Introduction

The process of building velocity models for salt bodies is one that traditionally requires significant manual intervention. While approaches to completely automate the work flow of salt model building are ambitious, what is certainly more practical are methods that minimize the need for human input except at the most important junctures of the work flow. Previous work by Lewis et al. (2014) propose methods for utilizing interpreter guidance for directing FWI inversion. However, with salt bodies, FWI can have trouble creating the sharp boundaries that salt often has. Work by Santosa (1996), Burger (2003), Lewis et al. (2012), Guo and de Hoop (2013), and Dahlke et al. (2015) demonstrate how level sets can build sharp boundary salt models even without high frequencies in the data by using an implicit surface to track the boundary. These steepest descent approaches were improved upon by using the Hessian of a new objective function Dahlke et al. (2016). More recently, the work in Kadu et al. (2017) demonstrated the practical improvement that radial basis functions (RBF's) provide for defining the implicit surface as a sparse and more computationally feasible model space. However, the update gradient for these examples has support only along the salt boundaries. By expanding the support of the gradient, we can make updates in areas inside the salt body and thus allow for inclusion discovery. We can choose where to allow these updates using interpreter input as guidance. This guidance can further be used to modify the implicit surface to direct a level-set based salt body inversion while still allowing the final model to ultimately be determined by the data itself. In this work, we begin by explaining how level sets are combined with the FWI objective function for optimizing the shape of the salt. Next, we explain how to include the interpreter guidance in this inversion, and how it relates to an improved method for mapping the radial basis function centers. Last, we demonstrate and compare this work flow on the Sigsbee synthetic model.

Derivation

Shape optimization

The first step of this derivation is to describe the model space that we are working with. We will call our velocity model m, which we define at every 2D spatial position (i, j) as:

$$m(\phi_{i,j}, b_{i,j}) = \hat{H}(\phi_{i,j})(c_{\text{salt}} - b_{i,j}) + b_{i,j}, \tag{1}$$

where $\hat{H}(\circ)$ is a smooth approximation to the Heaviside function, $m(\phi_{i,j}, b_{i,j})$ is the velocity value, $\phi_{i,j}$ is the implicit surface value, and $b_{i,j}$ is the background velocity value. We first want to define a perturbation of m in terms of $\triangle b$ and $\triangle \phi$. To do this, we generalize these parameters for the entire spatial domain (ignoring i, j), and expand equation (1) with a Taylor series as:

$$m_1 = m_0 + \frac{\partial m}{\partial \phi}\bigg|_{m_0} \triangle \phi + \frac{\partial m}{\partial b}\bigg|_{m_0} \triangle b + ...$$

By truncating this series and ignoring higher order terms, we can create a linear approximation of $\triangle m$:

$$\triangle m \approx \begin{bmatrix} \frac{\partial m(\phi_0, b_o)}{\partial \phi} & \frac{\partial m(\phi_0, b_o)}{\partial b} \end{bmatrix} \begin{bmatrix} \triangle \phi \\ \triangle b \end{bmatrix}$$

$$\approx D \triangle p.$$

where we define $\triangle p = [\triangle \phi \quad \triangle b]^T$, and D as the derivative taken at $m(\phi_0, b_0)$:

$$D = \begin{bmatrix} \delta(\phi_0)(c_{\text{salt}} - b_0) & 1 - \hat{H}(\phi_0) \end{bmatrix}.$$

This operator D scales and masks the parameter fields $\triangle \phi$ and $\triangle b$. From the FWI objective function:

$$\psi = ||F(m(\phi, b)) - d_{\text{obs}}||$$

we can take the derivative with respect to ϕ and b to find the gradient:

$$\triangle p \approx D^T B^T \triangle d.$$

where $\triangle d$ is the data space residual, and B is the classic Born operator. Similarly, we can find the application of the Hessian to the search direction as:

$$D^T H D \triangle p \approx -D^T B^T \triangle d. \tag{2}$$

In equation 2, we can substitute H with the Gauss-Newton Hessian of the FWI objective function. The method we propose solves equation 2 for $\triangle p$ using a conjugate gradient algorithm.

Modifications for inclusion discovery

An underlying axiom of the previous derivation is that the salt model only changes at the boundary of the original shape, due to the $\delta(\phi_0)$ term. However, an advantage of level sets is that the implicit surface can be deformed at positions that are not on the boundary, even to the extent that it "punctures" the zero level set and creates a donut hole or other topology type. If we only update along the current boundary, we eliminate the possibility of topology changes, such as the discovery of inclusions.

For this reason, we change the masking term in our D operator to allow for updating outside of the boundary regions. Because the support of the gradient we previously derived is a subset of the actual objective function support (which is the entire model domain), we can expand the support of the masking term to include regions inside the current salt boundaries without affecting our ability to minimize our objective function. We represent this modification with a new term $\hat{\delta}(\phi_0, G)$, which takes into account the interpreter guidance G.

$$D = \begin{bmatrix} \hat{\delta}(\phi_0, G)(c_{\text{salt}} - b_0) & 1 - \hat{H}(\phi_0) \end{bmatrix}.$$

We can only discover an inclusion if the update perturbs the implicit surface below the zero-level set. This means that the way we initialize the height of the implicit surface matters. The opportunity here is to set the height of the implicit surface according to how likely we believe that an inclusion is present at that position. This allows us to input a probabilistic mapping of inclusion likelihood into our initialization of ϕ. For the Sigsbee example, Figure 1a shows the implicit surfaces with interpreter guidance applied, and Figure 1b shows it without.

Sparsifying with Radial Basis Functions

Kadu et al. (2017) replaces a regular grid parameterization of ϕ with an aggregate surface composed of many RBFs, resulting in a much sparser model:

$$\phi(\lambda; \varepsilon, r) = \sum_i^{N\lambda} \lambda_i \exp^{-(\varepsilon r)^2}$$

where λ is the new model parameter, r is the radial distance from the RBF center i, and ε controls the sharpness of the RBF taper (constant). Rather than use regular spacing of RBF centers as in Kadu et al. (2017), we cluster the RBF centers where we expect to see updating occur (Figure 1c). The RBF centers are randomly chosen from a spatial probability distribution based on $\hat{\delta}(\phi_0, G)$. Clustering allows us to use fewer RBF parameters to attain a higher resolution around the salt boundary than we would get using regular spacing of the RBF centers. Our operator D must be modified to account for the additional linear transformation inherent in the RBF formulation:

$$D = \begin{bmatrix} \frac{\partial m(\phi_0, b_o)}{\partial \phi} \frac{\partial \phi}{\partial \lambda} & \frac{\partial m(\phi_0, b_o)}{\partial b} \end{bmatrix}$$

$$= \begin{bmatrix} \hat{\delta}(\phi_0, G)(c_{\text{salt}} - b_0) \exp^{-(\varepsilon r)^2} & 1 - \hat{H}(\phi_0). \end{bmatrix} \tag{3}$$

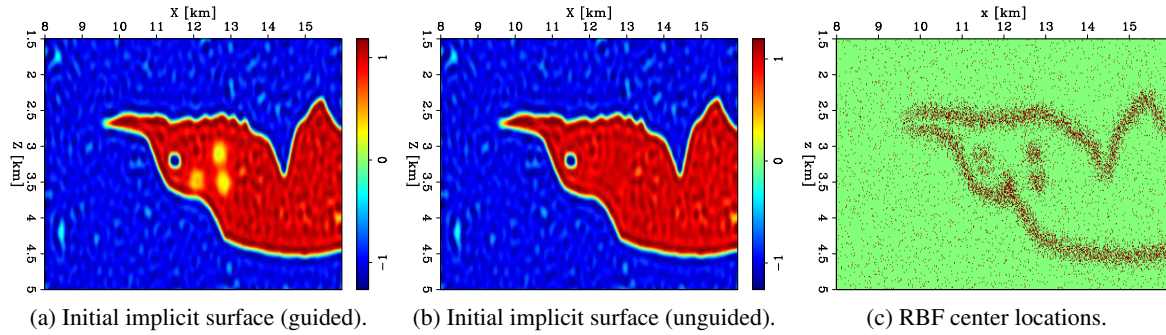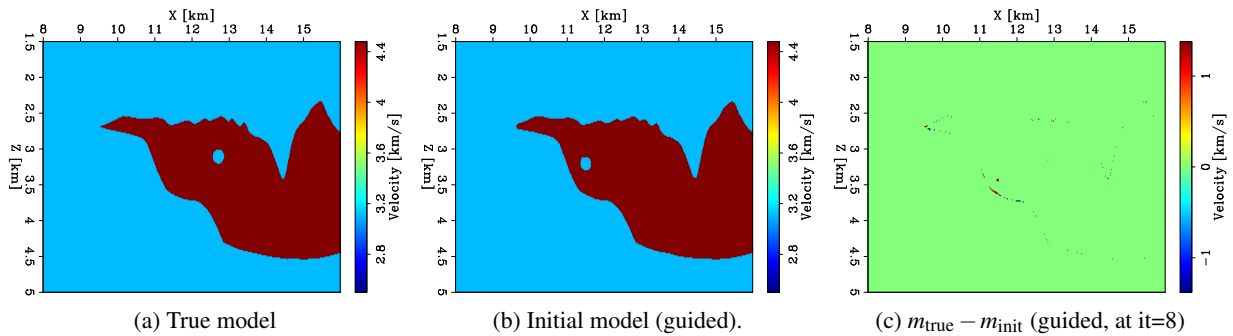| (a) Initial implicit surface (guided). | (b) Initial implicit surface (unguided). | (c) RBF center locations. |

Figure 1

| (a) True model | (b) Initial model (guided). | (c) $m_{\text{true}} - m_{\text{init}}$ (guided, at it=8) |

Figure 2

This also changes our model space to $\triangle p = [\triangle \lambda \quad \triangle b]^T$.

Once we have chosen the RBF function centers, we then select the variance (ε) to describe the shape of the Gaussian that each RBF creates. In our example these parameters were chosen by trial and error such that the aggregated surface covers the full model domain. This could certainly be performed automatically in a more advanced implementation. After, we can invert for the amplitudes of the RBF's (λ) with a linear conjugate gradient inversion that uses the D operator defined earlier. This gives us a set of RBF parameters that create an implicit surface that matches our initial implicit surface (and thus our initial velocity model). We can then begin the actual non-linear inversion from this starting model.

Demonstration on 2D Sigsbee model

We used a portion of the 2D Sigsbee velocity model and modified it with a low-velocity inclusion (Figure 2a). Our initial model was the same expect with the inclusion shifted left about 1.5km (Figure 2b). For each inversion the acquisition was 24 shots and 380 receivers evenly spaced, with a 8Hz source wavelet. The first inversion had an implicit surface initialized using interpreter guidance, while the second did not. Both inversions used the modified masking function ($\hat{\tilde{\delta}}(\phi_0, G)$) to allow for updating inside the salt body, and used the same RBF centers (Figure 1c). However, the "unguided" inversion did not have a modified implicit surface. In both cases, updating was done only on the λ parameter; the b parameter was fixed over all iterations.

As we expected, by expanding the support of the gradient to include the interpreter guidance areas, we are able to recover the correct inclusion and close the false one (Figure 2c) for both cases. Even with the some of the interpreter guidance being incorrect (compare Figure 1a with 2a), the inversion was not misguided. Further, we found that the additional step of initializing ϕ as per Figure 1a gave us better convergence in terms of both the model and data residual norms (Figure 3a and 3b).

(a) Objective function value (data residual norm)
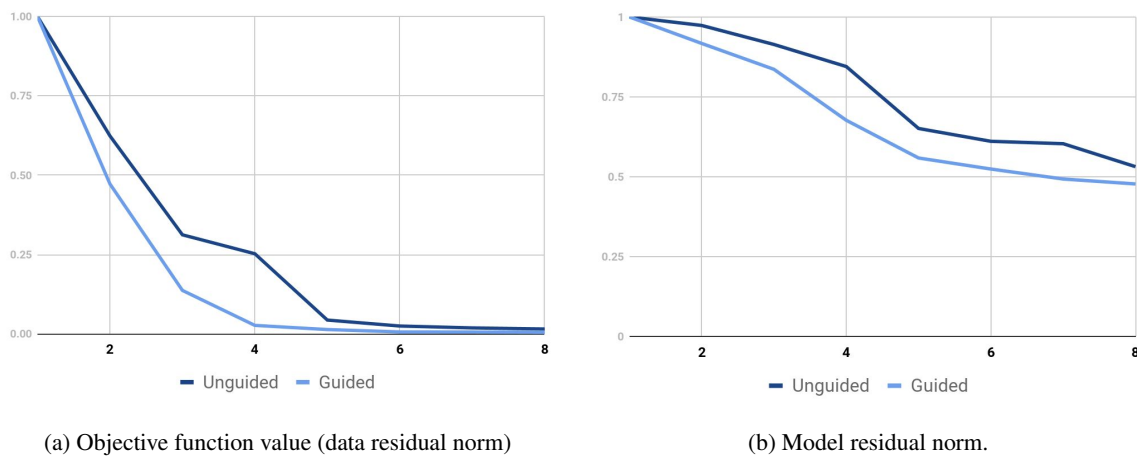
(b) Model residual norm.

Figure 3

Conclusions

We introduce the derivation of a shape optimization algorithm that minimizes the FWI objective function using a level set based on a radial basis parameterization. From this, we introduce a way to integrate interpreter guidance by using a masking term in our operator with expanded spatial support. We further use this guidance to augment the shape of the initial implicit surface. With the Sigsbee synthetic model, we demonstrate how this improves convergence of our non-linear inversion, even with poorly placed interpreter guidance present. We find that this approach elegantly allows for interpreter guidance with the level set formulation of salt inversion, and improves the ability to discover inclusions.

Acknowledgements

References

Burger, M. [2003] A framework for the construction of level set methods for shape optimization and reconstruction. *Interfaces and Free boundaries*, **5**(3), 301–330.

Dahlke, T., Biondi, B. and Clapp, R. [2015] *Domain decomposition of level set updates for salt segmentation*. 1366–1371.

Dahlke, T., Clapp, R. and Biondi, B. [2016] *Second-order updating in shape optimization for salt segmentation*. 5369–5373.

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

Kadu, A., Leeuwen, T.V. and Mulder, W. [2017] *Parametric level-set full-waveform inversion in the presence of salt bodies*. 1518–1522.

Lewis, W., Amazonas, D., Vigh, D. and Coates, R. [2014] *Geologically constrained full-waveform inversion using an anisotropic diffusion based regularization scheme: Application to a 3D offshore Brazil dataset*. 1083–1088.

Lewis, W., Starr, B., Vigh, D. et al. [2012] A Level Set Approach to Salt Geometry Inversion in Full-Waveform Inversion. In: *2012 SEG Annual Meeting*. Society of Exploration Geophysicists.

Santosa, F. [1996] A level-set approach for inverse problems involving obstacles. *ESAIM Controle Optim. Calc. Var*, **1**, 17–33.