

ripALM: A Relative-Type Inexact Proximal Augmented Lagrangian Method with Applications to Quadratically Regularized Optimal Transport

Jiayi Zhu*, Ling Liang[†], Lei Yang[‡], Kim-Chuan Toh[§]

November 21, 2024

Abstract

Inexact proximal augmented Lagrangian methods (pALMs) are particularly appealing for tackling convex constrained optimization problems because of their elegant convergence properties and strong practical performance. To solve the associated pALM subproblems, efficient methods such as Newton-type methods are essential. Consequently, the effectiveness of the inexact pALM hinges on the error criteria used to control the inexactness when solving these subproblems. However, existing inexact pALMs either rely on absolute-type error criteria (which may complicate implementation by necessitating the pre-specification of an infinite sequence of error tolerance parameters) or require an additional correction step when using relative error criteria (which can potentially slow down the convergence of the pALM). To address this deficiency, this paper proposes ripALM, a relative-type inexact pALM, which can simplify practical implementation while preserving the appealing convergence properties of the classical absolute-type inexact pALM. We emphasize that ripALM is the first relative-type inexact version of the vanilla pALM with provable convergence guarantees. Numerical experiments on quadratically regularized optimal transport (OT) problems demonstrate the competitive efficiency of the proposed method compared to existing methods. As our analysis can be extended to a more general convex constrained problem setting, including other regularized OT problems, the proposed ripALM may provide broad applicability and has the potential to serve as a basic optimization tool.

Keywords: augmented Lagrangian method; proximal term; relative-type error criterion; asymptotically Q-(super)linear convergence rate; quadratically regularized optimal transport

1 Introduction

Constrained optimization, admitting excellent modeling power for real-world applications across a wide range of fields, including machine learning and data science, engineering,

The first two authors contributed equally.

*School of Computer Science and Engineering, Sun Yat-Sen University (zhuji86@mail2.sysu.edu.cn).

[†]Department of Mathematics, University of Maryland at College Park (liang.ling@u.nus.edu).

[‡](Corresponding author) School of Computer Science and Engineering, and Guangdong Province Key Laboratory of Computational Science, Sun Yat-Sen University (yanglei39@mail.sysu.edu.cn). The research of this author is supported in part by the National Natural Science Foundation of China under grant 12301411, and the Natural Science Foundation of Guangdong under grant 2023A1515012026.

[§]Department of Mathematics, and Institute of Operations Research and Analytics, National University of Singapore, Singapore 119076 (matttohkc@nus.edu.sg).

operations research, is a central area in optimization, especially in the current big-data era [6, 7]. This paper is dedicated to solving the following linearly constrained convex optimization problem:

$$\min_{\mathbf{x} \in \mathbb{R}^N} f(\mathbf{x}), \quad \text{s.t.} \quad A\mathbf{x} = \mathbf{b}, \quad (1.1)$$

where $f : \mathbb{R}^N \rightarrow \mathbb{R} \cup \{+\infty\}$ is a (possibly nonsmooth) proper closed convex function, $A \in \mathbb{R}^{M \times N}$ and $\mathbf{b} \in \mathbb{R}^M$ are given data. While we focus on (1.1) in this paper, we would like to mention that the inexact algorithmic framework along with the associated theory developed in this paper can be extended to a more general convex constrained optimization problem studied in [11].

The augmented Lagrangian method (ALM) is recognized as one of the most popular and effective methods for solving constrained optimization problems [16, 34]. The essential component of the ALM for solving problem (1.1) involves penalizing the linear constraint $A\mathbf{x} = \mathbf{b}$ to derive the augmented Lagrangian function, defined as

$$\mathcal{L}_\sigma^{\text{prim}}(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) + \langle \mathbf{y}, A\mathbf{x} - \mathbf{b} \rangle + \frac{\sigma}{2} \|A\mathbf{x} - \mathbf{b}\|^2, \quad (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^N \times \mathbb{R}^M,$$

where $\mathbf{y} \in \mathbb{R}^M$ is the Lagrangian multiplier associated with the linear constraint and $\sigma > 0$ is a penalty parameter. Then, for a given sequence of penalty parameters $\{\sigma_k\} \subseteq \mathbb{R}_{++}$ and an initial Lagrangian multiplier $\mathbf{y}^0 \in \mathbb{R}^M$, the ALM iteratively performs the following steps:

$$\begin{cases} \mathbf{x}^{k+1} \in \arg \min_{\mathbf{x} \in \mathbb{R}^N} \left\{ \mathcal{L}_{\sigma_k}^{\text{prim}}(\mathbf{x}, \mathbf{y}^k) \right\}, \\ \mathbf{y}^{k+1} = \mathbf{y}^k + \sigma_k \nabla_{\mathbf{y}} \mathcal{L}_{\sigma_k}^{\text{prim}}(\mathbf{x}^{k+1}, \mathbf{y}^k) = \mathbf{y}^k + \sigma_k (A\mathbf{x}^{k+1} - \mathbf{b}), \end{cases}$$

which consists of minimizing one ALM-subproblem to get \mathbf{x}^{k+1} and one gradient ascent step (associated with $\mathcal{L}_{\sigma_k}^{\text{prim}}$) with step size σ_k to get \mathbf{y}^{k+1} . Since the augmented Lagrangian function $\mathcal{L}_\sigma^{\text{prim}}(\cdot)$ is associated with the primal problem (1.1), the above ALM is usually named as a primal-based ALM. Alternatively, one may also consider a dual-based ALM which is associated with the dual problem of (1.1) given by (modulo a minus sign)

$$\min_{\mathbf{y} \in \mathbb{R}^M} f^*(A^\top \mathbf{y}) - \mathbf{b}^\top \mathbf{y}, \quad (1.2)$$

where $f^* : \mathbb{R}^N \rightarrow \mathbb{R} \cup \{+\infty\}$ denotes the conjugate function of f and A^\top is the transpose of A . Then, given a penalty parameter $\sigma > 0$, the augmented Lagrangian function associated with problem (1.2) is given by (see Section 2 for the detailed derivation)

$$\mathcal{L}_\sigma^{\text{dual}}(\mathbf{y}, \mathbf{x}) = -\mathbf{b}^\top \mathbf{y} + \frac{1}{2\sigma} \|\mathbf{x} + \sigma A^\top \mathbf{y}\|^2 - \frac{1}{2\sigma} \|\mathbf{x}\|^2 - \mathsf{M}_{\sigma f}(\mathbf{x} + \sigma A^\top \mathbf{y}), \quad (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^N \times \mathbb{R}^M,$$

where $\mathsf{M}_{\sigma f}(\mathbf{x}) := \min_{\mathbf{y}} \left\{ f(\mathbf{y}) + \frac{1}{2\sigma} \|\mathbf{y} - \mathbf{x}\|^2 \right\}$ denotes the Moreau envelope of the function $\sigma f(\cdot)$ at \mathbf{x} . Consequently, for a given sequence of penalty parameters $\{\sigma_k\} \subseteq \mathbb{R}_{++}$ and an initial point $\mathbf{x}^0 \in \mathbb{R}^N$, the dual-based ALM reads as follows:

$$\begin{cases} \mathbf{y}^{k+1} \in \arg \min_{\mathbf{y} \in \mathbb{R}^M} \left\{ \mathcal{L}_{\sigma_k}^{\text{dual}}(\mathbf{y}, \mathbf{x}^k) \right\}, \\ \mathbf{x}^{k+1} = \mathbf{x}^k + \sigma_k \nabla_{\mathbf{x}} \mathcal{L}_{\sigma_k}^{\text{dual}}(\mathbf{y}^{k+1}, \mathbf{x}^k) = \text{prox}_{\sigma_k f}(\mathbf{x}^k + \sigma_k A^\top \mathbf{y}^{k+1}). \end{cases}$$

The choice between using a primal-based or dual-based approach is problem-dependent. For example, when f is nonsmooth, but its proximal mapping is easy-to-compute, the dual-based ALM may be more preferable. This is because, in the dual-based ALM, the optimization problem for updating \mathbf{y} involves minimizing a continuously differentiable objective function, which can substantially simplify the optimization process.

The applicability of the ALM has been significantly expanded since Rockafellar’s seminal works [38, 39], which established a deep connection between the ALM and the proximal point algorithm (PPA). Moreover, the elegant convergence properties of the ALM can be established under the Lipschitz continuity of a certain solution mapping. Subsequent works have shown that this stringent condition can be further relaxed; see, for example, [8, 29] and references therein. Indeed, understanding the convergence properties of the ALM continues to be a topic of interest in the literature. Apart from theoretical advancements, numerous studies have demonstrated the remarkable efficiency of the ALM in solving various convex problems, including conic programming problems [20, 22, 24, 55], statistical optimization problems [19, 25, 53], optimization problems in machine learning [47, 48] and image/signal processing [18, 27], to mention just a few. These successful applications highlight the critical roles of both the solution methods used for solving ALM-subproblems and the error criteria that control the accuracy required for solving these subproblems, while preserving the appealing convergence properties of the ALM.

To solve ALM-subproblems efficiently, second-order methods, such as Newton-type methods [35], are strong candidates due to their fast local convergence rates. For example, once the iterate of a Newton-type method falls into the fast convergence region, only a few more iterations are needed to achieve a highly accurate solution. However, certain regularity conditions are needed to ensure the nonsingularity of the (generalized) Hessian of the ALM subproblem’s objective function at its solution point. Theoretically, such regularity conditions are generally not verifiable in advance, except in some special cases [24]. Numerically, ill-conditioned or singular (generalized) Hessians can lead to unpredictable numerical behaviors, resulting in unstable implementations and excessive efforts for parameter tuning. To enhance the applicability of a second-order method for solving ALM’s subproblems, a common simple approach is to add a proximal term to the ALM subproblem’s objective function, resulting in the proximal augmented Lagrangian method (pALM)¹. Specifically, given $\{\tau_k\} \subseteq \mathbb{R}_{++}$, $\{\sigma_k\} \subseteq \mathbb{R}_{++}$ and $(\mathbf{x}^0, \mathbf{y}^0) \in \mathbb{R}^N \times \mathbb{R}^M$, the dual-based pALM iteratively performs the following steps:

$$\begin{cases} \mathbf{y}^{k+1} = \arg \min_{\mathbf{y} \in \mathbb{R}^M} \left\{ \mathcal{L}_{\sigma_k}^{\text{dual}}(\mathbf{y}, \mathbf{x}^k) + \frac{\tau_k}{2\sigma_k} \|\mathbf{y} - \mathbf{y}^k\|^2 \right\}, \\ \mathbf{x}^{k+1} = \mathbf{x}^k + \sigma_k \nabla_{\mathbf{x}} \mathcal{L}_{\sigma_k}^{\text{dual}}(\mathbf{y}^{k+1}, \mathbf{x}^k) = \text{prox}_{\sigma_k f}(\mathbf{x}^k + \sigma_k A^\top \mathbf{y}^{k+1}). \end{cases} \quad (1.3)$$

By incorporating a proximal term, the pALM is able to bypass the regularity conditions when applying a second-order method for solving its subproblem. This typically results in more robust implementations with reduced parameter tuning. We refer readers to [9, 10, 15, 17, 21, 23, 25, 26, 33, 51] for successful applications of the pALM in addressing various important optimization problems.

To make the pALM truly implementable and practical, it must allow approximate solutions to the subproblem with progressively improved accuracy, and the associated error criterion must be practically verifiable, while preserving desirable convergence properties of the outer loop. To achieve this, an absolute-type error criterion was introduced by Rockafellar for the pALM in [38] and is now widely used in the literature; see, for example, [21, 23, 25, 26, 33]. However, this absolute-type error criterion requires the specification of a summable sequence of tolerance parameters, which must be tuned to avoid being too conservative or too aggressive. As a result, careful parameter tuning of this sequence is often necessary to achieve superior convergence performance. This process can result in excessive effort and potential inefficiencies in practical implementations. A notable alternative

¹The pALM is also known as the proximal method of multipliers; see [38].

is to adopt a relative-type error criterion, which seeks to control the errors in minimizing the subproblems based on some quantity related to the progress of the algorithm. This approach eliminates the need for a summable tolerance sequence, thereby enhancing practical implementability. The adoption of a relative-type error criterion started from the seminal works of Solodov and Svaiter [44, 42, 43, 45, 46] on inexact versions of the PPA, and has since influenced the development of inexact versions of numerous algorithms, including the variable metric PPA [30], ALM [11], ADMM [12], FISTA [3], and the Bregman-type methods [49, 52]. However, to the best of our knowledge, research on relative-type inexact versions of the pALM remains in its early stage, with limited exploration in [10, 17, 51]. In particular, all these inexact versions are derived by applying a (variable metric) hybrid proximal extragradient method [30, 42, 43] to a certain primal-dual solution mapping of the convex constrained problem, and therefore require an additional correction step to ensure convergence. Consequently, the algorithms developed in [10, 17, 51] essentially deviate from the vanilla pALM framework (1.3). Moreover, our numerical results also indicate that the correction step tends to slow down the convergence of the pALM and thus increase the computational cost. These observations naturally lead us to raise the following question:

*Can we design a relative-type error criterion for the **vanilla** pALM, while preserving desirable convergence properties?*

In light of this, the primary goal of this paper is to address the above question by designing a relative-type error criterion for the *vanilla* pALM (1.3). Note that the relative-type error criterion proposed by Eckstein and Silva [11] for the ALM innovatively introduces an auxiliary error variable to control the inexactness while ensuring convergence. In this work, we shall demonstrate that a similar idea can be adapted to the vanilla pALM, but with essential modifications. We should emphasize that the corresponding theoretical convergence analysis also requires substantial modifications, due to the inclusion of the proximal term. Interestingly, while the initial motivation for considering pALM was to ensure the nonsingularity of the (generalized) Hessian of the ALM subproblem’s objective function at its solution point, our analysis reveals that this proximal term can further improve the existing convergence properties of the relative-type inexact ALM studied in [1, 11, 54]. Specifically, under some common assumptions, it ensures the convergence of both the primal and dual sequences, as well as their asymptotically Q-(super)linear convergence rates. The key contributions and findings of this paper are summarized as follows:

- We develop a relative-type inexact proximal augmented Lagrangian method (ripALM) to solve the dual problem (1.2). This proposed ripALM is the first inexact version of the vanilla pALM (1.3). It not only shares the same convergence properties for the sequence $\{\mathbf{x}^k\}$ as the relative-type inexact ALM studied in [1, 11, 54], but also offers other theoretical advantages in that the sequence $\{\mathbf{y}^k\}$ is guaranteed to converge, and both primal and dual sequences achieve asymptotically Q-(super)linear convergence rates under a relatively weaker error bound assumption. These results provide strong theoretical support for the applicability of ripALM, particularly in scenarios where accurate solutions of both primal and dual problems are required.
- Numerically, we implement the proposed ripALM and apply it to solve quadratically regularized optimal transport (QROT) problems. Extensive numerical results validate the promising performance of the proposed ripALM for solving large-scale QROT problems. Moreover, comparisons with existing methods underscore our motivation and contributions in developing a relative-type error criterion for the vanilla pALM. We

also emphasize that our ripALM can be extended to solve optimal transport problems with other convex regularizers such as the group quadratic regularizer [51], provided that the regularizers are proximal-friendly and the generalized Jacobians associated with the pALM-subproblems are easy to compute. Given the growing interests in developing efficient algorithmic frameworks for computational optimal transport [31], our work may open up new possibilities for applying the powerful pALM algorithmic framework in a wide variety of real-world applications.

The remaining parts of this paper are organized as follows. Section 2 describes the main algorithmic framework of ripALM, whose convergence analysis is conducted in Section 3. Section 4 showcases how to apply the proposed ripALM for solving QROT problems. Numerical experiments are conducted in Section 5. Finally, some concluding remarks are summarized in Section 6.

Notation. We use \mathbb{R}^n , \mathbb{R}_+^n , $\mathbb{R}^{m \times n}$ and $\mathbb{R}_+^{m \times n}$ to denote the sets of n -dimensional real vectors, n -dimensional non-negative vectors, $m \times n$ real matrices and $m \times n$ real non-negative matrices, respectively. For a vector $\mathbf{x} \in \mathbb{R}^n$, x_i denotes its i -th entry, $\|\mathbf{x}\|$ denotes its Euclidean norm, and $\|\mathbf{x}\|_H := \sqrt{\langle \mathbf{x}, H\mathbf{x} \rangle}$ denotes its weighted norm associated with a symmetric positive definite matrix $H \in \mathbb{R}^{n \times n}$. For a matrix $X \in \mathbb{R}^{m \times n}$, X_{ij} denotes its (i, j) -th entry, $\|X\|_F$ denotes its Frobenius norm, and $\text{vec}(X)$ denotes the vectorization of X , where $[\text{vec}(X)]_{i+(j-1)m} = X_{ij}$ for any $1 \leq i \leq m$ and $1 \leq j \leq n$. For simplicity, given an integer $n > 0$, we use $\mathbf{1}_n \in \mathbb{R}^n$ to denote the n -dimensional vector of all ones, and use I_n to denote the $n \times n$ identity matrix.

For an extended-real-valued function $f : \mathbb{R}^n \rightarrow [-\infty, \infty]$, we say that it is *proper* if $f(\mathbf{x}) > -\infty$ for all $\mathbf{x} \in \mathbb{R}^n$ and its effective domain $\text{dom } f := \{\mathbf{x} \in \mathbb{R}^n : f(\mathbf{x}) < \infty\}$ is nonempty. A proper function f is said to be closed if it is lower semicontinuous. Assume that $f : \mathbb{R}^n \rightarrow (-\infty, \infty]$ is a proper and closed convex function, the subdifferential of f at $\mathbf{x} \in \text{dom } f$ is defined by $\partial f(\mathbf{x}) := \{\mathbf{d} \in \mathbb{R}^n : f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \mathbf{d}, \mathbf{y} - \mathbf{x} \rangle, \forall \mathbf{y} \in \mathbb{R}^n\}$ and its conjugate function $f^* : \mathbb{R}^n \rightarrow (-\infty, \infty]$ is defined by $f^*(\mathbf{y}) := \sup \{\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^n\}$. For any $\nu > 0$, the Moreau envelope of νf at \mathbf{x} is defined by $M_{\nu f}(\mathbf{x}) := \min_{\mathbf{y}} \{f(\mathbf{y}) + \frac{1}{2\nu} \|\mathbf{y} - \mathbf{x}\|^2\}$, and the proximal mapping of νf at \mathbf{x} is defined by $\text{prox}_{\nu f}(\mathbf{x}) := \arg \min_{\mathbf{y}} \{f(\mathbf{y}) + \frac{1}{2\nu} \|\mathbf{y} - \mathbf{x}\|^2\}$.

Let \mathcal{S} be a closed convex subset of \mathbb{R}^n . Its indicator function $\delta_{\mathcal{S}}$ is defined by $\delta_{\mathcal{S}}(\mathbf{x}) = 0$ if $\mathbf{x} \in \mathcal{S}$ and $\delta_{\mathcal{S}}(\mathbf{x}) = +\infty$ otherwise. Moreover, we denote the weighted distance of $\mathbf{x} \in \mathbb{R}^n$ to \mathcal{S} by $\text{dist}_H(\mathbf{x}, \mathcal{S}) := \inf_{\mathbf{y} \in \mathcal{S}} \|\mathbf{x} - \mathbf{y}\|_H$ associated with a symmetric positive definite matrix H . When H is the identity matrix, we omit H in the notation and simply use $\text{dist}(\mathbf{x}, \mathcal{S})$ to denote the Euclidean distance of $\mathbf{x} \in \mathbb{R}^n$ to \mathcal{S} . Moreover, we use $\Pi_{\mathcal{S}}(\mathbf{x})$ to denote the projection of \mathbf{x} onto \mathcal{S} .

2 A relative-type inexact pALM

In this section, we focus on developing a relative-type inexact proximal augmented Lagrangian method (ripALM) to solve the dual problem (1.2). The algorithmic framework is developed based on the parametric convex duality framework (see, for example, [36, 37] and [40, Chapter 11]).

We first identify problem (1.2) with the following problem

$$\min_{\mathbf{y} \in \mathbb{R}^M} G(\mathbf{y}, \mathbf{0}), \quad (2.1)$$

where $G : \mathbb{R}^M \times \mathbb{R}^N \rightarrow (-\infty, +\infty]$ is defined by

$$G(\mathbf{y}, \boldsymbol{\xi}) := f^*(A^\top \mathbf{y} + \boldsymbol{\xi}) - \mathbf{b}^\top \mathbf{y}. \quad (2.2)$$

Note that G is proper closed convex since f^* is proper closed convex. Then, the (ordinary) Lagrangian function of problem (1.2) can be defined by taking the concave conjugate of G with respect to its second argument (see [40, Definition 11.45]), that is,

$$\ell(\mathbf{y}, \mathbf{x}) := \inf_{\boldsymbol{\xi} \in \mathbb{R}^N} \{G(\mathbf{y}, \boldsymbol{\xi}) - \langle \mathbf{x}, \boldsymbol{\xi} \rangle\} = -\mathbf{b}^\top \mathbf{y} + \langle \mathbf{x}, A^\top \mathbf{y} \rangle - f(\mathbf{x}). \quad (2.3)$$

Clearly, ℓ is convex in its first argument and concave in the second argument. For a given penalty parameter $\sigma > 0$, the augmented Lagrangian function of problem (1.2) is defined by (see [40, Example 11.57])

$$\begin{aligned} \mathcal{L}_\sigma(\mathbf{y}, \mathbf{x}) &:= \sup_{\mathbf{s} \in \mathbb{R}^N} \left\{ \ell(\mathbf{y}, \mathbf{s}) - \frac{1}{2\sigma} \|\mathbf{s} - \mathbf{x}\|^2 \right\} \\ &= -\mathbf{b}^\top \mathbf{y} + \frac{1}{2\sigma} \|\mathbf{x} + \sigma A^\top \mathbf{y}\|^2 - \frac{1}{2\sigma} \|\mathbf{x}\|^2 - M_{\sigma f}(\mathbf{x} + \sigma A^\top \mathbf{y}). \end{aligned}$$

From the property of the Moreau envelope (see [2, Proposition 12.30]), we know that \mathcal{L}_σ is continuously differentiable with respect to its first argument and

$$\nabla_{\mathbf{y}} \mathcal{L}_\sigma(\mathbf{y}, \mathbf{x}) = \text{Aprox}_{\sigma f}(\mathbf{x} + \sigma A^\top \mathbf{y}) - \mathbf{b}.$$

With the above preparations, we are ready to present our ripALM for solving problem (1.2) in Algorithm 1.

Algorithm 1 A relative-type inexact proximal augmented Lagrangian method (ripALM) for solving problem (1.2)

Input: $\rho \in [0, 1)$, $\{\sigma_k\}_{k=0}^\infty$ is a sequence of real numbers such that $\inf_{k \geq 0} \{\sigma_k\} > 0$, and $\{\tau_k\}_{k=0}^\infty$ is a sequence of real numbers such that $\inf_{k \geq 0} \{\tau_k\} > 0$ and $\sup_{k \geq 0} \{\tau_k\} < \infty$. Choose $\mathbf{y}^0, \mathbf{w}^0 \in \mathbb{R}^M$ and $\mathbf{x}^0 \in \mathbb{R}^N$ arbitrarily. Set $k = 0$.

while the termination criterion is not met, **do**

Step 1. Approximately solve the subproblem:

$$\min_{\mathbf{y} \in \mathbb{R}^M} \mathcal{L}_{\sigma_k}(\mathbf{y}, \mathbf{x}^k) + \frac{\tau_k}{2\sigma_k} \|\mathbf{y} - \mathbf{y}^k\|^2 \quad (2.4)$$

to find $\mathbf{y}^{k+1} \in \mathbb{R}^M$ such that

$$\begin{aligned} &2|\langle \mathbf{w}^k - \mathbf{y}^{k+1}, \sigma_k \Delta^{k+1} \rangle| + \|\sigma_k \Delta^{k+1}\|^2 \\ &\leq \rho \left(\|\text{prox}_{\sigma_k f}(\mathbf{x}^k + \sigma_k A^\top \mathbf{y}^{k+1}) - \mathbf{x}^k\|^2 + \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2 \right), \end{aligned} \quad (2.5)$$

where $\Delta^{k+1} := \nabla_{\mathbf{y}} \mathcal{L}_{\sigma_k}(\mathbf{y}^{k+1}, \mathbf{x}^k) + \tau_k \sigma_k^{-1} (\mathbf{y}^{k+1} - \mathbf{y}^k)$.

Step 2. Update

$$\mathbf{x}^{k+1} = \text{prox}_{\sigma_k f}(\mathbf{x}^k + \sigma_k A^\top \mathbf{y}^{k+1}), \quad \mathbf{w}^{k+1} = \mathbf{w}^k - \sigma_k \Delta^{k+1}. \quad (2.6)$$

Step 3. Set $k = k + 1$ and go to **Step 1**.

end while

Output: $(\mathbf{y}^k, \mathbf{x}^k)$

In line with pALM-type methods, at each iteration, our ripALM in Algorithm 1 *approximately* minimizes the sum of the augmented Lagrangian function $\mathcal{L}_{\sigma_k}(\cdot, \mathbf{x}^k)$ and a proximal

term $\frac{\tau_k}{2\sigma_k} \|\cdot - \mathbf{y}^k\|^2$ under the error criterion (2.5), followed by the updates of the multiplier \mathbf{x}^k and the auxiliary error variable \mathbf{w}^k . Due to the strong convexity of the objective function in (2.4), the subproblem (2.4) has a unique solution, denoted as $\mathbf{y}^{k,*}$, satisfying $\Delta^{k,*} := \nabla_y \mathcal{L}_{\sigma_k}(\mathbf{y}^{k,*}, \mathbf{x}^k) + \tau_k \sigma_k^{-1}(\mathbf{y}^{k,*} - \mathbf{y}^k) = \mathbf{0}$. Then, for any minimizing sequence $\{\mathbf{y}^{k,t}\}$ for (2.4) converging to $\mathbf{y}^{k,*}$, we have that $\Delta^{k,t} := \nabla_y \mathcal{L}_{\sigma_k}(\mathbf{y}^{k,t}, \mathbf{x}^k) + \tau_k \sigma_k^{-1}(\mathbf{y}^{k,t} - \mathbf{y}^k) \rightarrow \mathbf{0}$ and consequently $2|\langle \mathbf{w}^k - \mathbf{y}^{k,t}, \sigma_k \Delta^{k,t} \rangle| + \|\sigma_k \Delta^{k,t}\|^2 \rightarrow 0$ as $t \rightarrow \infty$. Meanwhile, if \mathbf{y}^k is not the solution of (2.4) and $\tau_k \neq 0$, the right-hand side of (2.5) cannot be zero. Therefore, in this case, our error criterion (2.5) must hold after finitely many t iterations, and thus it is achievable.

Compared to the recent semismooth Newton based inexact proximal augmented Lagrangian (SNIPAL) method developed in [21, Section 3], our ripALM in Algorithm 1 employs a significantly different error criterion (2.5) for solving the subproblem (2.4). Specifically, in our context, SNIPAL requires the error term Δ^{k+1} to satisfy

$$\begin{aligned} (A) \quad \|\Delta^{k+1}\| &\leq \frac{\min(\sqrt{\tau_k}, 1)}{\sigma_k} \varepsilon_k, \quad \varepsilon_k \geq 0, \quad \sum_{k=0}^{\infty} \varepsilon_k < \infty, \\ (B) \quad \|\Delta^{k+1}\| &\leq \frac{\delta_k \min(\sqrt{\tau_k}, 1)}{\sigma_k} \sqrt{\|\Delta_x^{k+1}\|^2 + \tau_k \|\Delta_y^{k+1}\|^2}, \quad 0 \leq \delta_k < 1, \quad \sum_{k=0}^{\infty} \delta_k < \infty, \end{aligned} \quad (2.7)$$

with $\Delta_x^{k+1} := \mathbf{x}^{k+1} - \mathbf{x}^k$ and $\Delta_y^{k+1} := \mathbf{y}^{k+1} - \mathbf{y}^k$, to guarantee an asymptotically Q-(super)linear convergence rate.² Both error criteria (A) and (B) in (2.7) are of the absolute type, meaning that they require the pre-specification of two summable sequences of tolerance parameters, $\{\varepsilon_k\} \subseteq [0, \infty)$ and $\{\delta_k\} \subseteq [0, 1)$, to control the error incurred in the inexact minimization of the subproblem. Since there is generally no direct guidance on optimally selecting these tolerance parameters to achieve good convergence efficiency, this absolute-type criterion typically requires careful tuning for the tolerance parameters, which may make SNIPAL less user-friendly in practical implementations. In contrast, the error criterion (2.5) used in ripALM is of the relative type, meaning that the error $2|\langle \mathbf{w}^k - \mathbf{y}^{k+1}, \sigma_k \Delta^{k+1} \rangle| + \|\sigma_k \Delta^{k+1}\|^2$ is regulated by a tentative successive difference related to the progress of the algorithm. Moreover, this relative-type criterion only involves a *single* tolerance parameter $\rho \in [0, 1)$, thus simplifying the process of parameter tuning both computationally and in terms of implementation, as we shall see in subsection 5.1.

Thanks to the advantage of eliminating the need to select an infinite sequence of tolerance parameters, various versions of relative error criteria have been widely adopted in numerous well-known algorithms (e.g., PPA, ALM, ADMM, FISTA, etc) to approximately solve subproblems over the past two decades. This trend began with the seminal works of Solodov and Svaiter [42, 43, 44, 45, 46], which has since inspired the development of numerous algorithms; see, for example, [3, 11, 12, 30, 49, 51, 52]. Following the same research theme, Yang et al. [51] recently developed a corrected inexact proximal augmented Lagrangian method (ciPALM) for solving a class of group-quadratic regularized optimal transport problems. The error criterion used there can be described as follows: choose $\rho \in [0, 1)$, at each iteration, approximately solve the subproblem (2.4) to find a point $\tilde{\mathbf{y}}^{k+1}$ such that

$$\begin{aligned} \tilde{\Delta}^{k+1} &:= \nabla_y \mathcal{L}_{\sigma_k}(\tilde{\mathbf{y}}^{k+1}, \mathbf{x}^k) + \tau_k \sigma_k^{-1}(\tilde{\mathbf{y}}^{k+1} - \mathbf{y}^k), \\ \|\sigma_k \tilde{\Delta}^{k+1}\|^2 &\leq \rho \min(\tau_k, 1) \left\| \text{prox}_{\sigma_k f}(\mathbf{x}^k + \sigma_k A^\top \tilde{\mathbf{y}}^{k+1}) - \mathbf{x}^k \right\|^2 + \tau_k \left\| \tilde{\mathbf{y}}^{k+1} - \mathbf{y}^k \right\|^2. \end{aligned} \quad (2.8)$$

²Note that the error criterion (A) in (2.7) alone is sufficient for establishing the global convergence of the SNIPAL; see [21, Section 3].

When (2.8) is satisfied, the multiplier is updated as usual: $\mathbf{x}^{k+1} = \text{prox}_{\sigma_k f}(\mathbf{x}^k + \sigma_k A^\top \tilde{\mathbf{y}}^{k+1})$. This is then followed by an extra correction step: $\mathbf{y}^{k+1} = \mathbf{y}^k - \tau_k^{-1} \sigma_k (A\mathbf{x}^{k+1} - \mathbf{b})$. After these updates, the algorithm proceeds to the next iteration. One can see that the error criterion (2.8) used in the ciPALM is also of the relative type and looks even simpler than (2.5) used in our ripALM. However, we should point out that the ciPALM requires an extra correction step to guarantee the convergence, and hence it may only be viewed as a pALM-like algorithm. In contrast, our relative error criterion (2.5) is designed for the vanilla pALM, where the error variable \mathbf{w}^k is used solely in the construction of the error criterion (2.5) and does not directly influence either the objective function in (2.4) or the updating rules of the primal and dual variables. Our experimental results also show that the correction step tends to slow down the convergence of the pALM, whereas our ripALM can offer greater robustness and efficiency; see subsection 5.1 for numerical comparisons.

Our relative error criterion (2.5) is inspired by Eckstein and Silva's practical relative error criterion [11], which was developed for the approximate minimization of the subproblems in the vanilla ALM (i.e., without the proximal term $\frac{\tau_k}{2\sigma_k} \|\mathbf{y} - \mathbf{y}^k\|^2$ in the subproblem (2.4) in our context). Unlike their work, we suggest incorporating $\frac{\tau_k}{2\sigma_k} \|\mathbf{y} - \mathbf{y}^k\|^2$ in the subproblem (2.4). This proximal term guarantees the existence and uniqueness of the optimal solution of the strongly convex subproblem (2.4). Introducing the proximal term also ensures the positive definiteness of the generalized Hessian of the objective function in (2.4), thereby facilitating the application of the semi-smooth Newton method to solve the subproblem (2.4) effectively, as shown in Section 4. In addition, as we will see later in the next section, our ripALM not only shares the same convergence properties for the sequence $\{\mathbf{x}^k\}$ as Eckstein and Silva's relative-type inexact ALM developed in [1, 11, 54], but also offers other theoretical advantages in that the sequence $\{\mathbf{y}^k\}$ is guaranteed to converge, and both primal and dual sequences achieve asymptotically Q-(super)linear convergence rates under a relatively weaker error bound assumption.

3 Convergence analysis

In this section, we study the convergence properties of ripALM in Algorithm 1. Before proceeding, we recall the definition of G from (2.2) and further define $F : \mathbb{R}^N \times \mathbb{R}^M \rightrightarrows \mathbb{R}^N \times \mathbb{R}^M$ to be the concave conjugate of G :

$$F(\boldsymbol{\theta}, \mathbf{x}) := \inf_{\mathbf{y} \in \mathbb{R}^M, \boldsymbol{\xi} \in \mathbb{R}^N} \{G(\mathbf{y}, \boldsymbol{\xi}) - \langle \boldsymbol{\theta}, \mathbf{y} \rangle - \langle \mathbf{x}, \boldsymbol{\xi} \rangle\}, \quad (3.1)$$

which is a closed (upper semicontinuous) concave function. Then, the dual problem of (2.1) is given by

$$\max_{\mathbf{x} \in \mathbb{R}^N} F(\mathbf{0}, \mathbf{x}), \quad (3.2)$$

which can be rewritten as problem (1.1). Next, let $\partial G : \mathbb{R}^M \times \mathbb{R}^N \rightrightarrows \mathbb{R}^M \times \mathbb{R}^N$ and $\partial F : \mathbb{R}^M \times \mathbb{R}^N \rightrightarrows \mathbb{R}^M \times \mathbb{R}^N$ denote the subgradient maps of G and F , respectively, that is,

$$\begin{aligned} (\boldsymbol{\theta}, \mathbf{x}) \in \partial G(\mathbf{y}, \boldsymbol{\xi}) &\Leftrightarrow G(\mathbf{y}', \boldsymbol{\xi}') \geq G(\mathbf{y}, \boldsymbol{\xi}) + \langle \boldsymbol{\theta}, \mathbf{y}' - \mathbf{y} \rangle + \langle \mathbf{x}, \boldsymbol{\xi}' - \boldsymbol{\xi} \rangle, \quad \forall (\mathbf{y}', \boldsymbol{\xi}'), \\ (\mathbf{y}, \boldsymbol{\xi}) \in \partial F(\boldsymbol{\theta}, \mathbf{x}) &\Leftrightarrow F(\boldsymbol{\theta}', \mathbf{x}') \leq F(\boldsymbol{\theta}, \mathbf{x}) - \langle \mathbf{y}, \boldsymbol{\theta}' - \boldsymbol{\theta} \rangle - \langle \boldsymbol{\xi}, \mathbf{x}' - \mathbf{x} \rangle, \quad \forall (\boldsymbol{\theta}', \mathbf{x}'). \end{aligned}$$

We also recall the definition of ℓ from (2.3) and let $\partial \ell : \mathbb{R}^M \times \mathbb{R}^N \rightrightarrows \mathbb{R}^M \times \mathbb{R}^N$ denote the subgradient map of ℓ , that is,

$$(\boldsymbol{\theta}, \boldsymbol{\xi}) \in \partial \ell(\mathbf{y}, \mathbf{x}) \Leftrightarrow \begin{cases} \ell(\mathbf{y}', \mathbf{x}) \geq \ell(\mathbf{y}, \mathbf{x}) + \langle \boldsymbol{\theta}, \mathbf{y}' - \mathbf{y} \rangle, & \forall \mathbf{y}' \in \mathbb{R}^M, \\ \ell(\mathbf{y}, \mathbf{x}') \leq \ell(\mathbf{y}, \mathbf{x}) - \langle \boldsymbol{\xi}, \mathbf{x}' - \mathbf{x} \rangle, & \forall \mathbf{x}' \in \mathbb{R}^N. \end{cases}$$

By this definition, one can verify that

$$\partial\ell(\mathbf{y}, \mathbf{x}) := \{A\mathbf{x} - \mathbf{b}\} \times \left\{v - A^\top \mathbf{y} \mid v \in \partial f(\mathbf{x})\right\}.$$

Moreover, all the subgradient maps ∂G , ∂F and $\partial\ell$ are maximal monotone operators, and satisfy that (see also [11, equation (23)]):

$$(\boldsymbol{\theta}, \mathbf{x}) \in \partial G(\mathbf{y}, \boldsymbol{\xi}) \Leftrightarrow (\boldsymbol{\theta}, \boldsymbol{\xi}) \in \partial\ell(\mathbf{y}, \mathbf{x}) \Leftrightarrow (\mathbf{y}, \boldsymbol{\xi}) \in \partial F(\boldsymbol{\theta}, \mathbf{x}). \quad (3.3)$$

If $(\mathbf{y}^*, \mathbf{x}^*) \in \mathbb{R}^M \times \mathbb{R}^N$ satisfies $(\mathbf{0}, \mathbf{0}) \in \partial\ell(\mathbf{y}^*, \mathbf{x}^*)$, then \mathbf{y}^* solves problem (2.1) (i.e., problem (1.2)) and \mathbf{x}^* solves problem (3.2) (i.e., problem (1.1)). In this case, we call $(\mathbf{y}^*, \mathbf{x}^*)$ a *saddle point* of the Lagrangian function $\ell(\mathbf{y}, \mathbf{x})$. If such a saddle point exists, then strong duality holds, that is, the optimal values of problems (2.1) and (3.2) are finite and equal, i.e., $G(\mathbf{y}^*, \mathbf{0}) = F(\mathbf{0}, \mathbf{x}^*)$. Moreover, the set of saddle points can be written as $\mathcal{Y}^* \times \mathcal{X}^* \subseteq \mathbb{R}^M \times \mathbb{R}^N$, where \mathcal{Y}^* is the solution set of problem (2.1) (i.e., problem (1.2)) and \mathcal{X}^* is the solution set of problem (3.2) (i.e., problem (1.1)).

With the above preparations, we are now ready to establish the convergence results of the proposed ripALM in Algorithm 1. The analysis is inspired by [1, 11, 54], but is more involved. We need to be particularly careful in establishing a proper recursive relation with respect to \mathbf{x}^k , \mathbf{y}^k and \mathbf{w}^k , due to the inclusion of the proximal term $\frac{\tau_k}{2\sigma_k} \|\mathbf{y} - \mathbf{y}^k\|^2$.

Theorem 3.1. *Let the functions G , F and ℓ be defined as in (2.2), (3.1) and (2.3), respectively. Let $\rho \in [0, 1)$, $\{\sigma_k\} \subset \mathbb{R}_{++}$ be a positive sequence satisfying that $\sigma_k \geq \sigma_{\min} > 0$ for all $k \geq 0$, and $\{\tau_k\}$ be a positive sequence satisfying that*

$$\tau_k \geq \tau_{\min} > 0, \quad \tau_{k+1} \leq (1 + \nu_k)\tau_k \quad \text{with } \nu_k \geq 0 \quad \text{and} \quad \sum_{k=0}^{\infty} \nu_k < +\infty.$$

Let $\{\mathbf{y}^k\}$, $\{\Delta^k\}$, $\{\mathbf{w}^k\} \subset \mathbb{R}^M$ and $\{\mathbf{x}^k\} \subset \mathbb{R}^N$ be sequences generated by Algorithm 1. If ℓ admits a saddle point (i.e., $(\partial\ell)^{-1}(\mathbf{0}, \mathbf{0}) \neq \emptyset$), then the following statements hold.

(i) *The sequences $\{\mathbf{y}^k\}$, $\{\mathbf{w}^k\}$ and $\{\mathbf{x}^k\}$ are bounded.*

(ii) *$\lim_{k \rightarrow \infty} \Delta^{k+1} = \mathbf{0}$, $\lim_{k \rightarrow \infty} \boldsymbol{\theta}^{k+1} = \mathbf{0}$ and $\lim_{k \rightarrow \infty} \boldsymbol{\xi}^{k+1} = \mathbf{0}$, where $\boldsymbol{\theta}^{k+1}$ and $\boldsymbol{\xi}^{k+1}$ are defined by*

$$\boldsymbol{\theta}^{k+1} := \Delta^{k+1} - \tau_k \sigma_k^{-1} (\mathbf{y}^{k+1} - \mathbf{y}^k) \quad \text{and} \quad \boldsymbol{\xi}^{k+1} := \sigma_k^{-1} (\mathbf{x}^k - \mathbf{x}^{k+1}), \quad \forall k \geq 0.$$

(iii) *Both the sequences $\{G(\mathbf{y}^{k+1}, \boldsymbol{\xi}^{k+1})\}$ and $\{F(\boldsymbol{\theta}^{k+1}, \mathbf{x}^{k+1})\}$ converge to the common optimal value of problems (2.1) and (3.2).*

(iv) *Any accumulation point of $\{\mathbf{y}^k\}$ is an optimal solution of problem (2.1) (i.e., problem (1.2)), and any accumulation point of $\{\mathbf{x}^k\}$ is an optimal solution of problem (3.2) (i.e., problem (1.1)).*

(v) *The sequence $\{\mathbf{x}^k\}$ converges to an optimal solution of problem (3.2).*

Proof. See Appendix A.1. \square

We next study the asymptotically Q-(super)linear convergence rate of our ripALM under an error-bound condition presented in Assumption A. As noted in [21, Lemma 2.4], this error-bound condition is weaker than the local upper Lipschitz continuity of $(\partial\ell)^{-1}$ at the origin. The latter condition was used in [54] to establish the asymptotic Q-(super)linear convergence rate for Eckstein and Silva's relative-type inexact ALM, while the former, weaker condition has also been employed in [21] and [51] to establish the asymptotic Q-(super)linear convergence rate for the SNIPAL and ciPALM, respectively.

Assumption A. For any $r > 0$, there exist a constant $\kappa > 0$ such that, for any $(\mathbf{y}, \mathbf{x}) \in \{(\mathbf{y}, \mathbf{x}) \in \mathbb{R}^M \times \mathbb{R}^N \mid \text{dist}((\mathbf{y}, \mathbf{x}), (\partial\ell)^{-1}(\mathbf{0}, \mathbf{0})) \leq r\}$,

$$\text{dist}((\mathbf{y}, \mathbf{x}), (\partial\ell)^{-1}(\mathbf{0}, \mathbf{0})) \leq \kappa \text{dist}((\mathbf{0}, \mathbf{0}), \partial\ell(\mathbf{y}, \mathbf{x})). \quad (3.4)$$

Theorem 3.2. Under the same assumptions as in Theorem 3.1, suppose additionally that Assumption A holds, and the sequences of parameters ρ , $\{\sigma_k\}$ and $\{\tau_k\}$ satisfy that

$$\sqrt{\tau_{\min}} - 2\sqrt{\rho} > 0 \quad \text{and} \quad \liminf_{k \rightarrow \infty} \sigma_k > c \cdot \frac{2\kappa\sqrt{\tau_{\max}} \left(\rho + \sqrt{\rho \max\{1, \tau_{\max}\}} \right)}{\sqrt{\tau_{\min}} - 2\sqrt{\rho}}, \quad (3.5)$$

where $c > 1$ is an arbitrarily given positive constant and $\tau_{\max} := \tau_0 \prod_{k=0}^{\infty} (1 + \nu_k)$. Let $\Lambda^k := \text{Diag}(\tau_k I_M, I_N)$, $\bar{\tau}_k := \max\{1, \tau_k\}$ and

$$\gamma_k := \left(1 - \frac{2\kappa\sqrt{\tau_k} (\rho + \sqrt{\rho\bar{\tau}_k}) + 2\sigma_k\sqrt{\rho}}{\sigma_k\sqrt{\tau_k}} \right) \frac{\sigma_k^2}{\kappa^2 (\sqrt{\rho} + \sqrt{\bar{\tau}_k})^2 \bar{\tau}_k}.$$

Then, the following statements hold.

(i) For any sufficiently large k , we have that $\gamma_k > 0$ and

$$\text{dist}_{\Lambda^{k+1}} \left((\mathbf{y}^{k+1}, \mathbf{x}^{k+1}), (\partial\ell)^{-1}(\mathbf{0}, \mathbf{0}) \right) \leq \mu_k \text{dist}_{\Lambda^k} \left((\mathbf{y}^k, \mathbf{x}^k), (\partial\ell)^{-1}(\mathbf{0}, \mathbf{0}) \right),$$

where

$$\limsup_{k \rightarrow \infty} \left\{ \mu_k := \sqrt{\frac{1 + \nu_k}{1 + \gamma_k}} \right\} < 1.$$

(ii) The sequence $\{\mathbf{y}^k\}$ is convergent.

Proof. See Appendix A.2. \square

Remark 3.1 (Comments on μ_k). One can see from the expression of γ_k that, after a finite number of iterations, γ_k becomes proportional to the penalty parameter σ_k , provided that ρ , $\{\sigma_k\}$ and $\{\tau_k\}$ satisfy conditions in (3.5). Therefore, if $\sqrt{\tau_{\min}} - 2\sqrt{\rho} > 0$, the coefficient μ_k can be less than 1 when σ_k is sufficiently large, and it approaches 0 as σ_k tends to $+\infty$. This readily demonstrates that the sequence $\{(\mathbf{y}^k, \mathbf{x}^k)\}$ converges to the set of saddle points at a Q -(super)linear rate if σ_k is sufficiently large and $\sqrt{\tau_{\min}} - 2\sqrt{\rho} > 0$. In practical implementations, one could simply choose an increasing sequence of $\{\sigma_k\}$ with $\sigma_k \uparrow +\infty$, and set $\tau_{\min} \geq 4$ to ensure that $\sqrt{\tau_{\min}} - 2\sqrt{\rho} > 0$ for any $\rho \in [0, 1)$. In contrast, as discussed in [51, Remark 2.1], the ciPALM (which is also of relative-type but uses the error criterion (2.8)) should require $\rho < \frac{1}{3}$ to guarantee an asymptotically Q -(super)linear convergence rate under the same error bound condition. This highlights another advantage of our ripALM, as it offers greater flexibility in choosing ρ .

4 Application to the QROT problem

In this section, we will present the application of ripALM in Algorithm 1 for solving quadratically regularized optimal transport (QROT) problems. As a significant variant of the classical optimal transport problem, the QROT problem has garnered notable attention in recent years; see, e.g., [4, 13, 28] for more details. Mathematically, the QROT problem is given as follows:

$$\min_X \left\{ \frac{\lambda}{2} \|X\|_F^2 + \langle C, X \rangle \mid X\mathbf{1}_n = \boldsymbol{\alpha}, \quad X^\top \mathbf{1}_m = \boldsymbol{\beta}, \quad X \geq 0 \right\}, \quad (4.1)$$

where $\lambda > 0$ is a given regularization parameter, $C \in \mathbb{R}_+^{m \times n}$ is a given cost matrix, $\boldsymbol{\alpha} := (\alpha_1, \dots, \alpha_m)^\top \in \Sigma_m$ and $\boldsymbol{\beta} := (\beta_1, \dots, \beta_n)^\top \in \Sigma_n$ are given probability vectors with Σ_m (resp., Σ_n) denoting the m (resp., n)-dimensional unit simplex, and $\mathbf{1}_m$ (resp., $\mathbf{1}_n$) denotes the m (resp., n)-dimensional vector of all ones.

To apply our proposed ripALM, we first reformulate (4.1) as follows:

$$\min_{X \in \mathbb{R}^{m \times n}} \left\{ f_q(X) := \frac{\lambda}{2} \|X\|_F^2 + \langle C, X \rangle + \delta_{\mathbb{R}_+^{m \times n}}(X) \mid X\mathbf{1}_n = \boldsymbol{\alpha}, X^\top \mathbf{1}_m = \boldsymbol{\beta} \right\}. \quad (4.2)$$

Clearly, it falls into the form of problem (1.1) (upon vectorization of X). Further, one can show that the dual problem of (4.2) is given by (modulo a minus sign)

$$\min_{\mathbf{u}, \mathbf{v}} f_q^* \left(\mathbf{u}\mathbf{1}_n^\top + \mathbf{1}_m\mathbf{v}^\top \right) - \boldsymbol{\alpha}^\top \mathbf{u} - \boldsymbol{\beta}^\top \mathbf{v}, \quad (4.3)$$

and its associated augmented Lagrangian function is given by (using the similar deduction as Section 2)

$$\begin{aligned} \mathcal{L}_\sigma(\mathbf{u}, \mathbf{v}, X) &= -\boldsymbol{\alpha}^\top \mathbf{u} - \boldsymbol{\beta}^\top \mathbf{v} + \frac{1}{2\sigma} \left\| X + \sigma \left(\mathbf{u}\mathbf{1}_n^\top + \mathbf{1}_m\mathbf{v}^\top \right) \right\|_F^2 \\ &\quad - \frac{1}{2\sigma} \|X\|_F^2 - \mathsf{M}_{\sigma f_q} \left(X + \sigma \left(\mathbf{u}\mathbf{1}_n^\top + \mathbf{1}_m\mathbf{v}^\top \right) \right), \end{aligned}$$

where $\mathbf{u} \in \mathbb{R}^m$, $\mathbf{v} \in \mathbb{R}^n$ are the Lagrange multipliers corresponding to $X\mathbf{1}_n = \boldsymbol{\alpha}$ and $X^\top \mathbf{1}_m = \boldsymbol{\beta}$, respectively, and the conjugate function f_q^* admits the following expression:

$$f_q^*(Z) = \begin{cases} \delta_{\mathbb{R}_+^{m \times n}}(C - Z), & \text{if } \lambda = 0, \\ \frac{1}{2\lambda} \left\| \Pi_{\mathbb{R}_+^{m \times n}}(Z - C) \right\|_F^2, & \text{if } \lambda > 0. \end{cases}$$

Thus, given an arbitrary initial guess $(\mathbf{u}^0, \mathbf{v}^0, X^0) \in \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^{m \times n}$, the basic iterative scheme of our ripALM for solving (4.3) reads as follows:

$$\begin{cases} \left(\mathbf{u}^{k+1}, \mathbf{v}^{k+1} \right) \approx \arg \min_{\mathbf{u} \in \mathbb{R}^m, \mathbf{v} \in \mathbb{R}^n} \{ \Psi_k(\mathbf{u}, \mathbf{v}) \}, & (4.4a) \end{cases}$$

$$\begin{cases} X^{k+1} = \text{prox}_{\sigma_k f_q} \left(X^k + \sigma_k \left(\mathbf{u}^{k+1} \mathbf{1}_n^\top + \mathbf{1}_m (\mathbf{v}^{k+1})^\top \right) \right), & (4.4b) \end{cases}$$

where

$$\Psi_k(\mathbf{u}, \mathbf{v}) := \mathcal{L}_{\sigma_k}(\mathbf{u}, \mathbf{v}, X^k) + \frac{\tau_k}{2\sigma_k} \left\| \mathbf{u} - \mathbf{u}^k \right\|^2 + \frac{\tau_k}{2\sigma_k} \left\| \mathbf{v} - \mathbf{v}^k \right\|^2.$$

To truly implement our ripALM for solving problem (4.2), it is essential to efficiently solve the subproblem (4.4a) to find $(\mathbf{u}^{k+1}, \mathbf{v}^{k+1})$ satisfying the error criterion (2.5). In the following, we shall describe how to apply a semismooth Newton (SSN) method to achieve this goal. For simplicity, we drop the index k and explicitly rewrite the subproblem (4.4a) as follows:

$$\min_{\mathbf{u} \in \mathbb{R}^m, \mathbf{v} \in \mathbb{R}^n} \left\{ \begin{aligned} \Psi(\mathbf{u}, \mathbf{v}) &:= -\boldsymbol{\alpha}^\top \mathbf{u} - \boldsymbol{\beta}^\top \mathbf{v} + \frac{1}{2\sigma} \left\| \bar{X} + \sigma \left(\mathbf{u}\mathbf{1}_n^\top + \mathbf{1}_m\mathbf{v}^\top \right) \right\|_F^2 - \frac{1}{2\sigma} \|\bar{X}\|_F^2 \\ &\quad - \mathsf{M}_{\sigma f_q} \left(\bar{X} + \sigma \left(\mathbf{u}\mathbf{1}_n^\top + \mathbf{1}_m\mathbf{v}^\top \right) \right) + \frac{\tau}{2\sigma} \|\mathbf{u} - \bar{\mathbf{u}}\|^2 + \frac{\tau}{2\sigma} \|\mathbf{v} - \bar{\mathbf{v}}\|^2 \end{aligned} \right\}, \quad (4.5)$$

where $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$ and \bar{X} are given. From the property of the Moreau envelope (see, for example, [2, Proposition 12.30]), we see that Ψ is strongly convex and continuously differentiable with the gradient

$$\nabla \Psi(\mathbf{u}, \mathbf{v}) = \begin{bmatrix} \left(\text{prox}_{\sigma f_q} \left(\bar{X} + \sigma \left(\mathbf{u}\mathbf{1}_n^\top + \mathbf{1}_m\mathbf{v}^\top \right) \right) \right) \mathbf{1}_n - \boldsymbol{\alpha} + \tau\sigma^{-1} (\mathbf{u} - \bar{\mathbf{u}}) \\ \left(\text{prox}_{\sigma f_q} \left(\bar{X} + \sigma \left(\mathbf{u}\mathbf{1}_n^\top + \mathbf{1}_m\mathbf{v}^\top \right) \right) \right)^\top \mathbf{1}_m - \boldsymbol{\beta} + \tau\sigma^{-1} (\mathbf{v} - \bar{\mathbf{v}}) \end{bmatrix}.$$

Note that the proximal mapping $\text{prox}_{\sigma f_q}(\cdot)$ can be easily computed as follows:

$$\text{prox}_{\sigma f_q}(Z) = \frac{1}{1 + \lambda\sigma} \Pi_{\mathbb{R}_+^{m \times n}}(Z - \sigma C), \quad \forall Z \in \mathbb{R}^{m \times n}.$$

Then, from the first-order optimality condition, solving problem (4.5) is equivalent to solving the following non-smooth equation:

$$\nabla \Psi(\mathbf{u}, \mathbf{v}) = \mathbf{0}. \quad (4.6)$$

In view of the nice property of $\nabla \Psi$, we are able to follow [19, 21, 22, 51] to apply a globally convergent and locally superlinearly convergent SSN method to solve (4.6). Specifically, let $\mathcal{W}(\mathbf{u}, \mathbf{v}) := \bar{X} + \sigma(\mathbf{u}\mathbf{1}_n^\top + \mathbf{1}_m\mathbf{v}^\top - C)$ and define the multifunction $\widehat{\partial}(\nabla \Psi) : \mathbb{R}^m \times \mathbb{R}^n \rightrightarrows \mathbb{R}^{(m+n) \times (m+n)}$ as follows:

$$\widehat{\partial}(\nabla \Psi)(\mathbf{u}, \mathbf{v}) := \left\{ H \in \mathbb{R}^{(m+n) \times (m+n)} \left| \begin{array}{l} H := \frac{\sigma}{1 + \lambda\sigma} B \text{Diag}(\text{vec}(\Omega)) B^\top + \frac{\tau}{\sigma} I_{m+n}, \\ \forall \Omega \in \partial \Pi_{\mathbb{R}_+^{m \times n}}(\mathcal{W}(\mathbf{u}, \mathbf{v})) \end{array} \right. \right\},$$

where $B := \begin{bmatrix} \mathbf{1}_n^\top \otimes I_m \\ I_n \otimes \mathbf{1}_m^\top \end{bmatrix} \in \mathbb{R}^{(m+n) \times mn}$ with “ \otimes ” denoting the Kronecker product, $\text{vec}(\Omega)$ denotes the vectorization of Ω with $[\text{vec}(\Omega)]_{i+(j-1)m} = \Omega_{ij}$ for any $1 \leq i \leq m$ and $1 \leq j \leq n$, $\text{Diag}(\mathbf{z})$ denotes the diagonal matrix whose i th diagonal element is given by z_i , and $\partial \Pi_{\mathbb{R}_+^{m \times n}}(Z)$ denotes the generalized Jacobian of the Lipschitz continuous mapping $\Pi_{\mathbb{R}_+^{m \times n}}$ at Z , which is defined by

$$\partial \Pi_{\mathbb{R}_+^{m \times n}}(Z) := \left\{ \Omega \in \mathbb{R}^{m \times n} \left| \Omega_{ij} \in \begin{cases} \{1\}, & \text{if } Z_{ij} > 0, \\ [0, 1], & \text{if } Z_{ij} = 0, \\ \{0\}, & \text{if } Z_{ij} < 0, \end{cases} \right. \right\}.$$

Then, using similar arguments as in the proof of [51, Proposition 4], one can show that $\nabla \Psi(\cdot)$ is strongly semi-smooth with respect to $\widehat{\partial}(\nabla \Psi)(\cdot)$, and thus the SSN method is applicable. More importantly, for any $(\mathbf{u}, \mathbf{v}) \in \mathbb{R}^m \times \mathbb{R}^n$, all elements of $\widehat{\partial}(\nabla \Psi)(\mathbf{u}, \mathbf{v})$ are positive definite. This ensures the direct applicability of the SSN method without the need for a specific regularity condition, such as the primal constraint nondegeneracy condition [55], highlighting the advantage of incorporating a proximal term. The detailed description of the SSN method for solving equation (4.6) is presented in Algorithm 2. We refer readers to [19, Theorem 3.6] for its detailed convergence results.

5 Numerical experiments

In this section, we conduct numerical experiments to evaluate the performance and validate the efficiency of our ripALM in Algorithm 1 for solving the QROT problem (4.1). Specifically, we will conduct experiments from the following two aspects:

- In Section 5.1, we compare our ripALM with SNIPAL [21] and ciPALM [51], which represent the latest absolute-type and relative-type inexact pALMs, respectively. These comparisons aim to illustrate how different error criteria with different tolerance parameters influence the practical numerical performance of the inexact pALM.
- In Section 5.2, we compare our ripALM with Gurobi and a dual alternating direction method of multipliers (dADMM, see Appendix B for more details) for solving

Algorithm 2 A semismooth Newton (SSN) method for solving equation (4.6)

Input: Choose $\bar{\mu} \in (0, 1)$, $\mu \in (0, 1]$, $\eta \in (0, 1/2)$, $\delta \in (0, 1)$, and an initial point $(\mathbf{u}^0, \mathbf{v}^0) \in \mathbb{R}^m \times \mathbb{R}^n$. Set $t = 0$.

while *the termination criterion is not met* **do**

Step 1. Compute $\nabla\Psi(\mathbf{u}^t, \mathbf{v}^t)$ and choose an element $H^t \in \widehat{\partial}(\nabla\Psi)(\mathbf{u}^t, \mathbf{v}^t)$. Solve the linear system

$$H^t \mathbf{d}^t = -\nabla\Psi(\mathbf{u}^t, \mathbf{v}^t), \quad (4.7)$$

nearly exactly by the (sparse) Cholesky decomposition with forward and backward substitutions, or approximately by the preconditioned conjugate gradient (CG) method to find $\mathbf{d}^t := (\mathbf{d}_u^t, \mathbf{d}_v^t)$ such that

$$\|H^t \mathbf{d}^t + \nabla\Psi(\mathbf{u}^t, \mathbf{v}^t)\| \leq \min \left\{ \bar{\mu}, \|\nabla\Psi(\mathbf{u}^t, \mathbf{v}^t)\|^{1+\mu} \right\}.$$

Step 2. (Line search) Find a step size $\alpha_t := \delta^{i_t}$, where i_t is the smallest non-negative integer such that

$$\Psi(\mathbf{u}^t + \delta^{i_t} \mathbf{d}_u^t, \mathbf{v}^t + \delta^{i_t} \mathbf{d}_v^t) - \Psi(\mathbf{u}^t, \mathbf{v}^t) \leq \eta \delta^{i_t} \langle \nabla\Psi(\mathbf{u}^t, \mathbf{v}^t), \mathbf{d}^t \rangle.$$

Step 3. Set $\mathbf{u}^{t+1} = \mathbf{u}^t + \alpha_t \mathbf{d}_u^t$, $\mathbf{v}^{t+1} = \mathbf{v}^t + \alpha_t \mathbf{d}_v^t$, $t = t + 1$, and go to **Step 1**.
end while

large-scale QROT problems. Gurobi represents one of the state-of-the-art commercial solvers, while the dADMM is a widely used first-order method for solving large-scale constrained convex optimization problems; see, for example, [5, 14].

All experiments are run in MATLAB R2023a on a PC with Intel processor i7-12700K@3.60GHz (with 12 cores and 20 threads) and 64GB of RAM, equipped with a Windows OS. The implementation details are given as follows.

Termination conditions. Let $\mathcal{Z}(\mathbf{u}, \mathbf{v}, X) := C + \lambda X - \mathbf{u} \mathbf{1}_n^\top - \mathbf{1}_m \mathbf{v}^\top$. The Karush-Kuhn-Tucker (KKT) system for problem (4.1) and its dual (4.3) is given by

$$X \mathbf{1}_n = \boldsymbol{\alpha}, \quad X^\top \mathbf{1}_m = \boldsymbol{\beta}, \quad \langle X, \mathcal{Z}(\mathbf{u}, \mathbf{v}, X) \rangle = 0, \quad X \geq 0, \quad \mathcal{Z}(\mathbf{u}, \mathbf{v}, X) \geq 0. \quad (5.1)$$

Note that $(\mathbf{u}, \mathbf{v}, X)$ satisfies the KKT system (5.1) if and only if X solves (4.1) and (\mathbf{u}, \mathbf{v}) solves (4.3), respectively. Based on (5.1), we define the relative KKT residual for any $(X, \mathbf{u}, \mathbf{v})$ as follows:

$$\Delta_{\text{kkt}}(\mathbf{u}, \mathbf{v}, X) := \max \{ \Delta_p(X), \Delta_d(\mathbf{u}, \mathbf{v}, X), \Delta_c(\mathbf{u}, \mathbf{v}, X) \},$$

where

$$\Delta_p(X) := \max \left\{ \frac{\|X \mathbf{1}_n - \boldsymbol{\alpha}\|}{1 + \|\boldsymbol{\alpha}\|}, \frac{\|X^\top \mathbf{1}_m - \boldsymbol{\beta}\|}{1 + \|\boldsymbol{\beta}\|}, \frac{\|\min\{X, 0\}\|_F}{1 + \|X\|_F} \right\},$$

$$\Delta_d(\mathbf{u}, \mathbf{v}, X) := \frac{\|\min\{\mathcal{Z}(\mathbf{u}, \mathbf{v}, X), 0\}\|_F}{1 + \|C\|_F}, \quad \Delta_c(\mathbf{u}, \mathbf{v}, X) := \frac{|\langle X, \mathcal{Z}(\mathbf{u}, \mathbf{v}, X) \rangle|}{1 + \|C\|_F}.$$

Moreover, we define the relative duality gap as follows:

$$\Delta_{\text{gap}}(\mathbf{u}, \mathbf{v}, X) := \frac{|\text{pobj}(X) - \text{dobj}(\mathbf{u}, \mathbf{v})|}{1 + |\text{pobj}(X)| + |\text{dobj}(\mathbf{u}, \mathbf{v})|},$$

where $\text{pobj}(X) := f_q(X)$ and $\text{dobj}(\mathbf{u}, \mathbf{v}) := -f_q^*(\mathbf{u}\mathbf{1}_n^\top + \mathbf{1}_m\mathbf{v}^\top) + \boldsymbol{\alpha}^\top \mathbf{u} + \boldsymbol{\beta}^\top \mathbf{v}$. Using these relative residuals, we will terminate our ripALM when it returns a point $(\mathbf{u}^k, \mathbf{v}^k, X^k)$ satisfying

$$\Delta_{\text{res}}^k := \max \left\{ \Delta_{\text{kkt}}(\mathbf{u}^k, \mathbf{v}^k, X^k), \Delta_{\text{gap}}(\mathbf{u}^k, \mathbf{v}^k, X^k) \right\} < 10^{-6}.$$

Initial points. For the experiments in Section 5.1, we simply initialize all inexact pALMs at the origin to amplify the impact of inexactness. For the experiments in Section 5.2, we will employ a *warm-start* strategy for more efficiency. Indeed, our numerical experience (see, e.g., [21, 23, 51]) have suggested that a reasonably good initial point would benefit the practical performance of such a “pALM + SSN” algorithmic framework. Therefore, for the experiments in Section 5.2, we first employ an inexact Bregman proximal gradient method (iBPGM) with Sinkhorn’s algorithm as a subsolver for solving the QROT problem (4.1) to generate an initial point for our ripALM; see Appendix C for more details. Specifically, we terminate this iBPGM as long as it produces a point $(\mathbf{u}^k, \mathbf{v}^k, X^k)$ such that $\Delta_{\text{res}}^k < 10^{-3}$, or it reaches the maximal number of iterations 500. Note that the time consumed during the warm-starting phase is included in the total computational time for our ripALM. Additionally, the initial error variable \mathbf{w}^0 in our ripALM is always set to $\mathbf{0}$.

For the SSN method in Algorithm 2, we will initialize it with the origin at the first pALM iteration and then employ a *warm-start* strategy thereafter. Specifically, at each pALM iteration, we initialize the SSN method with the approximate solution obtained by the SSN method in the previous pALM iteration.

Hyperparameters. Our ripALM as well as SNIPAL and ciPALM require appropriate choices of $\{\sigma_k\}$ and $\{\tau_k\}$ to achieve superior performance. In our experiments, for all algorithms, we simply set $\sigma_k = \min \{10^4, \max \{10^{-4}, 1.5^k\}\}$ and $\tau_k \equiv 5$ for all $k \geq 0$. These choices of $\{\tau_k\}$ and $\{\sigma_k\}$ can satisfy the required conditions in Theorems 3.1 and 3.2; see also Remark 3.1. Moreover, we would like to emphasize that more sophisticated updating rules for σ_k and τ_k are possible and may lead to improved numerical performance. In addition, for the SSN method in Algorithm 2, we set $\eta = 10^{-4}$, $\delta = 0.5$, $\bar{\mu} = 10^{-3}$ and $\mu = 0.2$.

5.1 Comparison with SNIPAL and ciPALM

In this part of experiments, we compare our ripALM with SNIPAL [21] and ciPALM [51] for solving the QROT problem (4.1) to illustrate how different error criteria with different tolerance parameters influence the numerical performance of the inexact pALM. Note that both our ripALM and the ciPALM are of relative-type and only require a single tolerance parameter ρ , as shown in (2.5) and (2.8), respectively, while the SNIPAL is of absolute-type and requires two summable tolerance parameter sequences $\{\varepsilon_k\}$ and $\{\delta_k\}$, as shown in (2.7). For simplicity, in our comparisons, we set $\varepsilon_k = \varepsilon_0/(k+1)^p$, $\delta_k = \delta_0/(k+1)^q$ with $\varepsilon_0 = \delta_0 \in \{0.01, 1\}$ and $p, q \in \{1.1, 2.1, 3.1\}$ (hence, there are 18 combinations in total) for SNIPAL. For our ripALM and the ciPALM, we consider $\rho \in \{0.01, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9, 0.99, 0.999\}$ (hence, there are 9 choices).

We use images from the `ClassicImages` class in the DOTmark collection [41], which serves as a benchmark dataset for the OT problem and its variants, to generate the QROT instance. Note that the images in the `ClassicImages` class consist of ten different images, each with different resolutions of 32×32 , 64×64 , 128×128 and 512×512 . Thus, for each resolution, we can pair any two different images and compute the QROT problem, resulting in 45 QROT problems. Moreover, the cost matrix C is obtained by calculating the squared Euclidean distances between pixels.

Tables 1 and 2 present the average results (over 45 instances) for 64×64 resolution with the regularization parameter $\lambda \in \{1, 0.1\}$. From the results, one can see that the

performance of all algorithms depends on the choices of tolerance parameters, and with proper tuning, their performances can be comparable. This is indeed expected since all the algorithms essentially employ the same “pALM + SSN” framework, but differ in their error criteria for solving the subproblems. Notably, since our ripALM and the ciPALM involve only a single tolerance parameter $\rho \in [0, 1)$, they are more user-friendly and easier to tune since a simple 1-D grid-search strategy is sufficient. We also observe that the number of pALM iterations for the three methods remains unchanged (e.g., 17). However, using an absolute-type error criterion often leads to redundant efforts in solving the subproblems. In contrast, a relative-type error criterion has the potential to alleviate this issue. This supports the main motivation for developing a relative-type error criterion. Moreover, one can see that our ripALM always outperforms the ciPALM in terms of the total number of the SSN iterations, resulting in a substantial reduction in computational time. This is because the ciPALM requires an extra correction step to guarantee the convergence, which tends to increase the number of SSN iterations³ needed to achieve its error criterion (2.8), though this excessive cost does not contribute to the progress of pALM iterations. These observations further support the use of vanilla inexact pALM, as advocated in this paper. Clearly, our ripALM exhibits greater robustness and efficiency, as shown in Tables 1 and 2. Moreover, observe that the performance of ripALM is relatively stable across different parameter values of ρ , with slight advantage for $\rho \geq 0.5$.

5.2 Comparison with Gurobi and dADMM

In this part of experiments, we compare our ripALM with Gurobi (version 10.0.1) and dADMM for solving large-scale QROT problems. For our ripALM, we set $\rho = 0.99$ based on the numerical observations from the previous section. For Gurobi, we use its default termination conditions and set the corresponding termination tolerances as 10^{-6} , aligning it with our tolerance. For dADMM, we initialize the penalty parameter as $\sigma_0 = 0.01\|C\|_F^{-1}$, and dynamically adjust it based on the primal-dual residuals, following the approach described in [50, Section 5.1], to achieve superior empirical performance. Moreover, we will terminate dADMM when it returns a point $(\mathbf{u}^k, \mathbf{v}^k, X^k)$ satisfying $\Delta_{\text{res}}^k < 10^{-6}$, or its number of iterations reaches 10000.

We follow [52, Section 5.1] to generate a random QROT instance. Specifically, we first generate two discrete probability distributions $\{(a_i, \mathbf{p}_i) \in \mathbb{R}_+ \times \mathbb{R}^3 : i = 1, \dots, m\}$ and $\{(b_j, \mathbf{q}_j) \in \mathbb{R}_+ \times \mathbb{R}^3 : j = 1, \dots, n\}$. Here, $\mathbf{a} := (a_1, \dots, a_m)^\top$ and $\mathbf{b} := (b_1, \dots, b_n)^\top$ are probabilities/weights, which are generated from the uniform distribution on the open interval $(0, 1)$ and further normalized such that $\sum_i^m a_i = \sum_j^n b_j = 1$. Moreover, $\{\mathbf{p}_i\}$ and $\{\mathbf{q}_j\}$ are support points whose entries are drawn from a five-component multivariate Gaussian mixture distribution, with a mean vector $(-20, 10, 0, 10, 20)^\top$ and a variance vector $(5, 5, 5, 5, 5)^\top$, using randomly generated weights. Then, the cost matrix C is generated by $c_{ij} = \|\mathbf{p}_i - \mathbf{q}_j\|^2$ for $1 \leq i \leq m$ and $1 \leq j \leq n$ and normalized by dividing (element-wise) by its maximal entry.

We then generate a set of random instances with $m = n \in \{1000, 2000, \dots, 10000\}$. For each m , we generate 10 instances with different random seeds, and present the average numerical performances of our ripALM, dADMM and Gurobi in Table 3 and Figure 1, with $\lambda \in \{1, 0.1\}$. It can be observed that both ripALM and Gurobi are able to solve the tested problems accurately, in the sense that the residual Δ_{res} is smaller than 10^{-6} . In contrast, the dADMM, even after 10000 iterations, can only produce lower-quality solutions, especially

³One possible reason is that the variable derived from performing the corrected step is no longer a good initial point for the SSN method in the next iteration.

Table 1: Comparisons between ripALM, SNIPAL and ciPALM under different choices of tolerance parameters, where $\lambda = 1$ and the instances are generated using images with the resolution of 64×64 in the `ClassicImages` class from the DOTmark collection. In the table, “ Δ_{res} ” denotes the terminating Δ_{res}^k ; “#” denotes the number of iterations (the total number of the SSN iterations is given in the bracket); “time” denotes the computational time in seconds.

SNIPAL							
$(\varepsilon_0 = \delta_0, p, q)$	Δ_{res}	#	time	$(\varepsilon_0 = \delta_0, p, q)$	Δ_{res}	#	time
(1, 1.1, 1.1)	7.85e-07	17 (52)	9.56	(0.01, 1.1, 1.1)	7.85e-07	17 (61)	11.12
(1, 1.1, 2.1)	7.85e-07	17 (55)	9.97	(0.01, 1.1, 2.1)	7.85e-07	17 (63)	11.36
(1, 1.1, 3.1)	7.85e-07	17 (58)	10.33	(0.01, 1.1, 3.1)	7.85e-07	17 (64)	11.57
(1, 2.1, 1.1)	7.85e-07	17 (52)	9.55	(0.01, 2.1, 1.1)	7.85e-07	17 (61)	11.12
(1, 2.1, 2.1)	7.85e-07	17 (55)	9.98	(0.01, 2.1, 2.1)	7.85e-07	17 (63)	11.36
(1, 2.1, 3.1)	7.85e-07	17 (58)	10.33	(0.01, 2.1, 3.1)	7.85e-07	17 (64)	11.56
(1, 3.1, 1.1)	7.85e-07	17 (52)	9.55	(0.01, 3.1, 1.1)	7.85e-07	17 (61)	11.13
(1, 3.1, 2.1)	7.85e-07	17 (55)	9.99	(0.01, 3.1, 2.1)	7.85e-07	17 (63)	11.36
(1, 3.1, 3.1)	7.85e-07	17 (58)	10.34	(0.01, 3.1, 3.1)	7.85e-07	17 (64)	11.57

ripALM				ciPALM			
ρ	Δ_{res}	#	time	ρ	Δ_{res}	#	time
0.999	7.85e-07	17 (47)	8.86	0.999	7.85e-07	17 (68)	15.49
0.99	7.85e-07	17 (47)	8.86	0.99	7.85e-07	17 (68)	15.53
0.9	7.85e-07	17 (47)	8.86	0.9	7.85e-07	17 (68)	15.17
0.7	7.85e-07	17 (47)	8.99	0.7	7.85e-07	17 (66)	14.08
0.5	7.85e-07	17 (48)	9.07	0.5	7.85e-07	17 (63)	12.95
0.3	7.85e-07	17 (49)	9.21	0.3	7.85e-07	17 (60)	11.51
0.1	7.85e-07	17 (50)	9.45	0.1	7.85e-07	17 (57)	10.65
0.05	7.85e-07	17 (51)	9.61	0.05	7.85e-07	17 (56)	10.51
0.01	7.85e-07	17 (53)	9.94	0.01	7.85e-07	17 (58)	10.76

when the problem size becomes large. Moreover, we have also observed that Gurobi can be rather time-consuming and memory-consuming for large-scale problems. As an example, for the case where $m = n = 10000$, a large-scale QP containing 10^8 nonnegative variables and 20000 equality constraints was solved. One can see that Gurobi is around $2 \sim 4$ times slower than our ripALM. In addition, Gurobi may lack robustness, especially for solving large-scale problems. Indeed, as observed from Figure 1, the computational times taken by Gurobi can vary a lot among the 10 randomly generated instances. In conclusion, our ripALM shows potential for greater efficiency and robustness in solving large-scale QROT problems.

6 Conclusions

In this paper, we developed a relative-type inexact proximal augmented Lagrangian method (ripALM) for solving a class of linearly constrained convex optimization problems. The proposed ripALM is the first relative-type inexact version of the vanilla pALM with provable convergence guarantees. By employing a relative-type error criterion, it simplifies implementation and parameter tuning, compared to the absolute-type inexact pALM. We conducted

Table 2: Same as Table 1 but for $\lambda = 0.1$.

SNIPAL							
$(\varepsilon_0 = \delta_0, p, q)$	Δ_{res}	#	time	$(\varepsilon_0 = \delta_0, p, q)$	Δ_{res}	#	time
(1, 1.1, 1.1)	6.43e-07	19 (92)	28.48	(0.01, 1.1, 1.1)	6.43e-07	19 (101)	30.04
(1, 1.1, 2.1)	6.43e-07	19 (95)	28.91	(0.01, 1.1, 2.1)	6.43e-07	19 (103)	30.31
(1, 1.1, 3.1)	6.43e-07	19 (98)	29.26	(0.01, 1.1, 3.1)	6.43e-07	19 (105)	30.50
(1, 2.1, 1.1)	6.43e-07	19 (92)	28.49	(0.01, 2.1, 1.1)	6.43e-07	19 (101)	30.06
(1, 2.1, 2.1)	6.43e-07	19 (95)	28.91	(0.01, 2.1, 2.1)	6.43e-07	19 (103)	30.28
(1, 2.1, 3.1)	6.43e-07	19 (98)	29.27	(0.01, 2.1, 3.1)	6.43e-07	19 (105)	30.47
(1, 3.1, 1.1)	6.43e-07	19 (92)	28.49	(0.01, 3.1, 1.1)	6.43e-07	19 (101)	30.02
(1, 3.1, 2.1)	6.43e-07	19 (95)	28.92	(0.01, 3.1, 2.1)	6.43e-07	19 (103)	30.29
(1, 3.1, 3.1)	6.43e-07	19 (98)	29.28	(0.01, 3.1, 3.1)	6.43e-07	19 (105)	30.48

ripALM				ciPALM			
ρ	Δ_{res}	#	time	ρ	Δ_{res}	#	time
0.999	6.43e-07	19 (86)	27.77	0.999	6.43e-07	19 (108)	34.70
0.99	6.43e-07	19 (86)	27.78	0.99	6.43e-07	19 (108)	34.71
0.9	6.43e-07	19 (86)	27.78	0.9	6.43e-07	19 (107)	34.11
0.7	6.43e-07	19 (87)	27.88	0.7	6.43e-07	19 (106)	33.29
0.5	6.43e-07	19 (87)	27.95	0.5	6.43e-07	19 (103)	32.10
0.3	6.43e-07	19 (88)	28.10	0.3	6.43e-07	19 (100)	30.60
0.1	6.43e-07	19 (90)	28.32	0.1	6.43e-07	19 (97)	29.66
0.05	6.43e-07	19 (91)	28.47	0.05	6.43e-07	19 (96)	29.45
0.01	6.43e-07	19 (92)	28.83	0.01	6.43e-07	19 (98)	29.73

a thorough convergence analysis and demonstrated the competitive efficiency of our ripALM through numerical experiments on solving quadratically regularized optimal transport problems. Since our algorithm is applicable to a more general linearly constrained convex optimization problem, future work may explore potential applications of this framework to other practical problems, including general regularized optimal transport problems studied in [51].

A Missing Proofs in Section 3

A.1 Proof of Theorem 3.1

Proof. Statement (i). First, recall that $\ell(\mathbf{y}, \mathbf{x}) = -\mathbf{b}^\top \mathbf{y} + \langle \mathbf{x}, A^\top \mathbf{y} \rangle - f(\mathbf{x})$, which is convex in \mathbf{y} and concave in \mathbf{x} . We see that $\partial \ell(\mathbf{y}, \mathbf{x}) = \{A\mathbf{x} - \mathbf{b}\} \times \{v - A^\top \mathbf{y} \mid v \in \partial f(\mathbf{x})\}$. By using the relations in (2.5) and (2.6), along with some manipulations, we can obtain the following results for all $k \geq 0$:

$$\begin{cases} (\Delta^{k+1} - \tau_k \sigma_k^{-1}(\mathbf{y}^{k+1} - \mathbf{y}^k), \sigma_k^{-1}(\mathbf{x}^k - \mathbf{x}^{k+1})) \in \partial \ell(\mathbf{y}^{k+1}, \mathbf{x}^{k+1}), & \text{(A.1a)} \\ 2|\langle \mathbf{w}^k - \mathbf{y}^{k+1}, \sigma_k \Delta^{k+1} \rangle| + \|\sigma_k \Delta^{k+1}\|^2 \leq \rho(\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 + \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2), & \text{(A.1b)} \\ \mathbf{w}^{k+1} = \mathbf{w}^k - \sigma_k \Delta^{k+1}. & \text{(A.1c)} \end{cases}$$

Let $(\mathbf{y}^*, \mathbf{x}^*) \in \mathbb{R}^M \times \mathbb{R}^N$ be an arbitrary saddle point of ℓ and hence $(\mathbf{0}, \mathbf{0}) \in \partial \ell(\mathbf{y}^*, \mathbf{x}^*)$.

Table 3: Numerical results of Gurobi, dADMM, and ripALM. In the table, “ Δ_{res} ” denotes the terminating Δ_{res}^k ; “#” denotes the number of iterations (the total number of the SSN iterations is given in the bracket); “time” denotes the computational time.

$m = n$	Gurobi			dADMM			ripALM		
	Δ_{res}	#	time	Δ_{res}	#	time	Δ_{res}	#	time
$\lambda = 1$									
1000	2.70e-07	32	4.53	2.71e-06	10000	15.21	5.36e-07	16 (48)	1.29
2000	3.61e-07	38	22.15	1.20e-05	10000	99.73	6.89e-07	17 (62)	6.12
3000	2.61e-07	40	60.87	2.11e-05	10000	240.03	5.47e-07	17 (70)	15.93
4000	4.23e-07	43	125.35	4.68e-05	10000	431.83	6.19e-07	17 (78)	29.84
5000	4.22e-07	45	205.56	7.32e-05	10000	678.86	6.66e-07	17 (82)	49.36
6000	5.07e-07	45	312.47	1.04e-04	10000	979.18	6.11e-07	17 (86)	72.30
7000	5.54e-07	45	407.12	1.32e-04	10000	1337.06	5.25e-07	18 (90)	104.18
8000	5.17e-07	48	625.94	1.56e-04	10000	1746.24	4.75e-07	18 (96)	147.89
9000	1.47e-06	50	842.62	2.16e-04	10000	2208.15	6.03e-07	18 (98)	185.94
10000	5.63e-07	49	1031.17	2.53e-04	10000	2724.33	6.09e-07	18 (98)	240.24
$\lambda = 0.1$									
1000	2.55e-07	33	4.63	3.17e-05	10000	15.31	7.50e-07	18 (83)	1.65
2000	5.55e-07	38	22.26	1.39e-04	10000	100.22	5.77e-07	19 (107)	8.76
3000	1.75e-06	41	57.58	2.78e-04	10000	238.54	5.29e-07	19 (119)	22.82
4000	6.43e-07	43	118.20	5.01e-04	10000	430.89	6.85e-07	19 (129)	44.08
5000	4.71e-07	45	191.02	7.26e-04	10000	678.03	5.49e-07	20 (141)	78.06
6000	5.57e-07	47	298.67	1.08e-03	10000	977.75	5.92e-07	20 (148)	125.25
7000	6.98e-07	49	437.49	1.30e-03	10000	1332.25	6.21e-07	20 (149)	173.81
8000	7.95e-07	49	629.63	1.62e-03	10000	1740.13	7.59e-07	20 (157)	240.14
9000	5.87e-07	51	794.23	2.15e-03	10000	2202.89	6.03e-07	21 (168)	346.25
10000	6.02e-07	52	1073.12	2.43e-03	10000	2718.42	4.96e-07	21 (176)	462.52

For all $k \geq 0$,

$$\begin{aligned} \|\mathbf{x}^k - \mathbf{x}^*\|^2 &= \|\mathbf{x}^k - \mathbf{x}^{k+1} + \mathbf{x}^{k+1} - \mathbf{x}^*\|^2 \\ &= \|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 + 2\langle \mathbf{x}^k - \mathbf{x}^{k+1}, \mathbf{x}^{k+1} - \mathbf{x}^* \rangle + \|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2. \end{aligned}$$

By letting $\boldsymbol{\xi}^{k+1} := \sigma_k^{-1}(\mathbf{x}^k - \mathbf{x}^{k+1})$, the above equation can be reformulated as

$$\|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 = \|\mathbf{x}^k - \mathbf{x}^*\|^2 - 2\sigma_k \langle \mathbf{x}^{k+1} - \mathbf{x}^*, \boldsymbol{\xi}^{k+1} \rangle - \|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2. \quad (\text{A.2})$$

Then, using the relation $\mathbf{w}^{k+1} = \mathbf{w}^k - \sigma_k \Delta^{k+1}$ (by (A.1c)), we see that

$$\begin{aligned} \|\mathbf{w}^{k+1} - \mathbf{y}^*\|^2 &= \|\mathbf{w}^k - \sigma_k \Delta^{k+1} - \mathbf{y}^*\|^2 \\ &= \|\mathbf{w}^k - \mathbf{y}^*\|^2 - 2\langle \mathbf{w}^k - \mathbf{y}^*, \sigma_k \Delta^{k+1} \rangle + \|\sigma_k \Delta^{k+1}\|^2 \\ &= \|\mathbf{w}^k - \mathbf{y}^*\|^2 - 2\langle \mathbf{w}^k - \mathbf{y}^{k+1}, \sigma_k \Delta^{k+1} \rangle + \|\sigma_k \Delta^{k+1}\|^2 \\ &\quad - 2\sigma_k \langle \mathbf{y}^{k+1} - \mathbf{y}^*, \boldsymbol{\theta}^{k+1} \rangle - 2\tau_k \langle \mathbf{y}^{k+1} - \mathbf{y}^*, \mathbf{y}^{k+1} - \mathbf{y}^k \rangle. \end{aligned} \quad (\text{A.3})$$

where $\boldsymbol{\theta}^{k+1} := \Delta^{k+1} - \tau_k \sigma_k^{-1}(\mathbf{y}^{k+1} - \mathbf{y}^k)$. Similarly,

$$\tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^*\|^2 = \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2 - \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2 + 2\tau_k \langle \mathbf{y}^{k+1} - \mathbf{y}^*, \mathbf{y}^{k+1} - \mathbf{y}^k \rangle. \quad (\text{A.4})$$

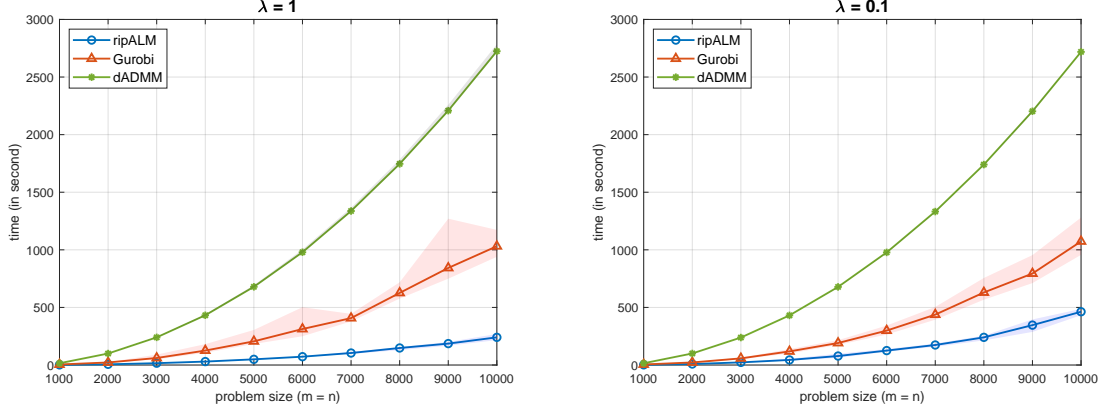


Figure 1: Comparisons among ripALM, Gurobi, and dADMM for the QROT problem with $m = n \in \{1000, 2000, \dots, 10000\}$. The shaded region indicates the maximum and minimum computation times taken to solve the ten instances for each problem dimension.

By summing (A.2), (A.3) and (A.4), we have that

$$\begin{aligned}
& \|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 + \|\mathbf{w}^{k+1} - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^*\|^2 \\
&= \|\mathbf{x}^k - \mathbf{x}^*\|^2 + \|\mathbf{w}^k - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2 \\
&\quad - 2\sigma_k \left(\langle \mathbf{y}^{k+1} - \mathbf{y}^*, \boldsymbol{\theta}^{k+1} \rangle + \langle \mathbf{x}^{k+1} - \mathbf{x}^*, \boldsymbol{\xi}^{k+1} \rangle \right) - 2\langle \mathbf{w}^k - \mathbf{y}^{k+1}, \sigma_k \Delta^{k+1} \rangle \\
&\quad + \|\sigma_k \Delta^{k+1}\|^2 - \|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 - \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2.
\end{aligned} \tag{A.5}$$

Note from (A.1a) that

$$(\boldsymbol{\theta}^{k+1}, \boldsymbol{\xi}^{k+1}) \in \partial \ell(\mathbf{y}^{k+1}, \mathbf{x}^{k+1}), \tag{A.6}$$

which, together with $\mathbf{0} \in \partial \ell(\mathbf{y}^*, \mathbf{x}^*)$ and the monotonicity of $\partial \ell$, yields

$$\langle \mathbf{y}^{k+1} - \mathbf{y}^*, \boldsymbol{\theta}^{k+1} \rangle + \langle \mathbf{x}^{k+1} - \mathbf{x}^*, \boldsymbol{\xi}^{k+1} \rangle \geq 0.$$

Moreover, by using (A.1b), we see that

$$\begin{aligned}
-2\langle \mathbf{w}^k - \mathbf{y}^{k+1}, \sigma_k \Delta^{k+1} \rangle + \|\sigma_k \Delta^{k+1}\|^2 &\leq 2|\langle \mathbf{w}^k - \mathbf{y}^{k+1}, \sigma_k \Delta^{k+1} \rangle| + \|\sigma_k \Delta^{k+1}\|^2 \\
&\leq \rho \left(\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 + \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2 \right).
\end{aligned}$$

Substituting the above two inequalities into (A.5), we obtain a key inequality for the subsequent convergence analysis:

$$\begin{aligned}
& \|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 + \|\mathbf{w}^{k+1} - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^*\|^2 \\
&\leq \|\mathbf{x}^k - \mathbf{x}^*\|^2 + \|\mathbf{w}^k - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2 \\
&\quad - (1 - \rho) \left(\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 + \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2 \right).
\end{aligned} \tag{A.7}$$

The inequality (A.7), together with $\rho \in [0, 1)$ and $\tau_{k+1} \leq (1 + \nu_k)\tau_k$ with $\nu_k \geq 0$ and $\sum \nu_i < \infty$ for all $k \geq 0$, implies that

$$\begin{aligned}
& \|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 + \|\mathbf{w}^{k+1} - \mathbf{y}^*\|^2 + \tau_{k+1} \|\mathbf{y}^{k+1} - \mathbf{y}^*\|^2 \\
&\leq (1 + \nu_k) \left(\|\mathbf{x}^k - \mathbf{x}^*\|^2 + \|\mathbf{w}^k - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2 \right).
\end{aligned} \tag{A.8}$$

Since $\{\nu_k\}$ is a non-negative summable sequence, it then follows from [32, Lemma 2 in Section 2.2] that the sequence $\{\|\mathbf{x}^k - \mathbf{x}^*\|^2 + \|\mathbf{w}^k - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2\}$ is convergent. This together with $\tau_k \geq \tau_{\min} > 0$ implies that all sequences $\{\mathbf{x}^k\}$, $\{\mathbf{w}^k\}$, $\{\mathbf{y}^k\}$ are bounded.

Statement (ii). Using (A.7) again with $\tau_{k+1} \leq (1 + \nu_k)\tau_k$ with $\nu_k \geq 0$ and $\sum \nu_i < \infty$ for all $k \geq 0$, we have that

$$\begin{aligned} 0 &\leq (1 - \rho)(1 + \nu_k) \left(\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 + \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2 \right) \\ &\leq (1 + \nu_k) \left(\|\mathbf{x}^k - \mathbf{x}^*\|^2 + \|\mathbf{w}^k - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2 \right) \\ &\quad - \left(\|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 + \|\mathbf{w}^{k+1} - \mathbf{y}^*\|^2 + \tau_{k+1} \|\mathbf{y}^{k+1} - \mathbf{y}^*\|^2 \right). \end{aligned} \quad (\text{A.9})$$

Since $\{\|\mathbf{x}^k - \mathbf{x}^*\|^2 + \|\mathbf{w}^k - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2\}$ is convergent, $\nu_k \rightarrow 0$ (due to $\nu_k \geq 0$ and $\sum \nu_i < \infty$) and $\rho \in [0, 1)$, it then follows from (A.9) that

$$\lim_{k \rightarrow \infty} \|\mathbf{x}^{k+1} - \mathbf{x}^k\| = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \|\sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k)\| = 0. \quad (\text{A.10})$$

Note that both sequences $\{\sigma_k\}$ and $\{\tau_k\}$ are bounded away from 0. Thus, we further have that $\lim_{k \rightarrow \infty} \boldsymbol{\xi}^{k+1} (:= \sigma_k^{-1}(\mathbf{x}^k - \mathbf{x}^{k+1})) = \mathbf{0}$ and $\lim_{k \rightarrow \infty} \|\mathbf{y}^{k+1} - \mathbf{y}^k\| = 0$. Moreover, using (A.10) together with (A.1b) implies that

$$\lim_{k \rightarrow \infty} |\langle \mathbf{w}^k - \mathbf{y}^{k+1}, \sigma_k \Delta^{k+1} \rangle| = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \|\sigma_k \Delta^{k+1}\|^2 = 0.$$

Since $\{\sigma_k\}$ is bounded away from 0, we then obtain that $\lim_{k \rightarrow \infty} \langle \mathbf{w}^k - \mathbf{y}^{k+1}, \Delta^{k+1} \rangle = 0$ and $\lim_{k \rightarrow \infty} \Delta^{k+1} = \mathbf{0}$. Finally, recall again that $\tau_{k+1} \leq (1 + \nu_k)\tau_k$ with $\nu_k \geq 0$ and $\sum \nu_i < \infty$ for all $k \geq 0$. Thus, τ_k must be bounded from above and hence $\tau_k \sigma_k^{-1}$ is also bounded from above. Consequently, we can obtain that $\lim_{k \rightarrow \infty} \tau_k \sigma_k^{-1}(\mathbf{y}^{k+1} - \mathbf{y}^k) = \mathbf{0}$ and hence $\lim_{k \rightarrow \infty} \boldsymbol{\theta}^{k+1} = \mathbf{0}$.

Statement (iii). We first study the limit of $\{F(\boldsymbol{\theta}^{k+1}, \mathbf{x}^{k+1})\}$. Using relations (A.6) and (3.3), we have that $(\mathbf{y}^{k+1}, \boldsymbol{\xi}^{k+1}) \in \partial F(\boldsymbol{\theta}^{k+1}, \mathbf{x}^{k+1})$. Then, by the concavity of F , it holds that, for all $k \geq 0$,

$$F(\mathbf{0}, \mathbf{x}^*) \leq F(\boldsymbol{\theta}^{k+1}, \mathbf{x}^{k+1}) + \langle \mathbf{y}^{k+1}, \boldsymbol{\theta}^{k+1} \rangle + \langle \boldsymbol{\xi}^{k+1}, \mathbf{x}^{k+1} - \mathbf{x}^* \rangle.$$

Since $\lim_{k \rightarrow \infty} \boldsymbol{\theta}^{k+1} = \mathbf{0}$, $\lim_{k \rightarrow \infty} \boldsymbol{\xi}^{k+1} = \mathbf{0}$, and the sequences $\{\mathbf{x}^k\}$ and $\{\mathbf{y}^k\}$ are bounded, we can obtain from the above inequality that

$$\liminf_{k \rightarrow \infty} F(\boldsymbol{\theta}^{k+1}, \mathbf{x}^{k+1}) \geq F(\mathbf{0}, \mathbf{x}^*). \quad (\text{A.11})$$

On the other hand, since $\{\mathbf{x}^k\}$ is bounded, it has at least one accumulation point. Suppose that \mathbf{x}^∞ is an accumulation point and $\{\mathbf{x}^{k_i}\}$ is a convergent subsequence such that $\lim_{i \rightarrow \infty} \mathbf{x}^{k_i} = \mathbf{x}^\infty$. Since $\lim_{k \rightarrow \infty} \|\mathbf{x}^{k+1} - \mathbf{x}^k\| = 0$, we also have that $\lim_{i \rightarrow \infty} \mathbf{x}^{k_i+1} = \mathbf{x}^\infty$. Thus, by passing to a further subsequence if necessary, we may assume without loss of generality that the subsequence $\{F(\boldsymbol{\theta}^{k_i+1}, \mathbf{x}^{k_i+1})\}$ satisfies

$$\lim_{i \rightarrow \infty} F(\boldsymbol{\theta}^{k_i+1}, \mathbf{x}^{k_i+1}) = \limsup_{k \rightarrow \infty} F(\boldsymbol{\theta}^{k+1}, \mathbf{x}^{k+1}).$$

Note that F is closed upper semicontinuous concave (see, for example, [37, Theorem 7]), and thus $\text{dom } F$ is closed. This together with $(\mathbf{0}, \mathbf{x}^\infty) = \lim_{i \rightarrow \infty} (\boldsymbol{\theta}^{k_i+1}, \mathbf{x}^{k_i+1})$ induces that

$(\mathbf{0}, \mathbf{x}^\infty) \in \text{dom } F$. Then, we see that

$$\begin{aligned}
& F(\mathbf{0}, \mathbf{x}^*) \\
& \geq F(\mathbf{0}, \mathbf{x}^\infty) && \text{(since } \mathbf{x}^* \text{ is optimal for problem (3.2))} \\
& = F\left(\lim_{i \rightarrow \infty} \boldsymbol{\theta}^{k_i+1}, \lim_{i \rightarrow \infty} \mathbf{x}^{k_i+1}\right) && \text{(since } \lim_{k \rightarrow \infty} \boldsymbol{\theta}^{k+1} = \mathbf{0} \text{ and } \lim_{i \rightarrow \infty} \mathbf{x}^{k_i+1} = \mathbf{x}^\infty) \\
& \geq \limsup_{i \rightarrow \infty} F(\boldsymbol{\theta}^{k_i+1}, \mathbf{x}^{k_i+1}) && \text{(since } F \text{ is upper semicontinuous)} \\
& = \limsup_{k \rightarrow \infty} F(\boldsymbol{\theta}^{k+1}, \mathbf{x}^{k+1}). && \text{(by the choice of subsequence } \{\mathbf{x}^{k_i+1}\})
\end{aligned}$$

This together with (A.11) implies that

$$\lim_{k \rightarrow \infty} F(\boldsymbol{\theta}^{k+1}, \mathbf{x}^{k+1}) = F(\mathbf{0}, \mathbf{x}^*). \quad (\text{A.12})$$

We next study the limit of $\{G(\mathbf{y}^{k+1}, \boldsymbol{\xi}^{k+1})\}$. Since $-F$ and G are convex conjugate and $(\boldsymbol{\theta}^{k+1}, \mathbf{x}^{k+1}) \in \partial G(\mathbf{y}^{k+1}, \boldsymbol{\xi}^{k+1})$, we can get the following equality by using the Fenchel equality (see, for example, [36, Theorem 23.5]):

$$G(\mathbf{y}^{k+1}, \boldsymbol{\xi}^{k+1}) = F(\boldsymbol{\theta}^{k+1}, \mathbf{x}^{k+1}) + \langle \boldsymbol{\theta}^{k+1}, \mathbf{y}^{k+1} \rangle + \langle \mathbf{x}^{k+1}, \boldsymbol{\xi}^{k+1} \rangle.$$

Since $\lim_{k \rightarrow \infty} \boldsymbol{\theta}^{k+1} = \mathbf{0}$, $\lim_{k \rightarrow \infty} \boldsymbol{\xi}^{k+1} = \mathbf{0}$, and the sequences $\{\mathbf{x}^k\}$ and $\{\mathbf{y}^k\}$ are bounded, we obtain that

$$\lim_{k \rightarrow \infty} G(\mathbf{y}^{k+1}, \boldsymbol{\xi}^{k+1}) = F(\mathbf{0}, \mathbf{x}^*) = G(\mathbf{y}^*, \mathbf{0}).$$

This proves statement (iii).

Statement (iv). We first prove that any accumulation point of $\{\mathbf{y}^k\}$ is an optimal solution of problem (2.1). Since $\{\mathbf{y}^k\}$ is bounded by statement (i), the sequence $\{\mathbf{y}^k\}$ has at least one accumulation point. Suppose that \mathbf{y}^∞ is an accumulation point and $\{\mathbf{y}^{k_j}\}$ is a convergent subsequence such that $\lim_{j \rightarrow \infty} \mathbf{y}^{k_j} = \mathbf{y}^\infty$. Since $\lim_{k \rightarrow \infty} \|\mathbf{y}^{k+1} - \mathbf{y}^k\| = 0$, we also have that $\lim_{j \rightarrow \infty} \mathbf{y}^{k_j+1} = \mathbf{y}^\infty$. Then, using the fact that G is lower semicontinuous and convex, and $\lim_{k \rightarrow \infty} \boldsymbol{\xi}^{k+1} = \mathbf{0}$, we obtain that

$$G(\mathbf{y}^\infty, \mathbf{0}) = G(\lim_{j \rightarrow \infty} \mathbf{y}^{k_j+1}, \lim_{j \rightarrow \infty} \boldsymbol{\xi}^{k_j+1}) \leq \liminf_{j \rightarrow \infty} G(\mathbf{y}^{k_j+1}, \boldsymbol{\xi}^{k_j+1}) = G(\mathbf{y}^*, \mathbf{0}).$$

This implies that \mathbf{y}^∞ is an optimal solution of problem (2.1). Similarly, using the upper semicontinuity of F and analogous manipulations, we can prove that any accumulation point of $\{\mathbf{x}^k\}$ is an optimal solution of problem (3.2). This proves statement (iv).

Statement (v). We next prove that the whole sequence $\{\mathbf{x}^k\}$ is convergent. Let

$$D_{\tau_k} \left((\mathbf{w}^k, \mathbf{y}^k), \mathcal{Y}^* \right) := \inf_{\mathbf{y}^* \in \mathcal{Y}^*} \left\{ \|\mathbf{w}^k - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2 \right\},$$

and define that

$$\phi := \liminf_{k \rightarrow \infty} D_{\tau_k} \left((\mathbf{w}^k, \mathbf{y}^k), \mathcal{Y}^* \right),$$

where \mathcal{Y}^* is the solution set of problem (2.1) (i.e., problem (1.2)). Since $\{\mathbf{w}^k\}$, $\{\mathbf{y}^k\}$ and $\{\tau_k\}$ are bounded, we see that $0 \leq \phi < \infty$ and there exists a subsequence $\{(\mathbf{w}^{k_j}, \mathbf{y}^{k_j}, \tau_{k_j})\}$ such that

$$\lim_{j \rightarrow \infty} D_{\tau_{k_j}} \left((\mathbf{w}^{k_j}, \mathbf{y}^{k_j}), \mathcal{Y}^* \right) = \phi.$$

Then, by passing to a further subsequence if necessary, we may also assume without loss of generality that the subsequence $\{\mathbf{x}^{k_j}\} \subseteq \{\mathbf{x}^k\}$ converges to some accumulation point \mathbf{x}^∞ , which, in view of statement (iv), belongs to \mathcal{X}^* (the optimal solution set of (3.2)). Thus, for such \mathbf{x}^∞ and any $\mathbf{y}^* \in \mathcal{Y}^*$, using (A.8) with some manipulations, we can obtain that, for all $k > k_j$,

$$\begin{aligned} & \|\mathbf{x}^k - \mathbf{x}^\infty\|^2 + \|\mathbf{w}^k - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2 \\ & \leq \left(\prod_{i=k_j}^{k-1} (1 + \nu_i) \right) \left(\|\mathbf{x}^{k_j} - \mathbf{x}^\infty\|^2 + \|\mathbf{w}^{k_j} - \mathbf{y}^*\|^2 + \tau_{k_j} \|\mathbf{y}^{k_j} - \mathbf{y}^*\|^2 \right). \end{aligned}$$

Since $0 \leq \phi \leq \liminf_{k \rightarrow \infty} \{\|\mathbf{w}^k - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2\}$ for any $\mathbf{y}^* \in \mathcal{Y}^*$, passing to the limit superior when $k \rightarrow \infty$ on the both sides of the above inequality, we obtain that, for any $\mathbf{y}^* \in \mathcal{Y}^*$,

$$\begin{aligned} & \phi + \limsup_{k \rightarrow \infty} \left\{ \|\mathbf{x}^k - \mathbf{x}^\infty\|^2 \right\} \\ & \leq \limsup_{k \rightarrow \infty} \left\{ \|\mathbf{x}^k - \mathbf{x}^\infty\|^2 + \|\mathbf{w}^k - \mathbf{y}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2 \right\} \\ & \leq \left(\prod_{i=k_j}^{\infty} (1 + \nu_i) \right) \left(\|\mathbf{x}^{k_j} - \mathbf{x}^\infty\|^2 + \|\mathbf{w}^{k_j} - \mathbf{y}^*\|^2 + \tau_{k_j} \|\mathbf{y}^{k_j} - \mathbf{y}^*\|^2 \right), \quad \forall j \geq 0. \end{aligned}$$

Taking the infimum in $\mathbf{y}^* \in \mathcal{Y}^*$ on the right-hand side of the last inequality, we have that

$$\begin{aligned} & \limsup_{k \rightarrow \infty} \left\{ \|\mathbf{x}^k - \mathbf{x}^\infty\|^2 \right\} \\ & \leq \left(\prod_{i=k_j}^{\infty} (1 + \nu_i) \right) \|\mathbf{x}^{k_j} - \mathbf{x}^\infty\|^2 + \left(\prod_{i=k_j}^{\infty} (1 + \nu_i) \right) D_{\tau_{k_j}} \left((\mathbf{w}^{k_j}, \mathbf{y}^{k_j}), \mathcal{Y} \right) - \phi, \quad \forall j \geq 0. \end{aligned}$$

Since $\ln \left(\prod_{i=k_j}^{\infty} (1 + \nu_i) \right) = \sum_{i=k_j}^{\infty} \ln(1 + \nu_i) \leq \sum_{i=k_j}^{\infty} \nu_i$ and $\lim_{j \rightarrow \infty} \sum_{i=k_j}^{\infty} \nu_i = 0$ (due to the summability of $\{\nu_k\}$), we see that $\lim_{j \rightarrow \infty} \prod_{i=k_j}^{\infty} (1 + \nu_i) = 1$. Using this fact, we can observe that the right-hand side of the above inequality converges to 0 as $j \rightarrow \infty$. Then, we conclude that $\lim_{k \rightarrow \infty} \mathbf{x}^k = \mathbf{x}^\infty$, which completes the proof. \square

A.2 Proof of Theorem 3.2

Proof. Statement (i). For the sake of clarity, we will present our proof in three steps.

Step I. Let $(\mathbf{y}^*, \mathbf{x}^*) \in \mathbb{R}^M \times \mathbb{R}^N$ be an arbitrary saddle point of ℓ . Similar to the proof of Theorem 3.1(i), we combine (A.2) with (A.4) to obtain that

$$\begin{aligned} & \|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 + \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^*\|^2 \\ & = \|\mathbf{x}^k - \mathbf{x}^*\|^2 + \tau_k \|\mathbf{y}^k - \mathbf{y}^*\|^2 - 2\sigma_k \left(\langle \mathbf{y}^{k+1} - \mathbf{y}^*, \boldsymbol{\theta}^{k+1} \rangle + \langle \mathbf{x}^{k+1} - \mathbf{x}^*, \boldsymbol{\xi}^{k+1} \rangle \right) \\ & \quad + 2\sigma_k \langle \mathbf{y}^{k+1} - \mathbf{y}^*, \Delta^{k+1} \rangle - \|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 - \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2. \end{aligned}$$

Since $(\mathbf{0}, \mathbf{0}) \in \partial \ell(\mathbf{y}^*, \mathbf{x}^*)$ and $(\boldsymbol{\theta}^{k+1}, \boldsymbol{\xi}^{k+1}) \in \partial \ell(\mathbf{y}^{k+1}, \mathbf{x}^{k+1})$, it then follows from the monotonicity of $\partial \ell$ that

$$\langle \mathbf{y}^{k+1} - \mathbf{y}^*, \boldsymbol{\theta}^{k+1} \rangle + \langle \mathbf{x}^{k+1} - \mathbf{x}^*, \boldsymbol{\xi}^{k+1} \rangle \geq 0.$$

Thus, we conclude that

$$\begin{aligned} & \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^k - \mathbf{y}^*)}{\mathbf{x}^k - \mathbf{x}^*} \right\|^2 - \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^*)}{\mathbf{x}^{k+1} - \mathbf{x}^*} \right\|^2 \\ & \geq \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k)}{\mathbf{x}^{k+1} - \mathbf{x}^k} \right\|^2 - 2\sigma_k \|\mathbf{y}^{k+1} - \mathbf{y}^*\| \|\Delta^{k+1}\|. \end{aligned} \quad (\text{A.13})$$

Define the sequences $\{\bar{\mathbf{y}}^k\} \subseteq \mathbb{R}^M$ and $\{\bar{\mathbf{x}}^k\} \subseteq \mathbb{R}^N$ as follows:

$$\bar{\mathbf{y}}^k := \Pi_{\mathcal{Y}^*}(\mathbf{y}^k) \quad \text{and} \quad \bar{\mathbf{x}}^k := \Pi_{\mathcal{X}^*}(\mathbf{x}^k), \quad \forall k \geq 0,$$

where \mathcal{Y}^* is the solution set of problem (2.1) (i.e., problem (1.2)), \mathcal{X}^* is the solution set of problem (3.2) (i.e., problem (1.1)), and $\Pi_{\mathcal{Y}^*}(\mathbf{y}^k)$ (resp. $\Pi_{\mathcal{X}^*}(\mathbf{x}^k)$) denotes the projection of \mathbf{y}^k (resp. \mathbf{x}^k) onto set \mathcal{Y}^* (resp. \mathcal{X}^*). Since (A.13) holds for any $\mathbf{y}^* \in \mathcal{Y}^*$ and $\mathbf{x}^* \in \mathcal{X}^*$, we can replace \mathbf{y}^* and \mathbf{x}^* with $\bar{\mathbf{y}}^k$ and $\bar{\mathbf{x}}^k$, respectively, to obtain that

$$\begin{aligned} & \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^k - \bar{\mathbf{y}}^k)}{\mathbf{x}^k - \bar{\mathbf{x}}^k} \right\|^2 - \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^{k+1} - \bar{\mathbf{y}}^k)}{\mathbf{x}^{k+1} - \bar{\mathbf{x}}^k} \right\|^2 \\ & \geq \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k)}{\mathbf{x}^{k+1} - \mathbf{x}^k} \right\|^2 - 2\sigma_k \|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^k\| \|\Delta^{k+1}\|. \end{aligned}$$

Moreover, from the definitions of $\bar{\mathbf{y}}^k$ and $\bar{\mathbf{x}}^k$, we have that $\|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}\| \leq \|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^k\|$ and $\|\mathbf{x}^{k+1} - \bar{\mathbf{x}}^{k+1}\| \leq \|\mathbf{x}^{k+1} - \bar{\mathbf{x}}^k\|$. These, together with the above inequality, yield that

$$\begin{aligned} & \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^k - \bar{\mathbf{y}}^k)}{\mathbf{x}^k - \bar{\mathbf{x}}^k} \right\|^2 - \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1})}{\mathbf{x}^{k+1} - \bar{\mathbf{x}}^{k+1}} \right\|^2 \\ & \geq \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k)}{\mathbf{x}^{k+1} - \mathbf{x}^k} \right\|^2 - 2\sigma_k \|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^k\| \|\Delta^{k+1}\|. \end{aligned} \quad (\text{A.14})$$

Step II. We next derive an upper bound for $\|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^k\| \|\Delta^{k+1}\|$. On the one hand, we have from (A.1b) that

$$\sigma_k^2 \|\Delta^{k+1}\|^2 \leq \rho \left(\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 + \tau_k \|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2 \right),$$

which implies that

$$\|\Delta^{k+1}\| \leq \frac{\sqrt{\rho}}{\sigma_k} \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k)}{\mathbf{x}^{k+1} - \mathbf{x}^k} \right\|. \quad (\text{A.15})$$

On the other hand, we see that

$$\begin{aligned} \|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^k\| & \leq \|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}\| + \|\bar{\mathbf{y}}^{k+1} - \bar{\mathbf{y}}^k\| \leq \|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}\| + \|\mathbf{y}^{k+1} - \mathbf{y}^k\| \\ & \leq \|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}\| + \frac{1}{\sqrt{\tau_k}} \left\| \frac{\sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k)}{\mathbf{x}^{k+1} - \mathbf{x}^k} \right\|, \end{aligned} \quad (\text{A.16})$$

where the second inequality follows from the non-expansiveness of the projection operator $\Pi_{\mathcal{Y}^*}(\cdot)$. Moreover, since $\{\mathbf{y}^k\}$ and $\{\mathbf{x}^k\}$ are bounded (by Theorem 3.1(i)), there must exist a positive scalar r such that

$$\text{dist}((\mathbf{y}^k, \mathbf{x}^k), (\partial\ell)^{-1}(\mathbf{0}, \mathbf{0})) \leq r, \quad \forall k \geq 0.$$

Thus, we apply Assumption A with this r and know that, there exists a $\kappa > 0$ such that

$$\text{dist}((\mathbf{y}^{k+1}, \mathbf{x}^{k+1}), \mathcal{Y}^* \times \mathcal{X}^*) \leq \kappa \text{dist}((\mathbf{0}, \mathbf{0}), \partial\ell(\mathbf{y}^{k+1}, \mathbf{x}^{k+1})) \leq \kappa \left\| (\boldsymbol{\theta}^{k+1}, \boldsymbol{\xi}^{k+1}) \right\|,$$

where the last inequality is due to (A.6) (i.e., $(\boldsymbol{\theta}^{k+1}, \boldsymbol{\xi}^{k+1}) \in \partial \ell(\mathbf{y}^{k+1}, \mathbf{x}^{k+1})$) with $\boldsymbol{\theta}^{k+1} := \Delta^{k+1} - \tau_k \sigma_k^{-1}(\mathbf{y}^{k+1} - \mathbf{y}^k)$ and $\boldsymbol{\xi}^{k+1} := \sigma_k^{-1}(\mathbf{x}^k - \mathbf{x}^{k+1})$. This inequality further implies that

$$\begin{aligned}
\|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}\| &\leq \sqrt{\|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}\|^2 + \|\mathbf{x}^{k+1} - \bar{\mathbf{x}}^{k+1}\|^2} \\
&\leq \kappa \left\| \begin{array}{c} \Delta^{k+1} - \tau_k \sigma_k^{-1}(\mathbf{y}^{k+1} - \mathbf{y}^k) \\ \sigma_k^{-1}(\mathbf{x}^k - \mathbf{x}^{k+1}) \end{array} \right\| \leq \kappa \left(\|\Delta^{k+1}\| + \frac{1}{\sigma_k} \left\| \begin{array}{c} \tau_k(\mathbf{y}^{k+1} - \mathbf{y}^k) \\ \mathbf{x}^{k+1} - \mathbf{x}^k \end{array} \right\| \right) \\
&\leq \kappa \left(\|\Delta^{k+1}\| + \frac{\sqrt{\bar{\tau}_k}}{\sigma_k} \left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k) \\ \mathbf{x}^{k+1} - \mathbf{x}^k \end{array} \right\| \right) \\
&\leq \frac{\kappa(\sqrt{\rho} + \sqrt{\bar{\tau}_k})}{\sigma_k} \left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k) \\ \mathbf{x}^{k+1} - \mathbf{x}^k \end{array} \right\|,
\end{aligned} \tag{A.17}$$

where $\bar{\tau}_k$ is defined as $\bar{\tau}_k := \max\{1, \tau_k\}$ and the last inequality follows from (A.15). Now, combing (A.15), (A.16) and (A.17), with some manipulations, we can obtain that

$$\|\mathbf{y}^{k+1} - \bar{\mathbf{y}}^k\| \|\Delta^{k+1}\| \leq \left(\frac{\kappa \sqrt{\tau_k} (\rho + \sqrt{\rho \bar{\tau}_k}) + \sigma_k \sqrt{\rho}}{\sigma_k^2 \sqrt{\tau_k}} \right) \left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k) \\ \mathbf{x}^{k+1} - \mathbf{x}^k \end{array} \right\|^2.$$

Then, substituting this inequality into (A.14) yields that

$$\begin{aligned}
&\left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^k - \bar{\mathbf{y}}^k) \\ \mathbf{x}^k - \bar{\mathbf{x}}^k \end{array} \right\|^2 - \left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}) \\ \mathbf{x}^{k+1} - \bar{\mathbf{x}}^{k+1} \end{array} \right\|^2 \\
&\geq \left(1 - \frac{2\kappa \sqrt{\tau_k} (\rho + \sqrt{\rho \bar{\tau}_k}) + 2\sigma_k \sqrt{\rho}}{\sigma_k \sqrt{\tau_k}} \right) \left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k) \\ \mathbf{x}^{k+1} - \mathbf{x}^k \end{array} \right\|^2.
\end{aligned} \tag{A.18}$$

Step III. In the following, we will establish the convergence rate based on (A.18). First, by recalling the conditions on $\{\tau_k\}$: $\tau_k \geq \tau_{\min} > 0$ and $\tau_{k+1} \leq (1 + \nu_k)\tau_k$ with $\nu_k \geq 0$ and $\sum_{k=0}^{\infty} \nu_k < +\infty$, we know that there exists $\tau_{\max} := (\prod_{k=0}^{\infty} (1 + \nu_k)) \tau_0$ such that $0 < \tau_{\min} \leq \tau_k \leq \tau_{\max} < +\infty$ for all $k \geq 0$. This together with condition (3.5) implies that there exists a positive integer k_0 such that

$$\sqrt{\tau_k} - 2\sqrt{\rho} > 0 \quad \text{and} \quad \sigma_k > c \cdot \frac{2\kappa \sqrt{\tau_k} (\rho + \sqrt{\rho \bar{\tau}_k})}{\sqrt{\tau_k} - 2\sqrt{\rho}}, \quad \forall k \geq k_0,$$

where $c > 1$. Hence, one can verify that

$$\left(1 - \frac{2\kappa \sqrt{\tau_k} (\rho + \sqrt{\rho \bar{\tau}_k}) + 2\sigma_k \sqrt{\rho}}{\sigma_k \sqrt{\tau_k}} \right) > \tilde{c} := \frac{c-1}{c} \cdot \frac{\sqrt{\tau_{\min}} - 2\sqrt{\rho}}{\sqrt{\tau_{\min}}} > 0, \quad \forall k \geq k_0, \tag{A.19}$$

which means that the factor in the right-hand side of (A.18) will be positive when $k \geq k_0$. On the other hand, using (A.17) again, we deduce that

$$\begin{aligned}
\left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k) \\ \mathbf{x}^{k+1} - \mathbf{x}^k \end{array} \right\|^2 &\geq \frac{\sigma_k^2}{\kappa^2 (\sqrt{\rho} + \sqrt{\bar{\tau}_k})^2} \left\| \begin{array}{c} \mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1} \\ \mathbf{x}^{k+1} - \bar{\mathbf{x}}^{k+1} \end{array} \right\|^2 \\
&\geq \frac{\sigma_k^2}{\kappa^2 (\sqrt{\rho} + \sqrt{\bar{\tau}_k})^2 \bar{\tau}_k} \left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}) \\ \mathbf{x}^{k+1} - \bar{\mathbf{x}}^{k+1} \end{array} \right\|^2.
\end{aligned}$$

This, together with (A.18) and (A.19), yields that

$$\left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^k - \bar{\mathbf{y}}^k) \\ \mathbf{x}^k - \bar{\mathbf{x}}^k \end{array} \right\|^2 \geq (1 + \gamma_k) \left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}) \\ \mathbf{x}^{k+1} - \bar{\mathbf{x}}^{k+1} \end{array} \right\|^2, \quad \forall k \geq k_0, \tag{A.20}$$

where

$$\begin{aligned}
\gamma_k &:= \left(1 - \frac{2\kappa\sqrt{\tau_k}(\rho + \sqrt{\rho\bar{\tau}_k}) + 2\sigma_k\sqrt{\rho}}{\sigma_k\sqrt{\tau_k}}\right) \frac{\sigma_k^2}{\kappa^2(\sqrt{\rho} + \sqrt{\bar{\tau}_k})^2\bar{\tau}_k} \\
&\geq \tilde{c} \cdot \frac{\sigma_k^2}{\kappa^2(\sqrt{\rho} + \sqrt{\bar{\tau}_{\max}})^2\bar{\tau}_{\max}} \\
&\geq \gamma_{\min} := \frac{\tilde{c}\sigma_{\min}^2}{\kappa^2(\sqrt{\rho} + \sqrt{\bar{\tau}_{\max}})^2\bar{\tau}_{\max}} > 0, \quad \forall k \geq k_0.
\end{aligned} \tag{A.21}$$

Let $\Lambda^k := \text{Diag}(\tau_k I_M, I_N)$. Since $\{\tau_k\}$ is bounded away from 0 and satisfies that $(1 + \nu_k)\tau_k \geq \tau_{k+1}$, we have that $(1 + \nu_k)\Lambda^k \succeq \Lambda^{k+1} \succ 0$. Then, one can obtain from (A.20) that

$$(1 + \nu_k) \left\| \begin{array}{c} \mathbf{y}^k - \bar{\mathbf{y}}^k \\ \mathbf{x}^k - \bar{\mathbf{x}}^k \end{array} \right\|_{\Lambda^k}^2 \geq (1 + \gamma_k) \left\| \begin{array}{c} \mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1} \\ \mathbf{x}^{k+1} - \bar{\mathbf{x}}^{k+1} \end{array} \right\|_{\Lambda^{k+1}}^2,$$

which readily implies that

$$\text{dist}_{\Lambda^{k+1}} \left((\mathbf{y}^{k+1}, \mathbf{x}^{k+1}), (\partial\ell)^{-1}(\mathbf{0}, \mathbf{0}) \right) \leq \mu_k \text{dist}_{\Lambda^k} \left((\mathbf{y}^k, \mathbf{x}^k), (\partial\ell)^{-1}(\mathbf{0}, \mathbf{0}) \right),$$

where $\mu_k := \sqrt{\frac{1+\nu_k}{1+\gamma_k}}$. Since $\nu_k \rightarrow 0$ and $\gamma_k \geq \gamma_{\min} > 0$ for all $k > k_0$, one can verify that $\limsup_{k \rightarrow \infty} \{\mu_k\} < 1$. Thus, we obtain the desired results in statement (i).

Statement (ii). Using (A.18) and (A.19) again, we see that

$$\tilde{c}\tau_{\min} \|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2 \leq \tilde{c} \left\| \begin{array}{c} \sqrt{\tau_k}(\mathbf{y}^{k+1} - \mathbf{y}^k) \\ \mathbf{x}^{k+1} - \mathbf{x}^k \end{array} \right\|^2 \leq \text{dist}_{\Lambda^k} \left((\mathbf{y}^k, \mathbf{x}^k), (\partial\ell)^{-1}(\mathbf{0}, \mathbf{0}) \right),$$

for any $k \geq k_0$. Using this inequality and the fact that the sequence $\{\text{dist}_{\Lambda^k}((\mathbf{y}^k, \mathbf{x}^k), (\partial\ell)^{-1}(\mathbf{0}, \mathbf{0}))\}$ is asymptotically Q-(super)linear convergent, we can conclude that there exist a positive integer k_1 , $0 < \beta < 1$ and $C > 0$ such that

$$\|\mathbf{y}^{k+1} - \mathbf{y}^k\| \leq C\beta^k, \quad \forall k \geq k_1,$$

which further implies that $\sum_{k=0}^{\infty} \|\mathbf{y}^{k+1} - \mathbf{y}^k\| < \infty$. Consequently, $\{\mathbf{y}^k\}$ is a Cauchy sequence and hence convergent. Therefore, the proof is completed. \square

B A dual ADMM for QROTs

In this section, we present the alternating direction method of multipliers (ADMM, see, e.g. [5, 14]) for solving the dual problem (4.3), which can be reformulated as:

$$\min_{\mathbf{u} \in \mathbb{R}^m, \mathbf{v} \in \mathbb{R}^n, W \in \mathbb{R}^{m \times n}} \left\{ f_{\text{q}}^*(W) - \boldsymbol{\alpha}^\top \mathbf{u} - \boldsymbol{\beta}^\top \mathbf{v} \mid \mathbf{u} \mathbf{1}_n^\top + \mathbf{1}_m \mathbf{v}^\top = W \right\}. \tag{B.1}$$

Given a penalty parameter $\sigma > 0$, the augmented Lagrangian function of (B.1) is

$$\begin{aligned}
&\mathcal{L}_\sigma(\mathbf{u}, \mathbf{v}, W; X) \\
&:= f_{\text{q}}^*(W) - \boldsymbol{\alpha}^\top \mathbf{u} - \boldsymbol{\beta}^\top \mathbf{v} + \left\langle X, \mathbf{u} \mathbf{1}_n^\top + \mathbf{1}_m \mathbf{v}^\top - W \right\rangle + \frac{\sigma}{2} \left\| \mathbf{u} \mathbf{1}_n^\top + \mathbf{1}_m \mathbf{v}^\top - W \right\|_F^2.
\end{aligned}$$

Then, the ADMM for solving (B.1) can be described as in **Algorithm 3**.

Algorithm 3 ADMM for solving (B.1)

Input: a penalty parameter $\sigma > 0$, and initializations $\mathbf{u}^0 \in \mathbb{R}^m$, $\mathbf{v}^0 \in \mathbb{R}^n$, $W^0, X^0 \in \mathbb{R}^{m \times n}$. Set $k = 0$.

while the termination criterion is not met, **do**

Step 1. Compute

$$(\mathbf{u}^{k+1}, \mathbf{v}^{k+1}) = \arg \min_{\mathbf{u}, \mathbf{v}} \mathcal{L}_\sigma(\mathbf{u}, \mathbf{v}, W^k; X^k).$$

Step 2. Compute

$$W^{k+1} = \arg \min_W \mathcal{L}_\sigma(\mathbf{u}^{k+1}, \mathbf{v}^{k+1}, W; X^k).$$

Step 3. Set $X^{k+1} = X^k + \tau\sigma (\mathbf{u}^{k+1} \mathbf{1}_n^\top + \mathbf{1}_m (\mathbf{v}^{k+1})^\top - W^{k+1})$, where $\tau \in (0, \frac{1+\sqrt{5}}{2})$ is the dual step-size that is typically set to 1.618.

Step 4. Set $k \leftarrow k + 1$ and go to **Step 1**.

end while

Output: $(\mathbf{u}^k, \mathbf{v}^k, W^k, X^k)$.

Both subproblems in ADMM can be solved efficiently. Specifically, $(\mathbf{u}^{k+1}, \mathbf{v}^{k+1})$ can be obtained by solving the following unconstrained convex minimization problem:

$$\min_{\mathbf{u}, \mathbf{v}} h_k(\mathbf{u}, \mathbf{v}) := -\boldsymbol{\alpha}^\top \mathbf{u} - \boldsymbol{\beta}^\top \mathbf{v} + \frac{\sigma}{2} \left\| \mathbf{u} \mathbf{1}_n^\top + \mathbf{1}_m \mathbf{v}^\top - S^k \right\|_F^2, \quad (\text{B.2})$$

where $S^k := W^k - \sigma^{-1} X^k$. From the first-order optimality conditions of (B.2), we see that solving problem (B.2) is equivalent to solving the equation $\nabla h_k(\mathbf{u}, \mathbf{v}) = \mathbf{0}$. This, in turn, reduces to solving the following linear system

$$\begin{cases} n\mathbf{u} + (\mathbf{1}_n^\top \mathbf{v}) \mathbf{1}_m = \sigma^{-1} \boldsymbol{\alpha} + S^k \mathbf{1}_n, & (\text{B.3a}) \\ (\mathbf{1}_m^\top \mathbf{u}) \mathbf{1}_n + m\mathbf{v} = \sigma^{-1} \boldsymbol{\beta} + (S^k)^\top \mathbf{1}_m. & (\text{B.3b}) \end{cases}$$

With some algebraic manipulations, it is not difficult to show that

$$\begin{aligned} \mathbf{u}^*(t) &= \frac{\sigma^{-1} \boldsymbol{\alpha} + S^k \mathbf{1}_n}{n} + t \mathbf{1}_m, \quad \forall t \in \mathbb{R}, \\ \mathbf{v}^*(t) &= \frac{\sigma^{-1} \boldsymbol{\beta} + (S^k)^\top \mathbf{1}_m}{m} - \frac{\mathbf{1}_m^\top \mathbf{u}^*(t)}{m} \mathbf{1}_n, \quad \forall t \in \mathbb{R}. \end{aligned}$$

solves the above linear system. Thus, we obtain a solution pair $(\mathbf{u}^*(t), \mathbf{v}^*(t))$ with any $t \in \mathbb{R}$. It can be routinely shown that $(\mathbf{u}^*(t), \mathbf{v}^*(t))$ satisfies the linear system (B.3a) and (B.3b), and therefore solves problem (B.2). On the other hand, W^{k+1} can be obtained by computing the proximal operator of the function $\sigma^{-1} f_q^*$, i.e.,

$$W^{k+1} := \text{prox}_{\sigma^{-1} f_q^*}(Q^k) = \begin{cases} C - \Pi_{\mathbb{R}_+^{m \times n}}(C - Q^k), & \lambda = 0, \\ Q^k - (1 + \lambda\sigma)^{-1} \Pi_{\mathbb{R}_+^{m \times n}}(Q^k - C), & \lambda > 0, \end{cases}$$

where $Q^k := \mathbf{u}^{k+1} \mathbf{1}_n^\top + \mathbf{1}_m (\mathbf{v}^{k+1})^\top + \sigma^{-1} X^k$.

C An iBPGM for QROTs

In this section, we briefly discuss how to employ an inexact Bregman proximal gradient method (iBPGM) with Sinkhorn's algorithm as a subsolver for solving the QROT problem

(4.1). We refer readers to [52, Section 5] for more details. Specifically, the iBPGM with the entropy kernel function for solving (4.1) can be given as follows: let $X^0 > 0$ and $\phi(X) := \sum_{ij} x_{ij}(\log x_{ij} - 1)$, at the k -th iteration, compute

$$X^{k+1} \approx \min_X \left\{ \langle C + \lambda X^k, X \rangle + \mu_k \mathcal{D}_\phi(X, X^k) \mid X \mathbf{1}_n = \boldsymbol{\alpha}, X^\top \mathbf{1}_m = \boldsymbol{\beta} \right\}, \quad (\text{C.1})$$

where $\mu_k \geq \lambda$ is a positive proximal parameter, and $\mathcal{D}_\phi(U, V)$ denotes the Bregman distance between U and V associated with the kernel function ϕ which is defined as $\mathcal{D}_\phi(U, V) := \phi(U) - \phi(V) - \langle \nabla \phi(V), U - V \rangle$. Problem (C.1) can be rewritten as

$$\min_X \left\{ \langle M^k, X \rangle + \mu_k \sum_{ij} x_{ij}(\log x_{ij} - 1) \mid X \mathbf{1}_n = \boldsymbol{\alpha}, X^\top \mathbf{1}_m = \boldsymbol{\beta} \right\}, \quad (\text{C.2})$$

where $M^k := C + \lambda X^k - \mu_k \log X^k$. Note that problem (C.2) has the same form as the entropic regularized optimal transport problem and hence can be readily solved by the popular Sinkhorn's algorithm; see [31, Section 4.2] for more details. Specifically, let $\Xi^k := e^{-M^k/\mu_k}$. Then, given an initial positive vector $\mathbf{v}^{k,0}$, the iterative scheme is given by

$$\mathbf{u}^{k,t} = \boldsymbol{\alpha} ./ (\Xi^k \mathbf{v}^{k,t-1}), \quad \mathbf{v}^{k,t} = \boldsymbol{\beta} ./ ((\Xi^k)^\top \mathbf{u}^{k,t}), \quad \forall t \geq 0, \quad (\text{C.3})$$

where './' denotes the entrywise division between two vectors. When a pair $(\mathbf{u}^{k,t}, \mathbf{v}^{k,t})$ is obtained based on a certain stopping criterion, an approximate solution of (C.2) (and hence (C.1)) can be recovered by setting $X^{k,t} := \text{Diag}(\mathbf{u}^{k,t}) \Xi^k \text{Diag}(\mathbf{v}^{k,t})$. Meanwhile, a pair of approximate dual solutions can be recovered by setting $\mathbf{f}^{k,t} := \mu_k \log \mathbf{u}^{k,t}$ and $\mathbf{g}^{k,t} := \mu_k \log \mathbf{v}^{k,t}$. In our experiments, we simply execute Sinkhorn's iteration (C.3) only *once* for each subproblem, and observe that this is sufficient for obtaining a promising initial point for warm-starting our ripALM.

References

- [1] M.M. Alves and B.F. Svaiter. A note on Fejér-monotone sequences in product spaces and its applications to the dual convergence of augmented Lagrangian methods. *Mathematical Programming*, 155(1):613–616, 2016.
- [2] H.H. Bauschke and P.L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, New York, 2nd edition, 2017.
- [3] Y. Bello-Cruz, M.L.N. Gonçalves, and N. Krislock. On FISTA with a relative error rule. *Computational Optimization and Applications*, 84(2):295–318, 2023.
- [4] M. Blondel, V. Seguy, and A. Rolet. Smooth and sparse optimal transport. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84, pages 880–889, Lanzarote, Spain, 2018. PMLR.
- [5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.
- [6] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2004.
- [7] Y. Cui and J.-S. Pang. *Modern Nonconvex Nondifferentiable Optimization*. MOS-SIAM Series on Optimization. SIAM, Philadelphia, 2021.

- [8] Y. Cui, D.F. Sun, and K.-C. Toh. On the R-superlinear convergence of the KKT residuals generated by the augmented Lagrangian method for convex composite conic programming. *Mathematical Programming*, 178(1):381–415, 2019.
- [9] A. De Marchi. *Augmented Lagrangian and Proximal Methods for Constrained Structured Optimization*. PhD thesis, Universität der Bundeswehr München, 2021.
- [10] J. Eckstein and P.J.S. Silva. Proximal methods for nonlinear programming: double regularization and inexact subproblems. *Computational Optimization and Applications*, 46(2):279–304, 2010.
- [11] J. Eckstein and P.J.S. Silva. A practical relative error criterion for augmented Lagrangians. *Mathematical Programming*, 141(1):319–348, 2013.
- [12] J. Eckstein and W. Yao. Relative-error approximate versions of Douglas–Rachford splitting and special cases of the ADMM. *Mathematical Programming*, 170(2):417–444, 2018.
- [13] M. Essid and J. Solomon. Quadratically regularized optimal transport on graphs. *SIAM Journal on Scientific Computing*, 40(4):A1961–A1986, 2018.
- [14] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17–40, 1976.
- [15] B. Hermans, A. Themelis, and P. Patrinos. QPALM: a Newton-type proximal augmented Lagrangian method for quadratic programs. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 4325–4330. IEEE, 2019.
- [16] M.R. Hestenes. Multiplier and gradient methods. *Journal of Optimization Theory and Applications*, 4(5):303–320, 1969.
- [17] C. Humes Jr., P.J.S. Silva, and B.F. Svaiter. Some inexact hybrid proximal augmented Lagrangian algorithms. *Numerical Algorithms*, 35:175–184, 2004.
- [18] C. Li, W. Yin, H. Jiang, and Y. Zhang. An efficient augmented Lagrangian method with applications to total variation minimization. *Computational Optimization and Applications*, 56(3):507–530, 2013.
- [19] X. Li, D.F. Sun, and K.-C. Toh. A highly efficient semismooth Newton augmented Lagrangian method for solving Lasso problems. *SIAM Journal on Optimization*, 28(1):433–458, 2018.
- [20] X. Li, D.F. Sun, and K.-C. Toh. QSDPNAL: a two-phase augmented Lagrangian method for convex quadratic semidefinite programming. *Mathematical Programming Computation*, 10(4):703–743, 2018.
- [21] X. Li, D.F. Sun, and K.-C. Toh. An asymptotically superlinearly convergent semismooth Newton augmented Lagrangian method for linear programming. *SIAM Journal on Optimization*, 30(3):2410–2440, 2020.
- [22] X. Li, D.F. Sun, and K.-C. Toh. On the efficient computation of a generalized Jacobian of the projector over the Birkhoff polytope. *Mathematical Programming*, 179(1):419–446, 2020.

- [23] L. Liang, X. Li, D.F. Sun, and K.-C. Toh. QPPAL: a two-phase proximal augmented Lagrangian method for high-dimensional convex quadratic programming problems. *ACM Transactions on Mathematical Software (TOMS)*, 48(3):1–27, 2022.
- [24] L. Liang, D.F. Sun, and K.-C. Toh. An inexact augmented Lagrangian method for second-order cone programming with applications. *SIAM Journal on Optimization*, 31(3):1748–1773, 2021.
- [25] M.X. Lin, Y.-J. Liu, D.F. Sun, and K.-C. Toh. Efficient sparse semismooth Newton methods for the clustered Lasso problem. *SIAM Journal on Optimization*, 29(3):2026–2052, 2019.
- [26] M.X. Lin, D.F. Sun, and K.-C. Toh. An augmented Lagrangian method with constraint generation for shape-constrained convex regression problems. *Mathematical Programming Computation*, 14(2):223–270, 2022.
- [27] Y.-F. Liu, X. Liu, and S. Ma. On the nonergodic convergence rate of an inexact augmented Lagrangian framework for composite convex programming. *Mathematics of Operations Research*, 44(2):632–650, 2019.
- [28] D.A. Lorenz, P. Manns, and C. Meyer. Quadratically regularized optimal transport. *Applied Mathematics & Optimization*, 83(3):1919–1949, 2021.
- [29] F.J. Luque. Asymptotic convergence analysis of the proximal point algorithm. *SIAM Journal on Control and Optimization*, 22(2):277–293, 1984.
- [30] L.A. Parente, P.A. Lotito, and M.V. Solodov. A class of inexact variable metric proximal point algorithms. *SIAM Journal on Optimization*, 19(1):240–260, 2008.
- [31] G. Peyré and M. Cuturi. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.
- [32] B.T. Polyak. *Introduction to Optimization*. Optimization Software Inc., New York, 1987.
- [33] S. Pougkakiotis, J. Gondzio, and D. Kalogieras. An efficient active-set method with applications to sparse approximations and risk minimization. *arXiv preprint arXiv:2405.04172*, 2024.
- [34] M.J.D. Powell. A method for nonlinear constraints in minimization problems. *Optimization*, pages 283–298, 1969.
- [35] L. Qi and J. Sun. A nonsmooth version of Newton’s method. *Mathematical Programming*, 58(1):353–367, 1993.
- [36] R.T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, 1970.
- [37] R.T. Rockafellar. *Conjugate Duality and Optimization*. SIAM, Philadelphia, 1974.
- [38] R.T. Rockafellar. Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Mathematics of Operations Research*, 1(2):97–116, 1976.
- [39] R.T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM Journal on Control and Optimization*, 14(5):877–898, 1976.

- [40] R.T. Rockafellar and R.J.-B. Wets. *Variational Analysis*. Springer Berlin, Heidelberg, 1998.
- [41] J. Schrieber, D. Schuhmacher, and C. Gottschlich. DOTmark—A benchmark for discrete optimal transport. *IEEE Access*, 5:271–282, 2017.
- [42] M.V. Solodov and B.F. Svaiter. A hybrid approximate extragradient–proximal point algorithm using the enlargement of a maximal monotone operator. *Set-Valued Analysis*, 7(4):323–345, 1999.
- [43] M.V. Solodov and B.F. Svaiter. A hybrid projection-proximal point algorithm. *Journal of Convex Analysis*, 6(1):59–70, 1999.
- [44] M.V. Solodov and B.F. Svaiter. Error bounds for proximal point subproblems and associated inexact proximal point algorithms. *Mathematical Programming*, 88(2):371–389, 2000.
- [45] M.V. Solodov and B.F. Svaiter. An inexact hybrid generalized proximal point algorithm and some new results on the theory of Bregman functions. *Mathematics of Operations Research*, 25(2):214–230, 2000.
- [46] M.V. Solodov and B.F. Svaiter. A unified framework for some inexact proximal point algorithms. *Numerical Functional Analysis and Optimization*, 22(7-8):1013–1035, 2001.
- [47] D.F. Sun, K.-C. Toh, and Y.C. Yuan. Convex clustering: Model, theoretical guarantee and efficient algorithm. *Journal of Machine Learning Research*, 22(9):1–32, 2021.
- [48] Y. Yan and Q. Li. An efficient augmented Lagrangian method for support vector machine. *Optimization Methods and Software*, 35(4):855–883, 2020.
- [49] L. Yang, J.J. Hu, and K.-C. Toh. An inexact Bregman proximal difference-of-convex algorithm with two types of relative stopping criteria. *arXiv preprint arXiv:2406.04646*, 2024.
- [50] L. Yang, J. Li, D.F. Sun, and K.-C. Toh. A fast globally linearly convergent algorithm for the computation of Wasserstein barycenters. *Journal of Machine Learning Research*, 22(21):1–37, 2021.
- [51] L. Yang, L. Liang, H.T.M. Chu, and K.-C. Toh. A corrected inexact proximal augmented Lagrangian method with a relative error criterion for a class of group-quadratic regularized optimal transport problems. *Journal of Scientific Computing*, 99(3):79, 2024.
- [52] L. Yang and K.-C. Toh. Inexact Bregman proximal gradient method and its inertial variant with absolute and relative stopping criteria. *arXiv preprint arXiv:2109.05690*, 2023.
- [53] Y.J. Zhang, N. Zhang, D.F. Sun, and K.-C. Toh. An efficient Hessian based algorithm for solving large-scale sparse group Lasso problems. *Mathematical Programming*, 179(1):223–263, 2020.
- [54] X.-Y. Zhao and L. Chen. The linear and asymptotically superlinear convergence rates of the augmented Lagrangian method with a practical relative error criterion. *Asia-Pacific Journal of Operational Research*, 37(04):2040001, 2020.
- [55] X.-Y. Zhao, D.F. Sun, and K.-C. Toh. A Newton-CG augmented Lagrangian method for semidefinite programming. *SIAM Journal on Optimization*, 20(4):1737–1765, 2010.