Diffusion Sequence Models for Generative In-Context Meta-Learning of Robot Dynamics

Gunes Cagin Aydin[1], Angelo Moroncelli[2], Matteo Rufolo[2], Asad Ali Shahid[1,2], Loris Roveda[1,2]

***Abstract*— Accurate modeling of robot dynamics is essential for model-based control, yet remains challenging under distributional shifts and real-time constraints. In this work, we formulate system identification as an in-context meta-learning problem and benchmark deterministic and generative sequence models for forward dynamics prediction.**

We take a Transformer-based meta-model, as a strong deterministic baseline, and introduce two complementary diffusion-based approaches: (i) joint trajectory modeling (Diffuser), which learns the full action–state distribution, and (ii) conditioned diffusion models (CNN and Transformer), which generate future states conditioned on control inputs.

Through large-scale randomized simulation, we analyze performance across in-distribution and out-of-distribution regimes, as well as computational trade-offs relevant for control. We show that diffusion models significantly improve robustness under distribution shift, with the joint formulation achieving the highest accuracy.

Finally, we demonstrate that warm-started sampling enables diffusion models to operate within real-time constraints, making them viable for MPC. These results highlight generative meta-models as a promising direction for robust system identification in robotics. Models checkpoints and datasets are available at `https://robo-meta.github.io`.

I. Introduction

Accurate modeling of system dynamics lies at the core of robot control [1], underpinning applications in model predictive control (MPC) [2], trajectory optimization, and model-based reinforcement learning [3]. However, reliable modeling of real-world robotic systems remains challenging, as classical physics-based methods often fail to capture effects such as friction, compliance, and parameter variability [4].

Data-driven approaches offer an appealing alternative by directly learning robot behavior from observations [5]. In particular, black-box models approximate system dynamics as a function of input-output trajectories without requiring explicit parameterization. Despite their flexibility, such methods often suffer from poor generalization, high data requirements, and limited robustness under distributional shifts [6].

Within this landscape, learning-based approaches to robot control can be broadly categorized into three paradigms: (i) policy learning methods, which directly map observations to actions [5]; (ii) world models, which learn latent representations optimized for planning and control [7]; and (iii) explicit dynamics models, which predict future system states and can be integrated into classical control frameworks [8]. While recent advances in diffusion models and large-scale architectures have achieved impressive results in policy generation and trajectory planning [5], these approaches typically bypass explicit modeling of system dynamics. Consequently, a gap remains between advances in generative modeling and the requirements of system identification for control [6].

To address this gap, we investigate system identification through the lens of meta-learning. We do this by adopting an implicit **black-box meta-model** framework for dynamics, casting system identification as an in-context learning problem. This paradigm was initially proposed in [9] and subsequently studied in [10], [11]. More recently, it has been successfully scaled to high-dimensional robotic manipulation tasks [12], [13]. The core premise relies on the in-context learning capabilities of modern neural architectures. Rather than optimizing a separate neural network for every distinct system, the meta-model implicitly learns the governing rules of entire classes of dynamical systems from contextual input-output trajectories. This framework provides a powerful, data-driven mechanism for generalization across similar systems by effectively "learning to learn" [14].

Recently, sequence modeling in this domain has been dominated by Transformers [15]. Transformer-based meta-models such as RoboMorph [12] provide a strong deterministic baseline via in-context learning, but produce uni-modal predictions and degrade under distributional shifts.

Denoising Diffusion Probabilistic Models [16] have recently emerged as stable generative frameworks capable of modeling multi-modal distributions [17]. Despite their success in policy learning, their application to explicit dynamics estimation remains largely unexplored, particularly in meta-learning settings [12], [18].

In this work, as shown in Fig. I, we consider two diffusion-based formulations for system identification: **inpainting diffusion** (*Diffuser*) [19], which models the full action–observation trajectory, and **conditioned diffusion** [5], which predicts future states conditioned on control inputs using CNN or Transformer backbones.

Our results show that deterministic models perform well in simple in-distribution regimes but degrade under distributional shifts, while diffusion models significantly improve robustness, particularly in multi-frequency dynamics. Inpainting diffusion achieves the highest accuracy, whereas conditioned diffusion offers a better trade-off between performance and inference latency. Warm-started sampling further enables real-time deployment, making diffusion models

Corresponding author: `loris.roveda@supsi.ch`.

[1]Politecnico di Milano, Institute of Industrial and Information Engineering, Milano, Italy.

[2]SUPSI-DTI-IDSIA, Dalle Molle Institute for Artificial Intelligence, Lugano, Switzerland.

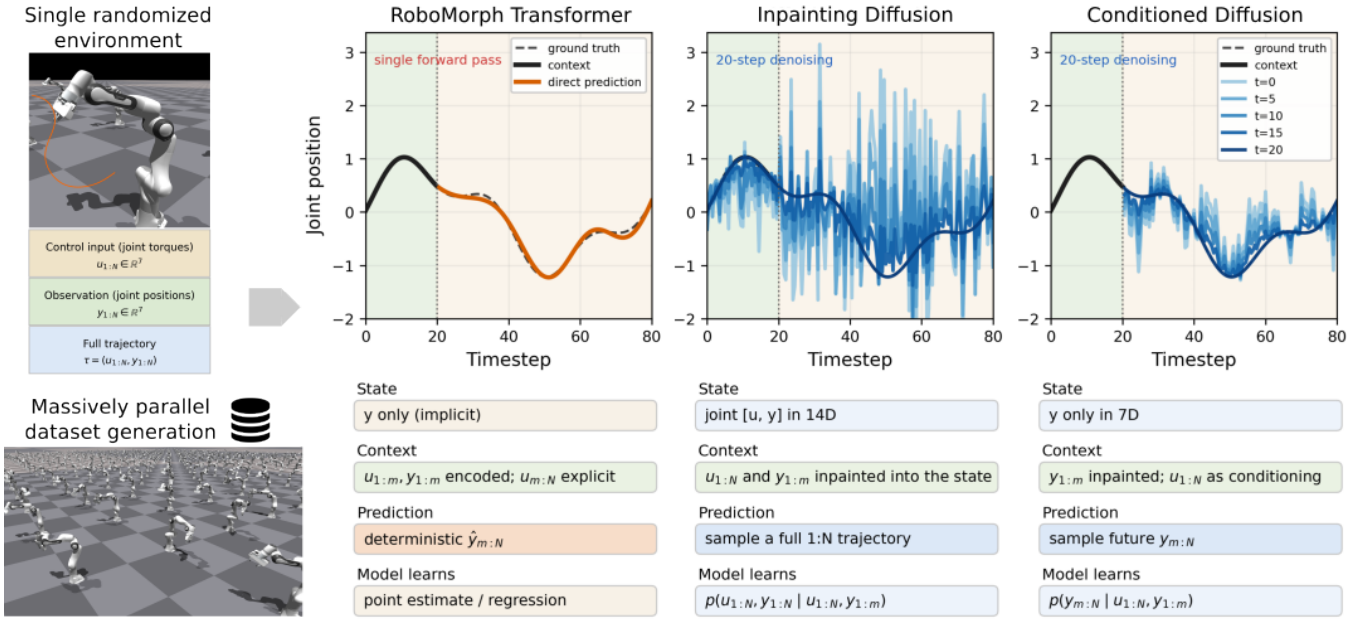

Fig. 1. **Comparison of sequence models.** Diffusion models (*Inpainting* and *Conditioned*) are contrasted with a deterministic Transformer (*RoboMorph*). Inpainting Diffusion learns the full trajectory distribution over $y_{1:N}$ from context. Conditioned Diffusion models system dynamics by generating $y_{m:N}$ conditioned on $u_{1:N}$, producing future trajectories under action conditioning with smoother denoising updates. Diffusion models require multiple iterative steps (e.g., 20 in this example) to progressively denoise an initial random distribution, refining predictions over time. In contrast, the deterministic Transformer predicts $y_{m:N}$ directly from $u_{m:N}$ and context in a single forward pass.

compatible with MPC.

This raises several key questions:

- Can generative models improve robustness and generalization in system identification?
- How do different diffusion formulations compare to deterministic architectures in modeling complex robot dynamics?
- What are the trade-offs between accuracy, uncertainty representation, and computational cost in control-oriented settings?

To answer these questions, we construct a unified experimental framework for meta-learned robot dynamics across a large distribution of randomized simulated systems and control tasks.

The main contributions of this work are:

- We extend prior work [12] with a large-scale randomized study over physical parameters, excitation signals, and initial conditions.
- We introduce two diffusion-based formulations: (i) inpainting diffusion (Diffuser), and (ii) conditioned diffusion (CNN and Transformer) for dynamics modeling.
- We benchmark these approaches against a strong deterministic baseline (RoboMorph), analyzing in-distribution accuracy and out-of-distribution robustness for different signals and frequencies.
- We study the trade-off between accuracy and latency, showing that conditioned diffusion enables MPC-compatible inference via warm-starting.

II. Problem Description

In this section, we formalize our meta-learning framework and motivate our architectural choices, domain randomization strategy, and training procedures.

Classical modeling of robotic dynamics relies on deriving a faithful mathematical representation of a physical plant. Formally, let $\mathscr{S}$ denote a specific physical *system* (e.g., a robotic manipulator with exact, fixed physical parameters) drawn from a broader *system class* $\mathscr{C}$ of similar systems, which represents the family of all such systems under varying physical conditions (e.g. different physical parameters). Because the true analytical equations governing $\mathscr{S}$ are often overly complex or intractable to derive from first principles, it is standard practice to approximate the system's behavior using a data-driven *model*.

When exact prior knowledge about the physical parameterization of $\mathscr{S}$ is unavailable, system identification relies on black-box modeling. This approach is agnostic to the underlying physical equations, instead approximating the true system dynamics via a parameterized function approximator $g(x,\theta)$. To optimize θ, the model is trained on a trajectory dataset $\mathscr{D} = (u_{1:N}, y_{1:N})$, which comprises a finite sequence of control inputs and corresponding system observations generated by exciting the specific physical system $\mathscr{S}$. Depending on the required expressiveness, this model can range from classical linear projections over a set of basis functions to highly non-linear, high-dimensional neural network architectures.

A. Model-Free Black-Box Meta-Models

While traditional black-box models are trained to identify a single, isolated dynamical system, recent advancements have expanded this paradigm to model entire classes of systems [9]. This is achieved by framing system identification as an *in-context learning* problem. In this framework, a neural meta-model $\mathscr{M}_\phi$ is trained directly over a broad trajectory distribution $p(\mathscr{D})$, which jointly encapsulates the variations in underlying physical systems and the corresponding control excitations.

For each sampled trajectory $D \sim p(\mathscr{D})$, we partition the data into a context window of length m, and a prediction horizon from m to N. The context, denoted as $D_{ctx} = (u_{1:m}, y_{1:m})$, provides the necessary historical information to implicitly identify the specific system dynamics. The meta-model is then tasked with predicting the future system response $y_{m:N}$, conditioned on both the context and the future control inputs $u_{m:N}$:

$$\hat{y}_{m:N} = \mathscr{M}_\phi\left(u_{m:N}, D_{ctx}\right). \tag{1}$$

The optimal model parameters ϕ^* are obtained by minimizing the expected prediction loss (e.g., Mean Squared Error) across the entire trajectory space:

$$\phi^* = \arg\min_\phi \mathbb{E}_{D\sim p(\mathscr{D})}\left[\|y_{m:N} - \hat{y}_{m:N}\|_2^2\right]. \tag{2}$$

Transformers, inherently designed for sequence-to-sequence mapping and in-context conditioning [20], serve as a natural architectural baseline for this meta-modeling task.

B. Deterministic vs. Generative Inference

Standard neural architectures, such as baseline Transformers and CNNs trained via the deterministic objective above, regress a single point estimate. Mathematically, they approximate the conditional expectation $\mathbb{E}[y_{m:N} \mid u_{m:N}, \mathscr{D}_{ctx}]$, rendering their outputs inherently uni-modal.

To account for complex model and data-born uncertainties, the meta-model must be framed generatively to explicitly learn the conditional probability distribution of the trajectories. This is achieved by shifting from a deterministic loss to a probabilistic meta-model p_ϕ that maximizes the expected log-likelihood over the trajectory distribution:

$$\phi^* = \arg\max_\phi \mathbb{E}_{\mathscr{D}\sim p(\mathscr{D})}\left[\log p_\phi\left(y_{m:N} \mid u_{m:N}, \mathscr{D}_{ctx}\right)\right]. \tag{3}$$

While standard architectures can be extended into this generative framework (e.g., via Variational Autoencoders), doing so typically requires explicitly defining complex, rigid priors over a reduced latent space, which can overly restrict expressiveness when modeling high-dimensional robotic dynamics.

Among possible parametrization of ϕ, diffusion processes are particularly interesting due to their apparent stability and inference flexibility [21]. A diffusion model gradually noises a trajectory and learns to reverse this process by stable Markov chains [16] of the form $p_\phi(x_0) = \int p_\phi(\tau_{0:T}\ d\tau_{0:T})$ where $0:T$ are the diffusion timesteps. Non-Markovian extensions also exist and provide a leeway between performance and compute [21]. The forward (prior) and reverse (posterior) processes are:

$$\begin{aligned} q(\tau_t|\tau_0) &= \mathscr{N}(\tau_t; \sqrt{\alpha}\tau_0, (1-\alpha_t)I) \\ q(\tau_{t-1}|\tau_t, \tau_0) &= \mathscr{N}(\tau_{t-1}; \mu_\phi(\tau_t, t), \textstyle\sum_\phi(\tau_t, t)) \end{aligned}$$

For action-observation pairs, the model inference relates to the learned denoising model $\hat{y}^{k-1}_{m:N} = \alpha(y^k_{m:N} - \gamma\mathscr{M}_\phi(u^k_{1:N}, y^k_{1:m}) + \mathscr{N}(0, \sigma^2 I))$ where y and/or u are updated at every diffusion timestep $k \in 0:T$.

Diffusion models are naively agnostic of inference directions. Steering the diffusion process to a desirable trajectory, requires goal-conditioned loss functions, inpainting, or conditioning. Details of these different modes are described in the following section.

C. Neural Architectures

We consider sequence models along two dimensions: (i) *deterministic vs. generative inference*, and (ii) *joint vs. conditional trajectory modeling*, as illustrated in Fig. I. This framing enables a unified analysis of expressiveness, robustness, and computational efficiency in meta-learned dynamics.

We adopt standard architectures in robotics, namely Transformers and CNNs [22], instantiated within the meta-learning framework described above. Hyperparameters are selected via extensive ablation.

a) ***Transformer (RoboMorph)****:* RoboMorph [12] serves as a deterministic baseline based on an encoder–decoder Transformer [15]. The context $(u_{1:m}, y_{1:m})$ is encoded and cross-attended with future inputs $u_{m:N}$ to predict $\hat{y}_{m:N}$. While effective in simple in-distribution settings, it performs uni-modal regression, approximating $\mathbb{E}[y_{m:N} \mid u_{m:N}, \mathscr{D}_{ctx}]$, and thus struggles to capture uncertainty, leading to degraded performance under distributional shifts.

b) ***Inpainting Diffusion (Diffuser)****:* To address this limitation, we introduce a generative approach based on diffusion models. The Diffuser [19] models the *joint distribution* over action–state trajectories using a DDPM [16] with a U-Net backbone [22]. Known values $(u_{1:N}, y_{1:m})$ are enforced via inpainting at each denoising step.

By modeling $p(u, y \mid u_{1:N}, y_{1:m})$, Diffuser captures rich control–state correlations, yielding expressive and multi-modal predictions with strong robustness, especially out-of-distribution. This increased expressiveness, however, comes with higher computational cost and greater sensitivity to truncated (warm-started) inference.

c) ***Conditioned Diffusion (CNN and Transformer)****:* We also consider conditioned diffusion, which models $p(y_{m:N} \mid u_{1:N}, y_{1:m})$, reducing the complexity of the generative task. Control inputs are injected via FiLM conditioning [23], following recent approaches for generative policies [5].

We instantiate this formulation with both CNN and Transformer backbones. CNN-based models enforce local temporal smoothness through convolutional filtering, producing physically coherent trajectories, while Transformer-based models capture long-range dependencies but may introduce higher-frequency oscillations due to the lack of local inductive bias. Despite these differences, both variants retain

TABLE I
TASK RANDOMIZATION OF DATASETS

Joint Torques	$A[Nm]$	$f[Hz]$
$\mathscr{D}_1 - CH$	$[-4,4]$	0.3
$\mathscr{D}_2 - CH$	$[-4,4]$	$[0.2,0.4]$
$\mathscr{D}_3 - CH$	$[-4,4]$	$[0.2,0.6]$
$\mathscr{D}_4 - CH$	$[-4,4]$	$[0.1,0.7]$
$\mathscr{D}_1 - MS$	$[-30f,30f]$	0.15
$\mathscr{D}_2 - MS$	$[-30f,30f]$	$[0.05,0.15]$
$\mathscr{D}_3 - MS$	$[-30f,30f]$	$[0.05,0.25]$
$\mathscr{D}_4 - CH$	$[-30f,30f]$	$[0.01,0.30]$

multi-modal expressiveness and benefit from more efficient and stable inference compared to joint diffusion.

Overall, these architectures define a clear trade-off. Deterministic Transformers are fast but brittle under distribution shifts. Joint diffusion maximizes expressiveness and robustness by modeling full trajectory distributions, but is computationally heavier. Conditioned diffusion provides an effective middle ground, achieving strong robustness with significantly improved efficiency. In particular, by simplifying the generative task, conditioned diffusion enables stable warm-started inference, making it more suitable for real-time control, whereas joint diffusion is preferable when modeling accuracy is important.

D. Datasets

We train our black-box meta-models over a wide range of geometric configurations and dynamical parameters of the Franka Emika Panda, using nominal values from previous work [24]. System parameters are randomized and 3×10^5–10^6 robots are simulated in parallel using IsaacGym [25].

For feedforward excitation, joint torques are generated using multi-sinusoidal and chirp signals. The multi-sinusoidal input is defined as $\tau_{MS}(t) = \sum_i A_i \sin(2\pi f_i t + \phi_i)$, where A_i, f_i, and ϕ_i are randomized amplitudes, frequencies, and phases. The chirp excitation is defined as $\tau_{CH}(t) = \sum_i A_i \sin(2\pi \phi_i(t) + \phi_i)$, where $\phi_i(t)$ is a time-varying phase corresponding to a frequency sweep (e.g., linear), inducing an instantaneous frequency $f_i(t)$.

Each dataset consists of 7-dimensional action sequences $u_{1:N}$ (joint torques) and 7-dimensional observations $y_{1:N}$ (joint states), with straightforward extensions to higher-dimensional representations including Cartesian and end-effector dynamics.

All the variables as well as the dynamical parameters are selectively randomized over the course of the training to cover the entire dynamical domain; the nominal task parameters as well as their randomization amounts for each training dataset is available on Table I.

E. Training

Training is exactly the same for all architectures both in feedforward and feedback control and is for 10 epochs over about $3.5e6$ robots of randomized trajectories as shown in Fig. I.

An *MSE* loss is adopted for all cases of the form $\mathscr{L} = MSE(\hat{y}, y)$ for *RoboMorph* and $\mathscr{L} = MSE(\varepsilon^k, w * \varepsilon_\theta^k)$ for diffusion models where $w = [w_u, w_y]$ is the weight mask for actions and observations and ε is the normalized noise estimate. The weight mask is taken to be $[1,3]$ to selectively improve inferences in Diffuser, otherwise $[1,1]$.

RoboMorph and cdTRF are configured to have 12 MLP layers, 8 attention heads and 384 embedding dimensions, following from the hyperparameter study in [12]. Diffuser and cdCNN have 128 initial convolution layers and 3 down-sampling/upsamping steps. Diffuser, cdCNN and cdTRF denoise in 100 steps and are trained to extract the Gaussian noise in noisy trajectories. The hyperparameters for both the convolution-based networks and the diffusion process were selected following a systematic ablation study to ensure optimal predictive performance for each respective architecture.

F. Properties of Dynamical Models

Here we highlight some of the properties of the approach and analyses inherited mostly from the architectures we are employing.

a) Amount of Prior Information: All dynamical models benefit from increased amount of priors. Functionally, this corresponds to an increase in the context m accounting for a larger portion of the horizon. In model-based predictive applications, this becomes a compromise between inference time latency and prediction accuracy.

b) Diffusion Multi-Modality: Diffusion models are inherently multi-modal Bayesian frameworks. This behavior may be crucial in representing unknown sources of dynamics, essentially proving a more robust estimator.

c) Robustness Against Uncertainties via Generative Modeling: In real-world robotic applications, physical plants are heavily subjected to unmodeled dynamics, varying parameters, and sensor noise. Standard deterministic regressors struggle with these uncertainties, as minimizing the Mean Squared Error over complex, stochastic data fundamentally forces the model to predict the conditional expected value (the mean), which often results in physically invalid or overly smoothed trajectories. Conversely, diffusion models offer inherent robustness by explicitly modeling the conditional probability distribution $p(y_{m:N} \mid u_{m:N}, \mathscr{D}_{ctx})$. Because inference is a true generative sampling process rather than an expected-value regression, diffusion architectures can accurately reconstruct physically viable, multi-peaked trajectory distributions, effectively preserving system fidelity even under significant data-born uncertainties.

d) CNN Inference Time Flexibility: Convolution-based models are adjustable in inference dimensions, which allows for enrolling low-dimensional models in high-dimensional tasks or vice-versa. This allows arbitrarily long (or short) horizon estimations than what the models are subjected to in training. For predictive applications, this behavior may be critical especially as an ailment to the longer inference times of diffusion models.

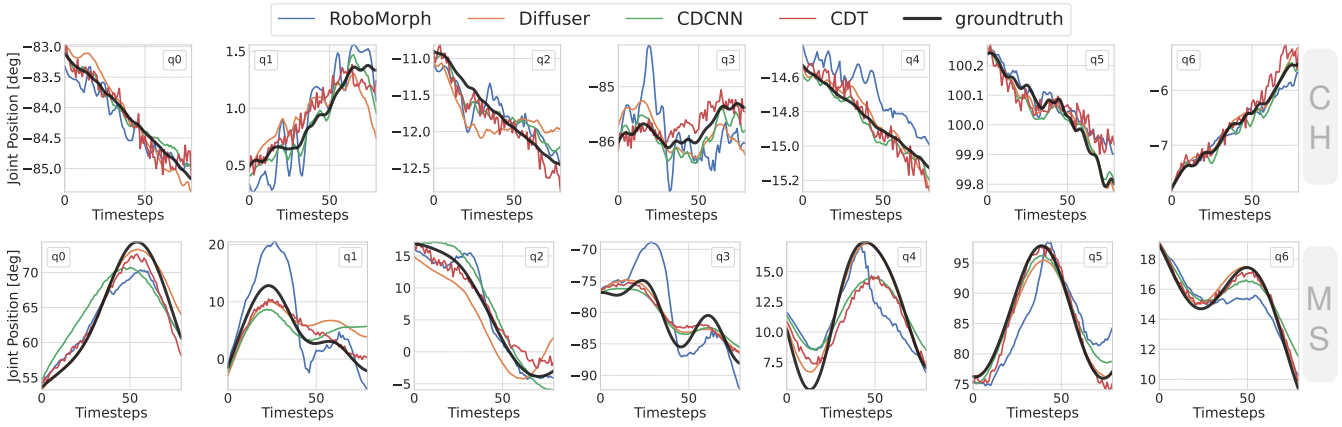

Fig. 2. Meta-modeling of feedforward controlled joint dynamics for chirp and multi-sinusoidal joint torques with $f = 1.0Hz, f = 0.45Hz$ master frequency, respectively: performance of architectures trained on $\mathscr{D}_1$. The results indicate the overall best among 100 randomized scenarios. Diffusion-based architectures are more faithful representations of global complexity especially in out of distribution.

e) Transformer Sequentiality: Transformers are naturally prompt at sequentially processing data which encapsulates the data in global attention mechanisms (until the embedding dimension) as opposed to the local receptive fields in CNNs.

III. Simulation Results and Analysis

In this section we cover how the adopted framework fares in multitudes of simulation scenarios. All the tests analyzed on both in-distribution (ID) and out-of-distribution (OOD) cases. We first focus in simulation performance where we evaluate the accuracy and adaptability of different architectures and provide an analysis on meta-model adaptability when trained with different dataset, secondly we focus on the different architecture inference time and try to make a fair comparison in a feedback control perspective.

A. Simulation Performances

1) Forward Dynamics Meta-Model: Here we consider a large selection of MS and CH signals. CH is naturally easier to portray since CH signals at lower and higher frequencies globally converge to stationary or monotonously increasing trajectories which are trivial tasks for neural networks to learn as in Fig. 2. However, multi-modality of the MS signal, is exacerbated in increased frequencies of about $0.3-0.5Hz$ with jagged dynamics, which poses a realistic challenge not foreseen in CH tasks. This is portrayed on Fig. 2 where sinusoidal signals are more cumbersome at virtually all of the explored domain.

By meta-modeling the forward dynamics, we are able to accurately portray even the most challenging of signals with errors bounded by 3.5 degrees in joint space as on Figs 2 and 2. Nevertheless, not all architectures behave the same. For lower frequencies in MS, RoboMorph is on par with the other models without any statistically significant gap in accuracy. All the architectures, as extensions of the simplest neural networks, are capable of inferring stationary trajectories and hence for low frequencies no architectural efficacy is justifiably attributable. For higher frequencies, mostly evident in MS tasks, RoboMorph performance visibly degrades compared to architectures with a diffusion model which can be attributed to the superior multi-modal generative capabilities that diffusion models profess.

Trajectories predicted by CNN-based models are inherently smooth, whereas Transformer models—most notably CDT—are highly prone to high-frequency oscillatory estimations. This behavior aligns with the fundamental architectural differences between the models. While the Transformer decoder employs causal attention to enforce strict temporal directionality, it still relies on a global self-attention mechanism over the past sequence. Because it lacks a strict inductive bias for local temporal continuity, adjacent time steps can fluctuate independently. Conversely, CNNs possess a strong local inductive bias; their convolutional kernels act as local filters over sliding temporal windows. This explicitly ties adjacent states together, mathematically enforcing temporal coherence and yielding naturally smooth trajectories. In applications where Transformer-induced high-frequency jitter is problematic, applying a standard low-pass denoising filter as a post-processing step effectively recovers physical continuity.

Figure 3 illustrates the predictive performance of the evaluated architectures across varying nominal frequencies, corresponding to the parameter ranges detailed in Table I. The shaded gray regions denote the ID training domains, while the white regions correspond to OOD scenarios. A primary observation is that the RoboMorph experiences severe performance degradation in OOD regimes for both chirp and sinusoidal signals. In contrast, the diffusion-based models exhibit significantly greater robustness, maintaining stable accuracy even well outside the training distribution.

Furthermore, expanding the training dataset from a narrow to a broader frequency domain substantially enhances the predictive accuracy of the deterministic RoboMorph baseline. This behavior highlights a core limitation of standard deterministic meta-learning: robust OOD generalization requires exposing the model to exhaustive dynamical variations, otherwise the framework collapses into narrow,

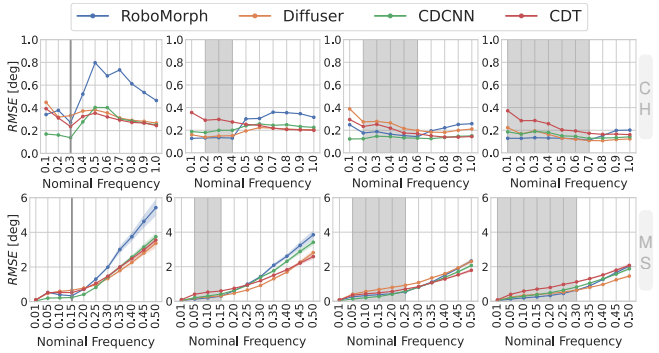

Fig. 3. By selectively covering parts of the domain, it is possible to meta-learn the class of frequency response. For chirp signals of different randomization: $f_{\mathscr{D}_1} = 0.30$, $f_{\mathscr{D}_2} = [0.02, 0.4]$, $f_{\mathscr{D}_3} = [0.2, 0.6]$, $f_{\mathscr{D}_4} = [0.1, 0.7]$ and sinusoidal signals of different randomization bounds $f_{\mathscr{D}_1} = 0.15$, $f_{\mathscr{D}_2} = [0.05, 0.15]$, $f_{\mathscr{D}_3} = [0.05, 0.25]$, $f_{\mathscr{D}_4} = [0.01, 0.30]$ the ID and OOD regions shift which effectively enlarges the accuracy of the predictive domain. This improvement on the edges of the domain minimally affects the accuracy observed for the centering points.

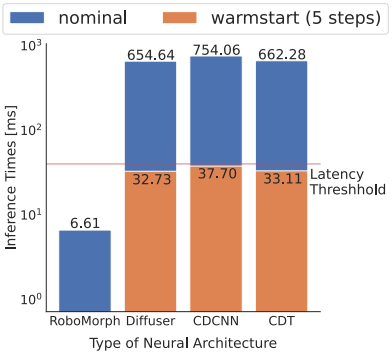

Fig. 4. Diffusion-based inference is about 2 order of magnitude larger than non-diffusion-based inference and is a bottleneck of predictive latency. This can be alleviated by warmstarting diffusion models from past trajectories, bringing the inference time to about 40*ms* with 5 denoising steps.

task-specific memorization. Conversely, this degradation is far less pronounced in the diffusion-based architectures. By explicitly modeling the generative probability distribution rather than regressing a single point estimate, diffusion models inherently capture broader dynamical representations. Consequently, they maintain robust predictive performance even when subjected to limited training diversity. Overall, the diffusion-based architectures consistently outperform the classic RoboMorph across varied scenarios, including several ID cases.

Beyond predictive accuracy, it is necessary to consider the computational trade-offs. While transformers scale effectively with large datasets, choosing appropriate hyperparameters and stabilizing the optimization of their loss functions remain persistently challenging. In our experiments, transformer-based models required, on average, four times longer to train than their CNN counterparts. However, this offline training cost is offset by substantially faster online inference speeds. As shown in Fig. 4, this accelerated inference allows transformer-based models to be readily deployed in real-time control scenarios, and specific sampling techniques can be further employed to minimize the computational gap during execution.

2) Inference Comparison in a control perspective: Inference latency in diffusion models is inherently high, as they require a full forward pass at every denoising timestep. Naively reducing the number of timesteps during training degrades prediction accuracy. Instead, we adopt a more flexible strategy: we train on a dense diffusion schedule and accelerate inference via *warmstarting* [19]. The reverse process is initialized from a prior trajectory estimate (e.g., the solution from the previous control step) rather than from pure noise, so that only the final fraction of denoising steps must be executed. This substantially reduces latency while preserving most of the predictive performance, making diffusion architectures compatible with real-time receding-horizon control.

Figure 4 reports the resulting inference times. We impose a conservative inference latency threshold of approximately 40 ms, which corresponds to about 5% of the original diffusion steps (5 warmstarted iterations in our implementation). The RMSE degradation induced by this truncation is shown in Fig. 5. For this analysis, we focus exclusively on models trained on dataset $\mathscr{D}_2$. This choice is empirically justified by the results in Fig. 3, which demonstrate that expanding the training distribution to $\mathscr{D}_3$ yields only marginal accuracy improvements, indicating that the generalization performance has largely plateaued. The CNN-based diffusion architectures are the most affected, particularly the inpainted variant, suggesting a stronger dependence on the full denoising chain. In contrast, the cTRF remains largely insensitive to warmstarting, retaining superior OOD performance while only slightly underperforming the RoboMorph in the ID region.

From a control perspective, transformer-based models are naturally well suited to high-frequency operation once their inference pipeline is optimized, and we regard low-level control implementations with strict latency constraints as a straightforward extension. In such settings, diffusion models should be systematically warmstarted to comply with tight real-time budgets. On the other hand, diffusion models, by construction multi-modal over the trajectory space, are particularly attractive for planning and policy generation, where OOD generalization is critical and inference times are less restrictive.

IV. Conclusion

In this work, we studied black-box meta-modeling for robotic system identification through a systematic comparison of deterministic and generative sequence models. By casting dynamics learning as an in-context meta-learning problem, we evaluated how architectural choices impact accuracy, robustness, and control-oriented deployment.

Our results highlight three main findings. First, deterministic Transformer-based models such as RoboMorph perform well in simple in-distribution settings but degrade under distributional shifts, especially for complex multi-frequency

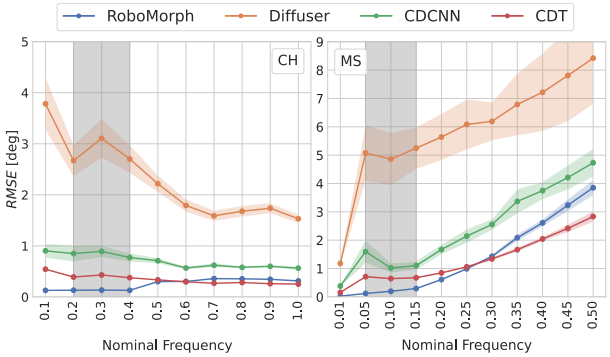

Fig. 5. For brevity, we focus on architectures trained on $\mathscr{D}_2$ for chirp and sinusoidal signals with 5 diffusion steps corresponding to about 40*ms* inference time: transformer-based models sustain warmstarted trajectories relatively well whereas convolution-based models suffer drastically. Among transformer-based models, diffusion processes are provide more versatile representations in modeling high frequency responses whereas for low frequency responses this versatility is practically insignificant.

dynamics. Second, diffusion-based models significantly improve robustness by modeling trajectory distributions; among them, the joint formulation (Diffuser) achieves the highest accuracy due to its richer action–state representation. Third, conditioned diffusion provides the best trade-off between performance and efficiency, retaining strong robustness while enabling warm-started inference compatible with real-time MPC.

Overall, the choice between joint and conditioned diffusion governs the balance between expressiveness and deployability. While joint diffusion is the most expressive, conditioned diffusion emerges as the most practical solution for control-oriented applications.

Future work will focus on real-world validation and integration within MPC pipelines, enabling data-driven receding-horizon control on physical systems. Additionally, exploring mechanistic interpretability to extract physically meaningful parameters from learned models offers a promising direction to bridge deep meta-learning with classical system identification.

REFERENCES

[1] P. K. Khosla and T. Kanade, "Parameter identification of robot dynamics," in *1985 24th IEEE Conference on Decision and Control*, 1985, pp. 1754–1760.

[2] T. M. Moerland, J. Broekens, A. Plaat, and C. M. Jonker, "Model-based reinforcement learning: A survey," *Foundations and Trends in Machine Learning*, vol. 16, no. 1, pp. 1–118, 2023.

[3] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger, "Learning-based model predictive control: Toward safe learning in control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, no. 1, pp. 269–296, 2020.

[4] A. Ramesh and B. Ravindran, "Physics-informed model-based reinforcement learning," in *Learning for Dynamics and Control Conference*. PMLR, 2023, pp. 26–37.

[5] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, vol. 44, no. 10-11, pp. 1684–1704, 2025.

[6] B. Ai, S. Tian, H. Shi, Y. Wang, T. Pfaff, C. Tan, H. I. Christensen, H. Su, J. Wu, and Y. Li, "A review of learning-based dynamics models for robotic manipulation," *Science Robotics*, vol. 10, no. 106, p. eadt1497, 2025.

[7] N. Hansen, H. Su, and X. Wang, "Td-mpc2: Scalable, robust world models for continuous control," 2024.

[8] G. Giacomuzzo, R. Carli, D. Romeres, and A. Dalla Libera, "A black-box physics-informed estimator based on gaussian process regression for robot inverse dynamics identification," *IEEE Transactions on Robotics*, vol. 40, pp. 4820–4836, 2024.

[9] M. Forgione, F. Pura, and D. Piga, "From system models to class models: An in-context learning paradigm," *IEEE Control Systems Letters*, vol. 7, pp. 3513–3518, 2023.

[10] D. Piga, M. Rufolo, G. Maroni, M. Mejari, and M. Forgione, "Synthetic data generation for system identification: leveraging knowledge transfer from similar systems," in *2024 IEEE 63rd Conference on Decision and Control (CDC)*. IEEE, 2024, pp. 6383–6388.

[11] M. Rufolo, D. Piga, G. Maroni, and M. Forgione, "Enhanced transformer architecture for in-context learning of dynamical systems," in *2025 European Control Conference (ECC)*. IEEE, 2025, pp. 819–824.

[12] M. B. Bazzi, A. A. Shahid, C. Agia, J. Alora, M. Forgione, D. Piga, F. Braghin, M. Pavone, and L. Roveda, "Robomorph: In-context meta-learning for robot dynamics modeling," *arXiv preprint arXiv:2409.11815*, 2024.

[13] M. Elseiagy, T. T. Alemayoh, R. Bezerra, S. Kojima, and K. Ohno, "Data-driven dynamic parameter learning of manipulator robots," in *2026 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, 2026, pp. 193–198.

[14] J. Schmidhuber, "Evolutionary principles in self-referential learning. on learning now to learn: The meta-meta-meta-hook (diploma thesis," 1987.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[16] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[17] Z. Wang, Y. Jiang, Y. Lu, P. He, W. Chen, Z. Wang, M. Zhou, *et al.*, "In-context learning unlocked for diffusion models," *Advances in Neural Information Processing Systems*, vol. 36, pp. 8542–8562, 2023.

[18] W. Meng, H. Ju, T. Ai, R. Gomez, E. Nichols, and G. Li, "Transferring meta-policy from simulation to reality via progressive neural networks," *IEEE Robotics and Automation Letters*, 2024.

[19] M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine, "Planning with diffusion for flexible behavior synthesis," *arXiv preprint arXiv:2205.09991*, 2022.

[20] J. Von Oswald, E. Niklasson, E. Randazzo, J. Sacramento, A. Mordvintsev, A. Zhmoginov, and M. Vladymyrov, "Transformers learn in-context by gradient descent," in *International Conference on Machine Learning*. PMLR, 2023, pp. 35 151–35 174.

[21] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[22] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[23] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[24] C. Gaz, M. Cognetti, A. Oliva, P. Robuffo Giordano, and A. De Luca, "Dynamic identification of the franka emika panda robot with retrieval of feasible parameters using penalty-based optimization," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4147–4154, 2019.

[25] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, *et al.*, "Isaac gym: High performance gpu-based physics simulation for robot learning," *arXiv preprint arXiv:2108.10470*, 2021.