Basis Pursuit Denoise with Nonsmooth Constraints
with applications to compressed sensing

Robert Baraldi[1], Rajiv Kumar[2], and Aleksandr Aravkin[1]

[1]Department of Applied Mathematics, University of Washington
[2]Formerly School of Earth and Atmospheric Sciences, Georgia Institute of Technology, USA; Currently DownUnder GeoSolutions, Perth, Australia

Acknowledgments: The authors acknowledge support from the Department of Energy Computational Science Graduate Fellowship, which is provided under grant number DE-FG02-97ER25308.

1 Introduction

- Basis Pursuit Denoise (BPDN) seeks sparse solution to an ill-posed system of equations corrupted by noise.
- Classic level set/Morozov formulation [1]:

$$\min_x \phi(\mathcal{C}(x)) \quad \text{s.t.} \quad \psi(\mathcal{A}(x)-b) \le \sigma, \tag{1}$$

for $\phi(\cdot) = \ell_1$, $\psi(\cdot) = \ell_2$ and $\mathcal{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^d$ a linear functional taking $x \in \mathbb{R}^{m\times n}$ to observations $b \in \mathbb{R}^d$ within error σ.
- **Applications** in low-rank interpolation and denoising; promote sparse representations in Fourier [4] or Curvelet [3] domains.
- **Noise** is falsely assumed to be smooth, Gaussian ℓ_2 norm; prior work exploits the smoothness of inequality constraint in Eq. 1.
- The **problem**: BPDN uses nonsmooth regularizers, but the inequality constraint is ubiquitously smooth.
- Contributions:
  - Fast, easily adaptable algorithm to solve non-smooth and nonconvex data constraints in general level-set formulations for large-scale interpolation and denoising problems.
  - Simple convergence criteria to critical points for nonconvex/nonsmooth formulations of Eq. 1.

2 Nonsmooth/nonconvex level-set

2.1 Problem Assumptions

- Eq. 1 ϕ and ψ may be nonsmooth, nonconvex, but have well-defined proximity and projection operators:

$$\begin{aligned}\text{prox}_{\eta\phi}(y) &= \arg\min_x \frac{1}{2\eta}\|x-y\|^2 + \phi(x)\\ \text{proj}_{\psi(\cdot)\le\sigma} &= \arg\min_{\psi(x)\le\sigma} \frac{1}{2\eta}\|x-y\|^2.\end{aligned} \tag{2}$$

- $\mathcal{C} : \mathbb{C}^{m\times n} \to \mathbb{R}^c$ is a linear transform domain operator.
- $\mathcal{A} : \mathbb{C}^{m\times n} \to \mathbb{R}^d$ is a linear observation/restriction operator.

2.2 Relaxation & Naive Algorithm

- Relax ϕ, ψ in Eq. (1) from $\mathcal{A}$ and $\mathcal{C}$ with $w_1 \in \mathbb{R}^c$ and $w_2 \in \mathbb{R}^d$

$$\begin{aligned}\min_{x,w_1,w_2} \phi(w_1) + \frac{1}{2\eta_1}\|\mathcal{C}(x)-w_1\|^2 + \frac{1}{2\eta_2}\|w_2-\mathcal{A}(x)+b\|_2^2\\ \text{s.t.} \quad \psi(w_2)\le\sigma.\end{aligned} \tag{3}$$

- Force $\eta_i \to 0$ in order to solve the original formulation (1).
- Naive algorithm: prox-gradient descent 1: $z = [x, w_1, w_2]^T$ and

$$f(z) = \frac{1}{2}\left\| \begin{bmatrix} \frac{1}{\sqrt{\eta_1}}\mathcal{C} & -\frac{1}{\sqrt{\eta_1}}I & 0 \\ \frac{1}{\sqrt{\eta_2}}\mathcal{A} & 0 & -\frac{1}{\sqrt{\eta_2}}I \end{bmatrix} z - \begin{bmatrix} 0 \\ b \end{bmatrix} \right\|^2$$

and $\Phi(z) = \phi(w_1) + \delta_{\psi(\cdot)\le\sigma}(w_2)$ for *indicator function* $\delta_{\psi(\cdot)\le\sigma}$.
- Apply the prox-gradient descent iteration with step-size β

$$z^{k+1} = \text{prox}_{\beta\Phi}(z^k - \beta\nabla f(z^k)) \tag{4}$$

Algorithm 1 Prox-gradient for (3).

Input: x^0, w_1^0, w_2^0
Initialize: $k = 0$
while not converged **do**
  $x^{k+1} \leftarrow x^k - \beta\left(\frac{1}{\eta_1}\mathcal{C}^T(\mathcal{C}(x) - w_1) + \frac{1}{\eta_2}\mathcal{A}^T(\mathcal{A}(x) - w_2^k - b)\right)$
  $w_1^{k+1} \leftarrow \text{prox}_{\beta\phi}\left(w_1^k - \frac{\beta}{\eta_1}(w_1^k - \mathcal{C}(x^{k+1}))\right)$
  $w_2^{k+1} \leftarrow \text{proj}_{\sigma\mathbb{B}_\psi}\left(w_2^k - \frac{\beta}{\eta_2}(w_2^k - (\mathcal{A}(x^{k+1}) - b))\right)$
  $k \leftarrow k+1$
end while
Output: w_1^k, w_2^k, x^k

2.3 Convergence and Reduction

- Problem 3 is semi-algebraic $\Rightarrow$ Alg. 1 $\to$ critical point [2].

Corollary 2.1 (Rate for Algorithm 1)**.** *For $min_z p(z) := \frac{1}{2}\|Gz - g\|^2 + \Phi(z)$, Problem 3 gives*

$$\min_{k=0,\dots,N}\|\nu^{k+1}\|^2 \le C(\eta_1,\eta_2,\mathcal{C},\mathcal{A})\frac{1}{N}(p(z^0) - \inf p)$$

with $\nu^k = (\|G\|_2^2 I - G^T G)(z^k - z^{k+1}) \in \partial p(z^{k+1})$ and

$$C(\eta_1,\eta_2,\mathcal{C},\mathcal{A}) = \frac{1}{\eta_1}(c + \|\mathcal{C}\|_F^2) + \frac{1}{\eta_2}(d + \|\mathcal{A}\|_F^2).$$

- Dependent on size of operators; can impose reductions.
- Solve x directly via the gradient, create block matrix $\mathcal{F}$:

$$x(w) = \mathcal{H}^{-1}\left(\begin{bmatrix}\frac{\mathcal{C}^T}{\eta_1} & \frac{\mathcal{A}^T}{\eta_2}\end{bmatrix} w + \frac{\mathcal{A}^T b}{\eta_2}\right),\ \mathcal{H} = \frac{\mathcal{C}^T\mathcal{C}}{\eta_1} + \frac{\mathcal{A}^T\mathcal{A}}{\eta_2}$$

$$\min_{w_1,w_2} p(w) := \phi(w_1) + \left\|\mathcal{F}w - \tilde{b}\right\|^2 \quad \text{s.t.} \quad \psi(w_2) \le \sigma \tag{5}$$

- Prox-gradient applied to the *value function* $p(w)$ in (5) with step β:

$$w^+ = \text{prox}_{\beta\Phi}(w^k - \beta\mathcal{F}^T(\mathcal{F}w - \tilde{b})) \tag{6}$$

- Compute optimal β by bounding singular values:

Lemma 2.2 (Bound on $\|\mathcal{F}^T\mathcal{F}\|_2$)**.** *The operator norm $\|\mathcal{F}^T\mathcal{F}\|_2$ is bounded above by* $\max\left(\frac{1}{\eta_1}, \frac{1}{\eta_2}\right)$.
- Combine iteration (6) with Corollary 2.1 to get a rate of convergence for Algorithm 2.

Corollary 2.3 (Convergence of Algorithm 2)**.** *When β satisfies*

$$\beta \le \min(\eta_1, \eta_2),$$

and $\eta_1 = \eta_2$, then for $\nu^k \in \partial p(w^k)$, the iterates of Alg. 2 satisfy

$$\min_{k=0,\dots,N}\|\nu^{k+1}\|^2 \le \frac{1}{N}\max\left(\frac{1}{\eta_1}, \frac{1}{\eta_2}\right)(p(w^0) - \inf p)).$$

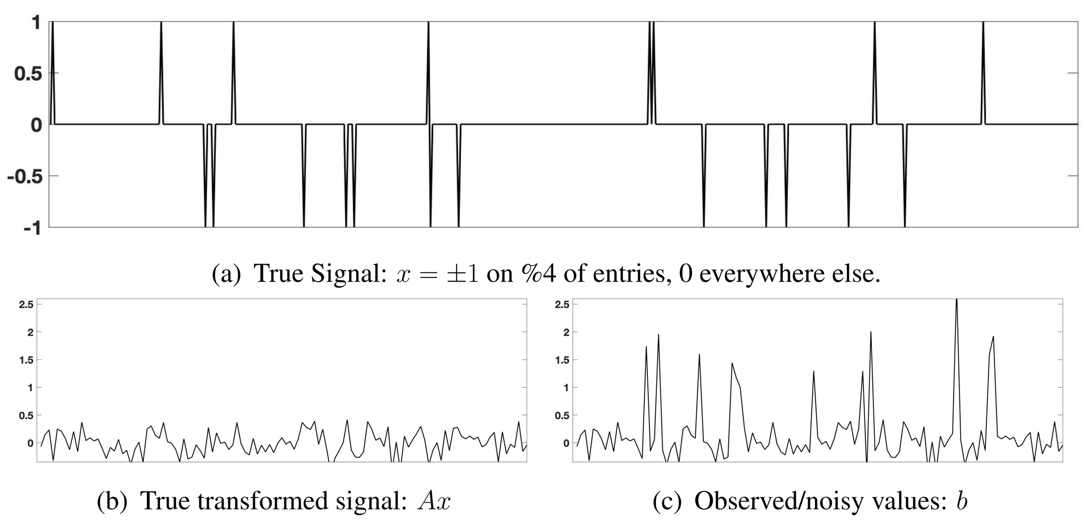
Algorithm 2 Block-coordinate descent for (3).

Input: x^0, w_1^0, w_2^0
Initialize: $k = 0$
Define: $\mathcal{H} = \frac{1}{\eta_1}\mathcal{C}^T\mathcal{C} + \frac{1}{\eta_2}\mathcal{A}^T\mathcal{A}$
while not converged **do**
  $x^{k+1} \leftarrow \mathcal{H}^{-1}\left(\frac{1}{\eta_1}\mathcal{C}^T w_1^k + \frac{1}{\eta_2}\mathcal{A}^T(b + w_2^k)\right)$
  $w_1^{k+1} \leftarrow \text{prox}_{\eta_1\phi}\left(\mathcal{C}(x^{k+1})\right)$
  $w_2^{k+1} \leftarrow \text{proj}_{\sigma\mathbb{B}_\psi}\left(\mathcal{A}(x^{k+1}) - b\right)$
  $k \leftarrow k+1$
end while
Output: w_1^k, w_2^k, x^k

- Convergence rate of Alg. 2 independent of $\mathcal{C}$ & $\mathcal{A}$; only η_i.
- Reduce FLOPs: compute x inexactly with fixed # PCG iterations.
- Continuation in η_i drives (η_1, η_2) to $(0, 0)$ at the same rate, and warm-starting each problem at the previous solution.

3 Basis Pursuit: Spike Train

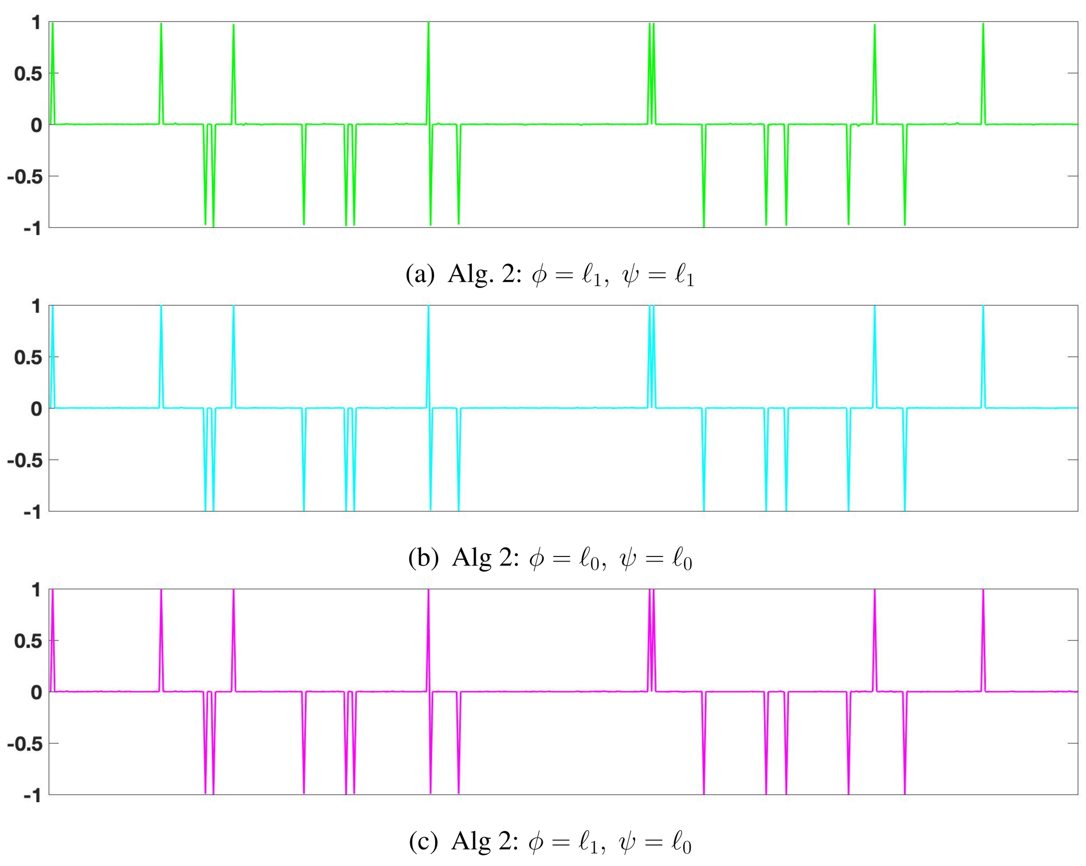
- **Goal:** Recapture spike train from observations $b \in \mathbb{R}^m$ corrupted with large sparse noise (10%, clean elsewhere) and known Gaussian operator $A \in \mathbb{R}^{n,m}$ ($(n, m) = (120, 512)$).



(a) True Signal: $x = \pm 1$ on %4 of entries, 0 everywhere else.
(b) True transformed signal: Ax
(c) Observed/noisy values: b

Figure 1: True signal, transformed signal, and noisy signals.

- **Results:** Recovered spike train with different ϕ and ψ.



(a) Alg. 2: $\phi = \ell_1$, $\psi = \ell_1$
(b) Alg 2: $\phi = \ell_0$, $\psi = \ell_0$
(c) Alg 2: $\phi = \ell_1$, $\psi = \ell_0$

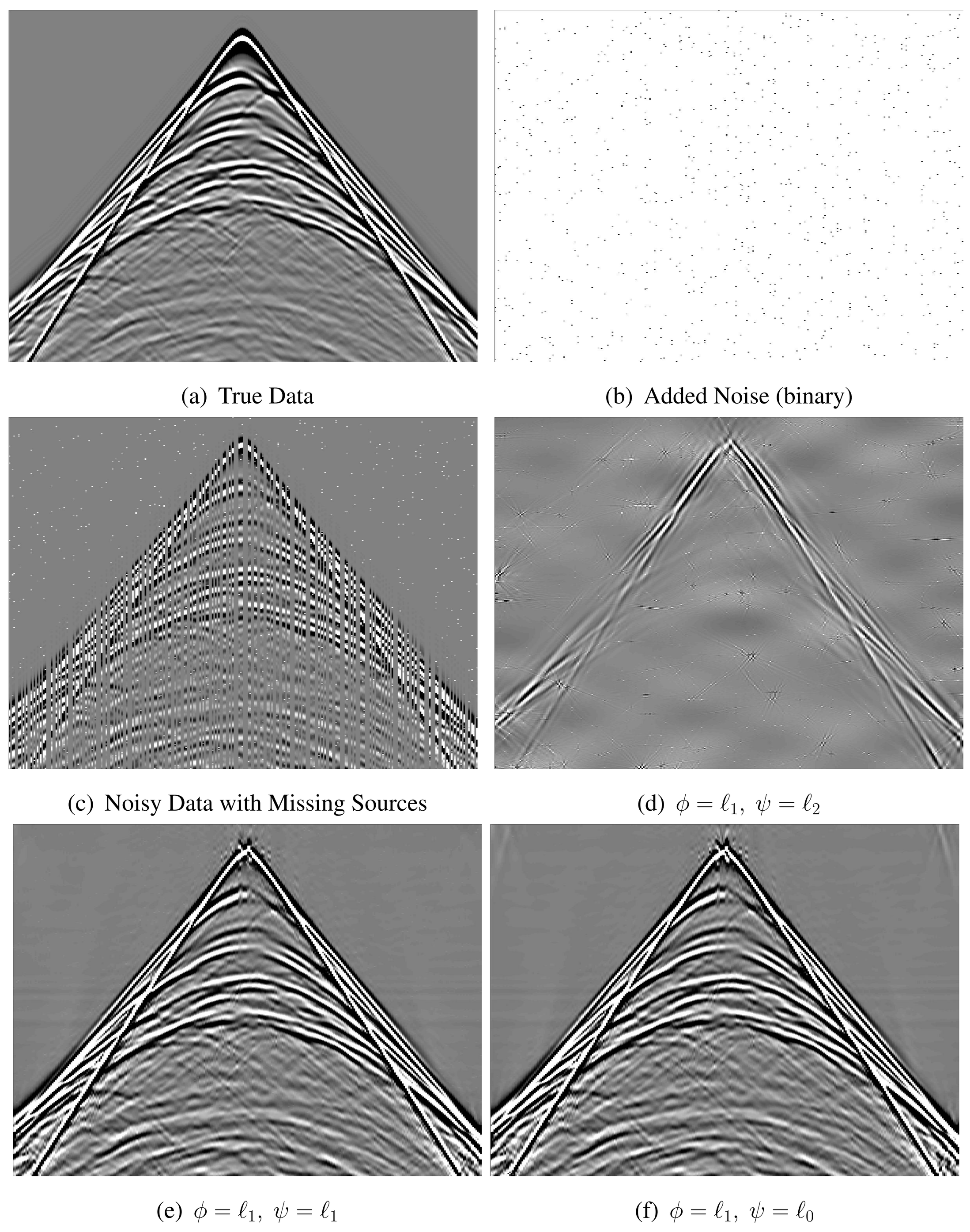
Figure 2: Sparse signal results solving Problem 1 where ϕ and ψ are varied. The ℓ_1- and ℓ_0 norms can capture the outliers only.

Table 1: SNR values against the true x for different combinations of sparsity-inducing $\phi = \ell_1$, ℓ_0 and $\psi = \ell_2$, ℓ_0 norms with SPGL1, CVX, and Alg. 2.

Spike-Train BPDN		
$\phi(\cdot)/\psi(\cdot)$	Method	SNR
ℓ_1 / ℓ_2	SPGL1	0.2007
ℓ_1 / ℓ_2	Alg.2	0.2032
ℓ_1 / ℓ_1	CVX	35.3611
ℓ_1 / ℓ_1	Alg.2	33.7281
ℓ_1 / ℓ_0	Alg.2	45.0601
ℓ_0 / ℓ_0	Alg.2	44.4239

4 Basis Pursuit: Curvelets

- **Goal:** Recover missing sources and denoise observed sources while enforcing sparsity in the Curvelet domain. Data has temporal sampling of 4ms, and spatial sampling is at 10ms.
- **Results:** Recaptured source-gathers with different choices of ψ and ϕ, successfully enforcing sparse-noise constraint.



(a) True Data
(b) Added Noise (binary)
(c) Noisy Data with Missing Sources
(d) $\phi = \ell_1$, $\psi = \ell_2$
(e) $\phi = \ell_1$, $\psi = \ell_1$
(f) $\phi = \ell_1$, $\psi = \ell_0$

Figure 3: Interpolation and denoising results for BPDN in the curvelet domain. Observe the complete inaccuracy of smooth norms with large, sparse noise.

Table 2: Curvelet Interpolation and Denoising results with different combinations of sparsity-inducing $\phi = \ell_1$, ℓ_0, and $\psi = \ell_2$, ℓ_0 norms for BPDN (1).

Curvelet Interpolation & Denoising				
$\phi(\cdot)/\psi(\cdot)$	Method	SNR	SNR w_1	Time (s)
ℓ_1 / ℓ_2	SPGL1	1.4857	-	51.4 (early stoppage)
ℓ_1 / ℓ_2	Alg.2	0.9769	0.9693	4043
ℓ_1 / ℓ_1	Alg.2	14.9574	14.9436	5037
ℓ_1 / ℓ_0	Alg.2	14.9212	14.9142	4256
ℓ_0 / ℓ_0	Alg.2	14.042	13.7999	4086

5 Conclusions & Future Directions

- Reduce Problem 1 to sum of quadratic and nonconvex regularizer, allowing simple proximal gradient method.
- Clear rate of convergence, independent of $\mathcal{C}$ and $\mathcal{A}$.
- Proposed a novel approach for level-set formulations, with extensions to residual-constrained low-rank formulations.
- Easily adapted to a variety of nonsmooth and nonconvex ϕ, ψ.
- Algorithms are simple, scalable, and efficient.

References

[1] A. Y. Aravkin, J. V. Burke, D. Drusvyatskiy, M. P. Friedlander, and S. Roy. Level-set methods for convex optimization. *To appear in Mathematical Programming, Series B.*, 2018.

[2] H. Attouch, J. Bolte, and B. Fux Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods. *Mathematical Programming*, 137(1-2):91–129, 2013.

[3] F. J. Herrmann and G. Hennenfent. Non-parametric seismic data recovery with curvelet frames. *Geophysical Journal International*, 173(1):233–248, 2008.

[4] M. D. Sacchi, T. J. Ulrych, and C. J. Walker. Interpolation and extrapolation using a high-resolution discrete fourier transform. *IEEE Transactions on Signal Processing*, 46(1):31–38, Jan 1998.