A UNIFIED ALGEBRAIC PERSPECTIVE ON LIPSCHITZ NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Important research efforts have focused on the design and training of neural networks with a controlled Lipschitz constant. The goal is to increase and sometimes guarantee the robustness against adversarial attacks. Recent promising techniques draw inspirations from different backgrounds to design 1-Lipschitz neural networks, just to name a few: convex potential layers derive from the discretization of continuous dynamical systems, Almost-Orthogonal-Layer proposes a tailored method for matrix rescaling. However, it is today important to consider the recent and promising contributions in the field under a common theoretical lens to better design new and improved layers. This paper introduces a novel algebraic perspective unifying various types of 1-Lipschitz neural networks, including the ones previously mentioned, along with methods based on orthogonality and spectral methods. Interestingly, we show that many existing techniques can be derived and generalized via finding analytical solutions of a common semidefinite programming (SDP) condition. We also prove that AOL biases the scaled weight to the ones which are close to the set of orthogonal matrices in a certain mathematical manner. Moreover, our algebraic condition, combined with the Gershgorin circle theorem, readily leads to new and diverse parameterizations for 1-Lipschitz network layers. Our approach, called SDP-based Lipschitz Layers (SLL), allows us to design non-trivial yet efficient generalization of convex potential layers. Finally, the comprehensive set of experiments on image classification shows that SLLs outperforms previous approaches on natural and certified accuracy.

1 INTRODUCTION

Robustness of deep neural networks is nowadays a great challenge to establish confidence in their decisions for real-life applications. Addressing this challenge requires guarantees on the stability of the prediction, with respect to adversarial attacks. In this context, the Lipschitz constant of neural networks is a key property at the core of many recent advances. Along with the margin of the classifier, this property allows us to certify the robustness against worst-case adversarial perturbations. This certification is based on a sphere of stability within which the decision remains the same for any perturbation inside the sphere. (Tsuzuku et al., 2018).

The design of 1-Lipschitz layers provides a successful approach to enforce this property for the whole neural network. For this purpose, many different techniques have been devised such as spectral normalization (Miyato et al., 2018; Farnia et al., 2019), orthogonal parameterization (Trockman et al., 2021; Li et al., 2019; Singla & Feizi, 2021), Convex Potential Layers (CPL) (Meunier et al., 2022), and Almost-Orthogonal-Layers (AOL) (Prach et al., 2022). While all these techniques share the same goal, their motivations and derivations can greatly differ, delivering different solutions. Nevertheless, their raw experimental comparison fails to really gain insight on their peculiar performance, soundness and at the end their possible complementarity. Therefore a question acts as a barrier for an in-depth analysis and future development:

Are there common principles underlying the developments of 1-Lipschitz Layers?

In this paper, we propose a novel perspective to answer this question based on a unified Semidefinite Programming (SDP) approach. We introduce a common algebraic condition underlying var-

ious types of methods like spectral normalization, orthogonality-based methods, AOL, and CPL. Our key insight is that this condition can be formulated as a unifying and simple SDP problem, and that the development of 1-Lipschitz architectures systematically arises by finding “analytical solutions" of this SDP. Our main contributions are summarized as follows.

- We provide a unifying algebraic perspective for 1-Lipschitz network layers by showing that existing techniques such as spectral normalization, orthogonal parameterization, AOL, and CPL can all be recast as a solution of a same simple SDP condition (Theorem 1 and related discussions). Consequently, any new analytical solutions of our proposed SDP condition will immediately lead to new 1-Lipschitz network structures.
- Built upon the above algebraic viewpoint, we give a rigorous mathematical interpretation for AOL explaining how this method promotes “almost orthogonality" in training (Theorem 2).
- Based on our SDPs, a new family of 1-Lipschitz network structures termed as SDP-based Lipschitz layers (SLL) has been developed. Specifically, we apply the Gershgorin circle theorem to obtain some new SDP solutions, leading to non-trivial extensions of CPL (Theorem 3). We derive new SDP conditions to characterize SLL in a very general form (Theorem 4).
- Finally, we show, by a comprehensive set of experiments, that our new SDP-based Lipschitz layers outperform previous approaches on natural and certified accuracy.

Our work is inspired by Fazlyab et al. (2019) that develops SDP conditions for numerical estimation of Lipschitz constants of given neural networks. A main difference is that we focus on “analytical SDP solutions" which can be used to characterize 1-Lipschitz network structures.

2 Related Work

In recent years, certified methods have been central to the development of trustworthy machine learning and especially for deep leaning. *Randomized Smoothing* (Cohen et al., 2019; Salman et al., 2019) is one of the first defense to offer provable robustness guarantees. The method simply extends a given classifier by the smart introduction of random noise to enhance the robustness of the classifier. Although this method offers an interesting level of certified robustness, it suffers from important downsides such as the high computational cost of inference and some impossibility results from information-theory perspective (Yang et al., 2020; Kumar et al., 2020).

Another approach to certify the robustness of a classifier is to control its Lipschitz constant. The main idea is to derive a certified radius in the feature space by computing the margin of the classifier (*e.g.*, the difference between the highest logits and second highest). See Proposition 1 of Tsuzuku et al. (2018) for more details. This radius, along with Lipschitz constant of the network can certify the robustness. One of the first approaches in this direction consists of normalizing each layer with its spectral norm (Miyato et al., 2018; Farnia et al., 2019). Each layer is, by construction, 1-Lipschitz. Later, a body of research replaces the normalized weight matrix by an orthogonal matrix. It improves upon the spectral normalization method by adding the gradient preservation (Li et al., 2019; Trockman et al., 2021; Singla & Feizi, 2021). These methods constrain the parameters by orthogonality during training. More specifically, the Cayley transform can be used to constrain the weights (Trockman et al., 2021) and, in a similar fashion, SOC (Singla & Feizi, 2021) parameterizes their layers with the exponential of a skew symmetric matrix making it orthogonal. To reduce computational cost, Trockman et al. (2021) and Yu et al. (2022) orthogonalize their convolutional kernel in the Fourier domain.

More recently, a work by Meunier et al. (2022) has studied Lipschitz networks from a dynamical system perspective. Starting from the continuous view of a residual network, they showed that the parameterization with the Cayley transform (Trockman et al., 2021) and exponential matrix (SOC) (Singla & Feizi, 2021) correspond respectively to two specific discretization schemes of the continuous flow. Furthermore, a new layer is introduced that derives from convex potential flows to ensure the 1-Lipschitz property[1]:

$$z = x - \frac{2}{\|W\|_2^2} W\sigma(W^\top x + b), \tag{1}$$

[1]We reverse the transposition from the original layer to have a consistent notation in the rest of the article.

where $\|W\|_2$ is the spectral norm of the weight matrix W and σ is the ReLU activation function. Although orthogonal layers enjoy great stability due to their gradient preservation, they make training computationally expensive and difficult. Indeed, orthogonalization via the Cayley transform involves a matrix inversion, and the implementation of the matrix exponential requires either an SVD or an iterative Taylor expansion. While more efficient, the CPL approach still requires computation of the largest singular value.

A recent work, *Almost-Orthogonal-layer* (AOL) (Prach et al., 2022) came up with a middle ground: a new normalization which makes the layer 1-Lipschitz by favoring orthogonality. More precisely, the fully-connected AOL layer is defined as follows:

$$z = WDx + b \tag{2}$$

where D is a diagonal matrix defined as: $D = \text{diag}\left(\sum_j |W^\top W|_{ij}\right)^{-\frac{1}{2}}$. They demonstrated that this layer is guaranteed to be 1-Lipschitz and they empirically show that, after training, the Jacobian of the layer (with respect to x) is almost orthogonal, hence facilitating the training.

Another source of inspiration is the application of SDPs for robustness certification of neural networks. The first work on this topic (Wong & Kolter, 2018) consists of a robust optimization procedure that minimizes the worst-case loss over a convex outer adversarial polytope via a linear program. Then the work of Fazlyab et al. (2020) combined quadratic constraints and SDPs to analyze local robustness of neural networks. Finally, Fazlyab et al. (2019) proposed various SDP conditions which can be numerically solved to give efficient estimates of the global Lipschitz constant of neural networks. It is also possible to solve similar SDPs numerically for training relatively small Lipschitz networks on MNIST (Pauli et al., 2021). However, due to the restrictions of existing SDP solvers, scalability has been one issue when deploying such approaches to deep learning problems with large data sets. Our focus is on the design of Lipschitz network structures, and hence the scalability issue can be avoided via constructing analytical SDP solutions.

3 Background

Notation. The $n \times n$ identity matrix and the $n \times n$ zero matrix are denoted as I_n and 0_n, respectively. The subscripts will be omitted when the dimension is clear from the context. When a matrix P is negative semidefinite (definite), we will use the notation $P \preceq (\prec) 0$. When a matrix P is positive semidefinite (definite), we will use the notation $P \succeq (\succ) 0$. Let e_i denote the vector whose i-entry is 1 and all other entries are 0. Given a collection of scalars $\{a_i\}_{i=1}^n$, we use the notation $\text{diag}(a_i)$ to denote the $n \times n$ diagonal matrix whose (i,i)-th entry is a_i. For a matrix A, the following notations $A^\top$, $\|A\|_2$, $\text{tr}(A)$, $\sigma_{\min}(A)$, $\|A\|_F$, and $\rho(A)$ stand for its transpose, largest singular value, trace, smallest singular value, Frobenius norm, and spectral radius, respectively.

Lipschitz functions. A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is L-Lipschitz with respect to the ℓ_2 norm iff it satisfies $\|f(x) - f(y)\| \leq L\|x - y\|$ for all $x, y \in \mathbb{R}^n$, where $\|\cdot\|$ stands for the ℓ_2 norm. The smallest possible value of L gives the so-called Lipschitz constant. An important fact is that the robustness of a neural network can be certified based on its Lipschitz constant (Tsuzuku et al., 2018). In this paper, we are particularly interested in the case where $L = 1$. Specifically, we consider the training of 1-Lipschitz neural networks. If each layer of a neural network is 1-Lipschitz, then the entire neural network is also 1-Lipschitz[2]. The Lipschitz constant also satisfies the triangle inequality, and hence convex combination will preserve the 1-Lipschitz property.

Matrix cones: Positive semidefiniteness and diagonal dominance. Let $\mathbf{S}^n$ denote the set of all $n \times n$ real symmetric matrices. Let $\mathbf{S}^n_+ \subset \mathbf{S}^n$ be the set of all $n \times n$ symmetric positive semidefinite matrices. It is well known that $\mathbf{S}^n_+$ is a closed pointed convex cone in $\mathbf{S}^n$. With the trace inner product, $\mathbf{S}^n_+$ is also self-dual. Consider two symmetric matrices A and B such that $A \succeq B \in \mathbf{S}^n$, then we have $A - B \in \mathbf{S}^n_+$, and $\text{tr}(A - B)$ provides a distance measure between A and B. In addition, we have $\|A - B\|_F \leq \text{tr}(A - B)$. Finally, the set of all $n \times n$ real symmetric diagonally dominant matrices with non-negative diagonal entries is represented by $\mathbf{D}^n$. It is known that $\mathbf{D}^n$ forms a closed, pointed, full cone (Barker & Carlson, 1975). Based on the Gershgorin circle theorem (Horn & Johnson, 2012), we know $\mathbf{D}^n \subset \mathbf{S}^n_+$. It is also known that $\mathbf{D}^n$ is smaller than $\mathbf{S}^n_+$ (Barker &

[2]Here we assume the activation function is 1-Lipschitz. This is true for ReLU, tanh, and sigmoid.

Carlson, 1975). For any $A \in \mathbf{D}^n$, we have $A_{ii} \geq \sum_{j:j\neq i} |A_{ij}|$. It is important to require $A_{ii} \geq 0$, and the set of real symmetric diagonally dominant matrices is not a cone by itself.

4 An Algebraic Unification of 1-Lipschitz Layers

In this section, we present a unified algebraic perspective for various 1-Lipschitz layers (Spectral Normalization, Orthogonalization, AOL, and CPL) via developing a common SDP condition characterizing the Lipschitz property. Built upon our algebraic viewpoint, we also present a new mathematical interpretation explaining how AOL promotes orthogonality in training.

4.1 The unifying Algebraic Condition

First, we present an algebraic condition which can be used to unify the developments of existing techniques such as SN, AOL, and CPL. Our main theorem is formalized below.

Theorem 1. *For any weight matrix $W \in \mathbb{R}^{m\times n}$, if there exists a nonsingular diagonal matrix T such that $W^\top W - T \preceq 0$, then the two following statements hold true.*

1. *The mapping $g(x) = WT^{-\frac{1}{2}}x + b$ is 1-Lipschitz.*
2. *The mapping $h(x) = x - 2WT^{-1}\sigma(W^\top x + b)$ is 1-Lipschitz if σ is ReLU,* tanh *or sigmoid.*

Proof. To prove the first statement, notice that we have

$$\|g(x) - g(y)\|^2 = \|WT^{-\frac{1}{2}}(x-y)\|^2 = (x-y)^\top T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}}(x-y).$$

Based on our algebraic condition $W^\top W \preceq T$, we immediately have

$$\|g(x) - g(y)\|^2 \leq (x-y)^\top T^{-\frac{1}{2}} T T^{-\frac{1}{2}}(x-y) = \|x-y\|^2.$$

Therefore, Statement 1 is true.

To prove Statement 2, we need to use the property of the nonlinear activation function σ. Notice that the condition $W^\top W \preceq T$ ensures that all the diagonal entries of the nonsingular matrix T are positive. Therefore, T^{-1} is also a diagonal matrix whose diagonal entries are all positive. For all the three activation functions listed in the above theorem, σ is slope-restricted on $[0,1]$, and the following inequality holds for any $\{x', y'\}$ (Fazlyab et al., 2019, Lemma 1):

$$\begin{bmatrix} x'-y' \\ \sigma(x')-\sigma(y') \end{bmatrix}^\top \begin{bmatrix} 0 & -T^{-1} \\ -T^{-1} & 2T^{-1} \end{bmatrix} \begin{bmatrix} x'-y' \\ \sigma(x')-\sigma(y') \end{bmatrix} \leq 0.$$

We can set $x' = W^\top x + b$ and $y' = W^\top y + b$, and the above inequality becomes

$$\begin{bmatrix} W^\top(x-y) \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix}^\top \begin{bmatrix} 0 & -T^{-1} \\ -T^{-1} & 2T^{-1} \end{bmatrix} \begin{bmatrix} W^\top(x-y) \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix} \leq 0.$$

We can rewrite the above inequality as

$$\begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix}^\top \begin{bmatrix} 0 & -WT^{-1} \\ -T^{-1}W^\top & 2T^{-1} \end{bmatrix} \begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix} \leq 0. \tag{3}$$

Now we can apply the following argument:

$$\begin{aligned}
&\|h(x)-h(y)\|^2 \\
=&\|x-y-2\Big(WT^{-1}\sigma(W^\top x+b) - WT^{-1}\sigma(W^\top y+b)\Big)\|^2 \\
=&\begin{bmatrix} x-y \\ 2WT^{-1}\Big(\sigma(W^\top x+b)-\sigma(W^\top y+b)\Big) \end{bmatrix} \begin{bmatrix} I & -I \\ -I & I \end{bmatrix} \begin{bmatrix} x-y \\ 2WT^{-1}\Big(\sigma(W^\top x+b)-\sigma(W^\top y+b)\Big) \end{bmatrix} \\
=&\begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix}^\top \begin{bmatrix} I & -2WT^{-1} \\ -2T^{-1}W^\top & 4T^{-1}W^\top W T^{-1} \end{bmatrix} \begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix} \\
\leq&\begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix}^\top \begin{bmatrix} I & -2WT^{-1} \\ -2T^{-1}W^\top & 4T^{-1} \end{bmatrix} \begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix},
\end{aligned}$$

where the last step follows from the fact that our condition $W^\top W \preceq T$ implies $T^{-1}W^\top W T^{-1} \preceq T^{-1}$. Finally, we can combine the above inequality with (3) to show

$$\begin{aligned}\|h(x)-h(y)\|^2 &\leq \begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b)\end{bmatrix}^\top \begin{bmatrix} I & 0 \\ 0 & 0\end{bmatrix} \begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b)\end{bmatrix} \\ &= \|x-y\|^2,\end{aligned}$$

which is the desired conclusion. □

This theorem allows us to design different 1-Lipschitz layers just with various choices of T, in two important cases: for a linear transformation with Statement 1, as well as for a residual and non-linear block with Statement 2. Moreover, for any weight matrix W, the condition $W^\top W \preceq T$ is linear in T, and hence can be viewed as an SDP condition with decision variable T. To emphasize the significance of this theorem, we propose to derive existing methods used for designing 1-Lipschitz layers by choosing specific T for the SDP condition $W^\top W \preceq T$. The 1-Lipschitz property is then automatically obtained.

- Spectral Normalization (SN) corresponds to an almost trivial choice, if we notice that $W^\top W \preceq \|W^\top W\|_2 I \preceq \|W\|_2^2 I$. Hence with $T = \|W\|_2^2 I$, we build the SN layer $g(x) = WT^{-\frac{1}{2}}x + b = \frac{1}{\|W\|_2} Wx + b$
- The Orthogonality-based parameterization is obtained by setting $T = I$ and enforcing the equality $W^\top W = T = I$. Then obviously $g(x) = Wx + b$ is 1-Lipschitz.
- AOL formula can be derived by letting $T = \text{diag}(\sum_{j=1}^n |W^\top W|_{ij})$. With this choice, we have $T - W^\top W \in \mathbf{D}^n \subset \mathbf{S}_+^n$, hence $W^\top W \preceq T$. Then Statement 1 in Theorem 1 implies that the AOL function of equation (2)), written as $g(x) = WT^{-\frac{1}{2}}x + b$, is 1-Lipschitz.
- CPL follows the same choice SN, $T = \|W\|_2^2 I$, but with Statement 2 of Theorem 1, we derive a different function $h(x) = x - \frac{2}{\|W\|_2^2} W\sigma(W^\top x + b)$ which is therefore 1-Lipschitz.

The above discussion illustrates the benefit of expressing all these methods within the same theoretical framework, offering us a new tool to characterize the similarity between different methods. For instance, SN and CPL share the same choice of $T = \|W\|_2^2 I$. The difference between them is which statement is used . Therefore, CPL can be viewed as the "residual version" of SN. Clearly, the residual network structure allows CPL to address the gradient vanishing issue more efficiently than SN. With the same approach, we can readily infer from our unified algebraic condition what are the "residual" counterparts for orthogonality-based parameterization and AOL. For orthogonality-based parameterization, if we enforce $W^\top W = T = I$ via methods such as SOC and ECO, then the function $h(x) = x - 2W\sigma(W^\top x + b)$ is 1-Lipschitz (by Statement 2 in Theorem 1). Finally, if we choose $T = \text{diag}\left(\sum_{j=1}^n |W^\top W|_{ij}\right)$, then the function $h(x) = x - 2W\,\text{diag}\left(\sum_{j=1}^n |W^\top W|_{ij}\right)^{-1} \sigma(W^\top x + b)$ is also 1-Lipschitz. Therefore it is straightforward to create new classes of 1-Lipschitz network structures from existing ones.

Another important consequence of Theorem 1 is about new layer development. Any new diagonal solution T for the SDP condition $W^\top W - T \preceq 0$ immediately leads to new 1-Lipschitz network structures in the form of $g(x) = WT^{-\frac{1}{2}}x + b$ or $h(x) = x - 2WT^{-1}\sigma(W^\top x + b)$. Therefore, the developments of 1-Lipschitz network structures can be reformulated as finding analytical solutions of the matrix inequality $W^\top W \preceq T$ with nonsingular diagonal T. As a matter of fact, the Gershgorin circle theorem can help to improve the existing choices of T in a systematic way. In Section 5, we will discuss such new choices of T and related applications to improve CPL. At this point, it is worth noticing that to develop deep Lipschitz networks, it is important to have analytical formulas of T. The goal is to get a fast computation of $WT^{-\frac{1}{2}}$ or WT^{-1}. Therefore, analytical SDP solutions are particularly relevant for the purpose of our paper.

Theorem 1 is powerful in building a connection between 1-Lipschitz network layers and the algebraic condition $W^\top W \preceq T$. Next, we will look closer at this algebraic condition and provide a new mathematical interpretation explaining how AOL generates "almost orthogonal" weights.

4.2 A NEW MATHEMATICAL INTERPRETATION FOR AOL

In Prach et al. (2022), it is observed that AOL can learn "almost orthogonal" weights and hence overcome the gradient vanishing issue. As a matter of fact, the choice of T used in AOL is optimal in a specific mathematical sense as formalized with the next theorem.

Theorem 2. *Given any W, define the set $\mathbf{T} = \left\{T : T \text{ is nonsingular diagonal, and } T - W^\top W \in \mathbf{D}^n\right\}$. Then the choice of T for the AOL method actually satisfies*

$$T = \text{diag}(\sum_{j=1}^{n} |W^\top W|_{ij}) = \arg\min_{T\in\mathbf{T}} \text{tr}(I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}}) = \arg\min_{T\in\mathbf{T}} \| T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}} - I\|_F. \quad (4)$$

Before presenting the proof, we first provide some interpretations for the above result. Obviously, the quantity $\| T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}} - I\|_F$ provides a measure for the distance between the scaled weight matrix $WT^{-\frac{1}{2}}$ and the set of $n \times n$ orthogonal matrices. If $\| T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}} - I\|_F = 0$, then the scaled weight $WT^{-\frac{1}{2}}$ is orthogonal. If $\| T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}} - I\|_F$ is small, it means that $WT^{-\frac{1}{2}}$ is "almost orthogonal" and close to the set of orthogonal matrices. Since we require $W^\top W - T \preceq 0$, we know that $I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}}$ is a positive semidefinite matrix, and that its trace provides an alternative metric quantifying the distance between $WT^{-\frac{1}{2}}$ and the set of orthogonal matrices. Importantly, we have the following inequality:

$$\| T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}} - I\|_F \le \text{tr}(I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}}).$$

If $\text{tr}(I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}})$ is small, then $\| T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}} - I\|_F$ is also small, and $WT^{-\frac{1}{2}}$ is close to the set of orthogonal matrices. Therefore, one interpretation for Theorem 2 is that among all the nonsingular diagonal scaling matrices T satisfying $T - W^\top W \in \mathbf{D}^n$, the choice of T used in AOL makes the scaled weight matrix $WT^{-\frac{1}{2}}$ the closest to the set of orthogonal matrices. This provides a new mathematical explanation of how AOL can generate "almost orthogonal" weights. Now we briefly discuss the proof idea for Theorem 2 to further illustrate the interesting underlying algebraic structure of AOL.

Proof of Theorem 2. Since T is nonsingular diagonal and $T - W^\top W \in \mathbf{D}^n$, then we must have $T_{ii} \ge \sum_j |W^\top W|_{ij}$. Given the following key relation:

$$\text{tr}(I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}}) = \sum_i \left(1 - \frac{|W^\top W|_{ii}}{T_{ii}}\right),$$

it becomes clear that we need to choose the smallest value of T_{ii} for all i to minimize $\text{tr}(I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}})$. Therefore the choice of T for AOL minimizes $\text{tr}(I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}})$ over $T \in \mathbf{T}$. The proof for the last of equation 4 is similar. Let us enote $X = I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}}$. For any (i, j), the quantity X_{ij}^2 is always monotone non-decreasing in T_{ii} and T_{jj}. To minimize $\|X\|_F$, we just need to choose the smallest value for all T_{ii} under the constraint $T_{ii} \ge \sum_j |W^\top W|_{ij}$. This completes the proof. □

One potential issue for AOL is that $\mathbf{D}_n$ is typically much smaller than $\mathbf{S}_+^n$, and the condition $T - W^\top W \in \mathbf{D}^n$ may be too conservative compared to the original condition $T - W^\top W \in \mathbf{S}_+^n$ in Theorem 1. If we denote the set $\hat{\mathbf{T}} = \left\{T : T \text{is nonsingular diagonal, and } T - W^\top W \in \mathbf{S}_+^n\right\}$, then we have $\arg\min_{T\in\hat{\mathbf{T}}} \text{tr}(I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}}) \le \arg\min_{T\in\mathbf{T}} \text{tr}(I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}})$, and $\arg\min_{T\in\hat{\mathbf{T}}} \| T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}} - I\|_F \le \arg\min_{T\in\mathbf{T}} \| T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}} - I\|_F$. This leads to interesting alternative choices of T which can further promote orthogonality:

$$T = \arg\min_{T\in\hat{\mathbf{T}}} \| T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}} - I\|_F \quad \text{or} \quad T = \arg\min_{T\in\hat{\mathbf{T}}} \text{tr}(I - T^{-\frac{1}{2}} W^\top W T^{-\frac{1}{2}}) \quad (5)$$

Although (5) may be solved as convex programs or seven SDPs on small toy examples, currently it is not implementable to use such choice of T for practical large-scale problems. It is our hope that our theoretic discussion above will inspire more future research on developing new practical choices of T for promoting orthogonality.

5 EXTENSIONS OF CPL: THE POWER OF GERSHGORIN CIRCLE THEOREM

In this section, we extend the original CPL layer (6) to a new family of 1-Lipschitz network structures via solving SDPs. We term this general family of layers as SDP-based Lipschitz layers (SLL), since the condition $W^\top W \preceq T$ can be viewed as an SDP for the decision variable T. First of all, we extend the existing CPL (Eq. (1)) via applying more general choices of T with Theorem 1. From the discussion after Theorem 1, we already know that we can use the choice of $T = \text{diag}(\sum_{j=1}^n |W^\top W|_{ij})$ to replace the original choice $T = \|W\|_2^2 I$. In this section, we will strengthen CPL via an even more general choice of T, which is based on a special version of Gershgorin circle theorem. Specifically, we will apply (Horn & Johnson, 2012, Corollary 6.1.6) to show the following result.

Theorem 3. *Let W be the weight matrix. Suppose T is a nonsingular diagonal matrix. If there exists some diagonal matrix Q with all positive diagonal entries such that $(T - QW^\top WQ^{-1})$ is a real diagonally dominant matrix with diagonal entries being all positive, then $T \succeq W^\top W$, and the function $h(x) = x - 2WT^{-1}\sigma(W^\top x + b)$ is 1-Lipschitz for σ being ReLU,* tanh *or sigmoid.*

Proof. Given nonsingular matrix Q, clearly the eigenvalues of $Q(T - W^\top W)Q^{-1}$ and $(T - W^\top W)$ are the same. If $Q(T - W^\top W)Q^{-1}$ is diagonally dominant and only has positive diagonal entries, then we can apply Gershgorin circle theorem (Horn & Johnson, 2012, Corollary 6.1.6) to show that all the eigenvalues of $Q(T - W^\top W)Q^{-1}$ (which is the same as $T - QW^\top WQ^{-1}$) are non-negative. Therefore, we know that all the eigenvalues of $(T - W^\top W)$ are non-negative. Since $(T - W^\top W)$ is symmetric, we have $T \succeq W^\top W$. Then we can apply Theorem 1 to reach our desired conclusion. □

If we choose $Q = I$, the above theorem just recovers the choice of T used in AOL, i.e. $T = \text{diag}(\sum_{j=1}^n |W^\top W|_{ij})$. However, it is expected that the use of more general Q will allow us to train a less conservative 1-Lipschiz neural network due to the increasing expressivity brought by these extra variables. We will present numerical results to demonstrate this. We also emphasize that $(T - QW^\top WQ^{-1})$ is typically not a symmetric matrix and hence is not in $\mathbf{D}^n$ even when it only has non-negative eigenvalues. However, this does not affect our proof on the positive-semidefiniteness of $(T - W^\top W)$.

Application of Theorem 3. We can parameterize $Q^{-1} = \text{diag}(q_i)$ with $q_i > 0$. Then the (i, j)-th entry of $QW^\top WQ^{-1}$ is equal to $(W^\top W)_{ij} q_j / q_i$. Therefore, we can simply choose the diagonal entry of T as

$$T_{ii} = \sum_{j=1}^n |(W^\top W)_{ij} q_j / q_i| = \sum_{j=1}^n |W^\top W|_{ij} \frac{q_j}{q_i}.$$

This leads to our new choice of $T = \text{diag}(\sum_{j=1}^n |W^\top W|_{ij} q_j / q_i)$. Notice that the layer function $h(x) = x - 2WT^{-1}\sigma(W^\top x + b)$ has a residual network structure. Hence it is expected that vanishing gradient will not be an issue. Therefore, we can simultaneously optimize the training loss over W and $\{q_i\}$. We will present numerical study to demonstrate that such a training approach will allow us to generate competitive results on training robust CIFAR10/CIFAR100 classifiers.

SDP conditions for more general network structures. It is also worth mentioning that the SDP condition in Theorem 1 can be generalized to address more complex network structures. Specifically, we can allow SLL to adopt the following general structure:

$$h(x) = Hx + G\sigma(W^\top x + b), \tag{6}$$

where H and G will be determined by the weight W in some manner, and the matrix dimensions are assumed to be compatible. If we choose $H = I$ and $G = -2WT^{-1}$, then (6) reduces to the residual network structure considered in Theorem 1. There are many other choices of (H, G) which can also ensure the Lipschitz constant of (6) to be smaller than or equal to 1. Our last theoretical result is a general SDP condition which generalizes Theorem 1 and provides a more comprehensive characterization of such choices of (H, G).

Table 1: This table presents the natural and provable accuracy of several concurrent work and our architecture based on SLL on CIFAR10 and CIFAR100 datasets. All results for SLL-based networks are the result of the average of 3 trainings.

Datasets	Models	Natural Accuracy	Provable Accuracy (ε)			
			$\frac{36}{255}$	$\frac{72}{255}$	$\frac{108}{255}$	1
CIFAR10	**Cayley Large** (Trockman et al., 2021)	74.6	61.4	46.4	32.1	-
	SOC 20 (Singla & Feizi, 2021)	78.0	62.7	46.0	30.3	-
	SOC+ 20 (Singla et al., 2022)	76.3	62.6	48.7	36.0	-
	CPL XL (Meunier et al., 2022)	78.5	64.4	48.0	33.0	-
	AOL Large (Prach et al., 2022)	71.6	64.0	56.4	49.0	23.7
	SLL Small	73.3	63.7	53.8	44.5	15.3
	SLL Medium	74.0	64.7	54.9	45.3	16.0
	SLL Large	74.6	65.3	55.2	45.8	16.2
	SLL X-Large	75.3	65.7	55.8	46.1	16.3
CIFAR100	**Cayley Large** (Trockman et al., 2021)	43.3	29.2	18.8	11.0	-
	SOC 20 (Singla & Feizi, 2021)	48.3	34.4	22.7	14.2	-
	SOC+ 20 (Singla et al., 2022)	47.8	34.8	23.7	15.8	-
	CPL XL (Meunier et al., 2022)	47.8	33.4	20.9	12.6	-
	AOL Large (Prach et al., 2022)	43.7	33.7	26.3	20.7	7.8
	SLL Small	46.7	35.2	26.4	20.1	5.9
	SLL Medium	47.2	36.1	27.1	20.7	6.5
	SLL Large	47.9	36.7	27.9	21.3	6.7
	SLL X-Large	48.3	37.2	28.3	21.8	6.9

Theorem 4. *Let n be the neuron number. For any non-negative scalars $\{\lambda_{ij}\}$, define*

$$\Lambda = \mathrm{diag}(\lambda_{11}, \lambda_{22}, \ldots, \lambda_{nn}) + \sum_{1 \le i \le j \le n} \lambda_{ij}(e_i - e_j)(e_i - e_j)^\top. \tag{7}$$

Suppose the activation function σ is ReLU or tanh *or sigmoid. If there exist non-negative scalars $\{\lambda_{ij}\}$ such that the following matrix inequality holds*

$$\begin{bmatrix} I - H^\top H & -H^\top G - W\Lambda \\ -G^\top H - \Lambda W^\top & 2\Lambda - G^\top G \end{bmatrix} \succeq 0 \tag{8}$$

then the network (6) *is* 1*-Lipschitz, i.e. $\|h(x) - h(y)\| \le \|x - y\|$ for all (x, y).*

The above theorem can be proved via modifying the argument used in (Fazlyab et al., 2019, Theorem 1), and we defer the detailed proof to the appendix. On one hand, if we choose $H = 0$, then our condition (8) reduces to (Fazlyab et al., 2019, Theorem 1)[3]. On the other hand, for residual network structure with $H = I$, we can choose $T = 0.5\Lambda^{-1}$ and $G = -W\Lambda = -2WT^{-1}$ to reduce (8) to our original algebraic condition $T \succeq W^\top W$. Therefore, Theorem 4 provides a connection between the SDP condition in Fazlyab et al. (2019) and our proposed simple algebraic condition in Theorem 1. It is possible to obtain new 1-Lipschitz network layers via providing new analytical solutions to (8). It is our hope that our proposed SDP condition (8) can lead to many more 1-Lipschitz network structures in the future.

6 Experiments

In this section we present a comprehensive set of experiments with 1-Lipschitz neural networks based on our proposed *SDP-based Lipschitz Layer*. More specifically, we build 1-Lipschitz neural networks based on the following layer:

$$h(x) = x - 2W\,\mathrm{diag}\left(\sum_{j=1}^{n} |W^\top W|_{ij} q_j / q_i\right)^{-1} \sigma(W^\top x + b) \tag{9}$$

[3]To see this connection, set the parameters in (Fazlyab et al., 2019, Theorem 1) as $(\alpha, \beta, W^0, W^1) = (0, 1, W^\top, G)$.

Table 2: The table describes the empirical robustness of our SLL-based classifiers on CIFAR10 ans CIFAR100 datasets. The empirical robustness is measured with *AutoAttacks*. All results are the average of 3 models.

Models	CIFAR10 – *AutoAttack* (ε)				CIFAR100 – *AutoAttack* (ε)			
	$\frac{36}{255}$	$\frac{72}{255}$	$\frac{108}{255}$	1	$\frac{36}{255}$	$\frac{72}{255}$	$\frac{108}{255}$	1
SLL Small	68.1	62.5	56.8	35.0	40.7	35.2	30.4	17.0
SLL Medium	69.1	63.8	58.4	37.0	41.5	36.4	31.5	17.9
SLL Large	69.8	64.5	59.1	37.9	42.1	37.1	32.6	18.7
SLL X-Large	70.3	65.4	60.2	39.4	42.7	37.8	33.2	19.5

where W is a parameter matrix being either dense or a convolution, $\{q_i\}$ forms a diagonal scaling matrix as described by Theorem 3, and $\sigma(\cdot)$ is the ReLU nonlinearity function. We use the same architectures proposed by Meunier et al. (2022) with *small*, *medium*, *large* and *xlarge* sizes. The architecture consists of several Conv-SLL and Linear-SLL. We provide details of the architectures in Table 3 in the Appendix. For CIFAR-100, we use the Last Layer Normalization proposed by Singla et al. (2022) which improves the certified accuracy when the number of classes becomes large. Note that the layer presented in Equation 9 can be easily implemented with convolutions following the same scaling as in Prach et al. (2022).

Hyper-parameters. We trained our networks with a batch size of 256 over 1000 epochs with the data augmentation used by Prach et al. (2022) (*i.e.*, random cropping, flipping and color transformation). We use an Adam optimizer Kingma et al. (2014) with 0.01 learning rate and parameters β_1 and β_2 equal to 0.5 and 0.9 respectfully and no weight decay. We use a piecewise triangular learning rate scheduler to decay the learning rate during training. We use the CrossEntropy loss as in Prach et al. (2022) with an offset and temperature parameters of $\sqrt{2}$ and 0.25 respectively.

Results in terms of Natural and Certified Accuracy. We evaluate our networks on CIFAR10 and CIAFR100 datasets (Krizhevsky et al., 2009) and compare the results against very recent approaches: Cayley (Trockman et al., 2021), SOC (Singla & Feizi, 2021), SOC+ (Singla et al., 2022), CPL (Meunier et al., 2022) and AOL (Prach et al., 2022). Table 1 presents the natural and certified accuracy with different radius of certification (*i.e.*, 36/255, 72/255, 108/255 and 1). Our approach outperforms the AOL-based networks on natural accuracy and on certified accuracy for $\varepsilon = 36/255$, also, our approach outperforms the CPL-based networks on certified accuracy for all values of ε. As a result, SLL-based 1-Lipschitz neural networks offer a good trade-off among previous approaches with respect to natural and certified accuracy.

Results on Empirical Robustness. We provide results of our approach on empirical robustness against an ensemble of diverse parameter-free attacks (*i.e.*, *AutoAttacks*) developed by Croce et al. (2020). Table 2 reports the empirical robustness accuracy for different levels of perturbations. Although *AutoAttacks* is a strong empirical attack consisting of an ensemble of several known attacks: APGD_{CE}, APGD_{DLR}, FAB (Croce & Hein, 2020) and Square (Andriushchenko et al., 2020). We can observe that the measure robustness is high and well above the certified radius. Indeed, on CIFAR10, we observe a robustness "gain" of up to 4.5%, 9.6%, 14.1% and 21.7% for respectively, 36, 72, 108 and 255 ε-perturbations.

7 Conclusion

In this paper, we present a unifying framework for designing Lipschitz layers. Based on a novel algebraic perspective, we identify a common SDP condition underlying the developments of spectral normalization, orthogonality-based methods, AOL, and CPL. Furthermore, we have shown that AOL and CPL can be re-derived and generalized using our theoretical framework. From this analysis, we introduce a family of SDP-based Lipschitz layers (SLL) that outperforms previous work. Although this work is a first step towards a universal theory on Lipschitz networks, much work remains to achieve the accuracy necessary for deploying these models into real-world applications. For example, investigating more expressive structures of T and extending our contributions to address multi-layer neural networks, allowing the intermediate layer-by-layer Lipschitz constant to be larger than 1, may lead to even greater performance.

REFERENCES

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European Conference on Computer Vision*, 2020.

George Barker and David Carlson. Cones of diagonally dominant matrices. *Pacific Journal of Mathematics*, 57(1):15–32, 1975.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, 2019.

Francesco Croce and Matthias Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack. In *International Conference on Machine Learning*. PMLR, 2020.

Francesco Croce et al. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning*, 2020.

Farzan Farnia, Jesse Zhang, and David Tse. Generalizable adversarial training via spectral normalization. In *International Conference on Learning Representations*, 2019.

Mahyar Fazlyab, Alexander Robey, Hamed Hassani, Manfred Morari, and George Pappas. Efficient and accurate estimation of lipschitz constants for deep neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Mahyar Fazlyab, Manfred Morari, and George J Pappas. Safety verification and robustness analysis of neural networks via quadratic constraints and semidefinite programming. *IEEE Transactions on Automatic Control*, 2020.

R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, 2012. ISBN 9781139788885.

Diederik Kingma et al. Adam: A method for stochastic optimization. In *International Conference for Learning Representations*, 2014.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Aounon Kumar, Alexander Levine, Tom Goldstein, and Soheil Feizi. Curse of dimensionality on randomized smoothing for certifiable robustness. In *International Conference on Machine Learning*, 2020.

Qiyang Li, Saminul Haque, Cem Anil, James Lucas, Roger B Grosse, and Joern-Henrik Jacobsen. Preventing gradient attenuation in lipschitz constrained convolutional networks. In *Advances in Neural Information Processing Systems*, 2019.

Laurent Meunier, Blaise J Delattre, Alexandre Araujo, and Alexandre Allauzen. A dynamical system perspective for lipschitz neural networks. In *International Conference on Machine Learning*, 2022.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

Patricia Pauli, Anne Koch, Julian Berberich, Paul Kohler, and Frank Allgöwer. Training robust neural networks using lipschitz bounds. *IEEE Control Systems Letters*, 6:121–126, 2021.

Bernd Prach et al. Almost-orthogonal layers for efficient general-purpose lipschitz networks. *arXiv preprint arXiv:2208.03160*, 2022.

Hadi Salman, Jerry Li, Ilya Razenshteyn, Pengchuan Zhang, Huan Zhang, Sebastien Bubeck, and Greg Yang. Provably robust deep learning via adversarially trained smoothed classifiers. In *Advances in Neural Information Processing Systems*, 2019.

Sahil Singla and Soheil Feizi. Skew orthogonal convolutions. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Sahil Singla, Surbhi Singla, and Soheil Feizi. Improved deterministic l2 robustness on CIFAR-10 and CIFAR-100. In *International Conference on Learning Representations*, 2022.

Asher Trockman et al. Orthogonalizing convolutional layers with the cayley transform. In *International Conference on Learning Representations*, 2021.

Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks. In *Advances in Neural Information Processing Systems*, 2018.

Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning*, pp. 5286–5295. PMLR, 2018.

Greg Yang, Tony Duan, J Edward Hu, Hadi Salman, Ilya Razenshteyn, and Jerry Li. Randomized smoothing of all shapes and sizes. In *International Conference on Machine Learning*, 2020.

Tan Yu, Jun Li, Yunfeng Cai, and Ping Li. Constructing orthogonal convolutions in an explicit manner. In *International Conference on Learning Representations*, 2022.

A PROOF OF THEOREM 4

A detailed proof for Theorem 4 is presented here. Our proof is based on modifying the arguments used in (Fazlyab et al., 2019, Theorem 1), and mainly relies on the quadratic constraint technique developed in the controls community.

First, notice that (8) is equivalent to the following condition:

$$\begin{bmatrix} H^\top H & H^\top G \\ G^\top H & G^\top G \end{bmatrix} \preceq \begin{bmatrix} I & -W\Lambda \\ -\Lambda W^\top & 2\Lambda \end{bmatrix}. \tag{10}$$

Suppose (10) holds. Next we will show that $h(x) = Hx + G\sigma(W^\top x + b)$ is 1-Lipschitz.

For all the three activation functions listed in the above theorem, σ is slope-restricted on $[0,1]$, and the following inequality holds for any $\{x', y'\}$ (Fazlyab et al., 2019, Lemma 1):

$$\begin{bmatrix} x'-y' \\ \sigma(x')-\sigma(y') \end{bmatrix}^\top \begin{bmatrix} 0 & -\Lambda \\ -\Lambda & 2\Lambda \end{bmatrix} \begin{bmatrix} x'-y' \\ \sigma(x')-\sigma(y') \end{bmatrix} \leq 0.$$

We can set $x' = W^\top x + b$ and $y' = W^\top y + b$, and the above inequality becomes

$$\begin{bmatrix} W^\top(x-y) \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix}^\top \begin{bmatrix} 0 & -\Lambda \\ -\Lambda & 2\Lambda \end{bmatrix} \begin{bmatrix} W^\top(x-y) \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix} \leq 0.$$

We can rewrite the above inequality as

$$\begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix}^\top \begin{bmatrix} 0 & -W\Lambda \\ -\Lambda W^\top & 2\Lambda \end{bmatrix} \begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix} \leq 0. \tag{11}$$

Now we can apply the following argument:

$$\begin{aligned}
&\|h(x)-h(y)\|^2 \\
=&\|H(x-y)+\Big(G\sigma(W^\top x+b)-G\sigma(W^\top y+b)\Big)\|^2 \\
=&\begin{bmatrix} H(x-y) \\ G\Big(\sigma(W^\top x+b)-\sigma(W^\top y+b)\Big) \end{bmatrix} \begin{bmatrix} I & I \\ I & I \end{bmatrix} \begin{bmatrix} H(x-y) \\ G\Big(\sigma(W^\top x+b)-\sigma(W^\top y+b)\Big) \end{bmatrix} \\
=&\begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix}^\top \begin{bmatrix} H^\top H & H^\top G \\ G^\top H & G^\top G \end{bmatrix} \begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix} \\
\leq&\begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix}^\top \begin{bmatrix} I & -W\Lambda \\ -\Lambda W^\top & 2\Lambda \end{bmatrix} \begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix},
\end{aligned}$$

where the last step follows from the condition (10). Finally, we can combine the above inequality with (11) to show

$$\begin{aligned}
\|h(x)-h(y)\|^2 &\leq \begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix}^\top \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x-y \\ \sigma(W^\top x+b)-\sigma(W^\top y+b) \end{bmatrix} \\
&= \|x-y\|^2,
\end{aligned}$$

which is the desired conclusion.

B ADDITIONAL RESULTS AND DISCUSSIONS ON THE EXPERIMENTS

Additional details on the architectures. If $W \in \mathbb{R}^{m\times n}$ then we denote m as the inner dimension. Similarly, when W is a convolution layer the inner dimension is the output channels number.

Table 3 describes various architectures for our SDP-based Lipschitz layers neural network for different size of hyperparameters. Channels refer to the number of channel in convolutional layers, linear features to the inner dimension of W in linear SLL layer, Conv-SLL layers and Linear-SLL Layers refer to the number of layers of those types in the network. They are reported as SLL Small, SLL Medium, SLL Large and SLL X-Large in this paper.

Table 3: The architecture use for the experiments relies on the architecture proposed by Meunier et al. (2022). For completeness we add this table which describes the *small*, *medium*, *large* and *xlarge* architectures.

	S	M	L	XL
Conv-SLL	20	30	90	120
Channels	45	60	60	70
Linear-SLL	7	10	15	15
Linear Features	2048	2048	4096	4096

Table 4: This table describes the time by epochs for training each architecture. We can observe that training SLL-based neural networks is efficient.

Model	Time per epoch (s)
SLL Small	20
SLL Medium	35
SLL Large	55
SLL X-Large	105

Efficiency of SLL-based neural networks. Training computation were done on 4 GPU V100 for all models . We can notice that the training time of SLL network scales well with model size. It is expected as SLL layers are just using regular straight forward convolution and cheap rescaling.

In comparison the Cayley approach builds a convolutional kernel in the Fourier domain which has the size of the input and performs the Cayley transform on it. In regards to SOC, it requires to computes several convolution at training and inference (from 6 to 12) to compute the exponential of a convolution up to a desired precision. These operations can be costly for large inputs and constraint scaling of deep and large architecture. Concerning CPL layer, it uses power method iteration to compute the spectral norm. It is cheap at training but quite expensive at inference to guarantee lipschitzness. Our approach is more simple to implement as it requires no power method algorithm inside the forward pass of our SLL layer.