Mathematical Programming (2020) 180:237–284
https://doi.org/10.1007/s10107-018-1355-4

FULL LENGTH PAPER

Series A

Communication-efficient algorithms for decentralized and stochastic optimization

Guanghui Lan[1] · Soomin Lee[1] · Yi Zhou[1]

Received: 16 January 2017 / Accepted: 1 December 2018 / Published online: 7 December 2018

Abstract

We present a new class of decentralized first-order methods for nonsmooth and stochastic optimization problems defined over multiagent networks. Considering that communication is a major bottleneck in decentralized optimization, our main goal in this paper is to develop algorithmic frameworks which can significantly reduce the number of inter-node communications. Our major contribution is to present a new class of decentralized primal–dual type algorithms, namely the decentralized communication sliding (DCS) methods, which can skip the inter-node communications while agents solve the primal subproblems iteratively through linearizations of their local objective functions. By employing DCS, agents can find an ϵ-solution both in terms of functional optimality gap and feasibility residual in $\mathcal{O}(1/\epsilon)$ (resp., $\mathcal{O}(1/\sqrt{\epsilon})$) communication rounds for general convex functions (resp., strongly convex functions), while maintaining the $\mathcal{O}(1/\epsilon^2)$ (resp., $\mathcal{O}(1/\epsilon)$) bound on the total number of intra-node subgradient evaluations. We also present a stochastic counterpart for these algorithms, denoted by SDCS, for solving stochastic optimization problems whose objective function cannot be evaluated exactly. In comparison with existing results for decentralized nonsmooth and stochastic optimization, we can reduce the total number of inter-node communication rounds by orders of magnitude while still maintaining the optimal complexity bounds on intra-node stochastic subgradient evaluations. The bounds on the (stochastic) subgradient evaluations are actually comparable to those required for centralized nonsmooth and stochastic optimization under certain conditions on the target accuracy.

Keywords Decentralized optimization · Decentralized machine learning · Communication efficient · Stochastic programming · Nonsmooth functions · Primal–dual method · Complexity

This work was funded by National Science Foundation Grants 1637473 and 1637474, and Office of Naval Research grant N00014-16-1-2802.

✉ Guanghui Lan
george.lan@isye.gatech.edu

Extended author information available on the last page of the article

Mathematics Subject Classification 90C25 · 90C06 · 90C22 · 49M37 · 93A14 · 90C15

1 Introduction

Decentralized optimization problems defined over complex multiagent networks are ubiquitous in signal processing, machine learning, control, and other areas in science and engineering (see e.g. [18,25,51,54]). In this paper, we consider the following decentralized optimization problem which is cooperatively solved by the network of m agents:

$$
\begin{aligned}
f^* := \min_x \; & f(x) := \sum_{i=1}^m f_i(x) \\
\text{s.t. } \; & x \in X, \quad X := \cap_{i=1}^m X_i,
\end{aligned} \tag{1.1}
$$

where $f_i : X_i \to \mathbb{R}$ is a convex and possibly nonsmooth objective function of agent i satisfying

$$
\tfrac{\mu}{2}\|x-y\|^2 \le f_i(x) - f_i(y) - \langle f_i'(y), x-y\rangle \le M\|x-y\|, \quad \forall x, y \in X_i, \tag{1.2}
$$

for some $M, \mu \ge 0$ and $f_i'(y) \in \partial f_i(y)$, where $\partial f_i(y)$ denotes the subdifferential of f_i at y, and $X_i \subseteq \mathbb{R}^d$ is a closed convex constraint set of agent i. Note that f_i and X_i are private and only known to agent i. Throughout the paper, we assume the feasible set X is nonempty.

In this paper, we also consider the situation where one can only have access to noisy first-order information (function values and subgradients) of the functions f_i, $i = 1, \ldots, m$ (see [27,45]). This happens, for example, when the function f_i's are given in the form of expectation, i.e.,

$$
f_i(x) := \mathbb{E}_{\xi_i}[F_i(x; \xi_i)], \tag{1.3}
$$

where the random variable ξ_i models a source of uncertainty and the distribution $\mathbb{P}(\xi_i)$ is not known in advance. As a special case of (1.3), f_i may be given as the summation of many components, i.e.,

$$
f_i(x) := \sum_{j=1}^l f_i^j(x), \tag{1.4}
$$

where $l \ge 1$ is a large number. Stochastic optimization problem of this type has great potential of applications in data analysis, especially in machine learning. In particular, problem (1.3) corresponds to the minimization of generalized risk and is particularly useful for dealing with online (streaming) data distributed over a network, while problem (1.4) aims at the collaborative minimization of empirical risk.

Currently the dominant approach to solve (1.1) is to collect all agents' private data on a server (or cluster) and to apply centralized machine learning techniques. However, this centralization scheme would require agents to submit their private data to the service provider without much control on how the data will be used, in addition to incurring high setup cost related to the transmission of data to the service provider. Decentralized optimization provides a viable approach to deal with these data privacy related issues. Each network agent i is associated with the local objective function $f_i(x)$ and all agents intend to cooperatively minimize the system objective $f(x)$ as the sum of all local objective f_i's in the absence of full knowledge about the global problem and network structure. A necessary feature in decentralized optimization is, therefore, that the agents must communicate with their neighboring agents to propagate the distributed information to every location in the network.

Decentralized optimization has been extensively studied in recent years due to the emergence of large-scale networks. The seminal work on distributed optimization [62,63] has been followed by distributed incremental (sub)gradient methods and proximal methods [4,40,52,64], and more recently the incremental aggregated gradient methods and its proximal variants [5,21,30]. All of these incremental methods are not fully decentralized in a sense that they require a special star network topology in which the existence of a central authority is necessary for operation. To consider a more general distributed network topology without a central authority, a decentralized subgradient algorithm was first proposed in [43], and further studied in many other literature (see e.g. [17,39,41,60,68]). These subgradient based methods require each node to compute a local subgradient and followed by the communication with neighboring agents iteratively, and achieve rate of convergence as $\mathcal{O}(1/\epsilon^2)$ to obtain an ϵ-optimal solution, i.e., a point $\hat{x} \in X$, s.t., $\mathbb{E}[f(\hat{x}) - f^*] \le \epsilon$. While the subgradient computation at each step can be inexpensive, due to the fact that one iteration in decentralized optimization is equivalent to at least one communication round among agents, these methods can incur a significant latency for solving (1.1). In fact, CPUs in these days can read and write the memory at over 10–100 GB per second whereas communication over TCP/IP is about 100 MB per second. Therefore, the gap between intra-node computation and inter-node communication is about 3 orders of magnitude. The communication start-up cost itself is also not negligible as it usually takes a few milliseconds. Improvements on communication complexity can be obtained when the objective function (1.1) is smooth and/or strongly convex (see, e.g., [42,49,56,57]). However, these algorithms do not apply to general nonsmooth and stochastic optimization to be studied in this work.

Besides subgradient based methods, another well-known type of decentralized algorithms relies on dual methods (see e.g., [3,6,32,55,59,65]), where at each step for a fixed dual variable, the primal variables are solved to minimize some local Lagrangian related function, then the dual variables associated with the consistency constraints are updated accordingly. More specifically, the decentralized dual decomposition method proposed in [59] obtained an implicit rate of converge for solving the Lagrangian dual problem of (1.1) with bounded communication delays. Furthermore, decentralized alternating direction method of multipliers (ADMM) algorithms (see, e.g., [6,32,55,65]) have received much attention recently. For relatively simple convex functions f_i, the decentralized ADMM proposed in [65] has been shown to require

$\mathcal{O}(1/\epsilon)$ communications (see also [23] for the application of mirror-prox method for solving these problems). An improved $\mathcal{O}(\log 1/\epsilon)$ complexity bound on communication rounds can be achieved for decentralized ADMM [32,55] if stronger assumptions, i.e., smoothness and strong convexity, are imposed on f_i. These dual-based methods have been further studied via proximal-gradients [3,11,12]. Although dual type methods usually require fewer numbers of iterations (hence, fewer communication rounds) than the subgradient based methods, the local Lagrangian minimization problem associated with each agent cannot be solved efficiently in many cases, especially when the problem is constrained. Second-order approximation methods [33,34] have been studied in order to handle this issue, but due to the nature of these methods differentiability of the objective function is necessary in this case.

Moreover, multi-step consensus has been considered in decentralized methods for solving (1.1) with smoothness assumption, and hence these methods require an increasing number of communication rounds iteratively. For example, the distributed Nesterov's accelerated gradient method [26] employs multi-consensus in the inner-loop. Although their method requires $\mathcal{O}(1/\sqrt{\epsilon})$ intra-node gradient computations, inter-node communications must increase at a rate of $\mathcal{O}(\log(k))$ as the iteration k increases. Similarly, the proximal gradient method with adapt-then-combine (ATC) multi-consensus strategy and Nesterov's acceleration under the assumption of bounded and Lipschitz gradients [13] requires that inter-node communications must increase at a rate of $\mathcal{O}(k)$.

However, the multi-consensus schemes in nested loop algorithms are less desirable, since they do not account for the fact that the time required for inter-node communications is higher by a few orders of magnitude than that for intra-node computations.

While decentralized algorithms for solving deterministic optimization problems have been extensively studied during the past few years, there exists only limited research on decentralized stochastic optimization, for which only noisy gradient information of functions f_i, $i = 1, \ldots, m$, in (1.1) can be easily computed. Existing decentralized stochastic first-order methods for problem (1.1) (e.g., [17,39,53]) require $\mathcal{O}(1/\epsilon^2)$ inter-node communications and intra-node gradient computations to obtain an ϵ-optimal solution for solving general convex problems. When the objective functions are strongly convex, multiagent mirror descent method for decentralized stochastic optimization can achieve an $\mathcal{O}(1/\epsilon)$ complexity bound [50]. An alternative form of mirror descent in the multiagent setting was proposed by [66] with an asymptotic convergence result. On a broader scale, decentralized stochastic optimization was also considered in the case of time-varying objective functions in the recent work [58,61]. All these previous works in decentralized stochastic optimization suffered from high communication costs due to the coupled scheme for stochastic subgradient evaluation and communication, i.e., each evaluation of stochastic subgradient will incur one round of communication.

The main goal of this paper is to develop dual based decentralized algorithms for solving (1.1) which are communication efficient and have local subproblems approximately solved by each agent through the utilization of (noisy) first-order information of f_i. More specifically, we will provide a theoretical understanding on how many rounds of inter-node communications and intra-node (stochastic) subgradient computations of f_i are required in order to find a certain approximate solution of (1.1) in which f_i's are convex or strongly convex, but not necessarily smooth, and their exact first-order information is not necessarily computable. Our contributions in this paper are listed below.

Firstly, we introduce a new decentralized primal-dual type method, called decentralized communication sliding (DCS), where the agents can skip communications while solving their local subproblems iteratively through successive linearizations of their local objective functions. We show that agents can still find an ϵ-optimal solution in $\mathcal{O}(1/\epsilon)$ (resp., $\mathcal{O}(1/\sqrt{\epsilon})$) communication rounds while maintaining the $\mathcal{O}(1/\epsilon^2)$ (resp., $\mathcal{O}(1/\epsilon)$) bound on the total number of intra-node subgradient evaluations when the objective functions are general convex (resp., strongly convex). The bounds on the subgradient evaluations are actually comparable to those optimal complexity bounds required for centralized nonsmooth optimization under certain conditions on the target accuracy, and hence are not improvable in general.

Secondly, we present a stochastic decentralized communication sliding method, denoted by SDCS, for solving stochastic optimization problems and show complexity bounds similar to those of DCS on the total number of required communication rounds and stochastic subgradient evaluations. In particular, only $\mathcal{O}(1/\epsilon)$ (resp., $\mathcal{O}(1/\sqrt{\epsilon})$) communication rounds are required while agents perform up to $\mathcal{O}(1/\epsilon^2)$ (resp., $\mathcal{O}(1/\epsilon)$) stochastic subgradient evaluations for general convex (resp., strongly convex) functions. Only requiring the access to stochastic subgradient at each iteration, SDCS is particularly efficient for solving problems with f_i given in the form of (1.3) and (1.4). In the former case, SDCS requires only one realization of the random variable at each iteration and provides a communication-efficient way to deal with streaming data and decentralized machine learning. In the latter case, each iteration of SDCS requires only one randomly selected component, leading up to a factor of $\mathcal{O}(l)$ savings on the total number of subgradient computations over DCS.

Thirdly, we demonstrate the possible advantages of our proposed methods through preliminary numerical experiments for solving decentralized support vector machine (SVM) problems with real data sets. For all our test problems, DCS and SDCS can significantly save communication costs over some existing state-of-the-art decentralized methods.

To the best of our knowledge, this is the first time that these communication sliding algorithms, and the aforementioned separate complexity bounds on communication rounds and (stochastic) subgradient evaluations are presented in the literature. Table 1 summarizes the improvement on communication complexity obtained by our algorithms over existing methods for decentralized nonsmooth and stochastic optimization.

This paper is organized as follows. In Sect. 2, we introduce the problem formulation and the definition of the gap function, which will be used as the termination criterion of our methods. We also provide some preliminaries on distance generating functions and prox-functions. In Sect. 3, we present the communication sliding algorithms when the exact subgradients of f_i's are available and establish their convergence properties for the general and strongly convex cases. In Sect. 4, we generalize the algorithm in Sect. 3 for stochastic optimization problems. The proofs of some important technical results in Sects. 3 and 4 are provided in Sect. 5. We also provide some preliminary numerical results in Sect. 6 to demonstrate the advantages of our algorithms. Some concluding remarks are made in Sect. 7.

Table 1 Summary of communication complexities for obtaining a (stochastic) ϵ-solution of (1.1)

Problem type: f_i	Communication rounds	
	Our results	Existing results
Deterministic, convex	$\mathcal{O}\{1/\epsilon\}$	$\mathcal{O}\left\{1/\epsilon^2\right\}$
Deterministic, strongly convex	$\mathcal{O}\left\{1/\sqrt{\epsilon}\right\}$	$\mathcal{O}\{1/\epsilon\}$
Stochastic, convex	$\mathcal{O}\{1/\epsilon\}$	$\mathcal{O}\left\{1/\epsilon^2\right\}$
Stochastic, strongly convex	$\mathcal{O}\left\{1/\sqrt{\epsilon}\right\}$	$\mathcal{O}\{1/\epsilon\}$

1.1 Notation and terminologies

Let $\mathbb{R}$ denote the set of real numbers. All vectors are viewed as column vectors, and for a vector $x \in \mathbb{R}^d$, we use $x^\top$ to denote its transpose. For a stacked vector of x_i's, we often use $(x_1, \ldots, x_m)$ to represent the column vector $[x_1^\top, \ldots, x_m^\top]^\top$. We denote by $\mathbf{0}$ and $\mathbf{1}$ the vector of all zeros and ones whose dimensions vary from the context. The cardinality of a set S is denoted by $|S|$. We use I_d to denote the identity matrix in $\mathbb{R}^{d\times d}$. We use $A \otimes B$ for matrices $A \in \mathbb{R}^{n_1\times n_2}$ and $B \in \mathbb{R}^{m_1\times m_2}$ to denote their Kronecker product of size $\mathbb{R}^{n_1 m_1\times n_2 m_2}$. For a matrix $A \in \mathbb{R}^{n\times m}$, we use A_{ij} to denote the entry of i-th row and j-th column. For any $m \geq 1$, the set of integers $\{1, \ldots, m\}$ is denoted by $[m]$.

2 Preliminaries

In Sects. 2.1 and 2.2 we introduce the saddle point reformulation of (1.1) and define appropriate gap functions which will be used for the convergence analysis of our algorithms. Moreover, in Sect. 2.3 we provide a brief review on the distance generating function and prox-function.

2.1 Problem formulation

Consider a multiagent network system whose communication is governed by an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N} = [m]$ indexes the set of agents, and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ represents the pairs of communicating agents. If there exists an edge from agent i to j which we denote by (i, j), agent i may send its information to agent j and vice versa. Thus, each agent $i \in \mathcal{N}$ can directly receive (resp., send) information only from (resp., to) the agents in its neighborhood

$$N_i = \{j \in \mathcal{N} \mid (i, j) \in \mathcal{E}\} \cup \{i\}, \tag{2.1}$$

where we assume that there always exists a self-loop (i, i) for all agents $i \in \mathcal{N}$. Then, the associated Laplacian $L \in \mathbb{R}^{m\times m}$ of $\mathcal{G}$ is $L := D - A$ where D is the diagonal degree matrix, and $A \in \mathbb{R}^{m\times m}$ is the adjacency matrix with the property that $A_{ij} = 1$ if and only if $(i, j) \in \mathcal{E}$ and $i \neq j$, i.e.,

$$L_{ij} = \begin{cases} |N_i| - 1 & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } (i, j) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \tag{2.2}$$

We consider a reformulation of problem (1.1) which will be used in the development of our decentralized algorithms. We introduce an individual copy x_i of the decision variable x for each agent $i \in \mathcal{N}$ and impose the constraint $x_i = x_j$ for all pairs $(i, j) \in \mathcal{E}$. The transformed problem can be written compactly by using the Laplacian matrix L:

$$\begin{aligned} &\min_{\mathbf{x}} F(\mathbf{x}) := \sum_{i=1}^{m} f_i(x_i) \\ &\text{s.t. } \mathbf{L}\mathbf{x} = \mathbf{0}, \quad x_i \in X_i, \text{ for all } i = 1, \ldots, m, \end{aligned} \tag{2.3}$$

where $\mathbf{x} = (x_1, \ldots, x_m) \in X_1 \times \ldots \times X_m$, $F : X_1 \times \ldots \times X_m \to \mathbb{R}$, and $\mathbf{L} = L \otimes I_d \in \mathbb{R}^{md \times md}$. The constraint $\mathbf{L}\mathbf{x} = \mathbf{0}$ is a compact way of writing $x_i = x_j$ for all agents i and j which are connected by an edge. By construction and Theorem 4.2.12 in [24], $\mathbf{L}$ is symmetric positive semidefinite and its null space coincides with the "agreement" subspace, i.e., $\mathbf{L}\mathbf{1} = \mathbf{0}$ and $\mathbf{1}^\top \mathbf{L} = \mathbf{0}$. To ensure each node gets information from every other node, we need the following assumption.

Assumption 1 *The graph $\mathcal{G}$ is connected.*

Under Assumption 1, problem (1.1) and (2.3) are equivalent. We let Assumption 1 be a blanket assumption for the rest of the paper.

We next consider a reformulation of the problem (2.3) as a saddle point problem. By the method of Lagrange multipliers, problem (2.3) is equivalent to the following saddle point problem:

$$\min_{\mathbf{x} \in \mathbf{X}} \left[F(\mathbf{x}) + \max_{\mathbf{y} \in \mathbb{R}^{md}} \langle \mathbf{L}\mathbf{x}, \mathbf{y} \rangle \right], \tag{2.4}$$

where $\mathbf{X} := X_1 \times \ldots \times X_m$ and $\mathbf{y} = (y_1, \ldots, y_m) \in \mathbb{R}^{md}$ are the Lagrange multipliers associated with the constraints $\mathbf{L}\mathbf{x} = \mathbf{0}$. We assume that there exists an optimal solution $\mathbf{x}^* \in \mathbf{X}$ of (2.3) and that there exists $\mathbf{y}^* \in \mathbb{R}^{md}$ such that $(\mathbf{x}^*, \mathbf{y}^*)$ is a saddle point of (2.4). In fact, since our objective function $F(\mathbf{x})$ is convex, strong duality holds if constraint qualification (CQ) condition holds (see Strong Duality Theorem in [31, Section 14.1]). In particular, CQ condition states that there exists $\bar{\mathbf{x}} \in \mathbf{X}$ such that $\mathbf{L}\bar{\mathbf{x}} = \mathbf{0}$, which is implied by the assumption that there exists an optimal solution to (2.3).

2.2 Gap functions: termination criteria

Given a pair of feasible solutions $\mathbf{z} = (\mathbf{x}, \mathbf{y})$ and $\bar{\mathbf{z}} = (\bar{\mathbf{x}}, \bar{\mathbf{y}})$ of (2.4), we define the *primal–dual gap function* $Q(\mathbf{z}; \bar{\mathbf{z}})$ by

$$Q(\mathbf{z}; \bar{\mathbf{z}}) := F(\mathbf{x}) + \langle \mathbf{L}\mathbf{x}, \bar{\mathbf{y}} \rangle - [F(\bar{\mathbf{x}}) + \langle \mathbf{L}\bar{\mathbf{x}}, \mathbf{y} \rangle]. \tag{2.5}$$

Sometimes we also use the notations $Q(\mathbf{z}; \bar{\mathbf{z}}) := Q(\mathbf{x}, \mathbf{y}; \bar{\mathbf{x}}, \bar{\mathbf{y}})$ or $Q(\mathbf{z}; \bar{\mathbf{z}}) := Q(\mathbf{x}, \mathbf{y}; \bar{\mathbf{z}}) = Q(\mathbf{z}; \bar{\mathbf{x}}, \bar{\mathbf{y}})$. One can easily see that $Q(\mathbf{z}^*; \mathbf{z}) \le 0$ and $Q(\mathbf{z}; \mathbf{z}^*) \ge 0$ for all $\mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$, where $\mathbf{z}^* = (\mathbf{x}^*, \mathbf{y}^*)$ is a saddle point of (2.4). For compact sets $\mathbf{X} \subset \mathbb{R}^{md}$, $Y \subset \mathbb{R}^{md}$, the gap function

$$\sup_{\bar{\mathbf{z}} \in \mathbf{X} \times Y} Q(\mathbf{z}; \bar{\mathbf{z}}) \tag{2.6}$$

measures the accuracy of the approximate solution $\mathbf{z}$ to the saddle point problem (2.4).

However, the saddle point formulation (2.4) of our problem of interest (1.1) may have an unbounded feasible set. We adopt the perturbation-based termination criterion by Monteiro and Svaiter [35–37] and propose a modified version of the gap function in (2.6). More specifically, we define

$$g_Y(\mathbf{s}, \mathbf{z}) := \sup_{\bar{\mathbf{y}} \in Y} Q(\mathbf{z}; \mathbf{x}^*, \bar{\mathbf{y}}) - \langle \mathbf{s}, \bar{\mathbf{y}} \rangle, \tag{2.7}$$

for any closed set $Y \subseteq \mathbb{R}^{md}$, $\mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$ and $\mathbf{s} \in \mathbb{R}^{md}$. If $Y = \mathbb{R}^{md}$, we omit the subscript Y and simply use the notation $g(\mathbf{s}, \mathbf{z})$.

This perturbed gap function allows us to bound the objective function value and the feasibility separately. We first define the following terminology.

Definition 1 A point $\mathbf{x} \in \mathbf{X}$ is called an (ϵ, δ)-solution of (2.3) if

$$F(\mathbf{x}) - F(\mathbf{x}^*) \le \epsilon \text{ and } \|\mathbf{L}\mathbf{x}\| \le \delta. \tag{2.8}$$

We say that $\mathbf{x}$ has primal residual ϵ and feasibility residual δ.

Similarly, a stochastic (ϵ, δ)-solution of (2.3) can be defined as a random point $\hat{\mathbf{x}} \in \mathbf{X}$ s.t. $\mathbb{E}[F(\hat{\mathbf{x}}) - F(\mathbf{x}^*)] \le \epsilon$ and $\mathbb{E}[\|\mathbf{L}\hat{\mathbf{x}}\|] \le \delta$ for some $\epsilon, \delta > 0$. Note that for problem (2.3), the feasibility residual measures the disagreement among the local copies x_i, for $i \in \mathcal{N}$.

In the following proposition, we adopt a result from [48, Proposition 2.1] to describe the relationship between the perturbed gap function (2.7) and the approximate solutions to problem (2.3). Although the proposition was originally developed for deterministic cases, the extension of this to stochastic cases is straightforward.

Proposition 1 *For any $Y \subset \mathbb{R}^{md}$ such that $\mathbf{0} \in Y$, if $g_Y(\mathbf{L}\mathbf{x}, \mathbf{z}) \le \epsilon < \infty$ and $\|\mathbf{L}\mathbf{x}\| \le \delta$, where $\mathbf{z} = (\mathbf{x}, \mathbf{y}) \in \mathbf{X} \times \mathbb{R}^{md}$, then $\mathbf{x}$ is an (ϵ, δ)-solution of (2.3). In particular, when $Y = \mathbb{R}^{md}$, for any $\mathbf{s}$ such that $g(\mathbf{s}, \mathbf{z}) \le \epsilon < \infty$ and $\|\mathbf{s}\| \le \delta$, we always have $\mathbf{s} = \mathbf{L}\mathbf{x}$.*

2.3 Distance generating function and prox-function

In this subsection, we define the concept of prox-function, which is also known as proximity control function or Bregman distance function [8]. Prox-function has played an important role in the recent development of first-order methods for convex programming as a substantial generalization of the Euclidean projection. Unlike the standard projection operator $\Pi_U[x] := \operatorname{argmin}_{u \in U} \|x - u\|^2$, which is inevitably tied to the Euclidean geometry, prox-function can be flexibly tailored to the geometry of a constraint set U.

For any convex set U equipped with an arbitrary norm $\|\cdot\|_U$, we say that a function $\omega : U \to \mathbb{R}$ is a *distance generating function* with modulus $\nu > 0$ with respect to $\|\cdot\|_U$, if ω is continuously differentiable and strongly convex with modulus ν with respect to $\|\cdot\|_U$, i.e.,

$$\langle \nabla\omega(x) - \nabla\omega(u), x - u \rangle \geq \nu\|x - u\|_U^2, \quad \forall x, u \in U. \tag{2.9}$$

The *prox-function*, or *Bregman distance function*, induced by ω is given by

$$V(x, u) \equiv V_\omega(x, u) := \omega(u) - [\omega(x) + \langle \nabla\omega(x), u - x \rangle]. \tag{2.10}$$

It then follows from the strong convexity of ω that

$$V(x, u) \geq \tfrac{\nu}{2}\|x - u\|_U^2, \quad \forall x, u \in U.$$

We now assume that the individual constraint set X_i for each agent in problem (1.1) are equipped with norm $\|\cdot\|_{X_i}$, and their associated prox-functions are given by $V_i(\cdot, \cdot)$. Moreover, we assume that each $V_i(\cdot, \cdot)$ shares the same strongly convex modulus $\nu = 1$, i.e.,

$$V_i(x_i, u_i) \geq \tfrac{1}{2}\|x_i - u_i\|_{X_i}^2, \quad \forall x_i, u_i \in X_i, \ i = 1, \ldots, m. \tag{2.11}$$

We define the norm associated with the primal feasible set $\mathbf{X} = X_1 \times \ldots \times X_m$ of (2.4) as follows:[1]

$$\|\mathbf{x}\|^2 \equiv \|\mathbf{x}\|_{\mathbf{X}}^2 := \sum_{i=1}^m \|x_i\|_{X_i}^2, \tag{2.12}$$

where $\mathbf{x} = (x_1, \ldots, x_m) \in \mathbf{X}$ for any $x_i \in X_i$. Therefore, the corresponding prox-function $\mathbf{V}(\cdot, \cdot)$ can be defined as

$$\mathbf{V}(\mathbf{x}, \mathbf{u}) := \sum_{i=1}^m V_i(x_i, u_i), \quad \forall \mathbf{x}, \mathbf{u} \in \mathbf{X}. \tag{2.13}$$

[1] We can define the norm associated with $\mathbf{X}$ in a more general way, e.g., $\|\mathbf{x}\|^2 := \sum_{i=1}^m p_i \|x_i\|_{X_i}^2$, $\forall \mathbf{x} = (x_1, \ldots, x_m) \in \mathbf{X}$, for some $p_i > 0$, $i = 1, \ldots, m$. Accordingly, the prox-function $\mathbf{V}(\cdot, \cdot)$ can be defined as $\mathbf{V}(\mathbf{x}, \mathbf{u}) := \sum_{i=1}^m p_i V_i(x_i, u_i)$, $\forall \mathbf{x}, \mathbf{u} \in \mathbf{X}$. This setting gives us flexibility to choose p_i's based on the information of individual X_i's, and the possibility to further refine the convergence results.

Note that by (2.11) and (2.12), it can be easily seen that

$$\mathbf{V}(\mathbf{x}, \mathbf{u}) \geq \tfrac{1}{2}\|\mathbf{x} - \mathbf{u}\|^2, \ \forall \mathbf{x}, \mathbf{u} \in \mathbf{X}. \tag{2.14}$$

Throughout the paper, we endow the dual space where the multipliers $\mathbf{y}$ of (2.4) reside with the standard Euclidean norm $\|\cdot\|_2$, since the feasible region of $\mathbf{y}$ is unbounded. For simplicity, we often write $\|\mathbf{y}\|$ instead of $\|\mathbf{y}\|_2$ for a dual multiplier $\mathbf{y} \in \mathbb{R}^{md}$.

3 Decentralized communication sliding

In this section, we introduce a primal-dual algorithmic framework, namely, the decentralized communication sliding (DCS) method, for solving the saddle point problem (2.4) in a decentralized fashion. Moreover, we will establish complexity bounds on the required number of inter-node communication rounds as well as the total number of required subgradient evaluations. Throughout this section, we consider the deterministic case where exact subgradients of f_i's are available.

3.1 The DCS algorithm

The basic scheme of the DCS algorithm is inspired by Chambolle and Pock's primal–dual method in [10]. The primal–dual method in [10] is an efficient and simple method for solving saddle point problems, which can be viewed as a refined version of the primal–dual hybrid gradient method by Arrow et al. [1]. However, its analysis is more closely related to a few recent important works for solving bilinear saddle point problems (e.g., [22,38,44,47]). When applied to our saddle point reformulation defined in (2.4), for any given initial points $\mathbf{x}^0 = \mathbf{x}^{-1} \in \mathbf{X}$ and $\mathbf{y}^0 \in \mathbb{R}^{md}$, and certain nonnegative parameters $\{\alpha_k\}$, $\{\tau_k\}$ and $\{\eta_k\}$, the primal–dual method updates $(\mathbf{x}^k, \mathbf{y}^k)$ according to

$$\tilde{\mathbf{x}}^k = \alpha_k(\mathbf{x}^{k-1} - \mathbf{x}^{k-2}) + \mathbf{x}^{k-1}, \tag{3.1}$$

$$\mathbf{y}^k = \mathrm{argmin}_{\mathbf{y} \in \mathbb{R}^{md}} \langle -\mathbf{L}\tilde{\mathbf{x}}^k, \mathbf{y}\rangle + \tfrac{\tau_k}{2}\|\mathbf{y} - \mathbf{y}^{k-1}\|^2, \tag{3.2}$$

$$\mathbf{x}^k = \mathrm{argmin}_{\mathbf{x} \in \mathbf{X}} \left\{ \Phi^k(\mathbf{x}) := \langle \mathbf{L}\mathbf{y}^k, \mathbf{x}\rangle + F(\mathbf{x}) + \eta_k \mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x}) \right\}. \tag{3.3}$$

Note that the incorporation of the Bregman distance into the primal–dual method (see (3.3)) was first introduced in [14].

In each iteration of the primal–dual method, only the computation of the matrix-vector products $\mathbf{L}\tilde{\mathbf{x}}^k$ and $\mathbf{L}\mathbf{y}^k$ will involve the communication among different agents, while the other computations such as the updating of $\tilde{\mathbf{x}}^k$, $\mathbf{y}^k$ and $\mathbf{x}^k$ can be performed separately by each agent. Under the assumption that the subproblem (3.3) can be easily solved, we can show that by properly choosing the algorithmic parameters α_k, τ_k and η_k one can find an ϵ-solution, i.e., a point $\bar{\mathbf{x}} \in \mathbf{X}$ such that $F(\bar{\mathbf{x}}) - F(\mathbf{x}^*) \leq \epsilon$ and $\|\mathbf{L}\bar{\mathbf{x}}\| \leq \epsilon$, within $\mathcal{O}(1/\epsilon)$ iterations ([2,9,15,22,38,44,47]). This implies that one can find such an ϵ-solution in $\mathcal{O}(1/\epsilon)$ rounds of communication, which already improves the existing $\mathcal{O}(1/\epsilon^2)$ communication complexity for decentralized nonsmooth optimization. However, such a communication complexity bound is not quite meaningful because F is a general nonsmooth convex function and it is often difficult to solve the primal subproblem (3.3) explicitly.

One natural way to address this issue is to approximately solve (3.3) through an iterative subgradient descent method. Inside this iterative subgradient descent method, we do not need to re-compute the matrix-vector products $\mathbf{L}\tilde{\mathbf{x}}^k$ and $\mathbf{L}\mathbf{y}^k$, and hence no communication cost is involved. However, a straightforward pursuit of this approach, i.e., to solve the subproblem accurately enough at each iteration, does not necessarily yield the best complexity bound in terms of the total number of subgradient computations. To achieve the best possible complexity bounds in terms of both subgradient computation and communication, the proposed DCS method (along with its analysis) are in fact more complicated than the aforementioned inexact primal–dual method in the following two aspects. Firstly, while in most inexact first-order methods one usually computes only one approximate solution of the subproblems, in the proposed DCS method we need to generate a pair of closely related approximate solutions $\mathbf{x}^k = (x_1^k, \ldots, x_m^k)$ and $\hat{\mathbf{x}}^k = (\hat{x}_1^k, \ldots, \hat{x}_m^k)$ to the subproblem in (3.3). Secondly, we need to modify the primal–dual method in a way such that one of these sequence (i.e.,$\{\hat{\mathbf{x}}^k\}$) will be used in the the extrapolation step in (3.1), while the other sequence $\{\mathbf{x}^k\}$ will act as the prox-center in $\mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x})$ (see (3.3)).

We formally describe our DCS method in Algorithm 1. An outer iteration of the DCS algorithm occurs whenever the index k in Algorithm 1 is incremented by 1. More specifically, each primal estimate x_i^0 is locally initialized from some arbitrary point in X_i, and x_i^{-1} and $\hat{x}_i^0$ are also set to be the same value. At each time step $k \geq 1$, each agent $i \in \mathcal{N}$ computes a local prediction $\tilde{x}_i^k$ using these three previous primal iterates (ref. (3.4)), and sends it to all of the nodes in its neighborhood, i.e., to all agents $j \in N_i$. In (3.5)–(3.6), each agent i then calculates the neighborhood disagreement v_i^k using the messages received from agents in N_i, and updates the dual subvector y_i^k. Then, another round of communication occurs in (3.7) when calculating w_i^k based on these updated dual variables. Therefore, each outer iteration k involves two communication rounds, one for the primal estimates and the other for the dual variables. Lastly, each agent i approximately solves the proximal projection subproblem (3.3), i.e.,

$$\operatorname{argmin}_{u\in U} \ \langle w, u\rangle + \phi(u) + \eta V(x, u) \tag{3.12}$$

with $u = x_i$, $U = X_i$, $w = w_i^k$, $\phi = f_i$, $\eta = \eta_k$ and $V = V_i$, by calling the CS procedure for $T = T_k$ iterations in (3.8).

Each iteration performed by the CS procedure, referred to as an inner iteration of the DCS method, is equivalent to a subgradient descent step applied to (3.12). More specifically, each inner iteration consists of the computation of the subgradient $\phi'(u^{t-1})$ in (3.9) and the solution of the projection subproblem in (3.10). Note that the objective function of (3.10) consists of two parts: (1) the inner product of u and the summation of w and the current subgradient $\phi'(u^{t-1})$; and (2) two Bregman distances requiring that the new iterate lies near x and u^{t-1}. By using the definition of Bregman distance, we can see that (3.10) is equivalent to

Algorithm 1 DCS from agent i's perspective

```
Let $x_i^0 = x_i^{-1} = \hat{x}_i^0 \in X_i$, $y_i^0 \in \mathbb{R}^d$ for $i \in [m]$ and the nonnegative parameters $\{\alpha_k\}$, $\{\tau_k\}$, $\{\eta_k\}$ and $\{T_k\}$ be given.
for $k = 1, \ldots, N$ do
    Update $z_i^k = (\hat{x}_i^k, y_i^k)$ according to
        $\tilde{x}_i^k = \alpha_k(\hat{x}_i^{k-1} - x_i^{k-2}) + x_i^{k-1},$    (3.4)
        $v_i^k = \sum_{j \in N_i} L_{ij}\tilde{x}_j^k,$    (3.5)
        $y_i^k = \mathrm{argmin}_{y_i \in \mathbb{R}^d} \langle -v_i^k, y_i\rangle + \frac{\tau_k}{2}\|y_i - y_i^{k-1}\|^2 = y_i^{k-1} + \frac{1}{\tau_k}v_i^k,$    (3.6)
        $w_i^k = \sum_{j \in N_i} L_{ij}y_j^k,$    (3.7)
        $(x_i^k, \hat{x}_i^k) = \mathrm{CS}(f_i, X_i, V_i, T_k, \eta_k, w_i^k, x_i^{k-1}).$    (3.8)
end for
return $z_i^N = \left(\sum_{k=1}^N \theta_k\right)^{-1} \sum_{k=1}^N \theta_k z_i^k$

The CS (Communication-Sliding) procedure called at (3.8) is stated as follows.
procedure: $(x, \hat{x}) = \mathrm{CS}(\phi, U, V, T, \eta, w, x)$
Let $u^0 = \hat{u}^0 = x$ and the parameters $\{\beta_t\}$ and $\{\lambda_t\}$ be given.
for $t = 1, \ldots, T$ do
        $h^{t-1} = \phi'(u^{t-1}) \in \partial\phi(u^{t-1}),$    (3.9)
        $u^t = \mathrm{argmin}_{u \in U}\left[\langle w + h^{t-1}, u\rangle + \eta V(x, u) + \eta\beta_t V(u^{t-1}, u)\right].$    (3.10)
end for
Set
        $\hat{u}^T := \left(\sum_{t=1}^T \lambda_t\right)^{-1} \sum_{t=1}^T \lambda_t u^t.$    (3.11)
Set $x = u^T$ and $\hat{x} = \hat{u}^T$.
end procedure
```

$$u^t = \mathrm{argmin}_{u \in U}\left[\langle w + h^{t-1} - \eta\nabla\omega(x) - \eta\beta_t\nabla\omega(u^{t-1}), u\rangle + \eta(1 + \beta_t)\omega(u)\right].$$

Similar to mirror-descent type methods, we assume that this problem is easy to solve. Also observe that the same dual information $w = w_i^k$ (see (3.7)) has been used throughout the $T = T_k$ iterations of the CS procedure, and hence no additional communication is required within the procedure, which explains the name of the DCS method.

Observe that the DCS method, in spirit, has been inspired by our recent work on gradient sliding [28]. However, the gradient sliding method in [28] focuses on how to save gradient evaluations for solving certain structured convex optimization problems, rather than how to save communication rounds (or matrix-vector products) for decentralized optimization, and its algorithmic scheme is also quite different from the DCS method. It should also be note that the description of the algorithm is only conceptual at this moment since we have not specified the parameters $\{\alpha_k\}$, $\{\eta_k\}$, $\{\tau_k\}$, $\{T_k\}$, $\{\beta_t\}$ and $\{\lambda_t\}$ yet. We will later instantiate this generic algorithm when we state its convergence properties.

3.2 Convergence of DCS on general convex functions

We now establish the main convergence properties of the DCS algorithm. More specifically, we provide in Lemma 1 an estimate on the gap function defined in (2.5) together with stepsize policies which work for the general nonsmooth convex case with $\mu = 0$ (cf. (1.2)). The proof of this lemma can be found in Sect. 5.

Lemma 1 *Let the iterates* $(\hat{\mathbf{x}}^k, \mathbf{y}^k)$, $k = 1, \ldots, N$ *be generated by Algorithm* 1 *and* $\hat{\mathbf{z}}^N$ *be defined as* $\hat{\mathbf{z}}^N := \left(\sum_{k=1}^N \theta_k\right)^{-1} \sum_{k=1}^N \theta_k (\hat{\mathbf{x}}^k, \mathbf{y}^k)$. *If the objective* f_i, $i = 1, \ldots, m$, *are general nonsmooth convex functions, i.e.,* $\mu = 0$ *and* $M > 0$, *let the parameters* $\{\alpha_k\}$, $\{\theta_k\}$, $\{\eta_k\}$, $\{\tau_k\}$ *and* $\{T_k\}$ *in Algorithm* 1 *satisfy*

$$\theta_k \tfrac{(T_k+1)(T_k+2)\eta_k}{T_k(T_k+3)} \leq \theta_{k-1} \tfrac{(T_{k-1}+1)(T_{k-1}+2)\eta_{k-1}}{T_{k-1}(T_{k-1}+3)}, \quad k = 2, \ldots, N, \tag{3.13}$$

$$\alpha_k \theta_k = \theta_{k-1}, \quad k = 2, \ldots, N, \tag{3.14}$$

$$\theta_k \tau_k = \theta_1 \tau_1, \quad k = 2, \ldots, N, \tag{3.15}$$

$$\alpha_k \|\mathbf{L}\|^2 \leq \eta_{k-1} \tau_k, \quad k = 2, \ldots, N, \tag{3.16}$$

$$\theta_N \|\mathbf{L}\|^2 \leq \theta_1 \tau_1 \eta_N, \tag{3.17}$$

and the parameters $\{\lambda_t\}$ *and* $\{\beta_t\}$ *in the CS procedure of Algorithm 1 be set to*

$$\lambda_t = t + 1, \quad \beta_t = \tfrac{t}{2}, \quad \forall t \geq 1. \tag{3.18}$$

Then, we have for all $\mathbf{z} := (\mathbf{x}, \mathbf{y}) \in \mathbf{X} \times \mathbb{R}^{md}$,

$$Q(\hat{\mathbf{z}}^N; \mathbf{z}) \leq \left(\sum_{k=1}^N \theta_k\right)^{-1} \Bigg[\tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) + \tfrac{\theta_1\tau_1}{2} \|\mathbf{y}^0\|^2 + \langle \hat{\mathbf{s}}, \mathbf{y} \rangle + \sum_{k=1}^N \tfrac{4mM^2\theta_k}{(T_k+3)\eta_k} \Bigg], \tag{3.19}$$

where $\hat{\mathbf{s}} := \theta_N \mathbf{L}(\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}) + \theta_1\tau_1(\mathbf{y}^N - \mathbf{y}^0)$ *and* Q *is defined in* (2.5). *Furthermore, for any saddle point* $(\mathbf{x}^*, \mathbf{y}^*)$ *of* (2.4), *we have*

$$\tfrac{\theta_N}{2}\left(1 - \tfrac{\|\mathbf{L}\|^2}{\eta_N \tau_N}\right) \max\left\{\eta_N \|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2, \tau_N \|\mathbf{y}^* - \mathbf{y}^N\|^2\right\} \leq \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{\theta_1\tau_1}{2} \|\mathbf{y}^* - \mathbf{y}^0\|^2 + \sum_{k=1}^N \tfrac{4mM^2\theta_k}{\eta_k(T_k+3)}. \tag{3.20}$$

In the following theorem, we provide a specific selection of $\{\alpha_k\}$, $\{\theta_k\}$, $\{\eta_k\}$, $\{\tau_k\}$ and $\{T_k\}$ satisfying (3.13)–(3.17). Using Lemma 1 and Proposition 1, we also establish the complexity of the DCS method for computing an (ϵ, δ)-solution of problem (2.3) when the objective functions are general convex.

Theorem 1 *Let* $\mathbf{x}^*$ *be an optimal solution of* (2.3)*, the parameters* $\{\lambda_t\}$ *and* $\{\beta_t\}$ *in the CS procedure of Algorithm 1 be set to* (3.18)*, and suppose that* $\{\alpha_k\}$*,* $\{\theta_k\}$*,* $\{\eta_k\}$*,* $\{\tau_k\}$ *and* $\{T_k\}$ *are set to*

$$\alpha_k = \theta_k = 1,\ \eta_k = 2\|\mathbf{L}\|,\ \tau_k = \|\mathbf{L}\|,\ \text{and } T_k = \left\lceil \tfrac{mM^2N}{\|\mathbf{L}\|^2\tilde{D}} \right\rceil, \quad \forall k = 1, \ldots, N, \tag{3.21}$$

for some $\tilde{D} > 0$*. Then, for any* $N \geq 1$*, we have*

$$F(\hat{\mathbf{x}}^N) - F(\mathbf{x}^*) \leq \tfrac{\|\mathbf{L}\|}{N}\left[3\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{1}{2}\|\mathbf{y}^0\|^2 + 2\tilde{D}\right] \tag{3.22}$$

and

$$\|\mathbf{L}\hat{\mathbf{x}}^N\| \leq \tfrac{\|\mathbf{L}\|}{N}\left[3\sqrt{6\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + 4\tilde{D}} + 4\|\mathbf{y}^* - \mathbf{y}^0\|\right], \tag{3.23}$$

where $\hat{\mathbf{x}}^N = \frac{1}{N}\sum_{k=1}^N \hat{\mathbf{x}}^k$*, and* y^* *is an arbitrary dual optimal solution.*

Proof It is easy to check that (3.21) satisfies conditions (3.13)–(3.17). Particularly,

$$\tfrac{(T_1+1)(T_1+2)}{T_1(T_1+3)} = 1 + \tfrac{2}{T_1^2+3T_1} \leq \tfrac{3}{2}.$$

Therefore, by plugging in these values to (3.19), we have

$$Q(\hat{\mathbf{z}}^N; \mathbf{x}^*, \mathbf{y}) \leq \tfrac{\|\mathbf{L}\|}{N}\left[3\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{1}{2}\|\mathbf{y}^0\|^2 + 2\tilde{D}\right] + \tfrac{1}{N}\langle \hat{\mathbf{s}}, \mathbf{y}\rangle. \tag{3.24}$$

Letting $\hat{\mathbf{s}}^N = \frac{1}{N}\hat{\mathbf{s}}$, then from (3.20), we have

$$\begin{aligned}\|\hat{\mathbf{s}}^N\| &\leq \tfrac{\|\mathbf{L}\|}{N}\left[\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\| + \|\mathbf{y}^N - \mathbf{y}^*\| + \|\mathbf{y}^* - \mathbf{y}^0\|\right] \\ &\leq \tfrac{\|\mathbf{L}\|}{N}\left[3\sqrt{6\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \|\mathbf{y}^* - \mathbf{y}^0\|^2 + 4\tilde{D}} + \|\mathbf{y}^* - \mathbf{y}^0\|\right].\end{aligned}$$

Furthermore, by (3.24), we have

$$g(\hat{\mathbf{s}}^N, \hat{\mathbf{z}}^N) \leq \tfrac{\|\mathbf{L}\|}{N}\left[3\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{1}{2}\|\mathbf{y}^0\|^2 + 2\tilde{D}\right].$$

Applying Proposition 1 to the above two inequalities, the results in (3.22) and (3.23) follow immediately. □

We now make some remarks about the results obtained in Theorem 1. Firstly, even though one can choose any $\tilde{D} > 0$ (e.g., $\tilde{D} = 1$) in (3.21), the best selection of $\tilde{D}$ would be $\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*)$ so that the first and third terms in (3.24) are about the same order. In practice, if there exists an estimate $\mathcal{D}_{\mathbf{X}} > 0$ s.t.

$$\mathbf{V}(\mathbf{x}_1, \mathbf{x}_2) \leq \mathcal{D}_{\mathbf{X}}^2, \quad \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathbf{X}, \tag{3.25}$$

then we can set $\tilde{D} = \mathcal{D}_{\mathbf{X}}^2$.

Secondly, the complexity of the DCS method directly follows from (3.22) and (3.23). For simplicity, let us assume that X is bounded, $\tilde{D} = \mathcal{D}_{\mathbf{X}}^2$ and $\mathbf{y}^0 = \mathbf{0}$. We can see that the total number of inter-node communication rounds and intra-node subgradient evaluations required by each agent for finding an (ϵ, δ)-solution of (2.3) can be bounded by

$$\mathcal{O}\left\{\|\mathbf{L}\| \max\left(\frac{\mathcal{D}_{\mathbf{X}}^2}{\epsilon}, \frac{\mathcal{D}_{\mathbf{X}}+\|\mathbf{y}^*\|}{\delta}\right)\right\} \text{ and } \mathcal{O}\left\{mM^2 \max\left(\frac{\mathcal{D}_{\mathbf{X}}^2}{\epsilon^2}, \frac{\mathcal{D}_{\mathbf{X}}^2+\|\mathbf{y}^*\|^2}{\mathcal{D}_{\mathbf{X}}^2\delta^2}\right)\right\}, \tag{3.26}$$

respectively. In particular, if ϵ and δ satisfy

$$\frac{\epsilon}{\delta} \leq \frac{\mathcal{D}_{\mathbf{X}}^2}{\mathcal{D}_{\mathbf{X}}+\|\mathbf{y}^*\|}, \tag{3.27}$$

then the previous two complexity bounds in (3.26), respectively, reduce to

$$\mathcal{O}\left\{\frac{\|\mathbf{L}\|\mathcal{D}_{\mathbf{X}}^2}{\epsilon}\right\} \text{ and } \mathcal{O}\left\{\frac{mM^2\mathcal{D}_{\mathbf{X}}^2}{\epsilon^2}\right\}. \tag{3.28}$$

Thirdly, it is interesting to compare DCS with the centralized mirror descent method [46] applied to (1.1). In the worst case, the Lipschitz constant of f in (1.1) can be bounded by $M_f \leq mM$, and each iteration of the method will incur m subgradient evaluations. Hence, the total number of subgradient evaluations performed by the mirror descent method for finding an ϵ-solution of (1.1), i.e., a point $\bar{x} \in X$ such that $f(\bar{x}) - f^* \leq \epsilon$, can be bounded by

$$\mathcal{O}\left\{\frac{m^3M^2\mathcal{D}_X^2}{\epsilon^2}\right\}, \tag{3.29}$$

where $\mathcal{D}_X^2$ characterizes the diameter of X, i.e., $\mathcal{D}_X^2 := \max_{x_1,x_2\in X} V(x_1, x_2)$. Noting that $\mathcal{D}_X^2/\mathcal{D}_{\mathbf{X}}^2 = \mathcal{O}(1/m)$, and that the second bound in (3.28) states only the number of subgradient evaluations for each agent in the DCS method, we conclude that the total number of subgradient evaluations performed by DCS is comparable to the classic mirror descent method as long as (3.27) holds and hence not improvable in general.

Finally, observe that the parameter setting (3.21) requires the knowledge of the norm of Laplacian matrix $\mathbf{L}$, i.e., $\|\mathbf{L}\| = \max_{\|x\|\leq 1}\{\|\mathbf{L}\mathbf{x}\|_2\}$. If we use l_2-norm for the primal space, $\|\mathbf{L}\|$ will be the maximum eigenvalue of L. We can estimate it using power iteration method or simply bound it by the maximum degree of the graph. If we use l_1-norm in the primal space, then $\|\mathbf{L}\|$ will be the $L_{1,2}$-norm for $\|\mathbf{L}\|$, i.e., $\|\mathbf{L}\| = \|\mathbf{L}\|_{1,2} = \left(\sum_{i=1}^{md} \|\mathbf{L}_i\|_1^2\right)^{1/2} = 2\sqrt{d\sum_{j=1}^{m} \deg_j^2}$, where $\mathbf{L}_i$'s denote the row vectors of $\mathbf{L}$ and $\deg_j$ denotes the degree of node j. The estimation of $\|\mathbf{L}\|$ will involve a few rounds of communication, however, these initial setup costs are independent of the target accuracy ϵ of the solution. It should also be noted that the number of inner iterations T_k given in (3.21) is fixed as a constant in order to achieve the best complexity bounds. In practice, it is reasonable to choose T_k dynamically so that a smaller number of inner iterations will be performed in the first few outer iterations. One simple strategy would be to set

$$T_k = \min\left(ck, \left\lceil \frac{mM^2N}{\|\mathbf{L}\|^2\tilde{D}} \right\rceil\right)$$

for some constant $c > 0$. While theoretically such a selection of T_k will result in slightly worse complexity bounds (up to an $\mathcal{O}(\log(1/\epsilon))$ factor) in terms of subgradient computations and communication rounds, it may improve the practical performance of the DCS method especially in the beginning of the execution of this method.

3.3 Boundedness of $\|\mathbf{y}^*\|$

In this subsection, we will provide a bound on the optimal dual multiplier $\mathbf{y}^*$. By doing so, we show that the complexity of DCS algorithm (as well as the stochastic DCS algorithm in Sect. 4) only depends on the parameters for the primal problem along with the smallest nonzero eigenvalue of $\mathbf{L}$ and the initial point $\mathbf{y}^0$, even though these algorithms are intrinsically primal–dual type methods.

Theorem 2 *Suppose that f_i's are Lipschitz continuous, i.e., the subgradients of f_i are bounded by a constant M_f w.r.t. $\|\cdot\|_2$. Let $\mathbf{x}^*$ be an optimal solution of* (2.3)*. Then there exists an optimal dual multiplier $\mathbf{y}^*$ for* (2.4) *s.t.*

$$\|\mathbf{y}^*\|_2 \le \frac{\sqrt{m}M_f}{\tilde{\sigma}_{min}(\mathbf{L})}, \tag{3.30}$$

where $\tilde{\sigma}_{min}(\mathbf{L})$ denotes the smallest nonzero eigenvalue of $\mathbf{L}$.

Proof Since we only relax the linear constraints in problem (2.3) to obtain the Lagrange dual problem (2.4), it follows from the strong Lagrange duality and the existence of $\mathbf{x}^*$ to (2.3) that an optimal dual multiplier $\mathbf{y}^*$ for problem (2.4) must exist. It is clear that

$$\mathbf{y}^* = \mathbf{y}_N^* + \mathbf{y}_C^*,$$

where $\mathbf{y}_N^*$ and $\mathbf{y}_C^*$ denote the projections of $\mathbf{y}^*$ over the null space and the column space of $\mathbf{L}^T$, respectively.

We consider two cases. Case 1) $\mathbf{y}_C^* = \mathbf{0}$. Since $\mathbf{y}_N^*$ belongs to the null space of $\mathbf{L}^T$, $\mathbf{L}^T\mathbf{y}^* = \mathbf{L}^T\mathbf{y}_N^* = \mathbf{0}$, which implies that for any $c \in \mathbb{R}$, $c\mathbf{y}^*$ is also an optimal dual multiplier of (2.4). Therefore, (3.30) clearly holds, because we can scale $\mathbf{y}^*$ to an arbitrary small vector.

Case 2) $\mathbf{y}_C^* \neq \mathbf{0}$. Using the fact that $\mathbf{L}^T\mathbf{y}^* = \mathbf{L}^T\mathbf{y}_C^*$ and the definition of a saddle point of (2.4), we conclude that $\mathbf{y}_C^*$ is also an optimal dual multiplier of (2.4). Since $\mathbf{y}_C^*$ in the column space of $\mathbf{L}$, we have

$$\|\mathbf{L}^T\mathbf{y}_C^*\|_2^2 = (\mathbf{y}_C^*)^T\mathbf{L}\mathbf{L}^T\mathbf{y}_C^* = (\mathbf{y}_C^*)^T\mathbf{U}^T\mathbf{U}\mathbf{y}_C^* \geq \tilde{\lambda}_{\min}(\mathbf{L}\mathbf{L}^T)\|\mathbf{U}\mathbf{y}_C^*\|_2^2 = \tilde{\sigma}_{\min}^2(\mathbf{L})\|\mathbf{y}_C^*\|_2^2,$$

where $\mathbf{U}$ is an orthonormal matrix whose rows consist of the eigenvectors of $\mathbf{L}\mathbf{L}^T$, is the diagonal matrix whose diagonal elements are the corresponding eigenvalues, $\tilde{\lambda}_{min}(\mathbf{L}\mathbf{L}^T)$ denotes the smallest nonzero eigenvalue of $\mathbf{L}\mathbf{L}^T$, and $\tilde{\sigma}_{min}(\mathbf{L})$ denotes the smallest nonzero eigenvalue of $\mathbf{L}$. In particular,

$$\|\mathbf{y}_C^*\|_2 \leq \frac{\|\mathbf{L}^T\mathbf{y}_C^*\|_2}{\tilde{\sigma}_{\min}(\mathbf{L})}. \tag{3.31}$$

Moreover, if we denote the saddle point problem defined in (2.4) as follows:

$$\mathcal{L}(\mathbf{x}, \mathbf{y}) := F(\mathbf{x}) + \langle \mathbf{L}\mathbf{x}, \mathbf{y} \rangle.$$

By the definition of a saddle point of (2.4), we have $\mathcal{L}(\mathbf{x}^*, \mathbf{y}_C^*) \leq \mathcal{L}(\mathbf{x}, \mathbf{y}_C^*)$, i.e.,

$$F(\mathbf{x}^*) - F(\mathbf{x}) \leq \langle -\mathbf{L}^T\mathbf{y}_C^*, \mathbf{x} - \mathbf{x}^* \rangle.$$

Hence, from the definition of subgradients, we conclude that $-\mathbf{L}^T\mathbf{y}_C^* \in \partial F(\mathbf{x}^*)$, which together with the fact that f_i's are Lipschitz continuous implies that

$$\|\mathbf{L}^T\mathbf{y}_C^*\|_2 = \|(f_1'(x_1^*), f_2'(x_2^*), \ldots, f_m'(x_m^*))\|_2 \leq \sqrt{m}M_f.$$

Our result in (3.30) follows immediately from the above relation, (3.31) and the fact that $\mathbf{y}_C^*$ is also an optimal dual multiplier of (2.4). □

Observe that our bound for the dual multiplier $\mathbf{y}^*$ in (3.30) contains only the primal information. Given an initial dual multiplier $\mathbf{y}^0$, this result can be used to provide an upper bound on $\|\mathbf{y}^0 - \mathbf{y}^*\|$ in Theorems 1–5 throughout this paper. Note also that we can assume $\mathbf{y}^0 = 0$ to simplify these complexity bounds.

3.4 Convergence of DCS on strongly convex functions

In this subsection, we assume that the objective functions f_i's are strongly convex (i.e., $\mu > 0$ (1.2)). In order to take advantage of the strong convexity of the objective functions, we assume that the prox-functions $V_i(\cdot, \cdot)$, $i = 1, \ldots, m$, (cf. (2.10)) are growing quadratically with the *quadratic growth constant* $\mathcal{C}$, i.e., there exists a constant $\mathcal{C} > 0$ such that

$$V_i(x_i, u_i) \leq \tfrac{\mathcal{C}}{2}\|x_i - u_i\|_{X_i}^2, \quad \forall x_i, u_i \in X_i,\ i = 1, \ldots, m. \tag{3.32}$$

By (2.11), we must have $\mathcal{C} \geq 1$.

We next provide in Lemma 2 an estimate on the gap function defined in (2.5) together with stepsize policies which work for the strongly convex case. The proof of this lemma can be found in Sect. 5.

Lemma 2 *Let the iterates* $(\hat{\mathbf{x}}^k, \mathbf{y}^k)$, $k = 1, \ldots, N$ *be generated by Algorithm 1 and* $\hat{\mathbf{z}}^N$ *be defined as* $\hat{\mathbf{z}}^N := \left(\sum_{k=1}^N \theta_k\right)^{-1} \sum_{k=1}^N \theta_k (\hat{\mathbf{x}}^k, \mathbf{y}^k)$. *If the objective* f_i, $i = 1, \ldots, m$ *are strongly convex functions, i.e.,* $\mu, M > 0$, *let the parameters* $\{\alpha_k\}$, $\{\theta_k\}$, $\{\eta_k\}$ *and* $\{\tau_k\}$ *in Algorithm 1 satisfy* (3.14)–(3.17) *and*

$$\theta_k \eta_k \leq \theta_{k-1}(\mu/\mathcal{C} + \eta_{k-1}), \quad k = 2, \ldots, N, \tag{3.33}$$

and the parameters $\{\lambda_t\}$ *and* $\{\beta_t\}$ *in the CS procedure of Algorithm 1 be set to*

$$\lambda_t = t, \quad \beta_t^{(k)} = \tfrac{(t+1)\mu}{2\eta_k \mathcal{C}} + \tfrac{t-1}{2}, \quad \forall t \geq 1. \tag{3.34}$$

Then, we have for all $\mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$

$$Q(\hat{\mathbf{z}}^N; \mathbf{z}) \leq \left(\sum_{k=1}^N \theta_k\right)^{-1} \Bigg[\theta_1 \eta_1 \mathbf{V}(\mathbf{x}^0, \mathbf{x}) + \tfrac{\theta_1 \tau_1}{2} \|\mathbf{y}^0\|^2 + \langle \hat{\mathbf{s}}, \mathbf{y} \rangle + \sum_{k=1}^N \sum_{t=1}^{T_k} \tfrac{2mM^2\theta_k}{T_k(T_k+1)} \tfrac{t}{(t+1)\mu/\mathcal{C} + (t-1)\eta_k}\Bigg], \tag{3.35}$$

where $\hat{\mathbf{s}} := \theta_N \mathbf{L}(\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}) + \theta_1 \tau_1 (\mathbf{y}^N - \mathbf{y}^0)$ *and Q is defined in* (2.5). *Furthermore, for any saddle point* $(\mathbf{x}^*, \mathbf{y}^*)$ *of* (2.4), *we have*

$$\begin{aligned} &\tfrac{\theta_N}{2}\left(1 - \tfrac{\|\mathbf{L}\|^2}{\eta_N \tau_N}\right) \max\left\{\eta_N \|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2, \tau_N \|\mathbf{y}^* - \mathbf{y}^N\|^2\right\} \\ &\quad \leq \theta_1 \eta_1 \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{\theta_1 \tau_1}{2} \|\mathbf{y}^* - \mathbf{y}^0\|^2 \\ &\quad + \sum_{k=1}^N \sum_{t=1}^{T_k} \tfrac{2mM^2\theta_k}{T_k(T_k+1)} \tfrac{t}{(t+1)\mu/\mathcal{C} + (t-1)\eta_k}. \end{aligned} \tag{3.36}$$

In the following theorem, we provide a specific selection of $\{\alpha_k\}$, $\{\theta_k\}$, $\{\eta_k\}$, $\{\tau_k\}$ and $\{T_k\}$ satisfying (3.14)–(3.17) and (3.33). Also, by using Lemma 2 and Proposition 1, we establish the complexity of the DCS method for computing an (ϵ, δ)-solution of problem (2.3) when the objective functions are strongly convex. The choice of variable stepsizes rather than using constant stepsizes will accelerate its convergence rate.

Theorem 3 *Let* $\mathbf{x}^*$ *be an optimal solution of* (2.3), *the parameters* $\{\lambda_t\}$ *and* $\{\beta_t\}$ *in the CS procedure of Algorithm 1 be set to* (3.34) *and suppose that* $\{\alpha_k\}$, $\{\theta_k\}$, $\{\eta_k\}$, $\{\tau_k\}$ *and* $\{T_k\}$ *are set to*

$$\alpha_k = \tfrac{k}{k+1}, \ \theta_k = k+1, \ \eta_k = \tfrac{k\mu}{2\mathcal{C}}, \ \tau_k = \tfrac{4\|\mathbf{L}\|^2 \mathcal{C}}{(k+1)\mu}, \quad \textit{and}$$

$$T_k = \left\lceil \sqrt{\tfrac{2m}{\tilde{D}}} \tfrac{\mathcal{C}MN}{\mu} \max\left\{ \sqrt{\tfrac{2m}{\tilde{D}}} \tfrac{4\mathcal{C}M}{\mu}, 1 \right\} \right\rceil, \tag{3.37}$$

$\forall k = 1, \ldots, N$, for some $\tilde{D} > 0$. Then, for any $N \geq 2$, we have

$$F(\hat{\mathbf{x}}^N) - F(\mathbf{x}^*) \leq \tfrac{2}{N(N+3)} \left[\tfrac{\mu}{\mathcal{C}} \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{2\|\mathbf{L}\|^2 \mathcal{C}}{\mu} \|\mathbf{y}^0\|^2 + \tfrac{2\mu\tilde{D}}{\mathcal{C}} \right], \tag{3.38}$$

and

$$\|\mathbf{L}\hat{\mathbf{x}}^N\| \leq \tfrac{8\|\mathbf{L}\|}{N(N+3)} \left[3\sqrt{2\tilde{D} + \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*)} + \tfrac{7\|\mathbf{L}\|\mathcal{C}}{\mu} \|\mathbf{y}^* - \mathbf{y}^0\| \right], \tag{3.39}$$

where $\hat{\mathbf{x}}^N = \frac{2}{N(N+3)} \sum_{k=1}^{N} (k+1)\hat{\mathbf{x}}^k$, and y^* is an arbitrary dual optimal solution.

Proof It is easy to check that (3.37) satisfies conditions (3.14)–(3.17) and (3.33). Moreover, we have

$$\begin{aligned}
&\sum_{k=1}^{N} \sum_{t=1}^{T_k} \tfrac{2mM^2\theta_k}{T_k(T_k+1)} \tfrac{t}{(t+1)\mu/\mathcal{C} + (t-1)\eta_k} \\
&\quad = \sum_{k=1}^{N} \tfrac{2mM^2\theta_k\mathcal{C}}{T_k(T_k+1)\mu} \sum_{t=1}^{T_k} \tfrac{2t}{2(t+1)+(t-1)k} \\
&\quad \leq \sum_{k=1}^{N} \tfrac{2mM^2\theta_k\mathcal{C}}{T_k(T_k+1)\mu} \left(\tfrac{1}{2} + \sum_{t=2}^{T_k} \tfrac{2t}{(t-1)(k+1)} \right) \\
&\quad \leq \sum_{k=1}^{N} \tfrac{mM^2\mathcal{C}(k+1)}{T_k(T_k+1)\mu} + \sum_{k=1}^{N} \tfrac{8mM^2\mathcal{C}(T_k-1)}{T_k(T_k+1)\mu} \leq \tfrac{2\mu\tilde{D}}{\mathcal{C}}.
\end{aligned}$$

Therefore, by plugging in these values to (3.35), we have

$$Q(\hat{\mathbf{z}}^N; \mathbf{x}^*, \mathbf{y}) \leq \tfrac{2}{N(N+3)} \left[\tfrac{\mu}{\mathcal{C}} \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{2\|\mathbf{L}\|^2\mathcal{C}}{\mu} \|\mathbf{y}^0\|^2 + \tfrac{2\mu\tilde{D}}{\mathcal{C}} \right] + \tfrac{2}{N(N+3)} \langle \hat{\mathbf{s}}, \mathbf{y} \rangle. \tag{3.40}$$

Furthermore, from (3.36), we have for $N \geq 2$

$$\begin{aligned}
\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2 &\leq \tfrac{8\mathcal{C}}{\mu(N+1)(N-1)} \left[\tfrac{\mu}{\mathcal{C}} \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{2\|\mathbf{L}\|^2\mathcal{C}}{\mu} \|\mathbf{y}^0 - \mathbf{y}^*\|^2 + \tfrac{2\mu\tilde{D}}{\mathcal{C}} \right], \\
\|\mathbf{y}^* - \mathbf{y}^N\|^2 &\leq \tfrac{N\mu}{(N-1)\|\mathbf{L}\|^2\mathcal{C}} \left[\tfrac{\mu}{\mathcal{C}} \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{2\|\mathbf{L}\|^2\mathcal{C}}{\mu} \|\mathbf{y}^0 - \mathbf{y}^*\|^2 + \tfrac{2\mu\tilde{D}}{\mathcal{C}} \right].
\end{aligned} \tag{3.41}$$

Let $\mathbf{s}^N := \frac{2}{N(N+3)} \hat{\mathbf{s}}$, then by using (3.41), we have for $N \geq 2$

$$\|\mathbf{s}^N\| \leq \tfrac{2}{N(N+3)} \left[(N+1)\|\mathbf{L}\| \|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\| + \tfrac{4\|\mathbf{L}\|^2\mathcal{C}}{\mu} \|\mathbf{y}^N - \mathbf{y}^*\| + \tfrac{4\|\mathbf{L}\|^2\mathcal{C}}{\mu} \|\mathbf{y}^* - \mathbf{y}^0\| \right]$$

$$
\begin{aligned}
&\le \tfrac{8\|\mathbf{L}\|}{N(N+3)}\left[3\sqrt{2\tilde{D}+\mathbf{V}(\mathbf{x}^0,\mathbf{x}^*)+\tfrac{2\|\mathbf{L}\|^2\mathcal{C}^2}{\mu^2}\|\mathbf{y}^0-\mathbf{y}^*\|^2}+\tfrac{\|\mathbf{L}\|\mathcal{C}}{\mu}\|\mathbf{y}^*-\mathbf{y}^0\|\right]\\
&\le \tfrac{8\|\mathbf{L}\|}{N(N+3)}\left[3\sqrt{2\tilde{D}+\mathbf{V}(\mathbf{x}^0,\mathbf{x}^*)}+\tfrac{7\|\mathbf{L}\|\mathcal{C}}{\mu}\|\mathbf{y}^*-\mathbf{y}^0\|\right].
\end{aligned}
$$

From (3.40), we further have

$$
g(\hat{\mathbf{s}}^N,\hat{\mathbf{z}}^N)\le \tfrac{2}{N(N+3)}\left[\tfrac{\mu}{\mathcal{C}}\mathbf{V}(\mathbf{x}^0,\mathbf{x}^*)+\tfrac{2\|\mathbf{L}\|^2\mathcal{C}}{\mu}\|\mathbf{y}^0\|^2+\tfrac{2\mu\tilde{D}}{\mathcal{C}}\right].
$$

Applying Proposition 1 to the above two inequalities, the results in (3.38) and (3.39) follow immediately. □

We now make some remarks about the results obtained in Theorem 3. Firstly, similar to the general convex case, the best choice for $\tilde{D}$ (cf. (3.37)) would be $\mathbf{V}(\mathbf{x}^0,\mathbf{x}^*)$ so that the first and the third terms in (3.40) are about the same order. If there exists an estimate $\mathcal{D}_{\mathbf{X}}>0$ satisfying (3.25), we can set $\tilde{D}=\mathcal{D}_{\mathbf{X}}^2$.

Secondly, the complexity of the DCS method for solving strongly convex problems follows from (3.38) and (3.39). For simplicity, let us assume that X is bounded, $\tilde{D}=\mathcal{D}_{\mathbf{X}}^2$ and $\mathbf{y}^0=\mathbf{0}$. We can see that the total number of inter-node communication rounds and intra-node subgradient evaluations performed by each agent for finding an (ϵ,δ)-solution of (2.3) can be bounded by

$$
\begin{aligned}
&\mathcal{O}\left\{\max\left(\sqrt{\tfrac{\mu\mathcal{D}_{\mathbf{X}}^2}{\mathcal{C}\epsilon}},\sqrt{\tfrac{\|\mathbf{L}\|}{\delta}\left(\mathcal{D}_{\mathbf{X}}+\tfrac{\mathcal{C}\|\mathbf{L}\|\|\mathbf{y}^*\|}{\mu}\right)}\right)\right\}\ \text{and}\\
&\mathcal{O}\left\{\tfrac{mM^2\mathcal{C}}{\mu}\max\left(\tfrac{1}{\epsilon},\tfrac{\|\mathbf{L}\|\mathcal{C}}{\mu\delta}\left(\tfrac{1}{\mathcal{D}_{\mathbf{X}}}+\tfrac{\mathcal{C}\|\mathbf{L}\|\|\mathbf{y}^*\|}{\mathcal{D}_{\mathbf{X}}^2\mu}\right)\right)\right\},
\end{aligned}
\tag{3.42}
$$

respectively. In particular, if ϵ and δ satisfy

$$
\tfrac{\epsilon}{\delta}\le \frac{\mu^2\mathcal{D}_{\mathbf{X}}^2}{\|\mathbf{L}\|\mathcal{C}(\mu\mathcal{D}_{\mathbf{X}}+\mathcal{C}\|\mathbf{L}\|\|\mathbf{y}^*\|)},
\tag{3.43}
$$

then the complexity bounds in (3.42), respectively, reduce to

$$
\mathcal{O}\left\{\sqrt{\tfrac{\mu\mathcal{D}_{\mathbf{X}}^2}{\mathcal{C}\epsilon}}\right\}\ \text{and}\ \mathcal{O}\left\{\tfrac{mM^2\mathcal{C}}{\mu\epsilon}\right\}.
\tag{3.44}
$$

Thirdly, we compare DCS method with the centralized mirror descent method [46] applied to (1.1). In the worst case, the Lipschitz constant and strongly convex modulus of f in (1.1) can be bounded by $M_f\le mM$, and $\mu_f\ge m\mu$, respectively, and each iteration of the method will incur m subgradient evaluations. Therefore, the total number of subgradient evaluations performed by the mirror descent method for finding an ϵ-solution of (1.1), i.e., a point $\bar{x}\in X$ such that $f(\bar{x})-f^*\le\epsilon$, can be bounded by

$$\mathcal{O}\left\{\frac{m^2M^2\mathcal{C}}{\mu\epsilon}\right\}. \tag{3.45}$$

Observed that the second bound in (3.44) states only the number of subgradient evaluations for each agent in the DCS method, we conclude that the total number of subgradient evaluations performed by DCS is comparable to the classic mirror descent method as long as (3.43) holds and hence not improvable in general for the nonsmooth strongly convex case.

4 Stochastic decentralized communication sliding

In this section, we consider the stochastic case where only the noisy subgradient information of the functions f_i, $i = 1, \ldots, m$, is available or easier to compute. This situation happens when the function f_i's are given either in the form of expectation or as the summation of lots of components. This setting has attracted considerable interest in recent decades for its applications in a broad spectrum of disciplines including machine learning, signal processing, and operations research. We present a stochastic communication sliding method, namely the stochastic decentralized communication sliding (SDCS) method, and show that the similar complexity bounds as in Sect. 3 can still be obtained in expectation or with high probability.

4.1 The SDCS algorithm

The first-order information of the function f_i, $i = 1, \ldots, m$, can be accessed by a stochastic oracle (SO), which, given a point $u^t \in X$, outputs a vector $G_i(u^t, \xi_i^t)$ such that

$$\mathbb{E}[G_i(u^t, \xi_i^t)] = f_i'(u^t) \in \partial f_i(u^t), \tag{4.1}$$

$$\mathbb{E}[\|G_i(u^t, \xi_i^t) - f_i'(u^t)\|_*^2] \le \sigma^2, \tag{4.2}$$

where ξ_i^t is a random vector which models a source of uncertainty and is independent of the search point u^t, and the distribution $\mathbb{P}(\xi_i)$ is not known in advance. We call $G_i(u^t, \xi_i^t)$ a *stochastic subgradient* of f_i at u^t.

The SDCS method can be obtained by simply replacing the exact subgradients in the CS procedure of Algorithm 1 with the stochastic subgradients obtained from SO. This difference is described in Algorithm 2.

We add a few remarks about the SDCS algorithm. Firstly, as in DCS, no additional communications of the dual variables are required when the subgradient projection (4.4) is performed for T_k times in the inner loop. This is because the same w_i^k has been used throughout the T_k iterations of the Stochastic CS procedure. Secondly, the problem will reduce to the deterministic case if there is no stochastic noise associated with the SO, i.e., when $\sigma = 0$ in (4.2). Therefore, in Sect. 5, we investigate the convergence analysis for the stochastic case first and then simplify the analysis for the deterministic case by setting $\sigma = 0$.

Algorithm 2 SDCS

The projection step (3.9)–(3.10) in the CS procedure of Algorithm 1 is replaced by

$$h^{t-1} = H(u^{t-1}, \xi^{t-1}), \tag{4.3}$$

$$u^t = \operatorname{argmin}_{u\in U}\left[\langle w + h^{t-1}, u\rangle + \eta V(x,u) + \eta\beta_t V(u^{t-1}, u)\right], \tag{4.4}$$

where $H(u^{t-1}, \xi^{t-1})$ is a stochastic subgradient of ϕ at u^{t-1}.

4.2 Convergence of SDCS on general convex functions

We now establish the main convergence properties of the SDCS algorithm. More specifically, we provide in Lemma 3 an estimate on the gap function defined in (2.5) together with stepsize policies which work for the general convex case with $\mu = 0$ (cf. (1.2)). The proof of this lemma can be found in Sect. 5.

Lemma 3 *Let the iterates $(\hat{\mathbf{x}}^k, \mathbf{y}^k)$ for $k = 1, \ldots, N$ be generated by Algorithm 2, $\hat{\mathbf{z}}^N$ be defined as $\hat{\mathbf{z}}^N := \left(\sum_{k=1}^N \theta_k\right)^{-1} \sum_{k=1}^N \theta_k(\hat{\mathbf{x}}^k, \mathbf{y}^k)$, and Assumptions (4.1)–(4.2) hold. If the objective f_i, $i = 1, \ldots, m$, are general nonsmooth convex functions, i.e., $\mu = 0$ and $M > 0$, let the parameters $\{\alpha_k\}$, $\{\theta_k\}$, $\{\eta_k\}$, $\{\tau_k\}$ and $\{T_k\}$ in Algorithm 2 satisfy (3.13)–(3.17), and the parameters $\{\lambda_t\}$ and $\{\beta_t\}$ in the CS procedure of Algorithm 2 be set as (3.18). Then, for all $\mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$,*

$$\begin{aligned} Q(\hat{\mathbf{z}}^N; \mathbf{z}) \le \left(\sum_{k=1}^N \theta_k\right)^{-1} & \Bigg\{ \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2 + \langle \hat{\mathbf{s}}, \mathbf{y}\rangle \\ & + \sum_{k=1}^N \sum_{t=1}^{T_k} \sum_{i=1}^m \tfrac{2\theta_k}{T_k(T_k+3)} \\ & \left[(t+1)\langle \delta_i^{t-1,k}, x_i - u_i^{t-1}\rangle + \tfrac{4(M^2 + \|\delta_i^{t-1,k}\|_*^2)}{\eta_k}\right]\Bigg\}, \end{aligned} \tag{4.5}$$

where $\hat{\mathbf{s}} := \theta_N \mathbf{L}(\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}) + \theta_1\tau_1(\mathbf{y}^N - \mathbf{y}^0)$ and Q is defined in (2.5). Furthermore, for any saddle point $(\mathbf{x}^*, \mathbf{y}^*)$ of (2.4), we have

$$\begin{aligned} & \tfrac{\theta_N}{2}\left(1 - \tfrac{\|\mathbf{L}\|^2}{\eta_N\tau_N}\right)\max\{\eta_N\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2, \tau_N\|\mathbf{y}^* - \mathbf{y}^N\|^2\} \\ & \le \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)}\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^* - \mathbf{y}^0\|^2 \\ & + \sum_{k=1}^N \sum_{t=1}^{T_k} \sum_{i=1}^m \tfrac{2\theta_k}{T_k(T_k+3)} \\ & \left[(t+1)\langle \delta_i^{t-1,k}, x_i^* - u_i^{t-1}\rangle + \tfrac{4(M^2 + \|\delta_i^{t-1,k}\|_*^2)}{\eta_k}\right]. \end{aligned} \tag{4.6}$$

In the following theorem, we provide a specific selection of $\{\alpha_k\}$, $\{\theta_k\}$, $\{\eta_k\}$, $\{\tau_k\}$ and $\{T_k\}$ satisfying (3.13)–(3.17). Also, by using Lemma 3 and Proposition 1, we establish the complexity of the SDCS method for computing an (ϵ, δ)-solution of problem (2.3) in expectation when the objective functions are general convex.

Theorem 4 *Let* $\mathbf{x}^*$ *be an optimal solution of* (2.3)*, the parameters* $\{\lambda_t\}$ *and* $\{\beta_t\}$ *in the CS procedure of Algorithm* 2 *be set as* (3.18)*, and suppose that* $\{\alpha_k\}$*,* $\{\theta_k\}$*,* $\{\eta_k\}$*,* $\{\tau_k\}$ *and* $\{T_k\}$ *are set to*

$$\alpha_k = \theta_k = 1,\ \eta_k = 2\|\mathbf{L}\|,\ \tau_k = \|\mathbf{L}\|,\ \textit{and}\ T_k = \left\lceil \tfrac{m(M^2+\sigma^2)N}{\|\mathbf{L}\|^2\tilde{D}} \right\rceil, \quad \forall k = 1, \ldots, N, \tag{4.7}$$

for some $\tilde{D} > 0$*. Then, under Assumptions* (4.1) *and* (4.2)*, we have for any* $N \geq 1$

$$\mathbb{E}[F(\hat{\mathbf{x}}^k) - F(\mathbf{x}^*)] \leq \tfrac{\|\mathbf{L}\|}{N}\left[3\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{1}{2}\|\mathbf{y}^0\|^2 + 4\tilde{D}\right], \tag{4.8}$$

and

$$\mathbb{E}[\|\mathbf{L}\hat{\mathbf{x}}^N\|] \leq \tfrac{\|\mathbf{L}\|}{N}\left[3\sqrt{6\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + 8\tilde{D}} + 4\|\mathbf{y}^* - \mathbf{y}^0\|\right]. \tag{4.9}$$

where $\hat{\mathbf{x}}^N = \frac{1}{N}\sum_{k=1}^N \hat{\mathbf{x}}^k$*, and* y^* *is an arbitrary dual optimal solution.*

Proof It is easy to check that (4.7) satisfies conditions (3.13)–(3.17). Moreover, by (2.7), we can obtain

$$\begin{aligned} g(\hat{\mathbf{s}}^N, \hat{\mathbf{z}}^N) &= \max_{\mathbf{y}} Q(\hat{\mathbf{z}}^N; \mathbf{x}^*, \mathbf{y}) - \left(\sum_{k=1}^N \theta_k\right)^{-1} \langle \hat{\mathbf{s}}, \mathbf{y}\rangle \\ &\leq \left(\sum_{k=1}^N \theta_k\right)^{-1} \left\{ \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2 \right. \\ &\quad + \sum_{k=1}^N \sum_{t=1}^{T_k} \sum_{i=1}^m \tfrac{2\theta_k}{T_k(T_k+3)} \\ &\quad \left. \left[(t+1)\langle \delta_i^{t-1,k}, x_i^* - u_i^{t-1}\rangle + \tfrac{4(M^2+\|\delta_i^{t-1,k}\|_*^2)}{\eta_k}\right] \right\}, \end{aligned} \tag{4.10}$$

where $\mathbf{s}^N = \left(\sum_{k=1}^N \theta_k\right)^{-1} \hat{\mathbf{s}}$. Particularly, from Assumption (4.1) and (4.2),

$$\mathbb{E}[\delta_i^{t-1,k}] = 0, \quad \mathbb{E}[\|\delta_i^{t-1,k}\|_*^2] \leq \sigma^2, \quad \forall i \in \{1, \ldots, m\},\ t \geq 1,\ k \geq 1,$$

and from (4.7)

$$\tfrac{(T_1+1)(T_1+2)}{T_1(T_1+3)} = 1 + \tfrac{2}{T_1^2+3T_1} \leq \tfrac{3}{2}.$$

Therefore, by taking expectation over both sides of (4.10) and plugging in these values into (4.10), we have

$$\begin{aligned}\mathbb{E}[g(\hat{\mathbf{s}}^N, \hat{\mathbf{z}}^N)] &\le \left(\sum_{k=1}^N \theta_k\right)^{-1} \\ &\quad \left\{\tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)}\mathbf{V}(\mathbf{x}^0, \mathbf{x}) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2 + \sum_{k=1}^N \tfrac{8m(M^2+\sigma^2)\theta_k}{(T_k+3)\eta_k}\right\} \\ &\le \tfrac{\|\mathbf{L}\|}{N}\left[3\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{1}{2}\|\mathbf{y}^0\|^2 + 4\tilde{D}\right], \end{aligned} \tag{4.11}$$

with

$$\mathbb{E}[\|\hat{\mathbf{s}}^N\|] = \tfrac{1}{N}\mathbb{E}[\|\hat{\mathbf{s}}\|] \le \tfrac{\|\mathbf{L}\|}{N}\mathbb{E}\left[\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\| + \|\mathbf{y}^N - \mathbf{y}^*\| + \|\mathbf{y}^* - \mathbf{y}^0\|\right].$$

Note that from (4.6) and Jensen's inequality, we have

$$\begin{aligned}(\mathbb{E}[\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}])^2 &\le \mathbb{E}[\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2] \le 6\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \|\mathbf{y}^* - \mathbf{y}^0\| + 8\tilde{D}, \\ (\mathbb{E}[\|\mathbf{y}^* - \mathbf{y}^N\|])^2 &\le \mathbb{E}[\|\mathbf{y}^* - \mathbf{y}^N\|^2] \le 12\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + 2\|\mathbf{y}^* - \mathbf{y}^0\| + 16\tilde{D}.\end{aligned}$$

Hence,

$$\mathbb{E}[\|\hat{\mathbf{s}}^N\|] \le \tfrac{\|\mathbf{L}\|}{N}\left[3\sqrt{6\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + 8\tilde{D}} + 4\|\mathbf{y}^* - \mathbf{y}^0\|\right].$$

Applying Proposition 1 to the above inequality and (4.11), the results in (4.8) and (4.9) follow immediately. □

We now make some observations about the results obtained in Theorem 4. Firstly, one can choose any $\tilde{D} > 0$ (e.g., $\tilde{D} = 1$) in (4.7), however, the best selection of $\tilde{D}$ would be $\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*)$ so that the first and third terms in (4.11) are about the same order. In practice, if there exists an estimate $\mathcal{D}_{\mathbf{X}} > 0$ satisfying (3.25), we can set $\tilde{D} = \mathcal{D}_{\mathbf{X}}^2$.

Secondly, the complexity of SDCS method immediately follows from (4.8) and (4.9). Under the above assumption, with $\tilde{D} = \mathcal{D}_{\mathbf{X}}^2$ and $\mathbf{y}^0 = \mathbf{0}$, we can see that the total number of inter-node communication rounds and intra-node subgradient evaluations required by each agent for finding a stochastic (ϵ, δ)-solution of (2.3) can be bounded by

$$\begin{aligned}&\mathcal{O}\left\{\|\mathbf{L}\| \max\left(\tfrac{\mathcal{D}_{\mathbf{X}}^2}{\epsilon}, \tfrac{\mathcal{D}_{\mathbf{X}}+\|\mathbf{y}^*\|}{\delta}\right)\right\} \quad \text{and} \\ &\mathcal{O}\left\{m(M^2+\sigma^2)\max\left(\tfrac{\mathcal{D}_{\mathbf{X}}^2}{\epsilon^2}, \tfrac{\mathcal{D}_{\mathbf{X}}^2+\|\mathbf{y}^*\|^2}{\mathcal{D}_{\mathbf{X}}^2\delta^2}\right)\right\},\end{aligned} \tag{4.12}$$

respectively. In particular, if ϵ and δ satisfy (3.27), the above complexity bounds, respectively, reduce to

$$\mathcal{O}\left\{\tfrac{\|\mathbf{L}\|\mathcal{D}_{\mathbf{X}}^2}{\epsilon}\right\} \text{ and } \mathcal{O}\left\{\tfrac{m(M^2+\sigma^2)\mathcal{D}_{\mathbf{X}}^2}{\epsilon^2}\right\}. \tag{4.13}$$

In particular, we can show that the total number stochastic subgradients that SDCS requires is comparable to the mirror-descent stochastic approximation in [45]. This implies that the sample complexity for decentralized stochastic optimization are still optimal (as the centralized one), even after we skip many communication rounds.

4.3 Convergence of SDCS on strongly convex functions

We now provide in Lemma 4 an estimate on the gap function defined in (2.5) together with stepsize policies which work for the strongly convex case with $\mu > 0$ (cf. (1.2)). The proof of this lemma can be found in Sect. 5.

Note that throughout this subsection, we assume that the prox-functions $V_i(\cdot,\cdot)$, $i = 1,\dots,m$, (cf. (2.10)) are growing quadratically with the quadratic growth constant $\mathcal{C}$, i.e., (3.32) holds.

Lemma 4 *Let the iterates* $(\hat{\mathbf{x}}^k, \mathbf{y}^k)$, $k = 1,\dots,N$ *be generated by Algorithm 2,* $\hat{\mathbf{z}}^N$ *be defined as* $\hat{\mathbf{z}}^N := \left(\sum_{k=1}^N \theta_k\right)^{-1} \sum_{k=1}^N \theta_k(\hat{\mathbf{x}}^k, \mathbf{y}^k)$, *and Assumptions (4.1)–(4.2) hold. If the objective* f_i, $i = 1,\dots,m$ *are strongly convex functions, i.e.,* $\mu, M > 0$, *let the parameters* $\{\alpha_k\}$, $\{\theta_k\}$, $\{\eta_k\}$ *and* $\{\tau_k\}$ *in Algorithm 2 satisfy (3.14)–(3.17) and (3.33), and the parameters* $\{\lambda_t\}$ *and* $\{\beta_t\}$ *in the CS procedure of Algorithm 2 be set as (3.34). Then, for all* $\mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$,

$$\begin{aligned} Q(\hat{\mathbf{z}}^N;\mathbf{z}) \le \left(\sum_{k=1}^N \theta_k\right)^{-1} & \Bigg\{ \theta_1\eta_1 \mathbf{V}(\mathbf{x}^0,\mathbf{x}) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2 + \langle \hat{\mathbf{s}}, \mathbf{y}\rangle \\ & + \sum_{k=1}^N \sum_{t=1}^{T_k} \sum_{i=1}^m \tfrac{2\theta_k}{T_k(T_k+1)} \\ & \left[t\langle \delta_i^{t-1,k}, x_i - u_i^{t-1}\rangle + \tfrac{2t(M^2+\|\delta_i^{t-1,k}\|_*^2)}{(t+1)\mu/\mathcal{C}+(t-1)\eta_k}\right]\Bigg\}, \end{aligned} \tag{4.14}$$

where $\hat{\mathbf{s}} := \theta_N \mathbf{L}(\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}) + \theta_1\tau_1(\mathbf{y}^N - \mathbf{y}^0)$ *and* Q *is defined in (2.5). Furthermore, for any saddle point* $(\mathbf{x}^*, \mathbf{y}^*)$ *of (2.4), we have*

$$\begin{aligned} & \tfrac{\theta_N}{2}\left(1 - \tfrac{\|\mathbf{L}\|^2}{\eta_N\tau_N}\right)\max\{\eta_N\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2, \tau_N\|\mathbf{y}^* - \mathbf{y}^N\|^2\} \\ & \le \theta_1\eta_1\mathbf{V}(\mathbf{x}^0,\mathbf{x}^*) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^* - \mathbf{y}^0\|^2 + \sum_{k=1}^N \sum_{t=1}^{T_k}\sum_{i=1}^m \tfrac{2\theta_k}{T_k(T_k+1)} \\ & \left[t\langle \delta_i^{t-1,k}, x_i^* - u_i^{t-1}\rangle + \tfrac{2t(M^2+\|\delta_i^{t-1,k}\|_*^2)}{(t+1)\mu/\mathcal{C}+(t-1)\eta_k}\right]. \end{aligned} \tag{4.15}$$

In the following theorem, we provide a specific selection of $\{\alpha_k\}, \{\theta_k\}, \{\eta_k\}, \{\tau_k\}$ and $\{T_k\}$ satisfying (3.14)–(3.17) and (3.13). Also, by using Lemma 4 and Proposition 1, we establish the complexity of the SDCS method for computing an (ϵ, δ)-solution of problem (2.3) in expectation when the objective functions are strongly convex. Similar to the deterministic case, we choose variable stepsizes rather than constant stepsizes.

Theorem 5 *Let* $\mathbf{x}^*$ *be an optimal solution of* (2.3)*, the parameters* $\{\lambda_t\}$ *and* $\{\beta_t\}$ *in the CS procedure of Algorithm 2 be set as* (3.34)*, and suppose that* $\{\alpha_k\}, \{\theta_k\}, \{\eta_k\}, \{\tau_k\}$ *and* $\{T_k\}$ *are set to*

$$\begin{aligned}&\alpha_k = \tfrac{k}{k+1},\ \theta_k = k+1,\ \eta_k = \tfrac{k\mu}{2\mathcal{C}},\ \tau_k = \tfrac{4\|\mathbf{L}\|^2\mathcal{C}}{(k+1)\mu},\ \textit{and}\\ &T_k = \left\lceil \sqrt{\tfrac{m(M^2+\sigma^2)}{\tilde{D}}}\tfrac{2N\mathcal{C}}{\mu}\max\left\{\sqrt{\tfrac{m(M^2+\sigma^2)}{\tilde{D}}}\tfrac{8\mathcal{C}}{\mu}, 1\right\}\right\rceil,\quad \forall k = 1,\ldots,N,\end{aligned} \tag{4.16}$$

for some $\tilde{D} > 0$*. Then, under Assumptions* (4.1) *and* (4.2)*, we have for any* $N \geq 2$

$$\mathbb{E}[F(\bar{\mathbf{x}}^N) - F(\mathbf{x}^*) \leq \tfrac{2}{N(N+3)}\left[\tfrac{\mu}{\mathcal{C}}\mathbf{V}(\mathbf{x}^0,\mathbf{x}^*) + \tfrac{2\|\mathbf{L}\|^2\mathcal{C}}{\mu}\|\mathbf{y}^0\|^2 + \tfrac{2\mu\tilde{D}}{\mathcal{C}}\right], \tag{4.17}$$

and

$$\mathbb{E}[\|\mathbf{L}\hat{\mathbf{x}}^N\|] \leq \tfrac{8\|\mathbf{L}\|}{N(N+3)}\left[3\sqrt{2\tilde{D} + \mathbf{V}(\mathbf{x}^0,\mathbf{x}^*)} + \tfrac{7\|\mathbf{L}\|\mathcal{C}}{\mu}\|\mathbf{y}^* - \mathbf{y}^0\|\right], \tag{4.18}$$

where $\hat{\mathbf{x}}^N = \frac{2}{N(N+3)}\sum_{k=1}^N (k+1)\hat{\mathbf{x}}^k$*, and* y^* *is an arbitrary dual optimal solution.*

Proof It is easy to check that (4.16) satisfies conditions (3.14)–(3.17) and (3.33). Similarly, by (2.7), Assumption (4.1) and (4.2), we can obtain

$$\begin{aligned}\mathbb{E}[g(\hat{\mathbf{s}}^N, \hat{\mathbf{z}}^N)] \leq \left(\sum_{k=1}^N \theta_k\right)^{-1} &\left\{\theta_1\eta_1\mathbf{V}(\mathbf{x}^0,\mathbf{x}^*) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2\right.\\ &\left.+\sum_{k=1}^N\sum_{t=1}^{T_k}\sum_{i=1}^m \tfrac{2\theta_k}{T_k(T_k+1)}\left[\tfrac{2t(M^2+\sigma^2)}{(t+1)\mu/\mathcal{C}+(t-1)\eta_k}\right]\right\},\end{aligned} \tag{4.19}$$

where $\mathbf{s}^N = \left(\sum_{k=1}^N \theta_k\right)^{-1}\hat{\mathbf{s}}$. Particularly, from (4.16), we have

$$\begin{aligned}&\sum_{k=1}^N\sum_{t=1}^{T_k} \tfrac{4m(M^2+\sigma^2)\theta_k}{T_k(T_k+1)}\tfrac{t}{(t+1)\mu/\mathcal{C}+(t-1)\eta_k}\\ &\quad= \sum_{k=1}^N \tfrac{4m(M^2+\sigma^2)\mathcal{C}\theta_k}{T_k(T_k+1)\mu}\sum_{t=1}^{T_k}\tfrac{2t}{2(t+1)+(t-1)k}\end{aligned}$$

$$
\begin{aligned}
&\le \sum_{k=1}^{N} \tfrac{4m(M^2+\sigma^2)\mathcal{C}\theta_k}{T_k(T_k+1)\mu} \left(\tfrac{1}{2} + \sum_{t=2}^{T_k} \tfrac{2t}{(t-1)(k+1)} \right) \\
&\le \sum_{k=1}^{N} \tfrac{2m(M^2+\sigma^2)\mathcal{C}(k+1)}{T_k(T_k+1)\mu} + \sum_{k=1}^{N} \tfrac{16m(M^2+\sigma^2)\mathcal{C}(T_k-1)}{T_k(T_k+1)\mu} \le \tfrac{2\mu\tilde{D}}{\mathcal{C}}.
\end{aligned}
$$

Therefore, by plugging in these values into (4.19), we have

$$
\mathbb{E}[g(\hat{\mathbf{s}}^N, \hat{\mathbf{z}}^N)] \le \tfrac{2}{N(N+3)} \left[\tfrac{\mu}{\mathcal{C}} \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{2\|L\|^2\mathcal{C}}{\mu} \|\mathbf{y}^0\|^2 + \tfrac{2\mu\tilde{D}}{\mathcal{C}} \right], \tag{4.20}
$$

with

$$
\begin{aligned}
\mathbb{E}[\|\hat{\mathbf{s}}^N\|] &= \tfrac{2}{N(N+3)} \mathbb{E}[\|\hat{\mathbf{s}}\|] \\
&\le \tfrac{2\|\mathbf{L}\|}{N(N+3)} \mathbb{E}\left[(N+1)\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\| + \tfrac{4\|\mathbf{L}\|\mathcal{C}}{\mu} \left(\|\mathbf{y}^N - \mathbf{y}^*\| + \|\mathbf{y}^* - \mathbf{y}^0\| \right) \right].
\end{aligned}
$$

Note that from (4.15), we have, for any $N \ge 2$,

$$
\begin{aligned}
\mathbb{E}[\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2] &\le \tfrac{8}{(N+1)(N-1)} \left[\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{2\|\mathbf{L}\|^2\mathcal{C}^2}{\mu^2} \|\mathbf{y}^0 - \mathbf{y}^*\|^2 + 2\tilde{D} \right], \\
\mathbb{E}[\|\mathbf{y}^* - \mathbf{y}^N\|^2] &\le \tfrac{N\mu}{(N-1)\|\mathbf{L}\|^2\mathcal{C}} \left[\tfrac{\mu}{\mathcal{C}} \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{2\|\mathbf{L}\|^2\mathcal{C}}{\mu} \|\mathbf{y}^0 - \mathbf{y}^*\|^2 + \tfrac{2\mu\tilde{D}}{\mathcal{C}} \right].
\end{aligned}
$$

Hence, in view of the above three relations and Jensen's inequality, we obtain

$$
\begin{aligned}
\mathbb{E}[\|\hat{\mathbf{s}}^N\|] &\le \tfrac{8\|\mathbf{L}\|}{N(N+3)} \left[3\sqrt{2\tilde{D} + \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{2\|\mathbf{L}\|^2\mathcal{C}^2}{\mu^2} \|\mathbf{y}^0 - \mathbf{y}^*\|^2} + \tfrac{\|\mathbf{L}\|\mathcal{C}}{\mu} \|\mathbf{y}^* - \mathbf{y}^0\| \right] \\
&\le \tfrac{8\|\mathbf{L}\|}{N(N+3)} \left[3\sqrt{2\tilde{D} + \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*)} + \tfrac{7\|\mathbf{L}\|\mathcal{C}}{\mu} \|\mathbf{y}^* - \mathbf{y}^0\| \right].
\end{aligned}
$$

Applying Proposition 1 to the above inequality and (4.20), the results in (4.17) and (4.18) follow immediately. □

We now make some observations about the results obtained in Theorem 5. Firstly, similar to the general convex case, the best choice for $\tilde{D}$ (cf. (4.16)) would be $\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*)$ so that the first and the third terms in (4.20) are about the same order. If there exists an estimate $\mathcal{D}_{\mathbf{X}} > 0$ satisfying (3.25), we can set $\tilde{D} = \mathcal{D}_{\mathbf{X}}^2$.

Secondly, the complexity of SDCS method for solving strongly convex problems follows from (4.17) and (4.18). Under the above assumption, with $\tilde{D} = \mathcal{D}_{\mathbf{X}}^2$ and $\mathbf{y}^0 = \mathbf{0}$, the total number of inter-node communication rounds and intra-node subgradient evaluations performed by each agent for finding a stochastic (ϵ, δ)-solution of (2.3) can be bounded by

$$\mathcal{O}\left\{\max\left(\sqrt{\frac{\mu\mathcal{D}_{\mathbf{X}}^2}{\mathcal{C}\epsilon}}, \sqrt{\frac{\|\mathbf{L}\|}{\delta}\left(\mathcal{D}_{\mathbf{X}} + \frac{\mathcal{C}\|\mathbf{L}\|\|\mathbf{y}^*\|}{\mu}\right)}\right)\right\} \text{ and}$$

$$\mathcal{O}\left\{\frac{m(M^2+\sigma^2)\mathcal{C}}{\mu}\max\left(\frac{1}{\epsilon}, \frac{\mathcal{C}\|\mathbf{L}\|}{\mu\delta}\left(\frac{1}{\mathcal{D}_{\mathbf{X}}} + \frac{\mathcal{C}\|\mathbf{L}\|\|\mathbf{y}^*\|}{\mathcal{D}_{\mathbf{X}}^2\mu}\right)\right)\right\}, \tag{4.21}$$

respectively. In particular, if ϵ and δ satisfy (3.43), the above complexity bounds, respectively, reduce to

$$\mathcal{O}\left\{\sqrt{\frac{\mu\mathcal{D}_{\mathbf{X}}^2}{\mathcal{C}\epsilon}}\right\} \text{ and } \mathcal{O}\left\{\frac{m(M^2+\sigma^2)\mathcal{C}}{\mu\epsilon}\right\}. \tag{4.22}$$

We can see that the total number of stochastic subgradient computations is comparable to the optimal complexity bound obtained in [19,20] for stochastic strongly convex case in the centralized case.

4.4 High probability results

All of the results stated in Sects. 4.2, 4.3 are established in terms of expectation. In order to provide high probability results for SDCS method, we additionally need the following “light-tail” assumption:

$$\mathbb{E}\left[\exp\left\{\|G_i(u^t, \xi_i^t) - f_i'(u^t)\|_*^2/\sigma^2\right\}\right] \le \exp\{1\}. \tag{4.23}$$

Note that (4.23) is stronger than (4.2), since it implies (4.2) by Jensen’s inequality. Moreover, we also assume that there exists $\bar{\mathbf{V}}(\mathbf{x}^*)$ s.t.

$$\bar{\mathbf{V}}(\mathbf{x}^*) := \sum_{i=1}^m \bar{V}_i(x_i^*) := \sum_{i=1}^m \max_{x_i \in X_i} V_i(x_i^*, x_i). \tag{4.24}$$

The following theorem provides a large deviation result for the gap function $g(\hat{\mathbf{s}}^N, \hat{\mathbf{z}}^N)$ when our objective functions f_i, $i = 1, \ldots, m$ are general nonsmooth convex functions.

Theorem 6 *Let x^* be an optimal solution of (2.3), Assumptions (4.1), (4.2) and (4.23) hold, the parameters $\{\alpha_k\}$, $\{\theta_k\}$, $\{\eta_k\}$, $\{\tau_k\}$ and $\{T_k\}$ in Algorithm 2 satisfy (3.13)–(3.17), and the parameters $\{\lambda_t\}$ and $\{\beta_t\}$ in the CS procedure of Algorithm 2 be set as (3.18). In addition, if X_i’s are compact, then for any $\zeta > 0$ and $N \ge 1$, we have*

$$\text{Prob}\left\{g(\hat{\mathbf{s}}^N, \hat{\mathbf{z}}^N) \ge \mathcal{B}_d(N) + \zeta\mathcal{B}_p(N)\right\} \le \exp\{-\zeta^2/3\} + \exp\{-\zeta\}, \tag{4.25}$$

where

$$\mathcal{B}_d(N) := \left(\sum_{k=1}^{N}\theta_k\right)^{-1}\left[\tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)}\mathbf{V}(\mathbf{x}^0,\mathbf{x}^*) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2 \right.$$
$$\left. + \sum_{k=1}^{N}\tfrac{8m(M^2+\sigma^2)\theta_k}{\eta_k(T_k+3)}\right], \tag{4.26}$$

and

$$\mathcal{B}_p(N) := \left(\sum_{k=1}^{N}\theta_k\right)^{-1}\left\{\sigma\left[2\bar{\mathbf{V}}(\mathbf{x}^*)\sum_{k=1}^{N}\sum_{t=1}^{T_k}\left(\tfrac{\theta_k\lambda_t}{\sum_{t=1}^{T_k}\lambda_t}\right)^2\right]^{1/2}\right.$$
$$\left. + \sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m}\tfrac{\sigma^2\theta_k\lambda_t}{\left(\sum_{t=1}^{T_k}\lambda_t\right)\eta_k\beta_t}\right\}. \tag{4.27}$$

In the next corollary, we establish the rate of convergence of SDCS in terms of both primal and feasibility (or consistency) residuals are of order $\mathcal{O}(1/N)$ with high probability when the objective functions are nonsmooth and convex.

Corollary 1 *Let $\mathbf{x}^*$ be an optimal solution of* (2.3)*, y^* be an arbitrary dual optimal solution, the parameters $\{\lambda_t\}$ and $\{\beta_t\}$ in the CS procedure of Algorithm 2 be set as* (3.18)*, and suppose that $\{\alpha_k\}$, $\{\theta_k\}$, $\{\eta_k\}$, $\{\tau_k\}$ and $\{T_k\}$ are set to* (4.7) *with $\tilde{D} = \bar{\mathbf{V}}(\mathbf{x}^*)$. Under Assumptions* (4.1)*,* (4.2) *and* (4.23)*, we have for any $N \geq 1$ and $\zeta > 0$*

$$\text{Prob}\left\{F(\hat{\mathbf{x}}^N) - F(\mathbf{x}^*) \geq \tfrac{\|\mathbf{L}\|}{N}\left[(7+8\zeta)\bar{\mathbf{V}}(\mathbf{x}^*) + \tfrac{1}{2}\|\mathbf{y}^0\|^2\right]\right\} \leq \exp\{-\zeta^2/3\} + \exp\{-\zeta\}, \tag{4.28}$$

and

$$\text{Prob}\left\{\|\mathbf{L}\hat{\mathbf{x}}^N\|^2 \geq \tfrac{18\|\mathbf{L}\|^2}{N^2}\left[(7+8\zeta)\bar{\mathbf{V}}(\mathbf{x}^*) + \tfrac{2}{3}\|\mathbf{y}^* - \mathbf{y}^0\|^2\right]\right\} \leq \exp\{-\zeta^2/3\} + \exp\{-\zeta\}. \tag{4.29}$$

Proof Observe that by the definition of λ_t in (3.18),

$$\sum_{t=1}^{T_k}\left[\tfrac{\theta_k\lambda_t}{\sum_{t=1}^{T_k}\lambda_t}\right]^2 = \left(\tfrac{2}{T_k(T_k+3)}\right)^2\sum_{t=1}^{T_k}(t+1)^2$$
$$= \left(\tfrac{2}{T_k(T_k+3)}\right)^2\tfrac{(T_k+1)(T_k+2)(2T_k+3)}{6} \leq \tfrac{8}{3T_k},$$

which together with (4.27) then imply that

$$\mathcal{B}_p(N) \le \tfrac{1}{N}\left\{\sigma\left[2\bar{\mathbf{V}}(\mathbf{x}^*)\sum_{k=1}^{N}\tfrac{8}{3T_k}\right]^{1/2} + \sum_{k=1}^{N}\tfrac{8m\sigma^2}{\|\mathbf{L}\|(T_k+3)}\right\}$$
$$\le \tfrac{4\|\mathbf{L}\|}{N}\left\{\sqrt{\tfrac{\bar{\mathbf{V}}(\mathbf{x}^*)\tilde{D}}{3m}} + \tilde{D}\right\} \le \tfrac{8\|\mathbf{L}\|\bar{\mathbf{V}}(\mathbf{x}^*)}{N}.$$

Hence, (4.28) follows from the above relation, (4.25) and Proposition 1. Note that from (4.6) and plugging in (4.7) with $\tilde{D} = \bar{\mathbf{V}}(\mathbf{x}^*)$, we obtain

$$\begin{aligned}
\|\hat{\mathbf{s}}^N\|^2 &= \left(\sum_{k=1}^{N}\theta_k\right)^{-2}\|\hat{\mathbf{s}}\|^2\\
&\le \left(\sum_{k=1}^{N}\theta_k\right)^{-2}\\
&\quad\left\{3\theta_N^2\|\mathbf{L}\|^2\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2 + 3\theta_1^2\tau_1^2\left(\|\mathbf{y}^N - \mathbf{y}^*\|^2 + \|\mathbf{y}^* - \mathbf{y}^0\|^2\right)\right\}\\
&\le \tfrac{3\|\mathbf{L}\|^2}{N^2}\left\{18\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + 4\|\mathbf{y}^* - \mathbf{y}^0\|^2 + \sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m}\tfrac{12\theta_k}{T_k(T_k+3)\|\mathbf{L}\|}\right.\\
&\quad\left.\left[(t+1)\langle\delta_i^{t-1,k}, x_i^* - u_i^{t-1}\rangle + \tfrac{4(M^2+\|\delta_i^{t-1,k}\|_*^2)}{\eta_k}\right]\right\}.
\end{aligned}$$

Hence, similarly, we have

$$\mathrm{Prob}\left\{\|\hat{\mathbf{s}}^N\|^2 \ge \tfrac{18\|\mathbf{L}\|^2}{N^2}\left[(7+8\zeta)\bar{\mathbf{V}}(\mathbf{x}^*) + \tfrac{2}{3}\|\mathbf{y}^* - \mathbf{y}^0\|^2\right]\right\} \le \exp\{-\zeta^2/3\} + \exp\{-\zeta\},$$

which in view of Proposition 1 immediately implies (4.29). □

5 Convergence analysis

This section is devoted to the proof of the main lemmas in Sects. 3 and 4, which establish the convergence results of the deterministic and stochastic decentralized communication sliding methods, respectively. After introducing some general results about these algorithms, we provide the proofs for Lemmas 1–4 and Theorem 6.

The following lemma below characterizes the solution of the dual projection step (3.6), as well as the projection in inner loop (3.10). The proof of this result can be found in Lemma 2 of [19].

Lemma 5 *Let the convex function $q : U \to \mathbb{R}$, the points $\bar{x}, \bar{y} \in U$ and the scalars $\mu_1, \mu_2 \in \mathbb{R}$ be given. Let $\omega : U \to \mathbb{R}$ be a differentiable convex function and $V(x, z)$ be defined in (2.10). If*

$$u^* \in \operatorname{argmin} \{q(u) + \mu_1 V(\bar{x}, u) + \mu_2 V(\bar{y}, u) : u \in U\},$$

then for any $u \in U$, we have

$$\begin{aligned} &q(u^*) + \mu_1 V(\bar{x}, u^*) + \mu_2 V(\bar{y}, u^*) \\ &\quad \le q(u) + \mu_1 V(\bar{x}, u) + \mu_2 V(\bar{y}, u) - (\mu_1 + \mu_2) V(u^*, u). \end{aligned}$$

Before we provide proofs for Lemma 1–4, we first need to present a result which summarizes an important convergence property of the CS procedure. It needs to be mentioned that the following proposition states a general result holds for CS procedure performed by individual agent $i \in \mathcal{N}$. For notation convenience, we use the notations defined in CS procedure (cf. Algorithm 1).

Proposition 2 *If $\{\beta_t\}$ and $\{\lambda_t\}$ in the* CS *procedure satisfy*

$$\lambda_{t+1}(\eta\beta_{t+1} - \mu/\mathcal{C}) \le \lambda_t(1 + \beta_t)\eta, \quad \forall t \ge 1. \tag{5.1}$$

then, for $t \ge 1$ and $u \in U$,

$$\begin{aligned} &\left(\sum_{t=1}^{T} \lambda_t\right)^{-1} \left[\eta(1 + \beta_T)\lambda_T V(u^T, u) + \sum_{t=1}^{T} \lambda_t \langle \delta^{t-1}, u - u^{t-1}\rangle\right] \\ &\quad + \Phi(\hat{u}^T) - \Phi(u) \\ &\le \left(\sum_{t=1}^{T} \lambda_t\right)^{-1} \left[(\eta\beta_1 - \mu/\mathcal{C})\lambda_1 V(u^0, u) + \sum_{t=1}^{T} \tfrac{(M + \|\delta^{t-1}\|_*)^2 \lambda_t}{2\eta\beta_t}\right], \end{aligned} \tag{5.2}$$

where Φ is defined as

$$\Phi(u) := \langle w, u\rangle + \phi(u) + \eta V(x, u) \tag{5.3}$$

and $\delta^t := \phi'(u^t) - h^t$.

Proof Noticing that $\phi := f_i$ in the CS procedure, we have by (1.2)

$$\begin{aligned} \phi(u^t) &\le \phi(u^{t-1}) + \langle \phi'(u^{t-1}), u^t - u^{t-1}\rangle + M\|u^t - u^{t-1}\| \\ &= \phi(u^{t-1}) + \langle \phi'(u^{t-1}), u - u^{t-1}\rangle + \langle \phi'(u^{t-1}), u^t - u\rangle + M\|u^t - u^{t-1}\| \\ &\le \phi(u) - \tfrac{\mu}{2}\|u - u^{t-1}\|^2 + \langle \phi'(u^{t-1}), u^t - u\rangle + M\|u^t - u^{t-1}\|, \end{aligned}$$

where $\phi'(u^{t-1}) \in \partial\phi(u^{t-1})$ and $\partial\phi(u^{t-1})$ denotes the subdifferential of ϕ at u^{t-1}. By applying Lemma 5 to (3.10), we obtain

$$\begin{aligned} &\langle w + h^{t-1}, u^t - u\rangle + \eta V(x, u^t) - \eta V(x, u) \\ &\quad \le \eta\beta_t V(u^{t-1}, u) - \eta(1 + \beta_t) V(u^t, u) - \eta\beta_t V(u^{t-1}, u^t), \quad \forall u \in U. \end{aligned}$$

Combining the above two relations together with (3.32),[2] we conclude that

$$\begin{aligned}&\langle w, u^t - u\rangle + \phi(u^t) - \phi(u) + \langle \delta^{t-1}, u - u^{t-1}\rangle + \eta V(x, u^t) - \eta V(x, u)\\ &\quad\le (\eta\beta_t - \mu/\mathcal{C})V(u^{t-1}, u) - \eta(1+\beta_t)V(u^t, u) + \langle \delta^{t-1}, u^t - u^{t-1}\rangle\\ &\qquad + M\|u^t - u^{t-1}\| - \eta\beta_t V(u^{t-1}, u^t), \quad \forall u \in U. \end{aligned} \tag{5.4}$$

Moreover, by Cauchy–Schwarz inequality, (2.11), and the simple fact that $-at^2/2 + bt \le b^2/(2a)$ for any $a > 0$, we have

$$\begin{aligned}&\langle \delta^{t-1}, u^t - u^{t-1}\rangle + M\|u^t - u^{t-1}\| - \eta\beta_t V(u^{t-1}, u^t)\\ &\quad\le (\|\delta^{t-1}\|_* + M)\|u^t - u^{t-1}\| - \tfrac{\eta\beta_t}{2}\|u^t - t^{t-1}\|^2 \le \tfrac{(M+\|\delta^{t-1}\|_*)^2}{2\eta\beta_t}.\end{aligned}$$

From the above relation and the definition of $\Phi(u)$ in (5.3), we can rewrite (5.4) as,

$$\begin{aligned}\Phi(u^t) - \Phi(u) + \langle \delta^{t-1}, u - u^{t-1}\rangle \le (\eta\beta_t - \mu/\mathcal{C})V(u^{t-1}, u) - \eta(1+\beta_t)V(u^t, u)\\ + \tfrac{(M+\|\delta^{t-1}\|_*)^2}{2\eta\beta_t}, \quad \forall u \in U.\end{aligned}$$

Multiplying both sides by λ_t and summing up the resulting inequalities from $t = 1$ to T, we obtain

$$\begin{aligned}&\sum_{t=1}^{T}\lambda_t\left[\Phi(u^t) - \Phi(u) + \langle \delta^{t-1}, u - u^{t-1}\rangle\right]\\ &\quad\le \sum_{t=1}^{T}\left[(\eta\beta_t - \mu/\mathcal{C})\lambda_t V(u^{t-1}, u) - \eta(1+\beta_t)\lambda_t V(u^t, u)\right]\\ &\qquad + \sum_{t=1}^{T}\tfrac{(M+\|\delta^{t-1}\|_*)^2\lambda_t}{2\eta\beta_t}.\end{aligned}$$

Hence, in view of (5.1), the convexity of Φ and the definition of $\hat{u}^T$ in (3.11), we have

$$\begin{aligned}&\Phi(\hat{u}^T) - \Phi(u) + \left(\sum_{t=1}^{T}\lambda_t\right)^{-1}\sum_{t=1}^{T}\lambda_t\langle \delta^{t-1}, u - u^{t-1}\rangle\\ &\quad\le \left(\sum_{t=1}^{T}\lambda_t\right)^{-1}\left[(\eta\beta_1 - \mu/\mathcal{C})\lambda_1 V(u^0, u) - \eta(1+\beta_T)\lambda_T V(u^T, u)\right.\\ &\qquad\left. + \sum_{t=1}^{T}\tfrac{(M+\|\delta^{t-1}\|_*)^2\lambda_t}{2\eta\beta_t}\right],\end{aligned}$$

[2] Observed that we only need condition (3.32) when $\mu > 0$, in other words, the objective functions f_i's are strongly convex.

which implies (5.2) immediately. □

As a matter of fact, the SDCS method covers the DCS method as a special case when $\delta^t = 0,\ \forall t \geq 0$. Therefore, we investigate the proofs for Lemmas 3 and 4 first and then simplify them for the proofs for Lemmas 1 and 2. We now provide a proof for Lemma 3, which establishes the convergence property of SDCS method for solving general convex problems.

Proof of Lemma 3 When f_i, $i = 1, \ldots, m$, are general convex functions, we have $\mu = 0$ and $M > 0$ (cf. (1.2)). Therefore, in view of $\phi := f_i$, and λ_t and β_t defined in (3.18) satisfying condition (5.1) in the CS procedure, Eq. (5.2) can be rewritten as the following,[3]

$$\begin{aligned}
&\left(\sum_{t=1}^{T}\lambda_t\right)^{-1}\left[\eta(1+\beta_T)\lambda_T V_i(u_i^T, u_i) + \sum_{t=1}^{T}\lambda_t\langle\delta_i^{t-1}, u_i - u_i^{t-1}\rangle\right]\\
&\quad + \Phi_i(\hat{u}_i^T) - \Phi_i(u_i)\\
&\quad \leq \left(\sum_{t=1}^{T}\lambda_t\right)^{-1}\left[\eta\beta_1\lambda_1 V_i(u_i^0, u_i) + \sum_{t=1}^{T}\frac{(M+\|\delta_i^{t-1}\|_*)^2\lambda_t}{2\eta\beta_t}\right],\ \forall u_i \in X_i.
\end{aligned}$$

In view of the above relation, the definition of Φ^k in (3.3), and the input and output settings in the CS procedure, it is not difficult to see that, for any $k \geq 1$,[4]

$$\begin{aligned}
&\Phi^k(\hat{\mathbf{x}}^k) - \Phi^k(\mathbf{x})\\
&\quad + \left(\sum_{t=1}^{T_k}\lambda_t\right)^{-1}\left[\eta_k(1+\beta_{T_k})\lambda_{T_k}\mathbf{V}(\mathbf{x}^k, \mathbf{x}) + \sum_{t=1}^{T_k}\sum_{i=1}^{m}\lambda_t\langle\delta_i^{t-1,k}, x_i - u_i^{t-1}\rangle\right]\\
&\quad \leq \left(\sum_{t=1}^{T_k}\lambda_t\right)^{-1}\left[\eta_k\beta_1\lambda_1\mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x}) + \sum_{t=1}^{T_k}\sum_{i=1}^{m}\frac{(M+\|\delta_i^{t-1,k}\|_*)^2\lambda_t}{2\eta_k\beta_t}\right],\quad \forall \mathbf{x} \in \mathbf{X}.
\end{aligned}$$

By plugging into the above relation the values of λ_t and β_t in (3.18), together with the definition of Φ^k in (3.3) and rearranging the terms, we have,

$$\begin{aligned}
&\langle\mathbf{L}(\hat{\mathbf{x}}^k - \mathbf{x}), \mathbf{y}^k\rangle + F(\hat{\mathbf{x}}^k) - F(\mathbf{x})\\
&\quad \leq \tfrac{(T_k+1)(T_k+2)\eta_k}{T_k(T_k+3)}\left[\mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x}) - \mathbf{V}(\mathbf{x}^k, \mathbf{x})\right] - \eta_k\mathbf{V}(\mathbf{x}^{k-1}, \hat{\mathbf{x}}^k)\\
&\quad + \tfrac{2}{T_k(T_k+3)}\sum_{t=1}^{T_k}\sum_{i=1}^{m}\left[(t+1)\langle\delta_i^{t-1,k}, x_i - u_i^{t-1}\rangle + \tfrac{2(M+\|\delta_i^{t-1,k}\|_*)^2}{\eta_k}\right],\qquad \forall \mathbf{x} \in \mathbf{X}.
\end{aligned}$$

[3] We added the subscript i to emphasize that this inequality holds for any agent $i \in \mathcal{N}$ with $\phi = f_i$. More specifically, $\Phi_i(u_i) := \langle w_i, u_i\rangle + f_i(u_i) + \eta V_i(x_i, u_i)$.

[4] We added the superscript k in $\delta_i^{t-1,k}$ to emphasize that this error is generated at the k-th outer loop.

Moreover, applying Lemma 5 to (3.6), we have, for $k \geq 1$,

$$\langle v_i^k, y_i - y_i^k \rangle \leq \tfrac{\tau_k}{2} \left[\|y_i - y_i^{k-1}\|^2 - \|y_i - y_i^k\|^2 - \|y_i^{k-1} - y_i^k\|^2 \right], \quad \forall y_i \in \mathbb{R}^d, \tag{5.5}$$

which in view of the definition of Q in (2.5) and the above two relations, then implies that, for $k \geq 1$, $\mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$,

$$\begin{aligned} Q(\hat{\mathbf{x}}^k, \mathbf{y}^k; \mathbf{z}) = F(\hat{\mathbf{x}}^k) - F(\mathbf{x}) + \langle \mathbf{L}\hat{\mathbf{x}}^k, \mathbf{y} \rangle - \langle \mathbf{L}\mathbf{x}, \mathbf{y}^k \rangle \\ \leq \langle \mathbf{L}(\hat{\mathbf{x}}^k - \tilde{\mathbf{x}}^k), \mathbf{y} - \mathbf{y}^k \rangle + \tfrac{(T_k+1)(T_k+2)\eta_k}{T_k(T_k+3)} \left[\mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x}) - \mathbf{V}(\mathbf{x}^k, \mathbf{x}) \right] \\ -\ \eta_k \mathbf{V}(\mathbf{x}^{k-1}, \hat{\mathbf{x}}^k) + \tfrac{\tau_k}{2} \left[\|\mathbf{y} - \mathbf{y}^{k-1}\|^2 - \|\mathbf{y} - \mathbf{y}^k\|^2 - \|\mathbf{y}^{k-1} - \mathbf{y}^k\|^2 \right] \\ +\ \tfrac{2}{T_k(T_k+3)} \sum_{t=1}^{T_k} \sum_{i=1}^{m} \left[(t+1)\langle \delta_i^{t-1,k}, x_i - u_i^{t-1} \rangle + \tfrac{2(M + \|\delta_i^{t-1,k}\|_*)^2}{\eta_k} \right]. \end{aligned}$$

Multiplying both sides of the above inequality by θ_k, and summing up the resulting inequalities from $k = 1$ to N, we obtain, for all $\mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$,

$$\begin{aligned} \sum_{k=1}^{N} \theta_k Q(\hat{\mathbf{x}}^k, \mathbf{y}^k; \mathbf{z}) \leq \sum_{k=1}^{N} \theta_k \Delta_k + \sum_{k=1}^{N} \sum_{t=1}^{T_k} \sum_{i=1}^{m} \tfrac{2\theta_k}{T_k(T_k+3)} \\ \left[(t+1)\langle \delta_i^{t-1,k}, x_i - u_i^{t-1} \rangle + \tfrac{2(M + \|\delta_i^{t-1,k}\|_*)^2}{\eta_k} \right], \end{aligned} \tag{5.6}$$

where

$$\begin{aligned} \Delta_k := \langle \mathbf{L}(\hat{\mathbf{x}}^k - \tilde{\mathbf{x}}^k), \mathbf{y} - \mathbf{y}^k \rangle + \tfrac{(T_k+1)(T_k+2)\eta_k}{T_k(T_k+3)} \left[\mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x}) - \mathbf{V}(\mathbf{x}^k, \mathbf{x}) \right] \\ -\ \eta_k \mathbf{V}(\mathbf{x}^{k-1}, \hat{\mathbf{x}}^k) + \tfrac{\tau_k}{2} \left[\|\mathbf{y} - \mathbf{y}^{k-1}\|^2 - \|\mathbf{y} - \mathbf{y}^k\|^2 - \|\mathbf{y}^{k-1} - \mathbf{y}^k\|^2 \right]. \end{aligned} \tag{5.7}$$

We now provide a bound on $\sum_{k=1}^{N} \theta_k \Delta_k$. Observe that from the definition of $\tilde{\mathbf{x}}^k$ in (3.1), (3.13) and (3.15) we have

$$\begin{aligned} \sum_{k=1}^{N} \theta_k \Delta_k \leq \sum_{k=1}^{N} \left[\theta_k \langle \mathbf{L}(\mathbf{x}^k - \mathbf{x}^{k-1}), \mathbf{y} - \mathbf{y}^k \rangle - \alpha_k \theta_k \langle \mathbf{L}(\mathbf{x}^{k-1} - \mathbf{x}^{k-2}), \mathbf{y} - \mathbf{y}^{k-1} \rangle \right] \\ - \sum_{k=1}^{N} \theta_k \left[\alpha_k \langle \mathbf{L}(\mathbf{x}^{k-1} - \mathbf{x}^{k-2}), \mathbf{y}^{k-1} - \mathbf{y}^k \rangle + \eta_k \mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x}^k) \right. \\ \left. + \tfrac{\tau_k}{2} \|\mathbf{y}^{k-1} - \mathbf{y}^k\|^2 \right] + \tfrac{(T_1+1)(T_1+2)\theta_1 \eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) \\ - \tfrac{(T_N+1)(T_N+2)\theta_N \eta_N}{T_N(T_N+3)} \mathbf{V}(\mathbf{x}^N, \mathbf{x}) + \tfrac{\theta_1 \tau_1}{2} \|\mathbf{y} - \mathbf{y}^0\|^2 - \tfrac{\theta_N \tau_N}{2} \|\mathbf{y} - \mathbf{y}^N\|^2 \end{aligned} \tag{5.8}$$

$$
\begin{aligned}
&\overset{(a)}{\le} \theta_N \langle \mathbf{L}(\mathbf{x}^N - \mathbf{x}^{N-1}), \mathbf{y} - \mathbf{y}^N \rangle - \theta_N \eta_N \mathbf{V}(\mathbf{x}^{N-1}, \mathbf{x}^N) \\
&\quad - \sum_{k=2}^{N} \Big[\theta_k \alpha_k \langle \mathbf{L}(\mathbf{x}^{k-1} - \mathbf{x}^{k-2}), \mathbf{y}^{k-1} - \mathbf{y}^k \rangle + \theta_{k-1}\eta_{k-1} \mathbf{V}(\mathbf{x}^{k-2}, \mathbf{x}^{k-1}) \\
&\quad + \tfrac{\theta_k \tau_k}{2} \|\mathbf{y}^{k-1} - \mathbf{y}^k\|^2 \Big] + \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) \\
&\quad - \tfrac{(T_N+1)(T_N+2)\theta_N\eta_N}{T_N(T_N+3)} \mathbf{V}(\mathbf{x}^N, \mathbf{x}) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y} - \mathbf{y}^0\|^2 - \tfrac{\theta_N \tau_N}{2}\|\mathbf{y} - \mathbf{y}^N\|^2 \\
&\overset{(b)}{\le} \theta_N \langle \mathbf{L}(\mathbf{x}^N - \mathbf{x}^{N-1}), \mathbf{y} - \mathbf{y}^N \rangle - \theta_N \eta_N \mathbf{V}(\mathbf{x}^{N-1}, \mathbf{x}^N) \\
&\quad + \sum_{k=2}^{N} \left(\tfrac{\theta_{k-1}\alpha_k \|\mathbf{L}\|^2}{2\tau_k} - \tfrac{\theta_{k-1}\eta_{k-1}}{2} \right) \|\mathbf{x}^{k-2} - \mathbf{x}^{k-1}\|^2 \\
&\quad + \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) - \tfrac{(T_N+1)(T_N+2)\theta_N\eta_N}{T_N(T_N+3)} \mathbf{V}(\mathbf{x}^N, \mathbf{x}) \\
&\quad + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y} - \mathbf{y}^0\|^2 - \tfrac{\theta_N \tau_N}{2}\|\mathbf{y} - \mathbf{y}^N\|^2 \\
&\overset{(c)}{\le} \theta_N \langle \mathbf{L}(\mathbf{x}^N - \mathbf{x}^{N-1}), \mathbf{y} - \mathbf{y}^N \rangle - \theta_N \eta_N \mathbf{V}(\mathbf{x}^{N-1}, \mathbf{x}^N) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y} - \mathbf{y}^0\|^2 \\
&\quad - \tfrac{\theta_N \tau_N}{2}\|\mathbf{y} - \mathbf{y}^N\|^2 + \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) \\
&\quad - \tfrac{(T_N+1)(T_N+2)\theta_N\eta_N}{T_N(T_N+3)} \mathbf{V}(\mathbf{x}^N, \mathbf{x}) \\
&\overset{(d)}{\le} \theta_N \langle \mathbf{y}^N, \mathbf{L}(\mathbf{x}^{N-1} - \mathbf{x}^N) \rangle - \theta_N \eta_N \mathbf{V}(\mathbf{x}^{N-1}, \mathbf{x}^N) - \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^N\|^2 \\
&\quad + \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2 + \langle \mathbf{y}, \theta_N \mathbf{L}(\mathbf{x}^N - \mathbf{x}^{N-1}) \\
&\quad + \theta_1\tau_1(\mathbf{y}^N - \mathbf{y}^0) \rangle, \\
&\overset{(e)}{\le} \left(\tfrac{\theta_N \|\mathbf{L}\|^2}{2\eta_N} - \tfrac{\theta_1\tau_1}{2} \right) \|\mathbf{y}^N\|^2 + \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2 \\
&\quad + \langle \mathbf{y}, \theta_N \mathbf{L}(\mathbf{x}^N - \mathbf{x}^{N-1}) + \theta_1\tau_1(\mathbf{y}^N - \mathbf{y}^0) \rangle,
\end{aligned}
$$

where (a) follows from (3.14) and the fact that $\mathbf{x}^{-1} = \mathbf{x}^0$, (b) follows from the simple relation that $b\langle u, v\rangle - a\|v\|^2/2 \le b^2\|u\|^2/(2a), \forall a > 0$, (3.14) and (2.14), (c) follows from (3.16), (d) follows from (3.15), $\|\mathbf{y} - \mathbf{y}^0\|^2 - \|\mathbf{y} - \mathbf{y}^N\|^2 = \|\mathbf{y}^0\|^2 - \|\mathbf{y}^N\|^2 - 2\langle \mathbf{y}, \mathbf{y}^0 - \mathbf{y}^N \rangle$ and arranging the terms accordingly, (e) follows from (2.14) and the relation $b\langle u, v\rangle - a\|v\|^2/2 \le b^2\|u\|^2/(2a), \forall a > 0$. Using the above bound in (5.6) we obtain

$$
\begin{aligned}
&\sum_{k=1}^{N} \theta_k Q(\hat{\mathbf{x}}^k, \mathbf{y}^k; \mathbf{z}) \\
&\quad \le \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2 + \langle \hat{\mathbf{s}}, \mathbf{y} \rangle \\
&\qquad + \sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m} \tfrac{2\theta_k}{T_k(T_k+3)} \left[(t+1)\langle \delta_i^{t-1,k}, x_i - u_i^{t-1} \rangle + \tfrac{4(M^2 + \|\delta_i^{t-1,k}\|_*^2)}{\eta_k} \right],
\end{aligned}
$$

$$\forall \mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}, \tag{5.9}$$

where

$$\hat{\mathbf{s}} := \theta_N \mathbf{L}(\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}) + \theta_1 \tau_1 (\mathbf{y}^N - \mathbf{y}^0). \tag{5.10}$$

Our result in (4.5) immediately follows from the convexity of Q. Furthermore, in view of (5.8)(c) and (5.6), we can obtain the following result,

$$\begin{aligned}\sum_{k=1}^{N} \theta_k Q(\hat{\mathbf{x}}^k, \mathbf{y}^k; \mathbf{z}) \leq\ & \theta_N \langle \mathbf{L}(\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}), \mathbf{y} - \mathbf{y}^N \rangle - \theta_N \eta_N \mathbf{V}(\mathbf{x}^{N-1}, \hat{\mathbf{x}}^N) \\ & + \tfrac{\theta_1 \tau_1}{2} \|\mathbf{y} - \mathbf{y}^0\|^2 - \tfrac{\theta_N \tau_N}{2} \|\mathbf{y} - \mathbf{y}^N\|^2 \\ & + \tfrac{(T_1+1)(T_1+2)\theta_1 \eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) - \tfrac{(T_N+1)(T_N+2)\theta_N \eta_N}{T_N(T_N+3)} \mathbf{V}(\mathbf{x}^N, \mathbf{x}) \\ & + \sum_{k=1}^{N} \sum_{t=1}^{T_k} \sum_{i=1}^{m} \tfrac{\theta_k}{T_k(T_k+3)} \\ & \left[(t+1) \langle \delta_i^{t-1,k}, x_i - u_i^{t-1} \rangle + \tfrac{4(M^2 + \|\delta_i^{t-1,k}\|_*^2)}{\eta_k} \right].\end{aligned}$$

Therefore, in view of the fact that $\sum_{k=1}^{N} \theta_k Q(\hat{\mathbf{x}}^k, \mathbf{y}^k; \mathbf{z}^*) \geq 0$ for any saddle point $\mathbf{z}^* = (\mathbf{x}^*, \mathbf{y}^*)$ of (2.4), and (2.14), by fixing $\mathbf{z} = \mathbf{z}^*$ and rearranging terms, we obtain

$$\begin{aligned}\tfrac{\theta_N \eta_N}{2} \|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2 \leq\ & \theta_N \langle \mathbf{L}(\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}), \mathbf{y}^* - \mathbf{y}^N \rangle - \tfrac{\theta_N \tau_N}{2} \|\mathbf{y}^* - \mathbf{y}^N\|^2 \\ & + \tfrac{(T_1+1)(T_1+2)\theta_1 \eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{\theta_1 \tau_1}{2} \|\mathbf{y}^* - \mathbf{y}^0\|^2 \\ & + \sum_{k=1}^{N} \sum_{t=1}^{T_k} \sum_{i=1}^{m} \tfrac{2\theta_k}{T_k(T_k+3)} \\ & \left[(t+1) \langle \delta_i^{t-1,k}, x_i^* - u_i^{t-1} \rangle + \tfrac{4(M^2 + \|\delta_i^{t-1,k}\|_*^2)}{\eta_k} \right] \\ \leq\ & \tfrac{\theta_N \|\mathbf{L}\|^2}{2\tau_N} \|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2 + \tfrac{(T_1+1)(T_1+2)\theta_1 \eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) \\ & + \tfrac{\theta_1 \tau_1}{2} \|\mathbf{y}^* - \mathbf{y}^0\|^2 + \sum_{k=1}^{N} \sum_{t=1}^{T_k} \sum_{i=1}^{m} \tfrac{2\theta_k}{T_k(T_k+3)} \\ & \left[(t+1) \langle \delta_i^{t-1,k}, x_i^* - u_i^{t-1} \rangle + \tfrac{4(M^2 + \|\delta_i^{t-1,k}\|_*^2)}{\eta_k} \right],\end{aligned} \tag{5.11}$$

where the second inequality follows from the relation $b\langle u, v \rangle - a\|v\|^2/2 \leq b^2\|u\|^2/(2a), \forall a > 0$.

Similarly, we obtain

$$\frac{\theta_N\tau_N}{2}\|\mathbf{y}^* - \mathbf{y}^N\|^2 \le \frac{\theta_N\|\mathbf{L}\|^2}{2\eta_N}\|\mathbf{y}^* - \mathbf{y}^N\|^2 + \frac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)}\mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \frac{\theta_1\tau_1}{2}\|\mathbf{y}^* - \mathbf{y}^0\|^2 + \sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m}\frac{2\theta_k}{T_k(T_k+3)} \left[(t+1)\langle\delta_i^{t-1,k}, x_i^* - u_i^{t-1}\rangle + \frac{4(M^2+\|\delta_i^{t-1,k}\|_*^2)}{\eta_k}\right], \tag{5.12}$$

from which the desired result in (4.6) follows. □

The following proof of Lemma 4 establishes the convergence of SDCS method for solving strongly convex problems.

Proof of Lemma 4 When f_i, $i = 1, \ldots, m$, are strongly convex functions, we have $\mu,\ M > 0$ (cf. (1.2)). Therefore, in view of Proposition 2 with λ_t and β_t defined in (3.34) satisfying condition (5.1), the definition of Φ^k in (3.3), and the input and output settings in the CS procedure, we have for all $k \ge 1$

$$\begin{aligned}
&\Phi^k(\hat{\mathbf{x}}^k) - \Phi^k(\mathbf{x}) \\
&\quad + \left(\sum_{t=1}^{T_k}\lambda_t\right)^{-1}\left[\eta_k(1+\beta_{T_k}^{(k)})\lambda_{T_k}\mathbf{V}(\mathbf{x}^k, \mathbf{x}) + \sum_{t=1}^{T_k}\sum_{i=1}^{m}\lambda_t\langle\delta_i^{t-1,k}, x_i - u_i^{t-1}\rangle\right] \\
&\le \left(\sum_{t=1}^{T_k}\lambda_t\right)^{-1}\left[(\eta_k\beta_1^{(k)} - \mu/\mathcal{C})\lambda_1\mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x}) + \sum_{t=1}^{T_k}\sum_{i=1}^{m}\frac{(M+\|\delta_i^{t-1,k}\|_*)^2\lambda_t}{2\eta_k\beta_t}\right], \\
&\quad \forall \mathbf{x} \in \mathbf{X}.
\end{aligned}$$

By plugging into the above relation the values of λ_t and $\beta_t^{(k)}$ in (3.34), together with the definition of Φ^k in (3.3) and rearranging the terms, we have

$$\begin{aligned}
&\langle\mathbf{L}(\hat{\mathbf{x}}^k - \mathbf{x}), \mathbf{y}^k\rangle + F(\hat{\mathbf{x}}^k) - F(\mathbf{x}) \\
&\quad \le \eta_k\mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x}) - (\mu/\mathcal{C} + \eta_k)\mathbf{V}(\mathbf{x}^k, \mathbf{x}) - \eta_k\mathbf{V}(\mathbf{x}^{k-1}, \hat{\mathbf{x}}^k) \\
&\quad + \frac{2}{T_k(T_k+1)}\sum_{t=1}^{T_k}\sum_{i=1}^{m}\left[t\langle\delta_i^{t-1,k}, x_i - u_i^{t-1}\rangle + \frac{(M+\|\delta_i^{t-1,k}\|_*)^2 t}{(t+1)\mu/\mathcal{C}+(t-1)\eta_k}\right],\ \forall \mathbf{x} \in \mathbf{X},\ k \ge 1.
\end{aligned}$$

In view of (5.5), the above relation and the definition of Q in (2.5), and following the same trick that we used to obtain (5.6), we have, for all $\mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$,

$$\sum_{k=1}^{N}\theta_k Q(\hat{\mathbf{x}}^k, \mathbf{y}^k; \mathbf{z}) \le \sum_{k=1}^{N}\theta_k\bar{\Delta}_k + \sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m}\frac{2\theta_k}{T_k(T_k+1)} \left[t\langle\delta_i^{t-1,k}, x_i - u_i^{t-1}\rangle + \frac{(M+\|\delta_i^{t-1,k}\|_*)^2 t}{(t+1)\mu/\mathcal{C}+(t-1)\eta_k}\right], \tag{5.13}$$

where

$$\begin{aligned}\bar{\varDelta}_k := \;& \mathbf{L}(\hat{\mathbf{x}}^k - \tilde{\mathbf{x}}^k), \mathbf{y} - \mathbf{y}^k\rangle + \eta_k \mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x}) - (\mu/\mathcal{C} + \eta_k)\mathbf{V}(\mathbf{x}^k, \mathbf{x}) - \eta_k \mathbf{V}(\mathbf{x}^{k-1}, \hat{\mathbf{x}}^k) \\ & + \tfrac{\tau_k}{2}\left[\|\mathbf{y} - \mathbf{y}^{k-1}\|^2 - \|\mathbf{y} - \mathbf{y}^k\|^2 - \|\mathbf{y}^{k-1} - \mathbf{y}^k\|^2\right]. \qquad (5.14)\end{aligned}$$

Since $\bar{\varDelta}_k$ in (5.14) shares a similar structure with $\varDelta_k$ in (5.7), we can follow similar procedure as in (5.8) to simplify the RHS of (5.13). Note that the only difference of (5.14) and (5.7) is in the coefficient of the terms $\mathbf{V}(\mathbf{x}^{k-1}, \mathbf{x})$, and $\mathbf{V}(\mathbf{x}^k, \mathbf{x})$. Hence, by using condition (3.33) in place of (3.13), we obtain $\forall \mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$

$$\begin{aligned}\sum_{k=1}^N \theta_k Q(\hat{\mathbf{x}}^k, \mathbf{y}^k; \mathbf{z}) \le \;& \theta_1 \eta_1 \mathbf{V}(\mathbf{x}^0, \mathbf{x}) + \tfrac{\theta_1 \tau_1}{2}\|\mathbf{y}^0\|^2 + \langle \hat{\mathbf{s}}, \mathbf{y}\rangle \\ & + \sum_{k=1}^N \sum_{t=1}^{T_k} \sum_{i=1}^m \tfrac{2\theta_k}{T_k(T_k+1)} \\ & \left[t\langle \delta_i^{t-1,k}, x_i - u_i^{t-1}\rangle + \tfrac{2t(M^2 + \|\delta_i^{t-1,k}\|_*^2)}{(t+1)\mu/\mathcal{C} + (t-1)\eta_k}\right], \qquad (5.15)\end{aligned}$$

where $\hat{\mathbf{s}}$ is defined in (5.10). Our result in (4.14) immediately follows from the convexity of Q.

Following the same procedure as we obtain (5.11), for any saddle point $\mathbf{z}^* = (\mathbf{x}^*, \mathbf{y}^*)$ of (2.4), we have

$$\begin{aligned}\tfrac{\theta_N \eta_N}{2}\|\hat{\mathbf{x}}^N - \mathbf{x}^{N-1}\|^2 \le \;& \tfrac{\theta_N \|\mathbf{L}\|^2}{2\tau_N}\|\mathbf{x}^N - \mathbf{x}^{N-1}\|^2 + \theta_1 \eta_1 \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{\theta_1 \tau_1}{2}\|\mathbf{y}^* - \mathbf{y}^0\|^2 \\ & + \sum_{k=1}^N \sum_{t=1}^{T_k} \sum_{i=1}^m \tfrac{2\theta_k}{T_k(T_k+1)} \\ & \left[t\langle \delta_i^{t-1,k}, x_i^* - u_i^{t-1}\rangle + \tfrac{2t(M^2 + \|\delta_i^{t-1,k}\|_*^2)}{(t+1)\mu/\mathcal{C} + (t-1)\eta_k}\right], \\ \tfrac{\theta_N \tau_N}{2}\|\mathbf{y}^* - \mathbf{y}^N\|^2 \le \;& \tfrac{\theta_N \|\mathbf{L}\|^2}{2\eta_N}\|\mathbf{y}^* - \mathbf{y}^N\|^2 + \theta_1 \eta_1 \mathbf{V}(\mathbf{x}^0, \mathbf{x}^*) + \tfrac{\theta_1 \tau_1}{2}\|\mathbf{y}^* - \mathbf{y}^0\|^2 \\ & + \sum_{k=1}^N \sum_{t=1}^{T_k} \sum_{i=1}^m \tfrac{2\theta_k}{T_k(T_k+1)} \\ & \left[t\langle \delta_i^{t-1,k}, x_i^* - u_i^{t-1}\rangle + \tfrac{2t(M^2 + \|\delta_i^{t-1,k}\|_*^2)}{(t+1)\mu/\mathcal{C} + (t-1)\eta_k}\right], \qquad (5.16)\end{aligned}$$

from which the desired result in (4.15) follows. □

We are ready to provide proofs for Lemmas 1 and 2, which demonstrates the convergence properties of the deterministic communication sliding method.

Proof of Lemma 1 When $f_i,\ i = 1, \ldots, m$ are general nonsmooth convex functions, we have $\delta_i^t = 0, \mu = 0$ and $M > 0$. Therefore, in view of (5.9), we have, $\forall \mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$,

$$\sum_{k=1}^{N} \theta_k Q(\hat{\mathbf{x}}^k, \mathbf{y}^k; \mathbf{z}) \le \tfrac{(T_1+1)(T_1+2)\theta_1\eta_1}{T_1(T_1+3)} \mathbf{V}(\mathbf{x}^0, \mathbf{x}) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2 + \langle \hat{\mathbf{s}}, \mathbf{y} \rangle$$
$$+ \sum_{k=1}^{N} \tfrac{4mM^2\theta_k}{(T_k+3)\eta_k},$$

where $\hat{\mathbf{s}}$ is defined in (5.10). Our result in (3.19) immediately follows from the convexity of Q. Moreover, our result in (3.20) follows from setting $\delta_i^{t-1,k} = 0$ in (5.11) and (5.12). □

Proof of Lemma 2 When $f_i,\ i = 1, \ldots, m$ are strongly convex functions, we have $\delta_i^t = 0$ and $\mu,\ M > 0$. Therefore, in view of (5.15), we obtain, $\forall \mathbf{z} \in \mathbf{X} \times \mathbb{R}^{md}$,

$$\sum_{k=1}^{N} \theta_k Q(\hat{\mathbf{x}}^k, \mathbf{y}^k; \mathbf{z}) \le \theta_1\eta_1 \mathbf{V}(\mathbf{x}^0, \mathbf{x}) + \tfrac{\theta_1\tau_1}{2}\|\mathbf{y}^0\|^2 + \langle \hat{\mathbf{s}}, \mathbf{y} \rangle$$
$$+ \sum_{k=1}^{N} \sum_{t=1}^{T_k} \tfrac{2mM^2\theta_k}{T_k(T_k+1)} \tfrac{t}{(t+1)\mu/\mathcal{C}+(t-1)\eta_k},$$

where $\hat{\mathbf{s}}$ is defined in (5.10). Our result in (3.35) immediately follows from the convexity of Q. Also, the result in (3.36) follows by setting $\delta_i^{t-1,k} = 0$ in (5.16). □

We now provide a proof for Theorem 6 that establishes a large deviation result for the gap function.

Proof of Theorem 6 Observe that by Assumption (4.1), (4.2) and (4.23) on the SO and the definition of $u_i^{t,k}$, the sequence $\{\langle \delta_i^{t-1,k}, x_i^* - u_i^{t-1,k} \rangle\}_{1 \le i \le m, 1 \le t \le T_k, k \ge 1}$ is a martingale-difference sequence. Denoting

$$\gamma_{k,t} := \tfrac{\theta_k \lambda_t}{\sum_{t=1}^{T_k} \lambda_t},$$

and using the large-deviation theorem for martingale-difference sequence (e.g. Lemma 2 of [29]) and the fact that

$$\begin{aligned}
&\mathbb{E}\left[\exp\left\{\gamma_{k,t}^2 \langle \delta_i^{t-1,k}, x_i^* - u_i^{t-1,k} \rangle^2 / (2\gamma_{k,t}^2 \bar{V}_i(x_i^*)\sigma^2)\right\}\right] \\
&\le \mathbb{E}\left[\exp\left\{\left\|\delta_i^{t-1,k}\right\|_*^2, \left\|x_i^* - u_i^{t-1,k}\right\|^2 / (2\bar{V}_i(x_i^*)\sigma^2)\right\}\right] \\
&\le \mathbb{E}\left[\exp\left\{\left\|\delta_i^{t-1,k}\right\|_*^2 / \sigma^2\right\}\right] \le \exp\{1\},
\end{aligned}$$

we conclude that, $\forall \zeta > 0$,

$$\text{Prob}\left\{\sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m}\gamma_{k,t}\langle\delta_i^{t-1,k}, u_i^{t-1,k} - x_i^*\rangle > \zeta\sigma\sqrt{2\bar{\mathbf{V}}(\mathbf{x}^*)\sum_{k=1}^{N}\sum_{t=1}^{T_k}\gamma_{k,t}^2}\right\} \\ \leq \exp\{-\zeta^2/3\}. \tag{5.17}$$

Now let

$$S_{k,t} := \frac{\theta_k\lambda_t}{\left(\sum_{t=1}^{T_k}\lambda_t\right)\eta_k\beta_t},$$

and $S := \sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m} S_{k,t}$. By the convexity of exponential function, we have

$$\mathbb{E}\left[\exp\left\{\tfrac{1}{S}\sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m} S_{k,t}\|\delta_i^{t-1,k}\|_*^2/\sigma^2\right\}\right] \\ \leq \mathbb{E}\left[\tfrac{1}{S}\sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m} S_{k,t}\exp\left\{\left\|\delta_i^{t-1,k}\right\|_*^2/\sigma^2\right\}\right] \leq \exp\{1\},$$

where the last inequality follows from Assumption (4.23). Therefore, by Markov's inequality, for all $\zeta > 0$,

$$\text{Prob}\left\{\sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m} S_{k,t}\|\delta_i^{t-1,k}\|_*^2 > (1+\zeta)\sigma^2\sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m} S_{k,t}\right\} \\ = \text{Prob}\left\{\exp\left\{\tfrac{1}{S}\sum_{k=1}^{N}\sum_{t=1}^{T_k}\sum_{i=1}^{m} S_{k,t}\|\delta_i^{t-1,k}\|_*^2/\sigma^2\right\} \geq \exp\{1+\zeta\}\right\} \\ \leq \exp\{-\zeta\}. \tag{5.18}$$

Combing (5.17), (5.18), (4.5) and (2.7), our result in (4.25) immediately follows. □

6 Numerical results

In this section, we demonstrate the advantages of our (stochastic) decentralized communication sliding method over distributed dual averaging method proposed in [17] through some preliminary numerical experiments.

Let us consider the decentralized linear Support Vector Machines (SVM) model with the following hinge loss function

$$\max\{0, 1 - v\langle x, u\rangle\}, \tag{6.1}$$

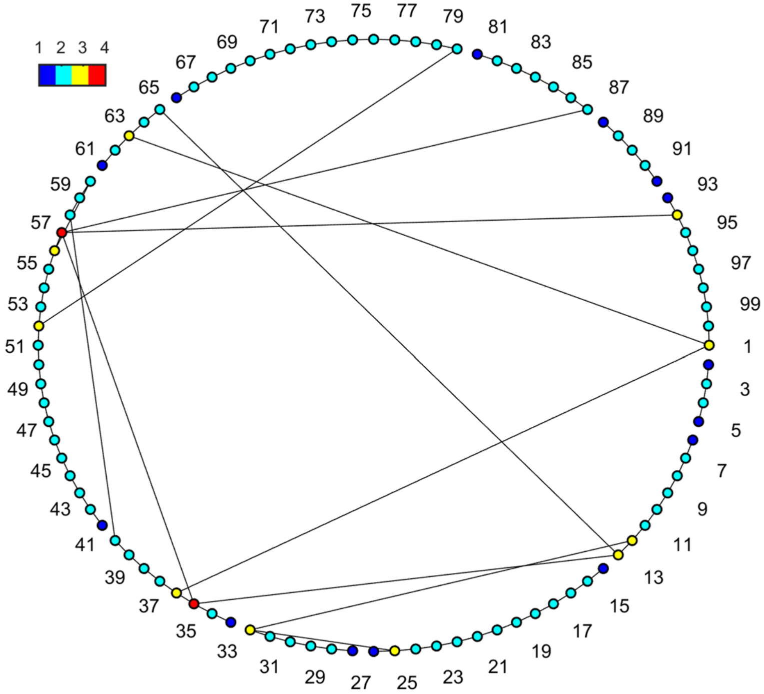

Fig. 1 The underlying decentralized network

where $(v, u) \in \mathbb{R} \times \mathbb{R}^d$ is the pair of class label and feature vector, and $x \in \mathbb{R}^d$ denotes the weight vector. Clearly, the hinge loss function is convex and nonsmooth with respect to x. For convex case, we study 1-norm SVM problem [7,67], i.e., the hinge loss function (6.1) plus $l1$-norm as the regularizer, while for strongly convex case, we study 2-norm SVM model. Moreover, we use the Erhos-Renyi algorithm[5] to generate the underlying decentralized network, i.e., a connected graph with $m = 100$ nodes shown in Fig. 1. Note that nodes with different degrees are drawn in different colors, in particular, nodes in red have maximum degree of 4. We also used the real dataset named "ijcnn1" from LIBSVM[6] and choose 20,000 samples from this dataset as our problem instance data to train the decentralized SVM model. Since we have $m = 100$ nodes (or agents) in the decentralized network, we evenly split these 20,000 samples over 100 nodes, and hence each network node has 200 samples.

With the same initial points $x^0 = \mathbf{1}$ and $y^0 = \mathbf{0}$, we compare the performances of our algorithms with the distributed dual averaging method [17] for solving (1.1)–(2.3) by showing the progress of objective values versus the number of communication rounds and subgradient evaluations (i.e. the number of sampling data) for three different types of problems. In all problem instances, we use $\|\cdot\|_2$ norm in both the primal and dual

[5] We implemented the Erhos-Renyi algorithm based on a MATLAB function written by Pablo Blider, which can be found in https://www.mathworks.com/matlabcentral/fileexchange/4206.

[6] This real dataset can be downloaded from https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/.

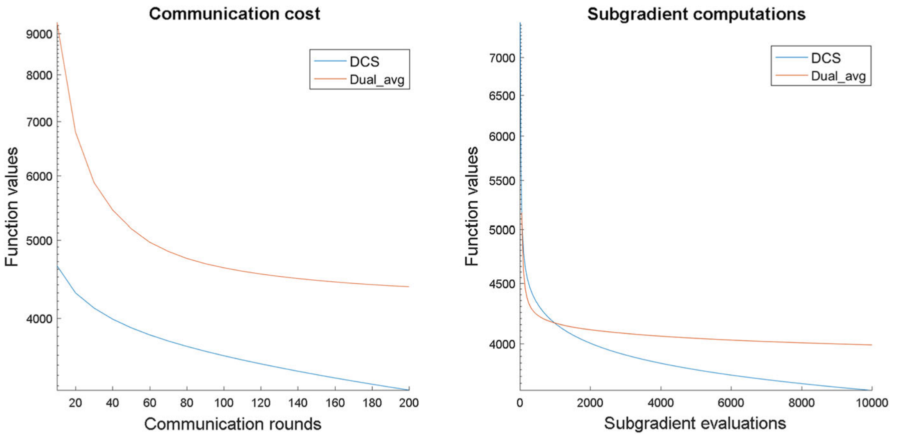

Fig. 2 The comparison of the DCS method with distributed dual averaging method for solving (6.2)

spaces, and hence in the parameter settings of DCS/SDCS $\|\mathbf{L}\|$ refers to the maximum eigenvalue of the Laplacian matrix L.

- *Deterministic convex problems* The decentralized linear SVM problem under the aforementioned network can be written as

$$\begin{aligned} \min_{\mathbf{x}} \; & \sum_{i=1}^{m} \left[f_i(x_i) := \sum_{(v_j,u_j)\in\mathcal{S}_i} \max\{0, 1 - v_j\langle x_i, u_j\rangle\} + \tfrac{1}{\|\mathcal{S}_i\|}\|x_i\|_1 \right] \\ & \text{s.t. } \; \mathbf{Lx} = \mathbf{0}, \end{aligned} \tag{6.2}$$

  where $\mathcal{S}_i$ denotes the dataset belonging to node i. Since the problem is deterministic and convex, we choose the parameters of the DCS method as suggested in Theorem 1, and set the inner iteration limit as $\min(10k, T_k)$ to illustrate the possibility of choosing inner iteration limit dynamically in practice. It needs to be pointed out that if we use a constant inner iteration limit as stated in Theorem 1, we can obtain similar results as shown in Fig. 2, but with a slightly slower convergence speed at the very beginning for the DCS method. For distributed dual averaging method, we choose the stepsize in the order of $\mathcal{O}(1/\sqrt{t})$ as suggested in [17].
  In Fig. 2, the vertical-axis represents the objective values, the horizontal-axis of the left subgraph is the number of inter-node communication rounds, and the horizontal-axis of the right subgraph is the number of intra-node subgradient evaluations. These numerical results are consistent with our theoretic results in that DCS significantly reduces the total number of inter-node communication rounds while still maintaining comparable bounds on the intra-node subgradient evaluations for solving (6.2).

- *Stochastic convex problems* We now consider a stochastic decentralized linear SVM problem under the aforementioned network as

$$\begin{aligned}\min_{\mathbf{x}} \sum_{i=1}^{m} &\left[f_i(x_i) := \mathbb{E}_{(v_i,u_i)}[\max\{0, 1 - v_i\langle x_i, u_i\rangle\}] + \tfrac{1}{\|\mathcal{S}_i\|}\|x_i\|_1 \right] \\ &\text{s.t.}\ \ \mathbf{Lx} = \mathbf{0}, \end{aligned} \tag{6.3}$$

  where (v_i, u_i) represents a uniform random variable with support $\mathcal{S}_i$. For stochastic decentralized communication sliding (SDCS) method, we choose parameters according to Theorem 4, and also set inner iteration limit as in the deterministic convex case. For distributed dual averaging method, we choose the same stepsize as suggested in [17].

  Figure 3 clearly shows that SDCS also saves inter-node communication rounds comparing to distributed dual averaging method while preserving the same order of sampling complexity for solving (6.3).
- *Stochastic strongly convex problems* Consider a decentralized linear SVM problem with l_2 regularizer under the aforementioned network as the following

$$\begin{aligned}\min_{\mathbf{x}} \ \sum_{i=1}^{m} &\left[f_i(x_i) := \mathbb{E}_{(v_i,u_i)}[\max\{0, 1 - v_i\langle x_i, u_i\rangle\}] + \tfrac{1}{2|\mathcal{S}_i|}\|x_i\|_2^2 \right] \\ &\text{s.t.}\ \ \mathbf{Lx} = \mathbf{0}. \end{aligned} \tag{6.4}$$

  Since f_i's in (6.4) are strongly convex, we choose the parameters of the SDCS method as suggested in Theorem 5 and fix the inner iteration limit $T_k = 10^4$, which is a rough estimate of the suggested T_k in Theorem 5. For distributed dual averaging method, we choose stepsize in the order of $\mathcal{O}(1/t)$ as suggested in [16] instead of [17]. This is because [17] did not cover strongly convex problems, while [16] extended the dual averaging method to solve strongly convex problems.

  Figure 4 shows that for stochastic strongly convex problems defined in (6.4), the SDCS method can achieve better performance than distributed dual averaging method in terms of both the number of communication rounds and subgradient computations. It should be pointed out that SDCS appears to be worse than distributed dual averaging method at the very beginning of the right subgraph because too few communication rounds are performed by SDCS at that time point, which provides little information about the loss function $F(\mathbf{x})$. However, as the number of communication rounds increases, SDCS gradually outperforms distributed dual averaging method in terms of the objective values. We can also observe similar phenomena in the first two experiments.

In addition to the objective value, we also report the progress of feasibility residual, $\|\mathbf{Lx}\|$, versus communication rounds in Fig. 5. It needs to be mentioned that the distributed dual averaging method [17] does not measure feasibility residual since it sets the final output to be the average of iterates obtained by one of the network agents,[7]

[7] We choose the average of iterates obtained by the first agent as the output solution for distributed dual averaging method in all three problems.

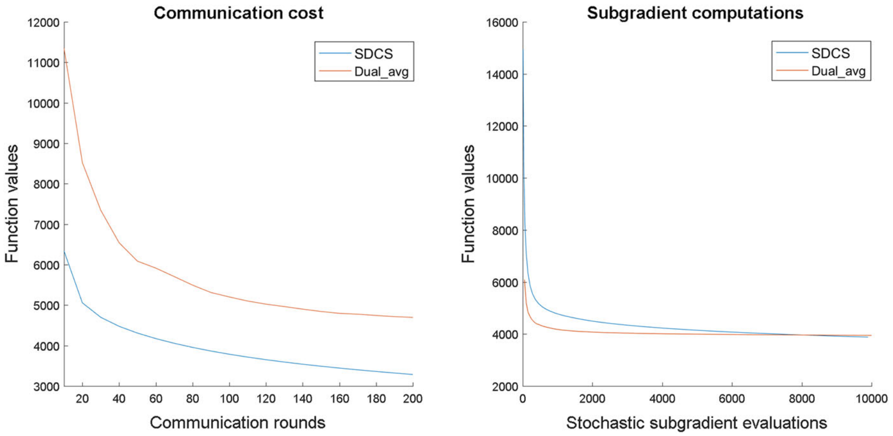

Fig. 3 The comparison of the SDCS method with distributed dual averaging method for solving (6.3)

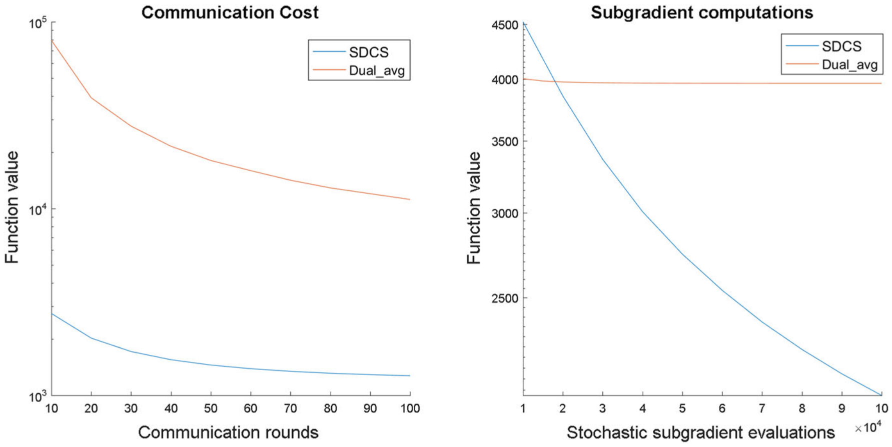

Fig. 4 The comparison of the SDCS method with distributed dual averaging method for solving (6.3)

and hence requires more rounds of communication to broadcast its final output to all agents, which we do not include in all comparisons.

7 Concluding remarks

In this paper, we present a new class of decentralized primal-dual methods which can significantly reduce the number of inter-node communications required to solve the distributed optimization problem in (1.1). More specifically, we show that by using these algorithms, the total number of communication rounds can be significantly reduced to $\mathcal{O}(1/\epsilon)$ when the objective functions f_i's are convex and not necessarily smooth. By properly designing the communication sliding algorithms, we demonstrate that the $\mathcal{O}(1/\epsilon)$ number of communications can still be maintained for general convex

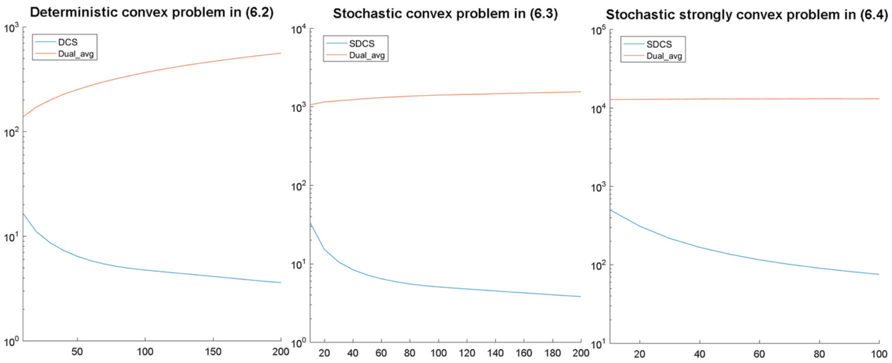

Fig. 5 The progress of feasibility residuals $\|\mathbf{Lx}\|$ versus communication rounds

objective functions (and it can be further reduced to $\mathcal{O}(1/\sqrt{\epsilon})$ for strongly convex objective functions) even if the local subproblems are solved inexactly through iterative procedure (cf. CS procedure) by the network agents. In this case, the number of intra-node subgradient computations that we need will be bounded by $\mathcal{O}(1/\epsilon^2)$ (resp., $\mathcal{O}(1/\epsilon)$) when the objective functions f_i's are convex (resp., strongly convex), which is comparable to that required in centralized nonsmooth optimization and not improvable in general. We also establish similar complexity bounds for solving stochastic decentralized optimization counterpart by developing the stochastic communication sliding methods, which can provide communication-efficient ways to deal with streaming data and decentralized statistical inference. As illustrated in our preliminary numerical experiments, all these decentralized communication sliding algorithms have the potential to significantly increase the performance of multiagent systems, where the bottleneck exists in the communication.

References

1. Arrow, K., Hurwicz, L., Uzawa, H.: Studies in Linear and Non-linear Programming. Stanford Mathematical Studies in the Social Sciences. Stanford University Press, Stanford (1958)
2. Aybat, N.S., Hamedani, E.Y.: A primal-dual method for conic constrained distributed optimization problems. In: Advances in Neural Information Processing Systems, pp. 5049–5057 (2016)
3. Aybat, N.S., Wang, Z., Lin, T., Ma, S.: Distributed linearized alternating direction method of multipliers for composite convex consensus optimization. IEEE Trans. Autom. Control **63**(1), 5–20 (2018)
4. Bertsekas, D.P.: Incremental proximal methods for large scale convex optimization. Math. Program. **129**, 163–195 (2011)
5. Bertsekas, D.P.: Incremental aggregated proximal and augmented lagrangian algorithms. Technical Report LIDS-P-3176, Laboratory for Information and Decision Systems (2015)
6. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends Mach. Learn. **3**(1), 1–122 (2011)
7. Bradley, Paul S., Mangasarian, O.L.: Feature selection via concave minimization and support vector machines. In: ICML, vol. 98, pp. 82–90 (1998)
8. Bregman, L.M.: The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. USSR Comput. Math. Math. Phys. **7**(3), 200–217 (1967)

9. Chambolle, A., Pock, T.: On the ergodic convergence rates of a first-order primal-dual algorithm. Math. Program. **159**(1–2), 253–287 (2016)
10. Chambolle, A., Pock, T.: A first-order primal–dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vis. **40**(1), 120–145 (2011)
11. Chang, T., Hong, M.: Stochastic proximal gradient consensus over random networks. arxiv:1511.08905 (2015)
12. Chang, T., Hong, M., Wang, X.: Multi-agent distributed optimization via inexact consensus admm. arxiv:1402.6065 (2014)
13. Chen, A., Ozdaglar, A.: A fast distributed proximal gradient method. In: 2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton), pp. 601–608 (2012)
14. Chen, Y., Lan, G., Ouyang, Y.: Optimal primal-dual methods for a class of saddle point problems. SIAM J. Optim. **24**(4), 1779–1814 (2014)
15. Dang, C., Lan, G.: Randomized first-order methods for saddle point optimization. Technical Report 32611, Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL (2015)
16. Deng, Q., Lan, G., Rangarajan, A.: Randomized block subgradient methods for convex nonsmooth and stochastic optimization. arXiv preprint arXiv:1509.04609 (2015)
17. Duchi, J., Agarwal, A., Wainwright, M.: Dual averaging for distributed optimization: convergence analysis and network scaling. IEEE Trans. Autom. Control **57**(3), 592–606 (2012)
18. Durham, J.W., Franchi, A., Bullo, F.: Distributed pursuit-evasion without mapping or global localization via local frontiers. Auton. Robots **32**(1), 81–95 (2012)
19. Ghadimi, S., Lan, G.: Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization I: a generic algorithmic framework. SIAM J. Optim. **22**(4), 1469–1492 (2012)
20. Ghadimi, S., Lan, G.: Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, II: shrinking procedures and optimal algorithms. SIAM J. Optim. **23**(4), 2061–2089 (2013)
21. Gurbuzbalaban, M., Ozdaglar, A., Parrilo, P.: On the convergence rate of incremental aggregated gradient algorithms. arxiv:1506.02081 (2015)
22. He, B., Yuan, X.: On the o(1/n) convergence rate of the Douglas–Rachford alternating direction method. SIAM J. Numer. Anal. **50**(2), 700–709 (2012)
23. He, N., Juditsky, A., Nemirovski, A.: Mirror prox algorithm for multi-term composite minimization and semi-separable problems. J. Comput. Optim. Appl. **103**, 127–152 (2015)
24. Hom, R.A., Johnson, C.R.: Topics in Matrix Analysis. Cambridge UP, New York (1991)
25. Jadbabaie, A., Lin, J., Morse, A.S.: Coordination of groups of mobile autonomous agents using nearest neighbor rules. IEEE Trans. Autom. Control **48**(6), 988–1001 (2003)
26. Jakovetic, D., Xavier, J., Moura, J.: Fast distributed gradient methods. IEEE Trans. Autom. Control **59**(5), 1131–1145 (2014)
27. Lan, G.: An optimal method for stochastic composite optimization. Math. Program. **133**(1), 365–397 (2012)
28. Lan, G.: Gradient sliding for composite optimization. Math. Program. **159**(1), 201–235 (2016)
29. Lan, G., Nemirovski, A., Shapiro, A.: Validation analysis of mirror descent stochastic approximation method. Math. Program. **134**(2), 425–458 (2012)
30. Lan, G., Zhou, Y.: An optimal randomized incremental gradient method. arxiv:1507.02000 (2015)
31. Luenberger, D.G., Ye, Y., et al.: Linear and Nonlinear Programming, vol. 2. Springer, Berlin (1984)
32. Makhdoumi, A., Ozdaglar, A.: Convergence rate of distributed admm over networks. arxiv:1601.00194 (2016)
33. Mokhtari, A., Shi, W., Ling, Q., Ribeiro, A.: Dqm: Decentralized quadratically approximated alternating direction method of multipliers. arxiv:1508.02073 (2015)
34. Mokhtari, A., Shi, W., Ling, Q., Ribeiro, A.: A decentralized second-order method with exact linear convergence rate for consensus optimization. arxiv:1602.00596 (2016)
35. Monteiro, R.D.C., Svaiter, B.F.: On the complexity of the hybrid proximal extragradient method for the iterates and the ergodic mean. SIAM J. Optim. **20**(6), 2755–2787 (2010)
36. Monteiro, R.D.C., Svaiter, B.F.: Complexity of variants of Tseng's modified F-B splitting and korpelevich's methods for hemivariational inequalities with applications to saddle-point and convex optimization problems. SIAM J. Optim. **21**(4), 1688–1720 (2011)
37. Monteiro, R.D.C., Svaiter, B.F.: Iteration-complexity of block-decomposition algorithms and the alternating direction method of multipliers. SIAM J. Optim. **23**(1), 475–507 (2013)

38. Monteiro, R.D.C., Svaiter, B.F.: On the complexity of the hybrid proximal projection method for the iterates and the ergodic mean. SIAM J. Optim. **20**, 2755–2787 (2010)
39. Nedić, A.: Asynchronous broadcast-based convex optimization over a network. IEEE Trans. Autom. Control **56**(6), 1337–1351 (2011)
40. Nedić, A., Bertsekas, D.P., Borkar, V.S.: Distributed asynchronous incremental subgradient methods. Inherently Parallel Algorithms in Feasibility and Optimization and Their Applications, pp. 311–407 (2001)
41. Nedić, A., Olshevsky, A.: Distributed optimization over time-varying directed graphs. IEEE Trans. Autom. Control **60**(3), 601–615 (2015)
42. Nedić, A., Olshevsky, A., Shi, W.: Achieving geometric convergence for distributed optimization over time-varying graphs. arxiv:1607.03218 (2016)
43. Nedić, A., Ozdaglar, A.: Distributed subgradient methods for multi-agent optimization. IEEE Trans. Autom. Control **54**(1), 48–61 (2009)
44. Nemirovski, A.S.: Prox-method with rate of convergence $o(1/t)$ for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. SIAM J. Optim. **15**, 229–251 (2005)
45. Nemirovski, A.S., Juditsky, A., Lan, G., Shapiro, A.: Robust stochastic approximation approach to stochastic programming. SIAM J. Optim. **19**, 1574–1609 (2009)
46. Nemirovski, A.S., Yudin, D.: Problem complexity and method efficiency in optimization. Wiley-Interscience Series in Discrete Mathematics. Wiley, XV (1983)
47. Nesterov, Y.E.: Smooth minimization of nonsmooth functions. Math. Program. **61**(2), 275–319 (2015)
48. Ouyang, Y., Chen, Y., Lan, G., Pasiliao Jr., E.: An accelerated linearized alternating direction method of multipliers. SIAM J. Imaging Sci. **8**(1), 644–681 (2015)
49. Qu, G., Li, N.: Harnessing smoothness to accelerate distributed optimization. arxiv:1605.07112 (2016)
50. Rabbat, M.: Multi-agent mirror descent for decentralized stochastic optimization. In: 2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), pp. 517–520 (2015)
51. Rabbat, M., Nowak, R.D.: Distributed optimization in sensor networks. In: IPSN, pp. 20–27 (2004)
52. Ram, S.S., Nedić, A., Veeravalli, V.V.: Incremental stochastic subgradient algorithms for convex optimization. SIAM J. Optim. **20**(2), 691–717 (2009)
53. Ram, S.S., Nedić, A., Veeravalli, V.V.: Distributed stochastic subgradient projection algorithms for convex optimization. J. Optim. Theory Appl. **147**, 516–545 (2010)
54. Ram, S.S., Veeravalli, V.V., Nedić, A.: Distributed non-autonomous power control through distributed convex optimization. In: IEEE INFOCOM, pp. 3001–3005 (2009)
55. Shi, W., Ling, Q., Wu, G., Yin, W.: On the linear convergence of the ADMM in decentralized consensus optimization. IEEE Trans. Sig. Process. **62**(7), 1750–1761 (2014)
56. 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)
57. Shi, W., Ling, Q., Wu, G., Yin, W.: A proximal gradient algorithm for decentralized composite optimization. IEEE Trans. Sig. Process. **63**(22), 6013–6023 (2015)
58. Simonetto, A., Kester, L., Leus, G.: Distributed time-varying stochastic optimization and utility-based communication. arxiv:1408.5294 (2014)
59. Terelius, H., Topcu, U., Murray, R.: Decentralized multi-agent optimization via dual decomposition. IFAC Proc. Vol. **44**(1), 11245–11251 (2011)
60. Tsianos, K., Lawlor, S., Rabbat, M.: Consensus-based distributed optimization: practical issues and applications in large-scale machine learning. In: Proceedings of the 50th Allerton Conference on Communication, Control, and Computing (2012)
61. Tsianos, K., Rabbat, M.: Consensus-based distributed online prediction and optimization. In: 2013 IEEE Global Conference on Signal and Information Processing, pp. 807–810 (2013)
62. Tsitsiklis, J., Bertsekas, D., Athans, M.: Distributed asynchronous deterministic and stochastic gradient optimization algorithms. IEEE Trans. Autom. Control **31**(9), 803–812 (1986)
63. Tsitsiklis, J.N.: Problems in Decentralized Decision Making and Computation. Ph.D. thesis, Massachusetts Inst. Technol., Cambridge, MA (1984)
64. Wang, M., Bertsekas, D.P.: Incremental constraint projection-proximal methods for nonsmooth convex optimization. Technical Report LIDS-P-2907, Laboratory for Information and Decision Systems (2013)
65. Wei, E., Ozdaglar, A.: On the $O(1/k)$ convergence of asynchronous distributed alternating direction method of multipliers. arxiv:1307.8254 (2013)

66. Xi, C., Wu, Q., Khan, U.A.: Distributed mirror descent over directed graphs. arxiv:1412.5526 (2014)
67. Zhu, J., Rosset, S., Tibshirani, R., Hastie, T.J.: 1-norm support vector machines. In: Advances in neural information processing systems, pp. 49–56 (2004)
68. 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.

Affiliations

Guanghui Lan[1] · Soomin Lee[1] · Yi Zhou[1]

Soomin Lee
soomin.lee@isye.gatech.edu

Yi Zhou
yizhou@gatech.edu

[1] Department of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA