FAST L0-BASED SPARSE SIGNAL RECOVERY

Yingsong Zhang, Nick Kingsbury

Signal Processing & Communication Laboratory
Department of Engineering, University of Cambridge, U.K.
{yz304, ngk10}@cam.ac.uk

ABSTRACT

This paper develops an algorithm on finding sparse signals from limited observations of a linear system. We assume an adaptive Gaussian model for sparse signals. This model results in a least square problem with an iteratively reweighted L2 penalty that approximates the L0-norm. We propose a fast algorithm to solve the problem within a continuation framework. In our examples, we show that the correct sparsity map and sparsity level are gradually learnt during the iterations even when the number of observations is reduced, or when observation noise is present. In addition, with the help of sophisticated interscale signal models, the algorithm is able to recover signal to a better accuracy with reduced number of observations.

1. INTRODUCTION

Sparsity has been widely used as a constraint when searching for feasible solutions of ill-conditioned inverse problems and underdetermined systems of linear equations. These are known to be problems of great practical interest.

Recovering the sparsest signal $\mathbf{x}$ from limited observations $\mathbf{y}$ can be expressed as

$$\min \|\mathbf{x}\|_0 \quad \text{subject to} \quad \|\mathbf{y} - \Phi\mathbf{x}\|_2 \leq \xi \qquad (1)$$

where Φ is known and of size $M \times N$, $\|\mathbf{x}\|_0$ denotes the number of non-zero elements of $\mathbf{x}$ and ξ quantifies the upper bound of measurement noise. However, this optimization problem is non-convex and the only known searching method for the exact solution is an intractable combinatorial search, which is well known to be NP-hard. Instead, people use relaxed measures for $\|\mathbf{x}\|_0$ to bring in efficient numerical methods with guaranteed performance. A popular such relaxed measure for sparseness is the L1 norm. L1 minimization produces a convex minimization problem, but its disadvantage is poor computation speed for large problems. The L1 penalty problem is often solved by linear programming, and one popular method is the interior point method, whose best currently attained complexity is $O(M^2N^{1.5})$ [1]. An alternative approach to L1 minimization is the use of greedy algorithms. Greedy algorithms can improve the computation speed, but most approaches (e.g. OMP [2], ROMP and CoSaMP [1]) require accurate knowledge of the signal's sparsity level, or are too complicated to implement well [1].

The geometry of the L_p ($0 < p < 1$)ball gives better approximations for sparsity. Candès et al [3] propose the iterative reweighted L1 (IRL1) minimization which outperforms the L1 minimization in the sense that substantially fewer measurements are needed for exact recovery. Gorodnitsky and Rao [4] propose FOCUSS, as an iteratively reweighted least square (IRLS) minimization for finding sparse solutions. At each iteration, FOCUSS solves the weighted L2 minimization with weights $\frac{1}{x_j}$ subject to $\mathbf{y} = \Phi\mathbf{x}$. As the iteration proceed, the zero coefficients are identified and removed from subsequent iterations, and then enforced to be zero. Chartrand and Yin [5] introduce a regularizer ϵ into the weights $\frac{1}{x_j+\epsilon}$ to avoid the artificial zero enforcement, which improves the performance of FOCUSS to be competitive with IRL1 [3]. In the same paper, they also show with numerical experiments that IRLS minimization can enhance the range of sparsity for which a given number of measurement M can possibly lead to the perfect reconstruction. For the case of random Gaussian measurements, Candès and Tao [6] show that the dependence of the sufficient number of measurements M on the signal size N decreases as $p \to 0$.

Daubechies et al [7] study the convergence of IRLS and prove local super-linear convergence when the IRLS is tailored to mimic L_p minimization with $0 < p \leq 1$. Improved numerical results are also shown to justify going to lower p.

In parallel, Baraniuk et al proposed model-based compressed sensing (CS), which exploits signal structure to reduce the freedom of a sparse / compressible signal. Hence it further reduces the number of measurements M required for stable recovery of the signal, and it can reduce recovery artefacts which are not part of the model [8]; but it does require clear structural constraints to be defined.

In this paper, we develop ideas from [4, 5, 7] and propose a fast L0-mimicking reweighted L2-norm minimization algorithm, which is equivalent to Bayesian MAP estimation with an adaptive Gaussian prior. This algorithm

requires almost no parameter tuning and is simple to implement. In addition, the Gaussian model is flexible enough to contain typical signal structure models, which help to reduce the number of observations. One example test on the integration of such a signal model is presented on the Heavisine signal. We also exploit the geometry implications of having regularizer ϵ in the weights, and the resulting geometry structure leads to a continuation strategy for finding optimal sparse solutions in Section 3.

2. L0 reweighted-L2 minimization

2.1. Basic model

Consider the noisy system

$$\mathbf{y} = \Phi\mathbf{x} + \mathbf{n} \tag{2}$$

where we assume Φ is known, $\mathbf{n}$ is Gaussian noise with zero mean and variance ν^2, and the prior of $\mathbf{x}$ is an adaptive Gaussian model such that $p(\mathbf{x}) \propto \sqrt{|\mathbf{S}|}\exp(-\frac{1}{2}\mathbf{x}^T\mathbf{S}\mathbf{x})$, where $\mathbf{S}$ is a diagonal matrix, whose j^{th} diagonal entry $S_j = 1/\sigma_j^2$.
We further assume an independent prior $\exp(-\epsilon^2/\sigma_j^2)$ for each σ_j. Then we obtain the log MAP function:

$$J(\mathbf{x},\mathbf{S}) = \nu^2\left(\mathbf{x}^T\mathbf{S}\mathbf{x} - \ln|\mathbf{S}| + \epsilon^2\sum_j S_j\right) + \|\mathbf{y}-\Phi\mathbf{x}\|^2 \tag{3}$$

This results in the following iteration rules:

$$\left.\begin{aligned} \mathbf{x} &= \arg\min_{\mathbf{x}} J(\mathbf{x},\mathbf{S}) = (\Phi^T\Phi + \nu^2\mathbf{S})^{-1}\Phi^T\mathbf{y} \\ \sigma_j^2 &= \arg\min_{\sigma_j^2} J(\mathbf{x},\mathbf{S}) = |x_j|^2 + \epsilon^2 \\ S_j &= \frac{1}{\sigma_j^2} = \frac{1}{|x_j|^2+\epsilon^2} \qquad \forall\, j \end{aligned}\right\} \tag{4}$$

Because $\lim_{\sigma^2\to 0}\exp(-\epsilon^2/\sigma^2) = 0$, prior $\exp(-\epsilon^2/\sigma_j^2)$ actually prevents σ^2 getting too small to avoid numerically instability. Meanwhile, $\exp(-\epsilon^2/\sigma^2) \to 1$ rapidly with increasing σ^2. Therefore, this prior can be regarded as an approximation to the lower bounded uniform prior $U(\epsilon^2, +\infty)$ and ϵ^2 may be regarded as a stabilizer to avoid infinity at this point to keep the argument simple, although we show in section 3 that ϵ^2 has a more fundamental meaning.
The iteration rule eq(4) is effectively reweighted least squares minimization, which promotes sparsity by mimicking the L0 norm. It can be viewed as a relaxation of the IRLS studied by Chartran and Yin [5] and Daubechies et al [7], whose iteration rule is

$$\mathbf{x} = \mathbf{S}^{-\frac{1}{2}}(\Phi\mathbf{S}^{-\frac{1}{2}})^{\dagger}\mathbf{y}, \quad S_j = \frac{1}{|x_j|^2+\epsilon^2} \qquad \forall\, j \tag{5}$$

where † denotes the pseudo inverse. This is because

$$\begin{aligned}\mathbf{S}^{-\frac{1}{2}}(\Phi\mathbf{S}^{-\frac{1}{2}})^{\dagger} &= \mathbf{S}^{-\frac{1}{2}}\lim_{\nu^2\to 0}(\mathbf{S}^{-\frac{1}{2}}\Phi^T\Phi\mathbf{S}^{-\frac{1}{2}} + \nu^2\mathbf{I})^{-1}\mathbf{S}^{-\frac{1}{2}}\Phi^T \\ &= \lim_{\nu^2\to 0}(\Phi^T\Phi + \nu^2\mathbf{S})^{-1}\Phi^T\end{aligned}$$

if $\mathbf{S}$ is diagonal and Φ has full rank, which is typical in compressive sensing [7].

2.2. Fast algorithm: $\mathrm{L_0RL_2}$

The iteration rule eq(4) requires the inversion of matrix $(\Phi^T\Phi + \nu^2\mathbf{S})$ with dimensions $N \times N$, which is computationally demanding in the context of an iterative algorithm, particularly if N is large such as for an image or 3-D dataset. In this section, we develop a fast algorithm for eq(4), which we call L0 reweighted-L2 minimization and denote as $\mathrm{L_0RL_2}$. This involves the application of a majorization-minimization (MM) technique [9], together with recent subband-adaptive MM, proposed by Vonesch and Unser in [10] and further generalized in [11, 12].

We apply the MM principle to cut down the computation complexity of eq(4); and hence we introduce

$$\bar{J}(\mathbf{x},\mathbf{S},\mathbf{z}) = J(\mathbf{x},\mathbf{S}) + \alpha(\mathbf{x}-\mathbf{z})^T(\mathbf{x}-\mathbf{z}) - \|\Phi(\mathbf{x}-\mathbf{z})\|^2 \tag{6}$$

where α must be no less than the radius of $\Phi^T\Phi$ to ensure $\bar{J}(\mathbf{x},\mathbf{S},\mathbf{z}) \geq J(\mathbf{x},\mathbf{S})\ \forall\, \mathbf{z},\mathbf{x}$. By setting $\frac{\partial\bar{J}}{\partial\mathbf{x}} = 0$, $\frac{\partial\bar{J}}{\partial\mathbf{z}} = 0$ and $\frac{\partial\bar{J}}{\partial\sigma_j^2} = 0$, we have the following iteration rules:

$$\left.\begin{aligned} \mathbf{x}_{n+1} &= (\alpha\mathbf{I} + \nu^2\mathbf{S}_n)^{-1}\left[(\alpha\mathbf{I} - \Phi^T\Phi)\mathbf{z}_n + \Phi^T\mathbf{y}\right] \\ \mathbf{z}_{n+1} &= \arg\min_{\mathbf{z}} \bar{J}(\mathbf{x}_{n+1},\mathbf{S}_n,\mathbf{z}) = \mathbf{x}_{n+1} \\ \mathbf{S}_{n+1} &= \mathrm{diag}\left(\left[\frac{1}{|x_{j,n+1}|^2+\epsilon^2}\right]_{j=1,\cdots,N}\right) \end{aligned}\right\} \tag{7}$$

The above rules are simple and have complexity of only $O(MN)$, or much less if Φ has a fast implementation. If $\psi = \left(\alpha\mathbf{I} + \nu^2\mathbf{S}\right)^{-1}(\alpha\mathbf{I} - \Phi^T\Phi)$, it can be shown that the convergence rate (in a fixed-point sense) of $\bar{J}(\mathbf{x},\mathbf{S},\mathbf{z})$ to $J(\mathbf{x},\mathbf{S})$ is fast if the eigenvalues of ψ is close to zero.

By using a subspace/subband-dependent MM algorithm, Vonesch and Unser [10] developed the Shannon-wavelet-based subband adaptive shrinkage algorithm, which accelerates the convergence of $\mathrm{L_0RL_2}$'s analogue, the L1-based iterative shrinkage/thresholding algorithm (ISTA). This work is generalized in [12] for the dual-tree complex wavelet frame and in [11] for arbitrary tight wavelet frames. We follow the notation in [11] and introduce the vector $\boldsymbol{\alpha} = [\alpha_1 \ldots \alpha_K]$ and the diagonal operator Λ_α that multiplies the k^{th} subspace / subband by α_k:

$$(\Lambda_\alpha\mathbf{x})_k = \alpha_k\mathbf{x}_k \quad \text{for } k = 1\cdots K$$

where $\mathbf{x}_k$ is a masked version of $\mathbf{x}$ with non-zeros only in the subspace / subband k. Now we have the new auxiliary function

$$J_\alpha(\mathbf{x},\mathbf{S},\mathbf{z}) = J(\mathbf{x},\mathbf{S}) + (\mathbf{x}-\mathbf{z})^T\Lambda_\alpha(\mathbf{x}-\mathbf{z}) - \|\Phi(\mathbf{x}-\mathbf{z})\|^2 \tag{8}$$

where α_k may be optimized independently for each subspace / subband to be the minimum such that

$$\alpha_k \mathbf{x}_k^T\mathbf{x}_k \geq \|\Phi\mathbf{x}_k\|^2 \quad \text{for any } k \text{ and } \mathbf{x}.$$

This is equivalent to requiring that $\Lambda_\alpha - \Phi^T\Phi$ is positive semi-definite[1]. Minimizing $J_\alpha(\mathbf{x},\mathbf{S},\mathbf{z})$ in eq(8) then results in the following iteration rules:

$$\left.\begin{aligned}\mathbf{x}_{n+1} &= (\Lambda_\alpha + \nu^2\mathbf{S}_n)^{-1}\left[(\Lambda_\alpha - \Phi^T\Phi)\mathbf{z}_n + \Phi^T\mathbf{y}\right] \\ \mathbf{z}_{n+1} &= \arg\min_{\mathbf{z}} J_\alpha(\mathbf{x}_{n+1},\mathbf{S}_n,\mathbf{z}) = \mathbf{x}_{n+1}\end{aligned}\right\} \tag{9a}$$

$$\mathbf{S}_{n+1} = \mathrm{diag}\left(\left[\frac{1}{|x_{j,n+1}|^2+\epsilon^2}\right]_{j=1,\cdots,N}\right) \tag{9b}$$

We call this the $\mathrm{L_0RL_2}$ algorithm.

Remark 1. If desired, $\mathbf{S}$ can be updated in eq(9b) only after several iterations of eq(9a), which makes the rules of eq(9) approximate the rules of eq(4) better while still monotonically reducing the cost function J_α in eq(8).

3. CONTINUATION STRATEGY

Our algorithm solves the minimization of $J(\mathbf{x},\mathbf{S})$ in eq(3), which involves parameters ν and ϵ. To obtain a good sparse solution, ν must be chosen to be neither too small to impose sparsity nor too big to fit the solution to the observations after a reasonable number of iterations. For example, when $\nu \to 0$, the algorithm converges so slowly that it often stops at a non-sparse solution under some numerically convergent criteria, whereas when $\nu \to \infty$, the solution becomes zero. Meanwhile, ϵ, as stabilizer, should be small enough to avoid artefacts. In fact, ϵ has a more fundamental meaning: it decides the geometry of the penalty function. Theoretical analysis shows ϵ and ν should be related in order to achieve convergence to a good solution. Therefore, we fit the algorithm to the continuation framework described below, which automatically selects proper values for ν and ϵ so as to achieve this.

3.1. Geometry of the penalty function

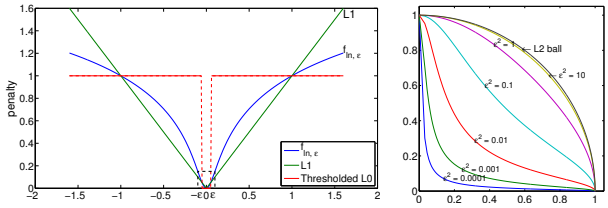
Substituting into eq(8) with eq(9b) and eq(3), and $\mathbf{z} = \mathbf{x}$, and introducing $-N\ln\epsilon^2$ into eq(8), it gives

$$F(\mathbf{x}_n) = \nu^2\left(N + \sum_{j=1}^{N}\ln\frac{x_{j,n}^2+\epsilon^2}{\epsilon^2}\right) + \|\mathbf{y}-\Phi\mathbf{x}_n\|_2^2 \tag{10}$$

Then the iteration rules of eq(9) result in:

$$F(\mathbf{x}_{n+1}) \leq J_\alpha(\mathbf{x}_{n+1},\mathbf{z}_{n+1},\mathbf{S}_n) - N\ln\epsilon^2 \leq F(\mathbf{x}_n)$$

[1] In fact,convergence is also guaranteed, if $2\Lambda_\alpha - \Phi^T\Phi$ is positive definite. Combettes and Wajs prove this result in [13].



(a) Comparison with L1 norm (b) Unit Ball

Fig. 1. Geometry of penalty function $P_\epsilon(\mathbf{x})$, eq(11).

Hence $F(\mathbf{x})$ is effectively the underlying cost function that the algorithm is minimizing. Thus the penalty function which the algorithm uses to promote sparsity is:

$$P_\epsilon(\mathbf{x}) = \sum_{j=1}^{N}\ln\frac{x_j^2+\epsilon^2}{\epsilon^2} \tag{11}$$

The geometry of the log-sum penalty function $F(\mathbf{x})$ lends itself well to detecting sparsity. In Figure 1(a) we plot $f_{\ln,\epsilon} = C\ln\frac{x^2+\epsilon^2}{\epsilon^2}$, $\|x\|_1$, and thresholded $\|x\|_0$; where $\epsilon = 0.1$, $C = \frac{1}{\ln(1+1/\epsilon^2)}$ is a scalar such that $f_{\ln,\epsilon}(1) = 1$, and the threshold is 0.05 where $f_{\ln,\epsilon}(x)$ and $\|x\|_1$ first intersect. Figure 1(a) shows that the log function imitates the thresholded L0 norm better than the L1 norm. Not only does it penalize much less on large coefficients than the L1 norm, it also penalizes more heavily in the region $[0.05,\ 1]$. In fact, as ϵ decreases towards 0, the log function $f_{\ln,\epsilon}(x)$ approximates the true L0 norm.

Figure 1(b) shows the interesting effect of ϵ on the geometry of $P_\epsilon(\mathbf{x})$. When ϵ is small, the geometry of $P_\epsilon(\mathbf{x})$ approximates that of the L0 norm; when ϵ becomes larger, the geometry of $P_\epsilon(\mathbf{x})$ approaches the L2 ball. Daubechies et al have shown experimentally that with weights that gradually move the penalty from the L1 norm to the L0 norm, IRLS achieves a higher success rate in exact sparse signal recovery (see figure 8.4 of [7]). This finding inspires the idea of reducing ϵ gradually.

3.2. Conditions for convergence

We assume $\mathbf{x}^*$ is the equilibrium solution for a given ϵ and ν, therefore from eq(9):

$$(\Lambda_\alpha + \nu^2\mathbf{S}^*)\mathbf{x}^* = (\Lambda_\alpha - \Phi^T\Phi)\mathbf{x}^* + \Phi^T\mathbf{y} \tag{12}$$

$$\mathbf{S}_{jj}^* = S_j^* = \frac{1}{|x_j^*|^2+\epsilon_n^2}, j = 1...N \tag{13}$$

Let the errors from $\mathbf{x}^*$ at iteration $n+1$ be $\eta_{n+1} = \mathbf{x}_{n+1} - \mathbf{x}^*$, $\eta_n = \mathbf{x}_n - \mathbf{x}^*$. For convergence we require $\|\eta_{n+1}\| \leq \|\eta_n\|$. By the iteration rules in eq(9), we have

$$(\Lambda_\alpha + \nu^2\mathbf{S}_n)(\eta_{n+1}+\mathbf{x}^*) = (\Lambda_\alpha - \Phi^T\Phi)(\eta_n+\mathbf{x}^*) + \Phi^T\mathbf{y}.$$

Substituting eq(12) into this gives

$$\eta_{n+1} = \left(\Lambda_\alpha + \nu^2\mathbf{S}_n\right)^{-1}\left((\Lambda_\alpha - \Phi^T\Phi)\eta_n + \nu^2(\mathbf{S}^* - \mathbf{S}_n)\mathbf{x}^*\right) \tag{14}$$

We denote the jth element of vector η_n as $\eta_{j,n}$. As long as we make sure that

$$|\eta_{j,n+1}| \leq |\eta_{j,n}| \tag{15}$$

then $\|\eta_{n+1}\| < \|\eta_n\|$ always holds. Assuming Λ_α is properly set such that the eigenvalues of $\left(\Lambda_\alpha - \Phi^T\Phi\right)$ are close to zero[2], by substituting eq(13) into eq(15), we require

$$\frac{|\eta_{j,n+1}|}{|\eta_{j,n}|} \approx \left|(\Lambda_\alpha + \nu^2\mathbf{S}_n)^{-1}_{jj} \frac{\nu^2(|x^*_j| + |x_{j,n}|)x^*_j}{(|x^*_j|^2 + \epsilon^2)(|x_{j,n}|^2 + \epsilon^2)}\right| \leq 1 \tag{16}$$

Hence, after some algebra:

$$\nu^2(\Lambda_\alpha)^{-1}_{jj} \leq \frac{|x^*_j|^2|x_{j,n}|^2 + \epsilon^2(|x^*_j|^2 + |x_{j,n}|^2) + \epsilon^4}{|x^*_j||x_{j,n}| - \epsilon^2} \tag{17}$$

for any j which satisfies $|x^*_j||x_{j,n}| > \epsilon^2$. Because the RHS of the above inequality is always larger than $8\epsilon^2$ (maximum when the derivative of the RHS equals zero), we set

$$\nu^2 = 8\epsilon^2 \min(\boldsymbol{\alpha}) \tag{18}$$

Remark 2. eq(18) is not sufficient to ensure $\|\eta_{n+1}\| \leq \|\eta_n\|$ when $\left(\Lambda_\alpha - \Phi^T\Phi\right)$ is not sufficiently close to zero. However, we find in practice that eq(18) is usually adequate for convergence, because $|\eta_{j,n+1}| \leq |\eta_{j,n}|$ holds for most j, which dominates in the relationship of $\|\eta_{n+1}\|$ to $\|\eta_n\|$.

3.3. $\mathrm{L_0RL_2}$ continuation

The continuation strategy is inspired by homotopy techniques in L1-sparse representations [14]. Malioutov et al [15] considered the solution path for

$$\arg\min_{\mathbf{x}} \lambda\|\mathbf{x}\|_1 + \|\Phi\mathbf{x} - y\|_2^2. \tag{19}$$

This algorithm terminates when it produces a desired number of non-zero components in the reconstructed $\hat{\mathbf{x}}$. Hale et al [14] introduce a fixed-point algorithm which approximately follows the solution path of eq(19) for values of $0 < \lambda < \|\Phi\mathbf{y}\|_\infty$, and hence the sparse solution may be found by solving a sequence of L1-norm penalized problems. It is also reported that it is faster to solve the above L1-norm penalized least square problem when λ is large. This observation greatly motivates the exploration of the convergence effect of different ϵ (because $\nu^2 = 8\min(\boldsymbol{\alpha})\epsilon^2$, it is equivalently the effect of ν.). In a series of experiments, we find larger ϵ gives faster convergence, but small ϵ results in better recovery. By reducing ϵ from a large value, we hope to accelerate the convergence speed. In fact, the idea of using continuation to speed up the convergence has been shown to be very successful when dealing with large-scale problems[16].

Our strategy for continuation is simple. Let $r_L(\mathbf{x})$ denote the Lth largest amplitude element of vector $\mathbf{x}$ and $L_{\max}$, the maximum number of nonzeros. Set initial $\epsilon = \|\Phi^T\mathbf{y}\|_\infty$, and then reduce ϵ gradually. We summarise the key steps of the $\mathrm{L_0RL_2}$ continuation algorithm as follows:

1. estimate $\mathbf{x}_{n+1}$ using eq(9a)
2. update ϵ and ν:
   $L = \min(L+1, L_{\max})$
   $\epsilon = \min(2r_L(\mathbf{x}_{n+1}), \epsilon)$; $\quad \nu^2 = 8\epsilon^2\min(\boldsymbol{\alpha})$
3. update $\mathbf{S}$ according to eq(9b)

However, we know of no formal proof of global convergence for nonconvex problem.

4. COMPRESSIVE SENSING RESULTS

We have chosen two experiments to demonstrate the performance of the $\mathrm{L_0RL_2}$ algorithm for compressed sensing scenarios.

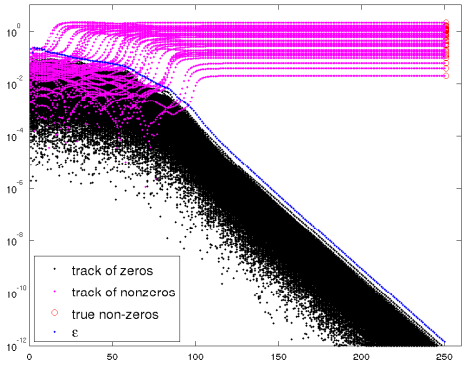
4.1. 1-D random signal

We have tested the algorithm on a similar example to Daubechies et al in [7]. The sparse input signal has 1500 elements, among which there are 45 non-zeros. The position of the nonzeros are picked randomly and the amplitudes are generated from a normal distribution with mean 0 and standard deviation (s.dev) 1. The sampling matrix Φ is of size 250×1500, with Gaussian $\mathcal{N}(0,1)$ i.i.d. entries. We also added Gaussian noise with s.dev 0.05 to the observations.

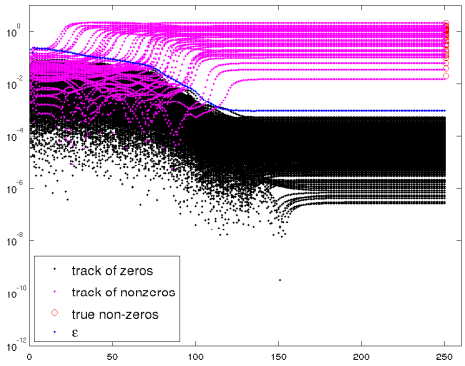
The evolution of $\mathbf{x}$ versus iteration number is shown in Figure 2. We use purple to trace the elements of $\mathbf{x}$ that should be non-zero, and black to trace the zeros. The blue line depicts ϵ. The true values of the non-zero elements are shown by red circles at the ends of the purple tracks. It shows that $\mathrm{L_0RL_2}$ gradually picks the non-zero elements of $\mathbf{x}$ as ϵ decreases, and all the non-zeros are well separated from the zeros, after ϵ drops below the smallest non-zero. In these experiments, we found the algorithm is not sensitive to $L_{\max}$, the nominal sparsity value. The algorithm still gets perfect recovery when $L_{\max}$ is set to several times the nominal value of 45. The results in Figure 2 are obtained with $L_{\max} = 80$. In the example with noisy observations, ν converges to 0.0576, whereas the true s.dev of the added noise is 0.05.

Since the problem is small, IRL1 also perfectly recover the signal in 1.230 seconds while $\mathrm{L_0RL_2}$ only takes 0.165 seconds. IRLS [7] takes 32.50 seconds to reach the same level of accuracy.

[2]This can often be achieved in compressed sensing applications where the sampling matrix is commonly a submatrix of a unitary transformation to allow a fast algorithm for matrix-vector products, e.g. a submatrix of the DCT or Fourier transform. Such matrices give an approximately diagonal $\Phi^T\Phi$.



(a) Noise Free



(b) Noise s.dev = 0.05, SNR $\approx$ 42dB

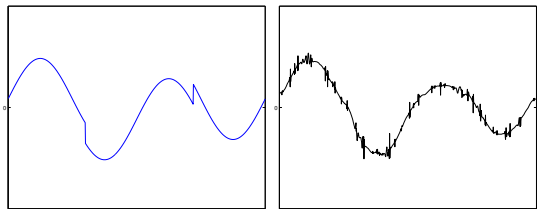
Fig. 2. Plots showing how $\mathrm{L_0RL_2}$ gradually selects the non-zero components of $\mathbf{x}$ as ϵ is reduced. Horizontal axes are iteration numbers and vertical axes are amplitudes of $\mathbf{x}$. It should be noted that the limiting value ϵ is about 10^{-3}, $\nu = \sqrt{8\alpha}\epsilon$, where $\sqrt{8\alpha} \approx 59$, where α is proportional to the largest eigenvalue of $\Phi^T\Phi$.

4.2. 1-D Heavisine signal

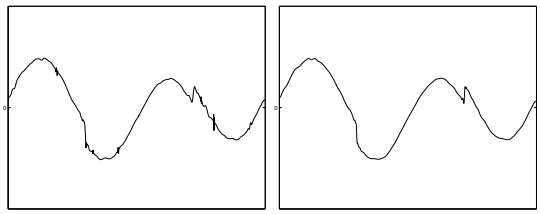
The original Heavisine signal is shown in Figure 3(a). It is a piecewise-smooth signal with length $N = 1024$, and we have chosen only $M = 80$ random Gaussian measurements of it to be available for its recovery, as in [8]. For comparison, the L1 and iterative reweighted L1 (IRL1) minimizations are performed by the Sparco [17] and SPGL1 [18] toolboxes.

We chose the dual-tree complex wavelet transform (DT $\mathbb{C}$WT) [19] as the sparse basis for implementing this experiment, because it has good shift invariant and sparsity-inducing properties. We also observed improved results for L1 and IRL1 minimization with DT $\mathbb{C}$WT. The initial guess to start $\mathrm{L_0RL_2}$ and IRL1 is the zero vector with unit weights. Results of IRL1 and $\mathrm{L_0RL_2}$ are shown in Figure 3. The recovery quality is quantitatively measured by $\mathrm{RMSE} = \|\mathbf{x} - \mathbf{s}\|_2/\|\mathbf{s}\|_2$, where $\mathbf{x}$ is the final estimation and $\mathbf{s}$ is the true input signal. α is set in the same way as in example 4.1. Specifically, $\alpha = 3.8$.

The sparse representation of natural signals on a wavelet basis is often well structured and utilizing such information



(a) original signal (b) L1-optimization



(c) IRL1, $\mathrm{RMSE} = 0.042$ (d) $\mathrm{L_0RL_2}$, $\mathrm{RMSE} = 0.026$

Fig. 3. Example performance on the Heavisine signal with 80 measurements. To obtain the above results, $\mathrm{L_0RL_2}$ updates weights for 50 times (updates every 2 iterations) and its computation time is about 14.5s, while IRL1 updates weights for 4 times and its computation time is 894.8s.

in signal reconstruction often results in improved performance in the sense that fewer samples are needed for perfect recovery as reported in [8]. Because the signal is represented on the DT $\mathbb{C}$WT frame, which will benefits from a strong parent-child model, we integrates a parent-child bivariate prior into the algorithm of eq(9), as described in [20]. For a wavelet coefficient x_j, we denote its parent as $x_{p(j)}$. In the implementation, we assume the latest estimates, $|x_j|$ and $|x_{p(j)}|$, are the noisy observations of σ_j and $\sigma_{p(j)}$. This gives a simple inter-scale denoising scheme, bivariate shrinkage for estimating the σ_j at each scale, from coarse to fine in turn. Implementation details are available in [20]. We denote this bivariate-shrinkage aided $\mathrm{L_0RL_2}$ as Bi-$\mathrm{L_0RL_2}$ in Table 1.

To demonstrate the effect on recovery quality, we ran Bi-$\mathrm{L_0RL_2}$, $\mathrm{L_0RL_2}$, IRL1 and L1 minimization on 20 random implementations of Φ. We also reduced the number of observation from 80 to 60 and 50. The recovery quality is quantified by RMSE in Table 1. To obtained the results in Table 1, IRL1 needs to update weights for 4 times, the average computation time is around 840 seconds for each implementation. $\mathrm{L_0RL_2}$ runs 100 iterations and updates weights every two iteration, and each implementation takes about 14 seconds; Bi-$\mathrm{L_0RL_2}$ also runs 100 iteration and each implementation takes around 23 seconds.

5. CONCLUSION AND DISCUSSION

We propose a reweighted L2 norm algorithm within a continuation framework for fast sparse signal recovery. It relies only on one preset parameter, the sparsity level $L_{\max}$, and

Table 1. Recovery error of different algorithms, $\frac{\|\mathbf{x}-\mathbf{s}\|_2}{\|\mathbf{s}\|_2}$.

Obs. No.	80		60		50	
	mean	std	mean	std	mean	std
L1	0.194	0.049	0.291	0.102	0.464	0.167
IRL1	0.042	0.007	0.061	0.010	0.178	0.157
$\mathrm{L_0RL_2}$	0.037	0.009	0.058	0.011	0.082	0.047
Bi-$\mathrm{L_0RL_2}$	0.037	0.006	0.053	0.010	0.073	0.013

a loose estimate of $L_{\max}$ is enough to achieve good results. We observe two learning behaviours of the algorithm: 1) the sparsity map is automatically recovered without any prior knowledge; 2) the noise level is automatically estimated. We observe that no prior information about the noise variance is needed and ν automatically converges to the true noise level.

The algorithm is suitable for large-scale problems, because it only requires matrix-vector multiplications and element-wise operations in each iteration. It also allows the integration of prior knowledge of signal structure, which reduces the number of observation needed for perfect reconstruction. In our experiments, a bivariate parent-child prior is used in the wavelet domain, and improved stability of recovery error is observed (see Table 1).

On the other hand, the basic model we discussed in Section 2.1 uses parameters σ_j to controls the sparsity of $\mathbf{x}$, for small σ_j will drive the weights S_j so large that small x_j will be actually pruned out. This model finds itself very similar with the hierarchical (sparse) Bayesian modelling. An important tool for solving similar hierarchical Bayesian modelling is the relevance vector machine [21]. It is possible to solve our problem with the relevance vector machine. Future work on this aspect is warranted.

6. REFERENCES

[1] Deanna Needell, *Topics in Compressed Sensing*, Ph.D. thesis, Mathematics, Univ. of California, Davis, 2007.

[2] S.G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. signal processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[3] E.J. Candès, M.B. Wakin, and S.P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, 2008.

[4] I.F. Gorodnitsky and B.D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm," *IEEE Trans. Signal Processing*, vol. 45, no. 3, pp. 600–616, 1997.

[5] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *ICASSP 2008. IEEE International Conference on*, 2008, pp. 3869–3872.

[6] E.J. Candès and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Trans. Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[7] I. Daubechies, R. DeVore, M. Fornasier, and S. Gunturk, "Iteratively re-weighted least squares minimization for sparse recovery," *Communications on Pure and Applied Mathematics*, 2009.

[8] R.G. Baraniuk, V. Cevher, M.F. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Trans. Information Theory*, vol. 56, no. 4, pp. 1982 –2001, april 2010.

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

[10] Cédric Vonesch and Michael Unser, "A fast thresholded landweber algorithm for wavelet-regularized multidimensional deconvolution," *IEEE Trans. Image Processing*, vol. 17, no. 4, pp. 539–549, 2008.

[11] I. Bayram and I.W. Selesnick, "A Subband Adaptive Iterative Shrinkage/Thresholding Algorithm," *Accepted for publication, IEEE Trans. Signal Processing*.

[12] Y. Zhang and N. Kingsbury, "A Bayesian wavelet-based multidimensional deconvolution with sub-band emphasis," in *EMBS 2008. IEEE International Conference on*, 2008, pp. 3024–3027.

[13] P.L. Combettes and V.R. Wajs, "Signal recovery by proximal forward-backward splitting," *SIAM Journal on Multiscale Modeling and Simulation*, vol. 4, no. 4, pp. 1164–1200, 2005.

[14] E. Hale, W. Yin, and Y. Zhang, "A fixed-point continuation method for l1-regularization with application to compressed sensing," Tech. Rep., 2007.

[15] D.M. Malioutov, M. Cetin, and A.S. Willsky, "Homotopy continuation for sparse signal representation," in *ICASSP 2005. IEEE International Conference on*. Citeseer, 2005.

[16] S. Becker, J. Bobin, and E.J. Candès, "NESTA: A fast and accurate first-order method for sparse recovery," *subbmitted for publication.*

[17] E. van den Berg, M. P. Friedlander, G. Hennenfent, F. Herrmann, R. Saab, and Ö. Yılmaz, "Sparco: A testing framework for sparse reconstruction," Tech. Rep. TR-2007-20, Dept. Computer Science, University of British Columbia, Vancouver, October 2007.

[18] E. van den Berg and M. P. Friedlander, "SPGL1: A solver for large-scale sparse reconstruction," June 2007, http://www.cs.ubc.ca/labs/scl/spgl1.

[19] I.W. Selesnick, R.G. Baraniuk, and N.G. Kingsbury, "The dual-tree complex wavelet transform," *Signal Processing Magazine, IEEE*, vol. 22, no. 6, pp. 123–151, 2005.

[20] L. Sendur and I.W. Selesnick, "Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency," *IEEE Trans. Signal Processing*, vol. 50, no. 11, pp. 2744–2756, 2002.

[21] C.M. Bishop and M.E. Tipping, "Variational relevance vector machines," in *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, pp. 46–53, 2000.