

Trust, but Verify: Fast and Accurate Signal Recovery from 1-bit Compressive Measurements

Jason N. Laska, Zaiwen Wen, Wotao Yin, and Richard G. Baraniuk

Abstract—The recently emerged *compressive sensing* (CS) framework aims to acquire signals at reduced sample rates compared to the classical Shannon-Nyquist rate. To date, the CS theory has assumed primarily real-valued measurements; it has recently been demonstrated that accurate and stable signal acquisition is still possible even when each measurement is quantized to just a single bit. This property enables the design of simplified CS acquisition hardware based around a simple sign comparator rather than a more complex analog-to-digital converter; moreover, it ensures robustness to gross non-linearities applied to the measurements. In this paper we introduce a new algorithm — *restricted-step shrinkage* (RSS) — to recover sparse signals from 1-bit CS measurements. In contrast to previous algorithms for 1-bit CS, RSS has provable convergence guarantees, is about an order of magnitude faster, and achieves higher average recovery signal-to-noise ratio. RSS is similar in spirit to *trust-region* methods for non-convex optimization on the unit sphere, which are relatively unexplored in signal processing and hence of independent interest.

Index Terms—1-bit compressive sensing, quantization, consistent reconstruction, trust-region algorithms

I. INTRODUCTION

THE great leap forward in digital processing over the last few decades has created an insatiable demand for the digitization of ever wider bandwidth signals and ever higher resolution images and video. In turn, this has led to an increased burden on signal acquisition devices, forcing them to sample faster or pack more sensors onto an imaging array. The emergence of these high resolution devices has also created a new problem — we now create more data than can be stored, transmitted, or in some cases even processed [1]. This phenomenon has been termed the *data deluge*.

To deal with the data deluge, a new acquisition framework, *compressive sensing* (CS), has emerged. This framework tackles both the acquisition and data deluge problems in the following way. Suppose that a conventional Nyquist sampling system acquires N samples; however, only K coefficients are

non-zero, i.e., the signal is K -sparse. Rather than acquiring all N samples, the CS framework explains that, for a small constant $C \sim 1 - 10$, only $M = CK \log(N/K) \ll N$ measurements need be acquired to capture the salient information about the signal, and in fact recover the signal itself [2, 3]. This theory can be extended to signals that are not exactly K -sparse and signals where the K elements are coefficients in some transform basis. In practice, fewer measurements translates to slower sampling rates, fewer sensors, or shorter sensing times. It also implies that the number of values to store or transmit is closer to the minimum K values that would be needed if the signal was known *a priori*.

In theory, CS measurements are assumed to be real-valued. In practice, measurements are quantized; that is, real-values are mapped to a finite set of values represented by bits. Quantization introduces measurement error, which can be treated as noise and mitigated using a variety of new CS techniques [4–8].

Remarkably, it has been shown that signals can be recovered with high accuracy from measurements quantized to just one bit each, at the cost of slightly increasing C [9]. This fact has numerous practical implications. First, the number of bits needed to be transmitted or stored is drastically reduced. In many applications the goal is to reduce the total number of bits, not just the number of measurements. Quantization to one bit per measurement is one approach toward meeting this goal. Second, simple 1-bit hardware quantizers consist only of a comparator and can operate at high speeds. Thus, we can reduce the sampling complexity by reducing the number of bits per measurement, rather than decreasing the number of measurements. Third, because 1-bit encoding is invariant under many kinds of non-linearities, 1-bit CS techniques are robust to gross non-linearities applied to measurements [10]. These deep practical benefits justify the further study of 1-bit quantization and signal recovery in CS.

Our work in this paper builds on the 1-bit CS setup and recovery problem proposed in [9]. In this framework, quantization is achieved by only retaining the signs of measurements, and for recovery, we find the sparsest signal, such that the signs of its measurements match those that we acquired, i.e., the solution is *consistent*. Since absolute scaling information is lost, we impose a constraint that the signal has unit energy, a non-convex constraint. While algorithms have been previously proposed to deal with this constraint in this framework [9, 11], they do not have provable convergence.

In this paper, we introduce a new 1-bit CS signal recovery algorithm that we dub the *restricted-step shrinkage* (RSS) algorithm. First, we formulate an algorithm, RSS-outer, to solve the 1-bit CS problem using the augmented Lagrangian

J. N. Laska and R. G. Baraniuk are with the Department of Electrical and Computer Engineering, Rice University, Houston, TX, 30332 USA. Email: laska@rice.edu, richb@rice.edu. Web: <http://dsp.rice.edu>

Z. Wen and W. Yin are with the Department of Computational and Applied Mathematics, Rice University, Houston, TX, 30332 USA. Email: zaiwen.wen@rice.edu, wotao.yin@rice.edu.

Z. W. was supported in part by NSF DMS-0439872 through UCLA IPAM. W. Y. was supported in part by NSF CAREER Award DMS-07-48839, ONR Grant N00014-08-1-1101, the U. S. Army Research Laboratory and the U. S. Army Research Office grant W911NF-09-1-0383 and an Alfred P. Sloan Research Fellowship. J. L. and R. B. were supported by the grants NSF CCF-0431150, CCF-0728867, CCF-0926127, CNS-0435425, and CNS-0520280, DARPA/ONR N66001-08-1-2065, ONR N00014-07-1-0936, N00014-08-1-1067, N00014-08-1-1112, and N00014-08-1-1066, AFOSR FA9550-07-1-0301 and FA9550-09-1-0432, ARO MURI W911NF-07-1-0185 and W911NF-09-1-0383, and the Texas Instruments Leadership University Program.

optimization framework. Second, we employ a restricted-step subroutine, RSS-inner, to solve a non-convex subproblem. The subroutine is computationally efficient, since it adaptively chooses its step-sizes and step directions in similar fashion to trust-region techniques [12]. Specifically, RSS-inner iteratively solves for the optimal value of a smoothed approximation of the objective function within a ball of a given radius. If this results in a feasible point, then the radius is increased in the next iteration; otherwise it is decreased.

The hallmarks of the RSS algorithm include: provable convergence, orders of magnitude speedup compared to existing techniques, and improved consistency or feasibility performance compared to existing techniques. This is demonstrated with a detailed numerical comparison between the RSS algorithm and those in the literature.

The organization of this paper is as follows. Section II defines some terminology used in this paper. Section III reviews the current literature on 1-bit CS and Section IV describes the basic procedure of trust-region methods. Section V introduces the RSS algorithm for 1-bit CS and provides convergence guarantees. Section VI demonstrates the performance of our algorithm and compares it with previous methods. We conclude in Section VII with a discussion the feasibility of this problem, and possible future directions of this work.

II. TERMINOLOGY

In the remainder of this paper, we will use the following terms. A *stationary point* of an optimization problem is a point that satisfies the Karush-Kuhn-Tucker (KKT) first-order optimality conditions [13]. By *convergence* we mean that an algorithm converges to a stationary point of the objective from any starting point, but not necessarily to a global minimizer of the objective. We say a point x is a *cluster point* of sequence $\{x_s\}_{s \in \mathbb{N}}$ if for any $\epsilon > 0$ there exist an infinite number of points of $\{x_s\}$ lying in the ϵ -ball of x . Note that the sequence $\{x_s\}$ may not converge. A *feasible solution* is a solution such that all constraints are satisfied. The *subgradient* ∂f of function $f(x)$ at point x_0 is defined as any vector z such that

$$f(x) - f(x_0) \geq z(x - x_0). \quad (1)$$

III. BACKGROUND ON COMPRESSIVE SENSING

A. Compressive sensing

In the CS framework, we acquire a signal $x \in \mathbb{R}^N$ via the linear measurements

$$y = \Phi x, \quad (2)$$

where Φ is an $M \times N$ measurement matrix modeling the sampling system and $y \in \mathbb{R}^M$ is the vector of samples acquired. If x is K -sparse when represented in the sparsity basis Ψ , i.e., $x = \Psi\beta$ with $\|\beta\|_0 := |\text{supp}(\beta)| \leq K$,¹ then one can acquire only $M = O(K \log(N/K))$ measurements and still recover the signal x [2, 3]. A similar guarantee can be obtained for approximately sparse, or compressible, signals. In this paper, without the loss of generality, we fix $\Psi = I$, the identity matrix, implying that $x = \beta$.

¹ $\|\cdot\|_0$ denotes the ℓ_0 quasi-norm, which simply counts the number of nonzero entries of a vector.

Signal recovery can be performed via the optimization

$$\hat{x} \leftarrow \min_x \|x\|_1 \quad \text{s.t.} \quad \Phi x = y, \quad (3)$$

where $\|x\|_1 = \sum_i |x_i|$. Under certain conditions on Φ , this program will recover K -sparse x exactly [14]. In particular, it is sufficient for Φ to have the *restricted isometry property* (RIP)

$$(1 - \delta)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1 + \delta)\|x\|_2^2, \quad (4)$$

for all K -sparse x and $\delta > 0$. The RIP is exhibited with high probability by a large class of *random* matrices. Specifically, matrices whose entries are drawn independently from a sub-Gaussian distribution are admissible [15].

While convex optimization techniques such as (3) are a powerful method for CS signal recovery, there also exist a variety of alternative algorithms that are commonly used in practice and for which comparable performance guarantees exist. In particular, greedy algorithms such as CoSaMP are known to satisfy similar guarantees to (3), but under slightly stronger assumptions on Φ [16]. Furthermore, alternative recovery strategies based on (3), often tailored for measurement noise, have been developed and analyzed [5, 7, 9, 17–22].

B. 1-bit compressive sensing

In practice measurements are *quantized*; that is, the real valued elements of y are mapped to a finite set of values, encoded by bits. Quantization inherently induces error on the measurements and, in most practical cases, the measurements cannot be represented exactly with a finite number of bits. The quantization of CS measurements has recently motivated new analysis and in many cases inspired several new algorithms [4–8]. In this paper, we focus on a particularly coarse quantization not considered in these works, where each measurement is represented by only a single bit, i.e., the measurement’s sign.

The 1-bit CS setup is as follows. Let the 1-bit quantized measurements be denoted as

$$y_s = \text{sgn}(\Phi x), \quad (5)$$

where the function $\text{sgn}(\cdot)$ denotes the sign of the variable, element-wise, and zero values are assigned to be $+1$. Note that the scaling of both the signal and the measurements is lost with this representation.

In this setup, M is not only the number of measurements acquired but also the number of bits acquired. Thus, the ratio M/N can be considered the “bits per coefficient” of the original N -length signal. In sharp contrast to conventional CS settings, this means that in cases where hardware allows, it may actually be beneficial to acquire $M > N$ measurements.

Let the matrix Y have the elements of y_s along the diagonal and zero elsewhere. We can approximately recover sparse signals up to a scaling factor with the following optimization program [9]:

$$\hat{x} \leftarrow \min_x \|x\|_1 \quad \text{s.t.} \quad Y\Phi x \geq 0 \quad \text{and} \quad \|x\|_2 = 1. \quad (6)$$

Much like (3), the ℓ_1 objective favors sparse solutions. The first constraint enforces consistency between the 1-bit quantized measurements and the solution. Since the signal scaling

information is lost, the second constraint forces the solution to lie on the unit sphere, and ensures a nontrivial solution. As in all quantization schemes, the accuracy of the solution will be dependent on the number of bits acquired.

To implement (6), the authors of [9] propose a modified version of *fixed point continuation* (FPC) [23]. FPC is modified by using the well-known technique of projecting the gradient and the intermediate solutions onto the sphere to enforce the constraint. While this modification is effective in adapting some optimization methods to the unit norm constraint, it exhibits poor performance within FPC. In addition to optimization techniques, a greedy algorithm, known as *matching sign pursuit* (MSP), has also been proposed [11]. This algorithm has empirically been shown to find good solutions but lacks any provable guarantees. It also requires that one solve a non-convex subproblem that can lead to slow computational performance. We compare the performance of 1-bit FPC and MSP against our proposed method in Section VI.

C. Benefits of 1-bit compressive sensing

There are numerous benefits to the 1-bit CS technique. First, efficient hardware quantizers can be built to operate at high speeds, because the quantizer can be a simple comparator that merely tests if a measurement is above or below zero. Second, significantly fewer bits need be stored or transmitted for a fixed number of measurements. Thus, significant gains are made when the cost of transmission is high. This benefit can work in conjunction with the simple hardware that reduces the cost of signal acquisition. Third, it has been shown that the program (6) can be used to recover signals with gross non-linearities applied to the measurements [10]. In particular, suppose a non-linearity $f(\cdot)$ is applied to the measurements. If the $f(\cdot)$ preserves the sign of the measurements, then clearly (6) can be still be used to recover x with the same performance as using the non-linearity-free measurements. Additionally, if we assume that the non-linearity preserves the relationship

$$f(x_i) < f(x_{i+1}) \text{ if } x_i < x_{i+1},$$

then the program

$$\hat{x} \leftarrow \min_x \|x\|_1 \text{ s.t. } Y'D\Phi x \geq 0 \text{ and } \|x\|_2 = 1, \quad (7)$$

can be used to recover x with similar guarantees as (6), where D is a difference matrix with 1's along the diagonal and -1 's along the first sub-diagonal, and Y' is a diagonal matrix with $\text{sign}(\text{diff}(f(\Phi x)))$ along the diagonal, with $\text{diff}(x) = x_{i+1} - x_i$, for $i = 1, \dots, N - 1$ [10].

D. Alternative 1-bit CS frameworks

Two alternative approaches have been introduced to acquire 1-bit measurements and recover sparse signals. In [24], the authors propose a convolution-based imaging system with 1-bit measurements. Reconstruction is performed using total variation (TV) minimization and a gradient descent algorithm. In addition, the authors introduce a convex regularization parameter that simultaneously enforces both sign consistency and non-zero signal norm. Because their algorithm makes

use of convolution-based measurement systems, and we only consider randomized measurement systems in this paper, we will not compare this algorithm with ours.

In [25], the authors propose both non-adaptive and adaptive 1-bit sensing schemes. The non-adaptive scheme, which most closely relates to the framework presented here, relies on knowledge of the dynamic range of the signal, as well as an assumption about the distribution of the values of the nonzero coefficients, and thus we also will not compare this algorithm with ours.

IV. BACKGROUND ON TRUST-REGION ALGORITHMS

A. Trust-region algorithms

One approach to solving optimization problems like (6) and (7) is to adapt standard CS optimization algorithms to seek a solution on the sphere. However, since these algorithms are intended to solve convex problems and the sphere constraint is non-convex, computational performance may suffer. In particular, the choice of an appropriate step-size is elusive. Common methods for choosing adaptive step-sizes, such as Barzilai-Borwein (BB) steps, do not necessarily perform well with a unit sphere constraint, since they were designed for unconstrained convex optimization [26]. In addition, to enforce the sphere constraint, many approaches must introduce an additional step that renormalizes intermediate solutions. It is not obvious that such approaches will converge.

The methods used in this work are inspired by a particular class of restricted step-size algorithms called *trust-region* methods [12]. Given the unconstrained nonlinear programming problem

$$\min_{x \in \mathbb{R}^N} f(x), \quad (8)$$

trust-region methods compute the next trial point iteratively by finding the minimizer of the approximation $m_s(x)$ of $f(x)$ within a trust-region defined by a ball centered at the current point x^s with radius Δ^s ; that is,

$$\min_{x \in \mathbb{R}^N} m_s(x) \text{ s.t. } \|x - x^s\|_2 \leq \Delta^s. \quad (9)$$

The size of the trust-region Δ^s is increased or decreased automatically according to the performance of the model (9) during previous iterations. These methods choose step directions and lengths simultaneously, and they have been proven to be reliable for solving difficult non-convex problems [12]. Additionally, these algorithms often have provable convergence guarantees.

To motivate the use of trust-region methods in 1-bit CS, consider the following simple example program:

$$\min_{x \in \mathbb{R}^2} \|x\|_1 \text{ s.t. } x_1 \leq x_2 \text{ and } \|x\|_2 = 1. \quad (10)$$

The behavior of the method can be best explained by examining both a successful iteration and a failure iteration of the algorithm applied to (10). Examples of these cases are depicted in Figure 1. The first constraint is depicted by the shaded area. The initial point is denoted by x^s , where s is the iteration number. The algorithm will take a step in a direction specified by an approximation $m_s(x)$ (not depicted) to point w and then project the result onto the unit sphere. The light dashed sphere

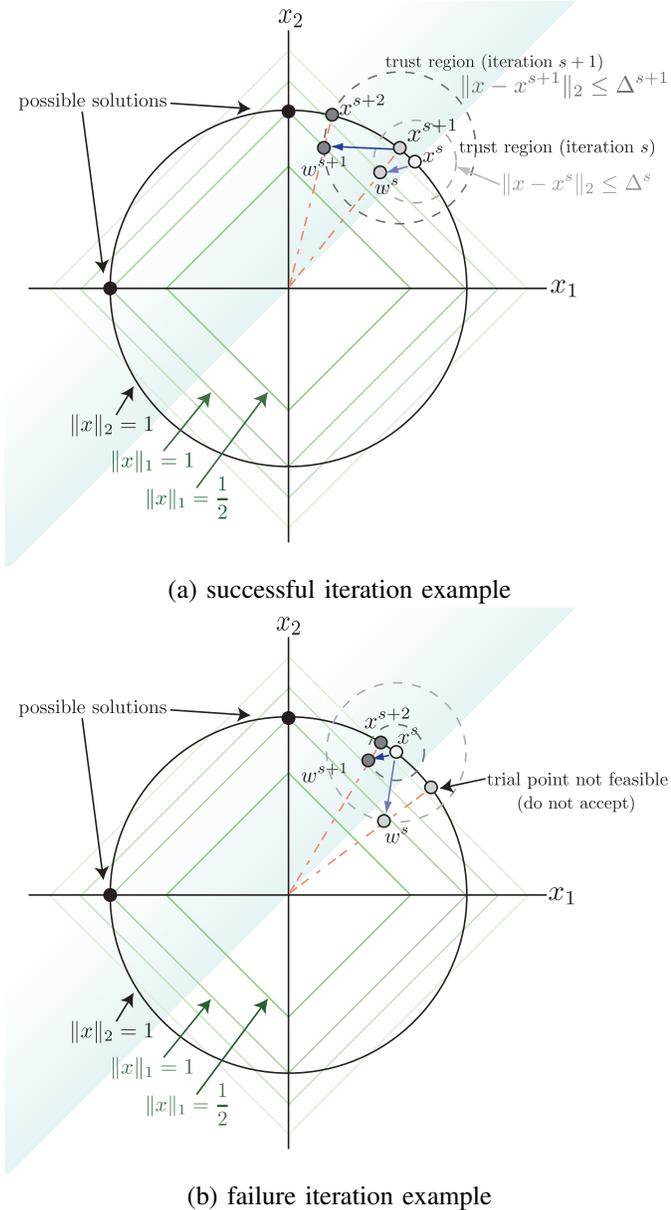


Fig. 1. Example scenarios of trust region algorithm iterations to solve (10). The goal is find an x with the minimum ℓ_1 -norm such that x has unit ℓ_2 -norm and $x_1 \leq x_2$. The shaded region denotes the feasible constraint region and the iteration number is denoted by s , with initial point x^s . The light dashed circle denotes trust region at iteration s and the dark dashed circle denotes trust region at iteration $s + 1$. w^s and w^{s+1} denote steps taken before projecting onto the unit circle. (a) During a successful iteration, the trial point falls within the feasible region and thus is accepted, denoted by x^{s+1} , and the radius of the trust region is enlarged. (b) During a failure iteration, the trust region radius is too large and the trial point falls outside the feasible region. In this case, the trust region radius is reduced and a new trial step is taken from the initial point x^s .

depicts the trust region at iteration s while the dark dashed sphere depicts the trust region at iteration $s + 1$. Depending on the success of the trial point, the trust region will expand or contract.

During a successful iteration, as depicted in Figure 1(a), the algorithm takes a step to point w^s and projects the point onto the sphere. This is depicted by the red dashed line. Since the result is within the feasible region, the point is accepted and

denoted by x^{s+1} . The trust region radius is expanded and the procedure repeats. During a failure iteration, as depicted in Figure 1(b), the trust region radius is too large and the trial point on the circle is not within the feasible region. Thus, we do not accept this trial point and take a new step from the initial point x^s , this time with a smaller trust region radius. In this example, the new step results in a feasible point.

The program (9) is generally not solvable in closed form. This includes the case studied in this paper where $f(x)$ is the ℓ_1 -norm. However, by relaxing the problem, a closed form optimal solution can often be obtained, resulting in lower cost computation at each iteration. In this paper, rather than solving (9), we iteratively solve a sequence of problems of the form

$$\min_{x \in \mathbb{R}^N} m_s(x) + \frac{\lambda^s}{2} \|x - x^s\|_2^2, \quad (11)$$

where the parameter λ^s essentially plays a role like the trust-region radius Δ^s in model (9). In fact, the solutions of (9) and (11) are the same under some properly chosen λ^s and Δ^s . We will show that our adaptation of this algorithm indeed also has guaranteed convergence, as with conventional trust region algorithms.

V. THE RESTRICTED STEP SHRINKAGE ALGORITHM FOR 1-BIT CS

In this section, we derive an algorithm for the generalized formulation of (6) and (7)

$$\min_{x \in \mathbb{R}^N} \|x\|_1 \quad \text{s.t.} \quad Ax \geq b \quad \text{and} \quad \|x\|_2 = 1. \quad (12)$$

Our strategy is as follows. First, using the augmented Lagrangian framework, we formulate an algorithm that solves (12) and denote it as RSS-outer. We choose the augmented Lagrangian framework since many state-of-the-art CS reconstruction algorithms are formulated this way [21, 27, 28]. Second, a step within RSS-outer requires that we solve a non-convex subproblem of the form

$$\min_{x \in \mathbb{R}^N} \zeta_\mu(x) = \|x\|_1 + \mu f(x) \quad \text{s.t.} \quad \|x\|_2 = 1, \quad (13)$$

where $f(x) : \mathbb{R}^N \rightarrow \mathbb{R}$ is differentiable and $\mu > 0$. We solve (13) with a trust-region-like subroutine, denoted as RSS-inner. The total procedure obtained by combining RSS-outer and RSS-inner is called the RSS algorithm.

The RSS-inner subroutine is the main contribution of this paper. Thus, we choose to describe RSS-inner in terms of the general program (13). Algorithm frameworks other than the augmented Lagrangian can be used to formulate an algorithm for (12), and in some cases may employ the RSS-inner subroutine. As an example, the quadratic penalty formulation to this problem is given in Appendix A. This formulation is simpler to implement, but does not perform as fast in practice.

A. Augmented Lagrangian formulation of (12) (RSS-outer)

We first formulate an algorithm to solve (12) using augmented Lagrangian framework. Starting from $\lambda^0 = 0$, at each iteration s we solve the Lagrangian function

$$\min_{x \in \mathbb{R}^N} \mathcal{L}(x, \lambda^s, \mu^s) \quad \text{s.t.} \quad \|x\|_2 = 1, \quad (14)$$

Algorithm 1: RSS-outer

S0 Initialize
 Given initial solution x^0
 Choose initial step-size μ^0 and $\kappa > 0$
 Set iteration $s := 0$, Lagrangian multiplier $\lambda^0 = 0$

S1 Compute next estimate (via RSS-inner)
 Set $x^{s+1} = \min \mathcal{L}(x, \lambda^s, \mu^s)$ s.t. $\|x\|_2 = 1$,
 where the objective is given by (15).

S2 Update multiplier and μ
 Set $\lambda^{s+1} := \max \{\lambda^s - \mu^s(Ax^{s+1} - b), 0\}$
 Set $\mu^{s+1} := \kappa\mu^s$

S3 Stopping rule
 If converged, then **STOP**
 otherwise, set $s := s + 1$ and go to **S1**

for x^{s+1} , where $\lambda \in \mathbb{R}^M$ and $\mu > 0$. We then set $\mu^{s+1} := \kappa\mu^s$, with $\kappa > 0$, and updates the Lagrangian multipliers λ^{s+1} according to

$$\lambda^{s+1} := \max \{\lambda^s - \mu^s(Ax^{s+1} - b), 0\}.$$

The augmented Lagrangian function for (12) is

$$\mathcal{L}(x, \lambda, \mu) := \|x\|_1 + \sum_{i=1}^m \rho((Ax - b)_i, \lambda_i, \mu), \quad (15)$$

where

$$\rho(t, \sigma, \mu) := \begin{cases} -\sigma t + \frac{1}{2}\mu t^2, & \text{if } t - \frac{\sigma}{\mu} \leq 0, \\ -\frac{1}{2\mu}\sigma^2, & \text{otherwise.} \end{cases} \quad (16)$$

Thus, the intermediate problem (14) is of the form of (13) and will be solved with RSS-inner. The complete augmented Lagrangian procedure, and how it relies on the RSS-inner subroutine is summarized in Algorithm 1.

B. Restricted-step subroutine to solve (13) (RSS-inner)

The RSS-inner subroutine finds the solution to the subproblem (13) and proceeds as follows. We begin with an initial signal estimate x^0 and an initial step-size τ^0 . At iteration s , from the point x^s , we compute a smooth approximation $m_s(x)$ to the original objective function $\zeta_\mu(x)$ in (13). The approximation is formed by adding the first-order Taylor expansion of $\mu f(x)$ and a proximal term with respect to x^s to the ℓ_1 -norm of x

$$m_s(x) = \|x\|_1 + \mu f(x^s) + \mu (g^s)^\top (x - x^s) + \frac{\tau^s}{2} \|x - x^s\|_2^2,$$

where the step size $\tau^s > 0$ and g^s is the gradient of $f(x)$. Next, we find the optimal solution to the smoothed approximation

$$z^s := \arg \min_{x \in \mathbb{R}^n} m_s(x) \text{ s.t. } \|x\|_2 = 1. \quad (17)$$

The relationship between the optimal solution z^s of the subproblem (17) and its subgradient $\partial\|z^s\|_1$, together with the norm constraint, implies that z^s can be expressed explicitly. In fact, z^s can be expressed in terms of the shrinkage (“soft threshold”) operator, defined for any $\alpha \in \mathbb{R}^N$, as

$$\mathcal{S}(\alpha, T) := \text{sign}(\alpha) \odot \max \{|\alpha| - T, 0\}, \quad (18)$$

where \odot denotes the element-wise product between two vectors and $|\cdot|$ denotes the magnitude of each element in the vector. This is demonstrated in the following Lemma.

Lemma 1. *Suppose that x^s is not a stationary point of ζ_μ .*

1) *If $\mathcal{S}^s := \mathcal{S}(\tau^s x^s - \mu g^s, 1) \neq 0$, then the closed-form solution of the subproblem (17) is*

$$z^s = \frac{\mathcal{S}^s}{\|\mathcal{S}^s\|_2}. \quad (19)$$

2) *If $|\tau^s x_i^s - \mu g_i^s| < 1$ for $i = 1, \dots, n$, then $z_i^s = 0$ for all i except that $z_i^s = \text{sgn}(\tau_i^s x_i^s - \mu g_i^s)$, where $i = \arg \max_{k=1, \dots, n} |\tau^s x_k^s - \mu g_k^s|$ (select only one i if there are multiple solutions).*

3) *Otherwise, the optimal Lagrangian multiplier λ with respect to $\|x\|_2 = 1$ satisfies $\tau^s - \lambda = 0$, $|\tau^s x^s - \mu g^s| \leq 1$, and the set $\{i \mid |\tau^s x_i^s - \mu g_i^s| = 1\}$ is not empty and the closed-form solutions of the subproblem (17) satisfy $\|z^s\|_2 = 1$ and*

$$\begin{cases} z_i^s \in (0, +\infty), & \text{if } \tau^s x_i^s - \mu g_i^s = 1, \\ z_i^s \in (-\infty, 0), & \text{if } \tau^s x_i^s - \mu g_i^s = -1, \\ z_i^s = 0, & \text{otherwise.} \end{cases} \quad (20)$$

The proof of this Lemma can be found in Appendix B. The Lemma implies that the next trial point z^s can be computed in closed form via the ratio (19).

We now present our strategy for choosing the step-size τ^s and updating the new iterate x^{s+1} from z^s . We first calculate the difference between the actual reduction of the objective function $\zeta_\mu(x)$ and predicted reduction

$$\delta(x^s, z^s) = \|x^s\|_1 - \|z^s\|_1 - \mu (g^s)^\top (z^s - x^s)$$

and then compute the ratio

$$r_s = \frac{\zeta_\mu(x^s) - \zeta_\mu(z^s)}{\delta(x^s, z^s)} \quad (21)$$

to decide whether to accept the trial point z^s as well as if the step-size should be updated. Specifically, if $r_s \geq \eta_1 > 0$, then the iteration was successful and we set $x^{s+1} = z^s$; otherwise, the iteration was not successful and we set $x^{s+1} = x^s$. Finally, the step-size τ^s is updated as

$$\tau^{s+1} \in \begin{cases} [\gamma_1 \tau^s, \gamma_2 \tau^s], & \text{if } r_s \geq \eta_2, \\ [\gamma_2 \tau^s, \tau^s], & \text{if } r_s \in [\eta_1, \eta_2), \\ [\gamma_3 \tau^s, \tau_{\max}], & \text{if } r_s \leq \eta_1, \end{cases} \quad (22)$$

where $0 < \eta_1 \leq \eta_2 < 1$ and $0 < \gamma_1 \leq \gamma_2 < 1 < \gamma_3$. The parameters $\eta_1, \eta_2, \gamma_1, \gamma_2, \gamma_3$ determine how aggressively the step-size is increased when an iteration is successful and how aggressively it is decreased when an iteration is unsuccessful. In practice, the performance of RSS-inner is not sensitive to the actual values of the parameters.

The complete RSS-inner procedure to solve subproblem (13) is summarized in Algorithm 2.

Algorithm 2: RSS-inner (subroutine)

S0 Initialize
 Given initial solution x^0 and initial step-size τ^0
 Choose $0 < \eta_1 \leq \eta_2 < 1$ and $0 < \gamma_1 \leq \gamma_2 < 1 < \gamma_3$
 Set iteration $s := 0$

S1 Compute step
 Compute a new trial point z^s via (19)
 Compute the ratio r_s via (21)

S2 Accept or reject the trial point
 If $r_s \geq \eta_1$, then set $x^{s+1} = z^s$
 otherwise, set $x^{s+1} = x^s$

S3 Adapt step-size
 Update τ^s according to (22)

S4 Stopping rule
 If converged, then **STOP**
 otherwise, set $s := s + 1$ and go to **S1**

C. Convergence

We next demonstrate that the RSS algorithm converges. Recall that by convergence we mean that the algorithm will converge to a stationary point of (13). Before proceeding, we first note that there exists $\lambda \in \mathbb{R}$ such that the first-order optimality conditions of (13) hold; that is,

$$p + \mu g(x) - \lambda x = 0, \quad \|x\|_2 = 1, \quad p \in \partial\|x\|_1, \quad (23)$$

where $g(x) = \nabla f(x)$. In addition, we make the following assumption on $g(x)$,

Assumption 1. *The gradient $g(x)$ of $f(x)$ is Lipschitz continuous with constant L :*

$$\|g(x) - g(y)\|_2 \leq L\|x - y\|_2.$$

Note that this assumption is valid for the objective function in (13).

We are now ready to establish convergence of the RSS-inner algorithm.

Theorem 1. *Suppose that for (13) $\mathcal{S}(\tau^s x^s - \mu g^s) \neq 0$ for every iteration. If the RSS-inner algorithm has finitely many successful iterations, then it converges to a stationary point. If the RSS-inner algorithm has infinitely many successful iterations, then there exists at least one cluster point of the sequence $\{x^s\}$ and every cluster point is a stationary point.*

To prove this, we first demonstrate that an iteration is successful if the step size at that iteration is sufficiently large. The remainder of the proof is by contradiction, by checking how much the objective function value of (13) decreases, for the successful iterations. The detailed proof can be found in Appendix B. The convergence of RSS-outer follows from the standard theory for non-smooth optimization [13].

In summary, in this section our goal was to solve (12). To do this, we formulated an algorithm for (12) using the augmented Lagrangian framework as given by Algorithm 1. Within the algorithm, we must solve the subproblem (14). Because (14) is of the form (13), it can be efficiently solved by the RSS-inner subroutine, as given by Algorithm 2, with

provable convergence. In the next section, we demonstrate that in practice, the algorithm outperforms previously proposed algorithms in terms of accuracy and speed.

VI. NUMERICAL SIMULATIONS

In this section, we perform numerical simulations to illustrate the performance of the RSS algorithm for 1-bit CS reconstruction as well as compare it to the performance of existing algorithms. We focus on K -sparse signals and Gaussian measurement matrices and measure performance in terms of accuracy, speed, and consistency. The purpose of these experiments is to give a reasonable comparison of the performance and behavior of the different available algorithms; however the specific values obtained should not be considered absolute. All experiments were performed in MATLAB on the same computer.

A. Experimental Setup

In each experimental trial, we generate a new length- N , K -sparse signal x with non-zero coefficient values drawn from a zero-mean Gaussian distribution. The locations of the non-zero coefficients are randomly permuted. The signal is then normalized to unit ℓ_2 norm. Note that in general the signal does not need to be normalized; normalization merely facilitates direct comparison between x and its 1-bit CS reconstruction. Indeed, in the noiseless setting, the scale of the input has no effect on performance. A new $M \times N$ measurement matrix is drawn for each trial, with entries from a zero-mean Gaussian distribution with variance $1/M$. The measurements are then computed as in (5). We obtain estimates \hat{x} using Matching Sign Pursuit (MSP) [11], 1-bit FPC [9], and the RSS algorithm introduced in this paper.

All simulations were performed for 200 trials, with $\mu = 50$ for the RSS algorithm. We found these parameters to perform well over many combinations of K , M , and N . Additionally, we tuned the parameters of both MSP and 1-bit FPC to perform well given the choice of parameters, to the best of our ability.

We consider three configurations of the parameters N , M , and K . In the first configuration we fix $N = 1000$ and $M = 500$ and we vary K between 1 and 15. In the second configuration we fix $N = 1000$ and $K = 10$ and we vary M such that M/N is between 0.05 and 2. In the third configuration, we fix $K = 10$, we vary N between 500 and 5500, and set $M = N$. In all figures, dashed lines denote MSP, dotted lines denote 1-Bit FPC, and solid triangle-marked lines denote the RSS algorithm.

B. Accuracy

We first compare the accuracy of the algorithms by computing the average mean squared error (MSE) $\|x - \hat{x}\|_2$ over the trials and expressing it in terms of the signal-to-noise ratio (SNR) in dB

$$\text{SNR}_{\text{dB}}(x, \hat{x}) = 10 \log_{10} \left(\frac{\|x\|_2^2}{\|x - \hat{x}\|_2^2} \right). \quad (24)$$

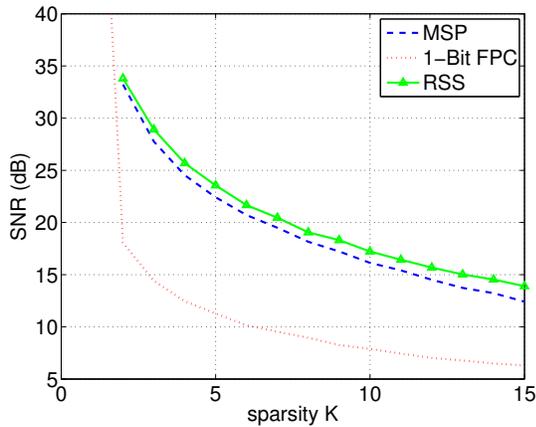
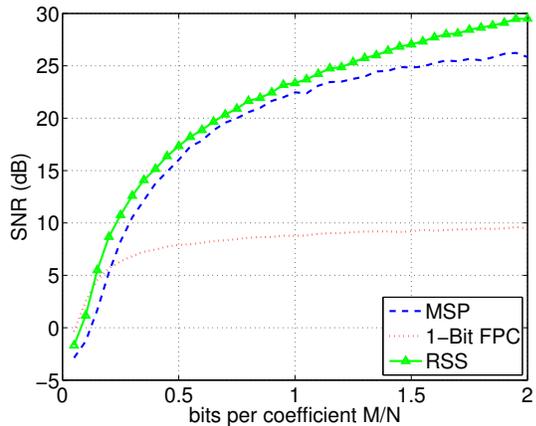
(a) Fixed M/N (b) Fixed K

Fig. 2. Accuracy comparison: Average SNR for (a) fixed $N = 1000$ and $M = 500$ with K between 1 and 15, and (b) $N = 1000$ and $K = 10$ with M/N is between 0.05 and 2. These plots demonstrate that MSP and RSS perform similarly in terms of accuracy, with 1-bit FPC performing worse than the others. Furthermore, we see that a signal can be recovered with SNR greater than 20dB using less than 0.5 bits per coefficient of x .

In most cases this is an appropriate metric for measuring reconstruction performance, since we acquire only M bits of information, the error is not likely to be zero and thus the SNR will not be infinite. One exception to this is when $K = 1$, since if the algorithm finds the correct support, then the value of the coefficient is exactly 1 due to the unit norm constraint.

The results of this experiment are depicted in Figure 2(a) for fixed M and in Figure 2(b) for fixed K . The plots demonstrate that the RSS algorithm and MSP algorithm perform similarly. The 1-bit FPC algorithm exhibits poorer performance than the others. When $K = 1$, the RSS and MSP algorithms find an exact solution, yielding an infinite SNR that is not visible on the plot.

C. Consistency

We next test whether the algorithms find a feasible solution, i.e., yield a consistent reconstruction. In this case, we are looking to see whether the signs of measurements of the estimate are the same as the signs of the original measurements. To this end, we measure the consistency, i.e., average number of incorrect signs from $\text{sign}(\Phi\hat{x})$. We say a reconstruction is

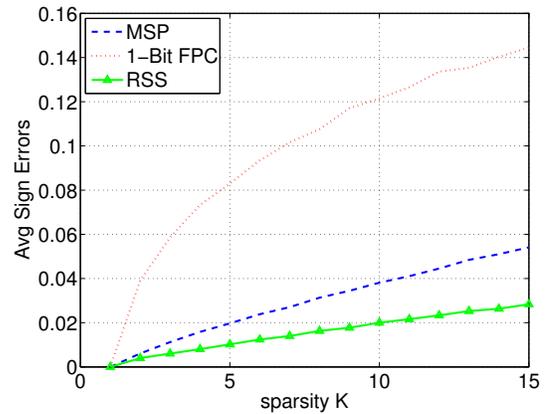
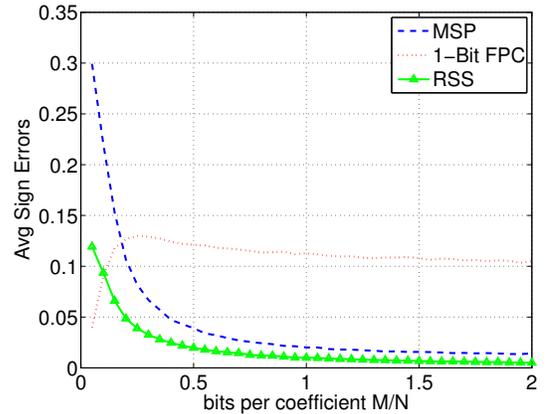
(a) Fixed M/N (b) Fixed K

Fig. 3. Average consistency comparison: Average number of incorrect signs for (a) fixed $N = 1000$ and $M = 500$ with K between 1 and 15, and (b) $N = 1000$ and $K = 10$ with M/N is between 0.05 and 2. None of the algorithms find a consistent solution except when $K = 1$. We also see that RSS obtains the best performance.

consistent if there are no incorrect signs.

Figure 3 depicts the consistency of the algorithms where Figure 3(a) is for fixed M and in Figure 3(b) is for fixed K . We see that the sign errors decrease toward zero as M increases, as expected. Similarly, the sign errors increase as K increases, with a seemingly linear trend. Note that in almost all of the cases tested, we did not, on average, achieve consistent reconstruction even when $M = 2N$. The only case where we do achieve consistency is when $K = 1$.

We consider the effect of consistency on error. For each trial, we record both the ℓ_2 -error and consistency. A summary of this experiment is depicted in Figure 4. In particular, we plot this for fixed $N = 1000$ and $K = 10$, with $M/N = 0.1$, $M/N = 0.7$, and $M/N = 1.5$. This experiment gives a glimpse into the dependence between consistency of the measurements and the the error of the solution. We see that when M/N is small (e.g., 0.1), there appears to be little clear relationship between the error and the consistency of the solution. This is particularly striking for the RSS algorithm, where the consistency can be very low but the error very high, and vice-versa. This unpredictable behavior is most likely due to the fact that too few measurements have been acquired. As M/N increases, there is a clear linear relationship between the

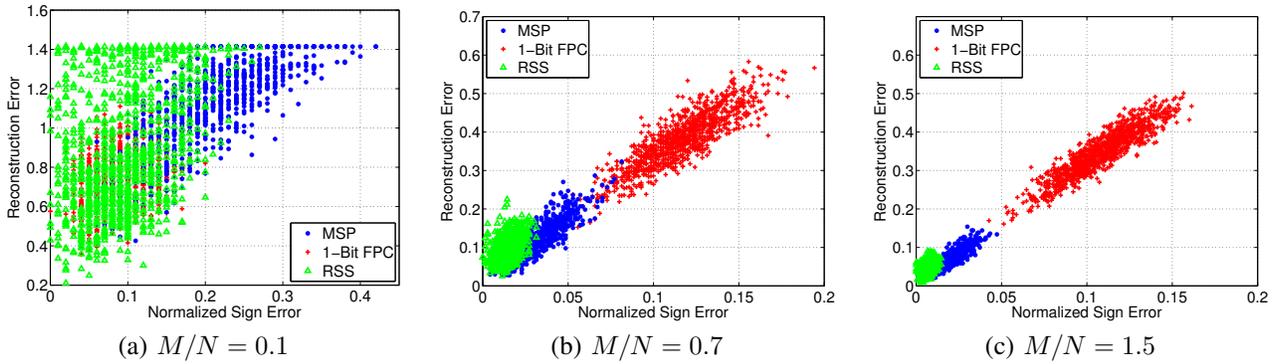


Fig. 4. Consistency-error-dependence comparison: Reconstruction error vs. normalized sign error for (a) fixed $N = 1000$, $M/N = 0.1$, and $K = 10$, (b) fixed $N = 1000$, $M/N = 0.7$, and $K = 10$, (c) fixed $N = 1000$, $M/N = 1.5$, and $K = 10$. When M/N is large enough (e.g., 0.7, 1.5), there is a linear relationship between the measurement sign errors and the signal reconstruction errors.

sign error and the reconstruction error. This means that in this regime the sign error is a good indication of the reconstruction error. It also emphasizes the importance of finding a feasible solution. The RSS algorithm concentrates closest to the origin and thus has the best performance.

D. Speed

We compare the speed of the algorithms by measuring the average time it takes each algorithm to terminate. The results of this experiment are depicted in Figure 5, where Figure 5(a) is for fixed M and Figure 5(b) is for fixed K . We find that the RSS algorithm is about an order of magnitude faster than the MSP algorithm. The main bottleneck in speed for MSP is the least-squares step, which is performed using gradient descent on the sphere. When we implemented the least-squares subroutine, we chose to maximize the SNR of the algorithm, however, speed performance can be improved at the cost of reducing SNR. The speed of the RSS algorithm will be sensitive to the choice of μ , but not significantly.

E. Missing and misidentified support

We are also interested in how well each algorithm finds the signal support, meaning the locations of the nonzero coefficients. To measure this we consider two metrics. First, we measure the number of nonzero coefficients that an algorithm “misses,” i.e., determines to be zero. Second, we measure the number of non-zero coefficients that are “misidentified,” i.e., coefficients that are determined to be nonzero when they should be zero. These two experiments are depicted in Figures 6 and 7, respectively.

Figure 6(a) depicts the number of missed coefficients as a function of K and Figure 6(b) depicts it as a function of M/N . In both cases, the RSS algorithm is significantly less likely to miss a coefficient than the MSP algorithm.

Figure 7(a) depicts the number of misidentified coefficients as a function of K and Figure 7(b) depicts it as a function of M/N . In this case, we find that the MSP algorithm outperforms all of the others. This is because the MSP algorithm returns exactly K coefficients and thus can misidentify at most K coefficients. Interestingly, the performance of the RSS algorithm does not change significantly as a function of M/N but does degrade linearly as K increases.

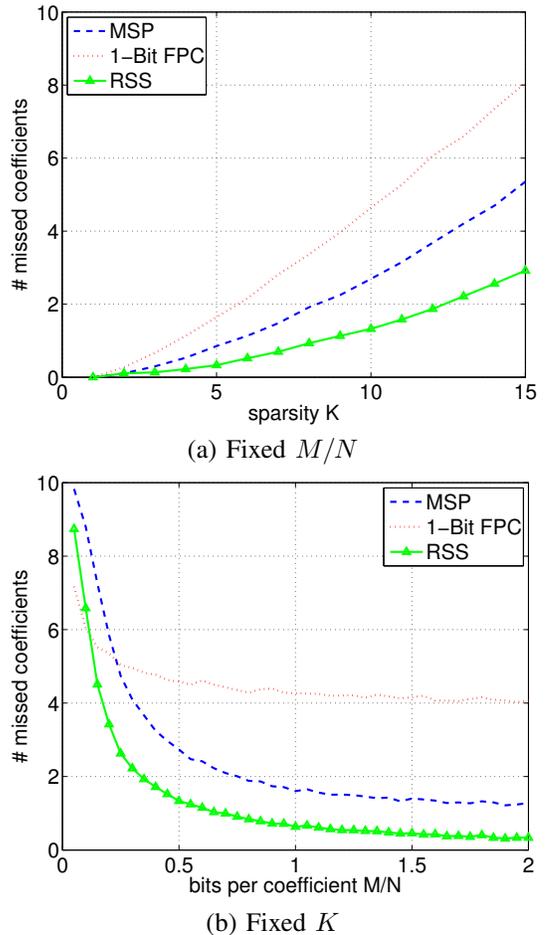


Fig. 6. Missed support comparison: Average number of coefficients not found for (a) fixed $N = 1000$ and $M = 500$ with K between 1 and 15, and (b) $N = 1000$ and $K = 10$ with M/N is between 0.05 and 2. The RSS algorithm is less likely to miss a non-zero coefficient.

F. Performance in noise

The 1-bit CS technique is quite robust to noise. To demonstrate this, we perform an experiment where in each trial, we add zero-mean Gaussian noise e to the measurements before quantization, i.e.,

$$y_s = \text{sgn}(\Phi x + e). \quad (25)$$

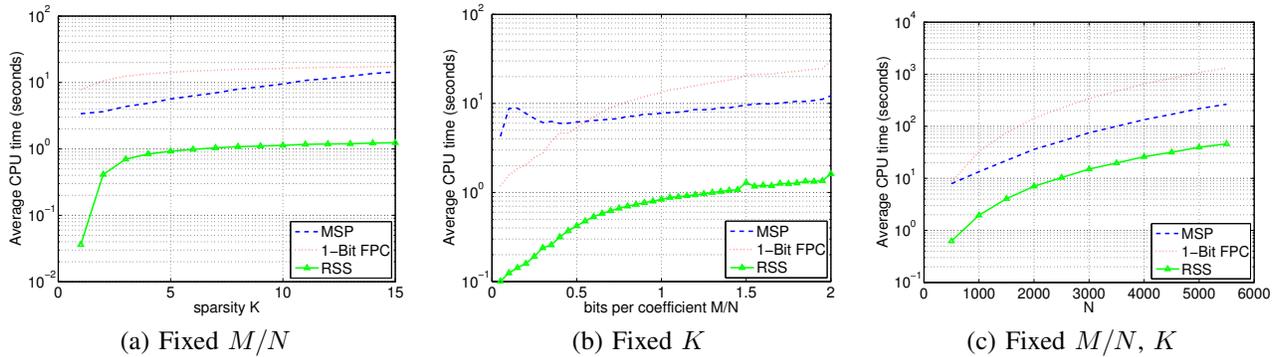


Fig. 5. Speed comparison: Average convergence time for (a) fixed $N = 1000$ and $M = 500$ with K between 1 and 15, (b) $N = 1000$ and $K = 10$ with M/N is between 0.05 and 2, and (c) $K = 10$, N is between 500 and 5500, and $M = N$. The RSS algorithm is at least an order of magnitude faster than the other algorithms.

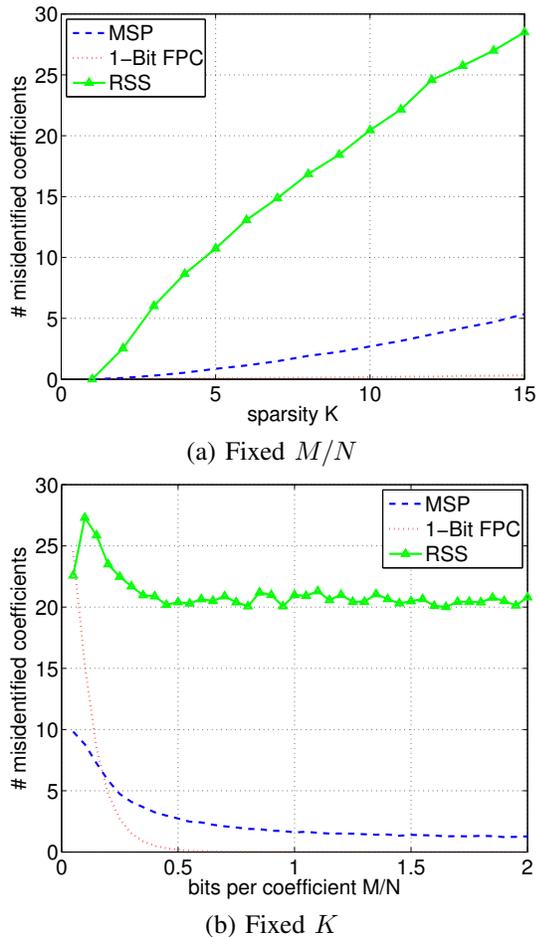


Fig. 7. Misidentified support comparison: Average number of misidentified coefficients for (a) fixed $N = 1000$ and $M = 500$ with K between 1 and 15, and (b) $N = 1000$ and $K = 10$ with M/N is between 0.05 and 2. MSP is the least likely to misidentify a coefficient, as expected.

We use the parameters $N = 1000$, $K = 10$, $M = 2N$ and scale the noise so that the measurement SNR varies between 0 dB and 40 dB. Once the measurements are quantized, we perform reconstruction. In this experiment, we only compare the RSS and MSP algorithms, since 1-bit FPC exhibited significantly poorer SNR performance. Furthermore, since MSP returns exactly K coefficients, we will also include the error on

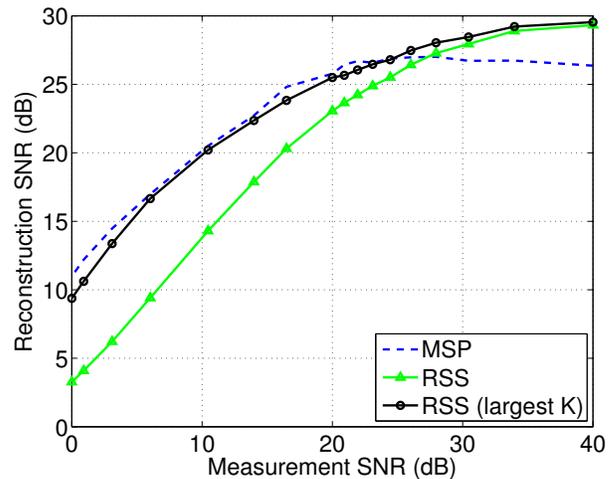


Fig. 8. Reconstruction SNR as a function of measurement SNR (before quantization). Reconstructions performed with MSP and RSS for fixed $N = 1000$, $K = 10$, $M = 2N$ and measurement SNR between 0 dB and 40 dB.

the maximum K coefficients returned by the RSS algorithm. Figure 8 depicts the results of this experiment. We find that for measurement SNR below 20 dB, the reconstruction SNR from the RSS algorithm is approximately a linear function of the measurement SNR. Above 20 dB, the performance plateaus, approaching the noiseless performance. We also find that below 25dB, the MSP algorithm performs better than the RSS algorithm. However, the error on the largest K elements returned by the RSS algorithm is comparable to the error from the MSP algorithm (also only K elements).

VII. DISCUSSION

In this paper, we have proposed a new reconstruction algorithm, the RSS algorithm, for 1-bit quantized CS measurements. Our algorithm takes a trust-region-like approach to solve a non-convex subproblem within the augmented Lagrangian framework. The RSS algorithm outperforms previously existing 1-bit CS algorithms in terms of speed, accuracy, and consistency.

Our algorithm demonstrates that signal recovery from 1-bit CS measurements can indeed be performed in a practical way,

with provable convergence guarantees. Thus, the possibilities for efficient and unique hardware designs have been extended.

When testing and comparing the 1-bit CS algorithms, one point of interest that arose is that none of them find a consistent or feasible solution with high probability, except for when $K = 1$. Thus, several possible questions could be answered in the future. Is there a formulation that offers guaranteed consistent reconstruction, that also solves the 1-bit reconstruction problem? Also, if consistent reconstruction is achieved, then is the accuracy of the solution guaranteed to increase on average? The results presented here hint that this might be the case.

APPENDIX A QUADRATIC PENALTY FRAMEWORK

The quadratic penalty framework can also make use of the RSS-inner subroutine to solve (12). This framework has been used by FPC and *gradient projection for sparse reconstruction* (GPSR) to solve conventional CS reconstruction problems [29]. This approach proceeds by iteratively minimizing a sequence of penalty functions:

$$\min_{x \in \mathbb{R}^N} \|x\|_1 + \frac{\mu^s}{2} \|\min\{Ax - b, 0\}\|_2^2 \text{ s.t. } \|x\|_2 = 1, \quad (26)$$

where $\mu^s > 0$ is the penalty parameter, and we increase $\mu^s \rightarrow +\infty$ by setting $\mu^{s+1} := \kappa \mu^s$ with $\kappa > 1$. In fact, (26) often only needs to be solved once for some values of $\mu^s = \mu$, as is done in practice with FPC and GPSR.

It is then straightforward to see that (26) is of the form (13) and can be solved by the RSS-inner subroutine.

APPENDIX B CONVERGENCE PROOF OF ALGORITHM 2

A. Lemmas

Before proving Lemma 1 from Section V-B, we introduce an additional Lemma 2 that provides bounds on the reduction of the first-order approximation $m_s(x)$. We then present a proof that demonstrates both Lemmas.

Lemma 2. *Suppose that x^s is not a stationary point of (13). Denote by $d^s := z^s - x^s$ the search direction computed at $x^s \in \mathbb{R}^N$. Then the predicted reduction satisfies $\delta(x^s, z^s) \geq \frac{\tau^s}{2} \|d^s\|_2^2$. In particular, if $\mathcal{S}(\tau^s x^s - \mu g^s) \neq 0$, the reduction of the objective function of the subproblem (17) satisfies*

$$m_s(x^s) - m_s(z^s) \geq \frac{1}{2} \|\mathcal{S}^s\|_2 \|d^s\|_2^2. \quad (27)$$

Proof of Lemmas 1 and 2: The corresponding first-order optimality conditions of (17) are

$$p + \mu g^s + \tau^s (z^s - x^s) - \lambda z^s = 0, \quad \lambda \in \mathbb{R}, \quad \|z^s\|_2 = 1, \quad (28)$$

where $p \in \partial \|z^s\|_1$. Given any feasible solution x with $\|x\|_2 =$

1, we have

$$\begin{aligned} m_s(x) - m_s(z^s) &= \|x\|_1 - \|z^s\|_1 + (\mu g^s + \tau^s (z^s - x))^\top (x - z^s) \\ &\quad + \frac{\tau^s}{2} \|x - z^s\|_2^2 \\ &= \|x\|_1 - \|z^s\|_1 + (\lambda z^s - p)^\top (x - z^s) + \frac{\tau^s}{2} \|x - z^s\|_2^2 \\ &= \|x\|_1 - p^\top x + (\lambda z^s)^\top x - \lambda + \frac{\tau^s}{2} \|x - z^s\|_2^2 \\ &= \|x\|_1 - p^\top x + \frac{\tau^s - \lambda}{2} \|x - z^s\|_2^2 \\ &\geq \frac{\tau^s - \lambda}{2} \|x - z^s\|_2^2, \end{aligned} \quad (29)$$

where the first equality follows from the Taylor expansion of the smooth terms of $m_s(x)$, the second equality comes from (28), the third equality uses $p^\top z^s = \|z^s\|_1$ and $\|z^s\|_2 = 1$, and the fact $\|x\|_1 = \max_{q \in \{-1, 1\}} q^\top x$ gives the last inequality.

It follows from (28) that $(\tau^s - \lambda)z^s := \tau^s x^s - \mu g^s - p$. We now discuss the following cases:

- 1) $\tau^s - \lambda > 0$ and $\mathcal{S}(\tau^s x^s - \mu g^s) \neq 0$. It can be verified that $\tau^s - \lambda = \|\mathcal{S}\|_2$, and $z^s = \frac{\mathcal{S}^s}{\|\mathcal{S}^s\|_2}$ is a global minimizer. Substituting $x = x^s$ into (30) gives (27).
- 2) $\tau^s - \lambda > 0$ and $|\tau^s x^s - \mu g^s| \leq 1$. Without loss of generality, we can assume that there exists a component $z_i^s > 0$. Then $p_i = 1$ and $\tau^s x^s - \mu g^s - 1 \leq 0$ which contradicts $(\tau^s - \lambda)z_i^s > 0$.
- 3) $\tau^s - \lambda < 0$. Suppose that z^s has at least two nonzero components. Without loss of generality, we can assume that there exists a component $z_i^s > 0$. Let $\bar{x}_i = z_i^s + \epsilon$ with $\epsilon > 0$ and $\bar{x}_j = z_j^s$ for all other $j \neq i$. It is obvious that $x = \frac{\bar{x}}{\|\bar{x}\|_2}$ is feasible, $x \neq z^s$ and $p \in \partial \|x\|_1$. Hence $\|x\|_1 - p^\top x = 0$ and (29) implies that $m_s(x) < m_s(z^s)$, which contradicts the fact $m_s(x) \geq m_s(z^s)$. Therefore, the solution z^s only has one nonzero element and its value must be either -1 or 1 . Note that

$$m_s(x) = \|x\|_1 + (\mu g^s - \tau^s x^s)^\top x + \text{constant}.$$

It can be verified that $z_i^s = \text{sgn}(\tau_i^s x^s - \mu g_i^s)$, where $i = \arg \max_{k=1, \dots, n} |\tau_k^s x_k^s - \mu g_k^s|$ (select only one i if there are multiple solutions); otherwise $z_i^s = 0$. In fact, the set $\{i \mid |\tau_i^s x_i^s - \mu g_i^s| = 1\}$ is empty.

- 4) $\tau^s - \lambda = 0$. Then we must have $|\tau^s x^s - \mu g^s| \leq 1$, $p \notin \partial \|x^s\|_1$, the set $\{i \mid |\tau_i^s x_i^s - \mu g_i^s| = 1\}$ is not empty and the closed-form solution of the subproblem (17) is given by (20). □

The next lemma shows that iteration s will be successful for a sufficient large τ^s , hence, the number of unsuccessful iterations between two successful iterations cannot be infinity.

Lemma 3. *Suppose that $\|d^s\|_2 > 0$ and $\tau^s \geq \bar{\tau} := \frac{2\mu L}{1-\eta_2}$. Then the s -th iteration is a very successful iteration which satisfies $\tau^{s+1} \leq \tau^s$.*

Proof: Using the definition of r_s , Lemma 2 and Assumption

1, we obtain

$$\begin{aligned}
|r_s - 1| &= \left| \frac{\zeta_\mu(x^k) - \zeta_\mu(z^s) - \delta(x^s, z^s)}{\delta(x^s, z^s)} \right| \\
&= \left| \frac{\mu f(x^s) + \mu(g^s)^\top d^s - \mu f(z^s)}{\delta(x^s, z^s)} \right| \\
&\leq \frac{2\mu \|g(x^s + \xi d^s) - g(x^s)\|_2 \|d^s\|_2}{\tau^s \|d^s\|_2^2}, (\xi \in (0, 1)) \\
&\leq \frac{2\mu L}{\tau^s} \leq 1 - \eta_2.
\end{aligned}$$

Therefore, $r_s \geq \eta_2$ and the s -th iteration is very successful. The rule (22) ensures that $\tau^{s+1} \leq \tau^s$. \square

The following lemma gives a useful alternative characterization of stationarity.

Lemma 4. *For any successful iteration k with $\tau_s < +\infty$, the point x^s is a stationary point of (13) if only if $d^s = 0$.*

Proof: Suppose that $d^s \neq 0$. Since iteration k is successful, Lemma 2 and the ratio (21) testing show that the function value at z^s is smaller than that of x^s , implying that x^s is not a stationary point. Conversely, if $d^s = 0$, then it follows from (28) and $x^s = z^s$ that

$$p + \mu g^s - \lambda x^s = 0, \quad \lambda \in \mathbb{R}, \quad \|x^s\|_2 = 1,$$

which are the first-order optimality conditions of (13). \square

We are now ready to prove Theorem 1.

B. Proof of Theorem 1

If Algorithm 2 has finitely many successful iterations, then for sufficiently large s , the iteration is unsuccessful. Thus, the sequence $\{\tau^s\}$ converges to $+\infty$. Suppose that s_0 is the index of the last successful iteration and $\|d^s\|_2 > 0$ for $s > s_0$. It follows from Lemma 3 that there must exist a very successful iteration of index s larger than s_0 , which is a contradiction to the assumption.

Suppose that Algorithm 2 has infinitely many successful iterations. Since an unsuccessful iterate in the sequence $\{x^s\}$ remains the same and makes no progress, it can be substituted by the same successful iterate. The substituted sequence which only consists of different successful iterates is still denoted by the same notation $\{x^s\}$. Since the sequence satisfying $\|x^s\| = 1$ lies in a compact set, there exists at least one cluster point x^* such that $\|x^*\| = 1$.

Suppose that the cluster point x^* is not a stationary point. According to Lemma 3, there exists a constant $\tilde{\tau}$ such that $\tau^s \leq \tilde{\tau} < +\infty$ for all s . Hence, there exists a subsequence $\{x^{s_i}\}$ approaches x^* and $\lim_{s_i \rightarrow \infty} \tau^{s_i} = t^* \geq 0$. Since x^* is not a stationary point, by Lemma 4, $d^* \neq 0$ and

$$\delta(x^*, x^* + d^*) = \|x^*\|_1 - \|x^* + d^*\|_1 - \mu(g^*)^\top d^* = \epsilon > 0.$$

Using the fact that the shrinkage operator is non-expansive, i.e.,

$$\|\mathcal{S}(x) - \mathcal{S}(y)\|_2 \leq \|x - y\|_2,$$

we obtain

$$\begin{aligned}
&\|\mathcal{S}^{s_i} - \mathcal{S}^*\|_2 \\
&= \|\mathcal{S}(\tau^{s_i} x^{s_i} - \mu g(x^{k_i})) - \mathcal{S}(\tau^* x^* - \mu g(x^*))\|_2 \\
&\leq \|\tau^{s_i} x^{s_i} - \tau^* x^* - \mu(g(x^{s_i}) - g(x^*))\|_2 \\
&\leq |\tau^*| \|x^{s_i} - x^*\|_2 + |\tau^{s_i} - \tau^*| \|x^{s_i}\|_2 + \mu M \|x^{s_i} - x^*\|_2,
\end{aligned}$$

which implies that $\lim_{s_i \rightarrow \infty} \mathcal{S}^{s_i} = \mathcal{S}^*$ and $\lim_{s_i \rightarrow \infty} d^{s_i} = d^*$. Note that $g(x)$ and $\|x\|_1$ are continuous. For s_i large enough, we have therefore that

$$\delta(x^{s_i}, x^{s_i} + d^{s_i}) = \|x^{s_i}\|_1 - \|x^{s_i} + d^{s_i}\|_1 - \mu(g^{s_i})^\top d^{s_i} \geq \frac{\epsilon}{2}.$$

It follows from the acceptance rule for successful iterations (22) that

$$\zeta_\mu(x^{s_i}) - \zeta_\mu(x^{s_i+1}) \geq \eta_1 \delta(x^s, x^{s_i} + d^{s_i}) \geq \frac{\eta_1 \epsilon}{2}. \quad (31)$$

However, since the series with positive terms

$$\begin{aligned}
\sum_{i=1}^{\infty} \zeta_\mu(x^{s_i}) - \zeta_\mu(x^{s_i+1}) &\leq \sum_{s=s_1}^{\infty} \zeta_\mu(x^{s_i}) - \zeta_\mu(x^{s_i+1}) \\
&= \zeta_\mu(x^{s_1}) - \zeta_\mu(x^*) < +\infty
\end{aligned}$$

is convergent, we have

$$\lim_{s_i \rightarrow \infty} \zeta_\mu(x^{s_i}) - \zeta_\mu(x^{s_i+1}) = 0,$$

which contradicts (31) and completes the proof. \square

REFERENCES

- [1] J. F. Gantz, C. Chute, A. Manfrediz, S. Minton, D. Reinsel, W. Schlichting, and A. Toncheva, "The diverse and exploding digital universe: An updated forecast of worldwide information growth through 2011," *IDC White Paper*, Mar. 2008.
- [2] E. Candès, "Compressive sampling," in *Proc. Int. Congress Math.*, Madrid, Spain, Aug. 2006.
- [3] D. Donoho, "Compressed sensing," *IEEE Trans. Inform. Theory*, vol. 6, no. 4, pp. 1289–1306, 2006.
- [4] J. Sun and V. Goyal, "Quantization for compressed sensing reconstruction," in *Proc. Sampling Theory and Applications (SampTA)*, Marseille, France, May 2009.
- [5] J. Laska, P. Boufounos, M. Davenport, and R. Baraniuk, "Democracy in action: Quantization, saturation, and compressive sensing," *Preprint*, 2009.
- [6] A. Zymnis, S. Boyd, and E. Candès, "Compressed sensing with quantized measurements," *IEEE Signal Processing Letters*, vol. 17, no. 2, Feb. 2010.
- [7] L. Jacques, D. Hammond, and M. Fadili, "Dequantizing compressed sensing: When oversampling and non-gaussian constraints combine," *Preprint*, 2009.
- [8] W. Dai, H. Pham, and O. Milenkovic, "Distortion-rate functions for quantized compressive sensing," *Preprint*, 2009.
- [9] P. Boufounos and R. Baraniuk, "1-bit compressive sensing," in *Proc. Conf. Inform. Science and Systems (CISS)*, Princeton, NJ, Mar. 2008.
- [10] P. Boufounos, "Reconstruction of sparse signals from distorted randomized measurements," in *Proc. Intl. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Dallas, TX, Mar. 2010.
- [11] —, "Greedy sparse signal reconstruction from sign measurements," in *Proc. Asilomar Conf. on Signals Systems and Comput.*, Asilomar, California, Nov. 2009.
- [12] A. R. Conn, N. I. M. Gould, and P. L. Toint, *Trust-region methods*, ser. MPS/SIAM Series on Optimization. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), 2000.
- [13] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed., ser. Springer Ser. in Operations Res. and Financial Eng. Springer, 2006.
- [14] E. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inform. Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.
- [15] M. Davenport, "Random observations on random observations: Sparse signal acquisition and processing," Aug. 2010.

- [16] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmon. Anal.*, vol. 26, no. 3, pp. 301–321, 2009.
- [17] E. Candès and T. Tao, "The Dantzig selector: Statistical estimation when p is much larger than n ," *Annals of Statistics*, vol. 35, no. 6, pp. 2313–2351, 2007.
- [18] P. Wojtaszczyk, "Stability and instance optimality for Gaussian measurements in compressed sensing," *Found. Comput. Math.*, vol. 10, no. 1, pp. 1–13, 2010.
- [19] E. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure and Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.
- [20] J. Laska, M. Davenport, and R. Baraniuk, "Exact signal recovery from corrupted measurements through the pursuit of justice," in *Proc. Asilomar Conf. on Signals Systems and Comput.*, Asilomar, CA, Nov. 2009.
- [21] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, "Bregman iterative algorithms for ℓ_1 minimization with applications to compressed sensing," *SIAM J. Imaging Sciences*, 2008.
- [22] Y. Wang and W. Yin, "Sparse signal reconstruction via iterative support detection," *SIAM J. Imaging Sciences*, vol. 3, no. 3, pp. 462–491, 2010.
- [23] E. T. Hale, W. Yin, and Y. Zhang, "Fixed-point continuation for l_1 -minimization: methodology and convergence," *SIAM J. Optim.*, vol. 19, no. 3, pp. 1107–1130, 2008.
- [24] A. Bourquard, F. Aguet, and M. Unser, "Optical imaging using binary sensors," *Optics Express*, vol. 18, no. 5, Mar. 2010.
- [25] A. Gupta, B. Recht, and R. Nowak, "Sample complexity for 1-bit compressed sensing and sparse classification," in *Proc. Intl. Symp. on Information Theory (ISIT)*, 2010.
- [26] J. Barzilai and J. M. Borwein, "Two-point step size gradient methods," *IMA J. Numer. Anal.*, vol. 8, no. 1, pp. 141–148, 1988.
- [27] N. S. Aybat and G. Iyengar, "A first-order augmented Lagrangian method for compressed sensing," *Preprint*, 2010.
- [28] C. Li, "An efficient algorithm for total variation regularization with applications to the single pixel camera and compressive sensing," 2009.
- [29] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. of Selected Topics in Signal Processing*, Sept. 2007.