Provable Super-Convergence With a Large Cyclical Learning Rate

Samet Oymak ⓘ, *Member, IEEE*

Abstract—**Conventional wisdom dictates that learning rate should be in the stable regime so that gradient-based algorithms don't blow up. This letter introduces a simple scenario where an unstably large learning rate scheme leads to a super fast convergence, with the convergence rate depending only logarithmically on the condition number of the problem. Our scheme uses a Cyclical Learning Rate where we periodically take one large unstable step and several small stable steps to compensate for the instability. These findings also help explain the empirical observations of [Smith and Topin, 2019] where they show that CLR with a large maximum learning rate can dramatically accelerate learning and lead to so-called "super-convergence". We prove that our scheme excels in the problems where Hessian exhibits a bimodal spectrum and the eigenvalues can be grouped into two clusters (small and large). The unstably large step is the key to enabling fast convergence over the small eigen-spectrum.**

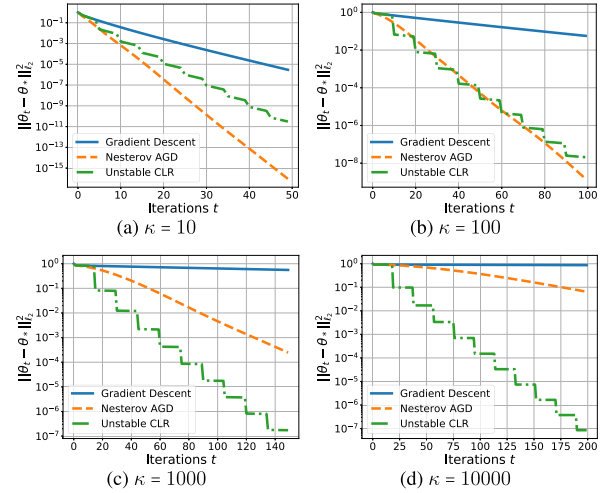*Index Terms*—**Convergence of numerical methods, Iterative algorithms, Gradient methods.**

Fig. 1. Convergence performance of gradient descent with $\eta_t = 1/L$, Nesterov's acceleration (AGD) and Unstable Cyclical Learning Rate of Theorem 1 on a linear regression task. In Figures (a) to (d), we vary the condition number κ from 10 to 10^4. In these experiments, eigenspectrum of Hessian is bimodal where eigenvalues lie over the intervals [1,2] and $[\kappa/2, \kappa]$. This means the local condition numbers are $\kappa_+ = \kappa_- = 2$. As the global condition number grows, Unstable CLR outperforms standard gradient descent or Nesterov AGD as it only requires logarithmic iteration in κ. We note that Unstable CLR can potentially further benefit from acceleration.

I. INTRODUCTION

$\mathbf{C}$ONSIDER a least-squares problem with design matrix $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ and labels $\boldsymbol{y} \in \mathbb{R}^n$. We wish to solve for

$$\boldsymbol{\theta}_\star = \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^p} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_{\ell_2}^2.$$

If we use a gradient-based algorithm the rate of convergence obviously depends on the condition number κ of $\boldsymbol{X}$. Here $\kappa = L/\mu$ where the smoothness L and strong convexity μ of the problem is given by the maximum and minimum eigenvalues of the Hessian matrix $\boldsymbol{X}^\top \boldsymbol{X}$ as follows

$$L = \|\boldsymbol{X}^\top \boldsymbol{X}\|, \quad \mu = \sigma_{\min}(\boldsymbol{X}^\top \boldsymbol{X}).$$

Here, $\sigma_{\min}(), \|\cdot\|$ denote the smallest/largest singular value of a matrix respectively. Standard gradient descent (GD) requires $\kappa \log(\varepsilon^{-1})$ iterations to achieve ε-accuracy. Nesterov's acceleration can improve this to $\sqrt{\kappa} \log(\varepsilon^{-1})$. In general, consider the iterations

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \boldsymbol{X}^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}_t).$$

Here, the contraction matrix $\boldsymbol{C}_t = \boldsymbol{I} - \eta_t \boldsymbol{X}^T \boldsymbol{X}$ governs the rate of convergence. Over the ith eigen-direction of $\boldsymbol{X}^\top \boldsymbol{X}$ with

eigenvalue λ_i, the convergence/contraction rate is given by $1 - \eta_t \lambda_i$. Setting a fixed stable learning rate of $\eta_t = 1/L$, the issue is that gradient descent optimizes small eigen-directions much slower than the large eigen-directions. Faster learning over small eigen-directions can be facilitated by a large learning rate so that $1 - \eta_t \lambda_i$ is closer to 0 even for small λ_i's. However this would lead to instability i.e. $\|\boldsymbol{C}_t\| > 1$.

Here, we point out the possibility that, one can use an unstably large learning rate once in a while to provide *huge improvements over small directions*. The resulting instability can be compensated quickly by following this unstable step with multiple stable steps which keep the larger directions under control. Overall, for problems with bimodal Hessian spectrum, where eigenvalues are clustered in large and small groups, this leads to substantial improvements with logarithmic dependency on the condition number. Fig. 1 highlights this phenomena. Our approach can be formalized by using a cyclical (aka periodic) learning rate schedule [1], [2]. We remark that cyclical learning rate (CLR) is related to SGD with Restarts (SGDR) and Stochastic Weight Averaging which attracted significant attention due to their fast

convergence, generalization benefits and flexibility [3]–[5]. [6] assesses certain theoretical benefits of cosine learning rates which are periodic. [7], [8] investigate periodic learning rates to facilitate escape from saddle points with stochastic gradients. Large initial learning rates are also known to play a critical role in generalization [9]–[11]. Recent work [12] provides further empirical evidence that practical learning rates can be at the edge of stability. However to the best of our knowledge, prior works do not consider potential theoretical benefits of unstably large learning rate choice. The closest work is by Smith and Topin [3]. Here authors observe that CLR can operate with very large maximum learning rates and converge "super fast". We believe this letter provides a rigorous theoretical support for such observations. We show that maximum learning rate can in fact be unstable and it can be much larger than the maximum stable learning rate. We also show that "super fast" convergence can be as fast as logarithmic in condition number (which is drastically better than GD or Nesterov AGD) for suitable problems (discussed further below).

Our CLR scheme simply takes two values $\eta_\pm$ as described below.

Definition 1: Fix an integer $T > 1$ and positive scalars η_+, η_-. Set periodic learning rate η_t for $t \geq 0$ as

$$\eta_t = \begin{cases} \eta_- & \text{if } \mod(t, T) = -1 \\ \eta_+ & \text{else} \end{cases}.$$

We call this schedule *Unstable CLR* if $\eta_- > 2/L$ where L is the smoothness of the problem. Indeed when this happens, the algorithm is susceptible to blow up in certain eigendirections as the contraction matrix $I - \eta_- X^\top X$ has operator norm larger than 1.

Theorem 1 (Linear regression): Let $\lambda \in \mathbb{R}_+^p$ be the decreasingly sorted eigenvalues of $X^\top X$ with $L = \lambda_1$ and $\mu = \lambda_p$. Fix some integer r with $p > r \geq 1$. Introduce the quantities

$$\kappa = \frac{L}{\mu}, \quad \kappa_+ = \frac{L}{\lambda_r}, \quad \kappa_- = \frac{\lambda_{r+1}}{\mu}.$$

Set period $T \geq \kappa_+ \log(\frac{2\kappa}{2\kappa_- - 1}) + 1$ and learning rate according to Def. 1 with $\eta_+ = \frac{1}{L}$ and $\eta_- = \frac{1}{\kappa_- \mu}$. For all t with $\mod(t, T) = 0$, the iterations obey

$$\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_\star\|_{\ell_2} \leq \left(1 - \frac{1}{2\kappa_-}\right)^{t/T} \|\boldsymbol{\theta}_0 - \boldsymbol{\theta}_\star\|_{\ell_2}. \tag{1}$$

Alternatively, for $t \geq 2T\kappa_- \log(\varepsilon^{-1})$, we have $\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_\star\|_{\ell_2} \leq \varepsilon \|\boldsymbol{\theta}_0 - \boldsymbol{\theta}_\star\|_{\ell_2}$ for $1 > \varepsilon > 0$.

Interpretation: Observe that in order to achieve ε accuracy, the number of required iterations grow as

$$\boxed{\kappa_+ \kappa_- \log(\kappa) \log(\varepsilon^{-1}).} \tag{2}$$

Here κ_+, κ_- are the *local condition numbers* for the eigenspectrum. κ_+ is the condition number over the subspace $\mathcal{S}$ (i.e. eigens from λ_1 to λ_r) and κ_- is the condition number over the orthogonal complement $\mathcal{S}^c$. Observe that $\kappa_+ \times \kappa_-$ is always upper bounded by the overall condition number κ i.e. $\kappa_+ \kappa_- \leq \kappa$. However if eigenvalues over $\mathcal{S}$ and $\mathcal{S}^c$ are narrowly clustered, we can have $\kappa_+ \kappa_- \ll \kappa$ leading to a much faster convergence.

Finally observe that the dependence on the (global) condition number κ is only logarithmic. Thus, in the worst case scenario of $\kappa_+ \kappa_- = \kappa$, the iteration complexity (2) is sub-optimal by a $\log(\kappa)$-factor compared to the standard gradient descent which requires $\kappa \log(\varepsilon^{-1})$ iterations. However there is a factor of $\kappa/\log(\kappa)$ improvement when the eigenvalues are clustered and κ_+, κ_- are small. Fig. 1 demonstrates the comparisons to gradient descent (with $\eta = 1/L$) and Nesterov's Accelerated GD. In these examples, we set local condition numbers to $\kappa_- = \kappa_+ = 2$ whereas κ varies from 10 to 10 000. As κ grows larger, Unstable CLR shines over the alternatives as the iteration number only grows as $\log(\kappa)$. Finally, note that inverse learning rate $\eta_+^{-1} = L$ is equal to the smoothness (top Hessian eigenvalue) of the whole problem whereas $\eta_-^{-1} = \kappa_- \mu$ is chosen based on the smoothness over the lower spectrum.

Bimodal Hessian: The bimodal Hessian spectrum is crucial for enabling super fast convergence. In essence, with bimodal structure, Unstable CLR acts as a *preconditioner* for the problem and helps not only large eigen-value/directions but also small ones. It is possible that other cyclical schemes will accelerate a broader class of Hessian spectrums. That said, we briefly mention that bimodal spectrum has empirical support in the deep learning literature. Several works studied the empirical Hessians (and Jacobians) of deep neural networks [13]–[15]. A common observation is that the Hessian spectrum has relatively few large eigenvalues and many more smaller eigenvalues [13], [15]. These observations are closely related to the seminal *spiked covariance model* [16] where the covariance spectrum has a few large eigenvalues and many more smaller eigenvalues. In connection, [14], [17], [18] studied the Jacobian spectrum. Note that, in the linear regression setting of Theorem 1, Jacobian is simply X. They similarly found that practical deep nets have a bimodal spectrum and the Jacobian is approximately low-rank. It is also known that behavior of the wide deep nets (i.e., many neurons per layer) can be approximated by their Jacobian-based linearization at initialization via neural tangent kernel [19], [20]. Thus the spectrum indeed closely governs the optimization dynamics [21] similar to our simple linear regression setup. While the larger eigendirections of a deep net can be optimized quickly [14], [18], intuitively their existence will slow down the small eigendirections. However it is also observed that learning such small eigendirections is critical for success of deep learning since typically achieving zero-training loss (aka interpolation) leads to the best generalization performance [22]–[24]. In light of this discussion, related empirical observations of [3] and our theory, it is indeed plausible that Unstable CLR does a stellar job at learning these small eigen-directions leading to much faster interpolation and improved generalization.

II. EXTENSION TO THE NONLINEAR PROBLEMS

To conclude with our discussion, we next provide a more general result that apply to strongly-convex functions. Recall $\mathcal{S}^c$ as the complement of a subspace $\mathcal{S}$. Let $\boldsymbol{\Pi}_\mathcal{S}$ denote the matrix that projects onto $\mathcal{S}$. Our goal is solving $\boldsymbol{\theta}_\star = \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^p} f(\boldsymbol{\theta})$ via gradient iterations $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \boldsymbol{\nabla} f(\boldsymbol{\theta}_t)$.

Definition 2 (Smoothness and strong-convexity): Let $f : \mathbb{R}^p \to \mathbb{R}$ be a smooth convex function and fix $L > \mu > 0$.

$f : \mathbb{R}^p \to \mathbb{R}$ is L-smooth and μ strongly-convex if its Hessian satisfies the following for all $\boldsymbol{\theta} \in \mathbb{R}^p$

$$L\boldsymbol{I}_p \succeq \boldsymbol{\nabla}^2 f(\boldsymbol{\theta}) \succeq \mu\boldsymbol{I}_p.$$

First, we recall a classical convergence result for strongly convex functions for the sake of completeness.

Proposition 1: Let f obey Definition 2 and suppose $\boldsymbol{\theta}_\star$ is its minimizer. Pick a learning rate $\eta \leq 1/L$, and use the iterations $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta\boldsymbol{\nabla} f(\boldsymbol{\theta}_t)$. The iterates obey $\|\boldsymbol{\theta}_\tau - \boldsymbol{\theta}_\star\|_{\ell_2}^2 \leq (1 - \eta\mu)^\tau \|\boldsymbol{\theta}_0 - \boldsymbol{\theta}_\star\|_{\ell_2}^2$.

This is the usual setup for gradient descent analysis. Instead, we will employ the bimodal Hessian definition.

Definition 3 (Bimodal Hessian): Let $f : \mathbb{R}^p \to \mathbb{R}$ be an L smooth μ strongly-convex function. Additionally, there exists a subspace $\mathcal{S} \in \mathbb{R}^p$ and local condition numbers $\kappa_+, \kappa_- \geq 1$ and cross-smoothness $\varepsilon \geq 0$ such that, the Hessian of f is satisfies

Upper spectrum: $L\boldsymbol{I}_p \succeq \boldsymbol{\Pi}_\mathcal{S} \boldsymbol{\nabla}^2 f(\boldsymbol{\theta}) \boldsymbol{\Pi}_\mathcal{S} \succeq (L/\kappa_+)\boldsymbol{I}_p$,

Lower spectrum: $\kappa_- \mu \boldsymbol{I}_p \succeq \boldsymbol{\Pi}_{\mathcal{S}^c} \boldsymbol{\nabla}^2 f(\boldsymbol{\theta}) \boldsymbol{\Pi}_{\mathcal{S}^c} \succeq \mu\boldsymbol{I}_p$,

Cross spectrum: $\|\boldsymbol{\Pi}_\mathcal{S} \boldsymbol{\nabla}^2 f(\boldsymbol{\theta}) \boldsymbol{\Pi}_{\mathcal{S}^c}\| \leq \varepsilon$. $\qquad$ (3)

Here κ_+, κ_- are the local condition numbers as previously. Observe that both of these are upper bounded by the global condition number $\kappa = L/\mu$ as f obeys Def. 2 as well. The cross-smoothness controls the interaction between two subspaces. For linear regression $\varepsilon = 0$ by picking $\mathcal{S}$ to be eigenspace. For general nonlinear models, as long as problem can be approximated linearly (e.g. wide deep nets can be approximated by their linearization [19], [20]), it is plausible that cross-smoothness is small for a suitable choice of $\mathcal{S}$. We have the following analogue of Theorem 1.

Theorem 2 (Nonlinear problems): Let f obey Definition 3 with non-negative scalars $L, \mu, \kappa_+, \kappa_-, \varepsilon$. Consider the learning rate schedule of Definition 1 and set

$$T \geq 2\kappa_+ \log\left(\frac{2L}{\kappa_-\mu}\right) + 1 \quad, \quad \eta_- = \frac{1}{\kappa_-\mu} \quad, \quad \eta_+ = \frac{1}{L}.$$

Additionally, suppose the cross-smoothness ε satisfies the following upper bound

$$4\varepsilon \leq \min(1, \kappa_-/T)\, \mu.$$

Let $\boldsymbol{\theta}_\star$ be the minimizer of $f(\boldsymbol{\theta})$. Starting from $\boldsymbol{\theta}_0$, apply the gradient iterations $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \boldsymbol{\nabla} f(\boldsymbol{\theta}_t)$. For all iterations $t \geq 0$ with $\mathrm{mod}(t, T) = 0$, we have

$$\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_\star\|_{\ell_2} \leq \sqrt{2}\left(1 - \frac{1}{4\kappa_-}\right)^{t/T} \|\boldsymbol{\theta}_0 - \boldsymbol{\theta}_\star\|_{\ell_2}.$$

Interpretation: This result is in similar spirit to the linear regression setup of Theorem 1. The required number of iterations is still governed by the quantity (2). A key difference is the cross-smoothness ε which controls the cross-Hessian matrix $\boldsymbol{\Pi}_\mathcal{S} \boldsymbol{\nabla}^2 f(\boldsymbol{\theta}) \boldsymbol{\Pi}_{\mathcal{S}^c}$. This term was simply equal to 0 for the linear problem. In essence, our condition for nonlinear problems essentially requires cross-Hessian to be dominated by the Hessian over the lower spectrum i.e. $\boldsymbol{\Pi}_{\mathcal{S}^c} \boldsymbol{\nabla}^2 f(\boldsymbol{\theta}) \boldsymbol{\Pi}_{\mathcal{S}^c}$. In particular, ε should be upper bounded by the strong convexity parameter μ as well as $\kappa_-\mu/T$ where $\kappa_-\mu$ is the smoothness over the lower

spectrum. It would be interesting to explore to what extent the conditions on the cross-Hessian can be relaxed. However, as mentioned earlier, the fact that wide artificial neural networks behave close to linear models [19] provides a decent justification for small cross-Hessian.

III. PROOFS

A. Proof of Theorem 1

Proof: Following Theorem 1's statement introduce $L = \lambda_1$, $\mu = \lambda_p$. Each gradient iteration can be written in terms of the residual $\boldsymbol{w}_t = \boldsymbol{\theta}_t - \boldsymbol{\theta}_\star$ and has the following form

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}_t)$$
$$\Rightarrow \boldsymbol{w}_{t+1} := \boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_\star = (\boldsymbol{I} - \boldsymbol{X}^T\boldsymbol{X})\boldsymbol{w}_t.$$

Let $\mathcal{S}$ be the principal eigenspace induced by the first r eigenvectors and $\mathcal{S}^c$ be its complement. Let $\boldsymbol{\Pi}_\mathcal{S}$ be the projection operator on $\mathcal{S}$. Set $\boldsymbol{a}_t = \boldsymbol{\Pi}_\mathcal{S}(\boldsymbol{w}_t)$, $\boldsymbol{b}_t = \boldsymbol{\Pi}_{\mathcal{S}^c}(\boldsymbol{w}_t)$. Also set $a_t = \|\boldsymbol{a}_t\|_{\ell_2}$, $b_t = \|\boldsymbol{b}_t\|_{\ell_2}$. Since the learning rate is periodic, we analyze a single period starting from $\boldsymbol{a}_0, \boldsymbol{b}_0$. During the first $T - 1$ iterations, we have that

$$a_t \leq \left(1 - \frac{1}{\kappa_+}\right)^t a_0 \quad \text{and} \quad b_t \leq \left(1 - \frac{\mu}{L}\right)^t b_0 = \left(1 - \frac{1}{\kappa}\right)^t b_0.$$

At the final (unstably large) iteration $t = T - 1$, we have

$$a_T \leq \frac{L}{\kappa_-\mu} a_{T-1} = \frac{\kappa}{\kappa_-} a_{T-1} \quad \text{and} \quad b_t \leq \left(1 - \frac{1}{\kappa_-}\right) b_0.$$

Now b_t term is clearly non-increasing and obeys $b_T \leq (1 - \frac{1}{\kappa_-})(1 - \frac{1}{\kappa})^{T-1} b_0 \leq (1 - \frac{1}{\kappa}) b_0$. Note that we forego the $(1 - \frac{1}{\kappa})^{T-1}$ for the sake of simplicity. We wish to make the growth of a_T similarly small by enforcing

$$\frac{\kappa}{\kappa_-}\left(1 - \frac{1}{\kappa_+}\right)^{T-1} \leq 1 - \frac{1}{2\kappa_-} \iff \frac{2\kappa}{2\kappa_- - 1}$$
$$\leq \left(\left(1 - \frac{1}{\kappa_+}\right)^{-1}\right)^{T-1}.$$

Observe that $(1 - \frac{1}{\kappa_+})^{-1} \geq e^{1/\kappa_+}$. Thus, we simply need

$$\log\left(\frac{2\kappa}{2\kappa_- - 1}\right) \leq \frac{T-1}{\kappa_+} \iff T \geq \kappa_+ \log\left(\frac{2\kappa}{2\kappa_- - 1}\right) + 1.$$

In short, at the end of a single period we are guaranteed to have (1) after observing $\|\boldsymbol{w}_t\|_{\ell_2}^2 = a_t^2 + b_t^2$. $\qquad\blacksquare$

B. Proof of Theorem 2

Proof: The proof idea is same as Theorem 1 but we additionally control the cross-smoothness terms. Denote the residual by $\boldsymbol{w}_t = \boldsymbol{\theta}_t - \boldsymbol{\theta}_\star$. We will work with the projections of the residual on the subspaces $\mathcal{S}, \mathcal{S}^c$ denoted by $\boldsymbol{a}_t, \boldsymbol{b}_t$ respectively. Additionally, set $B = \max(\|\boldsymbol{a}_0\|_{\ell_2}, \|\boldsymbol{b}_0\|_{\ell_2})$ and set $a_t = \|\boldsymbol{a}_t\|_{\ell_2}/B, b_t = \|\boldsymbol{b}_t\|_{\ell_2}/B$. Observe that by this definition $B \leq \|\boldsymbol{\theta}_\star - \boldsymbol{\theta}_0\|_{\ell_2}$ and

$$\max(a_0, b_0) = 1.$$

We will prove that at the end of a single period, (a_T, b_T) pair obeys

$$\max(a_T, b_T) \leq 1 - \frac{1}{4\kappa_-}. \tag{4}$$

The overall result follows inductively from this as follows. First, inductively we would achieve $\max(a_t, b_t) \leq (1 - \frac{1}{4\kappa_-})^{t/T}$ for $\mathrm{mod}(t, T) = 0$. That would in turn yield

$$\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_\star\|_{\ell_2} \leq \sqrt{2}\max(\|\boldsymbol{a}_t\|_{\ell_2}, \|\boldsymbol{b}_t\|_{\ell_2}) \leq \sqrt{2}B\max(a_t, b_t)$$

$$\leq \sqrt{2}(1 - \frac{1}{4\kappa_-})^{t/T}\|\boldsymbol{\theta}_\star - \boldsymbol{\theta}_0\|_{\ell_2}.$$

Establishing (4): Thus, let us show (4) for a single period of learning rate i.e. $0 \leq t \leq T$. The gradient is given by

$$\boldsymbol{\nabla} f(\boldsymbol{\theta}_t) = \boldsymbol{H}\boldsymbol{w}_t,$$

where $\boldsymbol{H}$ is obtained by integrating the Hessian's along the path from $\boldsymbol{\theta}_\star$ to $\boldsymbol{\theta}_t$. Write the next iterate as

$$\|\boldsymbol{w}_{t+1}\|_{\ell_2}^2 = \|\boldsymbol{w}_t - \eta_t\boldsymbol{\nabla} f(\boldsymbol{\theta}_t)\|_{\ell_2}^2 = \|\boldsymbol{w}_t - \eta_t\boldsymbol{H}\boldsymbol{w}_t\|_{\ell_2}^2.$$

By Definition 3 and triangle inequality, $\boldsymbol{H}$ satisfies the bimodal Hessian inequalities described in (3). To analyze a single period, we first focus on the lower learning rate η_+ which spans the initial $T - 1$ iterations. Hence, suppose $\eta_t = \eta_+ = 1/L$. Note that from strong convexity of $\boldsymbol{H}$ over $\mathcal{S}, \mathcal{S}^c$, we have that

$$\|\boldsymbol{a}_t - \eta_t\boldsymbol{\Pi}_\mathcal{S}\boldsymbol{H}\boldsymbol{a}_t\|_{\ell_2}^2 \leq (1 - \eta_t L/\kappa_+)\|\boldsymbol{a}_t\|_{\ell_2}^2 \quad \text{for} \quad \eta_t \leq 1/L$$

$$\|\boldsymbol{b}_t - \eta_t\boldsymbol{\Pi}_{\mathcal{S}^c}\boldsymbol{H}\boldsymbol{b}_t\|_{\ell_2}^2 \leq (1 - \eta_t\mu)\|\boldsymbol{b}_t\|_{\ell_2}^2 \quad \text{for} \quad \eta_t \leq 1/(\kappa_-\mu).$$

Following this with $\eta_t = \eta_+ = 1/L$, for $\boldsymbol{a}_t$, we find the recursions

$$\begin{aligned}\|\boldsymbol{a}_{t+1}\|_{\ell_2} &= \|\boldsymbol{a}_t - \eta_t\boldsymbol{\Pi}_\mathcal{S}\boldsymbol{H}\boldsymbol{w}_t\|_{\ell_2} \\ &\leq \|\boldsymbol{a}_t - \eta_t\boldsymbol{\Pi}_\mathcal{S}\boldsymbol{H}\boldsymbol{a}_t\|_{\ell_2} + \eta_t\|\boldsymbol{\Pi}_\mathcal{S}\boldsymbol{H}\boldsymbol{b}_t\|_{\ell_2} \\ &\leq \left(1 - \frac{\eta_t L}{2\kappa_+}\right)\|\boldsymbol{a}_t\|_{\ell_2} + \eta_t\varepsilon\|\boldsymbol{b}_t\|_{\ell_2} \\ &= \left(1 - \frac{1}{2\kappa_+}\right)\|\boldsymbol{a}_t\|_{\ell_2} + \frac{\varepsilon}{L}\|\boldsymbol{b}_t\|_{\ell_2}. \end{aligned} \tag{5}$$

Using the identical argument for $\boldsymbol{b}_t$ yields

$$\begin{aligned}\|\boldsymbol{b}_{t+1}\|_{\ell_2} &= \|\boldsymbol{b}_t - \eta_t\boldsymbol{\Pi}_{\mathcal{S}^c}\boldsymbol{H}\boldsymbol{w}_t\|_{\ell_2} \\ &\leq \|\boldsymbol{b}_t - \eta_t\boldsymbol{\Pi}_{\mathcal{S}^c}\boldsymbol{H}\boldsymbol{b}_t\|_{\ell_2} + \eta_t\|\boldsymbol{\Pi}_{\mathcal{S}^c}\boldsymbol{H}\boldsymbol{a}_t\|_{\ell_2} \\ &\leq \left(1 - \frac{\mu}{2L}\right)\|\boldsymbol{b}_t\|_{\ell_2} + \frac{\varepsilon}{L}\|\boldsymbol{a}_t\|_{\ell_2}. \end{aligned}$$

Setting $\bar{\varepsilon} = \frac{\varepsilon}{L}$, we obtain

$$a_{t+1} \leq \left(1 - \frac{1}{2\kappa_+}\right)a_t + \bar{\varepsilon}b_t. \tag{6}$$

Additionally, recall that, since f is μ strongly-convex we have $L/\kappa_+ \geq \mu$. Thus using $L/\kappa_+ \geq \mu \geq 2\varepsilon$ we have

$$\|\boldsymbol{b}_{t+1}\|_{\ell_2} \leq \left(1 - \frac{\mu}{2L}\right)\|\boldsymbol{b}_t\|_{\ell_2} + \bar{\varepsilon}\|\boldsymbol{a}_t\|_{\ell_2}$$

$$\leq \max(\|\boldsymbol{a}_t\|_{\ell_2}, \|\boldsymbol{b}_t\|_{\ell_2})$$

$$\|\boldsymbol{a}_{t+1}\|_{\ell_2} \leq \left(1 - \frac{1}{2\kappa_+}\right)\|\boldsymbol{a}_t\|_{\ell_2} + \bar{\varepsilon}\|\boldsymbol{b}_t\|_{\ell_2}$$

$$\leq \max(\|\boldsymbol{a}_t\|_{\ell_2}, \|\boldsymbol{b}_t\|_{\ell_2}).$$

That is, we are guaranteed to have $1 \geq a_t, b_t \geq 0$. Thus, using (6), recursively, for all $0 \leq t \leq T - 1$, a_t satisfies

$$a_t \leq \left(1 - \frac{1}{2\kappa_+}\right)^t + \bar{\varepsilon}\sum_{\tau=0}^{t-1}\left(1 - \frac{1}{2\kappa_+}\right)^{t-\tau-1}$$

$$b_t \leq \left(1 - \frac{1}{2\kappa_+}\right)^t + \bar{\varepsilon}\sum_{\tau=0}^{t-1}b_\tau \leq \left(1 - \frac{1}{2\kappa_+}\right)^t + t\bar{\varepsilon}. \tag{7}$$

At time $t = T - 1$, we use the larger learning rate $\eta_- = \frac{1}{\kappa_-\mu}$. The following holds via identical argument as (5)

$$a_{t+1} \leq \frac{L}{\kappa_-\mu}a_t + \frac{\varepsilon}{\kappa_-\mu}b_t$$

$$b_{t+1} \leq \left(1 - \frac{1}{2\kappa_-}\right)b_t + \frac{\varepsilon}{\kappa_-\mu}a_t. \tag{8}$$

To bound b_T at time $t = T - 1$, we recall $b_{T-1}, a_{T-1} \leq 1$ and the bound $\varepsilon \leq \mu/4$, to obtain

$$b_T \leq 1 - \frac{1}{2\kappa_-} + \frac{\varepsilon}{\kappa_-\mu} \leq 1 - \frac{1}{4\kappa_-}.$$

Bounding a_T is what remains. Noticing $\log(1 - x) \leq -x$, our period choice T obeys

$$T \geq 2\kappa_+\log(2L/(\kappa_-\mu)) + 1 \geq 1 - \frac{\log(2L/(\kappa_-\mu))}{\log(1 - \frac{1}{2\kappa_+})}$$

for $\kappa_+ \geq 1$. Thus, using the assumption $\varepsilon \leq \frac{\kappa_-\mu}{4T}$, and the bounds (7) and (8), the bound on a_T is obtained via

$$a_T \leq \frac{L}{\kappa_-\mu}\left(\left(1 - \frac{1}{2\kappa_+}\right)^{T-1} + (T-1)\frac{\varepsilon}{L}\right) + \frac{\varepsilon}{\kappa_-\mu}$$

$$\leq \frac{1}{2} + T\frac{\varepsilon}{\kappa_-\mu} \leq \frac{3}{4} \leq 1 - \frac{1}{4\kappa_-},$$

where we noted $\kappa_- \geq 1$. The above bounds on a_T, b_T establishes (4) and concludes the proof. ∎

IV. CONCLUSION

This letter introduced a setting where remarkably fast convergence can be attained by using cyclical learning rate schedule that carefully targets the bimodal structure of the Hessian of the problem. This is accomplished by replacing global condition number with local counterparts. While bimodal Hessian seems to be a strong assumption, recent literature provides rich empirical justification for our theory. Besides relaxing the assumptions in Theorem 2, there are multiple interesting directions for Unstable CLR. The results can likely be extended to minibatch SGD, overparameterized settings, or to non-convex problems (e.g. via Polyak-Lojasiewicz condition [25]). While our attention was restricted to bimodal Hessian and a CLR scheme with two values (η_+, η_- as in Def. 1), it would be exciting to explore whether more sophisticated CLR schemes can provably accelerate optimization for a richer class of Hessian structures. On the practical side, further empirical investigation (beyond [3], [12]) can help verify and explore the potential benefits of large cyclical learning rates.

REFERENCES

[1] L. N. Smith, "Cyclical learning rates for training neural networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vision*, 2017, pp. 464–472.

[2] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *Proc. Int. Conf. Learn. Representations*, 2017.

[3] L. N. Smith and N. Topin, "Super-convergence: Very fast training of residual networks using large learning rates," *Artif. Intell. Mach. Learn. Multi-Domain Operations Appl.*, vol. 11006, 2019.

[4] H. Fu, C. Li, X. Liu, J. Gao, A. Celikyilmaz, and L. Carin, "Cyclical annealing schedule: A simple approach to mitigating KL vanishing," in *Proc. 2019 Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol.*, vol. 1, 2019.

[5] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, and A. G. Wilson, "Averaging weights leads to wider optima and better generalization," in *Proc. 34th Conf. Uncertainty Artif. Intell. Assoc. Uncertainty Artif. Intell.*, 2018, pp. 876–885.

[6] X. Li, Z. Zhuang, and F. Orabona, "Exponential step sizes for non-convex optimization," in *Proc. Int. Conf. Mach. Learn.* 2020.

[7] K. Zhang, A. Koppel, H. Zhu, and T. Basar, "Global convergence of policy gradient methods to (almost) locally optimal policies," *SIAM J. Control Optim.*, vol. 58, no. 6, pp. 3586–3612, 2020.

[8] H. Daneshmand, J. Kohler, A. Lucchi, and T. Hofmann, "Escaping saddles with stochastic gradients," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1155–1164.

[9] G. Leclerc and A. Madry, "The two regimes of deep network training," 2020, *arXiv:2002.10376*.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Adv. Neural Inf. Process. Syst.*, vol. 25, pp. 1097–1105, 2012.

[11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015.

[12] J. M. Cohen, S. Kaur, Y. Li, Z. Kolter, and A. Talwalkar, "Gradient descent on neural networks typically occurs at the edge of stability," in *Proc. Int. Conf. Learn. Representations*, 2021, to be published.

[13] L. Sagun, U. Evci, V. U. Guney, Y. Dauphin, and L. Bottou, "Empirical analysis of the hessian of over-parametrized neural networks," in *Proc. Int. Conf. Learn. Representations*, 2018.

[14] S. Oymak, Z. Fabian, M. Li, and M. Soltanolkotabi, "Generalization guarantees for neural networks via harnessing the low-rank structure of the jacobian," 2019, *arXiv:1906.05392*.

[15] X. Li, Q. Gu, Y. Zhou, T. Chen, and A. Banerjee, "Hessian based analysis of sgd for deep nets: Dynamics and generalization," in *Proc. SIAM Int. Conf. Data Mining*, 2020, pp. 190–198.

[16] D. Paul, "Asymptotics of sample eigenstructure for a large dimensional spiked covariance model," *Statistica Sinica*, vol. 17, pp. 1617–1642, 2007.

[17] M. Li, M. Soltanolkotabi, and S. Oymak, "Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2020, pp. 4313–4324.

[18] D. Kopitkov and V. Indelman, "Neural spectrum alignment: Empirical study," in *Proc. Int. Conf. Artif. Neural Netw.*, 2020, pp. 168–179.

[19] A. Jacot, F. Gabriel, and C. Hongler, "Neural tangent kernel: Convergence and generalization in neural networks," in *Proc. Conf. Neural Inf. Process. Syst.*, pp. 8580–8589, 2018.

[20] J. Lee *et al.*, "Wide neural networks of any depth evolve as linear models under gradient descent," *Adv. Neural Inf. Process. Syst.*, vol. 32, pp. 8572–8583, 2019.

[21] G. Gur-Ari, D. A. Roberts, and E. Dyer, "Gradient descent happens in a tiny subspace," 2018, *arXiv:1812.04754*.

[22] M. Belkin, D. Hsu, S. Ma, and S. Mandal, "Reconciling modern machine-learning practice and the classical bias-variance trade-off," in *Proc. Nat. Acad. Sci. USA*, vol. 116, no. 32, pp. 15849–15854, 2019.

[23] M. Belkin, D. Hsu, and P. Mitra, "Overfitting or perfect fitting? Risk bounds for classification and regression rules that interpolate," *Adv. Neural Inf. Process. Syst.*, vol. 31, pp. 2300–2311, 2018.

[24] T. Poggio *et al.*, "Theory of deep learning III: Explaining the non-overfitting puzzle," 2017, *arXiv:1801.00173* .

[25] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, 2016, pp. 795–811.