Spherical Motion Dynamics: Learning Dynamics of Normalized Neural Network using SGD and Weight Decay

Anonymous Author(s)
Affiliation
Address
email

Abstract

In this paper, we comprehensively reveal the learning dynamics of normalized neural network using Stochastic Gradient Descent (with momentum) and Weight Decay (WD), named as Spherical Motion Dynamics (SMD). Most related works focus on studying behavior of "effective learning rate" in "equilibrium" state, i.e. assuming weight norm remains unchanged. However, their discussion on why this equilibrium can be reached is either absent or less convincing. Our work directly explores the cause of equilibrium, as a special state of SMD. Specifically, 1) we introduce the assumptions that can lead to equilibrium state in SMD, and prove equilibrium can be reached in a linear rate regime under given assumptions; 2) we propose "angular update" as a substitute for effective learning rate to depict the state of SMD, and derive the theoretical value of angular update in equilibrium state; 3) we verify our assumptions and theoretical results on various large-scale computer vision tasks including ImageNet and MSCOCO with standard settings. Experiment results show our theoretical findings agree well with empirical observations. We also show that the behavior of angular update in SMD can produce significant effect to the optimization of neural network in practice.

1 Introduction

Normalization techniques (e.g. Batch Normalization (Ioffe & Szegedy, 2015) or its variants) are one of the most commonly adopted techniques for training deep neural networks (DNN). A typical normalization can be formulated as following: consider a single unit in a neural network, the input is $\boldsymbol{X}$, the weight of linear layer is $\boldsymbol{w}$ (bias is included in $\boldsymbol{w}$), then its output is

$$y(\boldsymbol{X};\boldsymbol{w};\gamma;\beta) = g(\frac{\boldsymbol{X}\boldsymbol{w} - \mu(\boldsymbol{X}\boldsymbol{w})}{\sigma(\boldsymbol{w}\boldsymbol{X})}\gamma + \beta), \tag{1}$$

where g is a nonlinear activation function like ReLU or sigmoid, μ, σ are mean and standard deviation computed across specific dimension of $\boldsymbol{X}\boldsymbol{w}$ (like Batch Normalization (Ioffe & Szegedy, 2015), Layer Normalization Ba et al. (2016), Group Normalization (Wu & He, 2018), etc.). β, γ are learnable parameters to remedy for the limited range of normalized feature map. Aside from normalizing feature map, Salimans & Kingma (2016) normalizes weight by l^2 norm instead:

$$y(\boldsymbol{X};\boldsymbol{w};\gamma;\beta) = g(\boldsymbol{X}\frac{\boldsymbol{w}}{||\boldsymbol{w}||_2}\gamma + \beta), \tag{2}$$

where $||\cdot||_2$ denotes l_2 norm of a vector. Though formulated in different manners, all normalization techniques mentioned above share an interesting property: *scale-invariant*

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

Definition 1 (Scale-invariance)**.** *Given loss function $\mathcal{L}(\boldsymbol{w})$, $\boldsymbol{w}$ is scale-invariant w.r.t. $\mathcal{L}$ if and only if $\forall k \in \mathbb{R}^+$, we have $\mathcal{L}(\boldsymbol{w}) = \mathcal{L}(k\boldsymbol{w})$.*

By definition of scale-invariant property, we can directly derive the following properties of scale-invariant weights in Lemma 1

Lemma 1. *If $\boldsymbol{w}$ is scale-invariant with respect to $\mathcal{L}(\boldsymbol{w})$, then for all $k > 0$, we have:*

$$\langle \boldsymbol{w}_t, \frac{\partial \mathcal{L}}{\partial \boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}_t} \rangle = 0 \tag{3}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}}\Big|_{\boldsymbol{w}=k\boldsymbol{w}_t} = \frac{1}{k} \cdot \frac{\partial \mathcal{L}}{\partial \boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}_t}. \tag{4}$$

Proof is in appendix. Lemma 1 is also discussed in Hoffer et al. (2018); van Laarhoven (2017); Li & Arora (2020); Li et al. (2020), it makes the learning dynamics of normalized neural network exhibit an interesting phenomenon when using Stochastic Gradient Descent (SGD) with Weight Decay (WD): a typical SGD update rule with WD is

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta(\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}_t} + \lambda \boldsymbol{w}_t) = (1-\eta\lambda)\boldsymbol{w}_t - \eta \frac{\partial \mathcal{L}}{\partial \boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}_t}, \tag{5}$$

where η denotes learning rate, λ denotes WD factor. Then dynamics of $\boldsymbol{w}_t$ is like a physical process – Spherical Motion (see illustration in Fig.1): due to Eq.(3), $-\eta \partial \mathcal{L}/\partial \boldsymbol{w}\big|_{\boldsymbol{w}=\boldsymbol{w}_t}$ (green line in Fig.1) is always perpendicular to $\boldsymbol{w}_t$, providing "centrifugal effect" to make $||\boldsymbol{w}_{t+1}||_2$ larger than $||\boldsymbol{w}_t||_2$; while $-\eta\lambda\boldsymbol{w}_t$ (red line in Fig.1) is always in the opposite direction of $\boldsymbol{w}_t$, providing "centripetal effect" to make $||\boldsymbol{w}_{t+1}||_2$ smaller than $||\boldsymbol{w}_t||_2$. Because of this "tug of war" phenomenon between "centrifugal effect" and "centripetal effect", we formally call the learning dynamics of normalized neural network using SGD(M) and WD as *Spherical Motion Dynamics* (SMD) in this paper.

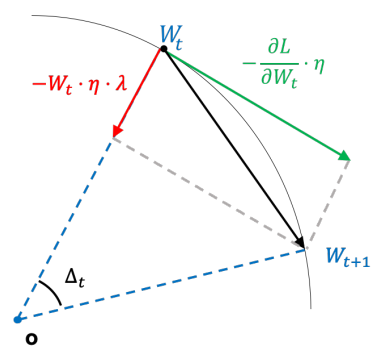


Figure 1: Illustration of optimization behavior with BN and WD. Angular update Δ_t represents the angle between the updated weight $\boldsymbol{w}_t$ and its former value $\boldsymbol{w}_{t+1}$.

Concept of "Equilibrium" Since "tug of war" in SMD influences the relative sizes of $||\boldsymbol{w}_{t+1}||_2$ and $||\boldsymbol{w}_t||_2$, a question naturally arises: what will happen if $||\boldsymbol{w}_{t+1}||_2 = ||\boldsymbol{w}_t||_2$? van Laarhoven (2017) discuss this question first; Chiley et al. (2019) named this state $||\boldsymbol{w}_{t+1}||_2 = ||\boldsymbol{w}_t||_2$ in SMD as "*equilibrium*", and discuss its properties; Li & Arora (2020) derives a lemma about equilibrium in SGD with Momentum (SGDM). Early literatures (van Laarhoven, 2017; Chiley et al., 2019; Li & Arora, 2020) do not discuss a crucial question: "Can equilibrium be reached in SMD?" van Laarhoven (2017) intuitively explains that equilibrium is caused by convergence of optimization. But there exists a contradiction between the interpretation of van Laarhoven (2017) and traditional view of optimization: if equilibrium ($||\boldsymbol{w}_t||_2 = ||\boldsymbol{w}_{t+1}||_2$) is caused by convergence of optimization, then gradient of loss $\partial \mathcal{L}/\partial \boldsymbol{w}\big|_{\boldsymbol{w}=\boldsymbol{w}_t}$ should be equal to $\mathbf{0}$, which makes the balance of "centrifugal effect" and "centripetal effect" impossible to reach (since centripetal effect comes from $\partial \mathcal{L}/\partial \boldsymbol{w}\big|_{\boldsymbol{w}=\boldsymbol{w}_t}$). Therefore, van Laarhoven (2017); Chiley et al. (2019); Li & Arora (2020) all essentially regard equilibrium as an assumption, and do not justify its existence in neither empirical nor theoretical aspects. "Equilibrium" was not even a phenomenon observed in practice, but only a concept until recently.

Recent work (Li et al., 2020) successfully exhibits the existence of equilibrium by formulating SGD in Eq.(5) via a Stochastic Differential Equation (SDE) in the continuous time limit. They theoretically prove equilibrium can be reached in SDE settings: the convergence of $||\boldsymbol{w}||_t$ is only driven by Brownian motion; the mixing time is $\mathcal{O}(1/(\lambda\eta))$ (λ, η denote WD factor and learning rate respectively). However, due to the gap between discrete formulation of SGD and continuous formulation of SDE,

theoretical results derived from SDE model can only provide intuitive understanding on empirical observations. Besides, SDE can hardly take SGD with momentum (Polyak, 1964) into account, which has become default setting in nearly all kinds of deep learning tasks. In summary, a thorough understanding on cause of "equilibrium" and its impact to learning dynamics of normalized neural network is still needed.

In this paper, we comprehensively reveal Spherical Motion Dynamics (SMD), i.e. the learning dynamics of normalized neural network using SGD(M) and weight decay (WD). Our analysis on SMD is directly established on discrete settings. We interpret why equilibrium can be reached in SMD in both theoretical and empirical aspects, and show how SMD affects the optimization trajectory of neural network. Specifically, our contributions are

- We introduce the assumptions which can lead to equilibrium in SMD, and justify their reasonableness by sufficient experiments. We also prove under given assumptions, equilibrium can be reached as weight norm approach to its theoretical value in a linear rate regime. Our theorem show equilibrium is a dynamic state in SMD, norm weight is unnecessary to be steady within equilibrium state;
- We define a novel index, *angular update*, to measure the change of normalized neural network within a single iteration. We also derive its theoretical value in equilibrium. Our results show that angular update is better than norm of weight to indicate if equilibrium has been reached reached in SMD. Our empirical results further show angular update is an important index to reflect the effect of SMD and equilibrium;
- We verify our theorems on different computer vision tasks (including one of most challenging datasets ImageNet (Russakovsky et al., 2015) and MSCOCO (Lin et al., 2014)) with various networks structures. Experiments show the theoretical value of angular update and weight norm agree well with empirical observation. We also show how SMD influence the optimization trajectory of normalized neural network by controlling angular update.

Our theorem on equilibrium implies equilibrium is a special state of SMD which only relies on the update rules of SGD/SGDM with WD, and scale-invariant property. The cause of equilibrium is independent of optimization trajectory, but equilibrium significantly affects update efficiency of normalized network in turn by controlling angular update. We believe SMD is one of the key reason why learning dynamics of normalized neural network is not consistent with traditional optimization theory (Li et al., 2020). We think it is of great potential to take SMD and its equilibrium state into account while studying leaning dynamics of modern normalized neural network or designing novel efficient training strategy.

2 Related work

This paper mainly discuss Spherical Motion Dynamics, i.e. joint effect of normalization and weight decay when training neural network work. Normalization techniques and weight decay are both relevant topics which should be carefully reviewed. But due to the limitation of the length, we have to leave the reviews on normalization techniques and weight decay separately in appendix. Here we only review the previous works focusing on joint effect of normalization and weight decay.

Since the scale invariant property caused by normalization makes euclidean metrics of weight meaningless, researchers start to study the behavior of effective learning rate. van Laarhoven (2017); Chiley et al. (2019) estimate the magnitude of effective learning rate under equilibrium assumptions in SGD case; Hoffer et al. (2018) quantify effective learning rate without equilibrium assumption; Arora et al. (2019) proves that without WD, normalized neural network still can converge using fixed/decaying learning rate in Gradient Descent(GD)/SGD cases respectively; Zhang et al. (2019) shows WD can increase effective learning rate; Li & Arora (2020) proves standard multi-stage learning rate schedule with BN and WD is equivalent to an exponential increasing learning rate schedule without WD. As a proposition, Li & Arora (2020) quantifies the magnitude of effective learning rate in SGDM case. But none of them have ever discussed why equilibrium condition can be reached. A recent work Li et al. (2020) studies the convergence of effective learning rate by SDE, proving that the convergence time is of $\mathcal{O}(1/(\lambda\eta))$, where λ, η are weight decay factor and learning rate respectively. Kunin et al. (2021) also depicts the equilibrium state by gradient flow.

3 Theoretical results

In this section, we theoretically formulate Spherical Motion Dynamics (SMD) in discrete SGD/SGDM settings, and provide a precise description on "equilibrium" phenomenon. First we prove equilibrium can be reached in SMD under specific assumptions. Then, we propose a new index to indicate the state of SMD, and derive its theoretical value in equilibrium.

First of all, we give a new definition. Eq.(4) implies though norm of scale-invariant weights does not affect the output of neural network, it can influence norm of gradients, thus we define *unit gradient* in order to eliminate the effect of weight norm makes to gradient norm.

Definition 2 (*Unit Gradient*)**.** *If $\boldsymbol{w}_t \neq \mathbf{0}$, $\tilde{\boldsymbol{w}} = \boldsymbol{w}/||\boldsymbol{w}||_2$, the unit gradient of $\partial\mathcal{L}/\partial\boldsymbol{w}|_{\boldsymbol{w}=\boldsymbol{w}_t}$ is $\partial\mathcal{L}/\partial\boldsymbol{w}|_{\boldsymbol{w}=\tilde{\boldsymbol{w}}_t}$.*

According to the definition of unit gradient, the unit gradient norm is independent of weight norm. Specifically, by setting k as $1/||\boldsymbol{w}_t||_2$ in Eq.(4), the relation among weight norm, gradient and unit gradient is

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{w}}\Big|_{\boldsymbol{w}=\boldsymbol{w}_t} = \frac{1}{||\boldsymbol{w}_t||} \cdot \frac{\partial\mathcal{L}}{\partial\boldsymbol{w}}\Big|_{\boldsymbol{w}=\tilde{\boldsymbol{w}}_t}. \tag{6}$$

Now, we can depict equilibrium of SGD and SGDM in theorem 1, 2 respectively.

Theorem 1. ***(Equilibrium in SGD)*** *Assume the loss function is $\mathcal{L}(\boldsymbol{X};\boldsymbol{w})$ with scale-invariant weight $\boldsymbol{w}$, denote $\boldsymbol{g}_t = \frac{\partial\mathcal{L}}{\partial\boldsymbol{w}}\big|_{\boldsymbol{X}_t,\boldsymbol{w}_t}$, $\tilde{\boldsymbol{g}}_t = \boldsymbol{g}_t \cdot ||\boldsymbol{w}_t||_2$. Consider the update rule of SGD with weight decay,*

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta \cdot (\boldsymbol{g}_t + \lambda\boldsymbol{w}_t) \tag{7}$$

where $\lambda, \eta \in (0,1)$. If the following assumptions hold:

1) $\lambda\eta \ll 1$ ($o(\lambda\eta)$ can be omitted);

2) Let $L_t = \mathbb{E}||\tilde{\boldsymbol{g}}_t||_2^2$. $\exists V \in \mathbb{R}^+$, $\forall t \in \mathbb{N}^+$, $\mathbb{E}[(||\tilde{\boldsymbol{g}}_t||_2^2 - L_t)^2|\boldsymbol{w}_t] \leq V$;

3) $\forall t \in \mathbb{N}^+$, L_t satisfies $|L_{t+1} - L_t| < 4\sqrt{V}(\lambda\eta)^{3/2}$;

4) $\exists l \in \mathbb{R}^+$, $\forall t \in \mathbb{N}^+$, $||\tilde{\boldsymbol{g}}_t||_2^2 > l$, $l > 2[\frac{2\lambda\eta}{1-2\lambda\eta}]^2 L_t$.

Then $\exists B > 0$, $\forall t \in \mathbb{N}^+$, $w_t^* = \sqrt[4]{L_t\eta/(2\lambda)}$, we have

$$\mathbb{E}[||\boldsymbol{w}_t||_2^2 - (w_t^*)^2]^2 \leq (1-2\lambda\eta)^t B + \frac{2V\eta^2}{l(1-2\lambda\eta)}. \tag{8}$$

Remark 1. *The theoretical value of weight norm w_t^* in Theorem 1 is consistent with the magnitude of weight norm ($\mathcal{O}(\sqrt[4]{\eta/\lambda})$) in equilibrium in van Laarhoven (2017), though van Laarhoven (2017) assumes the equilibrium has been reached in advance, hence van Laarhoven (2017) cannot provide the approaching rate and scale of bias/variance. The vanishing term ($(1-2\lambda\eta)^t B$) in Eq.(8) is consistent with the mixing time $\mathcal{O}(1/(\lambda\eta))$ presented in Li et al. (2020).*

The proof can be seen in appendix. Assumption 1 is consistent with commonly used settings in practice (Goyal et al., 2017; He et al., 2017; Ma et al., 2018); Assumptions 2, 3, 4 all concern unit gradient: unit gradient norm should change smoothly (assumption 3) with bounded variance (assumption 2); besides, unit gradient norm should have a lower bound (assumption 4). We will see these assumptions can easily hold in practice in section 4.1.

True meaning of "equilibrium": a dynamic state of SMD Recall as we demonstrate in introduction, the concept of equilibrium is originally established on the assumption that weight norm is steady ($||\boldsymbol{w}_t||_2 = ||\boldsymbol{w}_{t+1}||_2$). But the assumption ($||\boldsymbol{w}_t||_2 = ||\boldsymbol{w}_{t+1}||_2$) is unrealistic due to the complex dynamics of training process and the variance of stochastic gradients. Now theorem 1 provides a realistic meaning of equilibrium in SGD settings: equilibrium is just a dynamic state of SMD, meaning $||\boldsymbol{w}_t||_2^2$ oscillates around the theoretical value $(w_t^*)^2$ determined by hyperparameters and unit gradient norm. Its variance is bounded by $2V\eta^2/[l(1-2\lambda\eta)]$, which is relatively small comparing with $(w_t^*)^4$ because

$$\frac{2V\eta^2}{l(1-2\lambda\eta)}/(w_t^*)^4 = \frac{4V\lambda\eta}{L_t l(1-2\lambda\eta)} = \mathcal{O}(\lambda\eta) \ll 1. \tag{9}$$

Note besides the stochastic behavior of $||\boldsymbol{w}_t||_2^2$, the "dynamic state" also reflects in the variation of the theoretical value $(w_t^*)^2$. Because $(w_t^*)^2$ is determined by L_t, which is allowed to change smoothly across the whole training process in assumption 2 (See more discussion in appendix). In summary, the sign of equilibrium is neither the convergence of weight norm $||\boldsymbol{w}_t||_2^2$ (van Laarhoven, 2017; Chiley et al., 2019) nor the convergence of $||\boldsymbol{w}_t||_2$ in expectation (Li et al., 2020). The real sign of equilibrium is whether $\mathbb{E}||\boldsymbol{w}_t||_2^2$ is close to its theoretical value $(w_t^*)^2$.

Theorem 1 also shows the dynamic equilibrium can be reached in a linear rate regime when vanishing term is larger than constant term in Eq.(8). The approaching rate is only determined by predefined parameters λ, η. Moreover, based on the proof of theorem 1, the cause of equilibrium is independent of optimization process at all, which implies the possibility that equilibrium can be reached long before the convergence of loss function.

Now we extend theorem 1 to momentum case. SGDM is more complex than SGD since momentum is not always perpendicular to the weight, hence we need to modify assumptions.

Theorem 2. ***(Equilibrium in SGDM)*** *Considering the update rule of SGDM (heavy ball method (Polyak, 1964)):*

$$\boldsymbol{v}_t = \alpha \boldsymbol{v}_{t-1} + \boldsymbol{g}_t + \lambda \boldsymbol{w}_t \tag{10}$$

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta \boldsymbol{v}_t \tag{11}$$

where $\lambda, \eta \in (0,1), \alpha \in (\frac{1}{2}, 1)$. If following assumptions hold:

5) $\lambda\eta \ll 1$, $\lambda\eta < (1-\sqrt{\alpha})^2$;

6) Define $h_t = ||\boldsymbol{g}_t||_2^2 + 2\alpha\langle \boldsymbol{v}_{t-1}, \boldsymbol{g}_t\rangle$, $\tilde{h}_t = h_t \cdot ||\boldsymbol{w}_t||_2^2$, $L_t = \mathbb{E}\tilde{h}$. $\exists V \in \mathbb{R}^+$, $\forall t \in \mathbb{N}^+$, $\mathbb{E}[(\tilde{h}_t - L_t)^2 | \boldsymbol{w}_t] \leq V$;

7) $\forall t \in \mathbb{N}^+$, L_t satisfies $|L_{t+1} - L_t| < 4\sqrt{V}(\lambda\eta)^{3/2}$;

8) $\exists l \in \mathbb{R}^+, \forall t \in \mathbb{N}^+, \tilde{h}_t > l > 2[\frac{6\lambda\eta}{(1-\alpha)^3(1+\alpha)-8\lambda\eta(1-\alpha)}]^2 L_t$, ;

then $\exists B, C > 0$, C only depends on α, $w^* = \sqrt[4]{L_t\eta/(\lambda(1-\alpha)(2-\lambda\eta/(1+\alpha)))}$, we have

$$\mathbb{E}[||\boldsymbol{w}_t||_2^2 - (w_t^*)^2]^2 \leq (1 - \frac{2\lambda\eta}{1-\alpha})^t B + \frac{V\eta^2}{l} C, \tag{12}$$

Remark 2. *So far, no other work rigorously prove equilibrium can be reached in SGDM. The most relevant work (Li et al., 2020) only provides a conjecture on convergence rate of weight norm in SGDM. By regarding SGDM as SGD with larger learning rate, they guess that the mixing time to reach equilibrium in SGDM case should be $\mathcal{O}(1/(\lambda\eta))$, same order as mixing time in SGD case. Their conjecture cannot provide further insight on difference between SGD and SGDM. While our results (vanishing terms in Eq.(8), (12) respectively) clearly reflect the difference: the approaching rate of SGDM should be $1/(1-\alpha)$ times larger than rate of SGD with same $\eta\lambda$. α is usually set as 0.9 in practice, hence SGDM can reach equilibrium condition much faster than SGD.*

Proof can be seen in appendix. Like assumption 1, assumption 5 also holds for commonly used hyperparameter settings; Assumption 6, 7, 8 concerns not unit gradient norm $||\tilde{\boldsymbol{g}}_t||_2^2$ but an adjusted value $\tilde{h}_t$ which dominates the expectation and variance of $||\boldsymbol{w}_t||_2^2$. We empirically find the expectation of $\langle \boldsymbol{v}_{t-1}, g_t\rangle$ is very close to 0, therefore the behavior of $\tilde{h}_t$ is similar to that of $||\tilde{\boldsymbol{g}}_t||_2^2$ (see Figure 2(d)). We leave theoretical analysis on $\tilde{h}_t$ as future work. The experiments on justification of assumptions 6, 7, 8 can be seen in Figure 2. Comparing with Eq.(8) and Eq.(12), we can infer with same η, λ, SGDM can reach equilibrium state much faster than SGD, but it may have a larger variance, our experiments also verify our claim (see Figure 2(b), 2(e)).

We have derived the theoretical value of weight norm in equilibrium, it allows us to check if equilibrium has been reached in practice. But the theoretical value of weight norm still relies on the expectation of unit gradient norm, which is not easy to compute in practice. Besides, weight norm is of little value for studying normalized models since their weight is scale-invariant. Hence we introduce an new and meaningful index, *angular update*, to reflect the effect of SMD and equilibrium.

Definition 3 (*Angular Update*)**.** *Let $\boldsymbol{w}_t$ denote a scale-invariant weight from a neural network at iteration t, then* ***angular update*** *Δ_t is defined as*

$$\Delta_t = \measuredangle(\boldsymbol{w}_t, \boldsymbol{w}_{t+1}) = arccos\left(\frac{\langle \boldsymbol{w}_t, \boldsymbol{w}_{t+1}\rangle}{||\boldsymbol{w}_t|| \cdot ||\boldsymbol{w}_{t+1}||}\right), \tag{13}$$

where $\measuredangle(\cdot,\cdot)$ denotes the angle between two vectors, $\langle\cdot,\cdot\rangle$ denotes the inner product.

Angular update has a concrete geometric meaning (see illustration in Figure 1): it is exactly the geodesic distance between $\tilde{\boldsymbol{w}}_t$ and $\tilde{\boldsymbol{w}}_{t+1}$ on $\mathcal{S}^{p-1}$, where $\tilde{\boldsymbol{w}}_t = \boldsymbol{w}_t/||\boldsymbol{w}_t||_2$, $\tilde{\boldsymbol{w}}_{t+1} = \boldsymbol{w}_{t+1}/||\boldsymbol{w}_{t+1}||_2$. Comparing with the Euclidean distance $||\boldsymbol{w}_{t+1} - \boldsymbol{w}_t||_2$, angular update Δ_t better reflects the effective update of the scale-invariant weight $\boldsymbol{w}_t$ on its intrinsic domain $\mathcal{S}^{p-1}$. Angular update is determined by the relative sizes of gradient norm and weight norm, while the relative sizes of gradient/weight norm are influenced by SMD, hence angular update is strongly affected by SMD. The following theorem exhibits the behavior of angular update when equilibrium is reached in SMD.

Theorem 3. ***(Theoretical value of Angular Update)*** *In SGD(SGDM) case, if assumptions in theorem 1(2) hold, $\eta^2 \ll 1$, t is sufficiently large so that vanishing terms in Eq.*(8)*,* (12) *can be omitted, then with probability at least $1 - \sqrt[3]{\frac{V}{L_t l}}$ we have*

$$|\Delta_t - \sqrt{\frac{2\lambda\eta}{1+\alpha}}| < \mathcal{O}(\sqrt[3]{\frac{V}{L_t l}}). \tag{14}$$

In SGD case, $\alpha = 0$.

Remark 3. *Results of SGD and SGDM case are summarized in Eq.*(14) *in order to highlight the connection between SGD and SGDM. Theoretical value of angular update in Theorem 3 is partially consistent with previous works (Chiley et al., 2019; Li & Arora, 2020; Li et al., 2020; Kunin et al., 2021), detailed discussion is buried in appendix. Note bias term in right side of Eq.*(14) *is of $\mathcal{O}(\sqrt[3]{V/L_t l})$, which is too large comparing with its empirical value (see Figure 2(c), 2(f)), we leave it as a future work to improve the bound in Eq.*(14)*.*

Proof is in appendix. According to theorem 3, the theoretical value of angular update in equilibrium only depends on hyper-parameters: learning rate η, WD factor λ, and momentum factor α. Hence comparing with behavior of weight norm, angular update provides an easier way to check whether equilibrium is reached. Since equilibrium can be reached in a linear rate regime as theorem 1, 2 demonstrate, theorem 3 implies update efficiency of scale-invariant weights within a single step eventually will be determined only by predefined hyperparameters, regardless other attributes of the weights (shape, size, position in network structure, or effects from other weights).

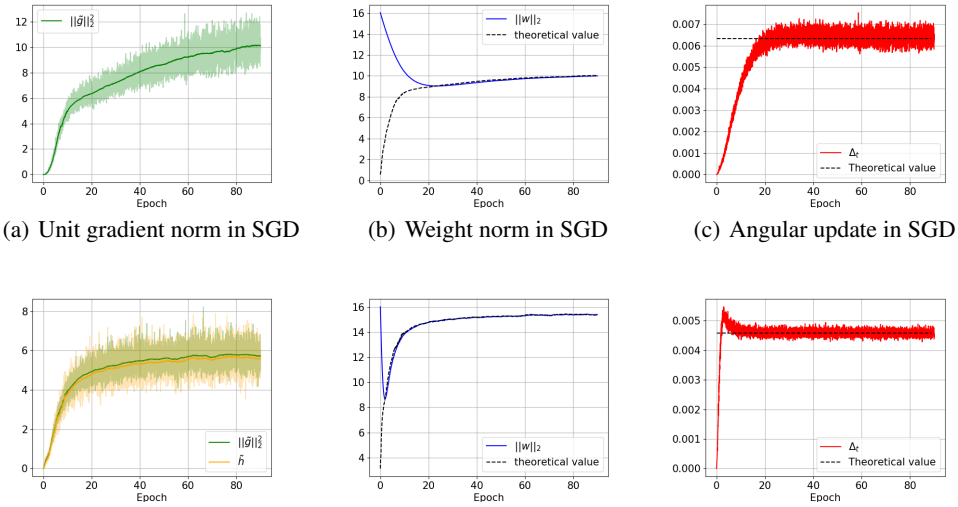
Besides, far beyond just an index indicating whether equilibrium is reached, angular update is also a important way by which SMD can affect the optimization process of normalized neural network. SMD cannot influence the direction of update (like gradient direction in SGD), but it can influence the scale of update $||\tilde{\boldsymbol{w}}_t - \tilde{\boldsymbol{w}}_{t+1}||_2$ on intrinsic domain by controlling the scale of angular update Δ_t. More detailed discussion on connections between angular update and performance of neural network can be seen appendix.

4 Experiments

In this section, we verify our theorems on SMD and equilibrium by empirical study. First, we show the equilibrium depicted in our theorems really occurs in various computer vision tasks including ImageNet (Russakovsky et al., 2015) and MSCOCO (Lin et al., 2014). Second, we analyze an interesting phenomenon as an example to show how SMD can affect training process in a way different from traditional view on optimization of neural network.

4.1 Verify the existence of equilibrium

We conduct proving experiments in two cases. In the first case we train neural network using fixed learning rate to verify our assumptions and theorems in SGD and SGDM respectively; in the second case we investigate behavior of angular update with a more practical setting, multi-stage learning rate schedule, to explore what will happen when equilibrium is broken by decaying learning rate.



(a) Unit gradient norm in SGD (b) Weight norm in SGD (c) Angular update in SGD

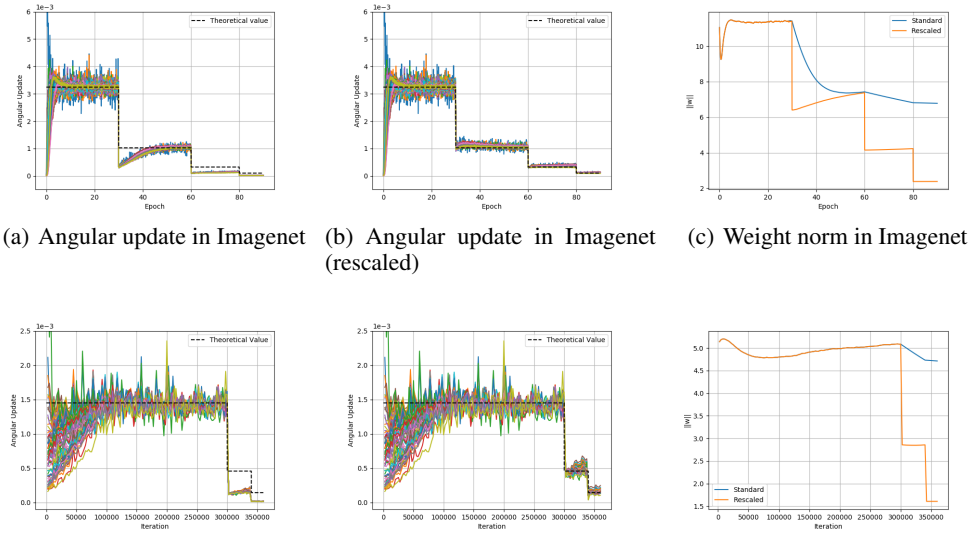
(d) Unit gradient norm square and $\tilde{h}$ in SGDM (e) Weight norm in SGDM (f) Angular update in SGDM

Figure 2: Performance of *layer.2.0.conv2* from Resnet50 in SGD and SGDM, respectively. In (a), (d), semitransparent line represents the raw value of $||\tilde{\boldsymbol{g}}_t||_2^2$ or $\tilde{h}_t$, while solid line represents the averaged value within consecutive 200 iterations to estimate the expectation of $||\tilde{\boldsymbol{g}}_t||_2^2$ or $\tilde{h}_t$ conditioning on t; In (b), (e), blue solid lines represents the raw value of weight norm $||\boldsymbol{w}_t||_2$, while dashed line represent the theoretical value of weight norm computed in Theorem 1, 2 respectively. To compute the theoretical value of weight norm, we use the estimated $\mathbb{E}||\tilde{\boldsymbol{g}}||_2^2$ and $\mathbb{E}\tilde{h}$ (solid lines) in (a) and (d) respectively; In (c), (f), red lines represent raw value of angular update during training, dashed lines represent the theoretical value of angular update computed by $\sqrt{2\lambda\eta}$ and $\sqrt{2\lambda\eta/(1+\alpha)}$ respectively.

Fixed learning rate We train Resnet50 (He et al., 2016) with SGD/SGDM on ImageNet. Learning rate is fixed as $\eta = 0.2$; WD factor is $\lambda = 10^{-4}$; In SGDM case, the momentum factor is $\alpha = 0.9$. Figure 2 presents the square norm of unit gradient, weight norm, and angular update of the weights from LAYER.2.0.CONV2 of Resnet50 in SGD and SGDM cases respectively. In Figure 2(a), 2(d), the solid lines represents the estimated expectation ($\mathbb{E}||\tilde{\boldsymbol{g}}||_2^2$, $\mathbb{E}\tilde{h}$). These estimated expectations allow us to estimate the theoretical value of weight norm in Figure 2(b), 2(e). In Figure 2 the behavior of $||\tilde{\boldsymbol{g}}_t||_2^2$ ($\tilde{h}_t$) and hyperparameter settings satisfy the assumptions used in theorem 1 and 2: $\mathbb{E}||\tilde{\boldsymbol{g}}||_2^2$ ($\mathbb{E}\tilde{h}$) changes slowly and smoothly (see solid line in Figure 2(a), 2(d), the whole training process consists of 450,000 iterations). $||\tilde{\boldsymbol{g}}||_2^2$ ($\tilde{h}$) has a lower bound and moderate variance.

Form Figure 2(b), 2(e), 2(c), 2(f), we can see empirical value of $||\boldsymbol{w}_t||_2^2$ and Δ_t differ from their theoretical value respectively at very beginning, because the initialized value of weight norm is handcrafted, far away from the theoretical value in equilibrium. After several iterations, empirical values of weight norm and angular update agree with their theoretical values very well, which implies equilibrium has been reached. We also observe SGDM can achieve equilibrium much faster than SGD. According to Eq.(8), (12), the underlying reason might be with same learning rate η and WD factor λ, approaching rate of SGDM ($\frac{\lambda\eta}{1-\alpha}$) is larger than approaching rate of SGD ($\lambda\eta$). Results in Figure 2 also prove our claim that equilibrium is a dynamical state: after equilibrium is reached, $\mathbb{E}||\tilde{\boldsymbol{g}}||_2^2$ ($\mathbb{E}\tilde{h}$) constantly increase, $||\boldsymbol{w}_t||_2^2$ increases accordingly, $||\boldsymbol{w}_t||_2^2$ and Δ_t always oscillate around their theoretical values respectively, showing equilibrium state maintains in SMD.

Multi-stage learning rate Now we study the behavior of angular update with SGDM and multi-stage learning rate schedule on Imagenet (Russakovsky et al., 2015) and MSCOCO (Lin et al., 2014). In ImageNet classification task, we still adopt Resnet50 as baseline. The training settings rigorously follow Goyal et al. (2017): learning rate is initialized as $\eta = 0.1$, and divided by 10 at $30, 60, 80$ epoch; WD factor is $\lambda = 10^{-4}$; momentum factor is $\alpha = 0.9$. In MSCOCO experiment, we conduct experiments on Mask-RCNN (He et al., 2017) benchmark using a Feature Pyramid Network (FPN) (Lin et al., 2017), ResNet50 backbone and SyncBN (Peng et al., 2018) following the 4x setting



(a) Angular update in Imagenet (b) Angular update in Imagenet (rescaled) (c) Weight norm in Imagenet

(d) Angular update in MSCOCO (e) Angular update in MSCOCO (rescaled) (f) Weight norm in MSCOCO

Figure 3: In (a),(b),(d),(e), solid lines with different colors represent raw value of angular update from all convolution layers; In (a), (d), training setting rigorously follows Goyal et al. (2017); He et al. (2019) respectively; In (b), (e), weight norm is divided by $\sqrt[4]{10}$ as long as learning rate is divided by 10; In (c), (f), weight norm is computed on *layer.1.0.conv2* in Resnet50 backbone. Blue line represent original settings, orange lines represent rescaled settings.

in He et al. (2019): total number of iteration is $360,000$, learning rate is initialized as 0.02, and divided by 10 at $300000, 340000$ step; WD factor is $\lambda = 10^{-4}$; momentum factor is $\alpha = 0.9$.

There appears to be a mismatch between theorems and empirical observations in Figure 3(a), 3(d): angular update Δ_t in the last two learning rate stages is smaller than its theoretical value. This mismatch can be well interpreted by our theory: according to Theorem 1, 2, when equilibrium state is reached, theoretical value of weight norm $||\boldsymbol{w}_t||_2$ satisfies $||\boldsymbol{w}_t||_2 \propto \sqrt[4]{\frac{\eta}{\lambda}}$. However, when learning rate is divided by k, equilibrium state is broken, theoretical value of weight norm $||\boldsymbol{w}_t||_2$ in the new equilibrium state is $\sqrt[4]{1/k}$ times smaller. But new equilibrium cannot be reached immediately (see Figure 3(c), 3(f)), following corollary gives the least number of iterations to reach new equilibrium.

Corollary 3.1. *In SGD case with learning rate η, WD factor λ, if learning rate is divided by k, and unit gradient norm remains unchanged, then at least $\lceil[\log(k)]/(2\lambda\eta)\rceil$ iterations are required to reach the new equilibrium state; In SGDM case with momentum coefficient α, then at least $\lceil[\log(k)(1-\alpha)]/(2\lambda\eta)\rceil$ iterations are required to reach the new equilibrium state.*

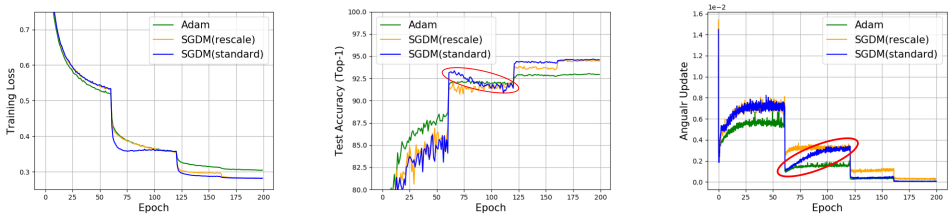
Corollary 3.1 implies SGD/SGDM with smaller learning rate requires more iterations to reach new equilibrium state. Hence, in second learning rate stage in Imagenet experiments, angular update Δ_t can reach its new theoretical value within 15 epochs. But in last two learning rate stages of Imagenet/MSCOCO experiments, SGDM cannot completely reach new equilibrium by the end of training. As a result, we observe empirical value of Δ_t is smaller than its theoretical value. Based on our theorem, we can bridge the gap by skipping the intermediate process between old equilibrium and new one. Specifically, when learning rate is divided by k, norm of scale-invariant weight is also divided by $\sqrt[4]{k}$, SGDM can reach new equilibrium immediately in new learning rate stage. Experiments((b),(e) in Figure 3) show this simple strategy can make angular update Δ_t always close to its theoretical value across the whole training process though learning rate changes.

4.2 "Overfitting" by Spherical Motion Dynamics

"Overfitting" issues often bother practitioners badly when training deep neural networks. The term refers to the phenomenon where trained models can fit training data very well, but fail to fit additional data for validation or prediction. Overfitting has various manifestations in practice, a typical one

is “decreasing test accuracy”, i.e. more training worsen generalization performance. To handle such issues, regularization methods like early-stopping (Prechelt, 1998) and dropout (Srivastava et al., 2014) are proposed. But here we want to explore a question: is the "decreasing test accuracy" phenomenon always caused by overfitting?

Let’s see the following experiment in which Resnet18 (He et al., 2016) is trained on CIFAR10 (Krizhevsky & Geoffrey, 2009) with standard settings (see experiment settings in appendix) Figure 4 presents the training curves of the experiments. It can be seen from figure 4(b)(blue line inside red ellipse) that with standard implementation of SGDM, test accuracy severely drops during the 2nd learning rate stage (this phenomenon does not only occur in our experiment with specific settings, it can be seen in many other papers). From traditional view of optimization, it seems to be a typical overfitting issue. However, two other phenomena cannot be well interpreted by overfitting: 1) test accuracy *only* severely decreases during 2nd learning rate stage, it performs normally in the 3rd, 4th learning rate stages (blue lines in Figure4(b)); 2) When other optimization method like Adam (Kingma & Ba, 2015) is used, test accuracy does not drop apparently in any learning rate stages (green line in Figure4(b)).



(a) Training loss (b) Test Accuracy (Top-1) (c) Angular update

Figure 4: Training curves of Resnet18 trained on CIFAR10 (averaged across 5 seeds). Angular update is from layer1.0.conv1 of Resnet18.

Fortunately, SMD can provide a more reasonable interpretation on this “decreasing test accuracy” phenomenon: as we demonstrate in section 4.1, after learning rate decays, equilibrium is broken, SMD will get in a new equilibrium state. In the intermediate process between two equilibrium states, angular update will constantly increase till approaching to its new theoretical value. Recall the geometric meaning of angular update is the effective update of the weights on its intrinsic domain $\mathcal{S}^{p-1}$. Hence, increasing angular update can force optimization trajectory to escape from the local optimum (defined on $\mathcal{S}^{p-1}$), resulting in decreasing test accuracy phenomenon. The increasing angular update phenomenon is easily visible in 2nd learning rate stage(see blue lines inside red ellipse of Figure 4(c)), but not obvious in 3rd and 4th stages. Therefore decreasing test accuracy phenomenon is not apparent in 3rd and 4th learning rate stages; Besides, Adam has no equilibrium state like SGDM, so there’s no apparent increasing angular update phenomenon (see green line in Figure 4(c)), decreasing test accuracy does not happen when Adam is used. Based on our interpretation, decreasing test accuracy phenomenon in SGDM could be avoided by “rescaling” strategy introduced in Section 4.1. The experiments in Figure 4 (blue lines) strongly proves our claim. We need to emphasize that our result does not implies all decreasing test accuracy phenomena are caused by SMD, but our analysis and experiment show SMD can dramatically affect the optimization trajectory in a way different from traditional view of optimization.

5 Conclusion

In this paper, we comprehensively reveal the learning dynamics of normalized neural network with SGD/SGDM and weight decay (WD), named as Spherical Motion Dynamics (SMD). With mild assumptions, we strictly prove SMD will reach equilibrium state in a linear regime. We also propose a novel index, angular update, to depict the state of SMD, and derive its theoretical property in equilibrium. Most importantly, we show our theorem is widely valid, they can be verified on challenging computer vision tasks, beyond synthetic datasets. Besides, we show SMD can dramatically effect the optimization of neural network by controlling angular update in practice. We believe our results on SMD make an important step to understand the mechanism of deep neural networks, and can inspire new deep learning techniques.

References

Arora, S., Li, Z., and Lyu, K. Theoretical analysis of auto rate-tuning by batch normalization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rkxQ-nA9FX.

Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Chiley, V., Sharapov, I., Kosson, A., Koster, U., Reece, R., Samaniego de la Fuente, S., Subbiah, V., and James, M. Online normalization for training neural networks. In *Advances in Neural Information Processing Systems 32*, pp. 8433–8443. Curran Associates, Inc., 2019.

Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., and He, K. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

He, K., Girshick, R., and Dollar, P. Rethinking imagenet pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Hoffer, E., Banner, R., Golan, I., and Soudry, D. Norm matters: efficient and accurate normalization schemes in deep networks. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 2160–2170. Curran Associates, Inc., 2018.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pp. 448–456, 2015.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015.

Krizhevsky, A. and Geoffrey, H. Learning multiple layers of features from tiny images. 2009.

Kunin, D., Sagastuy-Brena, J., Ganguli, S., Yamins, D. L., and Tanaka, H. Neural mechanics: Symmetry and broken conservation laws in deep learning dynamics. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=q8qLAbQBupm.

Li, Z. and Arora, S. An exponential learning rate schedule for deep learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rJg8TeSFDH.

Li, Z., Lyu, K., and Arora, S. Reconciling modern deep learning with traditional optimization analyses: The intrinsic learning rate. *Advances in Neural Information Processing Systems*, 33, 2020.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

Ma, N., Zhang, X., Zheng, H.-T., and Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 116–131, 2018.

Peng, C., Xiao, T., Li, Z., Jiang, Y., Zhang, X., Jia, K., Yu, G., and Sun, J. Megdet: A large mini-batch object detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6181–6189, 2018.

Polyak, B. T. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

Prechelt, L. Early stopping-but when? In *Neural Networks: Tricks of the trade*, pp. 55–69. Springer, 1998.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Salimans, T. and Kingma, D. P. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Advances in Neural Information Processing Systems*, pp. 901–909, 2016.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1): 1929–1958, 2014.

van Laarhoven, T. L2 regularization versus batch and weight normalization. In *Advances in Neural Information Processing Systems*. 2017.

Wu, Y. and He, K. Group normalization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 3–19, 2018.

Zhang, G., Wang, C., Xu, B., and Grosse, R. Three mechanisms of weight decay regularization. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=B1lz-3Rct7`.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section **??**.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See abstract and Section 1
   (b) Did you describe the limitations of your work? [Yes] See remark 3 and last discussion section 4.2
   (c) Did you discuss any potential negative societal impacts of your work? [No] This work is mainly a theoretical work. No improvement or new technique is proposed.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] Only on cifar10 experiments.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]