Binary Compressive Sensing via Smoothed ℓ_0 Gradient Descent

Tianlin Liu

Department of Computer Science and Electrical Engineering

Jacobs University Bremen

28759 Bremen, Germany

Email: t.liu@jacobs-university.de

Dae Gwan Lee

Mathematisch-Geographische Fakultät

Katholische Universität Eichstätt-Ingolstadt

85071 Eichstätt, Germany

Email: daegwans@gmail.com

Abstract—We present a Compressive Sensing algorithm for reconstructing binary signals from its linear measurements. The proposed algorithm minimizes a non-convex cost function expressed as a weighted sum of smoothed ℓ_0 norms which takes into account the binariness of signals. We show that for binary signals the proposed algorithm outperforms other existing algorithms in recovery rate while requiring a short run time.

I. INTRODUCTION

Compressive Sensing (CS) is a method in signal processing which aims to reconstruct signals from a relatively small number of measurements. It has been shown that sparse signals can be reconstructed with a sampling rate far less than the Nyquist rate by exploiting the sparsity [1].

In this paper, we focus on Binary Compressive Sensing (BCS) which restricts the signals of interest to binary $\{0, 1\}$-valued signals, which are widely used in engineering applications, such as fault detection [2], single-pixel image reconstruction [3], and digital communications [4]. Related works are as follows: Nakarmi and Rahnavard [5] designed a sensing matrix tailored for binary signal reconstruction. Wang et al. [6] combined ℓ_1 norm with ℓ_∞ norm to reconstruct sparse binary signals. Nagahara [7] exploited the sum of weighted ℓ_1 norms to effectively reconstruct signals whose entries are integer-valued and, in particular, binary signals and bitonal images. Keiper et al. [8] analyzed the phase transition of binary Basis Pursuit.

We note that most of the previous work on BCS are based on convex optimization. Indeed, convex optimization based algorithms allow performance guarantee via rich mathematical tools. However, they are found to be notoriously slow in large-scale applications compared to greedy methods such as the Orthogonal Matching Pursuit (OMP) [9]. On the other hand, greedy methods like OMP are fast but often have a worse recovery rate than convex optimization methods. In this work, we propose a fast BCS algorithm with a high recovery rate. Taking the binariness of signals into account, our algorithm is a gradient descent method based on the smoothed ℓ_0 norm [10]. Through numerical experiments, we show that the proposed algorithm compares favorably against previously proposed CS and BCS algorithms in terms of recovery rate and speed.

The rest of the paper is organized as follows. We give a short review on CS/BCS algorithms in Section II and present our algorithm in Section III. In Section IV, we present experimental results which compare the performance of the proposed algorithm with other algorithms. We conclude this paper with some remarks in Section V.

Notations:

For a vector $\mathbf{v} = (v_1, \cdots, v_N)^\top$ and $1 \leq p \leq \infty$, the ℓ_p norm of $\mathbf{v}$ is denoted by $\|\mathbf{v}\|_p$. The number of non-zero entries in $\mathbf{v}$ is denoted by $\|\mathbf{v}\|_0$. The probability of an event E is denoted by $\mathbb{P}(E)$. Let $[N] = \{1, \cdots, N\}$ for $N \in \mathbb{N}$. We denote by $\mathbf{1}_N$ the N-dimensional vector with all entries equal to 1.

II. BINARY COMPRESSIVE SENSING (BCS)

In the standard CS scheme, one aims to recover a sparse signal from its linear measurements. The constraints posed by the measurements can be formulated as

$$\Phi \mathbf{z} = \mathbf{y}, \quad \mathbf{z} \in \mathbb{R}^N, \tag{1}$$

where $\Phi \in \mathbb{R}^{m \times N}$, $m \ll N$, is the measurement matrix and $\mathbf{y} = \Phi \mathbf{x} \in \mathbb{R}^m$ is the measurement of a *sparse* signal $\mathbf{x} \in \mathbb{R}^N$. CS algorithms exploit the fact that $\mathbf{x}$ is sparse and seek a sparse solution $\mathbf{z}$ satisfying (1).

The BCS scheme considers binary signals for $\mathbf{x}$. Note that a binary signal $\mathbf{x}$ is sparse if and only if its complementary binary signal $\widetilde{\mathbf{x}} := \mathbf{1}_N - \mathbf{x}$ is dense, i.e., is almost fully supported. As the measurement matrix Φ is known, the equation (1) converts equivalently to

$$\Phi \widetilde{\mathbf{z}} = \widetilde{\mathbf{y}}, \tag{2}$$

where $\widetilde{\mathbf{z}} := \mathbf{1}_N - \mathbf{z}$ and $\widetilde{\mathbf{y}} := \Phi \widetilde{\mathbf{x}} = \Phi \mathbf{1}_N - \mathbf{y}$. This shows that reconstructing a sparse signal $\mathbf{z}$ under the constraint (1) is equivalent to reconstructing a dense signal $\widetilde{\mathbf{z}}$ under the constraint (2). For this reason, in contrast to the case of generic signals, binary signals that are dense can be recovered as well as those that are sparse.

Two types of models for binary signals have been considered in the literature (e.g., [11], [6], [7]): (i) $\mathbf{x}$ is a deterministic vector which is binary and sparse, i.e., most of its entries are 0 and only few are 1; (ii) $\mathbf{x}$ is a random vector whose entries are independent and identically distributed (i.i.d.) with probability

distribution $\mathbb{P}(x_j = 1) = p$ for some fixed $0 \leq p \leq 1$. If p is small, a realization of $\mathbf{x}$ is likely a sparse binary signal.

In this work, we shall consider the second model which can accommodate dense binary signals as well as sparse binary signals.

Below we give a short review of CS/BCS methods that are related to our work.

A. ℓ_0 minimization (L0)

A naive approach to finding sparse solutions is the ℓ_0 minimization,

$$\min_{\mathbf{z} \in \mathbb{R}^N} \quad \|\mathbf{z}\|_0 \quad \text{subject to} \quad \Phi\mathbf{z} = \mathbf{y}. \tag{P_0}$$

This method works generally for continuous-valued signals that are sparse, i.e., signals whose entries are mostly zero. However, solving the ℓ_0 minimization requires a combinatorial search and is therefore NP-hard [12].

B. Smoothed ℓ_0 minimization (SL0)

Smoothed ℓ_0 minimization (SL0) [10] replaces the ℓ_0 norm in (P_0) with a non-convex relaxation:

$$\min_{\mathbf{z} \in \mathbb{R}^N} \quad \sum_{i=1}^{N} \left(1 - \exp\left(\frac{-z_i^2}{2\sigma^2} \right) \right) \quad \text{subject to} \quad \Phi\mathbf{z} = \mathbf{y}.$$

This is motivated by the observation

$$\lim_{\sigma \to 0} \exp\left(\frac{-t^2}{2\sigma^2} \right) = \begin{cases} 1 & \text{if } t = 0 \\ 0 & \text{if } t \neq 0, \end{cases}$$

which implies that for any $\mathbf{z} = (z_1, \ldots, z_N)^\top \in \mathbb{R}^N$,

$$\lim_{\sigma \to 0} \sum_{i=1}^{N} \left(1 - \exp\left(\frac{-z_i^2}{2\sigma^2} \right) \right) = \|\mathbf{z}\|_0. \tag{3}$$

Noticing that $\mathbf{z} \mapsto \sum_{i=1}^{N} \left(1 - \exp\left(\frac{-z_i^2}{2\sigma^2} \right) \right)$ is a smooth function for any fixed $\sigma > 0$, Mohimani et al. [10] proposed an algorithm based on the gradient descent method. The algorithm iteratively obtains an approximate solution by decreasing σ.

Mohammadi et al. [13] adapted the SL0 algorithm particularly to non-negative signals. Their algorithm, called the Constrained Smoothed ℓ_0 method (CSL0), incorporates the non-negativity constraints by introducing some weight functions into the cost function. Empirically, CSL0 shows better performance than SL0 in the reconstruction of non-negative signals.

C. Basis Pursuit (BP)

A well-known and by now standard relaxation of (P_0) is the ℓ_1-minimization, also known as the *Basis Pursuit* (BP) [14]:

$$\min_{\mathbf{z} \in \mathbb{R}^N} \quad \|\mathbf{z}\|_1 \quad \text{subject to} \quad \Phi\mathbf{z} = \mathbf{y}. \tag{P_1}$$

Similar to (P_0), this method works generally for continuous-valued signals $\mathbf{x} \in \mathbb{R}^N$ that are sparse.
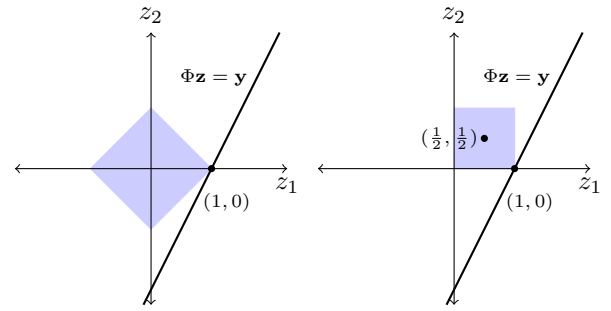

Fig. 1. Left: the minimization of $\|\mathbf{z}\|_1$ finds sparse solutions. Right: the minimization of $\|\mathbf{z} - \frac{1}{2} \cdot \mathbf{1}_N\|_\infty$ forces the entries $|z_i - \frac{1}{2}|$ to be small and of equal magnitude.
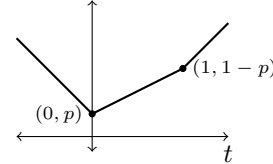

Fig. 2. The function f given in (4).

D. Boxed Basis Pursuit (Boxed BP)

Donoho et al. [11] proposed the Boxed Basis Pursuit (Boxed BP) for the reconstruction of *k-simple bounded signals*:

$$\min_{\mathbf{z} \in [0,1]^N} \quad \|\mathbf{z}\|_1 \quad \text{subject to} \quad \Phi\mathbf{z} = \mathbf{y}.$$

The intuition behind Boxed BP is straightforward: the ℓ_1 norm minimization promotes sparsity of the solution while the restriction $\mathbf{z} \in [0,1]^N$ reduces the set of feasible solutions. Recently, Keiper et al. [8] analyzed the performance of Boxed BP for reconstructing binary signals.

E. Sum of Norms (SN)

Wang et al. [6] introduced the following optimization problem which combines the ℓ_1 and ℓ_∞ norms:

$$\min_{\mathbf{z} \in \mathbb{R}^N} \quad \|\mathbf{z}\|_1 + \lambda \|\mathbf{z} - \frac{1}{2} \mathbf{1}_N\|_\infty \quad \text{subject to} \quad \Phi\mathbf{z} = \mathbf{y}.$$

Minimizing $\|\mathbf{z}\|_1$ promotes sparsity of $\mathbf{z}$ while minimizing $\|\mathbf{z} - \frac{1}{2} \mathbf{1}_N\|_\infty$ forces the entries $|z_i - \frac{1}{2}|$ to be small and of equal magnitude (see Fig. 1). The two terms are balanced by a tuning parameter $\lambda > 0$.

F. Sum of Absolute Values (SAV)

Nagahara [7] proposed the following method for reconstruction of discrete signals whose entries are chosen independently from a set of finite alphabets $\alpha = \{\alpha_1, \alpha_2, \ldots, \alpha_L\}$ with a priori known probability distribution. In the special case $\alpha = \{0, 1\}$ of binary signals, SAV is formulated as,

$$\min_{\mathbf{z} \in \mathbb{R}^N} \quad (1 - p) \|\mathbf{z}\|_1 + p \|\mathbf{z} - \mathbf{1}_N\|_1 \quad \text{subject to} \quad \Phi\mathbf{z} = \mathbf{y},$$

where $p = \mathbb{P}(x_j = 1)$, $j \in [N]$, is the probability distribution of the entries of $\mathbf{x}$. If $p \approx 0$, i.e., if $\mathbf{x}$ is sparse, then $(1 -$

$p) \|\mathbf{z}\|_1 + p \|\mathbf{z} - \mathbf{1}_N\|_1 \approx \|\mathbf{z}\|_1$ so that SAV performs similar to BP. We note that

$$(1-p) \|\mathbf{z}\|_1 + p \|\mathbf{z} - \mathbf{1}_N\|_1 = \sum_{i=1}^{N} f(z_i),$$

where

$$f(t) := \begin{cases} -t + p & \text{if } t < 0, \\ (1 - 2p)\, t + p & \text{if } 0 \le t < 1, \\ t - p & \text{if } t \ge 1. \end{cases} \qquad (4)$$

III. BOX-CONSTRAINED SUM OF SMOOTHED ℓ_0

L0 and SL0 utilize the ℓ_0 norm and its smoothed version respectively, however, they do not take into account that $\mathbf{x}$ is binary. On the other hand, Boxed BP, SN, and SAV utilize the ℓ_1 norm in one way or another and are specifically adjusted to the binary setting. A natural question arises: Can we achieve a better recovery rate for binary signals by adjusting L0 and SL0 to the binary setting?

We note that Boxed BP takes into account the binariness of $\mathbf{x}$ by imposing the restriction $\mathbf{x} \in [0,1]^N$. It is straightforward to apply this trick to L0 and SL0, and we will call the resulting algorithms *Boxed L0* and *Boxed SL0* respectively. Boxed L0 is still NP-hard like L0, but Boxed SL0 shows a clear improvement over SL0 while requiring a similar amount of run time (Fig. 3). However, the recovery rate of Boxed SL0 is significantly worse than Boxed BP or SN.

In this paper, we aim to adapt the SAV method and the restriction $\mathbf{x} \in [0,1]^N$ to SL0, in order to achieve a better performance. A straightforward adaptation leads to the following formulation. For $\sigma > 0$ small,

$$\min_{\mathbf{z} \in [0,1]^N} \quad F_\sigma(\mathbf{z}) \quad \text{subject to} \quad \Phi\mathbf{z} = \mathbf{y}, \qquad (5)$$

where

$$F_\sigma(\mathbf{z}) \triangleq (1-p) \sum_{i=1}^{N} \left(1 - e^{-z_i^2/(2\sigma^2)} \right)$$

$$+ p \sum_{i=1}^{N} \left(1 - e^{-(z_i - 1)^2/(2\sigma^2)} \right) \qquad (6)$$

$$= \sum_{i=1}^{N} \left(1 - (1-p)\, e^{-z_i^2/(2\sigma^2)} - p\, e^{-(z_i - 1)^2/(2\sigma^2)} \right)$$

and $p = \mathbb{P}(x_j = 1)$, $\forall j \in [N]$. Note that by (3), we have

$$\lim_{\sigma \to 0} F_\sigma(\mathbf{z}) = (1-p) \|\mathbf{z}\|_0 + p \|\mathbf{z} - \mathbf{1}_N\|_0$$

so that $F_0(\mathbf{z})$ can be approximated by $F_\sigma(\mathbf{z})$ with small $\sigma > 0$.

Next, we will use a weight function to incorporate the restriction $\mathbf{z} \in [0,1]^N$ into the function $F_\sigma(\mathbf{z})$. For integers $k \ge 1$, let

$$w_k(t) \triangleq \begin{cases} 1 & \text{if } 0 \le t \le 1 \\ k & \text{otherwise.} \end{cases}$$

For $\sigma > 0$ and integers $k \ge 1$, we define

$$F_{\sigma,k}^{\text{boxed}}(\mathbf{z})$$
$$\triangleq \sum_{i=1}^{N} w_k(z_i) \left(1 - (1-p)\, e^{-z_i^2/(2\sigma^2)} - p\, e^{-(z_i - 1)^2/(2\sigma^2)} \right).$$

Note that since $1 - (1-p)\, e^{-t^2/(2\sigma^2)} - p\, e^{-(t-1)^2/(2\sigma^2)} > 0$ for all $t \in \mathbb{R}$, minimizing $F_{\sigma,k}^{\text{boxed}}(\mathbf{z})$ forces $w_k(z_i)$ to be small so that all z_i's lie within $[0,1]$. In this way, the restriction $\mathbf{z} \in [0,1]^N$ is incorporated into the cost function. Our optimization problem now reads as follows: For $\sigma > 0$ small and $k \in \mathbb{N}$ large,

$$\min_{\mathbf{z} \in \mathbb{R}^N} \quad F_{\sigma,k}^{\text{boxed}}(\mathbf{z}) \quad \text{subject to} \quad \Phi\mathbf{z} = \mathbf{y}.$$

To solve this problem, we propose an algorithm which is based on the gradient descent method and is implemented similarly as algorithms in [10], [13]. A major difference in our algorithm is that the cost function $\sum_{i=1}^{N} \left(1 - \exp\left(\frac{-z_i^2}{2\sigma^2} \right) \right)$ of SL0 is replaced with $F_{\sigma,k}^{\text{boxed}}(\mathbf{z})$ which is designed specifically for binary signals by adapting the formulation of SAV [7].

Algorithm 1 Box-Constrained Sum of Smoothed ℓ_0 (BSSL0)

1: **Data:** Measurement matrix $\Phi \in \mathbb{R}^{m \times N}$, observation $\mathbf{y} \in \mathbb{R}^m$, probability distribution prior $p = \mathbb{P}(x_j = 1)$.
2: **Parameters:** Iters and L are the number of iterations in the outer and inner loops respectively, μ is a step-size parameter for gradient descent, and d is a decreasing factor for σ.
3: **Initialization:** $\hat{\mathbf{x}} = \Phi^\top (\Phi\Phi^\top)^{-1}\mathbf{y}$, $\sigma = 2 \max |\hat{\mathbf{x}}|$,
4: $\quad k = 1 + Np/\text{Iters}$;
5: **for** $1 : \text{Iters}$ **do**
6: $\quad$ **for** $1 : L$ **do**
7: $\quad\quad \hat{\mathbf{x}} \leftarrow \hat{\mathbf{x}} - \sigma^2 \mu \nabla F_{\sigma,k}^{\text{boxed}}(\mathbf{z});$ *% gradient descent*
8: $\quad\quad \hat{\mathbf{x}} \leftarrow \hat{\mathbf{x}} - \Phi^\top (\Phi\Phi^\top)^{-1}(\Phi\hat{\mathbf{x}} - \mathbf{y});$ *% projection*
9: $\quad$ **end for**
10: $\quad \sigma = \sigma \times d$;
11: $\quad k = k + Np/\text{Iters}$;
12: **end for**
13: $\hat{\mathbf{x}} \leftarrow \textbf{round}(\hat{\mathbf{x}});$ *% round to a binary vector*

The proposed algorithm is comprised of two nested loops. In the outer loop, we slowly decrease σ and iteratively search for an optimal solution from a coarse to a fine scale by decreasing σ by a factor of $0 < d < 1$. As σ decreases, we also gradually increase k so that a larger penalty is put on solutions that have entries outside the range $[0,1]$. The inner loop performs a gradient descent of L iterations for the function $F_{\sigma,k}^{\text{boxed}}(\mathbf{z})$, where σ and k are given from the outer loop. In each iteration of the gradient descent, the solution is projected into the set of feasible solutions $\{\mathbf{z} : \Phi\mathbf{z} = \mathbf{y}\}$.

Numerical experiments in Section IV show that for binary signals the proposed algorithm outperforms all other algorithms (BP, Boxed BP, SN, SAV, SL0, and Boxed SL0).

As already mentioned, our algorithm is implemented similarly as SL0 [10], [13]. The parameters used in our algorithm

are exactly the same as in [10] except k and p. As justified in [10, Section IV-B], we set the initial estimate of $\mathbf{x}$ as the minimum ℓ_2 norm solution of $\Phi\mathbf{z} = \mathbf{y}$, i.e., $\hat{\mathbf{x}} = \Phi^{\top}(\Phi\Phi^{\top})^{-1}\mathbf{y}$. The initialization value for σ is discussed in [10, Remark 5 in Section III]. Also, the choice of the step-size $\sigma^2\mu$ for gradient descent is justified in [10, Remark 2 in Section III] and the choice of k in [13, Lemma 1].

The gradient of $F_{\sigma,k}^{\text{boxed}}(\mathbf{z})$ used in Algorithm 1 is given by

$$\nabla F_{\sigma,k}^{\text{boxed}}(\mathbf{z}) = \left(\frac{\partial F_{\sigma,k}^{\text{boxed}}(\mathbf{z})}{\partial z_1}, \dots, \frac{\partial F_{\sigma,k}^{\text{boxed}}(\mathbf{z})}{\partial z_N}\right)^{\top},$$

where

$$\frac{\partial F_{\sigma,k}^{\text{boxed}}(\mathbf{z})}{\partial z_i} = \frac{w_k(z_i)}{\sigma^2}\left((1-p)\,z_i\,e^{-z_i^2/(2\sigma^2)}\right.$$
$$\left. + p\,(z_i - 1)\,e^{-(z_i-1)^2/(2\sigma^2)}\right) \quad a.e.$$

This is derived using the fact that $w_k'(t) = 0$ for all t except $t = 0, 1$; we have set $w_k'(0) = w_k'(1) = 0$ in the implementation. Let us point out that the discontinuity of $w_k(t)$ at $t = 0, 1$ does not deteriorate the performance of gradient descent. One can replace the function $w_k(t)$ with a smooth function, however, at the cost of increased run time.

IV. Numerical Experiments

In this section, we compare the performance of our algorithm BSSL0 with other CS/BCS algorithms described in Section II. The MATLAB codes for the experiments are available in [15].

A. Experiment 1: Binary Sparse Signal Reconstruction

In this experiment, we tested BSSL0 with randomly generated binary signals and compared it with other CS/BCS algorithms. Random Gaussian matrices are considered for the measurement matrix $\Phi \in \mathbb{R}^{40\times100}$, that is, all entries of Φ are drawn independently from the standard normal distribution. The parameter p is varied from 0 to 1 by step-size 0.05, and a binary signal $\mathbf{x} \in \{0, 1\}^{100}$ is generated by drawing its entries independently with $\mathbb{P}(x_i = 1) = p$ and $\mathbb{P}(x_i = 0) = 1 - p$. For Φ and $\mathbf{x}$, we compute the measurement vector $\mathbf{y} = \Phi\mathbf{x}$ and run the respective algorithms introduced in section II (BP, Boxed BP, SN, SAV, SL0, Boxed SL0, and BSSL0) to obtain a solution vector $\mathbf{z}$ as a approximated reconstruction of $\mathbf{x}$. Additionally, we consider the Orthogonal Matching Pursuit (OMP) [16] which is a fast greedy algorithm for sparse signal reconstruction. The following are considered for the performance evaluation: (i) **Failure of Perfect Reconstruction (FPR)**: 0 if $\mathbf{z} = \mathbf{x}$ (successfully recovered the signal perfectly) and 1 if $\mathbf{z} \neq \mathbf{x}$ (failed to recover perfectly); (ii) **Noise Signal Ratio (NSR)**: NSR $= \frac{\|\mathbf{x}-\mathbf{z}\|_2}{\|\mathbf{x}\|_2}$; (iii) **Run time**. For each p, experiments are repeated $10,000$ times and the results are averaged. For SN, we set the parameter λ to be 100 as fine-tuned in [6]. For BSSL0, we set $\sigma_{\min} = 0.1$, $d = 0.5$, $\mu = 2$, and $L = 1000$.

In Fig. 3, BSSL0 shows a better recovery rate than other CS/BCS algorithms and also shows a run time comparable to SL0.
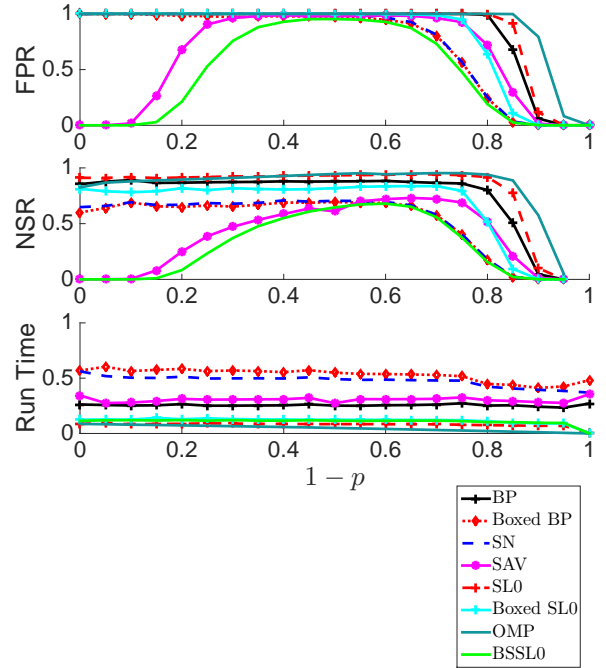

Fig. 3. Results for Experiment 1.

B. Experiment 2: Bitonal Image Reconstruction

As in [7], we considered reconstruction of the 37×37-pixel bitonal image given in Fig. 4 (left). Following the same setup in [7], we added to each pixel a random Gaussian noise with mean-zero and standard deviation of 0.1, as shown in Fig. 4 (right).

Fig. 4. Original image (left) and the image corrupted by Gaussian noise (right).

The noisy image is represented by a real-valued 37×37 matrix X and we apply the discrete Fourier transform (DFT) to obtain

$$\hat{X} = WXW \quad \in \mathbb{C}^{37\times37},$$

equivalently,

$$\text{vec}(\hat{X}) = (W \otimes W)\,\text{vec}(X) \quad \in \mathbb{C}^{1369},$$

where $W = [\omega^{k,\ell}]_{k,\ell=0}^{K-1}$ with $K = 37$ and $\omega = e^{-2\pi i/K}$ is the K-point DFT matrix. As in [7], we randomly subsampled $\text{vec}(\hat{X}) \in \mathbb{C}^{1369}$ to obtain a half-sized vector $\mathbf{y} \in \mathbb{C}^{685}$ and set the measurement matrix Φ as the corresponding 685×1369 submatrix of $W \otimes W$. Fig. 5 shows the reconstructed images by BP, SN, SAV, and BSSL0, all with entrywise rounding off

to $\{0,1\}$. For SN, an optimal tuning parameter λ was searched from 50 to 1000 by stepsize 50 and the value $\lambda = 800$ was chosen. For SAV and BSSL0, as in [7], we chose the parameter $p = \mathbb{P}(x_j = 0) = 0.5$ as a rough estimate for the sparsity of the bitonal image (see [7]). We set $\sigma_{\min} = 0.01$, $d = 0.9$, $\mu = 2$, and $L = 3$ for the parameters of BSSL0. The respective run time for BP, SN, SAV, and BSSL0 are also given in Tab. I.
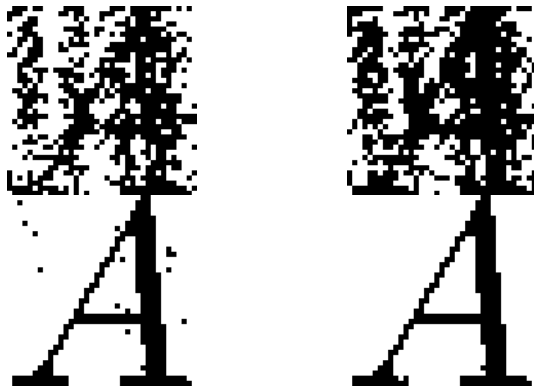

Fig. 5. Reconstructed images by BP (upper left), SN (upper right), SAV (lower left), and the proposed method BSSL0 (lower right).

TABLE I
THE RUN TIME COMPARISON

Algorithm	Run Time
Basis Pursuit	185.2044 seconds
SN	406.1007 seconds
SAV	191.5366 seconds
BSSL0 (proposed)	**0.92577 seconds**

V. CONCLUSION

In this work, we proposed a fast algorithm (BSSL0) for reconstruction of binary signals which is based on the gradient descent method and smooth relaxation techniques. We showed that for binary signals our algorithm outperforms other CS/BCS methods in terms of the recovery rate and speed. Future work includes a detailed analysis of BSSL0 in stability/robustness and extensions to ternary and finite alphabet signals.

ACKNOWLEDGMENT

REFERENCES

[1] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
[2] D. Bickson, D. Baron, A. Ihler, H. Avissar, and D. Dolev, "Fault identification via nonparametric belief propagation," *IEEE Transactions on Signal Processing*, vol. 59, no. 6, pp. 2602–2613, 2011.
[3] M. F. Duarte, M. A. Davenport, D. Takbar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE signal Processing Magazine*, vol. 25, no. 2, pp. 83–91, 2008.
[4] K. Wu and X. Guo, "Compressive sensing of digital sparse signals," in *Wireless Communications and Networking Conference (WCNC), 2011 IEEE*. IEEE, 2011, pp. 1488–1492.
[5] U. Nakarmi and N. Rahnavard, "Bcs: Compressive sensing for binary sparse signals," in *Military Communications Conference 2012*. IEEE, 2012, pp. 1–5.
[6] S. Wang and N. Rahnavard, "Binary compressive sensing via sum of ℓ_1-norm and ℓ_∞-norm regularization," in *Military Communications Conference 2013 IEEE*. IEEE, 2013, pp. 1616–1621.
[7] M. Nagahara, "Discrete signal reconstruction by sum of absolute values," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1575–1579, 2015.
[8] S. Keiper, G. Kutyniok, D. G. Lee, and G. E. Pfander, "Compressed sensing for finite-valued signals," *Linear Algebra and its Applications*, 2017.
[9] D. L. Donoho and Y. Tsaig, "Fast solution of ell_1-norm minimization problems when the solution may be sparse," *IEEE Transactions on Information Theory*, vol. 54, no. 11, pp. 4789–4812, 2008.
[10] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed ℓ_0 norm," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289–301, 2009.
[11] D. L. Donoho and J. Tanner, "Precise undersampling theorems," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 913–924, 2010.
[12] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, vol. 24, no. 2, pp. 227–234, 1995.
[13] M. Mohammadi, E. Fatemizadeh, and M. H. Mahoor, "Non-negative sparse decomposition based on constrained smoothed ℓ_0 norm," *Signal Processing*, vol. 100, pp. 42–50, 2014.
[14] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.
[15] T. Liu and D. G. Lee, "Matlab codes for BSSL0 (Box-constrained Sum of Smoothed ℓ_0) algorithm," https://github.com/liutianlin0121/BSSL0, 2018.
[16] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on information theory*, vol. 53, no. 12, pp. 4655–4666, 2007.