J Optim Theory Appl (2016) 168:153–171
DOI 10.1007/s10957-015-0757-1

An Inexact Dual Fast Gradient-Projection Method for Separable Convex Optimization with Linear Coupled Constraints

Jueyou Li[1] · **Zhiyou Wu**[1] · **Changzhi Wu**[2] · **Qiang Long**[4] · **Xiangyu Wang**[2,3]

Received: 14 October 2014 / Accepted: 12 May 2015 / Published online: 23 May 2015

Abstract In this paper, a class of separable convex optimization problems with linear coupled constraints is studied. According to the Lagrangian duality, the linear coupled constraints are appended to the objective function. Then, a fast gradient-projection method is introduced to update the Lagrangian multiplier, and an inexact solution method is proposed to solve the inner problems. The advantage of our proposed method is that the inner problems can be solved in an inexact and parallel manner. The established convergence results show that our proposed algorithm still achieves optimal convergence rate even though the inner problems are solved inexactly. Finally, several

Communicated by Kok Lay Teo.

✉ Changzhi Wu
C.Wu@curtin.edu.au

Jueyou Li
lijueyou@163.com

Zhiyou Wu
zywu@cqnu.edu.cn

Qiang Long
longqiang850407@gmail.com

Xiangyu Wang
xiangyu.wang@curtin.edu.au

1 School of Mathematical Sciences, Chongqing Normal University, Chongqing 400047, China

2 Australasian Joint Research Centre for Building Information Modelling, School of Built Environment, Curtin University, Perth 6102, Australia

3 Department of Housing and Interior Design, Kyung Hee University, Seoul 130701, Korea

4 School of Science, Southwest University of Science and Technology, Mianyang 621010, Sichuan, China

numerical experiments are presented to illustrate the efficiency and effectiveness of our proposed algorithm.

Keywords Convex optimization · Dual decomposition · Inexact gradient method · Suboptimality and constraint violations

Mathematics Subject Classification 90C25 · 90C46 · 65Y05

1 Introduction

In recent years, there has been a trend to solve large-scaled convex optimization problems with separable objective function and linear coupled constraints [1–4]. This class of optimization problems has extensive practical applications, such as the optimization problems in networks [5], transportation [6], model predictive control [7,8], distributed estimation and multistage stochastic optimization [9–11].

Several approaches are proposed to solve these kinds of optimization problems in the literature, such as Lagrangian relaxation-based methods [12], alternating direction methods [13] and interior point methods [14]. However, most of these methods cannot be implemented in parallel and are suffered from slow convergence. In order to solve the original problem in parallel, dual decomposition was introduced in [3]. The dual decomposition consists of decomposing the primal problem in the dual space. More precisely, for each of the given Lagrangian multipliers, only primal subproblems with simple constraints are required to be solved. Then, the Lagrangian multipliers are updated through (sub)gradient-based methods. The convergence rate has also been established [3]. In [8], Nesterov's fast gradient method [15] was further introduced to accelerate the convergence rate of the dual problem. In [16], an algorithm based on the combination of Lagrangian dual decomposition, excessive gap and smoothing together to solve non-smooth separable convex optimization problems with linear coupled constraints was developed. The convergence rate was also established. A drawback for all these methods is a consequence of the assumption that the solution of the inner primal subproblems is exact. However, it is impossible to find an exact solution for the inner primal subproblems, if they cannot be solved analytically.

Fast proximal-gradient methods, first proposed by Nesterov [17] for minimizing smooth convex functions, later extended by Beck et al. [1] for minimizing composite convex objective functions, and surveyed by Tseng [18,19] in a unifying manner, have proven to be very efficient in solving several classes of large-scaled convex optimization problems. The methods have better convergence rates over classical gradient-projection methods. In particular, the methods usually show good practical performance on solving problems with special structures. Recently, the authors in [20,21] have extended Nesterov's fast gradient methods to the inexact setting for solving general convex optimization problems with complex constraints. Although the proposed fast dual gradient methods use inexact first-order information, they have provided much better estimates for the convergence rates and constraint violations. In [22], the fast proximal-gradient method was extended to solve large-scaled convex quadratic semi-definite programming problems, where the subproblem in each iteration was solved only approximately. The results showed that the method there shared the same convergence rate as the exact counterpart, if the subproblems were solved within prefixed accuracy. Recently, extensive results in both theory and empirical performance for fast proximal-gradient methods have been published [23–25]. However, none of them focus on the optimization problems with separable objective function and coupled constraints.

In this paper, we will extend fast proximal-gradient methods to solve separable convex optimization problems with linear coupled constraints. The proposed method is based on the dual decomposition, the inexact first-order oracle [26] and fast proximal-gradient methods [18]. Different from existing dual decomposition methods, we will solve the inner primal subproblems in an inexact manner. Our work is closely related to the recent papers [8, 18, 20]. By contrast, our algorithm can be viewed as a generalized version of the algorithm in [8] since our method allows us to solve inner subproblems inexactly. Our work is also close to the work [20] that has proposed an inexact dual fast gradient method for solving convex problems with complex convex constraints. However, in essence, the fast gradient method we use in this paper is different from that in [20], since our method only uses the last inexact gradient, while the method in [20] involves the accumulated history of all past inexact gradients. For more differences between two algorithms, please refer to [18]. Moreover, the convergence analysis for two algorithms is different since our problem is separable with linear coupled constraints. In addition, during the solution of the inner subproblems, we utilize a simple fast gradient method in [17], while the authors in [20] utilize a coordinate descent method.

2 Problem Statement and Dual Decomposition

In this section, we first introduce the problem that we will solve and then recall briefly the Lagrangian dual decomposition for separable convex problems (1). Before this, we require the following notations:

The standard inner product of two vectors $x, y \in \mathbb{R}^n$ is denoted as $\langle x, y\rangle := x^T y$, where the subscript "$T$" denotes the transpose. For $x \in \mathbb{R}^n$, its Euclidean norm is $||x|| := \sqrt{\langle x, x\rangle}$. Denote l_1 and l_∞ norm by $||x||_1 := \sum_i^n |x_i|$ and $||x||_\infty := \max_i |x_i|$, respectively. Let $\mathbb{R}^n_+$ represent the nonnegative orthant and denote by $[x]_+$ the projection of $x \in \mathbb{R}^n$ onto $\mathbb{R}^n_+$. Denote $\mathrm{P}_X[y] := \arg\min_{x\in X} ||x - y||,\ \forall y \in \mathbb{R}^n$, be the Euclidean projection operator on X. For any real number a, $\lfloor a \rfloor$ denotes the largest integer, which is less than or equal to a. For a matrix $A \in \mathbb{R}^{m\times n}$, we introduce the notation $||A|| := \max_{||y||=||x||=1} \langle y, Ax\rangle,\ x \in \mathbb{R}^n, y \in \mathbb{R}^m$.

For a continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}$, iff there exists a constant $\sigma > 0$ such that $f(x) \geq f(y) + \nabla f(y)^T(x - y) + \frac{\sigma}{2}||x - y||^2$ for all $x, y \in \mathbb{R}^n$, then we say that f is σ-strongly convex on $\mathbb{R}^n$ with respect to the norm $||\cdot||$. For a continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}$, iff there exists a constant $L > 0$ such that $||\nabla f(x) - \nabla f(y)|| \leq L||x - y||$ for all $x, y \in \mathbb{R}^n$, then we say that the gradient of f is L-Lipschitz continuous on $\mathbb{R}^n$ with respect to the norm $||\cdot||$. For any $\epsilon \in [0, 1]$, we say that a quantity q is of order $\mathcal{O}(r(\epsilon))$ iff there exists a constant $c > 0$ such that $q \leq cr(\epsilon)$.

2.1 Problem Statement

Convex optimization problems with a separable objective function subject to linear coupled constraints are often encountered in many disciplines. They are known as separable convex optimization problems, which can be written as follows:

$$\min_x g(x) := \sum_{i=1}^{N} g_i(x_i) \ \text{ s.t. } \sum_{i=1}^{N} A_i x_i \le b, \quad x_i \in X_i \quad i = 1, \ldots, N, \tag{1}$$

where $g_i: \mathbb{R}^{n_i} \to \mathbb{R}$, $i = 1, \ldots, N$, are convex functions, $X_i \subset \mathbb{R}^{n_i}$, $i = 1, \ldots, N$, are nonempty closed convex sets, $x = (x_1^T, \ldots, x_N^T)^T$, $x_i \in \mathbb{R}^{n_i}$, $i = 1, \ldots, N$, $n_1 + n_2 + \ldots + n_N = n$, and $A_i \in \mathbb{R}^{m \times n_i}$, $b \in \mathbb{R}^m$.

2.2 Lagrangian Dual Decomposition

Denote $X := \prod_{i=1}^{N} X_i$. Let X^* and g^* be the optimal solution set and the optimal value of problem (1), respectively. Denote $D_i := \max_{x,y \in X_i} ||x - y||$ as the diameter of the set X_i. For $i = 1, \ldots, N$, we suppose that X_i is simple, i.e., the projection onto this set can be computed easily. Let $ri(X)$ be the relative interior of the convex set X, problem (1) is said to satisfy the *Slater's condition* [12], iff $ri(X) \cap \{x \in \mathbb{R}^n | \sum_{i=1}^{N} A_i x_i \le b\} \neq \varnothing$. Throughout this paper, we make the following assumption:

Assumption 1 The optimal solution set X^* is nonempty, and the Slater's condition of problem (1) holds. For each $i = 1, \ldots, N$, the function g_i is σ_i-strongly convex and continuously differentiable in $\mathbb{R}^{n_i}$.

Note that if the strong convexity assumption on the function g_i does not hold, then we can apply Nesterov's smoothing technique [15] by adding a prox-function term to g_i in order to obtain a strongly convex approximation of g_i.

The Lagrangian function for problem (1) is

$$\mathscr{L}(x; \lambda) := \sum_{i=1}^{N} g_i(x_i) + \lambda^T \left(\sum_{i=1}^{N} A_i x_i - b \right) = \sum_{i=1}^{N} \mathscr{L}_i(x_i; \lambda), \tag{2}$$

where $\lambda \in \mathbb{R}^m$ is the multiplier and $\mathscr{L}_i(x_i; \lambda) := g_i(x_i) + \lambda^T (A_i x_i - b/N)$. Then, by exploiting the dual function $d(\lambda) := \min_{x \in X} \mathscr{L}(x; \lambda)$, the dual problem of problem (1) can be written as:

$$d^* := \max_{\lambda \in \mathbb{R}^m_+} d(\lambda). \tag{3}$$

We also call the dual problem (3) the *outer problem*. Note that the dual function $d(\lambda)$ can be computed in a separable fashion as follows:

$$d(\lambda) = \sum_{i=1}^{N} d_i(\lambda) := \min_{x_i \in X_i} \mathscr{L}_i(x_i; \lambda). \tag{4}$$

For a given $\lambda \in \mathbb{R}^m_+$, let

$$x_i(\lambda) := \arg\min_{x_i \in X_i} \mathscr{L}_i(x_i; \lambda), \quad i = 1, \ldots, N. \tag{5}$$

We also call the problems (5) as the *inner subproblems*. From Assumption 1, it can be obtained that for $i = 1, \ldots, N$, $d_i(\lambda)$ is concave and continuously differentiable on $\lambda \in \mathbb{R}^m_+$, and its gradient $\nabla d_i(\lambda) = A_i x_i(\lambda) - b/N$ is Lipschitz continuous with Lipschitz constant $L_{d_i} = \frac{||A_i||^2}{\sigma_i}$ [3,15]. Thus, $d(\lambda)$ is concave and continuously differentiable on $\lambda \in \mathbb{R}^m_+$, and its gradient

$$\nabla d(\lambda) = \sum_{i=1}^{N} A_i x_i(\lambda) - b \tag{6}$$

is Lipschitz continuous with Lipschitz constant $L_d = \sum_{i=1}^{N} \frac{||A_i||^2}{\sigma_i}$. Moreover, the *strong duality* holds, i.e., $d^* = g^*$.

Let Λ^* be the dual optimal solution set of the dual problem (3). Based on dual decomposition, we can solve the dual problem (3) with gradient-based methods. However, the dual gradient (6) depends on the exact solution $x_i(\lambda)$ of the inner subproblems (5). In practice, the inner subproblems (5) cannot be solved exactly, except when all the inner subproblems (5) have closed-form solutions. Therefore, it is necessary to study the case that these subproblems are solved inexactly. Assuming that an approximate optimal solution $\tilde{x}_i(\lambda)$ of the inner subproblems (5) is obtained as follows:

$$\tilde{x}_i(\lambda) \approx \arg\min_{x_i \in X_i} \mathscr{L}_i(x_i; \lambda), \quad i = 1, \ldots, N. \tag{7}$$

Under this case, the use of inexact first-order information including the approximate dual gradients and approximate dual values is inevitable. Now we define

$$\tilde{\nabla} d(\lambda) := \sum_{i=1}^{N} A_i \tilde{x}_i(\lambda) - b \;\text{ and }\; \tilde{d}(\lambda) := \sum_{i=1}^{N} \mathscr{L}_i(\tilde{x}_i(\lambda), \lambda).$$

We assume that $\tilde{x}_i(\lambda)$ is computed as follows:

$$\tilde{x}_i(\lambda) \in X_i, \;\; \mathscr{L}_i(\tilde{x}_i(\lambda), \lambda) - \mathscr{L}_i(x_i(\lambda), \lambda) \leq \frac{\delta_i}{2}, \quad i = 1, \ldots, N, \tag{8}$$

where $\delta_i \geq 0$ is the accuracy parameter. The inequality (8) provides a stopping criterion for solving the inner subproblems (5) inexactly. Many optimization methods offer direct control of this criterion. There are also other stopping criteria depending on the characteristics of the problem under consideration [21,26].

3 Inexact Dual Fast Gradient-Projection Method

By dual decomposition, our goal is to compute an approximate optimal solution for the primal problem (1) for the case that the inner subproblems are solved inexactly. An ϵ-optimal feasible solution of problem (1) is defined as follows:

Definition 3.1 For any given target accuracy $\epsilon > 0$, iff there exist nonnegative constants c_1, c_2 such that $\hat{x}_i \in X_i$, $|\sum_{i=1}^N g_i(\hat{x}_i) - g^*| \leq c_1\epsilon$ and $||[\sum_{i=1}^N A_i\hat{x}_i - b]_+||_\infty \leq c_2\epsilon$. We say that $\hat{x} = (\hat{x}_1^T, \ldots, \hat{x}_N^T)^T \in \mathbb{R}^n$ is an ϵ-optimal feasible solution of primal problem (1).

Definition 3.1 characterizes the distance of the corresponding primal cost from the optimal value and the maximum constraint violation for the primal approximate solution, respectively. Next lemma provides lower and upper bounds on the dual function when the inner subproblems (5) are solved approximatively, whose derivation can be found in Section 3.2 in [26].

Lemma 3.1 *Suppose that Assumption 1 holds. For a given $\lambda \in \mathbb{R}_+^m$ and for each $i = 1, \ldots, N$, let $\tilde{x}_i(\lambda)$ be a solution of the inner subproblems such that (8) is satisfied. Then, for any $\mu \in \mathbb{R}_+^m$, the following results hold:*

$$\tilde{d}(\lambda) + \tilde{\nabla}d(\lambda)^T(\mu - \lambda) \geq d(\mu) \geq \tilde{d}(\lambda) + \tilde{\nabla}d(\lambda)^T(\mu - \lambda) - L_d||\mu - \lambda||^2 - \delta, \quad (9)$$

where $\delta = \sum_{i=1}^N \delta_i$.

For each given $\mu, \lambda \in \mathbb{R}_+^m$, let

$$\tilde{\ell}_d(\mu; \lambda) = \tilde{d}(\lambda) + \tilde{\nabla}d(\lambda)^T(\mu - \lambda) \text{ and } \tilde{\Delta}_d(\mu; \lambda) = \tilde{\ell}_d(\mu; \lambda) - d(\mu).$$

From Lemma 3.1, we have

$$0 \leq \tilde{\Delta}_d(\mu; \lambda) \leq L_d||\mu - \lambda||^2 + \delta. \quad (10)$$

3.1 Inexact Dual Fast Gradient-Projection Algorithm

Based on fast proximal-gradient methods [8,18], we now propose an inexact dual fast gradient-projection method for solving the dual problem (3) with inexact dual first-order information. Note that the proposed algorithm is different from Algorithm (IDFG) developed in [20], since our algorithm does not involve an accumulated history of all the past approximate gradients at each iteration. We call this algorithm as inexact dual fast gradient-projection (for short, IDFGP) algorithm and present it as follows:

Algorithm IDFGP
Initialization: Set $\lambda^0 = \lambda^{-1} \in \mathbb{R}^m_+,\ \theta^0 = \theta^{-1} = 1$.
Iteration: For $k \geq 0$, do
Step 1 $\mu^k = \lambda^k + \theta^k(\frac{1}{\theta^{k-1}} - 1)(\lambda^k - \lambda^{k-1})$.
Step 2 Receive μ^k and update primal variables in *parallel*: for $i = 1, \ldots, N$, $\tilde{x}_i^k \approx \arg\min_{x_i \in X_i} \mathscr{L}_i(x_i; \mu^k)$, where $\tilde{x}_i^k$ satisfies (8).
Step 3 Receive $\tilde{x}_i^k$ and compute $\tilde{\nabla} d(\mu^k) = \sum_{i=1}^N A_i \tilde{x}_i^k - b$, update dual variables:

$$\lambda^{k+1} = \left[\mu^k + \frac{1}{2L_d}\tilde{\nabla} d(\mu^k)\right]_+. \tag{11}$$

Step 4

$$\theta^{k+1} = \frac{\sqrt{(\theta^k)^4 + 4(\theta^k)^2} - (\theta^k)^2}{2}. \tag{12}$$

In terms of Step 4 of Algorithm IDFGP, it holds the following induction [18],

$$\vartheta^k := \sum_{i=0}^{k} \frac{1}{\theta^i} = \frac{1}{(\theta^k)^2}. \tag{13}$$

Moreover, the stepsize in Step 4 also satisfies $\theta^k \leq \frac{2}{k+2}$.

3.2 Convergence Analysis

Let

$$\nu^k := \lambda^{k-1} + \frac{1}{\theta^{k-1}}\left(\lambda^k - \lambda^{k-1}\right) = \lambda^k + \frac{1}{\theta^k}\left(\mu^k - \lambda^k\right), \tag{14}$$

where $\nu^0 = \lambda^0$ (see, [8,18]). To prove the convergence for Algorithm IDFGP, we provide a useful lemma, whose proof is similar to Proposition 2 in [18].

Lemma 3.2 *Let $\{\lambda^k, \mu^k, x^k, \theta^k\}_{k\geq 0}$ be the sequence generated by Algorithm IDFGP. Then, for any $k \geq 0$ and $\lambda \in \mathbb{R}^m_+$, it holds that*

$$\frac{d(\lambda) - d(\lambda^{k+1})}{(\theta^k)^2} + \sum_{l=0}^{k} \frac{\tilde{\Delta}_d(\lambda; \mu^l)}{\theta^l} + L_d\|\lambda - \nu^{k+1}\|^2 \leq L_d\|\lambda - \lambda^0\|^2 + \delta\sum_{l=0}^{k}\frac{1}{(\theta^l)^2}. \tag{15}$$

The next theorem provides an estimate on the dual suboptimality bound for Algorithm IDFGP by using Lemma 3.2.

Theorem 3.1 *Let $\{\lambda^k, \mu^k, x^k, \theta^k\}_{k\geq 0}$ be the sequence generated by Algorithm IDFGP. Then for all $k \geq 0$ and $\lambda^* \in \Lambda^*$, we have:*

$$0 \leq d(\lambda^*) - d(\lambda^{k+1}) \leq \frac{4L_d}{(k+2)^2}||\lambda^* - \lambda^0||^2 + \frac{4}{3}(k+2)\delta. \tag{16}$$

Proof Taking $\lambda = \lambda^* \in \Lambda^*$ in (15) and dropping both $\sum_{l=0}^{k} \frac{\tilde{\Delta}_d(\lambda^*;\mu^l)}{\theta^l}$ and $L_d\|\lambda^* - \nu^{k+1}\|^2$ since they are nonnegative, we obtain

$$0 \le d(\lambda^*) - d(\lambda^{k+1}) \le L_d(\theta^k)^2\|\lambda^* - \lambda^0\|^2 + \delta(\theta^k)^2 \sum_{l=0}^{k} \frac{1}{(\theta^l)^2}. \tag{17}$$

Now we need to estimate $\sum_{l=0}^{k} \frac{1}{(\theta^l)^2}$. Note that the sequence $\{\theta^k\}_{k\ge 0}$ generated by Step 4 is decreasing and satisfies:

$$\frac{1}{k+1} \le \theta^k \le \frac{2}{k+2}, \quad \forall k \ge 0. \tag{18}$$

Thus,

$$\sum_{l=0}^{k} \frac{1}{(\theta^l)^2} \le \sum_{l=0}^{k} (l+1)^2 \le \frac{(k+2)^3}{3}. \tag{19}$$

It follows from (17), (18) and (19) that (16) holds. □

We now consider the running average for the sequence $\{\tilde{x}_i^k\}_{k\ge 0}$:

$$\bar{x}_i^k := \frac{1}{\vartheta^k} \sum_{l=0}^{k} \frac{1}{\theta^i} \tilde{x}_i^l = \left(1 - \theta^k\right) \bar{x}_i^{k-1} + \theta^k \tilde{x}_i^k, \ i = 1, \ldots, N, \tag{20}$$

where $\bar{x}_i^{-1} = 0$. The important implication in the second equality of (20) is that each $\bar{x}_i^k$, $i = 1, \ldots, N$ can access the running average locally and does not need to store all previous iterates to obtain $\bar{x}_i^k$. Several convergence results for the running average sequence have been reported in [8,18].

In order to reconstruct a near optimal and feasible primal solution, we need to provide an upper bound on the running average of the sequence $\{\tilde{\nabla} d(\mu^k)\}_{k\ge 0}$.

Proposition 3.1 *Let* $\{\lambda^k, \mu^k, x^k, \theta^k\}_{k\ge 0}$ *be the sequence generated by Algorithm IDFGP. Then, for any* $k \ge 0$*, it holds that*

$$\frac{1}{\vartheta^k} \sum_{l=0}^{k} \frac{\tilde{\nabla} d(\mu^l)}{\theta^l} \le 2L_d(\theta^k)^2(\nu^{k+1} - \lambda^0). \tag{21}$$

Proof From Step 3 of Algorithm IDFGP, we have

$$\mu^k + \frac{1}{2L_d}\tilde{\nabla} d(\mu^k) \le \left[\mu^k + \frac{1}{2L_d}\tilde{\nabla} d(\mu^k)\right]_+ = \lambda^{k+1}.$$

Dividing by θ^k and rearranging terms, it gives rise to

$$\frac{\tilde{\nabla} d(\mu^k)}{\theta^k} \leq \frac{2L_d}{\theta^k}\left(\lambda^{k+1} - \mu^k\right) \overset{(14)}{=} 2L_d\left(\nu^{k+1} - \nu^k\right).$$

Summing up the above estimate and noting that $\nu^0 = \lambda^0$, we have

$$\sum_{l=0}^{k} \frac{\tilde{\nabla} d(\mu^l)}{\theta^l} \leq 2L_d\left(\nu^{k+1} - \lambda^0\right).$$

Dividing by ϑ^k and using (13), it yields the desired result. □

The next theorem provides an estimate on constraint violation for $\bar{x}^k$ in problem (1).

Theorem 3.2 *Let $\{\lambda^k, \mu^k, x^k, \theta^k\}_{k\geq 0}$ be the sequence generated by Algorithm IDFGP. Then, for any $k \geq 0$,*

$$\left\| \left[\sum_{i=1}^{N} A_i \bar{x}_i^k - b \right]_+ \right\|_\infty \leq e(k, \delta), \tag{22}$$

where $e(k,\delta) = \frac{16L_d}{(k+2)^2}||\lambda^* - \lambda^0|| + 8\sqrt{\frac{L_d}{3(k+2)}}\delta$.

Proof By Proposition 3.1, we have

$$\begin{aligned}\sum_{i=1}^{N} A_i \bar{x}_i^k - b &= \frac{1}{\vartheta^k}\sum_{l=0}^{k} \frac{\sum_{i=1}^{N} A_i \tilde{x}_i^l - b}{\theta^l} \\ &= \frac{1}{\vartheta^k}\sum_{l=0}^{k} \frac{\tilde{\nabla} d(\mu^l)}{\theta^l} \overset{(21)}{\leq} 2L_d\left(\theta^k\right)^2\left(\nu^{k+1} - \lambda^0\right).\end{aligned}$$

Hence, using the facts that $a \leq b \Rightarrow [a]_+ \leq [b]_+$ and $\|a_+\| \leq \|a\|$ for any vectors $a, b \in \mathbb{R}^m$, we get

$$\begin{aligned}\left\| \left[\sum_{i=1}^{N} A_i \bar{x}_i^k - b \right]_+ \right\| &\leq 2L_d\left(\theta^k\right)^2\left\|\left[\nu^{k+1} - \lambda^0\right]_+\right\| \leq 2L_d\left(\theta^k\right)^2\left\|\nu^{k+1} - \lambda^0\right\| \\ &\leq 2L_d\left(\theta^k\right)^2\left(\left\|\nu^{k+1} - \lambda^*\right\| + \left\|\lambda^* - \lambda^0\right\|\right).\end{aligned} \tag{23}$$

Taking $\lambda = \lambda^* \in \Lambda^*$ in (15) and dropping both $\sum_{l=0}^{k}\frac{\tilde{\Delta}_d(\lambda^*;\mu^l)}{\theta^l}$ and $d(\lambda^*) - d(\lambda^{k+1})$ since they are nonnegative, we have

$$L_d\left\|\lambda^* - \nu^{k+1}\right\|^2 \leq L_d\left\|\lambda^* - \lambda^0\right\|^2 + \delta\sum_{l=0}^{k}\frac{1}{(\theta^l)^2} \overset{(18)}{\leq} L_d\left\|\lambda^* - \lambda^0\right\|^2 + \delta\frac{(k+2)^3}{3}.$$

Making use of $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for $a, b \geq 0$, the above inequality yields

$$\left\|\lambda^* - \nu^{k+1}\right\| \leq \left\|\lambda^* - \lambda^0\right\| + (k+2)\sqrt{\frac{k+2}{3L_d}\delta}.$$

Combining the estimates obtained above with (23) gives rise to

$$\left\|\left[\sum_{i=1}^{N} A_i \bar{x}_i^k - b\right]_+\right\| \leq \frac{16L_d}{(k+2)^2}\left\|\lambda^* - \lambda^0\right\| + 8\sqrt{\frac{L_d}{3(k+2)}\delta},$$

where we use the fact that $\theta^k \leq \frac{2}{k+2}$. This completes the proof. □

The following result gives an estimate on primal suboptimality for $\bar{x}^k$ in problem (1).

Theorem 3.3 *Let* $\{\lambda^k, \mu^k, x^k, \theta^k\}_{k\geq 0}$ *be the sequence generated by Algorithm IDFGP. Then, for any* $k \geq 0$*,*

$$\sum_{i=1}^{N} g_i\left(\bar{x}_i^k\right) - g^* \leq \frac{4L_d}{(k+2)^2}\|\lambda^0\|^2 + 2(k+2)\delta. \tag{24}$$

Proof From Lemma 3.1, we have

$$\tilde{\ell}_d(\lambda;\mu) \geq d(\lambda) \overset{(5)}{=} \mathscr{L}(x(\lambda),\lambda) \overset{(8)}{\geq} \mathscr{L}(\tilde{x}(\mu),\lambda) - \frac{\delta}{2}.$$

Hence,

$$\tilde{\Delta}_d(\lambda;\mu) \geq \mathscr{L}(\tilde{x}(\mu),\lambda) - d(\lambda) - \frac{\delta}{2}, \quad \forall\lambda \in \mathbb{R}^m_+.$$

Taking $\mu = \mu^l$ in the above inequality and using Step 2 of Algorithm IDFGP yields

$$\tilde{\Delta}_d\left(\lambda;\mu^l\right) \geq \mathscr{L}\left(\tilde{x}^l;\lambda\right) - d\left(\lambda\right) - \frac{\delta}{2}.$$

Now dividing by θ^l and summing from 0 to k, we have

$$\sum_{l=0}^{k}\frac{1}{\theta^l}\tilde{\Delta}_d\left(\lambda;\mu^l\right) \geq \vartheta^k\left[\mathscr{L}\left(\bar{x}^k;\lambda\right) - d\left(\lambda\right) - \frac{\delta}{2}\right] \overset{(13)}{=} \frac{1}{(\theta^k)^2}\left[\mathscr{L}(\bar{x}^k;\lambda) - d(\lambda) - \frac{\delta}{2}\right],$$

where the first inequality uses the convexity of $\mathscr{L}(\cdot\,;\lambda)$ for any given $\lambda \in \mathbb{R}^m_+$. Taking the above estimate into (15) while dropping the nonnegative term $L_d\|\lambda - \nu^{k+1}\|^2$ gives

$$\mathscr{L}\left(\bar{x}^k;\lambda\right) - d\left(\lambda^{k+1}\right) \leq L_d\left(\theta^k\right)^2\left\|\lambda-\lambda^0\right\|^2 + \delta\left(\theta^k\right)^2\sum_{l=0}^{k}\frac{1}{\left(\theta^l\right)^2} + \frac{\delta}{2}, \quad \forall\lambda \geq 0.$$

By (2), for any $\lambda \geq 0$, the above inequality yields

$$\sum_{i=1}^{N} g_i\left(\bar{x}_i^k\right) + \lambda^T\left(\sum_{i=1}^{N} A_i\bar{x}_i^k - b\right) - d\left(\lambda^{k+1}\right)$$
$$\leq L_d\left(\theta^k\right)^2\|\lambda-\lambda^0\|^2 + \delta\left(\theta^k\right)^2\sum_{l=0}^{k}\frac{1}{\left(\theta^l\right)^2} + \frac{\delta}{2}.$$

By letting $\lambda = 0$ in the above relation, we obtain

$$\sum_{i=1}^{N} g_i\left(\bar{x}_i^k\right) - d\left(\lambda^{k+1}\right) \leq L_d\left(\theta^k\right)^2\left\|\lambda^0\right\|^2 + \delta\left(\theta^k\right)^2\sum_{l=0}^{k}\frac{1}{\left(\theta^l\right)^2} + \frac{\delta}{2}.$$

Thus, by using $d(\lambda^{k+1}) \leq d^* = g^*$, (18) and (19), the above relation yields

$$\sum_{i=1}^{N} g_i\left(\bar{x}_i^k\right) - g^* \leq \frac{4L_d}{(k+2)^2}\left\|\lambda^0\right\|^2 + \left[\frac{4}{3}(k+2) + \frac{1}{2}\right]\delta \leq \frac{4L_d}{(k+2)^2}\left\|\lambda^0\right\|^2 + 2(k+2)\delta.$$

□

In addition, a lower bound on $\sum_{i=1}^{N} g_i(\bar{x}_i^k) - g^*$ can be obtained from Theorem 3.2 as below corollary:

Corollary 3.1 *Let $\{\lambda^k, \mu^k, x^k, \theta^k\}_{k\geq 0}$ be the sequence generated by Algorithm IDFGP. Then, for any $k \geq 0$, it holds that*

$$\sum_{i=1}^{N} g_i(\bar{x}_i^k) - g^* \geq -(\|\lambda^* - \lambda^0\| + \|\lambda^0\|)e(k,\delta), \tag{25}$$

where $e(k,\delta)$ is defined in Theorem 3.2.

From Theorems 3.1, 3.2 and 3.3, we can observe that the first terms of the estimates (16), (22) and (24) are the standard convergence rates of fast gradient methods for the class of smooth functions [18], while the second terms represent the error induced by using inexact first-order information. This shows that Algorithm IDFGP accumulates the errors. Our results validate the theoretical development in [26].

Next theorem shows the bound between the averaged primal sequence and the unique optimizer.

Theorem 3.4 *Let $\{\lambda^k, \mu^k, x^k, \theta^k\}_{k\geq 0}$ be the sequence generated by Algorithm IDFGP. Then, for any $k \geq 0$,*

$$\left\|\bar{x}^k - x^*\right\|^2 \leq \frac{8L_d(\left\|\lambda^0\right\|^2 + 4\|\lambda^*\|\left\|\lambda^*-\lambda^0\right\|)}{\sigma(k+2)^2} + \frac{16\|\lambda^*\|}{\sigma}\sqrt{\frac{L_d\delta}{3(k+2)}}, \tag{26}$$

where $\sigma = \min_i\{\sigma_i\}$.

Proof Since $g(x) = \sum_{i=1}^N g_i(x)$ is continuously differentiable and σ-strongly convex on X, for any given $\lambda \in \mathbb{R}_+^m$, $\mathscr{L}(x;\lambda) = \sum_{i=1}^N g_i(x_i) + \lambda^T(\sum_{i=1}^N A_i x_i - b)$ is also σ-strongly convex on X. From Theorem 2.1.7 in [17], we have, for $x^* \in X^*$, $\lambda^* \in \Lambda^*$,

$$\frac{\sigma}{2}\left\|\bar{x}^k - x^*\right\|^2 \leq \mathscr{L}\left(\bar{x}^k;\lambda^*\right) - \mathscr{L}\left(x^*;\lambda^*\right) = \sum_{i=1}^N g_i\left(\bar{x}_i^k\right) + \lambda^{*T}\left(\sum_{i=1}^N A_i\bar{x}_i^k - b\right) - g^*$$
$$\leq \sum_{i=1}^N g_i\left(\bar{x}_i^k\right) - g^* + \|\lambda^*\| \left\|\left[\sum_{i=1}^N A_i\bar{x}_i^k - b\right]_+\right\|_\infty.$$

Combining the inequality above with (24) and (22), we obtain (26). □

For a given target accuracy ϵ, we provide a relationship between the number of outer iterations k_ϵ, the inner accuracy δ_i and ϵ such that primal suboptimality and constraint violation satisfy Definition 3.1. We denote $D_\lambda := ||\lambda^* - \lambda^0||$, $\lambda^* \in \Lambda^*$ and take the initial iterate $\lambda^0 = 0$ for simplicity, and thus $D_\lambda = ||\lambda^*||$. We also assume that all δ_i are equal, i.e., $\delta_i = \delta/N$. According to Theorems 3.1, 3.2, 3.3, 3.4, and Corollary 3.1, and taking

$$k_\epsilon = \frac{2D_\lambda\sqrt{L_d}}{\sqrt{\epsilon}} - 2 \ \text{ and } \ \delta_i = \frac{\epsilon\sqrt{\epsilon}}{2ND_\lambda\sqrt{L_d}}, \tag{27}$$

for any $k \geq k_\epsilon$, we have the following bounds for constraint violation as well as primal suboptimality and dual suboptimality:

$$\left\|[\sum_{i=1}^N A_i\bar{x}_i^k - b]_+\right\|_\infty \leq \frac{7}{D_\lambda}\epsilon, \ -7\epsilon \leq \sum_{i=1}^N g_i\left(\bar{x}_i^k\right) - g^*$$
$$\leq 2\epsilon \text{ and } 0 \leq d^* - d\left(\lambda^{k+1}\right) \leq \frac{7}{3}\epsilon.$$

From the choice of δ_i in (27), we can observe that, in Algorithm IDFGP, the inner subproblems (5) need to be solved with higher accuracy than the desired accuracy of the outer problem.

Remark 3.1 (i) By choosing the inner accuracy δ_i suitably, such as that chosen in (27), the convergence rate of Algorithm IDFGP can achieve $\mathscr{O}(1/k_\epsilon^2)$, which is the same order as that of the algorithm developed in [20]. The results above also hold in the case when the inner subproblems are solved exactly per outer iteration, i.e., $\delta_i = 0$ in (8). We call this special case the "exact counterpart." Furthermore, when all the g_i are quadratic convex functions, Algorithm IDFGP can reduce to Algorithm 1 developed in [8].

(ii) In practice we may not calculate exactly the value of $D_\lambda = ||\lambda^*||$, but we can offer an upper bound on the norm of dual optimal solutions for the dual problem (3) by Lemma 1 in [27], i.e.,

$$\max_{\lambda^* \in \Lambda^*} \{||\lambda^*||\} \leq \frac{g(\check{x}) - d(\check{\lambda})}{\gamma(\check{x})} \triangleq \Lambda_\lambda, \tag{28}$$

where $\gamma(\check{x}) = \min\{b - \sum_{i=1}^N A_i \check{x}_i\}$, $\check{x}$ is a Slater's vector, $\check{\lambda} \in \mathbb{R}^m_+$. Interestingly, Λ_λ can be calculated easily in a separable fashion: $\Lambda_\lambda = \frac{1}{\gamma(\check{x})} \sum_{i=1}^N [g_i(\check{x}_i) - \min_{x_i \in X_i} \mathscr{L}_i(x_i; \check{\lambda})]$. Thus, the upper bound of the norm of the dual optimal solution is always available. Clearly, we have $D_\lambda \leq \Lambda_\lambda$.

(iii) If the inequality constraints are replaced by the equality $\sum_{i=1}^N A_i x_i = b$ in (1), we can use the same reasoning as given above. However, the bound for the optimal dual solutions in (28) no longer satisfies for the dual problem $d^* = \max_{\lambda \in \mathbb{R}^m} d(\lambda)$. Fortunately, it follows from Theorem 3.5 in [3] that there exists a sufficiently large number $R > 0$ such that the set $\{\lambda \in \mathbb{R}^m : ||\lambda|| \leq R\}$ contains λ^* which means that $||\lambda^*|| \leq R$. Thus, all the above results still hold for problem (1) with linear coupled inequality and equality constraints.

3.3 Solving Inner Subproblems Inexactly

In this section we focus on solving the inner subproblems (5) inexactly. Based on a simple fast gradient algorithm from Section 2.2 in [17], we solve the inner subproblems (5) in parallel, with a certain degree of inner accuracy. For a fixed μ^k, the inner subproblems (5) can be written as

$$x_i^{k*} := \arg\min_{x_i \in X_i} \mathscr{L}_i(x_i; \mu^k), \quad i = 1, \ldots, N. \tag{29}$$

We assume that these problems can be solved easily. To proceed it further, we need the following assumption:

Assumption 2 Suppose for $i = 1, \ldots, N$, that the gradient of the function $g_i(x_i)$ is Lipschitz continuous with Lipschitz constant $L_i > 0$.

Under Assumptions 1 and 2, we note that $\mathscr{L}_i(x_i; \mu^k) = g_i(x_i) + \mu^{kT}(A_i x_i - b/N)$ is σ_i-strongly convex and has gradient $\nabla_1 \mathscr{L}_i(x_i; \mu^k) = \nabla g_i(x_i) + A_i^T \mu^k$ with Lipschitz constant L_i on $x_i \in X_i$ for any given μ^k. Thus, we can utilize the simple fast gradient method [17] to solve the inner subproblems (29) in parallel (for short, IFG).

Algorithm IFG (μ^k)
Initiation: Given μ^k, for every $i = 1, \ldots, N$, choose $y_i^{k,0} = x_i^{k,0} \in X_i$ and set $\beta_i = \frac{\sqrt{L_i} - \sqrt{\sigma_i}}{\sqrt{L_i} + \sqrt{\sigma_i}}$.
Iteration: For $p = 0, 1, \ldots$, execute in *parallel*:
Step 1 $x_i^{k,p+1} = \mathrm{P}_{X_i}[y_i^{k,p} - \frac{1}{L_i} \nabla_1 \mathscr{L}_i(y_i^{k,p}; \mu^k)]$.
Step 2 $y_i^{k,p+1} = x_i^{k,p+1} + \beta_i (x_i^{k,p+1} - x_i^{k,p})$.
Until the condition (8) is satisfied.

For Algorithm IFG, it can achieve the following computational complexity.

Theorem 3.5 *Suppose that Assumptions 1 and 2 hold. For given k and μ^k, let the sequences $\{x_i^{k,p}\}_{p\geq 0}$, $i = 1, \ldots, N$ be generated by Algorithm IFG. For $i = 1, \ldots, N$, then there exists a*

$$p_i = \sqrt{\frac{L_i}{\sigma_i}} \ln \frac{D_i^2(\sigma_i + L_i)}{\delta_i} + 1, \tag{30}$$

such that, for any $p \geq p_i$, the stopping criterion (8) is satisfied.

Proof From Theorem 2.2.3 in [17], we have

$$\begin{aligned} &\mathcal{L}_i\left(x_i^{k,p}; \mu^k\right) - \mathcal{L}_i\left(x_i^{k*}; \mu^k\right) \\ &\quad \leq \left(\mathcal{L}_i\left(x_i^{k,1}; \mu^k\right) - \mathcal{L}_i\left(x_i^{k*}; \mu^k\right) + \frac{\sigma_i}{2}\left\|x_i^{k,1} - x_i^{k*}\right\|^2\right) e^{-(p-1)\sqrt{\frac{\sigma_i}{L_i}}}. \end{aligned} \tag{31}$$

From Assumptions 1 and 2, we know that $\mathcal{L}_i(x_i; \mu^k)$ is σ_i-strongly convex (also convex) and has Lipschitz continuous gradient with Lipschitz constant L_i. In light of Lemma 1.2.3 [17], for any $x_i, y_i \in X_i$, we obtain

$$0 \leq \mathcal{L}_i\left(x_i; \mu^k\right) - \left[\mathcal{L}_i\left(y_i; \mu^k\right) + \nabla_1 \mathcal{L}_i\left(y_i; \mu^k\right)^T (x_i - y_i)\right] \leq \frac{L_i}{2}\|x_i - y_i\|^2. \tag{32}$$

Next we derive a upper bound for the term $\mathcal{L}_i(x_i^{k,1}; \mu^k) - \mathcal{L}_i(x_i^{k*}; \mu^k)$ in (31):

$$\begin{aligned} \mathcal{L}_i\left(x_i^{k,1}; \mu^k\right) &\overset{(32)}{\leq} \mathcal{L}_i\left(x_i^{k,0}; \mu^k\right) + \nabla_1 \mathcal{L}_i\left(x_i^{k,0}; \mu^k\right)^T \left(x_i^{k,1} - x_i^{k,0}\right) \\ &\quad + \frac{L_i}{2}\left\|x_i^{k,1} - x_i^{k,0}\right\|^2 \\ &= \min_{x_i \in X_i}\left\{\mathcal{L}_i\left(x_i^{k,0}; \mu^k\right) + \nabla_1 \mathcal{L}_i\left(x_i^{k,0}; \mu^k\right)^T \left(x_i - x_i^{k,0}\right) + \frac{L_i}{2}\left\|x_i - x_i^{k,0}\right\|^2\right\} \\ &\overset{(32)}{\leq} \min_{x_i \in X_i}\left\{\mathcal{L}_i\left(x_i; \mu^k\right) + \frac{L_i}{2}\left\|x_i - x_i^{k,0}\right\|^2\right\} \leq \mathcal{L}_i\left(x_i^{k*}; \mu^k\right) + \frac{L_i}{2}\left\|x_i^{k*} - x_i^{k,0}\right\|^2, \end{aligned}$$

where the equality uses Step 1 of Algorithm IFG for $p = 0$. Combining the last inequality above with (31), we have

$$\mathcal{L}_i\left(x_i^{k,p}; \mu^k\right) - \mathcal{L}_i\left(x_i^{k*}; \mu^k\right) \leq \frac{\sigma_i + L_i}{2} D_i^2 e^{-(p-1)\sqrt{\frac{\sigma_i}{L_i}}},$$

where D_i is the diameter of the set X_i. From this inequality, we can take p_i as in (30) to guarantee the criterion (8) being satisfied. This completes the proof. □

To sum up, we rewrite the inexact dual fast gradient-projection algorithm as follows:

Algorithm IDFGP
Initiation: Given the outer target accuracy ϵ, set $\lambda^0 = \lambda^{-1} = 0 \in \mathbb{R}^m$, $\theta^0 = \theta^{-1} = 1$ and $\bar{x}^{-1} = 0$. Compute the number of outer iteration k_ϵ and the inner accuracy δ_i as in (27).
Outer loop: For $k = 0, 1, \ldots, k_\epsilon$, execute,
Step 1. $\mu^k = \lambda^k + \theta^k(\frac{1}{\theta^{k-1}} - 1)(\lambda^k - \lambda^{k-1})$.
Step 2. Receive μ^k. **Inner loop**: For $i = 1, \ldots, N$, choose $x_i^{k,0} = \bar{x}_i^{k-1}$ and compute p_i as in (30). For $p = 0, 1, \ldots, \lfloor p_i \rfloor$, apply Algorithm IFG to obtain $\tilde{x}_i^k := x_i^{k,\lfloor p_i \rfloor}$.
Step 3. Receive $\tilde{x}_i^k$. Compute approximate gradient $\tilde{\nabla} d(\mu^k) = \sum_{i=1}^N A_i \tilde{x}_i^k - b$, and update λ^{k+1} and θ^{k+1} by (11) and (12), respectively.
Step 4. Update averaged primal sequence $\bar{x}_i^k = (1 - \theta^k)\bar{x}_i^{k-1} + \theta^k \tilde{x}_i^k$ as in (20).
Output: an ϵ-optimal and feasible primal solution $\bar{x}^k$ and an ϵ-optimal dual solution λ^{k+1} for problem (1).

At each outer iteration k, for a given multiplier μ^k, we terminate Algorithm IFG after $\lfloor p_i \rfloor$ inner iterations for each inner subproblem and obtain its output $\tilde{x}_i^k$, which is viewed as an approximation of the optimal solution $x_i(\mu^k)$ for the inner subproblems (5). In the inner loop, we can choose any feasible point $x_i^{k,0} \in X_i$ as a starting point in Algorithm IFG. In order to speed up the convergence of this algorithm, we have used a warm start technique by choosing $x_i^{k,0} = \bar{x}_i^{k-1}$.

4 Numerical Experiments

To illustrate the efficiency of the proposed algorithm, we consider numerical simulations on separable convex quadratic programming (SCQP) problems. It is well known that some engineering problems can be formulated as SCQP problems subject to linear coupled constraints, including finite-horizon linear model predictive control problems [8,20] and network utility maximum problems [5]. In these simulations, we first analyze the behavior of Algorithm IDFGP that depends on the parameters D_λ and δ and then compare our algorithm with Algorithm (IFDG) proposed in [20]. All the simulations were performed on a laptop with CPU AMD A6-3400M with 1.40GHz and 4GB RAM memory, using Matlab R2010b.

Consider the following SCQP problem:

$$\min_{x \in \mathbb{R}^n} g(x) := \sum_{i=1}^N \frac{1}{2} x_i^T Q_i x_i + q_i^T x_i \ \text{ s.t. } \sum_{i=1}^N A_i x_i \\ \le b, \ x_i \in [l_i, u_i] \subset \mathbb{R}^{n_i} \quad i = 1, \ldots, N, \tag{33}$$

where Q_i is symmetric, positive definite and $[l_i, u_i]$ is a bounded box in $\mathbb{R}^{n_i}$ for $i = 1, \ldots, N$.

The data of the test for problem (33) are generated as follows:

(1) Matrix $Q_i := R_i^T R_i + \alpha I_i$, where R_i is an $\lfloor \frac{n_i}{2} \rfloor \times n_i$ matrix generated randomly from a uniform distribution over $[l_Q, u_Q]$, I_i is the identity matrix in $\mathbb{R}^{n_i \times n_i}$, $\alpha \in (0, 1)$ is a random number.
(2) Matrix $A_i \in \mathbb{R}^{m \times n_i}$ is generated randomly from a uniform distribution over $[l_A, u_A]$.
(3) Vectors $b := \sum_{i=1}^N A_i \check{x}_i - \alpha \mathbf{1}_{m\times 1}$ and $q_i =: -Q_i \check{x}_i$, where $\check{x}_i$ is generated randomly from a uniform distribution over $[l_i, u_i]$ and $\mathbf{1}_{m\times 1} = (1, \ldots, 1)^T$.

In the following simulations, we let $[l_i, u_i] = [-1, 1]$, $[l_Q, u_Q] = [-0.5, 0.5]$, $[l_A, u_A] = [-1, 1]$, $\alpha = 0.1$, $N = 10$.

We first analyze the behavior of Algorithm IDFGP in terms of the parameter D_λ. For different dimensions of SCQP problems, we set $n = Nn_i, m = 2n_i, n_i = 10, 20, \ldots, 100$ and consider two different estimates for the number of outer iterations depending on the ways we estimate D_λ. In the first way, we compute the upper bound Λ_λ given in (28), where we choose $\check{\lambda} = 0$ and $\check{x}$ generated by the above data. In the second way, we calculate the exact bound $D_\lambda = ||\lambda^*||$, where λ^* is computed exactly using the subroutine "quadprog" in Matlab. We run 10 random SCQP problems and let $k_{\epsilon,\Lambda_\lambda}$ be the average number of iterations obtained using the upper bound Λ_λ and k_{ϵ,D_λ} be the average number of iterations obtained with $D_\lambda = ||\lambda^*||$. We also compute the average number of outer iterations $k_{\epsilon,real}$ obtained by imposing the following stopping criteria:

$$\left| g\left(\bar{x}^{k_{\epsilon,real}}\right) - g\left(\bar{x}^{k_{\epsilon,real}-1}\right) \right| \le \epsilon \ \text{ and } \ \frac{\left\| \left[\sum_{i=1}^N A_i \bar{x}_i^{k_{\epsilon,real}} - b \right]_+ \right\|_\infty}{\max\{1, ||b||\}} \le \epsilon. \quad (34)$$

For all the above three ways, we set $\epsilon = 10^{-2}$. We report the average number of iterations in Table 1. We can observe from Table 1 that the number of iterations of our algorithm $k_{\epsilon,real}$ offers a good approximation for the expected number of outer iterations k_{ϵ,D_λ} obtained from (27). However, when we replace D_λ by Λ_λ in (27), then $k_{\epsilon,\Lambda_\lambda}$ is between 15 to 60 times greater than the number of iterations $k_{\epsilon,real}$.

We now analyze the behavior of the algorithm with respect to the inner accuracy parameter δ. Since the estimates of primal suboptimality and constraint violation are dependent on the choice of δ where $\delta = N\delta_i$, we apply Algorithm IDFGP to solve a random SCQP problem in the setting of $n = 300, m = 60, N = 10, n_i = 30, i = 1, \ldots, N$, and a fixed outer accuracy $\epsilon = 5 \times 10^{-3}$ with different δ. In the simulations, we set the inner accuracy $\delta^1 = 3 \times 10^{-4}$, $\delta^2 = 10^{-2}$ and $\delta^3 = 10^{-1}$, where

Table 1 Average number of iterations of k_{ϵ,D_λ}, $k_{\epsilon,real}$, k_{ϵ,D_λ} with different dimensions

n	100	200	300	400	500	600	700	800	900	1000
k_{ϵ,D_λ}	81	103	135	198	492	588	650	676	819	1143
$k_{\epsilon,real}$	121	119	128	180	608	706	789	896	1224	1675
$k_{\epsilon,\Lambda_\lambda}$	2165	2414	4097	4752	18,450	25,125	35,607	50,355	55,989	63,229

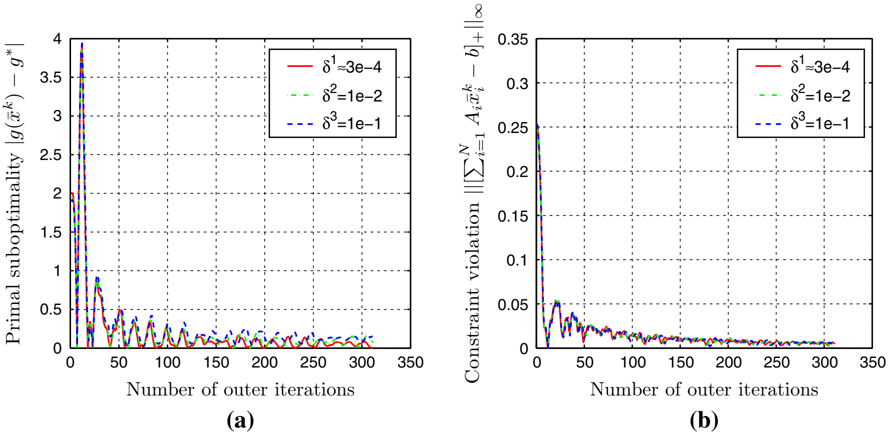

Fig. 1 Results of Algorithm IDFGP for ϵ=5e-3 with different inner accuracy: **a** primal suboptimality versus number of outer iterations; **b** constraint violation versus number of outer iterations

δ_i is computed through (27). Figure 1 shows the primal suboptimality and constraint violation in terms of the number of outer iterations with different values of δ. It can be observed from Fig. 1 that the desired accuracy of both primal suboptimality and constraint violation can be reached when we set the inner accuracy as that calculated through (27). However, if the inner accuracy δ is chosen too large, the desired accuracy of the primal suboptimality cannot be attained, see Fig. 1a. Indeed, Algorithm IDFGP is sensitive to the choice of inner accuracy δ due to the fact that it accumulates errors.

Next we compare the performance of our algorithm with other algorithms. We consider two scenarios of Algorithm IDFGP and set the inner accuracy $\delta^t := \delta$ and $\delta^h := 10^{-3}\delta$, respectively, where δ is calculated by (27). We denote this alternation as Algorithm IDFGP-H. This means that a solution of inner subproblems obtained by Algorithm IDFGP-H is more accurate than that of Algorithm IDFGP. We also compare the performance of our Algorithm IDFGP with Algorithm (IDFG) proposed in [20]. For comparisons, Algorithm IFG is used in the inner loop of all algorithms. Moreover, all algorithms are terminated when the stopping criterion (34) is satisfied by setting the outer accuracy $\epsilon = 10^{-2}$. We report the average number of outer iterations and CPU time by solving 10 random SCQP problems (33) with different sizes in Table 2.

By comparison, it can be seen from Table 2 that Algorithm IDFGP performs better than Algorithm IDFGP-H in the CPU time, although the outer number of iterations of Algorithm IDFGP is slightly worse than that of Algorithm IDFGP-H. Algorithm IDFGP-H can obtain a solution with a higher accuracy for solving inner subproblems, but it requires more iterations toward the desired solution. Therefore, the computational time is dominated by solving inner subproblems in Algorithm IDFGP-H. However, Algorithm IDFGP solves the inner subproblems with only a certain degree of accuracy. Moreover, it can also be observed from Table 2 that the performance of our Algorithm IDFGP is better than that of Algorithm (IDFG) in terms of the number of outer iterations and CPU time.

Table 2 Average number of outer iterations and CPU time (s) with different sizes

	(n, m)	(100, 50)	(200, 100)	(400, 200)	(800, 400)	(1000, 500)
Algorithm IDFGP-H	Ite	193	310	457	701	896
	CPU	0.7103	1.9011	6.1458	42.9791	330.5400
Algorithm IDFGP	Ite	289	421	555	747	957
	CPU	0.0513	0.5038	1.1456	2.6417	4.7091
Algorithm (IDFG)	Ite	295	478	681	927	1157
	CPU	0.0519	0.6251	1.2356	3.0113	5.9070

5 Conclusions

In this paper, we have proposed an inexact dual fast gradient-projection method for solving large-scale separable convex optimization problems with linear coupled constraints. Using Lagrangian dual decomposition together with an accelerated proximal-gradient algorithm, we have developed an inexact dual fast gradient-projection algorithm for solving outer dual problems, when the inner subproblems were solved only up to a certain degree of accuracy by means of a parallel fast gradient scheme. We have provided convergence analysis, deriving the explicitly upper bounds on dual and primal suboptimality and constraint violation. The theoretical results established in the paper are validated by numerical experiments on randomly generated quadratic programming problems.

Acknowledgments We are grateful to the anonymous referees and editor for their useful comments, which have made the paper clearer and more comprehensive than the earlier version. This work was partially supported by the Australia Research Council LP130100451, the Natural Science Foundation of China (61473326 and 11471062), the Natural Science Foundation of Chongqing (cstc2013jjB00001 and cstc2013jcyjA00029) and the Chongqing Normal University Research Foundation (15XLB005).

References

1. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. **2**(1), 183–202 (2009)
2. Gu, G., He, B.S., Yang, J.F.: Inexact alternating-direction-based contraction methods for separable linearly constrained convex optimization. J. Optim. Theory Appl. **163**, 105–129 (2014)
3. Necoara, I., Suykens, J.: Application of a smoothing technique to decomposition in convex optimization. IEEE Trans. Autom. Control **53**(11), 2674–2679 (2008)
4. Wang, K., Han, D.R., Xu, L.L.: A parallel splitting method for separable convex programs. J. Optim. Theory Appl. **159**, 138–158 (2009)
5. Palomar, D.P., Chiang, M.: A tutorial on decomposition methods for network utility maximization. IEEE J. Sel. Areas Commun. **24**(8), 1439–1451 (2006)
6. Oliveira, L.B., Camponogara, E.: Multi-agent model predictive control of signaling split in urban traffic networks. Transp. Res. C Emerg. Technol. **18**(1), 120–139 (2010)
7. Necoara, I., Nedelcu, V., Dumitrache, I.: Parallel and distributed optimization methods for estimation and control in networks. J. Process Control **21**(5), 756–766 (2011)
8. Patrinos, P., Bemporad, A.: An accelerated dual gradient-projection algorithm for embedded linear model predictive control. IEEE Trans. Autom. Control **59**(1), 18–33 (2014)

9. Venkat, A., Hiskens, I., Rawlings, J., Wright, S.: Distributed MPC strategies with application to power system automatic generation control. IEEE Trans. Control Syst. Technol. **16**(6), 1192–1206 (2008)
10. Li, J., Wu, C., Wu, Z., Long, Q., Wang, X.: Distributed proximal-gradient method for convex optimization with inequality constraints. ANZIAM J. **56**, 160–178 (2014)
11. Li, J., Wu, C., Wu, Z., Long, Q.: Gradient-free method for non-smooth distributed optimization. J. Glob. Optim. **61**, 325–340 (2015)
12. Bertsekas, D.P., Tsitsiklis, J.N.: Parallel and Distributed Computation: Numerical Methods. Athena Scientific, Belmont (1997)
13. Boyd, B., 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)
14. Dinh, Q.T., Necoara, I., Savorgnan, C., Diehl, M.: An inexact perturbed path-following method for lagrangian decomposition in large-scale separable convex optimization. SIAM J. Optim. **23**(1), 95–125 (2013)
15. Nesterov, Y.: Smooth minimization of non-smooth functions. Math. Program. **103**(1), 127–152 (2005)
16. Dinh, Q.T., Savorgnan, C., Diehl, M.: Combining lagrangian decomposition and excessive gap smoothing technique for solving large-scale separable convex optimization problems. Comput. Optim. Appl. **55**(1), 75–111 (2013)
17. Nesterov, Y.: Introductory Lectures on Convex Optimization: A Basic Course. Kluwer Academic Publishers, Boston (2004)
18. Tseng, P.: On accelerated proximal gradient methods for convex-concave optimization. Department of Mathematics. University of Washington, Technical Repore (2008)
19. Tseng, P.: Approximation accuracy, gradient methods, and error bound for structured convex optimization. Math. Program. **125**(2), 263–295 (2010)
20. Necoara, I., Nedelcu, V.: Rate analysis of inexact dual first order methods. Application to dual decomposition. IEEE Trans. Autom. Control **59**(5), 1232–1243 (2014)
21. Nedelcu, V., Necoara, I., Dinh, Q.T.: Computational complexity of inexact gradient augmented Lagrangian methods: application to constrained MPC. University Politehnica Bucharest, Technical Report. http://acse.pub.ro/person/ion-necoara (2012)
22. Jiang, K.F., Sun, D.F., Toh, K.C.: An inexact accelerated proximal gradient method for large scale linearly constrained convex SDP. SIAM J. Optim. **22**(3), 1042–1064 (2012)
23. D'Aspremont, A.: Smooth optimization with approximate gradient. SIAM J. Optim. **19**(3), 1171–1183 (2008)
24. Villa, S., Salzo, S., Baldassarre, L., Verri, A.: Accelerated and inexact forward-backward algorithms. SIAM J. Optim. **3**(23), 1607–1633 (2013)
25. He, B.S., Yuan, X.M.: An accelerated inexact proximal point algorithm for convex minimization. J. Optim. Theory Appl. **154**(2), 536–548 (2012)
26. Devolder, O., Glineur, F., Nesterov, Y.: First order methods of smooth convex optimization with inexact oracle. Math. Program. Ser. A **146**(1–2), 37–75 (2014)
27. Nedić, A., Ozdaglar, A.: Approximate primal solutions and rate analysis for dual subgradient methods. SIAM J. Optim. **19**(4), 1757–1780 (2009)