Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2012, Article ID 478931, 22 pages
doi:10.1155/2012/478931

Research Article

Sparse Signal Recovery via ECME Thresholding Pursuits

Heping Song[1] and Guoli Wang[2]

[1] *School of Computer Science and Telecommunication Engineering, Jiangsu University, Zhenjiang 212013, China*
[2] *School of Information Science and Technology, Sun Yat-Sen University, Guangzhou 510006, China*

Correspondence should be addressed to Heping Song, hepingsong@gmail.com

Received 17 February 2012; Revised 24 April 2012; Accepted 8 May 2012

Academic Editor: Jung-Fa Tsai

The emerging theory of compressive sensing (CS) provides a new sparse signal processing paradigm for reconstructing sparse signals from the undersampled linear measurements. Recently, numerous algorithms have been developed to solve convex optimization problems for CS sparse signal recovery. However, in some certain circumstances, greedy algorithms exhibit superior performance than convex methods. This paper is a followup to the recent paper of Wang and Yin (2010), who refine BP reconstructions via iterative support detection (ISD). The heuristic idea of ISD was applied to greedy algorithms. We developed two approaches for accelerating the ECME iteration. The described algorithms, named ECME thresholding pursuits (EMTP), introduced two greedy strategies that each iteration detects a support set I by thresholding the result of the ECME iteration and estimates the reconstructed signal by solving a truncated least-squares problem on the support set I. Two effective support detection strategies are devised for the sparse signals with components having a fast decaying distribution of nonzero components. The experimental studies are presented to demonstrate that EMTP offers an appealing alternative to state-of-the-art algorithms for sparse signal recovery.

1. Introduction

Sparsity exploiting has recently received a great amount of attention in the applied mathematics and signal processing community. Sparse signal processing algorithms have been developed for various applications. A recent Proceedings of the IEEE special issue on applications of sparse representation and compressive sensing devoted to this topic. Some of the exciting developments addressed in [1–7]. Compressed sensing, also known as compressive sensing, compressive sampling (CS) [8, 9], has been the subject of significant activity in sparse signal processing where one seeks to recover efficiently a sparse unknown signal vector of dimension n via a much smaller number (m) of undersampled linear measurements. For k-sparse unknown signal $x_0 \in \mathbb{R}^n$, the sparse signal recovery is intimately related to solving an underdetermined system of linear equations $y = Ax_0$ with sparseness constraint

$$(P_{\ell_0}){:}\min_{x} \|x\|_{\ell_0} \quad \text{s.t. } Ax = y, \tag{1.1}$$

where x is true signal to be recovered, $\|x\|_{\ell_0}$ is used to denote the number of nonzero components of x, $A \in \mathbb{R}^{m \times n}$ is the measurement matrix, and $y \in \mathbb{R}^m$ is the measurement vector. The key insight in the pioneering work on CS [8, 9] is to replace ℓ_0 by ℓ_1 for the (P_{ℓ_0}) problem due to nonconvexity and combinatorial effect. In [10], it is the basis pursuit problem

$$(\text{BP}){:}\min_{x_0} \|x_0\|_{\ell_1} \quad \text{s.t. } Ax = y. \tag{1.2}$$

Hereafter, numerous researchers developed various computational algorithms for sparse signal recovery [11, 12]. Generally, there are three major classes of computational algorithms: convex relaxation, bayesian inference, and greedy pursuit. Convex relaxation replaces the combinatorial problem (P_{ℓ_0}) with a convex optimization problem (BP), such as basis pursuit [10], basis pursuit denoising [13], the least absolute shrinkage and selection operator (LASSO) [14], and least angle regression (LARS) [15]. Bayesian inference derives the maximum a posteriori estimator from a sparse prior model, such as sparse bayesian learning (SBL) [16, 17] and bayesian compressive sensing (BCS) [18, 19]. Another popular approach is to use greedy algorithms which iteratively refine a sparse solution by successively selecting one or more elements. These algorithms include matching pursuit (MP) [20], orthogonal matching pursuit (OMP) [21–23], subspace pursuit (SP) [24], compressive sampling matching pursuit (CoSaMP) [25], and iterative thresholding algorithms [26, 27].

Iterative hard thresholding (IHT) [27] is a simple and powerful approach for sparse recovery. Recently, an alternative algorithm has been present to alleviate the convergence speed issue of IHT in [28, 29]. Qiu and Dogandzic [28, 29] derived an expectation conditional maximization either (ECME) iteration from a probabilistic framework based on the Gaussian-distributed signals with unknown variance and proposed an acceleration method, termed double overrelaxation (DORE) thresholding scheme, to improve the convergence speed of the ECME algorithm. In addition, Qiu and Dogandzic [28, 29] further proposed an automatic double overrelaxation (ADORE) thresholding method for conditions that the underlying sparsity level is not available. As in study [30], Wang and Yin presented an iterative support detection (ISD) algorithm to refine the failed reconstructions by thresholding the solution of a truncated (BP) problem.

Inspired by the theoretical and empirical evidence of favorable recovery performance of ECME [29] and ISD [30], we combine ECME [29] and ISD [30] to devise novel sparse signal recovery methods, dubbed as ECME thresholding pursuits (EMTP). EMTP detects a support set I using an ECME iteration and estimates the reconstructed signal by solving a truncated least-squares problem on the support set I, and it iterates these two steps for a small number of times. We present two effective support detection strategies (hard thresholding and dynamic thresholding) for the sparse signals with components having a fast-decaying distribution of nonzeros (called fast decaying signals in [30]), which include sparse Gaussian signals, sparse Laplacian signals, and certain power-law decaying signals.

This paper considers the iterative greedy algorithms and abstracts them into two types [31], One-Stage Thresholding (OST) and Two-Stage Thresholding (TST), as discussed in

Sections 2.1 and 2.2. Then, we review the initial work of Iterative Support Detection (ISD) in Section 2.3. In Section 3, we describe the proposed approaches. After that, Section 4 details the experimental results. Finally, we conclude this paper in Section 5.

1.1. Notation

We introduce the notation used in this paper.

(i) $x^{(t)}$: the algorithms described in this paper are iterative and the reconstructed signal x in current iteration t is denoted as $x^{(t)}$. The same convention is used for other vectors and matrices.

(ii) I, A_I: index set I, the matrix A_I denotes the submatrix of A containing only those columns of A with indexes in I. The same convention is used for vectors.

(iii) $[1,n] \setminus I$: the complement of set I in set $\{1,2,\ldots,n\}$.

(iv) $\operatorname{supp}(x)$: the support set of a vector x, that is, the index set corresponding to the nonzeros of x, $\operatorname{supp}(x) = \{i : x_i \neq 0\}$.

(v) $H_k(x)$: the hard thresholding that sets all but the largest in magnitude k elements of a vector x to zero.

(vi) $|x|, \|x\|_{\ell_p}, x^T$: the absolute value, ℓ_p norm, and transpose of a vector x, respectively.

(vii) $A^\dagger$: the Moore-Penrose pseudoinverse of matrix $A \in \mathbb{R}^{m\times n}$. $A^\dagger = A^T(AA^T)^{-1}$ for $m \leq n$; $A^\dagger = (A^TA)^{-1}A^T$ for $m \geq n$.

2. Related Works

2.1. One-Stage Thresholding

Qiu and Dogandzic derived an expectation conditional maximization either (ECME) iteration based on a probabilistic framework [29]

$$x^{(t)} = H_k\left(x^{(t-1)} + A^T\left(AA^T\right)^{-1}\left(y - Ax^{(t-1)}\right)\right). \tag{2.1}$$

Note that ECME iteration reduces to IHT step when the measurement matrix A has orthonormal rows (i.e., (AA^T) is the identity matrix). These one-stage thresholding algorithms (e.g., IHT [27] and ECME [29]) are guaranteed to recover sparse signals and converge with limited iterations. However, OST is not the empirical choice for practical applications due to slow convergence. To this end, Qiu and Dogandzic proposed an acceleration method, termed double overrelaxation (DORE) thresholding scheme [28, 29], to improve the convergence speed of the ECME algorithm. DORE utilizes two overrelaxation steps:

$$\begin{aligned} x_1^{(t)} &= x^{(t)} + \alpha_1\left(x^{(t)} - x^{(t-1)}\right), \\ x_2^{(t)} &= x_1^{(t)} + \alpha_2\left(x_1^{(t)} - x^{(t-2)}\right), \end{aligned} \tag{2.2}$$

where α_1, α_2 are the line search parameters. Finally, an additional hard thresholding step

$$x^{(t)} = H_k\left(x_2^{(t)}\right) \tag{2.3}$$

ensures that the resulting signal is guaranteed to be k-sparse. In addition, Qiu and Dogandzic further presented an automatic double overrelaxation (ADORE) thresholding method for conditions that the underlying sparsity level is not available.

2.2. Two-Stage Thresholding

The algorithms described in this paper fall into the category of a general family of iterative greedy pursuit algorithms. Following [31], we adopt the name "Two-Stage Thresholding" (TST). Considering a CS recovery problem, the sparse recovery algorithms aim to detect the support set I and approximate y using

$$y = A_I x_I. \tag{2.4}$$

Starting with initial solution $x = 0$, TST iterates between 2 main steps:

Step 1 (support detection). Detect the support set I of the signal x, that is, select atoms of measurement matrix A which have been used to generate y; in other words, determine active atoms in sparse representation of a signal x. In some literature, this step also is called basis selection or atom selection.

Step 2 (signal estimation). Update the signal x using the least-squares solution on the detected support set I. $x_I = \arg\min_z\{\|y - A_I z\|_2^2, \operatorname{supp}(z) \subseteq I\}$, $x_{[1,n]\setminus I} = 0$.

Many algorithms (e.g., Orthogonal Matching Pursuit (OMP) [23], Subapace Pursuit (SP) [24], Compressed Sensing Matching Pursuit (CoSaMP) [25], and Gradient Pursuits (GP) [32]) developed various approaches for Step 1 or Step 2.

2.3. Iterative Support Detection

Considering the failed reconstructions of BP, Wang and Yin [30] proposed an algorithmic framework to refine the BP constructions, called iterative support detection (ISD). ISD alternates between two steps: support detection and signal reconstruction. Initialize the detected support $I = \emptyset$ and set the iteration number $t = 0$; ISD iteratively calls the following steps:

(1) signal reconstruction:
solve the truncated BP with $T = I^C$:

$$x^{(t)} = \arg\min_x \|x_T\|_1 \quad \text{s.t. } y = Ax; \tag{2.5}$$

(2) support detection:
detect support set I using $x^{(t)}$ as the reference.

The reliability of ISD relies on the support detection. Wang and Yin devised serval detection strategies for the sparse signals with components having a fast decaying distribution of nonzero components (called fast decaying signals [30]). One of the detection strategies is based on thresholding (we use ISD defined by (2.6) to refer the implementation algorithm in the following context):

$$I^{(t)} = \left\{ i : \left| x_i^{(t)} \right| > \beta^t \max \left| x^{(t)} \right| \right\}, \quad \beta \in (0,1). \tag{2.6}$$

3. ECME Thresholding Pursuits

In this section, we describe our proposed approaches, named ECME Thresholding Pursuits (EMTP), that combine OST and TST using the heuristic idea of ISD. The detailed description of the proposed algorithms are presented as follows

Step 1 (initialization). We have the following.

Initialize the reconstruction signal $x^{(0)} = 0$,

initialize residual signal $r = y$,

initialize support set $I^{(0)} = \emptyset$,

set the iteration counter $t = 1$.

Step 2 (support detection). We have the following.

Update signal approximation:

$$x^{(t)} = x^{(t-1)} + A^{\dagger} r^{(t-1)}. \tag{3.1}$$

Detect the support set I:

strategy 1: hard thresholding $I^{(t)} = \mathrm{supp}(H_k(x^{(t)}))$;

strategy 2: dynamic thresholding $I^{(t)} = \{i : |x_i^{(t)}| > \beta^t \max |x^{(t)}|\}$.

Step 3 (signal estimation). We have the following.

Estimate the signal:

$$\begin{aligned} x_{I^{(t)}}^{(t)} &= A_{I^{(t)}}^{\dagger} y, \\ x_{[1,n]\setminus I^{(t)}}^{(t)} &= 0. \end{aligned} \tag{3.2}$$

Update the residual:

$$r^{(t)} = y - Ax^{(t)}. \tag{3.3}$$

```
Input: Measurement matrix $A$, measurements $y$, sparsity level $k$
Output: The reconstructed signal $x$
Initialization:
$t = 1$ //iteration number
$x^{(0)} = \mathbf{0}$ //initial signal
$r^{(0)} = y$ //initial residual
$I^{(0)} = \emptyset$ //initial support set
while halting criterion false do
    $x^{(t)} = H_k(x^{(t-1)} + A^{\dagger} r^{(t-1)})$
    $I^{(t)} = \{i : x_i^{(t)} \neq 0\}$
    $x_{I^{(t)}}^{(t)} = A_{I^{(t)}}^{\dagger} y$
    $x_{[1,n]\backslash I^{(t)}}^{(t)} = 0$
    $r^{(t)} = y - Ax^{(t)}$
    $t = t + 1$
end while
return $x$
```

Algorithm 1: EMTP-k algorithm.

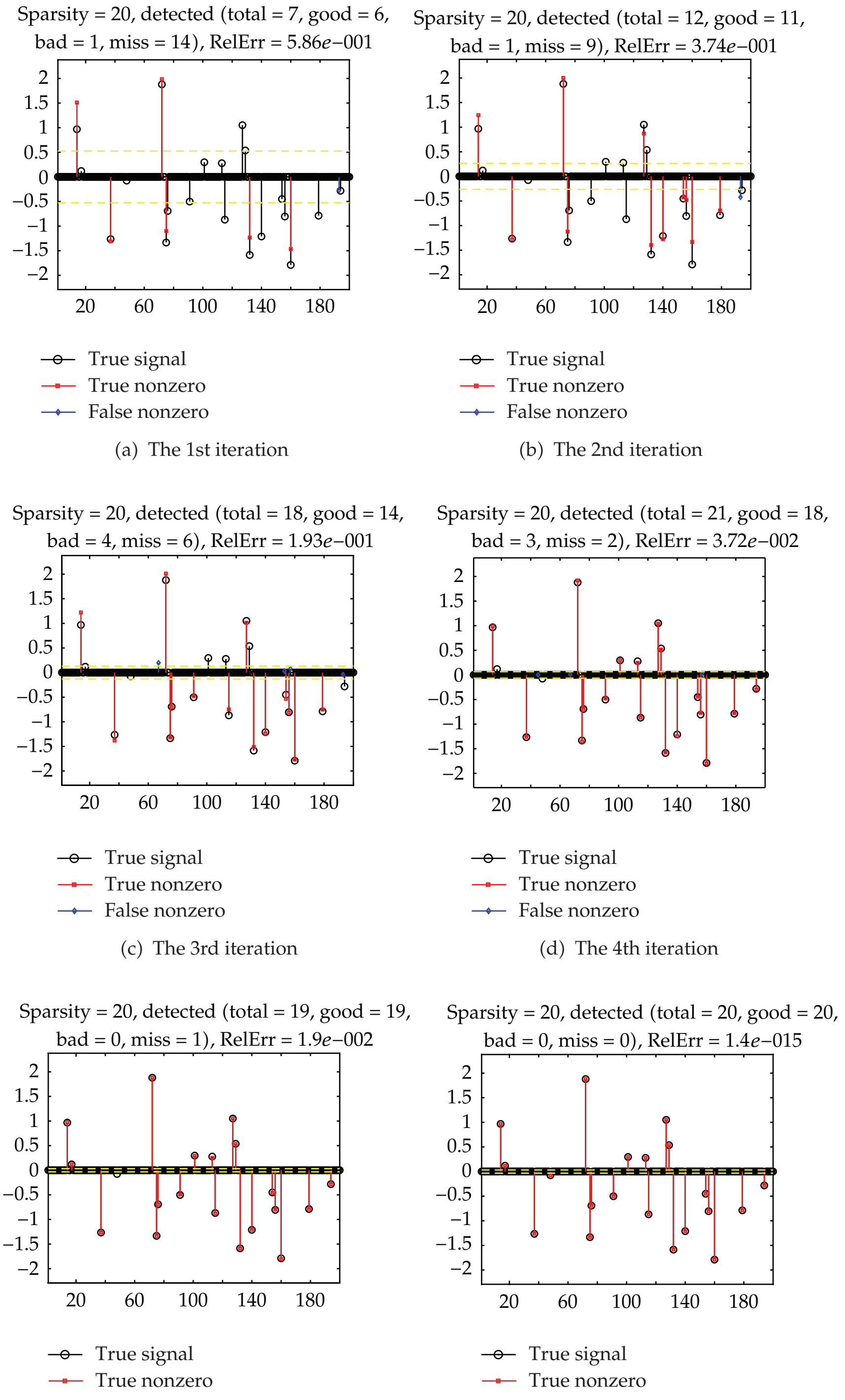
```
Input: Measurement matrix $A$, measurements $y$, thresholding parameter $\beta$
Output: The reconstructed signal $x$
Initialization:
$t = 1$ //iteration number
$x^{(0)} = \mathbf{0}$ //initial signal
$r^{(0)} = y$ //initial residual
$I^{(0)} = \emptyset$ //initial support set
while halting criterion false do
    $x^{(t)} = x^{(t-1)} + A^{\dagger} r^{(t-1)}$
    $I^{(t)} = \{i : |x_i^{(t)}| > \beta^t \max |x|\}$
    $x_{I^{(t)}}^{(t)} = A_{I^{(t)}}^{\dagger} y$
    $x_{[1,n]\backslash I^{(t)}}^{(t)} = 0$
    $r^{(t)} = y - Ax^{(t)}$
    $t = t + 1$
end while
return $x$
```

Algorithm 2: EMTP-β algorithm.

Step 4 (halting). Check the stopping condition and, if it is not yet time to stop, increment $t = t + 1$ and return to Step 2. If it is time to stop, the recovered signal x has nonzero entries in support set $I^{(t)}$ and corresponding support vector lies in $x_{I^{(t)}}^{(t)}$.

We present two thresholding proposals and corresponding algorithm EMTP-k (shown as Algorithm 1) and EMTP-β (shown as Algorithm 2), where $\beta \in (0, 1)$ is the thresholding parameter.

Remarks. (1) EMTP updates the leading elements of $x^{(t)}$ using a least-squares solution on the support set I. However, DORE updates all the elements of $x^{(t)}$ using double overrelaxation



Sparsity = 20, detected (total = 7, good = 6, bad = 1, miss = 14), RelErr = 5.86e−001

(a) The 1st iteration

Sparsity = 20, detected (total = 12, good = 11, bad = 1, miss = 9), RelErr = 3.74e−001

(b) The 2nd iteration

Sparsity = 20, detected (total = 18, good = 14, bad = 4, miss = 6), RelErr = 1.93e−001

(c) The 3rd iteration

Sparsity = 20, detected (total = 21, good = 18, bad = 3, miss = 2), RelErr = 3.72e−002

(d) The 4th iteration

Sparsity = 20, detected (total = 19, good = 19, bad = 0, miss = 1), RelErr = 1.9e−002

(e) The 5th iteration

Sparsity = 20, detected (total = 20, good = 20, bad = 0, miss = 0), RelErr = 1.4e−015

(f) The 6th iteration

Figure 1: An EMTP-β demo that recovers a Gaussian-distributed sparse vector with $n = 200, m = 80,\ k = 20,\ \beta = 0.5$.

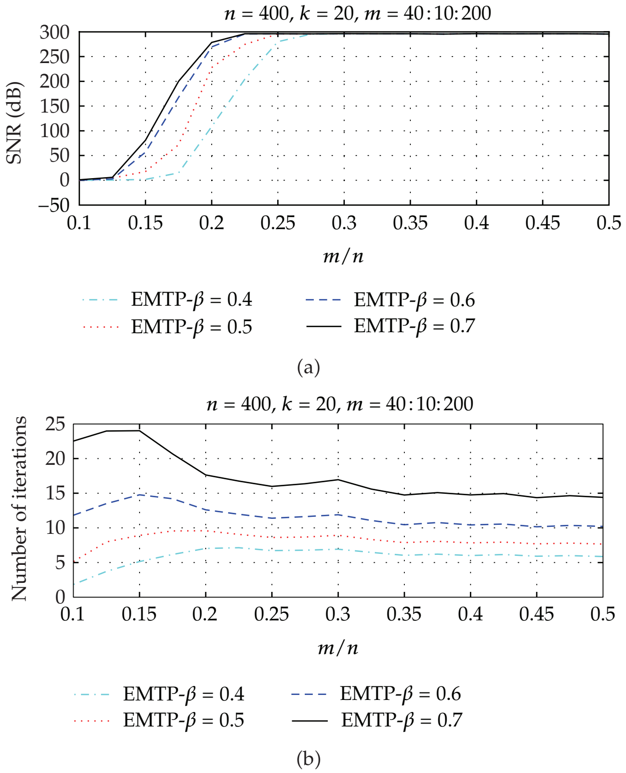
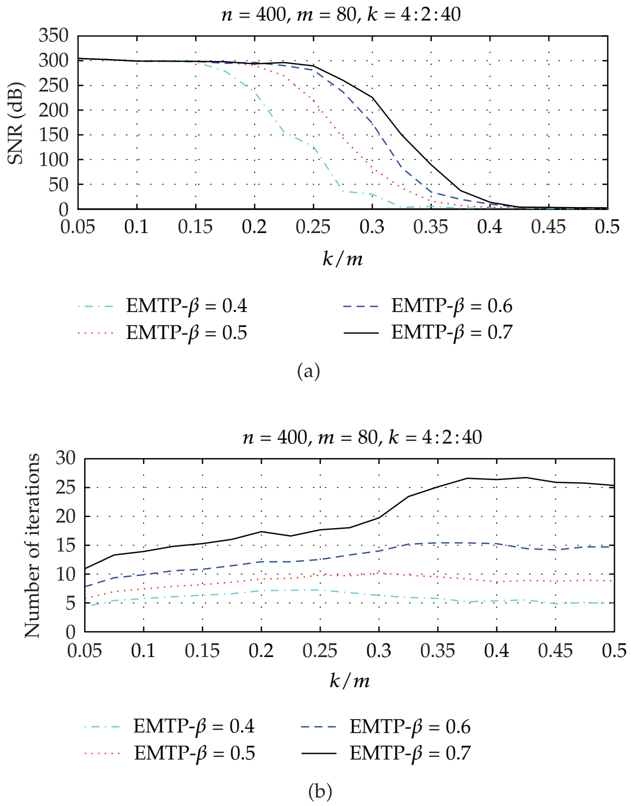


Figure 2: Influence of thresholding parameter β for sparse Gaussian signals with $n = 400$, $k = 20$: comparisons in terms of SNR (dB) and number of iterations.

steps. Finally, DORE needs a hard thresholding to ensure the new approximation is k-sparse. EMTP-β demands no prior knowledge about the underlying sparsity level k.

(2) EMTP is a special case of TST and utilizes OST in the support detection stage. It is different from OMP, SP, and CoSaMP. When the ECME iteration reduces to IHT step, that is, the reference for support detection $x^{(t)} = x^{(t-1)} + A^T r^{(t-1)}$ is the gradient descent step for least-squares, OMP, SP and CoSaMP use the correlation between the residual signal and the atoms of the measurement matrix A (i.e., $A^T r^{(t-1)}$) to detect support set. $A^T r^{(t-1)}$ is the negative gradient for least squares. It is clear that ECME iteration provides the estimation for the underlying sparse signal. We can employ the heuristic idea of ISD to devise various support detection strategies depending on the underlying sparse signal distribution. The dynamic thresholding method can be performed without the sparsity level k. EMTP directly detects the support of the underlying sparse signal by referencing the ECME iteration while OMP, SP, and CoSaMP augment the support by picking out the leading values of the negative gradient. OMP each step spots one index into the support, so it requires more iterations than EMTP. SP and CoSaMP spot several indexes into the support, so they need an additional step (i.e., orthogonal projection) to make sure the recovered signal is k-sparse. Like ISD, EMTP has an appealing self-correction capacity. An EMTP demo is presented in Figure 1.

(3) Like other greedy pursuits such as Subapace Pursuit [24] and CoSaMP [25], EMTP-k fixes the cardinality of support set I and removes previous false detections. However, EMTP-β refines the support set I which is not necessarily increasing and nested over the iterations.



(a)

(b)

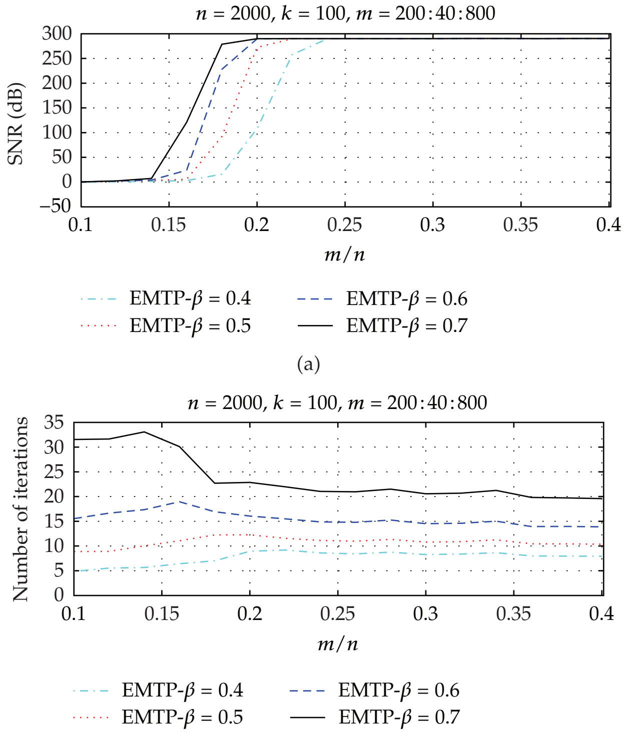
Figure 3: Influence of thresholding parameter β for sparse Gaussian signals with $n = 400$, $m = 80$: comparisons in terms of SNR (dB) and number of iterations.

(4) The dynamic thresholding strategy used in EMTP-β is inspired by ISD [30]. It finds significant nonzeros by comparing a threshold rather than maintaining fixed number (k) of items. It is appealing for the conditions that the underlying sparsity level k is not available.

(5) EMTP resembles ISD [30]. EMTP and ISD have the same idea in support detection step with iterative behavior. However, the support detection step is based on different sparse recovery methods, ECME and BP, respectively. EMTP updates the reconstruction signal using a least-squares solution on the detected support set I. However, ISD iteratively refines the BP solution on the complement of the detected support set I.

(6) EMTP-β with large β obtains high-quality reconstruction from a small number of measurements. However, because support set I grows slowly, EMTP-β takes a larger number of iterations (to be discussed in Section 4). EMTP-k wrongly detected elements can often be pruned out in later iterations.

(7) Like ISD (as discussed in [30]), EMTP-β only performs well for fast decaying signals. It does not work on sparse signals that decay slowly or have no decay at all (e.g., trinary and binary sparse signals). EMTP-k performs worse for the sparse signals that nonzero elements have similar magnitude [31, 33]. However, we can apply EMTP to non-fast-decaying signals via linear or nonlinear premapping [34, 35].



(a)

(b)

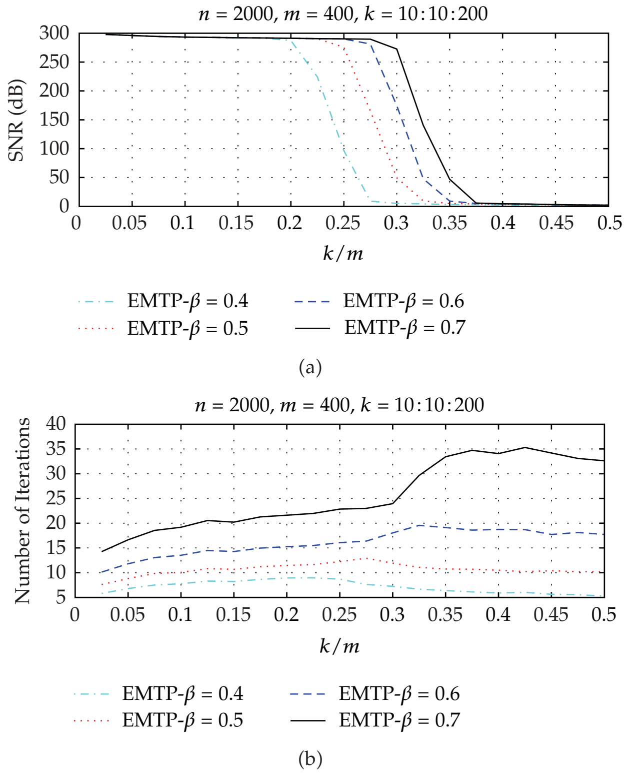
Figure 4: Influence of thresholding parameter β for sparse Gaussian signals with $n = 2000$, $k = 100$: comparisons in terms of SNR (dB) and number of iterations.

3.1. *Complexity Analysis*

Like DORE algorithm, the basic operation is matrix-vector multiplication and sorting operation. Assume that the pseudoinverse matrix $A^\dagger$ is precomputed (with the cost of $\mathcal{O}(m^2n+mn)$). Line 7 in Algorithms 1 and 2 for updating the ECME iteration requires $\mathcal{O}(mn)$. The computation complexity of thresholding operation (line 8 in Algorithms 1 and 2) is $\mathcal{O}(n\log k)$. Line 9 involves the partial least-square solver, costing $\mathcal{O}(k^2m + km)$. The step for updating residual requires an additional matrix-vector operation, costing $\mathcal{O}(km)$. To summarize, one EMTP iteration costs $\mathcal{O}(mn + n\log k + k^2m + 2km)$, which is significantly less than the complexity of one DORE step (with the cost of $\mathcal{O}(2n\log k + 2m^2n + 3mn)$).

4. Experimental Results

To assess the performance of the proposed approaches, we conduct numerical experiments on computer simulations. All algorithms were implemented and tested in MATLAB v7.6 running on Windows XP with 2.53 GHz Intel Celeron CPU and 2 GB of memory. We compared the proposed approaches with the accelerated ADORE/DORE approaches. The code of ADORE/DORE is available on the authors homepage (http://home.eng.iastate.edu/~ald/DORE.htm). We initialized x with zero sparse signal. The search length of ADORE was set to 1. All results were averaged



(a)

(b)

Figure 5: Influence of thresholding parameter β for sparse Gaussian signals with $n = 2000$, $m = 400$: comparisons in terms of SNR (dB) and number of iterations.

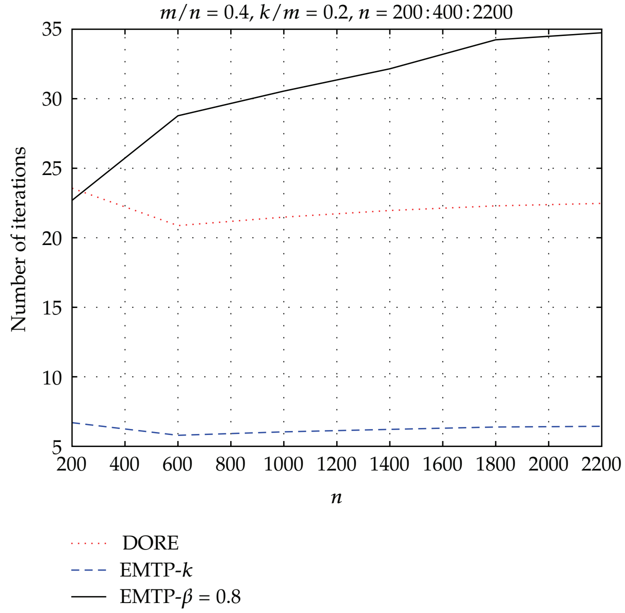


Figure 6: Number of iterations as a function of problem size with fixed ratios of $m/n = 0.4$, $k/m = 0.2$, $n = 200 : 400 : 2200$.

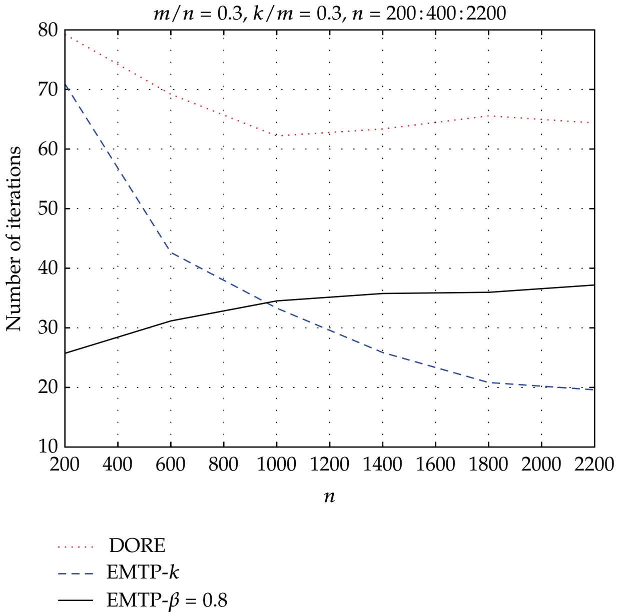


Figure 7: Number of iterations as a function of problem size with fixed ratios of $m/n = 0.3$, $k/m = 0.3$, $n = 200:400:2200$.

over 100 times Monte Carlo problem instances. We used our unoptimized code (http://cs-notes.googlecode.com/files/EMTP.zip). The least-squares solution $x = \operatorname{argmin}\|y - Ax\|_2^2$ was implemented using MATLAB pseudoinverse function by `x = pinv(A)* y`. The MATLAB code was partially adapted from [30, 33, 36]. In all the experiments, the measurement matrix A was generated by uniform spherical ensemble (USE), that is, we generate the measurement matrix by sampling each entry independently from the standard norm distribution and then normalize each column to have unit norm. The MATLAB code is given by

```
A = randn(m,n); A = A./repmat(sqrt(sum(A.^2)),[m 1]).
```

The underlying k-sparse vectors were generated by randomly selecting a support set of size k and each entry in the support set is sampled uniformly from a specific distribution. The sparse signals were generated in MATLAB by

```
x = zeros(n,1); support = randperm(n); x(support(1 : k)) = v;
```

and the nonzeros `v` generated by following code.

The sparse Gaussian signals were generated in MATLAB by

```
v = randn(k,1);
```

The sparse Laplacian signals were generated by

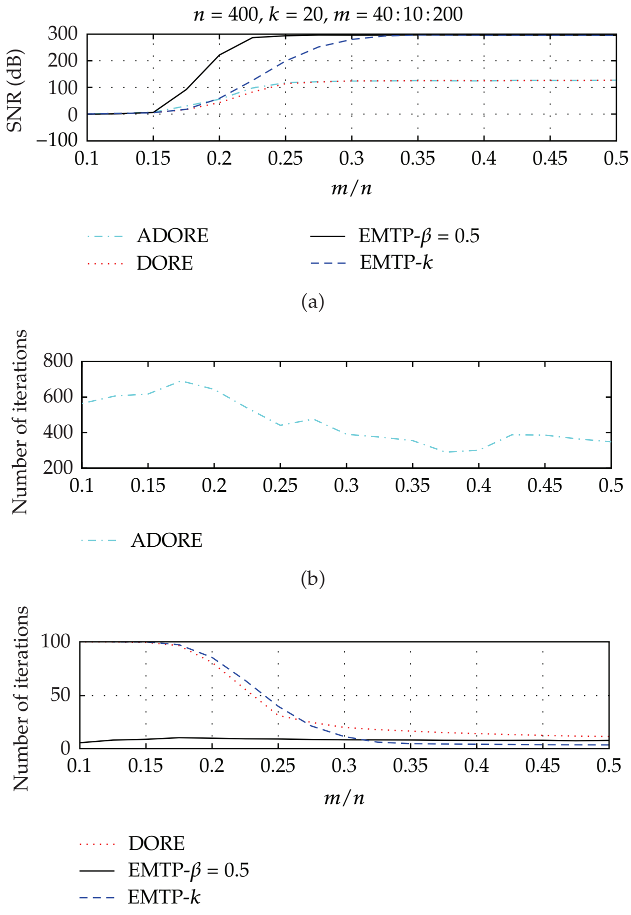
```
z = rand([k,1]);
v = zeros([k,1]);
 in = z <= 0.5;
 ip = z > 0.5;
```

$$\texttt{v(in)} = \frac{1}{\texttt{lambda}} * \texttt{log(2 * z(in));}$$

$$\texttt{v(ip)} = -\frac{1}{\texttt{lambda}} * \texttt{log(2 * (1} - \texttt{z(ip)))}.$$

The power-law decaying signals were generated by

```
v = sign(randn(k,1)).* ((1 : k). \^(−1/lambda))'.
```



(a)

(b)

(c)

Figure 8: Sparse Gaussian signals with $n = 400$, $k = 20$: comparisons in terms of SNR (dB) and number of iterations.

The variable lambda controls the rate of decay. We set `lambda` = 10 and `lambda` = 0.5 for sparse Laplacian signals and power-law decaying signals, respectively.

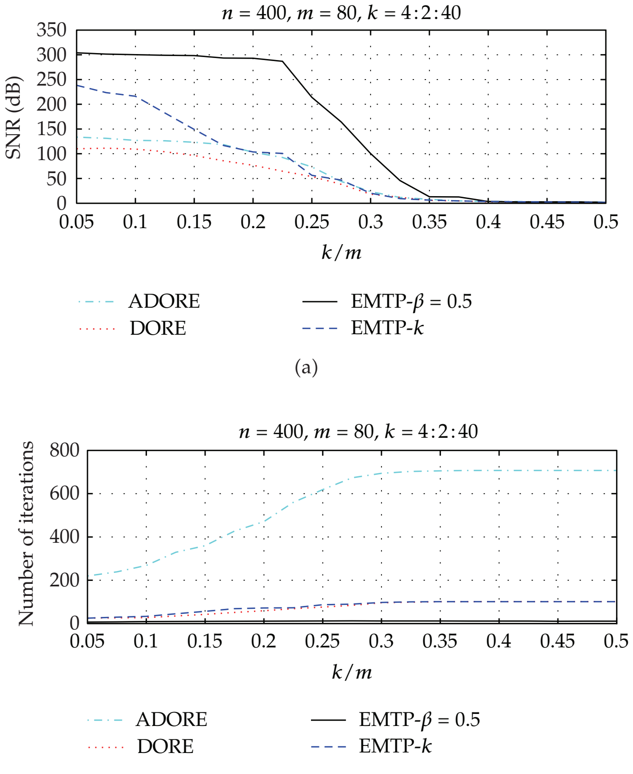
For fair comparison, we stopped iterations once the relative error fell below a certain convergence tolerance or the number of iterations is greater than 100. The convergence tolerance is given by

$$\frac{\|y - r^{(t)}\|_2}{\|y\|_2} \leq 10^{-6}. \tag{4.1}$$

We empirically evaluate reconstruction performance in terms of signal-to-noise ratio (SNR) and number of iterations. The SNR is defined as

$$\mathrm{SNR(dB)} = -20\log_{10}\frac{\|x_0 - x\|_2}{\|x_0\|_2}, \tag{4.2}$$

where x_0 is the true signal and x is the recovered signal.



(a)

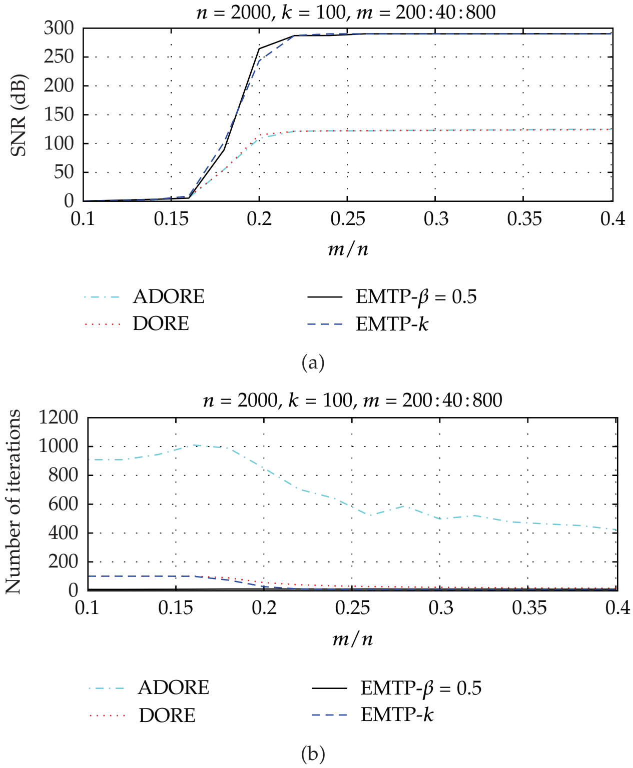
(b)

Figure 9: Sparse Gaussian signals with $n = 400$, $m = 80$: comparisons in terms of SNR (dB) and number of iterations.

4.1. Influence of Thresholding Parameter

We evaluated the influence of thresholding parameter β by varying β from 0.4 to 0.7. First, we fixed signal dimension $n = 400$, the underlying sparsity level $k = 20$, and varied the number of measurements. The plot of influence of thresholding parameter β for sparse Gaussian signals was presented in Figure 2. It is clear that larger thresholding parameter achieves better recovery performance with fewer measurements. However, the cost is more iterations. Large β is particularly competitive when the number of measurements is fairly small. It is worth noting that the number of iterations is smaller than the underlying sparsity level k for exact recovery, especially for large k. As shown in Figure 2, EMTP-β achieves the SNR (dB) around 300, that is, the relative error is almost as low as the double precision. Second, we tested the comparisons by fixing the number of measurements $m = 80$, as shown in Figure 3, respectively. Finally, we enlarged the dimension of signals ($n = 2000$), as shown in Figures 4 and 5. The latter tests come to similar conclusions as the first test.

We fixed the ratios of m/n, k/m and present plots cases for the number of iterations as a function of problem size n. Given sufficient measurements, as shown in Figure 6 with $m/n = 0.4$, $k/m = 0.2$, $n = 200:400:2200$, EMTP-k requires significantly fewest iterations and EMTP-β requires relatively more iterations than DORE. It is clear that the number of iterations for EMTP-k and DORE is stable to the problem size, and EMTP-β needs acceptably a few more iterations with increasing the problem size. For another case, as shown in Figure 7



(a)

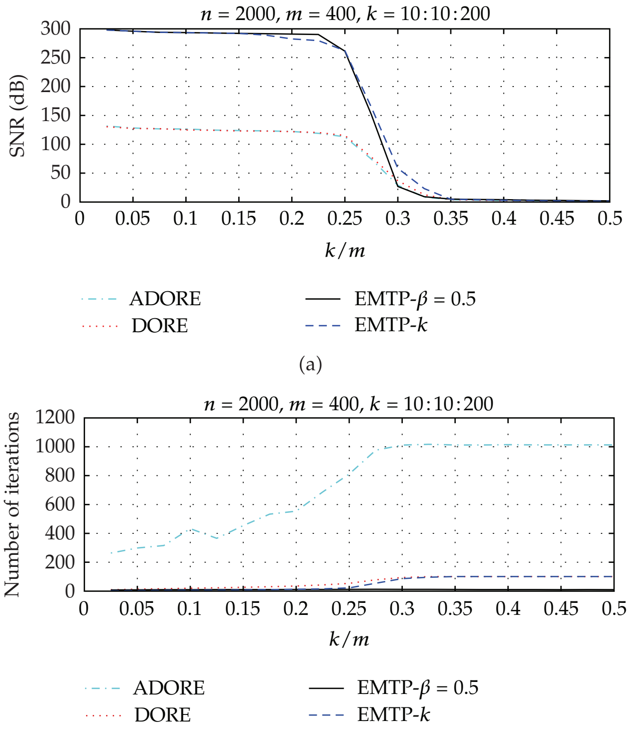
(b)

Figure 10: Sparse Gaussian signals with $n = 2000$, $k = 100$: comparisons in terms of SNR (dB) and number of iterations.

with $m/n = 0.3$, $k/m = 0.3$, $n = 200 : 400 : 2200$, DORE requires more iterations, EMTP-k needs significantly fewer iterations with increasing the problem size, and EMTP-β requires relatively stable iterations.

4.2. Comparisons with the Accelerated OST

We present comparisons in terms of SNR (dB) and number of iterations. We omit the comparisons with ℓ_1 minimization methods and some other OST/TST methods, as they have already been investigated in [31, 33]. First, we fixed signal dimension $n = 400$, the underlying sparsity level $k = 20$, and varied the number of measurements. The result was plotted in Figure 8. Next, we tested the comparisons by fixing the number of measurements $m = 80$, as shown in Figure 9. Finally, the test set used larger dimension signals ($n = 2000$). The corresponding results were depicted in Figures 10 and 11. Figures 10, and 11 show the average SNR (dB) (a) and number of iterations (b) for each indeterminacy level m/n. EMTP achieves significantly larger SNR (dB) than ADORE/DORE. For exact recovery, EMTP obtains the relative error almost as low as the double precision. EMTP-β constantly needs small number of iterations and EMTP-k needs the smallest number of iterations for exact recovery. However, ADORE needs hundreds of iterations. Figure 8 was zoomed in for better illustration. Figures 8, 9, 10 and 11 show the average SNR (dB) and number of iterations for each sparsity level k/m. For larger-dimension signals ($n = 2000$), the results are depicted in



(a)

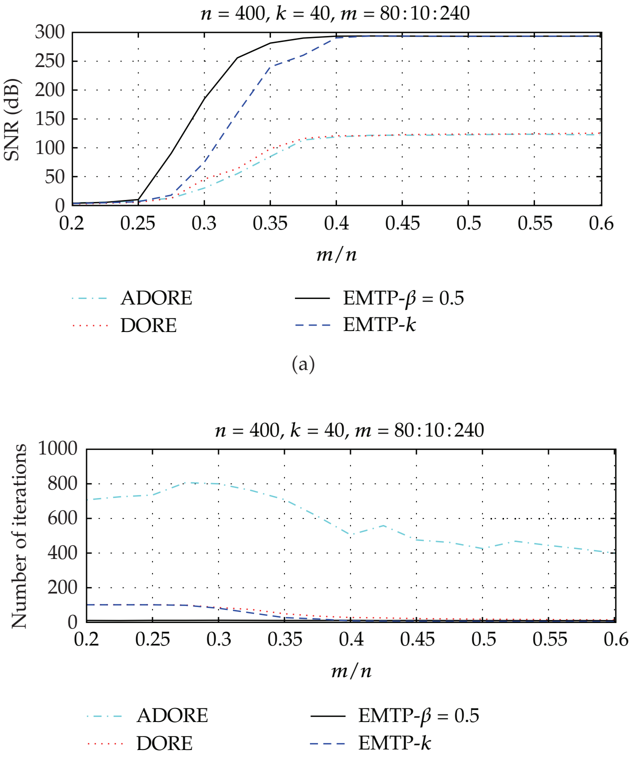
(b)

Figure 11: Sparse Gaussian signals with $n = 2000$, $m = 400$: comparisons in terms of SNR (dB) and number of iterations.

Figures 15 and 16. The latter tests come to similar conclusions as the first test. These figures do not fully capture the performance of EMTP-β since we only set $\beta = 0.5$; however EMTP-β achieved almost superior performance than ADORE/DORE.

4.3. Performance for Other Fast-Decaying Signals

As discussed in [30, 33], thresholding-based sparse recovery methods achieve superior performance only if the nonzero elements of signals have a fast-decaying distribution. We present empirical studies on other fast-decaying signals. Sparse Laplacian signals containing $k = 20$ nonzero elements for dimension $n = 400$ were tested and the corresponding result was plotted in Figure 12. For larger dimension signals ($n = 2000$), the results are shown in Figure 13. Power-law decaying signals containing $k = 20$ nonzero elements for dimension $n = 400$ were tested and the corresponding result was plotted in Figure 14. For larger-dimension signals ($n = 2000$), the results are depicted in Figure 15. ADORE/DORE were derived from a probabilistic framework based on Gaussian distribution. Surprisingly, they can also work for fast-decaying signals, as depicted in Figures 12, 13, 14, and 15. The conclusions for sparse Gaussian signals also can be generalized for sparse Laplacian signals and power-law decaying signals. Power-law decaying signals for low indeterminacy level m/n make an exception that EMTP-β achieves inferior performance.



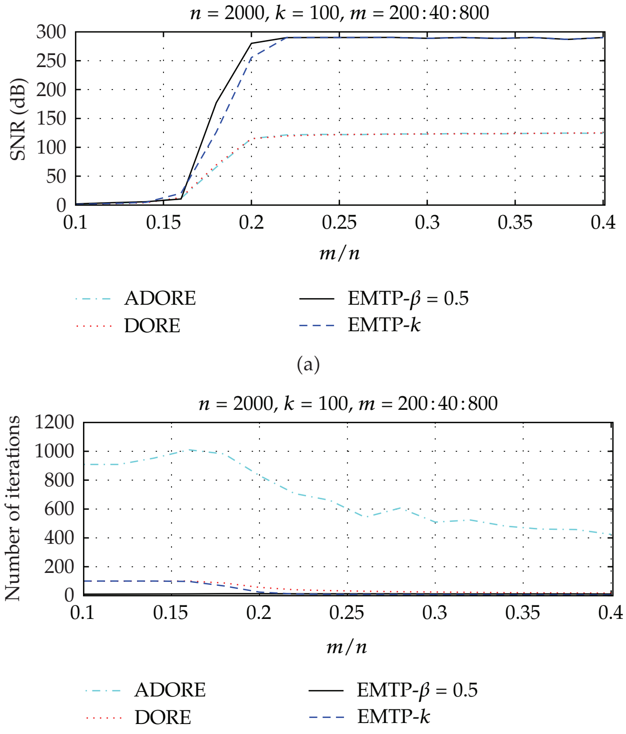
(a)

(b)

Figure 12: Sparse Laplacian signals with $n = 400$, $k = 20$: comparisons in terms of SNR (dB) and number of iterations.

4.4. Phase Transitions

Following [31, 33], we present numerical comparisons in terms of phase transitions. For the sparse signal recovery from compressed measurements, the indeterminacy $\delta = m/n$ defines the undersampling of a vector in compressed measurements. Let $\rho = k/m$ be a normalized measure of the sparsity. We fix the problem size $n = 400$ and test 100 Monte Carlo realizations of each sparse vector at each pair of (ρ, δ). We test a grid of 16×16 linearly spaced (ρ, δ) combinations with ρ, δ varying from 0.05 to 0.5 in 16 steps. For each sparsity and indeterminacy pair, then, we find the probability of exact recovery with defining exact recovery when $\|x_0 - x\|_2/\|x\|_2 < 0.01$. For each problem indeterminacy, we interpolate the results over all sparsities to find where successful recovery occurs with a probability of 0.5. As a result, we obtain a phase transition plot showing the boundary above which most recoveries fail and below which most recoveries succeed. Figure 16 presents the recovery rates of DORE, SP, and EMTP-β for Gaussian-distributed sparse vectors as a function of problem sparsity and four problem indeterminacies from the thickest to thinnest lines. In Figure 16, we observe EMTP-β outperforms SP, and DORE in recovery rates. Figure 17 shows the phase transitions of DORE, SP and EMTP-β for compressively sensed sparse vectors sampled from Gaussian distribution. Figure 17 indicates that these transitions obey the following hierarchy in recovery performance: EMTP-β > SP > DORE.



(a)

(b)

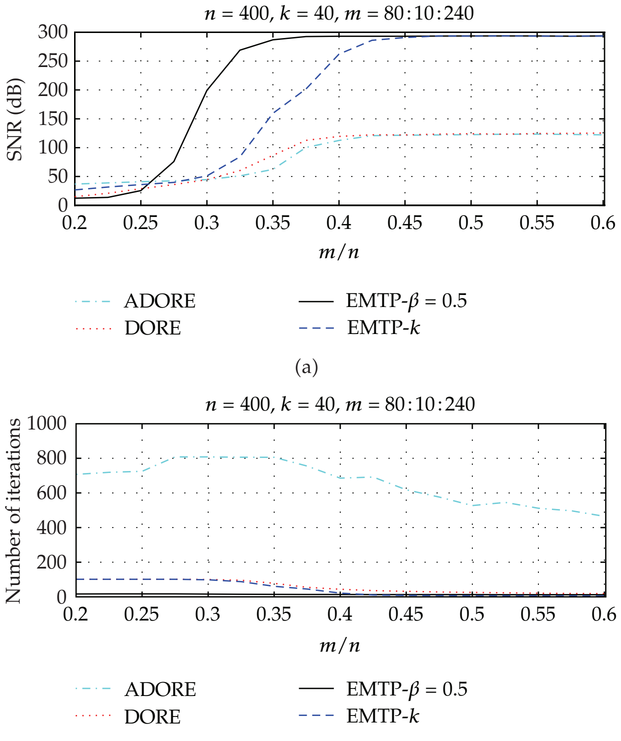
Figure 13: Sparse Laplacian signals with $n = 2000$, $k = 100$: comparisons in terms of SNR (dB) and number of iterations.

4.5. Summary

To sum up, we have compared EMTP with ADORE/DORE. EMTP has significant advantages over the accelerated OST in terms of SNR and number of iterations. EMTP can significantly reduce the number of iterations required and achieves significantly higher SNR. For low indeterminacy level m/n, EMTP requires fewer measurements. EMTP-β can work with no prior knowledge of the underlying sparsity level k yet achieves recovery performance better than ADORE. Furthermore, we compared EMTP with the state-of-the-art greedy algorithm SP in terms of phase transitions. Among all methods, EMTP-β appeared to be the best.

5. Conclusions and Future Work

In this paper, we propose ECME thresholding pursuits (EMTP) for sparse signal recovery. EMTP detects a support set I using the ECME iteration and estimates the reconstructed signal by solving a truncated least-squares problem on the support set I, and it iterates these two steps for a small number of times. We present two effective support detection strategies (hard thresholding and dynamic thresholding) for the sparse signals with components having a fast-decaying distribution of nonzero components. The experimental studies are presented to demonstrate that EMTP offers an attractive alternative to state-of-the-art algorithms for sparse signal recovery. EMTP can significantly reduce the number of iterations required. We



(a)

(b)

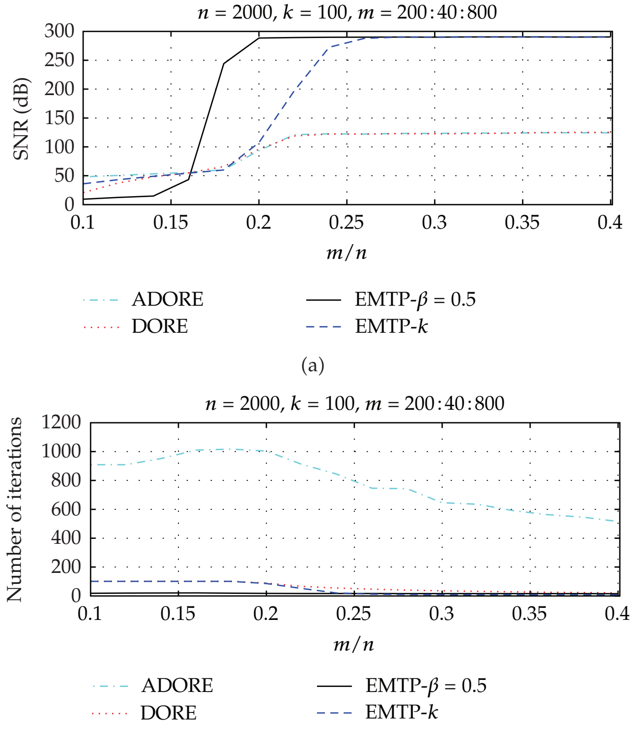
Figure 14: Power-law decaying signals with $n = 400$, $k = 20$: comparisons in terms of SNR (dB) and number of iterations.

then turn to the problem of reducing the computational cost of each iteration, especially for large scale applications. Future research includes three directions.

(1) EMTP requires the precomputation and storage of the pseudoinverse matrix $A^{\dagger}$. So it is the advisable choice that replacing ECME iteration by IHT step in the support detection stage. IHT does not require the computation and storage of the pseudoinverse matrix $A^{\dagger}$, which leads to significant advantage for large scale applications.

(2) In signal estimation stage, that is, update reconstructed signal by solving least-squares problem, EMTP uses the orthogonal projection, that is, by calculating $x_I = A_I^{\dagger} y$ where $A_I^{\dagger}$ is the pseudoinverse of A_I. We will approximate the orthogonal projection efficiently by gradient pursuits [32].

(3) This paper devotes efforts to devise computational algorithms which are experimentally reproducible and provides empirical studies as useful guidelines for practical applications. Further, future investigations will test other OST methods as the reference and sophisticated thresholding methods for support detection.



(a)

(b)

Figure 15: Power-law decaying signals with $n = 2000$, $k = 100$: comparisons in terms of SNR (dB) and number of iterations.

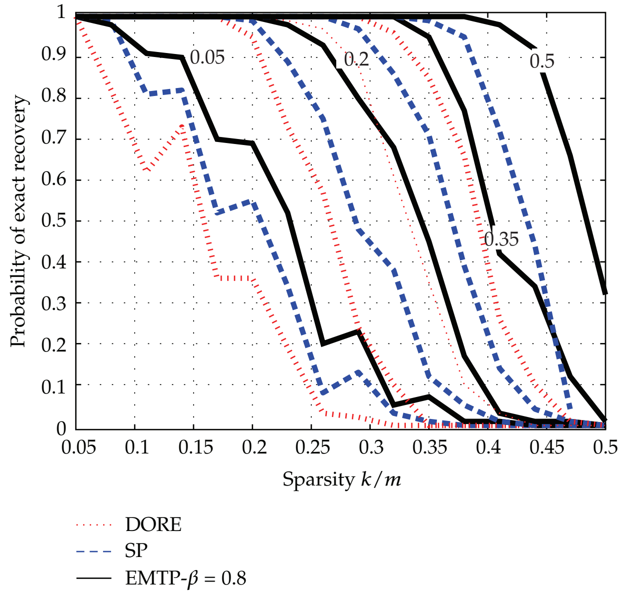


Figure 16: Probability of exact recovery for DORE, SP, and EMTP as a function of problem sparsity and four problem indeterminacies from the thickest to thinnest lines: $\delta = m/n = 0.05, 0.2, 0.35, 0.5$.

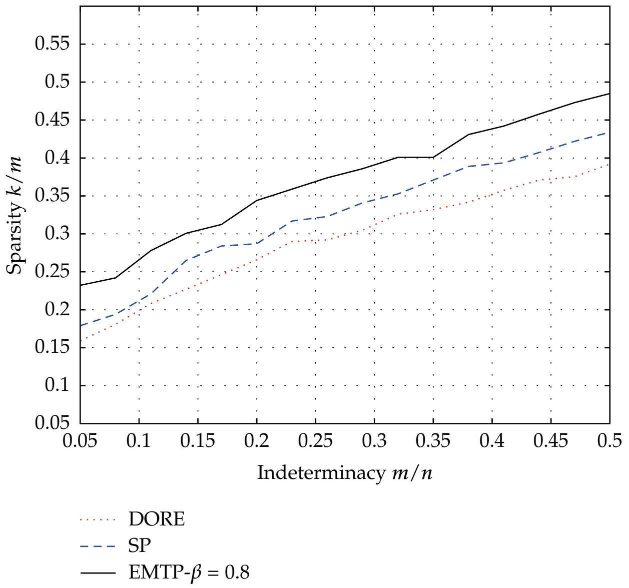


Figure 17: Comparison of phase transitions of DORE, SP, and EMTP for Gaussian-distributed sparse vectors.

Acknowledgment

This work was supported by the National Science Foundation of China under Grant no. 61074167, 61170126, and the Scientific Research Foundation for Advanced Talents by the Jiangsu University, no. 12JDG050.

References

[1] M. Elad, M. A. T. Figueiredo, and Y. Ma, "On the role of sparse and redundant representations in image processing," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 972–982, 2010.

[2] M. J. Fadili, J. L. Starck, J. Bobin, and Y. Moudden, "Image decomposition and separation using sparse representations: an overview," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 983–994, 2010.

[3] M. D. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M. E. Davies, "Sparse representations in audio and music: from coding to source separation," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 995–1005, 2010.

[4] L. C. Potter, E. Ertin, J. T. Parker, and M. Çetin, "Sparsity and compressed sensing in radar imaging," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1006–1020, 2010.

[5] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031–1044, 2010.

[6] A. Y. Yang, M. Gastpar, R. Bajcsy, and S. S. Sastry, "Distributed sensor perception via sparse representation," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1077–1088, 2010.

[7] R. Robucci, J. D. Gray, L. K. Chiu, J. Romberg, and P. Hasler, "Compressive sensing on a CMOS separable-transform image sensor," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1089–1101, 2010.

[8] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[9] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[10] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[11] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Review*, vol. 51, no. 1, pp. 34–81, 2009.

[12] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 948–958, 2010.

[13] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, vol. 24, no. 2, pp. 227–234, 1995.

[14] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[15] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[16] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, no. 3, pp. 211–244, 2001.

[17] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153–2164, 2004.

[18] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2346–2356, 2008.

[19] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 53–63, 2010.

[20] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[21] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proceedings of the 27th Asilomar Conference on Signals, Systems & Computers*, vol. 1, pp. 40–44, Pacific Grove, Calif , USA, November 1993.

[22] J. A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.

[23] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[24] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.

[25] D. Needell and J. A. Tropp, "CoSaMP: iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[26] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.

[27] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.

[28] K. Qiu and A. Dogandzic, "Double overrelaxation thresholding methods for sparse signal reconstruction," in *Proceedings of the 44th Annual Conference on Information Sciences and Systems (CISS '10)*, Princeton, NJ, USA, March 2010.

[29] K. Qiu and A. Dogandzic, "Sparse signal reconstruction via ECME hard thresholding," *IEEE Transactions on Signal Processing*, vol. 60, no. 9, pp. 4551–4569, 2012.

[30] Y. Wang and W. Yin, "Sparse signal reconstruction via iterative support detection," *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 462–491, 2010.

[31] A. Maleki and D. L. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE Journal on Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 330–341, 2010.

[32] T. Blumensath and M. E. Davies, "Gradient pursuits," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2370–2382, 2008.

[33] B. Sturm, "A study on sparse vector distributions and recovery from compressed sensing," http://arxiv.org/abs/1103.6246.

[34] X. Zhang, J. Wen, Y. Han, and J. Villasenor, "An improved compressive sensing reconstruction algorithm using linear/non-linear mapping," in *Proceedings of the Information Theory and Applications Workshop (ITA '11)*, pp. 146–152, San Diego, Calif, USA, February 2011.

[35] X. Zhang, Z. Chen, J. Wen, J. Ma, Y. Han, and J. Villasenor, "A compressive sensing reconstruction algorithm for trinary and binary sparse signals using pre-mapping," in *Proceedings of the Data Compression Conference (DCC '11)*, pp. 203–212, Snowbird, Utah, USA, March 2011.

[36] V. Stodden, L. Carlin, D. Donoho et al., "SparseLab: seeking sparse solutions to linear systems of equations," Tech. Rep., Stanford University, 2011.