POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

A Comprehensive Data-Driven Framework for Robotic Liquid Sloshing Control: From Extensive Experimental Model Validation and Enhancement to Trajectory Optimization

Laurea Magistrale in Mechanical Engineering - Ingegneria Meccanica

Author: Denis Camilotto

Advisor: Prof. Hermes Giberti

Co-advisor: Nicola Giulietti

Academic year: 2025-2026

Introduction

Liquid sloshing represents a productivity bottleneck in high-speed pharmaceutical and food industrial automation, as inertial forces cause undesired spills or contamination. While high-fidelity solvers (SPH, CFD) are too computationally heavy for real-time use, efficient Mass-Spring-Damper (MSD) models often lack the validation required for industrial reliability. This work proposes a data-driven framework for trajectory optimization, utilizing a custom vision-based platform for extensive experimental validation. This data-driven approach is used to enhance a multi-modal MSD model via deep learning, consequently enabling optimal slosh control through Input Shaping and Nonlinear Programming (NLP).

State of the art

Sloshing modeling involves a fundamental trade-off between numerical fidelity and real-time control efficiency. While mechanical analogies are standard, a gap exists in experimentally characterizing such simplified models, especially for small-scale containers. In this context, deep learning-based methods are increasingly utilized for non-intrusive sloshing recognition and data-driven modeling, offering a versatile alternative where traditional sensors are often impractical for small-scale vessels. Simultaneously, trajectory optimization has evolved from classical slosh-mitigation techniques toward numerical methods that respect actuator constraints while decoupling the container's motion from fluid inertia. Moreover, while most studies assume ideal tracking, industrial manipulators inherently exhibit errors from servo dynamics and kinematic singularities. Consequently, large-scale experimental validation is essential to ensure robustness against the aggressive motions and hardware non-idealities of real-world automation.

1. Methodology and Setup

The experimental framework (Fig. 1) integrates a T75 Cell Culture rectangular flask (SPL Life Sciences) of base dimensions $L \times \ell$ filled with a coffee solution to level h. Actuation is handled by a Kawasaki MC004N six-axis manipulator and F60 controller, receiving Cartesian way-

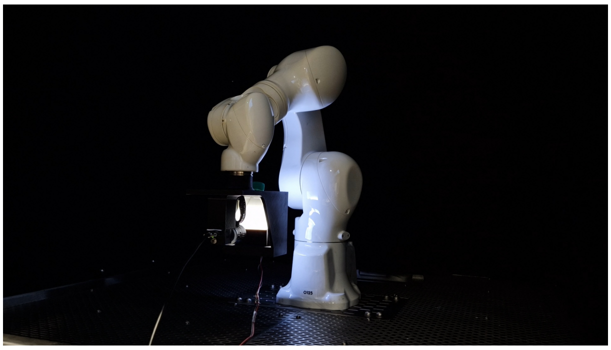

Figure 1: Overall experimental setup.

Figure 2: Robot-mounted custom support.

points via an Ethernet connection. The motion interface is optimized for low latency with a 4 ms update period, single-sample buffer, and linear interpolation to support future closed-loop applications.
Sloshing is captured by a vision-based system comprising an IDS U3-3160CP monochrome camera (2.30 MP) equipped with an 8 mm lens and a 1 mm spacer to ensure a minimum focusing distance. The camera is aligned perpendicular to the container's longitudinal axis, utilizing a high-contrast back-illumination method to simplify edge detection of the liquid surface. This lighting is provided by a custom-built LED grid diffused through 5 mm of BLUESIL RTV 3527 A-B silicone on an aluminum heat sink. To maintain rigid alignment during aggressive motion, the camera, flask, and backlight are secured in a custom-designed 3D-printed PLA frame (Fig. 2) mounted directly to the robot's end-effector. Geometric calibration is carried out using an enhanced planar checkerboard procedure based on Zhang's method while the effective resolving capability of the optical setup is quantified using the slanted-edge method. The measured modulation transfer function (MTF) yields an MTF value of 60.0 lp/mm at 50% of the contrast, corresponding to an object-plane spatial resolution of 0.06 mm. The exposure time

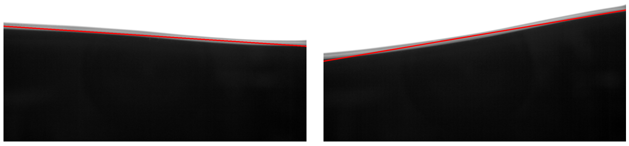

Figure 3: Liquid free-surface profile approximated by a local linear fit.

is fixed to 2 ms in order to operate at 180 fps.

1.1. Free-surface profile extraction

To maximize frame rate and simplify processing, a Region of Interest (ROI) is selected (Fig. 3). The grayscale ROI undergoes Gaussian smoothing to suppress noise, followed by Otsu's thresholding for liquid segmentation and Canny edge detection to extract contour pixels. Given the flask geometry, the free surface is approximated as linear and fitted to $y = mx + b$. The resulting slope m is used to determine the surface angle θ and the wave height at the container wall $\eta = m \cdot L/2$, where L is the container length.

1.2. Experimental Automation

The sloshing acquisition is fully automated via a cross-platform interface. A Python supervisor coordinates the vision system and the robot controller over TCP/IP, relying on two concurrent Kawasaki tasks: a background communication server that receives reset commands and updates shared variables, and a foreground program that manages Real-Time Control mode, allowing trajectories to be executed by an external C++ application interfaced with the controller. For each trial, the supervisor selects the desired trajectory, launches the motion executable, triggers image recording, and then resets the robot state. Trials are scheduled back-to-back so that the final pose of one trajectory coincides with the initial pose of the next, eliminating the need for manual repositioning.

2. Liquid Sloshing Modeling

A multi-modal Mass-Spring-Damper (MSD) model is selected for its balance of efficiency and suitability for rectangular containers [2]. Assuming an incompressible, irrotational fluid and neglecting surface tension, the system represents the liquid via a stationary base mass m_0 and oscillating modal masses m_n, each with specific stiffness k_n and damping c_n. The wave height η at the container wall is predicted, enabling direct validation against vision-based experimental data.

2.1. Model Parametrization

The model assumes longitudinal translation without tilting, with the system defined by dimensions (L, ℓ, h), fluid properties (ρ,ν), and gravity g. Conserving total mass, center of mass height, and moments of inertia, the model partitions m_F into a rigid base m_0 and modal masses m_n (Eq. 1), with natural frequencies ω_n derived from the Laplace equation for velocity potential (Eq. 2). For lateral translation, the model focuses on antisymmetric modes, as horizontal acceleration does not excite symmetric modes (Eq. 3). Specific modal masses m_n and spring constants k_n are determined via analytical relations (Eq. 4-5). The damping ratio ζ, assumed uniform across all modes, is derived from the fluid properties and container width ℓ (Eq. 6).

$$m_F = \rho \cdot \ell \cdot L \cdot h = m_0 + \sum_{n=1}^{\infty} m_n \tag{1}$$

$$\omega_n^2 = g \cdot j_n \tanh(j_n h) \tag{2}$$

$$j_n = (2n+1)\pi/L \tag{3}$$

$$\frac{m_n}{m_F} = \frac{8}{\pi^3} \frac{\tanh((2n+1)\pi h/L)}{(2n+1)^3\, h/L} \tag{4}$$

$$k_n = \omega_n^2 \cdot m_n \tag{5}$$

$$\zeta = C_1 \left(\frac{\nu}{\ell^{3/2}\sqrt{g}} \right)^{n_1} \tag{6}$$

Because damping parameters depend heavily on tank geometry, the constant C_1 and exponent n_1 are tuned to 1 and 0.3, respectively, to better align the model with experimental data.

2.2. Equation of motion and Sloshing Height Estimation

The system dynamics are derived via Lagrange's equations, relating kinetic energy (T), potential energy (V), and Rayleigh dissipation (D) for purely longitudinal translation (Eq. 7).

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{x}_n}\right) - \frac{\partial T}{\partial x_n} + \frac{\partial D}{\partial \dot{x}_n} + \frac{\partial V}{\partial x_n} = 0 \tag{7}$$

Applying the Lagrange operator to each modal coordinate x_n yields the governing equation of motion for the n-th sloshing mode (Eq. 8). Driven by the robotic manipulator's longitudinal acceleration $\ddot{x}_0$ this second-order differential equation predicts the displacements of the modal masses.

$$\ddot{x}_n + 2\omega_n \zeta \dot{x}_n + \omega_n^2 x_n = -\ddot{x}_0 \tag{8}$$

To relate modal displacements x_n to the physical wave height $\bar{\eta}$, the fluid's center of mass, assuming a linear surface ramp, is equated to the center of mass of the discrete MSD system [1] (Eq. 9-10). This allows the n-th mode wave height to be expressed as a function of its modal mass displacement (Eq. 11). The total sloshing height, calculated by summing all considered modes, provides the final scalar for validation against vision-based experimental data.

$$x_G(t) = \frac{1}{V} \int_{-\frac{L}{2}}^{\frac{L}{2}} \int_{-\frac{\ell}{2}}^{\frac{\ell}{2}} \int_{-\frac{h}{2}}^{\frac{h}{2}+\bar{\eta}_n(t)\frac{2x}{L}} x \, dV \tag{9}$$

$$x_G m_F = 0 \cdot m_0 + \sum_n x_n m_n = \sum_n x_n m_n \tag{10}$$

$$\bar{\eta}_n = \frac{6h}{L} \frac{m_n}{m_F} x_n \tag{11}$$

2.3. Deep Learning Enhanced Models

To enhance predictive accuracy, the analytical model is augmented with neural architectures trained on 9,822 acquisitions. This dataset spans standard industrial profiles (splines, harmonic, cycloidal) and optimized trajectories incorporating high-frequency, impulsive loads for slosh minimization. Profiles are truncated to 6 s and padded for uniform dimensions, with training focused on the energy-dominant first sloshing mode. The objective is to minimize the Mean Squared Error (MSE) between the predicted and experimental sloshing profiles (Eq. 12).

$$\mathcal{L}_\eta = \frac{1}{N_{tot}} \sum_{t=1}^{N_{tot}} (\eta_{mod}(t) - \eta_{acq}(t))^2 \tag{12}$$

Recurrent Neural Networks (RNNs) are selected for their ability to process time-history data. Optimal layout and training hyperparameters are identified via Bayesian Optimization using

the Optuna library. Specifically, layer type between LSTM and GRU, hidden layer dimensions, number of layers, dropout rate, learning rate, batch size and loss function weights are optimally tuned. The resulting models are exported into ONNX format to ensure high-performance, real-time deployment.
Three distinct architectures are developed to evaluate the balance between physical interpretability and data-driven flexibility:

- **Parameter Tuning**: Predicts scaling constants for modal parameters (ω_n,ζ, and amplitude) and corrections for container geometry measurement inaccuracies. To ensure physical consistency, the loss function includes regularization terms that penalize deviations from nominal values, forcing the network to provide subtle refinements rather than redefining the system physics.
- **Residual Correction**: Building on the tuned physical model, a parallel NN generates an additive correction $\Delta\eta$ to account for phase shifts and unmodeled dissipative effects. The loss function includes penalties for correction magnitude and temporal smoothness, ensuring the NN remains a secondary refinement to the robust physical baseline.
- **Black-box**: A purely data-driven model that maps the acceleration profile directly to the sloshing time-history. While highly accurate within the training domain, this architecture lacks physical constraints, making a diverse training dataset essential for generalization to unseen maneuvers.

3. Sloshing Control

Two feedforward compensation strategies to mitigate sloshing are evaluated by pre-filtering or reshaping the reference motion: Input Shaping and NLP. These methods leverage known system dynamics to ensure sloshing height control.

3.1. Input Shaping

Input Shaping involves convolving the reference trajectory with a sequence of impulses designed to cancel residual vibrations by targeting the fundamental frequency ω_n and damping ratio ζ [3]. For this application, a Zero Vibration (ZV) shaper is implemented, utilizing two impulses (A1,A2) where the second is delayed by half the damped natural period (Eq. 13).

$$A_1 = \frac{1}{1+K}, \quad A_2 = \frac{K}{1+K}, \quad \Delta t_2 = \frac{\pi}{\omega_d} \quad (13)$$

where K is a scaling factor derived from the damping ratio. In the discrete-time implementation, the trajectory is modified by summing its scaled version with a version delayed by $n = round(\Delta t_2/dt)$ indices, using padding to maintain continuity. This strategy is robust against model inaccuracies and avoids extreme acceleration peaks. However, it introduces a fixed delay to the maneuver. While time-compression can mitigate this delay, it also leads to higher acceleration peaks resulting in actuator saturation.

3.2. Nonlinear Programming

The sloshing control problem is addressed as a gradient-based Optimal Control Problem (OCP) solved via NLP for a 1D, 600 mm straight-line path. Acceleration is selected as the primary control variable (± 3 m/s^2) to enforce box constraints and facilitate model interfacing, with position setpoints derived via numerical integration to ensure smooth motion. To protect the robotic structure, hard constraints are imposed on jerk (± 100 m/s^3), sloshing limits ($\|\eta\| < \eta_{lim}$), and zero terminal velocity.
The algorithm employs a Single Shooting technique in MATLAB using the *fmincon* interior-point solver, utilizing a fixed step fourth-order Runge-Kutta (RK4) integrator for high numerical fidelity. A two-pass strategy, consisting of a feasibility pass followed by a refinement pass, is implemented to minimize a multi-objective cost function J that balances sloshing amplitude, rate of change, and trajectory jerk. It supports Time-optimal control, treating duration T as an additional control variable, and fixed-time minimization for stability-sensitive processes. The framework is further adaptable for rapid mixing or sustained oscillations by reweighting the cost function. While Input Shaping can be superimposed on the optimized motion to reduce residual vibrations, it may compromise the native optimality of NLP profiles.

4. Results and Discussion

The performance of the analytical and deep-learning enhanced models, together with the

Metric	Mean	Std Dev	Unit
MAE	1.0	1.5	[mm]
RMSE	1.4	1.8	[mm]
Max error	4.2	4.2	[mm]
P2P Ratio	6.1	10.1	[%]

Table 1: Analytical model validation.

optimization framework is evaluated through experimental validation.
The vision-based acquisition provides high-fidelity measurements with minimal post-processing. While the robotic manipulator generally performs consistently, joint vibrations and kinematic singularities introduce sporadic outliers and minor flask tilting. However, given the dataset's volume, these effects have a marginal impact on aggregate metrics.

4.1. MSD Model Validation

The baseline MSD model (4 modes) is validated (Table 1) against the aforementioned training set (η_{mean}=2.19 mm) using Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Max Error, together with the percentage difference for peak estimation (P2P). While it successfully captures the primary dynamics, it exhibits a systematic phase lag [4] and overestimated peak amplitudes. Moreover, experimental decay is slower than predicted, likely due to the linear damping assumption neglecting surface tension and wall friction.

4.2. RNN Architectures Performance

The data-driven enhanced models are compared against the analytical benchmark using a validation set (η_{mean}=1.85 mm) of 1000 optimally generated unseen trajectories (Fig. 4):

- Parameter Tuning: shows improvements in MAE/RMSE but systematically underestimates peak sloshing, leading to less conservative predictions.
- Residual Correction: Achieves the highest fidelity, reducing all mean error metrics effectively compensating for the nonlinear phase lag while maintaining physical consistency, but showing a reduced reliability in maximum peak prediction.

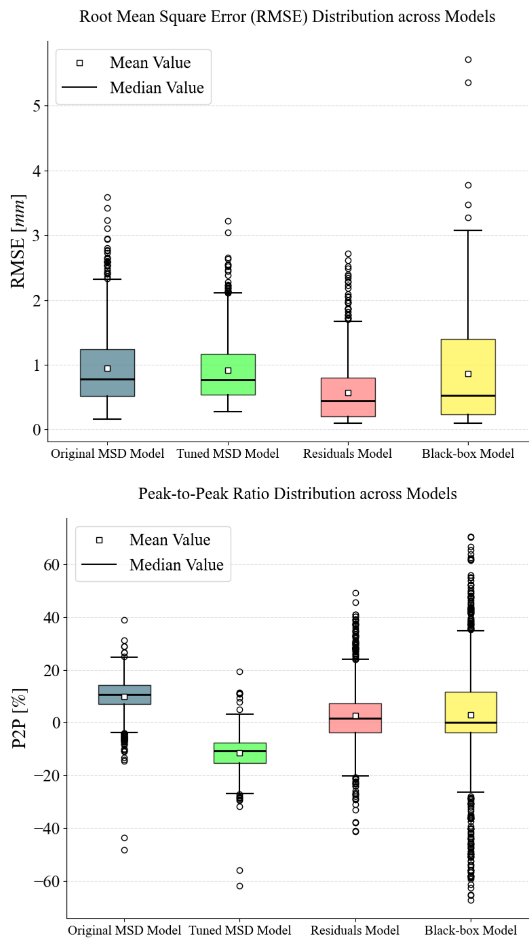

Figure 4: RMSE and Peak-to-Peak metrics distributions over the validation set

- Black-box: While superior on training data, it shows a significant performance drop and high variance on unseen trajectories, occasionally diverging without a physical foundation.

4.3. Trajectory Control Outcomes

Input Shaping (ZV) is applied to a 1.2 s trapezoidal velocity motion profile. The unscaled shaper reduces peak slosh from 20.18 mm to 10.68 mm but adds a 13.8% time delay. Time-scaled shaping maintains the original duration still providing a 62.1% peak reduction compared to unshaped motion, although requiring higher acceleration magnitudes.
Trajectory Optimization is executed across all architectures with different goals:

- Time-Optimality: Based on the NN-tuned MSD model, it successfully reduces execution time from 2 s to 1.33 s. However, model underestimation leads to a real-world peak

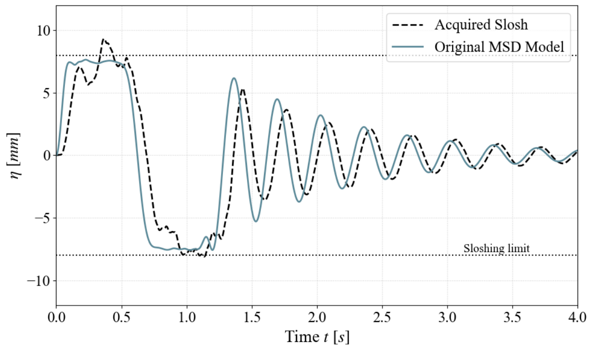

Figure 5: Slosh minimization trajectory optimization utilizing the baseline MSD model

of 7.29 mm against a 5 mm constraint.

- Slosh Minimization: All models effectively shape acceleration to reduce oscillations (Fig. 5), though gradient-based solvers tend to exploit model inaccuracies resulting in underestimations of the sloshing profiles.
- Mixing Maximization: Successfully maximizes free-surface displacement rates, though measured peaks slightly exceed constraints.

4.4. Hybrid Control Performance

The integration of NLP Optimization and Input Shaping proves effective (Figure 6). When trajectory scaling is not performed applying a ZV filter to optimized profiles suppress residual vibrations while preserving the peak attenuation achieved during optimization. The results suggest a superior strategy: performing optimization with a target time reduced by the shaper's delay to robustly mitigate slosh without violating total cycle time constraints.

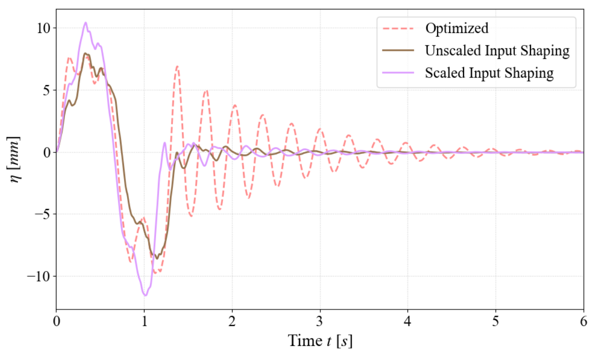

Figure 6: Hybrid control sloshing response

5. Conclusions

This research establishes a comprehensive data-driven framework for robotic sloshing control, validated by an extensive experimental approach. While analytical MSD models provide the necessary speed for optimization, experimental results reveal critical unmodeled phase lags and peak discrepancies. Deep Learning architectures show promising results successfully mitigating average errors and temporal shifts, though peak-height reliability remains challenging during aggressive maneuvers. The integration of NLP-based trajectory optimization and ZV Input Shaping provides a robust pathway for high-speed transport, suppressing residual vibrations and minimizing peak sloshing amplitudes while respecting actuator constraints.

This methodology effectively bridges the gap between theoretical modeling and industrial hardware non-idealities, offering a scalable solution for diverse container geometries. Future developments should focus on refining models to account for nonlinear phase-lag and surface tension, alongside iterative training of the data-driven models on optimized profiles to improve robustness. Additionally, implementing global optimization solvers and extending the framework to 3D trajectories and real-time closed-loop control will unlock the full potential of this approach for complex industrial environments.

References

[1] Di Leva, Roberto, et al. "Sloshing dynamics estimation for liquid-filled containers performing 3-dimensional motions: modeling and experimental validation." Multibody System Dynamics 56.2 (2022): 153-171.

[2] R. A. Ibrahim, Liquid sloshing dynamics: theory and applications. Cambridge University Press, 2005.

[3] Aboel-Hassan, et al. "Design and optimization of input shapers for liquid slosh suppression." Journal of Sound and Vibration 320.1-2 (2009): 1-15.

[4] Bäuerlein, et al. "Phase lag predicts nonlinear response maxima in liquid-sloshing experiments." Journal of Fluid Mechanics 925 (2021): A22.