OPT2022: 14th Annual Workshop on Optimization for Machine Learning

Self-Stabilization: The Implicit Bias of Gradient Descent at the Edge of Stability

Alex Damian* AD27@PRINCETON.EDU
Eshaan Nichani* ESHNICH@PRINCETON.EDU
Jason D. Lee JASONLEE@PRINCETON.EDU
Princeton University

Abstract

Traditional analyses of gradient descent with learning rate η show that when the largest eigenvalue of the Hessian of the loss, also known as the sharpness $S(\theta)$, is bounded by $2/\eta$, training is "stable" and the training loss decreases monotonically. However, Cohen et al. [7] recently observed two important phenomena. The first, *progressive sharpening*, is that the sharpness steadily increases throughout training until it reaches the instability cutoff $2/\eta$. The second, *edge of stability*, is that the sharpness hovers at $2/\eta$ for the remainder of training while the loss non-monotonically decreases. We demonstrate that, far from being chaotic, the dynamics of gradient descent at the edge of stability can be captured by a cubic Taylor expansion: as the iterates diverge in direction of the top eigenvector of the Hessian due to instability, the cubic term in the local Taylor expansion of the loss function causes the curvature to decrease until stability is restored. This property, which we call *self-stabilization*, is a general property of gradient descent and explains its behavior at the edge of stability. A key consequence of self-stabilization is that gradient descent at the edge of stability implicitly follows *projected* gradient descent (PGD) under the constraint $S(\theta) \leq 2/\eta$. Our analysis provides precise predictions for the loss, sharpness, and deviation from the PGD trajectory throughout training, which we verify both empirically in a number of standard settings and theoretically under mild conditions. Our analysis uncovers the mechanism for gradient descent's implicit bias towards stability.

1. Introduction

1.1. Gradient Descent at the Edge of Stability

Neural networks are often trained using variants of gradient descent such as stochastic gradient descent (SGD) or ADAM [17]. When deciding on an initial learning rate, practitioners draw intuition from the following classical lemma, known as the "descent lemma."

Definition 1 *Given a loss function $L(\theta)$, the sharpness at θ is defined to be $S(\theta) := \lambda_{max}(\nabla^2 L(\theta))$. When this eigenvalue is unique, the associated eigenvector is denoted by $u(\theta)$.*

Lemma 2 (Descent Lemma) *Assume that $S(\theta) \leq \ell$ for all θ. If $\theta_{t+1} = \theta_t - \eta \nabla L(\theta_t)$,*

$$L(\theta_{t+1}) \leq L(\theta_t) - \eta(2 - \eta\ell) \cdot \|\nabla L(\theta_t)\|^2/2.$$

Here, loss decrease is maximized at $\eta = 1/\ell$, a popular learning rate criterion. For any $\eta < 2/\ell$, the descent lemma guarantees that the loss will decrease, and hence learning rates below $2/\ell$ are

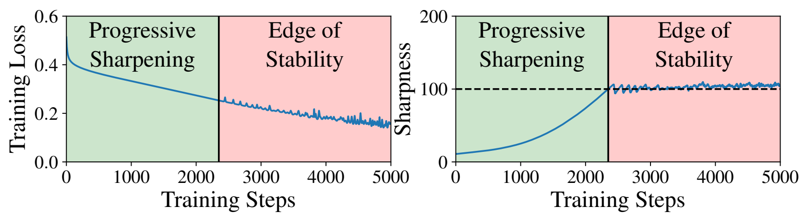
*. Equal contribution

considered "stable" while those above $2/\ell$ are considered "unstable." For quadratic loss functions, this is tight. Any learning rate above $2/\ell$ provably leads to exponentially increasing loss.

However, it has recently been observed that in neural networks, the descent lemma is not predictive of the optimization dynamics. Recently, Cohen et al. [7] observed two interesting phenomena:

Progressive Sharpening Throughout most of the optimization trajectory, the gradient of the loss is negatively aligned with the gradient of sharpness, i.e. $\nabla L(\theta) \cdot \nabla S(\theta) < 0$. As a result, for any reasonable learning rate η, the sharpness increases throughout training until it reaches $S(\theta) = 2/\eta$.

Edge of Stability Once the sharpness reaches $2/\eta$, it ceases to increase and remains around $2/\eta$ for the rest of training. Despite the fact that the descent lemma no longer guarantees the loss decreases, the loss still continues to rapidly decrease, albeit non-monotonically.



1.2. Self-stabilization: The Implicit Bias of Instability

In this work we explain the second stage, "edge of stability." We identify a new implicit bias of gradient descent which we call *self-stabilization*. Self-stabilization is the mechanism by which the sharpness remains bounded around $2/\eta$, despite the continued force of progressive sharpening, and by which the gradient descent dynamics do not diverge, despite instability.

Traditional non-convex optimization analyses involve Taylor expanding the loss function to second order around θ to prove loss decrease when $\eta \leq 2/S(\theta)$. When this is violated, the iterates diverge exponentially in the top eigenvector direction, u, thus leaving the region in which the loss is locally quadratic. Understanding the dynamics thus necessitates a *cubic* Taylor expansion.

Our key insight is that the missing term in the Taylor expansion of the gradient after diverging in the u direction is $\nabla^3 L(\theta)(u, u)$, which is exactly equal to the gradient of the sharpness at θ:

Lemma 3 (Self-Stabilization Property) *If the top eigenvalue of $\nabla^2 L(\theta)$ is unique, then the sharpness $S(\theta)$ is differentiable at θ and $\nabla S(\theta) = \nabla^3 L(\theta)(u(\theta), u(\theta))$.*

As the iterates move in the negative gradient direction, this term *decreases the sharpness*. So long as the iterates are unstable, the strength of this term grows until the sharpness goes below $2/\eta$, at which point the iterates in the u direction shrink and the dynamics re-enter the quadratic regime.

This negative feedback loop prevents both the sharpness $S(\theta)$ and the movement in the top eigenvector direction, u, from growing out of control. As a consequence, we show that gradient descent *implicitly* solves the *constrained minimization problem*: $\min_\theta L(\theta)$ such that $S(\theta) \leq 2/\eta$.

Specifically, if the stable set $\mathcal{M}$ is defined by $\mathcal{M} := \{\theta \,:\, S(\theta) \leq 2/\eta$ and $\nabla L(\theta) \cdot u(\theta) = 0\}$ then the gradient descent trajectory $\{\theta_t\}$ tracks the following projected gradient descent trajectory $\{\theta_t^\dagger\}$ which solves the constrained problem [3]:

$$\theta_{t+1}^\dagger = \operatorname{proj}_{\mathcal{M}}\left(\theta_t^\dagger - \eta \nabla L(\theta_t^\dagger)\right) \quad \text{where} \quad \operatorname{proj}_{\mathcal{M}}(\theta) := \arg\min_{\theta' \in \mathcal{M}} \left\|\theta - \theta'\right\|. \tag{1}$$

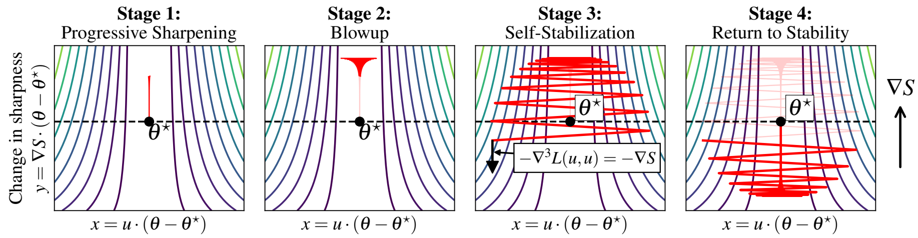


Figure 1: The four stages of edge of stability, demonstrated on a toy loss function (Appendix C).

Our main contribution is that we explain self-stabilization as a generic property of gradient descent for a large class of loss functions, and provide precise predictions for the loss, sharpness, and deviation from the constrained trajectory $\{\theta_t^\dagger\}$ throughout training (Section 3). We verify our predictions by replicating the experiments in Cohen et al. [7] and show that they model the true gradient descent dynamics (Section 4). In Appendix D, we prove that under mild conditions (which we verify empirically), our predictions track the true gradient descent dynamics up to higher order error terms.

2. Setup

We denote the loss function by $L \in C^3(\mathbb{R}^d)$. Let $\theta \in \mathbb{R}^d$ follow gradient descent with learning rate η, i.e. $\theta_{t+1} := \theta_t - \eta \nabla L(\theta_t)$. Recall that $\mathcal{M} := \{\theta : S(\theta) \leq 2/\eta \text{ and } \nabla L(\theta) \cdot u(\theta) = 0\}$ is the set of stable points and $\operatorname{proj}_{\mathcal{M}} := \arg\min_{\theta' \in \mathcal{M}} \|\theta - \theta'\|$ is the orthogonal projection onto $\mathcal{M}$. For notational simplicity, we will shift time so that θ_0 is the first point such that $S(\operatorname{proj}_{\mathcal{M}}(\theta)) = 2/\eta$. The constrained trajectory $\theta^\dagger$ is initialized with $\theta_0^\dagger := \operatorname{proj}_{\mathcal{M}}(\theta_0)$ after which it follows (1).

Our key assumption is the existence of progressive sharpening along the constrained trajectory, which is captured by $\alpha(\theta)$. We also assume that there is a single unstable eigenvalue.

Assumption 1 (Progressive Sharpening) *Let $\alpha(\theta) := -\nabla L(\theta) \cdot \nabla S(\theta)$. Then $\alpha(\theta_t^\dagger) > 0$.*
Assumption 2 (Eigengap) *For some absolute constant $c < 2$ we have $\lambda_2(\nabla^2 L(\theta_t^\dagger)) < c/\eta$.*

3. The Self-stabilization Property of Gradient Descent

3.1. The Four Stages of Edge of Stability: A Heuristic Derivation

In this section we present a heuristic derivation of the self-stabilization property (see Appendix D for the more general analysis). The derivation proceeds by a cubic Taylor expansion around a fixed reference point $\theta^\star := \theta_0^\dagger$. For notational simplicity, we will define the following quantities at $\theta^\star$:

$$\nabla L := \nabla L(\theta^\star),\ \nabla^2 L := \nabla^2 L(\theta^\star),\ u := u(\theta^\star),\ \nabla S := \nabla S(\theta^\star),\ \alpha := \alpha(\theta^\star),\ \beta := \|\nabla S\|^2,$$

where $\alpha = -\nabla L \cdot \nabla S > 0$ is the progressive sharpening. Our heuristic analysis assumes that $\nabla S \perp u$ and $\nabla L, \nabla S \in \ker(\nabla^2 L)$, and ignores higher order error terms.[1]

We track the movement in the unstable direction u and the direction of changing sharpness ∇S, and thus define $x_t := u \cdot (\theta_t - \theta^\star)$ and $y_t := \nabla S \cdot (\theta_t - \theta^\star)$. Note that y_t is approximately to the change in sharpness from $\theta^\star$ to θ_t, since Taylor expanding the sharpness yields $S(\theta_t) \approx S(\theta^\star) + \nabla S \cdot (\theta_t - \theta^\star) = 2/\eta + y_t$. At a high level, the mechanism for edge of stability can be described in 4 stages (see Figure 1):

1. We give an explicit example of a loss function satisfying these assumptions in Appendix C.

Stage 1: Progressive Sharpening While x, y are small, $\nabla L(\theta_t) \approx \nabla L$. In addition, because $\nabla L \cdot \nabla S < 0$, gradient descent naturally increases the sharpness at every step. In particular,

$$y_{t+1} - y_t = \nabla S \cdot (\theta_{t+1} - \theta_t) \approx -\eta \nabla L \cdot \nabla S = \eta\alpha.$$

The sharpness therefore increases linearly with rate $\eta\alpha$.

Stage 2: Blowup As x_t measures the deviation from $\theta^\star$ in the u direction, the dynamics of x_t can be modeled by gradient descent on a quadratic with sharpness $S(\theta_t) \approx 2/\eta + y_t$. In particular, the rule for gradient descent on a quadratic gives

$$x_{t+1} = x_t - \eta u \cdot \nabla L(\theta_t) \approx x_t - \eta S(\theta_t) x_t \approx x_t - \eta[2/\eta + y_t]x_t = -(1 + \eta y_t)x_t.$$

When the sharpness exceeds $2/\eta$, i.e. when $y_t > 0$, $|x_t|$ begins to grow exponentially.

Stage 3: Self-Stabilization Once the movement in the u direction is sufficiently large, the loss is no longer locally quadratic. Understanding the dynamics necessitates a third order Taylor expansion. The missing cubic term in the Taylor expansion of $\nabla L(\theta_t)$ is $\nabla^3 L(u, u)\frac{x_t^2}{2} = \nabla S \frac{x_t^2}{2}$ by Lemma 3. This biases the optimization trajectory in the $-\nabla S$ direction, which decreases sharpness. Recalling $\beta = \|\nabla S\|^2$, the new update for y becomes:

$$y_{t+1} - y_t = \eta\alpha + \nabla S \cdot (-\eta \nabla^3 L(u, u) x_t^2/2) = \eta\big(\alpha - \beta x_t^2/2\big)$$

Therefore once $x_t > \sqrt{2\alpha/\beta}$, the sharpness begins to decrease and continues to do so until the sharpness goes below $2/\eta$ and the dynamics return to stability.

Stage 4: Return to Stability At this point $|x_t|$ is still large from stages 1 and 2. However, the self-stabilization of stage 3 eventually drives the sharpness below $2/\eta$ so that $y_t < 0$. Because the rule for gradient descent on a quadratic with sharpness $S(\theta_t) = 2/\eta + y_t < 2/\eta$ is $x_{t+1} \approx -(1 + \eta y_t)x_t$, $|x_t|$ begins to shrink exponentially and the process returns to stage 1.

Combining the update for x_t, y_t in all four stages, we obtain the following simplified dynamics:

$$x_{t+1} \approx -(1 + \eta y_t)x_t \quad \text{and} \quad y_{t+1} \approx y_t + \eta(\alpha - \beta x_t^2/2) \tag{2}$$

3.2. Analyzing the simplified dynamics

We now analyze the dynamics in (2). First, note that x_t changes sign at every iteration, and that $x_{t+1} \approx -x_t$. While (2) cannot be modeled by an ODE due to these oscillations, we can instead model $|x_t|, y_t$. We couple the dynamics of $|x_t|, y_t$ to the following ODE $X(t), Y(t)$:

$$X'(t) = X(t)Y(t) \quad \text{and} \quad Y'(t) = \alpha - \beta X(t)^2/2. \tag{3}$$

This system has the unique fixed point $(X, Y) = (\delta, 0)$ where $\delta := \sqrt{2\alpha/\beta}$. Furthermore, we can prove conservation of a quantity g,which allows us to explicitly bound the size of the trajectory:

Lemma 4 *Let* $h(z) := z - \log z - 1$. *Then* $g(X(t), Y(t)) := h\left(\frac{\beta X(t)^2}{2\alpha}\right) + \frac{Y(t)^2}{\alpha}$ *is conserved.*

Proof $\frac{d}{dt} g(X(t), Y(t)) = \frac{\beta X(t)^2 Y(t)}{\alpha} - 2Y(t) + \frac{2}{\alpha} Y(t)\left[\alpha - \beta \frac{X(t)^2}{2}\right] = 0.$ ∎

Corollary 5 *For all t,* $X(0) \le X(t) \lesssim \delta\sqrt{\log(\delta/X(0))}$ *and* $|Y(t)| \lesssim \sqrt{\alpha \log(\delta/X(0))}$.

The fluctuations are $\tilde{O}(\sqrt{\alpha})$ in the sharpness and $\tilde{O}(\delta)$ in the unstable direction. Moreover, the *normalized* displacement in the ∇S direction, is also bounded by $\tilde{O}(\delta)$, so the entire process remains bounded by $\tilde{O}(\delta)$. Note that the fluctuations increase as α grows, and decrease as β grows.

3.3. Relationship with the constrained trajectory $\theta_t^\dagger$

(2) determines the displacement $\theta_t - \theta^\star$ in the $u, \nabla S$ directions. However, θ_t still evolves in the orthogonal directions by $-\eta P_{u,\nabla S}^\perp \nabla L$ at every step. This can be interpreted as taking a gradient step of $-\eta \nabla L$ and then projecting out the ∇S direction to ensure the sharpness does not change. Lemma 16, shows that this is precisely the update for $\theta_t^\dagger$ ((1)) up to higher order terms. Therefore $\|\theta_t - \theta_t^\dagger\| \le \tilde{O}(\delta)$ and that this $\tilde{O}(\delta)$ error term is controlled by the self-stabilizing dynamics in (2).

4. Experiments

Appendix D introduces a generalization of eq. (2). In Figure 2, we replicate the experiments in [7]. We show that these dynamics accurately model gradient descent at the edge of stability and can predict the loss, sharpness, and distance from constrained trajectory. In addition, while gradient flow diverges linearly from gradient descent, the gradient descent and constrained trajectories remain close *throughout training*. See Appendix F for details and Appendix I for more experiments.

5. Discussion

Non-Monotonic Loss Decrease Cohen et al. [7] observed that, despite non-monotonic fluctuations of the loss, the loss still decreases over long time scales. Our analysis explains this decrease by showing that gradient descent remains close to the constrained trajectory. Since this trajectory is *stable*, it satisfies a descent lemma (Lemma 17), and has monotonically decreasing loss. Over short time periods, the loss is dominated by the rapid fluctuations of x_t described in Section 3. Over longer time periods, the loss decrease of the constrained trajectory overpowers the bounded fluctuations of x_t, leading to an overall loss decrease. This is reflected in our experiments in Section 4.

Generalization & the Role of Large Learning Rates Prior work shows that decreasing sharpness of the learned solution [9, 15, 16, 24] and increasing the learning rate [18, 19, 26] are correlated with better generalization. Our analysis shows that gradient descent implicitly constrains the sharpness to stay near $2/\eta$, which suggests larger learning improves generalization by reducing the sharpness. In Figure 3 we confirm that gradient descent generalizes better with large learning rates.

Training Speed [7] also shows that larger learning rates lead to faster convergence despite instability. This phenomenon is explained by our analysis. Gradient descent is coupled to the constrained trajectory which minimizes the loss while constraining movement in the $u_t, \nabla S_t^\perp$ directions. However, the constrained trajectory can still move quickly in the orthogonal directions, using the large learning rate to accelerate convergence. We demonstrate this empirically in Figure 3.

We defer additional discussion of our work, including a generalization of the predicted dynamics in Appendix H, and the effect of multiple unstable eigenvalues and connections to Sharpness Aware Minimization [10], warm-up [11], and scale-invariant loss functions [22] to Appendix J.

Future Work An important question is to understand the dynamics when there are multiple unstable eigenvalues, which we address in Appendix J. Another direction is to understand the *global* convergence properties at the edge of stability, including convergence to a KKT point of the constrained update (1). Next, our analysis focused on the edge of stability dynamics but left open the question of why progressive sharpening occurs. Finally, we would like to understand how self-stabilization interacts with the implicit biases of *stochastic*-gradient descent [4, 8, 20].

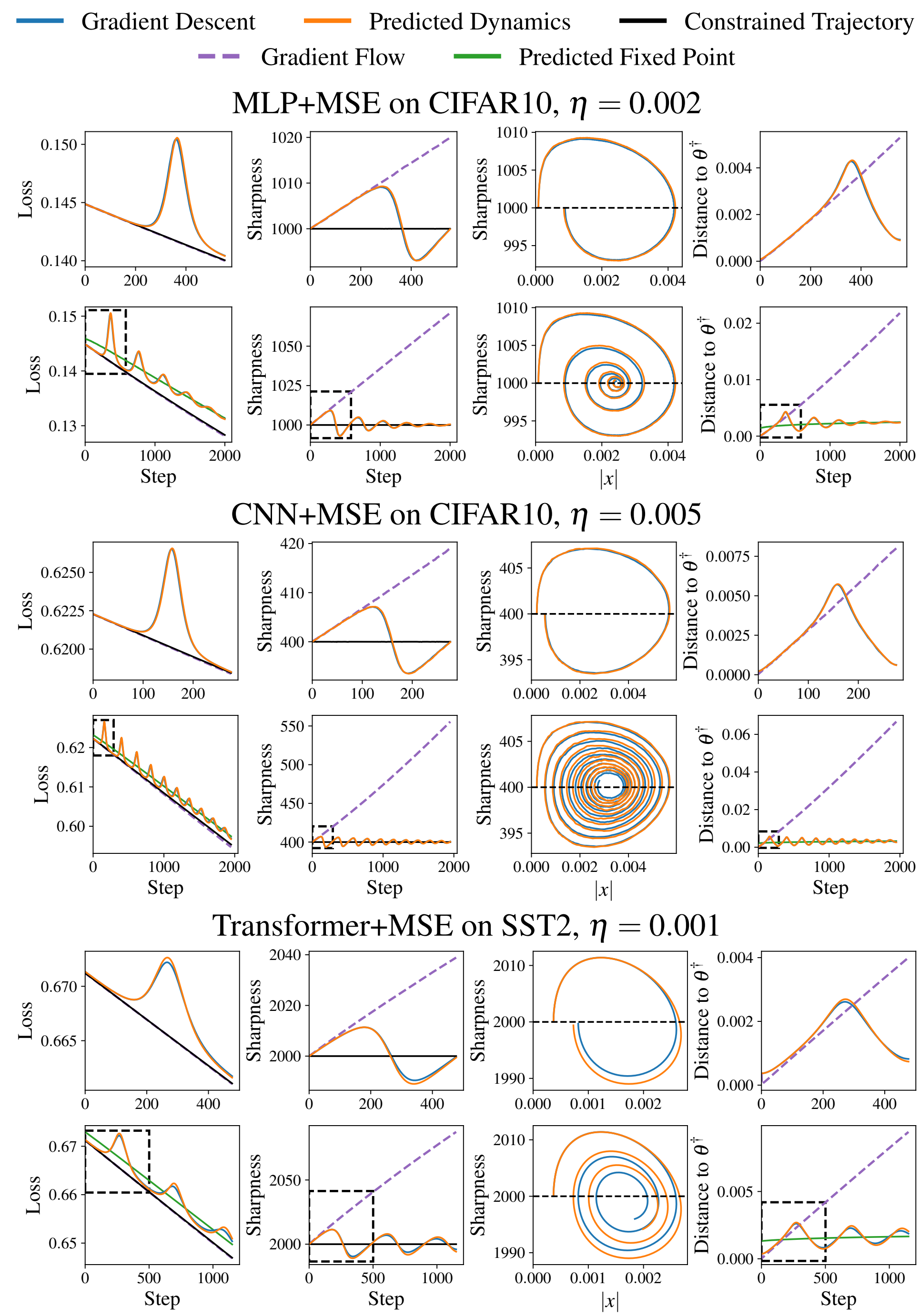


Figure 2: We empirically demonstrate that the predicted dynamics given by (4) track the true dynamics of gradient descent at the edge of stability. For each learning rate, the top row is a zoomed in version of the bottom row which isolates one cycle and is reflected by the dashed rectangle in the bottom row. Reported sharpnesses are two-step averages for visual clarity. For additional experimental details, see Section 4 and Appendix F.

References

[1] Kwangjun Ahn, J. Zhang, and Suvrit Sra. Understanding the unstable convergence of gradient descent. *ArXiv*, abs/2204.01050, 2022.

[2] Sanjeev Arora, Zhiyuan Li, and Abhishek Panigrahi. Understanding gradient descent on the edge of stability in deep learning. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 948–1024. PMLR, 2022.

[3] Rina Foygel Barber and Wooseok Ha. Gradient descent with nonconvex constraints: local concavity determines convergence. *arXiv: Optimization and Control*, 2017.

[4] Guy Blanc, Neha Gupta, Gregory Valiant, and Paul Valiant. Implicit regularization for deep neural networks driven by an ornstein-uhlenbeck like process. In *Conference on Learning Theory*, pages 483–513, 2020.

[5] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

[6] Lei Chen and Joan Bruna. On gradient descent convergence beyond the edge of stability. *ArXiv*, abs/2206.04172, 2022.

[7] Jeremy Cohen, Simran Kaur, Yuanzhi Li, J Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. In *International Conference on Learning Representations*, 2021.

[8] Alex Damian, Tengyu Ma, and Jason D. Lee. Label noise SGD provably prefers flat global minimizers. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[9] Gintare Karolina Dziugaite and Daniel M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. In *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence*, 2017.

[10] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021.

[11] Justin Gilmer, Behrooz Ghorbani, Ankush Garg, Sneha Kudugunta, Behnam Neyshabur, David Cardoze, George Dahl, Zachary Nado, and Orhan Firat. A loss curvature perspective on training instability in deep learning, 2021. URL https://arxiv.org/abs/2110.04369.

[12] Stanislaw Jastrzebski, Zachary Kenton, Devansh Arpit, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos J. Storkey. Three factors influencing minima in sgd. *ArXiv*, abs/1711.04623, 2017.

[13] Stanislaw Jastrzebski, Maciej Szymczak, Stanislav Fort, Devansh Arpit, Jacek Tabor, Kyunghyun Cho, and Krzysztof Geras. The break-even point on optimization trajectories of deep neural networks. In *International Conference on Learning Representations*, 2020.

[14] Stanisław Jastrzębski, Zachary Kenton, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amost Storkey. On the relation between the sharpest directions of DNN loss and the SGD step length. In *International Conference on Learning Representations*, 2019.

[15] Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *International Conference on Learning Representations*, 2020.

[16] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017.

[17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

[18] Aitor Lewkowycz, Yasaman Bahri, Ethan Dyer, Jascha Sohl-Dickstein, and Guy Gur-Ari. The large learning rate phase of deep learning: the catapult mechanism. *ArXiv*, abs/2003.02218, 2020.

[19] Yuanzhi Li, Colin Wei, and Tengyu Ma. Towards explaining the regularization effect of initial large learning rate in training neural networks. In *Advances in Neural Information Processing Systems*, 2019.

[20] Zhiyuan Li, Tianhao Wang, and Sanjeev Arora. What happens after SGD reaches zero loss? –a mathematical framework. In *International Conference on Learning Representations*, 2022.

[21] Zhouzi Li, Zixuan Wang, and Jian Li. Analyzing sharpness along gd trajectory: Progressive sharpening and edge of stability. *arXiv*, abs/2207.12678, 2022.

[22] Kaifeng Lyu, Zhiyuan Li, and Sanjeev Arora. Understanding the generalization benefit of normalization layers: Sharpness reduction. *ArXiv*, abs/2206.07085, 2022.

[23] Chao Ma, Lei Wu, and Lexing Ying. The multiscale structure of neural network loss functions: The effect on optimization and origin. *ArXiv*, abs/2204.11326, 2022.

[24] Behnam Neyshabur, Srinadh Bhojanapalli, David Mcallester, and Nati Srebro. Exploring generalization in deep learning. In *Advances in Neural Information Processing Systems*, 2017.

[25] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. Swish: a self-gated activation function. *arXiv: Neural and Evolutionary Computing*, 2017.

[26] Samuel L. Smith, Pieter-Jan Kindermans, and Quoc V. Le. Don't decay the learning rate, increase the batch size. In *International Conference on Learning Representations*, 2018.

[27] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D13-1170.

[28] Lei Wu, Chao Ma, and E Weinan. How sgd selects the global minima in over-parameterized learning: A dynamical stability perspective. In *NeurIPS*, 2018.

[29] Chen Xing, Devansh Arpit, Christos Tsirigotis, and Yoshua Bengio. A walk with sgd. *ArXiv*, abs/1802.08770, 2018.

Table of Contents

Appendix A. Related Work

Cohen et al. [7] conducted an extensive empirical study showing progressive sharpening and edge of stability in a wide range of models. Prior work [28, 29] had also observed that for neural networks full-batch gradient descent reaches instability and the loss is not monotonically decreasing. Lewkowycz et al. [18] observed that when the initial sharpness is larger than $2/\eta$, gradient descent "catapults" into a stable region and converges.

Recent works have sought to provide a theoretical analysis for the edge of stability phenomenon. Ma et al. [23] analyzes edge of stability when the loss satisfies a "subquadratic growth" assumption. Ahn et al. [1] argues that unstable convergence is possible when there exists a "forward invariant subset" near the set of minimizers. Arora et al. [2] analyzes progressive sharpening and the edge of stability phenomenon for normalized gradient descent close to the manifold of global minimizers. Lyu et al. [22] uses the edge of stability phenomenon to analyze the effect of normalization layers on sharpness for scale-invariant loss functions. Chen and Bruna [6] show global convergence despite instability for certain 2D toy problems and in a 1-neuron student-teacher setting. The concurrent work Li et al. [21] proves progressive sharpening for a two-layer network and analyzes the edge of stability dynamics through four stages similar to ours using the norm of the output layer as a proxy for sharpness.

Beyond the edge of stability phenomenon itself, prior work has also shown that SGD with large step size or small batch size will lead to a decrease in sharpness [12–14, 16]. Gilmer et al. [11] also describes connections between edge of stability, learning rate warm-up, and gradient clipping.

At a high level, our proof relies on the idea that oscillations in an unstable direction prescribed by the quadratic approximation of the loss cause a longer term effect arising from the third-order Taylor expansion of the dynamics. This overall idea has also been used to analyze the implicit regularization of SGD [4, 8, 20]. In those settings, oscillations come from the stochasticity, while in our setting the oscillations stem from instability.

Appendix B. Notation

We denote by $\nabla^k L(\theta)$ the k-th order derivative of the loss L at θ. Note that $\nabla^k L(\theta)$ is a symmetric k-tensor in $(\mathbb{R}^d)^{\otimes k}$ when $\theta \in \mathbb{R}^d$.

For a symmetric k-tensor T, and vectors $u_1, \ldots, u_j \in \mathbb{R}^d$ we will use $T(u_1, \ldots, u_j)$ to denote the tensor contraction of T with $u_1, \ldots, u_j$, i.e.

$$[T(u_1, \ldots, u_k)]_{i_1,\ldots,i_{k-j}} := T_{i_1,\ldots,i_k}(u_1)_{i_{k-j+1}} \cdots (u_j)_{i_k}.$$

We use $P_{u_1,\ldots,u_k}$ to denote the orthogonal projection onto $\operatorname{span}(u_1, \ldots, u_k)$ and $P^\perp_{u_1,\ldots,u_k}$ is the projection onto the corresponding orthogonal complement.

For matrices $A_1, \ldots, A_k$, we define

$$\prod_{k=1}^{t} A_k := A_1 \ldots A_t \quad \text{and} \quad \prod_{k=t}^{1} A_k := A_t \ldots A_1.$$

Appendix C. A Toy Model for Self-Stabilization

For $\alpha, \beta > 0$, consider the function

$$L(x, y, z) := \left(\frac{2}{\eta} + \sqrt{\beta} y\right) \frac{x^2}{2} - \frac{\alpha}{\sqrt{\beta}} y - z$$

initialized at the point $(x_0, 0, 0)$. Note that the constrained trajectory will follow $x_t^\dagger = 0$, $y_t^\dagger = 0$, $z_t^\dagger = -\eta t$ as it cannot decrease y without increasing the sharpness past $2/\eta$. We therefore have:

$$\nabla L_t = \left[0, -\frac{\alpha}{\sqrt{\beta}}, 1\right],\ u_t = [1, 0, 0],\ S_t = 2/\eta + \sqrt{\beta} y,\ \nabla^2 L_t = S_t u_t u_t^t,\ \nabla S_t = \left[0, \sqrt{\beta}, 0\right].$$

Note that this satisfies all of the assumptions in Section 3 and it satisfies $\alpha = -\nabla L_t \cdot \nabla S_t = 0$ and $\beta = \|\nabla S_t\|^2$. This process will then follow (3) in the x, y directions while it tracks the constrained trajectory $\theta_t^\dagger$ moving linearly in the $-P_{u,\nabla S}^\perp \nabla L = [0, 0, -1]$ direction.

Appendix D. The Predicted Dynamics and Theoretical Results

We now present the equations governing edge of stability for general loss functions.

D.1. Notation

Our general approach Taylor expands the gradient of each iterate θ_t around the corresponding iterate $\theta_t^\dagger$ of the constrained trajectory. We define the following Taylor expansion quantities at $\theta_t^\dagger$:

Definition 6 (Taylor Expansion Quantities at $\theta_t^\dagger$)

$$\nabla L_t := \nabla L(\theta_t^\dagger), \quad \nabla^2 L_t := \nabla^2 L(\theta_t^\dagger), \quad \nabla^3 L_t := \nabla^3 L(\theta_t^\dagger), \quad \nabla S_t := \nabla S(\theta_t^\dagger), \quad u_t := u(\theta_t^\dagger).$$

Furthermore, for any vector-valued function $v(\theta)$, we define $v_t^\perp := P_{u_t}^\perp v(\theta_t^\dagger)$ where $P_{u_t}^\perp$ is the projection onto the orthogonal complement of u_t.

We also define the following quantities which govern the dynamics near $\theta_t^\star$.

Definition 7 *Let $\alpha_t := -\nabla L_t \cdot \nabla S_t$, $\beta_t := \left\|\nabla S_t^\perp\right\|^2$, and $\delta_t := \sqrt{\frac{2\alpha_t}{\beta_t}}$. Furthermore, we define $\beta_{s\to t} := \nabla S_{t+1}^\perp \left[\prod_{k=t}^{s+1} (I - \eta \nabla^2 L_k) P_{u_k}^\perp\right] \nabla S_s^\perp$ and $\delta := \sup_t \delta_t$.*

Recall that α_t is the progressive sharpening force, β_t is the strength of the stabilization force, and δ_t controls the size of the deviations from $\theta_t^\dagger$ and was the fixed point in the x direction in Section 3.2. The scalars $\beta_{s\to t}$ capture the effect of the interactions between ∇S and the Hessian.

D.2. The equations governing edge of stability

We now introduce the equations governing edge of stability. We track the following quantities:

Definition 8 *Define $v_t := \theta_t - \theta_t^\dagger$, $x_t := u_t \cdot v_t$, $y_t := \nabla S_t^\perp \cdot v_t$.*

Our predicted dynamics directly predict the displacement v_t and the full definition is deferred to Appendix E. However, they have a relatively simple form in the $u_t, \nabla S_t^\perp$ directions:

Lemma 9 (Predicted Dynamics for x, y) *Let $\overset{\star}{v}_t$ denote our predicted dynamics (defined in Appendix E). Letting $\overset{\star}{x}_t = u_t \cdot \overset{\star}{v}_t$ and $\overset{\star}{y}_t = \nabla S_t^\perp \cdot \overset{\star}{v}_t$, we have*

$$\overset{\star}{x}_{t+1} = -(1 + \eta \overset{\star}{y}_t)\overset{\star}{x}_t \quad \textit{and} \quad \overset{\star}{y}_{t+1} = \eta \sum_{s=0}^{t} \beta_{s \to t} \left[\frac{\delta_s^2 - \overset{\star}{x}_s{}^2}{2} \right]. \tag{4}$$

Note that when $\beta_{s \to t}$ are constant, our update reduces to the simple case discussed in Section 3, which we analyze fully. When x_t is large, (4) demonstrates that there is a self-stabilization force which acts to decrease y_t; however, unlike in Section 3, the strength of this force changes with t.

D.3. Coupling Theorem

We now show that, under a mild set of assumptions which we verify to hold empirically in Appendix G, the true dynamics are accurately governed by the predicted dynamics. This lets us use the predicted dynamics to predict the loss, sharpness, and the distance to the constrained trajectory $\theta_t^\dagger$.

Our errors depend on the unitless quantity ϵ, which we verify is small in Appendix G.

Definition 10 *Let $\epsilon_t := \eta\sqrt{\alpha_t}$ and $\epsilon := \sup_t \epsilon_t$.*

To control Taylor expansion errors, we require upper bounds on $\nabla^3 L$ and its Lipschitz constant:[2]

Assumption 3 *Let ρ_3, ρ_4 to be the minimum constants such that for all θ, $\left\|\nabla^3 L(\theta)\right\|_{op} \le \rho_3$ and $\nabla^3 L$ is ρ_4-Lipschitz with respect to $\|\cdot\|_{op}$. Then we assume that $\rho_4 = O(\eta\rho_3^2)$.*

Next, we require the following generalization of Assumption 1:

Assumption 4 *For all t, $\frac{-\nabla L_t \cdot \nabla S_t}{\|\nabla L_t\| \left\|\nabla S_t^\perp\right\|} = \Theta(1)$ and $\left\|\nabla S_t^\perp\right\| = \Theta(\rho_3)$.*

Finally, we require a set of "non-worst-case" assumptions, which are that the quantities $\nabla^2 L$, $\nabla^3 L$, and $\lambda_{min}(\nabla^2 L)$ are nicely behaved in the directions orthogonal to u_t, which generalizes the eigengap assumption. We verify the assumptions on $\nabla^2 L$ and $\nabla^3 L$ empirically in Appendix G.

Assumption 5 *For all t and $v, w \perp u_t$, $\frac{\left\|\nabla^3 L_t(v,w)\right\|}{\|\nabla^3 L_t\|_{op}\|v\|\|w\|}, \frac{|\nabla^2 L_t(\overset{\star}{v}_t^\perp, \overset{\star}{v}_t^\perp)|}{\|\nabla^2 L_t\| \left\|\overset{\star}{v}_t^\perp\right\|^2}, \frac{\left|\lambda_{min}(\nabla^2 L_t)\right|}{\|\nabla^2 L_t\|_2} \le O(\epsilon)$.*

With these assumptions in place, we can state our main theorem which guarantees $\overset{\star}{x}, \overset{\star}{y}, \overset{\star}{v}$ predict the loss, sharpness, and deviation from the constrained trajectory up to higher order terms:

Theorem 11 *Let $\mathscr{T} := O(\epsilon^{-1})$ and assume that $\min_{t \le \mathscr{T}} |\overset{\star}{x}_t| \ge c_1\delta$. Then for any $t \le \mathscr{T}$, we have*

$$L(\theta_t) = L(\theta_t^\dagger) + \overset{\star}{x}_t^2/\eta + O\big(\epsilon\delta^2/\eta\big) \tag{Loss}$$
$$S(\theta_t) = 2/\eta + \overset{\star}{y}_t + (S_t \cdot u_t)\overset{\star}{x}_t + O\big(\epsilon^2/\eta\big) \tag{Sharpness}$$
$$\theta_t = \theta_t^\dagger + \overset{\star}{v}_t + O(\epsilon\delta) \tag{Deviation from $\theta^\dagger$}$$

2. For simplicity of exposition, we make these bounds on $\nabla^3 L$ globally, however our proof only requires them in a small neighborhood of the constrained trajectory $\theta^\dagger$.

The sharpness is controlled by the slowly evolving quantity $\overset{\star}{y}_t$ and the period-2 oscillations of $(\nabla S \cdot U)\overset{\star}{x}_t$. This combination of gradual and rapid periodic behavior was observed by Cohen et al. [7] and appears in our experiments. Theorem 11 also shows that the loss at θ_t spikes whenever $\overset{\star}{x}_t$ is large. On the other hand, when $\overset{\star}{x}_t$ is small, $L(\theta_t)$ approaches the loss of the constrained trajectory.

Appendix E. Definition of the Predicted Dynamics

Below, we present the full definition of the predicted dynamics:

Definition 12 (Predicted Dynamics, full) *Define* $\overset{\star}{v}_0 = v_0$, *and let* $\overset{\star}{x}_t = \overset{\star}{v}_t \cdot u_t, \overset{\star}{y}_t = \nabla S^\perp \cdot \overset{\star}{v}_t$. *Then*

$$v^*_{t+1} = P^\perp_{u_{t+1}}(I - \eta\nabla^2 L_t)P^\perp_{u_t} v^*_t + \eta P^\perp_{u_{t+1}} \nabla S^\perp_t \left[\frac{\delta_t^2 - x_t^{*2}}{2}\right] - (1 + \eta y^*_t)x^*_t \cdot u_{t+1} \tag{5}$$

For convenience, we will define the map $\mathrm{step}_t : \mathbb{R}^d \to \mathbb{R}^d$ as follows:

Definition 13 *Given a vector* v *and a timestep* t*, define* $\mathrm{step}_t(v)$ *by*

$$P^\perp_{u_{t+1}} \mathrm{step}_t(v) = P^\perp_{u_{t+1}}\left[(I - \eta\nabla^2 L_t)P^\perp_{u_t} v + \eta\nabla S^\perp_t \left[\frac{\delta_t^2 - x^2}{2}\right]\right] \tag{6}$$

$$u_{t+1} \cdot \mathrm{step}_t(v) = -(1 + \eta y)x. \tag{7}$$

where $x = u_t \cdot v$ *and* $y = \nabla S^\perp_t \cdot v$.

It is easy to see that $\overset{\star}{v}_{t+1} = \mathrm{step}_t(\overset{\star}{v}_t)$.

Proof [Proof of Theorem 9] Defining $A_t = (I - \eta\nabla^2 L_t)P^\perp_{u_t}$, we can unfold the recursion in (5) to obtain the following formula for $\overset{\star}{v}_t$.

$$v^*_{t+1} = \eta \sum_{s=0}^{t} P^\perp_{u_{t+1}} \left[\prod_{k=t}^{s+1} A_k\right] \nabla S^\perp_s \left[\frac{\delta_s^2 - x_s^{*2}}{2}\right] - (1 + \eta y^*_t)x^*_t \cdot u_{t+1}. \tag{8}$$

It is then immediate to see that $\overset{\star}{x}_t = \overset{\star}{v}_t \cdot u_t, \overset{\star}{y}_t = \nabla S^\perp_t \cdot \overset{\star}{v}_t$ have the following simple update:

$$x^*_{t+1} = -(1 + \eta y^*_t)x^*_t \quad \text{and} \quad y^*_{t+1} = \eta \sum_{s=0}^{t} \beta_{s \to t}\left[\frac{\delta_s^2 - x_s^{*2}}{2}\right],$$

where we recall that we have defined

$$\beta_{s \to t} := \nabla S^\perp_{t+1} \left[\prod_{k=t}^{s+1} A_k\right] \nabla S^\perp_s. \tag{9}$$

■

Appendix F. Experimental Details

F.1. Architectures

We evaluated our theory on four different architectures. The 3-layer MLP and CNN are exact copies of the MLP and CNN used in [7]. The MLP has width 200, the CNN has width 32, and both are using the swish activation [25]. We also evaluate on a ResNet18 with progressive widths $16, 32, 64, 128$ and on a 2-layer Transformer with hidden dimension 64 and two attention heads.

F.2. Data

We evaluated our theory on three primary tasks: CIFAR10 multi-class classification with both categorical MSE loss and cross-entropy loss, CIFAR10 binary classification (cats vs dogs) with binary MSE loss and logistic loss, and SST2 [27] with binary MSE loss and logistic loss.

F.3. Experimental Setup

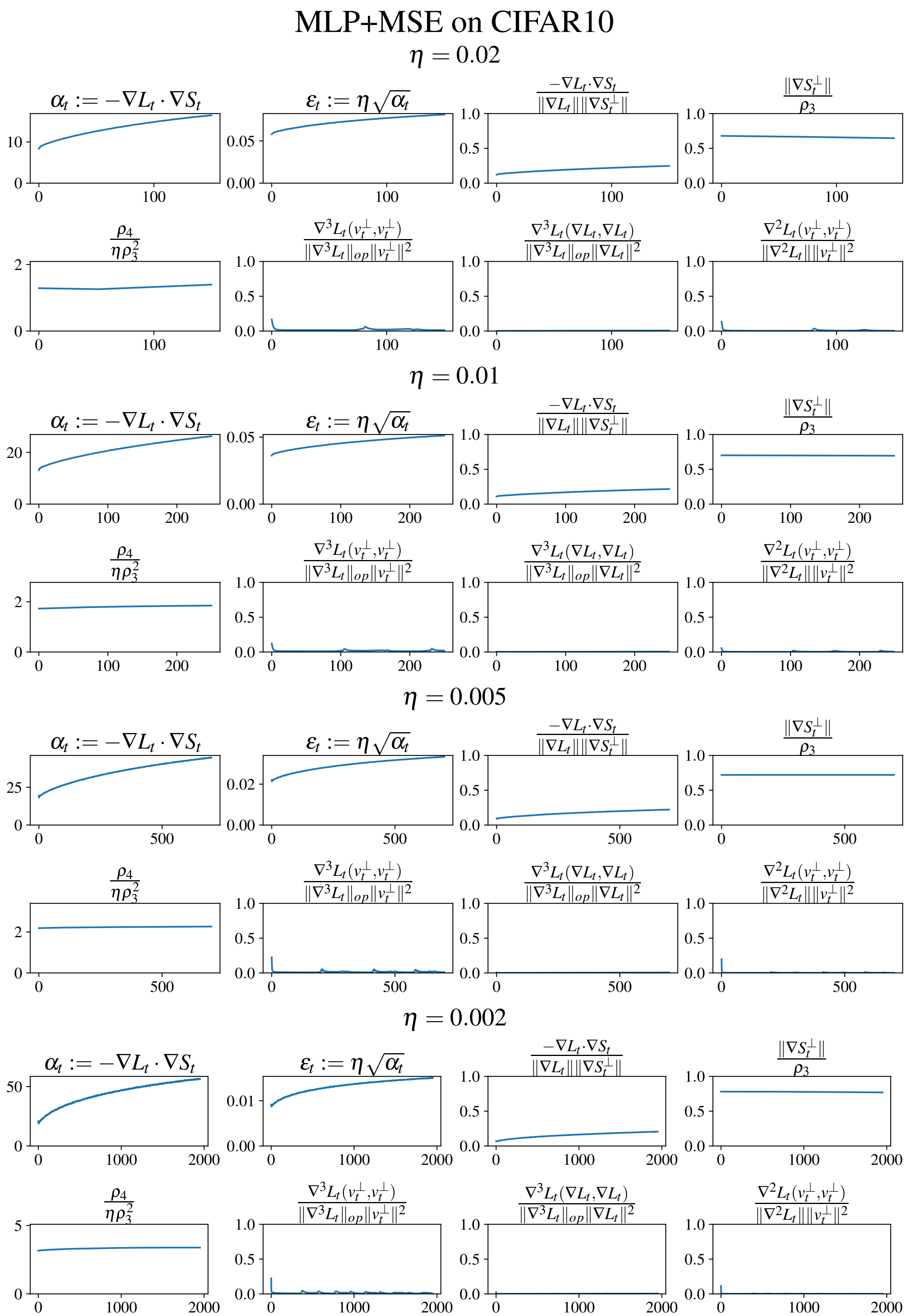
For every experiment, we tracked the gradient descent dynamics until they reached instability and then began tracking the constrained trajectory, gradient descent, gradient flow, and both our predicted dynamics (Appendix D) and our generalized predicted dynamics (Appendix H). In addition, we tracked the various quantities on which we made assumptions for Appendix D in order to validate these assumptions. We also tracked the second eigenvalue of the Hessian at the constrained trajectory throughout training and stopped training once it reached $1.9/\eta$, to ensure the existence of a single unstable eigenvalue. Finally, as the edge of stability dynamics are very sensitive to small perturbation when $|x|$ is small (see **??**), we switched to computing gradients with 64-bit precision after first reaching instability to avoid propagating floating point errors.

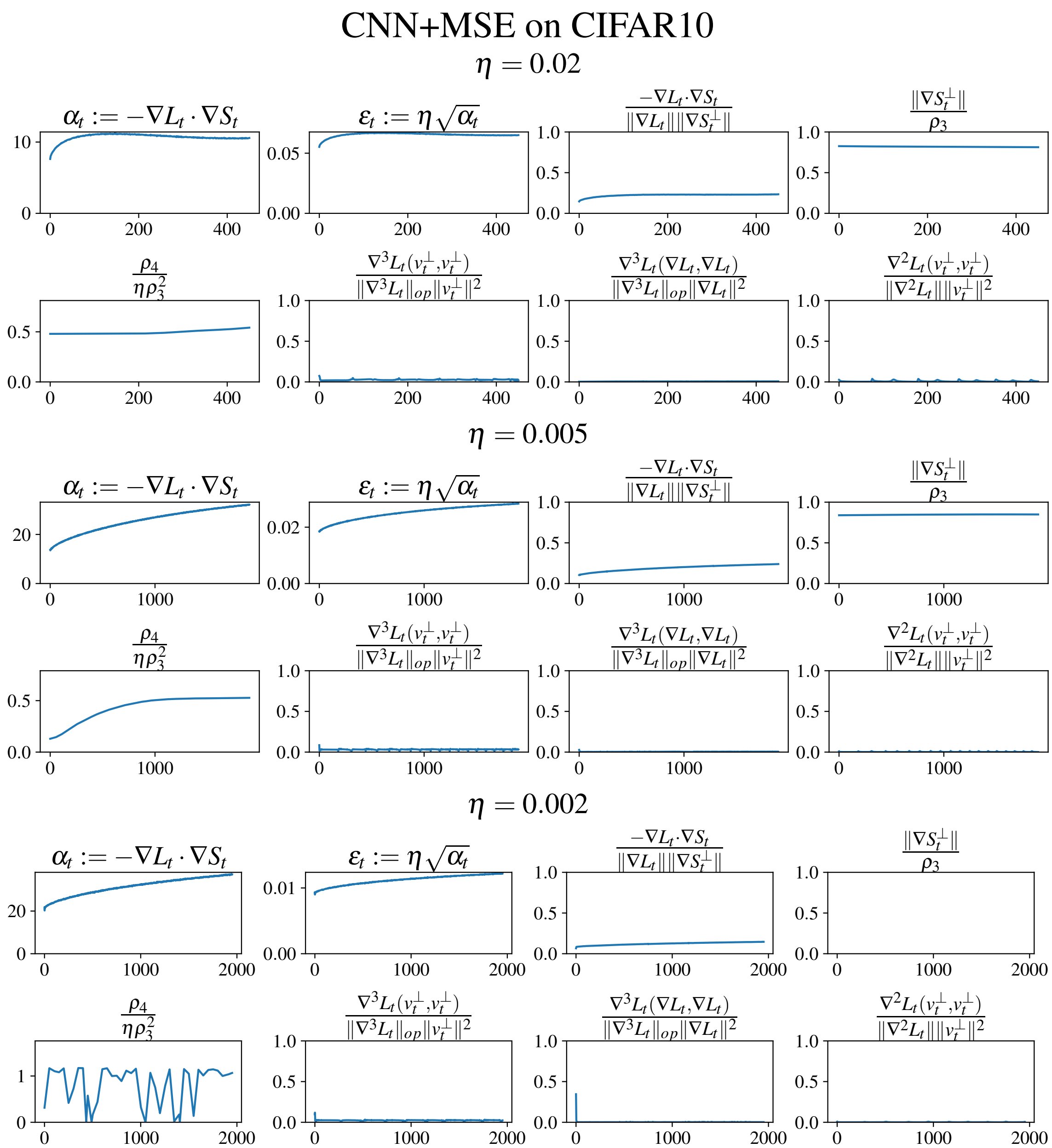
Eigenvalues were computed using the LOBPCG sparse eigenvalue solver in JAX [5]. To compute the constrained trajectory, we computed a linearized approximation for $\mathrm{proj}_{\mathcal{M}}$ inspired by Lemma 16 along with a Newton step in the u_t direction to ensure that $\nabla L \cdot u = 0$. Each linearized approximation step required recomputing the sharpness and top eigenvector and each projection step then consisted of three linearized projection steps, for a total of three eigenvalue computations per projection step.

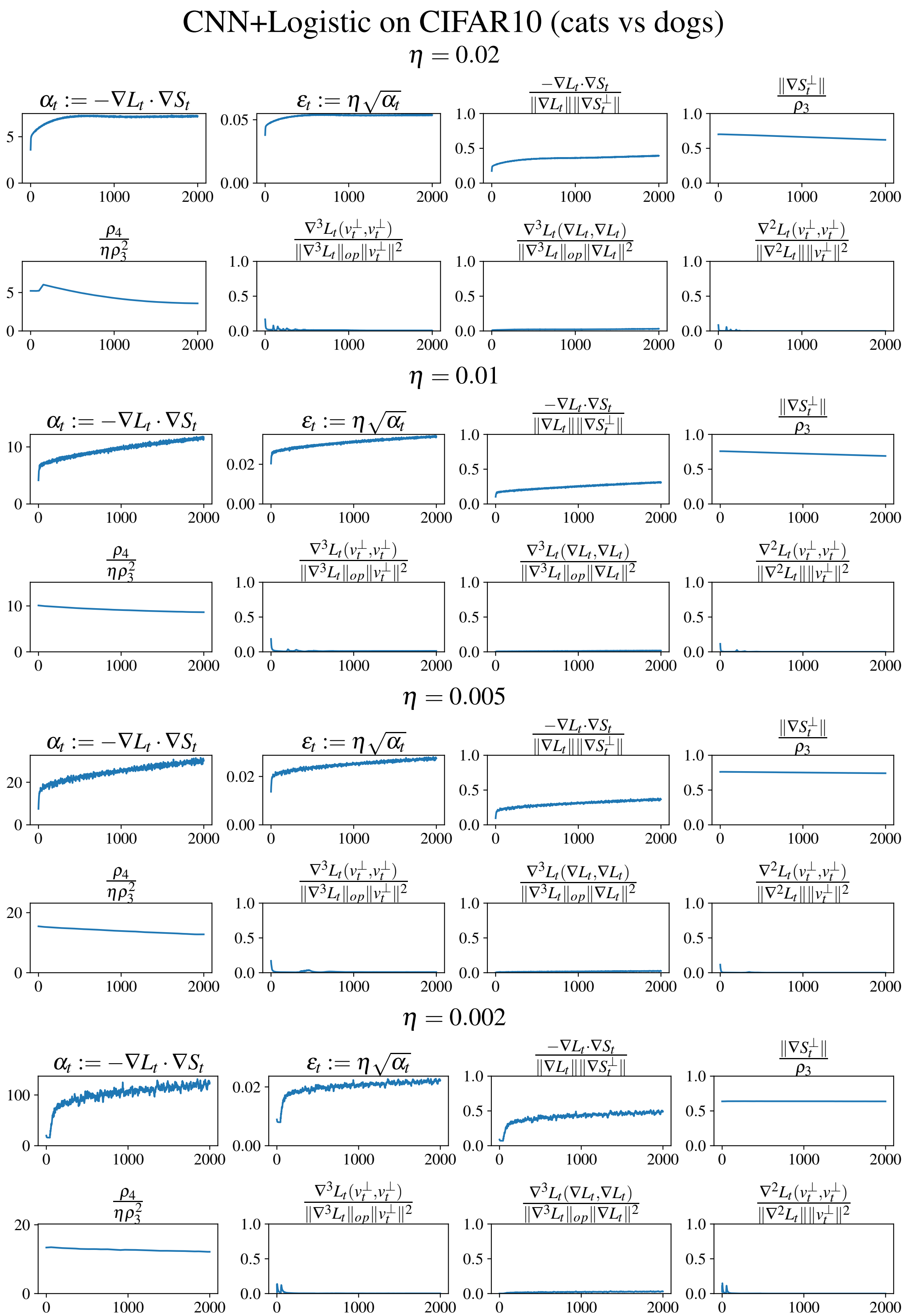
Our experiments were conducted in JAX [5], using `https://github.com/locuslab/edge-of-stability` as a reference for replicating the experimental setup used in [7]. All experiments were conducted on two servers, each with 10 NVIDIA GPUs. Code is provided in the supplementary material and will be made public on GitHub upon acceptance.

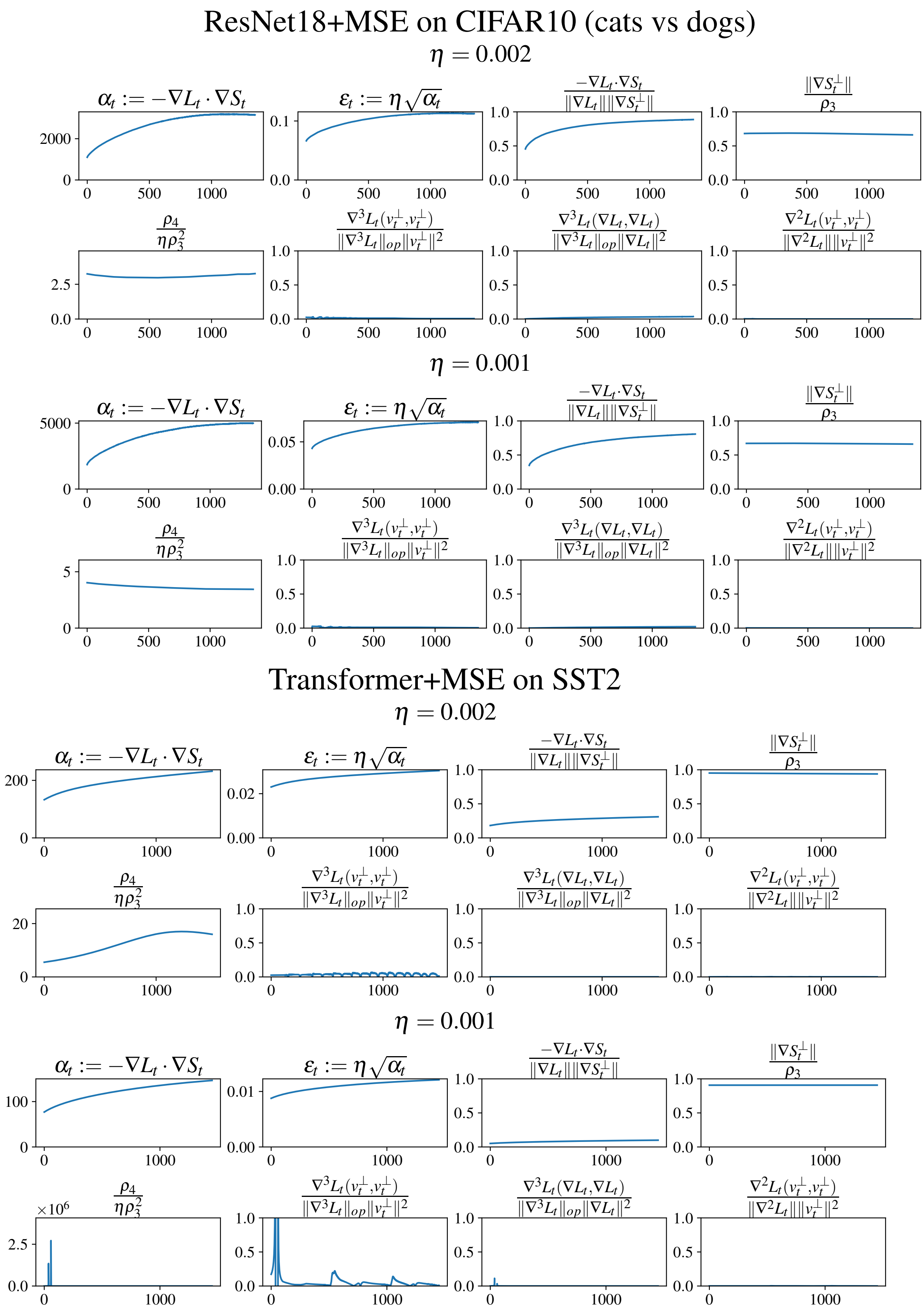
Appendix G. Empirical Verification of the Assumptions

For each of the experimental settings considered (MLP+MSE, CNN+MSE, CNN+Logistic, ResNet18+MSE, Transformer+MSE, Transformer+Logistic), we plot a number of quantities along the constrained trajectory to verify that the assumptions made in the main text hold. For each learning rate η we have 8 plots tracking various quantities, which verify the assumptions as follows: Assumption 1 is verified by the 1st plot, ϵ being small is verified by the 2nd plot, Assumption 4 is verified by the 3rd and 4th plots, Assumption 3 is verified by the 5th plot, and Assumption 5 is verified by the last 3 plots. As described in the experimental setup, training is stopped once the second eigenvalue is $1.9/\eta$, so Assumption 2 always holds with $c = 1.9$ as well.

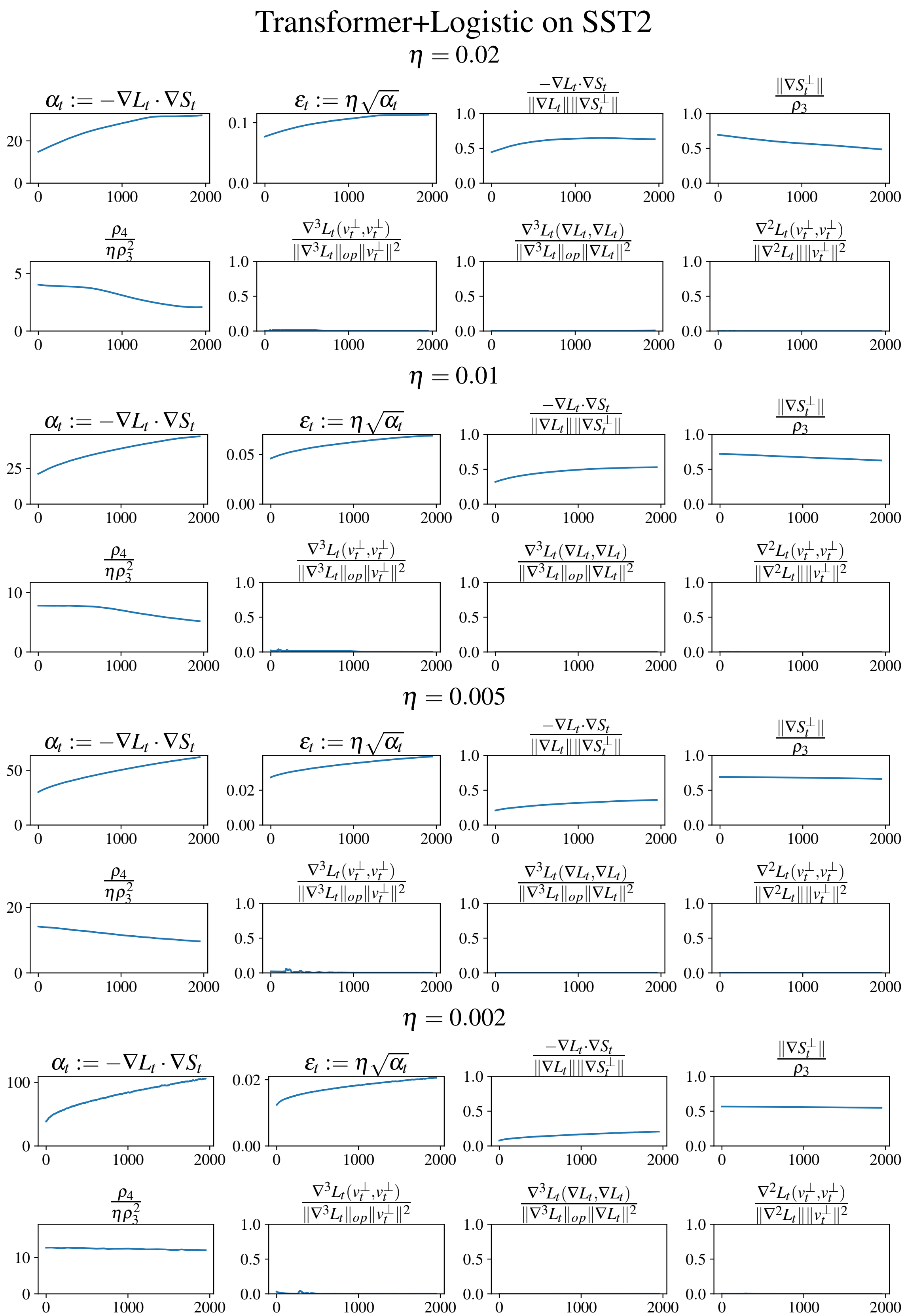

MLP+MSE on CIFAR10
$\eta = 0.02$
$\alpha_t := -\nabla L_t \cdot \nabla S_t$
$\epsilon_t := \eta\sqrt{\alpha_t}$
$\frac{-\nabla L_t \cdot \nabla S_t}{\|\nabla L_t\| \|\nabla S_t^\perp\|}$
$\frac{\|\nabla S_t^\perp\|}{\rho_3}$
$\frac{\rho_4}{\eta \rho_3^2}$
$\frac{\nabla^3 L_t(v_t^\perp, v_t^\perp)}{\|\nabla^3 L_t\|_{op} \|v_t^\perp\|^2}$
$\frac{\nabla^3 L_t(\nabla L_t, \nabla L_t)}{\|\nabla^3 L_t\|_{op} \|\nabla L_t\|^2}$
$\frac{\nabla^2 L_t(v_t^\perp, v_t^\perp)}{\|\nabla^2 L_t\| \|v_t^\perp\|^2}$
$\eta = 0.01$
$\eta = 0.005$
$\eta = 0.002$


CNN+MSE on CIFAR10
$\eta = 0.02$
$\alpha_t := -\nabla L_t \cdot \nabla S_t$
$\epsilon_t := \eta\sqrt{\alpha_t}$
$\frac{-\nabla L_t \cdot \nabla S_t}{\|\nabla L_t\|\|\nabla S_t^\perp\|}$
$\frac{\|\nabla S_t^\perp\|}{\rho_3}$
$\frac{\rho_4}{\eta\rho_3^2}$
$\frac{\nabla^3 L_t(v_t^\perp, v_t^\perp)}{\|\nabla^3 L_t\|_{op}\|v_t^\perp\|^2}$
$\frac{\nabla^3 L_t(\nabla L_t, \nabla L_t)}{\|\nabla^3 L_t\|_{op}\|\nabla L_t\|^2}$
$\frac{\nabla^2 L_t(v_t^\perp, v_t^\perp)}{\|\nabla^2 L_t\|\|v_t^\perp\|^2}$
$\eta = 0.005$
$\eta = 0.002$


CNN+Logistic on CIFAR10 (cats vs dogs)
$\eta = 0.02$
$\eta = 0.01$
$\eta = 0.005$
$\eta = 0.002$
$\alpha_t := -\nabla L_t \cdot \nabla S_t$
$\epsilon_t := \eta\sqrt{\alpha_t}$
$\frac{-\nabla L_t \cdot \nabla S_t}{\|\nabla L_t\| \|\nabla S_t^\perp\|}$
$\frac{\|\nabla S_t^\perp\|}{\rho_3}$
$\frac{\rho_4}{\eta \rho_3^2}$
$\frac{\nabla^3 L_t(v_t^\perp, v_t^\perp)}{\|\nabla^3 L_t\|_{op} \|v_t^\perp\|^2}$
$\frac{\nabla^3 L_t(\nabla L_t, \nabla L_t)}{\|\nabla^3 L_t\|_{op} \|\nabla L_t\|^2}$
$\frac{\nabla^2 L_t(v_t^\perp, v_t^\perp)}{\|\nabla^2 L_t\| \|v_t^\perp\|^2}$



ResNet18+MSE on CIFAR10 (cats vs dogs)
$\eta = 0.002$
$\alpha_t := -\nabla L_t \cdot \nabla S_t$
$\epsilon_t := \eta\sqrt{\alpha_t}$
$\frac{-\nabla L_t \cdot \nabla S_t}{\|\nabla L_t\|\|\nabla S_t^\perp\|}$
$\frac{\|\nabla S_t^\perp\|}{\rho_3}$
$\frac{\rho_4}{\eta\rho_3^2}$
$\frac{\nabla^3 L_t(v_t^\perp, v_t^\perp)}{\|\nabla^3 L_t\|_{op}\|v_t^\perp\|^2}$
$\frac{\nabla^3 L_t(\nabla L_t, \nabla L_t)}{\|\nabla^3 L_t\|_{op}\|\nabla L_t\|^2}$
$\frac{\nabla^2 L_t(v_t^\perp, v_t^\perp)}{\|\nabla^2 L_t\|\|v_t^\perp\|^2}$
$\eta = 0.001$
Transformer+MSE on SST2
$\eta = 0.002$
$\eta = 0.001$
$\times 10^6$


Transformer+Logistic on SST2
η = 0.02
η = 0.01
η = 0.005
η = 0.002

Appendix H. The Generalized Predicted Dynamics

Our analysis relies on a cubic Taylor expansion of the gradient. However, in order for this Taylor expansion to accurately track the gradients we need a bound on the fourth derivative of the loss (Assumption 3). Section 4 and Appendix G show that this approximation is sufficient to capture the dynamics of gradient descent at the edge of stability for many standard models when the loss criterion is the mean squared error. However, for certain architectures and loss functions, including ResNet18 and models trained with the logistic loss, this condition is often violated.

In these situations, the loss function in the top eigenvector direction is either *sub-quadratic*, meaning that the quadratic Taylor expansion overestimates the loss and sharpness[3], or *super-quadratic*, meaning that the quadratic Taylor expansion underestimates the loss and sharpness. To capture this phenomenon, we derive a more general form of the predicted dynamics which reduces to the standard predicted dynamics in Appendix D when the loss in the top eigenvector direction is approximately quadratic. In addition, Appendix I shows that the generalized predicted dynamics capture the dynamics of gradient descent at the edge of stability for both mean squared error and cross-entropy in all settings we tested.

H.1. Deriving the Generalized Predicted Dynamics

To derive the generalized predicted dynamics, we will abstract away the dynamics in the top eigenvector direction. Specifically, for every t we define

$$F_t(x) := L(\theta_t^\dagger + x u_t) - L(\theta_t^\dagger) - \frac{x^2}{\eta}.$$

We say that L is sub-quadratic at t if $F_t(x) < 0$ and super-quadratic if $F_t(x) > 0$.

Note that knowing F_t is not sufficient to capture the dynamics in the u_t direction. Specifically,

$$x_{t+1} = x_t - \eta u_t \cdot \nabla L(\theta_t^\dagger + v_t) \neq x_t - \eta u_t \cdot \nabla L(\theta_t^\dagger + x u_t).$$

It is still critically important to track the effect that the movement in the $\nabla S_t^\perp$ direction has on the dynamics of x. As in Section 3.1, the effect of the movement in the $\nabla S_t^\perp$ direction on the dynamics of x is changing the sharpness by y_t. This gives us the generalized predicted dynamics update:

$$v_{t+1}^* = P_{u_{t+1}}^\perp (I - \eta \nabla^2 L_t) P_{u_t}^\perp v_t^* + \eta P_{u_{t+1}}^\perp \nabla S_t^\perp \left[\frac{\delta_t^2 - x_t^{*2}}{2}\right] - x_{t+1}^\star \cdot u_{t+1}$$

where $x_{t+1}^\star = -(1 + \eta y_t^\star) x_t^\star - \eta F'(x_t^\star)$.

Note that when $F_t(x) = 0$ is exactly quadratic, this reduces to the standard predicted dynamics update in (5). Note that the update for y is completely unchanged:

Lemma 14 *Restricted to the $u_t, \nabla S_t$ directions, the generalized predicted dynamics $v_t^\star$ imply:*

$$x_{t+1}^\star = -(1 + \eta y_t^\star) x_t^\star - \eta F'(x_t^\star) \quad \textit{and} \quad y_{t+1}^\star = \eta \sum_{s=0}^{t} \beta_{s \to t} \left[\frac{\delta_s^2 - x_s^{*2}}{2}\right]. \tag{10}$$

The proof is identical to the proof of Theorem 9.

3. This sub-quadratic phenomenon was also observed in [23].

H.2. Properties of the Generalized Predicted Dynamics

Note that due to the sign flipping argument in Appendix K, we can assume that F is an even function as the odd part will only influence the dynamics through additional oscillations of period 2, so throughout the remainder of this section we will assume that $F_t(x) = F_t(-x)$. Otherwise, we can simply redefine F by its even part.

Next, note that the fixed point of (10) is still when $x_t = \delta_t$, regardless of the shape of F_t, due to the need to stabilize the $\nabla S_t^\perp$ direction. This contradicts previous 1-dimensional analyses of edge of stability in which the fixed point in the top eigenvector direction strongly depends on the shape of F_t, the loss in the u_t direction.

The limiting value of y_t can therefore be read from the update for x_t. If (δ_t, y) is an orbit of period 2 of (10), then

$$-\delta_t = -(1+\eta y)\delta_t - \eta F'(\delta_t) \implies y = -\frac{F'(\delta_t)}{\delta_t}.$$

In addition, note that the sharpness can no longer be approximated as $S(\theta_t) \approx 2/\eta + y_t$ as the sharpness now changes along the u_t direction. In particular, it changes by $F''(x)$ so that

$$S(\theta_t) \approx 2/\eta + y_t + F''(x_t).$$

Therefore, the limiting sharpness of (10) is

$$S(\theta_t) \to 2/\eta - \frac{F_t'(\delta_t)}{\delta_t} + F_t''(\delta_t).$$

When $F_t = 0$ and the loss is exactly quadratic in the u direction, this update reduces to fixed point predictions in Section 3.1.

One interesting phenomenon observed by Cohen et al. [7] is that with cross-entropy loss, the sharpness was never exactly $2/\eta$, but usually hovered above it. This contradicts the predictions of the standard predicted dynamics which predict that the fixed point has sharpness 0. However, using the generalized predicted dynamics (10), we can give a clear explanation.

When the loss is sub-quadratic, e.g. when $F_t(x) = -\rho_4 \frac{x^4}{24}$, we have

$$S(\theta_t) \to 2/\eta + \rho_4 \frac{\delta_t^2}{6} - \rho_4 \frac{\delta_t^2}{2} = 2/\eta - \rho_4 \frac{\delta_t^2}{3} < 2/\eta$$

so the sharpness will converge to a value *below* $2/\eta$. On the other hand if the loss is super-quadratic, the sharpness converges to a value *above* $2/\eta$. More generally, whether the loss converges to a value above or below $2/\eta$ depends on the sign of $F_t''(\delta_t) - \delta_t F_t'(\delta_t)$.

In our experiments in Appendix I, we observed both sub-quadratic and super-quadratic loss functions. In particular, the loss was usually sub-quadratic when it first reached instability but gradually became super-quadratic as training progressed at the edge of stability.

Appendix I. Additional Experiments

I.1. The Benefit of Large Learning Rates: Training Time and Generalization

We trained ResNet18 with full batch gradient descent on the full 50k training set of CIFAR10 with various learning rates, in addition to the commonly proposed learning rate schedule $\eta_t := 1/S(\theta_t)$. We show that despite entering the edge of stability, large learning rates converge much faster. In addition, due to the self-stabilization effect of gradient descent, the final sharpness is bounded by $2/\eta$ which is smaller for larger learning rates and leads to better generalization (see Figure 3).

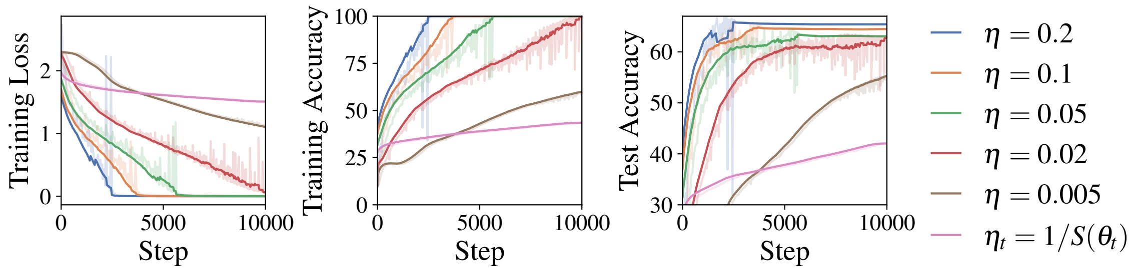
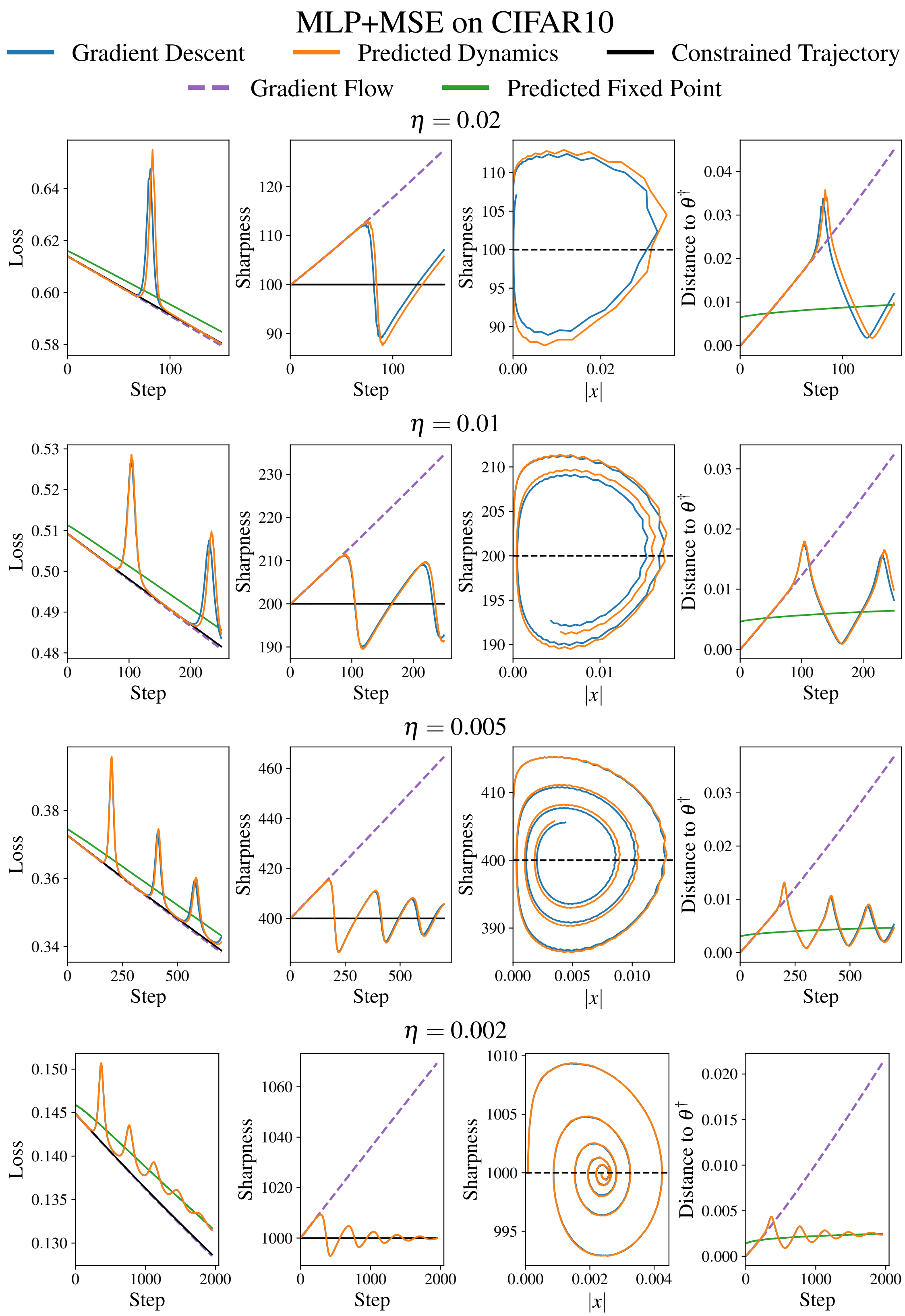
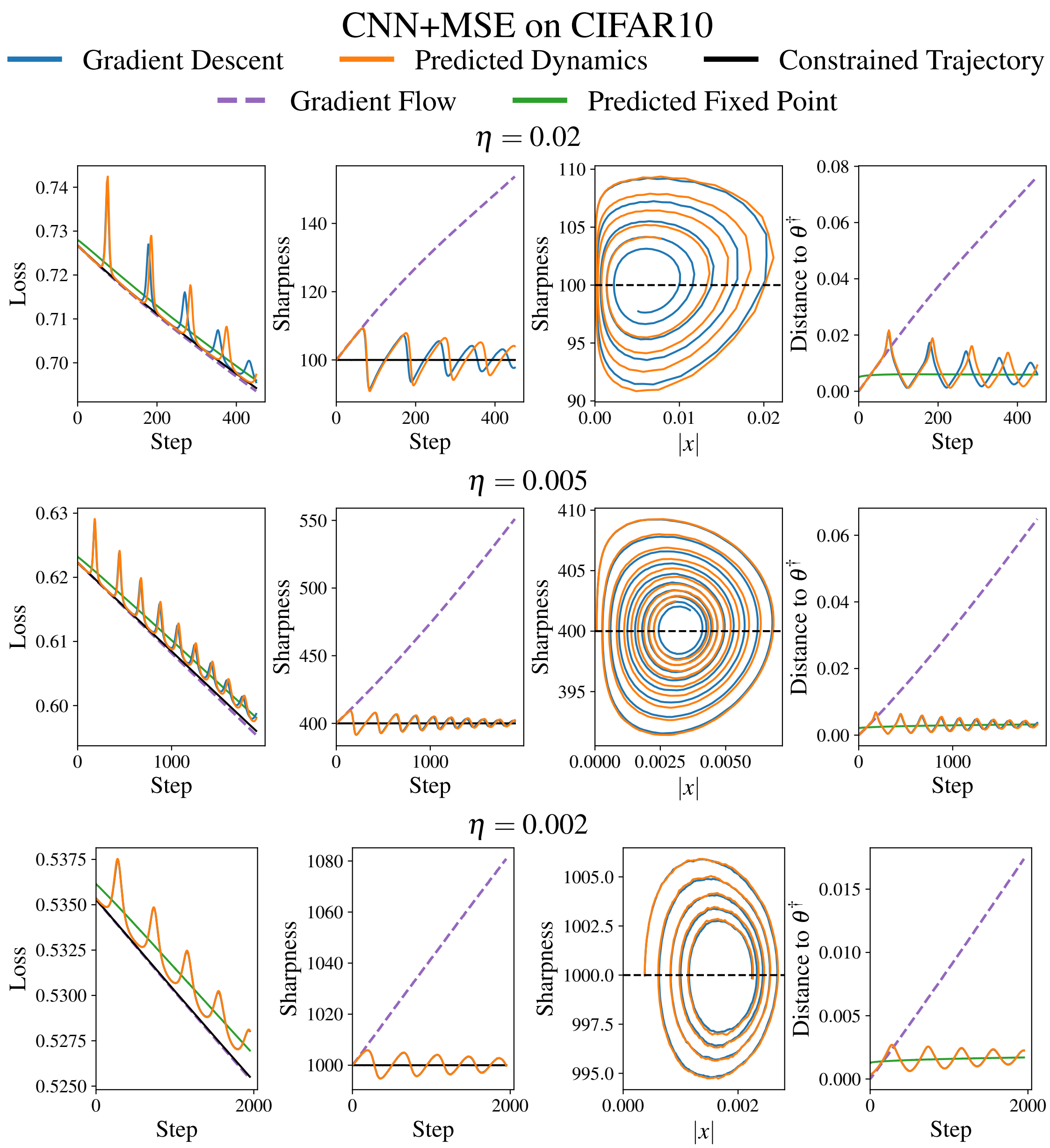
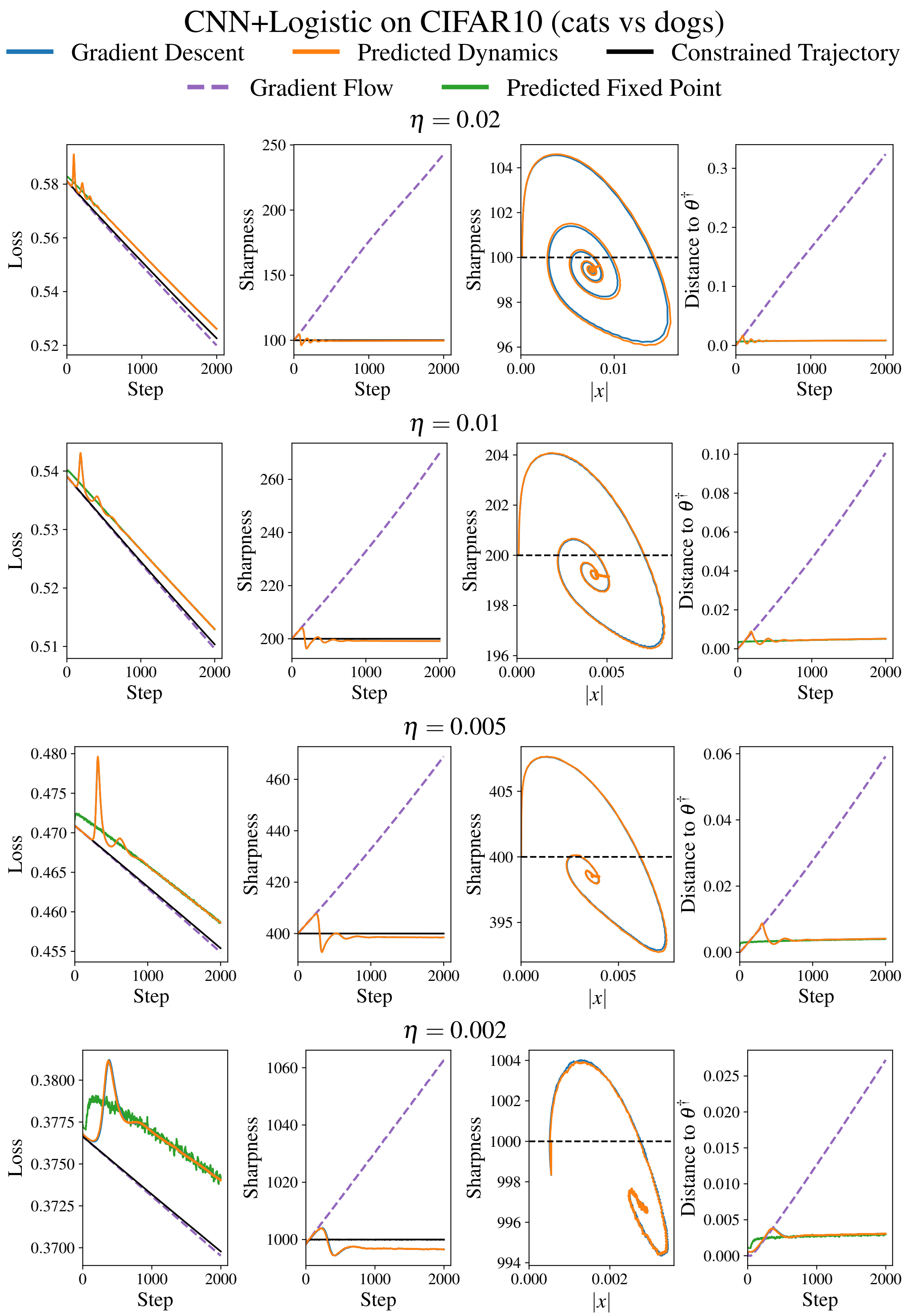


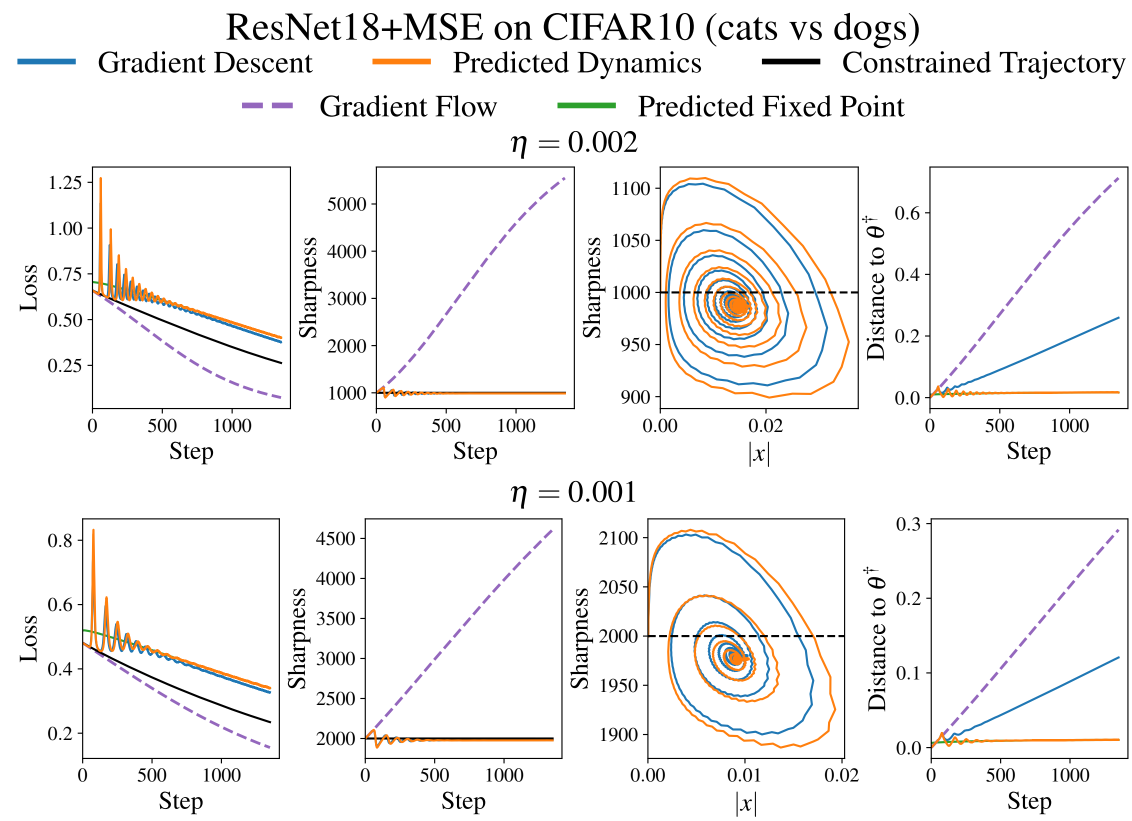
Figure 3: Large learning rates converge faster and generalize better (ResNet18 and CIFAR10).

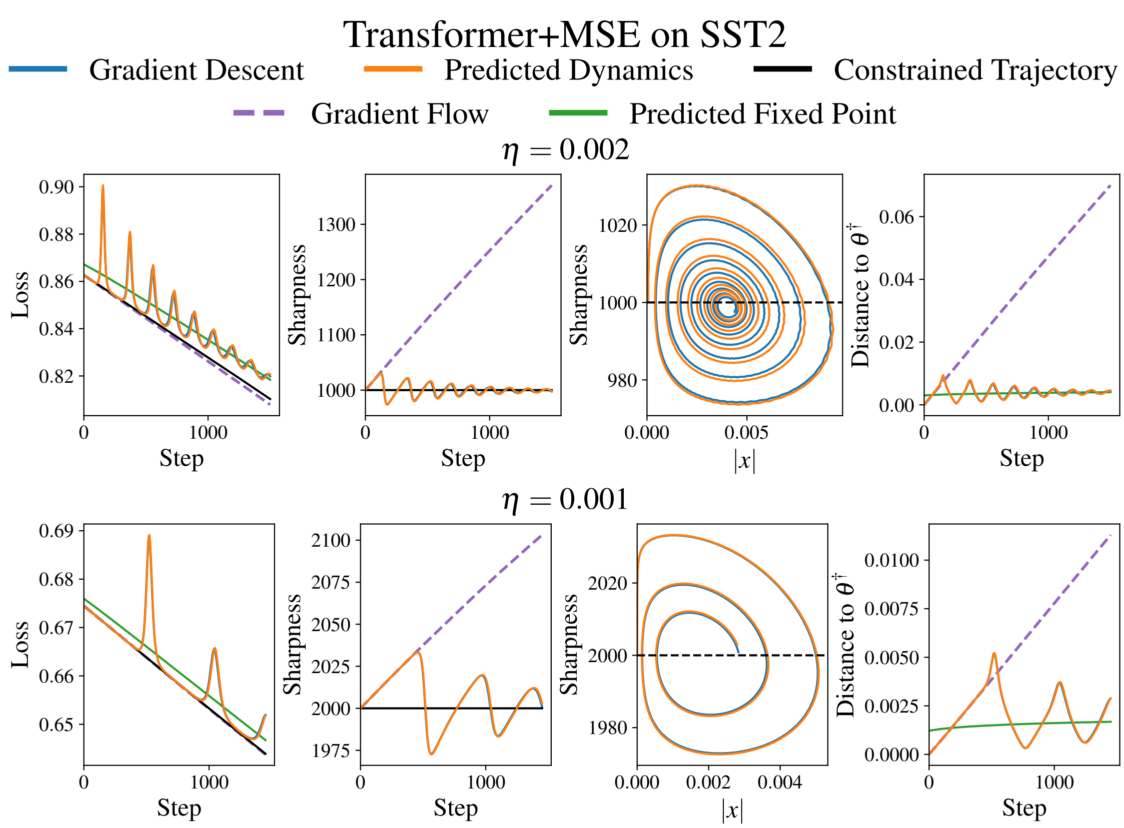
I.2. Experiments with the Generalized Predicted Dynamics

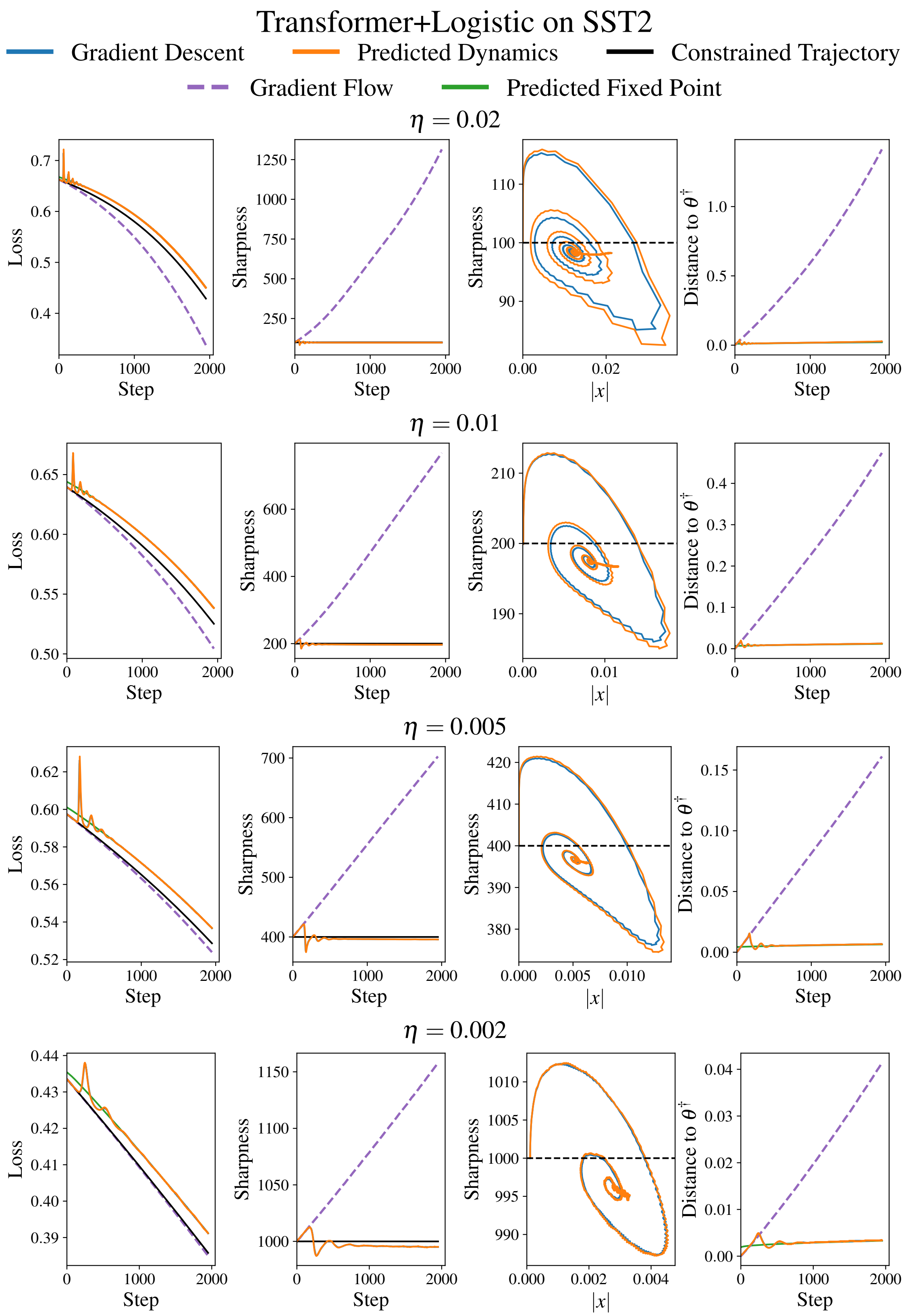



CNN+MSE on CIFAR10
Gradient Descent
Predicted Dynamics
Constrained Trajectory
Gradient Flow
Predicted Fixed Point
$\eta = 0.02$
$\eta = 0.005$
$\eta = 0.002$
Loss
Sharpness
Distance to $\theta^\dagger$
Step
$|x|$


CNN+Logistic on CIFAR10 (cats vs dogs)
Gradient Descent
Predicted Dynamics
Constrained Trajectory
Gradient Flow
Predicted Fixed Point
η = 0.02
η = 0.01
η = 0.005
η = 0.002
Loss
Sharpness
Distance to θ†
Step
|x|


ResNet18+MSE on CIFAR10 (cats vs dogs)
Gradient Descent
Predicted Dynamics
Constrained Trajectory
Gradient Flow
Predicted Fixed Point
$\eta = 0.002$
$\eta = 0.001$
Loss
Sharpness
Distance to $\theta^\dagger$
Step
$|x|$


Transformer+MSE on SST2
Gradient Descent
Predicted Dynamics
Constrained Trajectory
Gradient Flow
Predicted Fixed Point
$\eta = 0.002$
$\eta = 0.001$
Loss
Sharpness
Distance to $\theta^\dagger$
Step
$|x|$


Transformer+Logistic on SST2
Gradient Descent
Predicted Dynamics
Constrained Trajectory
Gradient Flow
Predicted Fixed Point
$\eta = 0.02$
$\eta = 0.01$
$\eta = 0.005$
$\eta = 0.002$
Loss
Sharpness
Distance to $\theta^\dagger$
Step
$|x|$

Appendix J. Additional Discussion

Multiple Unstable Eigenvalues Our work focuses on explaining edge of stability in the presence of a single unstable eigenvalue (Assumption 2). However, Cohen et al. [7] observed that progressive sharpening appears to apply to *all* eigenvalues, even after the largest eigenvalue has become unstable. As a result, all of the top eigenvalues will successively enter edge of stability (see Figure 4). In particular, Figure 4 shows that the dynamics are fairly well behaved in the period when only a single eigenvalue is unstable, yet appear to be significantly more chaotic when multiple eigenvalues are unstable.

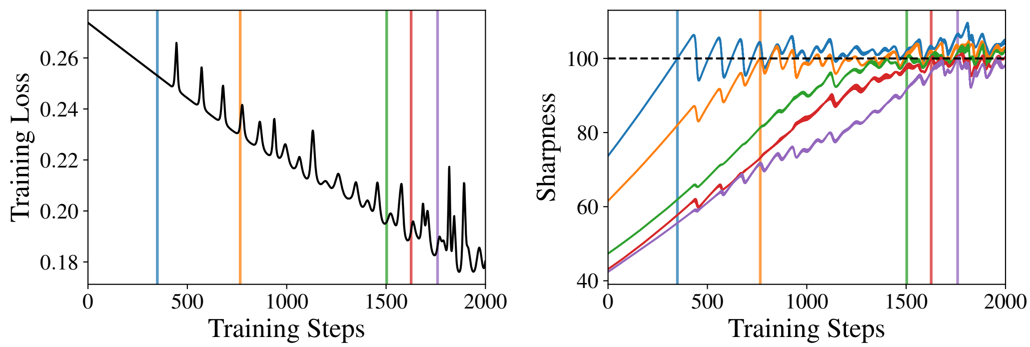


Figure 4: Edge of stability with multiple unstable eigenvalues. Each vertical line is the time at which the corresponding eigenvalue of the same color becomes unstable.

One technical challenge with dealing with multiple eigenvalues is that, when the top eigenvalue is not unique, the sharpness is no longer differentiable and it is unclear how to generalize our analysis. However, one might expect that gradient descent can still be coupled to projected gradient descent under the non-differentiable constraint $S(\theta_T^\dagger) \le 2/\eta$. When there are k unstable eigenvalues, with corresponding eigenvectors $u_t^1, \dots, u_t^k$, the constrained update is roughly equivalent to projecting out the subspace $\operatorname{span}\{\nabla^3 L_t(u_t^i, u_t^j) : i, j \in [k]\}$ from the gradient update $-\eta \nabla L_t$. Demonstrating self-stabilization thus requires analyzing the dynamics in the subspace $\operatorname{span}\Big(\{u_t^i : i \in [k]\} \cup \{\nabla^3 L_t(u_t^i, u_t^j) : i, j \in [k]\}\Big)$. We leave investigating the dynamics of multiple unstable eigenvalues for future work.

Connection to Sharpness Aware Minimization (SAM) Foret et al. [10] introduced the sharpness-aware minimization (SAM) algorithm, which aims to control sharpness by solving the optimization problem $\min_\theta \max_{\|\delta\| \le \epsilon} L(\theta + \delta)$. This is roughly equivalent to minimizing $S(\theta)$ over all global minimizers, and thus SAM tries to explicitly minimize the sharpness. Our analysis shows that gradient descent *implicitly* minimizes the sharpness, and for a fixed η looks to minimize $L(\theta)$ subject to $S(\theta) = 2/\eta$.

Connections to Warmup. Gilmer et al. [11] demonstrated that *learning rate warmup*, which consists of gradually increasing the learning rate, empirically leads to being able to train with a larger learning rate. The self-stabilization property of gradient descent provides a plausible explanation for this phenomenon. If too large of an initial learning rate η_0 is chosen (so that $S(\theta_0)$ is much greater than $2/\eta_0$), then the iterates may diverge before self stabilization can decrease the sharpness to $2/\eta_0$. On the other hand, if the learning rate is chosen that $S(\theta_0)$ is only slightly greater than $2/\eta_0$, self-stabilization will decrease the sharpness to $2/\eta_0$. Repeatedly increasing the learning rate slightly could then lead to small decreases in sharpness without the iterates diverging, thus allowing training to proceed with a large learning rate.

Connection to Weight Decay and Sharpness Reduction. Lyu et al. [22] proved that when the loss function is scale-invariant, gradient descent with weight decay and sufficiently small learning rate converges leads to reduction of the *normalized* sharpness $S(\theta/\|\theta\|)$. In fact, the mechanism behind the sharpness reduction is exactly the self-stabilization force described in this paper restricted to the setting in [22]. We present here a heuristic derivation of this equivalence.

Our primary result is that gradient descent solves the constrained problem $\min_\theta L(\theta)$ such that $S(\theta) \leq 2/\eta$. To prove equivalence to the sharpness reduction, we will need the following lemma from [22] which follows from the scale invariance of the loss:

$$S(\theta) = \frac{1}{\|\theta\|^2} S(\theta/\|\theta\|).$$

Let $L_\lambda(\theta) := L(\theta) + \frac{\lambda}{2}\|\theta\|^2$ and $S_\lambda(\theta) = S(\theta) + \lambda$ denote the regularized loss and sharpness respectively and let $\overline{\theta} := \frac{\theta}{\|\theta\|}$. Then we have the following equality between minimization problems:

$$\begin{aligned}
&\min_\theta L_\lambda(\theta) \quad \text{such that} \quad S_\lambda(\theta) \leq 2/\eta \\
&\iff \min_\theta L(\theta) + \lambda \frac{\|\theta\|^2}{2} \quad \text{such that} \quad S(\theta) \leq 2/\eta - \lambda \\
&\iff \min_{\overline{\theta}, \|\theta\|} L(\overline{\theta}) + \lambda \frac{\|\theta\|^2}{2} \quad \text{such that} \quad \frac{1}{\|\theta\|^2} S(\overline{\theta}) \leq \frac{2 - \eta\lambda}{\eta} \\
&\iff \min_{\overline{\theta}} L(\overline{\theta}) + \frac{\eta\lambda}{2 - \eta\lambda} S(\overline{\theta})
\end{aligned}$$

where the last line follows from the scale-invariance of the loss function. In particular if $\eta\lambda$ is sufficiently small and the dynamics are initialized near a global minimizer of the loss, this will converge to the solution of the constrained problem:

$$\min_{\|\overline{\theta}\|=1} S(\overline{\theta}) \quad \text{such that} \quad L(\overline{\theta}) = 0.$$

Appendix K. Proofs

K.1. Properties of the Constrained Trajectory

We next prove several nice properties of the constrained trajectory. Before, we require the following auxiliary lemma, which shows that several quantities are Lipschitz in a neighborhood around the constrained trajectory:

Lemma 15 (Lipschitz Properties)

1. *$\theta \to \nabla L(\theta)$ is $O(\eta^{-1})$-Lipschitz in each set $\mathcal{S}_t$.*
2. *$\theta \to \nabla^2 L(\theta)$ is ρ_3-Lipschitz with respect to $\|\cdot\|_2$.*
3. *$\theta \to \lambda_i(\nabla^2 L(\theta))$ is ρ_3-Lipschitz.*

4. *$\theta \to u(\theta)$ is $O(\eta\rho_3)$-Lipschitz in each set $\mathcal{S}_t$.*

5. *$\theta \to \nabla S(\theta)$ is $O(\eta\rho_3^2)$-Lipschitz in each set $\mathcal{S}_t$.*

Proof The Lipschitzness of $\nabla^2 L(\theta)$ follows immediately from the bound $\left\|\nabla^3 L(\theta)\right\|_{op} \le \rho_3$. Weil's inequality then immediately implies the desired bound on the Lipschitz constant of the eigenvalues of $\nabla^2 L(\theta)$. Therefore for any t, we have for all $\theta \in \mathcal{S}_t$:

$$\lambda_1(\nabla^2 L(\theta)) - \lambda_2(\nabla^2 L(\theta)) \ge \lambda_1(\nabla^2 L(\theta)) - \lambda_2(\nabla^2 L(\theta)) - 2\rho_3 \frac{2-c}{4\eta\rho_3} \ge \frac{2-c}{2\eta}.$$

Next, from the derivative of eigenvector formula:

$$\begin{aligned}\|\nabla u(\theta)\|_2 &= \left\|(\lambda_1(\nabla^2 L(\theta))I - \nabla^2 L(\theta))^\dagger \nabla^3 L(\theta)(u(\theta))\right\|_2 \\ &\le \frac{\rho_3}{\lambda_1(\nabla^2 L(\theta)) - \lambda_2(\nabla^2 L(\theta))} \\ &\le \frac{2\eta\rho_3}{2-c} \\ &= O(\eta\rho_3)\end{aligned}$$

which implies the bound on the Lipschitz constant of u restricted to $\mathcal{S}_t$. Finally, because $\nabla S(\theta) = \nabla^3 L(\theta)(u(\theta), u(\theta))$,

$$\left\|\nabla^2 S(\theta)\right\|_2 \le \left\|\nabla^4 L(\theta)\right\|_{op} + 2\left\|\nabla^3 L(\theta)\right\|_{op}\|\nabla u(\theta)\|_2 \le O(\rho_4 + \eta\rho_3^2) \le O(\eta\rho_3^2)$$

where the second to last inequality follows from the bound on $\|\nabla u(\theta)\|_2$ restricted to $\mathcal{S}_t$ and the last inequality follows from Assumption 3. ■

Lemma 16 (First-order approximation of the constrained trajectory update $\{\theta_t^\dagger\}$) *For all $t \le \mathscr{T}$,*

$$\theta_{t+1}^\dagger = \theta_t^\dagger - \eta P_{u_t,\nabla S_t}^\perp \nabla L_t + O\big(\epsilon^2 \cdot \eta\|\nabla L_t\|\big) \quad \text{and} \quad S_t = 2/\eta.$$

Proof We will prove by induction that $S_t = 2/\eta$ for all t. The base case follows from the definitions of $\theta_0, \theta_0^\dagger$. Next, assume $S(\theta_t^\dagger) = 0$ for some $t \ge 0$. Let $\theta' = \theta_t^\dagger - \eta\nabla L_t$. Then because $\theta_t^\dagger \in \mathscr{M}$ we have $\left\|\theta_{t+1}^\dagger - \theta'\right\| \le \left\|\theta_t^\dagger - \theta'\right\| = \eta\|\nabla L_t\|$. Then because $\theta_{t+1}^\dagger = \mathrm{proj}_{\mathcal{M}}(\theta')$, the KKT conditions for this minimization problem imply that there exist x, y with $y \ge 0$ such that

$$\begin{aligned}\theta_{t+1}^\dagger &= \theta_t^\dagger - \eta\nabla L_t - x\nabla_\theta[\nabla L(\theta)\cdot u(\theta)]\Big|_{\theta=\theta_{t+1}^\dagger} - y\nabla S_{t+1} \\ &= \theta_t^\dagger - \eta\nabla L_t - x\big[S_{t+1}u_{t+1} + \nabla u_{t+1}^T \nabla L_{t+1}\big] - y\nabla S_{t+1} \\ &= \theta_t^\dagger - \eta\nabla L_t - x[S_{t+1}u_{t+1} + O(\eta\rho_3\|\nabla L_{t+1}\|)] - y\nabla S_{t+1} \\ &= \theta_t^\dagger - \eta\nabla L_t - x[S_t u_t + O(\eta\rho_3\|\nabla L_t\|)] - y\big[\nabla S_t + O(\eta^2\rho_3^2\|\nabla L_t\|)\big] \\ &= \theta_t^\dagger - \eta\nabla L_t - xS_t u_t - y\nabla S_t + O\big((|x|\eta\rho_3 + |y|\eta^2\rho_3^2)\|\nabla L_t\|\big).\end{aligned}$$

Next, note that we can decompose $\nabla S_t = u_t(\nabla S_t \cdot u_t) + \nabla S_t^\perp$:

$$\theta_{t+1}^\dagger = \theta_t^\dagger - \eta\nabla L_t - [xS_t + y(\nabla S_t \cdot u_t)]u_t - y\nabla S_t^\perp + O\big((|x|\eta\rho_3 + |y|\eta^2\rho_3^2)\|\nabla L_t\|\big).$$

Let $s_t = \frac{\nabla S_t^\perp}{\|\nabla S_t^\perp\|}$. We can now perform the change of variables

$$(x', y') = \left(xS_t + y(\nabla S_t \cdot u_t), y\left\|\nabla S_t^\perp\right\|\right), \quad (x, y) = \left(\frac{x' - y' \frac{\nabla S_t \cdot u_t}{\|\nabla S_t^\perp\|}}{S_t}, \frac{y'}{\left\|\nabla S_t^\perp\right\|}\right)$$

to get

$$\theta_{t+1}^\dagger = \theta_t^\dagger - \eta\nabla L_t - x'u_t - y's_t + O\big(\eta^2\rho_3\|\nabla L\|(\left|x'\right| + \left|y'\right|)\big).$$

Note that

$$O(\eta^2\rho_3\|\nabla L\|(|x| + |y|)) \leq \frac{\sqrt{x^2 + y^2}}{2} \tag{11}$$

for sufficiently small ϵ so because $\left\|\theta_{t+1}^\dagger - \theta'\right\| \leq \eta\|\nabla L_t\|$ we have

$$\frac{\sqrt{x^2 + y^2}}{2} \leq \left\|\theta_{t+1}^\dagger - \theta'\right\| \leq \eta\|\nabla L_t\|$$

so $x, y = O(\eta\|\nabla L_t\|)$. Therefore,

$$\begin{aligned}\theta_{t+1}^\dagger &= \theta_t^\dagger - \eta\nabla L_t - x'u_t - y's_t + O\Big(\eta^3\rho_3\|\nabla L\|^2\Big) \\ &= \theta_t^\dagger - \eta\nabla L_t - x'u_t - y's_t + O\big(\epsilon^2 \cdot \eta\|\nabla L_t\|\big)\end{aligned}$$

Then Taylor expanding ∇L_{t+1} around $\theta_t^\dagger$ gives

$$\begin{aligned}\nabla L_{t+1} \cdot u_{t+1} &= \nabla L_t \cdot u_t + (\nabla L_{t+1} - \nabla L_t) \cdot u_t + \nabla L_{t+1} \cdot (u_{t+1} - u_t) \\ &= u_t^T \nabla^2 L_t\big[-\eta\nabla L_t - x'u_t - y's_t + O(\epsilon^2 \cdot \eta\|\nabla L_t\|\big] + O\big(\epsilon^2 \cdot \|\nabla L_t\|\big) \\ &= -x'S_t + O\big(\epsilon^2 \cdot \|\nabla L_t\|\big)\end{aligned}$$

so $x' = O(\epsilon^2 \cdot \eta\|\nabla L_t\|)$. We can also Taylor expand S_{t+1} around $\theta_t^\dagger$ and use that $S_t = 2/\eta$ to get

$$\begin{aligned}S_{t+1} &= 2/\eta + \nabla S_t \cdot \Big[-\eta\nabla L_t - x'u_t - y's_t + O\Big(\eta^3\rho_3\|\nabla L_t\|^2\Big)\Big] + O\big(\epsilon^2 \cdot \rho_3\eta\|\nabla L_t\|\big) \\ &= 2/\eta + \eta\alpha_t - y'\|\nabla S_t^\perp\| + O\big(\epsilon^2 \cdot \rho_3\eta\|\nabla L_t\|\big).\end{aligned}$$

Now note that for ϵ sufficiently small we have

$$O\big(\epsilon^2 \cdot \rho_3\eta\|\nabla L_t\|\big) \leq O\big(\epsilon^2 \cdot \eta\alpha_t\big) \leq \eta\alpha_t.$$

Therefore if $y' = 0$, we would have $S_{t+1} > 2/\eta$ which contradicts $\theta_{t+1}^\dagger \in \mathcal{M}$. Therefore $y' > 0$ and therefore $y > 0$, which by complementary slackness implies $S_{t+1} = 2/\eta$. This then implies that

$$-\eta\nabla L_t \cdot \nabla S_t^\perp - y'\|\nabla S_t^\perp\| + O(\epsilon^2 \cdot \rho_3\eta\|\nabla L_t\|) = 0 \implies y' = -\eta\nabla L_t \cdot \frac{\nabla S_t^\perp}{\left\|\nabla S_t^\perp\right\|} + O\big(\epsilon^2 \cdot \eta\|\nabla L_t\|\big).$$

Putting it all together gives

$$\begin{aligned}\theta_{t+1}^\dagger &= \theta_t^\dagger - \eta P_{\nabla S_t^\perp}^\perp \nabla L_t + O\big(\epsilon^2 \cdot \eta\|\nabla L_t\|\big) \\ &= \theta_t^\dagger - \eta P_{u_t, \nabla S_t}^\perp \nabla L_t + O\big(\epsilon^2 \cdot \eta\|\nabla L_t\|\big)\end{aligned}$$

where the last line follows from $u_t \cdot \nabla L_t = 0$. ■

Lemma 17 (Descent Lemma for $\theta^\dagger$) *For all $t \le \mathscr{T}$,*

$$L(\theta^\dagger_{t+1}) \le L(\theta^\dagger_t) - \Omega\left(\eta\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2\right).$$

Proof Taylor expanding $L(\theta^\dagger_{t+1})$ around $L(\theta^\dagger_t)$ and using Lemma 16 gives

$$\begin{aligned}
L(\theta^\dagger_{t+1}) &= L(\theta^\dagger_t) + \nabla L_t \cdot (\theta^\dagger_{t+1} - \theta^\dagger_t) + \frac{1}{2}(\theta^\dagger_{t+1} - \theta^\dagger_t)^T \nabla^2 L_t(\theta^\dagger_{t+1} - \theta^\dagger_t) + O\left(\rho_3\left\|\theta^\dagger_{t+1} - \theta^\dagger_t\right\|^3\right) \\
&= L(\theta^\dagger_t) - \eta\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2 + \frac{\eta^2\lambda_2(\nabla^2 L_t)\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2}{2} + O\left(\eta^3\rho_3\|\nabla L_t\|^3\right) \\
&= L(\theta^\dagger_t) - \frac{\eta(2-c)}{2}\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2 + O\left(\eta^3\rho_3\|\nabla L_t\|^3\right).
\end{aligned}$$

Next, note that because $\gamma_t = \Theta(1)$ we have $\|\nabla L_t\| = O(\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|)$.Therefore for ϵ sufficiently small,

$$O\left(\eta^3\rho_3\|\nabla L_t\|^3\right) = O(\epsilon^2 \cdot \eta\|\nabla L_t\|^2) \le \frac{\eta(2-c)}{4}\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2.$$

Therefore,

$$L(\theta^\dagger_{t+1}) \le L(\theta^\dagger_t) - \frac{\eta(2-c)}{4}\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2 = L(\theta^\dagger_t) - \Omega(\eta\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2)$$

which completes the proof. ■

Corollary 18 *Let $L^\star = \min_\theta L(\theta)$. Then there exists $t \le \mathscr{T}$ such that*

$$\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2 \le O\left(\frac{L(\theta^\dagger_0) - L^\star}{\eta\mathscr{T}}\right).$$

Proof Inductively applying Lemma 17 we have that there exists an absolute constant c such that

$$L^\star \le L(\theta^\dagger_{\mathscr{T}}) \le L(\theta^\dagger_0) - c\eta\sum_{t<\mathscr{T}}\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2$$

which implies that

$$\min_{t<\mathscr{T}}\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2 \le \frac{\sum_{t<\mathscr{T}}\left\|P^\perp_{u_t,\nabla S_t}\nabla L_t\right\|^2}{\mathscr{T}} \le O\left(\frac{L(\theta^\dagger_0) - L^\star}{\eta\mathscr{T}}\right).$$

■

K.2. Proof of Theorem 11

We first require the following three lemmas, whose proofs are deferred to Appendix K.3.

Lemma 19 (2-Step Lemma) *Let*

$$r_t := v_{t+2} - \mathrm{step}_{t+1}(\mathrm{step}_t(v_t)).$$

Assume that $\|v_t\| \le \epsilon^{-1}\delta$. *Then*

$$\|r_t\| \le O\left(\epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right).$$

Lemma 20 *Assume that there exists constants* c_1, c_2 *such that for all* $t \le \mathscr{T}$, $\|\overset{\star}{v}_t\| \le c_2\delta$, $|\overset{\star}{x}_t| \ge c_1\delta$. *Then, for all* $t \le \mathscr{T}$, *we have*

$$\|v_t - \overset{\star}{v}_t\| \le O(\epsilon\delta)$$

Lemma 21 *For* $t \le \mathscr{T}$, $\|\overset{\star}{v}_t\| \le O(\delta)$.

With these lemmas in hand, we can prove Theorem 11.

Proof [Proof of Theorem 11]

First, by Lemma 21, we have $\|\overset{\star}{v}_t\| \le O(\delta)$.

Next, by Lemma 20, we have

$$\theta_t - \theta_t^\dagger = v_t = \overset{\star}{v}_t + O(\epsilon\delta).$$

Next, we Taylor expand to calculate $S(\theta_t)$:

$$\begin{aligned}
S(\theta_t) &= S(\theta_t^\dagger) + \nabla S_t \cdot v_t + O(\eta\rho_3^2\|v_t\|^2) \\
&= 2/\eta + \nabla S_t^\perp \cdot v_t + \nabla S_t \cdot u_t u_t \cdot v_t + O(\eta\rho_3^2\delta^2) \\
&= 2/\eta + \nabla S_t^\perp \cdot \overset{\star}{v}_t + \nabla S_t \cdot u_t u_t \cdot \overset{\star}{v}_t + O(\rho_3\epsilon\delta + \eta\rho_3^2\delta^2) \\
&= 2/\eta + y_t + (\nabla S_t \cdot u_t)x_t + O(\eta^{-1}\epsilon^2).
\end{aligned}$$

Finally, we Taylor expand the loss:

$$\begin{aligned}
L(\theta_t) &= L(\theta_t^\dagger) + \nabla L_t \cdot v_t + \frac{1}{2}v_t^T\nabla^2 L_t v_t + O(\rho_3\|v_t\|^3) \\
&= L(\theta_t^\dagger) + \frac{1}{\eta}x_t^2 + \frac{1}{2}{v_t^\perp}^T\nabla^2 L_t v_t^\perp + O(\rho_1\|v_t\| + \rho_3\|v_t\|^3) \\
&= L(\theta_t^\dagger) + \frac{1}{\eta}\overset{\star}{x}_t^2 + \frac{1}{2}{\overset{\star}{v}_t^\perp}{}^T\nabla^2 L_t \overset{\star}{v}_t^\perp + O(\eta^{-1}\delta^2\epsilon) \\
&= L(\theta_t^\dagger) + \frac{1}{\eta}\overset{\star}{x}_t^2 + O(\eta^{-1}\delta^2\epsilon),
\end{aligned}$$

where the last line follows from Assumption 5.

■

K.3. Proof of Auxiliary Lemmas

Proof [Proof of Lemma 19] Taylor expanding the update for θ_{t+1} about $\theta_t^\dagger$, we get

$$\begin{aligned}
\theta_{t+1} &= \theta_t - \eta\nabla L(\theta_t) \\
&= \theta_t - \eta\nabla L_t - \eta\nabla^2 L_t v_t - \frac{1}{2}\eta\nabla^3 L_t(v_t, v_t) + O\left(\eta\rho_4\|v_t\|^3\right)
\end{aligned}$$

Additionally, recall that the update for $\theta_{t+1}^\dagger$ is

$$\theta_{t+1}^\dagger = \theta_t^\dagger - \eta P_{\nabla S_t^\perp}^\perp \nabla L_t + O\left(\epsilon^2 \cdot \eta\|\nabla L_t\|\right).$$

Subtracting the previous 2 equations and expanding out $\nabla^3 L(v_t, v_t)$ via the non-worst-case bounds, we obtain

$$\begin{aligned}
v_{t+1} &= (I - \eta\nabla^2 L_t)v_t - \eta(\nabla L_t - P^{\perp}_{\nabla S_t^{\perp}}\nabla L_t) - \frac{1}{2}\eta x_t^2 \nabla S_t - \eta x_t \nabla^3 L_t(u_t, v_t^{\perp}) - \frac{1}{2}\eta\nabla^3 L_t(v_t^{\perp}, v_t^{\perp}) \\
&\quad + O\Big(\eta\rho_4\|v_t\|^3 + \epsilon^2 \cdot \eta\|\nabla L_t\|\Big) \\
&= (I - \eta\nabla^2 L_t)v_t - \eta\left[\frac{\nabla L \cdot \nabla S^{\perp}}{\|\nabla S^{\perp}\|^2}\right]\nabla S_t^{\perp} - \frac{1}{2}\eta x_t^2 \nabla S_t - \eta x_t \nabla^3 L_t(u_t, v_t^{\perp}) \\
&\quad + O\Big(\eta\rho_3\epsilon\|v_t\|^2 + \eta\rho_4\|v_t\|^3 + \epsilon^2 \cdot \eta\|\nabla L_t\|\Big) \\
&= (I - \eta\nabla^2 L_t)v_t + \eta\nabla S_t^{\perp}\left[\frac{\epsilon_t^2 - x_t^2}{2}\right] - \frac{1}{2}\eta x_t^2 \nabla S_t \cdot u_t u_t - \eta x_t \nabla^3 L_t(u_t, v_t^{\perp}) \\
&\quad + O\left(\epsilon^2 \cdot \frac{\|v_t\|^2}{\delta} + \epsilon^2 \cdot \frac{\|v_t\|^3}{\delta^2} + \epsilon^3\delta\right) \\
&= (I - \eta\nabla^2 L_t)v_t + \eta\nabla S_t^{\perp}\left[\frac{\epsilon_t^2 - x_t^2}{2}\right] - \frac{1}{2}\eta x_t^2 \nabla S_t \cdot u_t u_t - \eta x_t \nabla^3 L_t(u_t, v_t^{\perp}) \\
&\quad + O\left(\epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right)
\end{aligned}$$

We would first like to compute the magnitude of v_{t+1}.

$$\|v_{t+1}\| = O\left(\|v_t\| + \eta\rho_3\|v_t\|^2 + \eta\|\nabla L_t\| + \epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right).$$

Observe that by definition of ϵ and δ, and since $\|v_t\| \le \epsilon^{-1}\delta$

$$O(\eta\rho_3\|v_t\|^2) \le O\big(\|v_t\| \cdot \epsilon^{-1}\eta\rho_3\delta\big) \le O\big(\|v_t\| \cdot \epsilon^{-1}\eta\sqrt{\rho_1\rho_3}\big) \le O(\|v_t\|)$$

$$O(\epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3) \le O\big(\epsilon^2\delta + \|v_t\| \cdot \epsilon^2 \cdot (\epsilon^{-1})^2\big) \le O\big(\epsilon^2\delta + \|v_t\|\big).$$

Hence

$$\|v_{t+1}\| = O\big(\|v_t\| + \eta\|\nabla L_t\| + \epsilon^2\delta\big) = O(\|v_t\| + \epsilon\delta).$$

Note that we can bound

$$\begin{aligned}
\|u_{t+1} - u_t\| \cdot \|v_{t+1}\| &= O\big(\eta^2\rho_3\|\nabla L_t\| \cdot (\|v_t\| + \epsilon\delta)\big) \\
&= O\big(\epsilon^2 \cdot (\|v_t\| + \epsilon\delta)\big) \\
&\le O\big(\epsilon^2 \cdot \max(\|v_t\|, \delta)\big).
\end{aligned}$$

Therefore, the one-step update in the u_t direction is:

$$\begin{aligned}
x_{t+1} &= v_{t+1} \cdot u_{t+1} \\
&= v_{t+1} \cdot u_t + O\big(\epsilon^2 \cdot \max(\|v_t\|, \delta)\big) \\
&= -v_t \cdot u_t - \frac{1}{2}\eta x_t^2 \nabla S_t \cdot u_t - \eta x_t \nabla S_t \cdot v_t^\perp + O\left(\epsilon^2 \cdot \max(\|v_t\|, \delta) + \epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right) \\
&= -x_t(1 + \eta y_t) - \frac{1}{2}\eta x_t^2 \nabla S_t \cdot u_t + O\left(\epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right) \\
&= -x_t(1 + \eta y_t) - \frac{1}{2}\eta x_t^2 \nabla S_t \cdot u_t + O\left(\epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right) \\
&= -x_t(1 + \eta y_t) - \frac{1}{2}\eta x_t^2 \nabla S_t \cdot u_t + O(E_t),
\end{aligned}$$

where we have defined the error term E_t as

$$E_t := \epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3.$$

The update in the $v^\perp$ direction is

$$\begin{aligned}
v_{t+1}^\perp &= P_{u_{t+1}}^\perp\left[(I - \eta\nabla^2 L_t)v_t + \eta \nabla S_t^\perp \left[\frac{\epsilon_t^2 - x^2}{2}\right]\right] - \frac{1}{2}\eta x_t^2 \nabla S_t \cdot u_t P_{u_{t+1}}^\perp u_t - \eta x_t P_{u_{t+1}}^\perp \nabla^3 L_t(u_t, v_t^\perp) \\
&\quad + O\left(\epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right) \\
&= P_{u_{t+1}}^\perp\left[(I - \eta\nabla^2 L_t)P_{u_t}^\perp v_t + \eta \nabla S_t^\perp \left[\frac{\epsilon_t^2 - x^2}{2}\right]\right] - x_t P_{u_{t+1}}^\perp u_t - \frac{1}{2}\eta x_t^2 \nabla S_t \cdot u_t P_{u_{t+1}}^\perp u_t - \eta x_t P_{u_{t+1}}^\perp \nabla^3 L_t(u_t, v_t^\perp \\
&\quad + O\left(\epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right)
\end{aligned}$$

First, observe that

$$\left\|P_{u_{t+1}}^\perp u_t\right\| = \left\|u_t - u_{t+1}u_{t+1}^T u_t\right\| \le \|u_t - u_{t+1}\|^2 \le O(\|u_t - u_{t+1}\|)$$

Therefore we can control the first of the error terms as

$$\begin{aligned}
\left\|x_t P_{u_{t+1}}^\perp u_t + \frac{1}{2}\eta x_t^2 \nabla S_t \cdot u_t P_{u_{t+1}}^\perp u_t\right\| &\le O\Big(\|u_t - u_{t+1}\| \cdot (\|v_t\| + \eta\rho_3\|v_t\|^2)\Big) \\
&\le O(\|u_t - u_{t+1}\| \cdot \|v_t\|) \\
&\le O\big(\epsilon^2\|v_t\|\big),
\end{aligned}$$

As for the second error term, we can decompose

$$\left\|\eta x_t P_{u_{t+1}}^\perp \nabla^3 L_t(u_t, v_t^\perp)\right\| \le \eta\|v_t\|\Big(\left\|P_{u_t}^\perp \nabla^3 L_t(u_t, v_t^\perp)\right\| + \left\|P_{u_t}^\perp - P_{u_{t+1}}^\perp\right\|\left\|\nabla^3 L_t(u_t, v_t^\perp)\right\|\Big).$$

By Assumption 5, we have $\left\|P_{u_t}^{\perp}\nabla^3 L_t(u_t, v_t^{\perp})\right\| \le O(\epsilon\rho_3\|v_t\|)$. Additionally, $\left\|P_{u_t}^{\perp} - P_{u_{t+1}}^{\perp}\right\| \le O(\|u_t - u_{t+1}\|)$. Therefore

$$\begin{aligned}\left\|\eta x_t P_{u_{t+1}}^{\perp}\nabla^3 L_t(u_t, v_t^{\perp})\right\| &\le O(\epsilon\rho_3\|v_t\|\cdot\eta\|v_t\| + \eta\|v_t\|\|u_{t+1} - u_t\|\cdot\rho_3\|v_t\|)\\ &\le O\left(\epsilon\eta\rho_3\|v_t\|^2 + \eta\rho_3\|v_t\|^2\epsilon^2\right)\\ &\le O\left(\epsilon^2\frac{\|v_t\|^2}{\delta} + \epsilon^2\|v_t\|\right)\\ &= O\left(\epsilon^2\delta\cdot\max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right)\end{aligned}$$

where we used $\eta\rho_3\|v_t\| = O(1)$. Altogether, we have

$$\begin{aligned}v_{t+1}^{\perp} &= P_{u_{t+1}}^{\perp}\left[(I - \eta\nabla^2 L_t)P_{u_t}^{\perp}v_t + \eta\nabla S_t^{\perp}\left[\frac{\epsilon_t^2 - x^2}{2}\right]\right] + O\left(\epsilon^2\delta\cdot\max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right)\\ &= P_{u_{t+1}}^{\perp}\left[(I - \eta\nabla^2 L_t)P_{u_t}^{\perp}v_t + \eta\nabla S_t^{\perp}\left[\frac{\epsilon_t^2 - x^2}{2}\right]\right] + O(E_t)\end{aligned}$$

We next compute the two-step update for x_t:

$$\begin{aligned}x_{t+2} &= -x_{t+1}(1 + \eta y_{t+1}) - \frac{1}{2}\eta x_{t+1}^2\nabla S_{t+1}\cdot u_{t+1} + O(E_{t+1})\\ &= x_t(1 + \eta y_t)(1 + \eta y_{t+1}) + \frac{\eta}{2}\left(\eta y_t x_t^2\nabla S_t\cdot u_t + x_t^2\nabla S_t\cdot u_t - x_{t+1}^2\nabla S_{t+1}\cdot u_{t+1}\right) + O((1 + \eta\rho_3\|v_t\|)E_t + E_{t+1}\end{aligned}$$

We previously obtained $\eta\rho_3\|v_t\| = O(1)$. Furthermore,

$$\begin{aligned}E_{t+1} &= \epsilon^2\delta\cdot\max\left(1, \frac{\|v_{t+1}\|}{\delta}\right)^3\\ &= O\left(\epsilon^2\delta\cdot\max\left(1, \frac{\|v_t\|}{\delta} + \epsilon\right)^3\right)\\ &= O\left(\epsilon^2\delta\cdot\max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right)\\ &= O(E_t).\end{aligned}$$

Hence

$$x_{t+2} = x_t(1 + \eta y_t)(1 + \eta y_{t+1}) + \frac{\eta}{2}\left(\eta y_t x_t^2\nabla S_t\cdot u_t + x_t^2\nabla S_t\cdot u_t - x_{t+1}^2\nabla S_{t+1}\cdot u_{t+1}\right) + O(E_t).$$

The first of these two error terms can be bounded as

$$\left|\frac{1}{2}\eta^2 y_t x_t^2\nabla S_t\cdot u_t\right| \le O\left(\eta^2\rho_3^2\|v_t\|^3\right) \le O\left(\epsilon^2\cdot\frac{\|v_t\|^3}{\delta^2}\right).$$

As for the second term, we can bound

$$\begin{aligned}|\nabla S_{t+1} \cdot u_{t+1} - \nabla S_t \cdot u_t| &\le |u_{t+1} \cdot (\nabla S_{t+1} - \nabla S_t)| + |\nabla S_t \cdot (u_{t+1} - u_t)| \\ &\le \|\nabla S_{t+1} - \nabla S_t\| + O(\rho_3) \cdot \|u_{t+1} - u_t\| \\ &\le O\big(\eta^2 \rho_3^2 \|\nabla L_t\|\big) \\ &\le O(\epsilon^2 \rho_3)\end{aligned}$$

Additionally, we have

$$x_{t+1} = -x_t + O(\eta\rho_3\|v_t\|^2 + E_t).$$

Therefore

$$\begin{aligned}\eta\big|x_{t+1}^2 \nabla S_{t+1} \cdot u_{t+1} - x_t^2 \nabla S_t \cdot u_t\big| &\le \eta x_t^2 |\nabla S_{t+1} \cdot u_{t+1} - \nabla S_t \cdot u_t| + \eta(x_{t+1}^2 - x_t^2)|\nabla S_{t+1} \cdot u_{t+1}| \\ &\le O\Big(\eta\rho_3\|v_t\|^2 \cdot \epsilon^2 + \eta\rho_3\|v_t\|\Big(\eta\rho_3\|v_t\|^2 + E_t\Big)\Big) \\ &\le O\left(\epsilon^2\|v_t\| + \epsilon^2 \cdot \frac{\|v_t\|^3}{\delta^2} + E_t\right) \\ &= O(E_t).\end{aligned}$$

Altogether, the two-step update for x_t is

$$x_{t+2} = x_t(1 + \eta y_t)(1 + \eta y_{t+1}) + O(E_t).$$

Additionally, the two-step update for $v_t^\perp$ is

$$\begin{aligned}v_{t+2}^\perp &= P_{u_{t+2}}^\perp\left[(I - \eta\nabla^2 L_{t+1})P_{u_{t+1}}^\perp v_{t+1} + \eta\nabla S_{t+1}^\perp\left[\frac{\epsilon_{t+1}^2 - x_{t+1}^2}{2}\right]\right] + O(E_{t+1}) \\ &= P_{u_{t+2}}^\perp(I - \eta\nabla^2 L_{t+1})P_{u_{t+1}}^\perp(I - \eta\nabla^2 L_t)P_{u_t}^\perp v_t + \eta P_{u_{t+2}}^\perp(I - \eta\nabla^2 L_{t+1})P_{u_{t+1}}^\perp \nabla S_t^\perp\left[\frac{\epsilon_t^2 - x_t^2}{2}\right] \\ &\quad + \eta P_{u_{t+2}}^\perp \nabla S_{t+1}^\perp\left[\frac{\epsilon_{t+1}^2 - x_{t+1}^2}{2}\right] + O(E_t).\end{aligned}$$

Define $\overline{v}_{t+1} = \mathrm{step}_t(v_t), \overline{v}_{t+2} = \mathrm{step}_{t+1}(\overline{v}_t)$, and $\overline{x}_i = \overline{v}_i \cdot u_i, \overline{y}_i = \nabla S_i^\perp \cdot \overline{v}_i$ for $i \in \{t+1, t+2\}$. By the definition of step, one sees that

$$\left\|\overline{v}_{t+1}^\perp - v_{t+1}^\perp\right\| \le O(E_t).$$

and

$$|\overline{x}_{t+1} - x_{t+1}| \le \frac{1}{2}\eta x_t^2|\nabla S_t \cdot u_t| + O(E_t) \le O(\eta\rho_3\|v_t\|^2 + E_t)$$

The update for x after applying step is

$$\begin{aligned}\overline{x}_{t+2} &= -\overline{x}_{t+1}(1 + \eta\overline{y}_{t+1}) \\ &= x_t(1 + \eta y_t)(1 + \eta\overline{y}_{t+1}).\end{aligned}$$

Therefore

$$\begin{aligned}|x_{t+2} - \overline{x}_{t+2}| &\le O\big(|x_t|\eta\big|y_{t+1} - \overline{y}_{t+1}\big|\big) + O(E_t) \\ &\le O\Big(\eta\rho_3\|v_t\|\Big\|v_{t+1}^\perp - \overline{v}_{t+1}^\perp\Big\|\Big) + O(E_t) \\ &\le O(E_t).\end{aligned}$$

Additionally, the update for $v^\perp$ is

$$\overline{v}_{t+2}^\perp = P_{u_{t+2}}^\perp(I - \eta\nabla^2 L_{t+1})P_{u_{t+1}}^\perp(I - \eta\nabla^2 L_t)P_{u_t}^\perp v_t + \eta P_{u_{t+2}}^\perp(I - \eta\nabla^2 L_{t+1})P_{u_{t+1}}^\perp \nabla S_t^\perp\left[\frac{\epsilon_t^2 - x_t^2}{2}\right]$$
$$+ \eta P_{u_{t+2}}^\perp \nabla S_{t+1}^\perp\left[\frac{\epsilon_{t+1}^2 - \overline{x}_{t+1}^2}{2}\right].$$

Therefore

$$\begin{aligned}\Big\|v_{t+2}^\perp - \overline{v}_{t+2}^\perp\Big\| &\le O\big(\eta\|\nabla S_{t+1}\|(x_{t+1}^2 - \overline{x}_{t+1}^2) + E_t\big) \\ &\le O(\eta\rho_3\|v_t\||\overline{x}_{t+1} - x_{t+1}| + E_t) \\ &\le O\Big(\eta^2\rho_3^2\|v_t\|^3 + E_t\Big) \\ &\le O\left(\epsilon^2 \cdot \frac{\|v_t\|^3}{\delta^2} + E_t\right) \\ &= O(E_t)\end{aligned}$$

Altogether, we get that

$$\|r_t\| \le O(E_t) = O\left(\epsilon^2\delta \cdot \max\left(1, \frac{\|v_t\|}{\delta}\right)^3\right),$$

as desired. ■

Proof [Proof of Lemma 20] Define

$$w_t = \begin{cases} 0 & t \text{ if is even} \\ r_{t-1} & t \text{ if is odd}\end{cases}$$

and define the auxiliary trajectory $\widehat{v}$ by $\widehat{v}_0 = v_0$ and $\widehat{v}_{t+1} = \mathrm{step}(\widehat{v}_t) + w_t$. I first claim that $\widehat{v}_t = v_t$ for all even $t \le \mathscr{T}$, which we will prove by induction on t. The base case is given by assumption so assume the result for some even $t \ge 0$. Then,

$$\begin{aligned}v_{t+2} &= \mathrm{step}_{t+1}(\mathrm{step}_t(v_t)) + r_t \\ &= \mathrm{step}_{t+1}(\mathrm{step}_t(\widehat{v}_t)) + r_t \\ &= \mathrm{step}_{t+1}(\widehat{v}_{t+1}) + w_{t+1} \\ &= \widehat{v}_{t+2}\end{aligned}$$

which completes the induction.

Next, we will prove by induction that for $t \le \mathscr{T}$,

$$\Big\|\widehat{v}_t^\perp - \overset{\star}{v}{}_t^\perp\Big\|, |\widehat{x}_t - \overset{\star}{x}_t| \le O(\epsilon\delta) \le c_2\delta.$$

By definition, $\widehat{v}_0 = v_0 = \overset{\star}{v}_0$, so the claim is clearly true for $t = 0$. Next, assume the claim holds for t. If t is even then $\|w_t\| = 0$; otherwise $\|v_t\| \le 2c_2\delta$, and thus

$$\|w_t\| \le O\Big(\epsilon^2\delta \cdot \max(1, c_2)^3\Big) \le O\big(\epsilon^2\delta\big).$$

First observe that

$$\begin{aligned}
\left\|\widehat{v}_{t+1}^{\perp} - \overset{\star}{v}_{t+1}^{\perp}\right\| &\le \left\|(I - \eta\nabla^2 L_t)(\widehat{v}_t^{\perp} - \overset{\star}{v}_t^{\perp})\right\| + \frac{\eta\rho_3\left|\widehat{x}_t^2 - \overset{\star}{x}_t^2\right|}{2} + \|w_t\| \\
&\le \big(1 + \eta\left|\lambda_{min}(\nabla^2 L_t)\right|\big)\left\|\widehat{v}_t^{\perp} - \overset{\star}{v}_t^{\perp}\right\| + O(\epsilon)\cdot|\widehat{x}_t - \overset{\star}{x}_t| + O\big(\epsilon^2\delta\big) \\
&\le \big(1 + \eta\left|\lambda_{min}(\nabla^2 L_t)\right|\big)\left\|\widehat{v}_t^{\perp} - \overset{\star}{v}_t^{\perp}\right\| + O(\epsilon\delta)\cdot\left|\frac{\widehat{x}_t - \overset{\star}{x}_t}{\overset{\star}{x}_t}\right| + O\big(\epsilon^2\delta\big)
\end{aligned}$$

Next, note that

$$\begin{aligned}
\frac{\widehat{x}_{t+1}}{\overset{\star}{x}_{t+1}} &= \frac{(1 + \eta\widehat{y}_t)\widehat{x}_t + O(\epsilon^2\delta)}{(1 + \eta\overset{\star}{y}_t)\overset{\star}{x}_t + O(\epsilon^2\delta)} \\
&= \frac{(1 + \eta\overset{\star}{y}_t)\widehat{x}_t + O(\epsilon^2\delta) + O(\epsilon)\cdot\left\|\widehat{v}_t^{\perp} - \overset{\star}{v}_t^{\perp}\right\|}{(1 + \eta\overset{\star}{y}_t)\overset{\star}{x}_t + O(\epsilon^2\delta)} \\
&= \frac{\widehat{x}_t}{\overset{\star}{x}_t} + O\Big(\epsilon^2 + \frac{\epsilon}{\delta}\left\|\widehat{v}_t^{\perp} - \overset{\star}{v}_t^{\perp}\right\|\Big).
\end{aligned}$$

Therefore

$$\left|\frac{\widehat{x}_{t+1} - \overset{\star}{x}_{t+1}}{\overset{\star}{x}_{t+1}}\right| \le \left|\frac{\widehat{x}_t - \overset{\star}{x}_t}{\overset{\star}{x}_t}\right| + O(\epsilon^2 + \frac{\epsilon}{\delta}\left\|\widehat{v}_t^{\perp} - \overset{\star}{v}_t^{\perp}\right\|).$$

Let $d_t = \max\left(\left\|\widehat{v}_t^{\perp} - \overset{\star}{v}_t^{\perp}\right\|, \delta\left|\frac{\widehat{x}_t - \overset{\star}{x}_t}{\overset{\star}{x}_t}\right|\right)$. Then

$$\begin{aligned}
\left\|\widehat{v}_{t+1}^{\perp} - \overset{\star}{v}_{t+1}^{\perp}\right\| &\le (1 + \eta\left|\lambda_{min}(\nabla^2 L_t)\right| + O(\epsilon))d_t + O(\epsilon^2\delta) \\
\delta\left|\frac{\widehat{x}_{t+1} - \overset{\star}{x}_{t+1}}{\overset{\star}{x}_{t+1}}\right| &\le (1 + O(\epsilon))d_t + O(\epsilon^2\delta).
\end{aligned}$$

Therefore

$$\begin{aligned}
d_{t+1} &\le (1 + \eta\left|\lambda_{min}(\nabla^2 L_t)\right| + O(\epsilon))d_t + O(\epsilon^2\delta) \\
&\le (1 + O(\epsilon))d_t + O(\epsilon^2\delta),
\end{aligned}$$

so for $t \le \mathscr{T}$ we have $d_{t+1} \le O(\epsilon\delta)$. Therefore

$$\left\|\widehat{v}_{t+1}^{\perp} - \overset{\star}{v}_{t+1}^{\perp}\right\|, |\widehat{x}_{t+1} - \overset{\star}{x}_{t+1}| \le O(\epsilon\delta) \le c_2\delta,$$

so the induction is proven. Altogether, we get $\|\widehat{v}_t - \overset{\star}{v}_t\| \le O(\epsilon\delta)$ for all such t, as desired. ■

Proof [Proof of Lemma 21] Recall that

$$x_{t+1}^* = -(1 + \eta y_t^*)x_t^* \quad \text{and} \quad y_{t+1}^* = \eta\sum_{s=0}^{t}\beta_{s\to t}\left[\frac{\delta_s^2 - x_s^{*2}}{2}\right],$$

Since $t \le \frac{1}{\eta\max_t|\lambda_{min}(\nabla^2 L_t)|}$, we have that $\beta_{s\to t} = O(\rho_3^2)$, and thus

$$\overset{\star}{y}_t \le O(\rho_3^2)t\eta\delta^2 = O(\sqrt{\rho_1\rho_3}).$$

Therefore

$$|\overset{\star}{x}_{t+1}| = (1 + \eta \overset{\star}{y}_t)|\overset{\star}{x}_t| \leq (1 + O(\epsilon))|\overset{\star}{x}_t|.$$

Since $t \leq O(\epsilon^{-1})$, $|\overset{\star}{x}_t|$ grows by at most a constant factor, and thus $|\overset{\star}{x}_t| \leq O(\delta)$. Finally, recall that

$$\overset{\star}{v}_{t+1}^{\perp} = \eta \sum_{s=0}^{t} P_{u_{t+1}}^{\perp} \left[\prod_{k=t}^{s+1} A_k \right] \nabla S_s^{\perp} \left[\frac{\delta_s^2 - x_s^{*2}}{2} \right].$$

By the triangle inequality,

$$\left\| \overset{\star}{v}_{t+1}^{\perp} \right\| \leq O(\eta t \rho_3 \delta^2) \leq O(\delta).$$

Therefore $\|\overset{\star}{v}_t\| \leq O(\delta)$. ∎