TEST-TIME ADAPTATION OF HIGH-DIMENSIONAL SIMULATION SURROGATES VIA D-OPTIMAL STATISTICS

Anna Zimmel[1] **Paul Setinek**[1] **Gianluca Galletti**[1] **Johannes Brandstetter**[12] **Werner Zellinger**[1]
[1]ELLIS Unit, LIT AI Lab, Institute for Machine Learning, JKU Linz, Austria
[2]Emmi AI GmbH, Linz, Austria
zimmel@ml.jku.at

ABSTRACT

Machine learning surrogates are increasingly used in engineering to accelerate costly simulations, yet distribution shifts between training and deployment often cause performance degradation (e.g., unseen geometries or configurations). Test-Time Adaptation (TTA) can mitigate such shifts, but existing methods are largely developed for lower-dimensional classification with structured outputs and visually aligned input-output relationships, making them unstable for the high-dimensional, unstructured and regression problems common in simulation. We address this challenge by proposing a TTA framework based on storing maximally informative (D-optimal) statistics, which jointly enables stable adaptation and principled parameter selection at test time. When applied to pretrained simulation surrogates, our method yields up to 7% out-of-distribution improvements at negligible computational cost. To the best of our knowledge, this is the first systematic demonstration of effective TTA for high-dimensional simulation regression and generative design optimization, validated on SIMSHIFT and EngiBench benchmarks.

1 INTRODUCTION

Neural surrogates have become powerful tools for accelerating Partial Differential Equation (PDE) simulations across engineering and science. They perform well when test conditions match the training data, but performance often drops on unseen configurations, i.e., when the data distribution shifts. This challenge becomes more evident in industrial simulation and design optimization, where configurations can vary widely across iterations.

Tackling distribution shifts (Quinonero-Candela et al., 2008) is central to various long-standing research directions (Ben-David et al., 2006; Blanchard et al., 2021; Hochreiter et al., 2001; Hospedales et al., 2021; Settles, 2009). For engineering tasks, where rapid adaptation is essential, and target domain distributions are unavailable a priori, Test-Time Adaptation (TTA) offers an efficient solution (Liang et al., 2020; Sun et al., 2020b; Wang et al., 2021).

However, existing TTA methods focus on classification tasks (Wang et al., 2021; Liang et al., 2020), while little research exists for high-dimensional regression (Liang et al., 2024). One recent exception is Significant-Subspace Alignment (SSA) (Adachi et al., 2025), which handles both classification and one-dimensional regression tasks. Other related TTA methods for high-dimensional outputs are mainly developed for vision tasks such as depth-estimation (Liu et al., 2023), super-resolution (Park et al., 2020; Deng et al., 2023), and image dehazing (Liu et al., 2022). Unfortunately, these approaches rely on image-specific inductive biases and regular grids, which do not transfer to unstructured, non-Euclidean engineering simulations. As a consequence, classical TTA methods often cannot overcome the severe instabilities in our considered problem setting.

We therefore introduce (to the best of our knowledge first) TTA framework, Stable Adaptation at Test-Time for Simulation (SATTS), targeting neural surrogates for high-dimensional engineering tasks. At the core of our approach is the use of maximally informative source statistics to stabilize the adaptation process, which we achieve via D-optimal (Atkinson & Donev, 1992) sample selection. This approach allows us to compress the source manifold into a small set of source statistics for realizing three core properties for robust TTA: (i) feature alignment, (ii) preservation of source

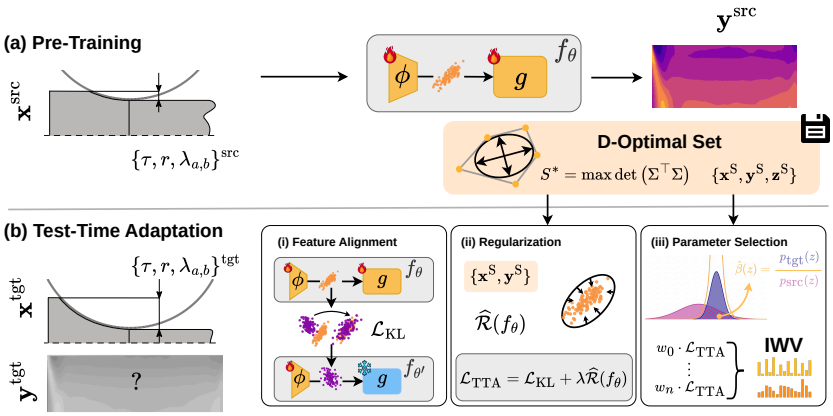

Figure 1: SATTS applied to *hot rolling* task from (Setinek et al., 2025). (a) Pre-training on the source domain with fixed input parameters (τ, r, λ_a, λ_b). The *representation learner* ϕ and the *predictor* g are optimized, and maximally informative (D-optimal) statistics are computed. (b) Test-time adaptation of ϕ using D-optimal statistics for realizing three TTA pillars: adaptation (feature alignment), source knowledge preservation (statistics-based regularization), and parameter tuning (importance weighted validation).

domain knowledge, and (iii) unsupervised tuning of adaptation hyperparameters. We summarize our contributions as follows:

- *Problem:* We are (to the best of our knowledge) the first one applying TTA to high-dimensional simulation regression.
- *Method:* We propose a novel adaptation framework that relies on D-optimal source statistics and stabilizes three main components: feature alignment, source knowledge preservation, and parameter tuning.
- *Performance:* We demonstrate in Table 1 and 2 that our approach reliably outperforms standard TTA methods on diverse engineering adaptation benchmarks, SIMSHIFT for high-dimensional regression and EngiBench for design generation.

2 Problem

Following (Xiao & Snoek, 2024; Liang et al., 2024), we assume access to a regressor $f_\theta : \mathcal{X} \to \mathbb{R}^d$ pre-trained on *source* samples $(\mathbf{x}_i, \mathbf{y}_i)_{i=1}^{N^{\text{src}}} \in \mathcal{X} \times \mathbb{R}^d$ drawn from a source distribution P^{src}, e.g., $f_\theta = g \circ \phi$ in Fig. 1. We also assume access to some ground truth matrix-valued source statistics.

The goal is, for any new *unlabeled* sample $(\mathbf{x}_i^{\text{tgt}})_{i=1}^{N^{\text{tgt}}}$ drawn from the input marginal of a *target* distribution $P^{\text{tgt}} \neq P^{\text{src}}$, to find θ which minimizes the empirical target risk

$$\widehat{\mathcal{R}}_{\text{tgt}}(f_\theta) = \frac{1}{N^{\text{tgt}}} \sum_{i=1}^{N^{\text{tgt}}} \left\| f_\theta(\mathbf{x}_i^{\text{tgt}}) - \mathbf{y}_i^{\text{tgt}} \right\|_2^2 . \tag{1}$$

Note that we have no access to any target labels $(\mathbf{y}_i^{\text{tgt}})_{i=1}^{N^{\text{tgt}}}$ and the target risk in Eq. (1) cannot be directly evaluated.

TTA in simulation. We emphasize that our TTA setting differs from the usual, computer vision-oriented problems. In particular, simulation surrogates are more challenging, as the output dimension d of $f_\theta : \mathcal{X} \to \mathbb{R}^d$ can reach $O(10^6)$ in our regime. This is typical for neural surrogates in simulation, but less common in the vision domain. Moreover, adaptation relies on small unlabeled target batches $\{\mathbf{x}_i^{\text{tgt}}\}_{i=1}^{N_{\text{tgt}}}$ with $N_{\text{tgt}} \ll d$. Finally, vision tasks often present structured, visually aligned inputs and outputs. Conversely, simulation data usually lacks geometric correspondence between $\mathbf{x}_i$ and $\mathbf{y}_i$, as $\mathbf{x}_i$ is often just coordinates (Lu et al., 2021; Kovachki et al., 2021). This, together with the high dimensionality render standard TTA methods ill-conditioned for neural surrogates, and necessitates explicit methodological mechanisms to stabilize the adaptation process.

3 METHOD

3.1 MAXIMALLY INFORMATIVE STATISTICS

In high-dimensional settings, naive summary statistics (e.g., global means or full covariances) are insufficient to support reliable TTA, as their estimation becomes ill-conditioned in the presence of many low-information or spurious feature directions. We approach this by selecting a subset of latent representations that preserves the most informative structure of the source model (Zhang et al., 2023).

Data generating assumption. We follow the common assumption (Fernando et al., 2013; Sun & Saenko, 2016; Adachi et al., 2025) that $\mathbf{z} = \phi(\mathbf{x})$ is normally distributed for each domain, such that the feature distribution is fully characterized by its first- and second-order moments.

D-optimal latent statistics. Under this assumption, we select source samples that maximize the retained information in latent space via D-optimality (Atkinson & Donev, 1992). D-optimality identifies a subset of m samples whose (latent) representations span the most informative subspace of the original (feature) space. In our setting, letting $\mathbf{Z}_S \in \mathbb{R}^{m \times d}$ denote the matrix of latent features $\mathbf{z} = \phi(\mathbf{x})$ for a subset $S \subset \{1, \ldots, N\}$, we select

$$S^\star = \arg\max_{|S|=m} \det\left(\mathbf{Z}_S^\top \mathbf{Z}_S\right).$$

Note that when the latent features are centered, $\mathbf{Z}_S^\top \mathbf{Z}_S$ is proportional to the empirical covariance matrix of the selected samples. This means that maximizing the determinant of $\mathbf{Z}_S^\top \mathbf{Z}_S$ corresponds to maximizing the generalized variance of the retained latent representation.

3.2 SATTS

We term our approach Stable Adaptation at Test-Time for Simulation (SATTS). In SATTS, D-optimal statistics are used at three key TTA components: feature alignment, source knowledge preservation, and parameter tuning.

Feature alignment is used to reduce the dissimilarity between source and target distributions (see Section A). Following a recent TTA method for regression, SSA (Adachi et al., 2025), latent directions that strongly influence the prediction are the right candidates for alignment. In SSA, feature importance is defined for one-dimensional regression through manually selecting a significant subspace. We extend the idea by assigning a *positive importance weight* to every principal direction,

$$\alpha_k = 1 + \left\|\mathbf{W}\mathbf{v}_k^{\mathrm{src}}\right\|_2, \tag{2}$$

where $\mathbf{v}_k^{\mathrm{src}} \in \mathbb{R}^C$ denotes the k-th source principal component. At deployment, mean-centered target features $\mathbf{z}^{\mathrm{tgt}} = \phi(\mathbf{x}^{\mathrm{tgt}})$ are projected onto the source principal components $\mathbf{V}^{\mathrm{src}}$, and reweighted by $\boldsymbol{\alpha}$. Finally, source and target feature distributions are aligned by minimizing a channel-wise symmetric empirical KL-divergence $\mathcal{L}_{\mathrm{KL}}$ (see Eq. (6)).

Source knowledge preservation is realized by regularization on the subsampled source statistics:

$$\mathcal{L}_{\mathrm{TTA}} = \mathcal{L}_{\mathrm{KL}} + \lambda \widehat{\mathcal{R}}_{\mathrm{src}}(f_\theta) \tag{3}$$

with $\hat{\mathcal{R}}_{\mathrm{src}}$ denoting the empirical source risk estimated on the D-optimal samples and $\lambda > 0$ being a regularization parameter. This ensures that the feature alignment updates driven by $\mathcal{L}_{\mathrm{KL}}$ do not deviate significantly from the known solution.

Parameter tuning We integrate Importance Weighted Validation (IWV) (Shimodaira, 2000) using D-optimal samples to tune the test-time adaptation learning rate. Since the target risk in Eq. (1) cannot be computed directly without access to target labels, we estimate it via an importance-weighted source risk under the covariate shift assumption $p_{\mathrm{src}}(\mathbf{y} \mid \mathbf{x}) = p_{\mathrm{tgt}}(\mathbf{y} \mid \mathbf{x})$:

$$\hat{\mathcal{R}}_{\mathrm{tgt}}(f_\theta) \;\approx\; \frac{1}{m} \sum_{i=1}^{m} \hat{\beta}(\mathbf{z}_i) \left\| f_\theta(\mathbf{x}_i^S) - \mathbf{y}_i^S \right\|_2^2, \tag{4}$$

where $\{(\mathbf{x}_i^S, \mathbf{y}_i^S)\}_{i=1}^m$ denotes the set of D-optimal source samples and $\hat{\beta}(\mathbf{z}) = p_{\mathrm{tgt}}(\mathbf{z}) / p_{\mathrm{src}}(\mathbf{z})$ is the density ratio estimated in latent space $\mathbf{z} = \phi(\mathbf{x})$ under the Gaussian data generating assumption. Using the estimate in Eq. (4), we perform model selection via line search over the TTA learning rate, evaluating performance after each adaptation step and stopping once further updates no longer improve the objective.

Table 1: Comparison of current baselines with TTA methods for two simulation datasets. Results are averaged across 20 TTA runs, including standard deviation. RMSE is normalized over all fields.

(a) Rolling

Model	RMSE (↓)	MAE (↓)	R^2 (↑)
Source	$0.561_{\pm 0.001}$	$0.484_{\pm 0.001}$	$0.781_{\pm 0.001}$
TENT	$1.825_{\pm 0.002}$	$1.553_{\pm 0.002}$	$-0.371_{\pm 0.004}$
SSA	$0.566_{\pm 0.020}$	$0.481_{\pm 0.018}$	$0.811_{\pm 0.014}$
SATTS	$\mathbf{0.545_{\pm 0.019}}$	$\mathbf{0.466_{\pm 0.018}}$	$\mathbf{0.831_{\pm 0.012}}$
Oracle	$0.529_{\pm 0.013}$	$0.453_{\pm 0.012}$	$0.832_{\pm 0.011}$

(b) Forming

Model	RMSE (↓)	MAE (↓)	R^2 (↑)
Source	$0.161_{\pm 0.001}$	$0.066_{\pm 0.001}$	$0.979_{\pm 0.001}$
TENT	$1.251_{\pm 0.001}$	$0.639_{\pm 0.001}$	$-0.081_{\pm 0.001}$
SSA	$0.215_{\pm 0.005}$	$0.098_{\pm 0.003}$	$0.965_{\pm 0.002}$
SATTS	$\mathbf{0.157_{\pm 0.001}}$	$\mathbf{0.066_{\pm 0.001}}$	$\mathbf{0.980_{\pm 0.001}}$
Oracle	$0.156_{\pm 0.004}$	$0.067_{\pm 0.002}$	$0.980_{\pm 0.001}$

Table 2: Comparison of current baselines with TTA methods for design optimization datasets. Results are averaged across 20 TTA runs, over one model with standard deviation reported.

(a) Beams2D

Model	COMP (↓)	MAE (↓)	MMD (↓)
Source	$123.7_{\pm 17.854}$	$\mathbf{0.026_{\pm 0.004}}$	$\mathbf{0.052_{\pm 0.002}}$
SSA	$119.4_{\pm 4.586}$	$0.040_{\pm 0.005}$	$0.062_{\pm 0.003}$
SATTS	$\mathbf{118.8_{\pm 12.409}}$	$0.027_{\pm 0.004}$	$0.053_{\pm 0.002}$

(b) HeatConduction2D

Model	COMP (10^{-3})	MAE (↓)	MMD (↓)
Source	$0.577_{\pm 0.561}$	$0.336_{\pm 0.057}$	$0.095_{\pm 0.000}$
SSA	$0.712_{\pm 0.615}$	$0.349_{\pm 0.057}$	$0.095_{\pm 0.000}$
SATTS	$\mathbf{0.537_{\pm 0.491}}$	$\mathbf{0.334_{\pm 0.057}}$	$0.095_{\pm 0.000}$

4 EXPERIMENTS

We evaluate SATTS on two simulation benchmarks, SIMSHIFT (Setinek et al., 2025) and EngiBench (Felten et al., 2025). In all experiments, the parameters of the quasi D-optimal algorithm (see Algorithm 1) are fixed with $m = 8$ indices and a threshold of $\tau = 0.95\%$.

We first analyze adaptation behavior on the SIMSHIFT benchmark (Setinek et al., 2025). Table 1 summarizes the results across two datasets, comparing our method with SSA and Tent as established TTA baselines, as well as the unadapted source model (*Source*) and the target-optimal selection (*Oracle*). Implementation details are provided in Appendix F. Across all settings, SATTS consistently outperforms SSA and yields the strongest performance among all adaptation methods. While the absolute gains over the source model are modest, they are achieved without sacrificing stability. In contrast, both SSA and Tent frequently degrade performance relative to the pre-trained model, indicating a lack of robustness to these high-dimensional distribution shifts.

Additionally, we evaluate on two EngiBench design-optimization tasks: *structural beam bending* and *2D heat conduction*. By default, these datasets do not include predefined source and target domains. We therefore define them following the approach in Setinek et al. (2025), which we describe in Appendix G. We report Mean Absolute Error (MAE), the Maximum Mean Discrepancy (MMD), and Compliance (COMP), a dataset specific objective value calculated with a Finite Element Method (FEM) solver. In Table 2 we show across both tasks that our approach typically matches or reduces errors relative to the unregularized model. Compared with our method, SSA exhibits unstable behavior across certain metrics, sometimes degrading performance substantially. Such behavior is highly undesirable in TTA deployments and underlines the strong suit of our approach: its stability.

5 CONCLUSION AND FUTURE WORK

In this work, we take an initial step toward reliable test-time adaptation for neural surrogates and, more broadly, for high-dimensional multivariate regression. Our main methodological contribution is the use of D-optimal samples at three critical stages: feature alignment, regularization, and parameter tuning. The proposed adjustments enable TTA to achieve consistent zero-shot performance improvements at negligible computational cost.

However, analyzing the *"Oracle"* reveals clear opportunities for improvement. This shows potential for a new class of TTA algorithms, specifically designed for physics simulation data. We foresee two paths: (i) use physics-informed constraints and priors (Raissi et al., 2019; Cai et al., 2021) and (ii) incorporate uncertainty quantification to localize failure regions in the fields where adaptation is necessary.

ACKNOWLEDGMENTS

We wish to thank Stephanie Holly and Florian Sestak for their helpful discussions and feedback.

The ELLIS Unit Linz, the LIT AI Lab, the Institute for Machine Learning, are supported by the Federal State Upper Austria. We thank the projects FWF AIRI FG 9-N (10.55776/FG9), AI4GreenHeatingGrids (FFG- 899943), Stars4Waters (HORIZON-CL6-2021-CLIMATE-01-01), FWF Bilateral Artificial Intelligence (10.55776/COE12). We thank NXAI GmbH, Silicon Austria Labs (SAL), Merck Healthcare KGaA, GLS (Univ. Waterloo), TÜV Holding GmbH, Software Competence Center Hagenberg GmbH, dSPACE GmbH, TRUMPF SE + Co. KG.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of our approach SATTS, by describing the models, algorithm, and experimental setup in the main paper, with additional implementation details in the appendix. All datasets used in this work are publicly available, and we include the details on the data splits for EngiBench (Felten et al., 2025) in the Appendix G.

LLM USAGE DISCLOSURE

In general, LLM tools were used to refine writing in multiple parts of the paper, such as introduction, method and experiment sections (GPT-5). Claude3.5 and GPT-5 were also used to make visualizations more aesthetically pleasing, accelerate the development of plotting functions, and present results in tables. The literature review was done manually, with web searches and interaction from experts in each field, as language models would end up in unsatisfactory results most of the times. Overall, AI assistants were strictly used for editing and never in ideation beyond understanding other work.

REFERENCES

Kazuki Adachi, Shin'Ya Yamaguchi, and Atsutoshi Kumagai. Covariance-aware feature alignment with pre-computed source statistics for test-time adaptation to multiple image corruptions. In *2023 IEEE International Conference on Image Processing (ICIP)*, pp. 800–804. IEEE, 2023.

Kazuki Adachi, Shin'ya Yamaguchi, Atsutoshi Kumagai, and Tomoki Hamagami. Test-time adaptation for regression by subspace alignment. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

Benedikt Alkin, Andreas Fürst, Simon Schmid, Lukas Gruber, Markus Holzleitner, and Johannes Brandstetter. Universal physics transformers: A framework for efficiently scaling neural operators. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 25152–25194. Curran Associates, Inc., 2024.

Anthony C. Atkinson and Anatoly N. Donev. *Optimum Experimental Designs*. Oxford University Press, Oxford, 1992.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016. URL `https://arxiv.org/abs/1607.06450`.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19, 2006.

Gilles Blanchard, Aniket Anand Deshmukh, Urun Dogan, Gyemin Lee, and Clayton Scott. Domain generalization by marginal transfer learning. *Journal of machine learning research*, 22(2):1–55, 2021.

Florent Bonnet, Jocelyn Ahmed Mazari, Paola Cinnella, and Patrick Gallinari. AirfRANS: High fidelity computational fluid dynamics dataset for approximating reynolds-averaged navier–stokes solutions. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. URL `https://arxiv.org/abs/2212.07564`.

Shengze Cai, Zhiping Mao, Zhicheng Wang, Minglang Yin, and George Em Karniadakis. Physics-informed neural networks (pinns) for fluid mechanics: A review, 2021. URL https://arxiv.org/abs/2105.09506.

Chao Chen, Zhihang Fu, Zhihong Chen, Sheng Jin, Zhaowei Cheng, Xinyu Jin, and Xian-Sheng Hua. Homm: Higher-order moment matching for unsupervised domain adaptation, 2019. URL https://arxiv.org/abs/1912.11976.

Zeshuai Deng, Zhuokun Chen, Shuaicheng Niu, Thomas H. Li, Bohan Zhuang, and Mingkui Tan. Efficient test-time adaptation for super-resolution with second-order degradation and reconstruction. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

Cian Eastwood, Ian Mason, Christopher KI Williams, and Bernhard Schölkopf. Source-free adaptation to measurement shift via bottom-up feature restoration. *arXiv preprint arXiv:2107.05446*, 2021.

Florian Felten, Gabriel Apaza, Gerhard Bräunlich, Cashen Diniz, Xuliang Dong, Arthur Drake, Milad Habibi, Nathaniel J. Hoffman, Matthew Keeler, Soheyl Massoudi, Francis G. VanGessel, and Mark Fuge. Engibench: A framework for data-driven engineering design research, 2025. URL https://arxiv.org/abs/2508.00831.

Basura Fernando, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *2013 IEEE International Conference on Computer Vision*, pp. 2960–2967, 2013. doi: 10.1109/ICCV.2013.368.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint*, 2017. arXiv:1703.03400.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks, 2015.

Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, 4th edition, 2013.

Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola. A kernel method for the two-sample-problem. In B. Schölkopf, J. Platt, and T. Hoffman (eds.), *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006. URL https://proceedings.neurips.cc/paper_files/paper/2006/file/e9fb2eda3d9c55a0d89c98d6c54b5b3e-Paper.pdf.

Yufan He, Aaron Carass, Lianrui Zuo, Blake E. Dewey, and Jerry L. Prince. Autoencoder based self-supervised test-time adaptation for medical image analysis. *Medical Image Analysis*, 72: 102136, 2021. ISSN 1361-8415. doi: https://doi.org/10.1016/j.media.2021.102136. URL https://www.sciencedirect.com/science/article/pii/S1361841521001821.

Sepp Hochreiter, A Steven Younger, and Peter R Conwell. Learning to learn using gradient descent. In *International conference on artificial neural networks*, pp. 87–94. Springer, 2001.

Markus Holzleitner, Sergei V Pereverzyev, and Werner Zellinger. Domain generalization by functional regression. *Numerical Functional Analysis and Optimization*, 45(3):259–281, 2024.

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

Masato Ishii and Masashi Sugiyama. Source-free domain adaptation via distributional alignment by matching batch normalization statistics. *arXiv preprint arXiv:2101.10842*, 2021.

Sanghun Jung, Jungsoo Lee, Nanhee Kim, Amirreza Shaban, Byron Boots, and Jaegul Choo. Cafa: Class-aware feature alignment for test-time adaptation. In *International Conference on Computer Vision (ICCV)*, pp. 19060–19071, 2023.

Neerav Karani, Ertunc Erdil, Krishna Chaitanya, and Ender Konukoglu. Test-time adaptable neural networks for robust medical image segmentation. *Medical Image Analysis*, 68:101907, February 2021. ISSN 1361-8415. doi: 10.1016/j.media.2020.101907. URL http://dx.doi.org/10.1016/j.media.2020.101907.

Takeshi Kojima, Yutaka Matsuo, and Yusuke Iwasawa. Robustifying vision transformer without retraining from scratch by test-time class-conditional feature alignment. *arXiv preprint arXiv:2206.13951*, 2022.

Nikola B. Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew M. Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces. *CoRR*, abs/2108.08481, 2021. URL https://arxiv.org/abs/2108.08481.

David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. *arXiv preprint*, 1994. arXiv:cmp-lg/9407020.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Learning to generalize: Meta-learning for domain generalization, 2017. URL https://arxiv.org/abs/1710.03463.

Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. *ArXiv*, abs/1603.04779, 2016. URL https://api.semanticscholar.org/CorpusID:5069968.

Zongyi Li, Nikola B. Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew M. Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. *CoRR*, abs/2003.03485, 2020.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International conference on machine learning (ICML)*, pp. 6028–6039. PMLR, 2020.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation, 2021. URL https://arxiv.org/abs/2002.08546.

Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, 133(1):31–64, July 2024. ISSN 1573-1405. doi: 10.1007/s11263-024-02181-w. URL http://dx.doi.org/10.1007/s11263-024-02181-w.

Huan Liu, Zijun Wu, Liangyan Li, Sadaf Salehkalaibar, Jun Chen, and Keyan Wang. Towards multi-domain single image dehazing via test-time training. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 5821–5830. IEEE, 2022. doi: 10.1109/CVPR52688.2022.00574. URL https://doi.org/10.1109/CVPR52688.2022.00574.

Huan Liu, Zhixiang Chi, Yuanhao Yu, Yang Wang, Jun Chen, and Jin Tang. Meta-auxiliary learning for future depth prediction in videos. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2023, Waikoloa, HI, USA, January 2-7, 2023*, pp. 5745–5754. IEEE, 2023. doi: 10.1109/WACV56688.2023.00571. URL https://doi.org/10.1109/WACV56688.2023.00571.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, March 2021. ISSN 2522-5839. doi: 10.1038/s42256-021-00302-5. URL http://dx.doi.org/10.1038/s42256-021-00302-5.

Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In Sanjoy Dasgupta and David McAllester (eds.), *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 10–18, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR. URL https://proceedings.mlr.press/v28/muandet13.html.

Daniel Musekamp, Marimuthu Kalimuthu, David Holzmüller, Makoto Takamoto, and Mathias Niepert. Active learning for neural PDE solvers. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=x4ZmQaumRg.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International conference on machine learning*, pp. 16888–16905. PMLR, 2022.

Seobin Park, Jinsu Yoo, Donghyeon Cho, Jiwon Kim, and Tae Hyun Kim. Fast adaptation to super-resolution networks via meta-learning. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXVII*, volume 12372 of *Lecture Notes in Computer Science*, pp. 754–769. Springer, 2020. doi: 10.1007/978-3-030-58583-9_45. URL https://doi.org/10.1007/978-3-030-58583-9_45.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4196–4206, 2023. URL https://arxiv.org/abs/2212.09748.

Joaquin Quinonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D Lawrence. *Dataset Shift in Machine Learning*. MIT Press, 2008.

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. doi: 10.1016/j.jcp.2018.10.045.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells III, and Alejandro F. Frangi (eds.), *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5 - 9, 2015, Proceedings, Part III*, volume 9351 of *Lecture Notes in Computer Science*, pp. 234–241. Springer, 2015. doi: 10.1007/978-3-319-24574-4_28. URL https://doi.org/10.1007/978-3-319-24574-4_28.

Paul Setinek, Gianluca Galletti, Thomas Gross, Dominik Schnürer, Johannes Brandstetter, and Werner Zellinger. Simshift: A benchmark for adapting neural surrogates to distribution shifts, 2025. URL https://arxiv.org/abs/2506.12007.

Burr Settles. Active learning literature survey. 2009.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000. ISSN 0378-3758. doi: https://doi.org/10.1016/S0378-3758(00)00115-4. URL https://www.sciencedirect.com/science/article/pii/S0378375800001154.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation, 2016. URL https://arxiv.org/abs/1607.01719.

Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation, 2015. URL https://arxiv.org/abs/1511.05547.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9229–9248. PMLR, 13–18 Jul 2020a. URL https://proceedings.mlr.press/v119/sun20b.html.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning (ICML)*, pp. 9229–9248. PMLR, 2020b.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts, 2020c. URL https://arxiv.org/abs/1909.13231.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, Tatsunori Hashimoto, and Carlos Guestrin. Learning to (learn at test time): Rnns with expressive hidden states, 2025. URL https://arxiv.org/abs/2407.04620.

Artur Toshev, Harish Ramachandran, Jonas A. Erbesdobler, Gianluca Galletti, Johannes Brandstetter, and Nikolaus A. Adams. JAX-SPH: A differentiable smoothed particle hydrodynamics framework. In *ICLR 2024 Workshop on AI4DifferentialEquations In Science*, 2024. URL https://openreview.net/forum?id=8X5PXVmsHW.

Artur P. Toshev, Gianluca Galletti, Fabian Fritz, Stefan Adami, and Nikolaus A. Adams. Lagrangebench: a lagrangian fluid mechanics benchmarking suite. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, 2023.

Jeya Maria Jose Valanarasu, Pengfei Guo, Vibashan VS, and Vishal M. Patel. On-the-fly test-time adaptation for medical image segmentation. In *MIDL (Medical Imaging with Deep Learning)*, 2023. Episodic, zero-shot adaptation with adaptive batch-normalization.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations (ICLR)*, 2021.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for PDEs on general geometries. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53681–53705. PMLR, 21–27 Jul 2024a. URL https://proceedings.mlr.press/v235/wu24r.html.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for pdes on general geometries. In *International Conference on Machine Learning*, 2024b.

Zehao Xiao and Cees GM Snoek. Beyond model adaptation at test time: A survey. *arXiv preprint arXiv:2411.03687*, 2024.

Shiqi Yang, Yaxing Wang, Joost van de Weijer, Luis Herranz, and Shangling Jui. Exploiting the intrinsic neighborhood structure for source-free domain adaptation, 2021. URL https://arxiv.org/abs/2110.04202.

Werner Zellinger, Thomas Grubinger, Edwin Lughofer, Thomas Natschläger, and Susanne Saminger-Platz. Central moment discrepancy (cmd) for domain-invariant representation learning, 2019. URL https://arxiv.org/abs/1702.08811.

Marvin Zhang, Sergey Levine, and Chelsea Finn. Memo: Test time robustness via adaptation and augmentation. *Advances in neural information processing systems*, 35:38629–38642, 2022.

Shihao Zhang, Linlin Yang, Michael Bi Mi, Xiaoxu Zheng, and Angela Yao. Improving deep regression with ordinal entropy. *International Conference on Learning Representations*, 2023.

Bowen Zhao, Chen Chen, and Shu-Tao Xia. Delta: degradation-free fully test-time adaptation. *arXiv preprint arXiv:2301.13018*, 2023.

Aurick Zhou and Sergey Levine. Bayesian adaptation for covariate shift. *Advances in neural information processing systems*, 34:914–927, 2021.

A Related Work

Neural surrogates have emerged as a widely used approach to accelerate traditional numerical simulation methods by providing fast approximations of the solutions. In general, surrogate models are trained on the solutions from numerical solvers, paired with the corresponding initial conditions and configurations under which they were generated, e.g., Setinek et al. (2025); Bonnet et al. (2022); Toshev et al. (2023; 2024). A particularly prominent line of work within neural surrogate modeling for PDEs is operator learning (Kovachki et al., 2021; Li et al., 2020; Lu et al., 2021; Alkin et al., 2024; Wu et al., 2024b). Such models aim to directly approximate the solution operator that maps initial functions (conditions and input terms) to output functions.

Test-Time Adaptation (TTA) refers to the emerging machine learning technique of adapting a pre-trained model to unlabeled target data, directly at inference time and prior to generating predictions. For this reason, TTA has recently attracted increasing attention as it can offer (nearly) free performance gains (Liang et al., 2024). While the majority of existing TTA methods have been developed for low-dimensional classification tasks (Liang et al., 2021; Yang et al., 2021), employing methodologies such as entropy minimization (Wang et al., 2021; Zhou & Levine, 2021; Niu et al., 2022; Zhang et al., 2022; Zhao et al., 2023) and feature alignment (Ishii & Sugiyama, 2021; Kojima et al., 2022; Eastwood et al., 2021; Adachi et al., 2023; Jung et al., 2023), recent works have begun to extend these ideas to image segmentation (Valanarasu et al., 2023; He et al., 2021; Karani et al., 2021). Research in TTA tackling regression problems is much sparser. Significant-Subspace Alignment (SSA) (Adachi et al., 2025) moves into this direction, showing positive performance in the one-dimensional cases. Additionally, there are specialized methods designed for image regression tasks such as depth-estimation (Liu et al., 2023), super-resolution (Park et al., 2020; Deng et al., 2023), or image dehazing (Liu et al., 2022). Finally, TTA should not be confused with Test-Time Training (TTT), often used in time series literature (Sun et al., 2020a; Wang et al., 2021; Sun et al., 2025; 2020c). While both solve the same problem, TTT typically refers to methods that employ time-series specific techniques, for example, updating hidden states during sequential inference.

Covariance alignment of latent feature distributions is common practice in Unsupervised Domain Adaptation (UDA) and TTA (Sun & Saenko, 2016; Sun et al., 2015; Li et al., 2016; Wang et al., 2021). Even though this can be extended to higher-order moments (Zellinger et al., 2019; Chen et al., 2019), second-order alignment often already achieves stable performance across datasets.

Domain generalization, meta-learning, and active learning represent alternative strategies that can be used to improve model robustness and generalization under distribution shifts. Domain generalization (Muandet et al., 2013; Li et al., 2017; Holzleitner et al., 2024) and UDA (Sun & Saenko, 2016; Gretton et al., 2006; Zellinger et al., 2019; Ganin et al., 2015) can be effective in some scenarios, however their reliance on specific training, model selection and diverse training distributions limits their applicability. Meta-learning methods (Hochreiter et al., 2001; Finn et al., 2017) and active learning (Lewis & Gale, 1994; Musekamp et al., 2025) are similarly motivated, but generally assume access to ground-truth information in the shifted domain. In our setting, all these approaches face a significant practical limitation: none of them can quickly adapt a pre-trained model leveraging unlabeled data at test-time, as they all rely on a priori knowledge and training. This motivates our exploration of TTA as a more suitable solution.

B Additional Method Material

This section contains additional methodological details on our approach Stable Adaptation at Test-Time for Simulation (SATTS).

D-Optimal Sampling Algorithm For tractability, we follow the approach of approximating the D-optimal criterion using QR pivoting on whitened principal components (Golub & Van Loan, 2013). See Algorithm 1 for the pseudocode of our Quasi D-optimal selection criteria.

KL Loss Function

$$\mathcal{L}_{\mathrm{KL}} = \frac{1}{2}\sum_{k=1}^{K}\left(\frac{(\tilde{\mu}_k^{\mathrm{tgt}})^2 + \lambda_k^{\mathrm{src}}}{\tilde{\sigma}_k^{\mathrm{tgt2}}} + \frac{(\tilde{\mu}_k^{\mathrm{tgt}})^2 + \tilde{\sigma}_k^{\mathrm{tgt2}}}{\lambda_k^{\mathrm{src}}} - 2\right). \tag{5}$$

Algorithm 1 Quasi D-optimal spanning set selection via PCA and QR pivoting.

Require: Inputs $\mathbf{x}^{src}$, eigendecomposition $(\boldsymbol{\lambda}, \mathbf{V})$ of $\phi(\mathbf{x}^{src})$, variance threshold τ, number of quasi D-optimal designs m
Ensure: Selected source dataset indices $S \subseteq \{1, \ldots, N\}$
$\mathbf{Z} \leftarrow \phi(\mathbf{x}^{src})$
$\mathbf{Z} \leftarrow \mathbf{Z} - \text{mean}(\mathbf{Z})$
$r \leftarrow \text{select\_components}(\boldsymbol{\lambda}, \tau)$ ▷ keep $\tau\%$ variance
$\mathbf{Y} \leftarrow \mathbf{Z}\mathbf{V}_{:,1:r}\boldsymbol{\Lambda}_r^{-1/2}$
$\mathbf{Q}, \mathbf{R}, \text{piv} \leftarrow \text{QR}(\mathbf{Y}^T)$
$S \leftarrow \text{piv}_{1:m}$
return S

C Dataset Description

Our evaluation is conducted on two simulation benchmarks, SIMSHIFT (Setinek et al., 2025) and EngiBench (Felten et al., 2025). SIMSHIFT is designed to evaluate how surrogate models adapt to distribution shifts on real-world industrial simulation tasks, while EngiBench is a collection of design optimization datasets, optimizers, and simulators to evaluate designs. In both benchmarks, the inputs $\mathbf{x}$ represent parameters like geometry, material properties, desired or operating conditions. The "labels" $\mathbf{y}$ correspond to high-dimensional fields such as stresses or deformation for SIMSHIFT, and material density of the generated design for EngiBench. In both cases, the target distributions are generated from unseen parameter configurations, and the goal is to predict the corresponding fields outside the training set. While SIMSHIFT formulates the problem as a regression task with neural operators (Kovachki et al., 2021), EngiBench treats it as an inverse problem solved by generative models. The diversity in task formulation and training paradigm across the two benchmarks highlights the model-agnostic nature of our method.

D Additional Results

D.1 SIMSHIFT Results

As described in Section 4, we conduct experiments on the SIMSHIFT benchmark (Setinek et al., 2025). All datasets have explicit source and target domain splits. Shifts happen in parametric space, as opposed to unstructured variations occurring in images. We conduct all experiments using the medium-difficulty domain-shift. For a detailed description of the datasets, their creation, and the defined distribution shifts, we refer the reader to the SIMSHIFT publication (Setinek et al., 2025).

Table 3: Comparison of current baselines with TTA methods for all simulation datasets. Results are averaged across 20 TTA runs, over a pretrained model with standard deviation reported. Reported RMSE is normalized over all fields.

(a) Rolling

Model	RMSE (↓)	MAE (↓)	R^2 (↑)
Source	$0.561_{\pm 0.001}$	$0.484_{\pm 0.001}$	$0.781_{\pm 0.001}$
TENT	$1.825_{\pm 0.002}$	$1.553_{\pm 0.002}$	$-0.371_{\pm 0.004}$
SSA	$0.566_{\pm 0.020}$	$0.481_{\pm 0.018}$	$0.811_{\pm 0.014}$
SATTS	$\mathbf{0.545_{\pm 0.019}}$	$\mathbf{0.466_{\pm 0.018}}$	$\mathbf{0.831_{\pm 0.012}}$
Oracle	$0.529_{\pm 0.013}$	$0.453_{\pm 0.012}$	$0.832_{\pm 0.011}$

(b) Motor

Model	RMSE (↓)	MAE (↓)	R^2 (↑)
Source	$0.109_{\pm 0.001}$	$0.058_{\pm 0.001}$	$0.989_{\pm 0.001}$
TENT	$1.132_{\pm 0.032}$	$0.753_{\pm 0.026}$	$-0.152_{\pm 0.065}$
SSA	$0.336_{\pm 0.001}$	$0.172_{\pm 0.006}$	$0.881_{\pm 0.008}$
SATTS	$\mathbf{0.109_{\pm 0.003}}$	$\mathbf{0.058_{\pm 0.001}}$	$\mathbf{0.989_{\pm 0.000}}$
Oracle	$0.108_{\pm 0.001}$	$0.058_{\pm 0.001}$	$0.989_{\pm 0.001}$

(c) Forming

Model	RMSE (↓)	MAE (↓)	R^2 (↑)
Source	$0.161_{\pm 0.001}$	$0.066_{\pm 0.001}$	$0.979_{\pm 0.001}$
TENT	$1.251_{\pm 0.001}$	$0.639_{\pm 0.001}$	$-0.081_{\pm 0.001}$
SSA	$0.215_{\pm 0.005}$	$0.098_{\pm 0.003}$	$0.965_{\pm 0.002}$
SATTS	$\mathbf{0.157_{\pm 0.001}}$	$\mathbf{0.066_{\pm 0.001}}$	$\mathbf{0.980_{\pm 0.001}}$
Oracle	$0.156_{\pm 0.004}$	$0.067_{\pm 0.002}$	$0.980_{\pm 0.001}$

(d) Heatsink

Model	RMSE (↓)	MAE (↓)	R^2 (↑)
Source	$0.747_{\pm 0.001}$	$0.565_{\pm 0.001}$	$0.237_{\pm 0.001}$
TENT	$0.876_{\pm 0.001}$	$0.694_{\pm 0.0}$	$-0.203_{\pm 0.007}$
SSA	$0.746_{\pm 0.001}$	$0.552_{\pm 0.001}$	$0.227_{\pm 0.001}$
SATTS	$\mathbf{0.738_{\pm 0.004}}$	$\mathbf{0.545_{\pm 0.003}}$	$\mathbf{0.244_{\pm 0.007}}$
Oracle	$0.732_{\pm 0.035}$	$0.541_{\pm 0.03}$	$0.265_{\pm 0.065}$

D.2 Ablations

Component Analysis To assess the contribution of D-optimal source selection and importance weighting, we perform an incremental ablation study. Starting from the existing alignment strategy SSA, we isolate the effect of D-optimal source importance weighting from source selection. For both the SSA baseline and the source importance weighting, we use the original parameter values ($lr = 0.01$) from Adachi et al. (2025). Results on the SIMSHIFT benchmark in Table 4 show that each incremental addition improves performance over the previous configuration.

Table 4: RMSE scores of SATTS with and without importance weighting and model selection. The baseline and IWV results were evaluated with $lr = 0.01$. Best scores are bolded.

$\hat{\mathcal{R}}_{\text{src}}(f_\theta)$	IWV	Rolling	Motor	Forming
		$0.566_{\pm 0.020}$	$0.336_{\pm 0.000}$	$0.215_{\pm 0.005}$
✓		$0.550_{\pm 0.020}$	$0.204_{\pm 0.010}$	$0.195_{\pm 0.005}$
✓	✓	$\mathbf{0.545_{\pm 0.019}}$	$\mathbf{0.109_{\pm 0.000}}$	$\mathbf{0.157_{\pm 0.001}}$
Source		$0.561_{\pm 0.001}$	$0.109_{\pm 0.001}$	$0.161_{\pm 0.001}$

Parameter Selection Beyond IWV, UDA provides several alternative strategies for model selection. A commonly used baseline is *source-best* selection, in which the model with the lowest loss on source samples is chosen. Comparing these two methods in Table 5, it becomes visible that IWV substantially stabilizes naive source-based selection. Especially when the gap between the distributions is not too large, source-best exhibits high variance and thereby selects optimal results. This is not the case for IWV, where only results that are on par with or better than the source model are selected.

Table 5: RMSE comparison of two model selection algorithms: IWV and *source-best* on the SIMSHIFT dataset. Best scores are bolded.

Selection Method	Rolling	Motor	Forming
Source Best	0.550	0.203	0.157
IWV	**0.545**	**0.109**	**0.157**

Compute Compared to SSA, our method adds a moderate computational overhead. At test-time, D-optimal source samples are forwarded through the network to estimate the density ratios used in the regularization term. This introduces only small memory overhead, since the source samples can be fed jointly with the target batch. The main source of additional runtime comes from the source regularization term, which increases the size of the computational graph. Overall, we observe an approximately $1.88\times$ increase in runtime compared to SSA. Our proposed learning rate sweeps can be executed in parallel, therefore they do not add significant runtime overhead. Table 6 provides an empirical runtime comparison.

Table 6: Runtime comparison between SSA and SATTS on the Rolling dataset, highlighting the additional overhead of the proposed method. Mean $\pm$ std across 10 runs.

TTA Method	Runtime	Increase
SSA	$0.472_{\pm 0.053}$	
SATTS	$0.889_{\pm 0.085}$	$(\uparrow 1.88\times)$

E Supplementary Information

Significant-Subspace Alignment is a TTA method for one-dimensional regression (Adachi et al., 2025). It consists of two steps: *feature alignment* and *significant-subspace alignment*. In the first step, source statistics such as mean μ^{src} and covariance Σ^{src} are computed after source training. In the second step, a significant subspace is detected by manually selecting the top eigenvalues λ_k of

the source covariance Σ^{src}. Each subspace direction v_k^{src} is then weighted by its influence on the regression output:

$$\boldsymbol{\alpha}_k = 1 + |\mathbf{w}^\top \mathbf{v}_k^{\text{src}}|,$$

where $\boldsymbol{\alpha}_k \geq 1$ ensures that dimensions that strongly affect the regression output are emphasized.

At test time, the precomputed source statistics are used to project the target features into the significant subspace. From the projected target features, their mean and variance $(\tilde{\mu}_k^{\text{tgt}}, \tilde{\sigma}_k^{\text{tgt}2})$ are calculated and aligned with the corresponding source statistics $(0, \lambda_k^{\text{src}})$. The adaptation objective is a weighted symmetric Kullback-Leibler divergence between assumed normal distributions:

$$\mathcal{L}_{\text{TTA}} = \frac{1}{2} \sum_{k=1}^{K} \boldsymbol{\alpha}_k \left(\frac{(\tilde{\mu}_k^{\text{tgt}})^2 + \lambda_k^{\text{src}}}{\tilde{\sigma}_k^{\text{tgt}2}} + \frac{(\tilde{\mu}_k^{\text{tgt}})^2 + \tilde{\sigma}_k^{\text{tgt}2}}{\lambda_k^{\text{src}}} - 2 \right). \tag{6}$$

F Experimental Setup

In the following paragraphs, we detail the experimental setup, including the selected models and our training and testing strategy.

F.1 Model Architectures & Pretraining

We employ different model architectures to evaluate our TTA method. The models are based on the architectures provided in the benchmark datasets Setinek et al. (2025) and Felten et al. (2025), implemented in PyTorch, and designed for conditional regression or optimization tasks. Node coordinates are provided as inputs and embedded using sinusoidal positional encodings. Conditioning is applied through a dedicated network that processes the simulation input parameters.

Conditioning Network. The conditioner maps simulation parameters into a latent representation of dimension 8. It consists of a sinusoidal encoding, followed by a small MLP, which includes two LayerNorms to stabilize training.

Transolver. The Transolver architecture (Wu et al., 2024a) starts by encoding node coordinates using sinusoidal position embeddings, followed by an MLP that produces initial feature vectors. A learned mapping then assigns each node to a slice, enabling attention operations both within slices and between them. The processed features are passed through an MLP readout to generate the final field outputs. Two conditioning mechanisms are available: concatenating the conditioning vector with input features or applying it via DiT-based modulation across the network. Conditioning is done with the dit-based modulation (Peebles & Xie, 2023). Where a latent dimension of 128, a slice base of 32, and four attention layers are used. This results in a model with 0.57M parameters. We additionally employ a larger model with 56, 128, and 8 layers for the more complex dataset, leading to 4.07M parameters.

Diffusion Model. As a diffusion model, we employ a conditional U-Net (Ronneberger et al., 2015) from Hugging Face's `diffusers` library[1].

The model works as a denoiser, taking a noisy field and a conditioning vector from the conditioning network described above and producing a noise prediction. We summarize all hyperparameters of our diffusion model in Table 7. To train the model, we use the standard Denoising Diffusion Probabilistic Models (DDPM) objective of noise prediction ("ϵ-prediction") with 100 diffusion steps and a `squaredcos_cap_v2` beta scheduler.

Pretraining setup. All unregularized baseline (*"Source"*) models are pretrained using the following setup: We use an initial learning rate of 10^3 with a cosine decay scheduler and weight decay of 10^{-5}. Training runs for up to 500, 1500, and 3000 epochs on *Beams2D*, *HeatConduction2D*, and *SIMSHIFT*, respectively, with early stopping if the validation loss does not improve for 500 epochs. We enable gradient clipping and maintain an Exponential Moving Average (EMA) of the model parameters with decay 0.95. Automatic Mixed Precision (AMP) ius enabled only for the large scale

[1] `UNet2DConditionModel`

heatsink dataset; for all others we train in `float32`. Batch size is 64 for EngiBench baselines and 16 for SIMSHIFT baselines.

Table 7: Hyperparameters for our conditional diffusion U-Net. This setup leads to a model size of 17.5M parameters.

Hyperparameter	**Value**	**HF Class Argument Name**
Block channels (low→high)	$[32, 64, 128, 256]$	`block_out_channels`
Layers per block	2	`layers_per_block`
Transformer layers / block	1	`transformer_layers_per_block`
Cross-attention dim	64	`cross_attention_dim`
Only cross-attention	`True`	`only_cross_attention=True`
Normalization groups	16	`norm_num_groups`
Activation	SiLU	`act_fn`

F.2 TTA TRAINING

Model Architecture and Representation In our specific setup, task-dependent parameters, such as thickness or temperature, are encoded through a conditioner network. The resulting conditioning output is passed to the base model, which, in our case, is a Transolver or Diffusion model. We extract features from the main body's output and define the split between the representation learner and the predictor.

The exact location of this split depends on the dataset. For SIMSHIFT, the network is split before the decoder, such that the conditioner and the Transolver body together constitute the representation learner ϕ, while the decoder acts as the predictor g. For EngiBench, the split is applied after the conditioner, meaning that the conditioner serves as the representation learner ϕ and the remaining Diffusion network functions as the predictor g.

Test-Time Adaptation and Training Procedure For all TTA experiments, validation source data are used to compute the statistical information μ_{src} and σ_{src}. In addition, a representative subset of source samples is selected from the validation set using Algorithm 1, and the corresponding set is stored for training and evaluation.

At test time, the precomputed source statistics enable the projection of the target features into the subspace. Based on the projected target features, mean and variance $(\tilde{\mu}_k^{\text{tgt}}, \tilde{\sigma}_k^{\text{tgt}\,2})$ are calculated and aligned with the corresponding source statistics $(0, \lambda_k^{\text{src}})$. We perform model updates as described in Eq. (3). For adaptation, we only utilize the target test data, and for the regularizer, the d-optimal selected samples. We balance these losses based on the number of source samples compared to the target batch size. We chose this weighting since there is a high imbalance in information between the two losses. The amount of adaptation updates is limited by the number of available batches in each target dataset. Adaptation is restricted to layer normalization (Ba et al., 2016) parameters: for EngiBench, only the layer normalization layers of the conditioner are updated, whereas for SIMSHIFT, layer normalization parameters of both the Transolver and the conditioner are adapted. All remaining parameters are kept fixed.

For parameter tuning, we compute the latent density ratio after a single forward pass through the test-time-adapted model. To estimate this ratio, the latent source and target mean and covariance are computed and stored prior to model adaptation. These statistics are the basis for estimating the density ratio between the source and target latent distributions. Since very-high dimensional settings are prone to a lot of noise in the covariance estimation Zhang et al. (2023), we decided to perform a dimension reduction to improve robustness. This enables reliable covariance estimation using the D-optimally selected source samples. For each D-optimal source sample, the density ratio is computed, and the resulting values are aggregated into a loss that is used for model selection.

As described in Section 3.2, model selection is performed after TTA based on IMV criterion. TThe search over learning rates (lr) is terminated based on performance measured on the D-optimally selected source samples. We use the Root Mean Squared Error (RMSE) for the SIMSHIFT dataset and the COMP metric for EngiBench. We set the hyperparameter search for the learning rates to $[0.05, 0.01, 0.005, 0.001, 0.0005, 0.0001]$.

We follow standard TTA practice and use batch size of 64 for all experiments. To ensure robustness, we repeat each experiment with 20 random seeds per model for the SIMSHIFT benchmark, 10 for the structural beam bending dataset, and 2 for the 2D heat conduction dataset. The varying seeds are determined by the number of data samples in each dataset. Since the 2D heat conduction dataset is small and effectively contained within a single test-time batch, increasing the number of seeds did not affect the performance of the TTA algorithm. This is particularly important since layer normalization is updated online, after every batch.

Baselines For model comparison, we evaluate existing TTA methods commonly used in both regression and classification tasks. For SSA as well as for Tent, we follow the procedures described in their respective method sections (Wang et al., 2021; Adachi et al., 2025). In the implementation of SSA, the top-K eigenvalues need to be identified to compute statistics only based on a sparse set of information. We do this for each dataset. For Tent, an additional modification is required for the SIMSHIFT dataset: since entropy minimization is applied by minimizing predictive uncertainty, we train a model that explicitly predicts both mean and variance. Additionally, we report the best-performing TTA model on SIMSHIFT that is not selected using the IWV criterion. This result serves as a lower bound, highlighting the impact of stability-aware model selection in our approach.

G Distribution Shifts for EngiBench

To establish a source and target split, we follow the approach provided by Setinek et al. (2025). We train models on the full datasets and subsequently analyze the t-SNE visualizations of the latent feature spaces as the input conditions are varied. Datasets are then partitioned into source and target domains based on the parameters that dominate the latent space variation.

Figs. 2 to 3 show t-SNE visualizations of the conditioning-networks' latent spaces for models trained across the full range condition variables. For *structural beam bending* (Fig. 2), `volfrac` and `rmin` cause the clearest structure in latent space. We therefore chose to split the source and target domain depending on the value range of `rmin`. For *2D heat conduction* (Fig. 3), `volume` and `length` exhibit comparable influence on the latent space distribution. Following the same protocol, split along `volume`. The resulting source and target ranges and sizes for both datasets can be found in Table 8.

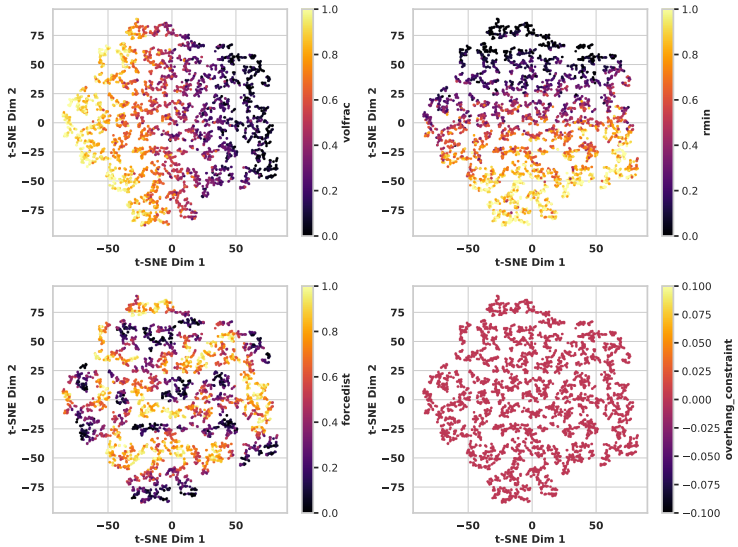

Figure 2: t-SNE visualization of the conditioner's latent space on the *structural beam bending* dataset. While `overhang_constraint` and `forcedist` are either constant or exhibit almost a uniform distribution, `volfrac` and `rmin` exhibit a clear structure.

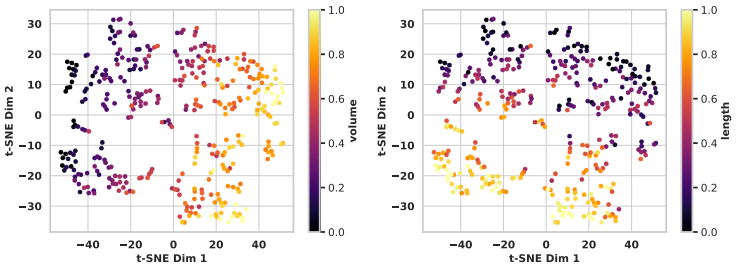

Figure 3: t-SNE visualization of the conditioner's latent space on the *structural beam bending* dataset. Both conditions (`volume` and `lentgh`) exhibit a clear structure in the latent space.

Table 8: Defined distribution shifts (source and target domains) for each dataset.

Dataset	Parameter	Description	Source range (no. samples)	Target range (no. samples)
Beams2D	rmin	Minimum feature length of beam members.	[1.5, 3.25) (3087)	[3.25, 4] (353)
HeatConduction2D	volume	Volume limits on the material distributions.	[0.3, 0.465) (231)	[0.465, 0.6] (39)