Many thanks to all the reviewers for their dedicated work and helpful comments—we will be sure to incorporate the suggestions to improve the paper!

For Reviewer 1: Thanks very much for your suggestions in improving the presentation of the theorem! b is indeed a free parameter. It can be any positive value without harming the asymptotic rate. In fact, Theorem 1 should have said "for any $b > 0$" rather than "for some $b > 0$". In principle, one could find a value for b in terms of the other problem-dependent parameters like σ and $F(x_1) - F^\star$ that would optimize the non-dominant term of the bound, although we did not do so. The notion to simplify the presentation M is a good one, we will do this in the final version.

For Reviewer 2: Indeed, the reference you point out is an independent work (it seems to have appeared only just before the submission deadline). It is quite interesting that the learning rates are so different!

In regards to the Lipschitz constant: We only need this assumption to support our adaptivity to σ. If instead we were given oracle knowledge of σ (as it is often assumed in other works), then we would not need the Lipschitz assumption—notice that in Lemma 2 we do not actually use this assumption as the lemma is stated in terms of the empirically observed magnitudes of the gradients (The text of Lemma 2 does mention the Lipschitz assumption, but that is an oversight: a quick inspection shows that it was not used). We use the assumption in the algebra of the proof of Theorem 1 in order to bound expected values of sums involving terms like $\frac{\|g_t\|^2}{\eta_t^3}$. If instead we knew σ, we could set η_t to $O((L + \sigma^2 t)^{-1/3})$ and obtain the exact same result (with a few more steps). We preferred to show a stronger and, in our opinion, more interesting result, with the additional assumption of Lipschitzness since σ is typically unknown. However, in the final version we will add the straightforward extension to oracle tuning of the learning rates without Lipschitz assumption. Interestingly, the issue of the Lipschitzness shows up frequently in the adaptive learning literature—see for example the dual averaging version of AdaGrad [Duchi et al., 2011] or adaptive FTRL analysis [McMahan, 2017], which require known Lipschitz bounds in order to obtain adaptivity, but not for convergence.

In regards to the experiments: we used all the default parameters in the Tensor2Tensor package, including random reshuffling, a batch size of 128, and 0.0001 weight decay constant. Zero attempt was made to tune this, so we actually suspect the default settings may mildly favor the Adam algorithm, which was the default optimizer. We concede that the theory does not perfectly apply in this problem (the training loss function of a neural net is not even smooth!), but we still think that the theoretical results provide strong motivation for practical performance.

We will add more detail on this to the text, and we *do* plan to release the code.

For Reviewer 3: Thanks for suggesting the references, we will happily discuss their relationship to our work! We would just briefly stress that, as far as we known, *none of these achieve adaptivity to sigma*. Also, even the methods that manage to have only one or $O(1)$ samples per iterations *still require at least one large batch in the first iteration*. Instead, both these issues are solved with our approach.

In regards to the assumption about bounded gradient with probability 1 vs in expectation: it is actually a bit tricky to go to expectation. A key place we use this assumption is bounding the term A_t in the proof of Theorem 1. Here, it is used to bound $\sum_{t=1}^T G_{t+1}^2 \eta_t^{-3}$ in terms of $\sum_{t=1}^T G_{t+1}^2 \eta_{t+1}^{-3}$ with probability 1. This allows us to perform the sum without taking expectations until the end. With only an in-expectation assumption, we would need to understand $\mathbb{E}[\eta_{t+1}^{-3}]$, which is more subtle. Note that this also underscores why the Lipschitz bound is needed for adaptivity only—if we knew σ then we would choose a deterministic schedule for η which would then be easy to work with.

In regards to the optimality: You are right, although we match the best-known rate in the stochastic setting, our reference actually only proved optimality for the finite-sum setting. We note that their proof, in the case when the number of items in the sum is $O(\epsilon^{-2})$, involves functions that are $O(1)$ Lipschitz and so our algorithm does actually match the lower bound (in fact, this reasoning may actually provide a lower bound for the stochastic setting). We will clarify these issues in the text. Thank you for pointing it out!

As far as resolving the tuning of parameters: certainly we do not claim that STORM completely resolves this, but we do feel that our techniques significantly ameliorate the problem, since in general adaptive algorithms like AdaGrad or Adam seem to be more robust to hyperparameter selection. Also, while we state our algorithm with many parameters, as also noted by Reviewer 1, there is only one free parameter (b) in Theorem 1. However, we completely agree that removing all parameters is an excellent goal and we certainly hope to see such an algorithm in the future!

References

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

H Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. *The Journal of Machine Learning Research*, 18(1):3117–3166, 2017.