Nonlinear regularization techniques for seismic tomography

I. Loris [a,*], H. Douma [b], G. Nolet [c], I. Daubechies [d], C. Regone [e]

[a] Mathematics Department, Vrije Universiteit Brussel, Pleinlaan 2, 1050 Brussels, Belgium
[b] Department of Geosciences, Princeton University, Guyot Hall, Washington Road, Princeton, NJ 08544, United States
[c] Geosciences Azur, Université de Nice–Sophia Antipolis, CNRS/IRD, 250 Rue Albert Einstein, Sophia Antipolis 06560, France
[d] Program in Applied and Computational Mathematics, Princeton University, Fine Hall, Washington Road, Princeton, NJ 08544, United States
[e] BP America Inc., 501 Westlake Park Blvd., Houston, TX 77079, United States

ARTICLE INFO

ABSTRACT

The effects of several nonlinear regularization techniques are discussed in the framework of 3D seismic tomography. Traditional, linear, ℓ_2 penalties are compared to so-called sparsity promoting ℓ_1 and ℓ_0 penalties, and a total variation penalty. Which of these algorithms is judged optimal depends on the specific requirements of the scientific experiment. If the correct reproduction of model amplitudes is important, classical damping towards a smooth model using an ℓ_2 norm works almost as well as minimizing the total variation but is much more efficient. If gradients (edges of anomalies) should be resolved with a minimum of distortion, we prefer ℓ_1 damping of Daubechies-4 wavelet coefficients. It has the additional advantage of yielding a noiseless reconstruction, contrary to simple ℓ_2 minimization ('Tikhonov regularization') which should be avoided. In some of our examples, the ℓ_0 method produced notable artifacts. In addition we show how nonlinear ℓ_1 methods for finding sparse models can be competitive in speed with the widely used ℓ_2 methods, certainly under noisy conditions, so that there is no need to shun ℓ_1 penalizations.

© 2009 Elsevier Inc. All rights reserved.

1. Introduction

Since geophysical inverse problems are almost always underdetermined, regularization techniques are essential to obtain a meaningful solution. Two major classes of techniques exist. The first one, named 'mollifying' in the mathematical literature, or 'optimally localized averaging' (OLA) in helioseismology, can be traced back to the groundbreaking work of Backus and Gilbert [1,2] in geophysics. In this approach one searches for the size of an averaging volume that can produce a local average of the model parameter with an acceptable variance. Since this method is computationally very expensive, it has found little application in large-scale geophysical inversions such as seismic tomography. To limit the computational effort, seismic tomographers instead search for a biased ('damped') solution. This has the disadvantage of introducing a systematic error – the bias – in lieu of the random error caused by the propagation of data errors. It can be turned into an advantage if the bias actually reflects a justified disposition of the scientist to prefer certain models over others, as long as the data are fit within their error bars.

Simple ℓ_2-norm damping, which biases model perturbations towards zero in the absence of information based on the data, is generally a bad choice to regularize the inverse problem for seismic tomography as it tends to introduce structures

* Corresponding author.
E-mail address: igloris@vub.ac.be (I. Loris).

reflecting ray coverage into the images. For that reason, most tomographers prefer to bias the model towards 'smooth' anomalies, in effect trying to forge a compromise between Backus–Gilbert theory and the efficiency of damped inversions. The smoothness of the images has the advantage that large structures become easily visible. Sharp discontinuities, however, are blurred, and smaller structures, even when resolved, may be diluted beyond recognition. Recently, Loris et al. [3] — hereafter referred to as Paper I — introduced a third option for the bias in geophysical inversions: to minimize the ℓ_1 norm of the wavelet decomposition. This also biases the model towards zero, but it turns out that such reconstructions always have many (or most) wavelet coefficients *exactly* equal to zero (i.e. they are sparse). In a synthetic 2D experiment using surface wave data, we showed how structurally coherent features (in a geophysical sense), were more faithfully reproduced using this technique than with a simple ℓ_2-norm damping. In addition, as a result of their inherent sparsity, ℓ_1 reconstructions exhibit much less noise than their ℓ_2 counterparts.

Though Paper I clearly showed the feasibility of wavelet-based ℓ_1 regularization it left a number of questions unanswered, in particular which wavelet families work best, how they would perform against more sophisticated ℓ_2 norms (e.g. smoothness damping) and whether the computational feasibility as well as the positive conclusions for wavelet regularization do scale up to large, 3D models.

In this paper we therefore aim to refine the original conclusions of Paper I and to enlarge the scope of the investigation. We extend tests to 3D inversions of body wave travel times and investigate the use of different families of wavelets (Haar, D4, and dual tree). We include a comparison with smoothness damping, and with a fourth option named 'total variation' damping. Unlike Paper I, we also discuss so-called ℓ_0 constrained recovery. In contrast to the ℓ_1-norm technique which relies on iterative soft-thresholding, the ℓ_0 recovery method uses iterative hard-thresholding of wavelet coefficients [4,5].

The (salt dome) model that we try to reconstruct here is more realistic than the model in Paper I and includes a wide range of length scales. The problem described in Paper I was also of a very limited size: there were only about 10^4 degrees of freedom in the reconstructed models. Here we perform 3D reconstructions, and increase the number of degrees of freedom by an order of magnitude to about $\sim 10^5$. The number of data also increases accordingly to 24000. Our approach is complementary to that of [6] who expand the Fréchet kernels into wavelets to obtain a significant reduction in the memory requirements to store the kernel.

One disadvantage of the ℓ_1 norm, ℓ_0 and total variation penalties is that they lack the convenient linearity of the more conventional ℓ_2-norm minimizations. Making use of recent algorithmic improvements, we do demonstrate that finding a (nonlinear) sparse model reconstruction is not necessarily more expensive, computationwise, than ℓ_2-based (linear) reconstructions.

The use of ℓ_1 norms in seismic tomography was to the best of our knowledge first proposed in [7] in the form of an iteratively reweighted least squares method (IRLS, see also [8]), but has never found much favor, possibly because the convergence of IRLS was not guaranteed. Besides its use in seismic tomography ℓ_1 norms have found application in other geophysical contexts such as deconvolution and interpolation [9–12]. The use of ℓ_1 norms in combination with a carefully chosen basis (such as wavelets) is, however, more recent and is largely inspired by the recent development of compressed sensing [13–15], that shows that under certain conditions an exact reconstruction can be obtained by solving an ℓ_1 regularized inverse problem, provided there is an underlying basis in which the desired model is sparse. This emphasizes the importance of studying different "dictionaries" as we do here. Recently [16] has successfully applied the compressed sensing idea to wavefield reconstruction, albeit on small-scale problems only. The success and promise of compressed sensing has therefore also increased interest in the speed-up of such ℓ_1 problems to be able to handle practical geophysical applications (e.g. [17,18]).

2. Forward problem formulation

We plan to test the regularization methods on a synthetic data set generated for a salt dome model. Since the main goal of this paper is to evaluate and compare a number of algorithms, numerical efficiency is more important than the wish to have a tomographic problem at hand that is fully realistic. We have thus taken a few shortcuts to be able to run inversions quickly in Matlab on a single processor. However, we took pains to ensure that we would invert for a model that has a large range of length scales, and that the ray coverage would encompass both dense and sparse regions.

In dimensionless variables, the expression for a finite frequency sensitivity kernel corresponding to a constant background and a Gaussian power spectrum is given by the formula:

$$K(x,y,z) = \frac{e^{-u^2}H_5(u)}{24\lambda d_s d_r},$$ (1)

where $u = \pi(d_s + d_r - d_{sr})/\lambda$. Here d_{sr} is the distance between source (earthquake) and receiver (station), and d_s, d_r are the distances to source and to receiver, measured from the point (x,y,z). λ is the dominant wavelength and $H_5(u) = 120u - 160u^3 + 32u^5$ denotes the Hermite polynomial of order 5. Eq. (1) can be derived from the expressions for the Fréchet kernel in a homogeneous medium using an analytical expression for the spectral integration (see [19]).

The constant background model that we use here is not at all realistic from a physical point of view for the salt dome input model that we will use in Section 4: in case of large velocity contrasts kernels are bent rather than straight as in expression (1). However, as we will use the same constant background kernels (1) for generating synthetic data as well

as for reconstructing models from these data, we believe it is possible to accurately evaluate the effects of the regularization technique on the inversions. Obviously, reconstructing a model from actual measurements requires kernels of a more complicated shape than (1) but their resolving power is not fundamentally different from those used in our simple application.

For each source–receiver pair and each dominant wavelength, the travel time differential and the model perturbation **m** are connected by the linear integral relation [20][1]

$$\delta T = \int_V K \mathbf{m} \, dV. \tag{2}$$

Given sufficiently many data, the aim of seismic tomography is to reconstruct the model **m** from a noisy version of the data vector **d** containing many travel time differentials corresponding to many source–receiver pairs.

In Section 4 we will perform a number of seismic reconstructions. The domain on which we will do this is the cube $V = [-1, 1]^3$. For discretization, this domain is subdivided in 64^3 voxels, a convenient choice for the digital reconstruction of a model **m**, leading to 262,144 degrees of freedom in **m**. In order to be able to produce meaningful reconstructions, we expect to need at least 10^4 data (about 1 datum for 10 degrees of freedom). Hence we will choose 4800 source–receiver pairs and five different dominant wavelengths so as to yield 24,000 data. This represents an overparametrization by a factor of more than 10. Thus regularization will be an essential requirement for any data inversion.

A very efficient set-up of our numerical experiment was obtained as follows: we first choose 100 source–receiver pairs in random positions on the surface of the cube, while making sure source and receiver are never on the same face (the kernels (1) are not curved and would not be able to cross the model domain very much if source and receiver were on the same face). From these initial 100 pairs, we construct the full set of 4800 pairs by using the 48 symmetry transformations of the cube. These 48 operations are constructed from the 3! permutations of the coordinates (x, y, z), and by the 2^3 reflections $(\pm x, \pm y, \pm z)$, as listed in Table 1. In other words, starting from one initial kernel $K(x, y, z)$ we easily obtain 47 other kernels: $K(z, x, y), K(y, x, -z), \ldots$, corresponding to differently positioned source–receiver pairs. The random nature of the positions of the original 100 source–receiver pairs (i.e. no coordinate is exactly zero or exactly equal to plus or minus another coordinate) ensures us that none of the 4800 source–receiver pairs are identical. The initial 100 source–receiver pairs and the final 4800 pairs are shown in Fig. 1.

For each of the 4800 source–receiver pairs we will construct five finite frequency sensitivity kernels of type (1) corresponding to five different dominant wavelengths $\lambda = \{0.5, 0.2, 0.08, 0.04, 0.025\}$. Thus, in total there will be 24,000 kernels at our disposal. Because of the symmetry transformations used and the random choice of the initial 100 source–receiver pairs, the coverage of the domain by these 24,000 kernels is quite uniform. A picture that illustrates this property is too large to include here, but it is available in the online Supplementary material as Fig. A.1.

In order to give the reader an idea of the size of the Fresnel zones of these finite frequency kernels, we include in Fig. 2 and in one of the panels of Figs. 6 and 7 a cross-sectional view of five kernels. Each of these kernels corresponds to one of the five wavelengths ($\lambda \in \{0.5, 0.2, 0.08, 0.04, 0.025\}$) and source–receiver distance $d_{sr} = 2$. The width of the Fresnel zone is proportional to $\sqrt{\lambda d_{sr}}$ [21]. The cross sections in Fig. 2 illustrates not only the typical widths of the kernels, but also their relative amplitudes.

Additionally we construct a second operator containing only 20,000 out of the 24,000 kernels. We choose to remove the 4000 kernels for which the line connecting source and receiver (i.e. the central ray) comes closest to the point $(0.24, -0.7, -0.23)$. In this way, we end up with an operator that has a 'hole' in its coverage of the cube (see Fig. A.1, right). Using this operator we will be able to study the effect of non-uniform coverage on reconstructions.

To discretize the model into voxels we calculate the integral of the sensitivity kernels over each voxel (using a Riemann sum with 4^3 terms/voxel) for each source–receiver pair and for each of the five dominant wavelengths we consider. The resulting values make up the operator **A** we want to invert: Because there are 24,000 kernels and $64^3 = 262,144$ voxels, the matrix **A** will have 24,000 rows and 262,144 columns.

The use of the 48 symmetry transformations of the cube allows us to save a factor of 48 in memory requirements for our calculations, i.e. we need to compute and store only 500 kernels corresponding to the initial 100 source–receiver pairs and 5 wavelengths. The remaining ones are easily (and quickly) generated from these 500 kernels by the symmetry transformations of the cube (quick rearrangements of the elements in a 3D array).

An additional saving in memory requirements is obtained by exploiting the fact that most kernels are well localized (i.e. they are thin), and thus that the A_{ij} are practically zero for many voxels. In other words, each row of the matrix **A** is relatively sparse.

Although this set-up is completely unrealistic from a physical perspective, this configuration of rays provides for an easy way to compare dense and partial data coverage of the model. It also allows us to focus on the relative merits of the inversion methods rather than on the difficulties a physically faithful modeling would entail.
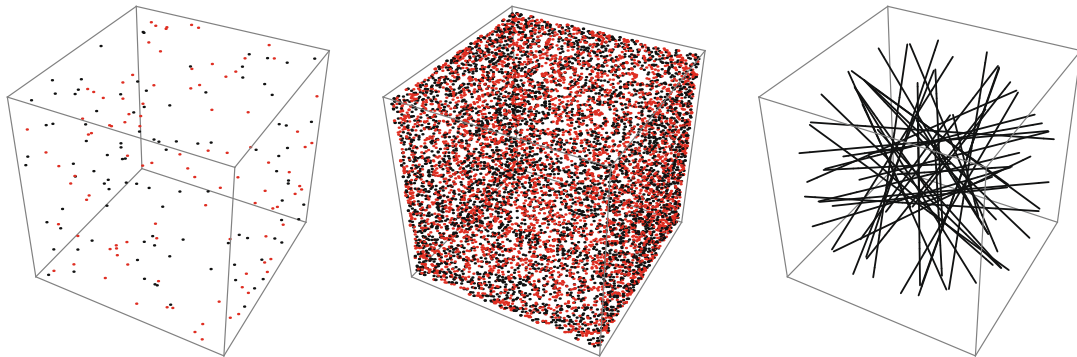
In reality, sensor coverage is limited to the surface and a few boreholes. High-frequency data with narrow Fresnel zones therefore leave significant areas at depth not illuminated by acoustic waves. Low-frequency data have wider Fresnel zones but suffer a reduced sensitivity to small length scales. Illumination itself does not guarantee resolution as becomes clear when one regards the singular value spectrum of **A**: though the spectrum depends strongly on the experimental set-up, it

---

[1] For convenience, we denote the model perturbation by **m** instead of $\delta \mathbf{m}$ as is often done.

Table 1
A list of rotations and reflections that map the unit cube $[-1,1]^3$ onto itself. They are constructed by combining the six permutations of (x,y,z) and eight sign changes of the components (x,y,z).

	Permutations					
Sign changes	(x,y,z)	(x,z,y)	(y,x,z)	(y,z,x)	(z,x,y)	(z,y,x)
	$(x,y,-z)$	$(x,z,-y)$	$(y,x,-z)$	$(y,z,-x)$	$(z,x,-y)$	$(z,y,-x)$
	$(x,-y,z)$	$(x,-z,y)$	$(y,-x,z)$	$(y,-z,x)$	$(z,-x,y)$	$(z,-y,x)$
	$(x,-y,-z)$	$(x,-z,-y)$	$(y,-x,-z)$	$(y,-z,-x)$	$(z,-x,-y)$	$(z,-y,-x)$
	$(-x,y,z)$	$(-x,z,y)$	$(-y,x,z)$	$(-y,z,x)$	$(-z,x,y)$	$(-z,y,x)$
	$(-x,y,-z)$	$(-x,z,-y)$	$(-y,x,-z)$	$(-y,z,-x)$	$(-z,x,-y)$	$(-z,y,-x)$
	$(-x,-y,z)$	$(-x,-z,y)$	$(-y,-x,z)$	$(-y,-z,x)$	$(-z,-x,y)$	$(-z,-y,x)$
	$(-x,-y,-z)$	$(-x,-z,-y)$	$(-y,-x,-z)$	$(-y,-z,-x)$	$(-z,-x,-y)$	$(-z,-y,-x)$

Fig. 1. (Left) The cube $[-1,1]^3$ with the initial 100 source–receiver pairs (black, source and red, receiver). (Middle) The 4800 source–receiver pairs one obtains by applying the 48 symmetry transformations of Table 1 on the initial 100 pairs. (Right) A single source–receiver pair joined by a straight line and its 48 transformations. Plotting all 4800 rays obtained in this way would fill the whole cube. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
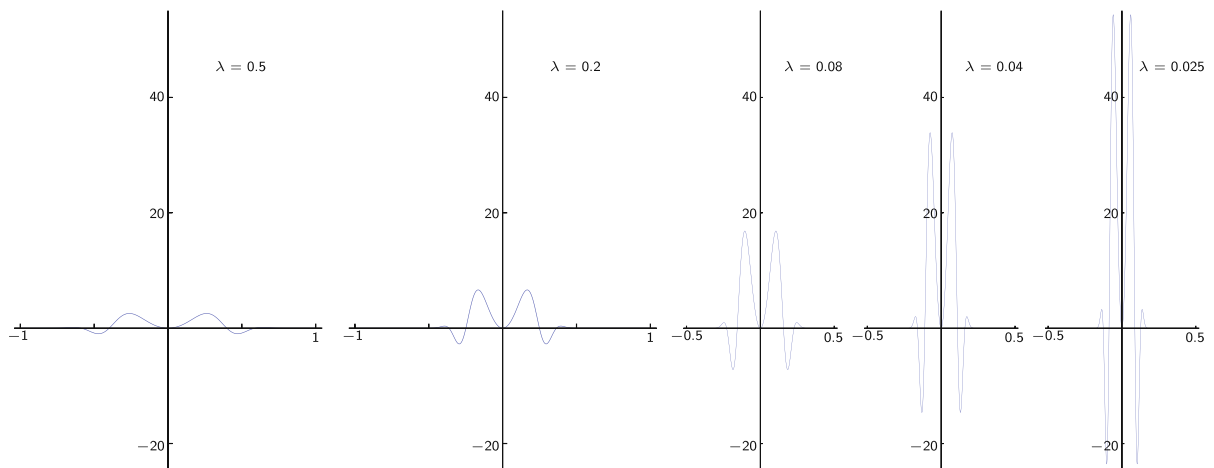
Fig. 2. Cross sections (perpendicular to the midpoint of the central ray) of five finite sensitivity kernels corresponding to $\lambda \in \{0.5, 0.2, 0.08, 0.04, 0.025\}$. Larger wavelength kernels are wider but have smaller amplitude.

often has a rapid drop-off and a significant fraction of eigenvalues is either zero or too small to be useful. In the present synthetic experiment, we chose to have about 1 datum for every 10 degrees of freedom in the model. The uniform random distribution of sources and receivers on the surface of the cube has a favorable influence on the singular value spectrum of **A** (we were able to confirm this on a scaled-downed version of the present operator **A**, but not on the full $24{,}000 \times 262{,}144$ matrix), and will aid in the reconstruction . This will allow us to focus on the characteristics of the different reconstruction methods, rather than on the lack of data.

3. Reconstruction methods

Reconstructing the model **m** from the data **d** is done, in principle, by solving the linear system **Am=d**, where **A** denotes, as before, the matrix containing the kernels discretized in the voxel basis. This system may contain incompatible equations (due to noise), and at the same time be underdetermined (not enough data to reconstruct all of **m**).

The problem of incompatible data can be solved by replacing the original problem with the minimization of a data fidelity term:

$$\bar{\mathbf{m}} = \arg\min_{\mathbf{m}} \|\mathbf{Am} - \mathbf{d}\|^2. \tag{3}$$

Here and in the following $\|u\|$ (without subscripts) always denotes the usual 2-norm of u: $\|u\| = \sqrt{\sum_i u_i^2}$. Although a minimizer always exists (because of the quadratic nature of the functional), it may not always be unique. In other words, the problem is still underdetermined and an iterative numerical scheme for finding a minimizer of (3) may diverge. In fact, because of the existence of data errors we are not even looking for the exact minimizer (3). We rather augment the functional in (3) by a term that will penalize a whole category of models that is thought to be unphysical.

3.1. ℓ_2 penalties

The prime example of this kind of method is Tikhonov regularization [22] whereby a penalty proportional to the ℓ_2 norm of the model is imposed:

$$\bar{\mathbf{m}} = \arg\min_{\mathbf{m}} \|\mathbf{Am} - \mathbf{d}\|^2 + \mu\|\mathbf{m}\|^2. \tag{4}$$

This will effectively prevent the model from growing unboundedly due to noise in **d** and ill-conditioning of the matrix **A**.

Another, closely related, possibility is to impose a penalty consisting of the ℓ_2 norm of the (discrete) Laplacian $\Delta\mathbf{m}$ of **m**:

$$\bar{\mathbf{m}} = \arg\min_{\mathbf{m}} \|\mathbf{Am} - \mathbf{d}\|^2 + \mu\|\Delta\mathbf{m}\|^2. \tag{5}$$

As we will see in Section 4, this will enforce a certain degree of smoothness on the reconstructed model. The Laplacian used here is defined as the difference of the model with the local average over the six nearest neighbors: $(\Delta\mathbf{m})_{ijk} = \mathbf{m}_{ijk} - (\mathbf{m}_{i-1jk} + \mathbf{m}_{i+1jk} + \ldots + \mathbf{m}_{ijk+1})/6$.

The above two versions of the Tikhonov regularization method have the advantage of being solvable by linear equations. The variational equations that determine the minimizers of (4) and (5) are:

$$\mathbf{A}^T\mathbf{Am} + \mu\mathbf{m} = \mathbf{A}^T\mathbf{d}, \tag{6}$$

and

$$\mathbf{A}^T\mathbf{Am} + \mu\Delta^T\Delta\mathbf{m} = \mathbf{A}^T\mathbf{d}, \tag{7}$$

respectively (with suitable treatment of boundary voxels, $\Delta^T = \Delta$ in the latter case). For these linear equations, we can use the conjugate gradient algorithm. With $\mathbf{m}^{(0)}$ = arbitrary and $\mathbf{r}^{(n)} = \mathbf{A}^T(\mathbf{d} - \mathbf{Am}^{(n)}) - \mu\mathbf{D}^T\mathbf{Dm}^{(n)}$, we set

$$\mathbf{v}^{(n)} = \begin{cases} \mathbf{r}^{(n)} & n = 0 \\ \mathbf{r}^{(n)} + \frac{\|\mathbf{r}^{(n)}\|^2}{\|\mathbf{r}^{(n-1)}\|^2}\mathbf{v}^{(n-1)} & n > 0 \end{cases}$$
$$\mathbf{m}^{(n+1)} = \mathbf{m}^{(n)} + \frac{\|\mathbf{v}^{(n)}\|^2}{\|\mathbf{Av}^{(n)}\|^2 + \mu\|\mathbf{Dv}^{(n)}\|^2}\mathbf{v}^{(n)}, \tag{8}$$

where **D** is either the unit matrix **I** (in case we seek to minimize functional (4)) or Δ (in case we use functional (5)). The model estimates $\mathbf{m}^{(n)}$ converge to the minimizer (4) or (5), respectively, as n increases.

3.2. ℓ_1 penalties

Another and much more recent method of regularization consists of imposing a carefully chosen ℓ_1-norm penalty [23]. It can be shown that this leads to a sparse model, i.e. a model with few nonzero components [14,15]. It would therefore *not* be a good idea to apply this technique to the model in the voxel basis (there is no reason to assume the model would be sparse in that basis); we would rather use this penalty on the coefficients of the model in a different basis in which we believe the model to be sparse.

Harmonic functions would allow us to select resolvable scales, but the complete lack of localization of these functions makes them even worse candidates than voxels. 3D wavelets offer a compromise between the concentration of power in both scale and location, and are intuitively more suitable to build geophysically reasonable models. In fact, our earlier experience (Paper I) showed the advantages of using a wavelet basis, and constructing the model by finding the minimum of the functional

$$\|\mathbf{Am} - \mathbf{d}\|^2 + 2\mu\|\mathbf{w}\|_1, \tag{9}$$

where $\mathbf{w} = W\mathbf{m}$ are the wavelet coefficients of $\mathbf{m}$. This minimization problem can be rewritten as:

$$\bar{\mathbf{m}} = \mathbf{W}^{-1}\bar{\mathbf{w}} \quad \text{and} \quad \bar{\mathbf{w}} = \arg\min_{\mathbf{w}} F(\mathbf{w}) \quad \text{with} \quad F(\mathbf{w}) = \|\mathbf{AW}^{-1}\mathbf{w} - \mathbf{d}\|^2 + 2\mu\|\mathbf{w}\|_1. \tag{10}$$

$\mathbf{W}$ is the wavelet decomposition matrix and $\mathbf{W}^{-1}$ the wavelet synthesis operator. This type of ℓ_1-norm penalty leads to a model that has a sparse wavelet representation, i.e. a model with (very) few nonzero wavelet coefficients. The aim is thus to rely on the properties of the wavelet basis to be able to represent the desired solution with few nonzero components. In geophysics wavelets are a good choice for seismic reconstruction as they allow for sparse representations of overall smooth functions, while still capable of taking into account the possibility of isolated sharp features [24,3].

Another advantage of the ℓ_1 method is that they yield noise-free model reconstructions. This is a consequence of the simple fact that noise in the model cannot be represented in a sparse way (in any reasonable basis). This method is thus able to produce clean models without the need for additional smoothing. It is important to note that the least-squares functional (10) is convex. This implies that a local minimum of (10) is always a global minimum as well.

In Section 4, we shall consider a number of different choices of orthonormal wavelet bases. For each of these choices, we have $\mathbf{W}^{-1} = \mathbf{W}^T$, which we will implicitly assume hereafter.

In order to find the minimizer $\bar{\mathbf{w}}$ of the ℓ_1-penalized functional (10), one may use the iterative soft-thresholding algorithm [23]:

$$\mathbf{w}^{(n+1)} = T(\mathbf{w}^{(n)}), \tag{11}$$

with

$$T(\mathbf{w}) = \mathcal{S}_{\alpha\mu}\left[\mathbf{w} + \alpha\mathbf{WA}^T\left(\mathbf{d} - \mathbf{AW}^T\mathbf{w}\right)\right], \tag{12}$$

and where $\mathbf{w}^{(0)}$ may be chosen arbitrarily. The soft-thresholding $\mathcal{S}_\tau(u)$ function operates componentwise and is defined by

$$\mathcal{S}_\tau(u) = \begin{cases} u - \tau & u \geqslant \tau \\ 0 & |u| \leqslant \tau \\ u + \tau & u \leqslant -\tau. \end{cases} \tag{13}$$

This algorithm was used in a 2D seismic tomography toy problem in Paper I, to which we also refer for a brief but elementary derivation (Section 2 of Paper I). In effect, it is a simple gradient descent algorithm (with fixed step length α) where the additional soft-thresholding operation $\mathcal{S}_{\alpha\mu}$ is a mathematical consequence of the ℓ_1-norm term present in functional (10).

The constant α should be chosen such that $\alpha\|\mathbf{A}^T\mathbf{A}\|$ is smaller than or equal to 1 ($\|\mathbf{A}^T\mathbf{A}\|$ is defined as the largest eigenvalue of $\mathbf{A}^T\mathbf{A}$) [23]. In our calculation we always choose $\alpha = 0.95/\|A^TA\|$.

The wavelet transform $\mathbf{W}$ and its inverse $\mathbf{W}^T$ are fast transforms. This means that they cost only a fraction of the computer time needed to perform one application of $\mathbf{A}$ or $\mathbf{A}^T$. In other words, working in a wavelet basis does not significantly change the computational complexity of the reconstruction algorithm.

As the iterative soft-thresholding algorithm (11) can be slow in practice, we have opted here for using the so-called Fast Iterative Soft-Thresholding Algorithm (FISTA) [25] (see also earlier work of Nesterov [26,27]):

$$\mathbf{w}^{(n+1)} = T\left(\mathbf{w}^{(n)} + \frac{t_n - 1}{t_{n+1}}\left(\mathbf{w}^{(n)} - \mathbf{w}^{(n-1)}\right)\right), \tag{14}$$

with $t_0 = 1$ and $t_{n+1} = \left(1 + \sqrt{1 + t_n^2}\right)/2$. The FISTA algorithm has practically the same computational complexity as the iterative soft-thresholding algorithm (11). It only requires one additional vector addition. With this algorithm the ℓ_1 penalized cost function (10), evaluated at $\mathbf{w} = \mathbf{w}^{(n)}$, is bounded by $\mathcal{O}(1/n^2)$ from its limiting value:

$$F(\mathbf{w}^{(n)}) - F(\bar{\mathbf{w}}) \leqslant 4\frac{\|\mathbf{w}^{(0)} - \bar{\mathbf{w}}\|^2}{\alpha(n+1)^2}, \tag{15}$$

as opposed to the $\mathcal{O}(1/n)$ decrease:

$$F(\mathbf{w}^{(n)}) - F(\bar{\mathbf{w}}) \leqslant \frac{\|\mathbf{w}^{(0)} - \bar{\mathbf{w}}\|^2}{\alpha n}, \tag{16}$$

that can be proven for algorithm (11). These upper bounds are valid non-asymptotically, i.e. even for small n [25,28].

3.3. ℓ_0 penalties

Recently, mathematical advances on direct ways of constraining the number of nonzero components in a reconstructed model have appeared. In [4,5] an iterative hard-thresholding algorithm is proposed of the following form:

$$\mathbf{w}^{(n+1)} = \widetilde{T}(\mathbf{w}^{(n)}), \tag{17}$$

with

$$\widetilde{T}(\mathbf{w}) = \mathcal{H}_k\left[\mathbf{w} + \alpha \mathbf{WA}^T\left(\mathbf{d} - \mathbf{AW}^T\mathbf{w}\right)\right] \tag{18}$$

and where the hard-thresholding operation $\mathcal{H}_k(\mathbf{u})$ sets all but the largest (in absolute value) k components of $\mathbf{u}$ to zero. This algorithm converges to a *local* minimum of

$$\|\mathbf{AW}^{-1}\mathbf{w} - \mathbf{d}\|^2 \qquad \text{under the condition} \qquad \|\mathbf{w}\|_0 \leqslant k. \tag{19}$$

Here $\|\mathbf{w}\|_0$ denotes the number of nonzero coefficients of $\mathbf{w}$. Just as with the ℓ_1 penalty method, we shall apply this technique using a wavelet basis. The underlying reason is again the suitability of a wavelet basis to represent a physically acceptable model with few nonzero wavelet coefficients. We stress that the appearance of the hard-thresholding operation $\mathcal{H}_k$ is a mathematical consequence of the constraint $\|\mathbf{w}\|_0 \leqslant k$, just as soft-thresholding in (12) is a mathematical consequence of the ℓ_1 term in (10) [23,3].

The algorithm (17) converges very slowly. We therefore propose the following FISTA/Nesterov-like modification:

$$\mathbf{w}^{(n+1)} = \widetilde{T}\left(\mathbf{w}^{(n)} + \frac{t_n - 1}{t_{n+1}}\left(\mathbf{w}^{(n)} - \mathbf{w}^{(n-1)}\right)\right), \tag{20}$$

with $t_0 = 1$ and $t_{n+1} = (1 + \sqrt{1 + t_n^2})/2$. As far as the authors know, this is the first time this algorithm is proposed. Although there is no proof of convergence yet, we found that it worked quite well on the examples that we studied (see Section 4). We used the same choice for the step-length α as with the ℓ_1 algorithm: $\alpha = 0.95/\|A^TA\|$. The choice of the number of nonzero model wavelet coefficients k in method (19) is discussed in Section 3.5.

On a side note, it would also be possible to calculate a local minimizer of the (non-convex) functional

$$\|\mathbf{AW}^{-1}\mathbf{w} - \mathbf{d}\|^2 + \mu\|\mathbf{w}\|_0 \tag{21}$$

using componentwise hard-thresholding (with a fixed threshold $\sqrt{\mu}$) [4]. However, [5] seems to prefer the formulation (19) that imposes a *fixed* number of nonzeros in each step of the iteration, rather than a fixed lower bound $\sqrt{\mu}$ for the absolute values of the wavelet coefficients.

Although no proof of convergence of algorithm (20) is given, we shall refer to it by the name 'ℓ_0 method' (instead of calling it 'hard-thresholded Nesterov accelerated gradient descent algorithm').

3.4. Total variation penalty

A final penalty term we will consider is the so-called 'total variation' (TV) penalty:

$$\bar{\mathbf{m}} = \arg\min_{\mathbf{m}} \|\mathbf{Am} - \mathbf{d}\|^2 + 2\mu \sum_{ijk} \sqrt{(\Delta_x\mathbf{m})^2 + (\Delta_y\mathbf{m})^2 + (\Delta_z\mathbf{m})^2}, \tag{22}$$

with $\Delta_x\mathbf{m} = \mathbf{m}_{ijk} - \mathbf{m}_{i-1jk}$ and similarly for Δ_y and Δ_z. This penalization will favor piecewise constant models in the voxel basis. Unfortunately, the equations that determine the minimizer (22) are again nonlinear. We will use a reweighed conjugate gradient method [29] to determine the minimum of the TV functional (22). More specifically, defining the weights $\mathbf{u}^{(n)} = (\Delta_x\mathbf{m}^{(n)} + \Delta_y\mathbf{m}^{(n)} + \Delta_z\mathbf{m}^{(n)})^{-1/2}$, we shall use algorithm (8) where we choose $\mathbf{D}^T\mathbf{D} = \Delta_x^T\mathbf{u}^{(n)}\Delta_x + \Delta_y^T\mathbf{u}^{(n)}\Delta_y + \Delta_z^T\mathbf{u}^{(n)}\Delta_z$ (which depends on the iteration n) and use that $\|\mathbf{Dv}^{(n)}\| = \langle\mathbf{D}^T\mathbf{Dv}^{(n)}, \mathbf{v}^{(n)}\rangle$. Because of the non-quadratic nature of functional (22), the conjugate gradient algorithm no longer preserves conjugacy between successive search directions $\mathbf{v}^{(n)}$ as n grows. For this reason, the iteration is also reset every so often (in accordance with [29]).

An algorithm similar to (11) and (14) may also be used to find the minimizer of the TV-penalized functional (22) [30,31]. In fact, in [30] it is shown how the ℓ_1-norm penalty and the TV penalty can be combined in a single functional.

3.5. Choice of the penalty parameter

As such, the minimizers defined by (4), (5), (10) and (22) still depend on the penalty parameter μ, or in case of the hard-thresholding algorithm (20) on the parameter k. In the reconstructions below, we select this parameter by requiring in each instance that the reconstructed model $\bar{\mathbf{m}}$ fits the data $\mathbf{d}$ as well as possible, but not any better than the noise level:

$$\|\mathbf{A}\bar{\mathbf{m}} - \mathbf{d}\| \approx \|\mathbf{n}\|, \tag{23}$$

with $\mathbf{n}$ representing the noise vector. In other words, the discrepancy principle tells us to choose the penalty parameter μ (or k) such that

$$\|\mathbf{A}\bar{\mathbf{m}} - \mathbf{d}\|^2 \Big/ \sigma^2 \approx \text{number of data}, \tag{24}$$

where σ is the noise variance (if different data have different variance, it is simplest to divide from the outset each row as well as the right hand side by the standard deviation of its datum, and set $\sigma = 1$ in (24)). In practice, this means that we will have to perform the minimization several times, until a suitable value of μ or k is determined for each reconstruction.

4. Reconstructions

In this section we present some sample reconstructions using the algorithms mentioned in Section 3, applied to the finite frequency tomography problem described in Section 2. First we will consider a simple checkerboard input model that we will try to reconstruct from incomplete and noisy data. We will also look at a more complicated salt dome model which we obtained from BP America, Inc.

In each case, our procedure will be the following. We start from a known input model $\mathbf{m}^{input}$ from which we construct synthetic noisy data $\mathbf{d}$:

$$\mathbf{d} = \mathbf{A}\mathbf{m}^{input} + \mathbf{n}. \tag{25}$$

The noise $\mathbf{n}$ is taken from a Gaussian distribution, with zero mean and variance σ chosen in such a way that $\|\mathbf{n}\|/\|A\mathbf{m}^{input}\| = 0.1$; in other words, we add 10% noise to the noiseless data. The goal is then to try to reconstruct $\mathbf{m}^{input}$ as well as possible from $\mathbf{d}$ and $\mathbf{A}$. For this we will use methods (4), (5), (10), (19) and (22), and compare the results. Since we know the noise variance σ, we can use the criterion (24) to choose the penalty parameter μ or k. This parameter will be different for the various synthetic data and for the various penalties that we impose.

For the wavelet-based methods (10) and (19), we also need to choose a specific wavelet family. There are many wavelet bases, with varying degrees of smoothness and approximation properties [24]. This gives us the opportunity to adapt our choice of wavelet basis to the model: we will choose the basis in which we suspect the model to be sparse. In particular, for the checkerboard reconstruction we will use Haar wavelets because we know that the true solution is very sparse in that basis. We thus expect a very accurate reconstruction in that case. For the salt dome input model, we will compare the effects of different choices of wavelets bases on the reconstruction. Our choice will include Haar wavelets, D4 wavelets and also directional dual tree wavelets. Haar wavelets are the least smooth, and directional wavelets are the most smooth of these three.

4.1. Checkerboard

The first example consists of a checkerboard pattern; in other words the input model $\mathbf{m}^{input}$ is piecewise constant (±1) on small cubes of 8 by 8 by 8 voxels. It mainly serves as a proof of principle for the ℓ_1 wavelet method (10) because we know that this input model is sparse in the Haar wavelet basis [32]. The Haar wavelets are piecewise constant. In fact, the model only has 99 nonzero coefficients in that basis (out of 262,144). Hence, we may expect that the ℓ_1 method will work very well with this model and this basis.

A single horizontal slice of the checkerboard input model and its four reconstructions are shown in Fig. 3. These four reconstructions use all 24,000 kernels so that no particular region in the model is favored or disadvantaged. The ℓ_1 reconstruction (10) with Haar wavelets is visually the most faithful to the original, closely followed by the ℓ_0-Haar reconstruction. The ℓ_2-reconstructions 4, 5 and the TV reconstruction (22) display smooth transitions between +1 and −1. The reconstruction result of the simple ℓ_2 method (4) is quite noisy, which is not the case for the other methods. For completeness and viewing convenience, the online Supplementary material includes a picture (Fig. A.2) of all the horizontal slices of the input model and its various reconstructions.

The smoothing effect of the reconstructions can quantitatively be seen from the histogram of the reconstructed model amplitudes (see Fig. 4). The ℓ_2 reconstruction takes on mostly values around zero, whereas the input model only has amplitudes +1 and −1 (vertical blue[2] lines). In this case, the ℓ_1 Haar reconstruction does a very good job at recovering the ±1 amplitude distribution, as it is naturally well suited for the particular checkerboard model used. The ℓ_0 method does second best and the TV method (22) does third best from this point of view. It is surprising that the ℓ_1 method outperforms the ℓ_0 method in this case.

The checkerboard model shows that good reconstructions are possible if one has very good prior knowledge on the model. When using the ℓ_0 and ℓ_1 methods this requires a basis in which the desired model is very sparse. The checkerboard model that was used here aligns optimally with the chosen Haar basis. The results of ℓ_1 and ℓ_0 reconstructions deteriorate when the checkerboard pattern is shifted w.r.t the Haar basis or when the size of the fields are changed (the Haar basis decomposition of such a checkerboard would seize to be sparse). As such, the checkerboard model we choose is very particular. For realistic reconstruction scenarios one should not expect such excellent results.

A discussion of the mean square error of the various reconstructions is given in Section 4.3, where other computational aspects are also discussed.

4.2. A 3D salt dome model

In this section we try to reconstruct a complex 3D model of a realistic salt body in the subsurface. The complex salt dome model was obtained from a prestack depth-migration of field seismic data in the deep-water part of the Gulf of Mexico, and was kindly provided to us by BP America, Inc. To better accommodate the straight-ray tomography used in this paper, the

---

[2] For interpretation of the references to color in this figure the reader is referred to the web version of this article.
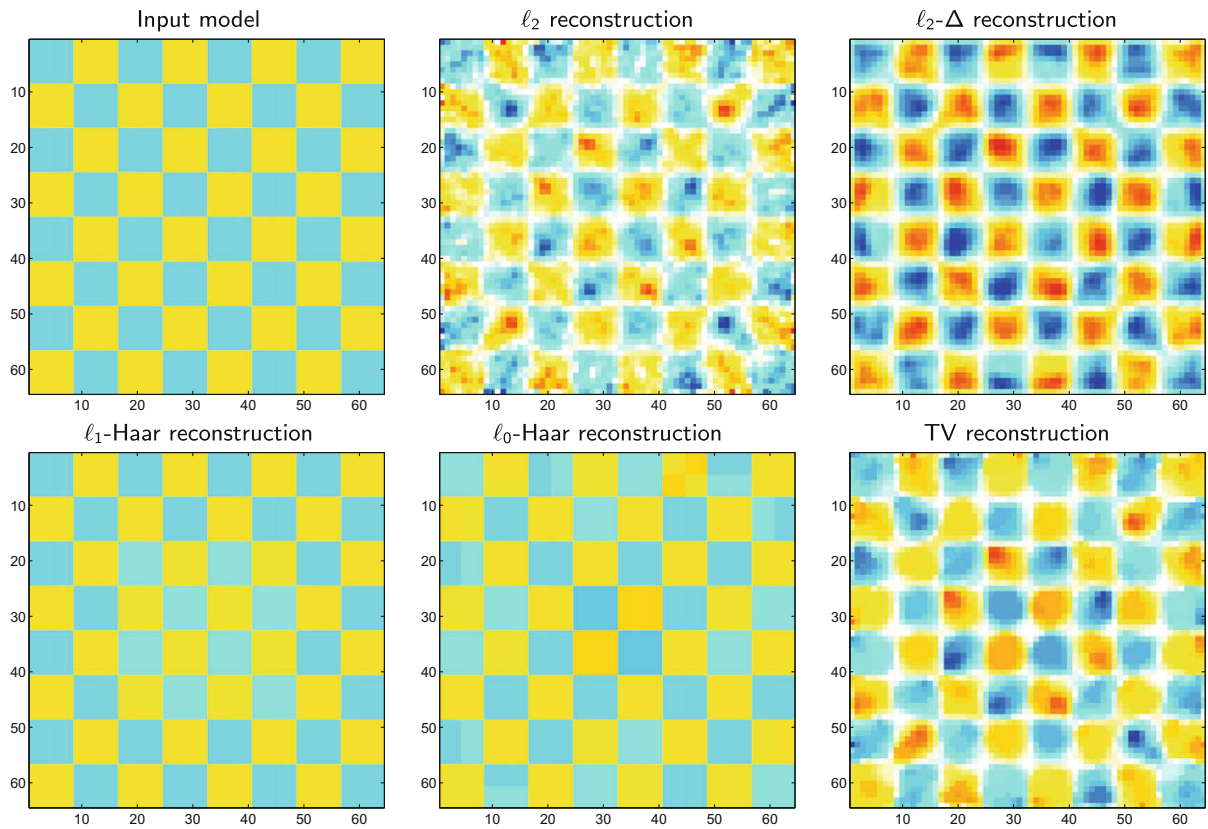
Fig. 3. A single horizontal slice (number 25 (from top) out of 64) of the checkerboard model and its reconstructions using all 24,000 data. The whitish transitions in several of the reconstructions are the result of smoothing between +1 and −1 of the input model. As the input checkerboard has fields of size $8 \times 8 \times 8$ with constant model value ±1, the smoothing occurs with period 8 as well. In addition, the ℓ_2 reconstruction has a distinctive noisy appearance. The ℓ_1 reconstruction using Haar wavelets is visually indistinguishable from the input model. The ℓ_0 method does almost as well.
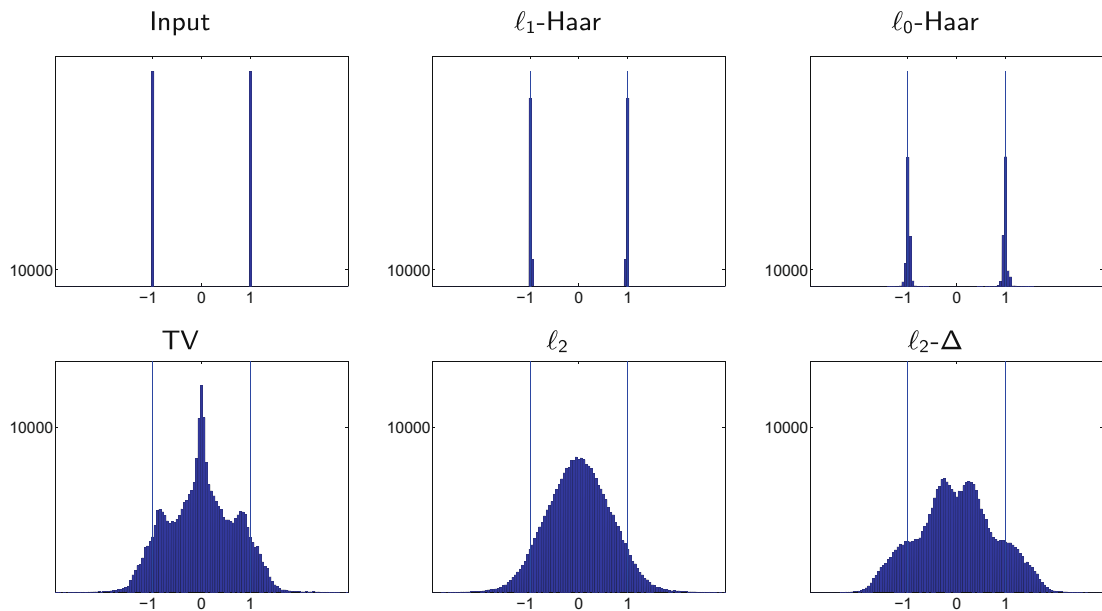
Fig. 4. Histograms of the amplitude distribution of the five checkerboard reconstructions. The input model takes on values +1 and −1 only. The ℓ_1 reconstruction in the Haar basis, and to a lesser extent the ℓ_0-Haar reconstruction, result in almost perfect reconstruction of the amplitude distribution whereas the amplitudes are shifted towards zero by the other reconstruction methods.

sediment velocities surrounding the salt dome model that were present in the original model provided to us, were replaced with a constant velocity. This model was superimposed on a background model with long-wavelength variations (smoothed Gaussian). The model has a rather sharp contrast between the velocity in the salt and in the surrounding background model, providing for sharp edges. A single horizontal slice through the resulting salt dome model is pictured in Figs. 6 and 7. For completeness, all 64 horizontal slices are shown in Fig. A.3 (right) of the online Supplementary material (as well as the smoothed Gaussian that was added in; Fig. A.3, center). Three contour plots of this model are shown in Fig. 5, corresponding to the model values $\mathbf{m} = 0.9, \mathbf{m} = 1$ and $\mathbf{m} = 1.1$. The model contains $64 \times 64 \times 64 = 262,144$ voxels as in the checkerboard examples.

We perform the same type of experiment as before: we construct synthetic data and add 10% Gaussian noise to it. From this noisy data, we try to reconstruct the input model. There are two differences with the checkerboard reconstructions. First, we will compare several different wavelet families for the ℓ_1 and ℓ_0 reconstructions (in this case, there is no obvious reason to prefer Haar wavelets over other wavelet bases). Second, we will repeat the reconstruction experiment for an operator that has only 20,000 kernels instead of 24,000, to simulate imperfect coverage of the model domain by the kernels. In other words, with the 20,000 kernel reconstruction, a region of the model is ill resolved.

The wavelet families used are, in order of increasing smoothness: Haar [32], D4 [33,34] and so-called directional dual tree (DT) wavelets [35,36]. The Haar and D4 wavelet transform on the cube are direct products of the corresponding wavelet transforms in 1D. The DT wavelet transform is not and it has, by construction, better directional sensitivity. The D4 wavelets that we will use do not suffer from edge effects as they do not use periodic boundary conditions, but follow the interval scheme proposed in [33,34]. Other model parameterizations that could be used are shearlets or curvelets (they are particularly suited to sparsely represent models with singularities along curves or surfaces), but we did not include them in our study [37–39].

Judging the success of an algorithm to reconstitute the input model invariably involves a degree of subjectiveness, even if one designs a numerical measure for goodness of model fit. Such measure might also depend on the goal of the scientific experiment conducted. For example, if one deducts temperatures from velocity variations, it is more important that the amplitudes are correct and less important that sharp edges of an anomaly are preserved, but a structural geologist may be more interested in the edges and may wish to involve the misfit of the gradient, for example.

In Table 2, we list the amplitude misfit ($\|\mathbf{m} - \mathbf{m}^{\text{input}}\| / \|\mathbf{m}^{\text{input}}\|$) and judge the fit to other features visually. A single horizontal slice (number 25) of the different reconstructions is pictured in Fig. 6 and all 64 horizontal slices are shown in Fig. A.4 in the online Supplementary material. For the reconstructions using all 24,000 kernels, the TV method works best based on the final resulting error, as well as visual inspection. It is closely followed by the ℓ_1 method using dual tree wavelets (ℓ_1-DT) and by the ℓ_2 method with Laplacian smoothing ($\ell_2 - \Delta$). The ℓ_1 method with D4 wavelets does better than with Haar wave-
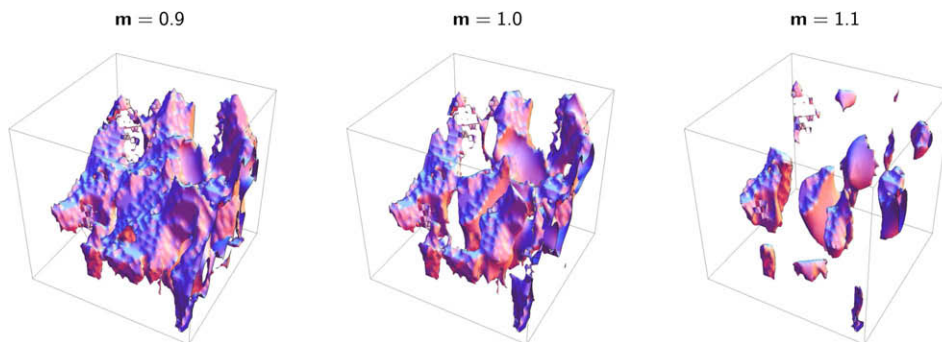

Fig. 5. Three contour plots, i.e. surfaces of constant model value, of the salt dome model ($\mathbf{m} = 0.9, 1.0, 1.1$).

Table 2
The number of iterations and the resulting reconstruction error for the various models and methods.

	Checkerboard		Saltdome w. 24,000 data		Saltdome w. 20,000 data	
	Iterations	Error (%)	Iterations	Error (%)	Iterations	Error (%)
ℓ_2	100	68.8	100	48.8	100	56.9
$\ell_2 - \Delta$	100	61.6	100	40.1	100	46.4
ℓ_1-Haar	100	1.8	1000	48.1	100	55.1
ℓ_1-D4	–	–	1000	42.8	100	50.0
ℓ_1-DT	–	–	1000	39.5	100	46.8
ℓ_0-Haar	100	4.4	1000	61.5	100	67.9
ℓ_0-D4	–	–	1000	52.6	100	63.2
ℓ_0-DT	–	–	1000	44.1	100	49.4
TV	100	64.0	1000	39.0	100	49.7
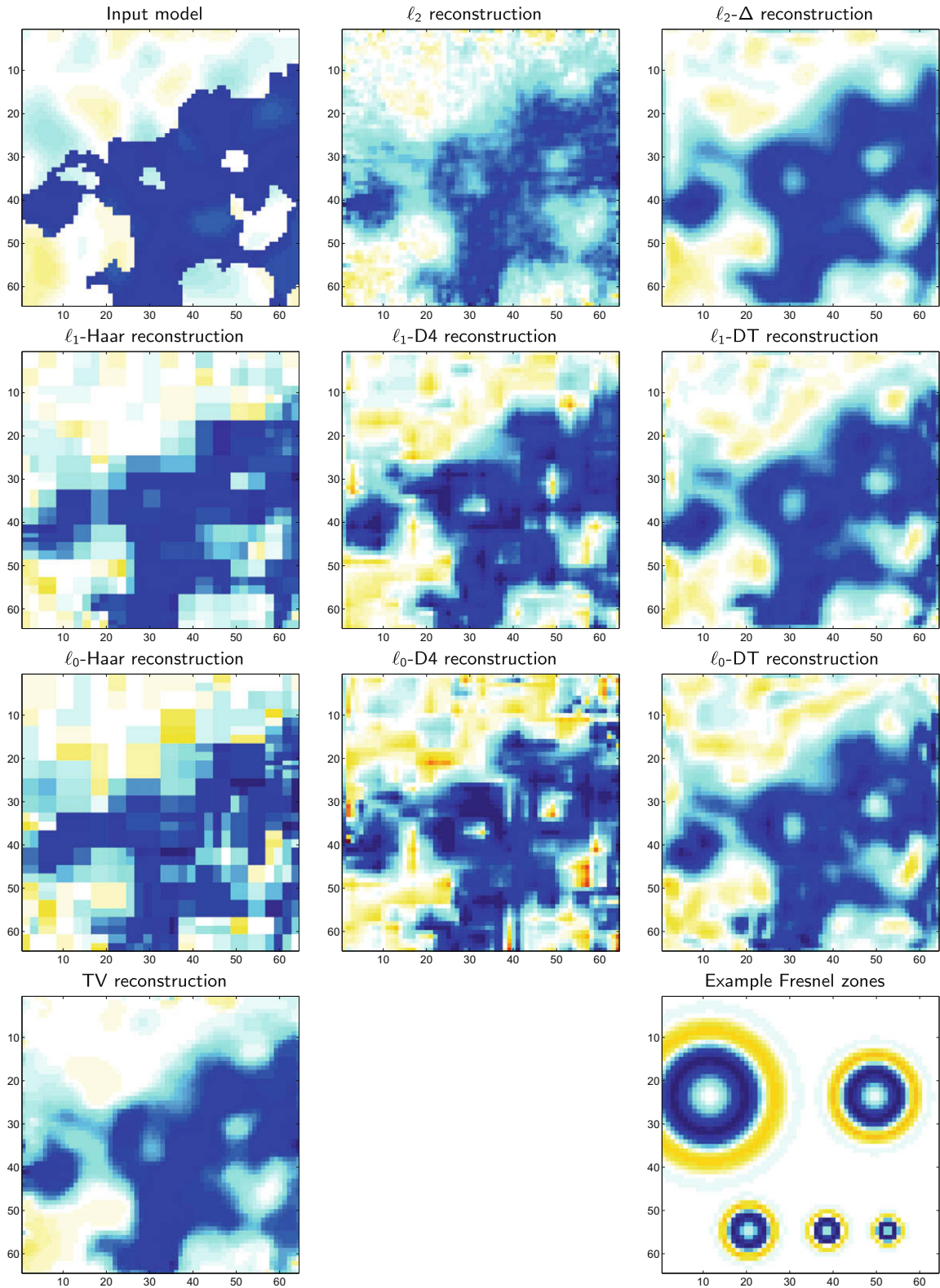
Fig. 6. Horizontal slice number 25 of the salt dome model and its various reconstructions using all 24,000 kernels. The TV, $\ell_2 - \Delta$, ℓ_1-DT and ℓ_0-DT methods introduce the most smoothing. The ℓ_1 and ℓ_0 Haar reconstructions have the least smoothing but are very blocky. A good compromise, in this respect, may be found in the ℓ_1-D4 reconstruction. The ℓ_0-D4 reconstruction is less appealing. The bottom right figure shows five representative kernel cross sections for different frequencies. These cross sections are taken perpendicular to and in the middle of the central rays , and give the reader an idea of the size of the kernels relative to the structure in the model and of the relative Fresnel zone widths of the different kernels.
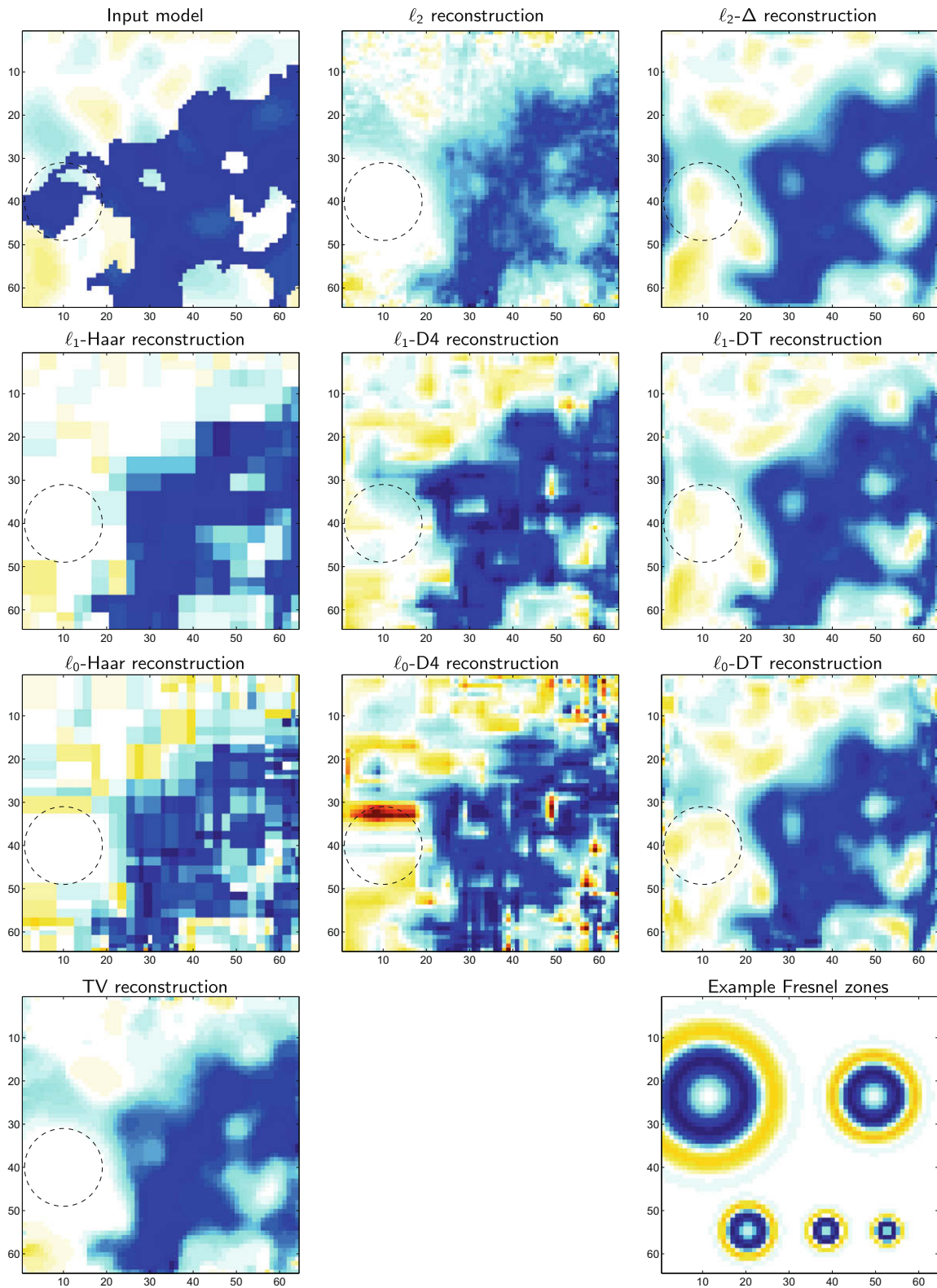
Fig. 7. Horizontal slice number 25 of the salt dome model and its reconstructions using only 20,000 kernels and thus non-uniform coverage. The area of the input model that lies in the region that is not covered by any kernel in this slice (indicated by a dashed circle), is not well reconstructed. Different methods compensate for the missing information in different ways. The ℓ_0-D4 reconstruction shows a distinctive artifact. As in Fig. 6, the bottom right pane shows five representative kernel cross sections.

lets, that has a relative reconstruction error almost as bad as obtained using the simple ℓ_2 penalized method. However, the "top three" methods (TV, ℓ_1-DT, $\ell_2 - \Delta$) produce much smoother models than the input model. In case the correct sharpness of features is a desirable characteristic of the solution, the ℓ_1 reconstructions with Haar or D4 wavelets are more faithful to the input data. In this case one may well prefer D4 over Haar to avoid the rather blocky nature of the shapes. The qualitative differences with the noisy ℓ_2 reconstruction are obvious. The ℓ_0 reconstructions that were obtained using iterative hard-thresholding, are less appealing. Numerically the ℓ_0-Haar and ℓ_0-D4 perform worst of all reconstructions, whereas ℓ_0-DT comes in fifth.

The reconstructions with only 20,000 kernels are shown in Fig. 7 (a single horizontal slice) and Fig. A.5 in the Supplementary material (all horizontal slices). The most interesting comparisons are again done visually. The lack of data coverage affects most strongly the areas around voxel (10, 40) in slice 25 (Fig. 7) and the lower left corner of slices 13–44 (third and fourth row in Fig. A.5 especially). Not surprisingly, none of the algorithms is able to 'recreate' the model where there are no data at all. But close inspection of the model near the edge of the region affected by the data gap shows that the Haar and D4 wavelets produce the model that is least contaminated by smoothing effects beyond the gap, with D4 occasionally trying to correctly 'fill in'. The ℓ_0-D4 reconstruction creates a distinctive artifact in this area. We speculate that this is caused by the non-convex nature of the ℓ_0 problem (19).

4.3. Computational aspects

Apart from the aspect of the visual reconstruction quality it is also important to compare reconstruction times. The four numerical algorithms that were used – conjugate gradient, fast iterative soft and hard thresholding, and reweighed conjugate gradient – all need one application of $\mathbf{A}$ and one application of $\mathbf{A}^T$ per iteration step. These dominate the other, but much faster, operations such as addition of vectors and vector norms, that are also present in each iteration step. The forward and inverse wavelet transforms that are used in some of the methods (via the Fast Wavelet Transform algorithm) also take a negligible time compared to an application of $\mathbf{A}$ and $\mathbf{A}^T$. It follows that it is sufficient to compare the number of iterations when we evaluate the efficiency of different reconstructions.

The number of iterations and corresponding relative reconstruction errors $\|\mathbf{m} - \mathbf{m}^{\text{input}}\|/\|\mathbf{m}^{\text{input}}\|$ are given in Table 2. In all cases, the iterative reconstruction algorithms were started from $\mathbf{w}^{(0)} = \mathbf{0}$ or $\mathbf{m}^{(0)} = \mathbf{0}$.

For the checkerboard model the data in Table 2 show that the ℓ_1 and ℓ_0 methods do extraordinarily well, with a mean square error far below what could be expected based on the data noise level of 10%. In the same sense, the total variation minimization and the Laplacian penalization perform somewhat better than the simple ℓ_2 method; a large number of simulations with different noise realizations would be necessary to verify whether this is a statistically significant difference. In case of the checkerboard reconstructions, only 100 iterations were performed for each method. This shows that the ℓ_1 and ℓ_0 methods can be very successful if the sought after model is very sparse in the basis used, even with a limited number of iterations. We have also verified that the functionals (4), (5), (10), (19) and (22) remain almost constant after this number of iterations, as did the relative distance to the input model.

In case of the salt dome reconstructions, the ℓ_0-Haar, ℓ_0-D4 and the simple ℓ_2 method do worst (in terms of reconstruction error) closely followed by the ℓ_1-Haar method. The latter is due to the inappropriateness of the Haar basis to represent the salt dome model in a sparse fashion. The other three methods (ℓ_1-D4, ℓ_1-DT and TV) do about equally as the $\ell_2 - \Delta$ method for the salt dome reconstruction with 24,000 data.

To gauge wether there is a significant difference in reconstruction error between the TV, ℓ_1-DT and $\ell_1 - \Delta$ methods (and possibly the ℓ_0-DT technique), one would also need to repeat the numerical experiment with many noise realizations.

The nonlinear reconstructions with 24,000 data were done with 1000 iterations. We used formula (15) to derive a rough upper bound on the relative error remaining in the functional (10) w.r.t to the minimum after this number of iterations:

$$\frac{F(\mathbf{w}^{(n)}) - F(\bar{\mathbf{w}})}{F(\mathbf{w}^{(n)})} \leqslant \frac{4\|\mathbf{w}^{(0)} - \bar{\mathbf{w}}\|^2}{\alpha(n+1)^2 F(\mathbf{w}^{(n)})} \lesssim 10^{-3}, \tag{26}$$

where we used $\mathbf{w}^{(0)} = \mathbf{0}$ and approximated $\|\bar{\mathbf{w}}\|$ by $\|\mathbf{w}^{(n)}\|$. In other words, the calculated value of the minimum of the functional is accurate up to three decimal places (this bound is valid for the three wavelet families). A simple plot of $F(\mathbf{w}^n)$ as a function of n also reveals that the functional is virtually constant after 1000 iterations, a conclusion which also holds for the TV method, the ℓ_0 method and for the ℓ_2 methods (after 100 iterations).

The corresponding reconstructions with 20,000 data were done with only 100 iterations. In this way we demonstrate that a reasonable result can already be obtained without an excessively long computation time. This is evident by comparing Figs. 6 and 7 (or Figs. A.4 and A.5 in the Supplementary material): apart from the unresolved region near $(0.24, -0.7, -0.23)$ the reconstructions are pairwise almost identical, despite the significant differences in number of iterations. In other words, the ℓ_1 and TV algorithms already succeed, after a small number of iterations, in producing qualitatively quite characteristic reconstructions.

In case of the ℓ_0 reconstructions the differences (far from the unresolved region) are somewhat larger, we believe, because the ℓ_0 method only finds a local minimum of (19).

As a result of the thresholding, the ℓ_0 and ℓ_1 algorithms provides sparse models at every iteration step (not just in the limit $n \rightarrow \infty$). In other words it is not necessary, or desirable, to run the FISTA/Nesterov style algorithm for a very long time. Even after a small number of iteration, they will provide a sparse model that fits the data to within its error bars.

5. Conclusions

In Paper I we showed how a large scale anomaly could be reconstituted even where it was ill resolved because of the selective nature of the wavelet coefficients and the ℓ_1 criterion: one wavelet coefficient reconstituting a large, circular, anomaly gave a better optimization than a couple of coefficients reconstituting only the resolved part. With the results of the much more complex salt dome model at hand, we must now conclude that this probably represents more a (lucky) exception than a rule. There is no magical solution for the absence of data.

For the checkerboard reconstructions, the ℓ_1 method with Haar wavelets is able to do very well – much better than could be expected based on the data themselves – because the Haar wavelets are very efficient in representing *this particular* checkerboard pattern in a sparse way. The success of the ℓ_1 method thus depends heavily on the choice of a suitable basis. For realistic models it is much more difficult to find a good – sparsifying – basis, and the reconstruction errors will be much larger. For the 3D salt dome reconstruction, one could argue that the ℓ_1-DT method does well because it has good directional sensitivity and is therefore able to adapt to the "curvy" nature of the outline of the salt body, as opposed to ℓ_1-Haar and ℓ_1-D4 methods. The $\ell_2 - \Delta$ method does well because the Gaussian background that is present in the model is smooth 'noise' and this is exactly the prior information put into the minimization criterion. The TV method does well as the main part of the salt dome model is roughly piecewise constant and TV favors that. The ℓ_0 methods do not perform particularly well, both from a quantitative as a qualitative side.

The wavelets, however, do have the distinctive quality of retaining sharp features even when regularizing by penalizing highly oscillatory models. If the preservation of sharp boundaries is not as important as the correct estimation of amplitudes, the smoothed solution, using the ℓ_2-Δ method, is to be preferred as it is fully linear and efficient to solve with conjugate gradients. Methods using wavelets with small support, however, are able to retain sharp features, despite their regularization effect that penalizes highly oscillatory models. These methods are thus preferable when edges are important; our preference would go to the ℓ_1-D4 algorithm which gives less blocky solutions than ℓ_1-Haar. In no cases should one use simple norm damping (ℓ_2 method). Without imposing additional smoothing, i.e. while still allowing for sharp transitions, the ℓ_1 methods yield models which do not show signs of noise.

The ℓ_0 methods, which use hard-thresholding of wavelet coefficients rather than soft-thresholding, cannot outperform the ℓ_1 methods. In some cases they appear to produce severe artifacts. Another reason not to favor ℓ_0 penalties is that they only produce a local minimum of the functional (19). This may lead to larger variability in the reconstructions (depending on the starting point of the iteration). There is currently no proven technique to tackle the minimization more efficiently that algorithm (17). The (unproven) method (20) proposed in this paper is, as far as the authors can tell, new.

We conclude that using hard-thresholding is less appealing than using soft-thresholding of wavelet coefficients: the mathematical theory is less developed, the hard-thresholded reconstruction may exhibit significant artifacts and the reconstructions are not better than the ones obtained with soft-thresholding (ℓ_1 method).

Speedwise the nonlinear methods cannot do better than the conjugate gradient algorithm for the ℓ_2 methods. Many applied mathematics groups [40,31,41,42,25,43,44] are currently working on speeding up the iterative soft-thresholding algorithm (11), but it is still at least as time-consuming to use the ℓ_1 norm as it is to use the ℓ_2 norm for penalization, especially for severely ill-conditioned matrices and low noise conditions [45].

In case the data is heavily contaminated by noise, it follows from relation (24) that a large value of the penalty parameter μ must be chosen. In [45] it was demonstrated that many competing algorithms for minimizing an ℓ_1 penalized functional converge quickly in such a case. We therefore expect that such methods remain competitive with the traditional ℓ_2 smoothing methods in case of travel time seismic tomography where the data noise level may reach 50%.

Based on the results in this paper, we can conclude that the nonlinear methods offer a way to invert data and denoise the resulting model in a single procedure without necessarily smoothing the model too much. The two salt dome examples also show that a good reconstruction, clearly showing the characteristic effects of the penalizations used, is still possible with a very limited number of iterations: This is a consequence of the FISTA algorithm producing sparse models at every iteration step. Sparse models can therefore be constructed with few iterations and little computer time (see [46] for a discussion of the number of iterations used as a regularization parameter).

As an alternative to D4 or complex DT wavelets one could consider using curvelets or shearlets, as they are naturally designed to sparsely represent singularities along smooth curves, such as, e.g. the sediment salt interface in our model. In this work we have not studied how the different regularization methods behave in conjunction with these particular choices of dictionaries.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.jcp.2009.10.020.

References

[1] G. Backus, J. Gilbert, Numerical applications of a formalism for geophysical inverse problems, Geophysical Journal of the Royal Astronomical Society 13 (1967) 247–276.
[2] G. Backus, J. Gilbert, Uniqueness on the inversion of inaccurate gross Earth data, Philosophical Transactions of the Royal Society of London A266 (1970) 123–192.
[3] I. Loris, G. Nolet, I. Daubechies, F.A. Dahlen, Tomographic inversion using ℓ_1-norm regularization of wavelet coefficients, Geophysical Journal International 170 (1) (2007) 359–370, doi:10.1111/j.1365-246X.2007.03409.x.
[4] T. Blumensath, M. Davies, Iterative thresholding for sparse approximations, Journal of Fourier Analysis and Applications 14 (5) (2008) 629–654, doi:10.1007/s00041-008-9035-z.
[5] T. Blumensath, M.E. Davies, Iterative hard thresholding for compressed sensing, Applied and Computational Harmonic Analysis 27 (3) (2009) 265–274, doi:10.1016/j.acha.2009.04.002.
[6] S. Chevrot, L. Zhao, Multiscale finite frequency Rayleigh wave tomography of the Kaapvaal craton, Geophysical Journal International 169 (2007) 201–215, doi:10.1111/j.1365-246X.2006.03289.x.
[7] J. Scales, A. Gersztenkorn, S. Treitel, Fast ℓ_p solutions of large sparse linear systems: application to seismic travel time tomography, Journal of Computational Physics 75 (1988) 314–333.
[8] G. Nolet, Seismic wave propagation and seismic tomography, in: G. Nolet (Ed.), Seismic Tomography, Reidel, Dordrecht, 1987, pp. 1–23.
[9] H.L. Taylor, S.C. Banks, J.F. McCoy, Deconvolution with the ℓ_1 norm, Geophysics 44 (1979) 39–52.
[10] D. Oldenburg, T. Scheuer, S. Levy, Recovery of the acoustic impedance from reflection seismograms, Geophysics 48 (1983) 1318–1337.
[11] F. Santosa, W.W. Symes, Linear inversion of band-limited reflection seismograms, SIAM Journal on Scientific and Statistical Computing 7 (1986) 1307–1330.
[12] M.D. Sacchi, T.J. Ulrych, C. Walker, Interpolation and extrapolation using a high-resolution discrete Fourier transform, IEEE Transactions on Signal Processing 46 (1998) 31–38.
[13] E.J. Candès, J.K. Romberg, T. Tao, Stable signal recovery from incomplete and inaccurate measurements, Communications on Pure and Applied Mathematics 59 (8) (2006) 1207–1223, doi:10.1002/cpa.20124.
[14] D. Donoho, Compressed sensing, IEEE Transactions on Information Theory 52 (4) (2006) 1289–1306, doi:10.1109/TIT.2006.871582.
[15] D.L. Donoho, For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution, Communications on Pure and Applied Mathematics 59 (2006) 797–829, doi:10.1002/cpa.20132.
[16] F.J. Herrmann, G. Hennenfent, Non-parametric seismic data recovery with curvelet frames, Geophysical Journal International 173 (1) (2008) 233–248, doi:10.1111/j.1365-246X.2007.03698.x.
[17] G. Hennenfent, E. van den Berg, M.P. Friedlander, F.J. Herrmann, New insights into one-norm solvers from the Pareto curve, Geophysics 73 (4) (2008) A23–A26, doi:10.1190/1.2944169.
[18] E. van den Berg, M.P. Friedlander, Probing the Pareto frontier for basis pursuit solutions, SIAM Journal on Scientific Computing 31 (2) (2008) 890–912, doi:10.1137/080714488.
[19] N. Favier, S. Chevrot, Sensitivity kernels for shear wave splitting in transverse isotropic media, Geophysical Journal International 153 (2003) 213–228, doi:10.1046/j.1365-246X.2003.01894.x.
[20] G. Nolet, A Breviary of Seismic Tomography, Cambridge University Press, 2008.
[21] F.A. Dahlen, Resolution limit of traveltime tomography, Geophysical Journal International 157 (1) (2004) 315–331, doi:10.1111/j.1365-246X.2004.02214.x.
[22] A. Tikhonov, Solution of incorrectly formulated problems and the regularization method, Soviet Mathematics – Doklady 4 (1963) 1035–1038.
[23] I. Daubechies, M. Defrise, C. De Mol, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint, Communications on Pure and Applied Mathematics 57 (11) (2004) 1413–1457, doi:10.1002/cpa.20042.
[24] I. Daubechies, Ten Lectures on Wavelets, Society for Industrial and Applied Mathematics, 1992.
[25] A. Beck, M. Teboulle, A fast iterative shrinkage-threshold algorithm for linear inverse problems, SIAM Journal on Imaging Sciences 2 (2009) 183–202, doi:10.1137/080716542.
[26] Y.E. Nesterov, A method for solving the convex programming problem with convergence rate $\mathcal{O}(1/k^2)$, Doklady Academii Nauk SSSR 269 (1983) 543–547 (in Russian).
[27] Y.E. Nesterov, A method for solving a convex programming problem with convergence rate $\mathcal{O}(1/k^2)$, Soviet Mathematics – Doklady 27 (1983) 372–376.
[28] Y. Nesterov, Introductory Lectures on Convex Optimization, Kluwer Academic Publishers, 2004.
[29] J. Oliveira, J. Bioucas-Dias, M. Figueiredo, Adaptive total variation image deblurring: a majorization–minimization approach, Signal Processing 89 (9) (2009) 1683–1693, doi:10.1016/j.sigpro.2009.03.018.
[30] J. Bect, L. Blanc-Féraud, G. Aubert, A. Chambolle, A ℓ^1-unified variational framework for image restoration, in: T. Pajdla, J. Matas (Eds.), Proceedings of the Eighth European Conference on Computer Vision, vol. IV, Springer Verlag, 2004, pp. 1–13.
[31] J. Bioucas-Dias, M. Figueiredo, A new TwIST: two-step iterative shrinkage/thresholding algorithms for image restoration, IEEE Transactions on Image processing 16 (2007) 2980–2991, doi:10.1109/TIP.2007.909319.
[32] A. Haar, Zur Theorie der orthogonalen Funktionensysteme, Mathematische Annalen 69 (1910) 331–371.
[33] A. Cohen, I. Daubechies, B. Jawerth, P. Vial, Multiresolution analysis, wavelets and fast algorithms on the interval, Comptes Rendus de l'Académie des Sciences Paris 316 (1992) 417–421.
[34] A. Cohen, I. Daubechies, P. Vial, Wavelets on the interval and fast wavelet transforms, Journal for Applied and Computational Harmonic Analysis 1 (1993) 54–81, doi:10.1006/acha.1993.1005.
[35] N. Kingsbury, Image processing with complex wavelets, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 357 (1760) (1999) 2543–2560, doi:10.1098/rsta.1999.0447.
[36] N. Kingsbury, Complex wavelets for shift invariant analysis and filtering of signals, Applied and Computational Harmonic Analysis 10 (2001) 234–253, doi:10.1006/acha.2000.0343.
[37] D. Labate, W. Lim, G. Kutyniok, G. Weiss, Sparse multidimensional representation using shearlets, in: SPIE Proc. 5914, SPIE, Bellingham, WA, 2005, pp. 254–262, Wavelets XI (San Diego, CA, 2005).
[38] G. Kutyniok, D. Labate, Construction of regular and irregular shearlets, Journal of Wavelet Theory and Applications 1 (2007) 1–10.
[39] E.J. Candes, L. Demanet, D.L.D.L. Ying, Fast discrete curvelet transforms, SIAM Multiscale Modeling and Simulation 5 (3) (2006) 861–899, doi:10.1137/05064182X.

[40] M.A.T. Figueiredo, R.D. Nowak, S.J. Wright, Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems, IEEE Journal of Selected Topics in Signal Processing (Special Issue on Convex Optimization Methods for Signal Processing) 1 (2007) 586–598. doi:10.1109/JSTSP.2007.910281.

[41] M. Elad, B. Matalon, M. Zibulevsky, Coordinate and subspace optimization methods for linear least squares with non-quadratic regularization, Applied and Computational Harmonic Analysis 23 (3) (2007) 346–367, doi:10.1016/j.acha.2007.02.002.

[42] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, D. Gorinevsky, An interior-point method for large-scale ℓ_1-regularized least squares, IEEE Journal of Selected Topics in Signal Processing 1 (4) (2007) 606–617, doi:10.1109/JSTSP.2007.910971.

[43] E. Hale, W. Yin, Y. Zhang, Fixed-point continuation for ℓ_1-minimization: methodology and convergence, Siam Journal on Optimization 19 (3) (2008) 1107–1130, doi:10.1137/070698920.

[44] W. Yin, S. Osher, D. Goldfarb, J. Darbon, Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing, SIAM Journal on Imaging Sciences 1 (1) (2008) 143–168, doi:10.1137/070703983.

[45] I. Loris, On the performance of algorithms for the minimization of ℓ_1-penalized functionals, Inverse Problems 25 (2009) 035008, doi:10.1088/0266-5611/25/3/035008.

[46] M. Defrise, C. De Mol, A note on stopping rules for iterative regularization methods and filtered SVD, in: P.C. Sabatier (Ed.), Inverse Problems: An interdisciplinary Study, Academic Press, 1987, pp. 261–268.