SIMULTANEOUS POLYNOMIAL APPROXIMATION AND TOTAL VARIATION DENOISING

Ivan W. Selesnick

Polytechnic Institute of New York University, Brooklyn, NY 11201, USA

ABSTRACT

This paper addresses the problem of smoothing data with additive step discontinuities. The problem formulation is based on least square polynomial approximation and total variation denoising. In earlier work, an ADMM algorithm was proposed to minimize a suitably defined sparsity-promoting cost function. In this paper, an algorithm is derived using the majorization-minimization optimization procedure. The new algorithm converges faster and, unlike the ADMM algorithm, has no parameters that need to be set. The proposed algorithm is formulated so as to utilize fast solvers for banded systems for high computational efficiency. This paper also gives optimality conditions so that the optimality of a result produced by the numerical algorithm can be readily validated.

1. INTRODUCTION

The problem of smoothing data with additive step discontinuties (jumps) was recently addressed in Ref. [15]. This problem arises in biological and biomedical signal processing, wherein step discontinues may represent either events to be detected, or artifacts to be removed. As in [15], we assume the noisy data $y(n)$ has the form

$$y(n) = p(n) + x(n) + w(n), \quad n = 0, \dots, N-1, \tag{1}$$

where $p(n)$ is a low-order polynomial of order $d \ll N$, $x(n)$ is approximately piecewise constant, and $w(n)$ is white Gaussian noise.

In the earlier work [15], the estimation of $p(n)$ and $x(n)$ was based on least square polynomial approximation and total variation denoising [5, 13]. The problem formulation (PATV) involved the minimization of a non-differentiable sparsity-promoting cost function; and an algorithm was derived using the alternating direction method of multipliers (ADMM) [1, 4, 7, 9]. The method is a suitable candidate for filtering any data consisting of step discontinuities on a low-frequency background. In [15], the method was demonstrated on data produced by a whispering gallery mode biosensor [2, 8].

This paper revisits the PATV problem and presents an improved algorithm. The new algorithm is derived using the majorization-minimization (MM) optimization procedure instead of ADMM. The new algorithm converges faster than the ADMM algorithm, and unlike the ADMM algorithm, has no parameters that need to be set. The algorithm is devised so that matrix operations involve only banded matrices; therefore, fast solvers for banded systems can be used to achieve very high computational efficiency.

This paper also gives the optimality conditions for the PATV problem so that the optimality of a result produced by the numerical algorithm can be readily validated. This condition was not given in the original paper [15]. In addition, the algorithm below is developed for a general penalty function, while the derivation in [15] is focused on the ℓ_1 norm. While general penalty functions via reweighted ℓ_1 minimization were described in [15] (based on a nested loop), the derivation here is simpler and more computationally efficient.

This work is supported by the NSF under Grant No. CCF-1018020.
Email: selesi@poly.edu

Notation. The N-point signal $\mathbf{x}$ is represented by the vector

$$\mathbf{x} = [x(0), \dots, x(N-1)]^T.$$

The matrices $\mathbf{D}$ and $\mathbf{S}$ denote the following forms:

$$\mathbf{D} = \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & -1 & 1 & \\ & & & -1 & 1 \end{bmatrix}, \qquad \mathbf{S} := \begin{bmatrix} 0 & & & \\ 1 & & & \\ 1 & 1 & & \\ 1 & 1 & 1 & \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

The first-order difference of an N-point signal $\mathbf{x}$ is given by $\mathbf{Dx}$ where $\mathbf{D}$ has size $(N-1) \times N$. The cumulative sum of an $(N-1)$-point signal $\mathbf{u}$ is given by $\mathbf{Su}$ where $\mathbf{S}$ has size $N \times (N-1)$. Note:

$$\mathbf{DS} = \mathbf{I}, \tag{2}$$

i.e., $\mathbf{S}$ is a discrete anti-derivative. As operators on vectors, each of $\mathbf{D}$, $\mathbf{D}^T$, $\mathbf{S}$, $\mathbf{S}^T$ require about N operations.

Let $\mathbf{G}$ denote the tall matrix of size $N \times (d+1)$, the columns of which form an orthonormal basis for polynomials of order d on $\{0, \dots, N-1\}$. The matrix $\mathbf{G}$ satisfies

$$\mathbf{G}^T\mathbf{G} = \mathbf{I}, \tag{3}$$

and may be obtained[1] by orthogonalizing the columns of the tall Vandermonde matrix, $\mathbf{V}$,

$$[\mathbf{V}]_{n,k} = n^k, \; 0 \le n \le N-1, \; 0 \le k \le d. \tag{4}$$

In the following, we will use the MM procedure [10], which minimizes a convex function $F(\mathbf{x})$, using the iteration

$$\mathbf{x}^{(k+1)} = \arg\min_{\mathbf{x}} G(\mathbf{x}, \mathbf{x}^{(k)}) \tag{5}$$

where $G(\mathbf{x}, \mathbf{v})$ is a convex majorizor of $F(\mathbf{x})$ that coincides with $F(\mathbf{x})$ at $\mathbf{x} = \mathbf{v}$. That is, $G(\mathbf{x}, \mathbf{v}) \geqslant F(\mathbf{x}) \; \forall \mathbf{x}$, and $G(\mathbf{v}, \mathbf{v}) = F(\mathbf{v})$. For more details, see Ref. [10] and references therein.

2. POLYNOMIAL APPROXIMATION AND TV DENOISING

In reference to (1), in order to simultaneously estimate $p(n)$ and $x(n)$, it was proposed in [15] that the polynomial coefficients $\mathbf{a}$ and the signal $\mathbf{x}$ be jointly found so as to solve the minimization problem:

$$(\mathbf{a}^*, \mathbf{x}^*) = \arg\min_{\mathbf{a},\mathbf{x}} \frac{1}{2} \sum_{n=0}^{N-1} |y(n) - p(n) - x(n)|^2 + \sum_{n=1}^{N-1} \phi(x(n) - x(n-1)) \tag{6}$$

[1]In MATLAB: `G = orth(bsxfun(@power,(0:N-1)',0:d))`

where $p(n)$ is a polynomial,

$$p(n) = a_0 + a_1\, n + \cdots + a_d\, n^d, \tag{7}$$

and $\phi : \mathbb{R} \to \mathbb{R}$ is a sparsity-promoting penalty function, such as

$$\phi(u) = \lambda |u| \quad \text{or} \quad \phi(u) = (\lambda/\alpha)\log(1 + \alpha|u|). \tag{8}$$

When $\phi(u) = \lambda|u|$, then the regularization term in (6) is equal to $\lambda\|\mathbf{D}\mathbf{x}\|_1$, where $\|\cdot\|_1$ denotes the ℓ_1 norm, the standard convex sparsity promoting penalty function [6, 17].

We refer to (6) as the PATV problem. It is defined by $\{\mathbf{y}, d, \phi\}$. An iterative algorithm is developed below to obtain the optimal $\mathbf{a}$ and $\mathbf{x}$. First, the PATV problem (6) is equivalent to[2]

$$(\mathbf{a}^*, \mathbf{x}^*) = \arg\min_{\mathbf{a},\mathbf{x}} \; \frac{1}{2}\,\|\mathbf{y} - \mathbf{G}\mathbf{a} - \mathbf{x}\|_2^2 + \sum_n \phi([\mathbf{D}\mathbf{x}]_n) \tag{9}$$

where $\mathbf{G}$ is given above. Note that $\mathbf{a}^*$ can be expressed explicitly:

$$\mathbf{a}^* = \mathbf{G}^T(\mathbf{y} - \mathbf{x}). \tag{10}$$

Note that (10) does not uniquely determine $\mathbf{x}$ from $\mathbf{a}^*$ because $\mathbf{G}^T$ is a wide matrix.

Substituting (10) into (9), the PATV problem can be written as

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{H}(\mathbf{y} - \mathbf{x})\|_2^2 + \sum_n \phi([\mathbf{D}\mathbf{x}]_n) \tag{11}$$

where $\mathbf{H}$ is given by

$$\mathbf{H} := \mathbf{I} - \mathbf{G}\mathbf{G}^T. \tag{12}$$

Algorithm. Note that adding a constant to $\mathbf{x}^*$ does not change the value of the cost function (11). (The operator $\mathbf{D}$ clearly annihilates constants. Also, as a constant signal is exactly represented as a polynomial of degree zero, $\mathbf{H}$ also annihilates constants.) The minimizer $\mathbf{x}^*$ is unique only upto an additive constant. Therefore, with out loss of generality, we may write $\mathbf{x} = \mathbf{S}\mathbf{u}$ where $\mathbf{S}$ is the cumulative sum matrix and $\mathbf{u}$ is length $N-1$. Then using (2), we have

$$\mathbf{D}\mathbf{x} = \mathbf{D}\mathbf{S}\mathbf{u} = \mathbf{u} \tag{13}$$

and the optimization problem in (11) can be written as

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} \Big\{ F(\mathbf{u}) = \frac{1}{2}\|\mathbf{H}(\mathbf{y} - \mathbf{S}\mathbf{u})\|_2^2 + \sum_n \phi(u(n)) \Big\}. \tag{14}$$

To minimize the cost function we will use the MM procedure. Let $g(u, v)$ be a quadratic majorizer of $\phi(u)$, defined as

$$g(u, v) = \frac{\phi'(v)}{2v}\, u^2 + \phi(v) - \frac{v}{2}\phi'(v). \tag{15}$$

See [14] for a derivation and illustration of $g(u, v)$. Then

$$g(u, v) \geq \phi(u) \quad \text{for all } u \in \mathbb{R} \tag{16}$$

$$g(v, v) = \phi(v) \quad \text{for } v \neq 0. \tag{17}$$

The majorizor g can be used to obtain a majorizor for $F(\mathbf{u})$ in (14). If $\mathbf{u}$ and $\mathbf{v}$ are vectors, then

$$\sum_n g(u(n), v(n)) \geq \sum_n \phi(u(n)) \tag{18}$$

[2] In (9), polynomial p is represented using the orthonormal basis $\mathbf{G}$.

with equality if $\mathbf{u} = \mathbf{v}$. That is, the left-hand-side of (18) is a majorizor for $\sum_n \phi(u(n))$. Moreover, the left-hand-side of (18) can be written compactly as

$$\sum_n g(u(n), v(n)) = \frac{1}{2}\mathbf{u}^T\mathbf{W}(\mathbf{v})\,\mathbf{u} + c \tag{19}$$

where $\mathbf{W}(\mathbf{v})$ is a diagonal matrix defined by

$$[\mathbf{W}(\mathbf{v})]_{n,n} = \frac{\phi'(v(n))}{v(n)} \tag{20}$$

and

$$c = \sum_n \phi(v(n)) - \frac{v}{2}\phi'(v(n)). \tag{21}$$

Therefore, using (18), a majorizor for $F(\mathbf{u})$ is given by

$$G(\mathbf{u}, \mathbf{v}) = \frac{1}{2}\|\mathbf{H}\mathbf{y} - \mathbf{H}\mathbf{S}\mathbf{u}\|_2^2 + \frac{1}{2}\mathbf{u}^T\mathbf{W}(\mathbf{v})\,\mathbf{u} + c. \tag{22}$$

$G(\mathbf{u}, \mathbf{v})$ is quadratic in $\mathbf{u}$. Hence, minimizing $G(\mathbf{u}, \mathbf{v})$ wrt $\mathbf{u}$ gives

$$\mathbf{u} = (\mathbf{S}^T\mathbf{H}^T\mathbf{H}\mathbf{S} + \mathbf{W}(\mathbf{v}))^{-1}\mathbf{S}^T\mathbf{H}^T\mathbf{H}\mathbf{y} \tag{23}$$

where $\mathbf{W}(\mathbf{v})$ depends on $\mathbf{v}$ per (20). Note that $\mathbf{H}^T\mathbf{H} = \mathbf{H}$, so

$$\mathbf{u} = (\mathbf{S}^T\mathbf{H}\mathbf{S} + \mathbf{W}(\mathbf{v}))^{-1}\mathbf{S}^T\mathbf{H}\mathbf{y}. \tag{24}$$

Therefore, the MM update produces the sequence

$$\mathbf{u}^{(k+1)} = \operatorname*{argmin}_{\mathbf{u}} G(\mathbf{u},\, \mathbf{u}^{(k)}) \tag{25}$$

$$= (\mathbf{S}^T\mathbf{H}\mathbf{S} + \mathbf{W}^{(k)})^{-1}\mathbf{S}^T\mathbf{H}\mathbf{y} \tag{26}$$

where we use the notation $\mathbf{W}^{(k)} := \mathbf{W}(\mathbf{u}^{(k)})$, i.e.,

$$[\mathbf{W}^{(k)}]_{n,n} = \frac{\phi'(u^{(k)}(n))}{u^{(k)}(n)}. \tag{27}$$

There are two problems with (26).

1. The update in (26) requires the solution to a large system of equations which involving order N^2 operations.
2. If components of $\mathbf{u}^{(k)}$ go to zero, then the entries of $\mathbf{W}^{(k)}$ go to infinity. Hence, the update equation (26) may become numerically inaccurate. Moreover, because the solution $\mathbf{u}$ is expected to be sparse, generally some components of $\mathbf{u}^{(k)}$ will go to zero.

Both problems are avoided, as described in Ref. [10], by using the matrix inverse lemma (MIL). Here, we use MIL twice to derive an efficient and stable implementation of (26). First, write:

$$\mathbf{S}^T\mathbf{H}\mathbf{S} + \mathbf{W}^{(k)} = \mathbf{S}^T(\mathbf{I} - \mathbf{G}\mathbf{G}^T)\mathbf{S} + \mathbf{W}^{(k)} \tag{28}$$

$$= \mathbf{W}^{(k)} + \mathbf{S}^T\mathbf{S} - \mathbf{S}^T\mathbf{G}\mathbf{G}^T\mathbf{S} \tag{29}$$

$$= \mathbf{A}^{(k)} - \mathbf{B}\mathbf{B}^T \tag{30}$$

where we define

$$\mathbf{A}^{(k)} := \mathbf{W}^{(k)} + \mathbf{S}^T\mathbf{S}, \qquad \mathbf{B} := \mathbf{S}^T\mathbf{G}.$$

With this notation, by the matrix inverse lemma, we can write

$$(\mathbf{S}^T\mathbf{H}\mathbf{S} + \mathbf{W}^{(k)})^{-1} = [\mathbf{A}^{(k)}]^{-1} + [\mathbf{A}^{(k)}]^{-1}\mathbf{B}\Big[\mathbf{I} - \mathbf{B}^T[\mathbf{A}^{(k)}]^{-1}\mathbf{B}\Big]^{-1}\mathbf{B}^T[\mathbf{A}^{(k)}]^{-1} \tag{31}$$

An efficient implementation of $[\mathbf{A}^{(k)}]^{-1}$ can be obtained as follows. Use the matrix inverse lemma again to write

$$[\mathbf{A}^{(k)}]^{-1} = [\mathbf{W}^{(k)} + \mathbf{S}^T\mathbf{S}]^{-1} \tag{32}$$

$$= [\mathbf{W}^{(k)}]^{-1} - [\mathbf{W}^{(k)}]^{-1}\Big[(\mathbf{S}^T\mathbf{S})^{-1} + [\mathbf{W}^{(k)}]^{-1}\Big]^{-1}[\mathbf{W}^{(k)}]^{-1}.$$

Several observations can be made. We use the notation $\mathbf{\Lambda}^{(k)} = [\mathbf{W}^{(k)}]^{-1}$. From (27), this is the diagonal matrix given by

$$[\mathbf{\Lambda}^{(k)}]_{n,n} = \frac{u^{(k)}(n)}{\phi'(u^{(k)}(n))}. \tag{33}$$

Note that if $\phi(u) = \lambda|u|$, then $u/\phi'(u) = |u|/\lambda$. If $\phi(u) = (\lambda/\alpha)\log(1+\alpha|u|)$, then $u/\phi'(u) = |u|(1+\alpha|u|)/\lambda$.

Note that, as $u^{(k)}(n)$ goes to zero, the value $[\mathbf{\Lambda}^{(k)}]_{n,n}$ does not go to infinity. Also, importantly, the matrix $(\mathbf{S}^T\mathbf{S})^{-1}$ is of the form

$$(\mathbf{S}^T\mathbf{S})^{-1} = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}. \tag{34}$$

In particular, this is a *banded* matrix. Defining

$$\mathbf{R}^{(k)} = (\mathbf{S}^T\mathbf{S})^{-1} + \mathbf{\Lambda}^{(k)}, \tag{35}$$

then (32) can be written as

$$[\mathbf{A}^{(k)}]^{-1} = \mathbf{\Lambda}^{(k)}\left(\mathbf{I} - [\mathbf{R}^{(k)}]^{-1}\mathbf{\Lambda}^{(k)}\right) \tag{36}$$

where $\mathbf{R}^{(k)}$ is a banded matrix. In fact, it is a tridiagonal matrix, so the operator $[\mathbf{R}^{(k)}]^{-1}$ can be implemented exactly and efficiently using fast solvers for banded systems [12, Sect 2.4].

The update in (31) also requires the implementation of the operator $[\mathbf{Q}^{(k)}]^{-1}$ where

$$\mathbf{Q}^{(k)} := \mathbf{I} - \mathbf{B}^T[\mathbf{A}^{(k)}]^{-1}\mathbf{B}. \tag{37}$$

However, $\mathbf{Q}$ is a small matrix of size $(d+1)\times(d+1)$, where d is the order of the low-order polynomial $p(n)$. The matrix $\mathbf{B}$ is of size $(N-1)\times(d+1)$ and need be computed a single time and saved. At each iteration, the matrix $\mathbf{Q}^{(k)}$ must be computed, which can be done efficiently using the implementation of $[\mathbf{A}^{(k)}]^{-1}$ described in (36) just above. By the preceding identities, (31) can be written as

$$\begin{aligned}&(\mathbf{S}^T\mathbf{H}\mathbf{S} + \mathbf{W}^{(k)})^{-1} \\ &\quad = [\mathbf{A}^{(k)}]^{-1}\left[\mathbf{I} + \mathbf{B}\,[\mathbf{Q}^{(k)}]^{-1}\,\mathbf{B}^T\,[\mathbf{A}^{(k)}]^{-1}\right]\end{aligned} \tag{38}$$

and can be implemented as an operator with order Nd operations.

Therefore, the update equation (26) that constitutes the MM algorithm, can be written as

$$\mathbf{b} = \mathbf{S}^T\mathbf{H}\mathbf{y} \tag{39}$$

$$\mathbf{u}^{(k+1)} = [\mathbf{A}^{(k)}]^{-1}\Big[\mathbf{b} + \mathbf{B}\,[\mathbf{Q}^{(k)}]^{-1}\,\mathbf{B}^T\,[\mathbf{A}^{(k)}]^{-1}\mathbf{b}\Big] \tag{40}$$

This is a computationally stable and efficient implementation to solve problem (14). The algorithm is summarized in Table 1.[3]

[3] A MATLAB program implementing the algorithm is available online at `http://eeweb.poly.edu/iselesni/patv/`.

Input: $\mathbf{y} \in \mathbb{R}^N$, d, ϕ

1. $\mathbf{b} = \mathbf{S}^T\mathbf{H}\mathbf{y}$
2. $\mathbf{B} = \mathbf{S}^T\mathbf{G}$
3. $\mathbf{u} = \mathbf{1}$ (initialization)

repeat

4. $\quad\mathbf{\Lambda}_{n,n} = \dfrac{u(n)}{\phi'(u(n))}$
5. $\quad\mathbf{R} = (\mathbf{S}^T\mathbf{S})^{-1} + \mathbf{\Lambda}$
6. $\quad\mathbf{A}^{-1} = \mathbf{v} \mapsto \mathbf{\Lambda}(\mathbf{v} - \mathbf{R}^{-1}\mathbf{\Lambda}\mathbf{v})$
7. $\quad\mathbf{Q} = \mathbf{I} - \mathbf{B}^T\mathbf{A}^{-1}(\mathbf{B})$
8. $\quad\mathbf{u} = \mathbf{A}^{-1}(\mathbf{b} + \mathbf{B}\,\mathbf{Q}^{-1}\mathbf{B}^T\mathbf{A}^{-1}(\mathbf{b}))$

until convergence

9. $\mathbf{x} = \mathbf{S}\mathbf{u}$
10. $\mathbf{a} = \mathbf{G}^T(\mathbf{y} - \mathbf{x})$

output: $\mathbf{x}$, $\mathbf{a}$

Table 1. An MM algorithm for solving the PATV problem. The algorithm uses fast solvers for banded systems.

Zero-locking. Note that if $u^{(k)}(n) = 0$ for some index n and some iteration k, then $[\mathbf{\Lambda}^{(k)}]_{n,n} = 0$ and therefore $u^{(k+1)}(n) = 0$. Once a component of $\mathbf{u}^{(k)}$ is zero, then it will be zero in all subsequent iterations of the algorithm. That is, the zero is 'locked-in'. This zero-locking behavior is a well known phenomenon of some classes of algorithms [10]. To account for this, the initialization should be entirely non-zero, $u^{(0)}(n) \neq 0$ for all n. As discussed in [10,11], the zero-locking behavior does not necessarily impede the convergence of algorithms in which it occurs. In Table 1, the initialization is $u^{(0)}(n) = 1$ for all n, but other initializations can be used. We have found experimentally, that the algorithm converges reliably, as illustrated in the example below.

2.1. Optimality Conditions

When the penalty function ϕ is convex, then the minimizer $\mathbf{u}^*$ of (14) must satisfy certain conditions, as described in Ref. [3, Prop 1.3]. These conditions can be used to verify the optimality of a solution produced by a numerical algorithm.

Suppose ϕ is convex. If $\mathbf{u}$ solves (14), then $\mathbf{u}$ must satisfy:

$$[\mathbf{S}^T\mathbf{H}^T\mathbf{H}(\mathbf{y} - \mathbf{S}\mathbf{u})]_n \in \partial\phi(u(n)), \quad \forall n \tag{41}$$

where $[\mathbf{v}]_n$ denotes the n-th component of the vector $\mathbf{v}$ and $\partial\phi(\cdot)$ is the subdifferential, a *set*-valued generalization of the derivative. Using $\mathbf{H}^T\mathbf{H} = \mathbf{H}$, $\mathbf{x} = \mathbf{S}\mathbf{u}$, and $\mathbf{u} = \mathbf{D}\mathbf{x}$, we may write

$$[\mathbf{S}^T\mathbf{H}(\mathbf{y} - \mathbf{x})]_n \in \partial\phi([\mathbf{D}\mathbf{x}]_n), \quad \forall n. \tag{42}$$

With $\phi(u) = \lambda|u|$, condition (42) is given by

$$\begin{aligned} \big[\mathbf{S}^T\mathbf{H}(\mathbf{y}-\mathbf{x})\big]_n &= \lambda, & u(n) > 0 \\ -\lambda \le \big[\mathbf{S}^T\mathbf{H}(\mathbf{y}-\mathbf{x})\big]_n &\le \lambda, & u(n) = 0 \\ \big[\mathbf{S}^T\mathbf{H}(\mathbf{y}-\mathbf{x})\big]_n &= -\lambda, & u(n) < 0 \end{aligned} \tag{43}$$

as illustrated in Fig. 1 below.

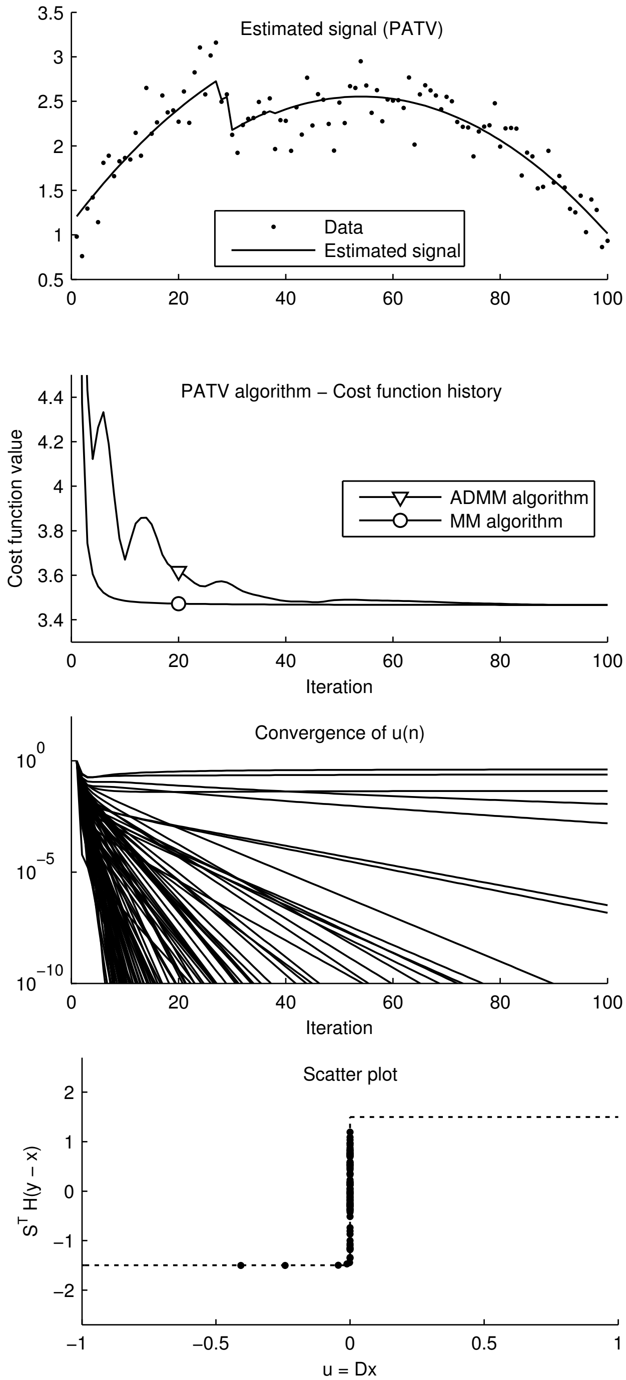

Fig. 1. Example: Polynomial approximation of noisy data with an additive step discontinuity. The penalty function is $\phi(u) = \lambda|u|$.

2.2. Example

To illustrate the PATV (polynomial approximation / total variation denoising) problem and its solution, we consider the data illustrated in Fig. 1, consisting of a second order polynomial, an additive step discontinuity, and white Gaussian noise ($\sigma = 0.25$). With $d = 2$, $\phi(u) = \lambda|u|$, and $\lambda = 1.5$, the result of the PATV algorithm is illustrated in Fig. 1. The figure also illustrates the cost function history

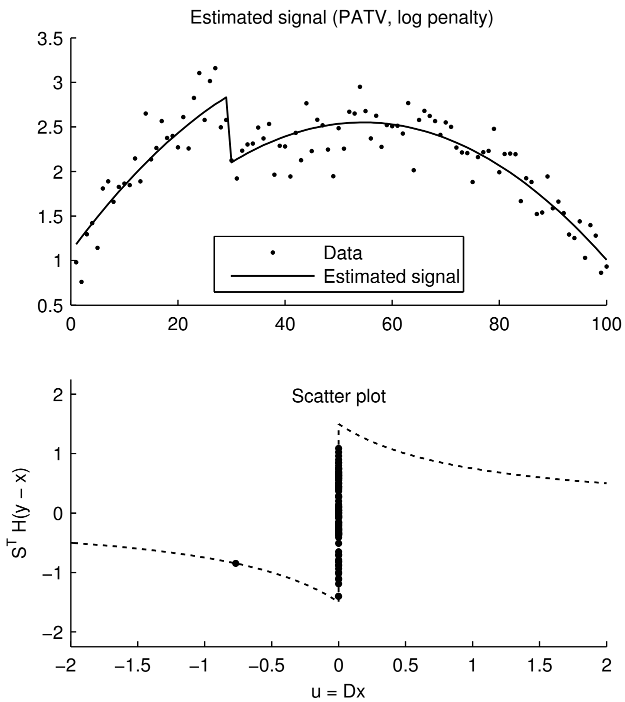

Fig. 2. Example as in Fig. 1. Here, the penalty function is non-convex, $\phi(u) = (\lambda/\alpha)\log(1 + \alpha|u|)$ with $\lambda = 1.5$, $\alpha = 1.0$.

$F(\mathbf{u}^{(k)})$ of both the proposed MM algorithm and of the ADMM algorithm [15]. The MM algorithm converges substantially faster. Moreover, the ADMM algorithm requires two parameters (μ_i) that must be carefully set so as to avoid slow convergence. The proposed MM algorithm requires no user specified parameter.

The convergence of the MM algorithm is further illustrated by showing the value $u^{(k)}(n)$ for each index n for 100 iterations. In this example, $u^{(0)}(n) = 1$ for all n. It can be seen that many $u(n)$ rapidly converge to zero. In addition, the optimality of the obtained solution is validated in the scatter plot of $\mathbf{S}^T\mathbf{H}(\mathbf{y} - \mathbf{x})$ versus $\mathbf{D}\mathbf{x}$. According to (42), the solution is optimal if the points in the scatter plot lie on the dashed lines.

Now, we set the penalty to $\phi(u) = (\lambda/\alpha)\log(1 + \alpha|u|)$ with $\lambda = 1.5$, $\alpha = 1.0$. This non-convex function promotes sparsity more strongly than the ℓ_1 norm. The PATV algorithm produces the result shown in Fig. 2. In this solution, $\mathbf{u}^*$ has only a single non-zero value, consistent with the simulated data. Although ϕ is not convex, the scatter plot of $\mathbf{S}^T\mathbf{H}(\mathbf{y} - \mathbf{x})$ versus $\mathbf{D}\mathbf{x}$ lies on $\phi'(u)$ indicated by the dashed lines.

3. CONCLUSION

This paper revisits the polynomial approximation / total variation denoising (PATV) problem formulated in [15]. A new algorithm is derived with faster convergence and without the need of user supplied parameters. The algorithm developed here is for a general penalty function ϕ; therefore non-convex penalty functions can be more readily and efficiently used to obtain enhanced sparsity, in comparison with [15]. This paper also gives the optimality conditions for the PATV problem, not given in [15], by which optimality/convergence can be readily validated. Recently, in [16], low-pass filtering is used in place of polynomial approximation.

4. REFERENCES

[1] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. Fast image recovery using variable splitting and constrained optimization. *IEEE Trans. Image Process.*, 19(9):2345–2356, September 2010.

[2] S. Arnold, M. Khoshima, I. Teraoka, S. Holler, and F. Vollmer. Shift of whispering-gallery modes in microspheres by protein adsorption. *Opt. Lett*, 28(4):272–274, February 15, 2003.

[3] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. *Foundations and Trends in Machine Learning*, 4(1):1–106, 2012.

[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

[5] T. F. Chan, S. Osher, and J. Shen. The digital TV filter and nonlinear denoising. *IEEE Trans. Image Process.*, 10(2):231–241, February 2001.

[6] S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20(1):33–61, 1998.

[7] P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, editors, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. Springer-Verlag (New York), 2010.

[8] V. R. Dantham, S. Holler, V. Kolchenko, Z. Wan, and S. Arnold. Taking whispering gallery-mode single virus detection and sizing to the limit. *Appl. Phys. Lett.*, 101(4), 2012.

[9] E. Esser, X. Zhang, and T. F. Chan. A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. *SIAM J. Imag. Sci.*, 3(4):1015–1046, 2010.

[10] M. Figueiredo, J. Bioucas-Dias, and R. Nowak. Majorization-minimization algorithms for wavelet-based image restoration. *IEEE Trans. Image Process.*, 16(12):2980–2991, December 2007.

[11] J. Oliveira, J. Bioucas-Dias, and M. A. T. Figueiredo. Adaptive total variation image deblurring: A majorization-minimization approach. *Signal Processing*, 89(9):1683–1693, September 2009.

[12] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C: the art of scientific computing (2nd ed.)*. Cambridge University Press, 1992.

[13] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[14] I. Selesnick. Penalty and shrinkage functions for sparse signal processing. *Connexions Web site*, 2012. http://cnx.org/content/m45134/1.1/.

[15] I. W. Selesnick, S. Arnold, and V. R. Dantham. Polynomial smoothing of time series with additive step discontinuities. *IEEE Trans. Signal Process.*, 60(12):6305–6318, December 2012.

[16] I. W. Selesnick, H. L. Graber, D. S. Pfeil, and R. L. Barbour. Simultaneous low-pass filtering and total variation denoising. Preprint, http://eeweb.poly.edu/iselesni/lpftvd/, 2012.

[17] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc., Ser. B*, 58(1):267–288, 1996.