



Deutsches Zentrum
für Luft- und Raumfahrt



Universität Bremen*

Fachbereich 4 - Produktionstechnik

Masterarbeit

Identification of liquid sloshing dynamics by CFD analysis on board of a spin stabilized satellite

Zur Erlangung des akademischen Grades

Master of Science

-M. Sc.-

Vorgelegt von

Rick Burow (Matrikelnummer: 2985669)

Erstprüfer: Prof. Dr. rer. nat. C. Braxmaier
Deutsches Zentrum für Luft- und Raumfahrt Bremen
Universität Bremen

Zweitprüfer: Dr.-Ing. St. Theil
Deutsches Zentrum für Luft- und Raumfahrt Bremen

Einreichdatum: 15.06.2016 (SoSe 2016)

Erklärung der Urheberschaft

Hiermit erkläre ich, Rick Burow, geboren am 15. Juni 1989 in Angren,
dass ich meine Master Thesis mit dem Titel

**Identification of liquid sloshing dynamics by CFD analysis on board of a spin
stabilized satellite**

an der Universität Bremen in der Kooperation mit DLR Bremen ohne Hilfe Dritter und
ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus
fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich
gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keiner anderen
Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Ort, Datum

Unterschrift

Contents

1. Motivation	1
2. Problem Definition	4
3. Theoretical Background	6
3.1. Lateral Sloshing	6
3.2. Sloshing in a rotating container	9
3.3. Equivalent Mechanical Model	10
3.4. System identification of liquid sloshing dynamics on board of a spin-stabilized satellite	13
3.5. Theory of system identification	15
3.6. Computational Fluid Dynamics	19
3.6.1. Solution Errors and Uncertainty	20
4. CFD-Simulation implementations	22
4.1. Solver and Physics Selection: Volume of Fluid (VoF) Method	22
4.2. Case Definitions	23
4.3. Pre-Processing and run controls	27
4.3.1. Geometry and Mesh generation	27
4.3.2. Boundary Conditions	31
4.3.3. Solver Settings and Run Controls	32
4.4. Results discussion	34
4.4.1. Data Processing	34
4.4.2. Grid Convergence Study	35
4.4.3. Simulation Results	40
4.4.4. Transfer Function Determinations	51
5. Conclusion	59
6. Outlook	61

References	62
A. ControlDict	64
B. fvSchemes	67
C. fvSolution	68

List of Figures

1.1. Eu:CROPIS spacecraft CAD-model	2
1.2. Eu:CROPIS spacecraft CAD-model, inside view	3
2.1. Water tank including baffles	4
3.1. First lateral sloshing modes	7
3.2. Hydrodynamic regimes	8
3.3. Equivalent mechanical models	11
3.4. Complex equivalent mechanical models	12
3.5. Block diagram of liquid sloshing system identification	15
3.6. Block diagram to generate an output error.	16
3.7. Modified block diagram	17
3.8. Computational Fluid dynamics workflow	20
4.1. Schematic representation of VoF-Method	23
4.2. Watertank with mounted baffles, axis definition	24
4.3. Initial gas-water phase, circumferential direction	25
4.4. Initial gas-water phase, radial direction	26
4.5. Mesh withouth baffles	28
4.6. Simplified tank geometry with implemented baffles	30
4.7. Mesh with implemented baffles	31
4.8. Results of the grid convergence study	36
4.9. Grid convergence study results	38
4.10. Richardson's extrapolation	39
4.11. Sloshing in circumferential direction of the tank without baffles	41
4.12. Sloshing in longitudinal direction of the tank without baffles	42
4.13. Sloshing in radial direction of the tank without baffles	43
4.14. Sloshing in circumferential direction showing the effect of implemented baffles.	44
4.15. Sloshing in radial direction, which shows the comparison between the simulation case with and without implemented baffles	45

4.16. Sloshing in a longitudinal direction showing the comparison between the simulation case with and without implemented baffles	46
4.17. Sloshing in circumferential direction of the tank with implemented baffles .	47
4.18. Sloshing in a longitudinal direction of the tank with implemented baffles .	48
4.19. Sloshing in a radial direction of the tank with implemented baffles	49
4.20. Additional CoM motion in a longitudinal direction with step response in radial direction	50
4.21. Step response of the estimated transfer fuction compared to OpenFOAM data. $\omega_1 = 0.5294 \frac{\text{rad}}{\text{s}}$	52
4.22. Step response of the estimated transfer fuction compared to OpenFOAM data. $\omega_1 = 0.5294 \frac{\text{rad}}{\text{s}}$	53
4.23. Step response of the estimated transfer fuction compared to OpenFOAM data. ω_2	54
4.24. Step response of the estimated transfer fuction compared to OpenFOAM data. ω_3	55
4.25. Step response in radial direction of the estimated transfer fuction compared to OpenFOAM data	56
4.26. Step response along y-axis of the estimated transfer fuction compared to OpenFOAM data	58

1. Motivation

Eu:CROPIS (Euglena and Combined Regenerative Organic-Food Production in Space)[1] is a spacecraft mission of the German Aerospace Center (DLR), which focuses on researching the behavior of two greenhouses at different gravity levels. To achieve this a cylindrical satellite is rotating around its longitudinal axis at different angular speeds. The goal of Eu:CROPIS is to grow tomatoes in Moon and Mars gravity levels. Water is of course needed to grow tomatoes. For this reason, two sector-annular cylindrical tanks of 3 l volume filled with approximately 1.5 l water are on-board of the satellite.[2] Due to the rotation of the satellite, the water motion can affect the stability of the Attitude and Orbit Control System (AOCS). Consequently, the study of sloshing and its effects on the spacecraft AOCS is of paramount importance.

The periodic motion of liquid (e.g. propellant) with free surface in reservoirs is defined as sloshing. The liquid sloshing interacts with the tank walls and affects the overall system dynamics. The satellite dynamics and the forces distribution causes the liquid sloshing, which in turn influences the whole system dynamics of the satellite. For this reason, it is very important to predict the sloshing behavior and thus be able to guarantee the dynamic stability of the system.[3]

Sloshing can be identified by different parameters such as hydrodynamic pressure distributions, forces and moments or natural frequencies. These parameters affect the rigid body motion of the satellite directly, since they describe the sloshing behavior, which ultimately affects the spacecraft AOCS.[3] Usually, Equivalent Mechanical Models (EMM's) are used to simplify the sloshing phenomena analysis. EMM is a system made of basic mechanic elements such as springs, pendulums etc., that simulate the sloshing behavior and make them easy to model and understand. The EMM is then incorporated in the spacecraft motion and the AOCS model, which allows us to check and modify the AOCS. [4]

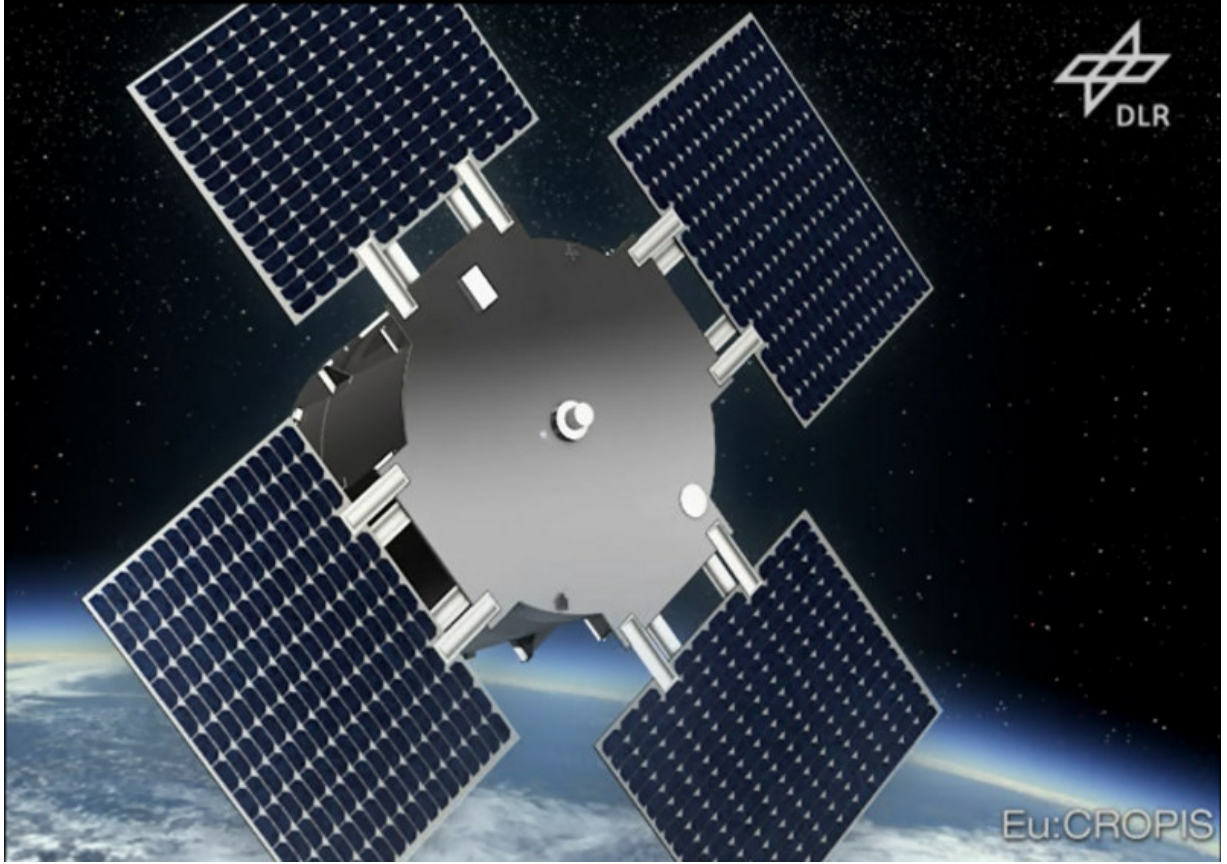


Figure 1.1: Eu:CROPIS spacecraft CAD-model as cylindrical body with four deployable solar arrays.[5]

In this thesis, the water sloshing inside the tank of the spin stabilized satellite (Eu:CROPIS) has been identified. First, a Computational Fluid Dynamics-Analysis was used to simulate a liquid motion under realistic boundary conditions. Here the water Center of Mass (CoM) motion was computed to determine the liquid sloshing transfer functions. The method of system identification was used to determine the transfer function parameters. Overall four cases were considered: a tank with and without baffles at the lowest gravity level and a tank with baffles at two other gravity levels. Each case includes three simulations in relevant axis direction: circumferential, radial and longitudinal. So altogether twelve simulations are realized. Implemented baffles in the water tank are used to damp the liquid motion.

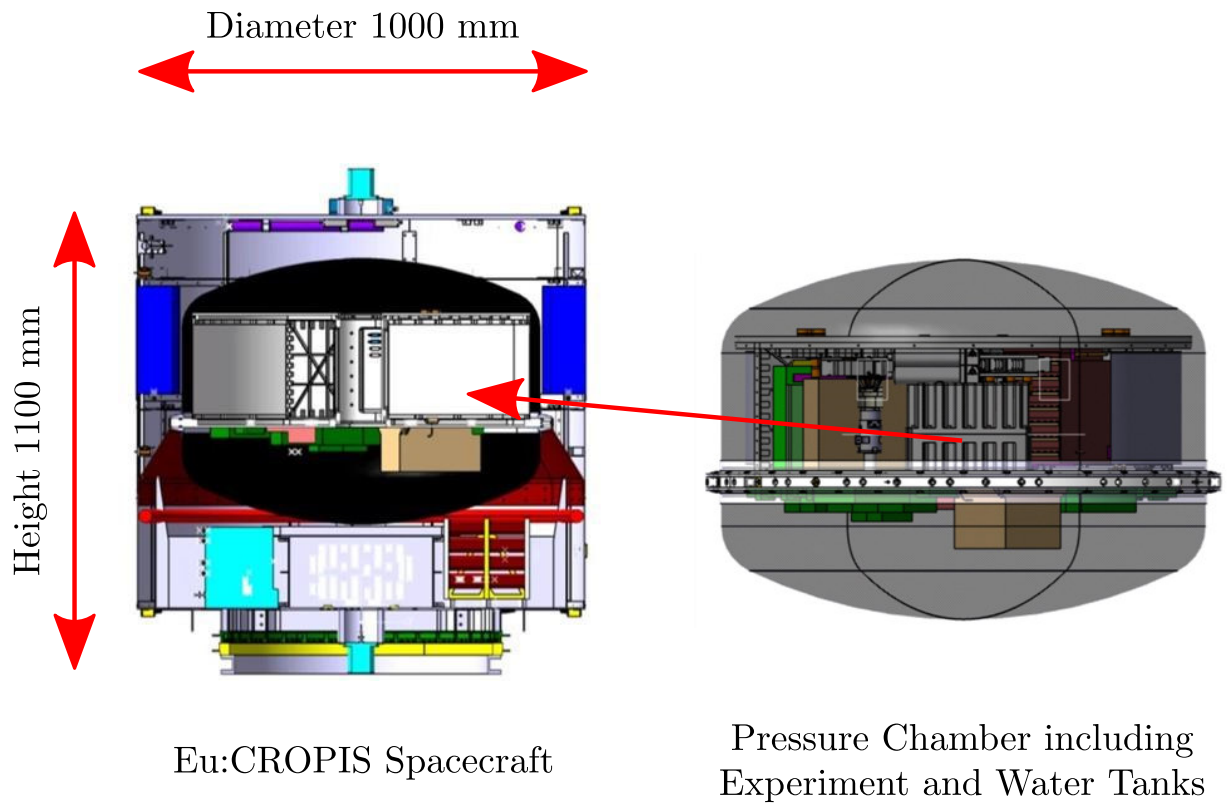


Figure 1.2: Eu:CROPIS spacecraft CAD-model, inside view with pressure chamber including a main experiment with water tanks. The Spacecraft is a cylinder with a diameter of 1000 mm and a height of 1100 mm.[5]

2. Problem Definition

The liquid slosh motion interacts with the satellite dynamics, i.e. the orbit perturbing forces and AOCS induces forces affect liquid sloshing and the other way around. To ensure the stability of the system, the sloshing dynamics must be considered, so the AOCS can be designed and optimized. Figure 2.1 shows the design of the water tanks with the implemented baffles inside:

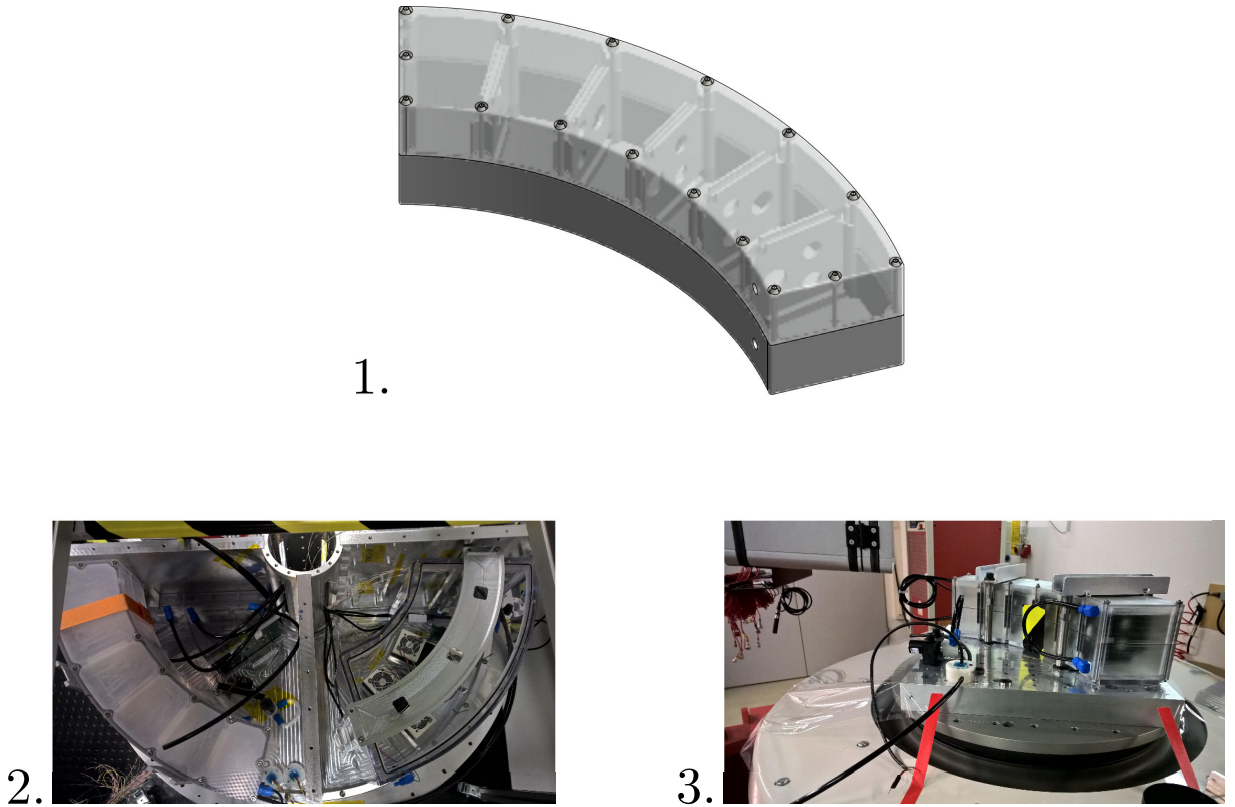


Figure 2.1: Water tank including baffles: 1. CAD-model of the water tank with baffles. 2. Equivalent Engineering Model (EEM) of the payload, on the left the water tank and on the right the chamber for tomato plant, mounted on the holding structure. 3. Water tank prototype on the shaker table.[5],[6]

The satellite includes two symmetrical mounted tanks 150 mm above the satellite Center of Mass (CoM). It is not possible to achieve micro-g on earth for the needed time span of ≈ 100 s (drop tower ≈ 9 s, parabolic flight ≈ 30 s), therefore numerical simulations are used to predict liquid motion in such gravity conditions. Numerical simulations are

2. Problem Definition

performed by means of Computational Fluid Dynamics (CFD)-Analysis with *OpenFOAM* software. *Open Source Field Operation and Manipulation (OpenFOAM)* is a free software including numerical solvers, pre- and post- processing tools to analyze mechanic, thermal or combined problems, but also Computational Fluid Dynamics. *OpenFOAM* has a lot of solvers to compute special numerical problems, like particle-tracking, combustion, multiphase flow, free-surface tracking etc.[7]

In order to reduce computational complexity and time, the simulation includes only one tank with simplified geometry (with and without baffles). However, all relevant geometrical tank parts are realized, like connections between the baffles, which influence the physical behavior of the water. Consequently, the results are still valid, as the tanks are equal and have a symmetrical configuration along spacecraft longitudinal axis. Different angular velocities are simulated to generate various equivalent gravity levels, which is calculated as follows [8]:

$$g_{equivalent} = (\omega^2 R)g_{earth}, \quad (2.1)$$

ω is the angular velocity, R is the radius, where the gravity appears and $g_{earth} = 9.81 \frac{m}{s^2}$ is the earth gravitational acceleration. For the Moon gravity of $0.16 g_{earth}$ at radius of $R = 340 \text{ mm}$ the angular velocity has to be $2.1177 \frac{rad}{s}$. For Mars gravity of $0.38 g_{earth}$ the spin rate is $3.2636 \frac{rad}{s}$. Another spin rate of $0.5294 \frac{rad}{s}$ with the equivalent gravity level of $0.01 g_{earth}$ is also simulated, which is the initial spin rate of the satellite after orbit injection and stabilization. In addition, various motions of the tank have to be implemented in the simulations to identify a sloshing system behavior.

Simulation results have to be analyzed in order to specifically identify the sloshing process. The liquid sloshing transfer functions are obtained by application of system identification algorithm, which calculates transfer function parameters with the objective to minimize the error between liquid sloshing systems and the simulated model. Once the transfer functions are known, they can be implemented into the nutation simulation of the AOCS at different spin velocities.

3. Theoretical Background

In this section, theoretical background about sloshing, EMMs, system identification techniques and CFD are presented. This information is needed to successfully tackle the problem described in Section 2. First, the sloshing background is divided in lateral sloshing and sloshing in a rotating container. The previous occurs in case of lateral oscillation of a container partially filled with liquid and the latter in case of the rotational movement (spinning). The EMM section describes the most common models used: the pendulum-model and the spring-mass model. The system identification section gives an approach of the algorithm used to determine the transfer functions. The CFD section describes the basic work-flow in numerical simulations.

3.1. Lateral Sloshing

Sloshing is defined as the motion of a liquid free surface inside a partially filled container in motion due to any disturbance. There are many types of sloshing modes, e.g. planar, non-planar, rotational, symmetric or asymmetric. The general problem of liquid sloshing involves the estimation of hydrodynamic pressure distribution, forces, moments and natural frequencies of the liquid free surface. The main dynamical effect of sloshing is oscillation of the liquid Center of Mass (CoM), which influences the rigid body motion of the satellite.[3]

Different modes (configurations) of free surface sloshing can be defined by their wave number m (number of waves in a circumferential direction, along the boundary line of a circle) and by their mode number n (number of waves in a radial direction). Two general modes of sloshing are the symmetric and the asymmetric. Figure 3.1 shows the first modes of lateral liquid sloshing for asymmetric (a.) and symmetric modes (b.).[9]

For the first asymmetric mode the CoM shift in lateral direction is the biggest. For this reason usually only the first mode is considered. The symmetric modes are less relevant in case of lateral sloshing, since they do not cause CoM shift in lateral direction, but in longitudinal.[4] Due to the micro-g environment, there is no regular gravitational potential anymore, which stabilizes the liquid volume to the bottom. In this case, the prediction

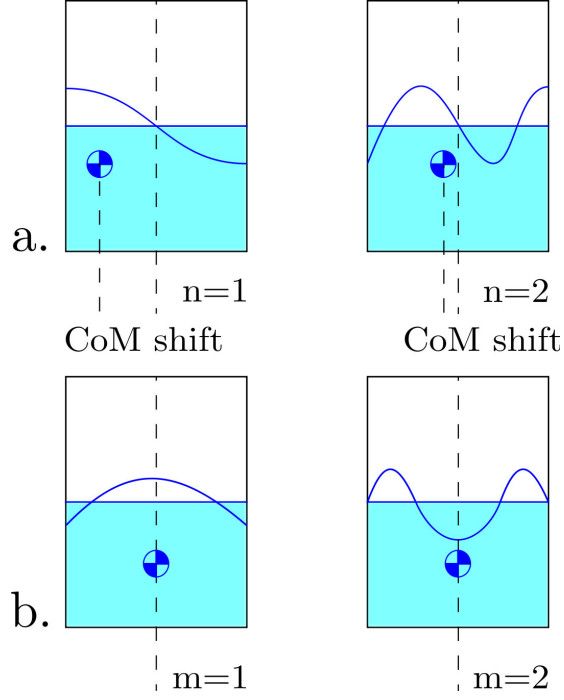


Figure 3.1: First sloshing modes: a. Asymmetrical modes with CoM shift in lateral direction. b. Symmetrical modes without CoM shift in lateral direction, however there is a CoM shift in longitudinal direction. [4]

about liquid reorientation becomes a problem. In general, the motion of sloshing systems is influenced by capillary action, body inertia and viscous forces. The analysis of liquid sloshing can be simplified if some forces are much smaller compared to others, so they can be ignored. Dimensionless parameters are used to find out the dominant effects of the system. These numbers separate the hydrodynamic behavior into three regimes (Figure 3.2) [4]:

- inertia dominated regime. In this case capillary forces are small compared to inertia forces, so they can be neglected.
- gravity dominated regime. Here gravity forces are large, so inertia and capillary forces can be neglected.
- capillary dominated regime. Here gravitational and inertia forces are small compared to capillary forces. This is only the case in micro-g environments with a very small disturbances forces.

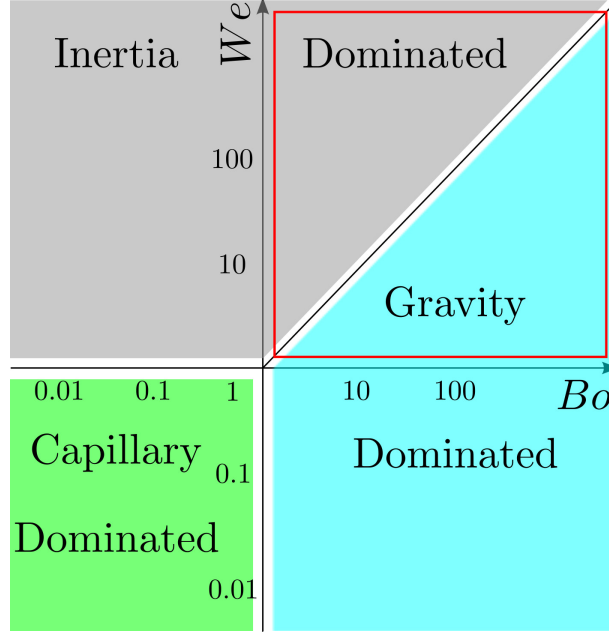


Figure 3.2: Hydrodynamic regimes divided into inertia, gravity and capillary, separated by the dimensionless parameters Bond and Weber numbers. Red box marks the dominated regime relevant for this thesis, so the capillary forces can be neglected in the numerical simulations.[4]

The ratio of gravitational to surface tension forces is defined by the Bond number [4]:

$$Bo = \frac{\rho g L^2}{\sigma}, \quad (3.1)$$

where ρ is the density of the fluid, g is the acceleration due to the gravity, L is the characteristic length (for example diameter of the capillary tube) and σ is the surface tension.

The ratio of inertia to capillary forces is defined by the Weber number as follows[4]:

$$We = \frac{\rho V^2 L}{\sigma}, \quad (3.2)$$

where v is the characteristic velocity of the fluid, L is the characteristic length of the system, ρ is the density of the fluid and σ is the surface tension. The Froude number defines the importance of inertia and body forces and is defined as the ratio between the Weber and the Bond number [4]:

$$Fr = \frac{V^2}{gL} = \frac{We}{Bo}. \quad (3.3)$$

In each case the importance of viscous effects must be determined separately from the Reynolds number, which is defined as [3]:

$$Re = \frac{v\rho L}{\mu}, \quad (3.4)$$

where ρ is the density, L is the characteristic length, v is the relative velocity of the object with respect to the fluid and μ is the dynamic viscosity of the fluid.

3.2. Sloshing in a rotating container

In the previous section we have seen that lateral sloshing shifts the CoM position of the liquid. This section gives an overview on the specific case of liquid in rotating containers, since this one applies to the mission Eu:CROPIS.

The aspect of sloshing in a rotating container is important due to the stabilization of the satellite by spinning around the major moment of inertia axis. Due to the fact that the transverse angular velocities are not exactly zero, the desired spacecraft spin axis and the angular momentum vector do not coincide. For this reason the spacecraft starts to nutate around its spin axis. Nutation effects the liquid motion relative to the tank and can lead to spacecraft instability, when it interacts with the AOCS. In general, the liquid motion tends to damp nutation, which is a positive effect for the AOCS, because there is no active system needed to damp the nutation angle.[4] Nevertheless it is essential to analyze liquid sloshing and its damping effect on the spacecraft. Two types of natural oscillation are possible in rotating containers [10]:

- free surface oscillations (sloshing waves),
- internal liquid oscillations (inertial waves).

Sloshing waves can only occur if free surfaces exist and are defined as the dynamic interaction between pressure forces and effective gravity (generally a combination between true gravity and centripetal forces, in the case of micro-g environment, mostly centripetal forces). Viscosity provides some damping of the motion, but does not change the slosh natural frequencies much. Resonant frequencies are more than twice the spin rate. [10]

3. Theoretical Background

Inertial waves can also occur in a fully filled tank, however the CoM shift is negligible in this case. Internal liquid oscillations are defined as dynamic interactions between pressure and Coriolis forces. Resonant frequencies are less than twice the spin rate. [10]

In the case of rotating tanks the characteristic (dimensionless) numbers can be written as a function of the spin rate. The centrifugal (rotational) Bond number is then defined as relation to the centrifugal acceleration [4]:

$$Bo_C = \frac{\rho \cdot R \cdot \omega^2 \cdot L^2}{\sigma}, \quad (3.5)$$

where ω is the spin rate, ρ is the density of the fluid, R is the radius (where equivalent gravity is calculated), L is the characteristic length and σ is surface tension.

The most common conditions for space environments are for $Bo_C \gg 1$, due to the large spin rates and micro-g environment (small disturbance forces). In this case the centrifugal acceleration (induced gravity) is the dominant force. For small spin rates surface tension becomes more dominant and plays an important role in shaping the free surface. Surface tension forces also influences liquid motion damping and has to be taken into account.[4]

In case of Eu:CROPIS payload the tanks are not fully filled with fluid, so sloshing waves occur and produce the CoM shift. The centrifugal force dominates the liquid sloshing behavior ($Bo_C \gg 1$), which allow to neglect capillary forces.

3.3. Equivalent Mechanical Model

The main dynamic effect of sloshing is the oscillation of the liquid CoM relative to the tank. Such oscillations can have a significant influence on the stability of a satellite. EMMs are typically used to ease the implementation of fluid dynamics into AOCS simulations, because their equations are relatively easy to implement. EMMs reproduce the forces and moments acting on tank walls due to the sloshing, by using simple mechanical components.[3]

In the case of a filled container, the fluid dynamics can be represented by an equivalent rigid body. However, when a partially filled container is considered, the sloshing has more than one natural frequency, so the model must contain free oscillating masses. The

3. Theoretical Background

linear motion of liquid CoM can be represented by another linear dynamic model. The parameters of the EMMs only depend on the tank shape and the liquid properties.[3]

Figure 3.3 shows two EMMs commonly used to represent linear sloshing: the pendulum and spring-mass models. Both models induce the same forces and torques (equivalent to the sloshing). The pendulum model has the advantage that its natural frequency depends on the acceleration (or gravity g), as the liquid sloshing frequency, so there is a direct dependence between natural frequency, pendulum length and the gravity. Compared to pendulum model, the spring constant K has to be changed, if there are changes in acceleration g . [4]

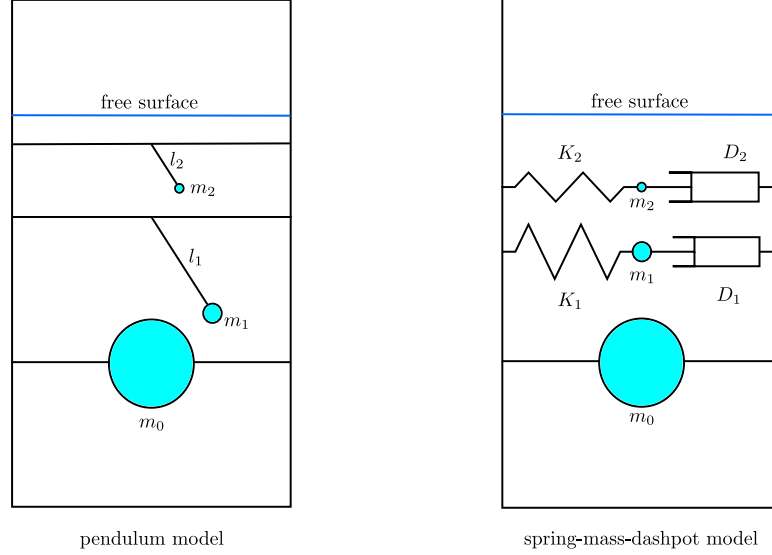


Figure 3.3: Equivalent mechanical models, which represent linear sloshing. EMM parameters are: pendulum length l_n , sloshing mass m_n , spring constant K_n and damping constant D_n . Different numbers of systems are used to realize different sloshing natural frequencies.[3]

3. Theoretical Background

The natural frequencies for both models are defined as [9]:

$$\omega_p = \sqrt{\frac{g}{l_n}}, \quad (3.6)$$

$$\omega_{sm} = \sqrt{\frac{K_n}{m_n}}, \quad (3.7)$$

where g is the acceleration or equivalent gravity field, l_n is the pendulum length, K_n is the spring constant and m_n is the mass. The index n represents the sloshing mode. Each model can be extended by a viscous dash-pot to include the liquid damping. More than one model can be added to represent different slosh modes or to characterize complex geometries.

The first task, when using EMM, is to choose a compatible model and derive its equations of motion. The model has to result into the same position shift, as the CoM shift caused by the liquid sloshing. It is important to know the dominant motion of the tank.

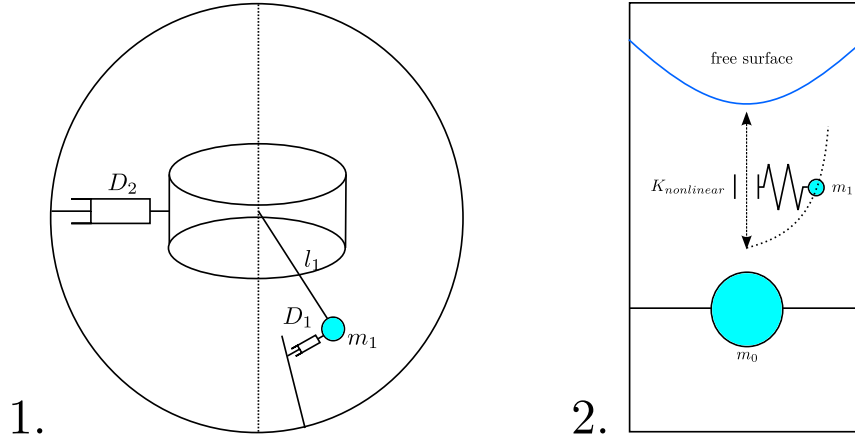


Figure 3.4: Complex EMMs. 1. EMM for a spinning spherical tank including rotor with damping and attached damped pendulum.[10] 2. EMM for cylindrical tank for slow spinning rates including a slider with non linear spring constant depended on the spin rate to represent non-linear sloshing. [3]

Once the model is chosen, the parameters, such as the spring constant or the pendulum length have to be determined analytically from liquid properties and depended on the tank geometry, or by an experiment. In general, more complex non-linear mechanical models, like sliding mass along longitudinal axis and additional non-linear springs (see Fig. 3.4) can be used.[10]

It is possible to add other mechanical models to characterize liquid sloshing for other excitations, like for example spinning. An example is shown in Figure 3.4, which consists of a rotor with an pendulum. Analytical derivation of model parameters requires a theoretical analysis of the sloshing. However, in many cases this is not possible or practical. In such cases the model parameters are determined by experimental measurements. The experimental procedure consists of exciting the tank (typically a scaled model) with a simple harmonic motion (horizontal) and measuring the resulting forces and torques caused by liquid sloshing on the walls. Finally, the model equations of motion are fitted to the torque and force measurements. The disadvantage of this approach is that the estimated model has to be scaled to the micro-g environment.[4]

3.4. System identification of liquid sloshing dynamics on board of a spin-stabilized satellite

In the previous sections an introduction about lateral and rotating sloshing and EMM has been given. This section describes the approach used to identify and model liquid sloshing in case of Eu:CROPIS spacecraft.

The main task is to identify the interaction between the spacecraft nutation and liquid dynamics. As mentioned before, the dominant dynamic effects of liquid sloshing are the CoM shift and the liquid forces acting on the spacecraft. The approach selected in this thesis consists in predicting the CoM shift by CFD-analysis under micro-g conditions in each relevant direction, i.e., circumference, radial and axial. It is important to stimulate all the frequencies of the system in each direction. To do so, a step function in acceleration can be used as the system excitation. By knowing the system excitation and its response the transfer function can be determined. Finally, the estimated transfer functions can be

3. Theoretical Background

implemented into the spacecraft nutation simulation to predict the influence of the liquid dynamics on the satellite dynamics. It is expected, that the liquid sloshing motion will damp the nutation angle. Therefore the system of liquid will act as a passive nutation damper.

The transfer function identification approach [11] has the advantage that no knowledge of the mechanical model is needed. All of the parameters, like spring constant, equivalent mass, etc. are integrated in the numerator and the denominator of the transfer function. Consequently, it is not necessary to derive the mechanical systems equation of motions. Before doing the simulations, it is important to identify the liquid motion regime by means of the Bond number, where the equivalent gravity g caused by centrifugal forces is used. The centrifugal Bond number is calculated as in Equation 3.5. The equivalent gravity acceleration is calculated as follows [8]:

$$g = (\omega^2 \cdot R), \quad (3.8)$$

where ω is the spin rate, R is the g-reference radius and $g_{earth} = 9.81 \frac{m}{s^2}$.

Table 3.1 summarizes the cases needed to identify liquid sloshing. For each case there are three numerical simulations in three relevant sloshing directions required. The simulation case without baffles was realized only for the smallest spin rate to show the effect of using them. The final configuration of the tank is with integrated baffles inside.

Case No.	Spin rate $[\frac{rad}{s}]$	Eq. g-level	Baffles	Bo
1	0.5294	0.01	no	5.93
2	0.5294	0.01	yes	5.93
3	2.1177	0.16	yes	94.9
4	3.2636	0.38	yes	225.4

Table 3.1: Cases needed to identify liquid sloshing with their important parameters as spin rate, g-level and Bo number.

As shown in Table 3.1 all of the cases have the $Bo \gg 1$ number, so the surface tension forces can be neglected. In this case the centrifugal acceleration is the dominant force, which presses the liquid against the outer tank wall. The negligence of the surface tension forces allows simplified simulations which reduces computational time.

3.5. Theory of system identification

Figure 3.5 shows liquid sloshing as a dynamic system with an input $u(t)$ and an output $y(t)$. This relationship between the input and output signals represent the system. This relationship is known as the transfer function (in the frequency domain) and the impulse response (in the time domain) of the system, its estimation is done by different system identification techniques.[12]

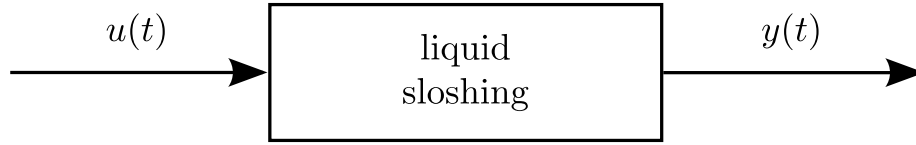


Figure 3.5: Block diagram of the liquid sloshing LTI-system. The system can be characterized if proper input and output signals are available.[12]

A key aspect in system identification problems [11] is the generation and measurement of the signals $u(t)$ and $y(t)$. It is important to stimulate all the modes of the unknown system. A suitable input signal is the step signal, as it contains all the frequencies. Here, the liquid sloshing is the system that needs to be identified. A step function (in acceleration) is applied to the liquid in each relevant direction separately in the simulation by realization of different tank motion. This signal is the input signal $u(t)$, or

$$u(t) = \begin{cases} 0 & \text{if } t < 0 \\ A & \text{if } t \geq 0 \end{cases} \quad (3.9)$$

where A is the acceleration amplitude. The output signal, $y(t)$, is defined as the motion of liquid's CoM (angle or position), which is calculated by the CFD program OpenFOAM.

The liquid sloshing is assumed to be a linear and time-invariant system (LTI), because the amount of water used by the plants is very small, so the following rational and time-

3. Theoretical Background

discrete transfer function is used as a model:

$$\hat{G}(z^{-1}) = \frac{\hat{b}_0 + \hat{b}_1 z^{-1} + \dots + \hat{b}_m z^{-m}}{1 + \hat{a}_1 z^{-1} + \dots + \hat{a}_n z^{-n}} = \frac{\hat{B}(z^{-1})}{\hat{A}(z^{-1})}. \quad (3.10)$$

The objective is to determine the coefficients \hat{a}_i and \hat{b}_j since these parameters characterize the LTI system frequency response. One of the methods is to calculate the output error $e_A(z)$ shown in Figure 3.6. In this case, the coefficients determination is a non linear optimization problem, in which no closed solution exists. In order to obtain a linear

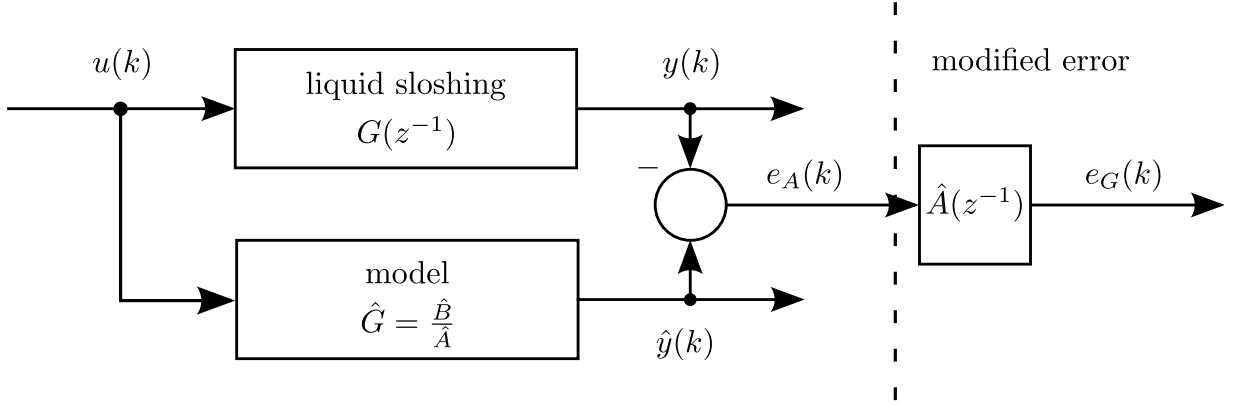


Figure 3.6: Block diagram to generate an output error from the actual model and the estimated model outputs. In order to deal with a linear optimization problem the output error $e_A(k)$ is multiplied by the denominator of the estimated transfer function $\hat{A}(z^{-1})$. [12]

optimization problem, the output error, $e_A(k)$, is multiplied by $\hat{A}(z^{-1})$, i.e.

$$\begin{aligned} e_G(z) = \hat{A}(z^{-1}) \cdot e_a(z) &= \hat{A}(z^{-1})[y(z) - \hat{y}(z)] \\ &= \hat{A}(z^{-1})[y(z) - u(z)\hat{G}(z^{-1})] \\ &= \hat{A}(z^{-1}) \left[y(z) - u(z) \frac{\hat{B}(z^{-1})}{\hat{A}(z^{-1})} \right] \\ &= y(z)\hat{A}(z^{-1}) - u(z)\hat{B}(z^{-1}). \end{aligned} \quad (3.11)$$

The modified error, $e_G(z)$, is now a linear combination of the output $y(z)$ weighted by $\hat{A}(z^{-1})$ and the input $u(z)$ weighted by $\hat{B}(z^{-1})$. According to this, the block diagram shown in Figure 3.7 is obtained. For $e_G(z) = 0$ the system G and the estimated system \hat{G} , are exactly the same. The target is to minimize modified error $e_G(z)$ by the Least squares

3. Theoretical Background

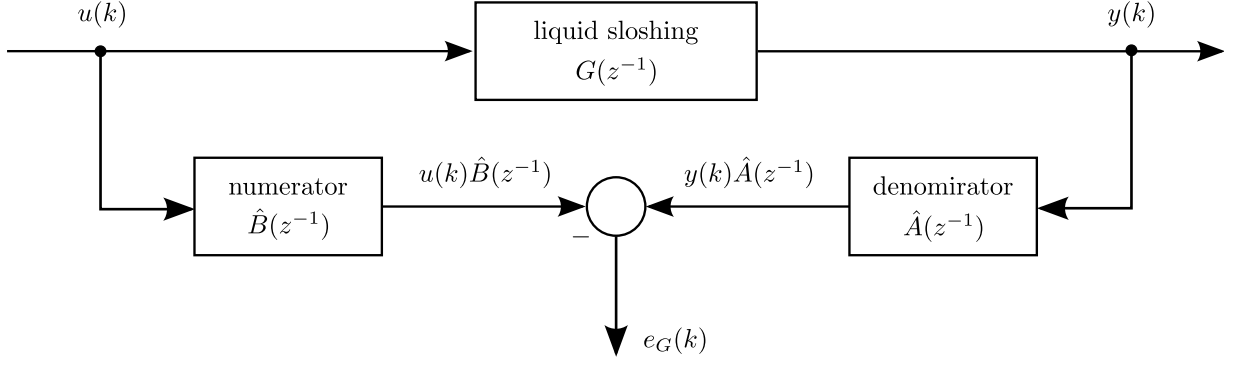


Figure 3.7: Modified block diagram, where error $e_G(K)$ is a linear combination of input and output signals.[12]

method. The determination of the numerator $\hat{B}(z^{-1})$ and the denominator $\hat{A}(z^{-1})$ lead to a system of linear equations:

$$y(z)\hat{A}(z^{-1}) - u(z)\hat{B}(z^{-1}) = e_G \quad (3.12)$$

or

$$\begin{aligned} & y(z)[1 + \hat{a}_1 z^{-1} + \hat{a}_2 z^{-2} + \dots + \hat{a}_n z^{-n}] \\ & - u(z)[\hat{b}_0 + \hat{b}_1 z^{-1} + \hat{b}_2 z^{-2} + \dots + \hat{b}_m z^{-m}] = e_G. \end{aligned} \quad (3.13)$$

This equation now can be back-transformed to the discrete-time domain, k , as follows:

$$\begin{aligned} & [y(k) + \hat{a}_1 y(k-1) + \hat{a}_2 y(k-2) + \dots + \hat{a}_n y(k-n)] - \\ & - [\hat{b}_0 u(k) + \hat{b}_1 u(k-1) + \hat{b}_2 u(k-2) + \dots + \hat{b}_m u(k-m)] = e_G, \end{aligned} \quad (3.14)$$

which can be re-written as:

$$\begin{aligned} & -[\hat{a}_1 y(k-1) + \hat{a}_2 y(k-2) + \dots + \hat{a}_n y(k-n)] + \\ & + [\hat{b}_0 u(k) + \hat{b}_1 u(k-1) + \hat{b}_2 u(k-2) + \dots + \hat{b}_m u(k-m)] = y(k). \end{aligned}$$

There are at least $N = m + n + 1$ equations needed to solve the system of equations for the unknown parameters $\hat{a}_1, \dots, \hat{a}_n, \hat{b}_0, \dots, \hat{b}_m$. The system of equations can be expressed in the following matrix form:

$$\mathbf{M} \cdot \hat{\mathbf{p}} - \mathbf{y} = \mathbf{e}_G \quad (3.15)$$

3. Theoretical Background

where

$$\mathbf{M} = \begin{bmatrix} u_k & u_{k-1} & \cdots & u_{k-m} & -y_{k-1} & -y_{k-2} & \cdots & -y_{k-m} \\ u_{k+1} & u_k & \cdots & u_{k+1-m} & -y_k & -y_{k-1} & \cdots & -y_{k+1-m} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ u_{k+m} & u_{k+m-1} & \cdots & u_k & -y_{k+m-1} & -y_{k+m-2} & \cdots & -y_k \\ u_{k+m+1} & u_{k+m} & \cdots & u_{k+1} & -y_{k+m} & -y_{k+m-1} & \cdots & -y_{k+1} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ u_{k+m+n} & u_{k+m+n-1} & \cdots & u_{k+n} & -y_{k+m+n-1} & -y_{k+m+n-2} & \cdots & -y_{k+n} \end{bmatrix} \quad (3.16)$$

$$\hat{\mathbf{p}} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_m \\ a_1 \\ \vdots \\ a_n \end{bmatrix} \quad (3.17)$$

and

$$\mathbf{y} = \begin{bmatrix} y_k \\ y_{k+1} \\ \vdots \\ y_{k+m} \\ y_{k+m+1} \\ \vdots \\ y_{k+m+n} \end{bmatrix} \quad (3.18)$$

For any matrix \mathbf{M} with at least a rank of N , the parameter vector, $\hat{\mathbf{p}}$, can be calculated by the Least squares method:

$$\hat{\mathbf{p}} = (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \cdot \mathbf{y}. \quad (3.19)$$

This algorithm is implemented in a *MATLAB System Identification Toolbox* and has been used to determine liquid sloshing transfer functions.

3.6. Computational Fluid Dynamics

Computational Fluid Dynamics (CFD) uses applied mathematics, physics and computational software to solve problems, which involve fluid flow. The CFD strategy is to replace the continuous problem domain with a discrete domain using a grid. The process of CFD can be divided in three main tasks as shown in Figure 3.8 [13]:

- pre-processor,
- solver,
- post-processor.

The pre-processor includes the geometry and mesh generation, including material properties and boundary conditions definition. The pre-processor is the input for the solver task. The process of mesh generation is very important for the solver task: no accurate and reasonable solutions without a good mesh are possible. It is a difficult process, because it is important to find balance between accuracy, which requires very fine mesh, and computational time, which is shorter for a coarse mesh. The problem definition affects the decision between accuracy and computational time.

The solver contains the solving of the governing equations for each mesh point and time step. Here the transport equations are solved (mass, momentum, energy or other transport variables) with the help of physical models (turbulence). It is important to control and monitor the solution process. In case of solver instabilities (non-physical behavior of transport quantities), which lead to aborting the simulation, the mesh and boundary conditions have to be checked and improved.

The post-processor includes result reporting and visualization. Here graphs, contour plots or calculated required physical parameters like drag-coefficient and others are shown. Post-processing is very important to analyze the CoM data, the free surface motion visualization gives indication about the validity of the results.

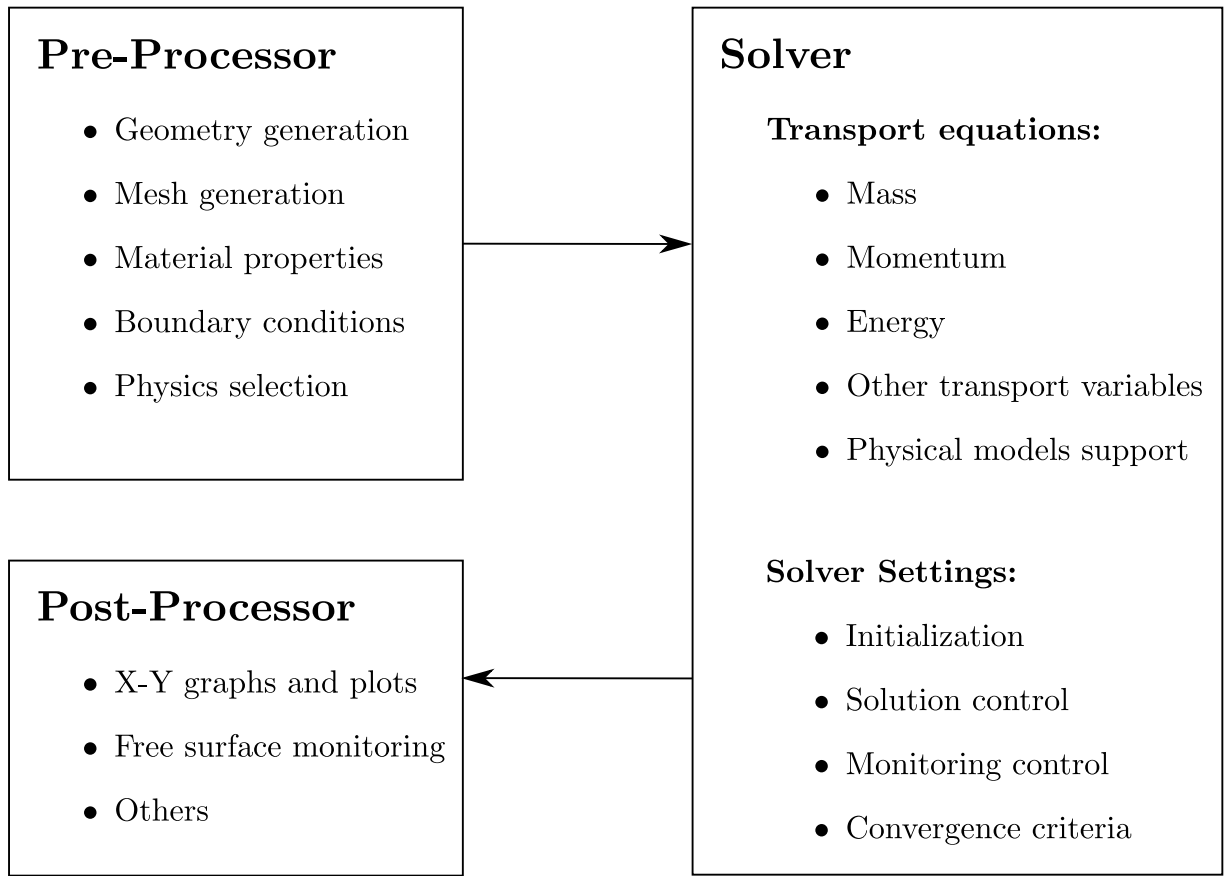


Figure 3.8: Computational Fluid dynamics workflow divided in three main tasks: pre-processor, solver and post-processor. [13]

3.6.1. Solution Errors and Uncertainty

The solver includes the continuous solving of the system of algebraic equations by means of numerical methods, which results in an approximate solution of the problem. It is important to point out, that the approximate solution is affected by uncertainties and errors through numerical calculations.

The errors of the solution can be separated in acknowledged and unacknowledged errors. The former ones can be identified and removed, or their value can be estimated. The latter are not under control and may occur in the simulation. Overall, simulation errors and uncertainties can be classified as follows [14], [15]:

Acknowledged Errors

- **Discretization errors** appear due to the representation of the governing equations as algebraic expressions in a discrete spatial domain (grid) and time (discrete time steps). This type of error is strongly affected by the density of the mesh. For this reason, the grid convergence study has to be implemented to determine their order.
- **Computer round-off errors** occur because of the limited number of computer digits given for a storage of physical value on the computer. They are not significant compared to other error sources and can be minimized by choosing higher precision of the floating number storage.
- **Iteration errors (convergence)** are defined as the difference between the exact and the iterative solution of the discretized equations. These errors are inherent to any numerical computation and they can be minimized by reducing the residual value to the desired level.
- **Physical modeling errors** result due to the idealization of models, which are used to reduce computational time (turbulence). Other processes, like multi-phase flow are difficult to describe precisely, so therefore approximate models are used.

Unacknowledged Errors

- **Computer programming errors** are mistakes in the CFD-code. These errors can be removed by performing validation studies of the code.
- **Usage errors** are caused by inexperience in handling CFD codes, for example setting wrong boundary conditions or using inviscid flow for computing known as turbulent flow. Usage errors can also exist in the CAD, mesh generation and result-analysis. These errors can be minimized by training and gaining experience in using CFD and other software.

4. CFD-Simulation implementations

This section describes the CFD process using the software OpenFOAM as stated in Section 3.6. First, the selection of the solver and the physics of the problem are presented. Then the geometry and mesh generation is explained, including geometry simplifications and quality mesh statistics. In addition, a grid study is performed to approve the mesh independence of the result and to estimate the errors caused by discretization. After that the CFD simulations results are presented. The last section deals with the problem of the transfer function identification for each simulation case.

4.1. Solver and Physics Selection: Volume of Fluid (VoF) Method

The VoF method is a numerical technique used in multiphase flow simulations. [16] The main aspect of this method is the definition of an indicator function, which includes information about, how each cell is occupied by one or another fluid (also mix of fluids). This fraction function represents the volume fraction of one phase. The phase fraction has values between $0 \leq \alpha \leq 1$. The value of zero stands for regions with gas and value of one for regions with fluid, in case of two phase flow with gas and liquid. (Figure 4.1) The numerical simulations are done using free CFD-Software *OpenFOAM* (Open Source Field Operation and Manipulation). The problem definition is implemented into *interDyMFoam* solver, which is convenient for two incompressible, isothermal and immiscible fluids. This solver uses a VoF phase-fraction based on interface capturing approach. It is possible to include mesh motion and a topology change in a *interDyMFoam* solver. [16]

It is important to guarantee the stability of the solver. One necessary stability criteria is the Courant number (or CFL-condition¹), which is defined as follows [14]:

$$Co = \frac{U \cdot \Delta t}{\Delta x}, \quad (4.1)$$

where U is the velocity of the fluid, Δx is a vector between calculation points of control volumes and Δt is a time step. In order to achieve a stable and an accurate run, the maximal Courant number was set to 0.25 in the simulation cases. It is relevant to mention, that the computational time increases by decreasing the maximal Courant number.

¹Courant-Friedrichs-Lewy condition

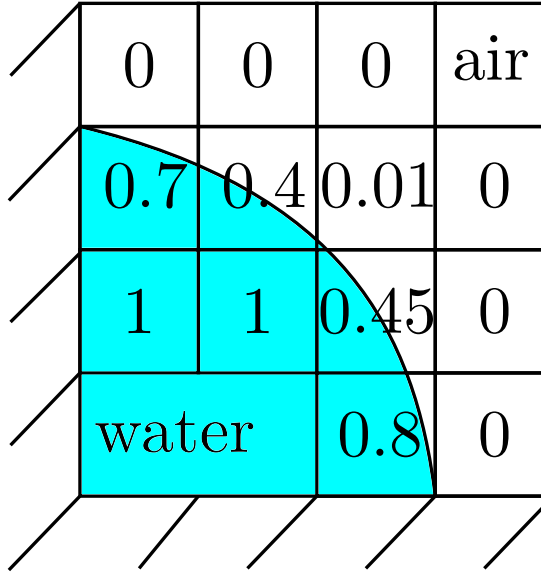


Figure 4.1: Schematic representation of VoF-Method using a phase fraction function. The value of zero stands for the air phase, the value of one stands for the water phase.

To realize stable solver run and accurate results the time step has to be adjusted to the stability criteria, as direct dependency exist between them (4.1). This is done by implementing an adaptive time step control. For each time step the Courant number is calculated. In case that the Courant number is greater than the allowed value, the new time step has to be calculated from flow velocity and the Courant number from the previous time step according to the equation 4.1. This algorithm ensures stable and accurate simulation run. Due to the fact that the Courant number and the velocity from the previous steps are needed, it is important to set a small initial time step, which was set in the following simulations to 10^{-4} s. [7]

4.2. Case Definitions

As mentioned before, the satellite spins around its longitudinal axis. This is defined as the y -axis. Due to the rotation, sloshing in circumferential, radial, and longitudinal axis have to be identified by applying a step function (acceleration) in each direction (Figure 4.2). It results in three simulations for each spinning rate.

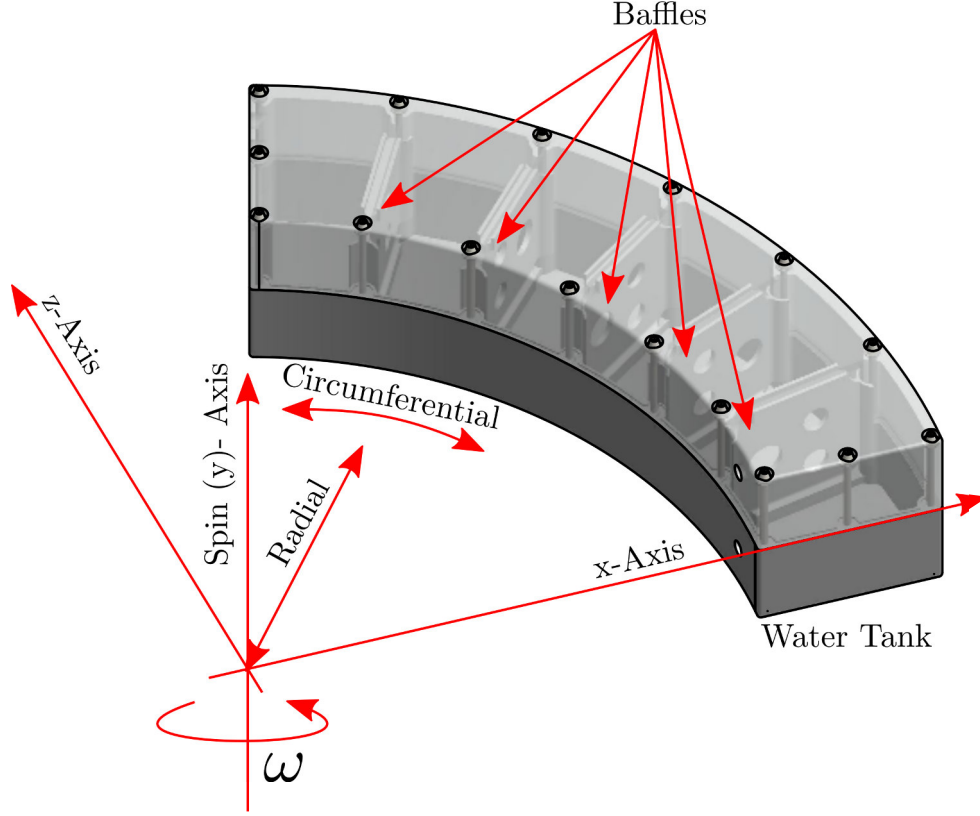


Figure 4.2: Axis and relevant direction definition of the water tank with mounted baffles. Sloshing in three relevant directions was analyzed: circumferential, radial and longitudinal.

Circumferential direction

To realize the step of acceleration in circumferential direction circular motion with constant acceleration was used. To describe the mesh motion in the simulation the position of the water tank for each time step is needed. In this case, the angular travel ϕ is used, defined as:

$$\phi = 0.5 \cdot \alpha \cdot t^2 + \omega_0 \cdot t, \quad (4.2)$$

where α is angular acceleration, t is time and ω_0 is the initial angular velocity. The initial gas-water phase was defined, as the steady state for constant initial spin rate ω_0 . In this case, the water is pressed by centrifugal forces against the outer wall of the water tank, as shown in the Figure 4.4. The water also has the initial velocity defined as follows:

$$V_{initial} = \omega_0 \times r_{cell}, \quad (4.3)$$

where r_{cell} is the radius from the rotating center to each mesh point and ω_0 is the initial angular velocity.

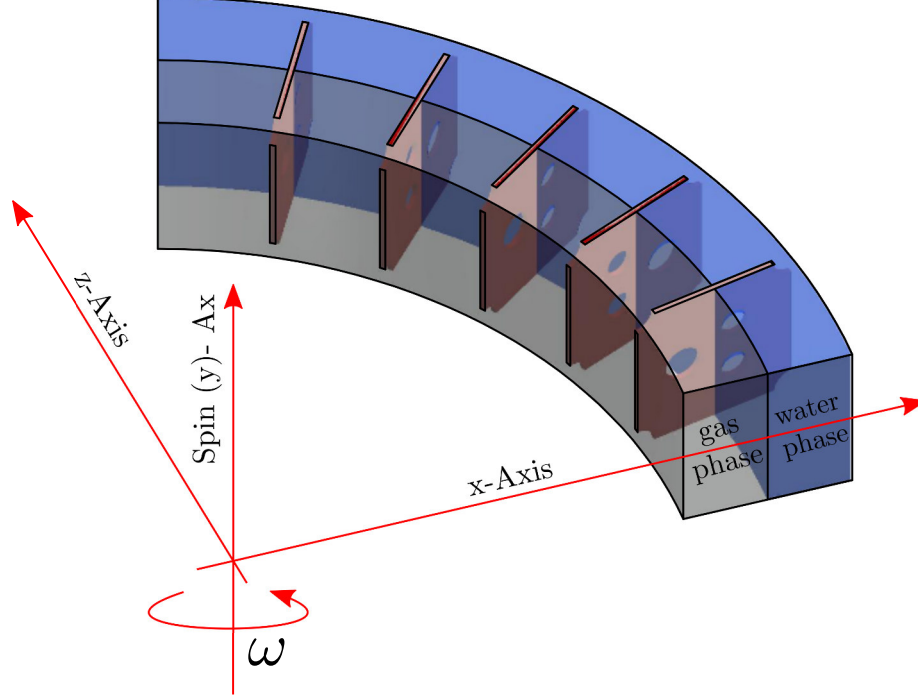


Figure 4.3: Initial gas-water phase to identify sloshing in circumferential and longitudinal direction. This position is the steady state solution in case water tank spins with a constant angular velocity.

Radial direction

To realize the step in radial direction, there is another initial gas-water phase required. In this case the water is set to the bottom of the water tank, but the initial velocity was defined as in previous case (Eq.: 4.3). To cancel out the sloshing in circumferential direction there is no angular acceleration, so the angular travel is calculated by:

$$\phi = \omega_0 \cdot t, \quad (4.4)$$

where ω_0 is the initial angular velocity and t is time. With the first time step the water experienced the step in acceleration caused by centrifugal forces as result of the spinning and starts to reorient to the water tank outer wall. The Figure shows the initial gas-water phase, which identifies sloshing in a radial direction.

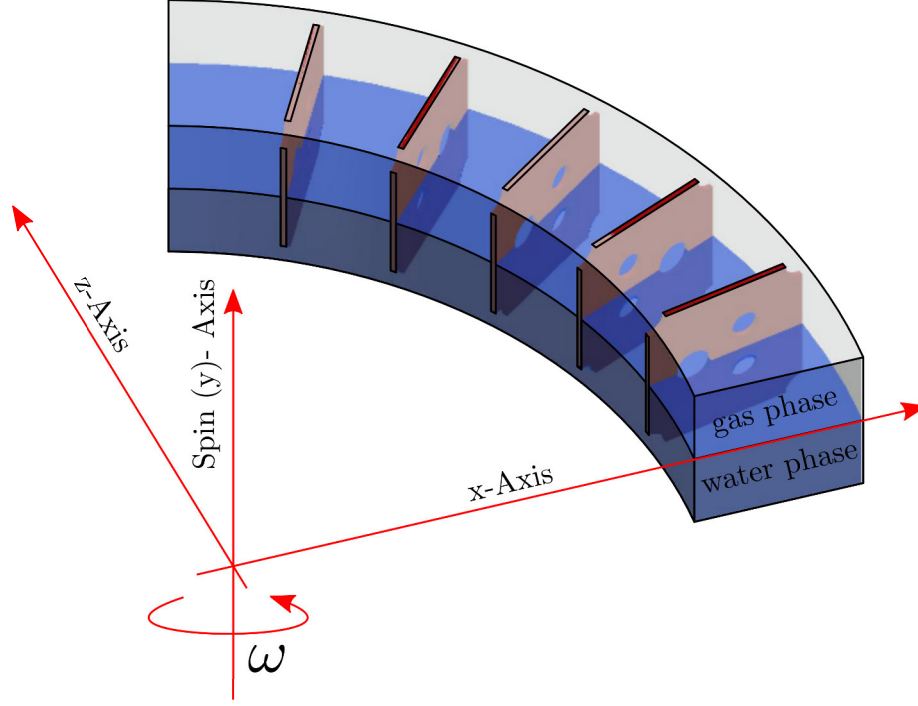


Figure 4.4: Initial gas-water phase supposed to identify sloshing in radial direction.

Longitudinal direction

Here, the rotating water tank with constant spin rate is translated in y -direction with a constant acceleration. So the angular travel is calculated as in Eq.: 4.4 and the translation position y as follows:

$$y = 0.5 \cdot a_y \cdot t^2, \quad (4.5)$$

a is the acceleration in y -direction and t is time. The initial velocity in the y -direction is set to zero. The initial gas-water phase is equal to the case of circumferential direction, which is showed in Figure 4.3. The initial velocity was calculated as in Eq.: 4.3. Following Table 4.1 summarizes all numerical simulations with their important initial conditions, which have to be implemented in *OpenFOAM*:

	Direction	Acceleration	Initial spin rate ω_0	Baffles
Case 1	Circumferential	$3 \frac{\text{deg}}{\text{s}^2}$	$0.5294 \frac{\text{rad}}{\text{s}}$	no
	Radial	$0.0981 \frac{\text{m}}{\text{s}^2}$	$0.5294 \frac{\text{rad}}{\text{s}}$	no
	Longitudinal	$0.001/0.01 \frac{\text{m}}{\text{s}^2}$	$0.5294 \frac{\text{rad}}{\text{s}}$	no
Case 2	Circumferential	$3 \frac{\text{deg}}{\text{s}^2}$	$0.5294 \frac{\text{rad}}{\text{s}}$	yes
	Radial	$0.0981 \frac{\text{m}}{\text{s}^2}$	$0.5294 \frac{\text{rad}}{\text{s}}$	yes
	Longitudinal	$0.001 \frac{\text{m}}{\text{s}^2}$	$0.5294 \frac{\text{rad}}{\text{s}}$	yes
Case 3	Circumferential	$3 \frac{\text{deg}}{\text{s}^2}$	$2.1177 \frac{\text{rad}}{\text{s}}$	yes
	Radial	$1.5696 \frac{\text{m}}{\text{s}^2}$	$2.1177 \frac{\text{rad}}{\text{s}}$	yes
	Longitudinal	$0.001 \frac{\text{m}}{\text{s}^2}$	$2.1177 \frac{\text{rad}}{\text{s}}$	yes
Case 4	Circumferential	$3 \frac{\text{deg}}{\text{s}^2}$	$3.2636 \frac{\text{rad}}{\text{s}}$	yes
	Radial	$3.7278 \frac{\text{m}}{\text{s}^2}$	$3.2636 \frac{\text{rad}}{\text{s}}$	yes
	Longitudinal	$0.001 \frac{\text{m}}{\text{s}^2}$	$3.2636 \frac{\text{rad}}{\text{s}}$	yes

Table 4.1: Simulation cases with their initial states realized in *OpenFOAM*. The direction in which sloshing can occur was the relevant parameter to classify the simulations. Step of acceleration in each direction was applied to identify the liquid sloshing behavior.

4.3. Pre-Processing and run controls

This section gives an overview about the relevant settings and assumptions made for the pre-processor, including geometry simplification and mesh generation, but also boundary conditions and physics selection. In addition, relevant parameters of solver settings and run controls are discussed. The detailed information about transport properties, discretization schemes and solution controls in *OpenFOAM* is attached to the document as Appendices A, B and C

4.3.1. Geometry and Mesh generation

The mesh generation process was realized with a *ANSYS* software and then exported to *OpenFOAM*. Before the mesh generation process was started, geometry was simplified to

reduce the cell number and associated computational time. This process of simplification affects also the grid quality. Nevertheless, all important physical properties of the geometry have to be retained.

First, the geometry simplification and the mesh generation process of the water tank without the baffles is described. In this case, the baffles and the other connection elements, like screws, are canceled. The result is sector-annular cylindrical tank as a solid body. All of the edges radii were removed. The simplified meshed geometry for the tank without baffles is shown in Figure 4.5.

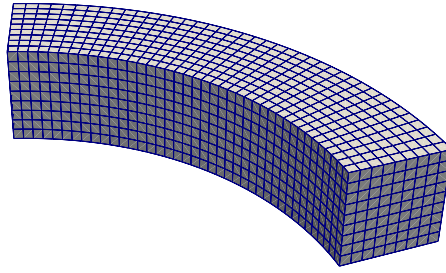


Figure 4.5: Simplified geometry of the tank, which is meshed by hexahedra cells resulting in a structured mesh.

It is important to check the mesh quality before the simulation. There are three important quality metrics:

- Non-Orthogonality measures the angle between the line connecting two cell centers and the normal between their common face. This parameter has a value between 0 and 180, however 0 is the better value. [7]
- Skewness measures the distance between the intersection of the line connecting two

4. CFD-Simulation implementations

cell centres with their common face and the centre of that face. Skewness has a value between 0 and 1, a smaller value is better.[17]

- Aspect Ratio describes the stretching of the cell and is the ratio between the longest and the shortest length. It has a value between 0 and 1, however 1 is the better value.[17]

Bad mesh quality parameters can decrease the accuracy of the solution, make the solver slow or stop the simulation. Good mesh quality is necessary to get accurate results.[18] Due to simple geometry, in this case, it is possible to generate structured mesh, which lead to a good quality mesh. The mesh metrics parameters are summarized in Table 4.2:

Mesh Metrics	Max. Value	Recomended value [18]
Non-Orthogonality	0.0523701	< 60
Skewness	0.0134738	< 0.9
Aspect Ratio	1.49973	< 40

Table 4.2: Mesh statistics of generated mesh for case without baffles compared with recommended values.

In the case of the implemented baffles inside, the tank geometry is more complex. First the simplification of baffles was done, each baffle thickens was reduced to constant 3 mm. The most important part of the baffles are the openings, trough which water can move from one sector to another. For this, three holes in the middle and quarter-holes on each edge of the baffles are made. In the end, these baffles were subtracted in the defined orientation from the simplified tank described before. The result was a solid body including all the connections between each sector.

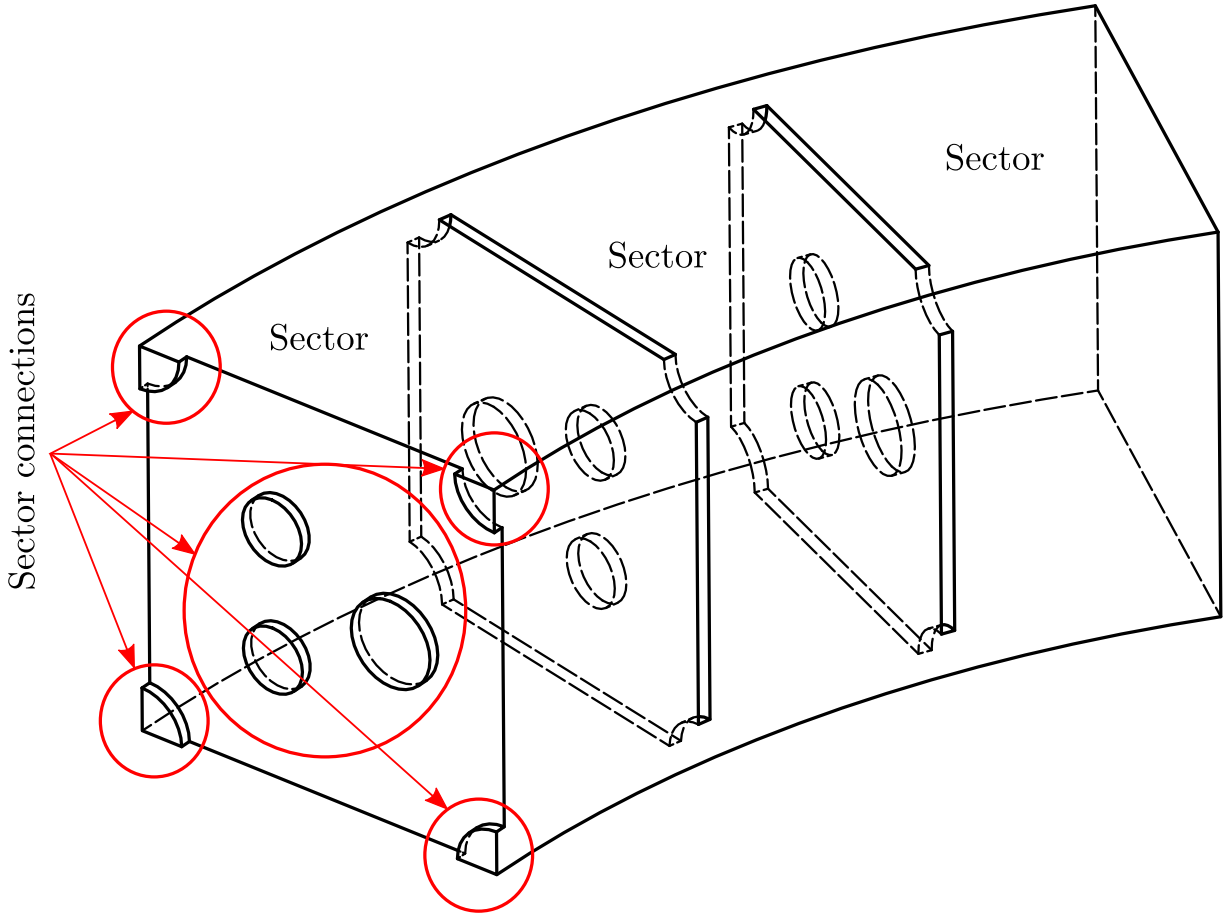


Figure 4.6: Simplified tank geometry with implemented baffles, realized as connections between each sector.

Due to the complex geometry in the case of the baffles, unstructured mesh is realized out of tetrahedra cells, however the cell size was refined in the area of the holes and connections between the sectors. The mesh metrics parameters are summarized in Table 4.3:

Mesh Metrics	Max. Value	Recommended value [18]
Non-Orthogonality	57.8207	< 60
Skewness	0.6895	< 0.9
Aspect Ratio	5.07	< 40

Table 4.3: Mesh statistics of a generated mesh for a case with baffles compared with recommended values.

The mesh statistics are poorer compared to the case without baffles, however all of the parameters are in the range of the recommended value. This mesh was chosen for all simulation cases with baffles. It is important to note, that this mesh is not the ideal one, yet the best compromise between the accuracy and computational time. To estimate the order of discretization error the grid study in Section 4.4.2 is realized, where finer grids are also compared. The simplified meshed geometry for the tank with baffles is shown in Figure 4.7:

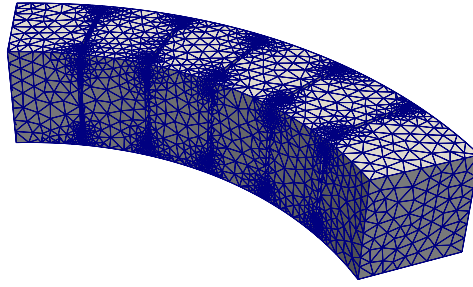


Figure 4.7: Simplified geometry of the tank with implemented baffles. The mesh is unstructured out of the tetrahedra. The size of the grid was refined in the area of the sector connections to improve the quality of the mesh.

4.3.2. Boundary Conditions

After the geometry and mesh generation process, boundary conditions have to be set. All faces of the geometry are set to boundary condition of a type wall. This means a rigid wall, which does not allow fluid flow through these faces.

The hydrostatic pressure in the water tank is negotiable small, therefore the initial hydrostatic pressure in the tank was set to 0 in all cases. The gas-liquid phase was realized as an air-water phase, which is an assumption to the real boundary conditions in the tank during the mission. In reality there is water, the mix of water vapour and

4. CFD-Simulation implementations

nitrogen gas. This assumption is valid due to the fact that the density of the water is a factor of 1000 higher than nitrogen or air. Only the water sloshing affects significant CoM shift of the water tank.

The initial velocity of the water and gas were set dependent on the simulation case. In case of a grid study, the velocity was set to $0 \frac{\text{m}}{\text{s}}$, however in other simulations it was set to the velocity, which equals the initial spin rate.

Water and air are Newtonian fluids, so consequently the transport model was set to Newtonian in a transport properties definition. As mentioned before the surface tension forces are small compared to the centrifugal and inertia forces, so the surface tension of each fluid was set to 0. Table 4.4 summarizes the transport properties of the simulations:

Phase	Density $[\frac{\text{kg}}{\text{m}^3}]$	Viscosity $[\frac{\text{m}^2}{\text{s}}]$	Transport model	Surface tension $[\frac{\text{kg}}{\text{s}^2}]$
Water	998.2	10^{-6}	Newtonian	0
Air	1	$1.48 \cdot 10^{-5}$	Newtonian	0

Table 4.4: Transport properties of air and water phase. Only water motion results in significant CoM shift, due to the higher density. Surface tension is neglected for both, water and air.

The flow was assumed to be laminar, so no physical turbulence model was used. The main point of the simulations was to realize a micro-gravity environment. Here, the assumption was made, that there is no constant gravity vector, so in the simulations it was set to $0 \frac{\text{m}}{\text{s}^2}$ in all directions.

4.3.3. Solver Settings and Run Controls

There are different terms, that appear in simulation run, like for example derivatives or gradients, different numerical schemes have to be specified and chosen. The Table 4.5 summarizes the numerical schemes chosen for different terms:

4. CFD-Simulation implementations

Term	Numerical scheme
Time derivative	Euler
Gradient	Gauss linear
Convective	Gauss vanLeerV/Gauss vanLeer/Gauss linear
Diffusive	Gauss linear corrected / limited
Interpolation	linear

Table 4.5: Numerical schemes chosen for different term in simulation run. High order numerical schemes yield accurate solutions.

The choice of discretization schemes has direct effect on the final solution, but also on the stability. The first order schemes are diffuse and yield smeared gradients, however they are stable compared to the high order schemes. The mesh quality has direct effect on which scheme has to be chosen, so with a good mesh quality directly high order schemes yield accurate and stable solutions, which were realized in this simulation cases. Beside, numerical schemes, the equation solvers, tolerances and algorithms have to be defined for the simulation run. [18]

For the phase equation a PISO-algorithm is used, which includes sub-cycles around it, the phase equation is solved 3 times in implemented simulations.[7]For each discretized equation there is a possibility to define their own linear solver. In order to achieve an accurate solution, it is important to set the solution tolerances to a small value. This is the solver criteria, in order to stop solving the equation. The detailed parameter overview implemented for this simulation is attached as Appendix B.

4.4. Results discussion

In this section the simulation results are discussed. First, the results of the grid convergence study are given in order to determine the discretization error caused by the grid generation process. After that, the simulation results for the liquid sloshing identification with a calculated transfer functions are presented.

4.4.1. Data Processing

In the OpenFOAM software, an implemented script writes the position of the CoM in x,y and z coordinates. However, this includes the CoM of the moving tank and the fluid inside of it. In order to obtain the CoM shift caused by the liquid motion, the tank motion needs to be subtracted.

For the circumferential sloshing case, the CoM motion of the tank can be expressed in x- and y- directions as follows:

$$x(t) = r \sin(\phi t - \phi_0), \quad (4.6)$$

$$y(t) = r \cos(\phi t - \phi_0), \quad (4.7)$$

$\phi = 0.5at^2 + \omega_0 t$ is the angular position of the tank and has a value from 0 to infinite, r is the initial radius of the CoM and ϕ_0 is the angle offset.

Due to the rotational movement of the tank, it is useful to transform both data from Cartesian to Polar coordinates. After the transformation the data has the value between $+\pi$ and $-\pi$, including jumps after every turn of the tank. Next, the simulated CoM data is subtracted from the calculated tank position. Due to the fact that both movements are not exactly the same, there are still jumps in the graphs with an exact value of 2π . This jumps were canceled by adding, respectively subtracting, the value of 2π at the jump. The result is the relevant CoM shift, only caused by water motion. This is the angle variance in radiant refer to the ideal movement of the tank.

In the radial direction the radius is calculated by:

$$r = \sqrt{x^2 + z^2}, \quad (4.8)$$

where x and z are the CoM coordinates of the rotating tank. The simulation data has the offset of ≈ 0.325 m due to the initial CoM radius. This offset was canceled later

on for a transfer function determination. For the longitudinal direction there was no data processing needed. Because of the time step adjustment loop implemented in the numerical simulations, the data has noise, so the data was filtered. This was necessary to realize the transfer function calculation.

The free surface movement of the air-water phase was analyzed in order to validate physical integrity of the fluid flow. However, the free surface motion is similar for each simulation case, only the grid convergence study results are presented, in form of figures.

4.4.2. Grid Convergence Study

A Grid convergence study is a method to determine the order of discretization errors in a CFD simulations. The approach is to perform the same simulation cases on multiple fine grids. With every finer grid the discretization errors asymptotically decrease to the true solution, excluding round-off errors. The method, used here, is based on Richardson's extrapolation.[19]

Four grids, generated with *ANSYS* Software, were used. The Table 4.6 summarizes the grids with their quality parameters. The overall quality of the elements, including the parameters described before, were used here. The mesh quality has a value between 1 and 0, where 1 is the better value. The value of 0.4 is acceptable by regarding, that only few elements have this value. Most of the elements have the value > 0.7 .

Mesh Nr. Parameter	Mesh 1	Mesh 2	Mesh 3	Mesh 4
Number of Elements	36891	57875	77719	160229
Min. Quality	0.39	0.39	0.39	0.37

Table 4.6: Mesh statistics for different grids used for the grid convergence study. The number of elements was increased by refining the mesh on the sector, but also on the connections between them.

The case was defined similar to the simulation case in circumferential direction, the starting air-water phase was set to be on the outer tank wall. The liquid start velocity

4. CFD-Simulation implementations

was set to $0 \frac{\text{m}}{\text{s}}$, however the spin rate was set to constant $0.5294 \frac{\text{rad}}{\text{s}}$ without acceleration in any direction. The spinning was in a counterclockwise direction. The simulation time was set to 100 s. Figure 4.8 shows the calculating air-water phase for the finest mesh at different time points.

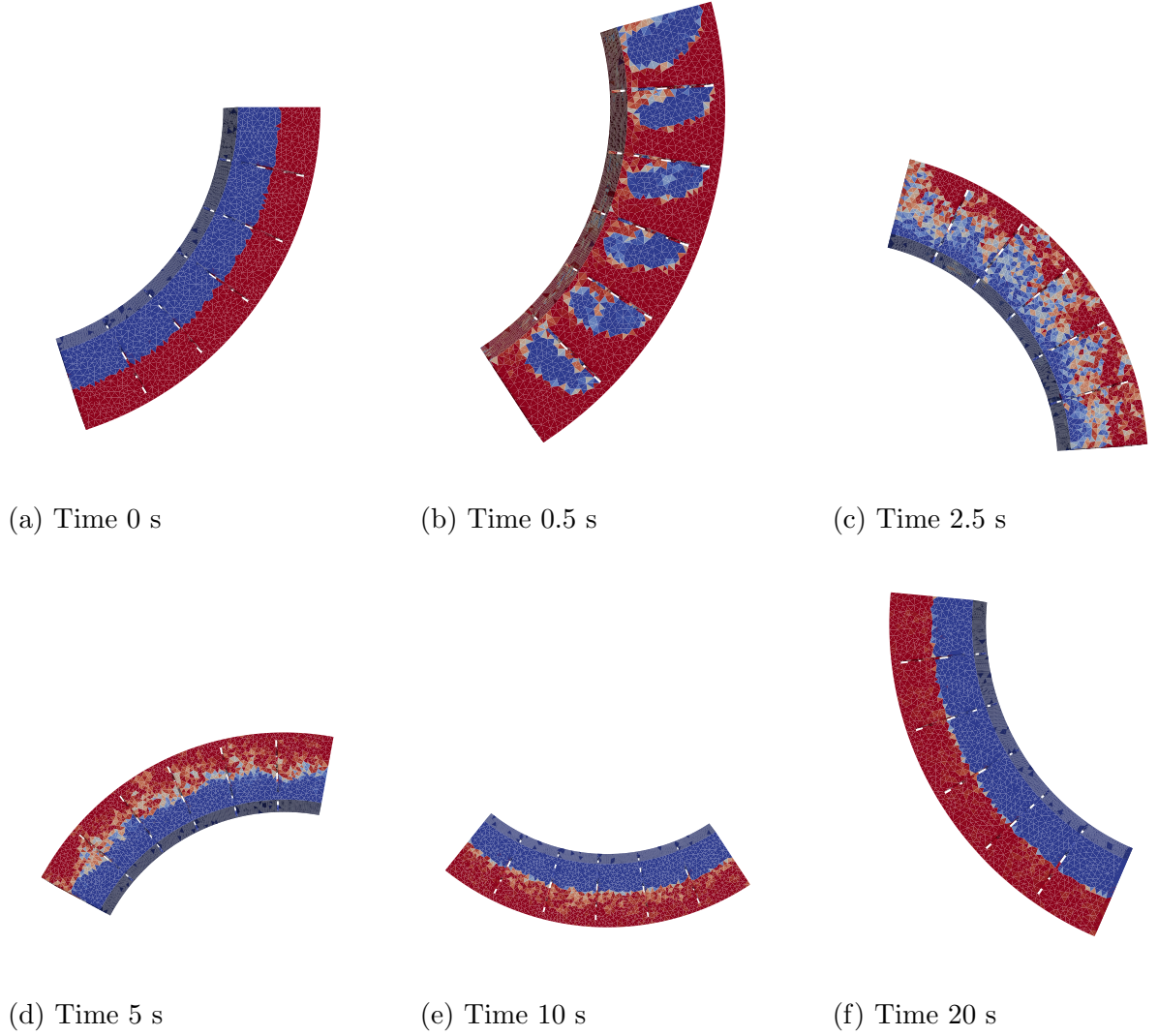


Figure 4.8: Example results of the grid convergence study. Calculated Water-air phase at different time points for the finest grid. Blue is equal to the air phase, red to the water phase.

4. CFD-Simulation implementations

The initial state with the water phase at the outer tank wall is defined at time 0s. With the next time step 0.5 s the water reorientation, due to the inertia forces caused by the sudden spinning movement, is seen. At time 5 s, the water starts to reorient at the outer tank wall, because of the centrifugal forces. At time 20 s there is an almost steady state position of the water-air phase. These figures are helpful for a validation of the water movement.

For the grid convergence study, the CoM variation in the circumferential direction was observed, until the steady state was achieved. The last values of the state were chosen as the final solution for each case, which also were used for the extrapolation described below. Figure 4.9 shows the results of the grid convergence study including four meshes.

The steady state values of each mesh are used for the extrapolation, these are summarized in Table 4.7:

Mesh Nr.	Value [rad]
Mesh 1	-0.009865
Mesh 2	-0.007497
Mesh 3	-0.006683
Mesh 4	-0.006014

Table 4.7: Steady state solutions of each grid. With finer mesh the solution approximates to the true solution.

The method of Richardson's extrapolation is used to determine the order of discretization errors caused by the grid. The Richardson's extrapolation calculates the final true solution and is defined as follows:

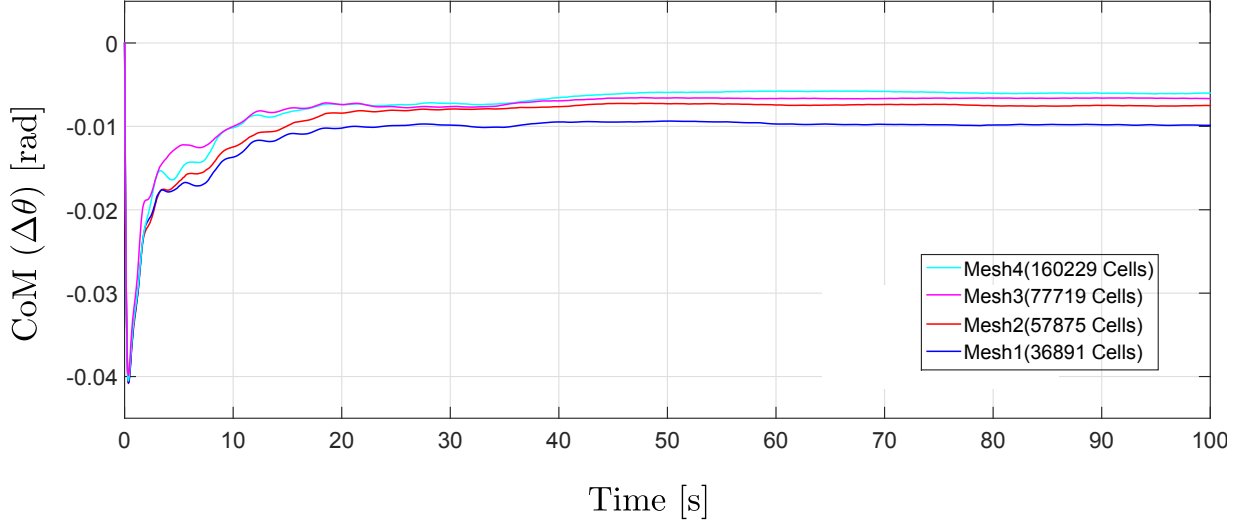


Figure 4.9: Grid convergence study for the 4 different meshes. The CoM position over time, subtracted from the tank movement, is shown. With higher grid refinement, the steady state solution converge to the true solution and the discretization error became smaller.

$$f_{true} \approx f_2 + \frac{(f_1 - f_2)r^p}{r^p - 1}, \quad (4.9)$$

where $f_1 - f_2$ are the solutions of the grid, r is the refinement ratio and p is the order of convergence. Since the refinement ratio is not constant, the effective grid refinement ratio was used instead:

$$r_{effective} = \left(\frac{N_1}{N_2} \right)^{\frac{1}{D}}, \quad (4.10)$$

where N is the number of grid points and $D = 3$ is the dimension of the 3D-flow domain. The order of grid convergence can be calculated from a steady state solutions of the following three grids:

$$p = \frac{\ln \left(\frac{f_3 - f_2}{f_2 - f_1} \right)}{\ln(r)}, \quad (4.11)$$

where f_1, f_2, f_3 are the values of the steady state solutions and $r_{effective}$ is the effective refinement ratio. The Results of the extrapolation are shown in Figure 4.10, including the

4. CFD-Simulation implementations

extrapolated true solution. For the simulations the mesh number 3 with 77719 cells was used. The error compared to the true solution is 13.3 %. This is an acceptable compromise between computational time and error in this stage of research. Nevertheless this error has to be implemented into the AOCS simulation.

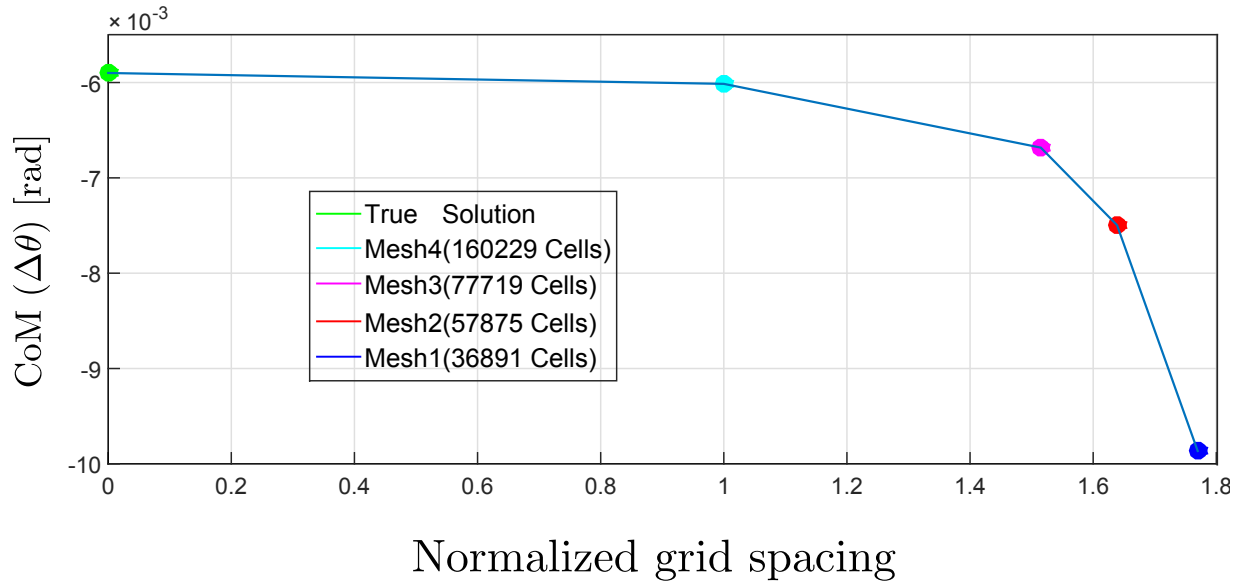


Figure 4.10: Richardson's extrapolation including the steady state results of four different meshes and extrapolated true solution.

4.4.3. Simulation Results

Circumferential direction without baffles $\omega = 0.5294$

As described in Section 4.4.1, the simulations data was post-processed. The results are shown in Figure 4.11. The first graph shows the filtered CoM data in circumferential direction as an angle variation from the initial tank movement, which is only caused by the liquid sloshing. The tank motion starts with $\omega = 0.5294 \frac{\text{rad}}{\text{s}}$ with a step in acceleration of $3 \frac{\text{deg}}{\text{s}^2}$, so the velocity starts to increase continuously. As expected the angle becomes negative with an oscillation caused by the step in acceleration with the starting tank motion. The amplitude becomes smaller until the steady state is reached. After $t=80$ s the amplitude of the oscillation starts to rise again. This is caused by the numerical problems, due to the fact, that the velocity has become too fast and at the same time there is no big motion in the water-air phase. This phenomena was observed over all the simulations. This effect increases with the higher starting rotation rate. More studies are needed to get rid off this numerical problem, yet the results up to this point are valid.

The second graph is a zoom of the first graph and shows the comparison between un-filtered (red) and filtered (green) data. As mentioned before this noise is caused by an adjustable time step control loop, which is implemented in *OpenFOAM*.

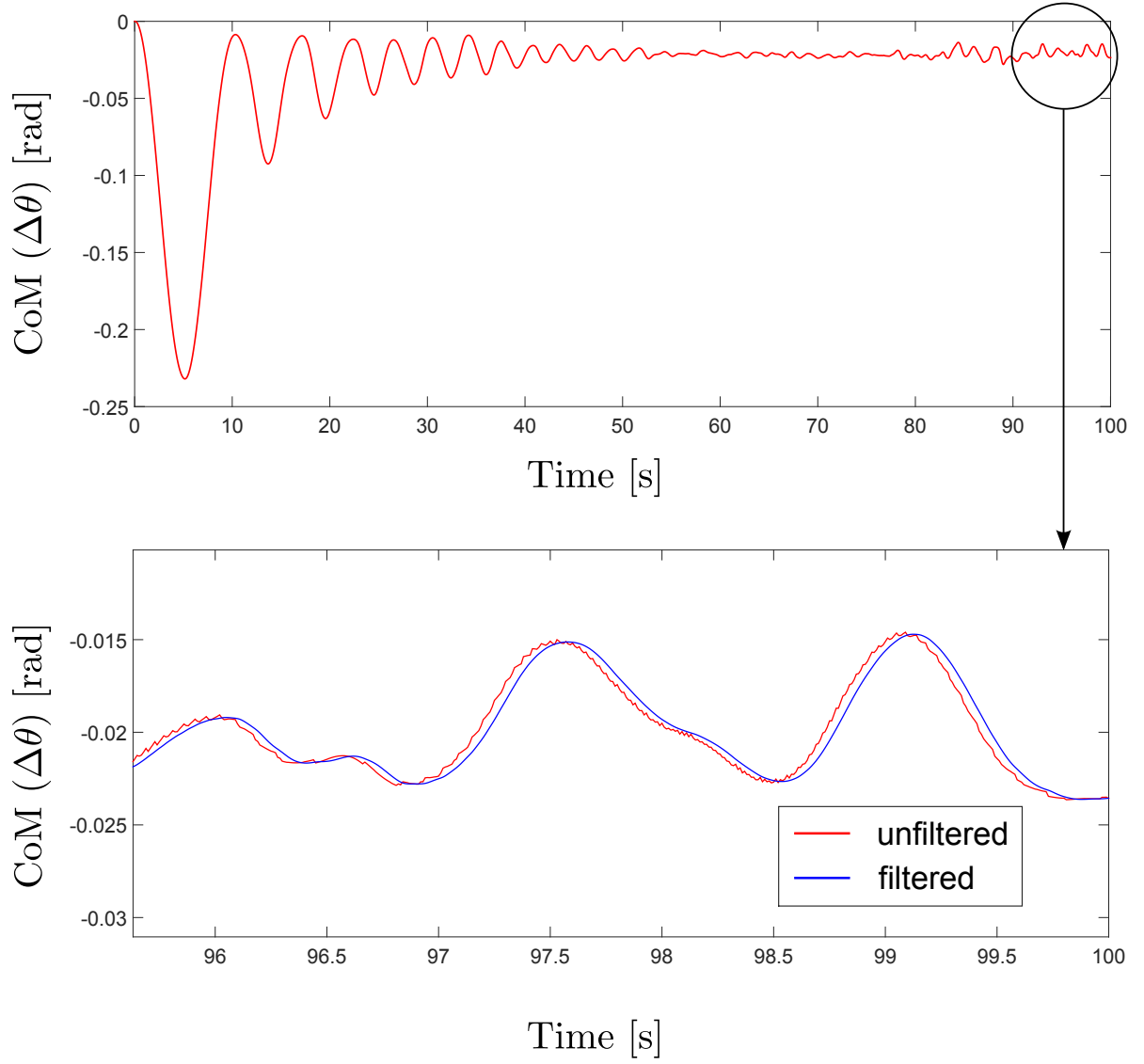


Figure 4.11: Sloshing in circumferential direction of the tank without baffles with the starting spin rate of $\omega_0 = 0.5294 \frac{\text{rad}}{\text{s}}$ and step in the acceleration of $3 \frac{\text{deg}}{\text{s}^2}$. The top graph shows the filtered data over the time range of 100 s. The graph below is a zoom in of the first graph starting at $t = 95$ s and shows the noise of the unfiltered data (red) compared to the filtered (blue).

Longitudinal direction without baffles $\omega = 0.5294$

Figure 4.12 shows the case results in a longitudinal direction of the tank without the implemented baffles. The CoM position in the y-direction in meter is plotted over the time. The spin rate is constant at $\omega_0 = 0.5294 \frac{\text{rad}}{\text{s}}$, the starting velocity of the fluid is also adapted to the spin velocity. Parallel to this motion the tank is accelerated by $a_y = 0.01 \frac{\text{m}}{\text{s}^2}$ along the y-axis.

With the starting acceleration in a positive direction the CoM starts moving in the negative direction and starts to oscillate. The behavior of the sloshing corresponded to the weakly damped system. After 40 s the oscillation is decayed. After ca. 45 s numerical problems occur again, however there is no relevant physical meaning anymore.

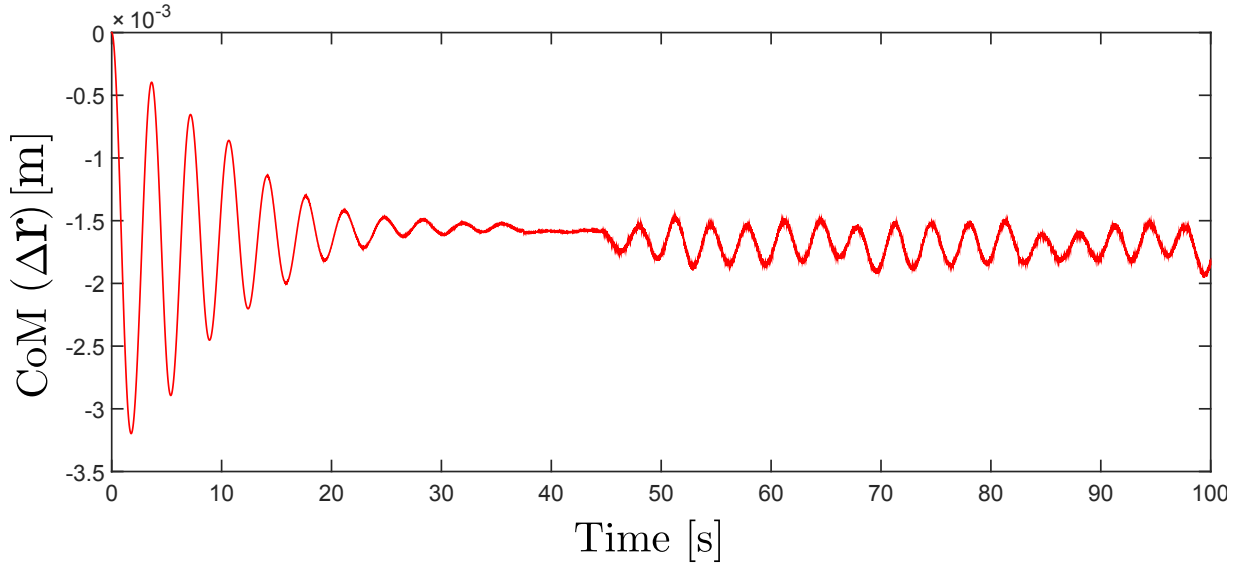


Figure 4.12: Sloshing in longitudinal direction of the tank without baffles with a constant spin rate of $\omega_0 = 0.5294 \frac{\text{rad}}{\text{s}}$ and a step in the acceleration along y-axis of $0.01 \frac{\text{m}}{\text{s}^2}$.

Radial direction without baffles $\omega = 0.5294$

Figure 4.12 shows the CoM variation in a radial direction. As expected, with a starting rotation rate, the water is pressed against the outer wall. This leads to CoM oscillation and shift in a positive radial direction. However, there is a CoM shift in longitudinal direction along the y-axis. The CoM motion along y-axis is shown in Figure 4.13. In this case, the acceleration of $0.0981 \frac{\text{m}}{\text{s}^2}$ is generated in one single time step. In reality, the equivalent g-level is generated slowly.

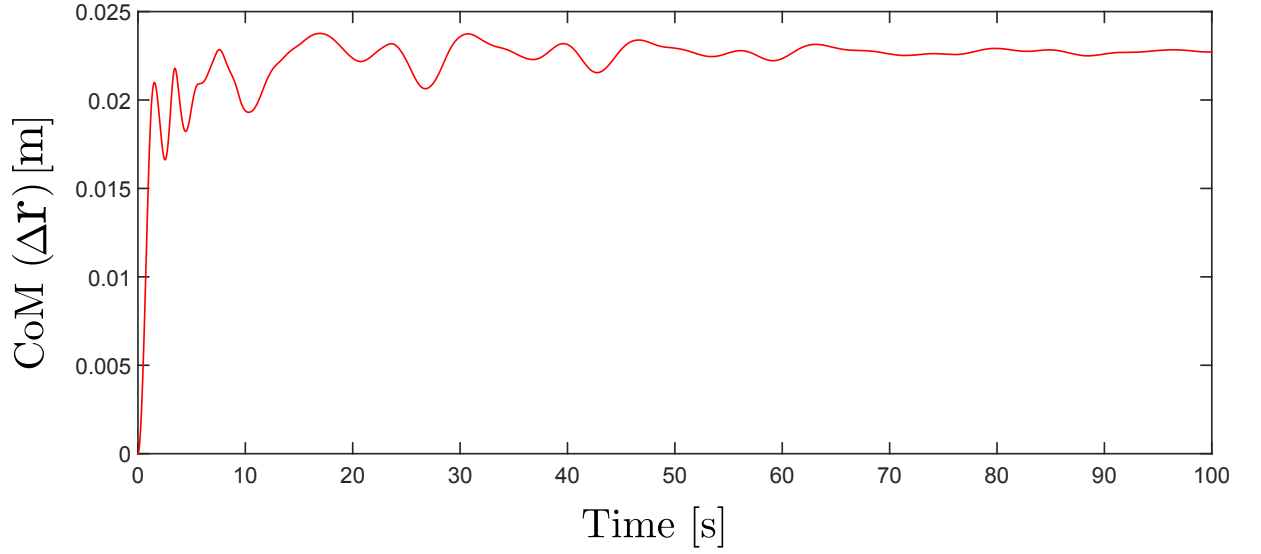


Figure 4.13: Sloshing in radial direction of the tank without baffles with constant spin rate of $\omega_0 = 0.5294 \frac{\text{rad}}{\text{s}}$ and a step in the acceleration of $0.0981 \frac{\text{m}}{\text{s}^2}$.

Comparison between simulation results of the tank with and without implemented baffles $\omega = 0.5294$

In order to demonstrate the effect of the implemented baffles, a comparison between simulation results of the tank with and without implemented baffles were made. All boundary conditions, including the water-air phase definition, but also the tank motion are equivalent. The only difference is the implemented baffles.

Figure 4.14 shows the results of two equivalent simulation cases for tank with (green) and without (red) the implemented baffles. As expected, implemented baffles damp the sloshing movement in a circumferential direction, therefore the amplitude is smaller, compared to the tank without baffles. Even the time, in which the sloshing is decayed, is reduced compared to the case without baffles. The sloshing behavior of the tank with baffles correlates to the behavior of a damped system. This demonstrates the baffles effect in order to reduce the sloshing in circumferential direction.

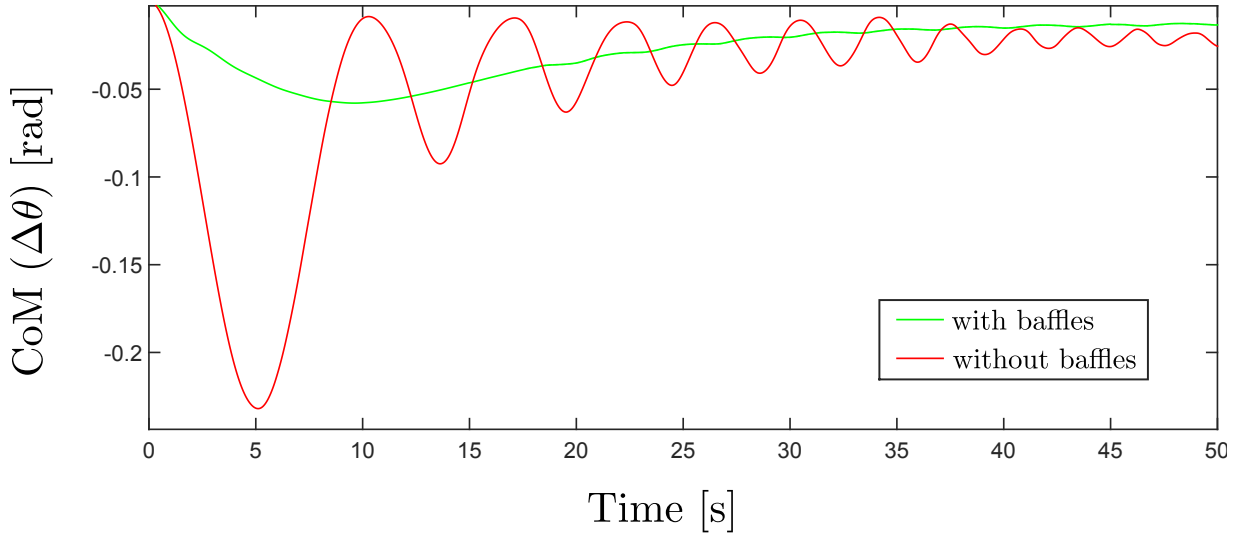


Figure 4.14: Sloshing in circumferential direction at $\omega_0 = 0.5294 \frac{\text{rad}}{\text{s}}$ and step in the acceleration of $3 \frac{\text{deg}}{\text{s}^2}$. Comparison between tank without (red line) and with (green line) implemented baffles.

The comparison of a simulation results in a radial direction between a tank with and without baffles are shown in Figure 4.15, there a similar behavior expected, because there are no baffles implemented in the radial direction. Yet, there are differences between these

two cases: the sloshing in a radial direction with the implemented baffles has a smaller amplitude. Also the oscillation is decayed after 10 s already. The reason of this behavior, is the fact, that the baffles increase the area for the fluid shear stress. This leads to a bigger damping of the liquid sloshing.

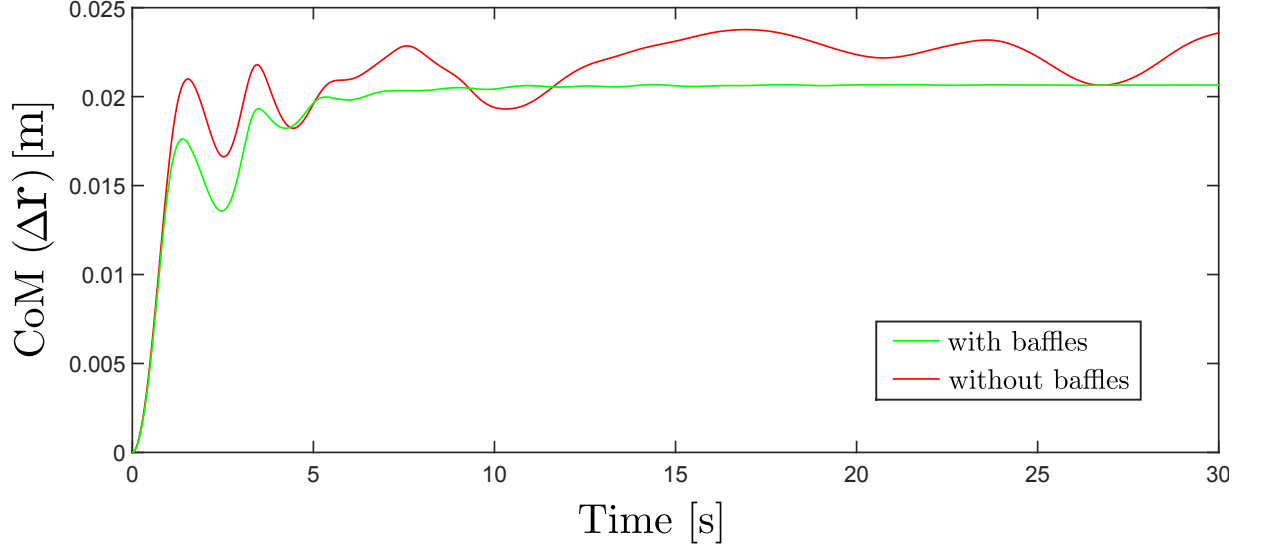


Figure 4.15: Sloshing in radial direction, showing the comparison between the simulation case with and without implemented baffles. There was a the similar behavior expected due to the fact that baffles damp sloshing in a circumferential direction. There is still a damping effect of the baffles in a radial direction, this is caused by liquid separation into smaller parts (increasing the wall area).

Figure 4.16 shows the simulation results in a longitudinal direction of the tank with and without implemented baffles. As expected, the CoM variation along y-axis is for both tank configurations nearly the same, because no baffles are implemented in this direction. The small variations in the amplitude are caused by the increased area of the fluid shear stress in case of the implemented baffles. The CoM variation has the order of 10^{-4} m, which is very small and can be neglected. Therefore no bigger disturbances in the orbit are expected. Summarizing the sloshing in longitudinal direction caused by external acceleration can be neglected.

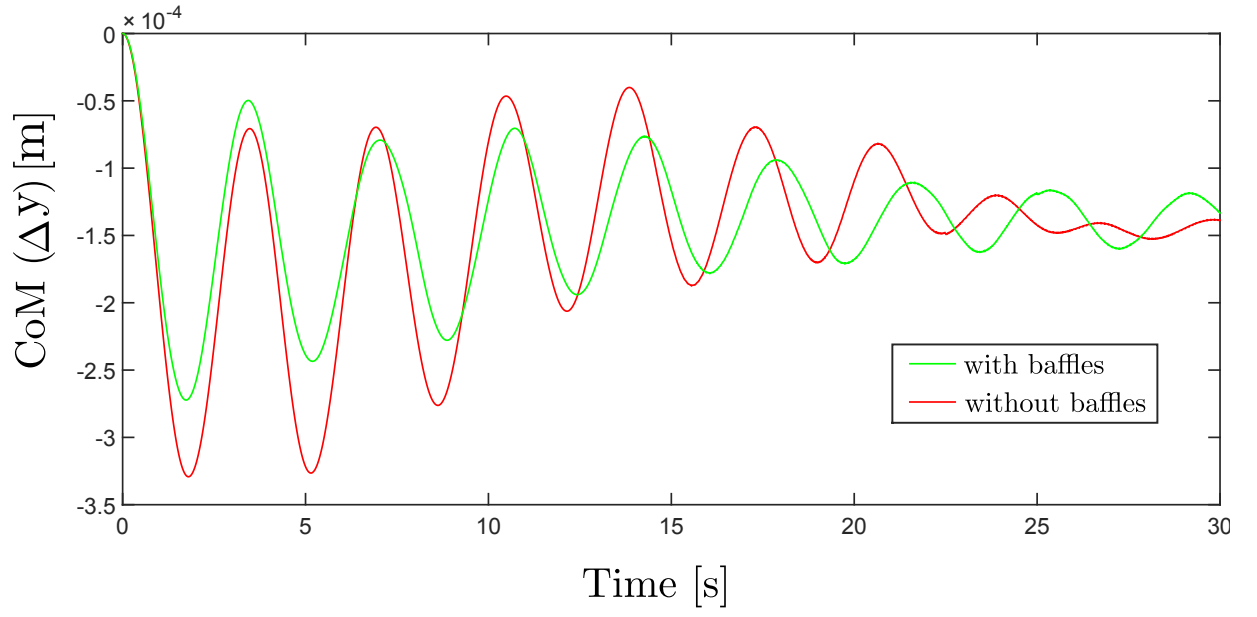


Figure 4.16: Sloshing in longitudinal direction showing the comparison between the simulation case with and without implemented baffles. The acceleration along y-axis has the value of $0.001 \frac{\text{m}}{\text{s}^2}$.

Simulation results for the tank with implemented baffles at different spin rates

For the tank with implemented baffles the sloshing in circumferential, radial and longitudinal direction at different spin rates was analyzed. The result is the CoM motion in each relevant direction at different spin rates. With this data, transfer functions were generated. The transfer functions for spin rates between analyzed cases can be interpolated. In order to realize comparable data, same cases were used, only the initial spin and liquid velocity were changed respectively.

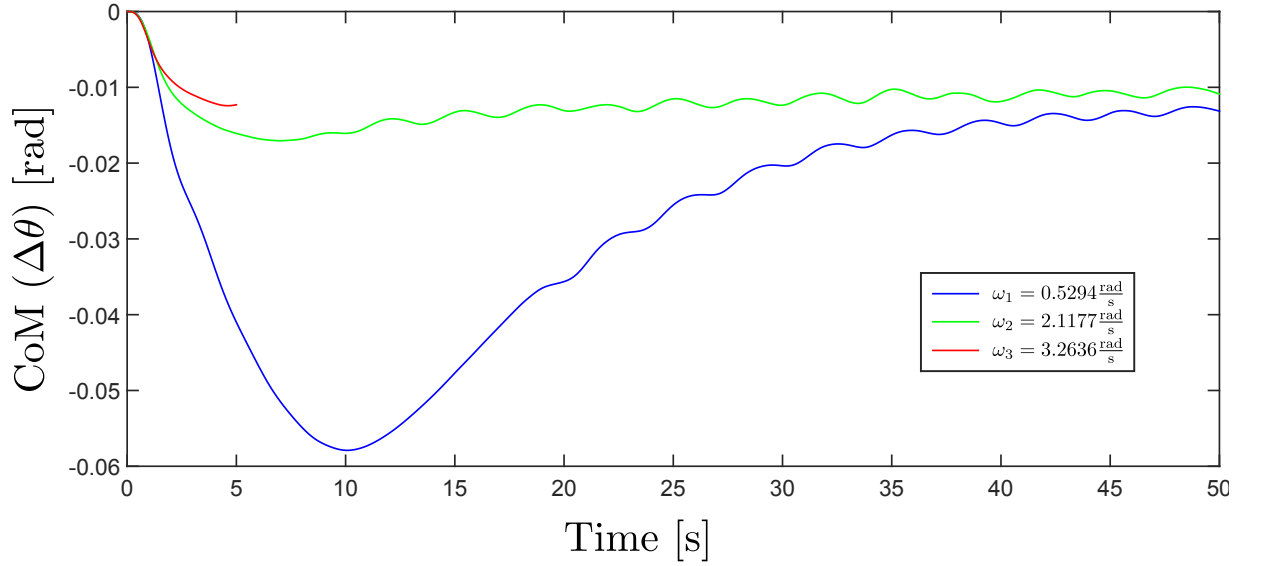


Figure 4.17: Sloshing in circumferential direction of the tank with implemented baffles at different spin rates with the step response in acceleration of $3 \frac{\text{deg}}{\text{s}^2}$. Numerical instabilities occur at higher spin velocities, so for the simulation case at ω_3 there is no valid data available after $t > 6$ s. However, similar behavior as for other spin rates is assumed. With the higher initial spin rate the sloshing amplitude became lower, due to the higher equivalent gravity level caused by centrifugal forces.

Figure 4.17 shows the simulations results in circumferential direction, the CoM angle variation is caused by liquid sloshing. Due to the instabilities at higher spin rates, the simulation case at the spin rate $\omega_3 = 3.2636$ has a valid data only until $t < 6$ s. However, the behavior is predicted to be the same as for other spin rates, but with a smaller amplitude and faster steady state solution.

The amplitude of the sloshing decrease with an increasing spin rate. The time, in which the maximal amplitude is reached decreases with the increasing spin rate. Higher spin rates lead to higher equivalent g-level acting on the water. This leads to the sloshing damping in a circumferential direction with an increased spin velocity.

The results of the sloshing in longitudinal direction are presented in Figure 4.18. As expected, with the higher spin rate the sloshing amplitude became smaller. For the spin rates ω_2 and ω_3 the sloshing can be completely neglected, because of the small amplitude. For the spin rate ω_1 there is a near linear CoM oscillation. However, the order of the amplitude has a range of 10^{-4} , the spin rate sloshing in this direction, caused by acceleration of $a_y = 0.001 \frac{\text{m}}{\text{s}^2}$ can be neglected. In the orbit the order of the disturbances is smaller than $a_y = 0.001 \frac{\text{m}}{\text{s}^2}$.

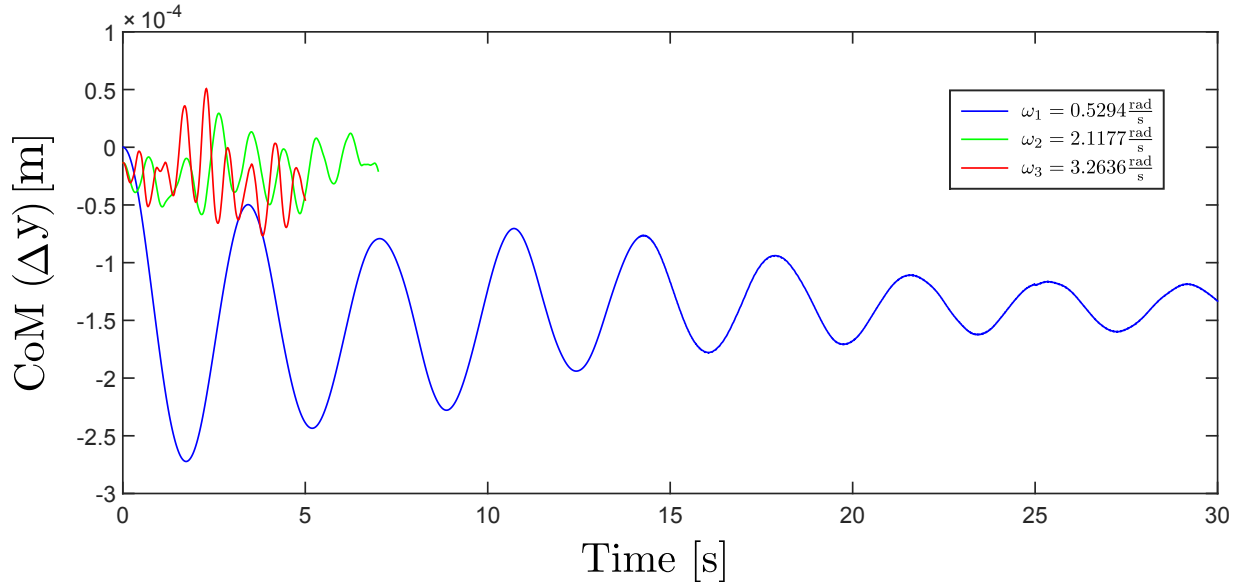


Figure 4.18: Sloshing in a longitudinal direction of the tank with implemented baffles at different spin rates with the step response in acceleration along y-axis of $a_y = 0.001 \frac{\text{m}}{\text{s}^2}$.

Figure 4.19 shows the CoM variation in a radial direction, caused by an abrupt tank acceleration to the initial spin rate. After this acceleration the spin rate of the tank stays constant. This way, for different spin rates, different values of step sizes appear in each case. The graph shows, that the amplitudes stay nearly the same at different spin velocities. Only the time constant, the time after the steady state is reached, became lower with a higher velocity. This behavior is caused by the rising equivalent gravity level within higher spin rates.

In radial case also the CoM shift in a longitudinal direction along y-axis occur. These results are shown in Figure 4.20. In this case linear sloshing occurs, so oscillation is damped with time. With the higher spin rate the time constant became smaller, like in CoM shift in radial direction. This simulation case in a radial direction can be used to identify sloshing in both directions, radial and longitudinal. The advantage of this case, compared to the case in pure longitudinal direction, is that the step in acceleration is large enough to excite sufficient CoM motion along the y-axis.

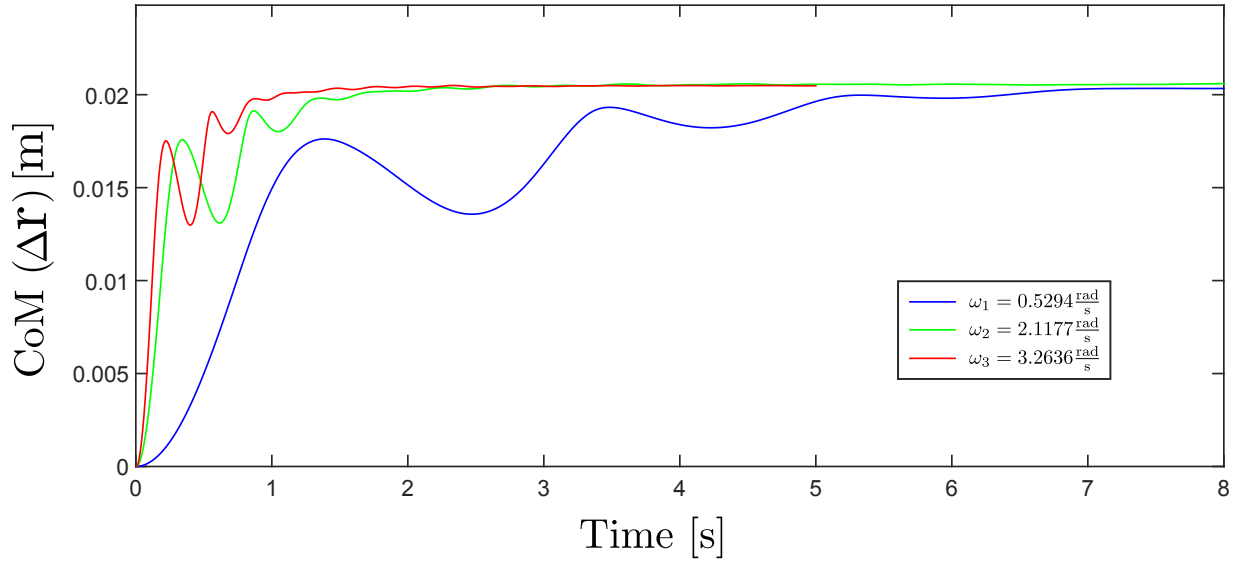


Figure 4.19: Sloshing in a radial direction of the tank with implemented baffles and step response in acceleration. The amplitudes at different spin rates has near the same value, yet the time in which steady state is reached, became smaller with the higher spin rate.

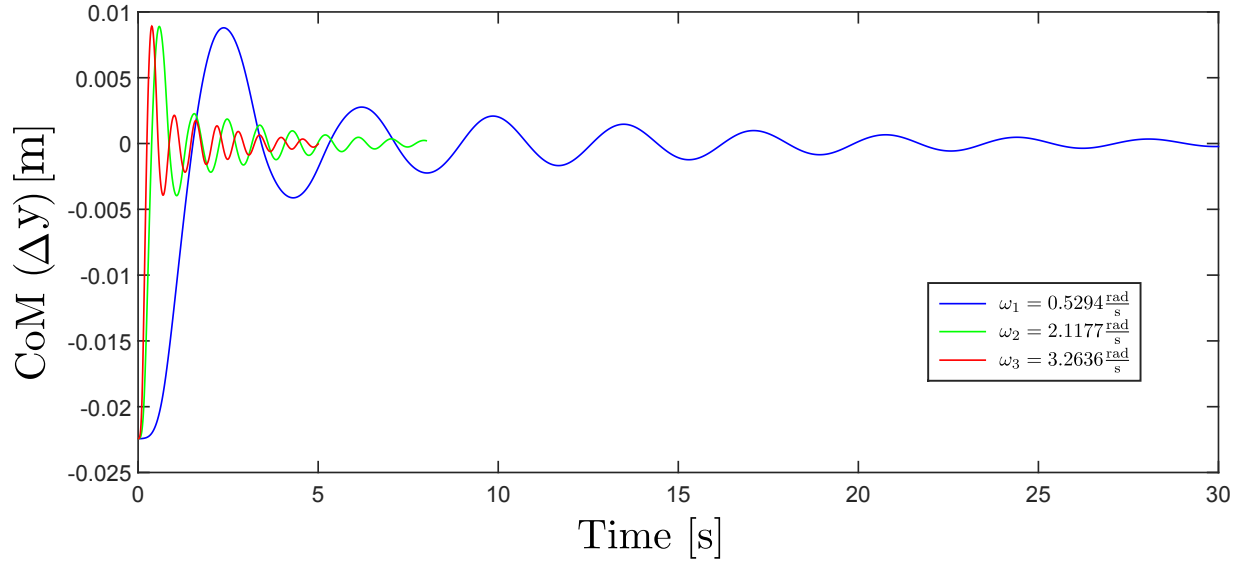


Figure 4.20: Additional CoM motion in a longitudinal direction by applying step response in radial direction. The sloshing behavior correlates to the linear system. With higher spin rates the time constant became smaller, so the oscillation is decayed after a shorter time period.

4.4.4. Transfer Function Determinations

In this section the simulation results with respective CoM motion are used to identify liquid sloshing in relevant directions. Only the transfer functions for the tank with the implemented baffles are generated, since this is the tank configuration implemented in the satellite. As mentioned in Section 3.5 the *MATLAB System Identification Toolbox* was used to generate transfer functions. For each simulation case and for each spin rate, an own transfer function is generated. Afterwards, the relationship between transfer function and the initial spin rate was analyzed, in order to generate transfer functions for spin rates among the simulated data.

To generate transfer functions, the denominator and numerator number has to be chosen. This is an iteration process: first the small number of coefficients is selected in order to keep the model as simply as possible. If the solution is not acceptable the number of coefficients increases until the results are adequate. There are two criteria used to rate the transfer function: the estimated fit to *OpenFOAM* data and the transient step response.

Circumferential direction

In this case the number of poles and zeros was set to 6 for the first estimation, because the tank is divided into 6 sectors trough the baffles. Figure 4.21 show the step response of the estimated transfer function with 6 coefficients (Eq.: 4.12) compared with the OpenFOAM data.

$$TF_{\omega_1}(s) = \frac{0.01335s^6 - 0.1151s^5 + 0.5635s^4 - 1.943s^3}{s^6 + 2.596s^5 + 9.116s^4 + 6.043s^3} \dots \frac{-0.5331s^2 - 0.6785s - 0.0083118}{+2.884s^2 + 2.884s + 0.03585} \dots \quad (4.12)$$

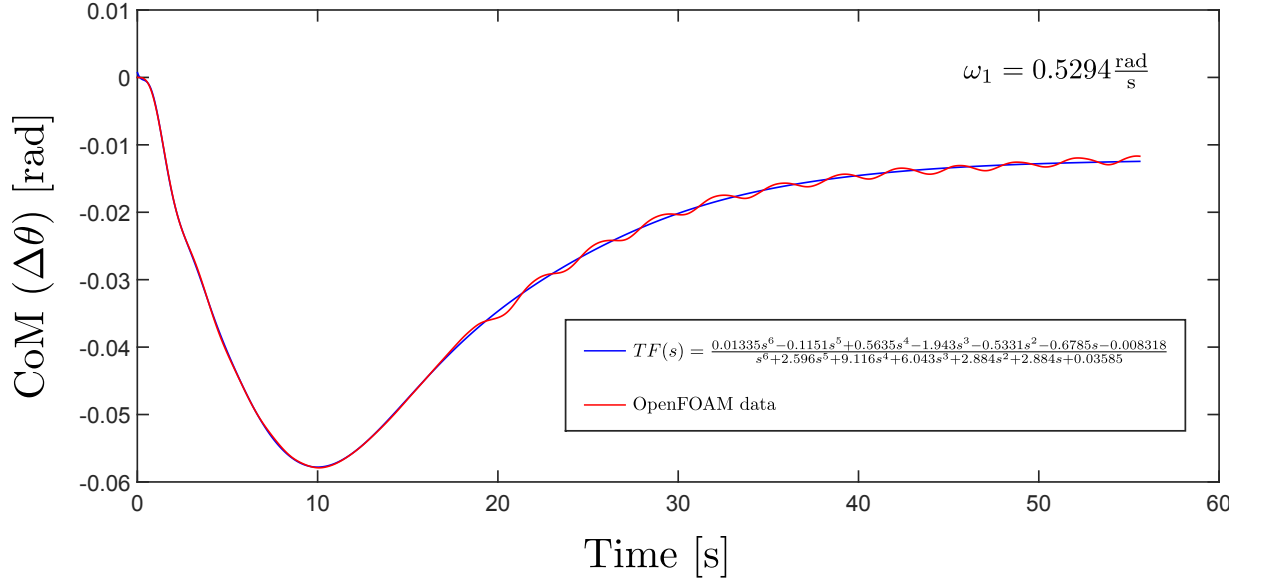


Figure 4.21: Step response of the estimated transfer function compared to OpenFOAM data for initial spin rate of $\omega_1 = 0.5294 \frac{\text{rad}}{\text{s}}$.

The estimated transfer function has acceptable estimation, however the small oscillation after the first amplitude is not simulated. To realize this oscillation, transfer function with higher coefficient numbers is needed. After some iteration steps the optimal solution was found by setting the number of the numerator coefficients to 9 and the number of the denominator coefficients to 15. The following transfer function was estimated:

$$\begin{aligned}
 TF_{\omega_1}(s) = & \frac{-10.29s^9 - 29.06s^8 - 135.9s^7 - 207.7s^6 - 571.7s^5 - 434.4s^4}{s^{15} + 6.174s^{14} + 41.03s^{13} + 140.7s^{12} + 514.5s^{11} + 1147s^{10} \cdots} \\
 & \frac{-828.4s^3 - 220.7s^2 - 196.3s - 2.349}{2807s^9 + 4329s^8 + 7267s^7 + 7784s^6 + 8301s^5 + 5790s^4 \cdots} \\
 & \frac{}{+2928s^3 + 1010s^2 + 175.1s + 10.32}
 \end{aligned} \tag{4.13}$$

Further studies are needed to decide, which transfer function provides satisfactory results. If there are no major differences affecting AOCS, it is useful to implement transfer functions with a lower order of coefficients on order to reduce computational time.

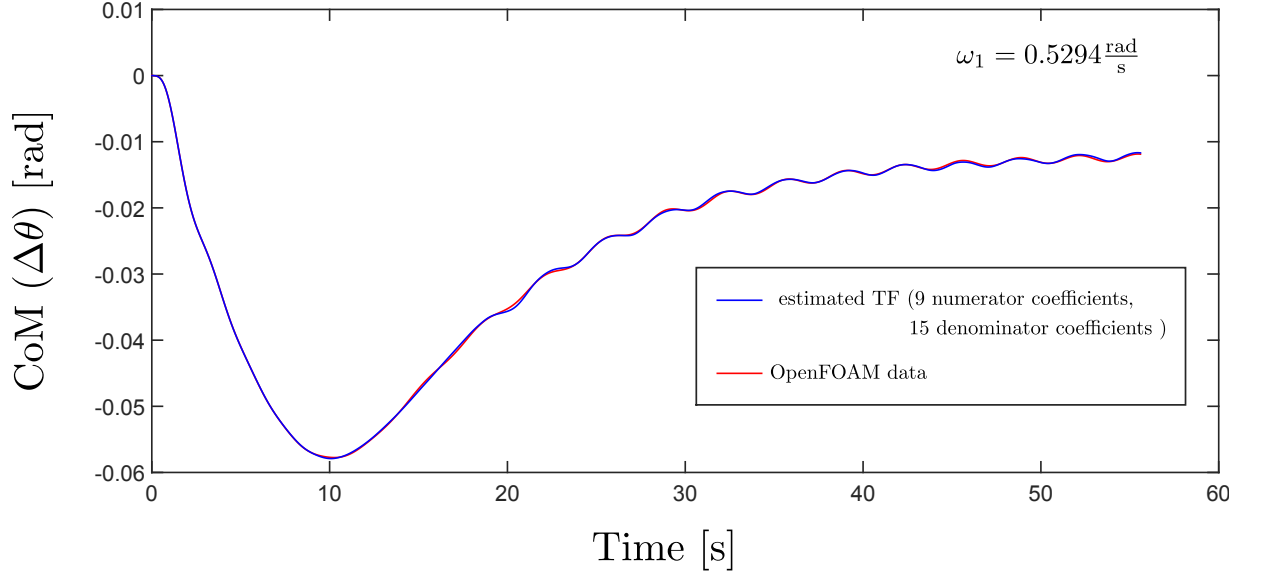


Figure 4.22: Step response of the estimated transfer function compared to OpenFOAM data.

The main difference of the transfer functions, is the value of the first amplitude with the following oscillation and transient response. For every simulation case the same step size in acceleration is used. This leads to the conclusion, that there is no relevant oscillation after the first peak, this happens because it is damped by the centrifugal forces. The oscillation, which occurs at higher spin rates, is expected to be a numerical oscillation of a non constant frequency. For this case, further numerical simulations with higher step sizes in acceleration are needed in order to determine this oscillation.

On this stage it is valid to use transfer functions with lower numbers of coefficients, which do not simulate the oscillations after the first peak in the amplitude. Figure 4.23 shows the comparison between the estimated transfer function and simulated *OpenFOAM* data. The estimated transfer function with 6 coefficients is defined as follows:

$$TF_{\omega_2}(s) = \frac{-0.03148s^6 - 0.1328s^5 - 0.245s^4}{s^6 + 3.985s^5 + 11.15s^4 + 11.65s^3} \cdots \frac{-0.8667s^2 - 0.4604s - 0.01384}{+5.334s^2 + 1.542s + 0.07202} \quad (4.14)$$

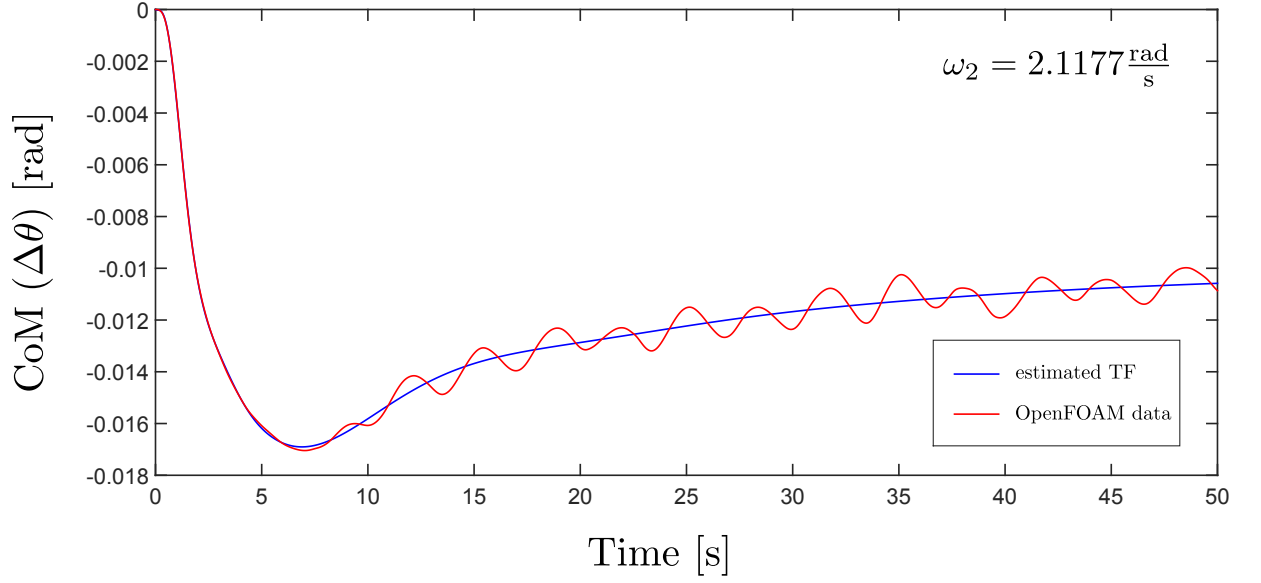


Figure 4.23: Step response of the estimated transfer function compared to OpenFOAM data at spin rate ω_2 . The transfer function has 6 coefficients.

The estimated function for spin rate ω_3 is shown in Figure 4.24. In this figure, difficulties appear by generating a transfer function, because the numerical simulation provides only valid data until 5 s. Here, the same steady state solution as for other spin rates is expected, following transfer function with 6 poles and zeros is generated:

$$TF_{\omega_3}(s) = \frac{-0.0001971s^6 + 0.006711s^5 - 0.131s^4}{s^6 + 7.756s^5 + 47.05s^4 + 146.8s^3} \cdots \frac{-16.01s^2 + 3.378s - 80.62}{+282.3s^2 + 557s + 329.7} \quad (4.15)$$

In order to realize the interpolation between each parameter of the transfer functions further numerical simulations are needed. Additional data will allow to generate transfer functions with equal numbers of the coefficients, this way the interpolation will be possible. Further studies are needed, in order to eliminate the solver stability problems at higher spin velocities.

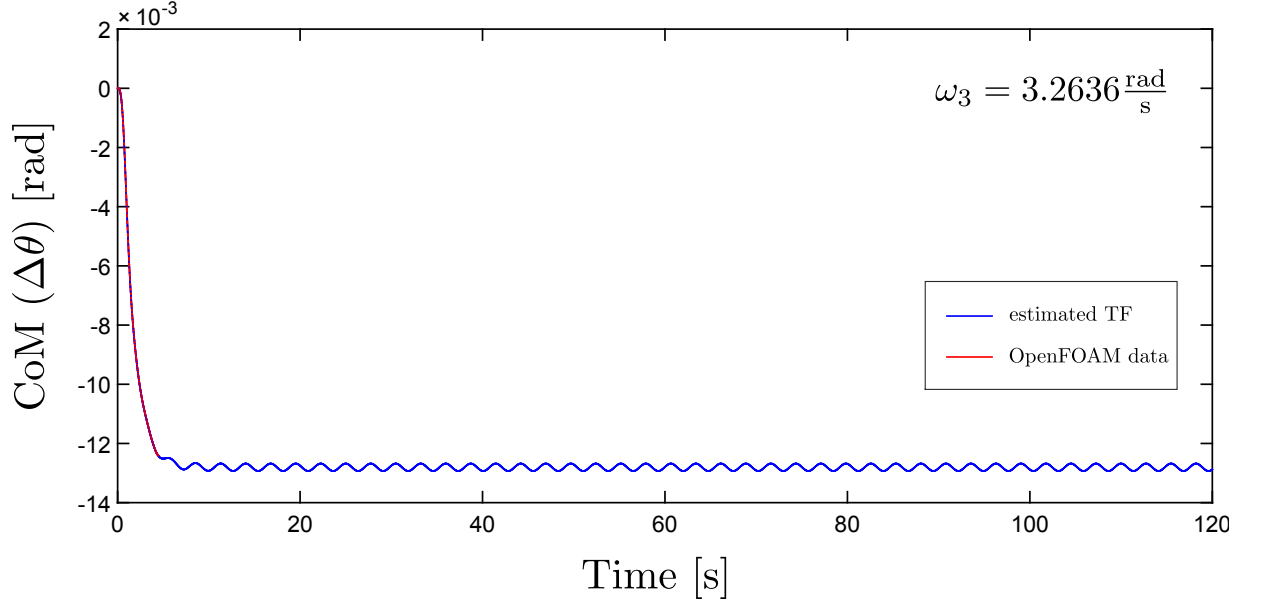


Figure 4.24: Step response of the estimated transfer function compared to OpenFOAM data at spin rate ω_3 .

Radial direction

To identify sloshing in a radial direction the transfer function with 3 coefficients is used. As described in the Section before, the amplitude, the steady state solution and the range of the CoM varies in a radial direction are the same for each spin rate. The only thing changing with the increasing spin velocity, is the time constant, which become lower.

Next transfer functions for each spin are generated in order to identify liquid sloshing behavior:

$$TF_{\omega_1}(s) = \frac{0.00054s^3 - 0.005023s^2 + 0.06326s + 0.0737}{s^3 + 1.577s^2 + 8.568s + 3.566} \quad (4.16)$$

$$TF_{\omega_2}(s) = \frac{0.0003167s^3 - 0.01292s^2 + 1.048s + 4.521}{s^3 + 6.088s^2 + 136.4s + 219.7} \quad (4.17)$$

$$TF_{\omega_3}(s) = \frac{0.0002122s^3 - 0.01187s^2 + 2.521s + 16.55}{s^3 + 9.299s^2 + 324.2s + 807.2} \quad (4.18)$$

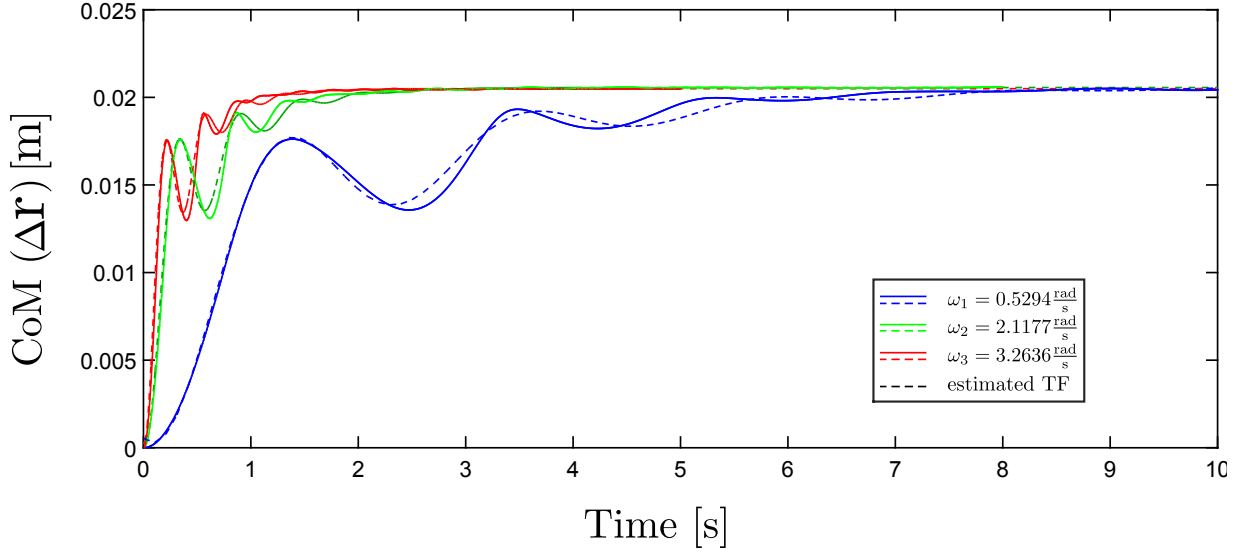


Figure 4.25: Step response in radial direction of the estimated transfer function compared to OpenFOAM for different spin rates

In order to generate transfer function as th function of the spin velocity, interpolation was used. For the numerator the fallowing exponential function was used:

$$f(\omega) = ae^{-b\omega} + c, \quad (4.19)$$

where a , b and c are the parameters generated with th curve fitting tool, and ω is the spin velocity. For the denominator the power function was used:

$$f(\omega) = \alpha\omega^\beta, \quad (4.20)$$

where α , β are parameters of the function fitted to the simulated data and ω is the spin velocity. Following transfer function depending on the spin velocity was generated. The parameter are shown in the following Table 4.8.

$$TF(\omega, s) = \frac{(a_1e^{-b_1\omega})s^3 + (a_2e^{-b_2\omega} + c_2)s^2 + (a_3e^{-b_3\omega})s + a_4e^{-b_4\omega}}{\alpha_1s^3 + (\alpha_2\omega^{\beta_2})s^2 + (\alpha_3\omega^{\beta_3})s^1 + \alpha_4\omega^{\beta_4}} \quad (4.21)$$

Longitudinal direction

The sloshing oscillation caused by the step in acceleration of $0.001 \frac{m}{s^2}$ along the y-axis is too small to identify the transfer functions. However, the results of CoM motion along the

4. CFD-Simulation implementations

Numerator	Value	Denominator	Value
a_1	0.0006387	α_1	1
a_2	0.01229	α_2	2.927
a_3	0.1504	α_3	30.35
a_4	0.3557	α_4	22.99
b_1	0.3122	β_1	0
b_2	0.4612	β_2	0.9771
b_3	-0.8666	β_3	2.002
b_4	-1.177	β_4	3.008
c_2	-0.01463		

Table 4.8: Parameter of the transfer function depending on the spin velocity.

y-axis in a radial case provide noticeable oscillation in this direction, this data is therefore used to generate transfer functions. The results are presented in Figure 4.26, where dashed lines are estimated transfer functions. Following transfer functions are generated:

$$TF_{\omega_1}(s) = \frac{-0.02073s^4 - 0.04774s^3 - 0.07579s^2 - 0.1208s + 8.79 \cdot 10^{-5}}{s^4 + 1.465s^3 + 5.855s^2 + 4.261s + 7.892} \quad (4.22)$$

$$TF_{\omega_2}(s) = \frac{-0.0211s^4 - 0.1769s^3 - 1.166s^2 - 7.292s - 0.04518}{s^4 + 5.372s^3 + 93.1s^2 + 259.7s + 2033} \quad (4.23)$$

$$TF_{\omega_3}(s) = \frac{-0.02142s^4 - 0.2601s^3 - 2.726s^2 - 24.69s - 0.4392}{s^4 + 8.095s^3 + 216.9s^2 + 883s + 11030} \quad (4.24)$$

As described in previous case, the transfer function over the spin rate is generated. The following function is used to fit the coefficients of the transfer function:

$$f(\omega) = a\omega^b + c, \quad (4.25)$$

where a , b and c are the coefficients of the fitted curve and ω is the spin velocity. The transfer function over the spin rate is defined as follows with the coefficient shown in Table 4.9:

$$TF(\omega, s) = \frac{(a_1\omega^{b_1} + c_1)s^4 + (a_2\omega^{b_2})s^3 + (a_3\omega^{b_3})s^2 + (a_4\omega^{b_4})s + a_5\omega^{b_5}}{\alpha_1s^4 + (\alpha_2\omega^{\beta_2})s^3 + (\alpha_3\omega^{\beta_3})s^2 + (\alpha_4\omega^{\beta_4})s + \alpha_5\omega^{\beta_5}} \quad (4.26)$$

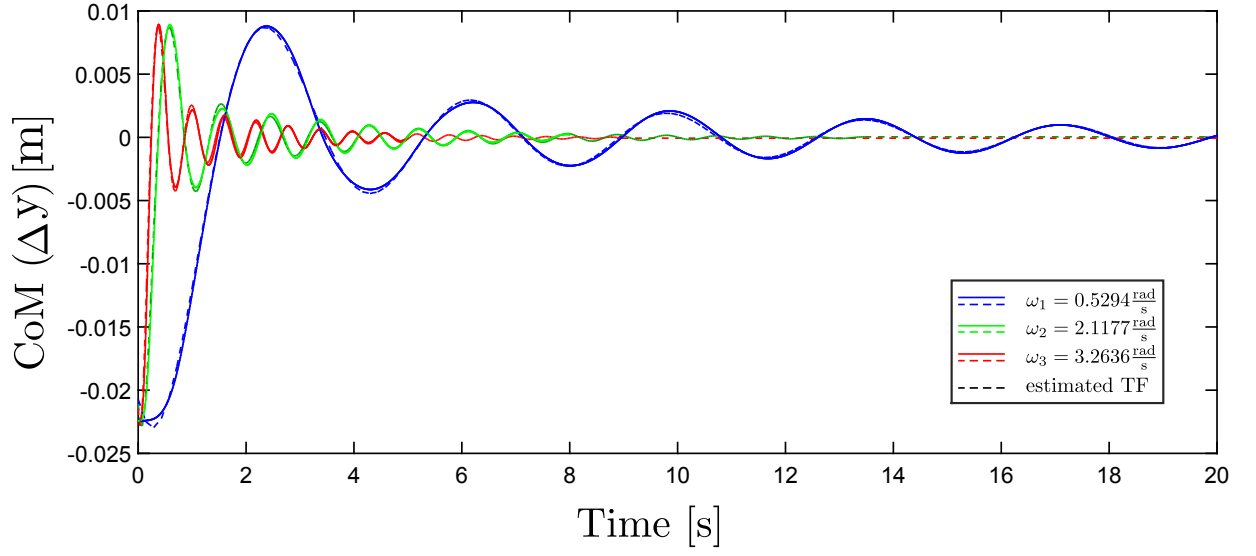


Figure 4.26: Step response along y-axis of the estimated transfer function compared to OpenFOAM data for different spin rates.

Numerator	Value	Denominator	Value
a_1	0.0002251	α_1	1
a_2	0.08797	α_2	2.653
a_3	-0.2671	α_3	21.4
a_4	-0.8782	α_4	27.92
a_5	-0.000875	α_5	108.1
b_1	1.106	β_1	0
b_2	0.9191	β_2	0.9427
b_3	1.964	β_3	1.958
b_4	2.82	β_4	2.92
b_5	5.257	β_5	3.91
c_1	-0.02059		

Table 4.9: Parameter of the transfer function depending on the spin velocity.

5. Conclusion

Liquid sloshing has an paramount influence on the dynamic stability of a spacecraft, as in majority cases there is liquid, e.g. propellant on board. It is therefore very important to identify liquid sloshing behavior and its effect on the spacecraft dynamics to guarantee the stability of the whole dynamical system.

In this thesis, liquid sloshing behavior, related to the EU:CROPIS space mission realized by German Aerospace Center (DLR) was analyzed, since there are two tanks on board of the spacecraft with 1.5 l water respectively, which is used to grow plants under different gravity levels. This is realized by spinning the spacecraft at different velocities. Computational Fluid Dynamics (CFD) were used to simulate the liquid sloshing behavior at different operational modes, spin rates were therefore varied, but also other relevant boundary conditions were implemented in order to initiate space conditions.

The first part of this work provides theoretical knowledge of all relevant topics, like liquid sloshing, system identification and CFD. The main part of the thesis investigates CFD-Simulation implementation. This part includes the determination of the relevant sloshing direction, which are circumferential, radial and longitudinal. Different simulation cases were implemented depending on the sloshing direction. *OpenFOAM* software was used to achieve the simulations. In this work a relevant description is given, on how the simulations were implemented. All significant CFD processes, like geometry and mesh generation, but also solver settings and run controls were described in this work.

The most important part of the thesis are the results discussion. Here, simulations data was post-processed, in order to extract pure CoM motion of the liquid. Grid convergence study was realized for the purpose of discretization error determination. The post-processed CoM data for each simulation case at different spin rates was an input for the transfer function determination processes. The results were transfer functions for each relevant spin velocity in each sloshing direction. For the sloshing in radial and longitudinal direction the transfer function was determined, depending on the spin rate, using the interpolation between each parameter of the transfer function. Further studies are needed to achieve the same interpolation for the circumferential direction.

5. Conclusion

In this thesis, the expected damping effect of the baffles in the circumferential direction was proven. In addition, it was demonstrated, that baffles in other directions are not necessary. Liquid sloshing is assumed to have a positive effect on the AOCS of the Eu:CROPIS spacecraft.

6. Outlook

The output of this thesis are the transfer functions, which represents the sloshing behavior of the water tank implemented in EU:CROPIS payload. In order to analyze the sloshing effect in the spacecraft dynamics, the transfer functions have to be implemented into EU:CROPIS AOCS simulation. In the first stage, it is sufficient to implement the transfer functions into a nutation simulation, in order to demonstrate the damping effect of the liquid sloshing relating to the nutation angle, since this is the most relevant case.

Numerical problems have been identified during the simulation implementation with *OpenFOAM*. Further studies are needed to solve this issue. One option is to decrease tolerance by solving the equations and to increase the number of iterations. Another option is to test various solvers. After this problem is solved, the data over a large time interval and higher spin rates can be generated. This will allow to achieve the transfer function interpolation for all sloshing directions.

The implemented grid convergence study showed, that the discretization error has a range of $\approx 15\%$. This error has to be implemented in further simulations. Also, numerical simulations with finer grids can be made to decrease this error.

To validate numerical simulation results experimental verification will be needed. To generate comparable data new simulation cases will be necessary, due to the gravity conditions on earth. At the, the experimental and numerical results have to be compared.

References

- [1] Deutsches Zentrum für Luft- und Raumfahrt. *Eu:CROPIS – Greenhouses for the Moon and Mars*. 2016. URL: http://www.dlr.de/dlr/en/desktopdefault.aspx/tabid-10081/151_read-17874/#/gallery/23027 (visited on 05/15/2016).
- [2] Friedrich-Alexander Universität Erlangen-Nürnberg. *Tomatoes in space*. 2014. URL: <https://www.fau.eu/2014/05/26/news/tomatoes-in-space/> (visited on 05/15/2016).
- [3] R. A. Ibrahim. *Liquid Sloshing Dynamics: Theory and Applications*. 1. ed. New York, United States of America: Cambridge University Press, 2005.
- [4] H. Lutz, W. Wendt. *Taschenbuch der Regelungstechnik*. 10. ed. Frankfurt am Main, Deutschland: Europa-Lehrmittel, 2014.
- [5] Deutsches Zentrum für Luft- und Raumfahrt. 2016. URL: <http://www.dlr.de> (visited on 05/01/2016).
- [6] Friedrich-Alexander Universität Erlangen-Nürnberg. 2016. URL: <https://www.fau.eu/> (visited on 05/15/2016).
- [7] CFD Direkt. *OpenFOAM: The Open Source CFD Toolbox, User Guide*. 2015. URL: <http://cfd.direct/>.
- [8] D. Halliday, R. Resnick, J. Walker. *Fundamentals of Physics*. 8. ed. Hoboken, NJ: Wiley, 2008.
- [9] N. Fries, P. Behruzi, T. Arndt, M. Winter, G. Netter, U. Renner. “Modelling of fluid motion in spacecraft propellant tanks - Sloshing”. In: *Space Propulsion Conference, Bordeaux* (2012).
- [10] Michael F. Zedd, Franklin T. Dodge. “Energy Dissipation of Liquids in Nutating Spherical Tanks Measured by a Forces Motion-Spin Table”. In: *Naval Research Laboratory, Washington, D.C.* (1985).
- [11] Ch. Ament. *Skript zur Vorlesung Systemidentifikation bzw. Prozessanalyse 2*. Technische Universität Ilmenau. Ilmenau, SoSe 2014.

- [12] F. T. Dodge. *THE NEW "Dynamic Behavior of Liquids in Moving Containers"*. San Antonio, Texas: Southwest Research Institute, 2000.
- [13] Jiyuan Tu, Guan Heng Yeoh, Chaoqun Liu. *Computational Fluid Dynamics: A Practical Approach*. 1. ed. München, Deutschland: Elsevier, 2008.
- [14] J. Blazek. *Computational Fluid Dynamics: Principles and Applications*. 2. ed. München, Deutschland: Elsevier, 2005.
- [15] John W. Slater. *Uncertainty and Error in CFD Simulations*. URL: <http://www.grc.nasa.gov/WWW/wind/valid/tutorial/errors.html> (visited on 06/01/2016).
- [16] C. W. Hirt, B. D. Nichols. "Volume of Fluid (VoF) Method for the Dynamics of Free Boundaries". In: *Journal of Computational Physics*, 1981 (1981).
- [17] ANSYS, Inc. *ANSYS Meshing: Application Introduction*. 2009.
- [18] J. Guerrero. *Tips and tricks in OpenFOAM*. University of Genova, DICCA. Genova, Italy, 2015.
- [19] John W. Slater. *Examining Spatial (Grid) Convergence*. URL: <http://www.grc.nasa.gov/WWW/wind/valid/tutorial/errors.html> (visited on 06/01/2016).

A. ControlDict

```

/*-----*- C++ -*-----*\
| =====|
| \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O p e r a t i o n | Version: 2.3.1 |
| \ \ / A n d | Web: www.OpenFOAM.org |
| \ \ / M a n i p u l a t i o n |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    location     "system";
    object       controlDict;
}
// * * * * *

application      interDyMFoam;

startFrom        startTime;

startTime        0;

stopAt           endTime;

endTime          100;

deltaT           1e-4;

writeControl      adjustableRunTime;

writeInterval     0.5;

purgeWrite       0;

writeFormat       ascii;

writePrecision    6;

writeCompression compressed;

timeFormat        general;

timePrecision     6;

runTimeModifiable yes;

adjustTimeStep    yes;

maxCo             0.25;
maxAlphaCo        0.25;
maxDeltaT         1;

```

A. ControlDict

```
functions
{
    forces
    {
        type forces;
        functionObjectLibs ( "libforces.so" );
        patches (walls);
        rhoName rho;
        rhoInf      ;
        UName      U;
        log         on;
        outputControl    timeStep;
        outputInterval  1;
        CofR (0 0 0);
    }

    CoM
    {
        type          CoM;
        functionObjectLibs ("libCoM.so");
        outputControl    adjustableTime;
        writeInterval    0.01;
        outputInterval  1;
    }
}

// ***** //
```


B. fvSchemes

```

/*----- C++ -----*/
| ===== |
| \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O p e r a t i o n | Version: 2.3.1 |
| \ \ / A n d | Web: www.OpenFOAM.org |
| \ \ / M a n i p u l a t i o n |
/*-----*/

FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    location     "system";
    object       fvSchemes;
}
// *****

ddtSchemes
{
    default      Euler;
}

gradSchemes
{
    default      Gauss linear;
}

divSchemes
{
    div(rhoPhi,U)  Gauss vanLeerV;
    div(phi,alpha) Gauss vanLeer;
    div(phirb,alpha) Gauss vanLeer;
    div((muEff*dev(T(grad(U)))) Gauss linear;
}

laplacianSchemes
{
    default      Gauss linear corrected;
}

interpolationSchemes
{
    default      linear;
}

snGradSchemes
{
    default      corrected;
}

fluxRequired
{
    default      no;
    p_rgh;
    pcorr;
}

// *****

```

C. fvSolution

```
/*----- C++ -----*\
|=====|
| \\  /  | F i e l d      | OpenFOAM: The Open Source CFD Toolbox
| \\  /  | O peration    | Version:  2.3.1
| \\  /  | A nd          | Web:      www.OpenFOAM.org
| \\  /  | M anipulation  |
|=====|\
\*-----*/

FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    location     "system";
    object       fvSolution;
}
// *****

solvers
{
    alpha.water
    {
        nAlphaCorr      1;
        nAlphaSubCycles 3;
        cAlpha          1.5;
    }

    pcorr
    {
        solver          PCG;
        preconditioner
        {
            preconditioner  GAMG;
            tolerance       2e-09;
            relTol          0;
            smoother        DICGaussSeidel;
            nPreSweeps      2;
            nPostSweeps     2;
            nFinestSweeps   2;
            cacheAgglomeration true;
            nCellsInCoarsestLevel 10;
            agglomerator     faceAreaPair;
            mergeLevels      1;
        }

        tolerance       2e-09;
        relTol          0;
        maxIter          150;
        minIter          50;
    }

    pcorrFinal
    {
        solver          PCG;
        preconditioner
        {
            preconditioner  GAMG;
            tolerance       2e-09;
            relTol          0;
            smoother        DICGaussSeidel;
            nPreSweeps      2;
            nPostSweeps     2;
            nFinestSweeps   2;
            cacheAgglomeration true;
        }
    }
}
```

C. fvSolution

```
        nCellsInCoarsestLevel 10;
        agglomerator    faceAreaPair;
        mergeLevels     1;
    }

    tolerance      2e-09;
    relTol         0;
    maxIter        150;
    minIter        50;
}

p_rgh
{
    solver          PCG;
    preconditioner
    {
        preconditioner  GAMG;
        tolerance      2e-09;
        relTol         0;
        nVcycles       2;
        smoother       DICGaussSeidel;
        nPreSweeps     2;
        nPostSweeps    2;
        nFinestSweeps  2;
        cacheAgglomeration true;
        nCellsInCoarsestLevel 10;
        agglomerator    faceAreaPair;
        mergeLevels     1;
    }

    tolerance      2e-09;
    relTol         0;
    maxIter        150;
    minIter        50;
}

p_rghFinal
{
    solver          PCG;
    preconditioner
    {
        preconditioner  GAMG;
        tolerance      2e-09;
        relTol         0;
        nVcycles       2;
        smoother       DICGaussSeidel;
        nPreSweeps     2;
        nPostSweeps    2;
        nFinestSweeps  2;
        cacheAgglomeration true;
        nCellsInCoarsestLevel 10;
        agglomerator    faceAreaPair;
        mergeLevels     1;
    }

    tolerance      2e-09;
    relTol         0;
    maxIter        150;
    minIter        50;
}
```

C. fvSolution

```
U
{
    solver          PBiCG;
    preconditioner   DILU;
    tolerance        1e-09;
    relTol           0;
    minIter          100;
    maxIter          200;
}

UFinal
{
    solver          PBiCG;
    preconditioner   DILU;
    tolerance        1e-09;
    relTol           0;
    minIter          100;
    maxIter          200;
}

PIMPLE
{
    momentumPredictor yes;
    nCorrectors        5;
    nNonOrthogonalCorrectors 0;
    correctPhi          yes;
    pRefCell 0;
    pRefValue          1e5;
}

relaxationFactors
{
    fields
    {
    }
    equations
    {
        "U.*"          1;
    }
}

// ***** //
```