A fast approach for overcomplete sparse decomposition based on smoothed L0 norm

Authors: H. Mohimani, M. B. Zadeh, C. Jutten
Publication: IEEE Trans. on Sig. Process, Jan.2009
Speaker: Oliver

Short summary:

This paper proposes a fast algorithm for overcomplete sparse decomposition. The algorithm is derived by directly minimizing the L0 norm after smoothing. Hence, the algorithm is named as smoothed L0 (SL0) algorithm. The authors demonstrate that their algorithm is 2-3 orders of magnitude faster than the state-of-the-art interior point solvers with same (or better) accuracy.

I. Introduction

- To introduce the algorithm the authors have used the context of source component analysis (SCA). SCA is a method to achieve separation of sparse sources.

- Suppose that *m* source signals are recorded by a set of *n* sensors each of which records a combination of all sources. In linear instantaneous (noiseless) model, it is assumed that $x(t) = As(t)$ in which $s(t)$ and $x(t)$ are $m \times 1$ and $n \times 1$ vectors of source and recorded signals, respectively, and $A \in R^{n \times m}$ is a mixing matrix.

- The goal of blind source separation (BSS) is then to find $s(t)$ only by observing $x(t)$. The general BSS problem is impossible for the case $n<m$. However, if the sources are sparse then this problem can be solved (using L1 minimization).

- We have the problem of finding sparse solutions of the undetermined system of linear equations (USLE) $As=x$. To obtain the sparsest solution of $As=x$, we may search for a solution with minimal L0 norm. (Intractable problem, sensitive to noise)

- Hence, researchers consider L1 approaches such as basis pursuit (BP), LP-norm approaches such as IRLS, and greedy approaches such as matching pursuit (MP).

- In this paper, authors present an approach for solving USLE by direct minimization of the L0 norm after smoothing (approximating with smooth functions).

- Performance of the algorithm is equal to (or better than) the interior point based algorithms with 2 to 3 orders of magnitude faster.

algorithm	total time (sec)	MSE
SL0	0.227	$5.53\,e-5$
LP (ℓ_1-magic)	30.1	$2.31\,e-4$
FOCUSS	20.6	$6.45\,e-4$

II. APPROACH

- L0 norm of a vector $x \in R^m$ is a *discontinuous* function of that vector.

$$\|x\|_0 = \sum_{i=1}^{m} I_i \qquad\qquad I_i = \begin{cases} 1 & if\ x_i \neq 0 \\ 0 & if\ x_i = 0 \end{cases}$$

- The idea then is to approximate the discontinuous function with a continuous function. The continuous function has a parameter (say σ) that determines the quality of the approximation.

- For example, consider the (one-variable) family of Gaussian functions

$$f_\sigma(s) \triangleq \exp\left(\tfrac{-s^2}{2\sigma^2}\right)$$

and note that

$$\lim_{\sigma \to 0} f_\sigma(s) = \begin{cases} 1 & if\ s = 0 \\ 0 & if\ s \neq 0 \end{cases}$$

or approximately

$$f_\sigma(s) \approx \begin{cases} 1 & if\ |s| \ll \sigma \\ 0 & if\ |s| \gg \sigma \end{cases}$$

- Now define

$$F_\sigma(s) = \sum_{i=1}^{m} f_\sigma(s_i)$$
$$\|s\|_0 \approx m - F_\sigma(s)$$

- For small values of σ, the approximation tends to equality. Hence, we can define the minimum L0 norm solution by maximizing $F_\sigma(s)$.

- The value of σ determines how smooth the function $F_\sigma(s)$ is: the larger value of σ, the smoother $F_\sigma(s)$ (but worse approximation to L0-norm); and the smaller value of σ, closer the behavior of $F_\sigma(s)$ to L0-norm.
- However, for smaller values of σ, $F_\sigma(s)$ is highly non-smooth and contains a lot of local maxima, and hence its maximization is not easy. On the other hand, for larger values of σ, $F_\sigma(s)$ is smoother and contains less local maxima (in fact, no local maxima for large σ).
- "Basic idea": In order to find an s that maximizes $F_\sigma(s)$, the authors start with maximum σ. For this maximum σ, they find the maximizer of $F_\sigma(s)$. Then they decrease σ and again find the maximizer of $F_\sigma(s)$.
- They claim that eventually this process (decreasing σ and maximizing $F_\sigma(s)$) results in the maximization of $F_\sigma(s)$ or equivalently minimization of the L0 norm.

- Other family of functions that approximates the Kronecker delta functions like family of triangular functions,

$$f_\sigma(s) = \begin{cases} 1, & \text{if } |s| \geq \sigma \\ \frac{(\sigma+s)}{\sigma}, & \text{if } -\sigma \leq s \leq 0 \\ \frac{(\sigma-s)}{\sigma}, & \text{if } 0 \leq s \leq \sigma \end{cases}$$

- or truncated hyperbolic functions

$$f_\sigma(s) = \begin{cases} 1, & \text{if } |s| \geq \sigma \\ 1 - \left(\frac{s}{\sigma}\right)^2, & \text{if } |s| \leq \sigma \end{cases}$$

- or functions of the form

$$f_\sigma(s) = \frac{\sigma^2}{(s^2 + \sigma^2)}.$$

- For sufficiently large values of σ the maximizer of $F_\sigma(s)$ subject to $As=x$ is the minimum L2-norm solution, i.e., $\hat{s} = A^T\left(AA^T\right)^{-1}x$.

  Justification of the statement for Gaussian family:

  We want to maximize $F_\sigma(s) = \sum_{i=1}^{m} f_\sigma(s_i) = \sum_{i=1}^{m} e^{\left(-s_i^2/2\sigma^2\right)}$ subject to $As=x$

  The Lagrangian is $F_\sigma(s) - \lambda^T(As - x)$. Differentiating the Lagrangian w.r.t s and λ and setting the result to zero gives the following KKT systems of $m+n$ non-linear equations of $m+n$ unknowns.

$$\left[s_1 e^{-s_1^2/2\sigma^2}, s_2 e^{-s_2^2/2\sigma^2}, \cdots, s_m e^{-s_m^2/2\sigma^2}\right]^T - A^T\lambda_1 = 0 \qquad \lambda_1 = -\sigma^2\lambda$$
$$As - x = 0$$

  Now, let us look at this problem: $\min \|s\|_2^2$ *s.t.* $As = x$. Again using Lagrange multipliers this minimization results in the system of equations

$$[s_1, s_2, \cdots, s_m]^T - A^T\lambda = 0$$
$$As - x = 0$$

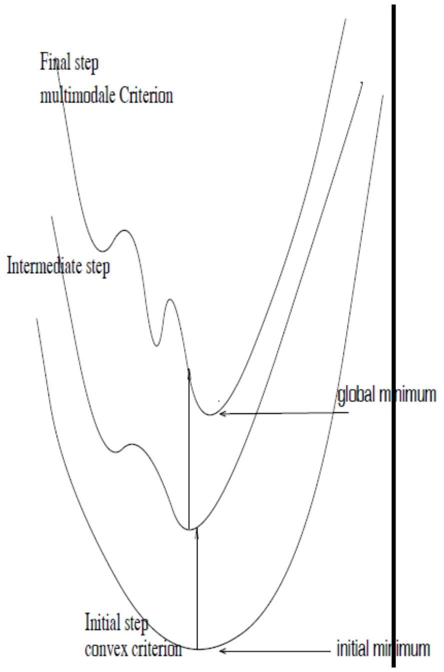
  Authors claim that for a larger σ these two systems of equations are identical and hence the maximizer of $F_\sigma(s)$ for larger σ is the minimum L2 norm solution.

- Hence, the authors start with large σ and maximize the corresponding $F_\sigma(s)$. They then decrease σ and repeat the maximization of $F_\sigma(s)$ again. They repeat the process for a few sequences of σ and shown that the subsequent maximization of $F_\sigma(s)$ leads to L0 solution. Their algorithm is based on the principles of graduated non-convexity.

  From Wikipedia:

- Graduated non-convexity is a global optimization technique that attempts to solve a difficult optimization problem by initially solving a greatly simplified problem, and progressively transforming that problem (while optimizing) until it is equivalent to the difficult optimization problem.

- Graduated optimization is an improvement to hill climbing that enables a hill climber to avoid settling into local optima. It breaks a difficult optimization problem into a sequence of optimization problems, such that the first problem in the sequence is convex (or nearly convex), the solution to each problem gives a good starting point to the next problem in the sequence, and the last problem in the sequence is the difficult optimization problem that it ultimately seeks to solve. Often, graduated optimization gives better results than simple hill climbing [1].



III. The SL0 Algorithm

The algorithm consists of 2 loops: the outer and inner loops.

In the outer loop, they vary the values of σ.

In the inner loop, for a given σ they use a steepest ascent algorithm for maximizing $F_\sigma(s)$.

- Initialization:
  1) Let $\hat{\mathbf{s}}_0$ be equal to the minimum ℓ^2 norm solution of $\mathbf{As} = \mathbf{x}$, obtained by pseudo-inverse of $\mathbf{A}$.
  2) Choose a suitable decreasing sequence for σ, $[\sigma_1 \dots \sigma_J]$ (see Remarks 5 and 6 of the text).
- For $j = 1, \dots, J$:
  1) Let $\sigma = \sigma_j$.
  2) Maximize (approximately) the function F_σ on the feasible set $\mathcal{S} = \{\mathbf{s} \mid \mathbf{As} = \mathbf{x}\}$ using L iterations of the steepest ascent algorithm (followed by projection onto the feasible set):
     - Initialization: $\mathbf{s} = \hat{\mathbf{s}}_{j-1}$.
     - For $\ell = 1 \dots L$ (loop L times):
       a) Let $\boldsymbol{\delta} \triangleq [s_1 \exp(-s_1^2/2\sigma^2), \dots, s_n \exp(-s_n^2/2\sigma^2)]^T$.
       b) Let $\mathbf{s} \leftarrow \mathbf{s} - \mu\boldsymbol{\delta}$ (where μ is a small positive constant).
       c) Project $\mathbf{s}$ back onto the feasible set $\mathcal{S}$:
       $$\mathbf{s} \leftarrow \mathbf{s} - \mathbf{A}^T(\mathbf{AA}^T)^{-1}(\mathbf{As} - \mathbf{x}).$$
  3) Set $\hat{\mathbf{s}}_j = \mathbf{s}$.
- Final answer is $\hat{\mathbf{s}} = \hat{\mathbf{s}}_J$.

Fig. 1. Final SL0 algorithm.

Remark 1: The internal loop is repeated for a fixed, say L, and small number of times. In other words, we do not wait for the steepest ascent algorithm (SAA) to converge. That is, we do not need the exact maximizer of $F_\sigma(s)$. We just need to enter the region near the (global) maximizer of for escaping from its local maximizer.

Remark 2: The SAA consists of update step $s \leftarrow s + \mu_j \nabla F_\sigma(s)$. Here, μ_j s are step size parameter and they should be chosen such that for decreasing values of σ, μ_j should be smaller. In the algorithm, they let $\mu_j = \mu\sigma^2$ for some constant μ. Then,

$$s \leftarrow s + \mu_j \nabla F_\sigma(s) = s - \mu\delta \text{ where } \delta = -\sigma^2 \nabla F_\sigma = \left[s_1 e^{-s_1^2/2\sigma^2}, s_2 e^{-s_2^2/2\sigma^2}, \cdots, s_m e^{-s_m^2/2\sigma^2} \right]^T.$$

Remark 3: Each iteration of the inner loop consists of gradient ascent step, followed by a projection step.

If we are looking for a suitable large μ (to reduce the required number of iterations), a suitable choice is to make the algorithm to force all those values of s_i satisfying $|s_i| \le \sigma$ toward zero. For this aim, we should have $\mu \exp\left(\frac{-s_i^2}{2\sigma^2}\right) \approx 1$, and because $\exp\left(\frac{-s_i^2}{2\sigma^2}\right) \le 1$ for $|s_i| \le \sigma$, the choice $\mu \ge 1$ seems reasonable.

Remark 4: The algorithm may work by initializing to an arbitrary solution. However, as discussion before the best initial value of is the minimum L2 solution. In fact, calculating minimum L2norm is one of the earliest approaches used for estimating the sparsest solution called the method of frames [4].

Remark 5: After initiating with minimum L2 solution, the next value for σ may be chosen about two to four times of the maximum absolute value of the obtained solution ($\max_i |s_i|$). For example, if we take $\sigma > 4\max_i |s_i|$, then $\exp\left(\frac{-s_i^2}{2\sigma^2}\right) \geq 0.96 \approx 1 \;\; i = 1, 2, \cdots, m$. This value of σ acts virtually like infinity for all values of s_i.

Remark 6: The term $F_\sigma(s)$ ($\|s\|_0 \approx m - F_\sigma(s)$) simply counts the number of zero components of s. However, instead of hard-thresholding that is "zero $\equiv |s_i| < \sigma$" and "non-zero $\equiv |s_i| > \sigma$" $f_\sigma(s) \approx \begin{cases} 1 & if\ |s| \ll \sigma \\ 0 & if\ |s| \gg \sigma \end{cases}$ uses a soft-thresholding, for which σ is a rough threshold.

Remark 7:

- If $\boldsymbol{s}$ is an exactly K-sparse signal, then σ can be decreased to arbitrarily small values. In fact, in this case, the minimum value of σ is determined by the desired accuracy (as will be discussed in Theorem 1).
- If $\boldsymbol{s}$ is an approximately K-sparse signal (say the source vector is noisy), then the smallest σ should be about one to two times of (a rough estimation of) the standard deviation of the noise (in the source vector). This is because, while σ is in this range, $f_\sigma(s)$ shows that the cost function treats small (noisy) samples as zeros (i.e., for which $f_\sigma(s) \approx 1$).
- However, below this range, the algorithm tries to 'learn' these noisy values, and moves away from the true answer. (According to the previous remark, the soft threshold should be such that all these noisy samples be considered zero).
- Restricting σ to be above the standard deviation of the noise, provides the robustness of this approach to noisy sources, which was one of the difficulties in using the exact L0 norm.

IV. Analysis of the algorithm

A. Convergence analysis

In this section, we try to answer two questions for the noiseless case (the noisy case will be considered in Section IV-C):

a) Does the SL0 solution converges to the actual minimizer of the L0 norm?
b) If yes, how much should we decrease σ to achieve a desired accuracy?

Assuming the maximization of $F_\sigma(s)$ for fixed σ is perfectly done (and we obtain the maximizer s^σ). The authors show that the sequence of 'global' maximizers of $F_\sigma(s)$'s will converge to the sparsest solution. That is we need to prove

$$\lim_{\sigma\to 0} s^\sigma = s^0$$

For proving the above statement the authors have introduced three intermediate results via lemmas.

Lemma 1: Assume a matrix $A \in R^{n\times m}$ has the property that all of its $n \times n$ sub-matrices are invertible, which is called the unique representation property (URP) in [3]. For any $\forall s \in N(A)$ if the *m-n* elements of ***s*** have absolute values less than $\alpha \;\Rightarrow\; \|s\| \le \beta$.

Proof: We have to show that

$\forall\, \beta > 0, \quad \exists \alpha > 0, \quad s.t \quad \forall s \in N(A):$

m-n elements of ***s*** have absolute values less than $\alpha \;\Rightarrow\; \|s\| \le \beta$ $\qquad \|\,.\,\|$ stands for L2 norm

Let $s \in N(A)$ and assume that the absolute values of at least *m-n* elements of it are smaller than α. Let I_α be the set of all indices *i*, for which $|s_i| > \alpha$. Consequently, $|I_\alpha| \le n$. Then we write

$$\sum_{i=1}^{m} s_i a_i = 0 \Rightarrow \sum_{i\in I_\alpha} s_i a_i + \sum_{i\notin I_\alpha} s_i a_i = 0$$

$$\Rightarrow \left\|\sum_{i\in I_\alpha} s_i a_i\right\| = \left\|\sum_{i\notin I_\alpha} s_i a_i\right\| \le \sum_{i\notin I_\alpha}\|s_i a_i\|$$

$$= \sum_{i\notin I_\alpha} \underbrace{|s_i|}_{\le\alpha}\underbrace{\|a_i\|}_{=1}$$

$$\le (m - |I_\alpha|)\alpha \le m\alpha$$

Let $\hat{A}$ be the submatrix of *A* containing only those columns of *A* that are indexed by the elements of I_α. Thus, $\hat{A}$ has at most *n* columns, and the columns of $\hat{A}$ are linearly independent, because of the URP of *A*. Therefore, there exists a left inverse $\hat{A}^{-1}$ for $\hat{A}$.
Let $\bar{s}$ and $\tilde{s}$ denote those sub-vectors of ***s*** which are, and which are not indexed by I_α, respectively. Then

$$\left(\sum_{i\in I_\alpha} s_i a_i\right) = \hat{A}\bar{s} \;\Rightarrow\; \bar{s} = \hat{A}^{-1}\left(\sum_{i\in I_\alpha} s_i a_i\right)$$

$$\|\bar{s}\| = \left\|\hat{A}^{-1}\left(\sum_{i\in I_\alpha} s_i a_i\right)\right\|$$

$$\|\bar{s}\| \le \|\hat{A}^{-1}\| \left\|\sum_{i\in I_\alpha} s_i a_i\right\| = \|\hat{A}^{-1}\| m\alpha$$

$$\|\tilde{s}\| \le \sum_{i\notin I_\alpha} |s_i| \le (m - |I_\alpha|)\alpha \le m\alpha$$

$$\|s\| \le \|\bar{s}\| + \|\tilde{s}\| \le \left(1 + \|\hat{A}^{-1}\|\right) m\alpha$$

Now, let $\mathcal{M}$ be the set of all submatrices $\hat{A}$ of *A*, consisting of at most *n* columns of *A*. Then, $\mathcal{M}$ is clearly a finite set (in fact $|\mathcal{M}| < 2^m$).

Let $M = \max\left\{\|\hat{A}^{-1}\| \,\middle|\, \hat{A} \in \mathcal{M}\right\}$ then

$$\|s\| \le \left(1 + \|\hat{A}^{-1}\|\right) m\alpha \le \left(1 + M\right) m\alpha.$$

M is a constant and its value depends only on the matrix *A*. Thus, for each β it is suffice to choose $\alpha = \beta / m(M+1)$

<u>*Corollary 1:*</u> If $A \in R^{n\times m}$ satisfies the URP, and $s \in N(A)$ has at most *n* elements with absolute values greater than α, then $\|s\| < \left(1 + M\right) m\alpha$.

<u>*Lemma 2:*</u>

Let a function $f_\sigma(s)$ have the properties $f_\sigma(0) = 1$ and $\forall s\ 0 \le f_\sigma(s) \le 1$, and let $F_\sigma(s) = \sum_{i=1}^{m} f_\sigma(s_i)$. Assume *A* satisfies the URP, and let $S = \left\{s \mid As = x\right\}$. Assume that there exists a (sparse) solution $s^0 \in S$ for which $\|s\|_0 = k \le \frac{n}{2}$ (such a sparse solution is unique). Then, if for a solution $\hat{s} = \left(\hat{s}_1, \hat{s}_2, \cdots, \hat{s}_m\right)^T \in S$, $F_\sigma(\hat{s}) \ge m - (n-k)$ and if $\alpha > 0$ is chosen such that the $\hat{s}_i$'s with absolute values greater than α satisfy $f_\sigma(\hat{s}_i) \le 1/m$, then $\left\|\hat{s} - s^0\right\| < \left(M+1\right) m\alpha$

<u>*Proof:*</u> Let I_α be the set of all indices *i*, for which $|\hat{s}_i| > \alpha$, then

$$F_\alpha(\hat{s}) = \sum_{i=1}^{m} f_\alpha(\hat{s}_i)$$

$$= \underbrace{\sum_{i\in I_\alpha} \underbrace{f_\alpha(\hat{s}_i)}_{\le(1/m)}}_{\le m.(1/m)=1} + \underbrace{\sum_{i\notin I_\alpha} \underbrace{f_\alpha(\hat{s}_i)}_{\le 1}}_{\le m-|I_\alpha|}$$

$$\le 1 + m - |I_\alpha|$$

We assume that we have chosen $f_\sigma(s)$ such that $F_\sigma(\hat{s}) \ge m-(n-k)$. (We prove this next)

Now, we get $m-(n-k) \le F_\sigma(\hat{s}) \le 1+m-|I_\alpha|$, from which we can get $|I_\alpha| \le n-k$.

As a result, at most *n-k* elements of $\hat{s}$ have absolute values greater than α.

Since s^0 has exactly *k* non-zero elements, we conclude that $\hat{s}-s^0$ has at most (*n-k*)+*k*=*n* elements with absolute values greater than α.

Moreover, $\hat{s}-s^0 \in N(A)$ and hence by Corollary 1 we have $\|\hat{s}-s^0\| < (M+1)m\alpha$.

Corollary 2: For the Gaussian family $f_\sigma(s) \triangleq \exp\left(\frac{-s^2}{2\sigma^2}\right)$, if $F_\sigma(\hat{s}) \ge m-(n-k)$ holds for a solution $\hat{s}$, then

$$\|\hat{s}-s^0\| < (M+1)m\sigma\sqrt{2\ln m}$$

Proof:

For the Gaussian family $f_\sigma(s) \triangleq \exp\left(\frac{-s^2}{2\sigma^2}\right)$, α required for lemma 2 can be chosen as $\alpha = \sigma\sqrt{2\ln m}$. Because for $|\hat{s}_i| > \sigma\sqrt{2\ln m}$,

$$f_\sigma(\hat{s}_i) = \exp\left(\frac{-\hat{s}_i^2}{2\sigma^2}\right) < \exp\left(\frac{-2\sigma^2\ln m}{2\sigma^2}\right) = \frac{1}{m}$$

Moreover, Gaussian family satisfies the other condition required in lemma 2.

Lemma 3: Let f_σ, F_σ, S and s^0 be as in Lemma 2, and let s^σ be the maximizer of $F_\sigma(s)$ on *S*, then s^σ satisfies $F_\sigma(\hat{s}) \ge m-(n-k)$.
Proof: We write

$$F_\sigma(s^\sigma) \ge F_\sigma(s^0)$$
$$\ge m-k$$
$$\ge m-(n-k) \qquad (\because\ k \le \tfrac{n}{2})$$

Note that Lemma 3 and Corollary 2 prove together that for the Gaussian family $f_\sigma(s) \triangleq \exp\left(\frac{-s^2}{2\sigma^2}\right)$ $\arg\max_{As=x} F_\sigma(s) \to s^0$ as $\sigma \to 0$. This result can however be stated for a larger class of functions, as done in Theorem 1 (next page).

Theorem 1: Consider a family of univariate functions f_σ, indexed $\sigma \in R^+$ satisfying the following set of conditions:

1. $\lim_{\sigma \to 0} f_\sigma(s) = 0; \quad \forall s \neq 0$
2. $f_\sigma(0) = 1; \quad \forall \sigma \in R^+$
3. $0 \le f_\sigma(s) \le 1; \quad \forall \sigma \in R^+ \quad s \in R$
4. For each positive values of ν and α, there exists $\sigma_0 \in R^+$ that satisfies

$$|s| > \alpha \;\Rightarrow\; f_\sigma(s) < \nu; \quad \forall \sigma < \sigma_0 \tag{1}$$

Assume *A* satisfies the URP, and let F_σ, *S* and s^0 be defined as in Lemma 2, and $s^\sigma = \left(s_1^\sigma, s_2^\sigma, \cdots, s_m^\sigma\right)^T \in S$ be the maximizer of F_σ on *S*. Then:

$$\lim_{\sigma \to 0} s^\sigma = s^0$$

Working definition of limit of a sequence

We say that $\lim_{n \to \infty} a_n = L$ if we can make a_n as close to *L* as we want for sufficiently large *n*.

Precise definition of the limit

We say that $\lim_{n \to \infty} a_n = L$ if for some positive error term ε the distance of the sequence at *n* from *L* must be less than the allowed error ε, that is, $|a_n - L| < \varepsilon$. But, it is important to remember that it is not enough that our sequence does converge once or twice; it must be within the error for all values from some point onwards, that is, $|a_n - L| < \varepsilon$, $\forall n > N$.

Analytically, $\forall \varepsilon > 0, \quad \exists N \in \mathbb{N}, \quad \forall n > N \quad |a_n - L| < \varepsilon.$

Proof: To prove $\lim_{\sigma \to 0} s^\sigma = s^0$, we have to show that

$$\forall \beta > 0, \quad \exists \sigma_0 > 0, \quad \forall \sigma < \sigma_0 \quad \left\| s^\sigma - s^0 \right\| < \beta. \tag{2}$$

For each $\beta > 0$, let $\alpha = \beta / m(M+1)$ .Then for this α and $\nu = 1/m$, condition 4 of the theorem gives a σ_0 for which the (1) holds. We show that this is the σ_0 we were seeking for in (2).

Note that $\forall \sigma < \sigma_0$, (1) states that for s_i^σ's with absolute values greater than α we have $f_\sigma(s_i^\sigma) \le 1/m$. Moreover, Lemma 3 states that s^σ satisfies $F_\sigma(s^\sigma) \ge m - (n-k)$. Consequently, all the conditions of Lemma 2 have been satisfied, and hence it implies that $\left\| s^\sigma - s^0 \right\| < (M+1)m\alpha = \beta$.

Remark 1: The Gaussian family $f_\sigma(s) \triangleq \exp\left(\frac{-s^2}{2\sigma^2}\right)$ satisfies conditions 1 through 4 of Theorem 1. Other Families of functions also satisfy the conditions of Theorem 1.

Remark 2: Using Corollary 2, where using Gaussian family, to ensure an arbitrary accuracy in estimation of the sparse solution s^0, it suffices to choose

$$\sigma < \frac{\beta}{m(M+1)\sqrt{2\ln m}} \quad \text{and do the optimization of } F_\sigma(s) \text{ subject to } A\boldsymbol{s}=\boldsymbol{x}.$$

Remark 3: Consider the set of solutions in $\hat{s}^\sigma$ in *S*, which might not be the absolute maxima of functions F_σ on *S*, but satisfy the condition

$$F_\sigma(\hat{s}^\sigma) \geq m-(n-k)$$

By following a similar approach to the proof of Theorem 1, it can be proved that $\lim_{\sigma\to 0} s^\sigma = s^0$. In other words, for the steepest ascent (internal loop), it is not necessary to reach the absolute maximum. It is enough reach a solution in which is F_σ large.

Remark 4: The previous remark proposes another version of SL0 in which there is no need to set a parameter L: Repeat the internal loop until $F_\sigma(s)$ exceeds *m*-*n*/2[the worst case of the limit given by $F_\sigma(\hat{s}) \geq m-(n-k)$] or $m-(n-k)$ if *k* is known a priori. The advantage of such a version is that if it converges, then it is guaranteed that the estimation error is bounded as $\left\|\hat{s}-s^0\right\| < (M+1)m\sigma\sqrt{2\ln m}$, in which σ is replaced with σ_J, the last element of the sequence of σ.

It has, however, two disadvantages: first, it slows down the algorithm because exceeding the limit $m-(n-k)$ for each σ is not necessary (it is just sufficient); and second, because of the possibility that the algorithm runs into an infinite loop because $F_\sigma(s)$ cannot exceed this limit (this occurs if the chosen sequence of σ has not been resulted in escaping from local maxima).

Remark 5: As another consequence, Lemma 1 provides an upper bound on the estimation error $\left\|\hat{s}-s^0\right\|$, only by having an estimation $\hat{s}$ (which satisfies $A\hat{s}=x$): Begin by sorting the elements of $\hat{s}$ in descending order and α let be the absolute value of the $\lfloor \frac{n}{2} \rfloor$+1'th element. Since s^0 has at most *n*/2 nonzero elements, $\hat{s}-s^0$ has at most *n* elements with absolute values greater than α. Moreover, $\hat{s}-s^0 \in N(A)$ and hence Corollary 1 implies that $\left\|\hat{s}-s^0\right\| < (M+1)m\alpha$. This result is consistent with the heuristic that "if $\hat{s}$ has at most *n*/2 'large' components, the uniqueness of the sparsest solution insures that $\hat{s}$ is close to the true solution."

B. Relation to minimum norm 2 solution

Kindly refer the paper for proof of $\lim_{\sigma\to\infty} \arg\max_{As=x} F_\sigma(s) = \hat{s}$, where $\hat{s}$ is the minimum L2 norm solution.

C. Noisy case

As shown in the proof of Theorem 1 (noiseless case), a smaller value of σ results in a more accurate solution and it is possible to achieve solutions as accurate as desired by choosing small enough values of σ. However, this is not the case in the presence of additive noise, that is, if x=As+n. In fact, the noise power bounds the maximum achievable accuracy.

V. Numerical Results

The performance of the SL0 algorithm is experimentally verified and is compared with BP (FOCUSS) and LP (L1 magic). The effects of the parameters, sparsity, noise, and dimension on the performance are also experimentally discussed (Please refer the paper).

In experiments, sparse sources are artificially created using a Bernoulli–Gaussian model: each source is "active" with probability *p*, and is "inactive" with probability (1-*p*). If it is active, each sample is a zero-mean Gaussian random variable with variance σ_{on}^2; if it is not active, each sample is a zero-mean Gaussian random variable with variance σ_{off}^2, where $\sigma_{\text{off}}^2 \ll \sigma_{\text{on}}^2$.

Each column of the mixing matrix is randomly generated using the normal distribution and then is normalized to unity.

To evaluate the estimation quality, signal-to-noise ratio (SNR) and mean-square error (MSE) are used. SNR (in dB) is defined as $20\log\left(\frac{\|s\|}{\|s-\hat{s}\|}\right)$ and MSE as $\frac{1}{m}\|s-\hat{s}\|^2$

TABLE I
Progress of SL0 for a Problem With $m = 1000$, $n = 400$ and $k = 100$ ($p = 0.1$)

itr. #	σ	MSE	SNR (dB)
1	1	4.84 $e-2$	2.82
2	0.5	2.02 $e-2$	5.19
3	0.2	4.96 $e-3$	11.59
4	0.1	2.30 $e-3$	16.44
5	0.05	5.83 $e-4$	20.69
6	0.02	1.17 $e-4$	28.62
7	0.01	5.53 $e-5$	30.85
algorithm	**total time (sec)**	**MSE**	**SNR (dB)**
SL0	0.227	5.53 $e-5$	30.85
LP (ℓ_1-magic)	30.1	2.31 $e-4$	25.65
FOCUSS	20.6	6.45 $e-4$	20.93

VI. CONCLUSIONS

In this paper, authors showed that the smoothed L0 norm can be used for finding sparse solutions of an USLE. They also showed that the smoothed version of the L0 norm results in an algorithm which is faster than the state-of-the-art algorithms based on minimizing the L1 norm.

Moreover, this smoothing solves the problem of high sensitivity of L0 norm to noise. In another point of view, the smoothed L0 provides a smooth measure of sparsity.

The basic idea of the paper was justified by both theoretical (convergence in both noiseless and noisy case, relation to the L2 norm solution) and experimental analysis of the algorithm.

Appendix (Not available in the paper)

The following are taken from [5]

Consider the following problem of the Euclidean orthogonal projection of a point to an affine set: For the given $A \in R^{m\times n}$, $b \in R^m$ and $p \in R^n$, find a vector $x^* \in R^n$ satisfying

$$Ax^* = b$$
$$\left\| p - x^* \right\| = \min_{Ax=b} \left\| p - x \right\|$$

The solution to the above problem exits and it is unique and it is

$$x^* = p - A^+ Ap + A^+ b \qquad [x^* = p - A^+(Ap - b)]$$
$$= \left[I - A^+ A \right] p + A^+ b$$
$$x^* = P_{N(A)} p + A^+ b$$

Reference

[1] A. Blake and A. Zisserman, *Visual Reconstruction*. Cambridge, MA: MIT Press, 1987.

[2] P. Bofill and M. Zibulevsky, "Underdetermined blind source separation using sparse representations," in *Signal Process.*, 2001, vol. 81, pp. 2353–2362.

[3] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS, A re-weighted minimum norm algorithm," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 600–616, Mar. 1997.

[4] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Scientif. Comput.*, vol. 20, no. 1, pp. 33–61, 1999.

[5] J. Ding, "Perturbation analysis for the projection of a point into the affine set," *Linear Algebra and its applications*, vol. 191, pp. 199-212, 1993.

Expectation-Maximization Belief Propagation for Sparse Recovery I: Algorithm Construction

Main ref.: F. Krzakala *et al.*,"Statistical physics-based re construction in compressed sensing," *Phys. Rev X*, May, 2012 [1]

Presenter: Jaewook Kang

In Proceeding of CS Journal Club in GIST

Date: Nov. 2012, updated at Jan, 2013

I. Introduction

In this report, the author introduces a expectation maximization (EM) based belief propagation algorithm (BP) for sparse recovery, named EM-BP. The algorithm have been mainly devised by Krzakala *et al.* from ParisTech in France [1]. The properties of EM-BP are as given below:

1) It is A low-computation approach to sparse recovery,
2) It works well without the prior knowledge of the signal,
3) It overcomes the l_1 phase transition given by Donoho and Tanner [11] under the noiseless setup,
4) It is further improved in conjunction with seeding matrices (or spatial coupling matrices).

The main purpose of this report regenerates a precise description of EM-BP algorithm construction from the reference paper [1]. It might be very helpful for understanding of EM-BP algorithm, and an answer for such a question: How and why does the algorithm work ? Therefore, we will focus on the explanation of 1) and 2) in the properties, and just show the result of the paper with respect to that of 3) and 4).

In addition to EM-BP, the belief propagation approach to the sparse recovery problem has been widely investigated in [2],[3],[4],[5],[6],[7].

II. Problem setup

In the sparse recovery problem, the aim is to recovery a sparse signal $\mathbf{X} \in \mathbb{R}^N$ whose elements have nonzero value independently each other, with a probability rate q called sparsity rate. Therefore, the q determines the density of signal $\mathbf{X}$. Then, the algorithm performs the recovery from the measurements $\mathbf{Y} \in \mathbb{R}^M$, given as

$$\mathbf{Y} = \mathbf{\Phi}\mathbf{X} + \mathbf{N}, \tag{1}$$

where $\mathbf{\Phi} \in \mathbb{R}^{M \times N}$ is a fat measurement matrix with $M < N$, and $\mathbf{N} \in \mathbb{R}^{M}$ denotes a additive Gaussian noise vector following $\mathcal{N}(0, \mathbf{I}\sigma_N^2)$.

III. Algorithm Construction of EM-BP

Krzakala *et al.* has taken a probabilistic approach to devise EM-BP. From the Bayesian point of view, the posterior density of the signal $\mathbf{X}$ is represented in the form of $\text{Posterior} = \text{Prior} \times \frac{\text{Likelihood}}{\text{Evidence}}$ as

$$f_{\mathbf{X}}(\mathbf{x}|\mathbf{y}, \mathbf{\Phi}) = f_{\mathbf{X}}(\mathbf{x}|\mathbf{\Phi}) \times \frac{f_{\mathbf{Y}}(\mathbf{y}|\mathbf{\Phi}, \mathbf{X})}{f_{\mathbf{Y}}(\mathbf{y}|\mathbf{\Phi})}. \tag{2}$$

Then, using the knowledge of $\mathbf{z}$ and $\mathbf{\Phi}$, the signal posterior is given as

$$f_{\mathbf{X}}(\mathbf{x}|\mathbf{y}, \mathbf{\Phi}) = \frac{1}{C} f_{\mathbf{X}}(\mathbf{x}) \times \prod_{j=1}^{M} \frac{1}{\sqrt{2\pi\sigma_N^2}} \exp\left[-\frac{1}{2\sigma_{N_j}^2}(y_j - \textstyle\sum_{i=1}^{N} \phi_{ji} x_i)^2\right], \tag{3}$$

where C is a normalization constant for $\int f_{\mathbf{X}}(\mathbf{x}|\mathbf{y}, \mathbf{\Phi})\, d\mathbf{x} = 1$. In addition, we consider a mixture type prior density function represented as

$$f_{\mathbf{X}}(\mathbf{x}) := \prod_{i=1}^{N} [(1-q)\delta_0 + q\theta(x_i)], \tag{4}$$

where $\theta(x_i)$ is a Gaussian PDF with mean $\bar{x}$ and variance σ_X^2.

Exact finding of the signal posterior is computationally infeasible. Therefore, researchers have employed BP as a standard approach to approximate the signal posterior where BP finds marginal posterior density of each signal element X_i. In addition, Guo *et al.* showed that the marginal posterior finding is exact if the matrix $\mathbf{\Phi}$ is a sparse matrix and $N \to \infty$ [8],[9],[10]. BP seeks the signal posterior by iteratively exchanging probabilistic messages over the signal elements, where the messages are classically described as

Measurement to signal (MtS) message :

$$m_{j\to i}(x_i) := \frac{1}{C_{j\to i}} \int\limits_{\{x_k\}_{k\neq i}} \prod_{k\neq i} m_{k\to j}(x_i) \times \exp\left[-\frac{1}{2\sigma_N^2}(\sum_{k\neq i} \phi_{jk} x_k + \phi_{ji} x_i - y_j)^2\right] \left(\prod_{k\neq i} dx_i\right), \tag{5}$$

Signal to measurement (StM) message :

$$m_{i\to j}(x_i) := \frac{1}{Z_{i\to j}} [(1-q)\delta_0 + q\theta(x_i)] \times \prod_{k\neq j} m_{k\to i}(x_i), \tag{6}$$

where $C_{j\to i}$ and $Z_{i\to j}$ are normalization constants to make the messages as PDFs. Then, the marginal posterior approximately is obtained as

$$f_{X_i}(x|\,\mathbf{y}, \mathbf{\Phi}) \overset{\text{BP}}{=} \frac{1}{C_i} [(1-q)\delta_{x_i} + q\theta(x_i)] \times \prod_{k} m_{k\to i}(x_i). \tag{7}$$

However, the message update rule in (5) and (6) is practically intractable because each BP message is probability density function (PDF). Therefore, we need to convert the density-passing procedure to a parameter-passing procedure using some relaxation techniques.

Using Hubbard-Stratonovich transformation (HST) from spin glass theory which is

$$\exp\left(-\frac{w^2}{2\sigma^2}\right) = \frac{1}{\sqrt{2\pi\sigma^2}} \int \exp\left(-\frac{\lambda^2}{2\sigma^2} + \frac{iw\lambda}{\sigma^2}\right) d\lambda, \tag{8}$$

the exponent in (5) can be rewritten as

$$\exp\left[-\frac{1}{2\sigma_{N_j}^2}(y_j - \sum_{i=1}^{N} \phi_{ji} x_i)^2\right] = \exp\left[\underbrace{-\frac{\left(\sum\limits_{k\neq i} \phi_{jk} x_k\right)^2}{2\sigma_{N_j}^2}}_{\text{Here, HST applied}} - \frac{\sum\limits_{k\neq i} \phi_{jk} x_k (\phi_{ji} x_i - y_j)}{\sigma_{N_j}^2} - \frac{(\phi_{ji} x_i - y_j)^2}{2\sigma_{N_j}^2}\right]$$

$$= \frac{1}{\sqrt{2\pi\sigma_{N_j}^2}} \int_{\lambda} \exp\left(-\frac{\lambda^2}{2\sigma_{N_j}^2} + \frac{\sum\limits_{k\neq i} \phi_{jk} x_k (\phi_{ji} x_i - y_j + i\lambda)}{\sigma_{N_j}^2} - \frac{(\phi_{ji} x_i - y_j)^2}{2\sigma_{N_j}^2}\right) d\lambda. \tag{9}$$

By applying (9) to (5), we have

$$\begin{aligned} m_{j\to i}(x_i) = & \frac{\exp\left(-\frac{(\phi_{ji} x_i - y_j)^2}{2\sigma_{N_j}^2}\right)}{C_{j\to i}\sqrt{2\pi\sigma_{N_j}^2}} \int_{\lambda} \exp\left(-\frac{\lambda^2}{2\sigma_{N_j}^2}\right) \\ & \times \left\{ \int_{\{x_k\}_{k\neq i}} \prod_{k\neq i} m_{k\to j}(x_i) \exp\left(\frac{\sum\limits_{k\neq i} \phi_{jk} x_k (\phi_{ji} x_i - y_j + i\lambda)}{\sigma_{N_j}^2}\right) \prod_{k\neq i} dx_i \right\} d\lambda \end{aligned} \tag{10}$$

In (10), we observe that the integration over $\{x_k\}_{k\neq i}$ can be decomposed into integration over each scalar x_k. In addition, the integration over scalar x_k takes the form of the moment generating function. Therefore,

$$\begin{aligned} m_{j\to i}(x_i) &= \frac{\exp\left(-\frac{(\phi_{ji} x_i - y_j)^2}{2\sigma_{N_j}^2}\right)}{C_{j\to i}\sqrt{2\pi\sigma_{N_j}^2}} \int_{\lambda} \exp\left(-\frac{\lambda^2}{2\sigma_{N_j}^2}\right) \times \prod_{k\neq i} \left\{ \int_{\{x_k\}_{k\neq i}} m_{k\to j}(x_i) \exp\left(\frac{x_k \phi_{jk} (\phi_{ji} x_i - y_j + i\lambda)}{\sigma_{N_j}^2}\right) dx_i \right\} d\lambda \\ &= \frac{\exp\left(-\frac{(\phi_{ji} x_i - y_j)^2}{2\sigma_{N_j}^2}\right)}{C_{j\to i}\sqrt{2\pi\sigma_{N_j}^2}} \int_{\lambda} \exp\left(-\frac{\lambda^2}{2\sigma_{N_j}^2}\right) \times \prod_{k\neq i} \mathbf{E}_{X_k}\left[\exp\left(\frac{x_k \phi_{jk} (\phi_{ji} x_i - y_j + i\lambda)}{\sigma_{N_j}^2}\right)\right] d\lambda \end{aligned} \tag{11}$$

By assuming that each scalar X_k is Gaussian distributed during the BP-iteration with mean $\mu_{i\to j}$ and variance $\sigma^2_{i\to j}$, we can approximate the MtS message expression as

$$m_{j\to i}(x_i) \approx \frac{\exp\left(-\frac{(\phi_{ji}x_i - y_j)^2}{2\sigma^2_{N_j}}\right)}{C_{j\to i}\sqrt{2\pi\sigma^2_{N_j}}} \times$$
$$\times \int_\lambda \exp\left(-\frac{\lambda^2}{2\sigma^2_{N_j}}\right) \prod_{k\neq i} \exp\left(\frac{\mu_{k\to j}\phi_{jk}(\phi_{ji}x_i - y_j + i\lambda)}{\sigma^2_{N_j}} + \frac{\sigma^2_{k\to j}}{2}\left(\frac{\phi_{jk}(\phi_{ji}x_i - y_j + i\lambda)}{\sigma^2_{N_j}}\right)^2\right) d\lambda. \tag{12}$$

By evaluating the Gaussian integration over λ, the expression in (12) becomes

$$m_{j\to i}(x_i) \simeq \frac{\sqrt{A_{j\to i}/2\pi}}{\phi_{ji}C_{j\to i}} \times \exp\left(-\frac{x_i^2}{2}A_{j\to i} + x_i B_{j\to i} + \frac{B^2_{j\to i}}{2A_{j\to i}}\right). \tag{13}$$

where

$$A_{j\to i} := \frac{\phi^2_{ji}}{\sigma^2_{N_j} + \sum\limits_{k\neq j}\sigma^2_{k\to j}\phi^2_{jk}}, \tag{14}$$

$$B_{j\to i} := \frac{\phi_{ji}(y_j - \sum\limits_{k\neq j}\mu_{k\to j}\phi_{jk})}{\sigma^2_{N_j} + \sum\limits_{k\neq j}\sigma^2_{k\to j}\phi^2_{jk}}. \tag{15}$$

Then, the expression of the StM message is rewritten as

$$m_{i\to j}(x_i) := \frac{1}{\widetilde{Z}_{i\to j}}[(1-q)\delta_0 + q\theta(x_i)] \times \exp\left(-\frac{x_i^2}{2}\sum_{k\neq j}A_{k\to i} + x_i\sum_{k\neq j}B_{k\to i} + \frac{1}{2}\frac{\left(\sum\limits_{k\neq j}B_{k\to j}\right)^2}{\sum\limits_{k\neq j}A_{k\to j}}\right), \tag{16}$$

where we use an approximation $\sum\limits_{k\neq j}B^2_{k\to j} \approx \left(\sum\limits_{k\neq j}B_{k\to j}\right)^2$. The exponent can be rewritten as

$$-\frac{x_i^2}{2}\sum_{k\neq j}A_{k\to i} + x_i\sum_{k\neq j}B_{k\to i} + \frac{1}{2}\frac{\left(\sum\limits_{k\neq j}B_{k\to j}\right)^2}{\sum\limits_{k\neq j}A_{k\to j}}$$
$$= -\frac{1}{2\frac{1}{\sum_{k\neq j}A_{k\to i}}}\left(x_i^2 - 2\frac{\sum\limits_{k\neq j}B_{k\to i}}{\sum\limits_{k\neq j}A_{k\to i}} + \left(\frac{\sum\limits_{k\neq j}B_{k\to i}}{\sum\limits_{k\neq j}A_{k\to i}}\right)^2\right) = -\frac{\left(x_i - \frac{\sum_{k\neq j}B_{k\to i}}{\sum_{k\neq j}A_{k\to i}}\right)^2}{2\frac{1}{\sum_{k\neq j}A_{k\to i}}} \tag{17}$$

Hence, equations (14) and (15) together with (19) fully describe the iterative BP-process. We define two variable given as

$$\Sigma_i^2 := \frac{1}{\sum\limits_{k\neq j} A_{k\to i}},\ R_i := \frac{\sum\limits_{k\neq j} B_{k\to i}}{\sum\limits_{k\neq j} A_{k\to i}}, \tag{18}$$

Using the notations, we rewrite the expression of the StM message given as Then, the expression of the StM message is rewritten as

$$m_{i\to j}(x_i) := \frac{1}{\widetilde{Z}_{i\to j}}[(1-q)\delta_0 + q\theta(x_i)] \times \exp\left(-\frac{(x_i - R_i)^2}{2\Sigma_i^2}\right), \tag{19}$$

Then, the mean $\mu_{k\to j}$ and variance $\sigma_{k\to j}^2$ of the StM message are calculated as

$$\begin{aligned}\mu_{i\to j} &:= \int\limits_{X_i} x_i m_{i\to j}(x_i)dx_i \\ &= \frac{q}{Z(\Sigma_i^2, R_i)} \int\limits_{X_i} x_i\theta(x_i)\exp\left(-\frac{(x_i-R_i)^2}{2\Sigma_i^2}\right)dx_i \\ &= \frac{q}{Z(\Sigma_i^2, R_i)} \times \frac{\Sigma_i(\overline{x}\Sigma_i^2 + R\sigma_X^2)}{(\Sigma_i^2+\sigma_X^2)^{3/2}}\exp\left(-\frac{(R-\overline{x})^2}{2(\Sigma_i^2+\sigma_X^2)}\right),\end{aligned} \tag{20}$$

and

$$\begin{aligned}\sigma_{i\to j}^2 &:= \int\limits_{X_i} x_i^2 m_{i\to j}(x_i)dx_i - \mu_{i\to j}^2 \\ &= \frac{q}{Z(\Sigma_i^2, R_i)}\int\limits_{X_i} x_i^2\theta(x_i)\exp\left(-\frac{(x_i-R_i)^2}{2\Sigma_i^2}\right)dx_i - \mu_{i\to j}^2 \\ &= \frac{q(1-q)\exp\left(-\frac{R_i^2}{2\Sigma_i^2} - \frac{(R-\overline{x})^2}{2(\Sigma_i^2+\sigma_X^2)}\right)\frac{\Sigma_i}{(\Sigma_i^2+\sigma_X^2)^{5/2}}\left(\sigma_X^2\Sigma_i^2(\Sigma_i^2+\sigma_X^2) + (\overline{x}\Sigma_i^2 + R\sigma_X^2)^2\right)}{Z(\Sigma_i^2, R_i)^2} \\ &+ \frac{q^2\exp\left(-\frac{(R-\overline{x})^2}{2(\Sigma_i^2+\sigma_X^2)}\right)\frac{\sigma_X^2\Sigma_i^4}{(\Sigma_i^2+\sigma_X^2)^2}}{Z(\Sigma_i^2, R_i)^2},\end{aligned} \tag{21}$$

where the normalization constant is

$$\begin{aligned}Z(\Sigma_i^2, R_i) :&= (1-q)\int\limits_{X_i}\delta_0\exp\left(-\frac{(x_i-R_i)^2}{2\Sigma_i^2}\right)dx_i + q\int\limits_{X_i}\theta(x_i)\exp\left(-\frac{(x_i-R_i)^2}{2\Sigma_i^2}\right)dx_i \\ &= (1-q)\exp\left(-\frac{R_i^2}{2\Sigma_i^2}\right) + q\frac{\Sigma_i}{\sqrt{\Sigma_i^2+\sigma_X^2}}\exp\left(-\frac{(R-\overline{x})^2}{2(\Sigma_i^2+\sigma_X^2)}\right).\end{aligned} \tag{22}$$

The authors stated that the parameters $\bar{x}$, σ_X^2, and q of the prior density $f_{\mathbf{X}}(\mathbf{X})$ can be learned and updated at every iteration. A statistical approach for the parameter learning is the use of EM. For the object function in EM, they used Bethe free-entropy. It is known that BP algorithm is constructed by applying Lagrange multipliers to Bethe entropy [12]. Therefore, the fixed point of BP-iteration corresponds to the stationary points of the Bethe free-entropy minimization, in the signal posterior finding problems. For details about the relationship between Bethe free-entropy and BP, please see Yedidia's paper.

The Bethe entropy is defined as

$$H_{\text{Bethe}} := -\sum_i^N H(Z_{x_i}) - \sum_j^M H(Z_{y_j}) + \sum_j^M \sum_{i\in N(j)} H(Z_{x_i}), \tag{23}$$

where the concept of free-entropy, defined as $H(Z) := \log Z$, is used and Z_{x_i} and Z_{y_j} are an approximated marginal partition function of x, that is,

$$Z_{x_i} = \int [(1-q)\delta_{x_i} + q\theta(x_i)] \times \prod_j m_{j\to i}(x_i) dx_i, \tag{24}$$

$$Z_{y_j} = \int \prod_i m_{i\to j}(x_i) \times \exp\left[-\frac{1}{2\sigma_N^2}(\sum_i \phi_{ji}x_i - y_j)^2\right] \prod_i (dx_i)\,. \tag{25}$$

Thus, the parameters $(\bar{x}, \sigma_X, q)$ are learned by seeking the stationary point of the Bethe free-entropy function given in (23). We update the parameter for the prior knowledge from

$$\overline{x} = \frac{\sum\limits_i \mu_i}{Nq} \tag{26}$$

$$\sigma_X^2 = \frac{\sum\limits_i (\sigma_i^2 + \mu_i^2)}{Nq} - \overline{x}^2 \tag{27}$$

$$q = \frac{\sum_i \frac{1/\sigma_X^2 + \sum_j A_{j\to i}}{\sum_j B_{j\to i} + \overline{x}/\sigma_X^2}\mu_i}{\sum_i \left(1 - q + \frac{q}{\sigma_X\sqrt{1/\sigma_X^2 + \sum_j A_{j\to i}}}\exp\left(\frac{(\sum_j B_{j\to i} + \overline{x}/\sigma_X^2)^2}{2(1/\sigma_X^2 + \sum_j A_{j\to i})} - \frac{\overline{x}^2}{2\sigma_X^2}\right)\right)^{-1}}. \tag{28}$$

I implemented the EM-BP algorithm using the equations of (14), (15), (20), (21), (22), (26), (27) in C language. I did not update the sparsity rate q in the BP-iteration. The performance is not working well as shown in Fig.1. I need to check my implementation by translating the code to MATLAB. I think the EM update not much improve the performance. So, we need to modify the update rule to elementwise update rule like SuPrEM Algorithm.

REFERENCES

[1] F. Krzakala, M. Mezard, F. Sausset, Y. Sun, L. Zdeborova, "Statistical physics-based reconstruction in compressed sensing," *Phys. Rev X*, No. 021005, May, 2012

[2] J. Kang, H.-N. Lee, and K. Kim, "Detection-directed sparse estimation using bayesian hypothesis test and belief propagation," submitted to *IEEE Trans. Signal Process.*, 2012

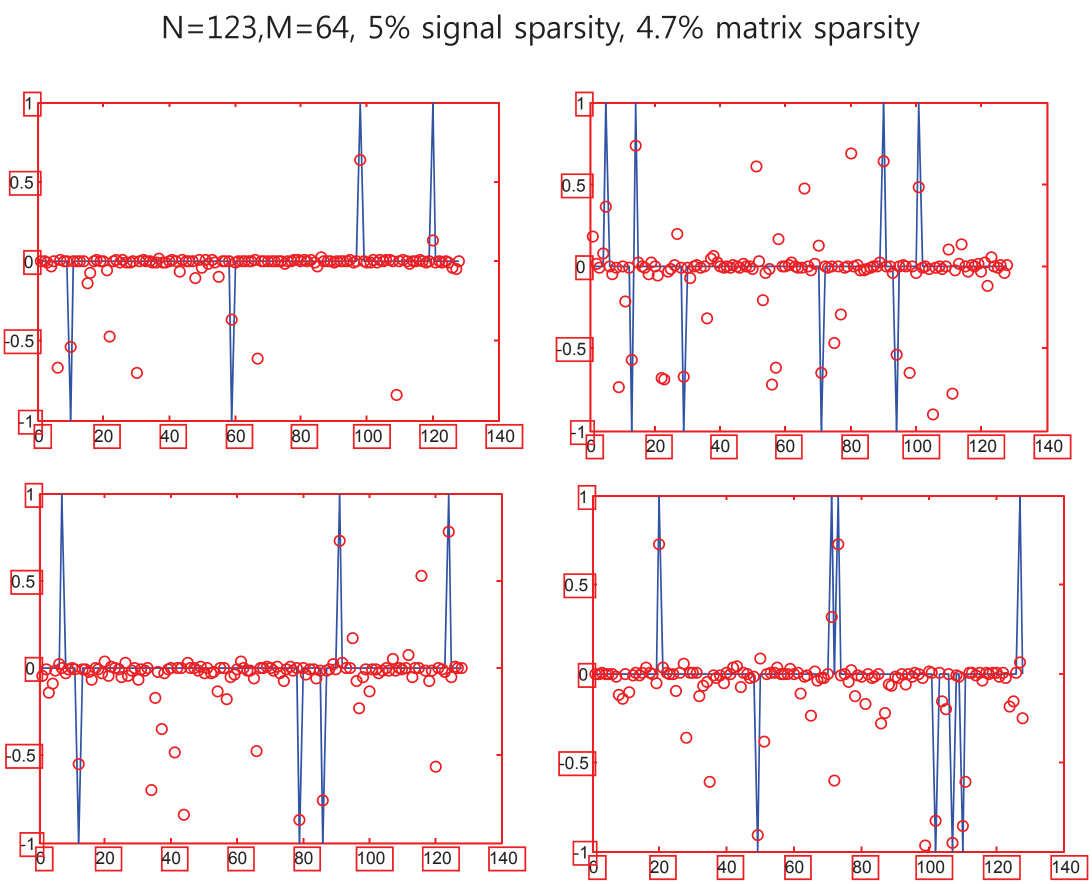


Fig. 1. Some simulation results of EM-BP when $N = 128, M = 64, q = 0.05, \sigma_X = 1$, and $L = 3$.

[3] D. Baron, S. Sarvotham, and R. Baraniuk, "Bayesian compressive sensing via belief propagation," *IEEE Trans. Signal Process.*, vol. 58, no. 1, pp. 269-280, Jan. 2010.

[4] X. Tan and J. Li, "Computationally efficient sparse Bayesian learning via belief propagation," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2010-2021, Apr. 2010.

[5] M. Akcakaya, J. Park, and V. Tarokh, "A coding theory approach to noisy compressive sensing using low density frame," *IEEE Trans. Signal Process.,* vol. 59, no. 12, pp. 5369-5379, Nov. 2011.

[6] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing,", *Proc. Nat. Acad. Sci.*, vol. 106, pp. 18914-18919, Nov. 2009.

[7] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," *Proc. in IEEE Int. Symp. Inform. Theory (ISIT)*, pp. 2168-2172, Aug. 2011.

[8] D. Guo and S. Verdu, "Randomly spread CDMA: Asymptotics via statistical physics," *IEEE Trans. Inform. Theory*, vol. 51, no. 6, pp. 1983-2010, Jun. 2005.

[9] D. Guo and C.-C. Wang, "Asymptotic mean-square optimality of belief propagation for sparse linear systems," *Proc. in*

IEEE Inform. Theory Workshop, pp. 194-198, Chengdu, China, Oct. 2006.

[10] D. Guo and C.-C. Wang, “Random sparse linear systems observed via arbitrary channels: a decoupling principle,”*Proc. in IEEE Int. Symp. Inform. Theory (ISIT)*, pp. 946-950, Nice, France, June. 2007.

[11] D.L. Donoho and J. Tanner, ”Precise undersampling theorems,” *Proceeding of the IEEE*, vol. 98, issue 6, pp. 913-924, June, 2010.

[12] J.S. Yedidia, W.T. Freeman, and Y. Weiss, “Construcing Free-Energy Approximations and Generalized Belief Propagation Algorithm,” *IEEE Trans. Inform. Theory*, vol. 51, no. 7, pp. 2282-2312, July. 2005.