Domain decomposition in shape optimization for segmenting salt bodies

Taylor Dahlke, Biondo Biondi and Robert Clapp

ABSTRACT

Level set methods can provide a sharp interpretation of the salt body by defining the boundary as an isocontour of a higher dimensional implicit representation, and then evolving that surface to minimize the Full Waveform Inversion objective function. I propose to take advantage of the benefits that the shape optimization approach has to offer. First, because the implicit surface update gradient is based on the tomographic update gradient, there is potential to utilize it to update the background velocity concurrently with the salt boundary. Second, we can decompose the update gradient into separate partitions with individual scaling parameters in order to better avoid local minima in our search and more effectively converge on the true model. Using a shape optimization approach on synthetic examples, we can achieve reasonable convergence both in terms of the residual L2 norm, as well as the evolution of the salt boundary and background velocity towards the true model, demonstrating the feasibility of this approach. Ultimately, this method can be integrated into the processing work flow as a tool that provides improved building and refining of the velocity models used for imaging.

INTRODUCTION

Oil producing regions like the Gulf of Mexico and offshore western Africa are known to have geologically complex salt body formations which can cause difficulties in producing seismic imagery. The velocity of these salt bodies often contrasts sharply with that of the background sediment layers. An inaccurate interpretation of the salt boundaries can cause significant errors in the velocity estimation process, because the formations themselves can act as lenses which focus or disperse seismic energy, influencing tomography. This can subsequently impact the imaging results that rely on accurate velocity models. Salt bodies can act as seals trapping hydrocarbons underneath, which are often the targets of imaging projects for reservior interpretation. For this reason the interpretation of salt body boundaries can also impact drilling and production activities.

Tomographic approaches to interpreting salt bodies can be less than effective, because the results tend to be too smooth to provide significantly accurate placement of the salt boundaries. Manual and semi-automatic picking of salt boundaries is a

common approach to interpreting the desired sharp delineations, but these methods can be time-consuming and tedious since expert input is necessary for either the actual picking, or the oversight and correction. Furthermore, once a model has been produced, it must be used to generate an image, and then be refined as necessary. A robust method for further automating the salt interpretation procedure would greatly alleviate this bottleneck.

Some previous approaches to performing salt body segmentation use a shape optimization approach for identifying salt body boundaries (Guo and de Hoop, 2013; Lewis et al., 2012). The boundaries of a salt body can be represented as the zero-isocontour of a higher dimensional surface (for example, a 2D boundary as a contour of a 3D surface). A gradient can be derived to evolve this shape / isosurface according to the Full Waveform Inversion (FWI) objective function. Unlike the smooth boundaries produced by tomographic approaches, the isocontour resulting from the shape optimization provides a sharp boundary, which is a more appropriate way to classify most salt-sediment interfaces. Guo and de Hoop (2013) utilize this approach using a frequency domain forward wave operator to evolve a salt boundary and velocity model. Their approach creates and applies a global gradient update, which can create problems updating the base-of-salt (BOS) once the top-of-salt (TOS) has gotten close to convergence. This issue has been observed in recent work (Guo and de Hoop, 2013), and is an inherent problem with the global gradient update approach.

To address the problem with global updating, we introduce domain decomposition for the salt boundary update gradient by splitting it; demonstrated here using "upper" and "lower" partitions. This introduces an additional degree of freedom to our parameter searching so that local minima can be avoided. Using a shape optimization approach with time domain forward wave-propagation, we take advantage of using a continuous range of frequencies (rather than discrete frequencies) in each iteration, which allows for sharper delineation of the boundary. Further, we take advantage of the fact that our boundary update gradient is based on the tomographic update gradient, and make updates to both concurrently with the use of scaling parameters. In theory, this algorithm has the potential to be more efficient than an alternating update approach.

In this paper we will begin by discussing the fundamentals of the level set method and its key properties, followed by the derivation of the boundary update gradient. Next we will describe and demonstrate the general algorithm, and discuss the assumptions and fundamental limitations of this approach. Following, we will introduce the domain decomposition method, and demonstrate the improved results that this approach offers.

THEORY

While it may seem counter-intuitive to add an extra dimension to our problem, by doing so, we gain the advantage of easily merging/separating bodies as the evolu-

tion proceeds, as well as the ability to handle sharp corners and cusps in the lower-dimensional (2D) plane on which the boundary exists.

Osher and Sethian (1988) and Burger (2003) describe the level set of ϕ that represents the salt body boundary as

$$\phi(x_\Gamma, \tau) = 0,$$

where x_τ is the spatial domain, and τ is the iteration count. By taking the derivative of this equation with respect to τ (to find the $\delta\phi$ between iterations), applying the chain rule, and re-arranging terms we can get:

$$\frac{\partial \phi}{\partial \tau} = -V(x_\Gamma, \tau) \left|\nabla \phi\right|. \tag{1}$$

The scalar speed term $V(x_\Gamma, \tau)$ describes the magnitude of the variation of ϕ that is normal to the boundary Γ. It determines the evolution of the implicit surface, and ultimately the boundary implied by it. We derive this normal velocity such that the FWI objective function is minimized

$$\min \left\|F(m) - d\right\|_2^2, \tag{2}$$

where $F(\cdot)$ is the forward wavefield modeling operator, m is the velocity model, and d is the observed data.

Calculus of variations

The shape derivative we use is based on a formal calculus of variations outlined in Santosa (1996). The objective is to define the variation of the model m with respect to the boundary variation (represented implicitly by the surface, ϕ).

We begin by considering an inner product of velocity model perturbation δm with a test function $f(x)$. Formally, this can be written as,

$$\langle \delta m, f(x) \rangle = \int_{\mathbb{R}^2} \delta m(x) f(x) dx = \int_{\partial\Omega} \delta m(x) f(x) dx. \tag{3}$$

Because the $\delta m(x)$ term equals zero in $\mathbb{R}^2 \setminus \partial\Omega$, it does not contribute to the overall inner product when integrating over that domain; therefore, we only integrate over $\partial\Omega$ where $\delta m(x)$ is non-zero. We know that $\delta m(x)$ will be $\pm(m_{\text{int}} - m_{\text{ext}})$, depending on the relative values of m_{int} and m_{ext} or the direction of the normal vector $\vec{n}$. We only care about the component of δx_Γ that occurs in the normal direction, because

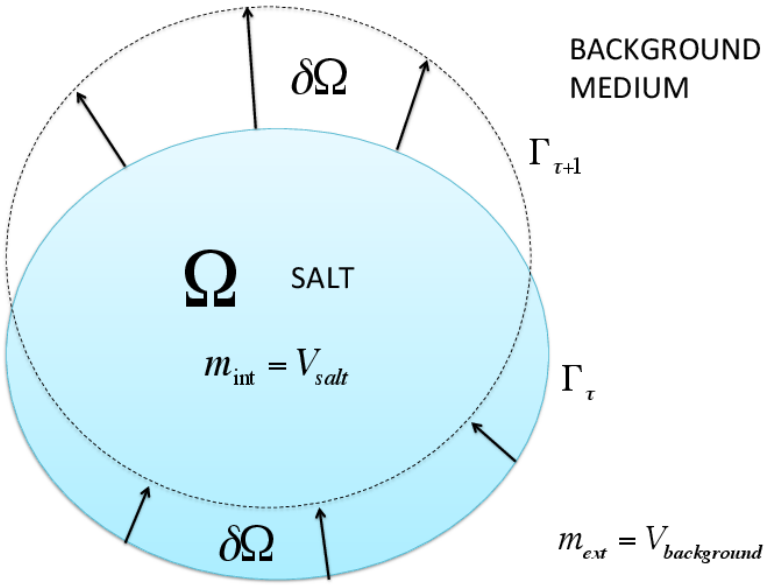

Figure 1: The geometry of the curve $\{x_\Gamma : \phi = 0\}$ for a variation $\delta\phi(x)$ for an evolution step τ. [**NR**]

a tangential variation of x_Γ does not affect m or ϕ. Furthermore, because δx_Γ is infinitesimal, we can replace dx with $\delta\vec{x}_\Gamma \cdot \vec{n}$ and simplify Equation 3 into

$$\langle \delta m, f(x) \rangle = \int_{\partial\Omega} (m_{\text{int}} - m_{\text{ext}}) \delta\vec{x}_\Gamma \cdot \vec{n} f(x) ds(x), \tag{4}$$

where $ds(x)$ is the incremental arc length along the boundary Γ. We can think of $\delta\vec{x}_\Gamma \cdot \vec{n} ds(x)$ as roughly the incremental area over which m varies at x.

We can identify δm from Equation 4. It can be considered a measure over $\partial\Omega$:

$$\delta m = (m_{\text{int}} - m_{\text{ext}}) \delta\vec{x}_\Gamma \cdot \vec{n} \,|_{x \in \partial\Omega}\,. \tag{5}$$

We remember that in the previous section we stated the goal of this derivation as being a solution of the scalar velocity function $V(x_\Gamma, \tau)$, such that the objective function is minimized. We recognize that the normal component of the variation δx_Γ satisfies:

$$\delta\vec{x}_\Gamma \cdot \vec{n} = V(x_\Gamma, \tau). \tag{6}$$

We can use the shape derivative formulation described in Santosa (1996) to find a

$V(x_\Gamma, \tau)$ that minimizes the FWI objective function (Equation 2) that we insert into Equation 1 to get a final implicit surface update gradient of:

$$\frac{\partial \phi}{\partial \tau} = (m_{\text{int}} - m_{\text{ext}}) \frac{\partial F}{\partial m} \left| \vec{\nabla} \phi \right|. \tag{7}$$

The adjoint state method as described in Plessix (2006) is used to derive $\frac{\partial F}{\partial m}$, which can be shown to be equivalent to $F(m)^T(A(m) - d)$. Because our case uses the FWI objective function (Equation 2), this term can be interpreted as least squares migration, more specifically as reverse time migration (RTM). This result is formulated as:

$$\frac{\partial F(m)}{\partial m} = -\sum_s \int_0^T \int_{x \in \Gamma} h_s(x,t) \frac{\partial^2 u_s(x,t)}{\partial t^2} d\sigma dt, \tag{8}$$

where h_s is the backpropagated residual wavefield, and u_s is the source wavefield. This term by itself is the velocity model perturbation, which we can process and use to make tomographic updates as described in the following section.

General evolution algorithm

We begin with an initial background velocity, and a binary valued (-1, 1) function as the initial implicit surface ϕ. Since we assume a constant salt velocity, we use both of these inputs to create a full initial-guess velocity model m_o. Using this m_o, we forward model to get our d_{syn} and subsequently find our residual. The residual is used to calculate both a tomographic and a boundary update gradient, as described in the derivation section previously. We then do non-linear line searches for α and β in a manner that minimizes the FWI objective function. Afterwards we apply an explicit forward Euler scheme that updates the implicit surface (ϕ) and the background velocity V_{back}:

$$\phi^{j+1} = \phi^j + (\beta \frac{\partial \phi}{\partial j} + \mu G_{\text{reg}}) \tag{9}$$

$$V_{\text{back}}^{j+1} = V_{\text{back}}^j + \alpha \frac{\partial V_{\text{back}}}{\partial j}. \tag{10}$$

where β and α are the step sizes, j is the current iteration point, and μG_{reg} is the weighted regularization term (described subsequently). This work flow is graphically represented in Figure 2.

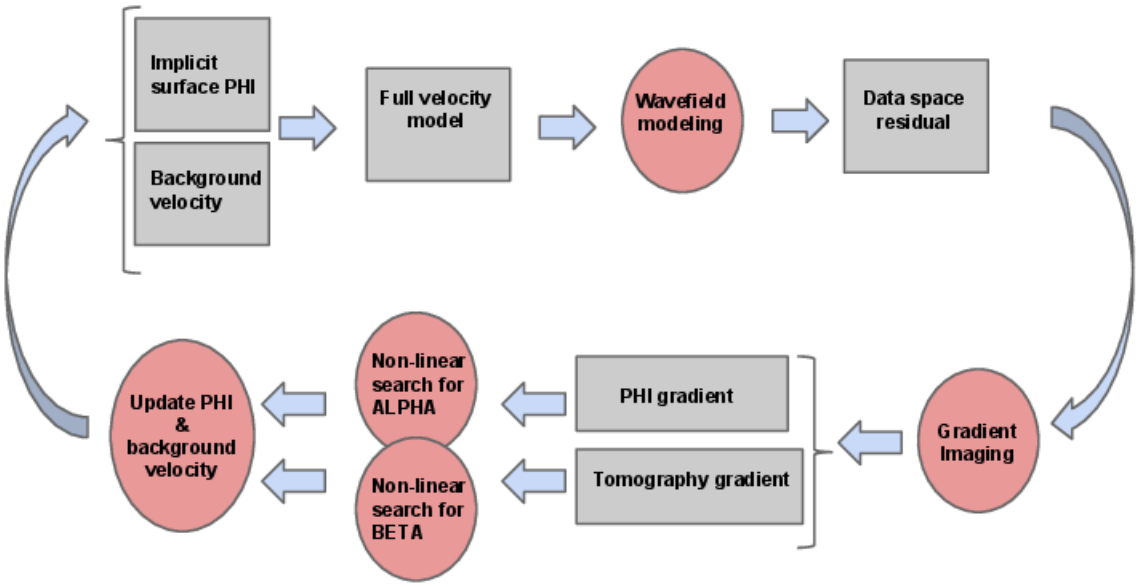

Figure 2: The general work flow used for shape optimization. [**NR**]

Implicit surface stability: CFL condition

As the implicit surface is evolved, it is important to maintain stability of the evolution. One relevant aspect of maintaining stability is keeping the implicit surface update step size (β) small enough to satisfy the Courant-Friedrich's-Levy (CFL) condition, which is stated by Chaudhury and Ramakrishnan (2007) (when applied to level set evolution) as being:

$$G_{\max} \cdot \beta \leq \min(h_x, h_y) \tag{11}$$

where h_x and h_y are the grid spacing in the x and y directions, and $G_{\max}$ is the maximum value of the update gradient.

Implicit surface stability: DRLSE regularization

It is well documented in level set literature that maintaining a smooth, differentiable implicit surface is important to maintaining stability as the surface evolves. Multiple approaches have been implemented towards this end, including freezing of the level set boundary and reinitializing the implicit surface to that of a signed distance function. This approach can be somewhat effective, but has the drawback of adding expense to the algorithm (since the boundary does not evolve during the iterations that reinitialization is occurring). Further, there is no clear approach for identifying how frequent the reinitialization should occur to maintain stability. For this reason,

the more effective approach of distance-regularized level set evolution (DRLSE) introduced by Li et al. (2010), is more commonly implemented in recent work. Unlike the reinitialization scheme, Li et al. (2010) show how DRLSE applies regularization updates at every iteration to maintain the implicit surface gradient as equaling one ($|\nabla\phi| = 1$). We calculate this DRLSE regularization term, which is then scaled and added to the boundary gradient before each update is applied. By driving the gradient of the implicit surface to equal one, we minimize irregularities and are able to continue evolution without having to reinitialize a signed-distance function to the salt boundary contour.

Scaling parameter optimization

As shown previously, the salt boundary gradient is based on the adjoint of the linearized-Born operator, which is the tomographic update gradient. Since the gradient for both a tomographic and boundary update are calculated in each step regardless, we attempt to take advantage of this by finding scaling parameters to apply to these gradient updates such that we minimize the residual space objective function 12. We use non-linear line searches to find α, then update the tomography, and immediately after perform a non-linear line search for β using the updated background velocity.

$$\min \|F(m(\gamma) - d_{\text{obs}})\| . \tag{12}$$

Where γ is a placeholder for either α or β.

APPLICATION

My demonstration of the shape optimization algorithm in both cases was performed on a 2D model, with the implicit surface evolved being a 3D surface. For the forward wave propagation, a wavelet with a 15.0 Hz central frequency was propagated using a time domain forward operator, so that continuous frequency information would be forward modeled in a single iteration.

Example: General algorithm

To implement the general algorithm, I used a model and acquisition that was symmetric on the horizontal (x) axis, with a salt model that is smaller than the true salt body. The background velocity had several layers, where the gradient in each layer was perturbed to be faster than actual, while preserving the reflector position (See Figure 3). The acquisition was composed of 11 shots and 75 receivers. The algorithm

does a reasonably good job at converging on the top of the salt body, but the updating along the bottom and sides has still not been able to converge to the true model correctly.

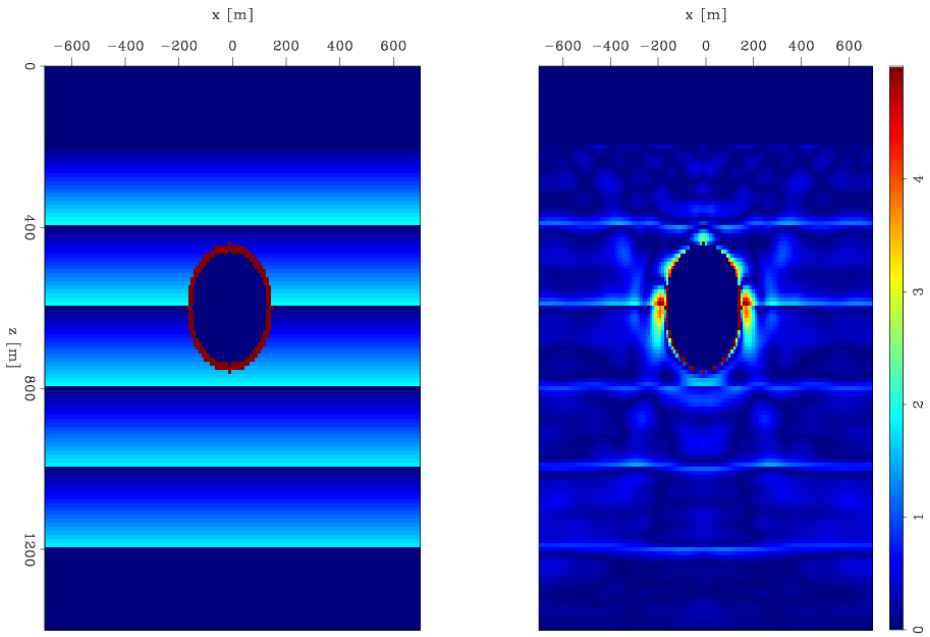

Figure 3: Percent velocity error for initial model (left), and for the model after 100 iterations (right) using general algorithm. [**CR**]

domain decomposition of boundary update gradient

One notices in Figure 3 that the convergence of the general algorithm appears to stall before convergence of the base of salt occurs, even after a significant number of iterations. The reason for this is that the more strongly illuminated top of salt (TOS) is the primary influence in the line search for β, used to apply the implicit surface update gradient. For this reason, even when the gradient update would push the base of salt (BOS) towards convergence, the updating on the TOS (which is unnecessary since the TOS has converged) has a saturating effect on the step length search, forcing the update towards a very small value.

The implicit surface update would be successful if the gradient did not show an update for the TOS after the TOS has converged. However, even when the TOS boundary and background velocity model are perfect, we will still realize an update for the TOS (see Figure 4).

The fundamental problem demonstrated in Figure 4 is that the reverse time migration (RTM) gradient contains information from both reflector position error as well as background velocity error. These can be difficult or impossible to separate. In this case, what could be correctly updated by changing the position of the bottom reflec-

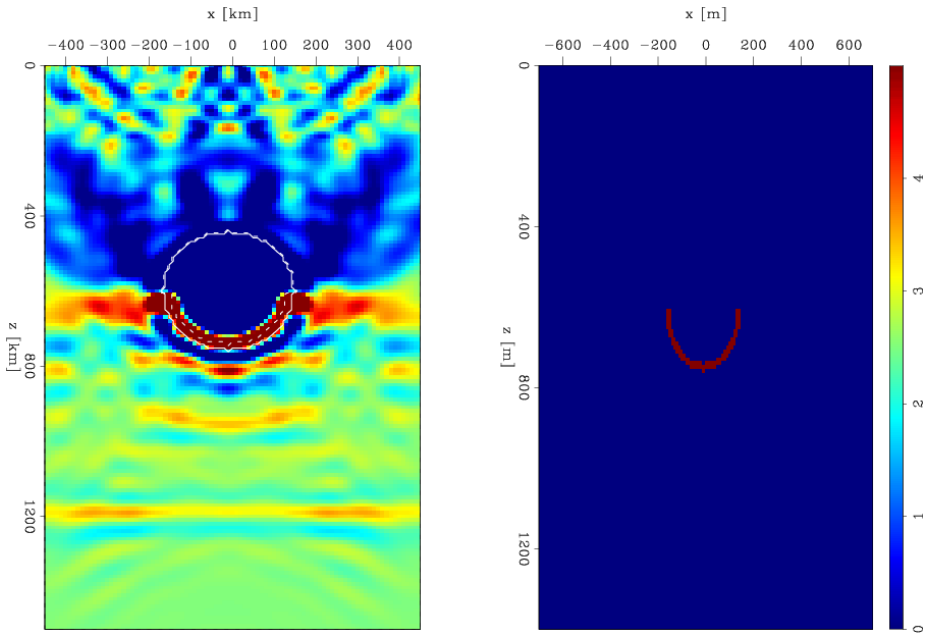

Figure 4: First gradient of FWI (left) for model with perfect TOS and background velocity. Velocity model perturbation (right). The FWI gradient wants to correct the residual by both increasing the velocity at the base of salt (red), while also reducing the velocity of the medium above it (blue). This RTM velocity updating maps to an implicit surface update that causes the TOS boundary to sink deeper in order to reach the same effect. [**CR**]

tor can also be corrected by making a velocity update above it. The RTM calculation always produces a gradient that applies both these effects. For this reason, while an increase/decrease of velocity from the raw RTM gradient is often correctly mapped to a reflector position change, the algorithm ultimately stalls at a local minima when illumination for a reflector (like the BOS) is sourced from ray paths that must first travel through another reflector edge of the same body being updated (like the TOS). Fundamentally, the update gradient is flawed in terms of directing convergence to the global solution of our objective function. The 'false' TOS updates consistently lead us to a local minima instead. This is because the TOS updates are always much stronger in the residual space, and thus our non-linear line search for a global scaling parameter will always prefer to correct the TOS, typically with a very small update step (when the TOS is near the true model). This causes a lack of update for the base of salt reflector. Figure 5 shows that even after a significant number of iterations, this approach cannot converge to the true model, and in fact, even becomes worse for some sections of the boundary.

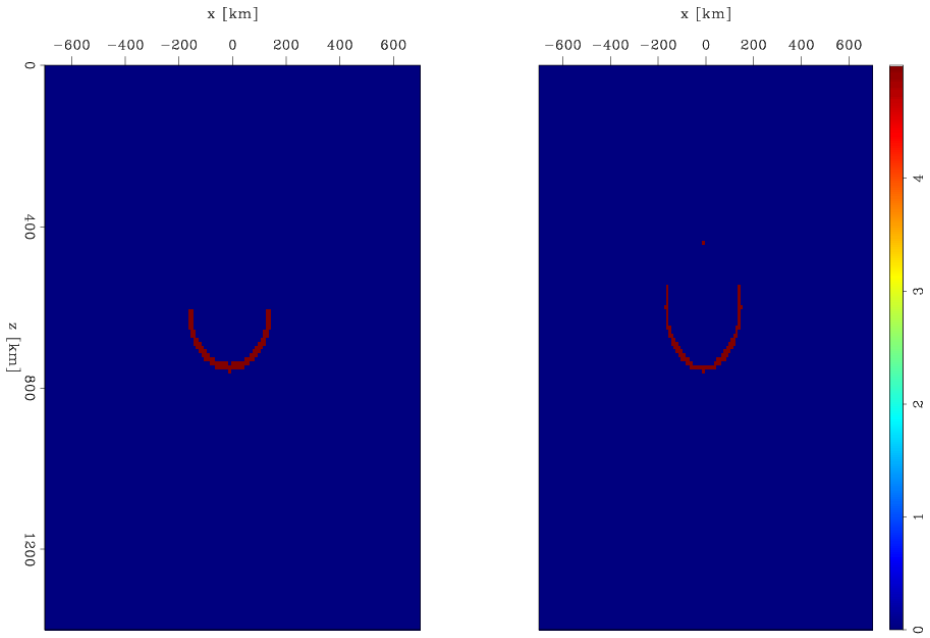

Figure 5: Velocity model error for starting model (left), and final model (right) after 200 iterations using general algorithm. [**CR**]

Examples: Partitioning approach

Any gradient that updates both the entry and exit reflectors for a given ray path will eventually meet the limitation described above, because the gradient couples both updates together. A method to de-couple the entry and exit reflectors is necessary (typically top and bottom when the acquisition is above the salt, as in most industry seismic) so that reasonable convergence can still be reached when spurious updating occurs in the upper reflector due to the mixing of tomography and reflector position.

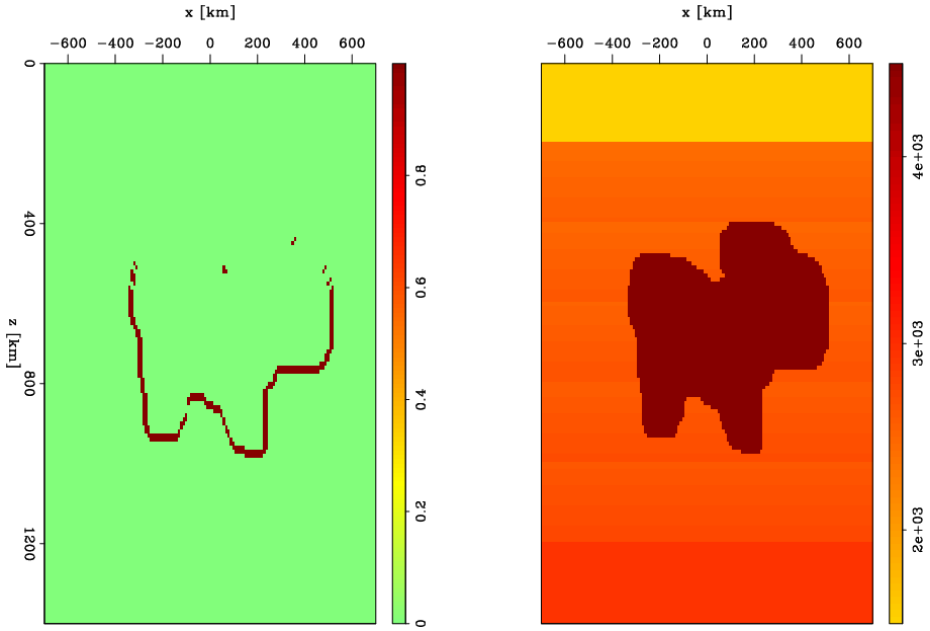

Figure 6: Example of weighting used for the 'bottom' ϕ gradient partition at first iteration (left), and velocity model that weighting is based on (right). [**CR**]

In order to decompose the ϕ update gradient into 'top' and 'bottom' components for the general case, we use an approach that makes use of straight rays by creating a mapping of the radial distance from each shot position. We take the gradient of this map in order to find the x and y gradient components of the radial vector field. We then find the x and y components of the normal vectors from the salt boundary. By taking the dot product of these straight ray and boundary normal vector fields, we generate a map that approximates the straight ray illumination of the body. However, more expensive techniques to produce ray path maps could be substituted to achieve better 'illumination' mapping. We perform these steps for each shot, and sum the dot-product fields to create a final cumulative map that is used for weighting on the ϕ gradient. Once this is done, thresholding is applied so that negative values in the weighting map are set to zero, while the positive values are set to 1.0. The inverse weighting map can be found by multiplying the original map by -1.0, and then thresholding as before. Figure 6 gives an example of what this boundary partitioning looks like.

This 'splitting' has the advantage of decomposing the boundary into sections that generally face away from or towards the acquisition line. Further partitioning of the boundary update can be done by setting thresholding to selectively partition the facing, steep, and shadowed reflecting edges.

Examples

Isolating from the effects that tomography updating might have, we perform an experiment where the algorithm does no tomography updating, and begins with a perfect background velocity model. Figure 7 shows that the difference between the two approaches is clear. After 25 iterations, the partitioning algorithm has already converged on the true model, while the global gradient approach has trouble, particularly on the base of salt.

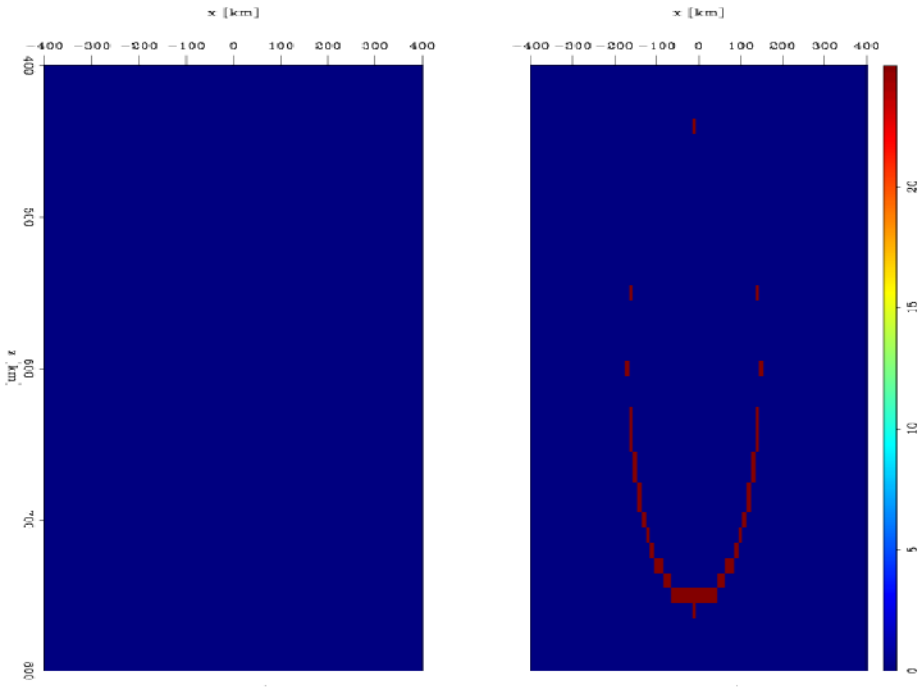

Figure 7: % velocity error at 25 iterations using domain decomposition algorithm (left) and general algorithm (right). Perfect background velocity model. [**CR**]

We test again with the same salt model, this time beginning with a perturbed background velocity model. We also have the algorithm perform tomography updates. Figure 8 shows that when we try this approach we get close convergence with the true model on both the top and base of salt.

We can compare the results of the algorithms shown in Figures 8 and 3 by plotting the differential error, as shown in Figure 9. From this we observe that while some of the tomographic updates are less accurate in the partitioning approach, the convergence on the salt model is significantly improved, especially for the base of salt and salt flanks. This further demonstrates that this domain decomposition approach can yield more accurate convergence, even when tomography is also concurrently being updated.

We next try the same comparison of the two algorithms on models with far more complex salt body geometry. Figure 10 also shows some slight improvements on the

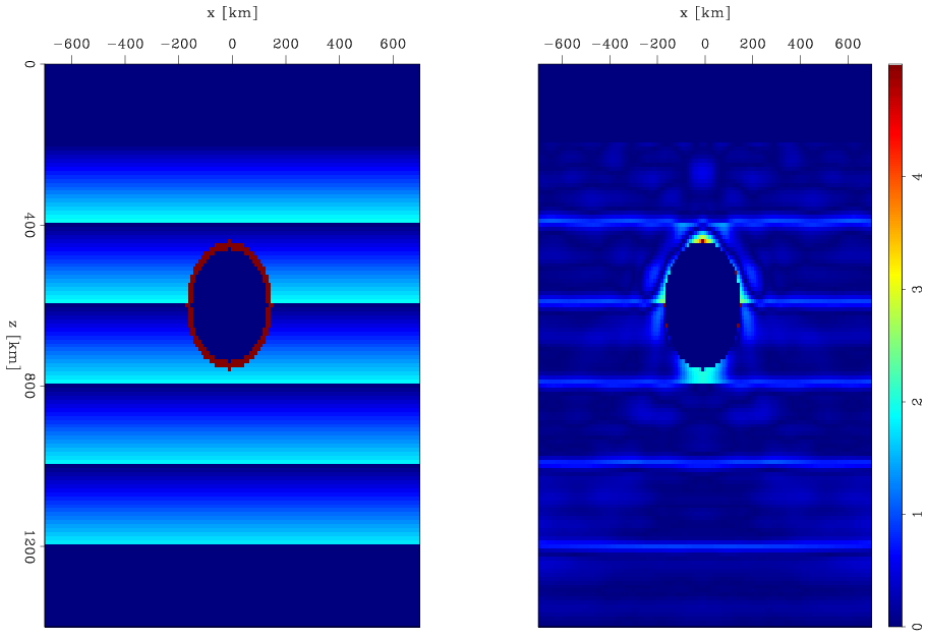

Figure 8: Initial model perturbation (left), and after 100 iterations (right) using split ϕ algorithm. [**CR**]

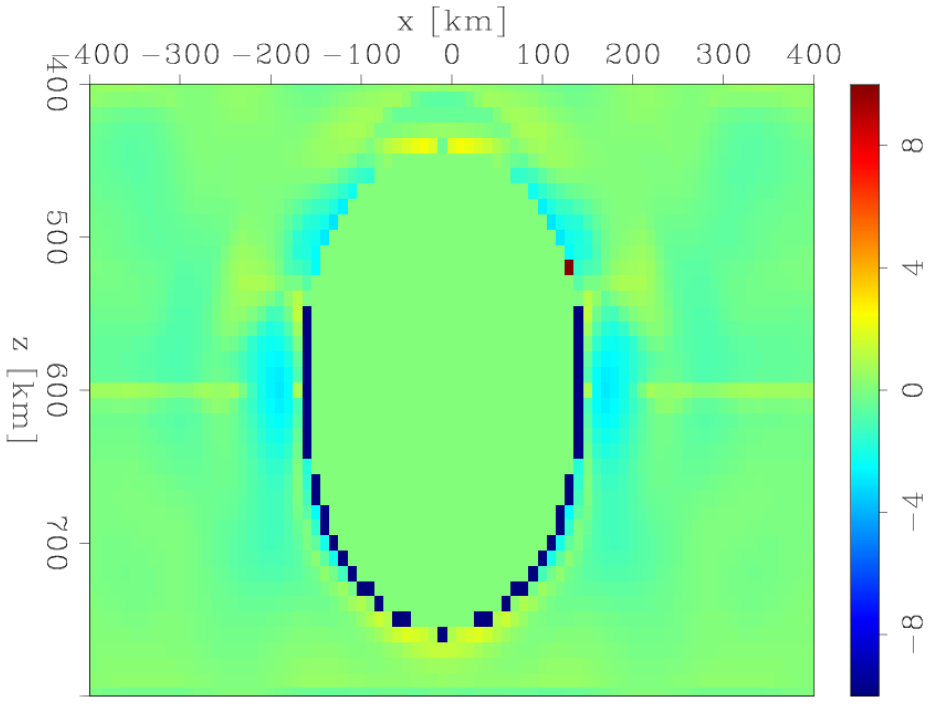

Figure 9: The differential error between the gradient partitioning approach and the general algorithms after 100 iterations, as expressed as a percentage error from the true model. Blue regions (negative differential) indicate areas where the domain partitioning approach performs better. Red (positive differential) indicates where it performs more poorly. [**CR**]

bottom of salt and even some of the tomography updates along the top of salt.

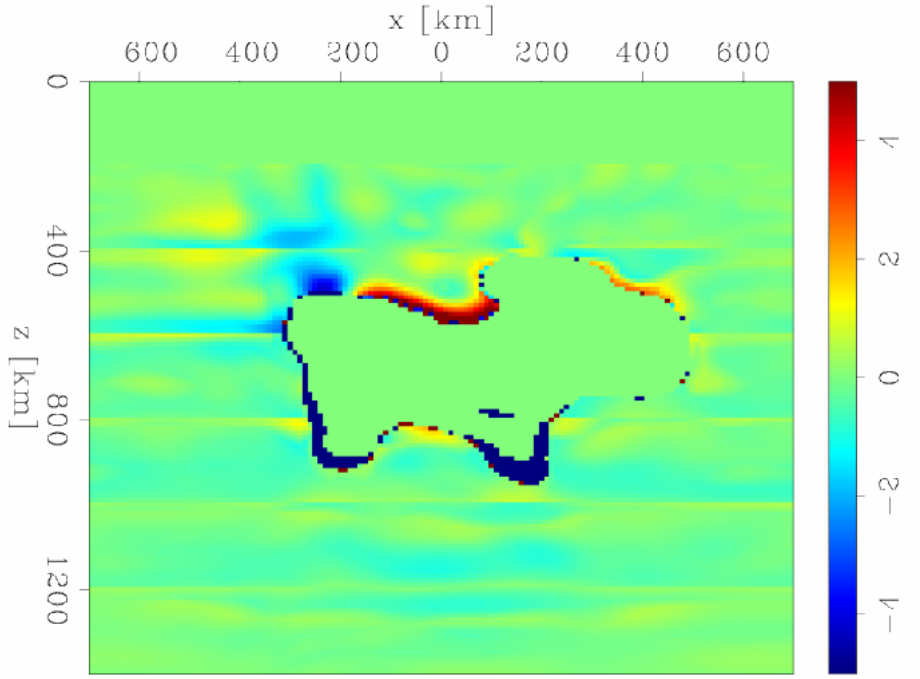

Figure 10: Velocity model difference between general and split ϕ methods, shown as a percentage error from the true model. Blue regions are where the domain partitioning approach performs better; red, more poorly. [**CR**]

Figure 12 demonstrates improved convergence on the lower flanks of the salt body, as well as an improvement on the tomography above the TOS. Figure 11 shows that the gradient partitioning algorithm performs better in the underside areas of the left hand salt body, as well as on the flanks and underside of the right hand body. While the very bottom of the right hand side body has not converged as well as the other regions, further partitioning of the model (into top, sides, bottom) would allow for better updating on this region. A simple two part split may be insufficient since there are multiple entry/exit points on the ray paths that illuminate this lowest region of the body, which exasperates the problem of coupling this reflector with other reflector updates. This example demonstrates the potential of the domain decomposition algorithm to determine salt boundaries in more realistic models.

CONCLUSIONS

RTM imaging is fundamentally unable to differentiate between reflector position errors and tomography errors. By splitting the spatial domain of the implicit surface update, and performing separate non-linear parameter searches for each, we are able to mitigate occurence of local minima due to stronger TOS updates overwhelming the line search. The results from using this approach give us much better convergence

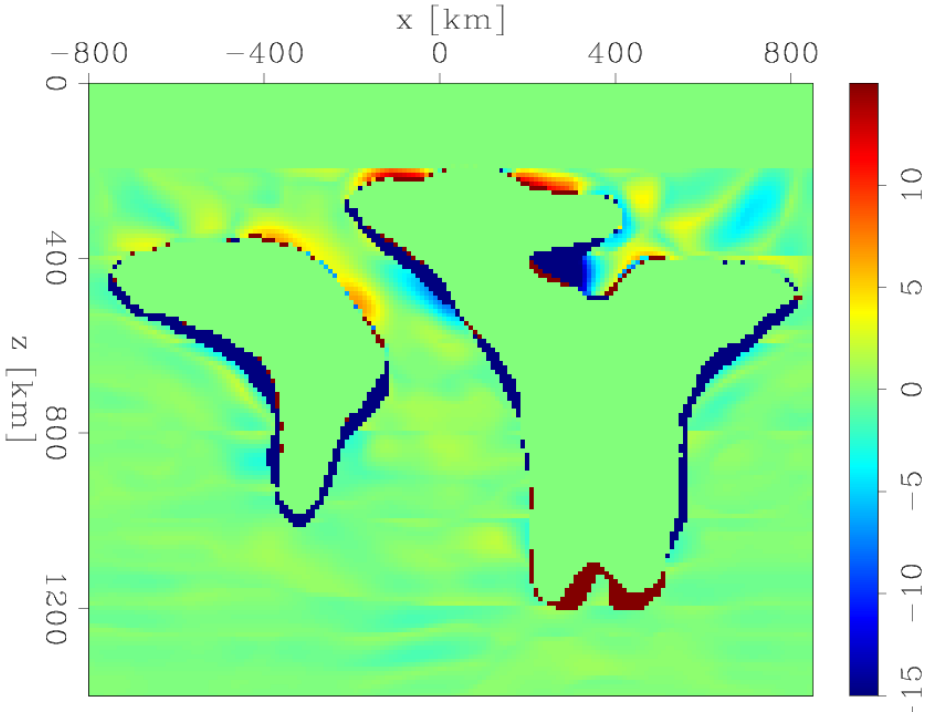

Figure 11: Velocity model difference between general and split ϕ methods, shown as a percentage error from the true model. Blue regions are where the domain partitioning approach performs better; red, more poorly. [**CR**]

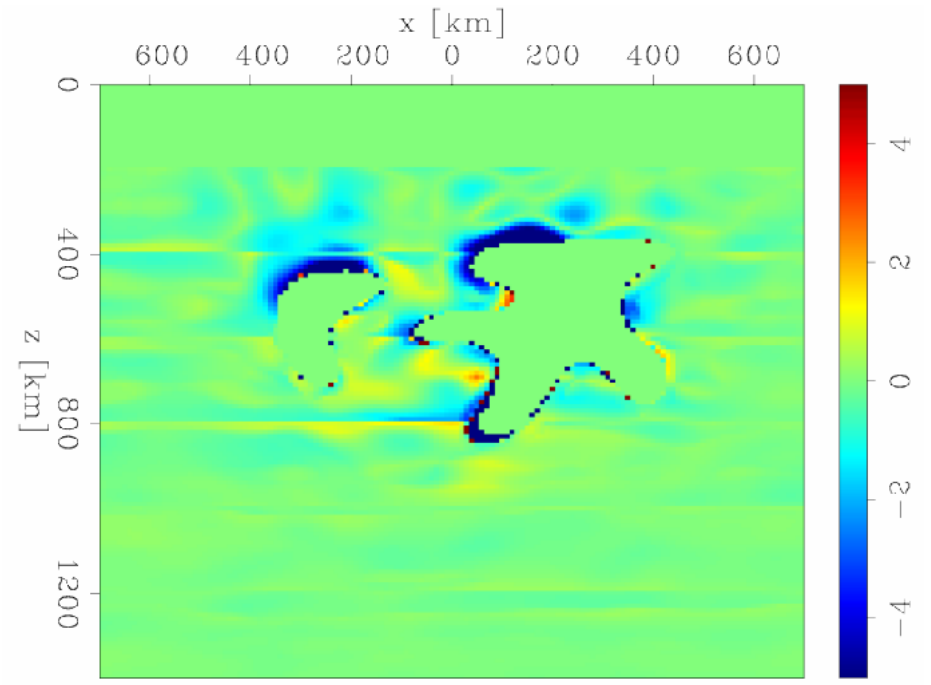

Figure 12: Velocity model difference between general and split ϕ methods, shown as a percentage error from the true model. Blue regions are where the domain partitioning approach performs better; red, more poorly. [**CR**]

on the base and flanks of the salt body. Further partitioning of the domain could improve convergence even more for complex salt models.

REFERENCES

Burger, M., 2003, A framework for the construction of level set methods for shape optimization and reconstruction: Interfaces and Free boundaries, **5**, 301–330.

Chaudhury, K. N. and K. R. Ramakrishnan, 2007, Stability and convergence of the level set method in computer vision: Pattern Recognition Letters, **28**, 884–893.

Guo, Z. and M. de Hoop, 2013, Shape optimization and level set method in full waveform inversion with 3d body reconstruction: SEG Technical Program Expanded Abstracts 2013, 1079–1083.

Lewis, W., B. Starr, and D. Vigh, 2012, A level set approach to salt geometry inversion in full-waveform inversion: SEG Technical Program Expanded Abstracts 2012, 1–5.

Li, C., C. Xu, C. Gui, and M. Fox, 2010, Distance regularized level set evolution and its application to image segmentation: Image Processing, IEEE Transactions on Image Processing, **19**, 3243–3254.

Osher, S. and J. A. Sethian, 1988, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations: Journal of computational physics, **79**, 12–49.

Plessix, R.-E., 2006, A review of the adjoint-state method for computing the gradient of a functional with geophysical applications: Geophysical Journal International, **167**, 495–503.

Santosa, F., 1996, A level-set approach for inverse problems involving obstacles: ESAIM Control, Optimization, and Calculus of Variations, **1**, 17–33.