Free vibration analysis for non-straight helicopter blades

Nicolás Capmany, Department of Mechanical Engineering, University of Bristol

ABSTRACT: This paper describes the derivation of two simple two degree of freedom (D.O.F) multibody models, implemented to evaluate the frequency spectrum of a helicopter blade. The first model proposed was a Lag-Flap blade model, accounting for the structure's typical in-plane and out-of-plane motion during aircraft flight. The second model, Flap-Flap, only considered the blade's out-of-plane vibrations. Lagrangian mechanics was applied to achieve the equations of motion of both systems. MATLAB's standard differential equation solver ODE45 was used to solve the system of differential equations for each model. Results achieved with ODE45 were validated with a physical simulation of the multibody systems. These were performed using Simscape blocks, and suitable results were achieved for a static case with an initial deformation applied to the structure. The models, were subsequently utilised to evaluate any alterations in the structure's frequency spectrum with a changing blade curvature. A shift of the natural frequency peaks towards the lower end of the spectrum was observed for an increasing blade curvature for the out-of-plane vibrations. Regarding the in-plane vibratory motion, no significant modifications were appreciated. A maximum number of two excitation modes were observed, the setup used did not prove to be ready for industrial applications. However, it provides a starting point to different methodologies, which could be applied as an alternative measure for the techniques used by industry.

1 INTRODUCTION & LITERATURE REVIEW

1.1 Introduction

Aircraft design implies considering the impact of a vast range of parameters. On one hand, such parameters can be key to assess the aircraft's effectiveness for a certain mission. An example could be the by-pass ratio of a civil aircraft's engine, which ensures reasonable fuel consumption and economic viability. On the other hand, different parameters are focused on guaranteeing the aircraft's safe performance subject to several flight conditions (e.g. helicopter blade's natural frequencies).

Helicopter blade's natural frequencies require higher rotary regimes than those needed to lift the vehicle. Meanwhile, lower regimes are desirable with the purpose of reducing fuel consumption, resulting in lower CO_2 emissions, since environmental issues are a major global concern. Natural frequencies of helicopter blades, must not coalesce with the main rotor's regime, as this would bring the blades into resonance. As a consequence, these would vibrate with a linearly increasing amplitude, leading to structural failure at a certain point in time. Therefore, the aircraft's rotary regime is determined by these natural frequencies, which are strongly dependent on the blade's geometry and inertial characteristics.

Helicopter blades are subject to a wide range of geometrical modifications in order to achieve the highest aerodynamic attainment, which is mainly accomplished by non-linear geometry aspects. Therefore, knowing the impact of the blade's geometric parameters on its natural frequencies allows tailoring these, hence their importance. Finite Element Analysis (FEA), is the standard design technique implemented by industry, requiring sophisticated software packages which are computationally intensive, require thorough training and are time consuming.

Here, we propose the use of a simple mathematical tool such as MATLAB's ODE45 solver, to compute the natural frequencies of non-straight blades, providing a reasonable trade-off

between accuracy and computational time. A relatively simple mathematical model was developed, based on three interlinked rigid bodies. The junctions amongst these solids, are provided by means of high stiffness torsional springs, in conjunction with torsional dampers with significant damping coefficients. The purpose of implementing these elements, was to approximate the multibody structure, to a single rigid body that resembled the non-straight profile of current commercial blades.

Subsequently, a kinematic study was conducted to achieve the position and velocity expressions required to apply Lagrangian mechanics. Calculations were performed in MATLAB, using its Symbolic Math Toolbox. Final results were accomplished using MATLAB ODE45 solver, and validated afterwards for a static case, with an initial blade deformation. Validation was achieved by means of a physical simulation creating a SIMSCAPE model. These kind of models are tools within MATLAB's Simulink environment, which offer an alternative method to tackle natural frequency acquisition, with a visual programming environment.

1.2 Software

MATLAB software is one the most relevant products of MathWorks corporation and "is a powerful technical computing system for handling scientific and engineering calculations." (Hahn and Valentine 2019, p. 3). With the aim of achieving this purpose, MATLAB implements its own high-level programming language. The software was designed to relieve the computational cost of matrix calculations, therefore, programs and routines developed with its programming language, are optimised with variable vectorisation. MATLAB includes a wide range of mathematical tools, enabling the user to solve numerical and analytical problems with different methodologies. Amongst these tools, we can find Symbolic Math Toolbox and ODE45 solver.

MathWorks (2020) provides the following definitions for Symbolic Math Toolbox and ODE45 solver:

What is Symbolic Math Toolbox and ODE45?

> MATLAB's Symbolic Math Toolbox, is a package that provides functions for solving, plotting and manipulating symbolic math equations. The toolbox provides functions in common mathematical areas such as calculus, ordinary differential equations, or equation simplification. In addition, it allows the user to analytically perform differentiation, integration, transforms, and equation solving.
>
> ODE45 is a MATLAB tool designed to solve systems of ordinary differential equations, containing one or more derivatives of a dependent variable, with respect to a single independent variable. Amongst the 8 solvers offered by MATLAB, ODE45 is specifically convenient for non-stiff problems providing a medium accuracy.

What is Simulink and Simscape?

> Simulink is a block diagram environment for multi-domain simulation and Model-Based Design. It supports system-level design, simulation, automatic code generation and continuous test and verification of embedded systems. Simulink provides a graphical editor, customisable block libraries, and solvers for modelling and simulating dynamic systems (MathWorks 2020).
>
> Simscape enables you to rapidly create models of physical systems within the Simulink environment. With Simscape, you build physical component models based on physical connections that directly integrate with block diagrams and other modelling paradigms. (MathWorks 2020).

2 METHODOLODY

2.1 Single Degree of Freedom Model

Configuration

We consider an inertial reference frame placed at the centre of the helicopter's rotor, the latter, has a constant rotational speed. In practice, a helicopter blade under a rotating prescribed motion, undergoes two constituent free motions outside this reference frame. The first occurs about a hinge commonly known as (L) or *lead-lag* hinge, which describes the structure's in-plane vibrations with respect to the blade's 2D motion plane. Figure 1a represents the first of these vibrational movements, usually referred to as the blade's *lead-lag motion*. The second, represents the blade's out-of-plane vibrations, occurring about a second hinge placed along the span, often known as (F) or *flapping* hinge. This second type of movement is often called *flapping motion,* illustrated by Figure 1b. Both vibrational phenomena take place throughout the aircraft's performance, and represent the structure's natural vibratory behaviour.

With the purpose of addressing the simulation of a helicopter blade in a first attempt, two simple separate models were suggested, where the blade's motion was restricted to the rotary motion provided by the hub and one of the two previously mentioned free motions, Figures 1a and 1b.

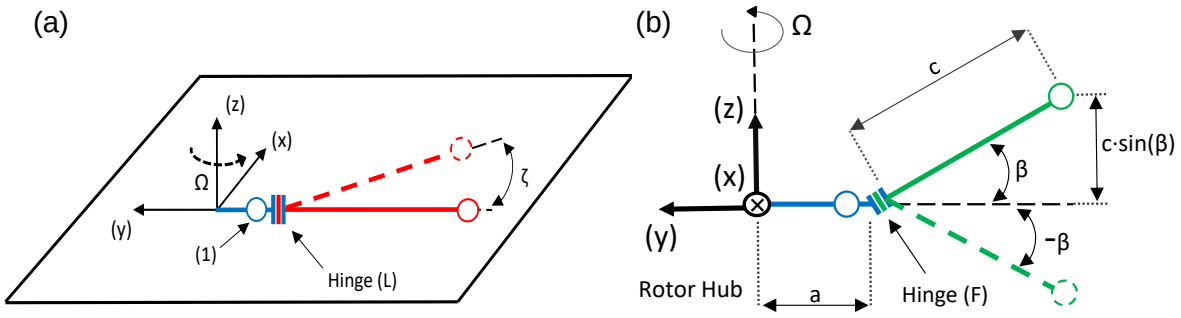


Figure 1 Some pictures of (a) an isometric view of a single D.O.F model with a rotating hub (blue) and a simple leag-lagging blade (red) moving about hinge (L) and (b) a side view of a single D.O.F model, showing a rotating hub (blue) and a simple flapping blade (green) linked by hinge (F).

Two final blade models resulted from the combination of these two simple cases. Figures 2, 3, 4 and 5 picture them. The first, known as the Lag-Flap model (L-F) has a blade subject to in-plane and out-of-plane vibrations. The second, known as the Flap-Flap model (F-F), has different parts of the blade's structure only undergoing out-of-plane vibrations.

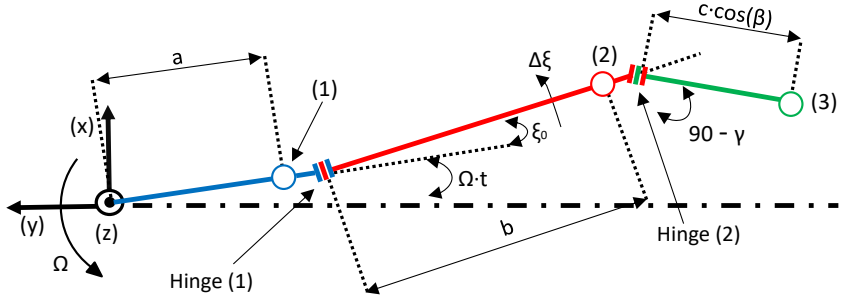


Figure 2 Plan view of the L-F model, according to the assumptions made. Generic configuration applied to derive the system's kinematic expressions.

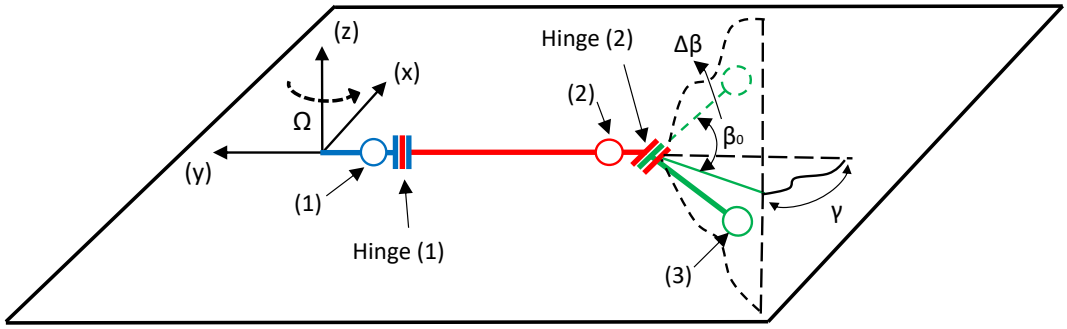


Figure 3 Isometric view of the L-F model according to the assumptions made.

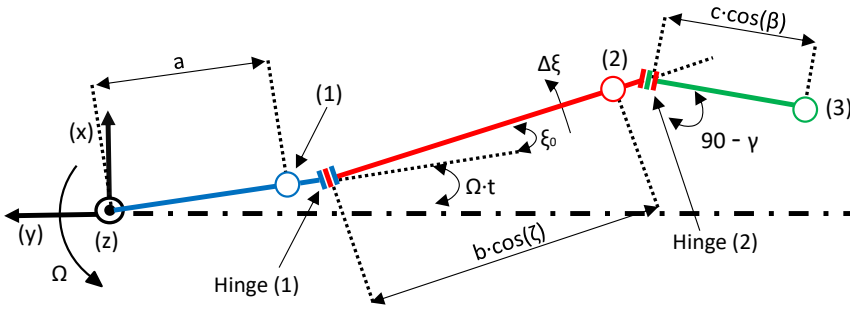


Figure 4 Plan view of the F-F model according to the assumptions made. Generic configuration applied to derive the systems kinematic expressions.

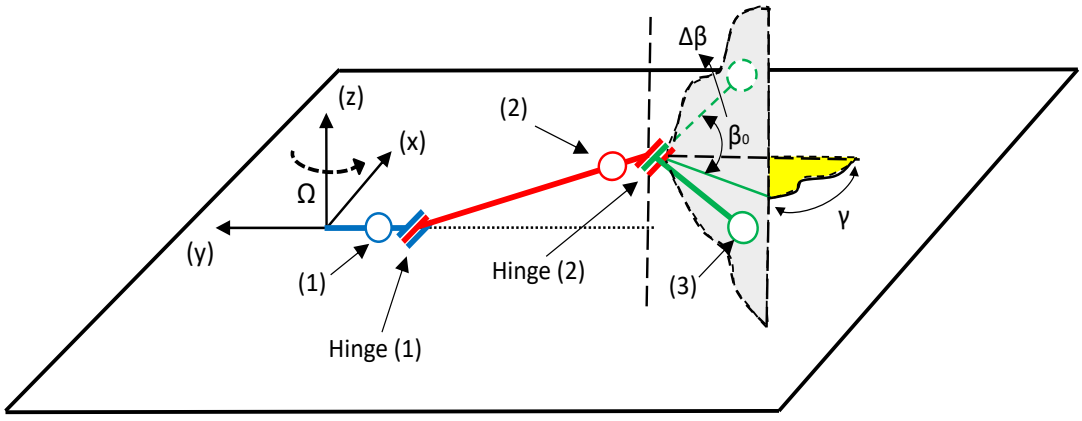


Figure 5 Isometric view of the F-F model according to the assumptions made.

Assumptions

The equations describing the motion of both models, are derived under several assumptions: 1) A helicopter hub simplification is implemented assuming a rod-like structure, the same procedure is applied to the rest of the blade to simplify visual perspective; 2) The mass of the hub and the lead-lag/flapping segments, are contemplated to act as point masses located at the end of each corresponding rod; 3) Torsional springs and dampers are used to represent the blade's non-linear geometry, while introducing rigidity and approximating the multibody system to a single body; 4) We considered a perfectly rigid hub, implying the absence of motion on the hub in case no rotary motion was prescribed. As a result, the same effect should be expected if a predefined deformation was applied to the structure, for example, with non-zero initial conditions; 5) Dimensions of the lead-lag and flapping hinges (L and F) respectively were only considered for calculations involving the curved section of the blade, (green segment). Hinges are represented by sets of short parallel lines at the end of each rod. The lengths of

the segments which comprise the blade, are represented by common letters of the roman alphabet, Figures (2-5).

2.2 Lag-Flap Model Equations

Position

The expressions implemented in our MATLAB routine comprised 12 equations describing the position of three key points in the blade's structure along the x, y and z axis. According to Figures 2 and 3, these points, labelled as (1), (2) and (3), denote the location of the concentrated mass of each segment in the multibody system. The equations of motion can be written as:

$$x_1 = \quad a \cdot \sin(\Omega \cdot t) \tag{1}$$

$$y_1 = -a \cdot \cos(\Omega \cdot t) \tag{2}$$

$$z_1 = \quad 0 \tag{3}$$

$$x_2 = \quad b \cdot \sin(\Omega \cdot t + \zeta_0 + \Delta\zeta) \tag{4}$$

$$y_2 = -b \cdot \cos(\Omega \cdot t + \zeta_0 + \Delta\zeta) \tag{5}$$

$$z_2 = \quad 0 \tag{6}$$

$$x_{2h} = \quad B \cdot \sin(\Omega \cdot t + \zeta_0 + \Delta\zeta) \tag{7}$$

$$y_{2h} = -B \cdot \cos(\Omega \cdot t + \zeta_0 + \Delta\zeta) \tag{8}$$

$$z_{2h} = \quad 0 \tag{9}$$

$$x_3 = \quad c \cdot \cos(\beta_0 + \Delta\beta) \cdot \sin(\Omega \cdot t + \zeta_0 + \Delta\zeta - \gamma) \tag{10}$$

$$y_3 = -c \cdot \cos(\beta_0 + \Delta\beta) \cdot \cos(\Omega \cdot t + \zeta_0 + \Delta\zeta - \gamma) \tag{11}$$

$$z_3 = \quad c \cdot \sin(\beta_0 + \Delta\beta) \tag{12}$$

where $\Delta\zeta(t)$, $\Delta\beta(t)$ represent the lead-lag and flapping angular displacements of the blade respectively; ζ_0, β_0 are their corresponding equilibrium positions, determined by design criteria; *a* stands for the hub's radius, *b* represents segment 1's length along the blade, from hinge (L) to hinge (F); *B* is segment 1's length plus an increment due to both hinges' dimensions; *c* is the blade's flapping segment length (refer to Figure 2); Ω describes the rotating regime of the vehicle's rotor; *t* represents the independent variable representing time; x_1, y_1, z_1 stand for the coordinates of the hub's concentrated mass, located at point (1) with respect to the reference frame; x_2, y_2, z_2 are the coordinates denoting the relative displacement of segment 1's point mass located at point (2), with respect to point (1)'s location; x_{2h}, y_{2h}, z_{2h} are the coordinates of hinge (2), which are computed adding a length increment to x_{2h}, y_{2h}, z_{2h}, to account for both hinges' dimensions; x_3, y_3, z_3 stand for the relative displacement of segment 2's point mass, located at point (3), with respect to point (2), and γ denotes the blade's sweep angle or curvature.

Note that the effect of the hinge's length in the model's dimensions was only considered when both hinges were involved to determine the location of a point within the model. Therefore, the effect of hinge (1)'s dimensions were neglected until reaching hinge (2) along the blade.

Considering that equations (1-12) are relative displacements, the absolute coordinates of these points about the inertial reference frame, are obtained with a suitable addition of these relative displacements, yielding the following equations:

$$x_{hub} = x_1 \tag{13}$$

$$y_{hub} = y_1 \tag{14}$$

$$z_{hub} = z_1 \tag{15}$$

$$x_{seg1} = x_1 + x_2 \tag{16}$$

$$y_{seg1} = y_1 + y_2 \tag{17}$$

$$z_{seg1} = z_1 + z_2 \tag{18}$$

$$x_{seg1h} = x_1 + x_{2h} \tag{19}$$

$$y_{seg1h} = y_1 + y_{2h} \tag{20}$$

$$z_{seg1h} = z_1 + z_{2h} \tag{21}$$

$$x_{seg2} = x_1 + x_{2h} + x_3 \tag{22}$$

$$y_{seg2} = y_1 + y_{2h} + y_3 \tag{23}$$

$$z_{seg2} = z_1 + z_{2h} + z_3 \tag{24}$$

where x_{hub}, y_{hub} are the absolute coordinates of point (1), x_{seg1}, y_{seg1} are the absolute coordinates of point (2), x_{seg1h}, y_{seg1h} are the absolute coordinates of hinge (2), x_{seg2}, y_{seg2} represent the absolute coordinates of point (3), and the rest of variables remain the same with respect to the description provided for equations (1-12). Note that expressions for x_{hub}, y_{hub} and z_{hub} remain the same as those achieved in equations (1, 2 and 3) respectively; their coordinates were already calculated with respect to the inertial reference frame, according to Figure 2.

Velocity

Differentiating expressions (13-24) with respect to time yields:

$$v_{xhub} = \frac{dx_{hub}}{dt} = \Omega \cdot a \cdot \cos(\Omega \cdot t) \tag{25}$$

$$v_{yhub} = \frac{dy_{hub}}{dt} = \Omega \cdot a \cdot \sin(\Omega \cdot t) \tag{26}$$

$$v_{zhub} = \frac{dz_{hub}}{dt} = 0 \tag{27}$$

$$v_{xseg1} = \frac{dx_{seg1}}{dt} = \Omega \cdot a \cdot \cos(\Omega \cdot t) + b \cdot \cos(\zeta_0 + \Delta\zeta + \Omega \cdot t) \cdot (\dot{\Delta\zeta} + \Omega) \tag{28}$$

$$v_{yseg1} = \frac{dy_{seg1}}{dt} = \Omega \cdot a \cdot \sin(\Omega \cdot t) + b \cdot \sin(\zeta_0 + \Delta\zeta + \Omega \cdot t) \cdot (\dot{\Delta\zeta} + \Omega) \tag{29}$$

$$v_{zseg1} = \frac{dz_{seg1}}{dt} = 0 \tag{30}$$

$$v_{xseg1h} = \frac{dx_{seg1h}}{dt} = \Omega \cdot a \cdot \cos(\Omega \cdot t) + B \cdot \cos(\zeta_0 + \Delta\zeta + \Omega \cdot t) \cdot (\dot{\Delta\zeta} + \Omega) \tag{31}$$

$$v_{yseg1h} = \frac{dy_{seg1h}}{dt} = \Omega \cdot a \cdot \sin(\Omega \cdot t) + B \cdot \sin(\zeta_0 + \Delta\zeta + \Omega \cdot t) \cdot (\dot{\Delta\zeta} + \Omega) \tag{32}$$

$$v_{zseg1h} = \frac{dz_{seg1h}}{dt} = 0 \tag{33}$$

$$v_{xseg2} = \frac{dy_{seg2}}{dt} = \begin{Bmatrix} \Omega \cdot a \cdot \cos(\Omega \cdot t) + B \cdot \cos(\zeta_0 + \Delta\zeta + \Omega \cdot t) \cdot (\dot{\Delta\zeta} + \Omega) + \\ +c \cdot \cos(\zeta_0 + \Delta\zeta + \Omega \cdot t - \gamma) \cdot \cos(\beta_0 + \Delta\beta) \cdot (\dot{\Delta\zeta} + \Omega) - \\ -c \cdot \sin(\zeta_0 + \Delta\zeta + \Omega \cdot t - \gamma) \cdot \sin(\beta_0 + \Delta\beta) \cdot \dot{\Delta\beta} \end{Bmatrix} \tag{34}$$

$$v_{yseg2} = \frac{dy_{seg2}}{dt} = \begin{Bmatrix} \Omega \cdot a \cdot \sin(\Omega \cdot t) + B \cdot \sin(\zeta_0 + \Delta\zeta + \Omega \cdot t) \cdot (\dot{\Delta\zeta} + \Omega) + \\ +c \cdot \sin(\zeta_0 + \Delta\zeta + \Omega \cdot t - \gamma) \cdot \cos(\beta_0 + \Delta\beta) \cdot (\dot{\Delta\zeta} + \Omega) - \\ -c \cdot \cos(\zeta_0 + \Delta\zeta + \Omega \cdot t - \gamma) \cdot \sin(\beta_0 + \Delta\beta) \cdot \dot{\Delta\beta} \end{Bmatrix} \tag{35}$$

$$v_{zseg2} = \frac{dz_{seg2}}{dt} = \mathrm{c} \cdot \cos(\beta_0 + \Delta\beta) \cdot \dot{\Delta\beta} \tag{36}$$

where v_{xhub}, v_{yhub}, v_{zhub} are the velocity components of the hub's point mass in the x, y and directions respectively; v_{xseg1}, v_{yseg1}, v_{zseg1} are the velocity components of the concentrated mass of the blade's lead-lagging segment; v_{xseg1h}, v_{yseg1h}, v_{zseg1h} are the velocity components at the

flapping hinge, (hinge 2), and v_{xseg2}, v_{yseg2}, v_{zseg2} are the velocity components of the flapping segment's point mass.

Kinetic Energy

$$E_{khub} = \frac{1}{2} \cdot M_{hub} \cdot \left({v_{xhub}}^2 + {v_{yhub}}^2 + {v_{zhub}}^2 \right) \quad (37)$$

$$E_{kseg1} = \frac{1}{2} \cdot M_{seg1} \cdot \left({v_{xseg1}}^2 + {v_{yseg1}}^2 + {v_{zseg1}}^2 \right) \quad (38)$$

$$E_{kseg2} = \frac{1}{2} \cdot M_{seg2} \cdot \left({v_{xseg2}}^2 + {v_{yseg2}}^2 + {v_{zseg2}}^2 \right) \quad (39)$$

$$E_k = E_{khub} + E_{kseg1} + E_{kseg2} \quad (40)$$

where M_{hub} represents the hub's concentrated mass, M_{seg1} is the blade's lead-lag section's mass, M_{seg2} is the mass of the blade's flapping section, E_{khub}, E_{kseg1}, E_{kseg2} stand for the kinetic energies of the hub, lead-lag section and flapping section respectively, and E_k is the system's total kinetic energy.

Potential Energy

Following the schematics depicted by Figures 6a and 6b, we derived the generic expressions to compute the elastic potential energy (EPE), for the lead-lag and flapping torsional springs included in the multibody model:

$$E_{p\,seg1} = \frac{1}{2} \cdot k_z \cdot (\Delta\zeta)^2 + M_{seg1} \cdot g \cdot z_{seg1} \quad (41)$$

$$E_{p\,seg2} = \frac{1}{2} \cdot k_b \cdot (\Delta\beta)^2 + M_{seg2} \cdot g \cdot z_{seg2} \quad (42)$$

$$E_p = E_{pseg1} + E_{pseg2} \quad (43)$$

where E_{pseg1} is the EPE of the lead-lag torsional spring, E_{pseg2} is the EPE of the flapping torsional spring, and E_p stands for the system's total EPE.

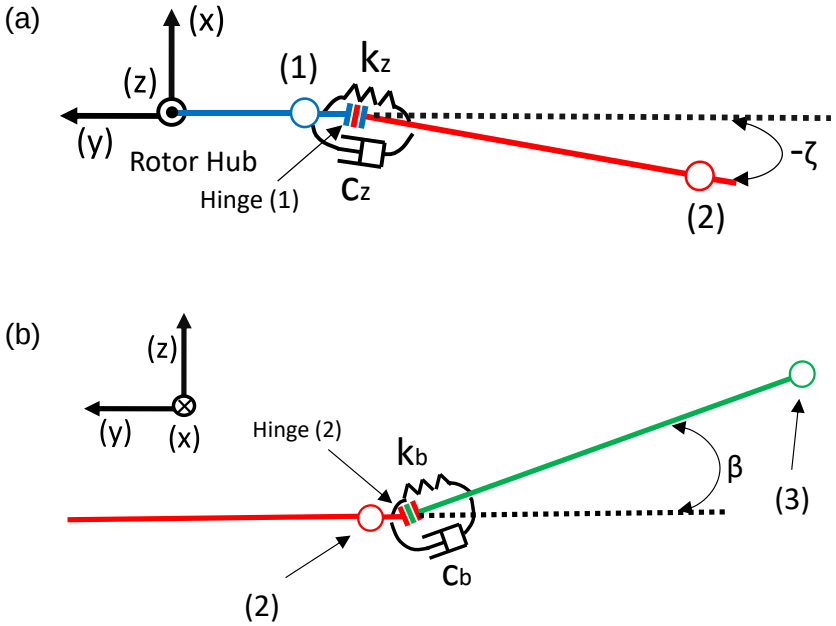


Figure 6 Some pictures of (a) a plan view of the rotor and lead-lag section with its corresponding spring and damper and (b) a side view of hinge (2) linking the lead-lag and flapping segments with its corresponding spring and damper.

Damping

Considering that two torsional dampers were included in the proposed multibody model, represented by Figures 6a and 6b, the following expressions are calculated to account for their energy dissipation:

$$D_z = \frac{1}{2} \cdot c_z \cdot \left(\dot{\Delta\zeta}\right)^2 \tag{44}$$

$$D_b = \frac{1}{2} \cdot c_b \cdot \left(\dot{\Delta\beta}\right)^2 \tag{45}$$

where D_z, D_b are the lead-lag torsional damper and the flapping torsional damper functions respectively, and c_z, c_b stand for the torsional damping coefficients of the lead-lag torsional damper and the flapping torsional damper in that order.

External Forces Applied

With the objective of exciting several vibration modes, a sinusoidal impulse with a varying (chirped) frequency was applied. Therefore, the following expression represents the external non-conservative forces applied on the system:

$$Q_1 = Q_{amp} \cdot \sin(s_{rate} \cdot t^2) \tag{46}$$

$$Q_2 = Q_{amp} \cdot \sin(s_{rate} \cdot t^2) \tag{47}$$

where Q is the external force applied, Q_{amp} is the force's amplitude, s_{rate} is the frequency's rate of change, and t denotes time. Noteworthy, two different cases were assessed for the L-F model: 1) A case with an external force Q_1 applied on the system's first D.O.F ($\Delta\zeta$); 2) A case with an external force Q_2 applied on the system's second D.O.F ($\Delta\beta$).

Lagrangian & Equations of motion

Subtracting equation (43) from (40) to represent the system's conservative forces, yields:

$$L = E_k - E_p \tag{48}$$

where L is the system's lagrangian. Considering the case where the L-F model is excited by the external force Q_1, L is substituted into Lagrange's equations of motion. The final step consisted on equating the expressions to the non-conservative forces:

$$\frac{d}{dt} \cdot \left(\frac{\partial L}{\partial \dot{\Delta\zeta}}\right) - \left(\frac{\partial L}{\partial \Delta\zeta}\right) = -\left(\frac{\partial D_z}{\partial \dot{\Delta\zeta}}\right) + Q_1 \tag{49}$$

$$\frac{d}{dt} \cdot \left(\frac{\partial L}{\partial \dot{\Delta\beta}}\right) - \left(\frac{\partial L}{\partial \Delta\beta}\right) = -\left(\frac{\partial D_b}{\partial \dot{\Delta\beta}}\right) \tag{50}$$

For the case where the system is excited by the external force Q_2, the resulting system of equations would be:

$$\frac{d}{dt} \cdot \left(\frac{\partial L}{\partial \dot{\Delta\zeta}}\right) - \left(\frac{\partial L}{\partial \Delta\zeta}\right) = -\left(\frac{\partial D_z}{\partial \dot{\Delta\zeta}}\right) \tag{51}$$

$$\frac{d}{dt} \cdot \left(\frac{\partial L}{\partial \dot{\Delta\beta}}\right) - \left(\frac{\partial L}{\partial \Delta\beta}\right) = -\left(\frac{\partial D_b}{\partial \dot{\Delta\beta}}\right) + Q_2 \tag{52}$$

For the L-F model, equations (49 and 50) and (51 and 52), showed the following arrangement:

$$\begin{Bmatrix} a \cdot \frac{\partial^2 \Delta\zeta}{\partial t^2} = b \cdot \left(\frac{\partial \Delta\zeta}{\partial t}\right)^2 + c \cdot \left(\frac{\partial \Delta\beta}{\partial t}\right)^2 + d \cdot \left(\frac{\partial \Delta\zeta}{\partial t}\right) + e \cdot \left(\frac{\partial \Delta\beta}{\partial t}\right) \\ f \cdot \left(\frac{\partial \Delta\zeta}{\partial t} \cdot \frac{\partial \Delta\beta}{\partial t}\right) + g \cdot \Delta\zeta + h \cdot \Delta\beta + j \cdot (\Delta\zeta \cdot \Delta\beta) + k_1 \end{Bmatrix} \tag{53}$$

$$\begin{Bmatrix} l \cdot \frac{\partial^2 \Delta\beta}{\partial t^2} = m \cdot \left(\frac{\partial \Delta\zeta}{\partial t}\right)^2 + n \cdot \left(\frac{\partial \Delta\beta}{\partial t}\right)^2 + p \cdot \left(\frac{\partial \Delta\zeta}{\partial t}\right) + q \cdot \left(\frac{\partial \Delta\beta}{\partial t}\right) \\ r \cdot \left(\frac{\partial \Delta\zeta}{\partial t} \cdot \frac{\partial \Delta\beta}{\partial t}\right) + s \cdot \Delta\zeta + t \cdot \Delta\beta + u \cdot (\Delta\zeta \cdot \Delta\beta) + k_2 \end{Bmatrix} \tag{54}$$

where the roman alphabet letters are the constant common factors multiplying each of the terms in equations (53 and 54) and k_1, k_2 are independent constants. This result was of significant importance, as it provided information about the coupling terms between both degrees of freedom; As a result, we were able to appreciate the model's parameters with a higher influence on the system's response to an external excitation, (Appendix C).

2.3 Flap-Flap Model Equations

Contemplating Figures 4 and 5, the location of each segment's point mass was initially computed with the following expressions:

$$x_1 = a \cdot \sin(\Omega \cdot t) \quad (55)$$

$$y_1 = -a \cdot \cos(\Omega \cdot t) \quad (56)$$

$$z_1 = 0 \quad (57)$$

$$x_2 = b \cdot \cos(\zeta_0 + \Delta\zeta) \cdot \sin(\Omega \cdot t) \quad (58)$$

$$y_2 = -b \cdot \cos(\zeta_0 + \Delta\zeta) \cdot \cos(\Omega \cdot t) \quad (59)$$

$$z_2 = b \cdot \sin(\zeta_0 + \Delta\zeta) \quad (60)$$

$$x_{2h} = B \cdot \cos(\zeta_0 + \Delta\zeta) \cdot \sin(\Omega \cdot t) \quad (61)$$

$$y_{2h} = -B \cdot \cos(\zeta_0 + \Delta\zeta) \cdot \cos(\Omega \cdot t) \quad (62)$$

$$z_{2h} = B \cdot \sin(\zeta_0 + \Delta\zeta) \quad (63)$$

$$x_3 = c \cdot \cos(\beta_0 + \Delta\beta) \cdot \sin(\Omega \cdot t - \gamma) \quad (64)$$

$$y_3 = -c \cdot \cos(\beta_0 + \Delta\beta) \cdot \cos(\Omega \cdot t - \gamma) \quad (65)$$

$$z_3 = c \cdot \sin(\beta_0 + \Delta\beta) \quad (66)$$

where $\Delta\zeta$ denotes the first flapping angle of the Flap-Flap model, $\Delta\beta$ is the model's second flapping angle, and the rest of variables are the same as those previously described in equations (1-12). An equivalent procedure to the steps followed for expressions (1-12) was applied to equations (55-66) of the F-F model. The steps will be omitted for the sake of brevity. Note that the elements accounting for the system's torsional springs and dampers are no longer pictured by Figures 6a and 6b, but by Figures 7a and 7b instead. Also, note that the inertial reference frame has now changed.

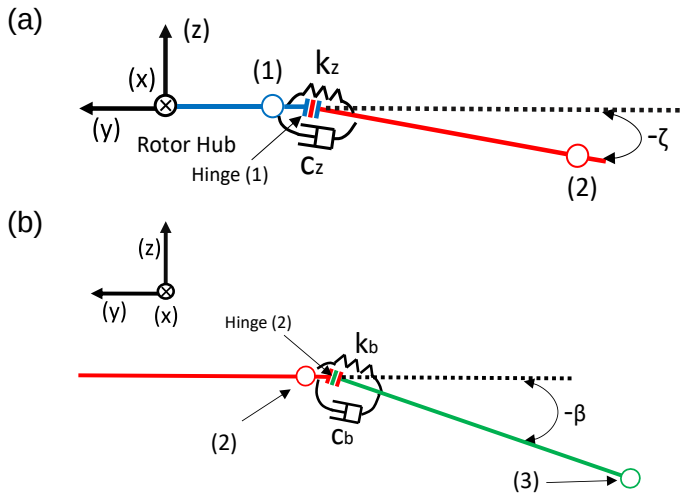


Figure 7 Some pictures of (a) a side view of the rotor and its first flapping section,linked by hinge (1), with its corresponding spring and damper and (b) a side view of hinge (2) linking the first and second flapping sections, with its corresponding spring and damper.

The final step yielded to expressions, showing the following general arrangement:

$$\left\{\begin{matrix} a\cdot\frac{\partial^2\Delta\zeta}{\partial t^2}+b\cdot\frac{\partial^2\Delta\beta}{\partial t^2}=c\cdot\left(\frac{\partial\Delta\zeta}{\partial t}\right)^2+d\cdot\left(\frac{\partial\Delta\beta}{\partial t}\right)^2+e\cdot\left(\frac{\partial\Delta\zeta}{\partial t}\right)+f\cdot\left(\frac{\partial\Delta\beta}{\partial t}\right) \\ g\cdot\left(\frac{\partial\Delta\zeta}{\partial t}\cdot\frac{\partial\Delta\beta}{\partial t}\right)+h\cdot\Delta\zeta+j\cdot\Delta\beta+l\cdot(\Delta\zeta\cdot\Delta\beta)+k_1 \end{matrix}\right\} \quad (67)$$

$$\left\{\begin{matrix} m\cdot\frac{\partial^2\Delta\zeta}{\partial t^2}+n\cdot\frac{\partial^2\Delta\beta}{\partial t^2}=p\cdot\left(\frac{\partial\Delta\zeta}{\partial t}\right)^2+q\cdot\left(\frac{\partial\Delta\beta}{\partial t}\right)^2+r\cdot\left(\frac{\partial\Delta\zeta}{\partial t}\right)+s\cdot\left(\frac{\partial\Delta\beta}{\partial t}\right) \\ t\cdot\left(\frac{\partial\Delta\zeta}{\partial t}\cdot\frac{\partial\Delta\beta}{\partial t}\right)+u\cdot\Delta\zeta+v\cdot\Delta\beta+w\cdot(\Delta\zeta\cdot\Delta\beta)+k_1 \end{matrix}\right\} \quad (68)$$

where the roman alphabet letters represent the constant common factors multiplying each of the terms in equations (67 and 68) and k_1, k_2 are independent constants. The F-F model's results showed additional coupling terms, corresponding to those represented by the D.O.F's second time derivatives. This topic will be further discussed in Section 3, where the effect of the blade's curvature (γ) will be evaluated.

Equation setup

Solver ODE45 is able to solve a system of higher order differential equations; as long these are represented as a system of ordinary differential equations. Both, L-F and F-F models, equations (53 and 54), and equations (67 and 68) respectively, contained second order time derivatives. Therefore, with the aim of satisfying the solver's requirements, both systems of equations were converted from a system of two second order differential equations, into a system of four ordinary differential equations. The following example shows the general procedure applied to the L-F model, equations (53 and 54).

Firstly, a change of variable is applied to the system's degrees of freedom (Δζ, Δβ):

$$\rho_1=\Delta\zeta(t) \quad (69)$$

$$\rho_2=\dot{\rho_1} \quad (70)$$

$$\sigma_1=\Delta\beta(t) \quad (71)$$

$$\sigma_2=\dot{\sigma_1} \quad (72)$$

yielding the expressions:

$$\left\{\begin{matrix} a\cdot\frac{\partial\rho_2}{\partial t}=b\cdot(\rho_2)^2+c\cdot(\sigma_2)^2+d\cdot\rho_2+e\cdot\sigma_2 \\ f\cdot(\rho_2\cdot\sigma_2)+g\cdot\rho_1+h\cdot\sigma_1+j\cdot(\rho_1\cdot\sigma_1)+k_1 \end{matrix}\right\} \quad (73)$$

$$\frac{\partial\rho_1}{\partial t}=\rho_2 \quad (74)$$

$$\left\{\begin{matrix} l\cdot\frac{\partial\sigma_2}{\partial t}=m\cdot(\rho_2)^2+n\cdot(\sigma_2)^2+p\cdot\rho_2+q\cdot\sigma_2 \\ r\cdot(\rho_2\cdot\sigma_2)+s\cdot\rho_1+t\cdot\sigma_1+u\cdot(\rho_1\cdot\sigma_1)+k_2 \end{matrix}\right\} \quad (75)$$

$$\frac{\partial\sigma_1}{\partial t}=\sigma_2 \quad (76)$$

which are now in a suitable form for using MATLAB's ODE45 solver. The system (73-76) consists of a set of equations of the following form:

$$M(t,x(t))\cdot\dot{x(t)}=F(t,x(t)) \quad (77)$$

where *M* is the system's mass matrix, *t* is the independent time variable, *x* is a generic time-dependent space variable, and *F* is the right-hand-side of equations from a system of first-order differential algebraic equations (DAEs). As a result, the term representing the system's degrees of freedom in equation (77) is isolated, yielding the final expression, introduced as one of ODE45's arguments:

$$\dot{x(t)}=[M(t,x(t))]^{-1}\cdot[F(t,x(t))] \quad (78)$$

Numerical values for the model's parameters can be found in Appendix E.

Amongst the parameters shown by the parameter table in Appendix E, certain quantities were not directly achieved from the literature, but derived from data provided by it. This process involved certain assumptions which must be highlighted. Firstly, we decided to compare our simple multibody model with a real commercial blade design. The design chosen was the BERP IV, implemented on AW101 helicopters (Titurus 2019). Figure 8, extracted from the same reference's lecture notes, illustrates the design:

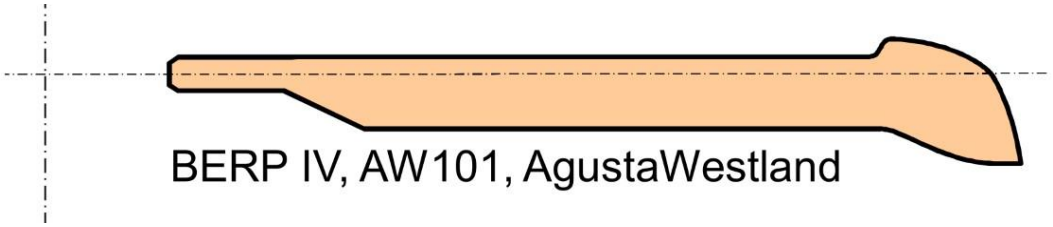


Figure 8 Planform view of the BERP IV design.

Consequently, the blade's span was calculated subtracting the generic hub radius (a), provided by Stroub et al. (1987), from the AW101's rotor radius given by Leonardo S.p.A Company.

$$Blade\ Span = b + c = Rotor\ Radius - Hub\ Radius \tag{79}$$

In addition, Seddon and Newman (2002) state that an analogous BERP blade model has 15% of its span swept back from the outboard, at an angle of approximately 20º. Therefore, the length of segments 1 and 2, (b and c) respectively, were computed with the following expressions:

$$c = 0.15 \cdot Blade\ Span \qquad b = (1 - 0.15) \cdot Blade\ Span \tag{80}$$

With respect to the segment's weights, a linear mass distribution was assumed, in order to provide an educated guess for the mass of each segment in the multibody system. Martínez Santín (2009), states a reasonable average blade weight of 40.55 kg for standard helicopters, basing his work on calculations from Prouty (1986).

The values for the torsional damping coefficients were accomplished with the use of an expression from Titurus (2018), which yields the damping coefficient of a passive damper:

$$c_{\Delta\zeta} = c_{\Delta\beta} = c_1 \cdot {A_p}^2 \tag{81}$$

Where c_1 is the pressure-volumetric flow coefficient and A_p stands for the wetted cross-sectional area of the damper's piston. Moreover, the required data to compute the value of the damping coefficients, was extracted from a case study provided by the same reference.

Parameter	**Units**	**Value**
Laminar Flow Coefficient (c_1)	$kg \cdot m^{-4} \cdot s^{-1}$	$1.0485 \cdot 10^{9}$
Wetted Piston Area (A_p)	m^2	$6.9115 \cdot 10^{-4}$

Table 1 Numerical values for calculating the torsional damping coefficientes, extracted from the case study provided by Titurus (2018).

An important detail to be noticed, are the units of the damping coefficients calculated with equation (81) and the data in Table 1. The units yielded were linear damping coefficient units, (N·m) instead of torsional damping units ($N \cdot m \cdot s \cdot rad^{-1}$). However, information regarding parameters of commercial helicopter dampers was scarce, and the magnitude achieved was assumed as a guide value (≈500 $N \cdot m \cdot s \cdot rad^{-1}$).

Considering the objective of this study is to evaluate the effect of the blade's curvature on its natural frequency spectrum, a set of values for the blade's sweepback angle (γ) was selected from the following range (-45º ≤ γ ≤45º).

3 RESULTS & DISCUSSION

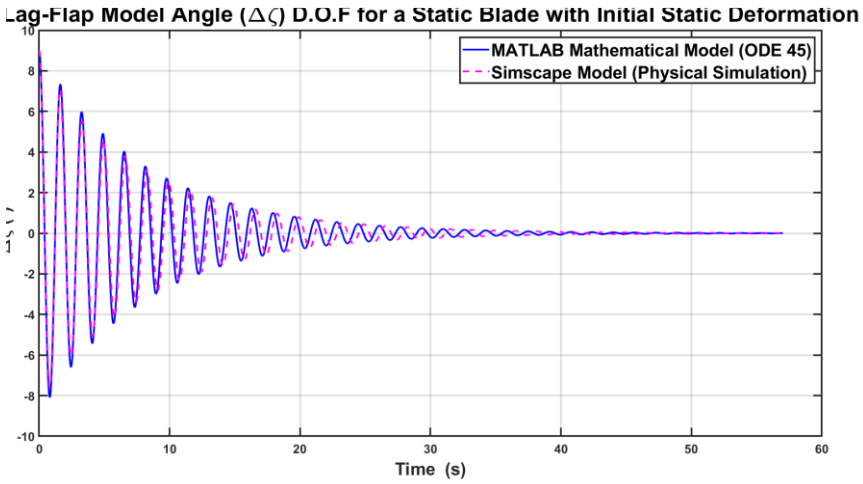
3.1 MATLAB Model Validation

With the purpose of validating our MATLAB multibody model, a Simscape physical simulation was conducted. The validating case, consisted on a static blade response to an initially applied deformation. The blade assessed in this case, had a sweepback angle of –π/4 rad and an applied external excitation, equation (46), at its $\Delta\zeta$ D.O.F; the case corresponded to that represented by equations (49 and 50). Table 2 illustrates the initial conditions applied to both L-F and F-F models:

Parameter	Units	Value
Initial Zeta Position $(\Delta\zeta_0)$	rad	π/20
Initial Beta Position $(\Delta\beta_0)$	rad	-π/20
Initial Zeta Velocity $(\Delta\dot{\zeta}_0)$	rad·s^{-1}	0
Initial Beta Velocity $(\Delta\dot{\beta}_0)$	rad·s^{-1}	0

Table 2 Initial conditions applied to MATLAB's multibody model and Simscape's simulation for model validation.

(a)



(b)

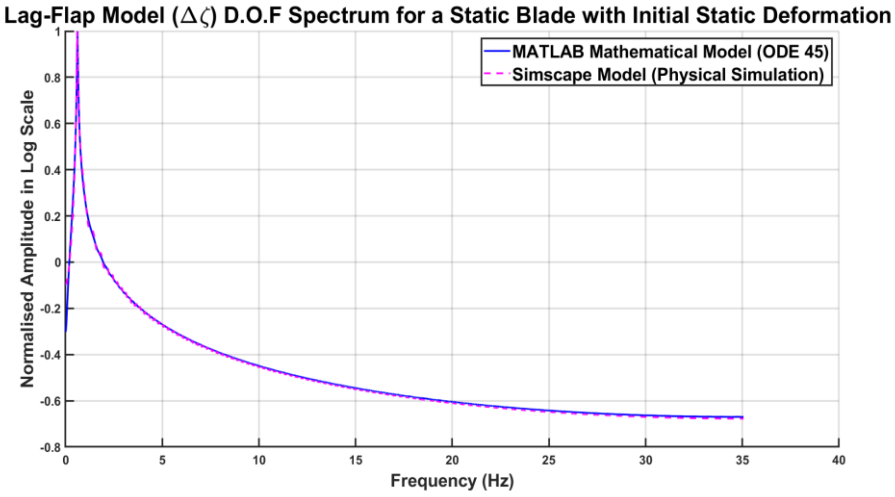
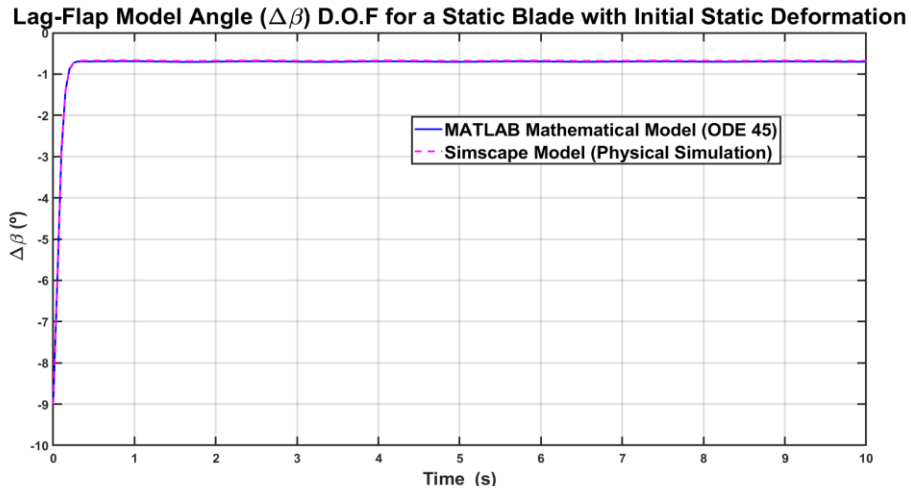


Figure 9 Some plots of the L-F model's (a) $\Delta\zeta$ time response (b) $\Delta\zeta$ frequency spectrum.

(a)



(b)

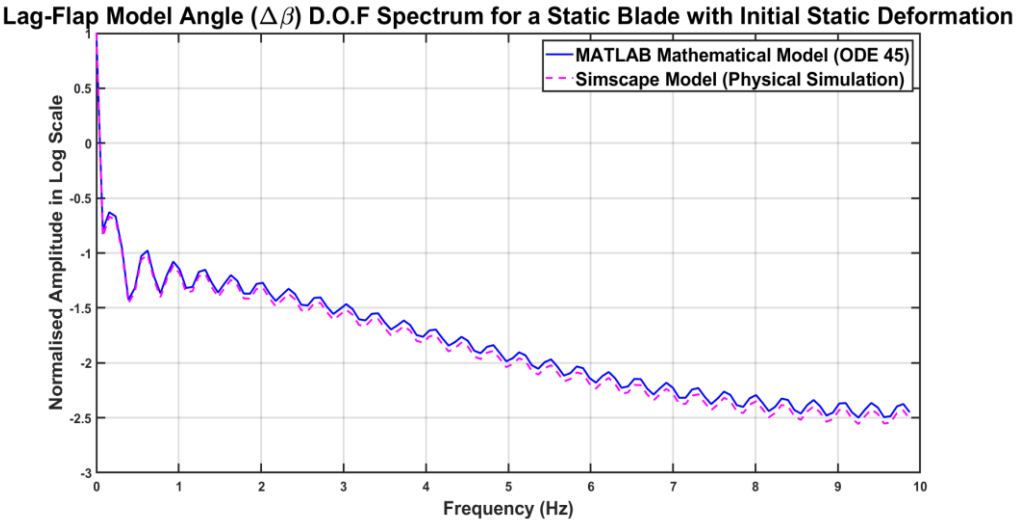


Figure 10 Some plots of the L-F model's (a) Δβ time response (b) Δβ frequency spectrum.

.

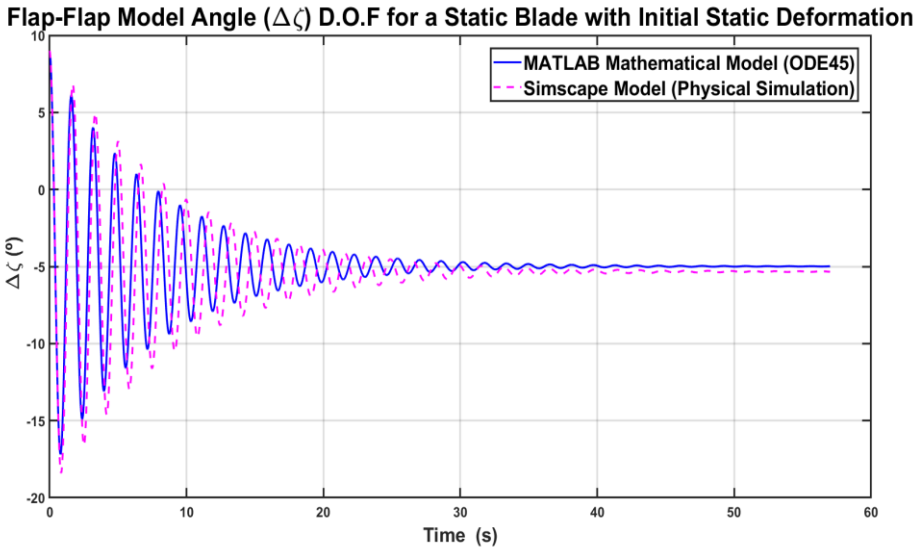


Figure 11 F-F model's Δζ time response.

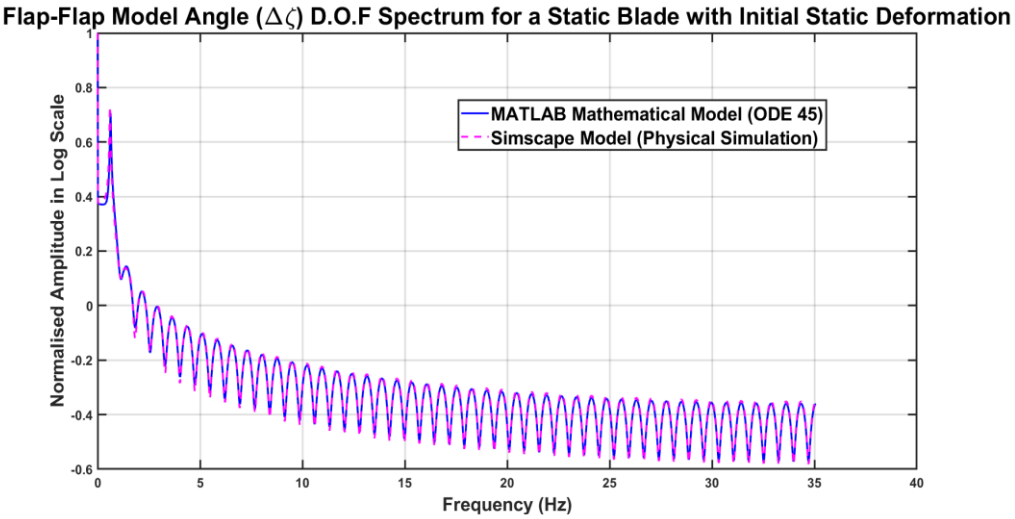
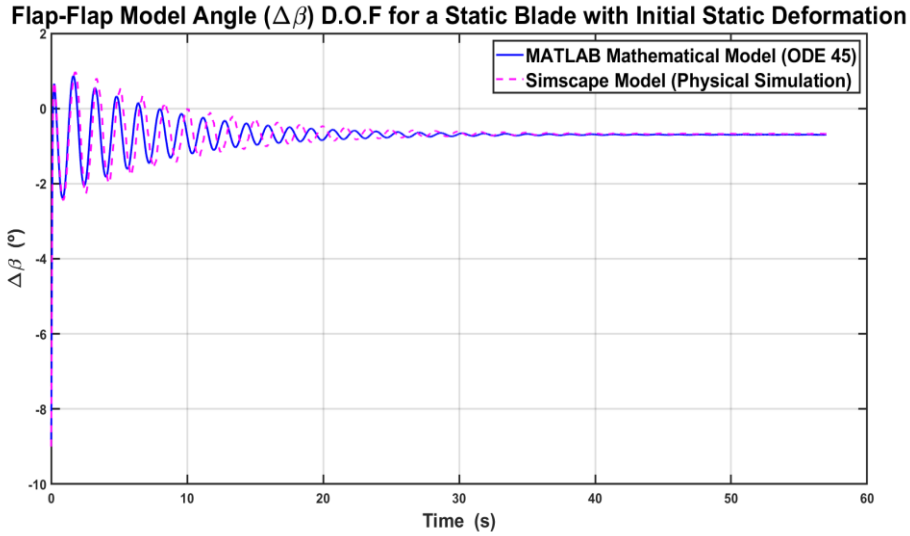


Figure 12 F-F model's Δζ frequency spectrum.

(a)



(b)

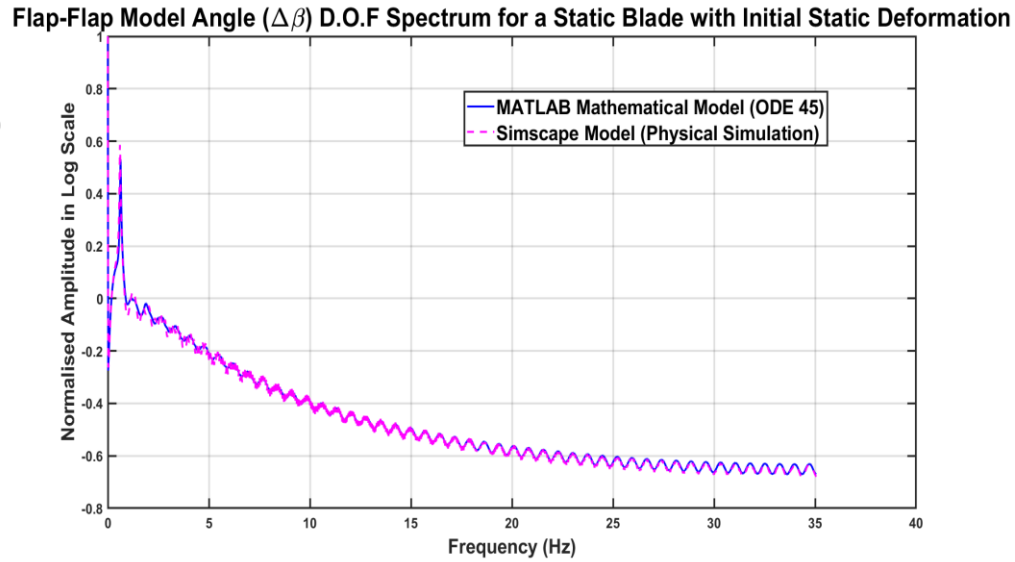


Figure 13 Some plots of the F-F model's (a) Δβ time response (b) Δβ frequency spectrum.

Plots illustrated by Figures (9-13) showed significant coincidence between the mathematical multibody system and the physical simulation performed with Simscape. A curious fact worth mentioning, is that even though both signal's frequency spectra show an almost identical

resemblance, the time responses predicted by Simscape appear to show a slightly higher amplitude and a positive phase shift with respect to the response predicted by the MATLAB model, despite being conducted in the same conditions. Despite achieving suitable validation results with Simscape for this case, the multibody model could not be validated for the helicopter's rotor regime and initial conditions typed in Table 1.

The main reason for the inconsistency amongst MATLAB's and Simscape's results, lies on the fact that Simscape limits the user to solely select initial conditions related to an element's position ($\Delta\zeta_0$, $\Delta\beta_0$). Therefore, considering that results yielded by the system's equations of motion, (e.g. equations 49 and 50), are strongly dependent on the initial conditions considered, validation for these situations was unattainable. In addition, trials were conducted with the rotor regime illustrated by the table in Appendix E; Moreover, zero initial velocity conditions ($\Delta\zeta_0=\Delta\beta_0=0$) were applied, with the aim of verifying if the responses' dissidence, was merely attributable to the initial condition's mismatch. As a result, we achieved a fairly suitable degree of similarity for the spectrums of certain degrees of freedom, but not as satisfactory as for the blade's static case.

As a consequence, the multibody model could not be validated for helicopter operating conditions due to Simscape's limitations. Nevertheless, the mathematical model it was still implemented to assess the effect of the blade's curvature angle (γ) on the blade's frequency spectrum. This fact implies that we assumed that the signal discordance between our mathematical model and Simscape's simulation, was only a consequence of Simscape's unfitness to predict a response for the helicopter's working conditions.

3.2 Lag-Flap Model

The system's response did not present any difference whether the external force applied was at the $\Delta\zeta$ D.O.F (Q_1) or at the $\Delta\beta$ D.O.F (Q_2) on both L-F and F-F models. Therefore, only the results upon Q_1 excitations are displayed. The lack of relevance related to the external force applied on the resulting frequency spectrums, might be attributable to an underestimated magnitude of the force's amplitude (Q_{amp}).

The results achieved for the L-F model's $\Delta\beta$ D.O.F are illustrated by Figures (14-17). A first look reveals that only a single flapping mode was excited with the parameters and simulation conditions implemented. Secondly, the model's first flapping frequency (F_1), experienced a shift towards the lower end of the spectrum with an increasing blade curvature. A table computing the values corresponding to the peaks showed by Figures 15 and 17 can be consulted in Appendix E.

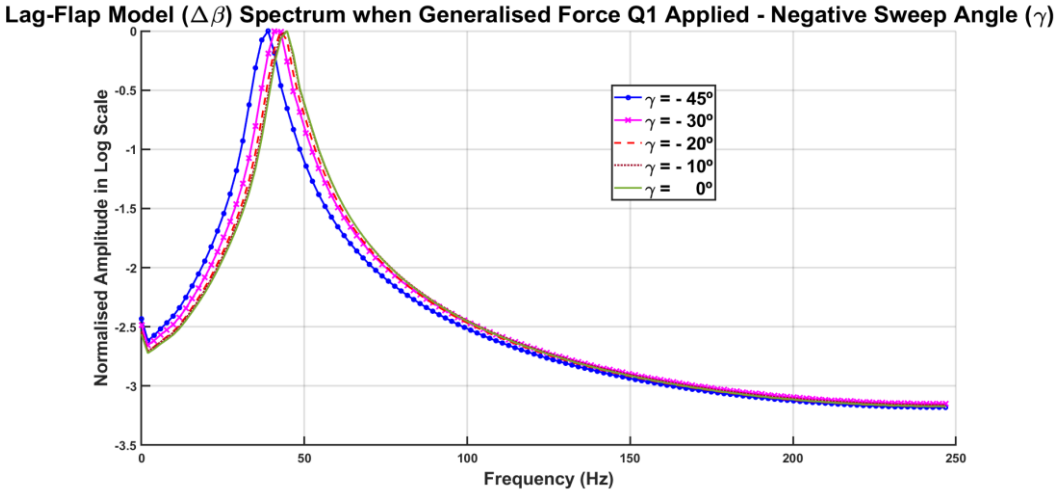


Figure 14 L-F model's $\Delta\beta$ spectrum for negative sweepback angles.

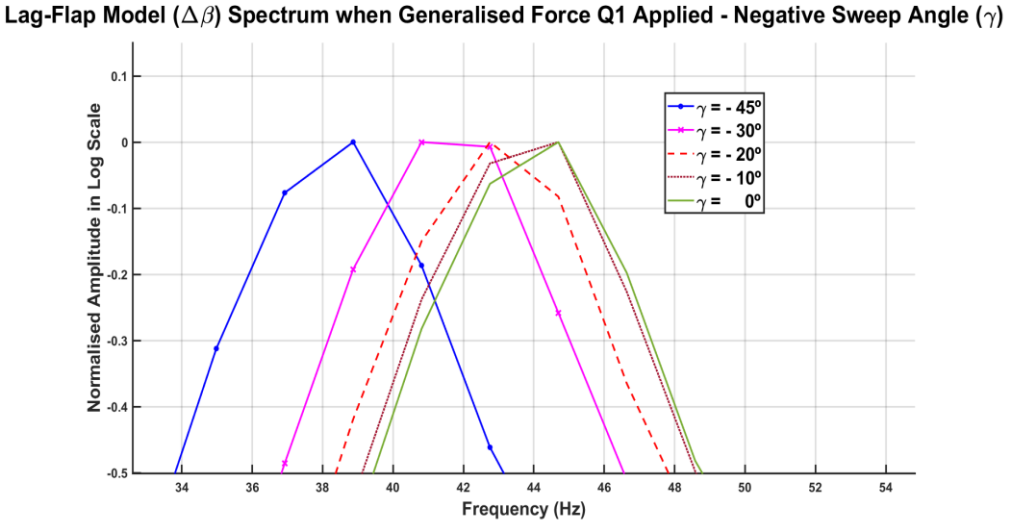


Figure 15 L-F model's Δβ frequency spectrum peaks, first flapping frequency (F_1), for negative curvature angles.

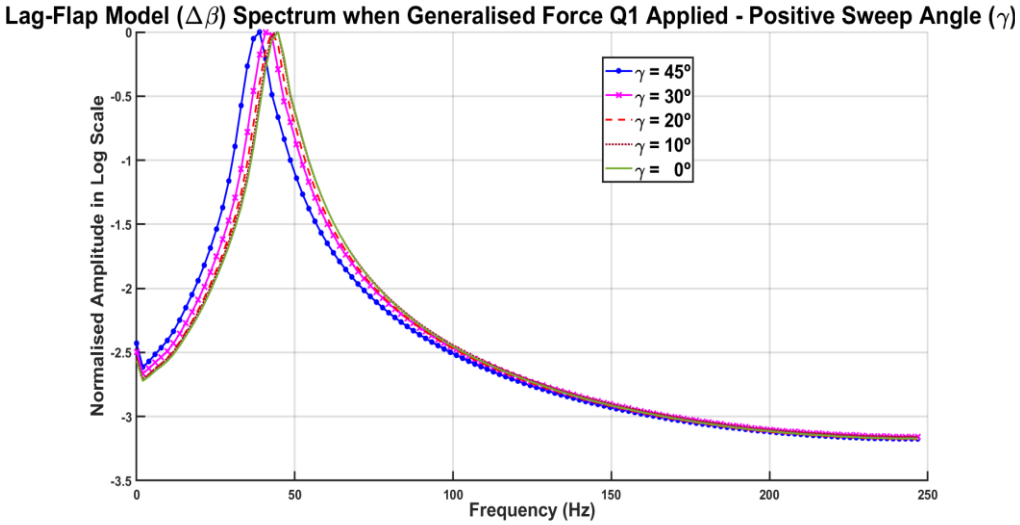


Figure 16 L-F model's Δβ spectrum for positive sweepback angles.

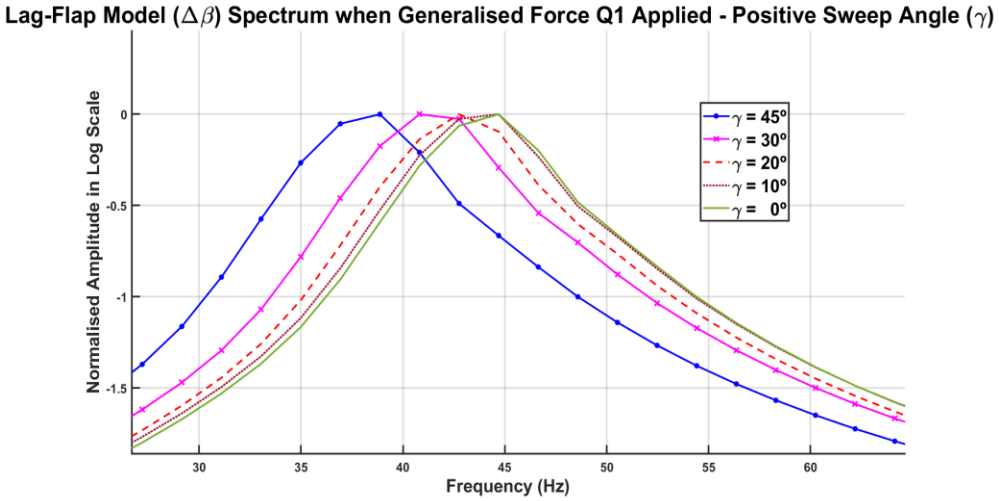


Figure 17 L-F model's Δβ frequency spectrum peaks, first flapping frequency (F_1), for positive curvature angles.

The remaining plots have been omitted for the sake of visibility. These were included in Appendix D of the present document.

In addition, it was noticed that the peak's values did not depend on the angle of curvature's sign, but on its magnitude exclusively. In order to explain this phenomena, the L-F model's equations of motion, (equations 49 and 50), were linearized. Subsequently, the Fourier transform of the expression was taken, (Appendix C), yielding the following frequency response function for the system's ∆β D.O.F:

$$\Delta\beta(\omega) = \frac{g \cdot c}{-c^2 \cdot \omega^2 + c \cdot (a+b+c) \cdot \Omega^2 \cdot \cos(\gamma) - (k_b / M_{seg2})} \tag{82}$$

where ω is the system's natural flapping frequency and the remaining variables have been previously defined. With the aim of accomplishing further insight on the parameters altering the frequency spectrum, an algebraic expression for the denominator's roots in equation 82 was calculated:

$$\omega = \sqrt{\frac{c \cdot (a+b+c) \cdot \Omega^2 \cdot \cos(\gamma) - (k_b / M_{seg2})}{c^2}} \tag{83}$$

From equation 83, light was shed with respect to the spectrum peak's behaviour. Firstly, it shows that the blade's curvature angle (γ) lies within a cosine function. This explains the spectrum's non-dependence on the direction of the blade's curvature, as this is an even function. Moreover, it also provides a reasonable explanation for the progressive decrease of the system's first flapping frequency (F1). Cosine function's values decrease with an increase in the angle lying within its operand. Therefore, the highest frequency value should be expected for a linear-straight blade (γ= 0º), and it should decrease with an increasing angular magnitude. The values yielded by equation 83, related to the model's first flapping frequency (F1), were also typed in a table located in Appendix F. An additional column was added to assess the relative error between the non-linear model (ODE 45) and its linearized version (equation 83).

On the other hand, the model's lead-lag frequency did not show any changes which could result of particular interest. Results can be consulted in Appendix F. Following the same procedure as for the flapping D.O.F (∆β), Lagrange's equations of motion were linearized, and the Fourier transform performed, with the purpose of achieving the lead-lag D.O.F (∆ζ) frequency response:

$$\Delta\zeta(\omega) = \frac{-M_{seg2} \cdot a \cdot c \cdot \Omega^2 \cdot \sin(\gamma)}{-\omega^2 \cdot [M_{seg2} \cdot (b^2 + c^2 + 2 \cdot b \cdot c \cdot \cos(\gamma)) + M_{seg1} \cdot b^2] + M_{seg2} \cdot [a \cdot b \cdot \Omega^2 + a \cdot c \cdot \Omega^2 \cdot \cos(\gamma)] + M_{seg1} \cdot a \cdot b \cdot \Omega^2 - k_z} \tag{84}$$

where ω is the system's natural lead-lag frequency, and the remaining variables are defined in Table 1. The linearized system's lead-lag frequency was computed following the same steps as those applied for equation 83:

$$\omega = \sqrt{\frac{a \cdot \Omega^2 \cdot (b + c \cdot \cos(\gamma) + b \cdot (M_{seg1} / M_{seg2})) - (k_z / M_{seg2})}{b^2 + c^2 + 2 \cdot b \cdot c \cdot \cos(\gamma) + (M_{seg1} / M_{seg2}) \cdot b^2}} \tag{85}$$

Similar to the case for the flapping D.O.F (∆β), the blade's curvature angle lies within a cosine function's operand, explaining why the lead-lag spectrum's remains unaffected from the sweep angle's sign. Additionally, the lead-lag frequency's rate of change with respect to the sweep angle was derived to justify the negligible variation appreciated. The expression, (Appendix C), reflects a stronger dependency on the sweep angle's sine function in comparison to its cosine. This fact could clarify why the lead-lag frequency suffers a slight shift towards the higher range of the spectrum, although a thorough analysis would provide deeper insight.

3.3 Flap-Flap Model

The results accomplished for the F-F model's ∆β D.O.F are illustrated by Figure 11. Contrary to the spectrum yielded by the L-F model, more than a single excitation mode was appreciated for the system. These results were expected, as the equations of motion developed with

Lagrange's expression, (equations 68 and 69), contained a higher number of coupling terms amongst the two degrees of freedom (Δζ and Δβ). Moreover, Figures 11b and 11d picture a magnified image of the spectrum's second flapping frequency (F_2), as it presents a clear view of the frequency dependence on the blade's curvature.

In acordance with the results achieved for the Δβ D.O.F of the L-F model, both of the blade's natural flapping frequencies (F_1 and F_2), undergo a noticeable shift towards the lower end of the frequency spectrum, with an increasing blade curvature. In addition, these were only dependent on the sweep angle's manitude, remaining indifferent to its sign. Table (Appendix F), compute the frequency values achieved for Δβ and Δζ degrees of freedom respectively. Following the same methodology as in Section 3.2, the system's frequency response function was derived:

$$\Delta\beta(\omega) = \frac{M_{seg2}\cdot g\cdot c}{-M_{seg2}\cdot c^2\cdot\omega^2+M_{seg2}\cdot c\cdot\Omega^2\cdot[c+(a+b)\cdot\cos(\gamma)]-k_b} \tag{86}$$

where ω is the system's natural flapping frequency and the remaining variables are defined in Table 1. Consequently, the natural frequencies of the linearized multibody system for the Δβ D.O.F were given by:

$$\omega = \sqrt{\frac{c\cdot\Omega^2\cdot[c+(a+b)\cdot\cos(\gamma)]-(k_b/M_{seg2})}{M_{seg2}\cdot c^2}} \tag{87}$$

Taking into consideration that, in order to achieve equation 87, several non-linearities were ignored, this expression was only able to provide the system's first flapping natural frequency (F_1). For the same reasons exposed in section 3.2, the system's natural frequencies decrease, with an increasing absolute value of the blade's curvature (γ).

Note that in this case, the degrees of freedom (Δζ and Δβ) represent the same type of motion (out-of-plane) of the system. Therefore, the frequency spectrum's corresponding to the Δζ D.O.F were omitted, as they presented equivalent results. In addition, the Δβ D.O.F spectrum plots offered a clearer view of the system's second excitation mode. Finally, the second natural flapping frequencies, accomplished with Figures 18 and 19, were computed in Table 6.

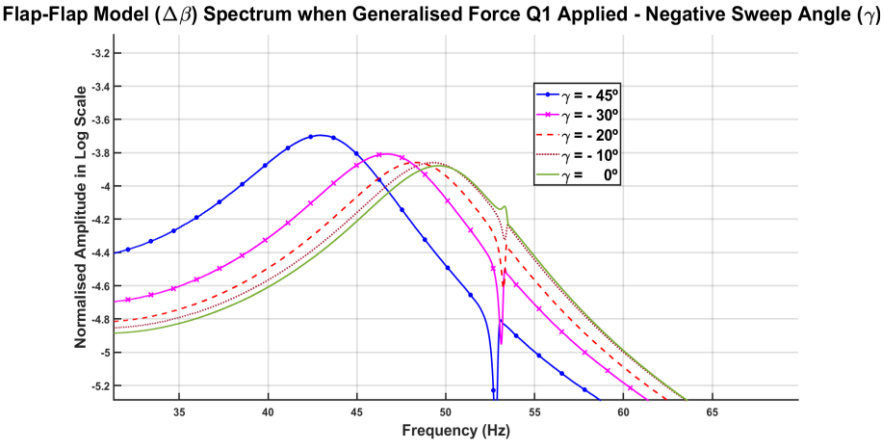


Figure 18 F-F model's Δβ frequency spectrum peaks, second flapping frequency (F_2), for negative curvature angles.

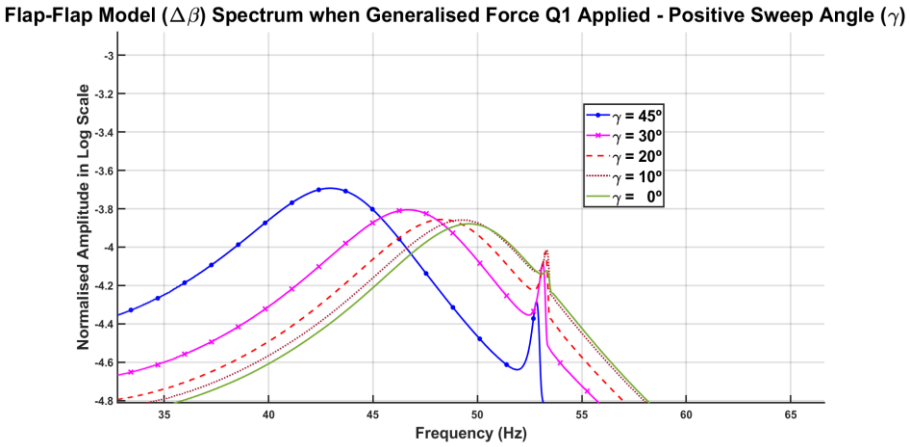


Figure 19 F-F model's Δβ frequency spectrum peaks, second flapping frequency (F_2), for positive curvature angles.

4 CONCLUSION AND FUTURE RESEARCH

The purpose of this study was to examine the capabilities of relatively simple methods to perform a suitable assessment of the frequency spectrum of helicopter blades. On one hand, one of the underlying motivations consisted on testing how many excitations modes would a common MATLAB solver such as ODE 45, be able to predict. On the other hand, verifying the solver's sensitivity to alterations related to the blade's geometry, which are customary of industrial innovations to enhance performance or for commercial purposes.

For multibody models that present a combination of different motions, such as the Lag-Flap model (in-plane and out-of-plane), only a single excitation mode could be appreciated in the resulting frequency spectrum for each degree of freedom. The reason behind this fact, was the reduced number of coupling terms within its equations of motion, linking the system's degrees of freedom. On the other hand, two excitation modes were detected for models with degrees of freedom reflecting a similar motion, such as the Flap-Flap model (out-of-plane and out-of-plane). Concerning the blade's sweep angle, similar effects were observed for the proposed models. The Lag-Flap system did not suffer any noticeable effect on its lead-lag natural frequency (Δζ D.O.F spectrum). Nevertheless, a remarkable decrease on its flapping frequency (Δβ D.O.F spectrum), was perceived with an increasing curvature angle, regardless of its direction. The Flap-Flap system suffered similar effects. This confirmed that for the range of frequencies we were enabled to inspect, only the natural frequencies arising from the blade's flapping motion, were affected by the geometric configurations proposed.

In order to enhance the solver's performance in the future, more realistic damping coefficient values could be implemented on the model's torsional dampers. These values could take into account other effects such as the blade's aerodynamic damping, providing a closer approach to a real helicopter blade. However, this would imply considering an aerofoil model for the blade, as certain properties like the section's moment of inertia and lift coefficient would be needed to compute this effect. In addition, if a particularly thin aerofoil was considered, thin aerofoil theory could be applied along the blade's span. This would enable the user to rapidly compute a relatively simple lift distribution. Consequently, a more realistic approach to the periodic loads suffered by the structure could be implemented, which would be more adequate than an educated guess. Moreover, taking into account that Lagrange's equations are strongly dependent on the initial conditions applied, a thorough selection might significantly improve ODE45's performance. If a scaled prototype of a helicopter blade-rotor system and a set of sensors were available, more realistic values of the blade's boundary conditions could be accomplished by means of dimensional analysis. A future study which might be worth conducting, would be performing a range different Simscape model arrangements, in order to verify the full extent of Simscape's competence for standard helicopter operating conditions.

Additionally, the blade structure could be discretised into a higher number of solid rigid bodies interlinked with more dampers and springs. This, despite being laborious, would allow for a homogeneous distribution of the blade's characteristics, such as its stiffness and damping along its span. Furthermore, the possibility to include other components of the rotor-blade system, such as the blade's pitch-link mechanism might be of interest. As it would provide a suitable approximation to the real rotor-blade system, and an assessment of the spectrum's accuracy could be conducted to verify any improvement accomplished.

REFERENCES

European Space Agency., (2020). Gravity in detail [online]. *ESA*. [Viewed 20 February 2020]. Available from: https://earth.esa.int/web/guest/-/gravity-in-detail-5728

Hahn, B. D. and Valentine, D. T, (2019). Essential MATLAB for Engineers and Scientists. 7th ed. Cambridge MA 02139, United States, Elsevier.

Leonardo S.p.A Company., (2020). AW101.The Superior Solution [online]. *Leonardo S.p.A*. [Viewed 5 April 2020]. Available from: https://www.leonardocompany.com/en/products/aw101

Martínez Santín., (2009). *Diseño preliminar de un helicóptero para aplicaciones civiles. [A preliminary helicopter design for civil Applications].* Universitat Politécnica de Catalunya: J [Viewed 10 February 2020]. Available from: https://upcommons.upc.edu/bitstream/handle/2099.1/9728/Documento%20%20Memoria%20y%20Presupuesto.pdf

MathWorks., (2020). Choose an ODE Solver [online]. MathWorks. [Viewed 14 March 2020]. Available from: https://uk.mathworks.com/help/matlab/math/choose-an-ode-solver.html

MathWorks., (2020). Simulink [online]. MathWorks. [Viewed 14 March 2020]. Available from: https://uk.mathworks.com/help/simulink/

MathWorks., (2020). Simscape [online]. MathWorks. [Viewed 14 March 2020]. Available from: https://uk.mathworks.com/products/simscape.html

MathWorks., (2020). Symbolic Math Toolbox [online]. MathWorks. [Viewed 14 March 2020]. Available from: https://uk.mathworks.com/products/symbolic.html

Oxley, G. et al, (2009). Smart Spring Control of Vibration on Helicopter Rotor Blades [online]. **46**(2), 692–696. [Viewed 10 March 2020]. Available from: doi: 10.2514/1.27914

Prouty, R.W., (1986). *Helicopter Performance, Stability and Control.* 1st ed. Malabar, Florida: Krieger Publishing Company.

Seddon, J. and Newman, S., (2002). *Basic Helicopter Aerodynamics.* 2nd ed. Oxford: Blackwell Science Ltd.

Stroub, R.H. et al., (1987). *Investigation of Generic Hub Fairing and Pylon Shapes to Reduce Hub Drag* [online]. NASA (United States): [Viewed 3 March 2020]. Available from: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19870019989.pdf

Takahasi, M.D. et al., (1990). *A Flight-Dynamic Helicopter Mathematical Model with a Single Flap-Lag Torsion Main Rotor* [online]. NASA (United States): [Viewed 20 April 2020]. Available from: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19910009728.pdf

Titurus, B., (2018). Generalized Liquid-Based Damping Device for Passive Vibration Control. AIAA Journal [online]. 56(10), 4134–4145. [Viewed 5 October 2015]. Available from: doi: 10.2514/1.J056636

Titurus, B., (2019). *Elastic Blade Characteristics* [PowerPoint presentation]. Dynamics of Rotors AENGM0004. 2 October. [Accessed 20 March 2020]. Available from: https://www.ole.bris.ac.uk/bbcswebdav/pid-3946499-dt-content-rid-12482598_2/courses/AENGM0004_2019_TB-1/dor_lec_04_elastic.pdf

Appendix

Appendix A
Implemented MATLAB routine

Appendix B
Simscape Block Table & Simulation Setup

Appendix C
Lagrangian & Frequency Response Calculations

Appendix D
Plots achieved

Appendix E
Numerical Parameter Table

Appendix F
Frequency Tables

Appendix A

Two D.O.F Blade Model - Non Linear

Defining Variables

```
clearvars;
clear all;
clc;
syms a b c M_hub M_seg1 M_seg2 kz cz kb cb Omega g A Qamp srate real
syms t gamma z10 b20 dz10 db20 real
syms ze(t) be(t) Q1(t) Q2(t)
d1ze = diff(ze,t);
d1be = diff(be,t);
d2ze = diff(ze,t,2);
d2be = diff(be,t,2);
```

Case Selection & Position Equations

Case Selection

```
 system_id = 'LF'; % Lag-Flap System
% system_id = 'FF'; % Flap-Lag System


switch system_id
    case 'LF'


% Position Equations for 'LF'


% Blade root
x1 =  a*sin(Omega*t);
y1 = -a*cos(Omega*t);
z1  =  0;


% First Segment (mass & hinge)


% Mass
x2 =  b*sin(Omega*t+ze);
y2 = -b*cos(Omega*t+ze);
z2  =  0;


% Hinge
```

```
x2h =  A*sin(Omega*t+ze);
y2h = -A*cos(Omega*t+ze);
z2h =  0;


% Second Segment


x3 =  c*cos(be)*sin(Omega*t+ze-gamma);
y3 = -c*cos(be)*cos(Omega*t+ze-gamma);
z3 =  c*sin(be);


    case 'FF'


%  Blade root
x1= a*sin(Omega*t);
y1=-a*cos(Omega*t);
z1=0;


% First Segment (mass and hinge)


% Mass
x2=  b*cos(ze)*sin(Omega*t);
y2= -b*cos(ze)*cos(Omega*t);
z2=  b*sin(ze);


% Hinge
x2h= A*cos(ze)*sin(Omega*t);
y2h=-A*cos(ze)*cos(Omega*t);
z2h= A*sin(ze);




% Second Segment
x3=  c*cos(be)*sin(Omega*t-gamma);
y3= -c*cos(be)*cos(Omega*t-gamma);
z3=  c*sin(be);


end
```

Position at the end of the Hub

```
x_hub = x1;
y_hub = y1;
z_hub = z1;
```

Position at the end of Segment 1

```
x_seg1 = x1 + x2;
y_seg1 = y1 + y2;
z_seg1 = z1 + z2;
```

Position at the end of Segment 2

```
x_seg2 = x1 + x2h + x3;
y_seg2 = y1 + y2h + y3;
z_seg2 = z1 + z2h + z3;
```

Velocity Equations

Velocity at the end of the Hub

```
dx_hub = diff(x_hub,t);
dy_hub = diff(y_hub,t);
dz_hub = diff(z_hub,t);
```

Velocity at the end of Segment 1

```
dx_seg1 = diff(x_seg1,t);
dy_seg1 = diff(y_seg1,t);
dz_seg1 = diff(z_seg1,t);
```

Velocity at the end of Segment 2

```
dx_seg2 = diff(x_seg2,t);
dy_seg2 = diff(y_seg2,t);
dz_seg2 = diff(z_seg2,t);
```

Kinetic Energy Equations

Kinetic Energy at the end of the Hub

```
Ek_hub = 0.5*M_hub*(dx_hub^2+dy_hub^2+dz_hub^2);
```

Kinetic Energy at the end of Segment 1

```
Ek_seg1 = 0.5*M_seg1*(dx_seg1^2+dy_seg1^2+dz_seg1^2);
```

Kinetic Energy at the end of Segment 2

```
Ek_seg2 = 0.5*M_seg2*(dx_seg2^2+dy_seg2^2+dz_seg2^2);
```

System's Total Kinetic Energy

```
Ek = Ek_hub + Ek_seg1 + Ek_seg2;
```

Potential Energy Equations

Potential Energy of the Lead-Lag Torsional Spring

```
Ep_seg1 = 0.5*kz*ze^2 + M_seg1*g*z_seg1;
```

Potential Energy of the Flapping Torsional Spring

```
Ep_seg2 = 0.5*kb*be^2 + M_seg2*g*z_seg2;
```

System's Total Potential Energy

```
Ep = Ep_seg1 + Ep_seg2;
```

Energy Dissipation Functions

Lead-Lag & Flapping Damping Function

```
D = 0.5*cz*d1ze^2 + 0.5*cb*d1be^2;
```

Lagrangian

```
L=Ek-Ep;
simplify(L);
```

Lagrange's Equation

```
LHS=functionalDerivative(L,[ze,be]);
dD_dqi=[cz*d1ze;cb*d1be];   % Manual diff of D
Q1=Qamp*sin(srate*t^2);
Q2=0;
LHS=-LHS-[Q1;Q2]*1+dD_dqi;  % Setting up the correct signs
simplify(LHS);
```

Space Model

```
[eqs,vars]=reduceDifferentialOrder(LHS,[ze,be]);
[M,F]=massMatrixForm(eqs,vars);
fs=M\F;
```

Numeric Parameters

Structural Parameters

```
a=0.381;                    % In m
b=7.162;                    % In m
A =7.362;                   % In m
c=1.367;                    % In m
g=9.81;                     % In m/s^2
M_hub=36.26;                % In kg
M_seg1=31.88;               % In kg
M_seg2=6.08;                % In kg
Omega= 109.22*1;              % In rad/s
kz=30659;                    % In Nm/(rad)
```

```
kb=6691;                        % In Nm/(rad)
cz= 500;                         % In Nm/(rad/s)
cb= 500;                         % In Nm/(rad/s)
gamma= -pi/18*0;                    % In rad
Qamp=15*4;                      % In (N)
srate=2;                        % [-]
```

Initial Conditions

```
z10 =      pi/20;
b20 =     -pi/20;
dz10 =          1;
db20 =         -1;
```

Initial Conditions (deg)

```
% z10


fprintf(['The initial condition position for the lead-lag angle z10 is %.3f rads' ...
    ' which is %.3f degrees'],z10,z10*(180/pi))
% b20


fprintf(['The initial condition position for the flapping angle b20  is %.3f rads' ...
    ' which is %.3f degrees'],b20,b20*(180/pi))
% dz10


fprintf(['The initial condition velocity for the lead-lag angle dz10 is %.3f rads' ...
    ' which is %.3f degrees'],dz10,dz10*(180/pi))
% db20


fprintf(['The initial condition velocity for the flapping angle db20 is %.3f rads' ...
    ' which is %.3f degrees'],db20,db20*(180/pi))
```

Numeric Substitutions and ODE 45 Equation Setup

```
fs=subs(fs);
odeFs=odeFunction(fs,vars);
```

ODE 45 Solver

```
t_start = 0;
```

```
t_end = 55;
samples = 2e3;
tspan=linspace(t_start,t_end,samples);
t_step = tspan(2)-tspan(1);
x0=[z10;b20;dz10;db20]; %x0=[(rand(2,1)*2-1)*pi/20;(rand(2,1)*2-1)*0.1];
[T,X]=ode45(odeFs,tspan,x0);
```

Graphs

4.1.1.1 $(\Delta\zeta)$ D.O.F Plot

```
X(:,1) = X(:,1)*(180/pi); % Conversion from Radians to Degrees
figure,plot(T,X(:,1),'b','LineWidth',2); hold on,
title(sprintf('Angle Zeta for Omega = %.3f (rad/s)',Omega))
xlabel('time(s)');
ylabel('angle zeta (deg)'),grid
hold off
```

4.1.1.2 $(\Delta\beta)$ D.O.F Plot

```
X(:,2) = X(:,2)*(180/pi); % Conversion from Radians to Degrees
figure, plot(T,X(:,2),'r','LineWidth',2); hold on;
title(sprintf('Angle Beta for Omega = %.3f (rad/s)',Omega))
xlabel('time(s)');
ylabel('angle beta (deg)');
grid
hold off
% Postprocessing
figure, box on, grid on, hold on
plot(T,X(:,1:2))
legend('1','2')
title('Results'), hold off
```

Data Post-Processing

*FFT applied to the (**Δζ**) D.O.F signal*

```
% Converting Delta-Zeta signal from (deg) to (rad)


FFT_X1= X(:,1)*(pi/180); % Angle
FFT_T = T; % Time


% Visualising Delta-Zeta Signal in Time Domain
figure, plot(FFT_T,FFT_X1,'m','LineWidth',2), hold on, grid
xlabel('Time (s)');
ylabel('Lead-Lag angle (rad)');
title(sprintf('Angle Zeta for Omega = %.3f (rad/s)',Omega))
```

```
hold off

% Applying the Fast Fourier Transform
fft_points_ze = 2^nextpow2(size(FFT_X1,1));
frequency_spectrum_ze = fft(FFT_X1,fft_points_ze);
frequency_spectrum_ze = frequency_spectrum_ze(1:end/2,:);
frequency_step_ze = 1/(fft_points_ze*t_step);
frequency_ze = ([1:fft_points_ze/2]-1)*frequency_step_ze;


% Normalising the spectrum with respect to the amplitude's maximum value


normalised_ze = abs(frequency_spectrum_ze/max(frequency_spectrum_ze));
normalised_ze = log(normalised_ze);




% Visualising Delta-Zeta Signal in Frequency Domain
figure,          plot(frequency_ze,normalised_ze,'Color',[0.9290          0.6940
0.1250],'LineWidth',2),hold on
grid
xlabel('Frequency (Hz)');
ylabel('Normalised Amplitude in Log Scale');
title(sprintf('Frequency Spectrum in Hertz of Lead-Lag Angle for Gamma = %.3f
(deg)',gamma*-180/pi))


hold off
```

*FFT applied to the (**Δβ**) D.O.F signal*

```
% Converting Delta-Beta signal from (deg) to (rad)



FFT_X2= X(:,2)*(pi/180); % Angle
FFT_T = T; % Time



% Visualising Delta-Beta Signal in Time Domain
figure, plot(FFT_T,FFT_X2,'g','LineWidth',2), hold on, grid
xlabel('Time (s)');
ylabel('Flapping angle (rad)');
title(sprintf('Angle Beta for Omega = %.3f (rad/s)',Omega))
hold off
% Applying the Fast Fourier Transform
fft_points_be = 2^nextpow2(size(FFT_X2,1));
frequency_spectrum_be = fft(FFT_X2,fft_points_be);
frequency_spectrum_be = frequency_spectrum_be(1:end/2,:);
frequency_step_be = 1/(fft_points_be*t_step);
frequency_be = ([1:fft_points_be/2]-1)*frequency_step_be;
```

```
% Normalising the spectrum with respect to the amplitude's maximum value
normalised_be = abs(frequency_spectrum_be/max(frequency_spectrum_be));
normalised_be = log(normalised_be);



% Visualising Delta-Beta Signal in Frequency Domain
figure,          plot(frequency_be,normalised_be,'Color',[0.6350          0.0780
0.1840],'LineWidth',2),hold on
grid
xlabel('Frequency (Hz)');
ylabel('Normalised Amplitude in Log Scale');
title(sprintf('Frequency Spectrum in Hertz of Flapping Angle for Gamma = %.3f
(deg)',gamma*-180/pi))
hold off
```

Simscape Model Parameter Setup [Lag-Flap Model]

Parameter Format Conversion from 'double' to 'string'

```
% Here I am converting the Numeric Parameters to a String value because
% the command 'set_param' allows me to substitute those parameters into
% my Simscape model. I am only substituting the
% parameters below into the model, the rest of them have been typed
% manually.


kz=      num2str(kz);
kb=      num2str(kb);
cz=      num2str(cz);
cb=      num2str(cb);
M_hub=   num2str(M_hub);
M_seg1=  num2str(M_seg1);
M_seg2=  num2str(M_seg2);
gamma =  num2str(gamma);
z10 = num2str(z10);
b20 = num2str(b20);
dz10 = num2str(dz10);
db20 = num2str(db20);
k_hub = (200e8);
k_hub = num2str(k_hub);
c_hub = (8e8);
c_hub = num2str(c_hub);
```

Parameter Substitution in Simscape Lag-Flap Model

```
% Joint 1


% Internal Mechanics
set_param('LF_Final_Simscape_Model/RevoluteJoint1','SpringStiffness',k_hub);


set_param('LF_Final_Simscape_Model/RevoluteJoint1','DampingCoefficient',c_hub;



% Joint 2


% Internal Mechanics
set_param('LF_Final_Simscape_Model/RevoluteJoint2','SpringStiffness',kz);


set_param('LF_Final_Simscape_Model/RevoluteJoint2','DampingCoefficient',cz);


set_param('LF_Final_Simscape_Model/RevoluteJoint2','PositionTargetValue',z10);



% Joint 3


% Internal Mechanics
set_param('LF_Final_Simscape_Model/RevoluteJoint3','SpringStiffness',kb);


set_param('LF_Final_Simscape_Model/RevoluteJoint3','DampingCoefficient',cb);


set_param('LF_Final_Simscape_Model/RevoluteJoint3','PositionTargetValue',b20);



% Mass of the Hub


set_param('LF_Final_Simscape_Model/SphericalSolid1','Mass',M_hub);


% Mass of Segment 1


set_param('LF_Final_Simscape_Model/SphericalSolid2','Mass',M_seg1);
```

```
% Mass of Segment 2

set_param('LF_Final_Simscape_Model/SphericalSolid3','Mass',M_seg2);
```

Simscape Model Parameter Setup [Flap-Flap Model]

Parameter Format Conversion from 'double' to 'string'

```
% Here I am converting the Numeric Parameters to a String value because
% the command 'set_param' allows me to substitute those parameters into
% my Simscape model. I am only substituting the
% parameters below into the model, the rest of them have been placed
% manually.


kz=      num2str(kz);
kb=      num2str(kb);
cz=      num2str(cz);
cb=      num2str(cb);
M_hub=   num2str(M_hub);
M_seg1=  num2str(M_seg1);
M_seg2=  num2str(M_seg2);
gamma =  num2str(gamma);
z10 = num2str(z10);
b20 = num2str(b20);
dz10 = num2str(dz10);
db20 = num2str(db20);
k_hub = (200e8);
k_hub = num2str(k_hub);
c_hub = (8e8);
c_hub = num2str(c_hub);
```

Parameter Substitution in Simscape Flap-Flap Model

```
% Joint 1


% Internal Mechanics
set_param('FF_Final_Simscape_Model/RevoluteJoint1','SpringStiffness',k_hub);
```

```
set_param('FF_Final_Simscape_Model/RevoluteJoint1','DampingCoefficient',c_hub;

% Joint 2

% Internal Mechanics
set_param('FF_Final_Simscape_Model/RevoluteJoint2','SpringStiffness',kz);

set_param('FF_Final_Simscape_Model/RevoluteJoint2','DampingCoefficient',cz);

set_param('FF_Final_Simscape_Model/RevoluteJoint2','PositionTargetValue',z10);

% Joint 3

% Internal Mechanics
set_param('FF_Final_Simscape_Model/RevoluteJoint3','SpringStiffness',kb);

set_param('FF_Final_Simscape_Model/RevoluteJoint3','DampingCoefficient',cb);

set_param('FF_Final_Simscape_Model/RevoluteJoint3','PositionTargetValue',b20);

% Mass of the Hub

set_param('FF_Final_Simscape_Model/SphericalSolid1','Mass',M_hub);

% Mass of Segment 1

set_param('FF_Final_Simscape_Model/SphericalSolid2','Mass',M_seg1);



% Mass of Segment 2

set_param('FF_Final_Simscape_Model/SphericalSolid3','Mass',M_seg2);
```

Exporting Signals From Simscape to Matlab Workspace

```
% Extracting Simscape's exported values from the Structure 'out'

Simscape_time = out.zeta(:,1);
```

```
Simscape_zeta = out.zeta(:,2);
Simscape_beta = out.beta(:,2);


% Plot to check if the Exported data is correct


% Simscape Delta-Zeta
figure, plot(Simscape_time,Simscape_zeta), grid, hold on
title(sprintf('Simscape Zeta Angle for Gamma = %.3f',gamma*-180/pi))
xlabel('Time (s)')
ylabel('Zeta angle (deg)')
hold off



% Simscape Delta-Beta
figure, plot(Simscape_time,Simscape_beta), grid, hold on
title(sprintf('Simscape Beta Angle for Gamma = %.3f',gamma*-180/pi))
xlabel('Beta angle (deg)')
ylabel('Time (s)')
hold off
```

Simscape Signal Post-Processing

Interpolation (Spline) for $(\Delta\zeta)$ *D.O.F*

```
int_Sim_zeta = interp1(Simscape_time,Simscape_zeta,tspan,'spline');



% Auxiliary Check Plot
figure,plot(tspan,int_Sim_zeta,'b','LineWidth',2); hold on,
title(sprintf('Interpolated Simscape Angle Zeta for Gamma = %.3f (rad/s)',gamma*-180/pi))
xlabel('time(s)');
ylabel('angle zeta (deg)'),grid
hold off
```

Interpolation (Spline) for $(\Delta\beta)$ *D.O.F*

```
int_Sim_beta = interp1(Simscape_time,Simscape_beta,tspan,'spline');



% Auxiliary Check Plot
figure,plot(tspan,int_Sim_beta,'r','LineWidth',2); hold on,
title(sprintf('Interpolated Simscape Angle Beta for Omega = %.3f (rad/s)',Omega))
xlabel('time(s)');
ylabel('angle beta (deg)'),grid
hold off
```

Fast Fourier Transform to Simscape Signal

FFT applied to the Simscape (Δζ) D.O.F signal

```
% Defining


FFT_SL= int_Sim_zeta*(pi/180); % Angle
FFT_SL = FFT_SL';
FFT_ST = tspan'; % Time


% Applying the Fast Fourier Transform
fft_points = 2^nextpow2(size(FFT_SL,1));
frequency_spectrum = fft(FFT_SL,fft_points);
frequency_spectrum = frequency_spectrum(1:end/2,:);
frequency_step = 1/(fft_points*t_step);
frequency = ([1:fft_points/2]-1)*frequency_step;


% Normalising the spectrum with respect to the amplitude's maximum value
normalised = abs(frequency_spectrum);
normalised = log(normalised);
maximum = max(normalised);
normalised = normalised./maximum;




% Visualising Simscape Delta-Zeta Signal in Frequency Domain
figure,            plot(frequency,normalised,'Color',[0.4660            0.6740
0.1880],'LineWidth',2),hold on
grid
xlabel('Frequency (Hz)');
ylabel('Amplitude');
title(sprintf('Spectrum in (Hz) Int Simscape Lead-Lag Angle for Omega = %.3f
(rad/s)',Omega))
hold off
```

4.1.1.3 FFT applied to the Simscape (Δβ) D.O.F signal

```
% Converting signal from (deg) to (rad)


FFT_SF= int_Sim_beta*(pi/180); % Angle
FFT_SF = FFT_SF';
FFT_ST = tspan'; % Time
```

```
% Applying the Fast Fourier Transform
fft_points = 2^nextpow2(size(FFT_SF,1));
frequency_spectrum = fft(FFT_SF,fft_points);
frequency_spectrum = frequency_spectrum(1:end/2,:);
frequency_step = 1/(fft_points*t_step);
frequency = ([1:fft_points/2]-1)*frequency_step;



% Normalising the spectrum with respect to the amplitude's maximum value
normalised = abs(frequency_spectrum);
normalised = log(normalised);
maximum = max(normalised);
normalised = normalised./maximum;


% Visualising Delta-Beta Signal in Frequency Domain
figure,            plot(frequency,normalised,'Color',[0.4940            0.1840
0.5560],'LineWidth',2),hold on
grid
xlabel('Frequency (Hz)');
ylabel('Amplitude');
title(sprintf('Frequency Spectrum in (Hz) Int Simscape Flapping Angle for Omega
= %.3f (rad/s)',Omega))
hold off
```

Appendix B

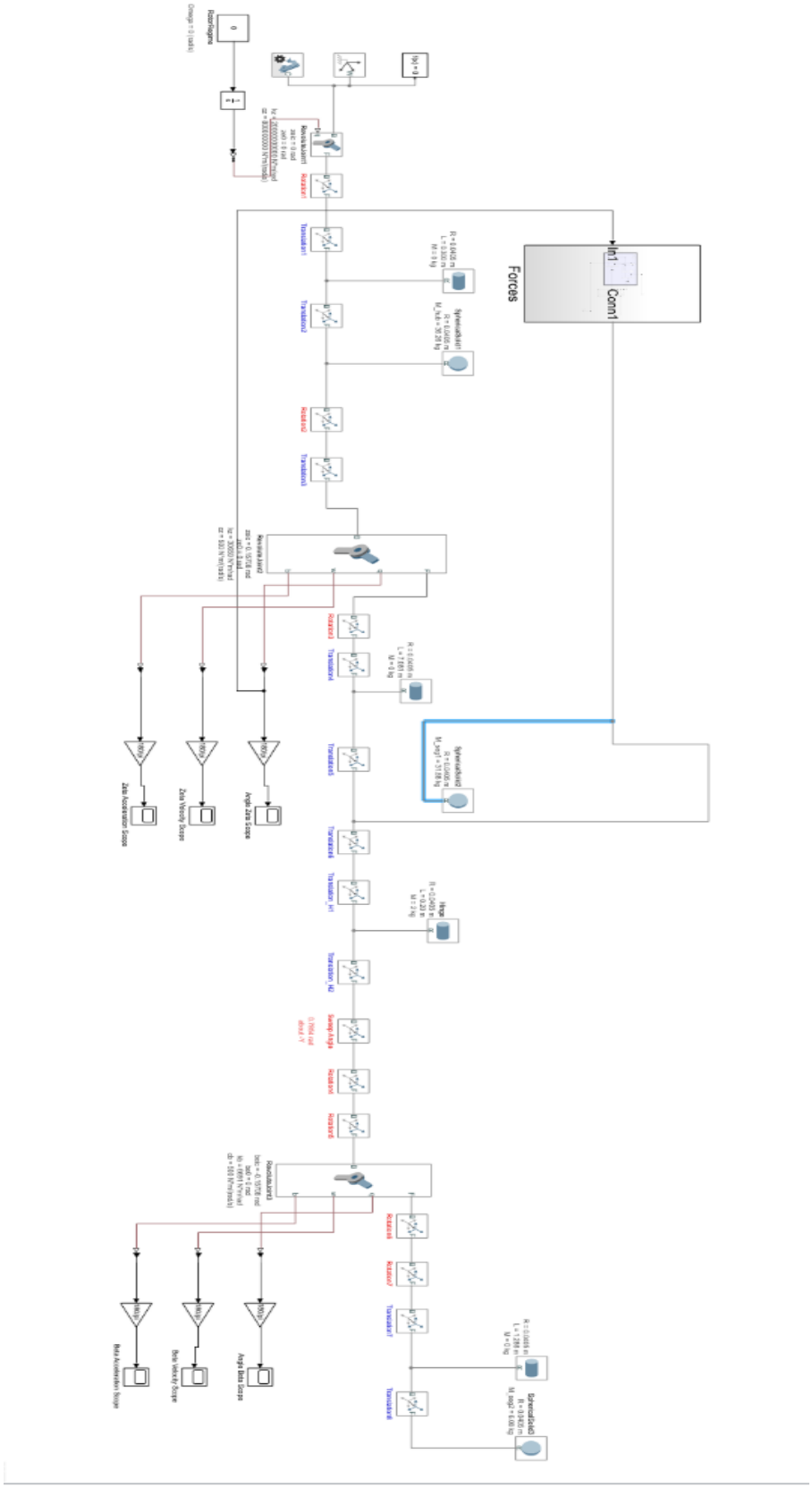
Lag-Flap Model - Simscape Blocks

Simscape Blocks Applied in the Blade Physical Simulation	
Block Name	**Number of Blocks**
Rigid Transform	18
Cylindrical Solid	4
Spherical Solid	3
Solver Configuration	1
World Frame	1
Mechanism Configuration	1
Constant	6
Product	7
Add	1
Integrator	1
Gain	8
Sin	3
Cos	1
Clock	3
Square	2
Simulink - Ps Converter	9
Scope	12
External Force and Torque	2
Out	3

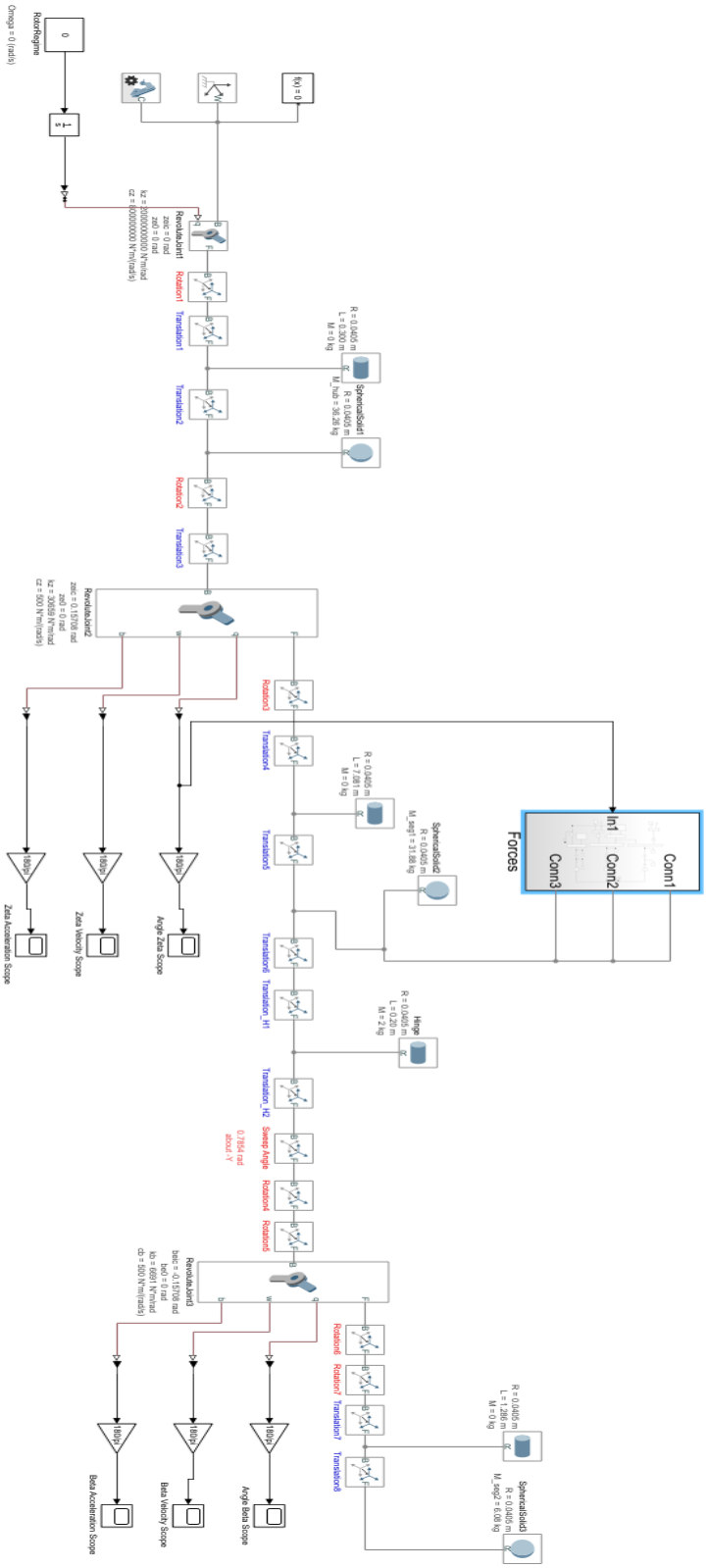
Flap-Flap Model - Simscape Blocks

Simscape Blocks Applied in the Blade Physical Simulation	
Block Name	**Number of Blocks**
Rigid Transform	18
Cylindrical Solid	4
Spherical Solid	3
Solver Configuration	1
World Frame	1
Mechanism Configuration	1
Constant	7
Product	10
Add	0
Integrator	1
Gain	9
Sin	4
Cos	2
Clock	4
Square	2
Simulink - Ps Converter	10
Scope	12
External Force and Torque	3
Out	3

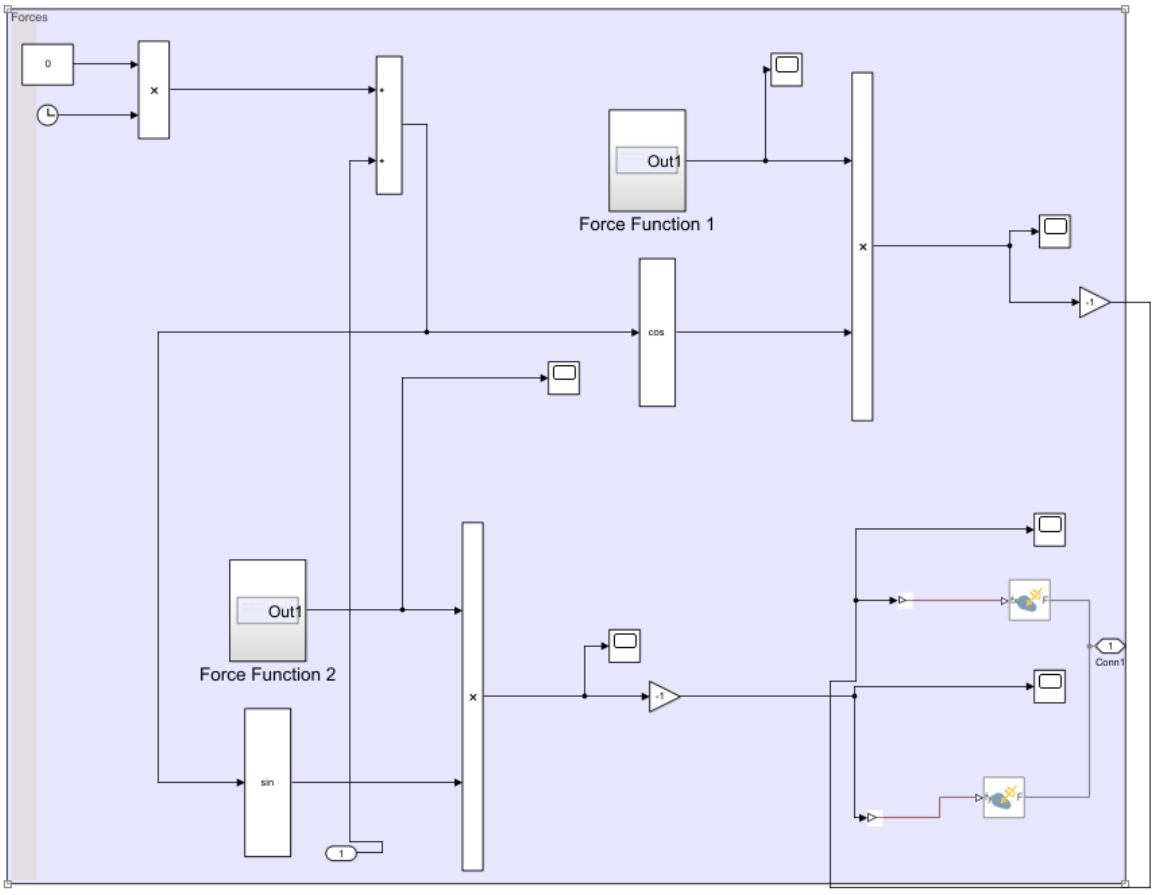
Lag-Flap Model Scheme



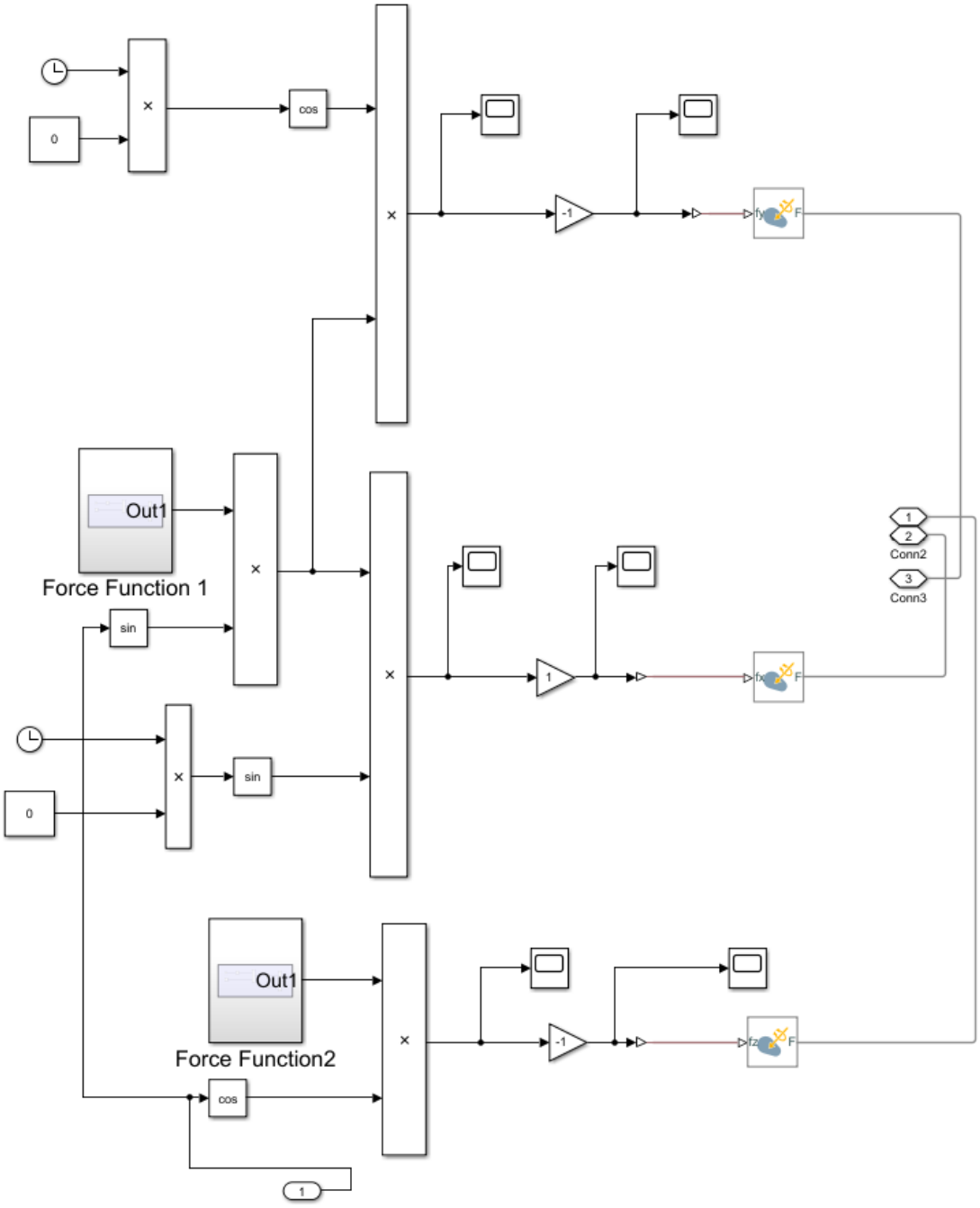
Flap-Flap Model Scheme



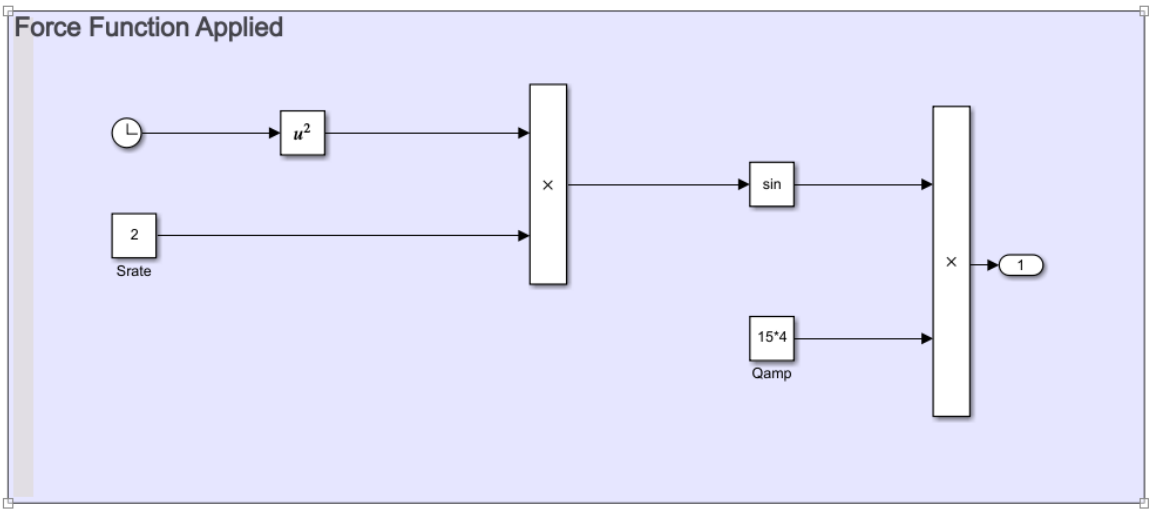
The Lag-Flap Model Force is decomposed (x, y and z axis):



The Flap-Flap Model Force is decomposed (x, y and z axis)



Force Function Scheme applied in both Models:



Appendix C

Lag-Flap Model Lagrangian

$$\frac{\partial}{\partial t}\left[\frac{\partial L}{\partial \Delta\dot{\zeta}}\right]+\left[\frac{\partial L}{\partial \Delta\zeta}\right]=$$

$$=\Delta\ddot{\zeta}\left[M_2\left(b^2+c^2\cos^2\left(\beta_o+\Delta\beta\right)+2bc\cos\left(\beta_o+\Delta\beta\right)\cos\left(\zeta_o-\gamma\right)\right)+M_1b^2\right]+$$

$$+\Delta\ddot{\beta}\left[M_2bc\sin\left(\beta_o+\Delta\beta\right)\sin\left(\zeta_o-\gamma\right)\right]+$$

$$+M_2\begin{bmatrix}-2c^2\left(\Omega+\Delta\dot{\zeta}\right)\Delta\dot{\beta}\cos\left(\beta_o+\Delta\beta\right)\sin\left(\beta_o+\Delta\beta\right)-ab\Omega\Delta\dot{\zeta}\sin\left(\zeta_o+\Delta\zeta\right)-\\-ac\Omega\Delta\dot{\beta}\sin\left(\beta_o+\Delta\beta\right)\cos\zeta-ac\Omega\Delta\dot{\zeta}\cos\left(\beta_o+\Delta\beta\right)\sin\left(\gamma+\Delta\zeta\right)+\\+bc\Delta\dot{\beta}^2\cos\left(\beta_o+\Delta\beta\right)\sin\left(\zeta_o-\gamma\right)-2bc\left(\Omega+\Delta\dot{\zeta}\right)\Delta\dot{\beta}\sin\left(\beta_o+\Delta\beta\right)\cos\left(\zeta_o-\gamma\right)+\\+ab\Omega\left(\Omega+\Delta\dot{\zeta}\right)\sin\left(\zeta_o+\Delta\zeta\right)+ac\Omega\Delta\dot{\beta}\sin\left(\beta_o+\Delta\beta\right)\cos\left(\gamma+\Delta\zeta\right)+\\+ac\Omega\left(\Omega+\Delta\dot{\zeta}\right)\cos\left(\beta_o+\Delta\beta\right)\sin\left(\gamma+\Delta\zeta\right)\end{bmatrix}+$$

$$+M_1\left[-ab\Omega\Delta\dot{\zeta}\sin\left(\zeta_o+\Delta\zeta\right)+ab\Omega\left(\Omega+\Delta\dot{\zeta}\right)\sin\left(\zeta_o+\Delta\zeta\right)\right]-k_z\Delta\zeta=0$$

$$\Delta\ddot{\zeta}\left[M_2\left(b^2+c^2\cos^2\left(\beta_o+\Delta\beta\right)+2bc\cos\left(\beta_o+\Delta\beta\right)\cos\left(\zeta_o-\gamma\right)\right)+M_1b^2\right]+$$

$$+\Delta\ddot{\beta}\left[M_2bc\sin\left(\beta_o+\Delta\beta\right)\sin\left(\zeta_o-\gamma\right)\right]+$$

$$+M_2\begin{bmatrix}-2c\left(\Omega+\Delta\dot{\zeta}\right)\Delta\dot{\beta}\sin\left(\beta_o+\Delta\beta\right)\left\{c\cos\left(\beta_o+\Delta\beta\right)+b\cos\left(\zeta_o-\gamma\right)\right\}-\\-bc\Delta\dot{\beta}^2\cos\left(\beta_o+\Delta\beta\right)\sin\left(\zeta_o-\gamma\right)+ab\Omega^2\sin\left(\zeta_o+\Delta\zeta\right)+\\+ac\Omega^2\cos\left(\beta_o+\Delta\beta\right)\sin\left(\gamma+\Delta\zeta\right)\end{bmatrix}+$$

$$+M_1\left[ab\Omega^2\sin\left(\zeta_o+\Delta\zeta\right)\right]-k_z\Delta\zeta=0$$

$$\frac{\partial}{\partial t}\left[\frac{\partial L}{\partial \Delta\dot{\beta}}\right]+\left[\frac{\partial L}{\partial \Delta\beta}\right]=$$

$$=\Delta\ddot{\beta}\left[M_2c^2\right]+\Delta\ddot{\zeta}\left[M_2bc\sin\left(\beta_o+\Delta\beta\right)\sin\left(\zeta_o-\gamma\right)\right]+$$
$$+M_2\left[\begin{array}{l}-ac\Omega\Delta\dot{\beta}\cos\left(\beta_o+\Delta\beta\right)\sin\left(\gamma+\Delta\zeta\right)-ac\Omega\Delta\dot{\zeta}\sin\left(\beta_o+\Delta\beta\right)\cos\left(\gamma+\Delta\zeta\right)+\\+bc\left(\Omega+\Delta\dot{\zeta}\right)\Delta\dot{\beta}\cos\left(\beta_o+\Delta\beta\right)\sin\left(\zeta_o-\gamma\right)+c^2\left(\Omega+\Delta\dot{\zeta}\right)^2\cos\left(\beta_o+\Delta\beta\right)\sin\left(\beta_o+\Delta\beta\right)+\\+ac\Omega\Delta\dot{\beta}\cos\left(\beta_o+\Delta\beta\right)\sin\left(\gamma+\Delta\zeta\right)+ac\Omega\left(\Omega+\Delta\dot{\zeta}\right)\sin\left(\beta_o+\Delta\beta\right)\cos\left(\gamma+\Delta\zeta\right)-\\-bc\left(\Omega+\Delta\dot{\zeta}\right)\Delta\dot{\beta}\cos\left(\beta_o+\Delta\beta\right)\sin\left(\zeta_o-\gamma\right)+bc\left(\Omega+\Delta\dot{\zeta}\right)^2\sin\left(\beta_o+\Delta\beta\right)\cos\left(\zeta_o-\gamma\right)-\\gc\cos\left(\beta_o+\Delta\beta\right)\end{array}\right]+$$
$$-k_b\Delta\beta=0$$

$$\Delta\ddot{\beta}\left[M_2c^2\right]+\Delta\ddot{\zeta}\left[M_2bc\sin\left(\beta_o+\Delta\beta\right)\sin\left(\zeta_o-\gamma\right)\right]+$$
$$+M_2\left[\begin{array}{l}c\left(\Omega+\Delta\dot{\zeta}\right)^2\sin\left(\beta_o+\Delta\beta\right)\left\{c\cos\left(\beta_o+\Delta\beta\right)+b\cos\left(\zeta_o-\gamma\right)\right\}+\\+ac\Omega^2\sin\left(\beta_o+\Delta\beta\right)\cos\left(\gamma+\Delta\zeta\right)-\\gc\cos\left(\beta_o+\Delta\beta\right)\end{array}\right]+$$
$$-k_b\Delta\beta=0$$

Lagrangian equations:

$$\Delta\ddot{\zeta}\left[M_2\left(b^2 + c^2\cos^2\left(\beta_o + \Delta\beta\right) + 2bc\cos\left(\beta_o + \Delta\beta\right)\cos\left(\zeta_o - \gamma\right)\right) + M_1 b^2 \right] +$$
$$+\Delta\ddot{\beta}\left[M_2 bc\sin\left(\beta_o + \Delta\beta\right)\sin\left(\zeta_o - \gamma\right)\right] +$$
$$+ M_2\left[\begin{array}{l} -2c\left(\Omega + \Delta\dot{\zeta}\right)\Delta\dot{\beta}\sin\left(\beta_o + \Delta\beta\right)\left\{c\cos\left(\beta_o + \Delta\beta\right) + b\cos\left(\zeta_o - \gamma\right)\right\} - \\ -bc\Delta\dot{\beta}^2\cos\left(\beta_o + \Delta\beta\right)\sin\left(\zeta_o - \gamma\right) + ab\Omega^2\sin\left(\zeta_o + \Delta\zeta\right) + \\ +ac\Omega^2\cos\left(\beta_o + \Delta\beta\right)\sin\left(\gamma + \Delta\zeta\right) \end{array}\right] +$$
$$+ M_1\left[ab\Omega^2\sin\left(\zeta_o + \Delta\zeta\right)\right] - k_z\Delta\zeta = 0$$

$$\Delta\ddot{\beta}\left[M_2 c^2 \right] + \Delta\ddot{\zeta}\left[M_2 bc\sin\left(\beta_o + \Delta\beta\right)\sin\left(\zeta_o - \gamma\right)\right] +$$
$$+ M_2\left[\begin{array}{l} c\left(\Omega + \Delta\dot{\zeta}\right)^2\sin\left(\beta_o + \Delta\beta\right)\left\{c\cos\left(\beta_o + \Delta\beta\right) + b\cos\left(\zeta_o - \gamma\right)\right\} + \\ +ac\Omega^2\sin\left(\beta_o + \Delta\beta\right)\cos\left(\gamma + \Delta\zeta\right) - \\ gc\cos\left(\beta_o + \Delta\beta\right) \end{array}\right] +$$
$$-k_b\Delta\beta = 0$$

Specialisation:

a) Curvature γ and $\beta_o=\varepsilon_o=0$

$$\Delta\ddot{\zeta}\left[M_2\left(b^2+c^2\cos^2\left(\Delta\beta\right)+2bc\cos\left(\Delta\beta\right)\cos\left(\gamma\right)\right)+M_1b^2\right]+$$
$$-\Delta\ddot{\beta}\left[M_2bc\sin\left(\Delta\beta\right)\sin\left(\gamma\right)\right]+$$
$$+M_2\begin{bmatrix}-2c\left(\Omega+\Delta\dot{\zeta}\right)\Delta\dot{\beta}\sin\left(\Delta\beta\right)\left\{c\cos\left(\Delta\beta\right)+b\cos\left(\gamma\right)\right\}-\\-bc\Delta\dot{\beta}^2\cos\left(\Delta\beta\right)\sin\left(\gamma\right)+ab\Omega^2\sin\left(\Delta\zeta\right)+\\+ac\Omega^2\cos\left(\Delta\beta\right)\sin\left(\gamma+\Delta\zeta\right)\end{bmatrix}+$$
$$+M_1\left[ab\Omega^2\sin\left(\Delta\zeta\right)\right]-k_z\Delta\zeta=0$$

$$\Delta\ddot{\beta}\left[M_2c^2\right]-\Delta\ddot{\zeta}\left[M_2bc\sin\left(\Delta\beta\right)\sin\left(\gamma\right)\right]+$$
$$M_2\begin{bmatrix}c\left(\Omega+\Delta\dot{\zeta}\right)^2\sin\left(\Delta\beta\right)\left\{c\cos\left(\Delta\beta\right)+b\cos\left(\gamma\right)\right\}+\\+ac\Omega^2\sin\left(\Delta\beta\right)\cos\left(\gamma+\Delta\zeta\right)-\\gc\cos\left(\Delta\beta\right)\end{bmatrix}+$$
$$-k_b\Delta\beta=0$$

b) $\varepsilon_o=\gamma$

$$\Delta\ddot{\zeta}\left[M_2\left(b^2+c^2\cos^2\left(\beta_o+\Delta\beta\right)+2bc\cos\left(\beta_o+\Delta\beta\right)\right)+M_1b^2\right]+$$
$$+M_2\begin{bmatrix}-2c\left(\Omega+\Delta\dot{\zeta}\right)\Delta\dot{\beta}\sin\left(\beta_o+\Delta\beta\right)\left\{c\cos\left(\beta_o+\Delta\beta\right)+b\right\}+\\+ab\Omega^2\sin\left(\zeta_o+\Delta\zeta\right)+ac\Omega^2\cos\left(\beta_o+\Delta\beta\right)\sin\left(\zeta_o+\Delta\zeta\right)\end{bmatrix}+$$
$$+M_1\left[ab\Omega^2\sin\left(\zeta_o+\Delta\varepsilon\right)\right]-k_z\Delta\zeta=0$$

$$\Delta\ddot{\beta}\left[M_2c^2\right]+$$
$$M_2\begin{bmatrix}c\left(\Omega+\Delta\dot{\zeta}\right)^2\sin\left(\beta_o+\Delta\beta\right)\left\{c\cos\left(\beta_o+\Delta\beta\right)+b\right\}+\\+ac\Omega^2\sin\left(\beta_o+\Delta\beta\right)\cos\left(\zeta_o+\Delta\zeta\right)-gc\cos\left(\beta_o+\Delta\beta\right)\end{bmatrix}+$$
$$-k_b\Delta\beta=0$$

c) Linearization of case with curvature γ and $\beta_o=\varepsilon_o=0$

$$\Delta\ddot{\zeta}\left[M_2\left(b^2+c^2+2bc\cos(\gamma)\right)+M_1b^2\right]+M_2\begin{bmatrix}+ab\Omega^2\Delta\zeta\\+ac\Omega^2\sin(\gamma)+ac\Omega^2\Delta\zeta\cos(\gamma)\end{bmatrix}+$$
$$+M_1ab\Omega^2\Delta\zeta-k_z\Delta\zeta=0$$

$$\Delta\ddot{\beta}\left[M_2c^2\right]+M_2\begin{bmatrix}c\Omega^2\Delta\beta\left\{c+b\cos(\gamma)\right\}+\\+ac\Omega^2\Delta\beta\cos(\gamma)-\\-gc\end{bmatrix}-k_b\Delta\beta=0$$

And now, taking the Fourier transform:

$$\Delta\zeta\leftrightarrow\Delta\zeta(\omega);\ \ \Delta\ddot{\zeta}\leftrightarrow j\omega\, j\omega\Delta\zeta(\omega)=-\omega^2\Delta\zeta(\omega),$$
$$\Delta\beta\leftrightarrow\Delta\beta(\omega);\ \ \Delta\ddot{\beta}\leftrightarrow j\omega\, j\omega\Delta\beta(\omega)=-\omega^2\Delta\beta(\omega):$$

$$-\omega^2\Delta\zeta(\omega)\left[M_2\left(b^2+c^2+2bc\cos(\gamma)\right)+M_1b^2\right]+M_2\begin{bmatrix}+ab\Omega^2\Delta\zeta(\omega)\\+ac\Omega^2\sin(\gamma)+ac\Omega^2\Delta\zeta\cos(\gamma)\end{bmatrix}+$$
$$+M_1ab\Omega^2\Delta\zeta(\omega)-k_z\Delta\zeta(\omega)=0$$

$$-\omega^2\Delta\beta(\omega)\left[M_2c^2\right]+M_2\begin{bmatrix}c\Omega^2\Delta\beta(\omega)\left\{c+b\cos(\gamma)\right\}+\\+ac\Omega^2\Delta\beta(\omega)\cos(\gamma)-\\-gc\end{bmatrix}-$$

Which become uncoupled to first order. Solutions are:

$$\Delta\beta(\omega)=-\frac{gc}{-c^2\omega^2+c(a+b+c)\Omega^2\cos(\gamma)-(k_b/M_2)}$$

That features resonance frequencies at:

$$\omega_{res,\Delta\beta}=\sqrt{\frac{c(a+b+c)\Omega^2\cos(\gamma)-(k_b/M_2)}{c^2}}$$

And:

$$\Delta\zeta(\omega)=\frac{-M_2 ac\Omega^2\sin(\gamma)}{-\omega^2\left[M_2\left(b^2+c^2+2bc\cos(\gamma)\right)+M_1 b^2\right]+M_2\left[ab\Omega^2+ac\Omega^2\cos(\gamma)\right]+M_1 ab\Omega^2-k_z}$$

That features resonance frequencies at:

$$\omega_{res,\Delta\zeta}=\sqrt{\frac{a\Omega^2\left(b+c\cos\gamma+b\left(M_1/M_2\right)\right)-\left(k_z/M_2\right)}{b^2+c^2+2bc\cos\gamma+\left(M_1/M_2\right)b^2}}$$

Its sensitivity to variations in g is given by:

$$\frac{\partial\omega_{res,\Delta\zeta}}{\partial\gamma}=\frac{\left(2b\omega_{res,\Delta\zeta}^2-a\Omega^2\right)c\omega_{res,\Delta\zeta}}{2\left[a\Omega^2\left(b+c\cos\gamma+b\left(M_1/M_2\right)\right)-\left(k_z/M_2\right)\right]}\sin(\gamma)$$

Flap-Flap Model Lagrangian

$$\frac{\partial}{\partial t}\left[\frac{\partial L}{\partial \Delta\dot{\zeta}}\right]+\left[\frac{\partial L}{\partial \Delta\zeta}\right]=\left(M_1+M_2\right)b^2\Delta\ddot{\zeta}+M_2bc\sin\left(\Delta\zeta\right)\sin\left(\Delta\beta\right)\cos\left(\gamma\right)\Delta\ddot{\beta}+$$
$$+M_2bc\left(\Delta\dot{\beta}\right)^2\left[\sin\left(\Delta\zeta\right)\cos\left(\Delta\beta\right)\cos\left(\gamma\right)-\cos\left(\Delta\zeta\right)\sin\left(\Delta\beta\right)\right]+$$
$$+M_2bc\Omega\Delta\dot{\beta}\left[\sin\left(\Delta\beta\right)-\cos\left(\Delta\beta\right)\right]\sin\left(\Delta\zeta\right)\sin\left(\gamma\right)+$$
$$+M_2b\Omega^2\sin\left(\Delta\zeta\right)\left[a+b\cos\left(\Delta\zeta\right)+c\cos\left(\Delta\beta\right)\cos\left(\gamma\right)\right]+$$
$$+M_1b\Delta\dot{\zeta}\left(a+b\cos\left(\Delta\zeta\right)\right)\sin\left(\Delta\zeta\right)-\left(M_1+M_2\right)gb\cos\left(\Delta\zeta\right)-k_z\Delta\zeta$$

$$\frac{\partial}{\partial t}\left[\frac{\partial L}{\partial \Delta\dot{\beta}}\right]+\left[\frac{\partial L}{\partial \Delta\beta}\right]=M_2c^2\Delta\ddot{\beta}+M_2bc\sin\left(\Delta\zeta\right)\sin\left(\Delta\beta\right)\cos\left(\gamma\right)+$$
$$+M_2bc\left(\Delta\dot{\zeta}\right)^2\left[\cos\left(\Delta\varepsilon\right)\sin\left(\Delta\beta\right)\cos\left(\gamma\right)-\sin\left(\Delta\zeta\right)\cos\left(\Delta\beta\right)\right]+$$
$$+M_2bc\Omega\Delta\dot{\zeta}\left[\cos\left(\Delta\beta\right)-\sin\left(\Delta\beta\right)\right]\sin\left(\Delta\zeta\right)\sin\left(\gamma\right)+$$
$$+M_2c\Omega^2\sin\left(\Delta\beta\right)\left[c\cos\left(\Delta\beta\right)+\left(a+b\cos\left(\Delta\zeta\right)\right)\cos\left(\gamma\right)\right]-$$
$$-M_2gc\cos\left(\Delta\beta\right)-k_b\Delta\beta$$

The Lagrangian equations are:

$$\left(M_1+M_2\right)b^2\Delta\ddot{\zeta}+M_2bc\sin\left(\Delta\zeta\right)\sin\left(\Delta\beta\right)\cos\left(\gamma\right)\Delta\ddot{\beta}+$$
$$+M_2bc\left(\Delta\dot{\beta}\right)^2\left[\sin\left(\Delta\zeta\right)\cos\left(\Delta\beta\right)\cos\left(\gamma\right)-\cos\left(\Delta\zeta\right)\sin\left(\Delta\beta\right)\right]+$$
$$+M_2bc\Omega\Delta\dot{\beta}\left[\sin\left(\Delta\beta\right)-\cos\left(\Delta\beta\right)\right]\sin\left(\Delta\zeta\right)\sin\left(\gamma\right)+$$
$$+M_2b\Omega^2\sin\left(\Delta\zeta\right)\left[a+b\cos\left(\Delta\zeta\right)+c\cos\left(\Delta\beta\right)\cos\left(\gamma\right)\right]+$$
$$+M_1b\Delta\dot{\zeta}\left(a+b\cos\left(\Delta\zeta\right)\right)\sin\left(\Delta\zeta\right)-\left(M_1+M_2\right)gb\cos\left(\Delta\zeta\right)-k_z\Delta\zeta=0$$

$$M_2c^2\Delta\ddot{\beta}+M_2bc\sin\left(\Delta\zeta\right)\sin\left(\Delta\beta\right)\cos\left(\gamma\right)+$$
$$+M_2bc\left(\Delta\dot{\zeta}\right)^2\left[\cos\left(\Delta\varepsilon\right)\sin\left(\Delta\beta\right)\cos\left(\gamma\right)-\sin\left(\Delta\zeta\right)\cos\left(\Delta\beta\right)\right]+$$
$$+M_2bc\Omega\Delta\dot{\zeta}\left[\cos\left(\Delta\beta\right)-\sin\left(\Delta\beta\right)\right]\sin\left(\Delta\zeta\right)\sin\left(\gamma\right)+$$
$$+M_2c\Omega^2\sin\left(\Delta\beta\right)\left[c\cos\left(\Delta\beta\right)+\left(a+b\cos\left(\Delta\zeta\right)\right)\cos\left(\gamma\right)\right]-$$
$$-M_2gc\cos\left(\Delta\beta\right)-k_b\Delta\beta=0$$

Which can be linearized:

$$\left(M_1+M_2\right)b^2\Delta\ddot{\zeta}+M_2b\Omega^2\Delta\zeta\left[a+b+c\cos\left(\gamma\right)\right]-\left(M_1+M_2\right)gb-k_z\Delta\zeta=0$$

$$M_2c^2\Delta\ddot{\beta}+M_2c\Omega^2\Delta\beta\left[c+\left(a+b\right)\cos\left(\gamma\right)\right]-M_2gc-k_b\Delta\beta=0$$

And now, taking the Fourier transform:

$$\Delta\zeta\leftrightarrow\Delta\zeta\left(\omega\right);\ \ \Delta\ddot{\zeta}\leftrightarrow j\omega\, j\omega\Delta\zeta\left(\omega\right)=-\omega^2\Delta\zeta\left(\omega\right),$$

$$\Delta\beta\leftrightarrow\Delta\beta\left(\omega\right);\ \ \Delta\ddot{\beta}\leftrightarrow j\omega\, j\omega\Delta\beta\left(\omega\right)=-\omega^2\Delta\beta\left(\omega\right):$$

$$-\left(M_1+M_2\right)b^2\Delta\zeta\left(\omega\right)\omega^2+M_2b\Omega^2\Delta\zeta\left(\omega\right)\left[a+b+c\cos\left(\gamma\right)\right]-\left(M_1+M_2\right)gb-k_z\Delta\zeta\left(\omega\right)=0$$

$$-M_2c^2\Delta\beta\left(\omega\right)\omega^2+M_2c\Omega^2\Delta\beta\left(\omega\right)\left[c+\left(a+b\right)\cos\left(\gamma\right)\right]-M_2gc-k_b\Delta\beta\left(\omega\right)=0$$

Yielding:

$$\Delta\zeta\left(\omega\right)=\frac{\left(M_1+M_2\right)gb}{-\left(M_1+M_2\right)b^2\omega^2+M_2b\Omega^2\left[a+b+c\cos\left(\gamma\right)\right]-k_z}$$

$$\Delta\beta\left(\omega\right)=\frac{M_2gc}{-M_2c^2\omega^2+M_2c\Omega^2\left[c+\left(a+b\right)\cos\left(\gamma\right)\right]-k_b}$$

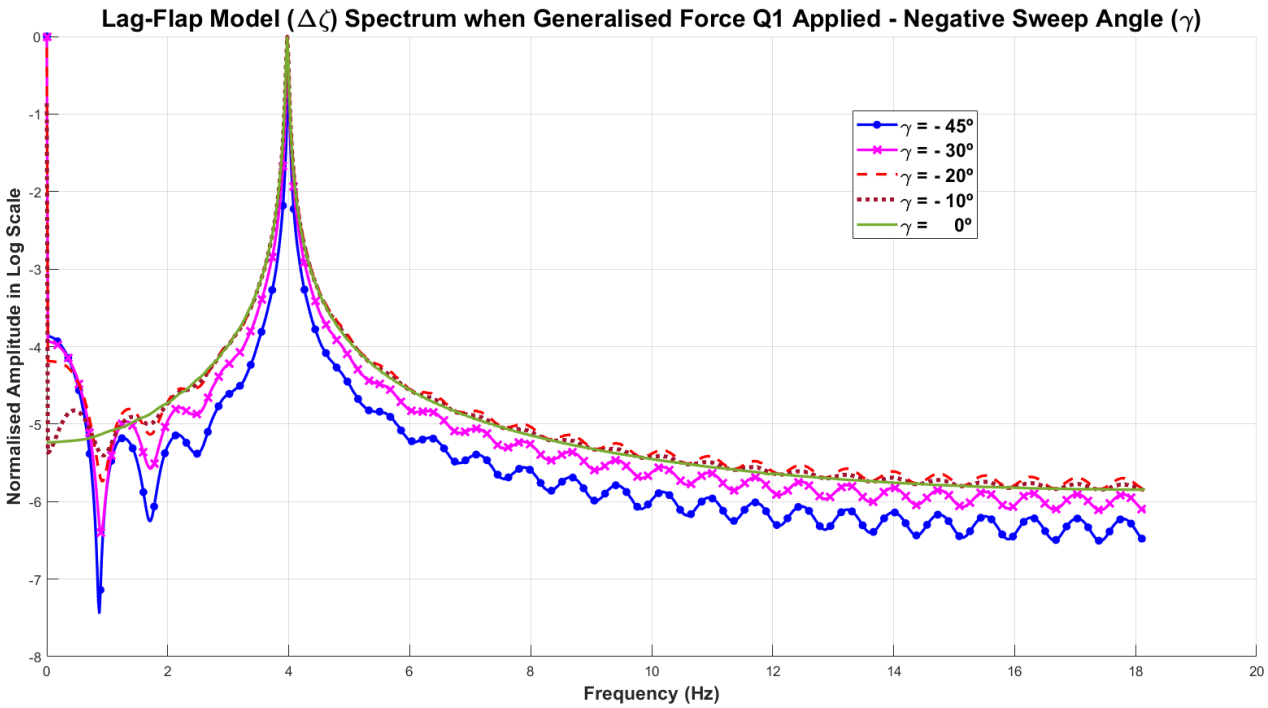
With resonance frequencies given by:

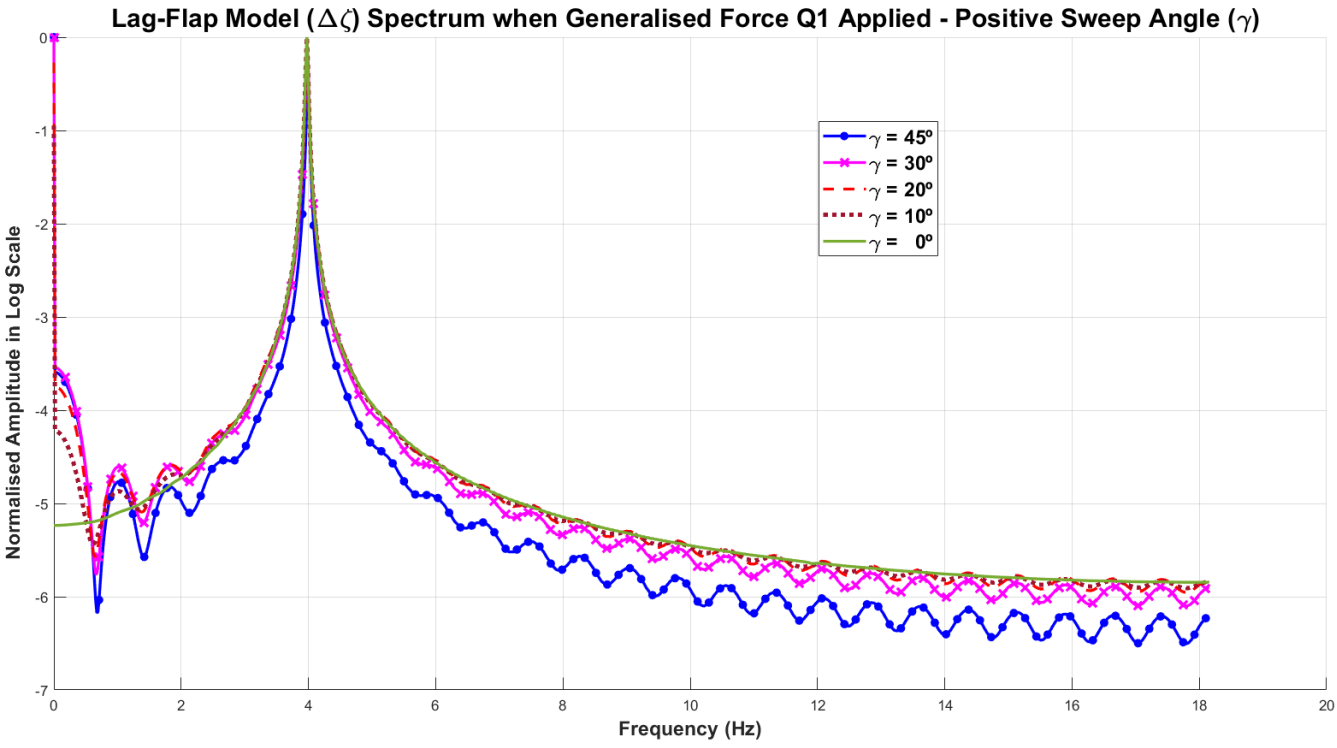
$$\omega_{res,\Delta\zeta}=\sqrt{\frac{b\Omega^2\left[a+b+c\cos\left(\gamma\right)\right]-\left(k_z/M_2\right)}{\left(1+M_1/M_2\right)b^2}}$$

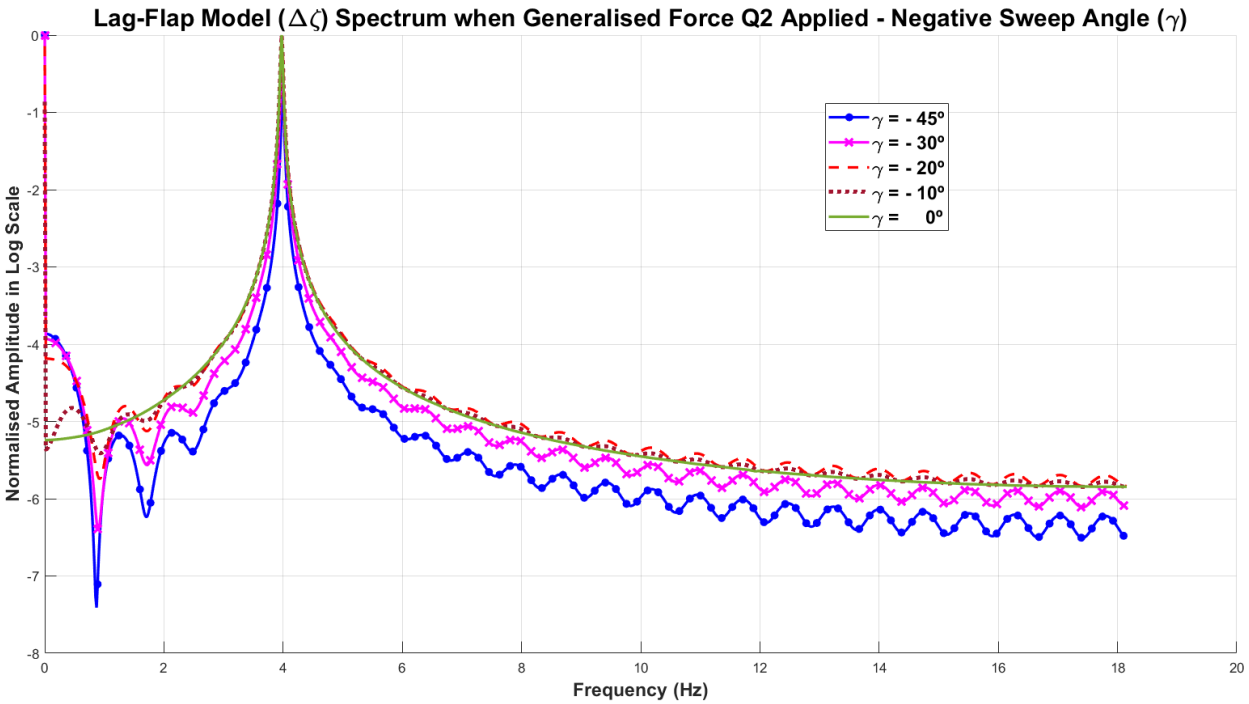
$$\omega_{res,\Delta\beta}=\sqrt{\frac{c\Omega^2\left[c+\left(a+b\right)\cos\left(\gamma\right)\right]-\left(k_b/M_2\right)}{M_2c^2}}$$

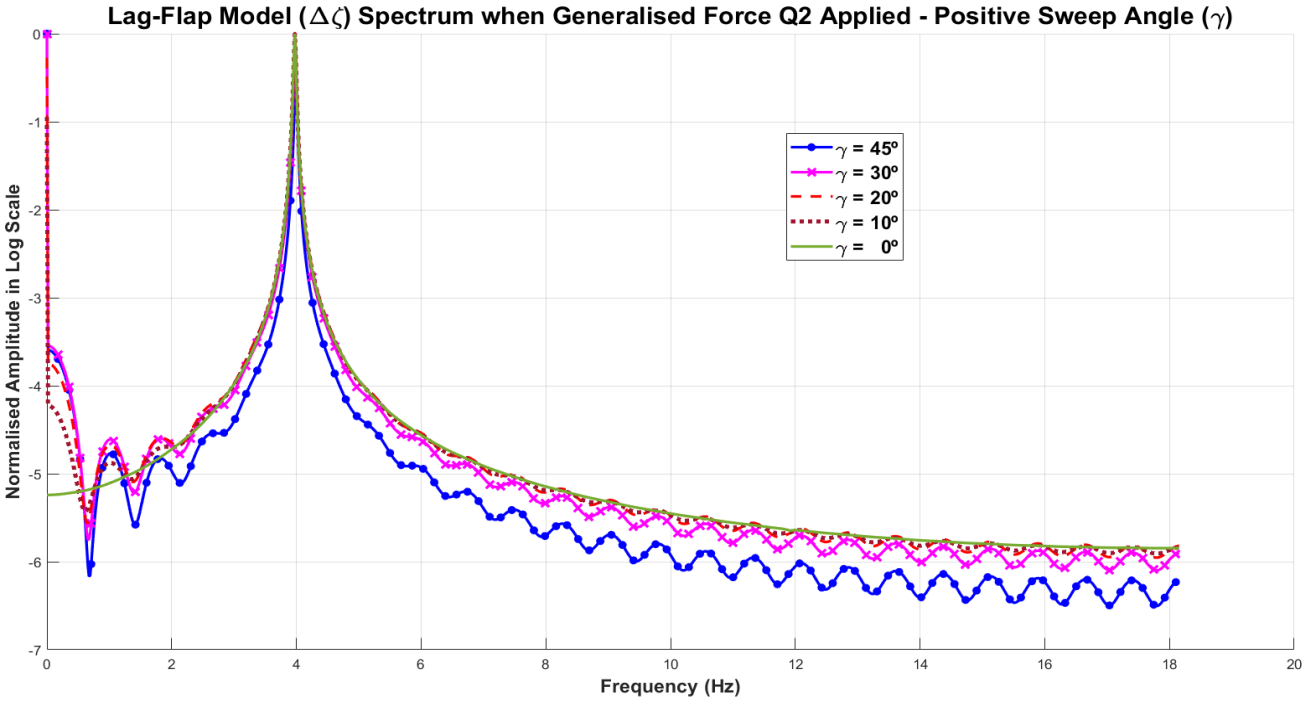
Appendix D

Lag-Flap Model Spectrum for $\Delta\zeta$ D.O.F:

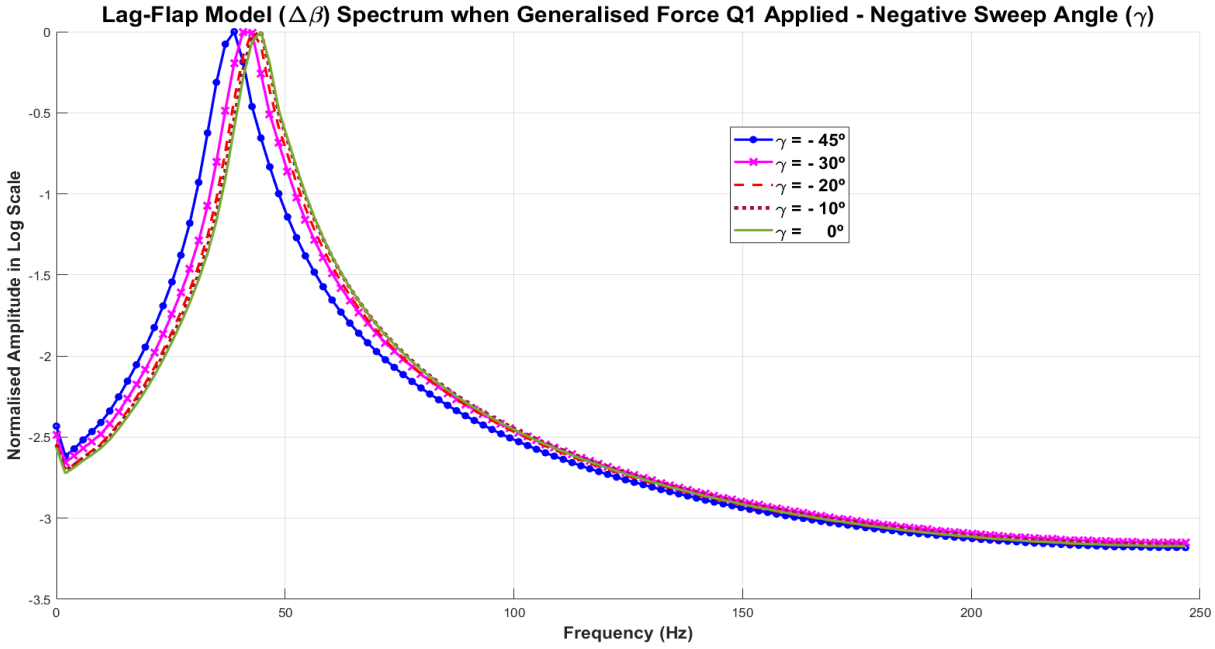


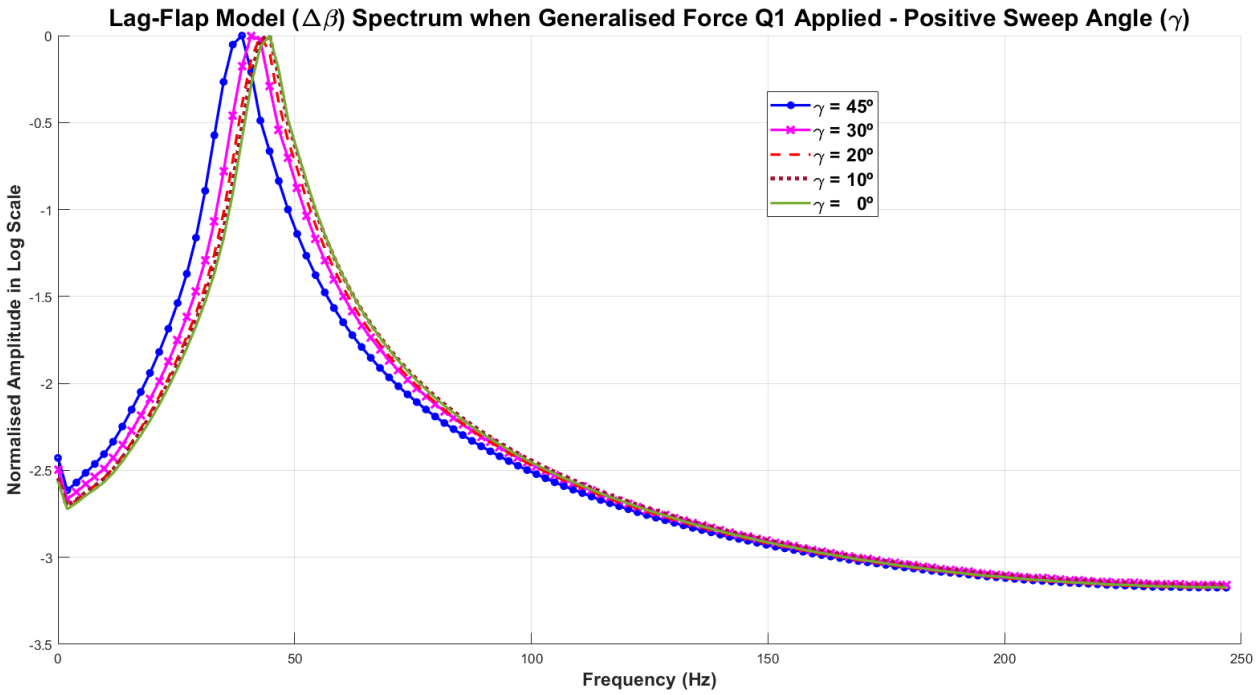


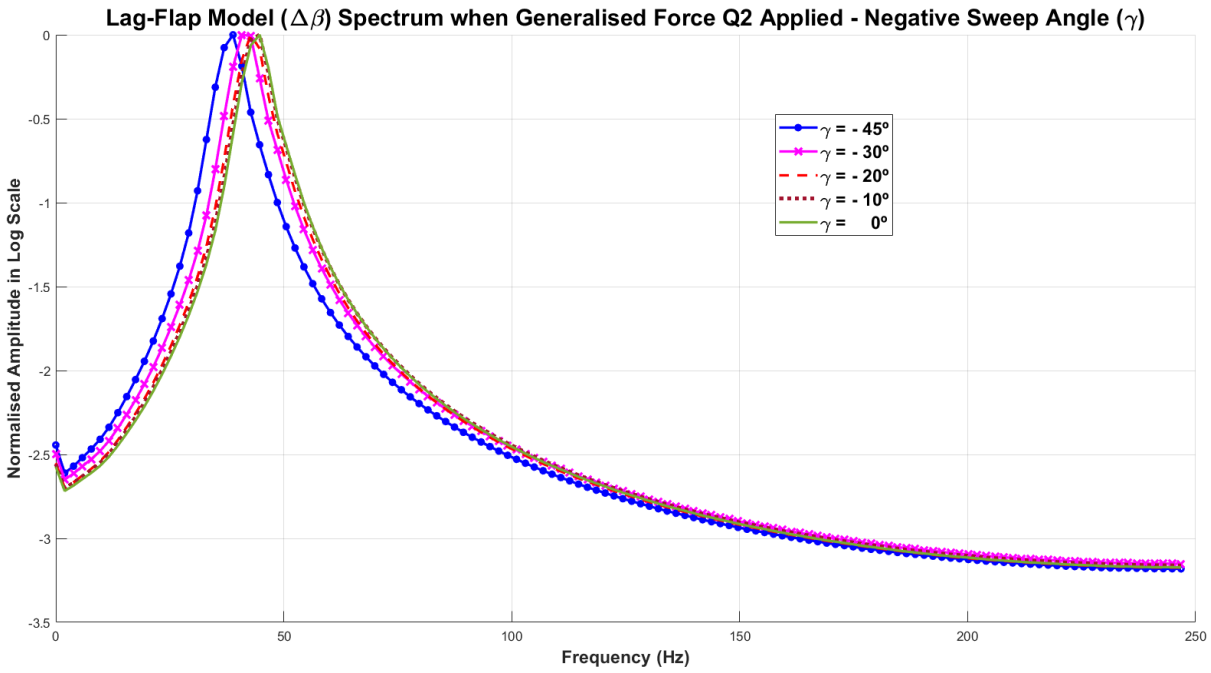

Lag-Flap Model ($\Delta\zeta$) Spectrum when Generalised Force Q2 Applied - Negative Sweep Angle (γ)
Normalised Amplitude in Log Scale
Frequency (Hz)
γ = - 45º
γ = - 30º
γ = - 20º
γ = - 10º
γ = 0º

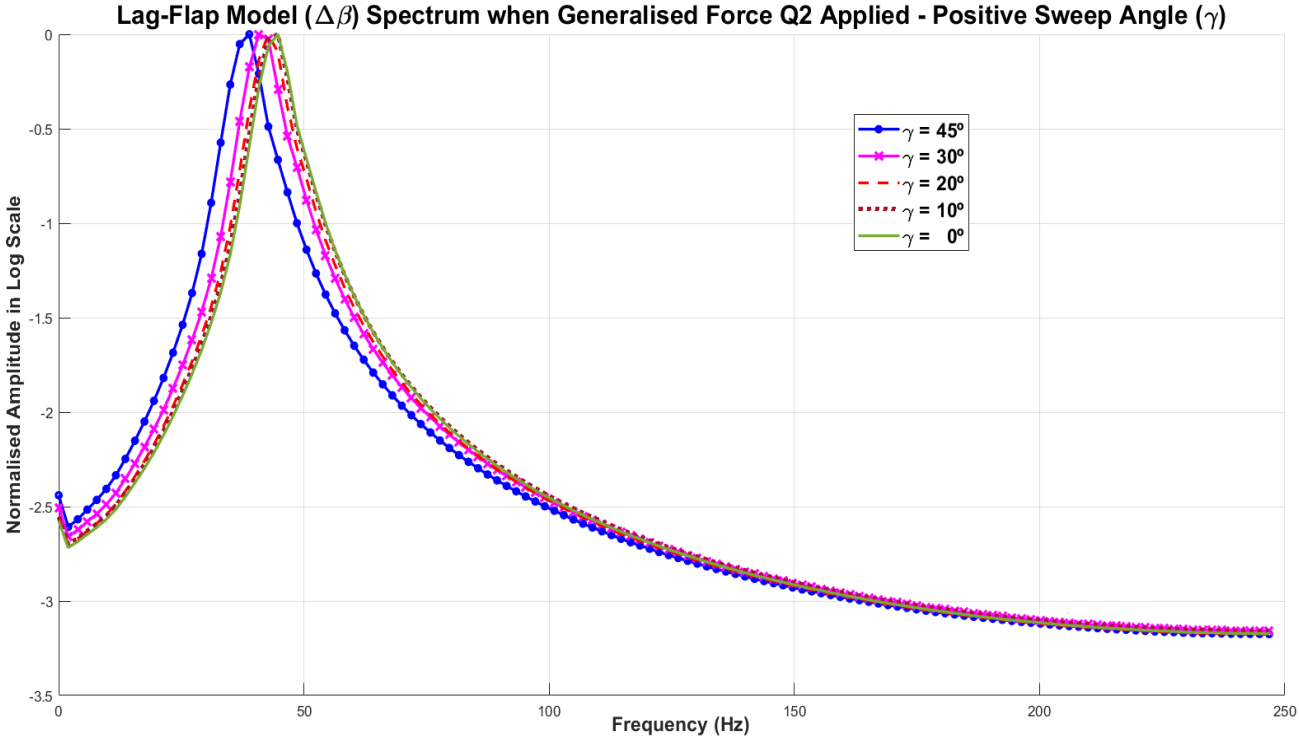

Lag-Flap Model ($\Delta\zeta$) Spectrum when Generalised Force Q2 Applied - Positive Sweep Angle (γ)
Normalised Amplitude in Log Scale
Frequency (Hz)
γ = 45º
γ = 30º
γ = 20º
γ = 10º
γ = 0º

Lag-Flap Model Spectrum for Δβ D.O.F:

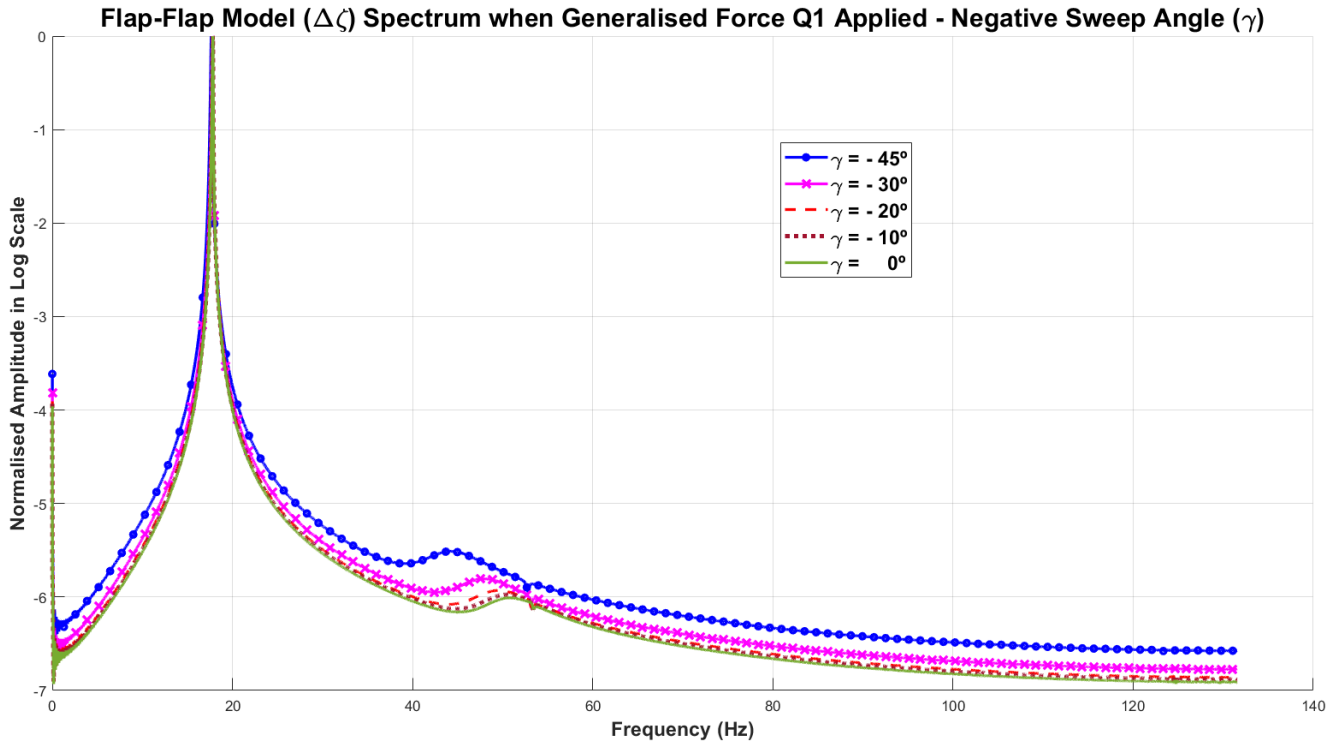


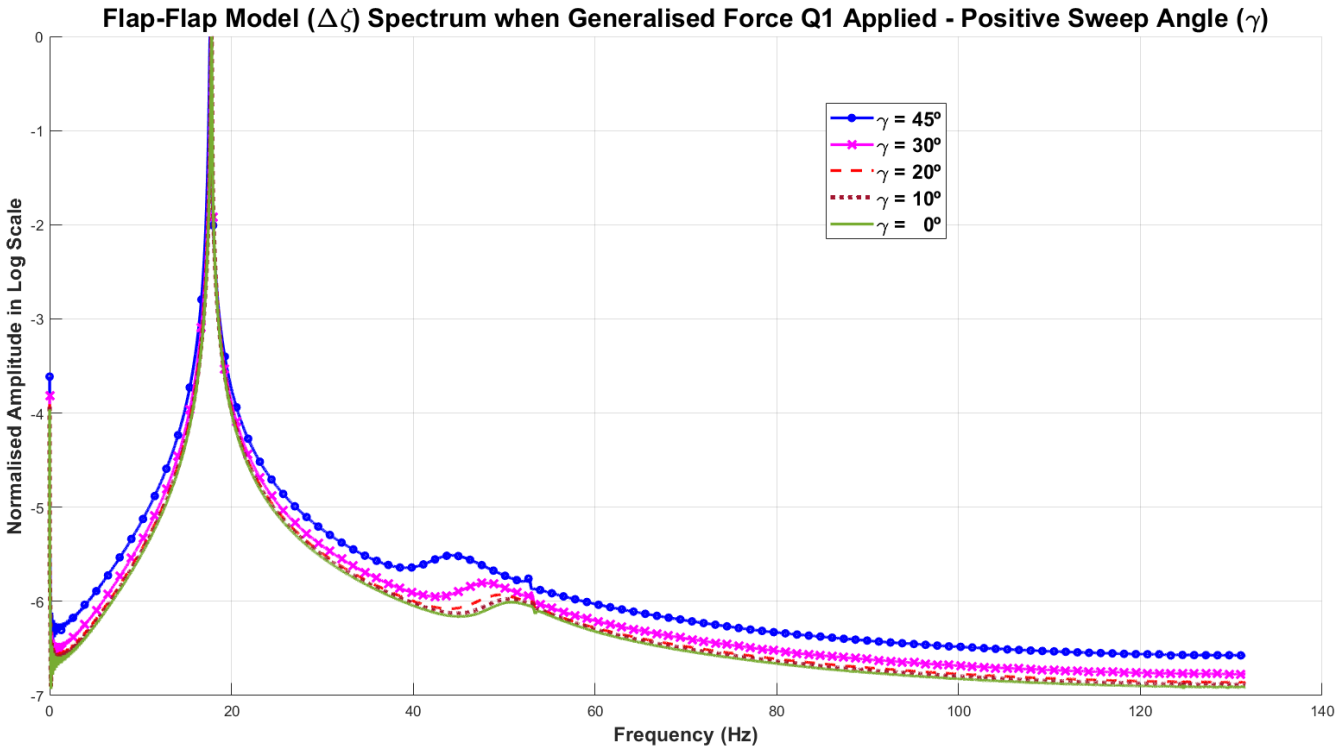


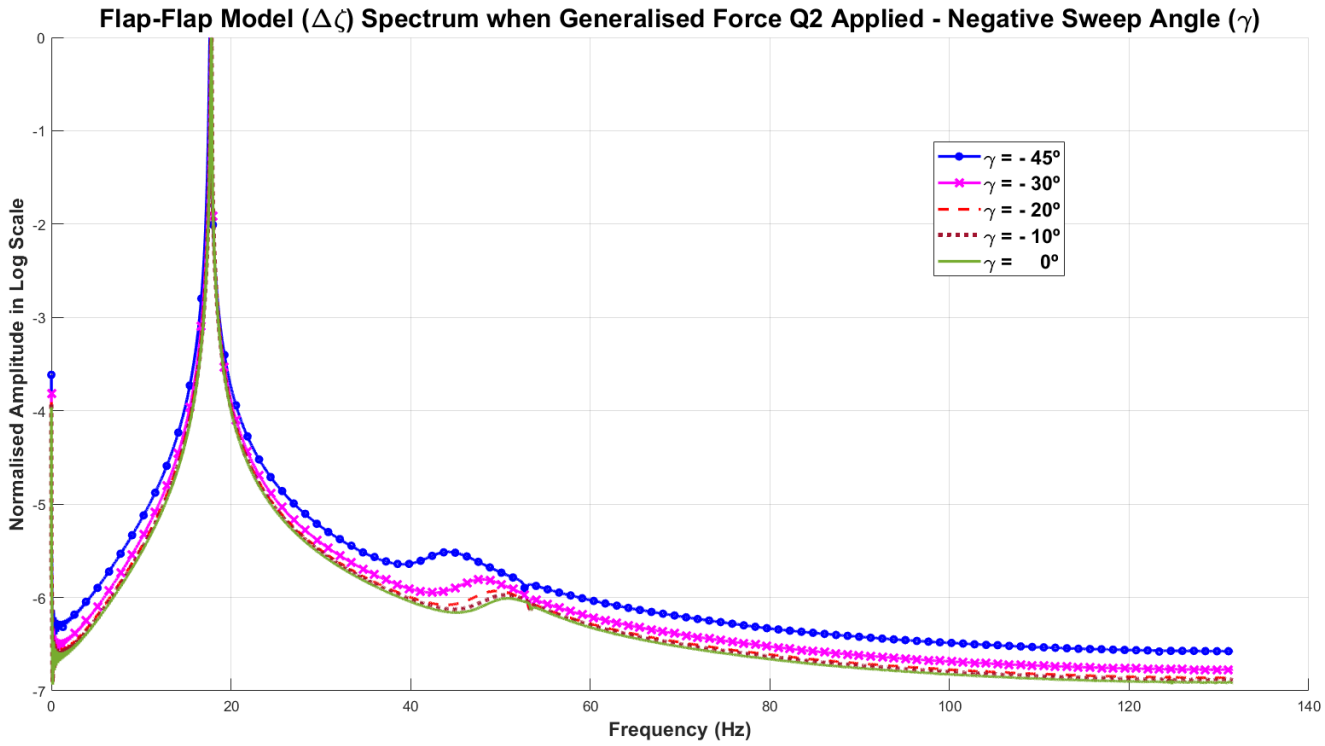

Lag-Flap Model ($\Delta\beta$) Spectrum when Generalised Force Q2 Applied - Negative Sweep Angle (γ)
Normalised Amplitude in Log Scale
Frequency (Hz)
γ = - 45º
γ = - 30º
γ = - 20º
γ = - 10º
γ = 0º
0
-0.5
-1
-1.5
-2
-2.5
-3
-3.5
0
50
100
150
200
250

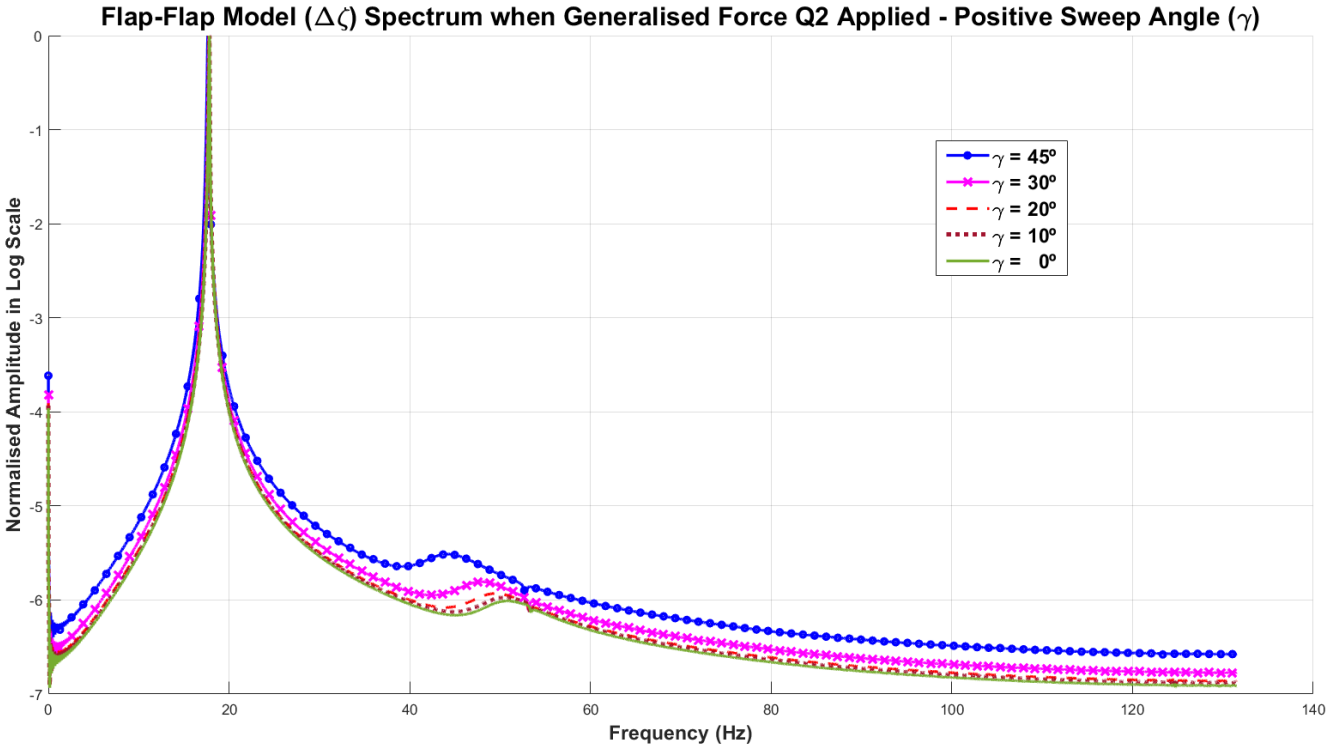

Lag-Flap Model ($\Delta\beta$) Spectrum when Generalised Force Q2 Applied - Positive Sweep Angle (γ)
Normalised Amplitude in Log Scale
Frequency (Hz)
γ = 45º
γ = 30º
γ = 20º
γ = 10º
γ = 0º
0
-0.5
-1
-1.5
-2
-2.5
-3
-3.5
0
50
100
150
200
250

Flap-Flap Model Spectrum for $\Delta\zeta$ D.O.F:

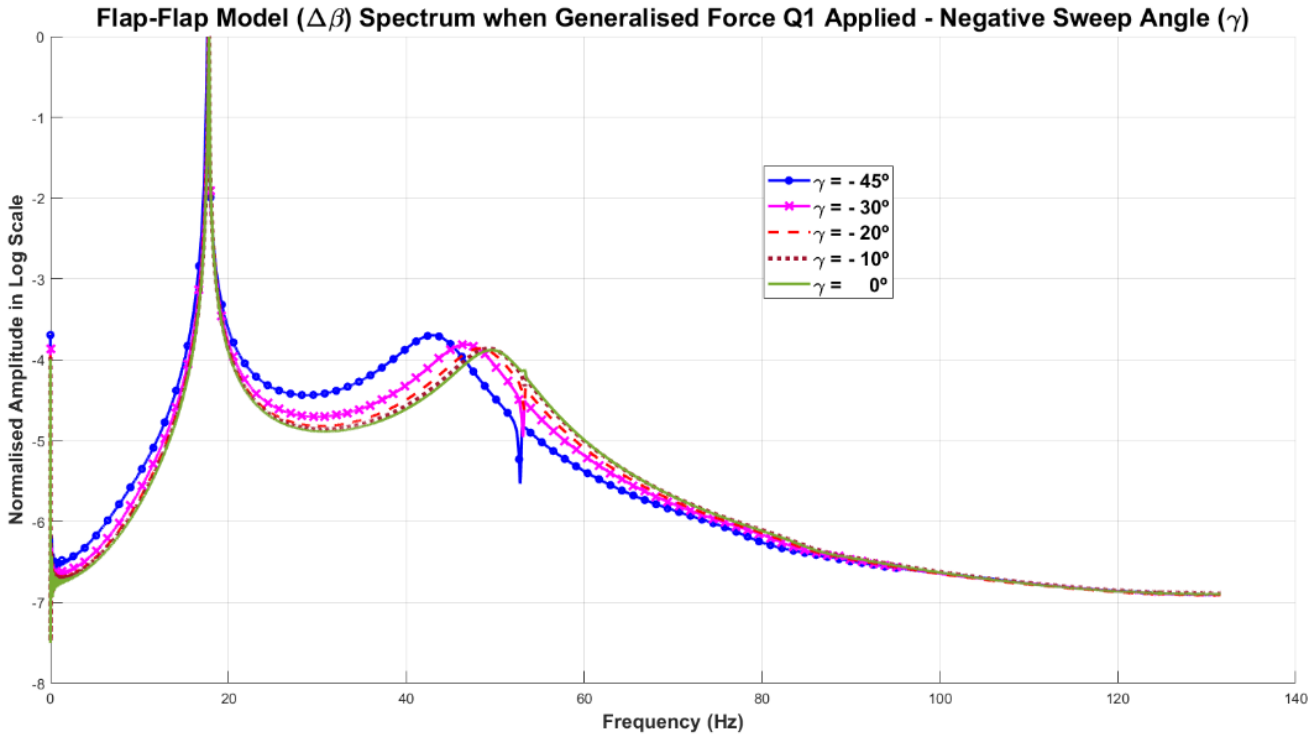


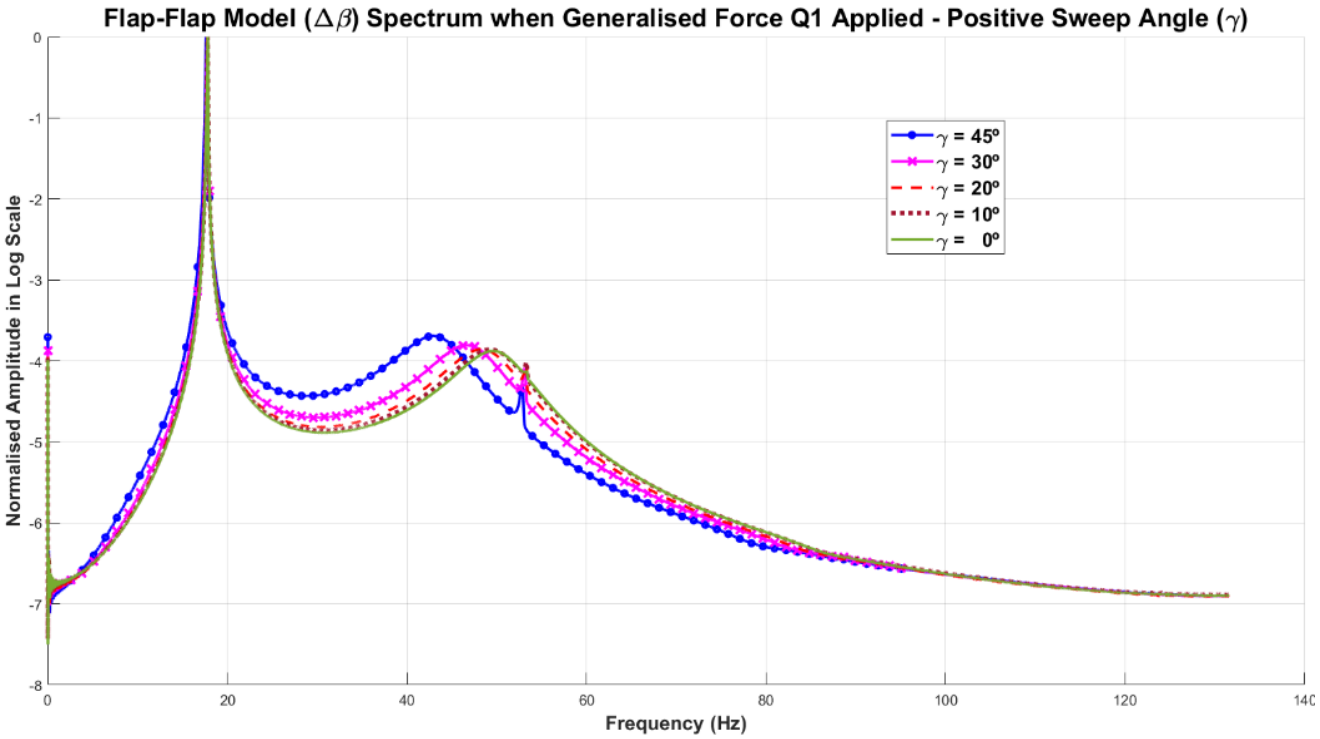


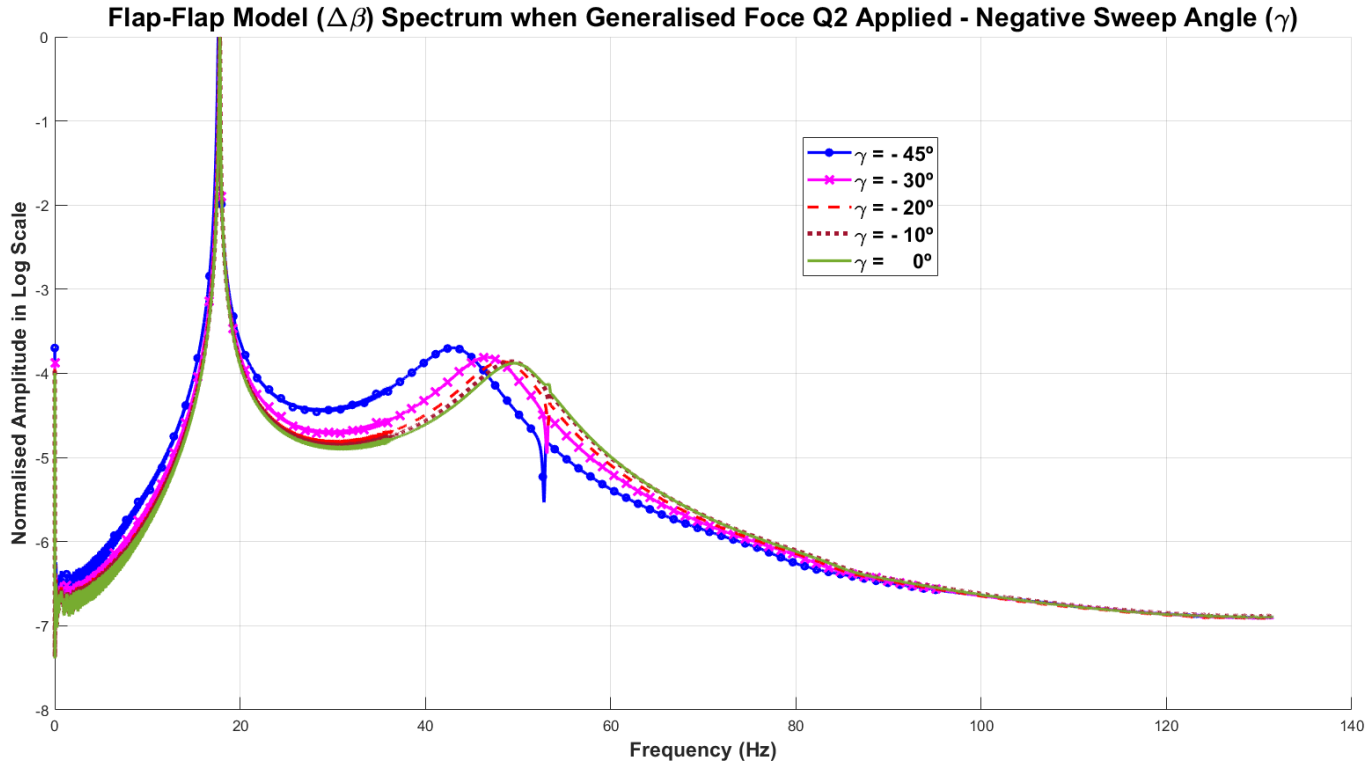

Flap-Flap Model (Δζ) Spectrum when Generalised Force Q2 Applied - Negative Sweep Angle (γ)
Normalised Amplitude in Log Scale
Frequency (Hz)
γ = - 45º
γ = - 30º
γ = - 20º
γ = - 10º
γ = 0º
0
-1
-2
-3
-4
-5
-6
-7
0
20
40
60
80
100
120
140

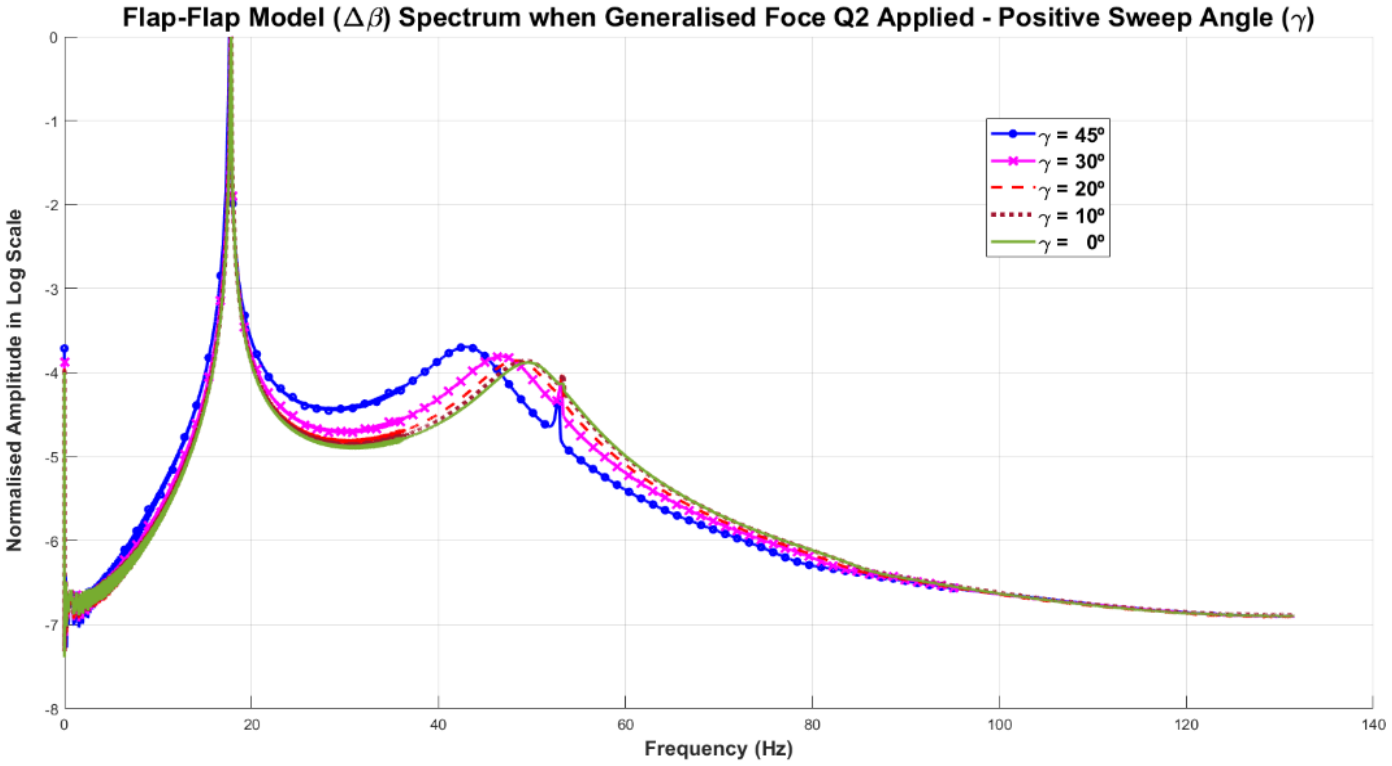

Flap-Flap Model (Δζ) Spectrum when Generalised Force Q2 Applied - Positive Sweep Angle (γ)
Normalised Amplitude in Log Scale
Frequency (Hz)
γ = 45º
γ = 30º
γ = 20º
γ = 10º
γ = 0º
0
-1
-2
-3
-4
-5
-6
-7
0
20
40
60
80
100
120
140

Flap-Flap Model Spectrum for ∆β D.O.F:






Flap-Flap Model (Δβ) Spectrum when Generalised Foce Q2 Applied - Negative Sweep Angle (γ)
Normalised Amplitude in Log Scale
Frequency (Hz)
γ = - 45º
γ = - 30º
γ = - 20º
γ = - 10º
γ = 0º


Flap-Flap Model (Δβ) Spectrum when Generalised Foce Q2 Applied - Positive Sweep Angle (γ)
Normalised Amplitude in Log Scale
Frequency (Hz)
γ = 45º
γ = 30º
γ = 20º
γ = 10º
γ = 0º

Appendix E

Parameter	Units	Value	Reference
Hub radius (a)	m	0.381	Stroub et al. (1987)
1st Segment length (b)	m	7.162	Leonardo S.p.A Company (2020)
Hinge (1 and 2) length	m	0.200	Estimation
1st Segment + Hinge length (B)	m	7.362	Estimation
2nd Segment length (c)	m	1.367	Leonardo S.p.A Company (2020)
Gravity (g)	$m \cdot s^{-2}$	9.810	European Space Agency (2020)
Hub Mas (M_{hub})	kg	36.26	Martínez Santín (2009)
Segment 1 Mass (M_{seg1})	kg	31.88	Martínez Santín (2009)
Segment 2 Mass (M_{seg2})	kg	6.08	Martínez Santín (2009)
Rotor Regime (Ω)	$rad \cdot s^{-1}$	109.220	Oxley et al. (2009)
Spring 1 Stiffness ($k_{\Delta\zeta}$)	$N \cdot m \cdot rad^{-1}$	2900.000	Takahasi (1990)
Spring 2 Stiffness ($k_{\Delta\beta}$)	$N \cdot m \cdot rad^{-1}$	2900.000	Takahasi (1990)
Damper 1 Coefficient ($c_{\Delta\zeta}$)	$N \cdot m \cdot rad^{-1} \cdot s$	500.000	Titurus (2018)
Damper 2 Coefficient ($c_{\Delta\beta}$)	$N \cdot m \cdot rad^{-1} \cdot s$	500.000	Titurus (2018)
Blade Curvature (γ)	[-]	-45º to 45º	Seddon and Newman (2002)
Force Amplitude (Q_{amp})	N	60.000	Estimation
Frequency rate of change (s_{rate})	[-]	2	Estimation
Δζ Initial Condition ($\Delta\zeta_0$)	rad	π/20	Estimation
Δβ Initial Condition ($\Delta\beta_0$)	rad	-π/20	Estimation
$\dot{\Delta}\zeta$ Initial Condition ($\Delta\dot{\zeta}_0$)	$rad \cdot s^{-1}$	1.000	Estimation
$\dot{\Delta}\beta$ Initial Condition ($\Delta\dot{\beta}_0$)	$rad \cdot s^{-1}$	-1.000	Estimation

Appendix F

Lag-Flap Model 1st Lagging Frequency (Δζ D.O.F):

Curvature Angle (γº)	L1 (Hz) [ODE 45]	L1 (Hz) [Linear]	Relative Error (%)
45.0	3.99	3.86	3.26
30.0	3.98	3.85	3.27
20.0	3.98	3.85	3.27
10.0	3.98	3.84	3.52
0.0	3.98	3.84	3.52
-10.0	3.98	3.84	3.52
-20.0	3.98	3.85	3.27
-10.0	3.98	3.84	3.52
-20.0	3.98	3.85	3.27
-30.0	3.98	3.85	3.27
-45.0	3.99	3.86	3.26

Lag-Flap Model 1st Flapping Frequency (Δβ D.O.F):

Curvature Angle (γº)	F1 (Hz) [ODE 45]	F1 (Hz) [Linear]	Relative Error (%)
45.0	38.87	37.54	3.42
30.0	40.81	41.58	-1.89
20.0	42.75	43.33	-1.36
10.0	44.70	44.36	0.76
0.0	44.70	44.71	-0.02
-10.0	44.70	44.36	0.76
-20.0	42.75	43.33	-1.36
-30.0	40.81	41.58	-1.89
-45.0	38.87	37.54	3.42

Flap-Flap Model 1st Flapping Frequency (Δβ D.O.F):

Curvature Angle (γº)	F1 (Hz) [ODE 45]	F1 (Hz) [Linear]	Relative Error (%)
45.0	17.65	15.69	11.10
30.0	17.75	17.06	3.89
20.0	17.80	17.66	0.79
10.0	18.81	18.01	4.25
0.0	17.83	18.13	-1.68
-10.0	17.81	18.01	4.25
-20.0	17.80	17.66	0.79
-30.0	17.75	17.06	3.89
-45.0	17.65	15.69	11.10

Flap-Flap Model 2nd Flapping Frequency (Δβ D.O.F):

Curvature Angle (γº)	F2 (Hz) [ODE 45]
45.0	43.06
30.0	46.79
20.0	48.47
10.0	49.34
0.0	49.74
-10.0	49.34
-20.0	48.47
-30.0	46.79
-45.0	43.06