Additional Experiments for the Referees

1 Reviewer W3sM

We include an additional comparison to the state-of-the-art DoCoM optimizer. We select DoCoM for comparison, as the theoretical convergence rate is is tighter than ours and the other suggested SOTA methods, and the code and experimental data used by the authors is readily available. We directly use the hyperparameters the authors tuned for the LeNet5 network and FEMNIST dataset, and replicate the experiments from Figures 1, 2, and 3 in our original paper submission. We give these updated figures below. While we are currently constrained for time in this rebuttal, we can provide additional comparisons to the other suggested optimizers and perform further hyper-parameter tuning in our final paper revision. However, a preliminary study suggests that hyper-parameter tuning with DoCoM by itself is unlikely to significantly improve upon the results presented here.

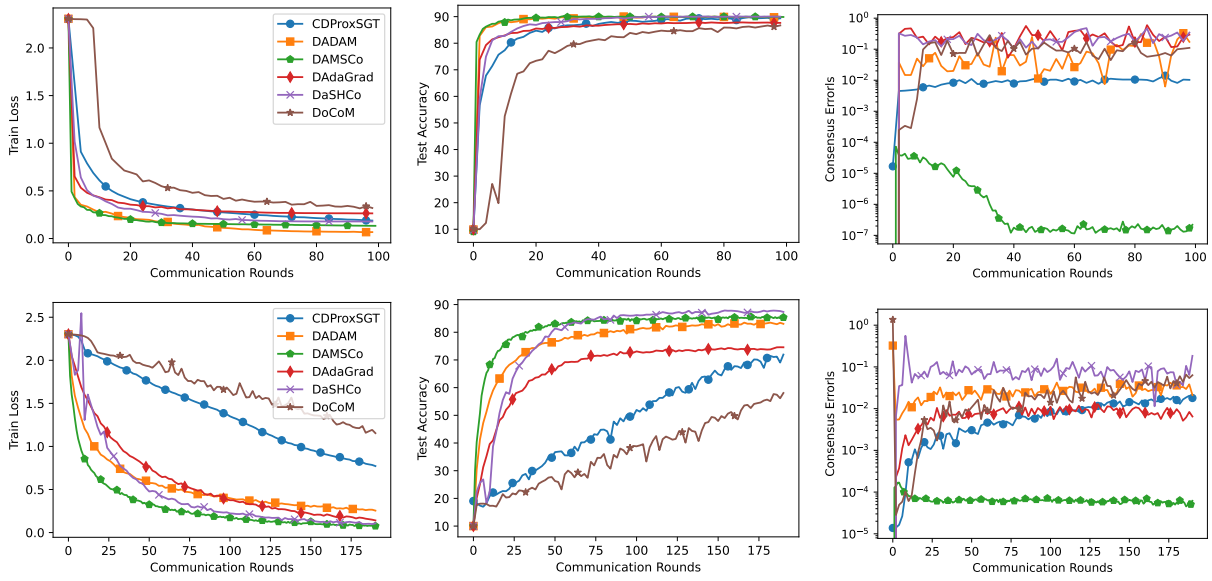

Figure 1: **Results with homogeneous data:** Plotted above are (from left to right) the training loss, test accuracy, and consensus error per communication round for the FashionMNIST on LeNet5 (top) and CIFAR-10 on Fixup-ResNet-20 (bottom) benchmarks, comparing DAMSCo and DaSHCo with DoCoM, CDProxSGT, Distributed AdaGrad, and Distributed Adam with Top-k compression.

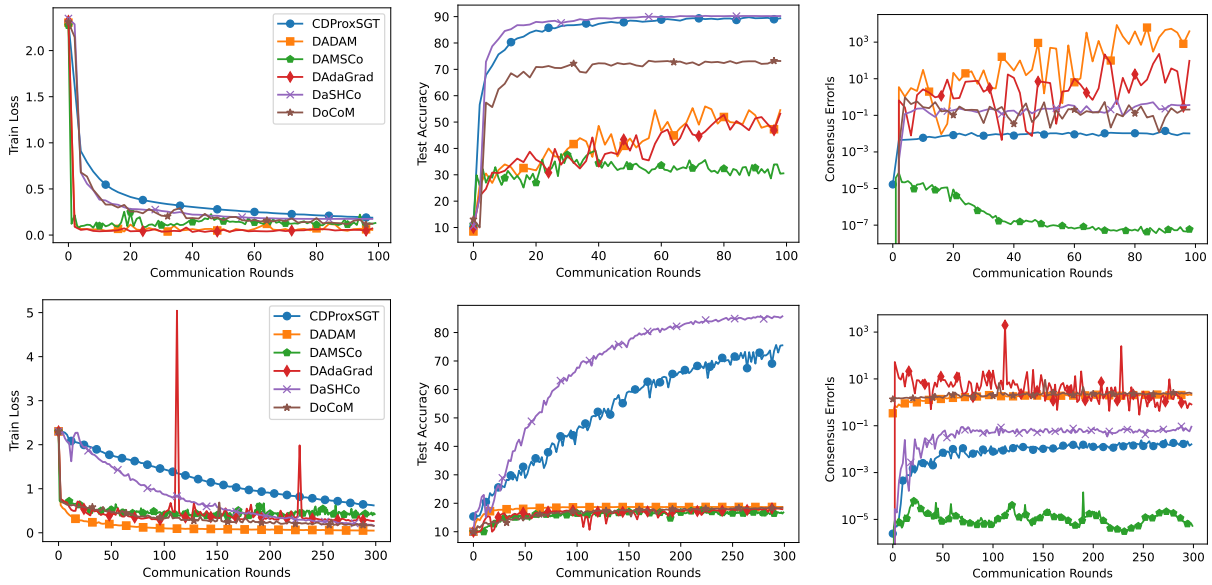

Figure 2: **Results with heterogeneous data:** Plotted above are (from left to right) the training loss, test accuracy, and consensus error per communication round for the FashionMNIST on LeNet5 (top) and CIFAR-10 on Fixup-ResNet-20 (bottom) benchmarks, comparing DAMSCo and DaSHCo with DoCoM, CDProxSGT, Distributed AdaGrad, and Distributed Adam with Top-k compression.

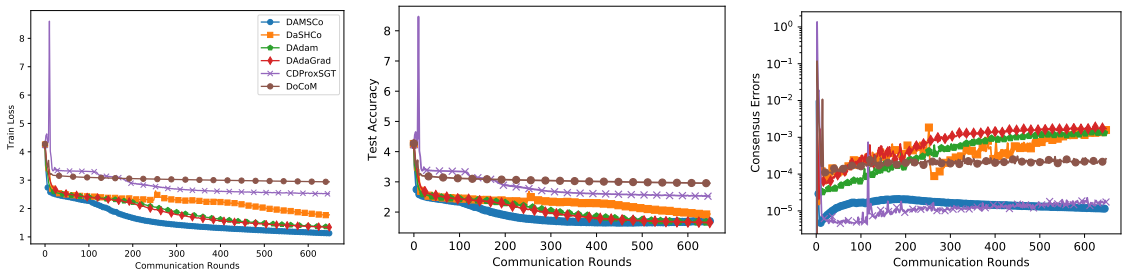

Figure 3: **GPT Results with homogeneous data:** Plotted above are (from left to right) the training loss, validation loss, and consensus error per communication round for the Shakespeare dataset, comparing DAMSCo and DaSHCo with DoCoM, CDProxSGT, Distributed AdaGrad, and Distributed Adam with Top-k compression.

2 Reviewer KRgU

We include additional plots here to demonstrate linear speedup for DAMSCo and DaSHCo, resulting from varying the number of agents to 5, 9, and 16 in a ring toplogy. We utilize the 5 and 9 agent results on FashionMNIST with homogenized data, as given in figures from the main body and appendix of our original submission, and we include an additional experiment with 16 agents. These are plotted in Figure 4. We note that the close overlap of curves for loss and accuracy provide experimental validation of our theoretical expectation of linear speedup.

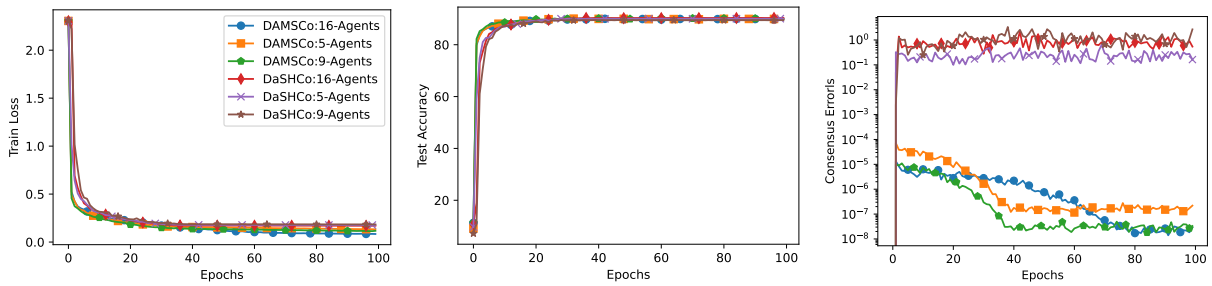

Figure 4: **Results demonstrating linear speedup:** Plotted above are (from left to right) the training loss, test accuracy, and consensus error per communication round for the FashionMNIST datasets on LeNet5, comparing DAMSCo and DaSHCo with 5, 9 and 16 agents in a ring topology.