Advances in Applied Mathematics 应用数学进展, 2024, 13(1), 466-484
Published Online January 2024 in Hans. https://www.hanspub.org/journal/aam
https://doi.org/10.12677/aam.2024.131046

Hans汉斯

关于Kaczmarz的一类加速免伪逆贪婪块方法

颜鑫鹏，时文雅，郇　战*

常州大学阿里云大数据学院，江苏 常州

收稿日期：2023年12月27日；录用日期：2024年1月21日；发布日期：2024年1月30日

摘　要

块贪婪Kaczmarz方法在解决大规模一致线性系统方面取得了成功应用。然而在每次迭代步骤中，GBK方法都涉及伪逆计算，这不仅复杂化了计算并减慢了收敛速度，且不适合分布式实现。在本文中基于Sketching技术提出了两种免伪逆计算的GBK方法，分别是杠杆得分抽样免伪逆GBK方法和稀疏随机投影免伪逆GBK方法，其算法效率更加高效，收敛速度可以达到指数收敛。为了进一步加快收敛速度，我们还提出了CountSketch免伪逆重力球GBK方法、杠杆得分抽样免伪逆重力球GBK方法和稀疏随机投影免伪逆重力球GBK方法。为了验证新方法的有效性，我们进行了一些数值示例。结果表明，这些新方法在解决大规模一致线性系统方面具有很高的效率和准确性。

关键词

贪婪块Kaczmarz方法，收敛性，大规模相容线性方程组，矩阵Sketching技术，免伪逆计算

Accelerated Pseudo-Inverse-Free Greedy Block Methods in the Kaczmarz Algorithm Category

Xinpeng Yan, Wenya Shi, Zhan Huan*

Aliyun School of Big Data, Changzhou University, Changzhou Jiangsu

Received: Dec. 27th, 2023; accepted: Jan. 21st, 2024; published: Jan. 30th, 2024

Abstract

The Greedy Block Kaczmarz (GBK) method has been successfully applied in solving large-scale consistent linear systems. However, each iteration of the GBK method involves the computation of

*通讯作者。

文章引用: 颜鑫鹏, 时文雅, 郇战. 关于 Kaczmarz 的一类加速免伪逆贪婪块方法[J]. 应用数学进展, 2024, 13(1): 466-484. DOI: 10.12677/aam.2024.131046

pseudoinverses, which complicates the process, slows down convergence, and is ill-suited for distributed implementations. In this paper, we introduce two free pseudoinverse GBK methods based on Sketching techniques: the Leverage Score Sampling Free Pseudoinverse GBK method and the Sparse Random Projection Free Pseudoinverse GBK method. These algorithms exhibit higher efficiency and can achieve exponential rates of convergence. To further accelerate convergence, we also propose the Count Sketch Free Pseudoinverse Heavy Ball GBK method, the Leverage Score Sampling Free Pseudoinverse Heavy Ball GBK method, and the Sparse Random Projection Free Pseudoinverse Heavy Ball GBK method. The effectiveness of these new methods is demonstrated through several numerical examples, showing that they are highly efficient and accurate in addressing large-scale consistent linear systems.

Keywords

Greedy Block Kaczmarz Method, Convergence, Large-Scale Compatible Linear Equation Systems, Matrix Sketching Techniques, Pseudo-Inverse Calculation

1. 引言

大规模相容线性方程组的高效求解对科学与工程应用[1] [2] [3] [4] [5]具有重要意义和实际价值。

$$Ax = b, A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m \tag{1}$$

经典 Kaczmarz 算法[6]以其结构简明和易于实施的特点，在大规模线性方程组求解中显现优势。该算法每次迭代仅处理单个行样本，保证了简洁高效，尽管其收敛性难以与其他迭代方法[7] [8]进行比较且可能较慢。研究人员正致力于优化 Kaczmarz 算法，以提升其在实际应用中的表现，确保在科学和工程领域的高效解决方案。

Strohmer 和 Vershynin [9]近期提出的随机 Kaczmarz (RK)方法，以其依赖于矩阵规范化条件数的快速收敛性而备受关注。该方法采用概率策略选取矩阵行指标，即根据公式

$$Pr = \frac{\left\| A_{(i_k)} \right\|_2^2}{\left\| A \right\|_F^2}$$

来选择行指标，能在特定条件下超越传统共轭梯度法。研究者对 Kaczmarz 方法持续优化，扩展至包括不相容、欠定和秩亏线性方程组[10] [11] [12]的收敛性分析，并引入 Nesterov 加速框架[13] [14]和贪婪策略[15] [16] [17] [18]等加速技巧。这些进展对解决线性方程组和优化问题领域产生了深远影响。

块 Kaczmarz (BK)方法[19]及其变体如随机块 Kaczmarz (RBK) [20]、随机双块 Kaczmarz (RDBK) [21]和贪婪块 Kaczmarz (GBK) [22]，针对线性方程组求解展现出高效迭代性能。这些算法通过分块策略，有效利用方程信息，实现快速收敛。特别是在处理不相容系统和大规模数据时，这些方法通过随机选择和贪婪选择块，优化了迭代过程，提升了稳定性和收敛性。块高斯 Kaczmarz 方法(BGK) [23]融合高斯 Sketching 技术，优化分布式计算性能，实现大型线性系统的有效分解与并行解算。然而，算法中伪逆计算和最小二乘问题求解的高计算量是挑战所在，优化这些关键步骤是当前研究的焦点。

Necoara 提出的随机平均块 Kaczmarz (RABK)方法[24]，Moorman 等[25]研究了 RABK 方法在解不相

容线性方程组中的收敛性理论，其衍生的简单随机扩展平均块 Kaczmarz (REABK)方法[26]，该方法结合了随机扩展 Kaczmarz (REK)方法[27]和 RABK 方法。Du 和 Sun 扩展了该方法，并提出了一种免于使用伪逆的随机块迭代算法[28]，适用于解决一致与不一致线性方程组。最近，Zhang Jianhua 等人在 REABK 的基础上推出了快速免伪逆贪婪块 Kaczmarz (CFGBK)方法[29]，在处理相容与不相容线性方程组方面展现出高效性，但是随着研究发现该方法由于每步迭代 $\left\|A_{(i)}\right\|_2^2$ 存在出现为零的情况导致收敛失败，本文提出了一种新型加速免伪逆贪婪对该问题进行了研究并解决。

矩阵 Sketching 技术作为提升矩阵运算速度的关键工具，在偏微分方程反问题[30]、优化[31]与回归分析[32] [33] [34] [35]等多个领域发挥着至关重要的作用。本研究创新性地提出了两种基于近似最大距离准则的免伪逆贪婪块 Kaczmarz 方法(LFGBK、CFGBK)，通过精心选择采样矩阵和迭代策略，显著缩短了运算时间，同时保持了精度，并深入分析了其收敛性。此外，借助重力球技术，进一步提出了三种加速方法(CMFGBK、LMFGBK、CMFGBK)，并建立了完备的收敛性理论框架。经过数值实验验证，这些方法不仅提高了计算效率，也为矩阵 Sketching 技术的进一步优化奠定了坚实基础。

本文结构如下：第二节介绍预备知识和贪婪块 Kaczmarz 方法和免伪逆贪婪块 Kaczmarz 方法。第三节提出改进免伪逆贪婪块 Kaczmarz 方法并给分析。第四节给出数值实验。第五节总结全文。

2. 预备知识

在本文中，我们采用文献中[17]同样的记号。例如 $A_{(i)}$、A^{T}、$A^{\dagger}$、$\|A\|$、$\|A\|_F$、$[n]$ 分别表示系数矩阵 A 的第 i 行、转置、广义逆、谱范数、F-范数和集合 $\{1,2,\cdots,n\}$。

我们首先考虑贪婪选择规则，然后使用贪婪策略提供一个贪婪块 Kaczmarz 算法。在研究贪婪选择规则在 kaczmarz 型算法中的应用的文献中，很少有结果。Nutini 等人在[16]中提出了 Kaczmarz 算法的最大残差(MR)和最大距离(MD)规则。然而，在许多应用程序中，由于其复杂的表达式，计算精确的 MR 或 MD 规则将过于低效，但我们可以通过使用更便宜的近似贪婪规则来近似它，如[18]方法。在本节中，我们将考虑计算贪婪规则直至乘法误差的方法。

再给出收敛性分析之前首先介绍引理。

引理 1 [36]让 $\{F_t\}_{t\ge 0}$ 是一个非负实数矩阵，其中 $F_0=F_1$ 满足关系式 $F_{t+1}\le a_1F_t+a_2F_{t-1}$ 对所有 $t\ge 1$ 成立，这里 $a_2>0$ 且 $a_1+a_2<1$。对所有 $t\ge 1$ 下列不等式成立：

$$F_{t+1}\le q^t(1+\delta)F_0$$

其中，$q=\dfrac{a_1+\sqrt{a_1^2+4a_1}}{2},\delta=q-a_1$ 和 $q\le a_1+a_2$。

引理 2 [17]如果任意向量 $u\in range\left(A^{\mathrm{T}}\right)$，则

$$\|Au\|_2^2\ge\lambda_{min}\left(A^{\mathrm{T}}A\right)\|u\|_2^2$$

引理 3 [35] [37]如果 S 是含有 $O\left(n^2/\delta\theta^2\right)$ 行稀疏随机变换，其中那么不等式

$$(1-\theta)\|Ax\|_2\le\|SAx\|_2\le(1+\theta)\|Ax\|_2,x\in\mathbb{R}^n$$

和

$$(1-\theta)\sigma_i(SA)\le\sigma_i(A)\le(1+\theta)\sigma_i(SA),1\le i\le d,0<\delta,\theta<1,$$

成立的概率均为 $1-\delta$。

定义 1 [35] (CountSketch 变换)：设 $h:[m]\to[d]$ 是一个随机映射，使得每个 $i\in[m]$，$h(i)=j$ 对每个 $j\in[d]$ 成立的概率为 1/d，$\Phi\in\{0,1\}^{d\times m}$ 是一个 $d\times m$ 二值矩阵，其中其余元素均为 0。D 是 $m\times m$。随机对

角矩阵，每个对角元素以相同的概率独立选择值为 1 或者−1。则 CountSketch 变换定义为 $S = \Phi D \in \mathbb{R}^{d\times m}$。

定义 1 描述了 CountSketch 变换的基本过程：首先，通过随机映射 h 和二值矩阵 Φ 将输入矩阵的行映射到较低的维度空间；然后，通过随机对角矩阵 D 对映射后的结果进行随机翻转。这个过程可以有效地减小矩阵的大小，同时保持某些重要的信息。

Algorithm 1. **基于流形式的 CountSketch 方法**

输入： $A \in \mathbb{R}^{m\times n}$
将初始化为 $d \times n$ 全零矩阵
for $k = 1, \cdots, m$
从集合 $[d]$ 中随机均匀地选择样本值 l
从集合 $\{-1,1\}$ 中随机均匀地选择样本值 g
使用 $c_{l:} \leftarrow c_{l:} + g a_{k:}$ 更新 C 的第 l 行
end for
返回 $C \in \mathbb{R}^{d\times n}$

定义 2 [38] [39] (Leverage Score Sampling 变换)：给定矩阵 $A \in \mathbb{R}^{m\times n}$，其奇异值分解为 $A = UDV^{\mathrm{T}}$，其行杠杆得分给出 U 行的欧几里得范数得平方，即对于每个 $j \in [d]$，$U\ell_j = \left\| e_j^{\mathrm{T}} U \right\|_2^2 = \left\| u_j \right\|_2^2$，同时满足 $0 \le \ell_j \le 1$ 和 $\sum_{i=j}^{n} \ell_j = p$。杠杆得分抽样也可以描述为“帽矩阵”。

定义 2 描述了 Leverage Score Sampling 变换的基本过程，首先设 A 是一个 $n \times d$ 的矩阵，每一行的 Leverage Score 是该行在 A 的奇异值分解中对应的右奇异向量的范数的平方。这个得分衡量了每一行在数据集中的重要性或影响力。这种采样方法的优点是，它可以从大数据集中选择出一小部分具有代表性的样本，从而进行更高效的计算。所得到的样本集 $S \in R^{k\times d}$，其中 k 是选定的样本数量，可以用于估计原矩阵 CA 的各种性质，例如奇异值分解、主成分分析等。

Algorithm 2. **基于流形式的 Leverage Score Sampling 方法**

输入： $A \in \mathbb{R}^{m\times n}$
将初始化为 $d \times n$ 全零矩阵
设置变量 $sum = 0$
计算 A 的奇异值分解 U, S, V
for $k = 1, \cdots, m$
对 U 的每一行求平方和，加入到 sum
计算每个数据点被抽样的概率：U 的每一行求平方和，除以 sum
根据计算出的概率进行抽样，得到抽样索引 $indices$
使用 $C = [indices, :]$
end for
返回 $C \in \mathbb{R}^{d\times n}$

定义 3 [40] (Sparse Random Projection 变换)：设矩阵 $A \in \mathbb{R}^{m\times n}$，其中 $A_{ij} \sim N(0,1)$，则 $P(A_{ij} \neq 0) = 1/\sqrt{m}$。

定义 3 描述了 Sparse Random Projection 变换的基本过程，首先 Sparse Random Projection 投影后的维度 k，然后构造一个 $d \times k$ 的稀疏矩阵 R。对于 R 中的每一列，我们随机选择一个元素并赋值为+1 或−1，其余元素设为 0。这个稀疏随机矩阵的每一列都只有一个非零元素，该元素的位置在每一列中都是随机选择的，其值是根据一个预先定义的分布随机选择的。这种方法的优点是它的计算效率高，因为乘以稀

DOI: 10.12677/aam.2024.131046

疏矩阵的计算复杂度低。这种方法在处理高维数据时，特别是在近似最近邻搜索和其他需要降维的应用中，被广泛使用。此外，由于 R 的元素大部分是 0，所以投影后的数据也会保持原始数据的稀疏性。

Algorithm 3. 基于流形式的 Sparse Random Projection 方法

1：输入：$A\in\mathbb{R}^{m\times n}$
2：将 C 初始化为 $d\times n$ 全零矩阵
3：生成 $d\times m$ 稀疏随机矩阵 S，其中非零元素的概率为 $1/\sqrt{m}$
4：使 $C=S$
5：返回 $C\in\mathbb{R}^{d\times n}$

定义 4 [36] (Heavy Ball Momentum 优化)：重球动量是一种广泛添加到梯度下降方法中的增强方法，它在每个迭代步骤中不仅采取梯度下降的步骤，还额外在前一迭代步骤的移动方向上采取一步。这一方法最初由 Polyak 在 1964 年提出，后来在机器学习领域广泛应用。

Algorithm 4. Heavy Ball Momentum 优化方法

1：将 α 初始化，$v=0$
2：Set $x_1=x_0$
3：**while** *ture*
4：$x_{k+1}=x_k+\alpha_k\left(\sum_{i\in\mathcal{T}_k}w_i\frac{b_i-A_{(i)}x_k}{\left\|A_{(i)}\right\|_2^2}A_{(i)}^{\mathrm{T}}\right)+\alpha\left(x_t-x_{t-1}\right)$
5：**end while**
6：返回 x

假设我们已经近似了 MD 规则，其中有一个参数 $\eta\in(0,1]$，用于选择指标 i_k

$$\frac{\left|b_{i_k}-A_{(i_k)}x_k\right|^2}{\left\|A_{(i_k)}\right\|_2^2}\geq\eta\max_{1\leq i\leq m}\left\{\frac{\left|b_i-A_{(i)}x_k\right|^2}{\left\|A_{(i)}\right\|_2^2}\right\}$$

Niu 和 Zheng 将贪婪策略与块 Kaczmarz 方法相结合，提出了求解大型相容线性方程组的贪婪块 Kaczmarz (GBK) [22]方法，具体过程见算法 5。

Algorithm 5. GBK 方法

1：输入：$A,b,l,x_0\in range\left(A^{\mathrm{T}}\right)$ 和参数 $\eta\in(0,1)$
2：输出：x_l
3：for $k=0,1,\cdots,l-1$ do
4：计算 $\varepsilon_k=\eta\max_{1\leq i\leq m}\left\{\frac{\left|b_i-A_{(i)x_k}\right|^2}{\left\|A_{(i)}\right\|_2^2}\right\}$
5：确定指标集序列 $\mathcal{T}_k=\left\{i_k:\left|b_{ik}-A_{(i_k)}x_k\right|^2\geq\varepsilon_k\left\|A_{(i_k)}\right\|_2^2\right\}\{x_k\}_{k=0}^{\infty}$
6：计算 $x_{k+1}=x_k+A_{\mathcal{T}_k}^{\dagger}\left(b_{\mathcal{T}_k}-A_{\mathcal{T}_k}x_k\right)$
7：end for

GBK 方法的收敛性分析描述如下：

定理 1 [22]设线性方程组(1)相容，则由算法 7 生成的迭代序列收敛到方程组的最小范数解 $x_* = A^{\dagger}b$，且对任意 $k \ge 0$ 满足

$$x_{k+1} - x_{*2}^{2} \le \left(1 - \gamma_k(\eta)\frac{\sigma_{min}^2(A)}{\|A\|_F^2}\right)^{k+1} \|x_0 - x_*\|_2^2$$

其中 $\gamma_k(\eta) = \eta \frac{\|A\|_F^2}{\|A\|_F^2 - \|A_{\mathcal{T}_{k-1}}\|_F^2} \frac{\|A_{\mathcal{T}_k}\|_F^2}{\sigma_{max}^2(A_{\mathcal{T}_k})}$ （记 $\gamma_{0(\eta)} \equiv \eta \frac{\|A_{\mathcal{T}_0}\|_F^2}{\sigma_{max}^2(A_{\mathcal{T}_0})}$），$\eta \in (0,1]$，$\sigma_{min}(A)$ 和 $\sigma_{max}(A)$ 分别表示矩阵 A 的非零最小奇异值和最大奇异值。

3. 改进免伪逆贪婪块 Kaczmarz 方法及其收敛性分析

由于 CFGBK 方法每步迭代 $\|A_{(i)}\|_2^2$ 存在出现为零的情况，本节提出了改进免伪逆贪婪块 Kaczmarz 方法。首先每步采用近似最大举例准则

$$\frac{|b_{i_k} - A_{(i_k)}x_k|^2}{\|A_{(i_k)}\|_2^2} \ge \eta \max_{1\le i\le m}\left\{\frac{|b_i - A_{(i)}x_k|^2}{\|A_{(i)}\|_2^2}\right\},$$

选择块矩阵 $A_{\mathcal{T}_k}$，的指标集 $\mathcal{T}_k$；其次，将当前估计值投影到构成块矩阵 $A_{\mathcal{T}_k}$ 的每一行上；最后，对得到的投影求平均值来计算下一次迭代，即

$$x_{k+1} = x_k + \alpha_k\left(\sum_{i\in\mathcal{T}_k} w_i \frac{b_i - A_{(i)}x_k}{\|A_{(i)}\|_2^2} A_{(i)}^{\mathrm{T}}\right)$$

Algorithm 6. LFGBK 方法

1: $A, b, l, x_0 \in range(A^{\mathrm{T}})$，参数 $\eta \in (0,1)$，步长序列 $(\alpha_k)_k \ge 0$ 和权重序列 $(w_k)_k \ge 0$

2: 输出：x_l

3: 初始化：由算法 2 生成 Leverage Score sampling 变换矩阵 $\tilde{A} = SA \in \mathbb{R}^{d\times n}$ 和向量 $\tilde{b} = Sb$，其中 $d \ll m$

4: for $k = 0,1,\cdots,l-1$ do

5: 计算 $\tilde{\varepsilon}_k = \eta \max_{1\le i\le m}\left\{\frac{|\tilde{b}_i - \tilde{A}_{(i)x_k}|^2}{\|\tilde{A}_{(i)}\|_2^2}\right\}$

6: 确定指标集序列 $\mathcal{T}_k = \left\{i_k : |\tilde{b}_{i_K} - \tilde{A}_{(i_k)}x_k|^2 \ge \varepsilon_k \|\tilde{A}_{(i_k)}\|_2^2\right\}$

7: 计算 $x_{k+1} = x_k + \alpha_k\left(\sum_{i\in\mathcal{T}_k} w_i \frac{\tilde{b}_i - \tilde{A}_{(i)}x_k}{\|\tilde{A}_{(i)}\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}\right)$

8: end for

Leverage Score Sampling 可以从大数据集中选择出一小部分具有代表性的样本，从而进行更高效的计算与 CFGBK 相比，采样比 Count Sketch 采样更高效，第四节中的数值实验将证实 LFGBK 方法比 CFGBK 更高效。

本文只讨论 $\alpha_k=1$ 的情况，下面给出 LFGBK 方法求解大型相容线性方程组(1)的收敛性理论。

定理 2 设 Leverage Score 变换 S 满足 $d=O\left(n^2/\delta\theta^2\right)$， $x_*=A^\dagger b$ 是相容线性方程组(1)的最小范数解，对任意 $k\geq 0$ 满足

$$\|x_{k+1}-x_*\|_2^2\leq\left(1-\tilde{\psi}_k(\eta)\frac{\sigma_{min}^2(\tilde{A})}{\|\tilde{A}\|_F^2}\right)\|x_k-x_*\|_2^2 \tag{2}$$

其中 $\tilde{\psi}_k(\eta)=\eta t\left(\frac{2}{t}-\frac{1}{t^2}\sigma_{max}^2\left(\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\right)\right)$， $\eta\in(0,1]$，且 $range(A),\sigma_{min}(A)$ 和 $\sigma_{max}(A)$ 分别表示矩阵 A 的值域，非零最小奇异值和最大奇异值。

证明 由算法 6 和 $\tilde{r}_k=\tilde{b}-\tilde{A}x_k$，我们可以得到

$$x_{k+1}=x_k+\left(\sum_{i\in\mathcal{T}_k}\frac{1}{|\mathcal{T}_k|}\frac{\tilde{r}_k^i\tilde{A}_{(i)}^{\mathrm{T}}}{\|\tilde{A}_{(i)}\|_2^2}\right) \tag{3}$$

设 $|\mathcal{T}_k|=t$ 和 $\mathcal{T}_k=\left\{j_{k_1},\cdots,j_{k_t}\right\}$，将(3)式子展开可得

$$x_{k+1}=x_k+\sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}^{\mathrm{T}}e_i^{\mathrm{T}}\tilde{r}_k}{\|\tilde{A}_{(i)}\|_2^2}=x_k+\frac{1}{t}\left(\frac{\tilde{A}_{(j_{k_1})}^{\mathrm{T}}e_{j_{k_1}}^{\mathrm{T}}\tilde{r}_k}{\|\tilde{A}_{(j_{k_1})}\|_2^2}+\cdots+\frac{\tilde{A}_{(j_{k_t})}^{\mathrm{T}}e_{j_{k_t}}^{\mathrm{T}}\tilde{r}_k}{\|\tilde{A}_{(j_{k_t})}\|_2^2}\right)=x_k+\frac{1}{t}\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\hat{I}_{\mathcal{T}_k}\tilde{r} \tag{4}$$

其中

$$\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}=\left[\frac{\tilde{A}_{(j_{k_1})}^{\mathrm{T}}}{\|\tilde{A}_{(j_{k_1})}\|_2},\frac{\tilde{A}_{(j_{k_2})}^{\mathrm{T}}}{\|\tilde{A}_{(j_{k_2})}\|_2},\cdots,\frac{\tilde{A}_{(j_{k_t})}^{\mathrm{T}}}{\|\tilde{A}_{(j_{k_t})}\|_2}\right]\in\mathbb{R}^{n\times t} \tag{5}$$

和

$$\hat{I}_{\mathcal{T}_k}=\left[\frac{e_{(j_{k_1})}}{\|\tilde{A}_{(j_{k_1})}\|_2},\frac{e_{(j_{k_2})}}{\|\tilde{A}_{(j_{k_2})}\|_2},\cdots,\frac{e_{(j_{k_t})}}{\|\tilde{A}_{(j_{k_t})}\|_2}\right]^{\mathrm{T}}\in\mathbb{R}^{t\times d} \tag{6}$$

(4)式同时减去 x_*，可得

$$x_{k+1}-x_*=x_k-x_*-\frac{1}{t}\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\hat{I}_{\mathcal{T}_k}\tilde{A}(x_k-x_*)=\left(I-\frac{1}{t}\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\hat{A}_{\mathcal{T}_k}\right)(x_k-x_*) \tag{7}$$

对(7)式两边同时取谱范数并平方，又对任意半正定矩阵 $Q\succcurlyeq 0$ 满足 $Q^2\preccurlyeq\lambda_{\max}(Q)Q$，从而可以得到

$$\|x_{k+1}-x_*\|_2^2=\left\|(x_k-x_*)-\frac{1}{t}\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\hat{A}_{\mathcal{T}_k}(x_k-x_*)\right\|\leq\|x_k-x_*\|_2^2-\left(\frac{2}{t}-\frac{1}{t^2}\sigma_{max}^2\left(\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\right)\right)\left\|\hat{A}_{\mathcal{T}_k}(x_k-x_*)\right\|_2^2 \tag{8}$$

由(5)式和 $\left|\tilde{b}_{j_k}-\tilde{A}_{(j_k)}x_k\right|^2\geq\tilde{\varepsilon}_k\left\|\tilde{A}_{(j_k)}\right\|_2^2$ 通过简单计算可得

$$\left\|\tilde{A}_{\mathcal{T}_k}(x_k-x_*)\right\|_2^2=\sum_{j_k\in\mathcal{T}_k}\frac{1}{\|\tilde{A}_{(j_k)}\|_2^2}\left|\tilde{r}_k^{j_k}\right|^2\geq\sum_{j_k\in\mathcal{T}_k}\frac{1}{\|\tilde{A}_{(j_k)}\|_2^2}\tilde{\varepsilon}_k\left\|\tilde{A}_{(j_k)}\right\|^2 \tag{9}$$

把 $\tilde{\varepsilon}_k=\eta\max\limits_{1\le i\le d}\left\{\dfrac{\left|\tilde{b}_i-\tilde{A}_{(i)x_k}\right|^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right\}$ 带入上式，可得

$$\left\|\hat{A}_{\mathcal{T}_k}\left(x_k-x_*\right)\right\|_2^2\ge\sum_{j_k\in\mathcal{T}_k}\eta\max_{1\le i\le d}\left\{\frac{\left|\tilde{r}_k^i\right|^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right\}\ge\eta t\sum_{i=1}^{d}\frac{\left|\tilde{r}_k^i\right|^2}{\left\|A_{(i)}\right\|_2^2}\frac{\left\|\tilde{A}_{(i)}\right\|_2^2}{\left\|A\right\|_F^2}=\eta t\frac{\left\|\tilde{r}_k\right\|^2}{\left\|\tilde{A}\right\|_F^2}\tag{10}$$

由 $x_0\in range\left(A^{\mathrm{T}}\right)$ 和 $x_*=A^{\dagger}b$，故 $x_k-x_*\in range\left(A^{\mathrm{T}}\right)$，从而由引理 2 可得

$$\left\|\tilde{r}_k\right\|^2=\left\|\tilde{A}\left(x_k-x_*\right)\right\|_2^2\ge\sigma_{min}^2\left(\tilde{A}\right)\left\|x_k-x_*\right\|_2^2$$

结合上式，可得

$$\left\|\hat{A}_{\mathcal{T}_k}\left(x_k-x_*\right)\right\|_2^2\ge\eta t\frac{\sigma_{min}^2\left(\tilde{A}^{\mathrm{T}}\tilde{A}\right)}{\left\|\tilde{A}\right\|_F^2}\left\|x_k-x_*\right\|_2^2\tag{11}$$

因此，联合(11)式和(8)式，我们可得(2)式，故定理 2 得证。

Algorithm 7. SFGBK 方法

$A,b,l,x_0\in range\left(A^T\right)$, 参数 $\eta\in(0,1)$，步长序列 $\left(\alpha_k\right)_k\ge0$ 和权重序列 $\left(w_k\right)_k\ge0$

输出：x_l

初始化：由算法 3 生成 Spara Random Project 变换矩阵 $\tilde{A}=SA\in\mathbb{R}^{d\times n}$ 和向量 $\tilde{b}=Sb$, 其中 $d\ll m$

for $k=0,1,\ldots l-1$ do

计算 $\tilde{\varepsilon}_k=\eta\max\limits_{1\le i\le m}\left\{\dfrac{\left|\tilde{b}_i-\tilde{A}_{(i)x_k}\right|^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right\}$

确定指标集序列 $\mathcal{T}_k=\left\{i_k:\left|\tilde{b}_{i_k}-\tilde{A}_{(i_k)}x_k\right|^2\ge\varepsilon_k\left\|\tilde{A}_{(i_k)}\right\|_2^2\right\}$

计算 $x_{k+1}=x_k+\alpha_k\left(\sum\limits_{i\in\mathcal{T}_k}w_i\dfrac{\tilde{b}_i-\tilde{A}_{(i)}x_k}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^T\right)$

end for

Sparse Random Projection 这个稀疏随机矩阵的每一列都只有一个非零元素，该元素的位置在每一列中都是随机选择的，其值是根据一个预先定义的分布随机选择的。这种方法的优点是它的计算效率高，因为乘以稀疏矩阵的计算复杂度低。从而进行更高效的计算作为参照，第四节中的数值实验将给出 SFGBK 的数据。

本文只讨论 $\alpha_k=1$的情况，下面给出 LFGBK 方法求解大型相容线性方程组(1)的收敛性理论。

定理 3 设 Sparse Random Projection 变换 S 满足 $d=O\left(n^2/\delta\theta^2\right)$，$x_*=A^{\dagger}b$ 是相容线性方程组(1)的最小范数解，则对任意 $k\ge0$，由 SFGBK 方法生成的迭代序列 $\left\{x_k\right\}_{k=0}^{\infty}$ 满足

$$x_{k+1}-x_{*}{}_2^2\le\left(1-\tilde{\psi}_k\left(\eta\right)\frac{\sigma_{min}^2\left(A\right)}{\left\|A\right\|_2^2}\right)\left\|x_k-x_*\right\|_2^2\tag{12}$$

的概率为$1-\delta$其中$\tilde{\psi}_k(\eta)=\frac{\eta}{|\mathcal{T}_k|}\left(2|\mathcal{T}_k|-\sigma_{max}^2\left(\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\right)\right)\frac{(1-\theta)^4}{\min\{d,n\}}$，$\eta\in(0,1]$，且$\sigma_{min}(A)$和$\sigma_{max}(A)$分别表示矩阵$A$的非零最小奇异值和最大奇异值。

证明 由算法7和$\tilde{r}_k=\tilde{b}-\tilde{A}x_k$，我们可以得到

$$x_{k+1}=x_k+\left(\sum_{i\in\mathcal{T}_k}\frac{1}{|\mathcal{T}_k|}\frac{\tilde{r}_k^i\tilde{A}_{(i)}^{\mathrm{T}}}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right) \tag{13}$$

设$|\mathcal{T}_k|=t$和$\mathcal{T}_k=\left\{j_{k_1},\cdots,j_{k_t}\right\}$，将(13)式子展开可得

$$x_{k+1}=x_k+\sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}^{\mathrm{T}}e_i^{\mathrm{T}}\tilde{r}_k}{\left\|\tilde{A}_{(i)}\right\|_2^2}=x_k+\frac{1}{t}\left(\frac{\tilde{A}_{(j_{k_1})}^{\mathrm{T}}e_{j_{k_1}}^{\mathrm{T}}\tilde{r}_k}{\left\|\tilde{A}_{(j_{k_1})}\right\|_2^2}+\cdots+\frac{\tilde{A}_{(j_{k_t})}^{\mathrm{T}}e_{j_{k_t}}^{\mathrm{T}}\tilde{r}_k}{\left\|\tilde{A}_{(j_{k_t})}\right\|_2^2}\right)=x_k+\frac{1}{t}\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\hat{I}_{\mathcal{T}_k}\tilde{r} \tag{14}$$

其中

$$\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}=\left[\frac{\tilde{A}_{(j_{k_1})}^{\mathrm{T}}}{\left\|\tilde{A}_{(j_{k_1})}\right\|_2},\frac{\tilde{A}_{(j_{k_2})}^{\mathrm{T}}}{\left\|\tilde{A}_{(j_{k_2})}\right\|_2},\cdots,\frac{\tilde{A}_{(j_{k_t})}^{\mathrm{T}}}{\left\|\tilde{A}_{(j_{k_t})}\right\|_2}\right]\in\mathbb{R}^{n\times t} \tag{15}$$

和

$$\hat{I}_{\mathcal{T}_k}=\left[\frac{e_{(j_{k_1})}}{\left\|\tilde{A}_{(j_{k_1})}\right\|_2},\frac{e_{(j_{k_2})}}{\left\|\tilde{A}_{(j_{k_2})}\right\|_2},\cdots,\frac{e_{(j_{k_t})}}{\left\|\tilde{A}_{(j_{k_t})}\right\|_2}\right]^{\mathrm{T}}\in\mathbb{R}^{t\times d} \tag{16}$$

(14)式同时减去$\boldsymbol{x}_*$，可得

$$x_{k+1}-x_*=x_k-x_*-\frac{1}{t}\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\hat{I}_{\mathcal{T}_k}\tilde{A}(x_k-x_*)=\left(I-\frac{1}{t}\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\hat{A}_{\mathcal{T}_k}\right)(x_k-x_*) \tag{17}$$

对(17)式两边同时取谱范数并平方，又对任意半正定矩阵$Q\succeq 0$满足$Q^2\preceq\lambda_{max}(Q)Q$，从而可以得

$$x_{k+1}-x_{*2}^{\;2}\le\left\|x_k-x_*\right\|_2^2-\left(\frac{2}{t}-\frac{1}{t^2}\sigma_{max}^2\left(\hat{A}_{\mathcal{T}_k}^{\mathrm{T}}\right)\right)\left\|\hat{A}_{\mathcal{T}_k}(x_k-x_*)\right\|_2^2 \tag{18}$$

由(15)式和$\left|\tilde{b}_{j_k}-\tilde{A}_{(j_k)}x_k\right|^2\ge\left\|\tilde{\varepsilon}_k\tilde{A}_{(j_k)}\right\|_2^2$，通过简单计算可得

$$\left\|\tilde{A}_{\mathcal{T}_k}(x_k-x_*)\right\|_2^2=\sum_{j_k\in\mathcal{T}_k}\frac{1}{\left\|\tilde{A}_{(j_k)}\right\|_2^2}\left|\tilde{r}_k^{j_k}\right|^2\ge\sum_{j_k\in\mathcal{T}_k}\frac{1}{\left\|\tilde{A}_{(j_k)}\right\|_2^2}\tilde{\varepsilon}_k\left\|\tilde{A}_{(j_k)}\right\|_2^2 \tag{19}$$

把$\tilde{\varepsilon}_k=\eta\max\limits_{1\le i\le d}\left\{\frac{\left|\tilde{b}_i-\tilde{A}_{(i)x_k}\right|^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right\}$带入(19)式，可得

$$\left\|\hat{A}_{\mathcal{T}_k}(x_k-x_*)\right\|_2^2\ge\sum_{j_k\in\mathcal{T}_k}\eta\max_{1\le i\le d}\left\{\frac{\left|\tilde{r}_k^i\right|^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right\}\ge t\eta\sum_{i=1}^{d}\frac{\left|\tilde{r}_k^i\right|^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{\left\|\tilde{A}_{(i)}\right\|_2^2}{\left\|\tilde{A}\right\|_F^2}=\eta t\frac{\left\|\tilde{r}_k\right\|^2}{\left\|\tilde{A}\right\|_F^2} \tag{20}$$

由 $x_0 \in range\left(A^{\mathrm{T}}\right)$ 和 $x_* = A^{\dagger}b$，故 $x_k - x_* \in range\left(A^{\mathrm{T}}\right)$，从而由引理 3 可得

$$\left\|\tilde{r}_k\right\|^2 = \left\|\tilde{A}\left(x_k - x_*\right)\right\|_2^2 \ge (1-\theta)^2\left\|A\left(x_k - x_*\right)\right\|_2^2 \ge (1-\theta)^2 \sigma_{min}^2(A)\left\|x_k - x_*\right\|_2^2 \tag{21}$$

成立的概率为 $1-\delta$，结合上式，可得

$$\left\|\hat{A}_{\mathcal{T}_k}\left(x_k - x_*\right)\right\|_2^2 \ge \eta t \frac{(1-\theta)^2 \sigma_{min}^2\left(\tilde{A}^{\mathrm{T}}\tilde{A}\right)}{\left\|\tilde{A}\right\|_F^2}\left\|x_k - x_*\right\|_2^2 \tag{22}$$

成立的概率为 $1-\delta$，又 $\left\|\tilde{A}\right\|_F^2 \le \min\{d,n\}\left\|\tilde{A}\right\|_2^2$，则由引理 3 可得 $\left\|\tilde{A}\right\|_F^2 \le \min\{d,n\}\left\|\tilde{A}\right\|_2^2 \le \min\{d,n\}\frac{\|A\|_2^2}{(1-\theta)^2}$ 成立的概率为 $1-\delta$。故

$$\left\|\hat{A}_{\mathcal{T}_k}\left(x_k - x_*\right)\right\|_2^2 \ge \eta t \frac{(1-\theta)^4 \sigma_{min}^2(A)}{\min\{d,n\}\|A\|_2^2}\left\|x_k - x_*\right\|_2^2 \tag{23}$$

成立的概率至少为 $1-\delta$。

因此我们可得(12)式成立的概率为 $1-\delta$。故定理 3 得证。

Algorithm 8. CMFGBK 方法

1：$A, b, l, x_0 \in range\left(A^{\mathrm{T}}\right)$，参数 $\eta \in (0,1)$，步长序列 $(\alpha_k)_k \ge 0$ 和权重序列 $(w_k)_k \ge 0$

2：输出：x_l

3：初始化：有算法 1 生成 CountSketch 变换矩阵 $\tilde{A} = SA \in \mathbb{R}^{d\times n}$ 和向量 $\tilde{b} = Sb$，其中 $d \ll m$

4：for $k = 0,1,\cdots,l-1$ do

5：计算 $\varepsilon_k = \eta \max\limits_{1\le i\le d}\left\{\frac{\left|\tilde{b}_i - \tilde{A}_{(i)}x_k\right|^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right\}$

6：确定指标集序列 $\mathcal{T}_k = \left\{i_k : \left|\tilde{b}_{ik} - \tilde{A}_{(i_k)}x_k\right|^2 \ge \varepsilon_k\left\|\tilde{A}_{(i_k)}\right\|_2^2\right\}$

7：计算 $x_{k+1} = x_k + \sum\limits_{i\in\mathcal{T}_k} w_i \frac{\tilde{b}_i - \tilde{A}_{(i)}x_k}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}} + \alpha\left(x_k - x_{k-1}\right)$

8：end for

定理 4 设 CountSketch 变换 S 满足 $d = O\left(n^2/\delta\theta^2\right)$，$x_* = A^{\dagger}b$ 是相容线性方程组(1)的最小范数解，则对任意 $x_0 \in range\left(A^{\mathrm{T}}\right)$ 由算法 8 生成迭代序列 $\{x_k\}_{k=0}^{\infty}$ 收敛到方程组最小范数解 x_* 且对任意 $k \ge 0$ 满足

$$\left\|x_k - x^*\right\|_2^2 \le q^t(1+\delta)\left\|x_0 - x^*\right\|_2^2$$

的概率为 $1-\delta$，其中 $4\alpha^2 + 4\alpha + \left(\frac{1}{t^2} - \frac{2}{t} - \alpha + \alpha\frac{1}{t}\right)\sum\limits_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2} < 0, \alpha > 0$，

$a_1 = 1 + 2\alpha^2 + 3\alpha + \left(\frac{1}{t^2} - \frac{2}{t} - \alpha\right)\sum\limits_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2}$，$a_2 = 2\alpha^2 + \alpha + \alpha\frac{1}{t}\sum\limits_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2}$，$q = \frac{a_1 + \sqrt{a_1^2 + 4a_1}}{2}$，

$\delta = q - a_1$ 和 $a_1 + a_2 \le q < 1$，有 $q \in (0,1)$。

证明 由算法 8，我们可以得到

$$x_{k+1}=x_k+\sum_{i\in\mathcal{T}_k}\frac{1}{|\mathcal{T}_k|}\frac{\tilde{b}_i-\tilde{A}_{(i)}x_k}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}+\alpha\left(x_k-x_{k-1}\right)$$

设 $|\mathcal{T}_k|=t$ 和 $\mathcal{T}_k=\left\{j_{k_1},\cdots,j_{k_t}\right\}$，其中 t 表示指标集 $\mathcal{T}_k$ 中元素的个数，可得

$$x_{k+1}=x_k-\sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k-\tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}+\alpha\left(x_k-x_{k-1}\right)$$

上式同时减去 x_*，可得

$$x_{k+1}-x^*=x_k-\sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k-\tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}+\alpha\left(x_k-x_{k-1}\right)-x^*$$

对上式两边同时取谱范数并平方，可得

$$\left\|x_{k+1}-x_*\right\|_2^2=\left\|x_k-x^*-\sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k-\tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}\right\|_2^2+\alpha^2\left\|x_k-x_{k-1}\right\|_2^2+2\alpha\left\langle x_k-x^*-\sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k-\tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}},x_k-x_{k-1}\right\rangle \tag{24}$$

由 Kaczmarz 收敛论证和 $b_i=A_{(i)}x^*$，对等式(24)的第一个项给出下界，可得

$$\left\|x_k-x^*-\sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k-\tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}\right\|_2^2=\left\|x_k-x_*\right\|_2^2+\left(\frac{1}{t^2}-\frac{2}{t}\right)\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}x_k-\tilde{b}_i\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}$$

由首项加上并减去 x^* 和 $\|a+b\|^2\le 2\|a\|^2+2\|b\|^2$，对等式(24)的第二个项给出下界，可得

$$\alpha^2\left\|x_k-x_{k-1}\right\|_2^2=\alpha^2\left\|\left(x_k-x^*\right)+\left(x^*-x_{k-1}\right)\right\|_2^2\le 2\alpha^2\left\|x_k-x^*\right\|_2^2+2\alpha^2\left\|x_{k-1}-x^*\right\|_2^2$$

等式(24)的第三个项给出下界，可得

$$\begin{aligned}&2\alpha\left\langle x_k-x^*-\sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k-\tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}},x_k-x_{k-1}\right\rangle\\&\le\left\|\alpha x_k-x^*\right\|^2+\alpha\left\|x_k-x_{k-1}\right\|^2-\alpha\left\|x_{k-1}-x^*\right\|^2-\alpha\left\langle\sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k-\tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}},x_t-x^*\right\rangle\\&\quad+\alpha\left\langle\sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_{k-1}-\tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}A_{(i)}^{\mathrm{T}},x_{k-1}-x^*\right\rangle\end{aligned}$$

结合三个下界，简化内积并归类相似项，可得：

$$\left\|x_{k+1}-x_*\right\|_2^2\le\left(1+2\alpha^2+3\alpha+\left(\frac{1}{t^2}-\frac{2}{t}-\alpha\right)\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right)\left\|x_k-x_*\right\|_2^2+\left(2\alpha^2+\alpha+\alpha\frac{1}{t}\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right)\left\|x_{k-1}-x^*\right\|_2^2$$

由引理 3，可得

$$\left(1+2\alpha^2+3\alpha+\left(\frac{1}{t^2}-\frac{2}{t}-\alpha\right)\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right)\left\|x_k-x_*\right\|_2^2+\left(2\alpha^2+\alpha+\alpha\frac{1}{t}\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right)\left\|x_{k-1}-x^*\right\|_2^2$$

$$\le\left(1+2\alpha^2+3\alpha+\left(\frac{1}{t^2}-\frac{2}{t}-\alpha\right)\sum_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2}\right)\left\|x_k-x_*\right\|_2^2$$

$$+\left(2\alpha^2+\alpha+\alpha\frac{1}{t}\sum_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2}\right)\left\|x_{k-1}-x^*\right\|_2^2$$

成立的概率为$1-\delta$，结合上式可得

$$\left\|x_{k+1}-x_*\right\|_2^2\le\left(1+2\alpha^2+3\alpha+\left(\frac{1}{t^2}-\frac{2}{t}-\alpha\right)\sum_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2}\right)\left\|x_k-x_*\right\|_2^2$$

$$+\left(2\alpha^2+\alpha+\alpha\frac{1}{t}\sum_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2}\right)\left\|x_{k-1}-x^*\right\|_2^2$$

最后，由引理 1，其中两个系数为$a_1=1+2\alpha^2+3\alpha+\left(\frac{1}{t^2}-\frac{2}{t}-\alpha\right)\sum_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2}$和$a_2=2\alpha^2+\alpha+\alpha\frac{1}{t}\sum_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2}$，由于我们假设$a_1+a_2<1$和$\alpha>0$，由$a_2>0$，可得

$$\left\|x_k-x^*\right\|_2^2\le q^t\left(1+\delta\right)\left\|x_0-x^*\right\|_2^2$$

其中，$q=\frac{a_1+\sqrt{a_1^2+4a_1}}{2}$，$\delta=q-a_1$和$a_1+a_2\le q<1$，有$q\in(0,1)$，可得我们证明了 CMFGBK 算法的收敛性。

Algorithm 9. LMFGBK 方法

1：$A,b,l,x_0\in range\left(A^{\mathrm{T}}\right)$，参数$\eta\in(0,1)$，步长序列$\left(\alpha_k\right)_k\ge0$和权重序列$\left(w_k\right)_k\ge0$
2：输出：x_l
3：初始化：有算法 2 生成 Leverage Score sampling 变换矩阵$\tilde{A}=SA\in\mathbb{R}^{d\times n}$和向量$\tilde{b}=Sb$，其中$d\ll m$
4：for $k=0,1,\cdots,l-1$ do
5：计算$\varepsilon_k=\eta\max_{1\le i\le m}\left\{\frac{\left|b_i-A_{(i)x_k}\right|^2}{\left\|A_{(i)}\right\|_2^2}\right\}$
6：确定指标集序列$\mathcal{T}_k=\left\{i_k:\left|b_{ik}-A_{(i_k)}x_k\right|^2\ge\varepsilon_k\left\|A_{(i_k)}\right\|_2^2\right\}$
7：$v=\alpha v+(1-\alpha)\left(\sum_{i\in\mathcal{T}_k}w_i\frac{b_i-A_{(i)}x_k}{\left\|A_{(i)}\right\|_2^2}A_{(i)}^{\mathrm{T}}\right)$
8：计算$x_{k+1}=x_k+v$
9：end for

定理 5 设 Leverage Score 变换 S 满足 $d = O\left(n^2/\delta\theta^2\right)$，$x_* = A^{\dagger}b$ 是相容线性方程组(1)的最小范数解和则对任意 $x_0 \in range\left(A^{\mathrm{T}}\right)$ 由算法 9 生成迭代序列 $\{x_k\}_{k=0}^{\infty}$ 收敛到方程组的最小范数解 x_* 且对任意 $k \ge 0$ 满足

$$\left\|x_k - x^*\right\|_2^2 \le q^t\left(1+\delta\right)\left\|x_0 - x^*\right\|_2^2$$

其中，$4\alpha^2 + 4\alpha + \left(\frac{1}{t^2} - \frac{2}{t} - \alpha + \alpha\frac{1}{t}\right)\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2} < 0, \alpha > 0$，$a_1 = 1 + 2\alpha^2 + 3\alpha + \left(\frac{1}{t^2} - \frac{2}{t} - \alpha\right)\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}$，$a_2 = 2\alpha^2 + \alpha + \alpha\frac{1}{t}\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}$，$q = \frac{a_1 + \sqrt{a_1^2 + 4a_1}}{2}$，$\delta = q - a_1$ 和 $a_1 + a_2 \le q < 1$，有 $q \in (0,1)$，可得我们证明了 LMFGBK 算法的收敛性。

证明 由算法 9，我们可以得到

$$x_{k+1} = x_k + \sum_{i\in\mathcal{T}_k}\frac{1}{\left|\mathcal{T}_k\right|}\frac{\tilde{b}_i - \tilde{A}_{(i)}x_k}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}} + \alpha\left(x_k - x_{k-1}\right)$$

设 $\left|\mathcal{T}_k\right| = t$ 和 $\mathcal{T}_k = \left\{j_{k_1}, \cdots, j_{k_t}\right\}$，其中 t 表示指标集 $\mathcal{T}_k$ 中元素的个数，可得

$$x_{k+1} = x_k - \sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k - \tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}} + \alpha\left(x_k - x_{k-1}\right)$$

上式同时减去 x_*，可得

$$x_{k+1} - x^* = x_k - \sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k - \tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}} + \alpha\left(x_k - x_{k-1}\right) - x^*$$

对上式两边同时取谱范数并平方，可得

$$\left\|x_{k+1} - x_*\right\|_2^2 = \left\|x_k - x^* - \sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k - \tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}\right\|_2^2 + \alpha^2\left\|x_k - x_{k-1}\right\|_2^2 + 2\alpha\left\langle x_k - x^* - \sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k - \tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}, x_k - x_{k-1}\right\rangle \quad (25)$$

由 Kaczmarz 收敛论证和 $b_i = A_{(i)}x^*$，对等式(25)的第一个项给出下界，可得

$$\left\|x_k - x^* - \sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k - \tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}\right\|_2^2 = \left\|x_k - x_*\right\|_2^2 + \left(\frac{1}{t^2} - \frac{2}{t}\right)\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}x_k - \tilde{b}_i\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}$$

由首项加上并减去 x^* 和 $\|a+b\|^2 \le 2\|a\|^2 + 2\|b\|^2$，对等式(25)的第二个项给出下界，可得

$$\alpha^2\left\|x_k - x_{k-1}\right\|_2^2 = \alpha^2\left\|\left(x_k - x^*\right) + \left(x^* - x_{k-1}\right)\right\|_2^2 \le 2\alpha^2\left\|x_k - x^*\right\|_2^2 + 2\alpha^2\left\|x_{k-1} - x^*\right\|_2^2$$

等式(25)的第三个项给出下界，可得

$$2\alpha\left\langle x_k - x^* - \sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k - \tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}, x_k - x_{k-1}\right\rangle$$

$$= 2\alpha\left\langle x_k - x^*, x_k - x_{k-1}\right\rangle + 2\alpha\left\langle \sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k - \tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}, x_{k-1} - x_k\right\rangle$$

$$\le \alpha\left\|x_k - x^*\right\|^2 + \alpha\left\|x_k - x_{k-1}\right\|^2 - \alpha\left\|x_{k-1} - x^*\right\|^2 - \alpha\left\langle \sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_k - \tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}\tilde{A}_{(i)}^{\mathrm{T}}, x_t - x^*\right\rangle$$

$$+\alpha\left\langle \sum_{i\in\mathcal{T}_k}\frac{1}{t}\frac{\tilde{A}_{(i)}x_{k-1} - \tilde{b}_i}{\left\|\tilde{A}_{(i)}\right\|_2^2}A_{(i)}^{\mathrm{T}}, x_{k-1} - x^*\right\rangle$$

结合三个下界，简化内积并归类相似项，可得：

$$\left\|x_{k+1} - x_*\right\|_2^2 \le \left(1 + 2\alpha^2 + 3\alpha + \left(\frac{1}{t^2} - \frac{2}{t} - \alpha\right)\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right)\left\|x_k - x_*\right\|_2^2 + \left(2\alpha^2 + \alpha + \alpha\frac{1}{t}\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\right)\left\|x_{k-1} - x^*\right\|_2^2$$

最后，由引理 1，其中两个系数为 $a_1 = 1 + 2\alpha^2 + 3\alpha + \left(\frac{1}{t^2} - \frac{2}{t} - \alpha\right)\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}$ 和 $a_2 = 2\alpha^2 + \alpha + \alpha\frac{1}{t}\sum_{i\in\mathcal{T}_k}\frac{\left(\tilde{A}_{(i)}\right)^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}$，由于我们假设 $a_1 + a_2 < 1$ 和 $\alpha > 0$，由 $a_2 > 0$，可得

$$\left\|x_k - x^*\right\|_2^2 \le q^t\left(1 + \delta\right)\left\|x_0 - x^*\right\|_2^2$$

其中，$q = \frac{a_1 + \sqrt{a_1^2 + 4a_1}}{2}$，$\delta = q - a_1$ 和 $a_1 + a_2 \le q < 1$，有 $q \in (0,1)$，可得我们证明了 LMFGBK 算法的收敛性。

定理 6 设 Sparse Random Projection 变换 S 满足 $d = O\left(n^2/\delta\theta^2\right)$，$x_* = A^\dagger b$ 是相容线性方程组(1)的最小范数解，则对任意 $x_0 \in range\left(A^{\mathrm{T}}\right)$ 由算法 10 生成迭代序列 $\left\{x_k\right\}_{k=0}^{\infty}$ 收敛到方程组的最小范数解 x_* 且对任意 $k \ge 0$ 满足

$$\left\|x_k - x^*\right\|_2^2 \le q^t\left(1 + \delta\right)\left\|x_0 - x^*\right\|_2^2$$

的概率为 $1-\delta$，其中 $4\alpha^2 + 4\alpha + \left(\frac{1}{t^2} - \frac{2}{t} - \alpha + \alpha\frac{1}{t}\right)\sum_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2} < 0, \alpha > 0$，

$a_1 = 1 + 2\alpha^2 + 3\alpha + \left(\frac{1}{t^2} - \frac{2}{t} - \alpha\right)\sum_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2}$，$a_2 = 2\alpha^2 + \alpha + \alpha\frac{1}{t}\sum_{i\in\mathcal{T}_k}\frac{A^2}{\left\|\tilde{A}_{(i)}\right\|_2^2}\frac{1}{(1-\theta)^2}$，$q = \frac{a_1 + \sqrt{a_1^2 + 4a_1}}{2}$，$\delta = q - a_1$ 和 $a_1 + a_2 \le q < 1$，有 $q \in (0,1)$，算法 8 证明类似，同理可得 SMFGBK 算法的收敛性。

定理 4~6 表明应用重力球技术的 Kaczmarz 算法(即本研究提出的算法 CFMGBK、LFMGBK 和 SFMGBK)呈指数级收敛，其收敛速度超过传统的 CFGBK 算法。

Algorithm 10. SMFGBK 方法

$A, b, l, x_0 \in range(A^{\mathrm{T}})$，参数 $\eta \in (0,1)$，步长序列 $(\alpha_k)_k \geq 0$ 和权重序列 $(w_k)_k \geq 0$

输出：x_l

初始化：有算法 3 生成 Spara Random Project 变换矩阵 $\tilde{A} = SA \in \mathbb{R}^{d \times n}$ 和向量 $\tilde{b} = Sb$，其中 $d \ll m$

for $k = 0, 1, \cdots, l-1$ do

计算 $\varepsilon_k = \eta \max\limits_{1 \leq i \leq m} \left\{ \dfrac{\left| b_i - A_{(i)x_k} \right|^2}{\left\| A_{(i)} \right\|_2^2} \right\}$

确定指标集序列 $\mathcal{T}_k = \left\{ i_k : \left| b_{ik} - A_{(i_k)} x_k \right|^2 \geq \varepsilon_k \left\| A_{(i_k)} \right\|_2^2 \right\}$

$v = \alpha v + (1-\alpha) \left(\sum\limits_{i \in \mathcal{T}_k} w_i \dfrac{b_i - A_{(i)} x_k}{\left\| A_{(i)} \right\|_2^2} A_{(i)}^{\mathrm{T}} \right)$

计算 $x_{k+1} = x_k + v$

end for

4. 数值实验

本节，我们通过几组数值算例来比较 GBK 方法、改进 FGBK 方法和 CFGBK 方法求[17]解大型相容线性方程组(1)的有效性。所有实验均通过 MATLAB 编程实现，IT 和 CPU 分别表示迭代步数和计算时间(单位：秒)。类似于文献，IT 和 CPU 的取值均为 50 次重复运行所需要的迭代步数和计算时间的平均值。在所有的计算过程中，我们令初始向量 $x_0 = zeros(n,1)$ 和右端项 $b = Ax$，其中 x 为相容线性方程组的解向量并由 MATLAB 函数 *randn* 生成。置停机准则为 $RSE = \dfrac{\|x_k - x_*\|_2^2}{\|x_*\|_2^2} \leq 10^{-6}$，在实际计算中条件 $O\left(n^2/\delta\theta^2\right)$ 非常严格，在很多实际问题计算过程中 Sketching 因子 $d = n^2$ 可以取得很好的数值效果。我们考虑的系数矩阵 $A \in \mathbb{R}^{m \times n}$。类型 a：$A = randn(m,n)$。类型 b：$A = UDV^{\mathrm{T}}$，其中 $U \in \mathbb{R}^{m \times r}$，$V \in \mathbb{R}^{n \times r}$ 和 $r = n$，且 U, V 和 D 分别由 $[U,\sim] = qr(randn(m,r),0), [V,\sim] = qr(randn(n,r),0)$ 和 $D = diag(1 + (k-1) * rand(r,1))$ 生成，易计算系数矩阵 A 的条件数的上界为 k。

Table 1. Numerical experimental results of FGBK, CFGBK, LFGBK, and SFGBK when $A = randn(m,n)$

表 1. $A = randn(m,n)$ **时，FGBK、CFGBK、LFGBK 和 SFGBK 的数值实验结果**

		30000*50		30000*100		50000*50		50000*100		100000*50		100000*100	
η		0.8	0.9	0.8	0.9	0.8	0.9	0.8	0.9	0.8	0.9	0.8	0.9
FGBK	IT	35	30	81	79	32	28	73	70	29	25	64	62
	CPU	1.495	2.046	3.919	4.253	2.271	5.598	5.824	6.892	4.126	4.008	10.300	10.045
CFGBK	IT	65	51	NA	NA	65	51	NA	NA	64	51	NA	NA
	CPU	0.242	0.182	NA	NA	0.248	0.648	NA	NA	0.486	0.659	NA	NA
LFGBK	IT	66	52	111	106	65	51	109	104	64	51	108	103
	CPU	0.26	0.209	1.912	1.828	0.274	0.223	1.892	1.791	0.302	0.3414	1.904	1.835
SFGBK	IT	64	51	106	102	65	51	106	102	64	51	107	103
	CPU	0.291	0.240	2.039	1.951	0.306	0.254	2.174	2.035	0.5229	0.6308	2.254	2.203

Table 2. Numerical experimental results of FGBK, CFGBK, LFGBK, and SFGBK when $A = UDV^{\mathrm{T}}$, $k = 1.5$
表 2. $A = UDV^{\mathrm{T}}$ 时，FGBK、CFGBK、LFGBK 和 SFGBK 的数值实验结果，$k = 1.5$

		30000*50		30000*100		50000*50		50000*100		100000*50		100000*100	
η		0.8	0.9	0.8	0.9	0.8	0.9	0.8	0.9	0.8	0.9	0.8	0.9
FGBK	IT	39	34	93	89	36	31	83	81	32	28	74	72
	CPU	2.504	2.150	6,625	6.382	3.834	3.316	10.732	10.551	7.028	6.049	17.654	17.118
CFGBK	IT	72	54	NA	NA	72	56	NA	NA	72	56	NA	NA
	CPU	0.467	0.298	NA	NA	0.391	0.841	NA	NA	0.404	0.323	NA	NA
LFGBK	IT	73	57	125	118	72	56	124	118	72	57	122	118
	CPU	0.403	0.320	3.071	2.901	0.417	0.332	3.039	5.478	0.603	0.775	3.112	2.948
SFGBK	IT	72	57	121	116	72	56	121	116	72	56	122	115
	CPU	0.457	0.363	3.231	3.106	0.474	0.377	3.298	3.170	0.523	0.424	3.572	3.395

从表 1 和表 2 的数值结果，我们可得如下结论：1) FGBK 方法、CFGBK 方法、LFGBK 方法和 SFGBK 方法都是有效的。2) FGBK 方法的迭代步数比 CFGBK 方法、LFGBK 方法和 SFGBK 方法的迭代步数少，但是在计算时间上，CFGBK 方法、LFGBK 方法和 SFGBK 方法更优于 FGBK 方法。3) 在问题类型 a 和问题类型 b 中，CFGBK 方法在运行过程中出现了 NA 报错，在鲁棒性上，FGBK 方法、LFGBK 方法和 SFGBK 方法更优于 CFGBK 方法。4) $\eta = 0.9$ 时加速效果要优于 $\eta = 0.8$ 时的情形。

Table 3. Numerical experimental results for LFGBK, CFMGBK, LFMGBK, and SFMGBK when $A = randn(m, n)$
表 3. $A = randn(m, n)$ 时，LFGBK、CFMGBK、LFMGBK 和 SFMGBK 的数值实验结果

$\eta = 0.8$		30000*50		30000*100		50000*50		50000*100		100000*50		100000*100	
冲量 α		0.3	0.7	0.3	0.7	0.3	0.7	0.3	0.7	0.3	0.7	0.3	0.7
LFGBK	IT	66		111		65		109		64		108	
	CPU	0.26		1.912		0.274		1.892		0.302		1.904	
CFMGBK	IT	40	56	NA	NA	40	57	NA	NA	40	58	NA	NA
	CPU	0.153	0.209	NA	NA	0.156	0.219	NA	NA	0.167	0.230	NA	NA
LFMGBK	IT	41	57	74	56	41	59	72	56	41	57	71	54
	CPU	0.205	0.243	1.299	0.933	0.186	0.2504	1.876	1.021	0.224	0.278	1.349	1.485
SFMGBK	IT	41	56	71	53	41	57	70	56	40	57	71	53
	CPU	0.203	0.256	1.435	1.195	0.220	0.280	1.519	1.282	0.255	0.316	1.744	1.399

Table 4. Numerical experimental results for LFGBK, CFMGBK, LFMGBK, and SFMGBK when $A = UDV^{\mathrm{T}}$, $k = 1.5$
表 4. $A = UDV^{\mathrm{T}}$ 时，LFGBK、CFMGBK、LFMGBK 和 SFMGBK 的数值实验结果，$k = 1.5$

$\eta = 0.8$		30000*50		30000*100		50000*50		50000*100		100000*50		100000*100	
冲量 α		0.3	0.7	0.3	0.7	0.3	0.7	0.3	0.7	0.3	0.7	0.3	0.7
LFGBK	IT	73		125		72		124		72		122	
	CPU	0.403		3.071		0.417		3.039		0.603		3.112	
CFMGBK	IT	47	53	NA	NA	47	57	NA	NA	47	57	NA	NA
	CPU	0.259	0.282	NA	NA	0.266	0.315	NA	NA	0.277	0.326	NA	NA

续表

LFMGBK	IT	47	57	84	57	47	59	83	58	47	57	82	54
	CPU	0.277	0.331	2.414	1.412	0.288	0.347	2.027	1.474	0.316	0.367	2.080	1.478
SFMGBK	IT	47	56	83	53	46	59	81	62	48	59	81	52
	CPU	0.226	0.245	2.590	1.768	0.331	0.399	2.356	1.726	0.376	0.437	2.768	2.038

从表 3 和表 4 的数值结果，我们可得如下结论：1) CFMGBK 方法、LFMGBK 方法和 SFMGBK 方法求解大型相容线性方程组都是有效的。2) CFMGBK 方法、LFMGBK 方法和 SFMGBK 方法在迭代步数和计算时间上，CFMGBK 方法、LFMGBK 方法更优于 LFGBK 方法。3) 在问题类型 a 和问题类型 b 中，CFMGBK 方法在运行过程中出现了 NA 报错，在鲁棒性上，SFMGBK 方法、LFMGBK 方法更优于 CFMGBK 方法。4) $\alpha = 0.7$ 时在某些情况下加速效果优于 $\alpha = 0.3$，α 选取问题属于 NP 难题范畴，如何确定其最优解是一项极具挑战性的任务。本研究仅提供初步尝试，具体的取值策略尚待解决，这将是未来研究的重点方向。

5. 总结

本文提出了求解大型相容线性方程组的快速免伪逆贪婪块 Kaczmarz (LFGBK、CFGBK)方法，并给出了该方法的收敛性理论。为进一步改进快速免伪逆贪婪块 Kaczmarz 方法的收敛性，使用重力球技术，本文建立了一类求解大型相容线性方程组的加速快速免伪逆贪婪块 Kaczmarz 方法，并对该类方法的收敛性进行了详细分析。数值实验验证了新方法的有效性。

参考文献

[1] Byrne, C. (2003) A Unified Treatment of Some Iterative Algorithms in Signal Processing and Image Reconstruction. *Inverse Problems*, **20**, 103-120. https://doi.org/10.1088/0266-5611/20/1/006

[2] Chung, J., Chung, M., Slagel, J.T., *et al.* (2020) Sampled Limited Memory Methods for Massive Linear Inverse Problems. *Inverse Problems*, **36**, Article ID: 054001. https://doi.org/10.1088/1361-6420/ab77da

[3] Leskovec, J., Rajaraman, A. and Ullman, J.D. (2020) Mining of Massive Data Sets. Cambridge University Press, Cambridge. https://doi.org/10.1017/9781108684163

[4] Herman, G.T. and Davidi, R. (2008) Image Reconstruction from a Small Number of Projections. *Inverse Problems*, **24**, Article ID: 045011. https://doi.org/10.1088/0266-5611/24/4/045011

[5] Lyu, H., Sha, N., Qin, S., *et al.* (2019) Advances in Neural Information Processing Systems.

[6] Kaczmarz, S. (1937) Angenäherte Auflösung von Systemen Linearer Gleichungen. *Bulletin L'Académie Polonaise des Science*, **35**, 355-357.

[7] Deutsch, F. and Hundal, H. (1997) The Rate of Convergence for the Method of Alternating Projections, II. *Journal of Mathematical Analysis and Applications*, **205**, 381-405. https://doi.org/10.1006/jmaa.1997.5202

[8] Galántai, A. (2005) On the Rate of Convergence of the Alternating Projection Method in Finite Dimensional Spaces. *Journal of Mathematical Analysis and Applications*, **310**, 30-44. https://doi.org/10.1016/j.jmaa.2004.12.050

[9] Strohmer, T. and Vershynin, R. (2009) A Randomized Kaczmarz Algorithm with Exponential Convergence. *Journal of Fourier Analysis and Applications*, **15**, 262-278. https://doi.org/10.1007/s00041-008-9030-4

[10] Needell, D. (2010) Randomized Kaczmarz Solver for Noisy Linear Systems. *BIT Numerical Mathematics*, **50**, 395-403. https://doi.org/10.1007/s10543-010-0265-5

[11] Gower, R.M. and Richtárik, P. (2015) Randomized Iterative Methods for Linear Systems. *SIAM Journal on Matrix Analysis and Applications*, **36**, 1660-1690. https://doi.org/10.1137/15M1025487

[12] Ma, A., Needell, D. and Ramdas, A. (2015) Convergence Properties of the Randomized Extended Gauss—Seidel and Kaczmarz Methods. *SIAM Journal on Matrix Analysis and Applications*, **36**, 1590-1604. https://doi.org/10.1137/15M1014425

[13] Richtárik, P. and Takác, M. (2017) Stochastic Reformulations of Linear Systems: Accelerated Method.

[14] Liu, J. and Wright, S. (2016) An Accelerated Randomized Kaczmarz Algorithm. *Mathematics of Computation*, **85**, 153-178. https://doi.org/10.1090/mcom/2971

[15] Bai, Z.-Z. and Wu, W.-T. (2018) On Relaxed Greedy Randomized Kaczmarz Methods for Solving Large Sparse Linear Systems. *Applied Mathematics Letters*, **83**, 21-26. https://doi.org/10.1016/j.aml.2018.03.008

[16] Zhang, J. and Guo, J. (2020) On Relaxed Greedy Randomized Coordinate Descent Methods for Solving Large Linear Least-Squares Problems. *Applied Numerical Mathematics*, **157**, 372-384. https://doi.org/10.1016/j.apnum.2020.06.014

[17] Bai, Z.-Z. and Wu, W.-T. (2018) On Greedy Randomized Kaczmarz Method for Solving Large Sparse Linear Systems. *SIAM Journal on Scientific Computing*, **40**, A592-A606. https://doi.org/10.1137/17M1137747

[18] Du, K. and Gao, H. (2019) A New Theoretical Estimate for the Convergence Rate of the Maximal Weighted Residual Kaczmarz Algorithm. *Numerical Mathematics*: *Theory*, *Methods and Applications*, **12**, 627-639. https://doi.org/10.4208/nmtma.OA-2018-0039

[19] Elfving, T. (1980) Block-Iterative Methods for Consistent and Inconsistent Linear Equations. *Numerische Mathematik*, **35**, 1-12. https://doi.org/10.1007/BF01396365

[20] Needell, D. and Tropp, J.A. (2014) Paved with Good Intentions: Analysis of a Randomized Block Kaczmarz Method. *Linear Algebra and Its Applications*, **441**, 199-221. https://doi.org/10.1016/j.laa.2012.12.022

[21] Needell, D., Zhao, R. and Zouzias, A. (2015) Randomized Block Kaczmarz Method with Projection for Solving Least Squares. *Linear Algebra and Its Applications*, **484**, 322-343. https://doi.org/10.1016/j.laa.2015.06.027

[22] Niu, Y.-Q. and Zheng, B. (2020) A Greedy Block Kaczmarz Algorithm for Solving Large-Scale Linear Systems. *Applied Mathematics Letters*, **104**, Article ID: 106294. https://doi.org/10.1016/j.aml.2020.106294

[23] Rebrova, E. and Needell, D. (2021) On Block Gaussian Sketching for the Kaczmarz Method. *Numerical Algorithms*, **86**, 443-473. https://doi.org/10.1007/s11075-020-00895-9

[24] Necoara, I. (2019) Faster Randomized Block Kaczmarz Algorithms. *SIAM Journal on Matrix Analysis and Applications*, **40**, 1425-1452. https://doi.org/10.1137/19M1251643

[25] Moorman, J.D., Tu, T.K., Molitor, D., *et al*. (2021) Randomized Kaczmarz with Averaging. *BIT Numerical Mathematics*, **61**, 337-359. https://doi.org/10.1007/s10543-020-00824-1

[26] Du, K., Si, W.-T. and Sun, X.-H. (2020) Randomized Extended Average Block Kaczmarz for Solving Least Squares. *SIAM Journal on Scientific Computing*, **42**, A3541-A3559. https://doi.org/10.1137/20M1312629

[27] Zouzias, A. and Freris, N.M. (2013) Randomized Extended Kaczmarz for Solving Least Squares. *SIAM Journal on Matrix Analysis and Applications*, **34**, 773-793. https://doi.org/10.1137/120889897

[28] Du, K. and Sun, X.-H. (2020) Pseudoinverse-Free Randomized Block Iterative Algorithms for Consistent and Inconsistent Linear Systems.

[29] 张建华, 郭静慧. 一类快速免伪逆贪婪块 KACZMARZ 方法[J]. 高等学校计算数学学报, 2022, 44(3): 285-298.

[30] Chen, K., Li, Q., Newton, K., *et al*. (2020) Structured Random Sketching for PDE Inverse Problems. *SIAM Journal on Matrix Analysis and Applications*, **41**, 1742-1770. https://doi.org/10.1137/20M1310497

[31] Pilanci, M. and Wainwright, M.J. (2017) Newton Sketch: A Near Linear-Time Optimization Algorithm with Linear-Quadratic Convergence. *SIAM Journal on Optimization*, **27**, 205-245. https://doi.org/10.1137/15M1021106

[32] Mor-Yosef, L. and Avron, H. (2019) Sketching for Principal Component Regression. *SIAM Journal on Matrix Analysis and Applications*, **40**, 454-485. https://doi.org/10.1137/18M1188860

[33] Yang, Y., Pilanci, M. and Wainwright, M.J. (2017) Randomized Sketches for Kernels: Fast and Optimal Nonparametric Regression. *Annals of Statistics*, **45**, 991-1023. https://doi.org/10.1214/16-AOS1472

[34] Avron, H., Clarkson, K.L. and Woodruff, D.P. (2017) Faster Kernel Ridge Regression Using Sketching and Preconditioning. *SIAM Journal on Matrix Analysis and Applications*, **38**, 1116-1138. https://doi.org/10.1137/16M1105396

[35] Woodruff, D.P. (2014) Sketching as a Tool for Numerical Linear Algebra. *Foundations and Trends® in Theoretical Computer Science*, **10**, 1-157. https://doi.org/10.1561/0400000060

[36] Jarman, B., Yaniv, Y. and Needell, D. (2022) Online Signal Recovery via Heavy Ball Kaczmarz. *Proceedings of the* 2022 56*th Asilomar Conference on Signals*, *Systems*, *and Computers*, Pacific Grove, 31 October-2 November 2022, 276-280. https://doi.org/10.1109/IEEECONF56349.2022.10052070

[37] Zhang, Y. and Li, H. (2021) A Count Sketch Maximal Weighted Residual Kaczmarz Method for Solving Highly Overdetermined Linear Systems. *Applied Mathematics and Computation*, **410**, Article ID: 126486. https://doi.org/10.1016/j.amc.2021.126486

[38] Rudi, A., Calandriello, D., Carratino, L., *et al*. (2018) On Fast Leverage Score Sampling and Optimal Learning. 32*nd*

Conference on Neural Information Processing Systems (*NeurIPS* 2018), Montréal, 3-8 December 2018, 5672-5682.

[39] Cohen, M.B., Musco, C. and Musco, C. (2017) Input Sparsity Time Low-Rank Approximation via Ridge Leverage Score Sampling. *Proceedings of the* 28*th Annual ACM-SIAM Symposium on Discrete Algorithms*, Barcelona, 16-19 January 2017, 1758-1777. https://doi.org/10.1137/1.9781611974782.115

[40] Ailon, N. and Chazelle, B. (2009) The Fast Johnson-Lindenstrauss Transform and Approximate Nearest Neighbors. *SIAM Journal on Computing*, **39**, 302-322. https://doi.org/10.1137/060673096