Modeling Dynamical Systems with Neural Hybrid System Framework via Maximum Entropy Approach

Yejiang Yang and Weiming Xiang

***Abstract*— In this paper, a data-driven neural hybrid system modeling framework via the Maximum Entropy partitioning approach is proposed for complex dynamical system modeling such as human motion dynamics. The sampled data collected from the system is partitioned into segmented data sets using the Maximum Entropy approach, and the mode transition logic is then defined. Then, as the local dynamical description for their corresponding partitions, a collection of small-scale neural networks is trained. Following a neural hybrid system model of the system, a set-valued reachability analysis with low computation cost is provided based on interval analysis and a split and combined process to demonstrate the benefits of our approach in computationally expensive tasks. Finally, a numerical examples of the limit cycle and a human behavior modeling example are provided to demonstrate the effectiveness and efficiency of the developed methods.**

I. INTRODUCTION

Neural networks are widely used in modeling for their effectiveness without relying on the explicit mathematical model or prior knowledge of the system in a variety of research activities, e.g., modeling nonlinear dynamical systems in the description of Ordinary Differential Equations (ODEs) [1], modeling thermal conductivity of water-based nanofluid containing magnetic copper nanoparticles in [2], studying the neural network models for groundwater-level forecasting in coastal aquifers in [3], etc. However, due to the high complexity of large-scale neural network models, some computationally expensive tasks such as reachability analysis, and safety verification are challenging to perform on neural-network-based models. Therefore, computationally efficient modeling methods are in critical need for neural network-based models.

Since often viewed as black boxes, in most cases of data-driven modeling, a neural network model, especially a Deep Neural Network (DNN) model, may represent significant computation challenges in the training and verification due to their complex structures. When it comes to safety-critical applications, the time consumption for verification of the neural network model will be unacceptable. The verification tools presented in [4], [5], [6] and [7] provide powerful solutions to verify the neural network model, yet the computational bottleneck still exists considering the complex structure of the model. The recently proposed Pathways Language Model (PaLM) [8] developed from the Mixture of Experts Model (MoE) [9] shines a new light on DNN modeling by reducing the computational complexity through multiple small-size neural networks which are activated by a sparse gating system. Compared with DNN models for complex applications such as graphics and language, modeling the dynamic system with multiple Shallow Neural Networks (SNNs) to approximate the local information of the system makes the model more competitive (e.g., reducing computational complexity) to train and verify.

Inspired by the results in [10], a complex dynamical system can be modeled by a hybrid system with a finite number of partitions plus transition logic among them. If several small-scale dynamic learning processes in terms of optimizations can be performed concurrently, the computational complexity of modeling the dynamical systems will be significantly decreased. In the case of neural network modeling, this will result in distributed training of neural networks. In the framework of hybrid system modeling, the neural network will be trained based on samples selected by state space partitions. In this paper, we apply the data-driven Maximum Entropy (ME) partitioning approach [11] in bisecting the partitions that subsequently define the mode transition logic. After that, we train a neural network to approximate the dynamics in each partition. As a result, the computational complexity in both training and post-training verification processes will be reduced and due to parallel training and much less computational complexity for each step, the scalability thus can be further increased. In summary, the main contributions of this paper include:

- A novel data-driven Maximum Entropy partitioning method is proposed to characterize the dynamics into different modes, i.e, to bisect the state space into multiple partitions, simplify the model, and maintain the training accuracy. This aims to advance state-of-the-art of dynamical system modeling techniques with a focus on improving the model scalability.
- The novel neural hybrid system modeling method will be verified by analysis of set-valued reachability with the help of our proposed Split and Combine process, which provides a practical solution for the reachable set computation of distributed neural network models.

The paper is organized as follows: Preliminaries are given in Section II. The main result, a data-driven neural hybrid system modeling framework, is given in Section III. In Section IV, the analysis of set-valued reachability is provided for the cases of neural hybrid system modeling in a simple limit cycle and LASA data sets, which illustrate the effectiveness

This research was supported by the National Science Foundation, under NSF CNS Award no. 2223035 and NSF CAREER Award no. 2143351.

The authors are with School of Computer and Cyber Sciences, Augusta University, Augusta GA 30912, USA. wxiang@augusta.edu

Yejiang Yang is also with School of Electrical Engineering, Southwest Jiaotong University, China.

of our method. Conclusions are given in Section V.

II. Preliminaries

In this paper, we aim to model a complex dynamical system in the general form of

$$x(k+1) = f(x(k), u(k)) \tag{1}$$

where f is an nonlinear function which assumed to precisely describe the system dynamics while the state $x \in \mathbb{R}^d$ and system input $u \in \mathbb{R}^n$ for any time step $k = 0, 1, \ldots$. Specifically, in neural network modeling, f is approximated by one neural network in the form of

$$x(k+1) = \Phi(x(k), u(k)), \tag{2}$$

where Φ is the trained neural network aiming to approximate f based on given samples. However, one general neural network will lead to a complex model, which results in a computational burden in training and verification. Instead of modeling the dynamics for state space $\mathbb{R}^d$, we prefer to model the dynamics in a localized state space where for all the samples of the dynamics $x \in \mathcal{X} \subset \mathbb{R}^d$.

In order to break down a large approximation work into several smaller tasks, localized state space can be split up into subspaces known as partitions, defined as follows.

Definition 1: Localized state space $\mathcal{X}$ can be divided to a collection of N subspaces, which satisfies $\mathcal{X} \subseteq \bigcup_{i=1}^{N} \mathcal{P}_i$ and $\mathcal{P}_i \bigcap \mathcal{P}_j = \emptyset,\ \forall i \neq j$, in which the collection of sets $\mathcal{P} = \{\mathcal{P}_1, \ldots, \mathcal{P}_N\}$, is called partitions.

Remark 1: By classifying the system dynamics into different modes based on partitions, the neural network will be able to approximate the local dynamics. The obtaining of partitions should be a data-driven process when there is no prior knowledge of dynamical systems. Partitions are useful when we classify the system dynamics from samples, and they pave the way for parallel training of neural networks in approximating the dynamics. Specifically, neural networks can approximate the dynamics from the samples within their corresponding partitions.

A. Maximum Entropy Partitioning

The Maximum Entropy (ME) partitioning proposed in [11] is able to divide the localized state space $\mathcal{X}$ into different partitions based on the variation of the Shannon Entropy. With ME partitioning, information-rich regions are allocated more partitions and hence a more precise set of partitions will be obtained while the regions with sparse information are allocated fewer partitions.

The Shannon Entropy of a system with k partitions is

$$H(k) = -\sum_{i=1}^{k} p_i \log_2 p_i, \tag{3}$$

where p_i denotes the probability of occurrence of the partition $\mathcal{P}_i$. It should be noted that $H(1) = 0$ for $p_1 = 1$. Accordingly, given the sufficient sample data set, the probability of one partition $\mathcal{P}_i$, p_i is defined by

$$p_i = \frac{N_i}{N},$$

where N denotes the number of all samples while N_i denotes the number of samples belonging to $\mathcal{P}_i$.

The variation of Shannon Entropy $h(k),\ k \geq 2$ after we obtain k partitions can be denoted by

$$h(k) = H(k) - H(k-1),\ \forall k = 2, \ldots\ . \tag{4}$$

We stop dividing the partitions when the variation between two partitions becomes less than a threshold, $\epsilon \leq h(k)$, in order to obtain suitable partitions for the dynamical systems [12].

B. Neural Hybrid System

A neural hybrid system model consists of variable components describing both dynamics and the switching logic. Given localized state space $\mathcal{X}$ and the set of partitions $\mathcal{P} = \{\mathcal{P}_1, \ldots, \mathcal{P}_N\}$, we define the neural hybrid system with N partitions as follows.

Definition 2: A neural hybrid system is a collection of sets $\mathcal{H} = \langle \mathcal{P}, \Phi, \delta \rangle$ that includes partition set $\mathcal{P}$, mode dynamics description set Φ, and the mode transition δ.

For a neural hybrid system, it approximates f in localized state space $\mathcal{X}$ with multiple neural networks in the form of

$$x(k+1) = \Phi_{\delta(x(k))}(x(k), u(k)), \tag{5}$$

by defining the index set $\mathcal{I} \triangleq \{1, 2, \ldots, N\}$ where N is the number of subsystems and $\delta : \mathcal{X} \to \mathcal{I}$ is a function of state x denoting the switching signal which will activate the neural network approximation $\Phi_i \in \Phi \triangleq \{\Phi_1, \Phi_2, \ldots, \Phi_N\}$. The switching function δ maps $x(k)$ to its corresponding subsystem in the form of

$$\delta(x(k)) = i \iff x(k) \in \mathcal{P}_i.$$

Remark 2: Compared with neural network model (2) with one single neural network, the dynamics of a neural hybrid system (5) utilizes multiple neural networks for modeling. It should be noted that the set of partitions plays an important role in this process, namely, the switching logic is subsequently defined once the set of partitions is obtained.

In this work, we consider feedforward neural networks in the form of $\Phi : \mathbb{R}^{n_0} \to \mathbb{R}^{n_L}$ defined by the following recursive equations in the form of

$$\begin{cases} \eta_{(\ell)} = \phi_\ell(W_\ell \eta_{(\ell-1)} + b_\ell),\ \ell = 1, \ldots, L \\ \eta_{(L)} = \Phi(\eta_{(0)}) \end{cases} \tag{6}$$

where $\eta_{(\ell)}$ denotes the output of the ℓ-th layer of the neural network, and in particular $\eta_{(0)} \in \mathbb{R}^{n_0}$ is the input to the neural network and $\eta_{(L)} \in \mathbb{R}^{n_L}$ is the output produced by the neural network, respectively. $W_\ell \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$ and $b_\ell \in \mathbb{R}^{n_\ell}$ are weight matrices and bias vectors for the ℓ-th layer. $\phi_\ell = [\psi_\ell, \ldots, \psi_\ell]$ is the concatenation of activation functions of the ℓ-th layer in which $\psi_\ell : \mathbb{R} \to \mathbb{R}$ is the activation function.

In addition, the training performance of the neural network can be measured from the statistical point of view, i.e., Mean Square Error (MSE). Given a training set $\mathcal{W}$ consists of q

input-output samples $\{x_i, t_i\}$, the MSE performance of Φ can be measured by

$$MSE(\Phi, \mathcal{W}) = \frac{1}{q}\sum_{i=1}^{q}\sqrt{(\Phi(x_i) - t_i)^2}. \tag{7}$$

ME partitioning provides automated data that yields various partitions, this can be further enhanced to improve the model's scalability. The neural hybrid model will perform better in terms of scalability than the conventional model because several smaller-sized neural networks may be trained and tested in parallel, which saves computational resources. The data-driven modeling problem with the neural hybrid system is summarized below:

Problem 1: Given samples, the localized state space $\mathcal{X}$, how does one extract the neural hybrid system model $\mathcal{H} = \langle \mathcal{P}, \Phi, \delta \rangle$, including the following sub-problems:

1) How does one determine the partitions that characterized the system dynamics into different modes through a data-driven ME partitioning process?
2) How does one approximate the dynamics within $\mathcal{P}_i$ with neural networks $\Phi_i,\ i \in \mathcal{I}$ and finally construct the neural hybrid system model?

The rest of the paper will be focusing on solving Problem 1 in detail.

III. Neural Hybrid System Modeling via ME Partitioning Approach

The proposed modeling method aims to characterize the dynamics based on the variation of Shannon Entropy. Specifically, we aim to obtain the partitions of the localized state space from ME partitioning.

First, we define the training set segmented by the partitions as follows.

Definition 3: Given a training set $\hat{\mathcal{W}} = \{X, T\}$ in which $X \in \mathbb{R}^{(d+n)\times q}$, $T \in \mathbb{R}^{d\times q}$ are the input, output matrices for q samples, and pre-specified partitions $\tilde{\mathcal{P}} = \{\mathcal{P}_1, \ldots, \mathcal{P}_M\}$, a collection of M segmented input-output data pair set $\mathcal{W}$ can be defined as

$$\mathcal{W} = \{\mathcal{W}^{(1)}, \mathcal{W}^{(2)}, \ldots, \mathcal{W}^{(M)}\}, \tag{8}$$

where any input-output pair $\{x_i, t_i\} \in \mathcal{W}^{(j)}$ satisfies

$$x_i \in \mathcal{P}_j,\ \forall \{x_i, t_i\} \in \mathcal{W}^{(j)}. \tag{9}$$

Remark 3: It is noted that the pre-specified partitions $\tilde{\mathcal{P}} = \{\tilde{\mathcal{P}}_1, \ldots, \tilde{\mathcal{P}}_M\}$ such as the initial lattices of localized state space $\mathcal{X}$ are not the partitions in $\mathcal{P} = \{\mathcal{P}_1, \ldots, \mathcal{P}_N\}$ in a neural hybrid system model. There is a Merging process developed to generate an optimized partition $\mathcal{P}$ out of $\tilde{\mathcal{P}}$ in the modeling framework, and normally $N \ll M$.

A. Bisecting via ME Partitioning

Maximum Entropy (ME) partitioning will divide the localized state space $\mathcal{X}$ into multiple partitions while subsequently segmenting the training set. In this paper, a partition is considered in the form of interval denoted by $\mathcal{P} = [\underline{x}, \overline{x}] \subset \mathcal{X}$, in which $\underline{x} = \begin{bmatrix}\underline{x}_1 & \ldots & \underline{x}_d\end{bmatrix}^\top$, $\overline{x} = \begin{bmatrix}\overline{x}_1 & \ldots & \overline{x}_d\end{bmatrix}^\top$.

Specifically, we define the maximum length of $\mathcal{P}$ as

$$D_{j,max} \triangleq \|\overline{x} - \underline{x}\|_\infty,$$

in which $j = \arg\max_i |\overline{x}_i - \underline{x}_i|,\ \forall i = 1, 2, \ldots, d$. The maximum length for $\mathcal{P}$ is denoted by $D_{j,\max}$ which means the maximum length is at jth dimension.

The bisection method in [13] divides one partition into two at jth dimension, i.e., $\mathcal{P}^{(i)} \to \{\mathcal{P}^{(i,1)}, \mathcal{P}^{(i,2)}\}$ where

$$\begin{aligned}
\overline{x}_j^{(i,1)} &= \underline{x}_j^{(i)} + \frac{1}{2}D_{j,\max},\\
\underline{x}_j^{(i,1)} &= \underline{x}_j^{(i)},\\
\underline{x}_j^{(i,2)} &= \overline{x}_j^{(i)} - \frac{1}{2}D_{j,\max},\\
\overline{x}_j^{(i,2)} &= \overline{x}_j^{(i)},
\end{aligned}$$

in which $\underline{x}_j^{(i,1)}$ denotes the lower bound of $\mathcal{P}^{(i,1)}$ at jth dimension, etc. The bisection process only bisects the interval in one dimension and results in two interval sets, i.e.,

$$\begin{cases}\underline{x}_k^{(i,1)} = \underline{x}_k^{(i,2)} = \underline{x}_k^{(i)}\\ \overline{x}_k^{(i,1)} = \overline{x}_k^{(i,2)} = \overline{x}_k^{(i)}\end{cases},\ \forall k \neq j,\ k = 1, \ldots, d.$$

After $\mathcal{P}^{(i)} \to \{\mathcal{P}^{(i,1)}, \mathcal{P}^{(i,2)}\}$, the corresponding training set $\mathcal{W}^{(i)} \to \{\mathcal{W}^{(i,1)}, \mathcal{W}^{(i,2)}\}$, thus the Shannon Entropy from (3) will change. Let $N_i = \left|\mathcal{W}^{(i)}\right|$ denotes the number of samples in $\mathcal{P}^{(i)}$, when bisecting $\mathcal{P}^{(i)} \to \{\mathcal{P}^{(i,1)}, \mathcal{P}^{(i,2)}\}$ the variation of Shannon Entropy can be written in

$$h_i = \frac{N_{i,1}\log_2 \frac{N_{i,1}}{N_{i,1}+N_{i,2}} + N_{i,2}\log_2 \frac{N_{i,2}}{N_{i,1}+N_{i,2}}}{N_i} \tag{10}$$

which suggests $h_i \geq 0$, i.e., the Shannon Entropy will increase when we bisect any elements from $\mathcal{P}$.

One partition will stop being bisected if the variation of the Shannon Entropy $h_i \leq \epsilon$. When $\forall \mathcal{P}_i \in \mathcal{P},\ h_i \leq \epsilon$, the data-driven partitioning process will stop and lead to M partitions in $\tilde{\mathcal{P}}$ obtained.

Remark 4: Given the partitioning process only bisects one partition in one step, the variation of Shannon Entropy will therefore apply to the whole partitioning process, specifically, the variation after one bisecting process can be written in (10). Besides, a small value of threshold ϵ leads to a large size of partitions which results in increased neural network training while a large value of ϵ leads to a small size of partitions which may provide inadequate for the system dynamics.

B. Merging and Training Neural Hybrid Systems

Entropy-based partitioning can be further optimized from the perspective of subsystems training after getting M partitions based on ME partitioning.

By setting a fix configuration of each individual neural network, i.e., layers, neurons from each layers and their activation function are the same, we can train a set of neural network approximations $\Phi = \{\Phi_1, \Phi_2, \ldots, \Phi_M\}$ under the set of segmented data $\mathcal{W} = \{\mathcal{W}^{(1)}, \mathcal{W}^{(2)}, \ldots, \mathcal{W}^{(M)}\}$ with

$$\min\nolimits_{W_i, b_i} \|\Phi_i(X_i) - T_i\| \tag{11}$$

in which W_i, b_i are the weight matrices and the bias vectors for ith subsystem, X_i are input data matrix and T_i is output data matrix from input-output pair $\mathcal{W}^{(i)}$, respectively.

The above training process will result in M neural networks trained as dynamical descriptions for M lattices, however, a large number of subsystems will result in a complex model which is not expected. We apply the Merging technique, which aims to simplify the model from the training point of view while maintaining accuracy in approximating the samples.

Considering two segmented training set $\mathcal{W}^{(i)}$, $\mathcal{W}^{(j)}$, $i \neq j$ for ith and jth partitions, a given neural network configuration, we train Φ with

$$\min_{W_{i,j}, b_{i,j}} \left\| \Phi_{i,j}(X_{i,j}) - T_{i,j} \right\|, \quad (12)$$

in which $W_{i,j}, b_{i,j}$ are the weight matrix and bias vector of $\Phi_{i,j}$, $X_{i,j}$, $T_{i,j}$ are the combined input matrix and target output matrix in the form of $X_{i,j} = \left[X_i, X_j\right]$, $T_{i,j} = \left[T_i, T_j\right]$.

Considering a training tolerance γ, if

$$\gamma \geq MSE(\Phi_{i,j}, \mathcal{W}^{(i,j)}) \quad (13)$$

holds, then the two partitions will be considered to have a similar training performance under the given neural network configuration, and hence their corresponding partitions $\mathcal{P}_i$ and $\mathcal{P}_j$ will be merged. The process of merging different partitions based on training sets is called Merging. After Merging, the set of partitions will be the set of partitions of our neural hybrid system model. Due to redundant partitions being merged, the complexity of our model will be reduced.

We merge the redundant partitions that have similar training performance under the same neural network configuration and obtain $\mathcal{P}$ that subsequently define switching logic δ for the neural hybrid automaton system after Merging, i.e., the switching of subsystems $\mathcal{P}_i \to \mathcal{P}_j$ can be abstracted if there exist system state $x(k) \in \mathcal{P}_i$ and successive state $x(k+1) \in \mathcal{P}_j$.

Neural hybrid system modeling with ME partitioning and Merging can be summarized as follows.

- As the only coefficient in the process of ME partitioning, the threshold ϵ provides the least variation condition of the Shannon Entropy to characterize the samples of the dynamics, which means this data-driven process can be easily tuned.
- The redundant partitions will be merged based on the training performance of a given neural network configuration, which will reduce the model complexity while maintaining the accuracy in approximating the dynamics.

After data-driven ME partitioning and dynamics learning with Merging, we will be able to obtain the explicit $\mathcal{H} = \langle \mathcal{P}, \Phi, \delta \rangle$ for complex dynamics.

IV. Application to Complex Dynamical Systems Modeling and Verification

We apply the neural hybrid system modeling framework in modeling the simple limit cycle and Human Cyber-Physical System (HCPS), in which a single hidden layer neural network known as ELM [14] will be served as the dynamical description for the neural hybrid system model. We will show the advantage of our proposed modeling method by verifying our neural hybrid system model and the conventional neural network model through analysis of set-valued reachability.

A. Set-Valued Reachability Analysis

Analysis of set-valued reachability for a neural network model can be referred to as safety verification in [15]. However, the computation of reachable sets usually represents high computational complexity due to the large size of the single neural network model. In this paper, we will demonstrate the advantages of our proposed method through the analysis of set-valued reachability. The reachable set of neural networks (6) is given as follows.

Definition 4: Given neural network (6) with a bounded input set $\mathcal{X}_{(k)}$ at kth time step, the following set

$$\mathcal{X}_{(k+1)} \triangleq \{x(k+1) \mid x(k+1) = \Phi(x(k)), \ \forall x(k) \in \mathcal{X}_{(k)}\} \quad (14)$$

is called the reachable set of neural network (6) at $k+1$th time step.

In the case of neural hybrid system modeling, $\mathcal{X}_{(k)}$ may intersect multiple elements from $\mathcal{P}$, which means there will be a split computation of reachable set for each intersection. This process is called Split and is defined as follows.

Definition 5: For a reachable set $\mathcal{X}_{(k)}$ of $\mathcal{H}$ intersects with l elements of $\mathcal{P}$, given a subspace $\mathcal{V}_{i,k}$ from $\mathcal{X}_{(k)}$ in which $\mathcal{V}_{i,k} : \mathcal{V}_{i,k} = (\mathcal{X}_{(k)} \cap \mathcal{P}_i); \ \cup \mathcal{V}_{i,k} = \mathcal{X}_{(k)}, \ \forall i = 1, \ldots, N$, the process of splitting analysis of the output space $\mathcal{V}_{i,k+1}$ is given by

$$\mathcal{V}_{i,k+1} \triangleq \{\eta_{i,k+1} \mid \eta_{i,k+1} = \Phi_m(\eta_{i,k}), \ \eta_{i,k} \in \mathcal{V}_{i,k}\} \quad (15)$$

where the process of obtaining $\mathcal{V}_{i,k+1}, \ \forall i = 1, 2, \ldots, N$ is called Split.

After Split, the Combine process is needed to obtain a complete reachable set for the next step.

Definition 6: For $\mathcal{V}_{i,k+1}, \ i = 1, \ldots, l$ the output reachable set $\mathcal{X}_{(k+1)}$ for a neural hybrid system model $\mathcal{H}$ at time step $k+1$ is given by

$$\mathcal{X}_{(k+1)} \triangleq \bigcup_{i=1}^{l} \mathcal{V}_{i,k} \quad (16)$$

by which the Combine derives the reachable set at $k+1$th time instance.

With the Split and Combine defined above, the reachable set of $\mathcal{H}$ can be parallel analyzed at time instance k if $\mathcal{X}_{(k)}$ intersects with multiple elements from $\mathcal{P}$.

Remark 5: According to [6], [15], [16], the computational cost for the set-valued analysis of the neural network is mainly affected by the scale of the neural network model, e.g., the layers and neurons of the neural network model. In our case, due to parallel training of shallow neural networks, the neural hybrid system model may have less computational cost compared with traditional methods.

B. Simple Limit Cycle

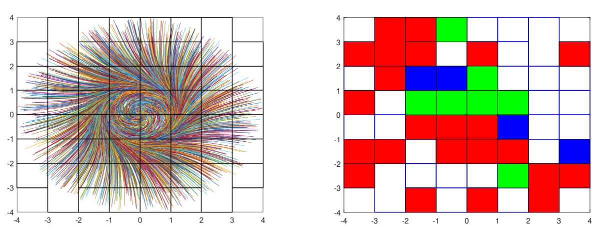
A numerical example of a limit cycle system borrowed from [17] is used to validate our approach. The mathematical model is in the form of

$$\begin{aligned} r(k+1) &= (1+\tau)r(k) - \tau r^3(k) + \tau u(k) \\ \theta(k+1) &= \theta(k) + \tau\omega \\ u(k) &= \mu + \delta\zeta(k) \end{aligned} \tag{17}$$

where $\omega = 2\pi/3$ and $\tau = 0.1$ are the angular velocity and time step width, respectively. The uniform random number $\zeta(k) \sim U(-1,1)$. Namely, the external input $u(k) \sim U(\mu-\delta, \mu+\delta)$ ($\mu = 0.2$ and $\delta = 1.5$) in which U denotes uniform distribution.

We obtain the samples by giving the mathematical model (17) random initial condition when $r(0) \in [-4,4]$, $\theta(0) \in [-\pi,\pi]$. The data obtained is represented by a right-angle coordinate system, namely, $x = [x_1, x_2]^\top$. Then, we bisect the state space with data-driven ME partitioning where the threshold $\epsilon \times |W| = 40$, the partitions and the trajectories are shown in Fig. 1 (a).

After we obtain the partitions, we merge the redundant partitions of the neural hybrid system model with a tolerance $\delta = 3\times 10^{-5}$, and a fixed ELM structure with 40 neurons and ReLU as the activation function. The merged set of partitions is shown in Fig. 1 (b).



(a) Samples and ME Partitioning. (b) Merging Redundant Partitions.

Fig. 1. (a) Trajectories of the limit cycle with random initial condition $r(0) \in [-4,4]$, $\theta(0) \in [-\pi,\pi]$ each of which contains 150 samples and the input $u \sim U(-1.3, 1.7)$ while we bisect the input space into 56 partitions using ME partitioning. (b) Redundant lattices are merged into three partitions, namely blue (4 redundant partitions are merged), green (7 redundant partitions are merged), and red partitions (21 redundant partitions are merged), there are 27 elements in the set of partitions in total.

As observed in Fig. 1, the merging process has significantly reduced the number of partitions, i.e., from 56 to 27 partitions, as a result, the neural hybrid system model will be simplified. Neural network model $\hat{\Phi}$ with 200 hidden neurons with similar MSE is trained as well for comparison study. The time consumption for neural networks training and analysis of set-valued reachability and the MSE performance are given in Tab. I. It can be observed that neural hybrid system model $\mathcal{H}$ is with a significant time reduction in the analysis of set-valued reachability while having a similar output set estimation Fig. 2, which indicates our modeling framework is computationally efficient while sacrificing little in accuracy.

TABLE I
MSE AND TIME CONSUMING OF $\mathcal{H}$ AND ELM MODEL

Method	MSE	Training	Set-valued Reachability
ELM	0.01669	4.614492 s	6.13474×10^4 s
$\mathcal{H}$	0.06913	0.242433 s	521.8130 s

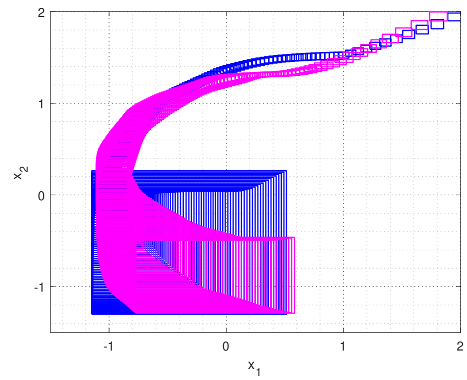


Fig. 2. Output reachable set of single ELM model (purple) and the neural hybrid system model (blue) given initial condition $x_1 \in [1.8, 2.0]$, $x_2 \in [1.8, 2.0]$ for 200 times steps, they have a similar reachable set estimation while the time consuming for $\mathcal{H}$ is significantly less.

TABLE II
MSE AND TIME CONSUMING OF $\mathcal{H}$ AND ELM MODEL (SPOON)

Method	MSE	Training	Set-valued Reachability
ELM	1.8556×10^{-4}	0.352899 s	8.4465×10^4 s
$\mathcal{H}$	5.7373×10^{-4}	0.005796 s	69.9705 s

TABLE III
MSE AND TIME CONSUMING OF $\mathcal{H}$ AND ELM MODEL (P SHAPE)

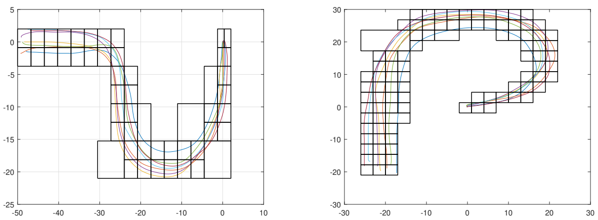
Method	MSE	Training	Set-valued Reachability
ELM	2.4879×10^{-4}	0.042067 s	4.53482×10^4 s
$\mathcal{H}$	4.5054×10^{-4}	0.004505 s	242.8855 s

We use parallel-trained ELMs as the descriptions for the local partitions of the system, hence the time of training $\mathcal{H}$ is the longest time from the training of a set of neural networks.

C. Human Cyber-Physical Systems

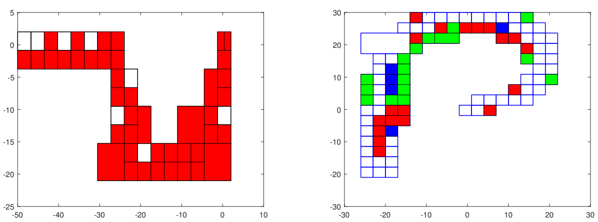
Learning-based methods have been promoted as an effective way to model motions [18], [19]. Consider a Human Cyber-Physical system (HCPS) borrowed from the LASA data set [20] which contains 20 handwriting motion demonstrations of human users. We model the Spoon and P shape with our proposed neural hybrid system models with 20 neurons each subsystem. Fig. 3 (a), (b) are the partitions obtained from the data-driven ME partitioning when $\epsilon|N| = 40$, while Fig. 3 (c), (d) are the merging partitions, while there are 9 partitions in (c), and 58 partitions for (d).

In summary, this HCPS modeling demonstrates that the proposed neural hybrid system models can capture behaviors of complex dynamical systems with high modeling accuracy but significantly lower computational complexities in training and verification, which are traditionally considered to be



(a) Samples and Partitions (Spoon Shape).

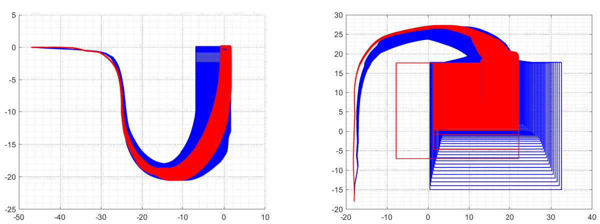
(b) Samples and Partitions (P Shape).



(c) Merging Redundant Partitions (Spoon Shape).

(d) Merging Redundant Partitions (P Shape).

Fig. 3. There are 50 partitions from (a) while 94 partitions from (b). From (a) to (c), there is one merged partition (red) in (c). In (d), there are three merged partitions where green (15 redundant partitions are merged), red (17 redundant partitions are merged), and blue (5 redundant partitions are merged), there are 27 elements in the set of partitions in total.



(a) Spoon Shape Modeling.

(b) P Shape Modeling.

Fig. 4. Output reachable set of single ELM model with 200 neurons (blue) and the hybrid system model with 20 neurons each subsystem (red) for 850 steps by given initial conditions wherein (a) $x_1 \in [-47.02, -47]$, $x_2 \in [0, 0.2]$ and in (b) $x_1 \in [-18.02, -18]$, $x_2 \in [-18.02, -18]$.

computationally expensive tasks for neural network models.

V. Conclusion

In this paper, a neural hybrid system modeling with ME partitioning is developed to model the dynamical system. First, sampled data is generated by the dynamical system given random initial conditions. Then, the state space is bisecting into multiple partitions based on the variation of the Shannon Entropy. Neural networks are trained as the dynamical description for their corresponding partitions. The set-valued reachability is analyzed by the reachable set computation with Split and Combine processes. Modeling a numerical example of the limit cycle and the application to human handwriting modeling with our proposed modeling method are presented to illustrate the effectiveness. Compared with modeling with one single neural network, the computational cost can be significantly reduced for computationally expensive tasks such as neural network training and reachable set computation, etc.

References

[1] P. V. Kuptsov, A. V. Kuptsova, and N. V. Stankevich, "Artificial neural network as a universal model of nonlinear dynamical systems," *arXiv preprint arXiv:2104.05402*, 2021.

[2] M. Ghazvini, H. Maddah, R. Peymanfar, M. H. Ahmadi, and R. Kumar, "Experimental evaluation and artificial neural network modeling of thermal conductivity of water based nanofluid containing magnetic copper nanoparticles," *Physica A: Statistical Mechanics and its Applications*, vol. 551, p. 124127, 2020.

[3] T. Roshni, M. K. Jha, and J. Drisya, "Neural network modeling for groundwater-level forecasting in coastal aquifers," *Neural Computing and Applications*, pp. 1–18, 2020.

[4] H. Zhang, T.-W. Weng, P.-Y. Chen, C.-J. Hsieh, and L. Daniel, "Efficient neural network robustness certification with general activation functions," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[5] S. Wang, H. Zhang, K. Xu, X. Lin, S. Jana, C.-J. Hsieh, and J. Z. Kolter, "Beta-crown: Efficient bound propagation with per-neuron split constraints for neural network robustness verification," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[6] H.-D. Tran, X. Yang, D. Manzanas Lopez, P. Musau, L. V. Nguyen, W. Xiang, S. Bak, and T. T. Johnson, "NNV: the neural network verification tool for deep neural networks and learning-enabled cyber-physical systems," in *International Conference on Computer Aided Verification*, pp. 3–17, Springer, 2020.

[7] M. Fazlyab, M. Morari, and G. J. Pappas, "Safety verification and robustness analysis of neural networks via quadratic constraints and semidefinite programming," *IEEE Transactions on Automatic Control*, vol. 67, no. 1, pp. 1–15, 2020.

[8] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, *et al.*, "Palm: Scaling language modeling with pathways," *arXiv preprint arXiv:2204.02311*, 2022.

[9] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, no. 1, pp. 79–87, 1991.

[10] O. A. Beg, H. Abbas, T. T. Johnson, and A. Davoudi, "Model validation of PWM DC-DC converters," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 9, pp. 7049–7059, 2017.

[11] V. Rajagopalan and A. Ray, "Symbolic time series analysis via wavelet-based partitioning," *Signal Processing*, vol. 86, no. 11, pp. 3309–3320, 2006.

[12] V. Rajagopalan, A. Ray, R. Samsi, and J. Mayer, "Pattern identification in dynamical systems via symbolic time series analysis," *Pattern Recognition*, vol. 40, no. 11, pp. 2897–2907, 2007.

[13] S. Skelboe, "Computation of rational interval functions," *BIT Numerical Mathematics*, vol. 14, no. 1, pp. 87–95, 1974.

[14] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, 2006.

[15] W. Xiang, H.-D. Tran, X. Yang, and T. T. Johnson, "Reachable set estimation for neural network control systems: A simulation-guided approach," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 5, pp. 1821–1830, 2021.

[16] W. Xiang, H.-D. Tran, and T. T. Johnson, "Output reachable set estimation and verification for multilayer neural networks," *IEEE transactions on Neural Networks and Learning Systems*, vol. 29, no. 11, pp. 5777–5783, 2018.

[17] S. H. Strogatz, *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. CRC press, 2018.

[18] R. F. Reinhart, Z. Shareef, and J. J. Steil, "Hybrid analytical and data-driven modeling for feed-forward robot control," *Sensors*, vol. 17, no. 2, p. 311, 2017.

[19] R. F. Reinhart and J. J. Steil, "Neural learning and dynamical selection of redundant solutions for inverse kinematic control," in *2011 11th IEEE-RAS International Conference on Humanoid Robots*, pp. 564–569, IEEE, 2011.

[20] M. Khansari, E. Klingbeil, and O. Khatib, "Adaptive human-inspired compliant contact primitives to perform surface–surface contact under uncertainty," *The International Journal of Robotics Research*, vol. 35, no. 13, pp. 1651–1675, 2016.