6-6-2014

Two-Part Reconstruction with Noisy-Sudocodes

Yanting Ma
North Carolina State University at Raleigh

Dror Baron
North Carolina State University at Raleigh

Deanna Needell
Claremont McKenna College

Two-Part Reconstruction with Noisy-Sudocodes

Yanting Ma,* Dror Baron,* and Deanna Needell†

*Department of Electrical and Computer Engineering

North Carolina State University; Raleigh, NC 27695, USA

Email: {yma7, barondror}@ncsu.edu

†Department of Mathematical Sciences

Claremont McKenna College; Claremont, CA 91711, USA

Email: dneedell@cmc.edu

Abstract

We develop a two-part reconstruction framework for signal recovery in compressed sensing (CS), where a fast algorithm is applied to provide partial recovery in Part 1, and a CS algorithm is applied to complete the residual problem in Part 2. Partitioning the reconstruction process into two complementary parts provides a natural trade-off between runtime and reconstruction quality. To exploit the advantages of the two-part framework, we propose a Noisy-Sudocodes algorithm that performs two-part reconstruction of sparse signals in the presence of measurement noise. Specifically, we design a fast algorithm for Part 1 of Noisy-Sudocodes that identifies the zero coefficients of the input signal from its noisy measurements. Many existing CS algorithms could be applied to Part 2, and we investigate approximate message passing (AMP) and binary iterative hard thresholding (BIHT). For Noisy-Sudocodes with AMP in Part 2, we provide a theoretical analysis that characterizes the trade-off between runtime and reconstruction quality. In a 1-bit CS setting where a new 1-bit quantizer is constructed for Part 1 and BIHT is applied to Part 2, numerical results show that the Noisy-Sudocodes algorithm improves over BIHT in both runtime and reconstruction quality.

Index Terms

compressed sensing, two-part reconstruction, 1-bit CS.

I. Introduction

In the compressed sensing (CS) signal acquisition paradigm [2, 3], sparse signals $\mathbf{x} \in \mathbb{R}^N$ containing only $K \ll N$ nonzero coefficients can be reconstructed from $M < N$ noisy linear measurements of the form $\mathbf{y} = \Phi\mathbf{x} + \mathbf{z}$, where $\Phi \in \mathbb{R}^{M\times N}$, and $\mathbf{y}$, $\mathbf{z} \in \mathbb{R}^M$. While reconstruction quality is an important criterion for algorithm design, the runtime is also of great concern in practical applications.

YM and DB were supported in part by the National Science Foundation under Grant CCF-1217749 and in part by the U.S. Army Research Office under Grant W911NF-04-D-0003. DN was supported by the National Science Foundation Career grant #1348721, Simons Foundation Collaboration grant #274305 and the Alfred P. Sloan Research Fellowship. Portions of the work appeared at the IEEE Global Conference on Signal and Information Processing (GlobalSIP), Austin, TX, Dec. 2013 [1].

Prior art: There is a vast literature on CS signal reconstruction algorithms [2, 4]; many existing algorithms can be classified as combinatorial or geometric. The combinatorial approach uses sparse and often binary measurement matrices [5, 6], and features fast recovery but requires a suboptimal number of measurements. Sparse binary measurement matrices based on expander graphs have been shown to have good properties for CS reconstruction problems [7]. The geometric approach often uses dense measurement matrices that satisfy the Restricted Isometry Property (RIP) [3]. Linear programming [8] can be applied to perform robust reconstruction with a smaller number of measurements at the expense of greater runtime. Greedy algorithms such as CoSaMP [9] and IHT [10] offer similar reconstruction quality while requiring less runtime.

Inference based on message passing was first introduced to CS by Sarvotham et al. [11]. The more recently proposed approximate message passing (AMP) algorithm [12] applies the central limit theorem to sum-product belief propagation (BP) (or quadratic approximation to max-sum BP) followed by Taylor expansion to simplify the messages passing between nodes. AMP has received considerable attention because of its fast convergence as an iterative algorithm and the state evolution (SE) [12, 13] formalism that characterizes the reconstruction problem at each iteration. The theory underlying AMP relies on dense i.i.d. random matrices, which would make the computational complexity of matrix operations in each iteration higher than desired for large signal dimension. Nevertheless, AMP with Fourier and Hadamard matrices [14] as well as spatially-coupled Fourier and Hadamard matrices [15] have been shown numerically to approximately match the SE derived from i.i.d. random matrices while having lower computational complexity due to special structures in the Fourier and the Hadamard matrices.

The Sudocodes algorithm [16], which is related to verification codes [17, 18], provides an alternative scheme for fast reconstruction of sparse signals when there is no measurement noise. The reconstruction process is partitioned into two parts: Part 1 efficiently recovers most of the zero coefficients and some of the nonzero coefficients from the measurements acquired via a sparse measurement matrix; Part 2 applies a dense measurement matrix and an algorithm with higher computational complexity. Despite the higher complexity, the runtime is still reasonable because Part 2 solves the smaller reconstruction problem left over from Part 1. A variation of the Sudocodes algorithm is group testing basis pursuit CS (GBCS) [19], which applies a CS reconstruction algorithm, Basis Pursuit, in Part 2. Sudocodes and GBCS are both fast. However, they can only be applied to noiseless measurements, i.e., $\mathbf{y} = \Phi\mathbf{x}$, which is impractical in many real-world applications. Nonetheless, the concept of two-part reconstruction motivates a more practical framework that performs fast reconstruction in the presence of noise.

Contributions: We generalize the Sudocodes algorithm [16] to a two-part reconstruction framework that provides a trade-off between runtime and reconstruction quality. We propose a Noisy Sudocodes algorithm within the two-part framework. A zero-identification algorithm is designed for Part 1 of Noisy-Sudocodes, whereas in Part 2 we employ existing CS algorithms. We develop the speed-quality trade-off of Noisy-Sudocodes with AMP [12] in Part 2 based on a theoretical characterization of our zero-identification algorithm and the well-established asymptotic analysis of AMP. In the 1-bit CS framework [20] where 1 bit is utilized to quantize each entry of the measurements, Part 1 of Noisy-Sudocodes motivates a new 1-bit magnitude-quantizer, which differs from the conventional 1-bit sign-quantizer [20]. Numerical results for Noisy-Sudocodes with binary iterative hard thresholding (BIHT) [21] in Part 2

are promising.

Organization: The remainder of the paper is arranged as follows. We introduce the two-part framework and our proposed Noisy-Sudocodes algorithm in Section II. A theoretical analysis for Noisy-Sudocodes with AMP in Part 2 is provided in Section III. Numerical results for an application of Noisy-Sudocodes to 1-bit CS are presented in Section IV, and we conclude the paper in Section V.

II. Two-Part Reconstruction

A. The two-part framework

In our two-part framework, Part 1 applies a fast algorithm to recover part of the coefficients of the input signal. The indices of the coefficients that are not recovered in Part 1 are recorded, and Part 2 only processes those remaining coefficients. In the noiseless Sudocodes algorithm [16] where perfect reconstruction is required, the coefficients sent to Part 2 are simply the ones that cannot be perfectly recovered in Part 1. However, when there is measurement noise, a trade-off between runtime and reconstruction quality needs to be considered. On the one hand, we want Part 1 to recover as many coefficients as possible in order to reduce runtime, because Part 2 is in general more complex and slower than Part 1. On the other hand, it is overly ambitious to expect a simple and fast algorithm used in Part 1 to perform high quality reconstruction, especially in the presence of noise, and so we should not allow Part 1 to reconstruct too many coefficients.

B. Noisy-Sudocodes Algorithm

We propose a Noisy-Sudocodes algorithm within the two-part framework. Specifically, we design a fast algorithm to identify the zero coefficients of the input signal, which is suitable for Part 1. The Noisy-Sudocodes algorithm is then defined as a two-part algorithm that applies the zero-identification algorithm to Part 1, and a CS reconstruction algorithm to Part 2. Two examples of CS algorithms that we explore are AMP (Section III) and BIHT (Section IV).

Let $\mathbf{x}$ be the input signal, and denote the jth coefficient of $\mathbf{x}$ by x_j. We assume that $\mathbf{x}$ is real-valued, and that any subset of the nonzero coefficients of $\mathbf{x}$ does not sum up to zero with high probability. Similar to Sudocodes [16], the measurements of Noisy-Sudocodes are acquired via a sparse measurement matrix $\Phi_1 \in \mathbb{R}^{M_1 \times N}$ in Part 1 and a dense matrix $\Phi_2 \in \mathbb{R}^{M_2 \times N}$ in Part 2, so that in total $M = M_1 + M_2$ measurements are used. Denote the measurement noise in Parts 1 and 2 by $\mathbf{z}_1$ and $\mathbf{z}_2$, respectively. The noisy measurement systems in the two parts are given by:

$$\text{Part 1: } \mathbf{y}_1 = \Phi_1 \mathbf{x} + \mathbf{z}_1, \tag{1}$$

$$\text{Part 2: } \mathbf{y}_2 = \Phi_2 \mathbf{x} + \mathbf{z}_2. \tag{2}$$

Let $\widehat{\mathbf{x}}_1$ be the reconstructed signal in Part 1, and denote the jth entry of $\widehat{\mathbf{x}}_1$ by $\widehat{x}_{1,j}$. A set of successive integers $\{1, ..., N\}$ is denoted by $[N]$. Define Ω_i^{row} and Ω_j^{col} as the support sets (sets of indices of nonzeros) of the ith row and the jth column of Φ_1, respectively, where $i \in [M_1]$ and $j \in [N]$. Let $\epsilon \geq 0$ be a constant that depends on the

noise level.[1] Define an index set that contains the indices of small-magnitude measurements as

$$\Omega^y = \{i : |y_{1,i}| \leq \epsilon, i \in [M_1]\}. \tag{3}$$

The Noisy-Sudocodes algorithm proceeds as follows:

Part 1: The measurement vector $\mathbf{y}_1$ is acquired via (1), and thus each $y_{1,i}$ is the summation of a subset of coefficients of $\mathbf{x}$ that depends on Ω_i^{row}. If there were no measurement noise, as in the Sudocodes algorithm [16], then under our assumptions on the input $\mathbf{x}$, a zero measurement can only be the summation of zero coefficients. In other words, if $y_{1,i} = 0$, then $\mathbf{x}_{\Omega_i^{\text{row}}} = \mathbf{0}$. However, in the presence of measurement noise, a measurement is (very) unlikely to be precisely zero. Moreover, a small-magnitude measurement could have measured a combination of multiple large-magnitude coefficients, though with small probability p. Nevertheless, it is unlikely that a large-magnitude coefficient could appear in multiple small-magnitude measurements (if p is small, then p^n decreases quickly as n increases). The Noisy-Sudocodes algorithm identifies a coefficient to be zero when it is involved in c or more small-magnitude measurements, where c is a tuning parameter that governs the zero-identification criterion.[2] For those coefficients of $\mathbf{x}$ that do not satisfy the zero-identification criterion, we record their indices in a set T. That is, $\mathrm{T} = \{j : |\Omega_j^{\text{col}} \cap \Omega^y| < c, j \in [N]\}$, where $|\cdot|$ denotes cardinality. Unlike Sudocodes [16], in which some of the nonzero coefficients can be perfectly recovered in Part 1 because the measurements are noiseless, Noisy-Sudocodes leaves the reconstruction of nonzero coefficients for Part 2, where a more robust algorithm is applied.

Part 2: Solve the remaining reconstruction problem with a CS algorithm F. The distribution of the measurement matrix Φ_2 depends on the algorithm F applied to Part 2. Let $\mathbf{x}_{\mathrm{T}}$ represent the coefficients of $\mathbf{x}$ at the indices T, and $\Phi_{2,\mathrm{T}}$ represent the submatrix formed by selecting columns of Φ_2 at column indices T. The measurement vector $\mathbf{y}_2$ is acquired via (2). After receiving T from Part 1, Part 2 first generates $\Phi_{2,\mathrm{T}}$ from Φ_2. The CS algorithm F then takes $\Phi_{2,\mathrm{T}}$ and $\mathbf{y}_2$, and computes $\widehat{\mathbf{x}}_2$, the reconstructed signal of $\mathbf{x}_{\mathrm{T}}$.

We complete the reconstruction by assigning $\widehat{\mathbf{x}}_2$ to the final reconstructed signal $\widehat{\mathbf{x}}$ at indices T. The Noisy-Sudocodes algorithm is summarized in Algorithm 1.

III. Analysis of Noisy-Sudocodes with AMP in Part 2

A. Problem setting

We analyze the Noisy-Sudocodes algorithm in a specific setting. The input signal $\mathbf{x} \in \mathbb{R}^N$ is i.i.d. sparse Gaussian distributed, $x_j \sim (1-s)\delta(x_j) + s\mathcal{N}(0,1)$, where $s \in (0,1)$ is the sparsity rate, and $\delta(\cdot)$ is the delta function [22].

Part 1: The sparse measurement matrix $\Phi_1 \in \mathbb{R}^{M_1 \times N}$ has i.i.d. Bernoulli entries, $\mathbb{P}(\Phi_{1,ij} \neq 0) = \frac{d}{sN}$, where d is a tuning parameter.[3] The measurement noise $\mathbf{z}_1$ is i.i.d Gaussian distributed, $z_{1,i} \sim \mathcal{N}(0, \sigma_z^2)$.

[1] We will see how to optimize ϵ in Section III-D.

[2] We will see how to optimize c in Section III-D.

[3] We will see how to optimize d in Section III-D.

Algorithm 1 Noisy-Sudocodes

Inputs: $\mathbf{y}_1$, ϵ, c, Ω^{col} (support sets of columns of Φ_1), $\mathbf{y}_2$, Φ_2

Initialization: $\widehat{\mathbf{x}} = 0$, $\mathrm{T} = \emptyset$, $\Omega^y = \emptyset$

Part 1: Apply zero-identification criterion

for $i = 1 : M_1$ **do**

if $|y_{1,i}| < \epsilon$ **do**

$\Omega^y = \Omega^y \cup \{i\}$

for $j = 1 : N$ **do**

if $|\Omega_j^{\text{col}} \cap \Omega^y| < c$ **do**

$\mathrm{T} = \mathrm{T} \cup \{j\}$

Part 2: Apply CS reconstruction algorithm F

$\widetilde{\Phi} = \Phi_{2,\mathrm{T}}$

$\widehat{\mathbf{x}}_2 = \mathrm{F}(\mathbf{y}_2, \widetilde{\Phi})$

$\widehat{\mathbf{x}}_{\mathrm{T}} = \widehat{\mathbf{x}}_2$

Outputs: $\widehat{\mathbf{x}}$

Part 2: The approximate message passing algorithm (AMP) is applied to Part 2. The measurement matrix $\Phi_2 \in \mathbb{R}^{M_2 \times N}$ has i.i.d. Gaussian entries, $\Phi_{2,ij} \sim \mathcal{N}(0, 1/N)$. The measurement noise $\mathbf{z}_2$ follows the same distribution as $\mathbf{z}_1$.

In order to make the input signal to noise ratio (SNR) in Parts 1 and 2 identical, that is $\|\Phi_1\mathbf{x}\|_2^2/\|\mathbf{z}_1\|_2^2 = \|\Phi_2\mathbf{x}\|_2^2/\|\mathbf{z}_2\|_2^2$, the nonzero entries of the Bernoulli matrix Φ_1 are scaled by $\sqrt{\frac{s}{d}}$.

Although we only consider Gaussian noise in our analysis, we believe that Noisy-Sudocodes can be extended to more general noise distributions by applying algorithms that can handle non-Gaussian noise in Part 2. The generalized approximate message passing algorithm (GAMP) [23] is one such algorithm; we leave the extension of Noisy-Sudocodes to other noise distributions for future work.

B. Analysis of Part 1

Asymptotic independence: Because only Part 1 will be discussed in this subsection, we drop the subscripts that distinguish Parts 1 and 2. The goal of Part 1 is to identify the zero coefficients of $\mathbf{x}$. Two types of errors could occur in Part 1. The first is missed detections, which are defined as $\mathrm{MD} = \{j : x_j = 0, \hat{x}_j \neq 0, j \in [N]\}$. The second is false alarms, which are defined as $\mathrm{FA} = \{j : x_j \neq 0, \hat{x}_j = 0, j \in [N]\}$. Let $\{I_{ij}\}_{i=1,j=1}^{M,N} = \{I_{ij} : i \in [M], j \in [N]\}$ be a set of binary random variables, where $I_{ij} = 1$ if the following two conditions are satisfied: (*i*) $|y_i| < \epsilon$ given that the value of the jth coefficient is x_j and that the jth coefficient is involved in y_i; and (*ii*) $\Phi_{ij} \neq 0$, which

means that the jth coefficient is indeed involved in y_i. Denoting $\mathbb{P}(I_{ij}=1)$ by $P_{\epsilon,d}(x_j)$, we have

$$P_{\epsilon,d}(x_j) = \mathbb{P}\left(|y_i|<\epsilon, \Phi_{ij}=\sqrt{\frac{s}{d}}\middle| x_j\right)$$
$$= \sum_{n=0}^{N-1} \frac{1}{2}\left(\text{erf}\left(\frac{\epsilon-\sqrt{\frac{s}{d}}x_j}{\sqrt{2\left(\frac{ns}{d}+\sigma_z^2\right)}}\right) - \text{erf}\left(\frac{-\epsilon-\sqrt{\frac{s}{d}}x_j}{\sqrt{2\left(\frac{ns}{d}+\sigma_z^2\right)}}\right)\right)\cdot\binom{N-1}{n}\left(\frac{d}{N}\right)^n\left(1-\frac{d}{N}\right)^{N-1-n}\cdot\frac{d}{sN}, \quad (4)$$

where $\text{erf}(x)=\frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\text{d}t$ is the error function. Further, define the sum of $\{I_{ij}\}_{i=1,j=1}^{M,N}$ along i as

$$S_j = \sum_{i=1}^{M} I_{ij}, \quad j\in[N]. \quad (5)$$

We can now rewrite the zero-identification criterion as $S_j \geq c$, and define the probability of missed detection (P_{MD}) and the probability of false alarm (P_{FA}) as:

$$P_{\text{MD}} = \mathbb{P}\left(\hat{x}_j\neq 0 \mid x_j=0\right) = \mathbb{P}\left(S_j<c \mid x_j=0\right),$$
$$P_{\text{FA}} = \mathbb{P}\left(\hat{x}_j= 0 \mid x_j\neq 0\right) = \mathbb{P}\left(S_j\geq c \mid x_j\neq 0\right).$$

Note that there are subtle dependencies in $\mathbf{y}$. If a subset of nonzero coefficients of $\mathbf{x}$ is involved in multiple entries of $\mathbf{y}$, then the magnitudes of those entries of $\mathbf{y}$ are not independent. Therefore, for each j, $\{I_{ij}\}_{i=1}^M$ is not independent along i, and thus S_j is a sum of dependent Bernoulli random variables. However, the following Lemma shows that dependencies in $\mathbf{y}$ vanish under certain conditions.

Lemma 1: Let the input signal and the measurement matrix of Part 1 be defined in Section III-A, and let $P_{\epsilon,d}(x_j)$ and S_j be defined in (4) and (5), respectively. In the limit of large systems as the signal dimension N goes to infinity, for each $j\in[N]$, S_j converges to S_B in distribution, where $S_B\sim\text{Binomial}(M, P_{\epsilon,d}(x_j))$.

The proof appears in Appendix A. The main point is that the joint characteristic function of $\mathbf{y}$ can be factorized as the product of its marginal characteristic functions, which implies that entries of $\mathbf{y}$ are asymptotically independent, and thus for each j we have that $\{I_{ij}\}_{i=1}^M$ is asymptotically independent along i. Therefore, S_j converges to a sum of i.i.d. Bernoulli random variables.

Using Lemma 1, P_{MD} and P_{FA} can be calculated as follows:

$$P_{\text{MD}} = \mathbb{P}\left(\hat{x}_j\neq 0 \mid x_j=0\right)$$
$$= \sum_{m=0}^{c-1}\binom{M}{m}P_{\epsilon,d}(0)^m\left(1-P_{\epsilon,d}(0)\right)^{M-m}, \quad (6)$$
$$P_{\text{FA}} = \int_{-\infty}^{\infty} P_{\text{FA}}(a)\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}a^2}\text{d}a, \quad (7)$$
$$P_{\text{FA}}(a) = \mathbb{P}\left(\hat{x}_j= 0 \mid x_j=a\right)$$
$$= 1-\sum_{m=0}^{c-1}\binom{M}{m}P_{\epsilon,d}(a)^m\left(1-P_{\epsilon,d}(a)\right)^{M-m}.$$

We can now compute the quantities that might affect the performance of Part 2. The expected length $\widetilde{N}$ and the expected sparsity rate $\widetilde{s}$ of $\mathbf{x}_{\mathrm{T}}$ can be calculated as:

$$\begin{aligned}\widetilde{N} &= N\mathbb{P}\left(\hat{x}_{1,j} \neq 0\right) \\ &= N\left[(1-s)P_{\mathrm{MD}} + s(1-P_{\mathrm{FA}})\right],\end{aligned}$$

and

$$\begin{aligned}\widetilde{s} &= \frac{sN(1-P_{\mathrm{FA}})}{\widetilde{N}} \\ &= \frac{s(1-P_{\mathrm{FA}})}{(1-s)P_{\mathrm{MD}} + s(1-P_{\mathrm{FA}})};\end{aligned}$$

the distribution of $\mathbf{x}_{\mathrm{T}}$, which is denoted by $P_{x_j}\left(a \mid j \in \mathrm{T}\right)$, can be calculated as:

$$\begin{aligned}P_{x_j}\left(a \mid j \in \mathrm{T}\right) &= \mathbb{P}\left(x_j = a \mid \hat{x}_{1,j} \neq 0\right) \\ &= \frac{(1-s)P_{\mathrm{MD}}\delta(a)}{(1-s)P_{\mathrm{MD}} + s(1-P_{\mathrm{FA}})} + \frac{s(1-P_{\mathrm{FA}}(a))\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}a^2}}{(1-s)P_{\mathrm{MD}} + s(1-P_{\mathrm{FA}})};\end{aligned} \tag{8}$$

the distribution of $\mathbf{x}_{\mathrm{FA}}$, which is denoted by $P_{x_j}\left(a \mid j \in \mathrm{FA}\right)$ can be calculated as:

$$\begin{aligned}P_{x_j}\left(a \mid j \in \mathrm{FA}\right) &= \mathbb{P}\left(x_j = a \mid \hat{x}_{1,j} = 0, x_j \neq 0\right) \\ &= \frac{P_{\mathrm{FA}}(a)\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}a^2}}{P_{\mathrm{FA}}};\end{aligned}$$

and the expected value of the norm of $\mathbf{x}_{\mathrm{FA}}$ can be calculated as

$$\mathbb{E}[\|\mathbf{x}_{\mathrm{FA}}\|_2^2] = sN\int_{-\infty}^{\infty} a^2 P_{\mathrm{FA}}(a)\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}a^2}\mathrm{d}a. \tag{9}$$

Numerical verification: To numerically verify the asymptotic independence property, we simulate Part 1 of Noisy-Sudocodes with different input lengths N, and record the empirical probability of missed detection ($P_{\mathrm{MD}}^{\mathrm{em}}$) and the empirical probability of false alarm ($P_{\mathrm{FA}}^{\mathrm{em}}$), where we remind the reader that the corresponding theoretical predictions P_{MD} and P_{FA} are given by (6) and (7), and these predictions rely on the asymptotic independence result of Lemma 1. Define the relative error between P_{MD} and $P_{\mathrm{MD}}^{\mathrm{em}}$ as

$$\mathrm{Err(MD)} = \frac{|P_{\mathrm{MD}} - P_{\mathrm{MD}}^{\mathrm{em}}|}{P_{\mathrm{MD}}};$$

the definition of Err(FA) is similar to that of Err(MD). We plot Err(MD) and Err(FA) as functions of N in Figure 1. It is shown in Figure 1 that the error due to the independence assumption in the measurements vanishes at a rate polynomial in N.

C. Noisy-Sudocodes with AMP in Part 2

Gaussianity of noise: Recall that Part 2 only considers the residual problem left over from Part 1. That is, Part 2 only solves for $\mathbf{x}$ at the indices T. The missed detection errors in Part 1 result in the zero entries of $\mathbf{x}_{\mathrm{T}}$, whereas the false alarm errors in Part 1 result in an extra noise term for Part 2. The extra noise term is generated

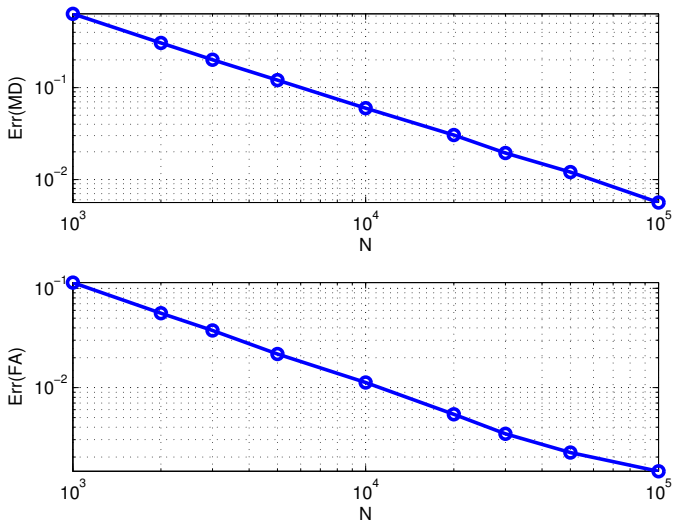

Figure 1: Top: Relative error between the empirical and theoretical probability of missed detection. Bottom: Relative error between the empirical and theoretical probability of false alarm. (The theoretical probabilities rely on the asymptotic independence result of Lemma 1.)

by $\mathbf{z}_{\text{FA}} = \Phi_{2,\text{FA}}\mathbf{x}_{\text{FA}}$, where $\Phi_{2,\text{FA}}$ represents the submatrix formed by selecting columns of Φ_2 at the indices FA. The problem for Part 2 is modeled as

$$\mathbf{y}_2 = \Phi_{2,\text{T}}\mathbf{x}_{\text{T}} + \mathbf{z}_{\text{FA}} + \mathbf{z}_2. \tag{10}$$

Because $\mathbf{z}_{\text{FA}}$ is a linear mixing of $\mathbf{x}_{\text{FA}}$, entries of $\mathbf{z}_{\text{FA}}$ are not independent. However, the following lemma shows that $\mathbf{z}_{\text{FA}}$ converges to an i.i.d. Gaussian random vector.

Lemma 2: Let $\mathbf{z}_{\text{FA}}$ be defined in (10), and $\mathbb{E}[\|\mathbf{x}_{\text{FA}}\|_2^2]$ be calculated in (9). The extra noise term $\mathbf{z}_{\text{FA}}$ converges to $\mathbf{w}$ in distribution, where $\mathbf{w} \sim \mathcal{N}(0, \sigma_{\text{FA}}^2 I)$, and $\sigma_{\text{FA}}^2 = \mathbb{E}[\|\mathbf{x}_{\text{FA}}\|_2^2]/N$.

The proof appears in Appendix B. The main point is that $\mathbf{z}_{\text{FA}}$ is a sum of i.i.d. random vectors, which converges to a multivariate Gaussian random vector in distribution. It can be shown that $\mathbf{z}_{\text{FA}}$ has uncorrelated entries. Therefore, $\mathbf{z}_{\text{FA}}$ converges to an uncorrelated Gaussian random vector.

To numerically verify the Gaussianity of $\mathbf{z}_{\text{FA}}$, we plot the sample quantiles of $\mathbf{z}_{\text{FA}}$ versus theoretical quantiles from a normal distribution (QQ plot). It is shown in the top panel of Figure 2 that the entries of $\mathbf{z}_{\text{FA}}$ lie on a straight line in the QQ plot, which implies that $\mathbf{z}_{\text{FA}}$ is marginally Gaussian. Next, we test the empirical correlation among the entries of $\mathbf{z}_{\text{FA}}$, and the resulting empirical correlation is 0.025, which is close to the empirical correlation of an i.i.d. Gaussian random vector of the same length. Therefore, it is verified that $\mathbf{z}_{\text{FA}}$ converges to an i.i.d. Gaussian random vector.

Performance analysis with AMP in Part 2: For notational simplicity, define $\widetilde{\mathbf{y}} = \mathbf{y}_2$, $\widetilde{\Phi}\widetilde{\mathbf{x}} = \Phi_{2,\text{T}}\mathbf{x}_{\text{T}}$, and $\widetilde{\mathbf{z}} = \mathbf{z}_{\text{FA}} + \mathbf{z}_2$. Problem (10) can now be rewritten as

$$\widetilde{\mathbf{y}} = \widetilde{\Phi}\widetilde{\mathbf{x}} + \widetilde{\mathbf{z}}, \tag{11}$$

where $\widetilde{\Phi} \in \mathbb{R}^{M_2 \times \widetilde{N}}$ has i.i.d. Gaussian entries, $\widetilde{\Phi}_{ij} \sim \mathcal{N}(0, 1/N)$, $\widetilde{\mathbf{x}} \in \mathbb{R}^{\widetilde{N}}$ is i.i.d. with $\widetilde{x}_j \sim P_{x_j}\left(x \mid j \in \mathrm{T}\right)$ (8), and $\widetilde{\mathbf{z}}$ is asymptotically i.i.d. Gaussian with zero mean and its variance satisfies $\sigma_{\widetilde{z}}^2 = \mathbb{E}[\|\mathbf{x}_{\mathrm{FA}}\|_2^2]/N + \sigma_z^2$, with σ_z^2 being the variance of $\mathbf{z}_2$.

Because $\widetilde{\mathbf{z}}$ can be approximated as i.i.d. Gaussian noise, we can apply the approximate message passing (AMP) algorithm [12] to approximate the minimum mean square error (MMSE) estimate of (11). AMP in Part 2 of Noisy-Sudocodes proceeds as follows:

$$\mathbf{x}^{t+1} = \eta_t\left(\frac{N}{M_2}\widetilde{\Phi}^T\mathbf{r}^t + \mathbf{x}^t\right), \tag{12}$$

$$\mathbf{r}^t = \mathbf{y} - \widetilde{\Phi}\mathbf{x}^t + \frac{\mathbf{r}^{t-1}}{\widetilde{R}}\left\langle \eta'_{t-1}\left(\frac{N}{M_2}\widetilde{\Phi}^T\mathbf{r}^{t-1} + \mathbf{x}^{t-1}\right)\right\rangle, \tag{13}$$

where $\widetilde{R} = M_2/\widetilde{N}$ is the measurement rate in problem (11), t represents the iteration index, and for a vector $\mathbf{u} \in \mathbb{R}^N$, $\langle \mathbf{u} \rangle = \frac{1}{N}\sum_{i=1}^N u_i$. Let $\mathbf{v}^t = \frac{N}{M_2}\widetilde{\Phi}^T\mathbf{r}^t + \mathbf{x}^t$. The scalar estimation function has the form $\eta_t(\mathbf{v}^t) = (\eta_t(v_1^t), ..., \eta_t(v_N^t))^T$ as in [12]. That is, η_t estimates $\widetilde{x}_j$ from its noisy observation v_j^t for each $j \in [\widetilde{N}]$. The derivative of $\eta_t(\mathbf{v}^t)$ is denoted by $\eta'_t(\mathbf{v}^t)$, and $\eta'_t(\mathbf{v}^t) = (\eta'_t(v_1^t), ..., \eta'_t(v_N^t))^T$. Due to different measurement matrix normalization schemes, a scaling factor of N/M_2 is applied to the AMP updating equations (12) and (13). It has been rigorously proved [13] that in each iteration, the input of the estimation function η_t is equivalent to the noisy observation of $\widetilde{\mathbf{x}}$ from an additive white Gaussian noise (AWGN) channel. That is, $\mathbf{v}^t = \widetilde{\mathbf{x}} + \sigma_t\mathbf{w}$, where $\mathbf{w} \sim \mathcal{N}(0, I)$. The noise variance σ_t^2 evolves following the scalar state evolution (SE) formalism [12, 13]:

$$\sigma_{t+1}^2 = \frac{N}{M_2}\sigma_{\widetilde{z}}^2 + \frac{1}{\widetilde{R}}\mathbb{E}\left[\left(\eta_t(X + \sigma_t W) - X\right)^2\right],$$

where $X \sim P_{x_j}\left(x \mid j \in \mathrm{T}\right)$ and $W \sim \mathcal{N}(0, 1)$. An unbiased estimator of σ_t^2 [24] can be applied to avoid the calculation of the expected estimation error in each iteration:

$$\widehat{\sigma}_t^2 = \frac{N}{M_2}\frac{\|\mathbf{r}^t\|_2^2}{M_2}.$$

In order to approximate the MMSE estimate, define the scalar estimation function η_t in AMP as the conditional expectation:

$$\eta_t(v_j^t) = \mathbb{E}[\widetilde{x}_j | v_j^t], \tag{14}$$

where the prior of $\widetilde{x}_j$ is $P_{x_j}(a | j \in \mathrm{T})$, and the likelihood $P(v_j^t | \widetilde{x}_j) = \mathcal{N}(\widetilde{x}_j, \sigma_t^2)$. Note that when the true distribution of $\mathbf{x}_{\mathrm{T}}$ (8) is applied to (14), the AMP algorithm with i.i.d. random measurement matrix yields the Bayesian optimal reconstruction for (11) in the limit of large systems (i.e., $M_2, \widetilde{N} \to \infty$ for constant $\widetilde{R}$) for a large region of parameters (signal sparsity, measurement rate, and measurement noise) [25, 26].

We notice that $\widetilde{\mathbf{x}}$ no longer follows a sparse Gaussian distribution due to the false alarm errors in Part 1. A comparison between the distribution of the nonzero coefficients of $\widetilde{\mathbf{x}}$ and a standard normal distribution is shown in the bottom panel of Figure 2. Significant discrepancies appear in bins centered around $x = 0$, because most false alarm errors occur when the coefficients have small magnitudes. Notice that the entire $\widetilde{\mathbf{x}}$ is a sparse signal, which has a probability mass at $x = 0$. We might think of $\widetilde{\mathbf{x}}$ as a sparse Gaussian signal whose small-magnitude

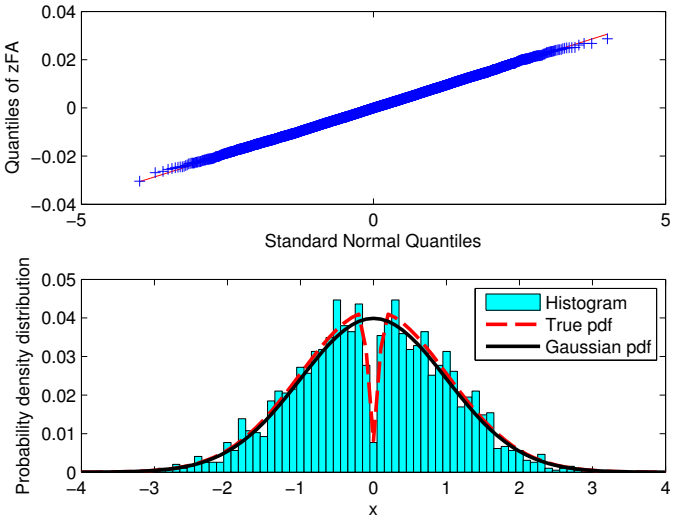

Figure 2: Top: QQ plot of the extra noise term $\mathbf{z}_{\text{FA}}$ due to false alarm errors in Part 1. Bottom: Comparison between the probability density function (pdf) of the nonzero coefficients of the input signal $\widetilde{\mathbf{x}}$ in Part 2 and the pdf of a standard normal distribution.

coefficients are approximated as 0, which results in a loss of density around $x = 0$ and an increase in the probability mass at $x = 0$. It would be interesting to see how large the performance gap would be if we approximate the prior of $\widetilde{\mathbf{x}}$ by a sparse Gaussian distribution when calculating the conditional expectation (14), because a sparse Gaussian distribution can simplify both the computation and the analysis.

Figure 3 compares the signal to distortion ratio (SDR), which is defined as

$$\text{SDR} = 10\log_{10}(\mathbb{E}[\|\mathbf{x}\|_2^2/\|\mathbf{x} - \widehat{\mathbf{x}}\|_2^2]), \tag{15}$$

achieved by the theoretical prediction and the numerical results for Noisy-Sudocodes with AMP in Part 2. The prediction for Part 1 follows the analysis in Section III-B, and the MMSE for Part 2 (11) applies the replica method for a sparse Gaussian input [27, 28]. The empirical results contain: (*i*) zero-identification in Part 1 followed by AMP with the sparse Gaussian prior for $\widetilde{\mathbf{x}}$ in Part 2; (*ii*) zero-identification in Part 1 followed by AMP with the true distribution of $\widetilde{\mathbf{x}}$ in Part 2. Figure 3 verifies that it is reasonable to approximate $P_{x_j}(a|j \in \text{T})$ (8) by a sparse Gaussian distribution; any deterioration in reconstruction quality seems minor.

D. Trade-off between runtime and reconstruction quality

The analysis of the Noisy-Sudocodes algorithm allows us to exploit the advantages provided by its two-part nature. We notice that 4 parameters in the algorithm can be tuned to provide different performances in runtime and reconstruction quality: (*i*) the parameter d that governs the sparsity of Φ_1; (*ii*) the threshold ϵ for defining small-magnitude measurements; (*iii*) the parameter c that governs the zero-identification criterion; and (*iv*) the ratio r of the number of measurements assigned to Part 1 and Part 2.

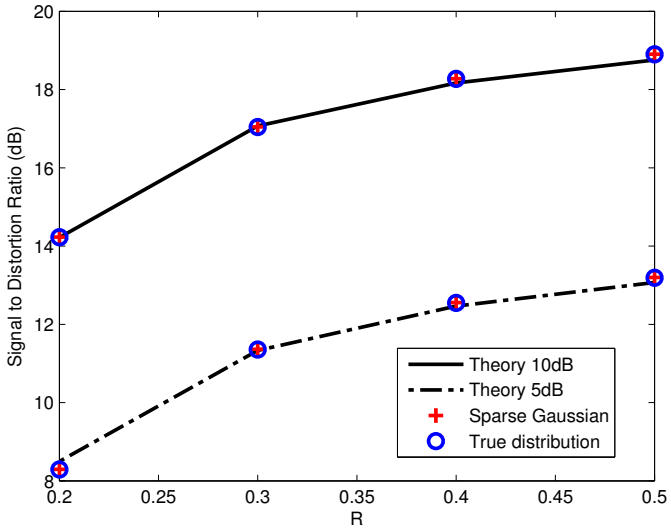

Figure 3: Numerical verification of sparse Gaussian approximation to the prior of $\mathbf{x}_{\mathrm{T}}$ (8). ($N = 20,000$, $s = 0.01$, and input SNR $= 5$ or 10 dB).

It is worth mentioning that the number of AMP iterations could also be tuned. Because AMP is merely one possible example for the algorithm F that can be applied to Part 2, we leave out this tuning parameter in our analysis and fix the number of iterations to be 20, within which AMP generally converges for the numerical settings considered in this paper.

Our goal is to find the parameters (d, ϵ, c, r) that optimize the trade-off between runtime and reconstruction quality for a given measurement rate. Both runtime and reconstruction quality are functions of (d, ϵ, c, r). We have seen how to evaluate the reconstruction quality in terms of SDR (15) in Sections III-B and III-C, and let us now model the runtime. Based on the operations performed in the Noisy-Sudocodes algorithm, we model the runtime of Part 1 by

$$t_1 = \alpha_1 N + \alpha_2 M_1 + \alpha_3 N M_1,$$

for some $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3)$. The runtime for Part 2 is modeled as

$$t_2 = \beta_1 \widetilde{N} + \beta_2 M_2 + \beta_3 \widetilde{N} M_2,$$

for some $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3)$.

We simulate Part 1 with several different values for N and M_1, and $\boldsymbol{\alpha}$ is acquired via data fitting with a least square criterion. We obtain $\boldsymbol{\beta}$ in a similar way.

The SDR (15) of Noisy-Sudocodes is evaluated with different parameter values of (d, ϵ, c, r) at measurement rates $R = M/N \in [0.2, 0.9]$. Each set of parameters results in a different $(M_1, M_2, \widetilde{N})$, and thus different (t_1, t_2). The total runtime of Noisy-Sudocodes, $t = t_1 + t_2$, is quantized to 30 quantization bins for each R, the optimal SDR corresponding to each quantization bin is the highest SDR achieved within that bin, and the parameters that

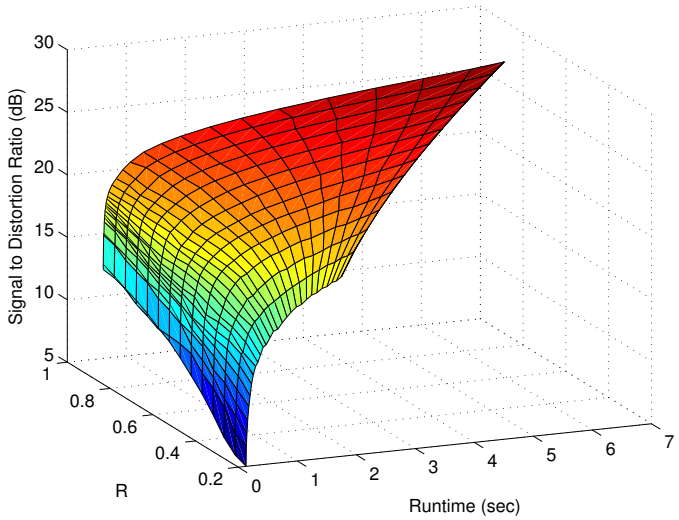

Figure 4: Trade-offs between reconstruction quality, measurement rate R, and runtime of Noisy-Sudocodes with AMP in Part 2. ($N = 20,000$, $s = 0.01$, and input SNR $= 10$ dB).

lead to the highest SDR are the optimal parameters.

A plot of SDR as a function of runtime and measurement rate is shown in Figure 4. To achieve low runtime, Part 1 needs to be aggressive in identifying zeros, which results in poor reconstruction quality. In the low runtime region, we see a significant improvement in SDR with a small increase in runtime. If we further increase the available runtime, then the high quality algorithm AMP in Part 2 eventually dominates, and thus high SDR is achieved.

To numerically verify the correctness of our predictions of SDR and runtime, we sample some points from Figure 4 and set up simulations that utilize the corresponding sets of parameters (d, ϵ, c, r). Figure 5 shows that our predictions match the simulation results in both SDR and runtime.

IV. Application to 1-bit compressed sensing

A. Noisy-Sudocodes in 1-bit compressed sensing

In the previous sections, we discussed Noisy-Sudocodes in CS settings where the measurements are allowed to have infinite quantization resolution. We notice that the fast zero-identification algorithm in Part 1 of Noisy-Sudocodes does not benefit from the high resolution measurements, because we only need to know if the entries of $\mathbf{y}_1$ are greater or less than ϵ. In other words, the measurements are implicitly quantized to a lower resolution when running Part 1. On the one hand, we see that the fast Part 1 leads to some compromises in reconstruction quality in settings where the measurements are unquantized. On the other hand, Part 1 is not penalized by the loss of quantization resolution in the measurements. This observation naturally leads us to apply Noisy-Sudocodes to a recently proposed 1-bit CS framework [20].

In 1-bit CS [20, 21, 29], the measurements are quantized to 1 bit per measurement. The problem model for

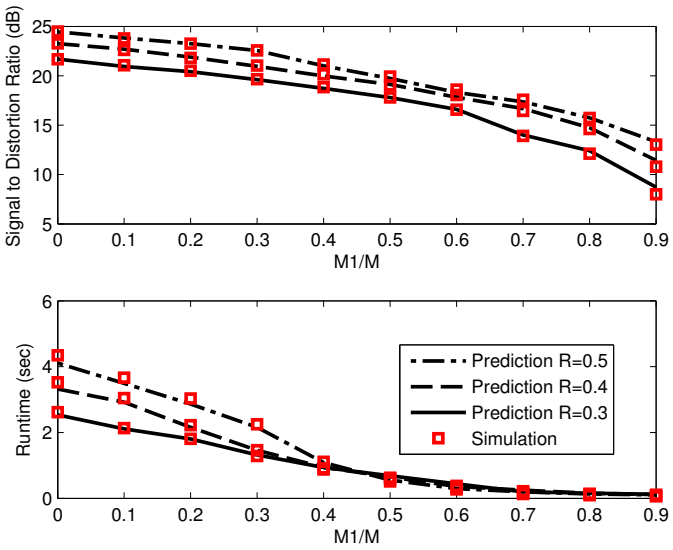

Figure 5: Top: Numerical verification of the prediction for SDR (15) of Noisy-Sudocodes with AMP in Part 2. Bottom: Numerical verification of the prediction for runtime of Noisy-Sudocodes with AMP in Part 2. ($N = 20,000$, $s = 0.01$, and input SNR $= 10$ dB).

noiseless and noisy 1-bit CS can be formulated as

$$\text{noiseless 1-bit CS:} \quad \mathbf{y} = \text{sign}(\Phi \mathbf{x}), \tag{16}$$

$$\text{noisy 1-bit CS:} \quad \mathbf{y} = \text{sign}(\Phi \mathbf{x} + \mathbf{z}), \tag{17}$$

where $\mathbf{z}$ is the measurement noise before quantization (pre-quantization noise), and

$$\text{sign}(x) = \begin{cases} -1, & \text{if } x \leq 0 \\ +1, & \text{if } x > 0 \end{cases}.$$

It is interesting to notice that Part 1 of Noisy-Sudocodes motivates a new 1-bit quantizer that performs magnitude quantization. In particular, we define our proposed 1-bit quantizer as:

$$y_i = \begin{cases} -1, & \text{if } |(\Phi \mathbf{x})_i + z_i| \leq \epsilon \\ +1, & \text{if } |(\Phi \mathbf{x})_i + z_i| > \epsilon \end{cases}. \tag{18}$$

Note that the threshold $\epsilon = 0$ when $\mathbf{z} = \mathbf{0}$. If we redefine the index set Ω^y (3) as

$$\Omega^y = \{i \mid |y_{1,i}| = -1, i \in [M_1]\}, \tag{19}$$

then Algorithm 1 can be used to solve 1-bit CS reconstruction problems with Ω^y defined in (19) and a 1-bit CS algorithm in Part 2.

A possible 1-bit CS algorithm that can be utilized is binary iterative hard thresholding (BIHT) [21]. BIHT often achieves better reconstruction performance than the previous 1-bit CS algorithms in the noiseless 1-bit CS setting.

We show by numerical results in Section IV-B that Noisy-Sudocodes with BIHT in Part 2 (Sudo-BIHT) achieves better reconstruction quality than directly applying BIHT. Moreover, Sudo-BIHT is substantially faster than BIHT.

B. Numerical results

We present simulation results that compare Sudo-BIHT and BIHT in terms of SDR (15) and runtime in both noiseless and noisy 1-bit CS settings. Runtime is measured in seconds on a Dell OPTIPLEX 9010 running an Intel(R) Core$^{\text{TM}}$ i7-3770 with 16GB RAM, and the environment is MATLAB R2012a.

The input signal $\mathbf{x}$ follows a sparse Gaussian distribution with sparsity rate $s = 0.005$. Because the amplitude information of the measurements is lost due to 1-bit quantization, it is usually assumed in the 1-bit CS framework that $\|\mathbf{x}\|_2^2 = 1$. Let M_1 and M_2 be the number of measurements for Parts 1 and 2 of Sudo-BIHT. Therefore, $M = M_1 + M_2$ is the number of measurements for BIHT. The measurement rate $R = M/N$ is set to be within the range $(0, 2)$, which is the same range utilized in the paper where BIHT is proposed [21]. Note that in 1-bit CS, we are interested in the number of quantization bits rather than the number of measurements. Therefore, the measurement rate is allowed to be greater than 1. In our simulation, we choose M_1 such that more than 90 percent of the zero coefficients can be identified in Part 1. The measurement matrix $\Phi_1 \in \mathbb{R}^{M_1 \times N}$ is i.i.d. Bernoulli distributed with $\mathbb{P}(\Phi_{1,ij} \neq 0) = \frac{d}{sN}$, where the parameter d is determined numerically. Note that the nonzero entries of the Bernoulli matrix are scaled by $\sqrt{\frac{sN}{d}}$ in order to have the same input SNR as in BIHT. The matrix $\Phi_2 \in \mathbb{R}^{M_2 \times N}$ has i.i.d. Gaussian entries, $\Phi_{2,ij} \sim \mathcal{N}(0, 1)$.

For BIHT, the measurement matrix $\Phi \in \mathbb{R}^{M \times N}$ has i.i.d. Gaussian entries, $\Phi_{ij} \sim \mathcal{N}(0, 1)$.

Finally, the pre-quantization noise $\mathbf{z}$, which we use in the noisy setting, is i.i.d. Gaussian distributed with zero mean and its variance is $10^{-2.5}$.

Noiseless setting: BIHT-ℓ_1 [21], in which the ℓ_1-norm is utilized in the objective function of the optimization problem solved by BIHT, is applied to the noiseless setting. The measurement vector $\mathbf{y}_1$ for Part 1 of Sudo-BIHT is acquired via (18) with $\mathbf{z} = \mathbf{0}$ and $\epsilon = 0$, and the measurement vectors $\mathbf{y}_2$ for Part 2 of Sudo-BIHT and $\mathbf{y}$ for BIHT are acquired via (16). In the noiseless setting, if any entry $y_{1,i}$ only measures zero coefficients, then $y_{1,i}$ will be strictly zero. Therefore, we set $c = 1$ in the zero-identification criterion. Note that Part 1 does not introduce any error in the noiseless setting. We iterate over BIHT until the consistency property[4] is satisfied or the number of iterations reaches 100.

In the top panel of Figure 6, we plot SDR as a function of the measurement rate R. The plot shows that Sudo-BIHT achieves slightly higher SDR than BIHT. As R increases, the SDR for both algorithms increases similarly. Note that the measurements acquired in noiseless 1-bit CS include quantization noise. The quantization noise explains why the SDR achieved in the noiseless 1-bit CS setting is finite, whereas unquantized noiseless measurements yield perfect reconstruction [2, 3]. In the bottom panel of Figure 6, we plot SDR as a function runtime. Note that Sudo-BIHT can achieve the same SDR as BIHT despite running an order of magnitude faster.

[4]We say that the consistency property of BIHT [20] is satisfied if applying the measurement and quantization system (16) and (17) to the reconstructed signal $\widehat{\mathbf{x}}$ yields the same measurements $\mathbf{y}$ as the original measurements.

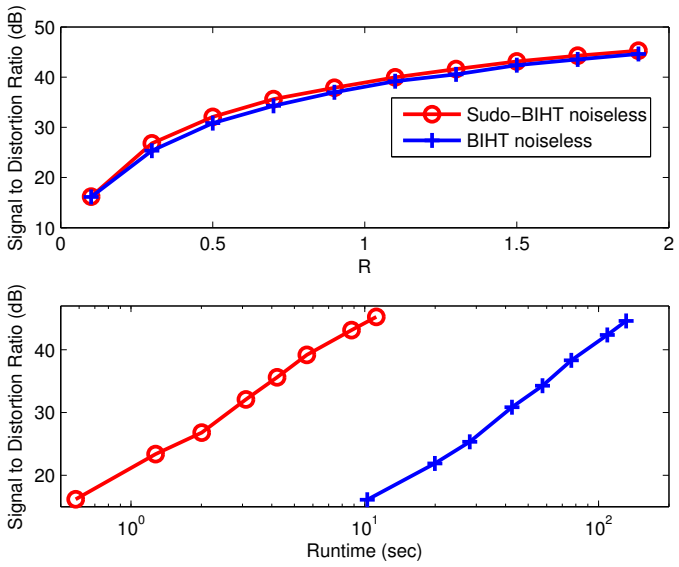

Figure 6: Numerical results of Noisy-Sudocodes with BIHT in Part 2 in a noiseless 1-bit CS setting. Top: SDR (15) as a function of measurement rate R. Bottom: SDR as a function of runtime. ($N = 10,000$, $s = 0.005$, $c = 1$, $d = 0.8$, $\epsilon = 0$, $M_1/N = 0.1$, and $M_2 = M - M_1$).

Noisy setting: BIHT-ℓ_2 [21], in which the ℓ_2-norm is utilized in the objective function, is applied to the noisy setting. Note that BIHT-ℓ_2 is more robust to pre-quantization noise than BIHT-ℓ_1. The measurement vector $\mathbf{y}_1$ for Part 1 of Sudo-BIHT is acquired via (18) with $\epsilon > 0$, and the measurement vectors $\mathbf{y}_2$ for Part 2 of Sudo-BIHT and $\mathbf{y}$ for BIHT are acquired via (17). We set $c = 3$, $d = 0.8$, and $\epsilon = 0.08$ in our simulations because they lead to sufficiently good performance in the sense that Sudo-BIHT improves over BIHT in both runtime and reconstruction quality.

The resulting SDR versus measurement rate R is shown in the top panel of Figure 7. When the number of iterations for BIHT is 30 in both Part 2 of Sudo-BIHT and BIHT, Sudo-BIHT yields better consistency and thus provides better reconstruction quality. With more iterations, the SDR for both Sudo-BIHT and BIHT improves. The SDR curve of BIHT tends to get closer to Sudo-BIHT as the number of iterations increases, because for Sudo-BIHT, the error introduced in Part 1 cannot be corrected by Part 2. We notice that Sudo-BIHT with 130 BIHT iterations (red solid line with circles) improves over BIHT with 30 iterations (blue dotted line with crosses) by roughly 5 dB for the same measurement rate, and the bottom panel of Figure 7 shows that the red solid line with circles can be 5 dB above the blue dotted line with crosses despite requiring approximately half of the runtime. In other words, problem size reduction due to zero-identification in Part 1 allows BIHT in Part 2 to run more iterations to improve reconstruction quality with reasonable runtime.

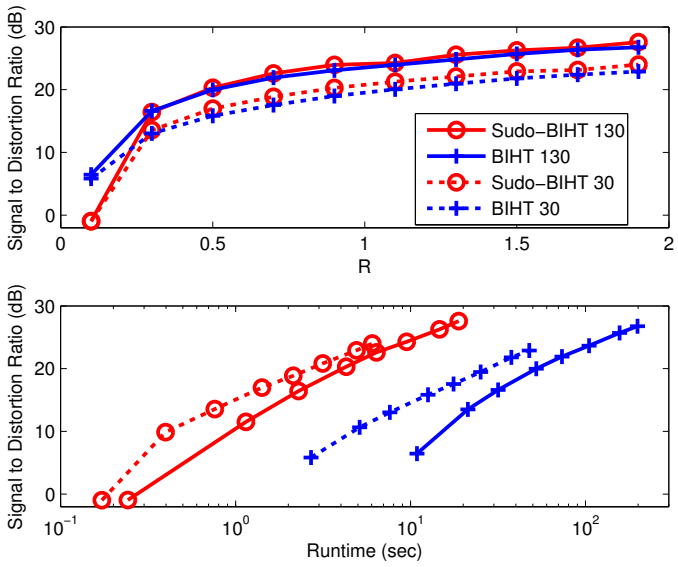

Figure 7: Numerical results of Noisy-Sudocodes with BIHT in Part 2 in a noisy 1-bit CS setting. Top: SDR (15) as a function of measurement rate R. Bottom: SDR as a function of runtime. ($N = 10,000$, $s = 0.005$, $c = 3$, $d = 0.8$, $\epsilon = 0.08$, $M_1/N = 0.1$, and $M_2 = M - M_1$).

V. CONCLUSION

We introduced a two-part reconstruction framework that partitions the reconstruction process into two complementary parts. The partitioning leads to a trade-off between runtime and reconstruction quality. A Noisy-Sudocodes algorithm was proposed within the two-part framework. Part 1 of Noisy-sudocodes is the zero-identification algorithm, whereas various CS reconstruction algorithms can serve as candidates for Part 2. We analyzed the speed-quality trade-off of Noisy-Sudocodes with AMP [12] in Part 2 based on the theoretical characterization that we derived for Part 1 and the well-established asymptotic properties of AMP. Moreover, numerical results for Noisy-Sudocodes with our 1-bit magnitude-quantizer in Part 1 and BIHT [21] in Part 2 imply that Noisy-Sudocodes could be promising for algorithm design in 1-bit CS reconstruction problems.

APPENDIX A: PROOF OF LEMMA 1

We will show that with the problem setting described in Section III-A, $\mathbf{y}_1$ is asymptotically independent in the limit of large N. The subscript that represents Part 1 is dropped in the following analysis. Denote the characteristic function of $\mathbf{x}$ by $\Psi_x(t) = \mathbb{E}\left[e^{itx}\right]$. We will show that for any constant $m \leq M$,

$$\lim_{N\to\infty} \Psi_{y_1 \ldots y_m}(t_1, ..., t_m) = \lim_{N\to\infty} \Psi_{y_1}(t_1) ... \Psi_{y_m}(t_m), \tag{20}$$

where $\Psi_{y_1 \ldots y_m}(t_1, ..., t_m) = \mathbb{E}\left[e^{it_1 y_1 + \ldots + it_c y_m}\right]$ is the joint characteristic function, and expectation is taken with respect to the joint probability density $P(y_1, ..., y_m)$. The joint characteristic function can be factorized as the product of the marginal characteristic functions as described in (20) if and only if $y_1, ..., y_m$ are independent [30].

To lighten the notation, we assume that the nonzero entries of the Bernoulli matrix Φ are ones (we adjusted the nonzero entries in the body of the paper to make the input SNR in Parst 1 and 2 identical), and we ignore the i.i.d. measurement noise.[5] Under these simplifications, the signal model is

$$\mathbf{y} = \Phi\mathbf{x} = \Phi_{*1}x_1 + \Phi_{*2}x_2 + ... + \Phi_{*N}x_N,$$

where Φ_{*j} represents the jth column of Φ. Define a sequence of random vectors $v_j = \Phi_{*j}x_j$, $j \in [N]$. Notice that $\{v_j\}_{j=1}^N$ are i.i.d. random vectors, and thus the characteristic function of the first m entries of $\mathbf{y}$ is

$$\Psi_y(t_1, ..., t_m) = \left(\Psi_{v_1}(t_1, ..., t_m)\right)^N.$$

It can be calculated that the characteristic function of a Gaussian random variable with probability density function (pdf) $\mathcal{N}(\mu, \sigma^2)$ is $e^{i\mu t - \frac{1}{2}\sigma^2 t^2}$.

To establish (20), it suffices to show that

$$\lim_{N\to\infty} \Psi_{y_1}(t_1)...\Psi_{y_m}(t_m) = e^{d\left(e^{-\frac{1}{2}t_1^2} + ... + e^{-\frac{1}{2}t_m^2} - m\right)} \tag{21}$$

and

$$\lim_{N\to\infty} \Psi_{y_1...y_m}(t_1, ..., t_m) = e^{d\left(e^{-\frac{1}{2}t_1^2} + ... + e^{-\frac{1}{2}t_m^2} - m\right)}. \tag{22}$$

First, we show (21). For $m = 1$, v_1 is a scalar. Recall that the Bernoulli parameter of the Bernoulli matrix in Part 1 is $\frac{d}{sN}$ and the sparsity rate of $\mathbf{x}$ is s. Let $g(x)$ denote the pdf of a Gaussian random variable x with mean 0 and variance 1. Denoting the probability distribution of v_1 by $P_{v_1}(u_1)$, we have

$$\begin{aligned}
P_{v_1}(u_1) &= \frac{d}{N}g(u_1) + \left(1 - \frac{d}{N}\right)\delta(u_1), \\
\Psi_{v_1}(t_1) &= 1 - \frac{d}{N} + \frac{d}{N}e^{-\frac{1}{2}t_1^2}, \\
\Psi_{y_1}(t_1) &= \left(1 - \frac{d}{N} + \frac{d}{N}e^{-\frac{1}{2}t_1^2}\right)^N, \\
\lim_{N\to\infty} \Psi_{y_1}(t_1) &= \lim_{N\to\infty} \left(1 + \frac{d}{N}\left(e^{-\frac{1}{2}t_1^2} - 1\right)\right)^N \\
&= e^{d\left(e^{-\frac{1}{2}t_1^2} - 1\right)}.
\end{aligned}$$

Because $\lim_{N\to\infty} \Psi_{y_i}(t_i)$ exists for every $i \in [M]$, for any finite constant m, we have

$$\begin{aligned}
\lim_{N\to\infty} \Psi_{y_1}(t_1)...\Psi_{y_m}(t_m) &= \lim_{N\to\infty} \Psi_{y_1}(t_1)... \lim_{N\to\infty} \Psi_{y_m}(t_m) \\
&= e^{d\left(e^{-\frac{1}{2}t_1^2} + ... + e^{-\frac{1}{2}t_m^2} - m\right)}.
\end{aligned}$$

Therefore, (21) is verified.

[5]Note that if entries of $\mathbf{y} = \Phi\mathbf{x}$ are independent, then after adding an i.i.d. noise vector $\mathbf{z}$, entries of $\mathbf{y}' = \mathbf{y} + \mathbf{z}$ are still independent. Therefore, these simplifications do not affect the independence relation among entries of $\mathbf{y}$.

Second, we show (22). For $m = 2$, v_1 is a vector of length 2. Denoting the probability distribution of v_1 by $P_{v_1}(u_1, u_2)$, we have

$$\begin{aligned}
&P_{v_1}(u_1,u_2) = P_{v_1}\left(u_1,u_2 \mid x_1=0\right)\mathbb{P}(x_1=0) + P_{v_1}\left(u_1,u_2 \mid x_1\neq 0\right)\mathbb{P}\left(x_1\neq 0\right)\\
&= \delta(u_1)\delta(u_2)\left(1-s\right)\\
&+\left[\left(1-\frac{d}{sN}\right)^2\delta_0(u_1)\delta(u_2) + \frac{d}{sN}\left(1-\frac{d}{sN}\right)\left(\delta(u_1)g(u_2)+\delta(u_2)g(u_1)\right) + \left(\frac{d}{sN}\right)^2\delta_0(u_2-u_1)g(u_1)\right]s\\
&= \left(1+\frac{d^2}{sN^2}-\frac{2d}{N}\right)\delta(u_1)\delta(u_2) + \left(\frac{d}{N}-\frac{d^2}{sN^2}\right)\left(\delta(u_1)g(u_2)+\delta(u_2)g(u_1)\right) + \frac{d^2}{sN^2}\delta(u_2-u_1)g(u_1),
\end{aligned}$$

$$\begin{aligned}
\Psi_{v_1}(t_1,t_2) &= \mathbb{E}\left[e^{it_1u_1+it_2u_2}\right]\\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{it_1u_1+it_2u_2}P_{v_1}(u_1,u_2)\mathrm{d}u_1\mathrm{d}u_2\\
&= \left(1+\frac{d^2}{sN^2}-\frac{2d}{N}\right) + \left(\frac{d}{N}-\frac{d^2}{sN^2}\right)\left(e^{-\frac{1}{2}t_1^2}+e^{-\frac{1}{2}t_2^2}\right) + \frac{d^2}{sN^2}e^{-\frac{1}{2}(t_1+t_2)^2}\\
&= 1+\frac{d}{N}\left(e^{-\frac{1}{2}t_1^2}+e^{-\frac{1}{2}t_2^2}-2\right) + \frac{d^2}{sN^2}\left(e^{-\frac{1}{2}(t_1+t_2)^2}-e^{-\frac{1}{2}t_1^2}-e^{-\frac{1}{2}t_2^2}+1\right),
\end{aligned}$$

$$\begin{aligned}
\lim_{N\to\infty}\Psi_{y_1y_2}(t_1,t_2) &= \lim_{N\to\infty}\left(\Psi_{v_1}(t_1,t_2)\right)^N\\
&= e^{d\left(e^{-\frac{1}{2}t_1^2}+e^{-\frac{1}{2}t_2^2}-2\right)}.
\end{aligned}$$

Similarly, it can be shown for any $m \leq M$ that

$$\begin{aligned}
\lim_{N\to\infty}\Psi_{y_1\ldots y_m}(t_1,\ldots,t_m) &= \lim_{N\to\infty}\left(\Psi_{v_1}(t_1,\ldots,t_m)\right)^N\\
&= e^{d\left(\sum_{i=1}^{c}e^{-\frac{1}{2}t_i^2}-m\right)}.
\end{aligned}$$

Therefore, (22) is also verified, which establishes (20).

We conclude that in the limit of large N, for each $j \in [N]$, the indicator variables $\{I_{ij}\}_{i=1,j=1}^{M,N}$ are independent along i. Therefore, $S_j = \sum_{i=1}^{M} I_{ij}$ converges to a Binomial random variable S_B in distribution [31], where $S_B \sim$ Binomial $\left(M, P_{\epsilon,d}(x_j)\right)$.

APPENDIX B: PROOF OF LEMMA 2

To simplify the notation, we drop the subscripts of $\Phi_{2,\text{FA}}$ and x_{FA}, and let Φ represent the submatrix formed by columns of Φ_2 at the indices FA, and $\mathbf{x}$ represent entries of $\mathbf{x}$ at the indices FA, where FA represents the false alarms defined in Section III-B. Define a sequence of vectors $v_j = \Phi_{*j}x_j$, $j \in [|\text{FA}|]$.

We notice that $\mathbf{z}_{\text{FA}}$ is a sum of i.i.d. random vectors, and the components in each vector are uncorrelated. That is,

$$\begin{aligned}\mathbf{z}_{\text{FA}} &= \sum_{j=1}^{|\text{FA}|} v_j,\\ \mathbb{E}[v_{jt}v_{js}] &= \mathbb{E}[(x_j\Phi_{jt})(x_j\Phi_{js})]\\ &= \mathbb{E}[x_j^2]\mathbb{E}[\Phi_{jt}\Phi_{js}]\\ &= \begin{cases} 0, & \text{if } s \neq t\\ \mathbb{E}[x_{\text{FA},j}^2]/N, & \text{if } s = t \end{cases}.\end{aligned}$$

The proof is completed by applying the Multivariate Central Limit Theorem.

Central Limit Theorem in $\mathbb{R}^d$ [31]: Let $(X_n)_{n\in\mathbb{N}}$ be i.i.d. random vectors with $\mathbb{E}[X_{n,i}] = 0$ and $\mathbb{E}[X_{n,i}X_{n,j}] = C_{ij}$, $i, j \in [d]$. Let $S_n^* = \frac{X_1+\ldots+X_n}{\sqrt{n}}$. Then

$$P_{S_n^*} \to \mathcal{N}(0, C) \quad \text{in distribution.}$$

By the multivariate central limit theorem, the distribution of the vector $\mathbf{z}_{\text{FA}}$ converges to $\mathcal{N}(0, C_{vv})$, where C_{vv} is a diagonal covariance matrix with $\mathbb{E}[\|\mathbf{x}_{\text{FA}}\|_2^2]/N$ on its diagonal. Therefore, $\mathbf{z}_{\text{FA}}$ converges to an i.i.d. Gaussian random vector in distribution.

ACKNOWLEDGMENTS

We thank Ilya Poltorak for useful discussions that led to the development of Noisy-Sudocodes; Wenbin Zhu for suggesting that we utilize the characteristic function for the proof of asymptotic independence in Lemma 1; Junan Zhu for providing us with Matlab code to calculate the MMSE for the matrix channel [27, 28]; and Jin Tan for suggestions that greatly improved the manuscript.

REFERENCES

[1] Y. Ma, D. Baron, and D. Needell, "Two-part reconstruction in compressed sensing," in *Proc. IEEE Global Conf. Signal Inf. Process.*, Austin, TX, Dec. 2013.

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

[3] E. Candès, 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.

[4] E. J. Candès, "Compressive sampling," in *Proc. Int. Congress of Mathematics*, vol. 3, Madrid, Spain, 2006, pp. 1433–1452.

[5] P. Indyk, "Explicit constructions for compressed sensing of sparse signals," in *Proc. 19th ACM-SIAM Symp. Discrete Algos.*, Jan. 2008, pp. 30–33.

[6] M. A. Iwen, "Compressed sensing with sparse binary matrices: Instance optimal error guarantees in near-optimal time," *Journal of Complexity*, vol. 30, Feb. 2014.

[7] S. Jafarpour, W. Xu, B. Hassibi, and R. Calderbank, "Efficient and robust compressed sensing using optimized expander graphs," *IEEE Trans. Inf. Theory*, vol. 55, no. 9, pp. 4299–4308, Sept. 2009.

[8] E. J. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure Appl. Math.*, vol. 59, pp. 1207–1223, Mar. 2006.

[9] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harm. Anal.*, vol. 26, no. 3, pp. 301–321, May 2009.

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

[11] S. Sarvotham, D. Baron, and R. G. Baraniuk, "Compressed sensing reconstruction via belief propagation," Rice University, Houston, TX, Tech. Rep. TREE0601, July 2006.

[12] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci.*, vol. 106, no. 45, pp. 18 914–18 919, Nov. 2009.

[13] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 764–785, Feb. 2011.

[14] A. Javanmard and A. Montanari, "Subsampling at information theoretically optimal rates," in *Proc. Int. Symp. Inf. Theory (ISIT)*, July 2012, pp. 2431–2435.

[15] J. Barbier, F. Krzakala, and C. Schulke, "Compressed sensing and approximate message passing with spatially-coupled Fourier and Hadamard operators," *Arxiv preprint arXiv:1312.1740*, Mar. 2014.

[16] S. Sarvotham, D. Baron, and R. G. Baraniuk, "Sudocodes – Fast measurement and reconstruction of sparse signals," in *Proc. Int. Symp. Inf. Theory (ISIT2006)*, Seattle, WA, July 2006.

[17] M. Luby and M. Mitzenmacher, "Verification-based decoding for packet-based low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 20, no. 1, pp. 120–127, Jan. 2005.

[18] F. Zhang and H. D. Pfister, "Verication decoding of high-rate LDPC codes with applications in compressed sensing," *IEEE Trans. Inf. Theory*, vol. 58, no. 8, pp. 5042–5058, Aug. 2012.

[19] A. Talari and N. Rahnavard, "GBCS: A two-step compressive sensing reconstruction based on group testing and basis pursuit," in *Military Comm. Conf.*, Nov. 2011, pp. 157–162.

[20] P. Boufounos and R. Baraniuk, "1-bit compressive sensing," in *Proc. 2008 Conf. Inf. Sciences Systems*, Mar. 2008, pp. 16–21.

[21] 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, Apr. 2013.

[22] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. McGraw Hill Book Co., 1991.

[23] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," *Arxiv preprint arXiv:1010.5141*, Oct. 2010.

[24] A. Montanari, "Graphical models concepts in compressed sensing," *Compressed Sensing: Theory and Applications*, pp. 394–438, 2012.

[25] F. Krzakala, M. Mézard, F. Sausset, Y. Sun, and L. Zdeborová, "Probabilistic reconstruction in compressed sensing: Algorithms, phase diagrams, and threshold achieving matrices," *J. Stat. Mech. - Theory E.*, vol. 2012, no. 08, p. P08009, Aug. 2012.

[26] J. Zhu and D. Baron, "Performance regions in compressed sensing from noisy measurements," in *Proc. 2013 Conf. Inf. Sciences Systems*, Baltimore, MD, Mar. 2013.

[27] S. Rangan, A. K. Fletcher, and V. K. Goyal, "Asymptotic analysis of MAP estimation via the replica method and applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 58, no. 3, pp. 1902–1923, Mar. 2012.

[28] D. Guo, D. Baron, and S. Shamai, "A single-letter characterization of optimal noisy compressed sensing," in *Proc. 47th Allerton Conf. Commun., Control, Comput.*, Sept. 2009, pp. 52–59.

[29] Y. Plan and R. Vershynin, "One-bit compressed sensing by linear programming," *Comm. Pure Appl. Math.*, vol. 66, pp. 1275–1297, Aug. 2013.

[30] D. Applebaum, B. Bhat, J. Kustermans, and J. Lindsay, *Quantum Independent Increment Processes I: From Classical Probability to Quantum Stochastic Calculus (Lecture Notes in Mathematics)*. New York, NY, USA: Springer, Feb. 2005.

[31] A. Klenke, *Probability Theory: A Comprehensive Course*. New York, NY, USA: Springer, Sept. 2013.