# A Supplementary Material

In what follows, we give some details of content omitted in the paper due to space limit. The supplements are organized as follows. We give some proof of Lemma 1, 2, Proposition 1, Lemma 3, 4, and Theorem 2 in Section A.1 –A.6, respectively. We present system dynamics for Darboux, obstacle avoidance, spacecraft rendezvous and hi-ord$_8$ in Section A.9–A.12. We provide some training details in Section A.13 as well as experiment details and results in Section A.14. We compare polynomial CBFs with NCBF in A.15, compare NCBFs with different activation functions in A.16. We present more details in A.17 on the example in 3.2.

## A.1 Proof of Lemma 1

We prove by induction on $L$. If $L = 1$, then $x \in \overline{\mathcal{X}}(\mathbf{S})$ if the pre-activation input to the $(1, j)$ neuron is nonnegative for all $j \in S_1$ and nonpositive for all $j \notin S_1$. We have that the pre-activation input is equal to $W_{1j}^T x + r_{1j}$, establishing the result for $L = 1$.

Now, inducting on $L$, we have that $x \in \overline{\mathcal{X}}(S_1, \ldots, S_{L-1})$ if and only if

$$x \in \bigcap_{i=1}^{L-1} \left( \bigcap_{j \in S_i} \{x : W_{ij}^T (\overline{\mathbf{W}}_{i-1}(\mathbf{S})^T x + \overline{\mathbf{r}}_{i-1}) + r_{ij} \geq 0\} \right.$$
$$\left. \cap \bigcap_{j \notin S_i} \{x : W_{ij}^T (\overline{\mathbf{W}}_{i-1}(\mathbf{S})^T x + \overline{\mathbf{r}}_{i-1}) + r_{ij} \leq 0\} \right)$$

by induction. If $x \in \overline{\mathcal{X}}(S_1, \ldots, S_{L-1})$, then $x \in \overline{\mathcal{X}}(S_L)$ if and only if the pre-activation input to the $j$-th neuron at layer $L$ is nonnegative for all $j \in S_L$ and nonpositive for $j \notin S_L$. The pre-activation input is equal to $W_{Lj}^T z_{L-1} + r_{Lj}$, which we can expand by induction as

$$\begin{aligned}
W_{Lj}^T z_{L-1} + r_{Lj} &= \sum_{j'=1}^{M_{L-1}} (W_{Lj})_{j'} z_{L-1,j'} + r_{Lj} \\
&= \sum_{j'=1}^{M_{L-1}} (W_{Lj})_{j'} (\overline{\mathbf{W}}_{L-1,j'}(\mathbf{S})^T x + \overline{\mathbf{r}_{L-1,j'}}(\mathbf{S}) + r_{Lj} \\
&= \left( \sum_{j'=1}^{M_{L-1}} (W_{Lj})_{j'} \overline{W}_{L-1,j'}(\mathbf{S}) \right)^T x + \overline{\mathbf{r}}_{Lj}(\mathbf{S}) \\
&= (\overline{\mathbf{W}}_{L-1}(\mathbf{S}) W_{Lj})^T x + \overline{\mathbf{r}}_{Lj}(\mathbf{S})
\end{aligned}$$

completing the proof.

## A.2 Proof of Lemma 2

The proof approach is based on Nagumo's Theorem, which gives necessary and sufficient conditions for positive invariance of a set. We first define the concept of tangent cone, and then present positive invariance conditions based on the tangent cone. The approach of the proof is to characterize the tangent cone to the set $\mathcal{D} = \{x : b(x) \geq 0\}$.

**Definition 2.** *Let $\mathcal{A}$ be a closed set. The tangent cone to $\mathcal{A}$ at $x$ is defined by*

$$\mathcal{T}_{\mathcal{A}}(x) = \left\{ z : \liminf_{\tau \to 0} \frac{dist(x + \tau z, \mathcal{A})}{\tau} = 0 \right\} \tag{19}$$

The following result gives an approach for constructing the tangent cone.

**Lemma 5** ([36]). *Suppose that the set $\mathcal{A}$ is defined by*

$$\mathcal{A} = \{x : q_k(x) \leq 0, k = 1, \ldots, N\}$$

*for some collection of differentiable functions $q_1, \ldots, q_N$. For any $x$, let $J(x) = \{k : q_k(x) = 0\}$. Then*

$$\mathcal{T}_{\mathcal{A}}(x) = \{z : z^T \nabla q_k(x) \leq 0 \; \forall k \in J(x)\}.$$

The following is a fundamental preliminary result for establishing positive invariance.

**Theorem 3** (Nagumo's Theorem [36], Section 4.2)**.** *A closed set $\mathcal{A}$ is controlled positive invariant if and only if, whenever $x(t) \in \partial\mathcal{A}$, $u(t) \in \mathcal{U}$ satisfies*

$$(f(x(t)) + g(x(t))u(t)) \in \mathcal{T}_{\mathcal{A}}(x(t)) \tag{20}$$

The following lemma characterizes the tangent cone to $\mathcal{D}$.

**Proposition 2.** *For any $x \in \partial\mathcal{D}$, we have*

$$\mathcal{T}_{\mathcal{D}}(x) = \bigcup_{\mathbf{S} \in \mathbf{S}(x)} \left[ \left( \bigcap_{(i,j) \in \mathbf{T}(x) \cap \mathbf{S}} \{z : (\overline{\mathbf{W}}_{i-1}(\mathbf{S}) W_{ij})^T z \geq 0\} \right) \cap \left( \bigcap_{(i,j) \in \mathbf{T}(x) \setminus \mathbf{S}} \{z : (\overline{\mathbf{W}}_{i-1}(\mathbf{S}) W_{ij})^T z \leq 0\} \right) \cap \{z : \overline{W}(\mathbf{S})^T z \geq 0\} \right] \tag{21}$$

*Proof.* Define $\overline{\mathcal{X}}_0(\mathbf{S}) = \overline{\mathcal{X}}(\mathbf{S}) \cap \mathcal{D}$. We will first show that, for all $x$ with $b(x) = 0$,

$$\mathcal{T}_{\mathcal{D}}(x) = \bigcup_{\mathbf{S} \in \mathbf{S}(\mathbf{x})} \mathcal{T}_{\overline{\mathcal{X}}_0(\mathbf{S})}(x). \tag{22}$$

We observe that

$$\text{dist}\left(x, \mathcal{D} \setminus \bigcup_{\mathbf{S} \in \mathbf{S}(x)} \overline{\mathcal{X}}_0(\mathbf{S})\right) > 0,$$

and hence

$$\text{dist}(x + \tau z, \mathcal{D}) = \min_{\mathbf{S} \in \mathbf{S}(x)} \text{dist}(x + \tau z, \overline{\mathcal{X}}_0(\mathbf{S}))$$

for $\tau$ sufficiently small.

Suppose that $z \in \mathcal{T}_{\overline{\mathcal{X}}_0(\mathbf{S})}(x)$. Then for any $\tau \geq 0$, $\text{dist}(x + \tau z, \mathcal{D}) \leq \text{dist}(x + \tau z, \overline{\mathcal{X}}_0(\mathbf{S}))$ since $\overline{\mathcal{X}}_0(\mathbf{S}) \subseteq \mathcal{D}$, and hence

$$\liminf_{\tau \to 0} \frac{\text{dist}(x + \tau z, \mathcal{D})}{\tau} \leq \liminf_{\tau \to 0} \frac{\text{dist}(x + \tau z, \overline{\mathcal{X}}_0(\mathbf{S}))}{\tau} = 0.$$

We therefore have $z \in \mathcal{T}_{\mathcal{D}}(x)$.

Now, suppose that $z \in \mathcal{T}_{\mathcal{D}}(\mathbf{x})$ and yet $z \notin \bigcup_{\mathbf{S} \in \mathbf{S}(x)} \mathcal{T}_{\overline{\mathcal{X}}_0(\mathbf{S})}(x)$. Then for all $\mathbf{S} \in \mathbf{S}(x)$, there exists $\epsilon_{\mathbf{S}} > 0$ such that

$$\liminf_{\tau \to 0} \frac{\text{dist}(x + \tau z, \overline{\mathcal{X}}_0(\mathbf{S}))}{\tau} = \epsilon_{\mathbf{S}}.$$

Let $\overline{\epsilon} = \min\{\epsilon_{\mathbf{S}} : \mathbf{S} \in \mathbf{S}(x)\}$. For any $\delta \in (0, \overline{\epsilon})$, there exists $\overline{\tau} > 0$ such that $\tau < \overline{\tau}$ implies

$$\frac{\text{dist}(x + \tau z, \mathcal{D})}{\tau} = \min_{\mathbf{S} \in \mathbf{S}(x)} \frac{\text{dist}(x + \tau z, \overline{\mathcal{X}}_0(\mathbf{S}))}{\tau} > \delta$$

implying that $\liminf_{\tau \to 0} \frac{\text{dist}(x + \tau z, \mathcal{D})}{\tau} > 0$ and hence $z \notin \mathcal{T}_{\mathcal{D}}(x)$. This contradiction implies (22).

It now suffices to show that, for each $\mathbf{S} \in \mathbf{S}(x)$,

$$\mathcal{T}_{\overline{\mathcal{X}}_0(\mathbf{S})}(x) = \left( \bigcap_{(i,j) \in \mathbf{T}(x) \cap \mathbf{S}} \{z : (\overline{\mathbf{W}}_{i-1}(\mathbf{S}) W_{ij})^T z \geq 0\} \right) \cap \left( \bigcap_{(i,j) \in \mathbf{T}(x) \setminus \mathbf{S}} \{z : (\overline{\mathbf{W}}_{i-1}(\mathbf{S}) W_{ij})^T z \leq 0\} \right) \cap \{z : \overline{W}(\mathbf{S})^T z \geq 0\}.$$

We have that each $\overline{\mathcal{X}}_0(\mathbf{S})$ is given by

$$\overline{\mathcal{X}}_0(\mathbf{S}) = \{x' : (\overline{\mathbf{W}}_{i-1}(\mathbf{S})W_{ij})^T x' + r_{ij}(\mathbf{S}) \geq 0 \; \forall (i,j) \in \mathbf{S}\}$$
$$\cap \{x' : (\overline{\mathbf{W}}_{i-1}(\mathbf{S})W_{ij})^T x' + r_{ij}(\mathbf{S}) \leq 0 \; \forall (i,j) \notin \mathbf{S}\} \cap \{x' : \overline{W}(\mathbf{S})^T x' + r(\mathbf{S}) \geq 0\},$$

thus matching the conditions of Lemma 5 when each $g_k$ function is affine. Furthermore, the set $J(x)$ is equal to the set of functions that are exactly zero at $x$, which consists of $\{(\overline{\mathbf{W}}_{i-1}(\mathbf{S})W_{ij})^T x + \overline{r}_{ij}(\mathbf{S}) : (i,j) \in T(\mathbf{x})\}$ together with $\overline{W}(\mathbf{S})^T x + \overline{r}(\mathbf{S})$. This observation combined with Lemma 5 gives the desired result. □

Lemma 2 is a consequence of Proposition 2. For ease of exposition, we first reproduce the lemma and then present the proof.

**Lemma 6.** *The set $\mathcal{D}$ is positive invariant if and only if, for all $x \in \partial\mathcal{D}$, there exist $\mathbf{S} \in \mathbf{S}(x)$ and $u \in \mathcal{U}$ satisfying*

$$(\overline{\mathbf{W}}_{i-1}(\mathbf{S})W_{ij})^T (f(x) + g(x)u) \geq 0 \; \forall (i,j) \in \mathbf{T}(x) \cap \mathbf{S} \tag{23}$$
$$(\overline{\mathbf{W}}_{i-1}(\mathbf{S})W_{ij})^T (f(x) + g(x)u) \leq 0 \; \forall (i,j) \in \mathbf{T}(x) \setminus \mathbf{S} \tag{24}$$
$$(\overline{\mathbf{W}}_{i-1}(\mathbf{S})W_{ij})^T (f(x) + g(x)u) \geq 0 \tag{25}$$

*Proof.* By Theorem 3, the set $\mathcal{D}$ is positive invariant if and only if for every $x \in \partial\mathcal{D}$, there exists $u$ such that $(f(x) + g(x)u) \in \mathcal{T}_{\mathcal{D}}(x)$. By Proposition 2, this condition holds iff there exists $\mathbf{S} \in \mathbf{S}(x)$ such that

$$(f(x) + g(x)u) \in \left( \left( \bigcap_{(i,j) \in \mathbf{T}(x) \cap \mathbf{S}} \{z : (\overline{\mathbf{W}}_{i-1}(\mathbf{S})W_{ij})^T z \geq 0\} \right) \cap \right.$$
$$\left. \left( \bigcap_{(i,j) \in \mathbf{T}(x) \setminus \mathbf{S}} \{z : (\overline{\mathbf{W}}_{i-1}(\mathbf{S})W_{ij})^T z \leq 0\} \right) \cap \{z : \overline{W}(\mathbf{S})^T z \geq 0\} \right)$$

The above condition is equivalent to the conditions of the lemma, completing the proof. □

### A.3 Proof of Proposition 1

First, suppose that condition (i) holds. Then for any $x \in \mathcal{D}$ with $\mathbf{S}(x) = \{\mathbf{S}_1, \ldots, \mathbf{S}_r\}$, there exists $l \in \{1, \ldots, r\}$ and $u \in \mathcal{U}$ such that $x \in \overline{\mathcal{X}}(\mathbf{S}_l)$ and (7)–(8) hold. For this choice of $u$, we have $(f(x) + g(x)u) \in \mathcal{T}_{\Psi_i}(x)$ by Proposition 2. Hence $\mathcal{D}$ is positive invariant under any control policy consistent with $b$ by Lemma 2.

Next, suppose that condition (ii) holds. Since $\mathcal{D}$ is contained in the union of the activation sets $\overline{\mathcal{X}}(\mathbf{S})$, this condition implies that $\mathcal{D} \subseteq \mathcal{C}$.

### A.4 Proof of Lemma 3

Suppose that condition 1 holds. Then for any $x \in \partial\mathcal{D}$ with $\mathbf{S}(x) = \{\mathbf{S}_1, \ldots, \mathbf{S}_r\}$, there exists $l \in \{1, \ldots, r\}$ such that $x \in \overline{\mathcal{X}}(\mathbf{S}_l)$ and $u \in \mathcal{U}$ satisfy (7) and (8). For this choice of $u$, we have $(f(x) + g(x)u) \in \mathcal{T}_{\mathcal{D}}(x)$ by Proposition 2. Hence $\mathcal{D}$ is positive invariant under any control policy consistent with $b$ by Theorem 3.

If Condition 2 holds, then there is no $x$ with $b(x) = 0$ and $x \in \text{int}(\overline{\mathcal{X}}(\mathbf{S}))$ such that $x \notin \mathcal{C}$. Hence, there are no counterexamples to condition (ii) of Proposition 1.

### A.5 Proof of Lemma 4

The approach is to prove that condition (ii) of Proposition 1 holds; condition (i) holds automatically if each $\mathbf{S}_1, \ldots, \mathbf{S}_r$ satisfies condition (ii) of Lemma 3. We have that conditions (a) and (b) are equivalent

to $b(x) = 0$ and (11). In order for $x$ to be a safety counterexample, for all $l = 1, \ldots, r$, at least one of Eqs. (7) and (8) must fail. Equivalently, for all $l = 1, \ldots, r$, there does not exist $u$ satisfying

$$\begin{aligned} -(\overline{\mathbf{W}}_{i-1}(\mathbf{S}_l)W_{ij})^T g(x)u &\leq (\overline{\mathbf{W}}_{i-1}(\mathbf{S}_l)W_{ij})^T f(x) \; \forall (i,j) \in T(\mathbf{S}_1, \ldots, \mathbf{S}_r) \cap \mathbf{S}_l \\ -(\overline{\mathbf{W}}_{i-1}(\mathbf{S}_l)W_{ij})^T g(x)u &\geq (\overline{\mathbf{W}}_{i-1}(\mathbf{S}_l)W_{ij})^T f(x) \; \forall (i,j) \in T(\mathbf{S}_1, \ldots, \mathbf{S}_r) \setminus \mathbf{S}_l \\ -\overline{W}_{ij}(\mathbf{S}_l)^T g(x)u &\leq \overline{W}(\mathbf{S}_l)^T f(x) \\ Au &\leq c \end{aligned}$$

By Farkas Lemma, non-existence of such a $u$ is equivalent to existence of $y_l$ satisfying $y_l \geq 0$ as well as (17) and (18).

### A.6 Proof of Theorem 2

Suppose that $x$ is a safety counterexample for the NCBF $b$ with $b(x) = 0$. If $x \in \text{int}\overline{\mathcal{X}}(\mathbf{S})$ for some $\mathbf{S}$, then we have that $\mathbf{S} \in \mathcal{S}$ and hence a contradiction with Lemma 3. If $x \in \overline{\mathcal{X}}(\mathbf{S}_1) \cap \cdots \overline{\mathcal{X}}(\mathbf{S}_r)$ for some $\mathbf{S}_1, \ldots, \mathbf{S}_r$, then there is a contradiction with Lemma 4.

### A.7 Details on the IBP Procedure

Interval bound propagation aims to compute an interval of possible output values by propagating a range of inputs layer-by-layer, and is integrated into our approach as follows. We first use partition the state space into cells and, for each cell, use LiRPA to derive upper and lower bounds on the value of b(x) when x takes values in that cell. When the interval of possible b(x) values in a cell contains zero, we conclude that that cell may intersect the boundary b(x) = 0. For each neuron, we use IBP to compute the pre-activation input interval for values of x within the cell. When the pre-activation input has a positive upper bound and negative lower bound, we identify the neuron as unstable, i.e., it may be either positive or negative for values of $x$ within the cell. Using this approach, we enumerate a collection of activation sets $\mathcal{S}$. We then identify the activation sets $\mathbf{S} \in \tilde{\mathcal{S}}$ such that $b(x) = 0$ for some $x \in \overline{\mathcal{X}}(\mathbf{S})$ by searching for an $x$ that satisfies the linear constraints in (16). This approach uses LiRPA and IBP to identify the activation regions that intersect the boundary $\{x : b(x) = 0\}$ without enumerating and checking all possible activation sets, which would have exponential runtime in the number of neurons in the network.

### A.8 Nonlinear Programming

The condition 2 of Lemma 3 suffices to solve the nonlinear program

$$\begin{array}{ll} \text{minimize} & h(x) \\ \text{s.t.} & \overline{W}_{ij}(\mathbf{S})^T x + \overline{r}_{ij}(\mathbf{S}) \geq 0 \; \forall (i,j) \in \mathbf{S} \\ & \overline{W}_{ij}(\mathbf{S})^T x + \overline{r}_{ij}(\mathbf{S}) \leq 0 \; \forall (i,j) \notin \mathbf{S} \\ & \overline{W}(\mathbf{S})^T x + \overline{r}(\mathbf{S}) = 0 \end{array} \tag{26}$$

and check whether the optimal value is nonnegative (unsafe) or negative (safe).

The verification problem of Lemma 4 can then be mapped to solving the nonlinear program

$$\begin{array}{ll} \min_{x, y_1, \ldots, y_r} & \max_{l=1,\ldots,r} \left\{ y_l^T \Lambda_l(\mathbf{S}_1, \ldots, \mathbf{S}_r, x) \right\} \\ \text{s.t.} & (\overline{\mathbf{W}}_{i-1}(\mathbf{S}_1)W_{ij})^T x + \overline{r}_{ij}(\mathbf{S}_1) < 0 \; \forall (i,j) \notin S_1 \cup \cdots \cup S_r \\ & (\overline{\mathbf{W}}_{i-1}(\mathbf{S}_1)W_{ij})^T x + \overline{r}_{ij}(\mathbf{S}_1) > 0 \; \forall (i,j) \in S_1 \cap \cdots \cap S_r \\ & (\overline{\mathbf{W}}_{i-1}(\mathbf{S}_1)W_{ij})^T x + \overline{r}_{ij}(\mathbf{S}_1) = 0 \; \forall (i,j) \in \mathbf{T}(\mathbf{S}_1, \ldots, \mathbf{S}_r) \\ & y_l^T \Theta_l(\mathbf{S}_1, \ldots, \mathbf{S}_r(x)) = 0 \; \forall l = 1, \ldots, r \\ & y_l \geq 0 \; \forall l = 1, \ldots, r \end{array} \tag{27}$$

and checking whether the optimal value is nonnegative (safe) or negative (unsafe).

### A.9 Experiment Settings: Darboux

We show the settings of NCBF verification for Darboux system whose dynamic is defined as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 + 2x_1 x_2 \\ -x_1 + 2x_1^2 - x_2^2 \end{bmatrix}. \tag{28}$$

We define state space, initial region, and unsafe region as $\mathcal{X} : \{\mathbf{x} \in \mathbb{R}^2 : x \in [-2, 2] \times [-2, 2]\}$, $\mathcal{X}_I : \{\mathbf{x} \in \mathbb{R}^2 : 0 \leq x_1 \leq 1, 1 \leq x_2 \leq 2\}$ and $\mathbf{x}_U : \{\mathbf{x} \in \mathbb{R}^2 : x_1 + x_2^2 \leq 0\}$ respectively.

### A.10 Experiment Settings: Obstacle Avoidance

We next evaluate that our proposed method on a controlled system [39]. The system state consists of 2-D position and aircraft yaw rate $x := [x_1, x_2, \psi]^T$. We let $u$ denote the control input to manipulate yaw rate and define the dynamics as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} v \sin \psi \\ v \cos \psi \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ u \end{bmatrix}. \tag{29}$$

We define the state space, initial region and unsafe region as $\mathcal{X}$, $\mathcal{X}_I$ and $\mathcal{X}_U$, respectively as

$$\begin{aligned} \mathcal{X} &: \{\mathbf{x} \in \mathbb{R}^3 : x_1, x_2, \psi \in [-2, 2] \times [-2, 2] \times [-2, 2]\} \\ \mathcal{X}_I &: \{\mathbf{x} \in \mathbb{R}^3 : -0.1 \leq x_1 \leq 0.1, -2 \leq x_2 \leq -1.8,\ -\pi/6 < \psi < \pi/6\} \\ \mathcal{X}_U &: \{\mathbf{x} \in \mathbb{R}^3 : x_1^2 + x_2^2 \leq 0.04\} \end{aligned} \tag{30}$$

### A.11 Experiment Settings: Spacecraft Rendezvous

The state of the chaser is expressed relative to the target using linearized Clohessy–Wiltshire–Hill equations, with state $x = [p_x, p_y, p_z, v_x, v_y, v_z]^T$, control input $u = [u_x, u_y, u_z]^T$ and dynamics defined as follows.

$$\begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{p}_z \\ \dot{v}_x \\ \dot{v}_y \\ \dot{v}_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \\ v_x \\ v_y \\ v_z \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}. \tag{31}$$

We define the state space and unsafe region as $\mathcal{X}$ and $\mathcal{X}_U$, respectively as

$$\begin{aligned} \mathcal{X} &: \{\mathbf{x} \in \mathbb{R}^3 : p, v, \in [-1.5, 1.5] \times [-1.5, 1.5]\} \\ \mathcal{X}_U &: \left\{0.25 \leq r \leq 1.5, \text{where } r = \sqrt{p_x^2 + p_y^2 + p_z^2}\right\} \end{aligned} \tag{32}$$

We obtain the trained NCBF with neural CLBF training in [13] with a nominal model predictive controller.

### A.12 Experiment Settings: hi-ord$_8$

The dynamic model of the system is captured by an ODE as follows.

$$x^{(8)} + 20x^{(7)} + 170x^{(6)} + 800x^{(5)} + 2273x^{(4)} + 3980x^{(3)} + 4180x^{(2)} + 2400x^{(1)} + 576 = 0 \tag{33}$$

where we denote the $i$-th derivative of variable $x$ by $x^{(i)}$. We define the state space and unsafe region as $\mathcal{X}$ and $\mathcal{X}_U$, respectively as

$$\begin{aligned} \mathcal{X} &: \{x_1^2 + \ldots + x_8^2 \leq 4\} \\ \mathcal{X}_U &: \left\{(x_1 + 2)^2 + \ldots + (x_8 + 2)^2 \leq 0.16\right\} \end{aligned} \tag{34}$$

We obtain the trained NCBFs with training method proposed in [21].

### A.13 Training Details

We trained the NCBFs for Darboux and obstacle avoidance via the approach proposed in [23]. Models are trained with their open source code [1] with default settings. Detailed parameters for both cases listed in Table 3a.

[1] https://github.com/zhaohj2017/HSCC20-Repeatability

We then trained NCBFs for Spacecraft Rendezvous by following approach proposed in [13] with the empirical loss defined in Eq. (5) in [12]. Models are trained with their open source code [2] with default settings. The hyper-parameters are listed in Table 3b.

Table 3: Hyper-parameters for training NCBFs to be verified

(a) Hyper-parameters for Darboux and OA

| Hyper-Parameters | Value |
|---|---|
| LEARNING_RATE | 0.01 |
| LOSS_OPT_FLAG | 1e-16 |
| TOL_MAX_GRAD | 6 |
| EPOCHS | 5 |
| TOL_INIT | 0.0 |
| TOL_SAFE | 0.0 |
| TOL_BOUNDARY | 0.05 |
| TOL_LIE | 0.0 |
| TOL_NORM_LIE | 0.0 |
| WEIGHT_LIE | 1 |
| WEIGHT_NORM_LIE | 0 |
| DECAY_LIE | 1 |
| DECAY_INIT | 1 |
| DECAY_UNSAFE | 1 |

(b) Hyper-parameters for Spacecraft Randezvous

| Hyper-Parameters | Value |
|---|---|
| LEARNING_RATE | 0.01 |
| BATCH_SIZE | 512 |
| CONTROLLER_PERIOD | 0.01 |
| SIMULATION_DT | 0.01 |
| CBF_HIDDEN_LAYERS | 1 |
| CBF_HIDDEN_SIZE | 16 |
| CBF_LAMBDA | 0.1 |
| CBF_RELAXATION_PEN | 1e4 |
| SCALE_PARAMETER | 10.0 |
| PRIMAL_LEARNING_RATE | 1e-3 |
| LEARN_SHAPE_EPOCHS | 100 |

### A.14 Experiment Details and Results

We use translators and verifiers proposed in FOSSIL [3] for SMT-based verification with solver dReal and Z3 as baselines. Our proposed enumerating algorithm utilize auto-LiRPA [4] with default settings and linear program with HiGHS solvers provided by SciPy [5]. Detailed settings can be found in our attached code.

We further visualize the trend of the number of activation sets and run-time with respect to the total number of neurons with ReLU activation function in Fig. 3. We can find that the logarithm of the activation sets size grows with the size of the neural network. The dimensionality of the state is the dominant factor in determining the run-time. The logarithm of the run-time is determined by both the state dimension and the number of ReLU hidden layers. The potential result can be cause by loose activation set estimation. Methods deriving tighter bounds than IBP may mitigate the influence of the ReLU hidden layers.

### A.15 Comparison of NCBF and SOS-based Synthesis

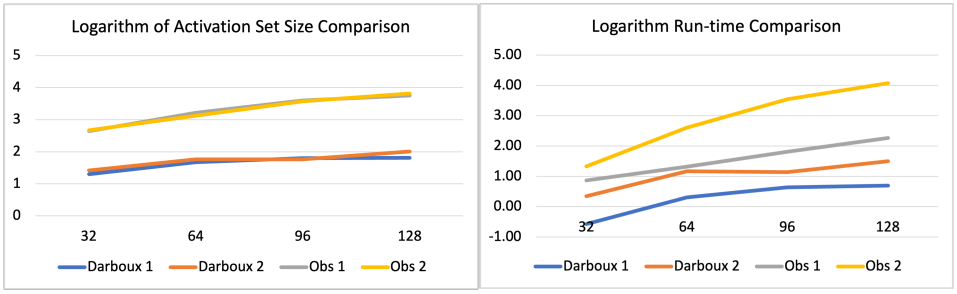
We compare NCBF with traditional SOS synthesized polynomial CBF for the obstacle avoidance case study in two aspects, namely, training time $T_t$ and volume $V$ of the guaranteed safe region. In order to synthesize the polynomial CBF, we adopt the procedure introduced in [44]. This procedure first constructs a nominal controller $\mu(x)$, and then uses SOS programming to construct a barrier certificate for the system $\dot{x}(t) = f(x) + g(x)\mu(x)$. We choose $\mu(x) = -x_3$ as the nominal controller and synthesize CBFs of degree 2, 4, 6, 8, and 10 using the Matlab SOSTOOLS toolbox. We compared the result with an NCBF with one hidden layer of 32 neurons trained using the method proposed in [23] with the same nominal controller. The experiment results are shown below. The time of SOS CBF synthesis grows with the degree of the barrier function. Degree 10 CBF takes twice the time compared to NCBF. On the other hand, NCBF outperforms all SOS synthesized CBFs by having the largest safe region volume.

[2] https://github.com/MIT-REALM/neural_clbf

[3] https://github.com/oxford-oxcav/fossil

[4] https://github.com/Verified-Intelligence/auto_LiRPA

[5] https://docs.scipy.org/doc/scipy-1.10.1/reference/optimize.linprog-highs.html




(a) (b)

Figure 3: Comparison of the number of activation sets and run-time with respect to number of neurons in total. (a) shows logarithm of activation set size. (b) shows logarithm of run-time. We denote NCBFs for Darboux with 1 and 2 hidden layers as Darboux 1 and Darboux 2, respectively. We denote NCBFs for obstacle avoidance with 1 and 2 hidden layers as obs 1 and obs 2, respectively.

Table 4: Comparison of the training time $T_t$ and safe region volume $V$ of a NCBF and SOS synthesized CBFs for Obstacle Avoidance

| Types | $T_t$ $(s)$ | $V$ $(m^2 \times deg)$ |
|---|---|---|
| NCBF 3-32-$\sigma$-1 | 262.89s | 37.76 |
| SOS Degree 2 | 7.36s | 16.14 |
| SOS Degree 4 | 6.65s | 13.44 |
| SOS Degree 6 | 19.88s | 31.36 |
| SOS Degree 8 | 125.10s | 25.93 |
| SOS Degree 10 | 551.31s | 19.99 |

### A.16 Comparison between Activation Functions

We considered three case studies, namely, Darboux, obstacle avoidance, and spacecraft rendezvous. For each case study, we trained and verified three NCBFs with the same architecture (2 hidden layers of 32 neurons each) but different activation functions, namely, ReLU, sigmoid, and tanh. We found

Table 5: Comparison of training time $T_t$, safety region volume $V$ and verification time $T_v$ of ReLU, Sigmoid and Tanh NCBF for Darboux, Obstacle Avoidance and Spacecraft Rendezvous. ReLU NNs are verified by proposed method while others are verified by dReal and Z3. We write UTD when the method cannot be not directly used for verification

| Case | Darboux | | | Obstacle Avoidance | | | Spacecraft Rendezvous | |
|---|---|---|---|---|---|---|---|---|
| | ReLU | Sigmoid | Tanh | ReLU | Sigmoid | Tanh | ReLU | Tanh |
| $T_t$ | 28.53s | 51.14s | 69.49s | 71.44s | 78.66s | 76.49s | 879.388s | 953.469s |
| $V$ | 2.27 | 1.62 | 2.8 | 4.99 | 2.17 | 3.50 | 0.20 | 0.22 |
| $T_v$ | 14.64 | >3hrs | >3hrs | 273.37s | >3hrs | >3hrs | 13906.19s | UTD |

that, for the Darboux and obstacle avoidance case studies, the ReLU NCBF completed training faster than both sigmoid and tanh NCBFs. The volume of the safe region was comparable for all three activation functions, with the tanh outperforming the ReLU NCBF in Darboux and the ReLU NCBF providing the largest volume for obstacle avoidance. The most significant difference between the three activation functions was at the verification stage. Our proposed method for verifying ReLU NCBFs terminated within 15 and 274 seconds in the Darboux and obstacle avoidance, respectively, while SMT-based methods did not terminate within three hours for both test cases. In the spacecraft rendezvous example, the ReLU NCBF completed training before the tanh NCBF. Moreover, while our approach verified the correctness of the ReLU NCBF within 4 hours, the tanh NCBF exhibited a safety violation.

### A.17 Example Details

Consider the setting of the example in Section 3.2. Let $b_c$ denote the NCBF defined in the example, which fails our defined safety conditions. For comparison, we trained an NCBF $b_\theta$ and verified it using our proposed approach. We then constructed a nominal controller $\mu_{nom}$ as a Linear Quadratic Regulator (LQR) controller that drives the system from initial point $(0, 0.1)$ to the origin. We compared the trajectories arising from the optimization-based controller defined by Eq. (10) using the $b_\theta$ and $b_c$. For the unsafe NCBF $b_c$, the optimization-based controller is unable to satisfy the safety constraints at the boundary point $(0, 1)$, resulting in a safety violation as described in the manuscript. On the other hand, while the NCBF $b_\theta$ contained multiple non-differentiable points, it is possible to choose $u$ to ensure safety at these points. For example, the point $(-0.19, 2.91)$ is a non-differentiable point on the boundary $b_\theta = 0$. There are four activation sets intersecting at this point, with corresponding values of $\frac{\partial b_c}{\partial x} g(x)$ given by $\{-0.0455, -0.053, -0.025, -0.033\}$. Since any control input $u$ with negative sign and sufficiently large magnitude will satisfy $\frac{\partial b_c}{\partial x}(f(x)+g(x)u) \geq 0$ for all of these values, this non-differentiable point does not compromise safety of the system, and the trajectory of the system constrained by $b_\theta$ remains in the safe region for all time.

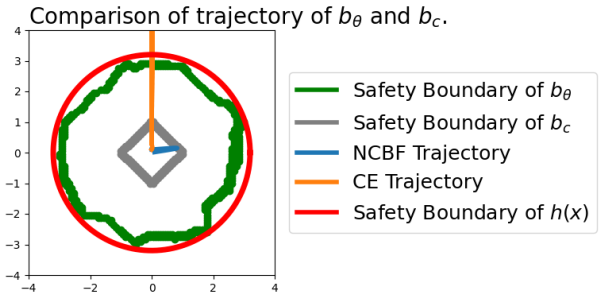



Figure 4: Comparison of optimization-based controller using trained NCBF $b_\theta$ and unsafe NCBF $b_c$.