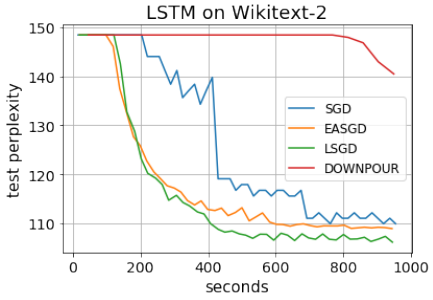
We thank all the Reviewers for their time and raising several interesting questions. We appreciate kind comments. Please see our responses below.

Reviewer #1: We will try to reduce dependence on the Supplement. Regarding the non-leader workers in LSGD, we believe that these two facts together —- (1) the schedule of leader switching recorded in the experiments shows frequent switching, and (2) the leader point itself is not pulled away from minima — suggest that the 'pulling away' in LSGD is beneficial: non-leader workers that were pulled away from local minima later became the leader, and thus likely obtained an even better solution than they originally would have. We will add an explanation of this interesting phenomenon in the final paper. The name Vol in §3.3 refers to Volume, which for the ellipsoid $\{x : x^T Ax \leq 1\}$ is given by $\det(A)^{-1/2}\,\mathrm{Volume}(S_n)$, where S_n is the unit ball. We will add this definition.

Reviewer #2: We will add a comment comparing the convergence rate of LSGD to other distributed methods. For the DOWNPOUR method specifically, the original paper is purely empirical, and we are not aware of any published convergence analysis for it. The closest proxy for DOWNPOUR with a known rate is the Hogwild method of Recht, B. et al. [2011], which achieves a rate of $O(1/k)$ (up to a logarithmic term). The EASGD method also achieves $O(1/k)$ on strongly convex objective functions. In both cases, this rate matches that of LSGD. Unfortunately many distributed algorithms are presented without theoretical analysis (e.g. DOWNPOUR, PARLE).

Reviewer #3: Regarding the asynchronous algorithm, we opted to present results for two settings, one-step round and 'arbitrarily long' round, in the main paper because numerous variations of communication schedules are possible. The behavior of the algorithm given an unknown round length >1 is very difficult to measure in a useful way (i.e., to find quantifiable improvements) since it requires estimating the lowest value obtained along the trajectory when the leader is kept fixed, which makes it difficult to define a rate for the 'general' asynchronous method. Note that several useful lemmas for the analysis of stochastic leader selection are presented in the Supplement, which can also be combined to analyze combinations of the asynchronous algorithm with stochastic leader selection.

Regarding the similarity to Newton directions, note that our volume theorems apply only for convex functions (or in the neighborhoods of local minimizers). They do not apply in general for nonconvex functions, and thus they do not imply improvements or, conversely, harmful behavior. For nonconvex functions, our intuition is that many candidate leaders will be in directions of negative curvature which would actually draw us away from saddle points, but measuring this volume is significantly more complicated because the set of candidate leaders is a priori unbounded.



Regarding using the leader's parameter versus the average of the parameters of all the workers, note that we use the leader's parameter to pull to at training and we report the averaged parameters at testing deliberately. It is demonstrated in our paper (e.g.: Fig. 1) that pulling workers to the averaged parameters at training may slow down convergence and we address this problem. Note that after training, the parameters that workers obtained after convergence will likely lie in the same valley of the landscape (see Baldassi, C. et al. [2016]) and thus their average is expected to have better generalization ability (e.g. Chaudhari P. et al. [2017], Izmailov, P. et al. [2018]), which is why we report the results for averaged parameters at testing.

We report additional experiment with LSTM on the NLP task (world-level language modeling) in the figure above. We are also running additional experiments. We will add them to the paper.

Regarding SGD and ResNet50, SGD is consistently worse than all reported methods (training on ImageNet with SGD on a single GTX1080 GPU until convergence usually takes about a week and gives slightly worse final performance), which is why the SGD curve was deliberately omitted (other methods converge in around two days).

	LSGD	EASGD	DOWNPOUR
CNN7: 4/16 workers	1%/2%	2%/4%	20%/57%
ResNet20: 4/16 workers	1%/2%	2%/4%	21%/50%
VGG16	2%	3%	34%
ResNet50	1%	2%	17%

Regarding learning rate selection for each method, we chose the one leading to the smallest achievable test error under similar convergence rates (we rejected small learning rates which led to unreasonably slow convergence).

The breakdown of the total time will be added, see provided table reporting the proportion of communication costs with respect to the total time. LSGD is roughly twice more communication-efficient than EASGD. Local SGD is indeed effective in terms of communication, but at the cost of a significant drop in performance compared to DOWNPOUR, which is why we have not been reporting this method.

References

Baldassi, C. et al. Unreasonable effectiveness of learning neural networks: From accessible states and robust ensembles to basic algorithmic schemes. In *PNAS*, 2016.

Chaudhari P. et al. Entropy-SGD: Biasing gradient descent into wide valleys. In *ICLR*, 2017.

Izmailov, P. et al. Averaging weights leads to wider optima and better generalization. *arXiv:1803.05407*, 2018.

Recht, B. et al. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In *NIPS*, 2011.