Many thanks to the reviewers for their deep, thoughtful reviews and constructive suggestions.

R1. ADA-LUPA clarification & Advantages: Thanks for the suggestion. We will elaborate more on the adaptive algorithm. We note that despite very recent observations on empirical superiority of adaptive synchronization (e.g., Figure 4 in our experiments, and [12], and [40] that demonstrate adaptivity can reduce the number of communication in minimizing wall-clock time or leads to a better generalization error), it lacks theoretical understanding. Indeed, Theorem 2 in our paper is the first non-asymptotic analysis of convergence of adaptive local SGD which matches the improved non-additive counterpart. Surely, it would be interesting to see if our bound can be tightened.

R1. Ada-LUPA vs. existing works: Thanks for pointing out these references. Here we briefly highlight a few key differences and will add a detailed comparison in the subsequent version of our paper: 1) The result of [Wang & Joshi] is based on minimizing convergence error with respect to the wall-clock time using an adaptive synchronization schema, while our focus is on reducing the number of communications in terms of number of iterations. Obviously, our analysis can be extended to take into account the wall-clock time into consideration to further improve the communication-computation complexity of [Wang & Joshi], and 2) The convergence analysis of their algorithm is asymptotic in essence, while ours is non-asymptotic. The LAG algorithm proposed in [Chen et al], aims at solving the distributed optimization as low communication **overhead** as possible in an adaptive manner (skipping over some local gradients and using outdated gradients instead), while ADA-LUPA reduces the **number** of communication rounds by reducing communication frequency. It is noticeable that the adaptive schema used in [Wang & Joshi] is different from ours as it starts with infrequent averaging to improve convergence speed, and then increases the communication frequency in order to achieve a low error floor. Our schema is consistent with [40] which uses frequent communication at the beginning (warm-up stage) and then infrequent communication with **fixed** number of local updates to reduce the number of communications as we aimed for, but the empirical results are not supported by theoretical analysis.

R1. $\log \mathbf{T}$ **communication rounds clarification:** At first glance, given that adaptively reducing the communication frequency works well in practice, it might seem that in our adaptive schema by exponentially increasing the number of local updates, one can get linear speedup with log T number of communications. However, the main theoretical insight of ADA-LUPA algorithm is to provide some intuition on how big we can choose τ_i– under our setup and convergence techniques while preserving linear speed up. To see this let $\tau_i = a\tau_{i-1}$ for some $a > 1, i \geq 2$, which means that $O(\log(\frac{Ta}{\tau_1}))$ communication rounds is needed. However, according to Theorem 2, this choice of τ_i does not allow linear speed up with respect to the number of workers. Quantifying improvement of ADA-LUPA over LUPA from a theoretical standpoint or requiring log T communications with some other tweaks (e.g., using increasing batch sizes) are interesting future directions that are worthy of investigation.

R2. P-L vs. strong-convexity & proof novelty: We agree with the reviewer that the convergence proof of *non-local* gradient descent based algorithms with P-L condition is similar or even simpler than the strong-convexity based analysis. However, for local SGD with periodic averaging the proof techniques are more involved. The key challenge is to periodically cancel out the effect of growing $\|\mathbf{g}\|_2^2$ from upper bound (Lemma 4) which requires a different set of novel techniques which distinguishes our analysis from analysis of non-local methods. **We note that even under strong-convexity assumption and removing bounded gradient assumption we obtain an improved communication complexity while preserving the same convergence upper compared to [14] due to our tight analysis as pointed out in the paper**. Regarding the applicability, we remark that while many convex optimization problems (like linear or logistic regression) does not satisfy strong convexity generally, they satisfy P-L condition (e.g. see [15]). Also, the P-L condition allows the generalization of convergence analysis to general non-convex optimization problems similar to [43], which we leave as future work.

R2. Practicality of ADA-LUPA & optimality: Many thanks for your meticulous attention. We will make this point clear in the statement of the theorem. We note that having access to the function $F(\mathbf{x}^{(t)})$ is only for theoretical analysis purposes and is not necessary in practice as long as the choice of τ_i satisfies the conditions in the statement of the theorem. In fact as it is explained in our experiments, we do NOT use the function value oracle and increase τ_i within each communication period linearly (please see Figure 4) which demonstrates huge improvement. However, we believe that with a more intelligent adjustment we can achieve faster convergence, hence, we will investigate other adjustment schemes and their effects as well. Also, as indicated in [Wang & Joshi] having access to the zero oracles of functions could be possible and we believe it is possible to derive some optimality criteria similar to [Wang & Joshi].

R2. Numerical investigation & more experiments: We do not tune the learning rate. The goal of experiments is simply to show the effectiveness of our algorithm compared to other baselines such as syncSGD. Hence, we kept all the hyperparameters the same for different experiments in this figure to have a fair comparison. Although the experiments on this dataset are showing promising results, we will definitely run on several more datasets with different loss functions to better understand the effects of local updates and make our case stronger.

R3. Optimality: We note that unlike non-local distributed methods, the communication complexity of local SGD is not well-understood. Although our work makes a step towards tightening the state-of-the-art communication complexity, as our empirical results demonstrated, it could be further improved. That being said, our goal is to achieve a linear speed up with the smallest mini-batch size and the largest possible τ as the measure of optimality, considering there key parameters p,b, and τ involved in our convergence error. If we weaken these goals either by using larger mini-batch size as done in [43] or giving up on linear speed up, our convergence analysis can be extended to show convergence with a much smaller number of communication rounds. We will add some empirical experiments regarding the optimality of the bound for the subsequent version and leave the theoretical understanding as an interesting open question.