Amortized Neural-Symbolic Inverse Dynamics: Robust and Invariant Parameter Inference via Latent Dynamics

Matteo Rufolo [1] **Dario Piga** [1] **Marco Forgione** [1]

Abstract

We propose an amortized neural-symbolic framework for physical parameter inference in dynamical systems. Building on the CAMEL meta-learning framework, whose latent task embeddings carry provable identifiability guarantees, we train a post-hoc MLP to directly recover physical parameters that are nonlinearly related to trajectory data, without per-trajectory optimization or explicit knowledge of the dynamics. We distill this neural mapping into closed-form symbolic expressions via symbolic regression, achieving robust out-of-distribution generalization that validates the recovered physics. To handle realistic conditions, we address measurement noise through domain randomization and improve robustness to unobserved nuisance variables via a structured grouped training strategy. Experiments on a simulated 2-DOF robotic arm show competitive accuracy with classical identification baselines at $\sim 1000\times$ faster inference, with interpretable parameter mappings and superior noise robustness.

1. Introduction

The estimation of physical parameters from observed trajectories is a fundamental challenge across experimental sciences, robotics, and control (Leboutet et al., 2021). Classical approaches such as least squares and maximum likelihood estimation are well-understood and exact when the likelihood of observations given parameters is tractable, that is, when an explicit analytical model linking parameters to observations is available. However, in many realistic physical systems, this likelihood is intractable: the forward model is complex, nonlinear, or only accessible through simulation. In these settings, Simulation-Based Inference (SBI) (Cranmer et al., 2020a) offers a principled alternative, replacing explicit likelihood evaluation with neural approximators trained on simulated data, but these methods have no interpretability guarantee.

While the system identification (SYSID) and data-driven literature offer several robust approaches for parameter estimation, they often present significant computational or analytical bottlenecks, typically requiring explicit analytical knowledge of the system or relying on expensive per-trajectory optimization to fit coefficients.

To overcome these bottlenecks, we require a framework that provides both amortized inference and formal interpretability. For this reason, we build upon the CAMEL scientific meta-learning framework (Blanke & Lelarge, 2024) (see Figure 1 for an overview of the full pipeline). Rooted in the "learning to learn" paradigm (Schmidhuber, 1987), CAMEL operates under the structural assumption that a set of base physical parameters enters the system dynamics linearly. By enforcing this affine structure during observation prediction, it yields meta-learned task embeddings that are provably linearly identifiable with respect to these true physical parameters.

However, this strict assumption of parameter linearity severely limits its applicability to more complex, real-world systems. We extend this framework by relaxing this assumption, allowing the physical parameters of interest to govern the dynamics nonlinearly. Instead, we rely only on the structural decoupling between the system input and the parameters. Specifically, we extract these nonlinear relations of physical parameters by training a post-hoc multilayer perceptron (MLP) on top of the latent representations learned by CAMEL. To enhance interpretability, we subsequently distill this neural mapping into closed-form algebraic equations via Symbolic Regression (SR) (Cranmer et al., 2020b). The strong out-of-distribution (OOD) performance obtained by these learned symbolic expressions suggests that the true underlying physics has been accurately recovered.

Beyond the extraction of the governing equations, several open challenges remain in realistic settings, particularly regarding how to robustify parameter estimation against measurement noise and how to disentangle observations

[1]IDSIA, USI-SUPSI, Lugano, Switzerland. Correspondence to: Matteo Rufolo <matteo.rufolo@supsi.ch>.

Proceedings of the AI4Physics Workshop at the 43rd International Conference on Machine Learning (AI4Physics@ICML 2026), Seoul, South Korea. 2026. Copyright 2026 by the author(s).

from unmodeled, shifting nuisance variables. To illustrate this, consider a complex robotic use case: a setting with multiple manipulator arms where distal payloads are known, but proximal link parameters vary between arms and cannot be directly observed. In such a scenario, we aim to achieve accurate payload mass estimation for novel payloads and new manipulators whose link parameters may be OOD compared to the training data. The core question is whether a model can reliably estimate this mass using only sparse data, without explicit knowledge of the proximal link parameters.

While our amortized inference pipeline, bolstered by domain randomization (Tobin et al., 2017) and trajectory filtering (Gautier et al., 2012), generalizes well under measurement noise, it inherently struggles in OOD regimes where the distribution of unobserved nuisance variables shifts. To overcome this limitation, we enhance the framework by integrating a structured grouped training strategy. Finally, we introduce a novel *Drift* metric that allows us to quantify the severity of these distributional shifts and estimate the reliability of the model's predictions *a priori*.

We analyze this framework within a robotics context, though the methodology applies broadly to other physical systems under standard identifiability assumptions. As an instantiation of this problem, we study the inverse dynamics of a planar 2-DOF robotic arm. In our experiments, the robot's link parameters act alternately as the target physical parameters to estimate and as the unobserved nuisances, demonstrating how effectively the framework disentangles these complex, interacting dynamic factors. The full framework pipeline is illustrated in Figure 1.

Contributions

1. We extend CAMEL to an amortized inverse-dynamics setting, using a post-hoc MLP to recover physical parameters even when they are nonlinearly coupled to the dynamics.
2. Leveraging CAMEL's identifiability guarantees, we distill the learned mapping into interpretable closed-form expressions via Symbolic Regression, whose OOD performance serves as a validation of the recovered physics.
3. We improve robustness to real-world imperfections by (i) incorporating domain randomization to handle measurement noise and (ii) introducing a grouped training strategy that enhances and diagnoses robustness to unobserved nuisance variables.

2. Related Works

Classical Physical Parameter Inference. Estimating physical parameters from trajectory data has a long history. Classical SYSID approaches such as IDIM-LS (Gautier & Khalil, 1990) are intrinsically data-driven: they exploit the analytical linear relationship between joint torques and inertial parameters to fit physical parameters from measured trajectories via ordinary least squares (Swevers et al., 2002), with extensions like IDIM-PIV (Brunot et al., 2018) addressing sensitivity to measurement noise. While exact and interpretable, these methods require explicit analytical derivation of the system's regressor matrix and a separate per-trajectory optimization at test time. Similarly, data-driven methods like SINDy (Brunton et al., 2016) excel at discovering sparse governing equations but still rely on per-trajectory optimization to fit coefficients, meaning computation cannot be amortized across systems.

Simulation-Based Inference (SBI). A broader framework for amortized parameter inference is SBI (Cranmer et al., 2020a). SBI addresses the common scientific setting where the likelihood of observations given parameters is intractable, but forward simulation is available. Methods such as Neural Posterior Estimation (Lueckmann et al., 2017) train neural networks to approximate the posterior parameter distribution from simulated data. SBI has proven highly effective across domains (Alsing et al., 2019; Brehmer & Cranmer, 2022). However, standard SBI approaches are purely data-driven and black-box: they lack structural identifiability guarantees, do not embed analytical knowledge of the system dynamics, and typically require massive simulation budgets.

Meta-learning for Dynamical Systems. Structurally closer to our setting are meta-learning approaches such as CoDA (Kirchmeyer et al., 2022), DISCO (Morel et al., 2025), and CORAL (Serrano et al., 2023), which explicitly model a family of dynamical systems and learn task-specific latent embeddings for fast adaptation. Unlike SBI, they embed prior knowledge of the dynamical structure and do not require exhaustive simulation coverage. However, they share with SBI the absence of formal guarantees on what their learned representations encode. The CAMEL framework (Blanke & Lelarge, 2024) closes this gap by bridging the interpretability of classical methods with modern meta-learning. Our work extends CAMEL beyond its original affine regime via post-hoc MLP training, enabling parameter recovery in the nonlinear case, and distills the learned mapping into closed-form symbolic expressions, a capability absent in both SBI and purely neural meta-learning.

3. Background

3.1. Meta-learning formulation for physical systems

We assume access to a meta-dataset $\mathcal{D} := \bigcup_{t=1}^{T} D_t$ with $D_t = \{(x_{t,i}, y_{t,i})\}_{i=1}^{N_t}$, composed of T distinct datasets. Each D_t is generated by observing a physical system of the same class under varying experimental conditions; in our case, inter-task variation is driven purely by changes in the underlying physical parameters (e.g., masses, link lengths). In standard optimization-based meta-learning (Hospedales

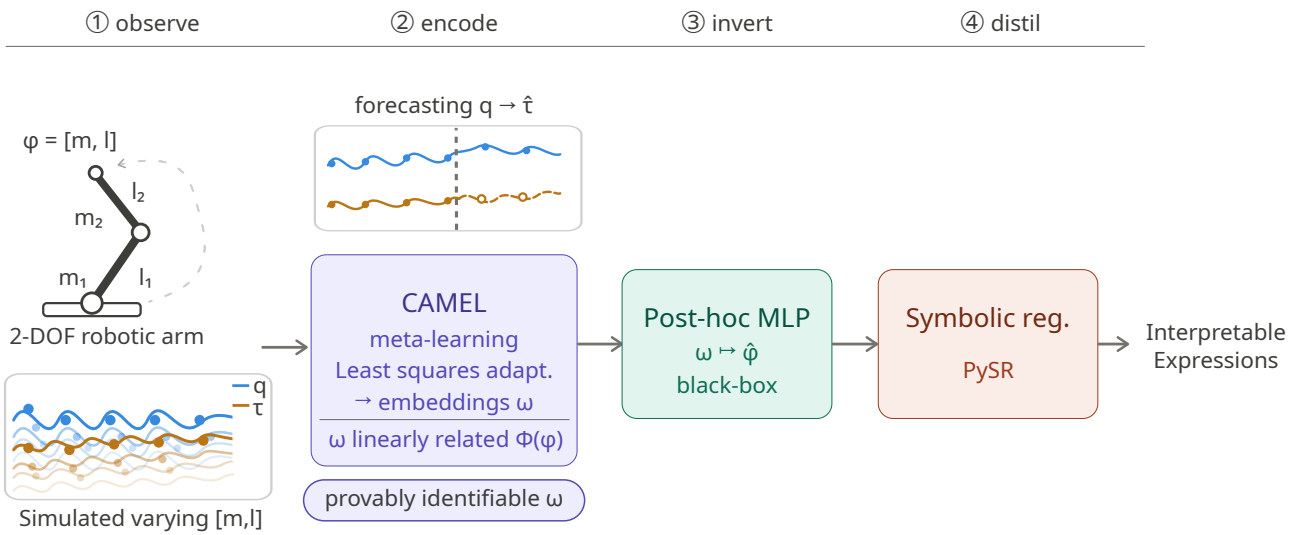

Figure 1. Overview of the proposed framework. (1) CAMEL meta-learns task embeddings from simulated trajectories; (2) a post-hoc MLP maps embeddings to physical parameters; (3) Symbolic Regression distills the learned mapping into closed-form expression.

et al., 2021), the goal is to learn a predictive dynamics model $x \mapsto y$ that can rapidly adapt to a new, unseen system using only a few trajectory samples. To achieve this fast adaptation, these architectures typically employ a dual parameterization $f(x; \theta, \omega)$. Here, θ represents task-agnostic parameters (capturing the shared physics across all environments), while ω represents task-specific weights that encode the unique physical parameters of the given system. For each task t, ω_t is computed via an adaptation rule $\omega_t = A(\pi, D_t)$, where π denotes trainable meta-parameters.

During meta-training, the parameters are optimized by minimizing an aggregated loss over all tasks:

$$L(\theta, \pi) = \sum_{t=1}^{T} \frac{1}{N_t} \sum_{i=1}^{N_t} \ell\big(f(x_{t,i}; \theta, A(\pi, D_t)), y_{t,i}\big), \quad (1)$$

where $\ell(\cdot, \cdot)$ is a task-specific loss function (e.g., Mean Square Error (MSE)). Since MAML (Finn et al., 2017), such models are typically trained via bi-level optimization: the inner adaptation step $A(\pi, D_t)$ performs gradient descent, so optimizing $L(\theta, \pi)$ requires expensive higher-order derivatives. At meta-test time, the shared parameters $\bar{\theta}$ are frozen, and for a new task $T + 1$ the task-specific weights ω_{T+1} are obtained by minimizing the prediction error on the new task's support data.

3.2. The CAMEL framework

The CAMEL framework (Blanke & Lelarge, 2024) belongs to optimization-based methods and relies on a core inductive bias: the equations governing many physical systems exhibit an affine dependence on their varying physical parameters. CAMEL enforces this structural assumption by parameterizing the two-fold parameterized meta-model f as an affine function of low-dimensional, task-specific weights $\omega_t \in \mathbb{R}^r$, combined with a task-agnostic feature map $v(x; \theta) \in \mathbb{R}^r$ and a task-agnostic bias $c(x; \theta) \in \mathbb{R}$, with $\theta \in \mathbb{R}^p$:

$$f(x; \theta, \omega_t) = c(x; \theta) + \omega_t^\top v(x; \theta). \quad (2)$$

Similarly to CoDA (Kirchmeyer et al., 2022), CAMEL is an optimization-based meta-learning method that avoids bi-level optimization by eliminating the inner gradient descent loop. Instead, the task adaptation is defined by a fixed, closed-form structural prior. This allows the model to be trained via standard joint optimization (over θ and π) using a single-level gradient descent. In particular, because θ and ω_t are decoupled in the affine formulation, CAMEL treats the task-specific weights $\{\omega_t\}_{t=1}^{T}$ directly as the meta-parameters π (i.e., $A(\pi, D_t) = \omega_t$). Thanks to the affine structure of f, at test time and given a new dataset D_{T+1}, the adaptation rule reduces to a simple OLS regression over the frozen feature map:

$$\hat{\omega}_{T+1} = \arg\min_{\omega \in \mathbb{R}^r} \frac{1}{2N_{T+1}} \sum_{i=1}^{N_{T+1}} \Big(c(x_{T+1,i}; \bar{\theta}) + \omega^\top v(x_{T+1,i}; \bar{\theta}) - y_{T+1,i}\Big)^2.$$

This yields a faster and more stable adaptation phase than other methods, at the cost of some expressivity.

The dimensionality r must be chosen such that it is larger than the number of underlying varying physical parameters, but smaller than the total number of training tasks T. This ensures that the feature map v is observed along a sufficient number of directions without overfitting.

Reliability and identifiability. Beyond computational efficiency, the most critical property of CAMEL for our objective is its reliability. If the underlying ground-truth dynamics is really linearly parameterized by the true physical parameters $\varphi_t \in \mathbb{R}^n$, taking the form

$$f_\star(x; \varphi_t) := \kappa(x) + \varphi_t^\top \nu(x), \quad \nu(x) \in \mathbb{R}^n, \tag{3}$$

it has been theoretically proven by Blanke & Lelarge (2024) that the task-specific weights ω_t learned by CAMEL are related to the true physical parameters φ_t via an affine transformation. Specifically, in the limit of vanishing training loss, there exists a matrix P such that

$$\varphi_t = P\omega_t \quad \text{for all tasks } t. \tag{4}$$

This provides a powerful guarantee: even though the task representations ω_t are learned purely from trajectory data in a black-box fashion, they map directly to the true, interpretable physical parameters of the system. However, while this affine assumption holds for many standard systems, it is often overly restrictive in practice, motivating the direction of our contribution.

4. Amortized Neural-Symbolic Inference

In many practical scenarios, the strict CAMEL linearity assumption is invalid, even in simple systems, such as a pendulum with friction, the physical parameters to be estimated often couple nonlinearly. In these cases, the system takes the more general form:

$$f_\star(x; \varphi_t) := \kappa(x) + \Phi(\varphi_t)^\top \nu(x), \quad \nu(x) \in \mathbb{R}^m, \tag{5}$$

where Φ is a nonlinear mapping of φ_t. CAMEL still functions as an observation predictor and its identifiability guarantee naturally extends to this setting: there exists a matrix P satisfying $\Phi(\varphi_t) = P\omega_t$ for all tasks t, meaning the latent embeddings capture a linear projection of the nonlinear parameter manifold, preserving identifiability up to the transformation Φ.

4.1. Amortized inference via post-hoc regression

To recover the true physical parameters from the meta-learned embeddings, the most straightforward approach is to train a post-hoc neural network $g_\vartheta : \omega \mapsto \varphi$ using the meta-training datasets where the physical parameters are known. Provided the latent dimensionality is sufficient to capture the parameter variations, this data-driven mapping naturally learns the inverse of the linear projection, bypassing the need to analytically derive the functional form of Φ. The model is trained via a standard supervised MSE loss:

$$\mathcal{L}_{\text{sup}}(\vartheta) = \frac{1}{T} \sum_{t=1}^{T} \left\| g_\vartheta(\omega_t) - \varphi_t \right\|^2. \tag{6}$$

At test time, given a new trajectory, CAMEL rapidly adapts the embedding ω_{t+1} via OLS, and the pre-trained network g_ϑ predicts the physical parameters within a forward pass.

It is crucial to emphasize why our framework builds specifically upon CAMEL rather than similarly structured meta-learning approaches like CoDA (Kirchmeyer et al., 2022). The fundamental distinction lies in CAMEL's theoretical reliability: we are guaranteed that the embeddings ω strictly capture a linear projection of $\Phi(\varphi)$. This identifiability guarantee forces the post-hoc mapping g_ϑ to mirror the true underlying physical relationships. Applying our MLP pipeline to CoDA embeddings would be physically unjustified, as black-box meta-learning methods lack these guarantees, resulting in latent spaces that do not necessarily match the true inverse mapping, a flaw that severely hinders OOD generalization. CAMEL's identifiability is therefore essential to ensure that the learned mapping is physically meaningful, allowing SR to extract accurate, generalized expressions. Nevertheless, this naive amortized inference still struggles under measurement noise, OOD parameter shifts, and unobserved nuisance variables.

Robustness to measurement noise and OOD shifts. In sim-to-real applications, test trajectories are frequently corrupted by measurement noise. In inverse dynamics, where coordinate derivatives are often obtained numerically, this noise is amplified. To mitigate this, we inject white noise into the training trajectories, a form of domain randomization (Tobin et al., 2017), combined with trajectory filtering (Gautier et al., 2012). This forces the amortized inference pipeline to build representations robust to signal degradation, an advantage difficult to achieve with classical SYSID.

Furthermore, neural networks are notoriously poor at extrapolating to OOD settings (Rahaman et al., 2019). To preserve the interpretability intrinsic to the CAMEL framework and to rigorously validate the recovered physics, we apply SR on top of the learned mapping $g_\vartheta(\omega)$. By extracting a closed-form algebraic expression for the parameter mapping, SR provides a physically structured hypothesis whose OOD performance serves as a stringent test of how accurately the underlying dynamics have been recovered.

4.2. Nuisance-Invariant Representations

A critical failure mode of naive amortized inference arises when dealing with unobserved nuisance variables. Consider a dynamical system governed by both target parameters φ_{tgt} (known in training, to be estimated at test time) and nuisance parameters φ_{nui} (unmodeled, unmeasured, and varying uncontrollably). Because the CAMEL framework $f(x; \theta, \omega)$ is optimized purely to minimize trajectory prediction error, it naturally exploits any correlation in the data that helps lower

this loss, entangling information about both φ_{tgt} and φ_{nui} within the latent embeddings ω. Consequently, a supervised mapping $g_\vartheta : \omega \mapsto \varphi_{\text{tgt}}$ learned naively via Eq. (6) does not isolate the targets, but instead relies on correlations tied to the specific training distribution of the nuisances. When this distribution shifts at test time, these correlations break down, causing the estimator to become brittle even if the target parameters remain unchanged.

Grouped training for implicit nuisance invariance. In standard scientific meta-learning, as in CAMEL (Blanke & Lelarge, 2024), the model is trained over a family of randomly sampled systems. Our key observation is that such random sampling provides no explicit signal about how embeddings change when nuisances vary while targets are held fixed. Instead of sampling trajectories uniformly at random, we propose a structured data collection strategy that explicitly exposes nuisance-induced variation at fixed targets. We construct N *anchor groups*, each defined by a fixed target value φ^i_{tgt}, and for each anchor simulate K trajectories by varying only the unrecorded nuisance conditions. Passing these trajectories through the CAMEL encoder yields embeddings $\{\omega^{(i,k)}\}_{k=1}^K$ that share the same target parameters while reflecting different nuisance environments.

Although fixing target parameters while varying unobserved nuisances may appear restrictive, this setting matches many experimental scenarios in which targets are intrinsic properties and nuisances are environmental fluctuations. For example, when estimating the thermal conductivity of a material sample, ambient temperature and humidity may vary across measurement days while the underlying material properties remain unchanged. Whenever these environmental factors vary largely independently of the target, they naturally provide the diversity required by our grouped training strategy.

The estimator g_ϑ is then optimized via a standard supervised objective over this grouped dataset:

$$\mathcal{L}_{\text{grouped}}(\vartheta) = \frac{1}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K}\left\|g_\vartheta(\omega^{(i,k)}) - \varphi^i_{\text{tgt}}\right\|^2. \quad (7)$$

Since the variation among these embeddings in each group is driven primarily by the nuisance parameters, the estimator is biased away from relying on nuisance-sensitive features in ω and towards features that are stable within each group. Importantly, this invariance is encouraged without requiring explicit labels or detailed priors about φ_{nui}.

We also explored explicit regularization strategies in the embedding space, including contrastive consistency penalties (Chen et al., 2020) and adversarial learning (Ganin et al., 2016). However, contrastive regularization did not produce consistent gains over our grouped supervised loss alone (Appendix B.1), and adversarial learning requires explicit labels for the nuisance parameters, a requirement that strictly violates our core assumption that nuisances are unobserved. Diagnostic analysis suggests that the main bottleneck lies upstream: φ_{nui} becomes deeply entangled in the CAMEL embeddings during the initial meta-learning phase. As a result, enforcing invariance exclusively at the post-hoc stage through g_ϑ is fundamentally insufficient if the relevant invariant structure has not been preserved by the encoder. Extending the CAMEL meta-loss to explicitly decouple target and nuisance factors during representation learning is therefore a promising direction for future work.

5. Experiments

5.1. Euler-Lagrange Dynamics

As a case study, we focus on the simulation of inverse dynamics of a 2-DOF planar robotic arm, described by the Euler-Lagrange formulation:

$$\tau = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q), \quad (8)$$

where $q \in \mathbb{R}^2$ is the generalized coordinate vector (joint angles), $M(q) \in \mathbb{R}^{2\times 2}$ is the inertia matrix, $C(q,\dot{q}) \in \mathbb{R}^{2\times 2}$ is the Coriolis and centrifugal force matrix, $g(q) \in \mathbb{R}^2$ is the gravity vector, and $\tau \in \mathbb{R}^2$ is the generalized torques.

For a planar 2-DOF arm, the four physical parameters $\varphi = [m_1, l_1, m_2, l_2]$ (masses and link lengths in kg and m) appear nonlinearly and are heavily entangled across M, C, and g. The explicit equations read:

$$\begin{aligned}\tau_1 &= (m_1 l_1^2 + m_2(l_1^2 + 2l_1 l_2 \cos q_2 + l_2^2))\ddot{q}_1 \\ &\quad + m_2(l_1 l_2 \cos q_2 + l_2^2)\ddot{q}_2 - m_2 l_1 l_2 \sin q_2 (2\dot{q}_1\dot{q}_2 + \dot{q}_2^2) \\ &\quad + (m_1 + m_2) g l_1 \cos q_1 + m_2 g l_2 \cos(q_1 + q_2), \quad (9) \\ \tau_2 &= m_2(l_1 l_2 \cos q_2 + l_2^2)\ddot{q}_1 + m_2 l_2^2 \ddot{q}_2 \\ &\quad + m_2 l_1 l_2 \sin q_2\, \dot{q}_1^2 + m_2 g l_2 \cos(q_1 + q_2).\end{aligned}$$

This algebraic entanglement forces any trajectory-based estimator to reason over all parameters simultaneously within the same latent representation, even when only a subset is of direct interest or measurable. In our setting, this entanglement is precisely the challenge that nuisance-aware inference must solve.

Trajectory design and persistent excitation. Following standard SYSID practices (Swevers et al., 2002), all tasks share a fixed multi-sinusoidal excitation trajectory: $q_j(t) = A_j \sin(\Omega_j t)$ for $j \in \{1, 2\}$, with $A_1 = 1.2, \Omega_1 = 1.5$ rad/s, and $A_2 = 1.0, \Omega_2 = 1.8$ rad/s. This trajectory is sampled at $\Delta t = 0.01$ s for $N = 400$ steps and kept identical across all tasks; inter-task variation arises strictly from changes in φ. These specific frequencies satisfy the persistent excitation condition for the 2-DOF arm (Gautier, 1991), yielding a full-column-rank regressor matrix that ensures all base parameters are jointly identifiable. In noiseless regimes,

velocities and accelerations are obtained analytically. Under measurement noise, they are recovered via numerical derivation (as required in a realistic scenario).

5.2. Experimental Setup and Embedding Ablation

To evaluate the 2-DOF inverse dynamics model (Eq. (9)), we generate meta-datasets comprising 400 task environments, each simulated for 400 time steps. Physical parameters φ are uniformly sampled from $[0.5, 2.0]$ kg $\times$ $[0.5, 1.5]$ m $\times$ $[0.5, 2.0]$ kg $\times$ $[0.5, 1.5]$ m. To modularize the architecture, we train two independent CAMEL models (one per torque component τ_1, τ_2) for 3000 epochs with 10^{-2} as learning rate. This yields two distinct task-specific embeddings per environment $[\omega_1, \omega_2] \in \mathbb{R}^{\mathbf{r}}$. Then, a shared post-hoc MLP maps the concatenated task embeddings $[\omega_1, \omega_2]$ back to φ, trained for 8000 epochs at 10^{-3} as learning rate.

Ablation on embedding dimension r. A crucial CAMEL hyperparameter is the latent capacity $\mathbf{r} = [r_1, r_2]$. Structural analysis of Eq. (9) dictates that the minimum ranks required for exact torque prediction are $\mathbf{r} = [5, 3]$. However, our ablation (evaluated on 100 test environments, averaged over 10 random seeds and reported in Figure 2) yields a surprising finding: choosing $\mathbf{r} \in \{[2, 2], [3, 1], [1, 3]\}$, configurations that match the strict mathematical minimum of underlying physical parameters φ, still enables reliable parameter estimation. This proves that even when CAMEL lacks the capacity for perfect torque reconstruction, it encodes a faithful low-dimensional manifold of the projection of $\Phi(\varphi)$ that the MLP can successfully invert.

Furthermore, amortized inference training remains highly efficient across all tested dimensions; on an NVIDIA RTX 3090 GPU (24GB) using Pytorch v2.4, training times for $r \in \{[1, 1], [2, 2], [3, 1], [1, 3], [3, 3], [5, 3], [7, 5]\}$ are respectively 75.4, 55.3, 93.8, 103.8, 119.0, 111.2, and 119.9 seconds. For consistency across all experiments, we fix $\mathbf{r} = [5, 3]$ for the CAMEL configuration, as it achieves stable torque prediction with minimal additional training cost.

5.3. Noise robustness

To simulate realistic measurement noise, we corrupt only the observable joint positions q with additive white Gaussian noise, evaluating three noise levels: $\sigma \in \{0.0, 0.02, 0.1\}$ on normalized trajectories. Since velocities and accelerations must be obtained via numerical differentiation, this noise is severely amplified, leading to significant signal degradation. To mitigate this, following standard practices in classical system identification, we apply Savitzky-Golay filtering (Savitzky & Golay, 1964) (window 21, polynomial order 3) to the observed trajectories to obtain stable numerical derivatives.

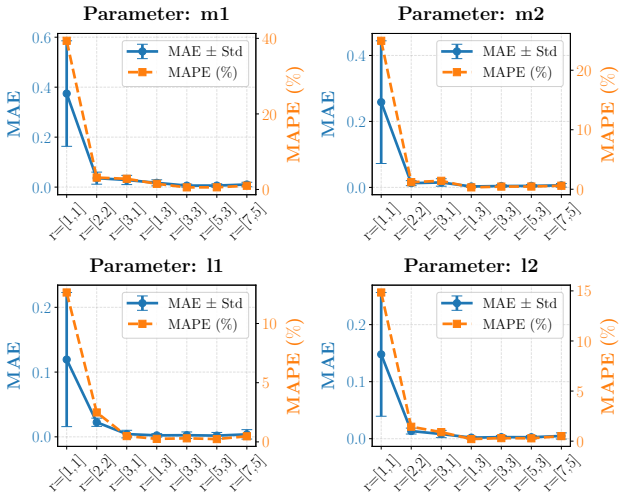

Figure 2. Ablation of CAMEL embedding dimension $\mathbf{r} = [r_1, r_2]$ on physical parameter estimation accuracy (MAE$\pm$ STD & MAPE). Results averaged over 100 test tasks and 10 seeds.

We compare our approach against IDIM-LS, a classical baseline that utilizes the exact analytical regressor. As expected, IDIM-LS achieves lower parameter estimation error in the noiseless scenario (first row of Table 1). However, the amortized inference remains competitive in absolute error while offering a dramatic $\sim 1000\times$ advantage in inference speed (reducing fitting from ~ 2 seconds to ~ 3 milliseconds).

The key advantage of our amortized approach over IDIM-LS is the ability to exploit domain randomization (Tobin et al., 2017). By injecting white noise at the same testing noise levels into the training trajectories, the CAMEL+MLP pipeline builds representations that are inherently robust to signal corruption. To justify the necessity of this MLP, we also evaluate a "Linear CAMEL" baseline mapping $\omega \mapsto \varphi$ via regularized Linear Regression (Ridge Regression[1]).

Table 1 reports the resulting Mean Absolute Percentage Error (MAPE) and Mean Absolute Error (MAE), along with the standard deviation over test tasks. The results clearly demonstrate the necessity of a nonlinear mapping: the MLP achieves $20 - 40\times$ lower error than Linear CAMEL at $\sigma = 0.0$ (e.g., 0.46% vs 13.58% for m_1). Interestingly, Linear CAMEL's error is almost flat across all noise levels, proving that its performance bottleneck is not measurement noise, but rather its fundamental inability to capture the nonlinear couplings between parameters.

This gap admits a clean theoretical explanation. As shown in (Blanke & Lelarge, 2024), CAMEL's affine structure is motivated by a first-order Taylor expansion of the true

[1]Ridge regularization is strictly required; without it, the noiseless $\sigma = 0.0$ scenario learns a highly collinear ω space, rendering the OLS matrix ill-conditioned and resulting in catastrophic parameter divergence.

Table 1. Parameter estimation error under different measurement noise levels σ on normalized trajectories. "IDIM-LS" is the classical inverse dynamics baseline. "Lin CAMEL Matched" uses Ridge Regression and "NonLin CAMEL Matched" denotes our MLP, trained both with the exact test noise level, and "NonLin CAMEL Fixed" uses a single MLP trained at $\sigma = 0.02$ evaluated everywhere. We report MAPE and MAE across test tasks, with the standard deviation denoted in subscripts. Best values are **bold**, second-best are underlined.

		MAPE [%]				MAE (std in subscripts)			
σ_{test}	Param.	IDIM-LS	Lin CAMEL Matched	NonLin CAMEL Matched	NonLin CAMEL Fixed	IDIM-LS	Lin CAMEL Matched	NonLin CAMEL Matched	NonLin CAMEL Fixed
0.00	m_1	$\mathbf{1.71e{-5}}$	13.58	0.46	1.80	$\mathbf{1.94e{-7}}_{\pm 1.67e-7}$	$0.147_{\pm 0.123}$	$0.005_{\pm 0.004}$	$0.018_{\pm 0.010}$
	m_2	$\mathbf{1.02e{-5}}$	7.00	0.34	0.68	$\mathbf{1.16e{-7}}_{\pm 8.87e-8}$	$0.079_{\pm 0.054}$	$0.003_{\pm 0.002}$	$0.008_{\pm 0.006}$
	l_1	$\mathbf{4.60e{-6}}$	8.29	0.22	0.40	$\mathbf{4.47e{-8}}_{\pm 5.15e-8}$	$0.074_{\pm 0.061}$	$0.002_{\pm 0.002}$	$0.004_{\pm 0.003}$
	l_2	$\mathbf{5.59e{-6}}$	9.22	0.35	0.56	$\mathbf{5.66e{-8}}_{\pm 5.74e-8}$	$0.092_{\pm 0.071}$	$0.003_{\pm 0.002}$	$0.006_{\pm 0.003}$
0.02	m_1	365.0	13.86	**1.80**	**1.80**	$3.69_{\pm 0.51}$	$0.150_{\pm 0.122}$	$\mathbf{0.019}_{\pm 0.011}$	$\mathbf{0.019}_{\pm 0.011}$
	m_2	176.0	7.37	**0.76**	**0.76**	$2.08_{\pm 1.04}$	$0.082_{\pm 0.056}$	$\mathbf{0.009}_{\pm 0.008}$	$\mathbf{0.009}_{\pm 0.008}$
	l_1	71.7	8.40	**0.46**	**0.46**	$0.725_{\pm 0.218}$	$0.075_{\pm 0.062}$	$\mathbf{0.004}_{\pm 0.003}$	$\mathbf{0.004}_{\pm 0.003}$
	l_2	56.2	9.26	**0.66**	**0.66**	$0.588_{\pm 0.205}$	$0.093_{\pm 0.071}$	$\mathbf{0.007}_{\pm 0.005}$	$\mathbf{0.007}_{\pm 0.005}$
0.10	m_1	376.0	14.12	**2.02**	4.63	$3.77_{\pm 0.44}$	$0.155_{\pm 0.127}$	$\mathbf{0.022}_{\pm 0.016}$	$0.052_{\pm 0.050}$
	m_2	359.0	7.81	**2.03**	2.38	$3.69_{\pm 0.47}$	$0.088_{\pm 0.065}$	$\mathbf{0.024}_{\pm 0.020}$	$0.030_{\pm 0.028}$
	l_1	83.2	8.56	**1.07**	2.41	$0.845_{\pm 0.261}$	$0.076_{\pm 0.063}$	$\mathbf{0.011}_{\pm 0.008}$	$0.023_{\pm 0.019}$
	l_2	88.0	9.67	**1.48**	2.18	$0.926_{\pm 0.274}$	$0.098_{\pm 0.072}$	$\mathbf{0.014}_{\pm 0.011}$	$0.020_{\pm 0.017}$

dynamics around a nominal parameter φ^*:

$$f_\star(x;\, \varphi_t) \approx f_\star(x;\, \varphi^*) + \delta\varphi_t^\top \nabla f_\star(x;\, \varphi^*),$$

which is affine in $\delta\varphi_t = \varphi_t - \varphi^*$ and therefore consistent with CAMEL's structural assumption. However, this linearization is exact only when parameter variations are infinitesimally small. In our 2-DOF robotic arm, the physical parameters couple nonlinearly through terms such as $m_2 l_1 l_2 \cos(q_2)$, so even within the moderate training range $[0.5,\, 2.0]$ kg $\times$ $[0.5,\, 1.5]$ m, the affine approximation accumulates a non-negligible residual. This is precisely the gap our post-hoc MLP addresses: it corrects the residual nonlinearity that a linear $\omega \to \varphi$ mapping cannot invert.

Overall, the proposed approach yields substantially more stable parameter estimates across noise levels than IDIM-LS, which does not benefit from noise augmentation. While IDIM-LS achieves near-perfect accuracy under idealized conditions, it completely collapses (m_1-MAPE $> 300\%$) under even minor measurement noise ($\sigma = 0.02$). Conversely, our MLP-based mapping trades a negligible approximation error in the noiseless case ($< 0.5\%$) for massive robustness, maintaining bounded errors ($1 - 3\%$) across all noise regimes.

To provide a fair comparison, we also evaluate a realistic scenario where test-time noise level is not perfectly known. We train the model with a fixed domain randomization level of $\sigma_{\text{train}} = 0.02$ and evaluate it across all test noise levels. Table 1 confirms that even under this mismatch, the amortized approach degrades gracefully relative to the noise-matched model while remaining consistently more accurate than IDIM-LS at moderate and high noise levels.

5.4. Symbolic distillation and OOD extrapolation

As analyzed in Section 4.1, CAMEL guarantees that the learned mapping $\omega \mapsto \varphi$ reflects the true inverse of the physical parameter mapping Φ. To provide the interpretability required for sim-to-real transfer and robotic control, where black-box predictions are often untrustworthy, we use SR to distill this data-driven MLP into compact, fixed-form algebraic expressions.

Because the MLP must simultaneously invert both the nonlinear mapping Φ and CAMEL's linear projection, the effective parameter nesting is highly complex. To unroll this, we employ an iterative, sequential SR strategy using PySR library (Cranmer, 2023). Treating the concatenated latent embeddings $\omega = [\omega_1, \omega_2] \in \mathbb{R}^8$ as input features and the MLP outputs as pseudo–ground truth, we generate 3000 $(\omega, \text{MLP}(\omega))$ pairs and regress each physical parameter independently. The best-performing symbolic expression is then appended as a new input feature for the next iteration, repeating until all physical parameters are recovered.

Crucially, if these distilled symbolic expressions capture the true underlying physics (and assuming Φ is invertible), they will inherently generalize OOD, this OOD performance thus serves as a rigorous validation test to verify whether SR has genuinely uncovered the true generative mechanism.

To evaluate this, we perform SR on the trained CAMEL+MLP pipeline and assess its predictive accuracy in a strictly OOD test regime bounded by $\theta \in [2.0, 2.5]$ kg $\times$ $[1.5, 1.9]$ m $\times$ $[2.0, 2.5]$ kg $\times$ $[1.5, 1.9]$ m. Table 2 details the resulting symbolic expressions and their OOD extrapolation performance (evaluated over 100 test trajectories for a single random seed), clearly demonstrating the generalization advantage of symbolic distillation over standard MLP

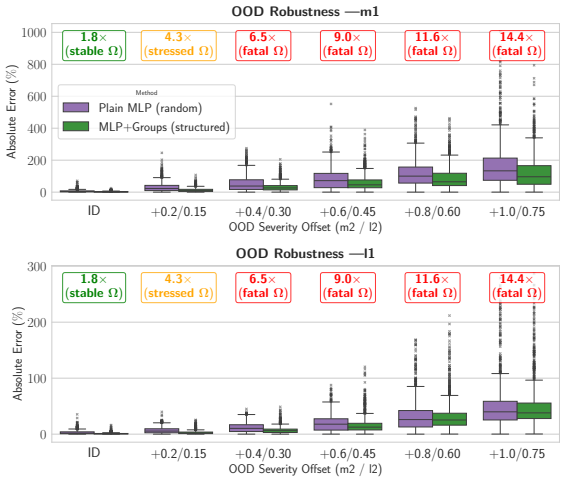

Figure 3. Nuisance robustness across OOD regimes. Boxplots show parameter estimation error (MAPE) for target parameters $[m_1, l_1]$ under progressively severe OOD shifts in unobserved nuisances $[m_2, l_2]$ comparing plain MLP with the structured grouped training variants. *Drift* values (colored annotations) categorize sub-regimes. Results over 10 seeds, 100 test tasks per OOD level.

predictions.

We stress that the interpretability delivered by SR operates at two distinct levels. At the *expression level*, the distilled formulas are closed-form algebraic functions of the latent components ω, which are not directly interpretable as physical quantities but that relate to the actual $\Phi(\varphi)$ thanks to the CAMEL starting framework. At the *structural level*, however, the sequential distillation order, recovering l_1, l_2 before m_1, m_2, mirrors the physical entanglement in the true dynamics (Eq. 9), and the strong OOD extrapolation performance provides formal evidence that the learned mapping faithfully inverts $\Phi(\varphi)$. The primary gain of SR over the MLP is therefore compactness, functional stability, and OOD robustness, rather than human-readable physical laws in the classical sense.

5.5. Nuisance robustness via contrastive groups

To evaluate our grouped training strategy under unobserved nuisance variations, we design a controlled comparison against a classic MLP baseline. We generate $N = 40$ anchor groups, each containing $K = 10$ nuisance variants, ensuring the total training trajectories ($N \times K$) perfectly match the random-sampling baseline. The target parameters are $\varphi_{\text{tgt}} = [m_1, l_1]$, while the unobserved nuisances are $\varphi_{\text{nui}} = [m_2, l_2]$. Both MLPs share identical architectures and hyperparameters; they differ solely in batch structure.

While training bounds match the standard ID ranges, we evaluate across ID and progressively severe OOD shifts in φ_{nui}, utilizing five test regimes scaling from $[2.0, 2.2] \times [1.5, 1.65]$ up to $[2.8, 3.0] \times [2.1, 2.25]$. To rigorously quantify how these nuisance shifts distort the meta-learned embeddings ω, we introduce a practical diagnostic *Drift* metric. *Drift* is defined as the ratio of the expected Mahalanobis distance D_M of the test embeddings from the training centroid μ over that of the training embeddings from the same centroid:

$$\textit{Drift} = \frac{\frac{1}{N_{\text{test}}} \sum_{j=1}^{N_{\text{test}}} D_M(\omega_{\text{test}}^{(j)})}{\frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} D_M(\omega_{\text{train}}^{(i)})}, \tag{10}$$

where $D_M(\omega) = \sqrt{(\omega - \mu)^\top \Sigma^{-1} (\omega - \mu)}$, with Σ denoting the sample covariance of the training embeddings. This scalar measures how severely OOD tasks push the latent representations off the learned training manifold. We categorize this shift into three sub-regimes: *Stable* (*Drift* < 2), representations remain near the training manifold; *Stressed* ($2 \leq$ *Drift* < 5), forcing significant extrapolation; *Fatal* (*Drift* ≥ 5), representations diverge drastically from the training support.

Figure 3 visualizes performance across these regimes. As hypothesized, the MLP trained over contrastive groups consistently outperforms the naive MLP. Crucially, grouped training achieves its most significant relative improvements exactly at the critical boundary between the *Stressed* and *Fatal* regimes. However, deeper into the *Fatal* regime, the covariate shift becomes so extreme that the CAMEL embeddings are pushed completely off the learned training manifold. As a result, they lose their physical correlation to the target parameters entirely, leading to high-variance performance that fluctuates across initialization seeds. In contrast, within the *Stable* and *Stressed* regimes, grouped training delivers consistent, robust improvements across all tested seeds, confirming the method's reliability under moderate extrapolation (see Appendix B.2 for extended details).

6. Conclusions

We have presented an extension of the CAMEL meta-learning framework for amortized physical parameter estimation in dynamical systems. By exploiting CAMEL's theoretical reliability guarantees, we demonstrated that the learned task-specific embeddings encode a projection of the nonlinear parameter manifold $\Phi(\varphi)$. This enables a post-hoc neural mapping $\omega \mapsto \varphi$ that yields physically meaningful parameter recovery, remaining competitive with classical baselines (IDIM-LS) while achieving a $\sim 1000\times$ speedup in inference cost. Furthermore, we leveraged this reliable latent space to distill interpretable algebraic expressions via SR, significantly increasing model transparency and enabling superior OOD extrapolation. We also showed that combining domain-randomized training with amortized inference yields parameter estimates that remain stable under realistic measurement noise levels. Finally, we addressed the challenge of unobserved nuisance variables by intro-

Table 2. Out-of-distribution ($\theta \in [2.0, 2.5]\,\text{kg} \times [1.5, 1.9]\,\text{m} \times [2.0, 2.5]\,\text{kg} \times [1.5, 1.9]\,\text{m}$) extrapolation performance comparing the post-hoc MLP to the distilled SR expressions. The SR expressions use components of the learned latent vector $\omega = [\omega_1, \omega_2] \in \mathbb{R}^8$.

SR parameter expression	**SR MAE ($\pm$ std)**	**SR MAPE (%)**	**MLP MAPE (%)**
$l_{1,sr} = -0.487 - \frac{2.142}{\frac{\omega_5+0.934\omega_7}{\omega_6-1.081\omega_7}-0.552}$	0.0098 ± 0.0011	0.57	15.25
$l_{2,sr} = \left(0.443 + \frac{1.575\omega_5-\omega_6}{1.676(\omega_5+\omega_7)-\omega_6}\right)^2 - 0.044$	0.0135 ± 0.0064	0.76	12.67
$m_{1,sr} = \frac{0.484\omega_0+\omega_2-0.674\left(\omega_4-\omega_5(0.018l_{1,sr}-0.074)-0.909(-1.747m_{2,sr}l_{1,sr}+\omega_3)\right)}{\frac{l_{1,sr}}{0.0077l_{1,sr}+0.927}}$	0.0409 ± 0.0158	1.13	6.31
$m_{2,sr} = \frac{\omega_6-\omega_7}{l_{2,sr}\left(l_{1,sr}+0.012l_{2,sr}^3+0.423\right)\left(1.202-\frac{0.016}{(l_{2,sr}+0.390)^6}\right)}$	0.0033 ± 0.0024	0.15	5.24

ducing a structured grouped training strategy alongside a principled *Drift* metric. This approach significantly improves nuisance robustness under moderate covariate shift without requiring auxiliary losses or structural penalties.

Key limitations of our current formulation include the upstream entanglement of nuisances in the initial CAMEL embeddings (which post-hoc adjustments cannot fully resolve), the requirement that nuisances are present during the meta-training phase (not always satisfied in sim-to-real settings), and validation primarily restricted to simulation. While preliminary results on other nonlinear systems are encouraging (Appendix C), broader evaluation on real-world experimental data remains an active area for future work. A discussion of the limitations is provided in Appendix A.

Impact Statement

This paper presents work whose goal is to advance the field of scientific machine learning, specifically amortized inference and interpretable parameter estimation for physical systems. The primary application context is simulation-based robotics and engineering system identification. We do not foresee direct near-term ethical concerns specific to this work; the broader societal consequences are those generally associated with advancing machine learning methodology. Potential downstream uses in autonomous systems warrant standard considerations around deployment safety and validation on real hardware before use in safety-critical settings.

Acknowledgements

This work was supported as a part of NCCR Automation, a National Centre of Competence in Research, funded by the Swiss National Science Foundation (grant number 51NF40_225155).

References

Alsing, J., Charnock, T., Feeney, S., and Wandelt, B. Fast likelihood-free cosmology with neural density estimators and active learning. *Monthly Notices of the Royal Astronomical Society*, 488(3):4440–4458, 2019.

Blanke, M. and Lelarge, M. Interpretable meta-learning of physical systems. In *ICLR 2024-The Twelfth International Conference on Learning Representations*, 2024.

Brehmer, J. and Cranmer, K. Simulation-based inference methods for particle physics. In *Artificial Intelligence for High Energy Physics*, pp. 579–611. World Scientific, 2022.

Brunot, M., Janot, A., Young, P. C., and Carrillo, F. An improved instrumental variable method for industrial robot model identification. *Control Engineering Practice*, 74: 107–117, 2018.

Brunton, S. L., Proctor, J. L., and Kutz, J. N. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the national academy of sciences*, 113(15):3932–3937, 2016.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

Cranmer, K., Brehmer, J., and Louppe, G. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020a.

Cranmer, M. Interpretable machine learning for science with pysr and symbolicregression. jl. *arXiv preprint arXiv:2305.01582*, 2023.

Cranmer, M., Sanchez Gonzalez, A., Battaglia, P., Xu, R., Cranmer, K., Spergel, D., and Ho, S. Discovering symbolic models from deep learning with inductive biases. *Advances in neural information processing systems*, 33: 17429–17442, 2020b.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pp. 1126–1135. PMLR, 2017.

Fossen, T. I. *Handbook of marine craft hydrodynamics and motion control*. John wiley & sons, 2011.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., March, M., and Lempitsky, V. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

Gautier, M. Numerical calculation of the base inertial parameters of robots. *Journal of robotic systems*, 8(4):485–506, 1991.

Gautier, M. and Khalil, W. Direct calculation of minimum set of inertial parameters of serial robots. *IEEE Transactions on robotics and Automation*, 6(3):368–373, 1990.

Gautier, M., Janot, A., and Vandanjon, P.-O. A new closed-loop output error method for parameter identification of robot dynamics. *IEEE Transactions on Control Systems Technology*, 21(2):428–444, 2012.

Hospedales, T., Antoniou, A., Micaelli, P., and Storkey, A. Meta-learning in neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):5149–5169, 2021.

Kirchmeyer, M., Yin, Y., Donà, J., Baskiotis, N., Rakotomamonjy, A., and Gallinari, P. Generalizing to new physical systems via context-informed dynamics model. In *International conference on machine learning*, pp. 11283–11301. PMLR, 2022.

Leboutet, Q., Roux, J., Janot, A., Guadarrama-Olvera, J. R., and Cheng, G. Inertial parameter identification in robotics: A survey. *Applied Sciences*, 11(9):4303, 2021.

Lueckmann, J.-M., Goncalves, P. J., Bassetto, G., Öcal, K., Nonnenmacher, M., and Macke, J. H. Flexible statistical inference for mechanistic models of neural dynamics. *Advances in neural information processing systems*, 30, 2017.

Morel, R., Han, J., and Oyallon, E. Disco: learning to discover an evolution operator for multi-physics-agnostic prediction. In *International Conference on Machine Learning*, pp. 44750–44774. PMLR, 2025.

Rahaman, N., Baratin, A., Arpit, D., Draxler, F., Lin, M., Hamprecht, F., Bengio, Y., and Courville, A. On the spectral bias of neural networks. In *International conference on machine learning*, pp. 5301–5310. PMLR, 2019.

Savitzky, A. and Golay, M. J. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36(8):1627–1639, 1964.

Schmidhuber, J. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. PhD thesis, Technische Universität München, 1987.

Serrano, L., Le Boudec, L., Kassaï Koupaï, A., Wang, T. X., Yin, Y., Vittaut, J.-N., and Gallinari, P. Operator learning with neural fields: Tackling pdes on general geometries. *Advances in Neural Information Processing Systems*, 36: 70581–70611, 2023.

Swevers, J., Ganseman, C., Tukel, D. B., De Schutter, J., and Van Brussel, H. Optimal robot excitation and identification. *IEEE transactions on robotics and automation*, 13(5):730–740, 2002.

Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., and Abbeel, P. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 23–30. IEEE, 2017.

A. Limitations and future work

Several important limitations remain, and they help clarify the next steps required to make the proposed framework more broadly applicable.

First, the current empirical validation remains limited in scope. All the analysis is conducted on a simulated planar 2-DOF robotic arm, which provides a controlled benchmark but does not capture the full complexity of real-world identification problems. Although preliminary experiments on additional simulated systems are encouraging (see Appendix C), stronger evidence is needed to demonstrate that the proposed neural-symbolic amortized inference pipeline generalizes across broader classes of dynamical systems. In particular, validation on real robotic platforms, where sensor noise, friction, delays, and modeling mismatch are unavoidable, would be essential. Beyond robotics, testing the framework in other scientific domains such as fluid dynamics, structural mechanics, or energy systems would help clarify whether the assumptions underlying CAMEL, the grouped nuisance strategy, and the symbolic distillation step remain effective outside robotic inverse dynamics.

Second, the main bottleneck for robustness to unobserved nuisances appears to lie upstream in the CAMEL representation itself, rather than only in the downstream regressor. As shown by the performance degradation in the *Fatal Drift* regimes, a post-hoc MLP can only partially compensate when severe nuisance shifts have already strongly perturbed the latent embedding ω. In other words, if nuisance information is deeply entangled in the meta-learned representation, the downstream predictor is forced to extrapolate from a corrupted latent space and cannot reliably recover the target parameters. A more principled solution would therefore require modifying the CAMEL meta-training objective itself, for example by explicitly encouraging nuisance-invariant representations or by constraining the latent space to separate target-relevant and nuisance-dependent factors already during meta-learning.

Third, the current grouped training strategy inherently relies on observing variations of the nuisance factors during the training phase. The estimator learns to be invariant precisely because it sees these nuisances change across different training groups. Consequently, this excludes a more challenging but practically relevant scenario: when a nuisance factor is completely static or absent during training, but suddenly varies after deployment (as is common in zero-shot sim-to-real transfer). In such settings, where the model has no prior exposure to this variability to learn from, the current framework offers no formal guarantee of robustness.

Fourth, robustness to measurement noise is currently achieved through explicit domain randomization and trajectory filtering, which implicitly assume that the expected noise scale is approximately known beforehand. While the experiments show that the method degrades gracefully under moderate train-test mismatch in the noise level, this remains a relatively favorable setting. In many real applications, noise is heteroscedastic, time-varying, or not well modeled by additive Gaussian perturbations. A more realistic extension would therefore be to develop a noise-agnostic variant of the framework by incorporating an adaptive uncertainty model that estimates the noise characteristics jointly with the physical parameters.

Finally, the symbolic regression stage itself introduces an additional limitation. Although the extracted expressions improve interpretability and can yield better OOD extrapolation than a pure black-box regressor, their quality still depends strongly on the fidelity of the upstream MLP and on the complexity of the latent-to-parameter mapping. If the learned neural mapping is only approximately correct, symbolic regression may produce compact but only locally valid formulas, rather than recovering the true physical law. This suggests that future work should investigate tighter integration between amortized inference and symbolic discovery, so that interpretability is not treated purely as a post-hoc compression step, but as part of the training objective itself, to robustify its validation.

B. Additional analysis on unobserved nuisances variable

B.1. Extended analysis including Contrastive loss

The contrastive model is trained on the grouped data described in the main paper. For each anchor i, the target parameter vector φ^i_{tgt} is kept fixed, while the nuisance variables vary across K group instances $\omega^{(i,k)}$, $k = 1, \ldots, K$. The goal is to learn a representation and predictor that are invariant to these nuisance changes while remaining predictive of the anchor target.

To enforce this behavior, we combine a supervised anchor-matching loss with an intra-anchor consistency regularizer.

Table 3. Mean OOD parameter estimation error (%) across all severity levels, with standard deviation. Best mean values are in **bold** and second-best mean values are underlined.

Target	Severity	Plain MLP	MLP+Groups	Contrastive
m_1	ID	6.34 ± 7.70	$\mathbf{2.31} \pm 2.54$	$\underline{2.39} \pm 2.66$
	$+0.2/0.15$	31.43 ± 31.20	$\underline{12.74} \pm 12.30$	$\mathbf{11.78} \pm 11.53$
	$+0.4/0.30$	54.72 ± 51.11	$\underline{32.77} \pm 28.17$	$\mathbf{31.44} \pm 27.82$
	$+0.6/0.45$	83.54 ± 70.46	$\underline{60.71} \pm 53.52$	$\mathbf{60.18} \pm 52.86$
	$+0.8/0.60$	118.15 ± 86.85	$\mathbf{89.77} \pm 74.86$	$\underline{91.94} \pm 75.19$
	$+1.0/0.75$	169.04 ± 146.63	$\underline{127.00} \pm 112.41$	$\mathbf{125.82} \pm 110.26$
l_1	ID	3.11 ± 3.65	$\underline{1.05} \pm 1.44$	$\mathbf{1.00} \pm 1.44$
	$+0.2/0.15$	6.82 ± 5.61	$\underline{3.04} \pm 3.36$	$\mathbf{2.92} \pm 2.98$
	$+0.4/0.30$	11.75 ± 8.67	$\mathbf{7.13} \pm 6.25$	$\underline{7.21} \pm 6.04$
	$+0.6/0.45$	20.00 ± 15.63	$\mathbf{16.34} \pm 15.81$	$\underline{16.75} \pm 15.89$
	$+0.8/0.60$	$\underline{32.44} \pm 29.00$	32.48 ± 29.19	$\mathbf{32.16} \pm 29.79$
	$+1.0/0.75$	$\underline{50.29} \pm 45.03$	51.83 ± 44.99	$\mathbf{49.11} \pm 46.74$

Specifically, letting g_ϑ denote the predictor, we minimize the usual loss from Eq. (7) together with the variance penalty

$$\mathcal{L}_{\text{cons}}(\vartheta) = \frac{1}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} \left\| g_\vartheta\left(\omega^{(i,k)}\right) - \frac{1}{K} \sum_{k'=1}^{K} g_\vartheta\left(\omega^{(i,k')}\right) \right\|^2 .$$

The final objective is

$$\mathcal{L} = \mathcal{L}_{\text{grouped}}(\vartheta) + \lambda_{\text{cons}}\, \mathcal{L}_{\text{cons}}(\vartheta).$$

The consistency term encourages predictions associated with the same anchor to agree across all nuisance variations, thereby promoting nuisance-invariant representations without requiring explicit environment labels. In our implementation, λ_{cons} is ramped up gradually during training to stabilize optimization until it reaches a final value of 1.

Results We evaluated the different approaches in an ablation study, comparing: (i) a Plain MLP trained on randomly sampled tasks, and (ii) two variants trained on the grouped data from the main paper: a standard MLP (MLP+Groups) and a Contrastive model. The experimental protocol mirrors the main text: we use $N = 40$ anchor groups with $K = 10$ nuisance variations each, and evaluate on the same OOD grid over 100 trajectories and 10 random seeds.

Figure 4 and Table 3 report the mean and standard deviation of the parameter MAPE across methods. While contrastive regularization provides modest improvements at specific OOD levels, the results do not show a consistent or significant gains over the baseline MLP trained only with grouped data. The performance margins between the two grouped methods are minimal. This suggests that the proposed grouping scheme alone is sufficient to achieve strong nuisance robustness, and that more complex invariance mechanisms bring limited additional benefit in this setting.

B.2. *Drift* prior estimation: Stable vs. Stressed vs. Fatal

To validate that the proposed *Drift* metric is an accurate measure of how the latent parameter distribution shifts away from the training distribution, and that it meaningfully separates three operating regimes, we report per-seed performance for a plain MLP and for an MLP trained with the grouped strategy. We construct environments by slicing the nuisance parameters bounds of φ_{nui} into ranges $[0.5, 2.0] \times [0.5, 1.5]$, $[2.0, 2.2] \times [1.5, 1.65]$, and $[2.2, 2.4] \times [1.65, 1.8]$, which correspond to the Stable, Stressed, and Fatal regimes, respectively, according to the average *Drift* ratio.

As shown in Table 4, MLP+Groups consistently outperforms the plain MLP in estimating m_1 across all 10 random seeds in both the Stable and Stressed regimes. Although the plain MLP occasionally achieves a lower MAPE on individual seeds under the Fatal regime, the grouped training approach remains superior on average. This seed-level analysis aligns with the aggregate results previously reported in Table 3.

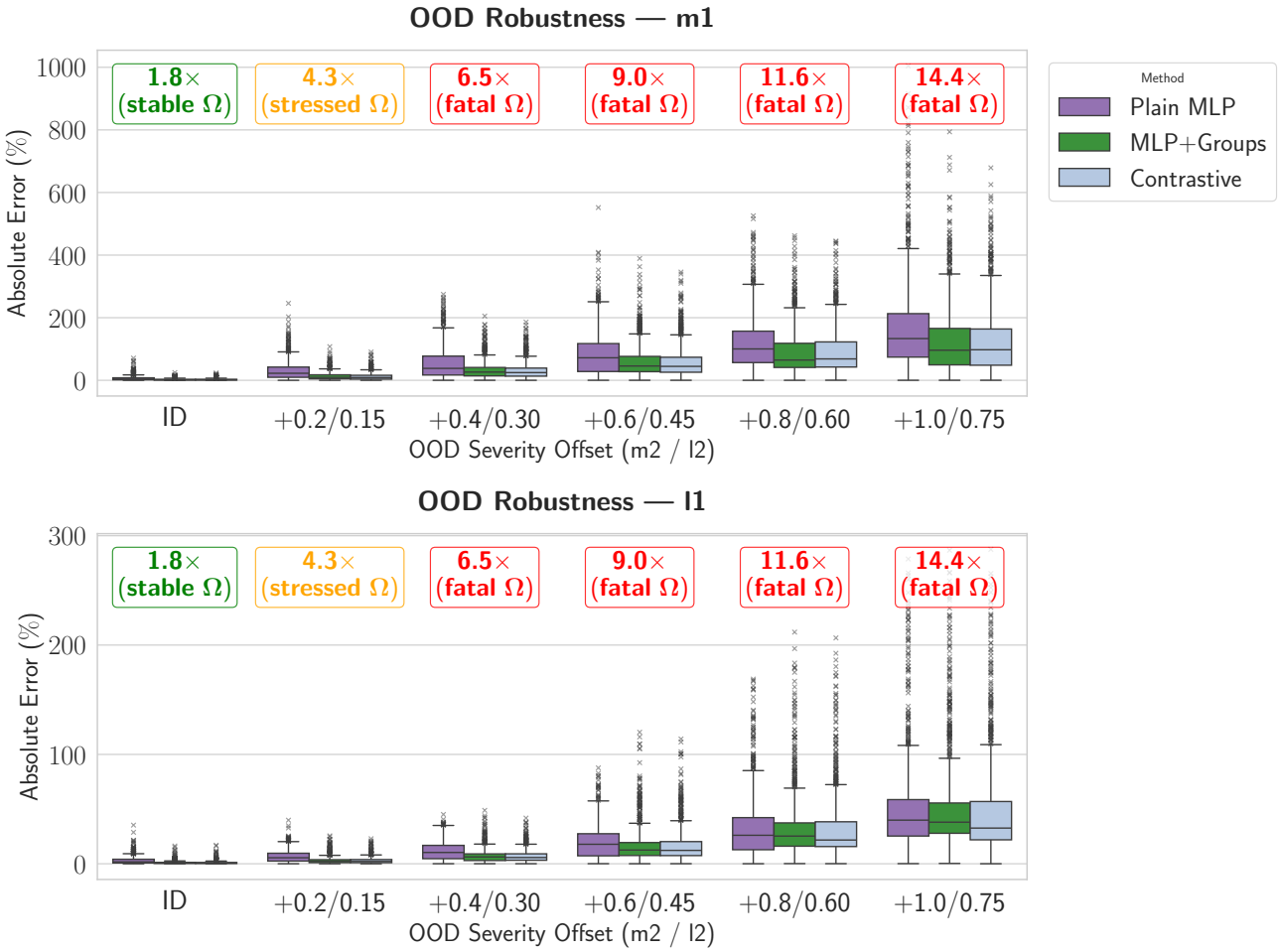

Figure 4. Nuisance robustness across OOD regimes. Boxplots show parameter estimation error (MAPE) for target parameters $[m_1, l_1]$ under progressively severe OOD shifts in unobserved nuisances $[m_2, l_2]$ extending the comparison to include the contrastive loss variant. *Drift* values (colored annotations) categorize sub-regimes. Results over 10 seeds, 100 test tasks per OOD level.

C. Additional Non-Robotic Domain: Marine Surface Vessel

As a second physical domain, we consider a 3-DOF autonomous surface vehicle (ASV) operating in surge (u), sway (v), and yaw (r). The equations of motion follow the standard maneuvering model (Fossen, 2011). While this serves as a control-oriented approximation rather than a high-fidelity hydrodynamic representation, it is widely used in practice for preliminary design and system identification:

$$M_{\mathrm{RB}}\,\dot{\nu} + C_{\mathrm{RB}}(\nu)\,\nu + M_A\,\dot{\nu} + C_A(\nu)\,\nu + D(\nu)\,\nu = \tau, \tag{11}$$

where $\nu = [u, v, r]^\top$ is the body-fixed velocity vector and $\tau = [\tau_u, \tau_v, \tau_r]^\top$ is the vector of generalized forces and moments.

Physical parameters and nonlinear regressor structure. The four underlying geometric parameters are $\vartheta = [L,\, B,\, d,\, C_B]^\top$, representing the vessel's length, beam, draft, and block coefficient, respectively. These parameters enter into nine regressors $\{\phi_i\}_{i=1}^{9}$ through a chain of nonlinear intermediate quantities. The rigid-body mass and moment of inertia are:

$$m = \rho\, LBd\, C_B, \qquad I_z = \frac{m\,(L^2 + B^2)}{12}, \tag{12}$$

where $\rho = 1000\,\mathrm{kg/m^3}$ is the water density. The hydrodynamic added-mass coefficients are approximated by empirical ellipsoid formulas (Fossen, 2011):

$$X_{\dot{u}} = -0.05\,m, \qquad Y_{\dot{v}} = -\frac{1}{2}\rho\pi d^2 L, \qquad N_{\dot{r}} = -\frac{1}{10}\rho\pi d^2 L^3. \tag{13}$$

Table 4. Per-seed OOD summary for m_1 (*Stable*, *Stressed*, *Fatal*). Entries are mean ± std and median % error over test tasks. Bold indicates the better method (lower median; if tied, lower mean).

		Plain MLP		MLP+Groups	
Seed	Regime (*Drift*)	Mean ± Std	Median	Mean ± Std	Median
42	Stable (1.79)	6.10 ± 7.04	4.37	$\mathbf{1.87 \pm 2.43}$	**0.71**
	Stressed (4.32)	11.50 ± 8.92	9.94	$\mathbf{4.68 \pm 4.52}$	**3.75**
	Fatal (6.57)	44.73 ± 32.31	34.16	$\mathbf{20.96 \pm 17.61}$	**14.44**
43	Stable (1.67)	5.54 ± 6.53	3.00	$\mathbf{1.64 \pm 1.72}$	**0.99**
	Stressed (2.70)	29.98 ± 21.36	26.23	$\mathbf{9.69 \pm 7.56}$	**7.46**
	Fatal (3.83)	59.87 ± 31.75	54.09	$\mathbf{29.67 \pm 15.17}$	**29.68**
44	Stable (2.23)	5.95 ± 6.80	4.78	$\mathbf{1.01 \pm 0.81}$	**0.82**
	Stressed (4.90)	21.70 ± 28.82	12.78	$\mathbf{17.22 \pm 26.92}$	**6.71**
	Fatal (7.20)	81.78 ± 69.11	72.54	$\mathbf{61.19 \pm 55.69}$	**54.54**
45	Stable (1.27)	3.94 ± 2.79	3.42	$\mathbf{1.29 \pm 0.99}$	**0.88**
	Stressed (3.46)	21.08 ± 19.74	18.40	$\mathbf{15.71 \pm 13.70}$	**11.97**
	Fatal (4.79)	$\mathbf{45.80 \pm 40.59}$	**30.97**	52.99 ± 32.71	50.52
46	Stable (1.53)	7.27 ± 8.63	4.68	$\mathbf{1.48 \pm 1.27}$	**1.18**
	Stressed (3.69)	23.28 ± 34.10	9.65	$\mathbf{11.06 \pm 9.77}$	**8.17**
	Fatal (5.78)	$\mathbf{24.89 \pm 33.12}$	**16.71**	52.15 ± 34.61	43.33
47	Stable (2.45)	6.64 ± 7.22	3.69	$\mathbf{1.74 \pm 1.60}$	**1.12**
	Fatal (5.11)	54.58 ± 43.04	48.74	$\mathbf{16.47 \pm 18.36}$	**9.88**
	Fatal (7.77)	98.54 ± 56.82	106.20	$\mathbf{28.13 \pm 28.95}$	**16.96**
48	Stable (1.49)	3.90 ± 3.30	3.17	$\mathbf{1.53 \pm 1.54}$	**0.88**
	Stressed (3.50)	25.21 ± 17.37	22.48	$\mathbf{13.16 \pm 14.99}$	**10.14**
	Fatal (5.63)	64.12 ± 50.96	52.07	$\mathbf{59.20 \pm 47.78}$	**54.66**
49	Stable (2.01)	4.72 ± 5.17	2.67	$\mathbf{0.96 \pm 1.16}$	**0.71**
	Stressed (4.69)	22.63 ± 15.66	17.73	$\mathbf{6.62 \pm 6.16}$	**4.30**
	Fatal (9.80)	32.82 ± 26.91	28.47	$\mathbf{7.03 \pm 7.45}$	**5.70**
50	Stable (1.74)	5.36 ± 5.04	4.22	$\mathbf{1.77 \pm 1.43}$	**1.59**
	Fatal (5.01)	26.02 ± 20.33	20.26	$\mathbf{9.93 \pm 7.50}$	**8.68**
	Fatal (7.82)	48.15 ± 40.10	34.05	$\mathbf{20.02 \pm 16.61}$	**16.10**
51	Stable (1.53)	4.94 ± 7.18	2.44	$\mathbf{1.13 \pm 1.23}$	**0.65**
	Stressed (3.92)	14.19 ± 10.44	12.19	$\mathbf{9.22 \pm 8.72}$	**6.62**
	Fatal (5.68)	$\mathbf{22.12 \pm 22.07}$	**16.46**	43.72 ± 36.19	34.42

The nine regressors are then explicit nonlinear functions of ϑ:

$$\phi_1 = 1.05\,\rho\,LBdC_B, \tag{14}$$

$$\phi_2 = \rho\,LBdC_B + \tfrac{1}{2}\rho\pi d^2 L, \tag{15}$$

$$\phi_3 = \frac{\rho\,LBdC_B(L^2+B^2)}{12} + \tfrac{1}{10}\rho\pi d^2 L^3, \tag{16}$$

$$\phi_4 = \tfrac{1}{2}\rho\,LBdC_B, \quad \phi_5 = \rho\,LBdC_B, \quad \phi_6 = \frac{\rho\,LBdC_B(L^2+B^2)}{24}, \tag{17}$$

$$\phi_7 = \tfrac{1}{2}\rho\,Bd, \quad \phi_8 = \tfrac{1}{2}\rho\,Ld, \quad \phi_9 = \tfrac{1}{10}\rho\,L^3 d. \tag{18}$$

These regressors govern the inverse dynamics formulation as follows:

$$\tau_u = \phi_1\,\dot{u} - \phi_2\,vr + \phi_4\,u + \phi_7\,u\,|u|, \tag{19}$$

$$\tau_v = \phi_2\,\dot{v} + \phi_1\,ur + \phi_5\,v + \phi_8\,v\,|v|, \tag{20}$$

$$\tau_r = \phi_3\,\dot{r} + (\phi_2 - \phi_1)\,uv + \phi_6\,r + \phi_9\,r\,|r|. \tag{21}$$

The mapping $\vartheta \mapsto \phi$ is therefore nonlinear and involves up to cubic monomials in the geometric parameters (e.g. ϕ_3 contains L^3 and L^2BdC_B terms), making direct inversion analytically intractable. This is exactly analogous to the role of the link

parameters (m_1, m_2, l_1, l_2) in the robotic arm: the regressors are observable, but the underlying physical mechanisms are not.

Because each torque component is affine in a distinct subset of the basis functions, we train three independent CAMEL models (with embedding dimensions $r = [4, 4, 4]$) and concatenate their latent representations before passing them to the post-hoc MLP. To generate trajectory data, we apply a persistently exciting multi-sinusoidal input across all three degrees of freedom, combined with a constant forward surge bias of 1.0 m/s to simulate realistic vessel operation. All tasks share this excitation profile and vary only in their underlying geometric parameters.

C.1. Results

We adopt the same experimental protocol as the main paper: $T = 400$ training trajectories of $N = 400$ samples each, evaluated on 100 held-out tasks over 10 independent seeds. Table 5 reports in-distribution identification performance for both the post-hoc MLP and the distilled SR expressions.

Table 5. In-distribution identification performance for the marine surface vessel. Results report the mean $\pm$ standard deviation evaluated across 10 independent seeds.

Parameter	**MLP MAE**	**SR MAE**	**MLP MAPE (%)**	**SR MAPE (%)**
Length L	0.0123 ± 0.01	0.0297 ± 0.0211	0.8 ± 0.7	2 ± 1.5
Beam B	0.0040 ± 0.0045	0.0053 ± 0.0037	1.0 ± 1.4	1.2 ± 0.9
Draft d	0.0018 ± 0.0013	0.0057 ± 0.004	0.9 ± 0.9	2.8 ± 1.9
Block coeff. C_B	0.0059 ± 0.0053	0.0137 ± 0.0106	0.8 ± 0.7	1.9 ± 1.4

Despite this being a preliminary experiment on a domain structurally different from robotics, the results are highly encouraging on two fronts. First, the MLP consistently achieves sub-percent MAPE across all four parameters, confirming that CAMEL's latent embeddings carry sufficient information for accurate identification well beyond the robotic setting. Second, the SR expressions remain competitive, demonstrating that the latent-to-parameter mapping retains an interpretable structure even for a hydrodynamic system driven by cubic nonlinearities in the underlying parameters (e.g. $\phi_3 \propto L^3 d$). Overall, these preliminary findings suggest that our proposed framework, extending CAMEL to capture nonlinear relations of physical parameters, generalizes naturally to a broader class of physical systems where structured parameter identification is of interest.