Learning Power System Dynamics with Nearly-Hamiltonian Neural Network

Shaorong Zhang and Nanpeng Yu
Department of Electrical and Computer Engineering
University of California, Riverside
California, USA
nyu@ece.ucr.edu

***Abstract*—The ability to learn power system dynamic model and predict transient trajectories using data is crucial to realizing closed-loop control of the system with artificial intelligence. This paper proposes a Nearly-Hamiltonian neural network to predict transient trajectories and dynamic parameters of the power system by embedding energy conservation laws in the proposed neural network architecture. This inductive bias empowers the proposed model to learn the power system dynamics without explicitly using the exact functional form of the power system dynamic equations. The numerical study results on the single machine infinite bus system show that the proposed model produces accurate system trajectories and damping coefficient predictions. Furthermore, the proposed model significantly outperforms the baseline and Hamiltonian neural network.**

***Index Terms*—Hamiltonian system, Nearly-Hamiltonian neural network, power system dynamics.**

I. INTRODUCTION

The ability to learn complex power system dynamics and predict post-disturbance transient trajectories is critical for the power system operators to perform stability analysis and take appropriate corrective control actions in real-time operations. This capability is becoming increasingly important as the renewable generation penetration level continue to grow.

Traditionally, power system operators have relied on online dynamic simulations to perform transient stability assessment. However, solving a large number of differential algebraic equations representing the power system dynamics is computationally intensive. Furthermore, it is challenging to obtain accurate dynamic parameters of various system components. Some of the dynamic parameters (e.g., parameters of load models) could be changing quickly over time.

The wide-spread adoption of phasor measurement units (PMUs) and the advancement of machine learning algorithms make it possible to develop deep neural networks to learn the power system dynamic model [1], estimate dynamic parameters [2], evaluate health index and stability margin [3]–[5], and predict transient trajectories [6], [7]. The machine learning models adopted and developed in the existing literature can be divided into three groups. In the first group of methods, tailored deep neural networks such as feed-forward neural network [3], cascaded convolutional neural network (CNN) [4] and hierarchical CNN [5] are used to predict critical clearing time, perform stability assessment, and predict stability margin. Since a limited amount of physical models are embeded in these machine learning models, a large number of training samples are needed to achieve reasonable prediction results.

In the second group of methods, physics-aware machine learning models such as the structure-informed graph learning approach [6] and Fourier neural operator were developed to predict system trajectories based on recent state trajectories [7]. By injecting system graph information and inductive bias about a small number of modes in the frequency domain, these models are capable of producing fairly accurate transient trajectory predictions in an online manner. However, there is no guarantee that the predicted system trajectories will satisfy the basic energy conservation laws in the electric network.

In the third group of methods, the exact functional forms of power system dynamic equations are embeded in the machine learning model yielding physics-informed neural network [1] and physics-based neural ordinary differential equations [2]. These physics-based neural network models require substantially less training data, have simpler and sometimes interpretable network structures, and achieve high accuracy. However, given the increasing penetration of renewable generation and distributed energy resources, power system operators often do not know the exact functional form of the power system dynamics equations. This makes it difficult to apply these powerful physics-based methods in practice.

This paper tries to embed key physics priors, energy conservation laws, as inductive bias in the proposed Nearly-Hamiltonian neural network (NHNN) to predict system trajectories and estimate the dynamic parameters without explicitly using the exact functional form of the power system dynamic equations. To illustrate how the proposed algorithm can be applied to learn power system dynamics, the single machine infinite bus (SMIB) system is first formulated as a Nearly-Hamiltonian system. Then a deep neural network is trained to learn the Hamiltonian function and the generator damping coefficient by minimizing the difference between the estimated and measured gradient of the generalized positions and generalized momentum of the Nearly-Hamiltonian system. This paper is inspired by the recent work on the Hamiltonian neural network (HNN) [8] but extends it by considering dissipative dynamical systems such as power systems with damping effects. The proposed NHNN has broader applicability than the physics-informed neural network as it does not explicitly use the exact functional form of the dynamic equations of power system. It also improves upon the physics-aware neural networks by endowing the model with the ability to learn conserved or nearly conserved quantities in the power system dynamics.

The remainder of this paper is organized as follows. Section II formulates the SMIB system as a Nearly-Hamiltonian system and proposes the NHNN to learn in this dynamical system. Section III presents the numerical studies. Section IV gives the conclusion.

II. Technical Methods

In this section, we first review the preliminaries of Hamiltonian and Nearly-Hamiltonian systems. Then, the SMIB power system dynamic model is introduced and formulated as a Nearly-Hamiltonian system. Finally, the NHNN is proposed to learn the power system dynamics.

A. Review of Hamiltonian and Nearly-Hamiltonian Systems

A dynamical system can be modeled as a Hamiltonian system governed by Hamiltonian's equations. Let us consider a dynamical system with N coordinate pairs $(q_1, p_1) \cdots (q_N, p_N)$, where $\mathbf{q} = [q_1, q_2, \cdots, q_N]^T$ represents the generalized positions and $\mathbf{p} = [p_1, p_2, \cdots, p_N]^T$ represents the generalized momentum. The Hamiltonian $H(\mathbf{q}, \mathbf{p})$ is defined as a scalar function of $\mathbf{q}$ and $\mathbf{p}$. The standard Hamiltonian differential equations for conservative mechanical systems are given by [8]:

$$\frac{d\mathbf{q}}{dt} = \frac{\partial H}{\partial \mathbf{p}}, \quad \frac{d\mathbf{p}}{dt} = -\frac{\partial H}{\partial \mathbf{q}} \tag{1}$$

The state space with local coordinates $(\mathbf{q}, \mathbf{p})$ is called the phase space. The Hamiltonian H is the total energy of the system. Using (1), it can be shown that the total energy is conserved:

$$\frac{dH}{dt} = \frac{\partial^T H}{\partial \mathbf{q}}\frac{d\mathbf{q}}{dt} + \frac{\partial^T H}{\partial \mathbf{p}}\frac{d\mathbf{p}}{dt} = 0 \tag{2}$$

(1) implies that moving coordinates in the direction of symplectic gradient $\mathbf{S}_H = (\frac{\partial H}{\partial \mathbf{p}}, -\frac{\partial H}{\partial \mathbf{q}})$ does not change the total system energy. Hence, the trajectory of $(\mathbf{q}, \mathbf{p})$ can be regarded as a contour of the Hamiltonian function $H(\mathbf{q}, \mathbf{p})$.

In many real-world physical systems, loss of energy to the outside environment cannot be neglected. To model such systems, nearly-Hamiltonian systems are studied by generalizing Hamiltonian systems to dynamic systems with energy dissipation [9]. The state equations of nearly-Hamiltonian systems can be represented as:

$$\frac{d\mathbf{q}}{dt} = \frac{\partial H}{\partial \mathbf{p}}, \quad \frac{d\mathbf{p}}{dt} = -\frac{\partial H}{\partial \mathbf{q}} - \mathbf{D}\frac{\partial H}{\partial \mathbf{p}}, \tag{3}$$

where $\mathbf{D} \in \mathcal{R}^{N \times N}$ is a positive semi-definite matrix. $-\mathbf{D}\frac{\partial H}{\partial \mathbf{p}}$ describes the damping effects of the system. It can be deduced that the derivative of H is non-positive,

$$\frac{dH}{dt} = \frac{\partial^T H}{\partial \mathbf{q}}\frac{d\mathbf{q}}{dt} + \frac{\partial^T H}{\partial \mathbf{p}}\frac{d\mathbf{p}}{dt} = -\frac{\partial^T H}{\partial \mathbf{p}}\mathbf{D}\frac{\partial H}{\partial \mathbf{p}} \leq 0 \tag{4}$$

B. Formulate the SMIB System as a Nearly-Hamiltonian System

In power system stability analysis and control, the SMIB model, shown in Fig. 1, represents the situation where a power plant is connected to the rest of the power grid, which has a much greater capacity than that of the plant [10]. This model has been used to analyze the fundamental dynamic phenomena occurring in power systems. In this paper, we focus on this basic model to introduce the Nearly-Hamiltonian neural network. In the future, we plan to learn more complex power system dynamics using the NHNN.

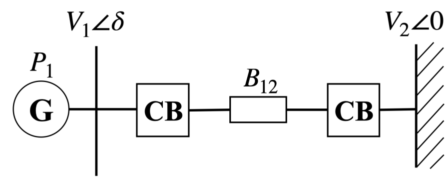

Fig. 1. SMIB system

The swing equation for the SMIB system is given by [1]:

$$m_1\ddot{\delta} + d_1\dot{\delta} + B_{12}V_1V_2\sin(\delta) - P_1 = 0, \tag{5}$$

where m_1 is the generator inertia constant, d_1 represents the damping coefficient, B_{12} is the bus susceptance of the line between bus 1 and bus 2, V_1 and V_2 are the voltage magnitudes at buses 1, 2. δ represent the voltage angles difference between buses 1 and 2. The position and momentum coordinates are defined as $q = \delta$, $p = m_1\dot{\delta}$, then (5) can be rewritten as:

$$\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -d_1 \end{bmatrix} \begin{bmatrix} B_{12}V_1V_2\sin(q) - P_1 \\ \frac{p}{m} \end{bmatrix} \tag{6}$$

The Hamiltonian H of the SMIB system satisfy:

$$\frac{\partial H}{\partial q} = B_{12}V_1V_2\sin(q) - P_1 \tag{7}$$

$$\frac{\partial H}{\partial p} = \frac{p}{m} \tag{8}$$

Thus, the Hamiltonian function can be derived as:

$$H = -P_1 q - B_{12}V_1V_2\cos(q) + \frac{p^2}{2m_1} \tag{9}$$

With the Hamiltonian definition, (6) can be rewritten as:

$$\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -d_1 \end{bmatrix} \begin{bmatrix} \frac{\partial H}{\partial q} \\ \frac{\partial H}{\partial p} \end{bmatrix} \tag{10}$$

C. Learning SMIB System Dynamics with a Nearly-Hamiltonian Neural Network

Although HNN [8] has been shown to be quite effective in learning the dynamics of physical systems that conserve energy, it is infeasible to apply it to learn the dynamics of the SMIB system when the damping effects are taken into consideration. In this subsection, a new physics-informed machine learning model called Nearly-Hamiltonian neural network (NHNN) is proposed to learn the Hamiltonian and damping coefficient concurrently.

The architecture of the baseline neural network, HNN and NHNN are shown in Fig. 2. The baseline neural network tries to learn the differential equations, that is: $\begin{bmatrix}\dot{\mathbf{q}} & \dot{\mathbf{p}}\end{bmatrix} = f_\theta(\mathbf{q}, \mathbf{p})$, where $f_\theta(\mathbf{q}, \mathbf{p})$ is a neural network with parameter vector θ. Here the baseline neural network is a multilayer perceptron. Similarly, the HNN takes $\mathbf{q}$ and $\mathbf{p}$ as inputs. Instead of learning the differential equations directly, HNN first aims at learning the Hamiltonian H of the system. Then the derivatives of the states $\dot{\mathbf{q}}$ and $\dot{\mathbf{p}}$ can be automatically calculated using (1). HNN yields a better performance in the task of learning the dynamics of energy conserving systems.

The proposed NHNN augments the architecture of HNN by considering the damping effects in the nearly-Hamiltonian system. Let the state variables $\mathbf{x}$ be the combination of the generalized positions and momentum $(\mathbf{q}, \mathbf{p})$. Then the nearly-Hamiltonian system can be rewritten as a linear combination of the gradient of the Hamiltonian: $\dot{\mathbf{x}} = (\mathbf{J} - \mathbf{R})\nabla H$. The matrix $\mathbf{J}$ is skew-symmetric and $\mathbf{R}$ is positive semi-definite and can be uniquely identified. In the SMIB system, $\mathbf{J}$ and $\mathbf{R}$ are 2×2 matrices defined as:

$$\mathbf{J} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \; \mathbf{R} = \begin{bmatrix} 0 & 0 \\ 0 & -d_1 \end{bmatrix} \tag{11}$$

Now the matrix $\mathbf{J} - \mathbf{R}$ can be embeded into the HNN architecture, leading to the NHNN framework as shown in Fig. 2. The NHNN has three essential modules. The first module approximates the parameterized Hamiltonian function H_θ using $(\mathbf{q}, \mathbf{p})$ as inputs. The second module automatically calculates the gradient of the Hamiltonian ∇H. The third module estimates the gradients of the generalized positions and momentum $(\hat{\dot{q}}, \hat{\dot{p}})$ using ∇H and elements of the matrix $\mathbf{J} - \mathbf{R}$. Note that this matrix includes the damping coefficient d_1 that needs to be learned. The last module can be represented in matrix form as:

$$\begin{bmatrix} \hat{\dot{q}} \\ \hat{\dot{p}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -d_1 \end{bmatrix} \begin{bmatrix} \frac{\partial H_\theta}{\partial q} \\ \frac{\partial H_\theta}{\partial p} \end{bmatrix} \tag{12}$$

The parameters of the approximate Hamiltonian function θ and damping coefficient d_1 will be iteratively updated by performing stochastic gradient descent with the following loss function:

$$L = \|\hat{\dot{q}} - \dot{q}\|_2^2 + \|\hat{\dot{p}} - \dot{p}\|_2^2 \tag{13}$$

Note that this loss function is shared among all three neural network architectures.

III. Numerical Studies

In this section, a SMIB system is set up to evaluate the system trajectory and parameter estimation performance of the proposed and baseline machine learning algorithms. Both the training and testing results of the baseline neural network, HNN and NHNN will be demonstrated in detail.

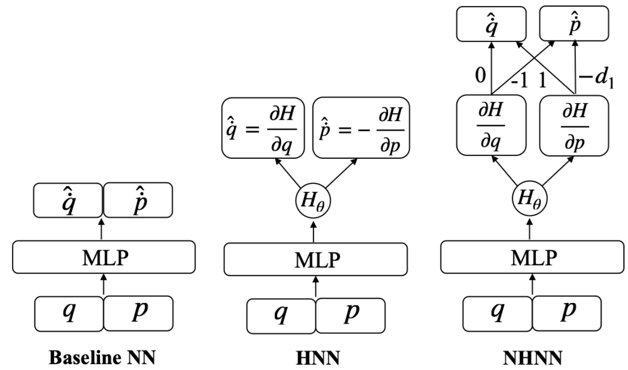

Fig. 2. Architecture of baseline neural network, HNN and NHNN

A. Set up of the SMIB Dynamic System

In this subsection, we set up the simulations for the SMIB dynamic system shown in Fig. 1. The full fault sequence will be modeled, which can be broken down into three stages [11].

Stage 1: The power system is in the prefault steady state condition with state variables (q_0, p_0);

Stage 2: A sudden fault occurs at $t = 0$ on the transmission line and the fault is severe enough to cause the circuit breakers (CBs) to open. In this stage, the system model (5) reduces to:

$$m_1\ddot{\delta} + d_1\dot{\delta} - P_1 = 0 \tag{14}$$

Stage 3: CBs reclose at time $t = \tau$. Note that at the end of stage 2, the state variables are changed to (q_τ, p_τ).

In the case study, it is assumed that the inertia coefficient $m_1 = 0.4$, the mechanical power $P_1 = 0.1$ p.u., the voltage magnitudes V_1 and V_2 are equal to 1 p.u., and $B_{12} = 0.2$ p.u. The above mentioned basic model parameters are assumed to be fixed throughout the numerical studies. On the other hand, the damping coefficient d_1 varies in different dynamic simulations and its value will be chosen from the following set $\{0.00, 0.01, 0.02, 0.05, 0.10\}$. Note that p_0 is always equal to 0 because the momentum is zero in equilibrium states. When the initial position q_0 in stage 1 and the fault duration τ of stage 2 are chosen, different system trajectories in both stage 2 and 3 are obtained. Fig. 3 illustrates the comparison between different values of q_0, τ and d_1. During stage 2, the state variables move further away from the equilibrium point. With certain fault duration and damping coefficient, the system gradually shifts back to a steady state condition.

For each damping coefficient, 40 transient trajectories with time step length of $\frac{1}{10}s$ and time interval $[0, 10s]$ are generated by choosing different q_0 and τ. q_0 has a uniform distribution on the interval $[0, 1]$ and τ follows a uniform distribution on the interval $[0.1s, 2.1s]$. Note that the SMIB system is transient stable under these settings. The proposed and baseline neural networks are trained to learn the system dynamics in stage 3, which is characterized by (5). The 40 transient trajectories for each damping coefficient in stage 3 is generated by performing dynamic simulations with different initial values of (q_τ, p_τ).

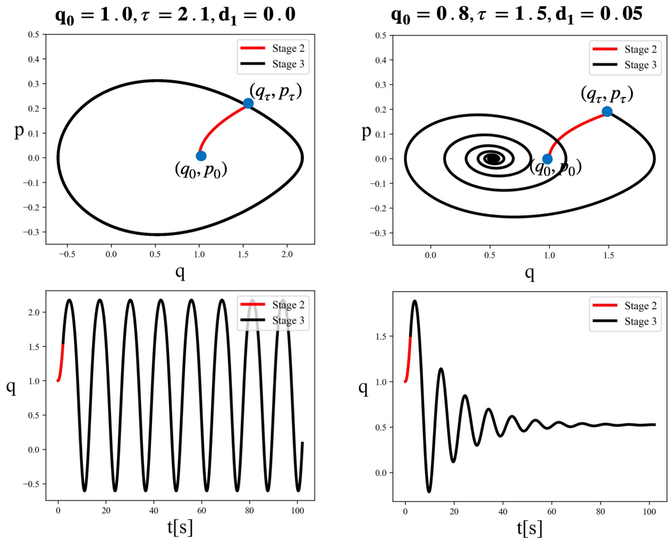

Fig. 3. Sample trajectories of (q, p) given different values of q_0, τ and d_1

The dynamic simulations' solver is Runge-Kutta with relative tolerance equal to 10^{-10}.

Next, measurement noise are superimposed on the generated trajectories. Is it assumed that the measurement noises on the position and momentum coordinates (q, p) follow Gaussian distributions: $e_q \sim N(0, 0.05)$ and $e_p \sim N(0, 0.01)$. Note that in practice, q usually cannot be directly measured and must be computed based on the measurements from PMUs.

B. Set up and Training of the Proposed and Baseline Machine Learning Algorithms

The proposed NHNN, the Baseline NN and the HNN share the same set of hyper-parameters. The input and output layers all have 2 neurons. There is 1 hidden layer with 200 neurons. The activation function is $\tanh$. The full batch training approach is taken. The model parameters θ of NHNN, HNN, and the Baseline NN are initialized as an orthogonal matrix. d_1 of the NHNN is initialized as 0. Automatic differential package "torch.autograd" is used to calculate the gradient of Hamiltonian H in the training process and Adam optimizer [12] is used to update the model parameters.

The total number of training iterations is selected to be 1000. The training losses of all models quickly reduces as the iteration number increases. As shown in Table I, when the damping coefficient is less than 0.1, the final training loss of both NHNN and HNN are much lower than that of the Baseline NN. With a large damping coefficient, the NHNN's loss function is significantly smaller than that of the HNN.

C. Trajectory and Parameter Estimation Results

Once the parameters of the baseline NN, HNN and NHNN models are trained, they can be used estimate $(\dot{q}, \dot{p})$ at any given coordinate (q, p). With the ability to estimate $(\dot{q}, \dot{p})$, all three neural networks can be used to solve the initial value problems directly for the post-disturbance system (5).

TABLE I
THE TRAINING LOSS OF THE BASELINE NN, HNN AND NHNN

Damping Coefficient	Final training loss (10^{-4})		
	Baseline NN	**HNN**	**NHNN**
$d_1 = 0.00$	4.37	3.39	**3.39**
$d_1 = 0.01$	4.25	3.79	**3.37**
$d_1 = 0.02$	4.12	3.46	**3.35**
$d_1 = 0.05$	5.45	3.85	**3.39**
$d_1 = 0.10$	3.52	4.42	**3.46**

The post-disturbance testing trajectory that the proposed and baseline neural network try to estimate are generated as follows. First, the initial post-fault state (q_τ, p_τ) is generated from (q_0, p_0) following the full fault sequence as described in Section III-A. p_0 is set to be 0. q_0 and τ are randomly sampled from the same uniform distribution described in III-A. Then, we use the fourth-order Runge-Kutta integrator to generate the sample trajectory in stage 3. Note that the neural network-based trajectory estimation only uses the initial post-fault state coordinates (q_τ, p_τ) as the inputs.

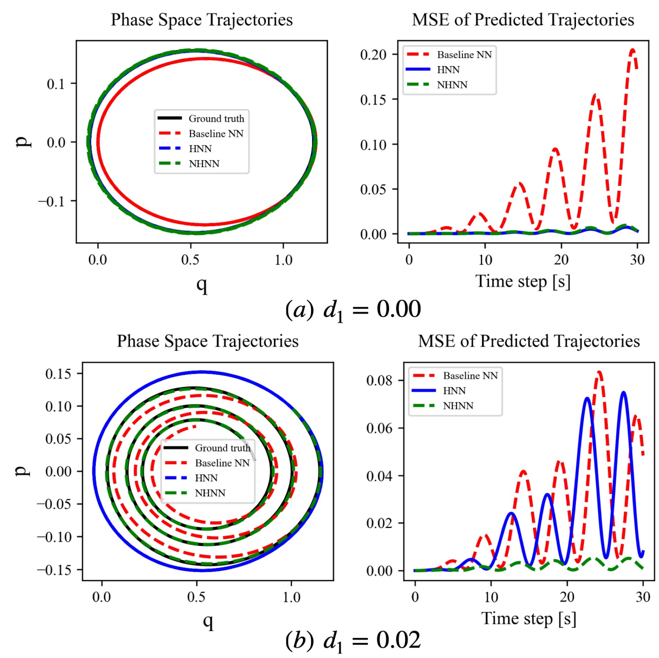

Fig. 4. Ground truth and predicted trajectories with mean squared errors

The ground truth and predicted transient trajectories together with mean squared error (MSE) for the predicted trajectories of all three neural networks can be found in Fig. 4. In the case of $d_1 = 0$, both HNN and NHNN yield more accurate prediction of transient trajectory than that of the baseline neural network. When damping effects are modeled (e.g., $d_1 = 0.02$), the predicted trajectory by NHNN model almost completely coincide with the ground truth trajectory, while the trajectory predicted by HNN and Baseline NN diverge from the ground truth. The ground truth and estimated total energy of the SMIB system at different time stamps are also shown Fig. 5. It can be seen that the proposed NHNN model outperforms the baseline NN and HNN in capturing the change in the total system energy yielding a lower MSE.

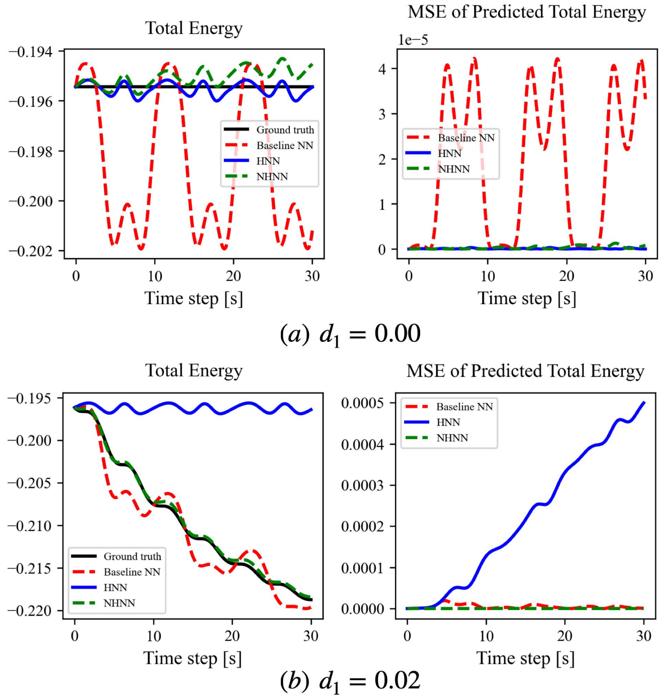

Fig. 5. Ground truth and predicted total energy of the SMIB system

The proposed NHNN model can also estimate the damping coefficient d_1 of the generator. The estimation results are shown in Table II. It can be seen that the damping coefficient estimation of the NHNN model is extremely accurate with absolute percentage error less than 1% in all five scenarios. Note that the performance of the proposed NHNN is not sensitive to the change in sampling rate of the sensors such as PMUs. Further increasing sampling frequency from 10 to 30 Hz has negligible impacts on reducing the trajectory and parameter estimation error. It is also discovered that the length of the training trajectories need to cover at least a full cycle of voltage angle fluctuation to yield reasonable prediction results.

TABLE II
ESTIMATION RESULT OF THE DAMPING COEFFICIENT

d_1	**Estimated $\hat{d}_1$**	**Absolute Percentage Error (%)**
0.00	-0.0004	0.04
0.01	0.0095	0.05
0.02	0.0193	0.07
0.05	0.0481	0.19
0.10	0.0946	0.44

IV. CONCLUSION

In this paper, an innovative NHNN is proposed to learn power system dynamics. By embedding the energy conservation law in the neural network architecture, the NHNN is capable of accurately tracking the total energy change without using the explicit functional form of the differential algebraic equations. Numerical study results on the SMIB system show that the NHNN significantly outperforms the baseline neural network and the HNN in terms of transient trajectories and damping coefficient predictions for both conservative and dissipative scenarios. In the future, we plan to further extend the proposed NHNN to learn the dynamics of larger power systems, which have more complicated differential algebraic equations without analytical solution for the Hamiltonian.

REFERENCES

[1] G. S. Misyris, A. Venzke, and S. Chatzivasileiadis, "Physics-informed neural networks for power systems," in *2020 IEEE Power & Energy Society General Meeting (PESGM)*, IEEE, 2020, pp. 1–5.

[2] X. Kong, K. Yamashita, B. Foggo, and N. Yu, "Dynamic parameter estimation with physics-based neural ordinary differential equations," in *2022 IEEE Power & Energy Society General Meeting (PESGM)*, IEEE, 2022, pp. 1–5.

[3] Y. Zhao, S. You, M. Mandich, *et al.*, "A fast and accurate transient stability assessment method based on deep learning: WECC case study," in *2022 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, 2022, pp. 1–5.

[4] R. Yan, G. Geng, Q. Jiang, and Y. Li, "Fast transient stability batch assessment using cascaded convolutional neural networks," *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 2802–2813, 2019.

[5] L. Zhu, D. J. Hill, and C. Lu, "Hierarchical deep learning machine for power system online transient stability prediction," *IEEE Trans. Power Syst.*, vol. 35, no. 3, pp. 2399–2411, 2020.

[6] T. Zhao, M. Yue, and J. Wang, "Structure-informed graph learning of networked dependencies for online prediction of power system transient dynamics," *IEEE Trans. Power Syst.*, vol. 37, no. 6, pp. 4885–4895, 2022.

[7] W. Cui, W. Yang, and B. Zhang, "Predicting power system dynamics and transients: A frequency domain approach," *arXiv preprint arXiv:2111.01103*, 2021.

[8] S. Greydanus, M. Dzamba, and J. Yosinski, "Hamiltonian neural networks," *Advances in neural information processing systems*, vol. 32, 2019.

[9] V. Vittal, A. Michel, and A. Fouad, "Power system transient stability analysis: Formulation as nearly hamiltonian systems," *Circuits, Systems and Signal Processing*, vol. 3, no. 1, pp. 105–122, 1984.

[10] J. Ma and S. Mei, "Hamiltonian realization of power system dynamic models and its applications," *Science in China Series E: Technological Sciences*, vol. 51, no. 6, pp. 735–750, 2008.

[11] A. Burgen and V. Vittal, *Power System Analysis*, 2nd ed. Tom Robbins, 2000.

[12] P. Kingma Diederik and J. B. Adam, "A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.