Inverse prediction of capacitor multiphysics dynamic parameters using deep generative model

Kart-Leong Lim, Rahul Dutta, and Mihai Rotaru
Institute of Microelectronics,
*Agency for Science, Technology and Research (A*STAR)*
Singapore, Singapore
{limkl, duttar, mihai_dragos_rotaru}@ime.a-star.edu.sg

***Abstract*—Finite element simulations are run by package design engineers to model design structures. The process is irreversible meaning every minute structural adjustment requires a fresh input parameter run. In this paper, the problem of modeling changing (small) design structures through varying input parameters is known as inverse prediction. We demonstrate inverse prediction on the electrostatics field of an air-filled capacitor dataset where the structural change is affected by a dynamic parameter to the boundary condition. Using recent AI such as deep generative model, we outperformed best baseline on inverse prediction both visually and in terms of quantitative measure.**

***Index Terms*—Boundary conditions, electrostatics field, inverse prediction, deep learning**

I. Introduction

Most multiphysics simulation [1], [2], [3] are centered around the forward input-output model and well studied for decades. In electrostatics, we can model the relationship between the Cartesian coordinates input and the electrostatics field output of a capacitor using Laplace's equation [4], [5], [3]. In fluid dynamics, we can model the spatial-temporal input to a heat sink viscosity output using Burger's equation [1]. However in a multi-billions industry such as in semiconductor, the development of forward models are usually well established and made available by commercial softwares. Computing multiphysics forward models such as using finite element method (FEM) software requires licensing. Also, it is impractical to run a new multiphysics design for each dynamic parameter changes to the forward model. On the other hand, performing inverse prediction/modeling [6], [7], [8], [9], [10] is an ill-posed AI/ML problem that is very challenging to solve for both the industry and researchers. In this work, we propose the use of generative models such as the variational autoencoder (VAE) [11] and autoencoder (AE) [12] to model dynamic parameter multiphysics simulation. Thus, we have a latent representation of the multiphysics simulation that models infinte dynamic parameter changes. We demonstrate in the experiment that inverse prediction in the latent space will have reduced error compared to traditional inverse prediction.

II. Background

The electrostatic field of an air-filled capacitor [13] can be described using collocation points and boundary points on a 2D grid in Fig 1. The collocation points are described by Laplace equation and the boundary points by boundary conditions . We consider five boundary conditions $C1$ to $C5$ defined using parameters (a, b, d, V_o) along a 2D axis (x, y) in Fig 1. Instead of traditionally fixing all the boundary parameters, we allow dynamic parameter $d = [0, 1]$ to vary, which changes the electrostatic field extensively as seen in Fig 2. In this work, we require a set of training data which comprises of electrostatic fields V and dynamic parameter d which can be obtained by commercial FEM softwares (e.g. ANSYS or MATLAB) or PDE solver (e.g SOR [5] or PINN [4], [3], [2]).

III. Problem statement:

A. Fullspace approach

A trained regression model map electrostatics fields $V_{1\ldots m}$ as input samples and their corresponding dynamic parameter $d\left(V_{1\ldots m}\right)$ as output samples in eqn (1). Inverse prediction on the trained regression model ϕ then predicts an unknown output V_i when presented with a given dynamic parameter input $d(V_i)$. We can formulate the objective of inverse prediction below which we refer to as the fullspace approach:

$$\begin{gathered}\arg\min_{\hat{V}_i} \left[\hat{V}_i\phi - d(V_i)\right] \\ where \;\; V_{1\ldots m}\phi = d\left(V_{1\ldots m}\right)\end{gathered} \tag{1}$$

The dimensionality of the input can reach a complexity as high as $V_i \in \mathbb{R}^{I\times I}$ e.g. $I = 401$, given the output $d(V_i) \in \mathbb{R}^1$. Thus, inverse prediction in eqn (1) is an ill-posed problem due to three main problems:

i) The root finding of $\hat{V}_i$ is difficult due to having too many coefficients to minimize.
ii) A good initial estimate is often necessary.
iii) If training size m is insufficient, it will lead to many redundant coefficients for ϕ.

IV. Proposed method

A. Latent approach

We proposed a latent approach as summarized in Fig 3 to address the aforementioned problems. The main advantages of the proposed method are:

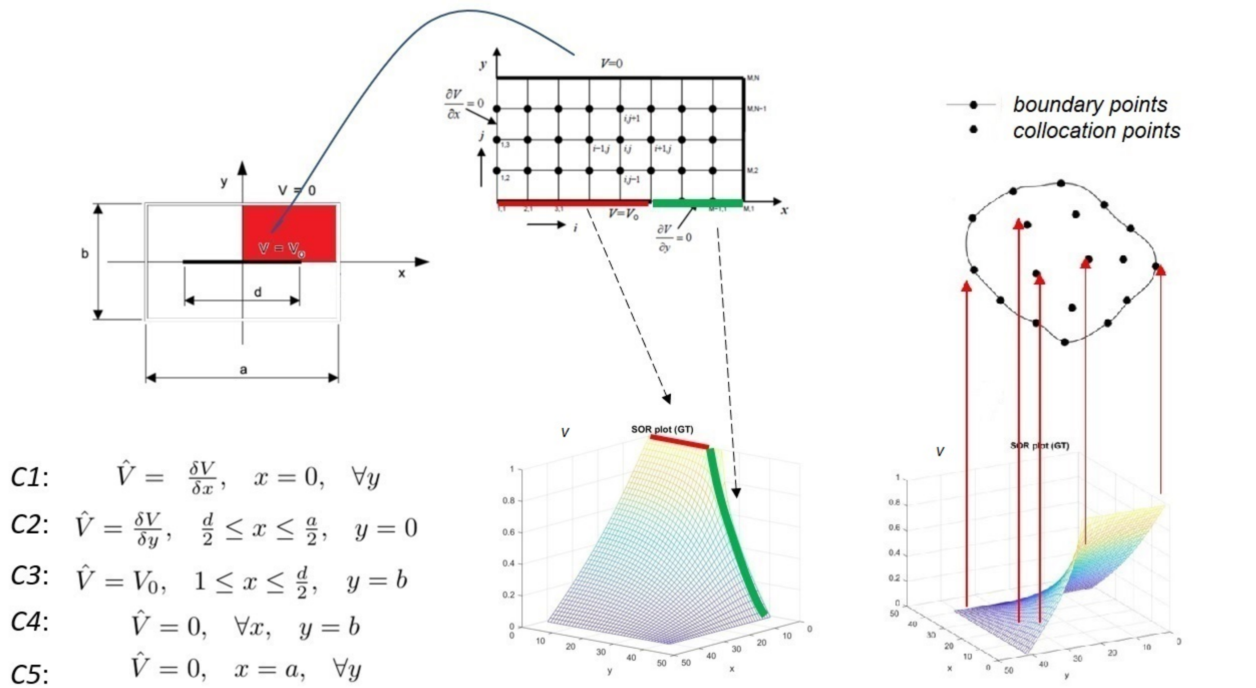

Figure 1. Modeling the boundaries of an electrostatic field (an air-filled capacitor) using five boundary conditions. In particular, d affects boundary conditions $C2$ and $C3$.

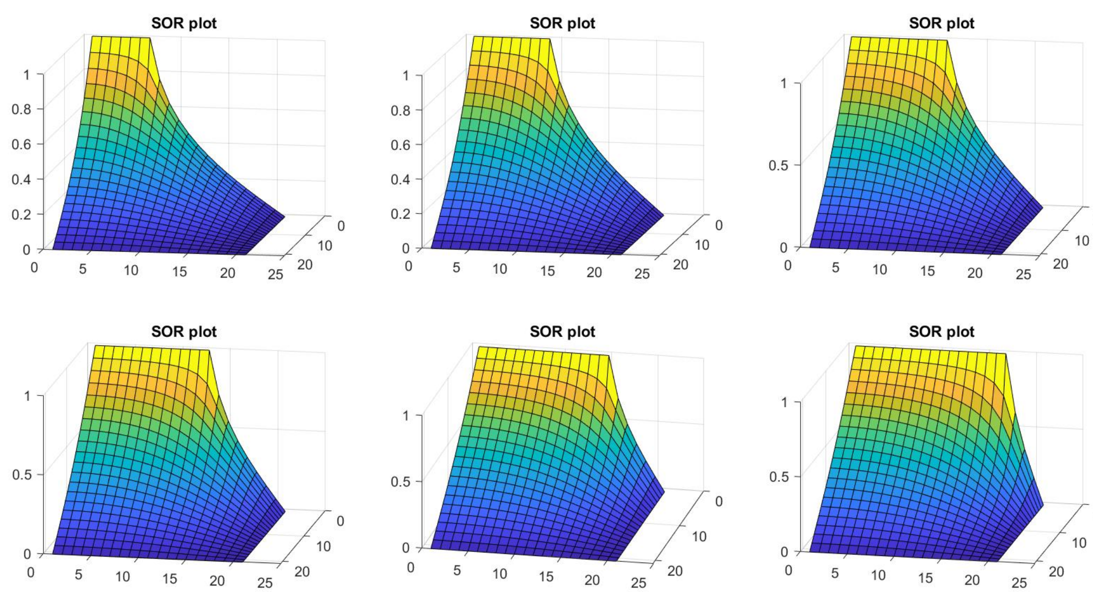

Figure 2. Groundtruth V plots of an air-filled capacitance, corresponding to dynamic parameter $d = [0.3, 0.4, ..., 0.8]$.

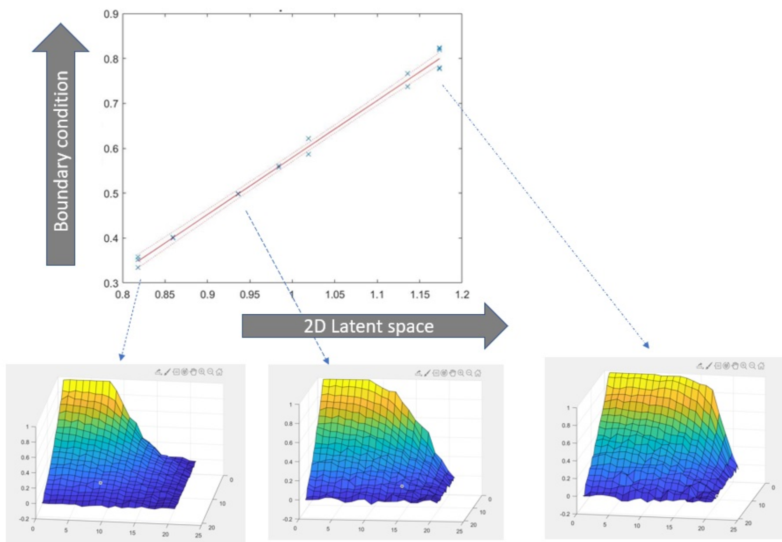

Figure 3. V plots modeled as points in the latent space. A regression model ϕ is learnt on these points with their corresponding d. Then, ϕ can be used to inverse predict an unknown latent point for a new d.

i) Lower error for modeling V: - using unsupervised deep learning as feature extraction.

ii) Faster computational time: - both regression and inverse prediction work less hard due to reduced dimensionality.

The latent approach uses eqn (2) for inverse prediction. Also, we note that the dimension of $V_{1\ldots m} \in \mathbb{R}^{I\times I}$ in the latent space denoted as $L(V_{1\ldots m}) \in \mathbb{R}^{1\times Z}$ is much smaller i.e. $Z \ll I \times I$.

$$\begin{gathered}\underset{L(\hat{V}_i)}{\arg\min} \left[L(\hat{V}_i)\phi - d(V_i)\right] \\ where \;\; L(V_{1\ldots m})\phi = d\left(V_{1\ldots m}\right)\end{gathered} \tag{2}$$

We detail our latent approach (Fig 4) as follows:

Deep generative model (training): During training phase, an unsupervised dataset of $V_{1\ldots n}$ plots alone (d is not used) is made available and a VAE [11], [14] or AE [12], [15] is trained. This is illustrated by the blue solid arrow path.

Regression model (training): A second supervised dataset is presented whereby $L(V_{1\ldots m})$ with known corresponding $d(V_{1\ldots m})$ is used to train a regression model. It is shown using the blue dashed arrow path connecting the green box ϕ.

Inverse prediction (testing): During testing, the red dashed arrow path connecting the green box $\hat{V}_i$ recovers an unknown $L(\hat{V}_i)$ for a new $d(V_i)$.

Reconstruction (testing): Lastly, the test sample $L(V_i)$ is fed to the decoder to reconstruct V_i denoted as V_i'.

B. Deep generative models

The latent space of AE denoted as $L^{AE}(V)$ is straightforward defined using the hidden layer μ as shown in Fig 5. Whereas the latent space of VAE denoted as $L^{VAE}(V)$ further consist of an additional hidden layer σ corrupted by Gaussian noise $\mathcal{N}(0,1)$.

$$\begin{gathered}L^{AE}(V) = \mu \\ L^{VAE}(V) = \mu + \sigma \cdot \mathcal{N}(0,1)\end{gathered} \tag{3}$$

While AE trains a **one-to-one** mapping of V to μ (and vice-versa), it is poor at reconstructing the targeted V when μ is not longer accurate. The novelty of VAE comes from the fact that it can train a **one-to-many** mapping of V to μ. Whereby μ is learnt together with many draws of Gaussian in eqn (3). This is a useful asset especially when the observed latent sample is noisy in real cases.

VAE achieve this **one-to-many** mapping through KLD loss ($\mathcal{L}^{KLD}$) on top of **one-to-one** mapping using reconstruction loss ($\mathcal{L}^{REC}$) in eqn (4).

$$\begin{gathered}\mathcal{L}^{REC} = -\tfrac{1}{2}\left(V - V'\right)^2 \\ \mathcal{L}^{KLD} = -\tfrac{1}{2}\left(\sigma^2 + \mu^2 - \ln\sigma^2 - 1\right)\end{gathered} \tag{4}$$

V. Experiments

We target the effectiveness of inverse prediction under noisy condition on electrostatic field simulation. We propose using VAE in comparison with strong and weak baselines. We refer to strong baseline as the latent approach based on AE and weak baseline using fullspace approach. We show that our proposed method consistently outperforms both baselines under metric benchmark. We use the parameters in Table 1 when training VAE and AE for producing the results in Fig 6-9.

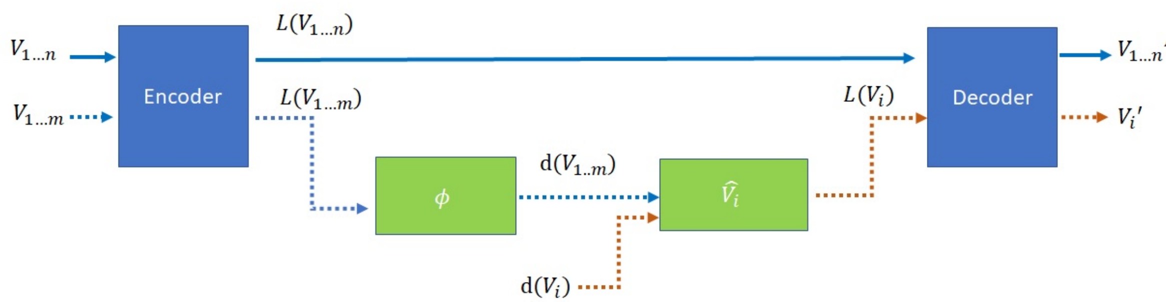

Figure 4. Proposed latent approach for inverse prediction.

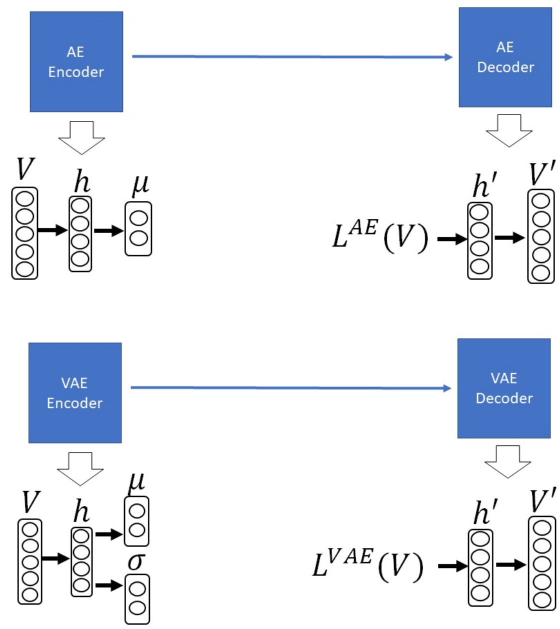

Figure 5. Architecture of deep generative models: AE vs VAE.

A. Noisy condition for initial estimate

We refer to X as the clean initial estimate for $V_{1...m}$ or $L(V_{1...m})$ both corresponding to $d = 0.5 \pm 0.2$. G refers to Gaussian noise with zero mean and variance, $e = [0.01, ...1.0]$. The noisy initial estimate Y when corrupted by additive white Gaussian noise (AWGN) is given as

$$Y = X + G$$

$$G \sim \mathcal{N}(0, e) \tag{5}$$

Fig 6 mainly shows that a latent approach (AE, VAE) is more robust than fullspace approach when using Y instead of X. When e is low, the fullspace approach shows error reproducing boundary condition $C2$. When e is large, the fullspace approach fails entirely to produce an acceptable V plot. Also, the output of AE becomes less smooth than VAE as e increases (which we shall make a detailed comparison later on).

Visually, the plots of AE and VAE under noise are not smooth in Fig 6. To improve this issue, we use Adam to train the loss functions for VAE. We compare VAE trained using Momentum [16] versus VAE trained using Adam [17] in Fig 7 and see that the latter produce significantly smoother plots very close to groundtruth.

B. Metric benchmark

In our findings (based on Momentum for both AE and VAE), as the initial estimate becomes noiser (e increases from 0.01 to 0.2), the decoder output of AE starts to become less smooth than VAE. In Fig 8, we use a quantitative measure based on the sum-of-square-difference (ssd) at each node of a 21x21 grid to capture this behavior.

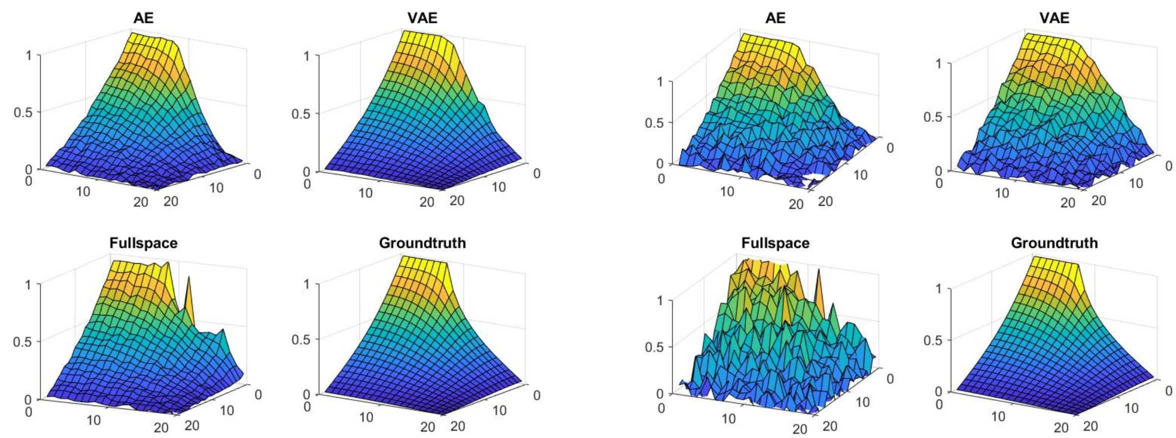

Figure 6. Reconstruction of V' plots ($d = 0.36$) when presented with AWGN ($e = 0.01$ and $e = 0.1$) in eqn (5).

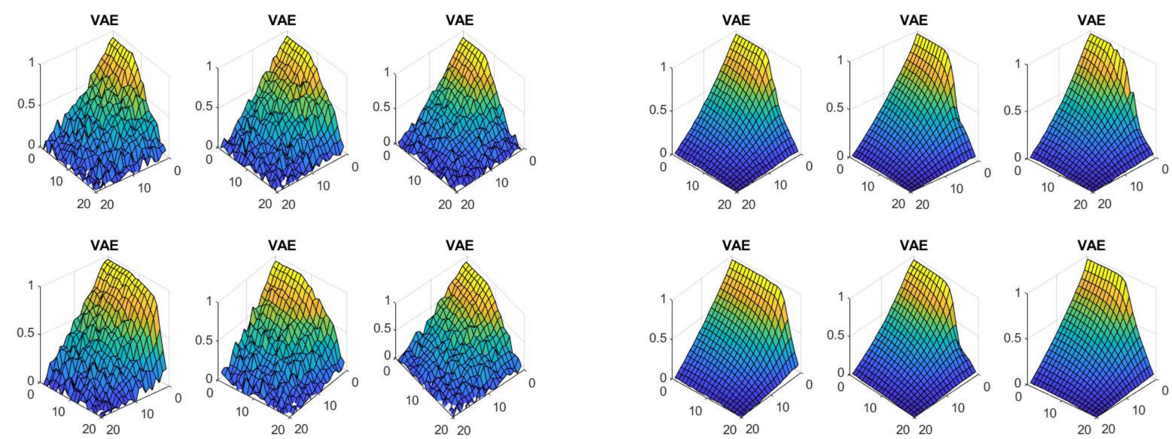

Figure 7. Plots of V for different d with $e = 0.1$. (Left plots) Momentum. (Right plots) Adam.

$$ssd = \sum_{i=1}^{T}(V_i - \hat{V}_i)^2 \tag{6}$$

Under low noise level of $e = 0.01$, AE has much lower error than VAE. However, as we increase the noise level from 0.1 towards 1.0, VAE consistently produces less error than AE across all d.

Although not a direct comparison with AE, we see that when comparing Fig 9 and Fig 8, VAE trained using Adam produce much lower error than using Momentum for ssd across all noise level.

C. CPU Time

We show in Table 2 the execution time taken for each operation in the latent and fullspace approach. In particular, performing regression and inverse prediction each took approximately 10x to 20x longer for the fullspace approach. When we use the entire training and test data, the fullspace approach takes 10x slower to compute than either AE or VAE.

Table I
PARAMETERS FOR DEEP GENERATIVE MODEL

	Parameters
Architecture	441-200-20-200-441
Activation function	tanh
Max iterations	20,000
Gradient learner	Momentum, Adam
Learning rate	$1e^{-3}$, $1e^{-5}$
Minibatch size	20, 100
Train samples	120
Test samples	6

This is due to both latent approach using a much smaller dimension at 20 vs 441 when performing regression and inverse prediction.

VI. Conclusion

We propose an inverse prediction on capacitor multiphysics in the latent space. It allows the user to use a dynamic parameter to reconstruct the exact electrostatic field distribution

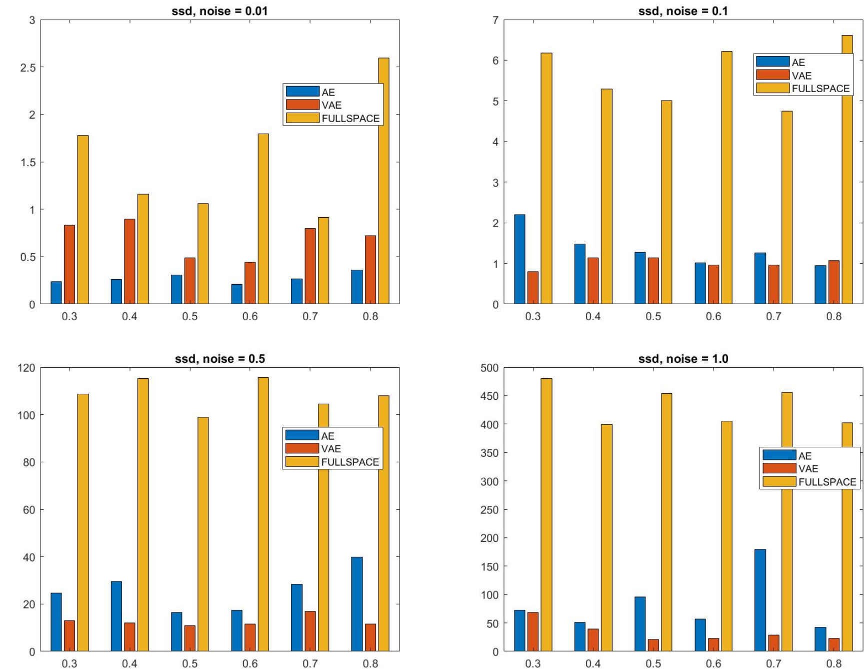

Figure 8. Measuring sum-of-square-difference error at the decoder.

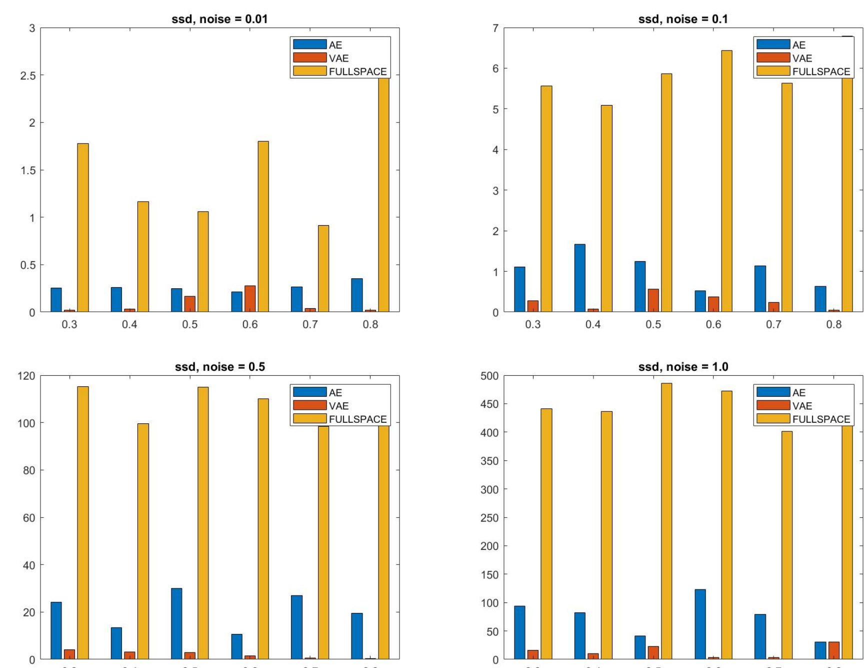

Figure 9. Measuring sum-of-square-difference error at the decoder (VAE trained with Adam).

Table II
COMPUTATIONAL TIME

CPU time (ms)	Fullspace	AE	VAE
encoder	-	1	1
regression	1	0.2	0.5
inverse	180	4	5
decoder	-	2	2
total	1142	148	163

that accurately corresponds to the boundary condition. Our proposed method is mainly based on a deep generative model such as VAE. We showed that using sophisticated gradient learner such as Adam our method can reconstruct field plots very close to groundtruth.

REFERENCES

[1] O. Hennigh, S. Narasimhan, M. A. Nabian, A. Subramaniam, K. Tangsali, Z. Fang, M. Rietmann, W. Byeon, and S. Choudhry, “Nvidia simnet: An ai accelerated multi-physics simulation framework,” in *International Conference on Computational Science*. Springer, 2021, pp. 447–461. I

[2] M. Raissi, P. Perdikaris, and G. E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019. I, II

[3] K. L. Lim, R. Dutta, and M. Rotaru, “Physics informed neural network using finite difference method,” in *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2022, pp. 1828–1833. I, II

[4] A. Khan and D. A. Lowther, “Physics informed neural networks for electromagnetic analysis,” *IEEE Transactions on Magnetics*, pp. 1–1, 2022. I, II

[5] J. R. Nagel *et al.*, “Solving the generalized poisson equation using the finite-difference method (fdm),” *Lecture Notes, Dept. of Electrical and Computer Engineering, University of Utah*, 2011. I, II

[6] R. Dutta, A. James, S. Raju, Y.-J. Jeon, C. S. Foo, and K. Tshun Chuan Chai, “Automated deep learning platform for accelerated analog circuit design,” in *2022 IEEE 35th International System-on-Chip Conference (SOCC)*, 2022, pp. 1–5. I

[7] J.-W. Lee, W. B. Park, B. Do Lee, S. Kim, N. H. Goo, and K.-S. Sohn, “Dirty engineering data-driven inverse prediction machine learning model,” *Scientific Reports*, vol. 10, no. 1, p. 20443, 2020. I

[8] M. Lebreux, M. Desilets, and M. Lacroix, “Fast inverse prediction of phase change banks in high temperature furnaces with a kalman filter coupled with a recursive least-square estimator,” *International Journal of Heat and Mass Transfer*, vol. 53, pp. 5250–5260, 2010. I

[9] V. Schäfer, “Generalization of physics-informed neural networks for various boundary and initial conditions,” Ph.D. dissertation, Technische Universität Kaiserslautern, 2022. I

[10] J. Kim, K. Lee, D. Lee, S. Y. Jhin, and N. Park, “Dpm: a novel training method for physics-informed neural networks in extrapolation,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 9, 2021, pp. 8146–8154. I

[11] D. P. Kingma and M. Welling, “Stochastic gradient vb and the variational auto-encoder,” in *Second International Conference on Learning Representations, ICLR*, 2014. I, IV-A

[12] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *science*, vol. 313, no. 5786, pp. 504–507, 2006. I, IV-A

[13] M. N. Sadiku and S. R. Nelatury, *Analytical Techniques in Electromagnetics*. CRC press, 2015. II

[14] K.-L. Lim, X. Jiang, and C. Yi, “Deep clustering with variational autoencoder,” *IEEE Signal Processing Letters*, vol. 27, pp. 231–235, 2020. IV-A

[15] C. Song, F. Liu, Y. Huang, L. Wang, and T. Tan, “Auto-encoder based data clustering,” in *Iberoamerican Congress on Pattern Recognition*. Springer, 2013, pp. 117–124. IV-A

[16] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, “On the importance of initialization and momentum in deep learning,” in *Proceedings of the 30th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, S. Dasgupta and D. McAllester, Eds., vol. 28, no. 3. Atlanta, Georgia, USA: PMLR, 17–19 Jun 2013, pp. 1139–1147. [Online]. Available: https://proceedings.mlr.press/v28/sutskever13.html V-A

[17] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014. V-A