Supplementary Material

Anonymous submission

Proofs

Theorem 1 ((Prajna, Jadbabaie, and Pappas 2007)) *Given a CCDS $\mathcal{C} = (\mathbf{f}, \Theta, \Psi)$ and an unsafe region X_u, if there exists a real-valued differentiable function $B : \Psi \to \mathbb{R}$ satisfying the following conditions: i) $\forall \mathbf{x} \in \Theta. B(\mathbf{x}) \geq 0$; ii) $\forall \mathbf{x} \in X_u. B(\mathbf{x}) < 0$; iii) $\forall \mathbf{x} \in \Psi. (B(\mathbf{x}) = 0 \Rightarrow \mathcal{L}_{\mathbf{f}} B(\mathbf{x}) > 0)$, where $\mathcal{L}_{\mathbf{f}} B(\mathbf{x})$ denotes the Lie-derivative of $B(\mathbf{x})$ along the vector field $\mathbf{f}(\mathbf{x})$, i.e., $\mathcal{L}_{\mathbf{f}} B(\mathbf{x}) = \sum_{i=1}^{n} \frac{\partial B}{\partial x_i} \cdot f_i(\mathbf{x})$, then $B(\mathbf{x})$ is called a barrier certificate of $\mathcal{C}$, and the safety of $\mathcal{C}$ is guaranteed.*

Proof. The theorem was first provided and proved in (Prajna, Jadbabaie, and Pappas 2007). □

Proposition 1 *For a controlled CCDS $\mathcal{C} = (\mathbf{f}, \Theta, \Psi)$ with $\mathbf{f}$ defined by (1) if there exists a state-wise admissible control set $U_{\mathbf{x}}$ such that there exists a uniform barrier certificate for the closed-loop system with any controller $\mathbf{k}$ satisfying $\forall \mathbf{x} \in \Psi. \mathbf{k}(\mathbf{x}) \in U_{\mathbf{x}}$, then $U_{\mathbf{x}}$ is a safe control set.*

Proof. For any continuous controller $\mathbf{k}$ satisfying $\forall \mathbf{x} \in \Psi. \mathbf{k}(\mathbf{x}) \in U_{\mathbf{x}}$, if there exists one uniform barrier certificate for the closed-loop system $\mathcal{C}$ with $\mathbf{k}(\mathbf{x})$ as the controller, then the safety of the control $\mathcal{C}$ is guaranteed for the existence of the barrier certificate by the Theorem 1. Since any controller $\mathbf{k}$ satisfying $\mathbf{k}(\mathbf{x}) \in U_{\mathbf{x}}, \forall \mathbf{x} \in \Psi$ makes the controlled system $\mathcal{C}$ safe, $U_{\mathbf{x}}$ is a safe control set as per Definition 4. □

Proposition 2 *Given a controlled CCDS $\mathcal{C} = (\mathbf{f}, \Theta, \Psi)$, a BNN controller $\pi^{\mathbf{w}}$ with $\mathbf{w} \sim q(\mathbf{w})$, and a safe control set $U_{\mathbf{x}}$, a set $W \subset supp(q(\mathbf{w}))$ is a safe weight set if $\pi^W(\mathbf{x}) \subset U_{\mathbf{x}}$ for all $\mathbf{x} \in \Psi$, where $\pi^W(\mathbf{x}) = \{\pi^{\mathbf{w}}(\mathbf{x}) \mid \mathbf{w} \in W\}$.*

Proof. We prove this proposition by contradiction. Suppose that W is not a safe weight set, then there exists a continuous instantiation $\mathbf{w}(t)$ from W such that the continuous dynamical system $\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = \mathbf{f}(\mathbf{x}, \pi^{\mathbf{w}(t)}(\mathbf{x}(t)))$ has an unsafe trajectory $\mathbf{x}(t)$ starting from $\mathbf{x}_0 \in \Theta$ stays in Ψ but intersects with X_u. Furthermore, suppose that the safe control set $U_{\mathbf{x}}$ is certified by a uniform barrier function $B(\mathbf{x})$. Then we assert that there exists $T > 0$ s.t. the above unsafe trajectory $\mathbf{x}(t)$ intersects with $B(\mathbf{x}) = 0$ at T and $B(\mathbf{x}(t))$ has a non-positive derivative at T, i.e. $B(\mathbf{x}(T)) = 0 \wedge \frac{\mathrm{d}B(\mathbf{x}(t))}{\mathrm{d}t}|_{t=T} \leq 0$, which means $\frac{\partial B(\mathbf{x})}{\partial \mathbf{x}}|_{\mathbf{x}=\mathbf{x}(T)} \cdot \mathbf{f}(\mathbf{x}(T), \pi^{\mathbf{w}(T)}(\mathbf{x}(T))) \leq 0$. Then we construct a controller $\mathbf{k}(\mathbf{x})$ such that $\forall \mathbf{x} \in \Psi, \mathbf{k}(\mathbf{x}) = \pi^{\mathbf{w}(T)}(\mathbf{x})$. Since $\mathbf{w}(T) \in W$ and $\pi^W(\mathbf{x}) \subset U_{\mathbf{x}}$ for all $\mathbf{x} \in \Psi$, we have that $\mathbf{k}(\mathbf{x}) \in U_{\mathbf{x}}$ for all $\mathbf{x} \in \Psi$. Again by the assumption that $U_{\mathbf{x}}$ is certified by a uniform barrier function $B(\mathbf{x})$, we have that for all $\mathbf{x} \in \Psi$ if $B(\mathbf{x}) = 0$ then $\mathcal{L}_{\mathbf{f}} B(\mathbf{x}) > 0$. Let $\mathbf{x}$ be $\mathbf{x}(T)$, and we get $\mathcal{L}_{\mathbf{f}} B(\mathbf{x}(T)) = \frac{\partial B(\mathbf{x})}{\partial \mathbf{x}}|_{\mathbf{x}=\mathbf{x}(T)} \cdot \mathbf{f}(\mathbf{x}(T), \pi^{\mathbf{w}(T)}(\mathbf{x}(T))) > 0$, which is a contradiction. Thus the proposition holds. □

Proposition 3 *$\mathcal{I}(\mathbf{x}) = p(\mathbf{x}) + [-\gamma^*, \gamma^*]$ with γ^* produced by (3) is a safe control set.*

Proof. We prove this proposition by contradiction. Suppose that $\mathcal{I}(\mathbf{x}) = p(\mathbf{x}) + [-\gamma^*, \gamma^*]$ with γ^* produced by (3) but $\mathcal{I}(\mathbf{x})$ is not a safe control set. Then by Definition 4, there exists $\mathbf{k}(\mathbf{x})$ satisfying $\forall \mathbf{x} \in \Psi. \mathbf{k}(\mathbf{x}) \in [p(\mathbf{x}) - \gamma^*, p(\mathbf{x}) + \gamma^*]$ which makes the systems (1) unsafe. Thus by the definition of safety, i.e., Definition 1, there exists a trajectory $\mathbf{x}(t)$ of system (1) and a constant $T > 0$ s.t. $\mathbf{x}(0) \in \Theta, \mathbf{x}(T) \in X_u$ and $\forall t \in [0, T]. \mathbf{x}(t) \in \Psi$. In addition, suppose that (3) also gives a barrier certificate $B(\mathbf{x}|\mathbf{b}^*)$ denoted by $B(\mathbf{x})$ for short. Then by the continuity of $B(\mathbf{x})$ and $\mathbf{x}(t)$, it is not difficult to show that there exists a constant $\tilde{T} \in [0, T]$ satisfying $B(\mathbf{x}(\tilde{T})) = 0$ and $\mathcal{L}_{\mathbf{f}(\mathbf{x}(\tilde{T}), \mathbf{k}(\mathbf{x}(\tilde{T})))} B(\mathbf{x}(\tilde{T})) \leq 0$, and the latter formula means that there exists $\tilde{\epsilon} \in [-\gamma^*, \gamma^*]$ s.t. $\mathcal{L}_{\mathbf{f}(\mathbf{x}(\tilde{T}), p(\mathbf{x}(\tilde{T})) + \tilde{\epsilon})} B(\mathbf{x}(\tilde{T})) \leq 0$. However, by our assumption that γ^* and $B(\mathbf{x})$ are provided by (3) and thus satisfy (2), we can obtain that $\mathcal{L}_{\mathbf{f}(\mathbf{x}(\tilde{T}), p(\mathbf{x}(\tilde{T})) + \tilde{\epsilon})} B(\mathbf{x}(\tilde{T})) > 0$, which is a contradiction. □

Proposition 4 *Let $\phi^{\mathbf{w}}(\mathbf{x}) = \pi^{\mathbf{w}}(\mathbf{x}) - p(\mathbf{x})$. For a nominal input $\mathbf{x}_0 \in \chi$ we denote the local Lipschitz constants for $\pi^{\mathbf{w}}(\mathbf{x})$ and $p(\mathbf{x})$ as $L_\pi^{\mathbf{w}}(\mathbf{x}_0, \chi)$ and $L_p(\mathbf{x}_0, \chi)$ respectively. Further, let $L = \max_{\mathbf{w} \in \bar{\mathbf{w}}} L_\pi^{\mathbf{w}}(\mathbf{x}_0, \chi) + L_p(\mathbf{x}_0, \chi)$. Then if $\max_{\mathbf{w} \in \bar{\mathbf{w}}} \|\phi^{\mathbf{w}}(\mathbf{x}_0)\| + L\|\mathbf{x} - \mathbf{x}_0\| \leq \gamma^*$ for all $\mathbf{x} \in \chi$, we have $\forall \mathbf{x} \in \chi. \pi^{\bar{\mathbf{w}}}(\mathbf{x}) \subset \mathcal{I}(\mathbf{x})$.*

Proof. For all $\mathbf{x} \in \chi$,

$$
\begin{aligned}
& \|\phi^{\mathbf{w}}(\mathbf{x}) - \phi^{\mathbf{w}}(\mathbf{x}_0)\| \\
= \; & \|\pi^{\mathbf{w}}(\mathbf{x}) - \pi^{\mathbf{w}}(\mathbf{x}_0) - (p(\mathbf{x}) - p(\mathbf{x}_0))\| \\
\leq \; & \|\pi^{\mathbf{w}}(\mathbf{x}) - \pi^{\mathbf{w}}(\mathbf{x}_0)\| + \|(p(\mathbf{x}) - p(\mathbf{x}_0))\| \\
\leq \; & L_\pi^{\mathbf{w}}(\mathbf{x}_0, \chi)\|\mathbf{x} - \mathbf{x}_0\| + L_p(\mathbf{x}_0, \chi)\|\mathbf{x} - \mathbf{x}_0\| \\
= \; & (L_\pi^{\mathbf{w}}(\mathbf{x}_0, \chi) + L_p(\mathbf{x}_0, \chi))\|\mathbf{x} - \mathbf{x}_0\|
\end{aligned}
$$

Thus, for $\mathbf{w} \in \bar{\mathbf{w}}$,

$$\begin{aligned} & \|\pi^{\mathbf{w}}(\mathbf{x}) - p(\mathbf{x})\| \\ = \; & \|\phi^{\mathbf{w}}(\mathbf{x})\| \\ \leq \; & \|\phi^{\mathbf{w}}(\mathbf{x}_0)\| + (L_\pi^{\mathbf{w}}(\mathbf{x}_0, \chi) + L_p(\mathbf{x}_0, \chi))\|\mathbf{x} - \mathbf{x}_0\| \\ \leq \; & \max_{\mathbf{w} \in \bar{\mathbf{w}}} \|\phi^{\mathbf{w}}(\mathbf{x}_0)\| + (\max_{\mathbf{w} \in \bar{\mathbf{w}}} L_\pi^{\mathbf{w}}(\mathbf{x}_0, \chi) \\ & + L_p(\mathbf{x}_0, \chi))\|\mathbf{x} - \mathbf{x}_0\| \\ = \; & \max_{\mathbf{w} \in \bar{\mathbf{w}}} \|\phi^{\mathbf{w}}(\mathbf{x}_0)\| + L\|\mathbf{x} - \mathbf{x}_0\| \\ \leq \; & \gamma^*, \end{aligned}$$

which means $\pi^{\mathbf{w}}(\mathbf{x}) \in \mathcal{I}(\mathbf{x})$ for any $\mathbf{w} \in \bar{\mathbf{w}}$ and for any $\mathbf{x} \in \chi$. Thus the conclusion follows immediately. □

Theorem 2 *The $\tilde{W}$ computed by Algorithm 1 is a safe weight set as per Problem 1.*

Proof. From Line 4 and Line 16 of Algorithm 1, $\bar{\mathbf{w}} \subset supp(q(\mathbf{w}))$ and then $\tilde{W} \subset supp(q(\mathbf{w}))$. For each $\bar{\mathbf{w}}_k$ and $\bar{\mathbf{x}}_i$, suppose that the subroutine in Line 9 computes the local Lipschitz constants L specified by Proposition 4 correctly. Then from Lines 7-10 we can assert that $\pi^{\bar{\mathbf{w}}_k}(\mathbf{x}) \subset \mathcal{I}(\mathbf{x})$ for any $\mathbf{x} \in \bar{\mathbf{x}}_i$ by Proposition 4. (Note that the $\sqrt{n_{\mathbf{x}}}$ in Line 10 is to transform ∞-norm to 2-norm.) Then by the iteration Line 6 it is obvious that $\pi^{\bar{\mathbf{w}}_k}(\mathbf{x}) \subset \mathcal{I}(\mathbf{x})$ for any $\mathbf{x} \in \Psi$. Since $\mathcal{I}(\mathbf{x})$ is a safe control set by Proposition 3, we have that $\bar{\mathbf{w}}_k$ is a safe weight set by Proposition 2. Furthermore, it is not difficult to check that for the same safe control set, the union of two safe weight sets is also a safe weight set. Therefore we conclude from Line 16 that $\tilde{W}$ is a safe weight set. □

Lipschitz constant computation for BNN

In this part, we will introduce how to estimate the upper bound of the local Lipschitz constant for BNN with ReLU activation function. The materials presented in this section can be seen as an extension to the approach proposed in (Avant and Morgansen 2021) to Lipschitz constants estimation for DNNs.

Affine-ReLU function

An affine function can be written as $\mathbf{y} = A\mathbf{x} + \mathbf{b}$, where $A \in \mathbb{R}^{m \times n}$, $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^m$. We describe neural networks with a single output, and the case with multiple outputs can be dealt with similarly. We define an affine-ReLU function as a ReLU composed with an affine function, which can be written as $\mathbf{z} = relu(A\mathbf{x} + \mathbf{b})$. Let $\widetilde{\mathbf{x}}$ be a high-dimensional interval $[\mathbf{x} - \delta, \mathbf{x} + \delta]$. And we denote the range of the affine function as $\widetilde{\mathbf{y}} = \{A\mathbf{x} + \mathbf{b} | \mathbf{x} \in \widetilde{\mathbf{x}}\}$, and the range of affine-ReLU function as $\widetilde{\mathbf{z}} = \{relu(\mathbf{y}) | \mathbf{y} \in \widetilde{\mathbf{y}}\}$.

Local Lipschitz constant upper bound

As we know, BNNs retain structures of DNN, and the output of the previous layer can become the input of the next layer. In a special case, if the output of some nodes in the previous layer is not activated in the ReLU function, the input in the next layer should be 0. Therefore, we use a diagonal binary matrix D to determine the zero elements in $\widetilde{\mathbf{x}}$. Let $\widetilde{\mathbf{x}_i}$ represent the i-th element of $\widetilde{\mathbf{x}}$, which is also an interval denoted by $\widetilde{\mathbf{x}_i} = [\underline{x_i}, \overline{x_i}]$. Then we let

$$d_i = \begin{cases} 0, & \overline{x_i} = \underline{x_i} = 0 \\ 1, & otherwise \end{cases}$$

and

$$D = diag(d_1, \ldots, d_n),$$

where the diagonal matrix $D \in \mathbb{R}^{n \times n}$, $diag : \mathbb{R}^n \to \mathbb{R}^{n \times n}$. Note that if we do not know any input indices to be zero, then D will be the identity matrix.

The following theorem is adapted from Theorem 2 of (Avant and Morgansen 2021) for BNNs by extending to affine functions with uncertain $\widetilde{A}$ and $\widetilde{\mathbf{b}}$.

Theorem 3 *Consider the affine-ReLU function $relu(\widetilde{A}\mathbf{x} + \widetilde{\mathbf{b}})$ with an input set $\widetilde{\mathbf{x}}$ and an input $\mathbf{x}_0 \in \widetilde{\mathbf{x}}$ with the diagonal binary matrix D defined above. Let $\widetilde{\mathbf{y}} = \{\widetilde{A}\mathbf{x} + \widetilde{\mathbf{b}} | \mathbf{x} \in \widetilde{\mathbf{x}}\}$, and let $\widetilde{\mathbf{y}_i}$ represent the i-th element of $\widetilde{\mathbf{y}}$, which is also an interval denoted by $\widetilde{\mathbf{y}_i} = [\underline{y_i}, \overline{y_i}]$. Define r_i and R as follows:*

$$r_i = \begin{cases} 1, & \overline{y_i} > 0 \\ 0, & \overline{y_i} \leq 0 \end{cases}$$

and

$$R = diag(r_1, \ldots, r_m).$$

Then the following is an upper bound on the affine-ReLU function's local Lipschitz constant:

$$L(x_0, \widetilde{\mathbf{x}}) \leq ||R\widetilde{A}D||.$$

Proof. Proof of this theorem can be done in a similar way to Theorem 2 of (Avant and Morgansen 2021). □

This theorem is the basis of our Lipschitz constant estimation algorithm presented next.

Algorithm $local\_L()$

The $local\_L$ algorithm is used to calculate the local Lipschitz constant for function $\phi^{\mathbf{w}}(\mathbf{x}) = \pi^{\mathbf{w}}(\mathbf{x}) - p(\mathbf{x})$ defined in Section 4.3. It accepts four parameters as input, i.e., $\mathbf{x}$: input states; δ: the size of the hyper-cube centered at $\mathbf{x}$; $\pi^{\bar{\mathbf{w}}}$: structure and weight of the BNN; and p: the polynomial in ϕ. Function $local\_L$ consists of two subroutines L_B, L_P for computing the Lipschitz constant of the BNN π and the Lipschitz constant of the polynomial p.

Details of the experiments

In this section, we will present the details of the experiments including the BNN training process, the benchmark descriptions and the intermediate results. First, we construct the MPC controller to simulate the system trajectory as the dataset for training the BNN controller. Subsequently, Variational Inference is applied to train a BNN controller π utilizing the data generated by MPC. Finally, the proposed BNN controller verification approach is applied to a series of benchmarks. All our experiments are carried out on a virtual machine with a 64GB RAM, an Intel(R) Core(TM) i9-10900K CPU, and an NVIDIA GeForce RTX 3090 super GPU.

Algorithm 1: *local Lipschitz constant (local_L)*

procedure $Local_L(\mathbf{x}\ ,\ \delta\ ,\ \pi^{\bar{\mathbf{w}}}\ ,\ p)$
  initialize nominal input $\widetilde{\mathbf{x}} = [\mathbf{x} - \delta, \mathbf{x} + \delta]$;
  $L \leftarrow L_B(\widetilde{\mathbf{x}}\ ,\ \pi^{\bar{\mathbf{w}}})$ + $L_P(\widetilde{\mathbf{x}},\ p)$;
  return L
end procedure

procedure $L_B(\widetilde{\mathbf{x}}\ ,\ \pi^{\bar{\mathbf{w}}})$
  $L_{net} \leftarrow 1$;
  initialize matrix D as an identity matrix;
  for each layer in network **do**
    $\widetilde{A}, \widetilde{\mathbf{b}} \leftarrow \pi^{\bar{\mathbf{w}}}$;
    if layer is affine-ReLU **then**
      compute: $\widetilde{\mathbf{y}} \leftarrow \widetilde{A}\widetilde{\mathbf{x}} + \widetilde{\mathbf{b}}$;
      compute: $\overline{\mathbf{y}} \leftarrow \max \widetilde{\mathbf{y}}$;
      compute: $R \leftarrow \text{diag}\{1 \text{ if } \overline{y}_i > 0 \text{ else } 0\}$;
      compute Lipschitz bound: $L \leftarrow ||R\widetilde{A}D||$;
      set nominal input for next layer:$\widetilde{\mathbf{x}} \leftarrow relu(\widetilde{A}\widetilde{\mathbf{x}} + \widetilde{\mathbf{b}})$;
    end if
    if layer is affine **then**
      compute Lipschitz constant: $L \leftarrow ||\widetilde{A}||$;
      set nominal input for next layer: $\widetilde{\mathbf{x}} \leftarrow \widetilde{A}\widetilde{\mathbf{x}} + \widetilde{\mathbf{b}}$;
    end if
    $D \leftarrow R$;
    update L_{net}: $L_{net} \leftarrow L_{net} * L$;
  end for
  return L_{net};
end procedure

procedure $L_P(\widetilde{\mathbf{x}}\ ,\ p)$
  solve: $L \leftarrow \max_{\mathbf{x}_i} \sum_{i=1}^{n} (\frac{\partial p(\mathbf{x}_i)}{\partial x_i})^2$, where $\mathbf{x}_i \in \widetilde{\mathbf{x}}$;
  return $\sqrt{L}$
end procedure

Model predictive control

Model predictive control (MPC) is based on iterative, finite-horizon optimization of a plant model (Nikolaou 2001). When sampling the present plant state at time t, a cost-minimizing control strategy is computed over a relatively short time horizon $[t, t+T]$ in the future. Specifically, an online or on-the-fly calculation is used to explore state trajectories that arise from the current state. Naturally, only the first step of the control strategy is implemented, then the plant state is sampled again and the calculations are redone from the new current state, yielding a new control and new predicted state path.

We consider the dynamical system with the form $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ where $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{u} \in U \subset \mathbb{R}^m$ denote the state of the system and the control policy, respectively. Simultaneously, it can be expressed in the following discrete form:

$$\mathbf{x}_{t+1} = \mathbf{x}_t + f(\mathbf{x}_t, \mathbf{u}_t)\Delta t,$$

where Δt represents the discrete time step. In this paper, the optimization objective of MPC it to make the system reach a steady state while ensuring safety. We define the distance between $\mathbf{x}_t$ and X_u as $dist(\mathbf{x}_t, X_u) = \min\{\|\mathbf{x}_t - \mathbf{x}\| \mid \mathbf{x} \in X_u\}$. Given a current state $\mathbf{x}_0$ and the calculation step size N, solving the optimal control policy can be reduced to the following optimization problem:

$$\begin{aligned} \min_{\mathbf{u}} J(\mathbf{x}_0) = \sum_{t=0}^{N} & -(1-\alpha-\beta)dist(\mathbf{x}_t, X_u) \\ & + \alpha \left\| f(\mathbf{x}_t, \mathbf{u}_t) \right\|_Q^2 + \beta \left\| \mathbf{u}_t \right\|_R^2, \\ s.t. \quad & \mathbf{x}_t \notin X_u, \end{aligned}$$

where α and β satisfy $\alpha, \beta \in [0, 1]$, and Q and R are predefined penalty matrices used to control the penalty ratio of each dimension to minimize the objective function and identify the best control input.

For the solving process, an initial state $\mathbf{x}_0$ is first picked from the initial set, and then the control output $\mathbf{u}_t$ is solved for this N-step according to the above optimization problem. After that, let $\mathbf{x}_0 = \mathbf{x}_1$ and solve the next step until the loss function is small enough. MPC can solve the controller accurately at each moment, but this comes at the cost of efficiency and can only be solved online. Therefore, we anticipate using MPC to generate simulation trajectories, construct training data, and generate explicit controllers in the form of BNNs.

BNN controller

For the linear benchmarks we studied, we adopt the same BNN controllers as the original paper. For the nonlinear benchmarks we train the BNN controllers from scratch employing Variational Inference (VI) (Blei, Ranganath, and Mohamed 2016; Tran et al. 2016). We use a uniform BNN structure π which is composed of three layers, an input layer, a hidden layer with 16 ReLU units and an output layer. The weight prior of BNNs is set to $\mathcal{N}(0, 0.05)$ consistently. Additional hyperparameters on the BNNs are outlined in the following table.

Table : Hyperparameters of BNN training

Parameter	Value
Optimizer	Adam
Learning rate	0.001
Iteration	20000

Probability calculation

We use the error function (erf) (Andrews 1998) to evaluate the lower bound on the probability of safe weights. Suppose the safe weight set $\tilde{W} = \{\bar{\mathbf{w}}_1, ..., \bar{\mathbf{w}}_m\}$ consists of m disjoint safe hyper-rectangles. Besides, suppose the BNN has k different weights, and each weight w follows the normal distribution $\mathcal{N}(\cdot|\mu, \Sigma)$ posterior. Denoting the j-th hyper-rectangle in $\tilde{W}$ as $\bar{\mathbf{w}}_j = [l_1^j, u_1^j] \times [l_2^j, u_2^j] \times \cdots \times [l_k^j, u_k^j]$, and then the lower bound of the probability of the set of all safe weights can be computed as follows:

$$p_{\text{safe}} = \sum_{j=1}^{m} \prod_{i=1}^{k} \frac{1}{2} \left(\text{erf}\left(\frac{\mu_i - l_i^j}{\sqrt{2\Sigma_i}} \right) - \text{erf}\left(\frac{\mu_i - u_i^j}{\sqrt{2\Sigma_i}} \right) \right).$$

Details of the benchmarks

Unstable LDS (Linear dynamical system) (Lechner et al. 2021) The linear dynamical system is defined as

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} 0.3x_2 + 0.05u \\ 0.2u \end{bmatrix},$$

with state variable $\mathbf{x} = [x_1, x_2]^T$. The initial set, system domain and unsafe set are given by

$$\begin{aligned} \Theta &= \left\{ \mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x}\|_\infty \leq 0.4 \right\}, \\ \Psi &= \left\{ \mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x}\|_\infty \leq 1.2 \right\}, \\ X_u &= \left\{ \mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x}\|_\infty \geq 1.2 \right\}, \end{aligned}$$

where $\|\cdot\|_\infty$ denotes the infinity-norm. For comparison, we employ the same BNN structure as (Lechner et al. 2021), which have two different versions. The first version places Gaussian distributions prior $\mathcal{N}(0, 0.1)$ over weights from the second layer on, and the second version with $\mathcal{N}(0, 0.05)$ prior for weights in all layers.

For the first version, the safe control set $p(\mathbf{x}) + [-\gamma^*, \gamma^*]$, the barrier certificate $B(\mathbf{x})$, and the probability measure of the safe weight set $\tilde{W}$ are computed as follows:

$$\begin{aligned} p(\mathbf{x}) = & -0.1751x_1^2 - 4.5044x_1x_2 - 8.4856x_1 \\ & - 3.8831x_2^2 - 5.9148x_2 - 0.0389, \\ \gamma^* = & \ 0.5, \\ B(\mathbf{x}) = & \ 530767.6485 - 13470.7594x_1 - 0.0015x_2 \\ & - 1468425.1260x_1^2 - 0.4049x_1x_2 \\ & - 804302.3648x_2^2, \\ P_{\tilde{W}} = & \ 0.93. \end{aligned}$$

The phase portrait of the Unstable LDS system and the computed barrier certificate are shown in Fig. 1. Also for the first version, to let people have an intuitive idea about how the safe weight set $\tilde{W}$ looks like, we provide a subset of $\tilde{W}$, denoted by S_μ, which is a hyper rectangle of dimension 17 containing the posterior mean weight μ, as well as its probability measure, as follows:

$$\begin{aligned} S_\mu = & \ [0.696, 1.272]_1 \times [-2.126, -1.550]_2 \times \\ & [1.692, 2.277]_3 \times [-2.389, -1.816]_4 \times \cdots \times \\ & [2.508, 3.092]_{15} \times [-2.063, -1.481]_{16} \times \\ & [0.696, 1.272]_{17}, \\ P_{S_\mu} = & \ 0.45. \end{aligned}$$

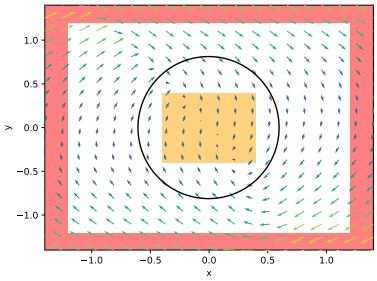

Figure 1: Phase portrait of the Unstable LDS system (Lechner et al. 2021). The zero level set of the barrier certificate $B(\mathbf{x})$ (the black curve) separates the initial set (the orange rectangle) from the unsafe set (the red area).

For the second version, the corresponding results are:

$$\begin{aligned} p(\mathbf{x}) = & -0.9619x_1^2 + 1.2904x_1x_2 - 1.2039x_1 \\ & - 0.2209x_2^2 - 3.1058x_2 + 0.1090, \\ \gamma^* = & \ 0.4, \\ B(\mathbf{x}) = & \ 1566907.2101 + 616058.7634x_1 + 1317.2776x_2 \\ & - 3712292.7097x_1^2 - 241.7785x_1x_2 \\ & - 3266622.6429x_2^2, \\ P_{\tilde{W}} = & \ 0.90, \\ S_\mu = & \ [0.237, 0.562]_1 \times [1.611, 1.924]_2 \times [0.461, 0.772]_3 \\ & \times [-0.694, -0.459]_4 \times \cdots \times [-0.009, 0.302]_{95} \\ & \times [-0.153, 0.158]_{96} \times [-0.681, -0.380]_{97}, \\ P_{S_\mu} = & \ 0.12. \end{aligned}$$

Pendulum (Lechner et al. 2021) The inverted pendulum system is defined as

$$\begin{bmatrix} \dot{\theta} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \omega \\ \frac{-3g \cdot \sin(\theta)}{2l} + \frac{7.5u}{m \cdot l^2} \end{bmatrix},$$

where the constant $g = 10.0, l = 1$ and $m = 0.8$, and state variable $\mathbf{x} = [\theta, \omega]^T$ where θ and ω denote the angle and angular velocity, respectively. Additionally, the action u refers to the torque applied to the pendulum. The initial set, system domain and unsafe set are defined by

$$\begin{aligned} \Theta &= \left\{ \mathbf{x} \in \mathbb{R}^2 \mid |\theta| \leq \pi/6 \ \wedge \ |\omega| \leq 0.2 \right\}, \\ \Psi &= \left\{ \mathbf{x} \in \mathbb{R}^2 \mid |\theta| \leq \pi \ \wedge \ |\omega| \leq 8 \right\}, \\ X_u &= \left\{ \mathbf{x} \in \mathbb{R}^2 \mid |\theta| \geq 0.9 \ \wedge \ |\omega| \geq 2 \right\}. \end{aligned}$$

Similar to the *Unstable LDS* benchmark, we employ the same BNN structure as (Lechner et al. 2021), which have two different versions. For the first version, the safe control set $p(\mathbf{x}) + [-\gamma^*, \gamma^*]$, the barrier certificate $B(\mathbf{x})$, and the probability measure of the safe weight set $\tilde{W}$ are computed

as follows:

$$\begin{aligned}
p(\mathbf{x}) = & -0.8813\theta^3 + 3.4688\theta^2\omega + 1.755\theta^2 \\
& + 1.0134\theta\omega^2 - 1.3146\theta\omega - 12.8534\theta \\
& - 1.4159\omega^3 + 0.9918\omega^2 - 6.284\omega + 1.1191, \\
\gamma^* = & \, 0.8, \\
B(\mathbf{x}) = & -0.1158\theta^2 - 0.0719\theta\omega + 0.0209\theta - 0.0437\omega^2 \\
& + 0.0250\omega + 0.0658, \\
P_{\tilde{W}} = & \, 0.96.
\end{aligned}$$

The phase portrait of the Pendulum system and the computed barrier certificate are shown in Fig. 2. We also provide the subset $S_\mu \subset \tilde{W}$, which is a hyper rectangle of dimension 17 containing the posterior mean weight μ, as well as its probability measure, as follows:

$$\begin{aligned}
S_\mu = & [1.990, 2.586]_1 \times [0.006, 0.619]_2 \times \\
& [-0.064, 0.444]_3 \times [1.447, 2.081]_4 \times \cdots \times \\
& [0.113, 0.702]_{15} \times [-0.041, 0.597]_{16} \times \\
& [1.990, 2.586]_{17}, \\
P_{S_\mu} = & \, 0.76.
\end{aligned}$$

For the second version, the corresponding results are:

$$\begin{aligned}
p(\mathbf{x}) = & -2.2353\theta^3 + 9.2215\theta^2\omega - 4.4027\theta^2 \\
& - 1.8368\theta\omega^2 + 1.5436\theta\omega - 32.4682\theta \\
& + 0.982\omega^3 + 0.9027\omega^2 - 11.4253\omega - 0.2507, \\
\gamma^* = & \, 1.8, \\
B(\mathbf{x}) = & -0.0898\theta^2 - 0.0233\theta\omega + 0.0097\theta - 0.0157\omega^2 \\
& - 0.0022\omega + 0.0487, \\
P_{\tilde{W}} = & \, 0.82, \\
S_\mu = & [2.219, 2.507]_1 \times [-4.445, -4.155]_2 \\
& \times [-1.389, -1.081]_3 \times [1.922, 2.228]_4 \times \cdots \\
& \times [-3.339, -3.045]_{95} \times [-0.056, 0.244]_{96} \\
& \times [0.233, 0.527]_{97}, \\
P_{S_\mu} = & \, 0.2.
\end{aligned}$$

Collision avoidance (Lechner et al. 2021) The collision avoidance system is defined as

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} u \\ 0 \\ -1 \end{bmatrix},$$

with state variable $\mathbf{x} = [x_1, x_2, y]^T$. The initial set, system domain and unsafe set are defined as

$$\begin{aligned}
\Theta &= \left\{ \mathbf{x} \in \mathbb{R}^3 \mid |x_1| \leq 2 \ \wedge \ |x_2| \leq 2 \ \wedge \ y = 5 \right\}, \\
\Psi &= \left\{ \mathbf{x} \in \mathbb{R}^3 \mid |x_1| \leq 2 \ \wedge \ |x_2| \leq 2 \ \wedge \ y \in [-1, 5] \right\}, \\
X_u &= \left\{ \mathbf{x} \in \mathbb{R}^3 \mid |x_1 - x_2| \leq 1 \ \wedge \ y = 0 \right\}.
\end{aligned}$$

For comparison, we employ the same BNN structure as (Lechner et al. 2021), which have two different versions. For the first version, the safe control set $p(\mathbf{x}) + [-\gamma^*, \gamma^*]$, the barrier certificate $B(\mathbf{x})$, and the probability measure of the safe weight set $\tilde{W}$ are computed as follows:

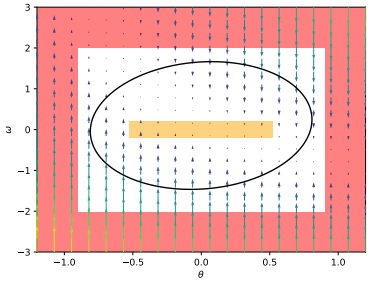

Figure 2: Phase portrait of the Pendulum system (Lechner et al. 2021). The zero level set of the barrier certificate $B(\mathbf{x})$ (the black curve) separates the initial set (the orange rectangle) from the unsafe set (the red area).

$$\begin{aligned}
p(\mathbf{x}) = & -0.0119x_1^2 + 0.0227x_1x_2 + 0.0010x_1y \\
& + 0.3602x_1 - 0.0024x_2^2 - 0.0531x_2y \\
& - 0.1425x_2 - 0.0014y^2 + 0.0072y + 0.0851, \\
\gamma^* = & \, 0.2, \\
B(\mathbf{x}) = & 0.0509x_1^2 - 0.0759x_1x_2 + 0.0090x_1y + 0.0492x_1 \\
& + 0.0198x_2^2 + 0.0532x_2y - 0.0259x_2 \\
& + 0.0403y^2 + 0.0667y - 0.1439, \\
P_{\tilde{W}} = & \, 0.99.
\end{aligned}$$

The phase portrait of the collision avoidance system is shown in Fig. 3. We provide the subset $S_\mu \subset \tilde{W}$, which is a hyper rectangle of dimension 17 containing the posterior mean weight μ, as well as its probability measure, as follows:

$$\begin{aligned}
S_\mu = & [-0.364, 0.574]_1 \times [-0.596, 0.392]_2 \times \\
& [0.104, 0.105]_3 \times [-0.132, 0.177]_4 \times \cdots \times \\
& [-0.525, 0.616]_{15} \times [0.446, 0.449]_{16} \times \\
& [-0.364, 0.574]_{17} \\
P_{S_\mu} = & \, 0.99.
\end{aligned}$$

For the second version, the corresponding results are:

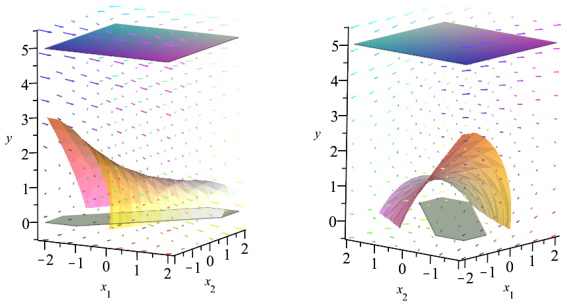

Figure 3: Phase portrait of the collision avoidance system (Lechner et al. 2021). The zero level set of the barrier certificate $B(\mathbf{x})$ (the orange surface) separates the initial set (the purple surface) from the unsafe set (the yellow surface).

$$
\begin{aligned}
p(\mathbf{x}) =& 0.0421x_1^2 + 0.0977x_1x_2 - 0.0306x_1y + 0.5049x_1 \\
&+ 0.0796x_2^2 - 0.1062x_2y + 0.1093x_2 \\
&+ 0.0024y^2 - 0.0259y - 0.0463, \\
\gamma^* =& 0.5, \\
B(\mathbf{x}) =& 0.0573x_1^2 - 0.0785x_1x_2 + 0.0164x_1y + 0.0254x_1 \\
&+ 0.0261x_2^2 + 0.0588x_2y - 0.0287x_2 \\
&+ 0.0487y^2 + 0.0489y - 0.1206, \\
P_{\tilde{W}} =& 0.95, \\
S_\mu =& [-0.649, 0.642]_1 \times [-0.460, 0.500]_2 \times \\
&[0.002, 0.011]_3 \times [-0.112, 0.136]_4 \times \cdots \times \\
&[-0.258, 0.090]_{127} \times [0.989, 0.991]_{128} \times \\
&[-0.267, 0.108]_{129}, \\
P_{S_\mu} =& 0.71.
\end{aligned}
$$

Dubin's Car (Deshmukh et al. 2019) The Dubin's Car system is defined as

$$
\begin{bmatrix} \dot{d} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \sin\theta \\ -u \end{bmatrix},
$$

with state variables $\mathbf{x} = [d, \theta]^T$ where d and θ stand for the distance error and angle error between the current position and the target position. The initial set, system domain and unsafe set are defined as

$$
\begin{aligned}
\Theta &= \left\{\mathbf{x} \in \mathbb{R}^2 \mid [-1, -\pi/16]^T \leq \mathbf{x} \leq [1, \pi/6]^T\right\}, \\
\Psi &= \left\{\mathbf{x} \in \mathbb{R}^2 \mid [-6, -7\pi/10]^T \leq \mathbf{x} \leq [6, 7\pi/10]^T\right\}, \\
X_u &= \left\{\mathbf{x} \in \mathbb{R}^2 \mid [-5, -\pi/2]^T \leq \mathbf{x} \leq [5, \pi/2]^T\right\}.
\end{aligned}
$$

We train a BNN controller composed of a hidden layer with 16 ReLU units. For this benchmark, the safe control set $p(\mathbf{x}) + [-\gamma^*, \gamma^*]$, the barrier certificate $B(\mathbf{x})$, and the probability measure of the safe weight set $\tilde{W}$ are computed as follows:

$$
\begin{aligned}
p(\mathbf{x}) =& -0.3095d^2 - 2.2112d\theta + 4.8151d - 2.4769\theta^2 \\
&+ 7.1023\theta + 0.0524, \\
\gamma^* =& 0.3, \\
B(\mathbf{x}) =& -0.094d^2 - 0.1571d\theta - 0.0048d - 0.1310\theta^2 \\
&- 0.0185\theta + 0.1384, \\
P_{\tilde{W}} =& 0.96.
\end{aligned}
$$

We provide the subset $S_\mu \subset \tilde{W}$, which is a hyper rectangle of dimension 65 containing the posterior mean weight μ, as well as its probability measure, as follows:

$$
\begin{aligned}
S_\mu =& [-0.179, 0.009]_1 \times [-0.805, 0.084]_2 \times \\
&[2.038, 2.072]_3 \times [0.035, 0.118]_4 \times \cdots \times \\
&[-0.378, -0.291]_{63} \times [0.834, 0.834]_{64} \times \\
&[-1.159, -0.685]_{65}, \\
P_{S_\mu} =& 0.54.
\end{aligned}
$$

Oscillator (Zhu et al. 2019) The Van der Pol's oscillator system is defined as

$$
\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} y \\ \beta(1 - x^2)y - x + u \end{bmatrix},
$$

with the constant $\beta = 1$. We denote the system state variable as $\mathbf{x} = [x, y]^T$. The initial set, system domain and unsafe set are defined as

$$
\begin{aligned}
\Theta &= \left\{\mathbf{x} \in \mathbb{R}^2 \mid [-0.51, 0.49]^T \leq \mathbf{x} \leq [-0.49, 0.51]^T\right\}, \\
\Psi &= \left\{\mathbf{x} \in \mathbb{R}^2 \mid [-2, -2]^T \leq \mathbf{x} \leq [2, 2]^T\right\}, \\
X_u &= \left\{\mathbf{x} \in \mathbb{R}^2 \mid [-0.4, 0.2]^T \leq \mathbf{x} \leq [0.1, 0.35]^T\right\}.
\end{aligned}
$$

We train a BNN controller consisting of a hidden layer with 16 ReLU units. For this benchmark, the safe control set $p(\mathbf{x}) + [-\gamma^*, \gamma^*]$, the barrier certificate $B(\mathbf{x})$, and the probability measure of the safe weight set $\tilde{W}$ are computed as follows:

$$
\begin{aligned}
p(\mathbf{x}) =& 0.1194x^2 - 0.0356xy + 0.8706x - 0.0150y^2 \\
&- 6.6555y + 0.2979, \\
\gamma^* =& 0.1, \\
B(\mathbf{x}) =& 0.1283x^2 - 0.1983xy + 0.0773x - 0.0333y^2 \\
&- 0.0904y + 0.0139, \\
P_{\tilde{W}} =& 0.98.
\end{aligned}
$$

We provide the subset $S_\mu \subset \tilde{W}$, which is a hyper rectangle of dimension 65 containing the posterior mean weight μ, as well as its probability measure, as follows:

$$
\begin{aligned}
S_\mu =& [0.464, 0.865]_1 \times [-0.016, 0.069]_2 \times \\
&[0.086, 1.807]_3 \times [-0.471, 0.275]_4 \times \cdots \times \\
&[-3.984, 1.338]_{63} \times [-0.220, -0.220]_{64} \times \\
&[-1.622, -0.434]_{65}, \\
P_{S_\mu} =& 0.84.
\end{aligned}
$$

Academic 3D (Deshmukh et al. 2019) The Academic 3D system is defined as

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix} = \begin{bmatrix} z + 8y \\ -y + z \\ -z - x^2 + u \end{bmatrix},$$

with state variables $\mathbf{x} = [x, y, z]^T$. The initial set, system domain and unsafe set are defined as

$$\begin{aligned}\Theta &= \left\{\mathbf{x} \in \mathbb{R}^3 \mid \left\|\mathbf{x} - [-0.75, -1, -0.4]^T\right\| \leq 0.35\right\},\\ \Psi &= \left\{\mathbf{x} \in \mathbb{R}^3 \mid [-5, -5, -5]^T \leq \mathbf{x} \leq [5, 5, 5]^T\right\},\\ X_u &= \left\{\mathbf{x} \in \mathbb{R}^3 \mid \left\|\mathbf{x} - [-0.3, -0.36, 0.2]^T\right\| \leq 0.30\right\}.\end{aligned}$$

where $\|\cdot\|$ denotes the 2-norm. We train a BNN policy made up of a hidden layer with 16 ReLU units. For this benchmark, the safe control set $p(\mathbf{x}) + [-\gamma^*, \gamma^*]$, the barrier certificate $B(\mathbf{x})$, and the probability measure of the safe weight set $\tilde{W}$ are computed as follows:

$$\begin{aligned}p(\mathbf{x}) =& 1.9252x^2 + 1.3410xy + 2.3829xz - 3.3200x\\ &- 1.5971y^2 - 9.6709yz - 5.6955y - 12.4129z^2\\ &- 10.4166z + 0.1471,\\ \gamma^* =& 1.6,\\ B(\mathbf{x}) =& -0.0018x^4 + 0.0188x^3y + 0.008x^3z\\ &- 0.0138x^3 - 0.0071x^2y^2 + 0.0425x^2yz\\ &- 0.0088x^2y + 0.0409x^2z^2 - 0.0533x^2z\\ &+ 0.0330x^2 + 0.0189xy^3 + 0.0394xy^2z\\ &+ 0.0007xy^2 + 0.0586xyz^2 + 0.0116xyz\\ &- 0.0291xy + 0.0252xz^3 - 0.0488xz^2\\ &+ 0.0109xz + 0.0144x + 0.0039y^4 - 0.0044y^3z\\ &- 0.0060y^3 + 0.0023y^2z^2 - 0.0043y^2z\\ &+ 0.0162y^2 + 0.0373yz^3 + 0.0350yz^2\\ &+ 0.0153yz + 0.0142y + 0.0057z^4 - 0.0230z^3\\ &- 0.0213z^2 + 0.0191z - 0.0008,\\ P_{\tilde{W}} =& 0.96.\end{aligned}$$

We provide the subset $S_\mu \subset \tilde{W}$, which is a hyper rectangle of dimension 81 containing the posterior mean weight μ, as well as its probability measure, as follows:

$$\begin{aligned}S_\mu =& [0.697, 1.127]_1 \times [-0.421, -0.416]_2 \times\\ &[-2.026, -2.026]_3 \times [0.397, 1.482]_4 \times \cdots \times\\ &[-1.956, -0.971]_{79} \times [0.073, 0.075]_{80} \times\\ &[-0.780, -0.717]_{81},\\ P_{S_\mu} =& 0.57.\end{aligned}$$

Bicycle Steering (Deshmukh et al. 2019) The Bicycle Steering system is defined as

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{m\ell}{J}\left(g\sin(x_1) + \frac{v^2}{b}\cos(x_1)\tan(x_3)\right) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{am\ell v}{Jb}\frac{\cos(x_1)}{\cos^2(x_3)} \\ 1 \end{bmatrix} u,$$

with the constants set as $m = 20, b = 1, \ell = 1, J = \frac{1}{3}mb^2, v = 10, a = 0.5$ and $g = 10$. The state variables x_1, x_2 and x_3 are expressed by the symbol $\mathbf{x} = [x_1, x_2, x_3]^T$. Besides, the following are the initial set, system domain and unsafe set

$$\begin{aligned}\Theta =& \left\{\mathbf{x} \in \mathbb{R}^3 \mid [-0.2, -0.2, -0.2]^T \leq \mathbf{x}\right.\\ &\left.\leq [0.2, 0.2, 0.2]^T\right\},\\ \Psi =& \left\{\mathbf{x} \in \mathbb{R}^3 \mid [-2.2, -2.2, -2.2]^T \leq \mathbf{x}\right.\\ &\left.\leq [2.2, 2.2, 2.2]^T\right\},\\ X_u =& \left\{\mathbf{x} \in \mathbb{R}^3 \mid [-2.2, -2.2, -2.2]^T \leq \mathbf{x}\right.\\ &\left.\leq [-2, -2, -2]^T\right\}.\end{aligned}$$

We train a BNN policy consisting of a hidden layer with 16 ReLU units. For this benchmark, the safe control set $p(\mathbf{x}) + [-\gamma^*, \gamma^*]$, the barrier certificate $B(\mathbf{x})$, and the probability measure of the safe weight set $\tilde{W}$ are computed as follows:

$$\begin{aligned}p(\mathbf{x}) =& -2.0968x_1^2 - 7.1743x_1x_2 + 3.4830x_1x_3\\ &- 7.1534x_1 - 4.0626x_2^2 + 5.9221x_2x_3\\ &- 4.6882x_2 - 3.9432x_3^2 - 15.2565x_3 + 0.0332,\\ \gamma^* =& 0.7,\\ B(\mathbf{x}) =& 0.0896x_1^2 - 0.1627x_1x_2 - 0.0423x_1x_3\\ &+ 0.0315x_1 + 0.0535x_2^2 + 0.0734x_2x_3\\ &+ 0.0407x_2 + 0.0698x_3^2 + 0.1364x_3 + 0.0443,\\ P_{\tilde{W}} =& 0.96.\end{aligned}$$

We provide the subset $S_\mu \subset \tilde{W}$, which is a hyper rectangle of dimension 81 containing the posterior mean weight μ, as well as its probability measure, as follows:

$$\begin{aligned}S_\mu =& [-0.155, 1.326]_1 \times [0.064, 0.326]_2 \times\\ &[-0.227, 0.119]_3 \times [-0.337, -0.337]_4 \times \cdots \times\\ &[1.610, 1.610]_{79} \times [-0.964, -0.963]_{80} \times\\ &[-1.470, -0.953]_{81},\\ P_{S_\mu} =& 0.80.\end{aligned}$$

Chesi 3 (Chesi 2004) The dynamic system is defined as

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix} -x_1 - x_4 + u \\ x_1 - x_2 + x_1^2 + u \\ -x_3 + x_4 + x_2^2 \\ x_1 - x_2 - x_4 + x_3^3 - x_4^3 \end{bmatrix},$$

with state variables $\mathbf{x} = [x_1, x_2, x_3, x_4]^T$. Besides, the following are the initial set, system domain and unsafe set

$$\begin{aligned}\Theta &= \left\{\mathbf{x} \in \mathbb{R}^4 \mid [-0.2, -0.2, -0.2, -0.2]^T \leq \mathbf{x} \right.\\ &\quad \left. \leq [0.2, 0.2, 0.2, 0.2]^T\right\},\\ \Psi &= \left\{\mathbf{x} \in \mathbb{R}^4 \mid \|\mathbf{x}\| \leq 16\right\},\\ X_u &= \left\{\mathbf{x} \in \mathbb{R}^4 \mid \left\|\mathbf{x} - [2, 2, 2, 2]^T\right\| \leq 1\right\}.\end{aligned}$$

We train a BNN policy consisting of a hidden layer with 16 ReLU units. For this benchmark, the safe control set $p(\mathbf{x}) + [-\gamma^*, \gamma^*]$, the barrier certificate $B(\mathbf{x})$, and the probability measure of the safe weight set $\tilde{W}$ are computed as follows:

$$\begin{aligned}p(\mathbf{x}) =& 0.1929x_1^2 + 0.7244x_1x_2 - 1.0765x_1x_3\\ &+ 0.4606x_1x_4 + 0.2695x_1 - 0.1219x_2^2\\ &- 0.1762x_2x_3 + 0.3806x_2x_4 + 0.2842x_2\\ &+ 0.4491x_3^2 - 0.9554x_3x_4 - 0.1810x_3\\ &+ 0.1767x_4^2 + 0.0464x_4 + 0.4008,\\ \gamma^* =& 0.1,\\ B(\mathbf{x}) =& -0.0586x_1^2 + 0.0118x_1x_2 + 0.0157x_1x_3\\ &+ 0.1143x_1x_4 + 0.0330x_1 - 0.0887x_2^2\\ &+ 0.0074x_2x_3 + 0.0208x_2x_4 + 0.0586x_2\\ &- 0.0140x_3^2 + 0.0020x_3x_4 + 0.0231x_3\\ &- 0.1094x_4^2 - 0.0815x_4 + 0.1981,\\ P_{\tilde{W}} =& 0.96.\end{aligned}$$

We provide the subset $S_\mu \subset \tilde{W}$, which is a hyper rectangle of dimension 97 containing the posterior mean weight μ, as well as its probability measure, as follows:

$$\begin{aligned}S_\mu =& [-0.169, -0.040]_1 \times [-0.421, 0.225]_2 \times\\ &[-0.454, 0.139]_3 \times [-0.888, -0.321]_4 \times ... \times\\ &[-1.027, -0.308]_{95} \times [0.407, 0.409]_{96} \times\\ &[-0.605, -0.250]_{97},\\ P_{S_\mu} =& 0.70.\end{aligned}$$

LALO20 (Laub and Loomis 1998) The dynamic system is defined as

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \\ \dot{x_5} \\ \dot{x_6} \\ \dot{x_7} \end{bmatrix} = \begin{bmatrix} 1.4x_3 - 0.9x_1 \\ 2.5x_5 - 1.5x_2 + u \\ 0.6x_7 - 0.8x_2x_3 \\ 2 - 1.3x_3x_4 \\ 0.7x_1 - x_4x_5 \\ 0.3x_1 - 3.1x_6 \\ 1.8x_6 - 1.5x_2x_7 \end{bmatrix},$$

with state variable $\mathbf{x} = [x_1, x_2, x_3, x_4, x_5, x_6, x_7]^T$. Besides, the initial set, system domain and unsafe set are as follows:

$$\begin{aligned}\Theta &= \left\{\mathbf{x} \in \mathbb{R}^7 \mid \mathbf{a}_1 \leq \mathbf{x} \leq \mathbf{a}_2\right\},\\ \Psi &= \left\{\mathbf{x} \in \mathbb{R}^7 \mid \mathbf{b}_1 \leq \mathbf{x} \leq \mathbf{b}_2\right\},\\ X_u &= \left\{\mathbf{x} \in \mathbb{R}^7 \mid \mathbf{c}_1 \leq \mathbf{x} \leq \mathbf{c}_2\right\},\end{aligned}$$

where

$$\begin{aligned}\mathbf{a}_1 &= [1.15, 1, 1.45, 2.35, 0.95, 0.05, 0.40]^T,\\ \mathbf{a}_2 &= [1.25, 1.10, 1.55, 2.45, 1.05, 0.15, 0.50]^T,\\ \mathbf{b}_1 &= [-3.8, -3.95, -3.5, -2.6, -4, -4.9, -4.55]^T,\\ \mathbf{b}_2 &= [6.2, 6.05, 6.5, 7.4, 6, 5.1, 5.45]^T,\\ \mathbf{c}_1 &= [-3.3, -3.45, -3, -2.1, -3.5, -4.4, -4.05]^T,\\ \mathbf{c}_2 &= [1.25, 1.1, 1.55, 2.45, 1.05, 1.05, 0.5]^T.\end{aligned}$$

We train a BNN policy consisting of a hidden layer with 16 ReLU units. For this benchmark, the safe control set $p(\mathbf{x}) + [-\gamma^*, \gamma^*]$, the barrier certificate $B(\mathbf{x})$, and the probability measure of the safe weight set $\tilde{W}$ are computed as follows:

$$\begin{aligned}p(\mathbf{x}) =& 0.3632x_1^2 + 0.4139x_1x_2 - 0.2586x_1x_3\\ &+ 0.2487x_1x_4 - 0.6496x_1x_5 - 0.5303x_1x_6\\ &+ 1.8798x_1x_7 - 1.6597x_1 + 0.0316x_2^2\\ &- 0.1971x_1x_3 + 0.2116x_1x_4 + 0.1894x_1x_5\\ &- 0.7529x_1x_6 + 0.1739x_1x_7 - 0.7287x_1\\ &- 0.0041x_3^2 - 0.1067x_3x_4 + 0.0231x_3x_5\\ &- 0.231x_3x_6 - 0.146x_3x_7 + 0.8764x_3\\ &+ 0.06x_4^2 - 0.3485x_4x_5 + 0.0728x_4x_6\\ &+ 0.8098x_4x_7 - 0.6889x_4 + 0.055x_5^2\\ &- 0.957x_5x_6 + 0.1202x_5x_7 + 1.4234x_5\\ &- 0.2866x_6^2 + 0.4249x_6x_7 + 2.4245x_6\\ &+ 0.1185x_7^2 - 4.5185x_7 + 1.619,\\ \gamma^* =& 0.1,\\ B(\mathbf{x}) =& -0.0409x_1^2 - 0.0409x_1x_2 - 0.0152x_1x_3\\ &+ 0.0282x_1x_4 - 0.0038x_1x_5 - 0.0104x_1x_6\\ &+ 0.0281x_1x_7 - 0.0289x_1 - 0.04x_2^2\\ &- 0.0086x_2x_3 + 0.0045x_2x_4 + 0.006x_2x_5\\ &- 0.0184x_2x_6 + 0.0197x_2x_7 - 0.0176x_2\\ &- 0.0314x_3^2 + 0.0211x_3x_4 - 0.0284x_3x_5\\ &+ 0.0146x_3x_6 - 0.0138x_3x_7 + 0.01x_3\\ &- 0.0188x_4^2 + 0.0217x_4x_5 - 0.0037x_4x_6\\ &+ 0.0152x_4x_7 - 0.0158x_4 - 0.0294x_5^2\\ &- 0.0104x_5x_6 - 0.0449x_5x_7 + 0.0107x_5\\ &- 0.0202x_6^2 + 0.0255x_6x_7 + 0.0099x_6\\ &- 0.0517x_7^2 + 0.0101x_7 + 0.3105,\\ P_{\tilde{W}} =& 0.95.\end{aligned}$$

We provide the subset $S_\mu \subset \tilde{W}$, which is a hyper rectangle of dimension 145 containing the posterior mean weight μ, as well as its probability measure, as follows:

$$\begin{aligned}S_\mu =& [-1.371, -0.964]_1 \times [-2.468, 2.020]_2 \times\\ &[-1.433, 0.869]_3 \times [-0.933, -0.561]_4 \times \cdots \times\\ &[-0.099, -0.099]_{143} \times [-0.268, -0.267]_{144} \times\\ &[-3.052, 0.989]_{145},\\ P_{S_\mu} =& 0.58.\end{aligned}$$

References

Andrews, L. C. 1998. *Special functions of mathematics for engineers*, volume 49. Spie Press.

Avant, T.; and Morgansen, K. A. 2021. Analytical bounds on the local Lipschitz constants of ReLU networks. *CoRR*, abs/2104.14672.

Blei, D.; Ranganath, R.; and Mohamed, S. 2016. Variational inference: Foundations and modern methods. *NIPS Tutorial*.

Chesi, G. 2004. Computing output feedback controllers to enlarge the domain of attraction in polynomial systems. *IEEE Transactions on Automatic Control*, 49(10): 1846–1853.

Deshmukh, J. V.; Kapinski, J. P.; Yamaguchi, T.; and Prokhorov, D. 2019. Learning deep neural network controllers for dynamical systems with safety guarantees. In *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 1–7. IEEE.

Laub, M. T.; and Loomis, W. F. 1998. A molecular network that produces spontaneous oscillations in excitable cells of Dictyostelium. *Molecular biology of the cell*, 9(12): 3521–3532.

Lechner, M.; Žikelić, D. o. e.; Chatterjee, K.; and Henzinger, T. 2021. Infinite Time Horizon Safety of Bayesian Neural Networks. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 10171–10185. Curran Associates, Inc.

Nikolaou, M. 2001. Model predictive controllers: A critical synthesis of theory and industrial needs.

Prajna, S.; Jadbabaie, A.; and Pappas, G. J. 2007. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8): 1415–1429.

Tran, D.; Kucukelbir, A.; Dieng, A. B.; Rudolph, M.; Liang, D.; and Blei, D. M. 2016. Edward: A library for probabilistic modeling, inference, and criticism. *arXiv preprint arXiv:1610.09787*.

Zhu, H.; Xiong, Z.; Magill, S.; and Jagannathan, S. 2019. An inductive synthesis framework for verifiable reinforcement learning. *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation*.