A Robust RFPI-Based 1-Bit Compressive Sensing Reconstruction Algorithm

Amin Movahed, Ashkan Panahi, Giuseppe Durisi

Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden

movahed@student.chalmers.se, ashkanp@chalmers.se, durisi@chalmers.se

Abstract—In this paper, we introduce a 1-bit compressive sensing reconstruction algorithm that is not only robust against bit flips in the binary measurement vector, but also does not require a priori knowledge of the sparsity level of signal to be reconstructed. Through numerical experiments, we show that our algorithm outperforms state-of-the-art reconstruction algorithms for the 1-bit compressive sensing problem in the presence of random bit flips and when the sparsity level of the signal deviates from its estimated value.

I. INTRODUCTION

Compressive sensing is an emerging method for signal acquisition in which the number of samples ensuring exact reconstruction of the signal to be acquired is far less than the one in the conventional Nyquist sampling approach [1]–[4]. In compressive sensing, the signal is acquired by means of few linear non-adaptive measurements, and then reconstructed by finding the sparsest solution of the resulting undetermined system of equations. Exact reconstruction is guaranteed when the matrix describing the system of equations satisfies the restricted isometry property and the signal is sufficiently sparse [4].

In the classic compressive sensing setup, each measurement outcome is described by a real value. In practice, for further processing and storage purposes, often the real-valued measurements need to be converted to finite-precision numbers. Thus, quantization after the measurement is virtually unavoidable. *Quantized compressive sensing* is the general term used to refer to linear non-adaptive measurements followed by quantization [5]–[8]. *1-bit compressive sensing* refers to the extreme case where the quantizer is a simple sign comparator and each measurement is represented using one bit only, i.e., $+1$ or -1 [5], [6], [9]–[14].

1-bit compressive sensing is appealing for hardware implementation, since a 1-bit quantizer is not only low-cost but also much faster than more sophisticated scalar quantizers [9]. In many applications in which high sampling rate is needed, e.g., imaging systems [15], 1-bit compressive sensing is an attractive solution because it does not suffer from dynamic range problems.

Several algorithms have been introduced in the literature for solving efficiently the reconstruction problem in the 1-bit compressive sensing setting [5], [6], [9]–[14]. In this paper, we shall mainly focus on *renormalized fixed point iteration* (RFPI) [9] and *binary iterative hard thresholding* (BIHT) [5], [12]. In order to reconstruct the signal perfectly,

BIHT needs as input the sparsity level of the signal (i.e., the number of its nonzero coefficients), while RFPI does not require such *a priori* information. These two algorithms have mostly been analyzed in the setting where the 1-bit quantized measurement vector is error-free, meaning that the reconstruction algorithm is fed with a 1-bit quantized version of the measurement vector. However, in many applications (e.g., wireless sensor networks) 1-bit quantized measurements need to be transmitted through a propagation channel to reach the device where signal reconstruction is performed and this process may introduce errors, i.e., bit flips.[1] Unfortunately, both RFPI and BIHT perform poorly in the presence of bit flips. Yan *et al.* recently introduced a modified version of BIHT, referred to as *adaptive outlier pursuit with sign flips* (AOP-f), that is robust against bit flips in the measurement vector [16]. Similarly to BIHT, also AOP-f requires *a priori* knowledge of the sparsity level of the signal. In many practical cases, however, the sparsity level of the signal is unknown and time variant.

In this paper, we introduce an algorithm, which we refer to as *noise-adaptive renormalized fixed point iteration* (NARFPI), for solving the reconstruction problem in the 1-bit compressive sensing setting and in the presence of bit flips. NARFPI is mainly based on RFPI with a modification derived from AOP-f [16] to make it robust against bit flips. The main feature of NARFPI is that, similarly to RFPI, it does not need *a priori* knowledge of the sparsity level of the signal.

II. SETUP

Assume a K-sparse data vector $\mathbf{x} \in \mathbb{R}^N$ and a sampling rule in which each sample is generated by computing the inner product between $\mathbf{x}$ and a measuring vector $\phi_i \in \mathbb{R}^N$, namely, the ith measurement is given by $y_i = \langle \mathbf{x}, \phi_i \rangle$, where $i = 1, \ldots, M$. The vector of measurements and the measuring matrix, whose rows are ϕ_i, are denoted by $\mathbf{y}$ and Φ respectively. Thus,

$$\mathbf{y} = \Phi \mathbf{x}. \qquad (1)$$

We focus on the 1-bit compressive sensing setting where the vector of 1-bit measurements $\mathbf{b}$ is generated by applying the sign function on $\mathbf{y}$ according to

$$\mathbf{b} = \text{sign}(\mathbf{y}) = \text{sign}(\Phi \mathbf{x}). \qquad (2)$$

---

[1]Bit flips might also occur during the measurement process [16].

The binary measurements vector $\mathbf{b}$ is transmitted over a noisy channel, which causes random bit flips. We denote the number of bit flips (a random variable) by L and the resulting binary measurement vector (after bit flips) by $\tilde{\mathbf{b}}$. Note that L is the number of negative elements in $\mathbf{b} \odot \tilde{\mathbf{b}}$, where $\odot$ denotes element-wise product.

III. Reconstruction Algorithms

In the first part of this section, we review RFPI and AOP-f. While RFPI has been designed for the noiseless case (no bit flips), AOP-f is robust against bit flips. However, AOP-f requires *a priori* information about the sparsity level K of $\mathbf{x}$. In the second part of this section, we present our contribution, namely, a reconstruction algorithm, which we refer to as NARFPI, that solves the 1-bit reconstruction problem in the presence of bit flips without requiring *a priori* knowledge of the sparsity level K of $\mathbf{x}$.

A. Review of RFPI and AOP-f

RFPI, which is designed for the noiseless case $\tilde{\mathbf{b}} = \mathbf{b}$, is based on the following reconstruction procedure [9]: find the vector $\hat{\mathbf{x}}$ with the smallest ℓ_1-norm satisfying

$$\mathbf{b} \odot \Phi\mathbf{x} \succeq \mathbf{0} \tag{3}$$

where $\succeq$ denotes element-wise inequality. Since the trivial solution to this problem is $\hat{\mathbf{x}} = \mathbf{0}$, an energy constraint over the reconstructed signal must be further imposed. The choice in [9] is to force the reconstructed signal to be on the ℓ_2-ball of unit radius, which yields the following constrained minimization problem:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1$$
$$\text{s.t. } \mathbf{b} \odot \Phi\mathbf{x} \succeq 0$$
$$\|\mathbf{x}\|_2 = 1. \tag{4}$$

To solve (4) efficiently, a barrier cost function is introduced in [9], which, together with Lagrange multiplier method, yields the following approximation of (4)

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 + \lambda \sum_i C([\mathbf{b} \odot \Phi\mathbf{x}]_i)$$
$$\text{s.t. } \|\mathbf{x}\|_2 = 1 \tag{5}$$

where

$$C(x) = \begin{cases} x^2/2, & \text{if } x < 0 \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

is the barrier cost function and $[\cdot]_i$ denotes the ith element of the vector argument. Note that when λ is sufficiently large, the solution of (5) coincides with the solution of (4). Since the function $C(\cdot)$ in (6) is convex, a simple gradient descent algorithm can be used to solve (5). The RFPI algorithm used in [9] to solve (5) is a variation of the fixed point continuation algorithm proposed in [17].

As we shall numerically illustrate in Section IV, RFPI has poor performance in the presence of bit flips. Differently from RFPI, AOP-f introduced in [16] is robust against bit flips.

Specifically, given *a priori* information on the number of bit flips L, AOP-f tries to identify the entries of $\tilde{\mathbf{b}}$ affected by bit flips. Let $\mathbf{\Omega} \in \{-1, 1\}^M$ be the vector containing the estimated positions of the bit flips ($[\mathbf{\Omega}]_l = -1$ means that a bit flip occurred in position l). AOP-f solves [16]

$$\left(\hat{\mathbf{x}}, \hat{\mathbf{\Omega}}\right) = \arg\min_{\mathbf{x}, \mathbf{\Omega}} \left\| \left(\tilde{\mathbf{b}} \odot \mathbf{\Omega} \odot \Phi\mathbf{x}\right)^- \right\|_2$$
$$\text{s.t. } \frac{1}{2} \sum_i \left(1 - [\mathbf{\Omega}]_i\right) \leq L$$
$$\|\mathbf{x}\|_0 \leq K$$
$$\|\mathbf{x}\|_2 = 1 \tag{7}$$

where K is the sparsity level of $\mathbf{x}$, L is the number of bit flips, and

$$\left[(\mathbf{x})^-\right]_i = \begin{cases} |[\mathbf{x}]_i|, & \text{if } [\mathbf{x}]_i < 0 \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

AOP-f solves (7) by iterating between the following two steps:
Step 1: given $\hat{\mathbf{\Omega}}$, find

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\| \left(\tilde{\mathbf{b}} \odot \hat{\mathbf{\Omega}} \odot \Phi\mathbf{x}\right)^- \right\|_2$$
$$\text{s.t. } \|\mathbf{x}\|_0 \leq K$$
$$\|\mathbf{x}\|_2 = 1. \tag{9}$$

Step 2: given $\hat{\mathbf{x}}$, find

$$\hat{\mathbf{\Omega}} = \arg\min_{\mathbf{\Omega}} \left\| \left(\tilde{\mathbf{b}} \odot \mathbf{\Omega} \odot \Phi\hat{\mathbf{x}}\right)^- \right\|_2$$
$$\text{s.t. } \frac{1}{2} \sum_i \left(1 - [\mathbf{\Omega}]_i\right) \leq L. \tag{10}$$

In words, first $\hat{\mathbf{x}}$ is determined by solving (9), based on the current estimation of $\hat{\mathbf{\Omega}}$. Then, $\hat{\mathbf{\Omega}}$ is updated by solving (10), based on the new estimation of $\hat{\mathbf{x}}$. As shown in [16], the optimization problem in (10) can be solved analytically and its solution is given by

$$\left[\hat{\mathbf{\Omega}}\right]_i = \begin{cases} -1, & \text{if } \left[\left(\tilde{\mathbf{b}} \odot \Phi\hat{\mathbf{x}}\right)^-\right]_i \geq \beta \\ 1, & \text{otherwise.} \end{cases} \tag{11}$$

Here, β is the Lth largest entry of the vector $\left(\tilde{\mathbf{b}} \odot \Phi\hat{\mathbf{x}}\right)^-$. Note that both the exact number of bit flips L in $\tilde{\mathbf{b}}$ and the sparsity level K of $\mathbf{x}$ need to be given to AOP-f as *a priori* information. The specific version of AOP-f we have described is referred to in [16] as AOP-ℓ_2-f. There exists also an ℓ_1 version of AOP-f, (AOP-ℓ_1-f) which minimizes $\left\| \left(\tilde{\mathbf{b}} \odot \mathbf{\Omega} \odot \Phi\mathbf{x}\right)^- \right\|_1$ instead of $\left\| \left(\tilde{\mathbf{b}} \odot \mathbf{\Omega} \odot \Phi\mathbf{x}\right)^- \right\|_2$ in (7). Determining the regime where AOP-ℓ_1-f outperforms AOP-ℓ_2-f is an open issue. In this paper, we will consider only AOP-ℓ_2-f, for simplicity.

B. Noise-Adaptive Renormalized Fixed Point Iteration (NARFPI)

Inspired by (7), we modify (4) to account for bit flips as follows:

$$\left(\hat{\mathbf{x}}, \hat{\boldsymbol{\Omega}}\right) = \arg\min_{\mathbf{x}, \boldsymbol{\Omega}} \|\mathbf{x}\|_1$$
$$\text{s.t. } \left(\tilde{\mathbf{b}} \odot \boldsymbol{\Omega}\right) \odot \Phi\mathbf{x} \succeq \mathbf{0}$$
$$\frac{1}{2}\sum_i \left(1 - [\boldsymbol{\Omega}]_i\right) \leq L$$
$$\|\mathbf{x}\|_2 = 1. \tag{12}$$

To solve (12) efficiently, we can apply the same relaxation step as in (5) and approximate (12) by

$$\left(\hat{\mathbf{x}}, \hat{\boldsymbol{\Omega}}\right) = \arg\min_{\mathbf{x}, \boldsymbol{\Omega}} \|\mathbf{x}\|_1 + \lambda \sum_i C\left(\left[\left(\tilde{\mathbf{b}} \odot \boldsymbol{\Omega}\right) \odot \Phi\mathbf{x}\right]_i\right)$$
$$\text{s.t. } \frac{1}{2}\sum_i \left(1 - [\boldsymbol{\Omega}]_i\right) \leq L$$
$$\|\mathbf{x}\|_2 = 1. \tag{13}$$

The optimization problem in (13) is still non-convex and consists of a combination of discrete and continuous variables. Similarly to [16], our approach is to use a two-step iterative algorithm. In the first step, $\hat{\boldsymbol{\Omega}}$ is fixed and the algorithm finds the optimum $\hat{\mathbf{x}}$ as follows:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 + \lambda \sum_i C\left(\left[\left(\tilde{\mathbf{b}} \odot \hat{\boldsymbol{\Omega}}\right) \odot \Phi\mathbf{x}\right]_i\right)$$
$$\text{s.t. } \|\mathbf{x}\|_2 = 1. \tag{14}$$

Note that the only difference between (5) and (14) is that $\mathbf{b}$ is replaced by $\left(\tilde{\mathbf{b}} \odot \hat{\boldsymbol{\Omega}}\right)$ in the argument of the function $C(\cdot)$. Hence, we can use RFPI to solve (14). In the second step, we use $\hat{\mathbf{x}}$ obtained from (14), to find $\hat{\boldsymbol{\Omega}}$ as follows:

$$\hat{\boldsymbol{\Omega}} = \arg\min_{\boldsymbol{\Omega}} \sum_i C\left(\left[\left(\tilde{\mathbf{b}} \odot \boldsymbol{\Omega}\right) \odot \Phi\hat{\mathbf{x}}\right]_i\right)$$
$$\text{s.t. } \frac{1}{2}\sum_i \left(1 - [\boldsymbol{\Omega}]_i\right) \leq L. \tag{15}$$

The minimization in (15) can be rewritten as (10). Therefore, the solution of (15) is (11).

The details of NARFPI are shown in Algorithm 1. Step a) of Algorithm 1 is taken one to one from RFPI in [9]. The algorithm is initialized with $\mathbf{x}_0 = \Phi^\dagger\tilde{\mathbf{b}}/\left\|\Phi^\dagger\tilde{\mathbf{b}}\right\|_2$ where $\Phi^\dagger$ denotes the pseudo-inverse of Φ; furthermore, $\hat{\boldsymbol{\Omega}}$ in (14) is set to the all-one vector in the first iteration.

IV. Numerical Results and Discussion

In this section, we study the robustness of NARFPI to bit flips and compare its performance with RFPI [9] and AOP-ℓ_2-f [16]. Throughout this section, the dimension of $\mathbf{x}$ is $N = 1000$. The position of the nonzero elements in $\mathbf{x}$ is chosen uniformly at random and the amplitude of the nonzero elements is generated according to a zero-mean Gaussian variable with unit variance. The $M \times N$ measuring matrix

---

Algorithm 1 NARFPI

1) **Inputs:** vector of 1-bit measurements $\tilde{\mathbf{b}} \in \{\pm 1\}^M$, measuring matrix Φ, number of bit flips L, number of outer iterations n_o, number of inner iterations n_i
2) **Initialization:** descent step-size δ, initial estimate $[\boldsymbol{\Omega}]_i = 1$ for all i, $\mathbf{b} = \tilde{\mathbf{b}}$ and $\mathbf{x}_0 = \frac{\Phi^\dagger\tilde{\mathbf{b}}}{\|\Phi^\dagger\tilde{\mathbf{b}}\|_2}$, initial coefficient $\lambda_1 = M$
3) **Outer iteration:** For $l = 1, \dots, n_o$
    a) **Inner iteration:** For $k = 1, \dots, n_i$
        i) **One-sided quadratic gradient:**
           $\mathbf{s} \leftarrow \left(\text{diag}\left(\mathbf{b}\right)\Phi\right)^T \left(\mathbf{b} \odot \Phi\mathbf{x}_{k-1}\right)^-$
        ii) **Gradient projection on sphere surface:**
           $\mathbf{g} \leftarrow \langle\mathbf{s}, \mathbf{x}_{k-1}\rangle \mathbf{x}_{k-1} - \mathbf{s}$
        iii) **One-sided quadratic gradient descent:**
           $\mathbf{h} \leftarrow \mathbf{x}_{k-1} - \delta\mathbf{g}$
        iv) **Shrinkage (ℓ_1 gradient descent):**
           $[\mathbf{u}]_i \leftarrow \text{sign}\left([\mathbf{h}]_i\right)\max\left\{\left|[\mathbf{h}]_i\right| - \frac{\delta}{\lambda_l}, 0\right\}$, for all i
        v) **Normalization:** $\mathbf{x}_k \leftarrow \frac{\mathbf{u}}{\|\mathbf{u}\|_2}$
    b) **Find the location of noisy bits and flip them:** Update $\boldsymbol{\Omega}$ from (11). $\mathbf{b} \leftarrow \boldsymbol{\Omega} \odot \tilde{\mathbf{b}}$.
    c) **Initialize next inner iteration:**
        $\mathbf{x}_0 \leftarrow \mathbf{x}_{n_i}$, $\lambda_{l+1} \leftarrow c\lambda_l$, where c is a fixed constant.
4) **Output:** $\hat{\mathbf{x}} = \mathbf{x}_{n_i}$

Φ has independent entries following a zero-mean Gaussian distribution with variance $1/M$. We set the probability that a bit flip occurs to $P = 3\%$ and give MP to NARFPI and AOP-ℓ_2-f as estimate of the number of bit flips L. Note that the actual number of bit flips L in general does not coincide with MP. Finally, the number of outer iterations n_o is set to 20, the number of inner iterations n_i is set to 200, and $c = 1.1$.

First, we consider the scenario in which the sparsity level $K \in [1, 19]$ is a random variable with symmetric discrete triangular distribution with mean 10 and variance $\sigma_K^2 \in [0, 25]$. We assume that the probability distribution of the sparsity level is not known to the reconstruction algorithms, which have knowledge only of the mean value 10 of the sparsity level. Consequently, we give 10 to AOP-ℓ_2-f as estimate of the sparsity level of the signal. Furthermore, we set the number of binary measurements to $M = 2000$. This setting is beyond the classical compressive sensing goal of few measurements, i.e., $M \ll N$. Note though that, since we represent each measurement by only one bit, we can afford more measurements for a given *bit budget* compared to more sophisticated quantized compressive sensing approaches. The quality of the reconstruction is measured in terms of the *received signal to noise ratio* (RSNR) defined as $\mathbb{E}\left(\|\mathbf{x}\|_2^2\right)/\mathbb{E}\left(\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2\right)$. The performance of the three algorithms is averaged over 100 realizations for each value of σ_K^2.

In Fig. 1, the RSNR (dB) of the three algorithms is shown. As it can be seen, NARFPI outperforms RFPI in the presence of bit flips. As can be seen from Fig. 1, the average RSNR of AOP-ℓ_2-f decreases as the variance of K increases. In contrast,
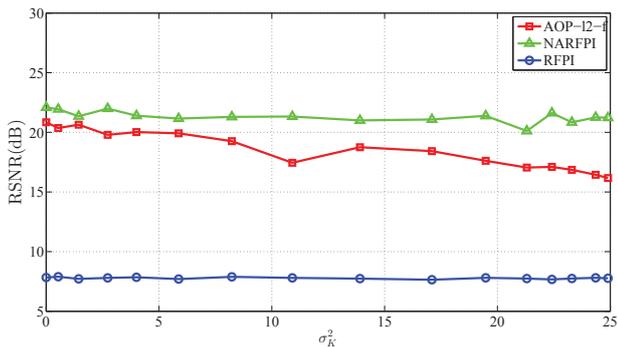
Fig. 1: The performance of NARFPI, RFPI and AOP-ℓ_2-f when K is generated according to a triangular distributions with mean 10 and different variance σ_K^2.

the average RSNR of NARFPI is constant and independent of σ_K^2. RFPI exhibits also a constant RSNR as σ_K^2 varies, but its performance is poor because of the presence of bit flips.

To investigate whether NARFPI apparent superior performance occurs also for other M values, and—at the same time—to assess in a more fundamental way the sensitivity of AOP-ℓ_2-f to mismatch between the actual sparsity level and the estimated sparsity level, we consider another scenario in which K is fixed to a value between 1 to 19 (but AOP-ℓ_2-f is still given 10 as estimate) and consider different values of M. The other parameters in this numerical experiment are the same as in the previous simulation.

In Fig. 2, we plot $1/\text{RSNR}$ (i.e., the reconstruction error) in linear scale as a function of K for both NARFPI (Fig. 2.a) and AOP-ℓ_2-f (Fig. 2.b). As expected, the reconstruction accuracy of NARFPI increases when the sparsity level K decreases. In addition, the reconstruction error of NARFPI seems to grow linearly. On the contrary, the reconstruction error of AOP-ℓ_2-f is almost constant for $K \leq 10$ but appears to grow faster than linearly in K when K exceeds 10. This comes perhaps as no surprise given that AOP-ℓ_2-f solves (7) under the constraint that $\|\mathbf{x}\|_0 \leq K$ and that we give $K = 10$ to AOP-ℓ_2-f as estimate of the signal sparsity level.

By comparing Fig. 2.a and Fig. 2.b we see that, in the regime where the number of measurements is large compared to the signal dimension (e.g., $M/N = 2$), NARFPI outperforms AOP-ℓ_2-f for all values of K. However, in the regime where the number of measurements is small compared to the signal dimension (e.g., $M/N = 0.3$) AOP-ℓ_2-f outperforms NARFPI for $K > 10$.

V. Conclusions

To summarize, Fig. 1 and Fig. 2 show that NARFPI yields a lower estimation error than RFPI and AOP-ℓ_2-f in the presence of random bit flips and when the sparsity level of the signal deviates from its estimated value. The linearity of NARFPI reconstruction error in the sparsity level K implies that when the sparsity level K randomly varies according to a symmetric distribution, the reconstruction error of NARFPI is constant,
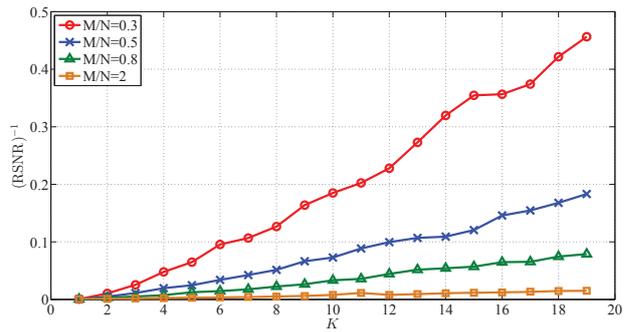
(a) NARFPI
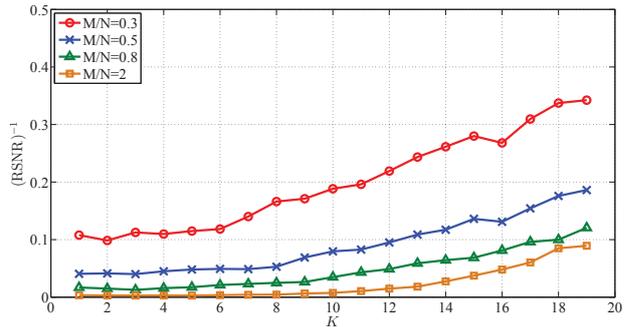
(b) AOP-ℓ_2-f

Fig. 2: The estimation error of NARFPI and AOP-ℓ_2-f as a function of the number of binary measurements M and of the sparsity level K of $\mathbf{x}$.

i.e., it does not depend on the variance of K.

Further gains both in performance and in complexity compared to AOP-ℓ_2-f might be obtained by replacing RFPI in step a) of our algorithm with the *restricted step shrinkage* method recently proposed in [6].

References

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

[2] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[3] E. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5404–5425, Dec. 2006.

[4] E. J. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

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

[6] J. Laska, Z. Wen, W. Yin, and R. 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.

[7] A. Zymnis, S. Boyd, and E. Candés, "Compressed sensing with quantized measurements," *IEEE Signal Process. Lett.*, vol. 17, no. 2, pp. 149–152, Feb. 2010.

[8] W. Dai and O. Milenkovic, "Information theoretical and algorithmic approaches to quantized compressive sensing," *IEEE Trans. Commun.*, vol. 59, no. 7, pp. 1857–1866, Jul. 2011.

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

[10] P. Boufounos, "Greedy sparse signal reconstruction from sign measurements," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Nov. 2009, pp. 1305–1309.

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

[12] L. Jacques, J. N. Laska, P. T. Boufounos, and R. G. Baraniuk, "Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors," Apr. 2011. [Online]. Available: http://arxiv.org/abs/1104.3160

[13] Y. Plan and R. Vershynin, "One-bit compressed sensing by linear programming," Oct. 2011. [Online]. Available: http://arxiv.org/abs/1109.4299

[14] ——, "Robust 1-bit compressed sensing and sparse logistic regression: A convex programming approach," Feb. 2012. [Online]. Available: http://arxiv.org/abs/1202.1212

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

[16] 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, Jul. 2012.

[17] E. T. Hale, W. Yin, and Y. Zhang, "A fixed-point continuation method for ℓ_1-regularized minimization with applications to compressed sensing," *SIAM J. Opt.*, vol. 19, no. 3, pp. 1107–1130, Oct. 2008.