Periodic Hamiltonian Neural Networks

Zi-Yu Khoo, Dawen Wu*, Jonathan Sze Choong Low, Stéphane Bressan

Abstract—**Modeling dynamical systems is a core challenge for science and engineering. Hamiltonian neural networks (HNNs) are state-of-the-art models that regress the vector field of a dynamical system under the learning bias of Hamilton's equations. A recent observation is that embedding biases regarding invariances of the Hamiltonian improves regression performance. One such invariance is the periodicity of the Hamiltonian, which improves extrapolation performance. We propose *periodic HNNs* that embed periodicity within HNNs using observational, learning, and inductive biases. An observational bias is embedded by training the HNN on data that reflects the periodicity of the Hamiltonian. A learning bias is embedded through the loss function of the HNN. An inductive bias is embedded by a periodic activation function in the HNN. We evaluate the performance of the proposed models on interpolation and extrapolation problems that either assume knowledge of the periods a priori or learn the periods as parameters. We show that the proposed models can interpolate well but are far more effective than the HNN at extrapolating the Hamiltonian and the vector field for both problems and can even extrapolate the vector field of the chaotic double pendulum Hamiltonian system.**

Impact Statement—**The modeling of many Hamiltonian systems, such as pendulums in physics, predator-prey dynamics in biology, particle motion in solid-state physics, and motion of stars in astronomy, is of scientific interest for the descriptive, predictive, and prescriptive analysis of dynamical systems. Neural networks are universal function approximations that can learn the dynamics of continuous dynamical systems, including Hamiltonian systems. Furthermore, the neural network of a dynamical system can be informed of its invariances to improve its modeling. One invariance that is common in Hamiltonian systems is periodicity. We propose several methods to embed periodicity in neural networks. We compare their performance on interpolating and extrapolating the dynamical system. Our methods can reduce the error in extrapolating the Hamiltonian from 10 to less than 0.1 and the error in extrapolating the vector field from 100 to less than 1.**

Index Terms—**Hamiltonian neural networks, dynamical systems, ordinary differential equations, physics-informed machine learning, extrapolation**

I. Introduction

MODELING dynamical systems is a core challenge for science and engineering. The movement of a pendulum, the wave function of a quantum-mechanical system, the movement of fluids around the wing of a plane, the weather patterns under climate change, and the populations forming an ecosystem are spatio-temporal behaviors of physical phenomena described by dynamical systems. Hamiltonian systems [1] are a class of dynamical systems governed by equations of motion derived from a Hamiltonian function. For autonomous (time-independent) Hamiltonian functions, these equations of motion imply the conservation of the Hamiltonian function value along an orbit of the system.

Physics-informed machine learning has emerged as a field that embeds knowledge of the physical laws governing the system in the learning process to solve complex problems [2], including Hamiltonian dynamical systems. Recent advancements in the modeling of Hamiltonian systems include Hamiltonian neural networks (HNNs) [3], [4], [5], port-HNNs [6], reservoir computing [7], and Hamiltonian Kolmogorov-Arnold Networks [8]. Some works also leverage Hamiltonian invariances [9], [10] by using a Leapfrog integrator or requiring the Hamiltonian to be mechanical [11] or additively separable [12]. The periodicity of a dynamical system is also an invariance. Dierkes et al. take advantage of the periodicity of the Hamiltonian to improve HNN performance [13]. Liu et al. take a more generalist approach and present an activation function to model periodic functions [14]. HNNs [4], [5] use a learning bias [15] based on physics information regarding Hamilton's equations [4], [5] to aid the neural network in converging towards solutions that adhere to physics laws [15]. They supervised-ly interpolate the vector field and the Hamiltonian of a dynamical system from discrete observations of its state space or vector field and outperform their physics-uninformed counterparts in doing so [16]. Generalizing with neural networks by extrapolation is often impossible for non-linear tasks, and only sometimes possible if the appropriate non-linearities are encoded in the architecture and input representation [17].

The main contributions of this work are:

- We propose periodic HNNs that embed periodicity using three modes of biases: observational bias, learning bias, and inductive bias.
- An observational bias is embedded by training the HNN on newly generated data that embody periodicity. A learning bias is embedded through the loss function of the HNN. An inductive bias is embedded by the activation function in the HNN. Each proposed periodic HNN may embed one or more biases.
- The proposed periodic HNNs show high performance in four tasks. The four tasks are the interpolation and extrapolation of the Hamiltonians and vector fields of periodic Hamiltonian systems, with or without the knowledge of their periods. The code for the HNNs is available at github.com/zykhoo/periodicNNs.

Date of current version 14 December 2024.

Zi-Yu Khoo is with the National University of Singapore School of Computing, Singapore 117417, Singapore (e-mail: ziyu@u.nus.edu).

Dawen Wu is with the National University of Singapore School of Computing, Singapore 117417, Singapore (Corresponding author, e-mail: dawen.wu@cnrsatcreate.sg) and CNRS@CREATE, Singapore 117417, Singapore.

Jonathan Sze Choong Low is with the Singapore Institute of Manufacturing Technology (SIMTech), Singapore 636732, Singapore (email sclow@simtech.a-star.edu.sg).

Stéphane Bressan is with the National University of Singapore School of Computing, Singapore 117417, Singapore (e-mail: steph@nus.edu.sg) and CNRS@CREATE, Singapore 117417, Singapore.

In this paper, Sec. II presents the necessary background on Hamiltonian and periodic Hamiltonian systems. Sec. III introduces the proposed periodic HNNs. Sec. IV compares the proposed HNNs against the baseline HNN in regressing periodic Hamiltonians and vector fields. Sec. V concludes the paper and discusses future work.

II. Background

A. Hamiltonian Systems

A Hamiltonian system is characterized by a Hamiltonian function, $H : \mathbb{R}^{2n} \to \mathbb{R}$ where n refers to the dimensionality of the Hamiltonian system. The input states to the Hamiltonian function are time-dependent functions $q : \mathbb{R} \to \mathbb{R}^n$ and $p : \mathbb{R} \to \mathbb{R}^n$, conventionally interpreted as position and momentum. The time derivatives $\dot{q} := \frac{dq}{dt}$ and $\dot{p} := \frac{dp}{dt}$ are governed by the following differential equations, called Hamilton's equations:

$$\dot{q} = \frac{\partial H}{\partial p}, \quad \dot{p} = -\frac{\partial H}{\partial q}. \tag{1}$$

The vector field is the set of time derivatives for all states.

Eq. (1) implies that the Hamiltonian is conserved along trajectories as

$$\frac{dH}{dt} = \frac{\partial H}{\partial q} \cdot \dot{q} + \frac{\partial H}{\partial p} \cdot \dot{p} = 0. \tag{2}$$

Let $z = (q^T, p^T)^T \in \mathbb{R}^{2\times n}$. For notational simplicity, we omit the superscripts and simplify it to $z = (q, p)$ unless otherwise stated. Let $\boldsymbol{J}$ be the $2n \times 2n$ matrix

$$\boldsymbol{J} = \begin{bmatrix} \mathbf{0} & \boldsymbol{I} \\ -\boldsymbol{I} & \mathbf{0} \end{bmatrix}, \tag{3}$$

where $\boldsymbol{I} \in \mathbb{R}^{n\times n}$ is the identity matrix and $\mathbf{0} \in \mathbb{R}^{n\times n}$ is the zero matrix. Then, Eq. (1) can be written compactly as

$$\dot{z} = \boldsymbol{J} \cdot \nabla_z H, \tag{4}$$

where $\nabla_z H = \frac{\partial H}{\partial z}$.

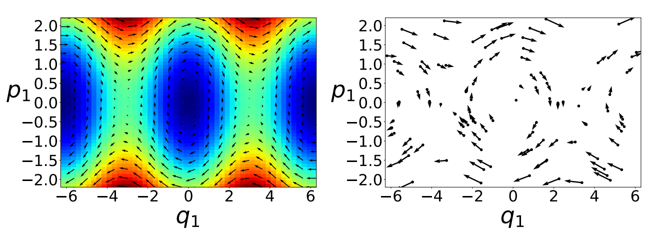

Fig. 1. Left: Non-linear pendulum vector field (black arrows) and Hamiltonian (heatmap). Right: Random samples $(q, p, \dot{q}, \dot{p})$ of the non-linear pendulum vector field.

A classic example of a Hamiltonian system is the non-linear pendulum. Fig. 1 (left) shows the vector field of a non-linear pendulum in its two-dimensional phase space with states position (angle when oscillating in a plane) and momentum (mass multiplied by velocity) on the x-axis and y-axis respectively.

For a non-linear pendulum of unitary mass, the Hamiltonian is the sum of kinetic and potential energy. The heatmap in Fig. 1 (left) shows the value of the Hamiltonian in the phase space. Fig. 1 (right) shows random samples of the same vector field.

B. Periodic Hamiltonian Systems

Definition 1 (Periodicity): A Hamiltonian system is called periodic if the Hamiltonian repeats itself in any of the q or p terms after a certain period. The periodic Hamiltonian is defined

$$H(z) = H(z + \zeta) \tag{5}$$

where $\zeta \in \mathbb{R}^{2\times n}$ is the period of the Hamiltonian in each q and p. The Hamiltonian may be periodic along any input q or p, and a periodic Hamiltonian has at least one non-zero element in ζ. For notational simplicity, we use ζ_i to denote the period of the i^{th} state.

Knowledge regarding periodicity may be embedded within machine learning models as biases. Karniadakis et al. present three modes of biasing a regression model: observational bias, learning bias, and inductive bias [15]. Observational biases are introduced directly through data that embody the underlying physics, or carefully crafted data augmentation procedures. With sufficient data to cover the input domain of a regression task, machine learning methods have demonstrated remarkable power in achieving accurate interpolation between the dots [15]. Learning biases are soft constraints introduced by appropriate loss functions, constraints and inference algorithms that modulate the training phase of a machine learning model to explicitly favor convergence towards solutions that adhere to the underlying physics [15], [4], [5]. Inductive biases are prior assumptions incorporated by tailored interventions to a machine learning model architecture, so regressions are guaranteed to implicitly and strictly satisfy a set of given physical laws [15].

Generally, and for completeness, four distinct periodicity-related learning tasks exist [18]. The first task involves interpolation with the knowledge of the periods a priori. The second task involves interpolation without the knowledge of the periods a priori, where the periods are learned parameters. The third task involves extrapolation with the knowledge of the periods a priori. The fourth task involves extrapolation without the knowledge of the periods a priori, where the periods are learned parameters. We evaluate the baseline and proposed models on all four tasks for completeness. However, we place emphasis on the third and fourth tasks, which are for extrapolation.

For illustration and comparative empirical evaluation of the HNNs presented, we consider six periodic Hamiltonian systems in Table I. The non-linear pendulum system is a classic example of a Hamiltonian system. In practical Hamiltonian systems in Physics, only inputs corresponding to the position, q, are periodic. To illustrate the potential applications of periodic HNNs, we consider their performance on an array of abstract Hamiltonian systems. Systems 1 to 4 are abstract Hamiltonian systems, created to demonstrate the performance of the baseline HNN and proposed periodic HNNs in predicting a range of Hamiltonian dynamics comprising different functions, different dimensionalities, and periodicity along various inputs, representative of potential practical Hamiltonian systems. System 4 is not periodic but has trigonometric components. It is used to demonstrate the performance of

TABLE I
EQUATIONS, PERIOD ($\zeta_{q_1}, \zeta_{q_2}, \zeta_{p_1}, \zeta_{p_2}$) AND NUMBER OF DIMENSIONS n IN EACH STATE OF EACH HAMILTONIAN SYSTEM.

System	Equation	ζ_{q_1}	ζ_{q_2}	ζ_{p_1}	ζ_{p_2}	n
Non-linear Pendulum	$H(q_1, p_1) = \frac{p_1^2}{2} + 1 - \cos q_1$	2π	-	0	-	1
System 1	$H(q_1, p_1) = \frac{p_1^2}{2} + 1 - (\sin^2 q_1)$	π	-	0	-	1
System 2	$H(q_1, q_2, p_1, p_2) = \frac{p_1^2+p_2^2}{2} + 1 - \sin(q_1 + q_2)$	2π	2π	0	0	2
System 3	$H(q_1, p_1) = \sin q_1 - \cos p_1$	2π	-	2π	-	1
System 4	$H(q_1, p_1) = \sin q_1 \times \cos p_1 + 2 \times q_1 + 2 \times p_1$	0	-	0	-	1
Double Pendulum	$H(q_1, q_2, p_1, p_2) = \frac{\frac{p_1^2+2\times p_2^2}{2} - 2\times p_1 \times p_2 \times \cos(q_1-q_2)}{2\times(1+\sin^2(q_1-q_2))} - 2\times\cos(q_1) - \cos(q_2)$	2π	2π	0	0	2

the baseline HNN and proposed periodic HNNs in predicting such non-periodic Hamiltonian dynamics. Lastly, the double pendulum system is a classic example of a Hamiltonian system with rich and chaotic dynamics [19]. It is strongly sensitive to initial conditions. The non-linear pendulum system and the double pendulum system are practical examples of Hamiltonian systems. They are shown in Fig. 2.

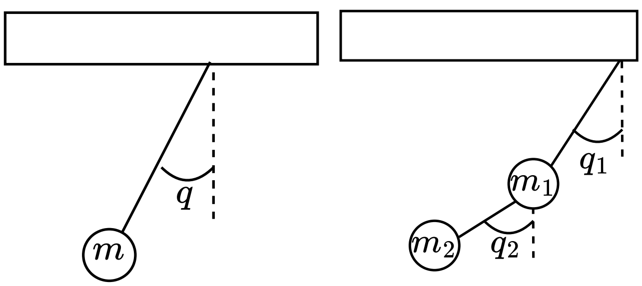

Fig. 2. Left: The non-linear pendulum system. q indicates the position of the pendulum, with mass m. Right: The double pendulum system. q_1 and q_2 indicate the position of the linked pendulum bobs, with mass m_1 and m_2. In Table I, $m = 1$, $m_1 = 1$, and $m_2 = 1$.

III. METHODOLOGY

A. The HNN Baseline

The baseline HNN (leftmost, Fig. 3), proposed in [4], [5], is described as follows,

$$\hat{H}(q, p; w) \approx H(q, p), \tag{6}$$

where (q, p) is an input state. $\hat{H}$ is a neural network with trainable parameters w. $\hat{H}(q, p; w)$ and $H(q, p)$ represent the HNN predicted Hamiltonian value and the true Hamiltonian value, respectively.

The loss function of the baseline HNN is

$$\ell \ = \textstyle\sum_{k=0}^{2} c_k \ell_k, \tag{7}$$

where $c_k \in \mathbb{R}$ is the coefficient of each ℓ_k, and

$$\ell_0 \ = \ \left(\hat{H}(q_0, p_0; w) - H_0\right)^2, \tag{8}$$

where ℓ_0 is an arbitrary pinning term that "pins" the regressed Hamiltonian at (q_0, p_0) to one solution, $H_0 \in \mathbb{R}$, among several solutions that are modulo and additive constant and reduces the search space for convergence. ℓ_1 and ℓ_2 are given as:

$$\ell_1 \ = \ \left\|\dot{\hat{q}} - \dot{q}\right\|^2, \tag{9}$$

$$\ell_2 \ = \ \left\|\dot{\hat{p}} - \dot{p}\right\|^2, \tag{10}$$

where $\dot{\hat{q}}$ and $\dot{\hat{p}}$ are the predicted time derivatives of the HNN:

$$\dot{\hat{q}} = \frac{\partial \hat{H}}{\partial p}, \quad \dot{\hat{p}} = -\frac{\partial \hat{H}}{\partial q}, \tag{11}$$

which can be computed by automatic differentiation using tools such as `Pytorch` [20] or `JAX` [21] to compute the $\dot{\hat{q}}$ and $\dot{\hat{p}}$. ℓ_1 and ℓ_2 are Hamilton's equations corresponding to (1). ℓ_0, ℓ_1, and ℓ_2 introduce biases that favor the convergence of the HNN toward the underlying physics of the regressed Hamiltonian. (7) defines the loss function of the HNN as a linear combination of (8) to (10). One can assume that $c_k = 1$ although additional knowledge of the system can be used to emphasize any ℓ_k [4].

The periodic HNNs introduced in Sec. III-B, III-C, III-D and III-E adopt the Hamiltonian learning bias of the HNN and further embed biases regarding periodicity. They perform the task of regressing, namely interpolating and extrapolating the Hamiltonian and vector fields in the same way. They empirically highlight the benefits of the proposed periodicity bias when combined with a Hamiltonian bias, over the benefits of solely the Hamiltonian bias, in the HNN.

B. The HNN with Observational Bias (HNN-O)

We inform the baseline HNN of periodicity via observational bias. We triple the quantity of the input data. Specifically, given a sample of instantaneous states (q, p), the sinusoidal, co-sinusoidal, and identity functions are used to triple the quantity of data. The transformed input is $(\sin\left(\frac{2\pi}{\zeta_q} \times q\right), \cos\left(\frac{2\pi}{\zeta_q} \times q\right), q, \sin\left(\frac{2\pi}{\zeta_p} \times p\right), \cos\left(\frac{2\pi}{\zeta_p} \times p\right), p)$. This is a data augmentation, as shown in Figure 3. The augmentation is performed directly and conveniently in the input layer of the HNN-O. The dotted lines indicate no trainable weights.

When the period of the Hamiltonian along each input is known, $\frac{2\pi}{\zeta_q}$ and $\frac{2\pi}{\zeta_p}$ are computed from the known ζ_q and ζ_p. When the period of the Hamiltonian along each input is unknown, $\frac{2\pi}{\zeta_q}$ and $\frac{2\pi}{\zeta_p}$ are trainable parameters of the HNN-O.

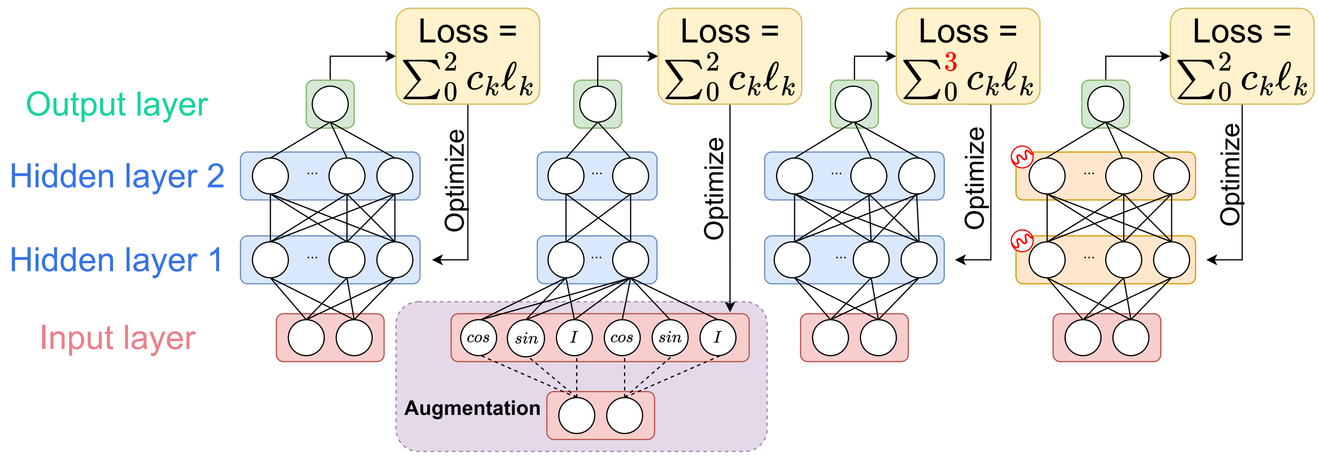

Fig. 3. Leftmost: Architecture of the baseline HNN. Second from left: Proposed periodic HNN with observational bias. Second from right: Proposed periodic HNN with learning bias. Rightmost: Proposed periodic HNN with inductive bias.

C. The HNN with Learning Bias (HNN-L)

We inform the baseline HNN of periodicity via its loss function. For a periodic Hamiltonian given in (5), the derivatives of a periodic Hamiltonian are:

$$\frac{\partial H}{\partial p}(q,p) = \frac{\partial H}{\partial p}(q+\zeta_q, p+\zeta_p) = \dot{q}(q,p), \tag{12}$$

$$-\frac{\partial H}{\partial q}(q,p) = -\frac{\partial H}{\partial q}(q+\zeta_q, p+\zeta_p) = \dot{p}(q,p). \tag{13}$$

where $\zeta_q \in \mathbb{R}^n$, $\zeta_p \in \mathbb{R}^n$, and $(\zeta_q^T, \zeta_p^T)^T = \zeta$. The following equations, based on (12) and (13), are placed in the loss function:

$$\begin{aligned} \ell_3 \quad &= \quad \left\| \frac{\partial H}{\partial p}(q \pm \zeta_q, p \pm \zeta_p; w) - \dot{q}(q,p) \right\|^2 \\ &+ \quad \left\| -\frac{\partial H}{\partial q}(q \pm \zeta_q, p \pm \zeta_p; w) - \dot{p}(q,p) \right\|^2. \end{aligned} \tag{14}$$

$$\ell_{period} \quad = \textstyle\sum_{k=0}^{3} c_k \ell_k, \tag{15}$$

where $\dot{q}(q,p)$ and $\dot{p}(q,p)$ are the time derivatives at the state (q,p). If the input state is not periodic, its element in ζ is set to zero. (14) simplifies to (9) and (10). ℓ_{period} is the loss function of the periodic HNN with learning bias. It is a linear combination of equations ℓ_0 to ℓ_3 from (7) and (14).

We note that (5) necessarily but not sufficiently implies (12) and (13). Therefore, (12) and (13) do not sufficiently imply periodicity. However, they are preferably placed in the loss function over (5). This is because they incorporate periodicity well in practice, as $\dot{q}$ and $\dot{p}$ from (12) and (13) respectively are already used to train the HNN, but H is not. The resulting HNN-L with learning bias (second from right, Fig. 3) favors convergence towards an HNN that is periodic. When the period of the Hamiltonian along each input is known, ℓ_3 is computed from the known ζ_q and ζ_p. When the period of the Hamiltonian along each input is unknown, ℓ_3 is computed using ζ_q and ζ_p, which are trainable parameters of the HNN-L.

D. The HNN with Inductive Bias (HNN-I)

We inform the baseline HNN of periodicity via its architecture. A periodic activation function is used. The proposed HNN-I (rightmost, Fig. 3) is described as follows,

$$\sigma(x;a) = x + \frac{1}{a}\sin^2(ax), \tag{16}$$

where $a = \frac{2\pi}{min(\zeta)}$ corresponds to the period of the states of the Hamiltonian. When the period of the Hamiltonian along each input is known, a is computed from the known periods. When the period of the Hamiltonian along each input is unknown, a is a trainable parameter of the HNN-I.

E. The HNN with Multiple Biases

As seen from Sec. III-B, III-C and III-D, different biases can be individually embedded into a HNN. The biases can be further combined to constrain a system to be periodic. Multiple biases regarding the periodicity of a Hamiltonian to be regressed can be embedded into a neural network simultaneously.

Pairwise combinations of the observational, learning, and inductive biases are first considered. There are three pairwise combinations of biases. The HNN-OL combines observational and learning biases, resulting in a periodic HNN with periodic training data and a loss function given by (15). The HNN-LI combines the learning and inductive biases, resulting in a periodic HNN with a loss function given by (15) and a periodic activation function. The third HNN-OI combines observational and inductive biases, resulting in a periodic HNN with periodic training data and a periodic activation function.

Lastly, the HNN-OLI combines observational, learning, and inductive biases, resulting in a periodic HNN with periodic training data, a loss function given by (15), and a periodic activation function.

IV. Experiments

The task is to interpolate or extrapolate the Hamiltonian following (6) and the vector field following (11). We compare the performance of the baseline HNN and the proposed periodic HNNs on the task, for the six periodic Hamiltonian systems shown in Table I.

We use the following two evaluation metrics.

- For the interpolation and extrapolation of the Hamiltonian:

$$E_H = \frac{1}{K}\sum_{k=1}^{K}\left(\hat{H}(q_k, p_k; w) - H(q_k, p_k)\right)^2, \quad (17)$$

  where $\hat{H}(q_k, p_k; w)$ is the predicted Hamiltonian by the HNN, $H(q_k, p_k)$ is the true Hamiltonian, and K is the size of a test batch. A smaller E_H is better.
- For the interpolation and extrapolation of the vector field:

$$E_V = \frac{1}{K}\sum_{k=1}^{K}\frac{\sqrt{||\dot{\hat{q}}_k - \dot{q}_k||_2^2 + ||\dot{\hat{p}}_k - \dot{p}_k||_2^2}}{\sqrt{||\dot{q}_k||_2^2 + ||\dot{p}_k||_2^2}} \times 100\%, \quad (18)$$

  where $\dot{\hat{q}}_k$ and $\dot{\hat{p}}_k$ are the predicted time derivatives, and $\dot{q}_k$ and $\dot{p}_k$ are the corresponding true derivatives. K is the size of a test batch. A smaller E_V is better.

The data used for the experiments is generated following the recent work of Khoo et al. on Hamiltonian neural networks [12]. Training data comprising 512 samples of $(q, p, \dot{q}, \dot{p})$ are generated uniformly at random within the sampling domain for each Hamiltonian system. The sampling domains are shown in Table II. The sampling domains are the same for the interpolation and extrapolation tasks. 20% of training data is set aside as validation data for validation-based early stopping [22]. For the interpolation task, the test data set comprises 10^{2n} evenly spaced samples of $(q, p, \dot{q}, \dot{p})$ within the sampling domain. For the extrapolation task, the test data set comprises 20^{2n} evenly spaced samples of $(q, p, \dot{q}, \dot{p})$, extended by one period along all periodic inputs.

The general experimental setup for all HNNs in all experiments is as follows. All HNNs are designed with an input layer with width $2 \times n$, two hidden layers with width shown in Table. II, and one output layer with width one. They use a softplus activation. For training, a batch size of 80 is used, with an Adam optimizer [23], with a learning rate shown in Table II. All experiments are repeated for 5 random seeds. All HNNs are trained in `Pytorch`. The complete code and data are available at github.com/zykhoo/periodicNNs.

In all experiments, unless stated otherwise, all HNNs are trained until convergence. In our context, convergence is defined when the validation loss does not decrease for 4,000 epochs. This is to find an optimal bias-variance trade-off.

In experiments where the periods are not known a priori, they are trainable parameters. The period of every input is ignorantly initialized to 2π, even for inputs that may not be periodic. For the HNN-O, the input is multiplied by $\frac{2\pi}{2\pi} = 1$ before transformation by the sinusoidal and co-sinusoidal functions. For the HNN-L, the ζ_q and ζ_p are 2π in the loss function. For the HNN-I, the activation function uses $a = 1$. The periods are updated during training.

Similar to the recent works on HNNs [6], [12], [8], we present only the proposed model and the various ablated HNNs. In [6], Desai et al. presented their model (Port Hamiltonian Neural Networks) and compared it against the HNN and a time-dependent HNN which removes their proposed architecture. In [12], Khoo et al. presented their model (Separable Hamiltonian Neural Networks) and compared it against the HNN and various separable HNNs with individual biases for additive separability. In [8], Shukla et al. compared their proposed model (Hamiltonian Chebyshev-KAN) against only the HNN. Our work is most similar to [12], therefore, we decided to present our model (periodic Hamiltonian neural network) and compare it against the HNN and various periodic HNNs with individual biases for periodicity.

Sections IV-A and IV-B present the results for interpolating the Hamiltonian and vector field with and without a priori knowledge of the periods respectively. Sections IV-C and IV-D present the results for extrapolating the Hamiltonian and vector field with and without a priori knowledge of the periods respectively.

A. Interpolating with Knowledge of the Period

This subsection focuses on the interpolation task with a priori knowledge of the period. We report E_H, E_V, and training time taken until convergence in seconds for all the HNNs in Fig. 4. Generally, from Figure 4, the HNN-OL is the HNN of choice in interpolating the vector field and Hamiltonian. It consistently performs well on all systems.

It is also observed that the baseline HNN performs well, and individual biases are insufficient to outperform the baseline. However, combination biases, especially in the HNN-OL are synergistic. The HNN-OL consistently outperforms the baseline HNN in interpolating the vector field and Hamiltonian. Furthermore, the HNN-OL requires a relatively lower training time for most systems except the double pendulum and System 2. The biases of the HNN-OL enable the loss to fall quickly during training. In summary, the HNN-OL is the model of choice for interpolating Hamiltonians with a priori knowledge of their periods.

B. Interpolating without Knowledge of the Period

This subsection focuses on the interpolation task without a priori knowledge of the period. We report E_H, E_V, and training time taken until convergence for all the HNNs in Fig. 5. Generally, from Figure 5, the HNN-OL and HNN-IO are the HNN of choice in interpolating the vector field and Hamiltonian.

It is also observed that the baseline HNN performs well, and individual biases are insufficient to outperform the baseline, although combination biases are synergistic. However, the HNN-LI that combines inductive and learning biases performs extremely poorly as the inductive bias hinders the learning of the period along each input. Both the HNN-OL and HNN-IO have low E_H and E_V and require less time to train. They are the models of choice for interpolating Hamiltonians with a priori knowledge of their periods.

Based on Sections IV-A and IV-B, the HNN-OL is the HNN of choice in interpolating the vector field and Hamiltonian.

TABLE II

CONFIGURATION OF EXPERIMENTS. Ω_{q_1}, Ω_{q_2}, Ω_{p_1}, AND Ω_{p_2} ARE THE DOMAINS OF q AND p RESPECTIVELY. THE LAST FOUR COLUMNS SHOW THE WIDTH AND TOTAL PARAMETERS FOR THE VARIOUS HNNS FOR EXPERIMENTS WHERE THE PERIODS ARE KNOWN. IN EXPERIMENTS WHERE THE PERIODS ARE UNKNOWN, THEY ARE ADDITIONAL TRAINABLE PARAMETERS. FOR ALL EXPERIMENTS, BY DESIGN, THE NUMBER OF PARAMETERS FOR THE HNN-O IS SLIGHTLY LESS THAN THE NUMBER OF PARAMETERS FOR THE BASELINE HNN, HNN-L, AND HNN-I.

Experiment	Ω_{q_1}	Ω_{q_2}	Ω_{p_1}	Ω_{p_2}	Learning rate	HNN, HNN-L, HNN-I Width	Params	HNN-O Width	Params
Non-linear Pendulum	$[-2\pi, 2\pi]$	-	$[-1.2, 1.2]$	-	0.001	16	337	11	332
System 1	$[-2\pi, 2\pi]$	-	$[-1.2, 1.2]$	-	0.001	16	337	14	323
System 2	$[-2\pi, 2\pi]$	$[-2\pi, 2\pi]$	$[-1.2, 1.2]$	$[-1.2, 1.2]$	0.001	32	1249	28	1205
System 3	$[-2\pi, 2\pi]$	-	$[-1.2, 1.2]$	-	0.001	16	337	14	323
System 4	$[-2\pi, 2\pi]$	-	$[-1.2, 1.2]$	-	0.001	16	337	14	323
Double Pendulum	$[-2\pi, 2\pi]$	$[-2\pi, 2\pi]$	$[-1.2, 1.2]$	$[-1.2, 1.2]$	0.001	32	1249	28	1205

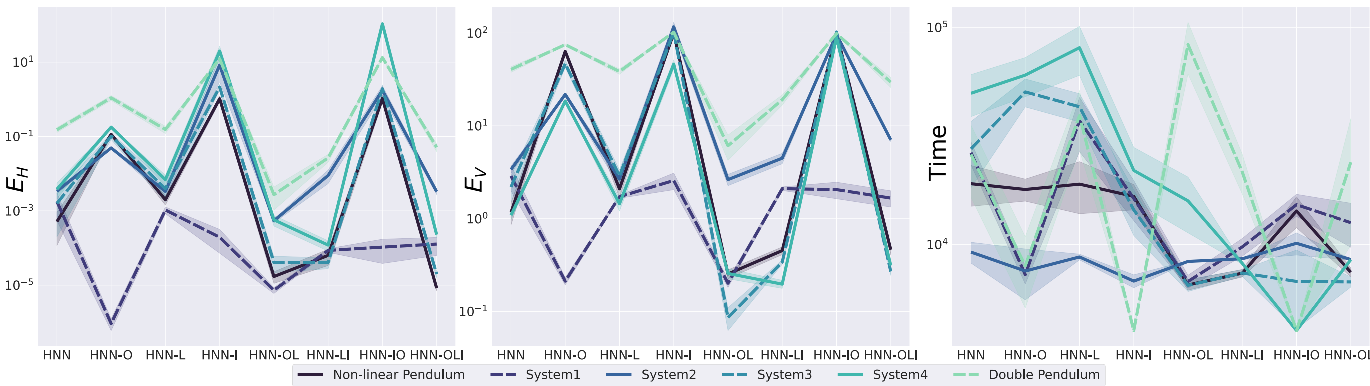

Fig. 4. Performance on the interpolation task with knowledge of the periods for the baseline HNN, the 3 proposed periodic HNNs with one bias each (HNN-O, HNN-L, HNN-I), the 3 proposed periodic HNNs with two biases each (HNN-OL, HNN-LI, HNN-IO) and the proposed periodic HNN with all biases (HNN-OLI). Left: E_H and standard error. Center: E_V and standard error. Right: Time taken until convergence in seconds and standard error. All graphs are logarithm-scaled for ease of visualization.

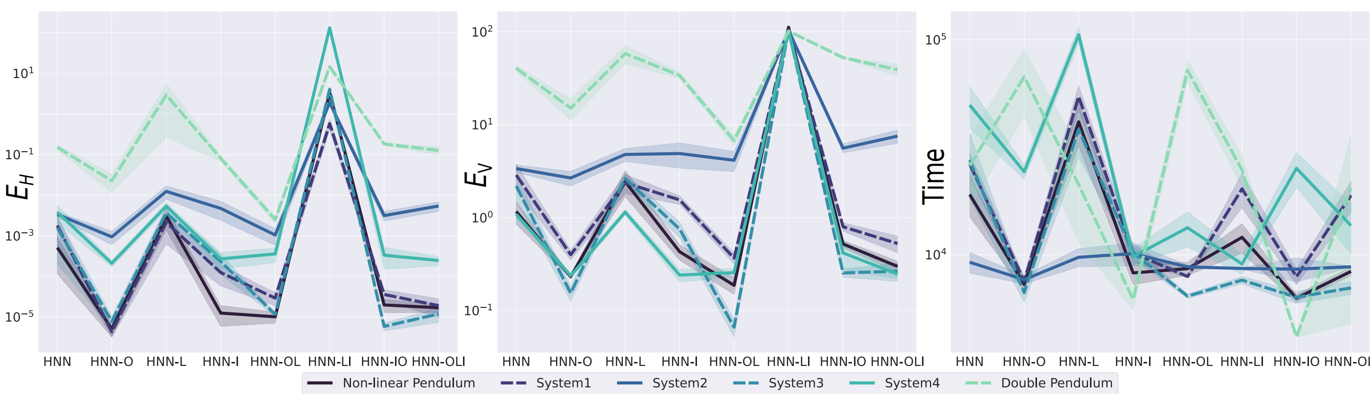

Fig. 5. Performance on the interpolation task without knowledge of the periods for the baseline HNN, the 3 proposed periodic HNNs with one bias each (HNN-O, HNN-L, HNN-I), the 3 proposed periodic HNNs with two biases each (HNN-OL, HNN-LI, HNN-IO) and the proposed periodic HNN with all biases (HNN-OLI). Left: E_H and standard error. Center: E_V and standard error. Right: Time taken until convergence in seconds and standard error.All graphs are logarithm-scaled for ease of visualization.

C. Extrapolating with Knowledge of the Period

This subsection focuses on the extrapolation task with a priori knowledge of the period.

We report E_H, E_V, and training time taken until convergence for all the HNNs in Fig. 6. Generally, the periodic HNNs outperform the baseline. The best-performing model is the HNN-OL, which combines observational and learning biases. The observational and learning biases are synergistic, and extrapolate the vector field and Hamiltonian well for various Hamiltonian systems with more dimensions and multiple periodicities. The HNN-OL outperforms the baseline by a magnitude of 10^6 and 10^2 on extrapolating the Hamiltonian and vector field respectively for the non-linear pendulum. The HNN-OL even performs well for System 4, which is not periodic.

Of notable interest are the results for the double pendulum. The HNN is not able to extrapolate the Hamiltonian and vector field of the double pendulum system, denoted by the black star in the plots for E_H and E_V. It returns `NaN`. Conversely, the HNN-OL performs the best in extrapolating the double pendulum.

Generally, the proposed periodic HNNs converge quickly.

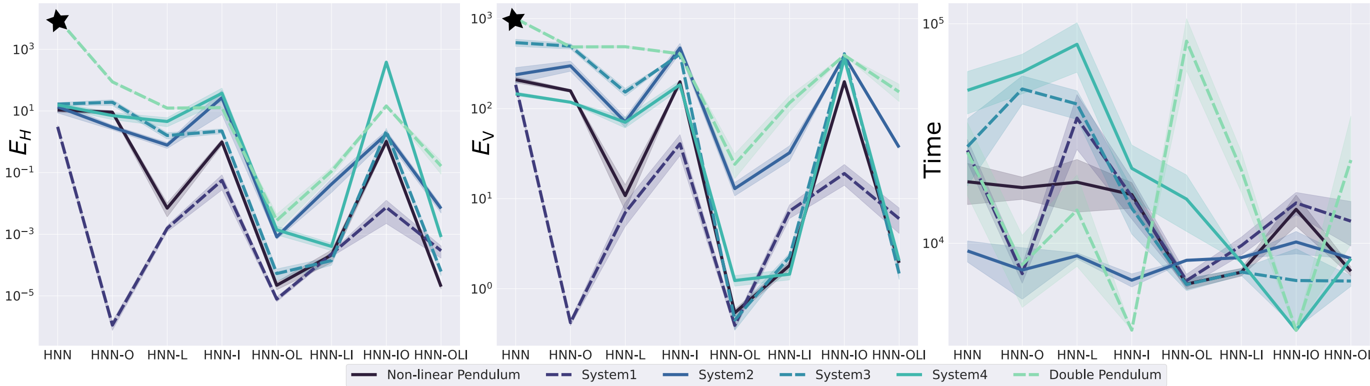

Fig. 6. Performance on the extrapolation task with knowledge of the periods for the baseline HNN, the 3 proposed periodic HNNs with one bias each (HNN-O, HNN-L, HNN-I), the 3 proposed periodic HNNs with two biases each (HNN-OL, HNN-LI, HNN-IO) and the proposed periodic HNN with all biases (HNN-OLI). Left: E_H and standard error. Center: E_V and standard error. Right: Time taken until convergence in seconds and standard error.All graphs are logarithm-scaled for ease of visualization.

However, in the case of the double pendulum, the HNN-OL takes a significantly longer time to converge as compared to the baseline. This is because the loss during training does not plateau like the baseline HNN. Instead, the biases in the HNN-OL allow it to search for a more optimal solution in the solution space. The loss continues to fall over many epochs, and the HNN-OL converges to an optimal and more generalizable solution that is agreeable to extrapolation. As the loss continues to fall over many epochs, the training time of the proposed periodic HNNs is longer.

D. Extrapolating without Knowledge of the Period

This subsection focuses on the extrapolation task without a priori knowledge of the period. We report E_H, E_V, and training time taken until convergence for all the HNNs in Fig. 7. Generally, the periodic HNNs outperform the baseline. The best-performing model is the HNN-OL, which combines observational and learning biases. The observational and learning biases are synergistic, and extrapolate the vector field and Hamiltonian well for various Hamiltonian systems with more dimensions and multiple periodicities. The HNN-OL outperforms the baseline by a magnitude of 10^5 and 10^2 on extrapolating the Hamiltonian and vector field respectively for the non-linear pendulum.

Generally, the observational and inductive biases do not depend much on the knowledge of the periods to extrapolate the Hamiltonian system. The performance of the HNN-O and HNN-I do not worsen significantly, and even improve slightly from the previous experiment with a priori knowledge of the periods. As the periods are trainable parameters, HNN-O and HNN-I have more flexibility in modeling the Hamiltonian system. The HNN-L depends heavily on knowledge of the periods, and sometimes learns the wrong periods during training, resulting in poor extrapolation. Nonetheless, the best-performing model is the HNN-OL. The combination of observational and learning biases helps the HNN-OL to find the correct periods for its loss function and extrapolate better.

It is also observed that while the HNN cannot extrapolate the double pendulum Hamiltonian system, the HNN-OL can significantly improve the extrapolation even without knowledge of its periods.

V. Conclusion

We propose *periodic HNNs* that embed periodicity using three modes of biases: observational bias, learning bias, and inductive bias. An observational bias is embedded by training the HNN on newly generated data that embody periodicity. A learning bias is embedded through the loss function of the HNN. An inductive bias is embedded by the activation function in the HNN. Each proposed periodic HNN may embed one or more biases. We experimentally compare the performance of the proposed periodic HNNs against a baseline HNN on four tasks of interpolating and extrapolating periodic Hamiltonians and their vector fields, with and without a priori knowledge of their periods. The proposed periodic HNNs can outperform the baseline HNN in interpolating and extrapolating vector fields and Hamiltonians from a variety of Hamiltonian systems. In particular, they show high performance in extrapolation by leveraging the periodicity property of the Hamiltonian systems. The HNN-OL is the best as it significantly outperforms the baseline in all tasks. The results apply for a variety of Hamiltonian systems, including the chaotic double pendulum system.

As machine learning develops in the context of traditional science and engineering fields, we highlight several specific areas that are ripe for exploration, including extensions to other properties, improvements in scalability and generalization, and explainability:

- Embedding other physical properties: Other important physical properties such as symmetry, additive separability, multiplicative separability, and continuity can be embedded in neural networks to improve model accuracy and interpretability across a wider range of physical systems.
- Scalability to complex and high-dimensional systems: Although the current approach has been demonstrated on the chaotic double pendulum, other application specific real-world systems may feature higher dimensionality and

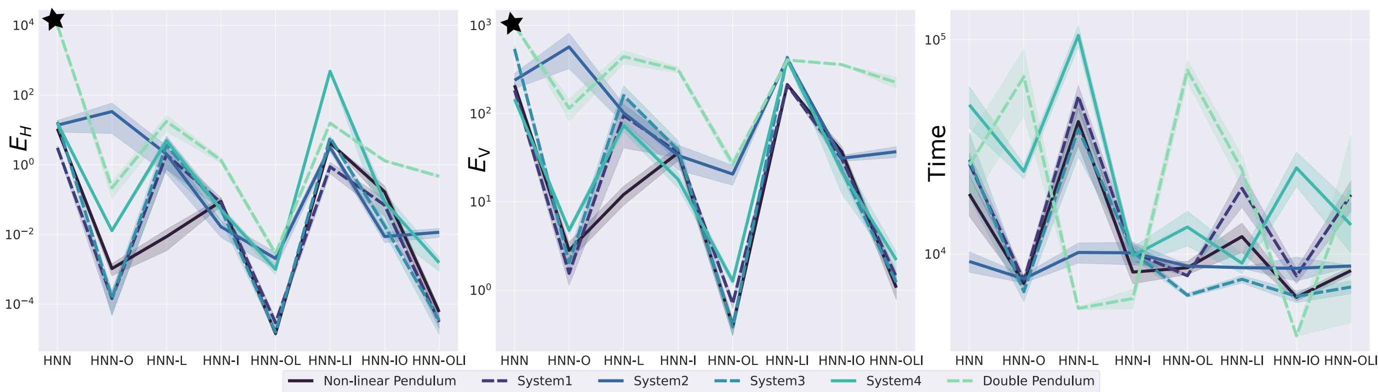

Fig. 7. Performance on the extrapolation task without knowledge of the periods for the baseline HNN, the 3 proposed periodic HNNs with one bias each (HNN-O, HNN-L, HNN-I), the 3 proposed periodic HNNs with two biases each (HNN-OL, HNN-LI, HNN-IO) and the proposed periodic HNN with all biases (HNN-OLI). Left: E_H and standard error. center: E_V and standard error. Right: Time taken until convergence in seconds and standard error. All graphs are logarithm-scaled for ease of visualization.

greater complexity. They open new avenues for future work.

- Generalization beyond periodic systems: Although periodicity is a key property in many Hamiltonian systems, we have demonstrated how periodic HNNs can outperform HNNs in modeling non-periodic systems as seen in System 4 in the experiments. Periodic HNNs may be generalized to handle systems that exhibit quasi-periodicity or are trigonometric.
- Explainability and theoretical insights: In addition to improving performance, a critical direction for future research lies in enhancing the explainability of periodic HNNs. On a surface level, the periodic HNNs learn the period of the Hamiltonian during training if the period is not known. This improves interpretability of the Hamiltonian. However, the comparative nature of this work also enhances the understanding how different biases contribute to the learning of a periodic Hamiltonian, and can shed light on the theoretical underpinnings of the periodic nature of Hamiltonian systems.

References

[1] K. R. Meyer and G. R. Hall, Hamiltonian Differential Equations and the N-Body Problem. New York, NY: Springer New York, 1992, pp. 1–32. [Online]. Available: https://doi.org/10.1007/978-1-4757-4073-8_1

[2] F. Piccialli, M. Raissi, F. A. C. Viana, G. Fortino, H. Lu, and A. Hussain, "Guest editorial: Special issue on physics-informed machine learning," IEEE Transactions on Artificial Intelligence, vol. 5, no. 3, pp. 964–966, 2024.

[3] M. Mattheakis, D. Sondak, A. S. Dogra, and P. Protopapas, "Hamiltonian neural networks for solving equations of motion," Phys. Rev. E, vol. 105, p. 065305, Jun 2022. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.105.065305

[4] T. Bertalan, F. Dietrich, I. Mezić , and I. G. Kevrekidis, "On learning hamiltonian systems from data," Chaos: An Interdisciplinary Journal of Nonlinear Science, vol. 29, no. 12, p. 121107, dec 2019. [Online]. Available: https://doi.org/10.1063%2F1.5128231

[5] S. Greydanus, M. Dzamba, and J. Yosinski, "Hamiltonian neural networks," in Advances in Neural Information Processing Systems, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/26cd8ecadce0d4efd6cc8a8725cbd1f8-Paper.pdf

[6] S. A. Desai, M. Mattheakis, D. Sondak, P. Protopapas, and S. J. Roberts, "Port-hamiltonian neural networks for learning explicit time-dependent dynamical systems," Phys. Rev. E, vol. 104, p. 034312, Sep 2021. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.104.034312

[7] H. Zhang, H. Fan, L. Wang, and X. Wang, "Learning hamiltonian dynamics with reservoir computing," Phys. Rev. E, vol. 104, p. 024205, Aug 2021. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.104.024205

[8] K. Shukla, J. D. Toscano, Z. Wang, Z. Zou, and G. E. Karniadakis, "A comprehensive and fair comparison between mlp and kan representations for differential equations and operator networks," Computer Methods in Applied Mechanics and Engineering, vol. 431, p. 117290, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045782524005462

[9] Z. Chen, J. Zhang, M. Arjovsky, and L. Bottou, "Symplectic recurrent neural networks," in International Conference on Learning Representations, 2020. [Online]. Available: https://openreview.net/forum?id=BkgYPREtPr

[10] P. Toth, D. J. Rezende, A. Jaegle, S. Racanière, A. Botev, and I. Higgins, "Hamiltonian generative networks," in International Conference on Learning Representations, 2020. [Online]. Available: https://openreview.net/forum?id=HJenn6VFvB

[11] Y. D. Zhong, B. Dey, and A. Chakraborty, "Symplectic ode-net: Learning hamiltonian dynamics with control," in International Conference on Learning Representations, 2020. [Online]. Available: https://openreview.net/forum?id=ryxmb1rKDS

[12] Z.-Y. Khoo, D. Wu, J. S. C. Low, and S. Bressan, "Separable hamiltonian neural networks," Phys. Rev. E, vol. 110, p. 044205, Oct 2024. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.110.044205

[13] E. Dierkes and K. Flaßkamp, "Learning hamiltonian systems considering system symmetries in neural networks," IFAC-PapersOnLine, vol. 54, no. 19, pp. 210–216, 2021, 7th IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control LHMNC 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896321021042

[14] L. Ziyin, T. Hartwig, and M. Ueda, "Neural networks fail to learn periodic functions and how to fix it," in Advances in Neural Information Processing Systems, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 1583–1594. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/1160453108d3e537255e9f7b931f4e90-Paper.pdf

[15] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "Physics-informed machine learning," Nature Reviews Physics, vol. 3, no. 6, pp. 422–440, Jun 2021. [Online]. Available: https://doi.org/10.1038/s42254-021-00314-5

[16] Z.-Y. Khoo, D. Zhang, and S. Bressan, "What's next? predicting hamiltonian dynamics from discrete observations of a vector field," in Database and Expert Systems Applications: 33rd International Conference, DEXA 2022, Vienna, Austria, August 22–24, 2022, Proceedings, Part II. Berlin, Heidelberg: Springer-Verlag, 2022, p. 297–302.

[17] K. Xu, M. Zhang, J. Li, S. S. Du, K. Kawarabayashi, and S. Jegelka, "How neural networks extrapolate: From feedforward to

graph neural networks," in 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=UH-cmocLJC

[18] P. Belcák and R. Wattenhofer, "Periodic extrapolative generalisation in neural networks," 2022.

[19] R. B. Levien and S. M. Tan, "Double pendulum: An experiment in chaos," American Journal of Physics, vol. 61, no. 11, pp. 1038–1044, Nov. 1993.

[20] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[21] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang, "JAX: composable transformations of Python+NumPy programs," 2023. [Online]. Available: http://github.com/google/jax

[22] L. Prechelt, "Early stopping-but when?" in Neural Networks: Tricks of the Trade, This Book is an Outgrowth of a 1996 NIPS Workshop. Berlin, Heidelberg: Springer-Verlag, 1998, p. 55–69.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.