GAMP: A Complete Derivation and its Application to 1-bit Measuring of AWGN Output

Jianyi Yang
Beijing University of Posts and Telecommunications
Email: yangjianyi@bupt.edu.cn
Nov. 2017

Generalized Approximate Message Passing (GAMP) is an effecient algorithm to solve the compressed sensing problem. Compared with original Approximate Message Passing (AMP), GAMP allows non-linear processing. Based on the work in [1], this article gives a detailed interpretation of the derivation of Sum-product GAMP and discusses the output function in the case of 1-bit measuring of additive white Gaussian noise (AWGN) output. The performance of GAMP is compared with that of Binary Iterative Hard Threshold (BIHT) in [2] by simulation.

I. Model of Compressed Sensing

The problem of compressed sensing is expressed as

$$\min \|x\|_0 \text{, subject to } \boldsymbol{z} = \mathrm{f}\left(\boldsymbol{y}\right) = \mathrm{f}\left(\boldsymbol{A}\boldsymbol{x}\right), \tag{1}$$

where $\boldsymbol{x} \in \mathbb{C}^{N\times 1}$ is the sparse vector, $\boldsymbol{A} \in \mathbb{C}^{M\times N}$ is the linear mixing matrix, $\mathrm{f}\left(\bullet\right)$ is a measuring function, which can be linear or non-linear. The linear mixing matrix $\boldsymbol{A}$ and measured vector $\boldsymbol{z}$ are known by us. We also know that the prior probability density of each element of the sparse vector $x_i\left(i=1,\cdots N\right)$ is $p_{X|Q}\left(x_i \mid q_i\right)$, and the output probability density is $p_{Z|Y}\left(z_j \mid y_j\right)$. The sparse vector $\boldsymbol{x}$ and the output vector $\boldsymbol{y}$ are unknown.

II. Factor Graph Representation

It is known to all that Sum-product AMP is based on the method of minimized mean square error (MMSE), which means the estimation of sparse vector is

$$\hat{\boldsymbol{x}} = \mathrm{E}_{p(\boldsymbol{x}|\boldsymbol{z})}\left[\boldsymbol{x} \mid \boldsymbol{z}\right] = \int \boldsymbol{x} p\left(\boldsymbol{x} \mid \boldsymbol{z}\right), \tag{2}$$

where the conditional probability in (2) is

$$p\left(\boldsymbol{x} \mid \boldsymbol{z}\right) = \frac{1}{Z}\prod_{j=1}^{M} p\left(\mathrm{z}_j \mid \boldsymbol{x}\right)\prod_{i=1}^{N} p\left(x_i \mid q_i\right), \tag{3}$$

where Z is a normalization variable which is uncorrelated with $\boldsymbol{x}$.

Now we transform this problem into multiple single-scale problems, namely

$$\hat{x}_i = \mathrm{E}_{p(x_i|\boldsymbol{z})}[x_i \mid \boldsymbol{z}] = \int x_i p(x_i \mid \boldsymbol{z}) \mathrm{d}x_i, i = 1, 2, \cdots N, \tag{4}$$

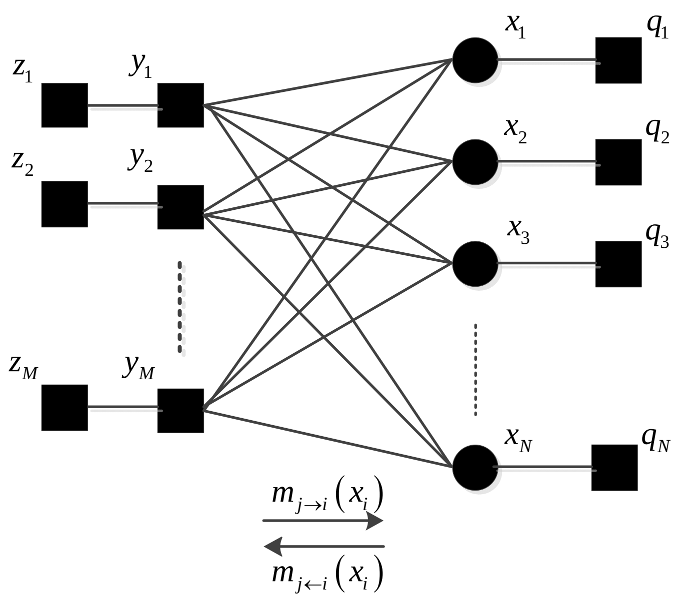

Fig. 1 The Factor Graph Representation of Compressed Sensing Model

where $p(x_i \mid \boldsymbol{z}) = \int_{\sim(x_i)} p(\boldsymbol{x} \mid \boldsymbol{z})$ is marginal conditional probability. The complexity of computing these marginal probabilities directly is very high, but the sum-product algorithm in factor graphs with cycles provides a possible method to solve this problem.

Substitute (3) into marginal probability, we can get

$$\begin{aligned} p(x_i \mid \boldsymbol{z}) &= \int_{\sim(x_i)} p(\boldsymbol{x} \mid \boldsymbol{z}) \\ &= \frac{1}{Z} \int_{\sim(x_i)} \prod_{j=1}^{M} p(\mathrm{z}_j \mid \boldsymbol{x}) \prod_{i=1}^{N} p(x_i \mid q_i) \\ &= \frac{1}{Z} \int_{\sim(x_i)} \prod_{j=1}^{M} p(z_j \mid y_j) \mathcal{X}_{\{y_j=(\boldsymbol{Ax})_j\}}(\boldsymbol{x}) \prod_{i=1}^{N} p(x_i \mid q_i) \end{aligned}, \tag{5}$$

where $\mathcal{X}_{\{y_j=(\boldsymbol{Ax})_j\}}(\boldsymbol{x})$ is an indicative function, which equals 1 if $\boldsymbol{x} \in \{y_j=(\boldsymbol{Ax})_j\}$ and 0 if $\boldsymbol{x} \notin \{y_j=(\boldsymbol{Ax})_j\}$ 。

The factor graph representation of (5) is shown in Fig.1. The sum-product update rules are[3]

$$m_{j\leftarrow i}^{t+1}(x_i) \cong p_{X|Q}(x_i \mid q_i) \prod_{l\neq j} m_{l\to i}^{t}(x_i) \tag{6}$$

$$m_{j\to i}^{t}(x_i) \cong \int_{\sim(x_i)} p_{Z|Y}(z_j \mid y_j) \prod_{l\neq i} m_{j\leftarrow l}^{t}(x_l) \tag{7}$$

where $m_{j\leftarrow i}^{t+1}(x_i)$ and $m_{j\to i}^{t}(x_i)$ are the message in tth iteration, and $\cong$ denotes identity between probability distributions up to a normalization constant.

III. Approximation of Message Passing

Before the approximation, we first give a lemma that is useful in our following derivation.

Lemma 1 Assume that the conditional probability of a random variable x is $p(x|r,\tau)=\frac{1}{Z(\tau,\ r)}q(x)\exp\left(-\frac{1}{2\tau}(x-r)^2\right)$, where $Z(\tau,\ r)$ is a normalization variable, and $Z(\tau,\ r)=\int q(x)\exp\left(-\frac{1}{2\tau}(x-r)^2\right)\mathrm{d}x$. If $\hat{x}$ and τ^x are respectively the expectation and the variance of x under $p(x|r,\tau)$, then $\frac{\partial}{\partial r}\hat{x}=\frac{\tau^x}{\tau}$.

Proof Calculate the derivative of $\hat{x}$, we get

$$\begin{aligned}\frac{\partial}{\partial r}\hat{x}=&\frac{1}{Z^2(\tau,\ r)}\int p_{X|Q}(x|q)\exp\left(-\frac{1}{2\tau}(x-r)^2\right)\frac{x-r}{\tau}\mathrm{d}x\bullet\hat{x}Z(\tau,\ r)\\&+\frac{1}{Z(\tau,\ r)}\int xp_{X|Q}(x|q)\exp\left(-\frac{1}{2\tau}(x-r)^2\right)\frac{x-r}{\tau}\mathrm{d}x\\=&-\frac{\hat{x}^2}{\tau}+\frac{r\hat{x}}{\tau}+\frac{\mathrm{E}\left[x^2\right]}{\tau}-\frac{r\hat{x}}{\tau}=\frac{\tau^x}{\tau}\end{aligned}$$

A. *Approximation of Update Rule (7)*

The aim of this part is to derive the message from factor nodes to variable nodes with the knowledge of the message from variable nodes to factor nodes. Assume that $\hat{x}_{j\leftarrow i}^t$ and $\hat{\tau}_{j\leftarrow i}^t$ are expectation and variance of x_i under the massage form the variable node *i* to the factor node *j*, namely $m_{j\leftarrow i}^t(x_i)$. According to Central Limit Theorem, with the knowledge of x_i, y_j follows Gaussian distribution. Here, assume that for all i, $\hat{\tau}_{j\leftarrow i}^t$ approximately equals to $\hat{\tau}_i^t$, so we have

$$p\left(y_j\mid x_i^t\right)=\mathcal{N}\left(y_j;\hat{p}_{j\leftarrow i}^t+a_{j,i}x_i^t,\tau_i^{t,p}\right)\quad,\tag{8}$$

where $\mathcal{N}(y;p,\tau)=\frac{1}{\sqrt{2\pi\tau}}\exp\left(-\frac{1}{2\tau}(x-p)^2\right)$ is Gaussian probability density, and

$$\hat{p}_{j\leftarrow i}^{t}=\sum_{l\neq i}a_{j,l}\hat{x}_{j\leftarrow l}^{t} \tag{9}$$

$$\tau_{j\leftarrow i}^{t,p}=\sum_{l\neq i}\left|a_{j,l}\right|^{2}\hat{\tau}_{l}^{t} \quad . \tag{10}$$

In (10), we ignored a term of order $O\left(\left|a_{j,i}\right|^{2}\right)$.

According to (7), we have

$$\begin{aligned} m_{j\to i}^{t}\left(x_{i}\right) &\cong \int_{\sim\left(x_{i}\right)} p_{Z|Y}\left(z_{j}\mid y_{j}\right)\prod_{l\neq i}m_{j\leftarrow l}^{t}\left(x_{l}\right)\mathrm{d}\,\boldsymbol{x} \\ &\overset{(a)}{=}\int p_{Z|Y}\left(z_{j}\mid y_{j}\right)p\left(y_{j}\mid x_{i}\right)\mathrm{d}\,y_{j} \\ &\overset{(b)}{=}\int p_{Z|Y}\left(z_{j}\mid y_{j}\right)\mathcal{N}\left(y_{j};\hat{p}_{j\leftarrow i}^{t}+a_{j,i}x_{i}^{t},\tau_{i}^{t,p}\right)\mathrm{d}\,y_{j} \end{aligned} \tag{11}$$

where (a) converts the multidimensional integral with respect to $\boldsymbol{x}$ into the single-dimensional integral with respect to y_j, and (b) comes from substituting (8) into (11).

Now let me give the interpretation of (11). If $y\sim\mathcal{N}\left(p,\tau\right)$, then it is obvious that the probability of y under z, p and τ is

$$\begin{aligned} p\left(y\mid z,p,\tau\right) &= \frac{1}{Z}p\left(z\mid y\right)p\left(y\mid p,\tau\right) \\ &= \frac{1}{Z}p\left(z\mid y\right)\mathcal{N}\left(y;p,\tau\right) \end{aligned} \tag{12}$$

where Z is a variable without depending on y. Thus, (11) can be rewritten as

$$\begin{aligned} m_{j\to i}^{t}\left(x_{i}\right) &= \int p\left(y_{j}\mid z_{j},\hat{p}_{j\leftarrow i}^{t}+a_{j,i}x_{i}^{t},\tau_{i}^{t,p}\right)\mathrm{d}\,y_{j} \\ &= \mathrm{E}\left[y_{j}\mid z_{j},\hat{p}_{j\leftarrow i}^{t}+a_{j,i}x_{i}^{t},\tau_{i}^{t,p}\right] \end{aligned} \quad , \tag{13}$$

If we define the function $\mathrm{H}\left(p,z,\tau\right)=\mathrm{E}\left[y\mid z,p,\tau\right]$, then we have

$$m_{j\to i}^{t}\left(x_{i}\right)=\mathrm{H}\left(\hat{p}_{j\leftarrow i}^{t}+a_{j,i}x_{i},z_{j},\tau_{j\leftarrow i}^{t,p}\right), \tag{14}$$

If $\hat{p}_{j}^{t}$ and $\tau_{j}^{t,p}$ are defined as

$$\hat{p}_{j}^{t}=\sum_{l\neq i}a_{j,l}\hat{x}_{j\leftarrow l}^{t}+a_{j,i}\hat{x}_{i}^{t} \quad , \tag{15}$$

$$\tau_{j}^{t,p}=\sum_{l}\left|a_{j,l}\right|^{2}\hat{\tau}_{l}^{t} \quad , \tag{16}$$

then it follows from (9) and (10) that

$$\hat{p}_{j\leftarrow i}^{t}=\hat{p}_{j}^{t}\text{-}a_{j,i}\hat{x}_{i}^{t} \tag{17}$$

$$\tau_{j\leftarrow i}^{t,p}=\tau_{j}^{t,p}-\left|a_{j,i}\right|^{2}\hat{\tau}_{i}^{t}\approx\tau_{j}^{t,p} \tag{18}$$

Neglect a term of order $O\left(\left|a_{j,i}\right|^2\right)$ in (16). Now (11) can be rewritten as

$$m_{j\to i}^t\left(x_i\right) = \mathrm{H}\left(\hat{p}_j^t + a_{j,i}\left(x_i - \hat{x}_i^t\right), z_j, \tau_j^{t,p}\right) \tag{19}$$

Next, we approximate $m_{j\to i}^t\left(x_i\right)$ by a 2nd-order Taylor expansion:

$$\begin{aligned} m_{j\to i}^t\left(x_i\right) &= \exp\left(\log\left(\mathrm{H}\left(\hat{p}_j^t + a_{j,i}\left(x_i - \hat{x}_i^t\right), z_j, \tau_j^{t,p}\right)\right)\right) \\ &\overset{\text{(a)}}{\approx} \exp\left(s_j^t a_{j,i}\left(x_i - \hat{x}_i^t\right) - \frac{\tau_j^{s,t}}{2} a_{j,i}^2\left(x_i - \hat{x}_i^t\right)^2 + \mathrm{C}\right) \\ &= \exp\left(\left(s_j^t a_{j,i} + \tau_j^{s,t} a_{j,i}^2 \hat{x}_i^t\right) x_i - \frac{\tau_j^{s,t}}{2} a_{j,i}^2 x_i^2 + \mathrm{C}\right) \end{aligned} \tag{20}$$

where (a) comes from 2-order Taylor expansion of $\log\left(m_{j\to i}^t\left(x_i\right)\right)$ at $\hat{x}_i^t$ and C does not depend on x_i.

From (20), we find that the message from factor nodes to variable nodes depends on s_j^t and $\tau_j^{s,t}$. To calculate the two important intermediate variables in the algorithm, the output function is defined as

$$g_{\text{out}}\left(\hat{p}, z, \tau^p\right) \equiv \frac{\partial}{\partial \hat{p}} \log \mathrm{H}\left(\hat{p}, z, \tau^p\right) \ . \tag{21}$$

then s_j^t and $\tau_j^{s,t}$ can be expressed as

$$s_j^t = g_{\text{out}}\left(\hat{p}_j^t, z_j, \tau_j^{t,p}\right) \tag{22}$$

$$\tau_j^{s,t} = -\frac{\partial}{\partial \hat{p}} g_{\text{out}}\left(\hat{p}_j^t, z_j, \tau_j^{t,p}\right) \tag{23}$$

The output function can be further simplified as

$$\begin{aligned} g_{\text{out}}\left(\hat{p}, z, \tau^p\right) &= \frac{\partial}{\partial \hat{p}} \log \mathrm{H}\left(\hat{p}, z, \tau^p\right) \\ &= \frac{1}{\mathrm{H}\left(\hat{p}, z, \tau^p\right)} \int \frac{y - \hat{p}}{\tau^p} p_{Z|Y}\left(z \mid y\right) \frac{1}{\sqrt{2\pi\tau^p}} \exp\left(\frac{1}{2\tau^p}\left(y - \hat{p}\right)^2\right) \mathrm{d}\, y \\ &= \frac{\mathrm{E}\left[y \mid \hat{p}, z, \tau^p\right] - \hat{p}}{\tau^p} \end{aligned}$$

And according to Lemma 1, we can get：

$$-\frac{\partial}{\partial\hat{p}} g_{\text{out}}\left(\hat{p}, z, \tau^{p}\right)=\frac{\tau^{p}-\operatorname{var}\left(y \mid \hat{p}, z, \tau^{p}\right)}{\left(\tau^{p}\right)^{2}} 。$$

B. ***Approximation of Update Rule (6)***

In this part, on the basis of the results of Part A, we continue to derive the message $m_{j\leftarrow i}^{t+1}\left(x_i\right)$ and the message $m_i^{t+1}\left(x_i\right)$ as well as respective expectation under them.

Substituting (20) into (6), the message from the variable node i to the factor node j is:

$$\begin{aligned} m_{j\leftarrow i}^{t+1}\left(x_i\right) &\cong p_{X|Q}\left(x_i \mid q_i\right)\prod_{l\neq j} m_{l\to i}^{t}\left(x_i\right) \\ &= p_{X|Q}\left(x_i \mid q_i\right)\exp\left(\sum_{l\neq j}\left(s_l^t a_{l,i}+\tau_l^{s,t} a_{l,i}^2 \hat{x}_i^t\right)x_i - \sum_{l\neq j}\frac{\tau_l^{s,t}}{2} a_{l,i}^2 x_i^2 + \mathrm{C}\right) \\ &= \frac{1}{Z\left(\tau_{j\leftarrow i}^{r,t},\ \hat{r}_{j\leftarrow i}^{t}\right)} p_{X|Q}\left(x_i \mid q_i\right)\exp\left(-\frac{1}{2\tau_{j\leftarrow i}^{r,t}}\left(x_i-\hat{r}_{j\leftarrow i}^{t}\right)^2\right) \end{aligned} \tag{24}$$

where $Z\left(\tau_{j\leftarrow i}^{r,t},\ \hat{r}_{j\leftarrow i}^{t}\right)=\int p_{X|Q}\left(x_i \mid q_i\right)\exp\left(-\frac{1}{2\tau_{j\leftarrow i}^{r,t}}\left(x_i-\hat{r}_{j\leftarrow i}^{t}\right)^2\right)\mathrm{d}x_i$ is a normalization variable without depending on x_i, and

$$\tau_{j\leftarrow i}^{r,t}=1/\sum_{l\neq j} a_{l,i}^2\tau_l^{s,t} \tag{25}$$

$$\begin{aligned} \hat{r}_{j\leftarrow i}^{t} &= \tau_{j\leftarrow i}^{r,t}\sum_{l\neq j}\left(s_l^t a_{l,i}+\tau_l^{s,t} a_{l,i}^2 \hat{x}_i^t\right) \\ &= \hat{x}_i^t+\tau_{j\leftarrow i}^{r,t}\sum_{l\neq j} s_l^t a_{l,i} \end{aligned} \tag{26}$$

Since the aim of one loop is to estimate x_i^{t+1}, it is necessary to derive the message at node i, namely $m_i^{t+1}\left(x_i\right)$. Then the estimation of x_i^{t+1} is the expectation under the probability measurement $m_i^{t+1}\left(x_i\right)$.

According to the factor graph theory in [3] and similar as the derivation in (24), the message at node i is

$$\begin{aligned} m_i^{t+1}\left(x_i\right) &\cong p_{X|Q}\left(x_i \mid q_i\right)\prod_{l} m_{l\to i}^{t}\left(x_i\right) \\ &= \frac{1}{Z\left(\tau_i^{r,t},\ \hat{r}_i^{t}\right)} p_{X|Q}\left(x_i \mid q_i\right)\exp\left(-\frac{1}{2\tau_i^{r,t}}\left(x_i-\hat{r}_i^{t}\right)^2\right) \end{aligned} \tag{27}$$

where

$$\tau_i^{r,t}=1/\left(\sum_{l} a_{l,i}^2\tau_l^{s,t}\right) \tag{28}$$

$$\hat{r}_i^t = \hat{x}_i^t + \tau_i^{r,t} \sum_l s_l^t a_{l,i} \quad . \tag{29}$$

So, the estimation of x_i^{t+1} is

$$\begin{aligned} \hat{x}_i^{t+1} &= \mathrm{E}_{m_i^{t+1}}\left[x_i \mid \hat{r}_i^t, q_i, \tau_i^{r,t} \right] \\ &= \int x_i \frac{1}{Z\left(\tau_i^{r,t},\ \hat{r}_i^t\right)} p_{X|Q}\left(x_i \mid q_i\right) \exp\left(-\frac{1}{2\tau_i^{r,t}} \left(x_i - \hat{r}_i^t\right)^2 \right) \mathrm{d}\, x_i \end{aligned} \tag{30}$$

To express it more explicitly, the input function is defined as

$$g_{\text{in}}\left(r, q, \tau\right) = \int x \frac{1}{Z\left(\tau, r\right)} p_{X|Q}\left(x \mid q\right) \exp\left(-\frac{1}{2\tau} \left(x - \hat{r}\right)^2 \right) \mathrm{d}\, x \,. \tag{31}$$

So (30) can be rewritten as

$$\hat{x}_i^{t+1} = g_{\text{in}}\left(\hat{r}_i^t, q_i, \tau_i^{r,t}\right). \tag{32}$$

Similarly, the expectation under the message $m_{j\leftarrow i}^{t+1}\left(x_i\right)$ can be expressed as

$$\hat{x}_{j\leftarrow i}^{t+1} = g_{\text{in}}\left(\hat{r}_{j\leftarrow i}^t, q_i, \tau_{j\leftarrow i}^{r,t}\right). \tag{33}$$

Note that as shown in (15) of Part A, in the next loop, the important variable $\hat{p}_j^{t+1}$ is dependent on $\hat{x}_{j\leftarrow i}^{t+1}$.

C. *Further Simplification*

Although the iteration can be carried out after the derivation in Part A and Part B, the algorithm is still complex for the existence of variables with respect to edges in factor graph. So it is necessary to replace them with variables with respect to nodes. In order to achieve that, we need to find the relationship between $\hat{x}_i^{t+1}$ and $\hat{x}_{j\leftarrow i}^{t+1}$.

First of all, we make the approximation of $\tau_{j\leftarrow i}^{r,t}$ and $\hat{r}_{j\leftarrow i}^t$ by neglecting a term of order $O\left(a_{j,i}^2\right)$. That is

$$\tau_{j\leftarrow i}^{r,t} = 1/\left(\sum_l a_{l,i}^2 \tau_l^{s,t} - a_{j,i}^2 \tau_j^{s,t} \right) \approx 1/\left(\sum_l a_{l,i}^2 \tau_l^{s,t} \right) = \tau_i^{r,t} \tag{34}$$

$$\hat{r}_{j\leftarrow i}^t \approx \hat{x}_i^t + \tau_i^{r,t} \sum_{l\neq j} s_l^t a_{l,i} = \hat{r}_i^t - \tau_i^{r,t} s_j^t a_{\mathrm{j},i} \tag{35}$$

Then, by substituting (34) and (35) into (33), we can get

$$\begin{aligned} \hat{x}_{j\leftarrow i}^{t+1} &= g_{\text{in}}\left(\hat{r}_i^t - \tau_i^{r,t} s_j^t a_{j,i}, q_i, \tau_i^{r,t}\right) \\ &\overset{\text{(a)}}{\approx} \hat{x}_i^{t+1} - s_j^t a_{j,i} D_i^{t+1} \end{aligned} \quad , \tag{36}$$

where (a) is the 1st-order Taylor serials of $\hat{x}_{j\leftarrow i}^{t+1}$ at s_j^t and $D_i^{t+1}=\tau_i^{r,t}\frac{\partial}{\partial\hat{r}_i^t}g_{\mathrm{in}}\left(\hat{r}_i^t,q_i,\tau_i^{r,t}\right)$.

According to *Lemma 1*, we have $\frac{\partial}{\partial\hat{r}_i^t}g_{\mathrm{in}}\left(\hat{r}_i^t,q_i,\tau_i^{r,t}\right)=\frac{\mathrm{var}\left(x_i^{t+1}\mid\hat{r}_i^t,q_i,\tau_i^{r,t}\right)}{\tau_i^{r,t}}=\frac{\tau_i^{t+1}}{\tau_i^{r,t}}$. Thus,

$$D_i^{t+1}=\tau_i^{r,t}\frac{\partial}{\partial\hat{r}_i^t}g_{\mathrm{in}}\left(\hat{r}_i^t,q_i,\tau_i^{r,t}\right)=\tau_i^{r,t}\frac{\tau_i^{t+1}}{\tau_i^{r,t}}=\tau_i^{t+1}. \tag{37}$$

Now the relationship between $\hat{x}_i^{t+1}$ and $\hat{x}_{j\leftarrow i}^{t+1}$ is

$$\hat{x}_{j\leftarrow i}^{t+1}\approx\hat{x}_i^{t+1}-s_j^t a_{j,i}\tau_i^{t+1} \quad . \tag{38}$$

After substituting (38) into (15) and neglecting a term of $O\left(a_{j,i}^2\right)$, we can get

$$\hat{p}_j^t\approx\sum_i\left(a_{j,i}\hat{x}_i^t-a_{j,i}^2\tau_i^t s_j^{t-1}\right)=\sum_i a_{j,i}\hat{x}_i^t-\tau_j^{p,t}s_j^{t-1} \tag{39}$$

Combining (39)(16)(22)(23)(28)(29)(32)(37), we can get GAMP algorithm in [1].

VI. GAMP with 1-bit Measuring of AWGN Output

In this section, the compressed sensing problem with 1-bit measuring of additive white Gaussian noise (AWGN) output is considered. Now the function $f(\bullet)$ in (1) has become

$$\boldsymbol{z}=\mathrm{f}\left(\boldsymbol{y}\right)=Q(\boldsymbol{y}+\boldsymbol{n}), \tag{40}$$

where $Q(x)=\mathrm{sign}\left(\mathrm{Re}(x)\right)+i\,\mathrm{sign}\left(\mathrm{Im}(x)\right)$ is the complex sign function and $\boldsymbol{n}\sim\mathcal{CN}\left(0,\sigma^2\right)$ is complex Gaussian noise with mean 0 and variance σ^2.

A. Output function of 1-bit Measuring of AWGN Output

Apparently, in order to solve the aforementioned problem, the output function of this case should be derived. For similarity, this article first discusses the real-number case and then generalizes the conclusions into complex-number case.

In this case, the output probability density is

$$p_{Z|Y}\left(z=1\mid y\right)=1-\phi\left(-\frac{y}{\sigma_n}\right) \tag{41}$$

$$p_{Z|Y}\left(z=-1\mid y\right)=\phi\left(-\frac{y}{\sigma_n}\right) \tag{42}$$

where $\phi(x)=\int_{-\infty}^{x}\frac{1}{\sqrt{2\pi}}e^{-\frac{t^2}{2}}\mathrm{d}t$ is the cumulative distribution function of the standard normal distribution. So the conditional functions in (12) become

$$p\left(y \mid z=1,\hat{p},\tau^{p}\right)=\frac{1}{Z}\left(1-\phi\left(-\frac{y}{\sigma_n}\right)\right)\exp\left(-\frac{1}{2\tau^{p}}\left(y-\hat{p}\right)^{2}\right) \tag{43}$$

$$p\left(y \mid z=-1,\hat{p},\tau^{p}\right)=\frac{1}{Z}\phi\left(-\frac{y}{\sigma_n}\right)\exp\left(-\frac{1}{2\tau^{p}}\left(y-\hat{p}\right)^{2}\right) \tag{44}$$

Further, the expectation functions and variance functions can be obtained:

$$E\left(y \mid z=-1,\hat{p},\tau^{p}\right)=\int_{-\infty}^{+\infty}y\frac{1}{Z}\phi\left(-\frac{y}{\sigma_n}\right)\exp\left(-\frac{1}{2\tau^{p}}\left(y-\hat{p}\right)^{2}\right)\mathrm{d}\,y \tag{45}$$

$$\begin{aligned}E\left(y \mid z=1,\hat{p},\tau^{p}\right)&=\int_{-\infty}^{+\infty}y\frac{1}{Z}\left(1-\phi\left(-\frac{y}{\sigma_n}\right)\right)\exp\left(-\frac{1}{2\tau^{p}}\left(y-\hat{p}\right)^{2}\right)\mathrm{d}\,y\\&=\hat{p}-\int_{-\infty}^{+\infty}y\frac{1}{Z}\phi\left(-\frac{y}{\sigma_n}\right)\exp\left(-\frac{1}{2\tau^{p}}\left(y-\hat{p}\right)^{2}\right)\mathrm{d}\,y\end{aligned} \tag{46}$$

$$\operatorname{var}\left(y \mid z=-1,\hat{p},\tau^{p}\right)=\int_{-\infty}^{+\infty}y^{2}\frac{1}{Z}\phi\left(-\frac{y}{\sigma_n}\right)\exp\left(-\frac{1}{2\tau^{p}}\left(y-\hat{p}\right)^{2}\right)\mathrm{d}\,y-E^{2}\left(y \mid z=-1,\hat{p},\tau^{p}\right) \tag{47}$$

$$\begin{aligned}\operatorname{var}\left(y \mid z=1,\hat{p},\tau^{p}\right)&=\int_{-\infty}^{+\infty}y^{2}\frac{1}{Z}\left(1-\phi\left(-\frac{y}{\sigma_n}\right)\right)\exp\left(-\frac{1}{2\tau^{p}}\left(y-\hat{p}\right)^{2}\right)\mathrm{d}\,y-E^{2}\left(y \mid z=1,\hat{p},\tau^{p}\right)\\&=\tau^{p}-\int_{-\infty}^{+\infty}y^{2}\frac{1}{Z}\phi\left(-\frac{y}{\sigma_n}\right)\exp\left(-\frac{1}{2\tau^{p}}\left(y-\hat{p}\right)^{2}\right)\mathrm{d}\,y-E^{2}\left(y \mid z=1,\hat{p},\tau^{p}\right)\end{aligned} \tag{48}$$

The simplification of (45)-(48) is difficult. I just have a rudimentary idea up to now. The abstract cumulative distribution function can be embodied as

$$\phi(x)=\int_{-\infty}^{x}\frac{1}{\sqrt{2\pi}}e^{-\frac{t^2}{2}}\mathrm{d}t\approx\frac{1}{1+e^{-ax}}\,. \tag{49}$$

where α is a constant approximately equal to 1.8.

When complex variables are the case, the expectation function and variance function are respectively

$$E\left(y \mid z,\hat{p},\tau^{p}\right)=E\left(y_r \mid z_r,\hat{p}_r,\tau^{p}\right)+iE\left(y_i \mid z_i,\hat{p}_i,\tau^{p}\right)$$

$$\operatorname{var}\left(y \mid z,\hat{p},\tau^{p}\right)=\operatorname{var}\left(y_r \mid z_r,\hat{p}_r,\tau^{p}\right)+\operatorname{var}\left(y_i \mid z_i,\hat{p}_i,\tau^{p}\right).$$

After the expectation and variance are derived, the output function and its derivative are able to be obtained via

$$g_{\text{out}}\left(\hat{p}, z, \tau^{p}\right)=\frac{\mathrm{E}\left[y \mid \hat{p}, z, \tau^{p}\right]-\hat{p}}{\tau^{p}}, \text{ and}$$

$$-\frac{\partial}{\partial \hat{p}} g_{\text{out}}\left(\hat{p}, z, \tau^{p}\right)=\frac{\tau^{p}-\operatorname{var}\left(y \mid \hat{p}, z, \tau^{p}\right)}{\left(\tau^{p}\right)^{2}}.$$

B. Performance Evaluation

In this part, the performance of GAMP is compared with that of Binary Iterative Hard Threshold (BIHT) [2] in terms of the aforementioned 1-bit compressed sensing problem. The length of the sparse vector is $N=1024$. Assuming that the input distribution is Bernoulli-Gaussian distribution with its probability density function

$$p(x)=(1-\eta)\delta(x)+\frac{\eta}{\pi\sigma_L^2}e^{-\frac{|x|}{\sigma_L^2}},$$

where η is sparsity rate, the input function is derived in [4]. $\boldsymbol{A}\in\mathbb{C}^{M\times N}$ is the linear mixing matrix with each element $a_{j,i}\sim\mathcal{N}\left(0,\frac{1}{\sqrt{M}}\right)$. The performance is evaluated by mean squared error (MSE), which is $MSE=\|\hat{\boldsymbol{x}}-\boldsymbol{x}\|_{\mathrm{F}}/\|\boldsymbol{x}\|_{\mathrm{F}}$. Monte-Carlo simulations are carried out in this part.

The MSE with respect to signal to noise (SNR) under different sparsity rates is

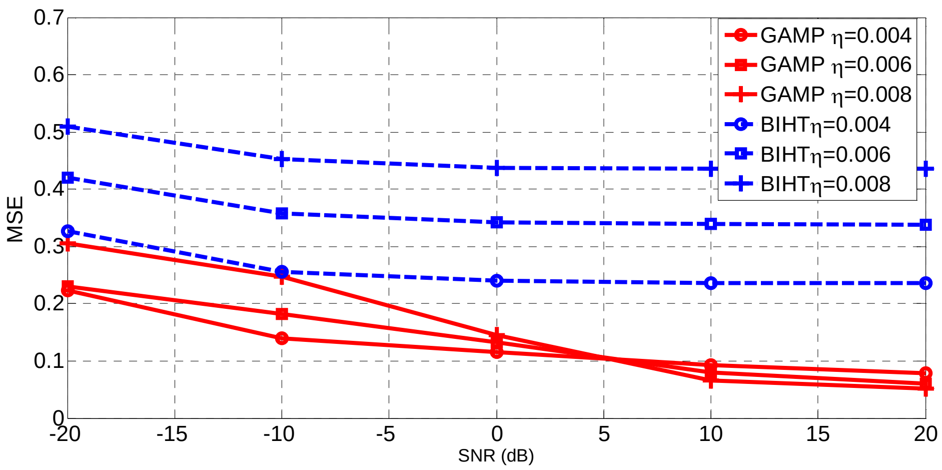

Fig. 2 MSE of GAMP and BIHT with respect to SNR under different sparstity rates.

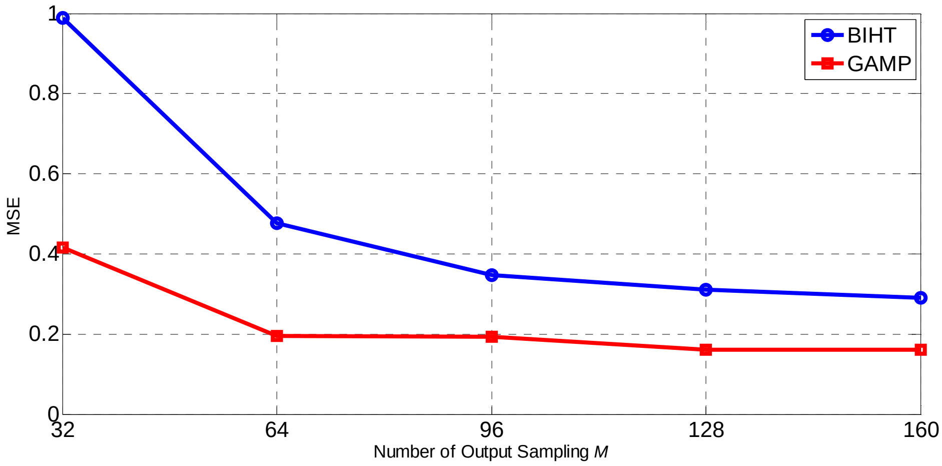

Fig. 3 MSE of GAMP and BIHT with respect to the number of output sampling

shown in Fig. 2. The number of output sampling is $M=256$. The results show that GAMP has smaller estimation error than BIHT. In addition, the performance of GAMP is robust to sparsity rate while the performance of BIHT deteriorates with increasing of sparsity rate. The devise of input function correlated with sparsity rate may account for this phenomenon.

The MSE with respect to the number of output sampling is shown in Fig. 3. The sparsity rate is $\eta=0.004$ and SNR is 20 dB. According to the simulation result, MSE of both algorithm decreases with the increasing of the number of output sampling. But for GAMP, when MSE gets relatively stable M is larger than 64 while for BIHT, M is larger than 128. Therefore, GAMP requires smaller measuring number.

Reference

[1] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," 2012. Available: http://arxiv.org/ abs/1010.5141v2.

[2]L. Jacques, J. N. Laska, P. T. Boufounos, and R. G. BaraniukRobust, "1-Bit Compressive Sensing via Binary Stable Embeddings of Sparse Vectors," IEEE Transactions on Information Theory, vol. 59, no. 4, April 2013.

[3] Frank R. Kschischang, Brendan J. Frey, and Hans-Andrea Loeliger, "Factor Graphs and the Sum-Product Algorithm," IEEE Transactions on Information Theory, vol. 47, no. 2, Feb. 2001.

[4] Xiangming Meng, Sheng Wu, Linling Kuang, Jianhua Lu, "Concise Derivation of Complex Bayesian Approximate Message Passing via Expectation Propagation," arXiv: Information Theory, 2015. Available: https://arxiv.org/pdf/1509.08658v2.pdf.