PROCEEDINGS OF THE ROYAL SOCIETY A

Proc. R. Soc. A (2011) **467**, 3115–3140
doi:10.1098/rspa.2010.0674
Published online 8 June 2011

Generalized methods and solvers for noise removal from piecewise constant signals. II. New methods

By Max A. Little[1,2,*] and Nick S. Jones[1,3]

[1]*Department of Physics and Oxford Centre for Integrative Systems Biology, University of Oxford, UK*
[2]*Media Lab, Massachussetts Institute of Technology, Cambridge, MA, USA*
[3]*Department of Mathematics, Imperial College London, London SW7 2AZ, UK*

Removing noise from signals which are piecewise constant (PWC) is a challenging signal processing problem that arises in many practical scientific and engineering contexts. In the first paper (part I) of this series of two, we presented background theory building on results from the image processing community to show that the majority of these algorithms, and more proposed in the wider literature, are each associated with a special case of a generalized functional, that, when minimized, solves the PWC denoising problem. It shows how the minimizer can be obtained by a range of computational solver algorithms. In this second paper (part II), using this understanding developed in part I, we introduce several novel PWC denoising methods, which, for example, combine the global behaviour of mean shift clustering with the local smoothing of total variation diffusion, and show example solver algorithms for these new methods. Comparisons between these methods are performed on synthetic and real signals, revealing that our new methods have a useful role to play. Finally, overlaps between the generalized methods of these two papers and others such as wavelet shrinkage, hidden Markov models, and piecewise smooth filtering are touched on.

Keywords: edge; jump; shift; step; change; level

1. Introduction

Piecewise constant (PWC) signals have flat sections with a number of abrupt jumps. Often, we record a noisy PWC signal and want to recover the signal buried in the noise. This is a ubiquitous problem found across many branches of science and engineering. In the first paper of this series of two (part I), we showed that this was a challenging problem because conventional linear digital filtering methods are ineffective, due to the overlap in the frequency domain between the noise and the jumps. We explained the need to use nonlinear techniques in order to achieve effective PWC denoising, and that the nonlinearity of these techniques makes them harder to understand than linear techniques, motivating our in-depth exploration of this problem.

*Author for correspondence (max.little@physics.ox.ac.uk).

Received 23 December 2010
Accepted 12 May 2011

Table 1. 'Components' for PWC denoising methods. All the methods in this paper can be constructed using all pairwise differences between input samples, output samples and sequence indices. These differences are then used to define kernel and loss functions. Loss functions and kernels are combined to make the generalized functional to be minimized with respect to the output signal m. Function $I(s)$ is the indicator function such that $I(s)=1$ if the condition s is true, and $I(s)=0$ otherwise.

(a) difference d	description
$x_i - m_j$	input–output value difference; used in likelihood terms
$m_i - m_j$	output–output value difference; used in regularization terms
$x_i - x_j$	input–input value difference; used in both likelihood and regularization terms
$i - j$	sequence difference; used in both likelihood and regularization terms

(b) kernel function	description
1	global
$I(\lvert d\rvert \leq W)$	hard (local in either value or sequence)
$I(\lvert d\rvert^2/2 \leq W)$	
$\exp(-\beta\lvert d\rvert)$	soft (semi-local in either value or sequence)
$\exp(-\beta\lvert d\rvert^2/2)$	
$I(d=1)$	isolates only sequentially adjacent terms when used as sequence kernel
$I(d=0)$	isolates only terms that have the same index when used as sequence kernel

(c) loss function	influence function (derivative of loss function) kernel × direction	composition
$L_0(d) = \lvert d\rvert^0$		simple
$L_1(d) = \lvert d\rvert^1$	$L'_1(d) = 1 \times \mathrm{sgn}(d)$	
$L_2(d) = \lvert d\rvert^2/2$	$L'_2(d) = 1 \times d$	
$L_{W,1}(d) = \min(\lvert d\rvert, W)$	$L'_{W,1}(d) = I(\lvert d\rvert \leq W) \times \mathrm{sgn}(d)$	composite
$L_{W,2}(d) = \min(\lvert d\rvert^2/2, W)$	$L'_{W,2}(d) = I(\lvert d\rvert^2 \leq W) \times d$	
$L_{\beta,1}(d) = 1 - \exp(-\beta\lvert d\rvert)/\beta$	$L'_{\beta,1}(d) = \exp(-\beta\lvert d\rvert) \times \mathrm{sgn}(d)$	composite
$L_{\beta,2}(d) = 1 - \exp(-\beta\lvert d\rvert^2/2)/\beta$	$L'_{\beta,2}(d) = \exp(-\beta\lvert d\rvert^2/2) \times d$	

In part I, we identified some broad principles at work common to many existing PWC denoising methods. We showed that the PWC denoising problem can be understood as 0-degree spline interpolation, or level-set recovery, because typically there will be either only a few isolated jumps in the signal, or just a few distinct levels. We formalized a generalized functional equation (table 2), and showed that each of the methods introduced in part I are associated with a special case of this functional, and that this functional is assembled from a few, general 'component' functions (table 1). Finally, we demonstrated that each PWC denoising method attempts to minimize the generalized functional obtained using some kind of computational solver, many of which are special cases of a handful of quite general algorithms.

Having introduced the components, the generalized functional and solver algorithms for existing methods in part I, in this second of these two papers we investigate how some of these existing concepts can be generalized. There is more than one potential starting point for this. One approach is to ask about the range of validity of their associated solvers: what properties must the functional satisfy to allow this solver to be applied? Another approach is to attempt to synthesize new functionals that are 'hybrids' of existing methods, leading to new methods that have their own merit as PWC denoising methods.

To recap, the notation used in part I is as follows. We wish to recover an N sample, discrete-time PWC signal $m_i \in \mathbb{R}$, for $i = 1, 2, \ldots, N$, from an observed signal corrupted by an additive noise random process $e_i \in \mathbb{R}$, i.e. $x = m + e$. As discussed in part I, all the PWC denoising methods investigated in these two papers are associated with special cases of the following general functional equation:

$$H[m] = \sum_{i=1}^{N} \sum_{j=1}^{N} \Lambda(x_i - m_j, m_i - m_j, x_i - x_j, i - j). \tag{1.1}$$

Here x is the input signal of length N, and m the output of the noise removal algorithm, of length N. This functional combines difference functions into kernels and losses (tables 1 and 2). A large number of existing methods can be expressed as special cases of the resulting functional assembled from these components (table 1). Various solvers can be used to minimize this functional to obtain the output m.

In part I, the 0-degree spline and level-set models for PWC signals were introduced. The PWC signal has only a few jumps occurring between indices i and $i+1$ where $m_i \neq m_{i+1}$. The M jumps in the signal occur at the spline knots with locations $\{r_1, r_2, \ldots, r_{M+1}\}$, together with the boundary knots $r_0 = 1$ and $r_{M+1} = N + 1$. The PWC signal is reconstructed from the values of the constant levels $\{l_1, l_2, \ldots, l_{M+1}\}$ and the knot locations, e.g. $m_i = l_j$ for $r_{j-1} \leq i < r_j$, where $j = 1, 2, \ldots, M + 1$. The level-set for the value $l \in \Omega$ (Ω refers to the set of all unique values in the PWC signal) is the set of indices corresponding to l, $\Gamma(l) = \{i : m_i = l\}$. The complete level-set over all values of the PWC signal Γ is formed from the union of these level-sets, which also makes up the complete index set, $\Gamma = \bigcup_{l \in \Omega} \Gamma(l) = \{1, 2, \ldots, N\}$. The level-sets form a partition of the index set, so that $\Gamma(l_A) \cap \Gamma(l_B) = \emptyset$ for all $l_A \neq l_B$ where $l_A, l_B \in \Omega$.

In part I, the definition of a PWC signal is that the number of jumps is small compared to the number of samples, e.g. $M/N \ll 1$, or, that the number of unique levels is small compared to the number of samples $|\Omega|/N \ll 1$. Here, we say that a signal satisfying either condition has the PWC property.

The structure of this paper is as follows. Synthesizing the knowledge from part I, §2 of this paper goes on to motivate and devise new PWC denoising methods and solvers. Section 3 compares the numerical results of several PWC denoising tasks on synthetic and real signals, and discusses the accuracy of methods and efficiency of different solvers, drawing implications for the choice of methods, solvers and parameter values. Finally, §4 summarizes the findings of the paper and connects to other approaches, including wavelets, hidden Markov models (HMMs), piecewise smooth (PWS) filters and nonlinear diffusion partial differential equations (PDEs), and mentions possible directions for future research.

Table 2. A generalized functional for noise removal from PWC signals. The functional combines differences, losses and kernel functions described in table 1 into a function to be minimized over all samples, pairwise.

generalized functional for piecewise constant noise removal

$$H[m] = \sum_{i=1}^{N}\sum_{j=1}^{N} \Lambda(x_i - m_j, m_i - m_j, x_i - x_j, i - j)$$

existing methods	function Λ	notes
linear diffusion	$(1/2)\lvert m_i - m_j\rvert^2 I(i-j=1)$	solved by weighted mean filtering; cannot produce PWC solutions; not PWC
step-fitting (Gill 1970; Kerssemakers *et al.* 2006)	$(1/2)\lvert x_i - m_j\rvert^2 I(i-j=0)$	termination criteria based on number of jumps; PWC
objective step-fitting (Kalafut & Visscher 2008)	$(1/2)\lvert x_i - m_j\rvert^2 I(i-j=0) + \lambda\lvert m_i - m_j\rvert^0 I(i-j=1)$	likelihood term the same up to log transformation; regularization parameter fixed by data; PWC
total variation regularization (Rudin *et al.* 1992)	$(1/2)\lvert x_i - m_j\rvert^2 I(i-j=0) + \gamma\lvert m_i - m_j\rvert I(i-j=1)$	convex; fused Lasso signal approximator is the same; PWC
total variation diffusion	$\lvert m_i - m_j\rvert I(i-j=1)$	convex; partially minimized by iterated 3-point median filter; PWC
mean shift clustering	$\min((1/2)\lvert m_i - m_j\rvert^2, W)$	non-convex; PWC
likelihood mean shift clustering	$\min((1/2)\lvert x_i - m_j\rvert^2, W)$	non-convex; K-means is similar but not a direct special case (see text); PWC
soft mean shift clustering	$1 - \exp(-\beta\lvert m_i - m_j\rvert^2/2)/\beta$	non-convex; PWC
soft likelihood mean shift clustering	$1 - \exp(-\beta\lvert x_i - m_j\rvert^2/2)/\beta$	non-convex; soft-K-means is similar but not a direct special case (see text); PWC
convex clustering shrinkage (Pelckmans *et al.* 2005)	$(1/2)\lvert x_i - m_j\rvert^2 I(i-j=0) + \gamma\lvert m_i - m_j\rvert$	convex; PWC
bilateral filter (Mrazek *et al.* 2006)	$[1 - \exp(-\beta\lvert m_i - m_j\rvert^2/2)/\beta] I(\lvert i-j\rvert \le W)$	non-convex
new methods proposed in this paper		
jump penalization	$(1/2)\lvert x_i - m_j\rvert^2 I(i-j=0) + \gamma\lvert m_i - m_j\rvert^0 I(i-j=1)$	non-convex; PWC
robust jump penalization	$\lvert x_i - m_j\rvert I(i-j=0) + \gamma\lvert m_i - m_j\rvert^0 I(i-j=1)$	non-convex; PWC
robust total variation regularization	$\lvert x_i - m_j\rvert I(i-j=0) + \gamma\lvert m_i - m_j\rvert I(i-j=1)$	convex; PWC
soft mean shift total variation diffusion	$1 - \exp(-\beta\lvert x_i - m_j\rvert^2/2)/\beta + \gamma\lvert m_i - m_j\rvert I(i-j=1)$	non-convex; PWC
weighted convex clustering shrinkage	$(1/2)\lvert x_i - m_j\rvert^2 I(i-j=0) + \gamma\lvert m_i - m_j\rvert I(\lvert x_i - x_j\rvert \le W)$	convex; PWC
convex mean shift clustering	$\lvert m_i - m_j\rvert \exp(-\beta\lvert x_i - x_j\rvert)$	convex; PWC

2. New methods and solvers for PWC denoising

We will start in §2.1, by seeing how the very simplest stepwise jump placement solvers introduced in part I can be generalized. We then discuss the connection between total variation regularization introduced in part I and *regression splines*, and in doing so motivate a novel *coordinate descent* method (§2.2). By considering a generalization of total variation regularization, we will give a novel convex method that can handle statistical *outliers* in the noise, and can be solved using off-the-shelf linear programming algorithms in §2.3. Next, in §2.4, in addressing an important limitation of convex clustering shrinkage (see part I), we will motivate a weighting trick that not only improves the usefulness of convex clustering shrinkage, but also leads to a novel version of mean shift clustering (defined in part I) that provides a new clustering method and associated solver algorithm. Finally, in §2.5, by exposing some of the limitations of total variation diffusion and mean shift clustering, we develop a hybrid method with improved performance, and derive a new solver algorithm for it.

(*a*) *Jump penalization and robust jump penalization*

Stepwise jump placement methods (see part I, §4.1) can ensure that the solutions have the PWC property, which makes it interesting to ask whether the idea can be generalized. The conceptual simplicity of the stepwise jump placement solver algorithm is compromised if the regularization term depends on the knot locations. This happens in the case of total variation regularization, where the regularization term involves the absolute value of adjacent differences. It also occurs where minimizing the likelihood term given the fixed knot configuration is not straightforward or requires considerable computational effort. Thus, the greatest appeal of stepwise jump placement algorithms is as a minimizer for functionals that combine the non-zero count regularization term with adjacent sequence kernel, $|m_i - m_j|^0 I(i-j=1)$, but, more generally, likelihood terms such as $(1/p)|x_i - m_j|^p I(i-j=0)$, where $p \geq 1$. We can therefore suggest novel *jump penalization* methods:

$$\Lambda = (1/p)\,|x_i - m_j|^p I(i-j=0) + \gamma |m_i - m_j|^0 I(i-j=1) \tag{2.1}$$

for $p \geq 1$ and freely chosen regularization parameter $\gamma \geq 0$. For $p=2$, the mean formula $l_j = (r_j - r_{j-1})^{-1} \sum_{i=r_{j-1}}^{(r_j)-1} x_i$ applies when calculating the levels of the spline fit, whereas for $p=1$ the median formula is required to calculate the levels instead:

$$l_j = \text{median}(x_{r_{j-1}}, x_{r_{j-1}+1}, \ldots, x_{(r_j)-1}) \tag{2.2}$$

(recall that r_j is the time index of the jth knot of the spline). From a statistical point of view, this jump penalization method with $p=1$ is valuable where the noise distribution is symmetric and heavy-tailed, because in this situation the mean will be heavily influenced by outliers, but the median is robust to these large deviations. The functional is non-convex and non-differentiable, and thus not amenable to methods such as linear or quadratic programming (as discussed in part I, §4.2), and will pose non-convergence challenges for numerical methods (see part I, §4.5) for the associated initial value problem. However, the greedy

search used in stepwise jump placement requires reconstructing the spline fit for each putative new jump location and this is not necessarily computationally efficient.

In the relevant literature (Gill 1970; Kerssemakers *et al.* 2006; Kalafut & Visscher 2008), we have only encountered the idea that stepwise jump placement proceeds with introducing new knots until a termination criteria is reached. However, this stepwise jump placement strategy has the disadvantage that the minimizer that leads to the smallest possible value of the functional might only be achievable by stepwise *removal* of jumps. Therefore, it may be necessary to place a jump at every location, and perform iterative *jump removal* to attempt to lower the functional. Similarly, because the non-zero count loss is non-convex, the functional is not convex either, and there may be another solution that lowers the functional further. In fact, minimizing the functional is a combinatorial optimization problem, because the number of knots is an integer quantity. Therefore, it can be addressed by the wide array of techniques that have been developed for such problems (Papadimitriou & Steiglitz 1998).

The jump penalization methods introduced above have another useful interpretation where the PWC signal represents a discrete-time stochastic process that can have both positive and negative jumps of any height. The count number of a *Poisson process* is an important special case of this where the jumps are all of the same height and positive only, and the time interval between jumps is exponentially distributed. In that case, the probability of obtaining a jump in any one discrete-time sampling interval is just $\rho = \tau/\mu$, where τ is the sampling interval and μ is the mean time between jumps. In the corresponding discrete-time setting, the number of jumps is a random variable that is Bernoulli distributed with parameter ρ. Then the appropriate choice of regularization parameter is $\gamma = \log((1-\rho)/\rho)$. At one extreme, when $\rho = \frac{1}{2}$, that is, a jump is exactly as likely as no jump in any one sampling interval, this factor is zero, so the number of jumps plays no role in the minimizer of the functional, which is just the input signal x. At the other extreme, when $\rho \to 0$, the mean time between jumps becomes infinite, so a jump in any interval becomes improbable, and $\gamma \to \infty$. This forces the number of jumps to zero when minimizing the functional.

(*b*) *Regression splines and coordinate descent*

In this section, we demonstrate the intimate connection between *total variation regularization* (table 2 and part I, §3.3), which is of major importance in PWC denoising applications and *spline regression*, and how a simple new solver can be applied to find the solution. For the special case of total variation regularization, for which $\Lambda = (\frac{1}{2})|x_i - m_j|^2 I(i-j=0) + \gamma|m_i - m_j|I(i-j=1)$, the functional becomes:

$$H[m] = \tfrac{1}{2}\sum_{i=1}^{N}(m_i - x_j)^2 + \gamma\|Dm\|_1, \tag{2.3}$$

where $\|.\|_1$ is the (entrywise) vector 1-norm, and D is the $N \times N$ first difference matrix with $+1$ on the main diagonal, and -1 on the diagonal above it. This is shown to be equivalent to the following functional (Kim *et al.* 2009):

$$H[\mu] = \tfrac{1}{2}\|S\mu - x\|_2^2 + \gamma\|\mu\|_1, \tag{2.4}$$

where $\|.\|_p$ is the (entrywise) vector p-norm, to be minimized over the new variables (these new variables are spline coefficients related to the original variables m, see below). The $N \times N$ matrix $S = (D^{-1})^T$ has the form:

$$S = \begin{bmatrix} 1 & & & & \\ 1 & 1 & & & \\ 1 & 1 & 1 & & \\ \vdots & \vdots & \vdots & \ddots & \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix}, \tag{2.5}$$

which contains a discrete, 0-degree (constant) spline in each row, with a knot placed at positions $1, 2, \dots, N$ respectively. This demonstrates that total variation denoising is also a *Lasso regression* problem using a set of constant splines as basis functions, and the aim is to produce a *sparse approximation* with as few non-zero knot coefficients as possible (Steidl *et al.* 2006; Kim *et al.* 2009).

The general Lasso regression problem has been studied extensively in the statistics and machine-learning literature, and there are a large number of solvers that can be used to find the only minimum of the functional above. These include *subgradient* techniques such as *Gauss–Seidel* and *grafting* (Schmidt *et al.* 2007), but also methods that use a smoothed approximation to the 1-norm including *EpsL1*, *log-barrier*, *SmoothL1*, and *expectation–maximization* (EM) (Schmidt *et al.* 2007).

Reformulation as a constrained least-squares problem leads to *interior-point*, *sequential quadratic programming* and variants (Schmidt *et al.* 2007). However, computational savings might be made by exploiting the special structure of this total variation regularization problem.

Minimizing the generalized functional with respect to variation in one of the variables alone (when the others are held fixed) can sometimes be conducted analytically, or is simple to compute approximately. This observation has led to a number of very simple *coordinate descent* solvers for regularization problems (Friedman *et al.* 2007; Schmidt *et al.* 2007). It has been shown that such coordinate descent solvers are minimizers for functionals of the form:

$$H[m] = F[m, x] + \sum_{i=1}^{N} G_i(m_i), \tag{2.6}$$

where the likelihood functional term on the left is convex and differentiable, and the regularization functions G_i are convex. The regularization term displayed here is *separable*: but the functionals in this paper do not have separable regularization terms. Special adaptations are therefore required in order to apply coordinate descent to the total variation regularization problem, for example see Friedman *et al.* (2007). This involves identifying the conditions under which groups of variables need to be merged and varied together. However, we make the observation here that the Lasso spline regression problem obtained from the total variation regularization method *is* separable, and that the spline regression matrix above has a particularly simple form. This allows us to develop a simple coordinate descent solver for total variation regularization that avoids the complexity of detecting and grouping variables altogether.

In particular, note that the original variables are obtained using $m = S\mu$ where are the spline coefficients, so that each element is just the cumulative sum of the spline coefficients:

$$m_i = \sum_{j=1}^{i} \mu_j. \tag{2.7}$$

Similarly, going the other way, the spline coefficients can be obtained from the original variables using successive differences:

$$\mu_i = m_i - m_{i-1} \tag{2.8}$$

with $\mu_1 = m_1$. Also, note that at $\gamma = 0$, the original variables are equal to the input signal x, therefore the descent algorithm can be usefully initialized with the successive differences of the input signal. It is useful to understand this descent algorithm as a two-step process, (1) an *update step*:

$$w = \mu^k + S^{*T}(x - S^*\mu^k) \tag{2.9}$$

followed by (2) the *shrinkage step*:

$$\mu_1^{k+1} = \mathrm{sgn}(w_i) \max\left(\frac{|w_i| - \gamma}{\|S\|}, 0\right) \tag{2.10}$$

with initial conditions $\mu_i^0 = (x_i - x_{i-1})\|S\|$, and $\mu_1^0 = x_1\|S\|$, and $S^* = S/\|S\|$, where $\|S\| = \sqrt{N(N+1)/2}$. Normalization of the spline matrix is required to prevent iterates from diverging. These steps (1) and (2) are repeated until convergence. The original variables at convergence can be recovered using $m_i = 1/\|S\| \sum_{j=1}^{i} \mu_j$. We can understand equation (2.9) followed by equation (2.10) as the regression coefficient obtained by regressing the error $x - S^*\mu^k$ in equation (2.9) onto the ith variable μ_i^k (Friedman *et al.* 2007). The shrinkage term (2.10) is just the solution to the absolute penalized least-squares regression (2.4), if we fix all the variables except the variable i.

Using the observations about the matrix S above, the update step can be simplified considerably:

$$w_i = \mu_i^k + \frac{1}{\|S\|}\sum_{j=1}^{N}\left(x_j - \frac{1}{\|S\|}\sum_{l=1}^{j}\mu_l^k\right) = \mu_i^k + \frac{1}{\|S\|}\sum_{j=i}^{N} x_j - \frac{1}{\|S\|^2}\sum_{j=i}^{N}\sum_{l=1}^{j}\mu_l^k. \tag{2.11}$$

The expanded form of the expression on the right shows that the term in x can be pre-computed, which can lead to further computational savings. Although simple, this coordinate descent algorithm requires a large number of iterations to reach convergence, particularly for small γ, because on each iteration, the variable μ_i does not change very much. Therefore, the speed of convergence is partly dependent upon the size of γ. Furthermore, the iterates before reaching convergence do not represent the solution at smaller values of γ, because γ is fixed during the iteration. Thus, iteration of this algorithm does not obtain the *regularization path* automatically, as it does for the *piecewise linear regularization path follower* for the same problem (Hofling 2009). For some applications,

where we want the whole set of solutions when varying $\gamma \geq 0$, this could be a drawback. We note here that a related approach to PWC denoising was proposed independently in the geosciences literature (Mehta *et al.* 1990).

(*c*) *Robust total variation regularization and linear programming*

Total variation regularization (see table 2 and part I, §3.3) is a useful technique if the noise distribution is Gaussian. If the noise is not Gaussian, or there are outliers in the noise, then we can adapt the technique to increase its robustness by replacing the square likelihood loss with the absolute loss instead. The robust total variation functional becomes

$$\mathcal{A} = |x_i - m_j| I(i - j = 0) + \gamma |m_i - m_j| I(i - j = 1), \tag{2.12}$$

which can be cast as a *least absolute regression* problem:

$$m = \underset{m}{\operatorname{argmin}}\, H[m] = \underset{m}{\operatorname{argmin}} \left\| \begin{bmatrix} x \\ 0_N \end{bmatrix} - \begin{bmatrix} I_N \\ -\gamma \tilde{D} \end{bmatrix} m \right\|_1, \tag{2.13}$$

where I_N is the $N \times N$ identity matrix, 0_N the $N \times 1$ zero matrix, $\|\cdot\|_1$ the vector 1-norm, and $\tilde{D}$ is the $N \times N$ first difference matrix (see §2.2), but with the last row all zero. This is in the form of a linear program (a linear problem with linear inequality constraints), which is solvable using, for example, *simplex* or interior-point methods (Boyd & Vandenberghe 2004; Koenker 2005). To our knowledge though, specialized fast or regularization path-following methods (see part I, §4.7) for this robust total variation regularization problem do not exist, as they do for non-robust total variation regularization (but see Koenker *et al.* 1994 for related ideas and Darbon & Sigelle 2006 for an approach in the case where the signals are integer rather than real, and also references therein).

(*d*) *Weighted convex clustering shrinkage*

Convex clustering shrinkage (see table 2 and part I, §3.3) has an advantage over mean shift and other clustering methods (see part I, §§3.3 and 4.6), in that the functional is convex, so there exists a unique solution that minimizes the functional and it can be found by fast quadratic programming algorithms such as the interior point technique. However, the method can be highly sensitive to the choice of regularization parameter γ: there is typically only a small range over which the solution transitions from every sample belonging to its own cluster, to the emergence of a single cluster for all samples. To reduce this sensitivity and expand the useful range of the regularization parameter, a simple proposal is to focus the clustering only on those samples in x that are initially close to each other. Samples that are far apart initially cannot therefore become clustered together. This leads to the following adaptation to the convex clustering shrinkage functional:

$$\mathcal{A} = \left(\tfrac{1}{2}\right) |x_i - m_j|^z I(i - j = 0) + \gamma |m_i - m_j| I(|x_i - x_j| \leq W). \tag{2.14}$$

This adaptation relaxes the sensitivity to γ by ensuring that, with an appropriate choice of W, the emergence of a single cluster for all samples cannot be reached. This method retains the convexity properties of the original, because the weights are based on the input signal which is fixed. It is therefore amenable to quadratic programming. The parameter W controls the extent of the value kernel, that is,

how close the input samples need to be subject to sample distance reduction. As before, a small regularization parameter constrains the solution to be similar to the input signal.

(*e*) *Convex mean shift clustering*

The use of input-signal dependent weights for enhancing the usefulness of a PWC method presented above is a trick that can be applied more widely. For example, mean shift clustering (table 2) is not convex, but it is possible to produce a simple adaptation that *is* convex:

$$\Lambda = |m_i - m_j| I(|x_i - x_j| \leq W) \tag{2.15}$$

for which the associated influence function is $I(|x_i - x_j| \leq W)\mathrm{sgn}(m_i - m_j)$. This should be contrasted with the influence function for mean shift clustering with absolute (rather than square) loss which is $I(|m_i - m_j| \leq W)\mathrm{sgn}(m_i - m_j)$. To see why this new method can be considered a convex version of mean shift clustering, consider that a solver for the descent ordinary differential equations (ODEs) for this method (see part I, §§4 and 4.5) would be initialized with $m^0 = x$, such that, the influence function for the first iteration of this solver is $I(|m_i - m_j| \leq W)\mathrm{sgn}(m_i - m_j)$, and this coincides exactly with the influence function for (absolute) mean shift (table 1). The adaptive Euler solvers (see part I, §4.6) for the absolute mean shift and convex mean shift are, respectively:

$$m_i^{k+1} = m_i^k - \left(\sum_{j=1}^{N} I\left(|m_i^k - m_j^k| \leq W\right) \right)^{-1} \sum_{j=1}^{N} I\left(|m_i^k - m_j^k| \leq W\right) \mathrm{sgn}\left(m_i^k - m_j^k\right), \tag{2.16}$$

$$m_i^{k+1} = m_i^k - \left(\sum_{j=1}^{N} I\left(|x_i^k - x_j^k| \leq w\right) \right)^{-1} \sum_{j=1}^{N} I\left(|x_i^k - x_j^k| \leq W\right) \mathrm{sgn}\left(m_i^k - m_j^k\right) \tag{2.17}$$

(Note: with the square loss in classical mean shift in equation (2.16), the adaptive solver simplifies to the iterated mean, as shown earlier.) One way of understanding the relationship to conventional mean shift is that the value kernel for convex mean shift does not change during iterations, whereas for mean shift the kernel weights are re-computed on each iteration.

(*f*) *Soft mean shift total variation diffusion and predictor–corrector integration*

We have seen in part I (§4.6), that clustering methods have the PWC property in terms of level-sets, and total variation regularization in terms of splines. These different methods have certain disadvantages. The level-set representation is described in terms of levels, and this determines the locations of the jumps. A consequence of this is that rapid changes in the mean of the noise can cause rapid, spurious transitions between levels. On the other hand, the spline representation sets the location of the jumps, which in turn determines the constant levels. Therefore, the spline model is vulnerable to gradual, systematic changes in the level of constant regions due to changes in the mean of the noise,

for example. Clustering methods such as mean shift provide constraints on the levels of constant regions and these could be used to alleviate the weaknesses of total variation algorithms, by contrast, the temporal constraints built into total variation algorithms could help prevent spurious transitions of clustering methods that are insensitive to temporal sequence.

Here we show that it is possible to synthesize the two representations using a novel PWC method that combines the global behaviour of mean shift clustering with the sequentially local behaviour of total variation regularization, using the following functional:

$$\Lambda = 1 - \exp\left(-\beta|x_i - m_j|^2/2\right)/\beta + \gamma|m_i - m_j|I(i-j=1). \tag{2.18}$$

Here, β is a kernel parameter that determines the effective 'precision' of the mean shift: if β is large, then the solution can differentiate small peaks in the amplitude distribution, if small, then only large peaks are detected. Because of the form of equation (2.18), we call this method *soft mean shift total variation diffusion*. The regularization parameter determines the relative influence of the total variation regularization term: if small, then locally sequential runs of close values have little influence over the solution; if large, then modes in the amplitude distribution can be broken up in order to find sequential constant runs instead.

Although not necessarily the best or most efficient solver, for the purposes of illustration, we invoke concepts from part I, §4.6 and propose a two-step, *midpoint predictor–corrector* integrator for the resulting descent ODEs (Iserles 2009):

$$m_i^* = m_i^k - \frac{\Delta\eta}{2}\sum_{j=1}^{N} F'(m_i^k - x_j)k_1(i-j) - \gamma\frac{\Delta\eta}{2}\sum_{j=1}^{K} G'(m_i^k - m_j^k)k_2(i-j), \tag{2.19}$$

$$m_i^{k+1} = m_i^k - \Delta\eta\sum_{j=1}^{N} F'(m_i^* - x_j)k_1(i-j) - \gamma\Delta\eta\sum_{j=1}^{K} G'(m_i^* - m_j^*)k_2(i-j) \tag{2.20}$$

with initial condition $m^0 = x$. Using this integrator, we obtain the following solver for this new PWC denoising algorithm:

$$\begin{aligned} m_i^* = m_i^k &- \frac{\Delta\eta}{2}\sum_{j=1}^{N}\exp\left(-\beta(m_i^k - x_j)^2/2\right)(m_i^k - x_j) \\ &- \gamma\frac{\Delta n}{2}\left[\operatorname{sgn}(m_i^k - m_{i+1}^k) - \operatorname{sgn}(m_i^k - m_{i-1}^k)\right], \end{aligned} \tag{2.21}$$

$$\begin{aligned} m_i^{k+1} = m_i^k &- \Delta\eta\sum_{j=1}^{N}\exp\left(-\beta(m_i^* - x_j)^2/2\right)(m_i^* - x_j) \\ &- \gamma\Delta\eta\left[\operatorname{sgn}(m_i^* - m_{i+1}^*) - \operatorname{sgn}(m_i^* - m_{i+1}^*)\right]. \end{aligned} \tag{2.22}$$

At the boundaries, we have $m_i, m_i^* \equiv 0$ for $i < 1$ and $i > N$ for the total variation part of the expression above. Although the regularization term is not

differentiable everywhere, this finite difference solver is reasonably stable for small $\Delta\eta$, and experience shows that convergence to a useful, approximate solution is possible within a few hundred iterations.

3. Numerical results and discussion

In this section, we discuss the results of applying the existing and new methods and solvers of this paper to typical PWC denoising problems. First, we focus on qualitative comparisons; subsequently, we report quantitative performance analysis and analysis of real signals. We use these comparisons to motivate some general observations about method, solver and parameter value choices in practical settings.

(*a*) *Qualitative comparisons: performance under outliers and drift*

We first tested the ability of the methods to recover the step while ignoring two isolated 'outliers' that could be incorrectly identified as level transitions, the results are shown in figure 1. Up until now, we have assumed that the noise is statistically independent, but in practice, it may have some kind of correlation. We, therefore, devised another challenging test: recover a unit step signal with linear drift in the mean of the noise as a confounding factor, see figure 2. In each case, method parameters were optimized to achieve the output that is closest (under the root mean square error (RMSE)) to the known step signal, by searching over a grid of parameter values.

In the case of outliers, the new jump penalization and mean shift total variation diffusion methods (figure 1*k*, *l*, *j*) appear to produce the most accurate results. Mean shift clustering and bilateral filtering are able to recover the step (figure 1*d*, *h*), but are unable to ignore the outliers. K-means can ignore the outliers (figure 1*f*), but exhibits an incorrect transition near the leading step edge, because a sample near the edge is closer in value to the height of the step. Total variation regularization and the robust total variation regularization (figure 1*b*,*i*) correctly ignore the outliers, but tend to identify many small, spurious edges; this is true also of iterated median filtering (figure 1*a*). Although these spurious jumps in total variation methods can be removed by further increasing the regularization parameter, this will be at the expense of introducing very significant bias into the estimate of the level of the constant regions (essentially, this is a consequence of the piecewise linearity of the regularization path). There is, however, no corresponding parametric control over the iterated length 3 median filter, which converges on a root signal that has many spurious jumps. Soft mean shift, convex mean shift and the weighted convex clustering shrinkage (figure 1*e*,*n*,*m*) fail to ignore the outliers and also show some spurious transitions between levels. The objective step-fitting (see table 2) method (figure 1*c*) also places jumps at the outliers, and in other, spurious locations. Convex clustering shrinkage fails to identify the step at all and is also influenced by the outliers (figure 1*g*).

With drift, we can see that mean shift, soft mean shift, K-means and mean shift total variation diffusion (figure 2*d*,*e*,*f*,*j*) are able to recover the step and ignore the drift very effectively. These methods are successful in this case because they are largely insensitive to the sequential ordering of the input samples (with the exception of mean shift total variation diffusion); they are simply

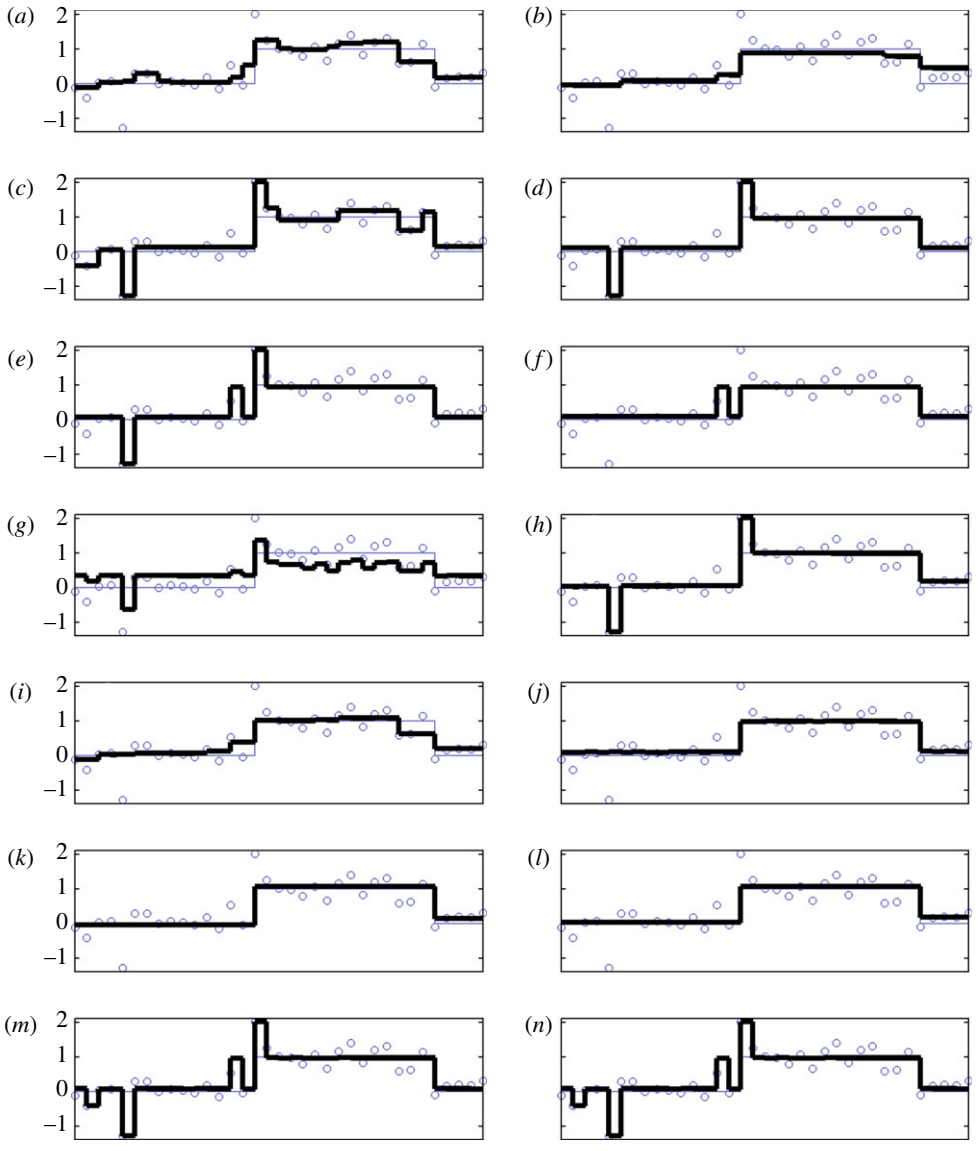

Figure 1. Response of PWC denoising methods to a step of unit height with additive Gaussian noise ($\sigma = 0.25$) and two extreme outliers. The methods are (a) iterated median filter for total variation diffusion, (b) total variation regularization ($\gamma = 1.5$), (c) objective step-fitting, (d) mean shift clustering ($W = 0.42$), (e) soft mean shift clustering ($\beta = 15$), (f) K-means ($K = 2$), (g) convex clustering shrinkage ($\gamma = 0.02$), (h) bilateral filter ($W = 2, \beta = 10$), (i) robust total variation regularization ($\gamma = 1.5$), (j) soft mean shift total variation diffusion ($\beta = 10, \gamma = 2.0$), (k) jump penalization ($\gamma = 1.0$), (l) robust jump penalization ($\gamma = 3.0$), (m) weighted convex clustering shrinkage ($\gamma = 1.0$, $W = 0.22$) and (n) convex mean shift clustering ($\gamma = 1.0$, $W = 0.22$). (Online version in colour.)

converging on peaks in the distribution of the input sample that turn out to be largely unaffected by the drift. Jump penalization, objective step-fitting and bilateral methods (figure 2k,l,c,h) are unable to ignore the drift, but produce

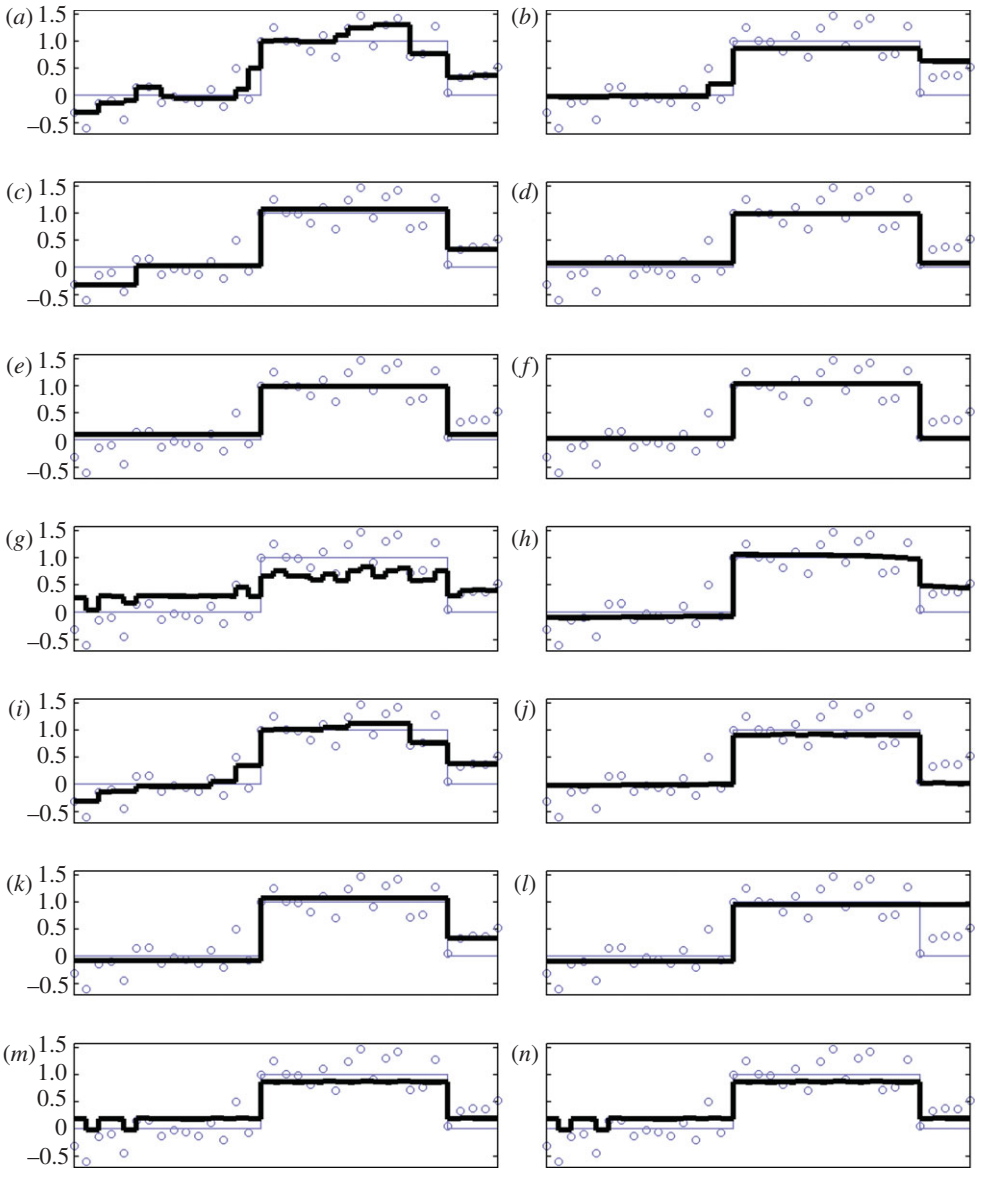

Figure 2. Response of PWC denoising methods to a step of unit height with additive Gaussian noise ($\sigma = 0.25$) and linear mean drift. The methods and parameters are as described in figure 1. (Online version in colour.)

the smoothest solutions. Weighted convex clustering shrinkage and convex mean shift (figure 2*m*,*n*) are not confused by the drift, but have some spurious edges. Total variation regularization is also adversely affected by the drift and introduces a small, incorrect jump, but is appreciably better than robust total variation regularization (figure 2*b*,*i*). Arguably, the worst performing methods are iterated median filtering and convex clustering shrinkage (figure 2*a*,*g*).

(*b*) *Qualitative comparisons: solver efficiency*

Next, in order to understand the efficiency of different methods and solvers, we apply a representative selection of iterative solvers to the basic task of noise removal from a short, unit synthetic step corrupted by the Gaussian noise. Figure 3 shows the resulting output signal, and the *iteration path* of the solver: that is, the curves traced out by the samples in the solution as the iterations proceed. This is a plot of the iteration number on the horizontal axis, against the values of the samples on the vertical axis. The distance reduction principle is apparent in the output as the solver iterates towards convergence to a minimum of the associated functional. It is also possible to discriminate methods that use only value kernels such as mean shift and K-means, from methods that use local sequence kernels (for example, total variation regularization and bilateral filtering). The former can only merge together samples that are close in value, therefore, the iteration paths do not intersect. On the other hand, the latter can constrain those that are sequentially close to merge together, and the iteration paths can intersect.

In terms of the number of iterations, the forward stepwise jump placement algorithm for jump penalization methods are the most effective, converging on a solution in two steps (figure 3*f*). Next, we find kernel adaptive step-size Euler integrators for mean shift, K-means and bilateral filtering taking at most five steps (figure 3*b*, *c*, *d*). The forward linear regularization path following solver for total variation regularization is next, taking 10 steps to reach the unique optimum solution (figure 3*a*). Weighted convex clustering shrinkage with non-adaptive step-size Euler integration takes some 300 steps to converge (figure 3*g*). Lastly, the two-step mid-point predictor–corrector integrator for mean shift total variation regularization converges to a solution after about 500 iterations (figure 3*e*).

Analytic minimizers for the generalized functional (1.1) are only available in the case of purely linear systems (simple quadratic loss functions). Therefore, numerical algorithms are required generally. The solvers described in this paper are not necessarily the most efficient that could be applied to each method. However, there are some general observations that can be made.

When the loss functions are convex and combined in convex combination this can be advantageous because then it is known that there is one unique minimizer for the functional, given fixed parameters. This avoids the uncertainty inherent to non-convex methods, where we do not know whether the solution obtained is the minimizer associated with the smallest possible value of the functional or not: there may be a better solution obtained by starting the solver from different initial conditions. This may require us to run the solver to convergence many times to gain confidence that the result is the best possible. Having said this, whether it matters that the solution is optimal depends on practical circumstances. For many PWC denoising methods the functionals are convex, and in terms of computational complexity, interior point algorithms are very efficient (Boyd & Vandenberghe 2004).

If there are only a few jumps then forward stepwise jump placement, as described in part I, §4.1 is very efficient. However, we cannot know whether a sequence of jumps placed by this forward-only algorithm is the best because the jump penalization functional is non-convex. Therefore, the same issues about uncertainty in the optimality of the results occur as with any non-convex

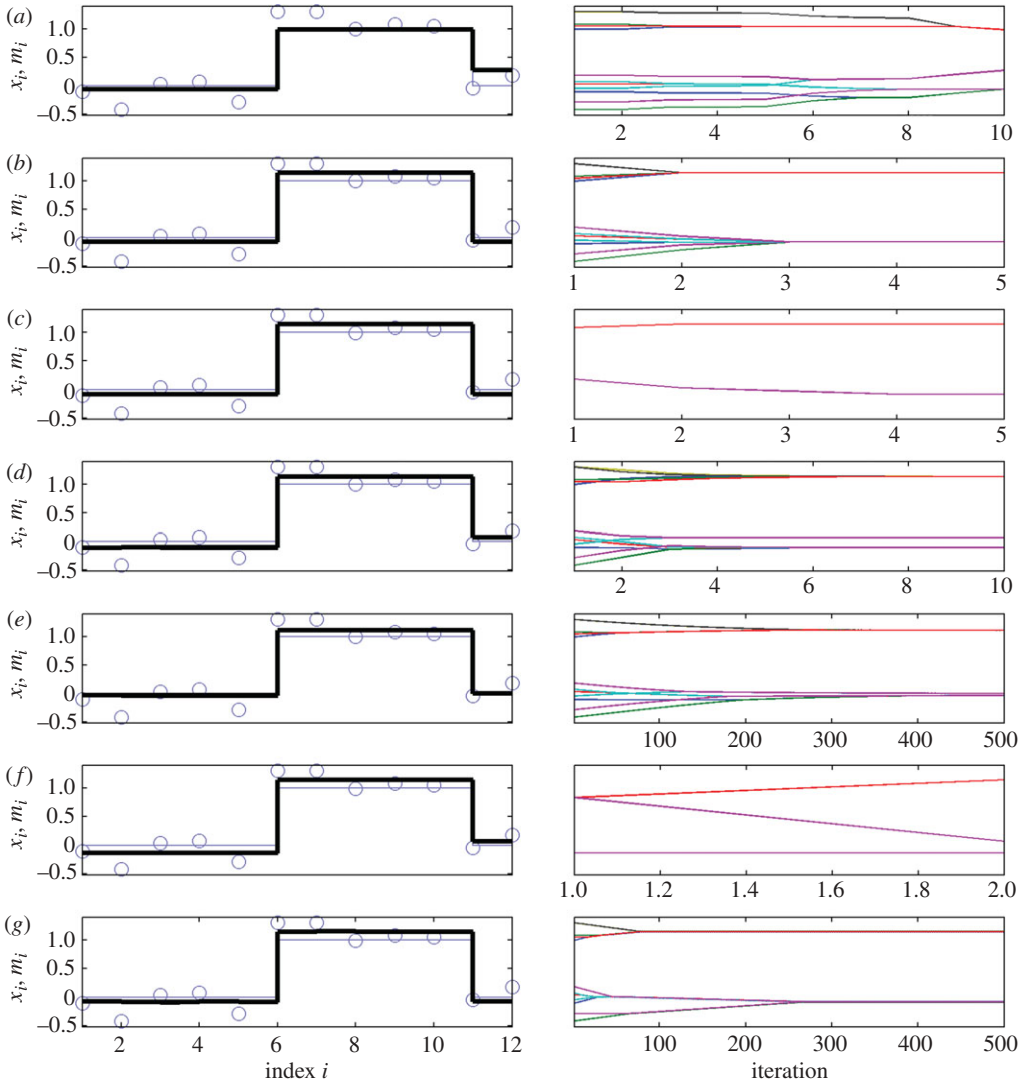

Figure 3. Iteration paths for solvers applied to a representative sample of PWC denoising methods. The noise is Gaussian ($\sigma = 0.25$). The left column shows the final, converged outputs of each method and the right column shows the associated iteration path taken towards convergence. The vertical axes are the values of the input (blue circles) and output (black line) samples, and the known PWC signal (thin blue line). The methods and solver algorithms are (*a*) total variation regularization by piecewise linear forward regularization path follower, (*b*) mean shift with adaptive step-size Euler integration, (*c*) *K*-means with adaptive step-size Euler integration, (*d*) bilateral filtering with adaptive step-size Euler integration (*e*) mean shift total variation diffusion with predictor–corrector two-step integration, (*f*) jump penalization with forward stepwise jump placement and (*g*) weighted convex clustering shrinkage with Euler integration. Method parameters are chosen to give good PWC recovery results. (Online version in colour.)

functional. The scope for stepwise jump placement algorithms is quite narrow, because it requires an easily solvable likelihood function given the fixed spline knots.

Although having the widest scope of all, we have seen that finite difference methods for the descent ODEs can take hundreds of steps to converge, and are therefore relatively inefficient. However, the simple measure of adapting the step-size can cut the number of iterations required to reach convergence enormously, as we have seen for the mean shift and other clustering methods. Simple finite differences are only practical then if modified with adaptive step-sizes or some other approach to speeding up convergence.

The scope for (forward) piecewise linear regularization path followers for PWC denoising turns out to be reasonably wide (Rosset & Zhu 2007), and if path linearity can be dropped, even wider (Rosset 2004). Therefore, if the full regularization path of solutions is required, path following methods can be efficient, as we have seen for total variation regularization. To our knowledge, backwards path following has only been investigated for total variation regularization.

Coordinate descent is probably the least efficient in terms of number of iterations and requires separability of the regularization term, which does not apply in general to the PWC denoising functionals in this paper. However, the update on each iteration is very simple and this may yet turn out to be competitive with other solvers applied where separability can be shown to hold.

(*c*) *Quantitative recovery performance: the unit step*

We now turn to the quantitative performance of each method at recovering a simple, known, unit height step signal u (length $N = 35$, first 17 samples are zero, the rest 1) with independent Gaussian noise of standard deviation σ. Performance is measured using the $\mathrm{RMSE} = \sqrt{\left(\frac{1}{N}\right) \sum_{i=1}^{N} |u_i - m_i|^2}$, and the normalized total variation $\mathrm{NTV} = \sum_{i=1}^{N-1} |m_{i+1} - m_i|$, where m is the final output of each algorithm. NTV measures the relative 'smoothness' of the resulting output with reference to the unit step input signal, which has unit total variation. Ideal recovery would occur when $\mathrm{RMSE} = 0$ and $\mathrm{NTV} = 1$. We vary the spread of the noise to probe the performance of each method as PWC noise removal becomes more challenging. Algorithm parameters were fixed according to the optimum values chosen for the qualitative analysis above. Results are averaged over 20 realizations of the noise.

In terms of RMSE, figure 4*a* shows that all methods generally get worse as the noise spread increases. Bearing in mind that the noisy input signal has RMSE equal to σ by definition, in fact at moderate to high noise spread, convex mean shift, weighted convex clustering shrinkage, K-means and the bilateral filter do not achieve significant noise reduction; but the rest of the methods do (because the RMSE is less than σ at all values of σ). Total variation regularization has the best RMSE for moderate to high noise spreads, but at the lowest spread, it is outperformed by the bilateral filter, mean shift total variation regularization, and both robust and non-robust jump penalization.

Taking into consideration NTV, figure 4*b* shows total variation regularization 'oversmoothing' the results at low noise spread, and 'undersmoothing' at high noise spread. By contrast, the novel robust jump penalization and soft mean shift total variation regularization methods produce outputs that are more consistent with the smoothness of the hidden step signal at both low and high noise spreads. The other methods all consistently underestimate the smoothness of the hidden

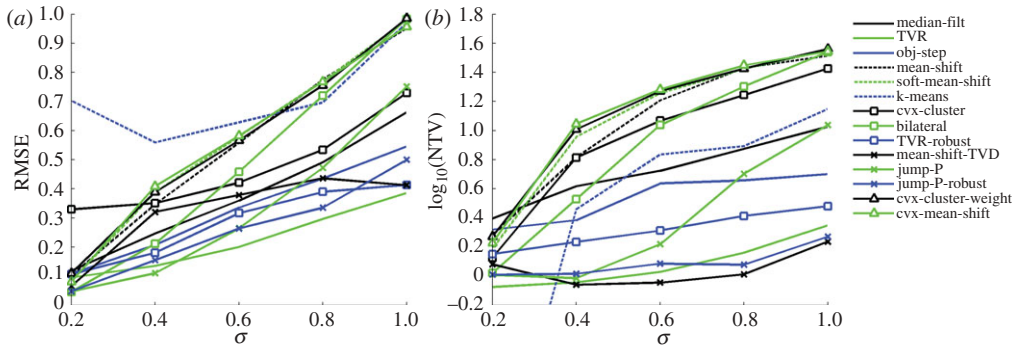

Figure 4. Quantitative performance of PWC denoising methods at recovering a unit step signal (length $N = 35$, first 17 samples are zero, the rest 1) corrupted by additive Gaussian noise of standard deviation σ. (*a*) RMSE (see text) of output signal with respect to increasing noise spread (note that perfect recovery would require RMSE = 0). (*b*) Logarithm of normalized total variation (see text) with increasing noise spread (N.B. perfect recovery would have $\log_{10}\mathrm{NTV} = 0$, positive indicates 'undersmoothing', and negative 'oversmoothing'). (Online version in colour.)

step, a situation that gets worse with increasing spread (with the exception of the K-means algorithm that overestimates smoothness at the very lowest noise spreads, because it has collapsed down to one single level).

Summarizing, these quantitative results point to the overall usefulness of total variation regularization and the novel robust jump penalization method presented here, when the noise is Gaussian, independent and stationary. However, by contrast to the qualitative analyses presented above, these results are averaged over the whole signal. In practice, it might be important to preserve and hence detect jumps of a certain height or location. In which case, as shown in the previous section, one might favour robust jump penalization over total variation regularization (because it tends to produce a PWC signal with a few, large jumps). Similarly, if we knew the noise spread, we could optimize the method parameters to improve performance.

(*d*) *Example application to real signals*

Here, we analyse a real signal that we suspect may be PWC: the DNA copy-number ratios arising from a genomic hybridization study (Snijders *et al.* 2001). We use this to illustrate some of the issues that arise with the choice of method, solver and parameter values in a practical setting. We suppose that the noise is independent and stationary, but that there are a few outliers. Further, we suppose we are interested in detecting a few, large jumps between different copy-number regimes. We, therefore, choose a method that tends to produce a handful of jumps within a few solver iterations, without any constraint on the number and value of levels: the robust jump penalization method that is a good performer on independent noise with or without outliers (see the results of the preceding sections). This will find a compact 0-degree spline, as opposed to level set, representation.

Results are shown in figure 5. As expected, different choices of the regularization parameter return PWC signals at different levels of detail, with a few knots breaking the genome sequence into different groups, see figure 5*c*.

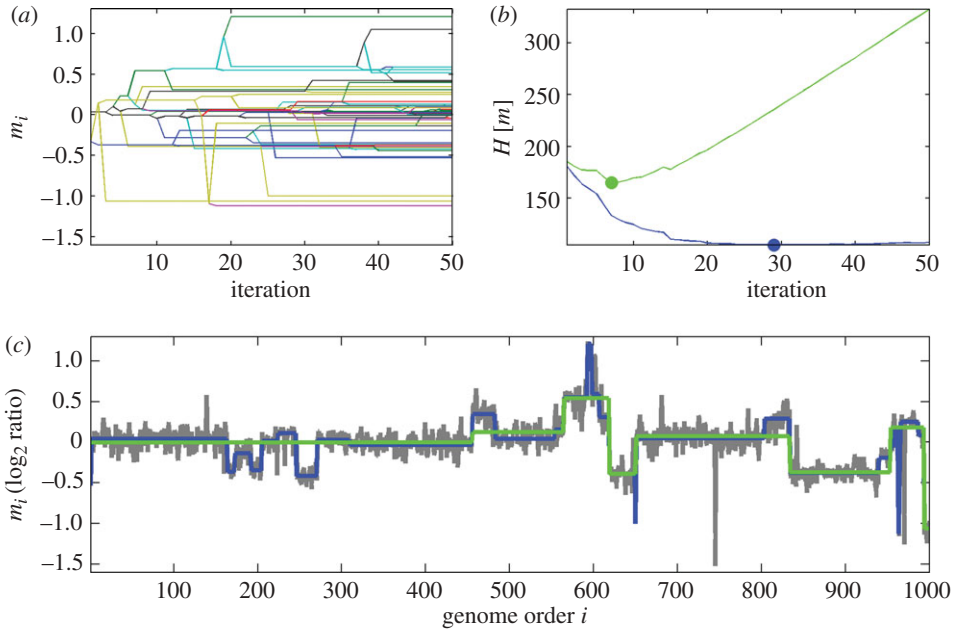

Figure 5. Analysis of DNA copy-number data obtained from a microarray genomic hybridization experiment (Snijders *et al.* 2001), using the novel robust jump penalization method described in the text. (*a*) Iteration paths traced out by the solver algorithm with increasing iterations. (*b*) Functional value $H[m]$ with increasing iteration number, for two different values of the regularization parameter γ. The filled circles represent the numerical minimum of this quantity over the first 50 iterations of the solver algorithm (blue line, $\gamma = 0.5$; green line, $\gamma = 5.0$). (*c*) PWC outputs obtained at the minimum of the functional for the two different regularization parameter values shown in (*b*) (blue line, m, $\gamma = 0.5$; green line, m, $\gamma = 5.0$), Overlaid on input signal x (shown in grey). (Online version in colour.)

In accordance with the robust nature of the median used in this method, the apparent 'spikes' at genome orders approximately 130 and 740 are ignored at both regularization values. The sample distance reduction principle (in reverse) is evident from the iteration paths in figure 5*a*.

We observe in figure 5*b* that the functional $H[m]$ is non-convex, so the choice of solver stopping criteria is open. Here, we make up to 50 iterations and select the output m where the functional attains the smallest value over all the previous iterations. At the larger regularization parameter value, the curvature of the functional is pronounced, making it easier to argue that the numerical minimum is meaningful: for the smaller regularization parameter value, it is not so clear where the minimum lies (despite the existence of a numerical minimum, it is very close to the functional value at other iterations).

In this paper, we have attempted to be neutral about the choice of parameters: their selection must depend on the peculiar information that the analyst has at hand. In selecting the range of the regularization parameter for this example, we are partly informed by what are considered 'reasonable' results by the standards of DNA copy-number analysts (Snijders *et al.* 2001). We noted in part I that regularization parameters, such as γ, can often be understood as parameters of a Bayesian prior and the choice of prior must be problem-specific. In some circumstances,

though not for the genomic data considered here, the analyst has access to a piece of signal which can be certified as having no steps. The distribution of this signal can be used to reason about the distribution of residuals of a PWC fit to a related signal which does have steps (assuming that the distribution of the noise is the same in both cases and the noise is additive). Since a good PWC denoising should leave a distribution of residuals identical to the distribution of the signal known to be step-free, this can be used to fix parameters by picking those parameters that make the two distributions as close as possible.

4. Summary, related and future ideas

In this second of two papers, by presenting an extensively generalized mathematical framework for performing PWC noise removal, several new PWC denoising methods and associated solver algorithms are proposed that attempt to combine the advantages of existing methods and solvers in potentially new and useful ways. Numerical tests on synthetic data compared the recovery accuracy and efficiency of these existing and novel methods head-to-head. It was found that under challenging conditions such as drifts in the noise, the new mean shift total variation denoising method is effective where existing methods show significant deficiencies. With or without outliers in the noise, the novel robust jump penalization method was shown to compare favourably to total variation regularization, but outperforms this method when only a few jumps in the signal are present or expected. Issues arising from the analysis of microarray genomic copy-number data using robust jump penalization were explored.

In order to devise these new PWC denoising methods, the first paper (part I) has presented a generalized approach to understanding and performing noise removal from PWC signals. While the structure of this study has encouraged us to make as inclusive an investigation as possible of PWC denoising methods, there are many other methods that cannot be associated with special cases of this generalized functional. Below, for completeness, we discuss the conceptual overlaps and relationships between some of these other methods that get significant use in practical PWC denoising applications.

(*a*) *Wavelets*

Wavelet techniques are ubiquitous, generic methods for signal analysis, and their use in general noise removal has been comprehensively explored (Mallat & Hwang 1992; Mallat & Zhong 1992; Wang 1995; Cattani 2004; Mallat 2009). Connections between wavelet techniques and some of the smoothing methods described in this paper, in particular total variation regularization (Steidl *et al.* 2004), have been established. Wavelet methods are powerful for many reasons, here we just mention a few of the basics: including (a) the existence of an $O(N)$ algorithm with computational complexity for the forward and reverse wavelet transforms in the discrete-time setting (Mallat 2009); (b) the statistical theory of *wavelet shrinkage* that exploits orthonormality of the wavelet basis to perform noise removal using very simple, coefficient-by-coefficient (*separable*) nonlinear transformations of the wavelet coefficients (Candes 2006); and (c) many signals, in the wavelet basis are *sparse*, that is, a large proportion of the coefficients are effectively zero making the wavelet representation very compact.

Wavelet methods require the choice of basis, and for PWC denoising, the *Haar basis*, itself composed of PWC functions, has been suggested many times in the wider literature (Cattani 2004; Taylor *et al.* 2010), although it is not the only basis that has been proposed. Removing noise typically requires removal of the small-scale detail in the signal. The result of removing this detail is that the time-localization of the remaining large scale PWC basis functions is poor, so that the jumps in the PWC signal cannot be accurately located and tend to become misaligned to the locations of the jumps in PWC bases instead. Furthermore, shrinkage causes 'oscillations' near jumps that are not aligned with the jumps in the basis; oscillations that are similar in character to Gibb's phenomena observed using linear low-pass filtering. These issues are an unavoidable consequence of the Heisenberg uncertainty inherent to time–frequency analysis (Mallat 2009).

The PWC denoising methods described in this paper are not based on time-frequency analysis. Perhaps because of this, historically, wavelet-based approaches, and the kind of methods discussed in this paper, have developed quite separately (Candes & Guo 2002). There are, however, some points of contact that have addressed how to prevent wavelet oscillations near jumps, yet retain some of the desirable conceptual and computational properties of wavelet methods. The literature on this topic is very extensive and we restrict ourselves to a few of the overlapping concepts that are of direct relevance to the PWC methods and solvers discussed in this paper.

If we are prepared to drop orthogonality, then we lose separability, but this does not mean that we lose the appealing concept of coefficient shrinkage: in fact, in the regression spline approach to total variation regularization discussed above, the use of the absolute function applied as a regularizer over the constant spline coefficients can be seen as *non-separable shrinkage* in the spline basis. The solver is more complex than separable shrinkage (we now have to solve a Lasso problem), but the jumps (spline knots) are no longer restricted by the Heisenberg uncertainty and can be placed precisely at the jumps in the PWC signal (Mallat 2009). Alternatively, Candes & Guo (2002) and Chan & Shen (2005) discuss how the wavelet reconstruction with absolute loss on the wavelet coefficients can be augmented with the total variation of the wavelet reconstruction to attempt to minimize the oscillations near discontinuities. The solution can no longer be obtained using separable shrinkage, but the orthogonality and potential sparsity of the wavelet transform is retained. A final example is that of *iterated translation invariant wavelet shrinkage* (Steidl *et al.* 2004), which has been shown to have similar performance to total variation regularization, but the connection is somewhat less direct.

(*b*) *Hidden Markov models*

HMMs play an important role in practical PWC denoising applications (Godfrey *et al.* 1980; Chung *et al.* 1990; Jong-Kae & Djuric 1996; McKinney *et al.* 2006). It is important, therefore, to understand the relationships between the generalized methods proposed in this paper and HMMs. The literature on the very many variants of HMMs is extensive (Blimes 2006), but we focus here on one of the most popular HMM variants that has seen repeated use in PWC denoising—the *discrete-state* HMM with *continuous, Gaussian emission probabilities*. This configuration has deep similarities to the (hard or soft) K-means clustering

algorithms discussed in this paper. The similarity emerges from the relationship between K-means clustering and (Gaussian) *mixture density modelling*.

In this HMM variant, there are K distinct *states* to the underlying *Markov chain*, each associated with a single Gaussian distribution, parameterized by K means and variances. If the underlying chain is in state $s_i = k \in \{1, 2, \ldots, K\}$ at index i, then the output sample from the noisy signal x_i is drawn from a Gaussian with mean μ_k and variance σ_k^2. The Markov chain is parameterized by K^2 additional *transition density* and *initial probability* variables, the transition density determining the statistical dependence of s_i upon s_{i-1} and earlier states if necessary (Blimes 2006).

The goal of fitting the HMM to the noisy signal is to find these transition and initial probabilities, and the parameters of the Gaussians associated with each state. If, however, s_i is independent of s_{i-1}, then this HMM variant collapses to a Gaussian mixture density model (Roweis & Ghahramani 1999), where the goal of fitting is to determine the parameters of the Gaussians alone. This is typically solved using expectation-maximization (EM) method (Hastie *et al.* 2001). There are two steps to this method, the *E-step*: in which the assignment of each index to each state is determined, and the *M-step* where the Gaussian parameters are re-estimated using the assignments. In this paper, the adaptive step-size Euler integrator applied to the K-means algorithms can be seen as a concatenation of these two steps, in the special case where the variances of the Gaussians are fixed. This arises because EM is equivalent to iterative, weighted mean and variance replacement, the weights determined by the state assignment. For soft K-means, the weights are the probabilities of assignment to each state given the means and variances from the previous iteration; for (hard) K-means, *most probable* assignments are used instead of probabilities, so the weights are either zero or one.

EM has been adapted to the HMM case of mixture modelling, where s_i depends on s_{i-1}. The E-step becomes more complex because calculating the state assignment probabilities requires 'tracing' through all possible states up until index i. Fortunately, conditional independence of the Markov chain makes a considerable algebraic simplification of this assignment possible, in the probabilistic assignment case the resulting method is known as the *Baum–Welch* algorithm, the most probable variant of which is *Viterbi* or *sequential K-means training* (Blimes 2006).

The means of the Gaussian associated with each state are analogous to the levels in the PWC level-set model, and this variant of HMM with continuous emission probabilities has the PWC property if the number of states because there will be many indices assigned sequentially to the same level. This explains why discrete-state HMMs with continuous emission probabilities are useful for general PWC denoising problems.

(*c*) *Piecewise constant versus piecewise smooth?*

The fact that PWC signals are also piecewise smooth implies that methods for noise removal from PWS signals can, in principle, be applied to the PWC denoising problem. Here, by PWS, we mean a signal that has a finite isolated

set of discontinuities (jumps), and everywhere else the function has one or many continuous derivatives. The PWS noise removal problem has attracted considerable attention, in particular from those applying wavelet analysis in the signal and image processing communities (Chaisinthop & Dragotti 2009; Mallat 2009). For PWS signals, the level-set model is no longer parsimonious (but see the *stack* or *threshold decomposition* representation that is of central importance to morphological signal processing, Arce 2005). The extension of the 0-degree spline model to higher degrees requires piecewise (first, second, etc.) differentiability, where the signal to be recovered is continuous everywhere, however, the PWC signals we refer to in this paper are discontinuous at the jump locations. Therefore, the higher degree spline model is not compact for PWS signals either. Here we discuss a small selection of PWS methods that are notable for their informative overlap with the algorithms in this paper.

Since noise removal from signals that are smooth everywhere is a problem for which the running mean filter is well suited, adapting the running linear filter to the existence of a few isolated jumps is a natural solution in many contexts. This requires some technique for (either implicitly or explicitly) detecting the existence of a jump. Many algorithms that provide jump capability to running filters (not just the running mean filter) exploit the concept of *data-adaptive weighting*, that is, some measure of the distance associated with samples inside (or outside) the local filtering window is used to provide a measure of whether a discontinuity exists within the window. This measure then changes the local weighting to mitigate the edge smoothing effect of filtering over the jump. In this paper, those techniques that place a kernel over the term $x_i - x_j$ are using such data-adaptive weighting.

In this context, it is informative to note that in the limit when $\beta \to 0$ in the bilateral filter formula, we obtain the iterated, running mean filter of width W, and with a soft (Gaussian) sequence kernel, we obtain the iterated running weighted mean filter. Therefore, one iteration of the bilateral filter can be viewed as a running weighted mean filter, where the weights are chosen to filter only those samples that are close in value (Elad 2002). Similar ideas have been proposed independently in many different disciplines. Chung & Kennedy (1991) describe a weighted running mean filter with a weighting scheme that is constant but different on the left-hand and right-hand sides of the window around each sample. The weights are inversely proportional to a positive power of the magnitude of the difference between the mean of the left or right sides of the window, and the sample in the middle of the window. The weights can be computed based on samples outside the filtering window, and the final output of the filter can be a summation over running means of differing lengths (Chung & Kennedy 1991). Running filters based on a variety of linear combinations of rank ordered samples in the window, such as the *trimmed mean filter* or the *double window modified trimmed mean filter* are conceptually similar and very useful for PWS noise removal (Gather *et al.* 2006).

The PWC denoising algorithms in this paper are therefore closely related to some PWS algorithms, but the PWC denoising problem is distinct. In particular, we present evidence here that the PWC denoising problem is one for which information across the whole signal can be efficiently exploited by constructing a

compact level-set representation, for example, using the full pairwise differences in sample values in the mean shift or weighted convex clustering shrinkage algorithms. This approach would not be efficient for PWS signals, because in between the jumps, a PWS signal is not generally constant, and so does not necessarily have a compact level-set description.

(*d*) *Continuum approaches and nonlinear partial differential equations*

The generalized functional (1.1) is based on a purely discrete-time setting. Most real signals are continuous in time, but despite continuous time being computationally inaccessible (it usually is), there are some mathematical advantages to going to a continuous time model of the signal, even if this has to be discretized later for computational reasons. The largest single class of continuous-time PWC denoising methods are those based on *nonlinear* PDEs, and have nearly all been developed in the image processing literature (Chan & Shen 2005). In the limit of infinitesimal time increments, the discrete-time, generalized functional becomes a double integral functional instead. Then, the *variational derivative* of the functional with respect to the continuous-time output signal is an *Euler–Lagrange* PDE, and it will be nonlinear if it is useful for PWC denoising. So, it is fairly easy to show that many, if not most, of the methods in this paper have an equivalent PDE form. Numerical solvers for this PDE would be very similar to numerical solvers for the descent ODEs derived earlier. Passing to the continuum also invites application of *Sethian's computational level-set algorithms* that, in the one-dimensional signal case, would correspond to techniques for evolving the jump locations between the distinct level-sets that comprise the PWC solution, as opposed to the levels (Chan & Shen 2005).

(*e*) *Future directions*

The new methods and solvers presented in this paper represent just a handful of directions that the generalized functional and solver description in part I suggests. Clearly, there are a very large number of other possible methods that can be constructed from the functional components we describe, that are as yet unexplored, that might be of value in PWC denoising. However, determining which of these methods would have minimizer(s) with the PWC property, and in addition, admit efficient and reliable solvers, will require additional work. We imagine one approach: a formal axiomatic system leading to the *scale-space equation* has been developed to the design of nonlinear PDEs for image analysis, that constrains their form to have universally useful properties (Chan & Shen 2005). It is quite possible that such axioms might be modified for PWC denoising purposes. The consequences of such axioms could be explored with respect to the functional components and their interactions with the solvers presented in this paper, with a view to asking what combinations lead to solutions with the PWC property.

Thanks to John Aston for comments. M.A.L. is funded through Wellcome Trust-MIT postdoctoral fellowship grant number WT090651MF, and BBSRC/EPSRC grant number BBD0201901. N.S.J. thanks the EPSRC and BBSRC and acknowledges grants EP/H046917/1, EP/I005765/1 and EP/I005986/1.

References

Arce, G. R. 2005 *Nonlinear signal processing: a statistical approach*. Hoboken, NJ: Wiley-Interscience.

Blimes, J. A. 2006 What HMMs can do. *IEICE Trans. Inform. Syst.* E89-D 869–891.

Boyd, S. P. & Vandenberghe, L. 2004 *Convex optimization*. Cambridge, UK: Cambridge University Press.

Candes, E. J. 2006 Modern statistical estimation via oracle inequalities. *Acta Numerica* **15**, 257–326. (doi:10.1017/S0962492906230010)

Candes, E. J. & Guo, F. 2002 New multiscale transforms, minimum total variation synthesis: applications to edge-preserving image reconstruction. *Signal Process.* **82**, 1519–1543. (doi:10.1016/S0165-1684(02)00300-6)

Cattani, C. 2004 Haar wavelet-based technique for sharp jumps classification. *Math. Comput. Model.* **39**, 255–278. (doi:10.1016/S0895-7177(04)90010-6)

Chaisinthop, V. & Dragotti, P. L. 2009 Semi-parametric compression of piecewise-smooth functions. In *Proc. of the Eur. Conf. on Signal Processing* (*EUSIPCO*). Glasgow, UK.

Chan, T. F. & Shen, J. 2005 *Image processing and analysis: variational, PDE, wavelet, and stochastic methods*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Chung, S. H. & Kennedy, R. A. 1991 Forward–backward nonlinear filtering technique for extracting small biological signals from noise. *J. Neurosci. Meth.* **40**, 71–86. (doi:10.1016/0165-0270(91)90118-J)

Chung, S. H., Moore, J. B., Xia, L., Premkumar, L. S. & Gage, P. W. 1990 Characterization of single channel currents using digital signal processing techniques based on hidden Markov models. *Phil. Trans. R. Soc. Lond. B* **329**, 265–285. (doi:10.1098/rstb.1990.0170)

Darbon, J. & Sigelle, M. 2006 Image restoration with discrete constrained total variation - part I: Fast and exact optimization. *J. Math. Imag. Vis.* **26**, 261–276. (doi:10.1007/s10851-006-8803-0)

Elad, M. 2002 On the origin of the bilateral filter and ways to improve it. *IEEE Trans. Image Process.* **11**, 1141–1151. (doi:10.1109/TIP.2002.801126)

Friedman, J., Hastie, T., Hofling, H. & Tibshirani, R. 2007 Pathwise coordinate optimization. *Ann. Appl. Statist.* **1**, 302–332. (doi:10.1214/07-AOAS131)

Gather, U., Fried, R. & Lanius, V. 2006 Robust detail-preserving signal extraction. In *Handbook of time series analysis* (eds B. Schelter, M. Winterhalder & J. Timmer), pp. 131–153. Weinheim, Germany: Wiley-VCH.

Gill, D. 1970 Application of a statistical zonation method to reservoir evaluation and digitized log analysis. *Am. Assoc. Petrol. Geol. Bull.* **54**, 719–729.

Godfrey, R., Muir, F. & Rocca, F. 1980 Modeling seismic impedance with Markov chains. *Geophysics* **45**, 1351–1372. (doi:10.1190/1.1441128)

Hastie, T., Tibshirani, R. & Friedman, J. H. 2001 *The elements of statistical learning: data mining, inference, and prediction*. Springer Series in Statistics. New York, NY: Springer.

Hofling, H. 2009 A path algorithm for the fused Lasso signal approximator. (http://arxiv.org/abs/0910.0526)

Iserles, A. 2009 *A first course in the numerical analysis of differential equations*. Cambridge, New York, NY: Cambridge University Press.

Jong-Kae, F. & Djuric, P. M. 1996 Automatic segmentation of piecewise constant signal by hidden Markov models. In *Proc. 8th IEEE Signal Processing Workshop on Statistical Signal and Array Processing, 1996* (*Cat. No.96TB10004*). pp. 283–286.

Kalafut, B. & Visscher, K. 2008 An objective, model-independent method for detection of non-uniform steps in noisy signals. *Comput. Phys. Commun.* **179**, 716–723. (doi:10.1016/j.cpc.2008.06.008)

Kerssemakers, J. W. J., Munteanu, E. L., Laan, L., Noetzel, T. L., Janson, M. E. & Dogterom, M. 2006 Assembly dynamics of microtubules at molecular resolution. *Nature* **442**, 709–712. (doi:10.1038/nature04928)

Kim, S. J., Koh, K., Boyd, S. & Gorinevsky, D. 2009 L1 trend filtering. *SIAM Rev.* **51**, 339–360. (doi:10.1137/070690274)

Koenker, R. 2005 *Quantile regression*. Econometric Society monographs; no. 38. Cambridge, UK: Cambridge University Press.

Koenker, R., Ng, P. & Portnoy, S. 1994 Quantile smoothing splines. *Biometrika* **81**, 673–680. (doi:10.1093/biomet/81.4.673)

Mallat, S. G. 2009 *A wavelet tour of signal processing: the sparse way*. Amsterdam, The Netherlands: Elsevier/Academic Press.

Mallat, S. & Hwang, W. L. 1992 Singularity detection and processing with wavelets. *IEEE Trans. Inform. Theory* **38**, 617–643. (doi:10.1109/18.119727)

Mallat, S. & Zhong, S. 1992 Characterization of signals from multiscale edges. *IEEE Trans. Pattern Anal. Mach. Intell.* **14**, 710–732. (doi:10.1109/34.142909)

McKinney, S. A., Joo, C. & Ha, T. 2006 Analysis of single-molecule FRET trajectories using hidden Markov modeling. *Biophys. J.* **91**, 1941–1951. (doi:10.1529/biophysj.106.082487)

Mehta, C. H., Radhakrishnan, S. & Srikanth, G. 1990 Segmentation of well logs by maximum likelihood estimation. *Math. Geol.* **22**, 853–869. (doi:10.1007/BF00890667)

Mrazek, P., Weickert, J. & Bruhn, A. 2006 On robust estimation and smoothing with spatial and tonal kernels. In *Geometric properties for incomplete data* (eds R. Klette, R. Kozera, L. Noakes & J. Weickert). Berlin, Germany: Springer.

Papadimitriou, C. H. & Steiglitz, K. 1998 *Combinatorial optimization: algorithms and complexity*. Mineola, NY: Dover.

Pelckmans, K., de Brabanter, J., Suykens, J. A. K. & de Moor, B. 2005 Convex clustering shrinkage. In *Proc. PASCAL Workshop on Statistics and Optimization of Clustering, London, UK, 4–5 July 2005*.

Rosset, S. 2004 Tracking curved regularization optimization solution paths. In *Advances in neural information processing*. Cambridge, MA: MIT Press.

Rosset, S. & Zhu, J. 2007 Piecewise linear regularized solution paths. *Ann. Statist.* **35**, 1012–1030. (doi:10.1214/009053606000001370)

Roweis, S. & Ghahramani, Z. 1999 A unifying review of linear Gaussian models. *Neural Comput.* **11**, 305–345. (doi:10.1162/089976699300016674)

Rudin, L. I., Osher, S. & Fatemi, E. 1992 Nonlinear total variation based noise removal algorithms. *Physica D* **60**, 259–268. (doi:10.1016/0167-2789(92)90242-F)

Schmidt, M., Fung, G. & Rosales, R. 2007 Fast optimization methods for L1 regularization: a comparative study and two new approaches. *Proc. Mach. Learn. ECML 2007,* **4701**, 286–297, 809. (doi:10.1007/978-3-540-74958-5_28)

Snijders, A. M., *et al.* 2001 Assembly of microarrays for genome-wide measurement of DNA copy number. *Nat. Genet.* **29**, 263–264. (doi:10.1038/ng754)

Steidl, G., Weickert, J., Brox, T., Mrazek, P. & Welk, M. 2004 On the equivalence of soft wavelet shrinkage, total variation diffusion, total variation regularization, and SIDEs. *SIAM J. Numer. Anal.* **42**, 686–713. (doi:10.1137/S0036142903422429)

Steidl, G., Didas, S. & Neumann, J. 2006 Splines in higher order TV regularization. *Int. J. Comput. Vis.* **70**, 241–255. (doi:10.1007/s11263-006-8066-7)

Taylor, J. N., Makarov, D. E. & Landes, C. F. 2010 Denoising single-molecule FRET trajectories with wavelets and Bayesian inference. *Biophys. J.* **98**, 164–173. (doi:10.1016/j.bpj.2009.09.047)

Wang, Y. 1995 Jump and sharp cusp detection by wavelets. *Biometrika* **82**, 385. (doi:10.1093/biomet/82.2.385)