AN APPROXIMATE L0 NORM MINIMIZATION ALGORITHM FOR COMPRESSED SENSING

Mashud Hyder and Kaushik Mahata

School of Electrical Engineering and Computer Science, University of Newcastle,
Callaghan, NSW 2308, Australia.
md.hyder@studentmail.newcastle.edu.au, kaushik.mahata@newcastle.edu.au

ABSTRACT

ℓ^0 norm based signal recovery is attractive in compressed sensing as it can facilitate exact recovery of sparse signal with very high probability. Unfortunately, direct ℓ^0 norm minimization problem is NP-hard. This paper describes an approximate ℓ^0 norm algorithm for sparse representation which preserves most of the advantages of ℓ^0 norm. The algorithm shows attractive convergence properties, and provides remarkable performance improvement in noisy environment compared to other popular algorithms. The sparse representation algorithm presented is capable of very fast signal recovery, thereby reducing retrieval latency when handling high dimensional signal.

***Index Terms*—** Compressive Sensing, random matrices, nonconvex optimization, ℓ^0 minimization, ℓ^1 minimization

1. INTRODUCTION

Consider a real-valued, discrete-time signal $z \in \mathbb{R}^N$. Often there exists a basis $\{\psi_k, k = 1, \ldots, N\}$, which provides a so-called K-sparse representation of z, *i.e.* z admits a representation $z = \psi_1 x_1 + \cdots \psi_N x_N$ where only $K \ll N$ elements in the set $\{x_k, k = 1, \ldots, N\}$ are non-zero. This phenomenon is very common in real-world, *e.g.* when z represents an image then the wavelets provide a sparse representation. State-of-the-art compression algorithms exploit this fact as a small number K of adaptively chosen transform coefficients x_k are transmitted or stored rather than $N \gg K$ signal samples [1]. However, in the standard transform coding framework, the complete set $\{x_k\}_{k=1}^N$ of transform coefficients is computed from z and only a small subset is retained.

This observation has motivates compressed sensing (CS), where we do not compute all the components of $x = [x_1 \cdots x_N]'$. Instead we compute $M < N$ inner products between z and an another collection of vectors $\theta_i, i = 1, \ldots, M$ form the measurements $y_i = \theta_i' z$. Then

$$y := [y_1 \cdots y_M]' = \Theta' z = \Theta' \Psi x = \Phi x, \qquad (1)$$

where $\Theta = [\theta_1 \cdots \theta_M]$ and $\Psi = [\psi_1 \cdots \psi_N]$. Clearly, $y \in \mathbb{R}^M$ and $\Phi \in \mathbb{R}^{M \times N}$. Since $M < N$, recovery of z from y is ill-posed in general. However, when the vectors $\{\theta_i\}$ are incoherent with the columns of Ψ, (*i.e.* none of elements of the set $\{\theta_k, k = 1, \ldots, M\}$ admits a sparse representation with respect to the basis $\{\psi_k, k = 1, \ldots, N\}$) and M is large enough, then it is possible to recover x from y [2, 3]. Furthermore, often the measurement process is nonadaptive, meaning that Θ is fixed and does not depend z.

While the measurers process is linear, the reconstruction process is decidedly nonlinear. It requires solving the optimization problem

$$x = \arg\min_v \|v\|_0 \quad \text{subject to} \quad y = \Phi v, \qquad (2)$$

where

$$\|v\|_0 := \lim_{\epsilon \to 0} \{|v_1|^\epsilon + \cdots + |v_N|^\epsilon\},$$

which is simply the number of nonzero components in v which is also known as ℓ^0 norm of v. Unfortunately, solving (2) is NP-hard. For that reason, different alternative approaches are used to approximate the ℓ^0 norm [4, 5, 6, 7]. Chartard and W. Yin proposed an regularized IRLS algorithm which is a modified version of FOCUSS, where they replaced the ℓ^p norm by a weighted ℓ^2 norm [4, 8]. Although The algorithm works better for ℓ^0 approximation, it takes a lot of iteration to converge and is not robust to noise. G.H. Mohimani, M. Babaie-Zadeh and C. Jutten approximate the ℓ_0 norm by a smooth function [5]. This algorithm works better in noisy environment however when the level of sparsity increases it performs poorly.

Basis Pursuit (BP) is another popular approach where the ℓ^0 norm in (2) is replaced by ℓ^1 norm [9]. Although BP is significantly more approachable than approximate ℓ^0 norm, it performs pitiably in noisy environment. In this paper we propose an iterative approximate ℓ^0 norm (IALZ) algorithm to reconstruct sparse signal. IALZ differs from previous work for solving ℓ^0 norm and sparse representation in a number of ways. It uses a fixed point iteration based strategy and enjoys a significantly improved convergence speed. Finally, the most attractive feature is that IALZ shows a large performance improvement in noisy environment.

The research is supported by the Australian Research Council.

2. THE PROPOSED ALGORITHM IALZ

Our goal is to approximately formulate the objective function in (2) to which gradient based method can be applied. The Gaussian functions which seems useful for this purpose. Define

$$f(\alpha) = \exp\left(-\frac{\alpha^2}{2\sigma^2}\right). \quad (3)$$

Clearly $f(0) = 1$. In addition, for any given $\alpha > 0$ we have

$$\lim_{\sigma \to 0} f(\alpha) = 0.$$

Consequently, the function

$$F(x) = \sum_{k=1}^{N} f(x_k).$$

behaves like $N - ||x||_0$ when $\sigma \to 0$. This motivates the following approximate way of reformulating (2):

$$x_* = \arg\max_x F(x) \quad \text{subject to} \quad y = \Phi x. \quad (4)$$

As described, $F(x)$ looks like ℓ^0 norm of x when $\sigma \to 0$. However, for small value of σ, $F(x)$ contains a lot of local maxima. Consequently, it is very difficult to directly maximize this function for very small value of σ. However, as the value of σ grows, the function become smoother and smoother, and for a sufficiently large value of σ, the solution x will be the minimum-2 norm solution of the system $\Phi x = y$ [5] and there will not exists any local maxima. So, the standard procedure is taking a large σ initially and maximize the function by a gradient ascent method. When the change of x becomes less than some specified value, at that point σ is reduced by a factor ρ. Since the value of σ changes slowly, the gradient ascent algorithm is initialize not far from the actual maximum and it has much less possibility to trap in local maxima.

2.1. Algorithm Derivation

The Lagrangian $L(x, \nu)$ associated with the problem (4) is given by

$$L(x, \nu) = F(x) + \nu'(\Phi x - y) \quad (5)$$

where $\nu \in \mathbb{R}^{M\times 1}$ is the vector of Lagrange multipliers. Now (4) implies that there exists ν_* such that (x_*, ν_*) is a stationary point of $L(x, \nu)$, *i.e*,

$$\frac{\partial L(x_*, \nu_*)}{\partial x} = \frac{\partial F(x_*)}{\partial x} + \Phi'\nu_* = 0 \quad (6)$$
$$\frac{\partial L(x_*, \nu_*)}{\partial \nu} = \Phi x_* - y = 0$$

Also it is readily verified that

$$\frac{\partial F(x)}{\partial x_n} = -\frac{x_n}{\sigma^2} f(x_n) \ \Rightarrow\ \frac{\partial F(x)}{\partial x} = -\frac{1}{\sigma^2} W(x)x. \quad (7)$$

Define $W(x) = \mathrm{diag}\{f(x_1).....f(x_n)\}$. Then (6)-(7) gives

$$x_* = \sigma^2 W^{-1}(x_*)\Phi'\nu_* \quad (8)$$

Substituting for x_* in the second equation of (6), solving for ν_* and substituting this expression for ν_* in (8) then

$$x_* = W^{-1}(x_*)\Phi' \left[\Phi W^{-1}(x_*)\Phi'\right]^{-1} y. \quad (9)$$

Equation (9) is nonlinear, and cannot be solved analytically. However one possible avenue is to use (9) in a fixed point iteration, which is the central idea of our algorithm.

Lemma 1 *Let us define the map* $g : \mathbb{R}^N \to \mathbb{R}^N$ *such that*

$$g(x) = W^{-1}(x)\Phi' \left[\Phi W^{-1}(x)\Phi'\right]^{-1} y. \quad (10)$$

Then $\Phi g(x) = y$*. Let* $x \in \mathbb{R}^N$ *such that* $\Phi x = y$*,* $g(x) \neq x$*, and*

$$\left[\frac{\partial F(x)}{\partial x}\right] \neq 0. \quad (11)$$

Then there exists λ *satisfying* $0 < \lambda \leq 1$ *such that*

$$F\{\lambda g(x) + (1-\lambda)x\} > F(x). \quad (12)$$

Proof: Pre-multiplying (10) by Φ we verify that $\Phi g(x) = y$ for all $x \in \mathbb{R}^N$. Now suppose x satisfies $\Phi x = y$, and (11) holds. Then using (7) we get

$$\begin{aligned} &\left[\frac{\partial F(x)}{\partial x}\right]' \{g(x) - x\} \\ &= \frac{1}{\sigma^2} x'\{\, W(x) - \Phi' \left[\Phi W^{-1}(x)\Phi'\right]^{-1} \Phi \,\}x \\ &= \frac{1}{\sigma^2}\{W^{0.5}(x)x\}'\Pi\{W^{0.5}(x)x\}. \quad (13)\end{aligned}$$

where

$$\Pi = I - W^{-0.5}(x)\Phi' \left[\Phi W^{-1}(x)\Phi'\right]^{-1} \Phi W^{-0.5}(x)$$

is the orthogonal projection operator on the nullspace of $\Phi W^{-0.5}(x)$. From (13) it is clear that

$$\left[\frac{\partial F(x)}{\partial x}\right]' \{g(x) - x\} \geq 0, \quad (14)$$

with equality is satisfied only if $W^{0.5}(x)x$ be in the columnspace of $W^{-0.5}(x)\Phi'$, which means

$$W^{0.5}(x)x = W^{-0.5}(x)\Phi'\beta \ \Rightarrow\ \frac{\partial F(x)}{\partial x} = \Phi'\frac{-\beta}{\sigma^2}$$

for some $\beta \neq 0$. This implies that we have equality in (14) only if

$$\frac{-\beta}{\sigma^2} = \left[\Phi W^{-1}(x)\Phi'\right]^{-1} y \ \Rightarrow x = g(x).$$

However $x \neq g(x)$ by assumption. Hence we have a strict inequality in (14). This means the innerproduct between $g(x) - x$ and the gradient of $F(x)$ is always positive. So, the value of $F(x)$ increases along the line joining x and $g(x)$ and there exists some λ such that $0 < \lambda \leq 1$ for which (12) holds. □

Table 1. IALZ Algorithm

```
- Initialization:
1. Initialize $x^{(0)}$ to the minimum 2-norm solution of $\Phi x = y$,
i.e. $x^{(0)} = \Phi^T(\Phi\Phi^T)^{-1}y$.
2. Set $\sigma = 1$ and choose decreasing factors $\rho$ and $\gamma$
repeat
  1. Set $\lambda = 1$
  2. Compute $x^{(i+1)} = \lambda g(x^{(i)}) + (1 - \lambda)x^{(i)}$
  3. Backtracking. If $F(x^{(i+1)}) < F(x^{(i)})$
     change $\lambda = \gamma\lambda$ and repeat step 2, 3.
  4. If $\tau = ||x^{(i+1)} - x^{(i)}||_2 < \frac{\sigma}{\rho}$
     change $\sigma = \frac{\sigma}{\rho}$
until $\tau < 10^{-8}$
```

2.2. Algorithm

Based on the main idea of the previous section, the final algorithm is given in Table 1. By choosing a proper decreasing factor γ i.e. $0 < \gamma < 1$, the algorithm can be accelerated to converge the optimal solution rapidly. In our experiment we fixed γ to 0.5. The final value of σ depends on the noise level. For noiseless sparse recovery, ρ was chosen to 10 and σ was allowed to decreased near to zero. However, in noisy case, it should be left at some smaller value as the system can not accurately approximate the optimal x and the solution fluctuate randomly.

3. EXPERIMENTS

This section illustrates experimentally that IALZ is a powerful algorithm for signal recovery. Two type of experiments are presented: exact recovery and approximate recovery. Each set of experiments is repeated 100 times with different random signals and randomly select entries of $M \times N$ matrix Φ from a mean-zero Gaussian distribution, then scale the columns to have unite 2-norm. The same Φ and x are used in each algorithm for a fair comparison. In exact recover, for each sparsity level K, we randomly choose the support of x, and randomly setting the peaks to either $+1$ or -1. The initial value of σ is 1. The iteration with a fixed σ is run until the change of x in relative 2-norm from the previous iterate is less than $\sigma/10$, at that point σ is decreased to $\sigma/10$. We compare our algorithm with regularized IRLS [1] and Basis Pursuit (ℓ^1-Magic[2] package). Our simulations are performed in MATLAB7 environment using an Intel 2 GHz processor with 1GB of memory.

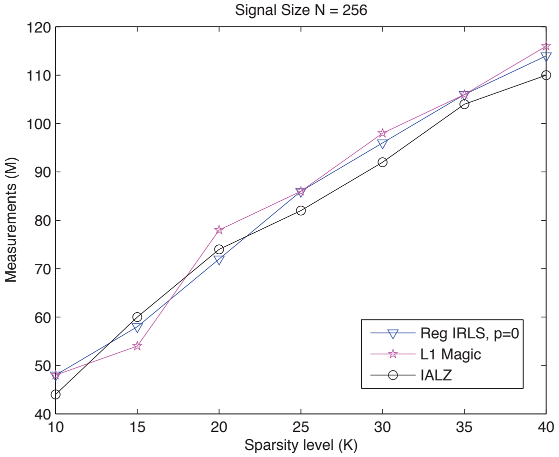

Fig. 1. The number of measurements (M) required to exact recovery of a signal for different sparsity levels Kin signal size $N = 256$.

The first plot, Figure 1 shows the minimum number of measurements M are necessary to recover a K-sparse signal in $\Re^N$ with high probability. As expected, in comparison with regularized IRLS and ℓ^1-Magic, IALZ requires much less measurements to recover signal for a fixed level of sparsity. Figure 2 presents another view of the same data. The first figure shows what percentage (of the 100 trial signals) of full signal were recovered correctly as a function of K for fixed $M = 100$. In this case, both IRLS with $p = 0$, ℓ^1-magic and IALZ, all performs almost similarly. However, the improvement of IALZ is noted in the second figure which displays the percentage of signal's component recovered correctly as a function of the sparsity level. Here, We assumed that a component x_n of signal x $(N \times 1)$)is recovered exactly if the recovered component is x_n^r and $|x_n - x_n^r| < \mu$, where μ is a small positive constant. It is clear from the figure that IALZ outperforms compared to regularized IRLS with $p = 1$ and $p = 0$. Moreover, IALZ performed better than Basis Pursuit for low level of sparsity.

Again, in terms of required iterations and computation time to reconstruct a signal, IALZ shows excellent performance improvement compared to IRLS. Figure 3 shows that regularized IRLS requires almost 3 times more iteration than IALZ in all cases for fixed measurements $M = 100$ with different sparsity. The computation time also follows same curves. This result indicate that IALZ is faster than other ℓ^0 algorithms.

For approximate recovery, we use the approach of SL0[3] [5] where sparse sources were generated by using Mixture of Gaussian (MoG) model:

$$x_i \sim p.\aleph(0, \epsilon_{on}) + (1 - p).\aleph(0, \epsilon_{off}) \qquad (15)$$

where p denotes probability of activity of the sources. ϵ_{on} and ϵ_{off} are the standard deviations of the sources in ac-

[1] Source code http://igorcarron.googlepages.com/cscodes

[2] http://www.acm.caltech.edu/l1magic/

[3] Source code http://igorcarron.googlepages.com/cscodes

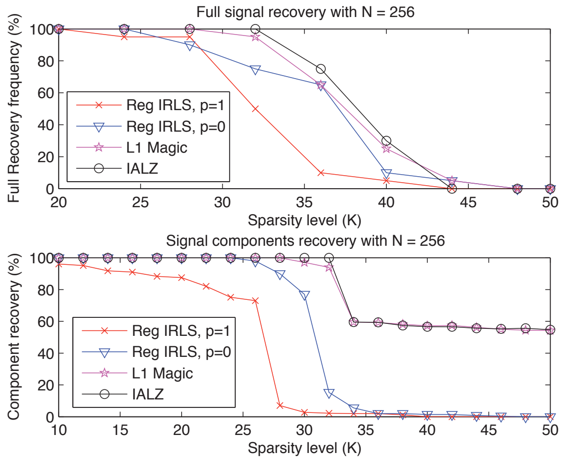

Fig. 2. The percentage of signals recovered as a function of the sparsity level K for fixed number of measurements $M = 100$.

tive and inactive mode, respectively. The mixtures are generated using the noisy model $y = \Phi x + \eta$, where η is an additive white Gaussian noise with variance $\epsilon_n I_m$ (I_m is the $m \times m$ identity matrix). The values used for the experiment are $p = 0.1$, $\epsilon_{on} = 1$, $\epsilon_{off} = 0.01$ and $\epsilon_n = 0.01$. The sequence of σ in IALZ was fixed to six different levels i.e. $\sigma = [1, 0.2, 0.1, 0.05, 0.03, 0.02]$. The recovery performance of different algorithms is shown in Figure 4. As expected, IALZ shows a massive performance improvement compared to other two algorithms. Also note that Basis Pursuit has very poor performance in noisy environment.

4. REFERENCES

[1] R. A. DeVore, B. Jawerth, and B. J. Lucier, "Image compression through wavelet transform coding," *IEEE Trans. Inf. Theory*, vol. 38, pp. 719–746, March 1992.

[2] E. J. Candés, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, pp. 489–509, February 2006.

[3] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, pp. 1289–1306, April 2006.

[4] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *33rd International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008, pp. 3869 – 3872.

[5] G. H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "Fast sparse representation based on smoothed l0 norm," in *7th International Conference on Independent Component Analysis and Signal Separation*, 2007, pp. 389–396.

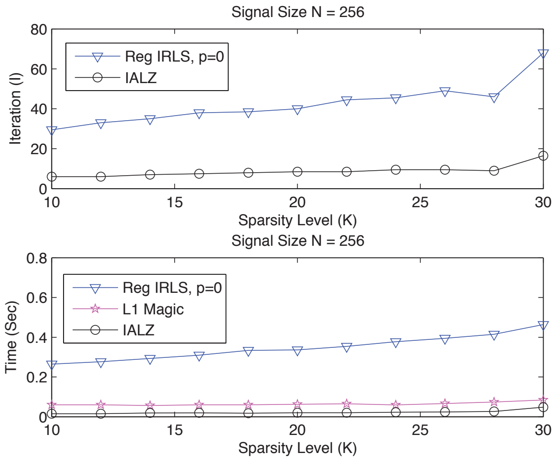

Fig. 3. The number of iterations (I) and computation time required to exact recovery of a signal as a function of sparsity levels K for fixed measurements $M = 100$.

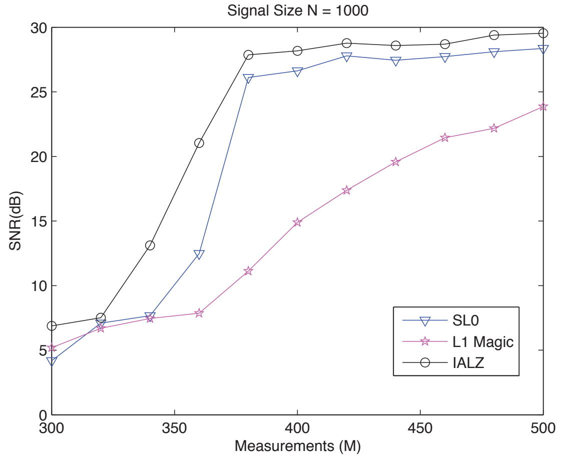

Fig. 4. The Signal to Noise ratio (dB) achieved to recover a noisy signal as a function of different measurements M.

[6] L. Mancera and J. Portilla, "L0-norm-based sparse representation through alternate projections," in *International Conference on Image Processing*. IEEE, 2006, pp. 2089–2092.

[7] C.R. Berger, J. Areta, K. Pattipati, and P. Willett, "Compressed sensing - a look beyond linear programming," in *33rd International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008, pp. 3857–3860.

[8] B. D. Rao and K. Kreutz-Delgado, "An affine scaling methodology for best basis selection," *EEE Trans. Signal Processing*, vol. 47, pp. 187–200, January 1999.

[9] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, pp. 33–61, 1998.