A NEW PERSPECTIVE ON RANDOMIZED GOSSIP ALGORITHMS

Nicolas Loizou   *Peter Richtárik*

School of Mathematics, The University of Edinburgh, United Kingdom

July 1, 2016

ABSTRACT

In this short note we propose a new approach for the design and analysis of randomized gossip algorithms which can be used to solve the average consensus problem. We show how that Randomized Block Kaczmarz (RKB) method—a method for solving linear systems—works as gossip algorithm when applied to a special system encoding the underlying network. The famous pairwise gossip algorithm arises as a special case. Subsequently, we reveal a hidden duality of randomized gossip algorithms, with the dual iterative process maintaining a set of numbers attached to the edges as opposed to nodes of the network. We prove that RBK obtains a superlinear speedup in the size of the block, and demonstrate this effect through experiments.

Index Terms— Average Consensus Problem, Linear Systems, Networks, Randomized Gossip Algorithms, Randomized Block Kaczmarz

1. INTRODUCTION

The average consensus problem and randomized gossip algorithms for solving it appear in many applications, including distributed data fusion in sensor networks [1], load balancing [2] and clock synchronization [3]. This subject was studied extensively in the last decade; for instance, the seminal 2006 paper of Boyd et al. [4] on randomized gossip algorithms motivated a fury of subsequent research and generated more than 1500 citations to date. For a survey of selected relevant work prior to 2010, we refer the reader to the work of Dimakis et al. [5]. For more recent results on randomized gossip algorithms we suggest [6, 7, 8]. See also [9, 10, 11].

1.1. The average consensus problem

In the average consensus (AC) problem, we are given an undirected connected network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with node set $\mathcal{V} = \{1, 2, \dots, n\}$ and edges $\mathcal{E}$. Each node $i \in \mathcal{V}$ "knows" a private value $c_i \in \mathcal{R}$. The goal of AC is for every node of the network to compute the average of these private values, $\bar{c} := \frac{1}{n}\sum_i c_i$, in a distributed fashion. That is, the exchange of information can only occur between connected nodes (neighbors).

1.2. Contributions.

In this paper we revisit, from a fresh perspective, the AC problem. Our starting point is the recent observation of Gower and Richtárik [12] that the most basic randomized gossip algorithm ("randomly pick an edge $(i, j) \in \mathcal{E}$ and then replace the values stored at vertices i and j by their average") is an instance of the randomized Kaczmarz (RK) method for solving consistent linear systems, applied to a specific linear system encoding the AC problem. The RK method was first analyzed in 2009 by Strohmer and Vershynin [13], and since then, there was an explosion of activity in refining, generalizing and extending the results [14, 15, 16, 17, 18]. In this paper, we examine the Stochastic Dual Ascent (SDA) method of Gower and Richtárik [12], which includes the RK method as a special case, in the context of AC problem. We show how the complexity result of SDA implies a bound on the ε-averaging time which is well-known in the literature for a more restricted class of randomized gossip algorithms. Further, we explain how SDA uncovers a fundamental but hitherto hidden duality of randomized gossip algorithms, and give a natural interpretation thereof. We then focus on a specific subclass of SDA which is identical to the randomized block Kaczmarz method [14] in the primal space, and which can be interpreted as a randomized Newton method in the dual space. In particular, we show that the method has a certain superlinear speedup property, and explain what this property means. Finally, we perform experiments to justify the last claim.

2. SDA: STOCHASTIC DUAL ASCENT

In this section we briefly review those aspects of the work of Gower and Richtárik [12] on Stochastic Dual Ascent (SDA) which we will need in the rest of the paper.

Consider an $m \times n$ real matrix $\mathbf{A}$ (assume it does not contain any zero rows) and vector $b \in \mathcal{R}^m$ such that the linear system $\mathbf{A}x = b$ is consistent (i.e., has a solution). Since we do not assume the solution is unique, we shall be interested in a particular solution:

$$\min_{x\in\mathcal{R}^n} \tfrac{1}{2}\|x-c\|^2 \quad \text{subject to} \quad \mathbf{A}x=b. \qquad (1)$$

Above, $c\in\mathcal{R}^n$ is a given vector and $\|\cdot\|$ is the standard Euclidean norm. In words, in (1) we are seeking the solution of the system which is closest to c. By x^* we denote the solution of (1). The *dual* of problem (1) is

$$\max_{y\in\mathcal{R}^m} D(y) := (b-\mathbf{A}c)^\top y - \tfrac{1}{2}\|\mathbf{A}^\top y\|^2. \qquad (2)$$

SDA is a randomized iterative algorithm for solving (2), performing the iteration $y^{k+1}=y^k+\mathbf{S}_k\lambda^k$, where $\mathbf{S}_k$ is a matrix chosen in an i.i.d. fashion throughout the iterative process from an arbitrary but fixed distribution (which is a parameter of the method) and λ^k is a vector chosen *afterwards* so that $D(y^k+\mathbf{S}_k\lambda)$ is maximized in λ. In general, the maximizer in λ is not unique. In SDA, we let λ^k to be the least-norm maximizer, which leads to the iteration

$$\boxed{y^{k+1}=y^k-\mathbf{R}_k(\mathbf{A}(c+\mathbf{A}^\top y^k)-b)} \qquad (3)$$

where $\mathbf{R}_k := \mathbf{S}_k((\mathbf{S}_k)^\top\mathbf{A}\mathbf{A}^\top\mathbf{S}_k)^\dagger(\mathbf{S}_k)^\top$ (this matrix is always symmetric and positive semidefinite). With the sequence of the *dual iterates* $\{y^k\}$ we associate a sequence of *primal iterates* $\{x^k\}$ as follows:

$$x^k := c+\mathbf{A}^\top y^k. \qquad (4)$$

By combining (4) with (3), we obtain the following algorithm:

$$\boxed{x^{k+1}=x^k-\mathbf{A}^\top\mathbf{R}_k(\mathbf{A}x^k-b)} \qquad (5)$$

If $\mathbf{S}_k$ is chosen randomly from the set of unit coordinate/basis vectors in $\mathcal{R}^m$, then the dual method (3) is randomized coordinate descent [19, 20], and the corresponding primal method (5) is RK. More generally, if $\mathbf{S}_k$ is a random column submatrix of the $m\times m$ identity matrix, the dual method is the randomized Newton method [21], and the corresponding primal method is a block version of RK [14]. We shall describe the more general case in more detail in Section 4.

The basic convergence guarantees for both the primal and the dual iterative processes are presented in the following theorem.

Theorem 2.1 (Complexity of SDA [12])**.** *Let $y^0=0$ and assume that the matrix $\mathbf{H}:=\mathbb{E}[\mathbf{R}_k]$ is well defined and nonsingular. Then the dual iterates $\{y^k\}$ of SDA defined in* (3) *for all $k\geq 0$ satisfy*

$$\mathbb{E}[D(y^*)-D(y^k)]\leq\rho^k(D(y^*)-D(y^0)). \qquad (6)$$

Likewise, the corresponding primal iterates, defined in (4) *and explicitly written in* (5)*, for all $k\geq 0$ satisfy*

$$\mathbb{E}[\|x^k-x^*\|^2]\leq\rho^k\|x^0-x^*\|^2. \qquad (7)$$

The convergence rate ρ is given by

$$\rho:=1-\lambda^+_{\min}(\mathbf{A}^\top\mathbf{H}\mathbf{A})\in(0,1), \qquad (8)$$

where $\lambda^+_{\min}(\cdot)$ denotes the minimum nonzero eigenvalue.

3. RANDOMIZED GOSSIP & SDA

We propose that randomized gossip algorithms be viewed as applications of SDA (either in the primal or dual form) to a particular problem of the form (1) (resp. (2)). In particular, we let $c=(c_1,\dots,c_n)$ be the initial values stored at the nodes of $\mathcal{G}$, and choose $\mathbf{A}$ and b as in the next definition.

Definition 3.1. *We say that $\mathbf{A}x=b$ is an "average consensus (AC) system" when $\mathbf{A}x=b$ iff $x_i=x_j$ for all $i,j\in\mathcal{V}$.*

It is easy to see that $\mathbf{A}x=b$ is an AC system precisely when $b=0$ and the nullspace of $\mathbf{A}$ is $\{t1_n : t\in\mathcal{R}\}$, where 1_n is the vector of all ones in $\mathcal{R}^n$. Hence, $\mathbf{A}$ has rank $n-1$. Moreover, it is easy to see that for any AC system, the solution of (1) necessarily is $x^*=\bar{c}\cdot 1_n$ — this is why we singled out AC systems. In this sense, *any* algorithm for solving (1) will "find" the average $\bar{c}$. However, in order to obtain a distributed algorithm we need to make sure that only "local" (with respect to $\mathcal{G}$) exchange of information is allowed.

3.1. Standard Form and Mass Preservation

Assume that $\mathbf{A}x=b$ is an AC system. Then the primal iterative process (5) can be written in the form

$$x^{k+1}=\mathbf{W}_k x^k, \qquad (9)$$

where $\mathbf{W}_k := \mathbf{I}-\mathbf{A}^\top\mathbf{R}_k\mathbf{A}$. Eq (9) is the standard form in which randomized gossip algorithms are written. What is new here is that the iteration matrix $\mathbf{W}_k$ has a specific structure which guarantees convergence to x^* under very weak assumption (see Theorem 2.1). Note that if $y^0=0$, then $x^0=c$, i.e., the starting primal iterate is the vector of private values (as should be expected from any gossip algorithm).

The primal iterates (5) of SDA enjoy a mass preservation property (the proof follows from (4) in view of $\mathbf{A}1_n=0$):

Theorem 3.2 (Mass preservation)**.** *If $\mathbf{A}x=b$ is an AC system, then the primal iterates* (5) *for all $k\geq 0$ satisfy: $\frac{1}{n}\sum_{i=1}^n x_i^k=\bar{c}$.*

3.2. ε-Averaging Time

Let $z^k := \|x^k-x^*\|$. The typical measure of convergence speed employed in the randomized gossip literature, called ε-averaging time and here denoted by $K(\varepsilon)$, represents the smallest time k for which x^k gets within εz^0 from x^*, with probability greater than $1-\varepsilon$, uniformly over all starting values $x^0=c$. More formally, we define

$$K(\varepsilon) := \sup_{c\in\mathcal{R}^n}\inf\{k : \mathbb{P}\left(z^k>\varepsilon z^0\right)\leq\varepsilon\}.$$

This definition differs slightly from the standard one in that we use z^0 instead of $\|c\|$.

Inequality (7), together with Markov inequality, can be used to give a bound on $K(\varepsilon)$, formalized in the following theorem.

Theorem 3.3. *Assume $\mathbf{A}x = b$ is an AC system. Let $y^0 = 0$ and assume $\mathbf{H} = \mathbb{E}[\mathbf{R}_k]$ is nonsingular. Then for any $0 < \varepsilon < 1$ we have $K(\epsilon) \leq 3\log(1/\varepsilon)/\log(1/\rho) \leq \frac{3}{1-\rho}\log(1/\epsilon)$, where ρ is defined in* (8).

It can be shown that under the assumptions of the above theorem, $\mathbf{A}^\top\mathbf{H}\mathbf{A}$ only has a single zero eigenvalue, and hence $\lambda_{\min}^{+}(\mathbf{A}^\top\mathbf{H}\mathbf{A})$ is the second smallest eigenvalue of $\mathbf{A}^\top\mathbf{H}\mathbf{A}$. Therefore, ρ is the second largest eigenvalue of $\mathbf{I} - \mathbf{A}^\top\mathbf{H}\mathbf{A} = \mathbb{E}[\mathbf{W}_k]$. The bound on $K(\varepsilon)$ appearing in Theorem 3.3 is often written with ρ replaced by $\lambda_2(\mathbb{E}[\mathbf{W}_k])$ [4].

4. BLOCK GOSSIP ALGORITHMS

In the previous section we highlighted some properties of SDA relevant to the randomized gossip literature, but without interpreting SDA as a gossip, or for that matter, distributed algorithm. In this section we remedy this by focusing on a particular AC system and a particular random matrix $\mathbf{S}_k$. By being specific, we will be able to give a natural interpretation of SDA as a gossip algorithm.

In particular, we choose $\mathbf{A}$ to be the $|\mathcal{E}| \times n$ matrix such that $\mathbf{A}x = 0$ directly encodes the constraints $x_i = x_j$ for $(i, j) \in \mathcal{E}$. That is, row $e = (i, j)$ of matrix $\mathbf{A}$ contains value 1 in column i, value -1 in column j (we use an arbitrary but fixed order of nodes defining each edge in order to fix $\mathbf{A}$) and zeros elsewhere.

Next, $\mathbf{S}_k$ is selected in each iteration to be a random column submatrix of the $m \times m$ identity matrix corresponding to columns indexed by a random subset of edges $\mathcal{S}_k \subseteq \mathcal{E}$. We shall write $\mathbf{S}_k = \mathbf{I}_{\mathcal{S}_k}$. If $\mathcal{S}_k = \{1, 2\}$, for instance, then $\mathbf{S}_k$ consists of the first and second column of $\mathbf{I}$. For simplicity, from now on we will drop the subscript and write $\mathcal{S}$ instead of $\mathcal{S}_k$. This choice means that primal SDA is the *randomized block Kaczmarz (RBK) method.*

4.1. Randomized Block Kaczmarz as a Gossip Algorithm

In our setup, the primal iterative process (5) has the form:

$$x^{k+1} = x^k - \mathbf{A}^\top\mathbf{I}_{\mathcal{S}}(\mathbf{I}_{\mathcal{S}}^\top\mathbf{A}\mathbf{A}^\top\mathbf{I}_{\mathcal{S}})^\dagger\mathbf{I}_{\mathcal{S}}^\top\mathbf{A}x^k. \qquad (10)$$

Algorithm (10) can be shown to be equivalent to the following "sketch and project" iteration (see [22] for additional equivalent viewpoints):

$$x^{k+1} = \underset{x\in\mathcal{R}^n}{\operatorname{argmin}}\{\|x - x^k\|^2 \ : \ \mathbf{I}_{\mathcal{S}}^\top\mathbf{A}x = 0\} \qquad (11)$$

which is a (more general) variant of the RBK method of Needell [14]. More specifically, this method works by projecting the last iterate x^k onto the solution set of a row subsystem of $\mathbf{A}x = 0$, where the selected rows correspond to a random subset $\mathcal{S} \subseteq \mathcal{E}$ of selected edges.

While (11) (resp. (11)) may seem to be a complicated algebraic (resp. variational) characterization of the method, due to our choice of $\mathbf{A}$ we have the following result which gives a natural interpretation of RKB as a gossip algorithm (see also Figure 1).

Theorem 4.1 (RBK as a Gossip Algorithm)**.** *Consider the AC problem. Then each iteration of RBK (Algorithm 10) works as follows: 1) Select a random set of edges $\mathcal{S} \subseteq \mathcal{E}$, 2) Form subgraph $\mathcal{G}_k$ of $\mathcal{G}$ from the selected edges 3) For each connected component of $\mathcal{G}_k$, replace node values with their average.*

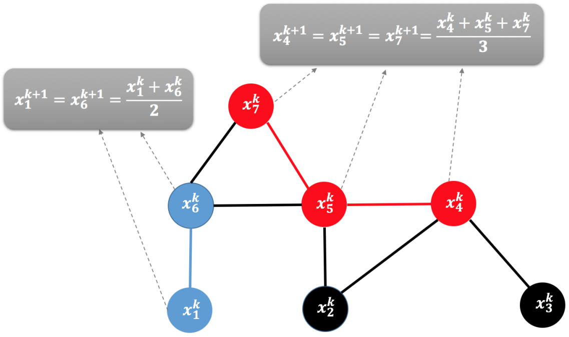

$x_4^{k+1} = x_5^{k+1} = x_7^{k+1} = \frac{x_4^k + x_5^k + x_7^k}{3}$
$x_1^{k+1} = x_6^{k+1} = \frac{x_1^k + x_6^k}{2}$
x_7^k
x_6^k
x_5^k
x_4^k
x_1^k
x_2^k
x_3^k

Fig. 1: Example of how the RBK method works as gossip algorithm. In the presented network 3 edges are randomly selected and a subgraph of two connected components (blue and red) is formed. Then the nodes of each connected component update their private values to their average.

Notice that in the special case in which $\mathcal{S}$ is always a singleton, Algorithm (10) reduces to the randomized Kaczmarz method. This means that only a random edge is selected in each iteration and the nodes incident with this edge replace their local values with their average. This is the pairwise gossip algorithm of Boyd [4]. Theorem 4.1 extends this interpretation to the case of the RBK method.

4.2. Randomized Newton as a Dual Gossip Algorithm

In this subsection we bring a new insight into the randomized gossip framework by presenting how the dual iterative process that is associated to RBK method solves AC problem. The dual iterative process (3) takes on the form:

$$y^{k+1} = y^k - \mathbf{I}_{\mathcal{S}}(\mathbf{I}_{\mathcal{S}}^\top\mathbf{A}\mathbf{A}^\top\mathbf{I}_{\mathcal{S}})^\dagger\mathbf{A}(c + \mathbf{A}^\top y^k). \qquad (12)$$

This is a randomized variant of the Newton method applied to the problem of maximizing the quadratic function $D(y)$. Indeed, as we have seen before, in each iteration we perform the update $y^{k+1} = y^k + \mathbf{I}_{\mathcal{S}}\lambda^k$, where λ^k is chosen greedily so that $D(y^{k+1})$ is maximized. In doing so, we invert a random principal submatrix of the Hessian of D, whence the name.

Randomized Newton Method (RNM) was first proposed by Qu et al. [21]. RNM was first analyzed as an algorithm for minimizing *smooth strongly convex functions*. In [12] it was also extended to the case of a *smooth but weakly convex quadratics*. This method was not previously associated with any gossip algorithm.

The most important distinction of RNM compared to existing gossip algorithms is that it operates with values that are associated to the *edges* of the network. To the bets of our knowledge, it the first *dual* gossip method. In particular, instead of iterating over values stored at the nodes, RNM uses these values to update "dual weights" $y^k \in \mathcal{R}^m$ that correspond to the edges $\mathcal{E}$ of the network.

Natural Interpretation. In iteration k, *RNM* (Algorithm (12)) executes the following steps: 1) Select a random set of edges $\mathcal{S}_k \subseteq \mathcal{E}$, 2) Form a subgraph $\mathcal{G}_k$ of $\mathcal{G}$ from the selected edges, 3) The values of the edges in each connected component of $\mathcal{G}_k$ are updated: their new values are a linear combination of the private values of the nodes belonging to the connected component and of the adjacent edges of their connected components.

Dual Variables as Advice. The weights y^k of the edges have a natural interpretation as *advice* that each selected node receives from the network in order to update its value (to one that will eventually converge to the desired average).

Consider RNM performing the k^{th} iteration and let $\mathcal{V}_r$ denote the set of nodes of the selected connected component that node i belongs to. Then, from Theorem 4.1 we know that $x_i^{k+1} = \sum_{i \in \mathcal{V}_r} x_i^k / |\mathcal{V}_r|$. Hence, by using (4), we obtain the following identity:

$$(\mathbf{A}^\top y^{k+1})_i = \tfrac{1}{|\mathcal{V}_r|} \textstyle\sum_{i \in \mathcal{V}_r} (c_i + (\mathbf{A}^\top y^k)_i) - c_i \qquad (13)$$

Thus in each step $(\mathbf{A}^\top y^{k+1})_i$ represents the term (advice) that must be added to the initial value c_i of node i in order to update its value to the average of the values of the nodes of the connected component i belongs to.

4.3. Importance of the dual perspective

It was shown in [21] that when RNM (and as a result, RBK) is viewed as a family of methods indexed by the size $\tau := \mathbb{E}[|\mathcal{S}|]$, then $\tau \to 1/(1-\rho)$, where ρ is defined in (8), decreases *superlinearly* fast in τ. In [21], this was only shown for full rank $\mathbf{A}$. In the next result we extend it to AC matrices $\mathbf{A}$ (which are necessarily rank-deficient).

Theorem 4.2. *RBK enjoys superlinear speedup in τ. That is, as τ increases by some factor, the iteration complexity drops by a factor that is at least as large.*

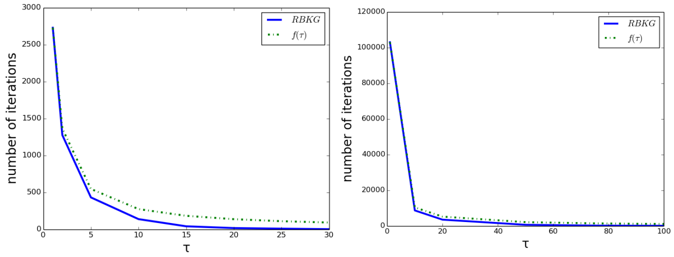
5. NUMERICAL EXPERIMENTS

We devote this section to experimentally evaluate the performance of the proposed gossip algorithms: RBK (the primal method) and RNM (the dual method). Recall that these methods solve the same problem, and their iterates are related via a simple affine transform. Hence, all results shown apply to both RBK and RNM.

Through these experiments we demonstrate the theoretical results presented in the previous section. That is, we show that for a connected network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, choosing more edges (i.e., larger $\tau := \mathbb{E}[|\mathcal{S}|]$) leads to a superlinear speedup in terms of iteration complexity.

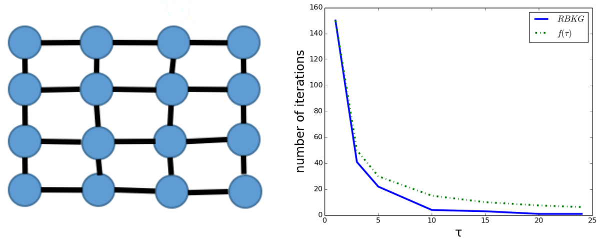
In comparing the number of iterations for different values of τ, we use the relative error $\varepsilon = \|x^k - x^*\|/\|c - x^*\|$. We let $c_i = i$ for each node $i \in \mathcal{V}$. We run RBK until the relative error becomes smaller than 0.01. The blue solid line in the figures denotes the actual number of iterations (after running the code) needed in order to achieve $\varepsilon \leq 10^{-2}$ for different values of τ. The green dotted line represents the function $f(\tau) := \frac{\ell}{\tau}$, where ℓ is the number of iterations of RBK with $\tau = 1$ (i.e., the pairwise gossip algorithm). In other words, the green line shows linear speedup, while the fact that the blue line obtained through our experiments is below the green line shows exhibits superlinear speedup.

The networks that we use in our experiments are the ring graph (cycle) with 30 and 100 nodes (Fig 2) and the 4×4 grid graph (Fig 3). When we choose $|\mathcal{S}| = m$ (i.e., we choose to update dual variables corresponding to all edges in each iteration), then $\rho = 0$, which means that the method converges in one step.



(a) ring graph with $n = 30$ (b) ring graph with $n = 100$

Fig. 2: Superlinear speedup of RBK on the ring graph.



(a) Illustration of 4×4 grid graph

(b)

Fig. 3: Superlinear speedup of RBK on the 4×4 grid graph

6. REFERENCES

[1] Lin Xiao, Stephen Boyd, and Sanjay Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*. IEEE, 2005, pp. 63–70.

[2] George Cybenko, "Dynamic load balancing for distributed memory multiprocessors," *Journal of Parallel and Distributed Computing*, vol. 7, no. 2, pp. 279–301, 1989.

[3] Nikolaos M Freris and Anastasios Zouzias, "Fast distributed smoothing of relative measurements," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*. IEEE, 2012, pp. 1411–1416.

[4] Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah, "Randomized gossip algorithms," *IEEE Transactions on Information Theory*, vol. 14, no. SI, pp. 2508–2530, 2006.

[5] Alexandros G Dimakis, Soummya Kar, José M.F Moura, Michael G Rabbat, and Anna Scaglione, "Gossip algorithms for distributed signal processing," *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1847–1864, 2010.

[6] Jun Y Yu and Michael Rabbat, "Performance comparison of randomized gossip, broadcast gossip and collection tree protocol for distributed averaging," in *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2013 IEEE 5th International Workshop on*. IEEE, 2013, pp. 93–96.

[7] Anastasios Zouzias and Nikolaos M Freris, "Randomized gossip algorithms for solving Laplacian systems," in *Control Conference (ECC), 2015 European*. IEEE, 2015, pp. 1920–1925.

[8] Ji Liu, Brian D.O Anderson, Ming Cao, and A.Stephen Morse, "Analysis of accelerated gossip algorithms," *Automatica*, vol. 49, no. 4, pp. 873–883, 2013.

[9] Alexandros G Dimakis, Anand D Sarwate, and Martin J Wainwright, "Geographic gossip: Efficient averaging for sensor networks," *Signal Processing, IEEE Transactions on*, vol. 56, no. 3, pp. 1205–1216, 2008.

[10] Tuncer C Aysal, Mehmet E Yildiz, Anand D Sarwate, and Anna Scaglione, "Broadcast gossip algorithms for consensus," *Signal Processing, IEEE Transactions on*, vol. 57, no. 7, pp. 2748–2761, 2009.

[11] Alex Olshevsky and John N Tsitsiklis, "Convergence speed in distributed consensus and averaging," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 33–55, 2009.

[12] Robert M Gower and Peter Richtárik, "Stochastic dual ascent for solving linear systems," *arXiv preprint arXiv:1512.06890*, 2015.

[13] Thomas Strohmer and Roman Vershynin, "A randomized Kaczmarz algorithm with exponential convergence," *Journal of Fourier Analysis and Applications*, vol. 15, no. 2, pp. 262–278, 2009.

[14] Deanna Needell and Joel A Tropp, "Paved with good intentions: analysis of a randomized block Kaczmarz method," *Linear Algebra and its Applications*, vol. 441, pp. 199–221, 2014.

[15] Anastasios Zouzias and Nikolaos M Freris, "Randomized extended Kaczmarz for solving least squares," *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 2, pp. 773–793, 2013.

[16] Yonina C Eldar and Deanna Needell, "Acceleration of randomized Kaczmarz method via the Johnson–Lindenstrauss lemma," *Numerical Algorithms*, vol. 58, no. 2, pp. 163–177, 2011.

[17] Ji Liu and Stephen Wright, "An accelerated randomized Kaczmarz algorithm," *Mathematics of Computation*, vol. 85, no. 297, pp. 153–178, 2016.

[18] Deanna Needell, Ran Zhao, and Anastasios Zouzias, "Randomized block Kaczmarz method with projection for solving least squares," *Linear Algebra and its Applications*, vol. 484, pp. 322–343, 2015.

[19] Dennis Leventhal and Adrian S Lewis, "Randomized methods for linear constraints: convergence rates and conditioning," *Mathematics of Operations Research*, vol. 35, no. 3, pp. 641–654, 2010.

[20] Peter Richtárik and Martin Takáč, "Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function," *Mathematical Programming*, vol. 144, no. 2, pp. 1–38, 2014.

[21] Zheng Qu, Peter Richtárik, Martin Takáč, and Olivier Fercoq, "SDNA: Stochastic dual Newton ascent for empirical risk minimization," in *Proceedings of The 33rd International Conference on Machine Learning*, 2016, pp. 1823–1832.

[22] Robert M Gower and Peter Richtárik, "Randomized iterative methods for linear systems," *SIAM Journal on Matrix Analysis and Applications*, vol. 36, no. 4, pp. 1660–1690, 2015.