Thank you for all the helpful comments. Several related works were raised by the reviewers which we discuss here.

Time-Delay Momentum: We note that the authors have marked their ArXiv submission as containing errors. While their algorithm also utilizes an inner-outer loop structure, it differs significantly from ours in motivation and implementation. Each of their inner loops uses SGD to solve the *distance-regularized* objectives. We instead allow arbitrary optimization algorithms in the inner-loop and solve the *original objective*. Our main contribution is introducing interpolation to update both the slow and fast weights. This is a critical difference which leads to an exponential moving average (EMA) with advantages such as variance reduction [Martens 2014, Polyak and Juditsky, 1992].[1] **SWA:** While both SWA and Lookahead average network weights, we believe they serve different purposes and are complementary. First, we use the EMA of slow weights to adjust the training parameters during optimization. Compared to using the EMA at inference, this gives much faster convergence via variance reduction along the trajectory (see comparison to Polyak averaging in Figure 5). Second, SWA is applied near convergence, whereas Lookahead is applied throughout training. This removes the challenge of deciding when to start iterate averaging. We include an evaluation with SWA in Figure 1. We use 3 random seeds with publicly available code and hyperparameters[2] from Izmailov et al. to show that Lookahead and SWA are complementary. Lookahead wrapped around SGD dominates SGD in performance during training and improves the weight averaged network.[3] We observe similar behavior on ResNet-110 and VGG-16.

Reviewer 3 Adaptive Learning Rate Result We believe this is an interesting result but agree that it is not critical and do not mind excluding it.

SUM Framework Yan et al. prove results for non-convex stochastic losses. Furthermore, Lookahead does not fit into their SUM framework. We do not believe that this work is relevant to Section 3.2 or otherwise strongly related to our own.

Strength of empirical results We respectfully disagree. It is difficult to improve on the convergence and final performance of the carefully tuned baseline methods, which is why they have remained the primary choices for so long. Despite this, we show significant wins in terms of convergence and final performance over a range of different tasks. For language modelling we provide 1.5 perplexity improvement with much faster convergence (Table 3). On image classification tasks we are consistently able to achieve better final performance in fewer update steps. We also show that Lookahead is very robust to different inner-optimizer settings (Figure 8). A method that reduces hyperparameter tuning, improves convergence, and strictly[4] improves final performance is useful for the community, especially to resource-constrained groups.

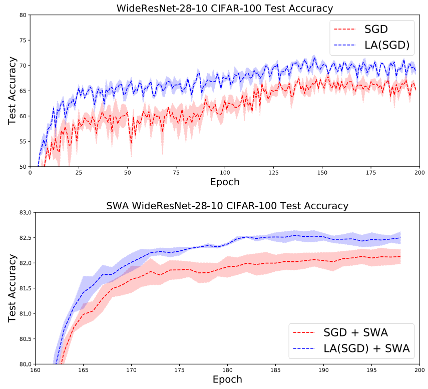

Figure 1: Test Accuracy on CIFAR-100 with SWA and Lookahead (Wide ResNet-28-10). Following Izmailov et al., SWA is started at epoch 161.

Reviewer 5 We used the ResNet-18 architecture, which was designed for ImageNet but has been used for CIFAR in work such as DeVries and Taylor [2017], with which our numbers agree.

Reviewer 6 We have corrected typos in Proposition 1 and Algorithm 1 and fixed other minor mistakes.

On NQM model: This model has been studied in prior work [Martens 2014, Wu et al., 2018, Lucas et al., 2018] and **is** equivalent to general stochastic convex quadratics under the studied schemes (see Sutskever et al. 2013, Proposition 6.1). This system is **not** equivalent to decoupled 1-D problems as the learning rate is shared over all dimensions. Furthermore, [Zhang et al., 2019] recently showed that insights from this NQM capture many essential features of neural net training.

LA inner learning rate for CIFAR: We used 0.1 in the reported results except Figure 8.

Rate of contraction: We will replace this with the decrease in the loss though the same conclusions apply. The linear dynamical system admits a unique fixed point at the minima (origin) for both lookahead and CM — thus the rate of contraction is a valid measure of convergence (see Lessard et al., 2014).

SWA multiple networks: We mean that different neural network weights are added to the moving average during training and this is interpreted (loosely) as ensembling. We will fix the wording here, thank you.

Additional references: Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

Guodong Zhang, Lala Li, Zachary Nado, James Martens, Sushant Sachdeva, George E Dahl, Christopher J Shallue, and Roger Grosse. Which algorithmic choices matter at which batch sizes? insights from a noisy quadratic model. *arXiv preprint arXiv:1907.04164*, 2019.

[1] Citations not in references below are in the original manuscript.

[2] 200 epochs of training with SWA initalized from epoch 161, initial LR 0.1, SWA LR 0.05 and weight decay of 5×10^{-4}

[3] Note that we do not tune the learning rate schedule for Lookahead and follow the schedule proposed by Izmailov et al. The learning rate is higher than is typical at the end of training, which explains the large gap to the non-SWA performance.

[4] We did not find tasks for which Lookahead performed worse than its inner optimizer, unlike adaptive vs non-adaptive gradient-based optimizers.