Basis Pursuit with Sequential Measurements and Time Varying Signals

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

***Abstract*—Sparse signal recovery from linear measurements is an important problem which arises in several signal processing applications. Basis pursuit is a standard convex optimization program which is often used for this purpose. In this paper we present two algorithms to dynamically update the solution of basis pursuit as 1) new measurements are sequentially added or 2) the underlying signal changes slightly. The goal is to avoid solving the (computationally expensive) optimization routine every time a small change occurs in the measurements. Our proposed update algorithms are based on homotopy principles, which iteratively update the solution by moving from an already solved problem towards the desired problem. Each homotopy step involves only a few matrix-vector multiplications. Simulation results show that the number of homotopy steps taken for the update is comparable to the sparsity of the underlying signals.**

I. Introduction

In [1], the authors presented a framework for dynamically updating the solutions of some ℓ_1 norm minimization problems including basis pursuit denoising [2] and Dantzig selector [3]. The solutions can be updated when either new measurements are added or the underlying signal changes. In this paper, we apply this general methodology to the basis pursuit [2].

Recovery of sparse signals from small number of linear measurements has become an important and extremely beneficial construct in signal processing. In past few years this idea been extensively studied, with applications in a diverse set of areas ranging from medical imaging to channel coding and beyond. This trend has been further accelerated with the advent of compressive sensing (CS) theory [4]. The results in CS suggest that exact signal recovery is possible under some suitable sparsity and incoherence conditions [5].

The general problem setup is as follows. We are given a set of m measurements

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

where A is an $m \times n$ measurement matrix with $m \ll n$ and $x \in \mathbb{R}^n$ is the unknown sparse signal we want to reconstruct. For the reconstruction, we typically solve the following ℓ_1 optimization problem, known as basis pursuit (BP):

$$\text{minimize } \ \|\tilde{x}\|_1 \ \text{ subject to } \ A\tilde{x} = y. \tag{2}$$

This is a convex program which can be recast into a linear program [6]. The computational cost for solving BP with a dense unstructured matrix is $O(m^2 n^{3/2})$ [7].

In this paper we discuss the *dynamic updating* of the BP, whenever there is a small change in the measurements. Assume that we have solved (2) for the system in (1), and consider the following two scenarios:

1) *Sequential measurement:* The original signal is not recovered, so we add one new measurement $w = bx$ to the system. The new system becomes
$$\begin{bmatrix} y \\ w \end{bmatrix} = \begin{bmatrix} A \\ b \end{bmatrix} x. \tag{3}$$
2) *Time varying signal:* The underlying signal x changes to $\breve{x}$ and we receive the new set of m measurements
$$\breve{y} = A\breve{x}. \tag{4}$$

We now want to solve BP for the systems in (3) or (4). Our goal is to avoid solving a new optimization program from scratch after each measurement update. Instead we show that we can quickly compute the solution using information from the already solved BP problem (2).

Motivation with ℓ_2: As a motivation, consider the sequential measurements in the least squares setting. Assume that the measurement matrix A in (1) has full column rank (system is overdetermined). The least squares solution to (1) has the following analytical form

$$\hat{x}_0 = (A^TA)^{-1}A^Ty.$$

In order to compute $\hat{x}_0$ we have to solve a system of linear equations. Typically we use Cholesky or QR factorization of A^TA to solve such systems, for which the computational cost is $O(mn^2)$ [8]. Similarly, the least squares solution for an overdetermined system in (3) is

$$\hat{x}_1 = (A^TA + b^Tb)^{-1}(A^Ty + b^Tw).$$

A naive way to compute $\hat{x}_1$ would be to solve this new system of equations from scratch. A more efficient way, however, is to update the previous solution $\hat{x}_0$ using $(A^TA)^{-1}$ and the rank-one update as

$$\hat{x}_1 = \hat{x}_0 + K_1(w - b\hat{x}_0),$$

where $K_1 = (A^TA)^{-1}b^T(1 + b(A^TA)^{-1}b^T)^{-1}$. This update procedure is the well established recursive least squares (RLS) method, where each update costs $O(mn)$ [9].

Homotopy with ℓ_1: In this paper, we show that the solution for BP can also be updated with the dynamic changes in the measurements. We use homotopy principles to update the solution in a series of rank-one updates.

Homotopy is a general framework in which we can continuously transform one problem into an easy but related problem. Then we traverse a homotopy path from the easy problem towards the original problem, while solving a sequence of simple intermediate problems. The progression on this homotopy path is controlled by the homotopy parameter. The homotopy path is followed by ensuring that certain *optimality conditions* are being maintained [1].

In order to build the homotopy for BP update we need some optimality conditions. We use the strong duality and complementary slackness between the primal and dual formulations of BP to derive these conditions. The dual program for (2) is

$$\text{maximize} \quad -\lambda^T y \quad \text{subject to} \quad \|A^T\lambda\|_\infty \leq 1, \tag{5}$$

where $\lambda \in \mathbb{R}^m$ is the dual optimization variable. The optimality conditions which must be obeyed by any primal-dual solution pair (x^*, λ^*) can be written as

$$A_\Gamma^T \lambda^* = -z, \quad \|A_{\Gamma^c}^T \lambda\|_\infty < 1, \quad Ax^* = y, \tag{Opt-BP}$$

where Γ is the support of x^* and z is its sign sequence on Γ. These conditions tell that primal-dual vectors (x^*, λ^*) are feasible and dual constraints are active (hold with equality) *only* on the indices corresponding to the set Γ. Throughout this paper, we use the assumption[1] that if the original signal is not recovered by BP, then the number of nonzero elements in its solution will be same as the number of measurements.

In section II we present the homotopy algorithm for the sequential measurements and in section III we discuss the update algorithm for the time varying signals. Section IV presents experimental performance results.

II. Sequential measurements

In this section we discuss the homotopy algorithm to update the solution of BP as the new measurements are sequentially added to the system. Assume that we have solved BP (2) for the system in (1) but did not recover the original signal. We add one new measurement $bx = w$ to the system as described in (3), for which the BP problem can be written as

$$\text{minimize} \quad \|\tilde{x}\|_1 \quad \text{subject to} \quad A\tilde{x} = y, \ b\tilde{x} = w. \tag{6}$$

Homotopy formulation: Our proposed homotopy formulation for (6) is

$$\text{minimize} \quad \|\tilde{x}\|_1 \quad \text{s.t.} \quad A\tilde{x} = y, \ b\tilde{x} = (1-\epsilon)bx_0 + \epsilon w, \tag{7}$$

where ϵ is the homotopy parameter and x_0 is the solution to (2). The dual formulation for (7) can be written as:

$$\begin{aligned} &\text{maximize} \quad -\lambda^T y - \nu^T[(1-\epsilon)bx_0 + \epsilon w] \\ &\text{subject to} \quad \|A^T\lambda + b^T\nu\|_\infty \leq 1, \end{aligned} \tag{8}$$

where $\lambda \in \mathbb{R}^m$ and $\nu \in \mathbb{R}$ are the dual optimization variables. The optimality conditions for any primal-dual solution pair (x_k, λ_k, ν_k) at any value of $\epsilon = \epsilon_k$ are

$$\begin{aligned} &A_\Gamma^T \lambda_k + b_\Gamma^T \nu_k = -z, \quad \|A_{\Gamma^c}^T \lambda + b_{\Gamma^c}^T \nu_k\|_\infty < 1, \\ &Ax_k = y, \quad bx_k = (1-\epsilon_k)bx_0 + \epsilon_k w, \end{aligned} \tag{Opt-S}$$

where Γ is the support of x_k and z is its sign sequence on Γ.

As we increase ϵ from 0 to 1 in (7), we move its solution from the old BP solution for (1) to the new BP solution for (3). Along this homotopy path parameterized by ϵ, there will be some critical values of ϵ where the support set Γ changes. We refer to these points as critical points. At every critical point along the homotopy path, one non-zero element of x at some index γ^- shrinks to zero (removing an element from Γ) and an inactive dual constraint becomes active at some index γ^+ (adding a new element to Γ). In the homotopy algorithm we traverse the homotopy path by increasing ϵ from 0 to 1 while jumping from one critical point to the next one.

Initialization: Assume that we already have the solution x_0 to (2), supported on the set Γ of size m with sign sequence z. The corresponding dual solution is λ_0 with m active dual constraints. When we add new measurement (at $\epsilon = 0$), a new dual constraint becomes active. This is because our assumption dictates that exactly $m+1$ dual constraints should be active, unless we have recovered the original signal x. We find an initial value of ν and the new value of λ at $\epsilon = 0$ in (8). From the optimality conditions in (Opt-S) we know that dual constraints are active on the set Γ and now one new constraint has to become active. Using these conditions, we first find an update direction $\partial\lambda$ as follows:

$$\begin{aligned} A_\Gamma^T(\lambda_0 + \partial\lambda) + \nu b_\Gamma^T &= -z \\ \partial\lambda &= -\nu (A_\Gamma^T)^{-1} b_\Gamma^T. \end{aligned}$$

Then we find the smallest value of ν such that a new dual constraint becomes active at an index γ (note that the value of ν is unconstrained in sign). Let us denote the new value of λ as $\lambda_0 = \lambda_0 + \partial\lambda$, ν as ν_0 and new support $\Gamma = \Gamma \cup \{\gamma\}$.

Homotopy algorithm: We can divide each step of the homotopy algorithm into two parts: *primal update* and *dual update*. During primal update we use primal feasibility conditions to compute the primal update direction ∂x. Then we find the new critical value of ϵ such that an existing element in x at index γ^- shrinks to zero. During dual update we use the primal update information to compute the dual update direction $\partial\lambda$. Then we update the dual vectors such that one new constraint becomes active at index γ^+.

Primal update: Let us assume that we already have the primal-dual solutions (x_k, λ_k, ν_k) for (7) and (8) at $\epsilon = \epsilon_k$ with primal support Γ. Denote $G = \begin{bmatrix} A \\ b \end{bmatrix}$. The primal update direction ∂x that increases ϵ from ϵ_k to a slightly larger value ϵ_k^+, can be computed as follows

$$G(x_k + \theta\partial x) = \begin{bmatrix} \mathbf{0}_m \\ (1-\epsilon_k^+)bx_0 + \epsilon_k^+ w \end{bmatrix}$$

[1]This condition is true for the Gaussian matrix with probability 1 and holds with high probability for other random matrices commonly used in CS [10].

$$\partial x = \begin{cases} (G_\Gamma)^{-1} \begin{bmatrix} \mathbf{0}_m \\ -bx_0 + w \end{bmatrix} & \text{on } \Gamma \\ 0 & \text{on } \Gamma^c \end{cases},$$
$$\theta = \epsilon_k^+ - \epsilon_k,$$

where $\mathbf{0}_m$ denotes a zero vector at m indices corresponding to the old measurements. Find the smallest positive value of θ such that an element of x_k at some index $\gamma^- \in \Gamma$ shrinks to zero. This gives us new value of primal solution $x_{k+1} = x_k + \theta \partial x$ at $\epsilon_{k+1} = \epsilon_k + \theta$.

Dual update: For the dual update, let us denote $\Gamma_d = \Gamma \backslash \{\gamma^-\}$. Since $A_\Gamma^T \lambda_k + b_\Gamma^T \nu_k = -z$ and γ^- is removed from Γ, the dual constraints will be active on the set Γ_d at this new critical point. Therefore, the dual update directions $\partial\lambda$ and $\partial\nu$ can be computed as

$$A_{\Gamma_d}^T(\lambda_k + \delta\partial\lambda) + b_{\Gamma_d}^T(\nu_k + \partial\nu) = -z$$
$$\underbrace{A_{\Gamma_d}^T\lambda_k + b_{\Gamma_d}^T\nu_k}_{a_k} + \delta\,\underbrace{(A_{\Gamma_d}^T\partial\lambda + b_{\Gamma_d}^T)}_{b_k} = -z$$
$$\partial\nu = \delta, \quad \partial\lambda = -(A_{\Gamma_d}^T)^{-1} b_{\Gamma_d}^T.$$

Find the smallest value of δ such that one dual constraint at some index $\gamma^+ \in \Gamma_d^c$ becomes active. The sign of δ is selected so that the dual constraint for the outgoing element γ^- is not violated, i.e., $|a_k(\gamma^-) + \delta b_k(\gamma^-)| < 1$. This gives us the new values for dual solutions at $\epsilon = \epsilon_{k+1}$ as $\lambda_{k+1} = \lambda_k + \delta\partial\lambda$, $\nu_{k+1} = \nu_k + \delta$, and the primal support as $\Gamma = \Gamma_d \cup \{\gamma^+\}$.

Repeat this procedure until ϵ becomes equal to 1.

The main computational cost at each homotopy step involves computing the step sizes and update directions. Since at every homotopy step there is one element change in the support. We can quickly compute the update directions using rank-one updates. Therefore, the total cost of each homotopy step is same as a few matrix-vector multiplications i.e., $O(mn)$.

III. Time varying signals

In this section we discuss the homotopy algorithm to update the solution of BP as the underlying sparse signal changes slightly. For example some new elements appear in the signal on the previously zero locations or some existing elements shrink to zero. Assume that we have solved BP for system in (1) and we receive a new set of measurements as described in (4). The new optimization problem for BP becomes

$$\text{minimize } \|\tilde{x}\|_1 \text{ subject to } A\tilde{x} = \breve{y}. \tag{9}$$

Homotopy formulation: Our proposed homotopy formulation is

$$\text{minimize } \|\tilde{x}\|_1 \text{ subject to } A\tilde{x} = (1-\epsilon)y + \epsilon\breve{y}, \tag{10}$$

where ϵ is the homotopy parameter. Changing ϵ from 0 to 1 takes us from the already solved problem (2) to the desired problem (9). The dual formulation for (10) is

$$\text{maximize } -\lambda^T((1-\epsilon)y + \epsilon\,\breve{y}) \text{ subject to } \|A^T\lambda\|_\infty \le 1.$$

The optimality conditions for any primal dual solution pair (x_k, λ_k) at any value of $\epsilon = \epsilon_k$ are

$$A_\Gamma^T\lambda_k = -z, \;\; \|A_{\Gamma^c}^T\lambda\|_\infty < 1, \;\; Ax_k = (1-\epsilon_k)y + \epsilon_k\breve{y},$$

where Γ is the support of x_k and z is its sign sequence on Γ.

Initialization: Assume that we have the solution x_0 for (2) supported on the set Γ of size p (usually less than m). The corresponding dual solution is λ_0 with p active dual constraints. In the case of sequential measurements, the size of Γ for the initial step was equal to m. Otherwise (according to our assumption) we have already recovered the original signal. In the case of time varying signals, the solution x_0 can actually be the original signal x. This means that the size of Γ will usually be less than m. Whereas the homotopy update for (10) requires m active dual constraints. Thus in the initialization phase we update the dual vector λ_0 in such a way that $m-p$ inactive dual constraints become active.

From the optimality conditions we know that the dual constraints are active on the set Γ. Therefore any feasible dual update direction $\partial\lambda$ must lie in the null space of A_Γ^T, i.e., $A_\Gamma^T\partial\lambda = 0$. We use the projection of the new measurements $\breve{y}$ onto the null space of A_Γ^T to update the dual vector. The dual update direction $\partial\lambda$ can be computed as

$$\partial\lambda = -P_{N(A_\Gamma^T)}\breve{y},$$

where $P_{N(A_\Gamma^T)}$ denotes projection matrix for the null space of A_Γ^T. As we move in this direction, at some point (with proper step size θ) a new dual constraint becomes active at some index $\gamma \in \Gamma^c$. This gives the new value of $\lambda_0 = \lambda_0 + \theta\partial\lambda$ and support $\Gamma = \Gamma \cup \{\gamma\}$. We repeat this iterative projection procedure for updating dual vector until m dual constraints are activated.

If at any point the dual update direction is the zero vector, it indicates that the support of new solution $\breve{x}$ is a subset of the current support Γ. We can then compute $\breve{x}$ by solving the least squares problem on Γ for the measurements $\breve{y}$.

Homotopy algorithm: Similar to the homotopy for sequential measurement update, every step of the main homotopy algorithm can be divided into primal and dual updates.

Primal update: Let us assume we have the primal-dual solution pair (x_k, λ_k) for (10) at $\epsilon = \epsilon_k$ with primal support Γ of size m and sign sequence z. The primal update direction ∂x, which increases ϵ from ϵ_k by an infinitesimal amount to ϵ_k^+, can be computed as

$$A(x_k + \theta\partial x) = (1-\epsilon_k^+)y + \epsilon_k^+\breve{y}$$
$$\partial x = \begin{cases} -(A_\Gamma)^{-1}(y - \breve{y}) & \text{on } \Gamma \\ 0 & \text{on } \Gamma^c \end{cases}$$
$$\theta = \epsilon_k^+ - \epsilon_k.$$

Find the smallest positive value of θ such that an element in x_k at some index $\gamma^- \in \Gamma$ shrinks to zero. This gives us the new primal solution $x_{k+1} = x_k + \theta\partial x$ at $\epsilon_{k+1} = \epsilon_k + \theta$.

Dual update: Let us denote $\Gamma_d = \Gamma\backslash\{\gamma^-\}$. The dual constraints corresponding to indices in Γ at $\epsilon = \epsilon_{k+1}$ can

TABLE I
DYNAMIC UPDATE WITH SEQUENTIAL MEASUREMENTS IN BP.

n	S	H-steps	Avg-M
256	$[n/20] = 13$	18.17	52.83
	$[n/15] = 17$	20.07	63.83
	$[n/10] = 26$	24.55	83.91
	$[n/5] = 51$	28.83	129.73
512	$[n/20] = 26$	35.09	106.18
	$[n/15] = 34$	38.52	127.35
	$[n/10] = 51$	49.37	169.96
	$[n/5] = 102$	56.76	262.22

TABLE II
DYNAMIC UPDATE WITH TIME VARYING SIGNALS IN BP.

$n,\ m$	S	I-steps	H-steps
$n = 256,\ m = 128$	$[m/5] = 26$	8.82	0
	$[m/4] = 32$	21.07	0
	$[m/3] = 43$	63.23	90.81
$n = 512,\ m = 256$	$[m/5] = 51$	16.87	0
	$[m/4] = 64$	39	4.38
	$[m/3] = 85$	153.07	314.75

be written as

$$A_\Gamma^T(\lambda_k + \delta\partial\lambda) = \begin{bmatrix} -z_{\Gamma_d} \\ -z_{\gamma^-}(1-\delta) \end{bmatrix},$$

where the last element corresponds to the the outgoing index γ^- and $z_{\gamma^-} = \text{sign}(x_k(\gamma^-))$. Thus the dual update direction $\partial\lambda$ is

$$\partial\lambda = (A_\Gamma^T)^{-1} \begin{bmatrix} 0 \\ z_{\gamma^-} \end{bmatrix}.$$

Find the smallest positive value of δ such that one dual constraint at some index $\gamma^+ \in \Gamma_d^c$ becomes active. This gives us the new value of dual solution at $\epsilon = \epsilon_{k+1}$ as $\lambda_{k+1} = \lambda_k + \delta\partial\lambda$ and new support as $\Gamma = \Gamma_d \cup \{\gamma^+\}$.

Repeat this procedure until ϵ becomes equal to 1.

IV. SIMULATION RESULTS

A. *Sequential measurements*

We first present experiments for the homotopy algorithm discussed in section II for the sequential measurement update. We start with a sparse signal $x \in \mathbb{R}^n$ which contains ± 1 spikes at randomly chosen S locations. The measurement vector y is generated as in (1) using an $m \times n$ matrix A whose entries are i.i.d. $N(0, 1)$ and we solve (2). Then we add one new measurement $w = bx$ to the system, where entries of b are also $N(0, 1)$. We update the solution using the sequential homotopy algorithm discussed in section II. We sequentially add new measurements to the system until the original signal is recovered exactly. The results for 100 simulations with different values of n and S are summarized in Table I. Two numbers are recorded: average number of homotopy steps (H-steps) per new measurement and average number of measurements (Avg-M) required to recover x exactly. The initial matrix A was started with 32 and 64 rows for $n = 256$ and $n = 512$ respectively. These results demonstrate that the average number of homotopy steps per new measurement is very close to the sparsity of the underlying signal.

B. *Time varying signal*

In this experiment we start with a sparse signal x which contains ± 1 spikes at randomly chosen S locations. We take m measurements using a Gaussian matrix A and solve (2). Then we construct $\breve{x}$ as follows. First we perturb all nonzero entries of x with a random variable $N(0, 0.5^2)$. Then we add K new entries in x which are distributed according to $N(0, 1)$, at randomly chosen locations. Finally we remove K randomly chosen entries from x. The new measurements are computed as $\breve{y} = A\breve{x}$. We use the homotopy algorithm discussed in section III to update the solution. The results for 100 simulations are presented in Table II for different values of S, where $K = [S/5]$. Two numbers are presented in the table: number of iterations in the initialization phase (I-steps) of the update and number of iterations in the main homotopy algorithm (H-steps). As we can see from these results, if the signal is reasonably sparse (e.g., $S = m/5$) the initialization phase will detect all the elements in the updated signal. In the cases where signal is comparably dense (e.g., $S = m/3$), large number of homotopy steps are required for the update.

V. CONCLUSIONS

We presented two homotopy based algorithms to dynamically update the solution of basis pursuit problem. These algorithms enable us to quickly update BP solution when either new measurements are added to the system or the signal under observation changes slightly. The homotopy methods discussed are computationally inexpensive as each step involves a simple rank-one update. The total cost is significantly lower than solving an entirely new optimization problem. The simulation results demonstrate that for sparse signals, solution update requires a small number of homotopy steps.

REFERENCES

[1] M. S. Asif and J. Romberg, "Dynamic updating for ℓ_1 minimization," *Submitted*, 2009. Preprint available: http://arxiv.org/abs/0903.1443.

[2] 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.

[3] E. Candès and T. Tao, "The Dantzig selector: Statistical estimation when p is much larger than n," *Annals of Statistics*, vol. 35, no. 6, pp. 2313–2351, 2007.

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

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

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

[7] Y. Nesterov and A. Nemirovsky, "Interior Point Polynomial Methods in Convex Programming," *Studies in Applied Mathematics (SIAM)*, vol. 13, 1994.

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

[9] S. Haykin, *Adaptive filter theory*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1996.

[10] D. Malioutov, S. Sanghavi, and A. Willsky, "Compressed sensing with sequential observations," *IEEE International Conference on Acoustics, Speech, and Signal Processing,*, pp. 3357–3360, April 2008.