# FLEXIBLE MULTIBODY DYNAMICS OF AN AFTERBURNER-TYPE FAIRGROUND ATTRACTION

J.J. Vermeulen

**Abstract** In this paper, a method to analyze flexible multibody systems is presented, based on the floating frame formulation and applied to the Afterburner. In this formulation a floating frame is fixed to the rigid body center of mass. The rigid body motions, which make the system highly nonlinear, are described by constraint equations in a global frame. While the flexible deformations are described in the local floating frame by a linear combination of mode shapes multiplied with flexible coordinates. Linear finite element models are made and reduced to the interface points using the Craig-Bampton method, resulting in a small amount of mode shapes. At the interface points are revolute joints positioned, which include large three-dimensional rotations. To overcome singularity problems a solution procedure is described using an update of the rotation matrices. The results of this model are validated using simulations.

**Keywords** Flexible multibody dynamics | Floating Frame of Reference (FFR) | Model Order Reduction (MOR) | Craig-Bampton method | Finite Element Method (FEM) | Superelement

## 1. INTRODUCTION

The design of fairground attractions encounters multiple challenges. Over the last decades, the designs have become higher, faster and more extreme than ever before. In addition, fairground attractions must be transportable, preferably fit on a single trailer and installation time should be as short as possible. Manufacturers of fairground attractions are looking for ways to reduce the weights of their attractions in order to reduce transportation costs. Within this industry there is a trend towards lightweight designs on one hand and a trend towards more extreme fairground attractions on the other. Therefore, the study of the dynamic behavior becomes more important to increase the reliability and safety of the lightweight attractions and to prolong lifetime.

Traditionally, the deformations and internal stresses of critical components are studied using a linear finite element model, in which load cases are applied in a quasi-static fashion. These load cases are obtained from rigid multibody dynamics simulations. To improve the prediction of the attraction's elastic deformations and stresses, the industry has an increased interest to use flexible multibody dynamics methods.

In this work, a flexible multibody dynamics analysis of the Afterburner is presented using the floating frame of reference formulation [1]. The most important contribution of this paper is in providing an overview and practical application of how various, well-developed and well-documented, methods can be combined to form a solution strategy for high-nonlinear dynamic problems, including large rotations.

The Afterburner was chosen in particular after a fatal accident in 2017 [2]. Although this accident had a different cause, it got our attention and we became curious about its dynamic behavior.

The outline of this paper is as follows: in Section 2, background information is given about the applied method, using literature and a schematic overview of the working procedure, followed by general information about the Afterburner. In Section 3, the theory of flexible dynamics using the floating frame formulation is briefly explained, first for a single body and continued for multibody systems. An outline of the terms in the constrained equations of motion is given and a solution procedure is described. Section 4 gives an overview of the Afterburner parts, how they are modeled and relevant characteristics. In Section 5, the rigid kinematics and kinetics simulation results are shown, as well as the flexible kinematics. Followed by the dynamic calculation of the equivalent stresses in the clapper of the Afterburner.

# 2. METHODOLOGY

## 2.1 THEORETICAL BACKGROUND

To model three dimensional rigid multibody systems, the augmented formulation can be used to describe the kinematics and kinetics. A body´s kinematics can be described by the motion of a set of coordinate frames, rigidly attached to a point on the body. At the interface points of the bodies, connections are made to other bodies or the ground using joints. These connections are made in the form of constraint equations, defining the position and orientation of the interface points with respect to the body's floating frame, which are explained in textbooks such as [1, 3, 4]. The equations of motion (EoM) can be compiled, which is a system of differential algebraic equations (DAEs) that can be solved, given a set of initial conditions for the generalized coordinates $(\mathbf{q}_0, \dot{\mathbf{q}}_0)$. The set of equations contain the degrees of freedom (DOF), which are the position and orientation of the rigidly attached coordinate frames of the bodies and the time derivatives. Lagrange multipliers $(\boldsymbol{\lambda})$ are used to enforce the constraints, since an analytical solution might not be found. The total number of unknowns is typically high and due to the large rigid body rotations between different bodies, this problem is of a geometrical nonlinear nature. Explicit numerical integration methods like Runge-Kutta and Adams-Bashforth Predictor are explained and applied in [4] as well as the Adams-Moulton Corrector which is an implicit method. Additionally, a Newton-Raphson method can be added to improve the solutions of the positions and to ensure that all constraint equations are satisfied.

Modeling joints can become challenging if large three-dimensional rotations are included. All methods using three parameters for the parameterization of three angles, suffer from singularities in the velocity transformation and are therefore not suitable for large rotations [5]. Schemes are developed to overcome this problem using different formulas in different quadrants of a circle [4], with the downside of discontinuity in the constraint equations. Using quaternions, three large rotations can be described using four parameters, but they have the drawback of giving a redundancy of description.

The rigid body motions are nonlinear, but the flexible behavior is small and therefore it can be assumed to be linear. In this case the Floating Frame of Reference (FFR) formulation is the preferred formulation to describe flexibility [6]. The augmented formulation can be extended for the flexible system using a floating frame describing the large rigid body motions of a body with respect to the inertia frame using the constraint equations, this is described in [1, 7]. The linear theory of elasticity is used to describe the flexible behavior locally, satisfying Hooke´s law and approximating the strains with the linear Cauchy strain tensor [7, 8]. The local generalized coordinates $(\mathbf{q})$ are written as a linear combination of a small number of mode shapes multiplied with the flexible coordinates $(\boldsymbol{\eta})$, which describe the behavior of the modes in time [9]. The mass $(\mathbf{M})$ and stiffness $(\mathbf{K})$ matrices can be used from a body's linear finite element model. This model typically contains a large number of degrees of freedom, which results in an unbeneficial large system of equations to solve. However, the required number of degrees of freedom to describe the flexible behavior is much less, because the lower natural frequencies and corresponding modes tend to dominate the global dynamic behavior [10].

Because the local model is described linear, the number of flexible coordinates can be reduced using well-developed linear model order reduction (MOR) techniques like the Craig-Bampton method [11, 12]. Its simplicity and computational stability make it a highly regarded reduction method. The local reference frame is placed in the center of mass for better accuracy. To overcome singularity problems with rigid body motion contained in the Craig-Bampton reduction basis, a second reduction technique is applied using the natural modes of the system. The natural modes $(\boldsymbol{\psi}_i)$ are obtained by solving a body's Eigenvalue problem for free vibrations: $(\mathbf{K} - \boldsymbol{\omega}_i^2\mathbf{M})\boldsymbol{\psi}_i = \mathbf{0}$, in which $\boldsymbol{\omega}_i$ are the corresponding natural frequencies. This second reduction step diagonalize the mass and stiffness matrix and is also used by the multibody dynamics simulation software MSC Adams [13, 14].

The final flexible constrained equations of motion grow with the number of flexible coordinates. The flexible dynamics equations are also differential algebraic equations which are 'stiff', the system Eigenfrequencies are distributed over a broad frequency range which make them hard to solve. Implicit numerical integration schemes are most popular for solving these equations, like the generalized-alpha method and the Newmark-beta method [15]. A practical application of the generalized-alpha method can be found in [16]. Although they are harder to implement, they benefit from unconditional stability properties whereas explicit schemes are only stable for sufficiently small timesteps.

## 2.2 WORKING PROCEDURE

The benefit of the augmented formulation is the ability to extend the calculations stepwise, each step increasing the difficulty and the possibility to verify for human errors. The first step is the rigid kinematics of the system, define the set of constraint equations and eventually calculate the constraint moments. The next step is the rigid kinetics, where driving moments are used to calculate the rigid body motions and the constraint forces and moments. This calculation is made to validate the model and in practice this is how the system is driven. The last step is the extension to the flexible kinematics, where the constraint equations must be rewritten slightly to calculate the constraint forces, moments and deformations at the same time. Figure 1 shows an overview of the working procedure, with blue arrows referencing to the steps for the rigid calculations and the orange arrows referencing to the additional steps for the flexible calculations.

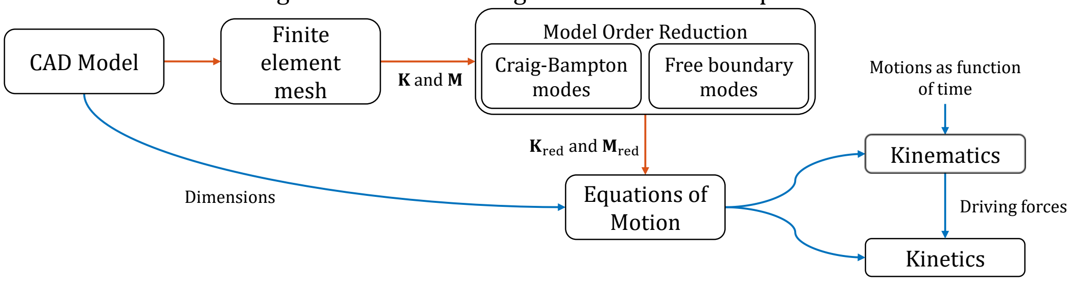


**Figure 1: Working scheme**

## 2.3 THE AFTERBURNER

The method described in this paper will be evaluated using the Afterburner. Figure 2 shows a model of the Afterburner, which consist of two main bodies: the clapper and hub. The fixed framework is out of scope for this study. The clapper is made from a twelve-sided tube, with a length of approximately 8 meters and a weight of ~1200 kg. The hub consists of a driving drum with six gondolas attached to it, each gondola contains four seats. The diameter of the hub is approximately 6 meters and has a weight of ~5000 kg (24 persons included).

The clapper swings harmonically with a period of 8 seconds and a maximum angle of $\pm 120$ degrees, while the hub continuously rotates at a constant speed of 15 rpm [17]. From the manufacturer additional drawings were retrieved to recreate a more accurate model.

**Figure 2: Model of the Afterburner**

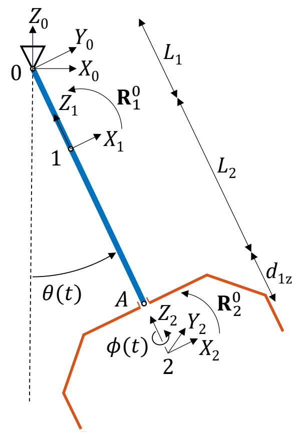


**Figure 3: Schematic representation of the Afterburner**

A body is an assembly of multiple parts which have a bonded contact, by using models of bodies fewer generalized coordinates are required to describe the rigid body motion and the flexible deformations. The two bodies of the Afterburner results in twelve degrees of freedom in three-dimensional space. Allowing two motions (swinging and rotation) indicates that it's a 2-DOF system. Figure 3 shows a schematic representation of the fairground attraction with the two bodies and the global and local Cartesian coordinate systems. The global coordinate system is placed in point 0, where the clapper is fixed to the ground with a revolute joint allowing for the swinging motion of the system around the $Y_0$-axis. The local coordinate systems are rigidly attached in the center of mass of both bodies, respectively point 1 and 2 indicated with the dimensions. In point $A$ the hub is attached to the clapper with a revolute joint, allowing for the rotation of the hub around the clapper, the local $Z_1$-axis.

# 3. THE FLEXIBLE MODEL

## 3.1 SINGLE FLEXIBLE BODY

Figure 4 shows an arbitrary flexible body, with a global frame $P_i$, a local floating frame $P_j$ which is fixed to the body and one interface point $P_k$. The blue dotted body is the rigid body, whereas the blue solid body is the flexible deformed body. The position vector $\mathbf{q}_j^{i,i}$ defines the position of $P_j$ (lower index $j$) relative to $P_i$ (second upper index $i$) and its components are expressed in the coordinate system $P_i$ (first upper index $i$). The rotation matrix $\mathbf{R}_j^i$ defines the orientation of coordinate system $P_i$ (upper index $i$) relative to coordinate system $P_j$ (lower index $j$). This rotation matrix also defines a coordinate transformation of a vector expressed in coordinate system $j$ into one in coordinate system $i$.

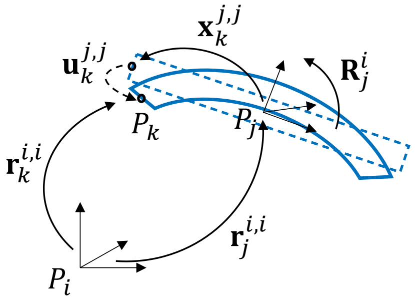


Figure 4: Position of an arbitrary point $P_k$ on a deformed body

The rigid body motion is described globally in frame $P_i$, using the position vector $\mathbf{r}_j^{i,i}$ for the position of the floating frame and the rotation matrix $\mathbf{R}_j^i$ for the orientation of this frame. The flexible behavior is described locally in frame $P_j$, where the position vector $\mathbf{r}_k^{j,j}$ is defined using a superposition of the position of $P_k$ on the undeformed body ($\mathbf{x}_k^{j,j}$) and the linear elastic displacement field ($\mathbf{u}_k^{j,j}$):

$$\mathbf{r}_k^{j,j} = \mathbf{x}_k^{j,j} + \mathbf{u}_k^{j,j} \tag{1}$$

The local linear elastic displacement field is generally described by a linear combination of a set of $N$ deformation shapes $\boldsymbol{\psi}_m$ multiplied with the corresponding time dependent flexible coordinate $\eta_m$:

$$\mathbf{u}_k^{j,j} = \sum_{m=1}^{N} \boldsymbol{\psi}_m\left(x_k^{j,j}\right)\eta_m = \boldsymbol{\Psi}_k\boldsymbol{\eta}_k \tag{2}$$

In which all deformation shapes are combined in one matrix $\boldsymbol{\Psi}$, which forms a total set of deformation shapes. Describing the rigid body motion and the flexible behavior together in the global frame can be done using the rotation matrix:

$$\mathbf{r}_k^{i,i} = \mathbf{r}_j^{i,i} + \mathbf{R}_j^i\left(\mathbf{x}_k^{j,j} + \mathbf{u}_k^{j,j}\right) \tag{3}$$

The rotation matrix is an orthogonal matrix of the proper kind, which means that its transpose equals its inverse ($\mathbf{R}_j^{i\,-1} = \mathbf{R}_j^{i\,T} = \mathbf{R}_i^j$) and its determinant equals one ($\mathbf{R}_j^i\mathbf{R}_i^j = \mathbf{1}$), where $\mathbf{1}$ is the $(3 \times 3)$ unity matrix. From these properties, it follows that the time derivatives of the rotation matrix can be expressed as:

$$\dot{\mathbf{R}}_j^i = \tilde{\boldsymbol{\omega}}_j^{i,i}\mathbf{R}_j^i \qquad\qquad \ddot{\mathbf{R}}_j^i = \tilde{\boldsymbol{\alpha}}_j^{i,i}\mathbf{R}_j^i + \tilde{\boldsymbol{\omega}}_j^{i,i}\tilde{\boldsymbol{\omega}}_j^{i,i}\mathbf{R}_j^i \tag{4}$$

In which vector $\boldsymbol{\omega}_j^{i,i}$ is the angular velocity vector and $\boldsymbol{\alpha}_j^{i,i}$ is the angular acceleration vector of frame $P_j$ with respect to $P_i$ and its components expressed in frame $P_i$. Whereas the tilde operator denotes the skew symmetric matrix constructed from a $(3 \times 1)$ vector as shown:

$$\mathbf{a} = \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \end{Bmatrix} \qquad\qquad \tilde{\mathbf{a}} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{5}$$

Differentiating equation (3) with respect to time gives the velocity of point $\mathbf{r}_k^{i,i}$, and a second time derivative gives the acceleration:

$$\begin{gathered} \dot{\mathbf{r}}_k^{i,i} = \dot{\mathbf{r}}_j^{i,i} + \dot{\mathbf{R}}_j^i\left(\mathbf{x}_k^{j,j} + \boldsymbol{\Psi}_k\boldsymbol{\eta}_k\right) + \mathbf{R}_j^i\left(\boldsymbol{\Psi}_k\dot{\boldsymbol{\eta}}_k\right) \\ \ddot{\mathbf{r}}_k^{i,i} = \ddot{\mathbf{r}}_j^{i,i} + \ddot{\mathbf{R}}_j^i\left(\mathbf{x}_k^{j,j} + \boldsymbol{\Psi}_k\boldsymbol{\eta}_k\right) + 2\dot{\mathbf{R}}_j^i\left(\boldsymbol{\Psi}_k\dot{\boldsymbol{\eta}}_k\right) + \mathbf{R}_j^i\left(\boldsymbol{\Psi}_k\ddot{\boldsymbol{\eta}}_k\right) \end{gathered} \tag{6}$$

## 3.2 FLEXIBLE MULTIBODY SYSTEM

Figure 5 shows a flexible multibody system, consisting of two arbitrary bodies with fixed local frames $P_j$ and $P_l$. They are connected with a revolute joint in $P_k$, where $P_k$ is an interface point on both the bodies. As done before, the position of point $P_k$ can also be described using local frame $P_l$:

$$\mathbf{r}_k^{i,i} = \mathbf{r}_l^{i,i} + \mathbf{R}_l^i \left(\mathbf{x}_k^{l,l} + \mathbf{u}_k^{l,l}\right) \tag{7}$$

Typically, three position constraints are formulated for a joint at interface point $P_k$ by equating equations (3) and (7). This can be done on the position level, but also on the velocity level when taken the first time derivative of these equations, see equation (6). The result must be rewritten to end up with constraint equations in the global coordinate frame $P_i$, as shown in Appendix A.

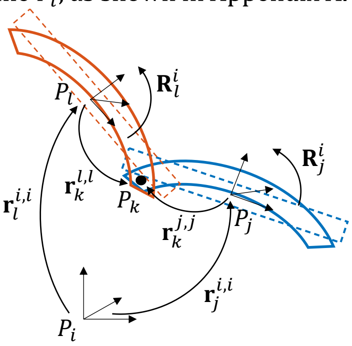


**Figure 5: Flexible multibody system with revolute joint in point $P_k$**

The orientation of coordinate $P_l$ is defined globally by $\mathbf{R}_l^i$, it can also be defined relative to the orientation of frame $P_j$:

$$\mathbf{R}_l^i = \mathbf{R}_j^i \mathbf{R}_l^j \tag{8}$$

For the angular velocities a similar relation holds as for the positions. This relationship also dependents on the flexible coordinates, if the angular velocities are calculated at interface point $P_k$:

$$\boldsymbol{\omega}_k^{i,i} = \boldsymbol{\omega}_j^{i,i} + \mathbf{R}_j^i\left(\boldsymbol{\omega}_k^{j,j} + \boldsymbol{\Psi}_k \dot{\boldsymbol{\eta}}_k\right) \tag{9}$$

This equation can be rewritten in terms of the local angular velocity vector $\boldsymbol{\omega}_k^{j,j}$, from which two angular velocities can be equated to zero. The sixth constraint equation is in general a driving constraint to prescribe the angular velocity. All constraint equations are ensembled in vector $\boldsymbol{\Phi}$.

## 3.3 EQUATION OF MOTION

For each flexible body in a multibody system, the standard equations of motion expressed in the global frame can be written as:

$$\mathbf{R}_j^i \mathbf{M}^j \mathbf{R}_i^j \ddot{\mathbf{q}} + \mathbf{R}_j^i \mathbf{C}^j \mathbf{R}_i^j \dot{\mathbf{q}} + \mathbf{K}^j \mathbf{q} = \mathbf{Q}_\mathrm{A} + \mathbf{Q}_\mathrm{C} \tag{10}$$

In which $\mathbf{M}^j$ and $\mathbf{K}^j$ are respectively the local mass and stiffness matrices, $\mathbf{C}^j$ is known as the velocity dependent matrix (or matrix of fictitious forces) containing Coriolis and gyroscopic effects. Vector $\mathbf{q}$ is a set of generalized coordinates, corresponding to the position and orientation of the floating frame expressed in global frame $i$, including the flexible coordinates ($\boldsymbol{\eta}$) which are described locally. $\mathbf{Q}_\mathrm{A}$ is the vector of externally applied forces and $\mathbf{Q}_\mathrm{C}$ is the vector of constraint forces. The derivation of this equation, based on the principle of virtual work, can be found in multibody dynamics textbooks [1].

The standard equation can be partitioned such that the rigid (r) and flexible (f) parts become visible:

$$\begin{aligned}[\mathbf{R}]\begin{bmatrix}\mathbf{M}_\mathrm{rr} & \mathbf{M}_\mathrm{rf}\\ \mathbf{M}_\mathrm{fr} & \mathbf{M}_\mathrm{ff}\end{bmatrix}[\mathbf{R}]^T\begin{Bmatrix}\ddot{\mathbf{q}}_\mathrm{r}\\ \ddot{\boldsymbol{\eta}}\end{Bmatrix} + [\mathbf{R}]\begin{bmatrix}\mathbf{C}_\mathrm{rr} & \mathbf{C}_\mathrm{rf}\\ \mathbf{C}_\mathrm{fr} & \mathbf{C}_\mathrm{ff}\end{bmatrix}[\mathbf{R}]^T\begin{Bmatrix}\dot{\mathbf{q}}_\mathrm{r}\\ \dot{\boldsymbol{\eta}}\end{Bmatrix} + \begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{K}_\mathrm{ff}\end{bmatrix}\begin{Bmatrix}\mathbf{q}_\mathrm{r}\\ \boldsymbol{\eta}\end{Bmatrix}\\ = \begin{Bmatrix}(\mathbf{Q}_\mathrm{A})_\mathrm{r}\\ (\mathbf{Q}_\mathrm{A})_\mathrm{f}\end{Bmatrix} + \begin{Bmatrix}(\mathbf{Q}_\mathrm{C})_\mathrm{r}\\ (\mathbf{Q}_\mathrm{C})_\mathrm{f}\end{Bmatrix}\end{aligned} \tag{11}$$

In which $\ddot{\mathbf{q}}_\mathrm{r}$ is the absolute coordinate vector of the floating frame and $\mathbf{R}$ is the rotation matrix, transforming the locally defined matrices into global matrices. This rotation matrix is diagonal, consisting of two rotation matrices for the rigid part and a unity matrix for the flexible part. The full derivation of the matrices $\mathbf{M}$ and $\mathbf{C}$, based on the principle of virtual work by inertia forces, can be found in [7]. Because this floating frame is in the center of mass, the rigid matrices $\mathbf{M}_\mathrm{rr}$ and $\mathbf{C}_\mathrm{rr}$ are diagonal and known as:

$$\mathbf{M}_\mathrm{rr} = \begin{bmatrix}\mathbf{m} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\end{bmatrix} \qquad \mathbf{C}_\mathrm{rr} = \begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \tilde{\boldsymbol{\omega}}\mathbf{I}\end{bmatrix} \tag{12}$$

In which $\mathbf{m}$ is the mass of the body and $\mathbf{I}$ the moments of inertia, these are both $(3 \times 3)$ diagonal matrices. For the kinematics the rigid part of the applied force vector only contains gravity terms, for the kinetics this vector also contains the driving forces. The constraint force vector can be calculated at once for both the rigid and the flexible part, using the Jacobian and the Lagrange multipliers. The Jacobian is the first derivative of the constraint vector $(\mathbf{\Phi})$ with respect to the generalized coordinates $(\mathbf{q})$, for clarity the distinction is made between the rigid and flexible part:

$$\mathbf{Q}_{\mathrm{C}} = [\mathbf{\Phi}_{q_{\mathrm{r}}} \quad \mathbf{\Phi}_{\eta}]^T\{\boldsymbol{\lambda}\} \tag{13}$$

The flexible matrices $\mathbf{M}_{\mathrm{ff}}$ and $\mathbf{K}_{\mathrm{ff}}$ are directly obtained from the linear finite element model of the body, on which the model order reduction methods are applied. Matrices $\mathbf{C}_{\mathrm{ff}}$, $\mathbf{C}_{\mathrm{rf}}$ and $\mathbf{C}_{\mathrm{fr}}$ represent the coupling between the rigid and flexible motions due to gyroscopic effects, $\mathbf{C}_{\mathrm{ff}}$ also contains damping terms. $(\mathbf{Q}_{\mathrm{A}})_{\mathrm{f}}$ represents the generalized forces acting on the elastic deformation shapes $\mathbf{\Psi}$. The derivation of this vector, based on the principle of virtual work by external forces, can be found in [7]. For point forces holds that they can be included directly by multiplying them with the value of the deformation shapes at the interface point where they apply, for body forces an integral must be solved.

$\mathbf{M}_{\mathrm{rf}}$ and $\mathbf{M}_{\mathrm{fr}}$ are coupling terms between the rigid and flexible mass matrices. The terms can be calculated separately, but a more practical way is using the Craig-Bampton reduced mass and stiffness matrix of the body's finite element model (respectively $\mathbf{M}_{\mathrm{CB}}$ and $\mathbf{K}_{\mathrm{CB}}$), the deformation shapes $(\mathbf{\Psi}_{\mathrm{flex}})$ and the rigid mode shapes $(\mathbf{\Psi}_{\mathrm{rig}})$. The rigid mode shapes are defined as in equation (14), with N as the number of interface points. By pre- and post-multiplication of the mode shapes, the finite element mass and stiffness matrices become the locally defined mass and stiffness matrices in the floating frame of reference, see (15) and (16). This is possible since the Craig-Bampton modes are able to describe rigid motions.

$$\mathbf{\Psi}_{\mathrm{rig}} = \begin{bmatrix} \begin{bmatrix} \mathbf{1} & -\tilde{\mathbf{r}}_1^{j.j} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \mathbf{1} & -\tilde{\mathbf{r}}_N^{j.j} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \end{bmatrix} \tag{14}$$

$$\mathbf{M}_{\mathrm{FFR}}^{\mathrm{local}} = \begin{bmatrix} \mathbf{\Psi}_{\mathrm{rig}} \\ \mathbf{\Psi}_{\mathrm{flex}} \end{bmatrix}^T \mathbf{M}_{\mathrm{CB}} \begin{bmatrix} \mathbf{\Psi}_{\mathrm{rig}} \\ \mathbf{\Psi}_{\mathrm{flex}} \end{bmatrix} \tag{15}$$

$$\mathbf{K}_{\mathrm{FFR}}^{\mathrm{local}} = \begin{bmatrix} \mathbf{\Psi}_{\mathrm{rig}} \\ \mathbf{\Psi}_{\mathrm{flex}} \end{bmatrix}^T \mathbf{K}_{\mathrm{CB}} \begin{bmatrix} \mathbf{\Psi}_{\mathrm{rig}} \\ \mathbf{\Psi}_{\mathrm{flex}} \end{bmatrix} \tag{16}$$

The equations of motion in (11) cannot be solved on its own, because there are more generalized coordinates and Lagrange multipliers to solve for than there are equations. By expanding the system of equations with the acceleration equation, this problem is solved. The acceleration equation pops up from the total set of constraint equations, denoted in (17). Differentiated with respect to time once, gives the velocity equation, shown in (18). The second time derivative gives the acceleration equation, given in (19).

$$\mathbf{\Phi} = \mathbf{0} \tag{17}$$

$$\mathbf{\Phi}_q\dot{\mathbf{q}} = \boldsymbol{\nu} \quad \boldsymbol{\nu} \equiv -\mathbf{\Phi}_t \tag{18}$$

$$\mathbf{\Phi}_q\ddot{\mathbf{q}} = \boldsymbol{\gamma} \quad \boldsymbol{\gamma} \equiv -\left[\mathbf{\Phi}_q\dot{\mathbf{q}}\right]_q\dot{\mathbf{q}} - 2\mathbf{\Phi}_{qt}\dot{\mathbf{q}} - \mathbf{\Phi}_{tt} \tag{19}$$

Combining and rewriting equations (11) and (19), gives the constrained equation of motion:

$$\boxed{\begin{bmatrix} \mathbf{M} & \mathbf{\Phi}_q{}^T \\ \mathbf{\Phi}_q & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{Q}_{\mathrm{A}} - \mathbf{C}\dot{\mathbf{q}} - \mathbf{K}\mathbf{q} \\ \boldsymbol{\gamma} \end{Bmatrix}} \tag{20}$$

## 3.4 SOLUTION PROCEDURE

From the theory above, twelve constraint equations can be composed which describe the system of the Afterburner. From a kinematic point of view, the two degrees of freedom results in an equal amount of driving constraint equations.

At the initial position the positions and velocities of the generalized coordinates are known ($\mathbf{q}_0$ and $\dot{\mathbf{q}}_0$). Equation (20) can then be solved for the acceleration $(\ddot{\mathbf{q}})$ and the Lagrange multipliers $(\boldsymbol{\lambda})$. Using the principle of virtual work, the relation between the Lagrange multipliers and the constraint forces can be found, this derivation is added in Appendix B. The Adams-Bashforth predictor and Adams-Moulton corrector methods are applied to calculate the velocities of the generalized coordinates at the next time step. The method is a linear multistep method, despite the Adams-Moulton corrector method is an implicit numerical integration scheme, the predictor-corrector method is zero-stable. This means that the time step should be sufficiently small in order to be stable, depending on the highest natural frequency of the system. It is not required to solve for the positions of the generalized coordinates, since they are not part of the Jacobian.

The standard definitions of the rotation matrix make use of sine and cosine functions, which suffer from singularity problems using large rotations. Therefore, we chose to not parametrize the rotation matrices. It is known that in initial position, when all angles are zero, the rotation matrices are equal to the identity-matrix. For the next time step the rotation matrices must be updated using a Taylor series expansion. The first- and second-time derivatives of the rotation matrices can be found in equation (4), which are used to approximate the next rotation matrix second order accurate using the polynomial:

$$\mathbf{R}|_{t=i+1} = \mathbf{R}|_{t=i} + \dot{\mathbf{R}}\big|_{t=i}\Delta t + \frac{1}{2}\ddot{\mathbf{R}}\big|_{t=i}\Delta t^2 \tag{21}$$

When solving equation (20) for the kinetics, the driving constraint equations are removed and two driving forces will be introduced in the applied force vector. With two equations less, two less Lagrange multipliers are calculated since there are equally less constraint forces.

# 4. MODDELING

Model data of the clapper and hub in the constrained equation of motion is obtained from a finite element model. Because we assume that the flexible deformations in the model remain small, the model can be linear and there is no need for updating each timestep. The mesh is made sufficiently small in order to converge the natural frequencies. The added interface points are part of the finite element mass and stiffness matrices, which are reduced using the Craig-Bampton method. The rigid body motions contained in the Craig-Bampton reduction basis are eliminated by taking the natural frequencies of the reduced system. The interface points are placed where parts connect to each other or to the ground. For the clapper, interface points are also located at the gondolas, close to the position of where the fracture occurs in 2017.

## 4.1 THE CLAPPER

Figure 6 and Figure 7 show the finite element mesh made for the clapper with the positions of the interface points and to which surface they rigidly connect. Six deformation modes are encountered in this model, varying from ~41 Hz up to ~222 Hz, which are: first and second bend modes in $X_1$ and $Y_1$ direction, torsion mode and axial mode. In between are inner modes which are eliminated by the Craig-Bampton method from the reduction basis. They can be calculated by solving the Eigenvalue problem with the interface nodes constrained in each direction. Simulations in Adams showed that these modes do not play a significant role in the flexible calculations of the Afterburner. Further details are given in Appendix C.

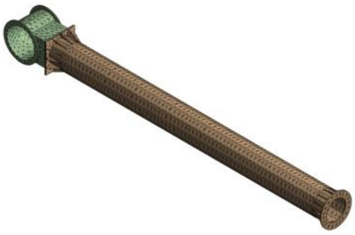

Figure 6: Finite element mesh of the clapper

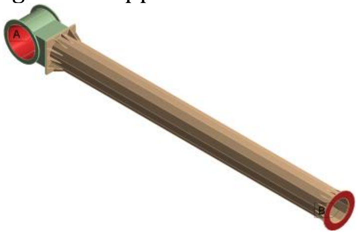

Figure 7: Positions of remote points of the clapper

## 4.2 THE HUB

Figure 8 and Figure 9 show the mesh made for the hub with the positions of the seven interface points and to which surface they rigidly connect. The natural modes are various combinations of deformations of gondola arms. The flexible model contains nine deformation modes, the first five modes are sideways bend modes (~6 Hz) followed by four vertical bend modes (~9 Hz). Further details are given in Appendix D.

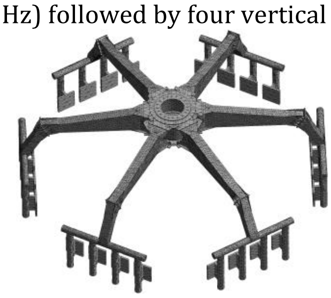

Figure 8: Finite element mesh

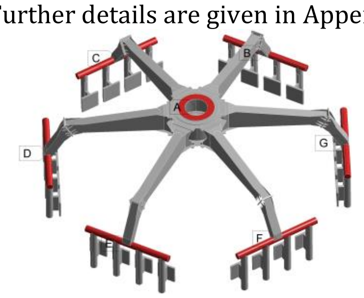

Figure 9: Positions of remote points

# 5. RESULTS

For validation purposes, the presented method in this work is verified with a practical application to the Afterburner. First the solutions of the rigid kinematics and kinetics are shown, followed by the flexible kinematics. The results are compared with simulations in MSC Adams, which also uses a floating frame of reference formulation. As last the results are shown of the dynamic stress analysis at the clapper.

## 5.1 RIGID KINEMATICS

For the kinematics, rigid body motion is prescribed by global angle $\theta(t)$ and local angle $\phi(t)$, which are shown in Figure 3. The simulation duration is 35 seconds in which the system starts up using a sigmoid curve, which is especially important for the flexible calculations. The angles are prescribed using:

$$\theta(t) = \frac{2\pi}{3}\sin(\frac{\pi}{4}t) \tag{22}$$

$$\phi(t) = \frac{\pi}{2}t \tag{23}$$

Figure 10 and Figure 11 give the constraint moments in point 0 and $A$, Appendix F gives additional graphs.

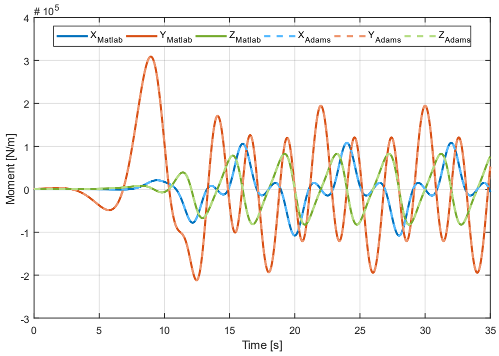


Figure 10: Components of global moment $\mathbf{T}_0^0$

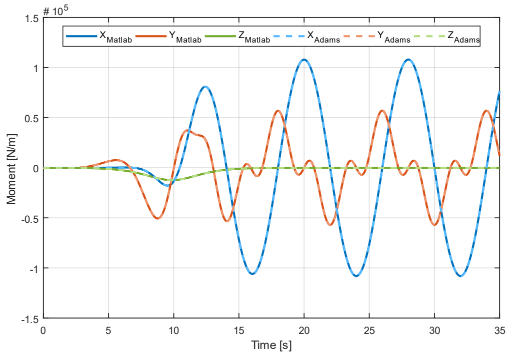


Figure 11: Components of local moment $\mathbf{T}_A^1$

The results from our calculations and the simulations in Adams are equivalent. The error in the calculations is time-step dependent, in this case a timestep of 0.01 seconds is used. In $\mathbf{T}_0^0$ the $X_0$ and $Z_0$-direction show the constraint moments of the gyroscopic effect, just as the $X_1$-direction in $\mathbf{T}_A^1$. It also proves that the applied solution method is valid. Although the update of the rotation matrix results in an error for its orthogonality conditions, this error is dependent on the time-step to the power four and therefore remains neglectable small.

## 5.2 RIGID KINETICS

For the kinetics the fast Fourier transform is used to identify the driving moments $\mathbf{T}_0^0$ in global $Y_0$-direction and $\mathbf{T}_A^1$ in local $Z_1$-direction, details of this transformation are given in Appendix G. The constraint moments in the direction of the applied moments are equal to zero, therefore the applied moments are also shown in Figure 12 and Figure 13.

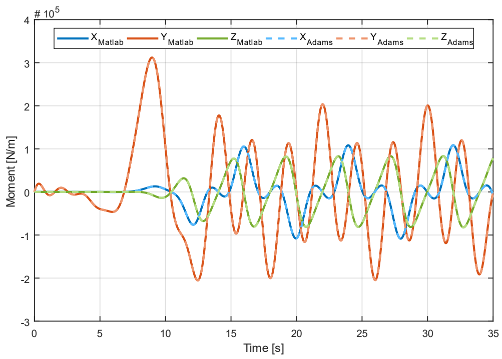


Figure 12: Components of global moment $\mathbf{T}_0^0$

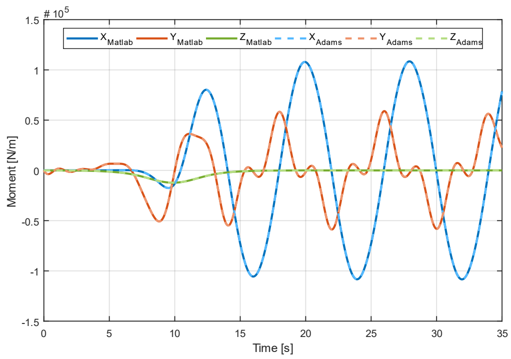


Figure 13: Components of local moment $\mathbf{T}_A^1$

Although the applied moments in this calculation are not exactly the same as the constraint moments calculated in the kinematics, the results from our calculations and Adams are again equivalent using the same time-step as in the kinematics.

## 5.3 FLEXIBLE KINEMATICS

For the flexible calculations it is most convenient to start up the simulation from rest. Steady state is important for all transient analyses, while the flexible coordinates can cause unwanted vibrations in the system. Where the axial mode of the clapper can be approximated in a quasi-static fashion, other modes are harder to determine on beforehand. Therefore, viscous proportional damping is added, which is naturally present, to dissipate energy from the system. Figure 14 shows this effect, where the underdamped system is initially not in steady state and damping ensures the absence of unwanted vibrations.

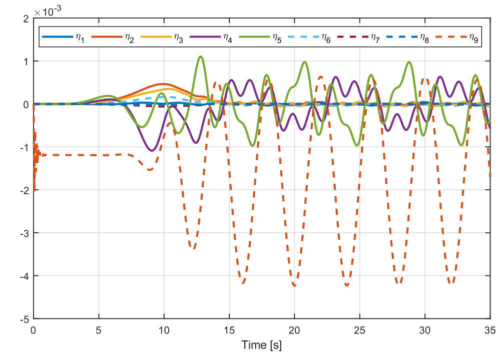


**Figure 14: Flexible coordinates of the hub ($\eta_2$)**

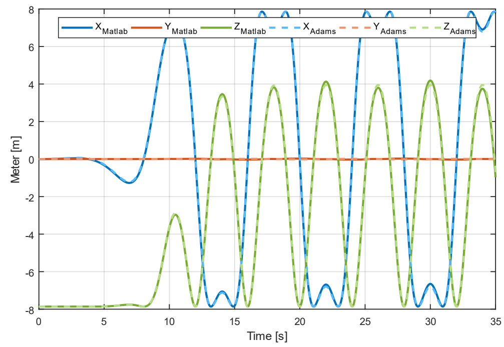


**Figure 15: Position of an interface point on a gondola**

Figure 15 shows the position of an interface point attached to one of the gondolas, where it is shown that our calculation results equal the simulation results of Adams. A small time-step of $5 \times 10^{-7}$ is used for the numerical time integration scheme in order to remain stable, the error builds up in time but for this simulation it remains less than 4% for the positions. This validates the solution procedure used.

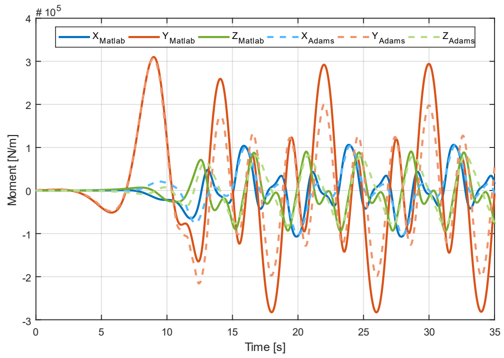


**Figure 16: Components of global moment $\mathbf{T}_0^0$**

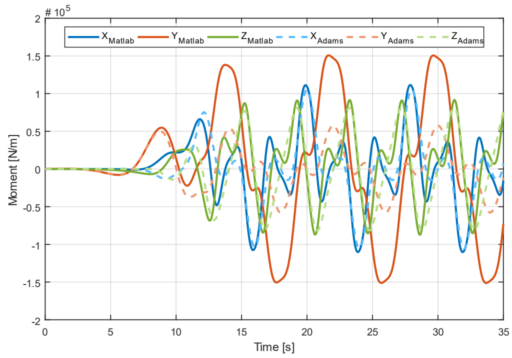


**Figure 17: Components of global moment $\mathbf{T}_A^0$**

Figure 16 and Figure 17 show the calculated constraint moments in point 0 and $A$, where the differences in amplitude of the $Y_0$-direction are clearly visible in both plots. The main reason for this is a somewhat more flexible model in our calculations, which results in larger deformations. Important are the shapes of the functions which are equivalent for all directions.

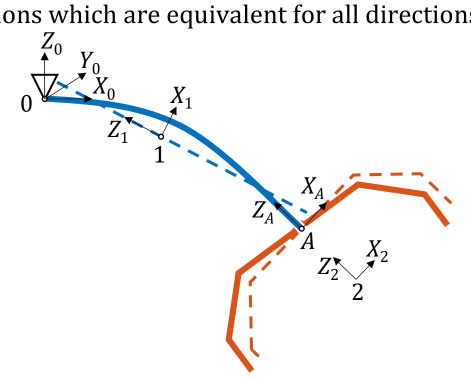


**Figure 18: Schematic representation of the deformed Afterburner**

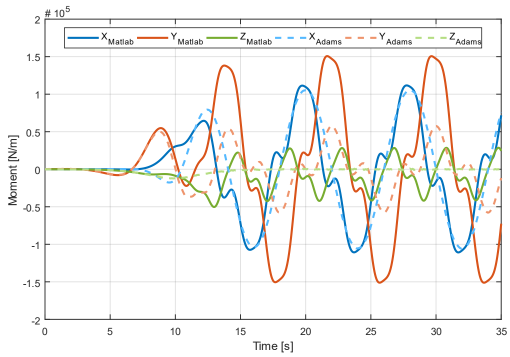


**Figure 19: Components of local moment $\mathbf{T}_A^A$**

Figure 18 shows a local coordinate frame A which is fixed to the flexible body of the clapper, this frame is used to express the constraint moments in Figure 19. Here it is visible that in the $Z_A$-direction an error occurs between our calculations and simulations in Adams. A torsion profile develops, with damping we made sure this is not due to the torsion mode of the clapper and we are sure that there is no imbalance in the system. The rotation matrices ($R_1^0$, $R_2^0$ and $R_A^0$) look correct and are orthogonal through time.

The systematic approach with intermediate results should prevent errors in the mathematical model. The position results of the interface points and the global moments proved that no errors are made in the solution procedure and thus only in the post-processing phase. Additional plots are added in Appendix H.

### 5.4 DYNAMIC STRESS ANALYSES

After calculating the flexible behavior of the Afterburner dynamically through time, also the stresses in the bodies can be calculated in a dynamic fashion. In Adams this is done using so called stress modes, which are derivatives of the deformation modes. Another method is using our results in a finite element package and applying the deformations to the interface coordinates. Appendix I gives additional explanation and results.

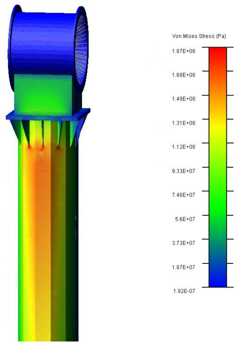

Figure 20: Von-Mises stress distribution in the clapper using Adams

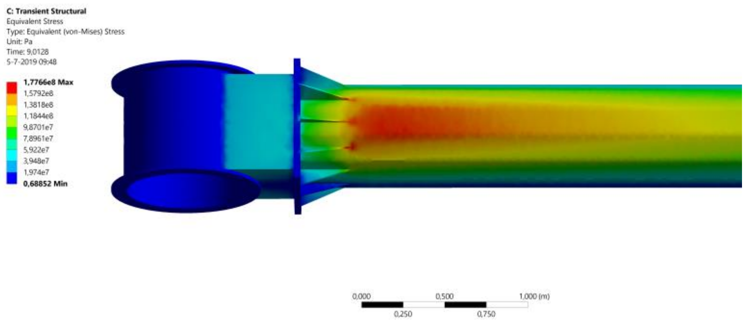

Figure 21: Von-Mises stress distribution in the clapper using Ansys

Figure 20 and Figure 21 show both methods applied to the clapper of the Afterburner and showing equivalent results of a maximum Von-Misses stress of ~187 MPa using Adams and ~178 MPa using Ansys. Furthermore, the region where the maximum equivalent stress occurs is equal, just as the stress distribution through the clapper. Figure 22 shows the Von-Mises stress as a function of time in the node where the maximum stress occurs. Clearly visible is the peak stress in the startup procedure and in the remaining multiple peaks with relatively small differences. Overall, the periodic motion looks similar.

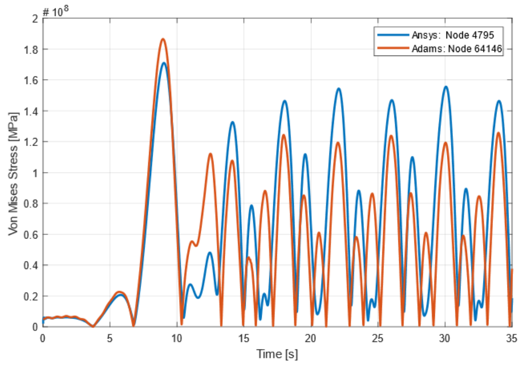


Figure 22: Von-Mises stress in a node through time

## 6. CONCLUSIONS

The most important assumption for this research is that deformations within the bodies remain small, in the results this is confirmed for the Afterburner. It is important to validate this assumption, because otherwise the linear theory of elasticity does not hold and stress stiffening effects play an important role.

The combination of the floating frame formulation and the Craig-Bampton reduction method results in a stable method to describe flexibility with a small number of flexible coordinates. Where the mass and stiffness matrices of each flexible body can be obtained from a linear finite element mesh.

Although the constraint moments are not calculated entirely correct in the post-processing phase, it is shown that the solution procedure described in this work can solve for the deformations with an acceptable error. There are differences between the calculated stresses using Adams and Ansys, but overall can be concluded that the results are equivalent. The method to describe large angles for revolute joints, using local coordinate frames and constraint equations defined on the velocity level, has proven to work. Also, the update of the rotation matrix has proven to work, where orthogonality conditions remain correct

## 7. RECOMMENDATIONS

As shown in the results, the flexible behavior in our calculations is larger than the flexible behavior in Adams. To validate both models, it is required to perform measurements on an actual Afterburner. Therefore, in Appendix J a measurement plan is added which shows what needs to be measured, what the results will be and how these results can validate both models.

It is known that the flexible related matrices in the squared velocity matrix ($\mathbf{C}_{ff}$, $\mathbf{C}_{rf}$ and $\mathbf{C}_{fr}$) are neglected, since they are time consuming to calculate and contain only higher order terms. It is generally known that this assumption is valid for systems with small deformations and low-speed angular velocities. Adams does not neglect these terms but approximate them using derivatives of the mass matrix [13]. In order to make this model more general applicable, it is interesting to look for a method to approximate these terms as well.

The constraint forces and moments are calculated in this work using the Lagrange multipliers. Jurnan Schilder proposed a new theory for creating superelements in his thesis [7], using absolute interface coordinates. In this way, it is possible to eliminate the Lagrange multipliers from the constrained equations of motion. Especially modeling the hub would be an interesting case using this theory, since this body has more than two interface points.

It is shown that the equivalent stresses can be calculated using the finite element models of the clapper with small differences. It would also be interesting to use the same procedure to validate the stresses occurring in the hub. As an improvement the Von-Mises stress can plotted as a function of time using an average of multiple nodes in the same region.

## REFERENCES


1. Shabana, A.A., *Dynamics of Multibody Systems*. 2005, New York: Cambridge University Press.
2. CNN, *Tragedy at Ohio State Fair as one person is killed after ride malfunction*. 2017.
3. Meriam, J.L. and L.G. Kraige, *Engineering Mechanics Dynamics*. 2013, Singapore: John Wiley & Sons.
4. Haug, E.J., *Computer-Aided Kinematics and Dynamics of Mechanical Systems*. Vol. 1: Basic Methods. 1989, Massachusetts: Allyn and Bacon.
5. Jonker, J.B., R.G.K.M. Aarts, and J.P. Meijaard, *Flexible multibody dynamics for (control) design purposes: A Finite Element Approach*. 2015, Enschede: University of Twente.
6. Ellenbroek, M. and J. Schilder, *On the use of absolute interface coordinates in the floating frame of reference formulation for flexible multibody dynamics.* Multibody System Dynamics, 2017. **43**: p. 193-208.
7. Schilder, J., *Flexible multibody dynamics: Superelements using absolute interface coordinates in the floating frame formulation*. 2018, University of Twente: Enschede.
8. Shabana, A.A., *Flexible Multibody Dynamics: Review of Pas and Recent Developments.* Multibody System Dynamics, 1997. **1**: p. 189-222.
9. Nowakowski, C., et al., *Model Order Reduction in Elastic Multibody Systems using the Floating Frame of Reference Formulation.* IFAC Proceedings Volumes, 2012. **45**(2): p. 40-48.
10. Johnson, D., *Advanced Structural Mechanics*. 2000, London: Thomas Telford Ltd.
11. Craig, R. and M. Bampton, *Coupling of Substructures for Dynamic Analyses.* AIAA Jornal, 1968. **6**(7): p. 1313-1319.
12. Craig, R. *Coupling of substructures for dynamic analyses: An overview*. in *Structures, Structural Dynamics and Material Conference*. 2000. Atlanta (USA).
13. MSC Software Corporation. *Docs: Using Adams/Flex*. 2014; Docs ID: DOC9304]. Available from: https://simcompanion.mscsoftware.com/infocenter/index?page=content&id=DOC9304&actp=RSS.
14. O'Shea, J.J., et al., *Reference Conditions and Substructuring Techniques in Flexible Multibody System Dynamics.* Journal of Computational and Nonlinear Dynamics, 2018. **13**(4).
15. Géradin, M. and A. Cardona, *Flexible Multibody Dynamices: A Finite Element Approach*. 2001, West Sussex (England): John Wiley & Sons.
16. Arnold, M. and O. Brüls, *Convergence of the generalized-alpha scheme for constrained mechanical systems.* Multibody System Dynamics, Springer Verlag, 2007. **85**: p. 187-202.
17. KMG International BV. *Afterburner*. 2018; Available from: https://www.kmg.nl/en/attractie/after-burner/.
18. ANSYS Inc., *ANSYS Mechanical APDL Modeling and Meshing Guide*, R. 14.5, Editor. 2012: Canonsburg.

# APPENDIX A: REWRITING THE CONSTRAINT EQUATIONS

In here the expression of the constraint equations for the Afterburner are worked out, using the theory of flexible multibody systems. In Figure 23 a schematic representation of the Afterburner is shown.

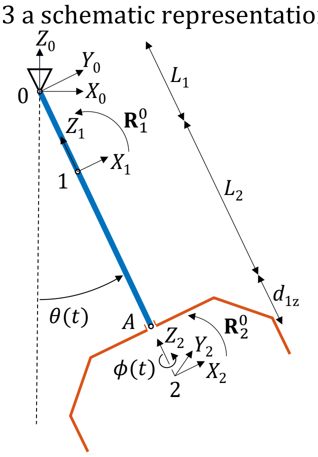


**Figure 23: Schematic representation of the Afterburner**

First the position constraint equations of body 1 with respect to point 0 and body 1 and 2 with respect to point $A$ are defined and rewritten:

$$\begin{gathered}\mathbf{r}_1^{0,0} + \mathbf{R}_1^0 \mathbf{r}_0^{1,1} = \mathbf{0} \\ \mathbf{r}_1^{0,0} + \mathbf{R}_1^0\left(\mathbf{x}_0^{1,1} + \mathbf{\Psi}_1^{\mathrm{t},0}\boldsymbol{\eta}_1\right) = \mathbf{0}\end{gathered} \tag{A.1}$$

$$\begin{gathered}\mathbf{r}_1^{0,0} + \mathbf{R}_1^0 \mathbf{r}_A^{1,1} = \mathbf{r}_2^{0,0} + \mathbf{R}_2^0 \mathbf{r}_A^{2,2} \\ \mathbf{r}_1^{0,0} + \mathbf{R}_1^0\left(\mathbf{x}_A^{1,1} + \mathbf{\Psi}_1^{\mathrm{t},A}\boldsymbol{\eta}_1\right) = \mathbf{r}_2^{0,0} + \mathbf{R}_2^0\left(\mathbf{x}_A^{2,2} + \mathbf{\Psi}_2^{\mathrm{t},A}\boldsymbol{\eta}_2\right) \\ \mathbf{r}_1^{0,0} + \mathbf{R}_1^0\left(\mathbf{x}_A^{1,1} + \mathbf{\Psi}_1^{\mathrm{t},A}\boldsymbol{\eta}_1\right) - \mathbf{r}_2^{0,0} - \mathbf{R}_2^0\left(\mathbf{x}_A^{2,2} + \mathbf{\Psi}_2^{\mathrm{t},A}\boldsymbol{\eta}_2\right) = \mathbf{0}\end{gathered} \tag{A.2}$$

In $\mathbf{\Psi}$, the deformation modes are arranged in columns and in the rows are the degrees of freedom of the interface points. The lower index refers to the relevant body, the fist upper index refers to the translational or rotational coordinates (t or r) and the second index refers to the relevant interface point. $\boldsymbol{\eta}_i$ refers to the flexible coordinates of corresponding body $i$. The rigid lengths are defined as:

$$\mathbf{x}_0^{1,1} = \begin{Bmatrix} 0 \\ 0 \\ L_1 \end{Bmatrix} \qquad \mathbf{x}_A^{1,1} = \begin{Bmatrix} 0 \\ 0 \\ -L_2 \end{Bmatrix} \qquad \mathbf{x}_A^{2,2} = \begin{Bmatrix} 0 \\ 0 \\ d_{1z} \end{Bmatrix} \tag{A.3}$$

The position constraints can be differentiated to time and rewritten to end up with the velocity equations:

$$\begin{gathered}\dot{\mathbf{r}}_1^{0,0} + \dot{\mathbf{R}}_1^0\left(\mathbf{x}_0^{1,1} + \mathbf{\Psi}_1^{\mathrm{t},0}\boldsymbol{\eta}_1\right) + \mathbf{R}_1^0\mathbf{\Psi}_1^{\mathrm{t},0}\dot{\boldsymbol{\eta}}_1 = \mathbf{0} \\ \dot{\mathbf{r}}_1^{0,0} + \tilde{\boldsymbol{\omega}}_1^{0,0} R_1^0\left(\mathbf{x}_0^{1,1} + \mathbf{\Psi}_1^{\mathrm{t},0}\boldsymbol{\eta}_1\right) + \mathbf{R}_1^0\mathbf{\Psi}_1^{\mathrm{t},0}\dot{\boldsymbol{\eta}}_1 = \mathbf{0} \\ \dot{\mathbf{r}}_1^{0,0} + \mathbf{R}_1^0\tilde{\boldsymbol{\omega}}_1^{1,0}\left(\mathbf{x}_0^{1,1} + \mathbf{\Psi}_1^{\mathrm{t},0}\boldsymbol{\eta}_1\right) + \mathbf{R}_1^0\mathbf{\Psi}_1^{\mathrm{t},0}\dot{\boldsymbol{\eta}}_1 = \mathbf{0} \\ \dot{\mathbf{r}}_1^{0,0} - \mathbf{R}_1^0\left(\widetilde{\mathbf{x}_0^{1,1} + \mathbf{\Psi}_1^{\mathrm{t},0}\boldsymbol{\eta}_1}\right)\boldsymbol{\omega}_1^{1,0} + \mathbf{R}_1^0\mathbf{\Psi}_1^{\mathrm{t},0}\dot{\boldsymbol{\eta}}_1 = \mathbf{0} \\ \boxed{\dot{\mathbf{r}}_1^{0,0} - \mathbf{R}_1^0\left(\widetilde{\mathbf{x}_0^{1,1} + \mathbf{\Psi}_1^{\mathrm{t},0}\boldsymbol{\eta}_1}\right)\mathbf{R}_0^1\boldsymbol{\omega}_1^{0,0} + \mathbf{R}_1^0\mathbf{\Psi}_1^{\mathrm{t},0}\dot{\boldsymbol{\eta}}_1 = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}}\end{gathered} \tag{A.4}$$

$$\begin{gathered}\dot{\mathbf{r}}_1^{0,0} + \dot{\mathbf{R}}_1^0\left(\mathbf{x}_A^{1,1} + \mathbf{\Psi}_1^{\mathrm{t},A}\boldsymbol{\eta}_1\right) + \mathbf{R}_1^0\mathbf{\Psi}_1^{\mathrm{t},A}\dot{\boldsymbol{\eta}}_1 - \dot{\mathbf{r}}_2^{0,0} - \dot{\mathbf{R}}_2^0\left(\mathbf{x}_A^{2,2} + \mathbf{\Psi}_2^{\mathrm{t},A}\boldsymbol{\eta}_2\right) - \mathbf{R}_2^0\mathbf{\Psi}_2^{\mathrm{t},A}\dot{\boldsymbol{\eta}}_2 = \mathbf{0} \\ \boxed{\begin{gathered}\dot{\mathbf{r}}_1^{0,0} - \mathbf{R}_1^0\left(\mathbf{x}_A^{1,1} + \mathbf{\Psi}_1^{\mathrm{t},A}\boldsymbol{\eta}_1\right)\mathbf{R}_0^1\boldsymbol{\omega}_1^{0,0} + \mathbf{R}_1^0\mathbf{\Psi}_1^{\mathrm{t},A}\dot{\boldsymbol{\eta}}_1 - \dot{\mathbf{r}}_2^{0,0} \\ + \mathbf{R}_2^0\left(\mathbf{x}_A^{2,2} + \mathbf{\Psi}_2^{\mathrm{t},A}\boldsymbol{\eta}_2\right)\mathbf{R}_0^2\omega_2^{0,0} - \mathbf{R}_2^0\mathbf{\Psi}_2^{\mathrm{t},A}\dot{\boldsymbol{\eta}}_2 = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}\end{gathered}}\end{gathered} \tag{A.5}$$

For the orientation of body 1 the same formulations do not hold, since rotations cannot be expressed as proper vectors. Instead of using the angles, the variation of the orientation can be used in which $\delta\boldsymbol{\pi}$ assembles the variation of the (3×1) vector with the angles:

$$\delta\boldsymbol{\pi}_1^{0,0} + \mathbf{R}_1^0\left(\boldsymbol{\Psi}_1^{\mathrm{r},0}\delta\boldsymbol{\eta}_1\right) = \begin{Bmatrix} 0 \\ \theta(t) \\ 0 \end{Bmatrix} \tag{A.6}$$

The variations of the orientation can be transferred to the angular velocity by differentiating with respect to time, which give similar results:

$$\boxed{\boldsymbol{\omega}_1^{0,0} + \mathbf{R}_1^0\boldsymbol{\Psi}_1^{\mathrm{r},0}\dot{\boldsymbol{\eta}}_1 = \begin{Bmatrix} 0 \\ \dot{\theta}(t) \\ 0 \end{Bmatrix}} \tag{A.7}$$

Note that this constraint equations contain one driving constraint for the swinging motion of the clapper. The last constraint equations are for the orientation of body 2 with respect to body 1, which also contain one driving constraint for the rotation of the hub. Therefore, two local coordinate frames are needed: frame $A$ which is located at point $A$ (see Figure 23) and attached to flexible body 1 and frame $A^*$ which is also located at point $A$ but attached to flexible body 2. The motion is around the local $Z$-axis of these frames, which are collinear. First the angular velocities are defined of point $A$ with respect to body 1 and 2:

$$\begin{aligned} \boldsymbol{\omega}_A^{0,0} &= \boldsymbol{\omega}_1^{0,0} + \mathbf{R}_1^0\boldsymbol{\omega}_A^{1,1} & \qquad \boldsymbol{\omega}_{A^*}^{0,0} &= \boldsymbol{\omega}_2^{0,0} + \mathbf{R}_2^0\boldsymbol{\omega}_{A^*}^{2,2} \\ \boldsymbol{\omega}_A^{0,0} &= \boldsymbol{\omega}_1^{0,0} + \mathbf{R}_1^0\boldsymbol{\Psi}_1^{\mathrm{r},A}\dot{\boldsymbol{\eta}}_1 & \qquad \boldsymbol{\omega}_{A^*}^{0,0} &= \boldsymbol{\omega}_2^{0,0} + \mathbf{R}_2^0\boldsymbol{\Psi}_2^{\mathrm{r},A}\dot{\boldsymbol{\eta}}_2 \end{aligned} \tag{A.8}$$

Next the constraint equations can be defined:

$$\mathbf{R}_0^A\left(\boldsymbol{\omega}_A^{0,0} - \boldsymbol{\omega}_{A^*}^{0,0}\right) = \begin{Bmatrix} 0 \\ 0 \\ \dot{\phi}(t) \end{Bmatrix} \tag{A.9}$$

In this equation a new rotation matrix is used from frame 0 to frame $A$, which is defined as:

$$\begin{aligned} \mathbf{R}_A^0 &= \mathbf{R}_1^0\mathbf{R}_A^1 \\ \mathbf{R}_A^0 &= \mathbf{R}_1^0\left(\mathbf{1} + \tilde{\boldsymbol{\pi}}_A^{1,1}\right) \\ \mathbf{R}_A^0 &= \mathbf{R}_1^0\left(\mathbf{1} + \left(\widetilde{\boldsymbol{\Psi}_1^{\mathrm{r},A}\boldsymbol{\eta}_1}\right)\right) \end{aligned} \tag{A.10}$$

The final constraint equation is written as:

$$\boxed{\mathbf{R}_0^A\boldsymbol{\omega}_1^{0,0} + \mathbf{R}_0^A\mathbf{R}_1^0\boldsymbol{\Psi}_1^{\mathrm{r},A}\dot{\boldsymbol{\eta}}_1 - \mathbf{R}_0^A\boldsymbol{\omega}_2^{0,0} - \mathbf{R}_0^A\mathbf{R}_2^0\boldsymbol{\Psi}_2^{\mathrm{r},A}\dot{\boldsymbol{\eta}}_2 = \begin{Bmatrix} 0 \\ 0 \\ \dot{\phi}(t) \end{Bmatrix}} \tag{A.11}$$

All boxed equations above are constraint equations on the velocity level, which can be combined and rewritten in matrix-vector notation to end up with the velocity equation ($\boldsymbol{\Phi}_q\dot{\mathbf{q}} = -\boldsymbol{\Phi}_t$):

$$\begin{bmatrix} \mathbf{1} & -\mathbf{R}_1^0\left(\widetilde{\mathbf{x}_0^{1,1} + \boldsymbol{\Psi}_1^{\mathrm{t},0}\boldsymbol{\eta}_1}\right)\mathbf{R}_0^1 & \mathbf{0} & \mathbf{0} & \mathbf{R}_1^0\boldsymbol{\Psi}_1^{\mathrm{t},0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{R}_1^0\boldsymbol{\Psi}_1^{\mathrm{r},0} & \mathbf{0} \\ \mathbf{1} & -\mathbf{R}_1^0\left(\widetilde{\mathbf{x}_A^{1,1} + \boldsymbol{\Psi}_1^{\mathrm{t},A}\boldsymbol{\eta}_1}\right)R_0^1 & -\mathbf{1} & \mathbf{R}_2^0\left(\widetilde{\mathbf{x}_A^{2,2} + \boldsymbol{\Psi}_2^{\mathrm{t},A}\boldsymbol{\eta}_2}\right)\mathbf{R}_0^2 & \mathbf{R}_1^0\boldsymbol{\Psi}_1^{\mathrm{t},A} & -\mathbf{R}_2^0\boldsymbol{\Psi}_2^{\mathrm{t},A} \\ 0 & \mathbf{R}_0^A & \mathbf{0} & -\mathbf{R}_0^A & \mathbf{R}_0^A\mathbf{R}_1^0\boldsymbol{\Psi}_1^{\mathrm{r},A} & -\mathbf{R}_0^A\mathbf{R}_2^0\boldsymbol{\Psi}_2^{\mathrm{r},A} \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{r}}_1^{0,0} \\ \boldsymbol{\omega}_1^{0,0} \\ \dot{\mathbf{r}}_2^{0,0} \\ \boldsymbol{\omega}_2^{0,0} \\ \dot{\boldsymbol{\eta}}_1 \\ \dot{\boldsymbol{\eta}}_2 \end{Bmatrix}$$

$$= \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \dot{\theta}(t) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \dot{\phi}(t) \end{Bmatrix} \tag{A.12}$$

For solving the problem, the velocity equation is not needed. But the Jacobian is needed, since this term is part of the constrained equations of motion. Differentiating the equation (A.12) once again with respect to time, will give the acceleration equations ($\boldsymbol{\Phi}_{\mathrm{q}}\ddot{\mathbf{q}} = \boldsymbol{\gamma}$), from which the $\ddot{\mathbf{q}}$ terms and the $\boldsymbol{\gamma}$ terms can be recognized. The last term is most important, since this is also part of the constrained equations of motion. For clarity the $\boldsymbol{\gamma}$ term is shown below.

$$\boldsymbol{\gamma} = \begin{Bmatrix} -\dot{\mathbf{R}}_1^0\left(\widetilde{\mathbf{x}_0^{1,1} + \boldsymbol{\Psi}_1^{\mathrm{t},0}\boldsymbol{\eta}_1}\right)\mathbf{R}_0^1\boldsymbol{\omega}_1^{0,0} - \mathbf{R}_1^0\left(\widetilde{\boldsymbol{\Psi}_1^{\mathrm{t},0}\dot{\boldsymbol{\eta}}_1}\right)\mathbf{R}_0^1\boldsymbol{\omega}_1^{0,0} - \mathbf{R}_1^0\left(\widetilde{\mathbf{x}_0^{1,1} + \boldsymbol{\Psi}_1^{\mathrm{t},0}\boldsymbol{\eta}_1}\right)\dot{\mathbf{R}}_0^1\boldsymbol{\omega}_1^{0,0} + \dot{\mathbf{R}}_1^0\boldsymbol{\Psi}_1^{\mathrm{t},0}\dot{\boldsymbol{\eta}}_1 \\ \dot{\mathbf{R}}_1^0\boldsymbol{\Psi}_1^{\mathrm{r},0}\dot{\boldsymbol{\eta}}_1 \\ -\dot{\mathbf{R}}_1^0\left(\widetilde{\mathbf{x}_A^{1,1} + \boldsymbol{\Psi}_1^{\mathrm{t,A}}\boldsymbol{\eta}_1}\right)\mathbf{R}_0^1\boldsymbol{\omega}_1^{0,0} - \mathbf{R}_1^0\left(\widetilde{\boldsymbol{\Psi}_1^{\mathrm{t,A}}\boldsymbol{\eta}_1}\right)\mathbf{R}_0^1\boldsymbol{\omega}_1^{0,0} - \mathbf{R}_1^0\left(\widetilde{\mathbf{x}_A^{1,1} + \boldsymbol{\Psi}_1^{\mathrm{t,A}}\boldsymbol{\eta}_1}\right)\dot{\mathbf{R}}_0^1\boldsymbol{\omega}_1^{0,0} + \dot{\mathbf{R}}_2^0\left(\widetilde{\mathbf{x}_A^{2,2} + \boldsymbol{\Psi}_2^{\mathrm{t,A}}\boldsymbol{\eta}_2}\right)\mathbf{R}_0^2\boldsymbol{\omega}_2^{0,0} + \mathbf{R}_2^0\left(\widetilde{\boldsymbol{\Psi}_2^{\mathrm{t,A}}\dot{\boldsymbol{\eta}}_2}\right)\mathbf{R}_0^2\boldsymbol{\omega}_2^{0,0} + \mathbf{R}_2^0\left(\widetilde{\mathbf{x}_A^{2,2} + \boldsymbol{\Psi}_2^{\mathrm{t,A}}\boldsymbol{\eta}_2}\right)\dot{\mathbf{R}}_0^2\boldsymbol{\omega}_2^{0,0} + \dot{\mathbf{R}}_1^0\boldsymbol{\Psi}_1^{\mathrm{t,A}}\dot{\boldsymbol{\eta}}_1 - \dot{\mathbf{R}}_2^0\boldsymbol{\Psi}_2^{\mathrm{t,A}}\dot{\boldsymbol{\eta}}_2 \\ \dot{\mathbf{R}}_0^A\boldsymbol{\omega}_1^{0,0} - \dot{\mathbf{R}}_0^A\boldsymbol{\omega}_2^{0,0} + \dot{\mathbf{R}}_0^A\mathbf{R}_1^0\boldsymbol{\Psi}_1^{\mathrm{r,A}}\dot{\boldsymbol{\eta}}_1 + \mathbf{R}_0^A\dot{\mathbf{R}}_1^0\boldsymbol{\Psi}_1^{\mathrm{r,A}}\dot{\boldsymbol{\eta}}_1 - \dot{\mathbf{R}}_0^A\mathbf{R}_2^0\boldsymbol{\Psi}_2^{\mathrm{r,A}}\dot{\boldsymbol{\eta}}_2 - \mathbf{R}_0^A\dot{\mathbf{R}}_2^0\boldsymbol{\Psi}_2^{\mathrm{r,A}}\dot{\boldsymbol{\eta}}_2 \end{Bmatrix} + \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \ddot{\theta}(t) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \ddot{\phi}(t) \end{Bmatrix} \tag{A.13}$$

# APPENDIX B: DERIVATION OF THE FORCE AND MOMENT EXPRESSIONS

In here the expression of the constraint forces and moments are worked out, based on the principle of virtual work by constraint forces. This is done in a practical way using the Afterburner as example, for elastic body 1 at point $A$, see Figure 24. In the equations the orange color denotes flexible terms.

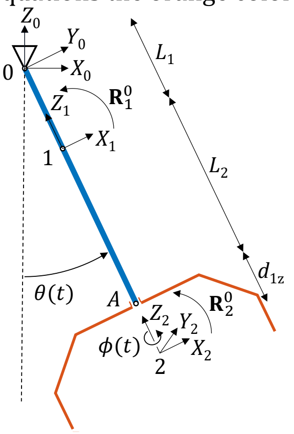


**Figure 24: Schematic representation of the Afterburner**

The equation of motion is for clarity written down below, for only body 1:

$$\mathbf{M}_1\ddot{\mathbf{q}}_1^{0,0} + \mathbf{\Phi}_{qA}^T\boldsymbol{\lambda}_A = \mathbf{Q}_{A,1} - \mathbf{C}_1\dot{\mathbf{q}}_1^{0,0} - \mathbf{K}_1\mathbf{q}_1^{0,0} \tag{B.1}$$

$\mathbf{q}_1^{0,0}$ Are the generalized coordinates of body 1 expressed in global frame 0, which contain the position of the body fixed floating frame ($\mathbf{r}_1^{0,0}$), the orientation of this frame ($\boldsymbol{\pi}_1^{0,0}$) and the flexible coordinates corresponding to body 1 ($\boldsymbol{\eta}_1$). The constraint equations in $\mathbf{\Phi}$ and subsequently the Jacobian ($\mathbf{\Phi}_\mathrm{q}$), are divided into $\mathbf{\Phi}_0$ which are the first six rows related to point 0 and $\mathbf{\Phi}_\mathrm{A}$ which are the last six rows related to point $A$. The same division is made for the Lagrange multipliers.

The equation of motion is pre-multiplied with the transpose of the virtual displacements of the generalized coordinates:

$$\left(\delta\mathbf{q}_1^{0,0}\right)^T\left(\mathbf{M}_1\ddot{\mathbf{q}}_1^{0,0} + \mathbf{C}_1\dot{\mathbf{q}}_1^{0,0} + \mathbf{K}_1\mathbf{q}_1^{0,0} + \mathbf{\Phi}_{qA}^T\boldsymbol{\lambda}_A - \mathbf{Q}_{A,1}\right) = \mathbf{0} \tag{B.2}$$

In which the virtual work due to the constraint forces is only the term with the Lagrange multipliers, since all other components are the forces itself. The constraint forces are in opposite direction and therefore a minus sign arises. Written in matrix-vector notation the columns of the Jacobian can be divided into a translational part ($\mathbf{\Phi}_{\mathrm{q}_{\mathrm{A}_\mathrm{trans}}}$), a rotational part ($\mathbf{\Phi}_{\mathrm{q}_{\mathrm{A}_\mathrm{rot}}}$) and a flexible part ($\mathbf{\Phi}_{\mathrm{q}_{\mathrm{A}_\mathrm{flex}}}$) of body 1 at point $A$:

$$\delta\mathbf{W}_\mathrm{constr} = -\left(\delta\mathbf{q}_1^{0,0}\right)^T\mathbf{\Phi}_{qA}^T\boldsymbol{\lambda}_A = -\begin{bmatrix}\delta\mathbf{r}_1^{0,0^T} & \delta\boldsymbol{\pi}_1^{0,0^T} & \delta\boldsymbol{\eta}_1^T\end{bmatrix}\begin{bmatrix}\mathbf{\Phi}_{\mathrm{q}_{\mathrm{A}_\mathrm{trans}}}^\mathrm{T}\\ \mathbf{\Phi}_{\mathrm{q}_{\mathrm{A}_\mathrm{rot}}}^\mathrm{T}\\ \mathbf{\Phi}_{\mathrm{q}_{\mathrm{A}_\mathrm{flex}}}^\mathrm{T}\end{bmatrix}\boldsymbol{\lambda}_A \tag{B.3}$$

$$\delta\mathbf{W}_\mathrm{constr} = -\delta\mathbf{r}_1^{0,0^T}\mathbf{\Phi}_{\mathrm{q}_{\mathrm{A}_\mathrm{trans}}}^\mathrm{T}\boldsymbol{\lambda}_A - \delta\boldsymbol{\pi}_1^{0,0^T}\mathbf{\Phi}_{\mathrm{q}_{\mathrm{A}_\mathrm{rot}}}^\mathrm{T}\boldsymbol{\lambda}_A - \delta\boldsymbol{\eta}_1^T\mathbf{\Phi}_{\mathrm{q}_{\mathrm{A}_\mathrm{flex}}}^\mathrm{T}\boldsymbol{\lambda}_A \tag{B.4}$$

Note that $\boldsymbol{\eta}_1$ does not describe a virtual displacement, but since the deformation modes are substituted in the constraint equations this multiplication does describe a virtual displacement.

Another way to express the virtual work by constraint forces is the general formula:

$$\delta\mathbf{W}_\mathrm{constr} = \delta\mathbf{r}_A^{0,0^T}\mathbf{F}_A^0 + \delta\boldsymbol{\pi}_A^{0,0^T}\mathbf{M}_A^0 \tag{B.5}$$

In which $\mathbf{F}_A^0$ and $\mathbf{M}_A^0$ are respectively the constraint forces and moments in point 1 expressed in global frame 0. In this equation the virtual displacements can be rewritten:

$$\begin{aligned}\mathbf{r}_A^{0,0} &= \mathbf{r}_1^{0,0} + \mathbf{R}_1^0\mathbf{r}_A^{1,1} \qquad & \delta\boldsymbol{\pi}_A^{0,0} &= \delta\boldsymbol{\pi}_1^{0,0} + \mathbf{R}_1^0\delta\boldsymbol{\pi}_A^{1,1}\\ \delta\mathbf{r}_A^{0,0} &= \delta\mathbf{r}_1^{0,0} + \delta\tilde{\boldsymbol{\pi}}_1^{0,0}\mathbf{R}_1^0\mathbf{r}_A^{1,1} + \mathbf{R}_1^0\delta\mathbf{r}_A^{1,1} & &\end{aligned} \tag{B.6}$$

$$\begin{aligned}\mathbf{r}_A^{1,1} &= \mathbf{x}_A^{1,1} + \mathbf{u}_A^{1,1} = \mathbf{x}_A^{1,1} + \mathbf{\Psi}_1^{\mathrm{t,A}}\boldsymbol{\eta}_1 \qquad & \delta\boldsymbol{\pi}_A^{1,1} &= \mathbf{\Psi}_1^{\mathrm{r,A}}\delta\boldsymbol{\eta}_1\\ \delta\mathbf{r}_A^{1,1} &= \mathbf{\Psi}_1^{\mathrm{t,A}}\delta\boldsymbol{\eta}_1 & &\end{aligned} \tag{B.7}$$

Substituting these relations in equation (B.5) and rewriting gives:

$$\begin{aligned} \delta \mathbf{W}_{\text{constr}} &= \left(\delta \mathbf{r}_1^{0,0^T}\right) \mathbf{F}_A^0 + \left(\mathbf{r}_A^{1,1^T} \mathbf{R}_0^1 \delta \tilde{\boldsymbol{\pi}}_1^{0,0^T}\right) \mathbf{F}_A^0 + \left(\delta \boldsymbol{\eta}_1^T \boldsymbol{\Psi}_1^{\mathrm{t,A}} \mathbf{R}_0^1\right) \mathbf{F}_A^0 \\ &+ \left(\delta \boldsymbol{\pi}_1^{0,0^T}\right) \mathbf{M}_A^0 + \left(\delta \boldsymbol{\eta}_1^T \boldsymbol{\Psi}_1^{\mathrm{t,A}} \mathbf{R}_0^1\right) \mathbf{M}_A^0 \end{aligned} \tag{B.8}$$

$$\begin{aligned} \delta \mathbf{W}_{\text{constr}} &= \left(\delta \mathbf{r}_1^{0,0^T}\right) \mathbf{F}_A^0 + \left(\delta \boldsymbol{\pi}_1^{0,0^T}\right) \left(\mathbf{R}_1^0 \tilde{\mathbf{r}}_A^{1,1} \mathbf{R}_0^1 \mathbf{F}_A^0 + \mathbf{M}_A^0\right) \\ &+ \left(\delta \boldsymbol{\eta}_1^T\right)\left(\boldsymbol{\Psi}_1^{\mathrm{t,A}} \mathbf{R}_0^1 \mathbf{F}_A^0 + \boldsymbol{\Psi}_1^{\mathrm{t,A}} \mathbf{R}_0^1 \mathbf{M}_A^0\right) \end{aligned} \tag{B.9}$$

Equations (B.4) and (B.9) can be equalized to form one new equation, in which both sides the virtual displacements appear. To satisfy this relation, the virtual terms should be equal and in this way the system can be solved for the constraint forces and the constraint moments:

$$\boxed{\mathbf{F}_A^0 = -\boldsymbol{\Phi}_{\mathrm{q}_{\mathrm{A_{trans}}}}^{\mathrm{T}} \boldsymbol{\lambda}_A} \tag{B.10}$$

$$\begin{aligned} \mathbf{R}_1^0 \tilde{\mathbf{r}}_A^{1,1} \mathbf{R}_0^1 \mathbf{F}_A^0 + \mathbf{M}_A^0 &= -\boldsymbol{\Phi}_{\mathrm{q}_{\mathrm{A_{rot}}}}^{\mathrm{T}} \boldsymbol{\lambda}_A \\ \mathbf{M}_A^0 &= \left(-\boldsymbol{\Phi}_{\mathrm{q}_{\mathrm{A_{rot}}}}^{\mathrm{T}} + \mathbf{R}_1^0 \tilde{\mathbf{r}}_A^{1,1} \mathbf{R}_0^1 \boldsymbol{\Phi}_{\mathrm{q}_{\mathrm{A_{trans}}}}^{\mathrm{T}}\right) \boldsymbol{\lambda}_A \\ &\boxed{\mathbf{M}_A^0 = \left(-\boldsymbol{\Phi}_{\mathrm{q}_{\mathrm{A_{rot}}}}^{\mathrm{T}} + \mathbf{R}_1^0 \widetilde{\left(\mathbf{x}_A^{1,1} + \boldsymbol{\Psi}_1^{\mathrm{t,A}} \boldsymbol{\eta}_1\right)} \mathbf{R}_0^1 \boldsymbol{\Phi}_{\mathrm{q}_{\mathrm{A_{trans}}}}^{\mathrm{T}}\right) \boldsymbol{\lambda}_A} \end{aligned} \tag{B.11}$$

The parts which remain in the equations, don't add new information but must be equal to zero. Similar derivations can be made for the forces and moments from body 1 to point 0 and from body 2 to point $A$.

# APPENDIX C: FINITE ELEMENT MODEL OF THE CLAPPER

In here the details of the finite element model of the clapper in Ansys are given. The mesh is made using quadratic tetrahedral elements, which have three degrees of freedom per node $(X, Y, Z)$ and ten nodes per element. Figure 25 shows the mesh, in which 24546 elements are used and 49194 nodes. Figure 26 shows the positions of two remote points ($A$ and $B$) and to which surface they connect marked in red, which form a rigid connection. These remote points have each six degrees of freedom, three translations and three rotations.

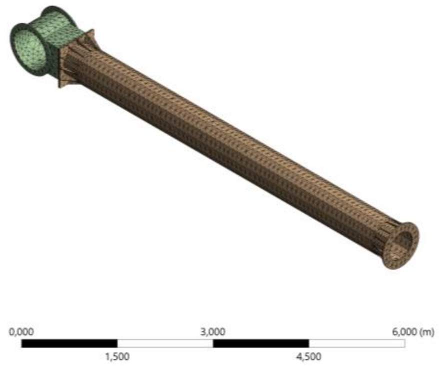


Figure 25: Finite element mesh

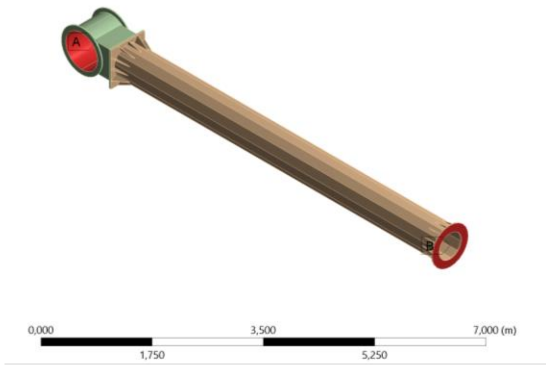


Figure 26: Positions of remote points

This is a bending dominated problem, since the mode shapes contain bend modes. Quadratic elements will yield in general better results than linear elements, due to the extra mid-side nodes extra shape functions. Therefore, a coarser mesh can be made, also with the curved surfaces of the clapper since the mid-side nodes can mimic curved edges for the element. Care must be taken to avoid distorted elements in the mesh (an aspect-ratio close to 1 is preferred), since they decrease the accuracy of the simulation.

When the mass and stiffness matrices are constructed in Matlab, these are diagonal band matrices which are sparse matrices with nonzero terms on a diagonal band. Figure 27 and Figure 28 shows the nonzero terms in the matrices, with the dimensions 143889×143889 which is equal to the degrees of freedom of the finite element mesh. In the bottom rows/columns are the remote points located.

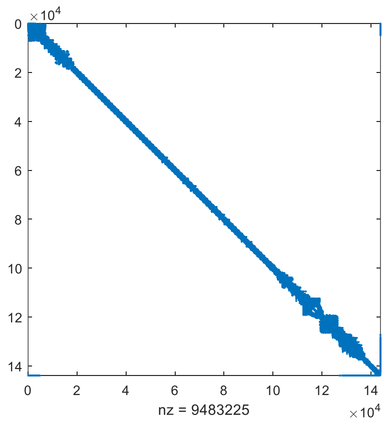


Figure 27: Nonzero terms of the stiffness matrix

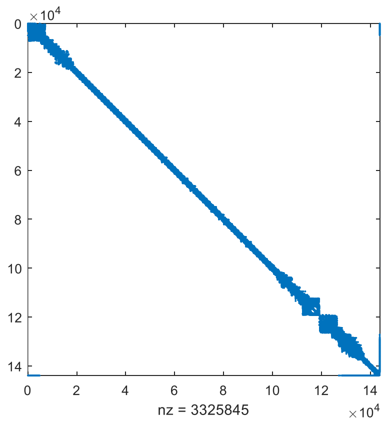


Figure 28: Nonzero terms of the mass matrix

The Eigenfrequencies can be calculated by solving the Eigenvalue problem, which are equal to the Eigenfrequencies calculated by Ansys in the modal analysis. From which the first six frequencies are approximately zero, these are the rigid body modes. In all figures below are the natural modes shown, with in the caption which Eigenfrequency it is, the frequency and what kind of mode it concerned.

Important to note is that there are multiple internal modes between these free boundary modes. For example, the axial mode is only the 32th Eigenfrequency.

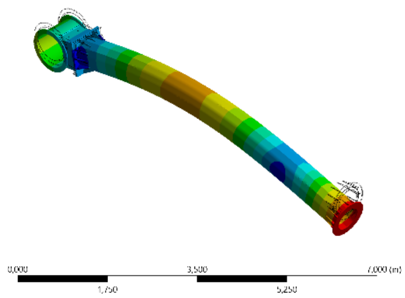


Figure 29: Eigenfrequency 7 – 41.161 Hz - First bend mode in $x$-direction

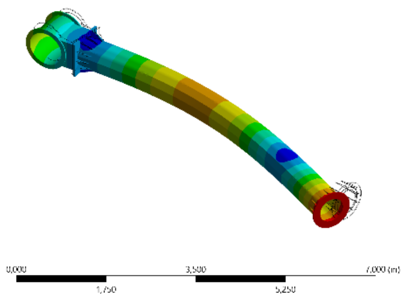


Figure 30: Eigenfrequency 8 – 41.258 Hz - First bend mode in $y$-direction

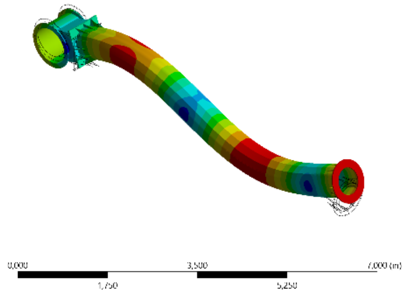


Figure 31: Eigenfrequency 13 – 114.85 Hz - Second bend mode in $x$-direction

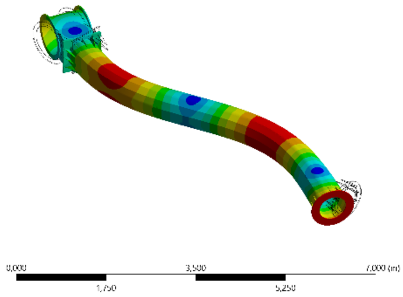


Figure 32: Eigenfrequency 14 – 115.58 Hz - Second bend mode in $y$-direction

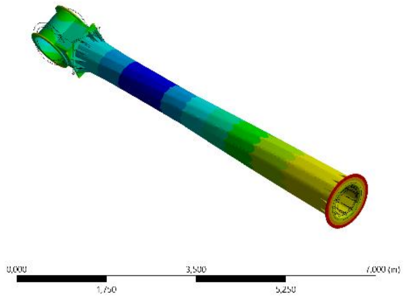


Figure 33: Eigenfrequency 15 – 126.95 Hz – Torsion mode

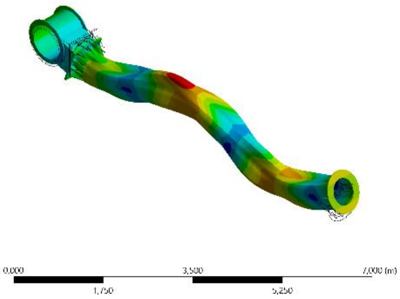


Figure 34: Eigenfrequency 30 – 212.3 Hz - Third bend mode in $x$-direction

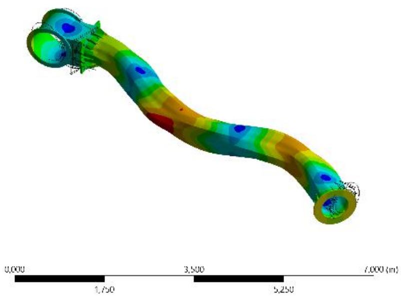


Figure 35: Eigenfrequency 31 – 214.2 Hz - Third bend mode in $y$-direction

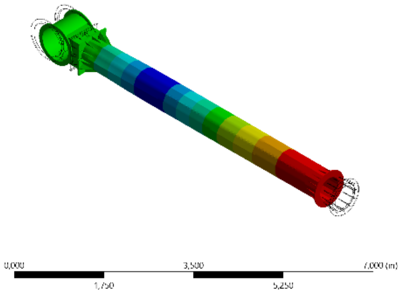


Figure 36: Eigenfrequency 32 – 221.8 Hz – Axial mode

# APPENDIX D: FINITE ELEMENT MODEL OF THE HUB

In here the details of the finite element model of the hub in Ansys are given. For this complex geometry a finer mesh is needed, in order to decrease the degrees of freedom, linear tetrahedral elements are used. They have three degrees of freedom per node ($X$, $Y$, $Z$) and four nodes per element. The natural frequencies in the hub are typically lower, due to a lower stiffness. Also, the differences between natural frequencies are smaller, therefore these can be calculated faster. Figure 37 shows the mesh, in which 140296 elements are used and 46162 nodes. Figure 38 shows the positions of seven remote points and to which surface they connect marked in red, which form a rigid connection. These remote points have each six degrees of freedom, three translations and three rotations. The remote point in the middle ($A$) is an interface point to which the clapper is attached, the other six remote points ($B$-$G$) are located at the end of the gondola arm.

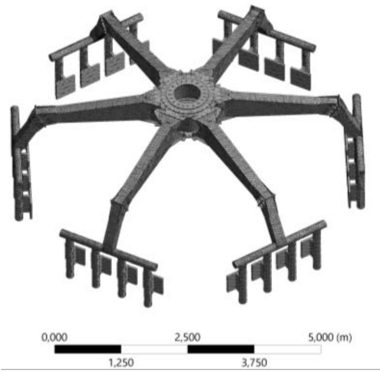


Figure 37: Finite element mesh

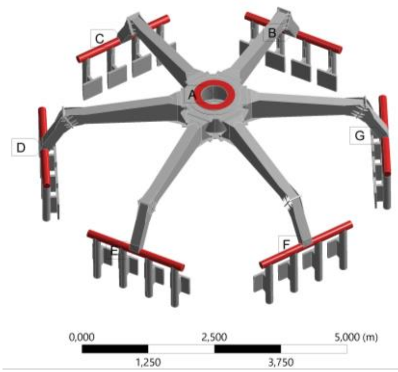


Figure 38: Positions of remote points

Although the quadratic elements yield in general better results than linear elements, in this case linear elements are applied. For the arbitrary shape of the hub, a finer mesh is needed to overcome distorted elements. In Ansys the linear elements have extra shape functions and will therefore yield an accurate solution in a reasonable amount of computational time [18]. Therefore, a finer mesh with linear elements will give more accurate results than a coarser quadratic mesh.

When the mass and stiffness matrices are constructed in Matlab, they are diagonal band matrices which are sparse matrices with nonzero terms on a diagonal band. Figure 39 and Figure 40 shows the nonzero terms in the matrices, with the dimensions 125580×125580 which is equal to the degrees of freedom of the finite element mesh. In the bottom rows/columns are the remote points located.

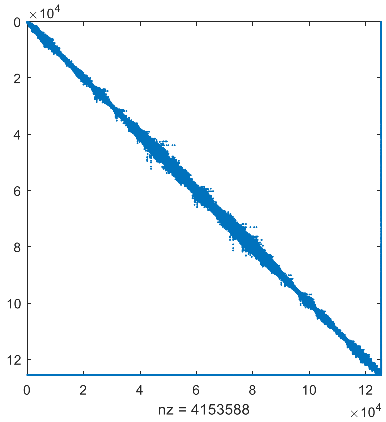


Figure 39: Nonzero terms of the stiffness matrix

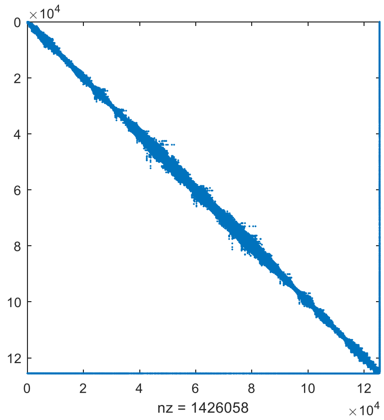


Figure 40: Nonzero terms of the mass matrix

The Eigenfrequencies can be calculated by solving the Eigenvalue problem, which are equal to the Eigenfrequencies calculated by Ansys in the modal analysis. From which the first six frequencies are almost zero, these are the rigid body modes. In all figures below are the first nine modes shown, with in the caption which Eigenfrequency it is and the frequency itself.

Of these modes the first five modes (mode 7-11) are sideways bend modes followed by four vertical bend modes (mode 12-15), both with different combinations between gondola arms. The next two set of modes are a combination of torsion modes with sideways bending (mode 16-21 and approximately 14-16 Hz) and a combination of torsion modes with vertical bending (mode 22-27 and approximately 21-24 Hz). These sets of modes have approximately the same Eigenfrequencies.

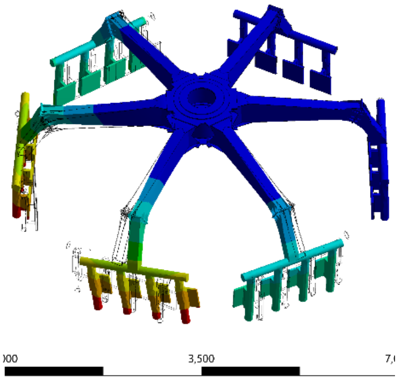

Figure 41: Eigenfrequency 7 – 5.689 Hz

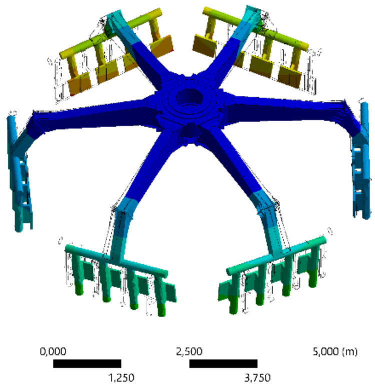

Figure 42: Eigenfrequency 8 – 5.710 Hz

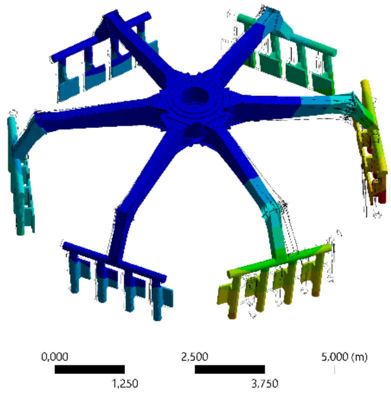

Figure 43: Eigenfrequency 9 – 5.723 Hz

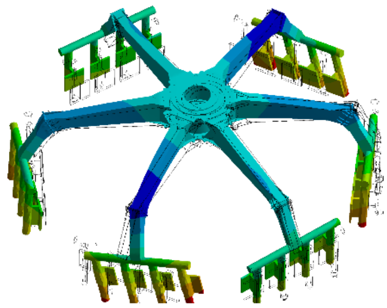
Figure 44: Eigenfrequency 10 – 6.634 Hz

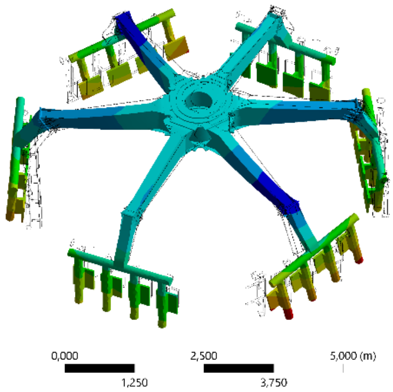

Figure 45: Eigenfrequency 11 – 6.647 Hz

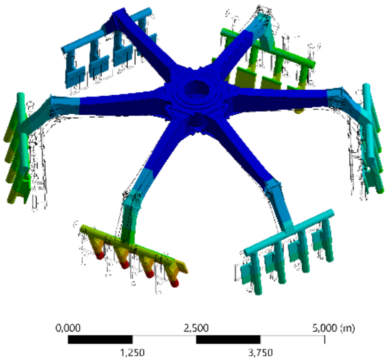

Figure 46: Eigenfrequency 12 – 8.281 Hz

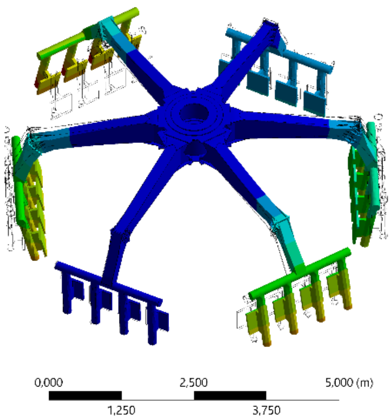

Figure 47: Eigenfrequency 13 – 8.290 Hz

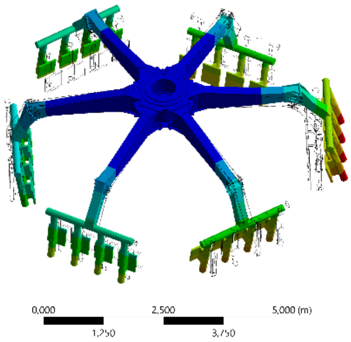

Figure 48: Eigenfrequency 14 – 8.339 Hz

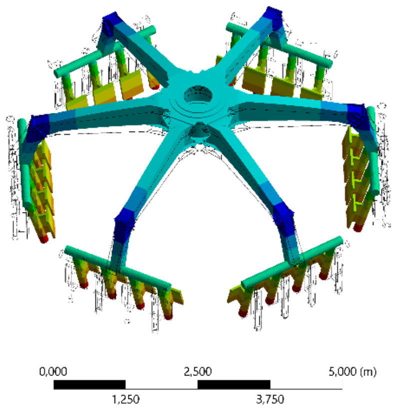

Figure 49: Eigenfrequency 15 – 9.574 Hz

# APPENDIX E: ANSYS MESH TO MATLAB MATRICES

Here the process is explained how to gain a detailed mass and stiffness-matrix in Matlab from a finite element mesh in Ansys. This process can be applied to arbitrary geometry. In the first section it is explained how to gain the mass and stiffness-matrix from Ansys to a text file and in the second section it is explained how this text file can be imported into Matlab.

It is important to note that there must be nodes on the places of the interface points. If for example the interface coordinate must be in the center of a hole, then a *remote point* can be used to create an extra node.

For this instructions **Matlab R2018b** and **Ansys 19.2 (Workbench)** are used. The geometry is drawn in SolidWorks and saved as STEP-file.

## ANSYS MESH TO MASS AND STIFFNESS-MATRIX

In a *modal analysis* an arbitrary geometry is imported on which no constraints are applied. Remote points can be added if necessary and a mesh can be made.

In the Solution tab, *APDL commands* can be inserted and in this the following code can be added:

```
FINISH
/AUX2
file,file,full,
hbmat,HBStiff,txt,,ascii,stiff,yes,yes        ! Generate HBmat stiffness file
hbmat,HBMass,txt,,ascii,mass,yes,yes          ! Generate HBmat mass file


*SMAT,MatK,D,IMPORT,HBMAT,HBStiff.txt,ASCII   ! Import HBmat stiffness file
*print, MatK, matk.txt                        ! Exports stiffness to text file
*export, MatK, mmf, matkMMF.txt               ! Exports stiffness as MMF format

*SMAT,MatM,D,IMPORT,HBMAT,HBMass.txt,ASCII    ! Import HBmat mass file
*print, MatM, matm.txt                        ! Exports mass to text file
*export, MatM, mmf, matmMMF.txt               ! Exports mass as MMF format

FINISH
```

All mass and stiffness information are read from the *file.full* file which Ansys creates while solving the modal analysis. Important is that all calculations are calculated with one core of the computer, otherwise multiple file.full files will be created and then this script doesn't work.

The script uses the *HBmat* file type, because this also gives the nodal vector with all degrees of freedom. Ansys changes the ordering of nodes to solve the modal calculations more efficiently. This file stores the results in a way which is not efficient for Matlab. Therefore, the generated files of HBmat are once again imported and eventually exported as *MMF-format*, which is more efficient for Matlab.

The final output are text files which are saved in a map which can be found in Workbench (dp0/SYS/MECH). The first text file is *"HBMass.mapping"* which is the mapping file, this contains the position of all nodes and degrees of freedom of the nodes in the rows/columns of the mass and stiffness-matrix. *"HBStiff.mapping"* is identical. The second file is *"matmMMF.txt"* which contains the information of the mass-matrix. The first rows in this file contains information about the size of the matrix. Then the matrix itself is shown in the form: row-number, column-number, value of this cell. The last file is *"matkMMF.txt"* which is in the same form but then with the information of the stiffness-matrix.

## TEXT-FILES TO MATLAB

The next step is to import the data into Matlab, when this must be done several times it could be useful to write a script for this process, but it can also be done by hand.

The data can be imported into Matlab using *Import Data* in the Home tab. The file (matkMMF or matmMMF) can be selected and now only the first three columns from row nine till the bottom must be selected. This must be done for both the mass and stiffness data. The results are two matrices in Matlab containing: in column one the row-number, in column two the column-number and in column three the value.

To convert this into the final mass and stiffness-matrices the following code is used:

```
k_mat = sparse(matkMMF(:,1), matkMMF(:,2), matkMMF(:,3)) ;
k_mat = k_mat + k_mat' - diag(diag(k_mat))               ;

m_mat = sparse(matmMMF(:,1), matmMMF(:,2), matmMMF(:,3)) ;
m_mat = m_mat + m_mat' - diag(diag(m_mat))               ;
```

The result is the mass-matrix *"m_mat"* and stiffness-matrix *"k_mat"*. Because these matrices are in general largebig (thousands of rows/columns), they are created as *sparse matrix*. Numeric matrices can give errors with RAM memory when trying to create this size of matrices.

With the first rule of the code only the diagonal and the lower half of the stiffness-matrix is filled. Therefore, a second line adds the transpose to it, to fill the entire matrix. The diagonal is now doubled so only the diagonal must be taken of the resulting matrix. The same process holds for the mass-matrix.

Matlab can calculate very efficiently using sparse matrices. Note that commands for sparse matrices will become slightly different, as an example for calculating the free boundary modes of the model:

```
[V,D] = eigs(k_mat, m_mat, N, 'smallestabs') ;
```

This only calculates the *N* smallest solutions of the Eigenvalue-problem. Otherwise all (thousands) solutions will be calculated, which take lots of additional time. Or the smallest solutions are not calculated, which are most interesting in general.

# APPENDIX F: RESULTS OF RIGID KINEMATICS

## GENERALIZED COORDINATES OF THE CLAPPER

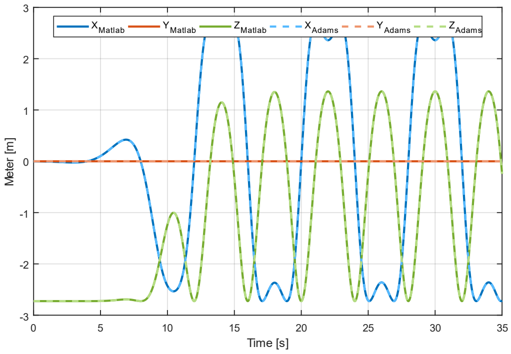


Figure 50: Components of position vector $r_1^{0,0}$

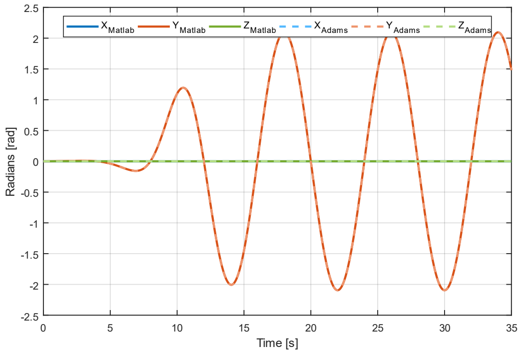


Figure 51: Components of orientation vector $\pi_1^{0,0}$

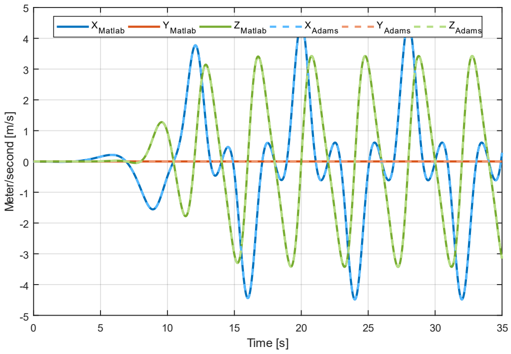


Figure 52: Components of velocity vector $\dot{r}_1^{0,0}$

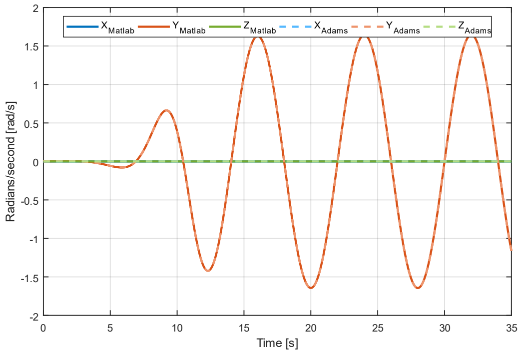


Figure 53: Components of angular velocity vector $\omega_1^{0,0}$

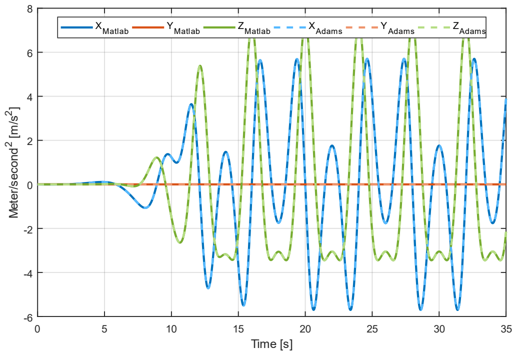


Figure 54: Components of acceleration vector $\ddot{r}_1^{0,0}$

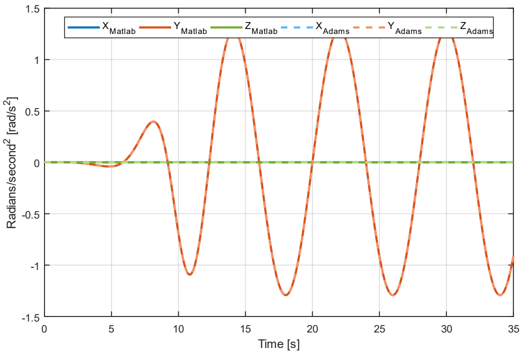


Figure 55: Components of angular accel. vector $\alpha_1^{0,0}$

## GENERALIZED COORDINATES OF THE HUB

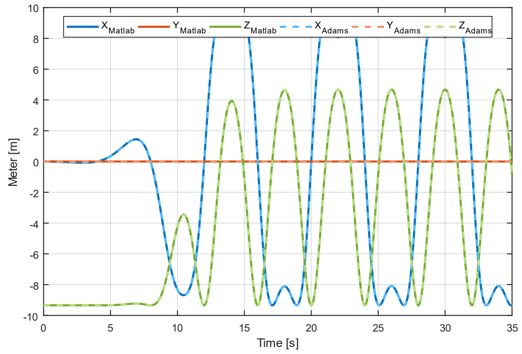


**Figure 56: Components of position vector $r_2^{0,0}$**

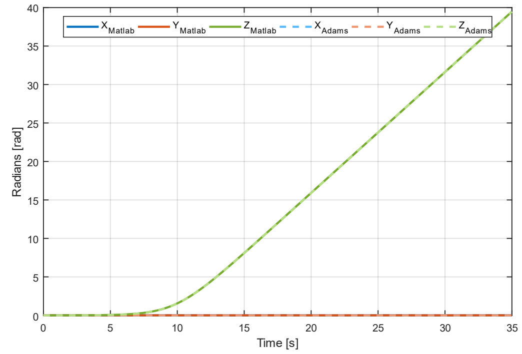


**Figure 57: Components of orientation vector $\pi_2^{1,1}$**

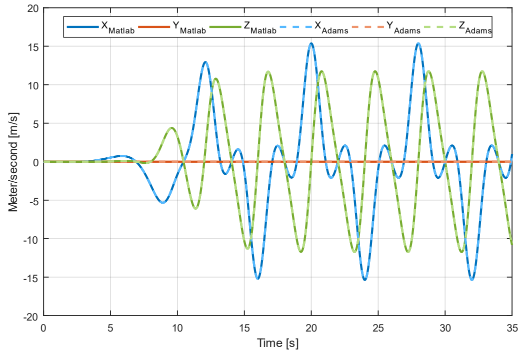


**Figure 58: Components of velocity vector $\dot{r}_2^{0,0}$**

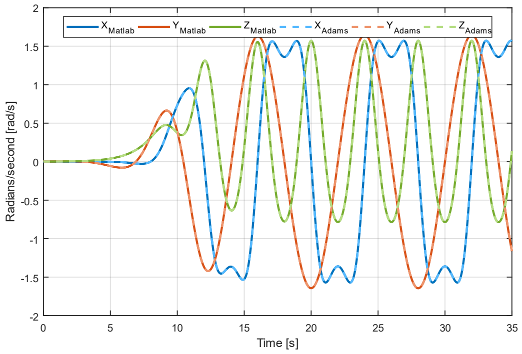


**Figure 59: Components of angular velocity vector $\omega_2^{0,0}$**

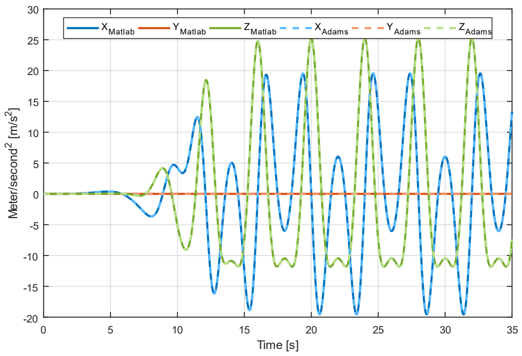


**Figure 60: Components of acceleration vector $\ddot{r}_2^{0,0}$**

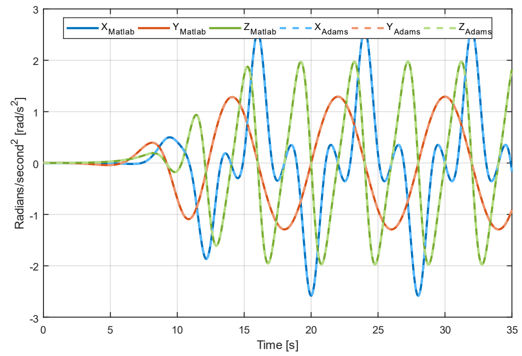


**Figure 61: Components of angular accel. vector $\alpha_2^{0,0}$**

## CONSTRAINT FORCES IN JOINT 0 AND *A*

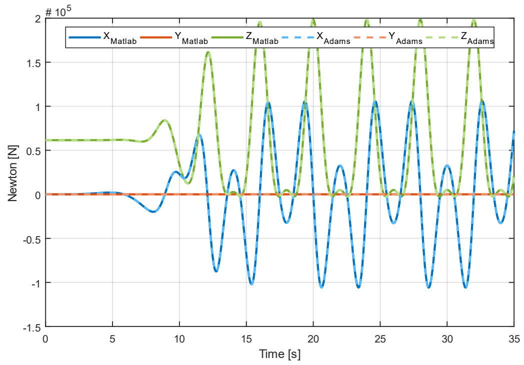


**Figure 62: Components of global force vector $F_0^0$**

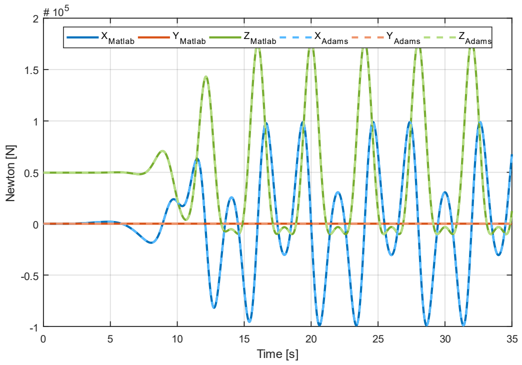


**Figure 63: Components of global force vector $F_A^0$**

# APPENDIX G: RIGID KINEMATICS FOURIER TRANSFORMATION

In here the fast Fourier transformation of the rigid kinematics is explained, which is used to identify the constraint moments and will serve as input for the rigid kinetics.

## TORQUE FOR SWINGING THE CLAPPER

Figure 64 shows the amplitude spectrum of torque $\mathbf{T}_0^0$ in $Y_0$-direction, in here the lowest frequencies have the highest amplitude. Therefore, the first 20 frequencies (0 - 0.55 Hz) are included in the new signal, in order to describe the kinematic calculated signal with enough detail. The result is shown in Figure 65.

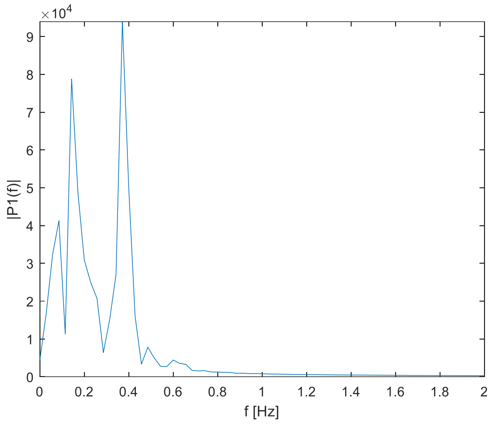


Figure 64: Amplitude spectrum

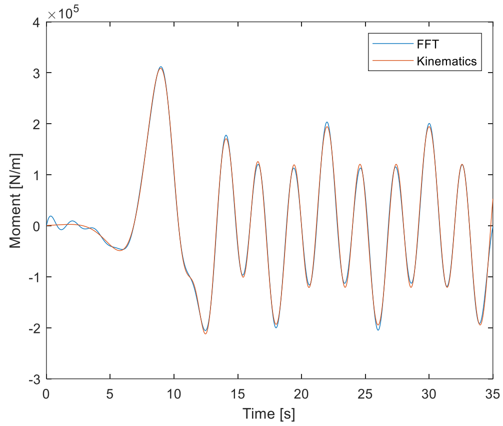


Figure 65: Fourier transform versus kinematic signal

## TORQUE FOR ROTATING THE HUB

Figure 66 shows the amplitude spectrum of torque $\mathbf{T}_A^1$ in local $Z_1$-direction, with one clear peak. Although one peak, the first 10 frequencies (0 - 0.26 Hz) are included in the new signal, in order to describe the kinematic calculated signal with enough detail. The result is shown in Figure 67.

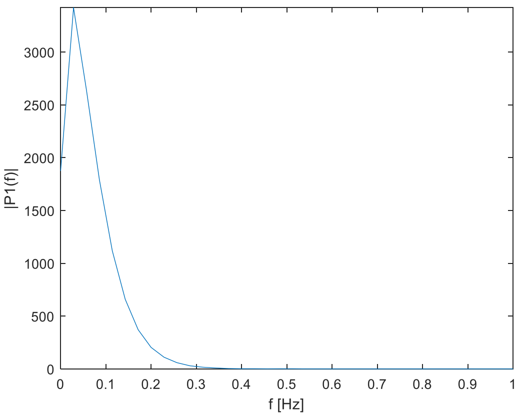


Figure 66: Amplitude spectrum

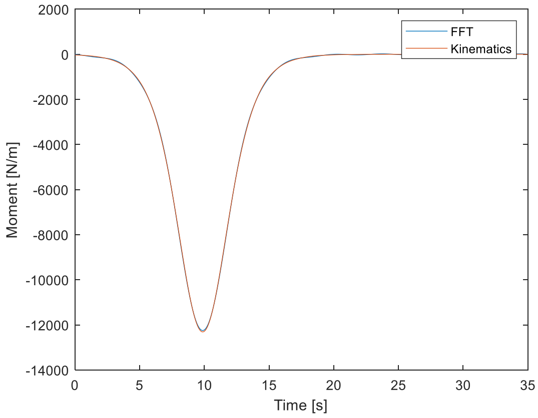


Figure 67: Fourier transform versus kinematic signal

# APPENDIX H: RESULTS OF FLEXIBLE KINEMATICS

It should be noted that the angles in Adams are measured on the joints, therefore these are the local deformed angles. In our calculations we measure the angles of the floating frames, which are slightly different.

## GENERALIZED COORDINATES OF THE CLAPPER

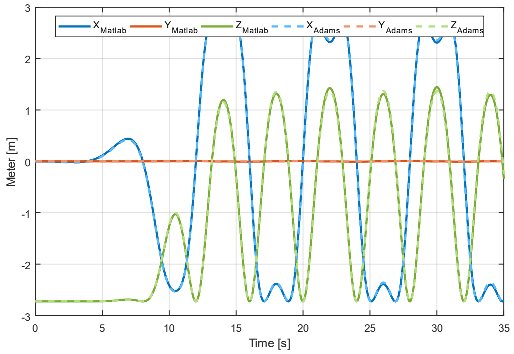


**Figure 68: Components of position vector $\mathbf{r}_1^{0,0}$**

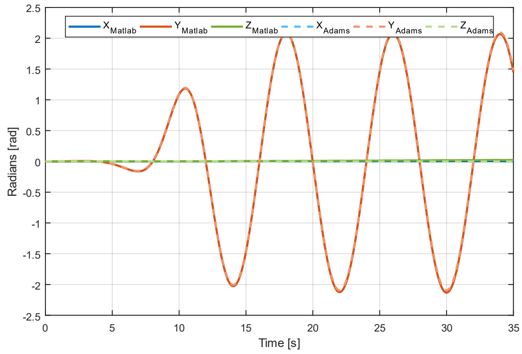


**Figure 69: Components of orientation vector $\boldsymbol{\pi}_1^{0,0}$**

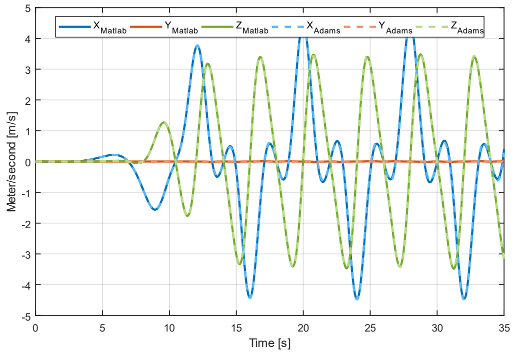


**Figure 70: Components of velocity vector $\dot{\mathbf{r}}_1^{0,0}$**

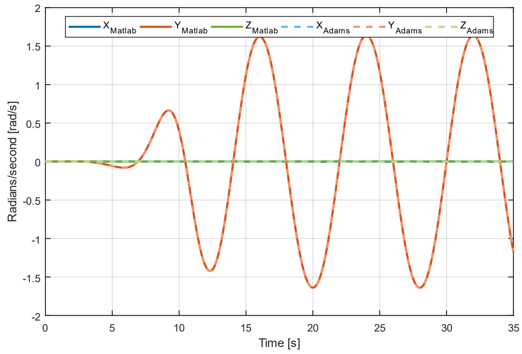


**Figure 71: Components of angular velocity vector $\boldsymbol{\omega}_1^{0,0}$**

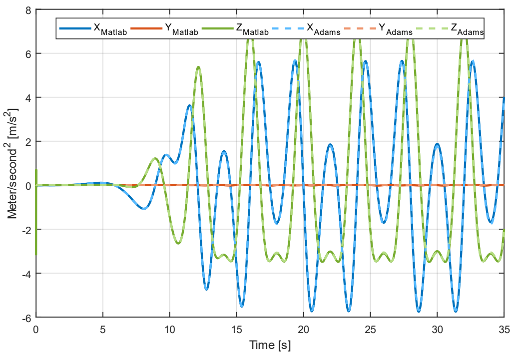


**Figure 72: Components of acceleration vector $\ddot{\mathbf{r}}_1^{0,0}$**

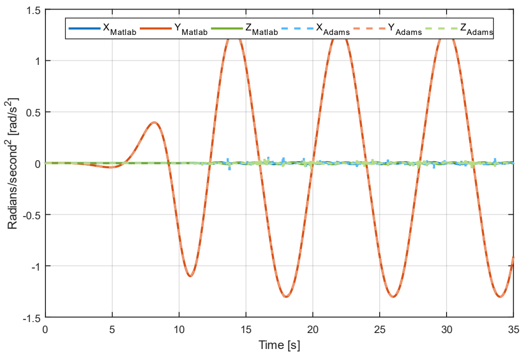


**Figure 73: Components of angular accel. vector $\boldsymbol{\alpha}_1^{0,0}$**

## GENERALIZED COORDINATES OF THE HUB

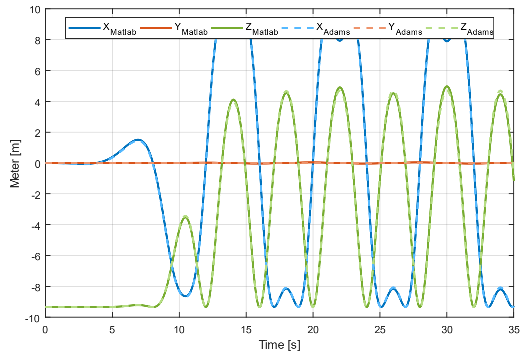


Figure 74: Components of position vector $\mathbf{r}_2^{0,0}$

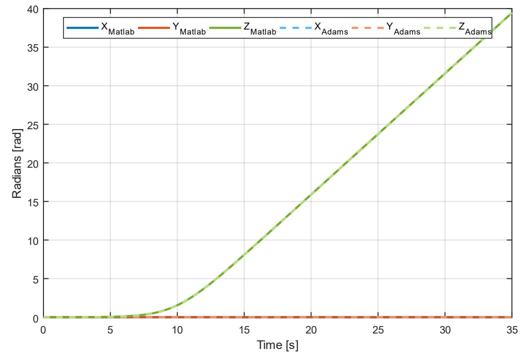


Figure 75: Components of orientation vector $\boldsymbol{\pi}_2^{1,1}$

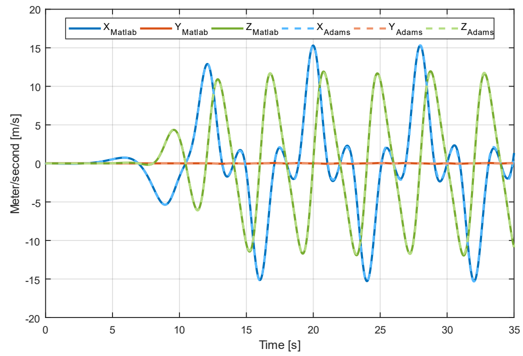


Figure 76: Components of velocity vector $\dot{\mathbf{r}}_2^{0,0}$

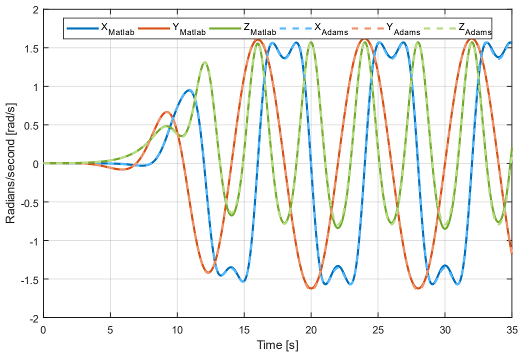


Figure 77: Components of angular velocity vector $\boldsymbol{\omega}_2^{0,0}$

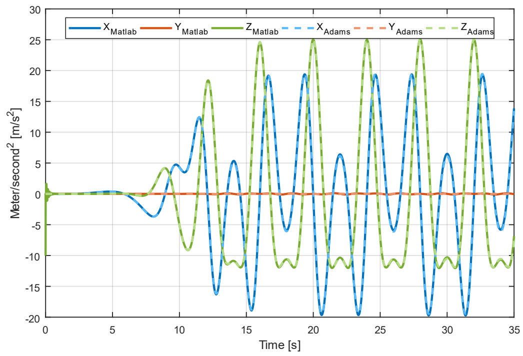


Figure 78: Components of acceleration vector $\ddot{\mathbf{r}}_2^{0,0}$

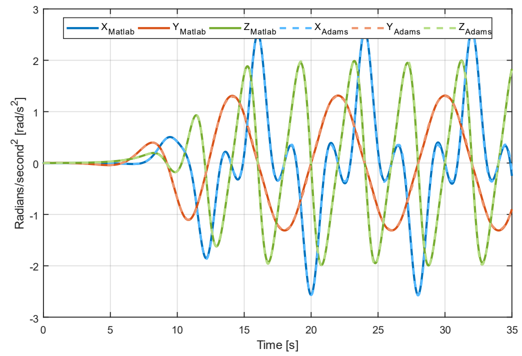


Figure 79: Components of angular accel. vector $\boldsymbol{\alpha}_2^{0,0}$

## CONSTRAINT FORCES IN JOINT 0 AND *A*

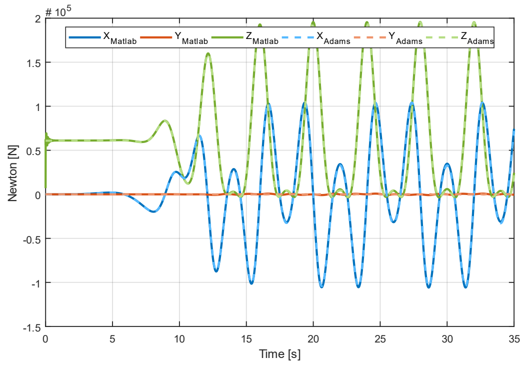


Figure 80: Components of global force vector $\mathbf{F}_0^0$

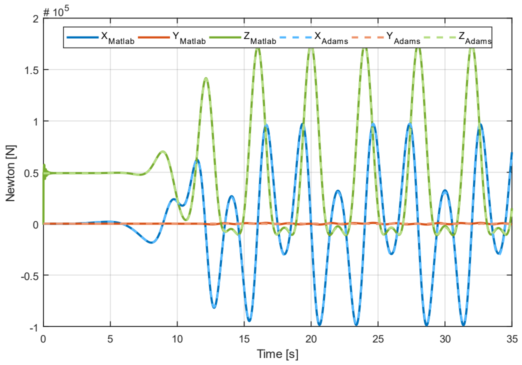


Figure 81: Components of global force vector $\mathbf{F}_A^0$

# FLEXIBLE COORDINATES OF CLAPPER AND HUB IN CALCULATIONS

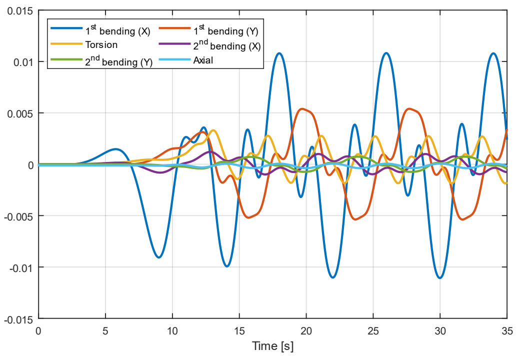


Figure 82: Flexible coordinates of clapper $\eta_1$

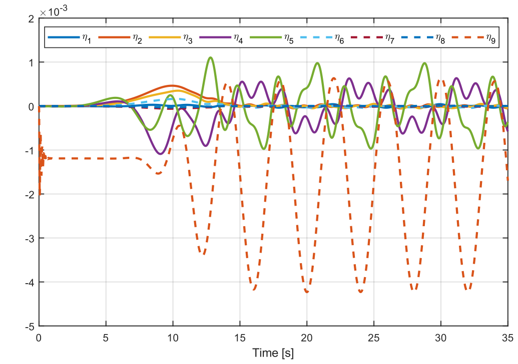


Figure 83: Flexible coordinates of hub $\eta_2$

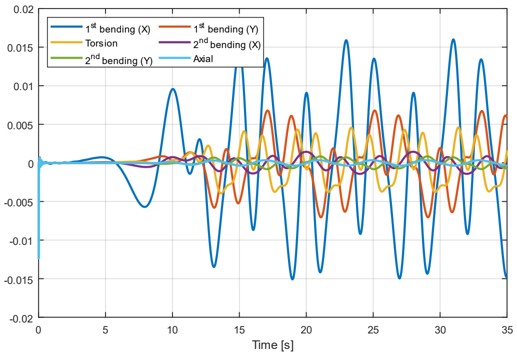


Figure 84: Flexible coordinates of clapper $\dot{\eta}_1$

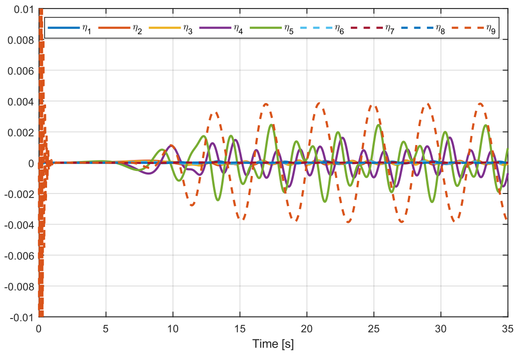


Figure 85: Flexible coordinates of hub $\dot{\eta}_2$

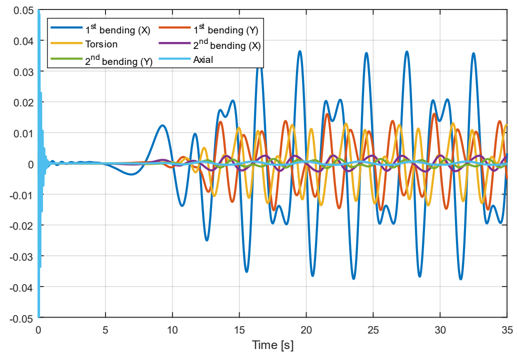


Figure 86: Flexible coordinates of clapper $\ddot{\eta}_1$

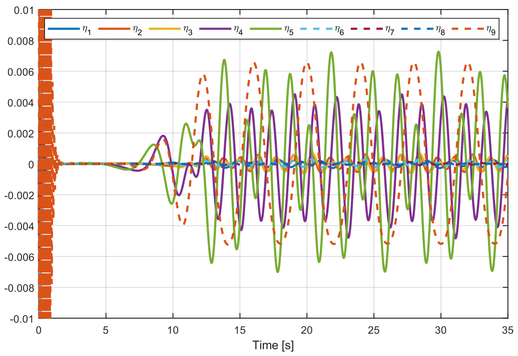


Figure 87: Flexible coordinates of hub $\ddot{\eta}_2$

## FLEXIBLE COORDINATES OF CLAPPER AND HUB IN ADAMS

The modes in Adams are normalized to the mass, while the modes in our calculation are normalized to one. Therefore, they the amplitudes are not equal, but the shapes are equivalent.

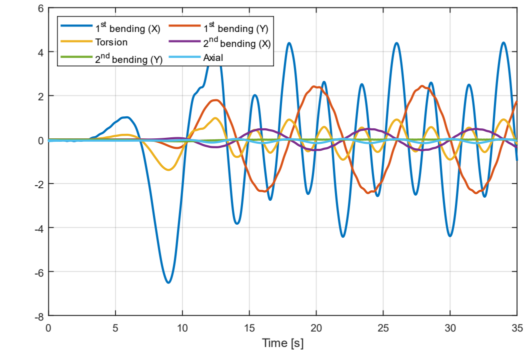


Figure 88: Flexible coordinates of clapper

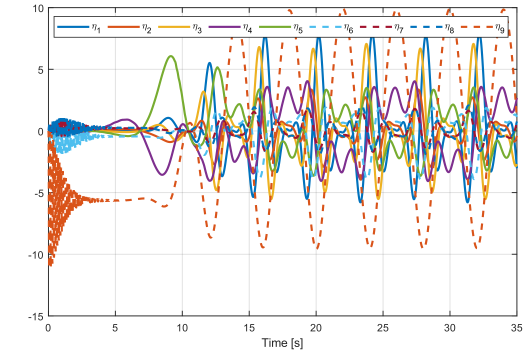


Figure 89: Flexible coordinates of hub

## DIFFERENCES OF GENERALIZED COORDINATES BETWEEN RIGID AND FLEXIBLE POSITIONS

Because the angles in Adams are measured on the joints (the deformed angle), there is no error in here.

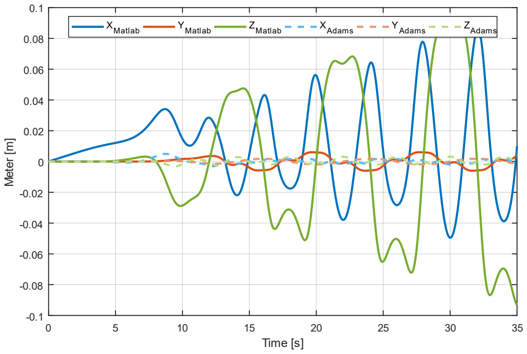


Figure 90: Difference in position vector $r_1^{0,0}$

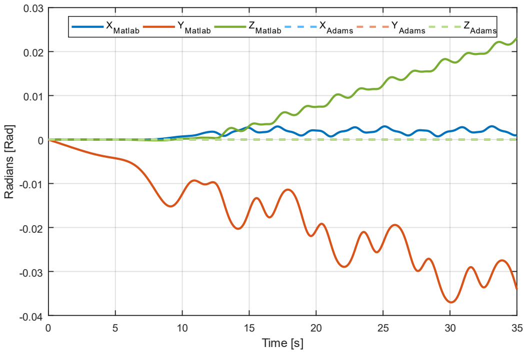


Figure 91: Difference in orientation vector $\pi_1^{0,0}$

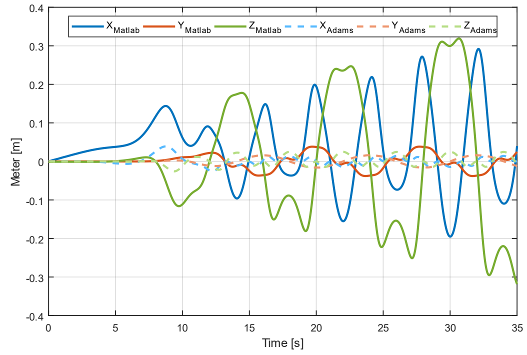


Figure 92: Difference in position vector $r_2^{0,0}$

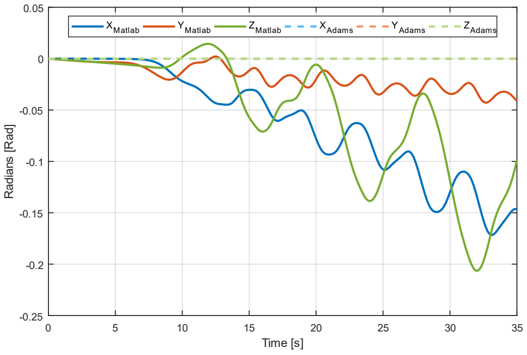


Figure 93: Difference in orientation vector $\pi_2^{0,0}$

## POSITIONS OF INTERFACE POINTS

In here, interface point $A$ is positioned at the bottom of the clapper, points $B$, $C$, $D$, $E$, $F$, $G$ and $H$ correspond to the interface points attached to the six gondolas in the hub.

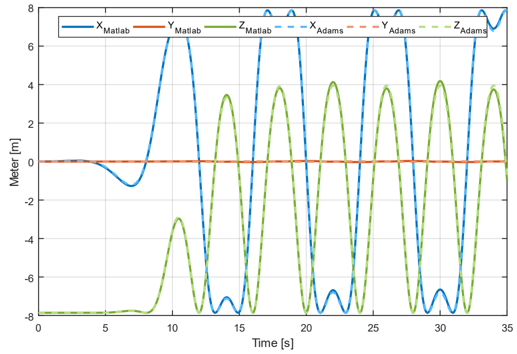


Figure 94: Position of interface point $A$ ($r_A^{0,0}$)

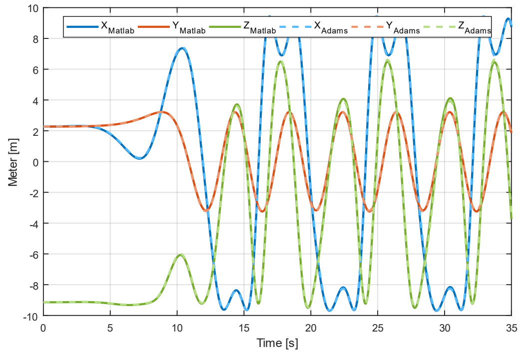


Figure 95: Position of interface point $B$ ($r_B^{0,0}$)

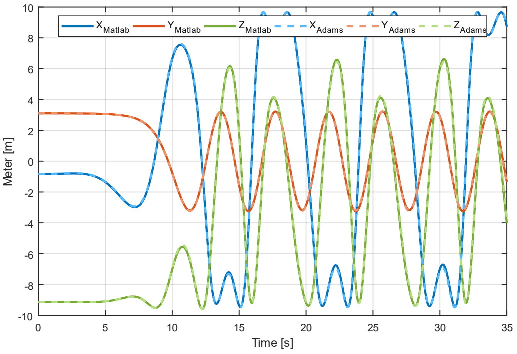


Figure 96: Position of interface point $C$ ($r_C^{0,0}$)

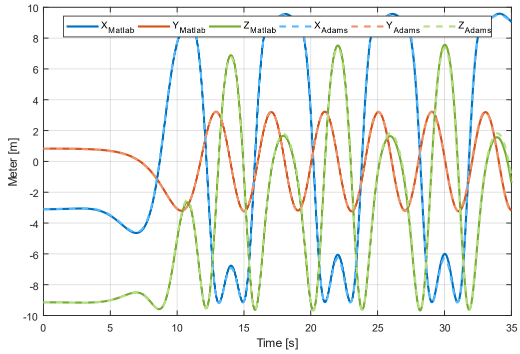


Figure 97: Position of interface point $D$ ($r_D^{0,0}$)

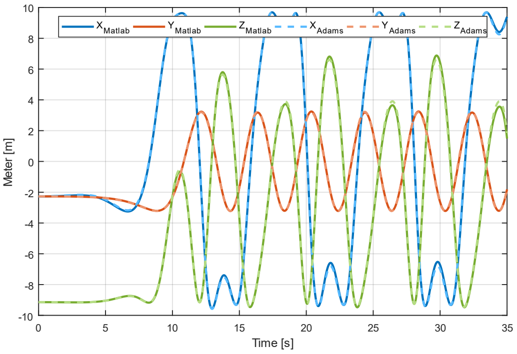


Figure 98: Position of interface point $E$ ($r_E^{0,0}$)

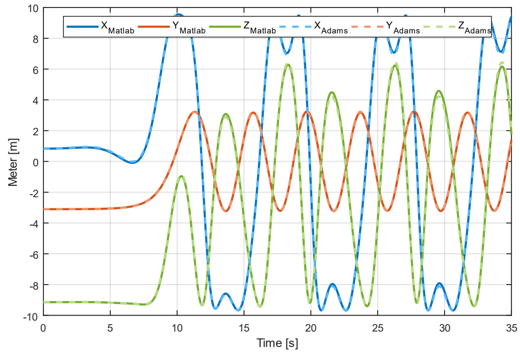


Figure 99: Position of interface point $F$ ($r_F^{0,0}$)

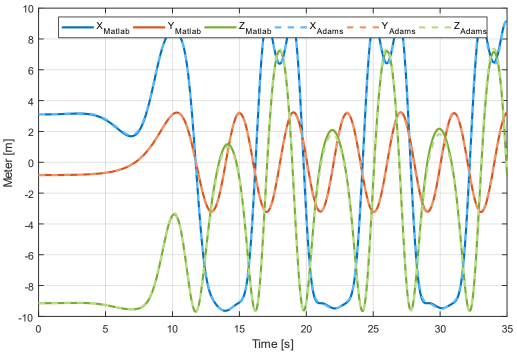


Figure 100: Position of interface point $G$ ($r_G^{0,0}$)

## DIFFERENCES OF INTERFACE POINTS BETWEEN RIGID AND FLEXIBLE POSITIONS

In here, interface point $A$ is positioned at the bottom of the clapper, points $B$, $C$, $D$, $E$, $F$, $G$ and $H$ correspond to the interface points attached to the six gondolas in the hub. To really see the errors made in the two bodies, the errors made in the generalized coordinates of the hub ($\mathbf{r}_2^{0,0}$) are subtracted from the global calculated errors in the interface points of the hub.

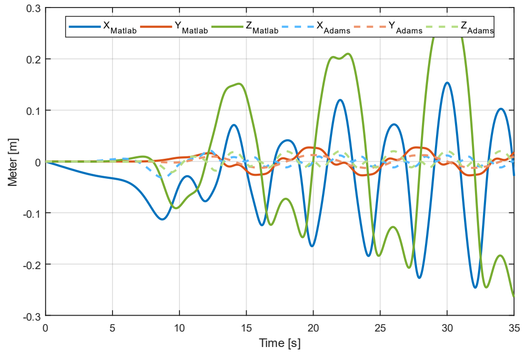


**Figure 101: Difference in position of $A$ ($\mathrm{r}_A^{0,0}$)**

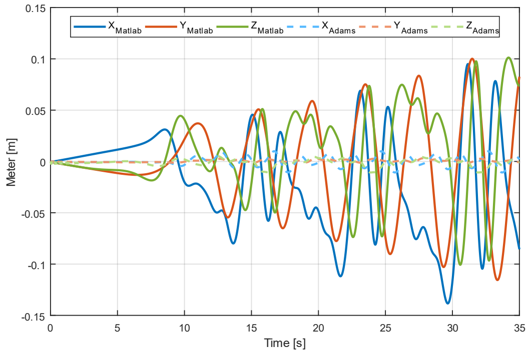


**Figure 102: Difference in position of $B$ ($\mathrm{r}_B^{0,2}$)**

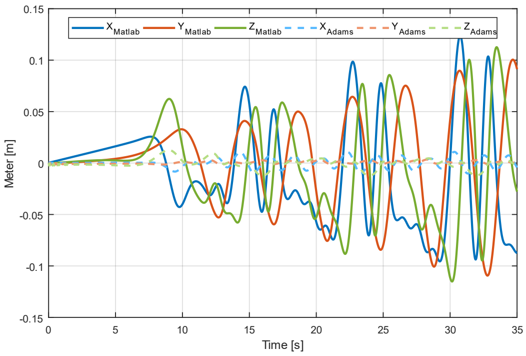


**Figure 103: Difference in position of $C$ ($\mathrm{r}_C^{0,2}$)**

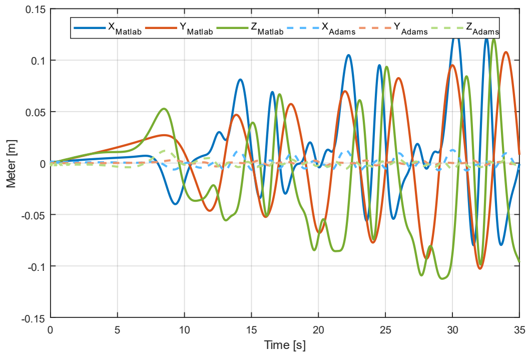


**Figure 104: Difference in position of $D$ ($\mathrm{r}_D^{0,2}$)**

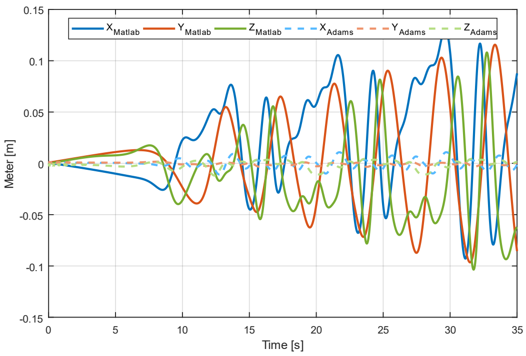


**Figure 105: Difference in position of $E$ ($\mathrm{r}_E^{0,2}$)**

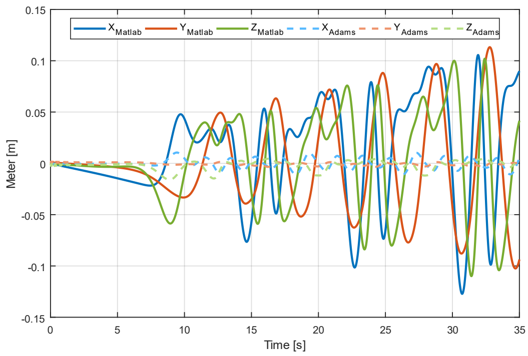


**Figure 106: Difference in position of $F$ ($\mathrm{r}_F^{0,2}$)**

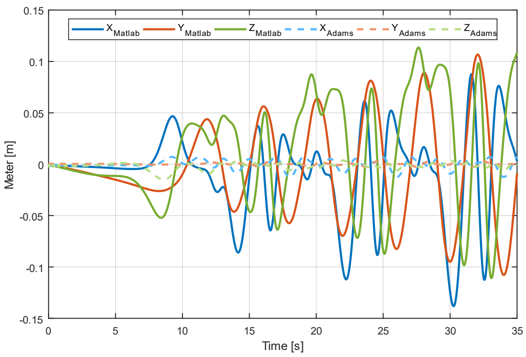


**Figure 107: Difference in position of $G$ ($\mathrm{r}_G^{0,2}$)**



UNIVERSITY OF TWENTE.

# APPENDIX I: RESULTS OF TRANSIENT DYNAMIC SIMULATIONS OF THE CLAPPER

In here, the results are given of transient dynamic simulations to calculate the stresses in the clapper. This is done using two methods, which are explained in the subchapters below.

## DYNAMIC SIMULATIONS OF THE CLAPPER USING ADAMS

In Adams, the stresses are calculated using stress modes which are derivatives of the deformation modes. This method is equal to calculating the deformations, multiplying the stress modes with the same flexible coordinates as for the deformations resulting in the stresses. Figure 108 and Figure 109 show the calculated equivalent stress in the clapper according to Von-Mises, based on the strain energy stored per unit volume. Only one timestep is shown, in which the maximum stress of ~187 MPa occurs. This maximum stress is a peak-stress in a small region around the tip of a reinforcement rib. Also important is the stress distribution through the clapper.

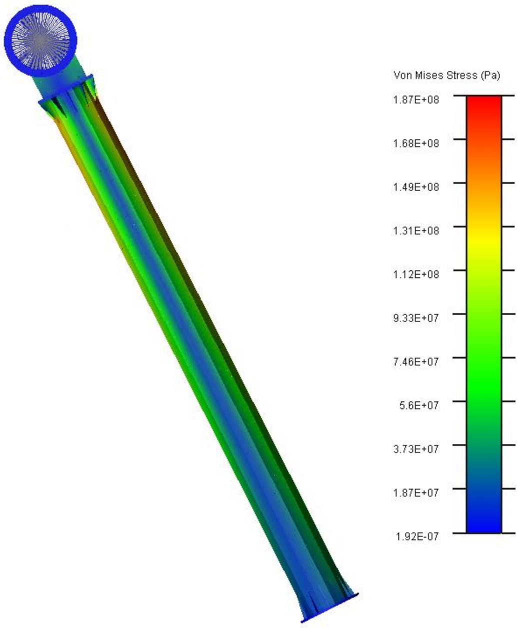


**Figure 108: Von-Mises stress distribution in the clapper from Adams**

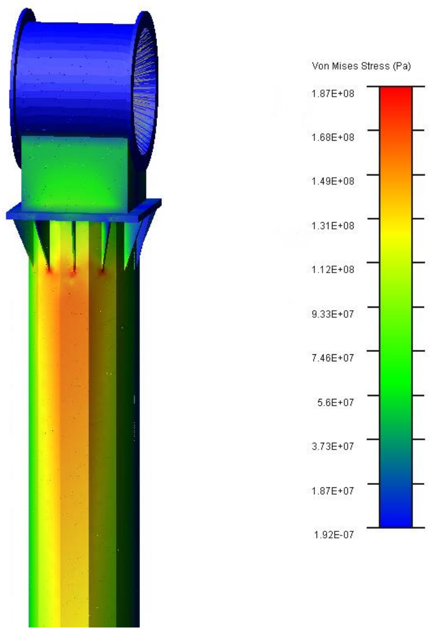


**Figure 109: Close-up of the Von-Mises stress distribution in the clapper from Adams**

The results are obtained using a dynamic simulation and a Newmark-beta integration scheme. A timestep size of 0.01 seconds is used over a period of 35 seconds. This timestep size showed to result in an accurate solution. The mesh is made using quadratic tetrahedral elements with 10 nodes per element and three degrees of freedom per element ($X$, $Y$ and $Z$ translation). A fine mesh is used to increase the accuracy, due to the higher number of degrees of freedom and the lower aspect-ratio for all elements (less distorted elements). The quadratic elements are used to avoid shear-locking, because the quadratic shape functions can describe bending (quadratic shape upon deformation). Using mesh refinement, it is ensured that the result is converged and therefore mesh independent.

## DYNAMIC SIMULATIONS OF THE CLAPPER USING ANSYS

In Ansys, the stresses can be calculated in a dynamic fashion by applying the calculated deformations to the two remote points (see Figure 26). The deformations can be calculated by multiplying the deformation modes with the flexible coordinates in time, resulting in 12 deformations. In Figure 110 this is shown, where the arrows represent the calculated translational deformations and rotational deformations.

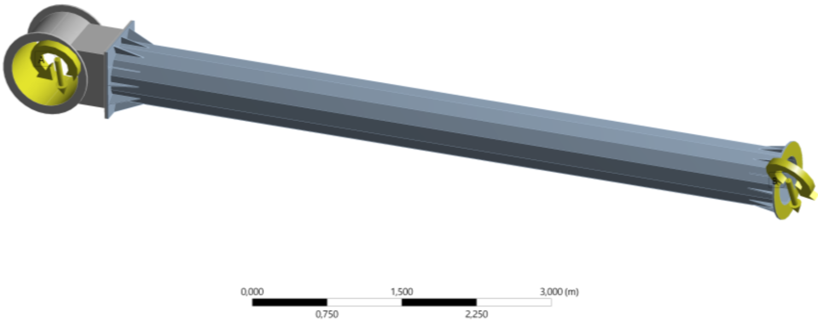


**Figure 110: Von-Mises stress distribution in the clapper from Ansys**

Although the transient analysis is linear, it costs significant more computational time than the quasi-static analysis due to the smaller timesteps taken. Results of the Von-Mises equivalent stresses are shown in Figure 111 and Figure 112. One timestep is shown, in which the maximum equivalent stress occurs which is approximately the same timestep as Adams. The maximum stress is calculated to be ~178 MPa which is equivalent to the simulation in Adams, the region where it occurs is also comparable. Also, the stress distribution through the clapper is equivalent with Adams.

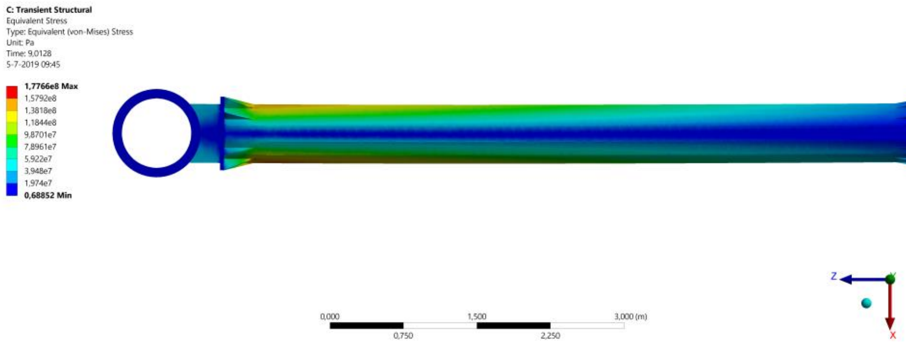


**Figure 111: Von-Mises stress distribution in the clapper from Ansys**

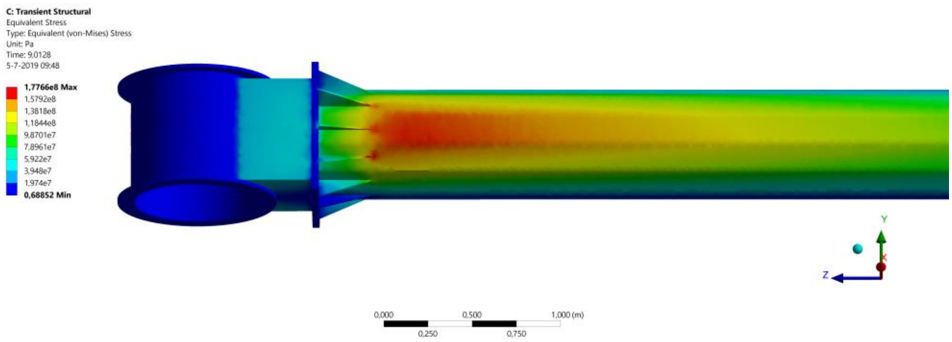


**Figure 112: Close-up of the Von-Mises stress distribution in the clapper from Ansys**

The results are obtained in a transient structural analysis, with an iterative solver named the preconditioned conjugate gradient solution. Variable time stepping is used with a minimum time step of 0.001 and a maximum time step of 0.1 seconds. The timestep size showed to result in an accurate solution. The mesh is made using quadratic tetrahedral elements with 10 nodes per element and three degrees of freedom per element ($X$, $Y$ and $Z$ translation). A fine mesh is used to increase the accuracy, due to the higher number of degrees of freedom and the lower aspect-ratio for all elements (less distorted elements). The quadratic elements are used to avoid shear-locking, because the quadratic shape functions can describe bending (quadratic shape upon deformation). Using mesh refinement, it is ensured that the result is converged and therefore mesh independent.

## COMPARISON OF ADAMS VSANALY ANSYS

A probe can be used to measure the Von-Mises stress in a node throughout the simulation. This is done in Ansys and Adams for the node at the tip of a reinforcement rib, where the maximum equivalent stress occurs. Figure 113 shows the result, where the maximum Von-Mises stress is clearly visible around 9 seconds which are almost equivalent. In the remaining of the simulation are multiple peaks visible, where a difference is between Ansys and Adams. Overall, the periodic motion through time looks similar.

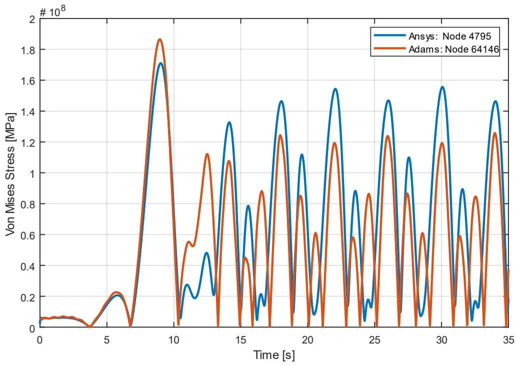


**Figure 113: Von-Mises stress in node through time**

This is the result of measuring the Von-Mises stress in one node throughout the simulation, which gives an indication of the equivalent stress progression through time. It can be improved by measuring the mean value of the Von-Mises stress of multiple nodes in a region, in this way peak stresses are avoided.

# APPENDIX J: MEASUREMENT PLAN

In here, a measurement plan is given which can be used to validate the results. It is composed of different measurements, with each a different purpose. Describing what is measured, why it is measured, how it is measured and where. Lastly it describes what the results will be and how these results can be interpreted.

## MEASURING THE ACCELERATIONS

The accelerations can be measured with an accelerometer in order to validate the calculated accelerations, velocities and positions of the system. From the acceleration measurements, the velocity and eventually the position can be estimated. Important is that the acceleration is measured with a high measurement frequency in order to make an adequate estimation. Noise-reduction techniques can be applied if necessary, for example using a weighted average of the last couple readings for calculating the velocity. Another way to reduce noise is by taking multiple sensor readings at each time step and use the median value of these.

Most beneficial would be to measure the accelerations at the position of the floating frame, since these coordinates are part of the generalized coordinates which are already solved. However, at this position no material is present to fix the accelerometer, so more practical would be to measure the accelerations at the given points in Figure 114. In here the acceleration of the clapper is measured at the bottom of the clapper, here it is possible to carry out multiple measurements at the same time. In the figure are acceleration measurements carried out on all gondolas, but only one gondola should be sufficient. Important to know is the orientation of the gondola in its initial position, this has influence on the eventual measurement results.

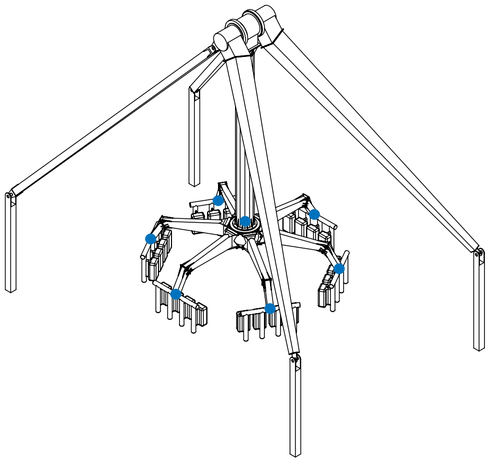

**Figure 114: Acceleration measurement positions**

The results of this measurement will be the accelerations over time in $X_i$, $Y_i$ and $Z_i$-direction in a local coordinate frame $i$. With the calculation model presented in this work, the acceleration, velocity and position of this measurement point can be calculated in the corresponding local coordinate frame.

## MEASURING THE STRAINS

Strain is measured using strain gauges at specific points on the Afterburner parts. There are different types of strain gauges for different applications. For the Afterburner, most interesting is to measure the bending strain, since the axial strain will be lower. Because the strain quantities are normally small (millistrain), it is most accurate to place the gauges in the areas where most deformations occur. Strain gauges are sensitive for errors by the mounting method and the surface to which they are mounted to.

The measured data consists of a resistance over time, where a gauge factor can be used to calculate the strain in each direction. To fully known the strain and afterwards the stresses in a point, it is required to measure the strain in three directions at least. With Hooke's law and the requirement that stresses on the free surface are equal to zero, all nine components of the stress tensor can be calculated. From the strains, the stresses can be calculated using the stress-strain relations, which then can be converted into an equivalent stress in one point. This quantity can be compared with calculated results in a finite element package or using Adams.