The randomized circumcentered-reflection iteration method for solving consistent linear equations

Nan Li
Tongji University
Jun-Feng Yin (yinjf@tongji.edu.cn)
Tongji University

Research Article

Keywords: Circumcenter, Reflection transformation, Randomized iteration method, Convergence

Posted Date: July 22nd, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1839532/v1

The randomized circumcentered-reflection iteration method for solving consistent linear equations

Nan Li

School of Mathematical Sciences, Tongji University,
Shanghai, 200092, PR China.
Email:sunshinekiwili@tongji.edu.cn
and
Jun-Feng Yin*
School of Mathematical Sciences, Tongji University,
Shanghai, 200092, PR China.
Email:yinjf@tongji.edu.cn

July 18, 2022

Abstract

After randomly reflecting on two hyperplanes, a new iteration method is established by making use of the circumceter of the reflective points from the viewpoint of geometry. The linear combination could be non-convex when the angle between the hyperplances is small. Theoretical analysis show that the proposed method converges and the convergence rate in expectation is also addressed in detail. The relation between our method and block Kaczmarz method is well discussed. Numerical experiments further verify that the new algorithms is efficient, and outperform the existing randomized Kaczmarz methods and randomized reflection methods in terms of the number of iterations and CPU time, especially when the coefficient matrix has highly coherent rows.

Keywords. Circumcenter, Reflection transformation, Randomized iteration method, Convergence

1 Introduction

Consider the solution of linear equations

$$Ax = b, \quad A \in \mathbb{R}^{m\times n}, \quad b \in \mathbb{R}^m, \tag{1.1}$$

where A has full column rank, which usually comes from the practical applications such as image reconstruction [10], phase retrieval [25], option pricing [8], biological computation [9] and machine learning [16].

Kaczmarz method [14] is a well-known and popular single projection method for solving consistent linear systems. Denote a_i be the ith row of A and b_i be the ith entry of the right-hand side vector b for $i = 1, 2, \dots, m$. Given an initial guess x_0, the iterate scheme of classical Kaczmarz method can be described as

$$x_{k+1} = x_k + \frac{b_{i_k} - a_{i_k}^T x_k}{\|a_{i_k}\|_2^2} a_{i_k}, \tag{1.2}$$

*Corresponding author.

where i_k can be chosen from index of rows cyclically. Due to its simplicity and efficiency, Kaczmarz method was quickly applied into many engineering applications and contributed extremely competent software, especially in CT image processing.

Strohmer and Vershynin [24] proved the exponential convergence of randomized Kaczmarz method to the unique solution x_* in expectation by

$$\mathbb{E}\|x_{k+1} - x_*\|_2^2 \leq \left(1 - \frac{\lambda_{\min}(A^T A)}{\|A\|_F^2}\right)^k \|x_k - x_*\|_2^2. \tag{1.3}$$

where i_k is chosen with probability proportional to $\|a_i\|_2^2$ and $\lambda_{\min}(A)$ denote the minimum eigenvalue of A. Since then, various randomized Kaczmarz method are studied and improved by greedy strategies [2], sampling techniques [5], extrapolating acceleration [17], block version [12, 20], averaging [18] and other improvements [6, 13, 23].

In recent years, Kaczmarz-type methods projecting onto some special subspaces has been drawn attention to researchers. Needell and Ward [19] proposed a two-subspace frame for randomized Kaczmarz methods to look forward the solution. The Kaczmarz method with oblique projection were presented in [15], which could be regarded as a orthogonal projection based on two rows. Furthermore, its performance were enhanced by applying greedy strategies in [26]. Wu [27] extended the randomized two-subspace Kaczmarz method for inconsistent linear systems.

Cimmino method [4] is another famous and efficient solver for linear equations. It utilized the gravity of deterministic reflective points obtained by Householder transformations as the next approximation. Some improvements were followed up later in [1, 7, 11]. Recently, taking i_k with probability proportional to $\|a_i\|_2^2$, Steinerberger [22] proposed a surrounding method for nonsingular $A \in \mathbb{R}^{n\times n}$, which generated the sequence of reflective points $\{x_k\}_{k=1}^{\infty}$ by consistent reflections on hyperplanes randomly as follow

$$x_{k+1} = x_k + 2\frac{b_{i_k} - a_{i_k}^T x_k}{\|a_{i_k}\|_2^2} a_{i_k}, \quad i_k \in \{1, 2, \ldots, n\}. \tag{1.4}$$

Then, all reflective points were taken the average for approximation. Analysis of the convergence rate in expectation were provided as well.

$$\mathbb{E}\left\|x_* - \frac{1}{N}\sum_{k=1}^{N} x_k\right\| \leq \frac{1 + \|A\|_F \left\|A^{-1}\right\|}{\sqrt{N}} \|x_* - x_0\|, \tag{1.5}$$

where N is the number of reflective points including the initial guess. Yin, Li and Zheng [28] further accelerated the surrounding method by restart techniques. Shao [21] demonstrate an interesting phenomenon of odd and even reflections when the rows are selected cyclically.

Inspired by the idea of circumcenter [3], we propose a randomized circumcentered-reflection iteration method based on two different rows. It generates a sequence from the view of geometry and it is possible to make nonconvex linear combinations of reflective points. Theoretical analysis shows that the convergence rate in expectation of the proposed method under some probability rules and it is pointed that our method is dominant when angles of selected hyperplanes are small. The relation between our method and block Kaczmarz method is well discussed. Numerical experiments further verify the efficiency of the new algorithm and it outperforms the existing randomized Kaczmarz method and randomized reflection method in terms of the iteration steps and CPU time, especially when the coefficient matrix with highly coherent rows.

The organization of the rest paper is as follows. In Section 2, we first give some preliminary knowledge and then establish the randomized circumcentered-reflection iteration method. In Section 3, the convergence rate in expectation of the randomized circumcentered-reflection iteration method is analyzed and it is discovered the equivalence with randomized block Kaczmarz method. Numerical experiments, in Section 4, are presented to show the efficient of the proposed method especially when the rows of the matrix are nearly parallel. Finally, in Section 5, the conclusion is drawn and some future work is discussed.

2 The randomized circumcentered-reflection iteration method

In this section, we firstly give the fundamental and then establish the randomized circumcentered-reflection iteration method.

It is seen that the surrounding method could be accelerated by restart techniques, while the linear convex combination of reflective points leads that the next estimation always located in the interior of the n-dimensional polytope. When the angle between any two hyperplanes is small, the shrinking of the error $\|x_k - x_*\|$ will be slight, which results in the slow convergence rate.

Our motivation is to accelerate the convergence of the restarted surrounding method, especially when the coefficient matrix has highly coherent rows. Hence, we utilize the circumcenter of a triangle constructed by reflective points and take it as the next initial guess to restart reflections to improve the convergence rate. In each iteration, the next estimation belongs to the intersection of two randomly selected hyperplanes and the calculation for circumcenter is the main cost.

Before introducing the new method, we first briefly review the basic geometry properties which we will use later.

Lemma 2.1. *Assume that there are three non-collinear points A, B, C which forms an triangle ΔABC, P is the circumcenter of ΔABC, O is the origin so that $\overrightarrow{OA}, \overrightarrow{OB}, \overrightarrow{OC}$ represent three vectors. Then, the following identity exists*

$$\sin 2A \cdot \overrightarrow{PA} + \sin 2B \cdot \overrightarrow{PB} + \sin 2C \cdot \overrightarrow{PC} = \overrightarrow{0}. \tag{2.1}$$

And it also holds that

$$\overrightarrow{OP} = \frac{\sin 2A \cdot \overrightarrow{OA} + \sin 2B \cdot \overrightarrow{OB} + \sin 2C \cdot \overrightarrow{OC}}{\sin 2A + \sin 2B + \sin 2C} := \alpha\overrightarrow{OA} + \beta\overrightarrow{OB} + \gamma\overrightarrow{OC}. \tag{2.2}$$

If so, the coefficients α, β and γ can be computed by the following formulas

$$\begin{cases} \alpha = \dfrac{\sin 2A}{\sin 2A + \sin 2B + \sin 2C} = \dfrac{\cos A}{2\sin B \sin C} = \dfrac{|\overrightarrow{AC}| \cos A}{2|\overrightarrow{AB}|(1 - \cos^2 B)} \\ \beta = \dfrac{\sin 2B}{\sin 2A + \sin 2B + \sin 2C} = \dfrac{\cos B}{2\sin A \sin C} = \dfrac{|\overrightarrow{BC}| \cos B}{2|\overrightarrow{AB}|(1 - \cos^2 A)} \\ \gamma = \dfrac{\sin 2C}{\sin 2A + \sin 2B + \sin 2C} = \dfrac{\cos C}{2\sin A \sin B} = \dfrac{|\overrightarrow{BC}| \cos C}{2|\overrightarrow{AC}|(1 - \cos^2 A)} \end{cases} \tag{2.3}$$

and $\alpha + \beta + \gamma = 1$.

Proof. It is easy to deduce from the identity

$$\sin 2A + \sin 2B + \sin 2C = 4 \sin A \sin B \sin C$$

and the law of sines. ∎

Assume that $x_k = y_k^{(0)}$ is given, in every step of iteration, $y_k^{(1)}$ and $y_k^{(2)}$ is computed by reflecting $y_k^{(0)}$ on two hyperplanes randomly. Then, the circumcenter is taken as the approximate solution x_{k+1} in the next iteration as follow.

$$x_{k+1} = \alpha_k y_k^{(0)} + \beta_k y_k^{(1)} + \gamma_k y_k^{(2)}. \tag{2.4}$$

where α_k, β_k and γ_k are computed according to the formula (2.3). The sketch map of the randomized circumcentered-reflection iteration method is illustrated in Figure 1.

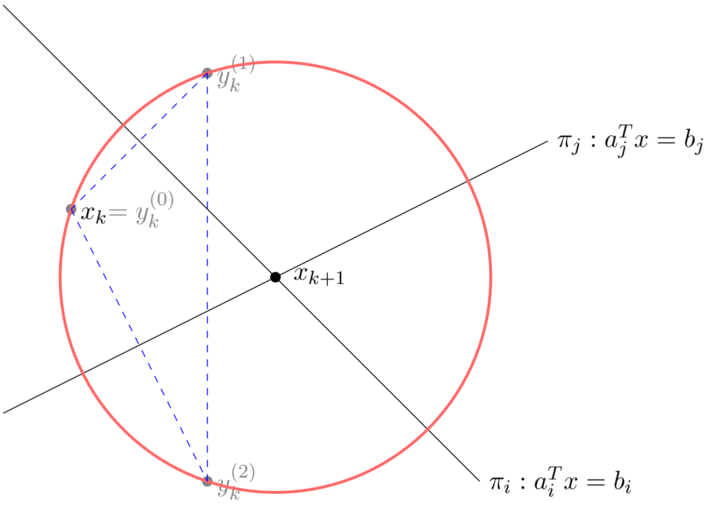

Fig. 1. The sketch map of the randomized circumcentered-reflection iteration method in k-th step

More precisely, the randomized circumcentered-reflection iteration method can be described in detail as follow.

Algorithm 1 Randomized circumcentered-reflection iteration method (RC)

Input: A, b, initial guess x_0, $y_0^{(0)} = x_0$.

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

2: Randomly choose two rows a_{i_k} and a_{j_k},

3: Compute the reflective points $\{y_k^{(1)}, y_k^{(2)}\}$, where

$$y_k^{(1)} = y_k^{(0)} + 2\frac{b_{i_k} - a_{i_k}^T y_k^{(0)}}{\|a_{i_k}\|_2^2} a_{i_k} \quad \text{and} \quad y_k^{(2)} = y_k^{(0)} + 2\frac{b_{j_k} - a_{j_k}^T y_k^{(1)}}{\|a_{j_k}\|_2^2} a_{j_k}.$$

4: Calculate the parameters

$$t_1 = \frac{\langle y_k^{(1)} - y^{(0)}, y_k^{(2)} - y_k^{(0)} \rangle}{|y_k^{(1)} - y_k^{(0)}| \cdot |y_k^{(2)} - y_k^{(0)}|}, \; t_2 = \frac{\langle y_k^{(0)} - y^{(1)}, y_k^{(2)} - y_k^{(1)} \rangle}{|y_k^{(1)} - y_k^{(0)}| \cdot |y_k^{(2)} - y_k^{(1)}|},$$

$$\alpha_k = \frac{|y_k^{(2)} - y_k^{(0)}| t_1}{2|y_k^{(1)} - y_k^{(0)}|(1 - t_2^2)}, \; \beta_k = \frac{|y_k^{(2)} - y_k^{(1)}| t_2}{2|y_k^{(1)} - y_k^{(0)}|(1 - t_1^2)}, \; \gamma_k = 1 - \beta_k - \alpha_k.$$

5: Compute the approximate solution: $x_{k+1} = \alpha_k y_k^{(0)} + \beta_k y_k^{(1)} + \gamma_k y_k^{(2)}$.

6: Let $y_{k+1}^{(0)} = x_{k+1}$.

7: **end for**

3 Convergence analysis

In this section, the theorems on the convergence performance of the randomized circumcentered-reflection iteration method are established and the relations between the new approach and the block Kaczmarz method are well discussed.

Theorem 3.1. *If a_{i_k} and a_{j_k} are linear independent, the sequence $\{x_k - x_*\}$ generated by the randomized circumcentered-reflection iteration method is monotonically decreasing for consistent linear systems.*

Proof. Assume the i_k and j_k rows are chosen at k-th step where $i_k, j_k \in \{1, 2, \ldots, m\}$. The scheme of circumcenters iteration is

$$\begin{aligned} x_{k+1} - x_* &= \alpha_k y_k^{(0)} + \beta_k y_k^{(1)} + \gamma_k y_k^{(2)} - x_* \\ &= \alpha_k x_k + \beta_k (x_k + 2\frac{b_{i_k} - a_{i_k}^T x_k}{\|a_{i_k}\|^2} a_{i_k}) + \gamma_k (x_k + 2\frac{b_{j_k} - a_{j_k}^T x_k}{\|a_{j_k}\|^2} a_{j_k}) - x_* \\ &= (\alpha_k + \beta_k + \gamma_k) x_k - x_* + 2\beta_k \frac{b_{i_k} - a_{i_k}^T x_k}{\|a_{i_k}\|^2} a_{i_k} + 2\gamma_k \frac{b_{j_k} - a_{j_k}^T x_k}{\|a_{j_k}\|^2} a_{j_k} \\ &= (I - 2\beta_k \frac{a_{i_k} a_{i_k}^T}{\|a_{i_k}\|^2} - 2\gamma_k \frac{a_{j_k} a_{j_k}^T}{\|a_{j_k}\|^2})(x_k - x_*) \\ &:= (I - S)(x_k - x_*), \end{aligned} \tag{3.1}$$

where $S = 2(\beta_k \frac{a_{i_k} a_{i_k}^T}{\|a_{i_k}\|^2} + \gamma_k \frac{a_{j_k} a_{j_k}^T}{\|a_{j_k}\|^2})$. And next, we need to show S is an idempotent operator.

Since

$$y_k^{(1)} - y_k^{(2)} = 2\left(\frac{b_{i_k} - a_{i_k}^T x_k}{\|a_{i_k}\|^2} a_{i_k} - \frac{b_{j_k} - a_{j_k}^T x_k}{\|a_{j_k}\|^2} a_{j_k}\right) = 2\left(\frac{a_{i_k} a_{i_k}^T}{\|a_{i_k}\|^2} - \frac{a_{j_k} a_{j_k}^T}{\|a_{j_k}\|^2}\right)(x_* - x_k), \tag{3.2}$$

we calculate the inner products

$$\langle a_{i_k}, y_k^{(1)} - y_k^{(2)} \rangle = 2(a_{i_k}^T - \frac{\langle a_{i_k}, a_{j_k} \rangle}{\|a_{j_k}\|^2} a_{j_k}^T)(x_* - x_k), \tag{3.3}$$

$$\langle a_{j_k}, y_k^{(1)} - y_k^{(2)} \rangle = 2(\frac{\langle a_{i_k}, a_{j_k} \rangle}{\|a_{i_k}\|^2} a_{i_k}^T - a_{j_k}^T)(x_* - x_k). \tag{3.4}$$

By substituting the formula (3.2)-(3.4) into the calculation, it is obtained that

$$\begin{aligned} S(x_* - x_k) &= 2\beta_k \frac{b_{i_k} - a_{i_k}^T x_k}{\|a_{i_k}\|^2} a_{i_k} + 2\gamma_k \frac{b_{j_k} - a_{j_k}^T x_k}{\|a_{j_k}\|^2} a_{j_k} \\ &= \frac{1}{\left(1 - \frac{\langle a_{i_k}, a_{j_k} \rangle^2}{\|a_{i_k}\|^2 \|a_{j_k}\|^2}\right)} \left(\frac{\langle a_{i_k}, y_k^{(1)} - y_k^{(2)} \rangle}{2\|a_{i_k}\|^2} a_{i_k} + \frac{\langle -a_{j_k}, y_k^{(1)} - y_k^{(2)} \rangle}{2\|a_{j_k}\|^2} a_{j_k} \right) \\ &= \frac{1}{\left(1 - \frac{\langle a_{i_k}, a_{j_k} \rangle^2}{\|a_{i_k}\|^2 \|a_{j_k}\|^2}\right)} \left[\frac{a_{i_k}}{\|a_{i_k}\|^2} (a_{i_k}^T - \frac{\langle a_{i_k}, a_{j_k} \rangle}{\|a_{j_k}\|^2} a_{j_k}^T) - \frac{a_{j_k}}{\|a_{j_k}\|^2} (\frac{\langle a_{i_k}, a_{j_k} \rangle}{\|a_{i_k}\|^2} a_{i_k}^T - a_{j_k}^T) \right] (x_* - x_k) \end{aligned} \tag{3.5}$$

Let $\tilde{S} = \frac{a_{i_k}}{\|a_{i_k}\|^2} (a_{i_k}^T - \frac{\langle a_{i_k}, a_{j_k} \rangle}{\|a_{j_k}\|^2} a_{j_k}^T) - \frac{a_{j_k}}{\|a_{j_k}\|^2} (\frac{\langle a_{i_k}, a_{j_k} \rangle}{\|a_{i_k}\|^2} a_{i_k}^T - a_{j_k}^T)$ then it holds that

$$(a_{i_k}^T - \frac{\langle a_{i_k}, a_{j_k} \rangle}{\|a_{j_k}\|^2} a_{j_k}^T) a_{j_k} = 0, \quad (\frac{\langle a_{i_k}, a_{j_k} \rangle}{\|a_{i_k}\|^2} a_{i_k}^T - a_{j_k}^T) a_{i_k} = 0,$$

and

$$(a_{i_k}^T - \frac{\langle a_{i_k}, a_{j_k} \rangle}{\|a_{j_k}\|^2} a_{j_k}^T) \frac{a_{i_k}}{\|a_{i_k}\|^2} = -(\frac{\langle a_{i_k}, a_{j_k} \rangle}{\|a_{i_k}\|^2} a_{i_k}^T - a_{j_k}^T) a_{j_k} = 1 - \frac{\langle a_{i_k}, a_{j_k} \rangle^2}{\|a_{i_k}\|^2 \|a_{j_k}\|^2}.$$

So, we can prove $S^2 = S$. In other word, it is verified that

$$x_{k+1} - x_k \perp x_{k+1} - x_*$$

i.e.

$$\|x_{k+1} - x_k\|^2 = \|x_k - x_*\|^2 - \|x_{k+1} - x_*\|^2. \tag{3.6}$$

From (3.6), we know the sequence $\{\|x_k - x_*\|\}_{k=1}^{\infty}$ is monotonically decreasing. ∎

In fact, $\|x_{k+1} - x_k\|^2 = \frac{|b_{i_k} - a_{i_k}^T x_k|^2}{\|a_{i_k}\|^2} \cdot \frac{1}{\sin^2\theta}$, where $\theta \in (0, \frac{\pi}{2})$ is the angle of two hyperplanes. However, it holds that $\|x_{k+1} - x_k\|^2 = \frac{|b_{i_k} - a_{i_k}^T x_k|^2}{\|a_{i_k}\|^2}$ in classical Kaczmarz mehtod. As we can see, the step is larger than that in original method at each iteration and the superiority is stand out especially when the angle is small.

Next, we analyze the convergence rates in expectation under two specific probability rules for our new method.

Theorem 3.2. *The randomized circumcentered-reflection iteration method converges in expectation to the exact solution of the consistent system $Ax = b$ with the rate*

(a)

$$\mathbb{E}\|x_k - x_*\|^2 \leq (1 - \frac{1}{m}\frac{\sigma_{\min}^2}{\max\{\sin\theta_i\}\|A\|_F^2})^k \|x_0 - x_*\|^2, \tag{3.7}$$

if rows are chosen uniformly at random.

(b)

$$\mathbb{E}\|x_k - x_*\|^2 \leq (1 - \frac{1}{\max\{\theta_i\}\|A\|_F} \inf_{x\neq 0} \frac{\|Ax\|_{\ell^{p+2}}^{p+2}}{\|Ax\|_{\ell^p}^p \|x\|})^k \|x_0 - x_*\|^2 \tag{3.8}$$

if rows are chosen according to $p_i = \frac{|r_i|^p}{\Sigma |r_i|^p}$.

Proof. When rows are chosen uniformly at random,

$$\begin{aligned}
\mathbb{E}\|x_{k+1} - x_*\|^2 &= \|x_k - x_*\|^2 - \mathbb{E}\|x_{k+1} - x_k\|^2 \\
&= \|x_k - x_*\|^2 - \frac{1}{m}\Sigma_{i=1, i\neq i_k, j_k}^m \frac{|b - Ax_k|_i^2}{\|a_i \sin\theta_i\|^2} \\
&\leq \|x_k - x_*\|^2 - \frac{1}{m}\frac{\sigma_{\min}^2}{\max\{\sin\theta_i\}\|A\|_F^2}\|x_k - x_*\|^2 \\
&\leq (1 - \frac{1}{m}\frac{\sigma_{\min}^2}{\max\{\sin\theta_i\}\|A\|_F})\|x_k - x_*\|^2,
\end{aligned}$$

where $\max\{\theta_i\}$ is the maximum angle between hyperplanes.

Similarly to the analysis in [22], when rows are chosen according to $p_i = \frac{|r_i|^p}{\Sigma |r_i|^p}$, it can be obtained that

$$\begin{aligned}
\mathbb{E}\|x_{k+1} - x_*\|^2 &= \|x_k - x_*\|^2 - \mathbb{E}\|x_{k+1} - x_k\|^2 \\
&= \|x_k - x_*\|^2 - \Sigma_{i=1}^m \frac{|r_i|^p}{\|A(x_k - x_*)\|_{\ell^p}^p} \frac{|r_i|^2}{\|a_i \sin\theta\|^2} \\
&\leq \|x_k - x_*\|^2 - \frac{1}{\max\{\theta_i\}} \frac{\|A(x_k - x_*)\|_{\ell^{p+2}}^{p+2}}{\|A(x_k - x_*)\|_{\ell^p}^p \|A\|_F^2 \|x_k - x_*\|} \|x_k - x_*\|^2 \\
&\leq (1 - \frac{1}{\max\{\theta_i\}\|A\|_F} \inf_{x\neq 0} \frac{\|Ax\|_{\ell^{p+2}}^{p+2}}{\|Ax\|_{\ell^p}^p \|x\|})\|x_k - x_*\|^2.
\end{aligned}$$

∎

In addition, we find the relation between the randomized circumcentered-reflection iteration method and block Kaczmarz method.

Theorem 3.3. *Assume rows a_{i_k} and a_{j_k} are linear independent, the iteration scheme of the randomized circumcentered-reflection iteration method is equivalent with that of block Kaczmarz method as follow:*

$$x_{k+1} = (I - A_\tau^\dagger A_\tau)x_k + A_\tau^\dagger b_\tau, \tag{3.9}$$

where $A_\tau = \begin{bmatrix} a_{i_k}^T \\ a_{j_k}^T \end{bmatrix} \in \mathbb{R}^{2\times n}, b_\tau = \begin{bmatrix} b_{i_k} \\ b_{j_k} \end{bmatrix} \in \mathbb{R}^2$ *if the same rows are selected.*

Proof. To prove the equivalence, it is to prove

$$S(x_k - x_*) = 2\beta_k \frac{b_{i_k} - a_{i_k}^T x_k}{\|a_{i_k}\|^2} a_{i_k} + 2\gamma_k \frac{b_{j_k} - a_{j_k}^T x_k}{\|a_{j_k}\|^2} a_{j_k} = A_\tau^\dagger A_\tau (x_k - x^*). \tag{3.10}$$

By calculation, it is easy to know

$$\det(A_\tau A_\tau^T) = \|a_1\|^2 \|a_2\|^2 - \langle a_1, a_2 \rangle^2. \tag{3.11}$$

Hence, we get

$$\begin{aligned} A_\tau^\dagger A_\tau &= \frac{1}{\det(A_\tau A_\tau^T)} \begin{bmatrix} a_1 & a_2 \end{bmatrix} \begin{bmatrix} \|a_2\|^2 & -(a_1, a_2) \\ -(a_1, a_2) & \|a_1\|^2 \end{bmatrix} \begin{bmatrix} a_1^T \\ a_2^T \end{bmatrix} \\ &= \frac{1}{\det(A_\tau A_\tau^T)} \left(\|a_2\|^2 a_1 a_1^T - (a_1, a_2) a_2 a_1^T - (a_1, a_1) a_1 a_2^T + \|a_1\|^2 a_2 a_2^T \right) \\ &= S. \end{aligned} \tag{3.12}$$

∎

We should remarkd that the randomized circumcentered-reflection iteration method is optimal in two-subspace Kaczmarz method, so that it reaches the same point with two-subspace Kaczmarz method in [19] and oblique Kaczmarz in [15] if the same rows selected in every iteration.

4 Numerical experiments

In this section, numerical experiments are presented to show the efficiency and superiority of the randomized circumcentered-reflection method (abbreviated as 'RC'), compared with randomized Kaczmarz-type methods (abbreviated as 'RK') and randomized restarted surrounding type methods in [28] (abbreviated as 'RS'), which restart every two reflections for fair.

In the experiments, all the methods start from the zero vector x_0 and terminate when the norm of relative error vector (denoted by 'ERR') is less than the tolerance, i.e.

$$\text{ERR} = \frac{\|x_k - x_*\|_2^2}{\|x_*\|_2^2} \le 10^{-6},$$

or achieve the maximal number of the iteration. The number of iteration steps (denoted by 'IT'), the elapsed CPU time in seconds (denoted by 'CPU') are compared.

Example 1. We test some Gaussian random matrices generated by $A = (1-c)randn(m, n) + c\hat{\mathbf{1}}$, where $\hat{\mathbf{1}}$ is a matrix whose elements are all one, $c \in [0, 1]$ is a parameter. The greater the parameter c is, the higher coherent rows the matrix has. Here, set the maximal number be 30000 and repeated 20 trials.

In Figure 2 and 3, the curves of the relative error with a shaded error bars versus the number of the iteration are plotted for the RK, RS, and RC methods for $A \in \mathbb{R}^{500\times 100}$, respectively. The solid line in the middle represents the averaging results.

It is seen from Figure 2 that the randomized circumcentered-reflection method required the least steps of iterations. Furthermore, it is observed that from Figure 3, the advantage of randomized circumcentered-reflection method will be more obvious when $c = 0.6$, that is, when the rows of the matrix is nearly parallel.

Then, the numerical results of CPU and iteration are listed in Table 1 when $n = 500$ and m varies from $2000, 4000, 6000, 8000$ to 10000 for $c = 0.6$.

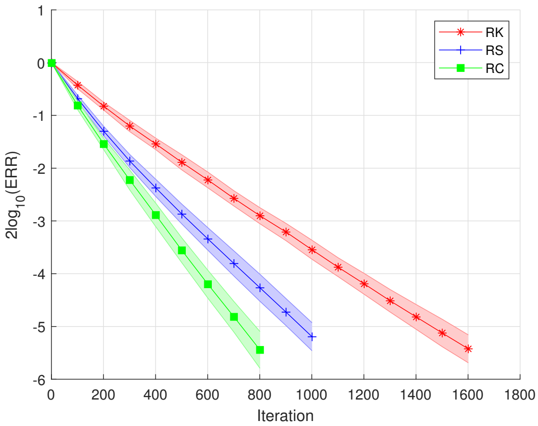

Fig. 2. $c = 0$

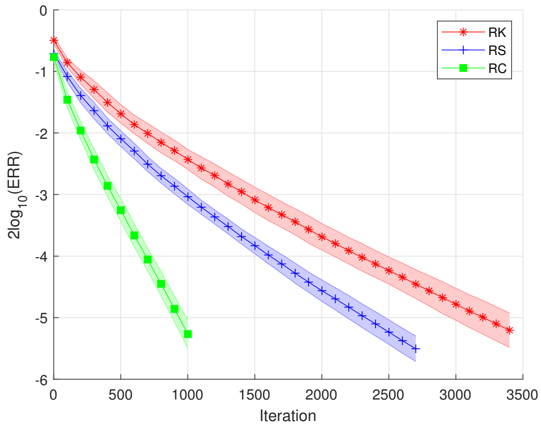

Fig. 3. $c = 0.6$

$c = 0.6$		2000×500	4000×500	6000×500	8000×500	10000×500
RK	IT	24531	17564	16011	15120	14732
	CPU	0.3017	0.2401	0.2158	0.2261	0.2124
	ERR	9.991e-07	9.989e-07	9.992e-07	9.986e-07	9.989e-07
RS	IT	17663	12351	10966	10407	9999
	CPU	0.4069	0.3234	0.2837	0.2923	0.2807
	ERR	9.995e-07	9.989e-07	9.992e-07	9.982e-07	9.990e-07
RC	IT	6908	5353	5059	4868	4811
	CPU	0.1964	0.1679	0.1536	0.1529	0.1500
	ERR	9.984e-07	9.980e-07	9.977e-07	9.974e-07	9.976e-07

Table 1. Numerical results for overdetermined systems when $c = 0.6$.

It is obvious from Table 1 that randomized circumcentered-reflection method perform the best in terms of both steps of iteration and the elapsed CPU time in case of matrices with highly coherent rows, which verifies our theoretical analysis.

Next, the number of iteration steps and the elapsed CPU time of three methods are listed in Table 2 when the parameters $m = 2000$, $n = 500$ and c varies from 0.1 to 0.9 with stepsize 0.2 respectively. It is further seen from Table 2 that the randomized circumcentered-reflection iteration method outperform the other two method in terms of the iteration steps and the elapsed CPU time. Particularly, the proposed method can still converge within a satisfying number of iteration when c is 0.7 and 0.9, while other two can not. This phenomenon confirms our explanation about the advantage of the randomized circumcentered-reflection iteration method.

c		0.1	0.3	0.5	0.7	0.9
RK	IT	9832	10802	16348	-	-
	CPU	0.1184	0.1386	0.1995	0.3635	0.3770
	ERR	9.982e-07	9.987e-07	9.991e-07	1.793e-05	2.542e-03
RS	IT	6466	7060	11493	-	-
	CPU	0.1523	0.1709	0.2630	0.7099	0.7050
	ERR	9.978e-07	9.977e-07	9.992e-07	1.806e-06	1.612e-03
RC	IT	4874	5298	6458	7192	5793
	CPU	0.1354	0.1419	0.1782	0.2045	0.1649
	ERR	9.980e-07	9.965e-07	9.983e-07	9.982e-07	9.983e-07

Table 2. Numerical results for overdetermined systems under different c

Example 2. The test matrices is taken from the SuiteSparse Matrix Collection. The fundamental information of the matrices are given in Table 3. The select rule of rows is uniformly random and the averaged performance after are reported after running 20 times.

name	Trefethen_20	WorldCities	abtaha2	abtaha1	cari	bibd_16_8
size	20×20	315×100	37932×331	14596×209	400×1200	120×12870
rank	20	100	331	209	400	120
density	39.5%	23.87%	1.09%	1.68%	31.83%	23.33%
cond	63.09	65.99	12.22	12.23	3.13	9.54

Table 3. Information of the matrices from the Matrix Market

The iteration steps, the elapsed CPU time and the relative error of randomized Kaczmarz method, randomized restarted surrounding method and the proposed method are listed respectively in Table 4. Further, it is found that the randomized circumcentered-reflection method can converge and keep the advantage in terms of iterations so that we can get similar conclusion as above. The CPU time of the randomized circumcenter method is comparable with randomized Kaczmarz method, since the new approach needs to cost extra time to choose two rows and calculate parameters of circumcenters.

Example 3. We studied the influence of different selected strategies for the new approach compared with randomized Kaczmarz method according to the probability rule $\frac{|r_i|^p}{\Sigma|r_i|^p}$, $i \in \{1, 2, \ldots, m\}$, where r_i is the i-th element of the residual vector, p is a positive integer. Here, set $p = 2, 4, 8$ and average the results after repeating 20 times.

In Figure 4, the curves of the relative error with shaded error bars versus the number of the

Method		Trefethen_20	WorldCities	abtaha2	abtaha1	*cari*	*bibd_16_8*
RK	IT	1082	22850	86578	58379	4522	2956
	CPU	0.0150	0.4955	4.9174	2.1023	0.3735	4.2892
	ERR	9.453e-07	9.993e-07	9.989e-07	9.982e-07	6.693e-07	9.723e-07
RS	IT	667	18023	63964	43238	2698	1924
	CPU	0.0170	0.7467	7.1447	3.0169	0.4357	5.2870
	ERR	9.841e-07	9.995e-07	9.987e-07	9.975e-07	8.024e-07	9.761e-07
RC	IT	394	9432	42608	28718	2260	1476
	CPU	0.0176	0.4125	4.7722	2.0764	0.3785	4.0172
	ERR	9.376e-07	9.988e-07	9.985e-07	9.975e-07	7.183e-07	9.927e-07

Table 4. Numerical results for the matrices from Matrix Market

iteration and CPU time are plotted for the randomized circumcentered-reflection method and randomized Kaczmarz method with different p respectively when $A \in \mathbb{R}^{500\times 100}$.

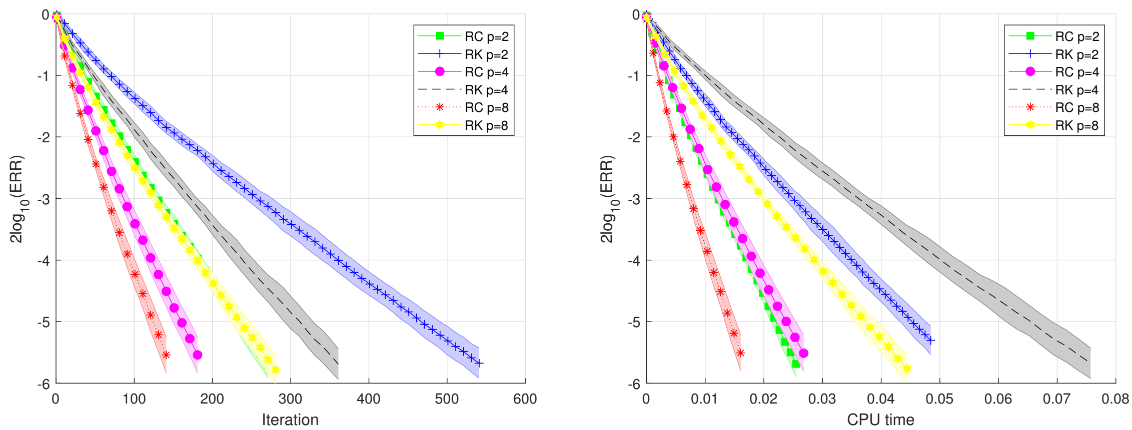

Fig. 4. Residual errors versus IT and CPU for different p

Compared with results in Example 1, it is seen from Figure 4 that the new rule could significantly accelerate the convergence rate of the proposed method for the same size of the matrix. It is indicated that different strategies of selecting rows indeed influence the performance of algorithms. More precisely, it is obvious that the iteration steps of both methods decrease as the value of p increases. Moreover, large p may cost more CPU time as shown in the right one. If the same p is chosen, the proposed methods outperform the corresponding Kaczmarz methods in terms of the number of iteration steps and the randomized circumcentered-reflection iteration method with $p = 8$ behaves best among them.

5 Conclusions

A randomized circumcentered-reflection iteration method is designed for solving consistent linear equations. Convergence rates of the proposed method in expectation are proved under two probability rules and the relation with block Kaczmarz method is revealed. Numerical experiments further verify that the randomized circumcentered-reflection iteration method can outperform existing methods especially when the coefficient matrix with highly coherent rows.

The random shuffle, sketch sampling and extrapolation strategies can be further considered to enhance the effect of our algorithm in the future.

Declarations

- Ethical Approval and Consent to participate: This manuscript is not applicable.
- Consent for publication: All authors read the manuscript and agreed to its publication.
- Human and Animal Ethics: This manuscript is not applicable.
- Availability of supporting data: All the data is open and available freely.
- Competing interests: The authors declare no competing interests.
- Funding: This work is sponsored and supported by the National Natural Science Foundation of China (Grant No. 11971354).
- Authors' contributions:

  - Nan Li
    School of Mathematical Sciences, Tongji University, Shanghai, 200092, PR China.
    Email: sunshinekiwili@tongji.edu.cn
    Writing - original draft, Formal analysis, Validation;
  - Jun-Feng Yin(Corresponding author)
    School of Mathematical Sciences, Tongji University, Shanghai, 200092, PR China.
    Email: yinjf@tongji.edu.cn
    Writing - review & editing, Supervision.

- Acknowledgments: The authors will be grateful to the comments and reviews which will greatly improve the quality of the paper.

References

[1] Rainer Ansorge. Connections between the cimmino-method and the kaczmarz-method for the solution of singular and regular systems of equations. *Computing*, 33(3):367–375, 1984.

[2] 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.

[3] Roger Behling, José Yunier Bello Cruz, and Luiz-Rafael Santos. Circumcentering the douglas–rachford method. *Numerical Algorithms*, 78(3):759–776, 2018.

[4] Gianfranco Cimmino. Cacolo approssimato per le soluzioni dei systemi di equazioni lineari. *La Ricerca Scientifica (Roma)*, 1:326–333, 1938.

[5] Jesus A De Loera, Jamie Haddock, and Deanna Needell. A sampling kaczmarz–motzkin algorithm for linear feasibility. *SIAM Journal on Scientific Computing*, 39(5):S66–S87, 2017.

[6] 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.

[7] Iain S Duff, Ronan Guivarch, Daniel Ruiz, and Mohamed Zenadi. The augmented block cimmino distributed method. *SIAM Journal on Scientific Computing*, 37(3):A1248–A1269, 2015.

[8] 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.

[9] Richard Gordon, Robert Bender, and Gabor T Herman. Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and x-ray photography. *Journal of theoretical Biology*, 29(3):471–481, 1970.

[10] Gabor T Herman and Ran Davidi. Image reconstruction from a small number of projections. *Inverse problems*, 24(4):045011, 2008.

[11] Alfredo N Iusem and Alvaro Rodolfo De Pierro. Convergence results for an accelerated nonlinear cimmino algorithm. *Numerische Mathematik*, 49(4):367–378, 1986.

[12] Xiang-Long Jiang, Ke Zhang, and Jun-Feng Yin. Randomized block kaczmarz methods with k-means clustering for solving large linear systems. *Journal of Computational and Applied Mathematics*, 403:113828, 2022.

[13] Yu-Tong Jiang, Gang Wu, and Long Jiang. A kaczmarz method with simple random sampling for solving large linear systems. *arXiv preprint arXiv:2011.14693*, 2020.

[14] S Karczmarz. Angenaherte auflosung von systemen linearer glei-chungen. *Bull. Int. Acad. Pol. Sic. Let., Cl. Sci. Math. Nat.*, pages 355–357, 1937.

[15] Wei-Guo Li, Qi-Feng Wang, Wen-Di Bao, and Li Liu. On kaczmarz method with oblique projection for solving large overdetermined linear systems. *arXiv preprint arXiv:2106.13368*, 2021.

[16] Jun-Hong Lin and Ding-Xuan Zhou. Learning theory of randomized kaczmarz algorithm. *The Journal of Machine Learning Research*, 16(1):3341–3365, 2015.

[17] Ji Liu and Stephen Wright. An accelerated randomized kaczmarz algorithm. *Mathematics of Computation*, 85(297):153–178, 2016.

[18] Jacob D Moorman, Thomas K Tu, Denali Molitor, and Deanna Needell. Randomized kaczmarz with averaging. *BIT Numerical Mathematics*, 61(1):337–359, 2021.

[19] Deanna Needell and Rachel Ward. Two-subspace projection method for coherent overdetermined systems. *Journal of Fourier Analysis and Applications*, 19(2):256–269, 2013.

[20] Deanna Needell, Ran Zhao, and Anastasios Zouzias. Randomized block kaczmarz method with projection for solving least squares. *Linear Algebra and its Applications*, 484:322–343, 2015.

[21] Chang-Peng Shao. A deterministic kaczmarz algorithm for solving linear systems. *arXiv preprint arXiv:2105.07736*, 2021.

[22] Stefan Steinerberger. Surrounding the solution of a linear system of equations from all sides. *Quart. Appl. Math.*, 79:419–429, 2021.

[23] Stefan Steinerberger. A weighted randomized kaczmarz method for solving linear systems. *Mathematics of Computation*, 90(332):2815–2826, 2021.

[24] Thomas Strohmer and Roman Vershynin. A randomized kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2):262–278, 2009.

[25] Yan-Shuo Tan and Roman Vershynin. Phase retrieval via randomized kaczmarz: theoretical guarantees. *Information and Inference: A Journal of the IMA*, 8(1):97–123, 2019.

[26] Fang Wang, Wei-Guo Li, Wen-Di Bao, and Li Liu. Greedy randomized and maximal weighted residual kaczmarz methods with oblique projection. *arXiv preprint arXiv:2106.13606*, 2021.

[27] Wen-Ting Wu. On two-subspace randomized extended kaczmarz method for solving large linear least-squares problems. *Numerical Algorithms*, 89(1):1–31, 2022.

[28] Jun-Feng Yin, Nan Li, and Ning Zheng. Restarted randomized surrounding methods for solving large linear equations. *Applied Mathematics Letters*, page 108290, 2022.