Incremental Reshaped Wirtinger Flow and Its Connection to Kaczmarz Method

Huishuai Zhang
Department of EECS
Syracuse University
Syracuse, NY 13244
hzhan23@syr.edu

Yingbin Liang
Department of EECS
Syracuse University
Syracuse, NY 13244
yliang06@syr.edu

Yuejie Chi
Department of ECE
The Ohio State University
Columbus, OH 43210
chi.97@osu.edu

Abstract

We study the problem of recovering a vector $\boldsymbol{x} \in \mathbb{R}^n$ from its magnitude measurements $y_i = |\langle \boldsymbol{a}_i, \boldsymbol{x} \rangle|, i = 1, ..., m$. We design an incremental/stochastic gradient-like algorithm, referred to as incremental reshaped Wirtinger flow (IRWF), based on minimizing a nonconvex nonsmooth loss function, and show that such an algorithm converges linearly to the true signal. We further establish performance guarantee of an existing Kaczmarz method for solving the same problem based on its connection to IRWF. We demonstrate IRWF outperforms previously developed batch algorithms as well as other incremental algorithms.

1 Introduction

Many problems in machine learning and signal processing can be reduced to solve an unknown signal for a quadratic system of equations, e.g., phase retrieval. Mathematically, the problem is formulated below.

Problem 1. *Recover $\boldsymbol{x} \in \mathbb{R}^n/\mathbb{C}^n$ from measurements y_i given by*

$$y_i = |\langle \boldsymbol{a}_i, \mathbf{x} \rangle|, \quad \textit{for } i = 1, \cdots, m, \tag{1}$$

where $\boldsymbol{a}_i \in \mathbb{R}^n/\mathbb{C}^n$ are random design vectors (known).

Recently, efficient nonconvex approaches such as AltMinPhase [2], Wirtinger flow (WF) [3] and truncated Wirtinger flow (TWF) [4] have been proposed to solve the above problem. Specifically, WF minimizes a nonconvex loss function via gradient descent together with a spectral initialization step, and is shown to recover the true signal with only $\mathcal{O}(n \log n)$ Gaussian measurements and attains $\epsilon-$accuracy within $\mathcal{O}(mn^2 \log 1/\epsilon)$ flops. TWF algorithm further improves the sample complexity to $\mathcal{O}(n)$ and the convergence time to $\mathcal{O}(mn \log 1/\epsilon)$ by truncating bad-behaved measurements when calculating the initial seed and the gradient.

The reshaped Wirtinger flow (RWF) [1] designed the loss function based on $|\boldsymbol{a}_i^T \boldsymbol{z}|$ rather than on its quadratic counterpart $|\boldsymbol{a}_i^T \boldsymbol{z}|^2$ used in WF and TWF. Although the loss function is not smooth everywhere, it reduces the order of $\boldsymbol{a}_i^T \boldsymbol{z}$ to be two, and the general curvature can be more amenable to convergence of the gradient method. Specifically, it minimizes the following the loss function

$$\ell(\boldsymbol{z}) := \frac{1}{2m} \sum_{i=1}^{m} \left(|\boldsymbol{a}_i^T \boldsymbol{z}| - y_i \right)^2. \tag{2}$$

For such a nonconvex and nonsmooth loss function, RWF [1] developed a gradient descent-like algorithm with a spectral initialization.

On the other hand, incremental/stochastic methods, e.g., Kaczmarz methods [5, 6] and incremental truncated Wirtinger flow (ITWF) [7], are proposed to solve Problem 1. Specifically, Kaczmarz methods are shown to have superb empirical performance, but no global convergence guarantee is established.

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

In this paper, we consider the incremental/stochastic version of RWF (i.e., IRWF) and show that IRWF converges to the true signal geometrically under an appropriate initialization. Interestingly, we further establish the connection between IRWF and Kaczmarz methods: the randomized Kaczmarz method can be seen as IRWF with a specific rule of choosing step size, which implies its global convergence. Empirically we demonstrate that IRWF often performs better than other competitors.

Throughout the paper, boldface lowercase letters such as $\boldsymbol{a}_i, \boldsymbol{x}, \boldsymbol{z}$ denote vectors, and boldface capital letters such as $\boldsymbol{A}, \boldsymbol{Y}$ denote matrices. For a complex matrix or vector, $\boldsymbol{A}^*$ and $\boldsymbol{z}^*$ denote conjugate transposes of $\boldsymbol{A}$ and $\boldsymbol{z}$ respectively. For a real matrix or vector, $\boldsymbol{A}^T$ and $\boldsymbol{z}^T$ denote transposes of $\boldsymbol{A}$ and $\boldsymbol{z}$ respectively. The indicator function $\mathbf{1}_A = 1$ if the event A is true, and $\mathbf{1}_A = 0$ otherwise. The Euclidean distance between two vectors up to a global phase difference [3] is, for complex signals,

$$\text{dist}(\boldsymbol{z}, \boldsymbol{x}) := \min_{\phi \in [0, 2\pi)} \|\boldsymbol{z} e^{-j\phi} - \boldsymbol{x}\|, \tag{3}$$

where it is simply $\min \|\boldsymbol{z} \pm \boldsymbol{x}\|$ for real case.

2 Incremental Reshaped Wirtinger Flow: Algorithm and Convergence

In this section, we first describe the algorithm of minibatch incremental reshaped Wirtinger flow (minibatch IRWF) and then establish its performance guarantee. In the end, we draw the connection between IRWF and Kaczmarz method.

Algorithm 1 Minibatch Incremetnal Reshaped Wirtinger Flow (minibatch IRWF)

Input: $\boldsymbol{y} = \{y_i\}_{i=1}^m$, $\{\boldsymbol{a}_i\}_{i=1}^m$;

Initialization: Let $\boldsymbol{z}^{(0)} = \lambda_0 \tilde{\boldsymbol{z}}$, where $\lambda_0 = \frac{mn}{\sum_{i=1}^m \|\boldsymbol{a}_i\|_1} \cdot \left(\frac{1}{m}\sum_{i=1}^m y_i\right)$ and $\tilde{\boldsymbol{z}}$ is the leading eigenvector of

$$\boldsymbol{Y} := \frac{1}{m}\sum_{i=1}^m y_i \boldsymbol{a}_i \boldsymbol{a}_i^* \mathbf{1}_{\{\alpha_l \lambda_0 < y_i < \alpha_u \lambda_0\}}.$$

Gradient loop: for $t = 0 : T-1$ do
Sample Γ_t uniformly at random from the subsets of $\{1, 2, \ldots, m\}$ with cardinality k

$$\boldsymbol{z}^{(t+1)} = \boldsymbol{z}^{(t)} - \mu \boldsymbol{A}_{\Gamma_t}^* \left(\boldsymbol{A}_{\Gamma_t}\boldsymbol{z}^{(t)} - y_{\Gamma_t} \odot \text{Ph}(\boldsymbol{A}_{\Gamma_t}\boldsymbol{z}^{(t)})\right), \tag{4}$$

where $\boldsymbol{A}_{\Gamma_t}$ is a matrix which stacks $\boldsymbol{a}_i^*$ for $i \in \Gamma_t$ as its rows, $\boldsymbol{y}_{\Gamma_t}$ is a vector which stacks y_i for $i \in \Gamma_t$ as its elements, $\odot$ denotes element-wise product, and $\text{Ph}(\boldsymbol{z})$ returns a phase vector of $\boldsymbol{z}$.
Output $\boldsymbol{z}^{(T)}$.

We set parameters $\alpha_l = 1, \alpha_u = 5$ and $\mu = 1/n$ in practice.

Due to the space limitation, we provide the minibatch IRWF only, which reduces to IRWF if the minibatch size $k = 1$. Compared to ITWF [7], IRWF does not employ any truncation in gradient loops and hence is easier to implement. We characterize the convergence of minibatch IRWF as follows.

Theorem 1. *Consider solving Problem 1 and assume that $\boldsymbol{a}_i \sim \mathcal{N}(0, I)$ are independent. There exist some universal constants $0 < \rho, \rho_0, \nu < 1$ and $c_0, c_1, c_2 > 0$ such that if $m \geq c_0 n$ and $\mu = \rho_0/n$, then with probability at least $1 - c_1 \exp(-c_2 m)$, Algorithm 1 yields*

$$\text{E}_{\Gamma^t}\left[dist^2(\boldsymbol{z}^{(t)}, \boldsymbol{x})\right] \leq \nu \left(1 - \frac{k\rho}{n}\right)^t \|\boldsymbol{x}\|^2, \quad \forall t \in \mathbb{N}, \tag{5}$$

where $\text{E}_{\Gamma^t}[\cdot]$ denotes the expectation with respect to algorithm randomness $\Gamma^t = \{\Gamma_1, \Gamma_2, \ldots, \Gamma_t\}$ conditioned on the high probability event of random measurements $\{\boldsymbol{a}_i\}_{i=1}^m$.

We note that Theorem 1 establishes that IRWF achieves linear convergence to the global optimum. It is not anticipated that incremental/stochastic first order method achieves linear convergence for general objectives due to the variance of stochastic gradient. However, for our specific problem (1), the variance of stochastic gradient reduces as the estimate approaches the true solution, and hence a fixed step size can be employed and linear convergence can be established (see [7] for similar justification). Another intuition comes from a side fact [8, 9] that stochastic gradient method

yields linear convergence to the minimizer $\boldsymbol{x}_\star$ when the objective $F(\boldsymbol{x}) = \sum_i f_i(\boldsymbol{x})$ is a smooth and strongly convex function and $\boldsymbol{x}_\star$ minimizes all components $f_i(\boldsymbol{x})$. Although our objective (2) is neither convex nor smooth, the summands share a same minimizer. We here establish a similar result for a nonconvex and nonsmooth objective.

3 Connection to the Kaczmarz method

[5] proposed a *Kaczmarz method* to solve Problem 1, which exhibits superb empirical performance in terms of sample complexity and convergence time. However, theoretical guarantee of Kaczmarz methods is not satisfying to date. We next draw the connection between IRWF and Kaczmarz method, and establish a linear convergence guarantee for Kaczmarz method motivated by the result of IRWF. We also note that [9] established a similar connection between Kaczmarz method and stochastic gradient method when solving the least-squares problem.

The Kaczmarz method (Algorithm 3 in [5]) employs the following update rule

$$\boldsymbol{z}^{(t+1)} = \boldsymbol{z}^{(t)} - \frac{1}{\|\boldsymbol{a}_{i_t}\|^2}\left(\boldsymbol{a}_{i_t}^*\boldsymbol{z}^{(t)} - y_{i_t}\cdot\frac{\boldsymbol{a}_{i_t}^*\boldsymbol{z}^{(t)}}{|\boldsymbol{a}_{i_t}^*\boldsymbol{z}^{(t)}|}\right)\boldsymbol{a}_{i_t}, \tag{6}$$

where i_t is selected either in a deterministic manner or randomly. We focus on the randomized Kaczmarz method where i_t is selected uniformly at random.

Comparing (6) with (4), the step size μ in (4) is replaced by $\frac{1}{\|\boldsymbol{a}_{i_t}\|^2}$ in (6). These two update rules are close if μ is set to be $\mu = \frac{1}{n}$, because $\|\boldsymbol{a}_{i_t}\|^2$ concentrates around n by law of large numbers. As we demonstrate in empirical results (see Table 1), these two methods have similar performance as anticipated. Thus, following the convergence result Theorem 1 for IRWF, we have the convergence guarantee for the randomized Kaczmarz method as follows.

Corollary 1. *Assume the measurement vectors are independent and each $\boldsymbol{a}_i \sim \mathcal{N}(0, \boldsymbol{I})$. There exist some universal constants $0 < \rho$ and $c_0, c_1, c_2 > 0$ such that if $m \geq c_0 n$, then with probability at least $1 - c_1 m\exp(-c_2 n)$, randomized Kaczmarz update rule* (6) *yields*

$$\mathrm{E}_{i_t}\left[\mathit{dist}^2(\boldsymbol{z}^{(t+1)}, \boldsymbol{x})\right] \leq \left(1 - \frac{\rho}{n}\right)\cdot \mathit{dist}^2(\boldsymbol{z}^{(t)}, \boldsymbol{x}) \tag{7}$$

holds for all $\boldsymbol{z}^{(t)}$ satisfying $\frac{\mathit{dist}(\boldsymbol{z}^{(t)}, \boldsymbol{x})}{\|\boldsymbol{z}\|} \leq \frac{1}{10}$.

The above corollary implies that once the estimate $\boldsymbol{z}^{(t)}$ enters the neighborhood of true solutions (often referred as to *basin of attraction*), the error shrinks geometrically by each update in expectation.

We can similarly draw the connection between the *block* Kaczmarz method and the minibatch IRWF, where the update rule of the block Kaczmarz method is given by

$$\boldsymbol{z}^{(t+1)} = \boldsymbol{z}^{(t)} - \boldsymbol{A}_{\Gamma_t}^{\dagger}\left(\boldsymbol{A}_{\Gamma_t}\boldsymbol{z}^{(t)} - y_{\Gamma_t}\odot \mathrm{Ph}(\boldsymbol{A}_{\Gamma_t}\boldsymbol{z}^{(t)})\right), \tag{8}$$

where $\dagger$ represents *Moore–Penrose pseudoinverse* and other notations follow those in Algorithm 1.

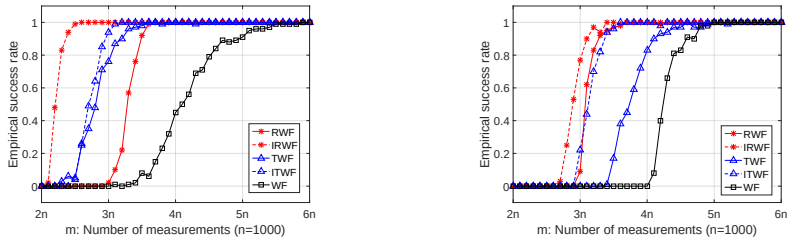
On the other hand, since block Kaczmarz method needs to calculate the matrix inverse or to solve an inverse problem, the block size cannot be too large. In contrast, minibatch IRWF works well for a wide range of batch sizes which can even vary with the signal dimension n as long as a batch of data is loadable into memory.

4 Numerical Comparison with Other Algorithms

In this section, we demonstrate the numerical efficiency of IRWF by comparing its performance with other competitive algorithms. Our experiments are run for both real Gaussian and complex Gaussian cases. All the experiments are implemented in Matlab and carried out on a computer equipped with Intel Core i7 3.4GHz CPU and 12GB RAM.

We first compare the sample complexity of IRWF with those of RWF, TWF, WF and ITWF via the empirical successful recovery rate versus the number of measurements. For IRWF, we adopt a minibatch size 64 and follow Algorithm 1 with suggested parameters. For RWF, TWF, ITWF and WF, we use the codes provided in the original papers with the suggested parameters. We conduct the experiment for real Gaussian, complex Gaussian respectively. We set the signal dimension n to be 1024, and the ratio m/n take values from 2 to 6 by a step size 0.1. For each m, we run 100 trials

and count the number of successful trials. For each trial, we run a fixed number of iterations/passes $T = 1000$ for all algorithms. A trial is declared to be successful if $\boldsymbol{z}^{(T)}$, the output of the algorithm, satisfies $\text{dist}(\boldsymbol{z}^{(T)}, \boldsymbol{x})/\|\boldsymbol{x}\| \leq 10^{-5}$. For the real Gaussian case, we generate signal $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, and the measurement vectors $\boldsymbol{a}_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ i.i.d. for $i = 1, \ldots, m$. For the complex Gaussian case, we generate signal $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I}) + j\mathcal{N}(0, \boldsymbol{I})$ and measurement vectors $\boldsymbol{a}_i \sim \frac{1}{2}\mathcal{N}(0, \boldsymbol{I}) + j\frac{1}{2}\mathcal{N}(0, \boldsymbol{I})$ i.i.d. for $i = 1, \ldots, m$.



(a) Real Gaussian case (b) Complex Gaussian case

Figure 1: Comparison of sample complexity among RWF, IRWF, TWF, ITWF and WF.

Figure 1 plots the fraction of successful trials out of 100 trials for all algorithms, with respect to m. It can be seen that IRWF exibits the best sample complexity for both cases, which is barely above the theoretical identifiable bound [10]. This is because the inherent noise in IRWF helps to escape bad local minima, which is helpful in the regime of small sample size where local minima do exist near the global ones. See Section 5 for more intuition.

Table 1: Comparison of iteration count and time cost among algorithms ($n = 1024, m = 8n$)

		Real Gaussian		Complex Gaussian	
		#passses	time(s)	# passes	time(s)
Batch	RWF	72	**0.52**	177	**4.81**
	TWF	182	1.30	484	13.5
methods	WF	217	2.22	922	24.9
Incremental	minibatch IRWF (64)	9	**0.28**	**21**	**1.53**
	minibatch ITWF (64)	15	0.72	28.6	3.28
methods	block Kaczmarz (64)	**8**	0.45	21	3.22

We next compare the convergence rate of minibatch IRWF with those of Kaczmarz, RWF, TWF, ITWF and WF. We run all algorithms with suggested parameter settings in the original papers. We set the minibatch size/block size to be 64 for incremental methods. We generate signal and measurements in the same way as those in the first experiment. All algorithms are seeded with minibatch IRWF initialization. In Table 1, we list the number of passes and time cost for those algorithms to achieve the relative error of 10^{-14} averaged over 10 trials. Clearly, minibatch IRWF achieves the best computational complexity for both cases. Moreover for full gradient algorithms, RWF outperforms TWF and WF in terms of passes and running time.

5 Further Direction

One interesting direction is to study the convergence of algorithms without spectral initialization. In the regime of large sample size ($m \gg n$), the empirical loss surface approaches the asymptotic loss for which all local minimums are global optimal. It is understandable that pure gradient descent converges to these minimizers from random starting point due to the result in [11]. Similar phenomenon has been observed in [12]. However, under moderate number of measurements ($m < 10n$), local minimums do exist which often locate not far from the global ones. In this regime, the batch gradient method often fails to converge to global optimums when starting from random points. As always believed, stochastic algorithms are efficient in escaping bad local minimums or saddle points in nonconvex optimization because of the inherent noise [13, 14]. We observe that IRWF and block IRWF without spectral initialization still converge to global optimums even with very small sample size which is close to the theoretical limits [10]. Therefore, stochastic methods do escape these local minimums (not just saddle points) efficiently. Further study of stochastic method with random initialization in non-convex non-smooth setting is of great interest.

References

[1] Huishuai Zhang and Yingbin Liang. Reshaped wirtinger flow for solving quadratic systems of equations. *arXiv preprint arXiv:1605.07719*, 2016.

[2] P. Netrapalli, P. Jain, and S. Sanghavi. Phase retrieval using alternating minimization. *Advances in Neural Information Processing Systems (NIPS)*, 2013.

[3] E. J. Candès, X. Li, and M. Soltanolkotabi. Phase retrieval via wirtinger flow: Theory and algorithms. *IEEE Transactions on Information Theory*, 61(4):1985–2007, 2015.

[4] Yuxin Chen and Emmanuel Candes. Solving random quadratic systems of equations is nearly as easy as solving linear systems. In *Advances in Neural Information Processing Systems 28*, pages 739–747. 2015.

[5] Ke Wei. Solving systems of phaseless equations via kaczmarz methods: a proof of concept study. *Inverse Problems*, 31(12):125008, 2015.

[6] Gen Li, Yuantao Gu, and Yue M Lu. Phase retrieval using iterative projections: Dynamics in the large systems limit. In *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1114–1118. IEEE, 2015.

[7] Ritesh Kolte and Ayfer Özgür. Phase retrieval via incremental truncated wirtinger flow. *arXiv preprint arXiv:1606.03196*, 2016.

[8] Eric Moulines and Francis R Bach. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In *Advances in Neural Information Processing Systems*, pages 451–459, 2011.

[9] Deanna Needell, Nathan Srebro, and Rachel Ward. Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm. *Mathematical Programming*, 155(1-2):549–573, 2016.

[10] Zhiqiang Xu. The minimal measurement number for low-rank matrices recovery. *arXiv preprint arXiv:1505.07204*, 2015.

[11] Jason D Lee, Max Simchowitz, Michael I Jordan, and Benjamin Recht. Gradient descent converges to minimizers. *arXiv preprint arXiv:1602.04915*, 2016.

[12] Ju Sun, Qing Qu, and John Wright. A geometric analysis of phase retrieval. *arXiv preprint arXiv:1602.06664*, 2016.

[13] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points—online stochastic gradient for tensor decomposition. *arXiv preprint arXiv:1503.02101*, 2015.

[14] Christopher De Sa, Kunle Olukotun, and Christopher Ré. Global convergence of stochastic gradient descent for some non-convex matrix problems. *arXiv preprint arXiv:1411.1134v3*, 2015.