Landscaping Linear Mode Connectivity

Sidak Pal Singh, *ETH Zürich, Switzerland* SSIDAK@ETHZ.CH
Linara Adilova, *Ruhr-Universität Bochum, Germany* ADYLOVA.LINARA.R@GMAIL.COM
Michael Kamp, *Institute for AI in Medicine IKIM, Germany* INFO@MICHAELKAMP.ORG
Asja Fischer, *Ruhr-Universität Bochum, Germany* ASJA.FISCHER@RUB.DE
Bernhard Schölkopf, *MPI-IS Tübingen, Germany* BS@TUEBINGEN.MPG.DE
Thomas Hofmann, *ETH Zürich, Switzerland* THOMAS.HOFMANN@INF.ETHZ.CH

Abstract

The presence of linear paths in parameter space between two different network solutions in certain cases, i.e., linear mode connectivity (LMC) [6], has garnered interest from both theoretical and practical fronts. There has been significant research that either practically designs algorithms catered for connecting networks by adjusting for the permutation symmetries as well as some others that more theoretically construct paths through which networks can be connected [11]. Yet, the core reasons for the occurrence of LMC, when in fact it does occur, in the highly non-convex loss landscapes of neural networks are far from clear. In this work, we take a step towards understanding it by providing a model of how the loss landscape needs to behave topographically for LMC (or the lack thereof) to manifest. Concretely, we present a 'mountainside and ridge' perspective that helps to neatly tie together different geometric features that can be spotted in the loss landscape along the training runs. We also complement this perspective by providing a theoretical analysis of the barrier height, for which we provide empirical support, and which additionally extends as a faithful predictor of layer-wise LMC. We close with a toy example that provides further intuition on how barriers arise in the first place, all in all, showcasing the larger aim of the work — to provide a working model of the landscape and its topography for the occurrence of LMC.

1. Introduction

The loss landscape of over-parameterized neural networks, in general, and especially when taken as a whole, is undoubtedly non-convex. Yet, the confrontation with the still somewhat puzzling success of a local gradient based method to find generalizing solutions has led to a body of work that explores inherent regularity within optimization, such as through the lens of implicit bias [9, 12]; or sources of structure and regularity within the landscape itself, like via its significant degeneracy [13, 17], inherent symmetries [4, 15], or existence of monotonic paths during training [8] as well as non-linear/linear paths between solutions [3, 6, 7] — all of which are likened to play a palliative role against the non-convexity. From this latter category, a particularly striking notion of regularity is that of linear mode connectivity [6], where it has been observed that if two networks are made to share a common initial path of sufficient length (usually around 10% of the whole training) and subsequently set apart on distinct paths from a 'fork' (e.g., achieved via enforcing different orderings of the samples), they can nevertheless be connected with a linear path at convergence, along which the loss remains negligible.

The particular interest of the research community in LMC can be attributed to, amongst other factors, the sheer simplicity with which a notion of convexity is demonstrated in the landscapes, as well as the practical implications it raises for model merging/fusion [2, 16]. As a result, there is a lot of research which tries to theoretically construct sets of paths [5, 11, 14] to demonstrate LMC, as well

as others which consider its extensions to convex hulls [18], feature [19] and layer connectivity [1], and increasingly, those that study its wider implications on generalization strategies [10]. However, amidst all these advances, our broader model of the loss landscape has seen little refinement. *Hence, **our objective** is to precisely take a step towards this, by providing a model of the landscape that allows for LMC (or lack thereof) and explicates the various observations in this context.*

A metaphor for LMC. Let us momentarily engage in a metaphor that explains our model.

> Imagine going down from near the top of a mountain towards the valley with a friend. With the weather being extremely foggy and windy, your visibility is negligible and you both chose to descend by locally following the downward slope. However, after a while into your hike downwards, you can't spot your friend and realize that you might have lost them on the way. Confident that you could not have diverged for long, you continue to march down with the hope that you will find them at the valley floor without strenuous effort. It takes some walking once you are finally down, but you soon run into them and are relieved. However, as you both gaze upwards, you realize that thankfully you did not separate around the top of the mountain itself, for a deviation then would tantamount to you both being on the different sides of the mountain, with a non-trivial barrier in between.

2. The Topography of Landscapes vis-à-vis LMC

Key Hypothesis. While the above is merely a metaphor, and not free from flaws, the resemblance to LMC is hard to ignore, and we take some inspiration from it to arrive at a hypothesis. Namely, that the occurrence of LMC can be explained by models moving down a mountain side with numerous ridges that are present at varying heights. If one forks on top of a plateau or a wide ridge, the child models may move to either side of it, effectively resulting in a barrier that completely prevents linear connectivity. In contrast, if one forks a little later on one side of the higher ridge, models largely remain on the same side with the mountain slope pulling them downwards, and if separated, are only divided by a lower ridge implying a small barrier.

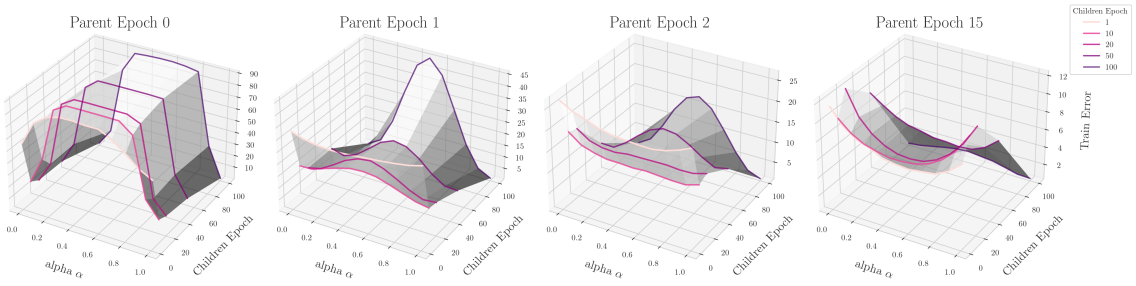
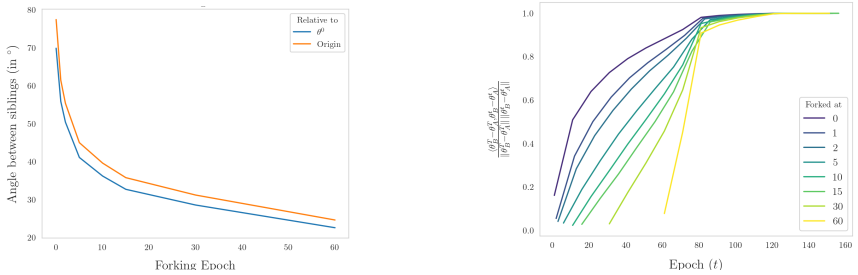


Figure 1: The evolution of training error curves, shown in different colors, when forked at different points. The barrier version (i.e., with the error of endpoints subtracted) is shown in Figure 10.

Empirical observations. Let us directly take a look at the kind of cross-sectional loss landscape that appears between the child models ('siblings') when the parent model is forked at different points in training, and in particular, how this varies while training the child models. Figure 1 shows a plot of the training error on the line segment joining the siblings for the case of ResNet20 trained on CIFAR-10, mimicking the hyperparameter setup of Frankle et al. [6] which is also detailed in Section B.1,

when forked at epochs 0 (i.e., initialization), $1, 2, 15$. The evolution of the training error is shown by laying them out in different planes, which also evokes how the siblings navigate the landscape.

We observe that when the siblings are forked from the initialization, a wide ridge (i.e., considerable portion of the line segment is at high error) separates them throughout their training just as if they set out on different sides of the mountain. But when they are forked after a few (parent) epochs, they seem to be both going downward but are soon (in the span of $10 - 20$ child epochs) separated by another ridge. This ridge although lower in height *extends all the way until convergence*. Finally, if the children get forked much later, they seem to be simply descending downwards, and are separated by essentially a bump. These observations align neatly with our 'mountain-and-ridge perspective' hypothesized above, and show most conspicuously, *that barriers do not just show up right at convergence, but can be traced a long way before in the form of a ridge.* This is in contrast to the 'static' final view of barriers portrayed in the literature, and as reproduced in Figure 7. Similar figures for different learning rate or without weight decay can be found in the Appendix B.2.2.



(*a*) Sibling angles (*b*) Sibling solution planes

Figure 2: The angle between sibling solutions (in degrees) as well as the determination of sibling solution planes for different forking epochs.

Sibling angles and solution planes. To further probe our hypothesis, we measure, for different forking points, the angle between the siblings $\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*$ formed at a base point $\boldsymbol{\theta}_{\text{base}}$, which is either the origin or the respective forking point itself. More concretely, this amounts to measuring $\arccos(\langle \boldsymbol{\theta}_1^* - \boldsymbol{\theta}_{\text{base}}, \boldsymbol{\theta}_2^* - \boldsymbol{\theta}_{\text{base}} \rangle / \|\boldsymbol{\theta}_1^* - \boldsymbol{\theta}_{\text{base}}\| \|\boldsymbol{\theta}_2^* - \boldsymbol{\theta}_{\text{base}}\|)$ and is shown in Figure 2(a). We observe that, regardless of the choice of the base point, the angle between earlier siblings is larger than that for later siblings, which is in line with our view, as shown pictorially in Figure 3 that the earlier ridges lead to more lateral or cross-sectional separation than later ridges.

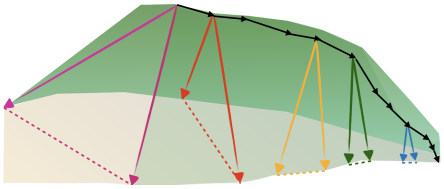


Figure 3: A sketch of different forks.

In a similar vein, we check, for different forking points, how early is the sibling solution plane determined, as measured by cosine similarity between sibling difference at step t of their training, $\boldsymbol{\theta}_1^t - \boldsymbol{\theta}_2^t$ and the final sibling difference $\boldsymbol{\theta}_1^* - \boldsymbol{\theta}_2^*$. The results are shown in Figure 2(b), where we find that the siblings forked earlier have their solution plane determined most quickly than the later ones, which would occur if the former were indeed traversing on different sides of the planes and their cross-section largely determined rapidly into training.

Lastly, we also carry out a similar experiment when the models are trained with a lower learning rate, and the results for which we can be found in Figure 13. The angles between siblings are even less and the sibling solution plane gets determined even faster, further corroborating our hypothesis.

3. Barrier Analysis

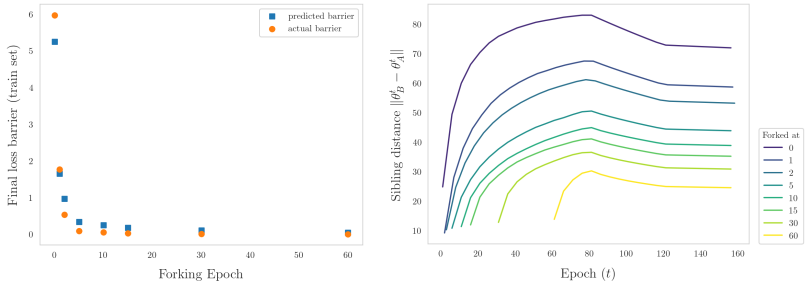
In the previous section, we saw that the various geometrical observations about the sibling solutions, across various forking points, fit neatly within the prescribed mountain-ridge hypothesis. Now, we would like to see if we can provide a model for the height of the ridge, and in particular, the eventual barrier that separates the sibling solutions.

Let us define a notion of barrier curve between the final solutions as, $\mathcal{B}(\alpha; \boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \mathcal{L}\big((1-\alpha)\boldsymbol{\theta}_1 + \alpha\boldsymbol{\theta}_2\big) - [(1-\alpha)\,\mathcal{L}(\boldsymbol{\theta}_1) + \alpha\,\mathcal{L}(\boldsymbol{\theta}_1)]$, parameterized for $\alpha \in [0, 1]$. Usually, what is reported as the barrier is the maximum value [4] of the barrier curve, and for the theoretical analysis we will consider the entire barrier curve. Further, the barrier curve is effectively a notion of non-convexity between the line segment joining two networks, which we refer to as 'cross-sectional' non-convexity. If the (maximum) barrier comes out as negative, it implies convexity on this line-segment. Below, we provide an analysis that models the barrier up to the second-order in the distance between them.

Proposition 1. *The loss barrier curve when linearly interpolating the final child networks $\boldsymbol{\theta}_1^*$ and $\boldsymbol{\theta}_2^*$ with weights $1-\alpha$ and α respectively, is given by*

$$\mathcal{B}(\alpha; \boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*) = \frac{\alpha(1-\alpha)}{2}(\boldsymbol{\theta}_2^* - \boldsymbol{\theta}_1^*)^\top \left(\alpha \nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_1^*) + (1-\alpha)\nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_2^*)\right)(\boldsymbol{\theta}_2^* - \boldsymbol{\theta}_1^*) + \mathcal{O}(\|\boldsymbol{\theta}_2^* - \boldsymbol{\theta}_1^*\|^3) \quad (1)$$

The proof is located in the appendix A. From the form of the barrier, we make the following observations. (a) We see that the barrier increases if the two child networks solutions are more distant, but more precisely, the metric is with respect to the geometry of the convex-combination of their Hessians. (b) Since the networks are at convergence, the Hessian will be positive semi-definite by second-order optimality condition, the predicted barrier will always be non-negative though it might be small, unless higher-order terms kick in. (c) If we assume that the form of the local curvatures is similar (this doesn't mean they have to be the same, but more that the dominant eigenvectors are aligned), then we can approximate the barrier by $\mathcal{B}(\alpha; \boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*) \approx \frac{\alpha(1-\alpha)}{2}(\boldsymbol{\theta}_2^* - \boldsymbol{\theta}_1^*)^\top \cdot \nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_1^*) \cdot (\boldsymbol{\theta}_2^* - \boldsymbol{\theta}_1^*)$, and from where it is easy to see that it is maximized for $\alpha = 1/2$, as often seen in practice.

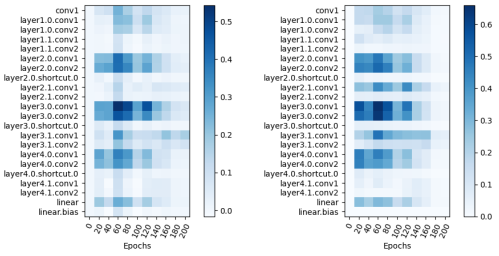


(*a*) Predicted vs Actual Barriers (*b*) Sibling Distances

Figure 4: *Left:* Barrier predictions versus actual barriers for final solutions obtained by forking the parent trajectory at different epochs in its training. *Right:* The evolution of the distance between sibling models for different forking points.

Next, in Figure 4(a) we compare the fidelity of our barrier predictions with what is observed empirically. We find that these predictions fall in decent ballpark of the actual barriers, even though the distance between the models under consideration is significant as evident from 4(b), and where the higher-order terms should come into the picture.

Extension to layerwise LMC. Adilova et al. [1] suggested a notion of layerwise LMC, where only the parameters of single individual layers are linearly interpolated. Layerwise LMC can thus provide a more fine-grained view of (non)convexity and has also been observed to hold even when the entire networks may not be linearly connected. We extend our barrier analysis to the layerwise case in Proposition 2, and Figure 5 shows that these predictions provide a rather faithful match with the true layerwise barriers. Besides, Appendix B.4.2 contains additional results, where we find that our barrier predictions capture the magnitude of the actual barriers, and that both the Hessian term in the barrier expression above as well as the distance between the parameters are important to model the barrier closely.



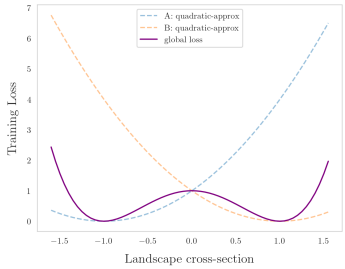
(*a*) loss barrier (*b*) predicted barrier

Figure 5: (Left) The layerwise loss barrier as per [1]. **(Right):** Layerwise predicted barrier, as per Proposition 2, during the course of training ResNet18 (see Appendix B.4.1 for details).

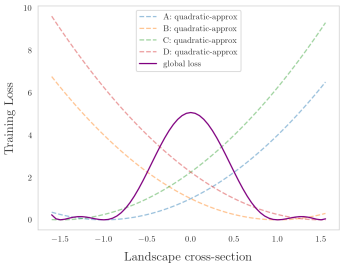
4. Discussion

Summary. To sum up, we provide a new unified perspective to think about LMC, inspired by a mountainside-ridge view, that explains various aspects of LMC such as how early and later forked solutions are situated in the landscape, and how they can be disconnected or connected. We also present a theoretical analysis which indicates that the extent of connectivity depends on the sibling distance and the local sibling curvature, and demonstrate that it can empirically provide decent barrier estimates.

A retrospective on barriers. In an alternate way, let us further consolidate our intuition on how barriers can arise at all. Consider a 1-dimensional toy example below, with two minima, one at $\theta_1^* = -1$ and the other at $\theta_2^* = 1$. Locally, since both are valid local minima, a quadratic approximation explains the loss surface, i.e., $(\theta - \theta_1^*)^2$ and $(\theta - \theta_2^*)^2$ respectively. Then the simplest model of the landscape that would jointly have both these local minima is simply the product of the two quadratic approximations, i.e., $(\theta - \theta_1^*)^2 \cdot (\theta - \theta_2^*)^2$. Notice how this joint loss model has a barrier in between the two minima, as shown in Figure 6(a). In fact, since we don't just have two solutions, we can extend the above example to include two further minima at $-1.5, 1.5$ as shown in Figure 6(b), and we can imagine the larger idea of how a hierarchy of barriers might emerge, with the barrier between distant solutions being higher — something which the Proposition 1 also shows, but which additionally accounts for the local curvatures.



(*a*) two solutions



(*b*) multiple solutions

Figure 6: Toy examples: loss landscape cross-section.

Future Work. There are still several interesting questions that require further study: (a) Is there an early geometric indicator which can predict the extent of the final barrier? (b) Can we go beyond the second-order model of the barrier, while being efficient, and further refine the barrier predictions? (c) Given the rapidly determined cross-sectional direction, can it be utilized for model fusion without having to train all the child networks until convergence?

References

[1] Linara Adilova, Maksym Andriushchenko, Michael Kamp, Asja Fischer, and Martin Jaggi. Layer-wise linear mode connectivity. In *The Twelfth International Conference on Learning Representations*, 2023.

[2] Samuel Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. In *The Eleventh International Conference on Learning Representations*, 2022.

[3] Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred Hamprecht. Essentially no barriers in neural network energy landscape. In *International conference on machine learning*, pages 1309–1318. PMLR, 2018.

[4] Rahim Entezari, Hanie Sedghi, Olga Saukh, and Behnam Neyshabur. The role of permutation invariance in linear mode connectivity of neural networks. In *International Conference on Learning Representations*, 2021.

[5] Damien Ferbach, Baptiste Goujaud, Gauthier Gidel, and Aymeric Dieuleveut. Proving linear mode connectivity of neural networks via optimal transport. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li, editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 3853–3861. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/ferbach24a.html.

[6] Jonathan Frankle, Gintare Karolina Dziugaite, Daniel Roy, and Michael Carbin. Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning*, pages 3259–3269. PMLR, 2020.

[7] Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew G Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. *Advances in neural information processing systems*, 31, 2018.

[8] Ian J Goodfellow, Oriol Vinyals, and Andrew M Saxe. Qualitatively characterizing neural network optimization problems. *ICLR*, 2015.

[9] Suriya Gunasekar, Jason Lee, Daniel Soudry, and Nathan Srebro. Characterizing implicit bias in terms of optimization geometry. In *International Conference on Machine Learning*, pages 1832–1841. PMLR, 2018.

[10] Jeevesh Juneja, Rachit Bansal, Kyunghyun Cho, João Sedoc, and Naomi Saphra. Linear connectivity reveals generalization strategies, 2023.

[11] Rohith Kuditipudi, Xiang Wang, Holden Lee, Yi Zhang, Zhiyuan Li, Wei Hu, Sanjeev Arora, and Rong Ge. Explaining landscape connectivity of low-cost solutions for multilayer nets, 2020.

[12] Edward Moroshko, Suriya Gunasekar, Blake Woodworth, Jason D. Lee, Nathan Srebro, and Daniel Soudry. Implicit bias in deep linear classification: Initialization scale vs training accuracy, 2020.

[13] Levent Sagun, Utku Evci, V Ugur Guney, Yann Dauphin, and Leon Bottou. Empirical analysis of the hessian of over-parametrized neural networks. *arXiv preprint arXiv:1706.04454*, 2017.

[14] Alexander Shevchenko and Marco Mondelli. Landscape connectivity and dropout stability of SGD solutions for over-parameterized neural networks. *CoRR*, abs/1912.10095, 2019. URL http://arxiv.org/abs/1912.10095.

[15] Berfin Simsek, François Ged, Arthur Jacot, Francesco Spadaro, Clément Hongler, Wulfram Gerstner, and Johanni Brea. Geometry of the loss landscape in overparameterized neural networks: Symmetries and invariances. In *International Conference on Machine Learning*, pages 9722–9732. PMLR, 2021.

[16] Sidak Pal Singh and Martin Jaggi. Model fusion via optimal transport. *Advances in Neural Information Processing Systems*, 33:22045–22055, 2020.

[17] Sidak Pal Singh, Gregor Bachmann, and Thomas Hofmann. Analytic insights into structure and rank of neural network hessian maps. *Advances in Neural Information Processing Systems*, 34:23914–23927, 2021.

[18] David Yunis, Kumar Kshitij Patel, Pedro Henrique Pamplona Savarese, Gal Vardi, Jonathan Frankle, Matthew Walter, Karen Livescu, and Michael Maire. On convexity and linear mode connectivity in neural networks. In *OPT 2022: Optimization for Machine Learning (NeurIPS 2022 Workshop)*, 2022.

[19] Zhanpeng Zhou, Yongyi Yang, Xiaojiang Yang, Junchi Yan, and Wei Hu. Going beyond linear mode connectivity: The layerwise linear feature connectivity, 2023.

Appendix A. Omitted Proofs

A.1. LMC barrier proof

Proposition 1. *The loss barrier when linearly interpolating the final child networks $\boldsymbol{\theta}_1^*$ and $\boldsymbol{\theta}_2^*$, forked from a common point $\boldsymbol{\theta}_0$, with weights $1-\alpha$ and α respectively, is given by*

$$\mathcal{B}(\alpha;\boldsymbol{\theta}_1^*,\boldsymbol{\theta}_2^*) = \frac{\alpha(1-\alpha)}{2}(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*)^\top \left(\alpha\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*) + (1-\alpha)\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_2^*)\right)(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*) + \mathcal{O}(\|\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*\|^3)$$

Proof

Using the Taylor series we have that,

$$\mathcal{L}\big((1-\alpha)\boldsymbol{\theta}_1^* + \alpha\boldsymbol{\theta}_2^*\big) = \mathcal{L}\big(\boldsymbol{\theta}_1^* + \alpha(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*)\big) \tag{2}$$

$$= \mathcal{L}(\boldsymbol{\theta}_1^*) + \alpha\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*)^\top(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*) + \frac{\alpha^2}{2}(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*)^\top\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*)(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*) + \mathcal{O}(\|\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*\|^3) \tag{3}$$

$$= \mathcal{L}(\boldsymbol{\theta}_1^*) + \frac{\alpha^2}{2}(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*)^\top\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*)(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*) + \mathcal{O}(\|\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*\|^3) \tag{4}$$

The last line follows from the fact that at the optimum, $\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*) = \mathbf{0}$ and $\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_2^*) = \mathbf{0}$. Likewise, we can repeat the above steps with $\boldsymbol{\theta}_2^*$ as the center of the Taylor series expansion, which results in:

$$\mathcal{L}\big((1-\alpha)\boldsymbol{\theta}_1^* + \alpha\boldsymbol{\theta}_2^*\big) = \mathcal{L}\big(\boldsymbol{\theta}_2^* - (1-\alpha)(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*)\big) \tag{5}$$

$$= \mathcal{L}(\boldsymbol{\theta}_2^*) + \frac{(1-\alpha)^2}{2}(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*)^\top\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_2^*)(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*) + \mathcal{O}(\|\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*\|^3) \tag{6}$$

Multiplying[1] eq. 4 by $1-\alpha$ and eq. 6 by α, and then adding them yields:

$$\mathcal{L}\big((1-\alpha)\boldsymbol{\theta}_1^* + \alpha\boldsymbol{\theta}_2^*\big) = (1-\alpha)\mathcal{L}(\boldsymbol{\theta}_1^*) + \alpha\mathcal{L}(\boldsymbol{\theta}_2^*) \tag{7}$$

$$+ \frac{\alpha(1-\alpha)}{2}(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*)^\top\left(\alpha\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*) + (1-\alpha)\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_2^*)\right)(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*) \tag{8}$$

$$+ \mathcal{O}(\|\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*\|^3) \tag{9}$$

Rearranging the terms we get that the following expression for the barrier between the two solutions:

$$\mathcal{B}(\alpha) = \frac{\alpha(1-\alpha)}{2}(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*)^\top\left(\alpha\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*) + (1-\alpha)\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_2^*)\right)(\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*) + \mathcal{O}(\|\boldsymbol{\theta}_2^*-\boldsymbol{\theta}_1^*\|^3) \tag{10}$$

■

1. Note while the factors with which the equations are multiplied are mathematically convenient to obtain the definition of the barrier, they also make sense in that when α is small, eq. 4 will be a more accurate model of the loss at the interpolation and gets a higher weight of $1-\alpha$, and likewise when α is large (or $1-\alpha$ is small) eq. 6 is weighed in more to yield an accurate model of the loss at the interpolated point.

A.2. Layerwise LMC barrier proof

Proposition 2. *The loss barrier when linearly interpolating only the layer ℓ parameters of the final child networks $\boldsymbol{\theta}_1^*$ and $\boldsymbol{\theta}_2^*$, forked from a common point $\boldsymbol{\theta}_0$, with weights $1-\alpha$ and α respectively, is*

$$\mathcal{B}_\ell(\alpha) = \frac{\alpha(1-\alpha)}{2} \Delta\boldsymbol{\theta}^*[\ell]^\top \left(\alpha\, \nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_1^*[\ell]) + (1-\alpha)\, \nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_2^*[\ell])\right) \Delta\boldsymbol{\theta}^*[\ell] + \mathcal{O}(\|\Delta\boldsymbol{\theta}^*[\ell]\|^3)$$

Proof

Instead of considering the line-segment between all the network parameters, the work of Adilova et al. [1] considers the case where only a single layer's parameters are interpolated into another model. Let us now repeat the analysis, assuming we interpolate only with the layer ℓ of the second network into the first. If we designate the layer-wise parameters by the superscript as $\boldsymbol{\theta}[\ell]$, this amounts to:

$$\boldsymbol{\theta}_{2\to1}[\ell] := (1-\alpha)\, \cdot\, \boldsymbol{\theta}_1^*[\ell] + \alpha\, \cdot \boldsymbol{\theta}_2^*[\ell]\,, \quad \text{and} \quad \boldsymbol{\theta}_{2\to1}[\ell'] := \boldsymbol{\theta}_1^*[\ell'] \quad \forall\, \ell' \neq \ell$$

Likewise, we can consider the interpolation of the layer ℓ of the first network into the second, yielding the parameters $\boldsymbol{\theta}_{1\to2}$ defined as follows:

$$\boldsymbol{\theta}_{1\to2}[\ell] := (1-\alpha)\, \cdot\, \boldsymbol{\theta}_1^*[\ell] + \alpha\, \cdot \boldsymbol{\theta}_2^*[\ell]\,, \quad \text{and} \quad \boldsymbol{\theta}_{1\to2}[\ell'] := \boldsymbol{\theta}_2^*[\ell'] \quad \forall\, \ell' \neq \ell$$

In other words, $\boldsymbol{\theta}_{2\to1}$ and $\boldsymbol{\theta}_{1\to2}$ only differ in terms of where to take the parameters for other layers, whether from $\boldsymbol{\theta}_1^*$ or from $\boldsymbol{\theta}_2^*$ respectively.

Furthermore, let $\mathbf{P}_\ell \in \mathbb{R}^{p\times p}$ stand for the diagonal matrix, with i^{th} entry is $\mathbb{1}\{\boldsymbol{\theta}_i \in \boldsymbol{\theta}[\ell]\}$, i.e., which contains 1 at the index which corresponds to the parameter from layer ℓ, represented by $\boldsymbol{\theta}[\ell]$, and 0 elsewhere. Then we can write the above parameter interpolations more succinctly as:

$$\boldsymbol{\theta}_{2\to1}^{(\ell)} = \boldsymbol{\theta}_1^* \,+\, \alpha\, \cdot\, \mathbf{P}_\ell \cdot \Delta\boldsymbol{\theta}^*\,, \quad \text{and} \quad \boldsymbol{\theta}_{1\to2}^{(\ell)} = \boldsymbol{\theta}_2^* \,-\, (1-\alpha)\, \cdot\, \mathbf{P}_\ell \cdot \Delta\boldsymbol{\theta}^*$$

where, $\Delta\boldsymbol{\theta}^* := \boldsymbol{\theta}_2^* - \boldsymbol{\theta}_1^*$. Next, we apply a second-order Taylor series to approximate the loss at these interpolations.

$$\mathcal{L}(\boldsymbol{\theta}_{2\to1}^{(\ell)}) = \mathcal{L}(\boldsymbol{\theta}_1^*) + \alpha \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_1^*)^\top \mathbf{P}_\ell \Delta\boldsymbol{\theta}^* + \frac{\alpha^2}{2} \Delta\boldsymbol{\theta}^{*\top}\, \mathbf{P}_\ell\, \nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_1^*)\, \mathbf{P}_\ell\, \Delta\boldsymbol{\theta}^* + \mathcal{O}(\|\mathbf{P}_\ell\, \Delta\boldsymbol{\theta}^*\|^3) \tag{11}$$

$$= \mathcal{L}(\boldsymbol{\theta}_1^*) + \frac{\alpha^2}{2} \Delta\boldsymbol{\theta}^{*\top}\, \mathbf{P}_\ell\, \nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_1^*)\, \mathbf{P}_\ell\, \Delta\boldsymbol{\theta}^* + \mathcal{O}(\|\mathbf{P}_\ell\, \Delta\boldsymbol{\theta}^*\|^3) \tag{12}$$

$$= \mathcal{L}(\boldsymbol{\theta}_1^*) + \frac{\alpha^2}{2} \Delta\boldsymbol{\theta}^*[\ell]^\top\, \nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_1^*[\ell])\, \Delta\boldsymbol{\theta}^*[\ell] + \mathcal{O}(\|\Delta\boldsymbol{\theta}^*[\ell]\|^3)\,, \tag{13}$$

where the second term indicates the quadratic form with the Hessian at layer ℓ multiplied by the difference in parameters on that layer ℓ (i.e., $\Delta\boldsymbol{\theta}^*[\ell]$) and the first-order terms go away since $\boldsymbol{\theta}_1^*$ is a stationary point. Similarly, we get for the other interpolation:

$$\mathcal{L}(\boldsymbol{\theta}_{1\to2}^{(\ell)}) = \mathcal{L}(\boldsymbol{\theta}_2^*) + \frac{(1-\alpha)^2}{2} \Delta\boldsymbol{\theta}^*[\ell]^\top\, \nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_2^*[\ell])\, \Delta\boldsymbol{\theta}^*[\ell] + \mathcal{O}(\|\Delta\boldsymbol{\theta}^*[\ell]\|^3)\,, \tag{14}$$

Multiplying eq. 13 by $1-\alpha$ and eq. 14 by α, we get:

$$(1-\alpha)\mathcal{L}(\boldsymbol{\theta}_{2\to 1}) + \alpha\mathcal{L}(\boldsymbol{\theta}_{1\to 2}) = (1-\alpha)\mathcal{L}(\boldsymbol{\theta}_1^*) + \alpha\mathcal{L}(\boldsymbol{\theta}_2^*) \tag{15}$$

$$+ \frac{\alpha(1-\alpha)}{2}\Delta\boldsymbol{\theta}^*[\ell]^\top \left(\alpha\,\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*[\ell]) + (1-\alpha)\,\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_2^*[\ell])\right)\Delta\boldsymbol{\theta}^*[\ell] \tag{16}$$

$$+ \mathcal{O}(\|\Delta\boldsymbol{\theta}^*[\ell]\|^3)\,, \tag{17}$$

If we now define the layer-wise barrier as,

$$\mathcal{B}_\ell(\alpha) := (1-\alpha)\mathcal{L}(\boldsymbol{\theta}^{(\ell)}_{2\to 1}) + \alpha\mathcal{L}(\boldsymbol{\theta}^{(\ell)}_{1\to 2}) - [(1-\alpha)\mathcal{L}(\boldsymbol{\theta}_1^*) + \alpha\mathcal{L}(\boldsymbol{\theta}_2^*)]\,,$$

the above analysis reduces to the following expression:

$$\mathcal{B}_\ell(\alpha) = \frac{\alpha(1-\alpha)}{2}\Delta\boldsymbol{\theta}^*[\ell]^\top \left(\alpha\,\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*[\ell]) + (1-\alpha)\,\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_2^*[\ell])\right)\Delta\boldsymbol{\theta}^*[\ell] + \mathcal{O}(\|\Delta\boldsymbol{\theta}^*[\ell]\|^3) \tag{18}$$

■

Remarks. Now, making the approximation of similar layer-wise Hessians $\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*[\ell]) \approx \nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_2^*[\ell])$, we can further simplify the above expression to:

$$\mathcal{B}_\ell(\alpha) \approx \frac{\alpha(1-\alpha)}{2}\Delta\boldsymbol{\theta}^*[\ell]^\top\ \nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*[\ell])\ \Delta\boldsymbol{\theta}^*[\ell]\,, \tag{19}$$

which also attains its maximum value for $\alpha = \frac{1}{2}$. The above expression is similar to what we had before, except our parameter update here concerns only the layer ℓ and the Hessian is also thus for this layer only. Although, here we only require much weaker assumptions, as our parameter update is local to layer ℓ and similarity of the Hessian is being considered only with respect to the diagonal block corresponding to layer ℓ.

Generalization to arbitrary set of layers. Notice that can be rewritten, at $\alpha = 1/2$, as

$$\mathcal{B}^* \approx \frac{1}{8}\sum_{\ell,\ell' \in [1,L]} \Delta\boldsymbol{\theta}^*[\ell]^\top\ \cdot\ \nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*[\ell,\ell'])\ \cdot\ \Delta\boldsymbol{\theta}^*[\ell'] = \sum_{\ell,\ell' \in [1,L]} \mathcal{B}^*_{\ell,\ell'}\,, \tag{20}$$

where $\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*[\ell,\ell'])$ denotes the $(\ell,\ell')^{\text{th}}$ cross diagonal block of the Hessian and where L denotes the network depth. So for an arbitrary set of layer indices $\mathcal{P}$, which we will use to analyze the cumulative barriers considered in [1], we have the expression:

$$\mathcal{B}^*_{\mathcal{P}} \approx \frac{1}{8}\sum_{\ell,\ell' \in \mathcal{P}} \Delta\boldsymbol{\theta}^*[\ell]^\top\ \cdot\ \nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_1^*[\ell,\ell'])\ \cdot\ \Delta\boldsymbol{\theta}^*[\ell']\,, \tag{21}$$

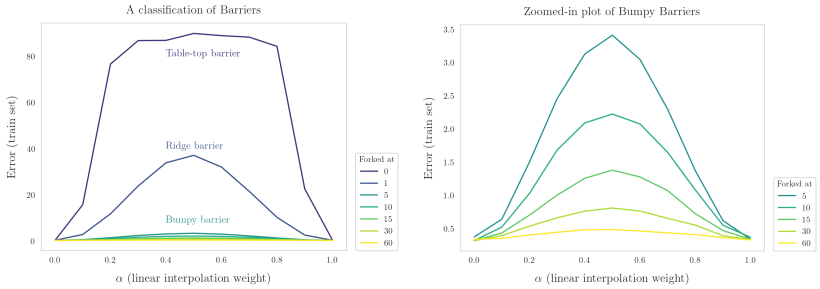
Appendix B. Additional Results and Details

B.1. Hyperparameter Setup

Unless stated otherwise, following Frankle et al. [6], we consider a ResNet20 trained on CIFAR10 with batch normalization enabled for 160 epochs with SGD. The other hyperparameters that were used are a learning rate 0.1 which is decreased by a factor of 10 at epochs 80 and 120. Besides, other hyperparameters are weight decay 0.0001, batch size 128, momentum 0.9.

B.2. Topography of LMC

B.2.1. 1D FINAL BARRIER VIEW



(*a*) The 3 kinds of barriers (*b*) Bumpy barriers when zoomed in

Figure 7: A one-dimensional summary of the barrier types and their classification.

B.2.2. TRAINING ERROR EVOLUTION FOR OTHER SETTINGS

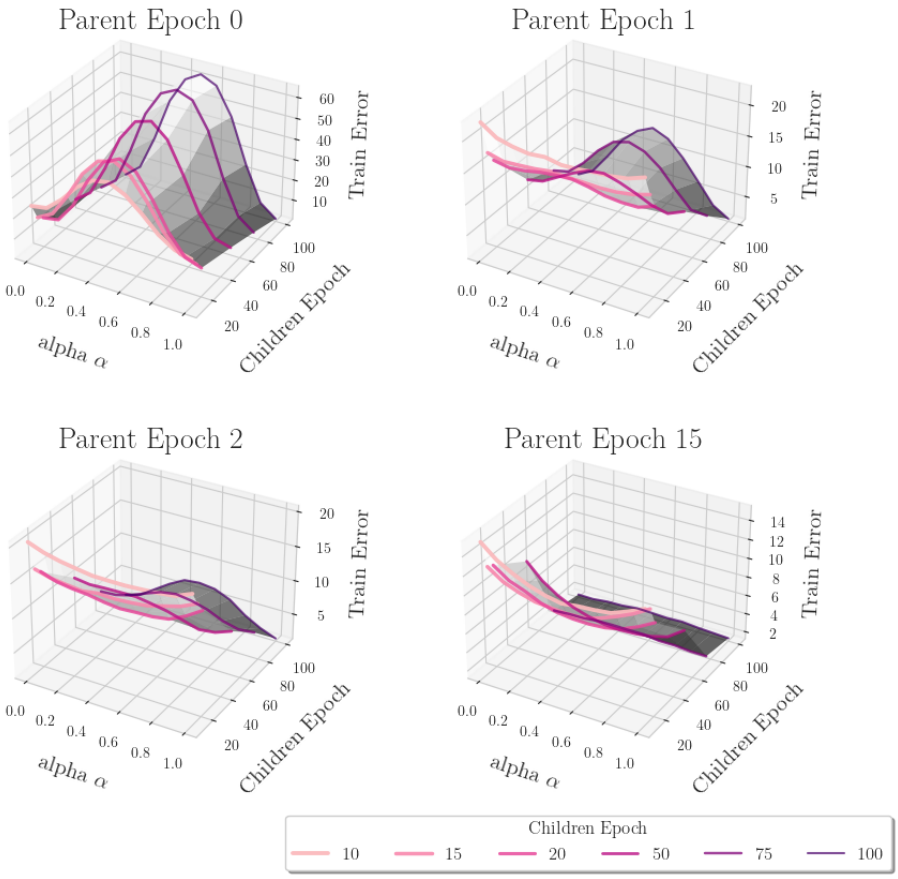


Figure 8: LR $\eta = 0.01$, Weight decay $\lambda = 0.0001$: The evolution of train error curves, with train error, in 3d when forked at different points.

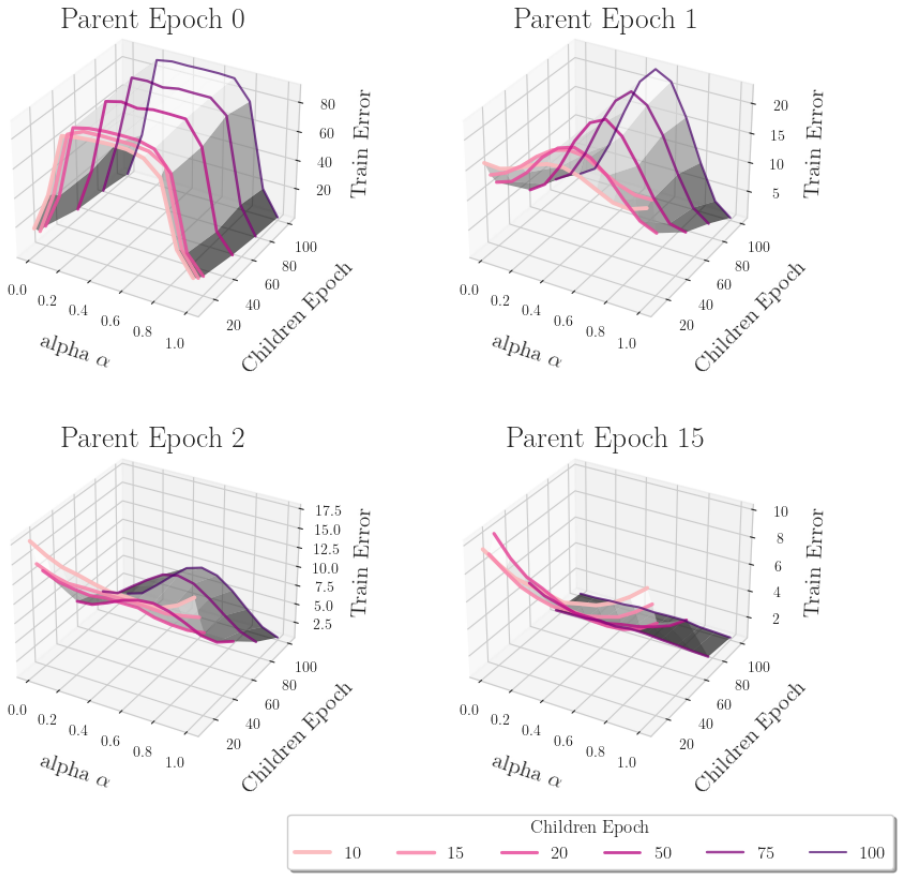


Figure 9: Weight decay $\lambda = 0$, LR $\eta = 0.1$: The evolution of train error curves, with train error, in 3d when forked at different points.

B.2.3. BARRIER CURVES EVOLUTION

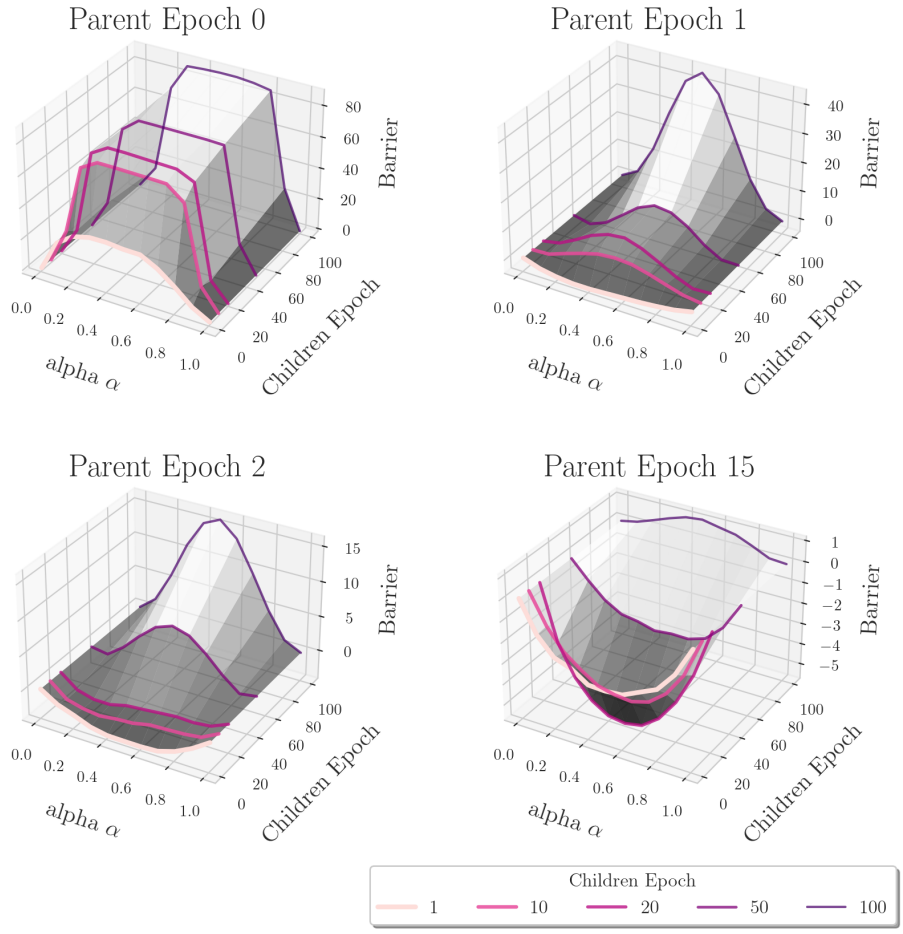


Figure 10: Weight decay $\lambda = 0.0001$, LR $\eta = 0.1$: The evolution of LMC barrier curves, with train error, in 3d when forked at different points.

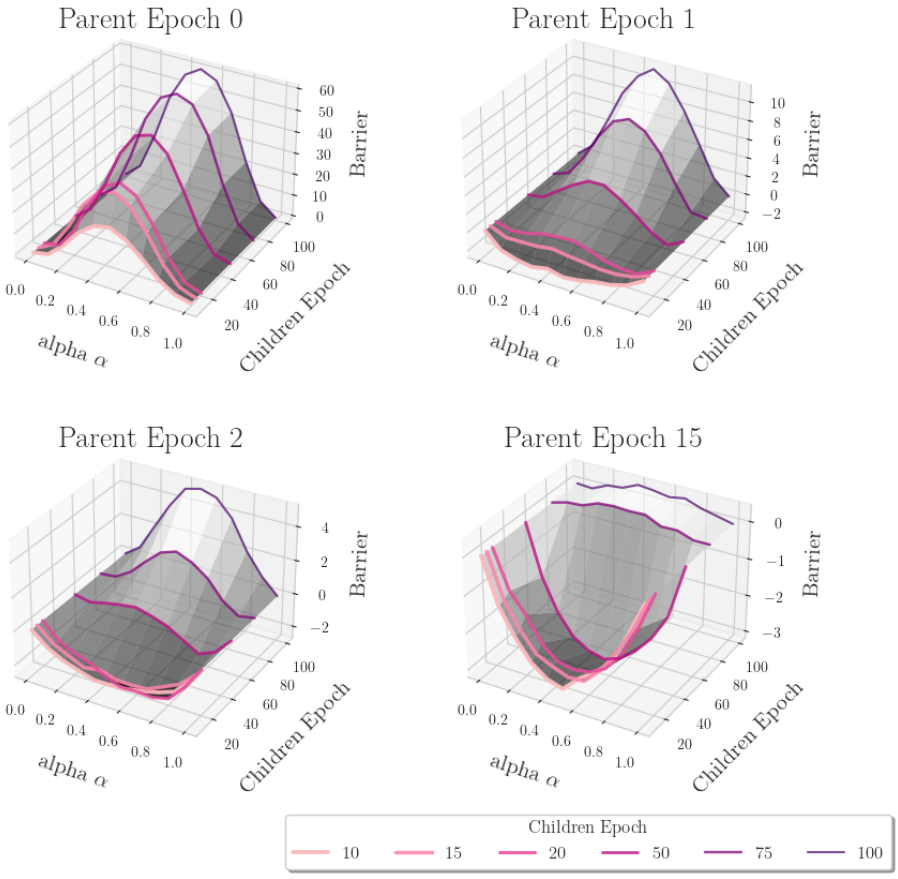


Figure 11: Weight decay $\lambda = 0.0001$, LR $\eta = 0.01$: The evolution of LMC barrier curves, with train error, in 3d when forked at different points.

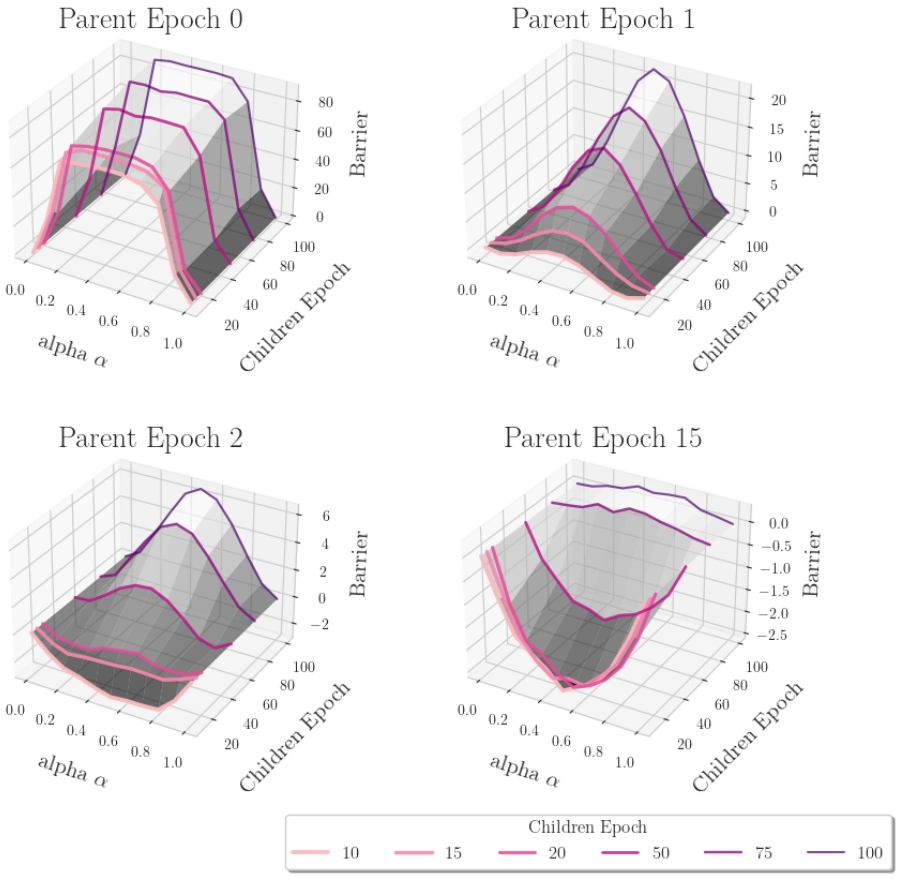
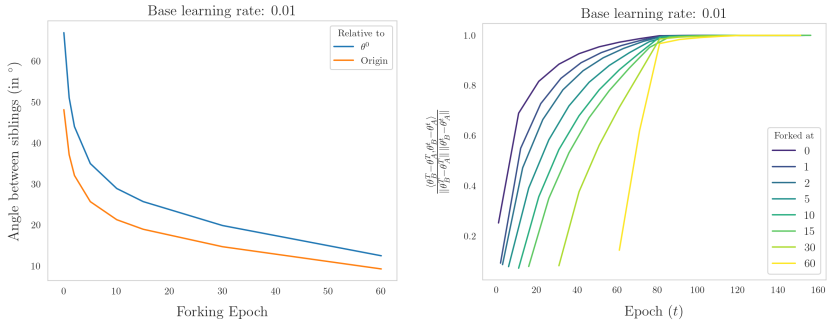


Figure 12: Weight decay $\lambda = 0$, LR $\eta = 0.1$: The evolution of LMC barrier curves, with train error, in 3d when forked at different points.

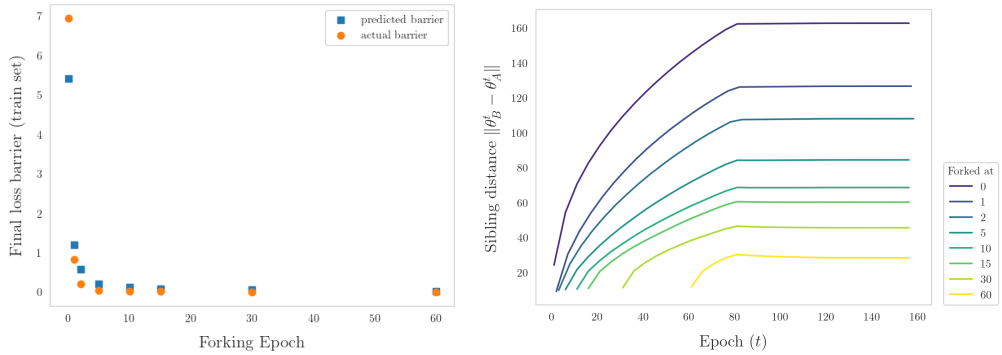
B.2.4. SIBLING GEOMETRY ANALYSIS



(a) Sibling angles, $\eta = 0.01$ (b) Sibling solution planes, $\eta = 0.01$

Figure 13: The angle between sibling solutions (in degrees) as well as the determination of sibling solution planes for different forks, for a small learning rate scenario ($\eta = 0.01$).

B.3. Barrier Prediction Analysis



(a) Predicted vs Actual Barriers (b) Sibling Distances

Figure 14: Without weight decay. *Left:* Barrier predictions versus actual barriers over different forking epochs. *Right:* The evolution of the distance between spawned models is plotted, and where the different lines denote different forking points.

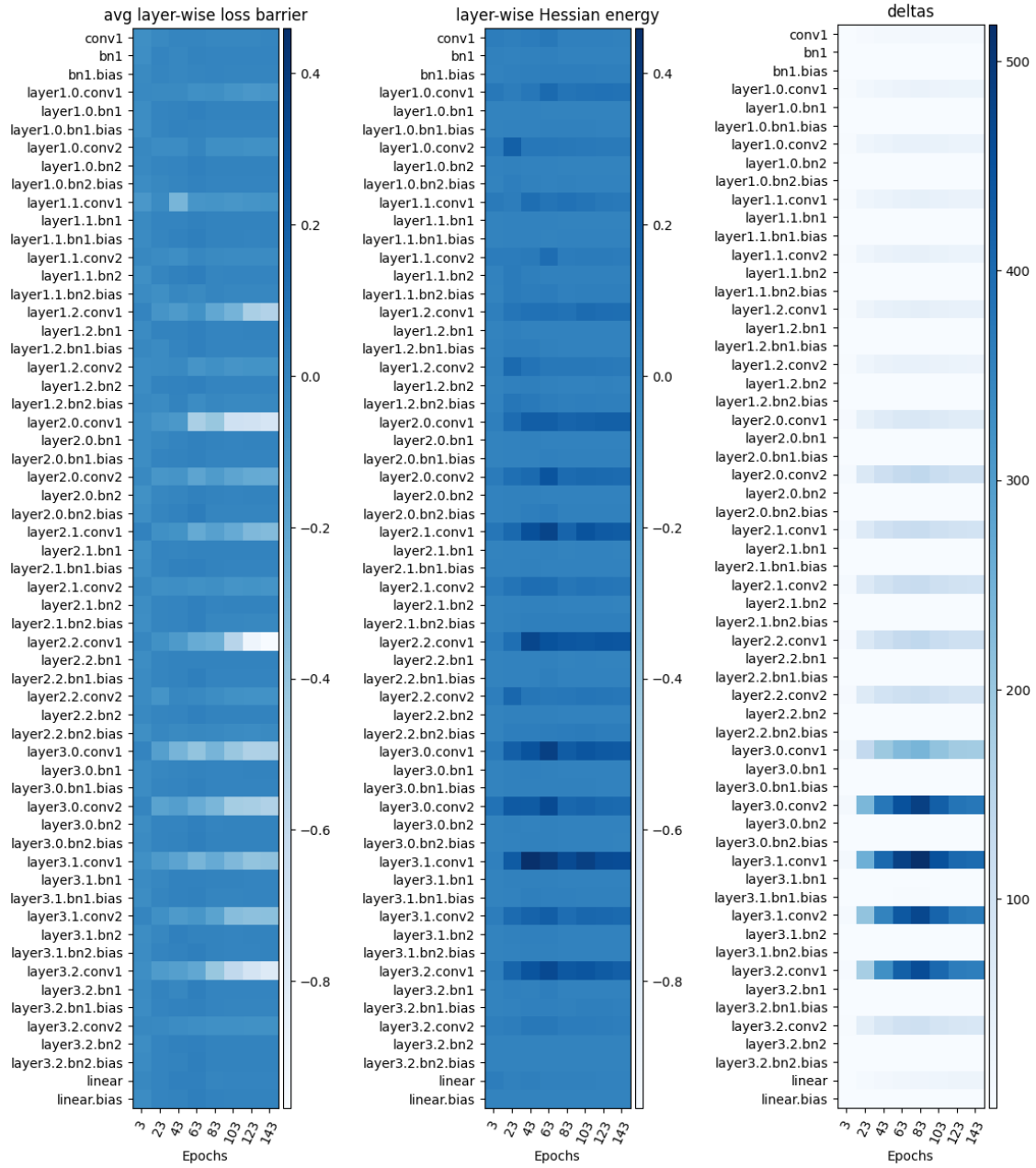
B.4. Layer-wise LMC results

B.4.1. EXPERIMENTAL DETAILS FOR EXPERIMENTS IN THE MAIN TEXT

The presented results consider ResNet18 on CIFAR10, without batch normalization layers, with the training hyperparameters being: learning rate 0.05 and batch size 64. The full Hessian is computed over the entire training set. While in this experiment Batch Normalization was disabled, in the experiments below, we also consider Batch Normalization.

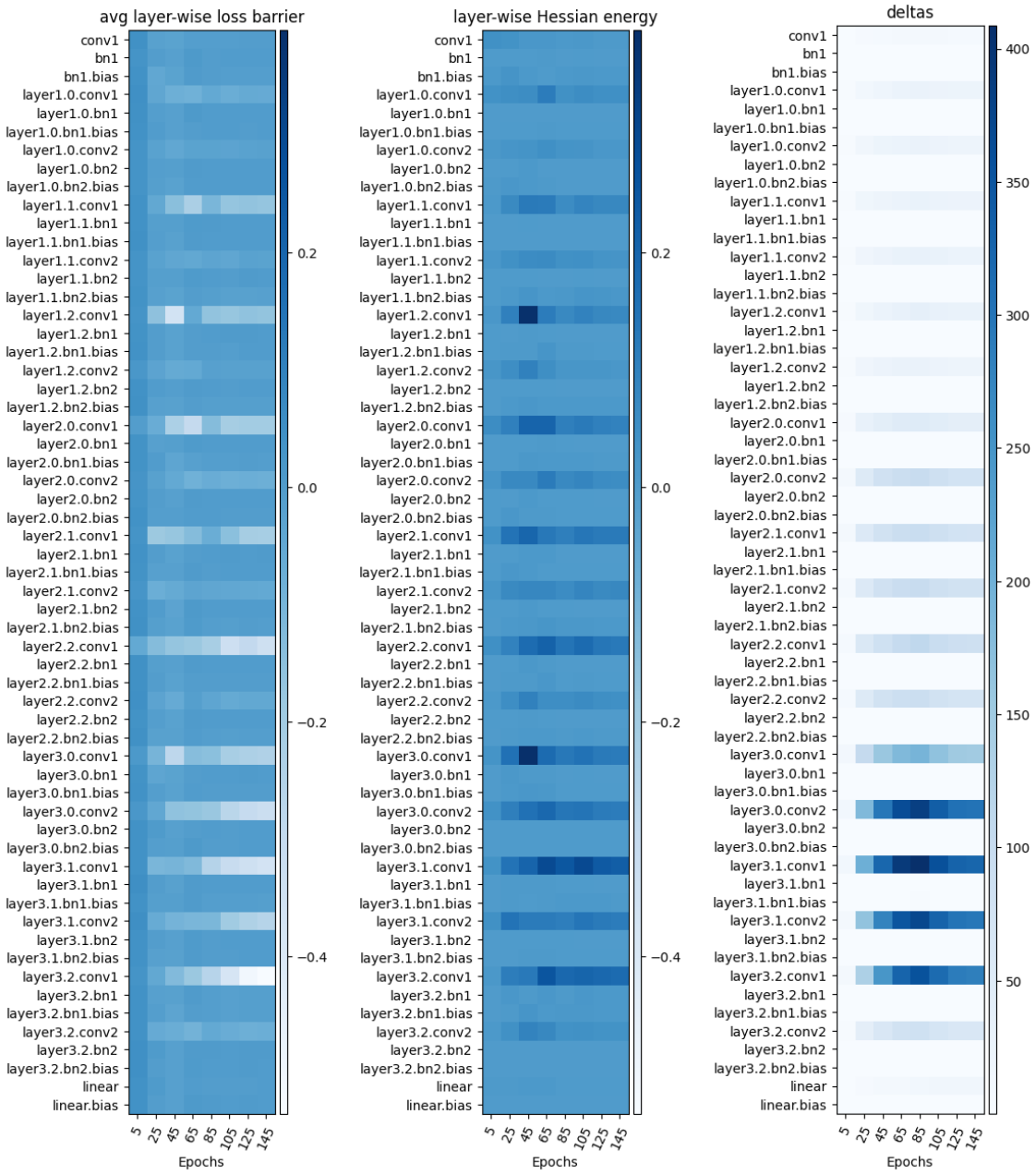
B.4.2. ADDITIONAL RESULTS

For these experiments, we use the hyperparameter detailed in Section B.1.



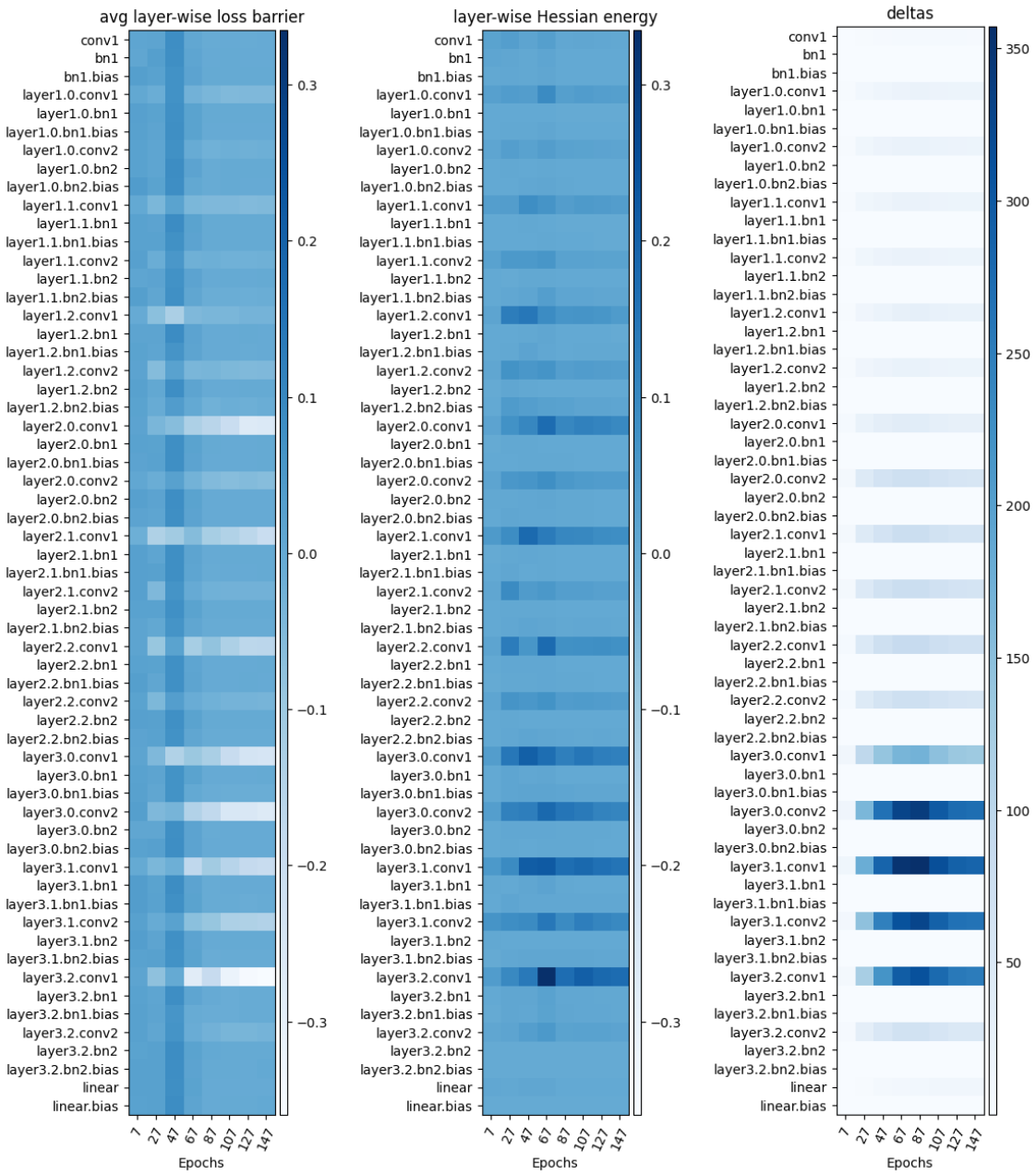
(*a*) layer-wise Loss barrier (*b*) layer-wise Hessian energy (*c*) deltas

Figure 15: (Left) The layer-wise loss barrier as per [1]. **(Middle):** Layer-wise Hessian Energy, computed as per Proposition 2 during the course of training. Forked on epoch 2. **(Right):** Distance between layers.



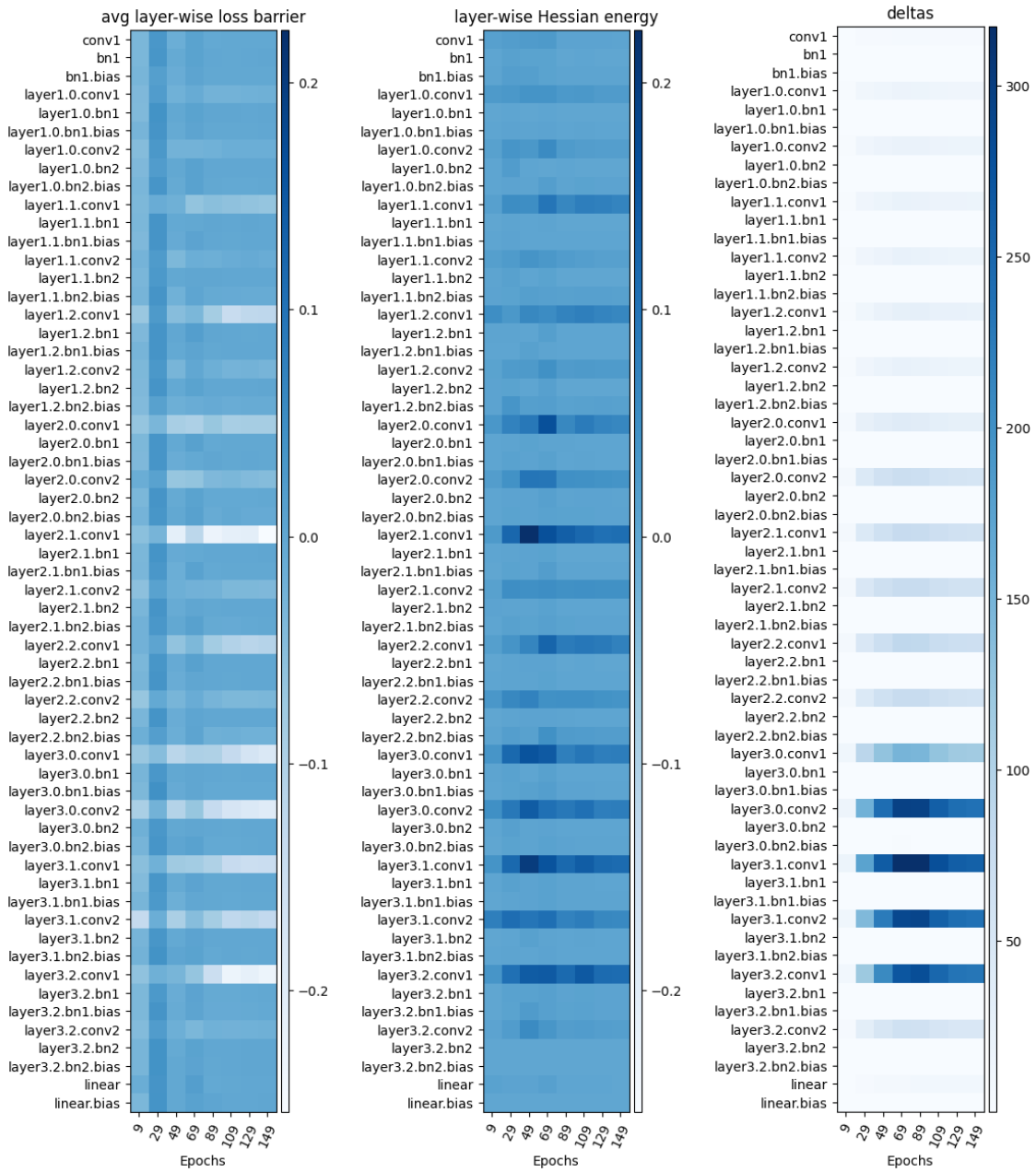
(*a*) layer-wise Loss barrier (*b*) layer-wise Hessian energy (*c*) deltas

Figure 16: (Left) The layer-wise loss barrier as per [1]. **(Middle):** Layer-wise Hessian Energy, computed as per Proposition 2 during the course of training. Forked on epoch 4. **(Right):** Distance between layers.



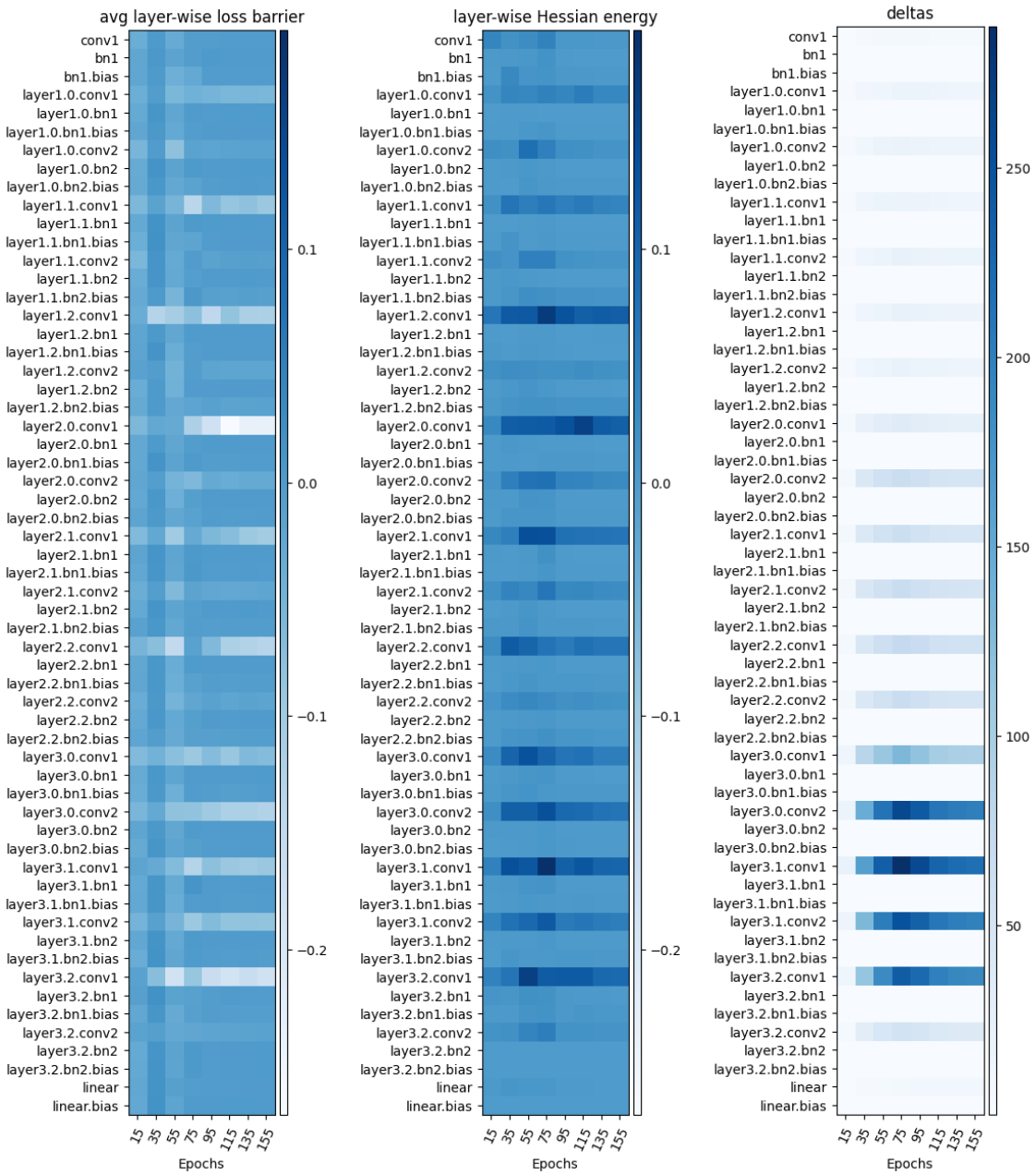
(*a*) layer-wise Loss barrier (*b*) layer-wise Hessian energy (*c*) deltas

Figure 17: (Left) The layer-wise loss barrier as per [1]. **(Middle):** Layer-wise Hessian Energy, computed as per Proposition 2 during the course of training. Forked on epoch 6. **(Right):** Distance between layers.



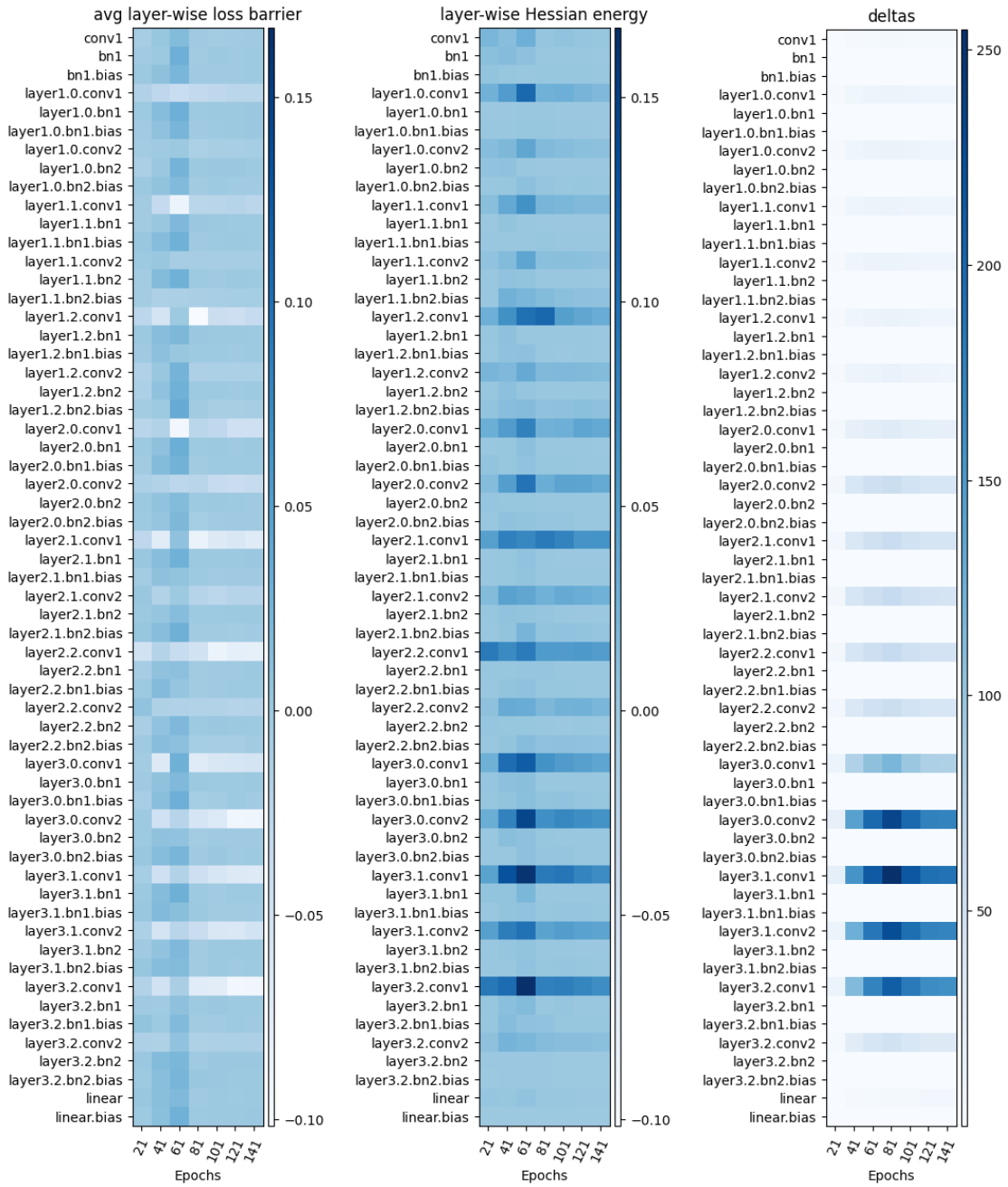
(*a*) layer-wise Loss barrier (*b*) layer-wise Hessian energy (*c*) deltas

Figure 18: (Left) The layer-wise loss barrier as per [1]. **(Middle):** Layer-wise Hessian Energy, computed as per Proposition 2 during the course of training. Forked on epoch 8. **(Right):** Distance between layers.



(*a*) layer-wise Loss barrier (*b*) layer-wise Hessian energy (*c*) deltas

Figure 19: (Left) The layer-wise loss barrier as per [1]. **(Middle):** Layer-wise Hessian Energy, computed as per Proposition 2 during the course of training. Forked on epoch 14. **(Right):** Distance between layers.



(*a*) layer-wise Loss barrier (*b*) layer-wise Hessian energy (*c*) deltas

Figure 20: (Left) The layer-wise loss barrier as per [1]. **(Middle):** Layer-wise Hessian Energy, computed as per Proposition 2 during the course of training. Forked on epoch 20. **(Right):** Distance between layers.