Sparse Signal Recovery and Dynamic Update of the Underdetermined System

M. Salman Asif and Justin Romberg
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332, USA
Email: {sasif, jrom}@gatech.edu

***Abstract*—Sparse signal priors help in a variety of modern signal processing tasks. In many cases, a sparse signal needs to be recovered from an underdetermined system of equations. For instance, sparse approximation of a signal with an overcomplete dictionary or reconstruction of a sparse signal from a small number of linear measurements. The reconstruction problem typically requires solving an ℓ_1 norm minimization problem. In this paper we present homotopy based algorithms to update the solution of some ℓ_1 problems when the system is updated by adding new rows or columns to the underlying system matrix. We also discuss a case where these ideas can be extended to accommodate for more general changes in the system matrix.**

I. Introduction

Sparsity plays a major role in many modern signal processing tasks; compression, denoising and signal restoration are typical examples where we benefit from the signal sparsity. In recent years, tremendous progress has been made in the area of sparse signal recovery from underdetermined systems [1], [2]. The sparse recovery problem can be formulated as follows. Suppose we are given a signal $y \in \mathbb{R}^M$ which obeys the following linear model

$$y = Ax, \tag{1}$$

where A is an $M \times N$ system matrix with $M \ll N$, and $x \in \mathbb{R}^N$ is the sparse vector we want to estimate. In the sparse approximation framework, we view y as an observed signal, A as an overcomplete dictionary, and x as a sparse decomposition of y in terms of columns of A [1], [3]. A related problem, commonly known as Compressive Sensing (CS) [4]–[6], considers y as indirect measurements of the unknown sparse signal x using the measurement matrix[1] A.

Ideally we would like to find the sparsest vector x which obeys the linear system in (1). Conceptually, it requires solving the following optimization problem

$$\text{minimize } \|x\|_0 \text{ subject to } Ax = y, \tag{2}$$

where $\|x\|_0$ denotes the number of nonzero entries in x. Unfortunately, it is a combinatorial problem and known to be NP-hard [7]. In practice, we instead solve a convex relaxation of (2), where ℓ_0 norm is replaced by ℓ_1 norm as follows

$$\underset{x}{\text{minimize}} \|x\|_1 \text{ subject to } Ax = y. \tag{3}$$

It is a convex program and can be efficiently solved using standard optimization routines [8], [9]. However, under some conditions on the sparsity of the signal x and *incoherence* or *restricted isometry* of the matrix A, solving the relaxed problem (3) can indeed recover the original sparse signal [1], [2].

A more general system model can be defined as

$$y = Ax + e, \tag{4}$$

where e denotes the system noise and x can be nearly-sparse or *compressible*. In such systems, we typically solve the basis pursuit denoising (BPDN) [10] (or LASSO [11]):

$$\underset{x}{\text{minimize}}\ \tau\|x\|_1 + \frac{1}{2}\|Ax - y\|_2^2, \tag{5}$$

with some suitable choice of regularization parameter $\tau > 0$, which controls the tradeoff between the sparsity of the solution and data fidelity. Over the last few years, a number of efficient schemes have been devised to efficiently solve (5) [12]–[14].

Most of the sparse recovery algorithms focus on solving problems like (3) and (5) for a *fixed* system. Recently, some homotopy based schemes have been proposed to update the solution of (5) when new measurements are sequentially added to the system (i.e., one new entry added to y and one row added to A) [15], [16], or when y is replaced with a completely new set of measurements of a closely related signal while keeping A the same [17], [18]. These algorithms are derived by introducing the new measurements into the optimization program gradually, and tracking how the solution changes. The problems are formulated such that the movement from one solution to the next can be broken down into a series of linear steps, with each link traversed using a low-rank update. The motivation behind these dynamic updating schemes is that when the underlying system changes, the solution may not change much from one instance to the next. Therefore, update from current estimate to the solution of the updated system would be substantially cheaper than solving a new optimization problem from scratch.

In this paper we discuss dynamic updating schemes for two types of changes in the system matrix. In Sec. III we discuss the dynamic updating scheme when new rows are added to the system. In Sec. IV we discuss the updating scheme when new columns are added to the system matrix. In Sec. VI we briefly discuss a special case where dynamic updating can be used for

[1] We assume that the sparsity basis and measurement matrix are combined together to form A.

more general modifications in the system matrix, under rather restrictive, so-called *incoherence condition*.

II. Homotopy

Our proposed dynamic updating schemes are based on the homotopy continuation principle [19]. Homotopy methods provide a general framework for transforming an optimization problem into a simpler (easily solvable) form, where the transformation is parameterized by the so-called *homotopy parameter*. In ℓ_1 problems, as the homotopy parameter is varied, the solution traverses a piecewise linear homotopy path towards the final solution, where each intermediate homotopy step requires a low rank update. The homotopy path is followed by ensuring that certain optimality conditions are being maintained.

Let us consider the standard LASSO homotopy [19], [20] for (5), where τ is the homotopy parameter. To be a solution of (5) at any given value of τ, a vector $x^\star$ must obey the following set of N conditions (constraints)

$$\|A^T(Ax^\star - y)\|_\infty \le \tau, \qquad \text{(L)}$$

more precisely

L1. $A_\Gamma^T(Ax^\star - y) = -\tau z$

L2. $\|A_{\Gamma^c}^T(Ax - y)\|_\infty < \tau$,

where Γ is the support (index set for nonzero elements) of $x^\star$, A_Γ is the $M \times |\Gamma|$ matrix formed from the respective columns in A and z is the sign sequence of $x^\star$ on Γ. From this we see that $x^\star$ can be calculated directly from the support Γ and signs z using

$$x^\star = \begin{cases} (A_\Gamma^T A_\Gamma)^{-1}(A_\Gamma^T y - \tau z) & \text{on } \Gamma \\ 0 & \text{otherwise.} \end{cases}$$

As we change τ, the solution moves along a line with direction $(A_\Gamma^T A_\Gamma)^{-1} z$ until one of two things happens: an element of $x^\star$ is shrunk to zero, removing it from the support of $x^\star$, or another constraint in (L) becomes active, adding a new element to the support of $x^\star$. At these so-called *critical points*, both the support of $x^\star$ and the direction of the solution path change. Also, at any point on the solution path it is straightforward to calculate how much we need to vary τ to take us to a critical point in either direction. Since there is only one element change in the support, computing new update direction requires a rank one update of the inverse $(A_\Gamma^T A_\Gamma)^{-1}$.

III. Adding rows to the system matrix

Suppose we have solved the optimization problem (5) for a given value of τ to get the solution x_0 with support Γ and sign sequence z. Now suppose we get P new measurements, given as $w = Bx + d$, where B is a $P \times N$ matrix and $d \in \mathbb{R}^P$ denotes noise in the new measurements. We now want to solve the following updated problem

$$\underset{x}{\text{minimize}}\ \tau\|x\|_1 + \frac{1}{2}\left(\|Ax - y\|_2^2 + \|Bx - w\|_2^2\right). \qquad (6)$$

As described in [15], [16], one new measurement can be incorporated by introducing a homotopy parameter ϵ as

$$\underset{x}{\text{minimize}}\ \tau\|x\|_1 + \frac{1}{2}\left(\|Ax - y\|_2^2 + \epsilon\|Bx - w\|_2^2\right).$$

However, this homotopy scheme does not work with more than one new measurement. We propose the following homotopy scheme to incorporate multiple measurements

$$\min_x.\ \tau\|x\|_1 + \frac{1}{2}\left(\|Ax - y\|_2^2 + \|Bx - (1-\epsilon)Bx_0 - \epsilon w\|_2^2\right), \qquad (7)$$

where x_0 is the solution of (5). Note that as ϵ varies from 0 to 1, we move from the old problem (5) to the new one (6).

The homotopy algorithm for (7) closely resembles the homotopy algorithm for dynamic update of time-varying signal, detailed in [17], [18]. By adapting the optimality conditions L1 and L2, we see that for any vector $x^\star$ to be the solution of (7) at any given value of ϵ it must obey the following conditions

$$\|A^T(Ax^\star - y) + B^T(Bx^\star - (1-\epsilon)Bx_0 - \epsilon w)\|_\infty \le \tau, \qquad (8)$$

or more precisely

$$A_\Gamma^T(Ax^\star - y) + B_\Gamma^T(Bx^\star - (1-\epsilon)Bx_0 - \epsilon w) = -\tau z \qquad \text{(8a)}$$

$$\|A_{\Gamma^c}^T(Ax^\star - y) + B_{\Gamma^c}^T(Bx^\star - (1-\epsilon)Bx_0 - \epsilon w)\|_\infty \le \tau, \qquad \text{(8a)}$$

where Γ is the support of $x^\star$ and z is its sign sequence on Γ. We can see from (8a) that the solution to (7) follows piecewise linear path as ϵ varies; critical points occur when an element is added to or removed from the solution $x^\star$. Suppose we are at a solution x_k (with support Γ and sign sequence z) to (7) at some value of $\epsilon = \epsilon_k$. As we increase ϵ by a small amount from ϵ_k to $\epsilon_k + \theta$, the solution moves to $x_k^+ = x_k + \theta\partial x$, where

$$\partial x = (A_\Gamma^T A_\Gamma + B_\Gamma^T B_\Gamma)^{-1} B_\Gamma^T (w - Bx_0). \qquad (9)$$

Moving in the direction of ∂x by increasing the step size θ, we eventually hit a *critical point* where either one of the entries in x_k shrinks to zero or one of the constraints in (8a) becomes active (equal to τ). The smallest amount we can move ϵ so that the former is true is simply

$$\theta^- = \min_{j\in\Gamma}\left(\frac{-x_k(j)}{\partial x(j)}\right)_+, \qquad (10)$$

where $\min(\cdot)_+$ denotes that the minimum is taken over positive arguments only. For the latter, set

$$p_k = A^T(Ax_k - y) + B^T(Bx_k - \epsilon_k w - (1-\epsilon_k)Bx_0)$$
$$d_k = (A^TA + B^TB)\partial x - B^T(w - Bx_0).$$

We are now looking for the smallest stepsize $\Delta\epsilon$ so that $p_k(j) + \Delta\epsilon \cdot d_k(j) = \pm\tau$ for some $j \in \Gamma^c$. This is given by

$$\theta^+ = \min_{j\in\Gamma^c}\left(\frac{\tau - p_k(j)}{d_k(j)}, \frac{\tau + p_k(j)}{-d_k(j)}\right)_+. \qquad (12)$$

The stepsize to the next critical point is

$$\theta = \min(\theta^+, \theta^-). \qquad (13)$$

With the direction ∂x and stepsize θ chosen, the next critical value of ϵ and the solution at that point become

$$\epsilon_{k+1} = \epsilon_k + \theta, \quad x_{k+1} = x_k + \theta \partial x.$$

The support for new solution x_{k+1} differs from Γ by one element. Let γ^- be the index for the minimizer in (10) and γ^+ be the index for the minimizer in (12). If we chose θ^- in (13), then we remove γ^- from the support Γ and the sign sequence z. If we chose θ^+ in (13), then we add γ^+ to the support, and add the corresponding sign to z. This procedure is repeated until $\epsilon = 1$.

The main computational cost at every homotopy step comes from solving a $|\Gamma| \times |\Gamma|$ system of equations to compute the direction in (9), and two matrix-vector multiplications to compute the d_k for the stepsize. Since the support changes by a single element at every homotopy step, the update direction can be computed using rank-1 update methods [21].

IV. Adding columns to the system matrix

Suppose we have solved (5) for a given value of τ to get the solution x_0 with support Γ and sign sequence z. Now suppose we modify the system in (4) by adding P new columns in the system matrix. The modified system can be written as

$$y = Ax + Bu + e, \tag{14}$$

where B is an $M \times P$ matrix and $u \in \mathbb{R}^P$ is a vector denoting the respective decomposition coefficients. Consequently, we want to solve the following updated BPDN problem

$$\underset{x,u}{\text{minimize}} \ \tau(\|x\|_1 + \|u\|_1) + \frac{1}{2}\|Ax + Bu - y\|_2^2. \tag{15}$$

In the context of sparse approximation using overcomplete dictionary, adding B amounts to adding new atoms to the dictionary. For the case of compressive sensing, this update can be considered as adding columns to the underlying sparsity inducing dictionary. The intuition behind this update is that adding new columns to the dictionary may enhance the sparsity of the solution. For example, suppose our signal is a superposition of spikes and sinusoids, whereas our representation basis A only contains sinusoidal elements. In such case, adding new columns with time-localized content will improve the reconstruction quality. As another example, consider the face recognition setup in [22], where each column in B would correspond to a new face in the face directory and may improve the feature extraction performance.

Our proposed update algorithm is a simple extension of the standard LASSO homotopy algorithm. We use the following homotopy formulation for (15)

$$\text{minimize} \ \tau(\|x\|_1 + \epsilon\|u\|_1) + \frac{1}{2}\|Ax + Bu - y\|_2^2, \tag{16}$$

where τ is fixed and ϵ is the homotopy parameter. Note that for a very large value of ϵ (e.g., $\epsilon\tau > \|B^T(Ax_0 - y)\|_\infty$) the solution of (16) is same as x_0, and the solution of (16) reaches the solution of (15) as ϵ is reduced to 1.

The optimality conditions for any solution $x^\star$ and $u^\star$ to (16) at the given values of τ and ϵ can be written as

$$\begin{aligned} A_{\Gamma_x}^T(Ax^\star + Bu^\star - y) &= -\tau z_x, \\ \|A_{\Gamma_x^c}^T(Ax^\star + Bu^\star - y)\|_\infty &< \tau \\ B_{\Gamma_u}^T(Ax^\star + Bu^\star - y) &= -\tau\epsilon z_u, \\ \|B_{\Gamma_u^c}^T(Ax^\star + Bu^\star - y)\|_\infty &< \tau\epsilon, \end{aligned}$$

where Γ_x and Γ_u denote the support of $x^\star$ and $u^\star$, and z_x and z_u denote their signs on respective supports. With these optimality conditions in hand, we can develop the update homotopy algorithm similar to the standard LASSO homotopy. Suppose x_k and u_k are the solutions to (16) at τ and $\epsilon = \epsilon_k$, with supports and sign sequence Γ_x, Γ_u, z_x and z_u. Let us denote $D = [A \ B]$, $q_k = \begin{bmatrix} x_k \\ u_k \end{bmatrix}$ and $\Gamma = \Gamma_x \cup \Gamma_u$ and $z = \begin{bmatrix} z_x \\ z_u \end{bmatrix}$. Using the active constraints in the optimality conditions (17), we can write the optimality conditions at $\epsilon^+ = \epsilon_k - \theta$ for a diminishingly small value of $\theta > 0$, with new optimal solution q^+, as follows

$$\begin{aligned} A_{\Gamma_x}^T(Ax^+ + Bu^+ - y) &= -\tau z_x \\ B_{\Gamma_u}^T(Ax^+ + Bu^+ - y) &= -\tau\epsilon^+ z_u \\ \equiv D_\Gamma^T(Dq^+ - y) &= -\begin{bmatrix} \tau \\ \tau\epsilon^+ \end{bmatrix} z. \end{aligned} \tag{18}$$

The update direction $\partial q = q^+ - q_k$ can therefore be written as

$$\partial q = \begin{cases} (D_\Gamma^T D_\Gamma)^{-1} \begin{bmatrix} 0 \\ z_u \end{bmatrix} & \text{on } \Gamma \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

We move q_k in this update direction to a critical point where support changes, similar to the standard LASSO homotopy—either an inactive constraint becomes active or an element in q shrinks to zero. Assuming that the support changes with a step size $\tau\theta$, the new value of homotopy parameter ϵ will be $\epsilon^+ = \epsilon - \theta$. We repeat the homotopy iterations until ϵ is 1.

Note that the very first iteration of update homotopy can be initialized with $u = 0$ at $\epsilon = \|B^T(Ax_0 - y)\|_\infty/\tau$, with one support element $\Gamma_u = \gamma$ with sign $z_u = z_\gamma$ where $B_\gamma^T(Ax_0 - y) = -\tau\epsilon z_\gamma$.

V. Experiments

A. Row update

In this experiment we test the performance of our proposed homotopy algorithm when multiple measurements are added to the system. Our experiment setup is as follows. The underlying sparse signal x contains ± 1 spikes at K randomly chosen locations. The $M \times N$ matrix A and $P \times N$ matrix B have Gaussian entries with distribution $\mathrm{Normal}(0, 1/(M+P))$. We observe $y = Ax + e$ and $w = Bx + d$ with e and d as Gaussian noise vectors whose entries are distributed $\mathrm{Normal}(0, 0.01^2)$. We start by solving (5) for a given value of τ. Then we solve (6) with the additional P measurements, using the algorithm described in Sec. III. The results of 50 simulations with $N =$

TABLE II
COMPARISON OF ITERATION COUNT FOR THE DYNAMIC UPDATE AND LASSO WITH COLUMN ADDITION AT $\tau = 0.02$.

Signal type	dynamic update	LASSO
Blocks	9.88	69.37
HeaviSine	4.29	22.23
Piece-Polynomial	8.25	59.74
Ramp	5.17	30.93

1024, $M = 512$, $K = M/5$ are summarized in Table I for different values of P and τ. We chose $\tau = \lambda\|A^T y\|_\infty$ with $\lambda \in \{0.5,\ 0.1,\ 0.05,\ 0.01\}$. The experiments were run on a standard desktop PC, and we recorded the average number of times we needed to apply[2] A^T and A (nProdAtA). The results are summarized in Table I, and are compared against the standard BPDN homotopy algorithm (LASSO), GPSR [12] with a warm start, and FPC_AS [13] with a warm start.

The average number of homotopy iterations taken for the update varies with the sparsity of the solution. At large values of τ, the solution has a small number of non-zero entries and the update requires lesser homotopy steps. For smaller values of τ, the solution has many more non-zero terms and the number of iterations in the update increases. The number of iterations scale nicely with the number of new measurements (P) too.

B. Column update

In this experiment we test the performance of homotopy update scheme when columns are added in the system matrix. We construct an $M \times 2M$ overcomplete dictionary D by taking union of M-dimensional DCT and orthogonal Wavelet bases (using Daubechies 8 filter). For a given signal y of length M, we start by solving (5) using an $M \times M$ matrix A whose columns are chosen at random from D. Then we sequentially add remaining columns to the system, one at a time, and solve (20) using the previous solution in the homotopy algorithm described in Sec. IV. We tested our algorithm for four different signals y, taken from WaveLab toolbox [23], which are normalized to have maximum magnitude of 1. Snapshots of the original signals along with approximations with a random subset of columns are shown in Fig. 1. The results over 20 simulations for $M = 128$ are tabulated in Table II, where we compare the average number of iterations taken by the dynamic update algorithm and standard LASSO homotopy algorithm (run from scratch).

VI. DISCUSSION

In Sec. III and IV we discussed updating schemes for two specific types of modifications in the system matrix, namely

[2]Each iteration of the dynamic update algorithm requires an application of A and A^T along with several much smaller matrix-vector multiplies to perform the rank-1 update, and an initial factorization of B_Γ. Since these smaller matrix-vector multiplies are so much cheaper, the numbers in the table include only applications of A^T and A.

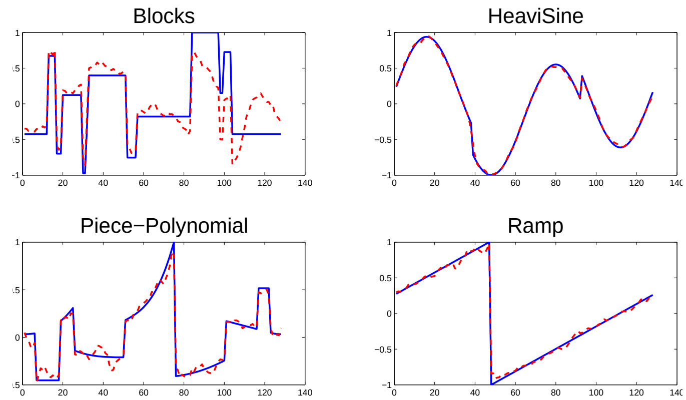

Fig. 1. Snapshots of different test signals. Solid line: original signal. Dashed line: approximation with 160 columns.

when rows and columns are added to the system. In some cases we may want to update the solution of (5) after a different modifications in the matrix A. Suppose we have solved (5) using matrix A and then we modify A to $\tilde{A}$ and we want to solve the following updated optimization problem

$$\underset{x}{\text{minimize}}\ \tau\|x\|_1 + \frac{1}{2}\|\tilde{A}x - y\|_2^2. \tag{20}$$

We usually encounter such updates when we solve the optimization problem of the above form simultaneously over x and A. For example, dictionary learning [24], [25], blind deconvolution [26], or simultaneous image/video transformation and sparse representation [27], where we simultaneously estimate sparse coefficients and update the system matrix. Let us consider the following dictionary learning problem

$$\underset{X_{i=1,\ldots,T},A}{\text{minimize}}\ \tau\sum_i^T \|X_i\|_1 + \frac{1}{2}\sum_i \|AX_i - Y_i\|_2^2, \tag{21}$$

where A is the representation dictionary, X_i denotes sparse coefficients for observations Y_i. This problem is separable for each X_i and is typically solved by alternately minimizing over all X_i and A. After each update of A we get a slightly different matrix $\tilde{A}$, and we have to solve a problem of the form (20) for each X_i.

So far, we do not have an updating algorithm for such matrix updates for an arbitrary system. However, if we can find a vector $\tilde{y}$ such that

$$\tilde{A}_\Gamma^T(\tilde{A}x_0 - \tilde{y}) = -\tau z \tag{22a}$$
$$\|\tilde{A}_{\Gamma^c}^T(\tilde{A}x_0 - \tilde{y})\|_\infty < \tau, \tag{22b}$$

where x_0 is the solution of (5) with system matrix A, having support Γ and sign sequence z, then we can solve the following homotopy formulation using update scheme similar to the one described in Sec. III

$$\underset{x}{\text{minimize}}\ \tau\|x\|_1 + \frac{1}{2}\|\tilde{A}x - (1-\epsilon)\tilde{y} - \epsilon y\|_2^2. \tag{23}$$

Note that as ϵ varies from 0 to 1, the solution of (23) follows a homotopy path from x_0 to the solution of (20). Although it

TABLE I
COMPARISON OF THE DYNAMIC UPDATE, LASSO, GPSR AND FPC WITH P NEW MEASUREMENTS.

P	λ ($\tau = \lambda\|A^T y\|_\infty$)	dynamic update nProdAtA	LASSO nProdAtA	GPSR-BB nProdAtA	FPC_AS nProdAtA
1	0.5	2.3	41.86	11.86	15.98
	0.1	4.72	159.76	42.64	50.70
	0.05	4.5	162.34	38.80	97.73
	0.01	8.02	233.70	55.46	79.83
5	0.5	5.88	42.00	14.24	15.96
	0.1	9.58	152.54	46.42	47.48
	0.05	10.70	161.36	47.96	98.75
	0.01	20.32	227.82	66.64	78.58
10	0.5	7.6	44.72	14.96	16.12
	0.1	14.98	155.26	53.12	47.05
	0.05	16.40	162.72	52.12	98.51
	0.01	29.34	241.52	75.44	82.91

may not be possible to find such vector in every problem, but in a very restrictive case, it is trivial to find such vector. Note that if $|\Gamma| < \frac{1}{2}\left(1 + \frac{1}{\mu}\right)$, where μ is the incoherence of $\tilde{A}$, defined as $\mu = \max_{i \neq j} |\langle \tilde{a}_i, \tilde{a}_j \rangle|$. Then it is guaranteed that $\|\tilde{A}_{\Gamma^c}^T \tilde{A}_\Gamma (\tilde{A}_\Gamma^T \tilde{A}_\Gamma)^{-1} z\|_\infty < 1$ [28], and

$$\tilde{y} = \tilde{A}x_0 + \tau \tilde{A}_\Gamma (\tilde{A}_\Gamma^T \tilde{A}_\Gamma)^{-1} z$$

would trivially satisfy (22), hence (20) can be solved efficiently using the previous solution x_0 in (23).

REFERENCES

[1] D. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 2845–2862, Nov 2001.

[2] E. Candès and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[3] D. Donoho and M. Elad, "Optimally sparse representation in general (non-orthogonal) dictionaries via ℓ_1 minimization," *Proc. Nat. Aca. Sci*, vol. 100, no. 5, pp. 2197–2202, 2003.

[4] E. 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, Feb. 2006.

[5] E. Candès, "Compressive sampling," *Proceedings of the International Congress of Mathematicians, Madrid, Spain*, vol. 3, pp. 1433–1452, 2006.

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

[7] B. Natarajan, "Sparse Approximate Solutions to Linear Systems," *SIAM Journal on Computing*, vol. 24, p. 227, 1995.

[8] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, March 2004.

[9] E. Candès and J. Romberg, "ℓ_1-MAGIC: Recovery of Sparse Signals via Convex Programming." http://www.acm.caltech.edu/l1magic/.

[10] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1999.

[11] 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.

[12] M. Figueiredo, R. Nowak, and S. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586–597, 2007.

[13] Z. Wen and W. Yin, "FPC_AS: A matlab solver for ℓ_1-regularized least squares problems." http://www.caam.rice.edu/∼optimization/L1/FPC_AS/.

[14] S. Becker, J. Bobin, and E. Candès, "Nesta: A fast and accurate first-order method for sparse recovery." Accepted in SIAM J. on Imaging Sciences, 2009.

[15] P. J. Garrigues and L. E. Ghaoui, "An homotopy algorithm for the Lasso with online observations," *Neural Information Processing Systems (NIPS) 21*, December 2008.

[16] M. S. Asif and J. Romberg, "Streaming measurements in compressive sensing: ℓ_1 filtering," *42nd Asilomar conference on Signals, Systems and Computers*, October 2008.

[17] M. S. Asif and J. Romberg, "Dynamic updating for ℓ_1 minimization," *IEEE Journal of Selected Topics in Signal Processing*, April 2010.

[18] M. S. Asif and J. Romberg, "Dynamic updating for sparse time varying signals," *Information Sciences and Systems, 43rd Annual Conference on (CISS)*, March 2009.

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

[20] M. Osborne, B. Presnell, and B. Turlach, "A new approach to variable selection in least squares problems," *IMA Journal of Numerical Analysis*, vol. 20, no. 3, pp. 389–403, 2000.

[21] Å. Björck, *Numerical Methods for Least Squares Problems*. Society for Industrial Mathematics, 1996.

[22] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 210–227, 2008.

[23] J. Buckheit, S. Chen, D. Donoho, and I. Johnstone, "Wavelab 850, Software toolbox." http://www-stat.stanford.edu/∼wavelab/.

[24] B. Olshausen and D. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1?," *Vision research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[25] M. Aharon, M. Elad, and A. Bruckstein, "¡ tex¿ rmK¡/tex¿-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *Signal Processing, IEEE Transactions on*, vol. 54, no. 11, pp. 4311–4322, 2006.

[26] M. S. Asif, W. Mantzel, and J. Romberg, "Channel Protection: Random coding meets sparse channels," *Information theory workshop*, October 2009. To appear.

[27] J. Huang, X. Huang, and D. Metaxas, "Simultaneous image transformation and sparse representation recovery," *IEEE Conferenece on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2008.

[28] J. Fuchs, "On sparse representations in arbitrary redundant bases," *IEEE Transactions on Information Theory*, vol. 50, no. 6, pp. 1341–1344, 2004.