



Primal-Dual Algorithm for Distributed Optimization with Coupled Constraints

Kai Gong¹ · Liwei Zhang¹

Received: 12 July 2022 / Accepted: 22 January 2024 / Published online: 13 March 2024

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

Abstract

This paper focuses on distributed consensus optimization problems with coupled constraints over time-varying multi-agent networks, where the global objective is the finite sum of all agents' private local objective functions, and decision variables of agents are subject to coupled equality and inequality constraints and a compact convex subset. Each agent exchanges information with its neighbors and processes local data. They cooperate to agree on a consensual decision vector that is an optimal solution to the considered optimization problems. We integrate ideas behind dynamic average consensus and primal-dual methods to develop a distributed algorithm and establish its sublinear convergence rate. In numerical simulations, to illustrate the effectiveness of the proposed algorithm, we compare it with some related methods by the Neyman–Pearson classification problem.

Keywords Distributed optimization · Coupled constraints · Time-varying networks · Convergence

1 Introduction

Distributed algorithms decompose an optimization problem into smaller and more manageable subproblems that can be solved in parallel by a group of agents or proces-

Communicated by Jalal M. Fadili.

Liwei Zhang: This author was supported by the Natural Science Foundation of China (Nos. 11971089 and 11731013) and partially supported by Dalian High-level Talent Innovation Project (No. 2020RD09).

✉ Kai Gong
gk1995___@mail.dlut.edu.cn

Liwei Zhang
lwzhang@dlut.edu.cn

¹ School of Mathematical Sciences, Dalian University of Technology, Liaoning 116024, China

sors. Such algorithms are used for large-scale problems in machine learning, wireless sensor networks, social networks, smart grids, etc.

To date, a large number of distributed algorithms have been developed for distributed unconstrained optimization problems. In general, these algorithms can be classified into two categories: dual-decomposition-based algorithms and dynamic-average-consensus-based algorithms. The dual-decomposition-based algorithms utilize a consensus equality constraint to reformulate unconstrained optimization problems into constrained ones such that decision variables of agents are decoupled, then integrate ideas from centralized primal-dual methods. Typical of such algorithms are EXTRA [40], PG-EXTRA [41], NIDS [18], ABC [48], and DCPA [5], etc. On the other hand, the dynamic-average-consensus-based algorithms introduce auxiliary variables to track the average of some quantities that need to reach a consensus among agents, such as tracking the average gradients of all agents’ local objectives. To name a few, Aug-DGM [49], DIGing [26], Push–Pull Gradient [35], Harnessing Smoothness [36], and TV- \mathcal{AB} [39], etc. Ideas behind these methods are generalized to design algorithms for solving distributed constrained optimization problems arising in practical applications.

In this paper, we consider the following constrained consensus optimization problem over a multi-agent network consisting of m agents,

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad & f(x) = \sum_{i=1}^m f_i(x) & \text{(DCOP)} \\ \text{s.t.} \quad & h(x) = \sum_{i=1}^m h_i(x) \in \mathcal{K}, \end{aligned}$$

where $\mathcal{K} = \mathbb{R}^p_- \times \{0_q\}$, implying that the constraints include inequality and equality. $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $h_i : \mathbb{R}^n \rightarrow \mathbb{R}^{p+q}$, $i = 1, 2, \dots, m$, and \mathcal{X} is a subset of \mathbb{R}^n . For each $i = 1, 2, \dots, m$, the local objective f_i and the constraint function h_i are only known by agent i , but are not available to other agents. All agents cooperate to find an optimal solution to (DCOP) through peer-to-peer communication and local computation.

1.1 Motivations

The optimization problem (DCOP) is motivated by its applications in stochastic programs, distributed optimization, optimal control, etc. Due to space limitations, we will present two important examples of applications.

Example 1 Consider the following stochastic programming with expectation constraints,

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad & f(x) = \mathbb{E}_\xi[F(x, \xi)] \\ \text{s.t.} \quad & g(x) = \mathbb{E}_\xi[G(x, \xi)] \leq 0, \end{aligned} \tag{1.1}$$

where ξ is a random vector supported on $\mathcal{P} \subseteq \mathfrak{R}^n$ and independent of x . $F(x, \xi) : \mathfrak{R}^p \times \mathcal{P} \rightarrow \mathfrak{R}$ and $G(x, \xi) : \mathfrak{R}^p \times \mathcal{P} \rightarrow \mathfrak{R}^q$. The problem (1.1) has many applications in machine learning, finance and data analysis, and operations research, e.g., Neyman–Pearson classification [44] and asset allocation problems [15, 38].

The sample average approximation (SAA) method is widely used for stochastic optimization problems. Applying SAA to solve (1.1), one first generates N independently and identically distributed (i.i.d.) random samples $\{\xi_i\}_{i=1}^N$ and then uses the averages $\frac{1}{N} \sum_{i=1}^N F(x, \xi_i)$ and $\frac{1}{N} \sum_{i=1}^N G(x, \xi_i)$ to approximate $f(x)$ and $g(x)$, respectively. The associated approximation optimization subproblem is

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad & \frac{1}{N} \sum_{i=1}^N F(x, \xi_i) \\ \text{s.t.} \quad & \frac{1}{N} \sum_{i=1}^N G(x, \xi_i) \leq 0. \end{aligned} \tag{1.2}$$

It is well known that optimal solutions of (1.2) asymptotically solve (1.1) as the sample size N tends to infinity. An immediately following question is how to solve such subproblems.

Example 2 If the objective $f(x)$ and constraint function $h(x)$ of (DCOP) both have separable structures, namely, $x = (x_1, x_2, \dots, x_m)$, $f_i(x) = \tilde{f}_i(x_i)$, and $h_i(x) = \tilde{h}_i(x_i)$, (DCOP) reduces to the following constraint-coupled optimization problem,

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad & \sum_{i=1}^m \tilde{f}_i(x_i) \\ \text{s.t.} \quad & \sum_{i=1}^m \tilde{h}_i(x_i) \in \mathcal{K}. \end{aligned} \tag{CCOP}$$

The problem (CCOP) has been widely investigated in smart grids, optimal control, machine learning, etc. For instance, in multi-micro energy grid systems [51], the coupled constraints character that the sum of users' power consumption is equal to the total power generation of grids and the total carbon emissions are not more than a given upper bound. In distributed resource allocation problems [27, 33], the relation of allocating some given resource to agents is formulated as the coupled constraints. Besides, (CCOP) also appears in problems such as distributed model predictive control [6], network utility maximization [20], and distributed estimation [13], etc.

Theoretically, algorithms for (CCOP) can be adopted to solve (DCOP) at the cost of adding $m - 1$ equality constraints, namely, $x_1 = x_2 = \dots = x_m$. However, in practical computations, this causes challenges if the network size m and the variable dimension n are relatively large, e.g., $m = 1000$, $n = 100$. Moreover, the consensus constraints, $x_i = x_j$, for each $j \in \mathcal{N}_i$ (the set of neighbors of agent i), play an important role in applications such as distributed regression problems [50] and optimal control [24, 45], which motivates us to consider the lifted consensus optimization problem (DCOP).

1.2 Related Work

To accommodate the demands of practical applications, researchers aim at designing distributed algorithms that converge faster and can apply to more general multi-agent networks.

The earlier studies of distributed constrained consensus optimization problems focused on the case where agents' decisions are only subject to local constraint sets. The distributed projection (sub)gradient method [30] is a seminal work in this direction. Afterward, a large number of dual-decomposition-based [16, 52] or dynamic-average-consensus-based [3, 21, 22] distributed algorithms with fast convergence rates have been proposed to solve such problems.

The first attempt to focus on distributed consensus optimization with equality and inequality constraints may be the work of [53], which presented a distributed primal-dual algorithm with vanishing step-sizes. Afterward, the community aims to design faster convergent distributed algorithms with a fixed step-size for such problems, e.g., see [12, 14, 23]. Under various assumptions such as smoothness and strong-convexity of functions, these algorithms have sublinear or linear convergence rates, while their considered equality and inequality constraints are shared with all agents.

The problem (CCOP) has attracted increasing research interest as the coupled constraints frequently appear in many fields. Approaches directly aiming at (CCOP) typically leverage the Lagrangian duality to deal with the coupled constraints, because the Lagrangian has a separable structure in the primal decision variables and the dual problem is a special case of (DCOP). Works based on the dual-decomposition method are references [2, 10, 27, 42], to mention a few. There are also consensus-based distributed algorithms for such problems, e.g., see [1, 8, 9, 22, 43]. While these mentioned papers considered only the linearly coupled constraints, i.e., $\mathcal{K} = \{0_q\}$ and each \tilde{h}_i is affine. The recent works [4, 19, 47] investigated (CCOP) with coupled equality and inequality constraints, and proposed the distributed primal-dual gradient method [19], the integrated primal-dual proximal method [47], and the augmented Lagrangian tracking method [4], in which the integrated primal-dual proximal method has an $O(1/k)$ convergence rate in terms of optimality and feasibility, the other two algorithms asymptotically solve (CCOP) with constant step-sizes but no results on rate analysis.

In general, the dual-decomposition-based algorithms have nice convergence and rates while lacking extensibility to directed and time-varying networks. The dynamic-average-consensus-based algorithms are inexact gradient methods. If the objective and constraint functions do not have good properties, such as strong-convexity or gradient Lipschitz continuity, convergence analysis of these algorithms is difficult. Moreover, such algorithms can be easily extended to directed and time-varying communication networks of multi-agent, but the complex networks also bring challenges in algorithm analysis.

It's worth noting that the distributed algorithms mentioned so far for the problems (CCOP) or (DCOP) are all executed over a fixed (static) multi-agent network. To my knowledge, there is very little work that investigates (DCOP) over time-varying

networks, except for [17]. The authors of [17] proposed a distributed proximal primal-dual algorithm that has an $O(1/\sqrt{k})$ convergence rate for solving (DCOP).

In this paper, we integrate ideas from the algorithm TV- \mathcal{AB} [39] and the primal-dual optimization method to develop a distributed algorithm for solving (DCOP) over time-varying and directed multi-agent networks. The main contributions of the article are threefold. First, this paper considers distributed consensus optimization problems with coupled affine equality and inequality (not necessarily affine) constraints. The objective and constraint functions are only convex and differentiable. Such assumptions are fundamental settings for consensus-based algorithms. The considered problem (DCOP) is more general and challenging than many papers in the literature, e.g., [2, 10, 27]. Second, the investigated multi-agent communication network is more general than the undirected and static ones considered by [2, 10, 19, 27, 42]. Third, The foundational assumptions on the objective and constraint functions, as well as the time-varying communication network of multi-agent bring challenges to analysis of the proposed inexact gradient algorithm, we still establish its convergence and iteration complexity.

The rest of this paper is organized as follows. Section 2 provides necessary preliminaries and constructs the saddle point problem of (DCOP). Section 3 develops a consensus-based distributed primal-dual algorithm. Section 4 presents the main results about the convergence properties of the proposed algorithm. Section 5 simulates the presented algorithm using a realistic example and compares it with some related methods. Finally, Sect. 6 makes some conclusions.

2 Preliminaries

In this section, we will present some preliminaries in graph theory and assumptions on the considered optimization problem and multi-agent network.

2.1 Notion and Notations

At each time slot $k \geq 0$, a multi-agent system consisting of m agents is modeled as a directed graph $\mathcal{G}(k) = (\mathcal{V}, \mathcal{E}(k))$, where $\mathcal{V} = \{1, 2, \dots, m\}$ and $\mathcal{E}(k)$ is the set of directed edges. Let $A(k) = [a_{ij}(k)]$ and $B(k) = [b_{ij}(k)]$ be two matrices that are compatible with the graph $\mathcal{G}(k)$, which means that $(i, j) \in \mathcal{E}(k)$ if and only if $a_{ij}(k) > 0$ and $b_{ij}(k) > 0$. We write $\mathcal{N}_i^{\text{out}}(k) = \{j \in \mathcal{V} : (i, j) \in \mathcal{E}(k)\}$ and $\mathcal{N}_i^{\text{in}}(k) = \{j \in \mathcal{V} : (j, i) \in \mathcal{E}(k)\}$ as the sets of out-neighbors and in-neighbors of agent i , respectively. The symbol $|\mathcal{N}_i^{\text{out}}(k)|$ (resp. $|\mathcal{N}_i^{\text{in}}(k)|$) represents the out-degree (resp. in-degree) of agent i .

The properties of adjacency matrices of connected (or strongly connected) graphs, such as their eigenvalues and eigenvectors, are closely related to stochastic vectors. Lagrangian function and its saddle points are also fundamental concepts in constrained optimization. To clarify the presentation of the considered problem and algorithm analysis, we introduce the following three notions.

Definition 2.1 (Stochastic vector [29]) A vector $\pi \in \mathfrak{N}^m$ is said to be stochastic if its components π_i satisfy that

$$\sum_{i=1}^m \pi_i = 1, \quad \pi_i \geq 0, \quad \forall i = 1, 2, \dots, m.$$

Definition 2.2 (Absolute probability sequence [39]) For row-stochastic matrices $\{\mathcal{R}_k\}$, an absolute probability sequence is a sequence $\{\pi_k\}$ of stochastic vectors such that

$$\pi_k^\top = \pi_{k+1}^\top \mathcal{R}_k, \quad \forall k \geq 0.$$

Definition 2.3 (Saddle point [37]) Consider a function $\mathcal{L} : X \times S \rightarrow \mathfrak{R}$, where X and S are non-empty subsets of \mathfrak{N}^n and \mathfrak{N}^{p+q} respectively, a pair of vectors $(x^*, \lambda^*) \in X \times S$ is called a saddle point of function \mathcal{L} over $X \times S$, if for all $(x, \lambda) \in X \times S$, it holds that

$$\mathcal{L}(x^*, \lambda) \leq \mathcal{L}(x^*, \lambda^*) \leq \mathcal{L}(x, \lambda^*).$$

In this paper, the gradient vector of a function f at x is denoted by $\nabla f(x)$, $\|\cdot\|$ represents the Frobenius-norm of a matrix with suitable dimensions, and $\langle \cdot, \cdot \rangle$ is the inner product matched with the norm $\|\cdot\|$. The projection of a point x onto a subset $\Omega \subseteq \mathfrak{N}^n$ is written as $P_\Omega(x)$. For a non-empty subset Ω , we denote its relative interior by $\text{ri}(\Omega)$. 1_m represents the column vector in \mathfrak{N}^m whose entries are all ones.

2.2 Assumptions and Saddle Point Problem

This paper focuses on time-varying communication networks of multi-agent. We make the following assumptions on the network communication graphs, which are standard for consensus-based algorithms, e.g., see [29, 30, 39].

Assumption 2.1 (Periodical strong connectivity) There exists a positive integer B such that the directed graph $(\mathcal{V}, \bigcup_{t=0}^{B-1} \mathcal{E}(t+k))$ is strongly connected, for all $k \geq 0$.

Assumption 2.2 (Weights) Let $A(k) = [a_{ij}(k)]_{m \times m}$ and $B(k) = [b_{ij}(k)]_{m \times m}$ be two matrices that are compatible with the graph $\mathcal{G}(k)$.

- (a) Stochasticity: Matrices $\{A(k)\}$ and $\{B(k)\}$ are row-stochastic and column-stochastic, respectively.
- (b) The graph $\mathcal{G}(k)$ has self-loops, i.e., $a_{ii}(k) > 0, b_{ii}(k) > 0, \forall i \in V, k \geq 0$.
- (c) Uniform positivity: There is a scalar $\eta \in (0, 1)$ such that $a_{ij}(k) \geq \eta$ and $b_{ij}(k) \geq \eta, \forall (i, j) \in \mathcal{E}(k), k \geq 0$.

The matrices $A(k)$ and $B(k)$ are singly stochastic. Agents can use their out-degrees and in-degrees to construct these two matrices easily, which avoids the difficulty of generating symmetric doubly stochastic adjacency matrices over directed and time-varying networks. Assumption 2.2 (b) implies that each agent performs local computation using information held by itself and received from its in-neighbors. Item (c) indicates that the influence of each agent on the network is not vanishing.

Smooth convex optimization problems are within the range of our consideration, so we make the following standard assumption on (DCOP).

Assumption 2.3 (Convexity and smoothness)

- (a) The local function f_i of agent i is convex and continuous differentiable over the subset \mathcal{X} , $\forall i = 1, 2, \dots, m$.
- (b) The function h_i^j (the j -th coordinate component of h_i) corresponding to the inequality constraints is convex and continuous differentiable, $j = 1, \dots, p$, and h_i^l corresponding to the equality constraints is affine, $l = p + 1, \dots, p + q$, $\forall i = 1, \dots, m$.
- (c) The non-empty subset \mathcal{X} of \mathfrak{R}^n is compact and convex.

To construct a primal-dual method, a fundamental assumption is that the strong duality holds, which is guaranteed by the following Slater’s condition.

Assumption 2.4 (Slater’s condition) Suppose that there exists a point $\bar{x} \in \text{ri}(\mathcal{X})$, such that $\sum_{i=1}^m h_i(\bar{x}) \in \text{ri}(\mathcal{K})$.

Let \mathcal{K}° represent the polar of \mathcal{K} , namely, $\mathcal{K}^\circ = \mathfrak{R}_+^p \times \mathfrak{R}^q$. The saddle point problem of (DCOP) can be formulated as

$$\max_{\lambda \in \mathcal{K}^\circ} \min_{x \in \mathcal{X}} \mathcal{L}(x, \lambda) = \sum_{i=1}^m \mathcal{L}_i(x, \lambda), \tag{2.1}$$

where $\mathcal{L}_i(x, \lambda) = f_i(x) + \lambda^\top h_i(x)$, λ^I and λ^E are the multipliers respectively corresponding to the inequality and equality constraints, and $\lambda = (\lambda^I, \lambda^E) \in \mathcal{K}^\circ$.

Under Slater’s condition, there does not exist a duality gap between the primal problem (DCOP) and its dual. We then attempt to develop a distributed algorithm to solve the max-min problem (2.1) and find an optimal solution to (DCOP).

For any $\bar{\lambda} \in \mathcal{K}^\circ$, denote

$$q(\lambda) = \inf_{x \in \mathcal{X}} \mathcal{L}(x, \lambda),$$

$$Q(\bar{\lambda}) = \left\{ \lambda = (\lambda^I, \lambda^E) \in \mathcal{K}^\circ : q(\lambda) \geq q(\bar{\lambda}) \right\}.$$

The boundness of multiplier λ^I corresponding to the inequality constraints is indispensable in the design and analysis of our algorithm, which is guaranteed by the following lemma.

Lemma 2.1 ([28], Lemma1) For any $\bar{\lambda} \in \mathcal{K}^\circ$ and $\alpha \in \mathfrak{R}$, it holds that

$$\max_{\lambda \in Q(\bar{\lambda})} \|\lambda^I\| \leq \frac{1}{\gamma(\bar{x})} (f(\bar{x}) - q(\bar{\lambda})), \tag{2.2}$$

where $\gamma(\bar{x}) = \min_{1 \leq j \leq p} \{-\sum_{i=1}^m h_i^j(\bar{x})\}$ and \bar{x} satisfies Assumption 2.4.

Let q^* denote the dual-optimal value and Q^* denote the dual optimal solutions set, namely

$$Q^* = \left\{ \lambda = (\lambda^I, \lambda^E) \in \mathcal{K}^\circ : q(\lambda) \geq q^* \right\} \subseteq Q(\bar{\lambda}).$$

According to Lemma 2.1, the optimal multiplier λ^I is bounded. Select an arbitrary vector $\bar{\lambda} \in \mathcal{K}^\circ$, define

$$Q_I(\alpha) = \left\{ \lambda^I \in \mathfrak{R}_+^p : \|\lambda^I\| \leq \alpha \right\}, \quad \forall \alpha \geq \frac{1}{\gamma(\bar{x})} (f(\bar{x}) - q(\bar{\lambda})),$$

$$Q = Q_I(\alpha) \times \mathfrak{R}^q.$$

It follows that $Q^* \subseteq Q(\bar{\lambda}) \subseteq Q$. The set Q will be used in designing a distributed algorithm in Sect. 3. Note that Q is the Cartesian product of a ball in \mathfrak{R}^p and \mathfrak{R}^q , and the projection onto Q has a simple closed-form solution.

3 Design of Distributed Algorithm

3.1 Primal-Dual Projected Gradient Method

Consider the following centralized optimization problem with equality and inequality constraints,

$$\min_{x \in \mathcal{X}} \tilde{f}(x) \quad \text{s.t.} \quad \tilde{h}(x) \in \mathcal{K}, \tag{3.1}$$

where $\mathcal{K} = \mathfrak{R}_-^p \times \{0_q\}$. The saddle point problem associated with (3.1) is

$$\max_{\lambda \in \mathcal{K}^\circ} \min_{x \in \mathcal{X}} \tilde{L}(x, \lambda), \tag{3.2}$$

where $\tilde{L}(x, \lambda) = \tilde{f}(x) + \lambda^\top \tilde{h}(x)$. Given a pair (x^k, λ^k) , the primal-dual projected gradient method [7] for solving (3.2) obeys the following rules to generate the successive primal-dual pair (x^{k+1}, λ^{k+1}) ,

$$x^{k+1} = P_{\mathcal{X}} \left(x^k - \alpha_k \nabla_x \tilde{L}(x^k, \lambda^k) \right), \tag{3.3a}$$

$$\lambda^{k+1} = P_{\mathcal{K}^\circ} \left(\lambda^k + \alpha_k \nabla_\lambda \tilde{L}(x^k, \lambda^k) \right), \tag{3.3b}$$

where $\alpha_k > 0$ is a given step-size. Review the first-order optimality conditions [37, Theorem 36.6] for (3.1), it holds that $x \in \mathcal{X}$ is an optimal solution to (3.1) if and only if there is a primal-dual pair (x, λ) such that

$$x = P_{\mathcal{X}} \left(x - \alpha \nabla_x \tilde{L}(x, \lambda) \right),$$

$$\lambda = P_{\mathcal{K}^\circ} \left(\lambda + \alpha \nabla_\lambda \tilde{L}(x, \lambda) \right),$$

for any scalar $\alpha > 0$. Therefore, the primal-dual projected gradient method also can be viewed as the fixed point method.

3.2 TV- \mathcal{AB} Algorithm

Over directed time-varying multi-agent networks consisting of m agents, consider an unconstrained distributed optimization problem:

$$\min_{x \in \mathbb{R}^n} f(x) = \sum_{i=1}^m f_i(x),$$

where f_i is the local objective function of agent i . The authors of [39] proposed a distributed algorithm referred to as the time-varying \mathcal{AB} (TV- \mathcal{AB}) gradient method for this problem. In specific, the updating rules of variables are

$$\begin{aligned} x_i^{k+1} &= \sum_{j=1}^m A(k)_{ij} x_j^k - \alpha y_i^k, \\ y_i^{k+1} &= \sum_{j=1}^m B(k)_{ij} y_j^k + \nabla f_i(x_i^{k+1}) - \nabla f_i(x_i^k), \end{aligned}$$

where $A(k)$ and $B(k)$ are row-stochastic and column-stochastic matrices, respectively. The auxiliary variables $\{y_i^k\}_{i=1}^m$ are used to track the average of $\{\nabla f_i(x_i^k)\}_{i=1}^m$, precisely, it holds that $\frac{1}{m} \sum_{i=1}^m y_i^k = \frac{1}{m} \sum_{i=1}^m \nabla f_i(x_i^k)$ for all $k \geq 0$. Each agent i then performs an inexact gradient descent step along the direction $-y_i^k$. Obeying such distributed strategies, all agents adjust their decision vectors $\{x_i^k\}_{i=1}^m$ to be consensual (i.e., $\|x_i^k - x_j^k\| \rightarrow 0$ as $k \rightarrow \infty$) and optimal (i.e., $\|x_i^k - x^*\| \rightarrow 0$ as $k \rightarrow \infty$ for some optimal solution x^*).

In a multi-agent system, if agents adopts the primal-dual gradient method (3.3) to solve the saddle point problem (2.1) associated with (DCOP), they must access to the global information $\sum_{i=1}^m \nabla \mathcal{L}_i$. However, in distributed settings, this information is not available to any agent. To avoid such an obstacle, we strategically integrate the ideas behind TV- \mathcal{AB} and the primal-dual gradient method to propose the following primal-dual distributed algorithm for solving (DCOP),

$$x_i^{k+1} = P_{\mathcal{X}} \left(\sum_{j=1}^m a_{ij}(k) x_j^k - \alpha_k z_i^k \right), \quad (3.4a)$$

$$\lambda_i^{k+1} = P_{\mathcal{Q}} \left(\sum_{j=1}^m a_{ij}(k) \lambda_j^k + \alpha_k y_i^k \right), \quad (3.4b)$$

$$z_i^{k+1} = \sum_{j=1}^m b_{ij}(k) z_j^k + \nabla_x \mathcal{L}_i(x_i^{k+1}, \lambda_i^{k+1}) - \nabla_x \mathcal{L}_i(x_i^k, \lambda_i^k), \quad (3.4c)$$

$$y_i^{k+1} = \sum_{j=1}^m b_{ij}(k)y_j^k + h_i(x_i^{k+1}) - h_i(x_i^k), \tag{3.4d}$$

where $z_i^0 = \nabla_x \mathcal{L}_i(x_i^0, \lambda_i^0)$ and $y_i^0 = h_i(x_i^0)$, for each $i = 1, 2, \dots, m$.

In the presented algorithm (3.3), each agent i generates a primal-dual pair (x_i^k, λ_i^k) to estimate the saddle point (x^*, λ^*) of the max-min problem (2.1), and employs the sequences $\{z_i^k\}_{k \geq 0}$ and $\{y_i^k\}_{k \geq 0}$ to track the average gradients $\frac{1}{m} \sum_{i=1}^m \nabla_x \mathcal{L}_i(x_i^k, \lambda_i^k)$ and $\frac{1}{m} \sum_{i=1}^m \nabla_\lambda \mathcal{L}_i(x_i^k, \lambda_i^k)$, respectively. After one round of communication, each agent i mixes the information received from its in-neighbors, then uses the mixed information and local data to output new estimate vector $(x_i^{k+1}, \lambda_i^{k+1})$. The local computation of each agent is independent and can be performed by agents in parallel.

Remark 3.1 For the updating formulas of the multipliers λ_i^k , we adopt the projection operator P_Q instead of $P_{\mathcal{X}^\circ}$. The reason is that we consider the saddle point problem

$$\max_{\lambda \in Q} \min_{x \in \mathcal{X}} \sum_{i=1}^m \mathcal{L}_i(x, \lambda) \tag{3.5}$$

instead of (2.1). One can get that the problems (2.1) and (3.5) have the same saddle points, e.g., see [53, Lemma 3.1]. The boundedness of the sequence $\{(\lambda_i^k)^I\}$ is guaranteed by the bounded set $Q_I(\alpha)$, where $\lambda_i^k = ((\lambda_i^k)^I, (\lambda_i^k)^E)$.

Algorithm 1 Distributed Primal-Dual Algorithm.

Input: Let n be the number of iterations of termination. Initialization: $(x_i^0, \lambda_i^0) \in \mathcal{X} \times \mathcal{K}^\circ$, $y_i^0 = h_i(x_i^0)$, and $z_i^0 = \nabla f_i(x_i^0) + \nabla h_i(x_i^0)\lambda_i^0$, for each $i = 1, 2, \dots, m$. Select the sequence $\{\alpha_k\}$ of step-sizes and a sufficiently large scalar α such that $Q^* \subset Q$. Set $k = 0$.

Output: $x_i^n, \lambda_i^n, i = 1, 2, \dots, m$.

- 1: **while** $k \leq n$ **do**
 - 2: **for** $i = 1, 2, \dots, m$ (in parallel) **do**
 - 3: $x_i^{k+1} = P_{\mathcal{X}} \left(\sum_{j=1}^m a_{ij}(k)x_j^k - \alpha_k z_i^k \right)$,
 - 4: $\lambda_i^{k+1} = P_Q \left(\sum_{j=1}^m a_{ij}(k)\lambda_j^k + \alpha_k y_i^k \right)$,
 - 5: $y_i^{k+1} = \sum_{j=1}^m b_{ij}(k)y_j^k + h_i(x_i^{k+1}) - h_i(x_i^k)$,
 - 6: $d_i^k = \nabla f_i(x_i^k) + \nabla h_i(x_i^k)\lambda_i^k$,
 - 7: $d_i^{k+1} = \nabla f_i(x_i^{k+1}) + \nabla h_i(x_i^{k+1})\lambda_i^{k+1}$,
 - 8: $z_i^{k+1} = \sum_{j=1}^m b_{ij}(k)z_j^k + d_i^{k+1} - d_i^k$,
 - 9: **end for**
 - 10: $k \leftarrow k + 1$,
 - 11: **end while**
-

In what follows, we assume that the step-size sequence $\{\alpha_k\}$ is squared summable, which is also considered by references [2, 10, 32], etc. The step-size conditions are easy to satisfy, e.g., choose $\alpha_k = c/k^\beta$ with $\beta \in (1/2, 1]$ and constant $c > 0$.

Assumption 3.1 Suppose that the sequence $\{\alpha_k\}_{k=0}^\infty$ of step-sizes satisfies that

$$\alpha_k \geq 0, \quad \sum_{k=0}^\infty \alpha_k = \infty, \quad \sum_{k=0}^\infty \alpha_k^2 < \infty.$$

Under Assumption 2.3, the subset \mathcal{X} is compact. According to the continuity and differentiability of f_i and h_i , the gradients of f_i and h_i are Lipschitz continuous over \mathcal{X} . Without loss of generality, for any $x, y \in \mathcal{X}$ and $i \in \mathcal{V}$, assume that there exists a constant $L > 0$ such that

$$\begin{aligned} \|f_i(x)\| \leq L, \quad \|h_i(x)\| \leq L, \quad \|h_i(x) - h_i(y)\| \leq L\|x - y\|, \\ \|\nabla_x \mathcal{L}_i(x, \lambda) - \nabla_x \mathcal{L}_i(y, \lambda)\| \leq L\|x - y\|. \end{aligned} \tag{3.6}$$

4 Convergence Analysis

In this section, we analyze the proposed Algorithm 1 and show the obtained theoretical results about the convergence and rate. To ease the analysis, we place some supporting lemmas in “Appendix C” and define the following notations,

$$\begin{aligned} \omega_i^k &= \left(x_i^k, \lambda_i^k\right), \quad \eta_i^k = \left(z_i^k, -y_i^k\right), \\ \nabla \mathcal{L}_i(\omega_i^k) &= \left(\nabla_x \mathcal{L}_i\left(x_i^k, \lambda_i^k\right), -h_i\left(x_i^k\right)\right), \\ \omega^k &= \left(\omega_1^k, \omega_2^k, \dots, \omega_m^k\right)^\top, \quad \eta^k = \left(\eta_1^k, \eta_2^k, \dots, \eta_m^k\right)^\top, \\ \nabla L(\omega^k) &= \left(\nabla \mathcal{L}_1\left(\omega_1^k\right), \dots, \nabla \mathcal{L}_m\left(\omega_m^k\right)\right)^\top, \\ \mathcal{A}_k &= \begin{pmatrix} A(k) & 0_{m \times m} \\ 0_{m \times m} & A(k) \end{pmatrix}, \quad \mathcal{B}_k = \begin{pmatrix} B(k) & 0_{m \times m} \\ 0_{m \times m} & B(k) \end{pmatrix}, \\ \Omega &= \underbrace{\mathcal{X} \times \mathcal{Q} \times \mathcal{X} \times \mathcal{Q} \times \dots \times \mathcal{X} \times \mathcal{Q}}_m. \end{aligned}$$

We then rewrite the iterative schemes of Algorithm 1 into the following compact form,

$$\omega^{k+1} = P_\Omega \left(\mathcal{A}_k \omega^k - \alpha_k \eta^k\right), \tag{4.1}$$

$$\eta^{k+1} = \mathcal{B}_k \eta^k + \nabla L(\omega^{k+1}) - \nabla L(\omega^k), \tag{4.2}$$

where the matrices $\{\mathcal{A}_k\}$ and $\{\mathcal{B}_k\}$ are also row-stochastic and column-stochastic, respectively. Furthermore, we make a state transformation: $s^k = V_k^{-1} \eta^k$, where $V_k =$

$\text{diag}(v_k)$ and v_k are given by

$$v_{k+1} = \mathcal{B}_k v_k, \quad v_0 = 1_{2m}. \tag{4.3}$$

Consequently, (4.1) and (4.2) can be equivalently rewritten as

$$\omega^{k+1} = P_\Omega \left(\mathcal{A}_k \omega^k - \alpha_k V_k s^k \right), \tag{4.4}$$

$$s^{k+1} = \mathcal{R}_k s^k + V_{k+1}^{-1} \left(\nabla L(\omega^{k+1}) - \nabla L(\omega^k) \right), \tag{4.5}$$

where $\mathcal{R}_k = V_{k+1}^{-1} \mathcal{B}_k V_k$. It can be verified that $\{\mathcal{R}_k\}$ is a sequence of row-stochastic matrices such that $\{\frac{1}{2m} v_k\}$ is an absolute probability sequence. For any $k \geq s \geq 0$, define

$$\begin{aligned} \Phi_{\mathcal{A}}(k, s) &= \mathcal{A}_k \times \mathcal{A}_{k-1} \times \dots \times \mathcal{A}_s, \\ \Phi_{\mathcal{R}}(k, s) &= \mathcal{R}_k \times \mathcal{R}_{k-1} \times \dots \times \mathcal{R}_s. \end{aligned}$$

As shown in Lemma C.1, for any index $s \geq 0$, the row-stochastic matrix sequence $\{\Phi_{\mathcal{A}}(k, s)\}$ linearly converges to a matrix $\Phi_{\mathcal{A}}(s)$ whose columns are stochastic vectors, as $k \rightarrow \infty$. The asymptotic properties of the matrix sequences $\{\Phi_{\mathcal{A}}(k, s)\}$ and $\{\Phi_{\mathcal{R}}(k, s)\}$ are used to estimate consensus errors of the the sequence $\{\omega^k\}$.

From (4.2), it holds that

$$\begin{aligned} 1_{2m}^\top \eta^{k+1} &= 1_{2m}^\top \mathcal{B}_k \eta_k + 1_{2m}^\top \left(\nabla L(\omega^{k+1}) - \nabla L(\omega^k) \right) \\ &= 1_{2m}^\top \eta^k + 1_{2m}^\top \left(\nabla L(\omega^{k+1}) - \nabla L(\omega^k) \right), \end{aligned}$$

which is equivalent to

$$1_{2m}^\top \eta^{k+1} - 1_{2m}^\top \nabla L(\omega^{k+1}) = 1_{2m}^\top \eta^k - 1_{2m}^\top \nabla L(\omega^k), \quad \forall k \geq 0.$$

Noticing that the initialization $\eta^0 = \nabla L(\omega^0)$, the following tracking equations hold,

$$1_{2m}^\top \eta^k = 1_{2m}^\top \nabla L(\omega^k), \quad \forall k \geq 0.$$

Define

$$\bar{\omega}^k = \left(\bar{x}^k, \bar{\lambda}^k \right) = \phi_k^\top \omega^k, \quad \bar{s}^k = \frac{1}{2m} v_k^\top s^k, \tag{4.6}$$

where the sequence $\{\phi_k\}$ is given by Lemma C.2. Therefore,

$$\bar{s}^k = \frac{1}{2m} v_k^\top V_k^{-1} \eta^k = \frac{1}{2m} 1_{2m}^\top \eta^k = \frac{1}{2m} 1_{2m}^\top \nabla L(\omega^k),$$

which means that \bar{s}^k tracks the average gradient of the Lagrange function $\mathcal{L}(x, \lambda)$. Since the subset \mathcal{X} is compact and convex, $\nabla_x \mathcal{L}(x, \lambda)$ and $\nabla_\lambda \mathcal{L}(x, \lambda)$ are bounded

over \mathcal{X} , which implies the sequence $\{\bar{s}^k\}$ is bounded. Furthermore, the following lemma claims that the sequence $\{s^k\}$ is also bounded.

Lemma 4.1 *There is a constant $M > 0$ such that*

$$\|s^k\| \leq M, \quad \forall k \geq 0.$$

Proof See “Appendix A”. □

According to Algorithm 1, each agent i generates the local estimate $\omega_i^k = (x_i^k, \lambda_i^k)$ to the saddle point $\omega^* = (x^*, \lambda^*)$. It’s necessary to evaluate the error between any two vectors ω_i^k and ω_j^k , or equivalently, the consensus error $\|\omega^k - 1_{2m}\bar{\omega}^k\|$, which is the constraint violation of $x_i = x_j$ and $\lambda_i = \lambda_j$ for all $(i, j) \in \mathcal{E}_k, k \geq 0$. The following lemma characterizes the consensus errors of the sequences $\{\omega^k\}$ and $\{s^k\}$.

Lemma 4.2 *Under Assumptions 2.1–2.4 and 3.1, for the sequences generated by Algorithm 1, it meets that*

$$\begin{aligned} \lim_{k \rightarrow \infty} \|\omega^k - 1_{2m}\bar{\omega}^k\| &= 0, & \sum_{k=0}^{\infty} \alpha_k \|\omega^k - 1_{2m}\bar{\omega}^k\| &< \infty, \\ \lim_{k \rightarrow \infty} \|s^k - 1_{2m}\bar{s}^k\| &= 0, & \sum_{k=0}^{\infty} \alpha_k \|s^k - 1_{2m}\bar{s}^k\| &< \infty. \end{aligned}$$

Proof See “Appendix B”. □

The rest of the convergence analysis is to evaluate the optimality error $\|\bar{\omega}^k - \omega^*\|$, which requires strategically applying the results of Lemmas 4.1 and 4.2, and the supporting lemmas in “Appendix C”.

Theorem 4.1 *Under Assumptions 2.1–2.4 and 3.1, for any $i \in \mathcal{V}$, the sequence $\{(x_i^k, \lambda_i^k)\}$ generated by Algorithm 1 converges to some saddle point (x^*, λ^*) of the Lagrange function $\mathcal{L}(x, \lambda)$, i.e.,*

$$\lim_{k \rightarrow \infty} \left\| \left(x_i^k, \lambda_i^k \right) - (x^*, \lambda^*) \right\| = 0, \quad \forall i = 1, 2, \dots, m. \tag{4.7}$$

Proof For any $\omega^* = (x^*, \lambda^*) \in \mathcal{X} \times \mathcal{K}^\circ$, define

$$\begin{aligned} I_1^k &= 2\alpha_k \left\langle 1_{2m}(\nabla L(\omega^k) - \nabla L(1_{2m}\bar{\omega}^k)), \bar{\omega}^k - \omega^* \right\rangle, \\ I_2^k &= 2\alpha_k \left\langle V_k s^k - 1_{2m}\bar{s}^k, \mathcal{A}_k \omega^k - 1_{2m}\omega^* \right\rangle, \\ I_3^k &= 2\alpha_k \left\langle 1_{2m}(\bar{\omega}^k - \omega^*), \mathcal{A}_k \omega^k - 1_{2m}\bar{\omega}^k \right\rangle. \end{aligned}$$

We then split the proof into the following three steps.

Step 1: Verify the following inequality,

$$\begin{aligned} & \|\omega^{k+1} - 1_{2m}\omega^*\|^2 \\ & \leq \|\omega^k - 1_{2m}\omega^*\|^2 + I_1^k + I_2^k + I_3^k + 4m^2M^2\alpha_k^2 \\ & \quad - 2\alpha_k \left(\mathcal{L}(x^*, \lambda^*) - \mathcal{L}(x^*, \bar{\lambda}^k) \right) - 2\alpha_k \left(\mathcal{L}(\bar{x}^k, \lambda^*) - \mathcal{L}(x^*, \lambda^*) \right). \end{aligned} \tag{4.8}$$

From (4.4) and noticing the non-expansive projection operator P_Ω , one has

$$\begin{aligned} \|\omega^{k+1} - 1_{2m}\omega^*\|^2 & \leq \|\mathcal{A}_k\omega^k - \alpha_k V_k s^k - 1_{2m}\omega^*\|^2 \\ & = \|\mathcal{A}_k\omega^k - 1_{2m}\omega^*\|^2 + \alpha_k^2 \|V_k s^k\|^2 - 2\alpha_k \left\langle V_k s^k, \mathcal{A}_k\omega^k - 1_{2m}\omega^* \right\rangle. \end{aligned}$$

Since $\{\frac{1}{2m} v_k\}$ is a stochastic vector sequence and $V_k = \text{diag}(v_k)$, it follows that

$$\|V_k s^k\| \leq \|V_k\| \|s^k\| \leq 2m \|s^k\| \leq 2mM,$$

where the last inequality derives from the boundedness of $\{s^k\}$ (Lemma 4.1). Given that the matrix \mathcal{A}_k is row-stochastic, it holds that

$$\|\mathcal{A}_k\omega^k - 1_{2m}\omega^*\|^2 \leq \|\omega^k - 1_{2m}\omega^*\|^2.$$

As to the term $2\alpha_k \langle V_k s^k, \mathcal{A}_k\omega^k - 1_{2m}\omega^* \rangle$, one has

$$\begin{aligned} & 2\alpha_k \left\langle V_k s^k, \mathcal{A}_k\omega^k - 1_{2m}\omega^* \right\rangle \\ & = 2\alpha_k \left\langle 1_{2m}\bar{s}^k, 1_{2m}\bar{\omega}^k - 1_{2m}\omega^* \right\rangle + 2\alpha_k \left\langle V_k s^k - 1_{2m}\bar{s}^k, \mathcal{A}_k\omega^k - 1_{2m}\omega^* \right\rangle \\ & \quad + \left\langle 1_{2m}\bar{\omega}^k - \omega^*, \mathcal{A}_k\omega^k - 1_{2m}\bar{\omega}^k \right\rangle \\ & = \left\langle 1_{2m}^\top \nabla L(1_{2m}\bar{\omega}^k), \bar{\omega}^k - \omega^* \right\rangle + I_1^k + I_2^k + I_3^k. \end{aligned}$$

Note that the Lagrange function $\mathcal{L}(x, \lambda)$ is convex-concave with respect to x and λ , applying the subdifferential inequality of convex function yields

$$\left\langle 1_{2m}^\top \nabla L(1_{2m}\bar{\omega}^k), \bar{\omega}^k - \omega^* \right\rangle \geq \left(\mathcal{L}(\bar{x}^k, \lambda^*) - \mathcal{L}(x^*, \bar{\lambda}^k) \right).$$

Combining the above relations to yield (4.8).

Step 2: Verify that

$$\sum_{k=0}^\infty |I_1^k| < \infty, \quad \sum_{k=0}^\infty |I_2^k| < \infty, \quad \sum_{k=0}^\infty |I_3^k| < \infty.$$

According to the Cauchy inequality and the Lipschitz continuity of $\nabla\mathcal{L}$, there exist positive constants L_1, L_2 and L_3 such that

$$\begin{aligned} |I_1^k| &\leq 2L_1\alpha_k\|\omega^k - 1_{2m}\bar{\omega}^k\|, \\ |I_2^k| &\leq 2L_2\alpha_k\|s^k - 1_{2m}\bar{s}^k\|, \\ |I_3^k| &\leq 2L_3\alpha_k\|\omega^k - 1_{2m}\bar{\omega}^k\|. \end{aligned}$$

By Lemma 4.2, the three scalar sequences $\{|I_1^k|\}$, $\{|I_2^k|\}$, and $\{|I_3^k|\}$ are summable.

Step 3: Prove the convergence. In the inequality (4.8), choose $\omega^* = (x^*, \lambda^*)$ as an arbitrary saddle point of the Lagrange function $\mathcal{L}(x, \lambda)$. The definition of saddle points indicates that

$$\begin{aligned} \mathcal{L}(\bar{x}^k, \lambda^*) - \mathcal{L}(x^*, \lambda^*) &\geq 0, \\ \mathcal{L}(x^*, \lambda^*) - \mathcal{L}(x^*, \bar{\lambda}^k) &\geq 0. \end{aligned}$$

Consequently, it follows from Lemma C.3 that the limit $\lim_{k \rightarrow \infty} \|\omega^k - 1_{2m}\omega^*\|$ exists and

$$\begin{aligned} \sum_{k=0}^{\infty} \alpha_k \left(\mathcal{L}(\bar{x}^k, \lambda^*) - \mathcal{L}(x^*, \lambda^*) \right) &< \infty, \\ \sum_{k=0}^{\infty} \alpha_k \left(\mathcal{L}(x^*, \lambda^*) - \mathcal{L}(x^*, \bar{\lambda}^k) \right) &< \infty. \end{aligned}$$

In view of $\sum_{k=0}^{\infty} \alpha_k = \infty$, it must hold that

$$\begin{aligned} \liminf_{k \rightarrow \infty} \mathcal{L}(\bar{x}^k, \lambda^*) &= \mathcal{L}(x^*, \lambda^*), \\ \limsup_{k \rightarrow \infty} \mathcal{L}(x^*, \bar{\lambda}^k) &= \mathcal{L}(x^*, \lambda^*). \end{aligned}$$

Furthermore, the Lagrange function $\mathcal{L}(x, \lambda)$ is continuous differentiable and convex-concave, hence,

$$\lim_{k \rightarrow \infty} \|\bar{x}^k - x^*\| = 0, \quad \lim_{k \rightarrow \infty} \|\bar{\lambda}^k - \lambda^*\| = 0,$$

or equivalently, $\lim_{k \rightarrow \infty} \|\bar{\omega}^k - \omega^*\| = 0$. From Lemma 4.2, it's clear that

$$\lim_{k \rightarrow \infty} \|\omega^k - 1_{2m}\omega^*\| = 0.$$

The proof is completed. □

The convergence results of Theorem 4.1 show that for each agent $i \in \{1, 2, \dots, m\}$, its decision variable sequence $\{x_i^k\}_{k=0}^{\infty}$ converges to an optimal solution x^* of (DCOP), and the Lagrange multiplier estimate sequence $\{\lambda_i^k\}_{k=0}^{\infty}$ converges to an dual-optimal

solution λ^* . In other words, by communicating with neighbors and computing locally, all agents cooperatively solve (DCOP).

To present results about the convergence rate of Algorithm 1, we define the following weighted sequences:

$$\tilde{x}_s^n = \frac{\sum_{k=s}^n \alpha_k \bar{x}^k}{\sum_{k=s}^n \alpha_k}, \quad \tilde{\lambda}_s^n = \frac{\sum_{k=s}^n \alpha_k \bar{\lambda}^k}{\sum_{k=s}^n \alpha_k}, \quad \forall 0 \leq s \leq n. \tag{4.9}$$

where \bar{x}^k and $\bar{\lambda}^k$ are given by (4.6). The following theorem characterizes a sublinear convergence rate of Algorithm 1 in terms of the primal-dual gap.

Theorem 4.2 *Let (x^*, λ^*) be an arbitrary saddle point of $\mathcal{L}(x, \lambda)$. Under Assumptions 2.1–2.4 and 3.1, there exists a constant $M_1 > 0$ such that*

$$0 \leq \mathcal{L}(\tilde{x}_s^n, \lambda^*) - \mathcal{L}(x^*, \tilde{\lambda}_s^n) \leq \frac{M_1}{\sum_{k=s}^n \alpha_k}, \quad \forall 0 \leq s \leq n. \tag{4.10}$$

Furthermore, if one chooses the step-size sequence as $\alpha_k = \frac{c}{k^{1/2+\beta}}$ with $\beta \in (0, 1/2)$ and constant $c > 0$, the primal-dual gap satisfies that

$$0 \leq \mathcal{L}(\tilde{x}_{\lfloor n/2 \rfloor}^n, \lambda^*) - \mathcal{L}(x^*, \tilde{\lambda}_{\lfloor n/2 \rfloor}^n) \leq \frac{2M_1}{c(n+1)^{1/2-\beta}}, \tag{4.11}$$

where $\lfloor n \rfloor$ is the integer part of a positive real number n .

Proof By the definition of saddle points, it’s clear that

$$\mathcal{L}(\tilde{x}_s^n, \lambda^*) - \mathcal{L}(x^*, \tilde{\lambda}_s^n) = \mathcal{L}(\tilde{x}_s^n, \lambda^*) - \mathcal{L}(x^*, \lambda^*) + \mathcal{L}(x^*, \lambda^*) - \mathcal{L}(x^*, \tilde{\lambda}_s^n) \geq 0.$$

For any fixed $x \in \mathfrak{N}^n$ and $\lambda \in \mathfrak{N}^{p+q}$, in view of the convexity of the functions $\mathcal{L}(\cdot, \lambda)$ and $-\mathcal{L}(x, \cdot)$, one has

$$\begin{aligned} & \mathcal{L}(\tilde{x}_s^n, \lambda^*) - \mathcal{L}(x^*, \tilde{\lambda}_s^n) \\ & \leq \left(\sum_{k=s}^n \alpha_k \right)^{-1} \sum_{k=s}^n \alpha_k \left[\mathcal{L}(\bar{x}^k, \lambda^*) - \mathcal{L}(x^*, \bar{\lambda}^k) \right] \\ & = \left(\sum_{k=s}^n \alpha_k \right)^{-1} \sum_{k=s}^n \alpha_k \left[\mathcal{L}(\bar{x}^k, \lambda^*) - \mathcal{L}(x^*, \lambda^*) + \mathcal{L}(x^*, \lambda^*) - \mathcal{L}(x^*, \bar{\lambda}^k) \right]. \end{aligned}$$

From the proof of Theorem 4.1, it holds that

$$0 < \sum_{k=0}^{\infty} \alpha_k \left(\mathcal{L}(\bar{x}^k, \lambda^*) - \mathcal{L}(x^*, \lambda^*) \right) = M' < \infty,$$

$$0 < \sum_{k=0}^{\infty} \alpha_k \left(\mathcal{L}(x^*, \lambda^*) - \mathcal{L}(x^*, \bar{\lambda}^k) \right) = M'' < \infty.$$

Let $M_1 := M' + M''$, it's clear that (4.10) holds. Take $\alpha_k = \frac{c}{k^{1/2+\beta}}$, one can get the following estimate,

$$\sum_{k=\lfloor n/2 \rfloor}^n \frac{1}{k^{1/2+\beta}} \geq \int_{\lfloor n/2 \rfloor}^{n+1} \frac{dt}{t^{1/2+\beta}} = \frac{2}{1-2\beta} \left((n+1)^{\frac{1}{2}-\beta} - \lfloor n/2 \rfloor^{\frac{1}{2}-\beta} \right).$$

Let $\phi(x) = x^{1/2-\beta}$ with $\beta \in (0, 1/2)$, it's easy to verify that $\phi(x)$ is a concave function, hence,

$$\begin{aligned} \phi(n+1) - \phi(\lfloor n/2 \rfloor) &\geq \nabla \phi(n+1)(n+1 - \lfloor n/2 \rfloor) \\ &\geq \left(\frac{1}{2} - \beta \right) \frac{(n+1) - (n+1)/2}{(n+1)^{1/2+\beta}} \\ &= \left(\frac{1}{4} - \frac{\beta}{2} \right) (n+1)^{\frac{1}{2}-\beta}. \end{aligned}$$

Choose $s = \lfloor n/2 \rfloor$ in (4.10), it follows that

$$\mathcal{L}(\tilde{x}_{\lfloor n/2 \rfloor}^n, \lambda^*) - \mathcal{L}(x^*, \tilde{\lambda}_{\lfloor n/2 \rfloor}^n) \leq \frac{M_1}{\sum_{k=\lfloor n/2 \rfloor}^n \frac{c}{k^{1/2+\beta}}} \leq \frac{2M_1}{c(n+1)^{1/2-\beta}}.$$

The proof is completed. \square

Theorem 4.2 shows that the primal-dual gap $\mathcal{L}(\tilde{x}_{\lfloor n/2 \rfloor}^n, \lambda^*) - \mathcal{L}(x^*, \tilde{\lambda}_{\lfloor n/2 \rfloor}^n)$ decays to zeros at a rate of $O(\frac{1}{n^{1/2-\beta}})$. If one chooses the value of β to be small enough, the rate is near $O(1/\sqrt{n})$, where n represents the number of iterations.

5 Numerical Simulations

In this section, we test the proposed algorithm using two examples. The first one is motivated by applications in wireless networks and is used to justify the theoretical results in the paper. Another one is an empirical risk optimization problem to be solved in the Neyman–Pearson (NP) classification [44], which is used to compare the performance of Algorithm 1 with the distributed proximal primal-dual algorithm [17].

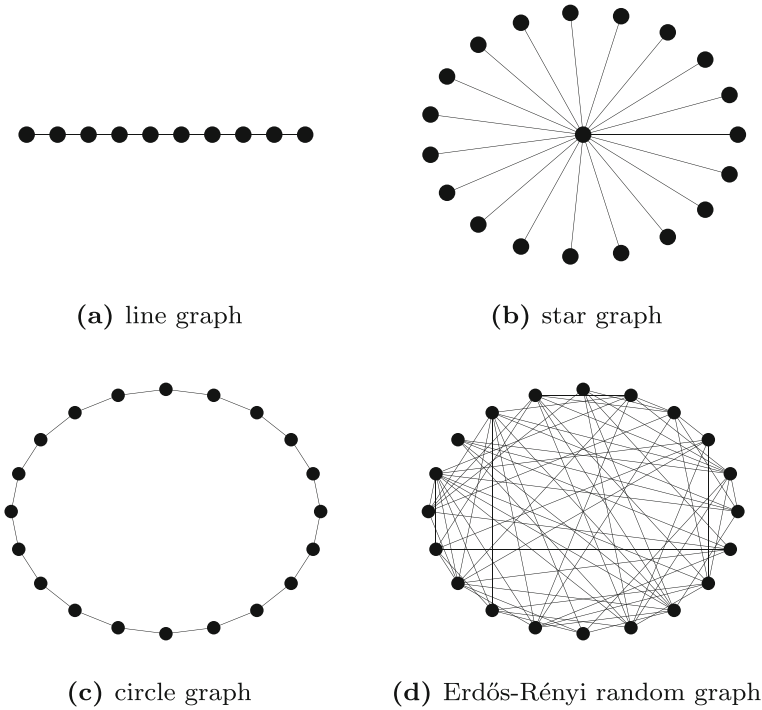


Fig. 1 Network topology of 100 agents

5.1 Example in Wireless Networks

Consider the following distributed optimization problem with nonlinear constraints,

$$\begin{aligned}
 & \min_{x \in [0,1]} \sum_{i=1}^{100} c_i x \\
 & \text{s.t.} \quad \sum_{i=1}^{100} -d_i \log(1 + x) \leq b,
 \end{aligned}
 \tag{5.1}$$

where the constraints with this form arise, for instance, in wireless networks to ensure quality of service. This problem is also considered by [17, 25]. Set $b = 5$, $c_i = i/m$, $d_i = i/(m + 1)$ for each $i = 1, 2, \dots, 100$, where the network size (number of agents) $m = 100$. The optimal value f^* is computed by the convex optimization toolbox [11] (CVX) under the best precision. We then use f^* as a benchmark for our algorithm.

As shown in Fig. 1, the network topology of 100 agents is selected in turn among the line, star, circle, and Erdős-Rényi random graphs with period 4. The corresponding

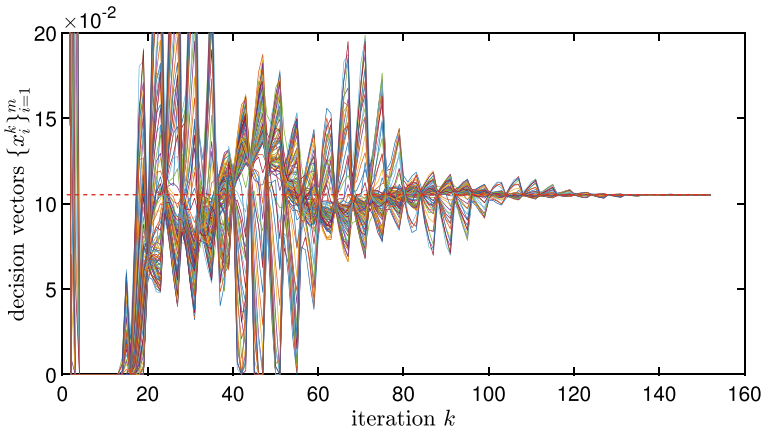


Fig. 2 Evolution of agents’ decision vectors $x_i^k, i = 1, 2, \dots, 100$, where the red dashed line represents the value of the optimal solution x^*

adjacency matrices $A(k)$ and $B(k)$ are constructed by

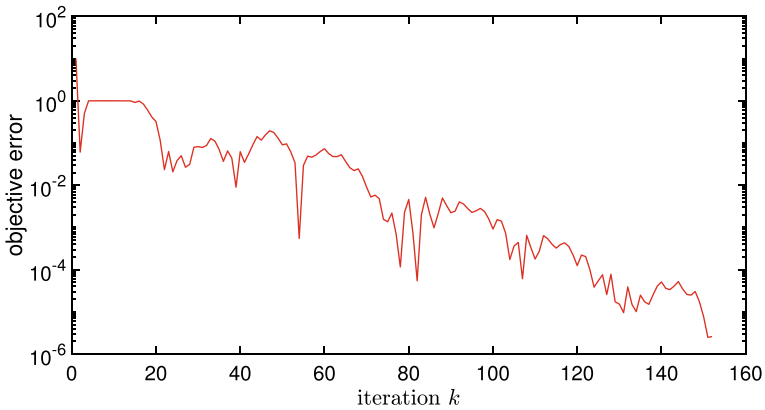
$$a_{ij}(k) = \begin{cases} \frac{1}{|\mathcal{N}_i^{\text{in}}(k)|}, & \text{if } (i, j) \in \mathcal{E}(k), \\ 0, & \text{otherwise,} \end{cases}$$

$$b_{ij}(k) = \begin{cases} \frac{1}{|\mathcal{N}_j^{\text{out}}(k)|}, & \text{if } (i, j) \in \mathcal{E}(k), \\ 0, & \text{otherwise,} \end{cases}$$

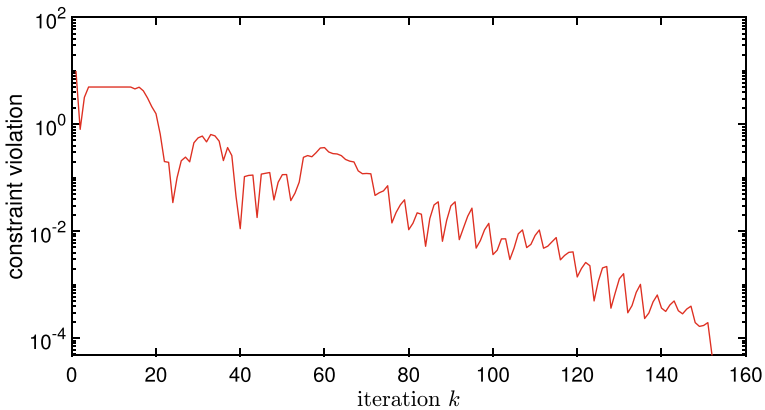
where $|\mathcal{N}_i^{\text{in}}(k)|$ and $|\mathcal{N}_j^{\text{out}}(k)|$ are the in-degree of agent i and the out-degree of agent j at time k , respectively.

We execute the proposed Algorithm 1 with step-size $\alpha_k = 1/k^{0.51}$ to solve (5.1) on a laptop computer. The procedure stops if the maximum of objective error and constraint violation decays to smaller than the given tolerance error 10^{-4} , where the objective error is $|f(\bar{x}^k) - f^*|/|f^*|$, the constraint violation is the sum of $\max\left(\sum_{i=1}^{100} -d_i \log(1 + \bar{x}^k) - b, 0\right)$ (feasibility error) and $\frac{1}{100} \sum_{i=1}^{100} \|x_i^k - \bar{x}^k\|$ (consistency error). Here, $\bar{x}^k = \frac{1}{100} \sum_{i=1}^{100} x_i^k$.

At the beginning of the procedure, each agent i individually chooses the local decision vector x_i^0 . As Algorithm 1 runs, Fig. 2 shows that all agents reach a consensus decision by local computations and exchanging information with their neighbors. Furthermore, Fig. 3 shows that this consensus decision vector is an optimal solution with precision of 10^{-4} to the problem (5.1), which verifies the theoretical result of Algorithm 1. In summary, Algorithm 1 guides all agents to adjust their decision vectors to be feasible, consistent, and optimal.



(a) Evolution of the objective error



(b) Evolution of the constraint violation

Fig. 3 Convergence of algorithm 1

5.2 Neyman–Pearson Classification Problem

Suppose that h is a classifier to predict 1 and -1 . The type I error (misclassifying class -1 as 1) and type II error (misclassifying class 1 as -1) are, respectively, defined by

$$\text{type I error} := \mathbb{E}[\psi(-bh(a))|b = -1], \quad \text{type II error} := \mathbb{E}[\psi(-bh(a))|b = 1],$$

where ψ is some merit function. Different from the conventional binary classification in machine learning, the NP classification paradigm is developed to learn a classifier by minimizing type II error subject to that type I error is below a user-specified level $\tau > 0$, see [44] and references therein. Specifically, let \mathcal{H} be a given class of classifiers. The NP classification is to solve the following problem,

$$\min_{h \in \mathcal{H}} \mathbb{E}[\psi(-bh(a))|b = 1]$$

$$\text{s.t. } \mathbb{E}[\psi(-bh(a))|b = -1] \leq \tau.$$

In what follows, we consider its empirical risk minimization counterpart. Given a labeled training dataset $\{a_i\}_{i=1}^N$ consists of the positive set $\{a_i^+\}_{i=1}^{N_+}$ and the negative set $\{a_j^-\}_{j=1}^{N_-}$, The associated empirical NP classification problem is

$$\begin{aligned} \min_x \quad & f(x) = \frac{1}{N_+} \sum_{i=1}^{N_+} \ell(x^\top a_i^+) \\ \text{s.t.} \quad & g(x) := \frac{1}{N_-} \sum_{i=1}^{N_-} \ell(x^\top a_i^-) - \tau \leq 0, \end{aligned} \quad (5.2)$$

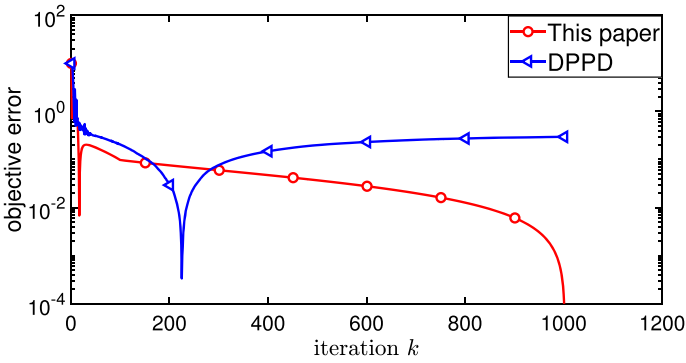
where $\ell(\cdot)$ is a loss function, and τ is a user-specified level, usually a small value. In this numerical test, we choose $\ell(\cdot)$ as the ℓ_2 -norm regularized logistic loss function, i.e., $\ell(x^\top a) := \log(1 + \exp(-x^\top a)) + \theta \|x\|_2^2$, $\theta = 10^{-2}$, $\tau = 5\%$. The data a_i^+ 's and a_i^- 's are randomly sampled from the dataset CINA [46]. The sample size $N_+ = N_- = 1000$ and the variable $x \in \mathfrak{R}^{132}$.

We run Algorithm 1 and the distributed proximal primal-dual (DPPD) method of [17] to solve (5.2). The step-size sequences of Algorithm 1 and DPPD are chosen as $0.1/k^{0.505}$ and $4/k^{0.5}$, respectively. The subproblems of DPPD are computed by Nesterov's accelerated gradient method [31] with a tolerance error of 10^{-6} . We start these two algorithms from the same initial point and plot the evolution of their objective errors and constraint violations during 1000 iterations, where the objective error is $|f(\bar{x}^k) - f^*|/|f^*|$ and the constraint violation is the sum of $\max(g(\bar{x}^k), 0)$ (feasibility error) and $\frac{1}{1000} \sum_{i=1}^{1000} \|x_i^k - \bar{x}^k\|$ (consistency error). Here, $\bar{x}^k = \frac{1}{1000} \sum_{i=1}^{1000} x_i^k$.

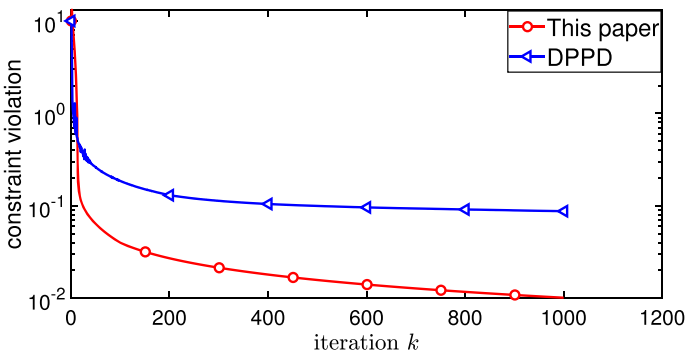
In Fig. 4a, the objective error of DPPD decays to a level of 10^{-3} after about 220 iterations, then suddenly climbs to near 10^{-1} and stays there. While the objective error of Algorithm 1 keeps decreasing except for the first few iterations and reaches 10^{-4} after 1000 iterations. As shown in Fig. 4b, the constraint violation of Algorithm 1 descends faster than that of DPPD, which indicates that the decision variable sequence generated by Algorithm 1 moves to be feasible at a faster speed. Moreover, the ratio of CPU time between Algorithm 1 and DPPD is 1 : 1.25. In summary, Fig. 4 illustrates that Algorithm 1 is efficient and has an advantage in convergence speed over DPPD.

6 Conclusions

This article focused on distributed convex optimization problems with coupled equality and inequality constraints over directed and time-varying multi-agent networks. We proposed a distributed primal-dual algorithm for such problems and established its convergence and iteration complexity. In numerical simulations, we showed that our algorithm is effective for the problem considered in this paper. However, there are still some respects that need to improve. One is that the presented algorithm adopts diminishing step-sizes, which leads to a slow convergence speed. Therefore, we expect



(a) Evolution of the objective error



(b) Evolution of the constraint violation

Fig. 4 Comparison of algorithm 1 and the distributed proximal primal-dual (DPPD) method [17]

to modify the diminishing step-sizes of Algorithm 1 to a fixed step-size in future studies.

Data Availability Statements The authors declare that all data supporting the findings of this study are available within the article and its supplementary information files.

A Proof of Lemma 4.1

For simplicity, denote $\delta_k = V_{k+1}^{-1} (\nabla L(\omega^{k+1}) - \nabla L(\omega^k))$, (4.5) can be equivalently written as $s^{k+1} = \mathcal{R}_k s^k + \delta_k$. Furthermore, $s^{k+1} = \Phi_{\mathcal{R}}(k, 1)s^1 + \sum_{l=1}^{k-1} \Phi_{\mathcal{R}}(k, l + 1)\delta_l + \delta_k$. Since $\{\frac{1}{2m}v_k\}$ is an absolute probability sequence for the matrix sequence $\{\mathcal{R}_k\}$, it follows that

$$1_{2m}\bar{s}^{k+1} = \frac{1}{2m} 1_{2m} v_{k+1}^\top s^{k+1}$$

$$\begin{aligned}
 &= \frac{1}{2m} 1_{2m} v_{k+1}^\top \left(\Phi_{\mathcal{R}}(k, 1) s^1 + \delta_k \right) + \frac{1}{2m} 1_{2m} v_{k+1}^\top \sum_{l=1}^{k-1} \Phi_{\mathcal{R}}(k, l+1) \delta_l \\
 &= \frac{1}{2m} 1_{2m} v_1^\top s^1 + \frac{1}{2m} \sum_{l=1}^{k-1} 1_{2m} v_l^\top \delta_l + \frac{1}{2m} 1_{2m} v_{k+1}^\top \delta_k.
 \end{aligned}$$

Applying the subadditivity of $\| \cdot \|$ yields

$$\begin{aligned}
 \|s^{k+1} - 1_{2m} \bar{s}^{k+1}\| \leq & \left\| \left(\Phi_{\mathcal{R}}(k, 1) - \frac{1}{2m} 1_{2m} v_1^\top \right) s^1 \right\| + \left\| \left(I_{2m} - \frac{1}{2m} 1_{2m} v_{k+1}^\top \right) \delta_k \right\| \\
 & + \sum_{l=1}^{k-1} \left\| \left(\Phi_{\mathcal{R}}(k, l+1) - \frac{1}{2m} 1_{2m} v_l^\top \right) \delta_l \right\|.
 \end{aligned} \tag{A.1}$$

For the matrix sequence $\{\Phi_{\mathcal{R}}(k, s)\}$, it holds the similar results in Lemma C.3, i.e.,

$$\left\| \Phi_{\mathcal{R}}(k, s) - \frac{1}{2m} 1_{2m} v_s^\top \right\| \leq C_1 \rho^{k-s}, \quad \forall k \geq s,$$

where $\rho \in (0, 1)$ and C_1 is some positive constant. Due to the compactness of the subset \mathcal{X} , it is clear that δ_k is bounded. Denote the upper bound of $\|\delta_k\|$ by \tilde{C} . Therefore,

$$\begin{aligned}
 \|s^{k+1} - 1_{2m} \bar{s}^{k+1}\| &\leq \rho^{k-1} \|s^1\| + \tilde{C} \sum_{l=1}^{k-1} \rho^{k-1-l} + \tilde{C} \\
 &\leq \|s^1\| + \frac{\tilde{C}}{1-\rho} + \tilde{C} = M, \quad \forall k \geq 1.
 \end{aligned}$$

B Proof of Lemma 4.2

Firstly, we need to verify the following inequality,

$$\left\| \omega^{k+1} - 1_{2m} \bar{\omega}^{k+1} \right\| \leq C_2 \beta^{k-1} + C_3 \sum_{l=1}^{k-1} \beta^{k-1-l} \alpha_l + C_4 \alpha_k.$$

Then, combining the conditions on the step-size sequence $\{\alpha_k\}$ in Assumption 3.1 and the results of Lemma C.4, it is easy to demonstrate the claimed results.

(a): From (4.4), there exist e^k such that $\omega^{k+1} = \mathcal{A}_k \omega^k - \alpha_k V_k s^k + e^k$. Similar to the inequality (A.1), it holds that

$$\begin{aligned} & \left\| \omega^{k+1} - 1_{2m} \bar{\omega}^{k+1} \right\| \\ & \leq \left\| \left(\Phi_{\mathcal{A}}(k, 1) - 1_{2m} \phi_1^\top \right) \omega^1 \right\| + \left\| \left(I_{2m} - 1_{2m} \phi_k^\top \right) \left(e^k - \alpha_k V_k s^k \right) \right\| \\ & \quad + \sum_{l=1}^{k-1} \left\| \left(\Phi_{\mathcal{A}}(k, l+1) - 1_{2m} \phi_l^\top \right) \left(e^l - \alpha_l V_l s^l \right) \right\|. \end{aligned} \tag{B.1}$$

Since ω^{k+1} is the projection of $\mathcal{A}_k \omega^k - \alpha_k V_k s^k$ onto Ω , for any $\omega \in \Omega$, one has

$$\left\langle \mathcal{A}_k \omega^k - \alpha_k V_k s^k - \omega^{k+1}, \omega - \omega^{k+1} \right\rangle \leq 0,$$

which implies that

$$\begin{aligned} & \left\| \omega^{k+1} - \mathcal{A}_k \omega^k \right\|^2 = \left\| e^k - \alpha_k V_k s^k \right\|^2 \\ & = \left\| e^k \right\|^2 - 2 \left\langle \alpha_k V_k s^k, e^k \right\rangle + \left\| \alpha_k V_k s^k \right\|^2 \\ & = \left\| e^k \right\|^2 + 2 \left\langle \omega^{k+1} - \mathcal{A}_k \omega^k - e^k, e^k \right\rangle + \left\| \alpha_k V_k s^k \right\|^2 \\ & \leq \alpha_k^2 \left\| V_k s^k \right\|^2 - \left\| e^k \right\|^2. \end{aligned}$$

From Lemma 4.1, it follows that

$$\left\| e^k \right\| \leq \alpha_k \left\| V_k s^k \right\| \leq \alpha_k \left\| s^k \right\| \leq M \alpha_k, \quad \forall k \geq 0.$$

In view of the results in Lemmas C.3 and C.4, one can get a modified version of the inequality (B.1) as follows,

$$\left\| \omega^{k+1} - 1_{2m} \bar{\omega}^{k+1} \right\| \leq C \left\| \omega^1 \right\| \beta^{k-1} + 2M \alpha_k + 2M \sum_{l=1}^{k-1} \beta^{k-1-l} \alpha_l. \tag{B.2}$$

Under Assumption 3.1 and Lemma C.4 (a), it’s clear that

$$\lim_{k \rightarrow \infty} \left\| \omega^k - 1_{2m} \bar{\omega}^k \right\| = 0.$$

(b): Multiplying the inequality (B.2) by α_{k+1} at the both sides yields

$$\begin{aligned} & \alpha_{k+1} \left\| \omega^{k+1} - 1_{2m} \bar{\omega}^{k+1} \right\| \\ & \leq C_2 \beta^{k-1} \alpha_{k+1} + 2C_3 \sum_{l=1}^{k-1} \beta^{k-1-l} \alpha_l \alpha_{k+1} + C_4 \alpha_k \alpha_{k+1} \end{aligned}$$

$$\begin{aligned} &\leq C_2 \left(\beta^{2(k-1)} + \alpha_{k+1}^2 \right) + C_3 \sum_{l=1}^{k-1} \beta^{k-1-l} \left(\alpha_l^2 + \alpha_{k+1}^2 \right) + C_4 \left(\alpha_k^2 + \alpha_{k+1}^2 \right) \\ &\leq C_1 \beta^{2(k-1)} + C_2 \sum_{l=1}^{k-1} \beta^{k-1-l} \alpha_l^2 + C_4 \alpha_k^2 + C_5 \alpha_{k+1}^2, \end{aligned}$$

where C_2, C_3, C_4, C_5 are some positive constants. Once again, under Assumption 3.1 and Lemma C.4 (b), it is concluded that

$$\sum_{k=0}^{\infty} \alpha_k \|\omega^k - 1_{2m} \bar{\omega}^k\| < \infty.$$

In Lemma 4.2, the claimed results about the sequence $\{s^k\}$ is parallel with $\{\omega^k\}$ and its proof is omitted here.

C Supporting Lemmas

Lemma C.1 ([29], Lemma 4) *Under Assumptions 2.1 and 2.2, it holds that*

- (a) $\lim_{k \rightarrow \infty} \Phi_A(k, s) = 1_{2m} \mu_s^\top$, where $\mu_s \in \mathbb{R}^{2m}$ is a stochastic vector for each s .
- (b) For any i , the entries $\Phi_A(k, s)_{ij}, j = 1, \dots, 2m$, converge to the same limit $(\mu_s)_i$ as $k \rightarrow \infty$ with a geometric rate, i.e., for each i and $s \geq 0$,

$$|\Phi_A(k, s)_{ij} - (\mu_s)_i| \leq C \beta^{k-s}, \quad \forall k \geq s,$$

where $C = 2 \frac{1+2\eta^{-B_0}}{1-2\eta^{B_0}} (1-\eta)^{1/B_0}$, $\beta = (1-\eta^{B_0}) \in (0, 1)$, η is the lower bound of Assumption 2.2, $B_0 = (m-1)B$, ingeter B is defined by Assumption 2.1.

Lemma C.2 ([39], Corollary 1) *Under the assumptions of Lemma C.1, the sequence $\{\phi_k\}$ is an absolute probability sequence for the matrix sequence $\{\mathcal{A}_k\}$, where*

$$\begin{aligned} \phi_k^\top &= \mu_s^\top, \quad k = sB, \\ \phi_k^\top &= \mu_{s+1}^\top \mathcal{A}_{(s+1)B-1} \cdots \mathcal{A}_k, \quad k \in (sB, (s+1)B), \end{aligned} \tag{C.1}$$

for $s = 0, 1, 2, \dots$.

Lemma C.3 ([34], Lemma 11, Chapter 2.2) *Let $\{b_k\}, \{c_k\}, \{d_k\}$ be non-negative sequences. Suppose that $\sum_{k=0}^{\infty} c_k < \infty$ and*

$$b_{k+1} \leq b_k - d_k + c_k, \quad \forall k \geq 1,$$

then the sequence $\{b_k\}$ converges and $\sum_{k=0}^{\infty} d_k < \infty$.

Lemma C.4 ([30], Lemma 7) *Let $\beta \in (0, 1)$, and $\{\gamma_k\}$ be a positive scalar sequence.*

- (a) *If $\lim_{k \rightarrow \infty} \gamma_k = 0$, then $\lim_{k \rightarrow \infty} \sum_{l=0}^k \beta^{k-l} \gamma_l = 0$.*
 (b) *Furthermore, if $\sum_{k=0}^{\infty} \gamma_k < \infty$, then $\sum_{k=0}^{\infty} \sum_{l=0}^k \beta^{k-l} \gamma_l < \infty$.*

References

- Alessandro, F., Ivano, N., Giuseppe, N., Maria, P.: Tracking-ADMM for distributed constraint-coupled optimization. *Automatica* **117**, 108,962 (2020)
- Alessandro, F., Kostas, M., Simone, G., Maria, P.: Dual decomposition for multi-agent distributed optimization with coupling constraints. *Automatica* **84**, 149–158 (2017)
- Alessandro, F., Maria, P.: Distributed decision-coupled constrained optimization via proximal-tracking. *Automatica* **135**, 109,938 (2022)
- Alessandro, F., Maria, P.: Augmented Lagrangian tracking for distributed optimization with equality and inequality coupling constraints. *Automatica* **157**, 111,269 (2023)
- Alghunaim, S.A., Lyu, Q., Yan, M., Sayed, A.H.: Dual consensus proximal algorithm for multi-agent sharing problems. *IEEE Trans. Signal Process.* **69**, 5568–5579 (2021)
- Arauz, T., Chanfreut, P., Maestre, J.: Cyber-security in networked and distributed model predictive control. *Annu. Rev. Control* **53**, 338–355 (2022)
- Arrow, K.J., Hurwicz, L., Uzawa, H.: *Studies in Linear and Nonlinear Programming*. Stanford University Press, Palo Alto (1958)
- Carli, R., Dotoli, M.: Distributed alternating direction method of multipliers for linearly constrained optimization over a network. *IEEE Control Syst. Lett.* **4**(1), 247–252 (2020)
- Chang, T.H.: A proximal dual consensus ADMM method for multi-agent constrained optimization. *IEEE Trans. Signal Process.* **64**(14), 3719–3734 (2016)
- Chang, T.H., Nedić, A., Scaglione, A.: Distributed constrained optimization by consensus-based primal-dual perturbation method. *IEEE Trans. Autom. Control* **59**(6), 1524–1538 (2014)
- CVX Research Inc.: CVX: Matlab software for disciplined convex programming (2012). <http://cvxr.com/cvx/>
- Hamedani, E.Y., Aybat, N.S.: Multi-agent constrained optimization of a strongly convex function over time-varying directed networks. In: 2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton), pp. 518–525 (2017)
- Hao, X., Liang, Y., Li, T.: Distributed estimation for multi-subsystem with coupled constraints. *IEEE Trans. Signal Process.* **70**, 1548–1559 (2022)
- Ion, N., Valentin, N.: On linear convergence of a distributed dual gradient algorithm for linearly constrained separable convex problems. *Automatica* **55**, 209–216 (2015)
- Lan, G., Zhou, Z.: Algorithms for stochastic optimization with function or expectation constraints. *Comput. Optim. Appl.* **76**, 461–498 (2020)
- Lei, J., Chen, H.F., Fang, H.T.: Primal-dual algorithm for distributed constrained optimization. *Syst. Control Lett.* **96**, 110–117 (2016)
- Li, X., Feng, G., Xie, L.: Distributed proximal algorithms for multiagent optimization with coupled inequality constraints. *IEEE Trans. Autom. Control* **66**(3), 1223–1230 (2021)
- Li, Z., Shi, W., Yan, M.: A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates. *IEEE Trans. Signal Process.* **67**(17), 4494–4506 (2019)
- Liang, S., Wang, L.Y., Yin, G.: Distributed smooth convex optimization with coupled constraints. *IEEE Trans. Autom. Control* **65**(1), 347–353 (2020)
- Liao, S.: A fast distributed algorithm for coupled utility maximization problem with application for power control in wireless sensor networks. *J. Commun. Netw.* **23**(4), 271–280 (2021)
- Liu, C., Li, H., Shi, Y.: A unitary distributed subgradient method for multi-agent optimization with different coupling sources. *Automatica* **114**, 108,834 (2020)
- Liu, H., Yu, W., Chen, G.: Discrete-time algorithms for distributed constrained convex optimization with linear convergence rates. *IEEE Trans. Cybern.* **52**(6), 4874–4885 (2022)
- Liu, Q., Yang, S., Hong, Y.: Constrained consensus algorithms with fixed step size for distributed convex optimization over multiagent networks. *IEEE Trans. Autom. Control* **62**(8), 4259–4265 (2017)
- Liu, T., Han, D., Lin, Y., Liu, K.: Distributed multi-UAV trajectory optimization over directed networks. *J. Frankl. Inst.* **358**(10), 5470–5487 (2021)

25. Mateos-Núñez, D., Cortés, J.: Distributed saddle-point subgradient algorithms with Laplacian averaging. *IEEE Trans. Autom. Control* **62**(6), 2720–2735 (2017)
26. Nedić, A., Olshevsky, A., Shi, W.: Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM J. Optim.* **27**(4), 2597–2633 (2017)
27. Nedić, A., Olshevsky, A., Shi, W.: Improved convergence rates for distributed resource allocation. In: 2018 IEEE Conference on Decision and Control (CDC), pp. 172–177 (2018)
28. Nedić, A., Ozdaglar, A.: Approximate primal solutions and rate analysis for dual subgradient methods. *SIAM J. Optim.* **19**(4), 1757–1780 (2009)
29. Nedić, A., Ozdaglar, A.: Distributed subgradient methods for multi-agent optimization. *IEEE Trans. Autom. Control* **54**(1), 48–61 (2009)
30. Nedić, A., Ozdaglar, A., Parrilo, P.A.: Constrained consensus and optimization in multi-agent networks. *IEEE Trans. Autom. Control* **55**(4), 922–938 (2010)
31. Nesterov, Y.: Gradient methods for minimizing composite functions. *Math. Program.* **140**, 125–161 (2013)
32. Notarnicola, I., Notarstefano, G.: Constraint-coupled distributed optimization: a relaxation and duality approach. *IEEE Trans. Control Netw. Syst.* **7**(1), 483–492 (2020)
33. Nowak, R.: Distributed EM algorithms for density estimation and clustering in sensor networks. *IEEE Trans. Signal Process.* **51**(8), 2245–2253 (2003)
34. Polyak, B.: Introduction to Optimization. Optimization Software, New York (2020)
35. Pu, S., Shi, W., Xu, J., Nedić, A.: Push-pull gradient methods for distributed optimization in networks. *IEEE Trans. Autom. Control* **66**(1), 1–16 (2021)
36. Qu, G., Li, N.: Harnessing smoothness to accelerate distributed optimization. *IEEE Trans. Control Netw. Syst.* **5**(3), 1245–1260 (2018)
37. Rockafellar, R.T.: *Convex Analysis*. Princeton Mathematical Series (1970)
38. Rockafellar, R.T., Uryasev, S.: Optimization of conditional value-at-risk. *J. Risk* **2**, 21–41 (2000)
39. Saadatniaqi, F., Xin, R., Khan, U.A.: Decentralized optimization over time-varying directed graphs with row and column-stochastic matrices. *IEEE Trans. Autom. Control* **65**(11), 4769–4780 (2020)
40. Shi, W., Ling, Q., Wu, G., Yin, W.: Extra: an exact first-order algorithm for decentralized consensus optimization. *SIAM J. Optim.* **25**(2), 944–966 (2015)
41. Shi, W., Ling, Q., Wu, G., Yin, W.: A proximal gradient algorithm for decentralized composite optimization. *IEEE Trans. Signal Process.* **63**(22), 6013–6023 (2015)
42. Simonetto, A., Jamali-Rad, H.: Primal recovery from consensus-based dual decomposition for distributed convex optimization. *J. Optim. Theory Appl.* **168**(1), 172–197 (2016)
43. Su, Y., Wang, Q., Sun, C.: Distributed primal-dual method for convex optimization with coupled constraints. *IEEE Trans. Signal Process.* **70**, 523–535 (2022)
44. Tong, X., Feng, Y., Zhao, A.: A survey on Neyman–Pearson classification and suggestions for future research. *WIREs Comput. Stat.* **8**(2), 64–81 (2016)
45. Wiltz, A., Chen, F., Dimos, V.D.: A consistency constraint-based approach to coupled state constraints in distributed model predictive control. In: 2022 IEEE 61st Conference on Decision and Control (CDC) pp. 3959–3964 (2022)
46. Workbench team C: A marketing dataset (2008). <http://www.causality.inf.ethz.ch/data/CINA.html>
47. Wu, X., Wang, H., Lu, J.: Distributed optimization with coupling constraints. *IEEE Trans. Autom. Control* **68**(3), 1847–1854 (2023)
48. Xu, J., Tian, Y., Sun, Y., Scutari, G.: Distributed algorithms for composite optimization: unified framework and convergence analysis. *IEEE Trans. Signal Process.* **69**, 3555–3570 (2021)
49. Xu, J., Zhu, S., Soh, Y.C., Xie, L.: Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes. In: 2015 54th IEEE Conference on Decision and Control (CDC), pp. 2055–2060 (2015)
50. Yuan, D., Proutiere, A., Shi, G.: Distributed online linear regressions. *IEEE Trans. Inf. Theory* **67**(1), 616–639 (2021)
51. Zhou, X., Ma, Z., Zou, S., Zhang, J.: Consensus-based distributed economic dispatch for multi micro energy grid systems under coupled carbon emissions. *Appl. Energy* **324**, 119641 (2022)
52. Zhu, K., Tang, Y.: Primal-dual ϵ -subgradient method for distributed optimization. *J. Syst. Sci. Complex.* **36**, 577–590 (2020)
53. Zhu, M., Martinez, S.: On distributed convex optimization under inequality and equality constraints. *IEEE Trans. Autom. Control* **57**(1), 151–164 (2012)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.