NODE, StartNode, LOC, X
*GET, Node1CoorY, NODE, StartNode, LOC, Y
*GET, Node1CoorZ, NODE, StartNode, LOC, Z
*GET, Node2CoorX, NODE, EndNode, LOC, X
*GET, Node2CoorY, NODE, EndNode, LOC, Y
```

```
*GET, Node2CoorZ, NODE, EndNode, LOC, Z
/COM,=================================================
/COM, CMOMEGA, ROTOR, OMEGAX, OMEGAY, OMEGAZ,
/COM, %Node1CoorX%, %Node1CoorY%, %Node1CoorZ%,
/COM, %Node2CoorX%, %Node2CoorY%, %Node2CoorZ%
/COM,=================================================
/EOF
```

Listings

List of Figures

Bibliography

[1] P. Tonella, A. Susi, and F. Palma, “Using interactive {GA} for requirements prioritization,” in *Search Based Software Engineering (SSBSE), 2010 Second International Symposium on*, September 2010, pp. 57 –66.

[2] Wikipedia, “Aérospatiale as 350 Écureuil — wikipedia, l’enciclopedia libera,” 2017, [Online; in data 1-maggio-2017]. [Online]. Available: http://it.wikipedia.org/w/index.php?title=A%C3%A9rospatiale_AS_350_%C3%89cureuil&oldid=86772639