Learning Dynamics of Nonlinear Field-Circuit Coupled Problems with a Physics-Data Combined Model

Shiqi Wu[a], **Gérard Meunier**[b], **Olivier Chadebec**[b], **Qianxiao Li**[a], **Ludovic Chamoin**[c]

[a] *Department of Mathematics, National University of Singapore, Sinagpore; CNRS@CREATE LTD, Singapore*

[b] *Université Grenoble Alpes, CNRS, Grenoble INP, G2Elab, France*

[b] *Université Paris-Saclay, CNRS, France*

* Presenting author

1. Introduction

Field-circuit coupled problems are fundamental in electrical engineering applications (e.g., electrical machines, actuators, sensors, power electronics), where Maxwell's equations at the component scale interact with circuit equations at the system scale [1]. This coupling results in nonlinear, high-dimensional Differential-Algebraic Equations (DAEs) [2], posing computational challenges. We propose a hybrid model that combines a linear state-space formulation with a Koopman-based machine-learning component for efficient field-circuit dynamics prediction. Using an extended non-intrusive algorithm, our approach achieves 1% prediction error, outperforms traditional models, and provides a 3-order-of-magnitude speedup over time-stepping volume integral methods, making it ideal for real-time and repeated simulations.

2. Time-Stepping Volume Integral Method For Nonlinear Field-ciruit Coupled Problems

Field-circuit coupling arises in applications such as current sensors, where Maxwell's magnetostatic equations in a nonlinear magnetic region must be solved alongside external circuit equations for coils [1, 3]. A time-stepping volume integral approach discretizes the magnetic field via volume elements, incorporates circuit currents, and enforces continuity through a mesh-current analysis. This yields a high-dimensional, ODE-algebraic coupled system:

$$\begin{aligned} [\mathbb{R}_m + \mathbb{P}_m]\mathbf{\Phi} + \mathbb{L}_{\text{mb}}\mathbf{I}_b &= \Delta\overline{\varphi}_r, \\ \frac{d}{dt}\big(\mathbb{L}_{bm}\mathbf{\Phi}\big) + \mathbb{R}_b\,\mathbf{I}_b + \frac{d}{dt}\big(\mathbb{L}_b\,\mathbf{I}_b\big) &= \Delta\mathbf{V}_b. \end{aligned} \tag{1}$$

Here, $\mathbf{\Phi} \in \mathbb{R}^{N_f}$ represents the flux unknowns across N_f finite-element faces, while $\mathbf{I}_b \in \mathbb{R}^{N_b}$ is the coil current vector. The terms $\Delta\overline{\varphi}_r \in \mathbb{R}^{N_f}$ and $\Delta\mathbf{V}_b \in \mathbb{R}^{N_b}$ denote magnetic potential differences and coil voltages, respectively. Matrices $\mathbb{R}_m$ and $\mathbb{P}_m$ arise from discretized reluctivity and boundary integral contributions, both being nonlinear functions of $\mathbf{\Phi}$. Coupling matrices $\mathbb{L}_{\text{mb}} \in \mathbb{R}^{N_f \times N_b}$ and $\mathbb{L}_{\text{bm}} \in \mathbb{R}^{N_b \times N_f}$ link the field and circuit variables, while $\mathbb{R}_b$ and $\mathbb{L}_b$ are diagonal resistance and mutual inductance matrices of size $N_b \times N_b$.

In practice, the coupled system is solved by Kirchhoff's mesh rule ensuring the free divergence of $\mathbf{B}$ in the magnetic region mesh and the current conservation in each coil [3]. Exploiting the electric circuit interpretation of the global problem, a mesh current analysis is made on the mesh of the magnetic region and in the circuit coils. This analysis makes it possible to express an incidence matrix that links $\mathbf{\Phi}$ to induction flux on closed loops $\mathbf{\Phi}_{loop}$ in the mesh. Consequently, the coupled system can be represented as a general dynamical system with $\mathbf{x}$ as state variable :

$$\mathbb{R}(\mathbf{x})\mathbf{x} + \frac{d}{dt}\mathbb{L}(\mathbf{x})\mathbf{x} = \mathbf{U}_{sc}(\mathbf{I}_p), \tag{2}$$

where $\mathbf{x} = [\mathbf{\Phi}_{loop}; \mathbf{I}_s]$, i.e. the concatenation of magnetic flux loops flowing in the mesh of the magnetic region and the current flowing in the secondary coil $\mathbf{I}_s$, and $\mathbf{U}_{sc}(\mathbf{I}_p)$ is a vector term which depends on the problem input $\mathbf{I}_p$. The loop resistance $\mathbb{R}(\mathbf{x})$ and the loop inductance $\mathbb{L}(\mathbf{x})$ are nonlinear functions of the variable $\mathbf{x}$. $\mathbb{R}(\mathbf{x})$ and $\mathbb{L}(\mathbf{x})$ are linked to matrix $\mathbb{R}_m$, $\mathbb{P}_m$, $\mathbb{L}_{\text{mb}}$ and $\mathbb{L}_{\text{b}}$, $\mathbb{L}_{\text{mb}}$ respectively by a change of basis involving the incidence matrix coming from the mesh current analysis. This is a high-dimensional input-output system with state $\mathbf{x}$ and time-dependent inputs I_p.

3. Framework and Training Strategy for the Physics-Data Combined Model

3.1 Physical-Data Combined Model

Physics-integrated machine learning models are built upon an analysis of the physical system. A simple or incomplete physical model can describe the general behavior of the system, while a data-driven model learns the corrections to the physical model [4, 5, 6].

To construct such combined models, the key observation is as follows: the nonlinearity of the target system (2) originates from the magnetic reluctivity ν. The magnetic field $\mathbf{H}$ is a nonlinear function of the induction $\mathbf{B}$, which, in turn, depends on the magnetic flux flowing through the mesh. If the reluctivity ν is constant, the system (2) can be expressed as a linear system:

$$\mathbb{R}\mathbf{x} + \frac{d}{dt}\mathbb{L}\mathbf{x} = \mathbf{U}_{sc}(\mathbf{I}_p). \tag{3}$$

We represent this linearized system in state-space

form:

$$\mathbf{x}(t_{k+1}) = \mathbb{C}\mathbf{x}(t_k) + \mathbb{D}\mathbf{u}(t_k), \tag{4}$$

where $\mathbf{u}(t_k) = [I_p(t_k), I_p(t_{k+1})]$ represents the input. This model captures key system behaviors but lacks the flexibility to account for nonlinear effects.

To enhance accuracy, we introduce a data-driven correction:

$$\mathbf{x}(t_{k+1}) = \mathbb{C}\mathbf{x}(t_k) + \mathbb{D}\mathbf{u}(t_k) + f(\mathbf{x}(t_k), \mathbf{u}(t_k)). \tag{5}$$

Using a Koopman-type model, f learns observables that evolve linearly:

$$f(\mathbf{x}(t_k), \mathbf{u}(t_k)) = \mathbb{E}(\mathbb{K}\mathbf{\Psi}(\mathbf{x}(t_k)) + \mathbb{L}\mathbf{u}(t_k)), \tag{6}$$

where $\mathbb{E}$ extracts field data from observables $\mathbf{\Psi}$, and $\mathbb{K}, \mathbb{L}, \mathbf{\Psi}$ are learned via data-driven optimization of Neural Network [7].

3.2 Training Strategy with a Non-Intrusive Algorithm

The training strategy for the combined model extends from our previous work [8], maintaining non-intrusiveness by training the two models separately. The state-space model is optimized using a closed-form least squares solution, while the Koopman-type machine learning model is trained with the Adam optimizer.

To ensure convexity and enable a closed-form solution, the state-space model parameters are selected via a one-step prediction optimization problem. Meanwhile, the Koopman-type model is trained with a multi-step prediction loss function to enhance long-term forecasting accuracy.

Given a dataset $\mathcal{D} = \{(\mathbf{x}, \mathbf{u})\}$ containing sequences of states and inputs, the multi-step prediction loss function is defined as:

$$\begin{aligned} \mathcal{L}(\mathbf{x}, \mathbf{u}; f) &= \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x},\mathbf{u})\in\mathcal{D}} \sum_{k=0}^{L-1} \|\mathbf{x}(t_k) - \hat{\mathbf{x}}(t_k)\|^2, \\ \hat{\mathbf{x}}(t_{k+1}) &= \mathbb{C}\hat{\mathbf{x}}(t_k) + \mathbb{D}\mathbf{u}_k + f(\hat{\mathbf{x}}(t_k), \mathbf{u}(t_k)), \\ \hat{\mathbf{x}}(t_0) &= \mathbf{x}(t_0), \end{aligned} \tag{7}$$

where L is the prediction horizon, set to 10 in our experiments. The full training algorithm is detailed in Algorithm 1.

Algorithm 1 The iterative model combination algorithm with multi-step loss function

Require: Dataset $\mathcal{D}$
$j \leftarrow 0$
solve the state-space model by least square: $\mathbb{C}_0, \mathbb{D}_0 \leftarrow \mathrm{argmin}_{\mathbb{C},\mathbb{D}} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}(t_k),\mathbf{x}(t_{k+1}),\mathbf{u}(t_k))\in\mathcal{D}} \|\mathbf{x}(t_{k+1}) - (\mathbb{C}\mathbf{x}(t_k) + \mathbb{D}\mathbf{u}(t_k))\|^2$
while stopping criterion is not satisfied **do**
  $j \leftarrow j + 1$
  solve the data-driven model by gradient-based optimization: $f_j \leftarrow \mathrm{argmin}_f \mathcal{L}(\mathbf{x}, \mathbf{u}; f)$
  $e \leftarrow \mathbf{x}(t_{k+1}) - f_j(\mathbf{x}(t_k), \mathbf{u}(t_k))$
  solve the state-space model by least square: $\mathbb{C}_j, \mathbb{D}_j \leftarrow \mathrm{argmin}_{\mathbb{C},\mathbb{D}} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}(t_k),\mathbf{x}(t_{k+1}),\mathbf{u}(t_k))\in\mathcal{D}} \|e - (\mathbb{C}\mathbf{x}(t_k) + \mathbb{D}\mathbf{u}(t_k))\|^2$
end while
return parameters in the combined model $\mathbb{C}_j, \mathbb{D}_j, f_j$

4. Experimental Results

4.1 Data generation and preprocessing

The dataset is generated by simulating a nonlinear field-circuit coupled problem in a ferromagnetic region with 6956 magnetic elements and 1 electrical element, forming a 6957-dimensional state vector $\mathbf{x}$. To reduce dimensionality, the Proper Orthogonal Decomposition (POD) method is applied, compressing $\mathbf{x}$ to four principal components, denoted as $\mathbf{x}^{pca} = \mathbb{P}\mathbf{x}$. The input current follows $I_p(t_k) = 200\sin(\frac{2\pi f_1}{50}t) + 200\cos(\frac{2\pi f_2}{50}t)$, where f_1, f_2 are sampled from $[40, 60]$ Hz. The dataset consists of 60 trajectories, each with 150 time steps at a 0.0004s interval, and is split into training and testing sets with a 4:1 ratio. For training, sequences of length 10 are extracted to construct the dataset $\{(\mathbf{x}^{pca}, \mathbf{u})\}$, where $\mathbf{u}(t_k) = [I_p(t_k), I_p(t_{k+1})]$. The dictionary $\mathbf{\Psi}$ of Koopman-type model is build by $\mathbf{\Psi} = [1, \mathbf{x}^{pca}, \mathbf{\Psi}_{NN}(\mathbf{x}^{pca})$, where $\mathbf{\Psi}_{NN}$ is a 6-layer ResNet [9].

4.2 Prediction Results

Our combined model demonstrates high accuracy in predicting the electric mesh currents $\mathbf{x} = [\mathbf{\Phi}_{loop}, \mathbf{I}_s]$, achieving a relative error of around 1% with a stable relative difference over prediction time:

$$\text{Relative error} = \frac{1}{|\mathcal{D}_{eval}|} \sum_{\mathbf{z}^{true}\in\mathcal{D}_{eval}} \frac{\|\mathbf{z}^{predict} - \mathbf{z}^{true}\|}{\|\mathbf{z}^{true}\|}, \tag{8}$$

It significantly outperforms baseline models, compared to 80% for the linear state-space model and 3-4% for the Koopman-type model.

4.3 Computational efficiency

When solving DAE systems, the time-stepping volume integral method is computationally expensive, requiring approximately 28 minutes to generate 150 prediction steps, whereas the combined model completes the same task in about 2 seconds with the same mesh. Although the combined model requires a one-time offline training cost of 14 hours on an NVIDIA RTX 3090 GPU, its online prediction is three orders of magnitude faster, with only a 1% error trade-off. This efficiency makes it highly suitable for applications requiring repeated or real-time predictions, especially as the problem size or prediction horizon increases.

Acknowledgments

This research is part of the program DesCartes and is supported by the National Research Foundation, Prime Minister's Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) program.

References

[1] Mayra Hernandez Alayeto. *Volume integral method for the electromagnetic modeling of self-powered current sensors.* phdthesis, Université Grenoble Alpes [2020-....], October 2023.

[2] Idoia Cortes Garcia, Herbert De Gersem, and Sebastian Schöps. A structural analysis of field/circuit coupled problems based on a generalised circuit element. *Numerical Algorithms,* 83(1):373–394, January 2020.

[3] Mayra Hernandez Alayeto, Gérard Meunier, Loic Rondot, Olivier Chadebec, Jean-Michel Guichon, and Matthieu Favre. A Time-Stepping Volume Integral Formulation for Nonlinear Field-Circuit Coupled Problems. *IEEE Transactions on Magnetics,* 60(3):1–4, March 2024. Conference Name: IEEE Transactions on Magnetics.

[4] René Felix Reinhart, Zeeshan Shareef, and Jochen Jakob Steil. Hybrid Analytical and Data-Driven Modeling for Feed-Forward Robot Control †. *Sensors,* 17(2):311, February 2017. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

[5] Naoya Takeishi, Yoshinobu Kawahara, and Takehisa Yairi. Learning Koopman Invariant Subspaces for Dynamic Mode Decomposition. In *Advances in Neural Information Processing Systems,* volume 30. Curran Associates, Inc., 2017.

[6] Naoya Takeishi and Alexandros Kalousis. Physics-Integrated Variational Autoencoders for Robust and Interpretable Generative Modeling. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems,* volume 34, pages 14809–14821. Curran Associates, Inc., 2021.

[7] Qianxiao Li, Felix Dietrich, Erik M. Bollt, and Ioannis G. Kevrekidis. Extended dynamic mode decomposition with dictionary learning: A data-driven adaptive spectral decomposition of the Koopman operator. *Chaos: An Interdisciplinary Journal of Nonlinear Science,* 27(10):103111, October 2017.

[8] Shiqi Wu, Ludovic Chamoin, and Qianxiao Li. Non-intrusive model combination for learning dynamical systems. *Physica D: Nonlinear Phenomena,* 463:134152, July 2024.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition,* pages 770–778, 2016.

Appendix A. Detailed Results in Long-time Prediction

We evaluate the prediction performance using two metrics: relative error, which measures overall accuracy across the entire prediction horizon, and relative difference at time t_k, which assesses robustness throughout the prediction process. Given an evaluation dataset $\mathcal{D}_{eval}$ for $\mathbf{z}$, the relative error is defined as:

$$\text{Relative error} = \frac{1}{|\mathcal{D}_{eval}|} \sum_{\mathbf{z}^{true} \in \mathcal{D}_{eval}} \frac{\|\mathbf{z}^{predict} - \mathbf{z}^{true}\|}{\|\mathbf{z}^{true}\|}, \tag{A1}$$

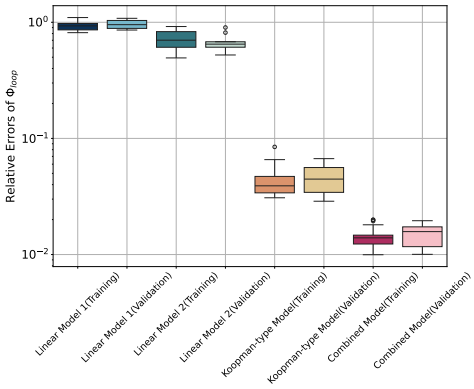
representing the mean relative error over a 150-step prediction.

To evaluate robustness at each time step, we define the relative difference as:

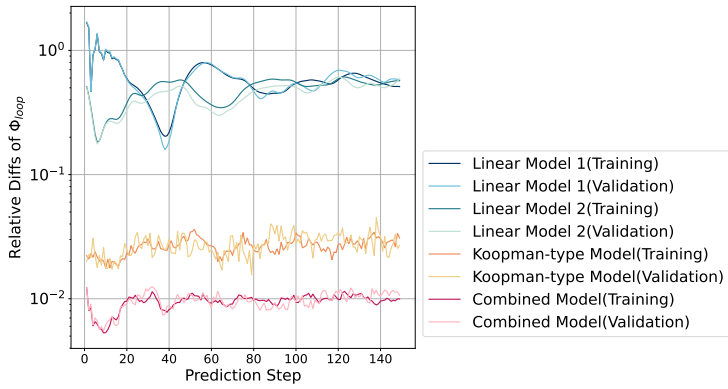
$$\text{Relative difference at time } t_k = \frac{1}{|\mathcal{D}_{eval}|} \sum_{\mathbf{z}^{true} \in \mathcal{D}_{eval}} \frac{\|\mathbf{z}^{predict}(t_k) - \mathbf{z}^{true}(t_k)\|}{\|\max_i \mathbf{z}^{true}(t_i)\|}. \tag{A2}$$

Our model demonstrates strong predictive performance for the electric mesh currents $\mathbf{x} = [\mathbf{\Phi}_{loop}, \mathbf{I}_s]$. Figure A1 compares the predicted and ground truth values of $\mathbf{\Phi}_{loop}$ across training and validation datasets. The combined model achieves a relative error of around 1% and a relative difference below 1%, consistently capturing the dynamics of the field-circuit coupled problem.

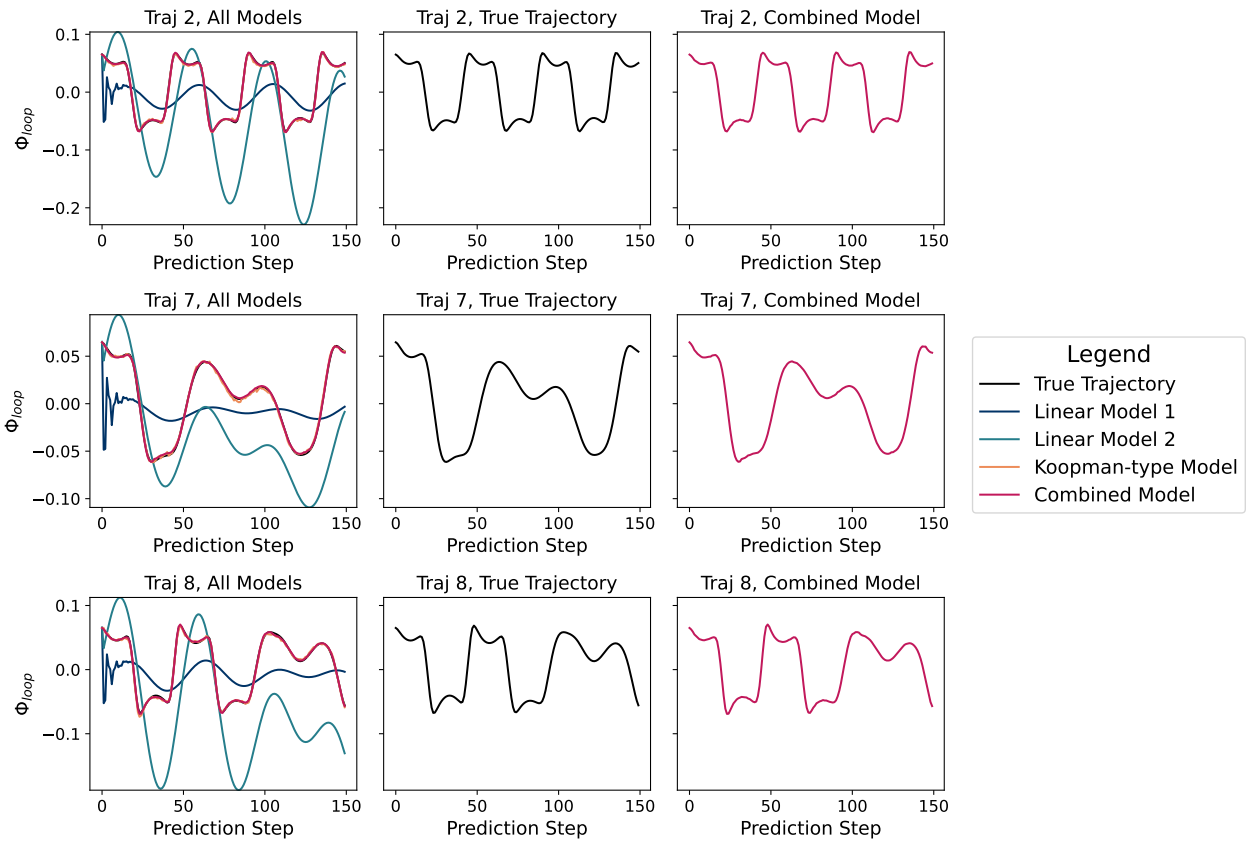
For the secondary coil current $\mathbf{I}_s$, figure A2 shows that the combined model significantly outperforms baseline models. The linear state-space model exhibits high errors (80%), while the Koopman-type model achieves 3-4% error. In contrast, our combined model maintains a relative error around 1% and a relative difference below 1%, demonstrating both accuracy and robustness.



(a) Relative error for training and validation datasets.

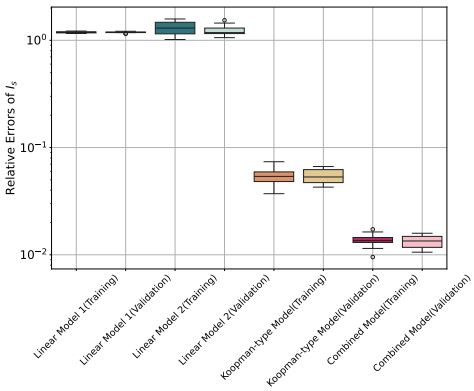


(b) Relative difference over prediction time.

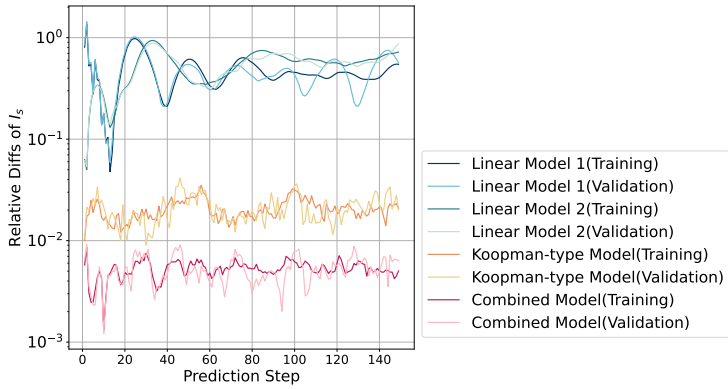


(c) Comparison of predictions for the 2829-th dimension from validation trajectories.

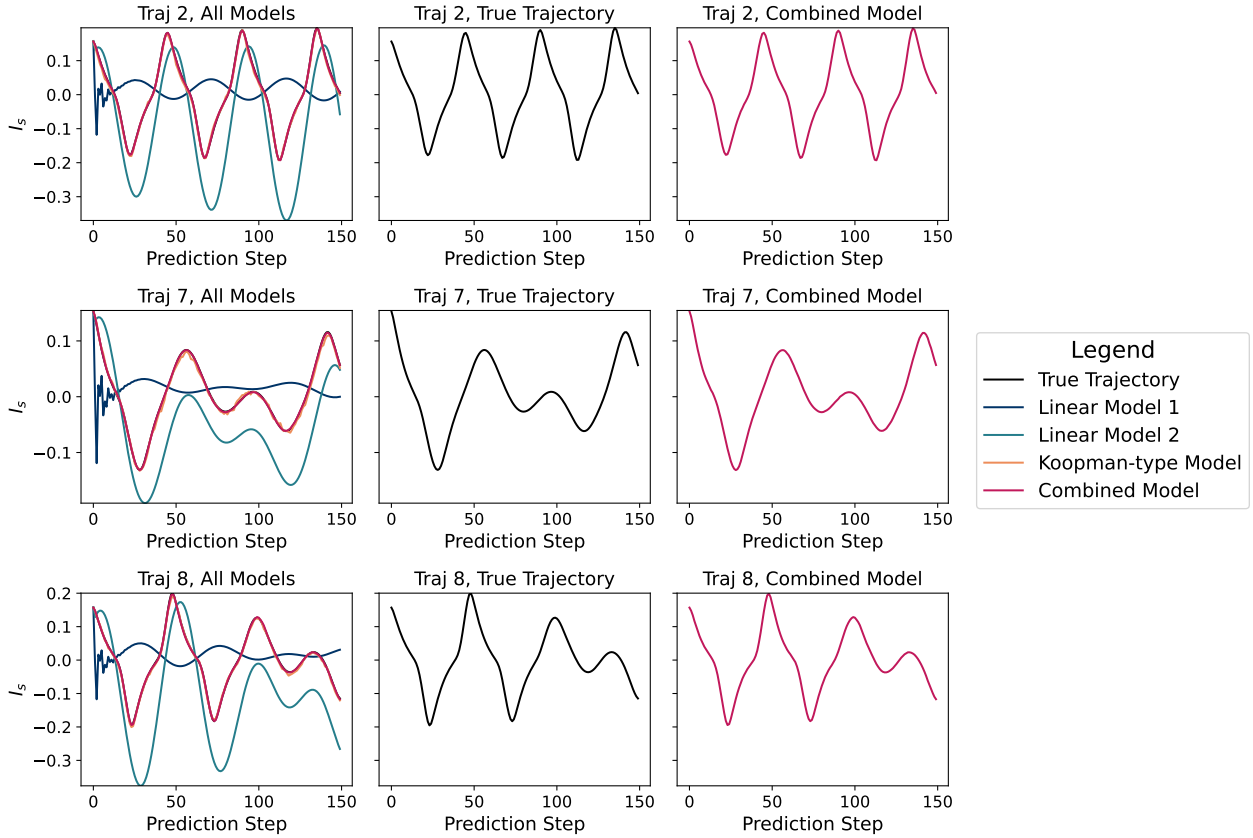
Fig. A1: Prediction of Φ_{loop} of the linear state-space models, the Koopman-type model, and the combined model. The upper left panel shows the relative error for training and validation datasets. The upper right panel displays the relative difference over prediction time. The lower panel compares predictions of the 2829-th dimension from the combined and baseline models for 3 validation trajectories.



(a) Relative error for training and validation datasets.



(b) Relative difference over prediction time.



(c) Comparison of predictions from validation trajectories.

Fig. A2: Prediction of the secondary coil current $\mathbf{I}_s$ of the linear state-space models, the Koopman-type model, and the combined model. The upper left panel shows the relative error for training and validation datasets. The upper right panel displays the relative difference over prediction time. The lower panel compares predictions from the combined and baseline models for 3 validation trajectories.