

New Algorithms for Sparse Representation of Discrete Signals Based on ℓ_p - ℓ_2 Optimization

Jie Yan and Wu-Sheng Lu

Department of Electrical and Computer Engineering

University of Victoria

Victoria, BC, Canada V8W 3P6

jyan@ece.uvic.ca, wslu@ece.uvic.ca

Abstract

This paper investigates a nonconvex relaxation of the popular ℓ_1 - ℓ_2 formulation for finding sparse representation of discrete signals in overcomplete dictionary. The specific nonconvex problem we propose to solve is an ℓ_p - ℓ_2 problem with $0 < p < 1$. Our algorithms are built on a recent algorithm, known as the monotone fast iterative shrinkage/thresholding algorithm, where a key step of soft shrinkage is replaced by a global solver for the minimization of a 1-D nonconvex ℓ_p problem. Two efficient techniques for solving the 1-D ℓ_p problem in question are proposed. Simulation studies are presented to evaluate the performance of the proposed algorithms with various values of p and compare with the well known basis pursuit (BP) algorithm with $p = 1$.

1. Introduction

Over the last two decades, modeling signals exploring sparsity has emerged as an effective technique in signal processing. A central point of sparse signal processing is to seek an approximate solution to an ill-posed or underdetermined linear system while requiring that the solution has fewest nonzero entries. This problem arises in various areas across engineering and science [1, 2]. Many of these applications lead to the minimization of a mixed ℓ_1 and ℓ_2 expressions in the form

$$F(\mathbf{s}) = \|\mathbf{x} - \Psi\mathbf{s}\|_2^2 + \lambda\|\mathbf{s}\|_1.$$

An attractive feature of this formulation is that function $F(\mathbf{s})$ is globally convex and its global minimizer can be identified easily using a convex program solver.

The problem of the ℓ_1 - ℓ_2 sparse approximation was traditionally treated using various classical iterative optimization algorithms, homotopy solvers and greedy techniques

like matching pursuit and orthogonal matching pursuit [3]. However, these algorithms are often inefficient to reach the solution. Furthermore, the homotopy and greedy techniques are impractical in high-dimensional problems, as often encountered in image processing applications [4]. Over the past several years, iterative-shrinkage algorithms have emerged as a family of highly effective numerical methods for above ℓ_1 - ℓ_2 optimization problems. Of particular interest is a state-of-the-art algorithm called the fast iterative shrinkage-thresholding algorithm (FISTA) developed in [5]. The FISTA is shown to provide a convergence rate of $O(1/k^2)$ compared to the rate of $O(1/k)$ (with k denoting the number of iterations used) by the well-known proximal-point algorithm known as the iterative shrinkage-thresholding algorithm (ISTA), while maintaining practically the same complexity as the ISTA. In [6], with slight increase in complexity an enhanced version of FISTA, known as MFISTA, was proposed that possesses a desired property of monotone convergence. In [7] and [8], algorithms based on ℓ_p minimization with $0 < p < 1$ were proposed for sparse representation and improved reconstruction results relative to those obtained by ℓ_1 minimization were demonstrated.

In this paper, new algorithms for sparse representation based on ℓ_p - ℓ_2 optimization are proposed. Our algorithms are built on MFISTA with several major changes. In particular, because of our methodological shift from ℓ_1 - ℓ_2 formulation to ℓ_p - ℓ_2 formulation, the problem at hand is no longer convex, and the soft shrinkage—a key step in MFISTA, is replaced by a global solver for the minimization of a 1-D nonconvex ℓ_p problem. To this end, two efficient techniques for solving the 1-D ℓ_p problem in question are proposed that become the technical cornerstone in the new algorithms. The computational complexity of the proposed algorithms is analyzed and simulation studies are presented to evaluate the performance of the proposed algorithms with various values of p and compare with the well known basis pursuit (BP) al-

gorithm with $p = 1$.

2. Preliminaries

2.1. Sparse representations in overcomplete bases

A typical sparse representation problem can be stated as finding the sparsest representation of a discrete signal \mathbf{x} under a (possibly overcomplete) dictionary Ψ . The sparsity of a vector \mathbf{s} is often expressed as the ℓ_0 norm of \mathbf{s} defined by $\|\mathbf{s}\|_0 =$ the number of nonzero entries in \mathbf{s} , although strictly speaking ℓ_0 norm is not a vector norm. With this notation, the above problem can be described as minimizing $\|\mathbf{s}\|_0$ subject to $\mathbf{x} = \Psi\mathbf{s}$. Unfortunately, this problem is known to be NP hard. A relaxed version of this problem permits a small perturbation in the representation, leading to the problem

$$\min_{\mathbf{s}} \|\mathbf{s}\|_0 \text{ subject to } \|\mathbf{x} - \Psi\mathbf{s}\|_2 \leq \epsilon. \quad (1)$$

This problem is also known to be NP hard, and one is motivated to develop efficient suboptimal approximation algorithms. An appealing solution method is the basis-pursuit (BP) [1] which solves a modified problem of (1) with the ℓ_0 norm replaced by a convex ℓ_1 norm. The problem so modified is a quadratic convex problem, known as second order cone programming (SOCP), which admits a unique global solution. In principle, the BP problem can be solved using a standard solver for convex problems. Recent studies exploring the specific structure of the problem have led to more efficient algorithms [3, 9]. Among these, a popular approach is to convert the constrained minimization encountered in BP type of problems into an ℓ_1 - ℓ_2 unconstrained convex problem as

$$\min_{\mathbf{s}} F(\mathbf{s}) = \|\mathbf{x} - \Psi\mathbf{s}\|_2^2 + \lambda\|\mathbf{s}\|_1 \quad (2)$$

where $\lambda > 0$ is a regularization parameter that controls the tradeoff between the sparsity of \mathbf{s} and the approximation error $\|\mathbf{x} - \Psi\mathbf{s}\|_2$.

2.2. FISTA and MFISTA

We begin with reviewing an algorithm, known as the iterative shrinkage-thresholding algorithm (ISTA), which also bears the names of ‘‘proximal-point method’’ and ‘‘separable surrogate functionals method’’ [4]. A key step in its k th iteration is to approximate the objective function in (2) by an easy-to-deal-with upper-bound (up to a constant) convex function given by

$$\hat{F}(\mathbf{s}) = \frac{L}{2}\|\mathbf{s} - \mathbf{c}_k\|_2^2 + \lambda\|\mathbf{s}\|_1 \quad (3)$$

where $\mathbf{c}_k = \mathbf{s}_{k-1} - \frac{1}{L}\nabla f(\mathbf{s}_{k-1})$, $f(\mathbf{s}) = \|\mathbf{x} - \Psi\mathbf{s}\|_2^2$, and L is the smallest Lipschitz constant of $\nabla f(\mathbf{s})$, which is found to be $L = 2\lambda_{\max}(\Psi\Psi^T)$. The k th iteration of the ISTA finds the next iterate by minimizing $\hat{F}(\mathbf{s})$. Because both terms in $\hat{F}(\mathbf{s})$ are coordinate-separable, it can be readily verified that the minimizer of $\hat{F}(\mathbf{s})$ can be calculated by a simple shrinkage of vector \mathbf{c}_k with a constant threshold λ/L as $\mathbf{s}_k = \mathcal{T}_{\lambda/L}(\mathbf{c}_k)$ where operator \mathcal{T} applies to a vector pointwisely with $\mathcal{T}_a(c) = \text{sign}(c)\max\{|c| - a, 0\}$. Once iterate \mathbf{s}_k is obtained, it is used to update vector from \mathbf{c}_k to \mathbf{c}_{k+1} and the shrinkage operator \mathcal{T} is then applied to it to get the next iterate, and so on.

Evidently, the iterations in ISTA are of very low complexity. However, the algorithm only provides a slow convergence rate of $O(1/k)$. In [5] and [6], an algorithm called the fast iterative shrinkage-thresholding algorithm (FISTA) was proposed and shown to provide a much improved convergence rate of $O(1/k^2)$, where the complexity of each iteration is practically the same as that of ISTA. The FISTA is built on ISTA with an extra step in each iteration that, with the help of a sequence of scaling factors t_k , creates an auxiliary iterate \mathbf{b}_{k+1} by moving the current iterate \mathbf{s}_k along the direction of $\mathbf{s}_k - \mathbf{s}_{k-1}$ so as to improve the subsequent iterate \mathbf{s}_{k+1} . The steps in the k th iteration of FISTA are outlined as follows with initial $\mathbf{b}_1 = \mathbf{s}_0$ and $t_1 = 1$.

1. Perform shrinkage

$$\mathbf{s}_k = \mathcal{T}_{\lambda/L}\left\{\frac{1}{L}\Psi^T(\mathbf{x} - \Psi\mathbf{b}_k) + \mathbf{b}_k\right\};$$

2. Compute $t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}$;

3. Update $\mathbf{b}_{k+1} = \mathbf{s}_k + \left(\frac{t_k - 1}{t_{k+1}}\right)(\mathbf{s}_k - \mathbf{s}_{k-1})$.

Furthermore, by including an additional step to FISTA, the algorithm is enhanced to possess desirable monotone convergence [6]. The algorithm is called the monotone FISTA or MFISTA.

Another interesting development in sparse representation and compressed sensing is to investigate a *nonconvex* variant of the basis pursuit by replacing the ℓ_1 norm term in BP with an ℓ_p norm with $0 < p < 1$ [7, 8]. (We remark that with $p < 1$, the ‘‘ ℓ_p norm’’ is no longer a norm, however $\|\mathbf{x} - \mathbf{y}\|_p$ remains a meaningful distance measure). It was demonstrated by numerical experiments [7] that fewer measurements than that of BP are required for exact reconstruction of a sparse signal. Naturally, an ℓ_p - ℓ_2 counterpart of (2) can be formulated as

$$\min_{\mathbf{s}} F(\mathbf{s}) = \|\mathbf{x} - \Psi\mathbf{s}\|_2^2 + \lambda\|\mathbf{s}\|_p^p \quad (4)$$

where $\mathbf{x} \in R^N$, $\Psi \in R^{N \times M}$ and $\mathbf{s} \in R^M$. For an overcomplete basis Ψ , $M > N$.

3. Algorithms for ℓ_p - ℓ_2 optimization in sparse representation

The algorithms we propose in this paper will be developed within the framework of FISTA/MFISTA in that

$$\mathbf{s}_k = \underset{\mathbf{s}}{\operatorname{argmin}} \left\{ \frac{L}{2} \|\mathbf{s} - \mathbf{c}_k\|_2^2 + \lambda \|\mathbf{s}\|_p^p \right\} \quad (5)$$

where $0 < p < 1$. At a glance, the objective function in (5) differs from (3) only slightly with its ℓ_1 term replaced by an ℓ_p term. However, this change turns out to be a rather major one in two aspects: First, with $p < 1$ (5) provides a problem setting closer to the ℓ_0 -norm problem, hence an improved sparse representation is expected, and this is indeed the very reason of the studies reported in this paper. Second, with $p < 1$ the problem in (5) becomes nonconvex. As a result, efficient technique like soft shrinkage fails to work in general. In what follows, we present two techniques that can be used to find the global solution of (5) with $p < 1$.

3.1. Method 1: When p is Rational

Assume $\mathbf{s} = [s_1 \ s_2 \ \cdots \ s_M]$ and $\mathbf{c}_k = [c_{k1} \ c_{k2} \ \cdots \ c_{kM}]$, Eq. (5) can be expressed as

$$\mathbf{s}_k = \underset{\mathbf{s}}{\operatorname{argmin}} \sum_{i=1}^M \left[\frac{L}{2} (s_i - c_{ki})^2 + \lambda |s_i|^p \right].$$

It is not hard to see that the computation of \mathbf{s}_k reduces to solving M one-dimensional (1-D) minimization problems, and it boils down to solving the 1-D problem

$$s^* = \underset{s}{\operatorname{argmin}} \{u(s) = \frac{L}{2}(s - c)^2 + \lambda |s|^p\}. \quad (6)$$

Suppose p is a rational number, namely $p = a/b$ with a, b positive integers and $a < b$. Let us first consider $s \geq 0$ so that the absolute value sign can be removed, then

$$u(s) = \frac{L}{2}(s - c)^2 + \lambda s^{a/b}.$$

To get rid of the rational power, let $z = s^{1/b}$. In this way,

$$v(z) = u(s)_{s=z^b} = \frac{L}{2}(z^b - c)^2 + \lambda z^a$$

and the problem of minimizing function $u(s)$ ($s \geq 0$) is converted to minimizing function $v(z)$ ($z \geq 0$), which is a polynomial and can be written in descending power as

$$v(z) = \frac{L}{2}z^{2b} - Lcz^b + \lambda z^a + \frac{L}{2}c^2.$$

Since $v(z)$ is differentiable, the global minimizer z_+^* must either be 0, or one of those stationary points where $\nabla v(z) =$

0. Consequently, the problem is reduced to solve the equation

$$\nabla v(z) = Lbz^{2b-1} - Lcbz^{b-1} + \lambda az^{a-1} = 0. \quad (7)$$

The solutions of this polynomial equation are the eigenvalues of the companion matrix of size $(2b - 1) \times (2b - 1)$. In our simulations, MATLAB function `roots` was applied to find all the roots of polynomial $\nabla v(z)$. Among the set of positive roots and the boundary point 0, the global minimizer z_+^* is identified as the one that yields the smallest $v(z)$, and $s_+^* = (z_+^*)^b$ is the solution that minimizes $u(s)$ for $s \geq 0$.

In a similar way, the global minimizer s_-^* that minimizes $u(s)$ for $s \leq 0$ can be computed, and the global minimizer s^* is obtained as

$$s^* = \underset{s}{\operatorname{argmin}} \{u(s) : s = s_+^*, s_-^*\}.$$

It follows that the global solution \mathbf{s}_k for problem (5) can be found by repeating the procedures described above M times.

If we incorporate the above ℓ_p solver into an FISTA/MFISTA type algorithm, then the computation required in each iteration of the algorithm is dominated by the ℓ_p solver that computes the eigenvalues of a matrix of size $2b-1$ by $2b-1$, hence a complexity of $O(M(2b-1)^3)$. Clearly, the value of b greatly affects the computational cost. In a practical implementation of the algorithm, let p be a given (desired) value in $(0, 1)$ and b' be an even integer so that $1/b'$ is considered acceptable tolerance for a rational approximation of p . The interval $(0, 1)$ is partitioned uniformly into subintervals of length $1/b'$. Now suppose value p falls into one of the subintervals, say $I_k = [k/b', (k+1)/b']$ for some integer k . Obviously, either end point of I_k serves as a rational approximation within tolerance $1/b'$. One then chooses the end point with an even-integer numerator. In this way, the rational approximation of p (within tolerance $1/b'$) assumes the form of a/b with $b = b'/2$ so as to yield a reduced complexity $O(M(b'-1)^3)$ for the ℓ_p solver.

In summary, the method proposed above works well whenever p is rational with a small denominator integer such as $p \in \{1/4, 1/3, 1/2, 2/3, 3/4\}$, or p is an irrational number that can be approximated by one of the above rational numbers within an acceptable tolerance. For a power p that does not fall within these cases, the global solution of (6) can be obtained using the method described below.

3.2. Method 2: When p is an Arbitrary Real in $(0, 1)$

The method proposed above produces precise solution for rational p and works efficiently when b is small. However, we remark that it only generates approximate solutions

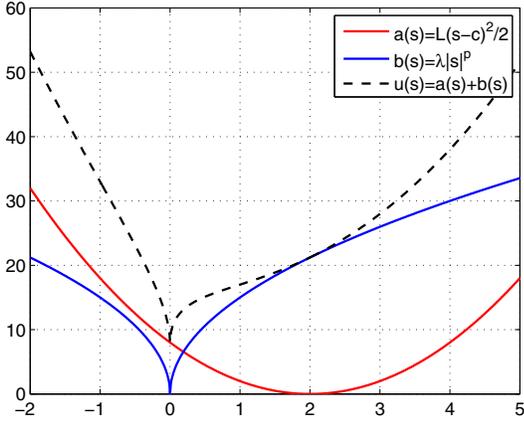


Figure 1: Function $u(s)$ with $c > 0$.

for irrational values of power p , and its complexity becomes considerable especially if a large denominator integer b is involved. The method presented below is designed for an arbitrary p in $(0, 1)$ with low complexity.

To begin with, we examine function $u(s)$ in (6) with respect to parameter c . If $c = 0$, it is obvious that $s^* = 0$. Next, we consider the case of $c > 0$. To illustrate the current circumstance, Fig. 1 plots $a(s) = \frac{L}{2}(s-c)^2$, $b(s) = \lambda|s|^p$ and $u(s) = a(s) + b(s)$ for some L, c, λ and p . It can be observed that when variable s is in the region $(-\infty, 0)$, $a(s)$ and $b(s)$ are both monotonically decreasing; in addition when $s \in (c, +\infty)$, $a(s)$ and $b(s)$ are both monotonically increasing. Hence the global minimizer s^* lies in $[0, c]$ where the function of interest becomes

$$u(s) = \frac{L}{2}(s-c)^2 + \lambda s^p \quad \text{for } s \in [0, c]. \quad (8)$$

As mentioned earlier, gradient information is not sufficient to identify the global minimizer. The convexity property of $u(s)$ can be analyzed by examining the 2nd-order derivative of (8), i.e.,

$$u''(s) = L + \lambda p(p-1)s^{p-2}. \quad (9)$$

By solving the equation $u''(s) = 0$, we obtain $s_c = \left[\frac{\lambda p(1-p)}{L}\right]^{1/(2-p)}$. Clearly, $s_c > 0$. For $0 \leq s < s_c$, $u(s)$ is concave as $u''(s) < 0$; for $s > s_c$, $u(s)$ is convex as $u''(s) > 0$. For s in interval $[0, c]$, two cases need to be examined.

1. If $s_c \geq c$, $u(s)$ is concave in $[0, c]$. As a result, s^* must be either 0 or c . Namely, $s^* = \underset{s}{\operatorname{argmin}} \{u(s) : s \in [0, c]\}$. This case is illustrated in Fig. 2.
2. If $s_c < c$, as illustrated in Fig. 3, since $u(s)$ is concave in $[0, s_c]$ and convex in $[s_c, c]$, we argue that s^* must

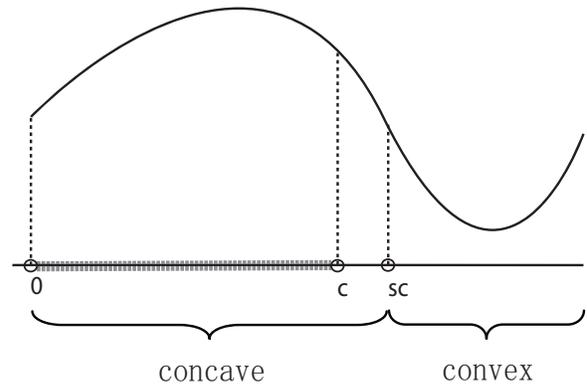


Figure 2: Function $u(s)$ when $s_c \geq c$ (with $c > 0$).

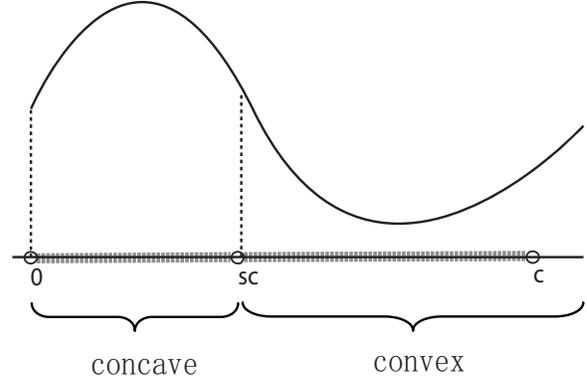


Figure 3: Function $u(s)$ when $s_c < c$ (with $c > 0$).

be either at the point s_t that minimizes convex function $u(s)$ in $[s_c, c]$, or at the boundary point 0. Hence the global solution is obtained as $s^* = \underset{s}{\operatorname{argmin}} \{u(s) : s \in [0, s_t]\}$.

To minimize convex function $u(s)$ in $[s_c, c]$, three situations may occur: (a) If $\nabla u(s_c) \geq 0$ and $\nabla u(c) > 0$, then $s_t = s_c$; (b) If $\nabla u(s_c) < 0$ and $\nabla u(c) \leq 0$, then $s_t = c$; (c) If $\nabla u(s_c) < 0$ and $\nabla u(c) > 0$, then a quadratic interpolation method can be applied as an approximation approach to find point s_t , see Sec. 4.5 of [10] for the details. Typically, it takes only a small number of iterations for this quadratic-approximation based algorithm to converge to minimizer s_t over a small interval $[s_c, c]$. A similar analysis can be carried out for the case of $c < 0$.

In summary, we have proposed two techniques for the global minimization of the 1-D subproblem in (6) for an arbitrary power p between 0 and 1. Based on this, an MFISTA type algorithm for nonconvex problem (4) can be developed by replacing the shrinkage step of the conventional MFISTA [6] with the above 1-D ℓ_p solver. In what follows, the algorithm proposed here will be referred to as the *modified MFISTA*.

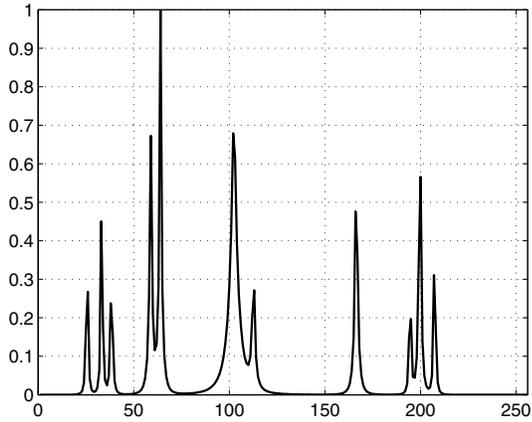


Figure 4: Bumps signal of length $N = 256$.

4. Simulations

A test signal \mathbf{x} of length $N = 256$, known as “bump-
s” [3], was used to examine the effectiveness of the
proposed algorithm, see Fig. 4. The dictionary adopted
here is a combination of three orthonormal bases
 $\Psi = [\Psi_1 \ \Psi_2 \ \Psi_3] \in R^{N \times 3N}$ where Ψ_1 is the Dirac basis,
 Ψ_2 is the 1-D DCT basis and Ψ_3 is the wavelet basis gener-
ated by orthogonal Daubechies wavelet D8. Our objective
is to find a representation vector $\mathbf{s} \in R^{3N \times 1}$ for signal \mathbf{x}
such that $\mathbf{x} \approx \Psi \mathbf{s}$ with \mathbf{s} as sparse as possible. To this
end we solve problem (4) with $p = 1, 0.95, 0.9, 0.85, 0.8$
and 0.75 , respectively. Our simulations are based on the
MFISTA [6] as the algorithm ensures monotonic conver-
gence. This algorithm was implemented with s_k calculated
from Eq. (5) either by Method 1 or Method 2 presented
in Sec. 3. For each ℓ_p - ℓ_2 problem with a particular p , the
experiment was carried out by the steps outlined below.

Step 1. Set $\mathbf{s}_0 = \mathbf{0}$ and $i = 1$. Generate a vector
 $\lambda = [\lambda_1 \ \lambda_2 \ \dots \ \lambda_T]$ with $\lambda_1 > \lambda_2 > \dots > \lambda_T$. The
number of iterations of the modified MFISTA was set to be
 $K = 200$.

Step 2. Apply the modified MFISTA to solve problem (4)
with initial point \mathbf{s}_{i-1} and parameter $\lambda = \lambda_i$ to obtain the
solution $\hat{\mathbf{s}}$. Set $\mathbf{s}_i = \hat{\mathbf{s}}$.

Step 3. Compute relative equation error

$$R_i = \frac{\|\Psi \mathbf{s}_i - \mathbf{x}\|_2}{\|\mathbf{x}\|_2}.$$

Compute the percentage of zeros in \mathbf{s}_i and denote it by Z_i (a
component of \mathbf{s}_i was regarded as zero if its absolute value
falls below $1e-5$).

Step 4. If $i = T$, stop; otherwise set $i = i + 1$ and repeat
from Step 2. ■

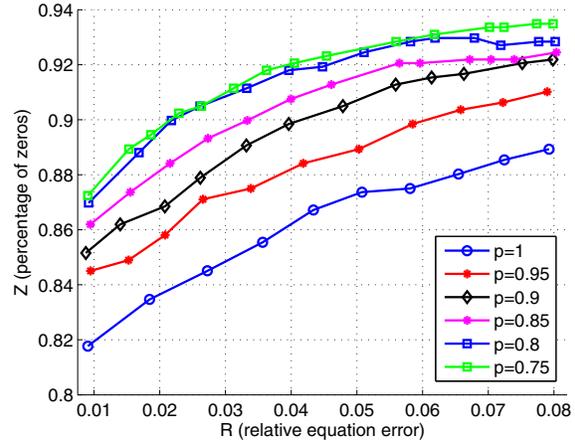


Figure 5: Comparison of ℓ_p - ℓ_2 sparse representation of
“bumps” signal for $p = 1, 0.95, 0.9, 0.85, 0.8, 0.75$ in terms
of relative equation error and signal sparsity in the dictio-
nary domain.

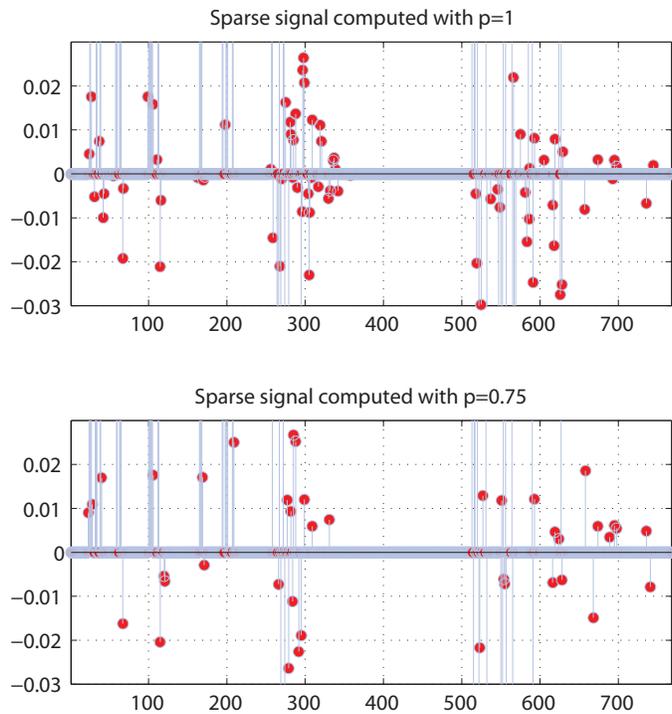


Figure 6: Sparse representation of the “bumps” signal based
on ℓ_1 and $\ell_{0.75}$ reconstruction.

It should be stressed that the parameter vector λ consists
of decreasing components $\lambda_1 > \lambda_2 > \dots > \lambda_T$. Take $p =$
 1 for instance, these components were set to be an arith-
metic progression from $\lambda_1 = 5e - 2$ to $\lambda_T = 5e - 3$ with

a common difference of $5e-3$. The components of λ were tuned for each individual value of p so that the same level of relative equation error is attained. As a result, for a given value of power p , two sequences $\mathbf{R} = [R_1 R_2 \cdots R_T]$ and $\mathbf{Z} = [Z_1 Z_2 \cdots Z_T]$ were produced.

The quality of a sparse representation can be evaluated by two criteria: (a) how sparse the coefficient vector $\hat{\mathbf{s}}$ is in the dictionary domain; and (b) how well $\hat{\mathbf{x}} = \Psi\hat{\mathbf{s}}$ resembles \mathbf{x} . In the experiment, sparsity is measured by computing the percentage of zeros in $\hat{\mathbf{s}}$ (as seen in vector \mathbf{R}). And the signal reconstruction precision is measured by the relative equation error $\|\hat{\mathbf{x}} - \mathbf{x}\|/\|\mathbf{x}\|$ (as seen in vector \mathbf{Z}). Since the value of regularization parameter λ controls the tradeoff between sparsity and equation error of the solution, a curve generated with the components of \mathbf{R} as its x -coordinates and the components of \mathbf{Z} as its y -coordinates provides a performance profile of the solution that shows how the sparsity/equation error evolves as parameter λ varies. A total of six such curves for $p = 1, 0.95, 0.9, 0.85, 0.8$ and 0.75 for signal “bumps” are depicted in Fig. 5.

From Fig. 5, several observations can be made. (1) For a fixed relative equation error, the sparsity improves as a smaller power p was used, and this justifies the usefulness of the proposed ℓ_p pursuit algorithm; (2) For a fixed level of sparsity, we see that the relative equation error decreases as a smaller power p was used. This is just a different perspective to justify the ℓ_p pursuit algorithm; (3) The performance improvement appears to be kind of nonlinear with respect to the change in power p . Starting from $p = 1$ (the BP pursuit), a 0.05 decrease in p leads to a significant performance improvement. As p continues to decrease, the performance continues to gain but the incremental gain becomes gradually less significant.

For further illustration, Fig 6 depicts the signals obtained by solving problem (4) with $p = 1$ and $p = 0.75$, respectively. For a fair comparison, the values of parameter λ were chosen such that both solutions yield the same relative equation error of 0.00905. Note that these two instances correspond to the two leftmost points on the two curves in Fig. 5 that are associated with the above two p values. The sparsity achieved was found to be 87.24% for $p = 0.75$ versus 81.77% for $p = 1$. The improvement in sparsity with $p = 0.75$ over that of $p = 1$ is visually clear in Fig. 6. Note that in Fig. 6 the components of two sparse signals are plotted over a value range of $[-0.03, 0.03]$ for better visualization.

5. Conclusions

New algorithms for sparse representation based on ℓ_p - ℓ_2 optimization with $0 < p < 1$ are proposed. The algorithms are built on MFISTA. In particular, the soft shrinkage step in MFISTA is replaced by a global solver

for the minimization of a 1-D nonconvex ℓ_p problem. Two efficient techniques for solving the 1-D ℓ_p problem in question are proposed. Simulation studies for sparse representations are presented to evaluate the performance of the proposed algorithms with various values of p and compare with the basis pursuit (BP) benchmark with $p = 1$.

ACKNOWLEDGEMENT

The authors are grateful to the Natural Sciences and Engineering Research Council of Canada (NSERC) for supporting this research.

References

- [1] S. Chen, D. Donoho, and M. Saunders, “Atomic decomposition by basis pursuit,” *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.
- [2] J. Tropp and S. Wright, “Computational methods for sparse solution of linear inverse problems,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 948–958, 2010.
- [3] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way, Third Edition*. Elsevier Inc., 2009.
- [4] M. Zibulevsky and M. Elad, “L1-L2 optimization in signal and image processing,” *Signal Processing Magazine, IEEE*, vol. 27, no. 3, pp. 76–88, 2010.
- [5] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, p. p. 183–202, 2009.
- [6] —, “Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems,” *Image Processing, IEEE Transactions on*, vol. 18, no. 11, pp. 2419–2434, 2009.
- [7] R. Chartrand, “Exact reconstruction of sparse signals via nonconvex minimization,” *Signal Processing Letters, IEEE*, vol. 14, no. 10, pp. 707–710, 2007.
- [8] R. Chartrand and W. Yin, “Iteratively reweighted algorithms for compressive sensing,” in *Acoustics, Speech and Signal Processing, IEEE International Conference on*. IEEE, 2008, pp. 3869–3872.
- [9] J. Tropp, “Greed is good: Algorithmic results for sparse approximation,” *Information Theory, IEEE Transactions on*, vol. 50, no. 10, pp. 2231–2242, 2004.
- [10] A. Antoniou and W.-S. Lu, *Practical Optimization: Algorithms and Engineering Applications*. Springer-Verlag New York Inc, 2007.