RECOVERING SIGNALS WITH VARIABLE SPARSITY LEVELS FROM THE NOISY 1-BIT COMPRESSIVE MEASUREMENTS

Amin Movahed[⋆] *Ashkan Panahi*[†] *Mark C. Reed*[⋆]

[⋆] School of Eng. and Information Tech., The University of New South Wales, Canberra, Australia
[†]Department of Signals and Systems, Chalmers University of Tech., Gothenburg, Sweden

ABSTRACT

In this paper, we consider the 1-bit compressive sensing reconstruction problem in a scenario that the sparsity level of the signal is unknown and time variant, and the binary measurements are contaminated with the noise. We introduce a new reconstruction algorithm which we refer to as *Noise-Adaptive Restricted Step Shrinkage* (NARSS). NARSS is superior in terms of performance, complexity and speed of convergence to the algorithms already introduced in the literature for 1-bit compressive sensing reconstruction from the noisy binary measurements.

***Index Terms*—** one bit quantization, compressive sensing (CS).

1. INTRODUCTION

Compressive sensing (CS) is a new method of signal acquisition in which certain signals can be sampled and perfectly reconstructed at a rate significantly below the Nyquist rate [1,2]. In CS, the signal is acquired by few non-adaptive linear projections. The signal is reconstructed from these projections typically using an optimization process [2]. The exact signal recovery is guaranteed when the matrix representing the linear projections has nearly orthonormal columns and satisfies the *restricted isometry property* [3], and the signal is sufficiently sparse, i.e. most of the elements in the signal vector are zero or near-zero [1, 3].

There is a variety of approaches to reconstruct a signal from compressive measurements [4–8]. They are generally based on the assumption that the obtained measurements have infinite bit precision, i.e., they can be any real-number. In practice, however, the obtained measurements need to be quantized in order to be stored and/or transmitted through a channel. In this context, *1-bit CS* is referred to a case where CS measurements are quantized by a one bit quantizer, i.e., the measurements are represented by only two alternative levels, e.g., -1 and $+1$. Since each measurement is shown by a single bit, more measurements can be afforded for a given bit budget than in the conventional quantized CS approaches [9]. The one bit quantizer is basically a simple comparator and very fast and easy to implement. Therefore, 1-bit CS is highly favourable in high-speed signal processing applications [10, 11].

Based on convex optimization techniques, some interesting 1-bit CS approaches are discussed in [12, 13]. In this work, the focus is on the iterative 1-bit CS reconstruction algorithms due to their simple implementation. There are various other iterative reconstruction algorithms for 1-bit CS in the literature, e.g., *Binary iterative hard thresholding* (BIHT) [14], *renormalized fixed point iterative* (RFPI) [15] and *restricted step shrinkage* (RSS) [16]. While BIHT needs to know the sparsity level of the signal, RFPI and RSS do not need such *a priori* information.

In many applications, the binary measurements are transmitted through a noisy channel to the reconstruction part. The channel noise causes errors in the transmitted binary data which can be modelled as random bit flips. In addition, the error in the measurement process may also cause random bit flips [17]. In the presence of the random bit flips, BIHT, RFPI and RSS have poor reconstruction performance. Then, *Adaptive outlier pursuit with bit flips* (AOP-f) [17] and *noise-adaptive renormalized fixed point iterative* (NARFPI) [18] have been introduced to cope with the random bit flip effect. In the case that the sparsity level of the signal is unknown and time-variant, NARFPI outperforms AOP-f since it does not need the sparsity level of the signal as an input. However, NARFPI has a high computational cost and its performance is strongly dependent on the choice of the initial point.

In this paper, we introduce a new reconstruction method which we refer to as *noise-adaptive restricted step shrinkage* (NARSS). Through numerical results, we show that NARSS outperforms NARFPI in terms of complexity, output performance and convergence speed.

2. 1-BIT COMPRESSIVE SENSING SETUP

Consider a signal represented by a vector $\mathbf{x} \in \mathbb{R}^N$. The vector $\mathbf{x}$ is called K-sparse when there are only K non-zero elements in $\mathbf{x}$. In addition, consider a finite set $\{y_1, y_2, \dots, y_M\}$ of linear projections of $\mathbf{x}$ each obtained by

$$y_i = \Phi_i^T \mathbf{x} \text{ and } i = 1, \dots, M \tag{1}$$

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

where Φ_i is the ith projection vector. The M linear projections of the signal can be shown by $\mathbf{y}^T = (y_1, y_2, \ldots, y_M)$. Hence, we have

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x} \tag{2}$$

where the ith row of matrix $\mathbf{\Phi} \in \mathbb{R}^{N\times M}$ is Φ_i^T. The binary measurements vector $\mathbf{b}$ is further obtained by

$$\mathbf{b} = \text{sign}(\mathbf{y}) \tag{3}$$

where $\mathbf{y}$ is given in (2). When the binary measurements are transmitted through a noisy channel, the elements of the vector $\mathbf{b}$ might flip randomly. Let us denote the received bits vector by $\tilde{\mathbf{b}}$ and define L as the number of bit flips. Therefore, L can be derived by computing the negative elements of $\mathbf{b} \odot \tilde{\mathbf{b}}$ where $\odot$ denotes element-wise product.

The goal is to reconstruct $\mathbf{x}$ from the received binary measurements $\tilde{\mathbf{b}}$, with the knowledge of L (or an estimate of L) and without any information about the sparsity level K of $\mathbf{x}$.

3. NOISE-ADAPTIVE RESTRICTED STEP SHRINKAGE

The main contribution of this paper is to propose a reconstruction algorithm for 1-bit CS problem explained in Section 2. As discussed in Section 1, RSS is designed for the case that $\tilde{\mathbf{b}} = \mathbf{b}$ (no bit flips). In the first part of this section, we review RSS [16] and in the second part, we modify RSS and introduce NARSS which is robust against the bit flips.

3.1. Restricted Step Shrinkage (RSS)

In the case that there are no bit flips, the received binary measurements are identical to (3). This implies that

$$\mathbf{b} \odot \mathbf{\Phi}\mathbf{x} \succeq \mathbf{0} \tag{4}$$

where $\succeq$ denotes element-wise inequality. Since the ℓ_1 minimization is shown to consistently result in a sparse solution [1], RSS searches for a signal estimate with the smallest ℓ_1-norm when (4) holds. Accordingly, RSS is defined as an algorithm which solves the following optimization

$$\begin{aligned} &\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \\ &\text{subject to} \quad \mathbf{b} \odot \Phi\mathbf{x} \succeq \mathbf{0} \\ &\text{and} \quad \|\mathbf{x}\|_2 = 1 \end{aligned} \tag{5}$$

where the ℓ_2-norm constraint is an energy normalization to avoid the trivial solution $\hat{\mathbf{x}} = \mathbf{0}$. To solve (5) efficiently, it is suggested in [16] to apply the augmented Lagrangian method, and update $\hat{\mathbf{x}}$ iteratively through solving the following minimization:

$$\begin{aligned} \hat{\mathbf{x}}^{n+1} &= \arg\min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{b}, \lambda^n, \mu^n) \\ &\text{subject to} \quad \|\mathbf{x}\|_2 = 1 \end{aligned} \tag{6}$$

where $\lambda^n \in \mathbb{R}^M$, $\mu^n > 0$ and $\mathbf{x}^n$ is the value of $\mathbf{x}$ in the nth iteration. The Lagrangian function $\mathcal{L}$ in (6) is

$$\mathcal{L}(\mathbf{x}, \mathbf{b}, \lambda, \mu) = \|\mathbf{x}\|_1 + \sum_{i=1}^{M} \nu\left([\mathbf{b} \odot \Phi\mathbf{x}]_i, [\lambda]_i, \mu\right) \tag{7}$$

where

$$\nu(t, \alpha, \mu) = \begin{cases} -\alpha t + \frac{1}{2}\mu t^2, & \text{if } t - \frac{\alpha}{\mu} \leq 0, \\ -\frac{1}{2\mu}\alpha^2, & \text{otherwise} \end{cases} \tag{8}$$

and $[\cdot]_i$ denotes the ith element of its argument. The RSS algorithm solves (6) (RSS-Inner). Then, based on the current estimate of the signal, λ and μ are updated (RSS-Outer) by

$$\lambda^{n+1} = \max\left\{\lambda^n - \mu^n\left(\mathbf{b} \odot \mathbf{\Phi}\mathbf{x}^{n+1} - \mathbf{b}\right), \mathbf{0}\right\} \tag{9}$$

$$\mu^{n+1} = k\mu^n \tag{10}$$

where $k > 0$ is a tuning parameter.

3.2. Noise-Adaptive Restricted Step Shrinkage (NARSS)

In this section, we modify RSS to make it robust against the noise. Inspired by the method in NARFPI [18], we introduce the *bit flip detector vector*, $\mathbf{\Omega} \in \{-1, +1\}^M$ which is defined by

$$\mathbf{\Omega} = \mathbf{b} \odot \tilde{\mathbf{b}} \tag{11}$$

where $\tilde{\mathbf{b}}$ denotes the noisy binary measurements vector. In other words, the -1 elements in $\mathbf{\Omega}$ show the positions of the bit flips. Now, RSS can be modified by estimating $\mathbf{\Omega}$ and compensating the effect of the noise in the binary measurements vector. Thus, we propose NARSS as an algorithm solving

$$\begin{aligned} &\left(\hat{\mathbf{x}}, \hat{\mathbf{\Omega}}\right) = \arg\min_{\mathbf{x},\mathbf{\Omega}} \mathcal{L}\left(\mathbf{x}, \tilde{\mathbf{b}} \odot \mathbf{\Omega}, \lambda, \mu\right) \\ &\text{subject to} \quad \|\mathbf{x}\|_2 = 1 \\ &\text{and} \quad \frac{1}{2}\sum_i \left(1 - [\mathbf{\Omega}]_i\right) = \bar{L} \end{aligned} \tag{12}$$

where $\bar{L}$ is an estimate of L. The optimization in (12) can be solved by iterating between the following two steps:
Step one: the signal is estimated when $\hat{\mathbf{\Omega}}$ is fixed to the value obtained from the previous iteration. Therefore, we have

$$\begin{aligned} \hat{\mathbf{x}}^{n+1} &= \arg\min_{\mathbf{x}} \mathcal{L}\left(\mathbf{x}, \tilde{\mathbf{b}} \odot \hat{\mathbf{\Omega}}^n, \lambda^n, \mu^n\right) \\ &\text{subject to} \quad \|\mathbf{x}\|_2 = 1. \end{aligned} \tag{13}$$

This is similar to (6) when $\mathbf{b}$ is replaced by $\tilde{\mathbf{b}} \odot \hat{\mathbf{\Omega}}$. Thus, (13) can be solved through RSS-Inner [16].
Step two: the optimal $\hat{\mathbf{\Omega}}$ is updated based on $\hat{\mathbf{x}}$ obtained from (13). Hence,

$$\begin{aligned} \hat{\mathbf{\Omega}}^{n+1} &= \arg\min_{\mathbf{\Omega}} \mathcal{L}\left(\hat{\mathbf{x}}^{n+1}, \tilde{\mathbf{b}} \odot \mathbf{\Omega}, \lambda^n, \mu^n\right) \\ &\text{subject to} \quad \frac{1}{2}\sum_i \left(1 - [\mathbf{\Omega}]_i\right) = \bar{L} \end{aligned} \tag{14}$$

Algorithm 1 NARSS

1. **Inputs:** vector of 1-bit measurements $\tilde{\mathbf{b}} \in \{\pm 1\}^M$, measuring matrix Φ, estimated number of bit flips $\bar{L}$
2. **Initialization:** $\hat{\mathbf{x}}^0$ and $\hat{\mathbf{\Omega}}^0$ are set to all-one vector, $\mu^0 = 100$, $k > 0$, $\lambda^0 = \mathbf{0}$ and $\mathbf{b} = \tilde{\mathbf{b}}$
3. **Iteration:** For $n = 0, \ldots, l$
   (a) **Estimate the signal via RSS-Inner:** $\hat{\mathbf{x}}^{n+1} = \arg\min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{b}, \lambda^n, \mu^n)$ subject to $\|\mathbf{x}\|_2 = 1$
   (b) **Estimate the position of the bit flips:** Find $\hat{\mathbf{\Omega}}^{n+1}$ from (15). $\mathbf{b} \leftarrow \mathbf{b} \odot \hat{\mathbf{\Omega}}^{n+1}$
   (c) **Initialize next iteration:** $\lambda^{n+1} \leftarrow \max\left\{\lambda^n - \mu^n \left(\mathbf{b} \odot \mathbf{\Phi}\hat{\mathbf{x}}^{n+1} - \mathbf{b}\right), \mathbf{0}\right\}$, $\mu^{n+1} \leftarrow k\mu^n$
4. **Output:** $\hat{\mathbf{x}} = \hat{\mathbf{x}}^{l+1}$

which can be solved by the following result.

Theorem 1 *The optimal $\hat{\mathbf{\Omega}}^{n+1}$ in (14) is given by*

$$\left[\hat{\Omega}^{n+1}\right]_i = \begin{cases} -1, & \text{if } [\varepsilon_- - \varepsilon_+]_i \leq \beta, \\ +1, & \text{otherwise} \end{cases} \tag{15}$$

where

$$\varepsilon_{\pm} = \left(\mu^n \left(\pm\tilde{\mathbf{b}} \odot \Phi \mathbf{x}^{n+1}\right) - \lambda^n\right)^- \tag{16}$$

$$\left[(\mathbf{x})^-\right]_i = \begin{cases} |[\mathbf{x}]_i|, & \text{if } [\mathbf{x}]_i < 0, \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

and β is the $\bar{L}$th smallest element in $\varepsilon_- - \varepsilon_+$.

A proof of Theorem 1 is given in the Appendix.

In summary, the subroutines of NARSS have been depicted in Algorithm 1 where step 3-b together with Theorem 1 are the key contribution of this work. The other steps have been borrowed from RSS-outer in [16]. In the first iteration, $\hat{\mathbf{x}}^0$ and $\hat{\mathbf{\Omega}}^0$ are set to the all-one vectors.

4. NUMERICAL RESULTS

In this section, we investigate the reconstruction performance of NARSS through numerical simulations. In the following simulations, we set the signal vector length $N = 1000$ and the number of measurements $M = 2000$. The sparsity level of the signal is a random variable with a symmetric discrete triangular distribution with mean 10 and variance σ_K^2 (where $K \in [1, 19]$). The non-zero elements in the signal vector are derived from a zero-mean Gaussian distribution with unit variance and are distributed uniformly through the signal vector. The elements of the measuring matrix $\mathbf{\Phi}$ are independent random variables generated based on a zero-mean Gaussian distribution with variance $1/M$. We set the probability of the bit flips $P = 3\%$ and $\bar{L} = MP$. We show the quality of reconstruction in terms of *received signal to noise ratio* (RSNR) which is defined by

$$\text{RSNR} = \mathbb{E}\left(\|\mathbf{x}\|_2^2\right) / \mathbb{E}\left(\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2\right). \tag{18}$$

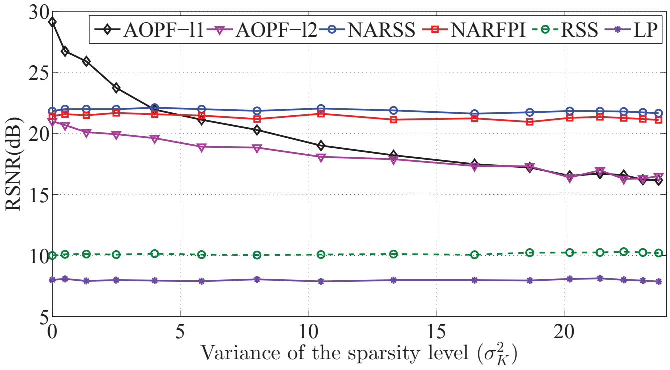

Fig. 1: Performance comparison of different algorithms when the sparsity level K is generated according to a triangular distribution with mean 10 and different variance σ_K^2.

First we compare the performance of NARSS with AOPf-ℓ_1, AOPf-ℓ_2 [17], NARFPI [18] and the linear program (LP) proposed in [12]. Note that AOP-f is a modified version of BIHT and its performance is an upper-bound for BIHT [17]. Therefore, we do not include BIHT in our simulations. As discussed in Section I, AOPf-ℓ_1 and AOPf-ℓ_2 need to know the sparsity level of the signal; However, we assume that the sparsity level is not known to the algorithms and its statistical distribution is the only available information. Therefore, for AOPf-ℓ_1 and AOPf-ℓ_2 we set the sparsity level K to its mean value, 10 as a reasonable estimate. Moreover, we fix the number of iterations in RSS-Inner to 100 and the number of outer iterations l to 20. To have a fair comparison, we choose the same number of iterations for the other algorithms. Figure 1 shows the performance of the algorithms versus the variance of the sparsity level ($\sigma_K^2 \in [0, 25]$) averaged over 100 realizations.

As it can be seen, there is a significant improvement in the performance of NARSS in comparison to that of RSS. In addition, as σ_K^2 increases the performance of AOPf-ℓ_1 and AOPf-ℓ_2 decreases dramatically. In contrast, the performance of NARFPI and NARSS has the same trend and does not depend on σ_K^2 [18]. NARSS outperforms the other algorithms for $\sigma_K^2 > 4$. Though NARSS marginally surpasses NARFPI in terms of performance, its significant computational advantages are shown in the next simulations. In the next experiment, we compare NARSS and NARFPI in terms of complexity and convergence rate. The maximum number of inner iterations in NARFPI and RSS-Inner is set to 100. In addition, the maximum number of outer iterations in NARFPI and NARSS is set to 20. We initialize NARFPI with two different values. In one case (NARFPI-i), similarly to NARSS, NARFPI is initialized with an all-one vector. In another case, NARFPI

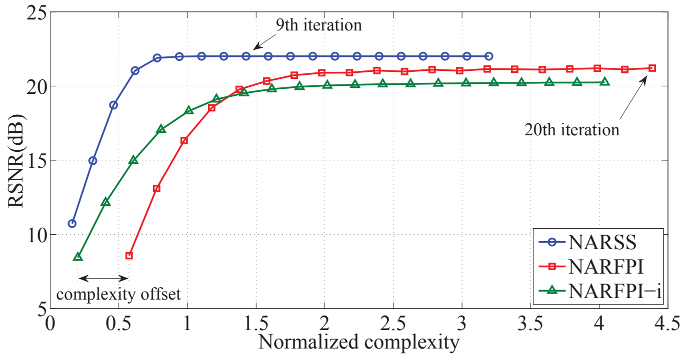

Fig. 2: Comparison of complexity and convergence speed of NARSS, NARFPI and NARFPI-i. NARSS and NARFPI convege after 9 and 20 iterations, respectively.

Table 1: Performance of NARSS and NARFPI in different noisy scenarios when $\sigma_K^2 = 24$.

Probability of bit flips	0.01%	0.1%	0.5%	1%	2%	3%
RSNR of NARSS (dB)	24.9	24.7	24.4	24	22.9	21.8
RSNR of NARFPI (dB)	24.6	24.3	23.6	23.2	22.1	21.2

is initialized by $\hat{\mathbf{x}}^0 = \mathbf{\Phi}^\dagger \tilde{\mathbf{b}} / \left\| \mathbf{\Phi}^\dagger \tilde{\mathbf{b}} \right\|_2$ where $\mathbf{\Phi}^\dagger$ denotes the pseudo-inverse of $\mathbf{\Phi}$, which introduces a relatively high computational burden. The normalized complexity of the algorithms is shown by their run time in MATLAB. Since built-in functions in MATLAB are highly optimized (especially for matrix calculation), the run time can be a relative comparator of the algorithm complexity. In Figure 2, the average run time versus reconstruction performance is illustrated for different number of iterations over 100 realizations when $\sigma_K^2 = 24$.

As Figure 2 shows, the performance of NARSS through iterations increases with a higher slope (faster convergence) in comparison to NARFPI and NARRPI-i. There is a significant offset in the run time of NARFPI compared to NARFPI-i due to the high complexity of the pseudo-inverse initialization. Comparing the two different initialization methods with different levels of complexity, Figure 2 highlights the trade off between the complexity and the reconstruction performance in NARFPI and NARFPI-i.

To investigate the reconstruction performance over different error rates, the reconstruction performance of NARSS and NARFPI for different bit flip probabilities in the expected error range of interest is shown in Table 1. As it is illustrated, NARSS performs considerably better than NARFPI.

5. CONCLUSION

To summarize, as we showed above, NARSS outperforms the other algorithms when the sparsity level of the signal deviates from its estimated value, and the signal is reconstructed from noisy binary measurements. In terms of complexity, NARSS surpasses NARFPI due to its less complex iterations and simpler initialization. In addition, NARSS converges to the optimal value with less number of iterations than the one in NARFPI.

Appendix: Proof of Theorem 1

We can write (14) as

$$\hat{\mathbf{\Omega}}^{n+1} = \arg\min_{\mathbf{\Omega}} \sum_{i=1}^{M} \nu \left(\left[\Omega \odot \tilde{\mathbf{b}} \odot (\Phi \hat{\mathbf{x}}^{n+1}) \right]_i, [\lambda^n]_i, \mu^n \right)$$
$$\text{subject to} \quad \frac{1}{2} \sum_i (1 - [\Omega]_i) = \bar{L}. \tag{19}$$

We define

$$\begin{aligned} \nu'(\mathbf{t}, \alpha, \mu) &= \mu\nu(\mathbf{t}, \alpha, \mu) - \frac{1}{2\mu}\alpha^2 \qquad (20) \\ &= \begin{cases} (\mu t - \alpha)^2, & \text{if } \mu t - \alpha \le 0, \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

Solving for ν in (20) and replacing in (19), after some straightforward simplifications, (19) can be written as

$$\hat{\mathbf{\Omega}}^{n+1} = \arg\min_{\mathbf{\Omega}} \left\| \left(\mu^n \left(\Omega \odot \tilde{\mathbf{b}} \odot (\Phi \hat{\mathbf{x}}^{n+1}) \right) - \lambda^n \right)^- \right\|_2^2$$
$$\text{subject to} \quad \frac{1}{2} \sum_i (1 - [\Omega]_i) = \bar{L}. \tag{21}$$

Since $\mu > 0$, the solution of (14) and (21) are identical. In order to solve combinatorial optimization (21), first we define

$$\mathbf{c} = \left(\mu^n \left(\Omega \odot \tilde{\mathbf{b}} \odot (\Phi \hat{\mathbf{x}}^{n+1}) \right) - \lambda^n \right)^- \tag{22}$$

and the set S including the position of the bit flips (i.e. $S = \left\{ i \mid \left[\hat{\mathbf{\Omega}}^{n+1} \right]_i = -1 \right\}$). In the rest of the proof, we equivalently look for the optimal S instead, which we refer to as $\hat{S}$. Since $[\mathbf{c}]_i \ge 0$ for all i, we can conclude from (21) that $\hat{S} = \arg\min_{S} \sum_{i=1}^{M} [\mathbf{c}]_i$ where $[\mathbf{c}]_i = \begin{cases} [\varepsilon_+]_i, & \text{if } i \notin S, \\ [\varepsilon_-]_i, & \text{if } i \in S \end{cases}$ and ε_+ and ε_- are given by (16). It can simply be seen that

$$\hat{S} = \arg\min_{S} \sum_{i=1}^{M} \Big[\varepsilon_+ + \underbrace{\mathbf{c} - \varepsilon_+}_{\varphi} \Big]_i = \arg\min_{S} \sum_{i=1}^{M} [\varphi]_i \tag{23}$$

where $[\varphi]_i = \begin{cases} 0, & \text{if } i \notin S, \\ [\varepsilon_- - \varepsilon_+]_i, & \text{if } i \in S. \end{cases}$ Therefore, to obtain $\hat{S}$ we need to find the $\bar{L}$ smallest elements of $\varepsilon_- - \varepsilon_+$. ■

Acknowledgement

The authors would like to thank Dr. J. Laska for the RSS software codes.

6. REFERENCES

[1] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[2] R. G. Baraniuk, "Compressive sensing [lecture notes]," *IEEE Signal Process. Mag.*, vol. 24, no. 4, pp. 118–121, Jul. 2007.

[3] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, no. 9, pp. 589–592, May 2008.

[4] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[5] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 1094–1121, Feb. 2012.

[6] D. Needell and R. Vershynin, "Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit," *Found. Comput. Math.*, vol. 9, no. 3, pp. 317–334, Jun. 2009.

[7] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmonic Anal.*

[8] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Appl. Comput. Harmonic Anal.*, vol. 27, no. 3, pp. 265–274, 2009.

[9] J. N. Laska and R. G. Baraniuk, "Regime change: Bit-depth versus measurement-rate in compressive sensing," *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3496–3505, Jul. 2012.

[10] P. T. Boufounos, "Reconstruction of sparse signals from distorted randomized measurements," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Dallas, Mar. 2010, pp. 3998–4001.

[11] A. Bourquard, F. Aguet, and M. Unser, "Optical imaging using binary sensors," *Optics Express*, vol. 18, no. 5, pp. 4876–4888, 2010.

[12] Y. Plan and R. Vershynin, "One-bit compressed sensing by linear programming," *Commun. Pure and Appl. Math.*, vol. 66, no. 8, pp. 1275–1297, 2013. [Online]. Available: http://dx.doi.org/10.1002/cpa.21442

[13] ——, "Robust 1-bit compressed sensing and sparse logistic regression: A convex programming approach," *IEEE Trans. Inf. Theory*, vol. 59, no. 1, pp. 482–494, Dec. 2012.

[14] L. Jacques, J. N. Laska, P. T. Boufounos, and R. G. Baraniuk, "Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors," *IEEE Trans. Inf. Theory*, vol. 59, no. 4, pp. 2082–2102, Apr. 2013.

[15] P. T. Boufounos and R. G. Baraniuk, "1-bit compressive sensing," in *Proc. Annual Conf. Inf. Sciences Syst. (CISS)*, Princeton, NJ, Mar. 2008, pp. 16–21.

[16] J. N. Laska, Z. Wen, W. Yin, and R. G. Baraniuk, "Trust, but verify: Fast and accurate signal recovery from 1-bit compressive measurements," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5289–5301, Nov. 2011.

[17] M. Yan, Y. Yang, and S. Osher, "Robust 1-bit compressive sensing using adaptive outlier pursuit," *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3868–3875, 2012.

[18] A. Movahed, A. Panahi, and G. Durisi, "A robust RFPI-based 1-bit compressive sensing reconstruction algorithm," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Laussane, Switzerland, Sep. 2012, pp. 567–571.