Supplementary Materials for
Towards Evaluating and Training Verifiably Robust Neural Networks

Contents

A Theory

In the appendix, we use the same notations as the ones used in the main text. If new notations are used, we will explain them at the place where they first arise.

A.1 Details of CROWN

Suppose we want to compute lower bound for the quantity $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$. $\mathbf{W}^{obj}$ and $\mathbf{b}^{obj}$ are the weight and bias that connect $\mathbf{z}^{(k)}$ to the quantity of interests. For example, the quantity becomes the margin $\boldsymbol{\omega}(\mathbf{x})$ if we choose appropriate $\mathbf{W}^{obj}$ and set $\mathbf{b}^{obj} = 0, k = m$. Assume we already know the bounds of the pre-activation of the $(k-1)$-th layer:

$$\mathbf{l}^{(k-1)} \leq \mathbf{z}^{(k-1)} \leq \mathbf{u}^{(k-1)}, \forall \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon). \tag{1}$$

Next CROWN finds two linear functions of $\mathbf{z}^{(k-1)}$ to bound $\mathbf{a}^{(k-1)} = \sigma(\mathbf{z}^{(k-1)})$ in the intervals specified by $\mathbf{l}^{(k-1)}, \mathbf{u}^{(k-1)}$.

$$\begin{aligned} &\mathbf{h}^{(k-1)L}(\mathbf{z}^{(k-1)}) \leq \sigma(\mathbf{z}^{(k-1)}) \leq \mathbf{h}^{(k-1)U}(\mathbf{z}^{(k-1)}), \\ &\forall \mathbf{l}^{(k-1)} \leq \mathbf{z}^{(k-1)} \leq \mathbf{u}^{(k-1)}, \end{aligned} \tag{2}$$

where

$$\begin{aligned} \mathbf{h}^{(k-1)L}(\mathbf{z}^{(k-1)}) &= \mathbf{s}^{(k-1)L} * \mathbf{z}^{(k-1)} + \mathbf{t}^{(k-1)L}, \\ \mathbf{h}^{(k-1)U}(\mathbf{z}^{(k-1)}) &= \mathbf{s}^{(k-1)U} * \mathbf{z}^{(k-1)} + \mathbf{t}^{(k-1)U}. \end{aligned} \tag{3}$$

Here we use "$*$" to denote elementwise product. $\mathbf{s}^{(k-1)L/U}, \mathbf{t}^{(k-1)L/U}$ are constant vectors of the same dimension of $\mathbf{z}^{(k-1)}$. The linear functions $\mathbf{h}^{(k-1)L/U}(\mathbf{z}^{(k-1)})$ are also called bounding lines, as they bound the nonlinear function $\sigma(\mathbf{z}^{(k-1)})$ in the intervals determined by $\mathbf{l}^{(k-1)}, \mathbf{u}^{(k-1)}$. These bounding lines are guaranteed to exist as long as the activation function σ is bounded in a given interval, which is true for most activation functions, *e.g.*, ReLU, Sigmoid, Arctanh. Now we can compute lower bound of $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ through the following procedure:

$$\begin{aligned} &\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \\ =&\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \\ =&\mathbf{W}^{obj}(\mathbf{W}^{(k)}\mathbf{a}^{(k-1)} + \mathbf{b}^{(k)}) + \mathbf{b}^{obj} \\ =&\mathbf{W}^{eqv}\mathbf{a}^{(k-1)} + \mathbf{b}^{eqv} \\ \geq&relu(\mathbf{W}^{eqv})\mathbf{h}^{(k-1)L}(\mathbf{z}^{(k-1)}) + neg(\mathbf{W}^{eqv})\mathbf{h}^{(k-1)U}(\mathbf{z}^{(k-1)}) + \mathbf{b}^{eqv}, \end{aligned} \tag{4}$$

where $\mathbf{W}^{eqv} = \mathbf{W}^{obj}\mathbf{W}^{(k)}, \mathbf{b}^{eqv} = \mathbf{W}^{obj}\mathbf{b}^{(k)} + \mathbf{b}^{obj}$, $relu$ is the elementwise ReLU function and neg is the elementwise version of the function $neg(x) = x$, if $x \leq 0$; $neg(x) = 0$, else. Next CROWN plugs in the definition of $\mathbf{h}^{(k-1)L}(\mathbf{z}^{(k-1)}$ and $\mathbf{h}^{(k-1)L}(\mathbf{z}^{(k-1)}$:

$$\begin{aligned} &\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \\ \geq&relu(\mathbf{W}^{eqv})\mathbf{h}^{(k-1)L}(\mathbf{z}^{(k-1)}) + neg(\mathbf{W}^{eqv})\mathbf{h}^{(k-1)U}(\mathbf{z}^{(k-1)}) + \mathbf{b}^{eqv} \\ =&relu(\mathbf{W}^{eqv})[\mathbf{s}^{(k-1)L} * \mathbf{z}^{(k-1)} + \mathbf{t}^{(k-1)L}] + neg(\mathbf{W}^{eqv})[\mathbf{s}^{(k-1)U} * \mathbf{z}^{(k-1)} + \mathbf{t}^{(k-1)U}] + \mathbf{b}^{eqv} \\ =&[relu(\mathbf{W}^{eqv}) * \mathbf{s}^{(k-1)L} + neg(\mathbf{W}^{eqv}) * \mathbf{s}^{(k-1)U}]\mathbf{z}^{(k-1)} + relu(\mathbf{W}^{eqv})\mathbf{t}^{(k-1)L} + neg(\mathbf{W}^{eqv})\mathbf{t}^{(k-1)U} + \mathbf{b}^{eqv} \\ =&\mathbf{W}^{(k,k-1)L}\mathbf{z}^{(k-1)} + \mathbf{b}^{(k,k-1)L}, \end{aligned} \tag{5}$$

where the operator "$*$" between a matrix $\mathbf{W}$ and a vector $\mathbf{s}$ is defined as $(\mathbf{W}*\mathbf{s})_{ij} = \mathbf{W}_{ij}\mathbf{s}_j$ and $\mathbf{W}^{(k,k-1)L}, \mathbf{b}^{(k,k-1)L}$ are defined as

$$\mathbf{W}^{(k,k-1)L} = relu(\mathbf{W}^{eqv}) * \mathbf{s}^{(k-1)L} + neg(\mathbf{W}^{eqv}) * \mathbf{s}^{(k-1)U}, \tag{6}$$

$$\mathbf{b}^{(k,k-1)L} = relu(\mathbf{W}^{eqv})\mathbf{t}^{(k-1)L} + neg(\mathbf{W}^{eqv})\mathbf{t}^{(k-1)U} + \mathbf{b}^{eqv}. \tag{7}$$

We use superscript $(k, k-1)L$ to denote the last line of (5) is a linear function of $\mathbf{z}^{(k-1)}$ that lower bounds a linear function of $\mathbf{z}^k$. Now let's compare the first line and last line of (5). The last line has the same form as the first line, except that the variable changes from $\mathbf{z}^{(k)}$ to $\mathbf{z}^{(k-1)}$. If we know the lower and upper bounds of $\mathbf{z}^{(k-2)}$ and find two linear functions of $\mathbf{z}^{(k-2)}$ to bound $\mathbf{a}^{(k-2)}$, similar to the ones shown in (1) and (2). We can derive a linear function of $\mathbf{z}^{(k-2)}$ to lower bound $\mathbf{W}^{(k,k-1)L}\mathbf{z}^{(k-1)} + \mathbf{b}^{(k,k-1)L}$ following the same gist in (4):

$$\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \geq \mathbf{W}^{(k,k-1)L}\mathbf{z}^{(k-1)} + \mathbf{b}^{(k,k-1)L} \geq \mathbf{W}^{(k,k-2)L}\mathbf{z}^{(k-2)} + \mathbf{b}^{(k,k-2)L}. \tag{8}$$

We can repeat the above the procedure: Back-propagate layer by layer until the first layer $\mathbf{z}^{(1)}$, which leads us to

$$\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \geq \mathbf{W}^{(k,1)L}\mathbf{z}^{(1)} + \mathbf{b}^{(k,1)L}. \tag{9}$$

Notice $\mathbf{z}^{(1)} = \mathbf{W}^{(1)}\mathbf{x} + \mathbf{b}^{(1)}$. We plug it in the right side of (9) and obtain a linear function of $\mathbf{x}$.

$$\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \geq \tilde{\mathbf{W}}^{(k)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k)L}, \tag{10}$$

where $\tilde{\mathbf{W}}^{(k)L} = \mathbf{W}^{(k,1)L}\mathbf{W}^{(1)}, \tilde{\mathbf{b}}^{(k)L} = \mathbf{W}^{(k,1)L}\mathbf{b}^{(1)} + \mathbf{b}^{(k,1)L}$. Now we can compute the closed-form lower bound of $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ through Holder's theorem:

$$\begin{aligned}\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} &\geq \tilde{\mathbf{W}}^{(k)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k)L} \\ &\geq \tilde{\mathbf{W}}^{(k)L}\mathbf{x}_0 + \tilde{\mathbf{b}}^{(k)L} - \epsilon||\tilde{\mathbf{W}}^{(k)L}||_q, \\ &\forall\, \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon),\end{aligned} \tag{11}$$

where $1/p + 1/q = 1$ and $||\tilde{\mathbf{W}}^{(k)L}||_q$ denotes a column vector that is composed of the q-norm of every row in $\tilde{\mathbf{W}}^{(k)L}$. We can compute a linear function of $\mathbf{x}$ to upper bound $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ in the same manner and then compute its closed-form upper bound.

A.1.1 Back-propagate through $\mathbf{a}^{(k)}$.

In the above discussion, CROWN back-propagates layer by layer through $\mathbf{z}^{(k)}$ as shown in (8). Equivalently, we can also back-propagate through $\mathbf{a}^{(k)}$. Assume we want to compute bounds for the quantity $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$.

$$\begin{aligned}&\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \\ =&\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \\ =&\mathbf{W}^{obj}(\mathbf{W}^{(k)}\mathbf{a}^{(k-1)} + \mathbf{b}^{(k)}) + \mathbf{b}^{obj} \\ =&\mathbf{W}^{eqv}\mathbf{a}^{(k-1)} + \mathbf{b}^{eqv},\end{aligned} \tag{12}$$

where $\mathbf{W}^{eqv} = \mathbf{W}^{obj}\mathbf{W}^{(k)}, \mathbf{b}^{eqv} = \mathbf{W}^{obj}\mathbf{b}^{(k)} + \mathbf{b}^{obj}$. Therefore, computing bounds for $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ is equivalent to computing bounds for $\mathbf{W}^{eqv}\mathbf{a}^{(k-1)} + \mathbf{b}^{eqv}$. Next we have

$$\begin{aligned}&\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \\ =&\mathbf{W}^{eqv}\mathbf{a}^{(k-1)} + \mathbf{b}^{eqv} \\ \geq&relu(\mathbf{W}^{eqv})\mathbf{h}^{(k-1)L}(\mathbf{z}^{(k-1)}) + neg(\mathbf{W}^{eqv})\mathbf{h}^{(k-1)U}(\mathbf{z}^{(k-1)}) + \mathbf{b}^{eqv} \\ =&relu(\mathbf{W}^{eqv})[\mathbf{s}^{(k-1)L} * \mathbf{z}^{(k-1)} + \mathbf{t}^{(k-1)L}] + neg(\mathbf{W}^{eqv})[\mathbf{s}^{(k-1)U} * \mathbf{z}^{(k-1)} + \mathbf{t}^{(k-1)U}] + \mathbf{b}^{eqv} \\ =&[relu(\mathbf{W}^{eqv}) * \mathbf{s}^{(k-1)L} + neg(\mathbf{W}^{eqv}) * \mathbf{s}^{(k-1)U}]\mathbf{z}^{(k-1)} + relu(\mathbf{W}^{eqv})\mathbf{t}^{(k-1)L} + neg(\mathbf{W}^{eqv})\mathbf{t}^{(k-1)U} + \mathbf{b}^{eqv} \\ =&\mathbf{W}^{(k,k-1)L}\mathbf{z}^{(k-1)} + \mathbf{b}^{(k,k-1)L} \\ =&\mathbf{W}^{(k,k-1)L}[\mathbf{W}^{(k-1)}\mathbf{a}^{(k-2)} + \mathbf{b}^{(k-1)}] + \mathbf{b}^{(k,k-1)L} \\ =&\mathbf{W}^{(k,k-1)L}\mathbf{W}^{(k-1)}\mathbf{a}^{(k-2)} + \mathbf{W}^{(k,k-1)L}\mathbf{b}^{(k-1)} + \mathbf{b}^{(k,k-1)L} \\ =&\boldsymbol{\Omega}^{(k,k-2)L}\mathbf{a}^{(k-2)} + \boldsymbol{\delta}^{(k,k-2)L},\end{aligned} \tag{13}$$

where

$$\mathbf{W}^{(k,k-1)L} = relu(\mathbf{W}^{eqv}) * \mathbf{s}^{(k-1)L} + neg(\mathbf{W}^{eqv}) * \mathbf{s}^{(k-1)U}, \tag{14}$$

$$\mathbf{b}^{(k,k-1)L} = relu(\mathbf{W}^{eqv})\mathbf{t}^{(k-1)L} + neg(\mathbf{W}^{eqv})\mathbf{t}^{(k-1)U} + \mathbf{b}^{eqv}, \tag{15}$$

$$\mathbf{\Omega}^{(k,k-2)L} = \mathbf{W}^{(k,k-1)L}\mathbf{W}^{(k-1)}, \tag{16}$$

$$\boldsymbol{\delta}^{(k,k-2)L} = \mathbf{W}^{(k,k-1)L}\mathbf{b}^{(k-1)} + \mathbf{b}^{(k,k-1)L}. \tag{17}$$

We can repeat this back-propagation procedure until the input $\mathbf{a}^{(0)}$:

$$\begin{aligned}&\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} = \mathbf{W}^{eqv}\mathbf{a}^{(k-1)} + \mathbf{b}^{eqv} \geq \mathbf{\Omega}^{(k,k-2)L}\mathbf{a}^{(k-2)} + \boldsymbol{\delta}^{(k,k-2)L} \\ &\geq \mathbf{\Omega}^{(k,k-3)L}\mathbf{a}^{(k-3)} + \boldsymbol{\delta}^{(k,k-3)L} \geq \cdots \geq \mathbf{\Omega}^{(k,0)L}\mathbf{a}^{(0)} + \boldsymbol{\delta}^{(k,0)L} = \mathbf{\Omega}^{(k,0)L}\mathbf{x} + \boldsymbol{\delta}^{(k,0)L}.\end{aligned} \tag{18}$$

Then the closed-form lower bound of $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ can be computed by

$$\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \geq \mathbf{\Omega}^{(k,0)L}\mathbf{x} + \boldsymbol{\delta}^{(k,0)L} \geq \mathbf{\Omega}^{(k,0)L}\mathbf{x}_0 + \boldsymbol{\delta}^{(k,0)L} - \epsilon||\mathbf{\Omega}^{(k,0)L}||_q, \forall \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon). \tag{19}$$

The linear upper bound and closed-form upper bound of $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ can be computed in a similar way.

We emphasize that back-propagation through $\mathbf{a}^{(k)}$ is equivalent to back-propagation through $\mathbf{z}^{(k)}$. Back-propagation through $\mathbf{a}^{(k)}$ introduced in this section will be useful when proving CROWN-LBP is tighter than LBP in Section A.6.3.

A.1.2 Summary of CROWN

Lyu *et al* [2] present a concise description of the computation process of CROWN in their Appendix A.2. In this section, we directly borrow their results and adapt to our own notation. This concise description will be useful when we prove IBP is a special case of CROWN in Section A.5.

Suppose we want to compute lower bound of $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$. Given the bounds of the previous $k-1, (k \geq 2)$ layers, $\mathbf{l}^{(v)}, \mathbf{u}^{(v)}, v = 1, 2, \cdots, k-1$, one could compute bounds of the k-th layer through the following steps. Repeat the following iteration:

$$\begin{aligned}\hat{\mathbf{W}}^{(v)} &= [relu(\hat{\mathbf{W}}^{(v+1)}) * \mathbf{s}^{(v)L} + neg(\hat{\mathbf{W}}^{(v+1)}) * \mathbf{s}^{(v)U}]W^{(v)}, \\ \hat{\mathbf{b}}^{(v)} &= relu(\hat{\mathbf{W}}^{(v+1)})(\mathbf{s}^{(v)L} * \mathbf{b}^{(v)} + \mathbf{t}^{(v)L}) + neg(\hat{\mathbf{W}}^{(v+1)})^\top(\mathbf{s}^{(v)U} * \mathbf{b}^{(v)} + \mathbf{t}^{(v)U}) + \hat{\mathbf{b}}^{(v+1)}, \\ &v = k-1, k-2, \cdots, 1.\end{aligned} \tag{20}$$

to obtain $\hat{\mathbf{W}}^{(1)}$ and $\hat{\mathbf{b}}^{(1)}$, where the starting variable is defined as

$$\hat{\mathbf{W}}^{(k)} = \mathbf{W}^{eqv} = \mathbf{W}^{obj}\mathbf{W}^{(k)}, \tag{21}$$

$$\hat{\mathbf{b}}^{(k)} = \mathbf{b}^{eqv} = \mathbf{W}^{obj}\mathbf{b}^{(k)} + \mathbf{b}^{obj}. \tag{22}$$

Then the closed form lower bound of $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ can be computed by the following formula:

$$\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \geq \hat{\mathbf{W}}^{(1)}\mathbf{x} + \hat{\mathbf{b}}^{(1)} \geq \hat{\mathbf{W}}^{(1)}\mathbf{x}_0 + \hat{\mathbf{b}}^{(1)} - \epsilon||\hat{\mathbf{W}}^{(1)}||_q, \forall \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon). \tag{23}$$

where $1/p + 1/q = 1$ and $||\hat{\mathbf{W}}^{(1)}||_q$ denotes a column vector that is composed of the q-norm of every row in $\hat{\mathbf{W}}^{(1)}$. To compute the upper bound of $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$, replace the iteration in (20) with the following iteration:

$$\begin{aligned}\hat{\mathbf{W}}^{(v)} &= [neg(\hat{\mathbf{W}}^{(v+1)}) * \mathbf{s}^{(v)L} + relu(\hat{\mathbf{W}}^{(v+1)}) * \mathbf{s}^{(v)U}]W^{(v)}, \\ \hat{\mathbf{b}}^{(v)} &= neg(\hat{\mathbf{W}}^{(v+1)})(\mathbf{s}^{(v)L} * \mathbf{b}^{(v)} + \mathbf{t}^{(v)L}) + relu(\hat{\mathbf{W}}^{(v+1)})^\top(\mathbf{s}^{(v)U} * \mathbf{b}^{(v)} + \mathbf{t}^{(v)U}) + \hat{\mathbf{b}}^{(v+1)}, \\ &v = k-1, k-2, \cdots, 1.\end{aligned} \tag{24}$$

Then the closed-form upper bound of $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ can be computed by the following formula:

$$\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \leq \hat{\mathbf{W}}^{(1)}\mathbf{x} + \hat{\mathbf{b}}^{(1)} \leq \hat{\mathbf{W}}^{(1)}\mathbf{x}_0 + \hat{\mathbf{b}}^{(1)} + \epsilon||\hat{\mathbf{W}}^{(1)}||_q, \forall \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon). \tag{25}$$

The bounds of the first layer $\mathbf{z}^{(1)}$ can be computed directly through the following formula:

$$\begin{aligned}\mathbf{l}^{(1)} &= \min_{\mathbf{a}^{(0)} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon)} \mathbf{W}^{(1)}\mathbf{a}^{(0)} + \mathbf{b}^{(1)} = \mathbf{W}^{(1)}\mathbf{x}_0 + \mathbf{b}^{(1)} - \epsilon||\mathbf{W}^{(1)}||_q, \\ \mathbf{u}^{(1)} &= \max_{\mathbf{a}^{(0)} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon)} \mathbf{W}^{(1)}\mathbf{a}^{(0)} + \mathbf{b}^{(1)} = \mathbf{W}^{(1)}\mathbf{x}_0 + \mathbf{b}^{(1)} + \epsilon||\mathbf{W}^{(1)}||_q.\end{aligned} \tag{26}$$

A.2 Details of Relaxed CROWN

As the same in the CROWN deduction process in Section A.1, suppose we want to compute bound for the quantity $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$. In the original crown process, we first compute linear functions of $\mathbf{x}$ to bound the pre-activation of the first $(k-1)$ layers:

$$\begin{aligned} \tilde{\mathbf{W}}^{(v)L}\mathbf{x} + \tilde{\mathbf{b}}^{(v)L} \leq \mathbf{z}^{(v)} \leq \tilde{\mathbf{W}}^{(v)U}\mathbf{x} + \tilde{\mathbf{b}}^{(v)U}, \\ \forall\, \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon), v = 1, 2, \cdots, k-1, \end{aligned} \tag{27}$$

and then use these linear functions of $\mathbf{x}$ to compute closed-form bounds for the first $(k-1)$ layers. We argue that in the back-propagation process of computing bounds for $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ in (8), we don't need to back-propagate to the first layer. We can stop at any intermediate layer and plug in the linear functions in (27) of this intermediate layer to get a linear function of $\mathbf{x}$ to bound $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$. Specifically, assume we decide to back propagate v layers in (8):

$$\begin{aligned} \mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} &\geq \mathbf{W}^{(k,k-1)L}\mathbf{z}^{(k-1)} + \mathbf{b}^{(k,k-1)L} \\ &\geq \cdots \geq \mathbf{W}^{(k,k-v)L}\mathbf{z}^{(k-v)} + \mathbf{b}^{(k,k-v)L}, v < k. \end{aligned} \tag{28}$$

We already know

$$\tilde{\mathbf{W}}^{(k-v)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k-v)L} \leq \mathbf{z}^{(k-v)} \leq \tilde{\mathbf{W}}^{(k-v)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-v)U}. \tag{29}$$

We can directly plug (29) to (28) to lower bound $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$:

$$\begin{aligned} &\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj} \\ \geq &\mathbf{W}^{(k,k-v)L}\mathbf{z}^{(k-v)} + \mathbf{b}^{(k,k-v)L} \\ \geq &relu(\mathbf{W}^{(k,k-v)L})[\tilde{\mathbf{W}}^{(k-v)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k-v)L}] + \\ &neg(\mathbf{W}^{(k,k-v)L})[\tilde{\mathbf{W}}^{(k-v)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-v)U}] + \mathbf{b}^{(k,k-v)L} \\ = &[relu(\mathbf{W}^{(k,k-v)L})\tilde{\mathbf{W}}^{(k-v)L} + neg(\mathbf{W}^{(k,k-v)L})\tilde{\mathbf{W}}^{(k-v)U}]\mathbf{x} \\ &+ relu(\mathbf{W}^{(k,k-v)L})\tilde{\mathbf{b}}^{(k-v)L} + neg(\mathbf{W}^{(k,k-v)L})\tilde{\mathbf{b}}^{(k-v)U} \\ &+ \mathbf{b}^{(k,k-v)L}. \end{aligned} \tag{30}$$

Now the last line of (30) is already a linear function of $\mathbf{x}$ and we can compute the closed-form lower bound of $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ in the same manner as shown in (11). The upper bound of $\mathbf{W}^{obj}\mathbf{z}^{(k)} + \mathbf{b}^{obj}$ can also be computed by back-propagate only v layers in the same gist.

Now we have shown that we can only back-propagate v layers, instead of back-propagating to the first layer, when computing bounds for the k-th layer. In fact, we can only back-propagate v layers when computing bounds for any layer. If the layer index k is less than or equal to v, we just back-propagate to the first layer. In other words, we back-propagate at most v layers when computing bounds for any layer in the process of CROWN. We call this relaxed version of CROWN **Relaxed-CROWN-v**. The benefits of Relaxed-CROWN-v are that it reduces the computation complexity of CROWN from $\mathcal{O}(m^2)$ to $\mathcal{O}(vm)$ and reduces memory cost from $\mathcal{O}(n_{k_1}n_{k_2})$ to

$$\mathcal{O}[\max_{k\in\{1,2,\cdots,m\}}(n_k \max\{n_{k-1}, n_{k-2}, \cdots, n_{k-v}\})], \tag{31}$$

where those n with negative subscript are defined as 0.

A.3 Details of LBP

Linear bound propagation (LBP) is a special case of the relaxed CROWN. Specifically, LBP is equivalent to Relaxed-CROWN-1, namely, we only back-propagate 1 layer when computing bounds for any layer in the process of CROWN.

Assume we already know two linear functions of $\mathbf{x}$ to bound $\mathbf{z}^{(k-1)}$:

$$\tilde{\mathbf{W}}^{(k-1)L}\mathbf{x}+\tilde{\mathbf{b}}^{(k-1)L} \leq \mathbf{z}^{(k-1)} \leq \tilde{\mathbf{W}}^{(k-1)U}\mathbf{x}+\tilde{\mathbf{b}}^{(k-1)U}. \tag{32}$$

We then compute the closed-form bounds $\mathbf{l}^{(k-1)}, \mathbf{u}^{(k-1)}$ of $\mathbf{z}^{(k-1)}$ using these two linear functions, and then choose two linear functions $\mathbf{h}^{(k-1)L}(\mathbf{z}^{(k-1)}), \mathbf{h}^{(k-1)U}(\mathbf{z}^{(k-1)})$ to bound $\mathbf{a}^{(k-1)}=\sigma(\mathbf{z}^{(k-1)})$ as shown in (3). Then

$$\begin{aligned}
\mathbf{z}^{(k)} &= \mathbf{W}^{(k)}\mathbf{a}^{(k-1)}+\mathbf{b}^{(k)} \\
&\geq relu(\mathbf{W}^{(k)})(\mathbf{s}^{L(k-1)}*\mathbf{z}^{(k-1)}+\mathbf{t}^{(k-1)L})+neg(\mathbf{W}^{(k)})(\mathbf{s}^{(k-1)U}*\mathbf{z}^{(k-1)}+\mathbf{t}^{(k-1)U})+\mathbf{b}^{(k)} \\
&= (relu(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)L}+neg(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)U})\mathbf{z}^{(k-1)} \\
&\quad + relu(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)L}+neg(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)U}+\mathbf{b}^{(k)} \\
&\geq [relu(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)L}][\tilde{\mathbf{W}}^{(k-1)L}\mathbf{x}+\tilde{\mathbf{b}}^{(k-1)L}] \\
&\quad + [neg(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)U}][\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x}+\tilde{\mathbf{b}}^{(k-1)U}] \\
&\quad + relu(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)L}+neg(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)U}+\mathbf{b}^{(k)} \\
&= \{[relu(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)L}]\tilde{\mathbf{W}}^{(k-1)L}+[neg(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)U}]\tilde{\mathbf{W}}^{(k-1)U}\}\mathbf{x} \\
&\quad + [relu(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)L}]\tilde{\mathbf{b}}^{(k-1)L}+[neg(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)U}]\tilde{\mathbf{b}}^{(k-1)U} \\
&\quad + relu(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)L}+neg(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)U}+\mathbf{b}^{(k)}.
\end{aligned} \tag{33}$$

Note that in the above derivation, we already assume that $\mathbf{s}^{(k-1)L} \geq 0, \mathbf{s}^{(k-1)U} \geq 0$. If we define

$$\begin{aligned}
\tilde{\mathbf{W}}^{(k)L} &= [relu(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)L}]\tilde{\mathbf{W}}^{(k-1)L}+[neg(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)U}]\tilde{\mathbf{W}}^{(k-1)U}, \\
\tilde{\mathbf{b}}^{(k)L} &= \mathbf{b}^{(k)}+[relu(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)L}]\tilde{\mathbf{b}}^{(k-1)L}+[neg(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)U}]\tilde{\mathbf{b}}^{(k-1)U} \\
&\quad + relu(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)L}+neg(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)U}.
\end{aligned} \tag{34}$$

Then we have

$$\mathbf{z}^{(k)} \geq \tilde{\mathbf{W}}^{(k)L}\mathbf{x}+\tilde{\mathbf{b}}^{(k)L}, \tag{35}$$

which is the conclusion presented in Theorem 1 in the main text.

In the same manner, we can define

$$\begin{aligned}
\tilde{\mathbf{W}}^{(k)U} &= [neg(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)L}]\tilde{\mathbf{W}}^{(k-1)L}+[relu(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)U}]\tilde{\mathbf{W}}^{(k-1)U}, \\
\tilde{\mathbf{b}}^{(k)U} &= \mathbf{b}^{(k)}+[neg(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)L}]\tilde{\mathbf{b}}^{(k-1)L}+[relu(\mathbf{W}^{(k)})*\mathbf{s}^{(k-1)U}]\tilde{\mathbf{b}}^{(k-1)U} \\
&\quad + neg(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)L}+relu(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)U},
\end{aligned} \tag{36}$$

and have

$$\mathbf{z}^{(k)} \leq \tilde{\mathbf{W}}^{(k)U}\mathbf{x}+\tilde{\mathbf{b}}^{(k)U}. \tag{37}$$

The closed-from bounds of $\mathbf{z}^{(k)}$ can be computed using $\tilde{\mathbf{W}}^{(k)L}\mathbf{x}+\tilde{\mathbf{b}}^{(k)L}$, $\tilde{\mathbf{W}}^{(k)U}\mathbf{x}+\tilde{\mathbf{b}}^{(k)U}$ through Holder's theorem as shown in (26).

We have shown that we can use bounds of the $(k-1)$-th layer to build bounds for the k-th layer. We can compute bounds starting from the first layer $\mathbf{z}^{(1)}$, which can be bounded by $\mathbf{W}^{(1)}\mathbf{x}+\mathbf{b}^{(1)} \leq \mathbf{z}^{(1)} \leq \mathbf{W}^{(1)}\mathbf{x}+\mathbf{b}^{(1)}$, and then compute bounds layer by layer in a forward manner until the final output of the network. The computation complexity is reduced to $\mathcal{O}(m)$ and memory cost is reduced to $\mathcal{O}(n_0 \max\{n_1, n_2, \cdots, n_m\})$, since we only need to record a matrix $\tilde{\mathbf{W}}^{(k)}$ from the input $\mathbf{x}$ to every intermediate layer $\mathbf{z}^{(k)}$. We call the resulting bound computation algorithm **Linear Bound Propagation (LBP)**, which is equivalent to Relaxed-CROWN-1.

A.3.1 Prove LBP and the forward mode in the work [3] are equivalent under certain conditions.

In the above section, we have shown that we can use linear bounds of $\mathbf{z}^{(k-1)}$ to build bounds for $\mathbf{z}^{(k)}$. However, we can also first build linear bounds for $\mathbf{a}^{(k-1)}$ and use them to build bounds for $\mathbf{z}^{(k)}$. This is the idea of the forward mode mentioned in the work [3]. We will show these two approaches are equivalent under the condition that $\mathbf{s}^{(k-1)L} \geq 0, \mathbf{s}^{(k-1)U} \geq 0$, namely, we choose bounding lines with non-negative slopes for $\mathbf{a}^{(k)}$. We will also use bounds of $\mathbf{a}^{(k)}$ when proving CROWN-LBP is tighter than LBP in Section A.6.3.

Assume we already know two linear functions of $\mathbf{x}$ to bound $\mathbf{z}^{(k-1)}$:

$$\tilde{\mathbf{W}}^{(k-1)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)L} \leq \mathbf{z}^{(k-1)} \leq \tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}. \tag{38}$$

Then

$$\begin{aligned}
\mathbf{a}^{(k-1)} &= \sigma(\mathbf{z}^{(k-1)}) \\
&\geq \mathbf{s}^{(k-1)L} * \mathbf{z}^{(k-1)} + \mathbf{t}^{(k-1)L} \\
&\geq \mathbf{s}^{(k-1)L} * [\tilde{\mathbf{W}}^{(k-1)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)L}] + \mathbf{t}^{(k-1)L} \\
&= [\mathbf{s}^{(k-1)L} * \tilde{\mathbf{W}}^{(k-1)L}]\mathbf{x} + \mathbf{s}^{(k-1)L} * \tilde{\mathbf{b}}^{(k-1)L} + \mathbf{t}^{(k-1)L}.
\end{aligned} \tag{39}$$

Note that we have used the condition that $\mathbf{s}^{(k-1)L} \geq 0$ and we define the operator "$*$" between a vector and a matrix as $(\mathbf{s} * \mathbf{W})_{ij} = \mathbf{s}_i \mathbf{W}_{ij}$. If we define

$$\begin{aligned}
\boldsymbol{\Lambda}^{(k-1)L} &= \mathbf{s}^{(k-1)L} * \tilde{\mathbf{W}}^{(k-1)L}, \\
\boldsymbol{\beta}^{(k-1)L} &= \mathbf{s}^{(k-1)L} * \tilde{\mathbf{b}}^{(k-1)L} + \mathbf{t}^{(k-1)L},
\end{aligned} \tag{40}$$

we will have

$$\mathbf{a}^{(k-1)} \geq \boldsymbol{\Lambda}^{(k-1)L}\mathbf{x} + \boldsymbol{\beta}^{(k-1)L}. \tag{41}$$

Similarly, we can define

$$\begin{aligned}
\boldsymbol{\Lambda}^{(k-1)U} &= \mathbf{s}^{(k-1)U} * \tilde{\mathbf{W}}^{(k-1)U}, \\
\boldsymbol{\beta}^{(k-1)U} &= \mathbf{s}^{(k-1)U} * \tilde{\mathbf{b}}^{(k-1)U} + \mathbf{t}^{(k-1)U},
\end{aligned} \tag{42}$$

and have

$$\mathbf{a}^{(k-1)} \leq \boldsymbol{\Lambda}^{(k-1)U}\mathbf{x} + \boldsymbol{\beta}^{(k-1)U}. \tag{43}$$

Next, we can build bounds for $\mathbf{z}^{(k)}$.

$$\begin{aligned}
\mathbf{z}^{(k)} &= \mathbf{W}^{(k)}\mathbf{a}^{(k-1)} + \mathbf{b}^{(k)} \\
&\geq relu(\mathbf{W}^{(k)})[\boldsymbol{\Lambda}^{(k-1)L}\mathbf{x} + \boldsymbol{\beta}^{(k-1)L}] + neg(\mathbf{W}^{(k)})[\boldsymbol{\Lambda}^{(k-1)U}\mathbf{x} + \boldsymbol{\beta}^{(k-1)U}] + \mathbf{b}^{(k)} \\
&= [relu(\mathbf{W}^{(k)})\boldsymbol{\Lambda}^{(k-1)L} + neg(\mathbf{W}^{(k)})\boldsymbol{\Lambda}^{(k-1)U}]\mathbf{x} + relu(\mathbf{W}^{(k)})\boldsymbol{\beta}^{(k-1)L} + neg(\mathbf{W}^{(k)})\boldsymbol{\beta}^{(k-1)U} + \mathbf{b}^{(k)}.
\end{aligned} \tag{44}$$

If we define

$$\begin{aligned}
\tilde{\mathbf{W}}^{(k)L} &= relu(\mathbf{W}^{(k)})\boldsymbol{\Lambda}^{(k-1)L} + neg(\mathbf{W}^{(k)})\boldsymbol{\Lambda}^{(k-1)U}, \\
\tilde{\mathbf{b}}^{(k)L} &= relu(\mathbf{W}^{(k)})\boldsymbol{\beta}^{(k-1)L} + neg(\mathbf{W}^{(k)})\boldsymbol{\beta}^{(k-1)U} + \mathbf{b}^{(k)},
\end{aligned} \tag{45}$$

we will have

$$\mathbf{z}^{(k)} \geq \tilde{\mathbf{W}}^{(k)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k)L}. \tag{46}$$

Similarly, we can define

$$\begin{aligned}
\tilde{\mathbf{W}}^{(k)U} &= relu(\mathbf{W}^{(k)})\boldsymbol{\Lambda}^{(k-1)U} + neg(\mathbf{W}^{(k)})\boldsymbol{\Lambda}^{(k-1)L}, \\
\tilde{\mathbf{b}}^{(k)U} &= relu(\mathbf{W}^{(k)})\boldsymbol{\beta}^{(k-1)U} + neg(\mathbf{W}^{(k)})\boldsymbol{\beta}^{(k-1)L} + \mathbf{b}^{(k)},
\end{aligned} \tag{47}$$

and have

$$\mathbf{z}^{(k)} \leq \tilde{\mathbf{W}}^{(k)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k)U}. \tag{48}$$

We will prove $\tilde{\mathbf{W}}^{(k)L}$ and $\tilde{\mathbf{b}}^{(k)L}$ defined in (45) are the same as those defined in (34) and $\tilde{\mathbf{W}}^{(k)U}$ and $\tilde{\mathbf{b}}^{(k)U}$ defined in (47) are the same as those defined in (36). In (45), we have

$$\begin{aligned}
\tilde{\mathbf{W}}^{(k)L} &= relu(\mathbf{W}^{(k)})\mathbf{\Lambda}^{(k-1)L} + neg(\mathbf{W}^{(k)})\mathbf{\Lambda}^{(k-1)U} \\
&= relu(\mathbf{W}^{(k)})[\mathbf{s}^{(k-1)L} * \tilde{\mathbf{W}}^{(k-1)L}] + neg(\mathbf{W}^{(k)})[\mathbf{s}^{(k-1)U} * \tilde{\mathbf{W}}^{(k-1)U}] \\
&= [relu(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)L}]\tilde{\mathbf{W}}^{(k-1)L} + [neg(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)U}]\tilde{\mathbf{W}}^{(k-1)U},
\end{aligned} \tag{49}$$

which is the same defined in (34). We also have

$$\begin{aligned}
\tilde{\mathbf{b}}^{(k)L} &= relu(\mathbf{W}^{(k)})\boldsymbol{\beta}^{(k-1)L} + neg(\mathbf{W}^{(k)})\boldsymbol{\beta}^{(k-1)U} + \mathbf{b}^{(k)} \\
&= relu(\mathbf{W}^{(k)})[\mathbf{s}^{(k-1)L} * \tilde{\mathbf{b}}^{(k-1)L} + \mathbf{t}^{(k-1)L}] + neg(\mathbf{W}^{(k)})[\mathbf{s}^{(k-1)U} * \tilde{\mathbf{b}}^{(k-1)U} + \mathbf{t}^{(k-1)U}] + \mathbf{b}^{(k)} \\
&= [relu(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)L}]\tilde{\mathbf{b}}^{(k-1)L} + [neg(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)U}]\tilde{\mathbf{b}}^{(k-1)U} \\
&\quad + relu(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)L} + neg(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)U} + \mathbf{b}^{(k)}.
\end{aligned} \tag{50}$$

which is the same defined in (34). In the same way, we can prove that $\tilde{\mathbf{W}}^{(k)U}$ and $\tilde{\mathbf{b}}^{(k)U}$ defined in (47) are the same as those defined in (36).

A.3.2 Prove tighter bounding lines lead to tighter bounds in LBP.

Lyu *et al* [2] prove that tighter bounding lines lead to tighter bounds in CROWN under the self-consistency condition. We follow the same gist in their work and borrow relevant notations. We prove that tighter bounding lines also lead to tighter bounds in LBP.

Define the linear lower bound of $\mathbf{z}^{(k)}$ in (35) as

$$\begin{aligned}
\mathbf{f}^{(k)L} &= \tilde{\mathbf{W}}^{(k)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k)L} \\
&= \{[relu(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)L}]\tilde{\mathbf{W}}^{(k-1)L} + [neg(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)U}]\tilde{\mathbf{W}}^{(k-1)U}\}\mathbf{x} \\
&\quad + [relu(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)L}]\tilde{\mathbf{b}}^{(k-1)L} + [neg(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)U}]\tilde{\mathbf{b}}^{(k-1)U} \\
&\quad + relu(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)L} + neg(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)U} + \mathbf{b}^{(k)} \\
&= [relu(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)L}][\tilde{\mathbf{W}}^{(k-1)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)L}] \\
&\quad + [neg(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)U}][\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}] \\
&\quad + relu(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)L} + neg(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)U} + \mathbf{b}^{(k)}.
\end{aligned} \tag{51}$$

Its i-th element is

$$\begin{aligned}
\mathbf{f}_i^{(k)L}(\mathbf{x}) &= \tilde{\mathbf{W}}_{i,:}^{(k)L}\mathbf{x} + \tilde{\mathbf{b}}_i^{(k)L} \\
&= [relu(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)L}]_{i,:}[\tilde{\mathbf{W}}^{(k-1)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)L}] \\
&\quad + [neg(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)U}]_{i,:}[\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}] \\
&\quad + relu(\mathbf{W}^{(k)})_{i,:}\mathbf{t}^{(k-1)L} + neg(\mathbf{W}^{(k)})_{i,:}\mathbf{t}^{(k-1)U} + \mathbf{b}_i^{(k)} \\
&= \sum_j [relu(\mathbf{W}_{ij}^{(k)})\mathbf{s}_j^{(k-1)L}][\tilde{\mathbf{W}}^{(k-1)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)L}]_j \\
&\quad + \sum_j [neg(\mathbf{W}_{ij}^{(k)})\mathbf{s}_j^{(k-1)U}][\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}]_j \\
&\quad + \sum_j relu(\mathbf{W}_{ij}^{(k)})\mathbf{t}_j^{(k-1)L} + \sum_j neg(\mathbf{W}_{ij}^{(k)})\mathbf{t}_j^{(k-1)U} + \mathbf{b}_i^{(k)}.
\end{aligned} \tag{52}$$

Therefore we have

$$\frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{s}_j^{(k-1)L}} = relu(\mathbf{W}_{ij}^{(k)})[\tilde{\mathbf{W}}^{(k-1)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)L}]_j \tag{53}$$

$$\frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{t}_j^{(k-1)L}} = relu(\mathbf{W}_{ij}^{(k)}) \geq 0. \tag{54}$$

and

$$\frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{s}_j^{(k-1)U}} = neg(\mathbf{W}_{ij}^{(k)})[\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}]_j \tag{55}$$

$$\frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{t}_j^{(k-1)U}} = neg(\mathbf{W}_{ij}^{(k)}) \leq 0. \tag{56}$$

Next we introduce two new variables $\mathbf{r}^{1(k-1)U}$ and $\mathbf{r}^{2(k-1)U}$:

$$\begin{aligned} \mathbf{r}_j^{1(k-1)U} &= \mathbf{s}_j^{(k-1)U}\mathbf{l}_j^{(k-1)} + \mathbf{t}_j^{(k-1)U}, \\ \mathbf{r}_j^{2(k-1)U} &= \mathbf{s}_j^{(k-1)U}\mathbf{u}_j^{(k-1)} + \mathbf{t}_j^{(k-1)U}, \\ j &= 1, 2, \cdots, n_{k-1}. \end{aligned} \tag{57}$$

They are the function values of the the function $\mathbf{s}_j^{(k-1)U}\mathbf{z}_j^{(k-1)} + t_j^{(k-1)U}$ at the two endpoints of the interval $[\mathbf{l}_j^{(k-1)}, \mathbf{u}_j^{(k-1)}]$. It's obvious that if we decrease $\mathbf{r}_j^{1(k-1)U}$ or $\mathbf{r}_j^{2(k-1)U}$, the upper bounding line specified by $\mathbf{r}_j^{1(k-1)U}$ and $\mathbf{r}_j^{2(k-1)U}$ will become tighter in the interval $[\mathbf{l}_j^{(k-1)}, \mathbf{u}_j^{(k-1)}]$. Similarly, we can define $\mathbf{r}_j^{1(k-1)L}$ and $\mathbf{r}_j^{2(k-1)L}$ as

$$\begin{aligned} \mathbf{r}_j^{1(k-1)L} &= \mathbf{s}_j^{(k-1)L}\mathbf{l}_j^{(k-1)} + \mathbf{t}_j^{(k-1)L}, \\ \mathbf{r}_j^{2(k-1)L} &= \mathbf{s}_j^{(k-1)L}\mathbf{u}_j^{(k-1)} + \mathbf{t}_j^{(k-1)L}, \\ j &= 1, 2, \cdots, n_{k-1}. \end{aligned} \tag{58}$$

If we increase $\mathbf{r}_j^{1(k-1)L}$ or $\mathbf{r}_j^{2(k-1)L}$, the lower bounding line specified by $\mathbf{r}_j^{1(k-1)L}$ and $\mathbf{r}_j^{2(k-1)L}$ will become tighter in the interval $[\mathbf{l}_j^{(k-1)}, \mathbf{u}_j^{(k-1)}]$.

According to the definition of $\mathbf{r}_j^{1(k-1)U}$ and $\mathbf{r}_j^{2(k-1)U}$, we have

$$\frac{\partial \mathbf{s}_j^{(k-1)U}}{\partial \mathbf{r}_j^{1(k-1)U}} = -\frac{1}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}}, \ \frac{\partial \mathbf{t}_j^{(k-1)U}}{\partial \mathbf{r}_j^{1(k-1)U}} = \frac{\mathbf{u}_j^{(k-1)}}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} \tag{59}$$

and

$$\frac{\partial \mathbf{s}_j^{(k-1)U}}{\partial \mathbf{r}_j^{2(k-1)U}} = \frac{1}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}}, \ \frac{\partial \mathbf{t}_j^{(k-1)U}}{\partial \mathbf{r}_j^{2(k-1)U}} = -\frac{\mathbf{l}_j^{(k-1)}}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}}. \tag{60}$$

Next, we can compute

$$\begin{aligned}
\frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{r}_j^{1(k-1)U}} &= \frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{s}_j^{(k-1)U}} \frac{\partial \mathbf{s}_j^{(k-1)U}}{\partial \mathbf{r}_j^{1(k-1)U}} + \frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{t}_j^{(k-1)U}} \frac{\partial \mathbf{t}_j^{(k-1)U}}{\partial \mathbf{r}_j^{1(k-1)U}} \\
&= -\frac{1}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} \frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{s}_j^{(k-1)U}} + \frac{\mathbf{u}_j^{(k-1)}}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} \frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{t}_j^{(k-1)U}} \\
&= -\frac{1}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} neg(\mathbf{W}_{ij}^{(k)}) [\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}]_j + \frac{\mathbf{u}_j^{(k-1)}}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} neg(\mathbf{W}_{ij}^{(k)}) \\
&= \frac{neg(\mathbf{W}_{ij}^{(k)})}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} (\mathbf{u}_j^{(k-1)} - [\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}]_j) \\
&\leq 0, \forall \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon).
\end{aligned} \tag{61}$$

The last line is because

$$\mathbf{u}_j^{(k-1)} = \max_{\mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon)} [\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}]_j \geq [\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}]_j, \forall \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon). \tag{62}$$

$$\mathbf{u}_j^{(k-1)} \geq \mathbf{l}_j^{(k-1)}, neg(\mathbf{W}_{ij}^{(k)}) \leq 0. \tag{63}$$

Similarly, we have

$$\begin{aligned}
\frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{r}_j^{2(k-1)U}} &= \frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{s}_j^{(k-1)U}} \frac{\partial \mathbf{s}_j^{(k-1)U}}{\partial \mathbf{r}_j^{2(k-1)U}} + \frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{t}_j^{(k-1)U}} \frac{\partial \mathbf{t}_j^{(k-1)U}}{\partial \mathbf{r}_j^{2(k-1)U}} \\
&= \frac{1}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} \frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{s}_j^{(k-1)U}} - \frac{\mathbf{l}_j^{(k-1)}}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} \frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{t}_j^{(k-1)U}} \\
&= \frac{1}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} neg(\mathbf{W}_{ij}^{(k)}) [\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}]_j - \frac{\mathbf{l}_j^{(k-1)}}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} neg(\mathbf{W}_{ij}^{(k)}) \\
&= \frac{neg(\mathbf{W}_{ij}^{(k)})}{\mathbf{u}_j^{(k-1)} - \mathbf{l}_j^{(k-1)}} ([\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}]_j - \mathbf{l}_j^{(k-1)}) \\
&\leq 0, \forall \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon).
\end{aligned} \tag{64}$$

The last line is because

$$\begin{aligned}
\mathbf{l}_j^{(k-1)} &= \min_{\mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon)} [\tilde{\mathbf{W}}^{(k-1)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)L}]_j \leq [\tilde{\mathbf{W}}^{(k-1)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)L}]_j \\
&\leq [\tilde{\mathbf{W}}^{(k-1)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k-1)U}]_j, \forall \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon).
\end{aligned} \tag{65}$$

$$\mathbf{u}_j^{(k-1)} \geq \mathbf{l}_j^{(k-1)}, neg(\mathbf{W}_{ij}^{(k)}) \leq 0. \tag{66}$$

Combining the above result, we have

$$\frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{r}_j^{1(k-1)U}} \leq 0, \frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{r}_j^{2(k-1)U}} \leq 0, \forall \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon). \tag{67}$$

In the same way, we can prove

$$\frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{r}_j^{1(k-1)L}} \geq 0, \frac{\partial \mathbf{f}_i^{(k)L}(\mathbf{x})}{\partial \mathbf{r}_j^{2(k-1)L}} \geq 0, \forall \mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon). \tag{68}$$

This means tighter bounding lines in the interval $[\mathbf{l}_j^{(k-1)}, \mathbf{u}_j^{(k-1)}]$ lead to larger lower bound $\mathbf{f}_i^{(k)L}(\mathbf{x})$. Note that

$$\mathbf{l}_i^{(k)} = \min_{\mathbf{x} \in \mathbb{B}_p(\mathbf{x}_0, \epsilon)} \mathbf{f}_i^{(k)L}(\mathbf{x}). \tag{69}$$

Therefore, tighter bounding lines in the interval $[\mathbf{l}_j^{(k-1)}, \mathbf{u}_j^{(k-1)}]$ also lead to larger closed-form lower bound $\mathbf{l}_i^{(k)}$. We can also prove tighter bounding lines in the interval $[\mathbf{l}_j^{(k-1)}, \mathbf{u}_j^{(k-1)}]$ also lead to smaller closed-form upper bound $\mathbf{u}_i^{(k)}$. In conclusion, tighter bounding lines in the $(k-1)$-th layer lead to tighter closed-form bounds in the k-th layer. Note that the above in the above proof we have assumed that $\mathbf{s}^{(k-1)L} \geq 0, \mathbf{s}^{(k-1)U} \geq 0$.

In fact, the closed-form bounds in the k-th layer not only depend on the bounding lines in the $(k-1)$-th layer, but also depend on bounding lines in the first $(k-2)$ layers. But we keep the bounding lines in the first $(k-2)$ layers the same when computing bounds for the k-th layer. This is the same self-consistency condition defined in the work [2]. Finally, we conclude that tighter bounding lines lead to tighter bounds in the LBP process under the self-consistency condition and bounding lines have non-negative slopes.

A.4 Details of IBP

Assume we know the lower and upper bounds of the pre-activation of the $(k-1)$-th layer:

$$\mathbf{l}^{(k-1)} \leq \mathbf{z}^{(k-1)} \leq \mathbf{u}^{(k-1)}. \tag{70}$$

Then bounds of $\mathbf{z}^{(k)}$ can be computed in the following way:

$$\begin{aligned} \mathbf{z}^{(k)} &= \mathbf{W}^{(k)}\mathbf{a}^{(k-1)} + \mathbf{b}^{(k)} \\ &\leq relu(\mathbf{W}^{(k)})\sigma(\mathbf{l}^{(k-1)}) + neg(\mathbf{W}^{(k)})\sigma(\mathbf{u}^{(k-1)}) + \mathbf{b}^{(k)}, \end{aligned} \tag{71}$$

and

$$\begin{aligned} \mathbf{z}^{(k)} &= \mathbf{W}^{(k)}\mathbf{a}^{(k-1)} + \mathbf{b}^{(k)} \\ &\geq neg(\mathbf{W}^{(k)})\sigma(\mathbf{l}^{(k-1)}) + relu(\mathbf{W}^{(k)})\sigma(\mathbf{u}^{(k-1)}) + \mathbf{b}^{(k)}. \end{aligned} \tag{72}$$

In the above deduction, we have assumed that σ is a monotonic increasing function.

IBP repeats the above procedure from the first layer and computes bounds layer by layer until the final output. Bounds of $\mathbf{z}^{(1)}$ are computed in the same way as shown in (26).

A.5 Prove IBP is a special case of LBP and CROWN.

A.5.1 Prove IBP is a special case of CROWN.

We prove IBP is a special case of CROWN where bounding lines are chosen as constants. We repeat the back-propagation process of CROWN in(20) here.

$$\begin{aligned}\hat{\mathbf{W}}^{(v)} &= [relu(\hat{\mathbf{W}}^{(v+1)}) * \mathbf{s}^{(v)L} + neg(\hat{\mathbf{W}}^{(v+1)}) * \mathbf{s}^{(v)U}]W^{(v)},\\ \hat{\mathbf{b}}^{(v)} &= relu(\hat{\mathbf{W}}^{(v+1)})(\mathbf{s}^{(v)L} * \mathbf{b}^{(v)} + \mathbf{t}^{(v)L}) + neg(\hat{\mathbf{W}}^{(v+1)})^\top(\mathbf{s}^{(v)U} * \mathbf{b}^{(v)} + \mathbf{t}^{(v)U}) + \hat{\mathbf{b}}^{(v+1)},\\ & v = k-1, k-2, \cdots, 1.\end{aligned} \tag{73}$$

If we choose constant bounding lines in the above iteration, namely,

$$\begin{aligned}\mathbf{s}^{(v)L} = 0, &\mathbf{t}^{(v)L} = \sigma(\mathbf{l}^{(v)}),\\ \mathbf{s}^{(v)U} = 0, &\mathbf{t}^{(v)U} = \sigma(\mathbf{u}^{(v)}),\\ & v = 1, 2, \cdots, k-1.\end{aligned} \tag{74}$$

We will have

$$\begin{aligned}\hat{\mathbf{W}}^{(k-1)} &= [relu(\hat{\mathbf{W}}^{(k)}) * \mathbf{s}^{(k-1)L} + neg(\hat{\mathbf{W}}^{(k)}) * \mathbf{s}^{(k-1)U}]W^{(k-1)} = 0,\\ \hat{\mathbf{b}}^{(k-1)} &= relu(\hat{\mathbf{W}}^{(k)})(\mathbf{s}^{(k-1)L} * \mathbf{b}^{(k-1)} + \mathbf{t}^{(k-1)L}) + neg(\hat{\mathbf{W}}^{(k)})^\top(\mathbf{s}^{(k-1)U} * \mathbf{b}^{(k-1)} + \mathbf{t}^{(k-1)U}) + \hat{\mathbf{b}}^{(k)}\\ &= relu(\hat{\mathbf{W}}^{(k)})\sigma(\mathbf{l}^{(k-1)}) + neg(\hat{\mathbf{W}}^{(k)})\sigma(\mathbf{u}^{(k-1)}) + \hat{\mathbf{b}}^{(k)}.\end{aligned} \tag{75}$$

$$\Downarrow$$

$$\hat{\mathbf{W}}^{(k-2)} = [relu(\hat{\mathbf{W}}^{(k-1)}) * \mathbf{s}^{(k-2)L} + neg(\hat{\mathbf{W}}^{(k-1)}) * \mathbf{s}^{(k-2)U}]W^{(k-2)} = 0, \tag{76}$$

$$\begin{aligned}\hat{\mathbf{b}}^{(k-2)} &= relu(\hat{\mathbf{W}}^{(k-1)})(\mathbf{s}^{(k-2)L} * \mathbf{b}^{(k-2)} + \mathbf{t}^{(k-2)L})\\ &\quad + neg(\hat{\mathbf{W}}^{(k-1)})^\top(\mathbf{s}^{(k-2)U} * \mathbf{b}^{(k-2)} + \mathbf{t}^{(k-2)U}) + \hat{\mathbf{b}}^{(k-1)}\\ &= \hat{\mathbf{b}}^{(k-1)}\\ &= relu(\hat{\mathbf{W}}^{(k)})\sigma(\mathbf{l}^{(k-1)}) + neg(\hat{\mathbf{W}}^{(k)})\sigma(\mathbf{u}^{(k-1)}) + \hat{\mathbf{b}}^{(k)}.\end{aligned} \tag{77}$$

$$\Downarrow$$
$$\vdots$$
$$\Downarrow$$

$$\hat{\mathbf{W}}^{(1)} = [relu(\hat{\mathbf{W}}^{(2)}) * \mathbf{s}^{(1)L} + neg(\hat{\mathbf{W}}^{(2)}) * \mathbf{s}^{(1)U}]W^{(1)} = 0, \tag{78}$$

$$\begin{aligned}\hat{\mathbf{b}}^{(1)} &= relu(\hat{\mathbf{W}}^{(2)})(\mathbf{s}^{(1)L} * \mathbf{b}^{(1)} + \mathbf{t}^{(1)L})\\ &\quad + neg(\hat{\mathbf{W}}^{(2)})^\top(\mathbf{s}^{(1)U} * \mathbf{b}^{(1)} + \mathbf{t}^{(1)U}) + \hat{\mathbf{b}}^{(2)}\\ &= \hat{\mathbf{b}}^{(2)}\\ &= relu(\hat{\mathbf{W}}^{(k)})\sigma(\mathbf{l}^{(k-1)}) + neg(\hat{\mathbf{W}}^{(k)})\sigma(\mathbf{u}^{(k-1)}) + \hat{\mathbf{b}}^{(k)}.\end{aligned} \tag{79}$$

Therefore, the lower bound is

$$\hat{\mathbf{W}}^{(1)}\mathbf{x}_0 + \hat{\mathbf{b}}^{(1)} - \epsilon||\hat{\mathbf{W}}^{(1)}||_q = relu(\hat{\mathbf{W}}^{(k)})\sigma(\mathbf{l}^{(k-1)}) + neg(\hat{\mathbf{W}}^{(k)})\sigma(\mathbf{u}^{(k-1)}) + \hat{\mathbf{b}}^{(k)}. \tag{80}$$

This lower bound is exactly the same one given by IBP in (71). Similarly, we can show the upper bound is also the same as the one given by IBP. Therefore, we conclude that CROWN degenerates to IBP when choosing constant bounding lines as shown in (74).

A.5.2 Prove IBP is a special case of LBP.

We prove IBP is a special case of LBP where bounding lines are chosen as constants. We repeat the LBP iteration in (33) here

$$\begin{aligned}\mathbf{z}^{(k)} &= \mathbf{W}^{(k)}\mathbf{a}^{(k-1)} + \mathbf{b}^{(k)} \\ &\geq \{[relu(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)L}]\tilde{\mathbf{W}}^{(k-1)L} + [neg(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)U}]\tilde{\mathbf{W}}^{(k-1)U}\}\mathbf{x} \\ &\quad + [relu(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)L}]\tilde{\mathbf{b}}^{(k-1)L} + [neg(\mathbf{W}^{(k)}) * \mathbf{s}^{(k-1)U}]\tilde{\mathbf{b}}^{(k-1)U} \\ &\quad + relu(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)L} + neg(\mathbf{W}^{(k)})\mathbf{t}^{(k-1)U} + \mathbf{b}^{(k)}.\end{aligned} \tag{81}$$

If we choose constant bounding lines in the above iteration, namely,

$$\begin{aligned}\mathbf{s}^{(k-1)L} = 0, &\mathbf{t}^{(k-1)L} = \sigma(\mathbf{l}^{(k-1)}), \\ \mathbf{s}^{(k-1)U} = 0, &\mathbf{t}^{(k-1)U} = \sigma(\mathbf{u}^{(k-1)}),\end{aligned} \tag{82}$$

we will have

$$\begin{aligned}\mathbf{z}^{(k)} &= \mathbf{W}^{(k)}\mathbf{a}^{(k-1)} + \mathbf{b}^{(k)} \\ &\geq relu(\mathbf{W}^{(k)})\sigma(\mathbf{l}^{(k-1)}) + neg(\mathbf{W}^{(k)})\sigma(\mathbf{u}^{(k-1)}) + \mathbf{b}^{(k)}.\end{aligned} \tag{83}$$

This is exactly the IBP iteration to compute lower bound in (71). In the same way, we can prove LBP gives the same upper bounds as IBP when choosing constant bounding lines as shown in (82).

A.6 Prove Theorem 2 in the main text.

A.6.1 Prove LBP is tighter than IBP.

In Section A.5, we have proven that IBP is a special case of LBP where bounding lines are chosen as constants. We have also proven tighter bounding lines lead to tighter bounds in Section A.3. It's obvious that bounding lines adopted by the tight strategy are always tighter than the constant bounding lines in any given interval. Therefore, we conclude that bounds computed by LBP are tighter than those computed by IBP if LBP adopts the tight strategy to choose bounding lines.

A.6.2 Prove CROWN-IBP is tighter than IBP.

To prove CROWN-IBP is tighter than IBP when CROWN-IBP adopts the tight strategy to choose bounding lines, we need to use a conclusion proven in Appendix A.6 in the work [2]. We repeat their conclusion in the following theorem.

We first introduce the necessary notations used in the work [2] to present the theorem. We use bracket in the superscript to group a set of variables, *e.g.*, we use $\mathbf{l}^{[m-1]}$ to denote the set $\{\mathbf{l}^{(1)}, \mathbf{l}^{(2)}, \cdots, \mathbf{l}^{(m-1)}\}$. In the CROWN computation process, to compute bounds for the k-th layer, we need to know bounds for the first $(k-1)$ layers and choose bounding lines for the first $(k-1)$ layers, $\{\mathbf{s}^{[k-1]U}, \mathbf{s}^{[k-1]L}, \mathbf{t}^{[k-1]U}, \mathbf{t}^{[k-1]L}\}$. We use

$$\boldsymbol{\gamma}^{kL}(\mathbf{s}^{[k-1]U}, \mathbf{s}^{[k-1]L}, \mathbf{t}^{[k-1]U}, \mathbf{t}^{[k-1]L}), \tag{84}$$

$$\boldsymbol{\gamma}^{kU}(\mathbf{s}^{[k-1]U}, \mathbf{s}^{[k-1]L}, \mathbf{t}^{[k-1]U}, \mathbf{t}^{[k-1]L}) \tag{85}$$

to denote the closed-form bounds computed by CROWN for the k-th layer using bounding lines

$$\{\mathbf{s}^{[k-1]U}, \mathbf{s}^{[k-1]L}, \mathbf{t}^{[k-1]U}, \mathbf{t}^{[k-1]L}\}$$

in the first $(k-1)$ layers. Remember that when computing bounds for the k-th layer, we need to know bounds of the first $(k-1)$ layers $\mathbf{l}^{[k-1]}$ and $\mathbf{u}^{[k-1]}$. However, $\mathbf{l}^{[k-1]}$ and $\mathbf{u}^{[k-1]}$ need not to be computed by CROWN. They only need to be valid bounds. They can be computed by IBP, LBP, CROWN, or any other robustness verification methods. Now we are ready to present the theorem.

Theorem 1 *Suppose the bounds of the pre-activations in the previous $(m-1)$ layers, $\mathbf{l}^{[m-1]}$ and $\mathbf{u}^{[m-1]}$, are known, and the robustness of a neural network is evaluated by CROWN on two trials with two different sets of bounding lines characterized by $\{\tilde{\mathbf{s}}^{[m-1]U}, \tilde{\mathbf{s}}^{[m-1]L}, \tilde{\mathbf{t}}^{[m-1]U}, \tilde{\mathbf{t}}^{[m-1]L}\}$ and $\{\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L}\}$. If $\{\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L}\}$, $\mathbf{l}^{[m-1]}$ and $\mathbf{u}^{[m-1]}$ satisfy Condition 1, then the closed-form bounds obtained via CROWN satisfy*

$$\begin{aligned}
&\boldsymbol{\gamma}^{mL}(\tilde{\mathbf{s}}^{[m-1]U}, \tilde{\mathbf{s}}^{[m-1]L}, \tilde{\mathbf{t}}^{[m-1]U}, \tilde{\mathbf{t}}^{[m-1]L}) \geq \\
&\boldsymbol{\gamma}^{mL}(\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L}), \\
&\boldsymbol{\gamma}^{mU}(\tilde{\mathbf{s}}^{[m-1]U}, \tilde{\mathbf{s}}^{[m-1]L}, \tilde{\mathbf{t}}^{[m-1]U}, \tilde{\mathbf{t}}^{[m-1]L}) \leq \\
&\boldsymbol{\gamma}^{mU}(\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L}),
\end{aligned}$$

when bounding lines determined by $\{\tilde{\mathbf{s}}^{[m-1]U}, \tilde{\mathbf{s}}^{[m-1]L}, \tilde{\mathbf{t}}^{[m-1]U}, \tilde{\mathbf{t}}^{[m-1]L}\}$ are the same as or tighter than those determined by $\{\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L}\}$.

Condition 1 *The bounding line parameters $\{\mathbf{s}^{[m-1]U}, \mathbf{s}^{[m-1]L}, \mathbf{t}^{[m-1]U}, \mathbf{t}^{[m-1]L}\}$ and bounds of the preactivations $\mathbf{l}^{[m-1]}, \mathbf{u}^{[m-1]}$ satisfy the following constraints:*

$$\begin{aligned}
&\boldsymbol{\gamma}^{kL}(\mathbf{s}^{[k-1]U}, \mathbf{s}^{[k-1]L}, \mathbf{t}^{[k-1]U}, \mathbf{t}^{[k-1]L}) \geq \mathbf{l}^{k}, \\
&\boldsymbol{\gamma}^{kU}(\mathbf{s}^{[k-1]U}, \mathbf{s}^{[k-1]L}, \mathbf{t}^{[k-1]U}, \mathbf{t}^{[k-1]L}) \leq \mathbf{u}^{k}, \\
&k = 1, 2, \cdots, m-1.
\end{aligned}$$

We refer readers to Appendix A.6 in the work [2] for proof of Theorem 1.

Recall that CROWN-IBP is a combination of IBP and CROWN: It uses IBP to compute bounds of the first $(m-1)$ layers, and then use CROWN to compute bounds of the last layer. We denote the bounds of the first $(m-1)$ layers computed by IBP as $\mathbf{l}^{[m-1]}$ and $\mathbf{u}^{[m-1]}$. We denote the bounding lines adopted by CROWN with the tight strategy in the first $(m-1)$ layers as $\{\tilde{\mathbf{s}}^{[m-1]U}, \tilde{\mathbf{s}}^{[m-1]L}, \tilde{\mathbf{t}}^{[m-1]U}, \tilde{\mathbf{t}}^{[m-1]L}\}$ and the closed-form bounds of the m-th layer computed by CROWN as $\boldsymbol{\gamma}^{mL}(\tilde{\mathbf{s}}^{[m-1]U}, \tilde{\mathbf{s}}^{[m-1]L}, \tilde{\mathbf{t}}^{[m-1]U}, \tilde{\mathbf{t}}^{[m-1]L})$ and $\boldsymbol{\gamma}^{mU}(\tilde{\mathbf{s}}^{[m-1]U}, \tilde{\mathbf{s}}^{[m-1]L}, \tilde{\mathbf{t}}^{[m-1]U}, \tilde{\mathbf{t}}^{[m-1]L})$. As for the bounds of the m-th layer computed by IBP, they can be seen as bounds computed by CROWN with constant bounding lines $\{\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L}\}$. Therefore, we can denote the bounds of the m-th layer computed by IBP as $\boldsymbol{\gamma}^{mL}(\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L})$ and $\boldsymbol{\gamma}^{mU}(\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L})$. We can verify that bounding lines $\{\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L}\}$ satisfy Condition 1. In fact, we should have

$$\begin{aligned}
&\boldsymbol{\gamma}^{kL}(\hat{\mathbf{s}}^{[k-1]U}, \hat{\mathbf{s}}^{[k-1]L}, \hat{\mathbf{t}}^{[k-1]U}, \hat{\mathbf{t}}^{[k-1]L}) = \mathbf{l}^{(k)},\\
&\boldsymbol{\gamma}^{kU}(\hat{\mathbf{s}}^{[k-1]U}, \hat{\mathbf{s}}^{[k-1]L}, \hat{\mathbf{t}}^{[k-1]U}, \hat{\mathbf{t}}^{[k-1]L}) = \mathbf{u}^{(k)},\\
&k = 1, 2, \cdots, m-1.
\end{aligned}$$

This is because CROWN degenerates to IBP when choosing constant bounding lines. Next, we know that bounding lines determined by $\{\tilde{\mathbf{s}}^{[m-1]U}, \tilde{\mathbf{s}}^{[m-1]L}, \tilde{\mathbf{t}}^{[m-1]U}, \tilde{\mathbf{t}}^{[m-1]L}\}$ are tighter than those determined by $\{\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L}\}$ in the intervals determined by $\mathbf{l}^{[m-1]}$ and $\mathbf{u}^{[m-1]}$. This is because the tight strategy always chooses tighter bounding lines than the constant strategy. Therefore, we should have

$$\begin{aligned}
&\boldsymbol{\gamma}^{mL}(\tilde{\mathbf{s}}^{[m-1]U}, \tilde{\mathbf{s}}^{[m-1]L}, \tilde{\mathbf{t}}^{[m-1]U}, \tilde{\mathbf{t}}^{[m-1]L}) \geq\\
&\boldsymbol{\gamma}^{mL}(\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L}),\\
&\boldsymbol{\gamma}^{mU}(\tilde{\mathbf{s}}^{[m-1]U}, \tilde{\mathbf{s}}^{[m-1]L}, \tilde{\mathbf{t}}^{[m-1]U}, \tilde{\mathbf{t}}^{[m-1]L}) \leq\\
&\boldsymbol{\gamma}^{mU}(\hat{\mathbf{s}}^{[m-1]U}, \hat{\mathbf{s}}^{[m-1]L}, \hat{\mathbf{t}}^{[m-1]U}, \hat{\mathbf{t}}^{[m-1]L})
\end{aligned}$$

as guaranteed by Theorem 1. This completes the proof that CROWN-IBP gives tighter bounds than IBP for the m-th layer if CROWN-IBP adopts the tight strategy to choose bounding lines.

A.6.3 Prove CROWN-LBP is tighter than LBP.

In this section, we prove that CROWN-LBP is tighter than LBP if they choose the same bounding lines with non-negative slopes. Recall that CROWN-LBP is a combination of CROWN and LBP: It uses LBP to compute bounds for the first $(m-1)$ layers, and then uses CROWN to compute bounds for the m-th layer. Assume the linear bounds of the first $(m-1)$ layers computed by LBP are

$$\tilde{\mathbf{W}}^{(k)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k)L} \leq \mathbf{z}^{(k)} \leq \tilde{\mathbf{W}}^{(k)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k)U}, k = 1, 2, \cdots, m-1, \tag{86}$$

and

$$\boldsymbol{\Lambda}^{(k)L}\mathbf{x} + \boldsymbol{\beta}^{(k)L} \leq \mathbf{a}^{(k)} \leq \boldsymbol{\Lambda}^{(k)U}\mathbf{x} + \boldsymbol{\beta}^{(k)U}, k = 1, 2, \cdots, m-1. \tag{87}$$

And the closed-form bounds computed by LBP for the first $(m-1)$ layers are

$$\mathbf{l}^{(k)} \leq \mathbf{z}^{(k)} \leq \mathbf{u}^{(k)}, k = 1, 2, \cdots, m-1. \tag{88}$$

Next we review the back-propagation process of CROWN from the last layer shown in (18). We set $\mathbf{W}^{obj} = \mathbf{I}, \mathbf{b}^{obj} = 0, k = m$ in (18) and obtain

$$\begin{aligned}
&\mathbf{z}^{(m)} = \mathbf{W}^{(m)}\mathbf{a}^{(m-1)} + \mathbf{b}^{(m)} \geq \boldsymbol{\Omega}^{(m,m-2)L}\mathbf{a}^{(m-2)} + \boldsymbol{\delta}^{(m,m-2)L}\\
&\geq \boldsymbol{\Omega}^{(m,m-3)L}\mathbf{a}^{(m-3)} + \boldsymbol{\delta}^{(m,m-3)L} \geq \cdots \geq \boldsymbol{\Omega}^{(m,0)L}\mathbf{a}^{(0)} + \boldsymbol{\delta}^{(m,0)L} = \boldsymbol{\Omega}^{(m,0)L}\mathbf{x} + \boldsymbol{\delta}^{(m,0)L}.
\end{aligned} \tag{89}$$

Note that we assume that CROWN chooses the same bounding lines as LBP in this back-propagation process. $\boldsymbol{\Omega}^{(m,0)L}\mathbf{x} + \boldsymbol{\delta}^{(m,0)L}$ will be the linear lower bound of the m-th layer computed by CROWN.

On the other hand, the linear lower bound of the m-th layer given by LBP is constructed by inserting the linear bounds of $\mathbf{a}^{(m-1)}$ directly to $\mathbf{W}^{(m)}\mathbf{a}^{(m-1)} + \mathbf{b}^{(m)}$. This is because we have proven in Section A.3.1 that when

choosing bounding lines with non-negative slopes, LBP is equivalent to the forward mode in the work [3], which builds bounds through $\mathbf{a}^{(k)}, k = 1, 2, \cdots, m-1$. We denote this bound as $\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)})$. $\mathbf{x}$ in $\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)})$ means it is a linear function of $\mathbf{x}$, and $\mathbf{a}^{(m-1)}$ in it means it is a linear lower bound obtained by inserting the linear bounds of $\mathbf{a}^{(m-1)}$. Specifically, we have

$$\begin{aligned}\mathbf{z}^{(m)} &= \mathbf{W}^{(m)}\mathbf{a}^{(m-1)} + \mathbf{b}^{(m)} \\ &\geq relu(\mathbf{W}^{(m)})[\boldsymbol{\Lambda}^{(m-1)L}\mathbf{x} + \boldsymbol{\beta}^{(m-1)L}] + neg(\mathbf{W}^{(m)})[\boldsymbol{\Lambda}^{(m-1)U}\mathbf{x} + \boldsymbol{\beta}^{(m-1)U}] + \mathbf{b}^{(m)} \\ &= \mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)}).\end{aligned} \tag{90}$$

We can prove

$$\boldsymbol{\Omega}^{(m,0)L}\mathbf{x} + \boldsymbol{\delta}^{(m,0)L} \geq \mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)}) \tag{91}$$

in order to prove the lower bound computed by CROWN-LBP is tighter than that computed by LBP.

We first prove

$$\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-2)}) \geq \mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)}), \tag{92}$$

where $\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-2)})$ is the linear lower bound obtained by inserting linear bounds of $\mathbf{a}^{(m-2)}$ to $\boldsymbol{\Omega}^{(m,m-2)L}\mathbf{a}^{(m-2)} + \boldsymbol{\delta}^{(m,m-2)L}$, namely,

$$\begin{aligned}\mathbf{z}^{(m)} &= \mathbf{W}^{(m)}\mathbf{a}^{(m-1)} + \mathbf{b}^{(m)} \geq \boldsymbol{\Omega}^{(m,m-2)L}\mathbf{a}^{(m-2)} + \boldsymbol{\delta}^{(m,m-2)L} \\ &\geq relu(\boldsymbol{\Omega}^{(m,m-2)L})[\boldsymbol{\Lambda}^{(m-2)L}\mathbf{x} + \boldsymbol{\beta}^{(m-2)L}] + neg(\boldsymbol{\Omega}^{(m,m-2)L})[\boldsymbol{\Lambda}^{(m-2)U}\mathbf{x} + \boldsymbol{\beta}^{(m-2)U}] + \boldsymbol{\delta}^{(m,m-2)L} \\ &= \mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-2)}).\end{aligned} \tag{93}$$

We define the following two shorthands for the two linear functions $\boldsymbol{\Lambda}^{(m-2)L}\mathbf{x}+\boldsymbol{\beta}^{(m-2)L}$ and $\boldsymbol{\Lambda}^{(m-2)U}\mathbf{x}+\boldsymbol{\beta}^{(m-2)U}$:

$$\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) = \boldsymbol{\Lambda}^{(m-2)L}\mathbf{x} + \boldsymbol{\beta}^{(m-2)L}, \tag{94}$$

$$\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) = \boldsymbol{\Lambda}^{(m-2)U}\mathbf{x} + \boldsymbol{\beta}^{(m-2)U}. \tag{95}$$

From (16), we know

$$\boldsymbol{\Omega}^{(m,m-2)L} = \mathbf{W}^{(m,m-1)L}\mathbf{W}^{(m-1)}, \tag{96}$$

$$\boldsymbol{\delta}^{(m,m-2)L} = \mathbf{W}^{(m,m-1)L}\mathbf{b}^{(m-1)} + \mathbf{b}^{(m,m-1)L}, \tag{97}$$

where

$$\mathbf{W}^{(m,m-1)L} = relu(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)L} + neg(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)U}, \tag{98}$$

$$\mathbf{b}^{(m,m-1)L} = relu(\mathbf{W}^{(m)})\mathbf{t}^{(m-1)L} + neg(\mathbf{W}^{(m)})\mathbf{t}^{(m-1)U} + \mathbf{b}^{(m)}. \tag{99}$$

Therefore

$$\begin{aligned}&\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-2)}) \\ =&relu(\boldsymbol{\Omega}^{(m,m-2)L})[\boldsymbol{\Lambda}^{(m-2)L}\mathbf{x} + \boldsymbol{\beta}^{(m-2)L}] + neg(\boldsymbol{\Omega}^{(m,m-2)L})[\boldsymbol{\Lambda}^{(m-2)U}\mathbf{x} + \boldsymbol{\beta}^{(m-2)U}] + \boldsymbol{\delta}^{(m,m-2)L} \\ =&relu(\mathbf{W}^{(m,m-1)L}\mathbf{W}^{(m-1)})\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) + neg(\mathbf{W}^{(m,m-1)L}\mathbf{W}^{(m-1)})\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) \\ &+ \mathbf{W}^{(m,m-1)L}\mathbf{b}^{(m-1)} + \mathbf{b}^{(m,m-1)L}.\end{aligned} \tag{100}$$

Next, we study $\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)})$. We know

$$\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)}) = relu(\mathbf{W}^{(m)})[\boldsymbol{\Lambda}^{(m-1)L}\mathbf{x} + \boldsymbol{\beta}^{(m-1)L}] + neg(\mathbf{W}^{(m)})[\boldsymbol{\Lambda}^{(m-1)U}\mathbf{x} + \boldsymbol{\beta}^{(m-1)U}] + \mathbf{b}^{(m)} \tag{101}$$

From (40) and (42), we know

$$\begin{aligned}\boldsymbol{\Lambda}^{(m-1)L} &= \mathbf{s}^{(m-1)L} * \tilde{\mathbf{W}}^{(m-1)L}, \\ \boldsymbol{\beta}^{(m-1)L} &= \mathbf{s}^{(m-1)L} * \tilde{\mathbf{b}}^{(m-1)L} + \mathbf{t}^{(m-1)L},\end{aligned} \tag{102}$$

and

$$\begin{aligned}\boldsymbol{\Lambda}^{(m-1)U} &= \mathbf{s}^{(m-1)U} * \tilde{\mathbf{W}}^{(m-1)U},\\ \boldsymbol{\beta}^{(m-1)U} &= \mathbf{s}^{(m-1)U} * \tilde{\mathbf{b}}^{(m-1)U} + \mathbf{t}^{(m-1)U}.\end{aligned} \tag{103}$$

From (45) and (47), we know

$$\begin{aligned}\tilde{\mathbf{W}}^{(m-1)L} &= relu(\mathbf{W}^{(m-1)})\boldsymbol{\Lambda}^{(m-2)L} + neg(\mathbf{W}^{(m-1)})\boldsymbol{\Lambda}^{(m-2)U},\\ \tilde{\mathbf{b}}^{(m-1)L} &= relu(\mathbf{W}^{(m-1)})\boldsymbol{\beta}^{(m-2)L} + neg(\mathbf{W}^{(m-1)})\boldsymbol{\beta}^{(m-2)U} + \mathbf{b}^{(m-1)},\end{aligned} \tag{104}$$

and

$$\begin{aligned}\tilde{\mathbf{W}}^{(m-1)U} &= relu(\mathbf{W}^{(m-1)})\boldsymbol{\Lambda}^{(m-2)U} + neg(\mathbf{W}^{(m-1)})\boldsymbol{\Lambda}^{(m-2)L},\\ \tilde{\mathbf{b}}^{(m-1)U} &= relu(\mathbf{W}^{(m-1)})\boldsymbol{\beta}^{(m-2)U} + neg(\mathbf{W}^{(m-1)})\boldsymbol{\beta}^{(m-2)L} + \mathbf{b}^{(m-1)}.\end{aligned} \tag{105}$$

Therefore, we have

$$\begin{aligned}&\boldsymbol{\Lambda}^{(m-1)L}\mathbf{x} + \boldsymbol{\beta}^{(m-1)L}\\ =&(\mathbf{s}^{(m-1)L} * \tilde{\mathbf{W}}^{(m-1)L})\mathbf{x} + \mathbf{s}^{(m-1)L} * \tilde{\mathbf{b}}^{(m-1)L} + \mathbf{t}^{(m-1)L}\\ =&\{\mathbf{s}^{(m-1)L} * [relu(\mathbf{W}^{(m-1)})\boldsymbol{\Lambda}^{(m-2)L} + neg(\mathbf{W}^{(m-1)})\boldsymbol{\Lambda}^{(m-2)U}]\}\mathbf{x}\\ &+ \mathbf{s}^{(m-1)L} * [relu(\mathbf{W}^{(m-1)})\boldsymbol{\beta}^{(m-2)L} + neg(\mathbf{W}^{(m-1)})\boldsymbol{\beta}^{(m-2)U} + \mathbf{b}^{(m-1)}] + \mathbf{t}^{(m-1)L}\\ =&[\mathbf{s}^{(m-1)L} * relu(\mathbf{W}^{(m-1)})][\boldsymbol{\Lambda}^{(m-2)L}\mathbf{x} + \boldsymbol{\beta}^{(m-2)L}] + [\mathbf{s}^{(m-1)L} * neg(\mathbf{W}^{(m-1)})][\boldsymbol{\Lambda}^{(m-2)U}\mathbf{x} + \boldsymbol{\beta}^{(m-2)U}]\\ &+ \mathbf{s}^{(m-1)L} * \mathbf{b}^{(m-1)} + \mathbf{t}^{(m-1)L}\\ =&[\mathbf{s}^{(m-1)L} * relu(\mathbf{W}^{(m-1)})]\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) + [\mathbf{s}^{(m-1)L} * neg(\mathbf{W}^{(m-1)})]\boldsymbol{\psi}^{(m-2)U}(\mathbf{x})\\ &+ \mathbf{s}^{(m-1)L} * \mathbf{b}^{(m-1)} + \mathbf{t}^{(m-1)L},\end{aligned} \tag{106}$$

and

$$\begin{aligned}&\boldsymbol{\Lambda}^{(m-1)U}\mathbf{x} + \boldsymbol{\beta}^{(m-1)U}\\ =&(\mathbf{s}^{(m-1)U} * \tilde{\mathbf{W}}^{(m-1)U})\mathbf{x} + \mathbf{s}^{(m-1)U} * \tilde{\mathbf{b}}^{(m-1)U} + \mathbf{t}^{(m-1)U}\\ =&\{\mathbf{s}^{(m-1)U} * [relu(\mathbf{W}^{(m-1)})\boldsymbol{\Lambda}^{(m-2)U} + neg(\mathbf{W}^{(m-1)})\boldsymbol{\Lambda}^{(m-2)L}]\}\mathbf{x}\\ &+ \mathbf{s}^{(m-1)U} * [relu(\mathbf{W}^{(m-1)})\boldsymbol{\beta}^{(m-2)U} + neg(\mathbf{W}^{(m-1)})\boldsymbol{\beta}^{(m-2)L} + \mathbf{b}^{(m-1)}] + \mathbf{t}^{(m-1)U}\\ =&[\mathbf{s}^{(m-1)U} * relu(\mathbf{W}^{(m-1)})][\boldsymbol{\Lambda}^{(m-2)U}\mathbf{x} + \boldsymbol{\beta}^{(m-2)U}] + [\mathbf{s}^{(m-1)U} * neg(\mathbf{W}^{(m-1)})][\boldsymbol{\Lambda}^{(m-2)L}\mathbf{x} + \boldsymbol{\beta}^{(m-2)L}]\\ &+ \mathbf{s}^{(m-1)U} * \mathbf{b}^{(m-1)} + \mathbf{t}^{(m-1)U}\\ =&[\mathbf{s}^{(m-1)U} * relu(\mathbf{W}^{(m-1)})]\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) + [\mathbf{s}^{(m-1)U} * neg(\mathbf{W}^{(m-1)})]\boldsymbol{\psi}^{(m-2)L}(\mathbf{x})\\ &+ \mathbf{s}^{(m-1)U} * \mathbf{b}^{(m-1)} + \mathbf{t}^{(m-1)U}.\end{aligned} \tag{107}$$

Finally, we have

$$
\begin{aligned}
&\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)}) \\
=&relu(\mathbf{W}^{(m)})[\mathbf{\Lambda}^{(m-1)L}\mathbf{x}+\boldsymbol{\beta}^{(m-1)L}]+neg(\mathbf{W}^{(m)})[\mathbf{\Lambda}^{(m-1)U}\mathbf{x}+\boldsymbol{\beta}^{(m-1)U}]+\mathbf{b}^{(m)} \\
=&relu(\mathbf{W}^{(m)})\{[\mathbf{s}^{(m-1)L}*relu(\mathbf{W}^{(m-1)})]\boldsymbol{\psi}^{(m-2)L}(\mathbf{x})+[\mathbf{s}^{(m-1)L}*neg(\mathbf{W}^{(m-1)})]\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) \\
&+\mathbf{s}^{(m-1)L}*\mathbf{b}^{(m-1)}+\mathbf{t}^{(m-1)L}\} \\
&+neg(\mathbf{W}^{(m)})\{[\mathbf{s}^{(m-1)U}*relu(\mathbf{W}^{(m-1)})]\boldsymbol{\psi}^{(m-2)U}(\mathbf{x})+[\mathbf{s}^{(m-1)U}*neg(\mathbf{W}^{(m-1)})]\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) \\
&+\mathbf{s}^{(m-1)U}*\mathbf{b}^{(m-1)}+\mathbf{t}^{(m-1)U}\}+\mathbf{b}^{(m)} \\
=&\{relu(\mathbf{W}^{(m)})[\mathbf{s}^{(m-1)L}*relu(\mathbf{W}^{(m-1)})]+neg(\mathbf{W}^{(m)})[\mathbf{s}^{(m-1)U}*neg(\mathbf{W}^{(m-1)})]\}\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) \\
&+\{relu(\mathbf{W}^{(m)})[\mathbf{s}^{(m-1)L}*neg(\mathbf{W}^{(m-1)})]+neg(\mathbf{W}^{(m)})[\mathbf{s}^{(m-1)U}*relu(\mathbf{W}^{(m-1)})]\}\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) \\
&+[relu(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)L}+neg(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)U}]\mathbf{b}^{(m-1)} \\
&+relu(\mathbf{W}^{(m)})\mathbf{t}^{(m-1)L}+neg(\mathbf{W}^{(m)})\mathbf{t}^{(m-1)U}+\mathbf{b}^{(m)} \\
=&\{relu(\mathbf{W}^{(m)})[\mathbf{s}^{(m-1)L}*relu(\mathbf{W}^{(m-1)})]+neg(\mathbf{W}^{(m)})[\mathbf{s}^{(m-1)U}*neg(\mathbf{W}^{(m-1)})]\}\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) \\
&+\{relu(\mathbf{W}^{(m)})[\mathbf{s}^{(m-1)L}*neg(\mathbf{W}^{(m-1)})]+neg(\mathbf{W}^{(m)})[\mathbf{s}^{(m-1)U}*relu(\mathbf{W}^{(m-1)})]\}\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) \\
&+\mathbf{W}^{(m,m-1)L}\mathbf{b}^{(m-1)}+\mathbf{b}^{(m,m-1)L} \\
=&\{[relu(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)L}]relu(\mathbf{W}^{(m-1)})+[neg(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)U}]neg(\mathbf{W}^{(m-1)})\}\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) \\
&+\{[relu(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)L}]neg(\mathbf{W}^{(m-1)})+[neg(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)U}]relu(\mathbf{W}^{(m-1)})\}\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) \\
&+\mathbf{W}^{(m,m-1)L}\mathbf{b}^{(m-1)}+\mathbf{b}^{(m,m-1)L}
\end{aligned}
\tag{108}
$$

We define shorthands for the coefficients before $\boldsymbol{\psi}^{(m-2)L}(\mathbf{x})$ and $\boldsymbol{\psi}^{(m-2)U}(\mathbf{x})$:

$$
① = [relu(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)L}]relu(\mathbf{W}^{(m-1)})+[neg(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)U}]neg(\mathbf{W}^{(m-1)}), \tag{109}
$$

$$
② = [relu(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)L}]neg(\mathbf{W}^{(m-1)})+[neg(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)U}]relu(\mathbf{W}^{(m-1)}). \tag{110}
$$

Recall that

$$
\begin{aligned}
&\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-2)}) \\
=&relu(\mathbf{W}^{(m,m-1)L}\mathbf{W}^{(m-1)})\boldsymbol{\psi}^{(m-2)L}(\mathbf{x})+neg(\mathbf{W}^{(m,m-1)L}\mathbf{W}^{(m-1)})\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) \\
&+\mathbf{W}^{(m,m-1)L}\mathbf{b}^{(m-1)}+\mathbf{b}^{(m,m-1)L} \\
=&relu\{[relu(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)L}+neg(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)U}]\mathbf{W}^{(m-1)}\}\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) \\
&+neg\{[relu(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)L}+neg(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)U}]\mathbf{W}^{(m-1)}\}\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) \\
&+\mathbf{W}^{(m,m-1)L}\mathbf{b}^{(m-1)}+\mathbf{b}^{(m,m-1)L}.
\end{aligned}
\tag{111}
$$

We define shorthands for the coefficients before $\boldsymbol{\psi}^{(m-2)L}(\mathbf{x})$ and $\boldsymbol{\psi}^{(m-2)U}(\mathbf{x})$:

$$
③ = relu\{[relu(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)L}+neg(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)U}]\mathbf{W}^{(m-1)}\}, \tag{112}
$$

$$
④ = neg\{[relu(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)L}+neg(\mathbf{W}^{(m)})*\mathbf{s}^{(m-1)U}]\mathbf{W}^{(m-1)}\}. \tag{113}
$$

To prove $\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-2)}) \geq \mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)})$, we only need to prove

$$
③\boldsymbol{\psi}^{(m-2)L}(\mathbf{x})+④\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) \geq ①\boldsymbol{\psi}^{(m-2)L}(\mathbf{x})+②\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}). \tag{114}
$$

First, we observe

$$
① + ② = ③ + ④. \tag{115}
$$

This is because

$$\begin{aligned}
&[relu(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)L}]relu(\mathbf{W}^{(m-1)}) + [neg(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)U}]neg(\mathbf{W}^{(m-1)}) \\
&+ [relu(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)L}]neg(\mathbf{W}^{(m-1)}) + [neg(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)U}]relu(\mathbf{W}^{(m-1)}) \\
=&[relu(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)L}]\mathbf{W}^{(m-1)} + [neg(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)U}]\mathbf{W}^{(m-1)} \\
=&[relu(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)L} + neg(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)U}]\mathbf{W}^{(m-1)} \\
=&relu\{[relu(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)L} + neg(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)U}]\mathbf{W}^{(m-1)}\} \\
&+ neg\{[relu(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)L} + neg(\mathbf{W}^{(m)}) * \mathbf{s}^{(m-1)U}]\mathbf{W}^{(m-1)}\}
\end{aligned} \tag{116}$$

Next, it's obvious that

$$① \geq ③, ② \leq ④. \tag{117}$$

Combining this with the fact that

$$\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) \leq a^{(m-2)} \leq \boldsymbol{\psi}^{(m-2)U}(\mathbf{x}), \tag{118}$$

we conclude

$$③\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) + ④\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}) \geq ①\boldsymbol{\psi}^{(m-2)L}(\mathbf{x}) + ②\boldsymbol{\psi}^{(m-2)U}(\mathbf{x}). \tag{119}$$

Now we have proven $\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-2)}) \geq \mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)})$. In the same manner, we can prove $\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-3)}) \geq \mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-2)})$. We can repeat this procedure and back-propagate to the input $\mathbf{a}^{(0)}$:

$$\mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-1)}) \leq \mathbf{g}(\mathbf{x}|\mathbf{a}^{(m-2)}) \leq \cdots \leq \mathbf{g}(\mathbf{x}|\mathbf{a}^{(0)}). \tag{120}$$

where

$$\begin{aligned}
\mathbf{g}(\mathbf{x}|\mathbf{a}^{(k)}) = &relu(\boldsymbol{\Omega}^{(m,k)L})[\boldsymbol{\Lambda}^{(k)L}\mathbf{x} + \boldsymbol{\beta}^{(k)L}] + neg(\boldsymbol{\Omega}^{(m,k)L})[\boldsymbol{\Lambda}^{(k)U}\mathbf{x} + \boldsymbol{\beta}^{(k)U}] + \boldsymbol{\delta}^{(m,k)L}, \\
&k = m-1, m-2, \cdots, 0.
\end{aligned} \tag{121}$$

Note that the last term of (120) is exactly the lower bound of the m-th layer computed by CROWN-LBP. Recall the first term is the lower bound of the m-th layer computed by LBP. Therefore, we conclude the lower bound of the last layer computed by CROWN-LBP is tighter than that computed by LBP. In the same way, we can also prove the upper bound of the last layer computed by CROWN-LBP is tighter than that computed by LBP.

A.6.4 Prove CROWN is tighter than CROWN-LBP.

We prove that CROWN is tighter than CROWN-LBP when both of them adopt the tight strategy to choose bounding lines. Recall that in CROWN-LBP, we use LBP to compute bounds for the first $(m-1)$ layers and then use CROWN to compute bounds for the m-th layer. Assume the linear bounds computed by CROWN-LBP are

$$\tilde{\mathbf{W}}^{(k)L}\mathbf{x} + \tilde{\mathbf{b}}^{(k)L} \leq \mathbf{z}^{(k)} \leq \tilde{\mathbf{W}}^{(k)U}\mathbf{x} + \tilde{\mathbf{b}}^{(k)U}, k = 1, 2, \cdots, m, \tag{122}$$

and the closed-form bounds computed by CROWN-LBP are

$$\mathbf{l}_{\text{LBP}}^{(k)} \leq \mathbf{z}^{(k)} \leq \mathbf{u}_{\text{LBP}}^{(k)}, k = 1, 2, \cdots, m-1. \tag{123}$$

$$\mathbf{l}_{\text{C.LBP}}^{(m)} \leq \mathbf{z}^{(m)} \leq \mathbf{u}_{\text{C.LBP}}^{(m)} \tag{124}$$

Assume the linear bounds computed by CROWN are

$$\hat{\mathbf{W}}^{(k)L}\mathbf{x} + \hat{\mathbf{b}}^{(k)L} \leq \mathbf{z}^{(k)} \leq \hat{\mathbf{W}}^{(k)U}\mathbf{x} + \hat{\mathbf{b}}^{(k)U}, k = 1, 2, \cdots, m, \tag{125}$$

and the closed-form bounds computed by CROWN are

$$\mathbf{l}_{\text{C}}^{(k)} \leq \mathbf{z}^{(k)} \leq \mathbf{u}_{\text{C}}^{(k)}, k = 1, 2, \cdots, m. \tag{126}$$

For the first layer, we know

$$\tilde{\mathbf{W}}^{(1)L} = \hat{\mathbf{W}}^{(1)L}, \tilde{\mathbf{b}}^{(1)L} = \hat{\mathbf{b}}^{(1)L}, \tag{127}$$

$$\mathbf{l}_{\text{LBP}}^{(1)} = \mathbf{l}_{\text{C}}^{(1)}, \mathbf{u}_{\text{LBP}}^{(1)} = \mathbf{u}_{\text{C}}^{(1)}, \tag{128}$$

according to the definition of LBP and CROWN.

For the second layer, the bounds of CROWN can be seen as bounds computed by CROWN-LBP. This is because the closed-form bounds of the first layer computed by CROWN are the same as those computed by LBP. Therefore, we can see the bounds of the second layer computed by CROWN as the bounds computed by CROWN-LBP: We use LBP to compute bounds for the first layer and then use CROWN to compute bounds for the second layer. We already prove CROWN-LBP is tighter than LBP in Section A.6.3. Therefore, we should have

$$\mathbf{l}_{\text{LBP}}^{(2)} \leq \mathbf{l}_{\text{C}}^{(2)}, \mathbf{u}_{\text{C}}^{(2)} \leq \mathbf{u}_{\text{LBP}}^{(2)}. \tag{129}$$

Next, we consider the third layer. Assume the bounds of the third layer computed by CROWN-LBP is

$$\mathbf{l}_{\text{C.LBP}}^{(3)}, \mathbf{u}_{\text{C.LBP}}^{(3)}. \tag{130}$$

We know

$$\mathbf{l}_{\text{LBP}}^{(3)} \leq \mathbf{l}_{\text{C.LBP}}^{(3)}, \mathbf{u}_{\text{C.LBP}}^{(3)} \leq \mathbf{u}_{\text{LBP}}^{(3)}, \tag{131}$$

because CROWN-LBP is tighter than LBP. The bounds of the third layer computed by CROWN-LBP can be seen as the bounds computed by CROWN when we know the bounds of the first 2 layers are $\mathbf{l}_{\text{LBP}}^{(1)}, \mathbf{u}_{\text{LBP}}^{(1)}; \mathbf{l}_{\text{LBP}}^{(2)}, \mathbf{u}_{\text{LBP}}^{(2)}$. The bounds of the third layer computed by CROWN can be seen as the bounds computed by CROWN when we know the bounds of the first 2 layers are $\mathbf{l}_{\text{C}}^{(1)}, \mathbf{u}_{\text{C}}^{(1)}; \mathbf{l}_{\text{C}}^{(2)}, \mathbf{u}_{\text{C}}^{(2)}$. In this case, if both CROWN-LBP and CROWN use the tight strategy to choose bounding lines, the bounding lines in the first 2 layers chosen by CROWN will be tighter than those chosen by CROWN-LBP in the intervals determined by $\mathbf{l}_{\text{LBP}}^{(1)}, \mathbf{u}_{\text{LBP}}^{(1)}; \mathbf{l}_{\text{LBP}}^{(2)}, \mathbf{u}_{\text{LBP}}^{(2)}$, because $\mathbf{l}_{\text{C}}^{(1)}, \mathbf{u}_{\text{C}}^{(1)}; \mathbf{l}_{\text{C}}^{(2)}, \mathbf{u}_{\text{C}}^{(2)}$ are tighter than $\mathbf{l}_{\text{LBP}}^{(1)}, \mathbf{u}_{\text{LBP}}^{(1)}; \mathbf{l}_{\text{LBP}}^{(2)}, \mathbf{u}_{\text{LBP}}^{(2)}$. Therefore, using Theorem 1 (set $m = 3$), we conclude

$$\mathbf{l}_{\text{C.LBP}}^{(3)} \leq \mathbf{l}_{\text{C}}^{(3)}, \mathbf{u}_{\text{C}}^{(3)} \leq \mathbf{u}_{\text{C.LBP}}^{(3)}. \tag{132}$$

Combining this with (131), we conclude

$$\mathbf{l}_{\text{LBP}}^{(3)} \leq \mathbf{l}_{\text{C}}^{(3)}, \mathbf{u}_{\text{C}}^{(3)} \leq \mathbf{u}_{\text{LBP}}^{(3)}. \tag{133}$$

For the same argument in the third layer, we can prove

$$\mathbf{l}_{\text{LBP}}^{(4)} \leq \mathbf{l}_{\text{C}}^{(4)}, \mathbf{u}_{\text{C}}^{(4)} \leq \mathbf{u}_{\text{LBP}}^{(4)}. \tag{134}$$

In fact, we can prove

$$\mathbf{l}_{\text{LBP}}^{(k)} \leq \mathbf{l}_{\text{C}}^{(k)}, \mathbf{u}_{\text{C}}^{(k)} \leq \mathbf{u}_{\text{LBP}}^{(k)}, k = 1, 2, \cdots, m-1. \tag{135}$$

Finally, following the same argument in the third layer when we prove

$$\mathbf{l}_{\text{C.LBP}}^{(3)} \leq \mathbf{l}_{\text{C}}^{(3)}, \mathbf{u}_{\text{C}}^{(3)} \leq \mathbf{u}_{\text{C.LBP}}^{(3)}, \tag{136}$$

we can prove

$$\mathbf{l}_{\text{C.LBP}}^{(m)} \leq \mathbf{l}_{\text{C}}^{(m)}, \mathbf{u}_{\text{C}}^{(m)} \leq \mathbf{u}_{\text{C.LBP}}^{(m)}. \tag{137}$$

A.7 Bounding Lines

A.7.1 Bounding Lines for LeakyReLU and ReLU

Assume that the slope of the left part of LeakyReLU is η, the input is z and the output is a. The input range of z is $[l, u]$.

$$a = \begin{cases} \eta z, \text{if } z < 0, \\ z, \text{else.} \end{cases} \tag{138}$$

The tight strategy and the adaptive strategy to choose bounding lines for LeakyReLU are presented in Table 1. In the constant strategy, the upper bounding line is chosen as $0z + \text{LeakyReLU}(u)$ and the lower bounding line is chosen as $0z + \text{LeakyReLU}(l)$. To choose bounding lines for ReLU, we only need to set $\eta = 0$.

Table 1: The tight strategy and the adaptive strategy to choose bounding lines for LeakyReLU.

The Tight Strategy			
Neuron status	Dead ($l \leq u \leq 0$)	Unstable ($l < 0 < u$)	Alive ($0 \leq l \leq u$)
Upper Bounding Line	$\eta z + 0$	$\frac{u-\eta l}{u-l} z + \frac{(\eta-1)l}{u-l} u$	$z + 0$
Lower Bounding Line	$\eta z + 0$	$\eta z + 0$	$z + 0$
The Adaptive Strategy			
Neuron status	Dead ($l \leq u \leq 0$)	Unstable ($l < 0 < u$)	Alive ($0 \leq l \leq u$)
Upper Bounding Line	$\eta z + 0$	$\frac{u-\eta l}{u-l} z + \frac{(\eta-1)l}{u-l} u$	$z + 0$
Lower Bounding Line	$\eta z + 0$	Case $\lvert l \rvert > u$: $\eta z + 0$ Case $\lvert l \rvert \leq u$: $z + 0$	$z + 0$

A.7.2 Bounding Lines for ParamRamp

Assume that the slope of the left part and the right part of ParamRamp is η, the input is z and the output is a. The input range of z is $[l, u]$.

$$a = \begin{cases} \eta z, \text{if } z < 0, \\ z, \text{if } 0 \leq z \leq r, \\ \eta z + (1-\eta) r, \text{if } z > r. \end{cases} \tag{139}$$

The tight strategy and the adaptive strategy to choose bounding lines for ParamRamp are presented in Table 2. In the constant strategy, the upper bounding line is chosen as $0z + \text{ParamRamp}(u)$ and the lower bounding line is chosen as $0z + \text{ParamRamp}(l)$.

Table 2: The tight strategy and the adaptive strategy to choose bounding lines for ParamRamp. We define $b = (1-\eta)r$, $k_1 = \frac{r-\eta l}{r-l}$, $b_1 = \frac{(\eta-1)l}{r-l} r$, $k_2 = \frac{\eta u + b}{u}$, $b_2 = 0$.

The Tight Strategy						
Neuron status	Left Dead $l \leq u \leq 0$	Left Unstable $l < 0 < u \leq r$	Alive $0 \leq l \leq u \leq r$	Right Unstable $0 \leq l < r < u$	Right Dead $r \leq l \leq u$	Unstable $l < 0 < r < u$
Upper BDL	$\eta z + 0$	$\frac{u-\eta l}{u-l} z + \frac{(\eta-1)l}{u-l} u$	$z + 0$	$\eta z + b$	$\eta z + b$	$\eta z + b$
Lower BDL	$\eta z + 0$	$\eta z + 0$	$z + 0$	$\frac{\eta u + b - l}{u-l} z + \frac{(1-\eta)u - b}{u-l} l$	$\eta z + b$	$\eta z + 0$
The Adaptive Strategy						
Neuron status	Left Dead $l \leq u \leq 0$	Left Unstable $l < 0 < u \leq r$	Alive $0 \leq l \leq u \leq r$	Right Unstable $0 \leq l < r < u$	Right Dead $r \leq l \leq u$	Unstable $l < 0 < r < u$
Upper BDL	$\eta z + 0$	$\frac{u-\eta l}{u-l} z + \frac{(\eta-1)l}{u-l} u$	$z + 0$	Case $u - r > r - l$: $\eta z + b$ Case $u - r \leq r - l$: $z + 0$	$\eta z + b$	Case $u - r > r - l$: $\eta z + b$ Case $u - r \leq r - l$: $k_1 z + b_1$
Lower BDL	$\eta z + 0$	Case $\lvert l \rvert > u$: $\eta z + 0$ Case $\lvert l \rvert \leq u$: $z + 0$	$z + 0$	$\frac{\eta u + b - l}{u-l} z + \frac{(1-\eta)u - b}{u-l} l$	$\eta z + b$	Case $\lvert l \rvert > u$: $\eta z + 0$ Case $\lvert l \rvert \leq u$: $k_2 z + b_2$

B Experiment

B.1 Network Structures used in the Main Text

We use 7 network structures in total in all of our experiments: DM-Small, DM-Medium, DM-Large, Small MNIST, Small CIFAR, CNN-7+BN and WideResNet. Structures of the first 6 networks are presented in Table 3. For the WideResNet network, we use 3 wide basic blocks with widen factor 10, which is the same setting in the work [3].

We use DM-Small, DM-Medium, and DM-Large to conduct IBP training and CROWN-IBP training on MNIST. DM-Large is used to train classifiers on CIFAR-10. And we use CNN-7+BN and WideResNet to train classifiers on Tiny-ImageNet. Small CIFAR is the network mentioned at the beginning of Section 3 in the main text. Small MNIST is the network mentioned in Figure 3 in the main text.

Table 3: DM-Small, DM-Medium and DM-Large are the same models used in the work [1]. The last fully connected layer is omitted. Activation function is ReLU or ParamRamp. *CONV K W×H+S* deotes a 2D convolutional layer using K filters of size W×H with a stride of S in both dimensions; *FC N* denotes a fully connected layer with an output dimension of N. For CNN-7+BN, there is a batch normalization layer before every activation layer.

DM-Small	**DM-Medium**	**DM-Large**	**Small CIFAR**	**Small MNIST**	**CNN-7+BN**
CONV 16 4×4+2	CONV 32 3×3+1	CONV 64 3×3+1	CONV 64 3×3+1	CONV 16 3×3+1	CONV 64 3×3+1
CONV 32 4×4+1	CONV 32 4×4+2	CONV 64 3×3+1	CONV 64 3×3+2	CONV 16 4×4+2	CONV 64 3×3+1
FC 100	CONV 64 3×3+1	CONV 128 3×3+2	FC 512	CONV 32 3×3+1	CONV 128 3×3+2
	CONV 64 4×4+2	CONV 128 3×3+1		CONV 32 4×4+2	CONV 128 3×3+1
	FC 512	CONV 128 3×3+1		FC 512	CONV 128 3×3+2
	FC 512	FC 512		FC 512	FC 512

B.2 Tightness Comparison of IBP, LBP and CROWN on Normally Trained Networks

In Section 4 in the main text, we compare IBP, LBP and CROWN on a normally trained MNIST classifier. The network we use is named Small MNIST (with ReLU activation) and its detailed structure can be seen in Table 3. We train the network on MNIST for 10 epochs with learning rate 5×10^{-4} using the Adam optimizer.

In this section, we compare them on a normally trained CIFAR-10 classifier. The network structure is DM-Large defined in Table 3 with ReLU activation. We train the network on CIFAR-10 for 300 epochs with learning rate 5×10^{-4} using the Adam optimizer. We normalize the input using their channel-wise mean and standard deviation. We also augment the training data by applying random horizontal flips and random crops. Results are shown in Figure 1. We don't test CROWN on this network because it exceeds GPU memory (12 GB) and takes about half an hour to verify a single image on one Intel Xeon E5-2650 v4 CPU. We can see LBP and CROWN-LBP is $1 \sim 3$ orders tighter than IBP and CROWN-IBP across a broad range of perturbation radius ϵ, and it generally yields tighter bounds to use the adaptive strategy to choose bounding lines than the tight strategy. Therefore, we conclude LBP and CROWN-LBP have great advantages to verify normally trained large networks.

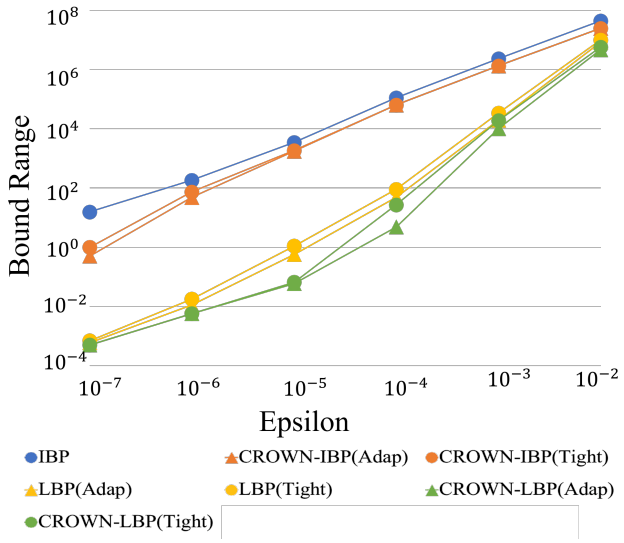


Figure 1: Tightness comparison of IBP, CROWN-IBP, LBP, CROWN-LBP, CROWN on a normally trained CIFAR-10 classifier. "Bound Range" is mean of $\mathbf{u}^{(m)} - \mathbf{l}^{(m)}$. The mean is taken over the 10 output logits and averaged over 100 test images in CIFAR-10. "Epsilon" is the radius of the l_∞ ball. "Adap" and "Tight" are the adaptive and tight strategies to choose bounding lines.

B.3 Investigate limited improvement of LBP and CROWN-LBP on IBP trained networks.

In Section 5 in the main text, we investigate why the improvement of LBP and CROWN-LBP over IBP and CROWN-IBP is so small on the IBP trained network compared with the normally trained network. We argue that this is because most neurons are dead in IBP trained networks. In this section, we conduct experiments to further support this explanation.

We use IBP to train a DM-large network with ReLU activation on CIFAR-10 at $\epsilon = 8.8/255$. See the detailed training method in Section B.4. In Table 1 in the main text, we already observe that the lower bounds of the margin computed by IBP and LBP are very close on this network. We argue that this is because most neurons are dead in this network. IBP and LBP choose the same bounding lines for dead ReLU neurons. To further verify this explanation, we replace ReLU activations in this IBP trained network with LeakyReLU activations while remain the model's other parameters. Note that for LeakyReLU activation, the bounding lines chosen by LBP in the tight strategy will be tighter than those constant bounding lines chosen by IBP for dead neurons. And the difference becomes significant if the slope of the left part of LeakyReLU activation is large. This can be seen from Table 1.

Now we gradually increase the slope of the left part of LeakyReLU and observe the change in bounds computed by IBP, CROWN-IBP, LBP and CROWN-LBP. Results are shown in Figure 2. We can see the difference in bounds computed by IBP, CROWN-IBP, LBP and CROWN-LBP increases as the slope increases. The lower bounds computed by LBP is much tighter than IBP when the slope of the left part of LeakyReLU is 0.01. This is exactly what we expect.

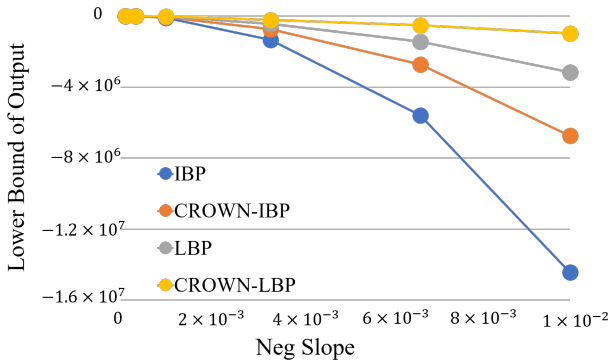


Figure 2: Change in lower bound of the margin if we replace all ReLU activations with LeakyReLU in the IBP trained CIFAR-10 network. The lower bound is computed by 4 methods: IBP, CROWN-IBP, LBP, and CROWN-LBP. The tight strategy is adopted when choosing bounding lines and the lower bound of the margin is computed at $\epsilon = 8/255$ using IBP. The horizontal axis is the slope of the left part of LeakyReLU activation. The vertical axis is mean of the lower bound of the margin averaged over the first 100 images in CIFAR-10 test set.

B.4 Detailed Experimental Setup

Experiments on MNIST and CIFAR-10. For experiments on MNIST and CIFAR-10 datasets, we follow the same training procedures in the works [4, 1]. We train 3 networks on MNIST dataset: DM-Small, DM-Medium, DM-Large. And we use the DM-Large network to train classifiers on CIFAR-10. The detailed network structures are presented in Table 3. The loss we use for IBP training is

$$\mathop{\mathbb{E}}_{(\mathbf{x}_0,y)\in\mathcal{X}} \kappa L(\mathbf{z}^{(m)}(\mathbf{x}_0),y) + (1-\kappa)L(-\mathbf{l}^{\boldsymbol{\omega}}_{\text{IBP}}(\mathbf{x}_0,\epsilon),y), \tag{140}$$

and the loss we use for CROWN-IBP training is

$$\begin{aligned}\mathop{\mathbb{E}}_{(\mathbf{x}_0,y)\in\mathcal{X}} \kappa L(\mathbf{z}^{(m)}(\mathbf{x}_0),y) + (1-\kappa)L(-\mathbf{l}^{\boldsymbol{\omega}}(\mathbf{x}_0,\epsilon),y),\\ \text{where } \mathbf{l}^{\boldsymbol{\omega}} = (1-\beta)\mathbf{l}^{\boldsymbol{\omega}}_{\text{IBP}} + \beta\mathbf{l}^{\boldsymbol{\omega}}_{\text{C.-IBP}}.\end{aligned} \tag{141}$$

For IBP training on MNIST, we train the networks at $\epsilon_{\text{train}} = 0.4$ and test them at $\epsilon_{\text{test}} = 0.2, 0.3, 0.4$, respectively. The batchsize we use is 256. We use the Adam optimizer to train the networks for 200 epochs with learning rate 5×10^{-4}. The learning rate is decreased by $10\times$ at epoch 130 and 190, respectively. We set $\epsilon = 0$ for the first 10 epochs (clean training with normal cross-entropy loss), then we linearly increases it from 0 to ϵ_{train} in the next 50 epochs. ϵ is kept to ϵ_{train} in the rest epochs. κ starts from κ_{start} and ends with κ_{end} during training: It is linearly decreased from κ_{start} to κ_{end} from the 10-th epoch to the 60-th epoch. Zhang *et al* [4] design 3 schedules for κ. They are $(\kappa_{\text{start}} = 1, \kappa_{\text{end}} = 0.5), (\kappa_{\text{start}} = 1, \kappa_{\text{end}} = 0)$, and $(\kappa_{\text{start}} = 0, \kappa_{\text{end}} = 0)$. We run experiments for all of the 3 schedules and report the one with the lowest verified error.

For CROWN-IBP training on MNIST, the training procedures are the same as IBP training, except that we apply a schedule for β. β starts from 1 and ends with 0 during training: It is linearly decreased from 1 to 0 from the 10-th epoch to the 60-th epoch. The adaptive strategy is adopted for CROWN-IBP training.

For IBP training on CIFAR-10, we train the networks at $\epsilon_{\text{train}} = 2.2/255, 8.8/255$ and test them at $\epsilon_{\text{test}} = 2/255, 8/255$, respectively. We normalize the input using their channel-wise mean and standard deviation. We augment the training data by applying random horizontal flips and random crops. The batchsize we use is 1024. We use the Adam optimizer to train the networks for 3200 epochs with learning rate 5×10^{-4}. The learning rate is decreased by $10\times$ at epoch 2600 and 3040, respectively. We set $\epsilon = 0$ for the first 320 epochs (clean training with normal cross-entropy loss), then we linearly increases it from 0 to ϵ_{train} in the next 1600 epochs. ϵ is kept to ϵ_{train} in the rest epochs. κ starts from κ_{start} and ends with κ_{end} during training: It is linearly decreased from κ_{start} to κ_{end} from the 320-th epoch to the 1920-th epoch. 3 schedules for κ are similarly designed. They are $(\kappa_{\text{start}} = 1, \kappa_{\text{end}} = 0.5), (\kappa_{\text{start}} = 1, \kappa_{\text{end}} = 0)$, and $(\kappa_{\text{start}} = 0, \kappa_{\text{end}} = 0)$. We observe that the lowest verified error is usually obtained with $(\kappa_{\text{start}} = 0, \kappa_{\text{end}} = 0)$ in the work [4]. Therefore, we only conduct experiments on this schedule. But when comparing with results from the work [4], we report their lowest verified error in the 3 schedules.

For CROWN-IBP training on CIFAR-10, the training procedures are the same as IBP training, except that we apply a schedule for β. β starts from 1 and ends with β_{end} during training: It is linearly decreased from 1 to β_{end} from the 320-th epoch to the 1920-th epoch. β_{end} is set to 0 for networks trained at $\epsilon_{\text{train}} = 8.8/255$, and is set to 1 for networks trained at $\epsilon_{\text{train}} = 2.2/255$. The adaptive strategy is adopted for CROWN-IBP training.

Experiments on Tiny-ImageNet. For experiments on Tiny-ImageNet, we follow the same training procedures in the work [3]. We train 2 networks on Tiny-ImageNet: CNN-7+BN and WideResNet. Their detailed structures are presented in Section B.1. The loss we use is the fusion loss defined in the work [3]. We refer readers to the original work for details about this loss. We train the networks and test them at $1/255$. We normalize the input using their channel-wise mean and standard deviation. We use the Adam optimizer to train the networks for 1000 epochs with learning rate 5×10^{-4}. The learning rate is decreased by $10\times$ at epoch 600 and 700, respectively. Batchsize is set to 110. We set $\epsilon = 0$ for the first 100 epochs (normal training with cross-entropy loss), then we linearly increase it from 0 to $1/255$ in the next 400 epochs. ϵ is kept to $1/255$ in the rest epochs. We augment the training data by applying random horizontal flips and random crops of 56×56. During test, we use a center crop of 56×56. For CROWN-IBP training, there is one more hyperparameter β, which is the same as the one defined in (141). β starts from 1 and ends with 0 during training: It is linearly decreased from 1 to 0 from the 100-th epoch to the 500-th epoch.

Settings of ParamRamp. Recall that every ParamRamp activation has a tunable parameter r. We include it to the parameters of the network and optimize over it during training. It is important to initialize it appropriately. As mentioned in the above paragraphs, when we use IBP or CROWN-IBP to train networks on MNIST, CIFAR-10 and Tiny-ImageNet, the first several epochs are clean training with normal cross-entropy loss. We replace ParamRamp with LeakyReLU in these epochs of clean training. At the end of the clean training, we record the mean activation of every neuron across all images in the training set. We use this mean activation as the initial value for r and restore the ParamRamp activation in the rest training process.

We find that networks with ParamRamp activation trained at $\epsilon = 2.2/255$ on CIFAR-10 using IBP have serious overfitting problems compared with ReLU networks. We speculate that the overfitting problem is caused by the additional parameters r brought by ParamRamp activation. To mitigate this problem, we add a random Gaussian noise $\mathcal{N}(0, 0.01r_{\text{mean}})$ to the parameter r after every gradient descent step. r_{mean} is the mean of the parameter r for all neurons in the corresponding intermediate layer.

Recall that we design a variant of ParamRamp activation: Ramp(0.01→0), where η starts from 0.01 and gradually decreases to 0 during training. For MNIST, η is kept to 0.01 in the first 70 epochs, then it is linearly decreased to 0 in the next 70 epochs. Finally, it is kept to 0 in the rest epochs. For CIFAR-10, η is kept to 0.01 in the first 1000 epochs, then it is linearly decreased to 0 in the next 1200 epochs. Finally, it is kept to 0 in the rest epochs. The η schedule for ReLU(0.01→0) is designed in the same way.

B.5 Complete Experiment Results

B.5.1 Complete Results on MNIST

In the main text, we only present results trained on the DM-Large network and tested at $\epsilon = 0.4$. We present the complete results on the DM-Large network in Table 4. Recall that there are 3 different schedules for κ mentioned in Section B.4. We conduct experiments for all of the 3 schedules and report the one with the lowest verified error in Table 4. We also conduct experiments on the DM-Medium and DM-Small networks to demonstrate that our proposed activation, ParamRamp, outperforms ReLU across different network structures. For simplicity, we only conduct experiments on one schedule of κ: $(\kappa_{\text{start}} = 0, \kappa_{\text{end}} = 0)$, which prioritizes the minimization of the second term in (140). Results on DM-Medium and DM-Small are presented in Table 5 and Table 6, respectively.

Table 4: Errors of IBP trained and CROWN-IBP trained networks with different activations on MNIST. The model structure we use is DM-Large. We report errors on clean images(Clean), IBP verified errors(IBP), CROWN-LBP verified errors(C.-LBP), and PGD attack errors(PGD). Experiments are conducted on 3 variants of ParamRamp: Ramp(0), ParamRamp with $\eta = 0$; Ramp(0.01), ParamRamp with $\eta = 0.01$; Ramp(0.01→0), η starts from 0.01 and gradually decreases to 0 during training. 3 variants of ReLU are similarly designed. Networks are trained at $\epsilon = 0.4$ and evaluated at $\epsilon = 0.2, 0.3, 0.4$, respectively. Results of ReLU(0) are directly copied from the original works [1, 4]. We compute C.-LBP verified errors based on our re-runned networks for these experiments. Therefore, C.-LBP verified errors are not comparable to IBP verified errors on these networks.

Training Method	Activation	Errors (%) for $\epsilon = 0.2$				Errors (%) for $\epsilon = 0.3$				Errors (%) for $\epsilon = 0.4$			
		Clean	IBP	C.-LBP	PGD	Clean	IBP	C.-LBP	PGD	Clean	IBP	C.-LBP	PGD
IBP	ReLU(0)	2.12	4.75	5.53	4.24	2.12	8.47	8.99	6.78	2.74	14.80	16.13	11.14
	ReLU(0.01)	2.36	5.30	5.16	4.59	3.02	8.78	8.63	7.76	3.02	14.49	14.06	12.18
	ReLU(0.01→0)	1.90	4.23	4.21	3.76	1.87	7.21	7.18	5.88	2.68	13.1	13.08	9.71
	Ramp(0)	**1.58**	3.91	3.91	**3.09**	**1.79**	6.62	6.61	**4.00**	**1.78**	11.86	11.86	**5.18**
	Ramp(0.01)	1.85	4.01	3.96	3.63	1.84	6.74	6.65	5.64	2.32	11.78	11.52	9.76
	Ramp(0.01→0)	1.79	**3.73**	**3.71**	3.24	2.16	**6.30**	**6.29**	4.17	2.16	**10.90**	**10.88**	6.59
CROWN -IBP	ReLU(0)	1.82	4.13	4.03	3.81	1.82	7.02	6.73	6.05	2.17	12.06	11.90	9.47
	ReLU(0.01)	2.12	4.62	4.47	4.13	2.12	7.69	7.51	6.65	2.68	13.17	12.75	11.13
	ReLU(0.01→0)	1.70	4.08	4.07	3.67	1.72	6.78	6.76	5.16	2.48	11.62	11.55	8.27
	Ramp(0)	**1.59**	**3.59**	**3.57**	**2.84**	**1.63**	6.16	6.14	**4.15**	**1.94**	10.76	10.74	**6.55**
	Ramp(0.01)	1.88	4.01	3.89	3.64	1.88	6.55	6.39	5.53	2.35	11.25	10.99	9.61
	Ramp(0.01→0)	1.77	3.69	3.68	3.32	1.76	**5.99**	**5.98**	4.29	2.36	**10.68**	**10.61**	6.61

Table 5: Results of the DM-Medium network on MNIST. Notations are the same as the ones in Table 4.

Training Method	Activation	Errors (%) for $\epsilon = 0.2$				Errors (%) for $\epsilon = 0.3$				Errors (%) for $\epsilon = 0.4$			
		Clean	IBP	C.-LBP	PGD	Clean	IBP	C.-LBP	PGD	Clean	IBP	C.-LBP	PGD
IBP	ReLU(0)	4.05	6.89	6.88	6.40	4.06	9.93	9.91	8.10	4.06	15.85	15.84	11.02
	ReLU(0.01)	4.19	7.42	7.36	7.18	4.18	11.03	10.77	10.15	4.18	17.92	17.66	16.04
	ReLU(0.01→0)	4.39	6.98	6.98	6.74	4.45	9.86	9.83	8.75	4.40	15.21	15.18	12.35
	Ramp(0)	**2.70**	5.27	**5.26**	**4.58**	**2.71**	7.95	7.95	**5.69**	**2.69**	13.50	13.48	**7.46**
	Ramp(0.01)	3.19	5.29	5.29	5.08	3.20	7.86	7.81	7.25	3.21	13.13	13.01	11.50
	Ramp(0.01→0)	2.90	**5.26**	**5.26**	4.61	2.96	**7.70**	**7.70**	6.01	2.98	**12.52**	**12.52**	8.12
CROWN -IBP	ReLU(0)	3.08	5.40	5.40	4.92	3.09	7.88	7.88	6.87	3.09	13.03	13.01	9.96
	ReLU(0.01)	3.88	6.88	6.77	6.58	3.88	9.85	9.72	9.30	3.88	15.65	15.30	14.20
	ReLU(0.01→0)	3.53	5.81	5.80	5.55	3.52	8.40	8.38	7.60	3.53	13.36	13.33	10.07
	Ramp(0)	**2.56**	4.66	4.66	**3.97**	**2.54**	7.18	7.18	**5.46**	**2.62**	12.24	12.23	**7.19**
	Ramp(0.01)	2.71	4.87	4.86	4.67	2.71	7.38	7.31	6.82	2.71	12.35	12.26	10.84
	Ramp(0.01→0)	2.78	**4.49**	**4.48**	4.16	2.74	**6.88**	**6.87**	5.53	2.75	**11.59**	**11.53**	7.96

Table 6: Results of the DM-Small network on MNIST. Notations are the same as the ones in Table 4.

Training Method	Activation	Errors (%) for $\epsilon = 0.2$				Errors (%) for $\epsilon = 0.3$				Errors (%) for $\epsilon = 0.4$			
		Clean	IBP	C.-LBP	PGD	Clean	IBP	C.-LBP	PGD	Clean	IBP	C.-LBP	PGD
IBP	ReLU(0)	4.66	8.38	8.26	7.80	5.06	11.77	11.76	10.42	5.05	18.25	18.22	14.79
	ReLU(0.01)	9.61	14.79	14.73	14.67	9.59	19.55	19.49	19.34	9.59	27.20	27.02	26.57
	ReLU(0.01→0)	7.10	10.25	10.24	7.10	7.11	13.30	13.30	12.50	7.11	19.00	18.97	16.23
	Ramp(0)	**3.17**	**5.73**	**5.73**	**5.16**	**3.19**	**8.52**	**8.48**	**6.48**	**3.18**	**13.55**	**13.50**	**8.43**
	Ramp(0.01)	4.89	8.05	8.00	7.89	4.88	11.37	11.22	10.83	4.92	17.69	17.52	16.48
	Ramp(0.01→0)	4.08	6.47	6.45	5.87	4.08	9.10	9.09	7.16	4.08	14.02	14.01	9.68
CROWN -IBP	ReLU(0)	4.45	7.33	7.32	6.94	4.46	10.30	10.26	9.28	4.47	16.16	16.11	13.91
	ReLU(0.01)	7.65	12.61	12.52	12.34	7.57	17.27	17.07	16.72	7.59	25.40	25.15	24.39
	ReLU(0.01→0)	6.84	10.35	10.32	10.00	6.82	13.66	13.64	12.71	6.83	19.66	19.65	17.32
	Ramp(0)	**3.28**	**5.84**	**5.81**	**5.25**	**3.28**	**8.59**	**8.56**	**6.55**	**3.27**	**13.75**	**13.70**	**9.23**
	Ramp(0.01)	4.70	8.21	8.15	8.01	4.73	12.08	11.88	11.49	4.74	18.56	18.29	17.75
	Ramp(0.01→0)	3.93	6.49	6.45	5.98	3.95	9.40	9.38	7.65	3.93	14.66	14.64	11.11

B.5.2 Complete Results on Tiny-ImageNet

We train 2 networks on Tiny-ImageNet: CNN-7+BN and WideResNet. Their detailed structures are presented in Section B.1. The loss we use is the fusion loss defined in the work [3]. We refer readers to the original work for details about this loss. We train the networks and test them at $1/255$. We only perform IBP training on this dataset, as CROWN-IBP training is computational expensive and we observe that the improvement of CROWN-IBP training over IBP training is not as significant as the improvement achieved on MNIST and CIFAR-10 in the work [3]. The results are shown in Table 7. Results of both IBP training and CROWN-IBP training for ReLU(0) are from the work [3]. We only perform IBP training for Ramp(0). We can see that IBP trained networks with ParamRamp activation outperforms both IBP trained and CROWN-IBP trained ReLU networks in terms of verified errors.

Table 7: Comparison of ParamRamp and ReLU on Tiny-ImageNet dataset. Notations are the same as those in Table 4. The network is both trained and tested at $\epsilon = 1/255$. Results for ReLU(0) are from the work [3].

Training Method	Activation	CNN-7+BN		WideResNet	
		IBP	Clean	IBP	Clean
IBP	ReLU(0)	87.96	78.54	85.15	73.54
	Ramp(0)	84.99	78.77	82.94	70.97
CROWN-IBP	ReLU(0)	87.31	78.42	84.14	72.18

B.5.3 Computational Overhead of ParamRamp

ParamRamp activation brings additional parameters to the network. We are concerned about its computational overhead compared with ReLU networks. We report the average training time per epoch for ParamRamp and ReLU networks on MNIST, CIFAR-10 and Tiny-ImageNet datasets in Table 8. On MNIST and CIFAR-10, the training time is tested on the DM-Large network, and on Tiny-ImageNet, the training time is tested on WideResNet. We can see that the computational overhead of ParamRamp for IBP training is small, and the computational overhead for CROWN-IBP training is within $2\times$.

Table 8: Computational overhead of ParamRamp compared with ReLU. Some experiments are run on multiple GPUs. Therefore, we report the average training time per epoch times the number of GPUs used for fair comparison.

MNIST				
Training Method	Activation	Device	Time per Epoch (s)	Overhead
IBP	ReLU	1 × NVIDIA GeForce GTX TITAN X	59.1	1.09×
	Ramp	1 × NVIDIA GeForce GTX TITAN X	64.7	
CROWN-IBP	ReLU	1 × NVIDIA GeForce GTX TITAN X	110.1	1.51×
	Ramp	2 × NVIDIA GeForce GTX TITAN X	166.1	
CIFAR-10				
Training Method	Activation	Device	Time per Epoch (s)	Overhead
IBP	ReLU	2 × NVIDIA GeForce RTX 2080 Ti	45.5	0.98×
	Ramp	2 × NVIDIA GeForce RTX 2080 Ti	44.7	
CROWN-IBP	ReLU	4 × NVIDIA GeForce RTX 2080 Ti	189.6	1.86×
	Ramp	4 × NVIDIA GeForce RTX 2080 Ti	353.5	
Tiny-ImageNet				
Training Method	Activation	Device	Time per Epoch (s)	Overhead
IBP	ReLU	4 × NVIDIA GeForce RTX 2080 Ti	403.2	1.20×
	Ramp	4 × NVIDIA GeForce RTX 2080 Ti	485.1	

B.5.4 Neuron Status of a Normally Trained Network

We present the neuron status of an IBP trained ReLU network on CIFAR-10 in Figure 4 in the main text. We find that most neurons are dead in this network. In this section, we present the neuron status of a normally trained ReLU network on CIFAR-10. The network we use is the DM-Large network. We train the network on CIFAR-10 for 300 epochs with learning rate 5×10^{-4} using the Adam optimizer. We normalize the input using their channel-wise mean and standard deviation. We also augment the training data by applying random horizontal flips and random crops. Neuron status of this network is shown in Figure 3. We can see that most neurons are unstable in this normally trained network, especially in the latter layers, basically all neurons are in the unstable status.

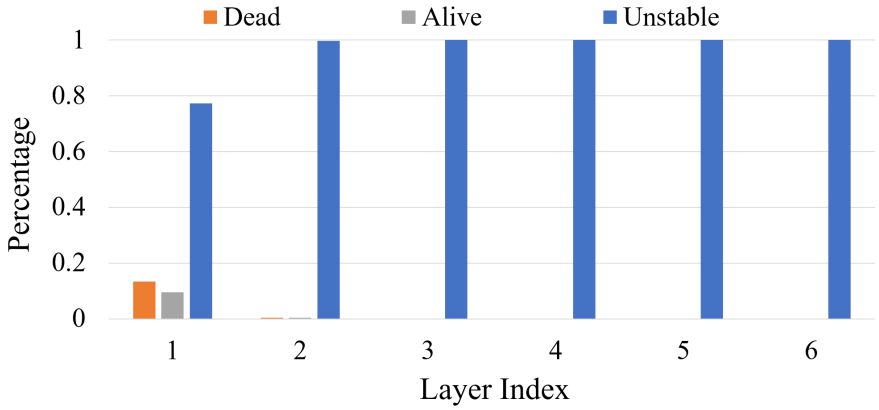
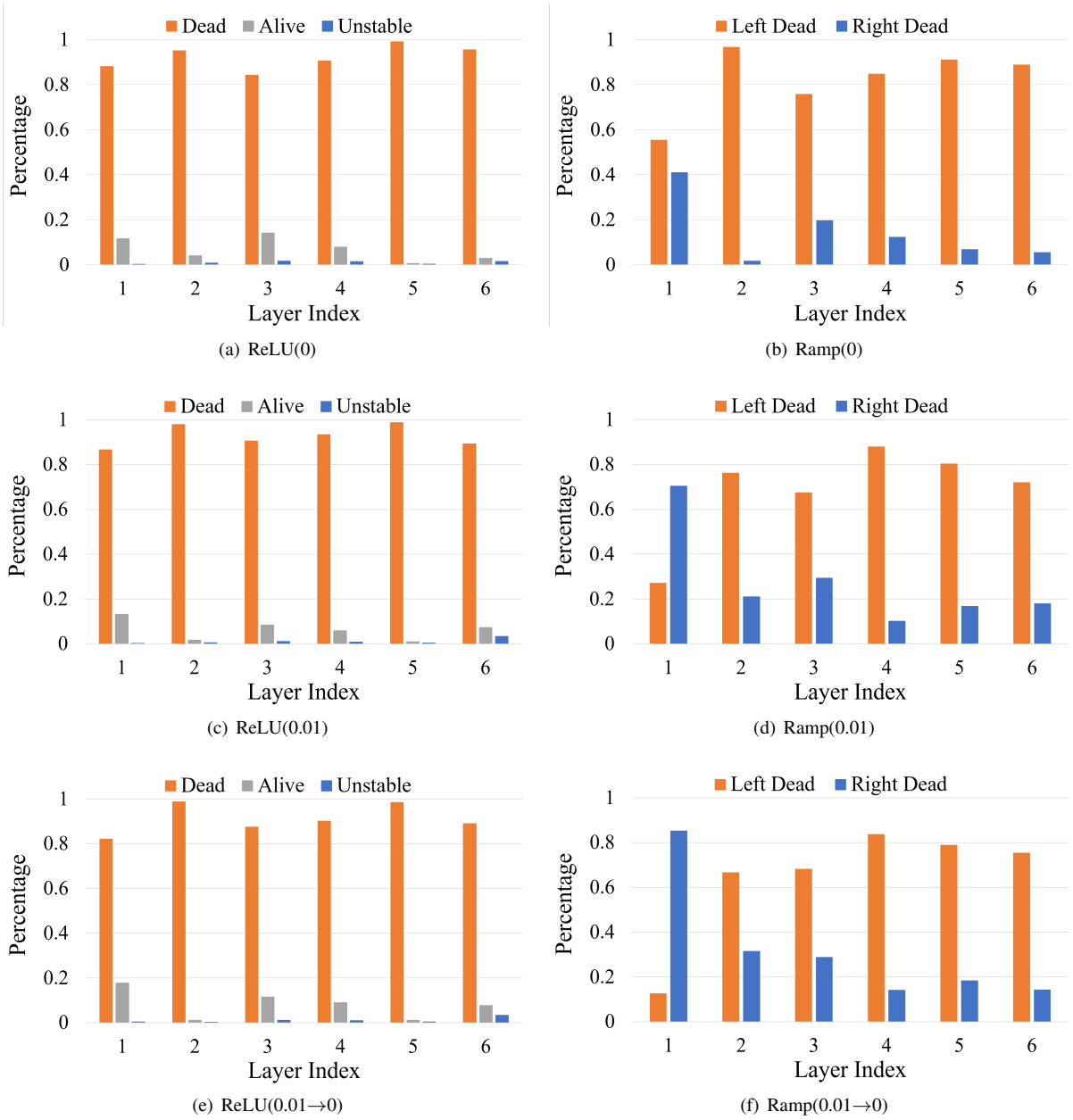


Figure 3: Neuron status of a normally trained DM-Large network on CIFAR-10. Bounds are computed using IBP at $\epsilon = 2/255$. The horizontal axis is the layer index, and the vertical axis is the percentage of every neuron status in the layer. The percentage is averaged over 100 CIFAR-10 test images.

B.5.5 Neuron Status Comparison of More Networks Trained on MNIST

In this section, we compare the neuron status of ReLU networks and ParamRamp networks trained on MNIST. The networks are IBP trained at $\epsilon = 0.4$. The κ schedule we use for training is $(\kappa_{\text{start}} = 1, \kappa_{\text{end}} = 0.5)$ described in Section B.4. The bounds of all layers are computed using IBP at $\epsilon = 0.2$. We train 3 variants of ReLU networks: ReLU(0), ReLU with $\eta = 0$; ReLU(0.01), LeakyReLU with $\eta = 0.01$; ReLU(0.01→0), η starts from 0.01 and gradually decreases to 0 during training. 3 variants of ParamRamp are similarly designed. Neuron status of these networks are shown in Figure 4. We can see most neurons are in the dead status in ReLU networks. In comparison, there are a considerable amount of neurons are in the right dead status in ParamRamp networks, especially in the first layer of the network.



(a) ReLU(0) (b) Ramp(0)

(c) ReLU(0.01) (d) Ramp(0.01)

(e) ReLU(0.01→0) (f) Ramp(0.01→0)

Figure 4: Neuron status comparison of ReLU networks and ParamRamp networks on MNIST. Bounds are computed using IBP at $\epsilon = 0.2$. The horizontal axis is the layer index, and the vertical axis is the percentage of every neuron status in the layer. The percentage is averaged over 100 MNIST test images.

References

[1] S. Gowal, K. Dvijotham, R. Stanforth, R. Bunel, C. Qin, J. Uesato, R. Arandjelovic, T. A. Mann, and P. Kohli. Scalable verified training for provably robust image classification. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4841–4850, 2019.

[2] Z. Lyu, C.-Y. Ko, Z. Kong, N. Wong, D. Lin, and L. Daniel. Fastened crown: Tightened neural network robustness certificates. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:5037–5044, 04 2020.

[3] K. Xu, Z. Shi, H. Zhang, Y. Wang, K.-W. Chang, M. Huang, B. Kailkhura, X. Lin, and C.-J. Hsieh. Automatic perturbation analysis for scalable certified robustness and beyond, 2020.

[4] H. Zhang, H. Chen, C. Xiao, S. Gowal, R. Stanforth, B. Li, D. Boning, and C.-J. Hsieh. Towards stable and efficient training of verifiably robust neural networks. In *International Conference on Learning Representations*, 2020.