AI-CFD Based Analysis for Propellant Management in Cryogenic Tanks

Y Le Guennec[1], **Y Blacodon**[2], **S Tollis**[2], **T Defoort**[1], **J.V. Aguado**[1,3], **D. Borziachello**[1,3]

[1] : Miura Simulation Le Village By CA – 140 Route de Paris, 44949 Nantes Cedex 9. France
[2] : MaiaSpace – Batiment A37 Arianegroup foret de Vernon, 27200 Vernon. France
[3] : Institut de Recherche en Génie Civil et Mécanique, Ecole Centrale de Nantes – 1 rue de la Noë 44000 Nantes. France

Keywords: Computational Fluid Dynamics, Sloshing simulation, Neural fields, Fourier features

Abstract. Sloshing simulations play a crucial role in the design of cryogenic tanks. In particular, analyzing the evolution of the center of mass helps mitigate damaging sloshing phenomena. Additionally, other key quantities, such as pressure distribution on the tank walls and fluid velocity, can be extracted from the simulation results. This study focuses on transient diphasic fluid dynamics involving both gas and propellant. These simulations are computationally expensive and often constitute a bottleneck in the design process of new launchers. Consequently, the ability to perform such simulations in a reduced time frame would be a significant asset, accelerating the design phase. Faster simulations would also enable efficient parametric exploration, facilitating the identification of innovative design solutions. In this work, Neural Fields Networks are employed as a surrogate model to accelerate sloshing simulations. These networks are trained on data generated from conventional high-fidelity simulations. This preliminary study aims to demonstrate the potential of the proposed approach and explore different network architectures to optimize performance.

1. Introduction

The advancement of space exploration technology has brought forward numerous challenges in ensuring the safety and efficiency of space launch vehicles. Among these challenges, the dynamics of cryogenic fuel sloshing inside tanks during rocket launches remain a significant concern. Sloshing can result in destabilizing forces that affect the launch vehicle's trajectory and structural integrity. To mitigate these risks, accurate simulation of sloshing behaviour in cryogenic tanks is mandatory.

Computational Fluid Dynamics (CFD) enables the detailed modelling of fluid flow within the cryogenic tanks, where the fluid dynamics are influenced by the interaction of the liquid with the container's walls and the external forces during launch. The goal is to understand how the liquid behaves under conditions such as acceleration, vibration, and the effects of varying gravitational forces, all of which contribute to sloshing phenomena.

In CFD simulations, the fluid flow is governed by the Navier-Stokes equations [1]. In the case of sloshing analysis, hydrodynamics modelling is required. Volume of Fluid (VOF) method is well adapted to capture free surface of the liquid which is essential. In the VOF approach, the fluid domain is represented by a single set of equations, with an additional equation for the volume fraction $\boldsymbol{\alpha}$, which describes the faction of the cell occupied by the fluid, where $\boldsymbol{\alpha}$ varies between 0 (gas) and 1 (liquid). Figure 1 outlines an illustration of discretization of α on a cell mesh.

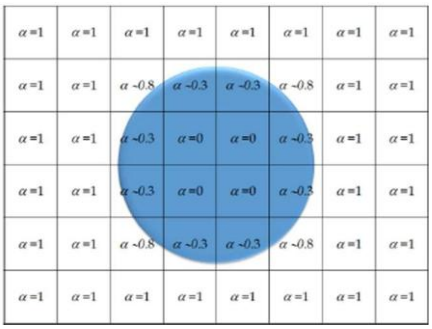

Figure 1 – Volume of Fluid modelling where $\boldsymbol{\alpha}$ is the volume fraction of the fluid.

One of the main challenges in simulating sloshing in cryogenic tanks using CFD and VOF methods is the long simulation times required. The fluid dynamics in these scenarios can be highly complex, with sloshing waves interacting with structural boundaries and fluctuating over time due to varying accelerations during launch. These simulations often require small time steps to capture the full range of fluid behaviour, from the initial sloshing motion to the damping of the waves as the liquid settles.
Moreover, high-fidelity simulations of cryogenic sloshing often require fine spatial resolution to accurately model the interactions between the fluid and the tank walls. This results in many computational cells and a significant amount of computational power. For realistic simulations, it is not uncommon for these studies to take weeks or even months to complete, particularly when performing parametric studies or optimizing the design of the tank for different launch scenarios.
Figure 2 illustrates physical phenomena that occurs during fluid sloshing in tank [2]. Nonlinear flow is considered when the filling level is close to the critical depth and where the fluid behaves as a stationary wave. Inner fluid displacement generates pressure gradient and internal momentum, and close wall fluid induces damping force. At the free surface level, surface tension occurs.
To simplify the complexity of sloshing dynamics reduced model as mechanical equivalent model can be used when fluid displacement can be associated to linear displacement. Figure 3 illustrates different approaches as spring-mass or pendulum. [3], describes the different approaches used for linear sloshing.

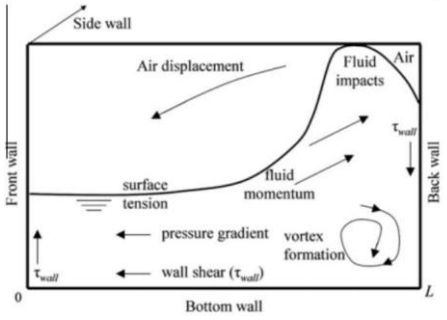

Figure 2 – Fluid forces due to sloshing phenomenon, [2]

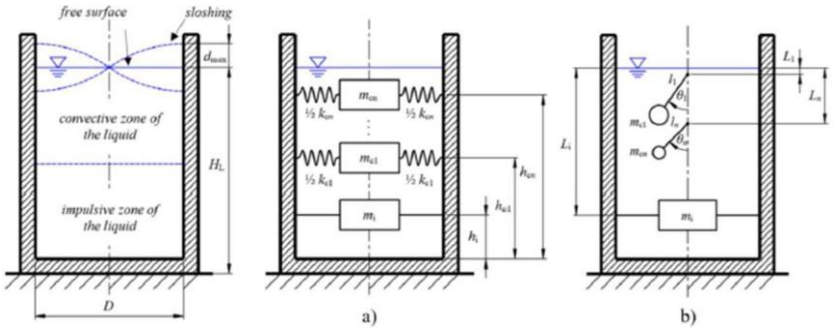

Figure 3 – Principle of mechanical modelling of sloshing (a) spring-mass (b) pendulum

When violent motion occurs, especially during stages separation, linear fluid displacement does not occur anymore. At this point, only complex CFD simulation can be used to assess fluid behaviour.
Thus, the use of computationally expensive simulations in industrial design processes presents significant challenges. These processes often require numerous queries of the computationally expensive high-fidelity (HF) models, involve handling large volumes of data, and demand highly specialized post-processing. A common approach to mitigate these challenges is to replace HF models with surrogate models that can rapidly approximate simulation responses for new parametric configurations. Such surrogate models include response surfaces [4], reduced-order models (ROMs) [5], and, more recently, neural networks [6].
Traditionally, response surfaces focus on modelling specific scalar quantities, whereas ROMs aim to predict entire fields over a computational mesh and across multiple time steps. This capability makes ROMs useful for a wide range of applications, including optimization and sensitivity analysis. However, a key limitation of ROMs is their reliance on a fixed mesh and time discretization, restricting their adaptability to varying geometric configurations and dynamic parameters.
To address this limitation, researchers are exploring the potential of neural networks to replace HF simulations while overcoming these constraints. Neural Field Networks [7] and Neural Radiance Fields (NeRFs) [8], offer a promising alternative, as they can take as input different meshes, geometries, or time-varying parameters, enhancing their flexibility and generalization capabilities.
In this preliminary study, we investigate the feasibility of using Neural Fields to replace sloshing simulations and perform specific post-processing tasks. The article first provides an overview of the sloshing simulations conducted using Converge CFD software. Next, we outline the workflow for integrating simulation data into an interactive plugin. Finally, we present initial application results and key insights derived from this approach.

2. The Converge CFD simulation

CFD simulation is chosen as simple as possible but complex enough to be representative of complex sloshing motion met in cryogenic tanks. To faster the data generation the geometry used is a simple cylinder filled with liquid methane which is the propellant used in recent developed rocket engines. Prometheus (see Figure 4), is one of them and is the selected engine that will be used to propel MaiaSpace launcher.

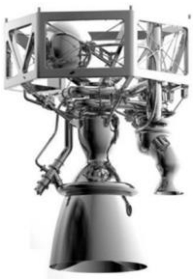

Figure 4 – Prometheus engine preliminary 3D model

Figure 5 describes the geometries used and filling levels. 2 m height cylinders are used with different diameters i.e 0.5 m and 0.75 m. 3 different filling levels are also used to generate a sufficient database. A focus is also done on a longitudinal acceleration representative of upper stage separation when a pusher technology is used. The acceleration and deceleration are represented with a triangular pulse.

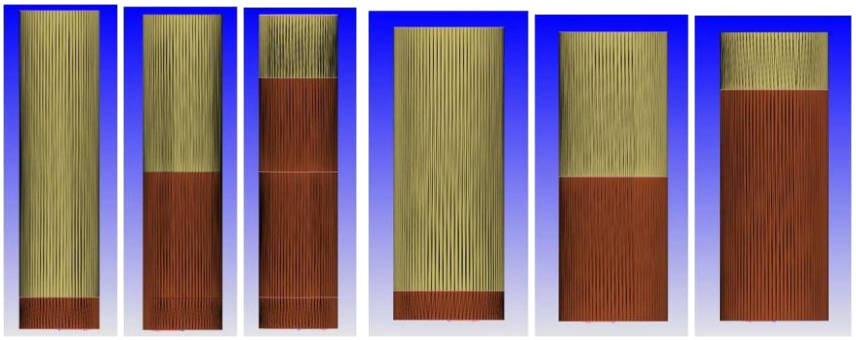

Figure 5 – left : 0.5 m diameter tank – Filling levels (10%, 50%, 80%)
Right : 0.75 m diameter tank – Filling levels (10%, 50%, 80%)

By varying all parameters (diameter, filling level, ascent slope and descendant slope) a Design of Experiment (DOE) of 54 simulations is ran to be used to build a model.

Figure 6 shows **the state of liquid methane during simulation for the 0.25 m diameter cylinder filled at 10% and using the dynamic with ascent slope equal to 50 and descendant slope equal to 50.**

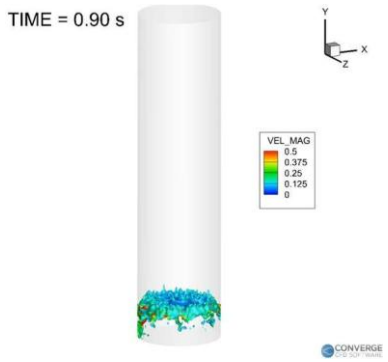

Figure 6 – Liquid methane behaviour at 0.9 s of simulation for 0.25 m diameter cylinder filled at 10% using dynamic with ascent slope equal to 50 and descendant slope equal to 50.

3. From data to ML models: workflow explanation

The proposed workflow, illustrated in Figure 7, outlines the process of transforming simulation data into a plugin suitable for interactive use.

Figure 7 - the implemented workflow to transform simulation data to interactive plugin

It consists of the following key components:

- Data Management Services: These services convert simulation results into the open and standardized XDMF format, ensuring efficient data utilization. The processed data is then used to generate datasets tailored for training a machine learning (ML) model based on predefined input/output specifications.
- Generative Simulation Services: This stage employs the generated datasets to train a dedicated neural network model.
- Model Deployment Services: This component facilitates the export of the trained neural network model as a plugin, enabling interactive use.

At each stage of the workflow, an appropriate computational toolbox is selected to perform the necessary operations. In the case study presented in this article, the litdata toolbox is used for dataset construction and streaming, while the model is trained and optimized using Keras 3. Finally, the trained model is integrated into ParaView to enable interactive visualization and analysis.
The XDMF format facilitates the handling of high-resolution simulation results in terms of mesh and time discretization while eliminating the need for proprietary software for visualization, interaction, and data analysis. A critical step in this process is converting raw simulation data into a machine learning-ready dataset. This transformation plays a pivotal role in both the efficiency of the training phase, leveraging a pipeline-as-code approach, and the accuracy of the model by ensuring the data is optimally transformed for learning. A simple example of such transformation is the normalization of the outputs.

4. The Neural field model

The neural network architecture employed in this study is a Neural Field. As illustrated in Figure 8, this architecture is a fully connected network with 5 layers having 128 units for the first two and 64 units for the third last. It takes as input the mesh point coordinates, time instants, and key simulation parameters, including the cylinder diameter, filling height, and acceleration/deceleration levels. The network outputs volume fraction α, where $\alpha = 0$ represents the fluid phase and $\alpha = 1$ represents the gas phase.

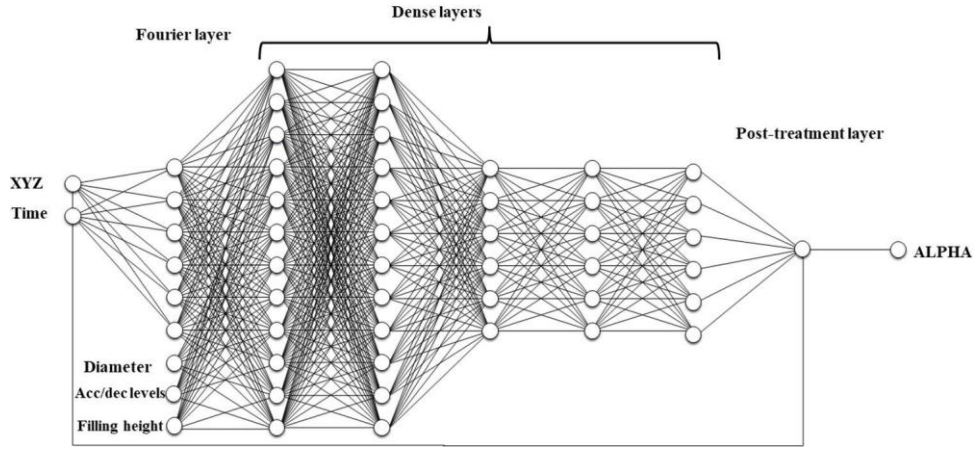

Figure 8 - Schema of the Neural Field network architecture.

On Figure 6, the small-scale fluctuations of the α variable are visible on the free surface between liquid and gas. To consider these variations, a Fourier features layer is used on the spatial coordinates and time scale [9]**.** It prevents the issue of the spectral bias induced by neural network that filter high frequencies content.
Moreover, the α variable is strongly influenced by its initial state. The sloshing will induce variation around this initial state that is directly related to the filling height of the tank (see Figures 3). Then having an architecture that consider this property will greatly help the convergence rate of the network and its accuracy. To do so, a post-treatment layer is added before the output layer. It consists in the multiplication of the two opposites sigmoid function and the addition of a third to have a smooth transition around the position of the free surface. It allows to have α likely to be close to zero as y coordinates tends toward 0 and equal to 1 as it tends toward 2 (top y coordinates of the tank). In region of interest, the model will take the value 1 or 0 to capture the variation due to sloshing.
The network's weights and biases are optimized to minimize the discrepancy between high-fidelity (HF) simulation results and the neural network predictions at each node and time step. This architecture enables the flexibility to handle meshes that vary dynamically over time and across simulations, as well as different time-step resolutions for each simulation as it is the case in the sloshing simulations.
Given binary nature of the α field, the loss function used for training is binary cross-entropy, and the optimization algorithm employed is RMSProp (Root Mean Square Propagation).

5. First training results

A model with the architecture described in the previous section has been trained using 5 computations with varying diameter, acceleration and deceleration levels. The validation set contains 2 simulations and the test set one. The size of the sets has been limited in a model development view to have short training time. Moreover, the training is done on a local PC and the disc memory print has be limited. Indeed, each computation is 15Gb heavy as it has more than 1,800,000 elements at each of the time step. The mesh is modified at each time step to track the free surface between fluid and gas. The number of time step depends on the value of the parameters and is more than 100. The elements in figure 7 are actually modified to be able to train from remote data.
The figures 11 shows comparisons between inference and ground truth for simulation in the training set at a given times step. It is worth noticing that the model can capture dynamics of the sloshing.

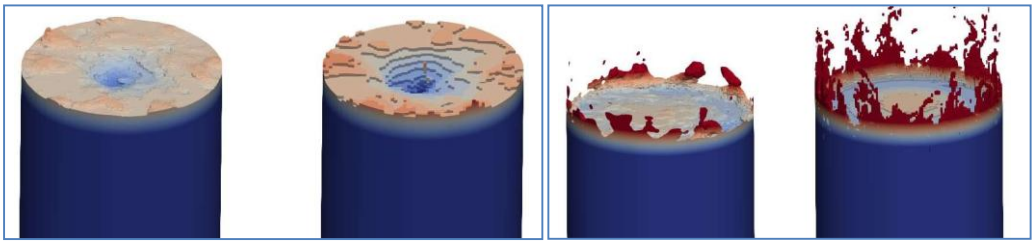

Figure 11 – Comparison between inference (left figures) and ground truth (right figure) for two parametric configurations. The color is the elevation.

The figure 12 highlights the accuracy of the model for parametric configuration outside the training set.

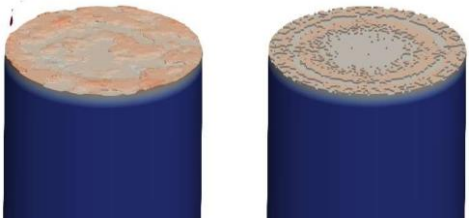

Figure 12 – Comparison between inference (left figures) and ground truth (right figure) for a parametric configuration in the test set. The color is the elevation.

This parametric neural model can be effectively utilized to perform dedicated post-treatment with high efficiency. In Figure 13, the coordinates of the system's center of gravity along the y-axis are shown in the right panel. This comparison between high-fidelity results and inference outcomes highlights the close agreement between the two sets of data.

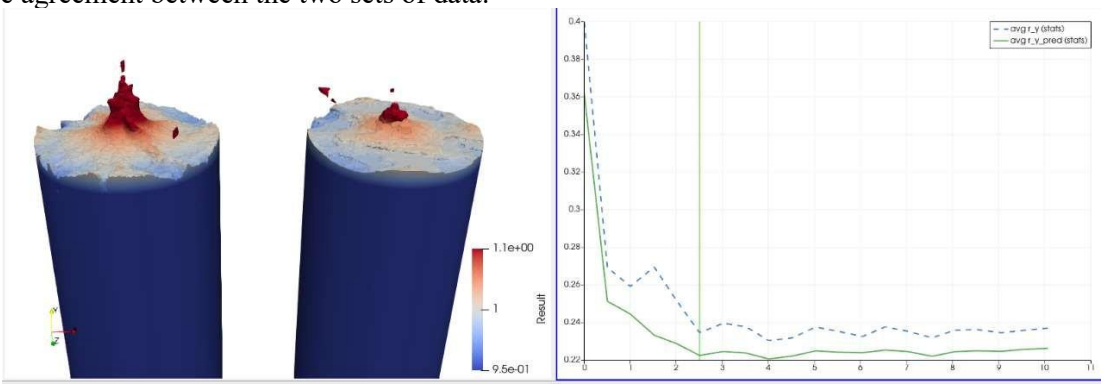

Figure 13 – On the right : high fidelity on the left and inference on the left for a selected instant. On the left, coordinates of the center of mass along the y-axis according to the time, high fidelity in dashed line and inference in solid line..

This parametric neural model also enables the development of a Paraview plugin that allows users to interact with parametric configurations. The plugin automatically refines the mesh around the free surface area and adjusts the cylinder's diameter based on the parameter value, as illustrated in Figure 14.

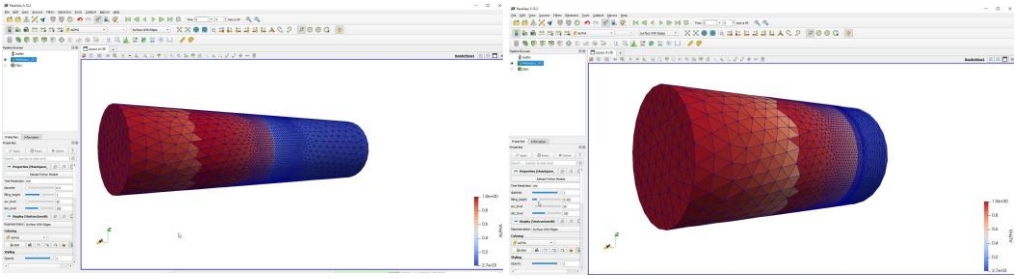

Figure 14 –Visualization of the plugin for two different parametric configurations.

6. Conclusions and outlooks

This study explores the potential of Neural Fields as a replacement for conventional sloshing simulations. Preliminary results are promising showing the relevancy of dedicated layers to take advantages of knowledge of the simulation data structure. Improvements in model architecture—such as the integration of Neural Radiance Fields (NeRF) or Long Short-Term Memory (LSTM) networks—could also enhance performance by accommodating a broader range of parameters and better handling transient simulations. The pipeline-as-code approach implemented in this framework enables rapid adaptation to these enhancements, facilitating iterative model refinement.

One existing limitation is the substantial data volume, which prevents the storage of numerous computations on a local machine. Utilizing an S3 bucket for data storage, combined with an appropriate toolbox, enables efficient streaming of simulation data, facilitating model training on comprehensive datasets. A critical consideration is the environmental impact of both storage and inference. To mitigate the carbon footprint of data storage, a parsimonious approach to data management can optimize data exchanges and the training process. Additionally, model compression techniques, such as pruning or quantization, can reduce model size and computational requirements, thereby minimizing the carbon footprint of inference. Moreover, design of experiments inspired by Sobol sequences [10] could be applied to limit the number of runs to feed the training set.

7. References

[1] G. K. Batchelor, ‘An Introduction to Fluid Dynamics’, Cambridge University Press, 1967

[2] B. Godderidge, S. R. Turnock, M. Tan, ‘A rapid method for the simulation of sloshing using a mathematical model based on the pendulum equation’. Computers & Fluids, vol. 57, p. 163-171. 2012

[3] R. A. Ibrahim, ‘Liquid Sloshing Dynamics,(Theory and Applications)’, Cambridge University Press. 2005

[4] R.H. Myers, D.C. Montgomery, C.M. Anderson-Cook, Response surface methodology: process and product optimization using designed experiments, John Wiley & Sons, 2016

[5] P. Benner, S. Gugercin, K. Willcox, A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems, SIAM Rev. 57 (2015) 483–531. doi:10.1137/130932715

[6] T. Omar, E. Eskandarian, N. Bedewi: Vehicle crash modelling using recurrent neural networks, Mathematical and Computer Modelling, 28, 9, 1998

[7] Y. Xie et al.: Neural Fields in Visual Computing and Beyond, Computer Graphics Forum, 41, 2, 2022

[8] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, Ng, Ren, ‘NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis’, Commun. ACM, 2021

[9] M. Tancik, P. P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. T. Barron, R. Ng, ‘Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains’, 34th Conference on Neural Information Processing Systems, Vancouver, Canada, 2020

[10] F. Noura, S. Marelli, B. Sudret. ‘Sequential design of experiment for sparse polynomial chaos expansions.’ SIAM/ASA Journal on Uncertainty Quantification