On the Randomized Kaczmarz Algorithm

Liang Dai, Mojtaba Soltanalian, Kristiaan Pelckmans
Department of Information Technology, Uppsala University, Sweden.

***Abstract*—The Randomized Kaczmarz Algorithm is a randomized method which aims at solving a consistent system of over determined linear equations. This note discusses how to find an optimized randomization scheme for this algorithm, which is related to the question raised by [2]. Illustrative experiments are conducted to support the findings.**

***Index Terms*—Randomized Kaczmarz Algorithm, Convex Optimization, Linear System Solver**

I. Problem Statement

In this note, we discuss the Kaczmarz Algorithm (KA)[4], in particular the Randomized Kaczmarz Algorithm (RKA) [1], to find the unknown vector $\mathbf{x} \in \mathbb{R}^n$ of the following set of *consistent* linear equations:

$$A\mathbf{x} = \mathbf{b}, \tag{1}$$

where matrix $A \in \mathbb{R}^{m \times n}, m \geq n$, is of full column rank, and $\mathbf{b} \in \mathbb{R}^m$. Since [4], the KA has been applied to different fields and many new developments are reported. For instance, in [6], the author study the RKA when applied to the case of the linear systems are inconsistent. In [5], RKA is applied to the Computer Tomography. In [7], the authors present a method to accelerate the convergence of the RKA with the application of the Johnson-Lindenstrauss Lemma. In [8], the authors analyze the almost sure convergence of the RKA when proper stochastic properties of matrix A are introduced. In [9], the authors presented a practically more efficient approach to solve the linear systems by projecting to different blocks of rows of A, and a randomization technique is applied to find a good partition of the rows.

The KA can be described as follows. Let us define the hyperplane H_i as:

$$H_i = \{\mathbf{x} | \mathbf{a}_i^T \mathbf{x} = b_i\},$$

where the i-th row of A is denoted as $\mathbf{a}_i^T$ and the i-th element of $\mathbf{b}$ is denoted as b_i. Geometrically, the solution of (1) can be thought as the intersection of all hyperplanes $\{H_i\}_{i=1}^m$, and the KA seeks to find the solution by successively projecting to the hyperplanes from an initial approximation $\mathbf{x}_0$. The process is mathematically written as

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \frac{b_i - \mathbf{a}_i^T \mathbf{x}_k}{\|\mathbf{a}_i\|_2^2} \mathbf{a}_i, \tag{2}$$

where $i = mod(k, m) + 1$. Here we use the Matlab convention $mod(\cdot, \cdot)$ to denote the *modulus after the division* operation. Fig. 1 illustrates the algorithm in a low dimensional case.

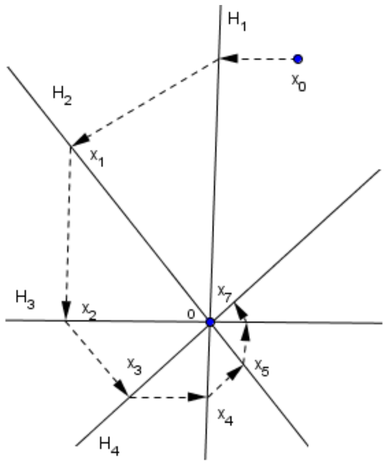

Fig. 1. A geometrical interpretation of the algorithm. Here, $m = 4$ and $n = 2$, and the solution $\mathbf{x}$ to $A\mathbf{x} = \mathbf{b}$ is represented by the point o. We can see that by this sequence of projections, $\mathbf{x}_k$ converges to the solution.

The key difference between the RKA and the KA is that RKA chooses the rows following a specified probability distribution. More precisely, the probability for selecting $\mathbf{a}_i^T$ is given as $\frac{\|\mathbf{a}_i\|_2^2}{\|A\|_F^2}$. Note that this probability is proportional to the row norms.

Although the KA is simple to state, its rate of convergence is still not completely explored. While for the RKA, with the predescribed choice of the probability distribution, the following convergence result is set up in [1]:

$$\mathbb{E}(\|\mathbf{x}_k - \mathbf{x}\|_2^2) \leq (1 - \kappa(A)^{-2})^k \|\mathbf{x}_0 - \mathbf{x}\|_2^2, \tag{3}$$

in which $\kappa(A) = \|A\|_F \|A^\dagger\|_2$, and with $\mathbb{E}$ concerning the random choices of rows in the RKA.

However, it is argued in [2] that *'Assigning probabilities corresponding to the row norms is in general certainly not optimal'*. In the follows, we will try to find an optimized probability distribution for selecting the rows from A, so that a better performance can be obtained. The distribution vector is derived by minimizing an upper bound to the convergence rate which can be obtained by solving a convex optimization problem.

This note is organized as follows. The next section discusses the main results; In section 3, we discuss how to approximately solve the arising Semi-Definite-Programming (SDP) problem with smaller computational cost; In section 4, illustrative experiments will be conducted to verify the findings; Finally, we draw some conclusions in section 5.

Research in this paper was supported by the VR project with number 106400801, and the European Research Council (ERC) grant with number 228044.

II. Optimized RKA

In the following, for convenience of discussion, we will introduce a new matrix $B \in \mathbb{R}^{m\times n}$. Let $\mathbf{b}_i^T$ denote the i-th row of B, which is defined as

$$\mathbf{b}_i = \frac{\mathbf{a}_i}{\|\mathbf{a}_i\|_2}, \forall i = 1, \cdots, m, \tag{4}$$

i.e. every row of the matrix B is a normalized version of the corresponding row of matrix A.

Let $\mathbf{p} \in \mathbb{R}^m$ be a probability distribution vector (i.e. $\mathbf{p} \geq 0$, $\mathbf{1}^T\mathbf{p} = 1$) for selecting the rows in the RKA method and let p_i denote the ith element of $\mathbf{p}$.

Assume that currently we have $\mathbf{x}_{k-1}$, and based on $\mathbf{x}_{k-1}$, the next approximation $\mathbf{x}_k$ is given by (2), in which the index i is chosen randomly according to $\mathbf{p}$. By the property of the projection operation, we have that

$$\|\mathbf{x}_k - \mathbf{x}\|_2^2 = \|\mathbf{x}_{k-1} - \mathbf{x}\|_2^2 \sin^2(\alpha_i), \tag{5}$$

in which α_i denotes the angle between $\mathbf{x}_{k-1} - \mathbf{x}$ and the selected $\mathbf{b}_i$, i.e. the normal direction of the chosen hyperplane.

Based on the previous formula, we have that

$$\mathbb{E}_{\cdot|\mathbf{x}_{k-1}}(\|\mathbf{x}_k - \mathbf{x}\|_2^2) = \|\mathbf{x}_{k-1} - \mathbf{x}\|_2^2 \sum_{i=1}^m p_i \sin^2(\alpha_i), \tag{6}$$

in which $\mathbb{E}_{\cdot|\mathbf{x}_{k-1}}$ denotes the expectation operator conditioned on $\mathbf{x}_{k-1}$. It follows that:

$$\sum_{i=1}^m p_i \sin^2(\alpha_i) \leq \sup_{\mathbf{y}\in\mathbb{R}^n, \mathbf{y}\neq\mathbf{0}} \sum_{i=1}^m p_i \sin^2(\beta_i) \triangleq \Omega_1, \tag{7}$$

and

$$\sum_{i=1}^m p_i \sin^2(\alpha_i) \geq \inf_{\mathbf{y}\in\mathbb{R}^n, \mathbf{y}\neq\mathbf{0}} \sum_{i=1}^m p_i \sin^2(\beta_i) \triangleq \Omega_2, \tag{8}$$

in which β_i denotes the angle between $\mathbf{y}$ and $\mathbf{b}_i$.

Based on the relations in (6), (7) and (8), we have that

$$\mathbb{E}_{\cdot|\mathbf{x}_{k-1}}(\|\mathbf{x}_k - \mathbf{x}\|_2^2) \leq \Omega_1 \|\mathbf{x}_{k-1} - \mathbf{x}\|_2^2, \tag{9}$$

and

$$\mathbb{E}_{\cdot|\mathbf{x}_{k-1}}(\|\mathbf{x}_k - \mathbf{x}\|_2^2) \geq \Omega_2 \|\mathbf{x}_{k-1} - \mathbf{x}\|_2^2. \tag{10}$$

By iterating the relations given in eq. (9) and eq. (10), the following results follow.

Theorem 1: We have that

$$\mathbb{E}(\|\mathbf{x}_k - \mathbf{x}\|_2^2) \leq \Omega_1^k \|\mathbf{x}_0 - \mathbf{x}\|_2^2, \tag{11}$$

and

$$\mathbb{E}(\|\mathbf{x}_k - \mathbf{x}\|_2^2) \geq \Omega_2^k \|\mathbf{x}_0 - \mathbf{x}\|_2^2, \tag{12}$$

in which the expectations are taken with respect to all the random choices of the rows up to time k.

Remark 1: Note that $\Omega_1 < 1$ can be guaranteed if $\mathbf{p}$ is a strictly positive vector. This can be proven by a contradiction argument as follows. If $\Omega_1 = 1$, and since $\sin^2(\beta_i) \leq 1$ for any i and $\sum_{i=1}^m p_i = 1$, we have that $\sin^2(\beta_i) = 1$, i.e. $\cos(\beta_i) = 0$ holds for all i. Considering that $rank(A) = n$, i.e. $rank(B) = n$, hence $\mathbf{x}_k - \mathbf{x}$ can not be orthogonal to the vectors $\{\mathbf{b}_i\}_{i=1}^m$, and the result follows. Based on this observation, we can see that exponential convergence in expectation can be obtained by a wide range of probability distribution vectors. This finding extends the result in [1], which only guarantees the exponential convergence for a given specific choice of the probability distribution vector. ■

According to Theorem 1, in order to get a better performance, we need to find a probability distribution vector, such that Ω_1 can be made as small as possible. When the optimized Ω_1 is obtained, we can also have a lower bound to the convergence speed of the RKA based on Ω_2. In the following, we will first derive a closed form for Ω_1 and Ω_2, and then introduce a convex optimization problem to calculate the probability distribution vector $\hat{\mathbf{p}}$ which minimizes Ω_1.

Notice that

$$\sum_{i=1}^m p_i \sin^2(\beta_i) = 1 - \sum_{i=1}^m p_i \cos^2(\beta_i),$$

so in order to minimize Ω_1, equivalently, we can maximize the following

$$\inf_{\mathbf{y}\in\mathbb{R}^n, \mathbf{y}\neq\mathbf{0}} \sum_{i=1}^m p_i \cos^2(\beta_i).$$

If we restrict $\|\mathbf{y}\|_2 = 1$, then we have that

$$\cos^2(\beta_i) = \mathbf{y}^T \mathbf{b}_i \mathbf{b}_i^T \mathbf{y}.$$

Therefore

$$\sum_{i=1}^m p_i \cos^2(\beta_i) = \sum_{i=1}^m p_i \mathbf{y}^T \mathbf{b}_i \mathbf{b}_i^T \mathbf{y},$$

where the right hand side equals

$$\mathbf{y}^T B^T \operatorname{diag}(\mathbf{p}) B \mathbf{y}.$$

Notice that

$$\min_{\mathbf{y}\in\mathbb{R}^n, \|\mathbf{y}\|_2=1} \mathbf{y}^T B^T \operatorname{diag}(\mathbf{p}) B \mathbf{y} = \sigma_n(B^T \operatorname{diag}(\mathbf{p}) B),$$

in which $\sigma_n(\cdot)$ denotes the smallest singular value of the matrix. The previous discussions can be summarized as:

Theorem 2:

$$\Omega_1 = 1 - \sigma_n(B^T \operatorname{diag}(\mathbf{p}) B). \tag{13}$$

Similarly, we have that:

Corollary 1:

$$\Omega_2 = 1 - \sigma_1(B^T \operatorname{diag}(\mathbf{p}) B), \tag{14}$$

in which $\sigma_1(\cdot)$ denotes the maximal singular value of the matrix.

Notice that minimizing Ω_1 is equivalent to maximizing $\sigma_n(B^T \operatorname{diag}(\mathbf{p}) B)$, then we can solve the following problem instead:

$$\begin{aligned} \max_{\mathbf{p}\in\mathbb{R}^m} \ & \sigma_n(B^T \operatorname{diag}(\mathbf{p}) B) \\ s.t. \quad & \mathbf{1}^T\mathbf{p} = 1; \\ & p_i \geq 0, \ i = 1, \ldots, m. \end{aligned} \tag{15}$$

This problem can be rewritten as the following SDP problem, in which $\hat{t}$ denotes the optimized σ_n and $\hat{\mathbf{p}}$ denotes the corresponding probability distribution vector:

$$\begin{aligned}(\hat{\mathbf{p}}, \hat{t}) = & \underset{\mathbf{p}\in\mathbb{R}^m, t\in\mathbf{R}}{\arg\max} \quad t \qquad (16)\\ s.t. \quad & \mathbf{1}^T\mathbf{p} = 1;\\ & p_i \geq 0,\ i = 1, \ldots, m;\\ & B^T \operatorname{diag}(\mathbf{p})B - tI_n \succeq 0.\end{aligned}$$

After solving the optimization problem of (16), $\hat{\mathbf{p}}$ is applied to the RKA to select the rows. Such a scheme will be abbreviated as ORKA in the following.

Remark 2: There exist cases such that $\Omega_1 = \Omega_2$, i.e. there exists a vector $\mathbf{p}$, such that

$$\sigma_1(B^T \operatorname{diag}(\mathbf{p})B) = \sigma_n(B^T \operatorname{diag}(\mathbf{p})B),$$

i.e. $B^T \operatorname{diag}(\mathbf{p})B = \frac{1}{n}I_n$. In such cases, $\Omega_1 = \Omega_2 = 1 - \frac{1}{n}$, and the optimized probability distribution obtained by solving eq. (16) is the same as suggested in [1]. It can be verified that when the columns of A are orthogonal and of equal norm, then such property will hold. ■

Remark 3: The optimization problem (16) can also be formulated as

$$\begin{aligned}\hat{\mathbf{q}} = & \underset{\mathbf{q}\in\mathbb{R}^m}{\arg\min} \ \mathbf{1}^T\mathbf{q} \qquad (17)\\ s.t. \ & B^T \operatorname{diag}(\mathbf{q})B - I_n \succeq 0;\\ & q_i \geq 0,\ i = 1, \ldots, m.\end{aligned}$$

in the sense that $\hat{t} = \frac{1}{\mathbf{1}^T\hat{\mathbf{q}}}$ and $\hat{\mathbf{p}} = \hat{t}\hat{\mathbf{q}}$.

Since $\mathbf{q}$ in (17) is nonnegative, one has that $\mathbf{1}^T\mathbf{q} = \|\mathbf{q}\|_1$. It is known that the l_1 norm minimization problem is likely to return sparse solutions[11], which gives that $\hat{\mathbf{q}}$ is likely to be sparse. In the experiment section, we will also illustrate this phenomena. ■

Next, we discuss the relation between the ORKA and the RKA. It is obvious that the projection operations in (2) depend only on the corresponding normal vectors of the hyperplanes $\{H_i\}_{i=1}^m$, so we can optimize $\kappa(A) = \|A\|_F\|A^\dagger\|_2$ subject to the norms of the rows of matrix A. The optimization problem is given as

$$\min_{\{\|\mathbf{a}_i\|_2\}_{i=1}^m} \quad \kappa(A) = \|A\|_F\|A^\dagger\|_2.$$

Define $\mathbf{q} \in \mathbb{R}^m$, in which $q_i = \|\mathbf{a}_i\|_2^2$ for $i = 1 \cdots m$. Then the previous optimization problem can be written as

$$\min_{\mathbf{q}} \quad \frac{\sqrt{\mathbf{1}^T\mathbf{q}}}{\sigma_n(A)}.$$

Set $\mathbf{1}^T\mathbf{q} = 1$ and notice the fact that $A^TA = B^T \operatorname{diag}(\mathbf{q})B$, then we can rewrite the previous problem as follows

$$\begin{aligned}(\hat{\mathbf{q}}, \hat{\sigma}_n) = & \underset{\mathbf{q}\in\mathbb{R}^m, \sigma_n(A)\in\mathbf{R}}{\arg\max} \quad \sigma_n^2(A) \qquad (18)\\ s.t. \quad & \mathbf{1}^T\mathbf{q} = 1;\\ & q_i \geq 0,\ i = 1, \ldots, m;\\ & B^T \operatorname{diag}(\mathbf{q})B - \sigma_n^2(A)I_n \succeq 0.\end{aligned}$$

It can be observed that this optimization is equivalent to the problem given by (16).

We conclude this observation in the following theorem.

Theorem 3: The ORKA can do at least as good as the RKA, in the sense that if we optimize $\kappa(A)$ over the norms of rows of A, we obtain the same probability distribution vector as the one obtained by the ORKA.

III. Further Discussions

Note that although the formulation in (16) is convex, it is still time consuming to solve this SDP optimization problem. In this section, we will discuss two possibilities to solve it approximately , which can alleviate some of the computational cost. One approximation of (16) is obtained by relaxing the constraint $B^T \operatorname{diag}(\mathbf{p})B - tI_n \succeq 0$ by the following linear constraints:

$$\mathbf{b}_i^T \operatorname{diag}(\mathbf{p})\mathbf{b}_i \geq t; \forall i = 1, \ldots, m. \qquad (19)$$

It is due to the fact that, for two positive semidefinite matrices $P_1, P_2 \in \mathbb{R}^{n\times n}$, if $P_1 \succeq P_2$, then $P_1(i,i) \geq P_2(i,i)$ holds for i = $1, \cdots, n$. Such relaxation reduces the SDP problem into a Linear Programming (LP) problem, which is computationally easier to solve.

In order to get a better relaxation, we introduce another approximation method which relates to the research of *Optimal Input Design* [10]. Notice that $tr(B^T \operatorname{diag}(\mathbf{p})B) = 1$, i.e. the summation of all the singular values of $B^T \operatorname{diag}(\mathbf{p})B$ is fixed, then maximizing $\sigma_n(B^T \operatorname{diag}(\mathbf{p})B)$ means that we want all the singular values of $B^T \operatorname{diag}(\mathbf{p})B$ to be close. This leads us to consider maximizing the product of the singular values of $B^T \operatorname{diag}(\mathbf{p})B$, or maximizing the determinant of $B^T \operatorname{diag}(\mathbf{p})B$. As the log function is monotonically increasing, we can optimize the following

$$\max_{\mathbf{p}\in\mathbb{R}^m} \log|B^T \operatorname{diag}(\mathbf{p})B|, \qquad (20)$$

in which $|\cdot|$ denotes the matrix determinant. Optimizing this quantity subject to the same constraints of (15) boils down to solve the so-called *D-Optimal Design* problem. One simple iterative algorithm to solve such problem has been suggested in [12], which is given as

$$\begin{aligned}p_i^0 &= \frac{\|\mathbf{a}_i\|^2}{\|A\|_F^2};\ i = 1, \ldots, m;\\ p_i^{t+1} &= p_i^t \frac{\mathbf{b}_i^T(B^T \operatorname{diag}(\mathbf{p}^t)B)^{-1}\mathbf{b}_i}{n};\ i = 1, \ldots, m. \qquad (21)\end{aligned}$$

Here, $\mathbf{p}^t$ denotes the estimation at time t, and p_i^t denotes its i-th element. It has been proven in [13] that for this algorithm, $\log|B^T \operatorname{diag}(\mathbf{p}^t)B|$ decreases monotonically w.r.t. t. We will make use of such property to approximately solve (15) when the objective function is replace by (20). More discussions will be given in next section.

IV. Experiments

In this section, we will conduct experiments to illustrate the efficacy of the presented methods. The setup of our experiment is given as follows. The matrix A is first generated by *randn(m,n)* in Matlab with m = 200 and n = 20, after that, each row is normalized, and then scaled with a random number which is uniformly distributed in $[0, 1]$. The reason for generating A as such is that in the first stage, the generated rows of A will have different directions which are uniformly distributed on the sphere S^{n-1}[14]; and in the second stage, different rows of A with be assigned with different norms, which is directly related to the probability distribution vector chosen in [1]. $\mathbf{x}$ is generated by *randn(n,1)*, and $\mathbf{b}$ is generated as $\mathbf{b} = A\mathbf{x}$. We will compare the Mean Square Error (MSE) along the projection path obtained by all these methods, the first is the one suggested in [1] (abbreviated as *RKA*), the second is the one obtained by the SDP optimization given by (16) (abbreviated as *ORKA*) and the third is the one obtained by the LP approximations given by (19) (abbreviated as *LPORKA*), the last is the one obtained by the iterative method to solve the D-Optimal Design criteria (abbreviated as *ITEORKA*). We iterate (21) for 10 times in this experiment. For each method, we run the experiment 2000 times to get the averaged performance. The CVX toolbox[1] is used to solve the SDP and LP optimization problems. From the experiment, we can observe that the time for solving the LP problem in LPORKA is close to the time needed for the 10 iterations of (21), and the time needed for solving (16) in ORKA is approximately 7 times as them.

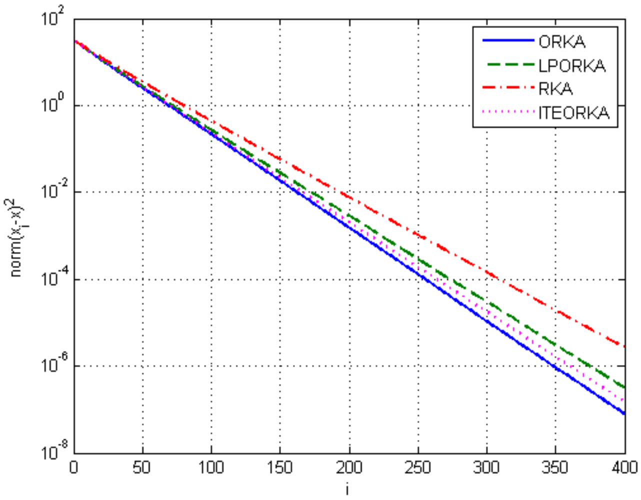

Fig. 2. The curves demonstrate the MSE for different methods. We can see that the ORKA improves the convergence speed the most; the LPORKA method and the ITERKA method also improve the convergence speed, and the ITEORKA method improves more than the LPORKA method.

V. Conclusion

This note discusses the possibility and methodology to find a probability distribution vector for selecting the rows of A to result in a better convergence speed of the Randomize Kaczmarz Algorithm. The lower bound and upper bound for the convergence speed is derived first. Then an optimized probability distribution vector is obtained by minimizing the upper bound, which turns to be given by solving a convex optimization problem. Properties of the approach are also discussed along the note.

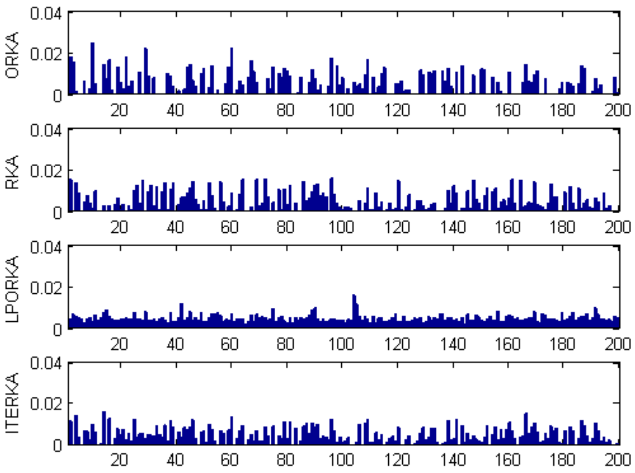

Fig. 3. An illustration of the probability distribution vectors obtained by different methods. Note that there are 68 zero elements of the probability distribution vector obtained by the ORKA method, which is 34% sparsity of the total length.

References

[1] T. Strohmer, R. Vershynin, A randomized Kaczmarz algorithm with exponential convergence, Journal of Fourier Analysis and Applications, 15(2), 262-278, 2009.

[2] Y. Censor, G.T. Herman, and M. Jiang, A note on the behavior of the randomized Kaczmarz algorithm of Strohmer and Vershynin, Journal of Fourier Analysis and Applications, 15(4), 431-436, 2009.

[3] T. Strohmer, R. Vershynin, Comments on the randomized Kaczmarz method, Journal of Fourier Analysis and Applications, 15(4), 437-440, 2009.

[4] S. Kaczmarz, Angenaherte Auflosung von Systemen linearer Gleichungen, Bulletin International de l'Acad$\acute{e}$mie Polonaise des Sciences et des Lettres, 35, 355-357, 1937.

[5] F. Natterer, The Mathematics of Computerized Tomography, Wiley, New York, 1986.

[6] D. Needell, Randomized Kaczmarz solver for noisy linear systems, BIT Numerical Mathematics, 50(2), 395-403, 2010.

[7] Y. Eldar, D. Needell, Acceleration of randomized Kaczmarz method via the Johnson-Lindenstrauss lemma, Numerical Algorithms, 58(2), 163-177, 2011.

[8] X. Chen, A. Powell, Almost sure convergence for the Kaczmarz algorithm with random measurements, Journal of Fourier Analysis and Applications, 18(6), 1195-1214, 2012.

[9] D. Needell and J. A. Tropp, Paved with Good Intentions: Analysis of a Randomized Block Kaczmarz Method, Linear Algebra and its Applications, 441, 199-221, 2014.

[10] V. V. Fedorov, Theory of Optimal Experiments, Academic Press, 1971.

[11] D. L. Donoho, Compressed Sensing, IEEE Transactions on Information Theory, 52(4), 1289-1306, 2006

[12] S. D. Silvey, D. M. Titterington and B. Torsney, An algorithm for optimal designs on a finite design space, Commun. Stat. Theory Methods, 14, 1379-1389, 1978.

[13] Y. Yu, Monotonic convergence of a general algorithm for computing optimal designs, The Annals of Statistics, 38(3), 1593-1606, 2010.

[14] G. Marsaglia, Choosing a Point from the Surface of a Sphere, The Annals of Mathematical Statistics, 43(2), 645-646, 1972.

[1] http://cvxr.com/