mathematics

Article

On the Convergence of the Randomized Block Kaczmarz Algorithm for Solving a Matrix Equation

Lili Xing, Wendi Bao and Weiguo Li *

College of Science, China University of Petroleum, Qingdao 266580, China; xinglily2010@upc.edu.cn (L.X.); baowd@upc.edu.cn (W.B.)
* Correspondence: liwg@upc.edu.cn

Abstract: A randomized block Kaczmarz method and a randomized extended block Kaczmarz method are proposed for solving the matrix equation $AXB = C$, where the matrices A and B may be full-rank or rank-deficient. These methods are iterative methods without matrix multiplication, and are especially suitable for solving large-scale matrix equations. It is theoretically proved that these methods converge to the solution or least-square solution of the matrix equation. The numerical results show that these methods are more efficient than the existing algorithms for high-dimensional matrix equations.

Keywords: matrix equation; randomized block Kaczmarz; randomized extended block Kaczmarz; convergence

MSC: 65F10; 65F45; 65H10

Citation: Xing, L.; Bao, W.; Li, W. On the Convergence of the Randomized Block Kaczmarz Algorithm for Solving a Matrix Equation. *Mathematics* **2023**, *11*, 4554. https://doi.org/10.3390/math11214554

Academic Editor: José Antonio Sanz

Received: 8 October 2023
Revised: 30 October 2023
Accepted: 3 November 2023
Published: 5 November 2023

1. Introduction

Consider the linear matrix equation

$$AXB = C, \tag{1}$$

where $A \in \mathbb{R}^{m\times p}$, $B \in \mathbb{R}^{q\times n}$ and $C \in \mathbb{R}^{m\times n}$. Such problems arise in many practical applications such as surface fitting in computer-aided geometric design (CAGD), signal and image processing, photogrammetry, etc.; see, for example, [1–4] and the large body of literature therein. If $AXB = C$ is consistent, $X^* = A^\dagger CB^\dagger$ is the minimum Frobenius norm solution. If $AXB = C$ is inconsistent, $X^* = A^\dagger CB^\dagger$ is the minimum Frobenius norm least-squares solution. When the matrices A and B are small and dense, direct methods based on QR fractions are attractive [5,6]. However, for large A and B matrices, iterative methods have attracted a lot of attention [7–11]. Recently, Du et al. proposed the randomized block coordinate descent (RBCD) method for solving the matrix least-squares problem $\min\limits_{X\in\mathbb{R}^{p\times q}} \|C - AXB\|_F$ without strong convexity assumption in [12]. This method requires that matrix B is a full row-rank matrix. Wu et al. [13] introduced two kinds of Kaczmarz-type methods to solve the consistent matrix equation $AXB = C$: relaxed greedy randomized Kaczmarz (ME-RGRK) and maximal weighted residual Kaczmarz (ME-MWRK). Although the row and column index selection strategy is time-consuming, the ideas of these two methods are suitable for solving large-scale consistent matrix equations.

In this paper, the randomized Kaczmarz method [14] and the randomized extended Kaczmarz method [15] are used to solve consistent and inconsistent matrix equation (1) with the product of the matrix and vector.

All the results in this paper hold in the complex field. But for the sake of simplicity, we only discuss them in terms of the real number field.

In this paper, we denote A^T, $A^\dagger$, $r(A)$, $R(A)$, $\|A\|_F = \sqrt{\text{trace}(A^TA)}$ and $\langle A, B\rangle_F = \text{trace}(A^TB)$ as the transpose, the Moore–Penrose generalized inverse, the rank of A, the column space of A, the Frobenius norm of A and the inner product of two matrices A and B, respectively. For an integer $n \geq 1$, let $[n] = \{1, 2, \ldots, n\}$. We use I to denote the identity matrix whose order is clear from the context. In addition, for a given matrix $G = (g_{ij}) \in \mathbb{R}^{m\times n}$, $G_{i,:}$, $G_{:,j}$, $\sigma_{\max}(G)$ and $\sigma_{\min}(G)$ are used to denote the ith row, the jth column, the maximum singular value and the smallest nonzero singular value of G, respectively. Let $\mathbb{E}_k$ denote the expected value conditional on the first k iterations, that is, $\mathbb{E}_k[\cdot] = \mathbb{E}[\cdot|i_0, j_0, i_1, j_1, \ldots, i_{k-1}, j_{k-1}]$, where i_s and $j_s(s = 0, 1, \ldots, k-1)$ are the row and the column chosen at the sth iteration. Let the conditional expectations with respect to the random row index be $\mathbb{E}_k^i[\cdot] = \mathbb{E}[\cdot|i_0, j_0, i_1, j_1, \ldots, i_{k-1}, j_{k-1}, j_k]$ and with respect to the random column index be $\mathbb{E}_k^j[\cdot] = \mathbb{E}[\cdot|i_0, j_0, i_1, j_1 \ldots, i_{k-1}, j_{k-1}, i_k]$. By the law of total expectation, it holds that $\mathbb{E}_k[\cdot] = \mathbb{E}_k^i[\mathbb{E}_k^j[\cdot]]$.

The organization of this paper is as follows. In Section 2, we will discuss the block Kaczmarz method (ME-RBK) for finding the minimal F-norm solution ($A^\dagger C B^\dagger$) of consistent matrix equation (1). In Section 3, we discuss the extended block Kaczmarz method (IME-REBK) for finding the minimal F-norm least-squares solution of matrix equation (1). In Section 4, some numerical examples are provided to illustrate the effectiveness of our new methods. Finally, some brief concluding remarks are described in Section 5.

2. The Randomized Block Kaczmarz Method for Consistent Equation

At the kth iteration, the Kaczmarz method selects randomized a row $i \in [m]$ of A and performs an orthogonal projection of the current estimate matrix $X^{(k)}$ onto the corresponding hyperplane $H_i = \{X \in \mathbb{R}^{p\times q} | A_{i,:}XB = C_{i,:}\}$, that is,

$$\min_{X\in\mathbb{R}^{p\times q}} \frac{1}{2}\|X - X^{(k)}\|_F^2 \;\; s.t. \;\; A_{i,:}XB = C_{i,:}. \tag{2}$$

The Lagrangian function of the conditional optimization problem (2) is

$$L(X, Y) = \frac{1}{2}\|X - X^{(k)}\|_F^2 + \langle Y, A_{i,:}XB - C_{i,:}\rangle, \tag{3}$$

where $Y \in \mathbb{R}^{1\times n}$ is a Lagrangian multiplier. Via the matrix differentiation, we obtain the gradient of $L(X, Y)$ and set $\nabla L(X, Y) = 0$ to find the stationary matrix:

$$\begin{cases} \nabla_X L(X, Y)\big|_{X^{(k+1)}} = X^{(k+1)} - X^{(k)} + A_{i,:}^T Y B^T = 0, \\ \nabla_Y L(X, Y)\big|_{X^{(k+1)}} = A_{i,:}X^{(k+1)}B - C_{i,:} = 0. \end{cases} \tag{4}$$

Using the first equation of (4), we have $X^{(k+1)} = X^{(k)} - A_{i,:}^T Y B^T$. Substituting this into the second equation of (4), we can obtain $Y = -\frac{1}{\|A_{i,:}\|_2^2}(C_{i,:} - A_{i,:}X^{(k)})(B^T B)^\dagger$. So, the projected randomized block Kacmarz (ME-PRBK) for solving $AXB = C$ iterates as

$$X^{(k+1)} = X^{(k)} + \frac{A_{i,:}^T}{\|A_{i,:}\|_2^2}(C_{i,:} - A_{i,:}X^{(k)}B)B^\dagger. \tag{5}$$

However, in practice, it is very expensive to calculate the pseudoinverse of large-scale matrices. Next, we generalize the average block Kaczmarz method [16] for solving linear equation to matrix equation.

At the kth step, we obtain the approximate solution $X^{(k+1)}$ by projecting the current estimate $X^{(k)}$ onto the hyperplane $H_{i,j} = \{X \in \mathbb{R}^{p\times q} | A_{i,:}X^{(k)}B_{:,j} = C_{i,j}\}$. Using the Lagrangian multiplier method, we can obtain the following Kaczmarz method for $AXB = C$:

$$X^{(k+1)} = X^{(k)} + \frac{A_{i,:}^T(C_{i,:} - A_{i,:}X^{(k)}B_{:,j})B_{:,j}^T}{\|A_{i,:}\|_2^2\|B_{:,j}\|_2^2}.$$

Inspired by the idea of the average block Kaczmaz algorithm for $Ax = b$, we consider the average block Kaczmaz method for $AXB = C$ with respect to B.

$$X^{(k+1)} = X^{(k)} + \lambda \frac{A_{i,:}^T}{\|A_{i,:}\|_2^2} \sum_{j=1}^{n} v_j^k \frac{(C_{i,:} - A_{i,:}X^{(k)}B_{:,j})B_{:,j}^T}{\|B_{:,j}\|_2^2},$$

where λ is stepsize and v_j^k are the weights that satisfy $v_j^k \geq 0$ and $\sum\limits_{j=1}^{n} v_j^k = 1$. If $v_j^k = \frac{\|B_{:,j}\|_2^2}{\|B\|_F^2}$, then

$$X^{(k+1)} = X^{(k)} + \frac{\lambda}{\|B\|_F^2} \frac{A_{i,:}^T}{\|A_{i,:}\|_2^2} (C_{i,:} - A_{i,:}X^{(k)}B)B^T.$$

Setting $\alpha = \frac{\lambda}{\|B\|_F^2} > 0$, we obtain the following randomized block Kaczmarz iteration:

$$X^{(k+1)} = X^{(k)} + \frac{\alpha}{\|A_{i,:}\|_2^2} A_{i,:}^T \Big((C_{i,:} - (A_{i,:}X^{(k)})B)B^T \Big), k = 0, 1, 2, \ldots, \tag{6}$$

where i is selected with probability $p_i = \frac{\|A_{i,:}\|_2^2}{\|A\|_F^2}$. We describe this method as Algorithm 1, which is called the ME-RBK algorithm.

Algorithm 1 Randomized Block Kaczmarz Method for $AXB = C$ (ME-RBK)

Input: $A \in R^{m \times p}, B \in R^{q \times n}, C \in R^{m \times n}, X^{(0)} = 0 \in R^{p \times q}$
1: **for** $k = 0, 1, 2, \ldots,$ **do**
2: Pick i with probability $p_i(A) = \frac{\|A_{i,:}\|_2^2}{\|A\|_F^2}$
3: Compute $X^{(k+1)} = X^{(k)} + \frac{\alpha}{\|A_{i,:}\|_2^2} A_{i,:}^T \Big((C_{i,:} - (A_{i,:}X^{(k)})B)B^T \Big)$
4: **end for**

We arrange the computational process of calculating $X^{(k+1)}$ in Table 1, which only costs $4q(n + p) + p + 1 - 2q$ flopping operations (flops) if the square of the row norm of A has been calculated in advance.

Table 1. The complexities of computing $X^{(k+1)}$ in ME-RBK.

$y_1 = A_{i,:}X^{(k)}$	$y_2 = C_{i,:} - y_1 B$	$y_3 = y_2 B^T$	$y_4^T = \frac{\alpha}{\|A_{i,:}\|_2^2} A_{i,:}^T$	$Y_1 = y_4^T y_3$	$X^{(k)} + Y_1$
$(2p - 1)q$	$(2q - 1)n + n$	$(2n - 1)q$	$1 + p$	pq	pq

Remark 1. *Note that the problem of finding a solution of $AXB = C$ can be posed as the following linear least-squares problem:*

$$\min_{X \in \mathbb{R}^{p \times q}} \frac{1}{2} \|AXB - C\|_F^2 = \min_{X \in \mathbb{R}^{p \times q}} \frac{1}{2} \sum_{i=1}^{m} \|A_{i,:}XB - C_{i,:}\|_2^2. \tag{7}$$

Define the component function

$$f_i(X) = \frac{1}{2} \|A_{i,:}XB - C_{i,:}\|_2^2,$$

then differentiate with X to obtain its gradient

$$\nabla f_i(X) = A_{i,:}^T (A_{i,:}XB - C_{i,:})B^T.$$

Therefore, the randomized block Kaczmarz method (6) is equivalent to one step of the stochastic gradient descent method [17] applied to (7) with stepsize $\frac{\alpha}{\|A_{i,:}\|_2^2}$.

First, we give the following lemma, whose proof can be found in [12].

Lemma 1 ([12])**.** *Let $A \in \mathbb{R}^{m\times p}$ and $B \in \mathbb{R}^{q\times n}$ be any nonzero matrix. Let*

$$\mathcal{M} = \{M \in R^{p\times q} \mid \exists Y \in \mathbb{R}^{m\times n} \ s.t.\ M = A^T Y B^T\}.$$

For any matrix $M \in \mathcal{M}$, it holds that

$$\|AMB\|_F^2 \geq \sigma_{\min}^2(A)\sigma_{\min}^2(B)\|M\|_F^2.$$

Remark 2. *$M \in \mathcal{M}$ means that $M_{:,j} \in R(A^T), j = 1,2,\ldots,q$ and $(M_{i,:})^T \in R(B)$, $i = 1,2,\ldots,p$. In fact, $\mathcal{M}$ is well defined because $0 \in \mathcal{M}$ and $A^\dagger C B^\dagger \in \mathcal{M}$.*

In the following theorem, with the idea of the RK method [14], we will prove that $X^{(k)}$ generated by Algorithm 1 converges to the least F-norm solution of $AXB = C$.

Theorem 1. *Assume $0 < \alpha < \frac{2}{\|B\|_2^2}$. If matrix equation (1) is consistent, the sequence $\{X^{(k)}\}$ generated by the ME-RBK method starting from the initial matrix $X^{(0)} \in \mathbb{R}^{p\times q}$, in which $X_{:,j}^{(0)} \in R(A^T), j = 1,2,\ldots,q$ and $(X_{i,:}^{(0)})^T \in R(B), i = 1,2,\ldots,p$, converges linearly to $A^\dagger C B^\dagger$ in mean square form. Moreover, the solution error in expectation for the iteration sequence $X^{(k)}$ obeys*

$$E\left[\left\|X^{(k)} - A^\dagger C B^\dagger\right\|_F^2\right] \leq \rho^k \left\|X^{(0)} - A^\dagger C B^\dagger\right\|_F^2, \tag{8}$$

where $\rho = 1 - \frac{2\alpha - \alpha^2\|B\|_2^2}{\|A\|_F^2}\sigma_{\min}^2(A)\sigma_{\min}^2(B)$, and $i \in [m]$ picked with probability $p_i(A) = \frac{\|A_{i,:}\|_2^2}{\|A\|_F^2}$.

Proof. For $k = 0,1,2,\ldots$, by (6) and $A_{i,:}A^\dagger C B^\dagger B = C_{i,:}$ (consistency), we have

$$\begin{aligned} X^{(k+1)} - A^\dagger C B^\dagger &= X^{(k)} + \frac{\alpha}{\|A_{i,:}\|_2^2} A_{i,:}^T (C_{i,:} - A_{i,:}X^{(k)}B)B^T - A^\dagger C B^\dagger \\ &= (X^{(k)} - A^\dagger C B^\dagger) - \frac{\alpha}{\|A_{i,:}\|_2^2} A_{i,:}^T A_{i,:}(X^{(k)} - A^\dagger C B^\dagger)BB^T, \end{aligned}$$

then

$$\begin{aligned} \|X^{(k+1)} - A^\dagger C B^\dagger\|_F^2 = &\left\|X^{(k)} - A^\dagger C B^\dagger\right\|_F^2 + \frac{\alpha^2}{\|A_{i,:}\|_2^4}\|A_{i,:}^T A_{i,:}(X^{(k)} - A^\dagger C B^\dagger)BB^T\|_F^2 \\ &- \frac{2\alpha}{\|A_{i,:}\|_2^2}\langle X^{(k)} - A^\dagger C B^\dagger, A_{i,:}^T A_{i,:}(X^{(k)} - A^\dagger C B^\dagger)BB^T\rangle_F. \end{aligned}$$

It follows from

$$\begin{aligned} &\frac{\alpha^2}{\|A_{i,:}\|_2^4}\|A_{i,:}^T A_{i,:}(X^{(k)} - A^\dagger C B^\dagger)BB^T\|_F^2 \\ &= \frac{\alpha^2}{\|A_{i,:}\|_2^4}\mathrm{trace}(BB^T(X^{(k)} - A^\dagger C B^\dagger)^T A_{i,:}^T A_{i,:} A_{i,:}^T A_{i,:}(X^{(k)} - A^\dagger C B^\dagger)BB^T) \\ &= \frac{\alpha^2}{\|A_{i,:}\|_2^2}\mathrm{trace}(BB^T(X^{(k)} - A^\dagger C B^\dagger)^T A_{i,:}^T A_{i,:}(X^{(k)} - A^\dagger C B^\dagger)BB^T) \\ &= \frac{\alpha^2}{\|A_{i,:}\|_2^2}\|A_{i,:}(X^{(k)} - A^\dagger C B^\dagger)BB^T\|_2^2 \text{ (by } \mathrm{trace}(uu^T) = \|u\|_2^2 \text{ for any vector } u) \\ &\leq \frac{\alpha^2\|B\|_2^2}{\|A_{i,:}\|_2^2}\|A_{i,:}(X^{(k)} - A^\dagger C B^\dagger)B\|_2^2 \text{ (by } \|u^T B^T\|_2 = \|Bu\|_2 \leq \|B\|_2\|u\|_2), \end{aligned}$$

and

$$\begin{aligned}
&\frac{2\alpha}{\|A_{i,:}\|_2^2}\langle X^{(k)} - A^\dagger CB^\dagger, A_{i,:}^T A_{i,:}(X^{(k)} - A^\dagger CB^\dagger)BB^T\rangle_F \\
&= \frac{2\alpha}{\|A_{i,:}\|_2^2}\mathrm{trace}(B^T(X^{(k)} - A^\dagger CB^\dagger)^T A_{i,:}^T A_{i,:}(X^{(k)} - A^\dagger CB^\dagger)B) \\
&= \frac{2\alpha}{\|A_{i,:}\|_2^2}\|A_{i,:}(X^{(k)} - A^\dagger CB^\dagger)B\|_2^2
\end{aligned}$$

that

$$\|X^{(k+1)} - A^\dagger CB^\dagger\|_F^2 \le \left\|X^{(k)} - A^\dagger CB^\dagger\right\|_F^2 - \frac{2\alpha - \alpha^2\|B\|_2^2}{\|A_{i,:}\|_2^2}\|A_{i,:}(X^{(k)} - A^\dagger CB^\dagger)B\|_2^2.$$

By taking the conditional expectation, we have

$$\begin{aligned}
\mathbb{E}_k\left[\left\|X^{(k+1)} - A^\dagger CB^\dagger\right\|_F^2\right] \le& \left\|X^{(k)} - A^\dagger CB^\dagger\right\|_F^2 \\
&- \mathbb{E}_k\left[\frac{2\alpha - \alpha^2\|B\|_2^2}{\|A_{i,:}\|_2^2}\|A_{i,:}(X^{(k)} - A^\dagger CB^\dagger)B\|_2^2\right] \\
=& \left\|X^{(k)} - A^\dagger CB^\dagger\right\|_F^2 - \frac{2\alpha - \alpha^2\|B\|_2^2}{\|A\|_F^2}\|A(X^{(k)} - A^\dagger CB^\dagger)B\|_F^2.
\end{aligned}$$

From $X^{(0)} \in \mathcal{M}$ and $A^\dagger CB^\dagger \in \mathcal{M}$, we have $X^{(0)} - A^\dagger CB^\dagger \in \mathcal{M}$. Noting $A_{i,:}^T = A^T I_{:,i}$, it is easy to show that $X^{(k)} - A^\dagger CB^\dagger \in \mathcal{M}$ through induction. Then, from Lemma 1 and $0 < \alpha < \frac{2}{\|B\|_2^2}$, we can obtain

$$\begin{aligned}
\mathbb{E}_k\left[\left\|X^{(k+1)} - A^\dagger CB^\dagger\right\|_F^2\right] \le& \left\|X^{(k)} - A^\dagger CB^\dagger\right\|_F^2 - \frac{2\alpha - \alpha^2\|B\|_2^2}{\|A\|_F^2} \\
&\cdot \sigma_{\min}^2(A)\sigma_{\min}^2(B)\left\|X^{(k)} - A^\dagger CB^\dagger\right\|_F^2 \\
=& \left(1 - \frac{2\alpha - \alpha^2\|B\|_2^2}{\|A\|_F^2}\sigma_{\min}^2(A)\sigma_{\min}^2(B)\right)\left\|X^{(k)} - A^\dagger CB^\dagger\right\|_F^2. \quad (9)
\end{aligned}$$

Finally, from (9) and induction on the iteration index k, we obtain the estimate (8). □

Remark 3. *Using a similar approach to that used in the proof of Theorem 1, we can prove that the iterate $X^{(k)}$ generated by ME-PRBK (5) satisfies the following estimate:*

$$\mathbb{E}\left[\left\|X^{(k)} - A^\dagger CB^\dagger\right\|_F^2\right] \le \hat{\rho}^k\left\|X^{(0)} - A^\dagger CB^\dagger\right\|_F^2,$$

where $\hat{\rho} = 1 - \frac{\sigma_{\min}^2(A)\sigma_{\min}^2(B)}{\|A\|_F^2\|B\|_2^2}$. The convergence factor of GRK in [18] is $\rho_{GRK} = 1 - \frac{\sigma_{\min}^2(A)\sigma_{\min}^2(B)}{\|A\|_F^2\|B\|_F^2}$. It is obvious that

$$\rho_{GRK} > \min_{0<\alpha<\frac{2}{\|B\|_2^2}} \rho = 1 - \frac{\sigma_{\min}^2(A)\sigma_{\min}^2(B)}{\|A\|_F^2\|B\|_2^2} = \hat{\rho}$$

and $\rho < \rho_{GRK}$ when $\frac{1-\sqrt{1-\frac{\|B\|_2^2}{\|B\|_F^2}}}{\|B\|_2^2} < \alpha < \frac{1+\sqrt{1-\frac{\|B\|_2^2}{\|B\|_F^2}}}{\|B\|_2^2}$. This means that the convergence factor of ME-PRBK is the smallest and the factor of ME-RBK can be smaller than that of GRK when α is properly selected.

3. The Randomized Extended Block Kaczmarz Method for Inconsistent Equation

In [15,19,20], the authors proved that the Kaczmarz method does not converge to the least-squares solution of $AX = b$ when $AX = b$ is inconsistent. Analogously, if the matrix equation (1) is inconsistent, the above ME-PRBK method dose not converge to $A^\dagger CB^\dagger$. The following theorem gives the error bound of the inconsistent matrix equation.

Theorem 2. *Assume that the consistent equation $AXB = C$ has a solution $X^* = A^\dagger CB^\dagger$. Let $\hat{X}^{(k)}$ denote the kth iterate of the ME-PRBK method applied to the inconsistent equation $AXB = C + W$ for any $W \in \mathbb{R}^{m\times n}$ starting from the initial matrix $X^{(0)} \in R^{p\times q}$, in which $X^{(0)}_{:,j} \in R(A^T), j = 1,2,\dots,q$ and $(X^{(0)}_{i,:})^T \in R(B), i = 1,2,\dots,p$. In exact arithmetic, it follows that*

$$\mathbb{E}\left[\left\|\hat{X}^{(k)} - A^\dagger CB^\dagger\right\|_F^2\right] \le \hat{\rho}^k\left\|X^{(0)} - A^\dagger CB^\dagger\right\|_F^2 + \frac{1-\hat{\rho}^k}{1-\hat{\rho}}\frac{\|WB^\dagger\|_F^2}{\|A\|_F^2}, \tag{10}$$

Proof. Set $H_i = \{X|A_iXB = C_i\}$, $\hat{H}_i = \{X|A_iXB = C_i + W_i\}$. Let Y denote the iterate of the PRBK method applied to the consistent equation $AXB = C$ at the kth step, that is,

$$Y = \hat{X}^{(k)} + \frac{A_{i,:}^T}{\|A_{i,:}\|_2^2}(C_{i,:} - A_{i,:}\hat{X}^{(k)}B)B^\dagger.$$

It follows from

$$\begin{aligned}
\langle Y - A^\dagger CB^\dagger, \hat{X}^{(k+1)} - A^\dagger CB^\dagger\rangle_F &= \langle Y - A^\dagger CB^\dagger, \frac{A_{i,:}^T}{\|A_{i,:}\|_2^2}W_iB^\dagger\rangle_F \\
&= \mathrm{trace}\left((B^\dagger)^T W_i^T \frac{A_{i,:}}{\|A_{i,:}\|_2^2}(Y - A^\dagger CB^\dagger)\right) \\
&= \mathrm{trace}\left(W_i^T \frac{A_{i,:}Y - A_{i,:}A^\dagger CB^\dagger}{\|A_{i,:}\|_2^2}(B^\dagger)^T\right) \\
&= \mathrm{trace}\left(W_i^T \frac{A_{i,:}Y - A_{i,:}A^\dagger CB^\dagger}{\|A_{i,:}\|_2^2}B(B^TB)^\dagger\right) \\
&\quad (\text{by } (B^\dagger)^T = B(B^TB)^\dagger) \\
&= \mathrm{trace}\left(W_i^T \frac{A_{i,:}YB - A_{i,:}A^\dagger CB^\dagger B}{\|A_{i,:}\|_2^2}(B^TB)^\dagger\right) = 0
\end{aligned}$$

and

$$\left\|\hat{X}^{(k+1)} - Y\right\|_F^2 = \frac{\left\|A_{i,:}^T W_i B^\dagger\right\|_F^2}{\|A_{i,:}\|_2^4} = \frac{1}{\|A_{i,:}\|_2^4}\mathrm{trace}\Big((B^\dagger)^T W_i^T A_{i,:}A_{i,:}^T W_i B^\dagger\Big) = \frac{\|W_iB^\dagger\|_2^2}{\|A_{i,:}\|_2^2}$$

that

$$\begin{aligned}
\left\|\hat{X}^{(k+1)} - A^\dagger CB^\dagger\right\|_F^2 &= \left\|Y - A^\dagger CB^\dagger\right\|_F^2 + \left\|\hat{X}^{(k+1)} - Y\right\|_F^2 \\
&= \left\|Y - A^\dagger CB^\dagger\right\|_F^2 + \frac{\|W_iB^\dagger\|_2^2}{\|A_{i,:}\|_2^2}.
\end{aligned} \tag{11}$$

By taking the conditional expectation on both sides of (11), we can obtain

$$\mathbb{E}_k\left[\left\|\hat{X}^{(k+1)} - A^{\dagger}CB^{\dagger}\right\|_F^2\right] = \mathbb{E}_k\left[\left\|Y - A^{\dagger}CB^{\dagger}\right\|_F^2\right] + \mathbb{E}_k\left[\frac{\left\|W_iB^{\dagger}\right\|_2^2}{\|A_{i,:}\|_2^2}\right]$$
$$\leq \hat{\rho}\left\|\hat{X}^{(k)} - A^{\dagger}CB^{\dagger}\right\|_F^2 + \frac{\left\|WB^{\dagger}\right\|_F^2}{\|A\|_F^2}$$

The inequality is obtained using Remark 3. Applying this recursive relation iteratively, we have

$$\begin{aligned}
\mathbb{E}\left[\left\|\hat{X}^{(k+1)} - A^{\dagger}CB^{\dagger}\right\|_F^2\right] &\leq \hat{\rho}\mathbb{E}\left[\left\|\hat{X}^{(k)} - A^{\dagger}CB^{\dagger}\right\|_F^2\right] + \frac{\left\|WB^{\dagger}\right\|_F^2}{\|A\|_F^2} \\
&\leq \hat{\rho}^2\mathbb{E}\left[\left\|\hat{X}^{(k-1)} - A^{\dagger}CB^{\dagger}\right\|_F^2\right] + (\hat{\rho}+1)\frac{\left\|WB^{\dagger}\right\|_F^2}{\|A\|_F^2} \\
&\leq \cdots \\
&\leq \hat{\rho}^{k+1}\left\|\hat{X}^{(0)} - A^{\dagger}CB^{\dagger}\right\|_F^2 + (\hat{\rho}^k + \cdots + \hat{\rho} + 1)\frac{\left\|WB^{\dagger}\right\|_F^2}{\|A\|_F^2} \\
&= \hat{\rho}^{k+1}\left\|X^{(0)} - A^{\dagger}CB^{\dagger}\right\|_F^2 + \frac{1-\hat{\rho}^{k+1}}{1-\hat{\rho}}\frac{\left\|WB^{\dagger}\right\|_F^2}{\|A\|_F^2}.
\end{aligned}$$

This completes the proof. □

Next, we use the idea of the randomized extended Kaczmarz method (see [20–22] for details) to solve the least-squares solution of the inconsistent Equation (1). At each iteration, $Z^{(k)}$ is the kth iterate of ME-RBK applied to $A^TZB^T = 0$ with the initial guess $Z^{(0)}$, and $X^{(k)}$ is the one-step ME-RBK update for $AXB = C - Z^{(k)}$. We can obtain the following randomized extended block Kaczmarz iteration:

$$\begin{cases} Z^{(k+1)} = Z^{(k)} - \frac{\alpha}{\|A_{:,j}\|_2^2}A_{:,j}(((A_{:,j}^TZ^{(k)})B^T)B), \\ X^{(k+1)} = X^{(k)} + \frac{\alpha}{\|A_{i,:}\|_2^2}A_{i,:}^T\left((C_{i,:} - Z_{i,:}^{(k+1)} - (A_{i,:}X^{(k)})B)B^T\right), \end{cases} \tag{12}$$

where $\alpha > 0$ is the step size, and i and j are selected with probability $p_i = \frac{\|A_{i,:}\|_2^2}{\|A\|_F^2}$ and $\hat{p}_j(A) = \frac{\|A_{:,j}\|_2^2}{\|A\|_F^2}$, respectively. The cost of each iteration of this method is $4n(q+m) + m + 1 - 2n - q$ for updating $Z^{(k)}$ and $(4q+1)(n+p) + 1 - 2q$ for updating $X^{(k)}$ if the square of the row norm and the column norm of A have been calculated in advance. We describe this method as Algorithm 2, which is called the ME-REBK algorithm.

Algorithm 2 Randomized Extended Block Kaczmarz Method for $AXB = C$ (ME-REBK)

Input: $A \in R^{m\times p}$, $B \in R^{q\times n}$, $C \in R^{m\times n}$, $X^{(0)} = 0 \in R^{p\times q}$, $Z^{(0)} = C$
1: **for** $k = 0, 1, 2, \ldots,$ **do**
2: Pick j with probability $\hat{p}_j(A) = \frac{\|A_{:,j}\|_2^2}{\|A\|_F^2}$
3: Compute $Z^{(k+1)} = Z^{(k)} - \frac{\alpha}{\|A_{:,j}\|_2^2}A_{:,j}(((A_{:,j}^TZ^{(k)})B^T)B)$
4: Pick i with probability $p_i(A) = \frac{\|A_{i,:}\|_2^2}{\|A\|_F^2}$
5: Compute $X^{(k+1)} = X^{(k)} + \frac{\alpha}{\|A_{i,:}\|_2^2}A_{i,:}^T\left((C_{i,:} - Z_{i,:}^{(k+1)} - (A_{i,:}X^{(k)})B)B^T\right)$
6: **end for**

Theorem 3. *Assume $0 < \alpha < \frac{2}{\|B\|_2^2}$. Let $\{Z^{(k)}\}$ denote the kth iteration of ME-RBK applied to $A^T Z B^T = 0$ starting from the initial matrix $Z^{(0)} \in R^{m\times n}$, in which $Z^{(0)}_{:,j} \in C_{:,j} + R(A)$, $j = 1, 2, \ldots, n$ and $(Z^{(0)}_{i,:})^T \in (C_{i,:})^T + R(B^T)$, $i = 1, 2, \ldots, m$. Then, $\{Z^{(k)}\}$ converges linearly to $C - AA^\dagger CB^\dagger B$ in mean square form, and the solution error in expectation for the iteration sequence $X^{(k)}$ obeys*

$$E\left[\left\|Z^{(k)} - (C - AA^\dagger CB^\dagger B)\right\|_F^2\right] \le \rho^k \left\|Z^{(0)} - (C - AA^\dagger CB^\dagger B)\right\|_F^2, \tag{13}$$

where the jth column of A is selected with probability $\hat{p}_j(A) = \frac{\|A_{:,j}\|_2^2}{\|A\|_F^2}$.

Proof. In Theorem 1, replacing A with A^T, B with B^T and C with 0, we can prove Theorem 3 based on the result of Theorem 1. For the sake of conciseness, we omit the proof process. □

Theorem 4. *Assume $0 < \alpha < \frac{2}{\|B\|_2^2}$. The sequence $\{X^{(k)}\}$ is generated using the ME-REBK method for $AXB = C$, starting from the initial matrix $X^{(0)} \in R^{p\times n}$ and $Z^{(0)} \in R^{m\times n}$, where $X^{(0)}_{:,j} \in R(A^T)$, $j = 1, 2, \ldots, q$, $(X^{(0)}_{i,:})^T \in R(B)$, $i = 1, 2, \ldots, p$ $Z^{(0)}_{:,j} \in C_{:,j} + R(A)$, $j = 1, 2, \ldots, n$ and $(Z^{(0)}_{i,:})^T \in (C_{i,:})^T + R(B^T)$, $i = 1, 2, \ldots, m$. For any $\varepsilon > 0$, it holds that*

$$\begin{aligned} E\left[\left\|X^{(k)} - A^\dagger CB^\dagger\right\|_F^2\right] \le & \frac{(1+\varepsilon)^{k+1} - (1+\varepsilon)}{\varepsilon^2} \frac{\alpha^2 \|B\|_2^2 \rho^k}{\|A\|_F^2} \left\|Z^{(0)} - (C - AA^\dagger CB^\dagger B)\right\|_F^2 \\ & + (1+\varepsilon)^k \rho^k \left\|X^{(0)} - A^\dagger CB^\dagger\right\|_F^2, \end{aligned} \tag{14}$$

where $i \in [m]$, $j \in [p]$ are picked with probability $p_i(A) = \frac{\|A_{i,:}\|_2^2}{\|A\|_F^2}$ and $\hat{p}_j(A) = \frac{\|A_{:,j}\|_2^2}{\|A\|_F^2}$, respectively.

Proof. Let $X^{(k)}$ denote the kth iteration of the ME-REBK method for $AXB = C$, and $\tilde{X}^{(k+1)}$ be the one-step Kaczmarz update for the matrix equation $AXB = AA^\dagger CB^\dagger B$ from $X^{(k)}$, i.e.,

$$\tilde{X}^{(k+1)} = X^{(k)} + \frac{\alpha}{\|A_{i,:}\|_2^2} A_{i,:}^T (A_{i,:} A^\dagger CB^\dagger B - A_{i,:} X^{(k)} B) B^T.$$

We have

$$\tilde{X}^{(k+1)} - A^\dagger CB^\dagger = X^{(k)} - A^\dagger CB^\dagger - \frac{\alpha}{\|A_{i,:}\|_2^2} A_{i,:}^T A_{i,:} (X^{(k)} - A^\dagger CB^\dagger) BB^T$$

and

$$X^{(k+1)} - \tilde{X}^{(k+1)} = \frac{\alpha}{\|A_{i,:}\|_2^2} A_{i,:}^T (C_{i,:} - Z_{i,:}^{(k+1)} - A_{i,:} A^\dagger CB^\dagger B) B^T.$$

For any $\varepsilon > 0$, via triangle inequality and Young's inequality, we can obtain

$$\begin{aligned} \left\|X^{(k+1)} - A^\dagger CB^\dagger\right\|_F^2 = & \left\|(X^{(k+1)} - \tilde{X}^{(k+1)}) + (\tilde{X}^{(k+1)} - A^\dagger CB^\dagger)\right\|_F^2 \\ \le & (\left\|X^{(k+1)} - \tilde{X}^{(k+1)}\right\|_F + \left\|\tilde{X}^{(k+1)} - A^\dagger CB^\dagger\right\|_F)^2 \\ \le & \left\|X^{(k+1)} - \tilde{X}^{(k+1)}\right\|_F^2 + \left\|\tilde{X}^{(k+1)} - A^\dagger CB^\dagger\right\|_F^2 \\ & + 2\left\|X^{(k+1)} - \tilde{X}^{(k+1)}\right\|_F \left\|\tilde{X}^{(k+1)} - A^\dagger CB^\dagger\right\|_F \\ \le & (1 + \frac{1}{\varepsilon})\left\|X^{(k+1)} - \tilde{X}^{(k+1)}\right\|_F^2 + (1+\varepsilon)\left\|\tilde{X}^{(k+1)} - A^\dagger CB^\dagger\right\|_F^2. \end{aligned} \tag{15}$$

By taking the conditional expectation on the both sides of (15), we have

$$\mathbb{E}_k\left[\left\|X^{(k+1)}-A^{\dagger}CB^{\dagger}\right\|_F^2\right] \leq (1+\frac{1}{\varepsilon})\mathbb{E}_k\left[\left\|X^{(k+1)}-\tilde{X}^{(k+1)}\right\|_F^2\right] + (1+\varepsilon)\mathbb{E}_k\left[\left\|\tilde{X}^{(k+1)}-A^{\dagger}CB^{\dagger}\right\|_F^2\right]. \tag{16}$$

It follows from

$$\left\|X^{(k+1)}-\tilde{X}^{(k+1)}\right\|_F^2 = \left\|\frac{\alpha}{\|A_{i,:}\|_2^2}A_{i,:}^T(C_{i,:}-Z_{i,:}^{(k+1)}-A_{i,:}A^{\dagger}CB^{\dagger}B)B^T\right\|_F^2$$
$$= \frac{\alpha^2}{\|A_{i,:}\|_2^2}\operatorname{trace}\left(B(C_{i,:}-Z_{i,:}^{(k+1)}-A_{i,:}A^{\dagger}CB^{\dagger}B)^T(C_{i,:}-Z_{i,:}^{(k+1)}-A_{i,:}A^{\dagger}CB^{\dagger}B)B^T\right)$$
$$\leq \frac{\alpha^2\|B\|_2^2}{\|A_{i,:}\|_2^2}\left\|C_{i,:}-Z_{i,:}^{(k+1)}-A_{i,:}A^{\dagger}CB^{\dagger}B\right\|_2^2$$

that

$$\begin{aligned}\mathbb{E}_k\left[\left\|X^{(k+1)}-\tilde{X}^{(k+1)}\right\|_F^2\right] &\leq \alpha^2\|B\|_2^2\mathbb{E}_k^j\mathbb{E}_k^i\left[\frac{\left\|C_{i,:}-Z_{i,:}^{(k+1)}-A_{i,:}A^{\dagger}CB^{\dagger}B\right\|_2^2}{\|A_{i,:}\|_2^2}\right]\\ &= \alpha^2\|B\|_2^2\mathbb{E}_k^j\left[\frac{1}{\|A\|_F^2}\sum_{i=1}^{m}\left\|C_{i,:}-Z_{i,:}^{(k+1)}-A_{i,:}A^{\dagger}CB^{\dagger}B\right\|_2^2\right]\\ &= \frac{\alpha^2\|B\|_2^2}{\|A\|_F^2}\mathbb{E}_k\left[\left\|Z^{(k+1)}-(C-AA^{\dagger}CB^{\dagger}B)\right\|_F^2\right].\end{aligned}$$

By Theorem 3, it yields

$$\begin{aligned}\mathbb{E}\left[\left\|X^{(k+1)}-\tilde{X}^{(k+1)}\right\|_F^2\right] &\leq \frac{\alpha^2\|B\|_2^2}{\|A\|_F^2}\mathbb{E}\left[\left\|Z^{(k+1)}-(C-AA^{\dagger}CB^{\dagger}B)\right\|_F^2\right]\\ &\leq \frac{\alpha^2\|B\|_2^2}{\|A\|_F^2}\rho^{k+1}\left\|Z^{(0)}-(C-AA^{\dagger}CB^{\dagger}B)\right\|_F^2.\end{aligned} \tag{17}$$

From $X^{(0)}-A^{\dagger}CB^{\dagger}\in\mathcal{M}$, we have $X^{(k)}-A^{\dagger}CB^{\dagger}\in\mathcal{M}$. Then, by using Theorem 1, we can obtain

$$\begin{aligned}\mathbb{E}_k[\|\tilde{X}^{(k+1)}-A^{\dagger}CB^{\dagger}\|_F^2] &= \mathbb{E}_k\left[\left\|X^{(k)}-A^{\dagger}CB^{\dagger}-\frac{\alpha}{\|A_{i,:}\|_2^2}A_{i,:}^TA_{i,:}(X^{(k)}-A^{\dagger}CB^{\dagger})BB^T\right\|_F^2\right]\\ &\leq \rho\left\|X^{(k)}-A^{\dagger}CB^{\dagger}\right\|_F^2,\end{aligned}$$

then

$$\mathbb{E}\left[\left\|\tilde{X}^{(k+1)}-A^{\dagger}CB^{\dagger}\right\|_F^2\right] \leq \rho\mathbb{E}\left[\left\|X^{(k)}-A^{\dagger}CB^{\dagger}\right\|_F^2\right]. \tag{18}$$

Combining (16)–(18) yields

$$
\begin{aligned}
\mathbb{E}\left[\left\|X^{(k+1)}-A^{\dagger}CB^{\dagger}\right\|_F^2\right] \leq& (1+\frac{1}{\varepsilon})\mathbb{E}\left[\left\|X^{(k+1)}-\tilde{X}^{(k+1)}\right\|_F^2\right] \\
&+(1+\varepsilon)\mathbb{E}\left[\left\|\tilde{X}^{(k+1)}-A^{\dagger}CB^{\dagger}\right\|_F^2\right] \\
\leq& (1+\frac{1}{\varepsilon})\frac{\alpha^2\|B\|_2^2\rho^{k+1}}{\|A\|_F^2}\left\|Z^{(0)}-(C-AA^{\dagger}CB^{\dagger}B)\right\|_F^2 \\
&+(1+\varepsilon)\rho\mathbb{E}\left[\left\|X^{(k)}-A^{\dagger}CB^{\dagger}\right\|_F^2\right] \\
\leq& (1+\frac{1}{\varepsilon})\frac{\alpha^2\|B\|_2^2\rho^{k+1}}{\|A\|_F^2}\left\|Z^{(0)}-(C-AA^{\dagger}CB^{\dagger}B)\right\|_F^2[1+(1+\varepsilon)] \\
&+(1+\varepsilon)^2\rho^2\mathbb{E}\left[\left\|X^{(k-1)}-A^{\dagger}CB^{\dagger}\right\|_F^2\right] \\
\leq& \cdots \\
\leq& (1+\frac{1}{\varepsilon})\frac{\alpha^2\|B\|_2^2\rho^{k+1}}{\|A\|_F^2}\left\|Z^{(0)}-(C-AA^{\dagger}CB^{\dagger}B)\right\|_F^2\sum_{i=0}^{k}(1+\varepsilon)^i \\
&+(1+\varepsilon)^{k+1}\rho^{k+1}\left\|X^{(0)}-A^{\dagger}CB^{\dagger}\right\|_F^2 \\
=& \frac{(1+\varepsilon)^{k+2}-(1+\varepsilon)}{\varepsilon^2}\frac{\alpha^2\|B\|_2^2\rho^{k+1}}{\|A\|_F^2}\left\|Z^{(0)}-(C-AA^{\dagger}CB^{\dagger}B)\right\|_F^2 \\
&+(1+\varepsilon)^{k+1}\rho^{k+1}\left\|X^{(0)}-A^{\dagger}CB^{\dagger}\right\|_F^2.
\end{aligned}
$$

This completes the proof. □

Remark 4. *Replacing B^T in (12) with $B^{\dagger}$, we obtain the following projection-based randomized extended block Kaczmarz mathod (ME-PREBK) iteration:*

$$
\begin{cases}
Z^{(k+1)} = Z^{(k)} - \frac{\alpha}{\|A_{:,j}\|_2^2}A_{:,j}(((A_{:,j}^T Z^{(k)})B^T)(B^{\dagger})^T), \\
X^{(k+1)} = X^{(k)} + \frac{\alpha}{\|A_{i,:}\|_2^2}A_{i,:}^T\left((C_{i,:}-Z_{i,:}^{(k+1)}-(A_{i,:}X^{(k)})B)B^{\dagger}\right),
\end{cases}
\tag{19}
$$

4. Numerical Experiments

In this section, we will present some experimental results of the proposed algorithms for solving various matrix equations, and compare them with ME-RGRK and ME-MWRK in [13] for consistent matrix equations and RBCD in [12] for inconsistent matrix equations. All experiments were carried out using MATLAB (version R2020a) on a DESKTOP-8CBRR86 with Intel(R) Core(TM) i7-4712MQ CPU @2.30GHz 2.29GHz, RAM 8GB and Windows 10.

All computations start from the initial guess $X^{(0)} = 0$, and are terminated once the relative error (RE) of the solution, defined by

$$
RE = \frac{\|X^{(k)} - X^*\|_F^2}{\|X^*\|_F^2}
$$

at the current iteration $X^{(k)}$, satisfies $RE < 10^{-6}$ or exceeds the maximum iteration K = 50,000, where $X^* = A^{\dagger}CB^{\dagger}$. We report the average number of iterations (denoted as "IT") and the average computing time in seconds (denoted as"CPU") for 20 repeated trial runs of the corresponding method. Three examples are tested, and A and B are generated as follows.

- Type I: For given m, p, q, n, the entries of A and B are generated from a standard normal distribution, i.e., $A = randn(m, p), B = randn(q, n)$.
- Type II: Like [18], for given m, p, and $r_1 = rank(A)$, we construct a matrix A by $A = U_1 D_1 V_1^T$, where $U_1 \in R^{m \times r_1}$ and $V_1 \in R^{p \times r_1}$ are orthogonal column matrices, $D \in R^{r_1 \times r_1}$ is a diagonal matrix whose first $r - 2$ diagonal entries are uniformly distributed numbers in $[\sigma_{\min}(A), \sigma_{\max}(A)]$, and the last two diagonal entries are $\sigma_{\max}(A), \sigma_{\min}(A)$. The entries of B are generated using a similar method with parameters $q, n, r_2 = rank(B)$.
- Type III: The real-world sparse data come from the Florida sparse matrix collection [23]. Table 2 lists the features of these sparse matrices.

Table 2. The detailed features of sparse matrices from [23].

Name	Size	Rank	Sparsity
ash219	219×85	85	2.3529%
ash958	958×292	292	0.68493%
divorce	50×9	9	50%

4.1. Consistent Matrix Equation

Given A, B, we set $C = AX^*B$ with $X^* = randn(p, q)$ to construct a consistent matrix equation. First, we test the impact of α in the ME-RBK method on the experimental results. Figure 1 plots the IT and CPU versus different "para" with different matrices in Table 3, where $para = 0.1{:}0.1{:}1.9$ so that $\alpha = \frac{para}{\|B\|_2^2}$ satisfies $0 < \alpha < \frac{2}{\|B\|_2^2}$ in Theorem 1. From Figure 1, it can be seen that the number of iteration steps and the running time decrease with the increase in parameters. However, when $para = 1.9$, both IT and CPU begin to increase. The same situations occur when solving consistent or inconsistent equations with different matrices in Tables 4 and 5. Therefore, we set $\alpha = \frac{1.8}{\|B\|_2^2}$ in all experiments.

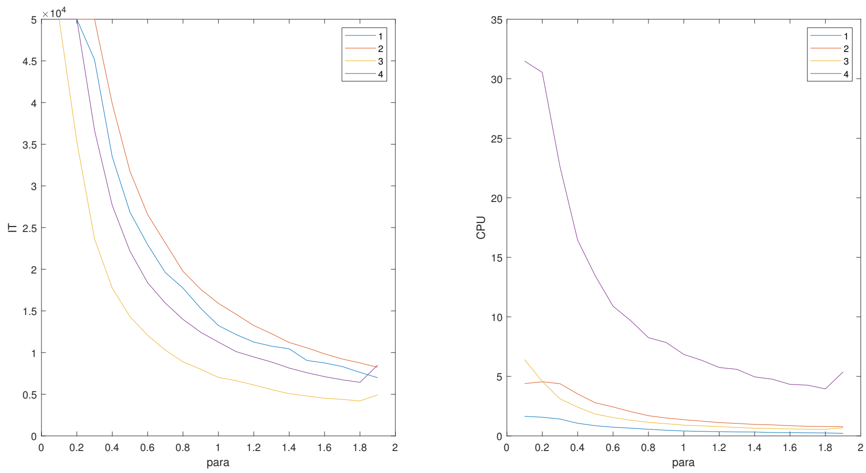

Figure 1. IT (**left**) and CPU (**right**) of different para of ME-RBK for consistent matrix equations with differnt matrices in Table 3.

In Tables 3–5, we report the average IT and CPU of the ME-RGRK, ME-MWRK, ME-RBK and ME-PRBK methods for solving consistent eqautions. In the following tables, the item ">" represents that the number of iteration steps exceeds the maximum iteration (50,000), and the item "-" represents that the method does not converge.

From these tables, we can see that the ME-RBK and ME-PRBK methods vastly outperform the ME-RGRK and ME-MWRK methods in terms of both IT and CPU times regardless of whether the matrices A and B are full column/row rank or not. As the matrix dimension increases, the CPU time of the ME-RBK and ME-PRBK methods increases slowly, while the running time of ME-RGRK and ME-MWRK increases dramatically.

In addition, when the matrix size is small, the ME-PRBK method is competitive, because the pseudoinverse is less expensive and the number of iteration steps is small. When the matrix size is large, the matrix is large, and the ME-RBK method is more challenging because it does not need to calculate the pseudoinverse (see the last line in Table 3).

Table 3. IT and CPU of ME-RGRK, ME-MWRK, ME-RBK and ME-PRBK for the consistent matrix equations with Type I.

No.	m	p	q	n		ME-RGRK	ME-MWRK	ME-RBK	ME-PRBK
1	100	40	40	100	IT CPU	49,707 0.78	27,579 2.01	7834.5 0.20	1152.8 0.04
2	40	100	100	40	IT CPU	> >	49,332.7 2.15	6334.7 0.33	1507.2 0.08
3	500	100	100	500	IT CPU	> >	32,109 57.61	4021.8 0.52	1866.1 0.19
4	1000	200	300	2000	IT CPU	> >	> >	6429.6 3.95	4450.4 0.72

Table 4. IT and CPU of ME-RGRK, ME-MWRK, ME-RBK and ME-PRBK for the consistent matrix equations with Type II.

m	p	r_1	$[\sigma_{\min}(A), \sigma_{\max}(A)]$	q	n	r_2	$[\sigma_{\min}(B), \sigma_{\max}(B)]$		ME-RGRK	ME-MWRK	ME-RBK	ME-PRBK
100	40	20	[1, 2]	40	100	40	[1, 2]	IT CPU	4361.0 0.06	1987.1 0.17	503.1 0.01	332.4 0.008
100	40	20	[1, 5]	40	100	20	[1, 5]	IT CPU	22,423.2 0.33	6439.8 0.57	9307.6 0.20	1056.3 0.02
1000	200	100	[1, 2]	100	1000	50	[1, 2]	IT CPU	20,055.5 78.45	7047.8 70.56	2587.4 0.42	1674.3 0.23
1000	100	50	[1, 5]	200	1000	200	[1, 5]	IT CPU	> >	> >	18,898.3 3.61	2833.6 0.42

Table 5. IT and CPU of ME-RGRK, ME-MWRK, ME-RBK and ME-PRBK for the consistent matrix equations with Type III.

A	B		ME-RGRK	ME-MWRK	ME-RBK	ME-PRBK
divorce	ash219^T	IT CPU	43,927.8 1.15	14,164.4 1.35	10,993.4 0.36	3873.5 0.13
divorce	ash219	IT CPU	40,198.7 0.63	17,251.4 0.80	11,557.4 0.43	3124.7 0.11
ash219	ash958^T	IT CPU	> >	> >	6042.3 1.04	2267.0 0.36
ash219	ash958	IT CPU	> >	> >	5745.4 1.22	2114.2 0.42

4.2. *Inconsistent Matrix Equation*

To construct an inconsistent matrix equation, we set $C = AX^*B + R$, where X^* and R are random matrices which are generated by $X^* = randn(p, q)$ and $R = \delta * randn(p, q)$, $\delta \in (0, 1)$. Numerical results of the RBCD, IME-REBK and IME-PREBK methods are listed in Tables 6–8. From these tables, we can see that the IME-PREBK method is better than the RBCD method in terms of IT and CPU time, especially when the $\frac{\sigma_{\max}}{\sigma_{\min}}$ is large (see the last line in Table 7). The IME-REBK method is not competitive for B with full row rank because it needs to solve two equations. However, when B does not have full row rank, the RBCD method does not converge, while the IME-REBK and IME-PREBK methods do.

Table 6. IT and CPU of RBCD, IME-REBK and IME-PREBK for the inconsistent matrix equations with Type I.

m	p	q	n		RBCD	IME-REBK	IME-PREBK
100	40	40	100	IT CPU	17,212.2 0.69	21,270.5 1.57	2469.4 0.19
40	100	100	40	IT CPU	- -	24,708.3 2.38	2174.6 0.21
500	100	100	500	IT CPU	6059.1 4.97	7352.8 7.28	2512.3 2.70
1000	200	300	2000	IT CPU	14,209.2 152.32	12,490.4 246.56	5183.5 99.68

Table 7. IT and CPU of RBCD, IME-REBK and IME-PREBK for the inconsistent matrix equations with Type II.

m	p	r_1	$[\sigma_{\min}(A), \sigma_{\max}(A)]$	q	n	r_2	$[\sigma_{\min}(B), \sigma_{\max}(B)]$		RBCD	IME-REBK	IME-PREBK
100	40	20	[1, 2]	40	100	40	[1, 2]	IT CPU	1035.9 0.04	762.2 0.05	384.5 0.03
100	40	20	[1, 5]	40	100	20	[1, 5]	IT CPU	- -	16,224.7 1.23	1507.8 0.12
1000	200	100	[1, 2]	100	1000	50	[1, 2]	IT CPU	- -	4067.5 24.23	2217.8 13.67
1000	200	100	[1, 5]	100	1000	100	[1, 5]	IT CPU	> >	48,269.6 365.78	4328.0 30.25

Table 8. IT and CPU of RBCD, IME-REBK and IME-PREBK for the inconsistent matrix equations with Type III.

A	B		RBCD	IME-REBK	IME-PREBK
divorce	ash219^T	IT CPU	> >	20,026.3 2.26	4308.4 0.48
divorce	ash219	IT CPU	- -	19,199.1 1.49	4026.2 0.31
ash219	ash958^T	IT CPU	22,313.4 15.69	10,823.5 19.03	2561.8 5.89
ash219	ash958	IT CPU	- -	10,020.7 15.83	2363.5 4.72

5. Conclusions

In this paper, we have proposed a randomized block Kaczmarz algorithm for solving the consistent matrix equation and its extended version for the inconsistent case. Theoretically, we have proved that the proposed algorithms converge linearly to the unique minimal F-norm solution or least-squares solution (i.e., $A^\dagger CB^\dagger$) without requirements on A and B having full column/row rank. The numerical results show the effectiveness of the algorithms. Since the proposed algorithms only require one row or one column of A at each iteration without a matrix–matrix product, they are suitable for the scenarios where the matrix A is too large to fit in the memory or matrix multiplication is considerably expensive.

Author Contributions: Conceptualization, L.X.; Methodology, L.X. and W.B.; Validation, L.X. and W.L.; Writing—Original Draft Preparation, L.X.; Writing—Review and Editing, L.X., W.B. and W.L.; Software, L.X.; Visualization, L.X. and W.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability Statement: The datasets that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments: The authors are thankful to the referees for their constructive comments and valuable suggestions, which have greatly improved the original manuscript of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rauhala, U.A. Introduction to array algebra, Photogramm. *Eng. Remote Sens.* **1980**, *46*, 177–192.
2. Regalia, P.A.; Mitra, S.K. Kronecker products, unitary matrices and signal processing applications. *SIAM Rev.* **1989**, *31*, 586–613. [CrossRef]
3. Liu, M.Z.; Li, B.J.; Guo, Q.J.; Zhu, C.; Hu, P.; Shao, Y. Progressive iterative approximation for regularized least square bivariate B-spline surface fitting. *J. Comput. Appl. Math.* **2018**, *327*, 175–187. [CrossRef]
4. Liu, Z.Y.; Li, Z.; Ferreira, C.; Zhang, Y.L. Stationary splitting iterative methods for the matrix equation $AXB = C$. *Appl. Math. Comput.* **2020**, *378*, 125195. [CrossRef]
5. Fausett, D.W.; Fulton, C.T. Large least squares problems involving Kronecker products. *SIAM J. Matrix Anal. Appl.* **1994**, *15*, 219–227. [CrossRef]
6. Zha, H.Y. Comments on large least squares problems involving Kronecker products. *SIAM J. Matrix Anal. Appl.* **1995**, *16*, 1172. [CrossRef]
7. Peng, Z.Y. An iterative method for the least squares symmetric solution of the linear matrix equation AXB = C. *Appl. Math. Comput.* **2005**, *170*, 711–723. [CrossRef]
8. Ding, F.; Liu, P.X.; Ding, J. Iterative solutions of the generalized Sylvester matrix equations by using the hierarchical identification principle. *Appl. Math. Comput.* **2008**, *197*, 41–50. [CrossRef]
9. Huang, G.X.; Yin, F.; Guo, K. An iterative method for the skew-symmetric solution and the optimal approximate solution of the matrix equation AXB = C. *J. Comput. Appl. Math.* **2008**, *212*, 231–244. [CrossRef]
10. Wang, X.; Li, Y.; Dai, L. On hermitian and skew-hermitian splitting iteration methods for the linear matrix equation AXB = C. *Comput. Math. Appl.* **2013**, *65*, 657–664. [CrossRef]
11. Shafiei, S.G.; Hajarian, M. Developing Kaczmarz method for solving Sylvester matrix equations. *J. Franklin Inst.* **2022**, *359*, 8991–9005. [CrossRef]
12. Du, K.; Ruan, C.C.; Sun, X.H. On the convergence of a randomized block coordinate descent algorithm for a matrix least squaress problem. *Appl. Math. Lett.* **2022**, *124*, 107689. [CrossRef]
13. Wu, N.C.; Liu, C.Z.; Zuo, Q. On the Kaczmarz methods based on relaxed greedy selection for solving matrix equation AXB = C. *J. Comput. Appl. Math.* **2022**, *413*, 114374. [CrossRef]
14. Strohmer, T.; Vershynin, R. A randomized Kaczmarz algorithm with exponential convergence. *J. Fourier. Anal. Appl.* **2009**, *15*, 262–278. [CrossRef]
15. Zouzias, A.; Freris, N.M. Randomized extended Kaczmarz for solving least squares. *SIAM J. Matrix. Anal. Appl.* **2013**, *34*, 773–793. [CrossRef]
16. Ion, N. Faster randomized block kaczmarz algorithms. *SIAM J. Matrix Anal. Appl.* **2019**, *40*, 1425–1452.
17. Nemirovski, A.; Juditsky, A.; Lan, G.; Shapiro, A. Robust stochastic approximation approach to stochastic programming. *SIAM J. Optimiz.* **2009**, *19*, 1574–1609. [CrossRef]
18. Niu, Y.Q.; Zheng, B. On global randomized block Kaczmarz algorithm for solving large-scale matrix equations. *arXiv* **2022**, arXiv:2204.13920.
19. Needell, D. Randomized Kaczmarz solver for noisy linear systems. *BIT Numer. Math.* **2010**, *50*, 395–403. [CrossRef]
20. Ma, A.; Needell, D.A. Ramdas, Convergence properties of the randomized extended Gauss-Seidel and Kaczmarz methods. *SIAM J. Matrix Anal. Appl.* **2015**, *36*, 1590–1604. [CrossRef]
21. Du, K. Tight upper bounds for the convergence of the randomized extended Kaczmarz and Gauss-Seidel algorithms. *Numer. Linear Algebra Appl.* **2019**, *26*, e2233. [CrossRef]

22. Du, K.; Si, W.T.; Sun, X.H. Randomized extended average block kaczmarz for solving least squares. *SIAM J. Sci. Comput.* **2020**, *42*, A3541–A3559. [CrossRef]
23. Davis, T.A.; Hu, Y. The university of Florida sparse matrix collection. *Math. Softw.* **2011**, *38*, 1–25.