


Physically Invertible System Identification for Monitoring System Edges with Unobservability

Jingyi Yuan¹[0000–0002–2850–1582] and Yang Weng ¹[0000–0002–5267–1303]

Arizona State University, Tempe AZ 85281, USA
{jyuan46, Yang.Weng}@asu.edu

Abstract. Nowadays, the data collected in physical/engineering systems allows various machine learning methods to conduct system monitoring and control, when the physical knowledge on the system edge is limited and challenging to recover completely. Solving such problems typically requires identifying forward system mapping rules, from system states to the output measurements. However, the forward system identification based on digital twin can hardly provide complete monitoring functions, such as state estimation, e.g., to infer the states from measurements. While one can directly learn the inverse mapping rule, it is more desirable to re-utilize the forward digital twin since it is relatively easy to embed physical law there to regularize the inverse process and avoid overfitting. For this purpose, this paper proposes an invertible learning structure based on designing parallel paths in structural neural networks with basis functionals and embedding virtual storage variables for information preservation. For such a two-way digital twin modeling, there is an additional challenge of multiple solutions for system inverse, which contradict the reality of one feasible solution for the current system. To avoid ambiguous inverse, the proposed model maximizes the physical likelihood to contract the original solution space, leading to the unique system operation status of interest. We validate the proposed method on various physical system monitoring tasks and scenarios, such as inverse kinematics problems, power system state estimation, etc. Furthermore, by building a perfect match of a forward-inverse pair, the proposed method obtains accurate and computation-efficient inverse predictions, given observations. Finally, the forward physical interpretation and small prediction errors guarantee the explainability of the invertible structure, compared to standard learning methods.

Keywords: Inverse system identification · Invertible neural network · System edge · System unobservability.

1 Introduction

Monitoring is essential for the sustainable operation of physical systems. However, physical knowledge may be partially unknown, and sensor measurements are limited for system identification on the system edges [22,11,19,10]. Such weak knowledge on the edge challenges traditional monitoring approaches based on

accurate physical models. To bridge the gap, there are works on machine learning models using collected data for system identification [1,4,36]. However, although the data-driven method can mimic the behavior of a physical system, they are not indeed a digital twin to be used for system operation at any operating point [5], e.g., at new operating points never happened in the past. The problems have two causes. One is the lack of physical interpretation, and the other is the mismatch between forward and inverse mapping. These two are natural properties when the physical governing function is available. Therefore, it is essential to build the digital twin with both logical check (consistency of two-way mappings) and physics for an actual replica of the physical counterpart. This paper looks into the inverse learning for state estimation that is consistent with the forward mapping and has physics embedded.

Specifically, an intuitive way for inverse learning is to directly learn the inverse mapping rule from collected data in a discriminative manner. However, it easily causes poor performance due to overfitting. Even worse, the inverse mapping is usually more complex than the forward. For example, unlike the physical priors of the forward system model, the inverse model usually does not have a pre-defined physical form as a reference. Therefore, it is hard to maintain high accuracy directly using fitting models like deep neural networks (DNNs), especially in the extrapolation scenario.

Therefore, this paper aims to learn an accurate forward system with physical regularization while enforcing invertibility. As the prior physical knowledge is embedded into the forward mapping, the physics will regularize the inverse process automatically against overfitting in the second. Such an idea has some similarities to the (variational) auto-encoder [15,25,13,33]. However, the forward-inverse pair in the auto-encoder is forced by the reconstruction loss instead of the interoperability. So, the auto-encoder has neither a decoder providing a perfect inverse nor a physical interpretability. Thus, we would like to build a forward mapping with physics and inverse the forward DNN if possible.

For invertible transformation, we propose splitting the input variables into two groups with a swap of DNN links in the forward mapping to invert the forward DNN for system states. Such a method is much better than auto-encoder, as it can create a perfect pair of encoder and decoder without the approximation errors in typical auto-encoder [7,8]. Now that we know the principles of designing invertible DNN, we want to systematically embed physicals with three considerations. First, we aim to embed physical functionals to reveal similar forms as the physical laws. Second, we aim to embed the physical size of input/output variables into the functionals. Third, we aim to have a unique solution, since the current system state is unique no matter how many possible algebraic solutions there can be according to the mathematical function.

To achieve the first goal, we split the input into a twin set so that we not only provide all possible candidates as input to the physical DNN to maximize the physical gains but also preserve the structural requirement of having separated inputs for the invertible DNN. For the second goal, we propose to add storage variables into the output of the forward mapping rule. This step is to ensure no

physical information loss, e.g., when the output of the forward mapping is with a smaller dimension than the input. But, how to pick up the correct output size in the forward mapping? The answer is the network size. The minimum number of states is the network size according to the definition of state estimation. For the third goal, we will utilize the Bayesian framework and the maximum likelihood estimation, for which we use the historical indicator to select the best outcome and avoid the confusion of multiple solutions [18,14]. For example, Fig. 1 shows the collected data in a power system case. The curves of power generation, consumption loads, and node voltages indicate the standard data pattern during system operations. Different quantities stay within the standard operation limit of physical systems. We incorporate such a pattern in the inverse learning problems to ensure a feasible solution and physical uniqueness.

The proposed model can be implemented on various physical/engineering systems for monitoring with unobservability, including manipulator inverse kinematics, structural health monitoring of high-rise buildings, position estimation of robotic system, state estimation of power and water system, etc. [18,14,29,35,32,11,27]. For example, photovoltaic (PV) and electric vehicle (EV) penetrations change the power distribution system dramatically, where a fast-monitoring tool like state estimation (SE) is necessary for operation. Nevertheless, it is hard to conduct traditional SE due to unavailable power system modeling, and partial observability [20,6]. Thus, we conduct experiments to demonstrate how the designed invertibility efficiently infers hidden system states of interest and how the embedded physics in the forward system identification leads to consistently better performance compared to the state-of-art learning methods.

Our main contributions include 1) designing an invertible system that can ensure strict consistency between forward and inverse mapping for edge systems with unobservability; 2) embedding the physical information in the forward mapping to indirectly regularize the inverse learning and avoid overfitting so that the state estimation can be conducted at an arbitrary operating point; and 3) showing how to embed the physical property comprehensively (functionals and variable size) so that information won't get lost due to dimension reduction for some use cases on the system edge.

2 Related Work

2.1 Solve the Inverse Problem of Physical Systems

It has been a basic task of interest to analyze the inverse process of physical/engineering system, which is to extract true states from observations for system operation and control [9,3]. In traditional works, researchers solve such problems by iterative simulations or algorithms based on models. These methods typically require prior system knowledge, e.g., solving power flow using the Newton–Raphson method with a detailed system model and estimating unknown states using Kalman Filter with system dynamics model [17,30,28,34]. However, the complete system information can often be unavailable or inaccurate in complex physical/engineering systems, especially on the system edge [22,11,19,10].

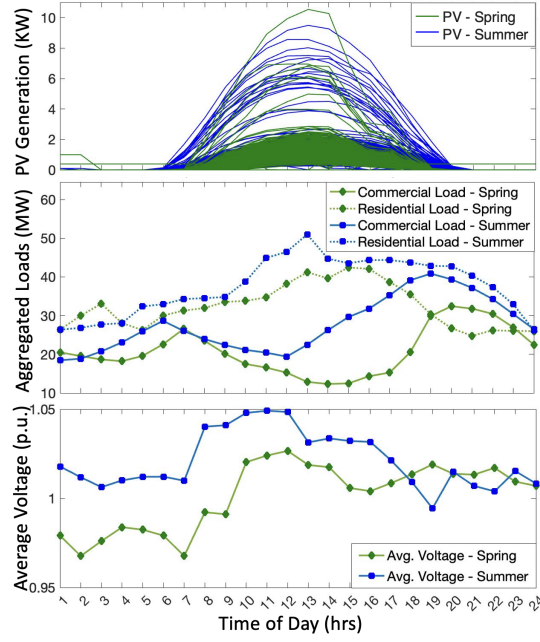


Fig. 1: Example of power distribution system to show the standard data pattern in spring and summer operations: (Top) generation of system-wide photovoltaics (PVs), (middle) aggregated commercial and residential loads, and (bottom) average voltage per unit values.

While traditional methods are limited, some studies propose to leverage machine learning tools, approximating the (inverse) mapping rule with data observations [23,35,37,14,26]. However, the data-driven methods either oversimplify the complex physical model or directly use intractable black-box approximation, lacking the interpretability and correctness for system operators to understand and trust. This calls for an invertible structure for learning the forward and inverse processes together with one-to-one mapping.

2.2 Enforce Inverse in Representation Learning

For the idea of unifying the forward and inverse learning, we trace back to the early work in conventional NN inversion that iteratively finds the optimal solutions [21,31]. However, inverting the highly nonlinear and implicit NN for optimum is difficult and computationally inefficient. Therefore, the family of representation learning uses a similar criterion but approximates inference instead of extensive optimization iterations. For example, the popular auto-encoders [15] connect two neural network models in sequence and in symmetry to approximate the inverse correlation while simultaneously training the forward NN. Specifically, the auto-encoder minimizes the reconstruction mismatch of inputs

to enforce an imperfect decoder that approximately inverts the encoder. In fact, the approximation error is unavoidable so that the true inverse counterpart cannot be reached in training. Moreover, as both the forward and inverse functions are black-box models, there is 1) no physical guarantee over implicit learning and 2) no physical meaning of the quantities in latent space.

In contrast, the flow-based models [7,8] construct a sequence of invertible transformations as the forward mapping. Compared to auto-encoders, they leverage the change of variable theorem to ensure a deterministic inverse of the forward mapping without any approximation. Previous work usually uses such a model for complex density estimation tasks like image generation, which are quite different from our target cases. To better represent complex image data, they map images to latent space with a simple distribution in the forward process first and then obtain an “easy” inverse. These models are trained by maximizing the likelihood, in an unsupervised learning manner, to find the solution in a high-density region, which can be viewed as the inverse of dataset [2]. Though the models show good performance in image generation, the design has a strong requirement for splitting the input and output, which is hard to be satisfied fully in physical systems. Also, the design doesn’t reveal any physical interpretability, which is necessary for physical system identification. Finally, the design does not consider the unobservability issue, either. Such problems require a comprehensive way to embed all possible physics knowledge from different perspectives.

3 Problem Formulation for Two-Way System Monitoring with Unobservability

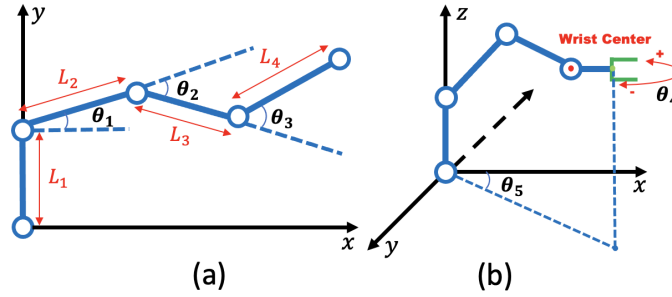


Fig. 2: Example: (a) geometry of the 3-DOF kinematics system in 2D space and (b) geometry of the 5-DOF kinematics system in 3D space.

The physical system identification is a supervised learning task to recover the forward system model f in $\mathbf{y} = f(\mathbf{x})$, mapping input variables $\mathbf{x} \in \mathcal{X}$ to output variables $\mathbf{y} \in \mathcal{Y}$. Subsequently, to infer desired states of physical systems, we aim to find an inverse mapping $g : \mathcal{Y} \rightarrow \mathcal{X}$ that satisfies $\mathbf{x} = g(\mathbf{y}) =$

$f^{-1}(\mathbf{y})$, $\forall \mathbf{y} \in \mathcal{Y}$. For instance, Figure 2 shows the end-effector position of a robot arm following forward kinematics functions of joint degrees of freedom. The inverse kinematics is to control the joint motions to reach the desired end-effector positions. For such a system, system information is usually required to understand the forward process, e.g., physical function types. When system information is unavailable, one can use machine learning to approximate the forward mapping in a data-driven manner. Unlike the forward mapping, state estimation is another monitoring tool based on inverse learning. However, the inverse learning process is even harder for problems like (1) hard to embed physical law, (2) can have multiple solutions, and (3) information loss due to uneven dimensions between input and output.

To solve these problems, we propose unifying the learning of two-way mappings in an invertible system identification. The two-way mappings indicate learning the forward-inverse pair. Specifically, we aim to learn the function of inverse mapping $\mathbf{x} = g(\mathbf{y})$ so that it work in a pair with the forward mapping $\mathbf{y} = f(\mathbf{x})$. Therefore, our goal is to let $\mathbf{y} = g^{-1}(\mathbf{x})$ approximate the analytical model $\mathbf{y} = f(\mathbf{x})$. invertible structure in the approximation model $g^{-1}(\cdot)$ to enforce an automatic inverse $\mathbf{x} = g(\mathbf{y})$ for state inference. In this learning process, we have two major targets: 1) to obtain a forward mapping rule that accurately approximates the system model and preserves physical interpretation as much as possible, and 2) to find a perfectly matched forward-inverse pair and estimate the most possible states under the partially observable scenario.

3.1 Optimization Objectives to Identify Invertible System Model

To reach the first goal, we form an optimization problem to find $g^{-1}(\cdot)$ as close as possible to the ground truth of the forward model $f(\cdot)$. For simple notation, we represent g^{-1} as hypothesis function h ,

$$h^* = \operatorname{argmin}_{h \in \mathcal{H}} \sum_{i=1}^N \ell_1(h(\mathbf{x}_i), \mathbf{y}_i), \quad (1)$$

where \mathcal{H} is a predefined class of hypothesis functions, e.g., parameterized neural networks. Since it is a supervised learning task, $\ell_1(\cdot)$ represents the regression loss function. We use mean square error to measure the mismatch in forward system model recovery. Moreover, for a perfect match of the two-way mappings, we follow the reconstruction loss used by auto-encoder,

$$h^* = \operatorname{argmin}_{h \in \mathcal{H}} \sum_{i=1}^N \ell_2(\mathbf{x}_i, h^{-1}(h(\mathbf{x}_i))), \quad (2)$$

where $\ell_2(\cdot)$ is the square loss and $h^{-1}(h(\mathbf{x}_i))$ denotes the reconstructed \mathbf{x}_i at the output of inverse mapping.

While the supervised learning loss penalizes errors in point estimates during training, it can not easily bypass the ill-conditioned problem for the inverse. Fortunately, physical/engineering systems have operation standards, as the power

system example in Fig. 1 shows. Only one solution is feasible to stay within the operating limits or satisfy specific patterns. To promote physically feasible solutions as the second target, we leverage the common criterion for statistical inference. In particular, estimating the probabilistic states \mathbf{x} is to maximize likelihood of the posterior probability density [7], which is

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmax}} p(\mathbf{x}|\mathbf{y}). \quad (3)$$

This process is to learn the invertible representation of real dataset. As long as we design an invertible function hypothesis function h , increasing the likelihood as in (3) contracts the original output data space to the high-density regions. Namely, it tends to locate a high-density data region and estimates the states that stay within the standard operation limit of physical systems

3.2 Virtual Storage Variables to Compensate System Unobservability

For system identification, the recovery of the forward model is sensitive to the data availability in the system. Unfortunately, modern physical/engineering systems are hard to guarantee full observability. Even worse, limited sensors behind the unobservability may lead to information reduction in the forward mapping, making inverse mapping with insufficient knowledge. Therefore, we propose adding virtual storage variables to the output of the forward mapping. All the input knowledge is preserved in the storage variables in the output of the forward mapping. For example, we propose using the network size to decide the number of storage variables. This is because the number of system states indicates the size of the minimum number of variables in a system that can recreate all the measurements in the network, according to the definition of state estimation. And, the number of state variables is typically the same as the network size. Using these variables will not only preserve information, but also format the physical units in the latent layer, which is due to a perfect match on the number of state variables. To exhibit such inherent properties in invertible system identification, we introduce virtual variables \mathbf{y}' on the output side. \mathbf{y}' is used to compensate the dimension reduction caused by unobservability while imitating the hidden quantities for homogeneous units in the final expression.

During training, the virtual quantities \mathbf{y}' are generated from simple orthogonal random variables, e.g., samples from standard isotropic Gaussian distributions. We observed that, compared to directly using Gaussian random variables, it's better to update the generation by a parameterized neural network. Specifically, we convert the virtual variables via a fully-connected NN and update this NN simultaneously with minimizing the reconstruction error in (2). It can better compensate for the information loss caused by unobservability. Thus, the inverse model changes to $\mathbf{x} = g(\mathbf{y}, \mathbf{y}')$. \mathbf{y}' are independent from observable \mathbf{y} and serve as factorial prior of system uncertainties to estimate the posterior.

4 Physically Invertible System Identification

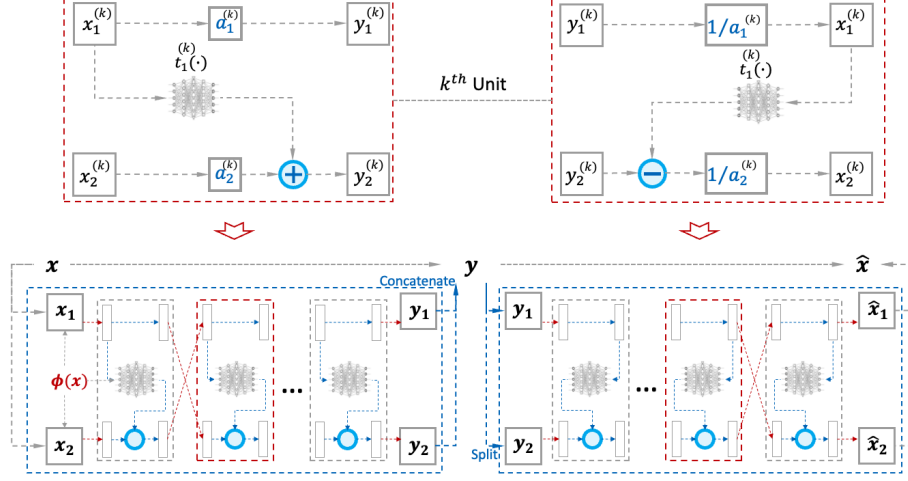


Fig. 3: Illustrate invertible transformation (top) and the structure of the proposed INN.

4.1 Invertible Transformation

To unify the learning of forward and inverse mappings, the key idea is to provide an invertible structure for system identification that find a pair of matched mappings. Enforcing the inversion of $g(\cdot)$ and $g^{-1}(\cdot)$, we consider change of variables, shown below. With $\mathbf{x} = g(\mathbf{y}, \mathbf{y}')$, the change of variables theorem shows

$$\begin{aligned} p(\mathbf{y}') &= p(\mathbf{x} = g(\mathbf{y}, \mathbf{y}') | \mathbf{y}) \left| \det \frac{\partial g(\mathbf{y}, \mathbf{y}')}{\partial \mathbf{y}, \mathbf{y}'} \right|, \\ p(\mathbf{x} | \mathbf{y}) &= p(\mathbf{y}') \left| \det \frac{\partial g(\mathbf{y}, \mathbf{y}')}{\partial \mathbf{y}, \mathbf{y}'} \right|^{-1}, \end{aligned} \quad (4)$$

where $\frac{\partial g(\mathbf{y}, \mathbf{y}')}{\partial \mathbf{y}, \mathbf{y}'}$ is the Jacobian matrix of function $g(\cdot)$ at \mathbf{x} and $\det(\cdot)$ represents the determinant of Jacobian. (4) serves as the theoretical basis of invertible function design. And, we need to find easily invertible functions with non-zero Jacobian determinant.

One intuitive way of invertible design is the linear and additive function, e.g., the forward $y = ax + b$ and the inverse $x = \frac{1}{a}y - \frac{b}{a}$. The determinant of

Jacobian is constant a to ensure invertibility. Motivated by the simple inverse, we follow the invertible design in [7] to split the multi-variate inputs and outputs and construct the following transformation unit:

$$\mathbf{y}_1^* = a_1 \mathbf{x}_1, \mathbf{y}_2^* = a_2 \mathbf{x}_2 + \mathbf{t}_1(\mathbf{x}_1), \quad (5)$$

$$\mathbf{x}_1 = \frac{1}{a_1}(\mathbf{y}_1^*), \mathbf{x}_2 = \frac{1}{a_2}(\mathbf{y}_2^* - \mathbf{t}_1(\mathbf{x}_1)), \quad (6)$$

where $\mathbf{y}^* = [\mathbf{y}, \mathbf{y}']$ for simple notation. Similar to the linear and additive function, the inverse mapping (6) is easy to derive and the determinant of Jacobian is $a_1 a_2$. Such a split formula is flexible that the nonlinear functions $\mathbf{t}_1(\cdot)$ can be arbitrarily complex for representation, without affecting the invertible property.

The unit transforms one part of inputs for invertibility and leaves the other untouched. To enable complete coupling of all dimensions, we composite several units and transform each part in turn. We show in the following proposition 1 that more than three compositions are necessary to completely transform all inputs dimensions and coupling with the output.

Proposition 1. *With each transformation unit in (5), more than three compositions are necessary to completely transform all input dimensions and coupling with the output.*

The proof is intuitive by deriving the Jacobian matrix of the composited invertible functions. For the k^{th} unit, the Jacobian is

$$J_k = \begin{bmatrix} \frac{\partial \mathbf{y}_1^{(k)}}{\partial \mathbf{x}_1^{(k)}} & \frac{\partial \mathbf{y}_1^{(k)}}{\partial \mathbf{x}_2^{(k)}} \\ \frac{\partial \mathbf{y}_2^{(k)}}{\partial \mathbf{x}_1^{(k)}} & \frac{\partial \mathbf{y}_2^{(k)}}{\partial \mathbf{x}_2^{(k)}} \end{bmatrix} = \begin{bmatrix} a_1^{(k)} I_1 & \mathbf{0} \\ \frac{\partial \mathbf{t}_1^{(k)}(\mathbf{x}_1)}{\partial \mathbf{x}_1^{(k)}} & a_2^{(k)} I_2 \end{bmatrix}. \quad (7)$$

For every other layer, the columns exchange due to the in-turn transformation. Using the chain rule, the Jacobian of the composited function is $\Pi_k J_k$. Only when $k \geq 3$, the 0's are eliminated from Jacobian matrix and thus indicate a full transformation of all dimensions.

Fig. 3 (top) illustrates the invertible transformation. As for the NN structure in the bottom, with each unit to be invertible, the sequence of composited units is invertible, and the Jacobian determinant is easily computed for optimizing (3).

4.2 Building Invertible NN Structure for Physical Interpretability

The sequence of invertible transformations is trained to maximize the likelihood of the training data. However, the unsupervised learning manner performs poorly in generalizing to the out-of-range dataset and reaching global optimum [16]. Furthermore, unlike image density estimation, extrapolation is often the primary concern in the physical system when new operation points occur and have never been recorded in the historical data. In such cases, an accurate inverse solution requires perfect forward mapping learning to recover the governing function of a

physical system. Thus, we aim to find a hypothesis that not only minimizes the empirical prediction error (1) but also reveals the underlying analytical function.

However, it is challenging to meet the latter target as any large physical systems (e.g., power, water, traffic systems) have limited sensor deployment for full observability. For these cases, we need to simultaneously recover governing functions in the observable region and approximate hidden correlation in data whenever physical recovery is impossible. According to [36], we express the ground truth $f(\cdot)$ in the form $\mathbf{y} = f(\mathbf{x}) = f_1(\mathbf{x}) + f_2(\mathbf{x}) = W_1\phi(\mathbf{x}) + f_2(\mathbf{x})$. $f_1(\cdot)$ denotes the recoverable physical law of the observable, and $f_2(\cdot)$ denotes the mapping regarding the unobservable region. Learning $f_1(\mathbf{x})$ only is a system identification problem, where $\phi(\mathbf{x})$ are the physical features of specific systems (e.g., coupling of quadratic and sinusoidal terms for power system) and W_1 represents unknown system parameters to be recovered. To enable physical interpretability, we embed $\phi(\mathbf{x})$ into the invertible hypothesis function (5). In this way, the invertible unit can reveal physics and match the underlying model during learning.

Thus, the proposed invertible NN structure unifies the forward and inverse mappings. The model is trained by optimizing two loss functions simultaneously to reach the optimal inverse solution. On one hand, using the supervised learning loss aims to minimize the mismatch of sample predictions and makes the forward mapping as close to the governing function as possible. On the other hand, using the unsupervised learning loss focuses on a high-density region to avoid ill-conditioned problems in an inverse process. In practice, we observe a trade-off between the two loss terms. Therefore, a hyperparameter is adopted to balance the penalization. The hyperparameter is chosen through cross-validation in the experiments. By training the invertible NN structure, if we find the optimal forward mapping that reveals physics, we naturally obtain the inverse following physical laws.

5 Experiments

The proposed invertible NN is applicable for various inverse problems in physical systems. We validate the algorithm on kinematics systems, power systems, robotic systems, and high-rise buildings (structural health). The results are similar, so we focus on the two most representative systems for in-depth evaluation with respect to each of the proposed designs. They are the inverse kinematics, where hidden states follow one-way cascading correlation, and the inverse power flow, where states yield two-way interactive correlation.

Evaluation Criteria: Learning the inverse mapping in physical system can be seen as a regression problem. Therefore, we use the evaluation metric mean square error (MSE) for state estimation. For the physical system analysis, the interpretability is essential so that we evaluate by the accuracy of learning system parameters for the forward system model. The higher the accuracy, the more reducible is the learned model.

Baselines: We compare the proposed model with the following state-of-the-art baselines on learning the inverse system mapping: support vector regression (SVR) with polynomial kernel or RBF kernel [35], residual neural network (ResNet) [12], variational autoencoder (VAE) to approximate the forward-inverse pair [15], NICE/RealNVP to learn the invertible transformation [7,8]. The first two methods directly learn the inverse mapping while the other two methods enforce the inverse model from forward model to obtain inverse solutions [13,37]. In particular, we use the same architecture (depth, width, and activation) for the NN $t_1(\cdot)$ in invertible structure, ResNet, and auto-encoder. We showed previously that at least three invertible units are required to completely transform all dimensions. Therefore, the depth of NNs is a hyper-parameter selected from 3 – 10 layers in validation, and the width depends on the problem size of the test system.

The Adam optimizer is used to train NNs for 200 epochs for each experiment, where we set up a learning rate hyper-parameter set $\{0.001, 0.0002, 0.00005\}$, and momentum parameters $\beta_1 = 0.5, \beta_2 = 0.999$. All the experiments are implemented on a computer equipped with Inter(R) Core(TM) i7-9700k CPU and Nvidia Geforce RTX 3090 GPU.

Table 1: The prediction errors ($\text{MSE} \times 10^{-3}$) of invertible kinematics system identification: the inverse solution and forward mapping recovery.

| Case | Model | Joint Angle Prediction | Forward Model Prediction |
|-------|--------------|------------------------|--------------------------|
| 3-DOF | SVR | 0.0004 ± 0.00 | N/A |
| | ResNet | 0.001 ± 0.00 | N/A |
| | VAE | 0.001 ± 0.01 | 0.0015 ± 0.00 |
| | RealNVP | 0.0005 ± 0.00 | 0.0004 ± 0.00 |
| | Proposed INN | 0.0002 ± 0.00 | 0.0001 ± 0.00 |
| 5-DOF | SVR | 0.19 ± 0.07 | N/A |
| | ResNet | 0.12 ± 0.03 | N/A |
| | VAE | 0.10 ± 0.04 | 0.09 ± 0.02 |
| | RealNVP | 0.08 ± 0.01 | 0.04 ± 0.02 |
| | Proposed INN | 0.06 ± 0.02 | 0.02 ± 0.00 |

5.1 Inverse Kinematics Problem

To test the applicability of the proposed model on physical inverse process, we start with a basic inverse kinematics problem, where instruments are not fully equipped to collect all the data. As shown in Figure 2(a), the movement of end-effectors is determined by multiple degrees of freedom (DOF) chains in the robotic systems. The manipulator in 2D space moves with the rotations of three joints (3 DOFs) that connect 4 rigid parts. The task is to find the most likely joint motions to reach the desired end-effector position. Given the configuration

of joint angles, the forward kinematics equations describe the motion of the hierarchical skeleton structure. However, the system parameters, e.g., joint lengths, are unknown. We aim to identify the possible rotation angles of three joints given the expected end-effector coordinates. In this case, 1000 different configurations are sampled for training and random Gaussian noises are added ($\mathcal{N}(0, 0.01)$).

For a more complex setup, we consider the manipulator in 3D space with 5 DOFs (Figure 2(b)). The new DOFs in the added dimension are intractable where measurements of θ_4 and θ_5 are unavailable. In this case, we evaluate the prediction of joint rotations in inverse process. Moreover, we evaluate the partial recovery of the governing function on observable parts in the forward system identification. Table 1 compares the numerical results of the proposed physics-interpretable invertible NN with the baselines.

For the 3-DOF setup, both SVR and the proposed model have good estimation results. Specifically, our physics-interpretable invertible NN outperforms the original RealNVP due to the physics embedding in the forward mapping. It can also be verified by the accuracy of system parameter recovery, where the proposed INN reaches near 100% for this fully observable case. For the 5-DOF case that has some unobservables, the variational auto-encoder and RealNVP have much lower errors than the first two models that directly approximate the inverse process. Although the proposed invertible neural network can not recover all the system parameters due to the unobservability, it shows a generally lower error in estimating inverse solution than the original RealNVP.

5.2 Inverse Power Flow Problem: Distribution System State Estimation

After the demonstration of the basic kinematics problem, we test the proposed model on more complex and larger systems. Different from the single link in the manipulator, the standard power system can be seen as a graph with many internal couplings. The real utility feeder usually has more complex connections and a larger scale. For an N -node power system, the governing physical law is the classic power flow equations (PF) [35]. The power system state estimation (SE) is of great interest for many downstream operation applications [24,37]. Estimating voltage phasor states from standard measurements (e.g., power injections, branch power flows, and current magnitudes) is an inverse process of power flow analysis. Test feeders IEEE 8- and 123-bus networks, and a utility feeder (2721 nodes with 371 active ones) are used for experiments, shown as 8-bus, 123-bus, and Utility in Table 2. Since ground truth data is not directly available, we conduct traditional simulations with one-year real power data (15-min interval) in MATPOWER [38]. The model information is only available to prepare the dataset and remains unknown during training. The real-world measurements usually have errors due to communication issues. We add random Gaussian noise with a 1% – 2% standard deviation to simulate the measurement errors (as usually used by state estimation). Moreover, we prepare out-of-range data ($3\times$ PV generation and loads) to validate extrapolation capability.

Table 2: The prediction errors ($\text{MSE} \times 10^{-3}/\text{p.u.}$) of power system cases: the inverse state estimation and the forward power flow mapping.

(a) Testing on the in-range data scenario.

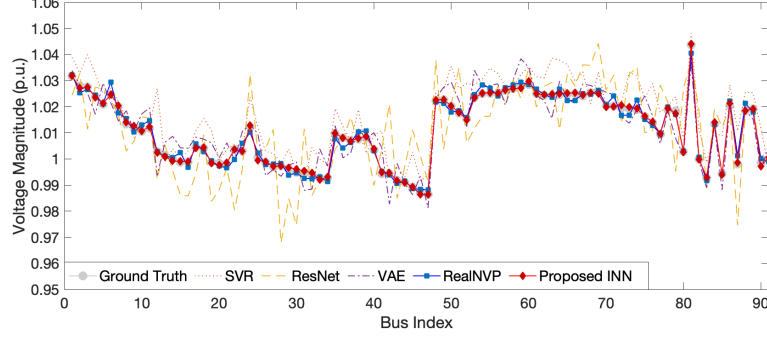
| Scenario | Case | SVR | ResNet | VAE | RealNVP | Proposed INN |
|------------------|---------|-----------------|-----------------|-----------------|------------------|------------------------------------|
| SE (In-Range) | 8-bus | 0.08 ± 0.02 | 0.04 ± 0.00 | 0.03 ± 0.01 | 0.008 ± 0.00 | 0.006 ± 0.00 |
| | 123-bus | 0.21 ± 0.04 | 0.17 ± 0.02 | 0.13 ± 0.03 | 0.09 ± 0.01 | 0.05 ± 0.03 |
| | Utility | 0.27 ± 0.12 | 0.23 ± 0.05 | 0.16 ± 0.03 | 0.13 ± 0.07 | 0.11 ± 0.02 |
| PF (In-Range) | 8-bus | N/A | N/A | 0.05 ± 0.01 | 0.007 ± 0.00 | 0.002 ± 0.00 |
| | 123-bus | N/A | N/A | 0.11 ± 0.06 | 0.06 ± 0.03 | 0.02 ± 0.01 |
| | Utility | N/A | N/A | 0.15 ± 0.01 | 0.13 ± 0.02 | 0.04 ± 0.03 |

(b) Testing on the out-of-range data scenario for extrapolation.

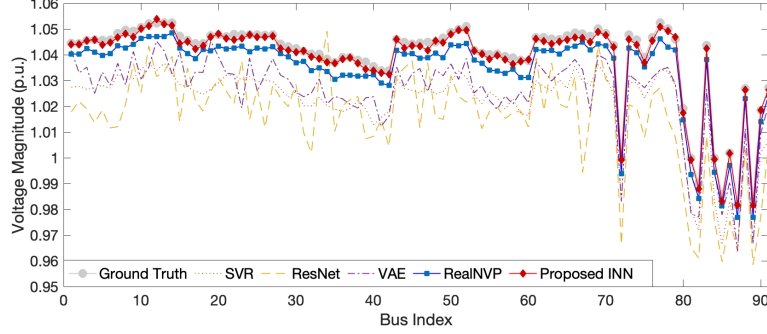
| Scenario | Case | SVR | ResNet | VAE | RealNVP | Proposed INN |
|-----------------------|---------|-----------------|-----------------|-----------------|-----------------|------------------------------------|
| SE (Extrapolation) | 8-bus | 0.14 ± 0.04 | 0.09 ± 0.03 | 0.09 ± 0.02 | 0.03 ± 0.01 | 0.009 ± 0.00 |
| | 123-bus | 0.29 ± 0.11 | 0.22 ± 0.06 | 0.25 ± 0.02 | 0.15 ± 0.03 | 0.07 ± 0.02 |
| | Utility | 0.43 ± 0.19 | 0.35 ± 0.02 | 0.31 ± 0.09 | 0.19 ± 0.06 | 0.15 ± 0.04 |
| PF (Extrapolation) | 8-bus | N/A | N/A | 0.07 ± 0.03 | 0.04 ± 0.02 | 0.004 ± 0.00 |
| | 123-bus | N/A | N/A | 0.21 ± 0.03 | 0.18 ± 0.07 | 0.06 ± 0.03 |
| | Utility | N/A | N/A | 0.24 ± 0.06 | 0.22 ± 0.05 | 0.11 ± 0.05 |

The numerical results of estimation are included in Table 2 and Fig. 4 to compare different methods. As we explained, SE denotes the inverse process while PF denotes the forward mapping recovery. First, we observe a general decrease in MSEs for forward-inverse learning methods compared to the direct inverse learning methods (SVR and ResNet).

While the errors of inverse solutions are small, we look back to the forward learning. VAE has a relatively poor result as the reconstruction errors cannot reach zeros in approximation. Although RealNVP naturally has the perfect correspondence to learn an explicit forward, the proposed INN outperforms it by a large margin for forward mapping recovery. This could also be explained by the ablation study of our proposed model. For the observable region, the governing PF function can be recovered by the proposed INN. The ablation study results (Table. 3) demonstrate how physics embedding greatly impacts the forward model recovery. Without physics consistency in learning model, both the MSEs of inverse estimation and forward output prediction are higher. Further, the comparison of state estimation given in-range and out-of-range inputs in Fig. 4a and Fig. 4b reveals a better extrapolation capability of the proposed INN. During the experiments, we observe that, when there is no physics embedding, increasing the weight of the density estimation loss can lower the MSE slightly.



(a) Estimated voltage magnitudes given in-range inputs (generation and load)



(b) Estimated voltage magnitudes given out-of-range inputs for extrapolation evaluation.

Fig. 4: Validating estimation results of all the nodes (from feeder head to end) on one phase of 123-bus system.

6 Conclusion

In this paper, we propose a physics-interpretable inverse learning method to tackle the challenge of solving the inverse process of physical systems. Rather than a direct approximation, we unify the forward and inverse learning, and simultaneously optimize over the pair of mappings. The proposed method takes advantage of the flexible NN structure and the recent advances in density estimation to guarantee a perfect forward-inverse pair and solve the ill-conditioned physical systems problem. Moreover, since the generative model has limitations in the adversarial task of physical system identification, we embed physics into the invertible structure to enable interpretability and further enforce the inverse solution following physical laws. Numerical experiments have been conducted on physical/engineering systems with typical couplings to evaluate the proposed

Table 3: Ablation study of the proposed invertible neural network.

| Scenario | Case | Proposed INN | W/o Physics Embedding | W/o Virtual Variables |
|-----------------------|---------|---------------------|-----------------------|-----------------------|
| SE (In-Range) | 8-bus | 0.006 ± 0.00 | 0.01 ± 0.00 | 0.02 ± 0.01 |
| | 123-bus | 0.05 ± 0.03 | 0.08 ± 0.02 | 0.07 ± 0.01 |
| | Utility | 0.11 ± 0.02 | 0.15 ± 0.05 | 0.17 ± 0.03 |
| PF (In-Range) | 8-bus | 0.002 ± 0.00 | 0.003 ± 0.00 | 0.002 ± 0.00 |
| | 123-bus | 0.02 ± 0.01 | 0.05 ± 0.02 | 0.02 ± 0.01 |
| | Utility | 0.04 ± 0.03 | 0.15 ± 0.06 | 0.05 ± 0.02 |
| SE (Extrapolation) | 8-bus | 0.009 ± 0.00 | 0.04 ± 0.01 | 0.04 ± 0.00 |
| | 123-bus | 0.07 ± 0.02 | 0.13 ± 0.05 | 0.16 ± 0.03 |
| | Utility | 0.15 ± 0.04 | 0.23 ± 0.03 | 0.23 ± 0.07 |
| PF (Extrapolation) | 8-bus | 0.004 ± 0.00 | 0.03 ± 0.02 | 0.005 ± 0.00 |
| | 123-bus | 0.06 ± 0.03 | 0.21 ± 0.05 | 0.08 ± 0.02 |
| | Utility | 0.11 ± 0.05 | 0.26 ± 0.02 | 0.15 ± 0.04 |

method. Our model outperforms the baseline methods on both the inverse process learning and the forward model recovery and output prediction.

References

1. Abdel-Majeed, A., Tenbohlen, S., Schöllhorn, D., Braun, M.: Development of state estimator for low voltage networks using smart meters measurement data. In: IEEE Grenoble Conference. pp. 1–6 (2013)
2. Bengio, Y., Mesnil, G., Dauphin, Y., Rifai, S.: Better mixing via deep representations. In: International conference on machine learning. pp. 552–560. PMLR (2013)
3. Benning, M., Burger, M.: Modern regularization methods for inverse problems. arXiv preprint arXiv:1801.09922 (2018)
4. Bhela, S., Kekatos, V., Veeramachaneni, S.: Enhancing Observability in Distribution Grids Using Smart Meter Data. IEEE Transactions on Smart Grid **9**(6), 5953–5961 (2018). <https://doi.org/10.1109/TSG.2017.2699939>
5. Botín-Sanabria, D.M., Mihaita, A.S., Peimbert-García, R.E., Ramírez-Moreno, M.A., Ramírez-Mendoza, R.A., Lozoya-Santos, J.d.J.: Digital twin technology challenges and applications: A comprehensive review. Remote Sensing **14**(6), 1335 (2022)
6. Deka, D., Backhaus, S., Chertkov, M.: Learning topology of the power distribution grid with and without missing data. In: European Control Conference. pp. 313–320 (2016)
7. Dinh, L., Krueger, D., Bengio, Y.: NICE: Non-linear Independent Components Estimation. arXiv preprint arXiv:1410.8516 (2014)
8. Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using Real NVP. arXiv preprint arXiv:1605.08803 (2016)
9. Engl, H.W., Hanke, M., Neubauer, A.: Regularization of inverse problems, vol. 375. Springer Science & Business Media (1996)
10. Hamdan, S., Ayyash, M., Almajali, S.: Edge-computing architectures for internet of things applications: A survey. Sensors **20**(22), 6441 (2020)

11. Haque, M.E., Zain, M.F., Hannan, M.A., Rahman, M.H.: Building structural health monitoring using dense and sparse topology wireless sensor network. *Smart Structures and Systems* **16**(4), 607–621 (2015)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016)
13. Hu, X., Hu, H., Verma, S., Zhang, Z.L.: Physics-guided deep neural networks for powerflow analysis. *arXiv preprint arXiv:2002.00097* (2020)
14. Karlik, B., Aydin, S.: An improved approach to the solution of inverse kinematics problems for robot manipulators. *Engineering applications of artificial intelligence* **13**(2), 159–164 (2000)
15. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013)
16. Kirichenko, P., Izmailov, P., Wilson, A.G.: Why normalizing flows fail to detect out-of-distribution data. *arXiv preprint arXiv:2006.08545* (2020)
17. Kucuk, S., Bingul, Z.: Robot kinematics: Forward and inverse kinematics. INTECH Open Access Publisher (2006)
18. Kuo, Y.L., Tang, S.C.: Dynamics and control of a 3-dof planar parallel manipulator using visual servoing resolved acceleration control. *Journal of Low Frequency Noise, Vibration and Active Control* p. 1461348419876154 (2019)
19. Liao, L., Fox, D., Hightower, J., Kautz, H., Schulz, D.: Voronoi tracking: Location estimation using sparse and noisy sensor data. In: *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*(Cat. No. 03CH37453). vol. 1, pp. 723–728. IEEE (2003)
20. Liao, Y., Weng, Y., Liu, G., Zhang, Z., Tan, C.W., Rajagopal, R.: Unbalanced Three-Phase Distribution Grid Topology Estimation and Bus Phase Identification. *IET Smart Grid* **2**(4), 557–570 (2019)
21. Linden, A., Kindermann, J.: Inversion of multilayer nets. In: *Proc. Int. Joint Conf. Neural Networks*. vol. 2, pp. 425–430 (1989)
22. Liu, M.Z., Ochoa, L.N., Riaz, S., Mancarella, P., Ting, T., San, J., Theunissen, J.: Grid and market services from the edge: Using operating envelopes to unlock network-aware bottom-up flexibility. *IEEE Power and Energy Magazine* **19**(4), 52–62 (2021)
23. Liu, Y., Zhang, N., Wang, Y., Yang, J., Kang, C.: Data-Driven Power Flow Linearization: A Regression Approach. *IEEE Transactions on Smart Grid* **10**(3), 2569–2580 (2019)
24. Mestav, K.R., Luengo-Rozas, J., Tong, L.: Bayesian state estimation for unobservable distribution systems via deep learning. *IEEE Transactions on Power Systems* **34**(6), 4910–4920 (2019)
25. Mnih, A., Gregor, K.: Neural variational inference and learning in belief networks. *International Conference on Machine Learning* pp. 1791–1799 (2014)
26. Müller, H.H., Rider, M.J., Castro, C.A.: Artificial Neural Networks for Load Flow and External Equivalents Studies. *Electric power systems research* **80**(9), 1033–1041 (2010)
27. Pan, S., Bonde, A., Jing, J., Zhang, L., Zhang, P., Noh, H.Y.: Boes: building occupancy estimation system using sparse ambient vibration monitoring. In: *Sensors and smart structures technologies for civil, mechanical, and aerospace systems 2014*. vol. 9061, p. 90611O. International Society for Optics and Photonics (2014)
28. Pei, Y., Biswas, S., Fussell, D.S., Pingali, K.: An elementary introduction to kalman filtering. *Communications of the ACM* **62**(11), 122–133 (2019)

29. Schmidt, M., Lipson, H.: Distilling free-form natural laws from experimental data. *science* **324**(5923), 81–85 (2009)
30. Tinney, W.F., Hart, C.E.: Power flow solution by newton’s method. *IEEE Transactions on Power Apparatus and Systems* (11), 1449–1460 (1967)
31. Varkonyi-Koczy, A.R., Rovid, A.: Observer based iterative neural network model inversion. In: *The 14th IEEE International Conference on Fuzzy Systems*, 2005. FUZZ ’05. pp. 402–407 (2005)
32. Vitus, M.P., Tomlin, C.J.: Sensor placement for improved robotic navigation. *Robotics: science and Systems VI* p. 217 (2011)
33. Wang, L., Zhou, Q., Jin, S.: Physics-guided deep learning for power system state estimation. *Journal of Modern Power Systems and Clean Energy* **8**(4), 607–615 (2020)
34. Weng, Y., Negi, R., Ilić, M.D.: Historical data-driven state estimation for electric power systems. In: *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. pp. 97–102. IEEE (2013)
35. Yu, J., Weng, Y., Rajagopal, R.: Robust Mapping Rule Estimation for Power Flow Analysis in Distribution Grids pp. 1–6 (2017)
36. Yuan, J., Weng, Y.: Physics interpretable shallow-deep neural networks for physical system identification with unobservability. *IEEE International Conference on Data Mining (ICDM)* (2021)
37. Zamzam, A.S., Sidiropoulos, N.D.: Physics-Aware Neural Networks for Distribution System State Estimation. *IEEE Transactions on Power Systems* (2020)
38. Zimmerman, R.D., Murillo-Sánchez, C.E., Thomas, R.J.: Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. *IEEE Transactions on Power Systems* **26**(1), 12–19 (2011)