Algorithm 1 BaB-prob

Input: $f(\mathbf{x})$, $\mathbf{P}$, $\mathcal{D}$, η

1: $\mathcal{B} \leftarrow [\,]$
2: $\underline{f}(\mathbf{x}), \bar{f}(\mathbf{x}), \underline{\mathbf{y}}_o^{[N-1]}(\mathbf{x}), \bar{\mathbf{y}}_o^{[N-1]}(\mathbf{x}) \leftarrow \mathit{ComputeLinearBounds}(f, \mathcal{D}, \varnothing)$
3: $p_{\ell,o}, p_{u,o} \leftarrow \mathit{BoundBranchProbability}(\mathbf{P}, \underline{f}(\mathbf{x}), \bar{f}(\mathbf{x}), \underline{\mathbf{y}}_o^{[N-1]}(\mathbf{x}), \bar{\mathbf{y}}_o^{[N-1]}(\mathbf{x}), \varnothing)$
4: $B_o \leftarrow \langle p_{\ell,o}, p_{u,o}, \varnothing \rangle$
5: **if** $p_{\ell,o} < p_{u,o}$ **then**
6: $\quad \mathit{MarkPreactivationToSplitOn}(B_o)$
7: $\mathcal{B}.\mathit{insert}(B_o)$
8: **while True do**
9: $\quad P_\ell, P_u \leftarrow \mathit{BoundGlobalProbability}(\mathcal{B})$
10: $\quad$ **if** $P_\ell \geq \eta$ **then return** TRUE
11: $\quad$ **if** $P_u < \eta$ **then return** FALSE
12: $\quad B = \langle p_\ell, p_u, \mathcal{C} \rangle \leftarrow \mathcal{B}.\mathit{pop}()$
13: $\quad \left\{ y_j^{(k)} \geq 0, y_j^{(k)} < 0 \right\} \leftarrow \mathit{GenerateNewConstraints}(B)$
14: $\quad$ **for** $c_i \in \left\{ y_j^{(k)} \geq 0, y_j^{(k)} < 0 \right\}$ **do**
15: $\quad\quad \mathcal{C}_i \leftarrow \mathcal{C} \cup \{c_i\}$
16: $\quad\quad \underline{f}_i(\mathbf{x}), \bar{f}_i(\mathbf{x}), \underline{\mathbf{y}}_i^{[N-1]}(\mathbf{x}), \bar{\mathbf{y}}_i^{[N-1]}(\mathbf{x}) \leftarrow \mathit{ComputeLinearBounds}(f, \mathcal{D}, \mathcal{C}_i)$
17: $\quad\quad p_{\ell,i}, p_{u,i} \leftarrow \mathit{BoundBranchProbability}(\mathbf{P}, \underline{f}_i(\mathbf{x}), \bar{f}_i(\mathbf{x}), \underline{\mathbf{y}}_i^{[N-1]}(\mathbf{x}), \bar{\mathbf{y}}_i^{[N-1]}(\mathbf{x}), \mathcal{C}_i)$
18: $\quad\quad B_i \leftarrow \langle p_{l,i}, p_{u,i}, \mathcal{C}_i \rangle$
19: $\quad\quad$ **if** $p_{l,i} < p_{u,i}$ **then**
20: $\quad\quad\quad \mathit{MarkPreactivationToSplitOn}(B_i)$
21: $\quad\quad \mathcal{B}.\mathit{insert}(B_i)$

A Detailed Framework of BaB-prob

The pseudocode of BaB-prob is shown in Algorithm 1. During initialization, $\mathcal{B}$ is created to maintain all candidate branches (Line 1). BaB-prob applies linear bound propagation over $\mathcal{D}$ **under no constraint** to compute linear bounds for f and $\mathbf{y}^{[N-1]}$, yielding $\underline{f}_o(\mathbf{x})$, $\bar{f}_o(\mathbf{x})$ and $\underline{\mathbf{y}}_o^{[N-1]}(\mathbf{x})$, $\bar{\mathbf{y}}_o^{[N-1]}(\mathbf{x})$, such that

$$\underline{f}_o(\mathbf{x}) \leq f(\mathbf{x}) \leq \bar{f}_o(\mathbf{x}), \qquad \forall \mathbf{x} \in \mathcal{D}, \tag{10a}$$

$$\underline{\mathbf{y}}_i^{(k)}(\mathbf{x}) \leq \mathbf{y}^{(k)}(\mathbf{x}) \leq \bar{\mathbf{y}}_o^{(k)}(\mathbf{x}), \qquad \forall \mathbf{x} \in \mathcal{D},\ k \in [N-1], \tag{10b}$$

(Line 2). BaB-prob then applies Equation (4) to compute the probability bounds for $\mathbb{P}\left(f(\mathbf{X}) > 0\right)$, obtaining the lower bound $p_{\ell,o}$ and the upper bound $p_{u,o}$, and creates the root branch $B_o = \langle p_{\ell,o}, p_{u,o}, \varnothing \rangle$ (Line 3-4). Before inserting B_o into $\mathcal{B}$, the preactivation in B_o to be split on is first identified (though not split immediately) (Line 5-6). This strategy reduces memory usage for BaB+BaBSR-prob, since the method relies on the linear bounds to choose the preactivation to split on. By marking the preactivation at this stage, we can discard the linear bound information before inserting B_o into $\mathcal{B}$. Besides, BaB-prob only performs the identification if $(p_u - p_\ell)$ is positive. At the end of initialization, B_o is inserted into $\mathcal{B}$ (Line 7).

At each iteration, BaB-prob first computes the global probability bounds for $\mathbb{P}\left(f(\mathbf{X}) > 0\right)$:

$$P_\ell = \sum_{\langle p_\ell, p_u, \mathcal{C} \rangle \in \mathcal{B}} p_\ell, \quad P_u = \sum_{\langle p_\ell, p_u, \mathcal{C} \rangle \in \mathcal{B}} p_u, \tag{11}$$

(Line 9). If the current global probability bounds are already enough to make a certification, BaB-prob terminates and make the corresponding certification (Line 10-11). Otherwise, BaB-prob pops the branch $B = \langle p_\ell, p_u, \mathcal{C} \rangle$ with the largest $(p_u - p_\ell)$ from $\mathcal{B}$ (Line 12) and generates the two new constraints on the identified preactivation (Line 13). For each new constraint c_i, BaB-prob generates a new set of constraints $\mathcal{C}_i = \mathcal{C} \cup \{c_i\}$ (Line 14-15). BaB-prob then applies linear bounds propagation over $\mathcal{D}$ **under the constraints in** $\mathcal{C}_i$ to compute linear bounds for f and $\mathbf{y}^{[N-1]}$, yielding

$\underline{f}_i(\mathbf{x})$, $\bar{f}_i(\mathbf{x})$ and $\underline{\mathbf{y}}_i^{[N-1]}(\mathbf{x})$, $\bar{\mathbf{y}}_i^{[N-1]}$, such that

$$\underline{f}_i(\mathbf{x}) \leq f(\mathbf{x}) \leq \bar{f}_i(\mathbf{x}), \qquad \forall \mathbf{x} \in \mathcal{D},\ \mathcal{C}_i \text{ satisfied}, \tag{12a}$$

$$\underline{\mathbf{y}}_i^{(k)}(\mathbf{x}) \leq \mathbf{y}^{(k)}(\mathbf{x}) \leq \bar{\mathbf{y}}_i^{(k)}(\mathbf{x}), \qquad \forall \mathbf{x} \in \mathcal{D},\ \mathcal{C}_i^{[k-1]} \text{ satisfied},\ k \in [N-1], \tag{12b}$$

(Line 16). Then, it uses Equation (4) to compute the probability bounds for $\mathbb{P}\left(f(\mathbf{X}) > 0, \mathcal{C}_i\right)$, obtaining $p_{\ell,i}$ and $p_{u,i}$. Same as the initialization, BaB-prob identifies the preactivation to split on for $B_i = \langle p_{\ell,i}, p_{u,i}, \mathcal{C}_i \rangle$ in advance (Line 19-20) and then inserts B_i into $\mathcal{B}$.

B Proof for Theoretical Results

Lemma 1. *Let $\mathcal{C} = \left\{ y_{j_\ell}^{k_\ell} \geq 0, \ell \in [s],\ y_{j_\ell}^{k_\ell}, \ell \in [s+1, t] \right\}$. Assume $\mathcal{C}$ can be decomposed by*

$$\mathcal{C}^{(k)} = \left\{ y_{j_{k,\ell}}^{(k)} \geq 0, \ell \in [s_k],\ y_{j_{k,\ell}}^{(k)} < 0, \ell \in [s_k+1, t_k] \right\}, \quad k \in [N-1]. \tag{13}$$

Let and $\underline{f}(\mathbf{x})$, $\bar{f}(\mathbf{x})$, $\underline{\mathbf{y}}^{[N-1]}(\mathbf{x})$, $\bar{\mathbf{y}}^{[N-1]}(\mathbf{x})$ be the linear bounds for f and $\mathbf{y}^{[N-1]}$ obtained by linear bound propagation under the constraints of $\mathcal{C}$. For $k \in [N-1]$, let

$$\begin{aligned}
\mathcal{C}_x^{(k)} &= \left\{ \mathbf{x} \in \mathcal{D} : y_{j_{k,\ell}}^{(k)}(\mathbf{x}) \geq 0, \ell \in [s_k],\ y_{j_{k,\ell}}^{(k)}(\mathbf{x}) < 0, \ell \in [s_k+1, t_k] \right\}, \\
\underline{\mathcal{C}}_x^{(k)} &= \left\{ \mathbf{x} \in \mathcal{D} : \underline{y}_{j_{k,\ell}}^{(k)}(\mathbf{x}) \geq 0, \ell \in [s_k],\ \bar{y}_{j_{k,\ell}}^{(k)}(\mathbf{x}) < 0, \ell \in [s_k+1, t_k] \right\}, \\
\bar{\mathcal{C}}_x^{(k)} &= \left\{ \mathbf{x} \in \mathcal{D} : \bar{y}_{j_{k,\ell}}^{(k)}(\mathbf{x}) \geq 0, \ell \in [s_k],\ \underline{y}_{j_{k,\ell}}^{(k)}(\mathbf{x}) < 0, \ell \in [s_k+1, t_k] \right\}.
\end{aligned} \tag{14}$$

Then, for all $k \in [N-1]$

$$\bigcap_{r=1}^{k} \underline{\mathcal{C}}_x^{(r)} \subseteq \bigcap_{r=1}^{k} \mathcal{C}_x^{(r)} \subseteq \bigcap_{r=1}^{k} \bar{\mathcal{C}}_x^{(r)}, \tag{15}$$

and,

$$\begin{aligned}
&\Big(\bigcap_{r=1}^{N-1} \underline{\mathcal{C}}_x^{(r)} \Big) \cap \{ \mathbf{x} \in \mathcal{D} : \underline{f}(\mathbf{x}) > 0 \} \\
\subseteq &\Big(\bigcap_{r=1}^{N-1} \mathcal{C}_x^{(r)} \Big) \cap \{ \mathbf{x} \in \mathcal{D} : f(\mathbf{x}) > 0 \} \\
\subseteq &\Big(\bigcap_{r=1}^{N-1} \bar{\mathcal{C}}_x^{(r)} \Big) \cap \{ \mathbf{x} \in \mathcal{D} : \bar{f}(\mathbf{x}) > 0 \}.
\end{aligned} \tag{16}$$

Proof. We first prove Equation (15) by induction.

By Equation (3b),

$$\underline{\mathbf{y}}^{(1)}(\mathbf{x}) \leq \mathbf{y}^{(1)}(\mathbf{x}) \leq \bar{\mathbf{y}}^{(1)}(\mathbf{x}), \quad \forall \mathbf{x} \in \mathcal{D}. \tag{17}$$

Therefore, $\underline{\mathcal{C}}_x^{(1)} \subseteq \mathcal{C}_x^{(1)} \subseteq \bar{\mathcal{C}}_x^{(1)}$, implying that Equation (15) holds for $k = 1$.

Assume Equation (15) holds for $k-1$, we will prove Equation (15) for k.

$\forall \mathbf{x} \in \bigcap_{r=1}^{k} \underline{\mathcal{C}}_x^{(r)} \subseteq \bigcap_{r=1}^{k-1} \underline{\mathcal{C}}_x^{(r)}$, Equation (15) holding for $k-1$ implies that $\mathbf{x} \in \bigcap_{r=1}^{k-1} \mathcal{C}_x^{(r)}$, i.e., $\mathcal{C}^{[k-1]}$ are satisfied. By Equation (3b),

$$\underline{\mathbf{y}}^{(k)}(\mathbf{x}) \leq \mathbf{y}^{(k)}(\mathbf{x}) \leq \bar{\mathbf{y}}^{(k)}(\mathbf{x}). \tag{18}$$

By Equation (18), and since $\mathbf{x} \in \underline{\mathcal{C}}_x^{(k)}$, it holds that $\mathbf{x} \in \mathcal{C}_x^{(k)}$. So we have $\mathbf{x} \in \bigcap_{r=1}^{k} \mathcal{C}_x^{(r)}$, implying that $\bigcap_{r=1}^{k} \underline{\mathcal{C}}_x^{(r)} \subseteq \bigcap_{r=1}^{k} \mathcal{C}_x^{(r)}$. For the second inequality, $\forall \mathbf{x} \in \bigcap_{r=1}^{k} \mathcal{C}_x^{(r)}$, Equation (18) also holds. Then, also, $\mathbf{x} \in \bar{\mathcal{C}}_x^{(k)}$. So $\mathbf{x} \in \bigcap_{r=1}^{k} \bar{\mathcal{C}}_x^{(r)}$, implying that $\bigcap_{r=1}^{k} \mathcal{C}_x^{(r)} \subseteq \bigcap_{r=1}^{k} \bar{\mathcal{C}}_x^{(r)}$. Thus, Equation (15) holds for k. Therefore, Equation (15) holds for all $k \in [N-1]$.

Using Equation (15) for $k = N-1$, we can prove Equation (16) similar to how we prove Equation (15) from $k-1$ to k. $\square$

Proof of Proposition 1. By Lemma 1 (Equation (16)),

$$\begin{aligned}\mathbb{P}\left(f(\mathbf{X})>0,\,\mathcal{C}\right) &\geq \mathbb{P}\left(\underline{f}(\mathbf{X})>0, \underline{y}_{j_\ell}^{k_\ell}(\mathbf{X})\geq 0, \ell\in[s],\ \bar{y}_{j_\ell}^{k_\ell}(\mathbf{X})<0, \ell\in[s+1,t]\right)\\ &= \mathbb{P}\left(\underline{\mathbf{P}}\mathbf{X}+\underline{\mathbf{q}}\right). \end{aligned} \tag{19}$$

Similarly, $\mathbb{P}\left(f(\mathbf{X})>0,\,\mathcal{C}\right)\leq \mathbb{P}\left(\bar{\mathbf{P}}\mathbf{X}+\bar{\mathbf{q}}\right)$. □

Proof of Proposition 2. We assume the constraint on $y_{j^\star}^{(k^\star)}$ in $\mathcal{C}$ is $y_{j^\star}^{(k^\star)}\geq 0$, the other case can be proved similarly. Assume that for $k\in[N-1]$, $k\neq k^\star$,

$$\mathcal{C}^{(k)}=\left\{y_{j_{k,\ell}}^{(k)}\geq 0,\ \ell\in[s_k],\ y_{j_{k,\ell}}^{(k)}<0,\ \ell\in[s_k+1,t_k]\right\}, \tag{20}$$

and

$$\mathcal{C}^{(k^\star)}=\left\{y_{j_{k^\star,\ell}}^{(k^\star)}\geq 0,\ \ell\in[s_{k^\star}],\ y_{j_{k^\star,\ell}}^{(k^\star)}<0,\ \ell\in[s_{k'}+1,t_{k^\star}],\ y_{j^\star}^{(k^\star)}\geq 0\right\}. \tag{21}$$

By Proposition 1,

$$p_\ell=\mathbb{P}\begin{pmatrix}\underline{f}(\mathbf{X})>0, & \\ \underline{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})\geq 0,\ \ell\in[s_k], & k\in[N-1],\\ \bar{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[N-1]\end{pmatrix}. \tag{22}$$

Applying Lemma 1 (Equation (15)) with $k=N-1$,

$$\bigcap_{r=1}^{N-1}\underline{\mathcal{C}}_x^{(r)}\subseteq\bigcap_{r=1}^{N-1}\mathcal{C}_x^{(r)}, \tag{23}$$

therefore,

$$\begin{aligned}&\mathbb{P}\begin{pmatrix}\underline{f}(\mathbf{X})>0, & \\ \underline{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})\geq 0,\ \ell\in[s_k], & k\in[N-1],\\ \bar{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[N-1]\end{pmatrix}\\ =&\mathbb{P}\begin{pmatrix}\underline{f}(\mathbf{X})>0, & \\ \underline{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})\geq 0,\ \ell\in[s_k], & k\in[N-1],\\ \bar{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[N-1],\\ \mathcal{C}^{[N-1]} & \end{pmatrix}\\ =&\mathbb{P}\left(\mathcal{C}^{[N-1]}\right)\mathbb{P}\left(\begin{matrix}\underline{f}(\mathbf{X})>0, & \\ \underline{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})\geq 0,\ \ell\in[s_k], & k\in[N-1],\\ \bar{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[N-1]\end{matrix}\ \middle|\ \mathcal{C}^{[N-1]}\right).\end{aligned} \tag{24}$$

By Equation (3a), we have

$$\begin{aligned}&\mathbb{P}\left(\begin{matrix}\underline{f}(\mathbf{X})>0, & \\ \underline{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})\geq 0,\ \ell\in[s_k], & k\in[N-1],\\ \bar{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[N-1]\end{matrix}\ \middle|\ \mathcal{C}^{[N-1]}\right)\\ \leq&\mathbb{P}\left(\begin{matrix}\bar{f}(\mathbf{X})>0, & \\ \underline{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})\geq 0,\ \ell\in[s_k], & k\in[N-1],\\ \bar{y}_{j_{k,\ell}}^{(k)}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[N-1]\end{matrix}\ \middle|\ \mathcal{C}^{[N-1]}\right).\end{aligned} \tag{25}$$

From Equation (22) (24) (25),

$$\begin{aligned}
p_\ell &\le \mathbb{P}\left(\mathcal{C}^{[N-1]}\right)\mathbb{P}\left(\begin{matrix}\bar{f}(\mathbf{X})>0, & & \\ \underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})\ge 0,\ \ell\in[s_k], & k\in[N-1], \\ \bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[N-1]\end{matrix}\ \middle|\ \mathcal{C}^{[N-1]}\right) \\
&= \mathbb{P}\begin{pmatrix}\bar{f}(\mathbf{X})>0, & \\ \underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})\ge 0,\ \ell\in[s_k], & k\in[N-1], \\ \bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[N-1], \\ \mathcal{C}^{[N-1]} & \end{pmatrix} \\
&\le \mathbb{P}\begin{pmatrix}\bar{f}(\mathbf{X})>0, & \\ \underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})\ge 0,\ \ell\in[s_k], & k\in[N-1], \\ \bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[N-1], \\ \mathcal{C}^{[N-2]} & \end{pmatrix}.
\end{aligned} \tag{26}$$

Applying the argument of Equation (24)-(26), except that we apply Equation (3b) instead of Equation (3a) within Equation (25), iteratively to the constrained preactivations in layers $N-1,\ldots,k^*+1$, we obtain

$$\begin{aligned}
p_\ell &\le \mathbb{P}\begin{pmatrix}\bar{f}(\mathbf{X})>0, & \\ \bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})\ge 0,\ \ell\in[s_k], & k\in[k^\star+1,N-1] \\ \underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[k^\star+1,N-1] \\ \underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})\ge 0,\ \ell\in[s_k], & k\in[k^\star] \\ \bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[k^\star] \\ \underline{y}^{(k^\star)}_{j^\star}(\mathbf{X})\ge 0, & \\ \mathcal{C}^{[k^\star-1]} & \end{pmatrix} \\
&= \mathbb{P}\left(\mathcal{C}^{[k^\star-1]}\right)\mathbb{P}\left(\begin{matrix}\bar{f}(\mathbf{X})>0, & \\ \bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})\ge 0,\ \ell\in[s_k], & k\in[k^\star+1,N-1] \\ \underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[k^\star+1,N-1] \\ \underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})\ge 0,\ \ell\in[s_k], & k\in[k^\star] \\ \bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X})<0,\ \ell\in[s_k+1,t_k], & k\in[k^\star] \\ \underline{y}^{(k^\star)}_{j^\star}(\mathbf{X})\ge 0 & \end{matrix}\ \middle|\ \mathcal{C}^{[k^\star-1]}\right).
\end{aligned} \tag{27}$$

Applying Equation (3b) with $k=k^*$,

$$\underline{y}^{(k^\star)}_{j_{k^\star,\ell}}(\mathbf{x})\le y^{(k^\star)}_{j_{k^\star,\ell}}(\mathbf{x})\le \bar{y}^{(k^\star)}_{j_{k^\star,\ell}}(\mathbf{x}),\qquad \forall\mathbf{x}\in\mathcal{D},\ \mathcal{C}^{[k^\star-1]}\text{ satisfied},\ \ell\in[t_{k^\star}], \tag{28a}$$

$$\underline{y}^{(k^\star)}_{j^\star}(\mathbf{x})\le y^{(k^\star)}_{j^\star}(\mathbf{x})\le \bar{y}^{(k^\star)}_{j^\star}(\mathbf{x}),\qquad \forall\mathbf{x}\in\mathcal{D},\ \mathcal{C}^{[k^\star-1]}\text{ satisfied}. \tag{28b}$$

Thus,

$$
\begin{aligned}
&\mathbb{P}\left(\begin{array}{ll}
\bar{f}(\mathbf{X}) > 0, & \\
\bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\, \ell \in [s_k], & k \in [k^\star+1, N-1] \\
\underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\, \ell \in [s_k+1, t_k], & k \in [k^\star+1, N-1] \\
\underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\, \ell \in [s_k], & k \in [k^\star] \\
\bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\, \ell \in [s_k+1, t_k], & k \in [k^\star] \\
\underline{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0 &
\end{array}\;\middle|\; \mathcal{C}^{[k^\star-1]}\right) \\
\leq\, &\mathbb{P}\left(\begin{array}{ll}
\bar{f}(\mathbf{X}) > 0, & \\
\bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\, \ell \in [s_k], & k \in [k^\star, N-1] \\
\underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\, \ell \in [s_k+1, t_k], & k \in [k^\star, N-1] \\
\underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\, \ell \in [s_k], & k \in [k^\star-1] \\
\bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\, \ell \in [s_k+1, t_k], & k \in [k^\star-1] \\
\underline{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0 &
\end{array}\;\middle|\; \mathcal{C}^{[k^\star-1]}\right) \\
=\, &\mathbb{P}\left(\begin{array}{ll}
\bar{f}(\mathbf{X}) > 0, & \\
\bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\, \ell \in [s_k], & k \in [k^\star, N-1] \\
\underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\, \ell \in [s_k+1, t_k], & k \in [k^\star, N-1] \\
\underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\, \ell \in [s_k], & k \in [k^\star-1] \\
\bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\, \ell \in [s_k+1, t_k], & k \in [k^\star-1] \\
\underline{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0,\, \bar{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0 &
\end{array}\;\middle|\; \mathcal{C}^{[k^\star-1]}\right). \qquad (29)
\end{aligned}
$$

From Equation (27) (29),

$$
\begin{aligned}
p_\ell &\leq \mathbb{P}\left(\mathcal{C}^{[k^\star-1]}\right)\mathbb{P}\left(\begin{array}{ll}
\bar{f}(\mathbf{X}) > 0, & \\
\bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\, \ell \in [s_k], & k \in [k^\star, N-1] \\
\underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\, \ell \in [s_k+1, t_k], & k \in [k^\star, N-1] \\
\underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\, \ell \in [s_k], & k \in [k^\star-1] \\
\bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\, \ell \in [s_k+1, t_k], & k \in [k^\star-1] \\
\underline{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0,\, \bar{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0 &
\end{array}\;\middle|\; \mathcal{C}^{[k^\star-1]}\right) \\
&= \mathbb{P}\left(\begin{array}{ll}
\bar{f}(\mathbf{X}) > 0, & \\
\bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\, \ell \in [s_k], & k \in [k^\star, N-1] \\
\underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\, \ell \in [s_k+1, t_k], & k \in [k^\star, N-1] \\
\underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\, \ell \in [s_k], & k \in [k^\star-1] \\
\bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\, \ell \in [s_k+1, t_k], & k \in [k^\star-1] \\
\underline{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0,\, \bar{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0, & \\
\mathcal{C}^{[k^\star-1]} &
\end{array}\right). \qquad (30)
\end{aligned}
$$

Then, again applying the argument of Equation (24)-(26), except that Equation (3b) is used instead of Equation (3a) within Equation (25), iteratively to the constrained preactivations in layers $k^\star -$

$1, \ldots, 1$, we obtain

$$p_\ell \leq \mathbb{P}\begin{pmatrix} \bar{f}(\mathbf{X}) > 0, & \\ \bar{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) \geq 0,\ \ell \in [s_k], & k \in [N-1] \\ \underline{y}^{(k)}_{j_{k,\ell}}(\mathbf{X}) < 0,\ \ell \in [s_k+1, t_k], & k \in [N-1] \\ \underline{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0,\ \bar{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0 & \end{pmatrix}$$

$$= \mathbb{P}\begin{pmatrix} \bar{f}(\mathbf{X}) > 0,\ \underline{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0,\ \bar{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0, \\ \bar{y}^{(k_\ell)}_{j_\ell}(\mathbf{X}) \geq 0, \ell \in [s],\ \underline{y}^{(k_\ell)}_{j_\ell}(\mathbf{X}) < 0, \ell \in [s+1, t] \end{pmatrix}. \tag{31}$$

On the other hand, from Proposition 1,

$$p_u = \mathbb{P}\begin{pmatrix} \bar{f}(\mathbf{X}) > 0,\ \bar{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0, \\ \bar{y}^{(k_\ell)}_{j_\ell}(\mathbf{X}) \geq 0, \ell \in [s],\ \underline{y}^{(k_\ell)}_{j_\ell}(\mathbf{X}) < 0, \ell \in [s+1, t] \end{pmatrix}. \tag{32}$$

Then, from Equation (31) (32),

$$p_u - p_\ell \geq \mathbb{P}\begin{pmatrix} \bar{f}(\mathbf{X}) > 0,\ \bar{y}^{(k^\star)}_{j^\star}(\mathbf{X}) \geq 0,\ \underline{y}^{(k^\star)}_{j^\star}(\mathbf{X}) < 0, \\ \bar{y}^{(k_\ell)}_{j_\ell}(\mathbf{X}) \geq 0, \ell \in [s],\ \underline{y}^{(k_\ell)}_{j_\ell}(\mathbf{X}) < 0, \ell \in [s+1, t] \end{pmatrix} \tag{33}$$

□

Proof of Proposition 3. Notice that after BaB-prob splits on a preactivation in B and generates B_1 and B_2, it holds that

$$\mathbb{P}(B) = \mathbb{P}(B_1) + \mathbb{P}(B_2). \tag{34}$$

Thus, at the beginning each iteration,

$$\mathbb{P}(f(\mathbf{X}) > 0) = \sum_{B \in \mathcal{B}} \mathbb{P}(B). \tag{35}$$

By Proposition 1, for all $B = \langle p_\ell, p_u, \mathcal{C} \rangle \in \mathcal{B}$, p_ℓ and p_u are lower bounds on $\mathbb{P}(B)$. Therefore, from Equation (35), P_ℓ and P_u are lower and upper bounds on $\mathbb{P}(f(\mathbf{X}) > 0)$. □

Proof of Proposition 4. Let $\underline{f}(\mathbf{x})$, $\bar{f}(\mathbf{x})$ and $\underline{\mathbf{y}}^{[N-1]}(\mathbf{x})$, $\bar{\mathbf{y}}^{[N-1]}(\mathbf{x})$ be the linear lower and upper bounds for f and $\mathbf{y}^{[N-1]}$ computed by linear bound propagation under the constraints in $\mathcal{C}$. Since there is no unstable preactivation in B, ino relaxation is performed during the linear bound propagation, the inequalities in Equation (3) become equalities. Thus, all the inclusions '$\subseteq$' in the proof of Lemma 1 become equalities '$=$', and subsequently, the inequality in the proof of Proposition 1 becomes equality. Therefore, $\mathbb{P}(B) = p_\ell = p_u$. □

Proof of Proposition 5. We prove Proposition 5 by contradiction. Suppose that BaB-prob does not terminate in finite time. Since there are only finitely many preactivations, there must exist a point at which every branch in $\mathcal{B}$ contains no unstable preactivation. At that point, by Proposition 4, all branches have exactly tight probability bounds. Consequently, $P_\ell = P_u$, which implies that BaB-prob has already terminated — a contradiction. □

Proof of Corollary 1. It follows directly from Proposition 3 and Proposition 5. □

C Experiments Details

C.1 Toy models (soundness and completeness check)

Setup. The MLP model has input dimension 5, two hidden FC layers (10 units each), ReLU after every hidden layer, and a scalar output. The CNN model takes a $1 \times 4 \times 4$ input, has two

	MLP model						CNN model					
	T/T	F/T	N/F	T/F	F/F	N/F	T/T	F/T	N/F	T/F	F/F	N/F
PROVEN	4/24	0/24	20/24	0/6	0/6	6/6	5/25	0/25	20/25	0/5	0/5	5/5
PV	24/24	0/24	0/24	0/6	6/6	0/6	25/25	0/25	0/25	0/5	5/5	0/5
SDP	0/24	24/24	0/24	0/6	6/6	0/6	0/25	25/25	0/25	0/5	5/5	0/5
BaB-prob-ordered	24/24	0/24	0/24	0/6	6/6	0/6	25/25	0/25	0/25	0/5	5/5	0/5
BaB+BaBSR-prob	24/24	0/24	0/24	0/6	6/6	0/6	25/25	0/25	0/25	0/5	5/5	0/5

Table 3: Results for toy models. Each cell indicates the number of instances where the algorithm's declaration ("T" for True, "F" for False, "N" for no declaration) aligns with the ground truth ("T" or "F"). For example, "T/T" means the ground truth is True and the algorithm declares it as True; "F/T" means the ground truth is True but the algorithm declares it as False; "N/T" means the ground truth is True but the algorithm fails to provide a declaration. Similar interpretations apply to the other entries.

Conv2d layers (3 channels, kernel 3, stride 2), ReLU after convolutions, then flatten + scalar FC. All weights/biases are i.i.d. $\mathcal{N}(0, 0.25)$. For each architecture we generate 30 instances, i.e., 30 different networks, such that $\mathbf{x}_0 = 0$ and $f(\mathbf{x}_0) > 0$. Noise is $\mathcal{N}(0, 0.1\mathbf{I})$. Because PROVEN, PV and BaB-prob assume bounded inputs, we truncate the Gaussian to its 99.7 %-confidence ball. We use batch size of 16384 for both versions of BaB-prob and PV, and split depth (maybe splitting multiple preactivations at one time) of 1 for this experiment. To verify the soundness and completeness, we do not set a time limit in this experiment. We use a toy MLP model and a toy CNN model to test the soundness and completeness of different solvers. We use `SaVer-Toolbox`[2] (Sivaramakrishnan et al. (2024)) to obtain an empirical reference $\hat{P}$ for $\mathbb{P}\left(f(\mathbf{X}) > 0\right)$ such that the deviation is $< 0.1\%$ with confidence $> 1 - 10^{-4}$. We treat the declaration of `SaVer-Toolbox` as the ground truth.

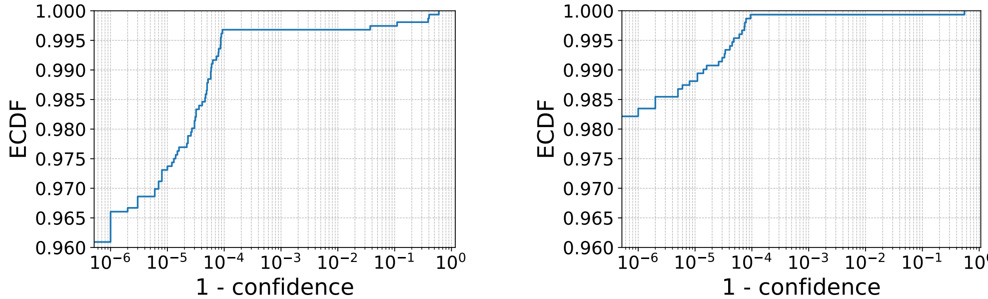
Results. Table 3 reports the counts of (declared TRUE/FALSE/No-declaration) vs. ground truth. Both BaB-prob versions and PV match ground truth on all toy problems; PROVEN returns only a subset (sound but incomplete), and SDP is mixed. It should be noted that this does not indicate that SDP is unsound, because it does not declare a TRUE problem as FALSE.

C.2 Other Experiment Setups

Untrained models. The architecture for the untrained MLP models are same as that for the toy MLP model with the difference that the number of input features D_i, the number of hidden features D_h, and the number of hidden layers N_h are not fixed. The architecture for the untrained CNN models are same as that for the toy CNN model with the difference that the input shape $(1, W_i, H_i)$ with $W_i = H_i$, the number of hidden channels C_h, and the number of hidden layers N_h are not fixed. The weights and biases of all layers in the MLP and CNN models are also randomized with Gaussian distribution $\mathcal{N}(0, 0.25)$. We test on different combination of (D_i, D_h, N_h) for MLP models and $(W_i/H_i, C_h, N_h)$ for CNN models. For each combination, we generated 30 different problems, each consists of a sample $\mathbf{x}_0 = 0$ and a randomly generated network f such that $f(\mathbf{x}_0) > 0$. The noise is zero-mean with diagonal covariance, chosen so that its 99.7%-confidence ball has radius 0.002 for MLP models and 0.01 for CNN models. We use batch size of 4 for BaB-prob and PV, and split depth of 1 for this experiment.

MNIST and CIFAR-10 models. The MNIST and CIFAR-10 models have similar architecture as untrained models with the difference that the number of input features (or input shape) is fixed, the number of output features is 10, and the trained CNN models have a BatchNorm2d layer before each ReLU layer. We trained MLP models with different combination of (D_h, N_h) and CNN models with different combination of (C_h, N_h). The models are trained with cross-entropy loss and Adam optimizer. The batch size for training is 64. The learning rate is 0.001. We train each model for 20 iterations and use the checkpoint with the lowest loss on validation dataset for verification. For each model, we randomly selected 30 correctly classified samples from the training set. The noise is zero-mean with diagonal covariance, chosen so that its 99.7%-confidence ball has radius 0.02 for MNIST models and 0.01 for CIFAR-10 models. The output specification for each sample is $\mathcal{Y} = \{\mathbf{y} \in \mathbb{R}^{10} : (\mathbf{e}_t - \mathbf{e}_a)^\mathsf{T}\mathbf{y} > 0\}$, where $\mathbf{e}_t$ and $\mathbf{e}_a$ are the standard basis vectors, t is the index

[2] `https://github.com/vigsiv/SaVer-Toolbox`



(a) BaB-prob-ordered (b) BaB+BaBSR-prob

Figure 4: ECDF of confidence for BaB-prob-ordered and BaB+BaBSR-prob.

of the ground-truth label and $a \neq t$ is a randomly selected attacking label. $f(\mathbf{x}) \in \mathcal{Y}$ indicates that $\mathbf{x}$ is not misclassified by index a. We use batch size of 4 for BaB-prob and PV, and split depth of 1 for this experiment.

VNN-COMP 2025 benchmarks. We conducted evaluations on the following benchmark suites: `acasxu_2023`, `cersyve`, `cifar100_2024`, `collins_rul_cnn_2022`, `cora_2024`, `linearizenn_2024`, `relusplitter`, and `safenlp_2024`. Due to GPU memory limitations, for `cifar100_2024` we evaluated only the medium models. For `relusplitter`, we tested all MNIST models but only the oval21 models among the CIFAR-10 models. Models from other benchmark datasets were excluded either because they contained layer types not supported by our method or were too large to fit within GPU memory. For simplicity, we randomly selected one input region and one output specification from each original problem. The noise distribution is zero-mean with diagonal covariance, scaled such that its 99.7%-confidence ellipsoid matches the axis-aligned radii given in the original problem. In terms of solver configuration, we used a batch size of 8 for both BaB-prob and PV, with a split depth of 2. For the `cifar100_2024` benchmark, the batch size and split depth were set to 1 for BaB-prob, and the batch size was set to 4 for PV. Furthermore, since the original radii in `cifar100_2024` were too small to present a meaningful verification challenge, we doubled their values.

C.3 SDP Experimental Results

Untrained models. SDP failed to solve any MLP problem within the time limit and ran out of RAM on the CNN problems.

MNIST and CIFAR-10 models. SDP failed to solve any MLP problem within the time limit and ran out of RAM on the CNN problems.

VNN-COMP 2025 benchmarks. The SDP solver does not directly support `cersyve`, `cifar100_2024`, `linearizenn_2024`, `relusplitter` (CIFAR-10). Besides, it ran out of RAM on `cora_2024`, `collins_rul_cnn_2022` and `relusplitter` (MNIST). On `acasxu_2023` and `safenlp_2024`, it hit the time limit on all the problems.

C.4 Confidence of BaB-prob

Derivation of confidence

In our experiments, BaB-prob evaluates the per-branch probability in Equation 4 by Monte Carlo sampling, using $N = 10^5$ i.i.d. samples for each probability it needs to compute. Let P_ℓ and P_u be the true global lower and upper probability bounds, and let $\hat{P}_\ell$ and $\hat{P}_u$ be their empirical values. When BaB-prob terminates, either $\hat{P}_\ell \geq \eta$ or $\hat{P}_u < \eta$. Then, the following proposition gives the confidence for the certification.

Proposition 6.

$$\mathbb{P}\left(P_\ell \geq \eta\right) \geq 1 - \exp\left(-\frac{N(\hat{P}_\ell - \eta)^2}{2V_1 + \frac{2}{3}(\hat{P}_\ell - \eta)}\right), \quad \textit{if } \hat{P}_\ell \geq \eta; \tag{36a}$$

$$\mathbb{P}\left(P_u < \eta\right) \geq 1 - \exp\left(-\frac{N(\eta - \hat{P}_u)^2}{2V_2 + \frac{2}{3}(\eta - \hat{P}_u)}\right), \quad \textit{if } \hat{P}_u < \eta. \tag{36b}$$

where

$$V_1 = \sum_{\langle p_\ell, p_u, \mathcal{C}\rangle \in \mathcal{B}} p_\ell(1 - p_\ell), \tag{37a}$$

$$V_2 = \sum_{\langle p_\ell, p_u, \mathcal{C}\rangle \in \mathcal{B}} p_u(1 - p_u). \tag{37b}$$

Proof. We prove the case of $\hat{P}_\ell \geq \eta$, and $\hat{P}_u < \eta$ can be proved similarly.

Denote $p_B \coloneqq p_\ell$ for $B = \langle p_\ell, p_u, \mathcal{C}\rangle \in \mathcal{B}$. Let $\hat{p}_B$ be the empirical estimation for p_B by Monte Carlo Sampling. Then,

$$\hat{p}_B = \frac{1}{N}\sum_{i=1}^{N} Z_{B,i}, \quad Z_{B,i} \sim \text{Bernoulli}(p_B), \tag{38}$$

and

$$P_\ell = \sum_{B\in\mathcal{B}} p_B, \quad \hat{P}_\ell = \sum_{B\in\mathcal{B}} \hat{p}_B. \tag{39}$$

Consider the estimation error

$$\hat{P}_\ell - P_\ell = \sum_{B\in\mathcal{B}}\sum_{i=1}^{N}\frac{1}{N}(Z_{B,i} - p_B). \tag{40}$$

The summands are independent, mean-zero, and bounded in $[-\frac{1}{N}, \frac{1}{N}]$; their total variance is

$$\text{Var}\Big(\sum_{B\in\mathcal{B}}\sum_{i=1}^{N}\frac{1}{N}(Z_{B,i} - p_B)\Big) = \frac{1}{N^2}\sum_{B\in\mathcal{B}}\sum_{i=1}^{N} p_B(1 - p_B) = \frac{1}{N}V_1. \tag{41}$$

Applying Bernstein's inequality with $\varepsilon = \hat{P}_\ell - \eta$ (Boucheron et al. (2013)),

$$\mathbb{P}\left(\hat{P}_\ell - P_\ell \geq \varepsilon\right) \leq \exp\left(-\frac{\frac{1}{2}\varepsilon^2}{\frac{1}{N}V_1 + \frac{1}{3N}\varepsilon}\right) = \exp\left(-\frac{N\varepsilon^2}{2V_1 + \frac{2}{3}\varepsilon}\right). \tag{42}$$

Therefore,

$$\mathbb{P}\left(P_\ell \geq \eta\right) \geq 1 - \exp\left(-\frac{N\varepsilon^2}{2V_1 + \frac{2}{3}\varepsilon}\right) \tag{43}$$

□

The true values of p_ℓ and p_u in Equation 37 are not directly accessible, so we use their empirical results as replacement. In our experiments, if BaB-prob-ordered or BaB+BaBSR-prob produces a declaration, that is, $\hat{P}_l \geq \eta$ or $\hat{P}_u < \eta$, but with confidence below $1 - 10^{-4}$, the algorithm continues running until the confidence reaches $1 - 10^{-4}$ or the time limit is hit. If when the algorithm terminates with a declaration but with confidence remaining below $1 - 10^{-4}$, we still count it a successful verification. The following results provide a statistical characterization of the achieved confidence levels.

Confidence results

Figure 4 presents the Empirical Cumulative Distribution Function (ECDF) of confidence values for BaB-prob-ordered and BaB+BaBSR-prob across all successfully verified problems. Both methods achieve confidence greater than $1 - 10^{-4}$ in over 99.5% of cases. This demonstrates that, in practice, the vast majority of problems are certified with very high confidence by both BaB-prob-ordered and BaB+BaBSR-prob.

D LLM USAGE

The authors acknowledge the use of GPT-4 and GPT-5 for polishing the main text.