Projection Robust Wasserstein Barycenters

Minhui Huang [1] **Shiqian Ma** [2] **Lifeng Lai** [1]

Abstract

Collecting and aggregating information from several probability measures or histograms is a fundamental task in machine learning. One of the popular solution methods for this task is to compute the barycenter of the probability measures under the Wasserstein metric. However, approximating the Wasserstein barycenter is numerically challenging because of the curse of dimensionality. This paper proposes the projection robust Wasserstein barycenter (PRWB) that has the potential to mitigate the curse of dimensionality, and a relaxed PRWB (RPRWB) model that is computationally more tractable. By combining the iterative Bregman projection algorithm and Riemannian optimization, we propose two algorithms for computing the RPRWB, which is a max-min problem over the Stiefel manifold. The complexity of arithmetic operations of the proposed algorithms for obtaining an ϵ-stationary solution is analyzed. We incorporate the RPRWB into a discrete distribution clustering algorithm, and the numerical results on real text datasets confirm that our RPRWB model helps improve the clustering performance significantly.

1. Introduction

The Wasserstein barycenter (WB) problem is attracting a lot of interest recently due to its wide applications in statistics and machine learning, including but not limited to image processing (Rabin et al., 2011), multi-level clustering (Ho et al., 2017), and text mining (Ye et al., 2017a;b). The WB serves as a geodesic interpolation between two or more distributions. It aggregates the underlying geometric structures of the input distributions under the Wasserstein metric. Therefore, the WB model provides deep insight when collecting information from probability distributions.

---
[1]Department of Electrical and Computer Engineering, University of California, Davis, CA, USA [2]Department of Mathematics, University of California, Davis, CA, USA. Correspondence to: Shiqian Ma <sqma@ucdavis.edu>.

However, computing the WB for a set of probability distributions is notoriously hard. The hardness comes from two aspects: the representation of the measure support and the potential curse of dimensionality. In many applications, the underlying distributions are unknown and we only have sampled data from these distributions. We wish to estimate the WB using the sampled data only. Therefore, the task reduces to compute WB from sampled discrete measures on a fixed number of support points. However, solving the free-support discrete WB is still very difficult (Borgwardt & Patterson, 2019). In this paper, we mainly consider the fixed-support WB problem. On the other hand, computing fixed-support WB can be challenging if the dimension of the problem is high. Recent theoretical developments have revealed that the sample complexity of approximating Wasserstein distances grows exponentially in dimension (Dudley, 1969; Weed & Bach, 2019). For the WB problem, (Altschuler & Boix-Adsera, 2021) has proved that computing WB is NP-hard since its runtime scales exponentially in the dimension. However, the sample complexity of original WB is still not well understood. Since the WB model minimizes a sum of Wasserstein distances, we conjecture that the WB problem would also have the issue of curse of dimensionality. To overcome this difficulty, we adopt a technique used in computing the Wasserstein distance (Paty & Cuturi, 2019) to the WB problem. The resulting projection robust WB (PRWB) model is an inf-sup-inf problem, which is computationally intractable. We further propose a relaxation of PRWB that is computationally more tractable. The idea of the new technique is to project the sampled data to a common low dimensional subspace and compute the WB of the projected data as an approximation to the original WB. The resulting problem is a max-min problem with Stiefel manifold constraint, and we propose two algorithms that can find an ϵ-stationary point of it efficiently.

Related work: Most existing works for fixed-support WB focus on designing efficient algorithms. (Cuturi & Doucet, 2014) proposed to add an entropy regularizer and solve its dual problem that is smooth. This idea was further studied by (Benamou et al., 2015) under the name of iterative Bregman projection (IBP) algorithm. The convergence behavior of IBP was studied in (Kroshnin et al., 2019). There exist some other algorithms for computing fixed-support WB, including the dual accelerated gradient descent method

(Kroshnin et al., 2019; Lin et al., 2020b), the stochastic gradient descent method (Claici et al., 2018), the Bregman ADMM method (Ye et al., 2017b) and the interior-point method (Ge et al., 2019). On the other hand, people have proposed some efficient ways to mitigate the curse of dimensionality of the optimal transport (OT) problem (Niles-Weed & Rigollet, 2019). Specifically, (Bonneel et al., 2015) proposed the sliced Wasserstein distance and applied it to the WB problem. The sliced OT projects the sampled data to a random line and reduces the problem to a one-dimensional OT, which can be solved very efficiently by sorting. This idea motivated the work of (Paty & Cuturi, 2019; Niles-Weed & Rigollet, 2019) that suggest projecting the data to a low dimensional subspace. This leads to the projection robust Wasserstein (PRW) distance, and algorithms for computing it include (Lin et al., 2020a; Huang et al., 2020).

Contributions: Our main contributions are below.

(i) We propose a projection robust Wasserstein barycenter (PRWB) model. The PRWB model has the potential to mitigate the curse of dimensionality by projecting the probability measures onto a low dimensional subspace. Since PRWB is still numerically challenging to solve, we further propose a relaxation of PRWB (RPRWB) that is more tractable. Our numerical results indicate that RPRWB is more robust to noise compared with the WB.

(ii) We propose two algorithms: Riemannian block coordinate descent (RBCD) and Riemannian gradient ascent with IBP algorithm (RGA-IBP), to compute the RPRWB. The RGA-IBP incorporates the IBP algorithm to a Riemannian gradient ascent algorithm, and the RBCD is based on a reformulation of the max-min problem that is suitable for BCD type algorithms. The complexities of arithmetic operations of both algorithms for obtaining an ϵ-stationary point are analyzed.

(iii) We conduct extensive numerical experiments to show the robustness and the practicality of the RPRWB model. We adopt RPRWB to the discrete distribution (D2) clustering algorithm, which we call the projected D2 clustering. We test this new algorithm on the real text datasets, and the numerical results show that the projected D2 clustering achieves better performance than the D2 clustering.

2. Optimal Transport and Wasserstein Barycenter

In this section, we review some background in optimal transport and Wasserstein barycenter.

Denote $\mathscr{P}(\mathbb{R}^d)$ as the set of Borel probability measures in $\mathbb{R}^d$ and $\mathscr{P}_2(\mathbb{R}^d)$ as a subset of $\mathscr{P}(\mathbb{R}^d)$ whose elements have finite second moment. The 2-Wasserstein distance between probability measures $\mu, \nu \in \mathscr{P}_2(\mathbb{R}^d)$ is defined as

$$\mathcal{W}(\mu,\nu) := \left(\inf_{\pi \in \Pi(\mu,\nu)} \int \|x - y\|^2 d\pi(x,y) \right)^{1/2}, \quad (1)$$

where $\Pi(\mu,\nu)$ is the set of all joint distributions with marginals μ and ν. We denote $\Delta^q = \{u \in \mathbb{R}^q_+ | u^\top \mathbf{1}_q = 1\}$ as the probability simplex in $\mathbb{R}^q$. The WB of m probability measures $\boldsymbol{\mu} := \{\mu^l\}_{l \in [m]}$ is the solution of the following problem:

$$\inf_{\nu \in \mathscr{P}_2(\mathbb{R}^d)} \mathcal{WB}(\boldsymbol{\mu},\boldsymbol{\omega}) := \sum_{l=1}^m \omega^l \mathcal{W}^2(\mu^l,\nu), \quad (2)$$

where $\boldsymbol{\omega} \in \Delta^m$ is a given weighting vector and $[m] := \{1, \ldots, m\}$. We use Proj_E to denote the orthogonal projector onto E for any $E \in \mathcal{G}_k$, where the Grassmannian $\mathcal{G}_k := \{E \in \mathbb{R}^d | \dim(E) = k\}$ is the set of all k-dimensional subspaces of $\mathbb{R}^d$. For Wasserstein distance, (Paty & Cuturi, 2019) proposed the projection robust Wasserstein distance as follows:

$$\mathcal{P}_k(\mu,\nu) := \sup_{E \in \mathcal{G}_k} \mathcal{W}(\text{Proj}_E\mu, \text{Proj}_E\nu). \quad (3)$$

That is, the probability measures μ and ν are projected onto the k-dimensional subspace E, and the Wasserstein distance between the projected measures is computed as an approximation to the original Wasserstein distance. Moreover, to measure the worst case approximation, the subspace E that maximizes this Wasserstein distance is sought. The study in (Niles-Weed & Rigollet, 2019) shows that the projection robust Wasserstein distance is able to improve the sample complexity from $O(n^{-1/d})$ for Wasserstein distance to $O(n^{-1/k})$, where n denotes the nubmer of sampled data. This is a significant improvement since usually $k \ll d$ for high dimensional OT. Therefore, the projection robust Wasserstein distance can mitigate the curse of dimensionality.

3. Projection Robust Wasserstein Barycenter

Our projection robust Wasserstein barycenter is motivated by the success of the projection robust Wasserstein distance and the sliced Wasserstein barycenter proposed in (Bonneel et al., 2015). By replacing the Wasserstein distance in (2) with the PRW distance (3), the fixed-support PRWB is defined as the solution of the following problem:

$$\inf_{\nu \in \mathscr{P}_2(\mathbb{R}^d)} \sum_{l=1}^m \omega^l \mathcal{P}_k^2(\mu^l,\nu). \quad (4)$$

Plugging (3) into (4), we have

$$
\inf_{\nu \in \mathscr{P}_2(\mathbb{R}^d)} \sum_{l=1}^{m} \omega^l \sup_{E_\ell \in \mathcal{G}_k} \mathcal{W}^2(\mathrm{Proj}_{E_\ell} \mu^l, \mathrm{Proj}_{E_\ell} \nu)
$$

$$
= \inf_{\nu \in \mathscr{P}_2(\mathbb{R}^d)} \sum_{l=1}^{m} \omega^l \sup_{U_\ell \in \mathrm{St}(d,k)} \inf_{\pi^l \in \Pi(\mu^l, \nu)} \quad (5)
$$

$$
\int \|U_\ell^\top (x^l - y)\|^2 d\pi^l(x^l, y).
$$

According to (Paty & Cuturi, 2019)[Proposition 1], PRW is a well defined distance over $\mathscr{P}_2(\mathbb{R}^d)$ and can be formulated as a sup-inf problem. Moreover, the support of the barycenter is fixed and our target barycenter ν lies on a probability simplex. Our PRWB formulation (5) is a inf-sup-inf problem over m Stiefel manifolds. Solving (5) directly is extremely difficult, because of the complex inf-sup-inf structure and also the existence of m Stiefel manifolds constraints. Therefore, we propose the following relaxation to PRWB (5) that is more computationally tractable:

$$
\sup_{E \in \mathcal{G}_k} \inf_{\nu \in \mathscr{P}_2(\mathbb{R}^d)} \sum_{l=1}^{m} \omega^l \mathcal{W}^2(\mathrm{Proj}_E \mu^l, \mathrm{Proj}_E \nu)
$$

$$
= \sup_{U \in \mathrm{St}(d,k)} \inf_{\nu \in \mathscr{P}_2(\mathbb{R}^d)} \sum_{l=1}^{m} \omega^l \inf_{\pi^l \in \Pi(\mu^l, \nu)} \quad (6)
$$

$$
\int \|U^\top (x^l - y)\|^2 d\pi^l(x^l, y).
$$

More specifically, we first use a common projector $\mathrm{Proj}_E(\cdot)$ for all PRW distances, and then we switch the order of $\sup$ and the first $\inf$. The relaxed model (6) searches for a common low-dimensional subspace, the union of all subspaces of m PRW distances, that maximizes the barycenter objective. Roughly speaking, we solve an easier problem in a low-dimensional subspace to approximate the original WB problem. We call (6) the Relaxed PRWB (RPRWB) and focus on solving this relaxed version in the rest of the paper. We first study some properties of RPRWB. The following proposition shows the existence of the optimal subspace E^*.

Proposition 1 *Given a probability measure set $\boldsymbol{\mu}$, the support of the barycenter ν, the weight vector $\boldsymbol{\omega}$, and $k \in [d]$, there exists an optimal E^* for the problem (6).*

Notice that the target barycenter ν lies on a probability simplex. This combined with Proposition 1 indicates that the fixed-support RPRWB problem can be written as a max-min problem. Using $U \in \mathrm{St}(d,k)$ to denote an orthonormal basis of E, the RPRWB can be formulated as

$$
\max_{U \in \mathrm{St}(d,k)} \min_{\pi^l \in \Pi(\mu^l, \nu)} \sum_{l=1}^{m} \omega^l \int \|U^\top (x^l - y)\|^2 d\pi^l(x^l, y),
$$
(7)

where $\mathrm{St}(d,k)$ denotes the Stiefel manifold, x^l is the support of μ^l and y is the support of ν.

Remark 2 *We remark here that analyzing the sample complexity PRWB is highly nontrivial and the analysis in (Niles-Weed & Rigollet, 2019) for PRW does not apply here. In fact, we are not aware of any results for the sample complexity of the empirical discrete WB problem. There are only some computational hardness results (Altschuler & Boix-Adsera, 2021) showing WB is NP-hard because of the "curse of dimensionality". Since the WB problem minimizes the sum of a set of Wasserstein distances, we conjecture that the "curse of dimensionality" should be inherited by WB. Deriving the sample complexity of WB and PRWB is an important future topic.*

In this paper, we consider solving WB for a set of discrete distributions. Specifically, we denote $X^l = [x_1^l; \cdots; x_n^l] \in \mathbb{R}^{d \times n}$ as the support of each μ^l and write $\mu^l = \sum_{i=1}^{n} p_i^l \delta_{x_i^l}$, where $p^l \in \Delta^n$ and δ_x denotes the Dirac function at x. The support of the barycenter ν is given and denoted as $Y = \{y_1, \ldots, y_n\} \in \mathbb{R}^{d \times n}$. Therefore, the barycenter can be written as $\nu = \sum_{i=1}^{n} q_j \delta_{y_j}$ with $q \in \Delta^n$. Denote $\boldsymbol{\pi} = \{\pi^l\}_{l \in [m]}$. Throughout this paper, we denote $\mathcal{M} = \mathrm{St}(d,k)$. Computing the fixed-support RPRWB is equivalent to solving

$$
\max_{U \in \mathcal{M}} \min_{q \in \Delta^n} \sum_{l=1}^{m} \omega^l \mathcal{W}^2(\mathrm{Proj}_E \mu^l, \mathrm{Proj}_E \nu)
$$

$$
= \max_{U \in \mathcal{M}} \min_{\boldsymbol{\pi} \in \Pi(\boldsymbol{p})} f(\boldsymbol{\pi}, U),
$$
(8)

where $f(\boldsymbol{\pi}, U) := \sum_{l=1}^{m} \omega^l \sum_{i,j=1}^{n} \pi_{i,j}^l \|U^\top(x_i^l - y_j)\|^2$, and $\Pi(\boldsymbol{p}) = \{\boldsymbol{\pi} \mid \pi^l \in \mathbb{R}_+^{n \times n}, \ \pi^l \mathbf{1} = p^l, (\pi^l)^\top \mathbf{1} = (\pi^{l+1})^\top \mathbf{1}, \ l \in [m]\}$.

4. The Riemannian Gradient Ascent and Riemannian BCD Algorithms

In this section, we propose two algorithms for solving (8): RGA-IBP and RBCD. We can show that both algorithms find an ϵ-stationy point of (8) defined as follows.

Definition 3 *We call $(\hat{\boldsymbol{\pi}}, \hat{U}) \in \Pi(\boldsymbol{p}) \times \mathcal{M}$ an ϵ-stationary point of the fixed-support RPRWB problem (8), if the following two inequalities hold:*

$$
\|\mathrm{grad}_U f(\hat{\boldsymbol{\pi}}, \hat{U})\|_F \leq \epsilon, \quad (9)
$$

$$
f(\hat{\boldsymbol{\pi}}, \hat{U}) - f(\boldsymbol{\pi}^*(\hat{U}), \hat{U}) \leq \epsilon, \quad (10)
$$

where $\mathrm{grad}_U f(\hat{\boldsymbol{\pi}}, \hat{U})$ is the Riemannian gradient w.r.t. U, $\boldsymbol{\pi}^*(\hat{U})$ is the optimal solution of the inner minimization problem of (8) when fixing U as $\hat{U}$. The corresponding ϵ-approximate barycenter $q \in \Delta^n$ can be computed as $q = (\hat{\pi}^l)^\top \mathbf{1}, \forall l \in [m]$.

Before we present the algorithms, we define some useful notation first.

Algorithm 1 The RGA-IBP Algorithm

1: **Input:** $\{\mu^l = (X^l, p^l)\}_{l \in [m]}$, $\{Y\}$, accuracy tolerance $\epsilon > 0$. Set parameters τ, η, and ρ as in (24).
2: **Initialization:** $U_0 \in \text{St}(d, k)$.
3: **for** $t = 0, 1, 2, \ldots,$ **do**
4: $\quad \boldsymbol{\pi}_{t+1} = \text{IBPsolver}(\boldsymbol{\mu}, Y, U_t, \eta, \epsilon)$;
5: $\quad \xi_{t+1} = \text{Proj}_{T_{U_t}\mathcal{M}}(2V_{\boldsymbol{\pi}_{t+1}}U_t)$;
6: $\quad U_{t+1} = \text{Retr}_{U_t}(\tau \xi_{t+1})$;
7: $\quad$ **if** $\|\xi_{t+1}\|_F \leq \epsilon$ **then**
8: $\quad\quad$ break;
9: $\quad$ **end if**
10: **end for**
11: Output: $\hat{U} = U_t$, $\hat{\boldsymbol{\pi}} = \boldsymbol{\pi}_{t+1}$.

Definition 4 *(Cost and Correlation Matrices) Given the support vectors* $\{X^l\}_{l \in [m]}$ *and* Y, *the cost matrices, denoted as* $\{C^l\}_{l \in [m]}$, *are defined as* $C^l_{i,j} = \|x^l_i - y_j\|^2, \forall l \in [m]$. *The correlation matrix, denoted as* $V_{\boldsymbol{\pi}}$, *is defined as* $V_{\boldsymbol{\pi}} = \sum_{l=1}^m \omega^l \sum_{i,j=1}^n \pi^l_{i,j}(x^l_i - y_j)(x^l_i - y_j)^\top \in \mathbb{R}^{d \times d}$.

4.1. The Riemannian Gradient Ascent with IBP Iterations

The RGA-IBP algorithm is a natural extension of the RGAS algorithm (Riemannian gradient ascent with Sinkhorn's iteration) that was proposed by (Lin et al., 2020a) for computing the projection robust Wasserstein distance. Here we extend it to solve the RPRWB problem (8). The RGA-IBP algorithm solves the following problem, which is obtained by adding an entropy regularization to (8).

$$\max_{U \in \mathcal{M}} \min_{\boldsymbol{\pi} \in \Pi(\boldsymbol{p})} f_\eta(\boldsymbol{\pi}, U) :=$$
$$\sum_{l=1}^m \omega^l \sum_{i,j=1}^n \pi^l_{i,j} \|U^\top(x^l_i - y_j)\|^2 - \eta H(\pi^l), \quad (11)$$

where $H(\pi) := -\sum_{i,j=1}^n (\pi_{i,j} \log \pi_{i,j} - \pi_{i,j})$ is the entropy regularizer, and $\eta > 0$ is a weighting parameter. By further defining

$$f_\eta(U) := \min_{\boldsymbol{\pi} \in \Pi(\boldsymbol{p})} f_\eta(\boldsymbol{\pi}, U), \quad (12)$$

we know that (11) is equivalent to the following Riemannian optimization problem with smooth objective $f_\eta(U)$:

$$\max_{U \in \mathcal{M}} f_\eta(U). \quad (13)$$

Problem (13) can be naturally solved by a Riemannian gradient ascent algorithm whose t-th iteration is:

$$U_{t+1} := \text{Retr}_{U_t}(\tau \text{grad} f_\eta(U_t))$$

where Retr denotes the retraction operation, $\text{grad} f_\eta$ denotes the Riemannian gradient of f_η, and $\tau > 0$ is a step size.

Moreover, it is easy to verify that

$$\text{grad} f_\eta(U) = \text{Proj}_{T_U\mathcal{M}}(2V_{\boldsymbol{\pi}^*_\eta(U)}U), \quad (14)$$

where $T_U\mathcal{M}$ denotes the tangent space of $\mathcal{M}$ at U, and $\boldsymbol{\pi}^*_\eta(U)$ is the optimal solution of (12) that can be found by the IBP algorithm (see details in Algorithm 4).

The RGA-IBP algorithm is detailed in Algorithm 1, where the IBP solver solves (12) up to an accuracy ϵ (see Algorithm 4 in the supplementary material).

4.2. The Riemannian Block Coordinate Descent Algorithm

Notice that the RGA-IBP requires to solve an optimization problem (12) in each iteration using an iterative solver. This can be quite expensive in practice. In this section, we propose the RBCD algorithm that can alleviate this computational burden. The RBCD algorithm presented here can be regarded as an extension of the algorithm recently proposed in (Huang et al., 2020) for computing the projection robust Wasserstein distance.

First, note that the optimization problem in (12) is convex and we have the following result about its dual.

Lemma 5 *The dual problem of* (12) *is equivalent to the following problem:*

$$\max_{\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{m \times n}, \sum_{l=1}^m \omega^l v^l = 0} -\sum_{l=1}^m \omega^l \left\{ \sum_{i,j=1}^n \pi^l_{ij} - \langle u^l, p^l \rangle \right\}, \quad (15)$$

where $\pi^l_{ij} = [\pi(u^l, v^l, U)]_{ij}$ *is given by:*

$$\pi(u^l, v^l, U)_{ij} = \exp\left(-\frac{\|U^\top(x^l_i - y_j)\|^2}{\eta} + u^l_i + v^l_j\right). \quad (16)$$

As a result, we know that (11) *is equivalent to:*

$$\min_{\substack{\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{m \times n}, U \in \mathcal{M}, \\ \sum_{l=1}^m \omega^l v^l = 0}} g(\boldsymbol{u}, \boldsymbol{v}, U) :=$$
$$\sum_{l=1}^m \omega^l \left\{ \sum_{i,j=1}^n \pi^l_{i,j} - \langle u^l, p^l \rangle \right\}. \quad (17)$$

Note that (17) has three block variables and it is suitable for block coordinate descent method. Our RBCD for solving (17) updates the iterates as follows:

$$\boldsymbol{u}_{t+1} := \underset{\boldsymbol{u}}{\text{argmin}} \, g(\boldsymbol{u}, \boldsymbol{v}_t, U_t) \quad (18)$$

$$\boldsymbol{v}_{t+1} := \underset{\boldsymbol{v}: \sum_{l=1}^m \omega^l v^l = 0}{\text{argmin}} \, g(\boldsymbol{u}_{t+1}, \boldsymbol{v}, U_t) \quad (19)$$

$$U_{t+1} := \text{Retr}_{U_t}(-\tau \text{grad}_U g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}, U_t)). \quad (20)$$

Algorithm 2 The RBCD Algorithm

1: **Input:** $\{\mu^l = (X^l, p^l)\}_{l \in [m]}, \{Y\}$, accuracy tolerance $\epsilon > 0$. Set parameters τ, η and ρ as in (27).
2: **Initialization:** $U_0 \in \text{St}(d, k)$, $\boldsymbol{u}_0, \boldsymbol{v}_0 \in \mathbb{R}^{m \times n}$,
3: **for** $t = 0, 1, 2, \ldots,$ **do**
4:      Compute $\boldsymbol{u_{t+1}}, \boldsymbol{v_{t+1}}$ by (21)-(22);
5:      Compute U_{t+1} by (20);
6:      **if** $\sum_{l=1}^m \omega^l \|q_t^l - \bar{q}_t\|_1 \leq \underline{w}^{3/2}\epsilon/(12\bar{c})$, and $\eta\|\text{grad}_U g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}, U_t)\|_F \leq \epsilon/3$ **then**
7:          break;
8:      **end if**
9: **end for**
10: **Output:** $q = \bar{q}_t = \sum_{l=1}^m \omega^l q_t^l$, $\hat{U} = U_t$, and $\hat{\pi}^l = Round(\pi^l(u_{t+1}^l, v_t^l, U_t), p^l, q), \forall l \in [m]$.

According to (Benamou et al., 2015)[Propositions 1, 2], $\boldsymbol{u}$ and $\boldsymbol{v}$ steps in (18) and (19) are the same as the steps in the IBP algorithm, and they admit closed-form solutions below:

$$u_{t+1}^l = u_t^l + \log \frac{p^l}{\pi(u_t^l, v_t^l, U_t)\mathbf{1}}, \qquad \forall l \in [m], \quad (21)$$

$$v_{t+1}^l = v_t^l + \log \frac{q_{t+1}}{q_t^l}, \qquad \forall l \in [m], \quad (22)$$

where we denote $q_t^l = (\pi(u_{t+1}^l, v_t^l, U_t))^\top \mathbf{1}$ and $q_{t+1} = \exp(\sum_{l=1}^m \omega^l \log q_t^l)$. Notice that (21)-(22) renormalize the sum of rows and columns of each π^l to be p^l and q_{t+1}, which yields $\langle q_t^l, \mathbf{1} \rangle = 1, \forall l \in [m]$. Moreover, the update (20) requires to compute $\text{grad}_U g$, and from (16) and (17) we know that

$$\text{grad}_U g(\boldsymbol{u}, \boldsymbol{v}, U) = \text{Proj}_{\text{T}_U \mathcal{M}}(-\frac{2}{\eta}V_{\boldsymbol{\pi}(\boldsymbol{u},\boldsymbol{v},U)}U). \quad (23)$$

By combining (18)-(23), we can summarize the details of the RBCD in Algorithm 2, in which we have adopted the following notation for the simplicity of presentation:

$$\bar{c} := \max_l \|C^l\|_\infty, \quad \omega = \min_l \omega^l.$$

Note that in Algorithm 2 we adopted a rounding procedure for the output. This is because that π computed according to (16) does not necessarily lie in the constraint set $\Pi(\boldsymbol{p})$. The rounding procedure proposed in (Altschuler et al., 2017) and outlined in Algorithm 3 can help round the solution to set $\Pi(\boldsymbol{p})$. Note that this rounding procedure is also adopted in the IBP algorithm and thus in the RGA-IBP algorithm.

5. Convergence Analysis

In this section, we give the complexities of both the iteration number and the arithmetic operations for both RGA-IBP and RBCD for obtaining an ϵ-stationary point of (8) as defined in

Algorithm 3 $Round(\pi, p, q)$

1: **Input:** $\pi \in \mathbb{R}^{n \times n}, p \in \mathbb{R}^n, q \in \mathbb{R}^n$.
2: $X = \text{Diag}(x)$ with $x_i = \frac{p_i}{[\pi\mathbf{1}]_i} \wedge 1$
3: $\pi' = X\pi$
4: $Y = \text{Diag}(y)$ with $y_j = \frac{q_j}{[(\pi')^\top \mathbf{1}]_j} \wedge 1$
5: $\pi'' = \pi'Y$
6: $err_p = p - \pi''\mathbf{1}, err_q = q - (\pi'')^\top \mathbf{1}$
7: **Output:** $\pi'' + err_p err_q^\top /\|err_p\|_1$.

Definition 3. The proofs are provided in the supplementary materials.

The next theorem and corollary are for RGA-IBP algorithm.

Theorem 6 *Choose parameters*

$$\tau = \frac{1}{8L_2\bar{c} + 2\rho L_1^2}, \eta = \frac{\epsilon}{4\log(n) + 2}, \rho = 2\bar{c} + \frac{4\bar{c}^2}{\eta}. \quad (24)$$

The Algorithm 1 returns an ϵ-stationary point defined in Definition 3 in

$$T = O(\log(n)^2 L_1^2 \bar{c}^4 k \epsilon^{-4}) \quad (25)$$

iterations.

Corollary 7 *The per iteration arithmetic operations complexity of Algorithm 1 is $O(mn^2 dk + mn^2\bar{c}^6\log(n)^2\epsilon^{-6})$. Therefore, the total arithmetic operations complexity of Algorithm 1 is*

$$O(mn^2 dk + mn^2\bar{c}^6\log(n)^2\epsilon^{-6}) \cdot O(\log(n)^2 L_1^2\bar{c}^4 k\epsilon^{-4})$$
$$= \log(n)^2 L_1^2\bar{c}^4 kO(mn^2 dk\epsilon^{-4} + mn^2\bar{c}^6\log(n)^2\epsilon^{-10}). \quad (26)$$

The next theorem and corollary are for RBCD algorithm.

Theorem 8 *Choose parameters*

$$\tau = \frac{1}{4L_2\bar{c}/\eta + \rho L_1^2}, \eta = \frac{\epsilon}{4\log(n) + 2}, \rho = \frac{2\bar{c}}{\eta} + \frac{4\sqrt{k}\bar{c}^2}{\eta^2}. \quad (27)$$

The Algorithm 2 returns an ϵ-stationary point defined in Definition 3 in

$$T = O\left(L_1^2\bar{c}^5\log(n)\sqrt{k}\underline{\omega}^{-3}\epsilon^{-3}\right) \quad (28)$$

iterations.

Corollary 9 *The per iteration arithmetic operations complexity of Algorithm 2 is $O(mn^2 d)$. Therefore, the total arithmetic operations complexity of Algorithm 2 is*

$$O\left(mn^2\log(n)dL_1^2\bar{c}^5 k^{\frac{3}{2}}\underline{\omega}^{-3}\epsilon^{-3}\right). \quad (29)$$

Remark 10 *Comparing* (26) *with* (29)*, we see that RBCD has a better complexity dependence on ϵ and $\bar{c}$. However, RBCD has an extra term $\underline{\omega} \leq \frac{1}{m}$. Therefore, RGA-IBP has a better theoretical complexity when m is large.*

6. Numerical Experiments

In this section, we conduct numerical experiments on both synthetic datasets and real datasets to evaluate the proposed RPRWB model (8)[1]. For the synthetic dataset, we consider solving RPRWB for a set of Gaussian distributions, which has closed-form solutions (Agueh & Carlier, 2011). We compare the convergence rate of WB and RPRWB to the ground truth for the sampled discrete distributions, as well as the robustness against noise for the RPRWB model. For real datasets, we incorporate the RPRWB model to the discrete distribution (D2) clustering algorithm (Ye et al., 2017b) and test it on text datasets. All experiments are conducted on a Linux server with a 32-core Intel Xeon CPU (E5-2667, v4, 3.20GHz per core).

6.1. Synthetic Dataset

Multi-variable Gaussian Distributions: It is well-known that the Wasserstein barycenter of a set of multi-variable Gaussian distributions $\{\mu^l\}_{l\in[m]}$ with $\mu^l = \mathcal{N}(a^l, \Sigma^l)$, where a^l is the mean and Σ^l is the covariance matrix, has a closed-form formula (Agueh & Carlier, 2011). Specifically, we have the following theorem.

Theorem 11 ((Agueh & Carlier, 2011)) *Let $\mu^1, ..., \mu^m$ be Gaussian distributions with respective means $a^1, \ldots, a^m$ and covariance matrices $\Sigma^1, ..., \Sigma^m$. The barycenter of $\mu^1, ..., \mu^m$ with weights $\omega^1, ..., \omega^m$ is the Gaussian distribution with mean $\bar{a} = \sum_{l=1}^{m} \omega^l a^l$ and covariance matrix Σ defined as the only positive definite matrix satisfying the equation*

$$S = \sum_{l=1}^{m} \omega^l \left(S^{1/2} \Sigma^l S^{1/2} \right)^{1/2}. \quad (30)$$

In this subsection, we compute the WB and RPRWB of a given set of zero-mean multi-variable Gaussian distributions $\{\mu^l\}_{l\in[m]}, \mu^l = \mathcal{N}(0, \Sigma^l)$. We set $\omega^l = \frac{1}{m}, \forall l \in [m]$ in all experiments.

The dependence of RPRWB on k. We first explore the dependence of the objective function value on k. For each μ^l, we sample an empirical measure μ_n^l. Specifically, we sample n points according to the Gaussian distribution $\mu^l = \mathcal{N}(0, \Sigma^l)$ to form the support matrix $X^l \in \mathbb{R}^{d\times n}$ and set $p_i^l = \frac{1}{n}, i \in [n]$. We set each of the covariance matrices

[1]Code available at https://github.com/mhhuang95/PRWB.

Σ^l to be a SPD matrix with rank k^*. Therefore, X^l lies in a k^*-dimensional subspace and the barycenter of $\{\mu_n^l\}_{l\in[m]}$ should be in a $(m \times k^*)$-dimensional subspace. The support of the barycenter $Y \in \mathbb{R}^{d\times n}$ is obtained by applying k-means clustering on $X = [X^1; \cdots; X^m] \in \mathbb{R}^{d\times mn}$. We set parameters as $d = 100, m = 3, n = 10$. We further set the step size $\tau = 0.0005$ for both RBCD and RGA-IBP algorithms and $\eta = 0.5 \cdot \text{mid}(\{C^l\}_{l\in[m]})$, where $\text{mid}(\{C^l\}_{l\in[m]})$ is the median of the entries of $\{C^l\}_{l\in[m]}$.
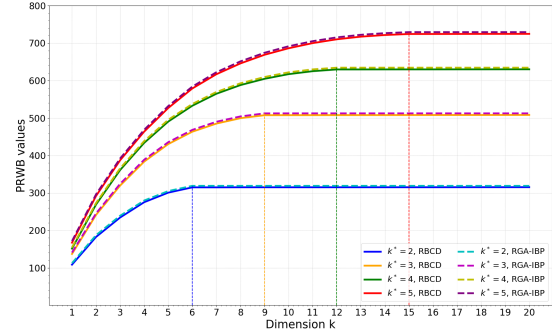

Figure 1. RPRWB function value versus projection dimension k. We run Algorithms 2 and 1 on different k and averaging over 100 runs,

We run both RBCD and RGA-IBP for solving (8) with different k^* and k, and report the results in Figure 1. From Figure 1 we see that the RPRWB values computed by the two algorithms are almost the same. We also notice that the RPRWB value increases when $k < m \times k^*$ and remains as a constant when $k \geq m \times k^*$, which verifies the fact that the barycenter of $\{\mu_n^l\}_{l\in[m]}$ lies in a $(m \times k^*)$-dimensional subspace.

Robustness Against Noise. We further conduct experiments on comparing the robustness of WB and RPRWB against noise. Specifically, we add Gaussian noise $\sigma \mathcal{N}(0, I)$, where σ is the noise level, to the discrete support $\{X^l\}_{l\in[m]}$. We compare the relative error of the objective function value for WB and RPRWB under different noise level σ. The relative error for WB and RPRWB is defined as

$$\text{Relative Error} = \frac{OBJ(\{\mu_n^l\}_\sigma) - OBJ(\{\mu_n^l\}_0))}{OBJ(\{\mu_n^l\}_0)},$$

where $\{\mu_n^l\}_\sigma$ denotes the distributions after adding noise $\sigma \mathcal{N}(0, I)$ and OBJ denotes the objective function of WB (the discrete version of (2)) or RPRWB (8). We set parameters as $d = 100, m = 3, n = 10, \sigma \in [0.01, 0.1, 1, 2, 4, 7, 10]$. We choose the step size $\tau = 0.001$ when $\sigma < 7$ and $\tau = 0.0005$ otherwise for both RBCD

and RGA-IBP algorithms and $\eta = 0.5 \cdot \text{mid}(\{C^l\}_{l \in [m]})$. The results are shown in Figure 2, which shows that the proposed RPRWB model is more robust to noise compared to the WB.
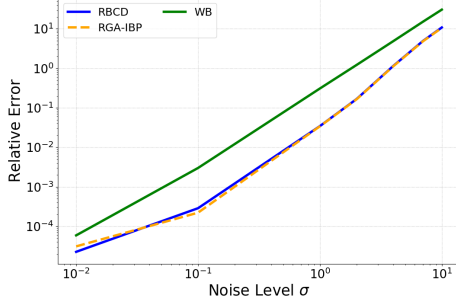

Figure 2. Relative error of the WB and RPRWB function value on different noise level σ. The results are averaged on 100 runs.

Convergence rate to the ground truth. We further consider approximating the Wasserstein barycenter for a set of continuous distributions by sampling data. Note that (Niles-Weed & Rigollet, 2019) proved that for a so-called spiked transport model, the mean projection robust Wasserstein distance between the sampled empirical distributions is $O(n^{-1/k})$, which improves the corresponding complexity of $O(n^{-1/d})$. We conjecture that similar results hold for WB and RPRWB and give some numerical evidence in this section. We set $d = 10, m = 2, k = 2$. The covariance matrices $\Sigma^1, \Sigma^2 \in \mathbb{R}^{d \times d}$ are diagonal matrices with $\Sigma^1(1, 1) = 10.1, \Sigma^2(2, 2) = 10.1$ and the rest of diagonal elements are all 0.1. In this case, a 2-dimensional subspace catches most of the information about the barycenter. We then sample n points as the support for each of μ^l. To have a better estimation, the probability for x_i^l is computed according to the Gaussian PDF:

$$Prob(x_i^l) = \frac{1}{(2\pi)^{d/2}\det(\Sigma^l)^{1/2}} e^{-\frac{1}{2}(x_i^l)^T(\Sigma^l)^{-1}(x_i^l)}.$$

We sampled the support of the barycenter Y according to a uniform distribution over $[-2, 2]^d$. The barycenter Mean Estimation Error is defined as

$$MEE = |OBJ(\{\mu^l\}) - OBJ(\{\mu_n^l\})|,$$

where the ground truth objective function $OBJ(\{\mu^l\})$ is calculated by solving (30) and $OBJ(\{\mu_n^l\})$ is the sampled barycenter objective function value of the WB or RPRWB model. We set the step size $\tau = 0.05$ for both RBCD and RGA-IBP algorithms and $\eta = 0.5 \cdot \text{mid}(\{C^l\}_{l \in [m]})$ and select $n \in \{20, 50, 100, 250, 500, 1000\}$. The results are shown in Figure 3, which shows that the proposed RPRWB model converges to the ground truth much faster than the WB.
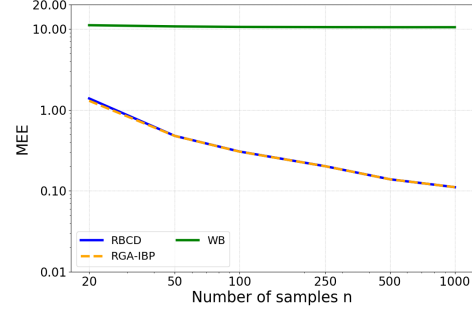

Figure 3. Mean Estimation Error (MEE) on different n. The results are averaged on 500 runs.

Computational time comparison. We compare the mean computational time of the WB solved by the IBP algorithm (Benamou et al., 2015) and the proposed RPRWB solved by RBCD and RGA-IBP. We set $d = 100, m = 3, k = 2$ and select $n \in \{20, 50, 100, 250, 500, 1000\}$. We generate the support matrices $X^l \in \mathbb{R}^{d \times n}$ from $\mu^l = \mathcal{N}(0, \Sigma^l)$ by empirical sampling. The support of the barycenter $Y \in \mathbb{R}^{d \times n}$ is obtained by k-means clustering. We further set the step size $\tau = 0.01$ for both RBCD and RGA-IBP algorithms and $\eta = 0.5 \cdot \text{mid}(\{C^l\}_{l \in [m]})$. We stop the RBCD algorithm when $\eta \|\text{grad}_U g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}, U_t)\|_F \leq \epsilon, \frac{1}{m} \sum_{l=1}^m \|q_t^l - \bar{q}_t\|_1 \leq \epsilon$, and the RGA-IBP algorithm when $\|\text{grad}_U f(\xi_{t+1})\|_F \leq \epsilon$, and we set $\epsilon = 10^{-4}$. The results are shown in Figure 4, which shows that RBCD always runs faster than RGA-IBP. Note that the IBP for solving WB runs much faster than the other two algorithms, and this is because the latter two solve a more difficult problem.
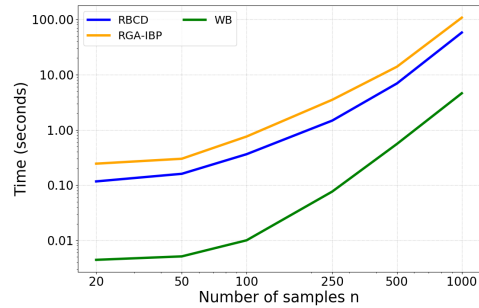

Figure 4. Computational time of the WB model solved by the IBP algorithm and the RPRWB model solved by RBCD and RGA-IBP algorithms on different n. The results are averaged on 100 runs.

6.2. Real Dataset: text data

We consider the discrete distributions (D2) clustering model proposed in (Ye et al., 2017b), which requires to solve the
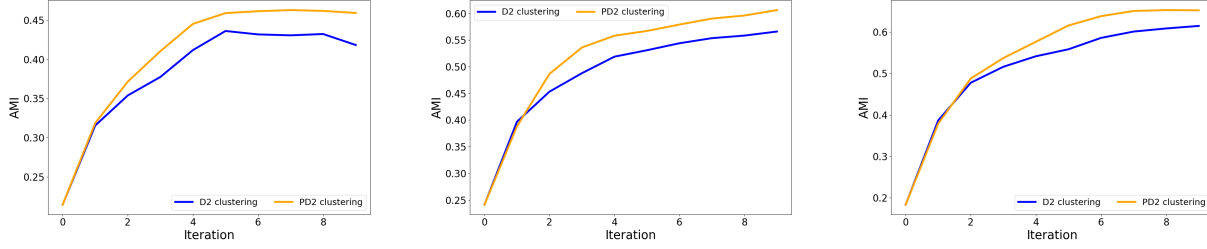
Figure 5. AMI scores for each iteration. **Left:** the "Reuters Subset" dataset, **Middle:** the "BBCsport Abstract" dataset, **Right:** the "BBCnews Abstract" dataset. The results are averaged on 5 runs.

free-support discrete Wasserstein barycenter model:

$$
\min_{\nu} \frac{1}{m} \sum_{l=1}^{m} \mathcal{W}(\mu^l, \nu)
$$
$$
= \min_{\boldsymbol{\pi} \in \Pi(\boldsymbol{p}), Y \in \mathbb{R}^{d \times n}} \frac{1}{m} \sum_{l=1}^{m} \sum_{i,j=1}^{n} \pi_{i,j}^l \|x_i^l - y_j\|^2, \tag{31}
$$

Note that there are two block variables: $\boldsymbol{\pi}$ and Y. (Ye et al., 2017b) proposed to solve (31) using an alternating mimization algorithm. That is, one alternatingly minimizes the objective function (31) with respect to one variable and with the other one fixed. This procedure is repeated until no progress can be made. When Y is fixed, problem (31) becomes a fixed-support WB problem. When $\Pi = [\pi^1; \cdots; \pi^m]$ is fixed, we have the following closed-form solution for Y:

$$
y_i = \frac{1}{mq_i} \sum_{l=1}^{m} \sum_{j=1}^{n} \pi_{i,j}^l x_j^l, \tag{32}
$$

which can be written more compactly as $Y = \frac{1}{m} X \Pi^\top \mathrm{diag}(1/q)$.

Since we have numerically demonstrated that RPRWB might be a better model than WB, we propose to replace the WB problem in D2 clustering by RPRWB. We call the resulting algorithm projection robust D2 clustering (PD2 clustering). More details of the D2 and PD2 clusterings can be found in the supplementary material. We compare the performance of D2 and PD2 clusterings on three text datasets listed in Table 1. The "Reuters Subset" is a 5-class subset of the "Reuters" dataset [2]. The "BBCnews Abstract" and "BBCsport Abstract"[3] (Greene & Cunningham, 2006) are truncated versions of 2,225 and 737 posts. Each document retains only the title and the first sentence of the original post.

Preprocessing. We follow the idea of treating each document as a bag of word-vectors. For all three datasets in

---

[2]https://www.nltk.org/book/ch02.html
[3]http://mlg.ucd.ie/datasets/bbc.html

Table 1. Text Datasets. N is the number of data, n is the number of samples, and K is the number of clusters.

Dataset	N	d	n	K
Reuters Subset	1209	300	16	5
BBCnews Abstract	2225	300	16	5
BBCsport Abstract	737	300	16	5

Table 1, we use the pre-trained word-vector dataset GloVe (Pennington et al., 2014) to transform a list of words to a measure over $\mathbb{R}^{300}$. The weight of each word is the normalized frequency modified by the TF-IDF scheme. We use the GloVe 300d (word vectors $\in \mathbb{R}^{300}$) that was trained on 6 billion tokens and contains a 400,000 lower case vocabulary. Before we transform words into vectors, we lower the capital letters, remove all punctuations and stop words and lemmatize each document. Finally, we restrict the number of support points to n by recursively merging the closest words. Specifically, when the number of different words in μ^l is larger than n, we solve the following discrete optimization problem:

$$
\min_{i,j} p_i^l p_j^l \|x_i^l - x_j^l\|^2 / (p_i^l + p_j^l), \tag{33}
$$

and merge p_i^l, p_j^l as $\bar{p}^l = p_i^l + p_j^l$, $\bar{x} = (p_i^l x_i^l - p_j^l x_j^l)/\bar{p}^l$.

Parameter setting and initialization. In each iteration of PD2 clustering, we run the RBCD algorithm with the step size $\tau = 0.05$, the regularization parameter $\eta = 1$. We choose $k = 2$ for the "BBCsport Abstract" dataset and $k = 3$ for the "Reuters Subset" and the "BBCnews Abstract" datasets. The initial K barycenters are chosen randomly from documents with more than n different words and recursively merged so the number of support points remains n.

The Adjusted Mutual Information. To measure the performance of the clustering results, we use the Adjusted Mutual Information (AMI) (Vinh et al., 2010). Denote $P_V(i) = |V_i|/N$ as the probability of clus-

ter i in the partition V. The entropy $H(V)$ is defined as $H(V) = -\sum_{i=1}^{S_V} P_V(i) \log P_V(i)$, where S_V is the number of clusters in V. The mutual information between the two partitions V_1, V_2 is defined as $MI(V_1, V_2) = \sum_{i,j=1}^{S_{V_1}, S_{V_2}} P_{V_1, V_2}(i,j) \log \frac{P_{V_1, V_2}(i,j)}{P_{V_1}(i) P_{V_2}(j)}$, where $P_{V_1, V_2}(i,j) = |V_{1,i} \cup V_{2,j}|/N$. The AMI score between two partitions V_1, V_2 is computed by

$$AMI(V_1, V_2) = \frac{MI(V_1, V_2) - \mathbb{E}MI(V_1, V_2)}{(H(V_1) + H(V_2))/2 - \mathbb{E}MI(V_1, V_2)}.$$

The AMI score lies in the interval $[0, 1]$, and it remains unchanged when we permute the cluster labels. In our experiments, we present the AMI scores between the ground truth labels and the predicted labels.

Clustering results. We run D2 and PD2 on the two real datasets in Table 1. The final AMI score and the average number of iterations for different datasets are given in Tables 2 and Table 3 respectively. We apply k-means clustering on the raw TF-IDF vectors as a baseline. Each result is averaged over five runs with different initialization. We stop the D2 and PD2 algorithms when the labels for each cluster are stable. Comparing the AMI scores in Table 2, we see that the proposed RPRWB model improves the performance of text clustering. One possible reason is that for many real high dimensional datasets, a low dimensional subspace catches most of the information. Notice that the D2 clustering AMI scores reported here are smaller than those in (Ye et al., 2017a). This is because the clustering performance highly depends on the barycenter initialization, and we are reporting the average AMIs with different initialization while (Ye et al., 2017a) reported the best AMI they obtained. We further see that the average number of iterations of the PD2 algorithm is smaller. Moreover, we plot the AMI scores for the first ten iterations of the D2 and PD2 clustering algorithm in Figure 5. We see that the PD2 clustering algorithm gives better AMI scores than the D2 clustering algorithm, which shows the advantage of the proposed RPRWB model.

Table 2. AMI scores for clustering results.

Dataset	k-means	D2	PD2
Reuters Subset	0.4627	0.4200	**0.4713**
BBCnews Abstract	0.3877	0.6095	**0.6557**
BBCsport Abstract	0.4276	0.6510	**0.6892**

7. Conclusion

In this paper, we have proposed a novel WB model called the projection robust Wasserstein barycenter, which has the potential to mitigate the curse of dimensionality for the WB

Table 3. Average number of clustering iteration.

Dataset	D2	PD2
Reuters Subset	24.2	23.2
BBCnews Abstract	23.8	22.4
BBCsport Abstract	29.8	14.4

problem. To resolve the computational issue of the PRWB, we have proposed a relaxed PRWB model: RPRWB. We have proposed two algorithms, the RBCD algorithm and the RGA-IBP algorithm for solving the fixed-support RPRWB problem. We have analyzed the iteration complexity and complexity of arithmetic operations for both algorithms. Numerical results on synthetic datasets have demonstrated the robustness and the better sample complexity of the proposed RPRWB model comparing with the WB model. Moreover, we have incorporated the RPRWB model to the D2 clustering algorithm, and proposed the projection robust D2 clustering algorithm. Numerical results on real text datasets show that the PD2 clustering improves the performance of the D2 clustering. Future directions include deriving sample complexity for WB, PRWB and RPRWB.

Acknowledgements

References

Absil, P.-A., Mahony, R., and Sepulchre, R. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

Agueh, M. and Carlier, G. Barycenters in the Wasserstein space. *SIAM Journal on Mathematical Analysis*, 43(2): 904–924, 2011.

Aldaz, J. A monotonicity property of variances. *Statistics & Probability Letters*, 83(5):1416–1419, 2013.

Altschuler, J., Niles-Weed, J., and Rigollet, P. Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration. In *Advances in neural information processing systems*, pp. 1964–1974, 2017.

Altschuler, J. M. and Boix-Adsera, E. Wasserstein

barycenters are NP-hard to compute. *arXiv preprint arXiv:2101.01100*, 2021.

Benamou, J.-D., Carlier, G., Cuturi, M., Nenna, L., and Peyré, G. Iterative Bregman projections for regularized transportation problems. *SIAM Journal on Scientific Computing*, 37(2):A1111–A1138, 2015.

Bonneel, N., Rabin, J., Peyré, G., and Pfister, H. Sliced and radon Wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51(1):22–45, 2015.

Borgwardt, S. and Patterson, S. On the computational complexity of finding a sparse Wasserstein barycenter. *arXiv preprint arXiv:1910.07568*, 2019.

Boumal, N., Absil, P.-A., and Cartis, C. Global rates of convergence for nonconvex optimization on manifolds. *IMA Journal of Numerical Analysis*, 39(1):1–33, 2019.

Claici, S., Chien, E., and Solomon, J. Stochastic Wasserstein barycenters. In *International Conference on Machine Learning*, pp. 999–1008, 2018.

Cuturi, M. and Doucet, A. Fast computation of Wasserstein barycenters. 2014.

Dudley, R. M. The speed of mean Glivenko-Cantelli convergence. *The Annals of Mathematical Statistics*, 40(1): 40–50, 1969.

Ge, D., Wang, H., Xiong, Z., and Ye, Y. Interior-point methods strike back: Solving the Wasserstein barycenter problem. In *Advances in Neural Information Processing Systems*, pp. 6894–6905, 2019.

Greene, D. and Cunningham, P. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proceedings of the 23rd international conference on Machine learning*, pp. 377–384, 2006.

Ho, N., Nguyen, X., Yurochkin, M., Bui, H. H., Huynh, V., and Phung, D. Multilevel clustering via Wasserstein means. In *International Conference on Machine Learning*, pp. 1501–1509, 2017.

Huang, M., Ma, S., and Lai, L. A Riemannian block coordinate descent method for computing the projection robust Wasserstein distance. *arXiv preprint arXiv:2012.05199*, 2020.

Kroshnin, A., Tupitsa, N., Dvinskikh, D., Dvurechensky, P., Gasnikov, A., and Uribe, C. On the complexity of approximating Wasserstein barycenters. In *International conference on machine learning*, pp. 3530–3540. PMLR, 2019.

Lin, T., Fan, C., Ho, N., Cuturi, M., and Jordan, M. Projection robust Wasserstein distance and Riemannian optimization. In *NeurIPS*, volume 33, 2020a.

Lin, T., Ho, N., Chen, X., Cuturi, M., and Jordan, M. I. Fixed-support Wasserstein barycenters: Computational hardness and fast algorithm. *Advances in Neural Information Processing Systems*, 33, 2020b.

Niles-Weed, J. and Rigollet, P. Estimation of Wasserstein distances in the spiked transport model. *arXiv preprint arXiv:1909.07513*, 2019.

Paty, F.-P. and Cuturi, M. Subspace robust Wasserstein distances. In *International Conference on Machine Learning*, pp. 5072–5081, 2019.

Pennington, J., Socher, R., and Manning, C. D. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

Rabin, J., Peyré, G., Delon, J., and Bernot, M. Wasserstein barycenter and its application to texture mixing. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pp. 435–446. Springer, 2011.

Vinh, N. X., Epps, J., and Bailey, J. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *The Journal of Machine Learning Research*, 11:2837–2854, 2010.

Weed, J. and Bach, F. Sharp asymptotic and finite-sample rates of convergence of empirical measures in Wasserstein distance. *Bernoulli*, 25(4A):2620–2648, 2019.

Ye, J., Li, Y., Wu, Z., Wang, J. Z., Li, W., and Li, J. Determining gains acquired from word embedding quantitatively using discrete distribution clustering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1847–1856, 2017a.

Ye, J., Wu, P., Wang, J. Z., and Li, J. Fast discrete distribution clustering using Wasserstein barycenter with sparse support. *IEEE Transactions on Signal Processing*, 65(9): 2317–2332, 2017b.

A. Preliminaries on Riemannian Optimization

When considering optimization problems over the matrix manifold $\mathcal{M}$, the Riemannian Gradient Descent (Absil et al., 2009) algorithm is widely used. A core ingredient of the Riemannian Optimization is the retraction operation:

Definition 12 ((Absil et al., 2009)) *(Retraction) Denote the tangent space of $\mathcal{M}$ at U as $\mathrm{T}_U\mathcal{M}$. A retraction on $\mathcal{M}$ is a smooth mapping $\mathrm{Retr}(\cdot)$ from the tangent bundle $\mathrm{T}\mathcal{M}$ onto $\mathcal{M}$ satisfying the following two conditions:*

- *$\mathrm{Retr}_U(0) = U$, $\forall U \in \mathcal{M}$, where 0 denotes the zero element of $\mathrm{T}_U\mathcal{M}$;*

- *For any $U \in \mathcal{M}$, it holds that*

$$\lim_{\mathrm{T}_U\mathcal{M} \ni \xi \to 0} \|\mathrm{Retr}_U(\xi) - (U + \xi)\|_F / \|\xi\|_F = 0.$$

In each step, the Riemannian Gradient Descent updates as

$$U_{t+1} = \mathrm{Retr}(-\tau \mathrm{grad} f(U_t)), \tag{34}$$

where $\mathrm{grad} f(U_t)$ is the Riemannian Gradient of $f(U)$ at U_t defined as

$$\mathrm{grad} f(U) = \mathrm{Proj}_{\mathrm{T}_U\mathcal{M}} \nabla f(U). \tag{35}$$

For the Stiefel manifold $\mathrm{St}(d, k) = \{U \in \mathbb{R}^{d \times k} | U^\top U = I_k\}$, the retraction has the following property.

Proposition 13 ((Boumal et al., 2019)) *There exists constants $L_1, L_2 > 0$ such that for any $U \in \mathcal{M}$ and $\xi \in \mathrm{T}_U\mathcal{M}$, the following inequalities hold:*

$$\begin{aligned}
\|\mathrm{Retr}_U(\xi) - U\|_F &\leq L_1\|\xi\|_F, \\
\|\mathrm{Retr}_U(\xi) - (U + \xi)\|_F &\leq L_2\|\xi\|_F^2.
\end{aligned}$$

B. Proof of Proposition 1

Proof. Notice that the Grassmannian $\mathcal{G}_k$ is compact and the function $E \to \sum_{l=1}^m \omega^l \mathcal{W}^2(\mathrm{Proj}_E \mu^l, \mathrm{Proj}_E \nu)$ is semi-continuous. These two facts lead to the desired result. $\qquad\square$

C. Proof of Lemma 5

Proof. We derive the dual problem of (12). The Lagrangian function of (12) is:

$$
\begin{aligned}
L(\boldsymbol{\pi}, U; \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \sum_{l=1}^m \omega^l \left\{ \sum_{i,j=1}^n \pi_{i,j}^l \|U^\top(x_i^l - y_j)\|^2 - \eta H(\pi^l) \right\} + \sum_{l=1}^m \langle \alpha^l, \pi^l \mathbf{1} - p^l \rangle + \sum_{l=1}^m \langle \beta^l, (\pi^{l+1})^\top \mathbf{1} - (\pi^l)^\top \mathbf{1} \rangle \\
&= \sum_{l=1}^m \omega^l \left\{ \sum_{i,j=1}^n \pi_{i,j}^l \|U^\top(x_i^l - y_j)\|^2 - \eta H(\pi^l) \right\} + \sum_{l=1}^m \langle \alpha^l, \pi^l \mathbf{1} - p^l \rangle + \sum_{l=1}^m \langle \beta^{l-1} - \beta^l, (\pi^l)^\top \mathbf{1} \rangle
\end{aligned}
\tag{36}
$$

where $\boldsymbol{\alpha} = \{\alpha^l \in \mathbb{R}^n\}_{l \in [m]}, \boldsymbol{\beta} = \{\beta^l \in \mathbb{R}^n\}_{l \in [0,\ldots,m]}$ are the Lagrange multipliers with $\beta^0 = \beta^m = 0$. By changing the variables as $u^l = -\frac{\alpha^l}{\omega^l \eta}, v^l = -\frac{\beta^{l-1} - \beta^l}{\omega^l \eta}$, we can rewrite the Lagrangian function as:

$$
L(\boldsymbol{\pi}, U; \boldsymbol{u}, \boldsymbol{v}) = \eta \sum_{l=1}^m \omega^l \left\{ \sum_{i,j=1}^n \pi_{i,j}^l \left(\frac{\|U^\top(x_i^l - y_j)\|^2}{\eta} + (\log \pi_{i,j} - 1) - u_i^l - v_j^l \right) + \sum_i u_i^l p_i^l \right\}. \tag{37}
$$

Note that the change of the variables results in a further constraint $\sum_{l=1}^{m} \omega^l v^l = 0$. We denote $\boldsymbol{u} = \{u^l\}_{l \in [m]}, \boldsymbol{v} = \{v^l\}_{l \in [m]}$. The dual problem of (12) is

$$
\max_{\substack{\boldsymbol{u},\boldsymbol{v} \in \mathbb{R}^{m \times n}, \\ \sum_{l=1}^{m} \omega^l v^l = 0}} \quad \min_{\pi^l \in \mathbb{R}_+^{n \times n}} L(\boldsymbol{\pi}, U; \boldsymbol{u}, \boldsymbol{v}) =
$$

$$
\max_{\substack{\boldsymbol{u},\boldsymbol{v} \in \mathbb{R}^{m \times n}, \\ \sum_{l=1}^{m} \omega^l v^l = 0}} \eta \sum_{l=1}^{m} \omega^l \left\{ \min_{\pi^l \in \mathbb{R}_+^{n \times n}} \sum_{i,j=1}^{n} \pi_{i,j}^l \left(\frac{\|U^\top(x_i^l - y_j)\|^2}{\eta} + (\log \pi_{i,j} - 1) - u_i^l - v_j^l \right) + \langle u^l, p^l \rangle \right\}. \tag{38}
$$

Notice that the inner minimization problem over $\boldsymbol{\pi}$ in (38) has a closed form solution:

$$
\pi_{i,j}^l = \exp\left(-\frac{\|U^\top(x_i^l - y_j)\|^2}{\eta} + u_i^l + v_j^l \right), \ \forall l \in [m]. \tag{39}
$$

Plugging (39) into (38), the dual problem becomes

$$
\max_{\substack{\boldsymbol{u},\boldsymbol{v} \in \mathbb{R}^{m \times n}, \\ \sum_{l=1}^{m} \omega^l v^l = 0}} \eta \sum_{l=1}^{m} \omega^l \left\{ -\sum_{i,j=1}^{n} \exp\left(-\frac{\|U^\top(x_i^l - y_j)\|^2}{\eta} + u_i^l + v_j^l \right) + \langle u^l, p^l \rangle \right\}. \tag{40}
$$

This completes the proof. $\qquad\square$

D. The Iterative Bregman Projection Algorithm

For the completeness of Algorithm 1, we include the IBP subroutine in this section, and present it as Algorithm 4. When fixing U in the dual formulation (15), the Iterative Bregman Projection algorithm updates $\boldsymbol{u}, \boldsymbol{v}$ by (21) - (22) in an alternative scheme. In Algorithm 4, we define

$$
\begin{aligned}
\pi(u_t^l, v_t^l, U)_{i,j} &= \exp\left(-\frac{\|U^\top(x_i^l - y_j)\|^2}{\eta} + u_{t,i}^l + v_{t,j}^l \right), \\
q_t^l &= (\pi(u_{t+1}^l, v_{t+1}^l, U))^\top \mathbf{1}, \\
q_{t+1} &= \exp\left(\sum_{l=1}^{m} \omega^l \log q_t^l \right).
\end{aligned} \tag{41}
$$

The stopping criteria $\sum_{l=1}^{m} \omega^l \|q_t^l - \bar{q}_t\|_1 \leq \frac{\eta \epsilon^2}{200 \bar{c}^3}$ guarantees the following inequality holds:

$$
\sum_{l=1}^{m} \omega^l \|\pi_{t+1}^l - \pi_\eta^*(U_t)^l\|_1 \leq= \frac{\epsilon}{10\bar{c}}, \tag{42}
$$

which will be proved in Section F.

E. Proof of Theorem 8 and Corollary 9

The proof of Theorem 8 consists of two parts. We first prove that when the Algorithm 2 stops, the output $(\hat{\boldsymbol{\pi}}, \hat{U})$ is an ϵ-stationary point defined in Definition 3. Secondly, we give the iteration complexity of terminating the RBCD algorithm. Finally, we analyze the per iteration complexity of the Algorithm 2. Below, we list some useful lemmas and theorems proved in the literature.

Lemma 14 shows the bound of the difference between the input and the output of the Rounding procedure (Algorithm 3).

Lemma 14 *(Altschuler et al., 2017)[Lemma 7] Let $p, q \in \Delta^n$, $\pi \in \mathbb{R}_+^{n \times n}$, and $\hat{\pi}$ be the output of $Round(\pi, p, q)$. The following inequality holds:*
$$
\|\hat{\pi} - \pi\|_1 \leq 2(\|\pi \mathbf{1} - p\|_1 + \|(\pi)^\top \mathbf{1} - q\|_1).
$$

Algorithm 4 The Iterative Bregman Projection Algorithm — IBPsolver($\boldsymbol{\mu}, Y, U, \eta, \epsilon$)

1: **Input:** $\{\mu^l = (X^l, p^l)\}_{l \in [m]}$, $\{Y\}$, U, η, accuracy tolerance $\epsilon > 0$.
2: **Initialization:** $\boldsymbol{u}_0, \boldsymbol{v}_0 \in \mathbb{R}^{m \times n}$.
3: **for** $t = 0, 1, 2, \ldots,$ **do**
4:    Compute $v^l_{t+1} = v^l_t + \log(\frac{q_{t+1}}{q^l_t}), \quad \forall l \in [m]$;
5:    Compute $u^l_{t+1} = u^l_t + \log(\frac{p^l}{\pi(u^l_t, v^l_{t+1}, U)\mathbf{1}}), \quad \forall l \in [m]$;
6:    **if** $\sum_{l=1}^m \omega^l \|q^l_t - \bar{q}_t\|_1 \leq \frac{\eta \epsilon^2}{200\bar{c}^3}$ **then**
7:       break;
8:    **end if**
9: **end for**
10: **Output:** $q = \bar{q}_t = \sum_{l=1}^m \omega^l q^l_t$, and $\hat{\pi}^l = Round(\pi^l(\hat{u}^l, \hat{v}^l, \hat{U}), p^l, q), \forall l \in [m]$.

Theorem 15 bounds the difference between the arithmetic mean and the geometric mean by the variance of the random variable.

Theorem 15 *(Aldaz, 2013)[Theorem 2.4] For $n \geq 2$ and $l \in [m]$, let $X = (x_1, \ldots x_n)$ be such that $x_l \geq 0$, and let $\omega = (\omega^1, \ldots, \omega^l)$ satisfy $\omega^l > 0$ and $\sum_{l=1}^m \omega^l = 1$. Then for all $s \in [1, \infty)$, we have*

$$\mathbb{E}_\omega X - \Pi_\omega X \leq \frac{1}{\min_l \omega^l} Var_\omega(X^{s/2})^{1/s},$$

where $\mathbb{E}_\omega X = \sum_{l=1}^m \omega^l x_l$ is the arithmetic mean, $\Pi_\omega X = \Pi_{l=1}^m x_l^{\omega^l}$ is the geometric mean and $Var_\omega(X) = \sum_{l=1}^m \omega^l(x_l - \mathbb{E}_\omega X)^2$ is the variance.

The following lemma shows the relation between the primal and dual objective function.

Lemma 16 *Denote $\boldsymbol{\pi} = \{\pi^l\}_{l=1}^m$. By (39), each π^l can be written as*

$$\pi^l_{i,j} = \pi(u^l, v^l, U)_{i,j} = \exp\left(-\frac{\|U^\top(x^l_i - y_j)\|^2}{\eta} + u^l_i + v^l_j\right),$$

where $u^l, v^l, l \in [m]$ are the corresponding dual variables. The following equality holds after $\boldsymbol{u}$ is updated by (21):

$$f_\eta(\boldsymbol{\pi}, U) = -\eta g(\boldsymbol{u}, \boldsymbol{v}, U) + \eta \sum_{l=1}^m \omega^l \langle v^l, q^l \rangle, \tag{43}$$

where $q^l = (\pi^l)^\top \mathbf{1}$, f_η is defined in (12), and g is defined in (17).

Proof. Fix U and denote $M^l \in \mathbb{R}^{n \times n}, l \in [m]$ with $M^l_{i,j} = \|U^\top(x^l_i - y_j)\|^2$. Plugging (39) into $H(\pi^l)$ leads to

$$f_\eta(\boldsymbol{\pi}, U) = \sum_{l=1}^m \omega^l(\langle M^l, \pi^l \rangle - \eta H(\pi^l)) = \eta \sum_{l=1}^m \omega^l(\sum_{i,j} \pi^l_{i,j} u^l_i + \pi^l_{i,j} v^l_j - \pi^l_{i,j}).$$

The update rule (21) yields $\pi^l \mathbf{1} = p^l$. Therefore, we have

$$
\begin{aligned}
f_\eta(\boldsymbol{\pi}, U) &= -\eta \sum_{l=1}^m \omega^l \sum_{i,j} (\pi^l_{i,j} - \pi^l_{i,j} u^l_i - \pi^l_{i,j} v^l_j) \\
&= -\eta \sum_{l=1}^m \omega^l \left[\sum_{i,j}(\pi^l_{i,j}) - \langle \pi^l \mathbf{1}, u^l \rangle - \langle (\pi^l)^\top \mathbf{1}, v^l \rangle \right] \\
&= -\eta \sum_{l=1}^m \omega^l \left[\sum_{i,j}(\pi^l_{i,j}) - \langle p^l, u^l \rangle - \langle v^l, q^l \rangle \right] \\
&= -\eta g(\boldsymbol{u}, \boldsymbol{v}, U) + \eta \sum_{l=1}^m \omega^l \langle v^l, q^l \rangle,
\end{aligned}
\tag{44}
$$

which completes the proof. $\qquad\square$

The next lemma shows that when Algorithm 2 terminates, it returns an ϵ-stationary point of the problem (8) as defined in Definition 3.

Lemma 17 *Assume Algorithm 2 terminates at iteration T. Then $(\hat{\boldsymbol{\pi}}, \hat{U})$ returned by Algorithm 2, i.e., $\hat{\pi}^l = Round(\pi(u_{T+1}^l, v_T^l, U_T), p^l, q)$ and $\hat{U} = U_T$, is an ϵ-stationary point of the problem (8) as defined in Definition 3.*

Proof. When Algorithm 2 terminates at the T-th iteration, we have (note that $q = \bar{q}_T$):

$$\sum_{l=1}^{m} \omega^l \|q_T^l - \bar{q}_T\|_1 \leq \underline{\omega}^{3/2} \epsilon / (12\bar{c}), \tag{45}$$

and

$$\eta \|\text{grad}_U g(\boldsymbol{u}_{T+1}, \boldsymbol{v}_{T+1}, U_T)\|_F \leq \frac{\epsilon}{3}. \tag{46}$$

Fix U_T and denote $\bar{\pi}^l = \pi^l(u_{T+1}^l, v_T^l, U_T)$. By Lemma 16, we have

$$f_\eta(\bar{\boldsymbol{\pi}}, U_T) = -\eta g(\boldsymbol{u}_{T+1}, \boldsymbol{v}_T, U_T) + \eta \sum_{l=1}^{m} \omega^l \langle v_T^l, q_T^l \rangle. \tag{47}$$

Let $\boldsymbol{\pi}_\eta^*(U_T) = \{(\pi_\eta^*(U_T))^l\}$ be the solution of (12) when fixing U as U_T. Denote $\boldsymbol{u}_\eta^*, \boldsymbol{v}_\eta^*$ as the corresponding optimal solution to the dual problem. We have the relation:

$$(\pi_\eta^*(U))_{i,j}^l = \exp\left(-\frac{\|U^\top(x_i^l - y_j)\|^2}{\eta} + (u_\eta^*)_i^l + (v_\eta^*)_j^l\right). \tag{48}$$

The optimal regularized Barycenter, denoted as q_η^*, can be obtained by $q_\eta^* = [(\pi_\eta^l)^*]^\top \mathbf{1} = [(\pi_\eta^{l+1})^*]^\top \mathbf{1}, \forall l \in \{1, ..., m-1\}$. Using Lemma 16 with variables $\boldsymbol{\pi}_\eta^*(U_T), \boldsymbol{u}_\eta^*, \boldsymbol{v}_\eta^*$, we have

$$\begin{aligned} f_\eta(\boldsymbol{\pi}_\eta^*(U_T), U_T) &= -\eta g(\boldsymbol{u}_\eta^*, \boldsymbol{v}_\eta^*, U_T) + \eta \sum_{l=1}^{m} \omega^l \langle (v_\eta^*)^l, q_\eta^* \rangle \\ &= -\eta g(\boldsymbol{u}_\eta^*, \boldsymbol{v}_\eta^*, U_T), \end{aligned} \tag{49}$$

where the last equality uses the fact that $\sum_{l=1}^{m} \omega^l (v_\eta^*)^l = 0$.

Let $b_T^l = \frac{\min_i [v_T^l]_i + \max_i [v_T^l]_i}{2}$ be a constant. Combining (47) and (49) yields

$$\begin{aligned} f_\eta(\bar{\boldsymbol{\pi}}, U_T) - f_\eta(\boldsymbol{\pi}_\eta^*(U_T), U_T) &= -\eta(g(\boldsymbol{u}_{T+1}, \boldsymbol{v}_T, U_T) - g(\boldsymbol{u}_\eta^*, \boldsymbol{v}_\eta^*, U_T)) + \eta \sum_{l=1}^{m} \omega^l \langle v_T^l, q_T^l \rangle \\ &\leq \eta \sum_{l=1}^{m} \omega^l \langle v_T^l, q_T^l \rangle \\ &= \eta \sum_{l=1}^{m} \omega^l \langle v_T^l, q_T^l - \bar{q}_T \rangle \\ &= \eta \sum_{l=1}^{m} \omega^l \langle v_T^l - b_T^l \mathbf{1}, q_T^l - \bar{q}_T \rangle \\ &\leq \eta \sum_{l=1}^{m} \omega^l \|v_T^l - b_T^l \mathbf{1}\|_\infty \|q_T^l - \bar{q}_T\|_1, \end{aligned} \tag{50}$$

where the second equality follows from the constraint $\sum_{l=1}^{m} \omega^l v^l = 0$ in (19), the third equality is due to $\langle q_T^l, \mathbf{1} \rangle = 1, \langle \bar{q}_T, \mathbf{1} \rangle = 1$, which follows from the optimality condition of (18), and the last inequality is by Hölder's inequality. By

(Kroshnin et al., 2019)[Lemma 4], we can bound $\|v_T^l - b_T^l \mathbf{1}\|_\infty$ by $\frac{\max_l \|M^l\|_\infty}{\eta} \leq \frac{\max_l \|C^l\|_\infty}{\eta}$. Therefore, it holds that

$$
f_\eta(\bar{\boldsymbol{\pi}}, U_T) - f_\eta(\boldsymbol{\pi}_\eta^*(U_T), U_T) = \sum_{l=1}^m \omega^l (\langle M^l, \bar{\pi}^l \rangle - \eta H(\bar{\pi}^l)) - \sum_{l=1}^m \omega^l (\langle M^l, (\pi_\eta^*(U_T))^l \rangle - \eta H((\pi_\eta^*(U_T))^l))
$$

$$
\leq \bar{c} \sum_{l=1}^m \omega^l \|q_T^l - \bar{q}_T\|_1 \leq \underline{\omega}^{3/2} \epsilon / 12 \leq \epsilon/12,
$$

where the second inequality is from (45). Further denote $\boldsymbol{\pi}^*(U_T)) = \{(\pi^*(U_T))^l\}$ as the solution of the unregularized inner minimization problem of (8) when fixing U as U_T. The above inequality implies

$$
\sum_{l=1}^m \omega^l \langle M^l, \bar{\pi}^l \rangle \leq \sum_{l=1}^m \omega^l \left(\langle M^l, (\pi_\eta^*(U_T))^l \rangle - \eta H((\pi_\eta^*(U_T))^l) + \eta H(\bar{\pi}^l) \right) + \frac{\epsilon}{12}
$$

$$
\leq \sum_{l=1}^m \omega^l \left(\langle M^l, (\pi^*(U_T))^l \rangle - \eta H((\pi^*(U_T))^l) + \eta H(\bar{\pi}^l) \right) + \frac{\epsilon}{12} \tag{51}
$$

$$
\leq \sum_{l=1}^m \omega^l \langle M^l, (\pi^*(U_T))^l \rangle + \frac{7\epsilon}{12},
$$

where in the last inequality, we use the fact $0 \leq H(\pi^l) \leq 2\log n + 1$, and $\eta = \frac{\epsilon}{4\log n + 2}$. By Lemma 14, since $\hat{\pi}^l = Round(\bar{\pi}^l, p^l, q)$, Algorithm 3 outputs $\hat{\pi}^l$ satisfying

$$
\|\hat{\pi}^l - \bar{\pi}^l\|_1 \leq 2(\|\bar{\pi}^l \mathbf{1} - p^l\|_1 + \|(\bar{\pi}^l)^\top \mathbf{1} - q\|_1) = 2\|q_T^l - q\|_1, \quad \forall l \in [m], \tag{52}
$$

where we note $q = \bar{q}_T$, and we have used the fact $\bar{\pi}^l \mathbf{1} - p^l = 0$ that comes from the optimality condition of (18). Equation (52) further implies

$$
\sum_{l=1}^m \omega^l \|\hat{\pi}^l - \bar{\pi}^l\|_1 \leq 2 \sum_{l=1}^m \omega^l \|q_T^l - q\|_1. \tag{53}
$$

Combining (53) with (51) and applying Hölder's inequality yields

$$
\sum_{l=1}^m \omega^l \langle M^l, \hat{\pi}^l \rangle \leq \sum_{l=1}^m \omega^l \left(\langle M^l, \bar{\pi}^l \rangle + \|M^l\|_\infty \|\hat{\pi}^l - \bar{\pi}^l\|_1 \right)
$$

$$
\leq \sum_{l=1}^m \omega^l \langle M^l, \bar{\pi}^l \rangle + (\max_l \|M^l\|_\infty) \sum_{l=1}^m \omega^l \|\hat{\pi}^l - \bar{\pi}^l\|_1
$$

$$
\leq \sum_{l=1}^m \omega^l \langle M^l, \bar{\pi}^l \rangle + 2(\max_l \|C^l\|_\infty) \sum_{l=1}^m \omega^l \|q_t^l - q\|_1 \tag{54}
$$

$$
\leq \sum_{l=1}^m \omega^l \langle M^l, (\pi^*(U_T))^l \rangle + \frac{7\epsilon}{12} + \frac{(\min_l \omega^l)^{3/2} \epsilon}{6}
$$

$$
\leq \sum_{l=1}^m \omega^l \langle M^l, (\pi^*(U_T))^l \rangle + \epsilon,
$$

where the fourth inequality follows from (45), and the last inequality holds since $(\min_l \omega^l)^{3/2} < 1$. Therefore, we have proved that $(\hat{\pi}, \hat{U})$ satisfies (10) in Definition 3.

The rest of the proof is to prove that $(\hat{\pi}, \hat{U})$ satisfies (9) in Definition 3. That is, we need to bound $\|\text{grad}_U f(\hat{\pi}, U_T)\|_F$. For simplicity of notation, we further denote $\tilde{\boldsymbol{\pi}} = \boldsymbol{\pi}(\boldsymbol{u}_{T+1}, \boldsymbol{v}_{T+1}, U_T)$. Since

$$
\nabla_U f(\tilde{\boldsymbol{\pi}}, U_T) = 2V_{\tilde{\boldsymbol{\pi}}} U_T = -\eta \nabla_U g(\boldsymbol{u}_{T+1}, \boldsymbol{v}_{T+1}, U_T),
$$

which combining with (46) leads to

$$
\|\text{grad}_U f(\tilde{\boldsymbol{\pi}}, U_T)\|_F = \|\text{Proj}_{T_{U_T}\mathcal{M}}(\nabla_U f(\tilde{\boldsymbol{\pi}}, U_T))\|_F \leq \frac{\epsilon}{3}.
$$

By triangle inequality, we have

$$
\begin{aligned}
\|\mathrm{grad}_U f(\hat{\boldsymbol{\pi}}, U_T)\|_F &= \|\mathrm{Proj}_{\mathrm{T}_{U_T}\mathcal{M}}(2V_{\hat{\boldsymbol{\pi}}}U_T)\|_F \\
&= \|\mathrm{Proj}_{\mathrm{T}_{U_T}\mathcal{M}}(2(V_{\hat{\boldsymbol{\pi}}} - V_{\bar{\boldsymbol{\pi}}} + V_{\bar{\boldsymbol{\pi}}} - V_{\tilde{\boldsymbol{\pi}}} + V_{\tilde{\boldsymbol{\pi}}})U_T)\|_F \\
&\leq 2\|(V_{\hat{\boldsymbol{\pi}}} - V_{\bar{\boldsymbol{\pi}}})U_T\|_F + 2\|(V_{\bar{\boldsymbol{\pi}}} - V_{\tilde{\boldsymbol{\pi}}})U_T\|_F + \|\mathrm{Proj}_{\mathrm{T}_{U_T}\mathcal{M}}(2V_{\tilde{\boldsymbol{\pi}}}U_T)\|_F \\
&\leq 2\|V_{\hat{\boldsymbol{\pi}}} - V_{\bar{\boldsymbol{\pi}}}\|_F + 2\|V_{\bar{\boldsymbol{\pi}}} - V_{\tilde{\boldsymbol{\pi}}}\|_F + \|\mathrm{grad}_U f(\tilde{\boldsymbol{\pi}}, U_T)\|_F \\
&\leq 2\|V_{\hat{\boldsymbol{\pi}}} - V_{\bar{\boldsymbol{\pi}}}\|_F + 2\|V_{\bar{\boldsymbol{\pi}}} - V_{\tilde{\boldsymbol{\pi}}}\|_F + \frac{\epsilon}{3}.
\end{aligned}
\tag{55}
$$

In the following, we will bound $\|V_{\hat{\boldsymbol{\pi}}} - V_{\bar{\boldsymbol{\pi}}}\|_F$ and $\|V_{\bar{\boldsymbol{\pi}}} - V_{\tilde{\boldsymbol{\pi}}}\|_F$. Equations (53) and (45) indicate that

$$
2\|V_{\hat{\boldsymbol{\pi}}} - V_{\bar{\boldsymbol{\pi}}}\|_F \leq 2\bar{c}\sum_{l=1}^{m}\omega^l\|\hat{\pi}^l - \bar{\pi}^l\|_1 \leq 4\bar{c}\sum_{l=1}^{m}\omega^l\|q_T^l - q\|_1 \leq 4\bar{c}\cdot\frac{\omega^{3/2}\epsilon}{12\bar{c}} \leq \frac{\epsilon}{3}.
\tag{56}
$$

We now bound $\|V_{\bar{\boldsymbol{\pi}}} - V_{\tilde{\boldsymbol{\pi}}}\|_F$. Note that

$$
\begin{aligned}
&\|\bar{\pi}^l - \tilde{\pi}^l\|_1 \\
&= \sum_{ij}\left|\exp\left(-\frac{1}{\eta}\|(U_T)^\top(x_i^l - y_j)\|^2 + [u_{T+1}^l]_i + [v_T^l]_j\right) - \exp\left(-\frac{1}{\eta}\|(U_T)^\top(x_i^l - y_j)\|^2 + [u_{T+1}^l]_i + [v_{T+1}^l]_j\right)\right| \\
&\leq \sum_{ij}\exp\left(-\frac{1}{\eta}\|(U_T)^\top(x_i^l - y_j)\|^2 + [u_{T+1}^l]_i + [v_T^l]_j\right)|1 - \exp([v_{T+1}^l]_j - [v_T^l]_j)| \\
&= \sum_{j}\sum_{i}\bar{\pi}_{i,j}^l|1 - \frac{[q_{T+1}]_j}{[q_T^l]_j}| \\
&= \|q_{T+1} - q_T^l\|_1,
\end{aligned}
\tag{57}
$$

where the second equality is from (22). This leads to

$$
2\|V_{\bar{\boldsymbol{\pi}}} - V_{\tilde{\boldsymbol{\pi}}}\|_F \leq 2\bar{c}\sum_{l=1}^{m}\omega^l\|\bar{\pi}^l - \tilde{\pi}^l\|_1 \leq 2\bar{c}\sum_{l=1}^{m}\omega^l\|q_T^l - q_{T+1}\|_1 \leq 2\bar{c}\left(\|\bar{q}_T - q_{T+1}\|_1 + \sum_{l=1}^{m}\omega^l\|q_T^l - \bar{q}_T\|_1\right).
\tag{58}
$$

To complete the proof, it remains to bound $\|\bar{q}_T - q_{T+1}\|_1$. Notice that $\bar{q}_T = \sum_{l=1}^{m}\omega^l q_T^l$ is the arithmetic mean and $q_{T+1} = \exp(\sum_{l=1}^{m}\omega^l\log q_T^l)$ is the geometric mean. Setting $s = 2$ in Theorem 15, we have

$$
\begin{aligned}
\|\bar{q}_T - q_{T+1}\|_1 &= \sum_{r=1}^{n}(\bar{q}_{T,r} - q_{T+1,r}) \leq \sum_{r=1}^{n}\frac{1}{\underline{\omega}}Var_\omega(q_{T,r}^l)^{1/2} = \frac{1}{\underline{\omega}}\sum_{r=1}^{n}\left[\sum_{l=1}^{m}\omega^l(q_{T,r}^l - \bar{q}_{T,r})^2\right]^{1/2} \\
&\leq \frac{1}{\underline{\omega}}\sum_{r=1}^{n}\left[\frac{1}{\underline{\omega}^l}\sum_{l=1}^{m}[\omega^l(q_{T,r}^l - \bar{q}_{T,r})]^2\right]^{1/2} \leq \frac{1}{\underline{\omega}^{3/2}}\sum_{r=1}^{n}\left[\left(\sum_{l=1}^{m}\omega^l\cdot|q_{T,r}^l - \bar{q}_{T,r}|\right)^2\right]^{1/2} \\
&= \frac{1}{\underline{\omega}^{3/2}}\sum_{r=1}^{n}\sum_{l=1}^{m}\omega^l\cdot|q_{T,r}^l - \bar{q}_{T,r}| = \frac{1}{\underline{\omega}^{3/2}}\sum_{l=1}^{m}\omega^l\|q_T^l - \bar{q}_T\|_1.
\end{aligned}
\tag{59}
$$

Combining (58) and (59) gives

$$
2\|V_{\bar{\boldsymbol{\pi}}} - V_{\tilde{\boldsymbol{\pi}}}\|_F \leq 2\bar{c}\left(\frac{1}{\underline{\omega}^{3/2}} + 1\right)\sum_{l=1}^{m}\omega^l\|q_T^l - \bar{q}_T\|_1 \leq \left(\frac{1}{\underline{\omega}^{3/2}} + 1\right)\frac{\omega^{3/2}\epsilon}{6} \leq \frac{\epsilon}{3}.
\tag{60}
$$

By (55), (56), and (60), we have

$$
\|\mathrm{grad}_U f(\hat{\pi}, \hat{U})\|_F \leq \epsilon.
\tag{61}
$$

Combining (54) and (61) completes the proof. $\qquad\square$

Now we analyze the iteration complexity of Algorithm 2.

We first present several technical lemmas. The first lemma shows that function g is lower bounded.

Lemma 18 *Denote* $(\boldsymbol{u}^*, \boldsymbol{v}^*, U^*)$ *as the global minimum of g defined in* (17). *The following inequality holds:*

$$g^* := g(\boldsymbol{u}^*, \boldsymbol{v}^*, U^*) \geq 1 - \bar{c}/\eta. \tag{62}$$

Proof. Notice that at the global minimum, we have for each π^l

$$\|\pi^l((u^l)^*, (v^l)^*, U^*)\|_1 = \sum_{i,j} \exp\left(-\frac{1}{\eta}\|(U^*)^\top(x_i^l - y_j)\|^2 + (u_i^l)^* + (v_j^l)^*\right) = 1, \tag{63}$$

which implies that

$$g(\boldsymbol{u}^*, \boldsymbol{v}^*, U^*) = \sum_{l=1}^m \omega^l \left\{ \sum_{i,j} [\pi^l((u^l)^*, (v^l)^*, U^*)]_{ij} - \langle (u^l)^*, p^l \rangle \right\} = 1 - \sum_{l=1}^m \omega^l \langle (u^l)^*, p^l \rangle. \tag{64}$$

Notice that $\|C^l\|_\infty \geq \|(U^*)^\top(x_i^l - y_j)\|^2$ for any i, j, together with (63) we have

$$\exp\left(-\frac{1}{\eta}\|C^l\|_\infty + (u^l)_i^* + (v^l)_j^*\right) \leq \exp\left(-\frac{1}{\eta}\|(U^*)^\top(x_i^l - y_j)\|^2 + (u^l)_i^* + (v^l)_j^*\right) = 1, \forall i, j,$$

which further implies

$$(u^l)_i^* + (v^l)_j^* \leq \frac{1}{\eta}\|C^l\|_\infty, \forall i, j. \tag{65}$$

Notice that we have $\sum_{l=1}^m \omega^l(v^l)^* = 0$, $p^l \in \Sigma^n, \forall l \in [m]$ and $q^* = (\pi^{l^*})^\top \mathbf{1} \in \Sigma^n$, (65) indicates that

$$\sum_{l=1}^m \omega^l\langle (u^l)^*, p^l \rangle = \sum_{l=1}^m \omega^l\langle (u^l)^*, p^l \rangle + \sum_{l=1}^m \omega^l\langle (v^l)^*, q^* \rangle \leq \frac{1}{\eta}\sum_{l=1}^m \omega^l\|C^l\|_\infty \leq \frac{(\max_l \|C^l\|_\infty)}{\eta},$$

which, combining with (64), yields the desired result. $\square$

The next two lemmas show that $g(\boldsymbol{u}, \boldsymbol{v}, U)$ is monotonically decreasing in updates (21) and (22).

Lemma 19 (Decrease of g in u) *Let $\{(u_t, v_t, U_t)\}$ be the sequence generated by Algorithm 2. For any $t \geq 0$, the following inequality holds*

$$g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_t, U_t) - g(\boldsymbol{u}_t, \boldsymbol{v}_t, U_t) \leq 0. \tag{66}$$

Proof. We have

$$g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_t, U_t) - g(\boldsymbol{u}_t, \boldsymbol{v}_t, U_t)$$

$$= \sum_{l=1}^m \omega^l \sum_i \left\{ \sum_j \left([\pi^l(u_{t+1}^l, v_t^l, U_t)]_{ij} - [\pi(u_t^l, v_t^l, U_t)]_{ij}\right) - p_i^l \log\left(\frac{p_i^l}{[\pi^l(u_t^l, v_t^l, U_t)\mathbf{1}]_i}\right) \right\}$$

$$= \sum_{l=1}^m \omega^l \sum_i \left\{ \sum_j [\pi(u_t^l, v_t^l, U_t)]_{ij}\left(\frac{p_i^l}{[\pi^l(u_t^l, v_t^l, U_t)\mathbf{1}]_i} - 1\right) - p_i^l \log\left(\frac{p_i^l}{[\pi^l(u_t^l, v_t^l, U_t)\mathbf{1}]_i}\right) \right\}$$

$$= \sum_{l=1}^m \omega^l \sum_i \left\{ (p_i^l - [\pi^l(u_t^l, v_t^l, U_t)\mathbf{1}]_i) + p_i^l \log\left(\frac{[\pi^l(u_t^l, v_t^l, U_t)\mathbf{1}]_i}{p_i^l}\right) \right\}$$

$$= \sum_{l=1}^m \omega^l \sum_i \left\{ (p_i^l - [\pi^l(u_t^l, v_t^l, U_t)\mathbf{1}]_i) + p_i^l \log\left(1 + \frac{[\pi^l(u_t^l, v_t^l, U_t)\mathbf{1}]_i - p_i^l}{p_i^l}\right) \right\} \leq 0,$$

where in the last inequality, we use $\log(1 + x) \leq x$. $\square$

Lemma 20 (Decrease of g in v) *Let $\{(u_t, v_t, U_t)\}$ be the sequence generated by Algorithm 2. For any $t \geq 0$, the following inequality holds*

$$g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}, U_t) - g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_t, U_t) \; \leq \; -\frac{1}{11}\left(\sum_{l=1}^{m}\omega^l\|q_t^l - \bar{q}_t\|_1\right)^2, \tag{67}$$

where $q_t^l = \pi^l(u_{t+1}^l, v_t^l, U_t)\mathbf{1}$ and $\bar{q}_t = \sum_{l=1}^{m}\omega^l q_t^l$.

Proof. Notice that we have

$$g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}, U_t) - g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_t, U_t)$$

$$= \sum_{l=1}^{m}\omega^l\sum_{i,j}\left\{\exp\left(-\frac{\|U_t^\top(x_i^l - y_j)\|^2}{\eta} + [u_{t+1}^l]_i + [v_{t+1}^l]_j\right) - \exp\left(-\frac{\|U_t^\top(x_i^l - y_j)\|^2}{\eta} + [u_{t+1}^l]_i + [v_t^l]_j\right)\right\}$$

$$= \sum_{l=1}^{m}\omega^l\sum_{i,j}\exp\left(-\frac{\|U_t^\top(x_i^l - y_j)\|^2}{\eta} + [u_{t+1}^l]_i + [v_t^l]_j\right)\left(\exp([v_{t+1}^l]_j - [v_t^l]_j) - 1\right)$$

$$= \sum_{l=1}^{m}\omega^l\sum_j\left(\sum_i\exp\left(-\frac{\|U_t^\top(x_i^l - y_j)\|^2}{\eta} + [u_{t+1}^l]_i + (v_t^l)_j\right)\right)\left(\frac{[q_{t+1}]_j}{[q_t^l]_j} - 1\right)$$

$$= \sum_{l=1}^{m}\omega^l\sum_j[q_t^l]_j\left(\frac{[q_{t+1}]_j}{[q_t^l]_j} - 1\right)$$

$$= \sum_{l=1}^{m}\omega^l\langle q_{t+1} - q_t^l, \mathbf{1}\rangle$$

$$= \langle q_{t+1} - \bar{q}_t, \mathbf{1}\rangle$$

$$\leq -\frac{1}{11}\sum_{l=1}^{m}\omega^l\|q_t^l - \bar{q}_t\|_1^2, \tag{68}$$

where the last inequality follows (Kroshnin et al., 2019)[Lemma 6]. We include its proof here for completeness. Denote $x^- = \max\{-x, 0\}$. By the definition of q_{t+1}, we have

$$\langle q_{t+1} - \bar{q}_t, \mathbf{1}\rangle = \left\langle \exp\left(\sum_{l=1}^{m}\omega^l\log q_t^l\right) - \bar{q}_t, \mathbf{1}\right\rangle \tag{69}$$

$$\leq -\frac{4}{11}\sum_{j=1}^{n}\frac{1}{[\bar{q}_t]_j}\sum_{l=1}^{m}\omega^l\left([q_t^l - \bar{q}_t]_j^-\right)^2$$

$$= -\frac{4}{11}\sum_{l=1}^{m}\omega^l\sum_{j=1}^{n}\frac{([q_t^l - \bar{q}_t]_j^-)^2}{[\bar{q}_t]_j}$$

$$\leq -\frac{4}{11}\sum_{l=1}^{m}\omega^l\frac{(\sum_{j=1}^{n}[q_t^l - \bar{q}_t]_j^-)^2}{\sum_{j=1}^{n}[\bar{q}_t]_j}$$

$$= -\frac{1}{11}\sum_{l=1}^{m}\omega^l\|q_t^l - \bar{q}_t\|_1^2.$$

Where the first inequality in (69) uses the fact: if $x \in \mathbb{R}_+^m$, $\bar{x} := \sum_{l=1}^{m}\omega^l x^l$, then (see the proof later):

$$\bar{x} - \prod_{l=1}^{m}(x^l)^{\omega^l} \geq \frac{4}{11}\sum_{l=1}^{m}\omega^l\frac{[(x_l - \bar{x})^-]^2}{\bar{x}}. \tag{70}$$

The second inequality in (69) uses the Cauchy-Schwarz inequality. The last equality is based on the fact that $\sum_{j=1}^{n}[q_t^l -$

$\bar{q}_t]_j^- = -\frac{1}{2}\|q_t^l - \bar{q}_t\|_1$, since $\langle \bar{q}_t, \mathbf{1} \rangle = \langle q_t^l, \mathbf{1} \rangle = 1$. Let $D^l = x^l - \bar{x}$, equation (70) can be proved as follows:

$$\bar{x} - \prod_{l=1}^m (x^l)^{\omega^l} = \bar{x} - \exp\left\{\sum_{l=1}^m \omega^l \ln(\bar{x} + D^l)\right\} = \bar{x}\left(1 - \exp\left\{\sum_{l=1}^m \omega^l \ln\left(1 + \frac{D^l}{\bar{x}}\right)\right\}\right), \tag{71}$$

$$\sum_{l=1}^m \omega^l \ln\left(1 + \frac{D^l}{\bar{x}}\right) \leq \sum_{l=1}^m \omega^l \left(\frac{D^l}{\bar{x}} - \frac{([D^l]^-)^2}{2\bar{x}^2}\right) = -\sum_{l=1}^m \omega^l \frac{([D^l]^-)^2}{2\bar{x}^2}. \tag{72}$$

Notice that $[D^l]^- = \max\{\bar{x} - x_l, 0\} \leq \bar{x}$, thus $\sum_{l=1}^m \omega^l \frac{([D^l]^-)^2}{\bar{x}^2} \leq 1$ and

$$\exp\left\{-\frac{1}{2}\sum_{l=1}^m \omega^l \frac{([D^l]^-)^2}{\bar{x}^2}\right\} \leq 1 - \left(1 - e^{-1/2}\right)\sum_{l=1}^m w_l \frac{([D^l]^-)^2}{\bar{x}^2} \leq 1 - \frac{4}{11}\sum_{l=1}^m \omega^l \frac{([D^l]^-)^2}{\bar{x}^2}. \tag{73}$$

This proves (70).

By Jensen's inequality, we have

$$\sum_{l=1}^m \omega^l \|q_t^l - \bar{q}_t\|_1^2 \geq \left(\sum_{l=1}^m \omega^l \|q_t^l - \bar{q}_t\|_1\right)^2,$$

which combining with (69) completes the proof. $\qquad\square$

Note that from (67) we immediately have

$$1 = \sum_{l=1}^m \omega^l \|\pi^l(u_{t+1}^l, v_t^l, U_t)\|_1 \geq \sum_{l=1}^m \omega^l \|\pi^l(u_{t+1}^l, v_{t+1}^l, U_t)\|_1, \tag{74}$$

where the equality is from the optimality condition of (18).

Notice (20) is a Riemannian gradient descent step. To prove the objective function $g(\mathbf{u}, \mathbf{v}, U)$ has sufficient decrease in (20), we first prove the following Lipschitz continuous condition. The proof of Lemma 21 mainly follows (Huang et al., 2020)[Lemma 4.8].

Lemma 21 *Let $\{(\mathbf{u}_t, \mathbf{v}_t, U_t)\}$ be the sequence generated by Algorithm 2. For any $U \in \mathcal{M}$, we have the following inequality holds:*

$$g(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U) \leq g(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t) + \langle \nabla_U g(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t), U - U_t \rangle + \frac{\rho}{2}\|U_t - U\|_F^2,$$

where $\rho = 2\bar{c}/\eta + 4\sqrt{k}\bar{c}^2/\eta^2$.

Proof. For any $\alpha \in [0, 1]$, denote $U_\alpha = \alpha U + (1 - \alpha)U_t$. Note that U_α is not necessarily on $\mathcal{M}$, though $U \in \mathcal{M}$ and $U_t \in \mathcal{M}$. Note that $\nabla_U g(\mathbf{u}, \mathbf{v}, U) = -\frac{2}{\eta}V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t)}U$. Therefore, we have

$$\|\nabla_U g(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t) - \nabla_U g(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_\alpha)\|_F \tag{75}$$

$$= \frac{2}{\eta}\|V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t)}U_t - V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_\alpha)}U_\alpha\|_F$$

$$= \frac{2}{\eta}\|V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t)}U_t - V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t)}U_\alpha + V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t)}U_\alpha - V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_\alpha)}U_\alpha\|_F$$

$$\leq \frac{2}{\eta}\|V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t)}(U_t - U_\alpha)\|_F + \frac{2}{\eta}\|(V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t)} - V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_\alpha)})U_\alpha\|_F$$

$$\leq \frac{2}{\eta}\|V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t)}\|_F\|U_t - U_\alpha\|_F + \frac{2\sqrt{k}}{\eta}\|V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t)} - V_{\boldsymbol{\pi}(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_\alpha)}\|_F,$$

where the last inequality is due to the fact that $\|U_\alpha\|_F \leq \alpha\|U\|_F + (1 - \alpha)\|U_t\|_F = \sqrt{k}$. By using (74), we have

$$\|V_{\pi(\mathbf{u}_{t+1}, \mathbf{v}_{t+1}, U_t)}\|_F \leq \sum_{l=1}^m \omega^l \sum_{i,j}[\pi^l(u_{t+1}^l, v_{t+1}^l, U_t)]_{ij}\|(x_i^l - y_j)(x_i^l - y_j)^\top\|_F \leq \max_{l,i,j}\|x_i^l - y_j\|^2 = \max_l \|C^l\|_\infty. \tag{76}$$

Note that for fixed U, each element of the objective function $f_\eta^l(\pi^l, U) := \sum_{i,j=1}^n \pi_{i,j}^l \|U^\top(x_i^l - y_j)\|^2 - \eta H(\pi^l)$ is η-strongly convex with respect to π^l under the ℓ_1 norm metric, which implies

$$
\begin{aligned}
f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_t), U_\alpha) &\geq f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha), U_\alpha) + \langle \nabla_{\pi^l} f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha), U_\alpha), \\
&\quad \pi^l(u_{t+1}^l, v_{t+1}^l, U_t) - \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha)\rangle + \frac{\eta}{2}\|\pi^l(u_{t+1}^l, v_{t+1}^l, U_t) - \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha)\|_1^2 \\
f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha), U_\alpha) &\geq f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_t), U_\alpha) + \langle \nabla_{\pi^l} f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_t), U_\alpha), \\
&\quad \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha) - \pi^l(u_{t+1}^l, v_{t+1}^l, U_t)\rangle + \frac{\eta}{2}\|\pi^l(u_{t+1}^l, v_{t+1}^l, U_t) - \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha)\|_1^2.
\end{aligned}
\tag{77}
$$

By adding the above two inequalities, we have

$$
\begin{aligned}
&\langle \nabla_{\pi^l} f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha), U_\alpha) - \nabla_{\pi^l} f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_t), U_\alpha), \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha) - \pi^l(u_{t+1}^l, v_{t+1}^l, U_t)\rangle \\
&\geq \eta\|\pi^l(u_{t+1}^l, v_{t+1}^l, U_t) - \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha)\|_1^2.
\end{aligned}
\tag{78}
$$

Moreover, note that

$$
(\nabla_{\pi^l} f_\eta^l(\pi^l, U))_{ij} = \|U^\top(x_i^l - y_j)\|^2 + \eta \log(\pi_{ij}^l),
\tag{79}
$$

which, combining with (39), yields

$$
\begin{aligned}
[\nabla_{\pi^l} f_\eta^l(\pi^l(u^l, v^l, U), U)]_{ij} &= \|U^\top(x_i^l - y_j)\|^2 + \eta \log([\pi^l(u^l, v^l, U)]_{ij}) \\
&= \|U^\top(x_i^l - y_j)\|^2 + \eta\left(-\frac{1}{\eta}\|U^\top(x_i^l - y_j)\|^2 + u_i^l + v_j^l\right) \\
&= \eta(u_i^l + v_j^l).
\end{aligned}
$$

This implies that $\nabla_{\pi^l} f_\eta^l(\pi^l(u^l, v^l, U), U)$ is independent of U. Therefore, (78) is equivalent to

$$
\begin{aligned}
&\langle \nabla_{\pi^l} f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_t), U_t) - \nabla_{\pi^l} f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_t), U_\alpha), \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha) - \pi^l(u_{t+1}^l, v_{t+1}^l, U_t)\rangle \\
&\geq \eta\|\pi^l(u_{t+1}^l, v_{t+1}^l, U_t) - \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha)\|_1^2,
\end{aligned}
$$

which, by Hölder's inequality, further yields,

$$
\begin{aligned}
&\eta\|\pi^l(u_{t+1}^l, v_{t+1}^l, U_t) - \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha)\|_1 \\
&\leq \|\nabla_{\pi^l} f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_t), U_t) - \nabla_{\pi^l} f_\eta^l(\pi^l(u_{t+1}^l, v_{t+1}^l, U_t), U_\alpha)\|_\infty \\
&\leq \max_{i,j} |\|(U_\alpha)^\top(x_i^l - y_j)\|_2^2 - \|(U_t)^\top(x_i^l - y_j)\|_2^2| \\
&= \max_{i,j} |(x_i^l - y_j)^\top(U_\alpha(U_\alpha)^\top - U_t(U_t)^\top)(x_i^l - y_j)| \\
&\leq (\max_{i,j}\|x_i^l - y_j\|^2)\|U_\alpha(U_\alpha)^\top - U_t(U_t)^\top\|_F \\
&= \|C^l\|_\infty\|U_\alpha(U_\alpha)^\top - U_t(U_t)^\top\|_F,
\end{aligned}
\tag{80}
$$

where the second inequality follows from (79). Furthermore, since $U, U_t \in \mathcal{M}$, we have

$$
\begin{aligned}
&\|U_\alpha(U_\alpha)^\top - U_t(U_t)^\top\|_F \\
&= \|U_\alpha(U_\alpha)^\top - U_t(U_\alpha)^\top + U_t(U_\alpha)^\top - U_t(U_t)^\top\|_F \\
&\leq \|(U_\alpha - U_t)(U_\alpha)^\top\|_F + \|U_t(U_\alpha - U_t)^\top\|_F \\
&\leq \|(U_\alpha - U_t)(\alpha U + (1-\alpha)U_t)^\top\|_F + \|U_\alpha - U_t\|_F \\
&\leq \alpha\|(U_\alpha - U^t)U^\top\|_F + (1-\alpha)\|(U_\alpha - U_t)(U_t)^\top\|_F + \|(U_\alpha - U_t)\|_F \\
&= 2\|U_\alpha - U_t\|_F.
\end{aligned}
\tag{81}
$$

By combining (80) and (81), we have

$$\|V_{\boldsymbol{\pi}(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t)} - V_{\boldsymbol{\pi}(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_\alpha)}\|_F \tag{82}$$

$$\leq \sum_{l=1}^{m} \omega^l \left[\sum_{i,j}^{n} |\pi^l(u_{t+1}^l, v_{t+1}^l, U_t)_{i,j} - \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha)_{i,j}| \times \|(x_i^l - y_j)(x_i^l - y_j)^T\|_F \right]$$

$$\leq \sum_{l=1}^{m} \omega^l \|C^l\|_\infty \|\pi^l(u_{t+1}^l, v_{t+1}^l, U_t) - \pi^l(u_{t+1}^l, v_{t+1}^l, U_\alpha)\|_1$$

$$\leq \sum_{l=1}^{m} \omega^l \frac{2\|C^l\|_\infty^2}{\eta} \|U_t - U_\alpha\|_F \leq \frac{2\bar{c}^2}{\eta} \sum_{l=1}^{m} \omega^l \|U_t - U_\alpha\|_F = \frac{2\bar{c}^2}{\eta} \|U_t - U_\alpha\|_F.$$

Plugging (76) and (82) into (75) yields:

$$\|\nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t) - \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_\alpha)\|_F \leq \rho\|U_t - U_\alpha\|_F.$$

We then have

$$|g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U) - g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t) - \langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), U - U_t \rangle|$$

$$= \left| \int_0^1 \langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},\alpha U + (1-\alpha)U_t) - \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), U - U_t \rangle d\alpha \right|$$

$$\leq \int_0^1 \|\nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},\alpha U + (1-\alpha)U_t) - \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t)\|_F \|U - U_t\|_F d\alpha \tag{83}$$

$$\leq \int_0^1 \rho\alpha\|U - U_t\|_F^2 d\alpha$$

$$= \frac{\rho}{2} \|U - U_t\|_F^2,$$

which completes the proof. $\qquad\square$

We now prove that function g is decreasing after updating U.

Lemma 22 (Decrease of g in U) *Let $\{(u^t, v^t, U^t)\}$ be the sequence generated by Algorithm 2. For any $t \geq 0$, the following inequality holds*

$$g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_{t+1}) - g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t) \leq -\frac{1}{8L_2\bar{c}/\eta + 2\rho L_1^2} \|\xi_{t+1}\|_F^2, \tag{84}$$

where ρ is defined in Lemma 21, L_1 and L_2 are defined in Proposition 13, and $\xi_{t+1} := \text{grad}_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t)$.

Proof. By setting $U = U_{t+1} = \text{Retr}_{U_t}(-\tau\xi_{t+1})$ in Lemma 21, we have,

$$g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_{t+1}) - g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t) \leq \langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), U_{t+1} - U_t \rangle + \frac{\rho}{2}\|U_{t+1} - U_t\|_F^2$$

$$\leq \langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), U_{t+1} - U_t \rangle + \frac{\rho\tau^2 L_1^2}{2}\|\xi_{t+1}\|_F^2, \tag{85}$$

where the last inequality follows from Proposition 13. We then have

$$\langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), U_{t+1} - U_t \rangle$$

$$= \langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), \text{Retr}_{U_t}(-\tau\xi_{t+1}) - U_t \rangle$$

$$= \langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), -\tau\xi_{t+1} \rangle + \langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), \text{Retr}_{U_t}(-\tau\xi_{t+1}) - (U_t - \tau\xi_{t+1}) \rangle$$

$$\leq -\tau\langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), \xi_{t+1} \rangle + \|\nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t)\|_F \|\text{Retr}_{U_t}(-\tau\xi_{t+1}) - (U_t - \tau\xi_{t+1})\|_F$$

$$\leq -\tau\langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), \xi_{t+1} \rangle + \frac{2}{\eta}\|V_{\boldsymbol{\pi}(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t)}U^t\|_F \cdot L_2\tau^2\|\xi_{t+1}\|_F^2 \tag{86}$$

$$\leq -\tau\langle \nabla_U g(\boldsymbol{u}_{t+1},\boldsymbol{v}_{t+1},U_t), \xi_{t+1} \rangle + \frac{2}{\eta}L_2\tau^2\bar{c}\|\xi_{t+1}\|_F^2$$

$$= -\tau\|\xi_{t+1}\|_F^2 + \frac{2}{\eta}L_2\tau^2\bar{c}\|\xi_{t+1}\|_F^2,$$

where the second inequality follows from Proposition 13, and the last inequality is due to (76). Combining (85) and (86) yields,

$$g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}, U_{t+1}) - g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}, U_t) \leq -\tau \left(1 - \left(\frac{2}{\eta} L_2 \bar{c} + \frac{\rho}{2} L_1^2 \right) \tau \right) \|\xi_{t+1}\|_F^2.$$

Finally, choosing $\tau = \frac{1}{4L_2\bar{c}/\eta + \rho L_1^2}$ gives the desired result (84). □

We now prove Theorem 8.

Proof. By combining Lemmas 22, 19 and 20, we have:

$$g(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}, U_{t+1}) - g(\boldsymbol{u}_t, \boldsymbol{v}_t, U_t) \tag{87}$$
$$\leq - \left(\frac{1}{11} \left(\sum_{l=1}^m \omega^l \|q_t^l - \bar{q}_t\|_1 \right)^2 + \frac{1}{8L_2\bar{c}/\eta + 2\rho L_1^2} \|\xi_{t+1}\|_F^2 \right).$$

Suppose Algorithm 2 terminates at the T-th iteration. Summing (87) over $t = 0, \dots, T-1$ yields

$$g(\boldsymbol{u}_T, \boldsymbol{v}_T, U_T) - g(\boldsymbol{u}_0, \boldsymbol{v}_0, U_0) \tag{88}$$
$$\leq - \sum_{t=0}^{T-1} \left(\frac{1}{11} \left(\sum_{l=1}^m \omega^l \|q_t^l - \bar{q}_t\|_1 \right)^2 + \frac{1}{8L_2\bar{c}/\eta + 2\rho L_1^2} \|\xi_{t+1}\|_F^2 \right)$$
$$= - \sum_{t=0}^{T-1} \left(\frac{1}{11} \left(\sum_{l=1}^m \omega^l \|q_t^l - \bar{q}_t\|_1 \right)^2 + \frac{\eta^2 \|\xi^{t+1}\|_F^2}{(8L_2\bar{c} + 4L_1^2\bar{c})\eta + 8L_1^2\sqrt{k}\bar{c}^2} \right)$$
$$\leq - \sum_{t=0}^{T-1} \min \left\{ \frac{1}{11}, \frac{1}{(8L_2\bar{c} + 4L_1^2\bar{c})\eta + 8L_1^2\sqrt{k}\bar{c}^2} \right\} \cdot \left(\left(\sum_{l=1}^m \omega^l \|q_t^l - \bar{q}_t\|_1 \right)^2 + \eta^2 \|\xi^{t+1}\|_F^2 \right)$$
$$\leq - T \cdot \min \left\{ \frac{1}{11}, \frac{1}{(8L_2\bar{c} + 4L_1^2\bar{c})\eta + 8L_1^2\sqrt{k}\bar{c}^2} \right\} \cdot \min \left\{ \frac{\omega^3 \epsilon^2}{144\bar{c}^2}, \frac{\epsilon^2}{9} \right\},$$

where the equality is obtained by plugging in the definition of ρ in (27), and the last inequality follows from the fact that the stopping criteria in Algorithm 2 does not hold for $t < T$. By combining with (62) and (27), (88) immediately leads to

$$T \leq (g(\boldsymbol{u}_0, \boldsymbol{v}_0, U_0) - g^*) \cdot \max \left\{ 11, (8L_2\bar{c} + 4L_1^2\bar{c})\eta + 8L_1^2\sqrt{k}\bar{c}^2 \right\} \cdot \max \left\{ \frac{144\bar{c}^2}{\omega^3 \epsilon^2}, \frac{9}{\epsilon^2} \right\} \tag{89}$$
$$\leq \left(g(u_0, v_0, U_0) - 1 + \frac{\bar{c}}{\eta} \right) \cdot \max \left\{ 11, (8L_2\bar{c} + 4L_1^2\bar{c})\eta + 8L_1^2\sqrt{k}\bar{c}^2 \right\} \cdot \max \left\{ \frac{144\bar{c}^2}{\omega^3 \epsilon^2}, \frac{9}{\epsilon^2} \right\}$$
$$= O \left(\frac{L_1^2\bar{c}^5 \log(n)\sqrt{k}}{\omega^3 \epsilon^3} \right),$$

where g^* is defined in (62). This completes the proof of Theorem 8. □

Proof of Corollary 9. We further analyze the per-iteration complexity for Algorithm 2. Notice that in each iteration, we need to compute the projected cost matrices $\{M^l\}_{l \in [m]}$, which takes $O(mn^2dk)$ arithmetic operations. Secondly, steps (21) - (22) can be done in $O(mn + mn^2)$ arithmetic operations. Moreover, the retraction operation requires $O(dk^2 + k^3)$ arithmetic operations and the complexity of computing $V_\pi U$ is $O(mn^2dk)$. Therefore, the per-iteration arithmetic operations complexity of Algorithm 2 is

$$O(mn^2dk + mn + mn^2 + dk^2 + k^3 + mn^2dk),$$

which combining with Theorem 8 proves Corollary 9.

F. Proof of Theorem 6 and Corollary 7

The proof of Theorem 6 includes two parts. We first show that the objective function $f_\eta(U)$ defined in (12) is monotonically increasing in Algorithm 1, which leads to the iteration complexity for obtaining an ϵ-stationary point. We then analyze the complexity of the WB subproblem in each iteration. The following lemma shows $f_\eta(U)$ is Lipschitz continuous.

Lemma 23 *For any $U_1, U_2 \in \mathrm{St}(d, k)$, we have the following inequality holds:*

$$|f_\eta(U_1) - f_\eta(U_2) - \langle \nabla f_\eta(U_2), U_1 - U_2 \rangle| \le \frac{\rho}{2}\|U_1 - U_2\|_F^2,$$

where $\rho = 2\bar{c} + \frac{4\bar{c}^2}{\eta}$.

Proof. The proof of this lemma mainly follows the proof of (Lin et al., 2020a)[Lemma 3.2] and Lemma 21. Denote $U_\alpha = \alpha U_1 + (1 - \alpha)U_2$. The gradient of $f_\eta(U)$ is $\nabla f_\eta(U) = 2V_{\boldsymbol{\pi}_\eta^*(U)}U$, indicating

$$\|\nabla f_\eta(U_\alpha) - \nabla f_\eta(U_2)\|_F \le 2\|V_{\boldsymbol{\pi}_\eta^*(U_\alpha)}\|_F\|U_\alpha - U_2\|_F + 2\|V_{\boldsymbol{\pi}_\eta^*(U_\alpha)} - V_{\boldsymbol{\pi}_\eta^*(U_2)}\|_F, \tag{90}$$

where $\boldsymbol{\pi}_\eta^*(U)$ denotes the optimal solution of (12). Notice that $\sum_{l=1}^m \omega^l \sum_{i,j}^n (\pi_\eta^*(U))_{i,j}^l = 1$, we have

$$\|V_{\boldsymbol{\pi}_\eta^*(U_\alpha)}\|_F \le \sum_{l=1}^m \omega^l \sum_{i,j} (\pi_\eta^*(U_\alpha))_{i,j}^l \|(x_i^l - y_j)(x_i^l - y_j)^\top\|_F \le \max_{l,i,j} |x_i^l - y_j|^2 = \max_l \|C^l\|_\infty. \tag{91}$$

Following the idea of (77) and (78) we have

$$\langle \nabla_{\pi^l} f_\eta^l((\pi_\eta^*(U_\alpha))^l, U_\alpha) - \nabla_{\pi^l} f_\eta^l((\pi_\eta^*(U_2))^l, U_\alpha), (\pi_\eta^*(U_\alpha))^l - (\pi_\eta^*(U_2))^l \rangle \ge \eta\|(\pi_\eta^*(U_\alpha))^l - (\pi_\eta^*(U_2))^l\|_1^2. \tag{92}$$

By first order optimality condition of (12), we have

$$\langle \nabla_{\pi^l} f_\eta^l((\pi_\eta^*(U_\alpha))^l, U_\alpha), (\pi_\eta^*(U_2))^l - (\pi_\eta^*(U_\alpha))^l \rangle \ge 0,$$
$$\langle \nabla_{\pi^l} f_\eta^l((\pi_\eta^*(U_2))^l, U_2), (\pi_\eta^*(U_\alpha))^l - (\pi_\eta^*(U_2))^l \rangle \ge 0, \tag{93}$$

which leads to

$$\langle \nabla_{\pi^l} f_\eta^l((\pi_\eta^*(U_2))^l, U_2) - \nabla_{\pi^l} f_\eta^l((\pi_\eta^*(U_\alpha))^l, U_\alpha), (\pi_\eta^*(U_\alpha))^l - (\pi_\eta^*(U_2))^l \rangle \ge 0. \tag{94}$$

By adding (92) and (94) together, we have

$$\langle \nabla_{\pi^l} f_\eta^l((\pi_\eta^*(U_2))^l, U_2) - \nabla_{\pi^l} f_\eta^l((\pi_\eta^*(U_2))^l, U_\alpha), (\pi_\eta^*(U_\alpha))^l - (\pi_\eta^*(U_2))^l \rangle \ge \eta\|(\pi_\eta^*(U_\alpha))^l - (\pi_\eta^*(U_2))^l\|_1^2. \tag{95}$$

The rest of the proof follows (80) - (83). $\qquad\square$

The next lemma proves equation (42).

Proof. Notice that when fixing U_t, each $f_\eta^l(\pi^l, U) = \sum_{i,j=1}^n \pi_{i,j}^l \|U^\top(x_i^l - y_j)\|^2 - \eta H(\pi^l)$ is η-strongly convex with respect to π^l under the ℓ_1 norm metric, which implies

$$f_\eta^l(\pi^l, U_t) \ge f_\eta^l((\pi_\eta^*)^l, U_t) + \frac{\eta}{2}\|\pi^l - (\pi_\eta^*)^l\|_1^2. \tag{96}$$

In the j-th iteration of the IBP subroutine (Algorithm 4), we have

$$\frac{\eta}{2}\left(\sum_{l=1}^m \omega^l\|\pi_j^l - (\pi_\eta^*)^l\|_1\right)^2 \le \frac{\eta}{2}\sum_{l=1}^m \omega^l\|\pi_j^l - (\pi_\eta^*)^l\|_1^2 \le f_\eta(\boldsymbol{\pi}_j, U_t) - f_\eta(\boldsymbol{\pi}_\eta^*, U_t)$$

$$= -\eta(g(\boldsymbol{u}_j, \boldsymbol{v}_j, U_t) - g(\boldsymbol{u}_\eta^*, \boldsymbol{v}_\eta^*, U_t)) + \eta\sum_{l=1}^m \omega^l\langle v_j^l, q^l \rangle \tag{97}$$

$$\le \eta\sum_{l=1}^m \omega^l\langle v_j^l, q^l \rangle,$$

where the first inequality is by Jensen's inequality, and the last equality uses Lemma 16 and the fact that

$$f_\eta(\boldsymbol{\pi}_\eta^*, U_t) = -\eta g(\boldsymbol{u}_\eta^*, \boldsymbol{v}_\eta^*, U_t), \tag{98}$$

which can be proved by plugging $\boldsymbol{\pi}_\eta^*, \boldsymbol{u}_\eta^*, \boldsymbol{v}_\eta^*$ into Lemma 16 and $\sum_{l=1}^m \omega^l (v_\eta^*)^l = 0$. Using the similar idea in (50), we bound the term $\sum_{l=1}^m \omega^l \langle v^l, q^l \rangle$ as follows.

$$
\begin{aligned}
\sum_{l=1}^m \omega^l \langle v_j^l, q_j^l \rangle &= \sum_{l=1}^m \omega^l \langle v_j^l, q_j^l - \bar{q}_j \rangle = \sum_{l=1}^m \omega^l \langle v_j^l - b_j^l \mathbf{1}, q_j^l - \bar{q}_j \rangle \\
&\leq \sum_{l=1}^m \omega^l \|v_j^l - b_j^l \mathbf{1}\|_\infty \|q_j^l - \bar{q}_j\|_1 \leq R \sum_{l=1}^m \omega^l \|q_j^l - \bar{q}_j\|_1,
\end{aligned}
\tag{99}
$$

where $b_j^l = \frac{\min_i [v_j^l]_i + \max_i [v_j^l]_i}{2}$, $R = \bar{c}/\eta$. The first equality in (99) comes from $\sum_{l=1}^m \omega^l v^l = 0$, the second equality in (99) is due to $\langle q_j^l, \mathbf{1} \rangle = 1, \langle \bar{q}_j, \mathbf{1} \rangle = 1$, and the last inequality in (99) is by Hölder's inequality and (Kroshnin et al., 2019)[Lemma 4]. Therefore, if the Algorithm 4 terminates at the t-th iteration, then combining (97) with (99) yields

$$\left(\sum_{l=1}^m \omega^l \|\pi_t^l - (\pi_\eta^*)^l\|_1 \right)^2 \leq 2 \sum_{l=1}^m \omega^l \langle v_t^l, q_t^l \rangle \leq 2R \sum_{l=1}^m \omega^l \|q_t^l - \bar{q}_t\|_1 \leq 2R \cdot \frac{\epsilon^2}{200\bar{c}^2 R} = \frac{\epsilon^2}{100\bar{c}^2}, \tag{100}$$

which leads to (42). $\qquad\square$

The next lemma shows the iteration complexity for obtaining U_T that satisfies $\|\xi_T\| \leq \epsilon$.

Lemma 24 *Choose parameters as in* (24). *The Algorithm 1 terminates in T iterations, where T is defined in* (25).

Proof. By Lemma 23 and the definition of U_{t+1}, we have

$$
\begin{aligned}
f_\eta(U_{t+1}) - f_\eta(U_t) &\geq \langle \nabla f_\eta(U_t), U_{t+1} - U_t \rangle - \frac{\rho}{2} \|U_{t+1} - U_t\|_F^2 \\
&= \langle \nabla f_\eta(U_t), \tau \xi_{t+1} \rangle + \langle \nabla f_\eta(U_t), \mathrm{Retr}_{U_t}(\tau\xi_{t+1}) - (U_t + \tau\xi_{t+1}) \rangle - \frac{\rho}{2} \|U_{t+1} - U_t\|_F^2 \\
&\geq \tau \langle \nabla f_\eta(U_t), \xi_{t+1} \rangle - \|\nabla f_\eta(U_t)\|_F \|\mathrm{Retr}_{U_t}(\tau\xi_{t+1}) - (U_t + \tau\xi_{t+1})\|_F - \frac{\rho}{2} \|U_{t+1} - U_t\|_F^2 \\
&\geq \tau \langle \mathrm{grad} f_\eta(U_t), \xi_{t+1} \rangle - 2\tau^2 L_2 \bar{c} \|\xi_{t+1}\|_F^2 - \frac{1}{2}\rho\tau^2 L_1^2 \|\xi_{t+1}\|_F^2,
\end{aligned}
\tag{101}
$$

where the last inequality uses Proposition 13 and $\|\nabla f_\eta(U_t)\|_F = \|2V_{\boldsymbol{\pi}_\eta^*(U_t)} U_t\|_F \leq 2\bar{c}$. We further bound (101) by the following ineuqalities:

$$\langle \mathrm{grad} f_\eta(U_t), \xi_{t+1} \rangle \geq \frac{1}{2}(\|\xi_{t+1}\|_F^2 - \|\xi_{t+1} - \mathrm{grad} f_\eta(U_t)\|_F^2). \tag{102}$$

Combining (101) - (102) yields

$$f_\eta(U_{t+1}) - f_\eta(U_t) \geq \tau \left[\frac{1}{2} - \tau \left(2L_2\bar{c} + \frac{1}{2}\rho L_1^2 \right) \right] \|\xi_{t+1}\|_F^2 - \frac{1}{2}\tau \|\xi_{t+1} - \mathrm{grad} f_\eta(U_t)\|_F^2. \tag{103}$$

Notice that the IBP subroutine in each iteration of Algorithm 1 returns $\boldsymbol{\pi}_{t+1}$ satisfying (42). Therefore, we have

$$
\begin{aligned}
\|\xi_{t+1} - \mathrm{grad} f_\eta(U_t)\|_F &= 2\|(V_{\boldsymbol{\pi}_{t+1}} - V_{\boldsymbol{\pi}_\eta^*(U_t)})U_t\|_F \leq 2\|V_{\boldsymbol{\pi}_{t+1}} - V_{\boldsymbol{\pi}_\eta^*(U_t)}\|_F \\
&\leq 2\bar{c} \sum_{l=1}^m \omega^l \|\pi_{t+1}^l - \pi_\eta^*(U_t)^l\|_1 \leq \frac{\epsilon}{5}.
\end{aligned}
\tag{104}
$$

Plugging $\tau = \frac{1}{8L_2\bar{c} + 2\rho L_1^2}$ into (103) and combining with (104), we have

$$f_\eta(U_{t+1}) - f_\eta(U_t) \geq \frac{\tau}{4}\|\xi_{t+1}\|_F^2 - \frac{\tau\epsilon^2}{50}. \tag{105}$$

Assume Algorithm 1 stops at the T-th iteration. For any $t < T$, we have $\|\xi_{t+1}\|_F > \epsilon$. Summing (105) over $t = 0, \ldots, T-1$ yields

$$f_\eta^* - f_\eta(U_0) \geq f_\eta(U_T) - f_\eta(U_0) \geq T \cdot \frac{23\tau\epsilon^2}{100}, \tag{106}$$

where f_η^* denotes the maximal value of f_η. By Lemma 23, we have

$$f_\eta^* - f_\eta(U_0) \leq \langle \nabla f_\eta(U^*), U_0 - U^* \rangle + \frac{\rho}{2}\|U^* - U_0\|_F^2 \leq \|\nabla f_\eta(U^*)\|_F \|U_0 - U^*\|_F + k\rho \leq 4\sqrt{k}\bar{c} + k\rho, \tag{107}$$

where the last inequality comes from $\nabla f_\eta(U) = 2V_{\boldsymbol{\pi}_\eta^*(U)}U$ and (91). Combining (106) with (107) and the definition of τ, ρ yields

$$T \leq \frac{100(4\sqrt{k}\bar{c} + k\rho)(8L_2\bar{c} + 2L_1^2\rho)}{23\epsilon^2} = O\left(\frac{k\log(n)^2L_1^2\bar{c}^4}{\epsilon^4}\right). \tag{108}$$

This completes the proof. $\qquad\square$

Proof of Theorem 6. *Proof.* By Lemma 24, we have the iteration complexity of Algorithm 1. The rest of the proof is to show that when Algorithm 1 stops, $(\hat{\boldsymbol{\pi}}, \hat{U})$ is an ϵ-stationary point of (8). We first notice that the stopping criteria guarantees

$$\|\mathrm{grad}_U f(\hat{\boldsymbol{\pi}}, \hat{U})\|_F = \|\mathrm{Proj}_{T_U \mathrm{St}}(2V_{\boldsymbol{\pi}_{T+1}}U_T)\|_F = \|\xi_{T+1}\|_F \leq \epsilon,$$

which verifies (9). Secondly, $\boldsymbol{\pi}_\eta^*$ is the optimal solution of the regularized WB problem and $\eta = \frac{\epsilon}{4\log(n)+2}$, we have

$$\langle U_T U_T^\top, V_{\boldsymbol{\pi}_\eta^*(U_T)}\rangle \leq \langle U_T U_T^\top, V_{\boldsymbol{\pi}^*(U_T)}\rangle - \eta\sum_{l=1}^m \omega^l H(\pi^*(U_T)^l) + \eta\sum_{l=1}^m \omega^l H(\pi_\eta^*(U_T)^l)$$
$$\leq \langle U_T U_T^\top, V_{\boldsymbol{\pi}^*(U_T)}\rangle + \frac{\epsilon}{2}, \tag{109}$$

where in the last step we use $0 \leq H(\pi) \leq 2\log(n) + 1$. By the stopping criteria of the IBP subroutine, we have

$$0 \leq \langle U_T U_T^\top, V_{\boldsymbol{\pi}_{T+1}} - V_{\boldsymbol{\pi}_\eta^*(U_T)}\rangle \leq \bar{c}\sum_{l=1}^m \omega^l \|\pi_{T+1}^l - \pi_\eta^*(U_T)^l\|_1 \leq \frac{\epsilon}{10}. \tag{110}$$

Adding the above two inequalities shows that (10) holds, which completes the proof. $\qquad\square$

Proof of Corollary 7. *Proof.* We first analyze the per-iteration complexity of Algorithm 1. The computation of $\{M^l\}_{l \in [m]}$, the retraction and $V_\pi U$ requires $O(mn^2d)$, $O(dk^2 + k^3)$ and $O(mn^2dk)$ arithmetic operations. By (Kroshnin et al., 2019)[Theorem 1], it takes $O(mn^2\epsilon'^{-2})$ arithmetic operations for the IBP algorithm to satisfy $\sum_{l=1}^m \omega^l\|q_j^l - \bar{q}_j\|_1 \leq \epsilon'$. In our case, we need to bound $\sum_{l=1}^m \omega^l\|q_j^l - \bar{q}_j\|_1$ by

$$\eta\epsilon^2/(200\bar{c}^3) = O\left(\frac{\epsilon^3}{\bar{c}^3\log(n)}\right). \tag{111}$$

Therefore, it takes

$$O\left(mn^2\frac{\bar{c}^6\log(n)^2}{\epsilon^6}\right)$$

arithmetic operations to satisfy $\sum_{l=1}^m \omega^l\|\pi_j^l - (\pi_\eta^*)^l\|_1 \leq \epsilon/(10\bar{c})$. The total per-iteration complexity of Algorithm 1 is

$$O\left(mn^2d + dk^2 + k^3 + mn^2dk + mn^2\frac{\bar{c}^6\log(n)^2}{\epsilon^6}\right),$$

which, together with (25) gives the total arithmetic operations complexity given in Corollary 7. $\qquad\square$

G. Additional Details for Numerical Experiments

We provide more details for the numerical experiments in Section 6.

G.1. Wasserstein Barycenter of Gaussian distributions

The ground truth Wasserstein Barycenter of a set of Gaussian distributions can be computed by an iterative method introduced in (Agueh & Carlier, 2011), specifically the following theorem.

Theorem 25 ((Agueh & Carlier, 2011)) *Assume $\Sigma^1, ..., \Sigma^m$ are symmetric $d \times d$ positive semidefinite matrices, with at least one of them positive definite. Consider some symmetric, positive definite S_0 and define*

$$S_{n+1} = S_n^{-1/2}(\sum_{l=1}^{m} \omega^l \left(S_n^{1/2} \Sigma^l S_n^{1/2} \right)^{1/2})^2 S_n^{-1/2}, n \geq 0. \tag{112}$$

If $\mathcal{N}(0, \Sigma^0)$ is the barycenter of $\mathcal{N}(0, \Sigma^1), ..., \mathcal{N}(0, \Sigma^m)$, then

$$W_2(\mathcal{N}(0, S_n), \mathcal{N}(0, \Sigma^0)) \to 0$$

as $n \to \infty$. Furthermore, the barycenter value can be computed as

$$WB(\{\mu^l\}_{l=1}^{m}) = Tr(\Sigma^0) + \sum_{l=1}^{m} \omega^l Tr(\Sigma^l) - 2 \sum_{l=1}^{m} \omega^l Tr(((\Sigma^0)^{1/2} \Sigma^l (\Sigma^0)^{1/2})^{1/2}).$$

G.2. D2 Clustering and Projected D2 Clustering

We present the D2 clustering and Projected D2 clustering in Algorithm 5. Specially, Algorithm 5 with Option 1 gives D2 clustering, and Algorithm 5 with Option 2 gives PD2 clustering. The D2 clustering follows the idea of k-means clustering but clusters discrete distributions under the Wasserstein metric. In each iteration, the D2 Clustering algorithm calculates the Wasserstein distance between each distribution and each barycenter and relabel the distributions. Based on the updated labels, we recalculate the Wasserstein Barycenter for each cluster.

Algorithm 5 (Projected) D2-clustering

1: **Input:** $\{\mu_n^i = (p^i, X^i)\}_{i \in [N]}$, K
2: Initialize the labels for each distribution, denoted as labels$[i], i \in [N]$;
3: Choosing K random distributions as the barycenters $\{Q_i = (q_i, Y_i), i \in [K]\}$;
4: **for** $t = 0, 1, 2, \ldots,$ **do**
5:    **for** $i = 0, \ldots, N$ **do**
6:       labels$[i] = \text{argmin}_{j \in K} \mathcal{W}(Q_j, \mu_n^i)$
7:    **end for**
8:    **for** $i = 0, \ldots, K$ **do**
9:       Option 1: $Q_i = FreeSupportWB(\{\mu^l : \text{labels}[l] = i\}, Y_i)$
10:      Option 2: $Q_i = FreeSupportRPRWB(\{\mu^l : \text{labels}[l] = i\}, Y_i)$
11:    **end for**
12: **end for**
13: **Output:** labels$[i], i \in [N]$.

The D2 clustering algorithm solves a fixed-support WB problem in each iteration using Algorithm 6, and PD2 clustering algorithm solves a fixed-support RPRWB problem in each iteration using Algorithm 7.

Algorithm 6 Free-support WB solver

1: **Input:** $\{\mu^l = (p^l, X^l)\}_{l \in [m]}, Y^0$
2: $OBJ = WB(\{\mu^l\}_{l \in [m]}, Y^0)$
3: **for** $t = 0, 1, 2, \ldots,$ **do**
4:    $OBJ\_old = OBJ$;
5:    Compute $\Pi^t = WBsolver(\{\mu^l\}_{l \in [m]}, Y^t)$, $q^t = \frac{1}{m}\Pi^t \mathbf{1}_n$, and OBJ;
6:    Compute $Y^{t+1} = \frac{1}{m}X(\Pi^t)^\top \mathrm{diag}(1/q)$;
7:    **if** $OBJ > OBJ\_old$ **then**
8:       break;
9:    **end if**
10: **end for**
11: **Output:** Π^t, Y^t.

Algorithm 7 Free-support RPRWB solver

1: **Input:** $\{\mu^l = (p^l, X^l)\}_{l \in [m]}, Y^0$
2: $OBJ = PRWB(\{\mu^l\}_{l \in [m]}, Y^0)$
3: **for** $t = 0, 1, 2, \ldots,$ **do**
4:    $OBJ\_old = OBJ$;
5:    Compute $\Pi^t = RBCD(\{\mu^l\}_{l \in [m]}, Y^t)$, $q^t = \frac{1}{m}\Pi^t \mathbf{1}_n$, and OBJ;
6:    Compute $Y^{t+1} = \frac{1}{m}X(\Pi^t)^\top \mathrm{diag}(1/q)$;
7:    **if** $OBJ > OBJ\_old$ **then**
8:       break;
9:    **end if**
10: **end for**
11: **Output:** Π^t, Y^t.