DIMAT: Decentralized Iterative Merging-And-Training for Deep Learning Models

Supplementary Material

1. Additional Analysis

In this section, we present additional analysis for completeness, primarily including the proof of all theorems presented in Section 4. Please note that the proof techniques for the proposed algorithms are different, while they share some similarities. For the analysis, we set the merging frequency n as 1 for a generic purpose.

1.1. Algorithmic Frameworks

DIMAT-ADAM is slightly different from its centralized counterpart due to an auxiliary variable $\hat{\mathbf{u}}_k^i$. Based on a recent work [9], the direct extension of Adam presented in [36] may not necessarily converge to a stationary point. r_k in Line 4 of Algorithm 3 can take different forms, leading to different variants such as AMSGrad. In this work, we will primarily investigate the convergence rate of DIMAT-AMSGRAD. $\oslash$ represents the division between two vectors.

Algorithm 2: DIMAT-MSGD

Input : mixing matrix $\mathbf{\Pi}$, the # of epochs K, initialization $\mathbf{x}_1^i$, $\mathbf{v}_1^i$, step size α, $0 \leq \beta < 1$, merging frequency n

Output: $\bar{\mathbf{x}}_K = \frac{1}{N}\sum_{i=1}^N \mathbf{x}_K^i$

1 **for** *k in* $1:K$ **do**
2 **for** *each agent* $i \in \mathcal{V}$ **do**
3 Calculate the stochastic gradient $\mathbf{g}_k^i$;
4 **if** *k mod n=0* **then**
5 $\mathbf{x}_{k+1/2}^i = \sum_{j\in Nb(i)} \pi_{ij}\mathbf{P}_k^{ij}\mathbf{x}_k^j$;
6 **else**
7 $\mathbf{x}_{k+1/2}^i = \mathbf{x}_k^i$;
8 $\mathbf{v}_{k+1}^i = \beta\mathbf{v}_k^i - \alpha\mathbf{g}_k^i$;
9 $\mathbf{x}_{k+1}^i = \mathbf{x}_{k+1/2}^i + \mathbf{v}_{k+1}^i$;

1.2. Additional Theoretical Results

Theorem 3. *Let Assumptions 1 and 3 hold. If the step size* $\alpha \leq \min\{\frac{(1-\sqrt{\rho'})(1-\beta)}{4L}, \frac{(1-\sqrt{\rho'})^2(1-\beta)^2}{6L}\}$ *in Algorithm 2, then for all* $K \geq 1$*, the following relationship holds true:*

$$\frac{1}{K}\sum_{k=1}^K \mathbb{E}[\|\nabla f(\bar{\mathbf{x}}_k)\|^2] \leq \frac{2(1-\beta)(f(\bar{\mathbf{x}}_0)-f^*)}{\alpha K} + \frac{L\alpha\sigma^2}{(1-\beta)^2 N} + \frac{4\alpha^2\sigma^2L^2}{(1-\beta)^2(1-\rho')} + \frac{16\alpha^2\kappa^2L^2}{(1-\beta)^2(1-\sqrt{\rho'})^2}, \tag{10}$$

where $\bar{\mathbf{x}}_k = \frac{1}{N}\sum_{i=1}^N \mathbf{x}_k^i$.

Corollary 2. *Let Assumptions 1 and 3 hold. If step size* $\alpha = \mathcal{O}(\sqrt{\frac{N}{K}})$ *in Algorithm 2, then for all* $K \geq \max\{\frac{32NL^2}{(1-\beta)^2(1-\sqrt{\rho'})^2}, \frac{36NL^2}{(1-\beta)^4(1-\sqrt{\rho'})^4}\}$*, we have*

$$\frac{1}{K}\sum_{k=1}^K \mathbb{E}[\|\nabla f(\bar{\mathbf{x}}_k)\|^2] \leq \mathcal{O}(\sqrt{\frac{1}{NK}} + \frac{N}{(1-\rho')K} + \frac{N}{(1-\sqrt{\rho'})^2 K}). \tag{11}$$

Algorithm 3: DIMAT-ADAM

Input : mixing matrix $\mathbf{\Pi}$, the # of epochs K, initialization $\mathbf{x}_1$, $\mathbf{m}_0^i = \mathbf{v}_0^i = 0$, $\hat{\mathbf{u}}_1^i = \mathbf{v}_1^i$, step size α, merging frequency n, small positive constant ϵ, $\beta_1 \in [0,1)$

Output: $\bar{\mathbf{x}}_K = \frac{1}{N}\sum_{i=1}^N \mathbf{x}_K^i$

1 **for** *k in* $1:K$ **do**
2 **for** *each agent* $i \in \mathcal{V}$ **do**
3 Calculate the stochastic gradient $\mathbf{g}_k^i$;
4 $\mathbf{m}_k^i = \beta_1\mathbf{m}_{k-1}^i + (1-\beta_1)\mathbf{g}_k^i$;
5 $\mathbf{v}_k^i = r_k(\mathbf{g}_1^i, ..., \mathbf{g}_k^i)$;
6 **if** *k mod n=0* **then**
7 $\mathbf{x}_{k+1/2}^i = \sum_{j\in Nb(i)} \pi_{ij}\mathbf{P}_k^{ij}\mathbf{x}_k^j$;
8 $\hat{\mathbf{u}}_{k+1/2}^i = \sum_{j\in Nb(i)} \pi_{ij}\mathbf{P}_k^{ij}\hat{\mathbf{u}}_k^j$;
9 **else**
10 $\mathbf{x}_{k+1/2}^i = \mathbf{x}_k^i$;
11 $\hat{\mathbf{u}}_{k+1/2}^i = \hat{\mathbf{u}}_k^i$;
12 $\mathbf{u}_k^i = \max(\hat{\mathbf{u}}_k^i, \epsilon)$;
13 $\mathbf{x}_{k+1}^i = \mathbf{x}_{k+1/2}^i - \alpha\mathbf{m}_k^i \oslash (\mathbf{u}_k^i)^{1/2}$;
14 $\hat{\mathbf{u}}_{k+1}^i = \hat{\mathbf{u}}_{k+1/2}^i - \mathbf{v}_{k-1}^i + \mathbf{v}_k^i$;

Before presenting the main result for DMM-ADAM, we define specifically r_k in Algorithm 3 as $\hat{\mathbf{v}}_k^i = \beta_2\hat{\mathbf{v}}_{k-1}^i + (1-$

$\beta_2)\mathbf{g}_k^i \odot \mathbf{g}_k^i$ and $\mathbf{v}_k^i = \max\{\hat{\mathbf{v}}_k^i, \mathbf{v}_{k-1}^i\}$, where $0 \leq \beta_2 < 1$ and $\hat{\mathbf{v}}_0^i = 0$, leading to DIMAT-AMSGRAD. To show the convergence rate, we need another assumption specifically for the adaptive gradient descent type of algorithms, which bounds the infinity norms of $\mathbf{g}_k^i$ and $\nabla f_i(\mathbf{x}_k^i)$ by a positive constant $G_\infty < \infty$. This assumption has actually been relaxed in many first-order methods such as SGD and MSGD types [58]. However, the relaxation of the assumption in adaptive gradient methods is out of our scope and we will still proceed with this assumption.

Theorem 4. *Let Assumptions 1 and 3 hold. Also suppose that* $\|\mathbf{g}_k^i\|_\infty \leq G_\infty$ *and that* $\|\nabla f^i(\mathbf{x}_k^i)\|_\infty \leq G_\infty$ *for all* $i \in \mathcal{V}$ *and* $k \geq 1$. *If step size* $\alpha = \mathcal{O}(\frac{1}{\sqrt{Kd}})$ *in Algorithm 3, then for all* $K \geq \frac{256L^2}{d\epsilon}$, *we have,*

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}[\|\nabla f(\bar{\mathbf{x}}_k)\|^2] \leq \mathcal{O}(\frac{d^{1.5}}{\sqrt{NK}} + \frac{dN}{(1-\sqrt{\rho'})^2 K} + \frac{N^{1.5}d^{0.5}}{K^{1.5}} + \frac{\sqrt{N}d^2}{(1-\sqrt{\rho'})K} + \frac{Nd^{1.5}}{(1-\sqrt{\rho'})K^{1.5}}) \tag{12}$$

where $\bar{\mathbf{x}}_k = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_k^i$.

Theorem 4 shows that Algorithm 3 converges with a rate of $\mathcal{O}(\frac{d^{1.5}}{\sqrt{NK}})$ when K is sufficiently large. Also, DIMAT-AMSGRAD enjoys the linear speed up as DIMAT-SGD and DIMAT-MSGD. The dependence on the dimension of $\mathbf{x}$ is attributed to the bounded assumption of the infinity norms of gradients. The interaction between topology and model merging from Remark 2 can comparably apply to DIMAT-AMSGRAD in this context. The transition between the transient and stable regimes depending on when $\mathcal{O}(\frac{d^{1.5}}{\sqrt{NK}})$ dominates also motivates the further future investigation of convergence dynamics.

1.3. Additional Analysis for DIMAT-SGD

With abuse of notation, we use some upper bold characters to represent vectors after they are expanded. Define

$$\mathbf{X}_k = [\mathbf{x}_k^1; \mathbf{x}_k^2; ...; \mathbf{x}_k^N]^\top \in \mathbb{R}^{dN},$$
$$\mathbf{G}_k = [\mathbf{g}_k^1; \mathbf{g}_k^2; ...; \mathbf{g}_k^N]^\top \in \mathbb{R}^{dN},$$
$$\mathbf{H}_k = [\nabla f^1(\mathbf{x}_k^1); \nabla f^2(\mathbf{x}_k^2); ...; \nabla f^N(\mathbf{x}_k^N))]^\top \in \mathbb{R}^{dN},$$
$$\mathbf{Q} = \frac{1}{dN}\mathbf{1}\mathbf{1}_{dN}^\top \in \mathbb{R}^{dN\times dN}$$

Without loss of generality, suppose that the initialization $\mathbf{X}_0 = \mathbf{0}$ throughout the rest of analysis. For DIMAT-SGD, we have

$$\mathbf{X}_k = -\alpha\sum_{\tau=1}^{k-1}\prod_{t=\tau+1}^{k-1}\mathbf{W}\mathbf{P}_t\mathbf{G}_\tau \tag{13}$$

For ease of exposition, we define $\prod_{\tau=k+1}^{k}\mathbf{W}\mathbf{P}_\tau = \mathbf{I}$ in our analysis. Left multiplying by $\mathbf{I}-\mathbf{Q}$ yields the following relationship

$$(\mathbf{I}-\mathbf{Q})\mathbf{X}_k = -\alpha\sum_{\tau=1}^{k-1}(\mathbf{I}-\mathbf{Q})\prod_{t=\tau+1}^{k-1}\mathbf{W}\mathbf{P}_t\mathbf{G}_\tau, \tag{14}$$

which will serve to characterize the optimal error bound. By taking the squared norm and expectation on both sides, we have

$$\mathbb{E}[\|(\mathbf{I}-\mathbf{Q})\mathbf{X}_k\|^2] = \alpha^2\mathbb{E}[\|\sum_{\tau=1}^{k-1}(\mathbf{I}-\mathbf{Q})\prod_{t=\tau+1}^{k-1}\mathbf{W}\mathbf{P}_t\mathbf{G}_\tau\|^2]. \tag{15}$$

The left side of above equation is equivalent to $\mathbb{E}[\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{x}_k^i - \bar{\mathbf{x}}_k\|^2]$. To further analyze the Eq. 15, we investigate its right side in the following.

$$\begin{aligned}&\alpha^2\mathbb{E}[\|\sum_{\tau=1}^{k-1}(\mathbf{I}-\mathbf{Q})\prod_{t=\tau+1}^{k-1}\mathbf{W}\mathbf{P}_t\mathbf{G}_\tau\|^2] \leq \\ &\underbrace{2\alpha^2\,\mathbb{E}[\|\sum_{\tau=1}^{k-1}(\mathbf{I}-\mathbf{Q})\prod_{t=\tau+1}^{k-1}\mathbf{W}\mathbf{P}_t(\mathbf{G}_\tau-\mathbf{H}_\tau)\|^2]}_{T_1} \\ &+\underbrace{2\alpha^2\,\mathbb{E}[\|\sum_{\tau=1}^{k-1}(\mathbf{I}-\mathbf{Q})\prod_{t=\tau+1}^{k-1}\mathbf{W}\mathbf{P}_t\mathbf{H}_\tau\|^2]}_{T_2},\end{aligned} \tag{16}$$

which follows by using the basic inequality $\|\mathbf{a}+\mathbf{b}\|^2 \leq 2\|\mathbf{a}\|^2 + 2\|\mathbf{b}\|^2$. We will next study the upper bounds for T_1 and T_2, respectively. Before that, we present some technical detail for how to derive $\rho' \leq \rho$ and then state two key lemmas to manipulate $(\mathbf{I}-\mathbf{Q})\prod_{\tau=1}^{k}\mathbf{W}\mathbf{P}_\tau$ and $\mathbf{G}_k - \mathbf{H}_k$.

Analysis of $\rho' \leq \rho$. As $\mathbf{W}\mathbf{P}_k$ is symmetric, the immediate outcome is that the singular values of $\mathbf{W}\mathbf{P}_k$ are equal to the absolute values of eigenvalues of $\mathbf{W}\mathbf{P}_k$, which results in $\zeta_l(\mathbf{W}\mathbf{P}_k) = |\lambda_l(\mathbf{W}\mathbf{P}_k)|$, where ζ_l is the l-th singular value of $\mathbf{W}\mathbf{P}_k$. This result is well-known and we skip the proof in this context. According to the Courant–Fischer–Weyl Min-Max Principle [23], the following relationship can be obtained:

$$\begin{aligned}&\zeta_l(\mathbf{W}\mathbf{P}_k) = \max_{S:dim(S)=l}\min_{x\in S,\|x\|=1}\|\mathbf{W}\mathbf{P}_k x\| \\ &\leq \max_{S:dim(S)=l}\min_{x\in S,\|x\|=1}\|\mathbf{W}\|\|\mathbf{P}_k x\| \\ &= \zeta_1(\mathbf{W})\cdot\max_{S:dim(S)=l}\min_{x\in S,\|x\|=1}\|\mathbf{P}_k x\| \\ &\leq \zeta_1(\mathbf{W})\zeta_i(\mathbf{P}_k),\end{aligned} \tag{17}$$

where $S : dim(S) = l$ is a subspace of $\mathbb{R}^{dN}$ of dimension l. Then,

$$\begin{aligned}&\zeta_l(\mathbf{W}\mathbf{P}_k) = \zeta_l([\mathbf{W}\mathbf{P}_k]^\top) \\ &= \zeta_l(\mathbf{P}_k^\top\mathbf{W}^\top) \leq \zeta_1(\mathbf{P}_k^\top)\zeta_l(\mathbf{W}^\top) = \zeta_l(\mathbf{W})\zeta_1(\mathbf{P}_k).\end{aligned} \tag{18}$$

We have known that $\mathbf{W}$, $\mathbf{P}_k$ and $\mathbf{W}\mathbf{P}_k$ are symmetric such that

$$|\lambda_l(\mathbf{W}\mathbf{P}_k)| \leq |\lambda_l(\mathbf{W})||\lambda_1(\mathbf{P}_k)| \tag{19}$$

Since all eigenvalues of $\mathbf{P}_k$ are contained in the roots of unity, the modulus of any eigenvalue of $\mathbf{P}_k$ is 1, i.e., $|\lambda_1(\mathbf{P}_k)| = 1$. With this in hand, we have

$$|\lambda_l(\mathbf{W}\mathbf{P}_k)| \leq |\lambda_l(\mathbf{W})| \tag{20}$$

The above inequality implies that

$$\begin{aligned} &\max\{|\lambda_2(\mathbf{W}\mathbf{P}_k)|, |\lambda_{dN}(\mathbf{W}\mathbf{P}_k)|\} \leq \\ &\max\{|\lambda_2(\mathbf{W})|, |\lambda_{dN}(\mathbf{W})|\}, \end{aligned} \tag{21}$$

which ensures the fact that $\sqrt{\rho'} \leq \sqrt{\rho}$.

Lemma 1. *Let Assumption 1 hold. Suppose that* $\mathbf{E}[\mathbf{g}^i] = \nabla f^i(\mathbf{x}^i), \forall i \in \mathcal{V}$. *Then, we have the following relationship*

$$\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\mathbf{g}^i\|^2] \leq \frac{1}{N}\sigma^2 + \mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}^i)\|^2]. \tag{22}$$

The proof for the Lemma 1 follows similarly Lemma 1 in [58] and we skip it in this context.

Lemma 2. *Let Assumption 3 hold. Then, for any integer* $k \geq 1$, *we have* $\|(\mathbf{I}-\mathbf{Q})\prod_{\tau=1}^{k}\mathbf{W}\mathbf{P}_\tau\| \leq (\sqrt{\rho'})^k < 1$, *where* $\|\cdot\|$ *denotes the spectral norm in this context.*

Proof. The proof can be easily obtained by using Assumption 3 and the similar analysis techniques in Lemma IV.2 in [7]. □

We are now able to bound T_1 and T_2. By following the similar proof techniques and adapting the analysis in [58], the following bounds are obtained accordingly.

$$T_1 \leq \frac{N\sigma^2}{1-\rho'} \tag{23}$$

$$\begin{aligned} T_2 \leq\ & \frac{1}{1-\sqrt{\rho'}}[(8L^2\sum_{\tau=1}^{k}(\rho')^{(k-\tau)/2}\mathbb{E}[\|(\mathbf{I}-\mathbf{Q})\mathbf{X}_\tau\|^2] \\ &+ 4N\sum_{\tau=1}^{k}(\rho')^{(k-\tau)/2}\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_\tau^i)\|^2]] \\ &+ \frac{4N\kappa^2}{1-\sqrt{\rho'}} \end{aligned} \tag{24}$$

Based on the upper bounds for T_1 and T_2, we can obtain the upper bound for $\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[\|\mathbf{x}_k^i - \bar{\mathbf{x}}_k\|^2]$.

Lemma 3. *Let Assumptions 1 and 3 hold. For* $\mathbf{x}_k^i$ *defined by Algorithm 1, if step size* $\alpha \leq \frac{1-\sqrt{\rho'}}{4\sqrt{2}L}$, *then* $\forall K \geq 1$, *the following relationship holds true*

$$\begin{aligned} &\sum_{k=1}^{K}\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[\|\mathbf{x}_k^i - \bar{\mathbf{x}}_k\|^2] \leq \frac{4K\alpha^2\sigma^2}{1-\rho'} \\ &+ \frac{16\alpha^2}{(1-\sqrt{\rho'})^2}\sum_{k=1}^{K}\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2] \\ &+ \frac{16K\alpha^2\kappa^2}{(1-\sqrt{\rho'})^2}, \end{aligned} \tag{25}$$

where $\bar{\mathbf{x}}_k = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_k^i$.

Combining Eqs 16, 23, and 24 and summing k over $\{1, 2, ..., K\}$, and following the proof techniques from [58] can complete the proof for Lemma 3. With Lemma 3 in hand, we are ready to give the proof of Theorem 2 in the following.

Proof. We start with the descent inequality given by the smoothness of f such that

$$\begin{aligned} &\mathbb{E}[f(\bar{\mathbf{x}}_{k+1})] \leq \mathbb{E}[f(\bar{\mathbf{x}}_k)] + \mathbb{E}[\langle\nabla f(\bar{\mathbf{x}}_k), \bar{\mathbf{x}}_{k+1} - \bar{\mathbf{x}}_k\rangle] \\ &+ \frac{L}{2}\mathbb{E}[\|\bar{\mathbf{x}}_{k+1} - \bar{\mathbf{x}}_k\|^2] \end{aligned} \tag{26}$$

We first process the second term on the right side of above inequality. Replacing $\bar{\mathbf{x}}_{k+1} - \bar{\mathbf{x}}_k$ with $-\alpha\frac{1}{N}\sum_{i=1}^{N}\mathbf{g}^i(\mathbf{x}_k^i)$ allows us to study $-\alpha\mathbb{E}[\langle\nabla f(\bar{\mathbf{x}}_k), \frac{1}{N}\sum_{i=1}^{N}\mathbf{g}^i(\mathbf{x}_k^i)\rangle]$. Thus, we have

$$\begin{aligned} &\langle\nabla f(\bar{\mathbf{x}}_k), \frac{1}{N}\sum_{i=1}^{N}\mathbf{g}^i(\mathbf{x}_k^i)\rangle = \frac{1}{2}(\|\nabla f(\bar{\mathbf{x}}_k)\|^2 + \\ &\|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2 - \|\nabla f(\bar{\mathbf{x}}_k) - \frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2) \\ &\geq \frac{1}{2}(\|\nabla f(\bar{\mathbf{x}}_k)\|^2 + \|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2 \\ &- L^2\frac{1}{N}\sum_{i=1}^{N}\|\bar{\mathbf{x}}_k - \mathbf{x}_k^i\|^2). \end{aligned} \tag{27}$$

The last inequality follows from the smoothness assumption. Therefore, we have

$$\begin{aligned} &\mathbb{E}[\langle\nabla f(\bar{\mathbf{x}}_k), \bar{\mathbf{x}}_{k+1} - \bar{\mathbf{x}}_k\rangle] \leq -\frac{\alpha}{2}\mathbb{E}[\|\nabla f(\bar{\mathbf{x}}_k)\|^2 \\ &+ \|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2] + \frac{\alpha L^2}{2}\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[\|\bar{\mathbf{x}}_k - \mathbf{x}_k^i\|^2]. \end{aligned} \tag{28}$$

With Eq. 26, the following relationship can be obtained.

$$\begin{aligned}\mathbb{E}[f(\bar{\mathbf{x}}_{k+1})] &\leq \mathbb{E}[f(\bar{\mathbf{x}}_k)] - \frac{\alpha}{2}\mathbb{E}[\|\nabla f(\bar{\mathbf{x}}_k)\|^2 \\ &+ \|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2] + \frac{\alpha L^2}{2}\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[\|\bar{\mathbf{x}}_k - \mathbf{x}_k^i\|^2] \\ &+ \frac{L}{2}\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\mathbf{g}_k^i\|^2].\end{aligned} \tag{29}$$

The last inequality holds due to $\bar{\mathbf{x}}_{k+1} - \bar{\mathbf{x}}_k = \frac{1}{N}\sum_{i=1}^{N}\mathbf{g}_k^i$. With Lemma 1 and some mathematical manipulations, we have

$$\begin{aligned}\mathbb{E}[f(\bar{\mathbf{x}}_{k+1})] &\leq \mathbb{E}[f(\bar{\mathbf{x}}_k)] - \frac{\alpha}{2}\mathbb{E}[\|\nabla f(\bar{\mathbf{x}}_k)\|^2 \\ &+ \|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2] + \frac{\alpha L^2}{2}\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[\|\bar{\mathbf{x}}_k - \mathbf{x}_k^i\|^2] \\ &+ \frac{L\alpha^2}{2}(\frac{\sigma^2}{N} + \mathbb{E}[\frac{1}{N}\sum_{i=1}^{N}\|\nabla f^i(\mathbf{x}_k^i)\|^2]) \\ &= \mathbb{E}[f(\bar{\mathbf{x}}_k)] - \frac{\alpha}{2}\mathbb{E}[\|\nabla f(\bar{\mathbf{x}}_k)\|^2] - \\ &(\frac{\alpha}{2} - \frac{L\alpha^2}{2})\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2] \\ &+ \frac{\alpha L^2}{2N}\sum_{i=1}^{N}\mathbb{E}[\|\bar{\mathbf{x}}_k - \mathbf{x}_k^i\|^2] + \frac{L\alpha^2\sigma^2}{2N},\end{aligned} \tag{30}$$

which implies the following inequality

$$\begin{aligned}\mathbb{E}[\|\nabla f(\bar{\mathbf{x}}_k)\|^2] &\leq \frac{2}{\alpha}(\mathbb{E}[f(\bar{\mathbf{x}}_k)] - \mathbb{E}[f(\bar{\mathbf{x}}_{k+1})]) \\ &- (1 - L\alpha)\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2] + \frac{L^2}{N}\mathbb{E}[\|\bar{\mathbf{x}}_k - \mathbf{x}_k^i\|^2] \\ &+ \frac{L\alpha\sigma^2}{N}.\end{aligned} \tag{31}$$

The above relationship is obtained by dividing $\alpha/2$ on both sides. Summing k over $\{1, 2, ..., K\}$ yields

$$\begin{aligned}&\sum_{k=1}^{K}\mathbb{E}[\|\nabla f(\bar{\mathbf{x}}_k)\|^2] \leq \frac{2}{\alpha}(f(\bar{\mathbf{x}}_0) - \mathbb{E}[f(\bar{\mathbf{x}}_K)]) \\ &- (1 - L\alpha)\sum_{k=1}^{K}\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2] \\ &+ L^2(\frac{4K\alpha^2\sigma^2}{1-\rho'} + \frac{16\alpha^2}{(1-\sqrt{\rho'})^2}\sum_{k=1}^{K}\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2] \\ &+ \frac{16K\alpha^2\kappa^2}{(1-\sqrt{\rho'})^2}) + \frac{L\alpha\sigma^2 K}{N} \\ &\leq \frac{2}{\alpha}(f(\bar{\mathbf{x}}_0) - f^*) - \\ &(1 - L\alpha - \frac{16L^2\alpha^2}{(1-\sqrt{\rho'})^2})\sum_{k=1}^{K}\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2] \\ &+ \frac{4KL^2\alpha^2\sigma^2}{1-\rho'} + \frac{16KL^2\alpha^2\kappa^2}{(1-\sqrt{\rho'})^2} + \frac{LK\alpha\sigma^2}{N}\end{aligned} \tag{32}$$

The last inequality is attained by substituting the conclusion from Lemma 3 into Eq. 31. Due to the condition for the step size α, we know that $1 - L\alpha - \frac{16L^2\alpha^2}{(1-\sqrt{\rho'})^2} \geq 0$, which would simplify the right side in the last inequality. Hence,

$$\begin{aligned}\sum_{k=1}^{K}\mathbb{E}[\|\nabla f(\bar{\mathbf{x}}_k)\|^2] &\leq \frac{2}{\alpha}(f(\bar{\mathbf{x}}_0) - f^*) + \frac{4KL^2\alpha^2\sigma^2}{1-\rho'} \\ &+ \frac{16KL^2\alpha^2\kappa^2}{(1-\sqrt{\rho'})^2} + \frac{LK\alpha\sigma^2}{N}\end{aligned} \tag{33}$$

The desirable result is obtained by dividing K on both sides. □

The proof for Corollary 1 is easily completed by substituting the step size $\alpha = \mathcal{O}(\sqrt{\frac{N}{K}})$ into the error bound in Theorem 2.

1.4. Additional Analysis for DIMAT-MSGD

To prove Theorem 3, we need another auxiliary variable to assist in establishing the relationship between two consecutive steps of $\bar{\mathbf{x}}$. By multiplying $\frac{1}{N}\mathbf{1}\mathbf{1}^\top$ on

$$\mathbf{v}_{k+1}^i = \beta\mathbf{v}_k^i - \alpha\mathbf{g}_k^i, \mathbf{x}_{k+1}^i = \mathbf{x}_{k+1/2}^i + \mathbf{v}_{k+1}^i,$$

we obtain

$$\begin{aligned}\bar{\mathbf{v}}_{k+1} &= \beta\bar{\mathbf{v}}_k - \alpha\frac{1}{N}\sum_{i=1}^{N}\mathbf{g}_k^i \\ \bar{\mathbf{x}}_{k+1} &= \bar{\mathbf{x}}_k + \bar{\mathbf{v}}_{k+1},\end{aligned} \tag{34}$$

which follows by the approximate equivalence between averaged permuted parameters and averaged parameters in Remark 1. To characterize the analysis, we define an auxiliary variable in the following

$$\bar{\mathbf{p}}_k := \frac{1}{1-\beta}\bar{\mathbf{x}}_k - \frac{\beta}{1-\beta}\bar{\mathbf{x}}_{k-1}, \tag{35}$$

where $k \geq 1$. If $k = 0$, then $\bar{\mathbf{p}}_0 = \bar{\mathbf{x}}_0 = 0$. The following lemma states the relationship between $\bar{\mathbf{p}}_{k+1}$ and $\bar{\mathbf{p}}_k$.

Lemma 4. *Let $\bar{\mathbf{p}}_k$ be defined in Eq. 35. Based on Algorithm 2, we have*

$$\bar{\mathbf{p}}_{k+1} - \bar{\mathbf{p}}_k = -\frac{\alpha}{(1-\beta)N}\sum_{i=1}^{N}\mathbf{g}_k^i. \tag{36}$$

The proof of Lemma 4 follows similarly from Lemma 3 in [17]. We next study the relationship between $\bar{\mathbf{p}}_k$ and $\bar{\mathbf{x}}_k$.

Lemma 5. *Based on Algorithm 2 and $\bar{\mathbf{p}}_k$ in Eq. 35, the following relationship holds true for all $K \geq 1$*

$$\sum_{k=1}^{K}\|\bar{\mathbf{p}}_k - \bar{\mathbf{z}}_k\| \leq \frac{\alpha^2\beta^2}{(1-\beta)^4}\sum_{k=1}^{K}\|\frac{1}{N}\sum_{i=1}^{N}\mathbf{g}_k^i\|^2. \tag{37}$$

The proof of Lemma 5 can be adapted from that of Lemma 4 in [17]. Next, we define

$$\mathbf{V}_k = [\mathbf{v}_k^1; \mathbf{v}_k^2; ...; \mathbf{v}_k^N]^\top \in \mathbb{R}^{dN}.$$

To prove Theorem 3, we need to get the upper bound for $\mathbb{E}[\|(\mathbf{I}-\mathbf{Q})\mathbf{X}_k\|^2]$ first, as done for DIMAT-SGD. Hence, we proceed with expanding $\|(\mathbf{I}-\mathbf{Q})\mathbf{X}_k\|^2$. Recursively applying $\mathbf{v}_{k+1}^i = \beta\mathbf{v}_k^i - \alpha\mathbf{g}_k^i$ and set $\mathbf{V}_0 = 0$ yields

$$\mathbf{V}_k = -\alpha\sum_{\tau=0}^{k-1}\beta^{k-1-\tau}\mathbf{G}_\tau \tag{38}$$

With $\mathbf{X}_k = \mathbf{W}\mathbf{P}_{k-1}\mathbf{X}_{k-1} + \mathbf{V}_k$, recursively applying it and using 0 initial condition attains

$$\mathbf{X}_k = \sum_{\tau=1}^{k}\prod_{t=\tau+1}^{k}\mathbf{W}\mathbf{P}_t\mathbf{V}_\tau. \tag{39}$$

Substituting Eq. 38 into Eq. 39 produces the following relationship:

$$\begin{aligned}
\mathbf{X}_k &= -\alpha\sum_{\tau=1}^{k}\prod_{t=\tau+1}^{k}\mathbf{W}\mathbf{P}_t\sum_{o=0}^{\tau-1}\beta^{\tau-1-o}\mathbf{G}_o \\
&= -\alpha\sum_{\tau=1}^{k}\sum_{o=0}^{\tau-1}\beta^{\tau-1-o}\prod_{t=\tau+1}^{k}\mathbf{W}\mathbf{P}_t\mathbf{G}_o \\
&= -\alpha\sum_{c=0}^{k-1}[\sum_{l=c+1}^{k}\beta^{l-1-c}]\prod_{t=c+1}^{k-1}\mathbf{W}\mathbf{P}_t\mathbf{G}_c \\
&= -\alpha\sum_{\tau=0}^{k-1}\frac{1-\beta^{k-\tau}}{1-\beta}\prod_{t=\tau+1}^{k-1}\mathbf{W}\mathbf{P}_t\mathbf{G}_\tau.
\end{aligned} \tag{40}$$

Multiplying $\mathbf{I}-\mathbf{Q}$ on both sides yields

$$(\mathbf{I}-\mathbf{Q})\mathbf{X}_k = -\alpha\sum_{\tau=0}^{k-1}\frac{1-\beta^{k-\tau}}{1-\beta}(\mathbf{I}-\mathbf{Q})\prod_{t=\tau+1}^{k-1}\mathbf{W}\mathbf{P}_t\mathbf{G}_\tau \tag{41}$$

It is observed that the above equality has the similar form of Eq. 14 and we process it as done in Eq. 16. With Lemma 2 in hand, we present a key lemma for assisting in the proof of Theorem 3.

Lemma 6. *Let Assumptions 1 and 3 hold. For $\{\bar{\mathbf{x}}_k\}$ defined in Algorithm 2, if $\alpha \leq \frac{(1-\beta)(1-\sqrt{\rho})}{4\sqrt{2}L}$, then for all $K \geq 1$, we have*

$$\begin{aligned}
\sum_{k=1}^{K}\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[\|\bar{\mathbf{x}}_k - \mathbf{x}_k^i\|^2] &\leq \frac{4\alpha^2\sigma^2 K}{(1-\beta)^2(1-\rho')} \\
&+ \frac{16\alpha^2}{(1-\sqrt{\rho'})^2(1-\beta)^2}\sum_{k=1}^{K}\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N}\nabla f^i(\mathbf{x}_k^i)\|^2] \\
&+ \frac{16K\alpha^2\kappa^2}{(1-\sqrt{\rho'})^2(1-\beta)^2},
\end{aligned} \tag{42}$$

where $\bar{\mathbf{x}}_k = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_k^i$.

The proof of Lemma 6 follows from the proof of Lemma 1 in [17] and Lemma 11 in [58]. With this in hand, we are now ready to prove Theorem 3. The proof techniques are quite similar as used in showing Theorem 2. Specifically, we apply the smoothness condition and conclusion from Lemma 6 to arrive at the conclusion. The proof also follows similarly from Theorem 1 in [17] and Theorem 3 in [58]. The proof for Corollary 2 is immediately shown by substituting the step size into the conclusion from Theorem 3.

1.5. Additional Analysis for DIMAT-AMSGRAD

The proof for Theorem 4 is fairly non-trivial and technical. In the proof, we need to use an auxiliary sequence as $\bar{\mathbf{p}}_k$ defined before. Therefore, we utilize the same auxiliary variable in the proof. Similarly, we next establish the relationship between $\bar{\mathbf{p}}_k$ and $\bar{\mathbf{p}}_{k+1}$. Please note that in the analysis, we may use different notations specified for the convenience of analysis.

Lemma 7. *For the sequence defined in Eq. 35, through Algorithm 3, we have the following relationship*

$$\begin{aligned}
\bar{\mathbf{p}}_{k+1} - \bar{\mathbf{p}}_k = \alpha\frac{\beta_1}{1-\beta_1}\frac{1}{N}\mathbf{m}_k^i \oslash ((\mathbf{u}_{k-1}^i)^{1/2} - (\mathbf{u}_k^i)^{1/2}) \\
- \alpha\frac{1}{N}\sum_{i=1}^{N}\mathbf{g}_k^i \oslash (\mathbf{u}_k^i)^{1/2}
\end{aligned} \tag{43}$$

The proof follows similarly from Lemma A.1 in [9], Due to the $\max(\cdot;\cdot)$ function in the update law, handling such a function can impose difficulties in the proof. Therefore, we present a lemma to pave the way.

Lemma 8. *Define a set of numbers, $c_1, c_2, ..., c_n \in \mathbb{R}$ and denote their mean by $\bar{c} = \frac{1}{n}\sum_{i=1}^{n} c_i$. Define $h_i(r) := \max\{c_i, r\}$ and $\bar{h}(r) = \frac{1}{n}\sum_{i=1}^{n} h_i(r)$. For any r and r' with $r' \geq r$, we have*

$$\sum_{i=1}^{n} |h_i(r) - \bar{h}_i(r)| \leq \sum_{i=1}^{n} |h_i(r') - \bar{h}_i(r')|, \tag{44}$$

and when $r \leq \min_{i\in[n]} c_i$, we have

$$\sum_{i=1}^{n} |h_i(r) - \bar{h}_i(r)| = \sum_{i=1}^{n} |c_i - \bar{c}|. \tag{45}$$

With the above two lemmas, we are now ready to show the detailed proof for Theorem 4. Please note that the proof techniques follow from the majority of proof of Theorems 2 and 3 in [9]. However, the significant difference is to incorporate the permutation matrix $\mathbf{P}$ into the update law such that it leads to the impact of the spectral gap on the error bounds. We will next arrive at this with the derivation. We first define two auxiliary variables:

$$\mathbf{M}_k = [\mathbf{m}_k^1; \mathbf{m}_k^2; ...; \mathbf{m}_k^N]^\top \in \mathbb{R}^{dN},$$

and

$$\mathbf{U}_k = [\mathbf{u}_k^1; \mathbf{u}_k^2; ...; \mathbf{u}_k^N]^\top \in \mathbb{R}^{dN}$$

Based on Algorithm 3, we have

$$\mathbf{X}_k = \mathbf{W}\mathbf{P}_{k-1}\mathbf{X}_{k-1} - \alpha \mathbf{M}_{k-1} \oslash \mathbf{U}_{k-1}^{1/2}. \tag{46}$$

Recursively applying the above equation yields

$$\mathbf{X}_k = \prod_{\tau=1}^{k-1} \mathbf{W}\mathbf{P}_\tau \mathbf{X}_1 - \alpha \sum_{\tau=1}^{k-1} \prod_{t=\tau+1}^{k-1} \mathbf{W}\mathbf{P}_t \mathbf{M}_\tau \oslash \mathbf{U}_\tau^{1/2} \tag{47}$$

Setting 0 initial condition and multiplying by $\mathbf{I} - \mathbf{Q}$ on both sides attains the following relationship:

$$(\mathbf{I} - \mathbf{Q})\mathbf{X}_k = -\alpha \sum_{\tau=1}^{k-1} (\mathbf{I} - \mathbf{Q}) \prod_{t=\tau+1}^{k-1} \mathbf{W}\mathbf{P}_t \mathbf{M}_\tau \oslash \mathbf{U}_\tau^{1/2} \tag{48}$$

We then calculate its squared norm and take the expectation to get the similar equation as Eq. 15.

$$\mathbb{E}[\|(\mathbf{I} - \mathbf{Q})\mathbf{X}_k\|^2] = \alpha^2 \mathbb{E}[\| \textstyle\sum_{\tau=1}^{k-1} (\mathbf{I} - \mathbf{Q}) \prod_{t=\tau+1}^{k-1} \mathbf{W}\mathbf{P}_t \mathbf{M}_\tau \oslash \mathbf{U}_\tau^{1/2} \|^2].$$

For DIMAT-SGD and DIMAT-MSGD, to acquire the upper bound of $\mathbb{E}[\|(\mathbf{I} - \mathbf{Q})\mathbf{X}_k\|^2]$, we used the trick $\mathbf{G}_\tau - \mathbf{H}_\tau + \mathbf{H}_\tau$ as there is no assumption for bounded (stochastic) gradients. However, For Adam type of algorithms, to the best of our knowledge, this assumption is still required to achieve the convergence. Regarding its relaxation we will leave in our future work. Hence, based on Lemma 2, we have the following relationship

$$\mathbb{E}[\|(\mathbf{I} - \mathbf{Q})\mathbf{X}_k\|^2] \leq \alpha^2 \mathbb{E}[\| \sum_{\tau=1}^{k-1} (\rho')^{k-1-\tau} \mathbf{M}_\tau \oslash \mathbf{U}_\tau^{1/2} \|^2] \tag{49}$$

Thus, based on the conditions in Theorem 4, we can easily get the inequality as follows

$$\mathbb{E}[\|(\mathbf{I} - \mathbf{Q})\mathbf{X}_k\|^2] \leq \frac{\alpha^2 N d G_\infty^2}{(1-\rho')^2 \epsilon}, \tag{50}$$

which holds due to $\|\mathbf{g}_k^i\| \leq G_\infty, [\mathbf{U}_k^i]_j \geq \epsilon$. Similarly, the upper bound of $\mathbb{E}[\|\bar{\mathbf{p}}_k - \bar{\mathbf{x}}_k\|^2]$ is as follows

$$\begin{aligned}
\mathbb{E}[\|\bar{\mathbf{p}}_k - \bar{\mathbf{x}}_k\|^2] &= \mathbb{E}[\| \frac{\beta_1}{1-\beta_1} (\bar{\mathbf{x}}_k - \bar{\mathbf{x}}_{k-1}) \|^2] \\
&= (\frac{\beta_1}{1-\beta_1})^2 \alpha^2 \mathbb{E}[\| \frac{1}{N} \sum_{i=1}^{N} \mathbf{m}_{k-1}^i \oslash (\mathbf{u}_k^i)^{1/2} \|^2] \\
&\leq (\frac{\beta_1}{1-\beta_1})^2 \frac{\alpha^2 d G_\infty^2}{N\epsilon}
\end{aligned} \tag{51}$$

Therefore, we can observe how the permutation matrix can be squeezed in the analysis such that the error bound is impacted with respect to the spectral gap $1-\rho'$. It also implies that existing analysis can be adapted to give the improved error bound shown in Theorem 4. Thus, we are not going to repeat all proof steps that are similar to existing analysis in [9], while, instead, giving the proof sketch, which assists in arriving at Theorem 4.

Proof. We now present the proof sketch for Theorem 4 and will refer interested readers to related works for more detail.

- *Step 1: Bounding gradient.* With the assistance of auxiliary sequence $\{\bar{\mathbf{p}}_k\}$, we don't have to consider the complicated update dependence on $\mathbf{m}_k$ and thus perform convergence analysis for the upper bound on $\nabla f(\bar{\mathbf{p}}_k)$. With this in hand, based on the smoothness of f, we subsequently construct the bound for $\frac{1}{K}\sum_{k=1}^{K} \mathbb{E}[\| \frac{\nabla f(\bar{\mathbf{x}}_k)}{(\bar{\mathbf{u}}_k)^{1/4}} \|^2]$, where $\bar{\mathbf{u}}_k = \frac{1}{N}\sum_{i=1}^{N} \mathbf{u}_k^i$.

$$\begin{aligned}
\frac{1}{K}\sum_{k=1}^{K} \mathbb{E}[\| \frac{\nabla f(\bar{\mathbf{x}}_k)}{(\bar{\mathbf{u}}_k)^{1/4}} \|^2] &\leq \frac{2\mathbb{E}[f(\bar{\mathbf{p}}_1) - f(\bar{\mathbf{p}}_{K+1})]}{K\alpha} \\
&+ \frac{2\beta_1 D_1}{K(1-\beta_1)} + \frac{2D_2}{K} + \frac{3D_3}{K} + \frac{L\mathbb{E}[\|\bar{\mathbf{p}}_{k+1} - \bar{\mathbf{p}}_k\|^2]}{K\alpha},
\end{aligned} \tag{52}$$

where

$$D_1 = \sum_{k=1}^{K} \mathbb{E}[\langle \nabla f(\bar{\mathbf{p}}_k), \frac{1}{N}\sum_{i=1}^{N} \mathbf{m}_{k-1}^i \oslash ((\mathbf{u}_{k-1}^i)^{1/2} - (\mathbf{u}_k^i)^{1/2})\rangle],$$

$$D_2 = \sum_{k=1}^{K} \mathbb{E}[\langle \nabla f(\bar{\mathbf{p}}_k), \frac{1}{N}\sum_{i=1}^{N} \nabla f^i(\mathbf{x}_k^i) \oslash ((\bar{\mathbf{u}}_k)^{1/2} - (\mathbf{u}_k^i)^{1/2})\rangle],$$

$$D_3 = \sum_{k=1}^{K} \mathbb{E}[\| \frac{\frac{1}{N}\sum_{i=1}^{N} \nabla f^i(\mathbf{x}_k^i) - \nabla f(\bar{\mathbf{x}}_k)}{(\bar{\mathbf{u}}_k)^{1/4}} \|^2 + \| \frac{\nabla f(\bar{\mathbf{p}}_k) - \nabla f(\bar{\mathbf{x}}_k)}{(\bar{\mathbf{u}}_k)^{1/4}} \|^2].$$

The smoothness condition and Eqs. 50 and 51 grant us the upper bound of D_3. Establishing the upper bounds for D_1 and D_2 give rise to the terms related to $\mathbb{E}[\sum_{k=1}^{K} \|\mathbf{V}_{k-1} - \mathbf{V}_{k-2}\|_{abs}]$, where $\|\mathbf{C}\|_{abs} = \sum_{i,j} |\mathbf{C}_{i,j}|$ denotes the entry-wise L_1 norm of a matrix. $\mathbf{V}_k$ is established as a non-decreasing function such that $\mathbb{E}[\sum_{k=1}^{K} \|\mathbf{V}_{k-1} - \mathbf{V}_{k-2}\|_{abs}] = \mathbb{E}[\sum_{i=1}^{N}\sum_{j=1}^{d}([\mathbf{v}_{K-1}^i]_j - [\mathbf{v}_0^i]_j)]$. Due to $\|\mathbf{g}_k^i\|_\infty \leq G_\infty$, it is proved that $[\mathbf{v}_k^i]_j \leq G_\infty^2$. With this, we can conclude that $\mathbb{E}[\sum_{k=1}^{K} \|\mathbf{V}_{k-1} - \mathbf{V}_{k-2}\|_{abs}] \leq NdG_\infty^2$.

- *Step 2: Bounding the drift term variance.* One important term in the proof is the stochastic gradient variance multiplied by the adaptive learning rate, $\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N} \mathbf{g}_k^i \oslash \mathbf{u}_k^i\|^2] \leq \mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N} \nabla f^i(\mathbf{x}_k^i) \oslash (\mathbf{u}_k^i)^{1/2}\|^2] + \frac{d\sigma^2}{N\epsilon}$. To process the first term on the right side of the above inequality, we can use $\bar{\mathbf{u}}_k$ and $\|\mathbf{a}+\mathbf{b}\|^2 \leq 2\|\mathbf{a}\|^2 + 2\|\mathbf{b}\|^2$ to transform from $\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N} \nabla f^i(\mathbf{x}_k^i) \oslash (\mathbf{u}_k^i)^{1/2}\|^2]$ to $\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N} \nabla f^i(\mathbf{x}_k^i) \oslash (\bar{\mathbf{u}}_k)^{1/2}\|^2]$. We then can bound them as performed for D_2 and D_3. Hence, we will reach to the bound in the following for $\frac{1}{K}\sum_{k=1}^{K} \mathbb{E}[\|\frac{\nabla f(\bar{\mathbf{x}}_k)}{(\bar{\mathbf{u}}_k)^{1/4}}\|^2]$:

$$\frac{1}{K}\sum_{k=1}^{K} \mathbb{E}[\|\frac{\nabla f(\bar{\mathbf{x}}_k)}{(\bar{\mathbf{u}}_k)^{1/4}}\|^2] \leq C_1(\frac{\mathbb{E}[f(\bar{\mathbf{p}}_1) - f^*]}{K\alpha} + \frac{\alpha d\sigma^2}{N}) + C_2\alpha^2 d + C_3\alpha^3 d + \frac{C_4 + C_5\alpha}{K\sqrt{N}} NdG_\infty^2, \tag{53}$$

where $C_1 = \max\{4, 4L/\epsilon\}$, $C_2 = 6(\frac{\beta_1^2}{(1-\beta_1)^2} + \frac{1}{(1-\rho')^2})\frac{LG_\infty^2}{\epsilon^{1.5}}$, $C_3 = \frac{16L^2G_\infty^2}{\epsilon^2}$, $C_4 = \frac{2(\lambda + \frac{\beta_1}{1-\beta_1})G_\infty^2}{\epsilon^{1.5}(1-\lambda)}$, $C_5 = \frac{2LG_\infty^2(\lambda + \frac{\beta_1}{1-\beta_1} + 2)}{\epsilon^2(1-\lambda)}$.

- *Setting explicit step size.* Setting the step size as $\alpha = \mathcal{O}(\sqrt{\frac{1}{dK}})$, substituting it into Eq. 53, and using the fact that $\|\bar{\mathbf{u}}_k\|_\infty \leq G_\infty^2$ yields the desired result.

With the above three steps, the desired conclusion is obtained. □

2. Additional Experimental Results

Within this section, we present supplementary experimental results encompassing additional datasets, specifically Tiny ImageNet and CIFAR10, along with an alternative architecture, ResNet50.

Moreover, our investigation delves into scalability, considering 10 and 20 models across various baseline algorithms. Additionally, we scrutinize the impact of different epoch configurations for DIMAT, providing insights into its performance over successive iterations.

Furthermore, we sought to explore the key factors contributing to the challenge of scalability in non-iid scenarios. By employing random initialization techniques and incorporating larger datasets, we aimed to assess the efficacy of our model under diverse conditions.

2.1. Additional Dataset Comparisons

In this subsection, we present an analysis of the Tiny ImageNet and CIFAR10 datasets. Table 4 provides a comprehensive overview of algorithmic performance across these additional datasets. The evaluation considers two key scenarios for each dataset: Fully Connected (FC) and Ring topologies in non-IID data. The reported values represent the mean and standard deviation of the performance metric obtained through multiple trial runs.

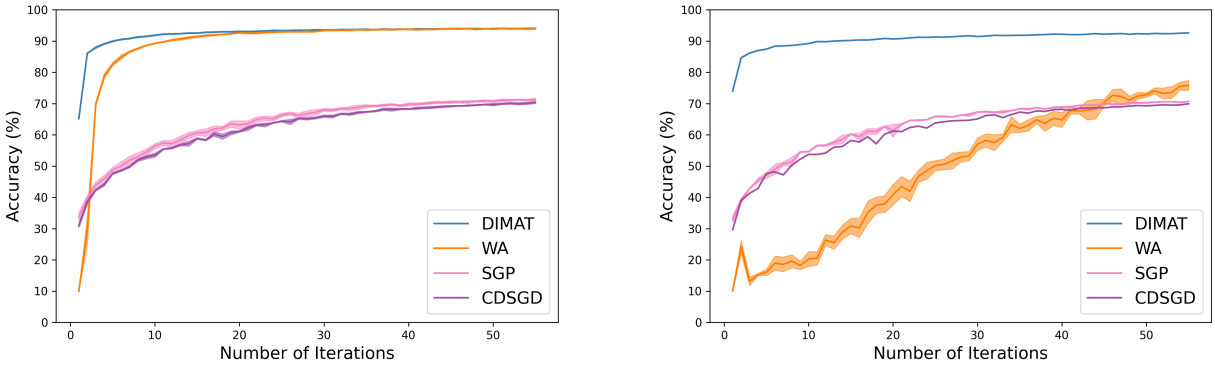
Table 4. Comparing algorithmic accuracy (mean±std) in fully connected and ring topologies with ResNet-20 architecture on Tiny ImageNet and CIFAR-10 non-IID data for 5 agents.

Algorithm	Tiny ImageNet		CIFAR10	
	FC	Ring	FC	Ring
SGP	9.49±0.43	7.42±0.24	19.18±0.11	19.04±0.27
CDSGD	9.05±0.15	7.36±0.25	18.85±0.08	19.20±0.16
WA	48.59±0.71	10.48±0.25	**49.25±3.95**	**23.14±1.46**
DIMAT (ours)	**49.09±0.23**	**17.70±0.14**	27.12±3.39	20.22±0.20

Notably, in the case of Tiny ImageNet, our proposed algorithm, DIMAT, emerges as a standout performer, outperforming all baseline algorithms across both non-IID and IID scenarios, as it can be seen in Table 4 and fig.7. However, for CIFAR10, the limited pretraining on only two classes results in a bias among agents, impacting their learning effectiveness. This bias is evident in their suboptimal performance. Nevertheless, in an IID setting, as depicted in fig. 6, DIMAT demonstrates superior performance on both fully connected and ring topologies.

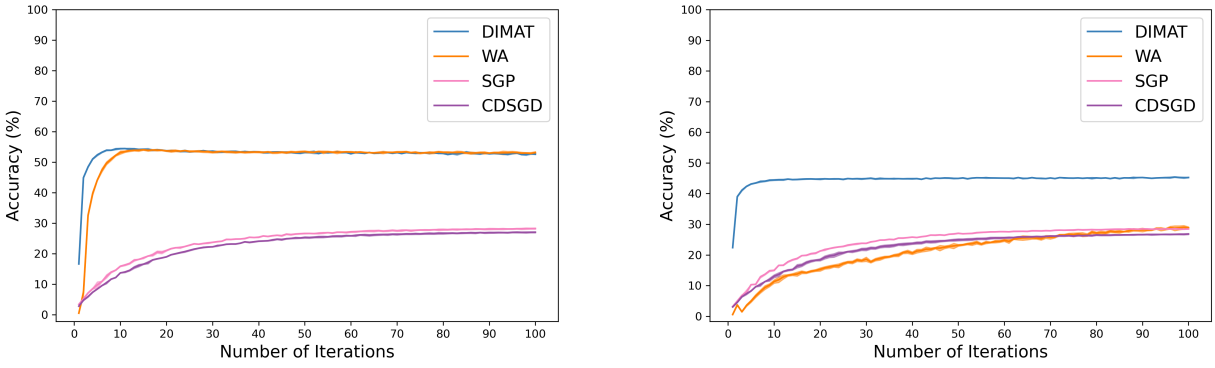
2.2. Additional Architecture Comparisons

The results for ResNet50 are presented in Table 5. Notably, the performances of WA and DIMAT are comparable. It



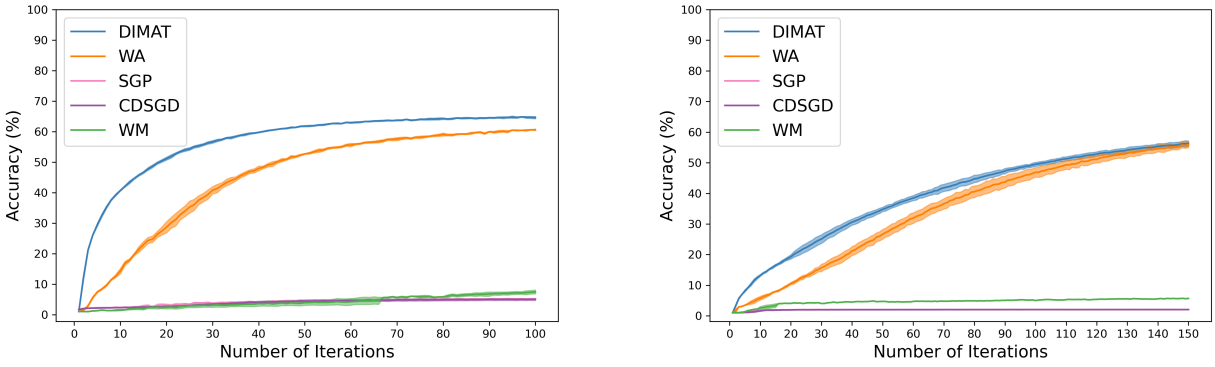
(a) Fully Connected (b) Ring

Figure 6. Comparing algorithmic accuracy (mean±std) in fully connected (a) and ring (b) topologies with ResNet-20 architecture on CIFAR-10 IID data for 5 agents.



(a) Fully Connected (b) Ring

Figure 7. Comparing algorithmic accuracy (mean±std) in fully connected (a) and ring (b) topologies with ResNet-20 architecture on Tiny ImageNet IID data for 5 agents.



(a) 10 Agents (b) 20 Agents

Figure 8. Comparing algorithmic accuracy (mean±std) for ten (a) and twenty (b) agents with VGG16 architecture on CIFAR-100 IID data.

is evident that DIMAT's performance is highly dependent on the chosen architecture. Additionally, in the IID ring

scenario, DIMAT emerges as the top performer, surpassing all other algorithms in terms of accuracy.

Table 5. Comparison of Test Accuracy (mean±std) on CIFAR-100 with ResNet-50 architecture for 5 agents under both IID and non-IID data distribution, considering fully connected (FC) and ring topologies.

Algorithm	**IID**		**non-IID**	
	FC	Ring	FC	Ring
SGP	39.99±0.45	39.74±0.04	13.27±0.09	13.33±0.12
CDSGD	37.99±0.16	37.94±0.38	12.90±0.10	9.10±5.73
WA	**49.07±0.16**	32.47±0.24	**46.76±0.57**	**24.21±0.39**
DIMAT (ours)	42.59±1.00	**42.06±0.13**	45.10±0.49	19.54±0.05

2.3. Additional Scalability Analysis

2.3.1 IID Data Scalability

Figures 8a and 8b depict algorithmic accuracy trends with varying agent numbers using CIFAR-100 IID data and the VGG16 architecture. Figure 8a illustrates accuracy trends for 10 agents, providing a snapshot of algorithmic behavior in a moderately scaled scenario. In fig. 8b, the analysis extends to 20 agents, offering valuable insights into the algorithm's robustness and scalability as agent numbers increase in the IID scenario. It's noteworthy that, even with this escalation in the number of agents, DIMAT consistently outperforms all baseline algorithms, showcasing its resilience and superior performance in the face of increased scalability.

2.3.2 Non-IID Data Scalability

In investigating scalability under non-IID scenarios, we hypothesized that the bias introduced by pretraining models might be a contributing factor. Given that the number of classes in non-IID settings decreases, pretraining could potentially lead to biased initializations. To test this hypothesis, we experimented with random initialization instead of starting with pretrained models. However, our results indicate that this change in initialization strategy does not significantly affect the performance of DIMAT. The observed trends remain consistent with those obtained when using pretrained models, suggesting that factors beyond initialization bias do not influence DIMAT's performance in non-IID scenarios. This can be seen in fig. 9. Another approach we explored to address this issue was utilizing larger datasets, such as Tiny ImageNet, for more agents in non-IID scenarios. As depicted in fig. 10, even with 10 agents, the accuracy continues to increase over time. This suggests that the underlying scalability issue in non-IID scenarios might be influenced by the dataset size. A larger dataset appears to improve performance, particularly for a higher number of agents. However, there seems to be a breakpoint. Unlike CIFAR-100, Tiny ImageNet is capable of handling a larger number of agents but still exhibits a breakpoint as we increase the number of agents, indicating that learning becomes progressively more challenging.

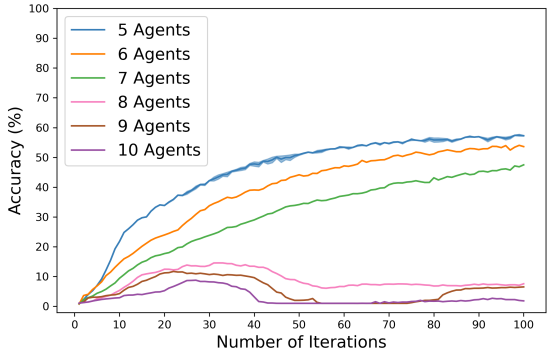


Figure 9. Impact of model's initialization on accuracy (Mean±Std) using ResNet-20 architecture and a fully connected topology on CIFAR-100 non-IID data. Results show the performance of the DIMAT algorithm with 5 to 10 agents.

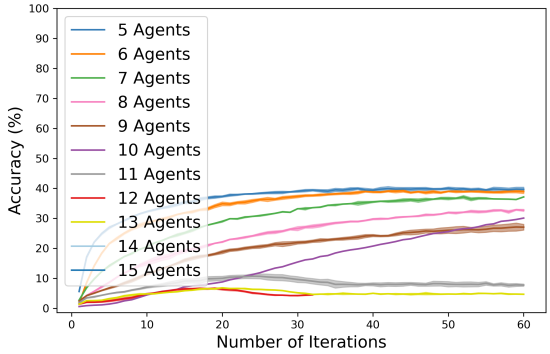


Figure 10. Effect of dataset size on accuracy (Mean±Std) using ResNet-20 architecture with a fully connected topology on Tiny ImageNet non-IID data. The figure illustrates the performance of the DIMAT algorithm with 5 to 15 agents.

2.4. Exploring Varied Training Epochs

In this subsection, we present results from experiments conducted with different numbers of training epochs for each iteration. Our analysis reveals that the optimal number of training epochs between iterations is 2, outperforming configurations with 1, 5, 7, and 10 training epochs. These findings are illustrated in Fig. 11.

2.5. Visualization of Communication Overhead

Figure 12 illustrates a comparison of communication overhead among DIMAT, SGP, CGA, and CDSGD for 5, 10, and 20 agents across fully connected and ring topologies.

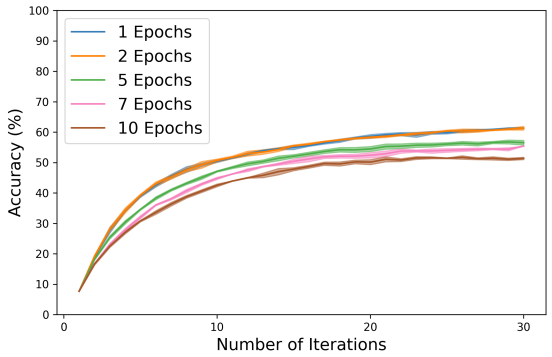


Figure 11. Impact of diverse training epochs on agents accuracy (Mean±Std) with fully Connected Topology using ResNet-20 architecture on CIFAR-100 non-IID data for 5 agents on the DIMAT algorithm. .

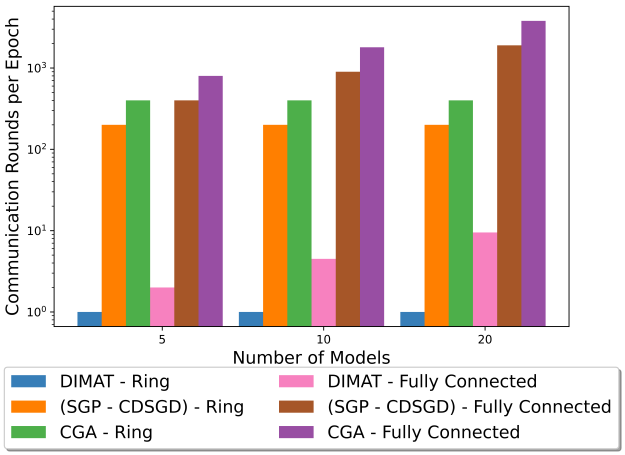


Figure 12. Number of communication rounds per epoch for fully connected and ring topologies.

DIMAT significantly requires fewer communication rounds compared to SGP, CGA, and CDSGD.

2.6. Computational Overhead

In this subsection, we examine the computational overhead of various algorithms when training 5 agents using ResNet20 architecture on the non-IID CIFAR-100 dataset for 100 iterations. We focus on GPU memory usage and computation time as key performance metrics. Table 6 compares GPU memory usage and computation time for SGP, CDSGD, and our proposed method, DIMAT.

Experiments were conducted on an NVIDIA A100 GPU (80 GB). It is important to note that the reported GPU memory usage is approximate. DIMAT demonstrates significantly lower GPU memory usage, requiring only 6 GB compared to 15 GB for both SGP and CDSGD. Furthermore, DIMAT achieves faster computation, completing the task in 15.95 hours compared to 16.88 hours for SGP and 16.99 hours for CDSGD.

These findings underscore the efficiency of DIMAT in terms of memory usage and computation time, rendering it a promising approach for decentralized learning tasks.

Table 6. Comparison of GPU memory usage and computation time for 5 agents using ResNet20 on non-IID CIFAR-100 data for 100 iterations. The experiments were conducted on an NVIDIA A100 GPU.

Algorithm	GPU	Time
SGP	15 GB	16.88 hrs.
CDSGD	15 GB	16.99 hrs.
DIMAT (ours)	**6 GB**	**15.95 hrs.**

3. Expanded Explanations of Selected Terminologies

3.1. Mixing Matrix

The mixing matrix, a doubly stochastic matrix, signifies inter-agent influences in collaborative learning systems. While various design choices exist, we adopt a vanilla version for illustration. In a fully connected topology, the matrix is uniform: for instance, in a 5-agent network, all elements are set to 0.2 for symmetrical collaboration. In a ring topology, where agents equally influence their two adjacent counterparts, the matrix takes a circular pattern. Specifically, elements corresponding to the three neighboring agents are 0.333, while the rest are 0. This matrix representation is as follows:

For fully connected topology:

$$\begin{bmatrix} 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \end{bmatrix}$$

For ring topology:

$$\begin{bmatrix} 0.333 & 0.333 & 0 & 0 & 0.333 \\ 0.333 & 0.333 & 0.333 & 0 & 0 \\ 0 & 0.333 & 0.333 & 0.333 & 0 \\ 0 & 0 & 0.333 & 0.333 & 0.333 \\ 0.333 & 0 & 0 & 0.333 & 0.333 \end{bmatrix}$$

3.2. Activation Matching

We adopt the method proposed by Ainsworth et al. [1]. This method aims to associate units across two models by performing regression between their activations, under the premise that models must learn similar features to effectively perform the same task.

Given the activations of each model, the objective is to link corresponding units between model 1 (M_1) and model 2 (M_2), assuming a potential linear relationship between their activations. For activations of the ℓth layer, represented by $\mathbf{Z}^{(M_1)}$ and $\mathbf{Z}^{(M_2)}$, the goal is to minimize the discrepancy between their activations using a linear assignment problem (LAP), for which efficient algorithms exist.

After solving the assignment problem for each layer, the weights of model 2 are adjusted to closely match those of model 1. This adjustment involves permuting both weights and biases for each layer, resulting in weights that generate activations closely aligned with those of model 1.

This method is computationally efficient, requiring only a single pass over the training dataset to compute activation matrices. Furthermore, activation matching at each layer operates independently of other layers, simplifying the optimization process.