J. Nonlinear Var. Anal. 8 (2024), No. 6, pp. 883-907
Available online at http://jnva.biemdas.com
https://doi.org/10.23952/jnva.8.2024.6.04

A DISTRIBUTED PRIMAL-DUAL HYBRID GRADIENT ALGORITHM FOR FAIR RESOURCE ALLOCATION

HONGMEI CHEN[1], XINGYU LU[2,*], ZENGYUN SHAN[1], JUNFENG YANG[1,*], JUN ZHOU[2]

[1]*School of Mathematics, Nanjing University, Nanjing, China*
[2]*Ant Group, Hangzhou, China*

Abstract. Fair resource allocation, which is vital in various fields, including advertising, cloud computing, and loan management, requires a delicate balance between maximizing revenue and ensuring fairness. This problem is commonly defined as constrained and regularized convex programming, where the objective function consists of a linear allocation cost and a nonseparable and nonlinear fairness regularizer. However, solving this problem for large cases, such as those with millions of variables, can be challenging and requires advanced computational methods and expertise. To address this issue, this paper proposes a distributed primal-dual hybrid gradient algorithm by using a tailored Bregman distance to solve a saddle point reformulation of the problem. Our algorithm allows for closed-form solutions to all subproblems and primarily employs matrix-vector multiplications, which can be efficiently executed via distributed parallel computations. Theoretical results demonstrate global iterate convergence and ergodic sublinear convergence rate under a practical stepsize condition. Furthermore, the proposed algorithm is shown to be more efficient and superior to off-the-shelf solvers such as the IPOPT and Gurobi, as evidenced by experimental results on synthetic and real-world industrial datasets.

Keywords. Bregman distance; Distributed computation; Large-scale; Primal-dual hybrid gradient; Resource allocation problem.

1. INTRODUCTION

Resource allocation problems (RAPs) are widely applicable across various fields such as advertising allocation [49], internal cloud computing [38], loan management [41], e-commerce [47], and many others. Extensive research has been conducted on RAPs over the past few decades, aiming to formulate optimization problems that maximize allocation revenues while satisfying resource constraints. For example, in advertising allocation, the platform must assign tailored advertisements to each user to maximize the overall revenue while staying within advertising budgets. In internal cloud computing, the system must allocate suitable execution machines for each computing task to enhance the overall execution efficiency.

Fairness is becoming increasingly important in the study of RAPs due to concerns of discrimination, bias, and inequality that may arise from resource allocation decisions. In internet

*Corresponding author.
E-mail address: hmchen0971@163.com (H. Chen), sing.lxy@antgroup.com (X. Lu), zyshan@smail.nju.edu.cn (Z. Shan), jfyang@nju.edu.cn (J. Yang), jun.zhoujun@antgroup.com (J. Zhou).
Received 12 March 2024; Accepted 26 April 2024; Published online 25 October 2024.

advertising [19], for instance, it is crucial to ensure that each advertiser's budget consumption is not too low when optimizing revenue. Similarly, in machine resource scheduling [1], network load balancing must be taken into account, and in portfolio optimization [25], risks arising from asset allocation should be evenly controlled. To incorporate fairness into resource allocation, [4] investigated regularized online allocation problems and proposed various regularizers for fairness. Additional regularization functions can be found in [24, 30, 39]. However, these regularizers are mostly nonseparable and nonsmooth, which entails significant optimization challenges for existing algorithms and solvers, especially those designed for additively separable objectives.

This paper investigates a general fair resource allocation problem that aims to minimize allocation costs and a fairness term with resource constraints. Let $\mathbb{I} := \{1,\dots,I\}$ denote a set of users and $\mathbb{J} := \{1,\dots,J\}$ denote a set of items. We define $\mathbb{T} := \{1,\dots,T\}$ as a set of indices relevant to the fairness vector. For each user i, we define the allocation strategy $\mathbf{x}_i := (x_{i,1}, x_{i,2}, \dots, x_{i,J})^\top \in \mathbb{R}^J$, where $x_{i,j}$ denotes the probability of allocating user i to item j. Denote the total number of resource types by K and the resource budget by $\mathbf{b} \in \mathbb{R}^K$. For each user $i \in \mathbb{I}$, let $\mathbf{c}_i \in \mathbb{R}^J$ and $M_i \in \mathbb{R}^{J\times K}$ be its the cost vector and the resource consumption matrix. To characterize the fairness of the allocation, we let $h : \mathbb{R}^T \to \mathbb{R}$ be a closed, proper, nonseparable convex fairness promoting regularizer. For each user $i \in \mathbb{I}$, we introduce a sequence of fairness weighting vector $\{\mathbf{r}_i^t\}_{t\in\mathbb{T}}$, where $\mathbf{r}_i^t = (r_{i,1}^t, r_{i,2}^t, \dots, r_{i,J}^t)^\top \in \mathbb{R}^J$. Let $\mathbf{p} := (p_1, p_2, \dots, p_T)^\top \in \mathbb{R}^T$ denote a bias term of the fairness vector and $\Delta := \{\mathbf{u} \in \mathbb{R}_+^J : \sum_{j=1}^J u_j = 1\}$ represent the unit simplex. Under the above definitions, we formulate the fair RAP in the following form

$$\min_{\{\mathbf{x}_i\in\Delta\}_{i\in\mathbb{I}}} \left\{ \alpha \sum_{i\in\mathbb{I}} \mathbf{c}_i^\top \mathbf{x}_i + \beta h\left(\left\{ \sum_{i\in\mathbb{I}} (\mathbf{r}_i^t)^\top \mathbf{x}_i - p_t \right\}_{t\in\mathbb{T}} \right) \text{ s.t. } \sum_{i\in\mathbb{I}} M_i^\top \mathbf{x}_i \le \mathbf{b} \right\}, \tag{1.1}$$

where $\alpha, \beta \ge 0$ are hyperparameters that balance the two objective terms. When $\alpha = 0$, model (1.1) disregards allocation costs and solely concerns the fairness of allocation. The financial asset allocation problem (5.1) is one particular instance of this. When $\beta = 0$, (1.1) reduces to the conventional resource allocation model. It's worth noting that various allocation problems fall under the umbrella of (1.1), such as allocation problems with regularizers applied to total resource consumption [4], packing proportional fairness problems [15], generalized bipartite matching problems [2], and robust allocation with diversity [42], among others. However, applying existing algorithms straightforwardly to solve problem (1.1) may not fully utilize the problem structures, resulting in lower computational efficiency.

To simplify the problem, one may regard (1.1) as a bipartite graph assignment problem, where $\mathbb{I}$ represents the set of nodes on one side of the graph and $\mathbb{J}$ contains nodes on the other side. Each node in $\mathbb{I}$ must be assigned to one node in $\mathbb{J}$. In many real-world scenarios, the cardinality of $\mathbb{I}$ (i.e., the number of elements in $\mathbb{I}$) is much larger than that of $\mathbb{J}$. For instance, in ads allocation, $\mathbb{I}$ and $\mathbb{J}$ correspond to impressions and advertisers, respectively, and the number of impressions usually far exceeds the number of advertisers. Similarly, in machine resource scheduling, $\mathbb{I}$ and $\mathbb{J}$ refer to instances/jobs and machines, where the number of jobs is often much larger than the number of machines. Therefore, we assume throughout this paper that $|\mathbb{I}| \gg |\mathbb{J}|$. Furthermore, in industrial applications, the number of variables in (1.1) (roughly $|\mathbb{I}| \cdot |\mathbb{J}|$) can reach millions or even billions.

Obtaining an optimal solution for (1.1) within a reasonable timeframe poses a significant challenge due to two factors. Firstly, the generic nonlinear fairness regularizer in (1.1) makes the problem highly complex. Classical methods for RAP without fairness regularizer, such as the dual methods including dual gradient [6] and dual subgradient method [26, 36], permit a distributed manner. However, in our problem setting, these dual methods are not available due to the challenge of evaluating the subgradient. Secondly, in real-world scenarios, the number of variables in (1.1) can be immense, reaching millions or even billions. The nonseparable fairness term makes problem (1.1) unsolvable by most existing optimization algorithms and solvers. Even when available solvers are applicable, their computational complexities are often extremely high for large-scale problems.

In this paper, we present a distributed primal-dual hybrid gradient algorithm with Bregman distance (B-PDHG) tailored for unit simplex constraints, based on the saddle point problem corresponding to (1.1). The Bregman distance, generated by the negative entropy, provides a significant advantage by enabling closed-form solutions for the $\{\mathbf{x}_i : i \in \mathbb{I}\}$-subproblems. The contributions of this paper are summarized as follows.

(1) We leverage a minimax problem formulation to transform the RAP problem in (1.1) and design a customized B-PDHG algorithm that has closed-form solutions for all subproblems and can be computed efficiently.
(2) Drawing from a refined convergence analysis, we propose a practical stepsize condition that relies solely on the norms of each M_i and $\mathbf{r}_i^t$, facilitating local computation. Under this stepsize condition, global iterate convergence and an ergodic $\mathcal{O}(1/N)$ sublinear convergence rate in terms of the primal-dual gap are guaranteed, where N is the iteration number.
(3) The proposed algorithm, with its separable nature and reliance solely on matrix-vector multiplications, is highly conducive to distributed computations. These features are essential in tackling the challenges posed by exceedingly large-scale problems.
(4) Experimental results on synthetic and real-world industrial datasets demonstrate that B-PDHG outperforms off-the-shelf solvers such as IPOPT and Gurobi regarding efficiency and superiority.

The remainder of this paper is organized as follows: In Section 2, we review some related work on RAPs. Section 3 provides a summary of the preliminaries and additional notation. The algorithm and associated theoretical results are developed in Section 4, followed by numerical results on both synthetic and real-world datasets in Section 5. Finally, we draw some concluding remarks in Section 6.

2. Related Work

2.1. **Fairness in RAPs.** Extensive studies have been conducted in the literature on RAPs with various and reasonable fairness metrics. In particular, the price of fairness in RAPs was investigated using metrics such as max-min fairness and proportional fairness [8]. Dominant resource fairness, introduced by [21], extends the metric of max-min fairness to address the allocation of multiple resources effectively.

In [48], a comprehensive study was conducted on a proportional-fair RAP in user-centric networks. In resource scheduling [30], individual fairness was incorporated using maximin share and envy-freeness metrics. However, our model includes a general regularization function

that captures fairness considerations in a broader sense. In [4], the authors studied a regularized online RAP, where the regularizer acts on the total resource consumption to promote fairness. In contrast, our emphasis is on the offline setting, and we propose an efficient algorithm that can scale effectively to handle large-scale datasets.

2.2. **Approaches for RAPs.** Several approaches have been proposed for online RAPs, such as [3, 4, 31, 33, 50], which aim to optimize the regret based on limited real-time information rather than focusing on resolving subproblems. In contrast, offline RAP is typically a large-scale optimization problem, and efficient resolution of its subproblems is crucial for an optimal allocation. Early approaches for offline RAP, such as search methods, relaxation methods, and pegging methods [29, 36, 37] may exhibit inefficient performance due to their inability to fully exploit the problem structure, mainly when dealing with large-scale cases. The authors of [2] proposed a distributed proportional allocation algorithm for the separable bipartite matching problem. However, due to the introduction of a regularizer, the optimization problem in (1.1) becomes nonseparable. The dual (sub)gradient methods [6, 26, 36] utilized for solving (1.1) entail addressing a complex optimization problem to obtain the (sub)gradient. As a primal-dual approach, the augmented Lagrangian method(ALM) [43] and alternating direction method of multipliers (ADMM) [9, 20] can be adopted to solve (1.1). Yet, they require solving large-scale linear systems. In addition, the separable structure w.r.t. $\mathbf{x}_i$'s will be destroyed by a quadratic penalty term in the augmented Lagrangian function. As a result, parallel computations are disabled.

2.3. **PDHG.** PDHG algorithm is a widely utilized scheme that decomposes complex saddle-point structured problems into more manageable smaller subproblems. It was initially introduced in a technical report [52], and its convergence properties were subsequently analyzed in several works, including [12, 18, 23]. PDHG has an advantage over ADMM in effectively utilizing parallel and distributed computation for a broader range of linearly constrained problems, making it suitable for large-scale problems. Indeed, the presence of augmented terms in ADMM inherently renders its subproblems inseparable. Numerous extensions and variants of the PDHG algorithm have been explored for various applications. For instance, there is a stochastic PDHG [11, 51] and a block-coordinate PD method [34] specifically tailored for large-scale problems. The Bregman PDHG [10, 13, 16, 27, 28] is designed for problems with complex constraints. PDHG with line search [35] is suitable for problems where evaluating the operator norm is challenging. In our problem setting, a straightforward application of the PDHG algorithm to the minimax saddle point form of (1.1) would require the computation of a large number of projections onto the unit simplex (specifically, I times per iteration). Even though the projection is relatively easy to compute, the cost becomes unbearable since I is extremely large. Additionally, the stepsize condition presented in [12, 18, 23, 28] (i.e., $\tau\sigma\|A\|^2 < 1$) is unrealistic for the large-scale problem since storing matrix A (see (4.2)) is not only memory-intensive but also impractical since its different parts could be available only locally. Furthermore, computing the spectral norm of A is prohibitively expensive.

3. Notation and Problem Reformulation

3.1. **Notation.** For any Euclidean space, we denote the endowed inner product and the induced norm by $\langle\cdot,\cdot\rangle$ and $\|\cdot\| = \sqrt{\langle\cdot,\cdot\rangle}$, respectively. The Hadamard product is denoted by "$\circ$".

The Frobenius norm of a matrix is denoted by $\|\cdot\|_F$. The interior and relative interior of any set S is denoted by $\mathrm{int}(S)$ and $\mathrm{ri}(S)$, respectively. Let ϕ be any extended real-valued closed proper convex function on $\mathbb{R}^n$. The effective domain of ϕ is denoted by $\mathrm{dom}\,\phi := \{\mathbf{u} \in \mathbb{R}^n : \phi(\mathbf{u}) < +\infty\}$. Furthermore, the proximal mapping of ϕ is given by $\mathrm{Prox}_\phi(\mathbf{u}) := \arg\min_{\mathbf{v}}\{\phi(\mathbf{v}) + \frac{1}{2}\|\mathbf{v}-\mathbf{u}\|^2\}$, which is uniquely well defined for any $\mathbf{u} \in \mathbb{R}^n$. Let $X := (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_I)^\top \in \mathbb{R}^{I\times J}$, $\mathscr{X} := \{X \in \mathbb{R}^{I\times J} : \mathbf{x}_i \in \Delta, i \in \mathbb{I}\}$, and $R(X) := (\mathbf{1}_I^\top (R^1 \circ X)\mathbf{1}_J, \ldots, \mathbf{1}_I^\top (R^T \circ X)\mathbf{1}_J)^\top \in \mathbb{R}^T$ with $R^t := (\mathbf{r}_1^t, \mathbf{r}_2^t, \ldots, \mathbf{r}_I^t)^\top \in \mathbb{R}^{I\times J}, t \in \mathbb{T}$.

3.2. **Problem Reformulation and the Assumption.** By introducing an auxiliary variable $\mathbf{y} \in \mathbb{R}^T$, we can express problem (1.1) in the following equivalent form

$$\begin{aligned} \min_{\{\mathbf{x}_i\in\Delta\}_{i\in\mathbb{I}},\mathbf{y}} \quad & \alpha\sum_{i\in\mathbb{I}} \mathbf{c}_i^\top \mathbf{x}_i + \beta h(\mathbf{y}) \\ \text{s.t.} \quad & \sum_{i\in\mathbb{I}} M_i^\top \mathbf{x}_i \le \mathbf{b}, \\ & y_t = \sum_{i\in\mathbb{I}} (\mathbf{r}_i^t)^\top \mathbf{x}_i - p_t,\ \forall t \in \mathbb{T}. \end{aligned} \tag{3.1}$$

Utilizing the definition of $R(X)$, the equality constraints in (3.1) can be reformulated as $R(X) - \mathbf{y} = \mathbf{p}$. In the following, we impose a standard assumption on (3.1).

Assumption 3.1. Assume that the set of solutions of (3.1) is nonempty and there exist $X \in \mathrm{ri}(\mathscr{X})$ and $\mathbf{y} \in \mathrm{ri}(\mathrm{dom}\, h)$ such that $\sum_{i\in\mathbb{I}} M_i^\top \mathbf{x}_i < \mathbf{b}$ and $R(X) - \mathbf{y} = \mathbf{p}$.

Let $\eta \in \mathbb{R}^K_+$ and $\gamma \in \mathbb{R}^T$ be the Lagrange multipliers attached to the constraints of (3.1). Then the Lagrangian function of (3.1) is given by

$$L(X,\mathbf{y},\eta,\gamma) = \sum_{i\in\mathbb{I}} \left(\alpha\mathbf{c}_i + M_i\eta - \sum_{t\in\mathbb{T}} \gamma_t \mathbf{r}_i^t\right)^\top \mathbf{x}_i + \beta h(\mathbf{y}) + (\mathbf{y}+\mathbf{p})^\top\gamma - \mathbf{b}^\top\eta,$$

which is defined on $\Omega := \mathscr{X} \times \mathbb{R}^T \times \mathbb{R}^K_+ \times \mathbb{R}^T$. Therefore, the saddle point problem corresponding to (3.1) reads

$$\min_{X\in\mathscr{X},\mathbf{y}\in\mathbb{R}^T} \max_{\eta\in\mathbb{R}^K_+,\gamma\in\mathbb{R}^T} L(X,\mathbf{y},\eta,\gamma). \tag{3.2}$$

Under Assumption 3.1, problem (3.1) is equivalent to (3.2) as shown in [40, Corollaries 28.2.2 and 28.3.1]. Specifically, for any optimal solution pair $(X^*,\mathbf{y}^*)$ to (3.1), there exists $(\eta^*,\gamma^*) \in \mathbb{R}^K_+ \times \mathbb{R}^T$ such that $(X^*,\mathbf{y}^*,\eta^*,\gamma^*)$ is a saddle point of (3.2). The saddle point problem (3.2) has a favorable structure compared to the original problem (1.1), as each variable is separable, allowing for convenient parallel computing. In the next section, we design an efficient method for solving (3.2).

4. The Proposed Algorithm and Theoretical Guarantees

We aim to fully exploit the inherent structure of (3.2) to develop a parallel algorithm that efficiently addresses the challenges posed by large-scale scenarios. The simplest approach for tackling the saddle point problem (3.2) is probably the classical Arrow-Hurwicz method, as introduced by Uzawa [44]. Specifically, given η^k and γ^k, each $\mathbf{x}_i^k$ is updated individually through a projected gradient step, while $\mathbf{y}^k$ is updated through a proximal step, all executed simultaneously. Similarly, η^k and γ^k are updated simultaneously through the projected gradient and

gradient steps, respectively, based on X^{k+1} and $\mathbf{y}^{k+1}$. However, these iterates generated in this manner often fail to converge. The PDHG algorithm [12] improves convergence by incorporating an additional extrapolation step. Nonetheless, straightforwardly applying PDHG to (3.2) would require the computation of a large number of projections onto the unit simplex (specifically, I times per iteration), which becomes computationally expensive for large-scale problems. Fortunately, the Bregman distance-based algorithm can successfully resolve this issue. Therefore, we present a tailored Bregman PDHG that enables every subproblem with a closed-form solution.

4.1. **Algorithmic Design.**

4.1.1. *Basic PDHG for* (3.2). To begin with, we introduce the basic PDHG to solve (3.2). Specifically, given the current iterate $(X^k, \mathbf{y}^k, \eta^k, \gamma^k)$, PDHG generates the next iterate via

$$(X^{k+1}, \mathbf{y}^{k+1}) = \underset{X\in\mathscr{X},\mathbf{y}\in\mathbb{R}^T}{\operatorname{argmin}} L(X, \mathbf{y}, \eta^k, \gamma^k) + \frac{1}{2\tau}(\|X - X^k\|_F^2 + \|\mathbf{y} - \mathbf{y}^k\|^2), \tag{4.1a}$$

$$(\bar{X}^{k+1}, \bar{\mathbf{y}}^{k+1}) = (2X^{k+1} - X^k, 2\mathbf{y}^{k+1} - \mathbf{y}^k), \tag{4.1b}$$

$$(\eta^{k+1}, \gamma^{k+1}) = \underset{\eta\in\mathbb{R}_+^K,\gamma\in\mathbb{R}^T}{\operatorname{argmax}} L(\bar{X}^{k+1}, \bar{\mathbf{y}}^{k+1}, \eta, \gamma) - \frac{1}{2\sigma}(\|\gamma - \gamma^k\|^2 + \|\eta - \eta^k\|^2), \tag{4.1c}$$

where $\tau, \sigma > 0$ are primal and dual stepsizes. The convergence of (4.1) was established in [12, 18, 23] under the condition $\tau\sigma\|A\|^2 < 1$, where

$$A := \begin{pmatrix} M_1^\top & M_2^\top & \cdots & M_I^\top & \mathbf{0} \\ -\mathfrak{R}_1^\top & -\mathfrak{R}_2^\top & \cdots & -\mathfrak{R}_I^\top & \operatorname{diag}(\mathbf{1}_T) \end{pmatrix}, \tag{4.2}$$

in which $\mathfrak{R}_i := (\mathbf{r}_i^1, \mathbf{r}_i^2, \ldots, \mathbf{r}_i^T) \in \mathbb{R}^{J\times T}, i \in \mathbb{I}$. Note that the primal variables X and $\mathbf{y}$ do not overlap, and thus the minimization w.r.t. each of them can be carried out independently, as do the dual variables η and γ. Furthermore, the minimization w.r.t. X can be divided into I independent subproblems, i.e., for each $i \in \mathbb{I}$, one needs to solve

$$\mathbf{x}_i^{k+1} = \underset{\mathbf{x}_i\in\Delta}{\operatorname{argmin}} \langle \mathbf{v}_i^{k+1}, \mathbf{x}_i\rangle + \frac{1}{2\tau}\|\mathbf{x}_i - \mathbf{x}_i^k\|^2,$$

where $\mathbf{v}_i^{k+1} := \alpha\mathbf{c}_i + M_i\eta^k - \sum_{t\in\mathbb{T}} \gamma_t^k \mathbf{r}_i^t$. The main cost lies in calculating the projections onto the unit simplex. As I is large, this could hinder fast practical convergence; see the remarks in [32, 46] which proposed efficient projection methods onto the intersection of a half-space and a box-like set. To address this issue, in the following we propose a variant of PDHG based on a tailored Bregman distance.

4.1.2. *Bregman Distance.* Many algorithms based on Bregman distance have received extensive research. The applications of these algorithms mainly include sparse semidefinite programming [27], optimal transport and Wasserstein barycenter problems [10], supervised machine learning [16], and so on. Let ψ be a convex function satisfying $\operatorname{int}(\operatorname{dom}\psi) \neq \emptyset$. Assume that ψ is continuous on $\operatorname{dom}\psi$ and continuously differentiable on $\operatorname{int}(\operatorname{dom}\psi)$. The Bregman distance $D_\psi : \operatorname{dom}\psi \times \operatorname{int}(\operatorname{dom}\psi) \to \mathbb{R}$ generated by the kernel function ψ is defined by

$D_\psi(\mathbf{u},\mathbf{v}) = \psi(\mathbf{u}) - \psi(\mathbf{v}) - \langle \nabla\psi(\mathbf{v}), \mathbf{u}-\mathbf{v}\rangle$. A kernel function useful in processing unit simplex constraint is the so-called negative entropy defined by

$$\psi(\mathbf{u}) = \begin{cases} \sum_{j=1}^{J} u_j \ln u_j & \text{if } \mathbf{u} \in \Delta, \\ +\infty & \text{otherwise,} \end{cases}$$

where $0\ln 0 \equiv 0$ is assumed. Substituting the above negative entropy function in the definition of D_ψ, we obtain $D_\psi(\mathbf{u},\mathbf{v}) = \sum_{j=1}^{J} u_j \ln \frac{u_j}{v_j}$, $\forall\ \mathbf{u},\mathbf{v} \in \Delta$. In the following, we utilize the aforementioned Bregman distance to design the algorithm.

4.1.3. *PDHG based on Bregman Distance.* To deal with the unit simplex constraints, we adopt the Bregman distance generated by the negative entropy for proximity. In particular, we replace the proximal term $\frac{1}{2}\|X - X^k\|_F^2 = \frac{1}{2}\sum_{i\in\mathbb{I}}\|\mathbf{x}_i - \mathbf{x}_i^k\|^2$ in (4.1a) by $\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i, \mathbf{x}_i^k)$, where the kernel function ψ is the negative entropy. Thus, we generates X^{k+1} via solving I separated subproblems of the form

$$\mathbf{x}_i^{k+1} = \underset{\mathbf{x}_i\in\Delta}{\operatorname{argmin}} \langle \mathbf{v}_i^{k+1}, \mathbf{x}_i\rangle + \frac{1}{\tau} D_\psi(\mathbf{x}_i, \mathbf{x}_i^k),\ i \in \mathbb{I}. \tag{4.3}$$

In fact, here $\mathbf{x}_i^{k+1}$ can be obtained via

$$x_{i,j}^{k+1} = \frac{x_{i,j}^k \exp(-\tau v_{i,j}^{k+1})}{\sum_{j\in\mathbb{J}} x_{i,j}^k \exp(-\tau v_{i,j}^{k+1})},\ j \in \mathbb{J}.$$

Since the minimization w.r.t. $\mathbf{y}$ (4.1a) and the maximization w.r.t. η and γ (4.1c) can be easily expressed by closed-form formulas, we keep them unchanged. The pseudo-code of the proposed PDHG with Bregman distance (denoted by B-PDHG) is given in Algorithm 1.

Algorithm 1 B-PDHG

Input: $X^0, \mathbf{y}^0, \eta^0, \gamma^0$, stepsizes τ and σ.

Step 1a: For $i \in \mathbb{I}$, **compute**

$$\mathbf{v}_i^{k+1} = \alpha\mathbf{c}_i + M_i\eta^k - \sum\nolimits_{t\in\mathbb{T}} \gamma_t^k \mathbf{r}_i^t, \tag{4.4}$$

$$\mathbf{x}_i^{k+1} = \frac{\mathbf{x}_i^k \circ \mathbf{exp}(-\tau\mathbf{v}_i^{k+1})}{\mathbf{1}_J^\top(\mathbf{x}_i^k \circ \mathbf{exp}(-\tau\mathbf{v}_i^{k+1}))}, \tag{4.5}$$

$$\bar{\mathbf{x}}_i^{k+1} = 2\mathbf{x}_i^{k+1} - \mathbf{x}_i^k. \tag{4.6}$$

Step 1b: Compute

$$\mathbf{y}^{k+1} = \mathrm{Prox}_{\tau\beta h}(\mathbf{y}^k - \tau\gamma^k), \tag{4.7}$$

$$\bar{\mathbf{y}}^{k+1} = 2\mathbf{y}^{k+1} - \mathbf{y}^k, \tag{4.8}$$

$$\eta^{k+1} = \left[\eta^k + \sigma\left(\sum\nolimits_{i\in\mathbb{I}} M_i^\top \bar{\mathbf{x}}_i^{k+1} - \mathbf{b}\right)\right]_+, \tag{4.9}$$

$$\gamma^{k+1} = \gamma^k + \sigma\left(\bar{y}^{k+1} + \mathbf{p} - R(\bar{X}^{k+1})\right). \tag{4.10}$$

Step 2: Let $k \longleftarrow k+1$ **and return to Step 1.**

We give some remarks for Algorithm 1.

(1) We update X^{k+1} through Step 1a (i.e.,(4.4)-(4.5), where $\mathbf{exp}(\mathbf{u}) := (\exp(u_j))_{j=1}^J$ for any $\mathbf{u} \in \mathbb{R}^J$), which consists of I independent computational steps. However, on a single-threaded computer, Step 1a can only be executed sequentially.

(2) We update $\mathbf{y}^{k+1}$ through (4.7), with the main computational burden lying in computing the proximal operator of h. It should be noted that evaluating the proximal mapping in (4.7) is itself an optimization problem in general. For many interesting examples, however, the proximal mapping of h has a closed-form solution or can be computed efficiently. Several practical examples of the fairness promoting regularizer h and the corresponding proximal mapping are summarized in Table 1. For the detailed procedure of computing the proximity maps, please refer to [5].

TABLE 1. Examples of regularizer and their proximal mappings. $\lambda > 0$ is a parameter and $B_{\|\cdot\|_1}[0,1] = \{\mathbf{u} \in \mathbb{R}^T : \|\mathbf{u}\|_1 \le 1\}$ denotes the l_1-norm unit ball. The shrinkage operator is given by $\text{Shrinkage}(\mathbf{y},\lambda) := \max(|\mathbf{y}| - \lambda, 0) \circ \text{sign}(\mathbf{y})$, where $|\mathbf{y}|$ and $\text{sign}(\mathbf{y})$ denote the magnitudes and signs of the components of $\mathbf{y}$, respectively.

$h(\mathbf{y})$	$\text{dom}(h)$	$\text{Prox}_{\lambda h}(\mathbf{y})$
$\|\mathbf{y}\|_\infty$	$\mathbb{R}^T$	$\mathbf{y} - \lambda\,\text{Proj}_{B_{\|\cdot\|_1}[0,1]}(\mathbf{y}/\lambda)$
$\|\mathbf{y}\|_1$	$\mathbb{R}^T$	$\text{Shrinkage}(\mathbf{y},\lambda)$
$\|\mathbf{y}\|^2$	$\mathbb{R}^T$	$\mathbf{y}/(2\lambda+1)$
$\max_{t\in\mathbb{T}}\{y_t\}$	$\mathbb{R}^T$	$\mathbf{y} - \lambda\,\text{Proj}_{\Delta}(\mathbf{y}/\lambda)$
$-\sum_{t\in\mathbb{T}}\ln(y_t)$	$\mathbb{R}^T_{++}$	$\left(\frac{y_t+\sqrt{y_t^2+4\lambda}}{2}\right)_{t\in\mathbb{T}}$

4.1.4. *Distributed B-PDHG.* To balance the computational load, we distribute the task of solving $\{\mathbf{x}_i : i \in \mathbb{I}\}$ parallel subproblems to m nodes (assume m divides I) and assign the remaining tasks, including solving the $\mathbf{y}$-subproblem and updating the multipliers η and γ, to a central node. The diagram of the overall distributed B-PDHG algorithm with an illustration of the flow of data is given in Figure 1.

To facilitate distributed computing, we split the problem data $\{M_i, \mathbf{c}_i : i \in \mathbb{I}\}$ and $\{\mathbf{r}_i^t : i \in \mathbb{I}, t \in \mathbb{T}\}$ into m blocks and stored them locally at the m nodes. In each iteration, the central node first transmits the current multipliers to the m nodes. Then, based on the multipliers and the local information, matrix-vector multiplications are carried out on the m nodes in parallel. After that, all the m nodes need to transfer the updated information of η^k and γ^k back to the central node. Let $\mathscr{I}_s := \left\{(s-1)\frac{I}{m}+1, (s-1)\frac{I}{m}+2, \ldots, \frac{sI}{m}\right\}$ for $s = 1,2,\ldots,m$. The pseudocode for solving the $\{\mathbf{x}_i : i \in \mathscr{I}_s\}$ subproblems at node s is given in Algorithm 2, and computations in the central node is described in Algorithm 3.

4.2. **Theoretical Guarantees.** In this subsection, we establish the iterate convergence and ergodic sublinear convergence rate results for Algorithm 1, measured by the primal-dual gap

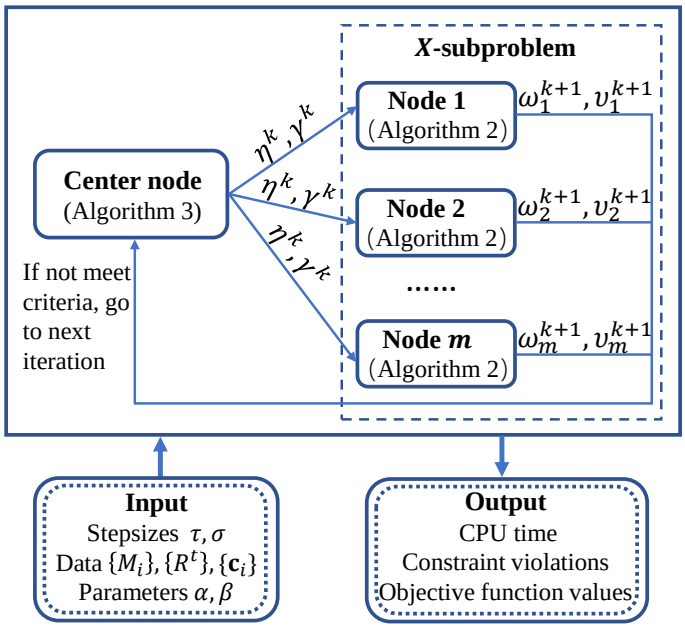

FIGURE 1. Diagram of the proposed distributed B-PDHG.

Algorithm 2 Distributed B-PDHG at iteration k (node s).

Local data: $\alpha, \mathscr{I}_s, \{\mathbf{c}_i, M_i, \mathbf{x}_i^k\}_{i\in\mathscr{I}_s}$, $\{\mathbf{r}_i^t\}_{i\in\mathscr{I}_s, t\in\mathbb{T}}$.
Input: τ, η^k, γ^k.
Output: v_s^{k+1} and $\omega_s^{k+1} = \{\omega_{s,t}^{k+1}\}_{t\in\mathbb{T}}$.
Main computations: First, for each $i \in \mathscr{I}_s$, compute $\mathbf{v}_i^{k+1}, \mathbf{x}_i^{k+1}$ and $\bar{\mathbf{x}}_i^{k+1}$ via (4.4)-(4.6). Then, compute

$$v_s^{k+1} = \sum_{i\in\mathscr{I}_s} M_i^\top \bar{\mathbf{x}}_i^{k+1},$$
$$\omega_{s,t}^{k+1} = \sum_{i\in\mathscr{I}_s} \mathbf{1}_J^T (\mathbf{r}_i^t \circ \bar{\mathbf{x}}_i^{k+1}) \text{ for each } t \in \mathbb{T}.$$

Algorithm 3 Distributed B-PDHG at iteration k (central node)

Local data: $\beta, \mathbf{y}^k, \gamma^k, \eta^k, \mathbf{b}, \mathbf{p}$.
Input: $\tau, \sigma, \{(v_s^{k+1}, \omega_s^{k+1})\}_{s=1}^m$.
Output: η^{k+1}, γ^{k+1}.
Main computations: First, compute $\mathbf{y}^{k+1}$ and $\bar{\mathbf{y}}^{k+1}$ via (4.7) and (4.8). Then, compute

$$v^{k+1} = \sum_{s=1}^m v_s^{k+1},$$
$$\omega^{k+1} = \sum_{s=1}^m \omega_s^{k+1},$$
$$\eta^{k+1} = [\eta^k + \sigma(v^{k+1} - \mathbf{b})]_+,$$
$$\gamma^{k+1} = \gamma^k + \sigma(\bar{\mathbf{y}}^{k+1} + \mathbf{p} - \omega^{k+1}).$$

function defined by

$$G(X,\mathbf{y},\eta,\gamma) = L(X,\mathbf{y},\eta^*,\gamma^*) - L(X^*,\mathbf{y}^*,\eta,\gamma),$$

where $(X^*,\mathbf{y}^*,\eta^*,\gamma^*)$ is any saddle point of (3.2). The primal-dual gap function is frequently adopted in the literature, as seen in, for example, [12, 35]. To maintain a smooth flow of the paper, in this section we only present the convergence results, while their proofs are postponed to the appendix.

Theorem 4.1 (Ergodic sublinear convergence rate)**.** *Let* $(X^*,\mathbf{y}^*,\eta^*,\gamma^*)$ *be any saddle point of* (3.2) *and* $\delta_i = \max_{t\in\mathbb{T}} \|\mathbf{r}_i^t\|$ *for* $i \in \mathbb{I}$*. Assume that the primal and dual stepsizes* $\tau,\sigma > 0$ *satisfy*

$$\tau\sigma \le 1 \Big/ \max\left\{2\textstyle\sum_{i\in\mathbb{I}} \|M_i\|^2,\ 1 + 2T\sum_{i\in\mathbb{I}} \delta_i^2\right\}. \tag{4.11}$$

Then, for any $N \ge 1$*, the sequence* $\{(X^k,\mathbf{y}^k,\eta^k,\gamma^k)\}$ *generated by Algorithm 1 satisfies*

$$\begin{aligned} &G(\tilde{X}^N,\tilde{\mathbf{y}}^N,\tilde{\eta}^N,\tilde{\gamma}^N) \\ &\le \frac{2}{N}\left(\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^0) + \frac{1}{2\tau}\|\mathbf{y}^0-\mathbf{y}^*\|^2 + \frac{1}{2\sigma}\left(\|\eta^0-\eta^*\|^2 + \|\gamma^0-\gamma^*\|^2\right)\right), \end{aligned} \tag{4.12}$$

where $(\tilde{X}^N,\tilde{\mathbf{y}}^N,\tilde{\eta}^N,\tilde{\gamma}^N) = \frac{1}{N}\sum_{k=0}^{N-1}(X^k,\mathbf{y}^k,\eta^k,\gamma^k)$.

Remark 4.1. The stepsizes condition prescribed by (4.11) provides a practical lower bound on the stepsizes. Additionally, an effective acceleration method is to utilize linesearch strategy proposed in [35] for PDHG, which allows for adaptive and potentially much larger stepsizes.

Theorem 4.2 (Iterate convergence)**.** *Assume that the inequality in* (4.11) *is strictly satisfied. Then, the sequence* $\{(X^k,\mathbf{y}^k,\eta^k,\gamma^k)\}$ *generated by Algorithm 1 converges to some saddle point of* (3.2).

Note that a general framework of Bregman splitting methods was proposed in [28], which requires the stepsize condition $\tau\sigma\|A\|^2 < 1$ in order to ensure convergence. Unfortunately, in our scenario, the large-scale of matrix A makes it impractical to efficiently compute its spectral norm. Moreover, the building blocks M_i and $\mathfrak{R}_i$ (see (4.2)) are available only locally at each computing node, which further complicates the computation of $\|A\|$. In contrast, our stepsize condition (4.11) is significantly easier to implement, as it is dependent only on quantities that can be computed locally at each node. This highlights the need for a refined analysis of our tailored algorithm.

5. Numerical Results

In this section, we present numerical results to validate that the proposed B-PDHG algorithm is effective and efficient when dealing with large-scale RAPs. Both synthetic and real-world industrial datasets with different regularizers are tested to demonstrate the favorable performance of the proposed distributed B-PDHG algorithm. The experiments were operated on the cloud computing platform in Ant Group, and every single virtual machine (VM) provides a virtual CPU and 1.5GB RAM. IPOPT and Gurobi are both operated on a single VM on the cloud, each equipped with a virtual CPU and 8GB RAM.

5.1. **Experiments on Synthetic Dataset.**

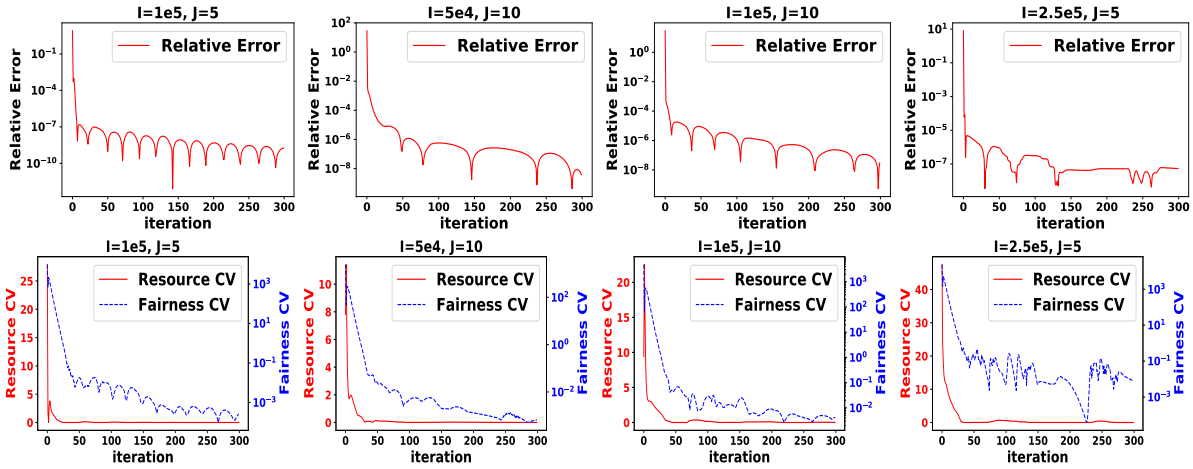

FIGURE 2. Convergence of the B-PDHG on a synthetic dataset.

TABLE 2. Computation time (in seconds) of the distributed B-PDHG with different nodes m.

Scale $I \times J$	I	J	Number of nodes (m)					
			0	2	4	8	16	32
1.0e7	1.0e6	10	436	222	107	57.8	39.7	32.8
1.0e7	5.0e5	20	519	289	134	79.7	49.7	37.7
2.0e7	1.0e6	20	882	423	225	107	62.4	42.0
2.5e7	5.0e6	5	1542	1039	510	219	140	96.1
5.0e7	5.0e6	10	3598	1657	917	452	216	141
5.0e7	1.0e7	5	>1h	2110	1192	583	240	176
1.0e8	5.0e6	20	>1h	>1h	>1h	>1h	1737	879
2.5e8	5.0e7	5	>1h	>1h	>1h	>1h	1795	998

5.1.1. *Experimental Setup.* In [24], a quadratic function was adopted to allocate a limited amount of vaccines in an equitable manner during an influenza pandemic. In this background, we conduct a synthetic experiment on problem (3.1) with $\alpha = \beta = 1$ and $h(\mathbf{y}) = \|\mathbf{y}\|^2$. The cost vector $\mathbf{c}_i$ and fairness coefficients p_t are drawn randomly in $(0,1)$. Each item j has a resource constraint with budget $b_j = I/2$ and each element $m_{j,k}$ of the resource consumption matrix M_i is also drawn randomly in $(0,1)$ if $j = k$. The dimension of the fairness term T is set to $T = J$. Specifically, for each $i \in \mathbb{I}$, $r^t_{i,j}$ is drawn uniformly at random in $(0,1)$ if $j = t$, and $r^t_{i,j} = 0$ if otherwise. Each component of the initial primal variable X^0 is initialized at $1/J$, and every component of the initial dual variables η^0 and γ^0 is set to be 0. The initial $\mathbf{y}^0$ is determined by the equality constraint in the RAP (3.1). The stepsizes τ and σ are tuned adaptively according to Remark 4.1.

5.1.2. *Effectiveness of B-PDHG.* In constrained optimization problems, numerical results tend to favor the utilization of function value residuals and constraint violation metrics over the

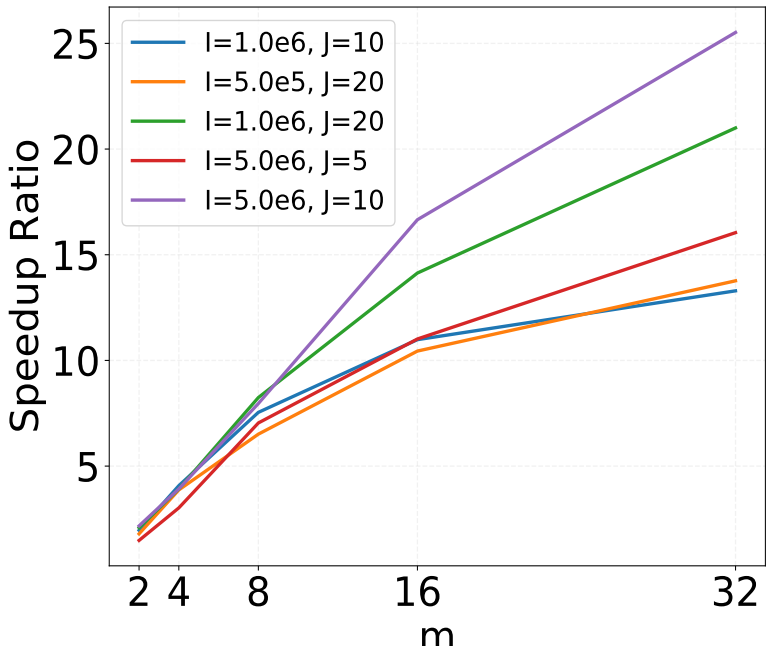

FIGURE 3. Speedup ratio of the distributed B-PDHG.

primal-dual gap metric. Therefore, we take the following metrics to evaluate the effectiveness of B-PDHG:

- The relative error of objective function value for problem (3.1) is defined as $\frac{|\text{obj}_{\text{cur}}-\text{obj}_{\text{opt}}|}{|\text{obj}_{\text{opt}}|}$, where obj_{cur} is the objective function value at the current iteration and obj_{opt} is the optimal objective function value calculated in advance.
- Resource constraint violation (CV) is defined as $\|\left(\sum_{i\in\mathbb{I}} M_i^\top \mathbf{x}_i - \mathbf{b}\right)_+\|_\infty$.
- Fairness CV is defined as $\max_{t\in\mathbb{T}} |\sum_{j\in\mathbb{J}}\sum_{i\in\mathbb{I}} r_{i,j}^t x_{i,j} - p_t - y_t|$.

We simulate the scale of the synthetic dataset by setting (I, J) to four different sizes (1e5, 5), (5e4, 10), (1e5, 10) and (2.5e5, 5). Since these problems are small-scale, we run B-PDHG on a single machine for the purpose of illustration. The first row of Figure 2 illustrates the trend of relative error of objective function value over the number of iterations. The second row of Figure 2 displays the behavior of resource constraint violation and fairness constraint violation versus number of iterations. As can be seen from Figure 2, B-PDHG converges stably in all cases. The relative errors and constraint violations were reduced sharply in the first few iterations and then converged roughly to stationarity within several hundreds of iterations.

5.1.3. *Speedup by Distributed Computation.* In this section, we provide numerical results to illustrate the effectiveness of the distributed B-PDHG on large-scale RAPs. Denote the number of nodes used for distributed computation by m, which does not include the central node. Table 2 shows the computation time (in seconds) on solving RAPs with different scales and number of nodes. It can be seen clearly that B-PDHG implemented on distributed machines accelerates the practical convergence significantly. In particular, B-PDHG without distributed

computation (i.e., $m = 0$) hardly converges in one hour when the problem scale is over 5.0e7. In contrast, the distributed B-PDHG converges, e.g., within 20 minutes for $m = 32$. Figure 3 further demonstrates the speedup ratio of distributed B-PDHG. Here, the speedup ratio is defined as speedup_ratio$(m) = T(m)/T(0)$, where $T(m)$ denotes the CPU time consumed by B-PDHG with m distributed nodes (not including the central one). For all the tested cases, we observed that distributed B-PDHG almost halved the CPU time as the number of distributed machines doubles.

5.1.4. *Comparison with Solvers.* In this experiment, we compare our distributed B-PDHG (with $m = 8$) with IPOPT [45] and Gurobi [22]. IPOPT is a popular interior-point-based convex programming solver that has been widely used in many applications and Gurobi is a powerful mathematical programming solver for linear programming, quadratic programming, etc. Problem (3.1) can be transformed into a quadratic programming problem, which falls into the applicable scope of IPOPT and Gurobi when $h(\mathbf{y}) = \|\mathbf{y}\|^2$. The left figure of Figure 4 plots the computation time using the three methods. We observe that the B-PDHG is significantly faster than IPOPT and Gurobi in general. In particular, IPOPT is capable of solving only the first two cases, each requiring approximately 7 and 13 minutes, respectively, whereas B-PDHG solves the problem within about 6 and 14 seconds, respectively. Furthermore, when the problem scale increases to (5.0e5, 5), IPOPT runs out of memory, while our distributed B-PDHG is able to solve the problem within 18 seconds. Similarly, for the cases (5.0e6, 5), Gurobi requires about 18 minutes, while our algorithm takes about 219 seconds and is about 5 times faster than Gurobi. For the case (5.0e6, 10), Gurobi runs out of memory, while our distributed algorithm running on 8 distributed machines is able to solve the problems within about 8 minutes.

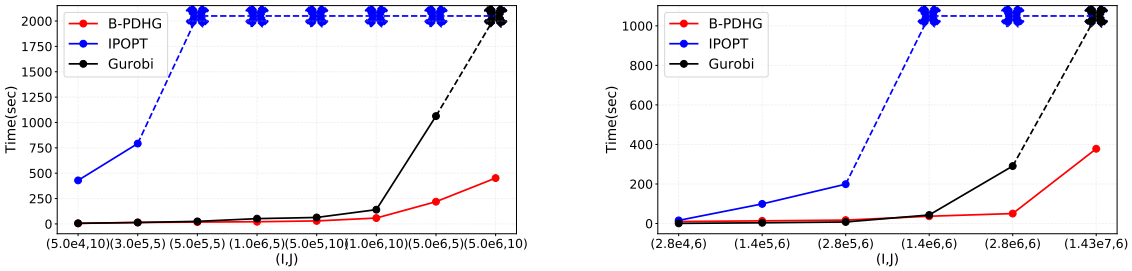

FIGURE 4. Computation time (sec) of the distributed B-PDHG, IPOPT and Gurobi. Left: Experimental result on a synthetic dataset and the distributed B-PDHG with $m = 8$. Right: Experimental result on the industrial dataset and the distributed B-PDHG with $m = 16$. "$\times$" means that IPOPT or Gurobi ran out of memory.

5.2. **Experiments on Real-world Problems.**

5.2.1. *Financial Asset Allocation.* Solving allocation problems at scale is crucial for many real-world applications such as capital management of financial institutes, portfolio management, marketing, and so on. In this section, we test our distributed B-PDHG algorithm on a real-world loan management problem from the Ant Group. In this problem, users' assets and financial institutes represent the two sides of the RAP. The decision maker aims to balance the asset risks

among different institutes under the asset demands of the institutes. Specifically, the formulation of this problem is given by

$$\min_{\{\mathbf{x}_i\in\Delta\}_{i\in\mathbb{I}}}\left\{\sum_{j\in\mathbb{J}}\left|\frac{1}{A_j\bar{r}}\sum_{i\in\mathbb{I}}r_i a_i x_{ij}-1\right| \text{ s.t. } \left|\sum_{i\in\mathbb{I}}a_i x_{ij}-A_j\right|\le\varepsilon A_j,\ \ \forall j\in\mathbb{J}\right\}. \tag{5.1}$$

Here, I and J are the numbers of users and institutes, respectively; for each i, a_i is the assets of user i and r_i is the risk coefficient of a_i; for each j, A_j is the asset demands of the jth institute; $\bar{r}$ represents the global asset risk. We use the mean absolute error (MAE) to control the risks among different institutes in (5.1). The inequality constraints in (5.1) ensure that, for each institute, the deviation between the allocated assets of the institute and its asset demands is less than a given threshold ε. The dataset contains 1.43e7 users with coefficients (a_i, r_i), as well as 6 institutes with coefficients A_j in total. The deviation threshold ε is set to be 0.05 in our experiments. Apparently, problem (5.1) can be reformulated as

$$\begin{aligned}\min_{\{\mathbf{x}_i\in\Delta\}_{i\in\mathbb{I}}}\ &\sum_{j\in\mathbb{J}}|y_j|\\ \text{s.t.}\ \ &(1-\varepsilon)A_j\le\sum_{i\in\mathbb{I}}a_i x_{ij}\le(1+\varepsilon)A_j,\ \forall j\in\mathbb{J},\\ &y_j=\frac{1}{A_j\bar{r}}\sum_{i\in\mathbb{I}}r_i a_i x_{ij}-1,\ \forall j\in\mathbb{J}.\end{aligned} \tag{5.2}$$

By setting $\mathbb{T}=\mathbb{J}$, $p_t=1$, $r^t_{i,j}=\frac{1}{A_t\bar{r}}r_i a_i$ if $j=t$ and 0 if otherwise,

$$\mathbf{b}=\begin{pmatrix}(\varepsilon+1)A_1\\(\varepsilon+1)A_2\\\vdots\\(\varepsilon+1)A_J\\(\varepsilon-1)A_1\\(\varepsilon-1)A_2\\\vdots\\(\varepsilon-1)A_J\end{pmatrix}\ \text{and}\ M_i=a_i\begin{pmatrix}1&&&&\\&1&&&\\&&\ddots&&\\&&&&1\\-1&&&&\\&-1&&&\\&&\ddots&&\\&&&&-1\end{pmatrix}^{\top},\quad\forall i\in\mathbb{I},$$

one can easily reformulate (5.1) as (3.1) with $\alpha=0,\beta=1$ and $h=\|\cdot\|_1$. Therefore, we can utilize the distributed B-PDHG algorithm to solve (5.2). The initialization of the primal and dual variables follows the same procedure as outlined in Section 5.1.1.

5.2.2. *Results.* Figure 5 demonstrates the effectiveness of our distributed B-PDHG based on the industrial dataset from the Ant Group. The number of distributed machines we use in this experiment for parallel computing is $m=16$. Figure 5 shows the evolution of objective values and constraint violations w.r.t. the number of iterations. It can be seen that our algorithm optimized the fairness objective in (5.1) to 10^{-5} (see the left plot in Figure 5) and converged within less than 300 iterations in about 378 seconds (see the right plot in Figure 4). This demonstrates the effectiveness of the proposed distributed B-PDHG sufficiently in solving large-scale problems from the enterprise.

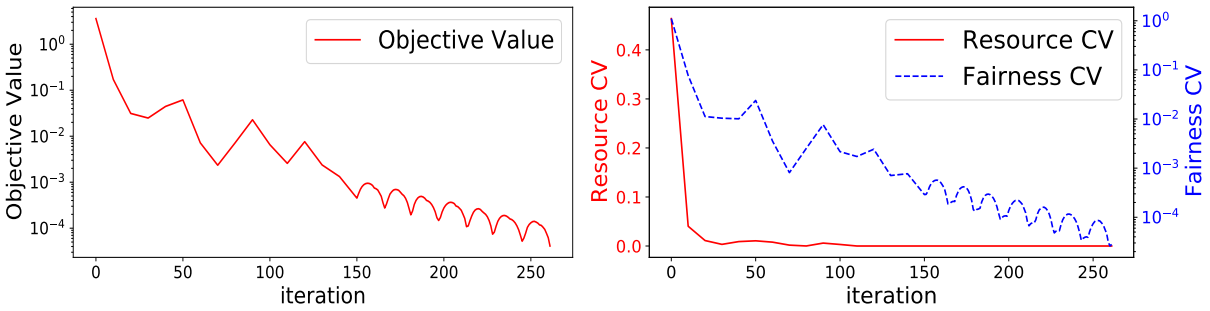

FIGURE 5. Convergence of the distributed B-PDHG (with $m = 16$) on an industrial dataset with $(I, J) = (1.43e7, 6)$.

We also compare our algorithm with IPOPT and Gurobi on problem (5.1). Since the two solvers cannot work on the entire dataset (due to the extremely large scale), we construct multiple experiments with different scales by taking random samples from the whole dataset. The right plot in Figure 4 compares the computation time results between the three methods, from which it can be seen that IPOPT only works with relatively small-scale problems (e.g., the first three cases in the figure), and our distributed B-PDHG is also faster in general. For the case (2.8e6, 6), IPOPT failed due to limited memory and Gurobi took about 5 minutes. In contrast, our distributed B-PDHG solved the problem within only 50 seconds. When working on the entire dataset, both IPOPT and Gurobi failed, while B-PDHG can still solve the problem in several minutes.

6. CONCLUSION

In this paper, we focused on a large-scale fair resource allocation problem with the objective function being the sum of an allocation cost and a fairness promoting regularizer. Motivated by the well-known primal-dual hybrid gradient (PDHG) algorithm, we proposed a distributed PDHG algorithm with tailored Bregman distance to solve a saddle point reformulation of the problem. The proposed algorithm mainly involves matrix-vector multiplications and allows distributed parallel computations in solving a large number of subproblems. It was shown that the algorithm converges globally at an ergodic sublinear rate $\mathcal{O}(1/N)$. Numerical results on both synthetic and real industrial datasets are provided to illustrate that our algorithm is efficient, stable and outperforms the off-the-shelf solvers IPOPT and Gurobi. In particular, the algorithm has been employed in practical industrial applications at the Ant Group.

Acknowledgments

This research was funded by Ant Group, the National Natural Science Foundation of China (grant numbers 12371301 and 12431011), and the Natural Science Foundation for Distinguished Young Scholars of Gansu Province (grant number 22JR5RA223).

REFERENCES

[1] A. Agnetis, B. Chen, G. Nicosia, A. Pacifici, Price of fairness in two-agent single-machine scheduling problems, European J. Oper. Res. 276 (2019), 79-87.

[2] S. Agrawal, M. Zadimoghaddam, V. Mirrokni, Proportional allocation: Simple, distributed, and diverse matching with high entropy, In: Proceedings of the 35th International Conference on Machine Learning, vol. 80, pp. 99-108, 2018.
[3] S. Balseiro, H. Lu, V. Mirrokni, Dual mirror descent for online allocation problems, In: Proceedings of the 37th International Conference on Machine Learning, vol. 119, pp. 613-628, 2020.
[4] S. Balseiro, H. Lu, V. Mirrokni, Regularized online allocation problems: Fairness and beyond, In: Proceedings of the 38th International Conference on Machine Learning, vol. 139, pp. 630-639, 2021.
[5] A. Beck, First-order methods in optimization, volume 25 of MOS-SIAM Series on Optimization, SIAM, Mathematical Optimization Society, Philadelphia, PA, 2017.
[6] A. Beck, A. Nedić, A. Ozdaglar, M. Teboulle, An $O(1/k)$ gradient method for network resource allocation problems, IEEE Trans. Control Netw. Syst. 1 (2014), 64-73.
[7] A. Beck, M. Teboulle, Mirror descent and nonlinear projected subgradient methods for convex optimization, Oper. Res. Lett. 31 (2003), 167-175.
[8] D. Bertsimas, V. F. Farias, N. Trichakis, The price of fairness, Oper. Res. 59 (2011), 17-31.
[9] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Found. Trends Mach. Learn. 3 (2010), 1-122.
[10] A. Chambolle, J. P. Contreras, Accelerated Bregman primal-dual methods applied to optimal transport and Wasserstein barycenter problems, SIAM J. Math. Data Sci. 4 (2022), 1369-1395.
[11] A. Chambolle, M. J. Ehrhardt, P. Richtárik, C.-B. Schönlieb, Stochastic primal-dual hybrid gradient algorithm with arbitrary sampling and imaging applications, SIAM J. Optim. 28 (2018), 2783–2808.
[12] A. Chambolle,T. Pock, A first-order primal-dual algorithm for convex problems with applications to imaging, J. Math. Imaging Vision 40 (2011), 120-145.
[13] A. Chambolle, T. Pock, On the ergodic convergence rates of a first-order primal-dual algorithm, Math. Program. 159 (2016), 253-287.
[14] G. Chen, M. Teboulle, Convergence analysis of a proximal-like minimization algorithm using Bregman functions, SIAM J. Optim. 3 (1993), 538-543.
[15] F. Criado, D. Martinez-Rubio, S. Pokutta, Fast algorithms for packing proportional fairness and its dual, In: Advances in Neural Information Processing Systems, volume 35, 2022.
[16] J. Darbon, G. P. Langlois, Accelerated nonlinear primal-dual hybrid gradient methods with applications to supervised machine learning, arXiv preprint arXiv:2109.12222, 2022.
[17] J. Eckstein, Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming, Math. Oper. Res. 18 (1993), 202-226.
[18] E. Esser, X. Zhang, T. F. Chan, A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science, SIAM J. Imaging Sci. 3 (2010), 1015-1046.
[19] J. Feldman, M. Henzinger, N. Korula, V. S. Mirrokni, C. Stein, Online stochastic packing applied to display ad allocation, In: Algorithms—ESA 2010. Part I, volume 6346 of Lecture Notes in Comput. Sci. pp. 182-194, 2010.
[20] D. Gabay, B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite element approximation, Comput. Math. Appl. 2 (1976), 17-40.
[21] A. Ghodsi, M. Zaharia, B. Hindman, A. Konwinski, S. Shenker, I. Stoica, Dominant resource fairness: Fair allocation of multiple resource types, In: Nsdi, vol. 11, pp. 24-24, 2011.
[22] Gurobi Optimization LLC. Gurobi optimizer reference manual, 2022.
[23] B. He, X. Yuan, On the $O(1/n)$ convergence rate of the Douglas-Rachford alternating direction method, SIAM J. Numer. Anal. 50 (2012), 700-709.
[24] H.C. Huang, B. Singh, D.B. Morton, G.P. Johnson, B. Clements, L.A. Meyers, Equalizing access to pandemic influenza vaccines through optimal allocation to public health distribution points, PLoS One 12 (2017), e0182720.
[25] D. A. Iancu, N. Trichakis, Fairness and efficiency in multiportfolio optimization, Oper. Res. 62 (2014), 1285-1301.
[26] A. Ivanova, P. Dvurechensky, A. Gasnikov, D. Kamzolov, Composite optimization for the resource allocation problem, Optim. Methods Softw. 36 (2021), 720-754.

[27] X. Jiang, L. Vandenberghe, Bregman primal-dual first-order method and application to sparse semidefinite programming, Comput. Optim. Appl. 81 (2022), 127-159.
[28] X. Jiang, L. Vandenberghe, Bregman three-operator splitting methods, J. Optim. Theory Appl. 196 (2023), 936–972.
[29] K. C. Kiwiel, Breakpoint searching algorithms for the continuous quadratic knapsack problem, Math. Program. 112 (2008), 473-491.
[30] B. Li, M. Li, R. Zhang, Fair scheduling for time-dependent resources, In: Advances in Neural Information Processing Systems, vol. 34, pp. 21744-21756, 2021.
[31] X. Li, C. Sun, Y. Ye, Simple and fast algorithm for binary integer and online linear programming, In: Advances in Neural Information Processing Systems, vo. 33, pp. 9412-9421, 2020.
[32] Y. J. Liu, Q. Zhu, A semismooth Newton based augmented Lagrangian algorithm for Weber problem, Pac. J. Optim. 18 (2022), 299-315.
[33] X. Lu, Q. Wu, W. Zhong, Multi-slots online matching with high entropy, In: Proceedings of the 39th International Conference on Machine Learning, vol. 162, pp. 14412-14428, 2022.
[34] D. R. Luke Y. Malitsky, Block-coordinate primal-dual method for nonsmooth minimization over linear constraints, In: Large-scale and distributed optimization, volume 2227 of Lecture Notes in Math. pp. 121-147, Springer, Cham, 2018.
[35] Y. Malitsky, T. Pock, A first-order primal-dual algorithm with linesearch, SIAM J. Optim. 28 (2018), 411-432.
[36] Y. Nesterov, V. Shikhman, Dual subgradient method with averaging for optimal resource allocation, European J. Oper. Res. 270 (2018), 907-916.
[37] M. Patriksson, C. Strömberg, Algorithms for the continuous nonlinear resource allocation problem—new implementations and numerical studies, European J. Oper. Res. 243 (2015), 703-722.
[38] S. Perez-Salazar, I. Menache, M. Singh, A. Toriello, Dynamic resource allocation in the cloud with near-optimal efficiency, Oper. Res. 70 (2022), 2517-2537.
[39] F. Petersen, D. Mukherjee, Y. Sun, M. Yurochkin, Post-processing for individual fairness, In: Advances in Neural Information Processing Systems, vol. 34, pp. 25944-25955, 2021.
[40] R. T. Rockafellar, Convex Analysis, Princeton University Press, Princeton, 1970.
[41] N. A. Saxena, K. Huang, E. DeFilippis, G. Radanovic, D. C. Parkes, Y. Liu, How do fairness definitions fare? Testing public attitudes towards three algorithmic definitions of fairness in loan allocations, Artificial Intelligence 283 (2020), 103238.
[42] Z. Shen, L. Gelauff, A. Goel, A. Korolova, K. Munagala, Robust allocations with diversity constraints, In: Advances in Neural Information Processing Systems, vol. 34, pp. 29684-29696, 2021.
[43] E. M. R. Torrealba, J. G. Silva, L. C. Matioli, O. Kolossoski, P. S. M. Santos, Augmented Lagrangian algorithms for solving the continuous nonlinear resource allocation problem, European J. Oper. Res. 299 (2022), 46-59.
[44] H. Uzawa, Iterative methods for concave programming, In: K. J. Arrow, L. Hurwicz, H. Uzawa, (ed.) Studies in Linear and Nonlinear Programming, Stanford University Press, Stanford, 1958.
[45] A. Wächter, L. T. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Math. Program. 106 (2006), 25-57.
[46] B. Wang, L. Lin, Y.-J. Liu, Efficient projection onto the intersection of a half-space and a box-like set and its generalized Jacobian, Optimization 71 (2022), 1073-1096.
[47] Y. Wang, L. Wu, Z. Wu, E. Chen, Q. Liu, Selecting valuable customers for merchants in e-commerce platforms, In: 2016 IEEE 16th International Conference on Data Mining (ICDM), pp. 1281-1286, 2016.
[48] S. Wu, Y. Wei, S. Zhang, W. Meng, Proportional-fair resource allocation for user-centric networks, IEEE Trans. Veh. Technol. 71 (2021), 1549-1561.
[49] J. Yan, Z. Xu, B. Tiwana, S. Chatterjee, Ads allocation in feed via constrained optimization, In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 3386–3394, 2020.
[50] S. Yin, S. Agrawal, A. Zeevi, Online allocation and learning in the presence of strategic agents, In: Advances in Neural Information Processing Systems, vol. 35, pp. 6333-6344, 2022.

[51] Y. Zhang, L. Xiao, Stochastic primal-dual coordinate method for regularized empirical risk minimization, J. Mach. Learn. Res. 18 (2017), 84.
[52] M. Zhu, T. F. Chan, An efficient primal-dual hybrid gradient algorithm for total variation image restoration, UCLA CAM Report 34 (2008), 8-34.

APPENDIX A. PROOFS OF THE MAIN RESULTS

In this section, we provide complete proofs for Theorems 4.1 and 4.2.

A.1. **Preliminaries.** We start with some useful identities and basic lemmas to facilitate the subsequent analysis. For any $\mathbf{u}_1,\mathbf{u}_2,\mathbf{u}_3 \in \mathbb{R}^n$, there holds

$$2\langle \mathbf{u}_1-\mathbf{u}_2,\mathbf{u}_3-\mathbf{u}_1\rangle = \|\mathbf{u}_3-\mathbf{u}_2\|^2-\|\mathbf{u}_3-\mathbf{u}_1\|^2-\|\mathbf{u}_1-\mathbf{u}_2\|^2. \tag{A.1}$$

Lemma A.1 ([14]). *Let $S\subset\mathbb{R}^n$ be an open set with closure $\bar{S}$ and let $\psi:\bar{S}\to\mathbb{R}$ be continuously differentiable on S. Then, for any three points $\mathbf{u}_1,\mathbf{u}_2\in S$ and $\mathbf{u}_3\in\bar{S}$, we have*

$$\langle \nabla\psi(\mathbf{u}_2)-\nabla\psi(\mathbf{u}_1),\mathbf{u}_1-\mathbf{u}_3\rangle = D_\psi(\mathbf{u}_3,\mathbf{u}_2)-D_\psi(\mathbf{u}_3,\mathbf{u}_1)-D_\psi(\mathbf{u}_1,\mathbf{u}_2). \tag{A.2}$$

For simplicity, in the following analysis, we denote

$$\Theta_\psi(\mathbf{u}_3,\mathbf{u}_2,\mathbf{u}_1) := D_\psi(\mathbf{u}_3,\mathbf{u}_2)-D_\psi(\mathbf{u}_3,\mathbf{u}_1)-D_\psi(\mathbf{u}_1,\mathbf{u}_2),$$

and

$$\Theta(\mathbf{u}_3,\mathbf{u}_2,\mathbf{u}_1) := \|\mathbf{u}_3-\mathbf{u}_2\|^2-\|\mathbf{u}_3-\mathbf{u}_1\|^2-\|\mathbf{u}_1-\mathbf{u}_2\|^2.$$

Lemma A.2 ([7]). *The negative entropy function is 1-strongly convex over* $\mathrm{int}(\Delta)$ *with respect to the l_1-norm.*

Remark A.1. Let ψ be the negative entropy function. It follows from Lemma A.2 that for any $(\mathbf{u},\mathbf{v})\in \mathrm{dom}\,\psi\times\mathrm{int}(\mathrm{dom}\,\psi)$, one has $D_\psi(\mathbf{u},\mathbf{v})\geq\frac{1}{2}\|\mathbf{u}-\mathbf{v}\|_1^2$. Furthermore, ψ is continuously differentiable on $\mathrm{int}(\Delta)$ and continuous on Δ. Then, it follows from [17] that for any converging sequence $\{\mathbf{u}^k\}$, we have

$$\mathbf{u}^k\to\mathbf{u} \implies \lim_{k\to\infty} D_\psi(\mathbf{u},\mathbf{u}^k)=0. \tag{A.3}$$

Next, we present a crucial technical lemma for our subsequent analysis.

Lemma A.3. *Let $\{(X^k,\mathbf{y}^k,\eta^k,\gamma^k)\}$ be the sequence generated by Algorithm 1. Then, for any $(X,\mathbf{y},\eta,\gamma)\in\Omega$, one has*

$$\begin{aligned}
L(X^{k+1},\mathbf{y}^{k+1},\eta,\gamma)-L(X,\mathbf{y},\eta^{k+1},\gamma^{k+1}) \leq{}& \frac{1}{\tau}\left(\sum_{i\in\mathbb{I}}\Theta_\psi(\mathbf{x}_i,\mathbf{x}_i^k,\mathbf{x}_i^{k+1})+\frac{1}{2}\Theta(\mathbf{y},\mathbf{y}^k,\mathbf{y}^{k+1})\right)\\
&+\frac{1}{2\sigma}\left(\Theta(\eta,\eta^k,\eta^{k+1})+\Theta(\gamma,\gamma^k,\gamma^{k+1})\right)+\left\langle \gamma-\gamma^{k+1},\mathbf{y}-\mathbf{y}^{k+1}\right\rangle\\
&-\sum_{i\in\mathbb{I}}\left\langle M_i(\eta-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t-\gamma_t^k),\mathbf{x}_i-\mathbf{x}_i^k\right\rangle+\left\langle \gamma^{k+1}-\gamma^k,\mathbf{y}^{k+1}-\mathbf{y}^k\right\rangle\\
&-\left\langle \gamma-\gamma^k,\mathbf{y}-\mathbf{y}^k\right\rangle+\sum_{i\in\mathbb{I}}\left\langle M_i(\eta-\eta^{k+1})-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t-\gamma_t^{k+1}),\mathbf{x}_i-\mathbf{x}_i^{k+1}\right\rangle\\
&+\sum_{i\in\mathbb{I}}\left\langle M_i(\eta^{k+1}-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^{k+1}-\gamma_t^k),\mathbf{x}_i^{k+1}-\mathbf{x}_i^k\right\rangle.
\end{aligned} \tag{A.4}$$

Proof. From the optimality condition of (4.3) and using the identity (A.2), for any $\mathbf{x}_i \in \Delta$, it follows that

$$\alpha(\mathbf{c}_i^\top \mathbf{x}_i^{k+1} - \mathbf{c}_i^\top \mathbf{x}_i) = \Big\langle M_i \eta^k - \sum_{t\in\mathbb{T}} \gamma_t^k \mathbf{r}_i^t, \mathbf{x}_i - \mathbf{x}_i^{k+1} \Big\rangle + \frac{1}{\tau}\Theta_\psi(\mathbf{x}_i, \mathbf{x}_i^k, \mathbf{x}_i^{k+1}).$$

Summing the above over $i \in \mathbb{I}$, for any $\mathbf{x}_i \in \Delta, i \in \mathbb{I}$, we have

$$\begin{aligned} &\alpha\Big(\sum_{i\in\mathbb{I}} \mathbf{c}_i^\top \mathbf{x}_i^{k+1} - \sum_{i\in\mathbb{I}} \mathbf{c}_i^\top \mathbf{x}_i\Big) \\ &= \sum_{i\in\mathbb{I}} \Big\langle M_i \eta^k - \sum_{t\in\mathbb{T}} \gamma_t^k \mathbf{r}_i^t, \mathbf{x}_i - \mathbf{x}_i^{k+1} \Big\rangle + \frac{1}{\tau} \sum_{i\in\mathbb{I}} \Theta_\psi(\mathbf{x}_i, \mathbf{x}_i^k, \mathbf{x}_i^{k+1}). \end{aligned} \tag{A.5}$$

Similarly, from the optimality condition of (4.7) and the identity (A.1) and recalling the notation $\Theta(\mathbf{u}_3, \mathbf{u}_2, \mathbf{u}_1)$, we deduce

$$\beta(h(\mathbf{y}^{k+1}) - h(\mathbf{y})) \le \Big\langle \gamma^k, \mathbf{y} - \mathbf{y}^{k+1} \Big\rangle + \frac{1}{2\tau}\Theta(\mathbf{y}, \mathbf{y}^k, \mathbf{y}^{k+1}),\ \forall\, \mathbf{y} \in \mathbb{R}^T. \tag{A.6}$$

In addition, combining (4.9) and using identity (A.1) and notation $\Theta(\mathbf{u}_3, \mathbf{u}_2, \mathbf{u}_1)$ again, for any $\eta \in \mathbb{R}_+^K$, it holds that

$$\Big\langle \mathbf{b}, \eta^{k+1} - \eta \Big\rangle = \sum_{i\in\mathbb{I}} \Big\langle M_i^\top \bar{\mathbf{x}}_i^{k+1}, \eta^{k+1} - \eta \Big\rangle + \frac{1}{2\sigma}\Theta(\eta, \eta^k, \eta^{k+1}). \tag{A.7}$$

From (4.10) and taking into account (A.1), we obtain

$$\begin{aligned} \Big\langle \mathbf{p}, \gamma - \gamma^{k+1} \Big\rangle = &\Big\langle R(\bar{X}^{k+1}), \gamma - \gamma^{k+1} \Big\rangle - \Big\langle \bar{\mathbf{y}}^{k+1}, \gamma - \gamma^{k+1} \Big\rangle \\ &+ \frac{1}{2\sigma}\Theta(\gamma, \gamma^k, \gamma^{k+1}),\ \forall\, \gamma \in \mathbb{R}^T. \end{aligned} \tag{A.8}$$

By recalling the notation of $R(X)$ and employing elementary calculations, one can deduce that $\Big\langle R(X), \gamma \Big\rangle = \sum_{i\in\mathbb{I}} \Big\langle \sum_{t\in\mathbb{T}} \gamma_t \mathbf{r}_i^t, \mathbf{x}_i \Big\rangle$. We thus have

$$\Big\langle R(\bar{X}^{k+1}), \gamma - \gamma^{k+1} \Big\rangle = \sum_{i\in\mathbb{I}} \Big\langle \bar{\mathbf{x}}_i^{k+1}, \sum_{t\in\mathbb{T}} \mathbf{r}_i^t (\gamma_t - \gamma_t^{k+1}) \Big\rangle. \tag{A.9}$$

Adding (A.5)-(A.8) together, substituting (A.9), and rearranging the terms, we derive

$$\begin{aligned} &\Big(\alpha \sum_{i\in\mathbb{I}} \mathbf{c}_i^\top \mathbf{x}_i^{k+1} + \beta h(\mathbf{y}^{k+1}) - \mathbf{b}^\top \eta + \mathbf{p}^\top \gamma\Big) - \Big(\alpha \sum_{i\in\mathbb{I}} \mathbf{c}_i^\top \mathbf{x}_i + \beta h(\mathbf{y}) - \mathbf{b}^\top \eta^{k+1} + \mathbf{p}^\top \gamma^{k+1}\Big) \\ \le &\frac{1}{\tau}\Big(\sum_{i\in\mathbb{I}} \Theta_\psi(\mathbf{x}_i, \mathbf{x}_i^k, \mathbf{x}_i^{k+1}) + \frac{1}{2}\Theta(\mathbf{y}, \mathbf{y}^k, \mathbf{y}^{k+1})\Big) + \frac{1}{2\sigma}\big(\Theta(\eta, \eta^k, \eta^{k+1}) \\ &+ \Theta(\gamma, \gamma^k, \gamma^{k+1})\big) + \sum_{i\in\mathbb{I}} \Big\langle M_i \eta^k - \sum_{t\in\mathbb{T}} \gamma_t^k \mathbf{r}_i^t, \mathbf{x}_i - \mathbf{x}_i^{k+1} \Big\rangle + \Big\langle \gamma^k, \mathbf{y} - \mathbf{y}^{k+1} \Big\rangle \\ &- \Big\langle \bar{\mathbf{y}}^{k+1}, \gamma - \gamma^{k+1} \Big\rangle + \sum_{i\in\mathbb{I}} \Big\langle \bar{\mathbf{x}}_i^{k+1}, \sum_{t\in\mathbb{T}} \mathbf{r}_i^t (\gamma_t - \gamma_t^{k+1}) - M_i(\eta - \eta^{k+1}) \Big\rangle. \end{aligned} \tag{A.10}$$

Adding

$$\sum_{i\in\mathbb{I}} \Big\langle M_i \eta - \sum_{t\in\mathbb{T}} \gamma_t \mathbf{r}_i^t, \mathbf{x}_i^{k+1} \Big\rangle + \gamma^\top \mathbf{y}^{k+1} - \sum_{i\in\mathbb{I}} \Big\langle M_i \eta^{k+1} - \sum_{t\in\mathbb{T}} \gamma_t^{k+1} \mathbf{r}_i^t, \mathbf{x}_i \Big\rangle - \mathbf{y}^\top \gamma^{k+1},$$

to the both sides of (A.10), then we arrive at

$$\begin{aligned}
&L(X^{k+1},\mathbf{y}^{k+1},\eta,\gamma)-L(X,\mathbf{y},\eta^{k+1},\gamma^{k+1})\\
\leq&\frac{1}{\tau}\left(\sum_{i\in\mathbb{I}}\Theta_{\psi}(\mathbf{x}_i,\mathbf{x}_i^k,\mathbf{x}_i^{k+1})+\frac{1}{2}\Theta(\mathbf{y},\mathbf{y}^k,\mathbf{y}^{k+1})\right)+\frac{1}{2\sigma}\big(\Theta(\eta,\eta^k,\eta^{k+1})\\
&+\Theta(\gamma,\gamma^k,\gamma^{k+1})\big)+\left\langle\gamma^k-\gamma^{k+1},\mathbf{y}-\mathbf{y}^{k+1}\right\rangle+\left\langle\gamma-\gamma^{k+1},\mathbf{y}^k-\mathbf{y}^{k+1}\right\rangle\\
&+\sum_{i\in\mathbb{I}}\left\langle M_i(\eta^k-\eta^{k+1})-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^k-\gamma_t^{k+1}),\mathbf{x}_i-\mathbf{x}_i^{k+1}\right\rangle\\
&+\sum_{i\in\mathbb{I}}\left\langle M_i(\eta-\eta^{k+1})-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t-\gamma_t^{k+1}),\mathbf{x}_i^k-\mathbf{x}_i^{k+1}\right\rangle.
\end{aligned}\tag{A.11}$$

Finally, we obtain (A.4) by some trivial calculating with (A.11). □

The following lemma establishes an essential inequality for the iterate convergence and the ergodic sublinear convergence rate.

Lemma A.4. *Suppose that the primal and dual stepsizes $\tau,\sigma>0$ satisfy (4.11). For any given $(\hat{X},\hat{\mathbf{y}},\hat{\eta},\hat{\gamma})$ and $(X,\mathbf{y},\eta,\gamma)\in\Omega$, there holds*

$$\begin{aligned}
&\left|\sum_{i\in\mathbb{I}}\left\langle M_i(\hat{\eta}-\eta)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\hat{\gamma}_t-\gamma_t),\hat{\mathbf{x}}_i-\mathbf{x}_i\right\rangle+\left\langle\hat{\gamma}-\gamma,\hat{\mathbf{y}}-\mathbf{y}\right\rangle\right|\\
\leq&\frac{1}{\tau}\left(\sum_{i\in\mathbb{I}}D_{\psi}(\hat{\mathbf{x}}_i,\mathbf{x}_i)+\frac{1}{2}\|\hat{\mathbf{y}}-\mathbf{y}\|^2\right)+\frac{1}{2\sigma}\big(\|\hat{\eta}-\eta\|^2+\|\hat{\gamma}-\gamma\|^2\big).
\end{aligned}\tag{A.12}$$

Proof. For each $i\in\mathbb{I}$, using the Cauchy-Schwarz inequality and triangle inequality, we obtain

$$\begin{aligned}
&\left|\left\langle M_i(\hat{\eta}-\eta)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\hat{\gamma}_t-\gamma_t),\hat{\mathbf{x}}_i-\mathbf{x}_i\right\rangle\right|\\
\leq&(\|M_i(\hat{\eta}-\eta)\|+\|\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\hat{\gamma}_t-\gamma_t)\|)\|\hat{\mathbf{x}}_i-\mathbf{x}_i\|\\
\leq&(\|M_i(\hat{\eta}-\eta)\|+\sqrt{T}\delta_i\|\hat{\gamma}-\gamma\|)\|\hat{\mathbf{x}}_i-\mathbf{x}_i\|\\
\leq&\tau\|M_i\|^2\|\hat{\eta}-\eta\|^2+\tau T\delta_i^2\|\hat{\gamma}-\gamma\|^2+\frac{1}{2\tau}\|\hat{\mathbf{x}}_i-\mathbf{x}_i\|^2,
\end{aligned}\tag{A.13}$$

where the second inequality follows from

$$\|\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\hat{\gamma}_t-\gamma_t)\|\leq\sum_{t\in\mathbb{T}}\|\mathbf{r}_i^t\|\|\hat{\gamma}_t-\gamma_t|\leq\delta_i\|\hat{\gamma}-\gamma\|_1\leq\sqrt{T}\delta_i\|\hat{\gamma}-\gamma\|,$$

in which $\delta_i=\max_{t\in\mathbb{T}}\|\mathbf{r}_i^t\|$. The third inequality follows from Young's inequality. Similarly, using Cauchy-Schwarz inequality again, it follows that

$$\left|\left\langle\hat{\gamma}-\gamma,\hat{\mathbf{y}}-\mathbf{y}\right\rangle\right|\leq\|\hat{\gamma}-\gamma\|\|\hat{\mathbf{y}}-\mathbf{y}\|\leq\frac{\tau}{2}\|\hat{\gamma}-\gamma\|^2+\frac{1}{2\tau}\|\hat{\mathbf{y}}-\mathbf{y}\|^2.\tag{A.14}$$

Summing (A.13) over $i \in \mathbb{I}$ and adding (A.14), we obtain

$$\begin{aligned}&\sum_{i\in\mathbb{I}}\left|\left\langle M_i(\hat{\eta}-\eta)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\hat{\gamma}_t-\gamma_t),\hat{\mathbf{x}}_i-\mathbf{x}_i\right\rangle\right|+\left|\left\langle\hat{\gamma}-\gamma,\hat{\mathbf{y}}-\mathbf{y}\right\rangle\right|\\&\leq\frac{1}{2\tau}\sum_{i\in\mathbb{I}}\|\hat{\mathbf{x}}_i-\mathbf{x}_i\|^2+\frac{1}{2\tau}\|\hat{\mathbf{y}}-\mathbf{y}\|^2+\left(\tau\sum_{i\in\mathbb{I}}\|M_i\|^2\right)\|\hat{\eta}-\eta\|^2+\left(\tau T\sum_{i\in\mathbb{I}}\delta_i^2+\frac{\tau}{2}\right)\|\hat{\gamma}-\gamma\|^2.\end{aligned}\tag{A.15}$$

The proof is completed by combining (A.15), (4.11) and

$$\frac{1}{2}\|\hat{\mathbf{x}}_i-\mathbf{x}_i\|^2\leq\frac{1}{2}\|\hat{\mathbf{x}}_i-\mathbf{x}_i\|_1^2\leq D_\psi(\hat{\mathbf{x}}_i,\mathbf{x}_i),$$

which is derived from Lemma A.2. □

A.2. **Proof of Theorem 4.1.**

Proof. Set $X=X^*,\mathbf{y}=\mathbf{y}^*,\eta=\eta^*$ and $\gamma=\gamma^*$ in (A.4). Recall the notation $\Theta_\psi(\mathbf{u}_3,\mathbf{u}_2,\mathbf{u}_1)$ and $\Theta(\mathbf{u}_3,\mathbf{u}_2,\mathbf{u}_1)$. Further rearranging the terms, yields

$$\begin{aligned}&L(X^{k+1},\mathbf{y}^{k+1},\eta^*,\gamma^*)-L(X^*,\mathbf{y}^*,\eta^{k+1},\gamma^{k+1})\\\leq&\frac{1}{\tau}\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^k-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\left(\|\eta^k-\eta^*\|^2+\|\gamma^k-\gamma^*\|^2\right)\\&-\sum_{i\in\mathbb{I}}\left\langle M_i(\eta^*-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^k),\mathbf{x}_i^*-\mathbf{x}_i^k\right\rangle-\left\langle\gamma^*-\gamma^k,\mathbf{y}^*-\mathbf{y}^k\right\rangle\\&-\left(\frac{1}{\tau}\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^{k+1})+\frac{1}{2\tau}\|\mathbf{y}^{k+1}-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\left(\|\eta^{k+1}-\eta^*\|^2+\|\gamma^{k+1}-\gamma^*\|^2\right)\right)\\&+\sum_{i\in\mathbb{I}}\left\langle M_i(\eta^*-\eta^{k+1})-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^{k+1}),\mathbf{x}_i^*-\mathbf{x}_i^{k+1}\right\rangle+\left\langle\gamma^*-\gamma^{k+1},\mathbf{y}^*-\mathbf{y}^{k+1}\right\rangle\\&-\left(\frac{1}{\tau}\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i^{k+1},\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^{k+1}-\mathbf{y}^k\|^2+\frac{1}{2\sigma}\left(\|\eta^{k+1}-\eta^k\|^2+\|\gamma^{k+1}-\gamma^k\|^2\right)\right)\\&+\sum_{i\in\mathbb{I}}\left\langle M_i(\eta^k-\eta^{k+1})-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^k-\gamma_t^{k+1}),\mathbf{x}_i^k-\mathbf{x}_i^{k+1}\right\rangle+\left\langle\gamma^k-\gamma^{k+1},\mathbf{y}^k-\mathbf{y}^{k+1}\right\rangle.\end{aligned}\tag{A.16}$$

Setting $(\widehat{X},X)=(X^{k+1},X^k),(\hat{\mathbf{y}},\mathbf{y})=(\mathbf{y}^{k+1},\mathbf{y}^k),(\hat{\eta},\eta)=(\eta^{k+1},\eta^k)$ and $(\hat{\gamma},\gamma)=(\gamma^{k+1},\gamma^k)$ in (A.12), we have

$$\begin{aligned}&\sum_{i\in\mathbb{I}}\left\langle M_i(\eta^{k+1}-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^{k+1}-\gamma_t^k),\mathbf{x}_i^{k+1}-\mathbf{x}_i^k\right\rangle+\left\langle\gamma^{k+1}-\gamma^k,\mathbf{y}^{k+1}-\mathbf{y}^k\right\rangle\\\leq&\frac{1}{\tau}\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i^{k+1},\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^{k+1}-\mathbf{y}^k\|^2+\frac{1}{2\sigma}\left(\|\eta^{k+1}-\eta^k\|^2+\|\gamma^{k+1}-\gamma^k\|^2\right).\end{aligned}\tag{A.17}$$

Further, substituting (A.17) into (A.16) and then taking a telescope sum over $k=0,1,\ldots,N-1$, we deduce

$$\sum_{k=0}^{N-1}\left(L(X^{k+1},\mathbf{y}^{k+1},\eta^*,\gamma^*)-L(X^*,\mathbf{y}^*,\eta^{k+1},\gamma^{k+1})\right) \tag{A.18a}$$

$$\leq \frac{1}{\tau}\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^0)+\frac{1}{2\tau}\|\mathbf{y}^0-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\left(\|\eta^0-\eta^*\|^2+\|\gamma^0-\gamma^*\|^2\right) \tag{A.18b}$$

$$-\sum_{i\in\mathbb{I}}\left\langle M_i(\eta^*-\eta^0)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^0),\mathbf{x}_i^*-\mathbf{x}_i^0\right\rangle-\left\langle\gamma^*-\gamma^0,\mathbf{y}^*-\mathbf{y}^0\right\rangle \tag{A.18c}$$

$$-\left(\frac{1}{\tau}\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^N)+\frac{1}{2\tau}\|\mathbf{y}^N-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\left(\|\eta^N-\eta^*\|^2+\|\gamma^N-\gamma^*\|^2\right)\right) \tag{A.18d}$$

$$+\sum_{i\in\mathbb{I}}\left\langle M_i(\eta^*-\eta^N)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^N),\mathbf{x}_i^*-\mathbf{x}_i^N\right\rangle+\left\langle\gamma^*-\gamma^N,\mathbf{y}^*-\mathbf{y}^N\right\rangle. \tag{A.18e}$$

It follows from (A.12) that (A.18d) + (A.18e) ≤ 0 and

$$\text{(A.18c)}\leq\frac{1}{\tau}\left(\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^0)+\frac{1}{2}\|\mathbf{y}^0-\mathbf{y}^*\|^2\right)+\frac{1}{2\sigma}\left(\|\eta^0-\eta^*\|^2+\|\gamma^0-\gamma^*\|^2\right).$$

Then, leveraging the convexity of $L(X,\mathbf{y},\eta,\gamma)$ in $(X,\mathbf{y})$ and the concavity in (η,γ), along with the definition of primal-dual gap function, we derive (4.12). □

If, in addition, we assume that (4.11) is strict, i.e.,

$$\tau\sigma<1/\max\left\{2\textstyle\sum_{i\in\mathbb{I}}\|M_i\|^2,\,1+2T\sum_{i\in\mathbb{I}}\delta_i^2\right\}. \tag{A.19}$$

Then the iterate convergence of Algorithm 1 is guaranteed.

A.3. **Proof of Theorem 4.2.**

Proof. Let $(X^*,\mathbf{y}^*,\eta^*,\gamma^*)$ be a saddle point of (3.2). Then, we have

$$L(X^{k+1},\mathbf{y}^{k+1},\eta^*,\gamma^*)-L(X^*,\mathbf{y}^*,\eta^{k+1},\gamma^{k+1})\geq 0. \tag{A.20}$$

Further substituting (A.17) and (A.20) into (A.16), we arrive at

$$\begin{aligned}&\frac{1}{\tau}\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^{k+1})+\frac{1}{2\tau}\|\mathbf{y}^{k+1}-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\left(\|\eta^{k+1}-\eta^*\|^2+\|\gamma^{k+1}-\gamma^*\|^2\right)\\&-\sum_{i\in\mathbb{I}}\left\langle M_i(\eta^*-\eta^{k+1})-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^{k+1}),\mathbf{x}_i^*-\mathbf{x}_i^{k+1}\right\rangle-\left\langle\gamma^*-\gamma^{k+1},\mathbf{y}^*-\right.\\&\left.\mathbf{y}^{k+1}\right\rangle\leq\frac{1}{\tau}\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^k-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\left(\|\eta^k-\eta^*\|^2+\|\gamma^k-\gamma^*\|^2\right)\\&-\sum_{i\in\mathbb{I}}\left\langle M_i(\eta^*-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^k),\mathbf{x}_i^*-\mathbf{x}_i^k\right\rangle-\left\langle\gamma^*-\gamma^k,\mathbf{y}^*-\mathbf{y}^k\right\rangle.\end{aligned} \tag{A.21}$$

This implies for any $k \geq 1$ that

$$\begin{aligned}
&\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^k-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\big(\|\eta^k-\eta^*\|^2+\|\gamma^k-\gamma^*\|^2\big)\\
&-\sum_{i\in\mathbb{I}}\Big\langle M_i(\eta^*-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^k),\mathbf{x}_i^*-\mathbf{x}_i^k\Big\rangle-\Big\langle\gamma^*-\gamma^k,\mathbf{y}^*-\mathbf{y}^k\Big\rangle\\
\leq&\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^0)+\frac{1}{2\tau}\|\mathbf{y}^0-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\big(\|\eta^0-\eta^*\|^2+\|\gamma^0-\gamma^*\|^2\big)\\
&-\sum_{i\in\mathbb{I}}\Big\langle M_i(\eta^*-\eta^0)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^0),\mathbf{x}_i^*-\mathbf{x}_i^0\Big\rangle-\Big\langle\gamma^*-\gamma^0,\mathbf{y}^*-\mathbf{y}^0\Big\rangle\\
\leq&\frac{2}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^0)+\frac{1}{\tau}\|\mathbf{y}^0-\mathbf{y}^*\|^2+\frac{1}{\sigma}(\|\eta^0-\eta^*\|^2+\|\gamma^0-\gamma^*\|^2),
\end{aligned}\tag{A.22}$$

where the last inequality follows from (A.12).

Set $\zeta=\tau/\sigma$ and $\vartheta=\sigma\max\Big\{\sqrt{2\zeta\sum_{i\in\mathbb{I}}\|M_i\|^2},\sqrt{(1+2T\sum_{i\in\mathbb{I}}\delta_i^2)\zeta}\Big\}$. Then, by applying the Cauchy-Schwarz inequality, it is elementary to derive

$$\begin{aligned}
&\sigma\sum_{i\in\mathbb{I}}\Big\langle M_i(\eta^*-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^k),\mathbf{x}_i^*-\mathbf{x}_i^k\Big\rangle+\sigma\Big\langle\gamma^*-\gamma^k,\mathbf{y}^*-\mathbf{y}^k\Big\rangle\\
\leq&\sum_{i\in\mathbb{I}}\sigma(\|M_i\|\|\eta^*-\eta^k\|+\sqrt{T}\delta_i\|\gamma^*-\gamma^k\|)\|\mathbf{x}_i^*-\mathbf{x}_i^k\|+\sigma\|\gamma^*-\gamma^k\|\|\mathbf{y}^*-\mathbf{y}^k\|\\
\leq&\sum_{i\in\mathbb{I}}\left(\frac{\vartheta\|M_i\|^2}{2\sum_{i\in\mathbb{I}}\|M_i\|^2}\|\eta^*-\eta^k\|^2+\frac{\vartheta T\delta_i^2}{1+2T\sum_{i\in\mathbb{I}}\delta_i^2}\|\gamma^*-\gamma^k\|^2+\frac{\vartheta}{2\zeta}\|\mathbf{x}_i^*-\mathbf{x}_i^k\|^2\right)\\
&+\frac{\vartheta}{2(1+2T\sum_{i\in\mathbb{I}}\delta_i^2)}\|\gamma^*-\gamma^k\|^2+\frac{\vartheta}{2\zeta}\|\mathbf{y}^*-\mathbf{y}^k\|^2\\
=&\frac{\vartheta}{2\zeta}\sum_{i\in\mathbb{I}}\|\mathbf{x}_i^*-\mathbf{x}_i^k\|^2+\frac{\vartheta}{2\zeta}\|\mathbf{y}^*-\mathbf{y}^k\|^2+\frac{\vartheta}{2}(\|\eta^*-\eta^k\|^2+\|\gamma^*-\gamma^k\|^2)\\
\leq&\frac{\vartheta}{\zeta}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^k)+\frac{\vartheta}{2\zeta}\|\mathbf{y}^*-\mathbf{y}^k\|^2+\frac{\vartheta}{2}(\|\eta^*-\eta^k\|^2+\|\gamma^*-\gamma^k\|^2),
\end{aligned}$$

which further implies

$$\begin{aligned}
&\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^k-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\big(\|\eta^k-\eta^*\|^2+\|\gamma^k-\gamma^*\|^2\big)\\
&-\sum_{i\in\mathbb{I}}\Big\langle M_i(\eta^*-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^k),\mathbf{x}_i^*-\mathbf{x}_i^k\Big\rangle-\Big\langle\gamma^*-\gamma^k,\mathbf{y}^*-\mathbf{y}^k\Big\rangle\\
\geq&(1-\vartheta)\left(\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^k-\mathbf{y}^*\|^2+\frac{1}{2\sigma}(\|\eta^k-\eta^*\|^2+\|\gamma^k-\gamma^*\|^2)\right).
\end{aligned}\tag{A.23}$$

Similarly, for any $k \geq 0$, we have

$$\begin{aligned}
&\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^{k+1},\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^k-\mathbf{y}^{k+1}\|^2+\frac{1}{2\sigma}\big(\|\eta^k-\eta^{k+1}\|^2+\|\gamma^k-\gamma^{k+1}\|^2\big)-\\
&\sum_{i\in\mathbb{I}}\Big\langle M_i(\eta^{k+1}-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^{k+1}-\gamma_t^k),\mathbf{x}_i^{k+1}-\mathbf{x}_i^k\Big\rangle-\Big\langle\gamma^{k+1}-\gamma^k,\mathbf{y}^{k+1}-\mathbf{y}^k\Big\rangle\\
\geq&(1-\vartheta)\left(\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^{k+1},\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^k-\mathbf{y}^{k+1}\|^2+\frac{1}{2\sigma}\|\eta^k-\eta^{k+1}\|^2\right)\\
&+\frac{1-\vartheta}{2\sigma}\|\gamma^k-\gamma^{k+1}\|^2.
\end{aligned}\tag{A.24}$$

Then, substituting (A.24) into (A.16) and combining with(A.20), we arrive at

$$\begin{aligned}
&(1-\vartheta)\left(\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^{k+1},\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^k-\mathbf{y}^{k+1}\|^2+\frac{1}{2\sigma}(\|\eta^k-\eta^{k+1}\|^2+\|\gamma^{k+1}-\gamma^k\|^2)\right)\\
&\leq\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^k-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\big(\|\eta^k-\eta^*\|^2+\|\gamma^k-\gamma^*\|^2\big)\\
&-\sum_{i\in\mathbb{I}}\Big\langle M_i(\eta^*-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^k),\mathbf{x}_i^*-\mathbf{x}_i^k\Big\rangle-\Big\langle\gamma^*-\gamma^k,\mathbf{y}^*-\mathbf{y}^k\Big\rangle\\
&-\left(\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^{k+1})+\frac{1}{2\tau}\|\mathbf{y}^{k+1}-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\big(\|\eta^{k+1}-\eta^*\|^2+\|\gamma^{k+1}-\gamma^*\|^2\big)\right)\\
&+\sum_{i\in\mathbb{I}}\Big\langle M_i(\eta^*-\eta^{k+1})-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^*-\gamma_t^{k+1}),\mathbf{x}_i^*-\mathbf{x}_i^{k+1}\Big\rangle+\Big\langle\gamma^*-\gamma^{k+1},\mathbf{y}^*-\mathbf{y}^{k+1}\Big\rangle.
\end{aligned}\tag{A.25}$$

Taking a sum of (A.25) over $k=0,1,\ldots,N-1$, and then using (A.12) and omitting the non-positive term, we have

$$\begin{aligned}
&(1-\vartheta)\sum_{k=0}^{N-1}\left(\frac{1}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^{k+1},\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^{k+1}-\mathbf{y}^k\|^2+\frac{1}{2\sigma}(\|\eta^{k+1}-\eta^k\|^2+\|\gamma^{k+1}-\gamma^k\|^2)\right)\\
&\leq\frac{2}{\tau}\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^0)+\frac{1}{\tau}\|\mathbf{y}^0-\mathbf{y}^*\|^2+\frac{1}{\sigma}(\|\eta^0-\eta^*\|^2+\|\gamma^0-\gamma^*\|^2).
\end{aligned}$$

According to (A.19), we have $\vartheta\in(0,1)$. Then, the above inequality implies that the sequences $\{\sum_{i\in\mathbb{I}} D_\psi(\mathbf{x}_i^{k+1},\mathbf{x}_i^k)\}$, $\{\|\mathbf{y}^{k+1}-\mathbf{y}^k\|^2\}$, $\{\|\eta^{k+1}-\eta^k\|^2\}$ and $\{\|\gamma^{k+1}-\gamma^k\|^2\}$ are all summable. On the other hand, it is easy to deduce that the sequence $\{(X^k,\mathbf{y}^k,\eta^k,\gamma^k)\}$ is bounded by substituting (A.23) into (A.22). Therefore, $(\tilde{X}^N,\tilde{\mathbf{y}}^N,\tilde{\eta}^N,\tilde{\gamma}^N)$ is also bounded. Assume that a subsequence $\{(\tilde{X}^{N_l},\tilde{\mathbf{y}}^{N_l},\tilde{\eta}^{N_l},\tilde{\gamma}^{N_l})\}$ of $(\tilde{X}^N,\tilde{\mathbf{y}}^N,\tilde{\eta}^N,\tilde{\gamma}^N)$ converges in Ω to $(\tilde{X},\tilde{\mathbf{y}},\tilde{\eta},\tilde{\gamma})$. From (4.12) and (A.3), it follows that the limit $(\tilde{X},\tilde{\mathbf{y}},\tilde{\eta},\tilde{\gamma})$ is a saddle point as well.

We next show the convergence of the whole sequence $\{(X^k,\mathbf{y}^k,\eta^k,\gamma^k)\}$ to a saddle point of problem (3.2). Let $\{(X^{k_l},\mathbf{y}^{k_l},\eta^{k_l},\gamma^{k_l})\}$ be a subsequence of $\{(X^k,\mathbf{y}^k,\eta^k,\gamma^k)\}$ that converges to

$(X',\mathbf{y}',\eta',\gamma')$. According to (A.11), for any given $(X,\mathbf{y},\eta,\gamma)\in\Omega$, there holds

$$\begin{aligned}
&L(X^{k_l+1},\mathbf{y}^{k_l+1},\eta,\gamma)-L(X,\mathbf{y},\eta^{k_l+1},\gamma^{k_l+1})\\
&\le\frac{1}{\tau}\sum_{i\in\mathbb{I}}\Theta_\psi(\mathbf{x}_i,\mathbf{x}_i^{k_l},\mathbf{x}_i^{k_l+1})+\frac{1}{2\tau}\Theta(\mathbf{y},\mathbf{y}^{k_l},\mathbf{y}^{k_l+1})+\frac{1}{2\sigma}\big(\Theta(\eta,\eta^{k_l},\eta^{k_l+1})\\
&+\Theta(\gamma,\gamma^{k_l},\gamma^{k_l+1})\big)+\Big\langle\gamma^{k_l}-\gamma^{k_l+1},\mathbf{y}-\mathbf{y}^{k_l+1}\Big\rangle+\Big\langle\gamma-\gamma^{k_l+1},\mathbf{y}^{k_l}-\mathbf{y}^{k_l+1}\Big\rangle\\
&+\sum_{i\in\mathbb{I}}\Big\langle M_i(\eta^{k_l}-\eta^{k_l+1})-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t^{k_l}-\gamma_t^{k_l+1}),\mathbf{x}_i-\mathbf{x}_i^{k_l+1}\Big\rangle\\
&+\sum_{i\in\mathbb{I}}\Big\langle M_i(\eta-\eta^{k_l+1})-\sum_{t\in\mathbb{T}}\mathbf{r}_i^t(\gamma_t-\gamma_t^{k_l+1}),\mathbf{x}_i^{k_l}-\mathbf{x}_i^{k_l+1}\Big\rangle.
\end{aligned}\tag{A.26}$$

Note that $\lim_{l\to\infty}\Theta(\mathbf{y},\mathbf{y}^{k_l},\mathbf{y}^{k_l+1})=\lim_{l\to\infty}\Theta(\eta,\eta^{k_l},\eta^{k_l+1})=\lim_{l\to\infty}\Theta(\gamma,\gamma^{k_l},\gamma^{k_l+1})=0$, and for each $i\in\mathbb{I}$, it follows from (A.3) that $\lim_{l\to\infty}\Theta_\psi(\mathbf{x}_i,\mathbf{x}_i^{k_l},\mathbf{x}_i^{k_l+1})=0$. Furthermore, taking the limit over both sides of (A.26), we have

$$L(X',\mathbf{y}',\eta,\gamma)-L(X,\mathbf{y},\eta',\gamma')\le 0,\ \forall\ (X,\mathbf{y},\eta,\gamma)\in\Omega,$$

which shows that $(X',\mathbf{y}',\eta',\gamma')$ is a saddle point of (3.2).

So far, it remains to show that $(X',\mathbf{y}',\eta',\gamma')$ is the only limit point of $\{(X^k,\mathbf{y}^k,\eta^k,\gamma^k)\}$. It is easy to deduce from (A.21) that $\Phi^k(X^*,\mathbf{y}^*,\eta^*,\gamma^*)$ is nonincreasing with

$$\begin{aligned}
\Phi^k(X^*,\mathbf{y}^*,\eta^*,\gamma^*):=&\frac{1}{\tau}\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i^*,\mathbf{x}_i^k)+\frac{1}{2\tau}\|\mathbf{y}^k-\mathbf{y}^*\|^2+\frac{1}{2\sigma}\|\eta^k-\eta^*\|^2+\frac{1}{2\sigma}\|\gamma^k-\gamma^*\|^2\\
&-\sum_{i\in\mathbb{I}}\Big\langle M_i(\eta^*-\eta^k)-\sum_{t\in\mathbb{T}}\mathbf{r}_i(\gamma_t^*-\gamma_t^k),\mathbf{x}_i^*-\mathbf{x}_i^k\Big\rangle-\Big\langle\gamma^*-\gamma^k,\mathbf{y}^*-\mathbf{y}^k\Big\rangle.
\end{aligned}$$

Since $\lim_{l\to\infty}(X^{k_l},\mathbf{y}^{k_l},\eta^{k,l},\gamma^{k,l})=(X',\mathbf{y}',\eta',\gamma')$, and together with (A.3), we have

$$\lim_{l\to\infty}\Phi^{k_l}(X',\mathbf{y}',\eta',\gamma')=0,$$

which implies $\lim_{k\to\infty}\Phi^k(X',\mathbf{y}',\eta',\gamma')=0$. Furthermore, we derive that

$$\lim_{k\to\infty}\sum_{i\in\mathbb{I}}D_\psi(\mathbf{x}_i',\mathbf{x}_i^k)=\lim_{k\to\infty}\|\mathbf{y}^k-\mathbf{y}'\|^2=\lim_{k\to\infty}\|\eta^k-\eta'\|^2=\lim_{\infty}\|\gamma^k-\gamma'\|^2=0.$$

Thus, considering $\frac{1}{2}\|\mathbf{u}-\mathbf{v}\|^2\le D_\psi(\mathbf{u},\mathbf{v})$, we complete the proof. □