Tight Semidefinite Relaxations for Verifying Robustness of Neural Networks

Godai Azuma[1, 2] Sunyoung Kim[3] Makoto Yamashita[2]

April 14, 2025

Abstract

For verifying the safety of neural networks (NNs), Fazlyab et al. (2019) introduced a semidefinite programming (SDP) approach called DeepSDP. This formulation can be viewed as the dual of the SDP relaxation for a problem formulated as a quadratically constrained quadratic program (QCQP). While SDP relaxations of QCQPs generally provide approximate solutions with some gaps, this work focuses on tight SDP relaxations that provide exact solutions to the QCQP for single-layer NNs. Specifically, we analyze tightness conditions in three cases: (i) NNs with a single neuron, (ii) single-layer NNs with an ellipsoidal input set, and (iii) single-layer NNs with a rectangular input set. For NNs with a single neuron, we propose a condition that ensures the SDP admits a rank-1 solution to DeepSDP by transforming the QCQP into an equivalent two-stage problem leads to a solution collinear with a predetermined vector. For single-layer NNs with an ellipsoidal input set, the collinearity of solutions is proved via the Karush-Kuhn-Tucker condition in the two-stage problem. In case of single-layer NNs with a rectangular input set, we demonstrate that the tightness of DeepSDP can be reduced to the single-neuron NNs, case (i), if the weight matrix is a diagonal matrix.

Key words. Neural network, Safety verification, Tight semidefinite relaxations, Rank-1 solutions, Decomposition into two-stage problem.

MSC Classification. 62M45, 90C20, 90C22, 90C25, 90C26.

[1]Department of Industrial and Systems Engineering, Aoyama Gakuin University, 5-10-1-O-410b Fuchinobe, Chuo-ku, Sagamihara-shi, Kanagawa 252-5258, Japan (azuma@ise.aoyama.ac.jp). The research of Godai Azuma was supported by JSPS KAKENHI Grant Number JP24K20738.

[2]Department of Mathematical and Computing Science, Institute of Science Tokyo, 2-12-1-W8-29 Oh-Okayama, Meguro-ku, Tokyo 152-8550, Japan. (Makoto.Yamashita@comp.isct.ac.jp). The research of Makoto Yamashita was partially supported by JSPS KAKENHI Grant Number 24K14836.

[3]Department of Mathematics, Ewha W. University, 52 Ewhayeodae-gil, Sudaemoon-gu, Seoul 03760, Korea (skim@ewha.ac.kr). This work was supported by NRF 2021-R1A2C1003810.

1 Introduction

Neural networks (NNs) have gained significant attention over the past two decades due to their theoretical foundations and practical applications in various fields, including image processing [10], sound recognition [7], and natural language processing [14, 15]. Ensuring the robustness and safety of NNs against adversarial perturbations to test inputs is a key challenge in safe machine learning. To address this issue, we present a semidefinite programming (SDP) verifier for a neural network by showing that the SDP relaxation provides exact solutions under suitable assumptions.

Consider an L-layer feed-forward NN [15] described as a function $f : \mathbb{R}^{n_0} \to \mathbb{R}^{n_{L+1}}$ of form:

$$\left.\begin{aligned} \boldsymbol{x}^{k+1} &\coloneqq \phi\left(W^k \boldsymbol{x}^k + \boldsymbol{b}^k\right), \quad k = 0, \ldots, L-1, \\ f(\boldsymbol{x}^0) &\coloneqq W^L \boldsymbol{x}^L + \boldsymbol{b}^L, \end{aligned}\right\} \tag{1}$$

where $\boldsymbol{x}^0 \in \mathbb{R}^{n_0}$ is the input vector, $\boldsymbol{x}^k \in \mathbb{R}^{n_k}$ $(k = 1, \ldots, L)$ corresponds to the neurons in the kth layer. The matrix $W^k \in \mathbb{R}^{n_{k+1} \times n_k}$ and the vector $\boldsymbol{b}^k \in \mathbb{R}^{n_{k+1}}$ are referred to as the weight matrix and bias vector at the kth layer, respectively. The function $\phi(\boldsymbol{x})$ is a vector-valued function where each element is an activation function. In this paper, we are interested in the rectified linear unit (ReLU) activation function, that is $\phi(\boldsymbol{x})_i = \max\{\alpha x_i, \beta x_i\}$ with $\alpha < \beta$ for every $i \in \mathbb{N}$. The ReLU function can be emulated with three quadratic constraints:

$$(\phi(\boldsymbol{x})_i - \alpha x_i)(\phi(\boldsymbol{x})_i - \beta x_i) = 0,\ \phi(\boldsymbol{x})_i \geq \alpha x_i,\ \phi(\boldsymbol{x})_i \geq \beta x_i. \tag{2}$$

As a result, the NN in (1) can be formulated as a quadratically-constrained quadratic programs (QCQP) with an appropriate objective function.

It is well known that small input perturbations ε in NNs can lead to significant changes in the output $f(\boldsymbol{x}^0+\varepsilon)$ [23, 27], which makes NNs vulnerable to adversarial inputs that produce incorrect results. This vulnerability undermines safety-critical applications including self-driving cars. To model the vulnerability behavior of NN (1), optimization methods such as QCQPs have been proposed with expressing the ReLU function as quadratic inequalities. Specifically, mixed-integer linear programming programs have been incorporated into the algorithms in [4, 8, 21, 28] and the duality of the convex relaxation has also been used in [9, 25, 30].

The SDP relaxation has been extensively studied as a powerful tool for generating approximate solutions to QCQPs [13, 22]. To address the robustness of NNs, Zhang [31] proposed a tight semidefinite program (SDP) relaxation by analyzing the *collinearity* of solutions of the SDP relaxation and presenting a condition under which the optimal value of the SDP relaxation is equivalent to that of the QCQP formulated for a single-layer NN. Based on this result, the approach in [31] was able to generate adversarial inputs with high accuracy.

Another approach that employs SDP relaxations to understand the vulnerability behavior of NNs is DeepSDP, which was recently proposed by Fazlyab et al. [11, 12]. DeepSDP has been developed in the context of safety verification [16, 17]. To evaluate the safety of NNs, safety verification determines whether the output set $f(\mathcal{X}) := \{f(\boldsymbol{x}^0) \mid \boldsymbol{x}^0 \in \mathcal{X}\}$, where $\mathcal{X} \subset \mathbb{R}^{n_0}$ is a given input set, remains within a predefined safety specification set

$S_y \subset \mathbb{R}^{n_{L+1}}$. Fazylab et al. [11, 12] employed the S-lemma to formulate DeepSDP as the following SDP problem:

$$\begin{array}{ll} \min\limits_{P,Q,S} & g(P,Q,S) \\ \text{s.t.} & M_{\text{in}}(P) + M_{\text{mid}}(Q) + M_{\text{out}}(S) \succeq O, \\ & P \in \mathcal{P}_{\mathcal{X}},\ Q \in \mathcal{Q}_{\phi},\ S \in \mathcal{S}, \end{array} \tag{3}$$

where g is a convex function, $\mathcal{P}_{\mathcal{X}}$, $\mathcal{Q}_{\phi}$, and $\mathcal{S}$ are matrix sets representing the domain space of variables. The set $\mathcal{P}_{\mathcal{X}}$ describes the information of the input set $\mathcal{X}$, $\mathcal{Q}_{\phi}$ emulates the ReLU activation function ϕ, and $\mathcal{S}$ specifies a safety specification set $S_y \subseteq \mathbb{R}^{n_{L+1}}$, where $\mathbb{R}^{n_{L+1}}$ denotes the n_{L+1}-dimensional Euclidean space of column vectors $\boldsymbol{y} = \begin{bmatrix} y_1, \ldots, y_{n_{L+1}} \end{bmatrix}^{\mathrm{T}}$. The three functions M_{in}, M_{mid} and M_{out} lift up the three variable matrices P, Q, S to an appropriate dimension, and $X \succeq O$ denotes that X is positive semidefinite. Fazlyab et al. [11, 12] showed that if (P, Q, S) is a feasible solution (3), DeepSDP provides an safety specification set $S_y \subset \mathbb{R}^{n_L+1}$ that encompasses the output set $f(\mathcal{X})$ (see Theorem 3.1 in Section 3.3 for details).

DeepSDP is less accurate than other safety verification methods as shown in [24], since (3) is not always tight for the corresponding QCQP. In this paper, we investigate the tightness of DeepSDP (3) as the SDP relaxation for the QCQP, using the collinearity. It should be noted that the approach in Zhang [31] is inapplicable to DeepSDP (1), since DeepSDP (1) includes the information of the input set $\mathcal{X}$ and the safety verification set S_y. Our approach is to show the collinearity utilizing the second projection theorem [5, Theorem 9.8] and the Karush-Kuhn-Tucker conditions.

The main contribution of this paper is to provide tightness conditions for DeepSDP (3) in (i) single-neuron NNs, i.e., $L = 1$ and $n_0 = n_1 = n_2 = 1$; (ii) single-layer NNs with ellipsoidal inputs $\mathcal{X}$ and polytope safety specification set S_y; and (iii) single-layer NNs with rectangular inputs $\mathcal{X}$ and polytope safety specification set S_y. Whether DeepSDP can provide a robust or exact solution can be analytically determined by examining the conditions. For (ii) and (iii), we consider the cases where $\mathcal{X}$ is an ellipsoid and a hyper-rectangle as the neighborhood $\{\boldsymbol{x} \in \mathbb{R}^{n_0} \mid \|\boldsymbol{x} - \hat{\boldsymbol{x}}\| \leq \varepsilon\}$ around the true input $\hat{\boldsymbol{x}}$ is commonly used for the input set $\mathcal{X}$ to evaluate uncertainty, where $\|\cdot\|$ is the ℓ_1-norm or ℓ_2-norm, and $\varepsilon \in \mathbb{R}$ is a given value. The problem in [31], while focusing on the tightness of the SDP relaxation in single-layer NNs, was not related to safety verification. In this paper, we present tightness conditions for the SDP relaxation applied to safety verification, particularly, in the estimation of the minimum safe specification set.

This paper is organized as follows. In Section 2, we describe SDP relaxations of QCQPs and the collinearlity-based tightness condition in Zhang [31]. Section 3 gives a concrete formulation of DeepSDP and discusses its tightness as the SDP relaxation. Sections 4 and 5 include the main results of this paper. In Section 4, we provide a tightness condition for DeepSDP (3) in case of a single-neuron NN. In Section 5, we derive tightness conditions for DeepSDP (3) with respect to two different shapes of input sets. We finally conclude in Section 6.

2 Preliminaries

2.1 Notation

The symbol $\mathbb{R}^n_+$ denotes the set of nonnegative vectors in $\mathbb{R}^n$. Let $\mathbf{0} \in \mathbb{R}^n$ and $\mathbf{1} \in \mathbb{R}^n$ be the zero vector and the vector of all ones. We also let $\boldsymbol{e}^i \in \mathbb{R}^n$ be the ith unit vector of an appropriate length, *i.e.,* $(\boldsymbol{e}^i)_i = 1$ and $(\boldsymbol{e}^i)_k = 0$ for all $k \neq i$. We denote by $\mathbb{S}^n$ the set of the $n \times n$ symmetric matrices. For any symmetric matrices $A, B \in \mathbb{S}^n$, $A \bullet B$ denote the Frobenius inner product of A and B defined as $A \bullet B \coloneqq \mathrm{tr}(AB) = \sum_{i=1}^n \sum_{j=1}^n A_{ij}B_{ij}$. For a subset $S \subset \Omega$, $\mathbf{1}_S(\cdot)$ denotes the indicator function of S, *i.e.,* $\mathbf{1}_S(x) = 1$ if $x \in S$; $\mathbf{1}_S(x) = 0$ otherwise. A matrix $\mathrm{diag}(\boldsymbol{x})$ denotes a diagonal matrix whose diagonal elements are $\boldsymbol{x}$.

Let $N \coloneqq \sum_{k=1}^L n_k$ be the total number of neurons in the entire network. If $\mathcal{X} \subseteq \mathbb{R}^{n_0}$ is a given input set of NN (1), the output of f on $\mathcal{X}$ is $\mathcal{Y} \coloneqq \{f(\boldsymbol{x}^0) \in \mathbb{R}^{n_{L+1}} \,|\, \boldsymbol{x}^0 \in \mathcal{X}\}$. Moreover, for the input set $\mathcal{X}$ in (1), the sets $\mathcal{X}_0, \ldots, \mathcal{X}_L$ are defined by $\mathcal{X}_0 \coloneqq \mathcal{X}$ and $\mathcal{X}_{k+1} \coloneqq \left\{\phi(W^k\boldsymbol{x}^k + \boldsymbol{b}^k) \,\middle|\, \boldsymbol{x}^k \in \mathcal{X}_k\right\}$.

2.2 Tight SDP relaxations of QCQPs and strong duality

We consider an inequality standard form QCQP:

$$\begin{array}{ll} \min\limits_{\boldsymbol{x}} & \boldsymbol{x}^{\mathrm{T}}Q^0\boldsymbol{x} + 2(\boldsymbol{q}^0)^{\mathrm{T}}\boldsymbol{x} \\ \text{s.t.} & \boldsymbol{x}^{\mathrm{T}}Q^k\boldsymbol{x} + 2(\boldsymbol{q}^k)^{\mathrm{T}}\boldsymbol{x} \leq b_k, \quad k = 1, \ldots, m, \\ & \boldsymbol{x} \in \mathbb{R}^n. \end{array} \tag{4}$$

As QCQPs are generally nonconvex, a well-known approach for obtaining a good approximate optimal value is to use SDP relaxations. The SDP relaxation of (4) can be expressed as:

$$\begin{array}{ll} \min\limits_{\boldsymbol{x}, X} & Q^0 \bullet X + 2(\boldsymbol{q}^0)^{\mathrm{T}}\boldsymbol{x} \\ \text{s.t.} & Q^k \bullet X + 2(\boldsymbol{q}^k)^{\mathrm{T}}\boldsymbol{x} \leq b_k, \quad k = 1, \ldots, m, \\ & \begin{bmatrix} 1 & \boldsymbol{x}^{\mathrm{T}} \\ \boldsymbol{x} & X \end{bmatrix} \succeq O, \end{array} \tag{5}$$

where X is a $n \times n$ symmetric matrix. The optimal value θ of (4) is generally bounded by the optimal value ζ of (5) from below, *i.e.,* $\theta \geq \zeta$. If $\theta = \zeta$, we say that the SDP relaxation (5) is tight (or equivalently, exact). It is well known that (5) is tight if and only if (5) has a rank-1 solution. A great deal of studies has been conducted for the tight SDP relaxation as the exact optimal value and/or solution of (4) can be computed in polynomial time with the SDP relaxation [1, 2, 3, 6, 18, 20, 26, 29].

The dual problem of (5) is

$$\begin{array}{ll} \max\limits_{\boldsymbol{\xi}, \psi} & -\boldsymbol{b}^{\mathrm{T}}\boldsymbol{\xi} + \psi \\ \text{s.t.} & \begin{bmatrix} -\psi & (\boldsymbol{q}^0)^{\mathrm{T}} \\ \boldsymbol{q}^0 & Q^0 \end{bmatrix} + \sum\limits_{k=1}^m \xi_k \begin{bmatrix} 0 & (\boldsymbol{q}^k)^{\mathrm{T}} \\ \boldsymbol{q}^k & Q^k \end{bmatrix} \succeq O, \\ & \boldsymbol{\xi} \geq \mathbf{0}. \end{array} \tag{6}$$

The optimal values of SDP (5) and its dual problem (6) generally do not coincide. The following lemma is used in [3] for strong duality, and it is based on the sufficient conditions in [19, Corollary 4.3].

Lemma 2.1. [3, Lemma 3.3] *If the pair of* (5) *and* (6) *satisfies both conditions:*

(i) both (5) *and* (6) *have optimal solutions; and*

(ii) the set of optimal solutions for (6) *is bounded,*

then strong duality holds between (5) *and* (6)*, i.e., they have optimal solutions and their optimal values are finite and equal.*

By Lemma 2.1, we see that if (5) is tight for (6) and strong duality holds between (6) and (5), then (6) is also tight for (6). In Sections 4 and 5, we demonstrate that the dual of DeepSDP (3) is tight under certain assumptions. Therefore, verifying strong duality between DeepSDP (3) and its dual is crucial when estimating the safety specification set with DeepSDP (3).

2.3 Existence of a rank-1 solution in rank-constrained SDP

This section describes a necessary and sufficient condition in Zhang [31] for ensuring that a given rank-constrained SDP has a rank-1 matrix solution. This condition is related to the tightness of the SDP relaxation, as shown in Lemma 2.3 below. Consider the SDP relaxation (5) with an additional rank constraint $\mathrm{rank}(X) \leq p$ of the following form:

$$\begin{array}{ll} \min\limits_{\boldsymbol{x},X} & Q^0 \bullet X + 2(\boldsymbol{q}^0)^{\mathrm{T}}\boldsymbol{x} \\ \text{s.t.} & Q^k \bullet X + 2(\boldsymbol{q}^k)^{\mathrm{T}}\boldsymbol{x} \leq b_k, \quad k = 1, \ldots, m, \\ & G \coloneqq \begin{bmatrix} 1 & \boldsymbol{x}^{\mathrm{T}} \\ \boldsymbol{x} & X \end{bmatrix} \succeq O, \quad \mathrm{rank}(X) \leq p. \end{array} \tag{7}$$

The problem (7) reduces into the SDP relaxation (5) if $p = n$. Obviously, if (7) has a rank-1 solution for any $p \geq 1$, then (5) also admits a rank-1 solution. Therefore, (5) is a tight relaxation of (4).

The key idea in [31] for analyzing a rank-1 feasible solution of (7) is to express G in terms of the Gram matrix. Fix an arbitrary vector $\boldsymbol{e} \in \mathbb{R}^p$ such that $\|\boldsymbol{e}\| = 1$. New variables $\boldsymbol{u}^1, \ldots, \boldsymbol{u}^n \in \mathbb{R}^p$ are introduced to substitute $\boldsymbol{x}$ and X with

$$G = \left[\begin{array}{c|c} \boldsymbol{e}^{\mathrm{T}}\boldsymbol{e} & \boldsymbol{x}^{\mathrm{T}} \\ \hline \boldsymbol{x} & X \end{array}\right] = \left[\begin{array}{c|ccc} \boldsymbol{e}^{\mathrm{T}}\boldsymbol{e} & \boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^1 & \cdots & \boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^n \\ \hline \boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^1 & (\boldsymbol{u}^1)^{\mathrm{T}}\boldsymbol{u}^1 & \cdots & (\boldsymbol{u}^1)^{\mathrm{T}}\boldsymbol{u}^n \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^n & (\boldsymbol{u}^n)^{\mathrm{T}}\boldsymbol{u}^1 & \cdots & (\boldsymbol{u}^n)^{\mathrm{T}}\boldsymbol{u}^n \end{array}\right].$$

The rank-1 condition $\mathrm{rank}(G) = 1$ holds if the vectors $\boldsymbol{u}^1, \ldots, \boldsymbol{u}^n$ and $\boldsymbol{e}$ are all collinear.

Definition 2.2. *The vectors* $\boldsymbol{u}^1, \ldots, \boldsymbol{u}^n$ *and* $\boldsymbol{e}$ *are all collinear if* $\left|\boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^i\right| = \|\boldsymbol{u}^i\|$ *for all* $i \in \{1, \ldots, n\}$.

By substituting G in (7) with the above Gram-matrix representation, (7) can be reformulated as a nonconvex optimization problem in variables $\boldsymbol{u}^1, \ldots, \boldsymbol{u}^n \in \mathbb{R}^p$:

$$\begin{array}{ll} \min\limits_{\boldsymbol{u}^1,\ldots,\boldsymbol{u}^n} & \sum\limits_{i=1}^{n}\sum\limits_{j=1}^{n} Q_{ij}^0 \left(\boldsymbol{u}^i\right)^{\mathrm{T}} \boldsymbol{u}^j + 2\sum\limits_{i=1}^{n} q_i^0 \boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^i \\ \text{s.t.} & \sum\limits_{i=1}^{n}\sum\limits_{j=1}^{n} Q_{ij}^k \left(\boldsymbol{u}^i\right)^{\mathrm{T}} \boldsymbol{u}^j + 2\sum\limits_{i=1}^{n} q_i^k \boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^i \leq b_k, \quad k = 1, \ldots, m, \\ & \boldsymbol{u}^1, \ldots, \boldsymbol{u}^n \in \mathbb{R}^p. \end{array} \tag{8}$$

The following lemma allows us to determine whether (7) is a tight relaxation for the original problem.

Lemma 2.3. *Fix $\boldsymbol{e} \in \mathbb{R}^p$ with $\|\boldsymbol{e}\| = 1$. Then, the following two conditions are equivalent.*

(i) the problem (7) has a rank-1 matrix solution.

(ii) the problem (8) has an optimal solution $(\boldsymbol{u}^1)^*, \ldots, (\boldsymbol{u}^n)^*$ which are collinear with $\boldsymbol{e}$.

When either of (i) and (ii) holds, (7) is a tight relaxation for (4). In addition, $\boldsymbol{x}^*$ and $(\boldsymbol{x}^*, X^*)$ are optimal solutions of (4) and (7), respectively, where $\boldsymbol{x}^* \coloneqq \left[\boldsymbol{e}^{\mathrm{T}}(\boldsymbol{u}^1)^* \quad \cdots \quad \boldsymbol{e}^{\mathrm{T}}(\boldsymbol{u}^n)^*\right]^{\mathrm{T}}$ and $X^* \coloneqq \boldsymbol{x}^* \left(\boldsymbol{x}^*\right)^{\mathrm{T}}$.

Proof. The equivalence between (i) and (ii) follows from [31, Theorem A.2]. Since (7) has a rank-1 solution, it is a tight relaxation for (4), as mentioned in Section 2.2. □

The direction $\boldsymbol{e} \in \mathbb{R}^p$ can be fixed for the discussion here, as described in Appendix A of [31]. For instance, let us consider a case that, after solving (8) with $\boldsymbol{e} = \bar{\boldsymbol{e}}$, we wish to have a rank-1 solution of (8) with $\boldsymbol{e} = \hat{\boldsymbol{e}}$. We can find an orthonormal matrix $U \in \mathbb{R}^{p\times p}$ such that $\hat{\boldsymbol{e}} = U\bar{\boldsymbol{e}}$ by the Gram-Schmidt process. Let $\bar{\boldsymbol{u}}^1, \ldots, \bar{\boldsymbol{u}}^n$ be an optimal solution of (8) with $\boldsymbol{e} = \bar{\boldsymbol{e}}$. Then, for all pair $(i, j) \in \{1, \ldots, n\}^2$, we have $\left(\bar{\boldsymbol{u}}^i\right)^{\mathrm{T}} \bar{\boldsymbol{u}}^j = \left(U\bar{\boldsymbol{u}}^i\right)^{\mathrm{T}} U\bar{\boldsymbol{u}}^j$ and $\bar{\boldsymbol{e}}^{\mathrm{T}}\bar{\boldsymbol{u}}^j = \left(U\bar{\boldsymbol{e}}\right)^{\mathrm{T}} U\bar{\boldsymbol{u}}^j = \hat{\boldsymbol{e}}^{\mathrm{T}} U\bar{\boldsymbol{u}}^j$. By taking $\hat{\boldsymbol{u}}^i \coloneqq U\bar{\boldsymbol{u}}^i$ for all $j \in \{1, \ldots, n\}$, an optimal solution $(\hat{\boldsymbol{u}}^1, \ldots, \hat{\boldsymbol{u}}^n)$ of (8) with $\boldsymbol{e} = \bar{\boldsymbol{e}}$ is obtained. Therefore, we may assume that $\boldsymbol{e} = \boldsymbol{e}^1$ in the proofs without loss of generality throughout the paper.

3 Tight SDPs for NNs

In this section, we first describe the safety specification set of DeepSDP (3). The connection between the tightness of (3) and its accuracy is presented in Section 3.4. Throughout, DeepSDP (3) with the standard ReLU activation function ϕ is discussed, *i.e.*, $\phi(\boldsymbol{x})_i = \max\{0, x_i\}$ holds.

3.1 Safety specification sets

To verify the safety of an input set $\mathcal{X}$, it is important to analyze the subset relationship between a given set S_y and the image of $\mathcal{X}$ under f, given by

$$\mathcal{Y} \coloneqq f(\mathcal{X}) = \left\{ f(\boldsymbol{x}^0) \,\middle|\, \boldsymbol{x}^0 \in \mathcal{X} \right\}.$$

Let $\mathcal{X}$ be an input set that either contains uncertainty or represents an adversarial attack that we wish to evaluate. We define a safety specification set $S_y \subset \mathbb{R}^{n_{L+1}}$ such that, for every $\boldsymbol{x}^0 \in \mathcal{X}$, it represents the range over which the output value $f(\boldsymbol{x}^0)$ can be correctly interpreted. If $\mathcal{Y} \subseteq S_y$, then for any $\boldsymbol{x}^0 \in \mathcal{X}$, the value $f(\boldsymbol{x}^0)$ is unaffected by uncertainty or adversarial attacks. Thus, the inputs $\mathcal{X}$ for the NN is considered safe. To represent candidate safety specification sets, DeepSDP (3) uses the matrix set $\mathcal{S} \subseteq \mathbb{S}^{1+n_0+n_{L+1}}$, where $\mathbb{S}^{1+n_0+n_{L+1}}$ is the space of symmetric matrices of dimension $1 + n_0 + n_{L+1}$.

3.2 Quadratic constraints in DeepSDP (3)

We describe the quadratic constraints that constitute DeepSDP (3). In Section 3.2.1, we define the matrix set $\mathcal{P}_{\mathcal{X}}$ used in (3). Quadratic constraints to encode the ReLU functions ϕ are discussed in Section 3.2.2. Section 3.2.3 formulates valid cuts which strengthen constraints in DeepSDP (3). (See [12] for details.)

3.2.1 Quadratic representation of input sets

For the nonempty input set $\mathcal{X} \subseteq \mathbb{R}^{n_0}$ of the NN, $\mathcal{P}_{\mathcal{X}} \subset \mathbb{S}^{n_0+1}$ is defined as the set of all symmetric indefinite matrices $P \in \mathbb{S}^{n_0+1}$ such that

$$\begin{bmatrix} 1 \\ \boldsymbol{x}^0 \end{bmatrix}^{\mathrm{T}} P \begin{bmatrix} 1 \\ \boldsymbol{x}^0 \end{bmatrix} \leq 0 \quad \text{for all } \boldsymbol{x}^0 \in \mathcal{X}. \tag{9}$$

Obviously,

$$\mathcal{X} \subseteq \bigcap_{P \in \mathcal{P}_{\mathcal{X}}} \left\{ \boldsymbol{x}^0 \in \mathbb{R}^{n_0} \;\middle|\; \begin{bmatrix} 1 \\ \boldsymbol{x} \end{bmatrix}^{\mathrm{T}} P \begin{bmatrix} 1 \\ \boldsymbol{x} \end{bmatrix} \leq 0 \right\}, \tag{10}$$

and $\mathcal{P}_{\mathcal{X}}$ serves as an over-approximation to the input set $\mathcal{X}$ using an infinite number of quadratic constraints. In [12], the set $\mathcal{P}_{\mathcal{X}}$ for common input set $\mathcal{X}$ was discussed. In particular, for a hyper-rectangle $\mathcal{X} := \{\boldsymbol{x} \in \mathbb{R}^{n_0} \,|\, \underline{\boldsymbol{x}} \leq \boldsymbol{x} \leq \overline{\boldsymbol{x}}\}$, where $\underline{\boldsymbol{x}} \in \mathbb{R}^{n_0}$ and $\overline{\boldsymbol{x}} \in \mathbb{R}^{n_0}$ are the given lower and upper bounds, the over-approximation set $\mathcal{P}_{\mathcal{X}}$ can be described by

$$\mathcal{P}_{\mathcal{X}} = \left\{ \begin{bmatrix} 2\underline{\boldsymbol{x}}^{\mathrm{T}} \operatorname{diag}(\boldsymbol{\gamma}) \overline{\boldsymbol{x}} & -(\underline{\boldsymbol{x}} + \overline{\boldsymbol{x}})^{\mathrm{T}} \operatorname{diag}(\boldsymbol{\gamma}) \\ -\operatorname{diag}(\boldsymbol{\gamma})(\underline{\boldsymbol{x}} + \overline{\boldsymbol{x}}) & 2\operatorname{diag}(\boldsymbol{\gamma}) \end{bmatrix} \;\middle|\; \boldsymbol{\gamma} \geq \boldsymbol{0} \right\},$$

and (10) holds with equality.

3.2.2 Global constraints

Let $\varphi : \mathbb{R} \to \mathbb{R}$ be the standard ReLU function $\varphi(x) = \max\{0, x\}$. We let $\phi(\boldsymbol{z}) := \begin{bmatrix} \varphi(z_1) & \cdots & \varphi(z_N) \end{bmatrix}^{\mathrm{T}}$ of $\boldsymbol{z} \in \mathbb{R}^N$ since the NN (1) invokes N ReLU activation functions φ to output $f(\boldsymbol{x}^0)$. To simplify the subsequent discussion, we define $\boldsymbol{w}^k := W^k \boldsymbol{x}^k + \boldsymbol{b}^k \in \mathbb{R}^{n_{k+1}}$ for every $k = 0, \ldots, L-1$ and $\boldsymbol{w} := \begin{bmatrix} (\boldsymbol{w}^0)^{\mathrm{T}} & \cdots & (\boldsymbol{w}^{L-1})^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^N$.

As all activation functions in the NN (1) can be expressed in terms of $\phi(\boldsymbol{w})$, we now discuss the constraints between $\boldsymbol{w}$ and $\phi(\boldsymbol{w})$. For all $i = 1, \ldots, N$, the constraint $\varphi(w_i) =$

$\max\{0, w_i\}$ for expressing the standard ReLU activation function can be equivalently transformed into quadratic constraints

$$\varphi(w_i)\left[\varphi(w_i) - w_i\right] = 0, \quad \varphi(w_i) \geq w_i, \quad \varphi(w_i) \geq 0. \tag{11}$$

Moreover, (11) can be equivalently rewritten in the matrix form:

$$\begin{bmatrix} 1 \\ \boldsymbol{w} \\ \phi(\boldsymbol{w}) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} 0 & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & O & -\boldsymbol{e}_i\boldsymbol{e}_i^{\mathrm{T}} \\ \mathbf{0} & -\boldsymbol{e}_i\boldsymbol{e}_i^{\mathrm{T}} & 2\boldsymbol{e}_i\boldsymbol{e}_i^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} 1 \\ \boldsymbol{w} \\ \phi(\boldsymbol{w}) \end{bmatrix} = 0, \quad i = 1, \ldots, N, \tag{12a}$$

$$\begin{bmatrix} 1 \\ \boldsymbol{w} \\ \phi(\boldsymbol{w}) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} 0 & \boldsymbol{e}_i^{\mathrm{T}} & -\boldsymbol{e}_i^{\mathrm{T}} \\ \boldsymbol{e}_i & O & O \\ -\boldsymbol{e}_i & O & O \end{bmatrix} \begin{bmatrix} 1 \\ \boldsymbol{w} \\ \phi(\boldsymbol{w}) \end{bmatrix} \leq 0, \quad i = 1, \ldots, N, \tag{12b}$$

$$\begin{bmatrix} 1 \\ \boldsymbol{w} \\ \phi(\boldsymbol{w}) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} 0 & \mathbf{0}^{\mathrm{T}} & -\boldsymbol{e}_i^{\mathrm{T}} \\ \mathbf{0} & O & O \\ -\boldsymbol{e}_i & O & O \end{bmatrix} \begin{bmatrix} 1 \\ \boldsymbol{w} \\ \phi(\boldsymbol{w}) \end{bmatrix} \leq 0, \quad i = 1, \ldots, N. \tag{12c}$$

In [12], the constraints (12) were employed and referred to as global quadratic constraints.

We mention that Fazlyab et al. [12] also proposed weaker inequalities than (12), called local quadratic constraints. The local quadratic constraints are not dealt with in this work, since their descriptions require more assumptions on the input space $\mathcal{X}$ and the activation function.

3.2.3 Valid cuts

Let $\boldsymbol{w} = \left[(\boldsymbol{w}^0)^{\mathrm{T}} \ \cdots \ (\boldsymbol{w}^{L-1})^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^N$ be a column vector. For the standard ReLU activate function $\varphi(x) = \max\{0, x\}$, the inequalities

$$\left[\varphi(w_j) - \varphi(w_i)\right]\left[\varphi(w_j) - \varphi(w_i) - (w_j - w_i)\right] \leq 0 \tag{13}$$

hold for all i, j $(1 \leq i < j \leq N)$. These valid cuts that couple between neurons were called as repeated nonlinearities in [12], since the same function φ repeatedly appears in the NN for many neurons. The inequality (13) can be equivalently written as

$$\begin{bmatrix} 1 \\ \boldsymbol{w} \\ \phi(\boldsymbol{w}) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} 0 & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & O & -(\boldsymbol{e}_i - \boldsymbol{e}_j)(\boldsymbol{e}_i - \boldsymbol{e}_j)^{\mathrm{T}} \\ \mathbf{0} & -(\boldsymbol{e}_i - \boldsymbol{e}_j)(\boldsymbol{e}_i - \boldsymbol{e}_j)^{\mathrm{T}} & 2(\boldsymbol{e}_i - \boldsymbol{e}_j)(\boldsymbol{e}_i - \boldsymbol{e}_j)^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} 1 \\ \boldsymbol{w} \\ \phi(\boldsymbol{w}) \end{bmatrix} \leq 0, \tag{14}$$

for all i, j $(1 \leq i < j \leq N)$.

The computation of $f(\boldsymbol{x}^0)$ of the NN can be accelerated by computing φ in parallel. While increasing the computational efficiency may be important, it is secondary to ensuring the solution accuracy in this paper. Our primary focus is whether the constraints can serve as valid cuts to improve the accuracy. In Sections 4 and 5, we prove that DeepSDP (3) is tight under certain conditions without these valid cuts (14). This implies that the tightness can be maintained even when valid cuts are added to (3).

3.3 Formulation of DeepSDP

In this subsection, we present the formulation of DeepSDP (3). We now consider a QCQP with the global quadratic constraints (see Section 3.2.2) and the repeated nonlinearities (see Section 3.2.3):

$$\begin{aligned} \min \quad & \begin{bmatrix} 1 \\ \boldsymbol{x}^0 \\ W^L\boldsymbol{x}^L + \boldsymbol{b}^L \end{bmatrix}^{\mathrm{T}} H \begin{bmatrix} 1 \\ \boldsymbol{x}^0 \\ W^L\boldsymbol{x}^L + \boldsymbol{b}^L \end{bmatrix} \\ \text{s.t.} \quad & \begin{bmatrix} 1 \\ \boldsymbol{x}^0 \end{bmatrix}^{\mathrm{T}} P \begin{bmatrix} 1 \\ \boldsymbol{x}^0 \end{bmatrix} \leq 0, \ \forall P \in \mathcal{P}_{\mathcal{X}}, \\ & (12), \ (14), \ \boldsymbol{w} \coloneqq \overline{W} \begin{bmatrix} \boldsymbol{x}^0 \\ \vdots \\ \boldsymbol{x}^L \end{bmatrix} + \overline{\boldsymbol{b}}, \ \phi(\boldsymbol{w}) \coloneqq \overline{E} \begin{bmatrix} \boldsymbol{x}^0 \\ \vdots \\ \boldsymbol{x}^L \end{bmatrix}, \\ & \boldsymbol{x}^k \in \mathbb{R}^{n_k}, \ k = 0, \dots, L, \end{aligned} \tag{15}$$

where $H \in \mathbb{S}^{1+n_0+n_{L+1}}$ is a given coefficient matrix for finding an extreme point of the output space $\mathbb{R}^{n_{L+1}}$ and

$$\overline{W} \coloneqq \left[\begin{array}{ccc|c} W^0 & & O & O \\ & \ddots & & \vdots \\ O & & W^{L-1} & O \end{array}\right], \quad \overline{\boldsymbol{b}} \coloneqq \begin{bmatrix} \boldsymbol{b}^0 \\ \vdots \\ \boldsymbol{b}^{L-1} \end{bmatrix}, \quad \overline{E} \coloneqq \begin{bmatrix} E^1 \\ \vdots \\ E^L \end{bmatrix}.$$

Here, $E^i \in \mathbb{R}^{n_i \times (n_0+\cdots+n_L)}$ is a matrix such that $E^i \left[\boldsymbol{x}^0; \dots; \boldsymbol{x}^L\right] = \boldsymbol{x}^i$, *i.e.,* E^i extracts $\boldsymbol{x}^i$ from $\left[\boldsymbol{x}^0; \dots; \boldsymbol{x}^L\right]$.

To simplify (15) by representing the variable vectors in the constraints as $[1; \boldsymbol{x}^0; \cdots; \boldsymbol{x}^L]$, we define three matrix functions:

$$M_{\mathrm{in}}(P) = \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & E^0 \end{bmatrix}^{\mathrm{T}} P \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & E^0 \end{bmatrix}, \ M_{\mathrm{mid}}(Q) = \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} \\ \overline{\boldsymbol{b}} & \overline{W} \\ \mathbf{0} & \overline{E} \end{bmatrix}^{\mathrm{T}} Q \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} \\ \overline{\boldsymbol{b}} & \overline{W} \\ \mathbf{0} & \overline{E} \end{bmatrix},$$

$$M_{\mathrm{out}}(S) = \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & E^0 \\ \boldsymbol{b}^L & W^L E^L \end{bmatrix}^{\mathrm{T}} S \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & E^0 \\ \boldsymbol{b}^L & W^L E^L \end{bmatrix}.$$

Then, the SDP relaxation of (15) is

$$
\begin{array}{ll}
\min & M_{\rm out}(H) \bullet G \\
\text{s.t.} & M_{\rm in}(P) \bullet G \le 0,\ \forall P \in \mathcal{P}_{\mathcal{X}}, \\
& M_{\rm mid}\left(\begin{bmatrix} 0 & \mathbf{0}^{\rm T} & \mathbf{0}^{\rm T} \\ \mathbf{0} & O & -\boldsymbol{e}_i\boldsymbol{e}_i^{\rm T} \\ \mathbf{0} & -\boldsymbol{e}_i\boldsymbol{e}_i^{\rm T} & 2\boldsymbol{e}_i\boldsymbol{e}_i^{\rm T} \end{bmatrix}\right) \bullet G = 0, \quad i = 1, \ldots, N, \\
& M_{\rm mid}\left(\begin{bmatrix} 0 & \boldsymbol{e}_i^{\rm T} & -\boldsymbol{e}_i^{\rm T} \\ \boldsymbol{e}_i & O & O \\ -\boldsymbol{e}_i & O & O \end{bmatrix}\right) \bullet G \le 0,\ M_{\rm mid}\left(\begin{bmatrix} 0 & \mathbf{0}^{\rm T} & -\boldsymbol{e}_i^{\rm T} \\ \mathbf{0} & O & O \\ -\boldsymbol{e}_i & O & O \end{bmatrix}\right) \bullet G \le 0, \quad i = 1, \ldots, N, \\
& M_{\rm mid}\left(\begin{bmatrix} 0 & \mathbf{0}^{\rm T} & \mathbf{0}^{\rm T} \\ \mathbf{0} & O & -(\boldsymbol{e}_i - \boldsymbol{e}_j)(\boldsymbol{e}_i - \boldsymbol{e}_j)^{\rm T} \\ \mathbf{0} & -(\boldsymbol{e}_i - \boldsymbol{e}_j)(\boldsymbol{e}_i - \boldsymbol{e}_j)^{\rm T} & 2(\boldsymbol{e}_i - \boldsymbol{e}_j)(\boldsymbol{e}_i - \boldsymbol{e}_j)^{\rm T} \end{bmatrix}\right) \bullet G \le 0, \quad 1 \le i < j \le N, \\
& G \coloneqq \begin{bmatrix} 1 & \boldsymbol{x}^{\rm T} \\ \boldsymbol{x} & X \end{bmatrix} \succeq O, \quad \boldsymbol{x} \in \mathbb{R}^{n_0+N}, \quad X \in \mathbb{S}^{n_0+N}.
\end{array}
\tag{16}
$$

DeepSDP (3) is formulated as the dual problem of (16). Let $\boldsymbol{\lambda}$, $\boldsymbol{\nu}$, $\boldsymbol{\eta}$, and $\boldsymbol{\mu}$ be dual variables for (12a), (12b), (12c), and (14), respectively. The terms associated with $M_{\rm mid}$, among the terms in the Lagrangian function of (16), can be written as follows:

$$
\mathcal{Q}_\phi \coloneqq \left\{ \begin{bmatrix} 0 & Q_{12} & Q_{13} \\ Q_{12}^{\rm T} & Q_{22} & Q_{23} \\ Q_{13}^{\rm T} & Q_{23}^{\rm T} & Q_{33} \end{bmatrix} \;\middle|\; \boldsymbol{\lambda}, \boldsymbol{\nu}, \boldsymbol{\eta} \in \mathbb{R}^N_+,\ \mu_{ij} \ge 0,\ 1 \le i < j \le N \right\} \subset \mathbb{S}^{1+2N}
$$

where

$$
\left.
\begin{array}{c}
Q_{12} \coloneqq \boldsymbol{\nu}^{\rm T}, \quad Q_{13} \coloneqq -\boldsymbol{\nu}^{\rm T} - \boldsymbol{\eta}^{\rm T}, \quad Q_{22} \coloneqq O, \quad Q_{23} \coloneqq -\left(\operatorname{diag}(\boldsymbol{\lambda}) + T\right), \\
Q_{33} \coloneqq 2\left(\operatorname{diag}(\boldsymbol{\lambda}) + T\right), \quad T \coloneqq \displaystyle\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \mu_{ij}(\boldsymbol{e}_i - \boldsymbol{e}_j)(\boldsymbol{e}_i - \boldsymbol{e}_j)^{\rm T}.
\end{array}
\right\}
\tag{17}
$$

With $\mathcal{X}$, $\mathcal{Q}_\phi$, and $\mathcal{S}$ defined in Section 3.1, DeepSDP (3) can be obtained. We can also use the following theorem in [12] to determine the safety specification set S_y of a given NN.

Theorem 3.1. [12, Theorem 2] *Let (P, Q, S) be a feasible solution of (3). Then, for any $\boldsymbol{x}^0 \in \mathcal{X}$, it holds that*

$$
\begin{bmatrix} 1 \\ \boldsymbol{x}^0 \\ f(\boldsymbol{x}^0) \end{bmatrix}^{\rm T} S \begin{bmatrix} 1 \\ \boldsymbol{x}^0 \\ f(\boldsymbol{x}^0) \end{bmatrix} \ge 0. \tag{18}
$$

Theorem 3.1 implies that once we find an optimal solution (P^*, Q^*, S^*) of (3), the output set $\mathcal{Y}$ is a subset of the safety specification set

$$
S_y \coloneqq \left\{ \boldsymbol{y} \in \mathbb{R}^{n_{L+1}} \;\middle|\; \begin{bmatrix} 1 \\ \boldsymbol{x}^0 \\ \boldsymbol{y} \end{bmatrix}^{\rm T} S^* \begin{bmatrix} 1 \\ \boldsymbol{x}^0 \\ \boldsymbol{y} \end{bmatrix} \ge 0 \text{ for all } \boldsymbol{x}^0 \in \mathcal{X} \right\}.
$$

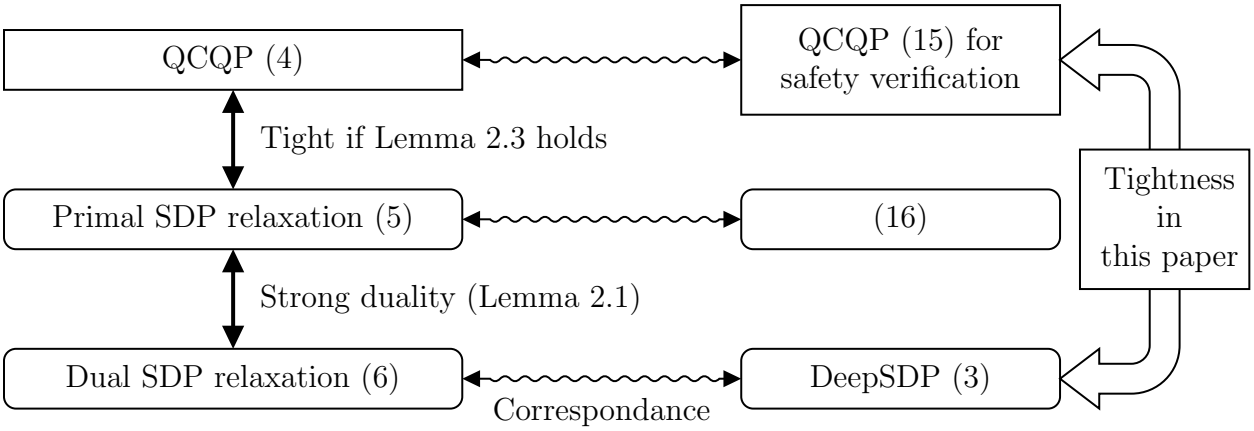


Figure 1: Correspondence between QCQPs and DeepSDP for the tightness. The solid lines display the equivalent optimal value of the problems. The wavy lines illustrate the corresponding relationship between the two problems.

3.4 DeepSDP (3) and tight SDP relaxations

The tight SDP relaxation, discussed in the previous subsection, plays a crucial role for the accuracy of a solution to DeepSDP (3). If the objective function h of (15) is linear or quadratic, the problem (15) can be regarded as a QCQP (4). In fact, all the constraints of (15) are at most degree 2, and each can be represented as

$$\begin{bmatrix}1\\ \boldsymbol{x}\end{bmatrix}^{\mathrm{T}} \begin{bmatrix}\tilde{b} & \tilde{\boldsymbol{q}}^{\mathrm{T}}\\ \tilde{\boldsymbol{q}} & \tilde{Q}\end{bmatrix} \begin{bmatrix}1\\ \boldsymbol{x}\end{bmatrix} \leq 0$$

with $\boldsymbol{x} \coloneqq \left[\boldsymbol{x}^0; \cdots; \boldsymbol{x}^L\right] \in \mathbb{R}^{n_0+N}$, appropriate coefficients $\tilde{b} \in \mathbb{R}$, $\tilde{\boldsymbol{q}} \in \mathbb{R}^{n_0+N}$, and $\tilde{Q} \in \mathbb{S}^{n_0+N}$. As DeepSDP (3) is the dual problem of the SDP relaxation (16) of QCQP (15), the relationship between (15) and (3) is analogous to the relationship between the standard QCQP (4) and its dual SDP relaxation (6). Figure 1 shows the correspondence between (15) and (3). In the subsequent discussion, we refer to (16) as the primal SDP relaxation of (15), and distinguish it from the dual SDP relaxation, DeepSDP (3). When strong duality holds between these relaxations (3) and (16), (3) is a tight relaxation of (15) if and only if (16) is also a tight relaxation of it, as shown in Figure 1. We focus on the tightness of DeepSDP (3) for (15).

Since the cardinality of $\mathcal{P}_{\mathcal{X}}$ can be infinite, selecting the appropriate constraints is crucial. In subsequent discussion, the structure of $\mathcal{P}_{\mathcal{X}}$ can be simplified by restricting the input set $\mathcal{X}$ to an ellipsoid or a rectangle. With a given center $\hat{\boldsymbol{x}}$ and a radius ε, a ellipsoidal input set can be represented by $\{\boldsymbol{x} \in \mathbb{R}^{n_0} \mid \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2 \leq \varepsilon\}$; and a rectangular input set by $\{\boldsymbol{x} \in \mathbb{R}^{n_0} \mid \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_1 \leq \varepsilon\}$.

Other factors can also affect the accuracy of DeepSDP (3). For example, if $\mathcal{S}$ does not contain the necessary matrices to represent the true safety specification set, the recovered set will not be minimal. We do not address other factors, such as the selection of $\mathcal{S}$, in this paper.

4 Tightness of DeepSDP (3) for a single-neuron ReLU network

We first analyze the tightness of DeepSDP (3) with a single-neuron case of the following:

$$f(x) = \varphi\left(x + b^0\right), \quad x \in \mathcal{X} \subseteq \mathbb{R},$$

which is equivalent to the NN (1) with $L = 1$, $n_0 = n_1 = 1$, $W^0 = W^1 = 1$ and $\boldsymbol{b}^1 = 0$. The output $y := f(x^0)$ is x^1 in (1). In this case, the smallest safety specification set is clearly $\mathcal{X} \cap \mathbb{R}_+$ as only one ReLU activation function is applied in the NN. The following assumptions hold throughout this section.

Assumption 4.1. *The input set $\mathcal{X}$ is a closed interval, i.e., $\mathcal{X} = \{x \,|\, \underline{x} \le x \le \bar{x}\}$ for some $\underline{x}$ and $\bar{x}$.*

Assumption 4.2. *The candidate of the safety specification set S_y on the output space is a polytope.*

Under these assumptions, the safety set S_y can be represented by a closed interval $[\underline{d}, \overline{d}]$, which can be further rewritten as $\{y \in \mathbb{R} \,|\, y - \underline{d} \ge 0\} \cap \left\{y \in \mathbb{R} \,\middle|\, -y - (-\overline{d}) \ge 0\right\}$. For $c \in \{-1, +1\}$ and $d \in \mathbb{R}$, to determine whether $f(x)$ lies within $\{y \in \mathbb{R} \,|\, cy - d \ge 0\}$ for all $x \in \mathcal{X}$, we check if the following equation holds:

$$\begin{bmatrix} 1 \\ x \\ f(x) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} -2d & 0 & c \\ 0 & 0 & 0 \\ c & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ f(x) \end{bmatrix} \ge 0.$$

By Assumptions 4.1 and 4.2, the matrix sets $\mathcal{P}_{\mathcal{X}}$, $\mathcal{Q}_{\phi}$, and $\mathcal{S}$ can be written as

$$\mathcal{P}_{\mathcal{X}} = \left\{ \begin{bmatrix} 2\underline{x}\bar{x}\gamma & -(\underline{x} + \bar{x})\,\gamma \\ -(\underline{x} + \bar{x})\,\gamma & 2\gamma \end{bmatrix} \,\middle|\, \gamma \ge 0 \right\},$$

$$\mathcal{Q}_{\phi} = \left\{ \begin{bmatrix} 2b^0\nu & \nu & -\nu - \eta - b^0\lambda \\ \nu & 0 & -\lambda \\ -\nu - \eta - b^0\lambda & -\lambda & 2\lambda \end{bmatrix} \,\middle|\, \lambda, \nu, \eta \in \mathbb{R}_+ \right\},$$

$$\mathcal{S} = \left\{ \begin{bmatrix} -2d & 0 & c \\ 0 & 0 & 0 \\ c & 0 & 0 \end{bmatrix} \,\middle|\, d \in \mathbb{R} \right\},$$

with $\gamma, \lambda, \nu, \eta \in \mathbb{R}_+$ and $d \in \mathbb{R}$. Since there is only one neuron, no constraints exist for the relationship between two neurons, and the repeated nonlinearities do not appear, *i.e.*, $T = 0$ in (3). Consequently, DeepSDP (3) for a single-neuron NN can be described as

$$\begin{aligned} \max \quad & 2d \\ \text{s.t.} \quad & \begin{bmatrix} 2\underline{x}\bar{x}\gamma & -(\underline{x} + \bar{x})\,\gamma & 0 \\ -(\underline{x} + \bar{x})\,\gamma & 2\gamma & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 2b^0\nu & \nu & -\nu - \eta - b^0\lambda \\ \nu & 0 & -\lambda \\ -\nu - \eta - b^0\lambda & -\lambda & 2\lambda \end{bmatrix} \\ & \quad + \begin{bmatrix} -2d & 0 & c \\ 0 & 0 & 0 \\ c & 0 & 0 \end{bmatrix} \succeq O, \quad \gamma, \lambda, \nu, \eta \in \mathbb{R}_+, \quad d \in \mathbb{R}. \end{aligned} \tag{19}$$

Here, we maximize d under a fixed c as DeepSDP (3) is to find the minimum interval (the minimal safety specification set).

By demonstrating that (19) is a tight SDP relaxation in the subsequent discussion, we can obtain an optimal solution d^*, which is also an optimal solution of the QCQP formulation (15).

4.1 Two-stage formulation

We derive a nonconvex optimization problem of the form (8) in this section.

The dual of the SDP (19) is:

$$\begin{array}{ll}\min\limits_{x^0,x^1,X} & \begin{bmatrix}0&0&c\\0&0&0\\c&0&0\end{bmatrix}\bullet G\\ \text{s.t.} & \begin{bmatrix}2\underline{x}\bar{x} & -(\underline{x}+\bar{x}) & 0\\ -(\underline{x}+\bar{x}) & 2 & 0\\ 0 & 0 & 0\end{bmatrix}\bullet G\le 0,\ \begin{bmatrix}-2&0&0\\0&0&0\\0&0&0\end{bmatrix}\bullet G\le -2,\\ & \begin{bmatrix}2b^0&1&-1\\1&0&0\\-1&0&0\end{bmatrix}\bullet G\le 0,\ \begin{bmatrix}0&0&-1\\0&0&0\\-1&0&0\end{bmatrix}\bullet G\le 0,\ \begin{bmatrix}0&0&-b^0\\0&0&-1\\-b^0&-1&2\end{bmatrix}\bullet G\le 0,\\ & G\coloneqq\left[\begin{array}{c|cc}1&x^0&x^1\\\hline x^0&\multicolumn{2}{c}{\multirow{2}{*}{X}}\\x^1&&\end{array}\right]\in\mathbb{S}^3_+,\quad x^0,x^1\in\mathbb{R},\quad X\in\mathbb{S}^2_+.\end{array}\tag{20}$$

Note that (20) is a rank-constrained SDP (7) with $p=3$, and corresponds to an SDP relaxation of the following QCQP:

$$\begin{array}{ll}\min\limits_{x^0,x^1} & cx^1\\ \text{s.t.} & (\bar{x}-x^0)(x^0-\underline{x})\ge 0,\ -2\le -2,\\ & x^1\ge x^0,\ x^1\ge 0,\ x^1(x^0-x^1)\ge 0.\end{array}$$

Since (5) is the primal SDP relaxation and (6) is the dual SDP relaxation, we now call (20) a primal problem and DeepSDP (19) a dual problem.

Let $\boldsymbol{e}\in\mathbb{R}^3$ such that $\|\boldsymbol{e}\|=1$. For $\boldsymbol{u}^1,\boldsymbol{v}^1\in\mathbb{R}^3$, we let $x^0=\boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^1$ and $x^1=\boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1$. We substitute

$$\left[\begin{array}{c|cc}1&x^0&x^1\\\hline x^0&\multicolumn{2}{c}{\multirow{2}{*}{X}}\\x^1&&\end{array}\right]=\left[\begin{array}{c|cc}\boldsymbol{e}^{\mathrm{T}}\boldsymbol{e}&\boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^1&\boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1\\\hline \boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^1&(\boldsymbol{u}^1)^{\mathrm{T}}\boldsymbol{u}^1&(\boldsymbol{u}^1)^{\mathrm{T}}\boldsymbol{v}^1\\ \boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1&(\boldsymbol{v}^1)^{\mathrm{T}}\boldsymbol{u}^1&(\boldsymbol{v}^1)^{\mathrm{T}}\boldsymbol{v}^1\end{array}\right],$$

into (20) as shown in Section 2.3. Then, we obtain the following nonlinear optimization problem:

$$\begin{array}{ll}\min\limits_{\boldsymbol{u}^1,\boldsymbol{v}^1} & c\boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1\\ \text{s.t.} & \boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1\ge\boldsymbol{e}^{\mathrm{T}}(\boldsymbol{u}^1+b^0\boldsymbol{e}),\quad \boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1\ge 0,\quad \|\boldsymbol{v}^1\|_2^2\le(\boldsymbol{u}^1+b^0\boldsymbol{e})^{\mathrm{T}}\boldsymbol{v}^1,\\ & \|\boldsymbol{u}^1-\hat{x}\boldsymbol{e}\|_2^2\le\rho,\end{array}\tag{21}$$

where $\hat{x} = (\underline{x} + \bar{x})/2$, and $\rho = \hat{x}^2 - 2\underline{x}\bar{x}$. The problems (20) and (21) correspond to (7) and (8), respectively. The tightness of (19) can be determined by testing whether all the optimal solutions (21) are collinear with $\boldsymbol{e}$.

Analyzing directly the collinearity of all optimal solutions is, however, challenging. To address this difficulty, we decompose (21) into two-stage problems, and then combine their optimal solutions. Using the approach in [31], the first-stage problem of the decomposed problem can be described as

$$\begin{array}{ll} \min\limits_{\boldsymbol{v}^1} & c\boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1 \\ \text{s.t.} & \boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1 \geq 0, \quad \Phi(\boldsymbol{v}^1) \leq \rho, \end{array} \tag{22}$$

and the second-stage problem

$$\begin{array}{rll} \Phi(\boldsymbol{z}) \coloneqq \min\limits_{\boldsymbol{u}^1} & \|\boldsymbol{u}^1 - \hat{x}\boldsymbol{e}\|_2^2 \\ \text{s.t.} & \boldsymbol{e}^{\mathrm{T}}\boldsymbol{z} \geq \boldsymbol{e}^{\mathrm{T}}\left(\boldsymbol{u}^1 + b^0\boldsymbol{e}\right), & \|\boldsymbol{z}\|_2^2 \leq \left(\boldsymbol{u}^1 + b^0\boldsymbol{e}\right)^{\mathrm{T}}\boldsymbol{z}. \end{array} \tag{23}$$

We note that the variables of the two-stage problem are different from those in [31]. Specifically, the variable of the first-stage problem (22) is x^1 (the output of NN), while that in [31] is x^0 (the input of NN).

4.2 Analyzing the tightness

Lemma 4.3. *Suppose that $\boldsymbol{e} \in \mathbb{R}^p$ satisfies $\|\boldsymbol{e}\| = 1$, $\boldsymbol{z} \in \mathbb{R}^p$ is a feasible point of (22), $\hat{x} \geq 0$, $b^0 = 0$, and the feasible set of (23) is nonempty. Then, $(\boldsymbol{u}^1)^* \coloneqq \boldsymbol{z}$ is a solution of (23). In addition, $\Phi(\boldsymbol{z}) = \|\boldsymbol{z} - \hat{x}\boldsymbol{e}\|^2$ follows.*

Proof. Let $\mathcal{F}$ be the feasible set of (23). Without loss of generality, we may assume that $\boldsymbol{e} = \boldsymbol{e}^1$. Using this and $b^0 = 0$, the first constraint in (23) is $z_1 \geq u_1^1$, where z_1 is the first element of $\boldsymbol{z}$. Thus, (23) can be equivalently rewritten as

$$\begin{array}{ll} \min\limits_{\boldsymbol{u}^1} & \|\boldsymbol{u}^1 - \hat{x}\boldsymbol{e}^1\|_2^2 \\ \text{s.t.} & z_1 \geq u_1^1, \quad \|\boldsymbol{z}\|_2^2 \leq \left(\boldsymbol{u}^1\right)^{\mathrm{T}}\boldsymbol{z}. \end{array} \tag{24}$$

It suffices to show that $\boldsymbol{z}$ is an optimal solution of (24). For the boundaries of two constraints in (24), we define $H_1 \coloneqq \{z_1\} \times \mathbb{R}^{p-1}$ and $H_2 \coloneqq \left\{\boldsymbol{u}^1 \in \mathbb{R}^p \,\middle|\, \|\boldsymbol{z}\|_2^2 = \left(\boldsymbol{u}^1\right)^{\mathrm{T}}\boldsymbol{z}\right\}$ where z_1 is the first element of $\boldsymbol{z}$. From the constraint $\|\boldsymbol{z}\|_2^2 = \left(\boldsymbol{u}^1\right)^{\mathrm{T}}\boldsymbol{z}$ in H_2, the hyperplane H_2 is perpendicular to $\boldsymbol{z}$ at $\boldsymbol{z}$ unless $\boldsymbol{z} = \boldsymbol{0}$.

We have $\boldsymbol{e}^{\mathrm{T}}\boldsymbol{z} \geq 0$ from the constraint $\boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1 \geq 0$ in (22), which indicates $z_1 \geq 0$. On the other hand, $|z_1| \leq \|\boldsymbol{z}\|_2^2$ holds generally. Therefore, $0 \leq z_1 \leq \|\boldsymbol{z}\|_2$ follows. The proof is presented for three cases, depending on the value of z_1.

First, suppose that $z_1 = \|\boldsymbol{z}\|_2$. This implies that there exists $\lambda \in \mathbb{R}_+$ such that $\boldsymbol{z} = \lambda\boldsymbol{e}^1$. If $\lambda = 0$, then $\boldsymbol{z} = \boldsymbol{0}$. While the second constraint $\|\boldsymbol{z}\|_2^2 \leq \left(\boldsymbol{u}^1\right)^{\mathrm{T}}\boldsymbol{z}$ vanishes when $\boldsymbol{z} = \boldsymbol{0}$, the first constraint $z_1 \geq u_1^1$ requires $u_1^1 \leq 0$, thus $\boldsymbol{u}^1 \in H_1^- \coloneqq (-\mathbb{R}_+) \times \mathbb{R}^{p-1}$. Here H_1^- is the half space below H_1. Since the vector $\boldsymbol{e}^1$ is perpendicular to H_1^- at the origin O and $\hat{x} \geq 0$,

the point $\mathbf{0}$ is closest to $\hat{x}\boldsymbol{e}^1$ in H_1^-. Thus, $(\boldsymbol{u}^1)^* = \boldsymbol{z} = \mathbf{0}$ under $\lambda = 0$. For $\lambda > 0$, from the second constraint $\|\boldsymbol{z}\|_2^2 \leq (\boldsymbol{u}^1)^{\mathrm{T}} \boldsymbol{z}$ of (24), we obtain $\boldsymbol{z}^{\mathrm{T}} (\lambda \boldsymbol{e}^1) \leq (\boldsymbol{u}^1)^{\mathrm{T}} \lambda \boldsymbol{e}^1$, and this is equivalent to $z_1 \leq u_1^1$. Combining with the first constraint, we deduce that the feasible set of (23) is $\{\boldsymbol{u}^1 \in \mathbb{R}^p \,|\, u_1^1 = z_1\}$ which is equal to H_1. Since $\hat{x}\boldsymbol{e}^1$ is perpendicular to H_1 at $\boldsymbol{z}$, the point $\boldsymbol{z}$ is an optimal solution $(\boldsymbol{u}^1)^*$ of (24).

Second, suppose $z_1 = 0$ with $\boldsymbol{z} \neq \mathbf{0}$. Then, the first constraint requires that $\boldsymbol{u}^1 \in H_1^-$, which is the half space below H_1. From the second constraint and $z_1 = 0$, $\mathcal{F}$ is a subset of

$$H_2^+ \coloneqq \left\{ \boldsymbol{u}^1 \in \mathbb{R}^p \,\middle|\, u_1^1 : \text{free variable}, \ \sum_{i=2}^{p} z_i \left(u_i^1 - z_i\right) \geq 0 \right\},$$

which is the half-space above H_2. Then, for any point $\boldsymbol{u}^1 \in \mathcal{F} \subseteq H_1^- \cap H_2^+$, we can compute the objective function of (24) as

$$\left\|\boldsymbol{u}^1 - \hat{x}\boldsymbol{e}^1\right\|_2^2 = \left|u_1^1 - \hat{x}\right|^2 + \sum_{i=2}^{p} \left|u_i^1\right|^2 = \left|u_1^1 - \hat{x}\right|^2 + \left\|\boldsymbol{u}^1_{\{2,\ldots,p\}}\right\|_2^2. \tag{25}$$

Thus, the first element of $\boldsymbol{u}^1$ and the other elements can be separately determined to minimize (24). Since $\hat{x} \geq 0$ and $0 = z_1 \geq u_1^1$, the term $|u_1^1 - \hat{x}|^2$ of (25) attains the smallest value when $u_1^1 = 0$. The last term $\left\|\boldsymbol{u}^1_{\{2,\ldots,p\}}\right\|_2^2$ is to find a point $\boldsymbol{u}^1$ nearest to the origin O on $H_1 \cap H_2^+$. Since H_2 is perpendicular to $\boldsymbol{z}$ at $\boldsymbol{z} \in H_1 \cap H_2^+$, the nearest point to the origin O is also $\boldsymbol{z}$. Hence, $\boldsymbol{z}$ is an optimal solution (see Figure 2(a)).

Lastly, suppose that $0 < z_1 < \|\boldsymbol{z}\|$. Then, $z_2^2 + \cdots + z_n^2 > 0$ holds. We also have $\hat{x}\boldsymbol{e}^1 \notin \mathcal{F}$, because if $\boldsymbol{u}^1 = \hat{x}\boldsymbol{e}^1$ satisfies the first constraint $z_1 \geq u_1^1$, then $z_1 \geq \hat{x}$ and

$$\|\boldsymbol{z}\|_2^2 - \left(\hat{x}\boldsymbol{e}^1\right)^{\mathrm{T}} \boldsymbol{z} = z_1 \left(z_1 - \hat{x}\right) + z_2^2 + \cdots + z_n^2 > 0,$$

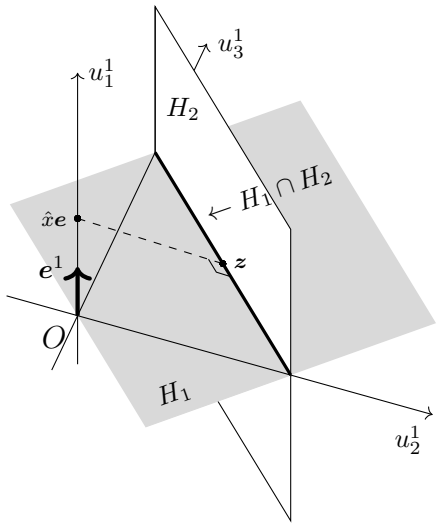
which implies that the second constraint $\|\boldsymbol{z}\|_2^2 \leq (\boldsymbol{u}^1)^{\mathrm{T}} \boldsymbol{z}$ does not hold. We now consider the projection of $\hat{x}\boldsymbol{e}^1$ onto $\mathcal{F}$ in (24). For any $\boldsymbol{u}^1 \in \mathcal{F}$, we have

$$\begin{aligned} \left(\boldsymbol{u}^1 - \boldsymbol{z}\right)^{\mathrm{T}} \left(\hat{x}\boldsymbol{e}^1 - \boldsymbol{z}\right) &= \left(\boldsymbol{u}^1 - \boldsymbol{z}\right)^{\mathrm{T}} \hat{x}\boldsymbol{e}^1 - \left(\boldsymbol{u}^1 - \boldsymbol{z}\right)^{\mathrm{T}} \boldsymbol{z} \\ &\leq \hat{x} \left(u_1^1 - z_1\right) \leq 0. \end{aligned}$$

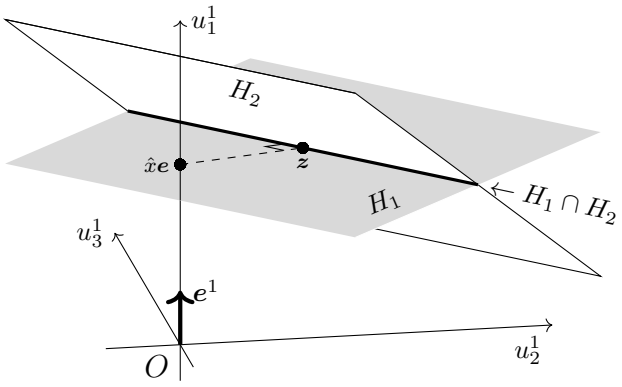
The first inequality follows from the second constraint $\|\boldsymbol{z}\|_2^2 \leq (\boldsymbol{u}^1)^{\mathrm{T}} \boldsymbol{z}$, and the second inequality follows from the first constraint $z_1 \geq u_1^1$. Furthermore, $z_1 \geq z_1$ and $\|\boldsymbol{z}\|_2^2 \leq (\boldsymbol{z})^{\mathrm{T}} \boldsymbol{z}$ ensure $\boldsymbol{z} \in \mathcal{F}$. By the second projection theorem [5, Theorem 9.8], $\boldsymbol{z}$ is the projection of $\hat{x}\boldsymbol{e}$ onto $\mathcal{F}$, therefore, $\boldsymbol{z}$ is the optimal solution on $\mathcal{F}$ (see Figure 2(b)). □

We note that the result above can be extended to a more general case $b^0 \neq 0$ by shifting the variables.

Proposition 4.4. *Suppose that $\boldsymbol{e} \in \mathbb{R}^p$ satisfies $\|\boldsymbol{e}\| = 1$, $\boldsymbol{z} \in \mathbb{R}^p$ is a feasible point of (22), $\hat{x} \geq -b^0$, and the feasible set of (23) is nonempty. Then, $(\boldsymbol{u}^1)^* \coloneqq \boldsymbol{z} - b^0\boldsymbol{e}$ is a solution of (23). In addition, $\Phi(\boldsymbol{z}) = \|\boldsymbol{z} - (b^0 + \hat{x})\,\boldsymbol{e}\|^2$ follows.*



(a) $\left(\boldsymbol{e}^1\right)^{\mathrm{T}} \boldsymbol{z} = 0$ case. The feasible set $\mathcal{F}$ is the set of points on the gray plane and to the right of the white plane.



(b) $0 < \left(\boldsymbol{e}^1\right)^{\mathrm{T}} \boldsymbol{z} < \|\boldsymbol{z}\|$ case. The feasible set $\mathcal{F}$ is the set of points below the gray plane and above the white plane.

Figure 2: Projection of $\hat{x}\boldsymbol{e}$ to $\boldsymbol{z}$ in the case $p = 3$. The gray and white planes denote H_1 and H_2, respectively. The bold line represents the intersection of them.

Proof. Let $\tilde{\boldsymbol{u}}^1 \coloneqq \boldsymbol{u}^1 + b^0\boldsymbol{e}$ and $\tilde{x} \coloneqq \hat{x} + b^0 \geq 0$. Then, (23) is rewritten as

$$\begin{array}{ll} \min\limits_{\tilde{\boldsymbol{u}}^1} & \|\tilde{\boldsymbol{u}}^1 - \tilde{x}\boldsymbol{e}\|_2^2 \\ \text{s.t.} & \boldsymbol{e}^{\mathrm{T}}\boldsymbol{z} \geq \boldsymbol{e}^{\mathrm{T}}\tilde{\boldsymbol{u}}^1, \quad \|\boldsymbol{z}\|_2^2 \leq \left(\tilde{\boldsymbol{u}}^1\right)^{\mathrm{T}} \boldsymbol{z}. \end{array}$$

Since $\tilde{x} \geq 0$, the above problem has an optimal solution $\left(\tilde{\boldsymbol{u}}^1\right)^* \coloneqq \boldsymbol{z}$ by Lemma 4.3. By definition of $\tilde{\boldsymbol{u}}^1$, the point $\boldsymbol{z} - b^0\boldsymbol{e}$ is a solution $(\boldsymbol{u}^1)^*$ of (23). □

Using Proposition 4.4, the first-stage problem (22) can be written as

$$\begin{array}{ll} \min\limits_{\boldsymbol{v}^1} & c\boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1 \\ \text{s.t.} & \boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1 \geq 0, \quad \|\boldsymbol{v}^1 - \hat{x}\boldsymbol{e}\|_q \leq \rho. \end{array} \tag{26}$$

The objective function and the left-hand side of the first constraint are parallel in the direction determined with $\boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1$. Thus, it is easy to solve (26), and there exists a solution $\boldsymbol{v}^1$ which is collinear with $\boldsymbol{e}$.

Theorem 4.5. *Let $\boldsymbol{e}$ be an arbitrary unit vector with $\|\boldsymbol{e}\| = 1$. Suppose $\hat{x} \geq -b^0$. There exists an optimal solution $((\boldsymbol{u}^1)^*, (\boldsymbol{v}^1)^*)$ of (21) such that the vectors $(\boldsymbol{u}^1)^*$ and $(\boldsymbol{v}^1)^*$ are collinear with $\boldsymbol{e}$. Thus, DeepSDP (20) has a rank-1 solution, and it is a tight relaxation of the corresponding QCQP (19).*

Proof. Without loss of generality, we may assume that $\boldsymbol{e} = \boldsymbol{e}^1$. Then, the objective function and the first constraint depend only on the value of v_1^1. The point $[\hat{x} + \rho, \mathbf{0}^{\mathrm{T}}]^{\mathrm{T}}$ attains the maximum value of v_1^1 on the feasible set when $c > 0$, while $[\hat{x} - \rho, \mathbf{0}^{\mathrm{T}}]^{\mathrm{T}}$ attains the minimum

value. It suffices to consider the points $\boldsymbol{v}^1$ on the line segment between $[\hat{x}+\rho, \mathbf{0}^{\mathrm{T}}]^{\mathrm{T}}$ and $[\hat{x}-\rho, \mathbf{0}^{\mathrm{T}}]^{\mathrm{T}}$. Hence, $\boldsymbol{v}^1$ and $\boldsymbol{e}^1$ are collinear, *i.e.,* $\boldsymbol{v}^1 = v_1^1 \boldsymbol{e}^1$. By Proposition 4.4, $\boldsymbol{u}^1$ is also collinear with $\boldsymbol{e}^1$. Applying Lemma 2.3, we conclude that (20) has a rank-1 solution. □

5 Tight DeepSDP for a single-layer neural network

We discuss the tightness of DeepSDP (3) for a single-layer NN ($L = 1$). More precisely,

$$f(\boldsymbol{x}^0) = W^1 \phi\left(W^0 \boldsymbol{x}^0 + \boldsymbol{b}^0\right) + \boldsymbol{b}^1.$$

In this case, the matrix functions in DeepSDP (3) can be rewritten as

$$\begin{aligned}
M_{\mathrm{in}}(P) &= \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & I_{n_0} & O \end{bmatrix}^{\mathrm{T}} P \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & I_{n_0} & O \end{bmatrix} \in \mathbb{S}^{1+n_0+n_1}, \\
M_{\mathrm{mid}}(Q) &= \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} \\ \boldsymbol{b}^0 & W^0 & O \\ \mathbf{0} & O & I_{n_1} \end{bmatrix}^{\mathrm{T}} Q \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} \\ \boldsymbol{b}^0 & W^0 & O \\ \mathbf{0} & O & I_{n_1} \end{bmatrix} \in \mathbb{S}^{1+n_0+n_1}, \\
M_{\mathrm{out}}(S) &= \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & I_{n_0} & O \\ \boldsymbol{b}^1 & O & W^1 \end{bmatrix}^{\mathrm{T}} S \begin{bmatrix} 1 & \mathbf{0}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & I_{n_0} & O \\ \boldsymbol{b}^1 & O & W^1 \end{bmatrix} \in \mathbb{S}^{1+n_0+n_1}.
\end{aligned}$$

We introduce the following assumptions.

Assumption 5.1. *The safety specification set S_y is a polytope.*

Assumption 5.2. *The last layer is the identity layer, i.e., $W^1 = I$ and $\boldsymbol{b}^1 = \mathbf{0}$.*

Assumption 5.2 may appear to impose a strong restriction on the DeepSDPs considered, and potentially limit the generalization of the analysis. However, it does not change the class of the DeepSDPs defined under Assumption 5.1. When S_y is obtained from the DeepSDP with $W_1 = I$, the safety specification set of the original DeepSDP is the projection of S_y by the original W^1, *i.e.,*

$$\left\{W^1 y + \boldsymbol{b}_1 \,\middle|\, y \in S_y\right\},$$

which is also a polytope. In addition, Assumption 5.2 induces $M_{\mathrm{out}}(S) = S$.

The repeated nonlinearities (14), described in Section 3.2.3, are redundant constraints in the corresponding QCQP (15). Thus, if the SDP relaxation of (15) without (14) is tight, then that of (15) with (14) is also tight. We impose the following assumption in the proofs to show the tightness of DeepSDP (3).

Assumption 5.3. *DeepSDP* (3) *has no repeated nonlinearities.*

In Section 5.1, we describe the matrix set $\mathcal{S}$ according to Assumption 4.2. Subsequently, we discuss the tightness of DeepSDP (3) if $\mathcal{X}$ is an ellipsoid in Section 5.2 and a rectangle in Section 5.3.

5.1 Safety specification sets for polytopes

We discuss the safety specification set S_y for estimating a polytope. Since any polytope can be represented by the intersection of a finite number of half spaces, we express S_y with

$$S_y = \bigcap_{\ell=1}^{M} \left\{ \boldsymbol{y} \in \mathbb{R}^{n_L+1} \,\middle|\, (\boldsymbol{c}^\ell)^{\mathrm{T}} \boldsymbol{y} - d_\ell \geq 0 \right\}, \tag{27}$$

where $\boldsymbol{c}^1, \ldots, \boldsymbol{c}^M \in \mathbb{R}^{n_{L+1}}$ and $d_1, \ldots, d_M \in \mathbb{R}$. The objective of DeepSDP (3) in this section is to optimize $d_1, \ldots, d_M$ by fixing the half-space directions $\boldsymbol{c}^1, \ldots, \boldsymbol{c}^M$, in order to sufficiently minimize S_y. Since the ℓth half-space depends only on $\boldsymbol{c}^\ell$ and d_ℓ, each half-space consisting of (27) can be separately considered. An inequality of the form $\boldsymbol{c}^{\mathrm{T}} \boldsymbol{y} - d \geq 0$ associated with the half space can be rewritten as

$$\begin{bmatrix} 1 \\ \boldsymbol{x}^0 \\ \boldsymbol{y} \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} -2d & \boldsymbol{0}^{\mathrm{T}} & \boldsymbol{c}^{\mathrm{T}} \\ \boldsymbol{0} & O & O \\ \boldsymbol{c} & O & O \end{bmatrix} \begin{bmatrix} 1 \\ \boldsymbol{x}^0 \\ \boldsymbol{y} \end{bmatrix} \geq 0. \tag{28}$$

The polytope-shaped safety specification set can be obtained from all the coefficient matrices of the above inequality:

$$\mathcal{S} = \left\{ \begin{bmatrix} -2d & \boldsymbol{0}^{\mathrm{T}} & \boldsymbol{c}^{\mathrm{T}} \\ \boldsymbol{0} & O & O \\ \boldsymbol{c} & O & O \end{bmatrix} \,\middle|\, d \in \mathbb{R} \right\}. \tag{29}$$

Using this $\mathcal{S}$, the resulting DeepSDP is

$$\begin{aligned} \max_{P, \boldsymbol{\lambda}, \boldsymbol{\nu}, \boldsymbol{\eta}, \boldsymbol{\mu}, d} \quad & 2d \\ \text{s.t.} \quad & R \coloneqq M_{\mathrm{in}}(P) + \begin{bmatrix} 0 & \boldsymbol{\nu}^{\mathrm{T}} W^0 & -\boldsymbol{\nu}^{\mathrm{T}} - \boldsymbol{\eta}^{\mathrm{T}} \\ (W^0)^{\mathrm{T}} \boldsymbol{\nu} & O & -(W^0)^{\mathrm{T}} \operatorname{diag}(\boldsymbol{\lambda}) \\ -\boldsymbol{\nu} - \boldsymbol{\eta} & -\operatorname{diag}(\boldsymbol{\lambda}) W^0 & 2\operatorname{diag}(\boldsymbol{\lambda}) \end{bmatrix} \\ & \qquad + \begin{bmatrix} 0 & \boldsymbol{0}^{\mathrm{T}} & \boldsymbol{0}^{\mathrm{T}} \\ \boldsymbol{0} & O & -(W^0)^{\mathrm{T}} T \\ \boldsymbol{0} & -T W^0 & 2T \end{bmatrix} + \begin{bmatrix} -2d & \boldsymbol{0}^{\mathrm{T}} & \boldsymbol{c}^{\mathrm{T}} \\ \boldsymbol{0} & O & O \\ \boldsymbol{c} & O & O \end{bmatrix} \succeq O, \\ & T = \sum_{i=1}^{n} \sum_{j=i+1}^{n} \mu_{ij} (\boldsymbol{e}_i - \boldsymbol{e}_j)(\boldsymbol{e}_i - \boldsymbol{e}_j)^{\mathrm{T}}, \quad \mu_{ij} \geq 0, \\ & P \in \mathcal{P}_{\mathcal{X}}, \quad \boldsymbol{\lambda}, \boldsymbol{\nu}, \boldsymbol{\eta} \in \mathbb{R}^n_+, \quad d \in \mathbb{R}, \end{aligned} \tag{30}$$

where the objective is to maximize $2d$ as we want to find the largest d such that $\boldsymbol{c}^{\mathrm{T}} \boldsymbol{x}^1 \geq d$ holds for all $\boldsymbol{x}^1 \in \mathcal{X}_1$. Notice that (30) is not an standard SDP due to the matrix variable P. In Sections 5.2 and 5.3, we show that the precise formulation for $\mathcal{P}_{\mathcal{X}}$ provides an SDP from (30).

The dual problem of (30) is

$$
\begin{aligned}
\min_{\substack{\boldsymbol{x}^0,\boldsymbol{x}^1,\\ X^{00},X^{10},X^{11}}} \quad & 2\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}^1 \\
\text{s.t.} \quad & M_{\mathrm{in}}(P) \bullet G \le 0,\ \forall P \in \mathcal{P}_{\mathcal{X}}, \\
& \begin{bmatrix} 0 & \mathbf{0}^{\mathrm{T}} & -b_i^0 (\boldsymbol{e}^i)^{\mathrm{T}} \\ \mathbf{0} & O & -(W^0)^{\mathrm{T}} \boldsymbol{e}_i \boldsymbol{e}_i^{\mathrm{T}} \\ -b_i^0 \boldsymbol{e}^i & -\boldsymbol{e}_i \boldsymbol{e}_i^{\mathrm{T}} W^0 & 2\boldsymbol{e}_i \boldsymbol{e}_i^{\mathrm{T}} \end{bmatrix} \bullet G \le 0, \quad i = 1, \dots, n_1, \\
& \begin{bmatrix} 2b_i^0 & \boldsymbol{e}_i^{\mathrm{T}} W^0 & -\boldsymbol{e}_i^{\mathrm{T}} \\ (W^0)^{\mathrm{T}} \boldsymbol{e}_i & O & O \\ -\boldsymbol{e}_i & O & O \end{bmatrix} \bullet G \le 0, \qquad i = 1, \dots, n_1, \qquad (31) \\
& \begin{bmatrix} 0 & \mathbf{0}^{\mathrm{T}} & -\boldsymbol{e}_i^{\mathrm{T}} \\ \mathbf{0} & O & O \\ -\boldsymbol{e}_i & O & O \end{bmatrix} \bullet G \le 0, \qquad i = 1, \dots, n_1, \\
& G \coloneqq \begin{bmatrix} 1 & (\boldsymbol{x}^0)^{\mathrm{T}} & (\boldsymbol{x}^1)^{\mathrm{T}} \\ \boldsymbol{x}^0 & X^{00} & (X^{10})^{\mathrm{T}} \\ \boldsymbol{x}^1 & X^{10} & X^{11} \end{bmatrix} \in \mathbb{S}_+^{(1+n_0+n_1)},
\end{aligned}
$$

which corresponds to the primal SDP relaxation (16) in Section 3.4. The repeated nonlinearities (14) are removed here by Assumption 5.3, thus the constraints corresponding to the dual variables μ_{ij} and T in (30) do not appear in (31). For the fixed vector $\boldsymbol{e} \in \mathbb{R}^p$ such that $\|\boldsymbol{e}\| = 1$, define $\boldsymbol{u}^1, \dots, \boldsymbol{u}^{n_0} \in \mathbb{R}^p$ and $\boldsymbol{v}^1, \dots, \boldsymbol{v}^{n_1} \in \mathbb{R}^p$ to substitute $\boldsymbol{x}^0$ and $\boldsymbol{x}^1$ with

$$
\boldsymbol{x}^0 = \begin{bmatrix} \boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^1 \\ \vdots \\ \boldsymbol{e}^{\mathrm{T}}\boldsymbol{u}^{n_0} \end{bmatrix} \in \mathbb{R}^{n_0}, \quad \boldsymbol{x}^1 = \begin{bmatrix} \boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^1 \\ \vdots \\ \boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^{n_1} \end{bmatrix} \in \mathbb{R}^{n_1}, \quad X^{00} = \begin{bmatrix} (\boldsymbol{u}^1)^{\mathrm{T}}\boldsymbol{u}^1 & \cdots & (\boldsymbol{u}^1)^{\mathrm{T}}\boldsymbol{u}^{n_0} \\ \vdots & \ddots & \vdots \\ (\boldsymbol{u}^{n_0})^{\mathrm{T}}\boldsymbol{u}^1 & \cdots & (\boldsymbol{u}^{n_0})^{\mathrm{T}}\boldsymbol{u}^{n_0} \end{bmatrix} \in \mathbb{S}^{n_0},
$$

$$
X^{10} = \begin{bmatrix} (\boldsymbol{v}^1)^{\mathrm{T}}\boldsymbol{u}^1 & \cdots & (\boldsymbol{v}^1)^{\mathrm{T}}\boldsymbol{u}^{n_0} \\ \vdots & \ddots & \vdots \\ (\boldsymbol{v}^{n_1})^{\mathrm{T}}\boldsymbol{u}^1 & \cdots & (\boldsymbol{v}^{n_1})^{\mathrm{T}}\boldsymbol{u}^{n_0} \end{bmatrix} \in \mathbb{R}^{n_1 \times n_0}, \quad X^{11} = \begin{bmatrix} (\boldsymbol{v}^1)^{\mathrm{T}}\boldsymbol{v}^1 & \cdots & (\boldsymbol{v}^1)^{\mathrm{T}}\boldsymbol{v}^{n_1} \\ \vdots & \ddots & \vdots \\ (\boldsymbol{v}^{n_1})^{\mathrm{T}}\boldsymbol{v}^1 & \cdots & (\boldsymbol{v}^{n_1})^{\mathrm{T}}\boldsymbol{v}^{n_1} \end{bmatrix} \in \mathbb{S}^{n_1}.
$$

Then, from (31), we have the following nonlinear formulation:

$$
\begin{aligned}
\min_{\boldsymbol{u}^j,\boldsymbol{v}^i} \quad & 2\textstyle\sum_{i=1}^{n_1} c_i\, \boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^i && \text{(32a)} \\
\text{s.t.} \quad & \boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^i \ge 0, && \text{(32b)} \\
& \boldsymbol{e}^{\mathrm{T}}\boldsymbol{v}^i \ge \boldsymbol{e}^{\mathrm{T}} \left(\textstyle\sum_{j=1}^{n_0} W_{ij}\boldsymbol{u}^j + b_i^0 \boldsymbol{e} \right), \qquad i = 1, \dots, n_1, && \text{(32c)} \\
& \left\| \boldsymbol{v}^i \right\|_2^2 \le \left(\textstyle\sum_{j=1}^{n_0} W_{ij}\boldsymbol{u}^j + b_i^0 \boldsymbol{e} \right)^{\mathrm{T}} \boldsymbol{v}^i, \quad i = 1, \dots, n_1, && \text{(32d)} \\
& M_{\mathrm{in}}(P) \bullet G \le 0,\ \forall P \in \mathcal{P}_{\mathcal{X}}. && \text{(32e)}
\end{aligned}
$$

The matrix G still exists in (32e). The transformation of this constraint depends on the definition of $\mathcal{X}$, and thus it will be discussed in the subsequent subsections.

5.2 Ellipsoidal input

Consider the case that the input set $\mathcal{X}$ is an ellipsoid with the center $\hat{\boldsymbol{x}} \in \mathbb{R}^{n_0}$ and radius ρ:

$$\mathcal{X} \coloneqq \{\boldsymbol{x} \mid \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2 \leq \rho\}. \tag{33}$$

For (33), the matrix set $\mathcal{P}_{\mathcal{X}}$ becomes

$$\mathcal{P}_{\mathcal{X}} = \left\{ \gamma \begin{bmatrix} \hat{\boldsymbol{x}}^{\mathrm{T}}\hat{\boldsymbol{x}} - \rho^2 & -\hat{\boldsymbol{x}}^{\mathrm{T}} \\ -\hat{\boldsymbol{x}} & I \end{bmatrix} \,\middle|\, \gamma \geq 0,\ \gamma \in \mathbb{R} \right\},$$

where γ is the dual variable of DeepSDP (3). By normalizing $\mathcal{P}_{\mathcal{X}}$ with $\gamma = 1$, we have

$$\begin{aligned} M_{\mathrm{in}} \left(\begin{bmatrix} \hat{\boldsymbol{x}}^{\mathrm{T}}\hat{\boldsymbol{x}} - \rho^2 & -\hat{\boldsymbol{x}}^{\mathrm{T}} \\ -\hat{\boldsymbol{x}} & I \end{bmatrix} \right) \bullet G &= \sum_{j=1}^{n_0} \left\{ \left(\boldsymbol{u}^j\right)^{\mathrm{T}} \boldsymbol{u}^j - 2\hat{x}_j \boldsymbol{e}^{\mathrm{T}} \boldsymbol{u}^j + \hat{x}_j^2 \boldsymbol{e}^{\mathrm{T}} \boldsymbol{e} \right\} - \rho^2 \\ &= \sum_{j=1}^{n_0} \left\| \boldsymbol{u}^j - \hat{x}_j \boldsymbol{e} \right\|_2^2 - \rho^2 \ \leq 0. \end{aligned}$$

Thus, the problem (32) can be described precisely as

$$\min_{\boldsymbol{u}^j, \boldsymbol{v}^i} \quad \text{(32a)} \quad \text{s.t.} \quad \text{(32b), (32c), (32d)}, \ \sum_{j=1}^{n_0} \|\boldsymbol{u}^j - \hat{x}_j \boldsymbol{e}\|_2^2 \leq \rho^2. \tag{34}$$

Applying the same decomposition method in Section 4 results in the following two-stage problem:

$$\begin{aligned} \min_{\boldsymbol{v}^1, \ldots, \boldsymbol{v}^{n_1}} \quad & \text{(32a)} \\ \text{s.t.} \quad & \text{(32b)}, \quad \Psi_{\mathrm{ellipsoid}}(\boldsymbol{v}^1, \ldots, \boldsymbol{v}^{n_1}) \leq \rho^2, \end{aligned} \tag{35}$$

$$\begin{aligned} \Psi_{\mathrm{ellipsoid}}(\boldsymbol{v}^1, \ldots, \boldsymbol{v}^{n_1}) \coloneqq \min_{\boldsymbol{u}^1, \ldots, \boldsymbol{u}^{n_0}} \quad & \sum_{j=1}^{n_0} \|\boldsymbol{u}^j - \hat{x}_j \boldsymbol{e}\|_2^2 \\ \text{s.t.} \quad & \text{(32c), (32d)}. \end{aligned} \tag{36}$$

The second-stage problem (36) is a convex optimization problem with a quadratic objective function and linear inequality constraints.

The following lemma characterizes solutions to the second-stage problem, and shows that all variables are a linear combination of $\boldsymbol{e}^1$, $\boldsymbol{v}^1, \ldots, \boldsymbol{v}^{n_1}$.

Lemma 5.4. *Suppose that $\boldsymbol{e} = \boldsymbol{e}^1$. For any optimal solution $((\boldsymbol{u}^1)^*, \ldots, (\boldsymbol{u}^{n_0})^*)$ of (36), there exist $\boldsymbol{m} \in \mathbb{R}^{n_0}$ and $M \in \mathbb{R}^{n_1 \times n_0}$ such that*

$$(\boldsymbol{u}^j)^* = m_j \boldsymbol{e}^1 + \sum_{i=1}^{n_1} M_{ij} \boldsymbol{v}^i \quad \textit{for each } j \in \{1, \ldots, n_0\}.$$

Proof. Since (36) is a convex optimization, any pair $((\tilde{\boldsymbol{u}}^1, \ldots, \tilde{\boldsymbol{u}}^{n_0}), (\tilde{\boldsymbol{\nu}}, \tilde{\boldsymbol{\lambda}}))$ of the primal and dual solutions satisfies the KKT conditions:

$$\begin{cases} \text{(32c), (32d)}, \ \boldsymbol{\nu} \in \mathbb{R}^{n_1}_+, \ \boldsymbol{\lambda} \in \mathbb{R}^{n_1}_+, & \text{(37a)} \\ \begin{bmatrix} \boldsymbol{u}^1 \\ \vdots \\ \boldsymbol{u}^{n_0} \end{bmatrix} = \begin{bmatrix} \hat{x}_1 \boldsymbol{e}^1 \\ \vdots \\ \hat{x}_{n_0} \boldsymbol{e}^1 \end{bmatrix} - \sum_{i=1}^{n_1} \frac{\nu_i}{2} \begin{bmatrix} W_{i1} \boldsymbol{e}^1 \\ \vdots \\ W_{in} \boldsymbol{e}^1 \end{bmatrix} + \sum_{i=1}^{n_1} \frac{\lambda_i}{2} \begin{bmatrix} W_{i1} \boldsymbol{v}^i \\ \vdots \\ W_{in} \boldsymbol{v}^i \end{bmatrix} & \text{(37b)} \\ \nu_i \left[(\boldsymbol{e}^1)^{\mathrm{T}} \left(\sum_{j=1}^{n_0} W_{ij} \boldsymbol{u}^j + b_i^0 \boldsymbol{e}^1 \right) - \left(\boldsymbol{e}^1 \right)^{\mathrm{T}} \boldsymbol{v}^i \right] = 0, \quad i = 1, \ldots, n_1, & \text{(37c)} \\ \lambda_i \left[\left\| \boldsymbol{v}^i \right\|^2 - \left(\sum_{j=1}^{n_0} W_{ij} \boldsymbol{u}^j + b_i^0 \boldsymbol{e}^1 \right)^{\mathrm{T}} \boldsymbol{v}^i \right] = 0, \quad i = 1, \ldots, n_1. & \text{(37d)} \end{cases}$$

By (37b), the desired result follows by taking

$$\begin{aligned} m_j &= \hat{x}_j - \sum_i \frac{\nu_i W_{ij}}{2} \quad \text{for } j = 1, \ldots, n_0, \\ M_{ij} &= \frac{\lambda_i W_{ij}}{2} \qquad\qquad \text{for } j = 1, \ldots, n_0, \ i = 1, \ldots, n_1. \end{aligned}$$

□

Using Lemma 5.4, the first-stage problem (35) can be rewritten as

$$\min_{\boldsymbol{v}^i} \quad \text{(32a)} \quad \text{s.t.} \quad \text{(32b)}, \ \sum_{j=1}^{n_1} \left\| m_j \boldsymbol{e}^1 + \sum_{i=1}^{n_1} M_{ij} \boldsymbol{v}^i \right\|_2^2 \leq \rho^2. \tag{38}$$

Lemma 5.5. *Suppose that* $\boldsymbol{e} = \boldsymbol{e}^1$. *The problem* (38) *has an optimal solution* $((\boldsymbol{v}^1)^*, \ldots, (\boldsymbol{v}^{n_1})^*)$ *such that* $(\boldsymbol{v}^1)^*, \ldots, (\boldsymbol{v}^{n_1})^*$ *are collinear with* $\boldsymbol{e}^1$.

Proof. Let $(\bar{\boldsymbol{v}}^1, \ldots, \bar{\boldsymbol{v}}^{n_1})$ be an optimal solution of (38). Assume on the contrary that at least one optimal solution is not collinear with $\boldsymbol{e}^1$. For all $i \in \{1, \ldots, n_1\}$, we define $\hat{\boldsymbol{v}}^i := \left[\bar{v}_1^i, \, 0, \, \cdots, \, 0 \right]^{\mathrm{T}} \in \mathbb{R}^n$ as the projection of $\bar{\boldsymbol{v}}^i$ onto the set $\{k\boldsymbol{e}^1 \,|\, k \in \mathbb{R}\}$. Then, as $(\boldsymbol{e}^1)^{\mathrm{T}} \hat{\boldsymbol{v}}^i = (\boldsymbol{e}^1)^{\mathrm{T}} \bar{\boldsymbol{v}}^i$ for all i, $\left(\hat{\boldsymbol{v}}^1, \ldots, \hat{\boldsymbol{v}}^{n_1} \right)$ satisfies the constraints (32b) with the same objective value, *i.e.*,

$$\sum_{i=1}^{n_1} c_i \left(\boldsymbol{e}^1 \right)^{\mathrm{T}} \hat{\boldsymbol{v}}^i = \sum_{i=1}^{n_1} c_i \left(\boldsymbol{e}^1 \right)^{\mathrm{T}} \bar{\boldsymbol{v}}^i.$$

For the last constraint of (38), from the equivalence of the first elements of vectors $M_{ij} \hat{\boldsymbol{v}}^i$

and $M_{ij}\bar{\boldsymbol{v}}^i$ for any pair (i,j), we have

$$\begin{aligned}\sum_{j=1}^{n_1}\left\|m_j\boldsymbol{e}^1+\sum_{i=1}^{n_1}M_{ij}\hat{\boldsymbol{v}}^i\right\|_2^2 &= \sum_{j=1}^{n_1}\left(m_j+\sum_{i=1}^{n_1}M_{ij}\bar{v}_1^i\right)^2\\ &\le \sum_{j=1}^{n_1}\left[\left(m_j+\sum_{i=1}^{n_1}M_{ij}\bar{v}_1^i\right)^2+\sum_{k=2}^{n_1}\left(\sum_{i=1}^{n_1}M_{ij}\bar{v}_k^i\right)^2\right]\\ &= \sum_{j=1}^{n_1}\left\|m_j\boldsymbol{e}^1+\sum_{i=1}^{n_1}M_{ij}\bar{\boldsymbol{v}}^i\right\|_2^2\le\rho^2. \end{aligned}\tag{39}$$

Hence, $\left(\hat{\boldsymbol{v}}^1,\ldots,\hat{\boldsymbol{v}}^{n_1}\right)$ is also an optimal solution of (38). Thus, $\left((\hat{\boldsymbol{v}}^1)^*,\ldots,(\hat{\boldsymbol{v}}^{n_1})^*\right)$ is an optimal solution with collinearity. □

From Lemmas 5.4 and 5.5, associated with each of the two stages, we obtain the tightness condition for the single-layer DeepSDP (3) with the ellipsoidal input set.

Theorem 5.6. *Suppose that Assumptions 5.1 and 5.2 hold. The problem* (31) *has a rank-1 matrix solution. In addition, if both* (31) *and* (3) *have optimal solutions, and the feasible set of* (3) *is bounded, then DeepSDP* (3) *is a tight relaxation for the original QCQP* (15).

Proof. Without loss of generality, we may assume that $\boldsymbol{e}=\boldsymbol{e}^1$. By Lemma 5.4, the second-stage problem (36) has an optimal solution:

$$\begin{bmatrix}(\boldsymbol{u}^1)^*\\ \vdots\\ (\boldsymbol{u}^{n_0})^*\end{bmatrix}=\begin{bmatrix}m_1\left(\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}\right)\boldsymbol{e}^1+\sum_{i=1}^{n_1}M_{i1}\left(\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}\right)\boldsymbol{v}^i\\ \vdots\\ m_{n_0}\left(\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}\right)\boldsymbol{e}^1+\sum_{i=1}^{n_1}M_{in_0}\left(\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}\right)\boldsymbol{v}^i\end{bmatrix}$$

with some $\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}$, where $m_j\left(\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}\right)$ and $M_{ij}\left(\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}\right)$ depend on the variable $\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}$. Then, (35) can be rewritten as the form

$$\begin{aligned}\min_{\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}}\quad & \text{(32a)}\\ \text{s.t.}\quad & \text{(32b)},\quad \sum_{j=1}^{n_0}\left\|\left[m_j\left(\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}\right)-\hat{x}_j\right]\boldsymbol{e}^1+\sum_{i=1}^{n_1}M_{ij}\left(\boldsymbol{v}^1,\ldots,\boldsymbol{v}^{n_1}\right)\boldsymbol{v}^i\right\|_2^2\le\rho^2,\end{aligned}$$

which is also represented by (38) with appropriate m_j and M_{ij}. By Lemma 5.5, if $((\boldsymbol{v}^1)^*,\ldots,(\boldsymbol{v}^{n_1})^*)$ is an optimal solution of the above problem, all n_1 points $(\boldsymbol{v}^1)^*,\ldots,(\boldsymbol{v}^{n_1})^*$ can be collinear with $\boldsymbol{e}^1$. Since each $(\boldsymbol{u}^j)^*$ depends on $(\boldsymbol{v}^1)^*,\ldots,(\boldsymbol{v}^{n_1})^*$, by applying Lemma 5.4 again, $(\boldsymbol{u}^1)^*,\ldots,(\boldsymbol{u}^{n_0})^*$ are also collinear with $\boldsymbol{e}^1$ under the points $(\boldsymbol{v}^1)^*,\ldots,(\boldsymbol{v}^{n_1})^*$. Therefore, Lemma 2.3 guarantees the existence of a rank-1 matrix solution to (31). The second result follows from strong duality in Lemma 2.1 and Figure 1. □

5.3 Rectangular input

We consider the case that the input set $\mathcal{X}$ is a rectangle, *i.e.,*

$$\mathcal{X} \coloneqq \{\boldsymbol{x} \,|\, |x_j - \hat{x}_j| \le \rho_j \text{ for all } j \in \{1, \dots, n_0\}\}, \tag{40}$$

where $\hat{\boldsymbol{x}} \in \mathbb{R}^{n_0}$ is the center of the rectangle, and ρ_j is the length of the rectangle along with the jth dimension. For (40), the matrix set $\mathcal{P}_{\mathcal{X}}$ becomes

$$\mathcal{P}_{\mathcal{X}} = \left\{ \sum_{j=1}^{n_0} \gamma_j \begin{bmatrix} \hat{x}_j^2 - \rho_j^2 & -\hat{x}_j \left(\boldsymbol{e}^j\right)^{\mathrm{T}} \\ -\hat{x}_j \boldsymbol{e}^j & \boldsymbol{e}^j \left(\boldsymbol{e}^j\right)^{\mathrm{T}} \end{bmatrix} \middle| \boldsymbol{\gamma} \ge \boldsymbol{0},\ \boldsymbol{\gamma} \in \mathbb{R}^{n_0} \right\},$$

where $\boldsymbol{\gamma} \in \mathbb{R}^{n_0}$ is the dual variable of DeepSDP (3). The corresponding constraints in the primal SDP relaxation are

$$M_{\mathrm{in}} \left(\begin{bmatrix} \hat{x}_j^2 - \rho_j^2 & -\hat{x}_j \left(\boldsymbol{e}^j\right)^{\mathrm{T}} \\ -\hat{x}_j \boldsymbol{e}^j & \boldsymbol{e}^j \left(\boldsymbol{e}^j\right)^{\mathrm{T}} \end{bmatrix} \right) \bullet G \le 0, \qquad j = 1, \dots, n_0.$$

We consider the same substitution of G as (31) with $\boldsymbol{u}^1, \dots, \boldsymbol{u}^{n_0}$ and $\boldsymbol{v}^1, \dots, \boldsymbol{v}^{n_1}$. For any $j \in \{1, \dots, n_0\}$, the left-hand side of the above inequality is

$$\begin{aligned} M_{\mathrm{in}} \left(\begin{bmatrix} \hat{x}_j^2 - \rho_j^2 & -\hat{x}_j \left(\boldsymbol{e}^j\right)^{\mathrm{T}} \\ -\hat{x}_j \boldsymbol{e}^j & \boldsymbol{e}^j \left(\boldsymbol{e}^j\right)^{\mathrm{T}} \end{bmatrix} \right) \bullet G &= \left(\boldsymbol{u}^j\right)^{\mathrm{T}} \boldsymbol{u}^j - 2\hat{x}_j \boldsymbol{e}^{\mathrm{T}} \boldsymbol{u}^j + \hat{x}_j^2 \boldsymbol{e}^{\mathrm{T}} \boldsymbol{e} - \rho_j^2 \\ &= \left\| \boldsymbol{u}^j - \hat{x}_j \boldsymbol{e} \right\|_2^2 - \rho_j^2. \end{aligned}$$

Thus, the problem (32) can be transformed into

$$\begin{aligned} \min_{\boldsymbol{u}^j, \boldsymbol{v}^i} \quad & \text{(32a)} \quad \text{s.t.} \quad \text{(32b), (32c), (32d),} \\ & \|\boldsymbol{u}^j - \hat{x}_j \boldsymbol{e}\|_2 \le \rho_j^2, \quad j = 1, \dots, n_0. \end{aligned} \tag{41}$$

To derive a two-stage problem as in Section 4, we assume the following.

Assumption 5.7. *W^0 is the identity matrix, i.e., $n_0 = n_1$ and $W^0 = I_{n_0}$.*

Then, (32c) and (32d) are reduced to

$$\begin{aligned} \boldsymbol{e}^{\mathrm{T}} \boldsymbol{v}^j &\ge \boldsymbol{e}^{\mathrm{T}} (\boldsymbol{u}^j + b_i^0 \boldsymbol{e}), && j = 1, \dots, n_0, \\ \left\| \boldsymbol{v}^j \right\|_2^2 &\le \left(\boldsymbol{u}^j + b_j^0 \boldsymbol{e}\right)^{\mathrm{T}} \boldsymbol{v}^j, && j = 1, \dots, n_0, \end{aligned}$$

and the following two-stage problem can be obtained:

$$\begin{aligned} \min_{\boldsymbol{v}^1, \dots, \boldsymbol{v}^n} \quad & \text{(32a)} \\ \text{s.t.} \quad & \text{(32b)}, \quad \Psi_{\mathrm{rect}}^j(\boldsymbol{v}^j) \le \rho_j^2,\ j = 1, \dots, n_0, \end{aligned} \tag{42}$$

and

$$\begin{aligned} \Psi_{\mathrm{rect}}^j(\boldsymbol{z}) \coloneqq \min_{\boldsymbol{u}^j} \quad & \|\boldsymbol{u}^j - \hat{x}_j \boldsymbol{e}\|_2 \\ \text{s.t.} \quad & \boldsymbol{e}^{\mathrm{T}} \boldsymbol{z} \ge \boldsymbol{e}^{\mathrm{T}} (\boldsymbol{u}^j + b_i^0 \boldsymbol{e}), \\ & \|\boldsymbol{z}\|_2^2 \le \left(\boldsymbol{u}^j + b_i^0 \boldsymbol{e}\right)^{\mathrm{T}} \boldsymbol{z}. \end{aligned} \tag{43}$$

When $\boldsymbol{e} = \boldsymbol{e}^1$, the second-stage problem $\Psi^j_{\text{rect}}(\boldsymbol{z})$ is equivalent to $\Phi(\boldsymbol{z})$ defined in (23). Thus, $\Psi^j_{\text{rect}}(\boldsymbol{z}) = \|\boldsymbol{z} - \hat{x}_j \boldsymbol{e}^1\|_2^2$ follows from Proposition 4.4, and the tightness of DeepSDP (3) can be shown as follows.

Theorem 5.8. *Let $\boldsymbol{e}$ be an arbitrary unit vector with $\|\boldsymbol{e}\| = 1$. Suppose that Assumptions 5.1, 5.2, and 5.7 hold. Then, any solution $\left\{(\boldsymbol{u}^1)^*, \ldots, (\boldsymbol{u}^n)^*, (\boldsymbol{v}^1)^*, \ldots, (\boldsymbol{v}^n)^*\right\}$ of (41) are collinear with $\boldsymbol{e}$. Thus, the problem (31) has a rank-1 matrix solution. In addition, if both (31) and (3) have optimal solutions, and the feasible set of (3) is bounded, then DeepSDP (3) is a tight relaxation for the original QCQP (15).*

Proof. Without loss of generality, we may assume that $\boldsymbol{e} = \boldsymbol{e}^1$. By Proposition 4.4, the optimal solution of the second-stage problem (43) is $\boldsymbol{u}^j = \boldsymbol{v}^j - b_i^0 \boldsymbol{e}$, and the first-stage problem (42) is

$$\begin{array}{ll} \min\limits_{\boldsymbol{v}^i} & 2\sum_{i=1}^{n_1} c_i \left(\boldsymbol{e}^1\right)^{\mathrm{T}} \boldsymbol{v}^i \\ \text{s.t.} & \left(\boldsymbol{e}^1\right)^{\mathrm{T}} \boldsymbol{v}^i \geq 0,\ i = 1, \ldots, n, \\ & \|\boldsymbol{v}^i - (b_i^0 + \hat{x}_i)\, \boldsymbol{e}^1\|_2^2 \leq \rho_i^2,\ i = 1, \ldots, n. \end{array}$$

Then, the same discussion in Theorem 4.5 can be applied to each $\boldsymbol{v}^i$, therefore, the first and second results of this theorem follows by Lemmas 2.3 and 2.1, respectively. □

6 Conclusion

We have presented sufficient conditions under which DeepSDPs for single-layer NNs are tight in three cases. A common aspect of these cases is the identification of a polytope safety specification set. For the first case, we have proved that the DeepSDP for a single-neuron NN provides a tight optimal solution if the given vectors $\hat{x}$ and b^0 satisfies $\hat{x} \geq -b^0$. In the second case for the DeepSDP with ellipsoidal inputs, we have proved that the DeepSDP is always a tight relaxation without any assumptions. The Karush-Kuhn-Tucker (KKT) conditions have been used to analyze the optimal solutions $(\boldsymbol{u}^j)^*, (\boldsymbol{v}^i)^*$ of the first- and second-stage problems, as shown in Lemma 5.4. For the third case where the input set is a rectangle, we have shown that the auxiliary two-stage problem can be reduced into the one in the first case and the tightness can be determined by the condition for the single-neuron NNs.

For future work, it would be interesting to investigate the extent to which the tightness condition can be weakened by incorporating the local quadratic constraints proposed in [12], which have not been included in this work. The DeepSDPs that satisfy the tightness conditions of this work remain tight even with the inclusion of local constraints. As a result, incorporating the local constraints may lead to tighter SDP relaxations even under milder assumptions. Also, examining the tightness of SDP relaxations for QCQPs with the ReLU function as a quadratic constraint, which was developed for different purposes, would be an interesting direction. Our approach, which decomposes the problem into a two-stage problem and use the KKT conditions, may be useful for solving the QCQPs.

Declarations

Conflict of interest The authors have no conflict of interest to disclose.

References

[1] C. J. Argue, F. Kılınç-Karzan, and A. L. Wang. Necessary and sufficient conditions for rank-one generated cones. arXiv:2007.07433, 2020.

[2] G. Azuma, M. Fukuda, S. Kim, and M. Yamashita. Exact SDP relaxations of quadratically constrained quadratic programs with forest structures. *Journal of Global Optimization*, 82(2):243–262, 2022.

[3] G. Azuma, M. Fukuda, S. Kim, and M. Yamashita. Exact SDP relaxations for quadratic programs with bipartite graph structures. *Journal of Global Optimization*, 86:671 – 691, 2023.

[4] O. Bastani, Y. Ioannou, L. Lampropoulos, D. Vytiniotis, A. Nori, and A. Criminisi. Measuring neural net robustness with constraints. *Advances in neural information processing systems*, 29, 2016.

[5] A. Beck. *Introduction to Nonlinear Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2014. doi: 10.1137/1.9781611973655. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611973655.

[6] S. Burer and Y. Ye. Exact semidefinite formulations for a class of (random and non-random) nonconvex quadratic programs. *Mathematical Programming*, 181(1):1–17, 2020. doi: 10.1007/s10107-019-01367-2.

[7] L. Deng, G. Hinton, and B. Kingsbury. New types of deep neural network learning for speech recognition and related applications: An overview. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 8599–8603. IEEE, 2013.

[8] S. Dutta, S. Jha, S. Sankaranarayanan, and A. Tiwari. Output range analysis for deep feedforward neural networks. In *NASA Formal Methods Symposium*, pages 121–138. Springer, 2018.

[9] K. Dvijotham, R. Stanforth, S. Gowal, T. A. Mann, and P. Kohli. A dual approach to scalable verification of deep networks. In *UAI*, volume 1, page 3, 2018.

[10] M. Egmont-Petersen, D. de Ridder, and H. Handels. Image processing with neural networks–a review. *Pattern recognition*, 35(10):2279–2301, 2002.

[11] M. Fazlyab, M. Morari, and G. J. Pappas. Probabilistic verification and reachability analysis of neural networks via semidefinite programming. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 2726–2731, 2019.

[12] M. Fazlyab, M. Morari, and G. J. Pappas. Safety verification and robustness analysis of neural networks via quadratic constraints and semidefinite programming. *IEEE Transactions on Automatic Control*, 67(1):1–15, 2022.

[13] B. Gärtner and J. Matousek. *Approximation Algorithms and Semidefinite Programming*. Springer Publishing Company, Incorporated, 2014. ISBN 978-3-642-22014-2. doi: 10.1007/978-3-642-22015-9.

[14] Y. Goldberg. A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, 57:345–420, 2016.

[15] Y. Goldberg. *Neural network methods in natural language processing*. Morgan & Claypool Publishers, 2017.

[16] X. Huang, M. Kwiatkowska, S. Wang, and M. Wu. Safety verification of deep neural networks. In R. Majumdar and V. Kunčak, editors, *Computer Aided Verification*, pages 3–29, Cham, 2017. Springer International Publishing.

[17] X. Huang, G. Jin, and W. Ruan. *Verification of Deep Learning*, pages 181–203. Springer Nature Singapore, Singapore, 2023. ISBN 978-981-19-6814-3.

[18] S. Kim and M. Kojima. Exact solutions of some nonconvex quadratic optimization problems via SDP and SOCP relaxations. *Computational Optimization and Applications*, 26(2):143–154, 2003.

[19] S. Kim and M. Kojima. Strong duality of a conic optimization problem with a single hyperplane and two cone constraints. *Optimization*, pages 1–21, 2023. doi: 10.1080/02331934.2023.2251987.

[20] F. Kılınç-Karzan and A. L. Wang. Exactness in sdp relaxations of qcqps: Theory and applications. arXiv:2107.06885, 2021.

[21] A. Lomuscio and L. Maganti. An approach to reachability analysis for feed-forward relu neural networks (2017). *arXiv preprint arXiv:1706.07351*, 2017.

[22] Z.-Q. Luo and T.-H. Chang. *SDP relaxation of homogeneous quadratic optimization: approximation bounds and applications*, pages 117–165. Cambridge University Press, 2009.

[23] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1765–1773, 2017.

[24] M. Newton and A. Papachristodoulou. Neural network verification using polynomial optimisation. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 5092–5097. IEEE, 2021.

[25] H. Salman, G. Yang, H. Zhang, C.-J. Hsieh, and P. Zhang. A convex relaxation barrier to tight robustness verification of neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

[26] S. Sojoudi and J. Lavaei. Exactness of semidefinite relaxations for nonlinear optimization problems with underlying graph structure. *SIAM Journal on Optimization*, 24(4): 1746–1778, 2014. doi: 10.1137/130915261.

[27] J. Su, D. V. Vargas, and K. Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, 2019.

[28] V. Tjeng, K. Xiao, and R. Tedrake. Evaluating robustness of neural networks with mixed integer programming. *arXiv preprint arXiv:1711.07356*, 2017.

[29] A. L. Wang and F. Kılınç-Karzan. On the tightness of SDP relaxations of QCQPs. *Mathematical Programming*, 2021. doi: 10.1007/s10107-020-01589-9.

[30] E. Wong and Z. Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International conference on machine learning*, pages 5286–5295. PMLR, 2018.

[31] R. Zhang. On the tightness of semidefinite relaxations for certifying robustness to adversarial examples. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 3808–3820. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/27b587bbe83aecf9a98c8fe6ab48cacc-Paper.pdf.