2024 American Control Conference (ACC)
July 8-12, 2024. Toronto, Canada

Hybrid Zonotope-Based Backward Reachability Analysis for Neural Feedback Systems With Nonlinear Plant Models

Hang Zhang, Yuhao Zhang and Xiangru Xu

Abstract— **The increasing prevalence of neural networks in safety-critical control systems underscores the imperative need for rigorous methods to ensure the reliability and safety of these systems. This work introduces a novel approach employing hybrid zonotopes to compute the over-approximation of backward reachable sets for neural feedback systems with nonlinear plant models and general activation functions. Closed-form expressions as hybrid zonotopes are provided for the over-approximated backward reachable sets, and a refinement procedure is proposed to alleviate the potential conservatism of the approximation. Two numerical examples are provided to illustrate the effectiveness of the proposed approach.**

I. Introduction

The integration of Neural Networks (NNs) in feedback control systems necessitates the development of effective methods to analyze the system's behavior with robust and reliable performance guarantees. NNs demonstrate remarkable adaptability and are capable of learning intricate mappings from input to output, rendering them a potent tool for controlling nonlinear dynamical systems. Nevertheless, NNs are highly sensitive to even minor perturbations in the input space [1], potentially leading Neural Feedback Systems (NFSs), which are systems with NN controllers in the feedback loop, to safety hazards. To analyze the behavior of such NFSs, various methods have been proposed for the reachability analysis of NFSs. The safety properties of these systems can be verified by the forward reachability analysis [2]–[9] or the backward reachability analysis [10]–[12].

The backward reachability problem is to compute a set of states, known as the Backward Reachable Set (BRS), from which the system's trajectories can reach a specified target region within a finite time horizon. When the target region is chosen as the unsafe set for safety verification problems, the computed BRS will bound the back-propagated trajectories that enter the unsafe set. Although there exist various methods for computing the BRSs of systems without NNs [13], [14], they are not directly applicable to NFSs due to the inherently complex and nonlinear nature of NNs. For isolated ReLU-activated NNs, [10] proposed a method to compute the exact BRS of NNs by representing the NNs as piecewise linear functions via the activation patterns of the ReLU functions. For the NFS with a linear plant model, a method based on Linear Programming (LP) relaxation was proposed in [15] to over-approximate the BRSs; this method was extended in [11] for NFSs with nonlinear plant models by utilizing the over-approximation tool OVERT [16] and a guided partition algorithm. Despite the effectiveness of these LP-based methods, the computed BRSs are usually limited in the form of intervals and tend to be conservative.

Recently, Hybrid Zonotope (HZ) was proposed as a new set representation that generalizes the polytope or constrained zonotope [17]. By including binary variables in the set representation, an HZ is equivalent to a finite union of polytopes, and can conveniently represent non-convex and disconnected sets with flat faces. HZs have proven advantages for set-based computations in terms of computational efficiency and accuracy [17]–[21]. In particular, HZs possess properties that make them well-suited for reachability analysis of NFSs by using set operations. For example, [8] showed that a Feed-forward Neural Network (FNN) with ReLU activation functions can be exactly represented by an HZ and provided algorithms to compute the exact and approximated forward reachable sets (FRSs) of NFSs; [12] considered NFSs with linear plant models and presented results for computing exact BRSs in the form of HZs.

This work aims to compute the HZ representations of the over-approximated BRSs for NFSs where the plant model is nonlinear and the controller is an FNN with general activation functions. The contributions of this work are at least twofold: (i) By leveraging over-approximation tools of nonlinear functions, closed-form identities are provided for over-approximating the BRSs of NFSs with general nonlinear plant models and ReLU-activated FNN controllers; (ii) The proposed method is extended to NNs with other commonly used activation functions such as leaky ReLU, tanh, and sigmoid. The remainder of this paper is organized as follows: Section II provides preliminaries on HZ and the problem statement; Section III presents the results for NFSs with nonlinear plant models and ReLU-activated FNNs; Section IV extends these results to NFSs with general activation functions; two numerical examples are provided in Section V to demonstrate the performance of the proposed method; and finally, the concluding remarks are made in Section VI.

Notation. Vectors and matrices are denoted as bold letters (e.g., and). The -th component of a vector is denoted by with . For a matrix , denotes the matrix constructed by the -th to -th rows of . The identity matrix in is denoted as and is the -th column of . The vectors and matrices whose entries are all 0 (resp. 1) are denoted as (resp.). Given sets , and a matrix , the Minkowski sum of and is

This work was supported in part by National Science Foundation grants CMMI-2237850 and CNS-2222541. H. Zhang, Y. Zhang and X. Xu are with the Department of Mechanical Engineering, University of Wisconsin-Madison, Madison, WI, USA. Email: {hang.zhang,yuhao.zhang2,xiangru.xu}@wisc.edu.

, the generalized intersection of and under is , and the -ary Cartesian power of is . Given _, ¯ such that _ ¯ where is in the element-wise sense, the set _ ¯ is denoted as _ ¯. Given two scalar _, ¯ such that _ ¯, the interval _ ¯ is denoted as _ ¯.

II. Preliminaries & Problem Statement

A. Hybrid Zonotopes

Definition 1: [17, Definition 3] The set is a *hybrid zonotope* if there exist , , , , , such that where is the unit hypercube in . The *HCG*-representation of the hybrid zonotope is given by .

Given an HZ , the vector is called the center, the columns of are called the *binary generators*, and the columns of are called the *continuous generators*. The representation complexity of the HZ is determined by the number of continuous generators , the number of binary generators , and the number of equality constraints [17]. For simplicity, we define the set . An HZ with binary generators is equivalent to the union of constrained zonotopes [17, Theorem 5]. Identities are provided to compute the linear map and generalized intersection [17, Proposition 7], the union operation [18, Proposition 1], and the Cartesian product of HZs [22, Proposition 3.2.5].

B. Problem Statement

Consider the following discrete-time nonlinear system:

(1)

where are the state and the control input, respectively. We assume and _ ¯, where is called the state set and is called the input set. The control input is given as

(2)

where is an -layer FNN. The -th layer weight matrix and the bias vector of are denoted as and , respectively, where . Denote as the neurons of the -th layer and as the dimension of . Then, for ,

(3)

where and is the vector-valued activation function constructed by component-wise repetition of the activation function , i.e., In the last layer, only the linear map is applied, i.e., . The activation function considered in this work is not restricted to ReLU; see Section IV for more discussion. In the following, we assume the FNN controller satisfies , ; this assumption is not restrictive since the output of any FNN can be saturated into a given range by adding two additional layers with ReLU activation functions [4], [12].

The NFS consisting of system (1) and the controller (2) is a closed-loop system denoted as:

(4)

Given a target set for the system (4), the set of states that can be mapped into the target set by (4) in exactly steps is defined as the -step BRS and denoted as

For simplicity, the one-step BRS is also denoted as , i.e., . In this work, we assume the state set , the input set , the target set , and the set that over-approximates the nonlinear plant model are all represented as HZs; this assumption enables us to handle sets and plant models using a unified HZ-based approach.

The following problem will be investigated in this work: *Given a target set represented as an HZ and a time horizon , compute the over-approximation of BRS of the neural feedback system* (4)*, for* .

Compared with our previous result [12] that computes the exact BRS for NFSs consisting of linear plant models and ReLU-activated FNN controllers, this work considers NFSs consisting of general nonlinear plant models and FNNs with general activation functions, and therefore, only over-approximated BRSs are computed.

III. Backward Reachability Analysis of ReLU Activated Neural Feedback Systems

Our proposed method includes three main steps: (i) compute an HZ envelope of the nonlinear function ; (ii) find an HZ graph representation of the input-output relationship of the FNN control policy ; (iii) calculate the one-step over-approximated BRS in closed form, and refine the set when an accurate result takes precedence over computational efficiency. In this section, we assume the controller shown in (2) is an FNN with ReLU activation functions across all the layers, i.e., ; this assumption will be relaxed in the next section. Note that the time index of and are omitted for clarity in the section.

A. Envelope of Nonlinear Functions

For NFSs with linear plant models and ReLU activation functions, HZ-based methods have been proposed to compute both the exact FRSs and the exact BRSs [8], [12], [19]. However, since HZs can only represent sets with flat faces (i.e., unions of polytopes), these exact analysis methods are inapplicable to the case of general nonlinear plant models shown in (4). As the first step towards computing the over-approximated BRS of the system (4), we will find an envelope of the nonlinear function .

We denote the *graph set* of a given function over the domain as . Since the exact representation of

the graph set $\mathcal{G}_f(\mathcal{X},\mathcal{U})$ is difficult to find, we will compute an HZ-represented over-approximation of $\mathcal{G}_f(\mathcal{X},\mathcal{U})$, denoted as $\overline{\mathcal{H}}_f(\mathcal{X},\mathcal{U})$, such that $\overline{\mathcal{H}}_f(\mathcal{X},\mathcal{U}) \supseteq \mathcal{G}_f(\mathcal{X},\mathcal{U})$. The envelope of a nonlinear function over a given domain can be found using off-the-shelf methods [23]–[25]. In this work, we make use of two approximation tools, namely Special Ordered Set (SOS) [26] and OVERT [16], mainly because the envelopes obtained by these two methods are easy to compute and can be readily incorporated with the HZ-based set computations.

The over-approximation method based on SOS was originally developed for solving nonlinear and nonconvex optimization problems [26]. In recent work, SOS approximations were utilized to compute the FRSs of nonlinear dynamical systems [20]. The SOS approximation $\mathcal{S}$ of a scalar-valued function was defined in [20, Definition 3], while the identity that converts $\mathcal{S}$ into an HZ was provided in [20, Theoem 4]. The maximum SOS approximation error δ for some common nonlinear functions was provided in [25]. Given an SOS approximation $\mathcal{H}_{SOS}$ as an HZ and the maximum approximation error $\delta \in \mathbb{R}^n$, an envelop of the function $\boldsymbol{f}$ can be given as an HZ $\overline{\mathcal{H}}_f(\mathcal{X},\mathcal{U}) \triangleq \mathcal{H}_{SOS} \oplus \mathcal{E}$, where $\mathcal{E} = \langle \text{diag}([\mathbf{0}_{n+m}^\top \ \ \delta^\top]), \emptyset, \mathbf{0}_{2n+m}, \emptyset, \emptyset, \emptyset \rangle$ is the error HZ. For nonlinear functions with multidimensional inputs, decomposition techniques can be applied to avoid the exponential complexity of SOS approximations [25].

For the OVERT method, a nonlinear function $\boldsymbol{f}$ with multidimensional inputs and outputs will be decomposed into multiple elementary functions $e(\cdot)$ with single inputs and single outputs by the functional rewriting technique, and then over-approximations of the element functions will be constructed using a set of optimally chosen piecewise linear upper and lower bounds [16]. In this work, we assume that the number of breakpoints of the piecewise linear upper and lower bounds, denoted as N ($N \geq 3$), are the same and only optimize the breakpoints along one of the piecewise linear bounds using OVERT. Denote the set of breakpoints for the piecewise linear upper (resp., lower) bound as $\{(x_i, y_i^{ub})\}_{i=1}^N$ (resp., $\{(x_i, y_i^{lb})\}_{i=1}^N$). Note that the x-coordinates of the lower and upper bound are the same. Since the endpoints satisfy $y_1^{lb} = y_1^{ub} = e(x_1)$ and $y_N^{lb} = y_N^{ub} = e(x_N)$, the set bounded by the piecewise linear upper and lower bounds can be seen as a union of $N-1$ polytopes. Each polytope can be constructed by the breakpoints of the upper and lower bounds, leading to $N-1$ V-rep polytopes that over-approximates the nonlinear element functions. Using [21, Theorem 5], an HZ envelope representing the union of V-rep polytopes can be found.

While both SOS-based and OVERT-based methods can utilize decomposition techniques to generate a tight over-approximation of $\boldsymbol{f}$, the resultant HZ envelopes exhibit varying levels of approximation accuracy and set complexity. For example, to compare the set complexity of the resulting HZs, we let the number of breakpoints be N for both the SOS and the OVERT approaches. Then, for any one-dimensional nonlinear elementary function $e(x)$, the complexity of the HZ representations computed by SOS and OVERT is given by $n_g^{SOS} = 2N+2, n_b^{SOS} = N-1, n_c^{SOS} = N+4, n_g^{OVERT} = 4N-4, n_b^{OVERT} = N-1, n_c^{OVERT} = 2N$. It can be observed that the HZ representation based on OVERT approximations tends to have a larger set complexity than the SOS method in terms of more continuous generators and more equality constraints. However, the OVERT method usually leads to more accurate over-approximations than the SOS method. This is because the approximation error δ in the SOS method remains constant across the entire domain, whereas the approximation error in the OVERT method varies based on the distance from the breakpoints. An example of over-approximation of the tanh function using both methods is shown in Figure 1. Comparisons of the two methods will be further discussed in Section V.

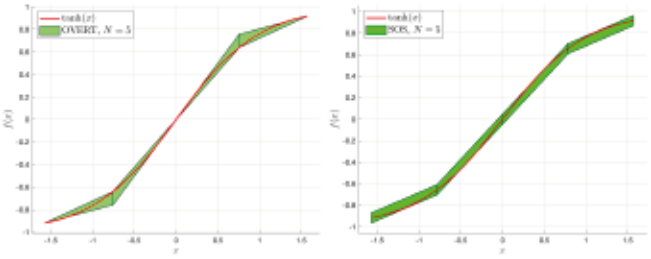


Fig. 1: Comparison between OVERT and SOS. Both over-approximate $\tanh(x)$ for $x \in [-\pi/2, \pi/2]$ with the union of 4 polytopes.

B. Over-approximation of One-step BRS

In this subsection, we will compute the closed-form expression of an over-approximation of the one-step BRS, utilizing the HZ envelope of $\boldsymbol{f}$ calculated in Section III-A.

In order to incorporate the FNN controller $\boldsymbol{\pi}$ into the HZ-based framework, we apply Algorithm 1 in our previous work [12] to compute an HZ-represented graph set of $\boldsymbol{\pi}$ over the domain $\mathcal{X}$, denoted as $\mathcal{H}_\pi = \langle \mathbf{G}_\pi^c, \mathbf{G}_\pi^b, \mathbf{c}_\pi, \mathbf{A}_\pi^c, \mathbf{A}_\pi^b, \mathbf{b}_\pi \rangle$, i.e., $\mathcal{H}_\pi = \mathcal{G}_\pi(\mathcal{X}) \triangleq \{[\boldsymbol{x}^\top \ \boldsymbol{u}^\top]^\top \mid \boldsymbol{u} = \boldsymbol{\pi}(\boldsymbol{x}), \boldsymbol{x} \in \mathcal{X}\}$. Let $\overline{\mathcal{H}}_f(\mathcal{X},\mathcal{U}) = \langle \mathbf{G}_f^c, \mathbf{G}_f^b, \mathbf{c}_f, \mathbf{A}_f^c, \mathbf{A}_f^b, \mathbf{b}_f \rangle$ be the computed HZ envelope of the nonlinear function $\boldsymbol{f}$ over the domain $\mathcal{X} \times \mathcal{U}$ using the approach presented in Section III-A, i.e. $\overline{\mathcal{H}}_f(\mathcal{X},\mathcal{U}) \supseteq \mathcal{G}_f(\mathcal{X},\mathcal{U})$.

With notations above, the following theorem provides the closed-form identities of an over-approximation of the one-step BRS, $\overline{\mathcal{P}}(\mathcal{T})$, with a given target set represented by an HZ. The proof of this theorem is given in [27].

Theorem 1: Given the state set $\mathcal{X}$, the input set $\mathcal{U}$, and a target set represented by an HZ $\mathcal{T} = \langle \mathbf{G}_\tau^c, \mathbf{G}_\tau^b, \mathbf{c}_\tau, \mathbf{A}_\tau^c, \mathbf{A}_\tau^b, \mathbf{b}_\tau \rangle \subset \mathbb{R}^n$, the following HZ is an over-approximated one-step BRS of the NFS (4):

$$\overline{\mathcal{P}}(\mathcal{T}) = \langle \mathbf{G}_p^c, \mathbf{G}_p^b, \mathbf{c}_p, \mathbf{A}_p^c, \mathbf{A}_p^b, \mathbf{b}_p \rangle \tag{5}$$

where $\mathbf{c}_p = \mathbf{c}_\pi[1:n,:]$, $\mathbf{G}_p^c = \begin{bmatrix} \mathbf{G}_\pi^c[1:n,:] & \mathbf{0} & \mathbf{0} \end{bmatrix}$, $\mathbf{G}_p^b = \begin{bmatrix} \mathbf{G}_\pi^b[1:n,:] & \mathbf{0} & \mathbf{0} \end{bmatrix}$ and

$$\mathbf{A}_p^c = \begin{bmatrix} \mathbf{A}_\pi^c & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_f^c & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_\tau^c \\ \mathbf{G}_\pi^c & -\mathbf{G}_f^c[1:n+m,:] & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_f^c[n+m+1:2n+m,:] & -\mathbf{G}_\tau^c \end{bmatrix},$$

C. Refinement of Over-approximated Multi-step BRS

Based on the results of the preceding subsections, the -step over-approximations of BRS for (4) can be computed iteratively as follows:

$$\overline{\quad} \quad \overline{\quad} \quad \overline{\quad}\,\overline{\quad} \tag{6}$$

Since Theorem 1 computes $\overline{\quad}$ over the entire state-input domain , a naive application of (6) based on Theorem 1 may lead to conservative approximations for multi-step computations. In order to mitigate the conservativeness, we introduce a refinement procedure aimed at computing $\overline{\quad}$ within more tightly constrained prior sets, denoted as where stands for "prior". Such procedure allows the nonlinearity to be approximated over tighter sets, thus leading to less conservative results. Concretely, the refinement procedure employs a recursive approach that utilizes the BRSs obtained from the previous $\overline{\quad}$ to compute the new prior sets.

With the same notations as in Theorem 1, the following lemma provides a closed-form expression of an over-approximation of BRS $\overline{\quad}$ over the prior sets . The proof of this lemma is given in [27].

Lemma 1: Given the state set , the input set , the target set , and the graph set of , , suppose that the prior sets satisfies

(7)

Let $\overline{\quad}$ be the over-approximated graph set of the nonlinear function over the domain . Then, the one-step BRS of the NFS (4) can be over-approximated by an HZ whose *HCG*-representation is the same as (5) by replacing with .

Algorithm 1 summarizes the over-approximation of BRS for the closed-loop system (4) with the refinement procedure. The prior sets for all the time horizons are initialized as (line 1) and is computed over with the large scalars and by using Algorithm 1 in [12] (line 2). For each time horizon , we compute the over-approximation of the nonlinear plant model over the prior set and $\overline{\quad}$ (line 5) and then the over-approximated one-step BRS $\overline{\quad}$ is computed by Lemma 1 (line 6). If or the refinement epoch reaches the maximum number , the refinement procedure will be skipped and the computed BRSs $\overline{\quad}$ for to will be returned; otherwise, the tighter prior sets are computed based on the over-approximated BRSs and then passed to the next refinement epoch (line 9). Concretely, in line 9, the bounding box of $\overline{\quad}$, is computed by the and is set to be the prior sets for the refinement procedure. The only shrinks the bounding intervals of over the dimensions contributing to the nonlinearity. Let . Such shrinking interval, denoted as , can be obtained by solving two Mixed-Integer linear Programs (MILPs): , .

Algorithm 1: BReachNonlin-HZ

Input: State domain , control input domain , target set , nonlinear plant model , neural network controller , large scalars , time horizon , the number of refinement epochs .

Output: Refined over-approximations of -step BRSs from $\overline{\quad}$ to $\overline{\quad}$.

1 **Initialization**: ; ; ;
2 Compute with and ; `// Alg 1 in [12]`
3 **for** **do**
4 **for** **do**
5 Compute $\overline{\quad}$; `// Section III-A`
6 $\overline{\quad}$ Pre $\overline{\quad}$; `// Over-approximate one-step BRS`
7 **if** **or** **then**
8 **continue**;
9 , $\overline{\quad}$; `// The prior sets for .`
10 **return** $\overline{\quad}$

Remark 1: The number of refinement epochs in Algorithm 1 represents the trade-off between the set approximation accuracy and the computation efficiency. Specifically, the refinement procedure mitigates the conservativeness of the set approximation but requires computing $\overline{\quad}$ and the bounding hyper-boxes of HZs at each epoch, which introduces extra computation time. Simulation results for different numbers of epochs are given in Section V.

Remark 2: The over-approximated BRSs can be used for the safety verification of NFSs [12], [17]. Consider an HZ initial state set and an HZ unsafe region . Suppose that the -step BRS of is over-approximated as $\overline{\quad}$ via Algorithm 1 for . Then any trajectory that starts from will not enter into the unsafe region within time steps if all the over-approximated BRSs $\overline{\quad}$ have no intersections with the initial set . By [17, Proposition 7] and [12, Lemma 1], verifying the emptiness of the intersection of and $\overline{\quad}$ is equivalent to solve an MILP.

IV. Neural Feedback System With General Types of Activation Functions

In this section, we extend the results of the preceding section that considers ReLU-activated FNNs to FNNs with more general activation functions. We categorize the activation

functions as either piecewise linear or non-piecewise linear. For FNNs with piecewise linear activation functions, their *exact* graph sets can be constructed, in which case Theorem 1 and Lemma 1 are directly applicable. For FNNs with non-piecewise linear activation functions, their graph sets can be over-approximated by using the methods in Section III-A, based on which the over-approximated BRSs can be computed.

A. Piecewise Linear Activation Functions

The graph of piecewise linear activation functions can be exactly represented by a finite union of polytopes that is equivalent to an HZ. For simplicity, we use the Parametric ReLU (PReLU) activation function as an example to show the construction of the graph set.

The PReLU is a parametric activation function defined as

$$\text{if}$$
$$\text{if}$$

Here is a parameter that is learned along with other neural network parameters [28]. When , it becomes the conventional ReLU; when is a small positive number, it becomes the Leaky ReLU. Similar to the construction of HZ for ReLU in [12], given an interval domain , the two line segments of PReLU can be exactly represented as two HZs: , The union of and can be directly computed as a single HZ, , using Proposition 1 in [18]. Applying Proposition 3 in [8] we can remove the redundant continuous generators to get

(8)

where

Since (8) exactly represents PReLU over an interval domain, Algorithm 1 in [12] can be directly applied to PReLU-activated FNNs, and based on that, the exact graph set of FNNs can be computed. Hence, Theorem 1, Lemma 1, and Algorithm 1 can be directly employed to over-approximate BRSs for a PReLU-activated NFS (4).

B. Non-Piecewise Linear Activation Functions

In addition to piecwise linear activation functions, non-piecewise linear activation functions such as sigmoid and tanh functions, are also widely employed in NNs [29]. Since the graph of non-piecewise linear functions cannot be represented exactly by HZs, we will construct an over-approximation of the graph set of FNNs .

Since an activation function is a scalar function, we use the methods discussed in Section III-A to construct an HZ over the interval domain such that

(9)

The following lemma gives the HZ over-approximation of the graph of vector-valued activation function that is constructed by component-wise repetition of the scalar activation function over a domain represented as an HZ. The proof of this lemma is given in [27].

Lemma 2: Assume that a domain is represented as an HZ such that . Let be the graph of the vector-valued activation function , i.e., . Then can be over-approximated by an HZ, denoted as , where

(10)

is a permutation matrix, and is given in (9).

Lemma 2 provides the transformation from the scalar-valued nonlinear activation function that is over-approximated based on Section III-A to vector-valued activation function , which is more favorable for layer-by-layer operations when constructing the graph set of FNNs.

Denote the output set of Algorithm 1 in [12] by replacing with as . The following theorem shows that is an over-approximation of the graph set of FNN over the HZ-represented domain set . The proof of this theorem is given in [27].

Theorem 2: Given an -layer FNN with non-piecewise linear activation functions and the domain set represented as an HZ , the output set is an over-approximation of the graph set of FNN , i.e., .

Theorem 2 extends our previous work [12] from the exact graph set of ReLU-activated FNN to the over-approximation of the graph set of FNN with more general activation functions. Let be the over-approximated graph set of FNN over the domain using Theorem 2, i.e., . With the same notations in Lemma 1, the following theorem provides the closed-form expression of an over-approximation of the one-step BRS for NFS (4) with non-piecewise activation functions. The proof of this theorem is given in [27].

Theorem 3: Given the state set , the input set , the target set , the prior sets , and the over-approximated graph set of given as , the one-step BRS of the NFS (4) can be over-approximated by an HZ whose *HCG*-representation is the same as (5) by replacing and with and , respectively.

V. Simulation Results

In this section, we use two simulation examples to illustrate the performance of the proposed method. The considered NFS consists of a Duffing Oscillator and an FNN controller with either ReLU (Example 1) or tanh (Example

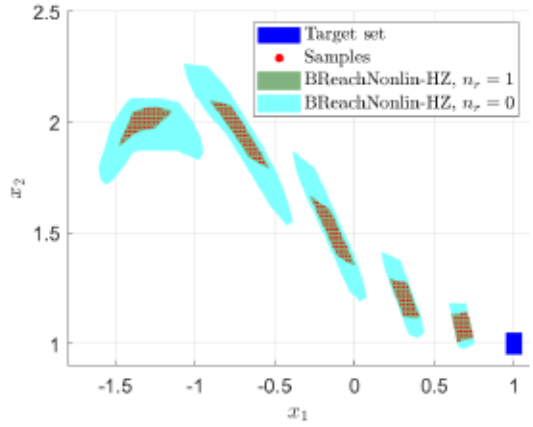
2) activation functions. The simulation examples are implemented in MATLAB R2022a and executed on a desktop with an Intel Core i9-12900k CPU and 32GB of RAM.

Example 1: Consider the following discrete-time Duffing Oscillator model from [30]:

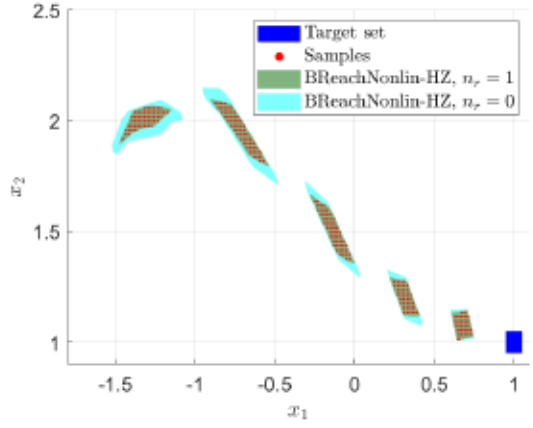
$$x_1(t+1) = x_1(t) + 0.3x_2(t),$$
$$x_2(t+1) = 0.3x_1(t) + 0.82x_2(t) - 0.3x_1(t)^3 + 0.3u(t),$$

where x_1, $x_2 \in \mathbb{R}$ are the states confined in the state set $\mathcal{X} = [\![-2, 1.1]\!] \times [\![-2, 3]\!]$ and $u \in \mathcal{U} = [\![0, 5]\!]$ is the control input. The controller u is a two-layer ReLU-activated FNN with 10-5 hidden neurons that is trained to learn the control policy given in [30]. The exact BRSs are obtained by selecting samples from which trajectories can reach the target set within T time steps; these samples are chosen from uniformly distributed grids over $\mathcal{X}$. To the best of our knowledge, there is no existing set-propagation-based method that can compute the over-approximated BRS for the system shown above.

Given the target set $\mathcal{T} = [\![0.95, 1.05]\!] \times [\![0.95, 1.05]\!]$, Algorithm 1 and Theorem 1 can be employed to over-approximate the BRSs since the $\mathcal{X}$, $\mathcal{U}$ and $\mathcal{T}$ are HZs, satisfying all the conditions of Theorem 1. Note that we use 10 breakpoints



(a) Over-approximated BRSs based on SOS.



(b) Over-approximated BRSs based on OVERT.

Fig. 2: Over-approximated BRSs without refinement (cyan), the refined over-approximated BRSs (dark green), and samples (red) from the true BRSs for Example 1.

to over-approximate the nonlinear term x_1^3 via SOS and OVERT and then $\overline{\mathcal{H}}_f(\mathcal{X}_t^p, \mathcal{U}_t^p)$ is constructed by following a similar procedure as the example in [20]. Fig. 2 shows the over-approximated and exact BRSs, indicating that the BRSs computed by Algorithm 1 over-approximates the exact BRSs, as shown in Theorem 1, and the conservativeness is greatly mitigated by the refinement procedure. In addition, the computed BRSs by using OVERT are less conservative than those by using SOS.

Fig. 3a shows the set complexity of the t-step BRS via SOS and OVERT for $t = 1, 2, \cdots, 8$. Note that the BRSs and refined BRSs have the same set complexity. The number of generators and equality constraints has a linear growth rate with respect to time steps. In addition, the number of binary generators of BRSs is identical due to the same number of breakpoints used in the OVERT method and the SOS method. Fig. 3b and Fig. 3c show the computational time and the approximation error for five-step BRS with different numbers of refinement epochs n_r. The approximation error is computed by the ratio between the volumes of the over-approximated BRS and exact BRS as error $= \frac{V_{\text{over}} - V_{\text{exact}}}{V_{\text{over}}}$, where V_{over} and V_{exact} are the volume of over-approximation of last-step BRS and exact BRS respectively, which are approximated by the Monte Carlo method. It can be observed that the computation time increases almost linearly when $n_r \geq 1$ because the refinement procedure does not affect the set complexity of the computed BRSs, and in addition, the approximation error is converged after just one epoch in this example.

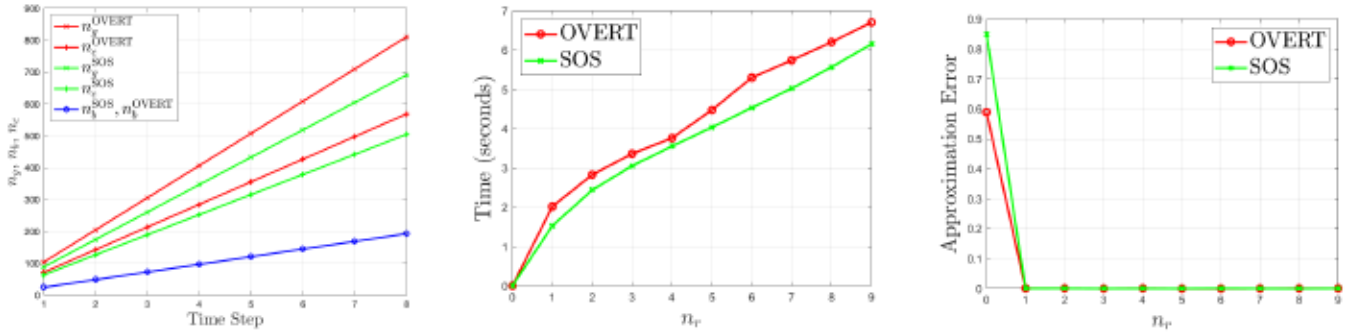
Example 2: Consider the same Duffing Oscillator model in Example 1. Instead of the ReLU-activated FNN controller, we train a tanh-activated FNN with 5-5 hidden neurons to learn the same control policy given in [30]. We use OVERT with 7 breakpoints to construct the over-approximation of the graph set of FNN $\overline{\mathcal{H}}_\pi$ based on Theorem 2, and then compute the over-approximated BRSs with the same configurations as those in Example 1 based on Theorem 3. The simulation results are shown in Figure 4. It is observed that all the samples are enclosed by the over-approximated BRSs, consistent with Theorem 3.

VI. Conclusion

In this paper, we proposed a novel HZ-based approach to over-approximate BRSs of NFSs with nonlinear plant models. For NFSs with ReLU activation functions, by combining techniques that over-approximate nonlinear functions with the exact graph set of ReLU-activated FNNs, we showed the closed-form of over-approximated BRSs and a refinement procedure to reduce the conservativeness. In addition, we extended the results to NFSs with piecewise and non-piecewise linear activation functions. The performance of the proposed approach was evaluated using two numerical examples.

References

[1] X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial examples: Attacks and defenses for deep learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2805–2824, 2019.



(a) Set complexities. (b) Computation time. (c) Approximation error.

Fig. 3: Comparison of the SOS and OVERT approximation methods at each refinement epoch n_r for Example 1.

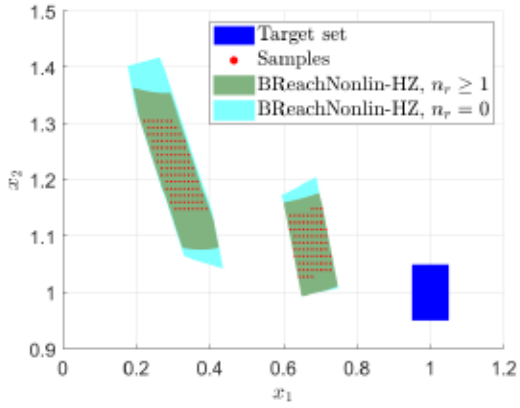


Fig. 4: Over-approximation of BRSs for Example 2. The approximation error cannot be further reduced when $n_r > 1$.

[2] S. Dutta, X. Chen, and S. Sankaranarayanan, "Reachability analysis for neural feedback systems using regressive polynomial rule inference," in *Proceedings of the 22nd ACM International Conference on Hybrid Systems: Computation and Control*, 2019, pp. 157–168.

[3] C. Huang, J. Fan, W. Li, X. Chen, and Q. Zhu, "ReachNN: Reachability analysis of neural-network controlled systems," *ACM Transactions on Embedded Computing Systems*, vol. 18, no. 5s, pp. 1–22, 2019.

[4] M. Everett, G. Habibi, C. Sun, and J. P. How, "Reachability analysis of neural feedback loops," *IEEE Access*, vol. 9, pp. 163 938–163 953, 2021.

[5] 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 *32nd International Conference on Computer-Aided Verification*. Springer, 2020, pp. 3–17.

[6] 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, 2022.

[7] Y. Zhang and X. Xu, "Safety verification of neural feedback systems based on constrained zonotopes," in *IEEE 61st Conference on Decision and Control*, 2022, pp. 2737–2744.

[8] ——, "Reachability analysis and safety verification of neural feedback systems via hybrid zonotopes," in *American Control Conference*. IEEE, 2023, pp. 1915–1921.

[9] N. Kochdumper, C. Schilling, M. Althoff, and S. Bak, "Open-and closed-loop neural network verification using polynomial zonotopes," in *NASA Formal Methods Symposium*. Springer, 2023, pp. 16–36.

[10] J. A. Vincent and M. Schwager, "Reachable polyhedral marching (RPM): A safety verification algorithm for robotic systems with deep neural network components," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 9029–9035.

[11] N. Rober, S. M. Katz, C. Sidrane, E. Yel, M. Everett, M. J. Kochenderfer, and J. P. How, "Backward reachability analysis of neural feedback loops: Techniques for linear and nonlinear systems," *IEEE Open Journal of Control Systems*, 2023.

[12] Y. Zhang, H. Zhang, and X. Xu, "Backward reachability analysis of neural feedback systems using hybrid zonotopes," *IEEE Control Systems Letters*, vol. 7, pp. 2779–2784, 2023.

[13] I. Mitchell, A. Bayen, and C. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 947–957, 2005.

[14] L. Yang, H. Zhang, J.-B. Jeannin, and N. Ozay, "Efficient backward reachability using the Minkowski difference of constrained zonotopes," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 11, pp. 3969–3980, 2022.

[15] N. Rober, M. Everett, and J. P. How, "Backward reachability analysis for neural feedback loops," in *IEEE 61st Conference on Decision and Control*, 2022, pp. 2897–2904.

[16] C. Sidrane, A. Maleki, A. Irfan, and M. J. Kochenderfer, "OVERT: An algorithm for safety verification of neural network control policies for nonlinear systems," *Journal of Machine Learning Research*, vol. 23, no. 117, pp. 1–45, 2022.

[17] T. J. Bird, H. C. Pangborn, N. Jain, and J. P. Koeln, "Hybrid zonotopes: A new set representation for reachability analysis of mixed logical dynamical systems," *Automatica*, vol. 154, p. 111107, 2023.

[18] T. J. Bird and N. Jain, "Unions and complements of hybrid zonotopes," *IEEE Control Systems Letters*, vol. 6, pp. 1778–1783, 2021.

[19] J. Ortiz, A. Vellucci, J. Koeln, and J. Ruths, "Hybrid zonotopes exactly represent ReLU neural networks," *arXiv:2304.02755*, 2023.

[20] J. A. Siefert, T. J. Bird, J. P. Koeln, N. Jain, and H. C. Pangborn, "Successor sets of discrete-time nonlinear systems using hybrid zonotopes," in *American Control Conference*. IEEE, 2023, pp. 1383–1389.

[21] ——, "Reachability analysis of nonlinear systems using hybrid zonotopes and functional decomposition," *arXiv:2304.06827*, 2023.

[22] T. J. Bird, "Hybrid zonotopes: A mixed-integer set representation for the analysis of hybrid systems," *Purdue University Graduate School*, 2022.

[23] G. P. McCormick, "Computability of global solutions to factorable nonconvex programs: Part I—Convex underestimating problems," *Mathematical programming*, vol. 10, no. 1, pp. 147–175, 1976.

[24] S. Caratzoulas and C. Floudas, "Trigonometric convex underestimator for the base functions in fourier space," *Journal of optimization theory and applications*, vol. 124, no. 2, pp. 339–362, 2005.

[25] E. Wanufelle, "A global optimization method for mixed integer nonlinear nonconvex problems related to power systems analysis," *Ph. D. dissertation*, 2007.

[26] E. Beale and J. Tomlin, "Special facilities in a general mathematical programming system for nonconvex problems using ordered sets of variables," *Operational Research*, vol. 69, pp. 447–454, 01 1969.

[27] H. Zhang, Y. Zhang, and X. Xu, "Hybrid zonotope-based backward reachability analysis for neural feedback systems with nonlinear system models," *arXiv:2310.06921*, 2024.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1026–1034.

[29] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, 2022.

[30] T. Dang and R. Testylier, "Reachability analysis for polynomial dynamical systems using the Bernstein expansion." *Reliabable Computing*, vol. 17, no. 2, pp. 128–152, 2012.