Thanks for all the valuable comments. Please check our responses below. We will address all minor comments.

Reviewer 1:

Q1: Please clarify the text and also state that W^t means W to the power of t (if I am correct!).

A: Thanks for the suggestion. You are absolutely correct! We will make it more clear in the revised version.

Reviewer 2:

Q1: Results do not seem to recover the single-machine guarantees when M=1 (only one node).

A: In the bound in Theorem 1, there are 3 terms. The only term that prevents from recovering the single-machine case is the third term (i.e., $\frac{48DL\sigma+\sigma^2}{\sqrt{mM}}$). This term completely comes from the procedure of bounding the LHS of (15) in the supplement, and is loose when $M=1$. However, thanks to the Assumption 1(iv), when $M=1$, the LHS of (15) is negative and disappears in the proof of Theorem 1. In terms of learning rate, when $M=1$, we have $\rho=0$, so $t=0$ and hence $\eta=O(1/L)$. Hence we can exactly recover [Theorem 1, 15], if we replace $\frac{48DL\sigma+\sigma^2}{\sqrt{mM}}$ by $\frac{48DL\sigma+\sigma^2}{\sqrt{mM}}\mathbf{1}_{M>1}$ with $\mathbf{1}$ being the indicator function in the statement of Theorem 1. We will change it in the revised version.

Q2: The claimed $O(\log(1/\varepsilon))$ communication complexity at the busiest node. If I am correct, this is the per-iteration complexity.

A: You are absolutely correct. We adopt the notion 'communication complexity on the busiest node' from [5, Table 1].

Q3: Parameter ρ depends on M, and in general tends to 1 as M grows (except from few topologies such as the complete graph). In Remark 2, it is noted that $\rho=1-O(1/M^2)$ for the ring graph. In this case, we roughly have that $\log(1/\rho)=O(1/M^2)$ and so it cannot be said that the convergence rate is logarithmic in ε.

A: Thanks for pointing it out. Indeed, for a fixed ring topology, ρ is close to 1 when M is large, and in this case the per-iteration communication complexity is no longer logarithmic. However, we want to emphasize that it is indeed logarithmic in ϵ when using *the random mixing strategy with a complete graph* as in Rand-DP-OAdam (line 281–283), in which any two nodes are connected and each node randomly selects two neighbors to communicate t times in each iteration. In this case, it is shown in [21] that $\mathbb{E}\|W_1\ldots W_t-\frac{1_M}{M}\|_2\leq\frac{\sqrt{M-1}}{(\sqrt{3})^t}$, to ensure that RHS$\leq\epsilon$, we only need $t=O(\log(1/\epsilon))$ when $M=poly(1/\epsilon)$. We make it more clear in the revised version.

Q4: In experiments, it seems that $t=1$ is used (which is not what theory suggests) but this is not clear.

A: In theory, since t is only a logarithmic term in ϵ, so it is almost a constant. In practice, we set $t=1$ and we did not incur any convergence problems. In addition, when $t=1$, the performance already matched the centralized synchronous version of our algorithm in terms of epochs, and it has much better run-time. We will make it more clear in revision.

Q5: Random mixing strategies are discussed in Remark 2, but Theorem 1 is presented for fixed matrices W.

A: Thanks for the suggestion. Using the random mixing strategy does not affect the proof of Theorem 1, since the two sources of randomness (gradient noise, random mixing) can be decoupled. We will mention it in revision.

Q6: Chebyshev acceleration (also called multi consensus) as in Scaman et al. [A].

A: Thanks for pointing it out. We have already cited Scaman et al. in reference [60]. We plan to consider it for random topologies in future work.

Reviewer 3:

Q1: Is there any difficulty in proving consensus of local iterates?

A: We can indeed prove the ϵ-consensus, since our algorithm allows t rounds of decentralized communication in each iteration and $t=\tilde{O}(\log(1/\epsilon))$. We will mention it in revision.

Q2: Please explain the meanings of barrier and lock-step.

A: In each iteration, a learner does not proceed until it finishes exchanging and averaging its weights with its neighbors. This data dependency forms an implicit barrier (i.e., we do not need to enforce an explicit barrier in the program) so that all the learners process the same number of iterations (i.e., mini-batches) at any given time (i.e., lock-step).

Reviewer 4:

Q1: The proposed algorithm and theoretical analysis are proposed, but they both are direct extensions of existing results, which significantly weaken the contribution.

A: We respectfully disagree. To handle this challenging nonconvex-nonconcave min-max problem, we have to design a novel algorithm such that (1) it is simple and user-friendly to large-scale decentralized training system; (2) it can be proved to have polynomial time complexity; (3) it is able to deliver good empirical performance in large-scale GAN training. Satisfying these requirements simultaneously is difficult. **It is *NOT* a direct extension of any existing results. Our algorithm is carefully designed and we conduct extensive and comprehensive empirical studies.** First, we use novel algorithm design: maintaining two update sequences, designing logarithmic communication rounds, and updating the discriminator and generator simultaneously. Second, in experiments, we consider both medium-scale (WGAN-GP on CIFAR10) and large-scale (SA-GAN on ImageNet) GAN training, with both high and low latency environment, and our proposed algorithms consistently deliver remarkable performance. It would be appreciated if the reviewer can provide us concrete references so that we can compare or argue against.