# Report 2

Group No.17

May 3, 2009

## 1 CS in Distributed Sensor Network

## 2 Fast Solution of l1-norm Minimization Problems

# CS in Distributed Sensor Network

SUN LILI

May 3, 2009

## 1 Introduction

The following report tries to clarify how distributed Sensor network benefit form CS. A comprehensive answer of the usage of CS-based encoding in sensor network can be found in [1]. And here, by adding some clarified proof proceedings and unspecified CS relevant content, we try to give a more easy to handle and CS-specific version answer of it.

The sensor network is a wireless network of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibrations, pressure, motion or pollutants, at different locations.

The essential task in many applications of sensor networks is to extract relevant information about the sensed data and deliver it with a desired fidelity to a distant destination, termed as the fusion center(FC).

The design of sensor network is to execute this task with least consumption of network resources(energy and bandwidth).

The system constructed here act less like networks and more like coherent ensembles of sensors by eliminating the need for in-network communications and processing, and requires phase synchronization among nodes. With regards to the role CS plays in the sensor network, we'd like to put the conclusion first to give a general idea of the compressive sensing usage in sensor networks.

1. If one has enough prior knowledge about senor network data. using that prior knowledge in encoding the data is superior to CS-based encoding.

2. If one doesn't have enough prior knowledge about the underlying data, using CS-based encoding can be strictly superior to using a mismatched encoding mechanism.

CS-based encoding gives one the ability to design universal encoding mechanisms that need not be change with the underlying data(nonadaptive), but the price that one has to pay is in terms of the performance as compared to data-specific encoding. "universality vs prior knowledge trade-off".

## 2 Sensing,Procession and Communication

Rather than encoding and transmitting samples from individual sensors, an alternate encoding paradigm is based on the projections of samples from many sensors onto appropriate spatial basis functions.

This joint source-channel communication architecture is an energy efficient method for communicating such projections to the FC. The projections are communicated in a phase-coherent fashion over the network-to-FC multiple-access channel.

One can choose to acquire samples in the domain of any basis that is particularly well-suited to the spatial structure of the signal field being sensed.

1. If one has reasonable prior knowledge about the signal, each sensing operation maximizes the potential gain in information per sample.

2. If one has little prior knowledge about the sensed field. The concept of compressive wireless sensing(CWS) is introduced.

CWS is a universal scheme based on delivering random projections of the sensor data to the FC in an

efficient manner. And the FC can recover a good approximation of the data from these random projections.

In CWS, neither the sensor nodes nor the FC need to know the optimal basis elements in advance, but a relatively small number of random projections of a compressible or sparse signal contain most of its salient information. But for not exploiting prior knowledge of the signal field in the choice of projections that are communicated to the FC, the consequence of this scheme is a less favorable power-distortion-latency relationship.

To remark, the idea of using random matrix by CS here is to use random projections from a source-channel communication perspective.

# 3 System model

A WSN with n nodes, each node takes a noisy sample at time index $k$:

$$x_j^k = s_j^k + w_j^k, \qquad j = 1, ..., n, k \in \mathbb{N} \tag{1}$$

$s_j^k, k \in \mathbb{N}$ at each sensor corresponds to a deterministic but unknown sequence in $\mathbb{R}$.$w_j^k$are measurement errors that are zero-mean Gaussian random variables with variance $\sigma_w^2$ that are independent and identically distributed across space and time.

The observed data $\{x_j^k = s_j^k + w_j^k\}_{j=1}^n$at time $k$ can be considered as a vector $\mathbf{x}^k \in \mathbb{R}^n$

The physical phenomenon under observation can be characterized by the deterministic but unknown sequence of n-dimensional vectors

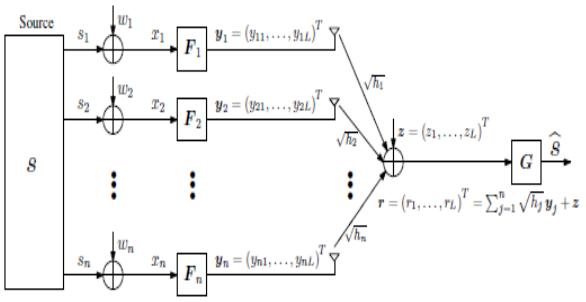
$$\mathbf{S} \triangleq \{\mathbf{s}^k\}_{k \in N} = \{\mathbf{s}^1, \mathbf{s}^2, ...\} \tag{2}$$

Take the transform coding point view in modeling the signal observed by the sensor nodes.Each noiseless snapshot $\mathbf{s}^k$ is well-approximated by a linear combination of $m$ vectors taken from an orthonormal basis of $\mathbb{R}^n$.

$\mathbf{s}^k$ can be classified into 2 kinds of signals: compressible signals and sparse signals.

1. If $\mathbf{s}^k$ is compressible, this means the largest $m$ coefficients of $\mathbf{s}^k$ in the basis $\{\psi_i\}_{i=1}^n$ can represent $\mathbf{s}^k$ well enough and the average squared-error has a upper bound $C_0 m^{-2\alpha}$.
2. If $\mathbf{s}^k$ is sparse, this means each noiseless temporal sample $\mathbf{s}^k$ can be fully represented by a few $\mathbf{\Psi}$-coefficients.

Given the observation vector $\mathbf{x}^k$, the sensor nodes communicate a reliable-enough estimate $\hat{\mathbf{s}}^k$ of the noiseless data vector $\mathbf{s}^k$ to a distant FC.




At each time instant $k$, given the observation $\{x_j^k\}_{\mathcal{K}=1}^k$, the encoders generate L-tuple $\mathbf{y}_j^k \triangleq \mathbf{F}_j(\{x_j^{\mathcal{K}}\})$ corresponding to L-channel uses per source observation. And the decoder $\mathbf{G}$ produces an estimate $\hat{\mathbf{s}}^k$ of the noiseless data vector $\mathbf{s}^k$.

The goal of the sensor network is to minimize the

1. The average total network power consumption per source observation $P_{tot}$.
2. The mean-squared error distortion measure $D$.
3. The latency $L$.

And the rest part will give how the above three quantities scale with $n$ in a given scheme.

# 4 Optimal Distortion Scaling in a Centralized System

Given the observation vector $\mathbf{x}^k$ at the FC, an optimal centralized estimator for a signal can be easily constructed by projecting $\mathbf{x}^k$ onto $m$ basis vectors.

That is to say that FC has precise knowledge of the ordering of coefficients of $\mathbf{s}^k$ in the compressing basis(indices of non-zero coefficients of $\mathbf{s}^k$ in the sparse basis).

1. For Compressible Signals

$$D^*_{cen} \asymp n^{-2\alpha/(2\alpha+1)} \tag{3}$$

2. For Sparse Signals

$$D^*_{cen} \asymp \frac{M}{n} \tag{4}$$

# 5 Distributed Projections in Wireless Sensor Networks

Here is the distributed method of communicating projections to the FC.

One way: the goal of the sensor is to obtain an estimate of the projection of noiseless sensor data onto a vector in $\mathbb{N}^n$ at the FC, $\phi_j x_j^k$, and aggregate these values up to obtain $\hat{v}^k = \Sigma_{j=1}^n \phi_j x_j^k$, then encode and transmit this value to FC. The alternative way: each sensor encoder $F_j$ corresponds to multiplying the sensor measurement $x_j^k$ with $(\sqrt{\frac{\rho}{h_j}}\phi_j)$, and all the nodes coherently transmit their respective $y_j^k$ in an analog fashion over the network-to-FC MAC. The decoder $G$ corresponds to a simple re-scaling of the received signal.

This distributed joint source-channel communication architecture requires only one channel use per source observation.

The end to end distortion is

$$D_v \triangleq \sigma_w^2 \parallel \phi \parallel_2^2 + \\ + (\frac{\sigma_z^2 d_u^\zeta (B^2+\sigma_w^2)}{\lambda P}) \parallel \phi \parallel_2^2 \tag{5}$$

The total network power consumption per source observation:

$$\lambda P \frac{\sigma_w^2}{d_u^\zeta (B^2+\sigma_w^2)} \leq P_{tot,v} \leq \lambda P \tag{6}$$

# 6 Distributed Estimation From Noisy Projections: Known Subspace

It's under the condition that the complete knowledge of the basis in which $\mathbf{S}$ is compressible and precise knowledge of the ordering of its coefficients in the compressing basis(indices of non-zero coefficients in the sparse basis).

1. Estimation of Compressible Signals

$$\frac{L}{n}\sigma_w^2 + (\frac{L}{n})(\frac{\sigma_z^2 d_u^\zeta (B^2+\sigma_w^2)}{\lambda P}) \leq \\ D \leq C_0 L^{-2\alpha} + \frac{L}{n}\sigma_w^2 + \\ (\frac{L}{n})(\frac{\sigma_z^2 d_u^\zeta (B^2+\sigma_w^2)}{\lambda P}) \tag{7}$$

$$\lambda LP(\frac{\sigma_w^2}{d_u^\zeta (B^2+\sigma_w^2)}) \leq P_{tot} \leq \lambda LP \tag{8}$$

2. Estimation of Sparse Signals

$$D = (\frac{M}{n})\sigma_w^2 + (\frac{M}{n})(\frac{\sigma_z^2 d_u^\zeta (B^2+\sigma_w^2)}{\lambda P}) \tag{9}$$

$$\lambda MP(\frac{\sigma_w^2}{d_u^\zeta (B^2+\sigma_w^2)}) \leq P_{tot} \leq \lambda MP \tag{10}$$

$$L = M \tag{11}$$

# 7 Distributed Estimation From Noisy Projections: Unknown Subspace

Even if little or no prior knowledge about the sensed data is assumed, compressive wireless sensing provides us with consistent estimation scheme, while $P_tot$ and $L$ grow at most sub-linearly with the number of nodes in the network.

The basic idea behind CWS is that instead of projecting the sensor network data onto a subset of a deterministic basis of $\mathbb{R}$, the FC tries to reconstruct

$\mathbf{s}^k$ from random projections of the sensor network data.

By employing $L$ random $\phi_i$projections, the projection estimates at the FC:

$$\hat{v}_i^k = \phi_i^T \mathbf{s}^k + \phi_i^T \mathbf{w}^k + \hat{z}_i^k, hspace0.5cmi = 1, ..., L \tag{12}$$

CWS estimate

$$\hat{\mathbf{s}}^k = arg \min_{\mathbf{s} \in \mathbf{S}_q} \{\hat{R}(\mathbf{s}) + \frac{c(\mathbf{s}) \log(2)}{L\epsilon}\} \tag{13}$$

$\mathbf{S}_q$denote a countable collection of candidate reconstruction vectors. And the first term:

$$\hat{R}(\mathbf{s}) = \frac{1}{L} \Sigma_{i=1}^{L} (\hat{v}_i^k - \phi_i^T \mathbf{s}^k) \tag{14}$$

The optimization problem is modified to be

$$\hat{\theta}^k = arg \min_{\theta \in \theta_q} \{\parallel \hat{v}_L^k - \phi_L^T \psi \theta \parallel_2^2 + \\ + \frac{(1+q)\log(2)\log(n)}{\epsilon} \parallel \theta_0\} \tag{15}$$

With reference to [2]

1. For an $\alpha$-compressible $\mathbf{S}$

$$D \preceq (\frac{L}{\log(n)})^{-2\alpha/(2\alpha+1)} \tag{16}$$

2. For an $M$-sparse signal

$$D \preceq (\frac{L}{M \log(n)})^{-1} \tag{17}$$

# 8 Impact of Fading and Imperfect Phase Synchronization

When the network is not fully synchronized and transmissions from the sensor nodes undergo fading.

1. Distributed Projections in Wireless Sensor Networks
   By modifying the decoder, the achievable distortion can be

$$D_v \leq (\frac{\sigma_w^2 \bar{\bar{\gamma}} + B^2(\bar{\bar{\gamma}} - \bar{\gamma}^2)}{\bar{\bar{\gamma}}}) \parallel \phi \parallel_2^2 + \\ + (\frac{\sigma_z^2 d_u^{\zeta}(B^2 + \sigma_w^2)}{\lambda \bar{\gamma}^2 P}) \parallel \phi \parallel_2^2 \tag{18}$$

   The same distortion scaling behavior with the only difference in the scaling constants. And as long as $\bar{\gamma} \neq 0$, $D_v \asymp \parallel \phi \parallel_2^2 \asymp D_v^*$

   Employ one channel use per source observation. Total network power consumption associated with achieving the distortion remains non-affected.

2. Distributed Estimation from Noisy Projections: Known Subspace
   By modifying the decoder, the resulting distortion of an $\alpha$-compressible signal is

$$D \leq C_0 L^{-2\alpha} + (\frac{L}{n})(\frac{\sigma_w^2 \bar{\bar{\gamma}} + B^2(\bar{\bar{\gamma}} - \bar{\gamma}^2)}{\bar{\bar{\gamma}}}) + \\ + (\frac{L}{n})(\frac{sigma_z^2 d_u^{\zeta}(B^2 + sigma_w^2)}{\lambda \bar{\gamma}^2 P}) \tag{19}$$

   Ignoring constants,

$$(\frac{L}{n}) + (\frac{L}{\lambda n}) \preceq D \preceq L^{-2\alpha} + (\frac{L}{n}) + (\frac{L}{\lambda n}) \tag{20}$$

   has the same scaling behavior. And the resulting distortion of an $M$-sparse signal is

$$D \asymp (\frac{M}{n}) + (\frac{M}{\lambda n}) \tag{21}$$

   Total network power consumption per source observation remains unaffected.

3. Compressive Wireless Sensing
   The net effect of phase synchronization errors is the introduction of a new noise-like term. The fading envelop of each sensor's transmission is given by $g_j^k \triangleq 1 + \epsilon_j^k$. The net result is a new noise-like term.

# 9 Appendix

In this report, we fail to have additive idea of our own, but we seemingly have found our direction of study in the field of compressive sensing. Given the above theoretical content, we hope to extend it further and do some research and simulation of our own. And the next report would be supposed to be the results of our own.

# Fast Solution of $l_1$-norm Minimization Problems

Luo dixin

***Abstract*—The minimum l1-norm solution to an underdetermined system of linear equations $y = Ax$, is often, remarkably, also the sparsest solution to that system. This sparsity-seeking property is of interest in signal processing and information transmission. However, general-purpose optimizers are much too slow for l1 minimization in many large-scale applications.**

**The Homotopy method was originally proposed for solving noisy overdetermined l1-penalized least squares problems. We here apply it to solve the noiseless underdetermined l1-minimization problem min kxk1 subject to $y = Ax$.**

March 29, 2009

## I. Theories

### A. l1 norm minimization

In an abstract fashion, our problem can be formulated as finding the sparsest non-negative solution to an underdetermined system of equations, which is the solution with the smallest number of nonzero elements. We would therefore like to solve the optimization problem:

$(P_0)$ min$\| x \|_0$ subject to $y = Ax$, $x \geq 0$.

However, it is well-known that this non-convex combinatorial optimization problem is NP-hard and therefore we consider the convex optimization problem:

$(P_1)$ min$\| x \|_1$ subject to $y = Ax$, $x \geq 0$.

which can be cast as a standard linear program, and solved using interior point methods [3]. When the solution is sufficiently sparse there exists equivalence between (P0) and (P1) [4, 5]. In most practical applications, we observe noisy data and would like to solve the problem:

$(P_{1,\varepsilon})$ min$\| x \|_1$ subject to $\| y - Ax \| \leq \varepsilon$

### B. Orthogonal Matching Pursuit (OMP)

Many of the applications of (P1) can instead be attacked heuristically by fitting sparse models, using greedy stepwise least squares. This approach is often called Matching Pursuit or Orthogonal Matching Pursuit (OMP)in the signal processing literature. Rather than minimizing an objective function, OMP constructs a sparse solution to a given problem by iteratively building up an approximation; the vector y is approximated as a linear combination of a few columns of A, where the active set of columns to be used is built column by column, in a greedy fashion. At each iteration a new column is added to the active set C the column that best correlates with the current residual.

### C. Algorithm To Solve (P1)

In parallel with developments in the signal processing literature, there has also been interest in the statistical community in fitting regression models while imposing l1-norm constraints on the regression coefficients. Donoho mentioned the so-called Lasso problem in his report, which we state using our notation as follows:

$(L_q)$ $min_x \| y - Ax \|_2^2$ subject to $\| x \|_1 \leq q$

in words: a least-squares fit subject to an l1-norm constraint on the coefficients. In Tibshiranis original proposal, $A_{d\times n}$ was assumed to have $d > n$, i.e. representing an overdetermined linear system. It is convenient to consider instead the unconstrained optimization problem

$(D_\lambda)$ $min_x \| y - Ax \|_2^2 /2 + \lambda \| x \|_1$

i.e. a form of l1-penalized least-squares. Indeed, problems (Lq) and $(D_\lambda)$ are equivalent under an appropriate correspondence of parameters. To see that, associate with each problem $(D_\lambda) : \lambda \epsilon [0, \infty)$ a solution $\widetilde{x_\lambda}$ (for simplicity assumed unique). The set $\widetilde{x_\lambda} : \lambda \epsilon [0, \infty)$identifies a solution path, with $\widetilde{x}_\lambda = 0$ for $\lambda$ large and, as $\lambda \to 0$, $\widetilde{x}_\lambda$ converging to the solution of (P1). Similarly, $\widetilde{x}_q : q \epsilon [0, \infty)$traces out a solution path for problem (Lq), with$\widetilde{x}_q = 0$for $q = 0$ and, as q increases, $\widetilde{x}_q$ converging to the solution of (P1). Thus, there is a reparametrization $q(\lambda)$ defined by $q(\lambda) = \| \widetilde{x_\lambda} \|_1$ so that the solution paths of $(D_\lambda)$and $(L_{q_\lambda})$ coincide.

### D. Homotopy Algorithm for Solving (P1)

Apply the Homotopy method: follow the solution path from $x_{\lambda_0} = 0$ to $\widetilde{x}_0$. Upon reaching the $\lambda = 0$ limit, $(P_1)$ is solved.

Traditionally, to solve $(P_1)$, one would apply the simplex algorithm or an interior-point method, which, in general, starts out with a dense solution and converges to the solution of $(P_1)$ through a sequence of iterations, each requiring the solution of a full linear system. In contrast, the Homotopy method starts out at $x_{\lambda_0} = 0$ , and successively builds a sparse solution by adding or removing elements from its active set. Clearly, in a sparse setting, this latter approach is much more favorable, since, as long as the solution has few nonzeros, Homotopy will reach the solution in a few steps.

Numerically, each step of the algorithm involves the rank-one update of a linear system, and so if the whole procedure stops in k steps, yielding a solution with k nonzeros, its overall complexity is bounded by $k^3 + kdn$ flops. For $k \ll d$ and $d \propto n$ , this is far better than the $d^3/3$ flops it would take to solve just one $d \times d$ linear system. Moreover, to solve $(D_\lambda)$ for all $\lambda \leq 0$ by a traditional approach, one would need to

repeatedly solve a quadratic program for every $\lambda$ value of interest. For any problem size beyond the very smallest, that would be prohibitively time consuming. In contrast, Homotopy delivers all the solutions $\widetilde{x_\lambda}$ to $(D_\lambda)$, $\lambda \geq 0$ .

### E. LARS

Some scientists developed an approximation to the Homotopy algorithm which is quite instructive. The Homotopy algorithm maintains an active set of nonzero variables composing the current solution. When moving to a new vertex of the solution polygon, the algorithm may either add new elements to or remove existing elements from the active set. The Lars procedure is obtained by following the same sequence of steps, only omitting the step that considers removal of variables from the active set, thus constraining its behavior to adding new elements to the current approximation. In other words, once activated, a variable is never removed.

In modifying the stepwise rule, of course, one implicitly obtains a new polygonal path, the Lars path, which in general may be different from the Lasso path. Yet, Efron et al. observed that in practice, the Lars path is often identical to the Lasso path. This equality is very interesting in the present context, because Lars is so similar to Omp. Both algorithms build up a model a step at a time, adding a new variable to the active set at each step, and ensuring that the new variable is in some sense the most important among the potential candidate variables. The details of determining importance differ but in both cases involve the inner product between the candidate new variables and the current residual.

In short, a stepwise algorithm with a greedy flavor can sometimes produce the same result as full-blown l1 minimization. This suggests a possibility which can be stated in two different ways:

1) ... that an algorithm for quasi l1 minimization runs just as rapidly as Omp.
2) ... that an algorithm visibly very similar to Omp can be just as effective as ‘1 minimization.

### F. Properties of Homotopy

The Homotopy algorithm is, algorithmically speaking, a variant of Lars, differing only in a few lines of code. And yet this small variation makes Homotopy rigorously able to solve a global optimization problem.

More explicitly, the difference between Homotopy and Lars lies in the provision by Homotopy for terms to leave as well as enter the active set. This means that the number of steps required by Homotopy can, in principle, be significantly greater than the number of steps required by Lars, as terms enter and leave the active set numerous times. If so, we observe model churning which causes Homotopy to run slowly. We present evidence that under favorable conditions, such churning does not occur. In such cases, the Homotopy algorithm is roughly as fast as both Lars and Omp.

In this paper, we consider two settings for performance measurement: deterministic incoherent matrices and random matrices. D.L.Donoho demonstrated that, when a k-sparse representation exists, $k \leq d$, the Homotopy algorithm finds it in k steps. Results in each setting parallel existing results about Omp in that setting. Moreover, each step of Homotopy is identical to a step of Lars and therefore very similar to a corresponding step of Omp.

### G. Incoherent Systems

The mutual coherence M(A) of a matrix A whose columns are normalized to unit length is the maximal off-diagonal entry of the Gram matrix $A^T A$. We call the collection of matrices A with $M(A) \leq \mu$ the incoherent ensemble with coherence bound $\mu$ (denoted $Inc_\mu$). Let S($Inc_\mu$; d, n, k) be the suite of problems with d n matrices drawn from the incoherent ensemble $Inc_\mu$ , with vectors $\alpha_0$ having $\| \alpha_0 \|_0 \leq k$ . For the incoherent problem suite, we have the following result:

Let (A, y) be a problem instance drawn from S($Inc_\mu$; d, n, k). Suppose that

$$k \leq (\mu^{-1} + 1)/2 \tag{1}$$

Then, the Homotopy algorithm runs k steps and stops, delivering the solution $\alpha_0$.

The condition (1) has appeared elsewhere; work by Donoho[6] stated that when (1) holds, the sparsest solution is unique and equal to $\alpha_0$, and both l1 minimization and Omp recover $\alpha_0$. In particular, Omp takes at most k steps to reach the solution.

In short we can learn from here that for the general class of problems S($Inc_\mu$; d, n, k), where a unique sparsest solution is known to exist, and where Omp finds it in k steps, Homotopy finds that same solution in the same number of steps. Note that Homotopy always solves the l1 minimization problem; the result shows that it operates particularly rapidly over the sparse incoherent suite.

## II. Bridging l1 Minimization and OMP

I learned from some earlier paper that there is undeniable parallelism in results about the ability of l1 minimization and Omp to recover sparse solutions to systems of underdetermined linear equations. Those papers [7,8,9,10,11] mentioned instances where, under similar conditions, both l1 minimization and Omp recover the sparsest solution. Yet, these works did not offer any insights as to why the two seemingly disparate techniques should offer similar performance in such cases. It is necessary to talk about the linkage between l1 minimization and Omp.

### A. l1 Minimization → Homotopy

For the phrase ’algorithm A has the k-step solution property’ to make sense, the algorithm must have a stepwise structure, building a solution approximation term by term. Thus, we cannot, in general, speak of l1 minimization as having the k-step solution property, as there are many algorithms for such minimization, including those with no meaningful notion of stepwise approximate solution construction. This is where the Homotopy algorithm fits in, bridging the high-level notion

of l1 minimization with the lower-level stepwise structure of Omp. Earlier work has shown that Homotopy is a correct algorithm for solving the l1 minimization problem (P1). In addition it builds an approximate solution in a stepwise fashion, thus making the k-step solution property applicable.

### B. Homotopy → LARS

As noted earlier, the Lars procedure is a simplification of the Homotopy algorithm, achieved by removing the condition for sign agreement of the current solution and the residual correlations. This brings us one step closer to Omp; while Homotopy allows for removal of terms from the active set, both Lars and Omp insert terms, one by one, into the active set, never removing any active elements.

### C. LARS → OMP

Lars also selects a new term according to the maximum correlation criterion, and then solves

$$A_I^T A_I x_l(I) = A_I^T y - \lambda_l \cdot s \tag{2}$$

where s is a vector of length ‘, recording the signs of residual correlations of each term at the point it entered the active set, and $\lambda_l$ is the correlation magnitude at the breakpoint on the Lars path.

To summarize, we exhibited a series of transformations, starting with l1 minimization, and ending with greedy pursuit. Each transformation is characterized clearly, and maintains the k-step property of the solution path. We believe this sequence clarifies initially surprising similarities between results about l1 minimization and Omp.

## III. Conclusion

This report mainly concludes some theories of l1 minimization. l1 minimization has a tight connection with OMP and Homotopy algorithm which is at peak effciency when it satisfies the k-step solution property. We plan to make further study about Homotopy algorithm and the comparison of the performance of Homotopy, Lars and Omp in recovering sparse solution.

## References


[1] Waheed U. Bajwa, Jarvis D. Haupt, Akbar M. Sayeed, and Robert D. Nowak, *Joint Source-Channel Communication for Distributed Estimation in Sensor Networks*. IT Trans. 2007

[2] J.Haupt and R. Nowak, *Signal reconstruction from noisy random porjections*. IEEE Trans. Inform. Theory Sept.2006

[3] Emmanuel Cand‘es and Terence Tao *Decoding by linear programming*. IEEE Transactions on Information Theory, 51(12):4203C4215, December 2005.

[4] Emmanuel J. Cand‘es and Justin Romberg *Practical signal recovery from random projections*. Manuscript, 2005.

[5] Emmanuel J. Cand‘es, Justin Romberg, and Terence Tao, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*. IEEE Trans. Inform. Theory, pages 489–509. Feb. 2006.

[6] David L. Donoho and Xiaoming Huo, *Uncertainty principles and ideal atomic decomposition*. IEEE Trans. Info. Theory, 47(7):2845C2862, 2001.

[7] Scott Shaobing Chen, David L. Donoho, *Atomic decomposition by basis pursuit*. SIAM J. Sci Comp., 20(1):33C61, 1999.

[8] David L. Donoho, Michael Elad, and Vladimir N. Temlyakov, *Stable recovery of sparse overcomplete representations in the presence of noise*. IEEE Transactions on Information Theory, 52(1):6C18, January 2006.

[9] David L. Donoho, *For most large underdetermined systems of linear equations, the minimal l1 solution is also the sparsest solution.*. Comm. Pure Appl. Math., 59(7):907C934, July 2006.

[10] Joel A. Tropp, *Algorithmic results for sparse approximation*. IEEE Transactions on Information Theory, 50(11):2231C2242, October 2004.

[11] Joel A. Tropp, *Convex programming methods for subset selection and sparse approximation*. IEEE Transactions on Information Theory, 51(3):1030C1051, March 2006.