Primal-dual stochastic gradient method for convex programs with many functional constraints

Yangyang Xu

Abstract

Stochastic gradient (SG) method has been popularly applied to solve optimization problems with objective that is stochastic or an average of many functions. Most existing works on SG assume that the underlying problem is unconstrained or has an easy-to-project constraint set. In this paper, we consider problems that have a stochastic objective and also many functional constraints. For such problems, it could be extremely expensive to project a point to the feasible set, or even compute subgradient and/or function value of all constraint functions. To find solutions of these problems, we propose a novel SG method based on the augmented Lagrangian function. Within every iteration, it inquires a stochastic subgradient of the objective, a subgradient and function value of one randomly sampled constraint function, and function value of another sampled constraint function. Hence, the per-iteration complexity is low. We establish its convergence rate for convex and also strongly convex problems. It can achieve the optimal $O(1/\sqrt{k})$ convergence rate for convex case and nearly optimal $O((\log k)/k)$ rate for strongly convex case. Numerical experiments on quadratically constrained quadratic programming are conducted to demonstrate its efficiency.

Keywords: stochastic gradient, augmented Lagrangian method (ALM), functional constraint, iteration complexity

Mathematics Subject Classification: 90C06, 90C25, 90C30, 68W40.

1 Introduction

In this paper, we consider the constrained stochastic programming

$$\min_{x \in X} f_0(x) \equiv \mathbb{E}_\xi[F_0(x; \xi)], \text{ s.t. } f_j(x) \leq 0, j = 1, \ldots, m, \quad (1)$$

where $X$ is a convex set in $\mathbb{R}^n$, $\xi$ is a random variable, and $f_j$ is a convex function for each $j = 0, 1, \ldots, m$. All nonlinear optimization problems can be formulated in the form of (1). We are particularly interested in the case that $m$ is a large number.

*This work is partly supported by NSF grant DMS-1719549.

†xuy21@rpi.edu. Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, New York.
To find a solution of (1), we aim at designing a novel primal-dual stochastic gradient method. We assume an oracle, which can return a stochastic approximation of a subgradient of \( f_0 \), and also the function value and a deterministic subgradient of each \( f_j \) at any inquired point \( x \in X \). Since \( m \) is big, it would be computationally very expensive if at every update, we inquire the objective value and/or subgradient of all \( f_j \)’s. Based on this observation, our algorithm, at every iteration, will simply call the oracle to return a subgradient and function value of one constraint function and the function value of another one.

The algorithm is derived based on the augmented Lagrangian function of an equivalent rescaled variant of (1), i.e.,

\[
\mathcal{L}_\beta(x, z) = f_0(x) + \Psi_\beta(x, z).
\]

Here, \( \beta \) is the penalty parameter, \( z \) is the Lagrangian multiplier vector or dual variable,

\[
\Psi_\beta(x, z) = \frac{1}{m} \sum_{i=1}^{m} \psi_\beta(f_i(x), z_i),
\]

and

\[
\psi_\beta(u, v) = \begin{cases} 
    uv + \frac{\beta}{2} u^2, & \text{if } \beta u + v \geq 0, \\
    -v^2 / 2\beta, & \text{if } \beta u + v < 0.
\end{cases}
\]

Note that \( \Psi \) is convex in \( x \) and concave in \( z \). At each iteration \( k \), we first obtain a stochastic subgradient of \( \mathcal{L}_{\beta_k} \) with respect to \( x \) by calling the oracle to return \( g_0^k \) that is a stochastic subgradient of \( f_0 \) at \( x^k \), and a subgradient and the function value of a sampled constraint function \( f_{i_k} \) to have

\[
\nabla_x \psi_{\beta_k}(f_{i_k}(x^k), z_{i_k}^k) = [\beta f_{i_k}(x^k) + z_{i_k}^k] + \nabla f_{i_k}(x^k).
\]

Then we perform a projected stochastic subgradient update as in (5) to the primal variable \( x \), and finally we update one randomly selected dual component \( z_{j_k} \) through calling the oracle again to return the function value of another sampled constrained function \( f_{j_k} \) at the updated \( x \).

The pseudocode of the proposed method is shown in Algorithm 1, which iteratively performs stochastic subgradient update to the primal variable \( x \) and randomized coordinate update to the dual variable \( z \).

We remark that if the potential application has any affine equality constraint \( a^\top x = b \), we can always write it into two affine inequality constraints \( a^\top x \leq b \) and \( -a^\top x \leq -b \) and thus formulate the problem in the form of (1), or we can use a technique similar to that in [21] to handle the equality and inequality constraints simultaneously.

### 1.1 Motivating examples

We give a few examples that can be written in the form of (1) with very big \( m \).
Algorithm 1: Primal-dual stochastic gradient method

1. **Initialization:** choose $x^1$ and $z^1 = 0$, and set $k = 1$

2. while not convergent do

3. Pick $i_k$ from $[m]$ uniformly at random and choose parameters $\alpha_k$ and $\beta_k$

4. Call the oracle to return a stochastic subgradient $g^k_0$ of $f_0$ and a subgradient and function value of $f_{i_k}$ at $x^k$

5. Update the primal variable $x$ by

   $$x^{k+1} = \arg \min_{x \in X} \left( g^k_0 + \tilde{\nabla}_x \psi_{\beta_k} (f^k_{i_k}(x^k), z^k_{i_k}) \right) + \frac{1}{2\alpha_k} \|x - x^k\|^2,$$

   (5)

6. Pick $j_k$ from $[m]$ uniformly at random and choose a stepsize $\rho_k$

7. Call the oracle to return the function value of $f_{j_k}$ at $x^{k+1}$

8. Update the dual variable $z$ by

   $$z^{k+1}_j = \begin{cases} 
   z^k_j, & \text{if } j \neq j_k \\
   z^k_j + \rho_k \cdot \max \left( -\frac{z^k_j}{\beta_k} \cdot f_{j}(x^{k+1}) \right), & \text{if } j = j_k 
   \end{cases},$$

   (6)

9. Let $k \leftarrow k + 1$.

---

**Stochastic linear programming.** A two-stage stochastic linear programming (c.f. [17, Sec. 2.1]) can be formulated as

$$\min_{x} c^\top x + \mathbb{E}\left[f_\xi(x)\right], \text{ s.t. } Ax \leq b,$$

(7)

where $\xi = (B, C, d, q)$ and $f_\xi(x)$ are respectively the data and the optimal value of the second stage linear programming

$$\min_{y} q^\top y, \text{ s.t. } Bx + Cy \leq d.$$

As there are $m$ scenarios in the second stage, i.e., $\xi \in \{\xi_1, \ldots, \xi_m\}$ with $\text{Prob}(\xi = \xi_i) = p_i > 0$ and $\sum_{i=1}^m p_i = 1$, then

$$\mathbb{E}\left[f_\xi(x)\right] = \sum_{i=1}^m p_i f_{\xi_i}(x) = \sum_{i=1}^m p_i \min \{q_i^\top y : B_i x + C_i y \leq d_i\}.$$

Hence, the two-stage problem (7) can be written as a single large-scale linear programming:

$$\min_{x, y_1, \ldots, y_m} c^\top x + \sum_{i=1}^m p_i q_i^\top y_i$$

s.t. $Ax \leq b$

$B_i x + C_i y_i \leq d_i$, $i = 1, \ldots, m$. 

(8)
Clearly, (8) is in the form of (1), and if there are many scenarios, i.e., $m$ is big, it could be extremely expensive to access all the data at every update to the variables.

**Chance constrained problems by sampling and discarding.** A nonlinear programming with chance constraint is formulated as

$$\min_{x \in X} f_0(x), \text{ s.t. } \text{Prob}(g(x; \xi) \leq 0) \geq 1 - \tau,$$

where $X \subseteq \mathbb{R}^n$ is a convex set, $\xi$ is an uncertain parameter on a support set $\Xi$, and $\tau$ is a user-specified risk level of constraint violation. Even though $g(\cdot; \xi)$ is convex for any $\xi \in \Xi$, the chance constraint set in (9) may not be convex. Hence, exactly solving (9) is hard in general. To numerically solve (9), the work [4] introduces a sample-based approximation method, called *sampling and discarding* approach. This method makes $N$ independent samples of $\xi$, then eliminates $p$ of them, and solves a deterministic problem with the remaining $m = N - p$ constraints, i.e.,

$$\min_{x \in X} f_0(x), \text{ s.t. } g(x; \xi_i) \leq 0, \forall i = 1, \ldots, m,$$

where $\{\xi_1, \ldots, \xi_m\}$ contains the $m$ samples after discarding. It is shown that under certain assumptions, for any $\varepsilon \in (0, 1)$, if

$$\left( \binom{p + n - 1}{p} \sum_{i=0}^{p+n-1} \binom{N}{i} \tau^i (1 - \tau)^{N-i} \right) \leq \varepsilon,$$

the solution of (10) is feasible for (9) with probability at least $1 - \varepsilon$.

Note that if no *discarding* is performed, the above method is similar to the scenario approximation approaches in [9,12]. For high-dimensional problems, i.e., $n$ is big, it is required to set a significantly bigger $N$ and also $N - p$ to have (11). Therefore, the sample-based approximation problem (10) will have many functional constraints and be in the form of (1).

**Robust optimization by sampling.** Different from the chance constrained problem (9), robust optimization requires the constraint $g(x; \xi) \leq 0$ to be satisfied for any $\xi \in \Xi$, i.e.,

$$\min_{x \in X} f_0(x), \text{ s.t. } g(x; \xi) \leq 0, \forall \xi \in \Xi.$$

Similar to the scenario approximation method for chance constrained problems, the sampling approach (e.g., [3]) has also been proposed to numerically solve (12). Let $\{\xi_1, \ldots, \xi_m\}$ be $m$ independently extracted samples. It is shown in [3] that for any $\tau \in (0, 1)$ and any $\varepsilon \in (0, 1)$, if the number of samples satisfies

$$m \geq \frac{n}{\tau \varepsilon} - 1,$$

then the solution to (10) will be a $\tau$-level robustly feasible solution with probability at least $1 - \varepsilon$. If $n$ is big, and high feasibility level and high probability are required, then $m$ would be a very big number, and thus (10) has an extremely big number of functional constraints.
1.2 Existing methods

In this subsection, we review a few existing methods that could be applied to solve (1) and show how our method relates to them.

**Stochastic mirror-prox method.** The proposed method is closely related to the stochastic mirror-prox method [1, 7] for saddle-point problems or more generally for variational inequality problems. By the augmented Lagrangian function, one can equivalently formulate (1) into the following saddle-point problem (c.f., [15]):

\[
\min_{x \in X} \max_{z} L_{\beta}(x, z). \tag{13}
\]

Assuming \( \nabla L_{\beta} \) to be Lipschitz continuous and \( z \) in a compact set \( Z \), we can apply the method in [1] to the above saddle-point problem and have the iterative update:

\[
x^{k+1} = \arg \min_{x \in X} \langle g_{k}^x, x \rangle + \frac{1}{2 \alpha_k} \| x - x^k \|^2, \tag{14}
\]

\[
z^{k+1} = \arg \min_{z \in Z} \langle g_{k}^z, z \rangle + \frac{1}{2 \rho_k} \| z - z^k \|^2, \tag{15}
\]

where \( (g_{k}^x; g_{k}^z) \) is a stochastic approximation of \( \nabla L_{\beta}(x^k, z^k) \). Note that if \( L_{\beta} \) is differentiable, then \( g_{k}^0 + \nabla_x \psi_{\beta_k}(f_{i_k}(x^k), z_{i_k}^k) \) is one stochastic approximation of \( \nabla_x L_{\beta_k}(x^k, z^k) \), and thus in this case, the update in (5) with \( \beta_k = \beta, \forall k \), reduces to that in (14). In addition, we have

\[
\nabla_x L_{\beta}(x, z) = \frac{1}{m} \left[ \begin{array}{c}
\max(-\frac{z_j^k}{\beta}, f_1(x)) \\
\vdots \\
\max(-\frac{z_m^k}{\beta}, f_m(x))
\end{array} \right].
\]

Hence, the vector \( g_{z}^k \) with all-zero components except the \( j_k \)-th one being \( \max(-\frac{z^k_j}{\beta}, f_j(x^k)) \) is a stochastic approximation of \( \nabla_x L_{\beta}(x^k, z^k) \), and with this \( g_{z}^k \) and \( Z = [\bar{z}_1, \bar{z}_1] \times \cdots \times [\bar{z}_m, \bar{z}_m] \), (15) becomes

\[
z_{j_k}^{k+1} = \begin{cases} 
  z_{j_k}^k, & \text{if } j \neq j_k \\
  \text{Proj}_{[\bar{z}_j, \bar{z}_j]} \left( z_{j_k}^k + \rho_k \cdot \max \left( -\frac{z^k_j}{\beta}, f_j(x^k) \right) \right), & \text{if } j = j_k.
\end{cases}
\]

The difference between (6) and the above update is that the former uses \( x^{k+1} \) to update \( z \) while the latter uses the old iterate \( x^k \), and also the latter requires \( z \) within a bounded set \( Z \).

Although the updates in (14) and (15) could be similar to those in (5) and (6), the analysis in [1] assumes Lipschitz continuity of \( \nabla L_{\beta} \) and thus \( \nabla f_j \) for all \( j = 0, 1, \ldots, m \). In addition, [1] requires the dual variable in a compact set, and the boundedness assumption is also made even

---

\[\text{Here, we use the Euclidean norm square as the proximal term, while [1] actually uses a more general Bregman distance function. In addition, the method in [1] performs two stochastic gradient updates during every iteration and finally outputs the weighted average of all points from every iteration’s first update.} \]
for deterministic subgradient method for saddle-point problems [11]. On the contrary, we allow nonsmoothness of the function $f_j$'s, and also we do not assume boundedness of $z$ but instead we can prove the boundedness of the sequence $\{z^k\}$ in expectation. The removal of boundedness assumption on $z$ is important. Without knowing the optimal dual variable $z^*$, it is generally difficult to give the set $Z$, and even if one can estimate an $Z$ that contains $z^*$, the convergence of the algorithm could be slow if the estimated set is too large.

The stochastic gradient method for saddle-point problems is also studied in [13]. However, it requires strong convexity for both primal and dual variables.

**Stochastic subgradient with random constraint projection.** Let $X_0 = X$ and

$$X_j = \{x \in \mathbb{R}^n : f_j(x) \leq 0\}, \ j = 1, \ldots, m. \quad (16)$$

Then (1) can be written to

$$\min_x f_0(x), \text{ s.t. } x \in \cap_{j=0}^m X_j. \quad (17)$$

On solving the above problem, we can apply the method in [18, 19] and iteratively perform the update:

$$x^{k+1} = \text{Proj}_{X_{i_k}} \left( x^k - \alpha_k g^k_0 \right), \quad (18)$$

where $i_k$ is randomly chosen from $\{0, 1, \ldots, m\}$, $\text{Proj}_{X_{i_k}}$ denotes the projection onto $X_{i_k}$, and $g^k_0$ is a stochastic approximation of a subgradient of $f_0$ at $x^k$. Various sampling schemes on $i_k$ are studied in [18]. Under the linear regularity assumption on the set collection $\{X_{i_k}\}_{i=0}^m$, a sublinear convergence result is established in terms of objective value with rate $O(1/\sqrt{k})$ and feasibility violation with rate $O(\log k/\sqrt{k})$. To have efficient computation in the update (18), $X_i$ is required to be a simple set for each $i = 0, 1, \ldots, m$. Hence, if $\text{Proj}_{X_{i_k}}$'s are difficult to evaluate, such as logistic loss function induced constraint set in the Neyman-pearson classification problem [14], the method in [18, 19] will be inefficient. By contrast, our update in (5) can be computed efficiently as long as $X$ is simple. In addition, we will show the same order of convergence rate without assuming linear regularity on $\{X_{i_k}\}_{i=0}^m$.

**Stochastic proximal-proximal gradient method.** Let $r(x) = \iota_X(x)$ and $g_j(x) = \iota_{X_j}(x)$, where $\iota_X$ denotes the indicator function on $X$, and $X_j$'s are defined in (16). Then (1) is equivalent to

$$\min_x r(x) + \frac{1}{m} \sum_{j=1}^m (f_0(x) + g_j(x)). \quad (19)$$

When $f_0$ is differentiable, the stochastic proximal-proximal gradient (S-PPG) method [16] can be applied to find a solution of (19). It starts from $(x^0, z^0_1, \ldots, z^0_m)$ and iteratively performs the update:

$$x^{k+\frac{1}{2}} = \text{Proj}_X \left( \frac{1}{m} \sum_{i=1}^m z^k_i \right),$$

$$x^{k+1} = \text{Proj}_{X_{i_k}} \left( 2x^{k+\frac{1}{2}} - z^k_{i_k} - \alpha \nabla f_0(x^{k+\frac{1}{2}}) \right),$$

$$z^{k+1}_{i_k} = \begin{cases} z^k_i + x^{k+1} - x^{k+\frac{1}{2}}, & \text{if } i = i_k \\ z^k_i, & \text{if } i \neq i_k \end{cases}$$

$$z^{k+1}_{i \neq i_k} = \begin{cases} z^k_{i_k} + x^{k+1} - x^{k+\frac{1}{2}}, & \text{if } i = i_k \\ z^k_{i \neq i_k}, & \text{if } i \neq i_k \end{cases}$$

$$x^{k+1} = \text{Proj}_{X} \left( \frac{1}{m} \sum_{i=1}^m z^{k+1}_i \right),$$

$$z^{k+1}_i = \begin{cases} 2z^{k+1}_{i_k} - z^{k+1}_i - \alpha \nabla f_0(x^{k+1}), & \text{if } i = i_k \\ z^{k+1}_i, & \text{if } i \neq i_k \end{cases}$$

$$z^{k+1}_{i \neq i_k} = \begin{cases} 2z^{k+1}_{i_k} - z^{k+1}_{i \neq i_k} - \alpha \nabla f_0(x^{k+1}), & \text{if } i = i_k \\ z^{k+1}_{i \neq i_k}, & \text{if } i \neq i_k \end{cases}$$

$\ldots$
where $i_k$ is chosen from $\{1, \ldots, m\}$ uniformly at random. Since $\text{Proj}_{X_{i_k}}$ needs be evaluated, S-PPG has the same issue as the update in (18). However, it could be more suitable in a distributed system, for which communication cost is a main concern.

**Stochastic subgradient with single projection.** Let $h(x) = \max_{1 \leq j \leq m} f_j(x)$. Then (1) is equivalent to

$$\min_{x \in X} f_0(x), \text{ s.t. } h(x) \leq 0. \quad (21)$$

For solving the above problem, we can apply the method in [10], which, at every iteration, inquires a stochastic subgradient of $f_0$ and also a subgradient of $h$. Although the method in [10] only needs to perform a single projection to the feasible set at the last step, computing the subgradient of $h$ would generally require evaluating the function value of all $f_j$'s, and thus it is inefficient for the big-$m$ case. This issue is partly addressed in [5], which only checks a batch of randomly sampled constraint functions every iteration. However, depending on the underlying problem and required accuracy, the batch size could be as large as $m$.

**Deterministic primal-dual first-order method.** Another related method is the deterministic primal-dual first-order algorithm [21], which is also based on the augmented Lagrangian function of (1). Different from Algorithm 1, the method in [21] assumes differentiability of $f_j$'s, and it requires exact gradient of $f_0$ and uses all $f_j$, $j = 1, \ldots, m$ to update $x$ and $z$. Hence, if exact gradient of $f_0$ is not available or very expensive to compute, or if $m$ is extremely big, the deterministic method is either inapplicable or inefficient. Similarly, the deterministic first-order methods in [23,24] are also very expensive or do not apply for the stochastic program with many constraints.

Besides the above primal-dual type methods, in the literature there are also purely primal methods that can also be applied to (1) such as the penalty method with stochastic approximation [8] and the cooperative stochastic approximation method [20]. We do not expand our discussion on all these methods but refer the interested readers to those papers and the references therein.

### 1.3 Contributions

We list our contributions below.

- We propose a novel primal-dual stochastic gradient method for solving stochastic programs with many functional constraints. The method is derived based on augmented Lagrangian function. Through a stochastic oracle, it alternatingly performs stochastic subgradient update to primal and dual variables. At every iteration, it only samples two out of many constraint functions and thus has low per-iteration complexity.

- We establish the convergence rate result of the proposed method for convex and also strongly convex problems. Different from existing analysis for saddle-point problems, we do not assume the boundedness of the dual variable $z$, but instead we prove the boundedness of the dual iterate in expectation. For convex problems, we show that the algorithm can achieve the
optimal $O(1/\sqrt{k})$ convergence rate, and for strongly convex case, we show that it can achieve the nearly optimal $O((\log k)/k)$ convergence rate, where $k$ is the number of subgradient inquiries. The results are in terms of both objective value and feasibility violation.

- We show the practical performance of the proposed algorithm by testing it on solving a convex quadratically constrained quadratic program. The numerical results demonstrate that the proposed primal-dual stochastic gradient method can be significantly better than the stochastic mirror-prox algorithm.

1.4 Notation and outline

We use bold lower-case letters $\mathbf{x}, \mathbf{z}, \ldots$ for vectors and $x_i, z_i, \ldots$ for their $i$-th components. The bold number $\mathbf{0}$ and $\mathbf{1}$ denote the all-zero and all-one vectors, respectively. $[m]$ is short for the set $\{1, 2, \ldots, m\}$, $[a]_+ = \max(0, a)$ and $[a]_- = \max(0, -a)$ respectively denote the positive and negative parts for any real number $a$. $||x||$ denotes the Euclidean norm of a vector $x$. For a convex function $f$, we denote by $\nabla f(\mathbf{x})$ a subgradient of $f$ at $\mathbf{x}$, and the set of all subgradients of $f$ at $\mathbf{x}$ is called the subdifferential of $f$, denoted by $\partial f(\mathbf{x})$. We let $\mathcal{H}^k$ contain the history of Algorithm 1 until $\mathbf{x}^k$ and $\mathcal{W}^k$ until $\mathbf{z}^k$, i.e.,

$$\mathcal{H}^k = \{\mathbf{x}^1, \mathbf{z}^1, \mathbf{x}^2, \mathbf{z}^2, \ldots, \mathbf{x}^{k-1}, \mathbf{z}^{k-1}, \mathbf{x}^k\}, \quad \mathcal{W}^k = \mathcal{H}^k \cup \{\mathbf{z}^k\}.$$  

$\mathbb{E}[\zeta]$ denotes the expectation of a random variable $\zeta$, and $\mathbb{E}[\zeta | \xi]$ is for the expectation of $\zeta$ conditional on $\xi$. In addition, we denote

$$\Phi(\mathbf{x}; \mathbf{x}, \mathbf{z}) = f_0(\mathbf{x}) - f_0(\mathbf{x}) + \frac{1}{m} \sum_{i=1}^{m} z_i f_i(\mathbf{x}).$$  

(22)

The rest of the paper is outlined as follows. In section 2, we give the technical assumptions required in our analysis, and in section 3, we analyze the algorithm and show its convergence rate results. Numerical results are provided in section 4, and finally section 5 concludes the paper.

2 Technical assumptions

Throughout our analysis, we make the following assumptions.

**Assumption 1** There exists a primal-dual solution $(\mathbf{x}^*, \mathbf{z}^*)$ satisfying the Karush-Kuhn-Tucker (KKT) conditions:

$$0 \in \partial f_0(\mathbf{x}^*) + \mathcal{N}_X(\mathbf{x}^*) + \frac{1}{m} \sum_{i=1}^{m} z_i^* \partial f_i(\mathbf{x}^*),$$  

(23a)

$$\mathbf{x}^* \in X, \quad f_i(\mathbf{x}^*) \leq 0, \forall i \in [m],$$  

(23b)

$$z_i^* \geq 0, \quad z_i^* f_i(\mathbf{x}^*) = 0, \forall i \in [m],$$  

(23c)
where \( N_X(x) \) denotes the normal cone of \( X \) at \( x \).

**Assumption 2** The stochastic approximation \( g_k \) is unbiased and has bounded variance, i.e., there is a constant \( \sigma \geq 0 \) such that
\[
\mathbb{E}[g_k] \in \partial f_0(x^k), \quad \mathbb{E}\left\|g_k - \mathbb{E}[g_k]\right\|^2 \leq \sigma^2, \forall k.
\]

In addition, there exist constants \( F \) and \( G \) such that
\[
|f_i(x)| \leq F, \|\tilde{\nabla}f_i(x)\| \leq G, \forall i \in [m], \forall x \in X.
\]

Assumption 1 holds if a certain constraint qualification holds such as the Slater’s condition \[2\].

In Assumption 2, the unbiaseness and variance boundedness assumption on \( g_k \) is standard in the literature of stochastic gradient method, and the boundedness of each \( f_i \) and \( \tilde{\nabla}f_i \) is satisfied if \( X \) is bounded.

As the KKT conditions in (23) hold, there are \( \tilde{\nabla}f_i(x^*) \), \( \forall i \in [m] \) such that
\[
-\frac{1}{m} \sum_{i=1}^{m} z_i^* \tilde{\nabla}f_i(x^*) \in \partial f_0(x^*) + N_X(x^*).
\]

Hence, from the convexity of \( f_0 \) and \( X \), it follows that
\[
f_0(x) \geq f_0(x^*) - \left\langle \frac{1}{m} \sum_{i=1}^{m} z_i^* \tilde{\nabla}f_i(x^*), x - x^* \right\rangle, \forall x \in X. \tag{24}
\]

Since \( z_i^* \geq 0 \) and \( f_i \) is convex for each \( i \in [m] \), we have
\[
z_i^* (f_i(x) - f_i(x^*)) \geq \langle z_i^* \tilde{\nabla}f_i(x^*), x - x^* \rangle.
\]

The above inequality together with (24) and the fact \( z_i^* f_i(x^*) = 0, \forall i \in [m] \) implies
\[
\Phi(x; x^*, z) = f_0(x) - f_0(x^*) + \frac{1}{m} \sum_{i=1}^{m} z_i^* f_i(x) \geq 0, \forall x \in X. \tag{25}
\]

### 3 Convergence analysis

In this section, we analyze the convergence of Algorithm 1 under Assumptions 1 and 2. We show that for convex problems, our method can achieve the optimal convergence rate \( O(1/\sqrt{k}) \), and for strongly convex problems, it can achieve a near-optimal rate \( O((\log k)/\sqrt{k}) \), where \( k \) is the number of iterations. While existing analysis \[1,11\] for saddle-point problems assumes the boundedness of the dual variable, we do not require such an assumption. Instead we can bound all \( z^k \) in expectation by choosing appropriate parameters.

We assume \( f_0 \) to be strongly convex with modulus \( \mu \geq 0 \), i.e.,
\[
f_0(y) \geq f_0(x) + \langle \tilde{\nabla}f_0(x), y - x \rangle + \frac{\mu}{2} \|y - x\|^2, \forall x, y. \tag{26}
\]

If \( \mu = 0 \), then \( f_0 \) is simply a convex function.
3.1 Preliminary lemmas

We first establish a few lemmas. The lemma below has appeared in [21, 22]. For completeness, we give a proof here.

**Lemma 3.1** Let \( \bar{x} \in X \) be a random vector. If for any \( z \geq 0 \) that may depend on \( \bar{x} \), it holds

\[
\mathbb{E}[\Phi(\bar{x}; x^*, z)] \leq \varepsilon_1 + \varepsilon_2 \mathbb{E}[\|z\|^2],
\]

where \( \Phi \) is defined in (22), then for any \((x^*, z^*)\) satisfying (23),

\[
\mathbb{E} |f_0(\bar{x}) - f_0(x^*)| \leq 2\varepsilon_1 + 9\varepsilon_2 \|z^*\|^2,
\]

\[
\mathbb{E} \left[ \frac{1}{m} \sum_{i=1}^{m} [f_i(\bar{x})]_+ \right] \leq \varepsilon_1 + \varepsilon_2 \|1 + z^*\|^2.
\]

**Proof.** Since \(-z_i^* f_i(\bar{x}) \geq -z_i^* [f_i(\bar{x})]_+\), we have from (25) that

\[
f_0(\bar{x}) - f_0(x^*) \geq -\frac{1}{m} \sum_{i=1}^{m} z_i^* [f_i(\bar{x})]_+.
\]

We obtain the inequality in (29), by substituting the above inequality into (27) with \( z \) given by \( z_i = 1 + z_i^* \) if \( f_i(\bar{x}) > 0 \) and \( z_i = 0 \) otherwise for any \( i \in [m] \).

Letting \( z_i = 3z_i^* \) if \( f_i(\bar{x}) > 0 \) and \( z_i = 0 \) otherwise for each \( i \in [m] \) in (27) and adding (30) together gives

\[
\mathbb{E} \left[ \frac{1}{m} \sum_{i=1}^{m} z_i^* [f_i(\bar{x})]_+ \right] \leq \frac{\varepsilon_1}{2} + \frac{9\varepsilon_2}{2} \|z^*\|^2.
\]

Hence, by the above inequality and (30), we obtain \( \mathbb{E} [f_0(\bar{x}) - f_0(x^*)] \leq \frac{\varepsilon_1}{2} + \frac{9\varepsilon_2}{2} \|z^*\|^2 \). In addition, from (27) with \( z = 0 \), it follows \( \mathbb{E} [f_0(\bar{x}) - f_0(x^*)] \leq \varepsilon_1 \). Since \( |a| = a + 2[a]_\|z^*\|^2 \) for any real number \( a \), we have

\[
\mathbb{E} [f_0(\bar{x}) - f_0(x^*)] = \mathbb{E} [f_0(\bar{x}) - f_0(x^*)] + 2\mathbb{E} [f_0(\bar{x}) - f_0(x^*)] \leq 2\varepsilon_1 + 9\varepsilon_2 \|z^*\|^2,
\]

which completes the proof. \( \square \)

The following three lemmas will be used to establish an important inequality for running one iteration of Algorithm 1.

**Lemma 3.2** For any \( z \geq 0 \), it holds

\[
-\Psi_{\beta_k}(x^{k+1}, z^k) + \frac{1}{m} \sum_{i=1}^{m} z_i f_i(x^{k+1}) + \frac{1}{2\rho_k} \mathbb{E} \left[ \|z^{k+1} - z\|^2 - \|z^k - z\|^2 \mid h^{k+1} \right]
\]

\[
\leq \left( \frac{1}{2\rho_k} - \frac{\beta_k}{2\rho_k^2} \right) \mathbb{E} \left[ \|z^{k+1} - z^k\|^2 \mid h^{k+1} \right]
\]

(31)
Proof. From the update of \( \mathbf{z} \), we have
\[
\langle \mathbf{z}^k - \mathbf{z}, \mathbf{z}^{k+1} - \mathbf{z}^k \rangle = \rho_k(z_j^k - z_{j_k}) \cdot \max \left( -\frac{z_j^k}{\beta_k}, f_{j_k}(\mathbf{x}^{k+1}) \right).
\]
Since \( j_k \) is chosen from \([m]\) uniformly at random, taking conditional expectation gives
\[
\frac{1}{\rho_k} \mathbb{E} \left[ \langle \mathbf{z}^k - \mathbf{z}, \mathbf{z}^{k+1} - \mathbf{z}^k \rangle \mid \mathcal{H}^{k+1} \right] = \frac{1}{m} \sum_{i=1}^{m} (z_i^k - z_i) \cdot \max \left( -\frac{z_i^k}{\beta_k}, f_i(\mathbf{x}^{k+1}) \right).
\]
Let
\[
I_+^k = \{ i \in [m] : \beta_k f_i(\mathbf{x}^{k+1}) + z_i^k \geq 0 \}, \quad I_-^k = [m] \setminus I_+^k.
\]
Then one can directly verify that
\[
-\Psi_{\beta_k}(\mathbf{x}^{k+1}, \mathbf{z}^k) + \frac{1}{m} \sum_{i=1}^{m} z_i f_i(\mathbf{x}^{k+1}) + \frac{1}{m} \sum_{i=1}^{m} (z_i^k - z_i) \cdot \max \left( -\frac{z_i^k}{\beta_k}, f_i(\mathbf{x}^{k+1}) \right)
\]
\[
= -\frac{1}{m} \sum_{i \in I_+^k} \frac{\beta_k}{2} [f_i(\mathbf{x}^{k+1})]^2 - \frac{1}{m} \sum_{i \in I_-^k} \frac{(z_i^k)^2}{2\beta_k} - z_i f_i(\mathbf{x}^{k+1}) + \frac{z_i^k}{\beta_k} \right].
\]
Note that for \( \mathbf{z} \geq \mathbf{0} \) and any \( i \in I_+^k \), \( z_i (f_i(\mathbf{x}^{k+1}) + \frac{z_i^k}{\beta_k}) \leq 0 \). In addition, note
\[
-\frac{1}{m} \sum_{i \in I_+^k} \frac{\beta_k}{2} [f_i(\mathbf{x}^{k+1})]^2 - \frac{1}{m} \sum_{i \in I_-^k} \frac{(z_i^k)^2}{2\beta_k} = -\frac{\beta_k}{2\rho_k} \mathbb{E} \left[ \| \mathbf{z}^{k+1} - \mathbf{z}^k \|^2 \mid \mathcal{H}^{k+1} \right].
\]
Hence, plugging (32) into (33) and noting
\[
\langle \mathbf{z}^k - \mathbf{z}, \mathbf{z}^{k+1} - \mathbf{z}^k \rangle = \frac{1}{2} \left[ \| \mathbf{z}^{k+1} - \mathbf{z} \|^2 - \| \mathbf{z}^k - \mathbf{z} \|^2 - \| \mathbf{z}^{k+1} - \mathbf{z}^k \|^2 \right]
\]
gives the desired result. \( \square \)

**Lemma 3.3** Under Assumption 2, for any \( \mathbf{x} \in X \) and any \( \mathbf{z} \), it holds
\[
\frac{1}{m} \sum_{i=1}^{m} \| \nabla_x \psi(f_i(\mathbf{x}), z_i) \|^2 \leq 2\beta^2 F^2 G^2 + \frac{2G^2}{m} \| \mathbf{z} \|^2.
\]

*Proof.* For any \( i \in [m] \), we have
\[
\tilde{\nabla}_x \psi(f_i(\mathbf{x}), z_i) = [\beta f_i(\mathbf{x}) + z_i]_+ \nabla f_i(\mathbf{x}).
\]
Note that \( \| \tilde{\nabla} f_i(\mathbf{x}) \| \leq G \) and \( [\beta f_i(\mathbf{x}) + z_i]_+ \leq 2\beta^2 F^2 + 2(z_i)^2 \). Hence,
\[
\| \tilde{\nabla}_x \psi(f_i(\mathbf{x}), z_i) \|^2 \leq [\beta f_i(\mathbf{x}) + z_i]_+ \| \tilde{\nabla} f_i(\mathbf{x}) \|^2 \leq 2G^2 (\beta^2 F^2 + (z_i)^2),
\]
which implies the desired result. \( \square \)

The lemma below can be directly verified from the definition of \( \psi \).
Lemma 3.4 For any \( x \in X \) such that \( f_i(x) \leq 0, \forall i \in [m] \) and any \( z \geq 0 \), it holds \( \Psi(x, z) \leq 0 \).

By the previous three lemmas, we establish an important result for running one iteration of Algorithm 1 and then use it to show the convergence rate results.

Lemma 3.5 (fundamental result) Under Assumption 2, let \((x, z)\) be any vector such that \( x \in X, f_i(x) \leq 0, \forall i \in [m] \) and \( z \geq 0 \). Then

\[
\mathbb{E}[\Phi(x^{k+1}; x, z)] + \frac{1}{2\alpha_k} \mathbb{E}\|x^{k+1} - x\|^2 + \frac{1}{2\rho_k} \mathbb{E}\|z^{k+1} - z\|^2 + \left(\frac{\beta_k}{2\rho_k^2} - \frac{1}{2\rho_k}\right) \mathbb{E}\|z^{k+1} - z^k\|^2 \leq \left(\frac{1}{2\alpha_k} - \frac{\mu}{2}\right) \mathbb{E}\|x^k - x\|^2 + \frac{1}{2\rho_k} \mathbb{E}\|z^k - z\|^2 + 2\alpha_k(4G^2 + \sigma^2) + 8\alpha_k \left(\beta_k F^2 G^2 + \frac{G^2}{m} \mathbb{E}\|z\|^2\right).
\]

Proof. From the update (5), it follows that for any \( x \in X \),

\[
\langle x^{k+1} - x, g_0^k \rangle + \hat{\nabla}_x \psi_{\beta_k}(f_{i_k}(x^k), z_{i_k}^k) + \frac{1}{\alpha_k}(x^{k+1} - x^k) \rangle \leq 0.
\]

We write

\[
\langle x^{k+1} - x, g_0^k \rangle = \langle x^{k+1} - x, \hat{\nabla}_0 f_0(x^{k+1}) \rangle - \langle x^{k+1} - x, \hat{\nabla}_0 f_0(x^{k+1}) - g_0^k \rangle + \langle x^k - x, g_0^k \rangle.
\]

Note \( \mathbb{E}[\langle x^k - x, g_0^k \rangle | \mathcal{W}^k] = \langle x^k - x, \hat{\nabla}_0 f_0(x^k) \rangle \), where \( \hat{\nabla}_0 f_0(x^k) = \mathbb{E}[g_0^k | \mathcal{W}^k] \in \partial f_0(x^k) \). Hence, from (26), it follows that

\[
\mathbb{E}\left[\langle x^{k+1} - x, g_0^k \rangle | \mathcal{W}^k\right] \geq \mathbb{E}\left[f_0(x^{k+1}) - f_0(x^k) - \langle x^{k+1} - x, \hat{\nabla}_0 f_0(x^{k+1}) - g_0^k \rangle | \mathcal{W}^k\right] + f_0(x^k) - f_0(x) + \frac{\mu}{2} \|x^k - x\|^2
\]

\[
= \mathbb{E}\left[f_0(x^{k+1}) - f_0(x) - \langle x^{k+1} - x, \hat{\nabla}_0 f_0(x^{k+1}) - g_0^k \rangle + \frac{\mu}{2} \|x^k - x\|^2 | \mathcal{W}^k\right].
\]

Similarly,

\[
\mathbb{E}\left[\langle x^{k+1} - x, \hat{\nabla}_x \psi_{\beta_k}(f_{i_k}(x^k), z_{i_k}^k) \rangle | \mathcal{W}^k\right] \geq \mathbb{E}\left[\Psi_{\beta_k}(x^{k+1}, z^k) - \Psi_{\beta_k}(x, z^k) - \langle x^{k+1} - x, \hat{\nabla}_x \Psi_{\beta_k}(x^{k+1}, z^k) - \hat{\nabla}_x \psi_{\beta_k}(f_{i_k}(x^k), z_{i_k}^k) \rangle | \mathcal{W}^k\right].
\]

In addition,

\[
\langle x^{k+1} - x, \frac{1}{\alpha_k} (x^{k+1} - x^k) \rangle = \frac{1}{2\alpha_k} \left[\|x^{k+1} - x\|^2 - \|x^k - x\|^2 + \|x^{k+1} - x^k\|^2\right].
\]

Taking expectation on both sides of (37) through (39) and substituting them into (36) gives

\[
\mathbb{E}\left[f_0(x^{k+1}) - f_0(x) - \langle x^{k+1} - x, \hat{\nabla}_0 f_0(x^{k+1}) - g_0^k \rangle \right] + \frac{1}{2\alpha_k} \mathbb{E}\left[\|x^{k+1} - x\|^2 + \|x^{k+1} - x^k\|^2\right]
\]

\[
+ \mathbb{E}\left[\Psi_{\beta_k}(x^{k+1}, z^k) - \Psi_{\beta_k}(x, z^k) - \langle x^{k+1} - x, \hat{\nabla}_x \Psi_{\beta_k}(x^{k+1}, z^k) - \hat{\nabla}_x \psi_{\beta_k}(f_{i_k}(x^k), z_{i_k}^k) \rangle \right]
\]

\[
\leq \left(\frac{1}{2\alpha_k} - \frac{\mu}{2}\right) \mathbb{E}\|x^k - x\|^2.
\]
Note
\[ \mathbb{E}\|\mathbf{\nabla} f_0(x^{k+1}) - g^k\|^2 \leq 2\mathbb{E}\left(\|\mathbf{\nabla} f_0(x^{k+1}) - \mathbb{E}[g^k]\|^2 + \|g^k - \mathbb{E}[g^k]\|^2 \right) \leq 8G^2 + 2\sigma^2. \]
Hence, by the Young’s inequality, we have
\[ -\frac{1}{4\alpha_k}\mathbb{E}\|x^{k+1} - x^k\|^2 - \alpha_k(8G^2 + 2\sigma^2) \leq -\mathbb{E}\langle x^{k+1} - x^k, \mathbf{\nabla} f_0(x^{k+1}) - g^k \rangle. \tag{41} \]
In addition, since \( i_k \) is chosen from \([m]\) uniformly at random, it holds that
\[ \mathbb{E}\|\mathbf{\nabla}_x \Psi \beta_k(x^{k+1}, z^k) - \mathbf{\nabla}_x \psi \beta_k(f_i(x^k), z^k_i)\|^2 \leq \frac{1}{m}\sum_{i=1}^m \mathbb{E}\left\|\mathbf{\nabla}_x \psi \beta_k(f_i(x^k), z^k_i)\right\|^2 + \frac{2}{m}\sum_{i=1}^m \mathbb{E}\left\|\mathbf{\nabla}_x \psi \beta_k(f_i(x^k), z^k_i)\right\|^2 \]
\[ = \mathbb{E}\left\|\frac{1}{m}\sum_{i=1}^m \mathbf{\nabla}_x \psi \beta_k(f_i(x^{k+1}), z^k_i)\right\|^2 + \frac{2}{m}\sum_{i=1}^m \mathbb{E}\left\|\mathbf{\nabla}_x \psi \beta_k(f_i(x^k), z^k_i)\right\|^2 \]
\[ \leq \mathbb{E}\left\|\frac{1}{m}\sum_{i=1}^m \mathbf{\nabla}_x \psi \beta_k(f_i(x^{k+1}), z^k_i)\right\|^2 + \frac{2}{m}\sum_{i=1}^m \mathbb{E}\left\|\mathbf{\nabla}_x \psi \beta_k(f_i(x^k), z^k_i)\right\|^2, \]
where the equality follows because \( \frac{1}{m}\sum_{i=1}^m \mathbf{\nabla}_x \psi \beta_k(f_i(x^{k+1}), z^k_i) \in \partial_x \Psi \beta_k(x^{k+1}, z^k). \) Hence, from Lemma 3.3 and the Young’s inequality, it follows that
\[ -\frac{1}{4\alpha_k}\mathbb{E}\|x^{k+1} - x^k\|^2 - \alpha_k\left(8\beta_k^2 F^2 G^2 + \frac{8G^2}{m}\mathbb{E}\|z^k\|^2\right) \leq -\mathbb{E}\langle x^{k+1} - x^k, \mathbf{\nabla} \Psi \beta_k(x^{k+1}, z^k) - \mathbf{\nabla} \psi \beta_k(f_i(x^k), z^k_i) \rangle. \tag{42} \]
Taking expectation on both sides of (31), adding it and also (41) and (42) into (40), and using Lemma 3.4 yield the desired result.

### 3.2 Convergence rate for convex problems

In this subsection, we establish the convergence rate of Algorithm 1 for convex problems, i.e., \( \mu = 0. \) Different from existing analysis for saddle-point problems, we do not assume the boundedness of the dual variable \( z \) but instead we can bound \( z^k \) in expectation.

**Proposition 3.1** Under Assumptions 1 and 2, let \( \{(x^k, z^k)\} \) be the sequence generated from Algorithm 1 with parameters satisfying
\[ \frac{\rho_k}{\alpha_k} \geq \rho_{k+1} \left(\frac{1}{\alpha_{k+1}} - \mu\right), \quad \beta_k \geq \rho_k, \forall k \geq 1 \tag{43} \]
then for any \( t \geq 1, \) it holds that
\[ \mathbb{E}\|z^{t+1}\|^2 \leq 2\rho_1 \left(\frac{1}{\alpha_1} - \mu\right) \|x^1 - x^*\|^2 + 4\|z^*\|^2 + \sum_{k=1}^t 8\alpha_k \rho_k \left(4G^2 + \sigma^2 + 4\beta_k^2 F^2 G^2 + \frac{4G^2}{m}\mathbb{E}\|z^k\|^2\right), \tag{44} \]
where \( (x^*, z^*) \) is any point satisfying the KKT conditions in (23).
Proof. Multiplying $2\rho_k$ to both sides of (35) with $(x, z) = (x^*, z^*)$ gives

$$2\rho_k \mathbb{E} \left[ \Phi(x^{k+1}; x^*, z^*) \right] + \frac{\rho_k}{\alpha_k} \mathbb{E} \|x^{k+1} - x^*\|^2 + \mathbb{E} \|z^{k+1} - z^*\|^2 + \left( \frac{\beta_k}{\rho_k} - 1 \right) \mathbb{E} \|z^{k+1} - z^k\|^2 \leq \left( \frac{\rho_k}{\alpha_k} - \mu \rho_k \right) \mathbb{E} \|x^k - x^*\|^2 + \sum_{i=k}^{t} \mathbb{E} \|z^k - z^*\|^2 + 2\alpha_k \rho_k \left( 4G^2 + \sigma^2 + 4\beta_k^2 F^2 G^2 + \frac{4G^2}{m} \mathbb{E} \|z^k\|^2 \right).$$

Summing the above inequality from $k = 1$ through $t$, we have by noting $\Phi(x^{k+1}; x^*, z^*) \geq 0, \forall k$ and using the conditions in (43) that

$$\mathbb{E} \|x^{t+1} - x^*\|^2 \leq \rho_1 \left( \frac{1}{\alpha_1} - \mu \right) \mathbb{E} \|x^1 - x^*\|^2 + \mathbb{E} \|z^t - z^*\|^2 + \sum_{k=1}^{t} 4\alpha_k \rho_k \left( 4G^2 + \sigma^2 + 4\beta_k^2 F^2 G^2 + \frac{4G^2}{m} \mathbb{E} \|z^k\|^2 \right).$$

From the Young’s inequality, it follows that $\|z^{t+1}\|^2 \leq 2\|z^{t+1} - z^*\|^2 + 2\|z^*\|^2$, which together with the above inequality gives the desired result. \qed

Below we specify the parameters and bound $\mathbb{E} \|z^k\|^2$.

**Proposition 3.2 (pre-determined maximum iterations)** Given a positive integer $K$, set

$$\alpha_k = \frac{\alpha}{\sqrt{K}}, \rho_k = \frac{\rho}{\sqrt{K}}, \beta_k = \rho_k, \forall k \leq K,$$

where $\alpha$ and $\rho$ are positive scalars satisfying $\alpha \rho < \frac{m}{32G^2}$. Then for any $1 \leq k \leq K$, it holds that

$$\mathbb{E} \|z^k\|^2 \leq \frac{C_1}{1 - \frac{32\alpha \rho G^2}{m}}$$

where

$$C_1 = \frac{2\rho}{\alpha} \mathbb{E} \|x^1 - x^*\|^2 + 4\mathbb{E} \|z^*\|^2 + 8\alpha \rho (4G^2 + \sigma^2) + \frac{32\alpha \rho^3 F^2 G^2}{K}. \quad (47)$$

**Proof.** It is easy to see that the parameters given in (45) satisfy the conditions in (43). Hence, for any $t < K$, it follows from (44) that

$$\mathbb{E} \|z^{t+1}\|^2 \leq \frac{2\rho}{\alpha} \mathbb{E} \|x^1 - x^*\|^2 + 4\mathbb{E} \|z^*\|^2 + 8\alpha \rho (4G^2 + \sigma^2) + \frac{32\alpha \rho^3 F^2 G^2}{K} + \frac{32\alpha \rho G^2}{mK} \sum_{k=1}^{t} \mathbb{E} \|z^k\|^2. \quad (48)$$

Now we show the result in (46) by induction. Since $z^1 = 0$, (46) holds trivially for $k = 1$. Assume it holds for $k \leq t$. Then from (48), it follows that

$$\mathbb{E} \|z^{t+1}\|^2 \leq C_1 + \frac{32\alpha \rho G^2}{mK} \sum_{k=1}^{t} \frac{C_1}{1 - \frac{32\alpha \rho G^2}{m}} \leq \frac{C_1}{1 - \frac{32\alpha \rho G^2}{m}}. \quad (49)$$
which completes the proof.

If the maximum number of iterations is not pre-determined, we set parameters adaptive to iteration numbers and can still bound $\mathbb{E}\|z^k\|^2$.

**Proposition 3.3 (varying maximum iterations)** Let $\{ (x^k, z^k) \}$ be the sequence generated from Algorithm 1 with parameters set to

$$
\alpha_k = \frac{\alpha}{\sqrt{k + 1 \log(k + 1)}}, \quad \rho_k = \frac{\rho}{\sqrt{k + 1 \log(k + 1)}}, \quad \beta_k = \rho_k, \quad \forall k \geq 1,
$$

where $\alpha$ and $\rho$ are chosen such that $\alpha \rho < \frac{m}{68G^2}$. Then for any $k \geq 1$, it holds that

$$
\mathbb{E}\|z^k\|^2 \leq \frac{C_2}{1 - \frac{68\alpha \rho G^2}{m}}
$$

where

$$
C_2 = \frac{2\rho}{\alpha} \|x^1 - x^*\|^2 + 4\|z^*\|^2 + 17\alpha \rho (4G^2 + \sigma^2) + 39\alpha \rho^3 F^2 G^2.
$$

**Proof.** It is easy to see that the parameters given in (49) satisfy the conditions in (43). Hence, plugging the specified parameters into (44) gives

$$
\mathbb{E}\|z^{t+1}\|^2 \leq \frac{2\rho}{\alpha} \|x^1 - x^*\|^2 + 4\|z^*\|^2 + \sum_{k=1}^{t} \frac{8\alpha \rho}{(k + 1)(\log(k + 1))^2} (4G^2 + \sigma^2) + \sum_{k=1}^{t} \frac{32\alpha \rho}{(k + 1)(\log(k + 1))^2} \left( \frac{\rho^2}{(k + 1)(\log(k + 1))^2} F^2 G^2 + \frac{G^2}{m} \mathbb{E}\|z^k\|^2 \right).
$$

Note that

$$
\sum_{k=1}^{t} \frac{1}{(k + 1)(\log(k + 1))^2} \leq 2.1, \quad \sum_{k=1}^{t} \frac{1}{(k + 1)^2(\log(k + 1))^4} \leq 1.2.
$$

Hence, it follows from (52) that

$$
\mathbb{E}\|z^{t+1}\|^2 \leq \frac{2\rho}{\alpha} \|x^1 - x^*\|^2 + 4\|z^*\|^2 + 17\alpha \rho (4G^2 + \sigma^2) + 39\alpha \rho^3 F^2 G^2 + \sum_{k=1}^{t} \frac{32\alpha \rho}{(k + 1)(\log(k + 1))^2} \frac{G^2}{m} \mathbb{E}\|z^k\|^2.
$$

Now we show the result in (50) by induction. When $k = 1$, it obviously holds. Assume the result holds for $k \leq t$. Then from (54), it follows that

$$
\mathbb{E}\|z^{t+1}\|^2 \leq C_2 + \sum_{k=1}^{t} \frac{32\alpha \rho}{(k + 1)(\log(k + 1))^2} \frac{G^2}{m} \frac{C_2}{1 - \frac{68\alpha \rho G^2}{m}} \leq C_2 + \frac{68\alpha \rho G^2}{m} \frac{C_2}{1 - \frac{68\alpha \rho G^2}{m}} = \frac{C_2}{1 - \frac{68\alpha \rho G^2}{m}}.
$$
where the second inequality uses (53). This completes the proof. □

Using Lemma 3.5 and also the boundedness of $E\|z^k\|^2$, we are now ready to show the convergence rate results for the case $\mu = 0$.

**Theorem 3.1 (Convergence rate for convex case)** Under Assumptions 1 and 2, let $\{(x^k, z^k)\}$ be the sequence generated from Algorithm 1. Then we have:

1. Given any positive integer $K$, if the parameters are set according to (45), then
   \[
   \mathbb{E}|f_0(x^{K+1}) - f_0(x^*)| \leq \frac{1}{\sqrt{K}} \left( 2\phi_1 + \frac{9}{2\rho} \|z^*\|^2 \right),
   \]
   \[
   \mathbb{E} \left[ \frac{1}{m} \sum_{i=1}^m [f_i(x^{K+1})]_+ \right] \leq \frac{1}{\sqrt{K}} \left( \phi_1 + \frac{\|1 + z^*\|^2}{2\rho} \right),
   \]
   where $x^{K+1} = \frac{1}{K} \sum_{k=1}^K x^{k+1}$, and
   \[
   \phi_1 = \frac{1}{2\alpha} \|x^1 - x^*\|^2 + 2\alpha(4G^2 + \sigma^2) + 8\alpha \left( \frac{\rho^2 F^2 G^2}{K} + \frac{G^2 C_1}{m - 32\alpha \rho G^2} \right),
   \]
   with $C_1$ defined in (47).

2. If the parameters are set according to (49), then for any $K \geq 1$
   \[
   \mathbb{E}|f_0(x^{K+1}) - f_0(x^*)| \leq \frac{\log(K + 1)}{2(\sqrt{K} + 2 - \sqrt{2})} \left( 2\phi_2 + \frac{9}{2\rho} \|z^*\|^2 \right),
   \]
   \[
   \mathbb{E} \left[ \frac{1}{m} \sum_{i=1}^m [f_i(x^{K+1})]_+ \right] \leq \frac{\log(K + 1)}{2(\sqrt{K} + 2 - \sqrt{2})} \left( \phi_2 + \frac{1}{2\rho} \|1 + z^*\|^2 \right),
   \]
   where $x^{K+1} = \frac{1}{\sum_{k=1}^K \alpha_k} \sum_{k=1}^K \alpha_k x^{k+1}$, and
   \[
   \phi_2 = \frac{1}{2\alpha} \|x^1 - x^*\|^2 + 5\alpha(4G^2 + \sigma^2) + 10\alpha \rho^2 F^2 G^2 \frac{17\alpha G^2 C_2}{m - 68\alpha \rho G^2},
   \]
   with $C_2$ defined in (51).

**Remark 3.1** By the Markov’s inequality $\text{Prob}(\xi \geq \varepsilon) \leq \frac{\mathbb{E}[\xi]}{\varepsilon}$, one can easily have a high-probability result from the theorem. This remark also applies to Theorem 3.2.

**Proof.** When the parameters are set according to (45), we have (46). Hence, letting $x = x^*$ in (35) and summing it from $k = 1$ through $K$ give

\[
\sum_{k=1}^K \mathbb{E}[\Phi(x^{k+1}; x^*, z)] \leq \frac{\sqrt{K}}{2\alpha} \|x^1 - x^*\|^2 + \frac{\sqrt{K}}{2\rho} \|z\|^2 + 2\alpha(4G^2 + \sigma^2) \sqrt{K} + \frac{8\alpha \rho^2 F^2 G^2}{\sqrt{K}} + \frac{8\alpha G^2 C_1 \sqrt{K}}{m(1 - 32\alpha \rho G^2/m)},
\]

(57)
By the convexity of $f_i$’s and $z \geq 0$, we have $\Phi(\bar{x}_K^{K+1}; \bar{x}^*, z) \leq \frac{1}{K} \sum_{k=1}^{K} \Phi(x_k^{k+1}; x^*, z)$. Hence, it follows from (57) that

$$
\mathbb{E}[\Phi(\bar{x}_K^{k+1}; x^*, z)] \leq \frac{1}{\sqrt{K}} \left( \phi_1 + \frac{1}{2\rho} \|z\|^2 \right).
$$

(58)

Therefore, from Lemma 3.1, we have the results in (55).

When the parameters are set according to (49), we have (50). Hence, multiplying $\alpha_k$ to both sides of (35) with $x = x^*$, summing it over $k$, and also using (53), we have

$$
\sum_{k=1}^{K} \alpha_k \mathbb{E} [\Phi(x_k^{k+1}; x^*, z)] \leq \frac{1}{2} \|x^1 - x^*\|^2 + \frac{\alpha}{2\rho} \|z\|^2 + 5\alpha^2 (4G^2 + \sigma^2) + 10\alpha^2 \rho^2 F^2 G^2 + \frac{17\alpha^2 G^2 C_2}{m(1 - \frac{68\alpha \rho G}{m})}.
$$

(59)

Note

$$
\sum_{k=1}^{K} \alpha_k = \sum_{k=1}^{K} \frac{\alpha}{\sqrt{k+1} \log(k+1)} \geq \frac{\alpha}{\log(K+1)} \int_1^{K+1} \frac{1}{\sqrt{x+1}} dx = \frac{2\alpha(\sqrt{K+2} - \sqrt{2})}{\log(K+1)}.
$$

In addition, by the convexity of $f_i$’s and $z \geq 0$, we have

$$
\Phi(\bar{x}_K^{k+1}; x^*, z) \leq \frac{1}{\sum_{k=1}^{K} \alpha_k} \sum_{k=1}^{K} \alpha_k \Phi(x_k^{k+1}; x^*, z).
$$

Hence, it follows from (59) that

$$
\mathbb{E}[\Phi(\bar{x}_K^{k+1}; x^*, z)] \leq \frac{\log(K+1)}{2(\sqrt{K+2} - \sqrt{2})} \left( \phi_2 + \frac{1}{2\rho} \|z\|^2 \right),
$$

which together with Lemma 3.5 gives the desired results in (56). This completes the proof. □

### 3.3 Convergence rate for strongly convex problems

In this subsection, we analyze the convergence rate of Algorithm 1 for strongly convex problems, i.e., $\mu > 0$ in (26). Similar to the convex case, we first bound $\mathbb{E}\|z^k\|^2$ by choosing appropriate parameters.

**Proposition 3.4** Assume $f_0$ to be strongly convex with modulus $\mu > 0$. For any given positive integer $K$, let $\{(x^k, z^k)\}$ be the sequence generated from Algorithm 1 with parameters set to

$$
\alpha_k = \alpha \frac{k}{k+1}, \quad \rho_k = \frac{\rho}{\log(K+1)}, \quad \beta_k = \rho,
$$

(60)
where \( \alpha \geq \frac{1}{\mu} \), and \( \alpha \rho < \frac{m}{32G^2} \). Then for any \( k \leq K \),

\[
E\|z^k\|^2 \leq \frac{C_3}{1 - \frac{32\alpha \rho G^2}{m}},
\]  

(61)

where

\[
C_3 = \frac{2\rho}{\log(K + 1)} \left( \frac{2}{\alpha} - \mu \right) \|x^1 - x^*\|^2 + 4\|z^*\|^2 + 8\alpha \rho (4G^2 + \sigma^2) + \frac{32\alpha \rho^3 F^2 G^2}{m \log(K + 1)^2}.
\]  

(62)

**Proof.** If \( \alpha \geq \frac{1}{\mu} \), then \( \frac{k}{\alpha} \geq \frac{k + 1}{\alpha} - \mu \). Hence, the parameters given in (60) satisfies the condition in (43), thus (44) holds and, with the specified parameters, becomes

\[
E\|z^{t+1}\|^2 \leq \frac{2\rho}{\log(K + 1)} \left( \frac{2}{\alpha} - \mu \right) \|x^1 - x^*\|^2 + 4\|z^*\|^2
\]  

(63)

\[+ \sum_{k=1}^{t} \frac{8\alpha \rho}{(k + 1) \log(K + 1)} \left( 4G^2 + \sigma^2 + \frac{4\rho^2 F^2 G^2}{m \log(K + 1)^2} + \frac{4G^2}{m} E\|z^k\|^2 \right). \]

Note that for any \( t \leq K \),

\[
\sum_{k=1}^{t} \frac{1}{k + 1} \leq \int_{1}^{t} \frac{1}{x} dx = \log t \leq \log(K + 1).
\]  

(64)

Hence, (63) implies

\[
E\|z^{t+1}\|^2 \leq C_3 + \sum_{k=1}^{t} \frac{8\alpha \rho}{(k + 1) \log(K + 1)} \frac{4G^2}{m} E\|z^k\|^2.
\]  

(65)

Now we show (61) by induction. When \( k = 1 \), it obviously holds since \( z^1 = 0 \). Assume (61) holds for any \( k \leq t \leq K \). Then, from (64) and (65), it follows that

\[
E\|z^{t+1}\|^2 \leq C_3 + \frac{32\alpha \rho G^2}{m} \left( \frac{C_3}{1 - \frac{32\alpha \rho G^2}{m}} \sum_{k=1}^{t} \frac{1}{(k + 1) \log(K + 1)} \right) \leq \frac{C_3}{1 - \frac{32\alpha \rho G^2}{m}},
\]

which completes the proof. \( \square \)

Using (35) and (61), we establish the convergence rate result of Algorithm 1 for the case \( \mu > 0 \) as follows.

**Theorem 3.2 (convergence rate for strongly convex case)** Under the assumptions of Proposition 3.4, we have

\[
E\|x^{K+1} - x^*\|^2 \leq \frac{2\alpha \log(K + 1)}{K + 1} \left( \phi_3 + \frac{1}{2\rho} \|z^*\|^2 \right).
\]  

(66)
where
\[ \phi_3 = \frac{2 - \alpha \mu}{2\alpha \log(K + 1)} ||x^1 - x^*||^2 + 2\alpha(4G^2 + \sigma^2) + 8\alpha \left( \frac{\rho^2 F^2 G^2}{\log(K + 1)^2} + \frac{G^2 C_3}{m - 32\alpha \rho G^2} \right), \]
with \( C_3 \) defined in (62). In addition,
\[ \mathbb{E}[f_0(\bar{x}^{K+1}) - f_0(x^*)] \leq \frac{\log(K + 1)}{K} \left( 2\phi_3 + \frac{9}{2\rho} ||z^*||^2 \right), \]
\[ \mathbb{E} \left[ \frac{1}{m} \sum_{i=1}^{m} [f_i(\bar{x}^{K+1})]_+ \right] \leq \frac{\log(K + 1)}{K} \left( \phi_3 + \frac{1}{2\rho} ||1 + z^*||^2 \right), \]
where \( \bar{x}^{K+1} = \frac{\sum_{k=1}^{K} x^{k+1}}{K} \).

Proof. Letting \( x = x^* \) in (35) and summing it from \( k = 1 \) through \( K \) yields
\[ \sum_{k=1}^{K} \mathbb{E}[\Phi(x^{k+1}; x^*, z)] + \frac{1}{2\alpha} \mathbb{E}||x^{K+1} - x^*||^2 + \frac{1}{2\rho} \mathbb{E}||z^{K+1} - z||^2 \]
\[ \leq \left( \frac{1}{2\alpha_1} - \frac{\mu}{2} \right) ||x^1 - x^*||^2 + \frac{1}{2\rho_1} \mathbb{E}||z||^2 + \sum_{k=1}^{K} 2\alpha_k (4G^2 + \sigma^2) + \sum_{k=1}^{K} 8\alpha_k \left( \beta_k^2 F^2 G^2 + \frac{G^2}{m} \mathbb{E}||z||^2 \right). \]
Plugging into the above inequality the parameters given in (60), letting \( z = z^* \), and using (25) and (64), we have
\[ \frac{K + 1}{2\alpha} \mathbb{E}||x^{K+1} - x^*||^2 \leq \left( \frac{1}{\alpha} - \frac{\mu}{2} \right) ||x^1 - x^*||^2 \]
\[ + \log(K + 1) \left[ \frac{1}{2\rho} ||z^*||^2 + 2\alpha(4G^2 + \sigma^2) + 8\alpha \left( \frac{\rho^2 F^2 G^2}{\log(K + 1)^2} + \frac{G^2}{m - 32\alpha \rho G^2} \right) \right], \]
which implies (66).

Dropping \( ||x^{K+1} - x^*||^2 \) and \( ||z^{K+1} - z||^2 \) terms on the right hand side of (68), we have
\[ \sum_{k=1}^{K} \mathbb{E}[\Phi(x^{k+1}; x^*, z)] \leq \log(K + 1) \left( \phi_3 + \frac{1}{2\rho} \mathbb{E}||z||^2 \right). \]
From the convexity of \( f_i \)'s, it follows that \( \Phi(\bar{x}^{K+1}; x^*, z) \leq \frac{1}{K} \sum_{k=1}^{K} \Phi(x^{k+1}; x^*, z) \), and thus
\[ \mathbb{E}[\Phi(\bar{x}^{K+1}; x^*, z)] \leq \frac{\log(K + 1)}{K} \left( \phi_3 + \frac{1}{2\rho} \mathbb{E}||z||^2 \right), \]
which together with Lemma 3.1 gives the results in (67). \( \square \)

Remark 3.2 For the strongly convex case, if the maximum number \( K \) is not given, we can set \( \alpha_k = \frac{\alpha}{k+1}, \rho_k = \frac{\rho}{\log(k+1)}, \beta_k = \rho_k, \forall k \). These parameters satisfy the condition in (43), and thus we can still have a sublinear convergence result through first bounding \( \mathbb{E}||z||^2 \). However, there will be an additional \( \log(K + 1) \) term in the obtained result, i.e., \( O(\log(K + 1)^2/(K + 1)) \) for any positive integer \( K \).
4 Numerical experiments

In this section, we test the proposed method on solving a quadratically constrained quadratic program (QCQP):

\[
\min_{x \in X} \frac{1}{2N} \sum_{i=1}^{N} \|H_i x - c_i\|^2, \quad \text{s.t. } \frac{1}{2} x^\top Q_j x + a_j^\top x \leq b_j, j = 1, \ldots, m.
\] (70)

Here \(X = [-10, 10]^n\); for each \(i \in [N]\), \(H_i \in \mathbb{R}^{p \times n}\) and \(c_i\) are randomly generated with components independently following standard normal distribution; the entries of every \(a_j\) also follow standard Gaussian distribution; \(Q_i\)’s are randomly generated symmetric positive semidefinite matrices; each \(b_j\) is generated according to uniform distribution on \([0, 1.1]\). Note that for the generated data, the Slater’s condition holds, and thus there must exist a KKT point for (70). Let \(\xi\) be a random variable with uniform distribution on \([N]\). Then the objective of (70) can be written to \(E_{\xi} \frac{1}{2} \|H_{\xi} x - c_{\xi}\|^2\), and thus (70) is in the form of (1).

In our experiment, we set \(N = m = 10,000\) and \(n = 100, p = n - 5\), in which case (70) is not strongly convex. The algorithm parameters are set according to (45) with \(\alpha = 1, \rho = 1\) and \(K = m \times \#\text{total epoch}\), where each epoch is equivalent to using \(m\) constraint functions once. For comparison purpose, we also apply the stochastic mirror-prox method in [7] on the QCQP problem (70). Projecting onto the set \(\{x : \frac{1}{2} x^\top Q x + a^\top x \leq b\}\) does not generally admit an analytic solution and requires an iterative method. Hence, the methods in [16,18] could be potentially inefficient on solving the QCQP problem. The stochastic mirror-prox method requires the dual variable within a bounded set. To have an estimated set, we first run the proposed algorithm and let \(z^*\) be the output of the dual variable. Then we use \(Z = \{z : \|z\|_\infty \leq z_{\max}\}\) as a constraint set of the dual variable for the stochastic mirror-prox method, where \(z_{\max} = \max (10, 10\|z^*\|_\infty)\). Figure 1 shows the objective error and also feasibility violation in terms of epoch, where the optimal solution is computed by CVX [6]. From the results, we see that the proposed algorithm performs significantly better than the stochastic mirror-prox method. One possible reason is that the former method performs Gauss-Seidel type update while the latter uses Jacobi-type update, and another reason could be the estimated set \(Z\) is too large and causes slow convergence.

5 Conclusions

We have proposed a primal-dual stochastic gradient method for stochastic programming with many functional constraints. Every iteration, the method only need to inquire an oracle to obtain a stochastic subgradient of the objective, a subgradient and function value of one randomly sampled constraint function, and the function value of another sampled constraint function. Under standard assumptions, we have established its convergence rate for both convex and strongly convex problems. The order of rate is optimal for convex case and nearly optimal for strongly convex case.
Figure 1: Results given by Algorithm 1 on solving an instance of the quadratically constrained quadratic programming (70). Left: the distance of objective value to optimal value $|f_0(x) - f_0(x^*)|$; Right: the error of constraint violation $\frac{1}{m} \sum_{j=1}^{m} [f_j(x)]_+$.

Numerical experiments on quadratically constrained quadratic programming demonstrate its nice practical performance.

Acknowledgements

The author would like to thank Professor Wotao Yin for his valuable comments and suggestions, which greatly improves the paper.

References

[1] M. Baes, M. Brgisser, and A. Nemirovski. A randomized mirror-prox method for solving structured large-scale matrix saddle-point problems. *SIAM Journal on Optimization*, 23(2):934–962, 2013. 5, 9

[2] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty. *Nonlinear programming: theory and algorithms*. John Wiley & Sons, 2013. 9

[3] G. Calafiore and M. C. Campi. Uncertain convex programs: randomized solutions and confidence levels. *Mathematical Programming*, 102(1):25–46, 2005. 4

[4] M. C. Campi and S. Garatti. A sampling-and-discarding approach to chance-constrained optimization: feasibility and optimality. *Journal of Optimization Theory and Applications*, 148(2):257–280, 2011. 4
[5] A. Cotter, M. Gupta, and J. Pfeifer. A light touch for heavily constrained sgd. In *Conference on Learning Theory*, pages 729–771, 2016.

[6] M. Grant, S. Boyd, and Y. Ye. CVX: Matlab software for disciplined convex programming, 2008.

[7] A. Juditsky, A. Nemirovski, C. Tauvel, et al. Solving variational inequalities with stochastic mirror-prox algorithm. *Stochastic Systems*, 1(1):17–58, 2011.

[8] G. Lan and Z. Zhou. Algorithms for stochastic optimization with expectation constraints. *arXiv preprint arXiv:1604.03887*, 2016.

[9] J. Luedtke and S. Ahmed. A sample approximation approach for optimization with probabilistic constraints. *SIAM Journal on Optimization*, 19(2):674–699, 2008.

[10] M. Mahdavi, T. Yang, R. Jin, S. Zhu, and J. Yi. Stochastic gradient descent with only one projection. In *Advances in Neural Information Processing Systems*, pages 494–502, 2012.

[11] A. Nedić and A. Ozdaglar. Subgradient methods for saddle-point problems. *Journal of optimization theory and applications*, 142(1):205–228, 2009.

[12] A. Nemirovski and A. Shapiro. Scenario approximations of chance constraints. In *Probabilistic and randomized methods for design under uncertainty*, pages 3–47. Springer, 2006.

[13] B. Palaniappan and F. Bach. Stochastic variance reduction methods for saddle-point problems. In *Advances in Neural Information Processing Systems*, pages 1416–1424, 2016.

[14] P. Rigollet and X. Tong. Neyman-pearson classification, convexity and stochastic constraints. *Journal of Machine Learning Research*, 12(Oct):2831–2855, 2011.

[15] R. T. Rockafellar. A dual approach to solving nonlinear programming problems by unconstrained optimization. *Mathematical programming*, 5(1):354–373, 1973.

[16] E. K. Ryu and W. Yin. Proximal-proximal-gradient method. *arXiv preprint arXiv:1708.06908*, 2017.

[17] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on stochastic programming: modeling and theory*. SIAM, 2009.

[18] M. Wang and D. P. Bertsekas. Stochastic first-order methods with random constraint projection. *SIAM Journal on Optimization*, 26(1):681–717, 2016.

[19] M. Wang, Y. Chen, J. Liu, and Y. Gu. Random multi-constraint projection: Stochastic gradient methods for convex optimization with many constraints. *arXiv preprint arXiv:1511.03760*, 2015.

[20] X. Wang, S. Ma, and Y. Yuan. Penalty methods with stochastic approximation for stochastic nonlinear programming. *Mathematics of Computation*, 86(306):1793–1820, 2017.
[21] Y. Xu. First-order methods for constrained convex programming based on linearized augmented lagrangian function. *arXiv preprint arXiv:1711.08020*, 2017. 2, 7, 10

[22] Y. Xu. Global convergence rates of augmented lagrangian methods for constrained convex programming. *arXiv preprint arXiv:1711.05812*, 2017. 10

[23] H. Yu and M. J. Neely. A primal-dual type algorithm with the $O(1/t)$ convergence rate for large scale constrained convex programs. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 1900–1905. IEEE, 2016. 7

[24] H. Yu and M. J. Neely. A primal-dual parallel method with $O(1/\epsilon)$ convergence for constrained composite convex programs. *arXiv preprint arXiv:1708.00322*, 2017. 7