IET Signal Processing

Research Article

IET Journals
The Institution of Engineering and Technology

Adaptive one-bit quantisation via approximate message passing with nearest neighbour sparsity pattern learning 自适应一比特量化通过近似消息传递与最近邻居稀疏模式学习

ISSN 1751-9675
Received on 12th December 2017
Accepted on 19th January 2018
E-First on 16th April 2018
doi: 10.1049/iet-spr.2017.0568
www.ietdl.org

Hangting Cao[1], Jiang Zhu[1] ✉, Zhiwei Xu[1]

[1]*Key Laboratory of Ocean Observation-Imaging Testbed of Zhejiang Province, Ocean College, Zhejiang University, No.1 Zheda Road, Zhoushan 316021, People's Republic of China*

✉ *E-mail: jiangzhu16@zju.edu.cn*

Abstract: In this study, the problem of recovering structured sparse signals with a priori distribution whose structure patterns are unknown is studied from one-bit adaptive (AD) quantised measurements. A generalised approximate message passing (GAMP) algorithm is utilised, and an expectation maximisation (EM) method is embedded in the algorithm to iteratively estimate the unknown a priori distribution. In addition, the nearest neighbour sparsity pattern learning (NNSPL) method is adopted to further improve the recovery performance of the structured sparse signals. Numerical results demonstrate the effectiveness of GAMP-EM-AD-NNSPL method with both simulated and real data.

本文从一比特自适应(AD)量化测量出发，研究了结构稀疏信号在未知结构模式的先验分布下的恢复问题。利用广义近似消息传递算法，并在算法中嵌入期望最大化方法，迭代估计未知的先验分布。此外，采用最近邻稀疏模式学习(NNSPL)方法进一步提高了结构稀疏信号的恢复性能。数值结果表明了gampemad - nnspl方法在模拟和实际数据上的有效性。

1 Introduction

压缩感知

Compressed sensing (CS) aims to reconstruct signals from under-determined measurements [1], which has drawn a great deal of attention in recent years due to its wide range of applications such as magnetic resonance imaging, lensless imaging and network tomography [2–4]. Assuming the underlying signal is sparse in the selected dictionary, various algorithms have been proposed and the corresponding performance guarantees are provided [5–8]. Conventional CS assumes that the measurements have infinite precision. In practice, however, quantisation is a necessary step and measurements are mapped into discrete values before performing reconstruction [9]. In addition, low-rate quantisation is appealing in bandwidth-constrained wireless sensor networks and millimetre wave large-scale multiple-input multiple-output (MIMO) systems, as it can reduce the hardware complexity and power dissipation of the analogue-to-digital converter [10–12].

低速率量化在带宽受限的无线传感器网络中很受欢迎

The extreme case of low-rate quantisation is one-bit CS involving a single comparator, which was originally introduced in [13] and the corresponding model is $\boldsymbol{y} = \mathrm{sign}(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{\tau})$, where $\mathrm{sign}(\cdot)$ is the element-wise sign of the vector, and returns 1 for positive numbers and −1 otherwise. With the quantisation threshold τ being zero, the amplitude of $\boldsymbol{x}$ cannot be uniquely identified, and the norm of $\boldsymbol{x}$ is required to be one. Consequently, several numerical algorithms have been proposed based on one-bit measurements [13–15].

Recently, approximate message passing (AMP) algorithms have been proposed to estimate the unknown signals [16–18]. It is shown that AMP achieves the optimal under-sampling and sparsity tradeoff, and the computational complexity is significantly reduced compared to convex optimisation based approaches. In [19], a generalised AMP (GAMP) is proposed to cope with non-linear models, including quantisation, phase retrieval, MIMO data detection and channel estimation [20–22]. In [23], an expectation maximisation (EM) algorithm is incorporated into the GAMP, and unknown system parameters are learned iteratively. It is shown that the performance degradation of EM-based GAMP is negligible compared to that of conventional GAMP, given that the number of observations is large enough.

To improve the recovery performance, various strategies including noise dithering, periodic thresholds and adaptive thresholds are proposed to further improve the recovery performance [24]. It is shown that adaptive quantisation is the most efficient way in terms of the accuracy of the signal reconstruction, at the cost of feedback. In [25, 26], a GAMP algorithm is proposed to estimate the signal with independent priors, and numerical results demonstrate its superior performance compared to that of the least absolute shrinkage selection operator.

For structured known sparse signals, a turbo idea can be easily incorporated into GAMP [27]. By iteratively updating and exchanging the prior of signals, recovery performance can be improved. In addition, model-based compressive sensing is also proposed for recovering structured sparse signals [28]. In practice, the precise structure of signals is often unknown. In [29], an efficient algorithm termed AMP with nearest neighbour sparsity pattern learning (AMP-NNSPL) is proposed to learn the sparsity pattern of signals adaptively. Besides, the AMP-NNSPL algorithm is utilised and the accuracy of channel estimation improves significantly in sparse massive MIMO-OFDM (orthogonal frequency-division multiplexing) channels with approximately common support [30].

Some optimisation methods are also proposed to reconstruct the sparse signals from one-bit compressive measurements, such as the iterative reweighted algorithm (IRA) and the binary iterative hard thresholding algorithm (BIHT). IRA iteratively minimises a convex surrogate function that bounds the original objective function, which leads to an iterative reweighted process [31]. As for BIHT which extends the iterative hard thresholding [32] algorithm to one-bit quantisation setting, it has a simple and intuitive explanation and the reconstruction performance of BIHT is demonstrated theoretically and numerically [33]. However, BIHT needs to know the sparsity of signals. For GAMP, it can incorporate an EM algorithm to learn the unknown priors and is very natural to be combined with adaptive quantisation strategy. In addition, in [25], the reconstruction performance of GAMP is better than that of BIHT numerically.

In this paper, the problem of reconstructing unknown structured sparse signals from one-bit quantised measurements is studied. The main contribution of this work is to design the GAMP-EM-AD-NNSPL algorithm to recover the structured sparse signals and substantial numerical experiments are conducted to verify its effectiveness using both synthetic and real data. Referring to [23], the Bernoulli–Gaussian mixture is utilised to model the signal's prior, and EM algorithm is used to learn the unknown nuisance parameters [The nuisance parameters in this paper refer to the unknown parameters of the prior distribution.]. In addition, we combine the NNSPL method and the adaptive quantisation strategy to further improve the reconstruction performance of structured sparse signals, where the thresholds are updated sequentially and the element's priori distribution is updated as the average of its neighbours. The results of numerical experiments show that

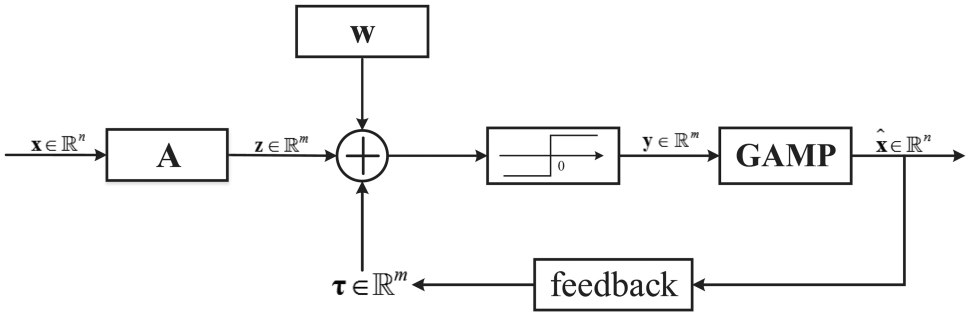

Fig. 1 *One-bit adaptive quantisation model. The noise* $\boldsymbol{w}$ *and thresholds* $\boldsymbol{\tau}$ *controlled by the output are added before quantisation. The GAMP algorithm is utilised to reconstruct the original signal* $\boldsymbol{x}$

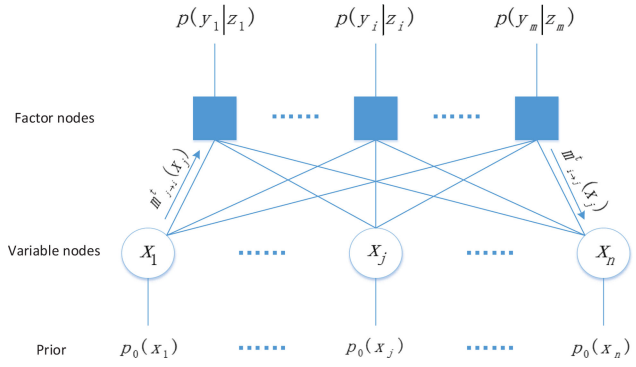

Fig. 2 *Factor graph representation of the joint distribution* $p(\boldsymbol{x}, \boldsymbol{y})$

GAMP-EM-AD-NNSPL performs well and sometimes runs faster than GAMP-EM-AD.

Notation: For matrices, we use capital boldface letters like $\boldsymbol{A}$ and we use $(\cdot)^{\mathrm{T}}$ for transpose operation. Let $\boldsymbol{A}_{i,j}$ denote the element in ith row and jth column of $\boldsymbol{A}$. For vectors, we use small boldface letters like $\boldsymbol{x}$ and we use $\| \boldsymbol{x} \|_p = (\sum_n |x_n|^p)^{1/p}$ to denote the ℓ_p norm of $\boldsymbol{x}$. Also, for a random vector $\boldsymbol{x}$, we denote its probability density function (pdf) by $p(\boldsymbol{x})$, and its expectation by $E[\boldsymbol{x}]$. Similarly, let $p(\boldsymbol{x}|\boldsymbol{y})$ and $E[\boldsymbol{x}|\boldsymbol{y}]$ denote the conditional pdf and expectation, respectively. Finally, we use $\mathbb{R}$ to denote the real field and $i \in \{a, b\}$ means i can only take values of a or b.

2 Problem setup

In this section, both the problem and the adaptive quantisation scheme are briefly introduced. The random vector $\boldsymbol{x} \in \mathbb{R}^n$ is passed through a random linear transform

$$\boldsymbol{z} = \boldsymbol{A}\boldsymbol{x}, \tag{1}$$

where $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ is a known random transform matrix. The goal is to recover $\boldsymbol{x}$ from its quantised samples $\boldsymbol{y} \in \mathbb{R}^m$, where

$$\boldsymbol{y} = \mathrm{sign}(\boldsymbol{z} + \boldsymbol{w} + \tau). \tag{2}$$

Here $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \Delta \boldsymbol{I})$ is an additive white Gaussian noise, $\boldsymbol{\tau}$ is the threshold. Both Δ and $\boldsymbol{\tau}$ are known. The elements of $\boldsymbol{x}$ are independent, and each x_i follows a Bernoulli–Gaussian distribution

$$p_0(x_i) = (1 - \pi_i)\delta(x_i) + \pi_i \mathcal{N}(x_i; \mu_0, \sigma_0), \tag{3}$$

where $\delta(\cdot)$ is the Dirac delta function, π_i denotes the probability of x_i being non-zero and $\mathcal{N}(x_i; \mu_0, \sigma_0)$ is the signal pdf whose mean and variance are μ_0 and σ_0 with x_i being active. The parameters $\theta = [\pi_1, \ldots, \pi_n, \mu_0, \sigma_0]^{\mathrm{T}}$ of the prior distribution are unknown. In addition, we also assume that the signal $\boldsymbol{x}$ has some specific block structure pattern which is unknown.

The adaptive quantiser works as follows (see Fig. 1). First, the whole threshold vector $\boldsymbol{\tau}$ is divided into two parts. The top m_0 elements $\boldsymbol{\tau}_{1:m_0}$ are designed offline and the remaining elements $\boldsymbol{\tau}_{m_0+1:m}$ are constructed sequentially according to previous measurements. Given the initial threshold vector $\boldsymbol{\tau}^0 = \boldsymbol{\tau}_{1:m_0} = \boldsymbol{0}_{m_0}$, then $\boldsymbol{y}_{1:m_0}$ is obtained and GAMP-EM-NNSPL is performed to obtain $\hat{\boldsymbol{x}}^0$. In the ensuing step, for the kth adaptive quantised measurements, the kth B elements of $\tau_{m_0+1:m}$ are sequentially acquired according to $\boldsymbol{\tau}_k = -\boldsymbol{A}_k \hat{\boldsymbol{x}}^{k-1}$, where $\boldsymbol{A}_k \in \mathbb{R}^{B \times n}$, $\hat{\boldsymbol{x}}^{k-1}$ is the final estimation from all the previous measurements and B is the block size of $\boldsymbol{\tau}$ updated in each adaptive quantisation. Here $\hat{\boldsymbol{x}}^{k-1}$ is reconstructed from the quantised samples with thresholds $\boldsymbol{\tau}^{k-1} = [\boldsymbol{\tau}^{k-2}; \boldsymbol{\tau}_{k-1}]$. The adaptive quantisation is ended when the number of measurements is reached.

3 Signal reconstruction

In this section, AMP is reviewed and then the GAMP-EM-NNSPL is presented.

To recover the signals, we aim to calculate minimum mean-squared error estimates of $\boldsymbol{x}$ which are the means of the marginal posteriors $p_{\boldsymbol{x}|\boldsymbol{y}}(\boldsymbol{x}|\boldsymbol{y})$. For our problem, the posterior distribution is

$$\begin{aligned} p_{\boldsymbol{x}|\boldsymbol{y}}(\boldsymbol{x}|\boldsymbol{y}) &= p_{\boldsymbol{y}|\boldsymbol{x}}(\boldsymbol{y}|\boldsymbol{x}) p_0(\boldsymbol{x}) / p_{\boldsymbol{y}}(\boldsymbol{y}) \\ &\propto p_{\boldsymbol{y}|\boldsymbol{z}}(\boldsymbol{y}|\boldsymbol{z}) p_0(\boldsymbol{x}) \\ &= \prod_i p_{y_i|z_i}(y_i | \sum_j a_{ij} x_j) \prod_j p_0(x_j), \end{aligned} \tag{4}$$

which can be represented by a factor graph as shown in Fig. 2. In the figure, circles denote variable nodes and squares denote factor nodes. Besides, each variable node is connected to every factor node. From the perspective of loopy belief propagation (LBP), beliefs about the random variable are in the form of pdfs or log pdfs and passed through the nodes in the factor graph until they converge [34]. One way to compute the beliefs is sum-product algorithm [35], where we calculate the belief $m^t_{j \to i}(x_j)$ from a variable node to a certain factor node as the product of the prior associated with that variable node and the incoming beliefs from all other factor nodes, and calculate the belief $m^t_{i \to j}(x_j)$ propagated in the opposite direction as the integral of the product of the factor related to that factor node and the incoming beliefs from all other variable nodes. The product of all beliefs gathered at a given variable node produces the posterior pdf for that variable. Then we can simplify the LBP via central limit theorem and Taylor expansion and obtain the AMP algorithm, which is precise in the large system limit [16, 36].

Next, the GAMP-EM-NNSPL is presented. Similar to [25], our proposed algorithm can be summarised as follows:

(1) Initialisation: Set $t = 1$, $\pi_i^0 = \alpha_i \in (0, 1)$, $\mu_0^0 = 0$ and $\sigma_0^0 = \beta > 0$, and evaluate

$$\boldsymbol{v}_x^0 = \mathrm{var}[\boldsymbol{x}], \quad \hat{\boldsymbol{x}}^0 = E[\boldsymbol{x}], \quad \hat{\boldsymbol{s}}^0 = \boldsymbol{0}, \tag{5a}$$

where the expected value and variance are taken with respect to the initialised prior.

(2) Factor update: First, compute the output linear step

$$\boldsymbol{v}_p^t = (\boldsymbol{A} \cdot \boldsymbol{A})\boldsymbol{v}_x^{t-1}, \tag{6a}$$

$$\hat{\boldsymbol{p}}^t = \boldsymbol{A}\hat{\boldsymbol{x}}^{t-1} - \boldsymbol{v}_p^t \cdot \hat{\boldsymbol{s}}^{t-1}, \tag{6b}$$

where $\cdot$ denotes the Hadamard product. Then evaluate the output non-linear step

$$\hat{\boldsymbol{s}}^t = E_1\big(\boldsymbol{y}, \hat{\boldsymbol{p}}^t, \boldsymbol{v}_p^t, \Delta\big), \tag{7a}$$

$$\boldsymbol{v}_s^t = V_1\big(\boldsymbol{y}, \hat{\boldsymbol{p}}^t, \boldsymbol{v}_p^t, \Delta\big), \tag{7b}$$

where the scalar functions E_1 and V_1 are applied componentwise and are given by

$$E_1\big(\boldsymbol{y}, \hat{\boldsymbol{p}}^t, \boldsymbol{v}_p^t, \Delta\big) = \frac{1}{v_p}(\mathrm{E}[z|y] - \hat{p}), \tag{8a}$$

$$V_1\big(\boldsymbol{y}, \hat{\boldsymbol{p}}^t, \boldsymbol{v}_p^t, \Delta\big) = \frac{1}{v_p}\left(1 - \frac{\mathrm{var}[z|y]}{v_p}\right), \tag{8b}$$

where the expected value and the variance are evaluated with respect to $z \sim \mathcal{N}(\hat{p}, v_p)$ given y. Note that $E[z|y]$ and $\mathrm{var}[z|y]$ can be obtained via the following equations:

$$E(z|y) = \hat{p} + yv_p\mathcal{N}(0, \hat{p} + \tau, v_p + \Delta)/\Phi\left(y\frac{\tau + \hat{p}}{\sqrt{\Delta + v_p}}\right), \tag{9a}$$

$$\begin{aligned} E\big(z^2|y\big) = {} & v_p + \hat{p}^2 + yv_p\mathcal{N}\big(0, \hat{p} + \tau, v_p + \Delta\big) \\ & \times \left(\frac{\hat{p}\Delta - \tau v_p}{v_p + \Delta} + \hat{p}\right)/\Phi\left(y\frac{\tau + \hat{p}}{\sqrt{\Delta + v_p}}\right). \end{aligned} \tag{9b}$$

(3) Variable update: First, compute the input linear step

$$\boldsymbol{v}_r^t = \left((\boldsymbol{A} \cdot \boldsymbol{A})^{\mathrm{T}}\boldsymbol{v}_s^t\right)^{-1}, \tag{10a}$$

$$\hat{\boldsymbol{r}}^t = \hat{\boldsymbol{x}}^{t-1} + \boldsymbol{v}_r^t \cdot \left(\boldsymbol{A}^{\mathrm{T}}\hat{\boldsymbol{s}}^t\right). \tag{10b}$$

Then evaluate the input non-linear step

$$\hat{\boldsymbol{x}}^t = E_2\big[\hat{\boldsymbol{r}}^t, \boldsymbol{v}_r^t; p_x\big], \quad \boldsymbol{v}_x^t = V_2\big[\hat{\boldsymbol{r}}^t, \boldsymbol{v}_r^t; p_x\big], \tag{11}$$

where the scalar functions E_2 and V_2 are applied componentwise and are given by

$$E_2[\hat{r}, v_r; p_x] = E[x|\hat{r}], \quad V_2[\hat{r}, v_r; p_x] = \mathrm{var}[x|\hat{r}], \tag{12}$$

where $\hat{r} = x + \tilde{w}$ and $\tilde{w} \sim \mathcal{N}(0, v_r)$ and $x \sim p_x(x)$. The expressions for $E[x|\hat{r}]$ and $\mathrm{var}[x|\hat{r}]$ are given by (18).

(4) EM learning for updating unknown parameters, i.e. updating π_i^t, μ_0^t, σ_0^t according to the later (21) and (22).

(5) Set $t = t + 1$ and return to step (2).

(6) Stop criterion: $t = T_{\max}$ or the number of events $\| \hat{\boldsymbol{x}}^t - \hat{\boldsymbol{x}}^{t-1} \|_1 / n < 10^{-6}$ in step (3) is equal to 3.

3.1 Derivation of (8)

In the following text, the variable index is omitted. The scalar signal generation model is $y = \mathrm{sign}(z + \tau + w)$, where $z \sim \mathcal{N}(\hat{p}, v_p)$ is given by (6). It follows that $p(y|z) = \Phi(y(z + \tau)/\sqrt{\Delta})$. Since $z + \tau + w \sim \mathcal{N}(\tau + \hat{p}, \Delta + v_p)$, we have

$$p(y) = \Phi(y(\tau + \hat{p})/\sqrt{\Delta + v_p}).$$

Using Bayes formula $p(z|y) = p(y|z)p(z)/p(y)$ and after some simplifications, we obtain (9), where $\Phi(\cdot)$ is a standard normal cumulative distribution function and $\mathrm{var}[z|y] = E[z^2|y] - E^2[z|y]$.

3.2 Derivation of (12)

For the variable update step, the unknown system parameter vector is $\boldsymbol{\theta} = [\pi_1, \ldots, \pi_n, \mu_0, \sigma_0]^{\mathrm{T}}$, and

$$\hat{r} = x + \tilde{w}^t, \tag{13}$$

where $\tilde{w}^t \sim \mathcal{N}(0, v_r^t)$ and $x \sim p_x\big(x; \boldsymbol{\theta}^t\big)$. The posterior distribution of x can be estimated as

$$p\big(x|y; \boldsymbol{\theta}^t\big) = q\big(x|\hat{r}^t, v_r^t\big) = \frac{1}{Z\big(\hat{r}^t, v_r^t\big)}p_x\big(x; \boldsymbol{\theta}^t\big)\mathcal{N}\big(x; \hat{r}^t, v_r^t\big),$$

where $Z\big(\hat{r}^t, v_r^t\big)$ is a normalisation constant, $\hat{r}^t$ and v_r^t are the noisy observation of x and the variance of noise $\tilde{w}$ (13). After some manipulations, the posterior distribution $q\big(x|\hat{r}^t, v_r^t\big)$ can be written as

$$q\big(x|\hat{r}^t, v_r^t\big) = \big(1 - \lambda^t\big)\delta(x) + \lambda^t\mathcal{N}\big(x; m^t, V^t\big), \tag{14}$$

where

$$m^t = \frac{\sigma_0^t\hat{r}^t + v_r^t\mu_0^t}{v_r^t + \sigma_0^t}, \quad V^t = \frac{\sigma_0^t v_r^t}{v_r^t + \sigma_0^t}, \tag{15}$$

$$\lambda^t = \frac{\pi^t}{\pi^t + (1 - \pi^t)\exp(-M^t)}, \tag{16}$$

where M^t in (16) is

$$M^t = \frac{1}{2}\log\frac{v_r^t}{v_r^t + \sigma_0^t} + \frac{\big(\hat{r}^t\big)^2}{2v_r^t} - \frac{\big(\hat{r}^t - \mu_0^t\big)^2}{2\big(v_r^t + \sigma_0^t\big)}. \tag{17}$$

Consequently, the posterior mean and variance are

$$E\big(x_i|\hat{r}^t\big) = \lambda^t m^t, \tag{18a}$$

$$\mathrm{var}\big[x_i|\hat{r}^t\big] = \lambda^t\Big(\big(m^t\big)^2 + V^t\Big) - \big(\lambda^t m^t\big)^2. \tag{18b}$$

3.3 EM learning

In the case of unknown system parameters, the EM algorithm can be adopted to reconstruct the parameters in the iterative process [37]. The conditional probability of $\boldsymbol{y}$ given $\boldsymbol{x}$ is

$$p(\boldsymbol{y}|\boldsymbol{x}) = \prod_{i=1}^{m}\Phi\left(y_i\frac{z_i + \tau_i}{\sqrt{\Delta}}\right). \tag{19}$$

Then we calculate the complete log-likelihood function $\log p(\boldsymbol{x}, \boldsymbol{y}) = \log p(\boldsymbol{x}) + \log p(\boldsymbol{y}|\boldsymbol{x})$, where $p(\boldsymbol{x})$ is given by (3). Given the current estimate $\boldsymbol{\theta}^t = [\pi_1^t, \ldots, \pi_n^t, \mu_0^t, \sigma_0^t]^{\mathrm{T}}$, we obtain the expectation of the complete log-likelihood function $Q\big(\boldsymbol{\theta}, \boldsymbol{\theta}^t\big) = E\big[\log p(\boldsymbol{x}, \boldsymbol{y})|\boldsymbol{y}, \boldsymbol{\theta}^t\big]$ given by (20). By setting $\partial Q(\boldsymbol{\theta}, \boldsymbol{\theta}^t)/\partial\boldsymbol{\theta} = \boldsymbol{0}$, one obtains

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^t) = {} & \sum_{i=1}^{n}((1 - \lambda_i^t)\log(1 - \pi_i) + \lambda_i^t\log\pi_i \\ & - \lambda_i^t\left(\frac{\big(m_i^t - \mu_0\big)^2 + V_i^t}{2\sigma_0} + \frac{1}{2}\log 2\pi\sigma_0\right)) \\ & + \sum_{i=1}^{m}E\left[\log\left(\Phi\left(y_i\frac{z_i + \tau_i}{\sqrt{\Delta}}\right)\right)|\boldsymbol{y}\right] \end{aligned} \tag{20}$$

$$\pi_i^{t+1} = \lambda_i^t, \tag{21a}$$

$$\mu_0^{t+1} = \sum_{i=1}^{n} \lambda_i^t m_i^t / \sum_{i=1}^{n} \lambda_i^t, \tag{21b}$$

$$\sigma_0^{t+1} = \sum_{i=1}^{n} \lambda_i^t \left[\left(\mu_0^t - m_i^t \right)^2 + V_i^t \right] / \sum_{i=1}^{n} \lambda_i^t . \tag{21c}$$

For the NNSPL learning method, instead of through (21a), its non-zero prior probability is updated via

$$\pi_i^{t+1} = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} \lambda_j^t, \tag{22}$$

where $\mathcal{N}(i)$ is the set of nearest neighbour indexes of element x_i in $\boldsymbol{x}$ and $|\mathcal{N}(i)|$ denotes the cardinality of $\mathcal{N}(i)$ [29]. Note that (22) updates non-zero prior probability π_i via utilising the information of its nearest neighbour's elements, which works well with block-sparse signals, as shown in numerical experiments. For the single vector setting, (22) can also be written as

$$\boldsymbol{\pi}^{t+1} = \boldsymbol{T} \boldsymbol{\lambda}^t, \tag{23}$$

where $\boldsymbol{T} \in \mathbb{R}^{n \times n}$ is

$$T_{i,j} = \begin{cases} 1, & i = 1 \& j = 2 \text{ or } i = n \& j = n-1 \\ 0.5, & j \in \{i-1, i+1\} \& i \notin \{1, n\} \\ 0, & \text{otherwise} \end{cases}, \tag{24}$$

$i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n$.

3.4 Extension to matrix setting

In applications such as image processing, one needs to represent the signal with a matrix. Therefore, we extend the algorithm to a matrix form.

For a matrix form, the problem model can be rewritten as

$$\boldsymbol{Y} = \text{sign}(\boldsymbol{AX} + \boldsymbol{W} + \boldsymbol{\Gamma}), \tag{25}$$

where $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ and $\boldsymbol{Y} \in \mathbb{R}^{m \times p}$. Let $\boldsymbol{y}_i$, $\boldsymbol{x}_i$, $\boldsymbol{w}_i$ and $\boldsymbol{\tau}_i$ represent the ith column of $\boldsymbol{Y}$, $\boldsymbol{X}$, $\boldsymbol{W}$ and $\boldsymbol{\Gamma}$, respectively. Then model (25) can be decomposed as

$$\boldsymbol{y}_i = \text{sign}(\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{w}_i + \boldsymbol{\tau}_i), \quad i = 1, 2, \ldots, p . \tag{26}$$

We apply the previous GAMP-EM-AD directly on model (26) separately, and finally update its non-zero prior probability through the NNSPL learning method which is different from that in the vector form. For a vector signal $\boldsymbol{x}$, $\mathcal{N}(i) = \{i-1, i+1\}$ and $|\mathcal{N}(i)| = 2$, while for a matrix signal $\boldsymbol{X}$, $\mathcal{N}(i) = \{(i, j-1), (i, j+1), (i-1, j), (i+1, j)\}$ and $|\mathcal{N}(i)| = 4$. So we need to apply the NNSPL learning method on the whole prior probability, instead of (23). After some simple manipulations, the NNSPL method in matrix setting can be described as

$$\boldsymbol{\Pi}^{t+1} = (\boldsymbol{L}\boldsymbol{\Lambda}^t + \boldsymbol{\Lambda}^t \boldsymbol{R}) \cdot \boldsymbol{H} . \tag{27}$$

Here $\boldsymbol{L} \in \mathbb{R}^{n \times n}$, $\boldsymbol{R} \in \mathbb{R}^{p \times p}$, $\boldsymbol{H} \in \mathbb{R}^{n \times p}$ and $\boldsymbol{\Lambda} = [\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \ldots, \boldsymbol{\lambda}_p] \in \mathbb{R}^{n \times p}$. $\boldsymbol{L}$ can be obtained through

$$\boldsymbol{L}_{i,j} = \begin{cases} 1, & j \in \{i-1, i+1\} \\ 0, & \text{otherwise} \end{cases}, \tag{28}$$

$i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n$. $\boldsymbol{R}$ has the same form as $\boldsymbol{L}$ and can be obtained by replacing n with p. Besides, $\boldsymbol{H}$ is

$$\boldsymbol{H}_{i,j} = \begin{cases} 1/2, & i \in \{1, n\} \& j \in \{1, p\} \\ 1/4, & i \notin \{1, n\} \& j \notin \{1, p\}, \\ 1/3, & \text{otherwise} \end{cases} \tag{29}$$

$i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, p$.

4 Numerical experiments

In this section, both synthetic and real data are used to validate the effectiveness of GAMP-EM-AD-NNSPL, compared to GAMP-EM-AD and GAMP-oracle-AD, where GAMP-oracle-AD denotes the signal reconstruction method with known prior distribution.

4.1 Synthetic data set

For the simulations in this subsection, synthetic block-sparse signals are generated in a similar way as [38], where K non-zero elements are partitioned into eight blocks with random sizes and random locations. The other parameters are set as follows: $n = 1024$, $\pi_i = K/n$, $\mu_0 = 0.5$, $\sigma_0 = 1$, $T_{\max} = 100$. We initialise $\pi_i^0 = 2\pi_i$, $\boldsymbol{v}_x^0 = 10\pi_i$, $\hat{\boldsymbol{x}}^0 = \boldsymbol{0}$, $\mu_0^0 = 0$, $\sigma_0^0 = 10$, and the damping factor is 0.5. For the algorithms using adaptive measurements, we initialise $\boldsymbol{\tau}^{m_0} = \boldsymbol{0}_{m_0}$. For $m = n$ and $m > n$, the initialised numbers of measurements are $m_0 = 0.5n$ and $m_0 = n$, respectively. The block size for adaptive quantisation is $B = 200$. Note that the signal-to-noise ratio (SNR) of the reconstruction is calculated as $\text{SNR} = 10\log\left(\| x \|_2^2 / \| x - \hat{x} \|_2^2 \right)$ and the success rate is calculated as the ratio of the number of successful trials to the total number of trials, where a trial is regarded as successful if the SNR of the reconstruction is larger than S dB, where S is a parameter we set later. All the results are averaged over 100 Monte Carlo trials.

The first simulation compares the three GAMP algorithms that do not utilise adaptive measurements against IRA and BIHT [The MATLAB codes of IRA are downloaded at http://www.junfang-uestc.net/codes/OnebitCS.rar and the MATLAB codes of BIHT are downloaded at http://dsp.rice.edu/software/.] in a noiseless scenario, where we set $K = 250$, $\Delta = 0$, $S = 15$, $\boldsymbol{\tau} = \boldsymbol{0}$. Since IRA and BIHT lose the amplitude information of signals, the $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ are both normalised in the calculation of SNR for fair comparison, i.e. $\boldsymbol{x} = \boldsymbol{x} / \| \boldsymbol{x} \|_2^2$ and $\hat{\boldsymbol{x}} = \hat{\boldsymbol{x}} / \| \hat{\boldsymbol{x}} \|_2^2$. Figs. 3*a* and *b* depict the average reconstruction SNR and the success rate, respectively. It can be seen that the three GAMP algorithms outperform IRA and BIHT over the whole range of the measurement ratio m/n and GAMP-EM-NNSPL works best. In addition, GAMP-EM-NNSPL also has the highest success rate at different measurement ratios and the success rates of GAMP-EM, IRA and BIHT do not achieve 1 even when m/n is 4. Fig. 3*c* presents the average running time of the algorithms. It can be seen that the average running time of BIHT is the shortest, while the average running time of the three GAMP algorithms are close to each other. Note that IRA involving matrix inversion runs too slowly and it takes about 8.4565 s to run a trial when m/n is 1. Since the average running time of IRA is so long, we do not compare with it.

The second and third simulations study the reconstruction performance of the three adaptive GAMP algorithms in a noiseless scenario and a noisy scenario, separately, where $K = 250$ and $S = 20$. We set Δ such that the SNR of the signal satisfies $\text{SNR} = 10\log(\| \boldsymbol{Ax} \|_2^2 / m\Delta) = 20\,\text{dB}$ in the noisy scenario and $\Delta = 0$ for the noiseless scenario. From Fig. 4*a*, we can see that GAMP-EM-AD-NNSPL works better than GAMP-oracle-AD and GAMP-oracle-AD outperforms GAMP-EM-AD at all the measurement ratios. Besides, the reconstructed SNR of GAMP-EM-AD-NNSPL is ∼15 dB higher than that of GAMP-EM-AD when m/n is 2.5. Fig. 4*b* shows that GAMP-EM-AD-NNSPL is the first to achieve the success rate of 1 as the measurement ratio m/n increases. Both figures illustrate the best reconstruction performance of GAMP-EM-AD-NNSPL. In Fig. 4*c*, GAMP-oracle-AD runs fastest and GAMP-GAMP-EM-AD-NNSPL runs faster than GAMP-EM-AD when the measurement ratio is larger than 2. From Figs. 5*a* and *b*, it can be seen that the result is similar to Figs. 4*a* and *b*. However, the performance of the three algorithms degrades significantly in the presence of noise, both in the reconstruction SNR and the success rate. In Fig. 5*c*, GAMP-EM-AD and GAMP-EM-AD-NNSPL run more slowly and GAMP-oracle-AD runs faster in the presence of noise.

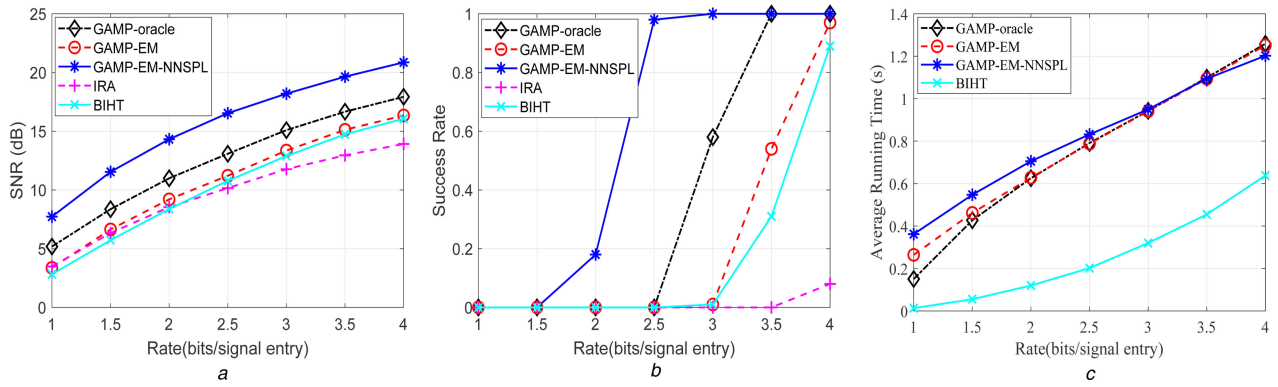

Fig. 3 *Noiseless scenario. Non-adaptive algorithms are utilised to reconstruct the signal*
(a) Average reconstruction SNR versus m/n ***(b)*** Success rate versus m/n ***(c)*** Average running time versus m/n

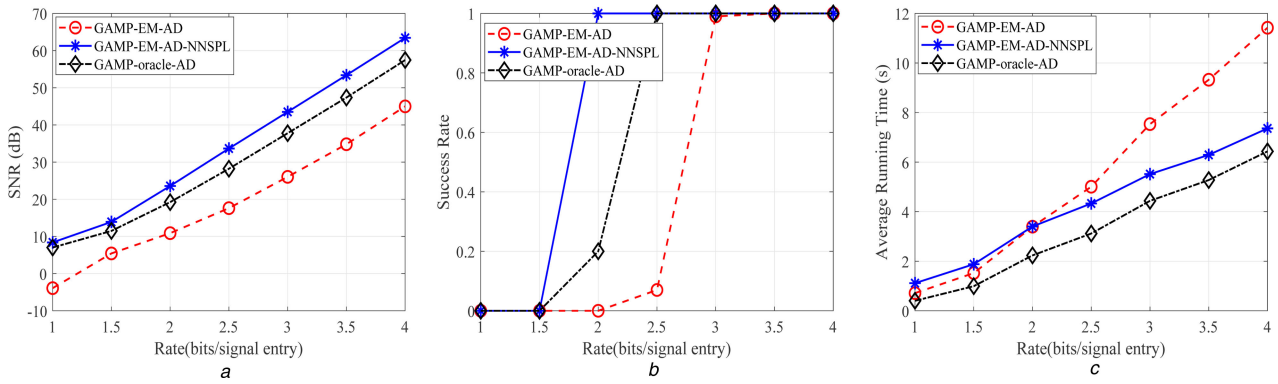

Fig. 4 *Noiseless scenario. Adaptive algorithms are utilised to reconstruct the signal*
(a) Average reconstruction SNR versus m/n ***(b)*** Success rate versus m/n ***(c)*** Average running time versus m/n

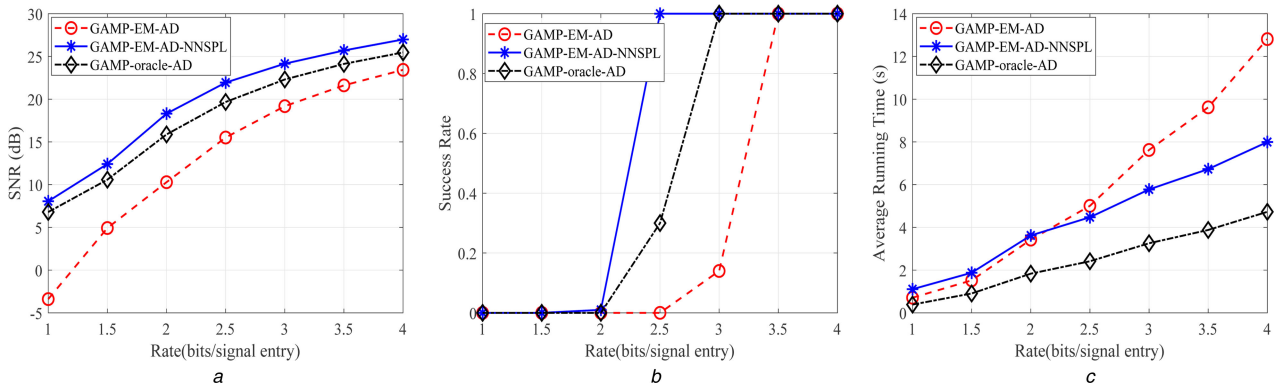

Fig. 5 *Noisy scenario. Adaptive algorithms are utilised to reconstruct the signal*
(a) Average reconstruction SNR versus m/n ***(b)*** Success rate versus m/n ***(c)*** Average running time versus m/n

For the last simulation in the subsection, we study the effectiveness of the three adaptive GAMP algorithms against the sparsity π_i of signals in the absence of noise. Note that here we set $m = 3n$, $S = 20$, $\Delta = 0$ and initialise $\pi_i^0 = 0.7\pi_i$. From Figs. 6 and 7, it can be seen that GAMP-EM-AD-NNSPL works best when $\pi_i > 0.15$ and the performance of the three algorithms decreases as π_i increases.

4.2 Real data set

For the real data set, we conduct experiments of image reconstruction from one-bit measurements. We use two standard 8-bit greyscale test images *Cameraman* and *Barbara* whose sizes are 128×128, as the image used in [25]. The measurement matrix $\boldsymbol{A}$ is generated from i.i.d. normal distribution, and the one-bit measurements are obtained via $\boldsymbol{Y} = \text{sign}(\boldsymbol{AX} + \boldsymbol{\Gamma})$, where $\boldsymbol{X} \in \mathbb{R}^{n\times n}$, $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and $n = 128$. We set $m = 3n$, i.e. the number of one-bit measurements per image pixel is 3. Other parameters are set as follows: the initial number of measurements is $m_0 = n$, the block size is $B = 15$, the thresholds are $\boldsymbol{\Gamma}^{m_0} = \boldsymbol{0}_{m_0\times n}$ and the maximal number of iterations is $T_{\max} = 500$. The small amplitudes of the Haar-wavelet coefficients of the image are set to zero, and the percentages of non-zero Haar-wavelet coefficients are 0.3156 and 0.4061, separately. For the oracle GAMP, the mean and variance of the active coefficients are estimated based on the thresholded coefficients. GAMP is applied with assumed Bernoulli–Gaussian prior on the Haar-wavelet coefficients. The peak SNR is also evaluated for the three methods. From Figs. 8 and 9, it can be seen that GAMP-EM-AD-NNSPL improves the reconstruction significantly compared to other methods. Table 1 shows the average running time of the three algorithms. It demonstrates that the running time of GAMP-EM-AD-NNSPL is close to that of GAMP-EM-AD.

5 Conclusion

We have presented a method to recover structured sparse signals in the one-bit adaptive quantisation model when the signal distribution and the sparsity pattern are both unknown. Although

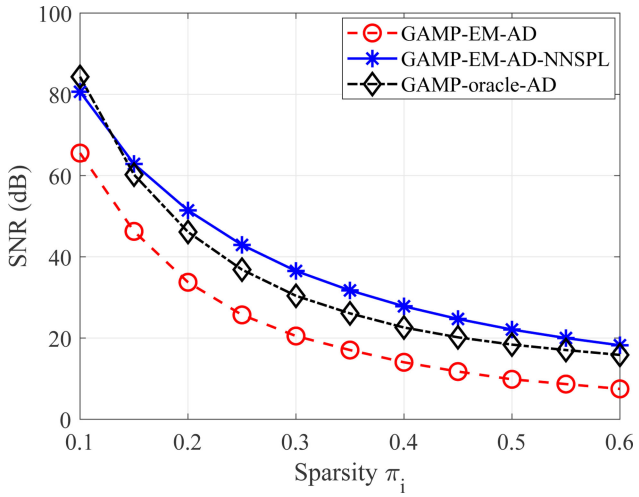

Fig. 6 *Noiseless scenario. The average reconstruction SNR is plotted against the sparsity of signals for three algorithms*

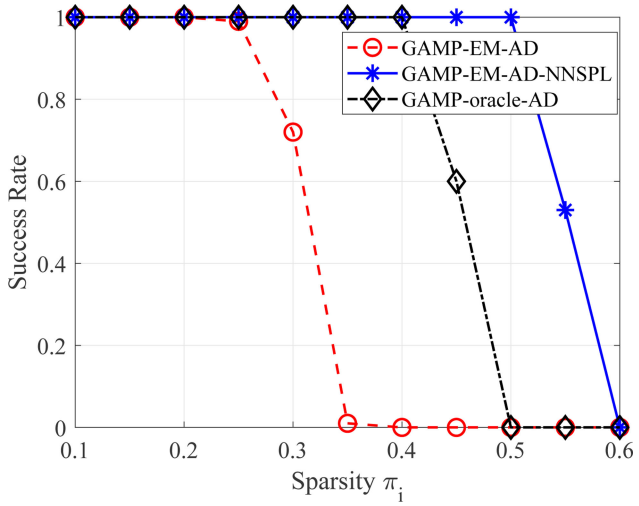

Fig. 7 *Noiseless scenario. The success rate is plotted against the sparsity of signals for three algorithms*

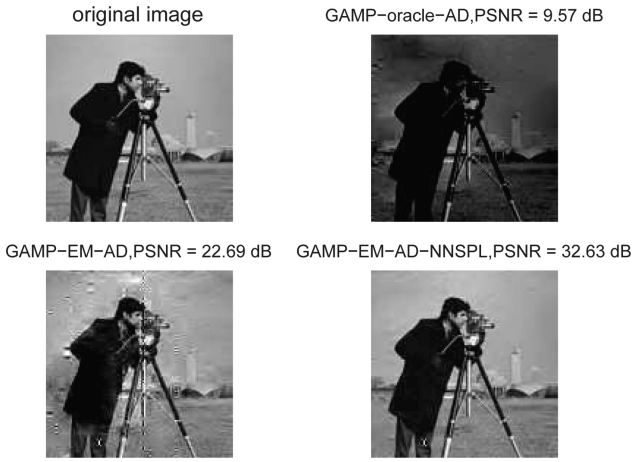

Fig. 8 *Recovering of the image Cameraman from one-bit measurements using three different methods*

Table 1 Running time (seconds) of the three algorithms used to recover the two images (averaged over 50 trials)

Image\algorithm	GAMP-oracle-AD	GAMP-EM-AD	GAMP-EM-AD-NNSPL
Cameraman	5.9592	6.6599	6.6635
Barbara	2.8559	7.0913	7.0444

Fig. 9 *Recovering of the image Barbara from one-bit measurements using three different methods*

the computational complexity of GAMP-EM-AD-NNSPL is a bit higher compared to that of GAMP-EM-AD in a single iteration, its whole running time is sometimes shorter because the NNSPL method accelerates the convergence of the GAMP-EM-AD algorithm. In addition, substantial numerical experiments have been conducted and the effectiveness of GAMP-EM-AD-NNSPL compared with GAMP-EM-AD in recovering structured signals is demonstrated.

6 Acknowledgments

This work was supported by the National Natural Science Foundation of China under grant nos. NSFC 61674128, 61731019 and Zhejiang Provincial Natural Science Foundation of China under grant no. LQ18F010001.

7 References

[1] Donoho, D.L.: 'Compressed sensing', *IEEE Trans. Inf. Theory*, 2006, **52**, (4), pp. 1289–1306
[2] Lustig, M., Donoho, D.L., Santos, J.M., *et al.*: 'Compressed sensing MRI', *IEEE Signal Process. Mag.*, 2008, **25**, (2), pp. 72–82
[3] Huang, G., Jiang, H., Matthews, K., *et al.*: 'Lensless imaging by compressive sensing'. 2013 IEEE Int. Conf. on Image Processing, 2013, pp. 2101–2105
[4] Firooz, M.H., Roy, S.: 'Network tomography via compressed sensing'. 2010 IEEE Global Telecommunications Conf. GLOBECOM, 2010, pp. 1–5
[5] Candès, E., Tao, T.: 'Decoding by linear programming', *IEEE Trans. Inf. Theory*, 2005, **51**, (12), pp. 4203–4215
[6] Wang, X., Gu, Y., Chen, L.: 'Proof of convergence and performance analysis for sparse recovery via zero-point attracting projection', *IEEE Trans. Signal Process.*, 2012, **60**, (8), pp. 4081–4093
[7] Tropp, J., Gilbert, A.: 'Signal recovery from random measurements via orthogonal matching pursuit', *IEEE Trans. Inf. Theory*, 2007, **53**, (12), pp. 4655–4666
[8] Zhu, J., Lin, X.: 'Sparse estimation from sign measurements with general sensing matrix perturbation'. Proc. Conf. on Digital Signal Processing, Hongkong, August 2014, pp. 48–53
[9] Zymnis, A., Boyd, S., Candès, E.: 'Compressed sensing with quantized measurements', *IEEE Signal Process. Lett.*, 2010, **17**, (2), pp. 149–152
[10] Ribeiro, A., Giannakis, G.B.: 'Bandwidth-constrained distributed estimation for wireless sensor networks-part I: Gaussian case', *IEEE Trans. Signal Process.*, 2006, **54**, (3), pp. 1131–1143
[11] Zhu, J., Lin, X., Blum, R.S., *et al.*: 'Parameter estimation from quantized observations in multiplicative noise environments', *IEEE Trans. Signal Process.*, 2015, **63**, (15), pp. 4037–4050
[12] Zhu, J., Wang, X., Lin, X., *et al.*: 'Maximum likelihood estimation from sign measurements with sensing matrix perturbation', *IEEE Trans. Signal Process.*, 2014, **62**, (15), pp. 3741–3753
[13] Boufounos, P.T., Baraniuk, R.G.: '1-bit compressive sensing'. Proc. Conf. on Information Sciences and Systems, Princeton, NJ, March 2008, pp. 16–21
[14] Plan, Y., Vershynin, R.: 'One-bit compressed sensing by linear programming', *Commun. Pure Appl. Math.*, 2011, **66**, (8), pp. 1275–1297
[15] Boufounos, P.T.: 'Greedy sparse signal reconstruction from sign measurements'. Proc. of Asilomar Conf. on Signals, Systems and Computers, Pacific Grove, CA, November 2009, pp. 1305–1309
[16] Donoho, D.L., Maleki, A., Montanari, A.: 'Message passing algorithms for compressed sensing'. *Proc. Natl. Acad. Sci.*, 2009, **106**, (45), pp. 18914–18919
[17] Donoho, D.L., Maleki, A., Montanari, A.: 'Message passing algorithms for compressed sensing: I. Motivation and construction'. IEEE Information Theory Workshop (ITW), 2010, pp. 1–5

[18] Montanari, A.: 'Graphical models concepts in compressed sensing', in arXiv preprint arXiv:1011.4328, 2010
[19] Rangan, S.: 'Generalized approximate message passing for estimation with random linear mixing', https://arxiv.org/pdf/1010.5141v2.pdf, accessed 2012
[20] Wen, C., Wang, C., Jin, S., *et al.*: 'Bayes-optimal joint channel-and-data estimation for massive MIMO with low-precision ADCs', *IEEE Trans. Signal Process.*, 2016, **64**, (10), pp. 2541–2556
[21] Kamilov, U. S., Rangan, S., Fletcher, A.K., *et al.*: 'Approximate message passing with consistent parameter estimation and applications to sparse learning', *IEEE Trans. Inf. Theory*, 2014, **60**, (5), pp. 2969–2985
[22] Mo, J., Schniter, P., Heath, R.: 'Channel estimation in broadband millimeter wave MIMO systems with Few-Bit ADCs', https://arxiv.org/pdf/1610.02735.pdf
[23] Vila, J., Schniter, P.: 'Expectation-maximization Gaussian-mixture approximate message passing', *IEEE Trans. Signal Process.*, 2013, **61**, (19), pp. 4658–4672
[24] Papadopoulos, H.C., Wornell, G.W., Oppenheim, A.V.: 'Sequential signal encoding from noisy measurements using quantizers with dynamic bias control', *IEEE Trans. Inform. Theory*, 2001, **47**, (3), pp. 978–1002
[25] Kamilov, U.S., Bourquard, A., Amini, A., *et al.*: 'One-Bit measurements with adaptive thresholds', *IEEE Signal Process. Lett.*, 2012, **19**, (10), pp. 607–610
[26] Kamilov, U.S., Goyal, V.K., Rangan, S.: 'Message-passing de-quantization with applications to compressed sensing', *IEEE Trans. Signal Process.*, 2012, **60**, (12), pp. 6270–6281
[27] Schniter, P.: 'Turbo reconstruction of structured sparse signals'. Proc. Conf. on Information Sciences and Systems, Princeton, NJ, March 2010
[28] Baraniuk, R.G., Cevher, V., Duarte, M.F., *et al.*: 'Model-based compressive sensing', *IEEE Trans. Inf. Theory*, 2010, **56**, (4), pp. 1982–2001
[29] Meng, X., Wu, S., Kuang, L., *et al.*: 'Approximate message passing with nearest neighbor sparsity pattern learning', https://arxiv.org/pdf/1601.00543.pdf, accessed 2016
[30] Lin, X., Wu, S., Kuang, L., *et al.*: 'Estimation of sparse massive MIMO-OFDM channels with approximately common support', *IEEE Commun. Lett.*, 2017, **21**, (5), pp. 1179–1182
[31] Fang, J., Shen, Y., Li, H., *et al.*: 'Sparse signal recovery from one-bit quantized data: An iterative reweighted algorithm', *Elsevier-Signal Processing*, 2014, **102**, pp. 201–206
[32] Blumensath, T., Davies, M.: 'Iterative hard thresholding for compressive sensing', *Appl. Comput. Harmon. Anal.*, 2009, **27**, (3), pp. 265–274
[33] Jacques, L., Laska, J., Boufounos, P., *et al.*: 'Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors', *IEEE Trans. Inf. Theory*, 2013, **59**, (4), pp. 2082–2102
[34] Frey, B.J., MacKay, D.J.C.: 'A revolution: belief propagation in graphs with cycles'. Proc. Neural Inform. Process. Syst. Conf., Denver, CO, 1997, pp. 479–485
[35] Kschischang, F.R., Frey, B.J., Loeliger, H.A.: 'Factor graphs and the sum-product algorithm', *IEEE Trans. Inf. Theory*, 2001, **47**, pp. 498–519
[36] Parker, J.T.: '*Approximate message passing algorithm for generalized bilinear inference*' (The Ohio State University Press, Ohio state, USA, 2014, 1st edn.), pp. 57–83
[37] Ziniel, J., Schniter, P., Sederberg, P.: 'Binary linear classification and feature selection via generalized approximate message passing'. Conf. on Information Sciences and Systems, 2014, pp. 1–6
[38] Fang, J., Shen, Y., Li, H., *et al.*: 'Pattern-coupled sparse Bayesian learning for recovery of block-sparse signals', *IEEE Trans. Signal Process.*, 2015, **63**, (2), pp. 360–372