# Sim-to-Real Transfer with Asymmetric Temporal Supervision via Latent Alignment of an INR-FiLM Architecture

Julian Agudelo[1, 2], Vincent Guigue[1], Cristina Manfredotti[1], Evelyne Lutton[1], and Hadrien Piot[2]

[1]UMR MIA Paris Saclay, AgroParisTech and INRAE
[2]Agrial, Agricultural Service

**Abstract**

Learning accurate models of dynamical systems often requires large amounts of labeled data, which are costly or impractical to obtain in real-world scientific and engineering settings. While physics-based simulators provide an abundant supervision option, transferring knowledge from simulation to reality remains challenging, particularly when real-world observations suffer from incomplete labeling. We study transfer learning for time series under asymmetric supervision, where simulated trajectories are fully labeled, but only a few time steps are observed in the target domain.

We propose a FiLM-modulated Implicit Neural Representation (INR-FiLM) architecture that combines a Fourier-encoded MLP with a GRU-based modulation network to incorporate contextual variables through feature-wise linear modulation. This design enables alignment between simulated and real dynamics by promoting domain invariant representations. We provide initial empirical evidence that the proposed approach reduces the prediction error for the entire trajectory despite having extremely limited real-world supervision.

**Keywords**: Implicit Neural Representations, Feature-wise transformations, Domain Alignement, Transfer Learning, Time Series.

## 1 Introduction

In many real-world applications, acquiring sufficient labeled data for training machine learning models remains a significant challenge, as annotations are often expensive or impractical to obtain. This limitation is particularly pronounced in scientific and engineering tasks, where reliable mechanistic and physics-based models exist, but are difficult to exploit in practice due to measurement constraints, complex parameterization, or incomplete knowledge of system variables. Transfer learning offers a natural framework for such setup, as it aims at reusing knowledge learned from a richly supervised source task, here simulated data, to improve the performance of a related target task with scarce data.

A particularly challenging, yet relatively underexplored setting, is transfer on time series with asymmetric supervision. In this setting, only a few time steps in the target sequences are observed, while the source sequences are fully labeled.

This paper addresses transfer learning for dynamical systems in which supervision is asymmetric across domains. Although we present a use case about voltage curve prediction during battery discharge cycles, similar asymmetric supervision scenarios arise in many real-world applications. Some examples include crop growth monitoring from sparse field measurements, weather forecasting with irregular sensor observations, environmental monitoring with intermittent sampling, and medical time-series analysis where clinical measurements are collected only at specific visits.

In particular, we tackle simulation-to-real transfer, where simulated data provide dense trajectories, while real-world observations are limited to sparse boundary measurements. We formulate this problem as a transfer learning task with asymmetric temporal supervision, in which learning from endpoint observations alone leads to an ill-posed inverse problem, due to the non-identifiability of intermediate dynamics. We provide the following contributions:

**Contributions**

- We propose an INR-FiLM (Implicit Neural Representation with Feature-Wise Linear Modulation) architecture that uses simulated data and physical priors as structural supervision to support transfer of domain-invariant knowledge (Section 4).
- We provide empirical evidence showing that, using appropriate domain alignment and training protocols on the proposed architecture, we reduce full-trajectory error despite extremely sparse real-world supervision (Section 6).
- We present a dataset for transfer on time series with asymmetric supervision, combining real battery measurements from a public dataset with synthetic simulations (Section 5).

This paper focuses primarily on the conceptual and methodological contribution of the proposed approach. As such, it should be viewed as an early-stage investigation that introduces the framework and presents initial empirical evidence consistent with the main hypothesis, rather than a comprehensive empirical study.

## 2 Related Work

**Transfer Learning for Time Series** Although transfer learning has been extensively studied in computer vision and natural language processing (NLP), its extension to time series remains comparatively recent (Weber et al., 2021). Recent advances include optimal transport–based temporal alignment (Painblanc et al., 2023), multi-source domain adaptation for sensor time series (Wilson et al., 2020), and benchmarks for unsupervised domain adaptation in time series classification (Fawaz et al., 2025). More recently, foundation models pretrained on large-scale datasets (Kottapalli et al., 2025) have reshaped transfer learning by providing transferable representations for classification (Feofanov et al., 2025, 2026) and forecasting (Das et al., 2024; Ma et al., 2024; Xiao et al., 2025; Shi et al., 2025), often requiring only lightweight fine-tuning. Nevertheless, task-specific transfer remains crucial in severely data-limited settings, where principled prior knowledge is necessary for successful adaptation.

**Sim-to-Real Transfer** Simulation-to-real (sim-to-real) is a transfer learning paradigm that leverages mechanistic simulators to train models for real-world tasks. Widely studied in robotics (Zhao et al., 2020; Wang et al., 2024; Yin et al., 2025), it also finds applications in scientific machine learning, such as High-Energy Physics (Baalouch et al., 2019) and traffic digital twins (Hu et al., 2024). Unlike conventional transfer learning, sim-to-real explicitly addresses discrepancies between simulated and real dynamics. Typically, models are pretrained on large simulated datasets and fine-tuned using limited real-world observations, often improving performance under sparse data when combined with physics-informed priors or uncertainty-aware adaptation methods.

**Implicit Neural Representations** Implicit Neural Representations (INRs) are neural network based functions that map coordinates to signal values. They have been successfully applied to images (Strümpler et al., 2022; Luigi et al., 2023), 3D shapes (Park et al., 2019), and neural radiance

fields (Mildenhall et al., 2020). To mitigate the spectral bias of MLPs toward low frequencies (Ramasinghe et al., 2022), sinusoidal activations (Sitzmann et al., 2020) or Fourier feature encodings (Zhong et al., 2020) are commonly used. INRs enable differentiable representations adaptable via latent codes or hypernetworks, supporting applications such as interpolation, style transfer, compression, and time-continuous modeling for forecasting, imputation and anomaly detection (Feng et al., 2022; Kim et al., 2022; Dupont et al., 2021; Naour et al., 2024; Li et al., 2025).

**Feature-Wise Modulations** Feature-wise modulations condition neural networks through learned transformations applied to intermediate features. Typically implemented via affine transformations, such as Feature-wise Linear Modulation (FiLM), these methods enable models to adapt their representations based on external signals, like latent variables or contextual information. By modulating features rather than network parameters, feature-wise approaches provide an efficient and flexible conditioning strategy (Perez et al., 2017). They are widely used in generative modeling (Oord et al., 2016), style transfer (Dumoulin et al., 2017), and domain adaptation (Oreshkin et al., 2019).

**Domain Alignment** Domain alignment aims to reduce distributional discrepancies between the source and the target domains to facilitate knowledge transfer (Ganin et al., 2017). Common approaches seek to learn domain-invariant representations by minimizing statistical differences between feature distributions (Sun and Saenko, 2016; Wang et al., 2020) or by enforcing consistency through adversarial or contrastive objectives (Tzeng et al., 2017; Jin et al., 2022; Chen et al., 2024).

# 3 Problem Formulation

## 3.1 Transfer learning

In the classical transfer learning framework (Pan and Yang, 2010), a domain is defined by a feature space $\mathcal{X}$ and a marginal distribution $P(X)$, where $X$ is a random variable such that $X \in \mathcal{X}$. Given an arbitrary domain $\mathcal{A} = \{\mathcal{X}, P(X)\}$, a task is defined by a label space $\mathcal{Y}$ and a conditional distribution $P(Y \mid X)$, induced by an unknown predictive function $f$. Here, $Y$ denotes a random variable such that $Y \in \mathcal{Y}$. Accordingly, a task can be denoted as $\mathcal{T} = \{\mathcal{Y}, f\}$.

Transfer learning considers two elements: (1) a source domain $\mathcal{S}$, corresponding here to the simulated domain, and (2) a target domain $\mathcal{R}$, corresponding here to the real domain, each with their associated tasks. The domain $\mathcal{S}$ is associated with a joint distribution $P_{\mathcal{S}}(X, Y)$ over inputs $X$ and their labels $Y$. Some realisations of $X$ and $Y$ form the dataset $\mathcal{D}_{\mathcal{S}} = \{(x^{(i)}, y^{(i)})\}_{i=1}^{N_{\mathcal{S}}}$. In the same way, to $\mathcal{R}$ is associated a distribution $P_{\mathcal{R}}(X, Y)$ and a dataset $\mathcal{D}_{\mathcal{R}}$, typically smaller or less richly annotated than $\mathcal{D}_{\mathcal{S}}$.

The goal is to learn an hypothesis $f_\theta^* : \mathcal{X} \to \mathcal{Y}$ that minimises the target risk:

$$R_{\mathcal{R}}(\theta) = \mathbb{E}_{(X,Y)\sim P_{\mathcal{R}}}\left[\ell\big(f_\theta^*(X),\, Y\big)\right], \tag{1}$$

For this purpose, we aim to use knowledge obtained from $\mathcal{D}_{\mathcal{S}}$ to compensate for the scarcity of information about $\mathcal{R}$. The domain gap, whether covariate shift ($P_{\mathcal{S}}(X) \neq P_{\mathcal{R}}(X)$), concept shift ($P_{\mathcal{S}}(Y \mid X) \neq P_{\mathcal{R}}(Y \mid X)$), or both, makes the problem non-trivial.

## 3.2 Transfer Learning for Time Series with Asymmetric Supervision

In the context of this paper, the random variable $X \in \mathcal{X}$ is a vector we denote $\mathbf{x}$ that is composed by dynamic $\mathbf{x}^{(d)}$ and static $\mathbf{x}^{(s)}$ features, while the label $Y \in \mathcal{Y}$ is a time series $\{y_\tau\}_{\tau=0}^{T}$. Consequently, the hypothesis becomes a dynamic model $f_\theta^*$ that predicts trajectory $\{\hat{y}_\tau\}_{\tau=0}^{T}$ from $\mathbf{x}$ in an autoregressive manner. It is important to note that, in the specific use case considered in this work (see Section 5), the value $y_o$ is passed within $\mathbf{x}^{(s)}$, since it is a fundamental part of the context of each instance.

However, this modeling choice is part of the adaptation to the specific problem and does not alter the general problem formulation or the learning objective, which remains the prediction of the target trajectory conditioned on the input features.

The characteristic of our problem is that the two domains differ in what is observed about each label trajectory. In the simulated domain $\mathcal{S}$, a high-fidelity physical or mechanistic simulator provides complete trajectories. Each sample $(i)$ is a tuple $(\mathbf{x}^{(i)}, \{y_\tau\}_{\tau=0}^{T}{}^{(i)})$ and the source loss penalises the model at each time step:

$$\mathcal{L}_{\mathcal{S}}(\theta) = \frac{1}{N_{\mathcal{S}}} \sum_{i=1}^{N_{\mathcal{S}}} \sum_{\tau=0}^{T} \ell\big(\hat{y}_\tau^{(i)},\, y_\tau^{(i)}\big) \tag{2}$$

In the real-world domain $\mathcal{R}$, only the initial and final states are measured. Each sample $(j)$ is a tuple $(\mathbf{x}^{(j)}, \{y_0, y_T\}^{(j)})$, hence the loss function sees only the endpoints of the label time series:

$$\mathcal{L}_{\mathcal{R}}(\theta) = \frac{1}{N_{\mathcal{R}}} \sum_{j=1}^{N_{\mathcal{R}}} \big[\ell\big(\hat{y}_0^{(j)},\, y_0^{(j)}\big) + \ell\big(\hat{y}_T^{(j)},\, y_T^{(j)}\big)\big] \tag{3}$$

Beyond the usual distributional shift ($P_{\mathcal{S}} \neq P_{\mathcal{R}}$), our problem presents a supervision asymmetry, in other words, $\mathcal{D}_{\mathcal{S}}$ provides $T$ labels per sample while $\mathcal{D}_{\mathcal{R}}$ provides only 2. The intermediate trajectory $\{y_\tau\}_{t=1}^{T-1}{}^{(j)}$ is latent in $\mathcal{R}$. The consequence is that minimizing $\mathcal{L}_{\mathcal{R}}$ alone defines an underdetermined inverse problem, as infinitely many trajectories with distinct intermediate $\hat{y}_\tau$ can interpolate between the same boundary conditions $(y_0, y_T)$. In general, this makes the problem ill-posed.

We are interested in systems where high-fidelity simulators provide virtually unlimited dense trajectories and governing laws can be used to impose structural constraints. Our central hypothesis is that in such systems, transfer the knowledge from simulated supervision can resolve the ambiguity that boundary observations alone cannot.

When modeling complex systems with simulators, it is common for the simulator to rely on a richer set of input variables than those accessible in real-world scenarios. While this paper does not focus on a specific methodology for addressing this challenge, the use case considered here (see Section 5) required implementing additional measures to adapt the model to this input mismatch.

# 4 Methodology

Given the problem presented in Section 3, our objective is to find parameters $\theta^*$ that minimise the true full-trajectory risk in the target domain $\mathcal{R}$, even though that quantity is never observable:

$$\theta^* = \arg\min_{\theta}\ \mathbb{E}_{P_{\mathcal{R}}}\left[\sum_{\tau=0}^{T} \ell(\hat{y}_\tau,\, y_\tau)\right] \tag{4}$$

Due to lack of supervision, problem (4) is intractable, but the simulated domain $\mathcal{S}$ offers additional inputs and dense trajectories that are unavailable in the real-world domain $\mathcal{R}$. Our approach can be interpreted through the Learning Using Privileged Information (LUPI) framework (Vapnik and Vashist, 2009), instantiated here via a teacher–student paradigm (Cornuéjols, 2024). In this paradigm, the teacher model provides privileged information to the student, which must, ultimately, solve the target task with the restricted supervision of the target domain.

Our setup assumes that privileged information emerges through the non-linear transformations learned by the teacher, which maps observations into continuous latent representations. These representations encode structured knowledge induced by dense supervision and architectural inductive biases.

We propose a novel modeling framework for transfer learning under the aforementioned conditions. First, we introduce an architecture in which the invariant dynamics of the target system are

parameterized as a continuous function via an INR, while context variables and external stimuli act as modulatory inputs. Next, we construct a mirror architecture that allows different modulation variables and different supervisions. Finally, we propose a set of transfer learning strategies based on the alignment of latent spaces associated with the continuous components of the mirrored architecture.

### 4.1 The Proposed Architecture

We propose a FiLM-modulated Implicit Neural Representation (INR-FiLM) that naturally supports an alignment strategy by introducing a mirrored architecture (see Figure 1). The main network is an INR (a Fourier-encoded MLP), while the FiLM modulation network is a Gated Recurrent Unit (GRU) that processes the context variables and introduces a linear modulation after each layer of the main network. By using an INR, the architecture can naturally handle both dense and sparse temporal signals, which is important in real-world contexts where measurements may be irregular or partially available.

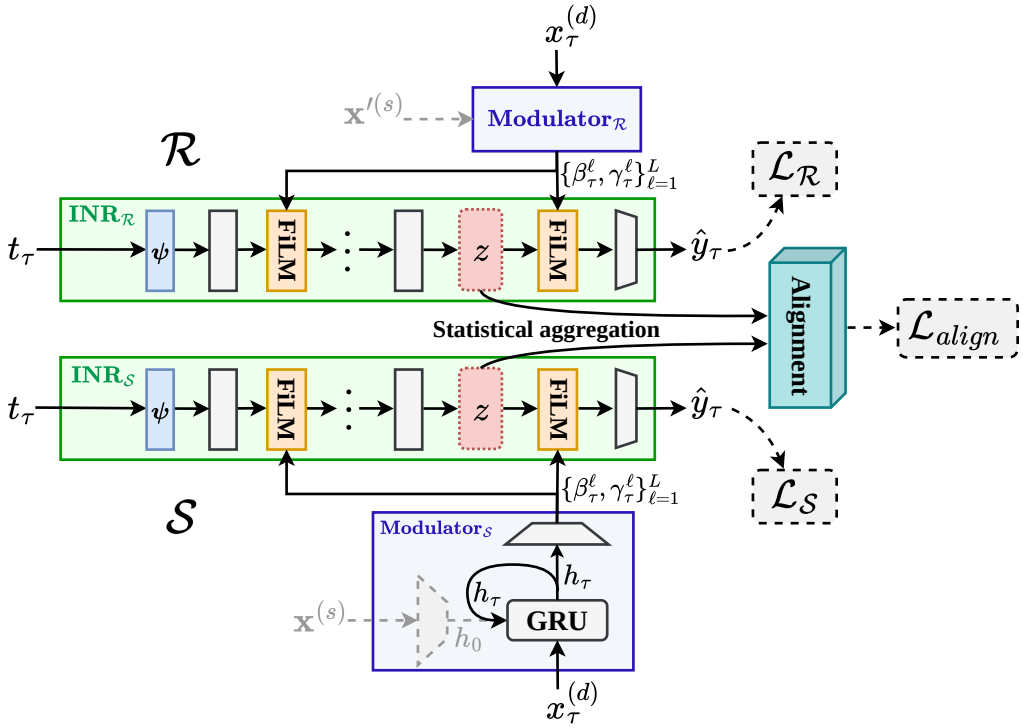


Figure 1: Overview of the proposed architecture. From this point on, we will refer to each of the two halves of this mirrored architecture as the simulated branch (bottom) and the real-world branch (top). $\mathbf{x}^{(s)}$ and $\mathbf{x}'^{(s)}$ are colored in lightgray, representing that they are only used at $t_0$ for initializing the GRU hidden state.

The model receives a time series of dynamic inputs $\mathbf{x}^{(d)} = \{\mathbf{x}_\tau\}_{\tau=1}^T$ together with static covariates $\mathbf{x}^{(s)}$, providing global information about the time series. The static covariates initialise the hidden state of the GRU encoder through a multilayer perceptron (MLP):

$$\mathbf{h}_0 = \text{MLP}\left(\mathbf{x}^{(s)}\right), \tag{5}$$

then, a Gated Recurrent Unit (GRU) processes the dynamic inputs sequentially, producing the hidden states $\mathbf{h}_\tau$ that encode the accumulated context up to step $\tau$:

$$\mathbf{h}_\tau = \text{GRU}\left(\mathbf{x}_\tau^{(d)}, \mathbf{h}_{\tau-1}\right), \qquad \tau = 1, \ldots, T. \tag{6}$$

**FiLM modulation** For each layer $\ell = 1, \ldots, L$ of the INR and for each step $\tau$, the modulation parameters are obtained from the current GRU hidden state $\mathbf{h}_\tau$ through linear projections:

$$\begin{aligned} \boldsymbol{\gamma}_\tau^\ell &= \mathbf{W}^{\ell,\gamma}\mathbf{h}_\tau + \mathbf{b}^{\ell,\gamma}, \\ \boldsymbol{\beta}_\tau^\ell &= \mathbf{W}^{\ell,\beta}\mathbf{h}_\tau + \mathbf{b}^{\ell,\beta}, \end{aligned} \qquad \boldsymbol{\gamma}_\tau^\ell, \boldsymbol{\beta}_\tau^\ell \in \mathbb{R}^d, \tag{7}$$

where $d$ is the latent dimension of the INR.

**Modulated INR forward pass** The INR is evaluated independently at each time step $t_\tau$ using modulation parameters generated from $\mathbf{h}_\tau$. Since MLP based implicit neural representations suffer from spectral bias and tend to favor low-frequency solutions when operating directly on low-dimensional coordinates, the scalar timestep is first mapped to a higher-dimensional representation using a Fourier feature encoding $\boldsymbol{\psi} : [0, 1] \to \mathbb{R}^{2K}$ as proposed by Zhong et al. (2020) or Mildenhall et al. (2020)

$$\boldsymbol{\psi}(t) = \big[\sin(2^0\pi t), \cos(2^0\pi t), \ldots, \sin(2^{K-1}\pi t), \cos(2^{K-1}\pi t)\big]. \tag{8}$$

This encoding injects multi-scale periodic features that enable the INR to represent high-frequency temporal variations. As a result, the first layer of the INR is $\mathbf{z}_\tau^{\ell=0} = \boldsymbol{\psi}(t_\tau)$, then, each of the remaining layers computes

$$\mathbf{z}_\tau^\ell = \boldsymbol{\gamma}_\tau^\ell \odot \phi\Big(\mathbf{W}^\ell \mathbf{z}_\tau^{\ell-1} + \mathbf{b}^\ell\Big) + \boldsymbol{\beta}_\tau^\ell, \qquad \ell = 1, \ldots, L, \tag{9}$$

where $\phi$ denotes the activation function and $\odot$ the element-wise multiplication.

The proposed architecture is designed in order to make the INR a continuous representation of the invariant dynamics of the system, while the GRU captures context-dependent variables that influence its behavior. This separation is both principled and advantageous for transfer, as the physical laws governing a system are universal, whereas the operating context is situational.

Finally, the network output, at time step $\tau$, is

$$\hat{y}_\tau = \hat{f}(t_\tau, \mathbf{h}_\tau) = \mathbf{w}^\top \mathbf{z}_\tau^L + b, \tag{10}$$

### 4.2 Training strategies

Using the proposed architecture, transfer is achieved by aligning statistics of the latent representations extracted from the INR's penultimate layer. Due to the differing lengths of the input time series, we first apply mean-pooling to obtain fixed size vectors before computing the statistical descriptors required for each alignment method. We explored several well-established alignment techniques, including CORAL (Sun and Saenko, 2016), Maximum Mean Discrepancy (MMD) alignment (Wang et al., 2020) and adversarial training with gradient reversal (Ganin et al., 2017). For adversarial alignment, we consider two variants: one based on the mean and the standard deviation and another based on the covariance, that differ only on the aggregation of the latent representations. Detailed descriptions of the alignment strategies are provided in Appendix A.

For brevity, we refer to the loss induced by any alignment strategy generically as $\mathcal{L}_{align}$, regardless of the specific method used. Note that when the alignment method uses an adversarial discriminator, an additional neural module (the classifier) is required, and the gradient of the $\mathcal{L}_{align}$ signal is reversed, following stablished practices (Ganin et al., 2017).

We propose two setups for perform training: teacher-student and end-to-end. In the teacher-student scheme, the synthetic branch is first pretrained on simulated data with full supervision over the label time series, after which its weights are frozen. The real branch is then trained using the fixed synthetic latent representations as a stable reference that the alignment loss encourages the real latent representations to match. In the end-to-end scheme, both branches are optimized jointly from the start of the adaptation phase, allowing the synthetic branch to adjust its representations in response to the real data and potentially tightening the domain alignment at the cost of revisiting the

pretraining objective. In the teacher-student setup, the teacher is optimized with $\mathcal{L}_{\mathcal{S}}$ (2), while the student minimizes the composite loss:

$$\mathcal{L} = \alpha\mathcal{L}_{\mathcal{R}} + \beta\mathcal{L}_{align}, \tag{11}$$

where $\mathcal{L}_{\mathcal{R}}$ is defined in (3). In contrast, the end-to-end approach jointly optimizes

$$\mathcal{L} = \alpha\mathcal{L}_{\mathcal{S}} + \beta\mathcal{L}_{\mathcal{R}} + \gamma\mathcal{L}_{align}. \tag{12}$$

Since we are dealing with a physical phenomenon we consider incorporating additional physical constraints to the objective functions.

# 5 Use Case

To assess the performance of the proposed architecture, we consider the task of predicting voltage curves $\{y_\tau\}_{\tau=0}^{T}$ from current measurement time series $\mathbf{x}^{(d)}$ and static contextual variables $\mathbf{x}^{(s)}$ for lithium-ion batteries under accelerated discharge conditions. Battery discharge dynamics provide a simple yet sufficiently challenging testbed.

## 5.1 Dataset

For this use case, we prepared a dataset consisting of experimental data from the `regular_alt_batteries` subset of Samsung INR18650-25R batteries (Fricke et al., 2023) and a set of simulated trajectories generated using the electrochemical model proposed by Daigle and Kulkarni (2013). The complete dataset is publicly available on Hugging Face [1].

Figure 2 shows a sample from each domain and visually highlights where supervision is available in each one.

**Simulated domain** To generate the synthetic data, we use the electrochemical model proposed by Daigle and Kulkarni (2013), whose default parameters correspond to the batteries in the real dataset, and which is publicly available through the ProgPy package (Teubert et al., 2025). For sampling from this model, the electrochemical parameters $R_o$, $q_{\max}$, $q_{\text{Mobile}}$ fraction, and surface fraction are drawn via Latin Hypercube Sampling (LHS) to efficiently capture a wide range of aging states. Each trajectory is assigned either a flat constant-current profile or a variable piecewise-constant profile with random segment durations. Trajectories are simulated at the single-cell level using the electrochemical model and, when needed, scaled to series packs by adjusting voltages and resistances accordingly.

In the simulated domain, $\mathbf{x}^{(s)}$ contains $R_o$ and $q_{max}$, which are well-known descriptors of battery aging (Daigle and Kulkarni, 2013) as well as the fully charged battery voltage.

**Real domain** We use the `regular_alt_batteries` subset of the accelerated life testing dataset for Samsung INR18650-25R lithium-ion batteries with constant and variable current conditions (Fricke et al., 2023)[2] to represent the real domain. The dataset consists of continuous measurements collected throughout the entire cycling process of the battery blocks, including discharge, rest, and charge phases. As a preprocessing step, we isolate the discharge cycles and focus specifically on the voltage load and current load (demand) variables. In addition, we compute the index of each discharge cycle, which serves as a proxy for battery aging. To introduce asymmetric supervision, we assume that only the initial and final voltages are observed.

[1] Anonymized, as the article is undergoing a double-blind review

[2] https://data.nasa.gov/dataset/randomized-and-recommissioned-battery-dataset

In the real domain, the static contextual variables include the cycle number and the fully charged battery voltage, where the cycle number is a proxy for the battery aging. As these variables differ from those in the simulated domain, we denote them by $\mathbf{x}'^{(s)}$.

Although access to current measurements usually implies access to voltage measurements, we intentionally restrict supervision to create a controlled setting for studying the proposed architecture under different training protocols.

Note that, given the use case, the number of variables in $\mathbf{x}'^{(s)}$ is smaller than those in $\mathbf{x}^{(s)}$. Consequently, we add a 2 layers MLP, called `CyclesEstimator` to the real-branch modulator to transform the cycle number into a representation that can model battery aging in place of the $R_o$ and $q_{\max}$ variables available in the simulated domain.

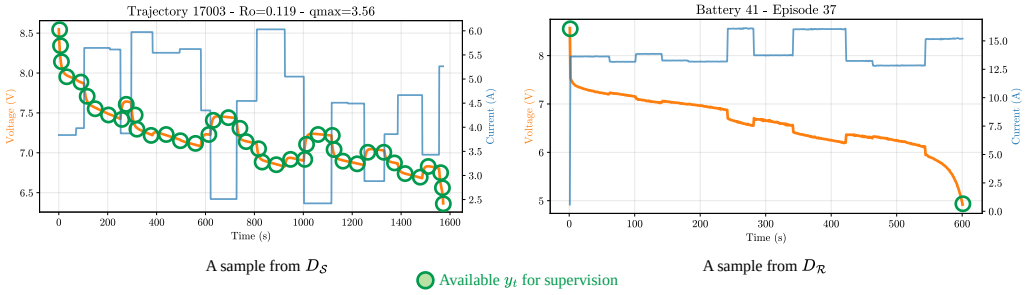


Figure 2: One example from the simulated domain and one from the real-world domain. Green circles indicate where supervision is available.

## 5.2 Experimental Setup

On the main experiments, we evaluate the four alignment methods (MMD, CORAL, adversarial $(\mu, \sigma)$, and adversarial (cov)), instantiated under the two aforementioned training setups: teacher-student (synthetic branch frozen after pretraining) and end-to-end (joint optimization). This results in eight alignment configurations, all sharing the same mirrored architecture (see Figure 1). The code needed to reproduce the experiments is publicly available on GitHub [3].

**Baselines** Four baselines are evaluated to contextualize the proposed approach. $\text{INR}_{\text{FT}}$ and $\text{INR}_{\text{mod}}$ reuse the FiLM-modulated INR backbone and follow a two-phase protocol: pretraining on synthetic data and fine-tuning on real data with boundary supervision. $\text{INR}_{\text{FT}}$ updates all weights, while $\text{INR}_{\text{mod}}$ freezes the INR and trains only the GRU modulation branch and the `CyclesEstimator`.

$\text{LSTM}_{\text{FT}}$ and $\text{LSTM}_{\text{HT}}$ use a standard LSTM taking the current series concatenated with the cycle count representation. They follow the same pretraining and fine-tuning protocol, $\text{INR}_{\text{FT}}$ updates all weights, while $\text{LSTM}_{\text{HT}}$ freezes the encoder and updates only the output head. None of the baselines includes explicit domain alignment, relying solely on boundary supervision in the real domain.

**Evaluation Metrics** To evaluate the performance of the proposed models, we use the Mean Squared Error (MSE) at time steps where supervision is available. To additionally assess the physical consistency of the generated sequences, as a proxy for evaluating sequences quality without supervision, we introduce a metric $\mathbf{V}_{mon}$ measuring the fraction of invalid consecutive time-step pairs $(t, t+1)$ for which the discharge monotonicity constraint is violated, i.e., when $\hat{y}_{t+1} - \hat{y}_t > 0$ while the current

[3] Anonymized, as the article is undergoing a double-blind review

$\mathbf{x}^{(d)} > 0$.

$$\mathbf{V}_{\text{mon}} = \frac{1}{K} \sum_{k=1}^{K} \frac{\sum_{t=1}^{T-1} \mathbf{1}\Big[\hat{y}_{t+1}^{(k)} - \hat{y}_{t}^{(k)} > 0 \ \wedge \ x_t^{(d)(k)} > 0\Big]}{\sum_{t=0}^{T-1} \mathbf{1}\big[x_t^{(d)(k)} > 0\big]} \tag{13}$$

In Equation (13) $k$ takes $i$ or $j$ values depending on whether the evaluation is based on simulated or real-world data.

# 6 Results

## 6.1 Experimental Results

It is important to recall that in a deployment scenario, the full-trajectory MSE is not an accessible metric: only the boundary voltages $y_{\{0,T\}}$ are observed, and no ground truth intermediate measurements are available to evaluate the quality of the reconstructed discharge curve. In this work, we use the full-trajectory MSE solely as a post-hoc diagnostic, made possible by the controlled experimental setting of the accelerated life testing for lithium-ion batteries dataset, and not as a training or model selection criterion. We acknowledge that it represents the best proxy we currently have to quantify how well the predicted trajectory matches reality, and that identifying a more practically grounded evaluation metric remains an open problem.

| **Alignment method** | **Simulated Domain $\mathcal{S}$** | | | | **Real-world Domain $\mathcal{R}$** | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | **T-S** | | **E2E** | | **T-S** | | | **E2E** | | |
| | **MSE** | $\mathbf{V}_{mon}$ | **MSE** | $\mathbf{V}_{mon}$ | $\mathbf{MSE}_{\{0,T\}}$ | $\mathbf{V}_{mon}$ | **MSE** | $\mathbf{MSE}_{\{0,T\}}$ | $\mathbf{V}_{mon}$ | **MSE** |
| MMD | 0.075 | 0.397 | 0.084 | 0.369 | 9.41e-4 | 0.380 | 0.337 | 9.29e-4 | 0.363 | 0.301 |
| CORAL | 0.074 | 0.377 | 0.081 | 0.397 | 7.18e-4 | 0.062 | 0.494 | 7.84e-4 | 0.199 | 0.147 |
| Adv $(\mu, \sigma)$ | 0.071 | 0.333 | 0.091 | 0.425 | 5.65e-4 | 0.451 | 0.391 | 1.98e-3 | 0.475 | 0.870 |
| Adv (cov) | 0.075 | 0.384 | 0.086 | 0.394 | 8.26e-4 | 0.028 | 0.897 | 1.43e-3 | 0.407 | 0.446 |

Table 1: Results over the test set by domain. In the real-world domain, the MSE computed over the entire trajectory is not available. We report it here solely as an objective evaluation metric and do not use it during training or hyperparameter optimization. T-S means teacher-student and E2E means end-to-end.

Table 1 show that the simulated domain results are consistent across alignment methods, with MSE ranging from 0.071 (Adv $(\mu, \sigma)$ T-S) to 0.091 (Adv $(\mu, \sigma)$ E2E). Teacher-student training strategy shows better performances, as the simulated branch is frozen before alignment begins. In the real-world domain the differences are more pronounced. CORAL E2E achieves the lowest full-trajectory MSE (0.147) alongside a competitive boundary error ($7.84 \times 10^{-4}$), while adversarial methods, despite their stronger alignment signal, do not systematically translate that advantage into better trajectory reconstruction, with Adv $(\mu, \sigma)$ E2E reaching an MSE of 0.870. This suggests that, for the current use case, second-order distribution matching is sufficient and that the additional complexity of adversarial training can destabilize the real-branch predictions. Regarding $\mathbf{V}_{\text{mon}}$, CORAL T-S stands out with a notably low score (0.062), indicating stronger preservation of the monotonic discharge profile expected from the underlying physics.

Table 2 shows that the proposed $\text{INR} - \text{FiLM}_{\{\text{E2E,CORAL}\}}$ model improves over all baselines by a large margin on both metrics. On full-trajectory MSE it reduces the error by a factor of roughly $10\times$ relative to the best baseline ($\text{LSTM}_{\text{HT}}$) and the boundary MSE is $3\times$ smaller than any baseline. Notably, both LSTM variants and the plain INR baselines exhibit high $\text{MSE}_{\{0,T\}}$ values (0.363–0.434), meaning they fail to satisfy even the weak boundary supervision that is the only signal available at training time in the real domain.

| Model | $\text{MSE}_{\{0,T\}}$ | MSE |
|---|---|---|
| $\text{LSTM}_{\text{FT}}$ | 0.371 | 1.851 |
| $\text{LSTM}_{\text{HT}}$ | 0.434 | 1.487 |
| $\text{INR}_{\text{FT}}$ | 0.366 | 2.968 |
| $\text{INR}_{\text{mod}}$ | 0.363 | 2.600 |
| $\text{INR} - \text{FiLM}_{\{\text{E2E,CORAL}\}}$ **(Best Ours)** | **7.84e-4** | **0.147** |

Table 2: Results for the baselines and for our best model in the real-world domain.

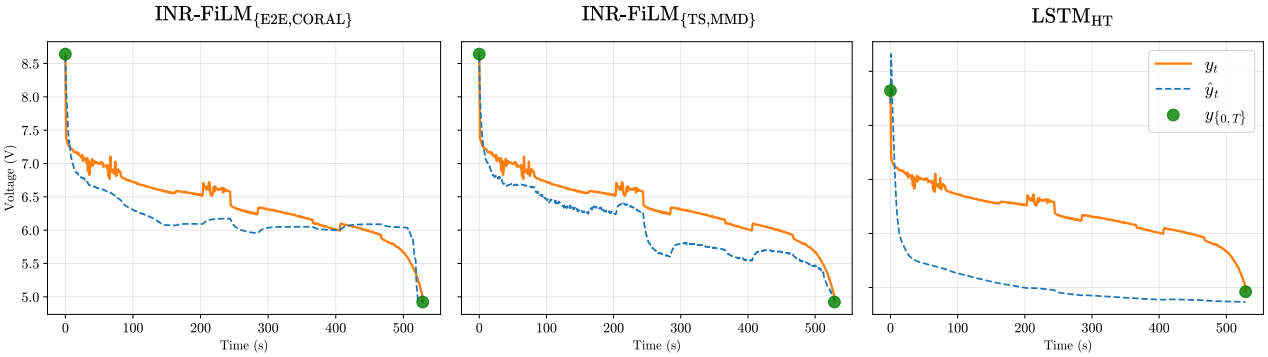


Figure 3: Results for 2 of our proposed models and for the $\text{LSTM}_{\text{HT}}$ baseline on the discharge cycle 158 of the battery 51 (sample from the test partition of $\mathcal{D}_{\mathcal{R}}$). The orange trajectory is unknown, only $y_0$ and $y_T$ are observed during training.

Figure 3 shows representative discharge curves predicted by three models on the same real test sample. All models correctly anchor the trajectory at the supervised boundary voltages $y_{\{0,T\}}$; however, $\text{INR} - \text{FiLM}_{\{\text{E2E,CORAL}\}}$ most faithfully recovers the intermediate discharge profile, while $\text{INR} - \text{FiLM}_{\{\text{TS,MMD}\}}$ underestimates the voltage magnitude but captures the overall curve shape with good fidelity. Finally the $\text{LSTM}_{\text{HT}}$ baseline model collapses too steeply in the early timesteps, failing to capture the plateau region, characteristic of the discharge dynamics. Notice that in a deployment scenario the 3 models are equivalent from the point of view of quantitative metrics and that without the orange curve, it's hard to decide which one is better.

## 6.2 Discussion

**Diversity and Covering of $\mathcal{S}$ are Critical for a Successful Transfer** The effectiveness of the proposed architecture and transfer protocols is highly dependent on the relevance and coverage of the simulated data. Figure 4 shows one of our preliminary experiments conducted to identify potential failure modes of the approach. On this experiment, the simulator was restricted to generating data from a single battery cell (whereas we know that the real system comprises two batteries connected in series). Moreover, the discharge of the current was limited to a narrow range of very low magnitudes. Consequently, adaptation to the real domain was catastrophic: while the model captured the overall dynamics, it suffered from an insurmountable scaling problem. Results were clearly improved when we added a wider variety of scenarios to the simulator, including setups with 2 and 3 batteries in series.

This observation highlights that sufficient diversity and accurate system representation on the simulated domain is critical for successful transfer: an inadequate simulation coverage can lead to severe degradation in real domain performance. This is particularly dangerous when failure modes may remain undetected and lead to silently degraded predictions. Accordingly, the proposed framework necessarily relies on expert knowledge to help reduce the gap between the source and the target

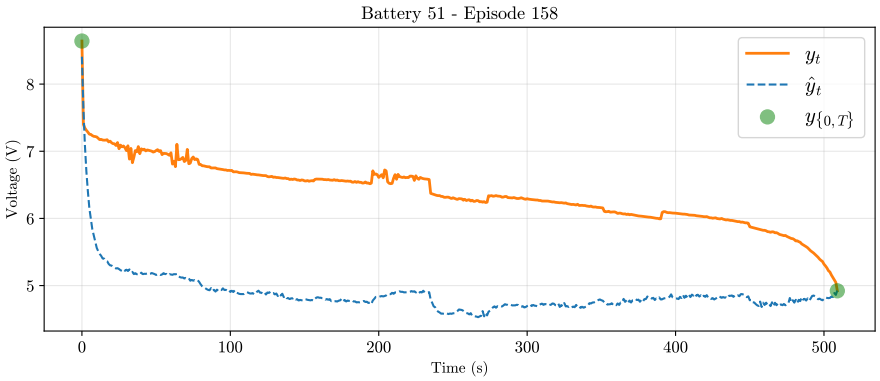


Figure 4: Results for our proposed INR − FiLM$_{\{\text{E2E,CORAL}\}}$ model on the discharge cycle 158 of the battery 51 when $\mathcal{D}_{\mathcal{S}}$ was composed by single battery cell simulations only.

domains.

**Beyond Domain Distance in Transfer Learning** Theoretical analyses of transfer learning emphasize the role of the distance between the source and the target domains to determine transfer effectiveness (Cornuéjols, 2024). Formal transfer learning analyses and heuristic considerations suggest that a small domain gap is generally important for successful adaptation (Blitzer et al., 2007; Redko et al., 2022). However, focusing exclusively on the distance between domains may be misleading, as it may overlook other factors that are critical for transferring useful information across domains or tasks (Cornuéjols, 2024). In our setup, measuring distances between conditional distributions is challenging due to the asymmetry present in the supervision.

Given its empirical effectiveness, we present this approach as a promising direction with future work aimed at refining evaluation metrics, characterizing failure modes, and clarifying the type and extent of expert knowledge required for reliable transfer.

# 7 Conclusions and Future Work

We introduced INR-FiLM, an architecture combining Fourier-encoded implicit neural representations with GRU based FiLM modulation for sim-to-real transfer under asymmetric temporal supervision. The mirrored design enables domain alignment at the level of latent representations, allowing the model to leverage dense simulated trajectories to resolve the ill-posed inverse problem resulting from boundary-only real-world supervision. Empirically, the proposed approach reduces full-trajectory MSE by an order of magnitude relative to fine-tuning baselines, and second-order alignment (CORAL) consistently outperforms adversarial strategies, suggesting that covariance matching is both sufficient and more stable.

Despite these promising results, several limitations motivate future work. First, the approach is sensitive to simulation coverage: mismatches in system scale or operating range can cause silent degradation, which calls for principled sim-to-real gap diagnostics. Second, the current evaluation is restricted to a single use case, broader empirical validation across domains with asymmetric supervision such as crop monitoring or irregular clinical time series is necessary to assess generalizability. Third, a theoretical characterization of transfer under supervision asymmetry remains an open problem: structural mismatches in label availability across domains appears to be underexplored in classical domain adaptation bounds (Blitzer et al., 2007; Redko et al., 2022). Finally, incorporating stronger physical constraints during training, and extending the architecture to handle stochastic or partially observed dynamics, are natural directions toward a more robust and deployable framework.

## References


Baalouch, M., Defurne, M., Poli, J.-P., and Cherrier, N. (2019). Sim-to-Real Domain Adaptation For High Energy Physics. arXiv:1912.08001 [cs].

Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Wortman, J. (2007). Learning Bounds for Domain Adaptation.

Chen, J., Zhang, Z., Li, L., Shahrasbi, B., and Mishra, A. (2024). Contrastive Adversarial Training for Unsupervised Domain Adaptation. arXiv:2407.12782 [cs].

Cornuéjols, A. (2024). Some thoughts about transfer learning. What role for the source domain? *International Journal of Approximate Reasoning*, 166:109107.

Daigle, M. and Kulkarni, C. S. (2013). Electrochemistry-based Battery Modeling for Prognostics.

Das, A., Kong, W., Sen, R., and Zhou, Y. (2024). A decoder-only foundation model for time-series forecasting. arXiv:2310.10688 [cs].

Dumoulin, V., Shlens, J., and Kudlur, M. (2017). A Learned Representation For Artistic Style. arXiv:1610.07629 [cs].

Dupont, E., Goliński, A., Alizadeh, M., Teh, Y. W., and Doucet, A. (2021). COIN: COmpression with Implicit Neural representations. arXiv:2103.03123 [eess].

Fawaz, H. I., Grosso, G. D., Kerdoncuff, T., Boisbunon, A., and Saffar, I. (2025). Deep Unsupervised Domain Adaptation for Time Series Classification: a Benchmark. *Data Mining and Knowledge Discovery*, 39(4):39. arXiv:2312.09857 [cs].

Feng, B. Y., Jabbireddy, S., and Varshney, A. (2022). VIINTER: View Interpolation with Implicit Neural Representations of Images. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, Daegu Republic of Korea. ACM.

Feofanov, V., Wen, S., Alonso, M., Ilbert, R., Guo, H., Tiomoko, M., Pan, L., Zhang, J., and Redko, I. (2025). Mantis: Lightweight Calibrated Foundation Model for User-Friendly Time Series Classification. arXiv:2502.15637 [cs].

Feofanov, V., Wen, S., Zhang, J., Pan, L., and Redko, I. (2026). MantisV2: Closing the Zero-Shot Gap in Time Series Classification with Synthetic Data and Test-Time Strategies. arXiv:2602.17868 [cs].

Fricke, K., Nascimento, R. G., Corbetta, M., Kulkarni, C. S., and Viana, F. A. C. (2023). An Accelerated Life Testing Data set for Lithium-Ion Batteries with Constant and Variable Loading Conditions.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2017). Domain-Adversarial Training of Neural Networks. In Csurka, G., editor, *Domain Adaptation in Computer Vision Applications*, pages 189–209. Springer International Publishing, Cham. Series Title: Advances in Computer Vision and Pattern Recognition.

Hu, X., Li, S., Huang, T., Tang, B., Huai, R., and Chen, L. (2024). How simulation helps autonomous driving: A survey of sim2real, digital twins, and parallel intelligence. *IEEE Transactions on Intelligent Vehicles*, 9(1):593–612.

Jin, X., Park, Y., Maddix, D. C., Wang, H., and Wang, Y. (2022). Domain Adaptation for Time Series Forecasting via Attention Sharing.

Kim, S., Min, Y., Jung, Y., and Kim, S. (2022). Controllable Style Transfer via Test-time Training of Implicit Neural Representation. arXiv:2210.07762 [cs].

Kottapalli, S. R. K., Hubli, K., Chandrashekhara, S., Jain, G., Hubli, S., Botla, G., and Doddaiah, R. (2025). Foundation Models for Time Series: A Survey. arXiv:2504.04011 [cs].

Li, M., Liu, K., Chen, H., Bu, J., Wang, H., and Wang, H. (2025). TSINR: Capturing Temporal Continuity via Implicit Neural Representations for Time Series Anomaly Detection. arXiv:2411.11641 [cs].

Luigi, L. D., Cardace, A., Spezialetti, R., Ramirez, P. Z., Salti, S., and Stefano, L. D. (2023). Deep Learning on Implicit Neural Representations of Shapes. arXiv:2302.05438 [cs].

Ma, H., Chen, Y., Zhao, W., Yang, J., Ji, Y., Xu, X., Liu, X., Jing, H., Liu, S., and Yang, G. (2024). A Mamba Foundation Model for Time Series Forecasting. arXiv:2411.02941 [cs].

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., and Ng, R. (2020). NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. arXiv:2003.08934 [cs].

Naour, E. L., Serrano, L., Migus, L., Yin, Y., Agoua, G., Baskiotis, N., Gallinari, P., and Guigue, V. (2024). TIMEFLOW: An implicit neural representation approach for continuous time series modeling.

Oord, A. v. d., Kalchbrenner, N., Vinyals, O., Espeholt, L., Graves, A., and Kavukcuoglu, K. (2016). Conditional Image Generation with PixelCNN Decoders. arXiv:1606.05328 [cs].

Oreshkin, B. N., Rodriguez, P., and Lacoste, A. (2019). TADAM: Task dependent adaptive metric for improved few-shot learning. arXiv:1805.10123 [cs].

Painblanc, F., Chapel, L., Courty, N., Friguet, C., Pelletier, C., and Tavenard, R. (2023). Match-And-Deform: Time Series Domain Adaptation through Optimal Transport and Temporal Alignment. arXiv:2308.12686 [cs].

Pan, S. J. and Yang, Q. (2010). A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359.

Park, J. J., Florence, P., Straub, J., Newcombe, R., and Lovegrove, S. (2019). DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. arXiv:1901.05103 [cs].

Perez, E., Strub, F., Vries, H. d., Dumoulin, V., and Courville, A. (2017). FiLM: Visual Reasoning with a General Conditioning Layer. arXiv:1709.07871 [cs].

Ramasinghe, S., MacDonald, L. E., and Lucey, S. (2022). On the Frequency-bias of Coordinate-MLPs. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35, pages 796–809. Curran Associates, Inc.

Redko, I., Morvant, E., Habrard, A., Sebban, M., and Bennani, Y. (2022). A SURVEY ON DOMAIN ADAPTATION THEORY: LEARNING BOUNDS AND THEORETICAL GUARANTEES.

Shi, X., Wang, S., Nie, Y., Li, D., Ye, Z., Wen, Q., and Jin, M. (2025). Time-MoE: Billion-Scale Time Series Foundation Models with Mixture of Experts. arXiv:2409.16040 [cs].

Sitzmann, V., Martel, J. N. P., Bergman, A. W., Lindell, D. B., and Wetzstein, G. (2020). Implicit Neural Representations with Periodic Activation Functions.

Strümpler, Y., Postels, J., Yang, R., Gool, L. v., and Tombari, F. (2022). Implicit Neural Representations for Image Compression. arXiv:2112.04267 [eess].

Sun, B. and Saenko, K. (2016). Deep CORAL: Correlation Alignment for Deep Domain Adaptation. In Hua, G. and Jégou, H., editors, *Computer Vision – ECCV 2016 Workshops*, volume 9915, pages 443–450. Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.

Teubert, C., Griffith, K. J., Corbetta, M., Kulkarni, C., Banerjee, P., Watkins, J., and Daigle, M. (2025). ProgPy Python Prognostics Packages.

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. (2017). Adversarial Discriminative Domain Adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2962–2971, Honolulu, HI. IEEE.

Vapnik, V. and Vashist, A. (2009). A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5):544–557.

Wang, J., Chen, Y., Feng, W., Yu, H., Huang, M., and Yang, Q. (2020). Transfer Learning with Dynamic Distribution Adaptation. *ACM Transactions on Intelligent Systems and Technology*, 11(1):1–25.

Wang, Z., Li, X., Yang, J., Liu, Y., and Jiang, S. (2024). Sim-to-Real Transfer via 3D Feature Fields for Vision-and-Language Navigation. arXiv:2406.09798 [cs].

Weber, M., Auch, M., Doblander, C., Mandl, P., and Jacobsen, H.-A. (2021). Transfer Learning With Time Series Data: A Systematic Mapping Study. *IEEE Access*, 9:165409–165432.

Wilson, G., Doppa, J. R., and Cook, D. J. (2020). Multi-Source Deep Domain Adaptation with Weak Supervision for Time-Series Sensor Data. arXiv:2005.10996 [cs].

Xiao, C., Zhou, J., Xiao, Y., Lu, X., Zhang, L., and Xiong, H. (2025). TimeFound: A Foundation Model for Time Series Forecasting. arXiv:2503.04118 [cs].

Yin, P., Westenbroek, T., Bagaria, S., Huang, K., Cheng, C.-a., Kobolov, A., and Gupta, A. (2025). Rapidly Adapting Policies to the Real World via Simulation-Guided Fine-Tuning. arXiv:2502.02705 [cs].

Zhao, W., Queralta, J. P., and Westerlund, T. (2020). Sim-to-Real Transfer in Deep Reinforcement Learning for Robotics: a Survey. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 737–744. arXiv:2009.13303 [cs].

Zhong, E. D., Bepler, T., Davis, J. H., and Berger, B. (2020). Reconstructing continuous distributions of 3D protein structure from cryo-EM images. arXiv:1909.05215 [q-bio].

# A Alignment Strategies

## A.1 Adversarial alignment

Adversarial alignment relies on a discriminator trained to distinguish between synthetic and real latent representations. The alignment objective is defined using a binary cross-entropy loss applied to the discriminator logits.

$$\mathcal{L}_{\text{adv}} = -\sum_i \left[ y_i \log \sigma(f_i) + (1 - y_i) \log \left(1 - \sigma(f_i)\right) \right], \tag{14}$$

where $y_i = 0$ denotes synthetic samples, $y_i = 1$ denotes real samples, $f_i$ is the discriminator logit, and $\sigma(\cdot)$ is the sigmoid function.

### A.2 CORAL alignment

Correlation Alignment (CORAL) minimizes the discrepancy between second-order statistics of the latent representations from both domains. The loss is defined as

$$\mathcal{L}_{\text{CORAL}} = \frac{1}{4d^2} \|C_s - C_r\|_F^2 , \tag{15}$$

where $C_s, C_r \in \mathbb{R}^{d\times d}$ are the unbiased sample covariance matrices computed from the INR latent representations of the synthetic and real branches, respectively, and $\|\cdot\|_F$ denotes the Frobenius norm.

### A.3 MMD alignment.

Maximum Mean Discrepancy (MMD) aligns feature distributions in a reproducing kernel Hilbert space using a radial basis function (RBF) kernel with bandwidth $\sigma = 1$. The loss is given by

$$\mathcal{L}_{\text{MMD}} = \mathbb{E}_{x,x'}[k(x, x')] + \mathbb{E}_{y,y'}[k(y, y')] - 2\,\mathbb{E}_{x,y}[k(x, y)], \tag{16}$$

where $x, x' \sim z_{\text{synth}}$, $y, y' \sim z_{\text{real}}$, and the kernel function is

$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right). \tag{17}$$