Relation between total variation and persistence distance and its application in signal processing

Gerlind Plonka* and Yi Zheng†

January 9, 2014

Abstract

In this paper we establish the new notion of persistence distance for discrete signals and study its main properties. The idea of persistence distance is based on recent developments in topological persistence for assessment and simplification of topological features of data sets. Particularly, we establish a close relationship between persistence distance and discrete total variation for finite signals. This relationship allows us to propose a new adaptive denoising method based on persistence that can also be regarded as a nonlinear weighted ROF model. Numerical experiments illustrate the ability of the new persistence based denoising method to preserve significant extrema of the original signal.

Key words: discrete total variation, persistence homology, persistence pairs, persistence distance.

Mathematics Subject Classification: 41A15, 49M25, 57M50, 65D10, 65D18, 94A12

1 Introduction

Motivated by recent developments in topological persistence for assessment of the importance of features in data sets, we study the ideas of persistence homology for one-dimensional digital signals and its application in signal denoising. The notions of persistence homology and persistence pairs were introduced in [11] for measuring the topological complexity of point sets in $\mathbb{R}^3$. Persistence pairs and corresponding persistence diagrams are well suited to quantify the topological significance of data structures and to develop a formalism for topological simplification [4, 7, 8, 9, 17]. In case of one-dimensional digital signals the idea of topological persistence boils down to the problem of pairing suitable local minima and maxima of the signal. Considering the persistence pairs and the corresponding persistences not only for the signal f but also for $-f$, we propose the new notion of persistence distance of f. Transferring from f to $-f$ switches the roles of the sets of local minima and local maxima of f. A comparison of the persistence pairs found for f and for $-f$ already provides us with an important categorization tool. Persistence pairs occurring for both, f and $-f$, are less significant than those occurring only once, for f or for $-f$.

We show that the persistence distance has a lot of favorable properties. Particularly, we show that the persistence distance is very closely related to the discrete total variation of f. This relation motivates us to employ the new notion of persistence distance for signal denoising.

Let u be a finite digital signal that is corrupted with white noise, i.e., we have given the data

$$f(x_j) = u(x_j) + n(x_j), \qquad j = 0, \ldots, N$$

*University of Göttingen, Institute for Numerical and Applied Mathematics, Lotzestr. 16-18, 37083 Göttingen, Germany. Email: plonka@math.uni-goettingen.de

†University of Göttingen, Institute for Numerical and Applied Mathematics, Lotzestr. 16-18, 37083 Göttingen, Germany. Email: y.zheng@math.uni-goettingen.de

for a partition $a = x_0 < x_1 < \ldots < x_N = b$, where n has zero mean and (unknown) deviation σ. Using the celebrated discrete ROF-model, a reconstruction of u can be obtained as the minimizer of the functional

$$J(u) := \frac{\lambda}{2} \sum_{j=0}^{N} |u(x_j) - f(x_j)|^2 + \sum_{\ell=0}^{N-1} |u(x_{\ell+1}) - u(x_\ell)|,$$

where the second term denotes the discrete total variation of u. We propose now to replace this second term using the persistence distance of u that inherits the topological properties of the signal u. We will show that the obtained new functional can be also regarded as a weighted ROF-functional of the form

$$J_w(u) := \frac{\lambda}{2} \sum_{j=0}^{N} |u(x_j) - f(x_j)|^2 + \sum_{\ell=0}^{N-1} w_\ell(u)\, |u(x_{\ell+1}) - u(x_\ell)|,$$

where the weights $w_\ell(u)$ depend on local chains of persistence pairs. In particular, the weights are taken in a way such that the denoised signal obtained by minimization of $J_w(u)$ preserves the essential peaks (discontinuities) of u well and yields good denoising performance at smooth subregions of u.

Related literature. Topological persistence and its application to extract important topological features from data has been extensively studied within the last years, see e.g. [4, 7, 8, 9, 11, 17] and references therein. In [8], it has been shown that persistence diagrams of real-valued functions are stable with regard to noise, i.e., for two functions f and g with corresponding persistence diagrams $D(f)$ and $D(g)$ one finds

$$d_B(D(f),\, D(g)) \leq \|f - g\|_\infty,$$

where d_B denotes the bottleneck distance and $\|\cdot\|_\infty$ the L_∞-norm. In [7], the p-norm of the persistence diagram and its changes under diffusion of f using a convolution with a Gaussian kernel with enlarging parameter is studied.

Though, for application of persistence in signal denoising we are only aware of the results in [2], where topological denoising methods have been proposed employing a persistence-based simplification and a so-called filling-based simplification of the signal. The latter method mimics the construction of cancelation of persistence pairs, where instead of the filling level the filling volume is increased, thereby taking into account both, the distance between points in one persistence pair and the distance of corresponding function values. This last approach is slightly related to the watershed transform, a frequently used tool in image segmentation, see e.g. [3, 13] and references therein.

Regarding weighted TV-minimization for signal and image denoising we refer to [1, 10, 12, 15, 16] etc. In the continuous setting, the adaptive TV denoising approaches usually consider the minimization functionals of the form

$$\frac{1}{2} \int_\Omega |u(x) - f(x)|^2 dx + \int_\Omega \alpha(x) |\mathcal{D}u(x)| dx$$

over $u \in BV(\Omega)$, the space of functions of bounded variation, where $\int_\Omega |\mathcal{D}u(x)| dx$ denotes the total variation of u. The parameter $\alpha(x)$ is adaptively chosen depending on geometric properties of signal features [15, 16], and can be further improved using noise statistics and robust adjustment [10].

The rest of the paper is organized as follows. In Section II.A we summarize the properties of the discrete total variation. In Section II.B we introduce the new notion of persistence distance and study its main properties. Section II.C is devoted to the close relation between the discrete total variation and the persistence distance. In Section III we attempt to apply the new notion of persistence distance to signal denoising. We propose a new weighted functional, where the regularization term is based on the persistence of the signal. We show that this functional can also be regarded as a weighted ROF functional. In Subsection III.B, we present a numerical algorithm for the proposed weighted TV minimization based on persistence. Finally, we provide some numerical results to show the performance of our algorithm and give a short conclusion.

2 Discrete total variation and persistence distance

In this section, we consider the discrete setting of total variation and introduce the new notion of persistence distance. We investigate the properties of the persistence distance and particularly establish a close connection between the persistence distance and the discrete total variation.

2.1 Discrete total variation

Let $\mathbf{X}$ be a partition of the form $a = x_0 < x_1 < \cdots < x_N = b$ of the interval $[a, b]$. Further, let us consider a sequence $\mathbf{y} = \{y_j\}_{j=0}^{N}$ that corresponds to the partition $\mathbf{X}$. Then (x_j, y_j), $j = 0, \ldots, N$, uniquely define a linear spline function $f : [a, b] \to \mathbb{R}$ with $f(x_j) = y_j$. We denote the space of linear splines with respect to the partition $\mathbf{X}$ by $S_1(\mathbf{X})$. With these assumption, the discrete total variation of f (resp. $\mathbf{y}$) is defined as the absolute sum of all changes of function values, i.e.,

$$TV(f) := \sum_{j=0}^{N-1} |f(x_{j+1}) - f(x_j)| \tag{2.1}$$

resp. $TV(\mathbf{y}) := \sum_{j=0}^{N-1} |y_{j+1} - y_j|$. Let us shortly summarize some well-known properties of $TV(f) = TV(\mathbf{y})$, for a proof, we refer e.g. to [5].

Proposition 2.1 *Let $TV(\mathbf{y})$ with $\mathbf{y} = (f(x_j))_{j=0}^{N} \in \mathbb{R}^{N+1}$ be the discrete total variation of $f \in S_1(\mathbf{X})$. Then we have:*
(i) $TV(\mathbf{y})$ is nonnegative and $TV(\mathbf{y}) = 0$ if and only if $\mathbf{y} = c\mathbf{1}$ with $\mathbf{1} := (1, \ldots, 1)^T \in \mathbb{R}^{N+1}$ and $c \in \mathbb{R}$.
(ii) $TV(\mathbf{y})$ is positively homogeneous, i.e., $TV(\lambda\mathbf{y}) = \lambda TV(\mathbf{y})$ for any $\lambda \geq 0$.
(iii) $TV(\mathbf{y})$ is invariant by addition of a constant, i.e., $TV(\mathbf{y} + c\mathbf{1}) = TV(\mathbf{y})$.
(iv) $TV(\mathbf{y}) : \mathbb{R}^{N+1} \to \mathbb{R}$ is a continuous functional.
(v) $TV(\mathbf{y})$ is submodular, i.e., for any two functions $f, g \in S_1(\mathbf{X})$ with $\mathbf{y} = (f(x_j))_{j=0}^{N}$ and $\mathbf{z} = (g(x_j))_{j=0}^{N}$, we have

$$TV(\mathbf{y}) + TV(\mathbf{z}) \geq TV(\max(\mathbf{y}, \mathbf{z})) + TV(\min(\mathbf{y}, \mathbf{z})),$$

where $\max(\mathbf{y}, \mathbf{z}) := (\max\{y_j, z_j\})_{j=0}^{N}$ and $\min(\mathbf{y}, \mathbf{z}) := (\min\{y_j, z_j\})_{j=0}^{N}$.
(vi) The discrete total variation is a semi-norm, i.e., for $\mathbf{y}, \mathbf{z} \in \mathbb{R}^{N+1}$,

$$TV(\mathbf{y} + \mathbf{z}) \leq TV(\mathbf{y}) + TV(\mathbf{z}).$$

2.2 Persistence distance

The notion of persistence homology originates from algebraic topology. It is usually introduced for simplicial complexes being filtered by a scalar function. However, for the one-dimensional signal $\mathbf{y} = (f(x_j))_{j=0}^{N}$ on the partition $\mathbf{X} = \{x_0, \ldots, x_N\}$, we aim to define persistence pairs and the persistence distance without using the background on the construction of homology groups. For this purpose we first need the following definitions.

Definition 2.2 *A knot $x_l \in \mathbf{X} \setminus \{x_0, x_N\}$ is called (left-sided) local minimum knot of $\mathbf{y} = (f(x_j))_{j=0}^N$ on $\mathbf{X}$ with the local minimum value $y_l = f(x_l)$, if $y_{l-1} = f(x_{l-1}) > f(x_l)$, and if there exists a $\nu \in \mathbb{N}_0$ such that $l + \nu + 1 \leq N$ and*

$$f(x_l) = f(x_{l+1}) = \cdots = f(x_{l+\nu}) < f(x_{l+\nu+1}).$$

Analogously, a knot $x_l \in \mathbf{X} \setminus \{x_0, x_N\}$ is called (left-sided) local maximum knot of $\mathbf{y} = (f(x_j))_{j=0}^N$ on $\mathbf{X}$ with the local maximum value $y_l = f(x_l)$, if $y_{l-1} = f(x_{l-1}) < f(x_l)$, and if there exists a $\nu \in \mathbb{N}_0$ such that $l + \nu + 1 \leq N$ and

$$f(x_l) = f(x_{l+1}) = \cdots = f(x_{l+\nu}) > f(x_{l+\nu+1}).$$

The boundary knot $x_0 \in \mathbf{X}$ is called (left-sided) local minimum (resp. maximum) knot of $\mathbf{y} = (f(x_j))_{j=0}^N$ on $\mathbf{X}$ with the local minimum (resp. maximum) value $y_0 = f(x_0)$, if there exists a $\nu \in \mathbb{N}_0$ with $\nu \leq N - 1$ such that

$$f(x_0) = f(x_1) = \cdots = f(x_\nu) < f(x_{\nu+1})$$

(resp. $f(x_0) = f(x_1) = \cdots = f(x_\nu) > f(x_{\nu+1})$). The boundary knot $x_N \in \mathbf{X}$ is called local minimum (resp. maximum) knot of $\mathbf{y} = (f(x_j))_{j=0}^N$ on $\mathbf{X}$ with the local minimum (resp. maximum) value $y_N = f(x_N)$, if $f(x_{N-1}) > f(x_N)$ (resp. $f(x_{N-1}) < f(x_N)$) holds.

We now consider the subsets of $\{y_j : j = 0, \ldots, N\}$,

$$Y_m := \{y_k = f(x_k) : y_k \text{ is a local minimum value of } \mathbf{y}\},$$

$$Y^m := \{y_k = f(x_k) : y_k \text{ is a local maximum value of } \mathbf{y}\},$$

as well as the corresponding subsets of the partition $\mathbf{X}$,

$$X_m := \{x_k : f(x_k) \in Y_m\},$$

$$X^m := \{x_k : f(x_k) \in Y^m\}.$$

Further, let $x_{max} := \max\{X_m, X^m\}$ be the extremum knot with highest index occurring in the set $X_m \cup X^m$. Observe that x_{max} not coincides with x_N if $f(x_\nu) = \ldots = f(x_{N-1}) = f(x_N)$ for some $\nu < N$. For the number of elements in Y_m and Y^m we obviously have the relation

$$\#Y_m - \#Y^m \in \{-1, 0, 1\},$$

since after ordering the knots $x_k \in X_m \cup X^m$ by size, a local minimum (maximum) knot always possesses a local maximum (minimum) as its neighbor.

Definition 2.3 *The knot $x_l \in X_m$ is called global minimum knot of $\mathbf{y} = (f(x_j))_{j=0}^N$ on $\mathbf{X}$ with the global minimum value $f(x_l)$ if $x_l = \underset{x \in X_m}{\operatorname{argmin}} f(x)$. The knot $x_l \in X^m$ is called global maximum knot of $\mathbf{y} = (f(x_j))_{j=0}^N$ on $\mathbf{X}$ with the global maximum value $f(x_l)$ if $x_l = \underset{x \in X^m}{\operatorname{argmax}} f(x)$.*

If the global maximum (or minimum) knot is not uniquely determined by Definition 2.3 then we take the knot x_l with smallest index l. We now construct persistence pairs (x_k, x_l) of $\mathbf{y} = (f(x_j))_{j=0}^N$ over the partition $\mathbf{X}$ by the following algorithm.

With the above procedure, we obtain at least $\#Y^m - 2$ persistence pairs, since each local maximum knot of f (resp. $\mathbf{y}$) that is not at the boundary (i.e. not in $\{x_0, x_{max}\}$) is paired with one local minimum knot by the above algorithm. Observe that in this way also each local minimum knot being not the global minimum knot, is contained in exactly one persistence pair while the global minimum knot is not paired. A boundary knot (i.e., x_0 or x_{max}) occurs as a knot in a persistence pair if it is a local but not the global minimum knot, and it is not contained in any persistence pair if it is a local maximum knot or the global minimum knot.

Input: Y_m, Y^m, X_m, X^m for $\mathbf{y} = (f(x_k))_{k=0}^N$.

1) Let $r := \#Y^m$, $P_1 := \emptyset$ and $X_{m,0} := X_m$.
   Fix the ordered set $K_0 := \{f(x_{k_1}) \leq f(x_{k_2}) \leq \cdots \leq f(x_{k_r})\}$
   of all local maximum values in Y^m using the convention that for
   $f(x_k) = f(x_l) \in Y^m$ we take $f(x_k)$ first if $x_k < x_l$.

2) **For** $l = 1, \ldots, r$ **do**
   Consider the l-th entry $f(x_{k_l})$ in the ordered set K_0.
   If $x_{k_l} \notin \{x_0, x_{\max}\}$ then find the two spatial neighbors $\tilde{x}_1, \tilde{x}_2 \in X_{m,l-1}$ of x_{k_l}.
   Put $\tilde{x} := \operatorname{argmin}_{x \in \{\tilde{x}_1, \tilde{x}_2\}} \mid f(x_{k_l}) - f(x) \mid$, where in case of
   $\mid f(x_{k_l}) - f(\tilde{x}_1) \mid = \mid f(x_{k_l}) - f(\tilde{x}_2) \mid$ we take $\tilde{x} = \max\{\tilde{x}_1, \tilde{x}_2\}$.
   Then $(\tilde{x},\, x_{k_l})$ resp. $(x_{k_l}, \tilde{x})$ is a persistence pair of f, and we set
   $P_1 = P_1 \cup \{(\tilde{x},\, x_{k_l})\}$ and $X_{m,l} := X_{m,l-1} \setminus \{\tilde{x}\}$.
   Here we apply the convention that the knots in the persistence pairs are ordered by size, i.e. we write $(\tilde{x}, x_{k_l})$ if $\tilde{x} < x_{k_l}$ and $(x_{k_l}, \tilde{x})$ if $\tilde{x} > x_{k_l}$.

Output: P_1 containing all persistence pairs of $\mathbf{y}$ (resp. f).

Table 1: Algorithm I: Computation of persistence pairs.

Example 2.4 *Let us consider the vector $\mathbf{y} = (0, 2, 1, 3, 1, 4, -1, 0, 1)$ on the equidistant partition $X = \{x_j\}_{j=0}^N$ with $x_j = j$, $j = 0, \ldots, N$, where $N = 8$.*

According to the definition, we find the sets $Y^m = \{2, 3, 4, 1\}$, $Y_m = \{0, 1, 1, -1\}$, $X^m = \{x_1, x_3, x_5, x_8\}$, $X_m = \{x_0, x_2, x_4, x_6\}$. Algorithm 1 provides now with $K_0 = \{1, 2, 3, 4\} = \{f(x_8), f(x_1), f(x_3), f(x_5)\}$ the set of persistence pairs $P_1 = \{(x_1, x_2), (x_3, x_4), (x_0, x_5)\}$. The global minimum knot x_6 and the local maximum knot x_8 at the boundary do not occur in any persistence pair.

Remark 2.5 *In computational topology, the persistence pairs are usually visualized by barcodes [4] or by a persistence diagram, see e.g. [8, 17]. Each persistence pair (x_k, x_l) corresponds to the point $(f(x_k), f(x_l))$ in the persistence diagram, and the distance of this point to the line $y = x$, i.e. the distance $|f(x_k) - f(x_l)|$ gives us some information about the "topological relevance" of these two local extrema of f. Important features correspond to points being further away from the diagonal, i.e., to persistence pairs (x_k, x_l) with significant distances $|f(x_l) - f(x_k)|$, see Figure 1 (right).*

Now, we want to construct a second set of persistence pairs for f (resp. for $\mathbf{y}$) on $\mathbf{X}$. For that purpose, we apply Algorithm 1 also to the sequence $\{-f(x_j)\}_{j=0}^N = \{-y_j\}_{j=0}^N$, and obtain a set P_2 of persistence pairs.

Obviously, the transfer from $\{f(x_j)\}_{j=0}^N$ to $\{-f(x_j)\}_{j=0}^N$ switches the roles of the sets Y_m and Y^m (and of X_m and X^m), i.e., using the notations $Y_m(-f)$, $Y^m(-f)$, $X_m(-f)$, $X^m(-f)$ for the sets of extremal values of $\{-f(x_j)\}_{j=0}^N$ and their corresponding knots $\{x_j\}_{j=0}^N$, we have

$$f(x_j) \in Y_m \Longleftrightarrow -f(x_j) \in Y^m(-f),$$

$$f(x_j) \in Y^m \Longleftrightarrow -f(x_j) \in Y_m(-f),$$

and $X_m(-f) = X^m$, $X^m(-f) = X_m$.

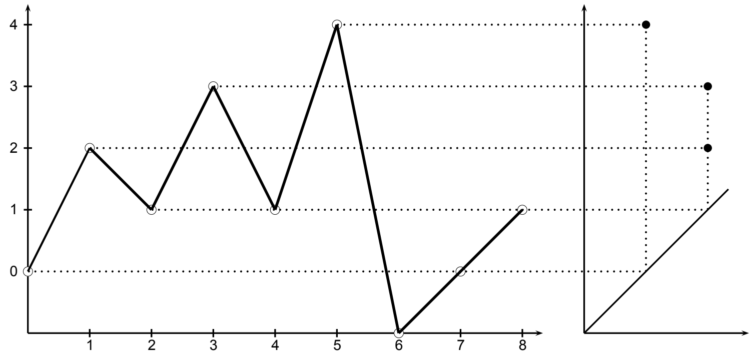


Figure 1: Spline function f in Example 2.4 (left), corresponding persistence diagram (right).

Considering again the Example 2.4, we then obtain a second set of persistence pairs

$$P_2 = \{(x_1, x_2), (x_3, x_4), (x_6, x_8)\}.$$

In particular, we observe that the global maximum knot x_5 of X^m does not occur in any persistence pair of P_2.

Note that the persistence pairs found in P_1 and P_2 partially coincide, but usually P_1 and P_2 are not equal. Further, the boundary extremum knots x_0 and x_{max} are included in at most one persistence pair, either in one from P_1 or in one from P_2, since they are not regarded when being a local maximum knot. Indeed, x_0 (resp. x_{max}) will not occur in any persistence pair, i.e., neither in P_1 nor in P_2, if it is a global extremum knot. We are now ready for the following new definition.

Definition 2.6 (Persistence distance)
For a given function $f \in S_1(\mathbf{X})$ respective the vector $\mathbf{y} = (f(x_j))_{x_j \in X}$, we define the **persistence distance** *by*

$$\|f\|_{per} = \|\mathbf{y}\|_{per} = \|\mathbf{y}|\mathbf{X}\|_{per} := \sum_{(x_k, x_l) \in P_1} |f(x_l) - f(x_k)| + \sum_{(x_k, x_l) \in P_2} |f(x_l) - f(x_k)|,$$

i.e., as the sum over all distances of function values for the persistence pairs in P_1 and P_2.

Observe that for persistence pairs that occur twice, i.e., are contained in $P_1 \cap P_2$, the corresponding absolute difference of function values is added twice.

Remark 2.7 *As far as we know, the persistence distance as given in Definition 2.6 has not been regarded before in the homology literature. The idea to consider a so-called p-norm of the persistence diagram of a function $f_t : \mathbb{R}^2 \to \mathbb{R}$ that is obtained by convolving the original function $f : \Omega \to \mathbb{R}$ with the isotropic Gaussian kernel with scale $t > 0$ (in the two-dimensional case), can be found already in [7]. This p-norm takes the p-th root of the sum of the p-th powers of all persistences. In contrast to the p-norm definition of the persistence diagram, we consider the persistence pairs for a function on a bounded interval and have to treat extremal values at the boundary with special care. Further, we consider the persistences for f and for $-f$.*

Let us derive some properties of the persistence distance $\|f\|_{per} = \|\mathbf{y}\|_{per}$.

Theorem 2.8 *Let $f \in S_1(\mathbf{X})$ be a spline function with $\mathbf{y} = (f(x_j))_{j=0}^N$ on the partition $\mathbf{X} = \{x_0, \ldots, x_N\}$ of $[a, b]$. Then the persistence distance $\|f\|_{per} = \|\mathbf{y}|\mathbf{X}\|_{per} = \|\mathbf{y}\|_{per}$ satisfies*

the following properties.
(i) $\|\mathbf{y}\|_{per} \geq 0$. *We have* $\|\mathbf{y}\|_{per} = 0$ *if and only if* $\mathbf{y} = (y_j)_{j=0}^N$ *is monotone.*
(ii) For each $c \in \mathbb{R}$, *we have* $\|c\mathbf{y}\|_{per} = |c| \cdot \|\mathbf{y}\|_{per}$.
(iii) The persistence distance is invariant under addition of a constant function,

$$\|\mathbf{y} + c\mathbf{1}\|_{per} = \|\mathbf{y}\|_{per},$$

where $\mathbf{1} = (1, \ldots, 1)^T \in \mathbb{R}^{N+1}$ *and* $c \in \mathbb{R}$. *In particular,* $\|c\mathbf{1}\|_{per} = 0$.
(iv) The persistence distance $\|\mathbf{y}\|_{per} : \mathbb{R}^{N+1} \to \mathbb{R}$ *is a continuous functional.*
(v) The persistence distance $\|\mathbf{y}\|_{per}$ *is submodular, i.e., for* $f, g \in S_1(\mathbf{X})$ *with* $\mathbf{y} = (f(x_j))_{j=0}^N$ *and* $\mathbf{z} = (g(x_j))_{j=0}^N$ *we have*

$$\|\mathbf{y}\|_{per} + \|\mathbf{z}\|_{per} \geq \|\max(\mathbf{y}, \mathbf{z})\|_{per} + \|\min(\mathbf{y}, \mathbf{z})\|_{per},$$

where $\max(\mathbf{y}, \mathbf{z}) := (\max\{y_j, z_j\})_{j=0}^N$ *and* $\min(\mathbf{y}, \mathbf{z}) := (\max\{y_j, z_j\})_{j=0}^N$.
(vi) There exist $\mathbf{y}, \mathbf{z} \in \mathbb{R}^{N+1}$ *such that the persistence distance* $\|\mathbf{y}\|_{per}$ *does not satisfy the triangle inequality, i.e.,*

$$\|\mathbf{y} + \mathbf{z}\|_{per} \leq \|\mathbf{y}\|_{per} + \|\mathbf{z}\|_{per}.$$

Hence, $\|\mathbf{y}\|_{per}$ *is not convex.*

Proof. (i) The property $\|\mathbf{y}\|_{per} \geq 0$ is obvious by definition, where $\|\mathbf{y}\|_{per} = 0$ can only occur if there are no persistence pairs, neither for f nor for $-f$, i.e., $P_1 \cup P_2 = \emptyset$. According to Algorithm 1, we have $P_1 = \emptyset$, if and only if the set Y^m is a subset of $\{f(x_0), f(x_{max})\}$, i.e., there are local maxima only at the boundary. Analogously, $P_2 = \emptyset$, if and only if $Y_m \subset \{f(x_0), f(x_{max})\}$, i.e., there are local minima only at the boundary. Hence, $P_1 \cup P_2 = \emptyset$ is true if and only if $\mathbf{y}$ is monotone.

(ii) Property (ii) is obvious, where for $c < 0$ the roles of X_m and X^m and hence of P_1 and P_2 are exchanged.

(iii) All persistence pairs and hence the persistence distance are invariant under addition of a constant.

(iv) Since f is a tame function, this assertion is a direct consequence of the stability of persistence diagrams, see e.g. [8]. In the special case considered here, we can also derive this property directly. Assume first, that the vector $\mathbf{y} = (f(x_j))_{j=0}^N$ does not contain constant parts, i.e. that $y_j \neq y_{j+1}$ for $j = 0, \ldots, N-1$. Then, there exists an $\epsilon > 0$ such that for each $\tilde{\mathbf{y}}$ with $\|\mathbf{y} - \tilde{\mathbf{y}}\|_\infty < \epsilon$ the sets of minimum and maximum knots for $\mathbf{y}$ and $\tilde{\mathbf{y}}$ coincide, i.e., $X^m = \tilde{X}^m$ and $X_m = \tilde{X}_m$, and such that the order of maximum and minimum values (i.e., the order of the values $f(x_{k_1}), \ldots, f(x_{k_r})$ in the set K_0 in Algorithm 1) does not change, and hence all persistence pairs (x_k, x_l) remain the same for $\mathbf{y}$ and $\tilde{\mathbf{y}}$. Hence

$$\begin{aligned}
|\|\mathbf{y}\|_{per} - \|\tilde{\mathbf{y}}\|_{per}| &\leq \sum_{(x_k,x_l)\in P_1} |(|y_l - y_k| - |\tilde{y}_l - \tilde{y}_k|)| \\
&\quad + \sum_{(x_k,x_l)\in P_2} |(|y_l - y_k| - |\tilde{y}_l - \tilde{y}_k|)| \\
&\leq \sum_{(x_k,x_l)\in P_1} |(y_l - \tilde{y}_l) - (\tilde{y}_k - y_k)| + \sum_{(x_k,x_l)\in P_2} |(y_l - \tilde{y}_l) - (\tilde{y}_k - y_k)| \\
&\leq \; 2N\epsilon.
\end{aligned}$$

The last inequality follows from the fact that $\#P_1 \leq \#Y^m$ and $\#P_2 \leq \#Y_m$, where Y^m resp. Y_m contain the maximum resp. minimum values of $\mathbf{y}$.

In the case of equal neighboring function values in $\mathbf{y}$, the sets $\tilde{P}_1$ and $\tilde{P}_2$ may enlarge for the perturbed vector $\tilde{\mathbf{y}}$. However, for each pair $(x_k, x_l) \in P_1 \cup P_2$ there exists a persistence pair $(x_{k'}, x_{l'}) \in \tilde{P}_1 \cup \tilde{P}_2$, with $y_k - y_{k'} = 0$, $y_l - y_{l'} = 0$ and $y_k - \tilde{y}_{k'} < \epsilon$, $y_l - \tilde{y}_{l'} < \epsilon$. Further, the new sets $\tilde{P}_1$ and $\tilde{P}_2$ of $\tilde{\mathbf{y}}$ may contain new persistence pairs, but these are due to components in $\tilde{\mathbf{y}}$ that correspond to equal neighboring values in $\mathbf{y}$ and hence have a distance of at most 2ϵ. Thus the same estimate as in the first case applies also here.

(v) The proof of submodularity is postponed to Remark 2.13.

(vi) We give a counterexample, where the triangle inequality is not satisfied. Choose $\mathbf{X} = (x_0, x_1, x_2, x_3)$ with $x_j = j$, and let $\mathbf{y}$ and $\mathbf{z}$ be determined by the vectors $\mathbf{y} = (0, 1, -1, 0)^T$ and $\mathbf{z} = (0.6, 1.2, 1.8, 2.4)^T$. For $\mathbf{y}$ we find the sets of persistence pairs $P_1 = \{(x_0, x_1)\}$, $P_2 = \{(x_2, x_3)\}$ and hence $\|\mathbf{y}\|_{per} = |y_1 - y_0| + |y_3 - y_2| = 2$. Since $\mathbf{z}$ is monotone, we find $P_1 = \emptyset$ and $P_2 = \emptyset$ and hence $\|\mathbf{z}\|_{per} = 0$. Finally, for the sum $\mathbf{y} + \mathbf{z} = (0.6, 2.2, 0.8, 2.4)^T$ we obtain $P_1 = \{(x_1, x_2)\}$, $P_2 = \{(x_1, x_2)\}$ yielding $\|\mathbf{y} + \mathbf{z}\|_{per} = 2.8$. Hence, $\mathbf{y}$ and $\mathbf{z}$ do not satisfy the triangle inequality. ∎

2.3 Relation between discrete total variation and persistence distance

While being not a semi-norm, the persistence distance (together with the sets of persistence pairs) contains a lot of information about the structure of a function $f \in S_1(\mathbf{X})$.

In this subsection, we show the following close relation to the discrete total variation $TV(f)$.

Theorem 2.9 *Let $\mathbf{X}$ be a partition of the form $a = x_0 < x_1 < \cdots x_N = b$. Then, for each function $f \in S_1(\mathbf{X})$ we have*

$$\|f\|_{per} + \max_{x,y \in \mathbf{X}} |f(x) - f(y)| = TV(f),$$

where $TV(f)$ is defined in (2.1)*. Analogously, for each sequence $\mathbf{y} \in \mathbb{R}^{N+1}$, we have*

$$\|\mathbf{y}\|_{per} + \max_{j,k \in \{0,1,\cdots,N\}} |y_j - y_k| = TV(\mathbf{y}).$$

The proof of Theorem 2.9 is based on an iterative topological simplification technique as used e.g. in [2]. Before we can prove this Theorem 2.9, we need the following Lemmata. In the first lemma we show a nesting principle for persistence pairs.

Lemma 2.10 *Let P_1 and P_2 be the two sets of persistence pairs of $f \in S_1(\mathbf{X})$. Let (x_k, x_l) be a persistence pair in $P_1 \cap P_2$. Then, for all $x \in X_m \cup X^m$ with $x_k < x < x_l$, there exists a further knot $\tilde{x} \in X_m \cup X^m$ with $x_k < \tilde{x} < x_l$ such that $(x, \tilde{x})$ or $(\tilde{x}, x)$ is also contained in $P_1 \cap P_2$.*

Proof. The above assertion is in fact a direct conclusion from Algorithm 1. Let (x_k, x_l) be a persistence pair in P_1 with $x_k < x_l$, and let us assume without loss of generality, that $x_k \in X_m$, and $x_l \in X^m$, i.e., x_k is a local minimum knot and x_l is a local maximum knot of f. Recalling Algorithm 1 it follows that in the iteration step ν, where $f(x_l) \in Y^m$ is considered, the knot x_k is a direct neighbor of x_l, i.e., there is no other minimum knot $x \in X_{m,\nu}$ left in the interval between x_k and x_l. Hence, if there exists a local maximum knot $x \in X^m$ with $x_k < x < x_l$, then it had been paired with a minimum knot contained in (x_k, x_l) and pulled out already in an earlier step of the iteration. In particular, it hence corresponds to a local maximum value smaller than (or equal to) $f(x_l)$.

Analogously, since $(x_k, x_l) \in P_2$, the arguments can be repeated for $-f(x_k) \in \tilde{Y}_m$ and $-f(x_l) \in \tilde{Y}^m$. Hence, each knot $x \in X_m$ with $x_k < x < x_l$ is paired with some $\tilde{x} \in X^m$ with $x_k < \tilde{x} < x_l$, and vice versa. Moreover, it cannot happen that one such $x \in X_m \cap (x_k, x_l)$ is paired with different knots $\tilde{x}_1 \neq \tilde{x}_2$ in P_1 and P_2, since the pairing procedure in Algoritm 1 is defined uniquely. ∎

Lemma 2.11 *Let $f \in S_1(\mathbf{X})$ with the sets P_1 and P_2 of persistence pairs, where $P_1 \cap P_2 = \emptyset$. Then, for each persistence pair $(x_k, x_l) \in P_1 \cup P_2$, the values x_k and x_l are neighbor knots in $X_m \cup X^m$.*

Proof. Assume by contrast that there is a persistence pair $(x_k, x_l) \in P_1 \cup P_2$ that does not satisfy this assertion. Without loss of generality let $x_k \in X_m$ and $x_l \in X^m$ and $(x_k, x_l) \in P_1$. Since x_k and x_l are no neighbor knots in $X_m \cup X^m$, there exist (by Lemma 2.10) $\tilde{x}_0 \in X_m$ and $\tilde{x}_1 \in X^m$ with $x_k < \tilde{x}_1 < \tilde{x}_0 < x_l$ that also form a persistence pair $(\tilde{x}_1, \tilde{x}_0)$ in P_1 and such that according to Algorithm 1 $f(\tilde{x}_0) \geq f(x_k)$ and $f(\tilde{x}_1) < f(x_l)$. Hence, for $\{-f(x_k)\}_{k=0}^N$, we find similarly as in the proof of Lemma 2.10 that $(\tilde{x}_1, \tilde{x}_0)$ is also a persistence pair in P_2, contradicting the assumption $P_1 \cap P_2 = \emptyset$. ■

Proof. (of Theorem 2.9) 1. Let $f \in S_1(\mathbf{X})$ be given with local minima and maxima sets X_m, Y_m , X^m, Y^m and sets P_1 and P_2 of persistence pairs. We order all persistence pairs $(x_k, x_l) \in P_1 \cup P_2$ by their distances $|f(x_l) - f(x_k)|$ starting with the smallest. Now we apply the following iterative simplification algorithm to $f^0 := f$. If (x_k, x_l) is the persistence pair in $P_1 \cap P_2$ of f^0 with smallest absolute difference $|f^0(x_l) - f^0(x_k)|$, we determine $f^1 \in S_1(\mathbf{X})$ by

$$f^1(x) = \begin{cases} \dfrac{f^0(x_k) + f^0(x_l)}{2}, & x \in \mathbf{X} \cap [x_k, x_l], \\ f^0(x), & x \in \mathbf{X} \setminus [x_k, x_l]. \end{cases}$$

Hence, since f^0 is monotone in $[x_k, x_l]$, we have changed the total variation by $2|f^0(x_k) - f^0(x_l)|$, i.e.,

$$TV(f^1) = TV(f^0) - 2|f^0(x_k) - f^0(x_l)|.$$

Consider now the change of persistences of f^1. Obviously, since $|f^0(x_l) - f^0(x_k)|$ was the smallest absolute difference, $f^1(x_k)$ and $f^1(x_l)$ are no longer extremal values of f^1 while all other extremal values remain the same compared to f^0. Due to Algorithm 1 and Lemma 2.10, f^1 possesses the same persistence pairs as f^0 up to (x_k, x_l), i.e.,

$$\|f^1\|_{per} = \|f^0\|_{per} - 2|f^0(x_k) - f^0(x_l)|.$$

This simplification can now be applied to f^1 removing the next persistence pair (x_k^1, x_l^1) of f^1 with smallest absolute difference $|f(x_k^1) - f(x_l^1)|$ to obtain f^2 etc. If m is the number of persistence pairs in $P_1 \cap P_2$, then, after m simplification steps we obtain $f^m \in S_1(\mathbf{X})$ with

$$\begin{aligned} \|f^m\|_{per} &= \|f^0\|_{per} - 2 \sum_{(x_k,x_l) \in P_1 \cap P_2} |f^0(x_k) - f^0(x_l)| \\ &= \sum_{(x_k,x_l) \in P_1 \setminus P_2} |f^0(x_k) - f^0(x_l)| + \sum_{(x_k,x_l) \in P_2 \setminus P_1} |f^0(x_k) - f^0(x_l)| \end{aligned}$$

and with

$$TV(f^m) = TV(f^0) - 2 \sum_{(x_k,x_l) \in P_1 \cap P_2} |f^0(x_k) - f^0(x_l)|. \tag{2.2}$$

Hence, it remains to consider the relation between $\|f\|_{per}$ and $TV(f)$ for a function $f = f^m \in S_1(\mathbf{X})$ with persistence sets P_1, P_2 satisfying $P_1 \cap P_2 = \emptyset$.

2. Assume now that we have $P_1 \cap P_2 = \emptyset$ for $f \in S_1(\mathbf{X})$. By construction of P_1 and P_2 in Algorithm 1, we know that each local extremum knot x_k that is not a global extremum and not a boundary knot (i.e., $x_0 < x_k < x_{max}$), occurs in two persistence pairs, one in P_1 and one in P_2. By Lemma 2.11, these two persistence pairs are different and connect x_k with its two spatial extremum knot neighbors. Further, each boundary knot (x_0 and x_{max}) that is not a global extremum, occurs in one persistence pair, namely in P_1 if it is a local minimum and in P_2 if it is a local maximum. This persistence pair connects the boundary knot with its spatial extremum knot neighbor. If x_k is a global extremum (maximum or minimum) knot of f and not a boundary knot, then it occurs in only one persistence pair (in P_1 if being the global maximum or in P_2 otherwise). By Lemma 2.11, this persistence pair connects x_k with one of its spatial extremum knot neighbors.

But since all other knots are already connected with their spatial extremum knot neighbors by persistence pairs, there is only one "connection" missing, namely between the global minimum knot and the global maximum knot. Hence, we obviously have

$$\|f\|_{per} = TV(f) - \max_{(x,y)\in X} |f(x) - f(y)|.$$

Finally, if x_k is a global extremum of f and a boundary knot, then it does not occur in any persistence pair. Also in this case, we can repeat the argument, observing that only the connection between the global minimum knot and global maximum knot is missing. Together with (2.2) the assertion of the theorem follows. ■

Example 2.12 *We consider* $\mathbf{X} = \{x_j\}_{j=0}^8$ *with* $x_j = j$*, and let* $f \in S_1(\mathbf{X})$ *be given by the vector* $\{f(x_j)\}_{j=0}^8 = (1, 0, 3, 2, 5, 1, 4, 0, 2)$*, see Figure 2(a). We obtain the sets*

$$\begin{aligned} X_m &= \{x_1, x_3, x_5, x_7\}, \qquad X^m = \{x_0, x_2, x_4, x_6, x_8\}, \\ P_1 &= \{(x_2, x_3), (x_5, x_6), (x_4, x_7)\}, \\ P_2 &= \{(x_2, x_3), (x_5, x_6), (x_7, x_8), (x_0, x_1)\}, \end{aligned}$$

Hence, we find the persistence distance $\|f\|_{per} = (1 + 3 + 5) + (1 + 3 + 2 + 1) = 16$*, and the discrete total variation* $TV(f) = 21$*. After simplification of* f *according to the procedure described in the proof of Theorem 2.9, where the persistence pairs* (x_2, x_3) *and* (x_5, x_6) *in* $P_1 \cap P_2$ *are removed, we obtain* f^2 *with* $\{f^2(x_j)\}_{j=0}^8 = (1, 0, 2.5, 2.5, 5, 2.5, 2.5, 0, 2)^T$*. For* f^2*, we regard*

$$\begin{aligned} X_m &= \{x_1, x_7\}, \qquad X^m = \{x_0, x_4, x_8\} \\ P_1 &= \{(x_4, x_7)\}, \qquad P_2 = \{(x_0, x_1), (x_7, x_8)\}, \end{aligned}$$

such that $\|f^2\|_{per} = 5 + 1 + 2 = 8$ *and* $TV(f^2) = 13$*.*

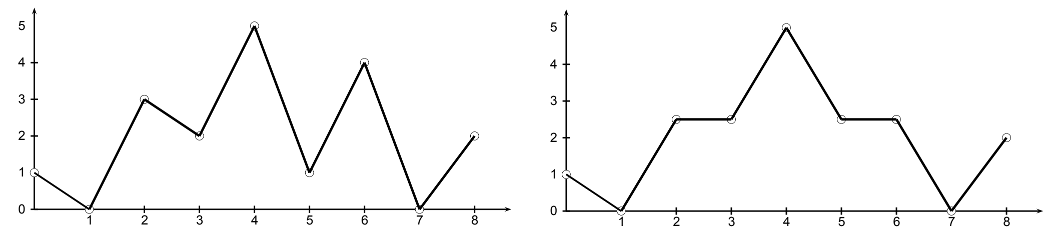


Figure 2: (a) Illustration of the spline functions f and (b) of f^2 in Example 2.12.

Remark 2.13 *The submodularity of the persistence distance stated in Theorem 2.8 follows now directly from the submodularity of the discrete total variation in Proposition 2.1. For all* $\mathbf{y}, \mathbf{z} \in \mathbb{R}^{N+1}$ *we find with* $y^m := \max\{y_j : j = 0, \ldots, N\}$*,* $y_m := \min\{y_j : j = 0, \ldots, N\}$*, and* $z^m := \max\{z_j : j = 0, \ldots, N\}$*,* $z_m := \min\{z_j : j = 0, \ldots, N\}$*,*

$$\begin{aligned} \|\mathbf{y}\|_{per} + \|\mathbf{z}\|_{per} &= TV(\mathbf{y}) + TV(\mathbf{z}) - (y^m - y_m) - (z^m - z_m) \\ &\geq TV(\max(\mathbf{y}, \mathbf{z})) + TV(\min(\mathbf{y}, \mathbf{z})) - y^m + y_m - z^m + z_m \\ &= TV(\max(\mathbf{y}, \mathbf{z})) + TV(\min(\mathbf{y}, \mathbf{z})) - (\max\{y^m, z^m\} \\ &\quad - \max\{y_m, z_m\}) - (\min\{y^m, z^m\} - \min\{y_m, z_m\}) \\ &= \|\max(\mathbf{y}, \mathbf{z})\|_{per} + \|\min(\mathbf{y}, \mathbf{z})\|_{per}. \end{aligned}$$

3 Application in signal denoising

Having found the close relationship between persistence distance and discrete total variation, we want to explore some first ideas of how this relationship can be applied to signal denoising.

One of the most famous and successful models for signal denoising is the celebrated Rudin-Osher-Fatemi model [14]. Considering the discrete setting, let $f \in S_1(\mathbf{X})$ be the noise contaminated version of a clean signal $u \in S_1(\mathbf{X})$, i.e.,

$$f(x_j) = u(x_j) + n(x_j), \qquad x_j \in \mathbf{X},$$

where $(n(x_j))_{x_j \in \mathbf{X}}$ denotes a vector of i.i.d. random variables simulating white noise, with mean value zero and variance σ^2. In order to reconstruct u, it is proposed to minimize the functional

$$J(u) := \frac{\lambda}{2} \sum_{j=0}^{N} |u(x_j) - f(x_j)|^2 + \sum_{j=0}^{N-1} |u(x_{j+1}) - u(x_j)|,$$

where the second term coincides with the discrete total variation $TV(u)$ in (2.1). The above functional $J(u)$ is strictly convex but not differentiable. The parameter $\lambda > 0$ balances the regularization term $TV(u)$ and the data fitting term, and a suitable choice of λ is crucial for the success of the method. The main advantage of the ROF model in comparison to other models involving a smoother regularization term is its ability to preserve sharp changes in the data.

From Theorem 2.9 it follows that the ROF functional can also be written as

$$\begin{aligned} J(u) &= \frac{\lambda}{2} \sum_{j=0}^{N} |u(x_j) - f(x_j)|^2 + \|u\|_{per} + \max_{x,\tilde{x} \in \mathbf{X}} |u(x) - u(\tilde{x})| \\ &= \frac{\lambda}{2} \sum_{j=0}^{N} |u(x_j) - f(x_j)|^2 + \sum_{(x,\tilde{x}) \in P_1} |u(x) - u(\tilde{x})| \\ &\quad + \sum_{(x,\tilde{x}) \in P_2} |u(x) - u(\tilde{x})| + \max_{j,k \in \{0,\ldots,N\}} |u(x_j) - u(x_k)|. \end{aligned}$$

In contrast to the total variation $TV(u)$, the persistence distance consists of a sum of distances of function values being local extrema of the function u, i.e., describing the topological properties of the function u, where small distances $|u(x) - u(\tilde{x})|$ (being related to small pairs $(x, \tilde{x})$) correspond to oscillatory behavior like noise while the large distances $|u(x) - u(\tilde{x})|$ describe the important features of the function u. Let for simplicity

$$P(u) = P = P_1 \cup P_2 \cup \{(x, \tilde{x})\}$$

be the set of all (persistence) pairs, where $(x, \tilde{x})$ denotes the pair of knots whose corresponding function values are the global minimum and the global maximum of f. Here, as before, pairs in $P_1 \cap P_2$ occur twice in $P(u)$.

3.1 Weighted ROF-model based on the persistence distance

We propose to consider the new weighted functional

$$\tilde{J}(u) = \frac{\lambda}{2} \sum_{j=0}^{N} |u(x_j) - f(x_j)|^2 + \sum_{(x_j,\tilde{x}_j) \in P(u)} \alpha_j(u) |u(x_j) - u(\tilde{x}_j)|, \tag{3.3}$$

where $\alpha_j = \alpha_j(u)$ should be large for small distances $|u(x_j) - u(\tilde{x}_j)|$ and rather small for large distances $|u(x_j) - u(\tilde{x}_j)|$. A suitable choice of weights α_j enables us to ensure that the denoised signal u obtained by minimization of $\tilde{J}(u)$ keeps the essential features of f, and in particular preserves the significant extremum values of f. From Lemma 2.10 and Lemma 2.11 it follows that there is a special structure of persistence pairs of a function $u \in S_1(\mathbf{X})$. We regard persistence pairs that occur only once, i.e., being not contained in $P_1(u) \cap P_2(u)$, as *single persistence pairs*, while persistence pairs in $P_1(u) \cap P_2(u)$ are denoted as *double persistence pairs*. Single persistence pairs have a special importance for the function structure, and the corresponding function values can be seen as *significant extremum values* of u. The pair $(\operatorname{argmin}_{x \in \mathbf{X}_m} u(x), \operatorname{argmax}_{x \in \mathbf{X}^m} u(x))$ of global minimum and maximum knots will be handled like a single persistence pair by introducing the set

$$S(u) := P(u) \setminus (P_1(u) \cap P_2(u)).$$

We observe that for each interval $[x_l, x_{l+1}]$ formed by neighboring knots in the knot set $\mathbf{X}$, there exists a unique chain of interlacing intervals corresponding to persistence pairs, such that

$$[x_l, x_{l+1}] \subseteq [x_1^l, \tilde{x}_1^l] \subset \ldots \subset [x_{r(l)}^l, \tilde{x}_{r(l)}^l],$$

where $(x_\nu^l, \tilde{x}_\nu^l) \in P_1(u) \cap P_2(u)$ for $\nu = 1, \ldots, r(l) - 1$ and $(x_{r(l)}^l, \tilde{x}_{r(l)}^l) \in S(u)$. If already $(x_l, x_{l+1}) \in S(u)$ then the chain collapses to this one interval and we have $[x_l, x_{l+1}] = [x_1^l, \tilde{x}_1^l]$, i.e., $r(l) = 1$. Analogously, for each double persistence pair $(x_j, \tilde{x}_j)$, there exists a unique chain of interlacing persistence intervals that contains $[x_j, \tilde{x}_j]$. We say that this double persistence pair has the *order* k if there are k further "persistence" intervals containing the interval $[x_j, \tilde{x}_j]$. In particular, all pairs in $S(u)$ are of order $k = 0$.

For the smoothing algorithm, we want to consider not only the local behavior of the function around the interval $[x_l, x_{l+1}]$ but also the function structure in the corresponding chain. We may have some a-priori information on the structure of the original signal regarding the number of levels in the above chains in order to judge which persistence pairs indeed represent important features. Note that our theoretical observations in Theorem 3.1 will be true for arbitrary choices of the weights $\alpha_j(u)$ in (3.3). In our numerical experiments, we choose for every (persistence) pair $(x_j, \tilde{x}_j) \in P(u)$ the weight

$$\alpha_j(u) = \frac{1}{1 + \beta |u(\tilde{x}_j) - u(x_j)|} \tag{3.4}$$

with some suitable $\beta > 0$. Thus, this weight is rather small for large distances $|u(\tilde{x}_j) - u(x_j)|$ and approximately 1 for small distances.

The proposed new functional $\tilde{J}(u)$ in (3.3) is highly nonlinear. For the numerical evaluation, we show that $\tilde{J}(u)$ can also be seen as a weighted ROF-functional.

Theorem 3.1 *Consider for each interval $[x_l, x_{l+1}]$, $l = 0, \ldots, N-1$, the corresponding complete chain of persistence intervals of u such that $[x_l, x_{l+1}] \subseteq [x_1^l, \tilde{x}_1^l] \subset \ldots \subset [x_{r(l)}^l, \tilde{x}_{r(l)}^l]$, and denote by $\alpha_\nu^l(u) = \alpha_\nu^l$, $\nu = 1, \ldots, r(l)$, the weight in the functional $\tilde{J}(u)$ in (3.3) corresponding to the persistence pair $(x_\nu^l, \tilde{x}_\nu^l)$. Then the weighted functional $\tilde{J}(u)$ in (3.3) is equivalent to the weighted ROF functional*

$$J_w(u) := \frac{\lambda}{2} \sum_{j=0}^{N} |u(x_j) - f(x_j)|^2 + \sum_{l=0}^{N-1} w_l(u) |u(x_{l+1}) - u(x_l)|, \tag{3.5}$$

where

$$w_l(u) = w_l := \sum_{\nu=1}^{r(l)} (-1)^{\nu-1} \alpha_\nu^l. \tag{3.6}$$

Proof. The following considerations can be carried out for each single pair $(x, \tilde{x})$ in $S(u)$ separately in order to compute the weights $w_l(u)$ for all $[x_l, x_{l+1}] \subseteq [x, \tilde{x}]$. Therefore we restrict ourselves to one interval $I = [x, \tilde{x}]$ with $(x, \tilde{x}) \in S(u)$.

For each $[x_l, x_{l+1}] \subset I$ we consider the corresponding chain $[x_l, x_{l+1}] \subseteq [x_1^l, \tilde{x}_1^l] \subset \ldots \subset [x_{r(l)}^l, \tilde{x}_{r(l)}^l] = [x, \tilde{x}]$ and apply the following procedure. If $r(l) = 1$, we find $w_l(u) = \alpha_{r(l)}^l = \alpha_1^l$ as the weight corresponding to $(x, \tilde{x})$ that has to be assigned to the term $|u(x_{l+1}) - u(x_l)|$. For $r(l) \geq 2$, we consider the adjacent smaller persistence interval $[x_{r(l)-1}^l, \tilde{x}_{r(l)-1}^l]$ being a subinterval of $[x_{r(l)}^l, \tilde{x}_{r(l)}^l] = [x, \tilde{x}]$. By construction of the persistence pairs it follows that either $u(x_{r(l)}^l) > u(\tilde{x}_{r(l)}^l)$ and $u(x_{r(l)-1}^l) < u(\tilde{x}_{r(l)-1}^l)$ or vice versa. Hence, since $|u(x_{r(l)}^l) - u(\tilde{x}_{r(l)}^l)| > |u(x_{r(l)-1}^l) - u(\tilde{x}_{r(l)-1}^l)|$ we obtain for $x = x_{r(l)}^l < x_{r(l)-1}^l < \tilde{x}_{r(l)-1}^l < \tilde{x}_{r(l)}^l = \tilde{x}$ that

$$
\begin{aligned}
&|u(x) - u(\tilde{x})| = |u(x_{r(l)}^l) - u(\tilde{x}_{r(l)}^l)| \\
&= |u(x_{r(l)}^l) - u(x_{r(l)-1}^l)| - |u(x_{r(l)-1}^l) - u(\tilde{x}_{r(l)-1}^l)| \\
&\quad + |u(\tilde{x}_{r(l)-1}^l) - u(\tilde{x}_{r(l)}^l)|.
\end{aligned} \tag{3.7}
$$

Multiplying $\alpha_{r(l)}^l(u)$ on both sides of (3.7) it follows that

$$
\begin{aligned}
& \alpha_{r(l)}^l(u)|u(x_{r(l)}^l) - u(\tilde{x}_{r(l)}^l)| \\
= \; & \alpha_{r(l)}^l(u)|u(x_{r(l)}^l) - u(x_{r(l)-1}^l)| - \alpha_{r(l)}^l(u)|u(x_{r(l)-1}^l) \\
& - u(\tilde{x}_{r(l)-1}^l)| + \alpha_{r(l)}^l(u)|u(\tilde{x}_{r(l)-1}^l) - u(\tilde{x}_{r(l)}^l)|
\end{aligned}
$$

thereby showing how the weighted difference $\alpha_{r(l)}^l(u)|u(x_{r(l)}^l) - u(\tilde{x}_{r(l)}^l)|$ can be rewritten for the three subintervals of $[x_{r(l)}^l, \tilde{x}_{r(l)}^l]$. Hence, the new coefficient corresponding to the term $|u(x_{r(l)-1}^l) - u(\tilde{x}_{r(l)-1}^l)|$ is now the sum of its original coefficient $\alpha_{r(l)-1}^l(u)$ and $-\alpha_{r(l)}^l(u)$. In case of $r(l) = 2$ we thus obtain the weight $w_l(u) = \alpha_{r(l)-1}^l(u) - \alpha_{r(l)}^l(u)$.

For $r(l) > 2$, the same argument can be applied to the persistence pairs $(x_{r(l)-1}^l, \tilde{x}_{r(l)-1}^l)$ and $(x_{r(l)-2}^l, \tilde{x}_{r(l)-2}^l)$, and we find the new coefficient for the term $|u(x_{r(l)-2}^l) - u(\tilde{x}_{r(l)-2}^l)|$ as the sum of $\alpha_{r(l)-2}$ and $-(-\alpha_{r(l)}(u) + \alpha_{r(l)-1}(u))$. We repeat this argument until the smallest interval $[x_1^l, \tilde{x}_1^l]$ in the chain of $[x_l, x_{l+1}]$ is reached and obtain the coefficient $w_l(u)$ of $|u(x_1^l) - u(\tilde{x}_1^l)|$ of the form

$$
w_l(u) = \sum_{\nu=1}^{r(l)} (-1)^{\nu-1} \alpha_\nu^l. \tag{3.8}
$$

This is exactly the weight that we have to assign to $|u(x_{l+1}) - u(x_l)|$. ■

3.2 Numerical algorithm

In this subsection, we propose a numerical scheme to minimize the nonlinear weighted TV-functional $J_w(u)$ in (3.5). Let $\mathbf{X}$ be an equidistant partition of the interval $[a, b]$, i.e. $x_j := a + (b-a)j/N$, $j = 0, \ldots, N$. Let $f_j := f(x_j)$ be the given noisy values and let $u_j := u(x_j)$ be the values of the wanted denoised signal for $j = 0, \ldots, N$. Starting with $\mathbf{u}^0 = (u_j^0)_{j=0}^N := (f_j)_{j=0}^N$, we iteratively apply the following data-dependent filter

$$
u_j^{k+1} = \frac{\lambda f_j + g_{1,j}(u^k)\, u_{j-1}^k + g_{2,j}(u^k)\, u_{j+1}^k}{\lambda + g_{1,j}(u^k) + g_{2,j}(u^k)} \tag{3.9}
$$

for $j = 1, \ldots, N-1$, where the filter coefficients $g_{1,j}(u^k)$ and $g_{2,j}(u^k)$ are given by

$$
\begin{aligned}
g_{1,j}(u^k) &:= \begin{cases} w_{j-1}(u^k)/|u_j^k - u_{j-1}^k| & \text{for } |u_j^k - u_{j-1}^k| \neq 0, \\ 0 & \text{else,} \end{cases} \\
g_{2,j}(u^k) &:= \begin{cases} w_j(u^k)/|u_j^k - u_{j+1}^k| & \text{for } |u_j^k - u_{j+1}^k| \neq 0, \\ 0 & \text{else.} \end{cases}
\end{aligned}
$$

For $j = 0$ and $j = N$, the filter (3.9) is simplified by assuming that $u_{-1}^k = u_{N+1}^k = 0$ and $g_{1,0}(u^k) = g_{2,N}(u^k) = 0$. Here $w_j(u^k)$, $j = 0, \ldots, N-1$, are the weights of the functional

$J_w(u)$ in (3.6). In order to stabilize the procedure and to reduce the computational costs, we will compute the weights $w_j(u^k)$ not in each iteration step but employ an outer iteration to recompute the persistence weights and an inner iteration, where we apply several filtering steps with fixed weights. The complete algorithm is given as follows.

Input: noisy vector $\mathbf{f}$, parameters λ, β.

1) Initialize $\mathbf{u}^0 = \mathbf{f}$.

2) **For** $k = 1, \ldots, n_{outer}$ **do**

  Compute the persistence weights $w_j(u^k)$ in (3.6) and (3.4) for $j = 0, \ldots, N-1$.

  Initialize $\mathbf{u}^{k,0} := \mathbf{u}^k$, i.e., $u_j^{k,0} := u_j^k$ for $j = 0, \ldots, N$.

  For $\ell = 1, \ldots, n_{inner}$ **do**

   Compute for $j = 0, \ldots, N$

   $u_j^{k,\ell+1} = \frac{\lambda f_j + g_{1,j}(u^{k,\ell})\, u_{j-1}^{k,\ell} + g_{2,j}(u^{k,\ell})\, u_{j+1}^{k,\ell}}{\lambda + g_{1,j}(u^{k,\ell}) + g_{2,j}(u^{k,\ell})}$,

   where as before $g_{1,j}(u^{k,\ell}) := w_{j-1}(u^k)/|u_j^{k,\ell} - u_{j-1}^{k,\ell}|$

   for $|u_j^{k,\ell} - u_{j-1}^{k,\ell}| \neq 0$ (and $g_{1,j}(u^{k,\ell}) := 0$ else)

   $g_{2,j}(u^{k,\ell}) := w_j(u^k)/|u_j^{k,\ell} - u_{j+1}^{k,\ell}|$ for $|u_j^{k,\ell} - u_{j+1}^{k,\ell}| \neq 0$

   (and $g_{2,j}(u^{k,\ell}) := 0$ else).

  end

  Put $\mathbf{u}^{k+1} := \mathbf{u}^{k,n_{inner}}$.

 end

Output: $\mathbf{u} = \mathbf{u}^{n_{outer}}$ approximates the minimizer of $\min\limits_u \; J_w(u)$.

Table 2: Algorithm II: Weighted TV minimization.

Observe the functional $J_w(u)$ is not necessarily convex. Within the inner iterations, where the weights in the second term of the functional are fixed, we have applied a usual gradient method that is justified by the next theorem. Of course, the filter in (3.9) for computing $u_j^{k,\ell+1}$ in Algorithm 2 can be also be replaced by slightly smoother versions as e.g. in [6].

Theorem 3.2 *Let $\mathbf{X} = \{x_j\}_{j=0}^N$ with $x_j = a + (b-a)j/N$. For given $f \in S_1(\mathbf{X})$ and $u^k \in S_1(\mathbf{X})$, let $J_w(u, u^k)$ be the weighted ROF-functional*

$$J_w(u, u^k) := \\ \frac{\lambda}{2} \sum_{j=0}^{N} |u(x_j) - f(x_j)|^2 + \sum_{l=0}^{N-1} w_l(u^k)|u(x_{l+1}) - u(x_l)| \tag{3.10}$$

with fixed weights $w_l(u^k)$. If the sequence $(u^{k,\ell})_{\ell=0}^{\infty}$ of functions obtained by the inner iteration in Algorithm 2 converges to some function u, then u satisfies

$$\frac{\partial J_w(u, u^k)}{\partial u(x_j)} = 0 \quad \textit{for} \quad j = 0, \ldots, N. \tag{3.11}$$

Proof. We first consider $j = 1, \ldots, N-1$. The limit u of the sequence $(u^{k,\ell})_{\ell=0}^{\infty}$ given by the inner iteration in Algorithm 2 satisfies

$$[\lambda + g_{1,j} + g_{2,j}]u_j = \lambda f_j + g_{1,j}u_{j-1} + g_{2,j}u_{j+1}$$

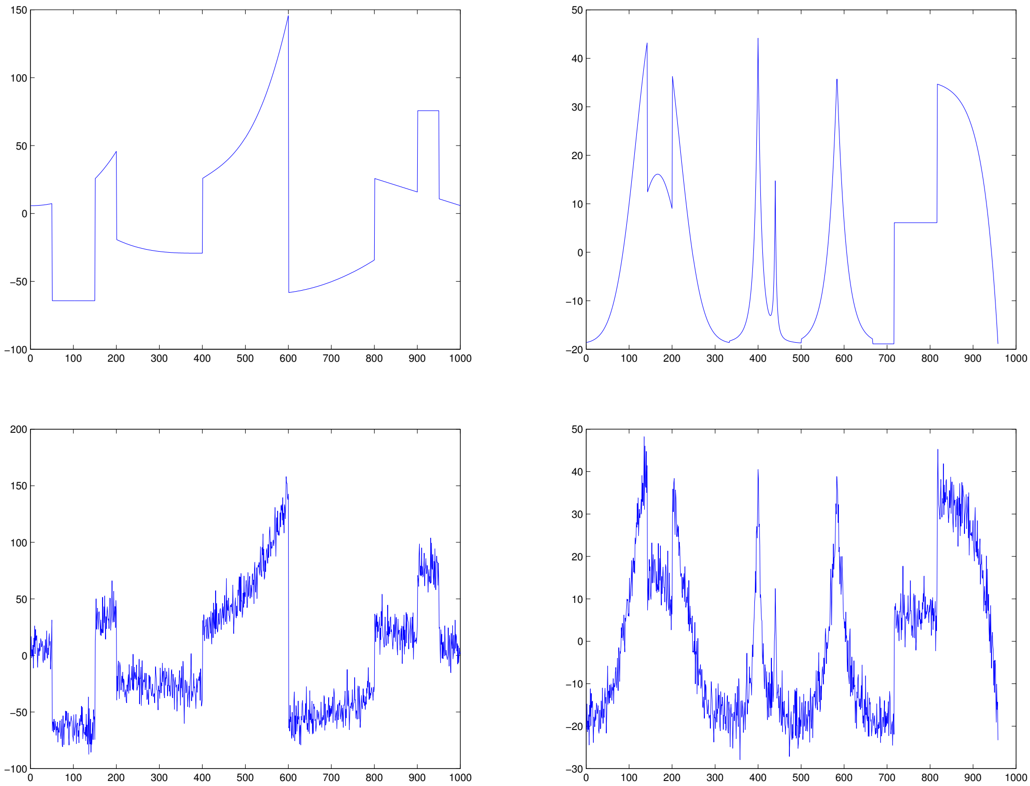


Figure 3: Top: two piecewise smooth test signals,
Bottom: noisy signals with PSNR 26.18 (left) and PSNR 24.77 (right).

with $g_{1,j} = w_{j-1}(u^k)/|u_j - u_{j-1}|$ for $|u_j - u_{j-1}| \neq 0$ (and $g_{1,j} = 0$ else) as well as $g_{2,j} = w_j(u^k)/|u_j - u_{j+1}|$ for $|u_j - u_{j+1}| \neq 0$ (and $g_{2,j} = 0$ else). Hence

$$\lambda(u_j - f_j) + g_{1,j}(u_j - u_{j-1}) + g_{2,j}(u_j - u_{j+1}) = 0.$$

Incorporating the definitions of $g_{1,j}$ and $g_{2,j}$ we find

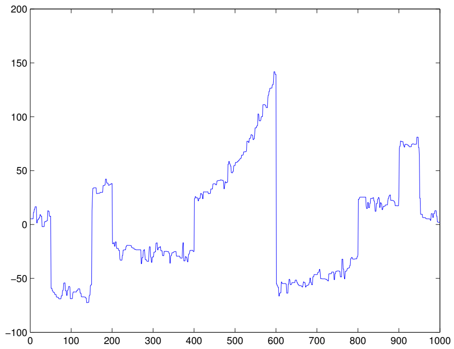
$$\begin{aligned} &\lambda(u_j - f_j) + w_{j-1}(u^k)\mathrm{sign}(u_j - u_{j-1}) \\ &+ w_j(u^k)\mathrm{sign}(u_j - u_{j+1}) = 0 \end{aligned}$$

and the assertion follows, where for $u_j - u_{j-1} = 0$ (resp. $u_j - u_{j+1} = 0$) the definition of a subdifferential has to be applied. The special cases $j = 0$ and $j = N$ are obtained analogously. ∎

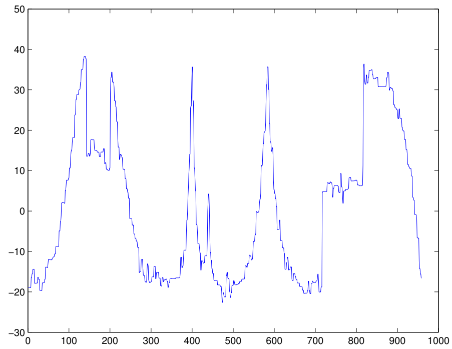
3.3 Numerical experiments

In this section, we study the performance of our proposed method which in the following is called persistence denoising method. We employ the two test signals consisting of piecewise smooth functions in Figure 3.2 taken from the WaveLab toolbox (see `http://statweb.stanford.edu/~wavelab`), where the original signals of length 1000 are shown in the first row, and the noisy signals (perturbation with white noise) are shown in the second row.

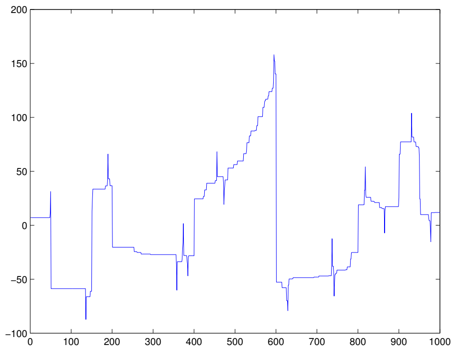
We compare the denoising performance of our method based on persistence with denoising results using the nonlinear digital TV-filter by Chan, Osher and Shen in [6] (called COS



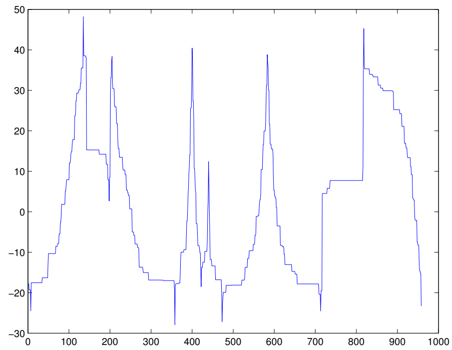
(a) COS method, PSNR 33.79



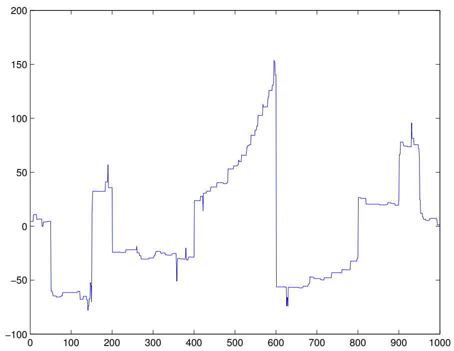
(b) COS method, PSNR 31.21



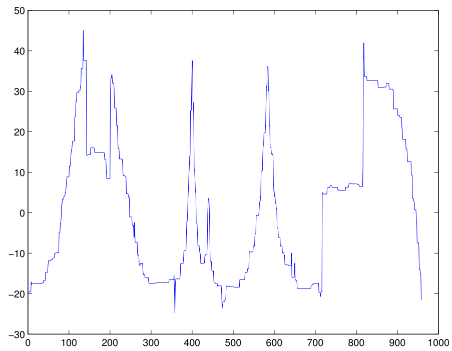
(c) persistence simplification, PSNR 32.08



(d) persistence simplification, PSNR 30.95



(e) persistence denoising, PSNR 35.56



(f) persistence denoising, PSNR 31.80

Figure 4: First column: comparison of the denoising performance of the three methods applied to the first test signal; Second column: comparison of the denoising performance of the three methods applied to the second test signal.

method), and with the denoising algorithm proposed in [2] (called persistence simplification). Figures 4(a) and (d) show the denoising result obtained by the COS model with 70000 iterations using $\lambda = 0.04$ for the first and $\lambda = 0.15$ for the second test signal. Despite its denoising performance, there remain small oscillations also in smooth parts of the signal. Taking a larger parameter λ, one obtains a stronger smoothing effect while the significant peaks of the signals will be smoothed out even further. Figures 4(b) and (e) are obtained by using the persistence simplification in [2] with thresholds 23.2 and 10.0, respectively. It performs well in some constant regions but suffers from strong oscillations at other smooth parts of the signals. This procedure keeps the significant peaks of the signal well and even overshoots, i.e. does not denoise at peak points.

The results of our method using Algorithm 2 are shown in Figures 4(c) and (f). Figure 4(c) is obtained with $\lambda = 0.0025$, $\beta = 0.8$ and using 100 outer iterations and 700 inner iterations. Figure 4(f) is found with $\lambda = 0.0042$, $\beta = 2.18$ with 100 outer and 700 inner iterations. As desired, the new algorithm keeps the significant features of the test signals very well and performs better for denoising in flat regions than the persistence simplification. Unfortunately, some isolated noisy points near jumps remain after denoising. This phenomenon is caused by significant persistence pairs (single pairs) that occur by pairing of local minima and maxima of the noisy function that are beyond the genuine local minima and maxima of the function. The denoising performance of the new algorithm can be strongly improved by application of a local median filter in a post-processing step.

For a better comparison, Figure 5 shows the results of the three methods for local regions, particularly for flat regions (see (a) and (b)) and for peak denoising (see (c) and (d)). These illustrations nicely show that our new method is able to perform better than the other two methods regarding denoising in flat regions as well as regarding correct peak preservation.

4 Conclusion

In this paper, we have proposed the new notion of persistence distance for piecewise linear functions with compact support. Note that the persistence distance can be simply generalized to continuous tame functions on the interval with a finite number of extremal values. We have shown the properties of the persistence distance and particularly found a close relationship between the persistence distance and the discrete total variation of functions on the interval. As a first attempt to apply the new topological notion of a signal, we proposed a novel signal noising scheme that is based on this relationship. The new denoising scheme can also be viewed as a weighted ROF model which exploits the nice combination of local information from discrete total variation and abundant structural information from persistence homology. The obtained functional is highly non-convex such that a convergence proof for the corresponding proposed iteration scheme cannot be achieved.

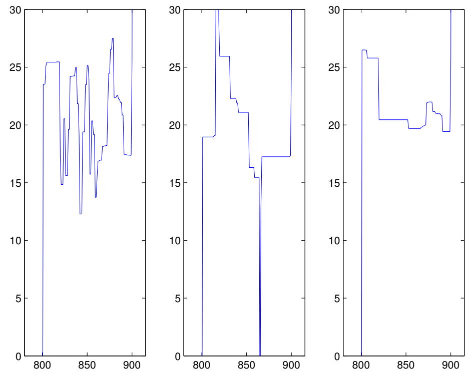
Numerical experiments show the better denoising performance of our proposed method in comparison with the nonlinear digital total variation filter from [6] and another persistence-based technique in [2]. In particular, the new algorithm preserves significant signal peaks very well and shows a good performance in smooth regions.

Acknowledgement

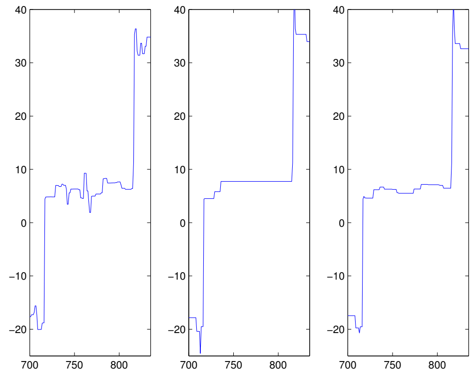
The first author is supported by the priority program SPP 1324 of the Deutsche Forschungsgemeinschaft (DFG), project PL 170/13-2. The second author is supported by the DFG as an associated member of the RTG 1023.

References

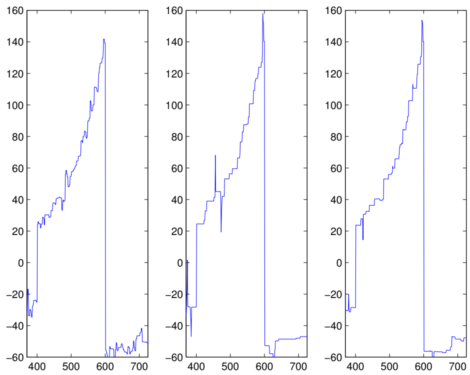
[1] A. Almansa, C. Ballester, V. Caselles, and G. Haro, A TV based restoration model with local constraints, J. Sci. Comput. **34**(3) (2008), 209–236.



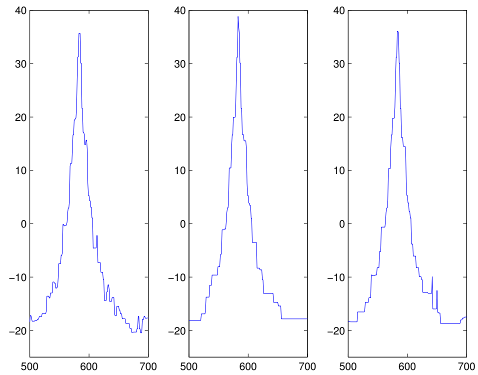
(a)



(b)



(c)



(d)

Figure 5: Comparison of the denoising performance of the three methods in flat regions and peak regions; (a) and (c) are subregions of the first test signal, (b) and (d) are subregions of the second test signal; for all subfigures the denoising performance (left to right) is given for COS denoising [6], for persistence simplification [2] and for our persistence distance denoising.

[2] U. Bauer, C.-B. Schönlieb, and M. Wardetzky, Total variation meets topological persistence: A first encounter, AIP Conf. Proc. **1281** (2010), 1022.

[3] G. Bertrand, On topological watersheds, J. Math. Imaging Vis. **22**(2-3), (2005), 217–230.

[4] G. Carlsson, A. Zomorodian, A. Collins, and L. Guibas, Persistence barcodes for shapes, Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, New York, 2004, 124–135.

[5] A. Chambolle and J. Darbon, On total variation minimization and surface evolution using parametric maximum flows, Int. J. Comput. Vis. **84** (2009), 288–307.

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

[7] C. Chen and H. Edelsbrunner, Diffusion runs low on persistence fast, IEEE Int. Conference on Computer Vision (ICCV) (2011), 423–430.

[8] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer, Stability of persistence diagrams, Discrete Comput. Geom. **37** (2007), 103–120.

[9] D. Cohen-Steiner, H. Edelsbrunner, J. Harer, and Y. Mileyko, Lipschitz functions have L_p-stable persistence, Found. Comput. Math. **10** (2010), 127–139.

[10] Y. Dong, M. Hintermüller, M. Monserrat Rincon-Camacho, Automated regularization parameter selection in multi-scale total variation models for image restoration, J. Math. Imaging Vis. **40** (2011), 82–104.

[11] H. Edelsbrunner, D. Letscher, and A. Zomorodian, Topological persistence and simplification, Discrete Comput. Geom. **28** (2002), 511–533.

[12] M. Grasmair, Locally adaptive total variation regularization, SSVM, Lecture Notes in Computer Science Volume 5567, Springer, (2009), 331–342.

[13] J. B. T. M. Roerdink and A. Meijster, The watershed transform: definitions, algorithms, and parallelization strategies, Fundam. Inform. **41** (2000), 187–228.

[14] L. Rudin, S. Osher, and E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys. D. **60** (1992), 259–268.

[15] D. M. Strong, J.-F. Aujol, and T.F. Chan, Scale recognition, regularization parameter selection, and Meyer's G norm in total variation regularization, Multiscale Model. Simul. **5**(1) (2006), 273–303.

[16] D. M. Strong, T. F. Chan, Spatially and scale adaptive total variation based regularization and anisotropic diffusion in image processing, Diffusion in Image Processing, UCLA Math Department CAM Report 96-46 (1996).

[17] A. Zomorodian and G. Carlsson, Computing persistent homology, Discrete Comput. Geom. **33**(2), (2005), 249–274., SSVM, Lecture Notes in Computer Science Volume 5567, Springer, (2009), 331–342.