Real-Time Safe Control of Neural Network Dynamic Models with Sound Approximation

Hanjiang Hu HANJIANGHU@CMU.EDU
Robotics Institute, Carnegie Mellon University

Jianglin Lan JIANGLIN.LAN@GLASGOW.AC.UK
James Watt School of Engineering, University of Glasgow

Changliu Liu CLIU6@ANDREW.CMU.EDU
Robotics Institute, Carnegie Mellon University

Editors: A. Abate, K. Margellos, A. Papachristodoulou

Abstract

Safe control of neural network dynamic models (NNDMs) is important to robotics and many applications. However, it remains challenging to compute an optimal safe control in real time for NNDM. To enable real-time computation, we propose to use a sound approximation of the NNDM in the control synthesis. In particular, we propose Bernstein over-approximated neural dynamics (BOND) based on the Bernstein polynomial over-approximation (BPO) of ReLU activation functions in NNDM. To mitigate the errors introduced by the approximation and to ensure persistent feasibility of the safe control problems, we synthesize a worst-case safety index using the most unsafe approximated state within the BPO relaxation of NNDM offline. For the online real-time optimization, we formulate the first-order Taylor approximation of the nonlinear worst-case safety constraint as an additional linear layer of NNDM with the ℓ_2 bounded bias term for the higher-order remainder. Comprehensive experiments with different neural dynamics and safety constraints show that with safety guaranteed, our NNDMs with sound approximation are 10-100 times faster than the safe control baseline that uses mixed integer programming (MIP), validating the effectiveness of the worst-case safety index and scalability of the proposed BOND in real-time large-scale settings.

Keywords: safe control, neural network dynamic model, Bernstein polynomial

1. Introduction

Safety is crucial to robotic systems. Safe control of dynamic systems has been well studied in literature (Nagumo, 1942; Blanchini, 1999). A safe control law maintains the states within the user-defined safety set by ensuring forward invariance and finite-time convergence, *i.e.* states remaining in it once entering; and returning to it in finite time steps once leaving. Although safe control laws can be designed for control-affine systems (Liu and Tomizuka, 2014; Wei and Liu, 2019; Agrawal and Panagou, 2021), it is challenging to construct an exact analytical model for complex real-world systems. Progress in deep neural networks has boosted learning-based methods to model the complicated dynamics (Nagabandi et al., 2018; Janner et al., 2019). However, these neural network dynamic models (NNDMs) have limited mathematical interpretability, making it difficult to design subsequent safe control laws.

This paper considers the safe tracking problem using NNDMs. Recent works (Wei and Liu, 2022; Liu et al., 2023; Li et al., 2023b) model the safe tracking problem as a constrained optimization problem by minimizing the state tracking error for the given system dynamics constraint while obeying the safety constraint. Since it is challenging to find optimal tracking control for these highly nonlinear black-box NNDMs through model inverse (Tolani et al., 2000) or shooting methods, researchers resort to mixed integer programming (MIP) to find the optimal tracking control (Wei and Liu, 2022; Liu et al., 2023; Li et al., 2023b), a method widely used in neural network verification. However, MIP is well-known for its poor time efficiency and limited scalability in the literature on neural network verification (Liu et al., 2021; Li et al., 2023a), making these complete MIP-based methods hardly applicable in real-time safety-critical robot applications.

To this end, we propose Bernstein over-approximated neural dynamics (BOND) with sound approximation of ReLU activation layers in NNDM to greatly speed up the computation of safe tracking problems. Specifically, inspired by sound verification of neural networks (Fatnassi et al., 2023; Huang et al., 2022; Khedr and Shoukry, 2023), we leverage Bernstein polynomial over-approximation (BPO) to address the nonlinearity of the activation function, replacing integer variables with inequality constraints for NNDMs in the safe tracking optimization. To deal with the approximation error caused by BPO and ensure persistent feasibility under safety constraints, we synthesize the worst-case safety index offline to make the optimization problem feasible even for the most potentially unsafe state of BOND, and linearize the safety constraint with a linear Taylor layer in the online optimization. The contributions are listed as below:

- We propose a sound approximation for NNDM using Bernstein polynomial over-approximation to optimize real-time safe tracking problems efficiently.
- We synthesize the worst-case safety index to ensure the persistent feasibility under approximation error caused by the over-approximation of NNDMs.
- Extensive experiments validate that BOND is 10-100 times faster and more scalable than MIP-based baseline in real-time collision avoidance and safe following with different NNDMs.

The remaining paper is organized as follows: Section 2 provides a problem formulation of neural network dynamics, Bernstein polynomial over-approximation and safe tracking problem. Section 3 describes the proposed method including worst-case safety index synthesis and linearization for online optimization. Section 4 presents the experimental results with ablation study. Section 5 concludes the paper and discusses potential future directions.

2. Formulation

2.1. Background of Safe Tracking with Neural Network Dynamic Models

Neural network dynamic models (NNDMs). Denote a discrete-time NNDM with state $\mathbf{x}_k$ and control $\mathbf{u}_k$ at time step k as

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)dt, \;\; \mathbf{x}_k \in \mathcal{X} = \mathbb{R}^{m_x}, \;\; \mathbf{u}_k \in \mathcal{U} \subset \mathbb{R}^{m_u} \tag{1}$$

where $\mathcal{U}$ is defined by linear constraints and dt is the sampling time. $\mathbf{f} : \mathbb{R}^{m_x} \times \mathbb{R}^{m_u} \mapsto \mathbb{R}^{m_x}$ is the dynamic model parameterized by n-layer feedforward neural networks with nonlinear activation functions, *i.e.* $\mathbf{f} = \mathbf{f}_n \circ \mathbf{f}_{n-1} \circ \cdots \circ \mathbf{f}_1$, where $\mathbf{f}_i : \mathbb{R}^{k_{i-1}} \mapsto \mathbb{R}^{k_i}$ is the ith linear mapping layer with a nonlinear activation $\boldsymbol{\sigma}_i : \mathbb{R}^{k_i} \mapsto \mathbb{R}^{k_i}$ over the k_i-dimensional hidden variable in layer i. More concretely, by denoting the weight matrix and bias vector in layer i as $\mathbf{W}_i \in \mathbb{R}^{k_i \times k_{i-1}}$ and $\mathbf{b}_i \in \mathbb{R}^{k_i}$,

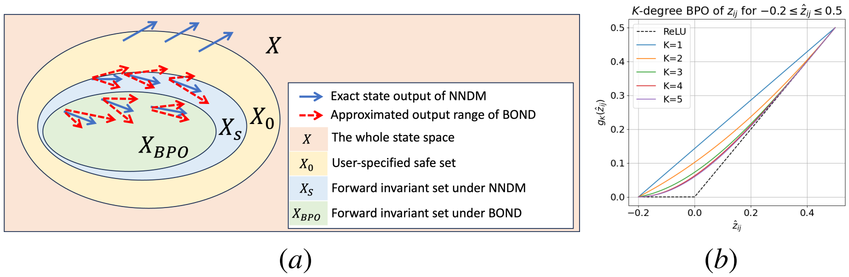

X
X_0
X_s
X_BPO
Exact state output of NNDM
Approximated output range of BOND
X The whole state space
X_0 User-specified safe set
X_s Forward invariant set under NNDM
X_BPO Forward invariant set under BOND
K-degree BPO of z_ij for −0.2 ≤ ẑ_ij ≤ 0.5
ReLU
K=1
K=2
K=3
K=4
K=5
g_K(ẑ_ij)
ẑ_ij
0.5
0.4
0.3
0.2
0.1
0.0
−0.2
0.0
0.2
0.4
(a)
(b)

Figure 1: (a) Illustration of the forward invariant sets under exact NNDM and BOND. (b) Approximation of the ReLU function using BPO of different degrees.

the hidden variable after the layer i is $\mathbf{z}_i = \mathbf{f}_i(\mathbf{z}_{i-1}) = \boldsymbol{\sigma}_i(\hat{\mathbf{z}}_i)$, where $\hat{\mathbf{z}}_i = \mathbf{W}_i\mathbf{z}_{i-1} + \mathbf{b}_i$ is the pre-activation variable. Specifically, it trivially holds that $\mathbf{z}_0 = [\mathbf{x}_k^\top, \mathbf{u}_k^\top]^\top, k_0 = m_x + m_u, k_n = m_x$. Let $\mathbf{w}_{ij} \in \mathbb{R}^{1\times k_{i-1}}$ be the jth row of $\mathbf{W}_i$ and b_{ij} be the jth entry of $\mathbf{b}_i$, so the jth entry of $\hat{\mathbf{z}}_i$ is calculated as $\hat{z}_{ij} = \mathbf{w}_{ij}\mathbf{z}_{i-1} + b_{i,j}$. We only focus on ReLU activation in this work, so for the jth entry of $\mathbf{z}_i$, we have that $z_{ij} = \sigma_i(\hat{z}_{ij}) = \max\{0, \hat{z}_{ij}\}$.

Optimization problem for tracking. Similar to Wei and Liu (2022), we focus on the tracking problem with the NNDM as a one-step model predictive control (MPC), optimizing control action $\mathbf{u}_k$ via minimizing the ℓ_p distance between the predicted next state $\mathbf{x}_{k+1}$ and the reference next state $\mathbf{x}_{k+1}^r$ (known ahead of time) at each time step k, which is shown as follows:

$$\begin{aligned}\min_{\mathbf{u}_k,\mathbf{x}_{k+1}} \quad & \|\mathbf{x}_{k+1} - \mathbf{x}_{k+1}^r\|_p \\ s.t.\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)dt, \quad \mathbf{u}_k \in \mathcal{U}.\end{aligned} \tag{2}$$

where $\|\cdot\|_p$ can be either ℓ_1-norm as a linear objective or ℓ_2-norm as a quadratic objective. Under the nonlinear constraint of NNDM with ReLU activation, the optimization problem (2) is challenging to solve using existing solvers (Wei and Liu, 2022; Liu et al., 2023).

Safety specification and constraint. In addition to the NNDM constraint in (2), the safety constraint is also indispensable for the safe tracking problem (Wei and Liu, 2022). We denote the user-specified safe set $\mathcal{X}_0$ as a connected and closed set in the state space, which can be defined as a zero-sublevel set of a continuous and differentiable function, *i.e.* $\mathcal{X}_0 = \{\mathbf{x} \in \mathcal{X} \mid \phi_0(\mathbf{x}) \leq 0\}$. If the system is already in a safe state, we should ensure forward invariance, *i.e.* $\phi_0(\mathbf{x}_k) \leq 0 \implies \phi_0(\mathbf{x}_{k+1}) \leq 0$. If the system is currently unsafe, we should ensure finite-time convergence, *i.e.* $\phi_0(\mathbf{x}_k) > 0 \implies \phi_0(\mathbf{x}_{k+1}) \leq \phi_0(\mathbf{x}_k) - \gamma dt$, so that the system will go back to the safe set within finite time steps $\phi_0(\mathbf{x}_k)/\gamma dt$ with constant γ. We combine these two constraints at step k as follows,

$$\mathcal{A}(\mathcal{X}_0, \mathbf{x}_k, \gamma) := \{\mathbf{x}_{k+1} \mid \phi_0(\mathbf{x}_{k+1}) \leq \max\{0, \phi_0(\mathbf{x}_k) - \gamma dt\}, \text{ with } \mathcal{X}_0 = \{\mathbf{x} \mid \phi_0(\mathbf{x}) \leq 0\}. \tag{3}$$

However, there may not always exist a feasible control input that results in $\mathbf{x}_{k+1} \in \mathcal{A}(\mathcal{X}_0, \mathbf{x}_k, \gamma)$. If such control always exists, we say the safe tracking problem is *persistently feasible*. To achieve persistent feasibility, the common practice is to find a subset of the safe set $\mathcal{X}_S \subset \mathcal{X}_0$ using safety index synthesis (SIS) (Wei and Liu, 2022) such that there always exist a control that ensures $\mathbf{x}_{k+1} \in \mathcal{A}(\mathcal{X}_S, \mathbf{x}_k, \gamma)$. Then $\mathcal{A}(\mathcal{X}_S, \mathbf{x}_k, \gamma)$ will be used as a constraint in (2) to ensure safety.

2.2. Safe Tracking with Bernstein Over-approximated Neural Dynamics (BOND)

Bernstein polynomial over-approximation for NNDMs. Inspired by neural network verification with Bernstein polynomial over-approximation (BPO) (Fatnassi et al., 2023; Huang et al., 2022;

Khedr and Shoukry, 2023), we adopt the following K-order BPO for the ReLU activation $z_{ij} = \sigma_i(\hat{z}_{ij})$ based on the bounded pre-activation values $\hat{l}_{ij} \leq \hat{z}_{ij} \leq \hat{u}_{ij}, \hat{u}_{ij} \neq \hat{l}_{ij}$:

$$z_{ij} \leq g_K(\hat{z}_{ij}) = \sum_{k=0}^{K} \max\{0, \frac{k}{K}(\hat{u}_{ij} - \hat{l}_{ij}) + \hat{l}_{ij}\} \cdot \binom{K}{k} \frac{(\hat{u}_{ij} - \hat{z}_{ij})^{K-k}(\hat{z}_{ij} - \hat{l}_{ij})^k}{(\hat{u}_{ij} - \hat{l}_{ij})^K}, \tag{4}$$

where the pre-activation bounds $\hat{l}_{ij}$ and $\hat{u}_{ij}$ can be found through interval arithmetic (IA) methods or dual networks (Liu et al., 2021; Wong and Kolter, 2018). When $\hat{u}_{ij} = \hat{l}_{ij}$, it trivially holds that $z_{ij} = \sigma_i(\hat{z}_{i,j}) = \sigma_i(\hat{l}_{i,j}) = \sigma_i(\hat{u}_{i,j})$ so the approximation is not needed. Note that when $K = 1$, BPO relaxation is degraded to the triangle relaxation (also called LP relaxation) (Wong et al., 2018; Tjeng et al., 2018; Ehlers, 2017). The nonlinear higher-degree BPOs are visualized in Fig. 1 (b).

Conservative forward invariant set with BPO. Combining BPO (4) with the linear under approximation for ReLU activation at each node, $z_{ij} \geq 0, z_{ij} \geq \hat{z}_{ij}$, the output of the dynamic model $\mathbf{f}(\mathbf{x}, \mathbf{u})$ can be lower-bounded by a function $\underline{\mathbf{f}}_B$ and upper-bounded by a function $\overline{\mathbf{f}}_B$, *i.e.* $\underline{\mathbf{f}}_B(\mathbf{x}, \mathbf{u}) \leq \mathbf{f}(\mathbf{x}, \mathbf{u}) \leq \overline{\mathbf{f}}_B(\mathbf{x}, \mathbf{u})$, relaxing the predicted output from the exact $\mathbf{f}(\mathbf{x}, \mathbf{u})$ to the range $[\underline{\mathbf{f}}_B(\mathbf{x}, \mathbf{u}), \overline{\mathbf{f}}_B(\mathbf{x}, \mathbf{u})]$. Therefore, to ensure persistent feasibility of the safety constraints w.r.t the unknown output $\mathbf{f}(\mathbf{x}, \mathbf{u}) \in [\underline{\mathbf{f}}_B(\mathbf{x}, \mathbf{u}), \overline{\mathbf{f}}_B(\mathbf{x}, \mathbf{u})]$, we need to find an even more conservative forward invariant set $\mathcal{X}_{BPO} \subseteq \mathcal{X}_S$, which is illustrated in Fig. 1 (a) and will be introduced in section 3.

Safe tracking with BOND. With the BPO relaxation of NNDM and more conservative forward invariant set $\mathcal{X}_{BPO}$, the tracking problem aims to optimize both the control strategy $\mathbf{u}_k$ and the hallucinated next state $\mathbf{x}_{k+1}$ within the approximated output range $[\underline{\mathbf{f}}_B(\mathbf{x}, \mathbf{u}), \overline{\mathbf{f}}_B(\mathbf{x}, \mathbf{u})]$ by minimizing a linear ($p = 1$) or quadratic ($p = 2$) objective of the distance between $\mathbf{x}_{k+1}$ and the reference state $\mathbf{x}^r_{k+1}$ whilst satisfying the safe control constraint that any possible future state should belong to $\mathcal{X}_{BPO}$. So the safe tracking problem with BOND $\mathbf{f}_B(\cdot)$ is formulated as the following constrained optimization problem at every time step k:

$$\min_{\mathbf{u}_k \in \mathcal{U}, \mathbf{x}_{k+1}} \|\mathbf{x}_{k+1} - \mathbf{x}^r_{k+1}\|_p \tag{5a}$$

$$s.t.\ \mathbf{x}_k + \underline{\mathbf{f}}_B(\mathbf{x}_k, \mathbf{u}_k)dt \leq \mathbf{x}_{k+1} \leq \mathbf{x}_k + \overline{\mathbf{f}}_B(\mathbf{x}_k, \mathbf{u}_k)dt \tag{5b}$$

$$\tilde{\mathbf{x}}_{k+1} \in \mathcal{A}(\mathcal{X}_{BPO}, \mathbf{x}_k, \gamma), \forall \tilde{\mathbf{x}}_{k+1} \in [\mathbf{x}_k + \underline{\mathbf{f}}_B(\mathbf{x}_k, \mathbf{u}_k)dt, \mathbf{x}_k + \overline{\mathbf{f}}_B(\mathbf{x}_k, \mathbf{u}_k)dt]. \tag{5c}$$

Note that $\mathcal{X}_S$ from Wei and Liu (2022) will not work in (5c) as it does not take the approximation into consideration. In the following section, we first discuss how to obtain $\mathcal{X}_{BPO}$ offline; and then discuss how to efficiently compute the optimization problem (5) online.

3. Method

3.1. Worst-Case Safety Index Synthesis

To characterize the forward invariant set $\mathcal{X}_{BPO}$ and $\mathcal{A}(\mathcal{X}_{BPO}, \mathbf{x}_k, \gamma)$, similar to (3), we propose to synthesize a worst-case safety index ϕ so that $\mathcal{X}_{BPO} = \{\mathbf{x} \in \mathcal{X} \mid \phi(\mathbf{x}) \leq 0\}$ and (5c) is persistently feasible. Following the evolutionary strategy-based safety index synthesis in Wei and Liu (2022), we parameterize the safety index with $\alpha_i, \beta \in \mathbb{R}, i = 1, 2, \ldots, q$ as $\phi(\mathbf{x}) = \phi_0^*(\boldsymbol{\alpha}_0, \mathbf{x}) + \sum_{i=1}^{q} \alpha_i \phi_0^{(i)}(\mathbf{x}) + \beta$, where $\phi_0^*(\boldsymbol{\alpha}_0, \mathbf{x})$ is consistent with user-specified sublevel set ϕ_0 and $\phi_0^{(i)}(\mathbf{x})$

is denoted as each $i$the order derivative of ϕ_0 to ensure the relative degree of 1 from $\phi_0^{(q)}$ to $\mathbf{u}$ (Liu and Tomizuka, 2014). Therefore, the safety constraint in (5c) can be equivalently written as

$$\phi(\mathbf{x}_{k+1}^{wc}) \leq \max\{0, \phi(\mathbf{x}_k) - \gamma dt\}, \tag{6}$$

where $\mathbf{x}_{k+1}^{wc} = \mathbf{x}_k + \mathbf{f}^{wc}(\mathbf{x}_k, \mathbf{u}_k)dt$ and $\mathbf{f}^{wc}(\mathbf{x}_k, \mathbf{u}_k) = \arg\max_{\underline{\mathbf{f}}_B(\mathbf{x}_k,\mathbf{u}_k) \leq \mathbf{f} \leq \bar{\mathbf{f}}_B(\mathbf{x}_k,\mathbf{u}_k)} \phi(\mathbf{x}_k + \mathbf{f}dt)$ is the worst case state and the worst case NN relaxation, respectively. Our goal is to find ϕ such that for all state $\mathbf{x}_k$, there exists a control $\mathbf{u}_k \in \mathcal{U}$ that satisfies (6) (persistent feasibility). For the BPO-relaxed dynamics, we define the whole legal state set without safety constraint as $B \subseteq \mathcal{X}$ and the infeasible state subset of B regarding persistent feasibility as $B^* = \{\mathbf{x}_k \in B \mid \forall \mathbf{u}_k, \phi(\mathbf{x}_{k+1}^{wc}) > \max\{0, \phi(\mathbf{x}_k) - \gamma dt\}\}$. The emptiness of $B^* = \emptyset$ implies persistent feasibility.

Following Wei and Liu (2022), we adopt the implementation (Feldt, 2018) of evolutionary methods (Das et al., 2016; Hansen, 2016) to optimize the parameters in ϕ. Specifically, the evolution algorithm runs for multiple generations. In each iteration, we uniformly sample a dense subset $S \subset B$ and find the minimal infeasible rate $r = |S \cap B^*|/|S|$ based on the sampled parameter candidates from a multivariate Gaussian distribution. The new Gaussian distribution will be updated based on the last candidates with the least infeasible rate r. Besides, through reachability-based methods like interval arithmetic, the Euclidean error of $\mathbf{f}$ can be upper-bounded by $\Delta f = \max_{\mathbf{x},\mathbf{u}} \|\mathbf{f}_B - \mathbf{f}\|$. Therefore, we propose the following Proposition 1 based on Assumption 1, showing that with dense sampling $S \in B$ and the convergence of r to 0, the optimized safety index can induce persistent feasibility even with the worst-case unsafe state update $\mathbf{f}^{wc}(\mathbf{x}, \mathbf{u})$ for any state in B.

Assumption 1 *$\mathbf{f}$ and ϕ are Lipschitz continuous functions over compact set $B \subseteq \mathcal{X}$ with Lipschitz constants k_f and k_ϕ under ℓ_2 norm, respectively.*

Proposition 1 *Suppose 1) we sample a state subset $S \subset B$ such that $\forall \mathbf{x} \in B$, $\min_{\mathbf{x}' \in S} \|\mathbf{x} - \mathbf{x}'\| \leq \delta$, where δ is the sampling density; and 2) $\forall \mathbf{x}' \in S$, there exists a safe control $\mathbf{u}$, s.t. $\phi(\mathbf{x}' + \mathbf{f}^{wc}(\mathbf{x}', \mathbf{u})dt) \leq \max\{-\epsilon, \phi(\mathbf{x}') - \gamma dt - \epsilon\}$, where $\epsilon = k_\phi(2\delta + 2\Delta f dt + k_f \delta dt)$. Then $\forall \mathbf{x} \in B, \exists \mathbf{u}$, s.t.*

$$\phi(\mathbf{x}^{wc}) = \max_{\mathbf{f}(\mathbf{x},\mathbf{u}) \in [\underline{\mathbf{f}}_B(\mathbf{x},\mathbf{u}), \bar{\mathbf{f}}_B(\mathbf{x},\mathbf{u})]} \phi(\mathbf{x} + \mathbf{f}(\mathbf{x}, \mathbf{u})dt) \leq \max\{0, \phi(\mathbf{x}) - \gamma dt\}. \tag{7}$$

Proof Based on 1), $\forall \mathbf{x} \in B$, we can find $\mathbf{x}' \in S$ such that $\|\mathbf{x} - \mathbf{x}'\| \leq \delta$. Based on 2), for this $\mathbf{x}'$, we can find $\mathbf{u}$ such that $\phi(\mathbf{x}' + \mathbf{f}^{wc}(\mathbf{x}', \mathbf{u})dt) \leq \max\{0, \phi(\mathbf{x}') - \gamma dt\} - \epsilon$. Based on Assumption 1, we show below that Eq. (7) holds by using the Lipschitz condition k_f and k_ϕ and triangle inequality:

$$\begin{aligned}
\phi(\mathbf{x}^{wc}) =& \phi(\mathbf{x} + \mathbf{f}^{wc}(\mathbf{x}, \mathbf{u})dt) - \phi(\mathbf{x}' + \mathbf{f}^{wc}(\mathbf{x}', \mathbf{u})dt) + \phi(\mathbf{x}' + \mathbf{f}^{wc}(\mathbf{x}', \mathbf{u})dt) \\
\leq& \underline{k_\phi \|\mathbf{x} - \mathbf{x}' + [\mathbf{f}^{wc}(\mathbf{x}, \mathbf{u}) - \mathbf{f}^{wc}(\mathbf{x}', \mathbf{u})]dt\|} + \max\{0, \phi(\mathbf{x}') - \gamma dt\} - \epsilon \\
\leq& \underline{k_\phi \|\mathbf{x} - \mathbf{x}'\| + k_\phi \|\mathbf{f}^{wc}(\mathbf{x}, \mathbf{u}) - \mathbf{f}^{wc}(\mathbf{x}', \mathbf{u})\|dt} + \max\{0, \phi(\mathbf{x}') - \gamma dt\} - \epsilon \\
\leq& k_\phi \delta + k_\phi \underline{\|\mathbf{f}^{wc}(\mathbf{x}, \mathbf{u}) - \mathbf{f}(\mathbf{x}, \mathbf{u}) - \mathbf{f}^{wc}(\mathbf{x}', \mathbf{u}) + \mathbf{f}(\mathbf{x}', \mathbf{u}) + \mathbf{f}(\mathbf{x}, \mathbf{u}) - \mathbf{f}(\mathbf{x}', \mathbf{u})}\|dt \\
&+ \max\{0, \phi(\mathbf{x}) - \gamma dt\} + \max\{0, \phi(\mathbf{x}') - \gamma dt\} - \max\{0, \phi(\mathbf{x}) - \gamma dt\} - \epsilon \\
\leq& k_\phi \delta + k_\phi (\underline{\|\mathbf{f}^{wc}(\mathbf{x}, \mathbf{u}) - \mathbf{f}(\mathbf{x}, \mathbf{u})\| + \|\mathbf{f}(\mathbf{x}', \mathbf{u}) - \mathbf{f}^{wc}(\mathbf{x}', \mathbf{u})\| + \|\mathbf{f}(\mathbf{x}, \mathbf{u}) - \mathbf{f}(\mathbf{x}', \mathbf{u})\|})dt \\
&+ \max\{0, \phi(\mathbf{x}) - \gamma dt\} + \|\phi(\mathbf{x}') - \phi(\mathbf{x})\| - \epsilon \\
\leq& k_\phi \delta + k_\phi (\Delta f + \Delta f + k_f \delta)dt + \max\{0, \phi(\mathbf{x}) - \gamma dt\} + k_\phi \delta - \epsilon \\
=& \max\{0, \phi(\mathbf{x}) - \gamma dt\},
\end{aligned} \tag{8}$$

which concludes the proof. ■

3.2. Linearization of the Safety Constraint with a Linear Taylor Layer

Although the persistent feasibility is guaranteed by the worst-case safety index ϕ, we still need to address the nonlinearity of the safety constraint (6), which is equivalent to

$$\phi(\mathbf{x}^{wc}_{k+1}) := \max_{\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)\in[\underline{\mathbf{f}}_B(\mathbf{x}_k,\mathbf{u}_k),\bar{\mathbf{f}}_B(\mathbf{x}_k,\mathbf{u}_k)]} \phi(\mathbf{x}_k + \mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)dt) \le \max\{0, \phi(\mathbf{x}_k) - \gamma dt\}. \tag{9}$$

We apply the first-order Taylor expansion with Lagrange Mean Value Theorem for $\phi(\mathbf{x}_{k+1})$ at the point $\mathbf{x}_k$ for $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)dt$ and obtain

$$\phi(\mathbf{x}_{k+1}) = \underbrace{\phi(\mathbf{x}_k) + \nabla^\top_{\mathbf{x}}\phi(\mathbf{x}_k)\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)dt}_{\phi_{\mathbf{f}}(\mathbf{x}_k,\mathbf{u}_k)} + \underbrace{\frac{1}{2}\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)^\top\nabla^2_{\mathbf{x}}\phi(\mathbf{x}')\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)(dt)^2}_{R^{\mathbf{x}_k}(\mathbf{x}')}, \tag{10}$$

which consists of the first-order Taylor polynomial $\phi_{\mathbf{f}}(\mathbf{x}_k,\mathbf{u}_k)$ and the Lagrange remainder term $R^{\mathbf{x}_k}(\mathbf{x}')$ with $\mathbf{x}' \in [\mathbf{x}_k, \mathbf{x}_{k+1}]$. Then we formulate the first-order Taylor approximation $\phi_{\mathbf{f}}(\mathbf{x}_k,\mathbf{u}_k) : \mathbb{R}^{m_x} \times \mathbb{R}^{m_u} \mapsto \mathbb{R}$ as the composite function of the neural network $\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k) : \mathbb{R}^{m_x} \times \mathbb{R}^{m_u} \mapsto \mathbb{R}^{m_x}$ and an additional linear mapping $\phi^{\mathbf{x}_k}$ (called the linear Taylor layer) with weight $\nabla^\top_{\mathbf{x}}\phi(\mathbf{x}_k)dt \in \mathbb{R}^{1\times m_x}$ and bias $\phi(\mathbf{x}_k) \in \mathbb{R}$, *i.e.* $\phi_{\mathbf{f}} = \phi^{\mathbf{x}_k} \circ \mathbf{f}$, $\phi^{\mathbf{x}_k}(\mathbf{f}) = \nabla^\top_{\mathbf{x}}\phi(\mathbf{x}_k)dt\mathbf{f} + \phi(\mathbf{x}_k)$.

Similar to the computation of $\underline{\mathbf{f}}_B(\mathbf{x},\mathbf{u}) \le \mathbf{f}(\mathbf{x},\mathbf{u}) \le \bar{\mathbf{f}}_B(\mathbf{x},\mathbf{u})$ in Section 2.2, with BPO for each ReLU activation at each layer, the first-order Taylor approximation $\phi_{\mathbf{f}} = \phi^{\mathbf{x}_k} \circ \mathbf{f}_n \circ \mathbf{f}_{n-1} \circ \cdots \circ \mathbf{f}_1$ can be relaxed to be $\underline{\phi}_{\mathbf{f}_B}(\mathbf{x},\mathbf{u}) \le \phi_{\mathbf{f}}(\mathbf{x},\mathbf{u}) \le \bar{\phi}_{\mathbf{f}_B}(\mathbf{x},\mathbf{u})$ given $\mathbf{x},\mathbf{u}$. For the Lagrange remainder term $R^{\mathbf{x}_k}(\mathbf{x}')$ with $\mathbf{x}' \in [\mathbf{x}_k, \mathbf{x}_{k+1}]$, we show that it can be bounded by $\frac{1}{2}M_f^2 M_\phi (dt)^2$ through Proposition 2, while $R^{\mathbf{x}_k}(\mathbf{x}')$ is usually neglected in the previous work (Wei and Liu, 2022).

Proposition 2 *If the ℓ_2 operator norm of the Hessian matrix $\nabla^2_{\mathbf{x}}\phi(\mathbf{x})$ is bounded by M_ϕ for any $\mathbf{x} \in [\mathbf{x}_k, \mathbf{x}_{k+1}]$ and the Euclidean norm of $\mathbf{f}(\mathbf{x},\mathbf{u})$ is bounded by M_f, it holds that*

$$|R^{\mathbf{x}_k}(\mathbf{x})| \le \frac{1}{2}M_f^2 M_\phi (dt)^2, \quad \forall \mathbf{x} \in [\mathbf{x}_k, \mathbf{x}_{k+1}] \tag{11}$$

Proof For $\mathbf{x} \in [\mathbf{x}_k, \mathbf{x}_{k+1}]$, we have $\|\nabla^2_{\mathbf{x}}\phi(\mathbf{x})\|_{op} \le M_\phi$, $\|\mathbf{f}(\mathbf{x},\mathbf{u})\|_2 \le M_f$, where *op* indicates the operator norm. Therefore, we show Eq. (11) by using the operator norm and Cauchy–Schwarz inequality as below:

$$\begin{aligned}
|R^{\mathbf{x}_k}(\mathbf{x})| =& \frac{(dt)^2}{2}\|\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)^\top\nabla^2_{\mathbf{x}}\phi(\mathbf{x}')\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)\|_2 \le \frac{(dt)^2}{2}\|\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)^\top\|_2\|\nabla^2_{\mathbf{x}}\phi(\mathbf{x}')\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)\|_{op} \\
\le& \frac{(dt)^2}{2}\|\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)^\top\|_2 \max_{\|\mathbf{f}\|_2=1} \|\mathbf{f}^\top\nabla^2_{\mathbf{x}}\phi(\mathbf{x}')\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)\|_2 \\
\le& \frac{(dt)^2}{2}\|\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)^\top\|_2 \max_{\|\mathbf{f}\|_2=1} \|\mathbf{f}\|_2\|\nabla^2_{\mathbf{x}}\phi(\mathbf{x}')\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)\|_2 \\
\le& \frac{(dt)^2}{2}\|\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)^\top\|_2\|\nabla^2_{\mathbf{x}}\phi(\mathbf{x}')\|_{op}\|\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)\|_2 \le \frac{1}{2}M_f^2 M_\phi (dt)^2.
\end{aligned}$$

which concludes the proof. ■

Therefore, the safety constraint in (9) can be rewritten as the linear inequality $\bar{\phi}_{\mathbf{f}_B}(\mathbf{x}_k, \mathbf{u}_k) + \frac{1}{2}M_f^2 M_\phi (dt)^2 \leq \max\{0, \phi(\mathbf{x}_k) - \gamma dt\}$, where $\bar{\phi}_{\mathbf{f}_B}(\mathbf{x}_k, \mathbf{u}_k)$ is the upper bound of $\phi_{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k)$ with BPO relaxation and $\frac{1}{2}M_f^2 M_\phi (dt)^2$ can be approximated as an optimizable parameter in Sec. 3.1.

3.3. Safe Control with BPO-Relaxed NNDM

Based on the worst case safety index (that ensures persistent feasibility) and the linearization of the worst-case safety constraint, we finally transform the original constrained optimization (5) into the following form:

$$\min_{\mathbf{u}_k, \mathbf{x}_{k+1}, \{\mathbf{z}_i\}_{i=0}^n} \quad \|\mathbf{x}_{k+1} - \mathbf{x}_{k+1}^r\|_p \tag{12a}$$

$$\textit{s.t. } \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{z}_n dt, \ \mathbf{z}_0 = [\mathbf{x}_k^\top, \mathbf{u}_k^\top]^\top, \ \mathbf{u}_k \in \mathcal{U}, \tag{12b}$$

$$z_{ij} \geq \hat{z}_{ij}, \ z_{ij} \geq 0, \ z_{ij} \leq g_K(\hat{z}_{ij}) \text{ in (4)}, \tag{12c}$$

$$\hat{z}_{ij} = \mathbf{w}_{ij}\mathbf{z}_{i-1} + b_{ij}, \ \forall i \in \{1, \ldots, n\}, \ \forall j \in \{1, \ldots, k_i\}, \tag{12d}$$

$$\bar{\phi}_{\mathbf{f}_B}(\mathbf{x}_k, \mathbf{u}_k) \leq \max\{-\zeta, \phi(\mathbf{x}_k) - \gamma dt - \zeta\}, \tag{12e}$$

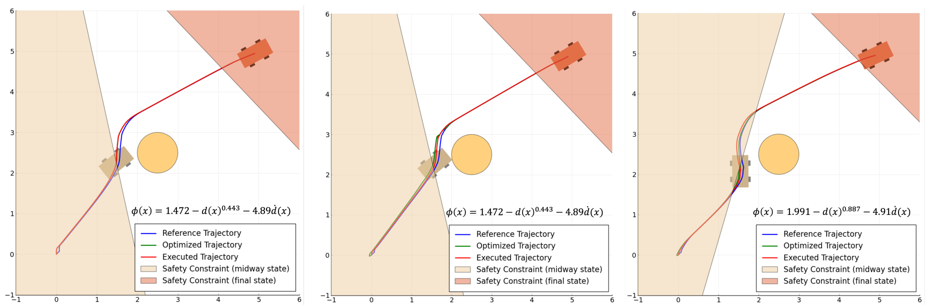
where $\zeta = \frac{1}{2}M_f^2 M_\phi (dt)^2$ is from Proposition 2. In this paper, we consider $K = 1, 2$ in Eq. 4 as the BPO degree to illustrate the proposed design. When $K = 1$, the optimization problem (12) is either Linear Programming ($p = 1$) or Quadratic Programming ($p = 2$); When $K = 2$, it is either Quadratically Constrained Linear Programming ($p = 1$) or Quadratically Constrained Quadratic Programming ($p = 2$). Besides, we approximate the upper bound of $\phi_{\mathbf{f}_B}(\mathbf{x}_k, \mathbf{u}_k)$ in (12e) by sampling and explore the use of the existing solvers, including CPLEX, Gurobi, and Ipopt, to solve the obtained optimization problems.

4. Experiment

In the experiment, we aim to answer the following questions: how scalable is the proposed BOND compared to the MIP-based baseline (Wei and Liu, 2022) considering different sizes of models and tasks? How is the performance influenced by different optimization solvers and the tightness of BPO relaxation? We answer the first question in Section 4.2 through the comparison of different dynamic models for collision avoidance and safe following of the unicycle, followed by the validation of the effectiveness of the worst-case safety constraint. Section 4.3 shows the influence of several key factors for the second question.

4.1. Experimental Setup

Environment and dynamics. To be consistent with Wei and Liu (2022), the experiment is based on the neural network dynamic models for a second-order unicycle in a 2D setting. The 4D states $\mathcal{X} \subset \mathbb{R}^4$ are the 2D positions, velocity and heading angle, and the 2D control inputs $\mathcal{U} \subset \mathbb{R}^2$ are the acceleration and angular velocity. The current states and control inputs are also the inputs of neural networks, and the outputs of the neural networks are the 2D velocity, acceleration, and angular velocity as the derivatives of the states. The states and inputs are bounded as $B \subset \mathcal{X}$: $[-10, 10] \times [-10, 10] \times [-2, 2] \times [-\pi, \pi]$ and $\mathcal{U}$: $[-4, 4] \times [-\pi, \pi]$. Collision avoidance and safe following are used for evaluation with different safety constraints, where the unicycle is supposed to be at least 0.5 away from the obstacle for collision avoidance and be within 1 and 2 away from the



(a) (b) (c)

Figure 2: Collision avoidance with (a) MIP-based NNDM with plain safety constraint; (b) BOND with plain safety constraint; (c) BOND with worst-case safety constraint. The safety constraint is visualized as the half-space of state space based on (12e). $\phi(x)$ is the corresponding safety index and $d(x)$ is the distance between the unicycle and obstacle.

Collision Avoidance		FC2-100		FC3-100		FC4-100	
		Error	Time (s)	Error	Time (s)	Error	Time (s)
Linear Objective	MIND-SIS	$\mathbf{2.31e^{-9}}$	0.0582	$\mathbf{1.56e^{-9}}$	0.255	$\mathbf{1.42e^{-9}}$	151
	Ours (BPO-1)	0.106	**0.0031**	0.0943	**0.0080**	0.0911	**0.0279**
	Ours (BPO-2)	0.0668	0.0296	0.0298	0.716	0.0757	2.76
Quadratic Objective	MIND-SIS	$\mathbf{1.78e^{-9}}$	0.0576	$\mathbf{1.31e^{-9}}$	0.338	$\mathbf{1.29e^{-9}}$	226
	Ours (BPO-1)	0.0435	**0.0085**	0.0338	**0.0222**	0.0401	**0.211**
	Ours (BPO-2)	0.0263	0.0191	0.0191	0.577	0.0273	2.05

Table 1: Comparison of the baseline and ours under different model complexity and optimization objective norms ($p = 1, 2$) for collision avoidance. Notations: **best** and second best results.

moving target for safe following. The neural networks have fully-connected layers with the ReLU activation, with different depths ($d = 2, 3, 4$) and widths ($w = 50, 100, 200$) and are denoted as FCd-w, *e.g.* FC3-100 means a model of 3-layer with 100 neurons per layer. To verify our small models (< 1000 neurons), MIP works the best according to α, β-CROWN (Wang et al., 2021).

Optimization and evaluation metrics. To solve the real-time optimization in (12), the reference states are generated through one-tenth interpolation between the current state and the goal as real-time planning. We solve the optimization using CPLEX, Gurobi and Ipopt with linear or quadratic objectives ($p = 1, 2$, respectively) of the tracking error term in (5a) for both the baseline and our BPO relaxation of the degree of 1 and 2 ($K = 1, 2$). The pre-activation bounds are computed using ConvDual (Wong and Kolter, 2018) and interval arithmetic (IA), where the former is much tighter (Liu et al., 2021; Gowal et al., 2018). The default setting is with ConvDual pre-activation bounds under CPLEX solver, for both baseline and our 1-degree BPO, while our 2-degree BPO is with Ipopt solver as default due to nonconvex quadratic constraints. More results regarding these factors can be found in Section 4.3. The evaluation metrics are *prediction time per step* and *prediction error per step*, where the latter is between the optimized and the executed states with corresponding norms in the optimization objective. The mean of each metric is calculated for 10 trajectories with random initial states under each setting, where the step number per trajectory is around 100. The code is available at `https://github.com/intelligent-control-lab/BOND`.

4.2. Performance Comparison with Baseline

Significance of the worst-case safety constraint for BPO-relaxed NNDM. As shown in Fig. 2(a), the MIP-based baseline MIND-SIS (Wei and Liu, 2022) works well under the plain safety

Safe Following		FC2-100		FC3-100		FC4-100	
		Error	Time (s)	Error	Time (s)	Error	Time (s)
Linear Objective	MIND-SIS	$\mathbf{2.86e^{-9}}$	0.0387	$\mathbf{1.55e^{-9}}$	0.232	$\mathbf{9.34e^{-10}}$	40.4
	Ours (BPO-1)	0.116	**0.0034**	0.0886	**0.0072**	0.112	**0.444**
	Ours (BPO-2)	0.0504	0.0360	0.0518	0.579	0.0794	2.40
Quadratic Objective	MIND-SIS	$\mathbf{1.55e^{-9}}$	0.0517	$\mathbf{1.12e^{-9}}$	0.267	$\mathbf{1.08e^{-9}}$	52.8
	Ours (BPO-1)	0.0243	**0.0071**	0.0417	**0.0223**	0.0498	**0.455**
	Ours (BPO-2)	0.0241	0.0283	0.0249	0.503	0.0477	2.08

Table 2: Comparison of the baseline and ours under different model complexity and optimization objective norms ($p = 1, 2$) for safe following. Notations: **best** and second best results.

Different solvers for two tasks		MIND-SIS		Ours (BPO-1)		Ours (BPO-2)	
		Error	Time (s)	Error	Time (s)	Error	Time (s)
Collision Avoidance	CPLEX	$\mathbf{1.31e^{-9}}$	**0.338**	0.0338	**0.0222**	—	—
	Gurobi	$2.02e^{-9}$	0.516	**0.0327**	0.0572	0.0396	529
	Ipopt	—	—	0.0332	0.338	**0.0191**	**0.577**
Safe Following	CPLEX	$\mathbf{1.12e^{-9}}$	**0.267**	**0.0417**	**0.0223**	—	—
	Gurobi	$1.25e^{-9}$	0.349	0.0418	0.0475	0.0495	333
	Ipopt	—	—	0.0439	0.317	**0.0249**	**0.503**

Table 3: Comparison of performance with different solvers using FC3-100 and quadratic objective for both baseline and ours. The best results among different solvers are in **bold** and "—" indicates infeasibility.

constraint of $\mathcal{X}_S$ in Wei and Liu (2022), while the plain safety constraint results in prediction error between the optimized and executed states under BOND, causing collision as (b) shows. However, with the proposed worst-case safety constraint (5c) of $\mathcal{X}_{BPO}$, collision avoidance under BOND is guaranteed in (c) even if the prediction error still exists between the optimized and executed states. This validates the significance of the more conservative worst-case safety constraint (5c) for BPO-relaxed NNDM. The offline time for synthesizing the plain safety index is 4.57h for (a) and (b), while the time for our worst-case one is 19.4h due to higher computation complexity.

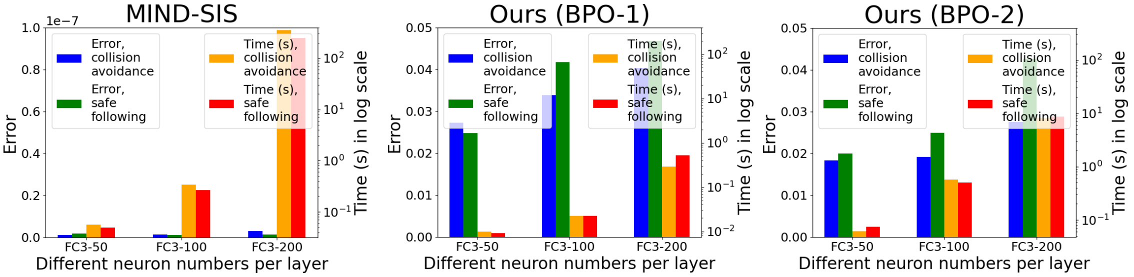
Performance comparison of prediction error and computation time. Table 1 and Table 2 present the results of baseline MIND-SIS (Wei and Liu, 2022) and ours with BPO degrees of 1 and 2 for collision avoidance and safe following. It can be seen that under all depths of models, our BPO relaxation results in 10-100 times less computation time per step compared to the baseline, although their prediction errors are larger than those of the baseline as ground truth. Across all the models, BPO-2 has smaller prediction errors but slower computation than BPO-1 due to tighter but non-convex quadratic relaxation in (4) when $K = 2$. As the models go deeper, we can see our prediction time does not drastically increase as MIND-SIS does, showing our method scales better.

4.3. Ablation Study

Influence of optimization solvers. Table 3 shows how commonly used solvers affect the tracking performance. We can find that CPLEX is usually the fastest for MIND-SIS and our BPO-1, but it cannot solve BPO-2 with non-convex quadratic constraints. Gurobi generally applies to solving different problems but suffers from longer computation time, especially for BPO-2. As a nonlinear optimizer, Ipopt performs satisfactory results for BPO-2 with local convergence at risk of unsoundness, and it is much slower for BPO-1 and cannot be used for the MIP-based baseline.

Different pre-activation bounds		MIND-SIS		Ours (BPO-1)		Ours (BPO-2)	
		Error	Time (s)	Error	Time (s)	Error	Time (s)
Collision Avoidance	IA	$1.34e^{-9}$	0.511	0.0385	0.0260	0.0254	0.812
	ConvDual	$\mathbf{1.31e^{-9}}$	**0.338**	**0.0338**	**0.0222**	**0.0191**	**0.577**
Safe Following	IA	$1.17e^{-9}$	0.365	0.0554	0.0265	0.0510	1.12
	ConvDual	$\mathbf{1.12e^{-9}}$	**0.267**	**0.0417**	**0.0223**	**0.0249**	**0.503**

Table 4: Comparison of performance with different pre-activation bounds for both baseline and ours using FC3-100 and quadratic objective. The better results between IA and ConvDual are in **bold**.



(a) *(b)* *(c)*

Figure 3: Comparison of performance with different neuron numbers per layer for both baseline and ours using quadratic objective. (a) MIND-SIS. (b) Ours (BPO-1). (c) Ours (BPO-2).

Influence of pre-activation bounds. Since the MIP-based baseline and ours both greatly rely on the pre-activation bounds of ReLU activation layers, we compare the results of different pre-activation bounds, interval arithmetic (IA) (Liu et al., 2021; Gowal et al., 2018) and ConvDual (Wong and Kolter, 2018), as shown in Table 4. It can be seen that IA causes longer prediction time and larger errors per step because of its poor tightness and large search space.

Influence of neuron numbers per layer. From Fig. 3, it can be seen that the time consumption of MIND-SIS exponentially explodes when layer width increases, while our BPO-based ones maintain a relatively linearly increased computation time, validating the remarkable scalability of our methods. Different from Table 1 and Table 2, the errors increase as the neuron number per layer goes up under BPO relaxation, implying that the relaxation becomes looser with more neurons per layer.

5. Conclusion

In this work, we introduce Bernstein over-approximated neural dynamics (BOND) with Bernstein polynomial over-approximation (BPO) of ReLU activation layers in NNDMs to speed up the optimization of safe tracking. To ensure the persistent feasibility of safety set under the approximation error of BOND, the worst-case safety index is synthesized offline to satisfy the safety constraint for the most unsafe potential predicted states of BOND. Comprehensive experiments validate the time efficiency and scalability of BOND. Our main limitation lies in the trade-off between optimality and conservativeness due to the worst-case safety constraint. Besides, the model mismatch has not been considered in our setting. Future directions can be exploring the robustness of BOND in more real-world robot settings to ensure safety under out-of-distribution model mismatch.

Acknowledgments

This work is in part supported by the National Science Foundation under Grant No. 2144489. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. Jianglin Lan is supported by a Leverhulme Trust Early Career Fellowship under Award ECF-2021-517. We also would like to thank Tianhao Wei from CMU for the support in the experiments.

References

Devansh R Agrawal and Dimitra Panagou. Safe control synthesis via input constrained control barrier functions. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 6113–6118. IEEE, 2021.

Franco Blanchini. Set invariance in control. *Automatica*, 35(11):1747–1767, 1999.

Swagatam Das, Sankha Subhra Mullick, and Ponnuthurai N Suganthan. Recent advances in differential evolution–an updated survey. *Swarm and evolutionary computation*, 27:1–30, 2016.

Ruediger Ehlers. Formal verification of piece-wise linear feed-forward neural networks. In *International Symposium on Automated Technology for Verification and Analysis*, pages 269–286. Springer, 2017.

Wael Fatnassi, Haitham Khedr, Valen Yamamoto, and Yasser Shoukry. Bern-nn: Tight bound propagation for neural networks using bernstein polynomial interval arithmetic. In *Proceedings of the 26th ACM International Conference on Hybrid Systems: Computation and Control*, pages 1–11, 2023.

Robert Feldt. Blackboxoptim.jl. https://github.com/robertfeldt/BlackBoxOptim.jl, 2018.

Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. On the effectiveness of interval bound propagation for training verifiably robust models. *arXiv preprint arXiv:1810.12715*, 2018.

Nikolaus Hansen. The cma evolution strategy: A tutorial. *arXiv preprint arXiv:1604.00772*, 2016.

Chao Huang, Jiameng Fan, Xin Chen, Wenchao Li, and Qi Zhu. Polar: A polynomial arithmetic framework for verifying neural-network controlled systems. In *International Symposium on Automated Technology for Verification and Analysis*, pages 414–430. Springer, 2022.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *arXiv preprint arXiv:1906.08253*, 2019.

Haitham Khedr and Yasser Shoukry. Deepbern-nets: Taming the complexity of certifying neural networks using bernstein polynomial activations and precise bound propagation. *arXiv preprint arXiv:2305.13508*, 2023.

Linyi Li, Tao Xie, and Bo Li. Sok: Certified robustness for deep neural networks. In *2023 IEEE symposium on security and privacy (SP)*, pages 1289–1310. IEEE, 2023a.

Xiao Li, Yutong Li, Anouck Girard, and Ilya Kolmanovsky. System-level safety guard: Safe tracking control through uncertain neural network dynamics models. *arXiv preprint arXiv:2312.06810*, 2023b.

Changliu Liu and Masayoshi Tomizuka. Control in a safe set: Addressing safety in human-robot interactions. In *ASME 2014 Dynamic Systems and Control Conference*. American Society of Mechanical Engineers Digital Collection, 2014.

Changliu Liu, Tomer Arnon, Christopher Lazarus, Christopher Strong, Clark Barrett, Mykel J Kochenderfer, et al. Algorithms for verifying deep neural networks. *Foundations and Trends® in Optimization*, 4(3-4):244–404, 2021.

Ziang Liu, Genggeng Zhou, Jeff He, Tobia Marcucci, Li Fei-Fei, Jiajun Wu, and Yunzhu Li. Model-based control with sparse neural dynamics. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Anusha Nagabandi, Gregory Kahn, Ronald S Fearing, and Sergey Levine. Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7559–7566. IEEE, 2018.

Mitio Nagumo. Über die lage der integralkurven gewöhnlicher differentialgleichungen. *Proceedings of the Physico-Mathematical Society of Japan. 3rd Series*, 24:551–559, 1942.

Vincent Tjeng, Kai Y Xiao, and Russ Tedrake. Evaluating robustness of neural networks with mixed integer programming. In *International Conference on Learning Representations*, 2018.

Deepak Tolani, Ambarish Goswami, and Norman I Badler. Real-time inverse kinematics techniques for anthropomorphic limbs. *Graphical models*, 62(5):353–388, 2000.

Shiqi Wang, Huan Zhang, Kaidi Xu, Xue Lin, Suman Jana, Cho-Jui Hsieh, and J Zico Kolter. Beta-crown: Efficient bound propagation with per-neuron split constraints for neural network robustness verification. *Advances in Neural Information Processing Systems*, 34:29909–29921, 2021.

Tianhao Wei and Changliu Liu. Safe control algorithms using energy functions: A unified framework, benchmark, and new directions. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 238–243. IEEE, 2019.

Tianhao Wei and Changliu Liu. Safe control with neural network dynamic models. In *Learning for Dynamics and Control Conference*, pages 739–750. PMLR, 2022.

Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning*, pages 5286–5295. PMLR, 2018.

Eric Wong, Frank Schmidt, Jan Hendrik Metzen, and J Zico Kolter. Scaling provable adversarial defenses. *Advances in Neural Information Processing Systems*, 31, 2018.