

On fast greedy block Kaczmarz methods for solving large consistent linear systems

A-Qin Xiao

School of Mathematical Sciences, Tongji University,
Shanghai, 200092, PR China.
Email:xiaoaqin@tongji.edu.cn

Jun-Feng Yin*

School of Mathematical Sciences, Tongji University,
Shanghai, 200092, PR China.
Email:yinjf@tongji.edu.cn

and

Ning Zheng

School of Mathematical Sciences, Tongji University,
Shanghai, 200092, PR China.
Email:nzheng@tongji.edu.cn

Abstract

A class of fast greedy block Kaczmarz methods combined with general greedy strategy and average technique are proposed for solving large consistent linear systems. Theoretical analysis of the convergence of the proposed method is given in detail. Numerical experiments show that the proposed methods are efficient and faster than the existing methods.

Keywords. Linear systems, Kaczmarz method, Modified greedy strategies, Average block, Convergence property.

1 Introduction

Consider the solution of consistent linear algebraic equations

$$Ax = b, \quad (1.1)$$

where $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$, one of the classical and popular iteration methods is the Kaczmarz method [1]. Due to its simplicity and efficiency, it was deeply studied and widely used in many practical scientific and engineering applications, for instance, computer tomography(CT) [2], image reconstruction [3], machine learning [4] and option pricing [5].

Let $A^{(i)}$ be the i -th row of matrix A and $b^{(i)}$ be the i -th entry of vector b , respectively. Given an initial guess vector $x_0 \in \mathbb{R}^n$, the classical Kaczmarz method iterates by

$$x_{k+1} = x_k + \frac{b^{(i_k)} - A^{(i_k)}x_k}{\|A^{(i_k)}\|_2^2} (A^{(i_k)})^T, \quad k = 0, 1, 2, \dots,$$

*Corresponding author.

where $i_k = (k \bmod m) + 1$. To improve the convergence of Kaczmarz method, Strohmer and Vershynin [6] proposed a randomized Kaczmarz method by selecting the row index i_k with probability proportion to $\|A^{(i_k)}\|_2^2$ and proved its linear convergence rate in expectation. Bai and Wu [7] constructed a greedy randomized Kaczmarz method to accelerate the convergence performance. For more variants of the randomized and greedy Kaczmarz methods, we refer the reader to [8–11].

The idea of block Kaczmarz method can date back to the work of Elfving in [12], which used many equations simultaneously at each iteration. The block Kaczmarz method can be described as

$$x_{k+1} = x_k + A_{\mathcal{J}_k}^\dagger (b_{\mathcal{J}_k} - A_{\mathcal{J}_k} x_k), \quad k = 0, 1, 2, \dots, \quad (1.2)$$

where $A_{\mathcal{J}_k}^\dagger$ represents the Moore-Penrose pseudoinverse of the chosen submatrix $A_{\mathcal{J}_k}$ and \mathcal{J}_k is the block row indices. Needell and Tropp [13] proposed a randomized block Kaczmarz method, where \mathcal{J}_k is chosen uniformly at random from $[m] = \{1, 2, \dots, m\}$. Further, Niu and Zheng [14] proposed a greedy block Kaczmarz method which adaptively choose the block row indices without predetermining a partition of the row indices of the matrix A .

However, each iterate of the block Kaczmarz method require the computation of the pseudoinverse of the selected submatrix corresponding to the residual subvector and it usually costs expensive. Necoara [15] established a unified framework of randomized average block Kaczmarz methods by taking a convex combination of several updatings as a new direction and implemented on the distributed computing units. Miao and Wu [16] proposed an average block variant of the greedy randomized Kaczmarz method [7] and studied its convergence.

In this work, we construct a class of fast greedy block Kaczmarz methods with average technique to avoid computing the pseudoinverse of submatrices of the coefficients matrix, where a modified greedy strategy utilizing the general norm of residual vectors is proposed and well studied, which can choose the working rows based on this greedy criterion dynamically and flexibly. Theoretical analysis of the convergence of the proposed method is given in detail. Numerical experiments further demonstrate that the proposed methods are efficient and faster than the existing methods.

The rest of this paper is organized as follows. In Section 2, a class of fast greedy block Kaczmarz methods is presented with a modified greedy row selection strategy and the convergence theory of the proposed method is established. Numerical experiments are reported in Section 3 to display the efficiency of the new method. Finally, we draw the conclusions in Section 4.

2 The fast greedy block Kaczmarz method

In this section, after reviewing the fast deterministic block Kaczmarz method, we present a class of fast greedy block Kaczmarz methods for solving large consistent linear systems (1.1) by using a modified greedy row selection strategy and the averaging technique.

Let ξ_k be a linear combination of unit column vectors $e_i \in \mathbb{R}^m$ ($i \in \tau_k$) and its coefficients are the corresponding entries of the residual vector, that is,

$$\xi_k = \sum_{i \in \mathcal{I}_k} (b^{(i)} - A^{(i)} x_k) e_i, \quad k = 0, 1, 2, \dots.$$

Similar to the iteration of average block Kaczmarz methods [15], the stepsize and weight are set to be $\alpha_k = \frac{\|\xi_k\|_2^2 \|A_{\mathcal{I}_k}\|_F^2}{\|A^T \xi_k\|_2^2}$ and $\omega_i^k = \frac{\|A^{(i)}\|_2^2}{\|A_{\mathcal{I}_k}\|_F^2}$, $i \in \mathcal{I}_k$ respectively, the fast deterministic block

Kaczmarz method can iterate as follows

$$\begin{aligned}
x_{k+1} &= x_k + \alpha_k \left(\sum_{i \in \mathcal{I}_k} \omega_i^k \frac{b^{(i)} - A^{(i)} x_k}{\|A^{(i)}\|_2^2} (A^{(i)})^T \right) \\
&= x_k + \frac{\|\xi_k\|_2^2 \|A_{\mathcal{I}_k}\|_F^2}{\|A^T \xi_k\|_2^2} \left(\sum_{i \in \mathcal{I}_k} \frac{\|A^{(i)}\|_2^2}{\|A_{\mathcal{I}_k}\|_F^2} \cdot \frac{b^{(i)} - A^{(i)} x_k}{\|A^{(i)}\|_2^2} (A^{(i)})^T \right) \\
&= x_k + \frac{\xi_k^T (b - Ax_k)}{\|A^T \xi_k\|_2^2} \left(\sum_{i \in \mathcal{I}_k} (b^{(i)} - A^{(i)} x_k) (A^{(i)})^T \right) \\
&= x_k + \frac{\xi_k^T (b - Ax_k)}{\|A^T \xi_k\|_2^2} A^T \xi_k. \tag{2.1}
\end{aligned}$$

where the block indices \mathcal{I}_k is chosen by

$$\mathcal{I}_k = \left\{ i \mid |b^{(i)} - A^{(i)} x_k|^2 \geq \gamma_k \|b - Ax_k\|_2^2 \|A^{(i)}\|_2^2 \right\}$$

with

$$\gamma_k = \frac{1}{2} \left(\frac{1}{\|b - Ax_k\|_2^2} \max_{1 \leq i \leq m} \left\{ \frac{|b^{(i)} - A^{(i)} x_k|^2}{\|A^{(i)}\|_2^2} \right\} + \frac{1}{\|A\|_F^2} \right).$$

One drawback of the fast deterministic block Kaczmarz method is the greedy strategy for choosing the working block indices. The size of the block \mathcal{I}_k may be small if the Frobenius norm of the coefficient matrix A is very small, which may lead to a very slow convergence.

In order to further accelerate the fast deterministic block Kaczmarz method, we propose to determine the working rows by a modified greedy strategy that utilizes the general norm of residual vectors. Let $\eta \in (0, 1]$ and $p \in [1, +\infty)$, the control index subset is defined as

$$\tau_k = \left\{ i \mid |b^{(i)} - A^{(i)} x_k|^p \geq \epsilon_k \|A^{(i)}\|_p^p \right\}, \quad k = 0, 1, 2, \dots,$$

where

$$\epsilon_k = \eta \cdot \max_{1 \leq i \leq m} \left\{ \frac{|b^{(i)} - A^{(i)} x_k|^p}{\|A^{(i)}\|_p^p} \right\}.$$

It follows that

$$\epsilon_k = \eta \cdot \max_{1 \leq i \leq m} \left\{ \frac{|b^{(i)} - A^{(i)} x_k|^p}{\|A^{(i)}\|_p^p} \right\} \leq \max_{1 \leq i \leq m} \left\{ \frac{|b^{(i)} - A^{(i)} x_k|^p}{\|A^{(i)}\|_p^p} \right\},$$

which implies that there is at least one index $j \in [m]$ such that

$$\frac{|b^{(j)} - A^{(j)} x_k|^p}{\|A^{(j)}\|_p^p} = \max_{1 \leq i \leq m} \left\{ \frac{|b^{(i)} - A^{(i)} x_k|^p}{\|A^{(i)}\|_p^p} \right\},$$

then $j \in \tau_k$, i.e. τ_k is not empty.

Given an initial guess vector $x_0 \in \mathbb{R}^n$, the fast greedy block Kaczmarz method is described in Algorithm 1.

Moreover, the theoretical analyses for the convergence performance of the fast greedy block Kaczmarz method are established as follows.

Algorithm 1 The fast greedy block Kaczmarz method (FGBK)

Input: $A, b, x_0, l, \eta \in (0, 1]$ and $p \in [1, +\infty)$

Output: x_l

1: **for** $k = 0, 1, 2, \dots, l - 1$ **do**

2: Compute

$$\epsilon_k = \eta \cdot \max_{1 \leq i \leq m} \left\{ \frac{|b^{(i)} - A^{(i)}x_k|^p}{\|A^{(i)}\|_p^p} \right\}.$$

3: Determine the control index set of positive integers

$$\tau_k = \left\{ i \mid |b^{(i)} - A^{(i)}x_k|^p \geq \epsilon_k \|A^{(i)}\|_p^p \right\}. \quad (2.2)$$

4: Compute

$$\xi_k = \sum_{i \in \tau_k} \left(b^{(i)} - A^{(i)}x_k \right) e_i. \quad (2.3)$$

5: Set

$$x_{k+1} = x_k + \frac{\xi_k^T (b - Ax_k)}{\|A^T \xi_k\|_2^2} A^T \xi_k. \quad (2.4)$$

6: **end for**

Theorem 2.1. *Given an initial vector $x_0 \in \mathbb{R}^n$ in the column space of A^T . Then, the sequence $\{x_k\}_{k=0}^\infty$ generated by the fast greedy block Kaczmarz method converges to the unique least-norm solution $x_* = A^\dagger b$. Moreover, the norm of the error of the approximate solution satisfies*

$$\|x_{k+1} - x_*\|_2^2 \leq (1 - \beta_k(\eta, p) \sigma_{\min}^2(A)) \|x_k - x_*\|_2^2, \quad k \geq 0,$$

where $\beta_k(\eta, p) = \frac{\eta^{\frac{2}{p}}}{\sum_{i \in [m] \setminus \tau_{k-1}} \|A^{(i)}\|_p^2} \cdot \frac{\sum_{i \in \tau_k} \|A^{(i)}\|_p^2}{\sigma_{\max}^2(A_{\tau_k})}$, $\eta \in (0, 1]$ and $p \in [1, +\infty)$.

Proof. For each $k > 0$, denote the projector $P_k = \frac{A^T \xi_k \xi_k^T A}{\|A^T \xi_k\|_2^2}$, since $P_k^T = P_k, P_k^2 = P_k$, so P_k is an orthogonal projection. From the iterate scheme (2.4), it holds that

$$\begin{aligned} x_{k+1} - x_* &= x_k - x_* + \frac{\xi_k^T (b - Ax_k)}{\|A^T \xi_k\|_2^2} A^T \xi_k \\ &= x_k - x_* - \frac{\xi_k^T A (x_k - x_*)}{\|A^T \xi_k\|_2^2} A^T \xi_k \\ &= x_k - x_* - \frac{A^T \xi_k \xi_k^T A}{\|A^T \xi_k\|_2^2} (x_k - x_*). \end{aligned}$$

By the Pythagorean theorem, it follows that

$$\begin{aligned} \|x_{k+1} - x_*\|_2^2 &= \|(I - P_k)(x_k - x_*)\|_2^2 \\ &= \|x_k - x_*\|_2^2 - \|P_k(x_k - x_*)\|_2^2 \\ &= \|x_k - x_*\|_2^2 - \left\| \frac{\xi_k^T A (x_k - x_*)}{\|A^T \xi_k\|_2^2} A^T \xi_k \right\|_2^2 \\ &= \|x_k - x_*\|_2^2 - \frac{|\xi_k^T (b - Ax_k)|^2}{\|A^T \xi_k\|_2^2}. \end{aligned} \quad (2.5)$$

Let $E_k \in \mathbb{R}^{m \times |\tau_k|}$ be the matrix whose columns are consisted of all the vector $e_i \in \mathbb{R}^m$ with $i \in \tau_k$. Denote $A_{\tau_k} = E_k^T A$, $\widehat{\xi}_k = E_k^T \xi_k$, then

$$\left\| \widehat{\xi}_k \right\|_2^2 = \xi_k^T E_k E_k^T \xi_k = \|\xi_k\|_2^2 = \sum_{i \in \tau_k} |b^{(i)} - A^{(i)} x_k|^2, \quad (2.6)$$

and

$$\left\| A^T \xi_k \right\|_2^2 = \xi_k^T A A^T \xi_k = \widehat{\xi}_k^T E_k^T A A^T E_k \widehat{\xi}_k = \widehat{\xi}_k^T A_{\tau_k} A_{\tau_k}^T \widehat{\xi}_k = \left\| A_{\tau_k}^T \widehat{\xi}_k \right\|_2^2. \quad (2.7)$$

Therefore,

$$\left\| A_{\tau_k}^T \widehat{\xi}_k \right\|_2^2 = \widehat{\xi}_k^T A_{\tau_k} A_{\tau_k}^T \widehat{\xi}_k \leq \sigma_{\max}^2(A_{\tau_k}) \left\| \widehat{\xi}_k \right\|_2^2, \quad (2.8)$$

where $\sigma_{\max}(A_{\tau_k})$ represents the largest singular value of selected submatrix A_{τ_k} . By the definition of ξ_k in (2.3) and (2.6), it holds that

$$\begin{aligned} \xi_k^T (b - Ax_k) &= \left(\sum_{i \in \tau_k} (b^{(i)} - A^{(i)} x_k) e_i^T \right) (b - Ax_k) \\ &= \sum_{i \in \tau_k} \left((b^{(i)} - A^{(i)} x_k) e_i^T (b - Ax_k) \right) \\ &= \sum_{i \in \tau_k} |b^{(i)} - A^{(i)} x_k|^2 \\ &= \left\| \widehat{\xi}_k \right\|_2^2. \end{aligned} \quad (2.9)$$

Since both x_k and $x_* \in R(A^T)$, $x_k - x_* \in \mathcal{R}(A^T)$, then

$$\|b - Ax_k\|_2^2 = \|A(x_k - x_*)\|_2^2 \geq \sigma_{\min}^2(A) \|x_k - x_*\|_2^2. \quad (2.10)$$

From the equalities (2.7)–(2.10) and the definition of τ_k in (2.2), it follows that

$$\begin{aligned} \frac{|\xi_k^T (b - Ax_k)|^2}{\|A^T \xi_k\|_2^2} &= \frac{\left(\sum_{i \in \tau_k} |b^{(i)} - A^{(i)} x_k|^2 \right) \left\| \widehat{\xi}_k \right\|_2^2}{\left\| A_{\tau_k}^T \widehat{\xi}_k \right\|_2^2} \\ &\geq \frac{\sum_{i \in \tau_k} |b^{(i)} - A^{(i)} x_k|^2}{\sigma_{\max}^2(A_{\tau_k})} \\ &\geq \frac{\sum_{i \in \tau_k} (|b^{(i)} - A^{(i)} x_k|^p)^{\frac{2}{p}}}{\sigma_{\max}^2(A_{\tau_k})} \\ &\geq (\epsilon_k)^{\frac{2}{p}} \cdot \frac{\sum_{i \in \tau_k} \left(\|A^{(i)}\|_p^p \right)^{\frac{2}{p}}}{\sigma_{\max}^2(A_{\tau_k})} \\ &\geq (\epsilon_k)^{\frac{2}{p}} \cdot \frac{\sum_{i \in \tau_k} \|A^{(i)}\|_p^2}{\sigma_{\max}^2(A_{\tau_k})}. \end{aligned} \quad (2.11)$$

In addition, it is seen that

$$\begin{aligned} b - Ax_k &= b - A \left(x_{k-1} + \frac{\xi_{k-1}^T (b_{\tau_{k-1}} - A_{\tau_{k-1}} x_{k-1})}{\|A_{\tau_{k-1}}^T \xi_{k-1}\|_2^2} A_{\tau_{k-1}}^T \xi_{k-1} \right) \\ &= (b - Ax_{k-1}) - \frac{A \xi_{k-1}^T (b_{\tau_{k-1}} - A_{\tau_{k-1}} x_{k-1})}{\|A_{\tau_{k-1}}^T \xi_{k-1}\|_2^2} A_{\tau_{k-1}}^T \xi_{k-1}, k = 1, 2, \dots \end{aligned}$$

Therefore,

$$\begin{aligned} b_{\tau_{k-1}} - A_{\tau_{k-1}} x_k &= (b_{\tau_{k-1}} - A_{\tau_{k-1}} x_{k-1}) \\ &\quad - \frac{A_{\tau_{k-1}} \xi_{k-1}^T A_{\tau_{k-1}}^T \xi_{k-1}}{\|A_{\tau_{k-1}}^T \xi_{k-1}\|_2^2} (b_{\tau_{k-1}} - A_{\tau_{k-1}} x_{k-1}) = 0. \end{aligned}$$

It is known that

$$\begin{aligned} \|b - Ax_k\|_2^2 &= \sum_{i \in [m] \setminus \tau_{k-1}} \frac{|b^{(i)} - A^{(i)} x_k|^2}{\|A^{(i)}\|_p^2} \|A^{(i)}\|_p^2 \\ &= \sum_{i \in [m] \setminus \tau_{k-1}} \left(\frac{|b^{(i)} - A^{(i)} x_k|^p}{\|A^{(i)}\|_p^p} \right)^{\frac{2}{p}} \|A^{(i)}\|_p^2 \\ &\leq \left(\max_{1 \leq i \leq m} \left\{ \frac{|b^{(i)} - A^{(i)} x_k|^p}{\|A^{(i)}\|_p^p} \right\} \right)^{\frac{2}{p}} \cdot \sum_{i \in [m] \setminus \tau_{k-1}} \|A^{(i)}\|_p^2. \end{aligned}$$

It follows that

$$\begin{aligned} (\epsilon_k)^{\frac{2}{p}} &= \left(\eta \cdot \max_{1 \leq i \leq m} \left\{ \frac{|b^{(i)} - A^{(i)} x_k|^p}{\|A^{(i)}\|_p^p} \right\} \right)^{\frac{2}{p}} \geq \eta^{\frac{2}{p}} \cdot \frac{\|b - Ax_k\|_2^2}{\sum_{i \in [m] \setminus \tau_{k-1}} \|A^{(i)}\|_p^2} \\ &\geq \frac{\eta^{\frac{2}{p}}}{\sum_{i \in [m] \setminus \tau_{k-1}} \|A^{(i)}\|_p^2} \cdot \sigma_{\min}^2(A) \|x_k - x_*\|_2^2. \end{aligned} \tag{2.12}$$

From (2.11) and (2.12), it deduces that

$$\frac{|\xi_k^T (b - Ax_k)|^2}{\|A^T \xi_k\|_2^2} \geq \beta_k(\eta, p) \cdot \sigma_{\min}^2(A) \|x_k - x_*\|_2^2. \tag{2.13}$$

where $\beta_k(\eta, p) = \frac{\eta^{\frac{2}{p}}}{\sum_{i \in [m] \setminus \tau_{k-1}} \|A^{(i)}\|_p^2} \cdot \frac{\sum_{i \in \tau_k} \|A^{(i)}\|_p^2}{\sigma_{\max}^2(A_{\tau_k})}$, $\eta \in (0, 1]$ and $p \in [1, +\infty)$.

Finally, by combining (2.5) and (2.13), it follows that

$$\|x_{k+1} - x_*\|_2^2 \leq (1 - \beta_k(\eta, p) \sigma_{\min}^2(A)) \|x_k - x_*\|_2^2.$$

■

Note that the upper bound of convergence rate of the fast greedy block Kaczmarz method is related to the relaxation parameter η , the parameter p , the geometric properties of the coefficient matrix A and its row submatrices at each iteration. However, the practical convergence speed of the fast greedy block Kaczmarz methods could be faster than the upper bound.

3 Numerical experiments

In this section, a number of numerical experiments are presented to illustrate the efficiency of the fast greedy block Kaczmarz (FGBK) method, compared with the greedy block Kaczmarz (GBK) method [14] and the fast deterministic block Kaczmarz (FDBK) method [17] in terms of the number of iteration steps (denoted as ‘IT’) and the elapsed computing time in seconds (denoted as ‘CPU’).

In the numerical experiment, the solution vector x is firstly constructed and $b = Ax$ so that the linear system is consistent. All the iterations are started from the initial vector $x_0 = 0$, and terminated when the relative solution error (denoted as ‘RSE’) satisfies

$$\text{RSE} = \frac{\|x_k - x_*\|_2^2}{\|x_0 - x_*\|_2^2} < 10^{-6},$$

or the number of iteration steps exceeds a maximal number, e.g., 10000. For the greedy block Kaczmarz method, the control index set is determined by

$$\mathcal{J}_k = \left\{ i \mid |b^{(i)} - A^{(i)}x_k|^2 \geq \delta_k \max_{1 \leq i \leq m} \left\{ \frac{|b^{(i)} - A^{(i)}x_k|^2}{\|A^{(i)}\|_2^2} \right\} \|A^{(i)}\|_2^2 \right\}$$

with parameter δ_k is

$$\delta_k = \frac{1}{2} + \frac{1}{2} \frac{\|b - Ax_k\|_2^2}{\|A\|_F^2} \left(\max_{1 \leq i \leq m} \left\{ \frac{|b^{(i)} - A^{(i)}x_k|^2}{\|A^{(i)}\|_2^2} \right\} \right)^{-1}$$

while the relaxation parameter η_{exp} in the fast greedy block Kaczmarz method is experimentally selected by minimizing the numbers of total iterations.

In the first example, the matrix A is generated by the MATLAB function ‘randn’, where the size of the matrices is chosen to be 5000×10000 , 5000×12000 , 5000×14000 , 5000×16000 and 5000×18000 , respectively. In Table 1, the number of iterations and elapsed CPU time of the greedy block Kaczmarz, the fast deterministic block Kaczmarz and fast greedy block Kaczmarz methods with $p = 1, 2$ and 3 are listed respectively.

From Table 1, it can be observed that the GBK, FDBK, FGBK($p = 1$), FGBK($p = 2$) and FGBK($p = 3$) methods converge successfully and the FGBK-type methods outperform the other two methods in terms of both the iteration count and CPU time. Moreover, the fast greedy block Kaczmarz method with $p = 1$ requires the least number of iterations. It shows that the efficiency of the modified greedy row selection strategy and indicates that a small value of p may further accelerate the convergence.

In the second example, the matrices are taken from the SuiteSparse Matrix Collection [18] to further compare the convergence performances of these Kaczmarz methods. The test matrices ‘stat96v5’ and ‘crew1’ come from linear programming problems while ‘bibd_17_8’ and ‘bibd_16_8’ come from combinatorial problems. In Table 2, the sizes ($m \times n$), rank, density and condition number of the test matrices are listed respectively.

In Table 3, the number of iterations and CPU time of the greedy block Kaczmarz method, the fast deterministic block Kaczmarz method and the fast greedy block Kaczmarz method with $p = 1, 2$ and 3 are reported respectively.

From Table 3, it is observed that the proposed methods require fewer iterations and less CPU time than the other two Kaczmarz methods. For the matrix ‘stat96v5’, the proposed method with $p = 1$ has the least number of iteration and the least CPU time. For the other three matrices, the proposed method with $p = 1$ requires the least number of iterations. It further indicates that a small value of p may further improve the speed of convergence.

In Figure 1, the curves of the relative solution error versus the number of iterations are plotted for GBK, FDBK, FGBK($p = 1$), FGBK($p = 2$) and FGBK($p = 3$) respectively.

Table 1. Numerical results for random matrices.

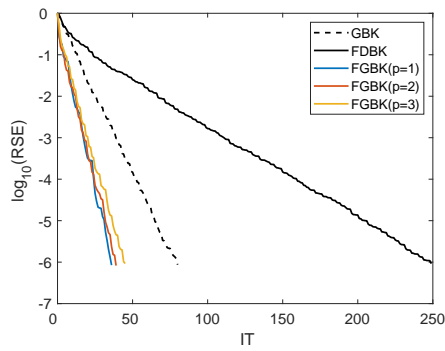
Method	$m \times n$	5000×10000	5000×12000	5000×14000	5000×16000	5000×18000
GBK	IT	543	349	251	208	169
	CPU	40.3710	36.0279	33.5738	35.6595	35.6040
	RSE	9.97×10^{-7}	9.92×10^{-7}	9.89×10^{-7}	9.82×10^{-7}	9.76×10^{-7}
FDBK	IT	559	356	256	209	170
	CPU	27.8735	21.4744	17.9981	16.5299	15.4575
	RSE	9.91×10^{-7}	9.85×10^{-7}	9.71×10^{-7}	9.54×10^{-7}	9.97×10^{-7}
FGBK(p=1)	η_{exp}	0.10	0.05	0.05	0.05	0.05
	IT	73	47	35	29	24
	CPU	4.5057	3.6111	3.1248	2.9545	2.7597
	RSE	9.53×10^{-7}	9.52×10^{-7}	9.29×10^{-7}	9.70×10^{-7}	8.80×10^{-7}
	η_{exp}	0.05	0.05	0.05	0.05	0.05
FGBK(p=2)	IT	74	48	36	30	25
	CPU	4.0051	3.1378	2.8106	2.6807	2.4948
	RSE	8.77×10^{-7}	8.03×10^{-7}	8.02×10^{-7}	7.92×10^{-7}	7.84×10^{-7}
	η_{exp}	0.05	0.05	0.05	0.05	0.05
FGBK(p=3)	IT	82	55	42	35	30
	CPU	6.0167	5.6553	5.5993	6.0705	6.3334
	RSE	9.36×10^{-7}	9.38×10^{-7}	8.38×10^{-7}	8.77×10^{-7}	9.37×10^{-7}
	η_{exp}	0.05	0.05	0.05	0.05	0.05

Table 2. Information of the matrices from SuiteSparse Matrix Collection.

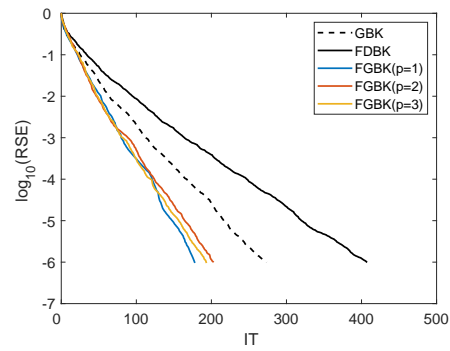
Name	stat96v5	crew1	bibd_17_8	bibd_16_8
$m \times n$	2307×75779	135×6469	136×24310	120×12870
rank	2307	135	136	120
density	0.13%	5.38%	20.59%	23.33%
cond(A)	19.52	18.20	9.04	9.54

Table 3. Numerical results for the matrices from SuiteSparse Matrix Collection.

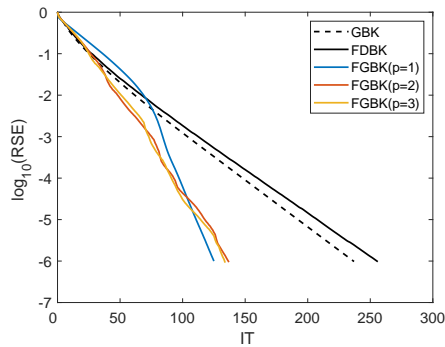
Method		stat96v5	crew1	bibd_17_8	bibd_16_8
GBK	IT	80	547	237	280
	CPU	1.2582	0.7068	19.7138	11.3608
	RSE	8.69×10^{-7}	9.85×10^{-7}	9.75×10^{-7}	9.83×10^{-7}
FDBK	IT	249	815	256	289
	CPU	0.2940	0.1438	0.3103	0.1846
	RSE	9.22×10^{-7}	9.89×10^{-7}	9.66×10^{-7}	9.91×10^{-7}
FGBK(p=1)	η_{exp}	0.05	0.20	0.10	0.10
	IT	36	356	125	138
	CPU	0.1787	0.1065	0.1829	0.1249
	RSE	8.46×10^{-7}	9.57×10^{-7}	9.98×10^{-7}	8.98×10^{-7}
FGBK(p=2)	η_{exp}	0.05	0.30	0.15	0.15
	IT	39	406	137	163
	CPU	0.1859	0.0907	0.1735	0.1057
	RSE	8.27×10^{-7}	9.93×10^{-7}	9.44×10^{-7}	9.33×10^{-7}
FGBK(p=3)	η_{exp}	0.05	0.15	0.05	0.05
	IT	45	387	134	163
	CPU	0.2124	0.0945	0.1802	0.1137
	RSE	9.26×10^{-7}	9.53×10^{-7}	9.17×10^{-7}	8.02×10^{-7}



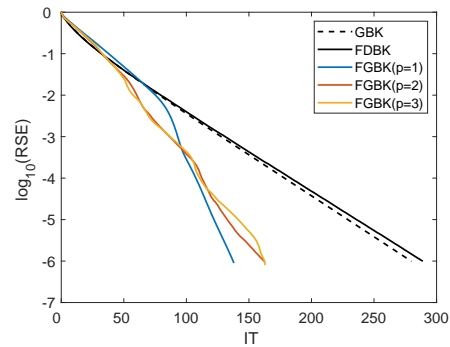
(a) stat96v5



(b) crew1



(c) bibd_17_8



(d) bibd_16_8

Fig. 1. Convergence curves for the matrices from SuiteSparse Matrix Collection.

From Figure 1, it is obviously seen that the fast greedy block Kaczmarz methods converge faster than the greedy block Kaczmarz method and the fast deterministic block Kaczmarz method, which confirms the numerical result in Table 3 and further shows the efficiency of the modified greedy row selection strategy.

4 Conclusions

A class of fast greedy block Kaczmarz methods is presented for solving large consistent linear systems. Theoretical analysis proves the convergence of the proposed methods and show that the upper bound of the convergence rate is related to the geometric properties of the coefficient matrix and its block submatrices. Numerical experiments further illustrate that the proposed methods are efficient and faster than the fast deterministic block Kaczmarz method.

Acknowledgements This work is supported by the National Natural Science Foundation of China (Grant No. 11971354).

References

- [1] S Kaczmarz. Angenäherte auflösung von systemen linearer gleichungen. *Bull. Int. Acad. Pol. Sic. Lett. A*, 35:355–357, 1937.
- [2] Avinash C Kak and Malcolm Slaney. *Principles of computerized tomographic imaging*. SIAM, Philadelphia, PA, 2001.
- [3] Gabor T Herman and Ran Davidi. Image reconstruction from a small number of projections. *Inverse problems*, 24(4):045011, 2008.
- [4] Deanna Needell, Nati Srebro, and Rachel Ward. Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm. *Mathematical Programming*, 155(1):549–573, 2016.
- [5] Damir Filipović, Kathrin Glau, Yuji Nakatsukasa, and Francesco Statti. Weighted monte carlo with least squares and randomized extended Kaczmarz for option pricing. *Swiss Finance Institute Research Paper*, (19–54), 2019.
- [6] Thomas Strohmer and Roman Vershynin. A randomized Kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2):262–278, 2009.
- [7] Zhong-Zhi Bai and Wen-Ting Wu. On greedy randomized Kaczmarz method for solving large sparse linear systems. *SIAM Journal on Scientific Computing*, 40(1):A592–A606, 2018.
- [8] Zhong-Zhi Bai and Wen-Ting Wu. On relaxed greedy randomized Kaczmarz methods for solving large sparse linear systems. *Applied Mathematics Letters*, 83:21–26, 2018.
- [9] Zhong-Zhi Bai and Wen-Ting Wu. On partially randomized extended Kaczmarz method for solving large sparse overdetermined inconsistent linear systems. *Linear Algebra and Its Applications*, 578(1):225–250, 2019.
- [10] Yi-Shu Du, Ken Hayami, Ning Zheng, Keiichi Morikuni, and Jun-Feng Yin. Kaczmarz-type inner-iteration preconditioned flexible gmres methods for consistent linear systems. *SIAM Journal on Scientific Computing*, 43(5):S345–S366, 2021.
- [11] Jun-Feng Yin, Nan Li and Ning Zheng. Restarted randomized surrounding methods for solving large linear equations. *Applied Mathematics Letters*, 133:108290, 2022.

- [12] Tommy Elfving. Block-iterative methods for consistent and inconsistent linear equations. *Numerische Mathematik*, 35(1):1–12, 1980.
- [13] Deanna Needell and Joel A Tropp. Paved with good intentions: analysis of a randomized block Kaczmarz method. *Linear Algebra and its Applications*, 441:199–221, 2014.
- [14] Yu-Qi Niu and Bing Zheng. A greedy block Kaczmarz algorithm for solving large-scale linear systems. *Applied Mathematics Letters*, 104:106294, 2020.
- [15] Ion Necoara. Faster randomized block kaczmarz algorithms. *SIAM Journal on Matrix Analysis and Applications*, 40(4):1425–1452, 2019.
- [16] Cun-Qiang Miao and Wen-Ting Wu. On greedy randomized average block Kaczmarz method for solving large linear systems. *Journal of Computational and Applied Mathematics*, 413:114372, 2022.
- [17] Jia-Qi Chen and Zheng-Da Huang. On a fast deterministic block Kaczmarz method for solving large-scale linear systems. *Numerical Algorithms*, 89(3):1007-1029, 2022.
- [18] Timothy A Davis and Yi-Fan Hu. The university of florida sparse matrix collection. *ACM Transactions on Mathematical Software*, 38(1):1–25, 2011.