CONTROLLING CHANGES TO ATTENTION LOGITS

Anonymous authors
Paper under double-blind review

ABSTRACT

Stability of neural network weights is critical when training transformer models. The query and key weights are particularly problematic, as they tend to grow large without any intervention. Applying normalization to queries and keys, known as 'QK norm', fixes stability issues in practice, but is not always applicable. For example, QK norm is not compatible with Multi-head Latent Attention (MLA) because QK norm requires full materialization of queries and keys during inference, which is not done in MLA. In this paper we hypothesize that instability is driven primarily by changes in attention logits, rather than by their absolute magnitude, and that controlling these changes is sufficient for stability. We show that these changes are controllable by assigning parameter-dependent learning rates to the query and key weights. Our cheap intervention allows us to increase the base learning rate of the network, outperform other methods in the MLA setting, and achieve performance competitive with QK norm when using Multi-head Attention.

1 INTRODUCTION

Principled scaling of transformer models is crucial for efficiently training larger and more capable architectures. Maximal Update Parametrization (μP) (Yang et al., 2022) has emerged as a key technique in this area, enabling the transfer of optimal hyperparameters from smaller to larger models by carefully parameterizing the model. A core desideratum of μP is to control the magnitude of activations and their updates (Dey et al., 2025), ensuring consistent training dynamics across different model widths. Regarding attention, Yang et al. (2022) addresses attention logits blowing up as we increase model width by proposing a static attention scaling factor. While this static scaling helps control logit magnitude across different model widths, it does not address logit changes during longer training runs, which can become a major source of instability, particularly at high learning rates.

Attention logits are a well-known source of training instability (Zhai et al., 2023; Bai et al., 2025), and we illustrate this in Figure 1. Several interventions have emerged such as QK norm (Henry et al., 2020) and QK clip/MuonClip (Bai et al., 2025) to ensure their stability. While QK norm is especially effective, it is ill-suited for Multi-head Latent Attention (MLA) (Liu et al., 2024), as queries and keys are not fully materialized at inference-time for efficiency reasons (Bai et al., 2025). Other methods like QK clip require a bespoke attention mechanism to track maximum attention logits, which can complicate integration into existing codebases.

We hypothesize that instability of attention is driven mostly by large changes to logits, rather than by large logits themselves. This is motivated by the observation that the quadratic nature of attention means that gradients for the logits (and queries and keys) depend on the size of activations. Thus we can encounter unstable dynamics where large activations lead to large changes, which lead to large activations, etc.. We propose a simple method to resolve this, that modulates learning rates of query and key weight matrices, and we validate that this removes training instabilities. Specifically, our results demonstrate that our method is as stable as QK norm, particularly at high base learning rates. While not quite reaching the same peak performance as QK norm in the standard Multi-Head Attention (MHA) setting, our method is computationally cheaper and is applicable to MLA. When used with MLA our method enables higher base learning rates, and outperforms QK clip, highlighting potential benefits when training modern, efficient transformer architectures.

In summary, our contributions are as follows,

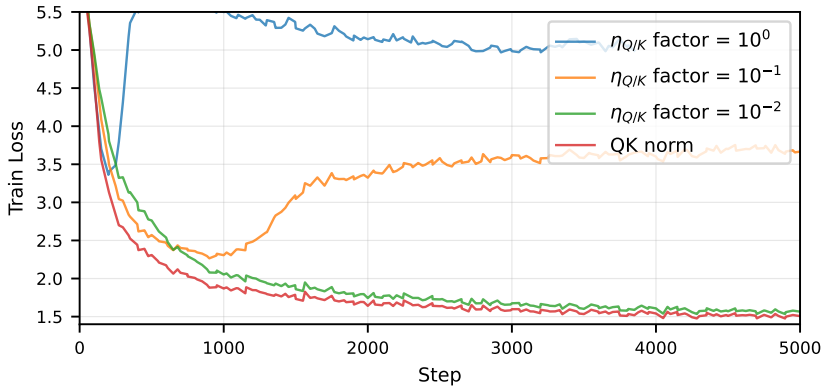

Figure 1: **Learning rate of query/key matrices is a critical factor for transformer pretraining stability.** 4 models are trained with a large base learning rate of $\eta = 3\text{e} - 2$ for each parameter. Decreasing the learning rates of query and key weights ($\eta_{Q/K}$) alone, fully stabilizes pretraining. QK norm is shown to illustrate a stable baseline.

- We propose that changes to attention logits are a key quantity to track for stability in attention.
- We show that we can control logit changes by modulating the learning rate of query weights based on the norms of corresponding key weights, and vice versa, dynamically at training time.
- We demonstrate that this learning-rate intervention yields stable training and competitive validation loss, outperforming alternatives in the MLA setting.

2 Related Work

Controlling attention logits. Training instabilities are often encountered in the attention layer itself. Attention logits may become large (Bai et al., 2025), potentially inducing collapse in attention entropy (Zhai et al., 2023), where attention distributions become highly concentrated. QK normalization (Henry et al., 2020), which applies normalization to query and key activations, has emerged as a simple and effective remedy, preventing large logits (Dehghani et al., 2023) and allowing larger learning rates (Wortsman et al., 2023). Similar methods such as logit soft-capping apply normalization to logits directly (Bello et al., 2016; Riviere et al., 2024). Other methods normalize the weights rather than activations: σReparam (Zhai et al., 2023) parameterizes weights into a matrix and a scalar component, with the matrix having unit spectral norm and a scalar that captures overall scale; QK clip (Bai et al., 2025) controls attention logits by clipping weights whenever the logits grow beyond a certain threshold.

Parameter-specific learning rates. While it is common to share the same learning rate across all parameters in a neural network, parameter-specific learning rates have been extensively examined (Milsom et al., 2025; You et al., 2017; Liu et al., 2019; Xu et al., 2019; Wang et al., 2025; Bernstein et al., 2020; Qi et al., 2025; Yang et al., 2023). Proposals often include adjusting the learning rate of a parameter according to the norm of step/gradient (Yang et al., 2023; Liu et al., 2019), as well as the parameter itself (Qi et al., 2025), such as LARS, LAMB, and Fromage (Bernstein et al., 2020; You et al., 2017; 2019).

Our work selects parameter-specific learning rates that control changes to attention logits. However, by considering attention logits as a whole, our parameter-specific learning rates are 'inter-parameter', unlike other methods, such as LARS, which consider each parameter tensor independently. Our method is also inspired by μP (Yang et al., 2022; Dey et al., 2025); in μP, one of the desiderata is that as we make changes to our parameters in a network, the residual stream should correspondingly change in a controlled, 'order 1-like' manner. Our work extends this notion to logits.

Algorithm 1 QuacK (MHA)

Require: Hyperparameter τ, base learning rate η

Make the following additions to the transformer training script:

```
# At initialization. Calculate initial norms for query/key weights for all heads
for all layers ℓ do
    for all heads h do
        W_Q^{ℓ,h}.init_norm ← ‖W_Q^{ℓ,h}‖
        W_K^{ℓ,h}.init_norm ← ‖W_K^{ℓ,h}‖
    end for
end for

# During training. Prior to each optimization step, adjust learning rates
for all layers ℓ do
    for all heads h do
        W_Q^{ℓ,h}.lr ← τ η · W_K^{ℓ,h}.init_norm / ‖W_K^{ℓ,h}‖
        W_K^{ℓ,h}.lr ← τ η · W_Q^{ℓ,h}.init_norm / ‖W_Q^{ℓ,h}‖
    end for
end for
```

3 METHODS

Unlike other transformer modules, attention has quadratic structure. In particular, the attention logits are given by,

$$\mathbf{L} = d^{-1/2}\mathbf{Q}\mathbf{K}^T = d^{-1/2}\mathbf{X}\mathbf{W}_Q\mathbf{W}_K^T\mathbf{X}^T. \tag{1}$$

We attempt to keep the changes to attention logits, $\Delta\mathbf{L}$, under control. By a first-order analysis, we see that if the queries are large, then perturbations due to the keys will be amplified, and vice versa:

$$\Delta\mathbf{L} = \frac{(\mathbf{Q}+\Delta\mathbf{Q})(\mathbf{K}+\Delta\mathbf{K})^T - \mathbf{Q}\mathbf{K}^T}{\sqrt{d}} \approx \frac{\mathbf{Q}(\Delta\mathbf{K})^T + (\Delta\mathbf{Q})\mathbf{K}^T}{\sqrt{d}}. \tag{2}$$

The main tool we have for controlling changes is the learning rate. Thus we propose to set the learning rates η_Q, η_K (for $\mathbf{W}_Q$, and $\mathbf{W}_K$ respectively) such that $\mathbf{Q}(\Delta\mathbf{K})^T$ and $(\Delta\mathbf{Q})\mathbf{K}^T$ are both 'order 1'. We formalize this notion in Lemma 1.

Following the Lemma we set,

$$\eta_Q \propto \|\mathbf{W}_K\|^{-1}, \quad \eta_K \propto \|\mathbf{W}_Q\|^{-1}. \tag{3}$$

In practice we treat the constant of proportionality in Eq. (3) as a hyperparameter: at initialization, we set the learning rate for each query and key weight to be equal to $\tau\eta$ and we tune τ. Thus τ acts as a relative initial learning rate (relative to η, the base learning rate). This allows for clear comparison to other methods in the experiments.

The above methodology applies to both the single- and multi-head (MHA) setting. In MHA, each head has its own query and key weight, so we apply Eq. (3) to each head separately. We summarize the resulting method in Algorithm 1.

We extend to MLA using a similar approach in Appendix C. A notable difference between MHA and MLA is that there are several more parameter matrices to consider; bounding the change in logits requires us to adjust the learning rate of each of these parameters. We detail exactly how to set the learning rates for MLA in Algorithm 2.

4 EXPERIMENTS

To evaluate the different approaches, we trained ∼1B-sized transformer LMs (Qwen3-based) with both MHA (Vaswani et al., 2017) and MLA (Liu et al., 2024). Runs used the same optimizer

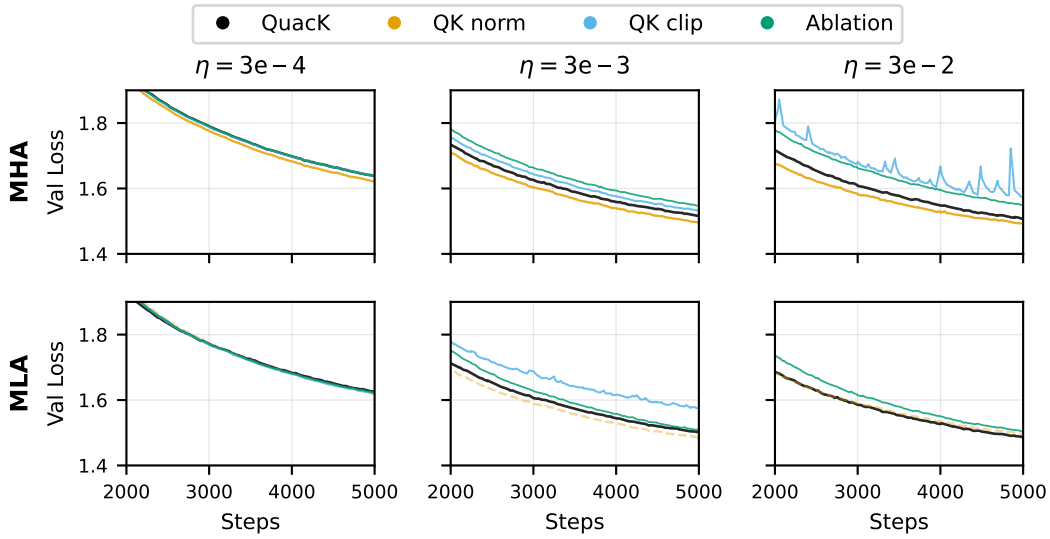

Figure 2: Validation losses when training each method with $\texttt{attn} \in \{\texttt{MHA}, \texttt{MLA}\}$, and learning rates, $\eta \in \{3e-4, 3e-3, 3e-2\}$. QK clip is unstable at high learning rates (omitted from the bottom right plot due to loss $\gg 2$). QK norm is overall the most performant, but it is not appropriate for use with MLA at inference-time for efficiency reasons (illustrated via dashed yellow line in the MLA row). QuacK is a sensible alternative, as it is stable in the high LR setting, performant, and applicable in the MLA setting.

(Muon (Jordan et al., 2024)) and data pipeline (Cosmopedia-V2 / SmolLM-corpus (Ben Allal et al., 2024)); details of model hyperparameters and compute setup are provided in the Appendix.

Our experiments varied attention type (MHA / MLA), base learning rate $\eta \in \{3e-4, 3e-3, 3e-2\}$, and the stabilization method: QK norm (Henry et al., 2020); QK clip (Bai et al., 2025); fixed LR scaling for Q/K weights (ablation), and QuacK (ours).

Higher learning rates are better. Figure 2, left column, shows that at the low learning rate of $\eta = 3e-4$, all logit interventions perform similarly, but with QK norm performing marginally better. The lack of variety in performance is likely due to the fact the learning rate is small enough that we don't encounter instabilities. However, performance is much improved by increasing the learning rate (column 2, 3).

QuacK maintains stability and strong performance, enabling higher base learning rates. QK clip is insufficient to prevent instabilities at the highest base learning rate of $\eta = 3e-2$ (column 3, Figure 2), and it underperforms, especially in the MLA setting, when $\eta = 3e-3$ (column 2). The ablation, which sets the learning rates for query and key weights to smaller fixed values, is stable, but underperforms QuacK in both the MHA and MLA settings. QuacK has similar but slightly worse performance compared to QK norm in the MHA setting, but is the best performing method in the MLA setting (we include QK norm results in the MLA setting for comparison, but it is not viable at inference-time like the other methods).

5 Conclusion

Our method, QuacK, stabilizes training by controlling changes in attention logits via coupled learning rates for query/key weights. In ∼1B-scale pretraining, QuacK enables higher base learning rates and improves performance over QK clip in the MLA setting (details and additional plots in the appendix).

REFERENCES

Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, et al. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*, 2025.

Irwan Bello, Hieu Pham, Quoc V Le, Mohammad Norouzi, and Samy Bengio. Neural combinatorial optimization with reinforcement learning. *arXiv preprint arXiv:1611.09940*, 2016.

Loubna Ben Allal, Anton Lozhkov, Guilherme Penedo, Thomas Wolf, and Leandro von Werra. Smollm-corpus, 2024. URL `https://huggingface.co/datasets/HuggingFaceTB/smollm-corpus`.

Jeremy Bernstein, Arash Vahdat, Yisong Yue, and Ming-Yu Liu. On the distance between two neural networks and the stability of learning. *Advances in Neural Information Processing Systems*, 33: 21370–21381, 2020.

Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *International conference on machine learning*, pp. 7480–7512. PMLR, 2023.

Nolan Dey, Bin Claire Zhang, Lorenzo Noci, Mufan Li, Blake Bordelon, Shane Bergsma, Cengiz Pehlevan, Boris Hanin, and Joel Hestness. Don't be lazy: Completep enables compute-efficient deep transformers. *arXiv preprint arXiv:2505.01618*, 2025.

Alex Henry, Prudhvi Raj Dachapally, Shubham Pawar, and Yuxuan Chen. Query-key normalization for transformers. *arXiv preprint arXiv:2010.04245*, 2020.

Tao Ji, Bin Guo, Yuanbin Wu, Qipeng Guo, Lixing Shen, Zhan Chen, Xipeng Qiu, Qi Zhang, and Tao Gui. Towards economical inference: Enabling deepseek's multi-head latent attention in any transformer-based llms. *arXiv preprint arXiv:2502.14837*, 2025.

Keller Jordan, Yuchen Jin, Vlado Boza, Jiacheng You, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks, 2024. URL `https://kellerjordan.github.io/posts/muon/`.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265*, 2019.

Edward Milsom, Ben Anson, and Laurence Aitchison. Function-space learning rates. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

Xianbiao Qi, Yelin He, Jiaquan Ye, Chun-Guang Li, Bojia Zi, Xili Dai, Qin Zou, and Rong Xiao. Taming transformer without using learning rate warmup. *arXiv preprint arXiv:2505.21910*, 2025.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: enhanced transformer with rotary position embedding. arxiv. *arXiv preprint arXiv:2104.09864*, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing syStems*, 30, 2017.

Jinbo Wang, Mingze Wang, Zhanpeng Zhou, Junchi Yan, Lei Wu, et al. The sharpness disparity principle in transformers for accelerating language model pre-training. *arXiv preprint arXiv:2502.19002*, 2025.

Mitchell Wortsman, Peter J Liu, Lechao Xiao, Katie Everett, Alex Alemi, Ben Adlam, John D Co-Reyes, Izzeddin Gur, Abhishek Kumar, Roman Novak, et al. Small-scale proxies for large-scale transformer training instabilities. *arXiv preprint arXiv:2309.14322*, 2023.

Zhen Xu, Andrew M Dai, Jonas Kemp, and Luke Metz. Learning an adaptive learning rate schedule. *arXiv preprint arXiv:1909.09712*, 2019.

Greg Yang, Edward J Hu, Igor Babuschkin, Szymon Sidor, Xiaodong Liu, David Farhi, Nick Ryder, Jakub Pachocki, Weizhu Chen, and Jianfeng Gao. Tensor programs v: Tuning large neural networks via zero-shot hyperparameter transfer. *arXiv preprint arXiv:2203.03466*, 2022.

Greg Yang, James B Simon, and Jeremy Bernstein. A spectral condition for feature learning. *arXiv preprint arXiv:2310.17813*, 2023.

Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large batch optimization for deep learning: Training bert in 76 minutes. *arXiv preprint arXiv:1904.00962*, 2019.

Shuangfei Zhai, Tatiana Likhomanenko, Etai Littwin, Dan Busbridge, Jason Ramapuram, Yizhe Zhang, Jiatao Gu, and Joshua M Susskind. Stabilizing transformer training by preventing attention entropy collapse. In *International Conference on Machine Learning*, pp. 40770–40803. PMLR, 2023.

A Experimental setup details

All models used $d_{\text{model}} = 2048$, $d_{\text{ff}} = 4d_{\text{model}}$, $n_{\text{head}} = 32$, $n_{\text{layer}} = 14$, and were trained for 5000 steps at context length 2048 with 96 sequences per batch. We used the GPT-2 tokenizer (Radford et al., 2019) (vocab size 49,152) with embedding/unembedding weight tying, and trained on Cosmopedia-V2 / SmolLM-corpus (Ben Allal et al., 2024) using Muon (Jordan et al., 2024) with constant LR and 500 warmup steps. MHA used $d_{\text{head}} = 64$; MLA used $d_{\text{head}} = 128$ (with $d_{\text{rope}} = d_{\text{nope}} = 64$), latent dims $d_{\text{cq}} = 512$ and $d_{\text{ckv}} = 256$.

Lemma 1. *Let $\mathbf{W}_Q, \mathbf{W}_K \in \mathbb{R}^{d_{model} \times d_{head}}$ be weight matrices corresponding to a particular attention head, and consider the worst-case change in logits, for unit normed input,*

$$\max_{\|x\|_2=\|y\|_2=1} |\Delta\ell| := \max_{\|x\|_2=\|y\|_2=1} |x^\top(\mathbf{W}+\Delta\mathbf{W})y - x^\top\mathbf{W}y|,$$

where $\mathbf{W} = d_{head}^{-1/2}\mathbf{W}_Q^\top\mathbf{W}_K$. Suppose that the steps for $\mathbf{W}_Q$ and $\mathbf{W}_K$ are given by $\Delta\mathbf{W}_{Q/K} = -\eta_{Q/K}\mathbf{G}_{Q/K}$, where $\|\mathbf{G}_{Q/K}\| \leq D$ for some constant D (which is the case for Adam and Muon). If there is a constant c such that $0 < c \leq \|\mathbf{W}_Q\|, \|\mathbf{W}_K\|$, and the learning rates satisfy $\eta_Q \propto \|\mathbf{W}_K\|^{-1}$, and $\eta_K \propto \|\mathbf{W}_Q\|^{-1}$, then the worst-case change in logits is bounded above independently of the norm of the weights.

B Proof of Lemma 1

Lemma 1. *Let $\mathbf{W}_Q, \mathbf{W}_K \in \mathbb{R}^{d_{model} \times d_{head}}$ be weight matrices corresponding to a particular attention head, and consider the worst-case change in logits, for unit normed input,*

$$\max_{\|x\|_2=\|y\|_2=1} |\Delta\ell| := \max_{\|x\|_2=\|y\|_2=1} |x^\top(\mathbf{W}+\Delta\mathbf{W})y - x^\top\mathbf{W}y|,$$

where $\mathbf{W} = d_{head}^{-1/2}\mathbf{W}_Q^\top\mathbf{W}_K$. Suppose that the steps for $\mathbf{W}_Q$ and $\mathbf{W}_K$ are given by $\Delta\mathbf{W}_{Q/K} = -\eta_{Q/K}\mathbf{G}_{Q/K}$, where $\|\mathbf{G}_{Q/K}\| \leq D$ for some constant D (which is the case for Adam and

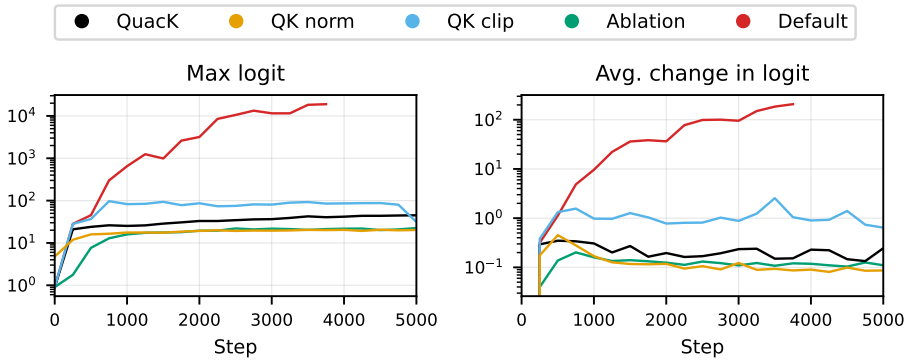

Figure 3: Max logit (left) and average absolute change in logit throughout training (right) with a base learning rate of $\eta = 3\mathrm{e} - 3$. Here we show the middle head of the middle layer (head 16 and layer 8) while training with MLA.

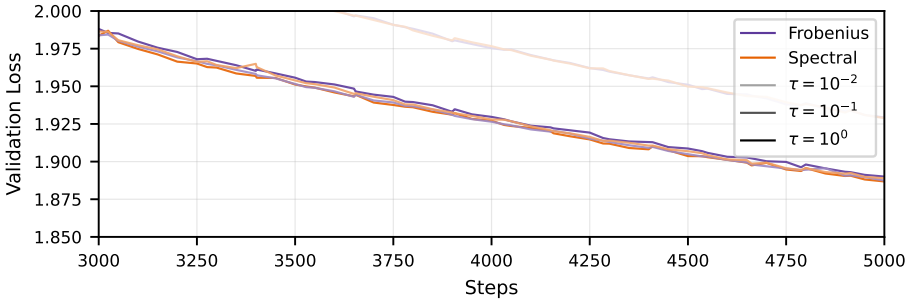

Figure 4: Performance differences when applying Algorithm 1 with different norms are small. We show validation losses when training a small model ($\sim$ 100M parameters) with Algorithm 1 to modulate the query and key weight learning rates. Different curves show results with different values of the hyperparameter τ and measuring the query and key weights with either Frobenius or spectral norm.

Muon). If there is a constant c such that $0 < c \leq \|\mathbf{W}_Q\|, \|\mathbf{W}_K\|$, and the learning rates satisfy $\eta_Q \propto \|\mathbf{W}_K\|^{-1}$, and $\eta_K \propto \|\mathbf{W}_Q\|^{-1}$, then the worst-case change in logits is bounded above independently of the norm of the weights.

Proof. The change in logits is given by,

$$\begin{aligned} d_{\text{head}}^{1/2}|\Delta\ell| &= |(q+\Delta q)^T(k+\Delta k) - q^T k| \\ &= |(\Delta q)^T k + q^T \Delta k + (\Delta q)^T \Delta k| \\ &\leq |(\Delta q)^T k| + |q^T \Delta k| + \|\Delta q\|\|\Delta k\|. \end{aligned} \tag{4}$$

where,

$$q = \mathbf{W}_Q x,\ k = \mathbf{W}_K y \tag{5}$$

The query and key perturbations are given by,

$$\Delta q = (\mathbf{W}_Q + \Delta\mathbf{W}_Q)x - \mathbf{W}_Q x = \Delta\mathbf{W}_Q x, \tag{6}$$

$$\Delta k = (\mathbf{W}_K + \Delta\mathbf{W}_K)y - \mathbf{W}_K y = \Delta\mathbf{W}_K y. \tag{7}$$

We now bound the first order terms in Eq. (4), assuming inputs are unit normed,

$$\begin{aligned} |(\Delta q)^T k| &= |(\Delta \mathbf{W}_Q x)^T (\mathbf{W}_K y)| = |x^T \Delta \mathbf{W}_Q^T \mathbf{W}_K y| \\ &\leq \|x\| \|\Delta \mathbf{W}_Q^T \mathbf{W}_K\| \|y\| \\ &\leq \eta_Q D \|\mathbf{W}_K\|, \end{aligned} \tag{8a}$$

$$\begin{aligned} |q^T \Delta k| &= |(\mathbf{W}_Q x)^T (\Delta \mathbf{W}_K y)| = |x^T \mathbf{W}_Q^T \Delta \mathbf{W}_K y| \\ &\leq \|x\| \|\mathbf{W}_Q^T \Delta \mathbf{W}_K\| \|y\| \\ &\leq \eta_K D \|\mathbf{W}_Q\|. \end{aligned} \tag{8b}$$

For some constants τ_Q, τ_K, set,

$$\eta_Q = \tau_Q \|\mathbf{W}_K\|^{-1} \tag{9a}$$

$$\eta_K = \tau_K \|\mathbf{W}_Q\|^{-1}. \tag{9b}$$

Substituting these into Eqs. (8), we obtain the bounds,

$$|(\Delta q)^T k| \leq (\tau_Q \|\mathbf{W}_K\|^{-1}) D \|\mathbf{W}_K\| = \tau_Q D, \tag{10}$$

$$|q^T \Delta k| \leq (\tau_K \|\mathbf{W}_Q\|^{-1}) D \|\mathbf{W}_Q\| = \tau_K D. \tag{11}$$

Note that even if RoPE is applied, such that $q = R_x \mathbf{W}_Q x$, the bound remains identical as $\|R\mathbf{W}\| = \|\mathbf{W}\|$ (if the Frobenius or spectral is used).

Finally, we consider the quadratic term $\|\Delta q\| \|\Delta k\|$,

$$\begin{aligned} \|\Delta q\| \|\Delta k\| &\leq \|\Delta \mathbf{W}_Q\| \|\Delta \mathbf{W}_K\| \\ &= \frac{\tau_Q \tau_K \|\mathbf{G}_Q\| \|\mathbf{G}_K\|}{\|\mathbf{W}_Q\| \|\mathbf{W}_K\|} \\ &\leq \frac{\tau_Q \tau_K D^2}{c^2}. \end{aligned} \tag{12}$$

Thus the change in logits is bounded by a constant. □

C Extension to MLA

In this section we motivate Algorithm 2, specifically the factors associated with each weight.

We use a similar approach to Section 3 / Appendix B when extending to MLA (Ji et al., 2025; Liu et al., 2024). For now, assume the single-head setting. MLA tells us to calculate queries and keys as follows,

$$q = \text{Concat}(q_{\text{nope}}, q_{\text{rope}}) \tag{13a}$$

$$k = \text{Concat}(k_{\text{nope}}, k_{\text{rope}}) \tag{13b}$$

$$q_{\text{nope}} = \mathbf{W}_{\text{uq}} \mathbf{W}_{\text{dq}} x \tag{13c}$$

$$q_{\text{rope}} = R_x(\mathbf{W}_{\text{qr}} \mathbf{W}_{\text{dq}} x) \tag{13d}$$

$$c_{\text{kv}} = \mathbf{W}_{\text{dkv}} y \tag{13e}$$

$$k_{\text{nope}} = \mathbf{W}_{\text{uk}} c_{\text{kv}} = \mathbf{W}_{\text{uk}} \mathbf{W}_{\text{dkv}} y \tag{13f}$$

$$k_{\text{rope}} = R_y(\mathbf{W}_{\text{kr}} y). \tag{13g}$$

Here, x and y are two token embeddings. The 'down' matrices, $\mathbf{W}_{\text{dq}}$ and $\mathbf{W}_{\text{dkv}}$, project queries and keys/values respectively down to a lower dimensional latent space. This enables efficient caching of c_{kv}. The 'up' matrices $\mathbf{W}_{\text{uq}}$, $\mathbf{W}_{\text{uk}}$ project these latents up to a higher dimensional space for attention calculations on each head. The $\mathbf{W}_{\text{qr}}$ and $\mathbf{W}_{\text{kr}}$ matrices are used to produce decoupled queries and keys for RoPE (Su et al., 2021) embeddings, with the position embedding applied via the rotation matrices R_x and R_y.

The change in logits is given by,

$$d_{\text{head}}^{1/2} |\Delta \ell| = |(q + \Delta q)^T (k + \Delta k) - q^T k| = |(\Delta q)^T k + q^T \Delta k + (\Delta q)^T \Delta k|, \tag{14}$$

Algorithm 2 QuacK (MLA)

Require: Hyperparameter τ, base learning rate η

Make the following additions to the transformer training script:

function `compute_lr_factors`()

for all layers ℓ **do**

for all heads h **do**

$\mathbf{W}_{\text{uq}}^{\ell,h}$.`factor` $\leftarrow (\|\mathbf{W}_{\text{dq}}^{\ell}\|\,\|\mathbf{W}_{\text{uk}}^{\ell,h}\|\,\|\mathbf{W}_{\text{dkv}}^{\ell}\|)^{-1}$

$\mathbf{W}_{\text{uk}}^{\ell,h}$.`factor` $\leftarrow (\|\mathbf{W}_{\text{uq}}^{\ell,h}\|\,\|\mathbf{W}_{\text{dq}}^{\ell}\|\,\|\mathbf{W}_{\text{dkv}}^{\ell}\|)^{-1}$

$\mathbf{W}_{\text{qr}}^{\ell,h}$.`factor` $\leftarrow (\|\mathbf{W}_{\text{dq}}^{\ell}\|\,\|\mathbf{W}_{\text{kr}}^{\ell}\|)^{-1}$

end for

$\mathbf{W}_{\text{dq}}^{\ell}$.`factor` $\leftarrow \min\Big\{(\max_h \|\mathbf{W}_{\text{uq}}^{\ell,h}\|\|\mathbf{W}_{\text{uk}}^{\ell,h}\|\|\mathbf{W}_{\text{dkv}}^{\ell}\|)^{-1},\ (\max_h \|\mathbf{W}_{\text{qr}}^{\ell,h}\|\|\mathbf{W}_{\text{kr}}^{\ell}\|)^{-1}\Big\}$

$\mathbf{W}_{\text{dkv}}^{\ell}$.`factor` $\leftarrow (\max_h \|\mathbf{W}_{\text{uq}}^{\ell,h}\|\|\mathbf{W}_{\text{dq}}^{\ell}\|\|\mathbf{W}_{\text{uk}}^{\ell,h}\|)^{-1}$

$\mathbf{W}_{\text{kr}}^{\ell}$.`factor` $\leftarrow (\max_h \|\mathbf{W}_{\text{qr}}^{\ell,h}\|\|\mathbf{W}_{\text{dq}}^{\ell}\|)^{-1}$

end for

end function

$\{$`attention_weights`$\}$ $\leftarrow \{\mathbf{W}_{\text{uq}}^{\ell,h}, \mathbf{W}_{\text{uk}}^{\ell,h}, \mathbf{W}_{\text{qr}}^{\ell,h}, \mathbf{W}_{\text{dq}}^{\ell}, \mathbf{W}_{\text{dkv}}^{\ell}, \mathbf{W}_{\text{kr}}^{\ell}$ for all layers ℓ for all heads $h\}$

At initialization. Compute initial learning rate factors for all attention weights

`compute_lr_factors`()

for all $\mathbf{W}$ in $\{$`attention_weights`$\}$ **do**

$\mathbf{W}$.`init_factor` $\leftarrow \mathbf{W}$.`factor`

end for

During training. Prior to each optimization step, adjust learning rates

`compute_lr_factors`()

for all $\mathbf{W}$ in $\{$`attention_weights`$\}$ **do**

$\mathbf{W}$.`lr` $\leftarrow \tau\,\eta \cdot \dfrac{\mathbf{W}\text{.factor}}{\mathbf{W}\text{.init\_factor}}$

end for

and we can bound the change,

$$\begin{aligned} d_{\text{head}}^{1/2}|\Delta\ell| &\le |(\Delta q)^T k| + |q^T \Delta k| + \|\Delta q\|\|\Delta k\| \\ &\le |(\Delta q_{\text{nope}})^T k_{\text{nope}}| + |(\Delta q_{\text{rope}})^T k_{\text{rope}}| + |q_{\text{nope}}^T \Delta k_{\text{nope}}| + |q_{\text{rope}}^T \Delta k_{\text{rope}}| + \|\Delta q\|\|\Delta k\|. \end{aligned} \tag{15}$$

Expanding further, for the queries, we have,

$$\begin{aligned} \Delta q_{\text{nope}} &= (\mathbf{W}_{\text{uq}} + \Delta\mathbf{W}_{\text{uq}})(\mathbf{W}_{\text{dq}} + \Delta\mathbf{W}_{\text{dq}})x - \mathbf{W}_{\text{uq}}\mathbf{W}_{\text{dq}}x \\ &= \Delta\mathbf{W}_{\text{uq}}\mathbf{W}_{\text{dq}}x + \mathbf{W}_{\text{uq}}\Delta\mathbf{W}_{\text{dq}}x + \Delta\mathbf{W}_{\text{uq}}\Delta\mathbf{W}_{\text{dq}}x, \qquad (16a) \\ \Delta q_{\text{rope}} &= R_x[(\mathbf{W}_{\text{qr}} + \Delta\mathbf{W}_{\text{qr}})(\mathbf{W}_{\text{dq}} + \Delta\mathbf{W}_{\text{dq}})x - \mathbf{W}_{\text{qr}}\mathbf{W}_{\text{dq}}x] \\ &= R_x[\Delta\mathbf{W}_{\text{qr}}\mathbf{W}_{\text{dq}}x + \mathbf{W}_{\text{qr}}\Delta\mathbf{W}_{\text{dq}}x + \Delta\mathbf{W}_{\text{qr}}\Delta\mathbf{W}_{\text{dq}}x], \qquad (16b) \end{aligned}$$

and for the keys,

$$\begin{aligned} \Delta k_{\text{nope}} &= (\mathbf{W}_{\text{uk}} + \Delta\mathbf{W}_{\text{uk}})(\mathbf{W}_{\text{dkv}} + \Delta\mathbf{W}_{\text{dkv}})y - \mathbf{W}_{\text{uk}}\mathbf{W}_{\text{dkv}}y \\ &= \Delta\mathbf{W}_{\text{uk}}\mathbf{W}_{\text{dkv}}y + \mathbf{W}_{\text{uk}}\Delta\mathbf{W}_{\text{dkv}}y + \Delta\mathbf{W}_{\text{uk}}\Delta\mathbf{W}_{\text{dkv}}y \qquad (17a) \\ \Delta k_{\text{rope}} &= R_y[(\mathbf{W}_{\text{kr}} + \Delta\mathbf{W}_{\text{kr}})y - \mathbf{W}_{\text{kr}}y] = R_y\Delta\mathbf{W}_{\text{kr}}y. \qquad (17b) \end{aligned}$$

We now use these expressions, and the expressions for q_{nope}, k_{nope}, q_{rope}, k_{rope}, to bound each of the terms in Eq. (15). We will make some assumptions (similar to Lemma 1),

- the inputs x and y are unit normed;
- we use a submultiplicative norm (e.g. the Frobenius or Spectral norm);
- conditioned gradients are bounded by a constant, i.e. $\Delta\mathbf{W}_{\mathtt{x}} = -\eta_{\mathtt{x}}\mathbf{G}_{\mathtt{x}}$ where $\|\mathbf{G}_{\mathtt{x}}\| \le D$ (valid for Muon and Adam);
- the weight norms are lower bounded by a constant c.

We consider the first order terms. We have,

$$\begin{aligned}|\Delta q_{\text{nope}}^T k_{\text{nope}}| &= |(\Delta \mathbf{W}_{\text{uq}}\mathbf{W}_{\text{dq}}x + \mathbf{W}_{\text{uq}}\Delta\mathbf{W}_{\text{dq}}x + \Delta\mathbf{W}_{\text{uq}}\Delta\mathbf{W}_{\text{dq}}x)^T k_{\text{nope}}| \\ &\leq \|\Delta\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|\|k_{\text{nope}}\| + \|\mathbf{W}_{\text{uq}}\|\|\Delta\mathbf{W}_{\text{dq}}\|\|k_{\text{nope}}\| + \|\Delta\mathbf{W}_{\text{uq}}\|\|\Delta\mathbf{W}_{\text{dq}}\|\|k_{\text{nope}}\| \\ &\leq \eta_{\text{uq}}D\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\| + \eta_{\text{dq}}D\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\| + O(\eta_{\text{uq}}\eta_{\text{dq}}\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\|)\end{aligned} \tag{18a}$$

$$\begin{aligned}|(\Delta q_{\text{rope}})^T k_{\text{rope}}| &= |(R_x[\Delta\mathbf{W}_{\text{qr}}\mathbf{W}_{\text{dq}}x + \mathbf{W}_{\text{qr}}\Delta\mathbf{W}_{\text{dq}}x + \Delta\mathbf{W}_{\text{qr}}\Delta\mathbf{W}_{\text{dq}}x])^T k_{\text{rope}}| \\ &\leq (\|\Delta\mathbf{W}_{\text{qr}}\|\|\mathbf{W}_{\text{dq}}\| + \|\mathbf{W}_{\text{qr}}\|\|\Delta\mathbf{W}_{\text{dq}}\| + \|\Delta\mathbf{W}_{\text{qr}}\|\|\Delta\mathbf{W}_{\text{dq}}\|)\|k_{\text{rope}}\| \\ &\leq \eta_{\text{qr}}D\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{kr}}\| + \eta_{\text{dq}}D\|\mathbf{W}_{\text{qr}}\|\|\mathbf{W}_{\text{kr}}\| + O(\eta_{\text{dq}}\eta_{\text{qr}}\|\mathbf{W}_{\text{kr}}\|),\end{aligned} \tag{18b}$$

$$\begin{aligned}|q_{\text{nope}}^T \Delta k_{\text{nope}}| &= |q_{\text{nope}}^T(\Delta\mathbf{W}_{\text{uk}}\mathbf{W}_{\text{dkv}}y + \mathbf{W}_{\text{uk}}\Delta\mathbf{W}_{\text{dkv}}y + \Delta\mathbf{W}_{\text{uk}}\Delta\mathbf{W}_{\text{dkv}}y)| \\ &\leq \|q_{\text{nope}}\|\|\Delta\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\| + \|q_{\text{nope}}\|\|\mathbf{W}_{\text{uk}}\|\|\Delta\mathbf{W}_{\text{dkv}}\| + \|q_{\text{nope}}\|\|\Delta\mathbf{W}_{\text{uk}}\|\|\Delta\mathbf{W}_{\text{dkv}}\| \\ &\leq \eta_{\text{uk}}D\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{dkv}}\| + \eta_{\text{dkv}}D\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{uk}}\| + O(\eta_{\text{uk}}\eta_{\text{dkv}}\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|),\end{aligned} \tag{18c}$$

$$|q_{\text{rope}}^T \Delta k_{\text{rope}}| = |q_{\text{rope}}^T(R_y\Delta\mathbf{W}_{\text{kr}}y)| \leq \|q_{\text{rope}}\|\|\Delta\mathbf{W}_{\text{kr}}\| \leq \eta_{\text{kr}}D\|\mathbf{W}_{\text{qr}}\|\|\mathbf{W}_{\text{dq}}\|. \tag{18d}$$

We used the fact that for rotation matrices R, $\|\mathbf{W}R\| = \|R\mathbf{W}\| = \|\mathbf{W}\|$.

Ultimately, Eqs. (18) suggest to set the learning rates for each attention weight parameter as,

$$\eta_{\text{uq}} = \tau(\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\|)^{-1} \tag{19a}$$

$$\eta_{\text{dq}} = \tau\ \min\left\{(\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\|)^{-1}, (\|\mathbf{W}_{\text{qr}}\|\|\mathbf{W}_{\text{kr}}\|)^{-1}\right\} \tag{19b}$$

$$\eta_{\text{qr}} = \tau(\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{kr}}\|)^{-1} \tag{19c}$$

$$\eta_{\text{uk}} = \tau(\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{dkv}}\|)^{-1} \tag{19d}$$

$$\eta_{\text{dkv}} = \tau(\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{uk}}\|)^{-1} \tag{19e}$$

$$\eta_{\text{kr}} = \tau(\|\mathbf{W}_{\text{qr}}\|\|\mathbf{W}_{\text{dq}}\|)^{-1}. \tag{19f}$$

We then substitute these learning rates into Eqs. (18), to see that the bounds are given by,

$$\begin{aligned}|(\Delta q_{\text{nope}})^T k_{\text{nope}}| &\leq \tau D + \tau D + O\left(\frac{\tau^2\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\|}{\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\| \cdot \|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\|}\right) \\ &= 2\tau D + O\left(\frac{\tau^2}{\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\|}\right),\end{aligned} \tag{20a}$$

$$\begin{aligned}|(\Delta q_{\text{rope}})^T k_{\text{rope}}| &\leq \tau D + \tau D + O\left(\frac{\tau^2\|\mathbf{W}_{\text{kr}}\|}{\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{kr}}\| \cdot \|\mathbf{W}_{\text{qr}}\|\|\mathbf{W}_{\text{kr}}\|}\right) \\ &= 2\tau D + O\left(\frac{\tau^2}{\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{qr}}\|\|\mathbf{W}_{\text{kr}}\|}\right),\end{aligned} \tag{20b}$$

$$\begin{aligned}|q_{\text{nope}}^T \Delta k_{\text{nope}}| &\leq \tau D + \tau D + O\left(\frac{\tau^2\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|}{\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{dkv}}\| \cdot \|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{uk}}\|}\right) \\ &= 2\tau D + O\left(\frac{\tau^2}{\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\|}\right),\end{aligned} \tag{20c}$$

$$|q_{\text{rope}}^T \Delta k_{\text{rope}}| \leq \tau D. \tag{20d}$$

It is reasonable to assume in practice that the weights are not arbitrarily small (i.e. their norm is lower bounded), and thus that these terms are bounded by a constant.

The only remaining term to bound in Eq. (15) is the quadratic term, $\|\Delta q\|\|\Delta k\|$. We can show that this is bounded by showing that the individual parts are bounded,

$$\begin{aligned}\|\Delta q_{\text{nope}}\| &\leq \eta_{\text{uq}} D\|\mathbf{W}_{\text{dq}}\| + \eta_{\text{dq}} D\|\mathbf{W}_{\text{uq}}\| + \eta_{\text{uq}}\eta_{\text{dq}} D^2 \\ &\leq \frac{2\tau D}{\|\mathbf{W}_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\|} + \eta_{\text{uq}}\eta_{\text{dq}} D^2, \end{aligned} \tag{21a}$$

$$\begin{aligned}\|\Delta q_{\text{rope}}\| &\leq \eta_{\text{qr}} D\|\mathbf{W}_{\text{dq}}\| + \eta_{\text{dq}} D\|\mathbf{W}_{\text{qr}}\| + \eta_{\text{qr}}\eta_{\text{dq}} D^2 \\ &\leq \frac{2\tau D}{\|\mathbf{W}_{\text{kr}}\|} + \eta_{\text{qr}}\eta_{\text{dq}} D^2, \end{aligned} \tag{21b}$$

$$\begin{aligned}\|\Delta k_{\text{nope}}\| &\leq \eta_{\text{uk}} D\|\mathbf{W}_{\text{dkv}}\| + \eta_{\text{dkv}} D\|\mathbf{W}_{\text{uk}}\| + \eta_{\text{uk}}\eta_{\text{dkv}} D^2 \\ &\leq \frac{2\tau D}{\|\mathbf{W}_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|} + \eta_{\text{uk}}\eta_{\text{dkv}} D^2, \end{aligned} \tag{21c}$$

$$\|\Delta k_{\text{rope}}\| \leq \eta_{\text{kr}} D \leq \frac{\tau D}{\|\mathbf{W}_{\text{qr}}\|\|\mathbf{W}_{\text{dq}}\|}. \tag{21d}$$

To extend further to the multi-head setting, we add head indices to the necessary matrices, $\mathbf{W}^h_{\text{uq}}$, $\mathbf{W}^h_{\text{uk}}$, and $\mathbf{W}^h_{\text{qr}}$, and their corresponding learning rates, η^h_{uq}, η^h_{uk}, η^h_{qr}. The key used for RoPE, k_{rope} is shared between all heads, therefore $\mathbf{W}_{\text{kr}}$ surprisingly does not have a head index. The down matrices project to a latent space, so also do not have head indices. Plugging these into Eqs. (19) we have,

$$\eta^h_{\text{uq}} = \tau(\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}^h_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\|)^{-1} \tag{22a}$$

$$\eta_{\text{dq}} = \tau \ \min\left\{\left(\max_h \|\mathbf{W}^h_{\text{uq}}\|\|\mathbf{W}^h_{\text{uk}}\|\|\mathbf{W}_{\text{dkv}}\|\right)^{-1}, \left(\max_h \|\mathbf{W}^h_{\text{qr}}\|\|\mathbf{W}_{\text{kr}}\|\right)^{-1}\right\} \tag{22b}$$

$$\eta^h_{\text{qr}} = \tau(\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{kr}}\|)^{-1} \tag{22c}$$

$$\eta^h_{\text{uk}} = \tau(\|\mathbf{W}^h_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}_{\text{dkv}}\|)^{-1} \tag{22d}$$

$$\eta_{\text{dkv}} = \tau(\max_h \|\mathbf{W}^h_{\text{uq}}\|\|\mathbf{W}_{\text{dq}}\|\|\mathbf{W}^h_{\text{uk}}\|)^{-1} \tag{22e}$$

$$\eta_{\text{kr}} = \tau(\max_h \|\mathbf{W}^h_{\text{qr}}\|\|\mathbf{W}_{\text{dq}}\|)^{-1}. \tag{22f}$$

The use of $\max_h(\cdot)$ comes from the requirement that we want logit changes to be bounded for all heads.