On sampling Kaczmarz–Motzkin methods for solving large-scale nonlinear systems

Feiyu Zhang\textsuperscript{1} · Wendi Bao\textsuperscript{1} · Weiguo Li\textsuperscript{1} · Qin Wang\textsuperscript{1}

Received: 26 May 2022 / Revised: 12 February 2023 / Accepted: 5 March 2023 / Published online: 23 March 2023
© The Author(s) under exclusive licence to Sociedade Brasileira de Matemática Aplicada e Computacional 2023

Abstract
In this paper, for solving large-scale nonlinear equations, we propose a nonlinear sampling Kaczmarz–Motzkin (NSKM) method. Based on the local tangential cone condition and the Jensen’s inequality, we prove convergence of our method with two different assumptions. Then, for solving nonlinear equations with the convex constraints, we present two variants of the NSKM method: the projected sampling Kaczmarz–Motzkin (PSKM) method and the accelerated projected sampling Kaczmarz–Motzkin (APSKM) method. With the use of the nonexpansive property of the projection and the convergence of the NSKM method, the convergence analysis is obtained. Numerical results show that the NSKM method with the sample of the suitable size outperforms the nonlinear randomized Kaczmarz method in terms of calculation times. The APSKM and PSKM methods are practical and promising for the constrained nonlinear problem.

Keywords Large-scale nonlinear equations · Finite convex constraints · Sampling Kaczmarz–Motzkin method · Projection method · Randomized accelerated projection method

Mathematics Subject Classification 65H10 · 65F20 · 65J20
1 Introduction

Consider the nonlinear equations with finite convex constraints

\[ f(x) = 0 \quad \text{subject to} \quad x \in C, \tag{1} \]

where \( f : \mathcal{D}(f) \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m \) is a nonlinear vector-valued function, \( x \in \mathbb{R}^n \) is an unknown vector, and \( C \) is a nonempty intersection of finite nonempty closed convex sets \( C_i \), i.e.,

\[ C = \bigcap_{i=1}^{k_c} C_i \] (\( k_c \) is potentially a large number). If \( x^* \in \mathbb{R}^n \) exists, such that \( f(x^*) = 0 \) and \( x^* \in C \), then \( x^* \) is a solution of (1). Such problems arise from many practical applications, e.g., electrical impedance tomography, circuit problems, and chemical equilibrium systems.

When \( C = \mathbb{R}^n \), the system (1) is an unconstrained problem. This problem has attracted widespread attention. Many computational methods have been proposed, for example, Newton method (Kelley 1995), Quasi–Newton method (Dennis and Moré 1977), Gauss–Newton method (Li and Fukushima 1999), and Levenberg–Marquardt method (Yamashita and Fukushima 2001). These methods require the information of the whole nonlinear system, which may needs a large amount of computation for solving a large-scale nonlinear system. Stochastic gradient descent (SGD) method (Jin et al. 2020) requires only evaluating one randomly selected nonlinear equation at each iteration, instead of the whole nonlinear system, which substantially reduces the computational cost per iteration and enables excellent to deal with the large-scale problems. Recently, Wang et al. (2022) proposed a class of randomized Kaczmarz algorithms for solving large-scale nonlinear equations with specific assumptions, which only needs to calculate one row of the Jacobian matrix instead of the entire Jacobian matrix at each iteration and reduced the amount of calculation and storage. In addition, numerical results showed that algorithms proposed in Wang et al. (2022) are superior to the SGD algorithm. However, in the nonlinear randomized Kaczmarz (NRK) method (Wang et al. 2022), to determine the row index of the Jacobian matrix, all entries of \( f(x) \) need to be calculated at each step, which is expensive and inefficient when the size of \( f(x) \) is very large. To overcome the problem, De Loera et al. (2017) presented the sampling Kaczmarz–Motzkin (SKM) method for solving large-scale systems of linear inequalities, which only needs to compute a portion of the residuals at each step.

In general, the system (1) is a large-scale nonlinear problem with a large number of convex constraints. Based the SGD method, Wang and Bertsekas (2013) studied the stochastic gradient descent method with a single random projection (PSGD). They randomly picked one out of all constraint sets and found the projection onto it after using stochastic gradient descent at each iteration. Meanwhile, using a linear combination of several projections, Qin and Etesami (2020) devised a randomized accelerated projection algorithm, which has the faster convergence rate than the classic cyclic projection method.

In this paper, motivated by De Loera et al. (2017), we first present a nonlinear sampling Kaczmarz–Motzkin (NSKM) method for an unconstrained problem (1) and establish the corresponding convergence theory with two different assumptions. Preliminary numerical results show that the NSKM method is more effective to solve the large-scale nonlinear equations than the NRK method in terms of calculation times. Then, inspired by the ideas in Wang and Bertsekas (2013), Qin and Etesami (2020) and the NSKM method, for the system (1) with convex constraints, we propose the projected sampling Kaczmarz–Motzkin (PSKM) method and the accelerated projected sampling Kaczmarz–Motzkin (APSKM) method. Applying the nonexpansive property of the projection and the local tangential cone condition of the system, we obtain the convergence analyses of the two new methods. The numerical results confirm...
that the PSKM and APSKM methods have advantages over the PSGD method in terms of the number of iteration steps and calculation times.

The remaining part of the paper is organized as follows. In Sect. 2, we present the NSKM method and analyzed its convergence. In Sect. 3, we extend the NSKM method to get two variants of the NSKM method and prove the convergences of these methods. In Sect. 4, we provide some numerical experiments to display the practical performances of the proposed methods. In Sect. 5, we discuss the optimal choice for the number of samples. Finally, we finish this paper with a conclusion.

Throughout the paper, we use $|\cdot|$ to denote the scalar absolute value and $\|\cdot\|$ to denote the vector 2-norm. The set of natural numbers is defined as $\mathbb{N}$. For a matrix $A \in \mathbb{R}^{m \times n}$, we use $\|A\|_F$, $\sigma_{\text{min}}(A)$ and $A(i, :)$ to denote the matrix Frobenius norm, the smallest non-zero singular value of matrix $A$, and the $i$th row of the matrix $A$, respectively. Let $P_C$ represent the metric projection onto the set $C$. We indicate by $E_{k-1}[\cdot]$ the expected value conditional on the first $k - 1$ iterations, and from the law of the iterated expectation, we have $E[E_{k-1}[\cdot]] = E[\cdot]$.

2 The nonlinear sampling Kaczmarz–Motzkin method

Consider the nonlinear equations

$$f(x) = 0, \quad \text{with } f : \mathcal{D}(f) \subseteq \mathbb{R}^n \to \mathbb{R}^m,$$

which can also be written in the following form:

$$f_i(x) = 0, \quad i = 1, 2, \ldots, m,$$

where at least one $f_i : \mathcal{D}(f_i) \subseteq \mathbb{R}^n \to \mathbb{R}(i = 1, 2, \ldots, m)$ are nonlinear operators, and $x \in \mathbb{R}^n$ is an unknown vector.

In this section, we will present the NSKM method for solving the system (2) and explore its convergence analysis.

2.1 The NSKM method

In each iteration of the NRK method, we note that the NRK method needs to calculate $f(x)$, which is very expensive and inefficient when the size of nonlinear equations is large. Thus, we are eager to design a more cheap algorithm at each update to avoid computing all $f(x)$. Motivated by the SKM method in De Loera et al. (2017), we propose the NSKM method, in which a sample of $\beta$ constraints is randomly selected from all entries of $f(x)$ according to uniform probability and a portion of $f(x)$ need to be calculated instead of all $f(x)$. Moreover, in the NSKM method, we choose indicator $i_k$ by the maximal-residual criterion from the selected sample. The NSKM method can be formulated as follows.

2.2 Convergence analysis of the NSKM method

To prove the convergence of Algorithm 1, we need to prepare some definitions, lemmas, and corollaries at the top of this section.

**Definition 1** (Wang et al. 2022) A matrix $A \in \mathbb{R}^{m \times n}$ is called row bounded below, if there exists a positive number $\varepsilon$, such that $\|A(i, :)\| \geq \varepsilon$, for $1 \leq i \leq m$. 
Algorithm 1 The nonlinear sampling Kaczmarz–Motzkin (NSKM) method

**Input:** $f(x), \beta, x_0, k = 1$, and maximum iteration steps $T$

**Output:** $x_k$

1: while iteration termination criterion does not hold and $k \leq T$ do
2: Choose a sample of $\beta$ constraints, $\tau_k$, uniformly at random from all entries of $f(x)$
3: Compute residual of the selected sample $r_{\tau_k} = -f_{\tau_k}(x_{k-1})$
4: Set $i_k = \arg \max_{i \in \tau_k} |r_i|$
5: Compute gradient $g_{i_k} = \nabla f_{i_k}(x_{k-1})$
6: Set $x_k = x_{k-1} + \frac{r_{i_k}}{\|g_{i_k}\|^2} g_{i_k}$
7: $k = k + 1$
8: end while

**Definition 2** (Haltmeier et al. 2007) If there is a point $x_0 \in \mathcal{D}(f)$, such that for every $i \in \{1, 2, \ldots, m\}$ and $\forall x_1, x_2 \in \mathcal{B}_\rho(x_0) \subset \mathcal{D}(f)$ ($\mathcal{B}_\rho(x_0) = \{x | \|x - x_0\| \leq \rho\}$), there exists $\eta_i \in [0, \eta]$ ($\eta = \max \eta_i < \frac{1}{2}$), such that

$$|f_i(x_1) - f_i(x_2) - \nabla f_i(x_1)(x_1 - x_2)| \leq \eta_i |f_i(x_1) - f_i(x_2)|,$$

then the function $f: \mathcal{D}(f) \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$ is referred to satisfy the local tangential cone condition in a ball $\mathcal{B}_\rho(x_0)$ of radius $\rho$ around $x_0$.

**Lemma 1** (Wang et al. 2022) Let $f(x)$ satisfy the local tangential cone condition in a ball $\mathcal{B}_\rho(x_0)$. Then, for $\forall x_1, x_2 \in \mathcal{B}_\rho(x_0) \subset \mathcal{D}(f)$, we have

$$|f_i(x_1) - f_i(x_2)| \geq \frac{1}{1 + \eta_i} |\nabla f_i(x_1)(x_1 - x_2)|, \quad i \in \{1, 2, \ldots, m\}.$$

By Lemma 1, the following corollary is naturally followed.

**Corollary 1** Let $f(x)$ satisfy the local tangential cone condition in a ball $\mathcal{B}_\rho(x_0)$. Then, for $\forall x_1, x_2 \in \mathcal{B}_\rho(x_0) \subset \mathcal{D}(f)$, we have

$$\|f(x_1) - f(x_2)\|^2 \geq \frac{1}{(1 + \eta)^2} \|f'(x_1)(x_1 - x_2)\|^2,$$

where $\eta = \max \eta_i < \frac{1}{2}$ ($i = 1, 2, \ldots, m$).

**Proof**

$$\|f(x_1) - f(x_2)\|^2 = \|(f_1(x_1) - f_1(x_2), f_2(x_1) - f_2(x_2), \ldots, f_m(x_1) - f_m(x_2))\|^2$$

$$= \|(f_1(x_1) - f_1(x_2))^2 + (f_2(x_1) - f_2(x_2))^2 + \cdots + (f_m(x_1) - f_m(x_2))^2\)$$

$$\geq \frac{1}{(1 + \eta_1)^2} (\nabla f_1(x_1)(x_1 - x_2))^2 + \frac{1}{(1 + \eta_2)^2} (\nabla f_2(x_1)(x_1 - x_2))^2 + \cdots$$

$$+ \frac{1}{(1 + \eta_m)^2} (\nabla f_m(x_1)(x_1 - x_2))^2$$

$$\geq \frac{1}{(1 + \eta)^2} \sum_{i=1}^{m} (\nabla f_i(x_1)(x_1 - x_2))^2$$

$$= \frac{1}{(1 + \eta)^2} \|f'(x_1)(x_1 - x_2)\|^2,$$

where the first inequality is obtained by Lemma 1.
Lemma 2 Let \( f(x) \) satisfy the local tangential cone condition in a ball \( B_\rho(x_0) \) with \( x^* \in B_\rho(x_0) \). Then, the sequence \( \{x_k\}_{k=0}^\infty \) generated by Algorithm 1 is contained in \( B_\rho(x_0) \subset \mathcal{D}(f) \). Furthermore, we have

\[
\|x_{k+1} - x^*\|^2 \leq \|x_k - x^*\|^2 - (1 - 2\eta_{i,k+1}) \frac{f_{ik+1}^2(x_k)}{\|\nabla f_{ik+1}(x_k)\|^2},
\]

where \( f(x^*) = 0 \).

Proof See Appendix.

Lemma 3 (Brinkhuis 2020) A proper convex function is a function \( f : \mathcal{D}(f) \to \mathbb{R} \), where \( \mathcal{D}(f) \subset \mathbb{R}^n \) is a nonempty convex set. Then, Jensen’s inequality

\[
f((1 - \alpha)x + \alpha y) \leq (1 - \alpha)f(x) + \alpha f(y), \forall \alpha \in [0, 1], \forall x, y \in \mathcal{D}(f)
\]

holds.

The following Lemma 4 is very important in the next convergence analysis, so we review its proof in Appendix.

Lemma 4 (Brinkhuis 2020) Let a convex set \( \mathcal{D}(f) \subset \mathbb{R}^n \) and a differentiable function \( f : \mathcal{D}(f) \to \mathbb{R} \) be given. If \( f \) is convex, then \( f(x) - f(y) \geq f'(y)(x - y) \) for all \( x, y \in \mathcal{D}(f) \).

Lemma 5 (De Loera et al. 2017) Suppose \( \{a_i\}_{i=1}^n \) and \( \{b_i\}_{i=1}^n \) are real sequences and \( a_{i+1} > a_i > 0 \) and \( b_{i+1} \geq b_i \geq 0 \). There exists

\[
\sum_{i=1}^n a_i b_i \geq \sum_{i=1}^n \tilde{a}b_i,
\]

where \( \tilde{a} \) is the average \( \tilde{a} = \frac{1}{n} \sum_{i=1}^n a_i \).

Lemma 6 (Wang et al. 2022) Let \( a = \{a_1, a_2, \ldots, a_n\} \) and \( b = \{b_1, b_2, \ldots, b_n\} \) be two arrays with real components and satisfy \( a_j \geq 0, b_j > 0, \forall j \in \{1, 2, \ldots, n\} \), then the following inequality is established:

\[
\sum_{j=1}^n a_j \leq \frac{\sum_{j=1}^n a_j}{\sum_{j=1}^n b_j}.
\]

2.2.1 Convergence analysis I

Assumption 1 The following assumptions hold:

(i) Nonlinear function \( f : \mathcal{D}(f) \subseteq \mathbb{R}^n \to \mathbb{R}^m \) satisfies the local tangential cone condition in a ball \( B_\rho(x_0) \).

(ii) For \( \forall x \in \mathcal{D}(f) \), \( f'(x) \) is row bounded below and full column rank matrix.

Theorem 1 Assume that \( f(x) \) satisfies Assumption 1 and \( f(x) = 0 \) is solvable in \( B_\rho(x_0) \). Then, the iteration sequence \( \{x_k\}_{k=0}^\infty \) generated by the NSKM method converges to a solution \( x^* \in B_\rho(x_0) \) of \( f(x) \) in expectation. Moreover, the mean squared iteration error satisfies

\[
E\|x_k - x^*\|^2 \leq (1 - \frac{(1 - 2\eta)\alpha_{\min}}{m(1 + \eta)^2\|f'(x_{k-1})\|^2})E\|x_{k-1} - x^*\|^2, \quad k = 1, 2, \ldots,
\]

where \( \eta = \max_i \eta_i < \frac{1}{2} \) (\( i = 1, 2, \ldots, m \)).

Proof From Lemma 2, we have
\[ \| x_k - x^* \|^2 \leq \| x_{k-1} - x^* \|^2 - (1 - 2\eta_{ik}) \frac{f^2_{ik}(x_{k-1})}{\| \nabla f_{ik}(x_{k-1}) \|^2}. \]

By taking the conditional expectation on both sides of the above formula, we obtain
\[ E_{k-1} \| x_k - x^* \|^2 \leq \| x_{k-1} - x^* \|^2 - E_{k-1} (1 - 2\eta_{ik}) \frac{f^2_{ik}(x_{k-1})}{\| \nabla f_{ik}(x_{k-1}) \|^2}. \]

Since \( \eta_{ik} \in [0, \eta) \) (\( \eta = \max_i \eta_i < \frac{1}{2} \)), we have that
\[ E_{k-1} \| x_k - x^* \|^2 \leq \| x_{k-1} - x^* \|^2 - (1 - 2\eta) E_{k-1} \frac{f^2_{ik}(x_{k-1})}{\| \nabla f_{ik}(x_{k-1}) \|^2}. \]

Next, we consider \( E_{k-1} \frac{f^2_{ik}(x_{k-1})}{\| \nabla f_{ik}(x_{k-1}) \|^2} \). Let \( f^2_j(x_{k-1}) \) denote the \((j + \beta)th\) smallest entry of \( \{ f^2_j(x_{k-1}) \}_{i=1}^m \) [i.e., if we order all entries of \( \{ f^2_j(x_{k-1}) \}_{i=1}^m \) from smallest to largest, \( f^2_j(x_{k-1}) \) is in the \((j + \beta)th\) position]. Each sample has equal probability of being selected, \((m)\)^{-1}. However, the selected frequency of each entry of \( \{ f^2_j(x_{k-1}) \}_{i=1}^m \) depends on its size. For example, the largest entry of \( \{ f^2_j(x_{k-1}) \}_{i=1}^m \) can be selected from all samples in which it appears, while the \( \beta \)th smallest entry of \( \{ f^2_j(x_{k-1}) \}_{i=1}^m \) can be selected from only one sample. Therefore, we have that
\[ E_{k-1} \frac{f^2_{ik}(x_{k-1})}{\| \nabla f_{ik}(x_{k-1}) \|^2} = \sum_{j=0}^{m-\beta} \frac{(j + \beta - 1)}{(m) \beta} f^2_j(x_{k-1}) \frac{f^2_{ik}(x_{k-1})}{\| \nabla f_{j}(x_{k-1}) \|^2} \]
\[ = \frac{1}{(m) \beta} \sum_{j=0}^{m-\beta} (j + \beta - 1) \frac{f^2_j(x_{k-1})}{\| \nabla f_{j}(x_{k-1}) \|^2}. \]

where the first inequality comes from Lemma 6, the second inequality is from Lemma 5, because \((\frac{(j + \beta - 1)}{(m) \beta})_{j=0}^{m-\beta}\) is strictly increasing and \( f^2_j(x_{k-1}) \) is non-decreasing, and the last equality follows from the fact that \( \sum_{l=0}^{m-\beta} \frac{(l + \beta - 1)}{(m) \beta} = \left(\frac{m}{\beta}\right) \), which is known as the column-sum property of Pascal’s triangle.

Let \( s_{k-1} \) be the number of zero entries in \( f(x_{k-1}) \) and \( V_{k-1} = \max \{ m - s_{k-1}, m - \beta + 1 \} \), then from (7), we can derive
\[ E_{k-1} \frac{f^2_{ik}(x_{k-1})}{\| \nabla f_{ik}(x_{k-1}) \|^2} \geq \frac{1}{m - \beta + 1} \min \left\{ \frac{m - \beta + 1}{m - s_{k-1}}, 1 \right\} \frac{\| f(x_{k-1}) \|^2}{\| f'(x_{k-1}) \|^2}. \]
In accordance with formula (8), the formula (5) then further results in the estimate

$$E_{k-1} \|x_k - x^*\|^2$$

$$\leq \|x_{k-1} - x^*\|^2 - (1 - 2\eta)E_{k-1} \frac{f_k^2(x_{k-1})}{\|\nabla f_k(x_{k-1})\|^2}$$

$$\leq \|x_{k-1} - x^*\|^2 - (1 - 2\eta) \frac{1}{V_{k-1}} \frac{\|f(x_{k-1})\|^2}{\|f'(x_{k-1})\|^2}$$

$$\leq \|x_{k-1} - x^*\|^2 - \frac{1 - 2\eta}{m} \frac{\|f(x_{k-1})\|^2}{\|f'(x_{k-1})\|^2}$$

$$= \|x_{k-1} - x^*\|^2 - \frac{1 - 2\eta}{m} \frac{\|f(x_{k-1}) - f(x^*)\|^2}{\|f'(x_{k-1})\|^2}$$

$$\leq \|x_{k-1} - x^*\|^2 - \frac{1 - 2\eta}{m} \frac{\|f(x_{k-1}) - f(x^*)\|^2}{\|f'(x_{k-1})\|^2 \|x_{k-1} - x^*\|^2}$$

$$= \left(1 - \frac{(1 - 2\eta)\sigma_{\min}^2(f'(x_{k-1}))}{m(1 + \eta)^2 \|f'(x_{k-1})\|^2} \right) \|x_{k-1} - x^*\|^2. \tag{9}$$

where the third inequality follows from the fact that $V_{k-1} \leq m$, the fourth inequality comes from Corollary 1, and the last inequality is from the fact that $\|f'(x_{k-1})(x_{k-1} - x^*)\|^2 \geq \sigma_{\min}^2(f'(x_{k-1})) \|x_{k-1} - x^*\|^2$ when $f'(x_{k-1})$ is full column rank.

By taking full expectation on the both sides, we can further obtain

$$E \|x_k - x^*\|^2 \leq \left(1 - \frac{(1 - 2\eta)\sigma_{\min}^2(f'(x_{k-1}))}{m(1 + \eta)^2 \|f'(x_{k-1})\|^2} \right) E \|x_{k-1} - x^*\|^2.$$

Since $\sigma_{\min}^2(f'(x_{k-1})) < \|f'(x_{k-1})\|^2$, and $0 < 1 - 2\eta < (1 + \eta)^2$, we have

$$0 < 1 - \frac{(1 - 2\eta)\sigma_{\min}^2(f'(x_{k-1}))}{m(1 + \eta)^2 \|f'(x_{k-1})\|^2} < 1.$$

Therefore, the iteration sequence $\{x_k\}_{k=0}^\infty$ generated by the NSKM method converges to $x^*$ in expectation.

**Remark 1**

1. To ensure that $\|\nabla f_{k+1}(x_k)\| \neq 0$ in $x_{k+1} = x_k - \frac{f_{k+1}(x_k)}{\|\nabla f_{k+1}(x_k)\|^2} \nabla f_{k+1}(x_k)^T$ ($k = 0, 1, 2, \ldots$), we require that $f'(x)$ is row bounded below for every $x \in \mathcal{D}(f)$.

2. To get the desired convergence rate of the NSKM method, we need $m \geq n$ and $f'(x)$ is full column rank for every $x \in \mathcal{D}(f)$. From the third inequality in (9), we can obtain that the NSKM method also converges to a solution of $f(x) = 0$ when $m < n$.

**2.2.2 Convergence analysis II**

**Assumption 2** The following assumptions hold:
(i) $f_i : \mathcal{D}(f) \subseteq \mathbb{R}^n \to \mathbb{R}$ ($i = 1, 2, \ldots, m$) on the nonempty convex set $\mathcal{D}(f)$ are convex functions.

(ii) For $\forall x \in \mathcal{D}(f)$, $f'(x)$ is row bounded below.

(iii) $f_i(x) \geq 0$ ($i = 1, 2, \ldots, m$) for $\forall x \in \mathcal{D}(f)$.

Theorem 2 Suppose that Assumption 2 holds and there exists $x^*$ such that $f(x^*) = 0$. Then, the iteration sequence $\{x_k\}_{k=0}^\infty$ generated by the NSKM method converges to a solution $x^*$ of $f(x)$. Moreover, iteration error satisfies

$$\|x_k - x^*\|^2 \leq \|x_{k-1} - x^*\|^2 - \frac{f_i^2(x_{k-1})}{\|\nabla f_i(x_{k-1})\|^2}, k = 1, 2, \ldots.$$  

Proof

\[
\|x_k - x^*\|^2 - \|x_{k-1} - x^*\|^2 \\
= \frac{f_i^2(x_{k-1})}{\|\nabla f_i(x_{k-1})\|^2} - 2 \frac{f_i(x_{k-1})}{\|\nabla f_i(x_{k-1})\|^2} \nabla f_i(x_{k-1})(x_{k-1} - x^*) \\
= \frac{f_i^2(x_{k-1})}{\|\nabla f_i(x_{k-1})\|^2} + 2 \frac{f_i(x_{k-1})}{\|\nabla f_i(x_{k-1})\|^2} (f_i(x_{k-1}) + \nabla f_i(x_{k-1})(x^* - x_{k-1})) \\
- 2 \frac{f_i(x_{k-1})}{\|\nabla f_i(x_{k-1})\|^2} f_i(x_{k-1}) \\
\leq \frac{f_i^2(x_{k-1})}{\|\nabla f_i(x_{k-1})\|^2} - 2 \frac{f_i^2(x_{k-1})}{\|\nabla f_i(x_{k-1})\|^2} \\
= - \frac{f_i^2(x_{k-1})}{\|\nabla f_i(x_{k-1})\|^2},
\]

where the first equality is obtained by the proof of Lemma 2 and the first inequality comes from Lemma 4 and (iii) of Assumption 2. \qed

Remark 2 1. When $f_i(x)$ is a convex function or satisfies (3) in a ball $\mathcal{B}_p(x_0)$ for $\forall i \in \{1, 2, \ldots, m\}$, by Theorems 1 and 2, we can get that the iteration sequence generated by the NSKM method also converges to a solution of (2). The result will be stated in the following Corollary 2.

2. When $f(x)$ satisfies Assumption 2 or the conditions of Corollary 2, the convergences of the NRK method and the NURK method can also be guaranteed.

Corollary 2 Suppose that $f_i(x) : \mathcal{D}(f_i) \to \mathbb{R}$ satisfies (3) in a ball $\mathcal{B}_p(x_0)$ or is a nonnegative convex function. Let $f'(x)$ is row bounded below for $\forall x \in \mathcal{D}(f)$ and $f(x) = 0$ is solvable in $\mathcal{B}_2(x_0)$. Then, the iteration sequence $\{x_k\}_{k=0}^\infty$ generated by the NSKM method converges to a solution $x^* \in \mathcal{B}_2(x_0)$ of $f(x)$.

3 The variants of the NSKM method

In this section, we will provide two variants of the NSKM method that converge to a solution of (1). We note that a projection onto the closed convex sets has one famous nonexpansive property, inspired by which we present the PSKM method. Furthermore, in Qin and Etesami (2020), Qin and Etesami presented a randomized accelerated projected algorithm which had...
the faster convergence rate compared with the alternating projected method. Thus, we devise the APSKM method.

### 3.1 The PSKM method

The PSKM method can be divided into two computational stages. The first stage uses the one-step NSKM update for the nonlinear equation \( f(x) = 0 \) and gets \( x_{k-1}/2 \), and the second stage outputs \( x_k \) by utilizing one-step randomized projection onto the finite nonempty closed convex sets \( C_i \) \( (i = 1, 2, \ldots, k_c) \). The pseudo-code of the PSKM method is given in the following Algorithm 2.

**Algorithm 2** The projected sampling Kaczmarz–Motzkin (PSKM) method

**Input:** \( f(x), \beta, x_0, k = 1 \), number of the nonempty closed convex sets \( k_c \) and maximum iteration steps \( T \)

**Output:** \( x_k \)

1. while iteration termination criterion does not hold and \( k \leq T \) do
2. Choose a sample of \( \beta \) constraints, \( \tau_k \), uniformly at random from all entries of \( f(x) \)
3. Compute residual of the selected sample \( r_{\tau_k} = -f_{\tau_k}(x_{k-1}) \)
4. Set \( i_k = \text{argmax } i \in \tau_k | r_i | \)
5. Compute gradient \( g_{i_k} = \nabla f_{i_k}(x_{k-1}) \)
6. Set \( x_k - 1/2 = x_k - 1 + r_{i_k} \parallel g_{i_k} \parallel^2 g_{i_k}^T \)
7. Choose an indicator, \( a_k \), uniformly at random from the set \{1, 2, \ldots, k_c\}
8. \( x_k = P_{C_{a_k}}(x_{k-1}/2) \)
9. \( k = k + 1 \)
10. end while

#### 3.1.1 Convergence analysis of the PSKM method

For proving the convergence of the PSKM method, we will first describe a crucial lemma.

**Lemma 7** (Qin and Etesami 2020) Given a nonempty closed convex set \( X \subset \mathbb{R}^d \) and a point \( y \in \mathbb{R}^d \), the following relations hold for the projection \( P_X(y) \) of the point \( y \) on \( X \) and for all \( x \in X \):

\[
\langle P_X(y) - y, x - P_X(y) \rangle \geq 0, \\
\|x - P_X(y)\|^2 \leq \|x - y\|^2 - \|y - P_X(y)\|^2, \\
\]

with equality in the second relation if and only if \( x = P_X(y) \) or \( y \in X \).

**Theorem 3** Suppose that \( f(x) \) satisfies Assumption 1 and (1) is solvable in \( \mathcal{B}_2(x_0) \). Then, the iteration sequence \( \{x_k\}_{k=0}^\infty \) generated by the PSKM method converges to a solution \( x^* \in \mathcal{B}_2(x_0) \) of (1) in expectation. Moreover, the mean squared iteration error satisfies

\[
E\|x_k - x^*\|^2 \leq \left( 1 - \frac{(1 - 2\eta)\sigma_{\min}^2(f'(x_{k-1})))}{m(1 + \eta)^2 \|f'(x_{k-1})\|^2} \right) E\|x_{k-1} - x^*\|^2, k = 1, 2, \ldots, \\
\]

where \( \eta = \max_i \eta_i < \frac{1}{2} \) \( (i = 1, 2, \ldots, m) \).

**Proof** By Lemma 7, we have that
\[ \| x_k - x^* \|^2 = \| P_{C_{a_k}}(x_{k-1}) - P_{C_{a_k}}(x^*) \|^2 \leq \| x_{k-\frac{1}{2}} - x^* \|^2. \]

Since \( x_{k-\frac{1}{2}} \) is generated by the NSKM method, from Theorem 1, we can get
\[
E\| x_{k-\frac{1}{2}} - x^* \|^2 \leq \left( 1 - \frac{(1 - 2\eta)\sigma^2_{\min}(f'(x_{k-1}))}{m(1 + \eta)^2\| f'(x_{k-1}) \|_{\mathcal{F}}^2} \right) E\| x_{k-1} - x^* \|^2.
\]

Thus
\[
E\| x_k - x^* \|^2 \leq \| x_{k-\frac{1}{2}} - x^* \|^2 \leq \left( 1 - \frac{(1 - 2\eta)\sigma^2_{\min}(f'(x_{k-1}))}{m(1 + \eta)^2\| f'(x_{k-1}) \|_{\mathcal{F}}^2} \right) E\| x_{k-1} - x^* \|^2.
\]

By deducing in Theorem 1, we have
\[
0 < 1 - \frac{(1 - 2\eta)\sigma^2_{\min}(f'(x_{k-1}))}{m(1 + \eta)^2\| f'(x_{k-1}) \|_{\mathcal{F}}^2} < 1.
\]

Therefore, the iteration sequence generated by the PSKM method converges to a solution \( x^* \in \mathcal{B}_\rho(x_0) \) of (1) in expectation. \( \Box \)

Similar to the proof of Theorem 3, we can get the following two theorems.

**Theorem 4** Suppose that \( f(x) \) satisfies Assumption 2 and there exists \( x^* \), such that \( f(x^*) = 0 \), \( x^* \in C \). Then, the iteration sequence \( \{x_k\}_{k=0}^{\infty} \) generated by the PSKM method converges to a solution \( x^* \) of (1). Moreover, iteration error satisfies
\[
\| x_k - x^* \|^2 \leq \| x_{k-1} - x^* \|^2 - \frac{f_{ik}^2(x_{k-1})}{\| \nabla f_{ik}(x_{k-1}) \|^2}, k = 1, 2, \ldots
\]

**Theorem 5** Assume that \( f_i(x) : \mathcal{D}(f_i) \to \mathbb{R} \) satisfies (3) in a ball \( \mathcal{B}_\rho(x_0) \) or is a nonnegative convex function. Let \( f'(x) \) is row bounded below for \( \forall x \in \mathcal{D}(f) \) and (1) is solvable in \( \mathcal{B}_{\rho}(x_0) \). Then, the iteration sequence \( \{x_k\}_{k=0}^{\infty} \) generated by the PSKM method converges to a solution \( x^* \in \mathcal{B}_{\rho}(x_0) \) of (1).

### 3.2 The APSKM method

When the correlation of the rows of coefficient matrix \( A \) is high in equality constraints \( C = \{x : \langle A, x \rangle = b\} \), the alternating projection makes little progress in each iteration, which might cause a slow speed of convergence in the PSKM method. To solve the problem, we note that Qin and Etesami (2020) proposed the randomized accelerated projection method (Algorithm 3), which arrives at a feasible point in the intersection of selected sets in each iteration. Motivated by this, we propose the APSKM method.

Before we give the APSKM method, we first provide a lemma.

**Lemma 8** Let \( x^* \) be a feasible point in the intersection of all closed convex sets \( \{C_i\}_{i=1}^{k} \). In Algorithm 3, assume that \( x_{k+\frac{1}{2}} \neq x_{k+\frac{1}{2}} \), then we have \( x_{k+\frac{1}{2}} \neq x_{k+\frac{1}{2}} \).

**Proof** See Appendix. \( \square \)
Algorithm 3 The randomized accelerated projection method

**Input:** $x_0$, the closed convex sets $\{C_i\}_{i=1}^C$, a sequence of uniform random variables $\{\alpha(k)\}_{k=0}^\infty$ over the set $\{1, 2, \ldots, k_c\}$ and the maximum iteration step $T$

**Output:** $x_k$

1: $k = 0$

2: while $k \leq T$ do

3: $x_{k+\frac{1}{4}} = PC_{\alpha(k)}(x_k)$

4: $x_{k+\frac{2}{4}} = PC_{\alpha(k+1)}(x_{k+\frac{1}{4}})$

5: $x_{k+\frac{3}{4}} = PC_{\alpha(k)}(x_{k+\frac{2}{4}})$

6: Calculate $\lambda_k = \frac{\|x_{k+\frac{1}{4}} - x_{k+\frac{2}{4}}\|^2}{\langle x_{k+\frac{1}{4}} - x_{k+\frac{3}{4}}, x_{k+\frac{1}{4}} - x_{k+\frac{2}{4}} \rangle}$

7: $x_{k+1} = x_{k+\frac{1}{4}} + \lambda_k(x_{k+\frac{3}{4}} - x_{k+\frac{1}{4}})$

8: $k \leftarrow k + 1$

end while

Algorithm 4 The accelerated projected sampling Kaczmarz–Motzkin (APSKM) method

**Input:** $f(x)$, $\beta$, $x_0$, $k = 1$, $\delta$, number of the closed convex sets $k_c$ and maximum iteration steps $T$

**Output:** $x_k$

1: while iteration termination criterion does not hold and $k \leq T$ do

2: Choose a sample of $\beta$ constraints, $\tau_k$, uniformly at random from all entries of $f(x)$

3: Compute residual of the selected sample $r_{\tau_k} = -f_{\tau_k}(x_k - 1)$

4: Set $i_k = \text{argmax}_{i \in \tau_k} |r_i|$

5: Compute gradient $g_{i_k} = \nabla f_{i_k}(x_k - 1)$

6: Set $x_{k+\frac{1}{5}} = x_k - 1 + \frac{|r_{i_k}|}{\|g_{i_k}\|} g_{i_k}$

7: Choose two indicators, $\alpha_{i_1}^k$, $\alpha_{i_2}^k$, uniformly at random from the set $\{1, 2, \ldots, k_c\}$

8: $x_{k-\frac{2}{5}} = PC_{\alpha_{i_1}^k}(x_{k-\frac{1}{5}})$

9: $x_{k-\frac{3}{5}} = PC_{\alpha_{i_2}^k}(x_{k-\frac{2}{5}})$

10: if $\|x_{k-\frac{2}{5}} - x_{k-\frac{3}{5}}\|_\infty < \delta$ then

11: $x_k = x_{k-\frac{3}{5}}$

else

12: $x_{k-\frac{1}{5}} = PC_{\alpha_{i_1}^k}(x_{k-\frac{2}{5}})$

13: Calculate $\lambda_k = \frac{\|x_{k-\frac{1}{5}} - x_{k-\frac{2}{5}}\|^2}{\langle x_{k-\frac{1}{5}} - x_{k-\frac{3}{5}}, x_{k-\frac{1}{5}} - x_{k-\frac{2}{5}} \rangle}$

14: $x_k = x_{k-\frac{3}{5}} + \lambda_k(x_{k-\frac{1}{5}} - x_{k-\frac{3}{5}})$

15: end if

16: $k \leftarrow k + 1$

end while

Next, we give the APSKM method in Algorithm 4. The APSKM method also contains two computational stages. The first stage is similar to that of the PSKM method. However, the second stage is vastly different, which is divided into two cases to get $x_k$.

**Remark 3** When $\|x_{k-\frac{2}{5}} - x_{k-\frac{1}{5}}\|_\infty \geq \delta$, we have $x_{k-\frac{2}{5}} \neq x_{k-\frac{3}{5}}$ and $x_{k-\frac{1}{5}} \neq x_{k-\frac{3}{5}}$ (Lemma 8) which guarantee that $\lambda_k$ can be calculated well.
3.2.1 Convergence analysis of the APSKM method

We will provide a lemma to prove the convergence of the APSKM method.

Lemma 9 (Qin and Etesami 2020) Given two closed convex sets $C_{\alpha_1}$ and $C_{\alpha_2}$, with $C_{\alpha_1} \cap C_{\alpha_2} \neq \emptyset$. The parameter $\lambda_k$ and the points $x_{k-\frac{3}{5}}, x_{k-\frac{2}{5}}, x_{k-\frac{1}{5}}, x_k$ are generated according to Step 14, Step 8, Step 9, Step 13, Step 16 of Algorithm 4, respectively. Then, we have the following property:

For $\forall x \in C_{\alpha_1} \cap C_{\alpha_2}$

$$\|x_k - x\|^2 \leq \|x_{k-\frac{4}{5}} - x\|^2. \quad (10)$$

Proof See Appendix.

Theorem 6 Suppose that $f(x)$ satisfies Assumption 1 and (1) is solvable in $B_{\rho_2}(x_0)$. Then, the iteration sequence $\{x_k\}_{k=0}^{\infty}$ generated by the APSKM method converges to a solution $x^* \in B_{\rho_2}(x_0)$ of (1) in expectation. Moreover, the mean squared iteration error satisfies

$$E\|x_k - x^*\|^2 \leq \left(1 - \frac{(1 - 2\eta)\sigma^2_{\min}(f'(x_{k-1}))}{m(1 + \eta)^2\|f'(x_{k-1})\|^2_F}\right) E\|x_{k-1} - x^*\|^2, \quad k = 1, 2, \ldots,$$

where $\eta = \max_i \eta_i < \frac{1}{2}$ ($i = 1, 2, \ldots, m$).

Proof If $\|x_{k-\frac{4}{5}} - x_{k-\frac{3}{5}}\|_{\infty} < \delta$, by Lemma 7, we have

$$\|x_k - x^*\|^2 \leq \|x_{k-\frac{4}{5}} - x^*\|^2.$$

If $x_k$ is obtained by Step 16 of Algorithm 4, from Lemma 9, we obtain that

$$\|x_k - x^*\|^2 \leq \|x_{k-\frac{4}{5}} - x^*\|^2.$$

Since $x_{k-\frac{4}{5}}$ is obtained by one-step update of the NSKM method, from Theorem 1, we can get

$$E\|x_{k-\frac{4}{5}} - x^*\|^2 \leq \left(1 - \frac{(1 - 2\eta)\sigma^2_{\min}(f'(x_{k-1}))}{m(1 + \eta)^2\|f'(x_{k-1})\|^2_F}\right) E\|x_{k-1} - x^*\|^2.$$

Thus

$$E\|x_k - x^*\|^2 \leq E\|x_{k-\frac{4}{5}} - x^*\|^2 \leq \left(1 - \frac{(1 - 2\eta)\sigma^2_{\min}(f'(x_{k-1}))}{m(1 + \eta)^2\|f'(x_{k-1})\|^2_F}\right) E\|x_{k-1} - x^*\|^2.$$

By the derivation of Theorem 1, we have that

$$0 < 1 - \frac{(1 - 2\eta)\sigma^2_{\min}(f'(x_{k-1}))}{m(1 + \eta)^2\|f'(x_{k-1})\|^2_F} < 1.$$

Thus, the iteration sequence $\{x_k\}_{k=0}^{\infty}$ generated by the APSKM method converges to a solution $x^* \in B_{\rho_2}(x_0)$ of (1) in expectation. \[\square\]

Similar to the proof of Theorem 6, we can get the following two theorems.
Theorem 7 Suppose that \( f(x) \) satisfies Assumption 2 and there exists \( x^* \), such that \( f(x^*) = 0 \), \( x^* \in C \). Then, the iteration sequence \( \{x_k\}_{k=0}^{\infty} \) generated by the APSKM method converges to a solution \( x^* \) of (1). Moreover, iteration error satisfies
\[
\|x_k - x^*\|^2 \leq \|x_{k-1} - x^*\|^2 - \frac{f^2_k(x_{k-1})}{\|\nabla f_j(x_{k-1})\|^2}, \quad k = 1, 2, \ldots
\]

Theorem 8 Assume that \( f_i(x) : \mathcal{D}(f_i) \to \mathbb{R} \) satisfies (3) in a ball \( \mathcal{B}_\rho(x_0) \) or is a nonnegative convex function. Let \( f' \) be row bounded below for \( \forall x \in \mathcal{D}(f) \) and (1) is solvable in \( \mathcal{B}_\rho(x_0) \). Then, the iteration sequence \( \{x_k\}_{k=0}^{\infty} \) generated by the APSKM method converges to a solution \( x^* \in \mathcal{B}_\rho(x_0) \) of (1).

4 Numerical experiments

In this section, first, on some given large-scale nonlinear equations, we run the NSKM method while varying the sample size, \( \beta \), and compare the NSKM method with the NRK method. Second, we investigate the performances of the PSKM and APSKM methods and compare them with the PSGD method by testing some large-scale constrained nonlinear equations. In our implementations, the stopping criterion is
\[
\text{RSE} = \frac{\|x_k - x^*\|^2}{\|x^*\|^2} \leq \varepsilon,
\]

or the maximum iteration steps 500,000 being reached. If the number of iteration steps exceeds 500,000, it is denoted as "-". IT and CPU denote the number of iteration steps and CPU times (in seconds), respectively. IT and CPU are the medians of the required iteration steps and the elapsed CPU times with respect to ten times repeated runs of the corresponding methods. All experiments are carried out using MATLAB (version R2020b) on a laptop with 2.20-GHZ intel Core i7-10870 H processor, 16 GB memory, and Windows 10 operating system.

4.1 Experiments on some large-scale nonlinear equations

In this subsection, we test the NSKM method on nonlinear equations
\[
f_i(x) = 0, \quad i = 1, 2, \ldots, m,
\]
where at least one \( f_i : \mathcal{D}(f_i) \subseteq \mathbb{R}^n \to \mathbb{R} \) are nonlinear operators and \( x \in \mathbb{R}^n \) is an unknown vector. We consider three cases: (i) \( m = n \); (ii) \( m > n \); (iii) \( m < n \).

Problem 1 The nonlinear equation is taken as
\[
f_i(x) = (e^{x_i-1} - 1)^2, \quad i = 1, 2, \ldots, m.
\]

Obviously, the solution of the problem is \( x^* = \text{ones}(m, 1) \) and the problem satisfies (i), (iii) of Assumption 2. For (ii), it is possible that there exists \( 1 \leq j \leq m \), such that \( \|\nabla f_j(x_k)\|^2 = 0 \), but if the iteration does not terminate, from the construction of Algorithm 1, we can know that the probability of \( \|\nabla f_j(x_k)\| = 0 \) is very small when \( \beta \) has the right size. Similarly, the NRK method can also be used to solve the problem. However, the nonlinear uniformly randomized (NURK) method (Wang et al. 2022) does not guarantee to solve the problem successfully. In our experiments, we set \( m = 5000 \), the initial value \( x_0 = 0.5 * \text{ones}(m, 1) \).
Fig. 1 Left: CPU versus the upper bound of the RSE on Problem 1. Middle: CPU versus the upper bound of the RSE on Problem 2. Right: CPU versus \( \|f(x)\|_2 \) on Problem 3

**Problem 2** The problem can be viewed as a modification of Chained Powell singular function in Lukšan et al. (2018)

\[
f_k(x) = x_i + 10x_{i+1} - 11, \quad \text{mod}(k, 4) = 1, \\
f_k(x) = \sqrt{5}(x_{i+2} - x_{i+3}), \quad \text{mod}(k, 4) = 2, \\
f_k(x) = (x_{i+1} - 2x_{i+2} + 1)^2, \quad \text{mod}(k, 3) = 3, \\
f_k(x) = \sqrt{10}(x_i - x_{i+3})^2, \quad \text{mod}(k, 4) = 0, \\
m = 2(n - 2), \quad i = 2\text{div}(k + 3, 4) - 1,
\]

where \( \text{div}(\cdot) \) is the division operation. The function satisfies conditions of Corollary 2 and (ii) of Assumption 2, as stated in Problem 1. Thus, the NSKM method and the NRK method can be utilized to solve the problem. The problem has a solution \( x^* = \text{ones}(n, 1) \). In our work, we set \( n = 5000, m = 2(n - 2) = 9996 \), and the initial value \( x_0 = 0.5 \times \text{ones}(n, 1) \).

**Problem 3** Problem NONDQUAR (Lukšan et al. 2018)

\[
f_k(x) = x_k - 1, \quad k = 1, \\
f_k(x) = (x_k + x_{k+1} + x_n)^2, \quad 1 < k < n, \\
f_k(x) = x_{k-1} - x_k, \quad k = n.
\]

We easily know that the function also satisfies conditions of Corollary 2 and (ii) of Assumption 2, as stated in Problem 1. Thus, the problem can be solved by the NSKM and NRK methods. To make \( m < n \), we set \( n = 20000 \) and select the equations whose indicators are odd. Under this setting, \( m = 10000 \) and the problem has an infinite number of solutions. Let the initial point \( x_0 = 0.5 \times \text{ones}(n, 1) \).

The numerical results for Problems 1, 2 and 3 are shown in Fig. 1. Note that when \( \beta = 1 \) the NSKM method is the NURK method. From Fig. 1, we observe that the NSKM method can work well for all cases. Moreover, we can find that the curves of the NSKM method with \( \beta = 50,100 \) are decreasing much more quickly than that of the other methods with respect to the increase of CPU times, which illustrates that the NSKM method with the sample of the right size performs better than the NRK and NURK methods in terms of CPU times.
4.2 Experiments on some large-scale nonlinear equations with finite convex constraints

To examine the convergence of the PSKM and APSKM methods, we use Problems 1, 2 and 3 with \( C = \{ x : Ax = b \} \) or \( C = \{ x : Ax \leq b \} \). The projector \( P_{C_i} \) onto the nonempty closed set \( C_i \) can be defined as follows:

1. When \( C = \{ x : Ax = b \} \), \( P_{C_i} x = x + \frac{b_i - \langle a_i, x \rangle}{\|a_i\|^2} a_i^T \).
2. When \( C = \{ x : Ax \leq b \} \), in Bauschke and Borwein (1996), \( P_{C_i} x = x - \frac{\langle a_i, x \rangle - b_i}{\|a_i\|^2} a_i^T \), where \( a_i \) and \( b_i \) are the \( i \)th row of the matrix \( A \) and the \( i \)th entry of \( b \), respectively.

\( \langle a_i, x \rangle - b_i \) is the maximum of \( \langle a_i, x \rangle - b_i \) and 0.

Thus, the constrained nonlinear equations are determined.

In all simulations, we choose \( \beta = 50 \) and \( \delta = 10^{-10} \). In the PSGD algorithm, we select a fixed step size, which is the best experimental result by trial and error. To construct convex constraints, we set \( b = Ax^* \) for equality constraints and \( b = Ax^* + \text{abs}(\delta 1) \) for inequality constraints where \( x^* \) is a solution of problem and \( \delta 1 \) is generated by the MATLAB function \textsc{randn}. In the first simulation, \( A \) is generated from the Gaussian distribution or the SuiteSparse Matrix Collection (Davis and Hu 2011). All real-world matrices and their properties are depicted in Table 1. The experimental results are shown in Tables 2, 3, 4 and Fig. 2, from which, we can clearly observe that the PSKM method requires less computing time than the PSGD and APSKM methods. In the second simulation, we consider \( C = \{ x : \langle A, x \rangle = b \} \) with coefficient matrix \( A \in \mathbb{R}^{k \times n} \) on \([\xi,1]\), which is generated from the MATLAB function \textsc{rand}. Note that the closer \( \xi \) tends to 1, the stronger the correlation of \( A \) is. The results are displayed in Figs. 3, 4, Tables 5 and 6. These results show that the APSKM method has a significantly better performance than other methods. Moreover, iteration counts and CPU times are decreasing with respect to the increase of \( \xi \), which reveals that the stronger the correlation of \( A \) is, the better performance the APSKM method has.

5 Remarks about the optimal choice of \( \beta \)

As shown in the experiment, \( \beta \) plays an important role in guaranteeing fast convergence. From (7), we know that the choice \( \beta = m \) yields the fastest convergence rate. However, this choice is also very expensive in terms of computation cost. Therefore, we need to find a balance between the computation cost and the convergence rate. For solving linear inequalities, De Loera et al. (2017) discussed the optimal selection of samples in the sampling Kaczmarz–Motzkin method. Inspired by their work, we build the following model to evaluate the convergence rate in terms of the computation cost.

We consider a fixed iteration \( j_1 \) and suppose that \( s \) equations are satisfied in this iteration. As seen from (6) in the proof of Theorem 1, the expected improvement \( \|x_{j_1} - x^*\|^2 - \)}
Table 2 IT and CPU of the PSGD, PSKM, and APSKM methods for Problem 1 with $C = \{x : \langle A, x \rangle \leq b\}$, $\varepsilon = 10^{-3}$ and $x_0 = 0.5 \ast \text{ones}(n, 1)$

| $m$  | 3000 | 5000 | 7000 | 9000 |
|------|------|------|------|------|
| $k_c = 300$ | PSGD IT | 101,567 | 98,220 | 125,724 | 220,820 |
|      | CPU  | 5.2485 | 6.6294 | 10.9787 | 25.8523 |
|      | PSKM IT | 8832 | 15,438 | 22,055 | 28,512 |
|      | CPU  | 0.5189 | 1.2335 | 2.1537 | 3.6497 |
|      | APSKM IT | 8855 | 15,459 | 22,024 | 28,471 |
|      | CPU  | 0.9579 | 2.1683 | 4.1890 | 6.5978 |

$k_c = 500$

| $k_c = 1000$ | PSGD IT | 67,795 | 195,818 | 162,349 | 198,686 |
| CPU  | 4.6462 | 16.4433 | 18.5343 | 29.8744 |
|      | PSKM IT | 8765 | 15,042 | 21,599 | 28,144 |
|      | CPU  | 0.6111 | 1.5549 | 2.8039 | 4.6362 |
|      | APSKM IT | 8882 | 15,092 | 21,604 | 28,144 |
|      | CPU  | 1.2722 | 2.5799 | 4.5439 | 8.7853 |

$k_c = 1000$

| $m$  | 3000 | 5000 | 7000 | 9000 |
|------|------|------|------|------|
| $k_c = 300$ | PSGD IT | 37,830 | 71,127 | 96,602 | 130,768 |
|      | CPU  | 0.8371 | 2.9615 | 4.6478 | 7.5615 |
|      | PSKM IT | 4545 | 7829 | 11,179 | 14,186 |
|      | CPU  | 0.1401 | 0.3957 | 0.6233 | 1.0056 |
|      | APSKM IT | 4580 | 7971 | 11,343 | 14,279 |
|      | CPU  | 0.2518 | 0.8005 | 1.3630 | 2.1831 |

$k_c = 500$

| $k_c = 1000$ | PSGD IT | 33,483 | 65,978 | 90,915 | 119,675 |
| CPU  | 0.9642 | 3.4890 | 5.1170 | 8.8733 |
|      | PSKM IT | 4066 | 7277 | 11,099 | 13,969 |
|      | CPU  | 0.1459 | 0.4430 | 0.7782 | 1.2064 |
|      | APSKM IT | 4096 | 7435 | 11,151 | 14,089 |
|      | CPU  | 0.2586 | 1.0215 | 1.8487 | 2.6214 |

Bold values indicates method which outperforms other methods

Table 3 IT and CPU of the PSGD, PSKM, and APSKM methods for Problem 2 with $C = \{x : \langle A, x \rangle = b\}$, $\varepsilon = 10^{-3}$ and $x_0 = 0.5 \ast \text{ones}(n, 1)$

| $m$  | 3000 | 5000 | 7000 | 9000 |
|------|------|------|------|------|
| $k_c = 300$ | PSGD IT | 19,537 | 52,287 | 80,825 | 90,823 |
|      | CPU  | 0.6997 | 3.5252 | 4.7413 | 6.3426 |
|      | PSKM IT | 3671 | 6614 | 9811 | 13,178 |
|      | CPU  | 0.1673 | 0.4941 | 0.6794 | 1.1053 |
|      | APSKM IT | 3295 | 6410 | 9781 | 13,391 |
|      | CPU  | 0.3137 | 1.0498 | 1.4678 | 2.4749 |

Bold values indicates method which outperforms other methods
Table 4 IT and CPU of the PSGD, PSKM, and APSKM methods for Problem 1 or Problem 2 with $C = \{ x : \langle A, x \rangle = b \}$ or $C = \{ x : \langle A, x \rangle \leq b \}$, $A$ from the SuiteSparse Matrix Collection, $x_0 = 0.5 \ast \text{ones}(n, 1)$ and $\varepsilon = 10^{-5}$

| Name       | IT       | CPU     | IT       | CPU     |
|------------|----------|---------|----------|---------|
| Problem 1  | PSGD IT  | 98,368  | 8.4058   | 9.1866  |
|            | PSKM IT  | 13,077  | 1.5072   | 1.6017  |
|            | APSKM IT | 12,080  | 2.3054   | 1.5406  |
| Problem 2  | PSGD IT  | 118,813 | 11.3259  | 5.2400  |
|            | PSKM IT  | 14,202  | 1.9406   | 1.5106  |
|            | APSKM IT | 14,541  | 3.1984   | 2.5610  |

Bold values indicates method which outperforms other methods

Fig. 2 Pictures of $\| f (x) \|_2$ versus CPU for the PSKM, APSKM, and PSGD methods on Problem 3 with $C = \{ x : \langle A, x \rangle = b \}$, $A$ is cis-n4c6-b14(left) or $C = \{ x : \langle A, x \rangle \leq b \}$, $A$ is n4c5-b5(right). $x_0 = 0.5 \ast \text{ones}(n, 1)$

Fig. 3 Pictures of $\| f (x) \|_2$ versus CPU and IT of the PSKM, APSKM, and PSGD methods on Problem 3 with $C = \{ x : \langle A, x \rangle = b \}$; $x_0 = \text{ones}(n, 1)$, $m = 7500$, $n = 15,000$ $k_c = 3000$ and $\xi = 0.3$ (left) or 0.7 (right)

$E_{j_1-1} \| x_{j_1} - x^* \|_2$ made in this iteration is given by $(1 - 2\eta) \sum_{j=0}^{m-\beta} \frac{\beta^j}{\beta^j} \frac{f_j^2(x_{j_1-1})}{\| \nabla f_j(x_{j_1-1}) \|_2^2}$. 
Fig. 4 Pictures of $\|f(x)\|_2$ versus IT and CPU for the APSKM method on Problem 3 with $C = \{x : \langle A, x \rangle = b\}$ when $x_0 = \text{ones}(n, 1)$, $m = 7500$, $n = 15000$, $k_c = 3000$ and $\xi$ is different.

Table 5 IT and CPU of the PSGD, PSKM, and APSKM methods for Problem 1 with $C = \{x : \langle A, x \rangle = b\}$, $m = 5000$, $x_0 = 0.5 \ast \text{ones}(n, 1)$ and $\varepsilon = 10^{-4}$

| $\xi$ | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 |
|-------|-----|-----|-----|-----|-----|
| $k_c = 300$ |
| PSGD | IT | - | - | 27,253 | 46,620 | 112,430 |
| CPU | - | - | 1.6779 | 2.8708 | 6.8706 |
| PSKM | IT | 6390 | 6201 | 7512 | 7494 | 1860 |
| CPU | 0.8447 | 2.1523 | 2.6123 | 2.5642 | 0.6620 |
| APSKM | IT | 3107 | 2029 | 1287 | 693 | 117 |
| CPU | 1.0001 | 1.4160 | 0.9176 | 0.5032 | 0.1019 |
| $k_c = 500$ |
| PSGD | IT | - | - | 40,275 | 78,502 | 169,179 |
| CPU | - | - | 2.7559 | 5.3622 | 11.4265 |
| PSKM | IT | 6966 | 8419 | 9328 | 8231 | 1862 |
| CPU | 2.4967 | 2.9682 | 0.7258 | 0.6372 | 0.1514 |
| APSKM | IT | 3488 | 3014 | 1784 | 1215 | 124 |
| CPU | 2.5679 | 2.2439 | 0.3219 | 0.2228 | 0.0352 |
| $k_c = 1000$ |
| PSGD | IT | - | - | 79,079 | 142,208 | 274,670 |
| CPU | - | - | 6.1641 | 10.7412 | 20.9303 |
| PSKM | IT | 9208 | 9286 | 10,695 | 8724 | 1894 |
| CPU | 0.7803 | 0.7830 | 0.9003 | 2.3935 | 0.1826 |
| APSKM | IT | 4675 | 3710 | 3245 | 1637 | 137 |
| CPU | 0.9159 | 0.7305 | 0.6383 | 0.9301 | 0.0553 |

Bold values indicates method which outperforms other methods.

If we use $\gamma$ to denote the smallest non-zero value of $\left\{ \frac{f_j^2(x_{j+1})}{\|\nabla f_j(x_{j+1})\|^2} \right\}_{j=0}^{m-\beta}$, then we have that

$$
(1 - 2\eta) \sum_{j=0}^{m-\beta} \left( \frac{\beta - 1}{\beta - 1} \right) \left( \begin{array}{c} m - \beta \\ \beta - 1 \end{array} \right) \frac{f_j^2(x_{j+1})}{\|\nabla f_j(x_{j+1})\|^2}.
$$
Table 6 IT and CPU of the PSGD, PSKM, and APSKM methods for Problem 2 with $C = \{x : \langle A, x \rangle = b \}$, $m = 10000$, $x_0 = 0.5 \times \text{ones}(n, 1)$ and $\varepsilon = 10^{-4}$

| $k_c$ = 300 | | | | | |
|---|---|---|---|---|---|
| $\xi$ | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 |
| PSGD IT | 40,928 | 35,472 | 38,647 | 55,725 | 18,762 |
| PSGD CPU | 2.5224 | 2.1545 | 2.4500 | 3.3833 | 1.1458 |
| PSKM IT | 10,988 | 10,661 | 9274 | 6993 | 2099 |
| PSKM CPU | 0.8504 | 0.7951 | 0.6914 | 0.5262 | 0.1611 |
| APSKM IT | 3692 | 1807 | 1639 | 859 | 180 |
| APSKM CPU | 0.6274 | 0.3066 | 0.2738 | 0.1465 | 0.0350 |

| $k_c$ = 500 | | | | | |
|---|---|---|---|---|---|
| $\xi$ | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 |
| PSGD IT | 42,065 | 42,246 | 56,107 | 75,285 | 20,291 |
| PSGD CPU | 4.0855 | 4.1093 | 5.4494 | 7.2859 | 1.9886 |
| PSKM IT | 11,415 | 10,690 | 9355 | 7103 | 2038 |
| PSKM CPU | 1.2713 | 1.2008 | 1.0503 | 0.8031 | 0.2384 |
| APSKM IT | 5070 | 3611 | 2433 | 1248 | 118 |
| APSKM CPU | 1.0334 | 0.7475 | 0.5114 | 0.2642 | 0.0411 |

| $k_c$ = 1000 | | | | | |
|---|---|---|---|---|---|
| $\xi$ | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 |
| PSGD IT | 61,130 | 72,535 | 88,528 | 108,304 | 21,198 |
| PSGD CPU | 61,130 | 72,535 | 88,528 | 108,304 | 21,198 |
| PSKM IT | 4,6189 | 5,4521 | 6,6552 | 8,1282 | 1,6060 |
| PSKM CPU | 1,0210 | 1,0029 | 0.8732 | 0.6582 | 0.2052 |
| APSKM IT | 6819 | 6746 | 3615 | 1924 | 151 |
| APSKM CPU | 1,3773 | 1,3554 | 0.7416 | 0.4027 | 0.0588 |

Bold values indicate method which outperforms other methods

\[
\begin{align*}
(1 - 2\eta)(1 - (\frac{s}{m})^\beta)\gamma & \approx (1 - 2\eta)(1 - (\frac{s}{m})^\beta)\gamma, \quad \beta \leq s, \\
(1 - 2\eta)\gamma & \quad \beta > s,
\end{align*}
\]

where the approximation comes from De Loera et al. (2017) in the first case.

Let $C_1$ and $C_2$ be the average cost of computing $f_i(x)$ ($i = 1, 2, \ldots, m$) and $\nabla f_i(x)$ ($i = 1, 2, \ldots, m$), respectively. In the NSKM method, we divide the computation cost of each iteration into two parts: the computation time of steps related to $\beta$ (Steps 2, 3, and 4) and that of other steps. We can model the computation cost in a fixed iteration as $\alpha_1 + \alpha_2 \beta$, where $\alpha_1$ consists of $C_2$ and computation costs in Step 6 and Step 7 of the NSKM method and $\alpha_2$ includes $\beta C_1$ and the computation costs of Step 2 and Step 4 of the NSKM method. We define a value for $\beta$ that evaluates the ratio of improvement made and the computation cost gain($\beta$) = \[
\begin{cases}
(1 - 2\eta)(1 - (\frac{s}{m})^\beta)\gamma & \quad \beta \leq s, \\
\frac{(1-2\eta)(1-(\frac{s}{m})^\beta)\gamma}{\alpha_1 + \alpha_2 \beta} & \quad \beta > s,
\end{cases}
\]

Since we only want to understand the changing trend of gain($\beta$) with $\beta$, we simplify

gain($\beta$) = \[
\begin{cases}
\frac{1-(\frac{s}{m})^\beta}{\alpha_1 + \alpha_2 \beta} & \quad \beta \leq s, \\
\frac{1}{\alpha_1 + \alpha_2 \beta} & \quad \beta > s.
\end{cases}
\]

In Fig. 5, we plot the picture of gain($\beta$) versus $\beta$ for different choices of $s$. From this picture, we can observe that the optimal size of $\beta$ also increases when the size of $s$ increases. This is
Fig. 5 Pictures of gain(β) versus β for various numbers of satisfied equations s when
m = 8000, α_1 = 1000, and α_2 = 16000

because that with the more equations are satisfied if we choose a small β, it is possible that many satisfied equations are in our selection. Thus, small progress is made in this iteration. Inspired by this phenomenon, we can increase β throughout the iterations.

6 Conclusions

We have proposed the NSKM method for solving large-scale nonlinear equations. At each step, only a part of residuals are computed. Furthermore, we have developed two variants of the NSKM method for solving large-scale nonlinear equations with finite convex constraints. Convergence analysis of the proposed methods is given.

From the numerical point of view, some numerical results show that the NSKM method with the sample of the right size is more effective than the NRK method in terms of CPU times. Moreover, two variants of the NSKM method can converge well to the solution of the constrained nonlinear problems and the APSKM method is superior to the PSKM method for nonlinear problems with large near-linear correlation equality constraints.

Acknowledgements This research was supported by the Fundamental Research Funds for the Central Universities (Grant No. 18CX02041A), the Shandong Provincial Natural Science Foundation (Grant No. ZR2020MD060), and the National Natural Science Foundation of China (Grant Nos. 42176011, 62231028).

Appendix

A Proof of Lemma 2

Proof

\[
\|x_{k+1} - x^+\|^2 - \|x_k - x^+\|^2 \\
= \|x_{k+1} - x_k\|^2 + 2 \langle x_{k+1} - x_k, x_k - x^+ \rangle \\
= -\frac{f_{ik+1}(x_k)}{\|\nabla f_{ik+1}(x_k)\|^2} \nabla f_{ik+1}(x_k)^T \|^2 + 2 \left(-\frac{f_{ik+1}(x_k)}{\|\nabla f_{ik+1}(x_k)\|^2} \nabla f_{ik+1}(x_k)^T, x_k - x^+ \right)
\]
\[
\begin{align*}
&= \frac{f^2_{ik+1}(x_k)}{\|\nabla f_{ik+1}(x_k)\|^2} - 2 \frac{f_{ik+1}(x_k)}{\|\nabla f_{ik+1}(x_k)\|^2} \nabla f_{ik+1}(x_k)(x_k - x^*) \\
&= \frac{f^2_{ik+1}(x_k)}{\|\nabla f_{ik+1}(x_k)\|^2} - 2 \frac{f_{ik+1}(x_k)}{\|\nabla f_{ik+1}(x_k)\|^2} f_{ik+1}(x_k) \\
&\quad + 2 \frac{f_{ik+1}(x_k)}{\|\nabla f_{ik+1}(x_k)\|^2} (f_{ik+1}(x_k) - f_{ik+1}(x^*) - \nabla f_{ik+1}(x_k)(x_k - x^*)).
\end{align*}
\]

When \( k = 0, x_0 \in \mathcal{B}_\rho(x_0) \) and \( |f_i(x_0) - f_i(x^*) - \nabla f_i(x_0)(x_0 - x^*)| \leq \eta_i |f_i(x_0) - f_i(x^*)| \) \((i = 1, 2, \ldots, m)\), then we have
\[
\|x_1 - x^*\|^2 - \|x_0 - x^*\|^2 \\
= \frac{f^2_{i_1}(x_0)}{\|\nabla f_{i_1}(x_0)\|^2} + 2 \frac{f_{i_1}(x_0)}{\|\nabla f_{i_1}(x_0)\|^2} (f_{i_1}(x_0) - f_{i_1}(x^*) - \nabla f_{i_1}(x_0)(x_0 - x^*)) \\
- 2 \frac{f_{i_1}(x_0)}{\|\nabla f_{i_1}(x_0)\|^2} f_{i_1}(x_0) \\
\leq \frac{f^2_{i_1}(x_0)}{\|\nabla f_{i_1}(x_0)\|^2} + 2 \eta_{i_1} \frac{f_{i_1}(x_0)}{\|\nabla f_{i_1}(x_0)\|^2} |f_{i_1}(x_0)| - 2 \frac{f^2_{i_1}(x_0)}{\|\nabla f_{i_1}(x_0)\|^2} \\
= -(1 - 2\eta_{i_1}) \frac{f^2_{i_1}(x_0)}{\|\nabla f_{i_1}(x_0)\|^2}.
\]

Since \( x^* \in \mathcal{B}_{\rho/2}(x_0) \) and (11), we have
\[
\|x_1 - x_0\| = \|x_1 - x^* + x^* - x_0\| \leq \|x_1 - x^*\| + \|x^* - x_0\| \leq \rho.
\]
Thus, \( x_1 \in \mathcal{B}_\rho(x_0) \).

We assume that when \( k \leq n \) (\( n \in \mathbb{N} \)), \( x_k \in \mathcal{B}_\rho(x_0) \) and (4) holds, then, for \( k = n + 1 \), similar to the derivation of \( k = 0 \), we have \( x_{n+1} \in \mathcal{B}_\rho(x_0) \) and (4) holds. \( \square \)

**B Proof of Lemma 4**

**Proof** Since \( f : \mathcal{D}(f) \rightarrow \mathbb{R} \) is a convex function, we have that
\[
f((1 - \alpha)x + \alpha y) \leq (1 - \alpha) f(x) + \alpha f(y), \forall \alpha \in [0, 1], \forall x, y \in \mathcal{D}(f). \tag{12}
\]
By Taylor formula, it holds
\[
f((1 - \alpha)x + \alpha y) = f(x) + \alpha f'(x)(y - x) + o(\|\alpha(y - x)\|). \tag{13}
\]
Combining (12) and (13), we obtain that
\[
f(y) - f(x) \geq f'(x)(y - x) + \frac{o(\|\alpha(y - x)\|)}{\alpha}.
\]
Let \( \alpha \rightarrow 0 \)
\[
f(y) \geq f(x) + f'(x)(y - x).
\]
This completes the proof. \( \square \)
C Proof of Lemma 8

**Proof** There are two cases to consider the following:

Case 1. $x_{k+\frac{2}{3}} = x_{k+\frac{2}{3}}$. In this case, we have

$$x_{k+\frac{1}{2}} = x_{k+\frac{2}{3}}.$$

Case 2. $x_{k+\frac{2}{3}} \neq x_{k+\frac{1}{2}}$. By Lemma 7, we obtain that

$$\|x_{k+\frac{2}{3}} - x^*\|_2^2 \leq \|x_{k+\frac{1}{2}} - x^*\|_2^2 - \|x_{k+\frac{1}{2}} - x_{k+\frac{2}{3}}\|_2^2.$$

Because $x_{k+\frac{1}{2}} \neq x_{k+\frac{2}{3}}$, we get that

$$\|x_{k+\frac{2}{3}} - x^*\|_2^2 < \|x_{k+\frac{1}{2}} - x^*\|_2^2.$$

Besides, from $x_{k+\frac{2}{3}} \neq x_{k+\frac{1}{2}}$, it can also be obtained that

$$\|x_{k+\frac{2}{3}} - x^*\|_2^2 \leq \|x_{k+\frac{2}{3}} - x^*\|_2^2 - \|x_{k+\frac{2}{3}} - x_{k+\frac{3}{4}}\|_2^2 < \|x_{k+\frac{2}{3}} - x^*\|_2^2.$$

Therefore, $\|x_{k+\frac{2}{3}} - x^*\|_2^2 < \|x_{k+\frac{2}{3}} - x^*\|_2^2 < \|x_{k+\frac{1}{2}} - x^*\|_2^2$, which implies that $x_{k+\frac{1}{2}} \neq x_{k+\frac{2}{3}}$.

This completes the proof. \(\square\)

D Proof of Lemma 9

**Proof** We first show that $\lambda_k \geq 1$. Observe that

$$2\left(x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}, x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right)$$

$$= \left\|x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\|^2 + \left\|x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\|^2 - \left\|x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\|^2$$

$$= 2\left(\left\|x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\|^2 - \left\|x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\|^2\right),$$

where the last inequality follows from Lemma 7.

Hence $\left\langle x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}, x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\rangle \leq \left\|x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\|^2$, which implies that

$$\lambda_k = \frac{\left\|x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\|^2}{\left\langle x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}, x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\rangle} \geq 1. \quad (14)$$

Next, we will prove that $x_k - x_{k+\frac{2}{3}}$ and $x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}$ are orthogonal

$$\left\langle x_k - x_{k+\frac{2}{3}}, x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\rangle = \left\langle \left(x_k - x_{k+\frac{2}{3}}\right) + \lambda_k \left(x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right), x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\rangle$$

$$= \left\|x_k - x_{k+\frac{2}{3}}\right\|^2 + \lambda_k \left\|x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\|^2$$

$$= \left\|x_k - x_{k+\frac{2}{3}}\right\|^2 \left(1 + \frac{\left\langle x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}, x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\rangle}{\left\langle x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}, x_{k+\frac{1}{3}} - x_{k+\frac{1}{3}}\right\rangle}\right)$$

$$= 0. \quad (15)$$
Finally, we utilize (14) and (15) to prove (10). For every $x \in C_{\alpha_1} \cap C_{\alpha_2}$, we have
\[
\|x_k - x\|^2 = \|x_k - x_{k-\frac{1}{5}}\|^2 + \|x_{k-\frac{1}{5}} - x\|^2 + 2\langle x_k - x_{k-\frac{1}{5}}, x_{k-\frac{1}{5}} - x \rangle.
\]
By writing $\langle x_k - x_{k-\frac{1}{5}}, x_{k-\frac{1}{5}} - x \rangle = \langle x_k - x_{k-\frac{1}{5}}, x_{k-\frac{1}{5}} - x_k \rangle + \langle x_k - x_{k-\frac{1}{5}}, x_k - x \rangle$, we find that
\[
\|x_k - x\|^2 = \|x_{k-\frac{1}{5}} - x\|^2 - \|x_k - x_{k-\frac{1}{5}}\|^2 + 2\langle x_k - x_{k-\frac{1}{5}}, x_k - x \rangle.
\]
By the definition of $x_k$, we obtain that
\[
\langle x_k - x_{k-\frac{1}{5}}, x_k - x \rangle = (1 - \lambda_k)\langle x_{k-\frac{1}{5}} - x_{k-\frac{1}{5}}, x_k - x \rangle
\]
\[
= (1 - \lambda_k)\langle x_{k-\frac{1}{5}} - x_{k-\frac{1}{5}}, x_k - x \rangle + \langle x_{k-\frac{1}{5}} - x_{k-\frac{1}{5}}, x_k - x \rangle.
\]
For the first inner product of the above formula, we have
\[
\langle x_{k-\frac{1}{5}} - x_{k-\frac{1}{5}}, x_k - x \rangle = \langle x_{k-\frac{1}{5}} - x_{k-\frac{1}{5}}, x_k - x_{k-\frac{1}{5}} \rangle + \langle x_{k-\frac{1}{5}} - x_{k-\frac{1}{5}}, x_{k-\frac{1}{5}} - x \rangle \geq 0,
\]
where the inequality comes from Lemma 7 and (15).
For the second inner product, we can obtain
\[
\langle x_{k-\frac{1}{5}} - x_{k-\frac{1}{5}}, x_k - x \rangle \geq \langle x_{k-\frac{1}{5}} - x_{k-\frac{1}{5}}, x_k - x_{k-\frac{1}{5}} \rangle
\]
\[
= (1 - \lambda_k)\langle x_{k-\frac{1}{5}} - x_{k-\frac{1}{5}}, x_{k-\frac{1}{5}} - x_{k-\frac{1}{5}} \rangle \geq 0,
\]
where the first inequality follows Lemma 7, the second equality comes from the definition of $x_k$, and the second inequality is from Lemma 7 and (14).

Thus
\[
\|x_k - x\|^2 \leq \|x_{k-\frac{1}{5}} - x\|^2 - \|x_k - x_{k-\frac{1}{5}}\|^2 \leq \|x_{k-\frac{1}{5}} - x\|^2.
\]
(16)

Since the iteration points $x_i$ ($i = k - \frac{3}{5}, k - \frac{2}{5}, k - \frac{1}{5}$) are obtained by projecting on the closed convex sets, by Lemma 7, it results in
\[
\|x_i - x\|^2 \leq \|x_{i-\frac{1}{5}} - x\|^2 - \|x_i - x_{i-\frac{1}{5}}\|^2.
\]
Thus
\[
\|x_{k-\frac{1}{5}} - x\|^2 \leq \|x_{k-\frac{1}{5}} - x\|^2 - \|x_{k-\frac{1}{5}} - x\|^2 \leq \|x_{k-\frac{1}{5}} - x\|^2.
\]
(17)

From (16) and (17), we get that
\[
\|x_k - x\|^2 \leq \|x_{k-\frac{4}{5}} - x\|^2.
\]

\[\square\]

References

Bauschke HH, Borwein JM (1996) On projection algorithms for solving convex feasibility problems. SIAM Rev 38(3):367–426. https://doi.org/10.1137/S0036144593251710
Brinkhuis J (2020) Convex analysis for optimization: a unified approach, 1st edn. Springer Cham, Berlin. https://doi.org/10.1007/978-3-030-41804-5
Davis TA, Hu Y (2011) The University of Florida sparse matrix collection. ACM Trans Math Softw (TOMS) 38(1):1–25
De Loera JA, Haddock J, Needell D (2017) A sampling Kaczmarz–Motzkin algorithm for linear feasibility. SIAM J Sci Comput 39(5):S66–S87. https://doi.org/10.1137/16M1073807
Dennis JEJ, Moré JJ (1977) Quasi-Newton methods, motivation and theory. SIAM Rev 19(1):46–89. https://doi.org/10.1137/1019005
Haltmeier M, Leitao A, Scherzer O (2007) Kaczmarz methods for regularizing nonlinear ill-posed equations I: convergence analysis. Inverse Probl Imaging 1(2):289–298. https://doi.org/10.3934/ipi.2007.1.289
Jin B, Zhou Z, Zou J (2020) On the convergence of stochastic gradient descent for nonlinear ill-posed problems. SIAM J Optim 30(2):1421–1450. https://doi.org/10.1137/1019005
Kelley CT (1995) Iterative Methods for Linear and Nonlinear Equations. Society for Industrial and Applied Mathematics, Philadelphia. https://doi.org/10.1002/cm.1286
Li D, Fukushima M (1999) A globally and superlinearly convergent Gauss–Newton based BFGS method for symmetric nonlinear equations. SIAM J Numer Anal 37(1):152–172. https://doi.org/10.1137/S0036142998335704
Lukšan L, Matonoha C, Vlcek J (2018) Problems for nonlinear least squares and nonlinear equations. Tech. Rep. 1259, Institute of Computer Science, Academy of Sciences of the Czech Republic. https://doi.org/10.13140/RG.2.2.16112.10248
Qin T, Etesami SR (2020) A randomized algorithm for generalized accelerated projection method. IEEE Control Syst. Lett. 5(1):85–90. https://doi.org/10.1109/LCNSYS.2020.3000714
Wang M, Bertsekas DP (2013) Incremental constraint projection-proximal methods for nonsmooth convex optimization. SIAM J Optimiz (to appear)
Wang Q, Li W, Bao W et al (2022) Nonlinear Kaczmarz algorithms and their convergence. J Comput Appl Math 399:113–720. https://doi.org/10.1016/j.cam.2021.113720
Yamashita N, Fukushima M (2001) On the rate of convergence of the Levenberg–Marquardt method. Topics in numerical analysis. Springer, Berlin, pp 239–249

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