Optimization Induced Equilibrium Networks: An Explicit Optimization Perspective for Understanding Equilibrium Models

Xingyu Xie*, Qiuhao Wang*, Zenan Ling, Xia Li
Guangcan Liu, *Senior Member, IEEE,* and Zhouchen Lin, *Fellow, IEEE*

Abstract—To reveal the mystery behind deep neural networks (DNNs), optimization may offer a good perspective. There are already some clues showing the strong connection between DNNs and optimization problems, e.g., under a mild condition, DNN's activation function is indeed a proximal operator. In this paper, we are committed to providing a unified optimization induced interpretability for a special class of networks—equilibrium models, i.e., neural networks defined by fixed point equations, which have become increasingly attractive recently. To this end, we first decompose DNNs into a new class of unit layer that is the proximal operator of an implicit convex function while keeping its output unchanged. Then, the equilibrium model of the unit layer can be derived, we name it Optimization Induced Equilibrium Networks (OptEq). The equilibrium point of OptEq can be theoretically connected to the solution of a convex optimization problem with explicit objectives. Based on this, we can flexibly introduce prior properties to the equilibrium points: 1) modifying the underlying convex problems explicitly so as to change the architectures of OptEq; and 2) merging the information into the fixed point iteration, which guarantees to choose the desired equilibrium point when the fixed point set is non-singleton. We show that OptEq outperforms previous implicit models even with fewer parameters.

Index Terms—Equilibrium models, DEQ, Optimization induced models, Interpretability.

1 Introduction

For a long time, DNNs have been regarded as powerful "black boxes" but lacking interpretability. Roughly, all we know is that different DNNs contain some specific inductive biases, such as convolutional structures aiming to extract local features, while Transformers can model global semantics more easily. However, a more detailed way to quantify this different bias remains missing. The implicit models have recently gained significant attention due to their comparable performance and much less memory-consuming. Instead of specifying the explicit forward procedure, the implicit model specifies some conditions held at its output. For Neural ODE [1], the neural network is replaced by an ordinary differential equation (ODE) flow, which can be seen as a continuous version of ResNets [2]. Another instance of such models is Deep Equilibrium Model (DEQ) [3], [4], which observed that, in the weight-tying case (share the weights in each layer), neural network's forward propagation is equivalent to the procedure of power iteration, and its output approaches the fixed point of a single layer. Thus the equilibrium model's output is created by finding a solution to the fixed point equation. Although having a simpler formulation, equilibrium model still lacks interpretability in some sense. Because the underlying meaning of the fixed point equation is unclear, therefore we still cannot analyze its properties and understand the intrinsic mechanisms.

It is interesting to provide interpretability for neural networks from the optimization perspective. In fact, there are already some clues showing that DNNs' components are strongly related to some underlying optimization problems. [5] shows that DNN's activation function is indeed a proximal operator if it is non-decreasing. [6] investigates the potential energy function of the self-attention operators. [7] reformulates the single layer of feed-forward NNs as an argument minimum operator, therefore performing forward propagation becomes solving several constrained convex problems. On the other hand, there is limited work to show the optimization connection for equilibrium models. Previous work [8], [9] proved that equilibrium model with the form $\mathbf{z}^* = \sigma(\mathbf{W}\mathbf{z}^* + \mathbf{U}\mathbf{x} + \mathbf{b})$ is equivalent to an operator splitting problem, but the operator splitting formulation does not have good physical interpretation and it can only be associated with an optimization problem under restricted conditions (e.g., ReLU activation and p.s.d. weight [9]).

Suppose that the equilibrium model can connect with an underlying optimization problem and the explicit objective exists. We may further understand the mechanism of the equilibrium model and the reasons for its empirical success. For example, we can introduce the customized property into the equilibrium model architectures by adding some regularization terms to the underlying optimization objective, which may help to inspire more equilibrium model architectures. Moreover, when finding the model's output is equivalent to minimizing a convex objective, we can utilize any optimization algorithm, especially the accelerated ones, to obtain the equilibrium point rather than the fixed-point

* *Equal Contribution*

- *X. Xie, Q. Wang, Z. Ling, and Z. Lin are with the Key Laboratory of Machine Perception (MOE), School of Artificial Intelligence, Peking University, Beijing 100871, China.*
- *X. Li. is with Swiss Federal Institute of Technology*
- *G. Liu is with School of Automation, Southeast University, Nanjing 210018, China. Email: gcliu1982@gmail.com.*

Corresponding Author: Zhouchen Lin. E-mail: zlin@pku.edu.cn. The patent on the essential ideas of this paper has been filed.

iteration and the root find methods, which are heavily used in the previous equilibrium models.

To understand the equilibrium models from the optimization perspective, this paper investigates a new equilibrium model— Optimization Induced Equilibrium Networks (OptEq), which can be well interpreted from the optimization perspective: solving OptEq's fixed point equation is equivalent to getting the minimizer of an underlying optimization problem. We first show that OptEq's structure can be naturally extracted from any feed-forward NN in Lemma 1. Specially, we reformulate the general feed-forward NN and decompose it into the composition of a new class of unit layers while keeping the output unchanged. We then reveal that OptEq's layer is the proximal operator of an underlying implicit convex function in Theorem 1. Our main contribution comes from two aspects: (1) we discover the intrinsic optimization-induced structure hidden in general feed-forward NNs; (2) the proposed new equilibrium model can be well understood from the optimization perspective. *In principle, we can customize the equilibrium model* by modifying the underlying optimization problem. For example, OptEq naturally produces the commonly used skip connection architecture by replacing the convex objective with its Moreau envelope, which keeps the equilibrium point unchanged. To strengthen the representation ability, we propose a deep version of OptEq, which includes the general feed-forward DNN as a special case, whose underlying convex objective is the sum of single ones of its contained layers. Finally, we utilize OptEq's optimization induced property to improve the equilibrium model further. We offer two methods to introduce any customized prior information to the equilibrium of OptEq. The first way is to modify the underlying optimization problem directly, which leads to the changes of OptEq's architecture. Another way is to use a modified SAM [10] iteration for selecting the fixed point with the minimal regularization values when the fixed point set is non-singleton. In summary, our contributions include:

- By decomposing the general DNN, we discover the intrinsic optimization-induced layer hidden in general feed-forward NNs, which is the proximal operator of a convex function. Then we extract the layer to make it an equilibrium model called OptEq, and further propose its deep version. Without further reparameterization, the equilibrium point of OptEq is a solution to an underlying convex problem. So OptEq's property can be well-studied by investigating the underlying optimization problem.
- We propose two methods to introduce customized properties to the model's equilibrium points. One is inspired by the underlying optimization, and the other is induced by a modified SAM [11] iteration. This is the first time that we can customize equilibrium models in a principled way.
- We conduct experiments on CIFAR-10 and ImageNet for image classification and Cityscapes for semantic segmentation. Deep OptEq significantly outperforms baseline equilibrium models. Moreover, we also provide several feature regularizations that can significantly improve the generalization.

2 Related Work

2.1 Deep Equilibrium Models

Bai et al. [3] pioneered the DEQ, an entirely implicit neural network, by replacing the DNN's forward propagation with a fixed point equation, which can be regarded as a weight-tied DNN with infinite layers. DEQ solves its fixed point equation by the quasi-Newton method and back-propagates itself with implicit differentiation. Many works have investigated DEQs from different perspectives. For example, previous work [8], [9] was devoted to ensuring the stability of DEQ, i.e., the existence and uniqueness of fixed point, and proved that DEQ is equivalent to an operator splitting problem under some reparameterization. There is also some work considering the training strategies [12], [13], and training dynamics [14] for DEQs.

The most relevant work is [9], which associates DEQ with an optimization problem under very restricted conditions (ReLU activation and p.s.d. weight matrix). By contrast, in this work, we try to understand equilibrium models from the optimization perspective, and hence propose our OptEq, which naturally has an underlying optimization objective function under quite mild conditions. Based on this, we suggest principled ways to incorporate prior information into OptEq.

2.2 Bilevel Optimization

Bilevel optimization is a popular research area in optimization and has a wide range of applications in machine learning, like Hyperparameter Optimization [15], Meta-Learning [16], and Neural Architecture Search [17]. Equilibrium models training has many similarities with solving a bilevel optimization problem [18]. For the proposed OptEq, whose output is the minimizer of a convex function exactly, model training is equivalent to solving a bilevel optimization problem. In specific, for training Opteq, we have:

$$\begin{aligned}&\min_{\boldsymbol{\theta}} \operatorname{loss}(\mathbf{z}^*(\mathbf{x},\boldsymbol{\theta}),\mathbf{y}),\\ &s.t.\ \mathbf{z}^* \in \operatorname{Fix}\left(\mathcal{T}(\mathbf{z},\boldsymbol{\theta})\right),\end{aligned} \quad\Leftrightarrow\quad \begin{aligned}&\min_{\boldsymbol{\theta}} \operatorname{loss}(\mathbf{z}^*(\mathbf{x},\boldsymbol{\theta}),\mathbf{y}),\\ &s.t.\ \mathbf{z}^* \in \operatorname*{argmin}_{\mathbf{z}} \Phi(\mathbf{z},\boldsymbol{\theta}).\end{aligned}$$

Note that, for genetal DEQ, this transformation is ill-defined, since the lower-level optimization objective function $\Phi(\cdot)$ does not exist in most cases. Inspired by the well-known SAM algorithm [11] in the bilevel optimization community, we propose the unrolling SAM method to perform OptEq training in Section 5.2.

Note that in Section 5.3, we discuss the difference between the implicit differentiation method and the unrolling algorithm method, which are two important methods for solving bilevel optimization problems [16], [19]. A more detailed discussion of the advantages and disadvantages of these two types of methods can be found in the survey [20]. In practice, the appropriate algorithm should be chosen based on the trade-off between memory and speed.

2.3 Optimization-inspired Neural Networks

In general, Optimization-inspired NNs design DNNs following the optimization algorithms and can be divided into two categories. One to unroll classical optimization or numerical iterative algorithms and introduce learnable parameters, so as to obtain learnable DNNs [21], [22], [23]. The other type replaces one NN's layer with an optimization solver, e.g., [24], [25] consider the solver of quadratic programming (QP) problem as a layer of DNN.

OptEq is relevant but not comparable to the optimization-inspired network. Given the optimization problem, different optimization algorithms may inspire various DNN architectures. However, conversely, given a DNN, it is much harder to identify the underlying optimization problem. Such an "**inverse**" problem is the focus of OptEq. By Lemma 1, we can easily conclude that finding the underlying optimization problem for OptEq is an important step towards understanding general DNNs.

3 The Proposed Optimization Induced Equilibrium Networks

3.1 Preliminaries and Notations

We provide some definitions that are frequently used throughout the paper. A function $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ is proper if the set $\{x : f(x) < +\infty\}$ is non-empty, where $\mathcal{H}$ is the Euclidean space. We write l.s.c for short of lower semi-continuous. The subdifferential, proximal operator and Moreau envelope of a proper convex function f are defined as:

$$\begin{cases} \partial f(\mathbf{x}) := \{\mathbf{g} \in \mathcal{H} : f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \mathbf{y}-\mathbf{x}, \mathbf{g}\rangle, \forall \mathbf{y} \in \mathcal{H}\}, \\ \operatorname{prox}_{\mu \cdot f}(\mathbf{x}) := \left\{ \mathbf{z} \in \mathcal{H} : \mathbf{z} = \operatorname*{argmin}_{\mathbf{u}} \frac{1}{2\mu}\|\mathbf{u}-\mathbf{x}\|^2 + f(\mathbf{u}) \right\}, \\ M_f^{\mu}(\mathbf{x}) := \min_{\mathbf{u}} \frac{1}{2\mu}\|\mathbf{u}-\mathbf{x}\|^2 + f(\mathbf{u}), \end{cases}$$

respectively, where $\langle \cdot, \cdot \rangle$ is the inner product and $\|\cdot\|$ is the induced norm. The conjugate f^* of a proper convex function f is defined as: $f^*(\mathbf{y}) := \sup\{\langle \mathbf{y}, \mathbf{x}\rangle - f(\mathbf{x}) : \mathbf{x} \in \mathcal{H}\}$. For the matrix $\mathbf{W}$, $\|\mathbf{W}\|_2$ is the operator norm. While for the vector ℓ_2-norm, we write $\|\cdot\|$ for simplicity. Given the map $\mathcal{T} : \mathbb{R}^m \to \mathbb{R}^m$, the fixed points set is denoted by $\mathrm{Fix}(\mathcal{T}) = \{\mathbf{z} \in \mathbb{R}^m : \mathcal{T}(\mathbf{z}) = \mathbf{z}\}$, whose cardinality is $|\mathrm{Fix}(\mathcal{T})|$.

DEQ is inspired by the observation on the feed-forward DNN (with an input-skip connection): for $k = 1, \cdots, L-1$,

$$\mathbf{z}_{k+1} = \sigma(\mathbf{W}_k \mathbf{z}_k + \mathbf{U}_k \mathbf{x} + \mathbf{b}_k), \quad \mathbf{y} = \mathbf{W}_{L+1}\mathbf{z}_L, \tag{1}$$

where $\sigma(\cdot)$ is a non-linear activation function, $\mathbf{W}_k \in \mathbb{R}^{n_k \times n_{k-1}}$ and $\mathbf{U}_k \in \mathbb{R}^{n_k \times d}$ are learnable weights and $\mathbf{b}_k \in \mathbb{R}^{n_k}$ is a bias term. A direct way to obtain the equilibrium point of this system is to consider the fixed point equation: $\mathbf{z}^* = \sigma(\mathbf{W}\mathbf{z}^* + \mathbf{U}\mathbf{x} + \mathbf{b})$. And we can utilize any root-finding algorithm to solve this equation. Although DEQ may achieve good performance with a smaller number of parameters than DNNs, its superiority heavily relies on the careful initialization and regularization due to the instability issue of the fixed point problems. Some recent work [8] is devoted to solving the instability issue by using a tricky reparametrization of the weight matrix $\mathbf{W}$. However, this may greatly weaken the expressive power of DEQ, see Prop. 8 in [9]. Most importantly, in the present equation form, we have difficulty in getting further properties of the equilibrium point.

3.2 One Layer OptEq

Here, we consider an alternative form of the system in Eq.(1).

Lemma 1 (Universal Hidden Unit)**.** *Given the parameters $\{(\mathbf{W}_k, \mathbf{U}_k, \mathbf{b}_k)\}_{k=1}^L$ of a general DNN in Eq.(1), there exists a set of weights $\{\overline{\mathbf{W}}_k \in \mathbb{R}^{n_k \times m}\}_{k=0}^L$ with $m \geq \max\{n_k + n_{k+1}, k = 1, \cdots, L-1\}$, such that the system in Eq.(1) can be re-written as the following network: for $k = 1, \cdots, L-1$,*

$$\overline{\mathbf{z}}_{k+1} = \overline{\mathbf{W}}_k^\top \sigma(\overline{\mathbf{W}}_k \overline{\mathbf{z}}_k + \mathbf{U}_k \mathbf{x} + \mathbf{b}_k), \quad \mathbf{y} = \overline{\mathbf{W}}_{L+1}\overline{\mathbf{z}}_L. \tag{2}$$

Notably, *without changing the output* $\mathbf{y}$, any feed-forward DNN has the reformulation in Eq.(2). The formal proof can be found in appendix, we present the main idea here:

$$\begin{aligned} \mathbf{y} =& \mathbf{W}_L \sigma\Big(\mathbf{W}_{L-1}\sigma(\cdots \mathbf{W}_2 \sigma(\mathbf{W}_1 \mathbf{z}_0 + \mathbf{U}_1\mathbf{x} + \mathbf{b}_1)\cdots)\Big) \\ =& \underbrace{\overline{\mathbf{W}}_L \, \overline{\mathbf{W}}_{L-1}^\top}_{\mathbf{W}_L} \sigma\left(\underbrace{\overline{\mathbf{W}}_{L-1} \, \overline{\mathbf{W}}_{L-2}^\top}_{\mathbf{W}_{L-1}} \sigma\left(\cdots \underbrace{\overline{\mathbf{W}}_2 \, \overline{\mathbf{W}}_1^\top}_{\mathbf{W}_2} \cdots \right)\right). \end{aligned}$$

In the sense of weight-tying (i.e., all the layers share the same weights), the DNN's output $\mathbf{y}$ is a linear transformation of $\overline{\mathbf{z}}_L$, where $\overline{\mathbf{z}}_L$ is a good approximation of the following fixed point equation under some mild assumptions:

$$\mathbf{z}^* = \mathbf{W}^\top \sigma(\mathbf{W}\mathbf{z}^* + \mathbf{U}\mathbf{x} + \mathbf{b}). \tag{3}$$

Hence, the feed-forward DNN also inspires an interesting and different equilibrium model Eq.(3). We call it Optimization Induced Equilibrium Networks (OptEq) since it is tightly associated with an underlying optimization problem. As shown in Theorem 1 that follows, the equilibrium point $\mathbf{z}^*$ is a solution of a convex problem that has an explicit formulation. From the perspective of optimization, we can easily solve the existence and the uniqueness problems of the fixed point equation, rather than resorting to the reparameterization trick. Most importantly, by studying the underlying optimization problem, we can investigate the properties of the equilibrium point of OptEq. The following theorem formally shows the relation between OptEq and optimization.

Assumption 1. *The activation function $\sigma : \mathbb{R} \to \mathbb{R}$ is monotone and $\tilde{L}_\sigma$-Lipschitz, i.e.,*

$$0 \leq \frac{\sigma(a) - \sigma(b)}{a - b} \leq \tilde{L}_\sigma, \quad \forall a, b \in \mathbb{R}, \quad a \neq b.$$

Theorem 1. *If Assumption 1 holds, for one NN layer $f : \mathbb{R}^m \to \mathbb{R}^m$ given by:*

$$f(\mathbf{z}) := \frac{1}{\mu}\mathbf{W}^\top \sigma(\mathbf{W}\mathbf{z} + \mathbf{U}\mathbf{x} + \mathbf{b}), \quad \mu \geq \tilde{L}_\sigma \|\mathbf{W}\|_2^2,$$

the solution to the fixed point equation $\mathbf{z} = f(\mathbf{z})$ is the minimizer of the convex function $\varphi(\cdot)$, where,

$$\varphi(\mathbf{z}) = \psi^*(\mathbf{z}) - \frac{1}{2}\|\mathbf{z}\|^2, \ \psi(\mathbf{z}) = \frac{1}{\mu}\mathbf{1}^\top \tilde{\sigma}(\mathbf{W}\mathbf{z} + \mathbf{U}\mathbf{x} + \mathbf{b}),$$

in which $\forall a \in \mathbb{R}$, $\tilde{\sigma}(a) = \int_0^a \sigma(t)\, dt$, applied element-wisely to vectors, and $\mathbf{1}$ is the all one vector. Furthermore, we have $f(\mathbf{z}) = \operatorname{prox}_\varphi(\mathbf{z})$, i.e., the NN layer is a proximal operator.

Theorem 1 shows that OptEq's layer given in Eq.(3) is a proximal operator of an underlying convex function given

TABLE 1: Examples to modify the underlying optimization problem.

Underlying Convex Function	OptEq Layer	Illustration
$\varphi(\mathbf{z})=\psi^*(\mathbf{z})-\frac{1}{2}\\|\mathbf{z}\\|^2$	$\mathbf{W}^\top\sigma(\mathbf{W}\mathbf{z}+\mathbf{U}\mathbf{x}+\mathbf{b})$	original OptEq
$\alpha M_\varphi^{1-\alpha}(\mathbf{z})$	$\alpha\mathbf{W}^\top\sigma(\mathbf{W}\mathbf{z}+\mathbf{U}\mathbf{x}+\mathbf{b})+(1-\alpha)\mathbf{z}$	introduce skip connection
$\psi^*(\mathcal{A}(\mathbf{z}))-\frac{1}{2}\\|\mathcal{A}(\mathbf{z})\\|^2$	$\mathcal{A}^{-1}\big(\mathbf{W}^\top\sigma(\mathbf{W}\mathcal{A}(\mathbf{z})+\mathbf{U}\mathbf{x}+\mathbf{b})\big)$	$\mathcal{A}(\cdot)$ is an invariable affine operator, including translation, rotation and scaling
$\mathbf{1}^\top\tilde{\sigma}^*\big(\mathbf{W}^{-\top}\mathbf{z}\big)\to\frac{1}{2}\mathbf{z}^\top\mathbf{W}^{-1}\mathbf{S}^{-1}\mathbf{W}^{-\top}\mathbf{z}$	$\mathbf{W}^\top\mathbf{S}(\mathbf{W}\mathbf{z}+\mathbf{U}\mathbf{x}+\mathbf{b})$	for ReLU case in Eq.(4), multivariate activation function
$(\varphi+\gamma\mathcal{R}_z)(\mathbf{z})$	$\mathbf{W}^\top\sigma\Big(\mathbf{W}(\mathbf{z}^*-\gamma\frac{\partial\mathcal{R}_z(\mathbf{z}^*)}{\partial\mathbf{z}})+\mathbf{U}\mathbf{x}+\mathbf{b}\Big)$	minimizer of $(\varphi+\gamma\mathcal{R}_z)(\cdot)$, see Theorem 4 for more details

by a conjugate function, and the equilibrium point of OptEq happens to be the minimizer of this function[1]. In the rest part of this paper, for ease of discussion, we focus on the case that $\mu=1$ for OptEq, which may correspond to the assumption $\tilde{L}_\sigma=1$ and $\|\mathbf{W}\|_2\leq 1$.

In some cases, we can write down the closed form of the optimization objection $\varphi(\cdot)$. For example, when the weight matrix $\mathbf{W}$ is invertible, and the activation $\sigma(\cdot)$ is ReLU, i.e., $\sigma(x)=\max\{x,0\},\forall x\in\mathbb{R}$, we have:

$$\varphi(\mathbf{z})=\mathbf{1}^\top\tilde{\sigma}^*\Big(\mathbf{W}^{-\top}\mathbf{z}\Big)-\Big\langle\mathbf{U}\mathbf{x}+\mathbf{b},\mathbf{W}^{-\top}\mathbf{z}\Big\rangle-\frac{1}{2}\|\mathbf{z}\|^2,\quad(4)$$

where $\tilde{\sigma}^*(x)=\begin{cases}\frac{1}{2}x^2, & x>0,\\ \infty, & x\leq 0\end{cases}$, applied element-wise to vectors. In the ReLU activation case, OptEq is equivalent to solving a QP problem. As a convex optimization layer, QP's powerfulness and effectiveness have been verified in [24], [25].

By Theorem 1, we can quickly obtain the well-posedness of OptEq. In general, any $\mathbf{W}$ that makes the underlying objective φ to be strictly convex will ensure the existence and uniqueness of OptEq's equilibrium. For example, when $\|\mathbf{W}\|_2<1$, the operator: $\mathbf{z}\mapsto\mathbf{W}^\top\sigma(\mathbf{W}\mathbf{z}+\mathbf{U}\mathbf{x}+\mathbf{b})$ is contractive, therefore the fixed point equation Eq.(3) has a unique solution, i.e., it exists and is unique. What's more, we show a training strategy that can actually deal with a much more general case — $|\text{Fix}(\mathcal{T})|\geq 1$. Please see Section 5.2 for more details.

OptEq's most attractive aspect is that, by modifying the underlying convex problem, it can inspire many different equilibrium model architectures. For example, we can introduce the commonly used skip connection structure only by replacing $\varphi(\mathbf{z})$ with its Moreau envelope $\alpha M_\varphi^{1-\alpha}(\mathbf{z})$, which does not change the equilibrium point but make OptEq's layer strongly monotone and invertible, and hence can stabilize the iteration. We provide more examples in Table 1. Moreover, we can introduce customized properties of equilibrium point into the model in this way, please see Section. 5.1.

3.3 Deep OptEq

Some work claims that one layer implicit equilibrium is enough [3] and improves the model expressive ability by stacking small DEQs to obtain a wide one-layer DEQ, i.e., considering a fixed point problem in a higher dimension. However, in practice, solving a high-dimensional fixed-point equation is very time-consuming. Therefore, to improve the efficiency of OptEq, we choose to concatenate OptEq and propose deep OptEq, which is a good extension of the one-layer wide one since it improves the model capability without changing the problem scale. Indeed, as we will show in the next section, wide single-layer Opteqs are special cases of deep Opteqs in the asymptotic sense.

In this subsection, we propose a multi-layer version of OptEq, which is also associated to an underlying optimization problem.We consider a multi-layer OptEq, where $\mathbf{x}_0\in\mathbb{R}^{d_x}$ denotes the input, $\mathbf{z}\in\mathbb{R}^m$ denotes the hidden unit, and $\mathbf{y}\in\mathbb{R}^{d_y}$ denotes the output. Namely, deep OptEq follows the implicit equation:

$$\begin{cases}\mathbf{x}=g(\mathbf{x}_0,\mathbf{W}_0),\\ \mathbf{z}=\mathcal{T}(\mathbf{z},\mathbf{x},\boldsymbol{\theta}):=f_L\circ f_{L-1}\cdots\circ f_1(\mathbf{z},\mathbf{x},\boldsymbol{\theta})\ ,\\ \mathbf{y}=\mathbf{W}_{L+1}\mathbf{z},\end{cases}\quad(5)$$

where for all $l\in[1,L]$ and $\alpha\in(0,1]$,

$$f_l(\mathbf{z},\mathbf{x})=\alpha\mathbf{W}_l^\top\sigma(\mathbf{W}_l\mathbf{z}+\mathbf{U}_l\mathbf{x}+\mathbf{b}_l)+(1-\alpha)\mathbf{z}.\quad(6)$$

Here, given the set of learnable parameters $\mathbf{W}_0, g(\cdot,\mathbf{W}_0):\mathbb{R}^{d_x}\to\mathbb{R}^d$ is a continuous function which we usually choose as the feature extractor, e.g., shallow NNs. $\boldsymbol{\theta}=\{(\mathbf{W}_l,\mathbf{U}_l,\mathbf{b}_l)\}_{l=1}^L$ is the set of all learnable parameters for our equilibrium network, where $\mathbf{W}_l\in\mathbb{R}^{n_l\times m}$, $\mathbf{U}\in\mathbb{R}^{n_l\times d}$, $\mathbf{b}_l\in\mathbb{R}^{n_l}$ are the learnable weight matrices and bias term, respectively. Note that $\mathbf{W}_{L+1}\in\mathbb{R}^{d_y\times m}$ is also learnable. $\sigma:\mathbb{R}\to\mathbb{R}$ is the activation function, when the input is multi-dimensional, we apply the function $\sigma(\cdot)$ element-wise. The hidden unit $\mathbf{z}$ is the equilibrium point of the fixed point equation $\mathbf{z}=\mathcal{T}(\mathbf{z},\mathbf{x},\boldsymbol{\theta})$ when $\mathbf{x}$ and $\boldsymbol{\theta}$ are given. Without loss of generality, we assume that the feature extractor satisfies a Lipschitz continuity assumption w.r.t. the learnable weight, i.e., $\|g(\mathbf{x}_0,\mathbf{W}_1)-g(\mathbf{x}_0,\mathbf{W}_2)\|\leq L_g\|\mathbf{W}_1-\mathbf{W}_2\|_2$.

At the first glance, deep OptEq seems very different from the traditional DNNs. Some negative results on DEQ are shown in previous work [9]: with improper weight re-parameterization, DEQ does not contain any feed-forward networks. By contrast, without weight re-parameterization, deep OptEq can include the general feed-forward DNN as its special case.

[1] Besides the forms we show, one may expect a common rule to determine whether a general mapping is a proximal operator or not, which can help to create the new equilibrium models from the optimization perspective. We provide the sufficient and necessary conditions in Lemma 5 (see appendix).

Lemma 2. *The deep OptEqs contain all feed-forward DNNs. More precisely, given a feed-forward DNN in the form:*

$$\mathbf{z}_{k+1} = \sigma(\mathbf{A}_k\mathbf{z}_k + \mathbf{c}_k), \quad k = 1, \cdots, L-1, \quad \mathbf{y} = \mathbf{A}_{L+1}\mathbf{z}_L,$$

where $\mathbf{z}_1 = \mathbf{x}$ is the input. Then there exists $\{(\mathbf{W}_l, \mathbf{U}_l, \mathbf{b}_l)\}_{l=1}^{L}$ such that $\mathbf{y}$ is also the output of the corresponding deep OptEq in Eq.(5).

4 Recover Underlying Optimization

This section provides our main results on the connection between convex optimization problem and our deep OptEq. In the previous section, we have shown that one layer of DNN is a proximal operator under a mild assumption. However, the composition of multiple proximal operators is not a proximal operator in most cases. Fortunately, we can still recover the underlying optimization problem of deep OptEq, and find that its equilibrium point is a zero point of a convex function's subdifferential with an additional permutation constraint. In addition, we can explicitly provide the optimization objectives corresponding to deep OptEq in some cases. Before providing the main results, we first show the connection between our deep OptEq given in Eq.(5) and a multi-block one-layer OptEq.

Lemma 3 (Deep OptEq and Multi-block OptEq)**.** *Let $\mathbf{z}_0^*$ be the hidden unit of deep OptEq. Namely $\mathbf{z}_0^*$ is an equilibrium point of the equation $\mathbf{z} = f_L \circ f_{L-1} \cdots \circ f_1(\mathbf{z}, \mathbf{x}, \boldsymbol{\theta})$, set $\mathbf{z}_1^* := f_1(\mathbf{z}_0^*, \mathbf{x}, \boldsymbol{\theta})$, $\mathbf{z}_2^* := f_2 \circ f_1(\mathbf{z}_0^*, \mathbf{x}, \boldsymbol{\theta}), \cdots,$ $\mathbf{z}_{L-1}^* := f_{L-1} \cdots \circ f_2 \circ f_1(\mathbf{z}_0^*, \mathbf{x}, \boldsymbol{\theta})$, then $\widetilde{\mathbf{z}}^* := [\mathbf{z}_1^*, \cdots, \mathbf{z}_{L-1}^*, \mathbf{z}_0^*]^\top \in \mathbb{R}^{mL}$ is an equilibrium point of the equation:*

$$\widetilde{\mathbf{z}} = \alpha\widetilde{\mathbf{W}}^\top \sigma\Big(\widetilde{\mathbf{W}}\mathbf{P}\widetilde{\mathbf{z}} + \widetilde{\mathbf{U}}\mathbf{x} + \widetilde{\mathbf{b}}\Big) + (1-\alpha)\mathbf{P}\widetilde{\mathbf{z}}, \tag{7}$$

where $\widetilde{\mathbf{W}}$ is block diagonal and $\mathbf{P}$ is a permutation matrix,

$$\widetilde{\mathbf{W}} := \begin{bmatrix} \mathbf{W}_1 & & \\ & \ddots & \\ & & \mathbf{W}_L \end{bmatrix}, \quad \mathbf{P} := \begin{bmatrix} 0 & & & \mathbf{I} \\ \mathbf{I} & 0 & & \\ & \ddots & \ddots & \\ & & \mathbf{I} & 0 \end{bmatrix},$$

$\widetilde{\mathbf{U}} := [\mathbf{U}_1, \cdots, \mathbf{U}_L]^\top$ and $\widetilde{\mathbf{b}} := [\mathbf{b}_1, \cdots, \mathbf{b}_L]^\top$ are the concatenated matrix and vector, respectively (see Eq.(15) in appendix for more details).

By Eq.(7), and the necessary and sufficient condition for one DNN layer to be a proximal-like operator (Lemma 6 in appendix), we can further reveal the connection between deep OptEq and optimization under a mild assumption.

Assumption 2. *Assumption 1 with $\tilde{L}_\sigma = 1$ and $\|\mathbf{W}_i\|_2 \leq 1, \forall i \in [1, L]$ holds.*

Note that we make this assumption just for the ease of discussion. The assumption is actually unnecessary since we can introduce an additional constant to re-scale the whole operator as we did in Theorem 1.

Theorem 2 (Recovering Optimization Problem from Deep OptEq)**.** *If Assumption 2 holds, then any equilibrium point $\widetilde{\mathbf{z}}^*$ of Eq.(7) satisfies:*

$$0 \in \partial\Phi(\widetilde{\mathbf{z}}^*) + (\mathbf{I} - \mathbf{P})\widetilde{\mathbf{z}}^*, \tag{8}$$

where $\mathbf{I}$ is the identity matrix and $\Phi(\widetilde{\mathbf{z}}^*)$ is given by a sequence Moreau envelopes of convex functions $\{\varphi_i\}_{i=1}^L$ such that $\mathrm{prox}_{\varphi_i}(\mathbf{z}) = \mathbf{W}_i^\top \sigma(\mathbf{W}_i\mathbf{z} + \mathbf{U}_i\mathbf{x} + \mathbf{b}_i)$, namely:

$$\Phi(\widetilde{\mathbf{z}}) = \sum_{i=1}^{L} \alpha M_{\varphi_i}^{1-\alpha}(\mathbf{z}_i),$$

where $\mathbf{z}_i$ is the i-th block of $\widetilde{\mathbf{z}}^*$ and $M_{\varphi_i}^{1-\alpha}(\mathbf{z})$ is the φ_i's Moreau envelope.

When the block size is 1, i.e., $L = 1$, we can immediately obtain that $0 \in \partial\Phi(\widetilde{\mathbf{z}}^*)$, namely, the equilibrium point is a solution of a convex optimization problem. So the results provided in Theorem 1 is a special case here. Note that one block does not mean that $\mathbf{z}$ is one-dimensional. Moreover, for two blocks, the deep OptEq is also an optimization solver.

Corollary 1. *If the block size $L = 2$ and Assumption 2 holds, then the equilibrium point $\widetilde{\mathbf{z}}^* = [\mathbf{z}_1^*, \mathbf{z}_0^*]^\top$ of Eq.(7) is also a solution to a convex problem:*

$$\min_{\mathbf{z}_1, \mathbf{z}_0} \left\{ \alpha M_{\varphi_1}^{1-\alpha}(\mathbf{z}_1) + \alpha M_{\varphi_2}^{1-\alpha}(\mathbf{z}_0) + \frac{1}{2}\|\mathbf{z}_1 - \mathbf{z}_0\|^2 \right\}.$$

For general $L > 2$, we can also write down the monotone inclusion equation Eq.(8)'s underlying optimization problem when $\alpha \to 0$. Interestingly, this result implies the equivalence between the composited deep models and the wide shallow ones in the asymptotic sense.

Theorem 3 (Connection between Wide and Deep OptEq)**.** *If Assumption 2 holds, and there is at least one $\|\mathbf{W}_i\|_2 < 1$. Assume $\widetilde{\mathbf{z}}^*(\alpha) := [\mathbf{z}_1^*(\alpha), \cdots, \mathbf{z}_{L-1}^*(\alpha), \mathbf{z}_0^*(\alpha)]^\top \in \mathbb{R}^{mL}$ is the equilibrium point of Eq.(7). When $\alpha \to 0$, all $\mathbf{z}_l^*(\alpha)$s tend to be equal, and the limiting point is $\mathbf{y}$, the last entry of the minimizer $(\mathbf{x}_1, \cdots, \mathbf{x}_L, \mathbf{y})$ of the following optimization problem:*

$$\min_{\mathbf{x}_1, \cdots, \mathbf{x}_L, \mathbf{y}} \left\{ \sum_{l=1}^{L} \left(\varphi_l(\mathbf{x}_l) + \frac{1}{2}\|\mathbf{x}_l - \mathbf{y}\|^2 \right) \right\},$$

where $\varphi_l(\cdot)$ is the same as that in Theorem 2.

Theorem 3 implies that, when $\alpha \to 0$, the equilibrium point of the deep OptEq is the same as the solution of $L\mathbf{z} = \sum_{i=1}^L \mathbf{W}_i^\top \sigma(\mathbf{W}_i\mathbf{z} + \mathbf{U}_i\mathbf{x} + \mathbf{b}_i)$, which is a wide one-layer OptEq with multiple blocks. Given the output dimension and the same amount of learnable parameters, the wide multi-block OptEq is actually a special case of deep OptEq in the asymptotic sense. Hence, deep OptEq is more expressive than wide one-layer OptEq.

In general, we can still loosely treat the equilibrium point as a minimizer of an implicit optimization problem, since the only difference between the monotone inclusion equation $0 \in \partial\Phi(\widetilde{\mathbf{z}}^*)$ and Eq.(8) is an additional constraint dominated by the operator $(\mathbf{I} - \mathbf{P})(\cdot)$, which aims to reduce the divergence between the multi-blocks. For example, when $L = 2$, the results in Corollary 1 show that we need to simultaneously consider the sum of convex objective and the distance $\|\mathbf{z}_1 - \mathbf{z}_0\|$.

5 Introducing Customized Properties

By employing the underlying optimization problem, we can investigate the potential property of the equilibrium points. A more advanced way to use the connection between (deep)

TABLE 2: Some choices for $\mathcal{R}_z$ and the corresponding $\mathcal{T}_{\mathcal{R}_z}$.

$\mathcal{R}_z$	Operator	Form	Prior Information
$\frac{\lVert\mathbf{z}\rVert^2}{2}$	$\text{prox}_{\gamma R_z}$	$\frac{\mathbf{z}}{1+\gamma}$	re-scale, feature decay
$\lVert\mathbf{z}\rVert_1$	$\text{prox}_{\gamma R_z}$	$(\lvert\mathbf{z}\rvert-\gamma)_+ \odot \text{sgn}(\mathbf{z})$	shrinkage operator, surrogate of sparsity
$\frac{2}{\lVert\mathbf{z}\rVert^2+2\epsilon}$	$\mathcal{I}-\gamma\frac{\partial\mathcal{R}_z}{\partial\mathbf{z}}$	$\left(1-\frac{4\gamma}{(\lVert\mathbf{z}\rVert^2+2\epsilon)^2}\right)\mathbf{z}$	feature incay, feature expansion
$\frac{1}{2}\sum_{i\neq j}\left(\frac{\mathbf{z}_i^\top\mathbf{z}_j}{\lVert\mathbf{z}_i\rVert\lVert\mathbf{z}_j\rVert}\right)^2$	$\mathcal{I}-\gamma\frac{\partial\mathcal{R}_z}{\partial\mathbf{z}_i}$	$\mathbf{z}_i-(\mathcal{I}+\text{Proj}\,(\mathbf{z}_i))\circ\sum\text{Proj}\,(\mathbf{z}_{j\neq i})(\frac{\mathbf{z}_i}{\lVert\mathbf{z}_i\rVert^2})$	feature decorrelation, improve independence, please see Eq.(13), where $\text{Proj}\,(\mathbf{z}_j):=\frac{\mathbf{z}_j\mathbf{z}_j^\top}{\lVert\mathbf{z}_j\rVert^2}$

OptEq and optimization is to introduce some customized properties to equilibrium points, i.e., the feature learned by OptEq. Note that none of the previous DEQs take into account the regularization of features, which has been proved to be effective both theoretically [26], [27] and empirically [28].

5.1 Underlying Optimization Inspired Feature Regularization

As we mentioned in Section 3.2, if we replace the underlying optimization objective with its Moreau envelope, OptEq will naturally have a skip-connection structure, which has been adopted in the construction of deep OptEq. Following this idea, when we modify the underlying optimization problem of deep OptEq, it should inspire more network architectures.

An exciting application of Theorem 2 is introducing customized properties by modifying $\Phi(\cdot)$: appending one layer after deep OptEq is equivalent to adding one term to the objective $\Phi(\cdot)$. Specifically, we have the following theorem.

Theorem 4. *With the same setting as in Theorem 1. The fixed point of the equation $\mathbf{z}^* = f\circ(\mathcal{I}-\gamma\frac{\partial\mathcal{R}_z}{\partial\mathbf{z}})(\mathbf{z}^*)$, namely:*

$$\mathbf{z}^* = \mathbf{W}^\top\sigma\bigg(\mathbf{W}(\mathbf{z}^*-\gamma\frac{\partial\mathcal{R}_z(\mathbf{z}^*)}{\partial\mathbf{z}})+\mathbf{U}\mathbf{x}+\mathbf{b}\bigg),$$

is the minimizer of the convex function $(\varphi+\gamma\mathcal{R}_z)(\cdot)$.

Hence, if We modify $\Phi(\widetilde{\mathbf{z}})$ to $\Phi(\widetilde{\mathbf{z}})+\mathcal{R}_z(\mathbf{z}_L)$, then deep OptEq becomes:

$$\mathbf{z} = \mathcal{T}(\mathcal{T}_{\mathcal{R}_z}(\mathbf{z}),\mathbf{x},\boldsymbol{\theta}),$$

where $\mathcal{T}_{\mathcal{R}_z}=\text{prox}_{\gamma R_z}$ or $\mathcal{T}_{\mathcal{R}_z}=\mathcal{I}-\gamma\frac{\partial\mathcal{R}_z}{\partial\mathbf{z}}$ when the proximal is hard to calculate. For the choice of $\mathcal{R}_z$, we give several examples in Table.

In general, $\mathcal{R}_z(\cdot)$ can be any convex function that contains the prior information of the feature. In summary, we introduce feature regularization by modifying the underlying optimization problem, which leads to a change of network structure. Once again, studying the equilibrium models from the perspective of optimization shows great advantages.

5.2 SAM Iteration Induced Feature Regularization

In this subsection, we provide another strategy for feature regularization. Note that most previous DEQs are devoted to ensuring a singleton fixed point set, relying on the tricky weight re-parameterization. Considering the general case — $|\text{Fix}(\mathcal{T})|\geq 1$, we can choose the equilibrium with desired property by solving the following constrained optimization problem:

$$\mathbf{z}^*(\mathbf{x},\boldsymbol{\theta}) := \underset{\mathbf{z}\in\text{Fix}\,(\mathcal{T}(\cdot,\mathbf{x},\boldsymbol{\theta}))}{\text{argmin}}\ \mathcal{R}_z(\mathbf{z}), \tag{9}$$

where $\mathcal{R}_z(\cdot)$ is the feature regularization that contains the prior information of the feature. Given the training data $(\mathbf{x}_0,\mathbf{y}_0)\in\mathbb{R}^{d_x}\times\mathbb{R}^{d_y}$, the whole training procedure becomes[2]:

$$\min_{\widetilde{\boldsymbol{\theta}}}\ell(\mathbf{W}_{L+1}\cdot\mathbf{z}^*(\mathbf{x},\ \boldsymbol{\theta}),\mathbf{y}_0)+\mathcal{R}_w(\widetilde{\boldsymbol{\theta}}), \tag{10}$$

where $\widetilde{\boldsymbol{\theta}}:=\{\mathbf{W}_0,\boldsymbol{\theta},\mathbf{W}_{L+1}\}$, $\mathbf{x}$ is given by Eq.(5), $\mathcal{R}_w(\cdot)$ is the regularizer on the parameters , e.g., weight decay, and $\ell:\mathbb{R}^{d_y}\times\mathbb{R}^{d_y}\to\mathbb{R}^+$ is the loss function.

We adopt the Sequential Averaging Method (SAM) [10] to solve the problem Eq.(9). Starting from any $\mathbf{z}_0\in\mathbb{R}^m$, we consider the following sequence $\{\mathbf{z}^k\}_{k\in\mathbb{N}}$:

$$\mathbf{z}^k = \beta_k\mathcal{S}_{\lambda_k}(\mathbf{z}^{k-1})+(1-\beta_k)\mathcal{T}(\mathbf{z}^{k-1},\mathbf{x},\boldsymbol{\theta}), \tag{11}$$

where $\{\beta_k\}_{k\in\mathbb{N}}$ and $\{\lambda_k\}_{k\in\mathbb{N}}$ are sequences of real numbers in $(0,1]$, $\mathcal{S}_\lambda(\mathbf{z})=(1-\gamma\lambda)\mathbf{z}-\gamma\frac{\partial\mathcal{R}_z(\mathbf{z})}{\partial\mathbf{z}}$, where $\gamma\in[0,1]$ is a hyper-parameter. Then we choose the K-th iteration $\mathbf{z}^K(\mathbf{x},\boldsymbol{\theta})$ as an approximate of $\mathbf{z}^*(\mathbf{x},\boldsymbol{\theta})$ and put it in the final loss term:

$$\min_{\widetilde{\boldsymbol{\theta}}}\ell\Big(\mathbf{W}_{L+1}\cdot\mathbf{z}^K(\mathbf{x},\ \boldsymbol{\theta}),\mathbf{y}_0\Big)+\mathcal{R}_w(\widetilde{\boldsymbol{\theta}}). \tag{12}$$

The unrolling term $\mathbf{z}^K(\mathbf{x},\ \boldsymbol{\theta})$ aggregates information from both $\mathcal{R}_z(\mathbf{z})$ and $\mathcal{T}$, making the prior information of feature being an inductive bias during training. And our model can be easily trained by any first-order optimization algorithms, e.g., GD, SGD, Adam, etc.

Remark 1. *SAM needs $\mathcal{R}_z(\cdot)$ to be strongly convex, here we use a modified version of SAM which only assumes convexity. Given well-chosen $\{\beta_k\}_{k\in\mathbb{N}}$ and $\{\lambda_k\}_{k\in\mathbb{N}}$, we can prove that the sequence generated by Eq.(11) converges to the point $\mathbf{z}^*(\mathbf{x},\boldsymbol{\theta})$. Furthermore, we prove that the whole training dynamic, using the unrolling SAM strategy with backpropagation (BP), converges with a linear convergence rate.*

Remark 2. *If we use the method in Section 5.1 to introduce the prior information, there is no need to use SAM iteration again. Therefore, we can let beta $\beta_k=0$, $\mathcal{S}_{\lambda_k}(\cdot)=0$, and use the unrolling fixed point iteration strategy during training. Note that we can also use the implicit function theorem (IFT) based training way.*

[2] For the sake of clarity, we utilize one training pair for discussion. In general, all the discussed results hold when we replace the single data point with the whole data set.

Previous equilibrium models utilize IFT in training to avoid the storage consumption of forward-propagation. The cost for that is it needs to solve two large-scale linear equations (or perform the matrix inversion directly) during training. Deep OptEqs can be trained both in the unrolling based and IFT based ways. In the sense of BP, the two training ways have different merits and limitations. We provide comparative experiments in Section 7 and detailed discussion in the following subsection.

5.3 Discussion about the Training Methods

The training method is summarised in Algorithm 1.

Algorithm 1: Training Algorithm for (Deep) OptEq

```
Input: training data $(\mathbf{x}_{0,i}, \mathbf{y}_i)_{i=1}^{n}$, unrolling number
       $K$, iteration number $T$, feature extractor $g(\cdot)$
for $t = 0$ to $T$ do
    forward: $\mathbf{x} = g(\mathbf{x}_0, \mathbf{W}_0)$, set $\mathbf{z}_0 = \mathbf{x}$
        if IFT-based training then
            solve the fixed point equation
             $\mathbf{z}^* = \mathcal{T}(\mathcal{T}_{\mathcal{R}_z}(\mathbf{z}^*), \mathbf{x})$;
        else if unrolling with SAM then
            for $k = 1$ to $K$ do
                $\mathbf{z}^k = \beta_k \mathcal{S}_{\lambda_k}(\mathbf{z}^{k-1}) + (1-\beta_k)\mathcal{T}(\mathbf{z}^{k-1}, \mathbf{x})$;
                 /* SAM by Eq.(11) */
            end
            $\mathbf{z}^* = \mathbf{z}^K$;
        else                                /* raw unrolling */
            for $k = 1$ to $K$ do
                $\mathbf{z}^k = \mathcal{T}(\mathcal{T}_{\mathcal{R}_z}(\mathbf{z}^{k-1}), \mathbf{x})$;
            end
            $\mathbf{z}^* = \mathbf{z}^K$;
        end
    backward:
        evaluate the loss $L := \ell(\mathbf{W}\mathbf{z}^*, \mathbf{y}) + \mathcal{R}_w(\widetilde{\boldsymbol{\theta}})$;
        if unrolling-based training then
            get $\partial L / \partial \widetilde{\boldsymbol{\theta}}$ by automatic differentiation;
        else if IFT-based training then
            get $\partial L / \partial \widetilde{\boldsymbol{\theta}}$ by implicit differentiation;
        update the learnable parameters $\widetilde{\boldsymbol{\theta}}$;
end
```

The IFT based implicit way utilizes the limited memory to train the model and is insensitive to the equilibrium point finding algorithms. However, it consumes much computation budget to solve the equation during the inference and BP. On the other hand, the way that unrolls the fixed point finding method may induce implicit bias [29], [30] and consumes much memory during training, but it is faster to infer and train. Note that implicit bias is a double-edged sword; The proposed SAM method can aggregate the information from the prior regularization and the fixed point equation. Hence, the implicit bias becomes a controllable inductive bias.

The tradeoff between memory and computing efficiency for the implicit and unrolling training methods is quite common in the other learning community, such as meta-learning [31] and hyper-parameter optimization [19]. Similarly, for DEQ, the two training ways are neither good nor bad. We should choose them in proper circumstances.

6 Convergence Analysis

This section offers the convergence results: (i) the sequence generated by Eq.(11) converges to some point $\mathbf{z}^* \in \mathrm{Fix}(\mathcal{T})$ such that $\mathcal{R}_z(\mathbf{z}^*) \leq \mathcal{R}_z(\mathbf{z})$, $\forall \mathbf{z} \in \mathrm{Fix}(\mathcal{T})$; (ii) gradient descent can find a global minimum for the model in Eq.(12).

6.1 Approximation of Equilibrium Point

Note that we approximate the points $\mathbf{z}^*(\mathbf{x}, \boldsymbol{\theta})$ by the iterative steps in Eq.(11). In fact, we take the iterative step by extending an existing algorithm, SAM, which was developed in [10] for solving a certain class of fixed-point problems, and then was applied to the bilevel optimization problems [11]. However, the existing SAM method can only deal with strongly convex $\mathcal{R}_z(\mathbf{z})$. Our method is the first SAM type algorithm that can solve the general convex problem restricted to a nonexpansive operator's fixed point set. The following theorem provides the formal statement and the required conditions. Since during the forward-propagation, $(\boldsymbol{\theta}, \mathbf{x})$ is fixed, for the sake of convenience, we simplify $\mathcal{T}(\mathbf{z}, \mathbf{x}, \boldsymbol{\theta})$ as $\mathcal{T}(\mathbf{z})$.

Theorem 5 (Convergence of Modified SAM Iterates)**.** *Suppose that $\nabla\mathcal{R}_z(\mathbf{z})$ is L_z-Lipschitz, and that for any $\beta \in [0, \frac{1}{2}], \lambda \in [0, \frac{L_z}{2}]$, the fixed point set of equation: $\mathbf{z} = \beta(\mathbf{z} - \gamma(\nabla\mathcal{R}_z(\mathbf{z}) + \lambda\mathbf{z})) + (1-\beta)\mathcal{T}(\mathbf{z})$ is uniformly bounded by B_1^* (in norm $\|\cdot\|$) w.r.t. β and λ. Suppose that convex function $\mathcal{R}_z(\mathbf{z})$ has a unique minimizer $\bar{\mathbf{z}}$ on $\mathrm{Fix}(\mathcal{T})$. Let $\beta_k = \frac{\eta}{k^\rho}, \lambda_k = \frac{\eta}{k^c}, \gamma = \frac{1}{2L_z}$, where $\rho, c > 0$, $\rho + 2c < 1$ and $\eta = \min\left\{\sqrt{2L_z}, \frac{L_z}{2}, \frac{1}{2}\right\}$, then the sequence $\{\mathbf{z}^k\}_{k\in\mathbb{N}^+}$ generated by Eq.(11) converge to $\bar{\mathbf{z}}$.*

The formal assumptions of Theorem 5 seem complicated, however, they can be easily fulfilled when $\mathcal{T}(\mathbf{z})$ is contractive. A sufficient condition that makes $\mathcal{T}(\mathbf{z})$ contractive is to let one $\|\mathbf{W}_i\|_2 \leq \zeta < 1$. More specifically, if some $\|\mathbf{W}_i\|_2 \leq \zeta < 1$, then $\mathbf{z} \mapsto \beta(\mathbf{z} - \gamma(\nabla\mathcal{R}_z(\mathbf{z}) + \lambda\mathbf{z})) + (1-\beta)\mathcal{T}(\mathbf{z})$ is contractive and has a unique fixed point, which depends continuously on β and λ [32], and thereby have a uniform bound.

6.2 Global Convergence of Implicit Model

Most previous works on DEQs lack the convergence guarantees for their training. However, analyzing the learnable parameters' dynamics is crucial since it may weaken many model constraints and greatly broaden the function class that the implicit model can represent. For example, the one-layer DEQ, given in [8], maintains the positive definiteness of $(\mathbf{I} - \mathbf{W})$ for all weight $\mathbf{W}$ in $\mathbb{R}^{m\times m}$ through a complicated parameterization technique. However, after analysis, we find that the learnable weight will stay in a small compact set during training, thus, we may only need the positive definiteness within a local region instead of global space for the DEQ [8].

Theorem 6 (Global Convergence (informal))**.** *Suppose that the initialized weight $\mathbf{W}_l$'s singular values are lower bounded away from zero for all $l \in [1, L+1]$, and the fixed point set $\mathrm{Fix}(\mathcal{T}(\cdot, \mathbf{X}, \boldsymbol{\theta}))$ is non-empty and uniformly bounded for any $\widetilde{\boldsymbol{\theta}}$ in a pre-defined compact set. Assume that the activation function is Lipschitz smooth, strongly monotone and 1-Lipschitz. Define constants Q_0, Q_1 and Q_3, which depend on the bounds for* *initialization parameters, initial loss value, and the datasize. Let the learning rate be $\eta < \min\{\frac{1}{Q_0}, \frac{1}{Q_1}\}$. If for all $l \in [1, L]$, we have $n_l = \Omega(poly(N))$, then the training loss vanishes at a linear rate as:*

$$\ell(\widetilde{\boldsymbol{\theta}}^t) \leq \ell(\widetilde{\boldsymbol{\theta}}^0)(1 - \eta Q_0)^t,$$

where t is the number of iteration. Furthermore, the network parameters also converge to a global minimizer $\widetilde{\boldsymbol{\theta}}^*$ at a linear speed:

$$\|\widetilde{\boldsymbol{\theta}}^t - \widetilde{\boldsymbol{\theta}}^*\| \leq Q_3(1 - \eta Q_0)^{t/2}.$$

Theorem 6 shows that GD converges to a global optimum for any initialization satisfying the boundedness assumption. In general, the lower bounded assumption on singular values is easy to fulfill. With high probability, the weight matrix's singular values are lower bounded away from zero when it is a rectangle and has independent, sub-Gaussian rows or has independent Gaussian entries, see Thm.4.6.1 and Ex.7.3.4 in [33].

6.2.1 About Boundness and Well-Posedness

Similar to the remark after Theorem 5, the existence and boundedness assumption on the set $\mathrm{Fix}(\mathcal{T})$ is mild. Suppose that there exists $l \in [1, L]$, such that $\mathbf{W}_l \leq \zeta < 1$, then the fixed point of $\mathcal{T}$ exists and is unique, and is continuous w.r.t. the parameters $\widetilde{\boldsymbol{\theta}}$. Moreover, via over-parameterization, a proper Gaussian initialization followed by gradient descent produces a sequence of iterates that stay inside a small perturbation region centered at the initial weights. In such a small perturbation region, the largest singular value of each weight $\mathbf{W}_l$ does not change a lot, namely, $\mathbf{W}_l \leq 1$ holds during training. Therefore, throughout the training process, the fixed points are uniformly upper bounded, and the fixed point of $\mathcal{T}$ exists and is unique.

7 Experiments

In this section, we investigate the empirical performance of deep OptEq from three aspects. First, on the image classification problem, we evaluate the performance of deep OptEqs along with our feature regularization strategies. The results trained with different αs are also reported. Second, we compare deep OptEqs with previous implicit models and traditional DNNs. Finally, we compare our unrolling-based method with the IFT-based method and investigate the influence of unrolling iteration number K. Furthermore, we present the results on Cityscapes for semantic segmentation.

Training Strategy of Deep OptEqs. In order to compare the effect of feature regularization on performance in detail, we compare three unrolling training ways based on Eq.(12): (1) $\mathcal{R}_z^\dagger$: strategy in Section 5.2, using the proposed SAM given in Eq.(11); (2) $\mathcal{R}_z^*$: strategy in Section 5.1, and set $\beta_k = 0$ in Eq.(11); and (3) no Reg: without feature regularization. Here we set the iterative number $K = 20$ for the experiments on CIFAR-10 and let $K = 5$ for the experiments on ImageNet. We discuss the influence of different K in Section 7.4.

7.1 Effects of Different Regularizers

We construct the deep OptEqs with 5 convolutional layers, using five 3×3 convolution kernels with the numbers of channels being $16, 32, 64, 128, 128$. During backward propagation, we utilize the commonly used first order optimization algorithm — SGD. We set the learning rate as 0.1 at the beginning and halve it after every 30 epochs. And the total training epoch is 200.

TABLE 3: (a) The testing accuracy (Acc.) of deep OptEq with different settings. $\mathcal{R}_z^*$ and $\mathcal{R}_z^\dagger$ represent the regularization given in Sec.5.1 and Sec.5.2, respectively. We set different λ for $\mathcal{R}_z^*$ and the mean values are taken on the whole feature tensor. The total number of parameters is 199k.

α	0.01	0.1	0.4	0.8	1.0
Acc-(no Reg)	58.4%	56.8%	86.9%	**87.4**%	87.2%
Acc-($\mathcal{R}_z^\dagger$)	72.7%	61.5%	86.5%	87.0%	**87.7%**
Acc-($\mathcal{R}_z^*$)	72.6%	60.0%	**88.0**%	87.6%	87.5%
Acc-DNN.			82.7%		
$\mathcal{R}_z^* = \lambda\|\cdot\|_1$	0.01		0.15		0.5
mean $\|\cdot\|_1$	> 5		0.81		**0.34**
Acc.	86.4%		**87.7**%		86.9%
$\mathcal{R}_z^* = \lambda\|\cdot\|^2$	0.01		1.0		10.0
mean $\|\cdot\|^2$	> 10		1.56		**0.82**
Acc.	87.3%		**87.6**%		86.7%

TABLE 4: Comparisons with previous implicit models.

Methods	Reg. or Settings	# params	Acc.
Deep OptEqs	Decorrelation ($\mathcal{R}^*\mathbf{z}$)	1.4M	91.0%
	Decorrelation ($\mathcal{R}^\dagger\mathbf{z}$)	162k	86.0%
	HSIC ($\mathcal{R}^*\mathbf{z}$)	162k	87.4%
	No Reg	162k	85.7%
ODEs	Neural ODE	172K	53.7%
	Aug. Neural ODE	172k	60.6%
MONs	Single conv	172K	74.1%
	Single conv (large)	854K	82.5%

In this experiment, we compare the performance of two ways to introduce the feature regularization on the CIFAR-10 dataset. We adopt the feature decorrelation as the $\mathcal{R}_z$ here. We also use two norm regularizations to show whether there is a corresponding effect on the learned feature of deep OptEq. Moreover, we show how the hyperparameter α affects the model performance.

The results are shown in Table 3. With the same size of parameters, deep OptEqs beats the general DNN (given in Eq.(1)) easily. It turns out that there is no linear relationship between the performance and the hyperparameter α. The hyperparameter α serves as a trade-off between the effect of fixed point equation and the regularization induced by operator $\mathcal{S}$. In our setting, with a small initialization for $\{\mathbf{W}_l\}$s, all weights will stay in a small compact set during training (see proof of Theorem 6). Therefore when α approaches 1, deep OptEq is an intense contraction (i.e., with a small contractive coefficient), and the SAM iterations will quickly converge to the fixed point, in which case regularization induced operator $\mathcal{S}$ has a limited impact. When α approaches 0, deep OptEq is to be more like the identity operator, so $\mathcal{S}$ dominates the whole iterations.

The optimization inspired implicit regularization ($\mathcal{R}_z^*$) is also an efficient feature regularization method since it modifies deep OptEq structure directly. Here we present two $\mathcal{R}_z^*(\cdot)$ candidates: $\lambda\|\cdot\|_1$ and $\lambda\|\cdot\|^2$. The outputs of the feature show decreases in the corresponding norm, and a suitable regularization coefficient can lead to a better performance.

7.2 Performance of Different Feature Regularizers

On the dataset CIFAR-10, the experiment in this subsection detailedly shows the effect of different settings on regularizer $\mathcal{R}_z^\dagger$ (for Section 5.2), regularizer $\mathcal{R}_z^*$ (for Section 5.1), and α. Note that in this paper, we mainly focus on feature regularizers. However, another line of work that considers some special (weight) regularization techniques for DEQ has also attracted some attention. For example, [34] suggests a regularization method to stabilize DEQ training by explicitly regularizing the Jacobian of the fixed-point iteration equations. [35] shows that state-dependent inexact gradient brings additional training stability regarding the Jacobian spectral radius, which can be understood as a kind of implicit Jacobian regularization. The reader can refer to [34], [35] and the reference therein for more details. We first introduce a regularizer—Hilbert-Schmidt Independence Criterion (HSIC), which is a feature disentanglement method.

7.2.1 *HSIC*

HSIC is a statistical method to test independence. Compared with the decorrelation method we will present in the following, HSIC can better capture the nonlinear dependency between random variables. We apply HSIC to the feature space. Many works [36], [37] show that when the features learned by the network are uncorrelated, the model usually obtains a good generalization performance. For any pair of random variables $\mathbf{X} = (\mathbf{x}_1, \cdots, \mathbf{x}_B), \mathbf{Y} = (\mathbf{y}_1, \cdots, \mathbf{y}_B)$, where B is the batch size, we utilize the biased finite-sample estimator of HSIC [38]:

$$\mathrm{HSIC}(\mathbf{X}, \mathbf{Y}) := (B-1)^{-2} \operatorname{tr}(\mathbf{K}_X \mathbf{H} \mathbf{K}_Y \mathbf{H}),$$

where $\mathbf{K}_X$ and $\mathbf{K}_Y$ are the kernel matrices w.r.t. Gaussian RBF kernel of $\mathbf{X}$ and $\mathbf{Y}$, and $\mathbf{H}$ is the centering matrix $\mathbf{H} = \mathbf{I} - B^{-1}\mathbf{1}_B \mathbf{1}_B \in \mathbf{R}^{B \times B}$. Following [28], we aim to eliminate all the correlations between feature maps. To this end, our HSIC regularization is:

$$\mathcal{R}_z(\mathbf{Z}) = \sum_{1 \le i < j \le m} \mathrm{HSIC}(\mathbf{Z}_{i,:}, \mathbf{Z}_{j,:}).$$

Note that HSIC is a nonparametric regularization term, so it does not increase the parameter size of deep OptEq. The computing cost of HSIC grows as the batch size and feature dimension increase. Some tricks, such as Random Fourier Features approximation [28], can be applied to speed up the calculation. In addition, Theorem 5 is only guaranteed for convex regularization, while HSIC regularization is non-convex. In this paper, we report the great empirical superiority of HSIC and leave the above issues to future work.

7.2.2 *Settings and Results*

Other Regularizers. Here is the function we utilize to introduce the customized property to the equilibrium point, see Section 5 for more details. For the regularizer $\mathcal{R}_z(\cdot)$ (both for $\mathcal{R}_z^\dagger$ and $\mathcal{R}_z^*$), we set four different settings: (1) $\mathcal{R}_z(\mathbf{Z}) = \sum_{1 \le i < j \le m} \mathrm{HSIC}(\mathbf{Z}_{i,:}, \mathbf{Z}_{j,:})$; (2) $\mathcal{R}_z(\mathbf{z}) = \frac{1}{2}\|\mathbf{z}\|^2$; (3) $\mathcal{R}_z(\mathbf{z}) = 1/\left(\|\mathbf{z}\|^2 + \epsilon\right)$ which is explored in [39]; (4) Decorrelation: for the B-batch equilibrium points matrix $\mathbf{Z} \in \mathbb{R}^{m \times B}$:

$$\mathcal{R}_z(\mathbf{Z}) = \frac{1}{2}\left\|\mathbf{D}\mathbf{Z}\mathbf{Z}^\top\mathbf{D} - \mathbf{I}\right\|_F^2 := \mathcal{F}_{Dz}(\mathbf{Z}), \tag{13}$$

where $\mathbf{D}$ is a diagonal matrix whose non-zero entries are $\frac{1}{\|\mathbf{Z}_{i,:}\|}$ and $\mathbf{Z}_{i,:}$ is the i-th row of the matrix $\mathbf{Z}$. Note that $\mathcal{R}_z(\mathbf{Z})$ here aims at reducing redundant information between feature dimensions, which has been discussed in [40].

Settings. In this experiment, we set $K = 20$ and utilize weight decay to regularize the learnable parameters, i.e., $\mathcal{R}_w(\cdot) = \xi\|\cdot\|^2$, where we choose $\xi = 3e-4$. We utilize the commonly used SGD to train the model. We set the learning rate as 0.1 at the beginning and decay it by 0.7 after every 20 epochs. And the total training epoch is 200. The batch size is 125 in this experiment. We construct the deep OptEqs with 5 convolutional layers, using five 3×3 convolution kernels with the numbers of channels being $16, 32, 64, 128, 128$. The total number of learnable parameters is 199k.

Results. The results with different regularizers are presented in Table 5. We can see that either adopting SAM iteration or changing the underlying convex optimization problem both improves classification performance. Note that, given the same type of regularization, modifying the underlying optimization problem, i.e., using $\mathcal{R}_z^*$, usually make more improvements. Indeed, to modify the underlying optimization problem, we need to change the architecture of deep OptEq, which has a more direct impact on the model than turning the training loss by $\mathcal{R}_z^\dagger$. When $\alpha = 0.01$, deep OptEq is almost equivalent to the one-layer wide OptEq (see Theorem 3), which is far outperformed by deep OptEq for $\alpha > 0.1$. Compared with other results, $\alpha = 0.1$ gives a poor result, which implies that the performance is not monotonic to parameter α. Fortunately, from the table, setting $\alpha > 0.4$ is a safe choice. We notice that the overall performance of feature disentanglement methods (decorrelation and HSIC) are better than the other types of regularization terms whether we utilize it as $\mathcal{R}_z^\dagger$ or $\mathcal{R}_z^*$.

7.3 Comparison with Previous Implicit Models

In this experiment, we compare deep OptEq with other implicit models MDEQ [4], NODEs [1], Augmented NODEs [41], single convolutional Monotone DEQs [8] (short as MON), and classical ResNet-xx [2]. Note that deep OptEqs do not require the additional re-parameterization like MONs [8]. Therefore, our OptDeq models cannot guarantee the uniqueness of the fixed point. However, the proposed SAM training strategy can select the point with the minimal regularization value when the fixed point set in not a singleton.

7.3.1 *Results on CIFAR-10*

For fair comparisons, we construct the deep OptEqs with 5 convolutional layers. In order to construct deep OptEqs

TABLE 5: The testing accuracy of deep OptEq with different settings. We denote by "no Reg" the SAM with $\beta_k = 0$. And the scripts † and $*$ means the regularizer given in Section 5.2 and Section 5.1, respectively.

α	No Reg	$\left(\frac{\|\cdot\|^2}{2}\right)^{\dagger}$	$\left(\frac{2}{\|\cdot\|^2+2\epsilon}\right)^{\dagger}$	$\left(\frac{2}{\|\cdot\|^2+2\epsilon}\right)^{*}$	$\mathcal{F}_{Dz}^{\dagger}$	$\mathcal{F}_{Dz}^{*}$	HSIC†	HSIC*
0.01	58.4%	69.4%	**75.0%**	64.9%	72.7%	72.6%	70.6%	72.2%
0.1	56.8%	61.9%	63.2%	64.7%	61.5%	60.0%	**66.1**%	64.5%
0.4	86.9%	87.3%	78.4%	87.7%	86.5%	**88.0%**	85.1%	85.5%
0.8	87.4%	87.3%	87.3%	87.5%	87.0%	**87.6%**	**87.6%**	87.5%
1.0	87.2%	87.4%	87.0%	87.6%	87.7%	87.5%	**88.1%**	87.9%

TABLE 6: The detailed Archit. of OptEqs. The c_{in}, c_{out}'s for the final class layer are $512 \rightarrow 1024 \rightarrow 2048$. The learnable parameters for each OptEq blocks are two depthwise separable convolution kernels.

	Extractor (size)	Archit. of OptEq	Archit. of cls. layer	Model Size
OptEq-Small	ResNet-50 (8M)	$\begin{bmatrix} 3\times3\times1, 512 \\ 1\times1\times512, 512 \end{bmatrix} \times 5$	$\begin{bmatrix} 1\times1\times c_{in}, c_{out} \\ \text{ReLU} \\ \text{MaxPool} \end{bmatrix} \times 2$ + AvgPool	18M (8M+7M+3M)
OptEq-Midddle	Wide-ResNet-50 (24M)	$\begin{bmatrix} 5\times5\times1, 1024 \\ 1\times1\times1024, 512 \\ 3\times3\times1, 512 \\ 1\times1\times512, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1\times c_{in}, c_{out} \\ \text{ReLU} \\ \text{MaxPool} \end{bmatrix} \times 2$ + AvgPool	40M (24M+13M+3M)

TABLE 7: Evaluation results on ImageNet classification with top-1 and top-5 accuracies.

	Models	top1 Acc.	top5 Acc.	Size
Explicit	ResNet-18	70.2%	89.9%	13M
	ResNet-34	74.8%	91.1%	21M
	Inception-V2	74.8%	92.2%	12M
	ResNet-50	75.1%	92.5%	26M
	Res-Extr. + Cls.	71.2%	89.5%	11M
	HRNet-W18	76.8%	**93.4%**	21M
Implicit	MDEQ-single branch	72.9%	91.0%	18M
	MDEQ-small	75.5%	92.7%	18M
	OptEq (learnable α's)	76.7%	93.1%	18M
	OptEq-small	**76.9%**	93.1%	18M
Explicit	ResNet-101	77.1%	93.5%	52M
	W-ResNet-50	78.1%	93.9%	69M
	W-Res-Extr. + Cls.	73.4%	90.9%	27M
Implicit	MDEQ-large	77.5%	93.6%	63M
	MDEQ (Unrolled)	75.9%	93.0%	63M
	OptEq-Mid.	**78.3%**	**94.0%**	40M

with a similar number of parameters as baseline methods, we use five 3×3 convolution kernels with the numbers of channels being $16, 32, 64, 64, 128$. Moreover, we only use a single convolutional layer as the feature extractor $g(\cdot)$ for the model with 162k parameters, which is the same as single convolution MONs [8].

The results are shown in Table 4. Notably, even *without feature regularization trick*, our deep OptEqs significantly outperform baseline methods. We highlight the performance of deep OptEqs on CIFAR-10 which outperforms Augmented Neural ODE by 25.1% and MON by 11.6% with *fewer parameters*. Without adding the number of parameters, feature regularization helps deep OptEq to achieve better performance easily. Notably, HSIC, a feature disentanglement regularization, provides a significant gain for the generalization.

7.3.2 Results on ImageNet

We now consider the ability of OptEq on a large-scale dataset with higher-resolution images—ImageNet [42].

Extractors. To train OptEq on ImageNet, we choose a slightly more complex extractor $g(\cdot)$. In this experiment, we test deep OptEq on two extractors: (1) the first two stages of ResNet-50, whose model size is 8M, and maps the images from the size $224 \times 224 \times 3$ to a feature map belongs to $\mathbb{R}^{28\times28\times512}$; (2) the first two stages of Wide-ResNet-50 [43], with model size 24M and also maps the image to the size of $\mathbb{R}^{28\times28\times512}$. Note that the parameters in these extractors are also trained from scratch. We *do not* utilize any pre-trained weights here. We also report the vanilla results on these extractors, i.e., directly append the classification layer after the extractors (refer as Extr. + Cls.).

Architecture. Given the feature map $\mathbf{z} \in \mathbb{R}^{c_{in}\times w\times h}$, different from other experiments in this paper, which performs the linear transformation by a general convolution operator, namely, $\mathbf{W}_l\mathbf{z} := \mathrm{conv}(\mathbf{z}, \mathbf{W}_l)$, where $\mathbf{W}_l \in \mathbb{R}^{c_{out}\times c_{in}\times 3\times 3}$ is the convolutional kernel. For OptEqs on ImageNet, we choose the depthwise separable convolutions [44], i.e., $\mathbf{W}_l\mathbf{z} := \mathrm{conv}(\mathrm{conv}(\mathbf{z}, \mathbf{W}_{l,1}), \mathbf{W}_{l,2}))$, where $\mathbf{W}_{l,1} \in \mathbb{R}^{c_{in}\times 1\times 3\times 3}$ and $\mathbf{W}_{l,2} \in \mathbb{R}^{c_{out}\times c_{in}\times 1\times 1}$ are the convolutional kernels. The architecture details are summarized in Table 6.

Settings. We choose the the decorrelation function eq.(13) as the regularizer here and adopt the regularizer setting given in Section 5.1. For computational and memory efficiency, we let $K = 5$ here. The optimizer in this experiment is AdamW with weight decay being 0.05, learning rate being 0.001 with cosine decay scheduler and batch size being 1024. We train OptEqs for 300 epochs.

Results. Table 7 shows the accuracy of two different size deep OptEqs, i.e., OptEq-small and OptEq-Mid, in comparison to well-known reference models in computer

TABLE 8: Comparison between Unrolling and IFT based Training (# params 199k)

Method	Acc.	Inference Time	Back-Prop Time	Relative Residual
Unrolling (K=5)	83.52%	**1.3**s	**1.8**s	1.22e-02
Unrolling (K=10)	87.28%	2.2s	3.3s	4.88e-03
Unrolling (K=20)	87.71%	3.9s	6.4s	4.81e-04
Unrolling (K=40)	**87.83**%	7.5s	12.7s	**1.20e-05**
IFT (thd = 1e-03)	87.63%	16.3s	6.7s	7.33e-04
IFT (thd = 1e-02)	85.95%	15.6s	1.3s	9.52e-03

TABLE 9: Comparison between FPS on ImageNet. We align all training-related hyper parameters, e.g., batch size, number of GPUs, etc. For MDEQ we set downsample to 2.

	Models	FPS (images/s)	Size
Explicit	ResNet-50	4597/s	26M
Implicit	MDEQ-single branch	257/s	18M
	MDEQ	1057/s	18M
	OptEq (unrolling K=5)	1835/s	18M
	OptEq (IFT with dw-conv)	200/s	18M
	OptEq (IFT w/o dw-conv)	1137/s	18M

TABLE 10: Evaluation on the validation set of Cityscapes semantic segmentation.

Method	mIoU	mAcc	aAcc
FCN	71.47	79.23	95.56
Deep OptEq	74.47	81.91	95.93

vision. OptEqs outperforms the current SOTA equilibrium models and are remarkably competitive with some strong explicit models. For example, the OptEq-small with 18M parameters outperforms DEQ (classical equilibrium models with 18M parameters), ResNet-34 (21M parameters), and even ResNet-50 (26M parameters). The larger OptEq-Mid (40M parameters) reaches higher level of performance comparing to ResNet-101 (52M parameters) and MDEQ-large (SOTA equilibrium models with 63M parameters). OptEq's results are far beyond the scale and accuracy levels of prior equilibrium models.

Notably, OptEq significantly improves the results in comparison to the pure extractor in addition to classification layer (e.g., ResNet-50 (Extr.) and W-ResNet-50 (Extr.)). Moreover, even with much smaller model size, the performance of OptEq early goes beyond the bars given by the original ResNet-50 and W-ResNet-50. Hence, we can conclude that the superiority mainly comes from our OptEqs rather than extractors.

We also set different α's for different layers. In general, it is hard to tune the parameters manually. Hence we let the model learn α by itself. We initiate them to 0.8 and update them by BP. The result in Table 7 shows that the learned alpha can obtain a comparable performance with the fixed one. Although intuitively learnable α's are more reasonable for model design, it increases the difficulty of model training. Hence, we may not always get better results in this case. How to overcome the unstable behaviors of univariate variables during DNN training is still an open problem. Note that, the learnable α's converge near $[0.63, 0.72, 0.86, 0.9, 0.87]$ finally.

7.4 Efficiency and Approximation Error

We train deep OptEqs by the IFT based way given in [3] and compare the results with the unrolling way. The time for inference and BP is provided, and it is the total time for 80 iteration steps with the batch size being 125 on GPU NVIDIA GTX 1070. The relative residual is averaged over all *test* batches: $\|\mathbf{z}^K - \mathcal{T}(\mathbf{z}^K, \mathbf{x}, \boldsymbol{\theta})\|_2 / \|\mathbf{z}^K\|_2$. For fair comparison, we do not utilize any feature regularization in this experiment. We set $\alpha = 0.8$ and let "thd" represent the residual threshold. In order to accelerate convergence speed of the iteration and stabilize OptEq, it is crucial to make sure that the operator norms of the initial weight matrices are small. Therefore, we use initialization schemes with small values (around 0). Moreover, experimental results indicate that initialization schemes with small values do not affect the performance of OptEqs.

The results are given in Table 8, although IFT based methods consume much less memory, given the comparable relative residual, the unrolling methods achieve better performance with much less inference and BP time. Note that a loose residual threshold may destroy the IFT based method significantly. We should choose the appropriate training method according to practice. For IFT, the inference time is longer than the BP one since the fixed point equation needed to solve during inference is non-linear, which is more challenging than the linear one during BP.

A more practice comparison of the efficiency on the large vision dataset is given in Table 9. We only consider the methods that release the codes on this dataset. We align all the hyper-parameters that may affect the FPS, such as fixed-point algorithms (Broyden methods), batch size (768), number of GPUs (8) and workers (8), etc. This experiment is performed in the distributed data-parallel model on 8 Tesla A100.

In general, at the cost of model size, the training speed for explicit models is much faster than the implicit models. When only considering the equilibrium methods, we can find that unrolling-based ways are much more efficient. The main reason is that the fixed point is hard to obtain when the forward function of the equilibrium model is complicated. On the other hand, without proper temporary results of computation graph, we may pay a lot of computing load to carry out BP. However, the IFT-based way is a good choice if the memory of GPUs is limited. By the way, in the case of utilizing depthwise separable convolution, our method is inefficient. If use raw convolution, our efficiency is comparable to MDEQ (we adjust some hidden layers to align the size of model). Therefore, we choose the unrolling way to train the large Opteqs.

7.5 Cityscapes Semantic Segmentation

In this experiment, we evaluate the empirical performance of our deep OptEq on a large-scale computer vision task: semantic segmentation on the Cityscapes dataset. We construct a deep OptEq with only three weighted layers and channels of 256, 512 and 512. The deep OptEq is used as the "backbone" of the segmentation network. We compare our method with FCN [45] on the Cityscapes test set. We employ the poly learning rate policy to adjust the learning rate, where the initial learning rate is multiplied by $(1 - iter/total\_iter)^{0.9}$ after each iteration. The initial learning rate is set to be 0.01 for both networks. Moreover, momentum and weight decay are set to 0.9 and 0.001, respectively. Note that we only train on finely annotated data. We train the model for 40K iterations, with mini-batch size set as 8. The results on the validation set are shown in Table 10. Notably, our deep OptEq significantly outperforms FCN with a similar number of parameters. Note that in this experiment, we have not introduced any customized property of the feature, so the performance improvement is entirely due to the superiority of the implicit structure of deep OptEq.

8 Conclusions

In this paper, we decompose the feed-forward DNN and find a more reasonable basic unit layer, which shows a close relationship with the proximal operator. Based on it, we propose new equilibrium models, OptEqs, and explore their underlying optimization problems thoroughly. We provide two strategies to introduce customized regularizations to the equilibrium points, and achieve significant performance improvement in experiments. We highlight that by modifying the underlying optimization problems, we can create more effective network architectures. Our work may inspire more interpretable equilibrium models from the optimization perspective.

Acknowledgments

This work is supported in part by New Generation AI Major Project of Ministry of Science and Technology of China under Grant 2018AAA0102501, in part by national Natural Science Foundation of China (NSFC) under Grant U21B2027. Z. Lin was supported by the NSF China under Grant 61625301 and 61731018, and Project 2020BD006 supported by PKU-Baidu Fund. We also appreciate Huan Li (NKU), Yifei Wang (PKU) and Zhenyang Geng (CMU) for their valuable and detailed discussions.

References

[1] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, "Neural ordinary differential equations," in *Advances in Neural Information Processing Systems*, 2018, pp. 6572–6583.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[3] S. Bai, J. Z. Kolter, and V. Koltun, "Deep equilibrium models," in *Advances in Neural Information Processing Systems*, 2019, pp. 690–701.

[4] S. Bai, V. Koltun, and J. Z. Kolter, "Multiscale deep equilibrium models," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[5] J. Li, M. Xiao, C. Fang, Y. Dai, C. Xu, and Z. Lin, "Training neural networks by lifted proximal operator machines," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[6] H. Ramsauer, B. Schäfl, J. Lehner, P. Seidl, M. Widrich, T. Adler, L. Gruber, M. Holzleitner, M. Pavlović, G. K. Sandve *et al.*, "Hopfield networks is all you need," *arXiv preprint arXiv:2008.02217*, 2020.

[7] J. Li, C. Fang, and Z. Lin, "Lifted proximal operator machines," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 4181–4188.

[8] E. Winston and J. Z. Kolter, "Monotone operator equilibrium networks," *arXiv preprint arXiv:2006.08591*, 2020.

[9] M. Revay, R. Wang, and I. R. Manchester, "Lipschitz bounded equilibrium networks," *arXiv preprint arXiv:2010.01732*, 2020.

[10] X. Hong-Kun, "Viscosity approximation methods for nonexpansive mappings," *Journal of Mathematical Analysis and Applications*, vol. 298, no. 1, pp. 279–291, 2004.

[11] S. Sabach and S. Shtern, "A first order method for solving convex bilevel optimization problems," *SIAM Journal on Optimization*, vol. 27, no. 2, pp. 640–660, 2017.

[12] S. W. Fung, H. Heaton, Q. Li, D. McKenzie, S. Osher, and W. Yin, "Fixed point networks: Implicit depth models with jacobian-free backprop," *arXiv preprint arXiv:2103.12803*, vol. 8, no. 9, 2021.

[13] S. Gurumurthy, S. Bai, Z. Manchester, and J. Z. Kolter, "Joint inference and input optimization in equilibrium networks," in *Advances in Neural Information Processing Systems*, 2021.

[14] K. Kawaguchi, "On the theory of implicit deep learning: Global convergence with implicit layers," *arXiv preprint arXiv:2102.07346*, 2021.

[15] C. B. Do, C.-S. Foo, and A. Y. Ng, "Efficient multiple hyperparameter learning for log-linear models." in *Advances in Neural Information Processing Systems*, vol. 2007, 2007, pp. 377–384.

[16] L. Franceschi, P. Frasconi, S. Salzo, R. Grazzi, and M. Pontil, "Bilevel programming for hyperparameter optimization and meta-learning," in *International Conference on Machine Learning*. PMLR, 2018, pp. 1568–1577.

[17] H. Liu, K. Simonyan, and Y. Yang, "Darts: Differentiable architecture search," *arXiv preprint arXiv:1806.09055*, 2018.

[18] Z. Ramzi, F. Mannel, S. Bai, J.-L. Starck, P. Ciuciu, and T. Moreau, "Shine: Sharing the inverse estimate from the forward pass for bilevel optimization and implicit models," *arXiv preprint arXiv:2106.00553*, 2021.

[19] J. Lorraine, P. Vicol, and D. Duvenaud, "Optimizing millions of hyperparameters by implicit differentiation," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 1540–1552.

[20] R. Liu, J. Gao, J. Zhang, D. Meng, and Z. Lin, "Investigating bilevel optimization for learning and vision from a unified perspective: A survey and beyond," *arXiv preprint arXiv:2101.11517*, 2021.

[21] J. Liu, X. Chen, Z. Wang, and W. Yin, "Alista: Analytic weights are as good as learned weights in lista," in *International Conference on Learning Representations*, 2019.

[22] X. Xie, J. Wu, G. Liu, Z. Zhong, and Z. Lin, "Differentiable linearized ADMM," in *International Conference on Machine Learning*, 2019, pp. 6902–6911.

[23] K. Wei, A. Aviles-Rivero, J. Liang, Y. Fu, C.-B. Schönlieb, and H. Huang, "Tuning-free plug-and-play proximal algorithm for inverse imaging problems," in *International Conference on Machine Learning*. PMLR, 2020, pp. 10 158–10 169.

[24] B. Amos and J. Z. Kolter, "OptNet: Differentiable optimization as a layer in neural networks," in *International Conference on Machine Learning*, 2017, pp. 136–145.

[25] A. Agrawal, B. Amos, S. Barratt, S. Boyd, S. Diamond, and J. Z. Kolter, "Differentiable convex optimization layers," in *Advances in Neural Information Processing Systems*, 2019, pp. 9562–9574.

[26] R. A. Amjad and B. C. Geiger, "Learning representations for neural network-based classification using the information bottleneck principle," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 9, pp. 2225–2239, 2019.

[27] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, "Deep variational information bottleneck," *arXiv preprint arXiv:1612.00410*, 2016.
[28] X. Zhang, P. Cui, R. Xu, L. Zhou, Y. He, and Z. Shen, "Deep stable learning for out-of-distribution generalization," *arXiv preprint arXiv:2104.07876*, 2021.
[29] D. Soudry, E. Hoffer, M. S. Nacson, S. Gunasekar, and N. Srebro, "The implicit bias of gradient descent on separable data," *The Journal of Machine Learning Research*, vol. 19, no. 1, pp. 2822–2878, 2018.
[30] N. Razin and N. Cohen, "Implicit regularization in deep learning may not be explainable by norms," *Advances in Neural Information Processing Systems*, vol. 33, 2020.
[31] A. Rajeswaran, C. Finn, S. Kakade, and S. Levine, "Meta-learning with implicit gradients," *Advances in Neural Information Processing Systems*, 2019.
[32] M. Frigon, "Fixed point and continuation results for contractions in metric and gauge spaces," *Banach Center Publications*, vol. 77, p. 89, 2007.
[33] R. Vershynin, *High-dimensional probability: An introduction with applications in data science*. Cambridge University Press, 2018, vol. 47.
[34] S. Bai, V. Koltun, and J. Z. Kolter, "Stabilizing equilibrium models by jacobian regularization," in *International Conference on Machine Learning (ICML)*, 2021.
[35] Z. Geng, X.-Y. Zhang, S. Bai, Y. Wang, and Z. Lin, "On training implicit models," *Advances in Neural Information Processing Systems*, 2021.
[36] M. Takada, T. Suzuki, and H. Fujisawa, "Independently interpretable lasso: A new regularizer for sparse regression with uncorrelated variables," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2018, pp. 454–463.
[37] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.
[38] L. Song, A. Smola, A. Gretton, J. Bedo, and K. Borgwardt, "Feature selection via dependence maximization." *Journal of Machine Learning Research*, vol. 13, no. 5, 2012.
[39] Y. Yuan, K. Yang, and C. Zhang, "Feature incay for representation regularization," *arXiv preprint arXiv:1705.10284*, 2017.
[40] B. O. Ayinde, T. Inanc, and J. M. Zurada, "Regularizing deep neural networks by enhancing diversity in feature extraction," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2650–2661, 2019.
[41] E. Dupont, A. Doucet, and Y. W. Teh, "Augmented neural ODEs," *arXiv preprint arXiv:1904.01681*, 2019.
[42] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 248–255.
[43] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.
[44] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1251–1258.
[45] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.
[46] R. Gribonval and M. Nikolova, "A characterization of proximity operators," *Journal of Mathematical Imaging and Vision*, vol. 62, p. 773–789, 2020.
[47] A. Beck, *First-order methods in optimization*. SIAM, 2017.
[48] H.-K. Xu, "Iterative algorithms for nonlinear operators," *Journal of the London Mathematical Society*, vol. 66, no. 1, pp. 240–256, 2002.
[49] H. H. Bauschke, P. L. Combettes *et al.*, *Convex analysis and monotone operator theory in Hilbert spaces*. Springer, 2011, vol. 408.

Xingyu Xie (S'17) is a Ph.D. candidate in the School of Artificial Intelligence, Peking University. His research interests include machine learning and optimization.

Qiuhao Wang is a master student in the School of Artificial Intelligence, Peking University. He received the B.S. degree in Department of Mathematical Sciences from Tsinghua University in 2019. His research interests include deep learning and optimization.

Zenan Ling is a postdoctoral researcher with the Key Lab. of Machine Perception, Artificial Intelligence, Peking University. He is also a postdoctoral researcher with the Pazhou Lab. He received the Ph.D. degree in Department of Electrical Engineering from Shanghai Jiaotong University in 2020, and the B.S. degree in Department of Mathematics from Nanjing University in 2015. His research interests include random matrix theory and machine learning.

Xia Li is a Ph.D. student in the Department of Computer Science, ETH Zurich. He received the Master's degree in School of Electronic and Computer Engineering from Peking University in 2020, and the B.S. degree in School of Computer Science from Beijing University of Posts and Telecommunications in 2017. His research interests include image segmentation and image translation.

Guangcan Liu (M'11–SM'18) received the bachelor's degree in mathematics and the PhD degree in computer science and engineering from Shanghai Jiao Tong University, Shanghai, China, in 2004 and 2010, respectively. He was a postdoctoral researcher with the National University of Singapore, Singapore, from 2011 to 2012, the University of Illinois at Urbana-Champaign, Champaign, IL, from 2012 to 2013, Cornell University, Ithaca, NY, from 2013 to 2014, and Rutgers University, Piscataway, NJ, in 2014. From 2014 to 2020, he was a professor with the School of Automation, Nanjing University of Information Science and Technology. Since 2020, he has been a professor with School of Automation, Southeast University, Nanjing, China. His research interests touch on the areas of machine learning, computer vision and signal processing. He is a senior member of the IEEE.

Zhouchen Lin (M'00–SM'08–F'18) received the Ph.D. degree in applied mathematics from Peking University in 2000. He is currently a Professor with the Key Laboratory of Machine Perception (MOE), School of Artificial Intelligence, Peking University. His research interests include computer vision, image processing, machine learning, pattern recognition, and numerical optimization. He is Area Chairs of ACML, ACCV, CVPR, ICCV, NIPS/NeurIPS, AAAI, IJCAI, ICLR, and ICML many times, and is a Program co-Chair of ICPR 2022 and a Senior Area Chair of ICML 2022. He was an Associate Editor of the IEEE Transactions on Pattern Analysis and Machine Intelligence and currently is an associate editor of the International Journal of Computer Vision. He is a Fellow of the IAPR and the IEEE.

Optimization Induced Equilibrium Networks: An Explicit Optimization Perspective for Understanding Equilibrium Models
(Supplementary Material)

APPENDIX A
PROOFS FOR THE DNN REFORMULATION

A.1 Proof of Lemma 1

The formal proof for Lemma 1 relies on the following auxiliary lemma.

Lemma 4. *If $k \geq 2\max\{m,n\}$, given any $\mathbf{W} \in \mathbb{R}^{m\times n}$, and a full rank matrix $\mathbf{A} \in \mathbb{R}^{m\times k}$, there exists a full rank matrix $\mathbf{B} \in \mathbb{R}^{n\times k}$, such that $\mathbf{W} = \mathbf{A}\mathbf{B}^\top$.*

Proof. Considering the full SVD of $\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$, where $\mathbf{U} \in \mathbb{R}^{m\times m}$, $\boldsymbol{\Sigma} \in \mathbb{R}^{m\times k}$, and $\mathbf{V} \in \mathbb{R}^{k\times k}$. Let

$$\mathbf{U}^\top\mathbf{W} := \boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{\Omega}_1 \\ \boldsymbol{\Omega}_2 \\ \vdots \\ \boldsymbol{\Omega}_m \end{bmatrix}.$$

Considering the equation:

$$\boldsymbol{\Omega} = \boldsymbol{\Sigma}\mathbf{C}.$$

We can easily find that:

$$\mathbf{C} = \begin{bmatrix} \boldsymbol{\Omega}_1/\sigma_1 \\ \boldsymbol{\Omega}_2/\sigma_2 \\ \vdots \\ \boldsymbol{\Omega}_m/\sigma_m \\ * \end{bmatrix},$$

is a solution, and we let $\operatorname{rank}(\mathbf{C}) = n$ by adjusting $*$. We let $\mathbf{B} = \mathbf{C}^\top\mathbf{V}^\top$, hence, we can conclude that $\operatorname{rank}(\mathbf{B}) = n$. It is easy to verify that $\mathbf{W} = \mathbf{A}\mathbf{B}^\top$. □

Recall that, we have:

$$\begin{aligned} \mathbf{y} =& \mathbf{W}_L\sigma\Big(\mathbf{W}_{L-1}\sigma(\cdots\mathbf{W}_2\sigma(\mathbf{W}_1\mathbf{z}_0 + \mathbf{U}_1\mathbf{x} + \mathbf{b}_1)\cdots)\Big) \\ =& \underbrace{\overline{\mathbf{W}}_L\,\overline{\mathbf{W}}_{L-1}^\top}_{\mathbf{W}_L}\sigma\left(\underbrace{\overline{\mathbf{W}}_{L-1}\,\overline{\mathbf{W}}_{L-2}^\top}_{\mathbf{W}_{L-1}}\sigma\left(\cdots\underbrace{\overline{\mathbf{W}}_2\,\overline{\mathbf{W}}_1^\top}_{\mathbf{W}_2}\sigma(\underbrace{\overline{\mathbf{W}}_1\,\overline{\mathbf{W}}_0^\top}_{\mathbf{W}_1}\mathbf{z}_0 + \mathbf{U}_1\mathbf{x} + \mathbf{b}_1)\cdots\right)\right). \end{aligned}$$

Based on the above Lemma 4, the existence of $\overline{\mathbf{W}}_k \in \mathbb{R}^{n_k\times m}$ for all k can be easily guaranteed by giving a rank m matrix $\overline{\mathbf{W}}_L$. Note that $\forall k, \operatorname{rank}(\overline{\mathbf{W}}_k) = n_k$ and $m \geq 2\max\{n_k, n_{k-1}\}$.

A.2 Proof of Lemma 2

Proof. Set $\alpha = 1$ and let $\mathbf{W}_1 = \mathbf{0}, \mathbf{U}_1 = \mathbf{U}_3 = \cdots = \mathbf{U}_L = 0$ and $\mathbf{b}_i = 0$ for all i, we have:

$$\mathbf{A}_L\sigma\Big(\mathbf{A}_{L-1}\sigma\cdots\sigma(\mathbf{A}_2\sigma(\mathbf{A}_1\mathbf{x}))\Big) = \underbrace{\mathbf{W}_{L+1}\mathbf{W}_L^\top}_{\mathbf{A}_L}\sigma\Big(\underbrace{\mathbf{W}_L\mathbf{W}_{L-1}^\top}_{\mathbf{A}_{L-1}}\sigma\cdots\sigma\Big(\underbrace{\mathbf{W}_3\mathbf{W}_2^\top}_{\mathbf{A}_2}\sigma\Big(\underbrace{\mathbf{U}_2}_{\mathbf{A}_1}\mathbf{x}\Big)\Big)\Big).$$

The existence of $\{\mathbf{W}_l\}_{l=2}^{L+1}$ can be easily obtained by Lemma 4. The bias term $\mathbf{c}_i$ can be easily included by changing each layer's output $\mathbf{x}_i$ to $[\mathbf{x}_i; 1]$ and set $\mathbf{A}_i$ to $\begin{bmatrix}\mathbf{A}_i & \mathbf{c}_i \\ 0 & 1\end{bmatrix}$. □

APPENDIX B
PROOFS FOR THE CONNECTION BETWEEN OPTIMIZATION AND OPTEQ

B.1 Conditions to be a Proximal Operator

Lemma 5 (modified version of Prop. 2 in [46]). *Consider $f : \mathcal{H} \to \mathcal{H}$ defined everywhere. The following properties are equivalent:*

(i) there is a proper convex l.s.c function $\varphi : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ such that $f(\mathbf{z}) \in \operatorname{prox}_\varphi(\mathbf{z})$ for each $\mathbf{z} \in \mathcal{H}$;

(ii) the following conditions hold jointly:

(a) there exists a convex l.s.c function $\psi : \mathcal{H} \to \mathbb{R}$ such that $\forall \mathbf{y} \in \mathcal{H}$, $f(\mathbf{y}) = \nabla\psi(\mathbf{y})$;

(b) $\|f(\mathbf{y}) - f(\mathbf{y}')\| \leq \|\mathbf{y} - \mathbf{y}'\|$, $\forall \mathbf{y},\ \mathbf{y}' \in \mathcal{H}$.

There exists a choice of $\varphi(\cdot)$ and $\psi(\cdot)$, satisfying (i) and (ii), such that $\varphi(\mathbf{x}) = \psi^*(\mathbf{x}) - \frac{1}{2}\|\mathbf{x}\|^2$.

Proof. (i)⇒(ii): Since $\varphi(\mathbf{x}) + \frac{1}{2}\|\mathbf{x}\|^2$ is a proper l.s.c 1-strongly convex function, then by Thm. 5.26 in [47], i.e, the conjugate function f^* is $\frac{1}{\sigma}$-smooth when f is proper, closed and σ strongly convex and vice versa. Thus, we have:

$$\psi(\mathbf{x}) := \left[\varphi(\mathbf{x}) + \frac{1}{2}\|\mathbf{x}\|^2\right]^*,$$

is 1-smooth with $\operatorname{dom}(\psi) = \mathcal{H}$. Then we get:

$$\begin{aligned} f(\mathbf{x}) \in \underset{\mathbf{u}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{u} - \mathbf{x}\|^2 + \varphi(\mathbf{u}) &= \{\mathbf{u} \mid \mathbf{x} \in \partial\varphi(\mathbf{u}) + \mathbf{u}\} \\ &= \left\{\mathbf{u} \mid \mathbf{x} \in \partial\left(\varphi(\mathbf{u}) + \frac{1}{2}\|\mathbf{u}\|^2\right)\right\} \\ &= \{\mathbf{u} \mid \mathbf{u} = \nabla\psi(\mathbf{x})\} = \{\nabla\psi(\mathbf{x})\} \end{aligned}$$

where the third equality comes from Thm. 4.20 in in [47], which notes that $\mathbf{y} \in \partial f(\mathbf{x})$ is equivalent to $\mathbf{x} \in \partial f^*(\mathbf{y})$. Hence $f(\mathbf{x}) = \nabla\psi(\mathbf{x})$, and 1-smoothness of ψ implies f is nonexpansive.

(ii)⇒(i): Let $\varphi(\mathbf{x}) = \psi^*(\mathbf{x}) - \frac{1}{2}\|\mathbf{x}\|^2$. Since ψ is 1-smooth, similarly we can conclude: ψ^* is 1-strongly convex. Hence, φ is convex, and:

$$\begin{aligned} \operatorname{prox}_\varphi(\mathbf{x}) &= \underset{\mathbf{u}}{\operatorname{argmin}} \left\{\frac{1}{2}\|\mathbf{u} - \mathbf{x}\|^2 + \varphi(\mathbf{u})\right\} \\ &= \{\mathbf{u} \mid \mathbf{x} \in \partial\varphi(\mathbf{u}) + \mathbf{u}\} \\ &= \{\nabla\psi(\mathbf{x})\} = \{f(\mathbf{x})\}, \end{aligned}$$

which means $f(\mathbf{x}) = \operatorname{prox}_\varphi(\mathbf{x})$. □

B.2 Proof for Theorem 1

Proof. In the proof, w.l.o.g, we let $\mu = 1$ for the ease of presentation, and hence let $\tilde{L}_\sigma = 1$ and $\|\mathbf{W}\| \leq 1$. Since $\mathbf{1}^\top\tilde{\sigma}(\mathbf{y}) = \sum_{i=1}^n \tilde{\sigma}(y_i)$, we have $\nabla(\tilde{\sigma}(\mathbf{y})) = [\sigma(y_1), \cdots, \sigma(y_n)]^\top = \sigma(\mathbf{y})$, by the chain rule, $\nabla\psi(\mathbf{z}) = \mathbf{W}^\top\sigma(\mathbf{W}\mathbf{z} + \mathbf{b}) = f(\mathbf{z})$.

Since $\sigma(a)$ is a single-valued function with slope in $[0, 1]$, the element-wise defined operator $\sigma(\mathbf{a})$ is nonexpansive (see the definition in Lemma 5). Combining with $\|\mathbf{W}\|_2 \leq 1$, operator $f(\mathbf{z}) = \mathbf{W}^\top\sigma(\mathbf{W}\mathbf{z} + \mathbf{b})$ is also nonexpansive.

Due to Lemma 5, we have $f(\mathbf{z}) = \operatorname{prox}_\varphi(\mathbf{z})$, and $\varphi(\mathbf{z})$ can be chosen as $\varphi(\mathbf{z}) = \psi^*(\mathbf{z}) - \frac{1}{2}\|\mathbf{z}\|^2$ □

B.3 Proof of Lemma 3

Proof. We can rewrite deep OptEq $\mathbf{z} = f_L \circ f_{L-1} \cdots \circ f_1(\mathbf{z}, \mathbf{x}, \boldsymbol{\theta})$ in a separated form: let $\mathbf{z} = \mathbf{z}_0$,

$$\begin{cases} \mathbf{z}_1 = \alpha\mathbf{W}_1^\top\sigma\left(\mathbf{W}_1\mathbf{z}_0 + \mathbf{U}_1\mathbf{x} + \mathbf{b}_1\right) + (1-\alpha)\mathbf{z}_0 \\ \mathbf{z}_2 = \alpha\mathbf{W}_2^\top\sigma\left(\mathbf{W}_2\mathbf{z}_1 + \mathbf{U}_2 x + \mathbf{b}_2\right) + (1-\alpha)\mathbf{z}_1 \\ \vdots \\ \mathbf{z}_{L-1} = \alpha\mathbf{W}_{L-1}^\top\sigma\left(\mathbf{W}_{L-1}\mathbf{z}_{L-2} + \mathbf{U}_{L-1} x + \mathbf{b}_{L-1}\right) + (1-\alpha)\mathbf{z}_{L-2} \\ \mathbf{z}_0 = \alpha\mathbf{W}_L^\top\sigma\left(\mathbf{W}_L\mathbf{z}_{L-1} + \mathbf{U}_L x + \mathbf{b}_L\right) + (1-\alpha)\mathbf{z}_{L-1} \end{cases}, \tag{14}$$

and it also has a compact matrix form:

$$
\begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_{L-1} \\ \mathbf{z}_0 \end{bmatrix} = \alpha \begin{bmatrix} \mathbf{W}_1^\top & & & & \\ & \mathbf{W}_2^\top & & & \\ & & \mathbf{W}_3^\top & & \\ & & & \ddots & \\ & & & & \mathbf{W}_L^\top \end{bmatrix} \sigma\left(\begin{bmatrix} 0 & & & & \mathbf{W}_1 \\ \mathbf{W}_2 & 0 & & & \\ & \mathbf{W}_3 & \ddots & & \\ & & \ddots & \ddots & \\ & & & \mathbf{W}_L & 0 \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_{L-1} \\ \mathbf{z}_0 \end{bmatrix} + \right.
$$
$$
\left. \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_{L-1} \\ \mathbf{U}_L \end{bmatrix} x + \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_{L-1} \\ \mathbf{b}_L \end{bmatrix} \right) + (1-\alpha)\mathbf{P} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_{L-1} \\ \mathbf{z}_0 \end{bmatrix}, \tag{15}
$$

where $\mathbf{P} = \begin{bmatrix} 0 & & & & \mathbf{I} \\ \mathbf{I} & 0 & & & \\ & \mathbf{I} & 0 & & \\ & & \ddots & \ddots & \\ & & & \mathbf{I} & 0 \end{bmatrix}$ is a permutation matrix.

Hence a multi-layer deep OptEq is actually a single-layer OptEq with multi-blocks. □

B.4 Proof of Theorem 2

Before proving the main results, we first present an auxiliary lemma.

Lemma 6 (an extension of Lemma 5)**.** *Consider $f : \mathcal{H} \to \mathcal{H}$ defined everywhere, $\mathcal{A} : \mathcal{H} \to \mathcal{H}$ is any invertible 1-Lipschitz operator. The following properties are equivalent:*

(i) there is a proper convex l.s.c function $\varphi : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ such that $f(\mathbf{z}) \in \operatorname*{argmin}_{\mathbf{u}} \left\{ \frac{1}{2}\|\mathbf{u}\|^2 - \langle \mathbf{u}, \mathcal{A}(\mathbf{z}) \rangle + \varphi(\mathbf{u}) \right\}$ for each $\mathbf{z} \in \mathcal{H}$;

(ii) the following conditions hold jointly:

- *(a) there exists a convex l.s.c function $\psi : \mathcal{H} \to \mathbb{R}$ such that for each $\mathbf{y} \in \mathcal{H}$, $f\big(\mathcal{A}^{-1}(\mathbf{y})\big) = \nabla\psi(\mathbf{y})$;*
- *(b) f is nonexpansive, i.e. $\|f(\mathbf{y}) - f(\mathbf{y}')\| \le \|\mathbf{y} - \mathbf{y}'\|, \ \ \forall \mathbf{y}, \ \mathbf{y}' \in \mathcal{H}$.*

Moreover, there exists a choice of $\varphi(\cdot)$, $\psi(\cdot)$, satisfying (i) (ii), such that $\varphi(\mathbf{x}) = \psi^*(\mathbf{x}) - \frac{1}{2}\|\mathbf{x}\|^2$.

Proof. (i)⇒(ii): Since $\varphi(\mathbf{x}) + \frac{1}{2}\|\mathbf{x}\|^2$ is a proper l.s.c 1-strongly convex function, then $\psi(\mathbf{x}) := \left[\varphi(\cdot) + \frac{1}{2}\|\cdot\|^2\right]^*(\mathbf{x})$ is 1-smooth with $\operatorname{dom}(\psi) = \mathcal{H}$. Note that $\psi \circ \mathcal{A}(\cdot)$ is 1-smooth due to $\mathcal{A}(\cdot)$ is 1-Lipschitz. Moreover, we have:

$$
\begin{aligned}
f(\mathbf{x}) \in \operatorname*{argmin}_{\mathbf{u}} \left\{ \frac{1}{2}\|\mathbf{u}\|^2 - \langle \mathbf{u}, \mathcal{A}(\mathbf{x}) \rangle + \varphi(\mathbf{u}) \right\} &= \{\mathbf{u} \mid \mathcal{A}(\mathbf{x}) \in (\partial\varphi(\mathbf{u}) + \mathbf{u})\} \\
= \left\{ \mathbf{u} \mid \mathcal{A}(\mathbf{x}) \in \partial\Big(\varphi(\mathbf{u}) + \frac{1}{2}\|\mathbf{u}\|^2\Big) \right\} &= \{\mathbf{u} \mid \mathbf{u} = \nabla\psi(\mathcal{A}(\mathbf{x}))\} = \{\nabla\psi(\mathcal{A}(\mathbf{x}))\},
\end{aligned} \tag{16}
$$

Hence $f(\mathbf{x}) = \nabla\psi(\mathcal{A}(\mathbf{x}))$, and 1-smoothness of $\psi \circ \mathcal{A}(\cdot)$ implies f is nonexpansive.

(ii)⇒(i): Let $\varphi(\mathbf{x}) = \psi^*(\mathbf{x}) - \frac{1}{2}\|\mathbf{x}\|^2$. Since ψ is 1-smooth, then ψ^* is 1-strongly convex, and φ is convex. Note that:

$$
\operatorname*{argmin}_{\mathbf{u}} \left\{ \frac{1}{2}\|\mathbf{u}\|^2 - \langle \mathbf{u}, \mathcal{A}(\mathbf{x}) \rangle + \varphi(\mathbf{u}) \right\} = \{\mathbf{u} \mid \mathcal{A}(\mathbf{x}) \in (\partial\varphi(\mathbf{u}) + \mathbf{u})\} = \{\nabla\psi(\mathcal{A}(\mathbf{x}))\} = \{f(\mathbf{x})\},
$$

which means $f(\mathbf{x}) = \operatorname*{argmin}_{\mathbf{u}} \left\{ \frac{1}{2}\|\mathbf{u}\|^2 - \langle \mathbf{u}, \mathcal{A}(\mathbf{x}) \rangle + \varphi(\mathbf{u}) \right\}$. □

Now, we are ready to prove the main theorem.

Proof. Denote the right hand side of equation 15 as $F(\widetilde{\mathbf{z}})$ (For convenience of notation, we omit $\mathbf{x}, \boldsymbol{\theta}$ here), then

$$
\begin{aligned}
F(\mathbf{P}^\top\widetilde{\mathbf{z}}) = \alpha & \begin{bmatrix} \mathbf{W}_1^\top & & & & \\ & \mathbf{W}_2^\top & & & \\ & & \mathbf{W}_3^\top & & \\ & & & \ddots & \\ & & & & \mathbf{W}_L^\top \end{bmatrix} \sigma\left(\begin{bmatrix} \mathbf{W}_1 & & & & \\ & \mathbf{W}_2 & & & \\ & & \mathbf{W}_3 & & \\ & & & \ddots & \\ & & & & \mathbf{W}_L \end{bmatrix}\begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_{L-1} \\ \mathbf{z}_0 \end{bmatrix} + \right. \\
& \left. \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_{L-1} \\ \mathbf{U}_L \end{bmatrix} x + \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_{L-1} \\ \mathbf{b}_L \end{bmatrix}\right) + (1-\alpha)\begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_{L-1} \\ \mathbf{z}_0 \end{bmatrix} \\
= & \begin{bmatrix} \alpha\nabla\psi_1(\mathbf{z}_1) + (1-\alpha)\mathbf{z}_1 \\ \alpha\nabla\psi_2(\mathbf{z}_2) + (1-\alpha)\mathbf{z}_2 \\ \vdots \\ \alpha\nabla\psi_{L-1}(\mathbf{z}_{L-1}) + (1-\alpha)\mathbf{z}_{L-1} \\ \alpha\nabla\psi_L(\mathbf{z}_0) + (1-\alpha)\mathbf{z}_0 \end{bmatrix} = \nabla\left[\alpha\Psi(\widetilde{\mathbf{z}}) + (1-\alpha)\frac{1}{2}\|\widetilde{\mathbf{z}}\|^2\right],
\end{aligned}
$$

where $\nabla\psi_i(\mathbf{z}_i) := \mathbf{W}_i^\top\sigma(\mathbf{W}_i\mathbf{z}_i + \mathbf{U}_i\mathbf{x} + \mathbf{b}_i)$ and $\Psi(\widetilde{\mathbf{z}}) := \psi_1(\mathbf{z}_1) + \psi_2(\mathbf{z}_2) + \cdots + \psi_{L-1}(\mathbf{z}_{L-1}) + \psi_L(\mathbf{z}_0)$.

Given $\|\mathbf{W}_i\|_2 \leq 1, \forall i \in [1, L]$, $\nabla\psi_i(\mathbf{z}_i)$ is a nonexpansive operator. Then for $\forall\,\widetilde{\mathbf{z}}_1, \widetilde{\mathbf{z}}_2$, we have:

$$
\begin{aligned}
\|F(\mathbf{P}^\top\widetilde{\mathbf{z}}_1) - F(\mathbf{P}^\top\widetilde{\mathbf{z}}_2)\|^2 &= \sum_{i=1}^{L}\|\alpha(\nabla\psi_i(\mathbf{z}_{1,i}) - \nabla\psi_i(\mathbf{z}_{2,i})) + (1-\alpha)(\mathbf{z}_{1,i} - \mathbf{z}_{2,i})\|^2 \\
\leq \sum_{i=1}^{L}&(\alpha\|(\nabla\psi_i(\mathbf{z}_{1,i}) - \nabla\psi_i(\mathbf{z}_{2,i}))\| + (1-\alpha)\|(\mathbf{z}_{1,i} - \mathbf{z}_{2,i})\|)^2 \\
\leq \sum_{i=1}^{L}&(\alpha\|(\mathbf{z}_{1,i} - \mathbf{z}_{2,i})\| + (1-\alpha)\|(\mathbf{z}_{1,i} - \mathbf{z}_{2,i})\|)^2 = \|\widetilde{\mathbf{z}}_1 - \widetilde{\mathbf{z}}_2\|^2.
\end{aligned}
$$

By the results, we have

$$
\|F(\widetilde{\mathbf{z}}_1) - F(\widetilde{\mathbf{z}}_2)\| = \left\|F(\mathbf{P}^\top\mathbf{P}\widetilde{\mathbf{z}}_1) - F(\mathbf{P}^\top\mathbf{P}\widetilde{\mathbf{z}}_2)\right\| \leq \|\mathbf{P}\widetilde{\mathbf{z}}_1 - \mathbf{P}\widetilde{\mathbf{z}}_2\| = \|\widetilde{\mathbf{z}}_1 - \widetilde{\mathbf{z}}_1\|,
$$

which means F is nonexpansive. By Lemma 6, let $\Phi(\widetilde{\mathbf{z}}) = \left[\alpha\Psi(\cdot) + (1-\alpha)\frac{1}{2}\|\cdot\|^2\right]^*(\widetilde{\mathbf{z}}) - \frac{1}{2}\|\widetilde{\mathbf{z}}\|^2$, we have:

$$
F(\widetilde{\mathbf{z}}) \in \operatorname*{argmin}_{\mathbf{u}}\left\{\frac{1}{2}\|\mathbf{u}\|^2 - \langle\mathbf{u}, \mathbf{P}\widetilde{\mathbf{z}}\rangle + \Phi(\mathbf{u})\right\}.
$$

Hence any fixed point $\widetilde{\mathbf{z}}$ of Eq.(15) satisfies:

$$
0 \in \partial\Phi(\widetilde{\mathbf{z}}^*) + (\mathbf{I} - \mathbf{P})\widetilde{\mathbf{z}}^*,
$$

By using Thm. 4.14:

$$
\text{if}\, h(\mathbf{x}) = \alpha f(\frac{\mathbf{x}}{\alpha})\,\text{then}\, h^*(\mathbf{y}) = \alpha f^*(\mathbf{y}),
$$

Thm. 4.19:

$$
(h_1^* + h_2^*)^*(\mathbf{x}) = \inf_{\mathbf{u}}\{h_1(\mathbf{u}) + h_2(\mathbf{x} - \mathbf{u})\},
$$

in [47] and the definition of Moreau envelope:

$$
M_f^\mu(\mathbf{x}) = \inf_{\mathbf{u}}\left\{f(\mathbf{u}) + \frac{1}{2\mu}\|\mathbf{x} - \mathbf{u}\|^2\right\},
$$

$\Phi(\widetilde{\mathbf{z}})$ can be formed in terms of $\varphi_i(\mathbf{z}_i)$:

$$\begin{aligned}
\Phi(\widetilde{\mathbf{z}}) &= \left[\alpha\Psi(\cdot) + (1-\alpha)\frac{1}{2}\|\cdot\|^2\right]^* (\widetilde{\mathbf{z}}) - \frac{1}{2}\|\widetilde{\mathbf{z}}\|^2 = \sum_{i=1}^{L}\left\{\left[\alpha\psi_i(\cdot) + (1-\alpha)\frac{1}{2}\|\cdot\|^2\right]^* (\mathbf{z}_i) - \frac{1}{2}\|\mathbf{z}_i\|^2\right\} \\
&= \sum_{i=1}^{L}\left\{\left[\left(\alpha\psi_i^*(\frac{\cdot}{\alpha})\right)^* + \left((1-\alpha)\frac{1}{2}\left\|\frac{\cdot}{1-\alpha}\right\|^2\right)^*\right]^* (\mathbf{z}_i) - \frac{1}{2}\|\mathbf{z}_i\|^2\right\} \\
&= \sum_{i=1}^{L}\inf_{\mathbf{y}_i}\left\{\alpha\psi_i^*(\frac{\mathbf{y}_i}{\alpha}) + (1-\alpha)\frac{1}{2}\left\|\frac{\mathbf{y}_i - \mathbf{z}_i}{1-\alpha}\right\|^2 - \frac{1}{2}\|\mathbf{z}_i\|^2\right\} \\
&= \sum_{i=1}^{L}\inf_{\mathbf{y}_i}\left\{\alpha\psi_i^*(\mathbf{y}_i) + \frac{1}{2(1-\alpha)}\|\alpha\mathbf{y}_i - \mathbf{z}_i\|^2 - \frac{1}{2}\|\mathbf{z}_i\|^2\right\} \\
&= \sum_{i=1}^{L}\inf_{\mathbf{y}_i}\left\{\alpha\left(\varphi_i(\mathbf{y}_i) + \frac{1}{2}\|\mathbf{y}_i\|^2\right) + \frac{1}{2(1-\alpha)}\|\alpha\mathbf{y}_i - \mathbf{z}_i\|^2 - \frac{1}{2}\|\mathbf{z}_i\|^2\right\} \\
&= \sum_{i=1}^{L}\inf_{\mathbf{y}_i}\left\{\alpha\varphi_i(\mathbf{y}_i) + \frac{\alpha}{2(1-\alpha)}\|\mathbf{y}_i - \mathbf{z}_i\|^2\right\} = \sum_{i=1}^{L}\alpha M_{\varphi_i}^{1-\alpha}(\mathbf{z}_i).
\end{aligned}$$

□

B.5 Proof for Corollary 1

Proof. When $L = 2$, $\mathbf{I} - \mathbf{P}$ is a symmetric matrix

$$\begin{bmatrix} \mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & \mathbf{I} \end{bmatrix},$$

and the operator

$$\begin{bmatrix} \mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & \mathbf{I} \end{bmatrix}\begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_0 \end{bmatrix},$$

is the gradient of convex function $\frac{1}{2}\|\mathbf{z}_1 - \mathbf{z}_0\|^2$. Hence the monotone operator splitting equation Eq.(8) is now an first optimality condition of convex optimization problem:

$$\min_{\mathbf{z}_1,\mathbf{z}_0}\left\{\alpha M_{\varphi_1}^{1-\alpha}(\mathbf{z}_1) + \alpha M_{\varphi_2}^{1-\alpha}(\mathbf{z}_0) + \frac{1}{2}\|\mathbf{z}_1 - \mathbf{z}_0\|^2\right\}.$$

□

B.6 Proof for Theorem 3

For proving the results, we need a lemma from [32].

Lemma 7. *$\mathcal{H}$ is a Hilbert space, and $k < 1$. If $\mathbf{T}_n : \mathcal{H} \to \mathcal{H}$ is a k-contraction, for all $n \in \mathbb{N}^+ \cup \{0\}$, and $\mathbf{T}_n \to \mathbf{T}_0$ point-wisely. Then the fixed point of $\mathbf{T}_n$ tends to the fixed point of $\mathbf{T}_0$ when $n \to \infty$.*

Proof. Given $\operatorname{prox}_{\varphi_i}(\mathbf{z}) = \mathbf{W}_i^\top\sigma(\mathbf{W}_i\mathbf{z} + \mathbf{U}_i\mathbf{x} + \mathbf{b}_i)$, $\mathbf{z}_0^*(\alpha)$ is the fixed point of composition equation:

$$\mathbf{z} = \left[\alpha\operatorname{prox}_{\varphi_L} + (1-\alpha)\mathcal{I}\right] \circ \cdots \circ \left[\alpha\operatorname{prox}_{\varphi_1} + (1-\alpha)\mathcal{I}\right](\mathbf{z})$$

For $\alpha \in (0, 1)$, we get:

$$\begin{aligned}
\mathbf{z} =& \frac{\left[\alpha\operatorname{prox}_{\varphi_L} + (1-\alpha)\mathcal{I}\right] \circ \cdots \circ \left[\alpha\operatorname{prox}_{\varphi_1} + (1-\alpha)\mathcal{I}\right] - \mathcal{I}}{L\alpha}(\mathbf{z}) + \mathbf{z} \\
=& \left(\frac{(1-\alpha)^L - 1}{L\alpha} + 1\right)\mathbf{z} + \frac{\alpha(1-\alpha)^{L-1}}{L\alpha}\left(\operatorname{prox}_{\varphi_1} + \cdots + \operatorname{prox}_{\varphi_L}\right)(\mathbf{z}) \\
&+ \frac{\alpha^2(1-\alpha)^{L-2}}{L\alpha}\left(\sum_{p>q}\operatorname{prox}_{\varphi_p}\circ\operatorname{prox}_{\varphi_q}\right)(\mathbf{z}) + \cdots + \frac{\alpha^L}{L\alpha}\left(\prod_{p=L}^{1}\operatorname{prox}_{\varphi_p}\right)(\mathbf{z}) \\
=& \left[\frac{1}{L}\sum_{p=2}^{L}(-1)^p\binom{L}{p}\alpha^{p-1}\right]\mathbf{z} + \frac{(1-\alpha)^{L-1}}{L}\left(\operatorname{prox}_{\varphi_1} + \cdots + \operatorname{prox}_{\varphi_L}\right)(\mathbf{z}) \\
&+ \frac{\alpha(1-\alpha)^{L-2}}{L}\left(\sum_{p>q}\operatorname{prox}_{\varphi_p}\circ\operatorname{prox}_{\varphi_q}\right)(\mathbf{z}) + \cdots + \frac{\alpha^{L-1}}{L}\left(\prod_{p=L}^{1}\operatorname{prox}_{\varphi_p}\right)(\mathbf{z}),
\end{aligned} \tag{17}$$

Note that $\mathbf{z}_0^*(\alpha)$ is also the fixed point of the above equation.

Denote the right hand side of Eq.(17) as $\mathbf{T}_\alpha(\mathbf{z})$, note that $\mathbf{T}_0(\mathbf{z})$ is also well-defined now. Estimate the Lipschitz constant of $\mathbf{T}_\alpha(\mathbf{z}), \alpha \in [0,1)$:

$$\begin{aligned}\mathrm{Lip}(\mathbf{T}_\alpha) \leq & \left|\frac{1}{L}\sum_{p=2}^{L}(-1)^p\binom{L}{p}\alpha^{p-1}\right| + \left|\frac{(1-\alpha)^{L-1}}{L}\right|\left(\|\mathbf{W}_1\|_2^2 + \cdots + \|\mathbf{W}_L\|_2^2\right) \\ & + \left|\frac{\alpha(1-\alpha)^{L-2}}{L}\right|\left|\sum_{p>q}\|\mathbf{W}_p\|_2^2\|\mathbf{W}_q\|_2^2\right| + \cdots + \left|\frac{\alpha^{L-1}}{L}\right|\left(\prod_{p=L}^{1}\|\mathbf{W}_p\|_2^2\right).\end{aligned}$$

Each terms in the right hand side of the above inequality (except the second term) is a polynomial of α with non-zero order for α, hence they tend to zero when $\alpha \to 0$. Note that $(1-\alpha)^L \to (1-L\alpha)$, when $\alpha \to 0$. Thus, the second term tend to $\frac{1}{L}\left(\|\mathbf{W}_1\|_2^2 + \cdots + \|\mathbf{W}_L\|_2^2\right)$, which is less than 1 by assumption. Hence there is a $\kappa \in (0,1)$, when $\alpha \in [0,\kappa]$, $\mathrm{Lip}(\mathbf{T}_\alpha) < \kappa$.

By using Lemma 7, the fixed point of $\mathbf{T}_\alpha$ (i.e. $\mathbf{z}_0^*(\alpha)$) tends to the fixed point of $\mathbf{T}_0$, i.e.

$$\mathbf{y}^* = \frac{\mathrm{prox}_{\varphi_1}(\mathbf{y}^*) + \cdots + \mathrm{prox}_{\varphi_L}(\mathbf{y}^*)}{L}.$$

Using first order optimality condition, $\left(\mathrm{prox}_{\varphi_1}(\mathbf{y}^*), \cdots, \mathrm{prox}_{\varphi_L}(\mathbf{y}^*), \mathbf{y}^*\right)$ is the minimizer of the following strongly convex problem:

$$\sum_{l=1}^{L}\left(\varphi_l(\mathbf{x}_l) + \frac{1}{2}\|\mathbf{x}_l - \mathbf{y}\|^2\right).$$

Finally, let $\mathbf{z}_1^*(\alpha)$ be the fixed point of the composition equation:

$$\mathbf{z} = \left[\alpha\,\mathrm{prox}_{\varphi_1} + (1-\alpha)\mathcal{I}\right] \circ \left[\alpha\,\mathrm{prox}_{\varphi_L} + (1-\alpha)\mathcal{I}\right] \circ \cdots \circ \left[\alpha\,\mathrm{prox}_{\varphi_2} + (1-\alpha)\mathcal{I}\right](\mathbf{z}).$$

A similar argument can be applied to $\mathbf{z}_1^*(\alpha)$ to show that, when $\alpha \to 0$, $\mathbf{z}_1^*(\alpha)$ tends to the same $\mathbf{y}^*$ defined above, and so do $\mathbf{z}_2^*(\alpha), \cdots, \mathbf{z}_{L-1}^*(\alpha)$.

□

B.7 Proof of Theorem 4

Proof. Since $f(\mathbf{z}) = \mathrm{prox}_\varphi(\mathbf{z})$, we have

$$\mathbf{z} = f(\mathbf{y}) = \mathrm{prox}_\varphi(\mathbf{y}) = (I + \partial\varphi)^{-1}(\mathbf{y}) \Leftrightarrow \mathbf{y} \in \mathbf{z} + \partial\varphi(\mathbf{z}).$$

Therefore,

$$\mathbf{z}^* = f(\mathbf{z}^* - \gamma\frac{\partial\mathcal{R}_z(\mathbf{z}^*)}{\partial\mathbf{z}}) \Leftrightarrow \mathbf{z}^* - \gamma\frac{\partial\mathcal{R}_z(\mathbf{z}^*)}{\partial\mathbf{z}} \in \mathbf{z}^* + \partial\varphi(\mathbf{z}^*) \Leftrightarrow 0 \in \partial\varphi(\mathbf{z}^*) + \gamma\frac{\partial\mathcal{R}_z(\mathbf{z}^*)}{\partial\mathbf{z}},$$

which means $\mathbf{z}^*$ is a minimizer of $f(\mathbf{z}) + \gamma\mathcal{R}_z(\mathbf{z})$. □

Appendix C
Proofs for the Convergence of SAM

C.1 Auxiliary Lemmas

Next lemma follows Lem. 2.5 in [48].

Lemma 8. *If* $a_1 \geq 0, 0 < t_1 < 1, t_2 > 0, 0 < r_1 < 1, r_2 > 0$, $\{a_n\}$ *is a sequence of non-negative numbers satisfying*

$$a_{k+1} = \left(1 - \frac{r_1}{(k+1)^{t_1}}\right)a_k + \frac{r_1}{(k+1)^{t_1}}\frac{r_2}{k^{t_2}}.$$

Then $\lim_{n\to\infty} a_n = 0$, *and there exists* $B = \mathcal{O}(r_1, r_2, t_1, t_2, a_1)$ *such that* $\|a_k\| \leq B$.

Proof. Denote $b_k = \frac{r_1}{(k+1)^{t_1}}, c_k = \frac{r_2}{k^{t_2}}$. Since $0 < t_1 < 1, 0 < r_1 < 1$, we have $0 < b_k < 1, \sum_{k=1}^{\infty} b_k = \infty$ and:

$$\begin{aligned}\prod_{k=1}^{\infty}(1-b_k) &= \exp\left(\sum_{k=1}^{\infty}\ln(1-b_k)\right) = \lim_{K\to\infty}\exp\left(\sum_{k=1}^{K}\ln(1-b_k)\right) \\ &\leq \limsup_{K\to\infty}\exp\left(\sum_{k=1}^{K}-b_k\right) = 0.\end{aligned}$$

For any $\epsilon > 0$, choose N big enough such that $c_k \leq \epsilon, \forall k \geq N$, then for all $k > N$, by induction, we have,

$$\begin{aligned}
a_{k+1} &= (1-b_k)a_k + b_k c_k \\
&= (1-b_k)(1-b_{k-1})a_{k-1} + (1-b_k)b_{k-1}c_{k-1} + b_k c_k = \cdots \\
&= \prod_{j=N}^{k}(1-b_j)a_N + \sum_{i=N}^{k}\left(\prod_{j=i+1}^{k}(1-b_j)\right)b_i c_i \leq \prod_{j=N}^{k}(1-b_j)a_N + \sum_{i=N}^{k}\left(\prod_{j=i+1}^{k}(1-b_j)\right)b_i\epsilon \\
&= \prod_{j=N}^{k}(1-b_j)a_N + \left(1-\prod_{j=N}^{k}(1-b_j)\right)\epsilon \leq \prod_{j=N}^{k}(1-b_j)a_N + \epsilon.
\end{aligned}$$

Hence $\limsup_{k\to\infty} a_k \leq \epsilon$, let $\epsilon \to 0^+$, we get $\lim_{n\to\infty} a_n = 0$. Since every convergence sequence is bounded, there exists $B = \mathcal{O}(r_1, r_2, t_1, t_2, a_1)$ such that: $\|a_k\| \leq B$. □

Next lemma follows from Prop. 3 in [11].

Lemma 9. *$l(\mathbf{x})$ is a L_z-smooth convex function (i.e. $\nabla l(\mathbf{x})$ is L_z-Lipschitz), and $\gamma = \frac{1}{2L_z}, 0 < \lambda \leq \frac{L_z}{2}$. Then the operator $\mathcal{S}_\lambda(\mathbf{x}) = \mathbf{x} - \gamma(\nabla l(\mathbf{x}) + \lambda\mathbf{x})$ satisfies $\frac{1}{4}\|\mathbf{x}-\mathbf{y}\| \leq \|\mathcal{S}_\lambda(\mathbf{x}) - \mathcal{S}_\lambda(\mathbf{y})\| \leq (1-\frac{\gamma\lambda}{2})\|\mathbf{x}-\mathbf{y}\|$.*

Proof. Note that $l(\mathbf{x}) + \frac{\lambda}{2}\|\mathbf{x}\|^2$ is λ-strongly convex and $(L_z+\lambda)$-smooth. By Prop. 3 in [11], it follows that:

$$\|\mathcal{S}_\lambda(\mathbf{x}) - \mathcal{S}_\lambda(\mathbf{y})\| \leq \sqrt{1 - \frac{2\gamma\lambda(L_z+\lambda)}{L_z+2\lambda}}\|\mathbf{x}-\mathbf{y}\|.$$

Note that $\sqrt{1-\frac{2\gamma\lambda(L_W+\lambda)}{L_W+2\lambda}} < \sqrt{1-\gamma\lambda} < 1-\frac{\gamma\lambda}{2}$, so the operator $\mathcal{S}_\lambda(\mathbf{x})$ is $(1-\frac{\gamma\lambda}{2})$-contractive.

On the other hand, by Cauchy–Schwarz inequality, $\langle(\nabla l(\mathbf{x})+\lambda\mathbf{x}) - (\nabla l(\mathbf{y})+\lambda\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle \leq \frac{2\gamma}{3}\|(\nabla l(\mathbf{x})+\lambda\mathbf{x}) - (\nabla l(\mathbf{y})+\lambda\mathbf{y})\|^2 + \frac{3}{8\gamma}\|\mathbf{x}-\mathbf{y}\|^2$, and note that $\|\mathcal{S}_\lambda(\mathbf{x}) - \mathcal{S}_\lambda(\mathbf{y})\|^2 = \|\mathbf{x}-\mathbf{y}\|^2 - 2\gamma\langle(\nabla l(\mathbf{x})+\lambda\mathbf{x}) - (\nabla l(\mathbf{y})+\lambda\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle + \gamma^2\|(\nabla l(\mathbf{x})+\lambda\mathbf{x}) - (\nabla l(\mathbf{y})+\lambda\mathbf{y})\|^2$, we have:

$$\begin{aligned}
\|\mathcal{S}_\lambda(\mathbf{x}) - \mathcal{S}_\lambda(\mathbf{y})\|^2 &\geq \frac{1}{4}\|\mathbf{x}-\mathbf{y}\|^2 - \frac{\gamma^2}{3}\|(\nabla l(\mathbf{x})+\lambda\mathbf{x}) - (\nabla l(\mathbf{y})+\lambda\mathbf{y})\|^2 \\
&\geq \frac{1}{4}\|\mathbf{x}-\mathbf{y}\|^2 - \frac{\gamma^2}{3}(L_z+\lambda)^2\|\mathbf{x}-\mathbf{y}\|^2 \\
&\geq \frac{1}{4}\|\mathbf{x}-\mathbf{y}\|^2 - \frac{\gamma^2}{3}(\frac{3L_z}{2})^2\|\mathbf{x}-\mathbf{y}\|^2 \\
&\geq (\frac{1}{4} - \frac{3}{16})\|\mathbf{x}-\mathbf{y}\|^2 \\
&= \frac{1}{16}\|\mathbf{x}-\mathbf{y}\|^2,
\end{aligned}$$

which is equivalent to $\|\mathcal{S}_\lambda(\mathbf{x}) - \mathcal{S}_\lambda(\mathbf{y})\| \geq \frac{1}{4}\|\mathbf{x}-\mathbf{y}\|$. □

Lemma 10. *If $\mathcal{T}(\mathbf{z})$ is a nonexpansive map with $\operatorname{dom}(\mathcal{T}) = \mathbb{R}^n$, then the fixed point set of $\mathcal{T}(\mathbf{z})$ is closed and convex. And for any $\mathbf{x}, \mathbf{y}$, $\langle(\mathbf{x}-\mathcal{T}(\mathbf{x})) - (\mathbf{y}-\mathcal{T}(\mathbf{y})), \mathbf{x}-\mathbf{y}\rangle \geq 0$.*

Proof. If Fix($\mathcal{T}$) is empty, then it is closed and convex. If Fix($\mathcal{T}$) is non-empty, for any $\mathbf{x}, \mathbf{y} \in$ Fix($\mathcal{T}$), let $\mathbf{z} = \theta\mathbf{x} + (1-\theta)\mathbf{y}$, where $\theta \in (0,1)$. Since $\mathcal{T}$ is nonexpansive, we have:

$$\begin{cases} \|\mathcal{T}(\mathbf{z}) - \mathbf{x}\| = \|\mathcal{T}(\mathbf{z}) - \mathcal{T}(\mathbf{x})\| \leq \|\mathbf{z}-\mathbf{x}\| = (1-\theta)\|\mathbf{x}-\mathbf{y}\| \\ \|\mathcal{T}(\mathbf{z}) - \mathbf{y}\| = \|\mathcal{T}(\mathbf{z}) - \mathcal{T}(\mathbf{y})\| \leq \|\mathbf{z}-\mathbf{y}\| = \theta\|\mathbf{x}-\mathbf{y}\|. \end{cases}$$

So the triangle inequality,

$$\|\mathbf{x}-\mathbf{y}\| \leq \|\mathcal{T}(\mathbf{z}) - \mathbf{x}\| + \|\mathcal{T}(\mathbf{z}) - \mathbf{y}\| \leq (1-\theta)\|\mathbf{x}-\mathbf{y}\| + \theta\|\mathbf{x}-\mathbf{y}\| = \|\mathbf{x}-\mathbf{y}\|,$$

holds with equality. So $\mathcal{T}(\mathbf{z})$ lies on the line segment between $\mathbf{x}, \mathbf{y}$, and $\|\mathcal{T}(\mathbf{z}) - \mathbf{y}\| = \theta\|\mathbf{x}-\mathbf{y}\|$, $\|\mathcal{T}(\mathbf{z}) - \mathbf{x}\| = (1-\theta)\|\mathbf{x}-\mathbf{y}\|$ hold, which means $\mathcal{T}(\mathbf{z}) = \theta\mathbf{x} + (1-\theta)\mathbf{y} = \mathbf{z}$, i.e., $\mathbf{z} \in$ Fix($\mathcal{T}$).

The inequality $\langle(\mathbf{x}-\mathcal{T}(\mathbf{x})) - (\mathbf{y}-\mathcal{T}(\mathbf{y})), \mathbf{x}-\mathbf{y}\rangle \geq 0$ follows directly form Cauchy–Schwarz inequality. □

C.2 Proof for Theorem 5

Proof. Since $\lambda_k \le \frac{L_z}{2}$, by Lemma 9, operator $\mathbf{z} \mapsto \mathbf{z} - \gamma(\nabla\mathcal{R}_z(\mathbf{z}) + \lambda_k \mathbf{z})$ is contractive, so is the operator $\mathbf{z} \mapsto \beta_k(\mathbf{z} - \gamma(\nabla\mathcal{R}_z(\mathbf{z}) + \lambda_k\mathbf{z})) + (1-\beta_k)\mathcal{T}(\mathbf{z})$, hence the latter operator has a unique fixed point $\bar{\mathbf{z}}^k$.

By assumption, $\|\bar{\mathbf{z}}^{k+1}\| \le B_1^*$. Note that in our setting, $\nabla\mathcal{R}_z(\cdot)$ and $\mathcal{T}(\cdot)$ are continuous, so we have $\|\mathcal{T}(\bar{\mathbf{z}}^{k+1})\| + \|\bar{\mathbf{z}}^{k+1} - \gamma\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k+1})\| \le B_2^*$. First we estimate the difference of two successive fixed point:

$$\begin{aligned}
&\|\bar{\mathbf{z}}^k - \bar{\mathbf{z}}^{k+1}\| \\
=&\Big\|\beta_k\Big(\bar{\mathbf{z}}^k - \gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^k) + \lambda_k\bar{\mathbf{z}}^k)\Big) + (1-\beta_k)\mathcal{T}(\bar{\mathbf{z}}^k) \\
&-\beta_{k+1}\Big(\bar{\mathbf{z}}^{k+1} - \gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k+1}) + \lambda_{k+1}\bar{\mathbf{z}}^{k+1})\Big) - (1-\beta_{k+1})\mathcal{T}(\bar{\mathbf{z}}^{k+1})\Big\| \\
\le&\Big\|\beta_k\Big(\bar{\mathbf{z}}^k - \gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^k) + \lambda_k\bar{\mathbf{z}}^k)\Big) - \beta_k\Big(\bar{\mathbf{z}}^{k+1} - \gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k+1}) + \lambda_k\bar{\mathbf{z}}^{k+1})\Big)\Big\| \\
+&\Big\|\beta_k\Big(\bar{\mathbf{z}}^{k+1} - \gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k+1}) + \lambda_k\bar{\mathbf{z}}^{k+1})\Big) - \beta_{k+1}\Big(\bar{\mathbf{z}}^{k+1} - \gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k+1}) + \lambda_{k+1}\bar{\mathbf{z}}^{k+1})\Big)\Big\| \\
+&\Big\|(1-\beta_k)(\mathcal{T}(\bar{\mathbf{z}}^k) - \mathcal{T}(\bar{\mathbf{z}}^{k+1}))\Big\| + \Big\|(\beta_{k+1} - \beta_k)\mathcal{T}(\bar{\mathbf{z}}^{k+1})\Big\| \\
\le&\beta_k(1 - \frac{\gamma\lambda_k}{2})\Big\|\bar{\mathbf{z}}^k - \bar{\mathbf{z}}^{k+1}\Big\| + (1-\beta_k)\Big\|\bar{\mathbf{z}}^k - \bar{\mathbf{z}}^{k+1}\Big\| + B_2^*|\beta_k - \beta_{k+1}| + B_1^*|\beta_k\lambda_k - \beta_{k+1}\lambda_{k+1}| \\
=&(1 - \frac{\gamma\beta_k\lambda_k}{2})\|\bar{\mathbf{z}}^k - \bar{\mathbf{z}}^{k+1}\| + B_2^*|\beta_k - \beta_{k+1}| + B_1^*|\beta_k\lambda_k - \beta_{k+1}\lambda_{k+1}|.
\end{aligned}$$

We get:

$$\begin{aligned}
\frac{\|\bar{\mathbf{z}}^k - \bar{\mathbf{z}}^{k+1}\|}{\frac{\gamma}{2}\beta_{k+1}\lambda_{k+1}} \le& \frac{4B_2^*}{\gamma^2}\frac{|\beta_k - \beta_{k+1}|}{(\beta_k\lambda_k)(\beta_{k+1}\lambda_{k+1})} + \frac{4B_1^*}{\gamma^2}\frac{|\beta_k\lambda_k - \beta_{k+1}\lambda_{k+1}|}{(\beta_k\lambda_k)(\beta_{k+1}\lambda_{k+1})} \\
=&\frac{4B_2^*}{\gamma^2\eta^3}\frac{|k^{-\rho} - (k+1)^{-\rho}|}{k^{-\rho-c}(k+1)^{-\rho-c}} + \frac{4B_1^*}{\gamma^2\eta^2}\frac{|k^{-\rho-c} - (k+1)^{-\rho-c}|}{k^{-\rho-c}(k+1)^{-\rho-c}} \\
\le&\frac{4B_2^*\rho}{\gamma^2\eta^3}k^{-\rho-1}k^{\rho+c}(k+1)^{\rho+c} + \frac{4B_1^*(\rho+c)}{\gamma^2\eta^2}k^{-\rho-c-1}k^{\rho+c}(k+1)^{\rho+c} \\
\le&\frac{4B_2^*\rho 2^{\rho+c}}{\gamma^2\eta^3}k^{\rho+2c-1} + \frac{4B_1^*(\rho+c)2^{\rho+c}}{\gamma^2\eta^2}k^{\rho+c-1} \\
\le&\left[\frac{4B_2^*\rho 2^{\rho+c}}{\gamma^2\eta^3} + \frac{4B_1^*(\rho+c)2^{\rho+c}}{\gamma^2\eta^2}\right]k^{\rho+2c-1} := B_3^* k^{\rho+2c-1}.
\end{aligned}$$

By the iteration equation:

$$\mathbf{z}^{k+1} = \beta_{k+1}\Big(\mathbf{z}^k - \gamma(\nabla\mathcal{R}_z(\mathbf{z}^k) + \lambda_{k+1}\mathbf{z}^k)\Big) + (1-\beta_{k+1})\mathcal{T}(\mathbf{z}^k),$$

we have:

$$\begin{aligned}
&\|\mathbf{z}^{k+1} - \bar{\mathbf{z}}^{k+1}\| \\
=&\Big\|\Big[\beta_{k+1}(\mathbf{z}^k - \gamma(\nabla\mathcal{R}_z(\mathbf{z}^k) + \lambda_{k+1}\mathbf{z}^k)) + (1-\beta_{k+1})\mathcal{T}(\mathbf{z}^k)\Big] \\
&-\Big[\beta_{k+1}(\bar{\mathbf{z}}^{k+1} - \gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k+1}) + \lambda_{k+1}\bar{\mathbf{z}}^{k+1})) + (1-\beta_{k+1})\mathcal{T}(\bar{\mathbf{z}}^{k+1})\Big]\Big\| \\
\le&\beta_{k+1}\Big\|(\mathbf{z}^k - \gamma(\nabla\mathcal{R}_z(\mathbf{z}^k) + \lambda_{k+1}\mathbf{z}^k)) - (\bar{\mathbf{z}}^{k+1} - \gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k+1}) + \lambda_{k+1}\bar{\mathbf{z}}^{k+1}))\Big\| + (1-\beta_{k+1})\|\mathcal{T}(\mathbf{z}^k) - \mathcal{T}(\bar{\mathbf{z}}^{k+1})\| \\
\le&\beta_{k+1}(1 - \frac{\gamma\lambda_{k+1}}{2})\|\mathbf{z}^k - \bar{\mathbf{z}}^{k+1}\| + (1-\beta_{k+1})\|\mathbf{z}^k - \bar{\mathbf{z}}^{k+1}\| \\
\le&(1 - \frac{\gamma}{2}\beta_{k+1}\lambda_{k+1})\Big(\|\mathbf{z}^k - \bar{\mathbf{z}}^k\| + \|\bar{\mathbf{z}}^k - \bar{\mathbf{z}}^{k+1}\|\Big) \\
\le&(1 - \frac{\gamma}{2}\beta_{k+1}\lambda_{k+1})\|\mathbf{z}^k - \bar{\mathbf{z}}^k\| + \frac{\gamma}{2}\beta_{k+1}\lambda_{k+1}B_3^* k^{\rho+2c-1}.
\end{aligned}$$

Since $\eta \le \sqrt{2L_z}$, so $\gamma\eta^2 \le 1$. Let $r_1 = \frac{\gamma\eta^2}{2}, r_2 = B_3^*, t_1 = \rho + c \in (0,1), t_2 = 1 - \rho - 2c > 0$, and $a_k = \|\mathbf{z}^k - \bar{\mathbf{z}}^k\|$, by Lemma 8 we have: $\|\mathbf{z}^k - \bar{\mathbf{z}}^k\| \to 0$ as $k \to \infty$ and $\|\mathbf{z}^k - \bar{\mathbf{z}}^k\| \le B_4^* = \mathcal{O}\big(\mathbf{z}^1, \rho, c, B_3^*\big)$.

Next, we denote the only minimizer of convex function $\mathcal{R}_z(\mathbf{z})$ on compact convex set $\mathrm{Fix}(\mathcal{T})$ as $\bar{\mathbf{z}}$, which is bounded by B_1^*. Note that the convexity follows from Lemma 10. We now prove the final results by contradiction. Since the sequence $\{\bar{\mathbf{z}}^k\}$ is bounded by B_1^*, if it does not converge to $\bar{\mathbf{z}}$, there exists $\delta > 0$ and a sub-convergent $\{\bar{\mathbf{z}}^{k_j}\}$ such that:

$$\|\bar{\mathbf{z}}^{k_j} - \bar{\mathbf{z}}\| \ge \delta, \quad \forall j \in \mathbb{N}^+,$$

and

$$\bar{\mathbf{z}}^{k_j} \to \bar{\mathbf{z}}', \quad \|\bar{\mathbf{z}}' - \bar{\mathbf{z}}\| \ge \delta.$$

By the fixed point equation and Lemma 10, we have

$$\begin{aligned}
&\bar{\mathbf{z}}^{k_j} = \beta_{k_j}\Big(\bar{\mathbf{z}}^{k_j} - \gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k_j}) + \lambda_{k_j}\bar{\mathbf{z}}^{k_j})\Big) + (1-\beta_{k_j})\mathcal{T}(\bar{\mathbf{z}}^{k_j})\\
\Rightarrow&\gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k_j}) + \lambda_{k_j}\bar{\mathbf{z}}^{k_j}) = \frac{1-\beta_{k_j}}{\beta_{k_j}}(\mathcal{T}(\bar{\mathbf{z}}^{k_j}) - \bar{\mathbf{z}}^{k_j})\\
\Rightarrow&\Big\langle\gamma(\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k_j}) + \lambda_{k_j}\bar{\mathbf{z}}^{k_j}), \mathbf{z} - \bar{\mathbf{z}}^{k_j}\Big\rangle = \frac{1-\beta_{k_j}}{\beta_{k_j}}\langle\mathcal{T}(\bar{\mathbf{z}}^{k_j}) - \bar{\mathbf{z}}^{k_j}, \mathbf{z} - \bar{\mathbf{z}}^{k_j}\rangle\\
&= \frac{1-\beta_{k_j}}{\beta_{k_j}}\Big\langle\mathcal{T}(\bar{\mathbf{z}}^{k_j}) - \bar{\mathbf{z}}^{k_j} - (\mathcal{T}(\mathbf{z}) - \mathbf{z}), \mathbf{z} - \bar{\mathbf{z}}^{k_j}\Big\rangle \geq 0, \quad \forall\mathbf{z}\in\mathrm{Fix}(\mathcal{T})\\
\Rightarrow&\langle\nabla\mathcal{R}_z(\bar{\mathbf{z}}^{k_j}) + \lambda_{k_j}\bar{\mathbf{z}}^{k_j}, \mathbf{z} - \bar{\mathbf{z}}^{k_j}\rangle \geq 0, \quad \forall\mathbf{z}\in\mathrm{Fix}(\mathcal{T}).
\end{aligned}$$

Let $j\to\infty$, we get:

$$\langle\nabla\mathcal{R}_z(\bar{\mathbf{z}}'), \mathbf{z} - \bar{\mathbf{z}}'\rangle \geq 0, \quad \forall\mathbf{z}\in\mathrm{Fix}(\mathcal{T}),$$

which is equivalent to $\mathcal{R}_z(\bar{\mathbf{z}}') \leq \mathcal{R}_z(\mathbf{z}), \forall\mathbf{z}\in\mathrm{Fix}(\mathcal{T})$. Namely $\bar{\mathbf{z}}' = \bar{\mathbf{z}}$, which is a contradiction.

Thus, the sequence $\{\bar{\mathbf{z}}^k\}$ converge to $\bar{\mathbf{z}}$. Combining the result $\|\mathbf{z}^k - \bar{\mathbf{z}}^k\|\to 0$, we have:

$$\mathbf{z}^k \to \bar{\mathbf{z}}.$$

We finish the proof. □

Appendix D
Proofs for Linear Convergence Training

In the proof, we will consider the whole data set, i.e., $\mathbf{Z}^K \in \mathbb{R}^{m\times N}$, where N is the training data size. We denote by $\mathbf{z}^k := \mathrm{vec}(\mathbf{Z}^k)$, where $\mathrm{vec}(\mathbf{Z}) \in \mathbb{R}^{mN}$ is the vectorization of the matrix $\mathbf{Z}\in\mathbb{R}^{m\times N}$. $\mathbf{Z}^k$ is the k-th iterates of the sequence generated by Eq.(11) applied on the data matrix. Then, we denote:

$$\mathbf{Z}^{(k,l)} := f_l \circ\cdots\circ f_1(\mathbf{Z}^k, \mathbf{Z}, \boldsymbol{\theta}), \quad \text{and} \quad \mathbf{z}^{(k,l)} := \mathrm{vec}\left(\mathbf{Z}^{(k,l)}\right).$$

$\mathbf{z}^{(k,l)}$ and $\mathbf{z}^k$ all depend on the parameter $\boldsymbol{\theta}$. However, for the sake of brevity, we omit the mark $\boldsymbol{\theta}$ when the meaning of the symbol is clear. We let $\mathbf{I}_n$ be the $n\times n$ identity matrix. For the output of the network, we let $\mathbf{y} := \mathrm{vec}(\mathbf{W}_{L+1}\mathbf{Z}^K)$.

Note that we have

$$\mathcal{S}_{\lambda_k}(\mathbf{Z}, \mathbf{W}_{L+1}) = \mathbf{Z} - \gamma(\nabla\mathcal{R}_z(\mathbf{Z}) + \lambda_k\mathbf{Z}), \quad \text{and} \quad \gamma < \frac{1}{2L_z+1}.$$

And we let $\mathbf{s}^k := \mathrm{vec}\left(\mathcal{S}(\mathbf{Z}^k, \mathbf{W}_{L+1})\right)$, then:

$$\mathbf{z}^{k+1} = \beta_{k+1}\,\mathrm{vec}\left(\mathcal{S}_{\lambda_k}(\mathbf{Z}^k, \mathbf{W}_{L+1})\right) + (1-\beta_{k+1})\mathcal{T}(\mathbf{z}^k, \mathbf{x}, \boldsymbol{\theta}) = \beta_{k+1}\mathbf{s}^k + (1-\beta_{k+1})\mathbf{z}^{(k,L)}.$$

We let:

$$\mathcal{V}_\gamma(\cdot) := (1-\gamma\lambda_k)\mathcal{I}(\cdot) - \gamma\nabla\mathcal{R}_z(\cdot), \quad \text{and denote by} \quad \mathbf{G}^{(k,l)} := \sigma\left(\mathbf{W}_l\mathbf{Z}^{(k,l-1)} + \mathbf{U}_l\mathbf{X} + \mathbf{b}_l\right).$$

Note that by our setting on γ and λ_k, we can easily conclude that $\|\mathcal{V}_\gamma\|_2 < 1$. Moreover, due to Lemma 9, we also get the lower bound: $\|\mathcal{V}_\gamma\|_2 \geq \frac{1}{4}$.

We also denote:

$$\mathbf{D}^{(k,l)} := \begin{bmatrix}\widetilde{\mathbf{D}}_1 & & \\ & \ddots & \\ & & \widetilde{\mathbf{D}}_N\end{bmatrix}, \quad \widetilde{\mathbf{D}}_j := \begin{bmatrix}\mathbf{d}_{1j} & & \\ & \ddots & \\ & & \mathbf{d}_{mj}\end{bmatrix},$$

where

$$d_{ij} = \sigma'\left(\mathbf{W}_l\mathbf{z}_j^k + \mathbf{U}_l\mathbf{x} + \mathbf{b}_l\right)_i,$$

is the (i,j)-th entry of the derivative matrix

$$\sigma'\left(\mathbf{W}_l\mathbf{Z}^{(k,l)} + \mathbf{U}_l\mathbf{X} + \mathbf{b}_l\right) \in \mathbb{R}^{m\times N}.$$

We let:

$$\mathbf{A}(k, l_2, l_1) := \prod_{l=l_1}^{l_2}\left(\alpha\left(\mathbf{I}_N\otimes\mathbf{W}_l^\top\right)\mathbf{D}^{(k,l)}(\mathbf{I}_N\otimes\mathbf{W}_l) + (1-\alpha)\mathbf{I}_{mN}\right),$$

where $\otimes$ is the Kronecker product.

For Theorem 6, we consider the square loss, i.e.,

$$\ell(\widetilde{\boldsymbol{\theta}}) = \ell(\mathbf{y}, \mathbf{y}^0) = \frac{1}{2}\|\mathbf{y} - \mathbf{y}_0\|^2.$$

D.1 Auxiliary Lemmas

We first offer several auxiliary lemmas.

Lemma 11. *The following results hold:*

$$\begin{cases} \dfrac{\partial \mathbf{z}^{k+1}}{\partial \operatorname{vec}(\mathbf{W}_l)} = \displaystyle\sum_{\widetilde{k}=1}^{k} \left(\left(\prod_{q=\widetilde{k}+1}^{k} (\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{q+1})\mathbf{A}(q,1)) \right) \mathbf{A}(\widetilde{k}, l+1) \dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(\widetilde{k},l-1)}, \mathbf{X}) \right)}{\partial \operatorname{vec}(\mathbf{W}_l)} \right), \\ \dfrac{\partial \mathbf{z}^{k+1}}{\partial \operatorname{vec}(\mathbf{U}_l)} = \displaystyle\sum_{\widetilde{k}=1}^{k} \left(\left(\prod_{q=\widetilde{k}+1}^{k} (\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{q+1})\mathbf{A}(q,1)) \right) \mathbf{A}(\widetilde{k}, l+1) \dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(\widetilde{k},l-1)}, \mathbf{X}) \right)}{\partial \operatorname{vec}(\mathbf{U}_l)} \right), \\ \dfrac{\partial \mathbf{z}^{k+1}}{\partial \operatorname{vec}(\mathbf{b}_l)} = \displaystyle\sum_{\widetilde{k}=1}^{k} \left(\left(\prod_{q=\widetilde{k}+1}^{k} (\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{q+1})\mathbf{A}(q,1)) \right) \mathbf{A}(\widetilde{k}, l+1) \dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(\widetilde{k},l-1)}, \mathbf{X}) \right)}{\partial \operatorname{vec}(\mathbf{b}_l)} \right), \end{cases} \tag{18}$$

where

$$\begin{cases} \dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X}) \right)}{\partial \operatorname{vec}(\mathbf{W}_l)} = \alpha\left(\left(\mathbf{G}^{(k,l)} \right)^\top \otimes \mathbf{I}_m \right) \mathbf{K}^{(n_l,m)} + \alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}^{(k,l)} \left(\left(\mathbf{Z}^{(k,l)} \right)^\top \otimes \mathbf{I}_{n_l} \right), \\ \dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X}) \right)}{\partial \operatorname{vec}(\mathbf{U}_l)} = \alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}^{(k,l)} \left(\mathbf{X}^\top \otimes \mathbf{I}_{n_l} \right), \quad \dfrac{\partial \operatorname{vec}(f_l(\mathbf{Z}, \mathbf{X}))}{\partial \operatorname{vec}(\mathbf{b}_l)} = \alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}^{(k,l)} \mathbf{1}_N, \end{cases} \tag{19}$$

here $\mathbf{1}_N$ is a N-dimensional all-one vector.

Proof. Note that $f_l(\mathbf{Z}^{(k,l)}, \mathbf{X}) = \alpha \mathbf{W}_l^\top \sigma\left(\mathbf{W}_l \mathbf{Z}^{(k,l)} + \mathbf{U}_l \mathbf{X} + \mathbf{b}_l \right) + (1-\alpha)\mathbf{Z}^{(k,l)}$. Hence, we have:

$$\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X}) \right)}{\partial \mathbf{z}^{(k,l)}} = \alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}^{(k,l)} (\mathbf{I}_N \otimes \mathbf{W}_l) + (1-\alpha)\mathbf{I}_{mN},$$

where $\otimes$ is the Kronecker product. Then, we can get:

$$\frac{\partial \mathcal{T}(\mathbf{z}^k, \mathbf{x}, \boldsymbol{\theta})}{\partial \mathbf{z}^k} = \prod_{l=1}^{L} \left(\alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}^{(k,l)} (\mathbf{I}_N \otimes \mathbf{W}_l) + (1-\alpha)\mathbf{I}_{mN} \right),$$

and

$$\begin{aligned} \frac{\partial \mathbf{z}^{k+1}}{\partial \mathbf{z}^k} =& \beta_{k+1} \frac{\partial \mathcal{S}(\mathbf{z}^k, \mathbf{W}_{L+1})}{\partial \mathbf{z}^k} + (1-\beta_{k+1}) \frac{\partial \mathcal{T}(\mathbf{z}^k, \mathbf{x}, \boldsymbol{\theta})}{\partial \mathbf{z}^k} \\ =& \beta_{k+1} \left(\mathbf{I}_N \otimes \left(\mathbf{I}_{d_y} - \gamma \mathbf{W}_{L+1}^\top \mathbf{W}_{L+1} \right) \right) + (1-\beta_{k+1}) \frac{\partial \mathcal{T}(\mathbf{z}^k, \mathbf{x}, \boldsymbol{\theta})}{\partial \mathbf{z}^k} \\ =& \beta_{k+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{k+1}) \prod_{l=1}^{L} \left(\alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}_l^k (\mathbf{I}_N \otimes \mathbf{W}_l) + (1-\alpha)\mathbf{I}_{mN} \right) \\ =& \beta_{k+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{k+1})\mathbf{A}(k, L, 1). \end{aligned} \tag{20}$$

On the other hand, we have:

$$\begin{aligned} & \frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X}) \right)}{\partial \operatorname{vec}(\mathbf{W}_l)} \\ =& \alpha\left(\sigma\left(\mathbf{W}_l \mathbf{Z}^{(k,l)} + \mathbf{U}_l \mathbf{X} + \mathbf{b}_l \right)^\top \otimes \mathbf{I}_m \right) \mathbf{K}^{(n_l,m)} + \alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}^{(k,l)} \left(\left(\mathbf{Z}^{(k,l)} \right)^\top \otimes \mathbf{I}_{n_l} \right) \\ =& \alpha\left(\left(\mathbf{G}^{(k,l)} \right)^\top \otimes \mathbf{I}_m \right) \mathbf{K}^{(n_l,m)} + \alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}^{(k,l)} \left(\left(\mathbf{Z}^{(k,l)} \right)^\top \otimes \mathbf{I}_{n_l} \right), \end{aligned}$$

where $\mathbf{K}^{(n_l,m)} \in \mathbb{R}^{n_l m \times n_l m}$ is the commutation matrix such that $\mathbf{K}^{(n_l,m)} \operatorname{vec}(\mathbf{W}) = \operatorname{vec}(\mathbf{W}^\top)$. And

$$\begin{cases} \dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X}) \right)}{\partial \operatorname{vec}(\mathbf{U}_l)} = \alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}^{(k,l)} \left(\mathbf{X}^\top \otimes \mathbf{I}_{n_l} \right), \\ \dfrac{\partial \operatorname{vec}(f_l(\mathbf{Z}, \mathbf{X}))}{\partial \operatorname{vec}(\mathbf{b}_l)} = \alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}^{(k,l)} \mathbf{1}_N. \end{cases}$$

Thus, we get:

$$
\begin{aligned}
&\frac{\partial \mathbf{z}^{k+1}}{\partial \operatorname{vec}(\mathbf{W}_l)} = \frac{\partial \mathbf{z}^{k+1}}{\partial \mathbf{z}^{k}}\frac{\partial \mathbf{z}^{k}}{\partial \operatorname{vec}(\mathbf{W}_l)} + \frac{\partial \mathbf{z}^{k+1}}{\partial \operatorname{vec}(\mathbf{W}_l)} \\
=&(\beta_{k+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{k+1})\mathbf{A}(k,1))\frac{\partial \mathbf{z}^{k}}{\partial \operatorname{vec}(\mathbf{W}_l)} + \mathbf{A}(k,l+1)\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l-1)},\mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{W}_l\right)} \\
=&\sum_{\widetilde{k}=1}^{k}\left(\left(\prod_{q=\widetilde{k}+1}^{k}(\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{q+1})\mathbf{A}(q,1))\right)\mathbf{A}(\widetilde{k},l+1)\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(\widetilde{k},l-1)},\mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{W}_l\right)}\right).
\end{aligned}
$$

Similarly, we can also obtain:

$$
\frac{\partial \mathbf{z}^{k+1}}{\partial \operatorname{vec}(\mathbf{U}_l)} = \sum_{\widetilde{k}=1}^{k}\left(\mathbf{R}_{\widetilde{k}}\mathbf{A}(\widetilde{k},l+1)\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(\widetilde{k},l-1)},\mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{U}_l\right)}\right),
$$

and

$$
\frac{\partial \mathbf{z}^{k+1}}{\partial \operatorname{vec}(\mathbf{b}_l)} = \sum_{\widetilde{k}=1}^{k}\left(\mathbf{R}_{\widetilde{k}}\mathbf{A}(\widetilde{k},l+1)\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(\widetilde{k},l-1)},\mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{b}_l\right)}\right),
$$

where

$$
\mathbf{R}_{\widetilde{k}} = \left(\prod_{q=\widetilde{k}+1}^{k}(\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{q+1})\mathbf{A}(q,1))\right).
$$

□

Similar to the proof in the previous lemma, we can easily obtain

$$
\begin{cases}
\dfrac{\partial \mathbf{z}^{(k,l')}}{\partial \operatorname{vec}(\mathbf{W}_l)} = \mathbf{1}_{\{l'>l\}}\mathbf{A}(k,l',l+1)\dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l-1)},\mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{W}_l\right)} + \mathbf{A}(k,l',1)\dfrac{\partial \mathbf{z}^{k-1}}{\partial \operatorname{vec}(\mathbf{W}_l)}, \\
\dfrac{\partial \mathbf{z}^{(k,l')}}{\partial \operatorname{vec}(\mathbf{U}_l)} = \mathbf{1}_{\{l'>l\}}\mathbf{A}(k,l',l+1)\dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l-1)},\mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{U}_l\right)} + \mathbf{A}(k,l',1)\dfrac{\partial \mathbf{z}^{k-1}}{\partial \operatorname{vec}(\mathbf{U}_l)}, \\
\dfrac{\partial \mathbf{z}^{(k,l')}}{\partial \operatorname{vec}(\mathbf{b}_l)} = \mathbf{1}_{\{l'>l\}}\mathbf{A}(k,l',l+1)\dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l-1)},\mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{b}_l\right)} + \mathbf{A}(k,l',1)\dfrac{\partial \mathbf{z}^{k-1}}{\partial \operatorname{vec}(\mathbf{b}_l)},
\end{cases}
\tag{21}
$$

where $\frac{\partial \mathbf{z}^{(k-1)}}{\partial \operatorname{vec}(\cdot)}$ and $\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l-1)},\mathbf{X})\right)}{\partial \operatorname{vec}\left(\cdot\right)}$ are give in Eq.(18) and Eq.(19) in the previous lemma, and $\mathbf{1}_{\{l'>l\}}$ is the indicator function. Before providing the lemmas, we present two assumptions commonly used in the following lemmas.

Assumption 3 (Compact Set of Parameters)**.** *Given the learnable parameters*

$$
\boldsymbol{\theta} = \{(\mathbf{W}_l, \mathbf{U}_l, \mathbf{b}_l)\}_{l=1}^{L}, \quad \forall l \in [1, L],
$$

we assume $\|\mathbf{W}_l\|_2 \leq 1$ *and* $\max\{\|\mathbf{U}_l\|_2, \|\mathbf{b}_l\|_2\} \leq B_{Ub}$*. Moreover, we let* $\|\mathbf{X}\|_F \leq B_x$ *and* $\|\mathbf{W}_{L+1}\|_2 \leq B_L$*.*

Assumption 4 (Existence and boundedness)**.** *Let* $\mathcal{T}_{\beta,\lambda}(\mathbf{z},\mathbf{X},\boldsymbol{\theta}) = \beta(\mathbf{z} - \gamma(\nabla\mathcal{R}_z(\mathbf{z}) + \lambda\mathbf{z})) + (1-\beta)\mathcal{T}(\mathbf{z},\mathbf{X},\boldsymbol{\theta})$*. For any learnable parameters* $\boldsymbol{\theta}$ *and* $\mathbf{X}$ *satisfy Assumption 3, and* $\beta \in [0, \frac{1}{2}], \lambda \in [0, \frac{L_z}{2}]$*, we assume the fixed point set* $\operatorname{Fix}(\mathcal{T}(\cdot,\mathbf{X},\boldsymbol{\theta})) \subset \mathbb{R}^{mN}$ *is non-empty and uniformly bounded, i.e., for any* $\boldsymbol{\theta}$ *and* $\mathbf{X}$ *satisfy Assumption 3 and* $\beta \in [0, \frac{1}{2}], \lambda \in [0, \frac{L_z}{2}]$, $\forall \mathbf{z} \in \operatorname{Fix}(\mathcal{T}_{\beta,\lambda}(\cdot,\mathbf{X},\boldsymbol{\theta}))$, $\|\mathbf{z}\|_2 \leq B_1^*$*. Without loss of generality, we also let* $\left\|\mathbf{z}^1\right\| \leq B_1^*$*.*

We now show the uniform boundedness of $\mathbf{z}^k$ and $\mathbf{z}^{(k,l)}$.

Lemma 12. *If Assumption 3 and Assumption 4 hold, and* $\alpha \leq \frac{\ln 2}{2L}$*, then:*

$$
\left\|\mathbf{z}^k\right\| \leq B^* \quad \textit{and} \quad \left\|\mathbf{z}^{(k,l)}\right\| \leq 3B^*,
$$

here $B^* = \mathcal{O}(B_x, B_{Ub}, B_1^*)$*, and* B_1^* *is the uniform bound for any* $\mathbf{z} \in \operatorname{Fix}(\mathcal{T}_{\beta,\lambda}(\cdot,\mathbf{X},\boldsymbol{\theta}))$*, as shown in Assumption 4.*

Proof. We first give the boundedness for $\mathbf{z}^k$. Following the proof of Theorem 5, we have $\|\mathbf{z}^k - \bar{\mathbf{z}}^k\| \leq B_4^* = \mathcal{O}(\mathbf{z}^1, \rho, c, B_3^*)$. Note that $B_3^* = \mathcal{O}(B_2^*, B_1^*, \rho, c, \gamma, \eta), B_2^* = \mathcal{O}(B_x, B_{Ub}, B_1^*)$, ρ, c, γ, η are constants, and $\|\mathbf{z}^1\| \leq B_1^*$ by Assumption 4, so $B_4^* = \mathcal{O}(B_x, B_{Ub}, B_1^*)$.

And by the definition of $\bar{\mathbf{z}}^k$, it is the fixed point of equation $\mathbf{z} = \beta_k(\mathbf{z} - \gamma(\nabla\mathcal{R}_z(\mathbf{z}) + \lambda_k\mathbf{z})) + (1-\beta_k)\mathcal{T}(\mathbf{z},\mathbf{X},\boldsymbol{\theta})$, therefore $\|\bar{\mathbf{z}}^k\| \leq B_1^*$.

Thus, we have

$$\left\|\mathbf{z}^k\right\| \leq \left\|\mathbf{z}^k - \bar{\mathbf{z}}^k\right\| + \left\|\bar{\mathbf{z}}^k\right\| \leq B_4^* + B_1^* = \mathcal{O}(B_x, B_{Ub}, B_1^*).$$

On the other hand, we can suppose that $\|f_l \circ \cdots \circ f_1(\mathbf{0})\| \leq B_5^* = \mathcal{O}(B_x, B_{Ub}), \forall l \in [1, L]$, since they are continuously depended on $\mathbf{W}, \mathbf{U}, \mathbf{b}$ and α. Let $B^* = \max\{B_4^* + B_1^*, B_5^*\}$.

Note that $\mathbf{z}^{(k,l)} = f_l \circ \cdots \circ f_1(\mathbf{z}^k)$, and each $f_k(\mathbf{z}) = \mathbf{z} + \alpha\left(\mathbf{W}_k^\top \sigma(\mathbf{W}_k\mathbf{z} + \mathbf{U}_k\mathbf{x} + \mathbf{b}) - \mathbf{z}\right)$ is $(1+2\alpha)$-Lipschitz. Therefore

$$\begin{aligned}\left\|\mathbf{z}^{(k,l)}\right\| &\leq \|f_l \circ \cdots \circ f_1(\mathbf{z}^k) - f_l \circ \cdots \circ f_1(\mathbf{0})\| + \|f_l \circ \cdots \circ f_1(\mathbf{0})\| \\ &\leq (1+2\alpha)^l \|\mathbf{z}^k - \mathbf{0}\| + B^* \leq \exp\{2\alpha L\} B^* + B^* \leq 2B^* + B^* = 3B^*.\end{aligned}$$

□

For $\zeta \in (0,1)$, we say an operator f is *ζ-averaged* if $f = (1-\zeta)\mathcal{I} + \zeta\mathcal{S}$ for some nonexpansive operator $\mathcal{S}$.

Lemma 13. *If Assumption 1 holds, given any learnable parameters satisfy Assumption 3, the function $f := f_l \circ \cdots \circ f_1(\cdot, \mathbf{x})$ is averaged for all $l \in [1, L]$.*

Proof. Note that:

$$\frac{\partial \operatorname{vec}(f_l(\mathbf{Z}, \mathbf{X}))}{\partial \mathbf{z}} = \alpha\Big(\mathbf{I}_N \otimes \mathbf{W}_l^\top\Big)\mathbf{D}(\mathbf{I}_N \otimes \mathbf{W}_l) + (1-\alpha)\mathbf{I}_{mN}.$$

Hence, we have:

$$\left\|\frac{\partial \operatorname{vec}(f_l(\mathbf{Z}, \mathbf{X}))}{\partial \mathbf{z}}\right\| \leq \alpha\|\mathbf{W}_l\|_2^2\|\mathbf{D}\|_2 + (1-\alpha) \leq 1,$$

where the last inequality comes from Assumption 1 and Assumption 3. Hence the function $f_l(\cdot, \mathbf{x})$ is averaged for any parameters satisfy Assumption 3. Moreover, by Proposition 4.46 in [49], the composition of l operators which are averaged with respect to the same norm is also averaged, i.e., $f_l \circ \cdots \circ f_1(\cdot, \mathbf{x})$ is averaged when each $f_l(\cdot)$ is averaged. □

We now provides the bounds for $\mathbf{A}_{k,l}$ and $\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \operatorname{vec}(\cdot)}$.

Lemma 14. *If Assumption 3 and Assumption 1 hold, then:*

$$\begin{cases} \|\mathbf{A}(k,l)\|_2 \leq 1, \quad \left\|\prod_{q=\widetilde{k}+1}^{k} (\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{q+1})\mathbf{A}(q,1))\right\| \leq 1, \\ \left\|\dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \operatorname{vec}(\mathbf{W}_l)}\right\| \leq \alpha(\sqrt{mn_l}\sigma(0) + 6B^* + 2B_{Ub}B_x), \\ \left\|\dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \operatorname{vec}(\mathbf{U}_l)}\right\| \leq \alpha B_x, \quad \left\|\dfrac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \operatorname{vec}(\mathbf{b}_l)}\right\| \leq \alpha\sqrt{N}. \end{cases} \tag{22}$$

Proof. First of all, we get:

$$\begin{aligned}\left\|\mathbf{A}(k,\widetilde{l})\right\|_2 \leq& \prod_{l=\widetilde{l}}^{L} \left\|\Big(\alpha\Big(\mathbf{I}_N \otimes \mathbf{W}_l^\top\Big)\mathbf{D}^{(k,l)}(\mathbf{I}_N \otimes \mathbf{W}_l) + (1-\alpha)\mathbf{I}_{mN}\Big)\right\|_2 \\ \leq& \prod_{l=\widetilde{l}}^{L} \Big(\alpha\left\|\mathbf{D}^{(k,l)}\right\|_2 \|\mathbf{W}_l\|_2^2 + (1-\alpha)\Big) \leq 1,\end{aligned}$$

where the last inenquality comes form Assumption 3 and Assumption 1. Note that $\gamma < \frac{1}{2L_z+1}$, hence we have $\|\mathcal{V}_\gamma\|_2 \leq 1$. Thus, we have:

$$\left\|\prod_{q=\widetilde{k}+1}^{k} (\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{q+1})\mathbf{A}(q,1))\right\|_2 \leq \prod_{q=\widetilde{k}+1}^{k} \left(\beta_{q+1}\|\mathcal{V}_\gamma\|_2 + (1-\beta_{q+1})\right) \leq 1,$$

where the first inequality we utilize $\left\|\mathbf{A}(k,\widetilde{l})\right\|_2 \leq 1$. By Assumption 1, we can easily obtian that $\sigma(a) \leq a + \sigma(0),\ \forall a \in \mathbb{R}$. Hence, for $\mathbf{G}^{(k,l)} := \sigma\Big(\mathbf{W}_l\mathbf{Z}^{(k,l-1)} + \mathbf{U}_l\mathbf{X} + \mathbf{b}_l\Big)$, we have:

$$\begin{aligned}\left\|\mathbf{G}^{(k,l)}\right\|_F \leq& \sqrt{mn_l}\sigma(0) + \left\|\mathbf{W}_l\mathbf{Z}^{(k,l-1)} + \mathbf{U}_l\mathbf{X} + \mathbf{b}_l\right\|_F \\ \leq& \sqrt{mn_l}\sigma(0) + \left\|\mathbf{z}^{(k,l)}\right\| + B_{Ub}\|\mathbf{X}\|_F + \sqrt{N}B_{Ub} \\ \leq& \sqrt{mn_l}\sigma(0) + 3B^* + 2B_{Ub}B_x,\end{aligned}$$

where, w.l.o.g, we use $\|\mathbf{X}\|_F = \Theta(\sqrt{N})$ in the last inequality. Then, by Eq.(19), we get:

$$\begin{aligned}&\left\|\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{W}_l\right)}\right\| \\ &\leq \alpha\Big(\left\|\mathbf{G}^{(k,l)}\right\|_2 + \|\mathbf{W}_l\|_2 \left\|\mathbf{D}^{(k,l)}\right\|_2 \left\|\mathbf{Z}^{(k,l)}\right\|_2\Big) \leq \alpha(\sqrt{mn_l}\sigma(0) + 6B^* + 2B_{Ub}B_x),\end{aligned}$$

where we utilize $\left\|\mathbf{K}^{(n_l,m)}\right\|_2 = 1$. Similarly, we have:

$$\left\|\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{U}_l\right)}\right\| \leq \alpha B_x, \quad \text{and} \quad \left\|\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{b}_l\right)}\right\| \leq \alpha\sqrt{N}.$$

□

Lemma 15. *If Assumption 3 and Assumption 1 hold, then:*

$$\begin{cases}\left\|\dfrac{\partial \mathbf{z}^k}{\partial \operatorname{vec}(\mathbf{W}_l)}\right\|_2 \leq (k-1)\alpha(\sqrt{mn_l}\sigma(0) + 6B^* + 2B_{Ub}B_x), \\ \left\|\dfrac{\partial \mathbf{z}^k}{\partial \operatorname{vec}(\mathbf{U}_l)}\right\|_2 \leq (k-1)\alpha B_x, \quad \left\|\dfrac{\partial \mathbf{z}^k}{\partial \operatorname{vec}(\mathbf{b}_l)}\right\|_2 \leq (k-1)\alpha\sqrt{N}, \\ \left\|\dfrac{\partial \mathbf{z}^{k,l'}}{\partial \operatorname{vec}(\mathbf{W}_l)}\right\|_2 \leq \big(\mathbf{1}_{\{l'>l\}} + k - 2\big)\alpha(\sqrt{mn_l}\sigma(0) + 6B^* + 2B_{Ub}B_x), \\ \left\|\dfrac{\partial \mathbf{z}^k}{\partial \operatorname{vec}(\mathbf{U}_l)}\right\|_2 \leq \big(\mathbf{1}_{\{l'>l\}} + k - 2\big)\alpha B_x, \quad \left\|\dfrac{\partial \mathbf{z}^k}{\partial \operatorname{vec}(\mathbf{b}_l)}\right\|_2 \leq \big(\mathbf{1}_{\{l'>l\}} + k - 2\big)\alpha\sqrt{N}.\end{cases} \tag{23}$$

Moreover, we get:

$$\begin{cases}\|\operatorname{vec}\left(\nabla_{\mathbf{W}_l}\ell(\mathbf{y}, \mathbf{y}_0)\right)\| \leq (K-1)\alpha(\sqrt{mn_l}\sigma(0) + 6B^* + 2B_{Ub}B_x)B_L\|\mathbf{y} - \mathbf{y}_0\|, \\ \|\operatorname{vec}\left(\nabla_{\mathbf{U}_l}\ell(\mathbf{y}, \mathbf{y}_0)\right)\| \leq (K-1)\alpha B_x B_L\|\mathbf{y} - \mathbf{y}_0\|, \\ \|\operatorname{vec}\left(\nabla_{\mathbf{b}_l}\ell(\mathbf{y}, \mathbf{y}_0)\right)\| \leq (K-1)\alpha\sqrt{N}B_l\|\mathbf{y} - \mathbf{y}_0\|.\end{cases} \tag{24}$$

Proof. By Eq.(18), we have:

$$\left\|\frac{\partial \mathbf{z}^k}{\partial \operatorname{vec}(\mathbf{W}_l)}\right\|_2 \leq \sum_{\tilde{k}=1}^{k-1}\left(\left\|\frac{\partial \operatorname{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \operatorname{vec}\left(\mathbf{W}_l\right)}\right\|\right) \leq (k-1)\alpha(\sqrt{mn_l}\sigma(0) + 6B^* + 2B_{Ub}B_x).$$

where we use the results in Eq.(22). Similarly, by Eq.(18), we can also have:

$$\left\|\frac{\partial \mathbf{z}^k}{\partial \operatorname{vec}(\mathbf{U}_l)}\right\|_2 \leq (k-1)\alpha B_x, \quad \text{and} \quad \left\|\frac{\partial \mathbf{z}^k}{\partial \operatorname{vec}(\mathbf{b}_l)}\right\|_2 \leq \sqrt{N}(k-1)\alpha.$$

By Eq.(21), an immediate consequence of these bounds is:

$$\left\|\frac{\partial \mathbf{z}^{(k,l')}}{\partial \operatorname{vec}(\mathbf{W}_l)}\right\|_2 \leq \big(\mathbf{1}_{\{l'>l\}} + k - 2\big)\alpha(\sqrt{mn_l}\sigma(0) + 6B^* + 2B_{Ub}B_x).$$

Similarly, we also get:

$$\left\|\frac{\partial \mathbf{z}^{(k,l')}}{\partial \operatorname{vec}(\mathbf{U}_l)}\right\|_2 \leq \big(\mathbf{1}_{\{l'>l\}} + k - 2\big)\alpha B_x, \quad \text{and} \quad \left\|\frac{\partial \mathbf{z}^{(k,l')}}{\partial \operatorname{vec}(\mathbf{b}_l)}\right\|_2 \leq \big(\mathbf{1}_{\{l'>l\}} + k - 2\big)\sqrt{N}\alpha.$$

Note that, we already have:

$$\operatorname{vec}\left(\nabla_{\mathbf{W}_l}\ell(\mathbf{y}, \mathbf{y}_0)\right) = \left(\frac{\partial \ell(\mathbf{y}, \mathbf{y}_0)}{\partial \mathbf{z}^K}\frac{\partial \mathbf{z}^K}{\partial \operatorname{vec}(\mathbf{W}_l)}\right)^\top = \left(\frac{\partial \mathbf{z}^K}{\partial \operatorname{vec}(\mathbf{W}_l)}\right)^\top \Big(\mathbf{I}_N \otimes \mathbf{W}_{L+1}^\top\Big)(\mathbf{y} - \mathbf{y}_0).$$

Hence, by Eq.(22), we can easily get:

$$\|\operatorname{vec}\left(\nabla_{\mathbf{W}_l}\ell(\mathbf{y}, \mathbf{y}_0)\right)\| \leq (K-1)\alpha(\sqrt{mn_l}\sigma(0) + 6B^* + 2B_{Ub}B_x)B_L\|\mathbf{y} - \mathbf{y}_0\|.$$

Similarly, we have:

$$\begin{cases}\|\operatorname{vec}\left(\nabla_{\mathbf{U}_l}\ell(\mathbf{y}, \mathbf{y}_0)\right)\| \leq (K-1)\alpha B_x B_L\|\mathbf{y} - \mathbf{y}_0\|, \\ \|\operatorname{vec}\left(\nabla_{\mathbf{b}_l}\ell(\mathbf{y}, \mathbf{y}_0)\right)\| \leq (K-1)\alpha\sqrt{N}B_L\|\mathbf{y} - \mathbf{y}_0\|.\end{cases}$$

□

Lemma 16. *We let for all $l \in [1, L]$, $\sigma_{\min}(\mathbf{W}_l) \geq \sigma_m$. If $\sigma'(\cdot) \geq \kappa > 0$ and Assumption 3 hold, then:*

$$\|\mathrm{vec}\left(\nabla \ell(\mathbf{y}, \mathbf{y}_0)\right)\|^2 \geq \sum_{l=1}^{L} \|\mathrm{vec}\left(\nabla_{\mathbf{b}_l} \ell(\mathbf{y}, \mathbf{y}_0)\right)\|^2 \geq cK^2 L \alpha^2 \kappa^2 \sigma_m^2 N \sigma_{\min}^2(\mathbf{W}_{L+1}) \|\mathbf{y} - \mathbf{y}_0\|^2, \tag{25}$$

for some $0 < c < 1$.

Proof. Recall that, we already have:

$$\frac{\partial \mathbf{z}^{k+1}}{\partial \,\mathrm{vec}(\mathbf{b}_l)} = \sum_{\widetilde{k}=1}^{k} \left(\mathbf{R}_{\widetilde{k}} \mathbf{A}(\widetilde{k}, L, l+1) \frac{\partial \,\mathrm{vec}\left(f_l(\mathbf{Z}^{(\widetilde{k}, L, l-1)}, \mathbf{X})\right)}{\partial \,\mathrm{vec}\left(\mathbf{b}_l\right)} \right),$$

where

$$\mathbf{R}_{\widetilde{k}} = \left(\prod_{q=\widetilde{k}+1}^{k} \left(\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1 - \beta_{q+1})\mathbf{A}(q, 1)\right) \right).$$

First of all, we note that

$$\|\mathcal{V}_\gamma\|_2 \geq \frac{1}{4}.$$

On the other hand, for any $k \in [K]$, we have:

$$\mathbf{A}(k, L, l+1) \geq (1 - \alpha)^{L-l}.$$

Hence, when $\alpha K L < 1$, we have

$$\begin{aligned} \left\| \prod_{q=\widetilde{k}+1}^{k} \left(\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1 - \beta_{q+1})\mathbf{A}(q, L, 1)\right) \right\|_2 \geq & \prod_{q=\widetilde{k}+1}^{k} \left(\frac{\beta_{q+1}}{4} + (1 - \beta_{q+1})(1 - \alpha L) \right) \\ \geq & (1 - \alpha L)^{(k - \widetilde{k})}. \end{aligned}$$

We observe that

$$\sum_{\widetilde{k}=1}^{k} (1 - \alpha L)^{(k - \widetilde{k})} \geq \frac{1 - (1 - \alpha L)^k}{\alpha L} \geq c_1 k,$$

for some $0 < c_1 < 1$. And for any $k \in [K]$,

$$\alpha \left(\mathbf{I}_N \otimes \mathbf{W}_l^\top \right) \mathbf{D}^{(k,l)} \mathbf{1}_N \geq \alpha \sigma_m \kappa \sqrt{N}.$$

Then by Eq.(18) and Eq.(19), we can obtain:

$$\left\| \frac{\partial \mathbf{z}^K}{\partial \,\mathrm{vec}(\mathbf{b}_l)} \right\|_2^2 \geq K^2 (1 - \alpha)^{2(L-l)} \alpha^2 \kappa^2 \sigma_m^2 N.$$

Observing that:

$$\sum_{l=1}^{L} (1 - \alpha)^{2L - l} = \frac{1 - (1 - \alpha)^{2L}}{1 - (1 - \alpha)^2} \geq c_2 L,$$

for some $0 < c_2 < 1$. Hence, we have:

$$\sum_{l=1}^{L} \left\| \frac{\partial \mathbf{z}^K}{\partial \,\mathrm{vec}(\mathbf{b}_l)} \right\|_2^2 \geq cK^2 L \alpha^2 \kappa^2 \sigma_m^2 N,$$

for some $0 < c < 1$. Finally, we can conclude that:

$$\sum_{l=1}^{L} \|\mathrm{vec}\left(\nabla \ell(\mathbf{y}, \mathbf{y}_0)\right)\|^2 \geq \|\mathrm{vec}\left(\nabla_{\mathbf{b}_l} \ell(\mathbf{y}, \mathbf{y}_0)\right)\|^2 \geq cK^2 L \alpha^2 \kappa^2 \sigma_m^2 N \sigma_{\min}^2(\mathbf{W}_{L+1}) \|\mathbf{y} - \mathbf{y}_0\|^2.$$

□

Before proceeding, let us consider how to calculate $\frac{\partial \mathbf{z}^k}{\partial \mathbf{x}}$.

Lemma 17. *If Assumption 1 and Assumption 3 hold, then:*

$$\left\| \frac{\partial \mathbf{z}^{k+1}}{\partial \mathbf{x}} \right\|_2 \leq \alpha K L B_{Ub}. \tag{26}$$

Moreover, if $g(\mathbf{X}_0, \mathbf{W})$ is smooth and L_g-Lipschitz continuous w.r.t $\mathbf{W}$, we can obtain:

$$\|\mathrm{vec}\left(\nabla_{\mathbf{W}_0}\ell(\mathbf{y}, \mathbf{y}_0)\right)\| \leq \alpha K L B_{Ub} B_L L_g \|\mathbf{y} - \mathbf{y}_0\|. \tag{27}$$

Proof.

$$\begin{aligned}
\frac{\partial \mathbf{z}^{k+1}}{\partial \mathbf{x}} &= \frac{\partial \mathbf{z}^{k+1}}{\partial \mathbf{z}^{k}}\frac{\partial \mathbf{z}^{k}}{\partial \mathbf{x}} + \sum_{l=1}^{L}\left(\mathbf{A}(k, L, l)\frac{\partial\,\mathrm{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \mathbf{x}}\right) \\
&= \beta_{k+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{k+1})\mathbf{A}(k, L, 1)\frac{\partial \mathbf{z}^{k-1}}{\partial \mathbf{x}} + \sum_{l=1}^{L}\left(\mathbf{A}(k, L, l)\frac{\partial\,\mathrm{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \mathbf{x}}\right) \\
&= \sum_{\widetilde{k}=1}^{k}\left(\left(\prod_{q=\widetilde{k}+1}^{k}\left(\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{q+1})\mathbf{A}(q, L, 1)\right)\right)\sum_{l=1}^{L}\left(\mathbf{A}(k, L, l)\frac{\partial\,\mathrm{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \mathbf{x}}\right)\right),
\end{aligned}$$

where the second inequality comes from Eq.(20) and

$$\frac{\partial\,\mathrm{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \mathbf{x}} = \alpha\left(\mathbf{I}_N \otimes \mathbf{W}_l^\top\right)\mathbf{D}^{(k,l)}(\mathbf{I}_N \otimes \mathbf{U}_l).$$

By the bounds given in Eq.(22), we can get:

$$\left\|\frac{\partial\,\mathrm{vec}\left(f_l(\mathbf{Z}^{(k,l)}, \mathbf{X})\right)}{\partial \mathbf{x}}\right\|_2 \leq \alpha B_{Ub}.$$

Hence, we can immediately get:

$$\left\|\frac{\partial \mathbf{z}^{k+1}}{\partial \mathbf{x}}\right\|_2 \leq \alpha K L B_{Ub}.$$

Note that:

$$\mathrm{vec}\left(\nabla_{\mathbf{W}_0}\ell(\mathbf{y}, \mathbf{y}_0)\right) = \left(\frac{\partial \mathbf{z}^{K}}{\partial \mathbf{x}}\frac{\partial \mathbf{x}}{\partial\,\mathrm{vec}\left(\mathbf{W}_0\right)}\right)^\top\left(\mathbf{I}_N \otimes \mathbf{W}_{L+1}^\top\right)(\mathbf{y} - \mathbf{y}_0).$$

Thus, we can conclude that:

$$\|\mathrm{vec}\left(\nabla_{\mathbf{W}_0}\ell(\mathbf{y}, \mathbf{y}_0)\right)\| \leq \alpha K L B_{Ub} B_L L_g \|\mathbf{y} - \mathbf{y}_0\|.$$

□

Lemma 18. *If Assumption 1 and Assumption 3 hold, then:*

$$\left\|\frac{\partial \mathbf{z}^{K}}{\partial\,\mathrm{vec}\left(\mathbf{W}_{L+1}\right)}\right\|_2 \leq 2K\gamma B^* B_L. \tag{28}$$

Moreover, we can obtain:

$$\left\|\mathrm{vec}\left(\nabla_{\mathbf{W}_{L+1}}\ell(\mathbf{y}, \mathbf{y}_0)\right)\right\| \leq 3KB^*\|\mathbf{y} - \mathbf{y}_0\|. \tag{29}$$

Proof. We already have:

$$\frac{\partial \mathbf{z}^{K}}{\partial\,\mathrm{vec}(\mathbf{W}_{L+1})} = \frac{\partial \mathbf{z}^{K}}{\partial \mathbf{z}^{K-1}}\frac{\partial \mathbf{z}^{K-1}}{\partial\,\mathrm{vec}(\mathbf{W}_{L+1})} + \mathbf{H}^k = \sum_{\widetilde{k}=1}^{K}\left(\left(\prod_{q=\widetilde{k}+1}^{K}\left(\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma) + (1-\beta_{q+1})\mathbf{A}(q, 1)\right)\right)\mathbf{H}^k\right),$$

where we let

$$\mathbf{H}^k := -\gamma\beta_K\left(\left(\left(\mathbf{W}_{L+1}\mathbf{Z}^K\right)^\top \otimes \mathbf{I}_m\right)\mathbf{K}^{(d_y,m)} + \left(\mathbf{I}_N \otimes \mathbf{W}_{L+1}^\top\right)\left(\left(\mathbf{Z}^K\right)^\top \otimes \mathbf{I}_{d_y}\right)\right).$$

Hence, we can obtain:

$$\left\|\frac{\partial \mathbf{z}^{K}}{\partial\,\mathrm{vec}(\mathbf{W}_{L+1})}\right\| \leq 2K\gamma B^* B_L,$$

where we utilize the bounds in Assumption 3. Note that:

$$\mathrm{vec}\left(\nabla_{\mathbf{W}_{L+1}}\ell(\mathbf{y}, \mathbf{y}_0)\right) = \left(\frac{\partial \mathbf{z}^{K}}{\partial\,\mathrm{vec}(\mathbf{W}_{L+1})}\right)^\top\left(\mathbf{I}_N \otimes \mathbf{W}_{L+1}^\top\right)(\mathbf{y} - \mathbf{y}_0) + \left(\mathbf{z}^K \otimes \mathbf{I}_{d_y}\right)(\mathbf{y} - \mathbf{y}_0).$$

Thus, we can conclude that:

$$\left\|\mathrm{vec}\left(\nabla_{\mathbf{W}_{L+1}}\ell(\mathbf{y}, \mathbf{y}_0)\right)\right\| \leq 3K\gamma B^* B_L^2\|\mathbf{y} - \mathbf{y}_0\| \leq 3KB^*\|\mathbf{y} - \mathbf{y}_0\|.$$

□

In the following lemma, we consider the Lipschitz continuity, we let $\widetilde{\boldsymbol{\theta}} := \{\mathbf{W}_{L+1}, \boldsymbol{\theta}, \mathbf{W}_0\}$. Moreover, we denote:

$$\mathcal{J}_{\mathbf{z}^K} := \left[\tfrac{\partial \mathbf{z}^K}{\partial \operatorname{vec}(\mathbf{W}_{L+1})} \quad \tfrac{\partial \mathbf{z}^K}{\partial \operatorname{vec}(\mathbf{W}_{L})} \quad \tfrac{\partial \mathbf{z}^K}{\partial \operatorname{vec}(\mathbf{U}_{L})} \quad \tfrac{\partial \mathbf{z}^K}{\partial \operatorname{vec}(\mathbf{b}_{L})} \quad \cdots \quad \tfrac{\partial \mathbf{z}^K}{\partial \operatorname{vec}(\mathbf{W}_{0})}\right]. \tag{30}$$

Lemma 19. *Let Assumption 1 hold. We assume that the activation function is L_σ-smooth, and $g(\mathbf{X}_0, \mathbf{W})$ is smooth and L_g-Lipschitz continuous w.r.t $\mathbf{W}$, given two parameters $\widetilde{\boldsymbol{\theta}}^a$ and $\widetilde{\boldsymbol{\theta}}^b$ satisfying Assumption 3, we have:*

$$\begin{cases} \left\|\mathbf{z}^k(\boldsymbol{\theta}^a) - \mathbf{z}^k(\boldsymbol{\theta}^b)\right\|_2 \le \left(2\alpha\sqrt{L}KC_zL_g\right)\left\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\right\|, \\ \left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a) - \mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\| \le \left(2\alpha\sqrt{L}KC_zL_g\right)\left\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\right\|, \\ \left\|\mathbf{D}^{(k,l)}(\boldsymbol{\theta}^a) - \mathbf{D}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2 \le L_\sigma\left(C_D\|\Delta\boldsymbol{\theta}_l\| + B_{Ub}L_g\left\|\mathbf{W}_0^a - \mathbf{W}_0^b\right\|_2 + \left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a) - \mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|\right), \\ \left\|\mathbf{A}(k, l_1, l_2, \boldsymbol{\theta}^a) - \mathbf{A}(k, l_1, l_2, \boldsymbol{\theta}^b)\right\|_2 \le \left(4\alpha\sqrt{L}L_\sigma C_zL_g\right)\left\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\right\|, \\ \left\|\mathcal{J}_{\mathbf{z}^K}(\widetilde{\boldsymbol{\theta}}^a) - \mathcal{J}_{\mathbf{z}^K}(\widetilde{\boldsymbol{\theta}}^b)\right\|_2 \le \left(9\alpha\sqrt{L}B^*L_\sigma C_zL_g + 2\sqrt{3L}\alpha KB^*L_\sigma C_D\right)\left\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\right\|, \end{cases} \tag{31}$$

where

$$\begin{cases} \|\Delta\boldsymbol{\theta}_l\| := \left(\left\|\mathbf{W}_l^a - \mathbf{W}_l^b\right\|_2 + \left\|\mathbf{U}_l^a - \mathbf{U}_l^b\right\|_2 + \left\|\mathbf{b}_l^a - \mathbf{b}_l^b\right\|\right), \\ C_D := \max\{3B^*, B_x\}, \quad \textit{and} \quad C_z := (\sqrt{mn_l}\sigma(0) + 6B^* + 3B_{Ub}B_x) \end{cases}$$

Proof. Due to the smoothness of the activation function, $\mathbf{z}^k$ is a continuous function of $\boldsymbol{\theta}$ on the compact set given in Assumption 3. By the mean value theorem in high dimension, we can have:

$$\left\|\mathbf{z}^k(\boldsymbol{\theta}^a) - \mathbf{z}^k(\boldsymbol{\theta}^b)\right\|_2 \le \left\|\left[\tfrac{\partial \mathbf{z}^k(\hat{\boldsymbol{\theta}})}{\partial \operatorname{vec}(\mathbf{W}_L)} \quad \tfrac{\partial \mathbf{z}^k(\hat{\boldsymbol{\theta}})}{\partial \operatorname{vec}(\mathbf{U}_L)} \quad \tfrac{\partial \mathbf{z}^k(\hat{\boldsymbol{\theta}})}{\partial \operatorname{vec}(\mathbf{b}_L)} \quad \cdots \quad \tfrac{\partial \mathbf{z}^k(\hat{\boldsymbol{\theta}})}{\partial \operatorname{vec}(\mathbf{W}_0)}\right]\right\|_2 \left\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\right\|,$$

where $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}^a + \xi(\boldsymbol{\theta}^b - \boldsymbol{\theta}^a)$ for some $\xi \in (0,1)$. Due to the convexity of the bounded ball in euclidean space, we can conclude that $\widetilde{\boldsymbol{\theta}}$ also satisfy Assumption 3. Hence, by Eq.(23) , we can get:

$$\begin{aligned} &\left\|\left[\tfrac{\partial \mathbf{z}^k(\hat{\boldsymbol{\theta}})}{\partial \operatorname{vec}(\mathbf{W}_L)} \quad \tfrac{\partial \mathbf{z}^k(\hat{\boldsymbol{\theta}})}{\partial \operatorname{vec}(\mathbf{U}_L)} \quad \tfrac{\partial \mathbf{z}^k(\hat{\boldsymbol{\theta}})}{\partial \operatorname{vec}(\mathbf{b}_L)} \quad \cdots \quad \tfrac{\partial \mathbf{z}^k(\hat{\boldsymbol{\theta}})}{\partial \operatorname{vec}(\mathbf{W}_0)}\right]\right\|_2 \le \left(L(k-1)^2\alpha^2(C_z^2 + B_x^2 + N) + \left\|\frac{\partial \mathbf{z}^k(\hat{\boldsymbol{\theta}})}{\partial \mathbf{x}}\frac{\partial \mathbf{x}(\hat{\boldsymbol{\theta}})}{\partial \mathbf{W}_0}\right\|_2^2\right)^{\frac{1}{2}} \\ \le&\left(3L(k-1)^2\alpha^2C_z^2 + (\alpha KLB_{Ub})^2\left\|\frac{\partial \mathbf{x}(\hat{\boldsymbol{\theta}})}{\partial \mathbf{W}_0}\right\|_2^2\right)^{\frac{1}{2}} \le \left(3L(k-1)^2\alpha^2C_z^2 + \alpha^2K^2L^2B_{Ub}^2L_g^2\right)^{\frac{1}{2}} \\ \le&\left(4LK^2\alpha^2C_z^2L_g^2\right)^{\frac{1}{2}} = 2\alpha\sqrt{L}KC_zL_g, \end{aligned}$$

where we utilize the bound in Eq.(26) in the penultimate inequality, and note that $C_z = \Theta(B_{Ub}\sqrt{N}) \gg L$. Hence, we have:

$$\left\|\mathbf{z}^k(\boldsymbol{\theta}^a) - \mathbf{z}^k(\boldsymbol{\theta}^b)\right\|_2 \le \left(2\alpha\sqrt{L}KC_zL_g\right)\left\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\right\|.$$

Similarly, by Eq.(21) and Eq.(23), we can have:

$$\left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a) - \mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\| \le \left(2\alpha\sqrt{L}KC_zL_g\right)\left\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\right\|.$$

Due to the activation function is L_σ-smooth, we have:

$$\begin{aligned} &\left\|\mathbf{D}^{(k,l)}(\boldsymbol{\theta}^a) - \mathbf{D}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2 = \left\|\sigma'\left(\mathbf{W}_l^a\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^a) + \mathbf{U}_l^a\mathbf{X}^a + \mathbf{b}_l^a\right) - \sigma'\left(\mathbf{W}_l^b\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^b) + \mathbf{U}_l^b\mathbf{X}^b + \mathbf{b}_l^b\right)\right\|_\infty \\ \le& L_\sigma\left\|\left(\mathbf{W}_l^a\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^a) + \mathbf{U}_l^a\mathbf{X}^a + \mathbf{b}_l^a\right) - \left(\mathbf{W}_l^b\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^b) + \mathbf{U}_l^b\mathbf{X}^b + \mathbf{b}_l^b\right)\right\|_\infty \\ \le& L_\sigma\left\|\left(\mathbf{W}_l^a\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^a) + \mathbf{U}_l^a\mathbf{X}^a + \mathbf{b}_l^a\right) - \left(\mathbf{W}_l^b\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^b) + \mathbf{U}_l^b\mathbf{X}^b + \mathbf{b}_l^b\right)\right\|_2 \\ \le& L_\sigma\left(\left\|\mathbf{W}_l^a - \mathbf{W}_l^b\right\|_2\left\|\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2 + \|\mathbf{W}_l^a\|_2\left\|\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^a) - \mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2 + \right. \\ &\qquad \left. \left\|\mathbf{U}_l^a - \mathbf{U}_l^b\right\|_2\left\|\mathbf{X}^b\right\|_2 + \|\mathbf{U}_l^a\|_2\left\|\mathbf{X}^a - \mathbf{X}^b\right\|_2 + \sqrt{N}\left\|\mathbf{b}_l^a - \mathbf{b}_l^b\right\|\right) \\ \le& L_\sigma\left(3B^*\left\|\mathbf{W}_l^a - \mathbf{W}_l^b\right\|_2 + B_x\left\|\mathbf{U}_l^a - \mathbf{U}_l^b\right\|_2 + \sqrt{N}\left\|\mathbf{b}_l^a - \mathbf{b}_l^b\right\| + B_{Ub}L_g\left\|\mathbf{W}_0^a - \mathbf{W}_0^b\right\|_2 + \left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a) - \mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|\right) \\ \le& L_\sigma\left(C_D\|\Delta\boldsymbol{\theta}_l\| + B_{Ub}L_g\left\|\mathbf{W}_0^a - \mathbf{W}_0^b\right\|_2 + \left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a) - \mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|\right), \end{aligned}$$

where

$$C_D := \max\{3B^*, B_x\}, \quad \text{and} \quad \|\Delta\boldsymbol{\theta}_l\| := \left(\left\|\mathbf{W}_l^a - \mathbf{W}_l^b\right\|_2 + \left\|\mathbf{U}_l^a - \mathbf{U}_l^b\right\|_2 + \left\|\mathbf{b}_l^a - \mathbf{b}_l^b\right\|\right).$$

Based on this upper bound, by the telescoping sum, we can easily have:

$$
\begin{aligned}
&\left\|\mathbf{A}(k,l_1,l_2,\boldsymbol{\theta}^a)-\mathbf{A}(k,l_1,l_2,\boldsymbol{\theta}^b)\right\|_2 \leq \sum_{l=l_1}^{l_2}\left\|\prod_{i=l+1}^{l_2}\mathbf{T}^{(k,i)}(\boldsymbol{\theta}^a)\Big(\mathbf{T}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{T}^{(k,l)}(\boldsymbol{\theta}^b)\Big)\prod_{j=l-1}^{l_1}\mathbf{T}^{(k,i)}(\boldsymbol{\theta}^b)\right\|_2 \\
\leq &\sum_{l=l_1}^{l_2}\left\|\mathbf{T}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{T}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2 \leq \sum_{l=l_1}^{l_2}\alpha\Big(2\left\|\mathbf{W}_l^a-\mathbf{W}_l^b\right\|_2\left\|\mathbf{D}^{(k,l)}\right\|_2+\|\mathbf{W}_l^a\|_2\left\|\mathbf{W}_l^b\right\|_2\left\|\mathbf{D}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{D}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2\Big) \\
\leq &\sum_{l=l_1}^{l_2}\alpha\Big(2\left\|\mathbf{W}_l^a-\mathbf{W}_l^b\right\|_2+\left\|\mathbf{D}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{D}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2\Big).
\end{aligned}
$$

where we note that for $\prod_{j=l-1}^{l_1}$, j runs in the reverse order, we also let

$$
\mathbf{T}^{(k,i)} \coloneqq \Big(\alpha\Big(\mathbf{I}_N\otimes\mathbf{W}_i^\top\Big)\mathbf{D}^{(k,i)}(\mathbf{I}_N\otimes\mathbf{W}_i)+(1-\alpha)\mathbf{I}_{mN}\Big),
$$

and we use that $\left\|\mathbf{T}^{(k,i)}\right\|_2 \leq 1$ from Eq.(22) for the second inequality. We note that:

$$
\begin{aligned}
&\Big(2\left\|\mathbf{W}_l^a-\mathbf{W}_l^b\right\|_2+\left\|\mathbf{D}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{D}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2\Big) \\
\leq &C_D' L_\sigma\|\Delta\boldsymbol{\theta}_l\|+B_{Ub}L_\sigma L_g\left\|\mathbf{W}_0^a-\mathbf{W}_0^b\right\|_2+L_\sigma\left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|,
\end{aligned}
$$

where

$$
C_D' \coloneqq \max\{3B^*+2/L_\sigma, B_x\}.
$$

Hence, we can obtain:

$$
\begin{aligned}
&\left\|\mathbf{A}(k,l_1,l_2,\boldsymbol{\theta}^a)-\mathbf{A}(k,l_1,l_2,\boldsymbol{\theta}^b)\right\|_2 \leq \alpha L_\sigma\sum_{l=l_1}^{l_2}\Big(C_D'\|\Delta\boldsymbol{\theta}_l\|+B_{Ub}L_g\left\|\mathbf{W}_0^a-\mathbf{W}_0^b\right\|_2+\left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|\Big) \\
\leq &\sqrt{3(l_1-l_2)}\alpha L_\sigma C_D'\left\|\widetilde{\boldsymbol{\theta}}^a-\widetilde{\boldsymbol{\theta}}^b\right\|+\alpha L_\sigma\sum_{l=l_1}^{l_2}\Big(B_{Ub}L_g\left\|\mathbf{W}_0^a-\mathbf{W}_0^b\right\|_2+\left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|\Big) \\
\leq &\Big(\alpha L_\sigma L\Big(B_{Ub}L_g+2\alpha\sqrt{L}KC_zL_g\Big)+\alpha\sqrt{3L}\,L_\sigma C_D'\Big)\left\|\widetilde{\boldsymbol{\theta}}^a-\widetilde{\boldsymbol{\theta}}^b\right\| \\
\leq &\Big(3\alpha^2L^{3/2}KL_\sigma C_zL_g+\alpha\sqrt{3L}C_D'\Big)\left\|\widetilde{\boldsymbol{\theta}}^a-\widetilde{\boldsymbol{\theta}}^b\right\| \\
\leq &\Big(3\alpha\sqrt{L}L_\sigma C_zL_g+\alpha\sqrt{3L}C_D'\Big)\left\|\widetilde{\boldsymbol{\theta}}^a-\widetilde{\boldsymbol{\theta}}^b\right\| \leq \Big(4\alpha\sqrt{L}L_\sigma C_zL_g\Big)\left\|\widetilde{\boldsymbol{\theta}}^a-\widetilde{\boldsymbol{\theta}}^b\right\|.
\end{aligned}
$$

where we use $\alpha KL \ll 1$, w.l.o.g, and recal that:

$$
C_z \coloneqq (\sqrt{mn_l}\sigma(0)+6B^*+3B_{Ub}B_x).
$$

Now, we consider the Lipschitz continuity for $\mathbf{G}^{(k,l)}$, similar to $\mathbf{D}^{(k,l)}$:

$$
\begin{aligned}
&\left\|\mathbf{G}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{G}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2 \leq \left\|\Big(\mathbf{W}_l^a\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^a)+\mathbf{U}_l^a\mathbf{X}^a+\mathbf{b}_l^a\Big)-\Big(\mathbf{W}_l^b\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^b)+\mathbf{U}_l^b\mathbf{X}^b+\mathbf{b}_l^b\Big)\right\|_F \\
\leq &L_\sigma\Big(C_D\|\Delta\boldsymbol{\theta}_l\|+B_{Ub}L_g\left\|\mathbf{W}_0^a-\mathbf{W}_0^b\right\|_2+\left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|\Big),.
\end{aligned}
$$

where we use the fact that the activation is 1-Lipschitz continuous in the first inequality. Therefore, by Eq.(19), we get:

$$
\begin{aligned}
&\frac{1}{\alpha}\left\|\frac{\partial\operatorname{vec}\Big(f_l(\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^a),\mathbf{X}^a)\Big)}{\partial\operatorname{vec}(\mathbf{W}_l)}-\frac{\partial\operatorname{vec}\Big(f_l(\mathbf{Z}^{(k,l)}(\boldsymbol{\theta}^b),\mathbf{X}^b)\Big)}{\partial\operatorname{vec}(\mathbf{W}_l)}\right\| \\
\leq &\left\|\mathbf{G}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{G}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2+\Big(B^*\left\|\mathbf{W}_l^a-\mathbf{W}_l^b\right\|_2+B^*\left\|\mathbf{D}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{D}^{(k,l)}(\boldsymbol{\theta}^b)\right\|_2+\left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|\Big) \\
\leq &L_\sigma(B^*+1)\Big(C_D\|\Delta\boldsymbol{\theta}_l\|+B_{Ub}L_g\left\|\mathbf{W}_0^a-\mathbf{W}_0^b\right\|_2+\left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|\Big)+ \\
&B^*\left\|\mathbf{W}_l^a-\mathbf{W}_l^b\right\|_2+\left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\| \\
\leq &2L_\sigma B^*\Big(C_D'\|\Delta\boldsymbol{\theta}_l\|+B_{Ub}L_g\left\|\mathbf{W}_0^a-\mathbf{W}_0^b\right\|_2+\left\|\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^a)-\mathbf{z}^{(k,l)}(\boldsymbol{\theta}^b)\right\|\Big).
\end{aligned}
$$

On the other hand, also by the telescoping sum , we have:

$$\begin{aligned}
&\left\| \prod_{q=\widetilde{k}+1}^{k} (\beta_{q+1}(\mathbf{I}_N \otimes \mathcal{V}_\gamma^a) + (1-\beta_{q+1})\mathbf{A}(\boldsymbol{\theta}^a)) - \prod_{q=\widetilde{k}+1}^{k} \Big(\beta_{q+1}\Big(\mathbf{I}_N \otimes \mathcal{V}_\gamma^b\Big) + (1-\beta_{q+1})\mathbf{A}(\boldsymbol{\theta}^b)\Big) \right\|_2 \\
\leq & \sum_{q=\widetilde{k}+1}^{k} \Big(2\beta_{q+1}\gamma B_L \Big\|\mathbf{W}_{L+1}^a - \mathbf{W}_{L+1}^b\Big\|_2 + (1-\beta_{q+1})\Big\|\mathbf{A}(q,L,1,\boldsymbol{\theta}^a) - \mathbf{A}(q,L,1,\boldsymbol{\theta}^b)\Big\|_2\Big) \\
\leq & k\Big\|\mathbf{W}_{L+1}^a - \mathbf{W}_{L+1}^b\Big\|_2 + \sum_{q=\widetilde{k}+1}^{k} \Big\|\mathbf{A}(q,L,1,\boldsymbol{\theta}^a) - \mathbf{A}(q,L,1,\boldsymbol{\theta}^b)\Big\|_2,
\end{aligned}$$

where we utilize $\gamma B_L^2 < 1$ in the last inequality. Since for any k, l_1 and l_2, $\left\|\mathbf{A}(\boldsymbol{\theta}^a) - \mathbf{A}(\boldsymbol{\theta}^b)\right\|_2$ share the same upper bound, in the following proof, we omit the index (k, l_2, l_2). Combing all things together, by Eq.(18), we have:

$$\begin{aligned}
&\left\| \frac{\partial \mathbf{z}^K(\boldsymbol{\theta}^a)}{\partial \operatorname{vec}(\mathbf{W}_l)} - \frac{\partial \mathbf{z}^K(\boldsymbol{\theta}^a)}{\partial \operatorname{vec}(\mathbf{W}_l)} \right\|_2 \\
\leq & K\left((2K^2\alpha C_z)\Big(\Big\|\mathbf{W}_{L+1}^a - \mathbf{W}_{L+1}^b\Big\|_2 + \Big\|\mathbf{A}(\boldsymbol{\theta}^a) - \mathbf{A}(\boldsymbol{\theta}^b)\Big\|_2\Big) + \left\| \frac{\partial \operatorname{vec}\Big(f_l(\mathbf{Z}^{(K,l)}(\boldsymbol{\theta}^a))\Big)}{\partial \operatorname{vec}(\mathbf{W}_l)} - \frac{\partial \operatorname{vec}\Big(f_l(\mathbf{Z}^{(K,l)}(\boldsymbol{\theta}^b))\Big)}{\partial \operatorname{vec}(\mathbf{W}_l)} \right\| \right) \\
\leq & 2\alpha K^3 C_z \Big\|\mathbf{W}_{L+1}^a - \mathbf{W}_{L+1}^b\Big\|_2 + 2\alpha K B^* L_\sigma B_{Ub} L_g \Big\|\mathbf{W}_0^a - \mathbf{W}_0^b\Big\|_2 + 2\alpha K B^* L_\sigma C_D \|\Delta\boldsymbol{\theta}_l\| + \\
& \Big(2\alpha L_\sigma B^* 2\alpha\sqrt{L} K C_z L_g + 4\alpha K \sqrt{L} L_\sigma C_z L_g\Big)\Big\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\Big\| \\
\leq & 2\alpha K^3 C_z \Big\|\mathbf{W}_{L+1}^a - \mathbf{W}_{L+1}^b\Big\|_2 + 2\alpha K B^* L_\sigma B_{Ub} L_g \Big\|\mathbf{W}_0^a - \mathbf{W}_0^b\Big\|_2 + 2\alpha K B^* L_\sigma C_D \|\Delta\boldsymbol{\theta}_l\| + \\
& \Big(8\alpha^2 K \sqrt{L} B^* L_\sigma C_z L_g\Big)\Big\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\Big\| \\
\leq & 2\alpha K B^* L_\sigma C_D \|\Delta\boldsymbol{\theta}_l\| + \Big(9\alpha^2 K \sqrt{L} B^* L_\sigma C_z L_g\Big)\Big\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\Big\|,
\end{aligned}$$

where, w.l.o.g, we utilize the fact $K^3\sqrt{L} \leq B^*$. An Immediate consequence we can get is:

$$\begin{aligned}
&\Big\|\mathcal{J}_{\mathbf{z}^K}(\widetilde{\boldsymbol{\theta}}^a) - \mathcal{J}_{\mathbf{z}^K}(\widetilde{\boldsymbol{\theta}}^b)\Big\|_2 \\
\leq & \sum_{l=1}^{L} \left(\left\| \frac{\partial \mathbf{z}^K(\boldsymbol{\theta}^a))}{\partial \operatorname{vec}(\mathbf{W}_l)} - \frac{\partial \mathbf{z}^K(\boldsymbol{\theta}^a))}{\partial \operatorname{vec}(\mathbf{W}_l)} \right\|_2 + \left\| \frac{\partial \mathbf{z}^K(\boldsymbol{\theta}^a))}{\partial \operatorname{vec}(\mathbf{U}_l)} - \frac{\partial \mathbf{z}^K(\boldsymbol{\theta}^a))}{\partial \operatorname{vec}(\mathbf{U}_l)} \right\|_2 + \left\| \frac{\partial \mathbf{z}^K(\boldsymbol{\theta}^a))}{\partial \operatorname{vec}(\mathbf{b}_l)} - \frac{\partial \mathbf{z}^K(\boldsymbol{\theta}^a))}{\partial \operatorname{vec}(\mathbf{b}_l)} \right\|_2 \right) \\
\leq & \Big(L\Big(9\alpha^2 K \sqrt{L} B^* L_\sigma C_z L_g\Big) + 2\sqrt{3L}\alpha K B^* L_\sigma C_D\Big)\Big\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\Big\| \\
\leq & \Big(9\alpha\sqrt{L} B^* L_\sigma C_z L_g + 2\sqrt{3L}\alpha K B^* L_\sigma C_D\Big)\Big\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\Big\|.
\end{aligned}$$

We now finish the proof. □

D.2 Proof for Theorem 6

Before presentation the main results, we assume a mild condition for initialization.

Assumption 5 (Initial conditions)**.** *We assume the initialized parameters of OptDeq in Eq.(5) satisfy:*

$$\begin{cases}
\|\mathbf{W}_l^0\|_2 \leq \frac{3}{4}, \quad \sigma_{\min}(\mathbf{W}_l^0) \geq \left(\frac{1}{4} + \sigma_m\right), \\
\max\{\|\mathbf{U}_l^0\|_2, \|\mathbf{b}_l^0\|\} \leq \frac{B_{Ub}}{2}, \quad \|\mathbf{W}_0^0\|_2 \leq \frac{B_x}{2L_g}, \\
\|\mathbf{W}_{L+1}^0\|_2 \leq \frac{B_L}{2}, \quad \sigma_{\min}(\mathbf{W}_{L+1}^0) > 0,
\end{cases}$$

where L_g is the Lipschitz constant of the function $g(\cdot)$ and $\sigma_{\min}(\cdot)$ is the smallest singular value of a matrix.

Due to the extractor $g(\mathbf{X}, \cdot)$ is L_g- Lipschitz continuous, the initial conditions is equivalent to:

$$\begin{cases}
\|\mathbf{W}_l^0\|_2 \leq \frac{3}{4}, \quad \sigma_{\min}(\mathbf{W}_l^0) \geq \left(\frac{1}{4} + \sigma_m\right), \\
\max\{\|\mathbf{U}_l^0\|_2, \|\mathbf{b}_l^0\|\} \leq \frac{B_{Ub}}{2}, \\
\|\mathbf{W}_{L+1}^0\|_2 \leq \frac{B_L}{2}, \quad \sigma_{\min}(\mathbf{W}_{L+1}^0) > 0, \\
\|\mathbf{X}^0\|_F \leq \frac{B_x}{2}.
\end{cases}$$

Now, we are ready to prove the main theorem. We denote

$$\begin{cases} C_D := \max\{3B^*, B_x\}, \\ C_z := (\sqrt{mn_l}\sigma(0) + 6B^* + 3B_{Ub}B_x) \\ B_J := 2\alpha\sqrt{L}KC_z, \\ C_J := \left(9\alpha\sqrt{L}B^*L_\sigma C_z L_g + 2\sqrt{3L}\alpha KB^*L_\sigma C_D\right), \\ Q_0 := \frac{1}{4}K^2L\alpha^2\kappa^2\sigma_m^2 N\sigma_{\min}^2(\mathbf{W}_{L+1}^0), \\ Q_1 := \left(C_J\sqrt{\ell(\widetilde{\boldsymbol{\theta}}^0)} + 2\alpha\sqrt{L}KC_zL_gB_JB_L + B_J\sqrt{\ell(\widetilde{\boldsymbol{\theta}}^0)}\right), \\ Q_2 := \frac{10C_zB_LL_g}{Q_0}, \end{cases} \tag{32}$$

where $\ell(\widetilde{\boldsymbol{\theta}}^0)$ is the training loss at initialization.

Theorem (Global Convergence)**.** *Suppose Assumption 4 and Assumption 5 hold. Assume that the activation function is L_σ-Lipschitz smooth, strongly monotone and 1-Lipschitz continuous. Let the learning rate be $\eta < \min\left\{\frac{1}{Q_0}, \frac{1}{Q_1}\right\}$. If the training data size N is in the order:*

$$\sqrt{N} = \Omega\left(\frac{B_L\sqrt{\ell(\widetilde{\boldsymbol{\theta}}^0)}}{\kappa^2\sigma_m^2\sigma_{\min}^2(\mathbf{W}_{L+1}^0)}\right),$$

then the training loss vanishes at a linear rate as:

$$\ell(\widetilde{\boldsymbol{\theta}}^t) \leq \ell(\widetilde{\boldsymbol{\theta}}^0)(1 - \eta Q_0)^t,$$

where t is the number of iteration. Furthermore, the network parameters also converge to a global minimizer $\widetilde{\boldsymbol{\theta}}^*$ at a linear speed:

$$\|\widetilde{\boldsymbol{\theta}}^t - \widetilde{\boldsymbol{\theta}}^*\| \leq Q_2(1 - \eta Q_0)^{t/2}.$$

Proof. We already have $B^*, B_x < C_z = \mathcal{O}\left(B_{Ub}\sqrt{N}\right)$. Hence, when the data size N is large enough, i.e., when

$$\sqrt{N} = \Omega\left(\frac{B_L\sqrt{\ell(\widetilde{\boldsymbol{\theta}}^0)}}{\kappa^2\sigma_m^2\sigma_{\min}^2(\mathbf{W}_{L+1}^0)}\right),$$

we have:

$$\frac{2}{Q_0}\alpha KC_zB_L\|\mathbf{y}^0 - \mathbf{y}_0\| = \frac{8C_zB_L\|\mathbf{y}^0 - \mathbf{y}_0\|}{KL\alpha\kappa^2\sigma_m^2N\sigma_{\min}^2(\mathbf{W}_{L+1}^0)} = \Theta\left(\frac{B_L\sqrt{\ell(\widetilde{\boldsymbol{\theta}}^0)}}{\kappa^2\sigma_m^2\sigma_{\min}^2(\mathbf{W}_{L+1}^0)\sqrt{N}}\right) = \Theta(1). \tag{33}$$

Hence, w.l.o.g we let:

$$\frac{2}{Q_0}\alpha KC_zB_L\|\mathbf{y}^0 - \mathbf{y}_0\| \leq \frac{1}{4}.$$

We denote the index t to represent the iteration number during training, i.e., $\{\mathbf{W}^t_{L+1\},\boldsymbol{\theta}^t,\mathbf{W}_0^t}$ is the learnable parameters at the t-th iteration. We show by induction that, for every $\widetilde{t} > 0, t \in [1, \widetilde{t}]$ and $l \in [1, L]$, the following holds:

$$\begin{cases} \|\mathbf{W}_l^t\|_2 \leq 1, \quad \max\{\|\mathbf{U}_l^t\|_2, \|\mathbf{b}_l^t\|\} \leq B_{Ub}, \quad \|\mathbf{W}_{L+1}^t\|_2 \leq B_L, \quad \|\mathbf{X}^t\|_F \leq B_x \\ \sigma_{\min}(\mathbf{W}_l^t) \geq \sigma_m, \quad \sigma_{\min}(\mathbf{W}_{L+1}^t) \geq \frac{1}{2}\sigma_{\min}(\mathbf{W}_{L+1}^0) \\ \ell(\widetilde{\boldsymbol{\theta}}^t) \leq (1 - \eta Q_0)^t\ell(\widetilde{\boldsymbol{\theta}}^0). \end{cases} \tag{34}$$

By the initial condition, Eq.(34) holds for $t = 0$ clearly. We now suppose that Eq.(34) holds for all iterations from 0 to $\widetilde{t}$, and show the claim for iteration $\widetilde{t} + 1$. Note that for every $l \in [1, L]$ and $t \in [1, \widetilde{t}]$, we have:

$$\begin{aligned} &\|\mathbf{W}_l^{t+1} - \mathbf{W}_l^0\|_2 \leq \sum_{i=1}^{t}\left\|\mathbf{W}_l^{i+1} - \mathbf{W}_l^i\right\|_2 \leq \eta\sum_{i=1}^{t}\|\mathrm{vec}\left(\nabla_{\mathbf{W}_l}\ell(\boldsymbol{\theta}^i)\right)\| \\ \leq&\eta K\alpha C_zB_L\sum_{i=1}^{t}(1 - \eta Q_0)^{i/2}\|\mathbf{y}^0 - \mathbf{y}_0\| \leq \frac{1}{Q_0}\alpha KC_zB_L(1 - s^2)\frac{1}{1 - s}\|\mathbf{y}^0 - \mathbf{y}_0\| \\ \leq&\frac{2}{Q_0}\alpha KC_zB_L\|\mathbf{y}^0 - \mathbf{y}_0\| \leq \frac{1}{4}, \end{aligned}$$

where we use the bound in Eq.(24) and let $s := (1 - \eta Q_0)^{1/2}$, the last inequality comes from the initial condition on Q_0, see Eq.(33). Similarly, when the data size is large enough, we can also have:

$$\begin{aligned}
\left\|\mathbf{U}_l^{t+1} - \mathbf{U}_l^0\right\|_2 &\leq \frac{2}{Q_0}\alpha K B_x B_L \left\|\mathbf{y}^0 - \mathbf{y}_0\right\| < 1/3 \leq \frac{B_{Ub}}{2}, \\
\left\|\mathbf{b}_l^{t+1} - \mathbf{b}_l^0\right\|_2 &\leq \frac{2}{Q_0}\alpha K \sqrt{N} B_L \left\|\mathbf{y}^0 - \mathbf{y}_0\right\| < 1/3 \leq \frac{B_{Ub}}{2}.
\end{aligned}$$

And by Eq.(26) and Eq.(28)

$$\begin{aligned}
&\left\|\mathbf{x}^{t+1} - \mathbf{x}^0\right\| \leq L_g \left\|\mathbf{W}_0^{t+1} - \mathbf{W}_0^0\right\|_F \leq \frac{2}{Q_0}\alpha K L B_{Ub} B_L L_g^2 \left\|\mathbf{y}^0 - \mathbf{y}_0\right\| < \frac{B_x}{2}, \\
&\left\|\mathbf{W}_{L+1}^{t+1} - \mathbf{W}_{L+1}^0\right\|_2 \leq \frac{2}{Q_0} 3\gamma K B^* \left\|\mathbf{y}^0 - \mathbf{y}_0\right\| < \frac{1}{2}\sigma_{\min}(\mathbf{W}_{L+1}^0).
\end{aligned}$$

Thus by Weyl's inequality, we obtain:

$$\begin{cases}
\left\|\mathbf{W}_l^{t+1}\right\|_2 \leq 1, \quad \max\left\{\left\|\mathbf{U}_l^{t+1}\right\|_2, \left\|\mathbf{b}_l^{t+1}\right\|\right\} \leq B_{Ub}, \quad \left\|\mathbf{W}_{L+1}^{t+1}\right\|_2 \leq B_L, \quad \left\|\mathbf{X}^{t+1}\right\|_F \leq B_x \\
\sigma_{\min}\left(\mathbf{W}_l^t\right) \geq \sigma_m, \quad \sigma_{\min}(\mathbf{W}_{L+1}^{t+1}) \geq \frac{1}{2}\sigma_{\min}(\mathbf{W}_{L+1}^0).
\end{cases}$$

We now provide a Lipschitz constant for the gradient of loss function. Given two parameters $\widetilde{\boldsymbol{\theta}}^a$ and $\widetilde{\boldsymbol{\theta}}^b$ such that satisfies Assumption 3 and has the bounds $\ell(\widetilde{\boldsymbol{\theta}}^a) \leq \ell(\widetilde{\boldsymbol{\theta}}^0)$ and $\ell(\widetilde{\boldsymbol{\theta}}^b) \leq \ell(\widetilde{\boldsymbol{\theta}}^0)$, we have

$$\begin{aligned}
&\left\|\nabla\ell(\widetilde{\boldsymbol{\theta}}^a) - \nabla\ell(\widetilde{\boldsymbol{\theta}}^a)\right\| \\
=&\left\|\mathcal{J}_{\mathbf{z}^K}^\top(\widetilde{\boldsymbol{\theta}}^a)\left(\mathbf{I}_N \otimes \left(\mathbf{W}_{L+1}^a\right)^\top\right)(\mathbf{y}^a - \mathbf{y}_0) - \mathcal{J}_{\mathbf{z}^K}^\top(\widetilde{\boldsymbol{\theta}}^b)\left(\mathbf{I}_N \otimes \left(\mathbf{W}_{L+1}^b\right)\right)\left(\mathbf{y}^b - \mathbf{y}_0\right)\right\| \\
\leq&\left\|\mathcal{J}_{\mathbf{z}^K}(\widetilde{\boldsymbol{\theta}}^a) - \mathcal{J}_{\mathbf{z}^K}(\widetilde{\boldsymbol{\theta}}^b)\right\|_2 \sqrt{\ell(\widetilde{\boldsymbol{\theta}}^0)} + B_J\left\|\mathbf{W}_{L+1}^a - \mathbf{W}_{L+1}^b\right\|\sqrt{\ell(\widetilde{\boldsymbol{\theta}}^0)} + B_J B_L^2\left\|\mathbf{z}^K(\widetilde{\boldsymbol{\theta}}^a) - \mathbf{z}^K(\widetilde{\boldsymbol{\theta}}^b)\right\| \\
\leq&\left(C_J\sqrt{\ell(\widetilde{\boldsymbol{\theta}}^0)} + 2\alpha\sqrt{L}KC_z L_g B_J B_L + B_J\sqrt{\ell(\widetilde{\boldsymbol{\theta}}^0)}\right)\left\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\right\| = Q_1\left\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\right\|.
\end{aligned}$$

where we use the bound in Eq.(31) in the second and third inequality, and the bound from Eq.(23), Eq.(23) and Eq.(28) for

$$\left\|\mathcal{J}_{\mathbf{z}^K}\right\| \leq \left(3LK^2\alpha^2C_z^2 + \alpha^2K^2L^2B_{Ub}^2 + 4K^2\gamma^2(B^*)^2B_L^2\right)^{\frac{1}{2}} \leq 2\alpha\sqrt{L}KC_z := B_J.$$

When $\eta \leq 1/Q_1$, the Lipschitz bound $\left\|\nabla\ell(\widetilde{\boldsymbol{\theta}}^a) - \nabla\ell(\widetilde{\boldsymbol{\theta}}^a)\right\| \leq Q_1\left\|\widetilde{\boldsymbol{\theta}}^a - \widetilde{\boldsymbol{\theta}}^b\right\|$ implies that:

$$\begin{aligned}
&\ell(\widetilde{\boldsymbol{\theta}}^{t+1}) \leq \ell(\widetilde{\boldsymbol{\theta}}^t) + \left\langle\nabla\ell(\widetilde{\boldsymbol{\theta}}^t), \widetilde{\boldsymbol{\theta}}^{t+1} - \widetilde{\boldsymbol{\theta}}^t\right\rangle + \frac{Q_1}{2}\left\|\widetilde{\boldsymbol{\theta}}^{t+1} - \widetilde{\boldsymbol{\theta}}^t\right\|^2 \\
\leq&\ell(\widetilde{\boldsymbol{\theta}}^t) - \eta\left\|\nabla\ell(\widetilde{\boldsymbol{\theta}}^t)\right\|^2 + \frac{Q_1}{2}\eta^2\left\|\nabla\ell(\widetilde{\boldsymbol{\theta}}^t)\right\|^2 \\
\leq&\ell(\widetilde{\boldsymbol{\theta}}^t) - \frac{\eta}{2}\left\|\nabla\ell(\widetilde{\boldsymbol{\theta}}^t)\right\|^2 \leq \ell(\widetilde{\boldsymbol{\theta}}^t) - \frac{\eta}{2}\left(K^2L\alpha^2\kappa^2\sigma_m^2N\sigma_{\min}^2(\mathbf{W}_{L+1}^t)\left\|\mathbf{y}^t - \mathbf{y}_0\right\|^2\right) \\
\leq&\left(1 - \frac{\eta}{4}K^2L\alpha^2\kappa^2\sigma_m^2N\sigma_{\min}^2(\mathbf{W}_{L+1}^0)\right)\ell(\widetilde{\boldsymbol{\theta}}^t) = (1 - \eta Q_0)\ell(\widetilde{\boldsymbol{\theta}}^t),
\end{aligned}$$

where the third inequality comes from the fact Eq.(25) and recall that

$$Q_0 := \frac{1}{4}K^2L\alpha^2\kappa^2\sigma_m^2N\sigma_{\min}^2(\mathbf{W}_{L+1}^0).$$

So far, we have proven the hypothesis in Eq.(34).

We start to show that the sequence $\{\widetilde{\boldsymbol{\theta}}^t\}_{t=1}^{\infty}$ is a Cauchy sequence. Given any $\epsilon > 0$ and the index $r > 0$, we chose two indices $j > i \geq r$. Then, we have:

$$
\begin{aligned}
&\left\|\widetilde{\boldsymbol{\theta}}^i - \widetilde{\boldsymbol{\theta}}^j\right\| \leq \left\|\mathbf{W}^i_{L+1} - \mathbf{W}^j_{L+1}\right\|_F + \left\|\boldsymbol{\theta}^i - \boldsymbol{\theta}^j\right\| + \left\|\mathbf{W}^i_0 - \mathbf{W}^j_0\right\|_F \\
\leq &\eta \sum_{s=i}^{j-1} \eta \Big(\left\|\mathbf{W}^{s+1}_{L+1} - \mathbf{W}^s_{L+1}\right\|_F + \left\|\boldsymbol{\theta}^{s+1} - \boldsymbol{\theta}^s\right\| + \left\|\mathbf{W}^{s+1}_0 - \mathbf{W}^s_0\right\|_F \Big) \\
\leq &\sum_{s=i}^{j-1} \eta \left(\left\|\mathbf{W}^{s+1}_{L+1} - \mathbf{W}^s_{L+1}\right\|_F + \sum_{l=1}^{L} \left(\left\|\mathbf{W}^{s+1}_l - \mathbf{W}^s_l\right\| + \left\|\mathbf{U}^{s+1}_l - \mathbf{U}^s_l\right\| + \left\|\mathbf{b}^{s+1}_l - \mathbf{b}^s_l\right\| \right) + \left\|\mathbf{W}^{s+1}_0 - \mathbf{W}^s_0\right\|_F \right) \\
\overset{(a)}{\leq} &\sum_{s=i}^{j-1} \eta \left\|\mathbf{y}^s - \mathbf{y}^0\right\| (3C_z B_L + B_{Ub} L_g B_L + 3KB^*) \\
\leq &(1 - \eta Q_0)^{i/2} \left(\sum_{s=0}^{j-i-1} (1 - \eta Q_0)^{s/2} \left\|\mathbf{y}^0 - \mathbf{y}_0\right\| \right) \eta (3C_z B_L + B_{Ub} L_g B_L + 3KB^*) \\
\overset{(b)}{=} &(1 - \eta Q_0)^{i/2} (3C_z B_L + B_{Ub} L_g B_L + 3KB^*) \left(\frac{1}{Q_0} (1 - s^2) \frac{1 - s^{j-i}}{1 - s} \right) \left\|\mathbf{y}^0 - \mathbf{y}_0\right\| \\
\leq &(1 - \eta Q_0)^{i/2} \frac{2}{Q_0} (3C_z B_L + B_{Ub} L_g B_L + 3KB^*) \left\|\mathbf{y}^0 - \mathbf{y}_0\right\|,
\end{aligned}
$$

where (a) comes from Eq.(24), Eq.(27) and Eq.(29) and the assumption $\alpha K L < 1$, and in (b) we set $s = \sqrt{1 - \eta Q_0}$. Note that $(1 - \eta Q_0)^{i/2} \leq (1 - \eta Q_0)^{r/2}$ and thus we can select a sufficiently large r such that $\left\|\widetilde{\boldsymbol{\theta}}^i - \widetilde{\boldsymbol{\theta}}^j\right\| \leq \epsilon$. Hence, we can conclude that $\{\widetilde{\boldsymbol{\theta}}^t\}_{t=1}^{\infty}$ is a Cauchy sequence, and thus has a convergent point $\widetilde{\boldsymbol{\theta}}^*$. Due to the continuity, we have:

$$
\ell(\widetilde{\boldsymbol{\theta}}^*) = \ell(\lim_{t \to \infty} \widetilde{\boldsymbol{\theta}}^t) = \lim_{t \to \infty} \ell(\widetilde{\boldsymbol{\theta}}^t) = 0,
$$

where the last equality comes from Eq.(34). Hence, $\widetilde{\boldsymbol{\theta}}^*$ is a global minimizer, and the rate of convergence is:

$$
\left\|\widetilde{\boldsymbol{\theta}}^i - \widetilde{\boldsymbol{\theta}}^*\right\| = \lim_{j \to \infty} \left\|\widetilde{\boldsymbol{\theta}}^i - \widetilde{\boldsymbol{\theta}}^j\right\| \leq (1 - \eta Q_0)^{i/2} Q_2,
$$

note that

$$
\frac{2}{Q_0} (3C_z B_L + B_{Ub} L_g B_L + 3KB^*) \left\|\mathbf{y}^0 - \mathbf{y}_0\right\| \leq \frac{10 C_z B_L L_g}{Q_0} := Q_2.
$$

We now finish the whole proof. □