Appendix

In the appendix, we first provide detailed proofs of all the remaining results in Section A. Then, we present an application of APDRCD and APDGCD algorithms for the Wasserstein barycenter problem in Section B. Finally, further comparative experiments between APDGCD algorithm versus APDRCD, APDAGD and APDAMD algorithms, and experiments on larger synthetic image datasets and CIFAR10 dataset are in Section C.

A Proofs for all results

In this appendix, we provide the complete proofs for remaining results in the main text.

A.1 Proof of Lemma 3.2

By Eq. (5), we have the following equations

$$\begin{aligned}&|\nabla\varphi_{i_k}(y^k)|^2\\&=2|\nabla\varphi_{i_k}(y^k)|^2-|\nabla\varphi_{i_k}(y^k)|^2\\&=2L\big(y^k_{i_k}-\lambda^{k+1}_{i_k}\big)\nabla_{i_k}\varphi(y^k)-L^2(\lambda^{k+1}_{i_k}-y^{k+1}_{i_k})^2.\end{aligned}$$

Combining the above equations with Lemma 3.1, we have

$$\begin{aligned}&\varphi(\lambda^{k+1})\\&\le\varphi(y^k)-\frac{1}{2L}(\nabla_{i_k}\varphi(y^k))^2\\&=\varphi(y^k)+(\lambda^{k+1}_{i_k}-y^k_{i_k})\nabla_{i_k}\varphi(y^k)+\frac{L}{2}(\lambda^{k+1}_{i_k}-y^{k+1}_{i_k})^2.\end{aligned}\tag{9}$$

Furthermore, the results from Eq. (5) and Eq. (6) lead to the following equations

$$\begin{aligned}\lambda^{k+1}_{i_k}-y^k_{i_k}&=-\frac{1}{L}\nabla_{i_k}\varphi(y^k),\\z^{k+1}_{i_k}-z^k_{i_k}&=-\frac{1}{2nL\theta_k}\nabla_{i_k}\varphi(y^k).\end{aligned}$$

Therefore, we have

$$\lambda^{k+1}_{i_k}-y^k_{i_k}=2n\theta_k(z^{k+1}_{i_k}-z^k_{i_k}).$$

The above equation together with Eq. (9) yields the following inequality

$$\begin{aligned}&\varphi(\lambda^{k+1})\\&\le\varphi(y^k)+2n\theta_k\big(z^{k+1}_{i_k}-z^k_{i_k}\big)\nabla_{i_k}\varphi_{i_k}(y^k)\\&+2n^2L\theta_k^2\big(z^{k+1}_{i_k}-z^k_{i_k}\big)^2.\end{aligned}\tag{10}$$

By the result of Eq. (6), we have

$$(z^{k+1}_{i_k}-z^k_{i_k})+\frac{1}{2nL\theta_k}\nabla_{i_k}\varphi(y^k)=0.$$

Therefore, for any $\lambda\in\mathbb{R}^{2n}$, we find that

$$(\lambda_{i_k}-z^{k+1}_{i_k})[(z^{k+1}_{i_k}-z^k_{i_k})+\frac{1}{2nL\theta_k}\nabla_{i_k}\varphi(y^k)]=0.$$

Note that the above equation is equivalent to the following:

$$\begin{aligned}&\frac{1}{nL\theta_k}(\lambda_{i_k}-z^{k+1}_{i_k})\nabla_{i_k}\varphi(y^k)\\&=-2(\lambda_{i_k}-z^{k+1}_{i_k})(z^{k+1}_{i_k}-z^k_{i_k})\\&=(\lambda_{i_k}-z^{k+1}_{i_k})^2-(\lambda_{i_k}-z^k_{i_k})^2+(z^{k+1}_{i_k}-z^k_{i_k})^2\end{aligned}$$

where the second equality in the above display comes from simple algebra. Rewriting the above equality, we have:

$$\begin{aligned}&(z^{k+1}_{i_k}-z^k_{i_k})^2\\&=\frac{1}{nL\theta_k}(\lambda_{i_k}-z^{k+1}_{i_k})\nabla_{i_k}\varphi(y^k)\\&-(\lambda_{i_k}-z^{k+1}_{i_k})^2+(\lambda_{i_k}-z^k_{i_k})^2\end{aligned}$$

Combining the above equation with Eq. (10) yields the following inequality:

$$\begin{aligned}&\varphi(\lambda^{k+1})\\&\le\varphi(y^k)+2n\theta_k(\lambda_{i_k}-z^k_{i_k})\nabla_{i_k}\varphi(y^k)\\&+2n^2L\theta_k^2\Big[(\lambda_{i_k}-z^k_{i_k})^2-(\lambda_{i_k}-z^{k+1}_{i_k})^2\Big].\end{aligned}\tag{11}$$

Recall the definition of y^k in Step 3 of Algorithm 1 as follows:

$$y^k=(1-\theta_k)\lambda^k+\theta^k z^k,$$

which can be rewritten as:

$$\theta_k(\lambda-z^k)=\theta_k(\lambda-y^k)+(1-\theta_k)(\lambda^k-y^k)\tag{12}$$

for any $\lambda\in\mathbb{R}^{2n}$.

The above equation implies that

$$\begin{aligned}&\varphi(y^k)+\theta_k(\lambda-z^k)^T\nabla\varphi(y^k)\\&\le\theta_k[\varphi(y^k)+(\lambda-y^k)^T\nabla\varphi(y^k)]\\&+(1-\theta_k)[\varphi(y^k)+(\lambda^k-y^k)^T\nabla\varphi(y^k)]\\&\le\theta_k[\varphi(y^k)+(\lambda-y^k)^T\nabla\varphi(y^k)]+(1-\theta_k)\varphi(\lambda^k)\end{aligned}$$

where the last inequality comes from the convexity of φ. Combining this equation and taking expectation over i_k for the first two terms of Eq. (11), we have:

$$\begin{aligned}&\varphi(y^k)+(2n\theta_k)\mathbb{E}_{i_k}[(\lambda_{i_k}-z^k_{i_k})\nabla_{i_k}\varphi(y^k)]\\&=\varphi(y^k)+\theta_k(\lambda-z^k)^T\nabla\varphi(y^k)\\&\le\theta_k[\varphi(y^k)+(\lambda-y^k)^T\nabla\varphi(y^k)]+(1-\theta_k)\varphi(\lambda^k)\end{aligned}\tag{13}$$

where we use Eq. (12) and the convexity of the dual function in the last step. For the last term in the right hand side of Eq. (11), by taking expectation over i_k, we have:

$$\mathbb{E}_{i_k}\Big[2n^2L\theta_k^2[(\lambda_{i_k}-z_{i_k}^k)^2-(\lambda_{i_k}-z_{i_k}^{k+1})^2]\Big] = 2n^2L\theta_k^2\mathbb{E}_{i_k}\big[||\lambda-z^k||^2-||\lambda-z^{k+1}||^2\big] \quad (14)$$

where the last equality comes from the following equations:

$$\begin{aligned}
&\mathbb{E}_{i_k}\big[(\lambda_{i_k}-z_{i_k}^k)^2-\big(\lambda_{i_k}-z_{i_k}^{k+1}\big)^2\big]\\
&=\mathbb{E}_{i_k}\left[(\lambda_{i_k}-z_{i_k}^k)^2-\left(\lambda_{i_k}-z_{i_k}^k+\frac{1}{2nL\theta_k}\nabla_{i_k}\varphi(y^k)\right)^2\right]\\
&=\frac{1}{2n}\Big(\|\lambda-z^k\|^2-\sum_{i_k=0}^{2n}\left(\lambda_{i_k}-z_{i_k}^k+\frac{1}{2nL\theta_k}\nabla_{i_k}\varphi(y^k)\right)^2\Big)\\
&=\frac{1}{2n}\|\lambda-z^k\|^2-\frac{1}{2n}\|\lambda-z^k+\frac{1}{2nL\theta_k}\nabla\varphi(y^k)\|^2\\
&=\frac{1}{2n}\left[-\frac{(\lambda-z^k)}{nL\theta_k}\nabla\varphi(y^k)-\frac{1}{4n^2L^2\theta_k^2}\|\nabla\varphi(y^k)\|^2\right]\\
&=(-2)(\lambda-z^k)\mathbb{E}_{i_k}[z^k-z^{k+1}]-\mathbb{E}_{i_k}[||z^k-z^{k+1}||^2]
\end{aligned}$$

where the last inequality is due to the fact that $\nabla\varphi(y^k)=4\mathbb{E}_{i_k}[(z^k-z^{k+1})n^2L\theta_k]$ and Jensen's inequality. Therefore, by simple algebra, we have

$$\begin{aligned}
&\mathbb{E}_{i_k}\Big[(\lambda_{i_k}-z_{i_k}^k)^2-(\lambda_{i_k}-z_{i_k}^{k+1})^2\Big]\\
&=(-2)(\lambda-z^k)\mathbb{E}_{i_k}[z^k-z^{k+1}]-\mathbb{E}_{i_k}[||z^k-z^{k+1}||^2]\\
&=\mathbb{E}_{i_k}[\|\lambda-z^k\|^2-\|\lambda-z^{k+1}\|^2].
\end{aligned}$$

Notice that equation (11) holds for any value of i_k. Hence, by combining the results from Eq. (13) and Eq. (14) with Eq. (11), at each iteration with a certain value of i_k, we obtain that

$$\begin{aligned}
&\mathbb{E}_{i_k}[\varphi(\lambda^{k+1})]\\
&\le(1-\theta_k)\varphi(\lambda^k)+\theta_k[\varphi(y^k)+(\lambda-y^k)^\top\nabla\varphi(y^k)]\\
&+2n^2L\theta_k^2\Big(||\lambda-z^k||^2-\mathbb{E}_{i_k}[||\lambda-z^{k+1}||^2]\Big).
\end{aligned}$$

As a consequence, we achieve the conclusion of the lemma.

A.2 Proof of Theorem 3.3

By the result of Lemma 3.2 and the definition of the sequence $\{\theta_k\}$ in Algorithm 1, we obtain the following bounds:

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{\theta_k^2}\varphi(\lambda^{k+1})\right]\\
&\le\mathbb{E}\Bigg[\frac{1-\theta_k}{=}\theta_k^2\varphi(\lambda^k)+\frac{1}{\theta_k}[\varphi(y^k)+(\lambda-y^k)^T\nabla\varphi(y^k)]\\
&+2Ln^2\big(||\lambda-z^k||^2-||\lambda-z^{k+1}||^2\big)\Bigg]\\
&=\mathbb{E}\left[\frac{1}{\theta_{k-1}^2}\varphi(\lambda^k)+\frac{1}{\theta_k}[\varphi(y^k)+(\lambda-y^k)^T\nabla\varphi(y^k)]\right.\\
&\left.+2Ln^2\big(||\lambda-z^k||^2-||\lambda-z^{k+1}||^2\big)\right]
\end{aligned}$$

where the outer expectations are taken with respect to the random sequence of the coordinate indices in Algorithm 1. Keep iterating the above bound and using the fact that $\theta_0=1$ and $C_k=1/\theta_k^2$, we arrive at the following inequalities:

$$\begin{aligned}
&C_k\mathbb{E}\big[\varphi(\lambda^{k+1})\big]\\
&\le\sum_{i=0}^k\frac{1}{\theta_i}\mathbb{E}\big[\varphi(y^i)+\big\langle\nabla\varphi(y^i),\lambda-y^i\big\rangle\big]\\
&+2Ln^2\big(||\lambda-z^0||^2-\mathbb{E}\big[||\lambda-z^{k+1}||^2\big]\big)\\
&\le\min_{\lambda\in\mathbb{R}^{2n}}\Bigg(\sum_{i=0}^k\frac{1}{\theta_i}\mathbb{E}\big[\varphi(y^i)+\big\langle\nabla\varphi(y^i),\lambda-y^i\big\rangle\big]\\
&+2Ln^2\big(||\lambda-z^0||^2-\mathbb{E}\big[||\lambda-z^{k+1}||^2\big]\big)\Bigg)\\
&\le\min_{\lambda\in B_2(2\hat{R})}\Bigg(\sum_{i=0}^k\frac{1}{\theta_i}\mathbb{E}\big[\varphi(y^i)+\big\langle\nabla\varphi(y^i),\lambda-y^i\big\rangle\big]\\
&+2Ln^2\big(||\lambda-z^0||^2-\mathbb{E}\big[||\lambda-z^{k+1}||^2\big]\big)\Bigg)
\end{aligned}$$

where $\hat{R}:=\eta\sqrt{n}(R+\frac{1}{2})$ is the upper bound for l_2-norm of optimal solutions of dual regularized OT problem (4) according to Lemma 3.2 in (Lin et al., 2019) and $\mathbb{B}_2(r)$ is defined as

$$\mathbb{B}_2(r):=\{\lambda\in\mathbb{R}^{2n}\mid||\lambda||_2\le r\}.$$

As $\mathbb{E}\big[||\lambda-z^{k+1}||^2\big]\ge0$, the inequality in the above display can be further rewritten as

$$\begin{aligned}
&C_k\mathbb{E}\big[\varphi(\lambda^{k+1})\big]\\
&\le\min_{\lambda\in\mathbb{B}_2(2\hat{R})}\Bigg(\sum_{i=0}^k\frac{1}{\theta_i}\mathbb{E}\big[\varphi(y^i)+\big\langle\nabla\varphi(y^i),\lambda-y^i\big\rangle\big]\\
&+2Ln^2||\lambda-z^0||^2\Bigg)\\
&\le\min_{\lambda\in\mathbb{B}_2(2\hat{R})}\Bigg(\sum_{i=0}^k\frac{1}{\theta_i}\mathbb{E}\big[\varphi(y^i)+\big\langle\nabla\varphi(y^i),\lambda-y^i\big\rangle\big]\\
&+8Ln^2\hat{R}^2\Bigg)
\end{aligned}\quad (15)$$

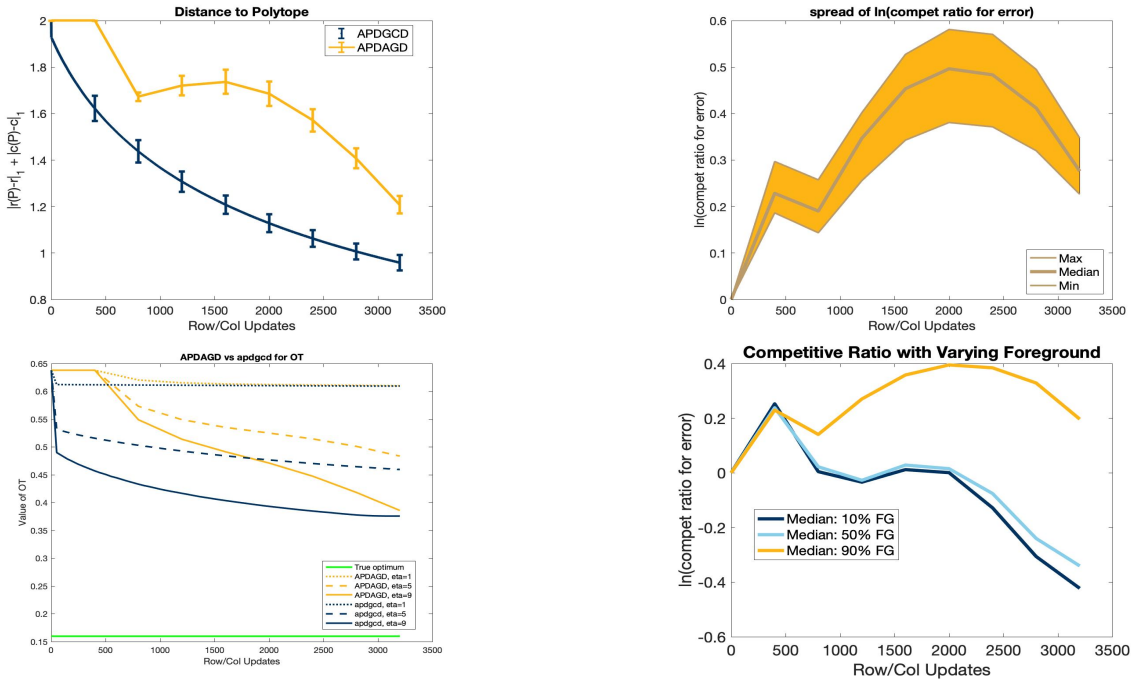

Figure 4: Performance of APDGCD and APDAGD algorithms on the synthetic images. The organization of the images is similar to those in Figure 1.

where the last inequality is due to $z^0 = 0$. Furthermore, by the definition of the dual entropic regularized OT objective function $\varphi(\lambda)$, we can verify the following equations:

$$\begin{aligned}
&\varphi(y^i) + \left\langle \nabla\varphi(y^i), \lambda - y^i \right\rangle \\
&= \left\langle y^i, b - Ax(y^i) \right\rangle - f(x(y^i)) + \left\langle \lambda - y^i, b - Ax(y^i) \right\rangle \\
&= -f(x(y^i)) + \left\langle \lambda, b - Ax(y^i) \right\rangle
\end{aligned}$$

where $f(x) := \langle C, x\rangle$, $x(\lambda) := \underset{x\in\mathbb{R}^{n\times n}}{\arg\max}\Big\{ -f(x) - \left\langle A^\top\lambda, x\right\rangle\Big\}$, and $b = \begin{pmatrix} r \\ l \end{pmatrix}$. The above equation leads to the following inequality:

$$\begin{aligned}
&\sum_{i=0}^{k} \frac{1}{\theta_i}\mathbb{E}\big[\varphi(y^i) + \left\langle \nabla\varphi(y^i), \lambda - y^i \right\rangle\big] \\
&= \sum_{i=0}^{k} \frac{1}{\theta_i}\mathbb{E}\big[-f(x(y^i)) + \left\langle \lambda, b - Ax(y^i) \right\rangle\big] \\
&\le -C_k f(\mathbb{E}\big[x^k\big]) + \sum_{i=0}^{k} \frac{1}{\theta_i}\left\langle \lambda, b - A\mathbb{E}\big[x(y^i)\big] \right\rangle \\
&= C_k\big(-f(\mathbb{E}\big[x^k\big]) + \left\langle \lambda, b - A\mathbb{E}\big[x^k\big] \right\rangle\big) \qquad (16)
\end{aligned}$$

where the second inequality is due to the convexity of f. Combining the results from (15) and (16), we achieve the following bound

$$\begin{aligned}
&C_k\mathbb{E}\big[\varphi(\lambda^{k+1})\big] \\
&\le -C_k f(\mathbb{E}\big[x^k\big]) + \min_{\lambda\in\mathbb{B}_2(2\hat{R})}\{C_k\left\langle \lambda, b - A\mathbb{E}\big[x^k\big]\right\rangle\} \\
&+ 8Ln^2\hat{R}^2 \\
&\le -C_k f(\mathbb{E}\big[x^k\big]) + 8Ln^2\hat{R}^2 - 2C_k\hat{R}\mathbb{E}\big[||Ax^k - b||_2\big].
\end{aligned}$$

which is equivalent to

$$\begin{aligned}
&f(\mathbb{E}\big[x^k\big]) + \mathbb{E}\big[\varphi(\lambda^{k+1})\big] + 2\hat{R}\mathbb{E}\big[||Ax^k - b||_2\big] \\
&\le \frac{8Ln^2\hat{R}^2}{C_k}. \qquad (17)
\end{aligned}$$

Denoting λ^* as the optimal solution for the dual entropic regularized OT problem (4). Then, we can verify the following inequalities

$$\begin{aligned}
&f(\mathbb{E}\big[x^k\big]) + \mathbb{E}\big[\varphi(\lambda^{k+1})\big] \\
&\ge f(\mathbb{E}\big[x^k\big]) + \varphi(\lambda^*) \\
&= f(\mathbb{E}\big[x^k\big]) + \langle\lambda^*, b\rangle + \max_{x\in\mathbb{R}^{n\times n}}\{-f(x) - \left\langle A^\top\lambda^*, x\right\rangle\} \\
&\ge f(\mathbb{E}\big[x^k\big]) + \langle\lambda^*, b\rangle - f(\mathbb{E}\big[x^k\big]) - \left\langle \lambda^*, A\mathbb{E}\big[x^k\big]\right\rangle \\
&= \left\langle \lambda^*, b - A\mathbb{E}\big[x^k\big]\right\rangle \\
&\ge -\hat{R}\mathbb{E}\big[||Ax^k - b||_2\big] \qquad (18)
\end{aligned}$$

where the last inequality comes from Hölder inequality and the fact that $||\lambda^*||_2 \le \hat{R}$. Plugging the inequality

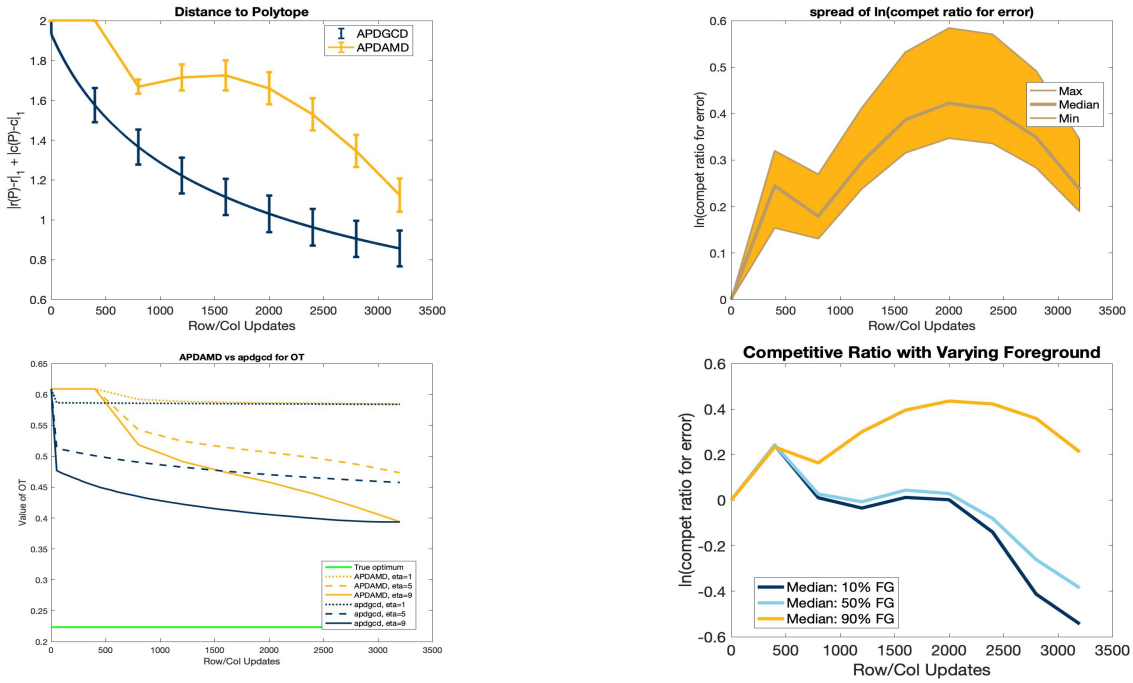

Figure 5: Performance of APDGCD and APDAMD algorithms on the synthetic images. The organization of the images is similar to those in Figure 5.

in (18) to the inequality in (17) leads to the following bound:

$$\begin{aligned}&\mathbb{E}\big[||Ax^k - b||_2\big] \qquad (19)\\ &\leq \frac{8Ln^2\hat{R}}{C_k}\\ &= \frac{8||A||_1^2 n^{\frac{5}{2}}(R+1/2)}{C_k}\\ &= \frac{32n^{\frac{5}{2}}(R+1/2)}{C_k}. \qquad (20)\end{aligned}$$

Therefore, $\mathbb{E}\big[||Ax^k - b||_1\big] \leq \frac{32n^3(R+1/2)}{C_k}$. It remains to bound C_k. We will use induction to show that $\theta_k \leq \frac{2}{k+2}$ for all $k \geq 0$. The inequality clearly holds for $k = 0$ as $\theta_0 = 1$. Suppose that the hypothesis holds for $k \geq 0$, namely, $\theta_k \leq \frac{2}{k+2}$ for $k \geq 0$. By the definition of θ_{k+1} and simple algebra, we obtain that

$$\theta_{k+1} = \frac{\theta_k^2}{2}\left(\sqrt{1 + \frac{4}{\theta_k^2}} - 1\right) \leq \frac{2}{k+3}$$

where the above inequality is due to $\theta_k \leq \frac{2}{k+2}$. Therefore, we achieve the conclusion of the hypothesis for $k+1$. Now, simple algebra demonstrates that $C_k \geq \frac{1}{4}(k+1)(k+4) \geq \frac{1}{4}(k+1)^2$. Combining this lower bound of C_k and the inequality in (20) leads to the following result:

$$\mathbb{E}\big[||Ax^k - b||_1\big] \leq \frac{128n^3(R+1/2)}{(k+1)^2}.$$

As a consequence, we conclude the desired bound on the number of iterations k required to satisfy the bound $\mathbb{E}\big[||A\text{vec}(X^k) - b||_1\big] \leq \varepsilon'$.

A.3 Proof of Theorem 3.4

The proof of the theorem follows the same steps as those in the proof of Theorem 1 in (Altschuler et al., 2017). Here, we provide the detailed proof for the completeness. In particular, we denote $\tilde{X}$ the matrix returned by the APDRCD algorithm (Algorithm 1) with $\tilde{r}$, $\tilde{l}$ and $\varepsilon'/2$. Recall that, X^* is a solution to the OT problem. Then, we obtain the following inequalities:

$$\begin{aligned}&\mathbb{E}[\langle C, \hat{X}\rangle] - \langle C, X^*\rangle\\ &\leq 2\eta\log(n) + 4\mathbb{E}\Big[||\tilde{X}\mathbf{1} - r||_1 + ||\tilde{X}^\top\mathbf{1} - l||_1\Big]||C||_\infty\\ &\leq \frac{\varepsilon}{2} + 4\mathbb{E}\Big[||\tilde{X}\mathbf{1} - r||_1 + ||\tilde{X}^\top\mathbf{1} - l||_1\Big]||C||_\infty,\end{aligned}$$

where the last inequality in the above display holds since $\eta = \frac{\varepsilon}{4\log(n)}$. Furthermore, we have

$$\begin{aligned}&\mathbb{E}\Big[||\tilde{X}\mathbf{1} - r||_1 + ||\tilde{X}^\top\mathbf{1} - l||_1\Big]\\ &\leq \mathbb{E}\Big[\Big\|\tilde{X}\mathbf{1} - \tilde{r}\Big\|_1 + \Big\|\tilde{X}^\top\mathbf{1} - \tilde{l}\Big\|_1\Big] + \|r - \tilde{r}\|_1 + \Big\|l - \tilde{l}\Big\|_1\\ &\leq \frac{\varepsilon'}{2} + \frac{\varepsilon'}{2} \;=\; \varepsilon'.\end{aligned}$$

Since $\varepsilon' = \dfrac{\varepsilon}{8\,\|C\|_\infty}$, the above inequalities demonstrate $\mathbb{E}[\langle C, \hat{X}\rangle] - \langle C, X^*\rangle \leq \varepsilon$. Hence, we only need to bound

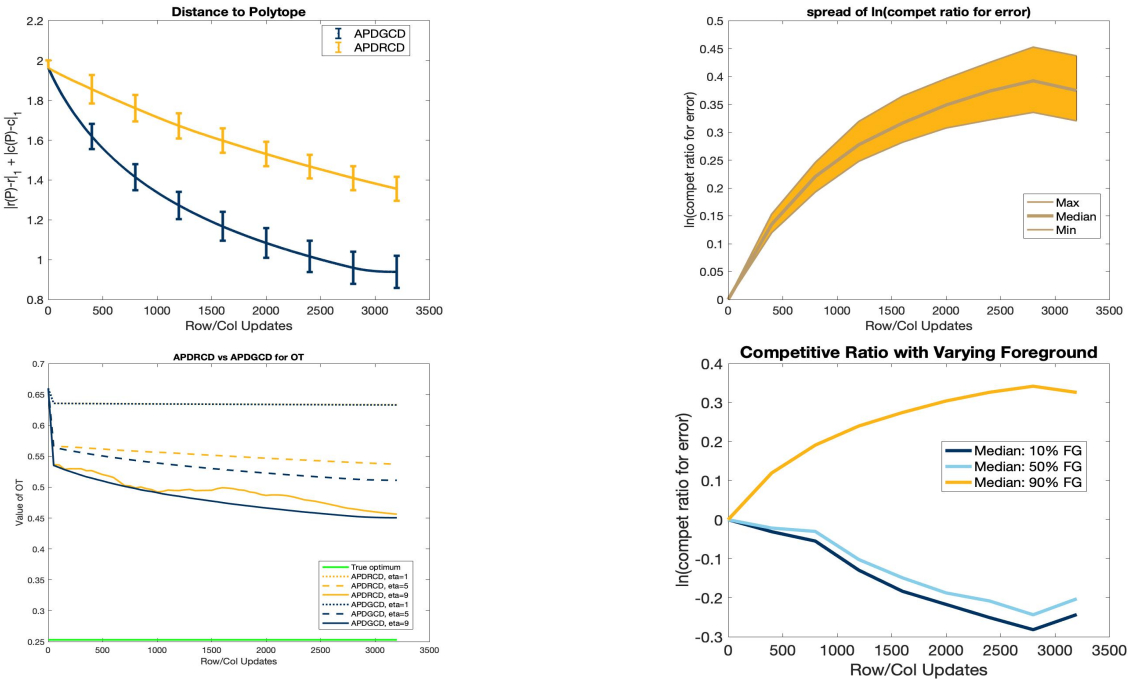

Figure 6: Performance of APDGCD and APDRCD algorithms on the synthetic images. The organization of the images is similar to those in Figure 1.

the complexity. Following the approximation scheme in Step 1 of Algorithm 2, we achieve the following bound

$$R = \frac{||C||_\infty}{\eta} + \log(n) - 2\log(\min_{1\le i,j\le n}\{\widetilde{r}_i, \widetilde{l}_i\})$$
$$\le \frac{4\,\|C\|_\infty \log(n)}{\varepsilon} + \log(n) - 2\log\left(\frac{\varepsilon}{64n\,\|C\|_\infty}\right).$$

Given the above bound with R, we have the following bound with the iteration count:

$$k \le 1 + 12n^{\frac{3}{2}}\sqrt{\frac{R+1/2}{\varepsilon'}}$$
$$\le 1 + 12n^{\frac{3}{2}}\left\{\frac{8||C||_\infty}{\varepsilon}\left(\frac{4||C||_\infty \log(n)}{\varepsilon} + \log(n)\right.\right.$$
$$\left.\left. - 2\log\left(\frac{\varepsilon}{64n||C||_\infty}\right) + \frac{1}{2}\right)\right\}^{\frac{1}{2}}$$
$$= \mathcal{O}\left(\frac{n^{\frac{3}{2}}||C||_\infty\sqrt{\log(n)}}{\varepsilon}\right).$$

Combining the above result with the fact that each iteration the APDRCD algorithm requires $\mathcal{O}(n)$ arithmetic operations, we conclude that the total number of arithmetic operations required for the APDRCD algorithm for approximating optimal transport is $\mathcal{O}\left(\frac{n^{\frac{5}{2}}||C||_\infty\sqrt{\log(n)}}{\varepsilon}\right)$. Furthermore, the column $\tilde{r}$ and row $\tilde{l}$ in Step 2 of Algorithm 2 can be found in $\mathcal{O}(n)$ arithmetic operations while Algorithm 2 in (Altschuler et al., 2017) requires $\mathcal{O}(n^2)$ arithmetic operations. As a consequence, we conclude that the total number of arithmetic operations is $\mathcal{O}\left(\frac{n^{\frac{5}{2}}||C||_\infty\sqrt{\log(n)}}{\varepsilon}\right)$.

Note that by Markov inequality,

$$P(\langle C, \hat{X}\rangle > a) \le \frac{\mathbb{E}[\langle C, \hat{X}\rangle]}{a}$$

for $a \ge 0$. Combining with theorem 3.4 gives us a high probability bound for obtaining an ϵ-optimal solution.

B Approximating Wasserstein Barycenter with the APDRCD algorithm

In this section, we introduce the distributed Wasserstein barycenter problem and present the adapted Accelerated Primal-Dual Coordinate Descent algorithm to approximate the Wasserstein barycenter efficiently for a family of probability measures. We first introduce the setup of the distributed Wasserstein barycenter problem and its entropic regularization. Then we construct the dual form of the problem. We further generalize the APDRCD and the APDGCD algorithms to compute the Wasserstein barycenter and demonstrate its flexibility for computations in the decentralized distributed setting.

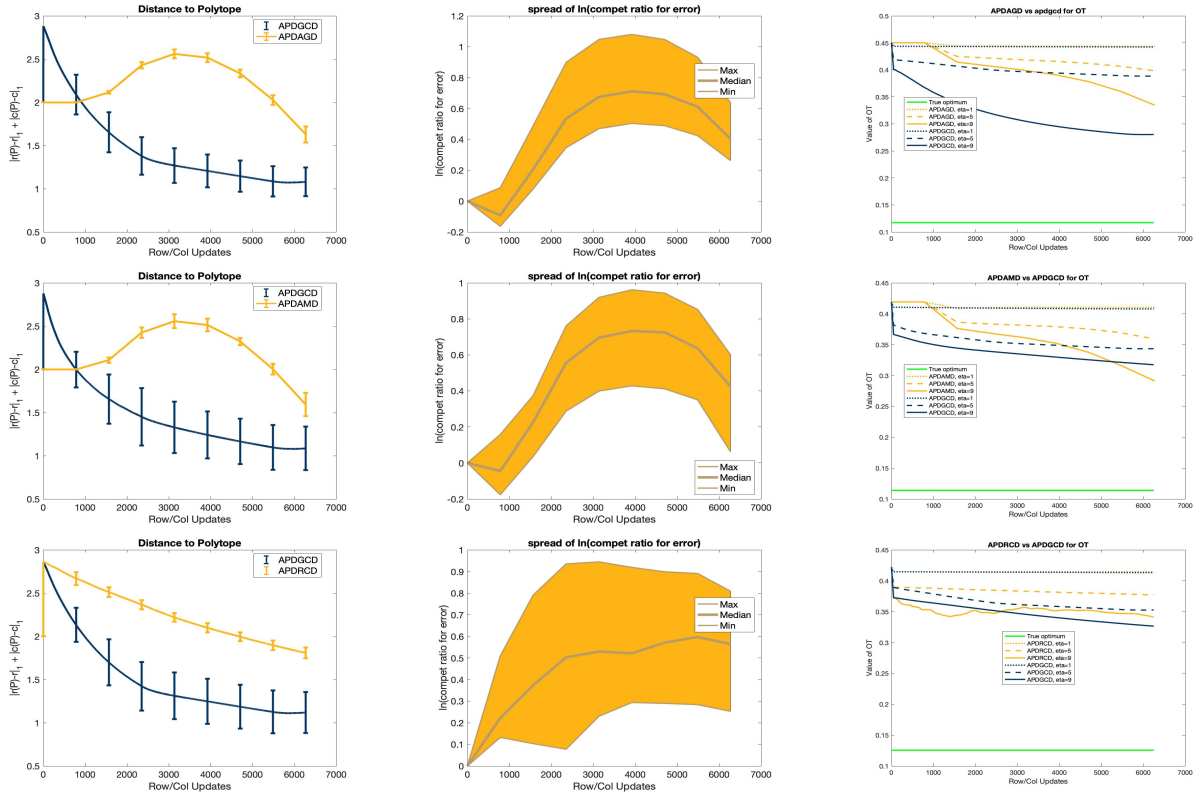

Figure 7: Performance of the APDGCD, APDAGD, APDAMD, and APDRCD algorithms on the MNIST real images. The organization of the images is similar to those in Figure 3.

B.1 Distributed Wasserstein Barycenter Problem

Given a network of probability measures, the optimal transport distance naturally defines the mean representative of the given set of measures. The Wasserstein barycenter problem consider finding the probability measure that is closest to all the given measures in terms of regularized Wasserstein distance. Wasserstein barycenters captur the structure of the given set of objects in a geometrically faithful way. For simplicity, we present the Wasserstein barycenter problem for m discrete measures or histograms with entropic regularizations, but the algorithm could be easily generalized to any continous measures by drawing samples from the given measures.

As introduced in Eq 1, given two probability measures $r, l \in \Delta^n$, we define the regularized Wasserstein distance between r and l as:

$$\mathcal{W}(r,l) := \min_{\pi \in \Pi(r,l)} \langle \pi, C \rangle \tag{21}$$

where $\Pi(r,l) = \{\pi \in \mathbb{R}_+^{n \times n} : \pi \mathbf{1} = r, \pi^T \mathbf{1} = l\}$ is the set of all coupling measures between the measure r and l. Using the entropic regularization as in Eq 2, the regularized OT distance is defined as:

$$\mathcal{W}_\gamma(r,l) := \min_{\pi \in \Pi(r,l)} \{\langle \pi, C \rangle + \gamma H(\pi)\} \tag{22}$$

where $\gamma \geq 0$ is the regularization parameter.

For a given set of probability measures $r_1, r_2, ..., r_m$ and corresponding cost matrices $C_1, C_2, ..., C_m \in \mathbb{R}_+^{n \times n}$, the weighted regularized Wasserstein barycenter problem is therefore:

$$\min_{q \in \Delta^n} \Sigma_{k=1}^m w_k \mathcal{W}_\gamma(r_k, q) \tag{23}$$

where $w_k \geq 0, k = 1, ...m, \Sigma_{k=1}^m w_k = 1$ are the weights over the given measures.

B.2 Network Constrains in the Barycenter Problem

The given set of probability measures form a network, where each measure r_i is held by an agent i on the network. Such a network can be modeled as a fixed connected undirected graph $\mathcal{G} = (V, E)$ where V is the set of m nodes and E is the set of edges. For convenience, we assume that the graph doesn't contain self-loops. The network structure add information constraints: each node can only exchange information with its direct neighbors.

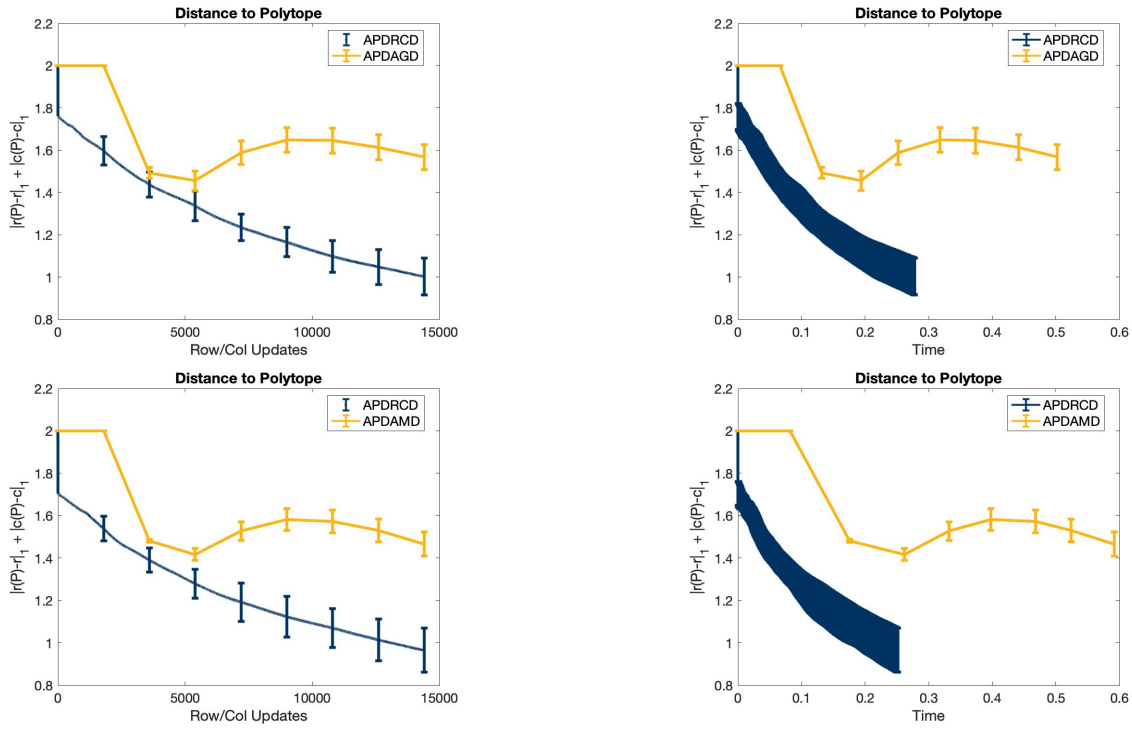

Figure 8: Performance of APDRCD, APDAGD and APDAMD algorithms on $30 * 30$ synthetic images.

The communication constraints can be well-captured by the Laplacian matrix $\overline{W} \in \mathbb{R}^{m\times m}$of the graph $\mathcal{G}$ such that $[\overline{W}]_{ij} = -1$ if $(i,j) \in E$, $[\overline{W}]_{ij} = deg(i)$ if $i = j$, $[\overline{W}]_{ij} = 0$ otherwise. We further define the communitation matrix by $W := \overline{W} \otimes I_n$.

We note some properties of the matrix $\mathcal{W}$. First, for connected and undirected $\mathcal{G}$, both $\overline{W}$ and $\mathcal{W}$ are positive semidefinite. Besides, $\sqrt{\mathcal{W}}\mathrm{q} = 0$ if and only if $q_1 = ... = q_m$ where $\mathrm{q} = [q_1, ..., q_m]^T \in \mathbb{R}^{mn}$.

B.3 Dual Formulation of the Wasserstein Barycenter Problem

To construct the dual problem, we first rewrite problem 23 as:

$$\min_{\substack{q_1,\dots,q_m\in\Delta^n \\ q_1=\dots=q_m}} W_\gamma(\mathrm{r},\mathrm{q}) := \Sigma_{k=1}^m w_k \mathcal{W}_{\gamma(k)}(r_k, q_k) \quad (24)$$

where $\mathrm{r} = [r_1, ..., r_m]^T, \mathrm{q} = [q_1, ..., q_m]^T$.

By using the property that $\sqrt{\mathcal{W}}\mathrm{q} = 0$ if and only if $q_1 = ... = q_m$ where $\mathrm{q} = [q_1, ..., q_m]^T \in \mathbb{R}^{mn}$, the above optimization problem can be further rewritten as:

$$\max_{\substack{q_1,\dots,q_m\in\Delta^n \\ \sqrt{\mathcal{W}}\mathrm{q}=0}} -\Sigma_{k=1}^m w_k \mathcal{W}_{\gamma(k)}(r_k, q_k) \quad (25)$$

Given the above optimization problem with linear constrains, we introduce a vector of dual variables $\lambda = [\lambda_1^T, ..., \lambda_m^T]^T \in \mathbb{R}^{mn}$ for the constraints $\sqrt{\mathcal{W}}\mathrm{q} = 0$. The Lagrangian dual problem for (25) is:

$$\begin{aligned} &\min_{\lambda\in\mathbb{R}^{mn}} \max_{\mathrm{q}\in\mathbb{R}^{mn}} \\ &\left\{\Sigma_{k=1}^m \left\langle \lambda_k, [\sqrt{W}\mathrm{q}]_k \right\rangle - \Sigma_{k=1}^m w_k \mathcal{W}_{\gamma(k)}(r_k, q_k)\right\} \\ &= \min_{\lambda\in\mathbb{R}^{mn}} \Sigma_{k=1}^m w_k \mathcal{W}^*_{\gamma(k),r_k}([\sqrt{\mathcal{W}}\lambda]_k / w_k) \end{aligned} \quad (26)$$

where $\mathcal{W}^*_{\gamma(k),r_k}(\cdot)$ is the Fenchel-Legendre transofrm of $\mathcal{W}_{\gamma(k)}(r_k, \cdot)$. The function $\mathcal{W}^*_{\gamma(k),r_k}(\cdot)$ enjoys the nice property that it is a smooth function with Lipschitz-continous gradient as shown in (Dvurechenskii et al., 2018).

B.4 Approximate Wasserstein Barycenters with Accelerated Coordinate Descent

We apply the accelerated primal-dual randomized coordinate descent (APDRCD) algorithm and accelerated primal-dual greedy coordinate descent (APDGCD) algorithm to solve the pair of primal and dual problems for Wasserstein barycenter. The algorithms are presented in Algorithm 5 and Algorithm 6.

C Further Experimental Results

In this appendix, we provide further comparative experiments between APDGCD algorithm versus APDRCD,

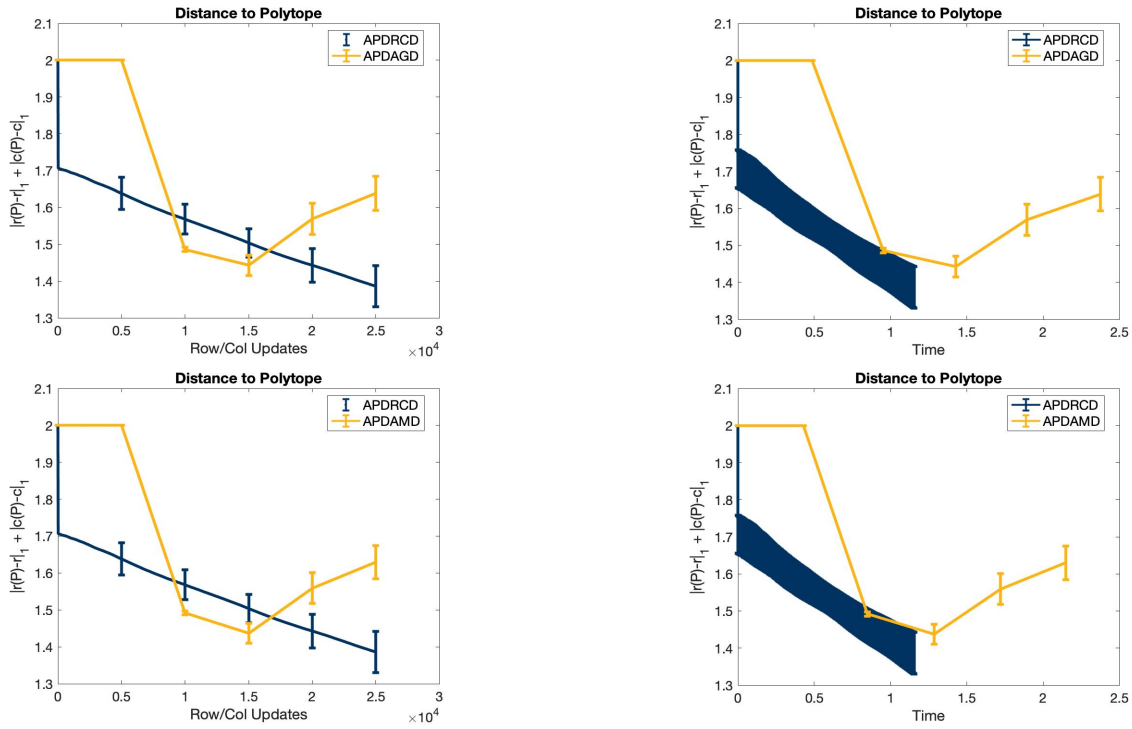

Figure 9: Performance of APDRCD, APDAGD and APDAMD algorithms on $50 * 50$ synthetic images.

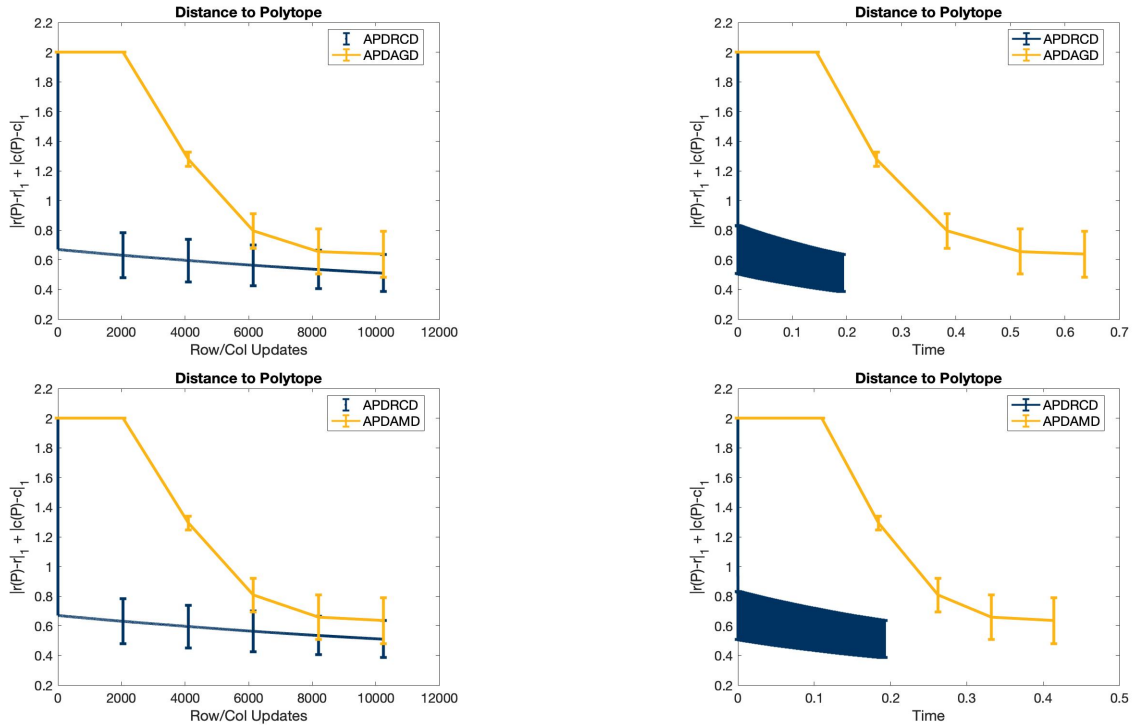

Figure 10: Performance of APDRCD, APDAGD and APDAMD algorithms on 10000 CIFAR10 test images.

APDAGD and APDAMD algorithms. We also provide further results on the performance of the APDRCD algorithm using larger synthetic datasets and CIFAR10. Experiments in Section 4 and Appendix C) show that APDRCD enjoys consistent favorable practical performance than APDAGD, APDAMD on larger synthetic

Algorithm 5: Generalized APDRCD for Computing Wasserstein Barycenters

Input:: Each agent $k \in V$ is assigned measure r_l, an upper bound L for the Lipschitz constant of the gradient of the dual objective, and N.

For all agents $k \in V$, set $r_k = (1-\frac{\epsilon}{8})(r_k + \frac{\epsilon}{n(8-\epsilon)}\mathbf{1})$,$\gamma(k) = \frac{\epsilon}{4mw_k \ln n}$, $\eta_k^0 = \xi_k^0 = \lambda_k^0 = \hat{q}_k^0 = \mathbf{0} \in \mathbb{R}^n$, $A_0 = \alpha_0 = 0$

For each agents $k \in V$:

for $t = 0, ..., N-1$ **do**

Compute α_{t+1} as the largest root of $A_{t+1} := A_t + \alpha_{t+1} = 2L\alpha_{t+1}^2$

Update $\lambda_k^{t+1} = \frac{\alpha_{t+1}\xi_k^t + A_k\eta_k^t}{A_{t+1}}$.

Calculate $\nabla \mathcal{W}^*_{\gamma(k),r_k}(\lambda_k^{t+1})$:

$$[\nabla \mathcal{W}^*_{\gamma(k),r_k}(\lambda_k^{t+1})]_i = \sum_{j=1}^{n} [p_k]_j \frac{\exp(([\lambda]_i - [C_k]_{ij})/\gamma(k))}{\sum_{s=1}^{n} \exp(([\lambda_s - [C_k]_{sj})/\gamma(k))}$$

Share $\nabla \mathcal{W}^*_{\gamma(k),r_k}(\lambda_k^{t+1})$ with $\{j|(i,j) \in E\}$

Randomly choose coordinate s from $\{1, 2, ..., n\}$:

Update

$$[\xi_k^{t+1}]_s = [\xi_k^t]_s - [\alpha_{t+1} \sum_{j=1}^{m} \mathcal{W}_{kj} \nabla \mathcal{W}^*_{\gamma(k),r_k}(\lambda_k^{t+1})]_s$$

Update $[\eta_k^{t+1}]_s = (\alpha_{t+1}[\xi_k^{t+1}]_s + A_k[\eta_k^{t+1}]_s)/A_{t+1}$

Update $[q_k^{t+1}]_s = \frac{1}{A_{t+1}} \sum_{k=0}^{t+1} \alpha_i [q_i(\lambda_k^{t+1})]_s = (\alpha_{t+1}[q_i([\lambda_{t+1}]_k)]_s + A_t[q_k^t]_s)/A_{t+1}$ where $q_k(\cdot)$ is defined as $\nabla \mathcal{W}^*_{\gamma(k),r_k}(\cdot)$

end for

Output: $\mathrm{q}^N = [q_1^T, ..., q_m^T]^T$

Algorithm 6: Generalized APDGCD for Computing Wasserstein Barycenters

Input:: Each agent $k \in V$ is assigned measure r_l, an upper bound L for the Lipschitz constant of the gradient of the dual objective, and N.

For all agents $k \in V$, set $r_k = (1-\frac{\epsilon}{8})(r_k + \frac{\epsilon}{n(8-\epsilon)}\mathbf{1})$,$\gamma(k) = \frac{\epsilon}{4mw_k \ln n}$, $\eta_k^0 = \xi_k^0 = \lambda_k^0 = \hat{q}_k^0 = \mathbf{0} \in \mathbb{R}^n$, $A_0 = \alpha_0 = 0$

For each agents $k \in V$:

for $t = 0, ..., N-1$ **do**

Compute α_{t+1} as the largest root of $A_{t+1} := A_t + \alpha_{t+1} = 2L\alpha_{t+1}^2$

Update $\lambda_k^{t+1} = \frac{\alpha_{t+1}\xi_k^t + A_k\eta_k^t}{A_{t+1}}$.

Calculate $\nabla \mathcal{W}^*_{\gamma(k),r_k}(\lambda_k^{t+1})$:

$$[\nabla \mathcal{W}^*_{\gamma(k),r_k}(\lambda_k^{t+1})]_i = \sum_{j=1}^{n} [p_k]_j \frac{\exp(([\lambda]_i - [C_k]_{ij})/\gamma(k))}{\sum_{s=1}^{n} \exp(([\lambda_s - [C_k]_{sj})/\gamma(k))}$$

Share $\nabla \mathcal{W}^*_{\gamma(k),r_k}(\lambda_k^{t+1})$ with $\{j|(i,j) \in E\}$

Select coordinate s from $\{1, 2, ..., n\}$ where $s = \mathrm{argmax}_s\, |[\nabla \mathcal{W}^*_{\gamma(k),r_k}(\cdot)]_s|$

Update

$$[\xi_k^{t+1}]_s = [\xi_k^t]_s - [\alpha_{t+1} \sum_{j=1}^{m} \mathcal{W}_{kj} \nabla \mathcal{W}^*_{\gamma(k),r_k}(\lambda_k^{t+1})]_s$$

Update $[\eta_k^{t+1}]_s = (\alpha_{t+1}[\xi_k^{t+1}]_s + A_k[\eta_k^{t+1}]_s)/A_{t+1}$

Update $[q_k^{t+1}]_s = \frac{1}{A_{t+1}} \sum_{k=0}^{t+1} \alpha_i [q_i(\lambda_k^{t+1})]_s = (\alpha_{t+1}[q_i([\lambda_{t+1}]_k)]_s + A_t[q_k^t]_s)/A_{t+1}$ where $q_k(\cdot)$ is defined as $\nabla \mathcal{W}^*_{\gamma(k),r_k}(\cdot)$

end for

Output: $\mathrm{q}^N = [q_1^T, ..., q_m^T]^T$

datasets and CIFAR10. Besides, APDGCD enjoys favorable practical performance than APDAGD, APDAMD, and APDRCD algorithms on both synthetic and real datasets. This demonstrates the benefit of choosing the best coordinate to descent to optimize the dual objective function of entropic regularized OT problems in the APDGCD algorithm comparing to choosing the random descent coordinate in the APDRCD algorithm.

C.1 APDGCD algorithm with synthetic images

The generation of synthetic images as well as the evaluation metrics are similar to those in Section 4.1. We respectively present in Figure 4, Figure 5 and Figure 6 the comparisons between APDGCD algorithm versus APDAGD, APDAMD and APDRCD algorithms.

According to Figure 4, Figure 5 and Figure 6, the APDGCD algorithm enjoys better performance than the APDAGD, APDAMD and also the APDRCD algorithms in terms of the iteration numbers in terms of both the evaluation metrics. Besides, at the same number of iteration number, the APDGCD algorithm achieves even faster decrements than other three algorithms with regard to both the distance to polytope and the value of OT metrics during the computing process. This is beneficial in practice for easier tuning and smaller error when the update number is limited.

C.2 APDGCD algorithm with MNIST images

We present comparisons between APDGCD algorithm versus APDAGD, APDAMD, and APDRCD algorithms in Figure 7 with MNIST images.

According to Figure 7, the APDGCD algorithm enjoys better performance than the APDAGD, APDAMD and also the APDRCD algorithms in terms of the iteration numbers in terms of both the evaluation metrics. Furthermore, the convergence of the APDGCD algorithm is faster than other three algorithms with regard to both the distance to polytope and the value of OT met-

rics during the computing process when the number of iterations are small. This is beneficial in practice for easier tuning and smaller error when the total update number is limited.

C.3 APDRCD algorithm with larger synthetic image datasets and CIFAR10

In this section, we provide further experiments on the APDRCD algorithm on larger syntheric image datasets and CIFAR10 dataset. Results are included in Figure 8, Figure 9 and Figure 10. First, we provided results on the comparisons of APDRCD with APDAGD, APDAMD algorithms on larger synthetic datasets, with $n = 30 * 30$ and $n = 50 * 50$. We also provided the results on the CIFAR10 dataset. For each comparison, we provide the plots of the error of the dual variable versus the number of updates; and the error of the dual variable versus CPUtime (CPU: 3.1 GHz Intel Core i7) per iteration. The supplementary experiments show that APDRCD is more stable and achieve faster convergence in both number of row/col updates of the dual variables, and CPU time/iteration. The experimental setup are the same as the previous experiments except the change of dataset, hence omitted here.