Exact algorithms for L^1-TV regularization of real-valued or circle-valued signals

Martin Storath*, Andreas Weinmann†, Michael Unser*

April 3, 2015

Abstract

We consider L^1-TV regularization of univariate signals with values on the real line or on the unit circle. While the real data space leads to a convex optimization problem, the problem is non-convex for circle-valued data. In this paper, we derive exact algorithms for both data spaces. A key ingredient is the reduction of the infinite search spaces to a finite set of configurations, which can be scanned by the Viterbi algorithm. To reduce the computational complexity of the involved tabulations, we extend the technique of distance transforms to non-uniform grids and to the circular data space. In total, the proposed algorithms have complexity $\mathcal{O}(KN)$ where N is the length of the signal and K is the number of different values in the data set. In particular, for boundedly quantized data, the complexity is $\mathcal{O}(N)$.

Keywords: Total variation regularization, total cyclic variation, circle-valued data, least absolute deviations, dynamic programming, distance transform

1 Introduction

Total variation (TV) minimization has become a standard method for jump or edge preserving regularization of signals and images. Whereas the classical L^2-TV model (i.e., TV with quadratic data fidelity term [31]) is optimally matched to the Gaussian noise model, L^1 data terms are more robust to noise with more heavy tailed distributions such as Laplacian noise, and in the presence of outliers; see, e.g., [29]. Further advantages are the better preservation of the contrast and the invariance to global contrast changes [7]. Since L^1-TV minimization is a convex problem for real- and vector-valued data, it is accessible by convex optimization techniques. In fact, there are several algorithms for L^1-TV minimization with scalar and vectorial data. The minimization methods are typically of iterative nature: for example, interior point methods [21], iterative thresholding [2], alternating methods of multipliers [23, 36], semismooth Newton methods [9], primal-dual strategies [6, 15], and proximal point methods [28] were employed. Further algorithms are based on recursive median filtering [1] or graph cuts [12].

*Biomedical Imaging Group, École Polytechnique Fédérale de Lausanne, Switzerland.
†Department of Mathematics, Technische Universität München, and Helmholtz Zentrum München, Germany.

For univariate real-valued signals, there are efficient exact algorithms available. It is well known that the univariate L^2-TV problem can be solved exactly in linear time using the taut string algorithm [13, 27]. A recent alternative is the algorithm of Condat [10] which shows a particularly good performance in practice. The L^1-TV problem is computationally more intricate. For data $y \in \mathbb{R}^N$ and a non-negative weight vector $w \in \mathbb{R}^N$, it is given by

$$\arg\min_{x \in \mathbb{R}^N} \; \alpha \sum_{n=1}^{N-1} |x_n - x_{n+1}| + \sum_{n=1}^{N} w_n |x_n - y_n|, \tag{1}$$

where $\alpha > 0$ is a model parameter regulating the tradeoff between data fidelity and TV prior. In a Bayesian framework, it corresponds to the maximum a posteriori estimator of a summation process with Laplace distributed increments under a Laplacian noise model; see, e.g., [34]. Kovac and Dümbgen [16] have derived an exact solver of complexity $O(N \log N)$ for (1). While finalizing the present paper, the authors became aware of the recent preprint of Kolmogorov et al. [25] which describes a solver of complexity $O(N \log \log N)$. The differences of these methods to our approach will be discussed below.

Recently, total variation regularization on non-vectorial data spaces such as, e.g., Riemannian manifolds has received a lot of interest [8, 11, 24, 26, 38]. The non-vectorial setting is a major challenge because the total variation problem is, in general, not anymore convex. One of the simplest examples, where the L^1-TV functional is a nonconvex functional on, are circle-valued data. Such data appear, for example, as phase signals (which are defined modulo 2π) and as time series of angles. Particular examples for the latter are the data on the orientation of the bacterial flagellar motor [32] and the data on wind directions [14]. The L^1-TV functional for circle-valued data $y \in \mathbb{T}^N$ is given by

$$\arg\min_{x \in \mathbb{T}^N} \; \alpha \sum_{n=1}^{N-1} d_{\mathbb{T}}(x_n, x_{n+1}) + \sum_{n=1}^{N} w_n \, d_{\mathbb{T}}(x_n, y_n), \tag{2}$$

where $d_{\mathbb{T}}(u, v)$ denotes the arc length distance of $u, v \in \mathbb{T} = \mathbb{S}^1$. Theoretical results on total cyclic variation can be found in Giaquinta et al. [22] and in Cremers and Strekalovskiy [11]. The authors of the latter paper have shown that the problem is computationally at least as complex as the Potts problem; this means in particular, that it is NP-hard in dimensions greater than one. Current minimization strategies for (2) are based on convex relaxations [11], proximal point splittings [38], or iteratively reweighted least squares [24]. However, due to the non-convexity, these iterative approaches do not guarantee convergence to a global minimizer. Furthermore, they are computationally demanding. To our knowledge, no exact algorithm for (2) has been proposed yet.

In this paper, we propose exact non-iterative algorithms for L^1-TV minimization on scalar signals (1) and on circle-valued signals (2). A key ingredient is the reduction of the infinite search space, $\mathbb{R}^N$ or $\mathbb{T}^N$, to a finite search space V^N. This reduction allows us to use the Viterbi algorithm [20, 35] for the minimization of discretized energies as presented in [17]. A time-critical step in the Viterbi algorithm is the computation of a distance transform w.r.t. the non-uniform grid induced by V. For the scalar case, we generalize the efficient two-pass algorithm of

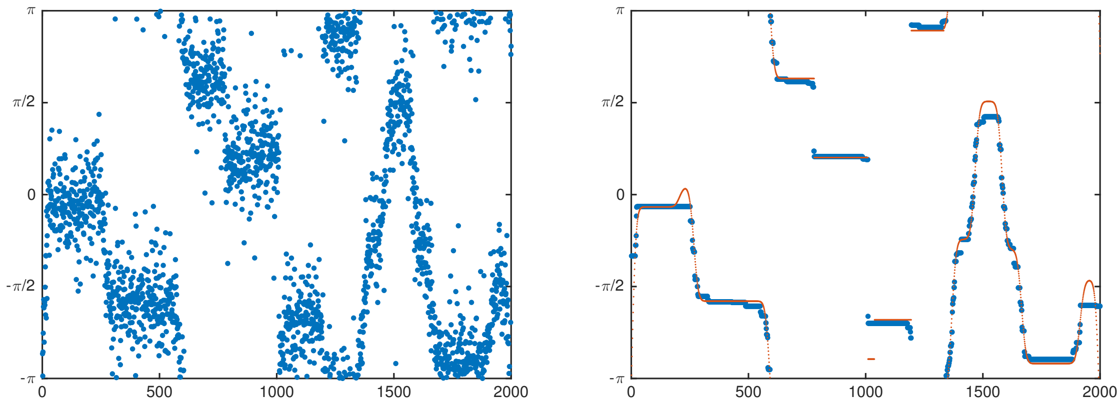

Figure 1: *Left:* Synthetic circle-valued signal corrupted by noise. *Right:* Global minimizer of the TV functional with $\alpha = 15$. (Ground truth displayed as small red points.) The noise is almost completely removed and the jumps are preserved. The phase jumps of 2π are taken into account properly.

Felzenszwalb and Huttenlocher [18, 19] from uniform grids to our non-uniform setup. We further propose a new method for efficiently computing the distance transforms in the circle-valued case. In total, our solvers have complexity $O(KN)$ where K denotes the number of different values in the data. In particular, if the data is quantized to finitely many levels, the algorithmic complexity is $O(N)$.

We briefly discuss the difference of our solver to other exact algorithms for the L^1-TV problem with real-valued data (1). The solver of Dümbgen and Kovac [16] is based on a generalization of the taut string algorithm combining isotonic and antitonic regression functions which is absolutely unrelated to our approach. The recent preprint of Kolmogorov et al. [25], which we became aware of while finishing the present manuscript, seems at first glance to be related to our method because the authors also utilize dynamic programming techniques. However, our method is fundamentally different from the approach of [25]: there, the solver is based on dynamically removing and appending breakpoints, whereas our method performs a scanning over the elements of the finite search space V^N. Finally, we stress that neither [16] nor [25] do propose solvers for the circle-valued case (2).

The basic principle of search space reduction and efficient scanning is related to the authors' work [33] on jump-penalized estimators. We emphasize that the similarity to the present work are limited to this meta strategy. The cornerstones of the methods, i.e., the search space reduction and the efficient computation, are based on completely different techniques.

1.1 Organization of the paper

In Section 2, we show that the search space can be reduced to a finite set. In Section 3, we present the dynamic programming strategy for the reduced problem. In Section 4, we drive numerical experiments based on synthetic and real data.

2 Reduction of the search space

A crucial step for our strategy to derive fast and exact algorithms is the reduction of the search space to a finite set. In the following, we denote the L^1-TV functional for data $y \in \mathbb{R}^N$ or $y \in \mathbb{T}^N$ by

$$T_{\alpha;y}(x) = \alpha \sum_{n=1}^{N-1} d(x_n, x_{n+1}) + \sum_{n=1}^{N} w_n \, d(x_n, y_n).$$

Here d denotes the distance that corresponds to the data space, i.e., the Euclidean distance for real-valued data and the arc length distance for circle-valued data. We further use the notation $\mathrm{Val}(y)$ to denote the set of values of the N-tuple y, i.e.,

$$\mathrm{Val}(y) = \{v : \text{ there is } n \text{ with } 1 \leq n \leq N \text{ s.t. } y_n = v\}.$$

Also recall that a (weighted) median of y is a minimizer of the functional

$$\mu \mapsto \sum_{n=1}^{N} w_n d(\mu, y_n).$$

We show that there are always minimizers of the TV problem whose values are all contained in the values $\mathrm{Val}(y)$ of the data y (united with the antipodal points $\mathrm{Val}(\tilde{y})$ in the circle-valued case).

2.1 Real-valued data

Let us first consider the real-valued case. We have learned that the following assertion had been shown already in the paper of Alliney [1]. However, the proof in [1] is based on results of convex analysis which hampers the transfer to the nonconvex circle-valued setup. Here, we give a completely different proof that does not exploit the convexity. The advantage of our technique is that it can be utilized in a similar form for the nonconvex circle-valued case.

Theorem 1. *Let $\alpha > 0$, $y \in \mathbb{R}^N$, and $V = \mathrm{Val}(y)$. Then*

$$\min_{x \in \mathbb{R}^N} T_{\alpha;y}(x) = \min_{x \in V^N} T_{\alpha;y}(x).$$

Proof. The method of proof is as follows: we consider an arbitrary $x \in \mathbb{R}^N$ and construct $x' \in V^N$ such that $T_{\alpha;y}(x') \leq T_{\alpha;y}(x)$. If we apply this procedure to a minimizer x^*, we obtain a minimizer with values in V^N which is the assertion of the theorem.

So let $x \in \mathbb{R}^N$ be arbitrary and let us construct $x' \in V^N$ with smaller or equal $T_{\alpha;y}$ value by the following procedure. Let $\mathcal{I}$ be the set of maximal intervals of $\{1, \ldots, N\}$ where x is constant on and where x does not attain its value in V. That is, for each interval of the form $\{l, \ldots, r\} \in \mathcal{I}$ we have that $a := x_l = \ldots = x_r \notin V$. We decrease the number of such intervals $|\mathcal{I}|$ by the following rule: Choose an interval $I = \{l, \ldots, r\} \in \mathcal{I}$. We construct $\bar{x}$ which equals x outside I and choose its constant value a' such that the corresponding number of intervals with values which are not in V is strictly smaller than $|\mathcal{I}|$. We distinguish three cases.

First assume that I is no boundary interval and that the values of the neighboring intervals, i.e., x_{l-1} and x_{r+1}, are both smaller than the value on I which equals a. We denote the nearest smaller and the nearest greater neighbors of a in V by b^- and b^+, respectively. Let $b' = \max(x_{l-1}, x_{r+1}, b^-)$. By replacing a by some $a' \in [b', b^+]$ we change the total variation penalty by $2\alpha(a' - a)$ and the data penalty by $(W^- - W^+)(a' - a)$. Here $W^+ = \sum_{\{i \in I: y_i > a\}} w_i$ and $W^- = \sum_{\{i \in I: y_i \leq a\}} w_i$ are the weights of elements in the interval I that are greater or smaller than a, respectively. If $W^- + 2\alpha < W^+$, we let a' equal its greater neighbor b^+. Otherwise, we let $a' = b'$ where, by the definition above, $b' = \max(x_{l-1}, x_{r+1}, b^-)$. If $a' = b^-$, then the value of $\bar{x}$ on I belongs to V. If $a' \in \{x_{l-1}, x_{r+1}\}$ the interval merges with one of its neighbors. In both cases, the number of intervals with "undesired" values, $|\mathcal{I}|$, decreases by one. By symmetry, the same argumentation is valid for the case that the values of the neighboring intervals x_{l-1} and x_{r+1} are both greater than a.

As second case we consider the situation where I is no boundary interval, and where x_{l-1} is smaller and x_{r+1} is greater than a. (Again, the case $x_{l-1} > a > x_{r+1}$ is dealt with by symmetry.) Since replacing a by any value in $[x_{l-1}, x_{r+1}]$ does not change the total variation penalty, we only need to look at the approximation error. This amounts to setting a' equal to a (weighted) median of $y_l, \ldots, y_r$. Note that there exists a (weighted) median that it is contained in $\{y_l, \ldots, y_r\} \subset V$. We use such a median in V to define a'. Hence, also in this case, $|\mathcal{I}|$ decreases by one.

Let us eventually consider the third case where the interval is located at the boundary. If either $1 \in I$ or $N \in I$ then we proceed analogously to the first case. The relevant difference is that we let a' equal its greater nearest neighbor b^+ if $W^- + \alpha < W^+$ (instead of $W^- + 2\alpha < W^+$). If the interval touches both boundaries, i.e., if $I = \{1, \ldots, N\}$, we proceed as in the second case, which is setting a' to be a (weighted) median y which is contained in V.

We repeat the above procedure until $|\mathcal{I}| = 0$ which implies that the final result x' is contained in V^N. By construction, the functional value $T_{\alpha;y}(x')$ is not exceeding the functional value of x, since all intermediately constructed $\bar{x}$ do so. This completes the proof. □

Note that the assertion of Theorem 1 is not true for quadratic data fidelities. As the following simple example shows, it is not uncommon that $\mathrm{Val}(\hat{x}) \cap \mathrm{Val}(y) = \emptyset$ for all L^2-TV minimizers $\hat{x}$. We consider toy data $y = (0, 1) \in \mathbb{R}^2$ and the corresponding L^2-TV functional given by $x \mapsto \alpha|x_1 - x_2| + x_1^2 + (x_2 - 1)^2$. It is easy to check that the unique minimizer of this L^2-TV problem is given by $\hat{x} = (\alpha/2, 1 - \alpha/2)$, if $\alpha < 1$, and by $\hat{x} = (1/2, 1/2)$, otherwise. We note that this is an example where $\mathrm{Val}(\hat{x}) \cap \mathrm{Val}(y) = \emptyset$ even for all $\alpha > 0$. This shows that one cannot even expect an analogous result when one chooses the correct parameter. For a more detailed discussion of this aspect we refer to the paper of Nikolova [29].

2.2 Circle-valued data

We use use the basic techniques developed for the real-valued case in our proof of Theorem 1 in the more involved situation of circle-valued data to prove the following theorem allowing for the reduction of the search space for minimizers of the L^1-TV functional for $\mathbb{S}^1$-valued data as well.

Theorem 2. *Let $\alpha > 0$, $y \in \mathbb{T}^N$, and $V = \mathrm{Val}(y) \cup \mathrm{Val}(\tilde{y})$, where $\tilde{y}$ denotes the tuple of antipodal points of y. Then*

$$\min_{x \in \mathbb{T}^N} T_{\alpha;y}(x) = \min_{x \in V^N} T_{\alpha;y}(x).$$

Proof. As in the proof of Theorem 1, we consider an arbitrary $x \in \mathbb{T}^N$ and construct $x' \in V^N$ such that $T_{\alpha;y}(x') \leq T_{\alpha;y}(x)$. Note that, in contrast to the proof of Theorem 1, $V = \mathrm{Val}(y) \cup \mathrm{Val}(\tilde{y})$ here. Similarly, we let $\mathcal{I}$ be the set of the maximal intervals I of $\{1, \ldots, N\}$ where x is constant on and where the attained value a of x on I is not contained in V. We decrease the number of such intervals $|\mathcal{I}|$ by the procedure explained below.

Before being able to give the explanation we need some notions related to $\mathbb{S}^1$ data. Let us consider a point a on the sphere and its antipodal point $\tilde{a}$. Then there are two hemisphere/half-circles connecting a and $\tilde{a}$. We use the convention that $\tilde{a}$ is contained in both hemispheres whereas a is contained in none of them. These two hemispheres can be distinguished into the hemisphere $H_1 = H_1(a)$ determined by walking from a in clockwise direction and the hemisphere $H_2 = H_2(a)$ obtained from walking in counter-clockwise direction.

Equipped with these preparations, we explain the procedure to reduce the number of intervals $|\mathcal{I}|$. We pick an arbitrary interval $I = \{l, \ldots, r\} \in \mathcal{I}$ and let $a = x_l = \ldots = x_r$ be the value of x on I. We let b_1 and b_2 be the nearest neighbors of a in $H_1 \cap V$ and in $H_2 \cap V$, which are the values of the data (or their antipodal points) on the clockwise and counter-clockwise hemisphere, respectively. We note that b_1, b_2 exist and both are not equal to the antipodal point $\tilde{a}$ of a. This is because, together with a point p, its antipodal point $\tilde{p}$ is also contained in V which implies that either p or $\tilde{p}$ is a member of H_1 and either $\tilde{p}$ or p is a member of H_2. Since a is no data point or its antipodal point, the distance to either p or $\tilde{p}$ is strictly smaller that π. We construct $\bar{x}$ which equals x outside I and with constant value a' on I such that $|\mathcal{I}|$ decreases. We have to differentiate three cases.

First we assume that I is no boundary interval and that the left and the right neighboring candidate item x_{l-1} and x_{r+1} are both located on the clockwise hemisphere H_1 and none of them agrees with $\tilde{a}$. Let $W_1 = \sum_{i:y_i \in H_1} w_i$ be the weight of y on H_1 and let $W_2 = \sum_{i:y_i \in H_2} w_i$ be the weight of y on H_2. (Note that $\tilde{a}$ which is the only point in both H_1 and H_2 is not a member of y.) If $W_1 > W_2 + 2\alpha$, which means that the clockwise hemisphere H_1 is "heavier" than the counterclockwise hemisphere H_2 plus the variation penalty, we set a' to be the nearest neighbor of a in $\{x_l, x_r, b_1\}$. This may be visualized as shifting the value on I in clockwise direction until we hit the first value in $\{x_l, x_r, b_1\}$. Since $W_1 > W_2 + 2\alpha$, we have that $T_{\alpha;y}(\bar{x}) \leq T_{\alpha;y}(x)$. Otherwise, we set $a' = b_2$ which means that we shift to the other direction. Since then $W_1 \leq W_2 + 2\alpha$, we get $T_{\alpha;y}(\bar{x}) \leq T_{\alpha;y}(x)$ also in this situation. By symmetry, the same argument applies when both x_{l-1} and x_{r+1} are located on the counterclockwise hemisphere.

In the second case we assume that I is no boundary interval and that x_{l-1} and x_{r+1} are located on different hemispheres. Here we also include the case where one or both x_{l-1} and x_{r+1} are antipodal to a. If only one neighbor is antipodal, we interpret it to lie on the opposite hemisphere of the non-antipodal member. If both neighbors are antipodal, we interpret them to lie on different hemispheres. We let C be the arc connecting x_{l-1} and x_{r+1} which has a as member. Letting a' equal any value on the arc C, leads to $TV(\bar{x}) \leq TV(x)$, meaning that it does not increase the variation penalty $TV(x) = \sum_n \alpha d_{\mathbb{T}}(x_n, x_{n+1})$. By definition, the data term is minimized by letting a' be a (weighted) median of $y_l, \ldots, y_r$. A (weighted) median of the circle-valued data can be chosen as an element of the data points $\{y_l, \ldots, y_r\}$ unified with the antipodal points $\{\tilde{y}_l, \ldots, \tilde{y}_r\}$. We choose a' as such a median. This implies $T_{\alpha;y}(\bar{x}) \leq T_{\alpha;y}(x)$.

It remains to consider the boundary intervals. If $I = \{1, \ldots, N\}$, we proceed as in the second case and set $\bar{u}_i = a'$ for all i, where a' is a (weighted) median of y which is contained in V. Else, if either $1 \in I$ or $N \in I$ we proceed analogously to the first case with the difference that we replace the decision criterion $W_1 > W_2 + 2\alpha$ employed there by $W_1 > W_2 + \alpha$.

We repeat the above procedure until $|\mathcal{I}| = 0$ which implies that the values of the final result x' all lie in V^N. Then plugging in a minimizer $x = x^*$, results in a minimizer $x' \in V^N$ which shows the theorem. □

As for scalar data, the assertion of Theorem 2 is not true for quadratic data terms. This can be seen using the previous example interpreting the data $y = (0, 1)$ as angles.

In order to illustrate the difference to the real-valued data case, let us point out a degenerate situation which is due to the circular nature of the data. Assume that the data only consists of a point $z \in \mathbb{T}$ and its antipodal point $\tilde{z}$, i.e., $y = (z, \tilde{z})$. For sufficiently large α, any minimizer $\hat{x}$ of (2) is constant; say $\hat{x} = (a, a)$. Since the TV penalty gets equal to zero, a must be equal to a median of y. It is not hard to check that every point on the sphere is a median of y. This behavior appears curious at first glance. However, the data shows no clear tendency towards a distinguished orientation. Thus, every estimate can be considered as equally good. The result seems even more natural than that of L^2-TV regularization. An L^2-TV minimizer would consists of one of the two "mean orientations" which are given by rotating z by $\pi/2$ in clockwise or counterclockwise direction. Both minimizers seem rather arbitrary, and, moreover, the two options point into opposing directions.

3 Efficient algorithms for the reduced problems

Theorem 1 and Theorem 2 allow us to reduce the infinite search spaces $\mathbb{R}^N$ and $\mathbb{T}^N$ in (1) and (2), respectively. to the finite sets V^N, which are specified in these theorems. Thus, it remains to solve the problems: find

$$x^* \in \arg\min_{x \in V^N} T_{\alpha;y}(x).$$

We use dynamic programming to compute minimizers of these reduced problems. For an early account on dynamic programming, we refer to [3]. The basic idea of dynamic programming is to decompose the problem into a series of similar, simpler and tractable subproblems.

3.1 The Viterbi algorithm for energy minimization on finite search spaces

We utilize a dynamic programming scheme developed by Viterbi [35]; see also [20]. Related algorithms have been proposed in [4, 5]. In this paragraph, we review a special instance of the Viterbi algorithm following the presentation of the survey [17].

We aim at minimizing an energy functional of the form

$$E(x_1, \ldots, x_N) = \alpha \sum_{n=1}^{N-1} d(x_n, x_{n+1}) + \sum_{n=1}^{N} w_n d(x_n, y_n) \tag{3}$$

where the arguments $x_1, \ldots, x_N$ can take values in a finite set $V = \{v_1, \ldots, v_K\}$. The Viterbi algorithm solves this problem in two steps: tabulation of energies and reconstruction by back-tracking.

For the tabulation step, the starting point is the table $B^1 \in \mathbb{R}^K$ given by

$$B_k^1 = w_1 d(v_k, y_1) \quad \text{for } k = 1, \ldots, K.$$

From now on, the symbol K denotes the cardinality of V. For $n = 2, \ldots, N$ we successively compute the tables $B^n \in \mathbb{R}^K$ which are given by

$$B_k^n = w_n d(v_k, y_n) + \min_l \{B_l^{n-1} + \alpha\, d(v_k, v_l)\}, \tag{4}$$

for $k = 1, \ldots, K$. The entry B_k^n represents the energy of a minimizer on data $(y_1, \ldots, y_n)$ whose endpoint is equal to v_k.

For the backtracking step, it is convenient to introduce an auxiliary tuple $l \in \mathbb{N}^N$ which stores minimizing indices. We initialize the last entry of l by $l_N = \arg\min_k B_k^N$. Then we successively compute the entries of l for $n = N-1, N-2, \ldots, 1$ by

$$l_n = \arg\min_k B_k^n + \alpha\, d(v_k, v_{l_{n+1}}). \tag{5}$$

Eventually, we reconstruct a minimizer $\hat{x}$ from the indices in l by

$$\hat{x}_n = v_{l_n}, \quad \text{for } n = 1, \ldots, N.$$

The result $\hat{x}$ is a global minimizer of the energy (3); see [17]. For a general functional, filling the table B^n in (4) costs $O(K^2)$. This implies that the described procedure is in $O(K^2 N)$. In the next subsections, we will derive procedures to reduce the complexity for filling the tables B^n for our concrete problem to $O(K)$.

3.2 Distance transform on a non-uniform real-valued grid

We first consider the case of real-valued data. The time critical part of the Viterbi algorithm is the computation of the minima

$$D_k = \min_l B_l + \alpha |v_k - v_l|, \quad \text{for all } k = 1, \ldots, K. \tag{6}$$

This problem is known as distance transform with respect to the ℓ^1 distance (weighted by α). Felzenszwalb and Huttenlocher [18, 19] describe an efficient algorithm for (6) when V forms an integer grid, i.e., $V = \{0, \ldots, K-1\}$. In our setup, V forms a non-uniform grid in general. Therefore, we generalize their method in a way such as to work with the nonuniform grid V.

In the following, we identify the elements of V with a K-dimensional vector v which is ordered in ascendingly, i.e., $v_1 < v_2 < \ldots < v_K$. The sorting causes no problems since we can sort v in $O(K \log K)$, and since the logarithm of the number of values K is smaller than the data length N, we have $O(K \log K) \subset O(KN)$.

As we will show below, the following two-pass procedure computes the real-valued distance transform D:

Algorithm 1: Real-valued distance transform distTransReal(B, v, α).

```
Input: B ∈ ℝ^K; v ∈ ℝ^K sorted in ascending order; α > 0;
Output: Distance transform D
begin
    D ← B;
    for k ← 2, 3, ..., K do
        D_k ← min(D_{k-1} + α(r_k − r_{k-1}); D_k)
    end
    for k ← K − 1, K − 2, ..., 1 do
        D_k ← min(D_{k+1} + α(r_{k+1} − r_k); D_k)
    end
    return D;
end
```

In order to show the correctness of the method, we build on the structurally related proof given in [18] for uniform grids. The major new idea is to pass from discrete to continuous infimal convolutions in order to deal with the nonequidistant grid. The (continuously defined) infimal convolution of two functions F and G on $\mathbb{R}$ with values on the extended real line $[-\infty, \infty]$ is given by

$$F \square G(r) = \inf_{u \in \mathbb{R}} \{F(u) + G(r - u)\},$$

see [30, Section 5]. In the following, the infimum will be always attained, so that it actually is a minimum; we use this fact in the notation we employ.

For real valued data, we get the following result accelerating the bottleneck operation in the general Viterbi algorithm from Section 3.1.

Theorem 3. *Algorithm 1 computes* (6) *in* $O(K)$.

Proof. We define the function F on $\mathbb{R}$ by $F(v_l) = B_l$ for $v_l \in V$ and by $F(r) = \infty$ for $r \in \mathbb{R} \setminus V$. Also define $G(u) = \alpha|u|$. Then, D_k can be formulated in terms of the infimal convolution of F and G evaluated at v_k, that is,

$$D_k = F \square G(v_k).$$

In order to decompose G, we define

$$G_+(r) = \begin{cases} \alpha r, & \text{for } r \geq 0, \\ \infty, & \text{otherwise,} \end{cases} \quad \text{and} \quad G_-(r) = \begin{cases} -\alpha r, & \text{for } r \leq 0, \\ \infty, & \text{otherwise.} \end{cases}$$

We see that G is the infimal convolution of G_+ and G_- by using that

$$G_+ \square G_-(r) = \min_{t \in \mathbb{R}} G_+(t) + G_-(r - t) = \alpha|r| = G(r).$$

By the associativity of the infimal convolution (see [30, Section 5]), we obtain

$$F \square G = F \square (G_+ \square G_-) = (F \square G_+) \square G_-. \tag{7}$$

We use the right-hand representation; for the right-hand term in brackets, we get, for $v_k \in V$,

$$\begin{aligned}
F\Box G_+(v_k) &= \min_j F(v_j) + G_+(v_k - v_j) \\
&= \min_{j\leq k} F(v_j) + \alpha(v_k - v_j) \\
&= \min\{\min_{j\leq k-1} F(v_j) + \alpha(v_k - v_j); F(v_k)\} \\
&= \min\{\min_{j\leq k-1} F(v_j) + \alpha(v_{k-1} - v_j + v_k - v_{k-1}); F(v_k)\} \\
&= \min\{F\Box G_+(v_{k-1}) + \alpha(v_k - v_{k-1}); F(v_k)\}.
\end{aligned}$$

Now, we denote the result by $F' = F\Box G_+$ and continue to manipulate the right-hand term of (7) noticing that, for all $r \notin V$, we have $F'(r) = \infty$. We obtain

$$\begin{aligned}
F'\Box G_-(v_k) &= \min_j F'(v_j) + G_-(v_k - v_j) \\
&= \min_{j\geq k} F'(v_j) - \alpha(v_k - v_j) \\
&= \min\{\min_{j\geq k+1} F'(v_j) - \alpha(v_k - v_j); F'(v_k)\} \\
&= \min\{\min_{j\geq k+1} F'(v_j) - \alpha(v_{k+1} - v_j + v_k - v_{k+1}); F'(v_k)\} \\
&= \min\{F\Box G_-(v_{k+1}) + \alpha(v_{k-1} - v_k); F'(v_k)\}.
\end{aligned}$$

The above recursive equations show that the forward pass and the backward pass of Algorithm 1 compute the desired infimal convolutions. □

3.3 Distance transform on a non-uniform circle-valued grid

Now we look at the circular case. In this case, the corresponding ℓ^1 distance transform is given by

$$D_k = \min_l B_l + \alpha d_{\mathbb{T}}(v_k, v_l), \quad \text{for all } k = 1, \ldots, K. \tag{8}$$

Our task is to compute the distance transform in the circle case as well. To this end, we employ the angular representation of values on the circle in the interval $(-\pi, \pi]$. As in the real-valued case, we identify the elements of V with a K-tuple v which is sorted in ascending order. In order to compute (8), we use the following algorithm:

Algorithm 2: Circle-valued distance transform distTransCirc(B, v, α).

Input: $B \in \mathbb{R}^K$; $v \in (-\pi, \pi]^K$ sorted in ascending order; $\alpha > 0$;
Output: Distance transform D
begin
  $B' \leftarrow (B_1, \ldots, B_K, B_1, \ldots, B_K, B_1, \ldots, B_K)$;
  $v' \leftarrow (v_1 - 2\pi, \ldots, v_K - 2\pi, v_1, \ldots, v_K, v_1 + 2\pi, \ldots, v_K + 2\pi)$;
  $D' \leftarrow$ distTransReal(B', v', α);
  $D \leftarrow (D'_{K+1}, \ldots, D'_{2K})$;
  return D;
end

We point out that this algorithm employs the real-valued distance transform of Section 3.2. The next result in particular shows that Algorithm 2 actually computes a minimizer of the distance transform (8). The proof uses infimal convolutions on the real line and employs the corresponding statement Theorem 3 for real-valued data.

Theorem 4. *Algorithm 2 computes* (8) *in* $O(K)$.

Proof. First we observe that the arc length distance on $\mathbb{S}^1 = \mathbb{T}$ can be written using the absolute value on $(-\pi, \pi]$ by

$$d_{\mathbb{T}}(u, w) = \min\{|u - 2\pi - w|; |u - w|; |u + 2\pi - w|\},$$

for $u, w \in (-\pi, \pi]$. We define the extended real-valued functions F, F' defined on $\mathbb{R}$ as follows: we let $F(v_k) = B_k$ on the points v_k and $F(r) = \infty$ for $r \in \mathbb{R} \setminus V$; to define F', we let

$$F'(t) = \min\{F(t - 2\pi), F(t), F(t + 2\pi)\}.$$

Our goal is to show that D_k is the infimal convolution of F' and G with G given by $G(v) = \alpha|v|$. We get that

$$\begin{aligned}
D_k &= \min_{r\in R}\{F(r) + \alpha\ \min\{|r - 2\pi - v_k|; |r - v_k|; |r + 2\pi - v_k|\}\} \\
&= \min_{r\in R} \min\{F(r) + \alpha|r - 2\pi - v_k|; \\
&\qquad F(r) + \alpha|r - v_k|; F(r) + \alpha|r + 2\pi - v_k|\}\} \\
&= \min\{\min_{r\in R} F(r + 2\pi) + \alpha|r - v_k|; \\
&\qquad \min_{r\in R} F(r) + \alpha|r - v_k|; \min_{r\in R} F(r - 2\pi) + \alpha|r - v_k|\}\} \\
&= \min\{\min_{r\in(-3\pi,\pi]} F(r + 2\pi) + \alpha|r - v_k|; \\
&\qquad \min_{r\in(-\pi,\pi]} F(r) + \alpha|r - v_k|; \min_{r\in(\pi,3\pi]} F(r - 2\pi) + \alpha|r - v_k|\}\} \\
&= \min_{r} F'(r) + \alpha|r - v_k| = F' \square G(v_k).
\end{aligned}$$

Hence, D_k is the infimal convolution of F' and G. We now shift the vector of assumed values v by -2π and 2π and consider the concatenation with v to obtain v' which is given by

$$v' = (v_1 - 2\pi, \ldots, v_K - 2\pi, v_1, \ldots, v_K, v_1 + 2\pi, \ldots, v_K + 2\pi).$$

We note that v' is ordered ascendingly. We let

$$D'_l = F' \square G(v'_l), \quad \text{for } l = 1, \ldots 3K. \tag{9}$$

By Theorem 3, we can compute (9) in $O(K)$ using Algorithm 1. Eventually, we observe that

$$D_k = F' \square G(v_k) = D'_{K+k}, \quad \text{for } k = 1, \ldots, K,$$

which completes the proof. □

Algorithm 3: Exact algorithm for the L^1-TV problem of real- or circle-valued signals

Input: Data $y \in \mathbb{R}^N$ or $y \in \mathbb{T}^N$; regularization parameter $\alpha > 0$; weights $w \in (\mathbb{R}_0^+)^N$;
Output: Global minimizer $\hat{x}$ of (1) or (2);
begin
  /* *1. Init candidate values* */
  $V \leftarrow \mathrm{Val}(y)$; /* *Real-valued case* */
  $V \leftarrow \mathrm{Val}(y) \cup \mathrm{Val}(\tilde{y})$; /* *Circle-valued case* */
  $v \leftarrow K$-tuple of elements of V, sorted ascendingly;
  /* *2. Tabulation* */
  for $k \leftarrow 1$ **to** K **do**
    $B_k^1 \leftarrow w_1 d(v_k, y_1)$;
  end
  for $n \leftarrow 2$ **to** N **do**
    $D \leftarrow \mathrm{distTransReal}(B^n, v, \alpha)$; /* *Real-valued case* */
    $D \leftarrow \mathrm{distTransCirc}(B^n, v, \alpha)$; /* *Circle-valued case* */
    for $k \leftarrow 1$ **to** K **do**
      $B_k^n \leftarrow w_n d(v_k, y_n) + D_k$;
    end
  end
  /* *3. Backtracking* */
  $l \leftarrow \arg\min_{k=1,\ldots,K} B_k^N$;
  $\hat{x}_n \leftarrow v_l$;
  for $n \leftarrow N-1, N-2, \ldots, 1$ **do**
    $l \leftarrow \arg\min_{k=1,\ldots,K} B_k^n + \alpha\, d(v_k, \hat{x}_{n+1})$;
    $\hat{x}_n \leftarrow v_l$;
  end
  return $\hat{x}$;
end

3.4 Complete algorithm

Eventually, we present the complete proposed procedure in Algorithm 3. Summarizing, we have obtained the following result:

Theorem 5. *Let $y \in \mathbb{R}^N$ and $V = \mathrm{Val}(y)$, or $y \in \mathbb{T}^N$ and $V = \mathrm{Val}(y) \cup \mathrm{Val}(\tilde{y})$. Further let K be the number of elements in V. Then Algorithm 3 computes a global minimizer of the L^1-TV problem with real-valued* (1) *or circle-valued data* (2) *in $O(KN)$. In particular, if data is quantized to a finite set, then the algorithms are in $O(N)$.*

4 Numerical results

We illustrate the effect of L^1-TV minimization algorithm based on synthetic and real life data. We focus on the circle-valued case. For illustrations of the real valued case, we exemplarily refer to [9, 16, 21, 25, 37]. We have implemented our algorithms in Matlab. The experiments were conducted on a desktop computer with 3.5 GHz Intel Xeon E5 and 32 GB memory. As it is common, the regularization parameter α is adjusted empirically.

In our first experiment (Figure 1), we compute the total variation minimizer for a synthetic signal with known ground truth. The noise is (wrapped) Laplacian distributed. The experiment illustrates the denoising capabilities of total variation minimization for circle-valued data. We also observe that the phase jumps by 2π are taken into account properly. The runtime was 3.5 seconds.

Next, we apply our algorithm to real data. The present data set consists of wind directions at the Station WPOW1 (West Point, WA), recorded every 10 minutes in the year 2014[1]. In total, the time series has $N = 52543$ elements. The values are quantized to $K = 360$ angles. The regularized signal allows to identify the time intervals of approximately constant wind direction, for instance.

5 Conclusion

We have derived exact algorithms for the L^1-TV problem with scalar and circle-valued data. A first crucial point was the reduction of the search space to a finite set, which allowed us to employ a dynamic programming strategy. The second key ingredient was a reduction of the computational complexity based on generalizations of distance transforms.

The algorithms have quadratic complexity in the worst case. The complexity is linear when the signal is quantized to a finite set. We note that such quantized signals appear frequently in practice, for example in digitalized audio signals or images.

To our knowledge, the proposed algorithm is the first exact solver for total variation regularization of circle-valued signals. Besides its application for the regularization of angular signals, it can be used as building block for higher dimensional problems as in [39] or as benchmark for iterative strategies, e.g., for those of [11, 24, 38].

The proposed approach appears to be unique for L^1 data terms. In particular, we have provided counterexamples that the utilized search space reduction is not valid for quadratic data terms. An exact and efficient algorithm for L^2-TV regularization of circle-valued signals remains an open question.

Acknowledgement

Martin Storath and Michael Unser were supported by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement no. 267439. Andreas Weinmann was supported by the Helmholtz Association within the young investigator group VH-NG-526. Martin Storath and Andreas Weinmann acknowledge the support by the DFG scientific network Mathematical Methods in Magnetic Particle Imaging.

[1] Data available at http://www.ndbc.noaa.gov/historical_data.shtml.

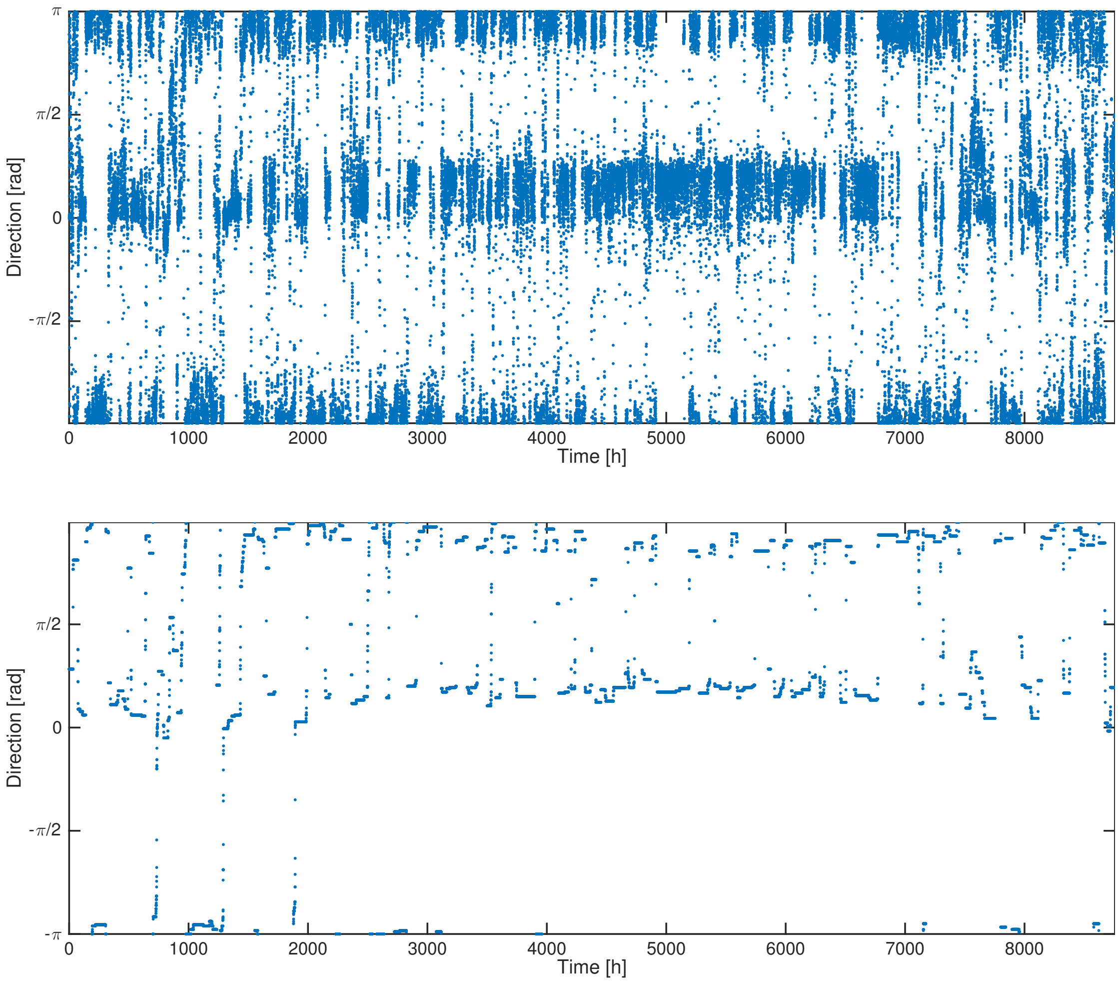

Figure 2: *Top:* Wind directions at Station WPOW1 (West Point, WA) recorded every 10 minutes in the year 2014. *Bottom:* Total variation regularization with parameter $\alpha = 50$. The data is given quantized to $K = 360$ angles. Hence, the time computation amounts to only 19.6 seconds for the signal of length $N = 52543$.

References

[1] S. Alliney. A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. *IEEE Transactions on Signal Processing*, 45(4):913–917, 1997.

[2] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[3] R. Bellman. *Dynamic Programming*. Princeton University Press, 1957.

[4] R. Bellman and R. Roth. Curve fitting by segmented straight lines. *Journal of the American Statistical Association*, 64(327):1079–1084, 1969.

[5] A. Blake and A. Zisserman. *Visual reconstruction*. MIT Press Cambridge, 1987.

[6] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.

[7] T. Chan and S. Esedoglu. Aspects of total variation regularized l^1 function approximation. *SIAM Journal on Applied Mathematics*, 65(5):1817–1837, 2005.

[8] T. Chan, S. Kang, and J. Shen. Total variation denoising and enhancement of color images based on the CB and HSV color models. *Journal of Visual Communication and Image Representation*, 12 (4):422–435, 2001.

[9] C. Clason, B. Jin, and K. Kunisch. A duality-based splitting method for ℓ^1-TV image restoration with automatic regularization parameter choice. *SIAM Journal on Scientific Computing*, 32(3): 1484–1505, 2009.

[10] L. Condat. A direct algorithm for 1-D total variation denoising. *IEEE Signal Processing Letters*, 20 (11):1054–1057, 2013.

[11] D. Cremers and E. Strekalovskiy. Total cyclic variation and generalizations. *Journal of Mathematical Imaging and Vision*, 47(3):258–277, 2013.

[12] J. Darbon and M. Sigelle. Image restoration with discrete constrained total variation part I: Fast and exact optimization. *Journal of Mathematical Imaging and Vision*, 26(3):261–276, 2006.

[13] P. Davies and A. Kovac. Local extremes, runs, strings and multiresolution. *Annals of Statistics*, 29 (1):1–65, 2001.

[14] J. Davis and R. Sampson. *Statistics and Data Analysis in Geology*. Wiley, New York, 2002.

[15] Y. Dong, M. Hintermüller, and M. Neri. An efficient primal-dual method for L^1 TV image restoration. *SIAM Journal on Imaging Sciences*, 2(4):1168–1189, 2009.

[16] L. Dümbgen and A. Kovac. Extensions of smoothing via taut strings. *Electronic Journal of Statistics*, 3:41–75, 2009.

[17] P. Felzenszwalb and R. Zabih. Dynamic programming and graph algorithms in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(4):721–740, 2011.

[18] P. Felzenszwalb and D. Huttenlocher. Distance transforms of sampled functions. Technical report, Cornell University, 2004.

[19] P. Felzenszwalb and D. Huttenlocher. Efficient belief propagation for early vision. *International Journal of Computer Vision*, 70(1):41–54, 2006.

[20] G. Forney Jr. The Viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973.

[21] H. Fu, M. Ng, M. Nikolova, and J. Barlow. Efficient minimization methods of mixed ℓ^1-ℓ^1 and ℓ^2-ℓ^1 norms for image restoration. *SIAM Journal on Scientific Computing*, 27(6):89–97, 2006.

[22] M. Giaquinta, G. Modica, and J. Souček. Variational problems for maps of bounded variation with values in S^1. *Calc. Var.*, 1(1):87–121, 1993.

[23] T. Goldstein and S. Osher. The split Bregman method for L1-regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.

[24] P. Grohs and M. Sprecher. Total variation regularization by iteratively reweighted least squares on Hadamard spaces and the sphere. Technical Report Research Report No. 2014-39, Seminar für Angewandte Mathematik, Eidgenössische Technische Hochschule, 2014.

[25] V. Kolmogorov, T. Pock, and M. Rolinek. Total variation on a tree. *Preprint arXiv:1502.07770*, 2015.

[26] J. Lellmann, E. Strekalovskiy, S. Koetter, and D. Cremers. Total variation regularization for functions with values in a manifold. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2944–2951, 2013.

[27] E. Mammen and S. van de Geer. Locally adaptive regression splines. *Annals of Statistics*, 25(1): 387–413, 1997.

[28] C. Micchelli, L. Shen, Y. Xu, and X. Zeng. Proximity algorithms for the L1/TV image denoising model. *Advances in Computational Mathematics*, 38(2):401–426, 2013.

[29] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. *SIAM Journal on Numerical Analysis*, 40(3):965–994, 2002.

[30] T. Rockafellar. *Convex analysis*. Number 28. Princeton University Press, 1970.

[31] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

[32] Y. Sowa, A. Rowe, M. Leake, T. Yakushi, M. Homma, A. Ishijima, and R. Berry. Direct observation of steps in rotation of the bacterial flagellar motor. *Nature*, 437(7060):916–919, 2005.

[33] M. Storath, A. Weinmann, and M. Unser. Jump-penalized least absolute values estimation of scalar or circle-valued signals. submitted.

[34] M. Unser and P. Tafti. *An introduction to sparse stochastic processes*. Cambridge University Press, 2014.

[35] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, 1967.

[36] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM Journal on Imaging Sciences*, 1(3):248–272, 2008.

[37] A. Weinmann, M. Storath, and L. Demaret. The L^1-Potts functional for robust jump-sparse reconstruction. *SIAM Journal on Numerical Analysis*, 53(1):644–673, 2015.

[38] A. Weinmann, L. Demaret, and M. Storath. Total variation regularization for manifold-valued data. *SIAM Journal on Imaging Sciences*, 7(4):2226–2257, 2014.

[39] A. Weinmann, L. Demaret, and M. Storath. Mumford-Shah and Potts regularization for manifold-valued data with applications to DTI and Q-ball imaging. *Preprint arXiv:1410.1699*, 2015.