Fast Algorithms for Computational Optimal Transport and Wasserstein Barycenter

Wenshuo Guo
UC Berkeley

Nhat Ho
UC Berkeley

Michael I. Jordan
UC Berkeley

Abstract

We provide theoretical complexity analysis for new algorithms to compute the optimal transport (OT) distance between two discrete probability distributions, and demonstrate their favorable practical performance compared to state-of-art primal-dual algorithms. First, we introduce the *accelerated primal-dual randomized coordinate descent* (APDRCD) algorithm for computing the OT distance. We show that its complexity is $\widetilde{\mathcal{O}}(\frac{n^{5/2}}{\varepsilon})$, where n stands for the number of atoms of these probability measures and $\varepsilon > 0$ is the desired accuracy. This complexity bound matches the best known complexities of primal-dual algorithms for the OT problems, including the adaptive primal-dual accelerated gradient descent (APDAGD) and the adaptive primal-dual accelerated mirror descent (APDAMD) algorithms. Then, we demonstrate the improved practical efficiency of the APDRCD algorithm through comparative experimental studies. We also propose a greedy version of APDRCD, which we refer to as *accelerated primal-dual greedy coordinate descent* (APDGCD), to further enhance practical performance. Finally, we generalize the APDRCD and APDGCD algorithms to distributed algorithms for computing the Wasserstein barycenter for multiple probability distributions.

1 Introduction

Optimal transport has become an important topic in statistical machine learning. It finds the minimal cost couplings between pairs of probability measures and provides a geometrically faithful way to compare two probability distributions, with diverse applications in areas including Bayesian nonparametrics (Nguyen, 2013, 2016), scalable Bayesian inference (Srivastava et al., 2015, 2018), topic modeling (Lin et al., 2018), isotonic regression (Rigollet and Weed, 2019), and deep learning (Courty et al., 2017; Arjovsky et al., 2017; Tolstikhin et al., 2018).

Nevertheless, the practical impact of OT has been limited by its computational burden. By viewing the optimal transport distance as a linear programming problem, interior-point methods have been employed as a computational solver, with a best known practical complexity of $\widetilde{\mathcal{O}}(n^3)$ (Pele and Werman, 2009). Recently, Lee and Sidford (Lee and Sidford, 2014) proposed to use the Laplace linear system solver to theoretically improve the complexity of interior-point methods to $\widetilde{\mathcal{O}}(n^{5/2})$. However, it remains a practical challenge to develop efficient interior-point implementations in the high-dimensional settings for OT applications in machine learning.

Several algorithms have been proposed to circumvent the scalability issue of the interior-point methods, including the Sinkhorn algorithm (Sinkhorn, 1974; Knight, 2008; Kalantari et al., 2008; Cuturi, 2013), which has a complexity bound of $\widetilde{\mathcal{O}}(\frac{n^2}{\varepsilon^2})$ where $\varepsilon > 0$ is the desired accuracy (Dvurechensky et al., 2018). The Greenkhorn algorithm (Altschuler et al., 2017) further improves the performance of the Sinkhorn algorithm, with a theoretical complexity of $\widetilde{\mathcal{O}}(\frac{n^2}{\varepsilon^2})$ (Lin et al., 2019). However, for large-scale applications of the OT problem, particularly in randomized and asynchronous scenarios such as computational Wasserstein barycenters, existing literature has shown that neither the Sinkhorn algorithm nor the Greenkhorn algorithm are sufficiently scalable and flexible (Cuturi and Doucet, 2014; Dvurechenskii et al., 2018).

Recent research has demonstrated the advantages of the family of accelerated primal-dual algorithms over the Sinkhorn algorithms. This family includes the adaptive primal-dual accelerated gradient descent (APDAGD)

Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108. Copyright 2020 by the author(s).

algorithm (Dvurechensky et al., 2018) and the adaptive primal-dual accelerated mirror descent (APDAMD) algorithms (Lin et al., 2019), with complexity bounds of $\widetilde{\mathcal{O}}(\frac{n^{2.5}}{\varepsilon})$ and $\widetilde{\mathcal{O}}(\frac{n^2\sqrt{\gamma}}{\varepsilon})$, respectively, where $\gamma \leq \sqrt{n}$ is the inverse of the strong complexity constant of Bregman divergence with respect to the l_∞-norm. In addition, the primal-dual algorithms possess the requisite flexibility and scalability compared to the Sinkhorn algorithm, which is crucial for computational OT problems in large-scale applications (Cuturi and Doucet, 2014; Ho et al., 2019). Specifically, they are flexible enough to be generalized to the computation of the Wasserstein barycenter for multiple probability distributions in decentralized and asynchronous settings (Cuturi and Doucet, 2014; Dvurechenskii et al., 2018).

In the optimization literature, primal-dual methods have served as standard techniques that are readily parallelized for high-dimensional applications (Combettes and Pesquet, 2012; Wainwright et al., 2005). On the other hand, coordinate descent methods have been shown to be well suited to the solution of large-scale machine learning problems (Nesterov, 2012; Richtárik and Takáč, 2016; Fercoq and Richtárik, 2015).

Our contributions. The contributions of the paper are three-fold.

1. We introduce a novel *accelerated primal-dual randomized coordinate descent* (APDRCD) algorithm for solving the OT problem. We provide a complexity upper bound of $\widetilde{\mathcal{O}}(\frac{n^{5/2}}{\varepsilon})$ for the APDRCD algorithm, which is comparable to the complexity of state-of-art primal-dual algorithms for OT problems, such as the APDAGD and APDAMD algorithms (Dvurechensky et al., 2018; Lin et al., 2019). To the best of our knowledge, this is the first accelerated primal-dual coordinate descent algorithm for computing the OT problem.

2. We show that the APDRCD algorithm outperforms the APDAGD and APDAMD algorithms with experiments on both synthetic and real image datasets. To further improve the practical performance of the APDRCD algorithm, we present a greedy version of it which we refer to as the *accelerated primal-dual greedy coordinate descent* (APDGCD) algorithm. To the best of our knowledge, the APDRCD and APDGCD algorithms achieve the best performance among state-of-art accelerated primal-dual algorithms on solving the OT problem.

3. We demonstrate that the APDRCD and APDGCD algorithms are suitable and parallelizable for other large-scale problems besides the OT problem, e.g., approximating the Wasserstein barycenter for a finite set of probability measures stored over a distributed network.

Organization. The remainder of the paper is organized as follows. In Section 2, we provide the formulation of the entropic OT problem and its dual form. In Section 3, we introduce the APDRCD algorithm for solving the regularized OT problem and provide a complexity upper bound for it. We then present the greedy APDGCD algorithm. In Section 4, we present comparative experiments between the APDRCD, the APDGCD algorithms and other primal-dual algorithms including the APDAGD and APDAMD algorithms. We conclude the paper with a few future directions in Section 5. Finally, the proofs of all the results are included in the Appendix A, and the generalized APDRCD and APDGCD algorithms for approximating Wasserstein barycenters are presented in Appendix B. Additional experiments are presented in Appendix C.

Notation. We denote the probability simplex $\Delta^n := \{u = (u_1, \ldots, u_n) \in \mathbb{R}^n : \sum_{i=1}^n u_i = 1, \; u \geq 0\}$ for $n \geq 2$. Furthermore, $[n]$ stands for the set $\{1, 2, \ldots, n\}$ while $\mathbb{R}^n_+$ stands for the set of all vectors in $\mathbb{R}^n$ with nonnegative components for any $n \geq 1$. For a vector $x \in \mathbb{R}^n$ and $1 \leq p \leq \infty$, we denote $\|x\|_p$ as its ℓ_p-norm and $\text{diag}(x)$ as the diagonal matrix with x on the diagonal. For a matrix $A \in \mathbb{R}^{n \times n}$, the notation $\text{vec}(A)$ stands for the vector in $\mathbb{R}^{n^2}$ obtained from concatenating the rows and columns of A. $\mathbf{1}$ stands for a vector with all of its components equal to 1. $\partial_x f$ refers to a partial gradient of f with respect to x. Lastly, given the dimension n and accuracy ε, the notation $a = \mathcal{O}\left(b(n, \varepsilon)\right)$ stands for the upper bound $a \leq C \cdot b(n, \varepsilon)$ where C is independent of n and ε. Similarly, the notation $a = \widetilde{\mathcal{O}}(b(n, \varepsilon))$ indicates the previous inequality may depend on the logarithmic function of n and ε, and where $C > 0$.

2 Problem Setup

In this section, we provide necessary background for the entropic regularized OT problem. The objective function for the problem is presented in Section 2.1 while its dual form as well as the key properties of that dual form are given in Section 2.2.

2.1 Entropic Regularized OT

As shown in (Kantorovich, 1942), the problem of approximating the OT distance between two discrete probability distributions with at most n components is equivalent to the following linear programming problem:

$$\min_{X\in\mathbb{R}^{n\times n}} \langle C, X\rangle \tag{1}$$
$$\text{s.t.} \quad X\mathbf{1} = r,\ X^\top\mathbf{1} = l,\ X \geq 0$$

where X is a *transportation plan*, $C = (C_{ij}) \in \mathbb{R}_+^{n\times n}$ is a cost matrix with non negative elements, and r and l refer to two known probability distributions in the probability simplex Δ^n. The best known practical complexity bound for (1) is $\widetilde{\mathcal{O}}(n^3)$ (Pele and Werman, 2009), while the best theoretical complexity bound is $\widetilde{\mathcal{O}}(n^{2.5})$ (Lee and Sidford, 2014), achieved via interior-point methods. However, these methods are not efficient in the high-dimensional setting of OT applications in machine learning. This motivates the *entropic regularized OT* problem (Cuturi, 2013):

$$\min_{X\in\mathbb{R}_+^{n\times n}} \langle C, X\rangle - \eta H(X) \tag{2}$$
$$\text{s.t.} \quad X\mathbf{1} = r,\ X^\top\mathbf{1} = l,$$

where $\eta > 0$ is the *regularization parameter* and $H(X)$ is the entropic regularization given by $H(X) := -\sum_{i,j=1}^n X_{ij}\log(X_{ij})$. The main focus of the paper is to determine an *ε-approximate transportation plan* $\hat{X} \in \mathbb{R}_+^{n\times n}$ such that $\hat{X}\mathbf{1} = r$ and $\hat{X}^\top\mathbf{1} = l$ and the following bound holds:

$$\langle C, \hat{X}\rangle \ \leq\ \langle C, X^*\rangle + \varepsilon, \tag{3}$$

where X^* is an optimal solution; i.e., an optimal transportation plan for the OT problem (1). To ease the ensuing presentation, we let $\langle C, \hat{X}\rangle$ denote an *ε-approximation* for the OT distance. Furthermore, we define matrix A such that $A\text{vec}(X) := \begin{pmatrix} X\mathbf{1} \\ X^\top\mathbf{1}\end{pmatrix}$ for any $X \in \mathbb{R}^{n\times n}$.

2.2 Dual Entropic Regularized OT

The Lagrangian function for problem (2) is given by

$$\mathcal{L}(X,\alpha,\beta) := \langle C, X\rangle - \eta H(X) + \langle \alpha, r\rangle + \langle \beta, l\rangle - \langle \alpha, X\mathbf{1}\rangle - \langle \beta, X^\top\mathbf{1}\rangle.$$

Given the Lagrangian function, the dual form of the entropic regularized OT problem can be obtained by solving the optimization problem $\min_{X\in\mathbb{R}^{n\times n}} \mathcal{L}(X,\alpha,\beta)$. Since the Lagrangian function $\mathcal{L}(\cdot,\alpha,\beta)$ is strictly convex, that optimization problem can be solved by setting $\partial_X \mathcal{L}(X,\alpha,\beta) = 0$, which leads to the following form of the transportation plan: $X_{ij} = e^{\frac{-C_{ij}+\alpha_i+\beta_j}{\eta}-1}$ for all $i,j \in [n]$. With this solution, we have $\min_{X\in\mathbb{R}^{n\times n}} \mathcal{L}(X,\alpha,\beta) = -\eta\sum_{i,j=1}^n e^{-\frac{C_{ij}-\alpha_i-\beta_j}{\eta}-1} + \langle \alpha, r\rangle + \langle \beta, l\rangle$. The *dual entropic regularized OT* problem is, therefore, equivalent to the following optimization problem:

$$\min_{\alpha,\beta\in\mathbb{R}^n} \varphi(\alpha,\beta)$$
$$:= \eta\sum_{i,j=1}^n e^{-\frac{C_{ij}-\alpha_i-\beta_j}{\eta}-1} - \langle \alpha, r\rangle - \langle \beta, l\rangle \tag{4}$$

Building on Lemma 4.1 in (Lin et al., 2019), the dual objective function $\varphi(\alpha,\beta)$ can be shown to be smooth with respect to $\|\cdot\|_2$ norm:

Lemma 2.1. *The dual objective function φ is smooth with respect to $\|.\|_2$ norm:*

$$\varphi(\lambda_1) - \varphi(\lambda_2) - \langle \nabla\varphi(\lambda_2), \lambda_1 - \lambda_2\rangle \leq \frac{2}{\eta}||\lambda_1 - \lambda_2||_2^2.$$

Proof. The proof is straightforward application of the result from Lemma 4.1 in (Lin et al., 2019). Here, we provide the details of this proof for the completeness. Indeed, invoking Lemma 4.1 in (Lin et al., 2019), we find that

$$\varphi(\lambda_1)-\varphi(\lambda_2)-\langle \nabla\varphi(\lambda_2), \lambda_1 - \lambda_2\rangle \leq \frac{||A||_1^2}{2\eta}||\lambda_1-\lambda_2||_\infty^2.$$

Since $||A||_1$ is equal to the maximum ℓ_1-norm of a column of A and each column of A contains only two nonzero elements which are equal to one, we have $||A||_1 = 2$. Combining with the fact that $||\lambda_1-\lambda_2||_\infty^2 \leq ||\lambda_1 - \lambda_2||_2^2$, we establish the result. □

3 Accelerated Primal-Dual Coordinate Descent Algorithms

In this section, we present and analyze an accelerated primal-dual coordinate descent algorithms to obtain an ε-approximate transportation plan for the OT problem (1). First, in Section 3.1, we introduce the accelerated primal-dual randomized coordinate descent (APDRCD) method for the entropic regularized OT problem. Then, following the approximation scheme of (Altschuler et al., 2017), we show how to approximate the OT distance within the APDRCD algorithm; see Algorithm 2 for the pseudo-code for this problem. Furthermore, we provide theoretical analysis to establish a complexity bound of $\mathcal{O}(\frac{n^{\frac{5}{2}}\sqrt{||C||_\infty \log(n)}}{\varepsilon})$ for the APDRCD algorithm to achieve an ε-approximate transportation plan for the OT problem in Section 3.2. This complexity upper bound of the APDRCD algorithm matches the best known complexity bounds of the APDAGD (Dvurechensky et al., 2018) and APDAMD algorithms (Lin et al., 2019). Finally, to further improve the practical performance of the algorithm, we present a greedy variant—the accelerated primal-dual greedy coordinate descent (APDGCD) algorithm—in Section 3.3.

Algorithm 1: APDRCD $(C, \eta, A, b, \varepsilon')$

```
1  Input: $\{\theta_i|\theta_0 = 1, \frac{1-\theta_{i+1}}{\theta_{i+1}^2} = \frac{1}{\theta_i^2}\}, C_0 = 1, \lambda^0 = z^0 = k = 0, L = \frac{4}{\eta}$
2  while $||Ax^k - b||_1 > \varepsilon'$ do
3      Set $y^k = (1-\theta_k)\lambda^k + \theta_k z^k$
4      Compute $x^k = \frac{1}{C_k}\left(\sum_{j=0}^{k} \frac{x(y^j)}{\theta_j}\right)$
5      Randomly sample one coordinate $i_k$ where $i_k \in \{1, 2, ..., 2n\}$:
6      Update
           $$\lambda_{i_k}^{k+1} = y_{i_k}^k - \frac{1}{L}\nabla_{i_k}\varphi(y^k) \qquad (5)$$
7      Update
           $$z_{i_k}^{k+1} = z_{i_k}^k - \frac{1}{2nL\theta_k}\nabla_{i_k}\varphi(y^k) \qquad (6)$$
8      Update $k = k+1$ and $C_k = C_k + \frac{1}{\theta_k}$
9  end
10 Output: $X^k$ where $x^k = vec(X^k)$
```

3.1 Accelerated Primal-Dual Randomized Coordinate Descent (APDRCD)

We denote by L the Lipschitz constant for the dual objective function φ, which means that $L := \frac{4}{\eta}$, and $x(\lambda) := \underset{x \in \mathbb{R}^{n\times n}}{\arg\max}\left\{-\langle C, x\rangle - \langle A^\top\lambda, x\rangle\right\}$. The APDRCD algorithm is initialized with the auxiliary sequence $\{\theta_i\}$ and two auxiliary dual variable sequences $\{\lambda_i\}$ and $\{\mathbf{z}_i\}$, where the first auxiliary sequence $\{\theta_k\}$ is used for the key averaging step and the two dual variable sequences are used to perform the accelerated randomized coordinate descent on the dual objective function φ as a subroutine. The APDRCD algorithm is composed of two main parts. First, exploiting the convexity property of the dual objective function, we perform a randomized accelerated coordinate descent step on the dual objective function as a subroutine in step 5 and 6. In the second part, we take a weighted average over the past iterations to get a good approximate solution for the primal problem from the approximate solutions to the dual problem (4). Notice that the auxiliary sequence $\{\theta_k\}$ is decreasing and the primal solutions corresponding to the more recent dual solutions have larger weight in this average.

Algorithm 2: Approximating OT by APDRCD

```
Input: $\eta = \frac{\varepsilon}{4\log(n)}$ and $\varepsilon' = \frac{\varepsilon}{8\,\|C\|_\infty}$.
Step 1: Let $\tilde{r} \in \Delta_n$ and $\tilde{l} \in \Delta_n$ be defined as
    $$\left(\tilde{r}, \tilde{l}\right) = \left(1 - \frac{\varepsilon'}{8}\right)(r, l) + \frac{\varepsilon'}{8n}(\mathbf{1}, \mathbf{1}).$$
Step 2: Let $A \in \mathbb{R}^{2n\times n^2}$ and $b \in \mathbb{R}^{2n}$ be defined by
    $$Avec(X) = \begin{pmatrix} X\mathbf{1} \\ X^T\mathbf{1} \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} \tilde{r} \\ \tilde{l} \end{pmatrix}$$
Step 3: Compute $\tilde{X} = \text{APDRCD}\,(C, \eta, A, b, \varepsilon'/2)$ with $\varphi$ defined in 4.
Step 4: Round $\tilde{X}$ to $\hat{X}$ by Algorithm 2 (Altschuler et al., 2017) such that $\hat{X}\mathbf{1} = r$, $\hat{X}^\top\mathbf{1} = l$.
Output: $\hat{X}$.
```

3.2 Complexity Analysis of APDRCD

Given the updates from APDRCD algorithm in Algorithm 1, we have the following result regarding the difference of the values of φ at λ^{k+1} and y^k:

Lemma 3.1. *Given the updates λ^{k+1} and y^k from the APDRCD algorithm, we have the following inequality:*

$$\varphi(\lambda^{k+1}) - \varphi(y^k) \leq -\frac{1}{2L}|\nabla_{i_k}\varphi(y^k)|^2,$$

where i_k is chosen in the APDRCD algorithm.

Proof. For convenience, we define a vector-valued function $h(i_k) \in \mathbb{R}^{2n}$ such that $h(i_k)_i = 1$ if $i = i_k$, and $h(i_k)_i = 0$ otherwise. By the update in Eq. (5) of Algorithm 1, we obtain:

$$\varphi(\lambda^{k+1}) - \varphi(y^k) = \varphi\left(y^k - h(i_k)\cdot\frac{1}{L}\nabla_{i_k}\varphi(y^k)\right) - \varphi(y^k). \tag{7}$$

Due to the smoothness of φ with respect to $\|.\|_2$ norm in Lemma 2.1, the following inequalities hold:

$$\begin{aligned}
&\varphi\left(y^k - h(i_k)\frac{1}{L}\nabla_{i_k}\varphi(y^k)\right) - \varphi(y^k) \\
&\leq \left\langle \nabla\varphi(y^k), -h(i_k)\frac{1}{L}\nabla_{i_k}\varphi(y^k)\right\rangle \\
&\quad + \frac{L}{2}||h(i_k)\frac{1}{L}(\nabla_{i_k}\varphi(y^k))||^2 \\
&= -\frac{1}{L}\left\langle \nabla\varphi(y^k), \nabla_{i_k}\varphi(y^k)h(i_k)\right\rangle + \frac{1}{2L}(\nabla_{i_k}\varphi(y^k))^2 \\
&= -\frac{1}{L}(\nabla_{i_k}\varphi(y^k))^2 + \frac{1}{2L}(\nabla_{i_k}\varphi(y^k))^2 \\
&= -\frac{1}{2L}(\nabla_{i_k}\varphi(y^k))^2. \qquad (8)
\end{aligned}$$

Combining the results of Eq. (7) and Eq. (8) completes the proof of the lemma. □

The result of Lemma 3.1 is vital for establishing an upper bound for $\mathbb{E}_{i_k}\varphi(\lambda^{k+1})$, as shown in the following lemma.

Lemma 3.2. *For each iteration ($k > 0$) of the APDRCD algorithm, we have*

$$\begin{aligned}&\mathbb{E}_{i_k}\left[\varphi(\lambda^{k+1})\right]\\ &\leq (1-\theta_k)\varphi(\lambda^k) + \theta_k\left[\varphi(y^k) + (\lambda - y^k)^T\nabla\varphi(y^k)\right]\\ &+ 2L^2n^2\theta_k^2\bigg(||\lambda - z^k||^2 - \mathbb{E}_{i_k}\left[||\lambda - z^{k+1}||^2\right]\bigg),\end{aligned}$$

where the outer expectation in the above display is taken with respect to the random coordinate i_k in Algorithm 1.

The proof of Lemma 3.2 is in Appendix A.1. Now, equipped with the result of Lemma 3.2, we are ready to provide the convergence guarantee and complexity bound of the APDRCD algorithm for approximating the OT problem. First, we start with the following result regarding an upper bound on k to reach the stopping rule $||A\text{vec}(X^k) - b||_1 \leq \varepsilon'$ for $\varepsilon' = \dfrac{\varepsilon}{8\left\|C\right\|_\infty}$. Here, the outer expectation is taken with respect to the random coordinates i_j in Algorithm 1 for $1 \leq j \leq k$.

Theorem 3.3. *The APDRCD algorithm for approximating optimal transport (Algorithm 2) returns an output X^k that satisfies the stopping criterion $\mathbb{E}\big[||Avec(X^k) - b||_1\big] \leq \varepsilon'$ in a number of iterations k bounded as follows:*

$$k \leq 12n^{\frac{3}{2}}\sqrt{\frac{R+1/2}{\varepsilon}} + 1,$$

where $R := \frac{||C||_\infty}{\eta} + \log(n) - 2\log(\min\limits_{1\leq i,j\leq n}\{r_i, l_i\})$. Here, ε' and η are chosen in Algorithm 2.

The proof of Theorem 3.3 is provided in Appendix A.2. Given an upper bound on k for the stopping rule in Theorem 3.3 where $\varepsilon' = \dfrac{\varepsilon}{8\left\|C\right\|_\infty}$ in Theorem 3.3, we obtain the following complexity bound for the APDRCD algorithm.

Theorem 3.4. *The APDRCD algorithm for approximating optimal transport (Algorithm 2) returns $\hat{X} \in \mathbb{R}^{n\times n}$ satisfying $\hat{X}\mathbf{1} = r$, $\hat{X}^T\mathbf{1} = l$ and $\mathbb{E}[\langle C, \hat{X}\rangle] - \langle C, X^*\rangle \leq \epsilon$ in a total number of*

$$\mathcal{O}\left(\frac{n^{\frac{5}{2}}\sqrt{||C||_\infty\log(n)}}{\varepsilon}\right)$$

arithmetic operations.

The proof of Theorem 3.4 is provided in Appendix A.3. We show in Appendix A.3 that Theorem 3.4 also directly implies a complexity bound for obtaining an ϵ-optimal solution with high probability. Theorem 3.4 indicates that the complexity upper bound of APDRCD matches the best known complexity $\widetilde{\mathcal{O}}(\frac{n^{5/2}}{\varepsilon})$ of the APDAGD (Dvurechensky et al., 2018) and APDAMD (Lin et al., 2019) algorithms. Furthermore, that complexity of APDRCD is better than that of the Sinkhorn and Greenkhorn algorithms, which is $\widetilde{\mathcal{O}}(\frac{n^2}{\varepsilon^2})$, in terms of the desired accuracy ε. Later in Section 4, we demonstrate that the APDRCD algorithm also has better practical performance than APDAGD and APDAMD algorithms on both synthetic and real datasets.

Algorithm 3: APDGCD $(C, \eta, A, b, \varepsilon')$

```
Input: $\{\theta_i|\theta_0 = 1, \frac{1-\theta_{i+1}}{\theta_{i+1}^2} = \frac{1}{\theta_i^2}\}, C_0 = 1, \lambda^0 = z^0 = k = 0, L = \frac{4}{\eta}$
while $||Ax^k - b||_1 > \varepsilon'$ do
    Set $y^k = (1-\theta_k)\lambda^k + \theta_k z^k$
    Compute $x^k = \frac{1}{C_k}\left(\sum_{j=0}^{k}\frac{x(y^j)}{\theta_j}\right)$
    Select coordinate $i_k = \underset{i_k\in\{1,2,\ldots,2n\}}{\text{argmax}}\ |\nabla_{i_k}\varphi(y^k)|$:
      Update
          $\lambda_{i_k}^{k+1} = y_{i_k}^k - \frac{1}{L}\nabla_{i_k}\varphi(y^k)$
      Update
          $z_{i_k}^{k+1} = z_{i_k}^k - \frac{1}{2nL\theta_k}\nabla_{i_k}\varphi(y^k)$
      Update $k = k+1$ and $C_k = C_k + \frac{1}{\theta_k}$
end
Output: $X^k$ where $x^k = vec(X^k)$
```

3.3 Accelerated Primal-Dual Greedy Coordinate Descent (APDGCD)

We next present a greedy version of APDRCD algorithm, which we refer to as the *accelerated primal-dual greedy coordinate descent* (APDGCD) algorithm. The detailed pseudo-code of that algorithm is in Algorithm 3 while an approximating scheme of OT based on the APDGCD algorithm is summarized in Algorithm 4.

Both the APDGCD and APDRCD algorithms follow along the general accelerated primal-dual coordinate descent framework. Similar to the APDRCD algorithm, the algorithmic framework of APDGCD is composed by two main parts: First, instead of performing randomized accelerated coordinate descent on the dual objective function as a subroutine, the APDGCD al-

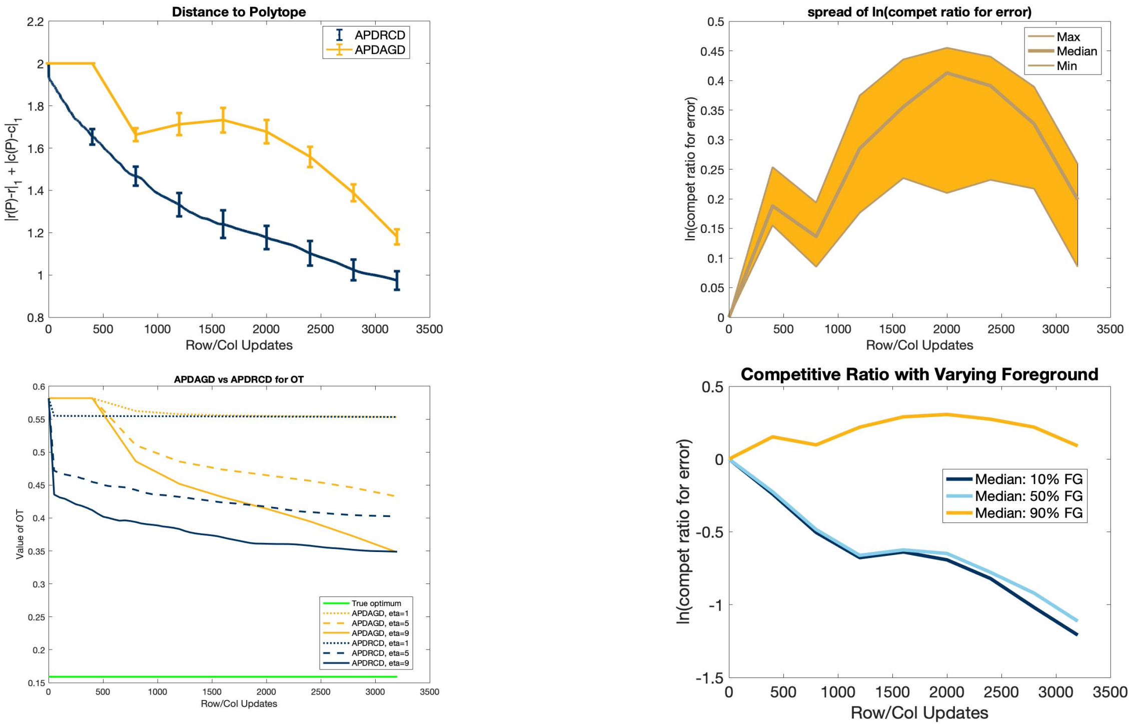

Figure 1: Performance of APDRCD and APDAGD algorithms on the synthetic images. In the top two images, the comparison is based on using the distance $d(P)$ to the transportation polytope, and the maximum, median and minimum of competitive ratios on ten random pairs of images. In the bottom left image, the comparison is based on varying the regularization parameter $\eta \in \{1, 5, 9\}$ and reporting the optimal value of the original optimal transport problem without entropic regularization. Note that the foreground covers 10% of the synthetic images here. In the bottom right image, we compare the algorithms by using the median of competitive ratios with varying coverage ratio of foreground in the range of $\{0.1, 0.5, 0.9\}$.

gorithm chooses the best coordinate that maximizes the absolute value of the gradient of the dual objective function of regularized OT problem among all the coordinates. In the second part, we follow the key averaging step in the APDRCD algorithm by taking a weighted average over the past iterations to get a good approximated solution for the primal problem from the approximated solutions to the dual problem. Since the auxiliary sequence is decreasing, the primal solutions corresponding to the more recent dual solutions have larger weight in this average.

We demonstrate that the APDGCD algorithm enjoys favorable practical performance than APDRCD algorithm on both synthetic and real datasets (cf. Appendix C).

4 Experiments

We carry out comparative experiments between the APDRCD, APDGCD algorithms and the state-of-art primal-dual algorithms for the OT problem including the APDAGD and APDAMD algorithms, on both synthetic images and the MNIST Digits dataset.[1] Due to space constraints, the comparative experiments between the APDGCD algorithm and APDAGD/APDAMD algorithms, further experiments for the APDRCD algorithm on larger synthetic datasets and CIFAR10 dataset are deferred to Appendix C. Note that for the above comparisons, we also utilize the default linear programming solver in MATLAB to obtain the optimal value of the original optimal transport problem without entropic regularization.

4.1 APDRCD Algorithm with Synthetic Images

We compared the performance of the APDRCD algorithm with the APDAGD and APDAMD algorithms on synthetic images. The generation of synthetic images follows the procedure of (Altschuler et al., 2017; Lin et al., 2019). The images are of size 20 × 20 and generated by randomly placing a foreground square in the otherwise black background. For the intensities of the background pixels and foreground pixels, we choose uniform distributions on [0,1] and [0, 50] respectively. We vary the proportion of the size of the foreground

[1] http://yann.lecun.com/exdb/mnist/

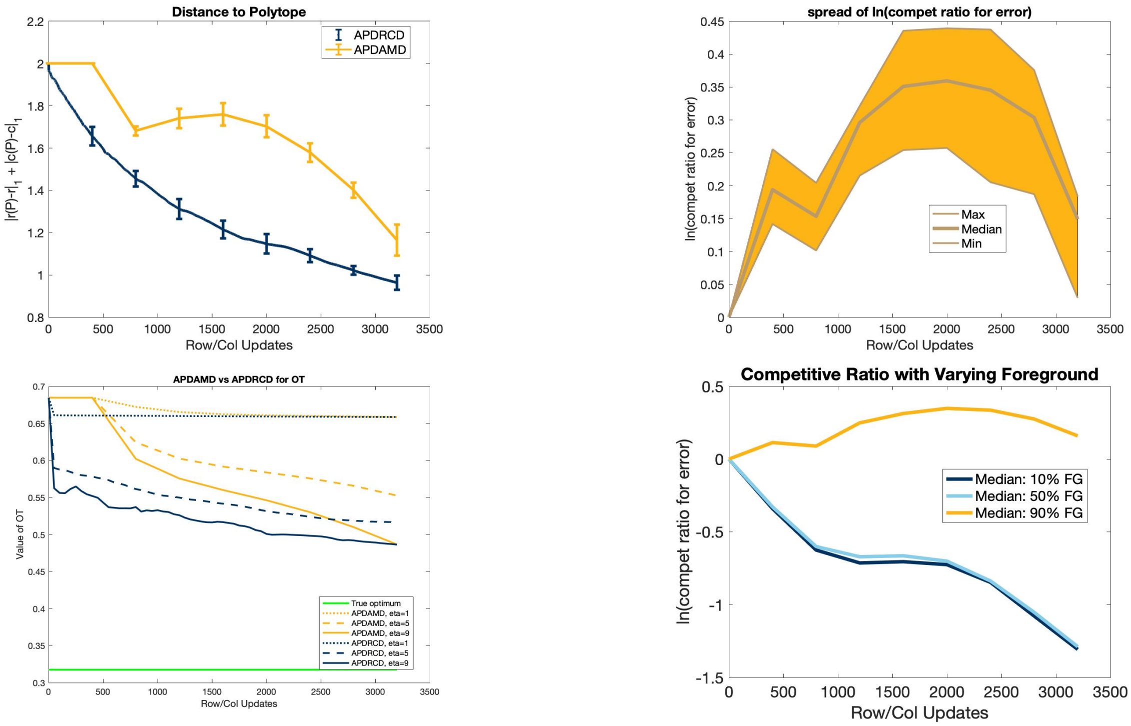

Figure 2: Performance of APDRCD and APDAMD algorithms on the synthetic images. The organization of the images is similar to those in Figure 1.

```
Algorithm 4: Approximating OT by APDGCD
    Input: $\eta = \frac{\varepsilon}{4\log(n)}$ and $\varepsilon' = \frac{\varepsilon}{8\|C\|_\infty}$.
    Step 1: Let $\tilde{r} \in \Delta_n$ and $\tilde{l} \in \Delta_n$ be defined as
        $\left(\tilde{r}, \tilde{l}\right) = \left(1 - \frac{\varepsilon'}{8}\right)(r, l) + \frac{\varepsilon'}{8n}(\mathbf{1}, \mathbf{1})$.
    Step 2: Let $A \in \mathbb{R}^{2n \times n^2}$ and $b \in \mathbb{R}^{2n}$ be defined by
        $Avec(X) = \binom{X\mathbf{1}}{X^T\mathbf{1}}$ and $b = \binom{\tilde{r}}{\tilde{l}}$
    Step 3: Compute $\tilde{X} = \text{APDGCD}(C, \eta, A, b, \varepsilon'/2)$ with $\varphi$ defined in 4.
    Step 4: Round $\tilde{X}$ to $\hat{X}$ by Algorithm 2 (Altschuler et al., 2017) such that $\hat{X}\mathbf{1} = r$, $\hat{X}^\top\mathbf{1} = l$.
    Output: $\hat{X}$.
```

square in 0.1, 0.5, 0.9 of the full size of the image and implement all the algorithms on different kinds of synthetic images.

Evaluation metric: We utilize the metrics from (Altschuler et al., 2017). The first metric is the distance between the output of the algorithm and the transportation polytope, $d(X) := ||r(X) - r||_1 + ||l(X) - 1||_1$, where $r(X)$ and $l(X)$ are the row and column marginal vectors of the output matrix X while r and l stand for the true row and column marginal vectors. The second metric is the competitive ratio, defined by $\log(\frac{d(X1)}{d(X2)})$ where $d(X1)$ and $d(X2)$ refer to the distance between the outputs of two algorithms and the transportation polytope.

Experimental settings and results: We perform two pairwise comparative experiments for the APDRCD algorithm versus the APDAGD and APDAMD algorithms by running these algorithms with ten randomly selected pairs of synthetic images. We also evaluate all the algorithms with varying regularization parameter $\eta \in \{1, 5, 9\}$ and the optimal value of the original OT problem without the entropic regularization, as suggested by (Altschuler et al., 2017; Lin et al., 2019).

We present the results in Figure 1 and Figure 2. The APDRCD algorithm has better performance than the APDAGD and APDAMD algorithms in terms of the iterations. When the number of iterations is small, the APDRCD algorithm achieves faster and more stable decrements than the other two algorithms with regard to both the distance to polytope and the value of OT during the computing process, which is beneficial for the purposes of tuning and illustrates the advantage of using randomized coordinate descent on the dual

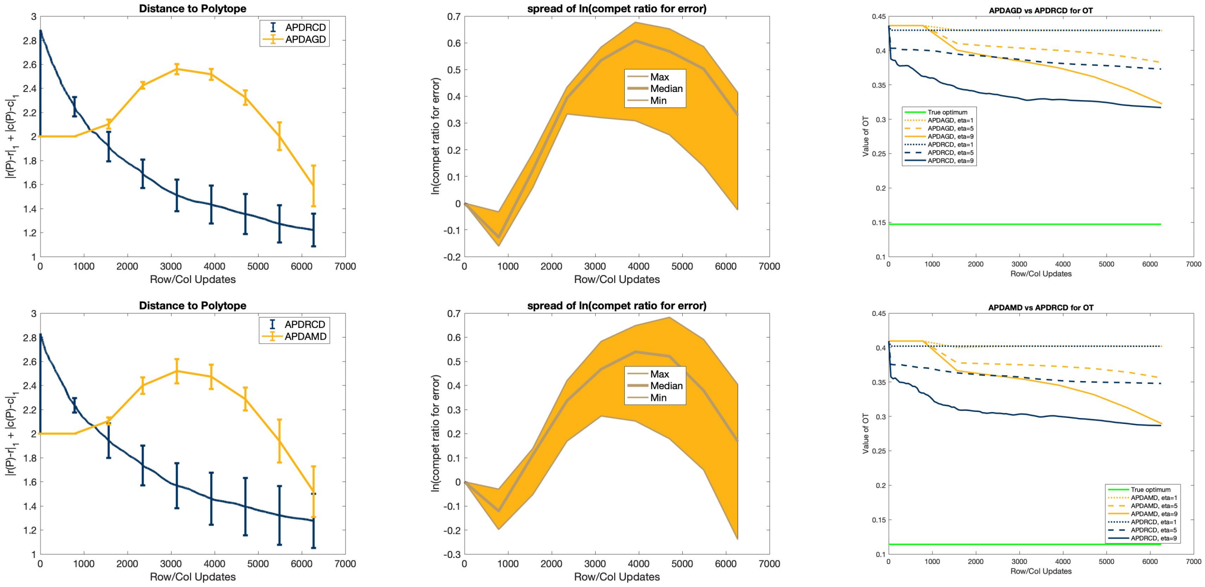

Figure 3: Performance of the APDRCD, APDAGD and APDAMD algorithms on the MNIST images. In the first row of images, we compare the APDRCD and APDAGD algorithms in terms of iteration counts. The leftmost image specifies the distances $d(P)$ to the transportation polytope for two algorithms; the middle image specifies the maximum, median and minimum of competitive ratios on ten random pairs of MNIST images; the rightmost image specifies the values of regularized OT with varying regularization parameter $\eta = \{1, 5, 9\}$. In addition, the second row of images present comparative results for APDRCD versus APDAMD.

regularized problem.

4.2 APDRCD Algorithm with MNIST Images

We compare the APDRCD algorithm and the APDAGD and APDAMD algorithms on MNIST images with the same set of evaluation metrics. The image pre-processing follows the same pre-processing procedure as suggested in (Lin et al., 2019). We omit the details for the sake of brevity.

We present the results with the MNIST images in Figure 3 with various values for the regularization parameter $\eta \in \{1, 5, 9\}$. We also evaluate the algorithms with the optimal value of the original OT problem without entropic regularization. As shown in Figure 3, the APDRCD algorithm outperforms both the APDAGD and APDAMD algorithms on the MNIST dataset in terms of the number of iterations. Additionally, the APDRCD algorithm displays faster and smoother convergence than the other algorithms at small iteration numbers with regard to both the evaluation metrics, which implies the advantage that it is easier to be tuned in practice.

5 Discussion

We have proposed and analyzed new accelerated primal-dual coordinate descent algorithms for approximating the optimal transport distance between two discrete probability measures. These accelerated primal-dual coordinate descent algorithms have comparable theoretical complexity as that of existing accelerated primal-dual algorithms while enjoying better experimental performance. Furthermore, we show that the APDRCD and APDGCD algorithms are suitable for other large-scale problems apart from computing the OT distance; we propose extensions that approximate the Wasserstein barycenters for multiple probability distributions.

There are several directions for future work. Given the favorable practical performance of the APDRCD and the APDGCD algorithms over existing primal-dual algorithms, it is of interest to carry out more experiments of the distributed APDRCD and APDGCD algorithms for computing Wasserstein barycenters. Another important direction is to construct fast distributed algorithms for the case of time-varying and directed machine networks to approximate the Wasserstein barycenters. It remains as an interesting and important open research question how the dynamics of the network affect the performance of these algorithms.

Acknowledgements

We thank Tianyi Lin for many helpful discussions. This work was supported in part by the Mathematical Data Science program of the Office of Naval Research under grant number N00014-18-1-2764.

References

J. Altschuler, J. Weed, and P. Rigollet. Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration. In *NIPS*, pages 1964–1974, 2017. 1, 3, 4, 6, 7, 13, 14

M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *ICML*, pages 214–223, 2017. 1

P. L. Combettes and J. C. Pesquet. Primal-dual splitting algorithm for solving inclusions with mixtures of composite, Lipschitzian, and parallel-sum type monotone operators. *Set-Valued and Variational Analysis*, 20(2):307–330, 2012. 2

N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1853–1865, 2017. 1

M. Cuturi. Sinkhorn distances: lightspeed computation of optimal transport. In *NIPS*, pages 2292–2300, 2013. 1, 3

M. Cuturi and A. Doucet. Fast computation of Wasserstein barycenters. In *ICML*, pages 685–693, 2014. 1, 2

P. Dvurechenskii, D. Dvinskikh, A. Gasnikov, C. Uribe, and A. Nedich. Decentralize and randomize: Faster algorithm for Wasserstein barycenters. In *Advances in Neural Information Processing Systems*, pages 10760–10770, 2018. 1, 2, 16

P. Dvurechensky, A. Gasnikov, and A. Kroshnin. Computational optimal transport: Complexity by accelerated gradient descent is better than by Sinkhorn´s algorithm. In *ICML*, pages 1367–1376, 2018. 1, 2, 3, 5

O. Fercoq and P. Richtárik. Accelerated, parallel, and proximal coordinate descent. *SIAM Journal on Optimization*, 25(4):1997–2023, 2015. 2

N. Ho, V. Huynh, D. Phung, and M. I. Jordan. Probabilistic multilevel clustering via composite transportation fistance. *AISTATS*, 2019. 2

B. Kalantari, I. Lari, F. Ricca, and B. Simeone. On the complexity of general matrix scaling and entropy minimization via the RAS algorithm. *Mathematical Programming*, 112(2):371–401, 2008. 1

L. V. Kantorovich. On the translocation of masses. In *Dokl. Akad. Nauk. USSR (NS)*, volume 37, pages 199–201, 1942. 2

P. A. Knight. The Sinkhorn–Knopp algorithm: Convergence and applications. *SIAM Journal on Matrix Analysis and Applications*, 30(1):261–275, 2008. 1

Y. T. Lee and A. Sidford. Path finding methods for linear programming: Solving linear programs in $\widetilde{O}(\text{sqrt}(\text{rank}))$ iterations and faster algorithms for maximum flow. In *FOCS*, pages 424–433. IEEE, 2014. 1, 3

T. Lin, Z. Hu, and X. Guo. Sparsemax and relaxed Wasserstein for topic sparsity. *ArXiv Preprint: 1810.09079*, 2018. 1

T. Lin, N. Ho, and M. I. Jordan. On efficient optimal transport: An analysis of greedy and accelerated mirror descent algorithms. *ArXiv Preprint arXiv:1901.06482*, 2019. 1, 2, 3, 5, 6, 7, 8, 11

Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012. 2

X. Nguyen. Convergence of latent mixing measures in finite and infinite mixture models. *Annals of Statistics*, 4(1):370–400, 2013. 1

X. Nguyen. Borrowing strength in hierarchical Bayes: Posterior concentration of the Dirichlet base measure. *Bernoulli*, 22(3):1535–1571, 2016. 1

O. Pele and M. Werman. Fast and robust earth movers distance. In *ICCV*. IEEE, 2009. 1, 3

P. Richtárik and M. Takáč. Parallel coordinate descent methods for big data optimization. *Mathematical Programming*, 156(1-2):433–484, 2016. 2

P. Rigollet and J. Weed. Uncoupled isotonic regression via minimum Wasserstein deconvolution. *Information and Inference: A Journal of the IMA*, 8(4): 691–717, 2019. 1

R. Sinkhorn. Diagonal equivalence to matrices with prescribed row and column sums. *Proceedings of the American Mathematical Society*, 45(2):195–198, 1974. 1

S. Srivastava, V. Cevher, Q. Dinh, and D. Dunson. WASP: Scalable Bayes via barycenters of subset posteriors. In *AISTATS*, pages 912–920, 2015. 1

S. Srivastava, C. Li, and D. Dunson. Scalable Bayes via barycenter in Wasserstein space. *Journal of Machine Learning Research*, 19(8):1–35, 2018. 1

I. Tolstikhin, O. Bousquet, S. Gelly, and B. Schoelkopf. Wasserstein auto-encoders. In *ICLR*, 2018. 1

M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Map estimation via agreement on trees: Message-passing and linear programming. *IEEE Transactions on Information Theory*, 51(11):3697–3717, 2005. 2