

# Chapter 3

## Sensitivity and Parameterization

In this chapter I study the sensitivity of different parameter combinations to seismic data in acoustic anisotropic FWI. Choosing a suitable set of parameters is important because it dictates the topology of the objective function, the invertibility of subsurface models, crosstalk patterns, and wavenumber content of the final estimations. The choice of parameters depend on types of data, acquisition, and the availability and accuracy of the initial models. Through a synthetic experiment, I examine eight parameterizations and observe that those combining one velocity (vertical,  $v_z$ , horizontal,  $v_h$ , or NMO,  $v_n$ ) and two anisotropic parameters (among  $\epsilon$ ,  $\delta$ , and  $\eta$ ) produce the best estimated models with lower cost functions and model residuals and better focused final RTM images. When using such combinations, one needs to reformulate the problem in terms of rescaled or normalized parameters to account for differences in sensitivity and dimensions. Having one parameter that is most sensitive to seismic data, velocity, helps the inversion to converge better, but the final anisotropic models suffer heavily from crosstalk artifacts.

I also observe that the choice of parameters determines correlation between model updates. For example, if the inversion is set up in terms of  $(v_z, \epsilon, \delta)$ , the final  $\epsilon$  update has a positive correlation with those of the other two parameters. However, when used in  $(v_h, \epsilon, \delta)$ , the  $\epsilon$  update is in the opposite direction from the other two. This

correlation depends on the starting models and on the topology of the objective function, which is a consequence of parameterization and acquisition. Because parameter tradeoff is controlled by the Hessian of the objective function, one can explain the model update correlation by studying the multi-parameter Point Spread Functions.

When the problem is formulated in terms of one velocity and two anisotropic parameters, it might be tempting to perform a mono-parameter inversion by updating only the most influential parameter, velocity, if the smooth background models for Thomsen parameters are deemed good enough. I find that a simultaneous inversion of all three parameters results in a better velocity estimation than that from a one-parameter inversion, even though inverted Thomsen parameters might be erroneous due to crosstalk from velocity.

## PARAMETER SCALING

The performance of a multi-parameter FWI depends heavily on how the problem is formulated. One crucial aspect of formulation is scaling. A poorly scaled problem is one in which variations in some parameters produce a much greater change in the objective function than variations in other parameters. In that case, the problem's condition number is high. What happens is that some parameters will have much larger gradients than others, and therefore get updated, while other parameters remain unchanged. The inversion eventually gets stuck at a local minimum or even diverges. One could potentially update each parameter at alternate iterations but that would require more iterations and increase the computational cost.

If one chooses a parameterization with different parameter types, which might have very different dimensions, ranges, and physical significance, proper scaling is necessary. In the context of anisotropic FWI, velocity usually has dimension of meter per second and typically ranges from 1500 m/s to 5000 m/s, while Thomsen parameters,  $\epsilon$  and  $\delta$  for example, are dimensionless and range between zero and one. Through a synthetic example, I show how scaling can affect the performance of anisotropic FWI.

In this example I choose to invert for vertical velocity,  $v_z$ , and two Thomsen

parameters,  $\epsilon$  and  $\delta$ . The true and starting models are respectively shown in Figures 3.1 and 3.2. The parameters are normalized by

$$\bar{v}_z = \frac{v_z}{v_0}, \bar{\epsilon} = \frac{\epsilon}{\epsilon_0}, \bar{\delta} = \frac{\delta}{\delta_0}, \quad (3.1)$$

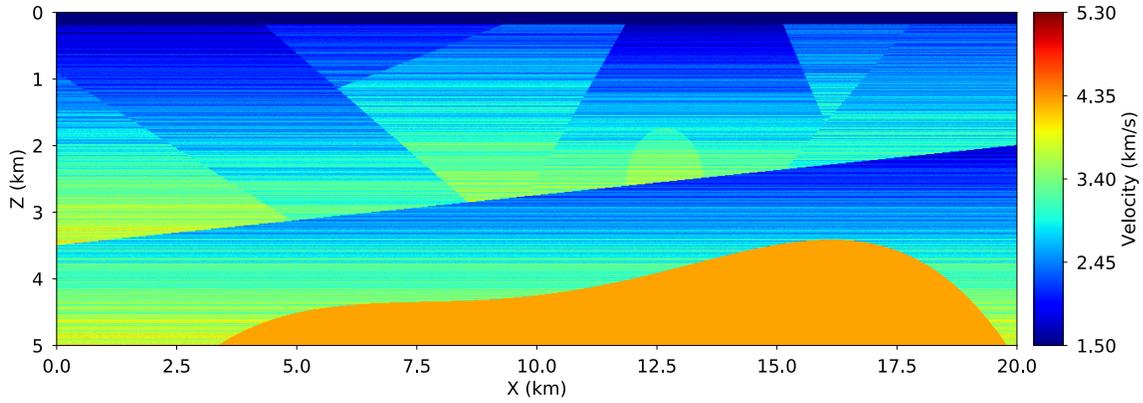
where the constant scaling factors  $v_0$ ,  $\epsilon_0$ , and  $\delta_0$  are chosen such that the ratios between the initial gradients and the initial models, measured in  $l_\infty$  norm, are the same for all three parameter types

$$\frac{\|g_{\bar{v}_z}^0\|_\infty}{\|\bar{v}_z^0\|_\infty} = \frac{\|g_{\bar{\epsilon}}^0\|_\infty}{\|\bar{\epsilon}^0\|_\infty} = \frac{\|g_{\bar{\delta}}^0\|_\infty}{\|\bar{\delta}^0\|_\infty}. \quad (3.2)$$

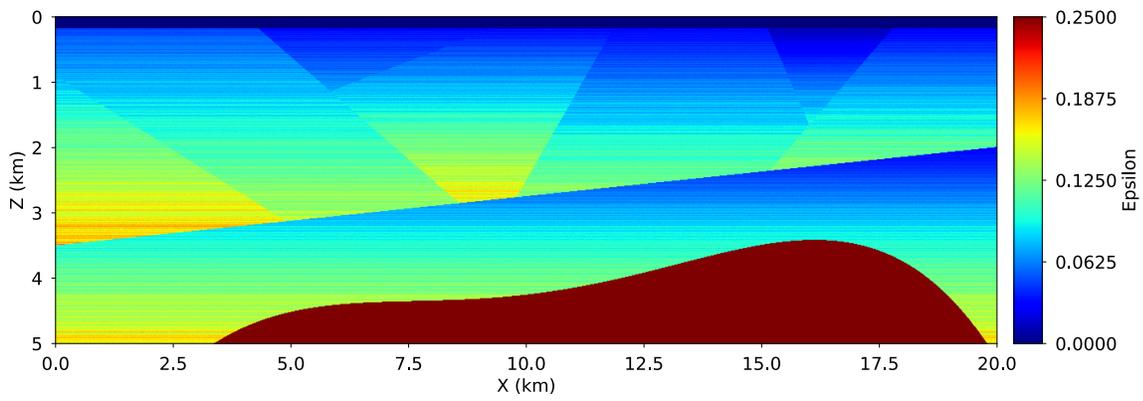
My intuition for choosing the scaling factors this way is that we would like to have the same amount of updates across parameters. An alternative choice can be the mean values of the initial models. Note that there are three scaling factors,  $v_0$ ,  $\epsilon_0$ , and  $\delta_0$ , but there are only two equations in 3.2, so one of the factors is freely chosen. I usually choose  $\delta_0 = 1$ .

I use a uniform acquisition with 100 shots at 200 meters spacing and 800 receivers every 25 meters along the surface. The source wavelet is a Ricker at 5 Hz center frequency. The inversion is performed using the LBFGS solver from Nocedal (1980) with 100 function evaluations. Figure 3.3 shows the inverted velocity and Thomsen parameters when no scaling is applied. Compared with the initial models, velocity is nearly unchanged while the other two parameters are erroneously updated. The "sprinkles" in the shallow part of the inverted  $\epsilon$  and  $\delta$  models are where the inversion attempts to step into infeasible region,  $\epsilon < \delta$ , because the gradients are poorly scaled.

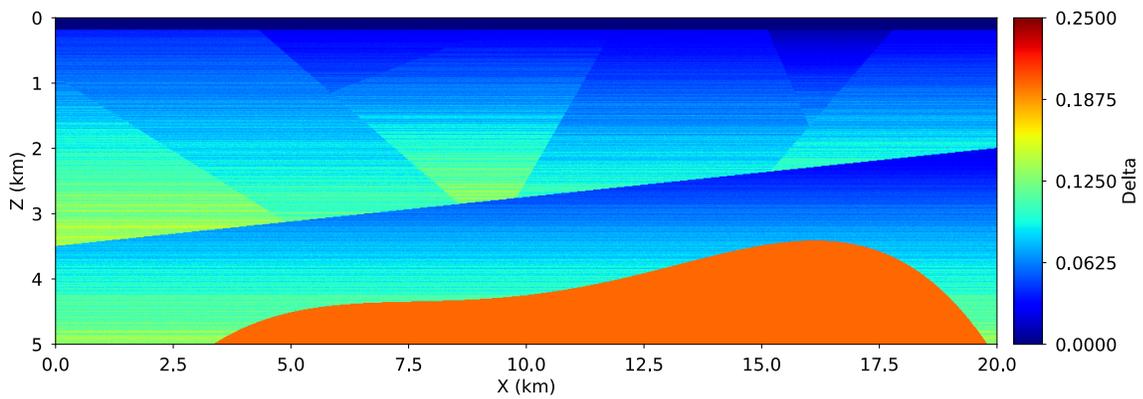
When properly scaled, the inversion converges to a much more meaningful model, shown in Figure 3.4. Now the final velocity model recovers a lot of features in the true model that are not present in the initial model. The final  $\epsilon$  and  $\delta$  models are, however, heavily affected by velocity update. This is a problem of crosstalk, which I will discuss in the next section. Figure 3.5 shows the two objective functions of the unnormalized and normalized inversions. Without normalization, the inversion stops after only 18 iterations and could not proceed because the line search routine cannot



(a)



(b)



(c)

Figure 3.1: A synthetic model: (a)  $v_z$ , (b)  $\epsilon$ , and (c)  $\delta$ .

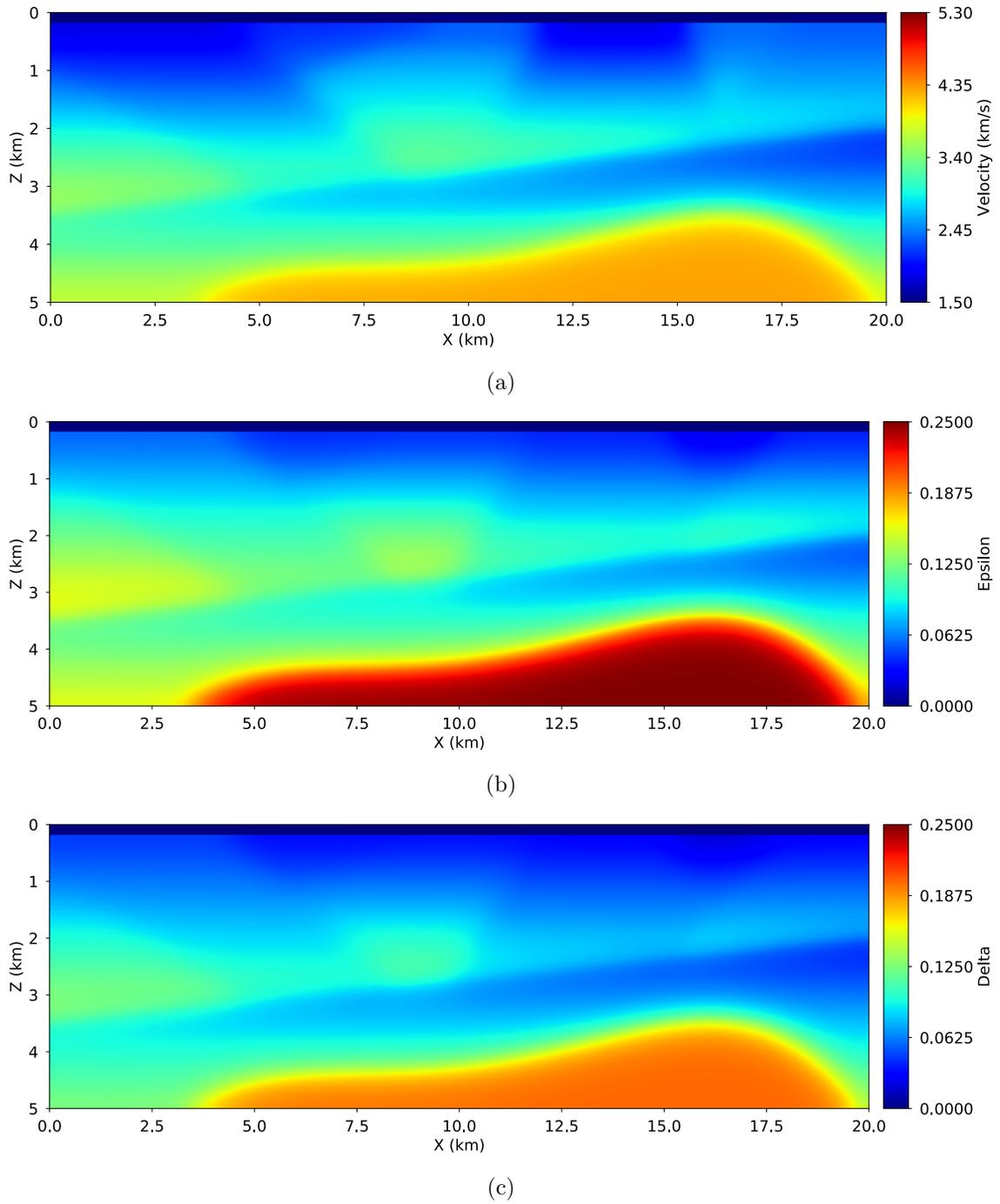


Figure 3.2: Starting models: (a)  $v_z$ , (b)  $\epsilon$ , and (c)  $\delta$ .

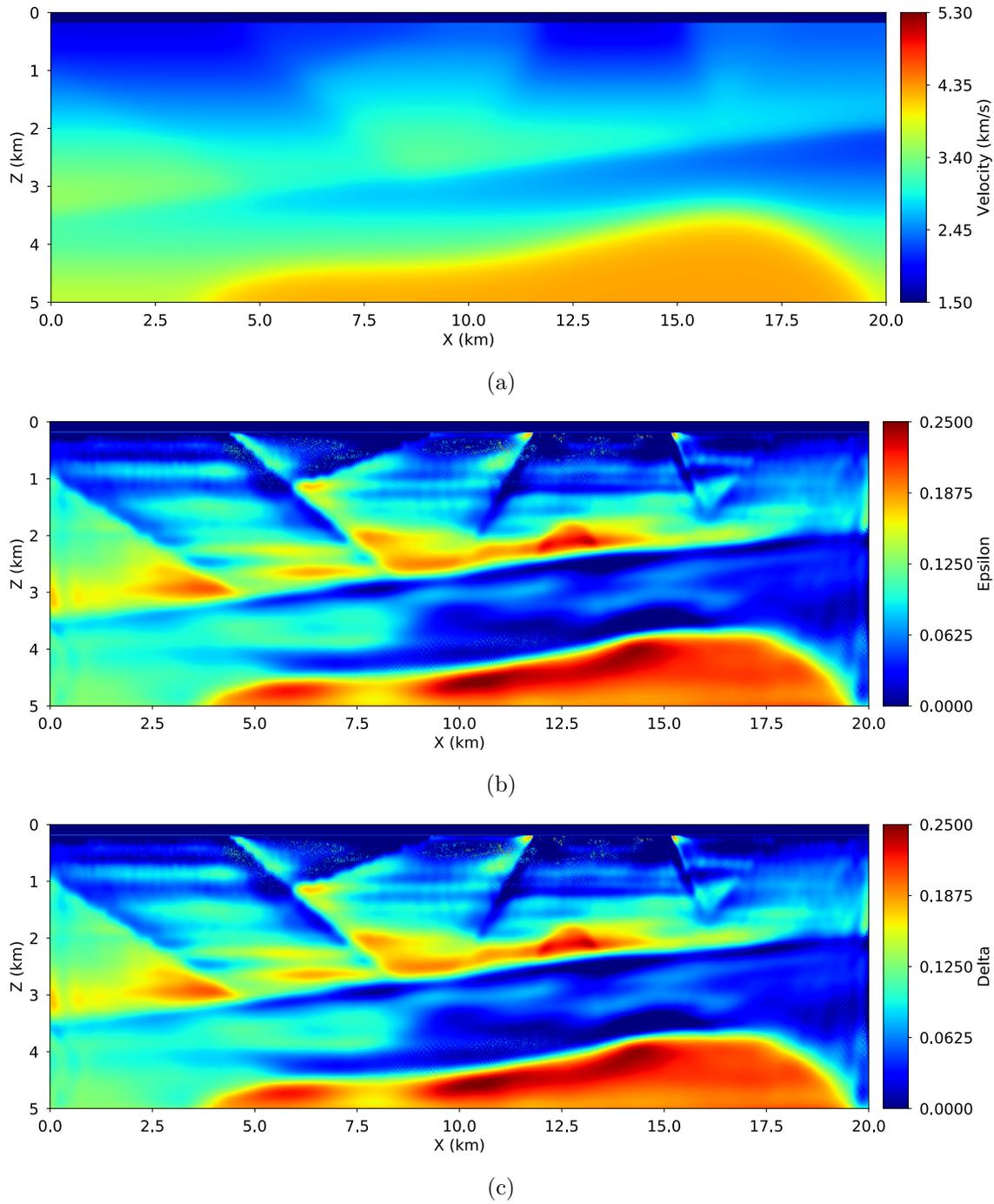


Figure 3.3: Final models when no scaling is applied. (a)  $v_z$ , (b)  $\epsilon$ , and (c)  $\delta$ .

find a satisfactory step length.

## PARAMETER CORRELATION

Besides scaling, another important aspect in formulating a multi-parameter FWI problem is parameterization, i.e. how to choose the set of parameters to invert for. For an acoustic VTI medium, there are three common parameter types: velocity (vertical  $v_z$ , horizontal  $v_h$ , and NMO  $v_n$ ), anisotropic parameters ( $\epsilon$ ,  $\delta$ , and  $\eta$ ), and stiffnesses ( $c_{ij}$ ), from which a number of equivalent set of three parameters can be candidates for anisotropic FWI. The choice of parameterization dictates the topology of the objective function, tradeoff, the wavenumber content of the final models, and the correlations between model updates. This section reveals some of these characteristics.

Parameterization in acoustic anisotropic FWI has been studied by a number of researchers. Plessix and Cao (2011) perform eigenvalue decompositions of Hessian matrices to find the most sensitive parameters in different acquisition settings. They conclude that diving waves are most sensitive to horizontal velocity while NMO velocity is more relevant for reflections. In their work, they also prove that the Thomsen parameter  $\delta$  cannot be inferred from surface seismic data due to its tradeoff with depth. Gholami et al. (2013) analyze the radiation patterns of various parameters to understand their crosstalk, sensitivity, and how they depend on the scattering angle. The message from their work is the choice of parameters is influenced by acquisition, which determines the angle and wavenumber coverage, and the accuracy of the initial models. Alkhalifah and Édouard Plessix (2000) employ both radiation pattern and eigenvalue decomposition techniques to demonstrate that  $(v_n, \eta, \delta)$  parameterization properly captures sensitivity of diving waves and reflections while  $(v_h, \epsilon, \eta)$  is best for a hierarchical inversion approach, which first inverts for  $v_h$  with diving waves. The role of  $\delta$  and  $\epsilon$  in these two parameterizations is to fit the amplitude, accounting for limitations of the acoustic approximation. Alkhalifah (2016) once again emphasizes  $(v_h, \epsilon, \eta)$  and Guitton and Alkhalifah (2017) illustrate improvements in inversion results with this parameterization by synthetic and field data examples. Djebbi et al.

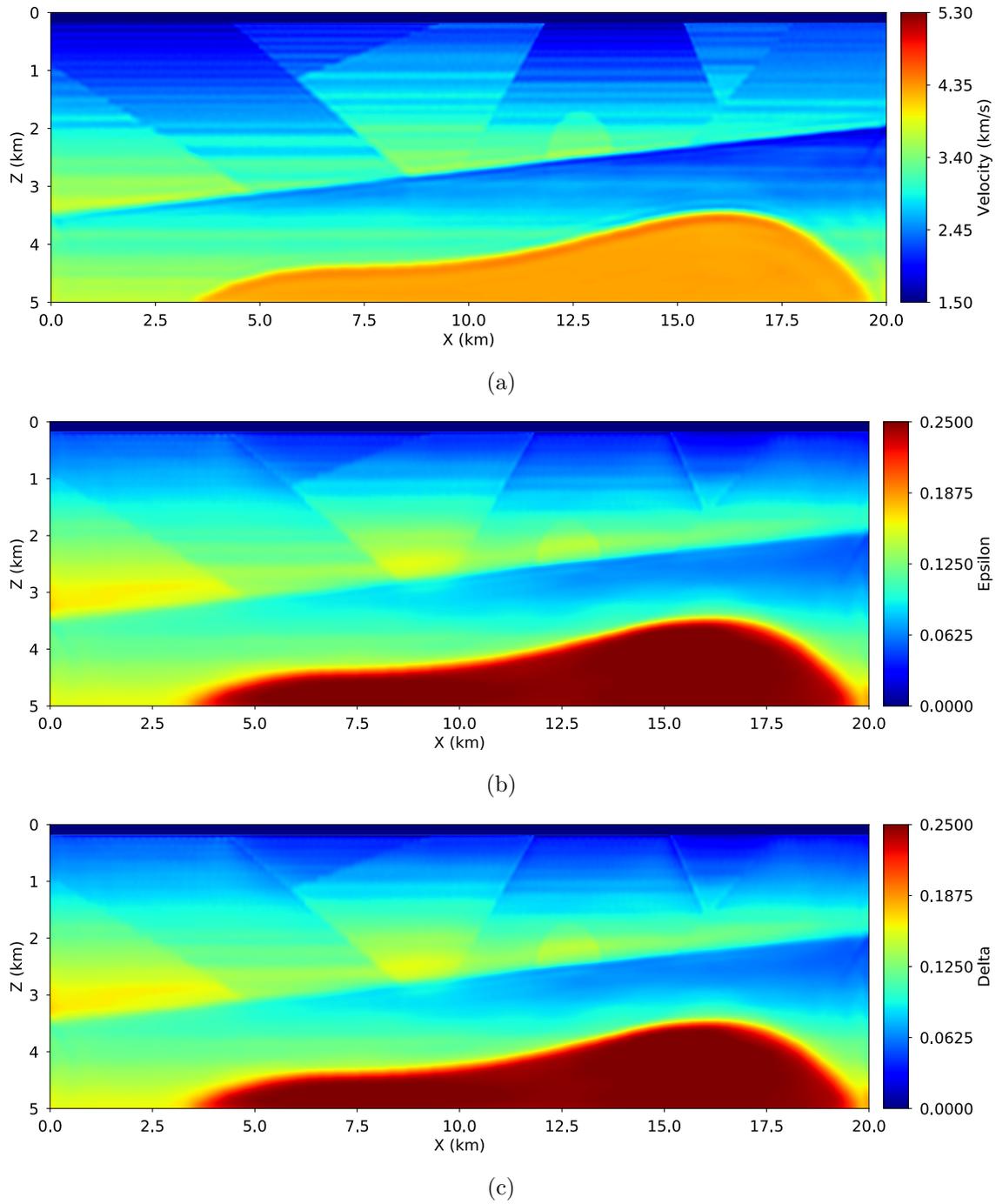
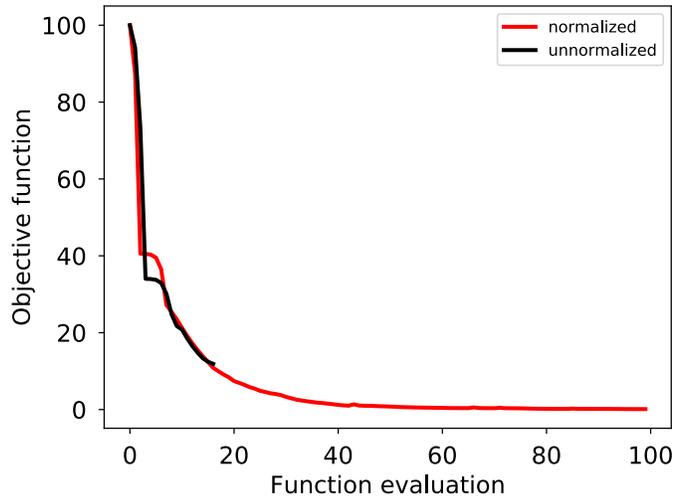


Figure 3.4: Final models when scaling is applied. (a)  $v_z$ , (b)  $\epsilon$ , and (c)  $\delta$ .

Figure 3.5: Compare objective functions of normalized and unnormalized inversions.



(2017) show that parameter sensitivity is also reflected in the traveltimes kernels computed with different source and receiver orientations.

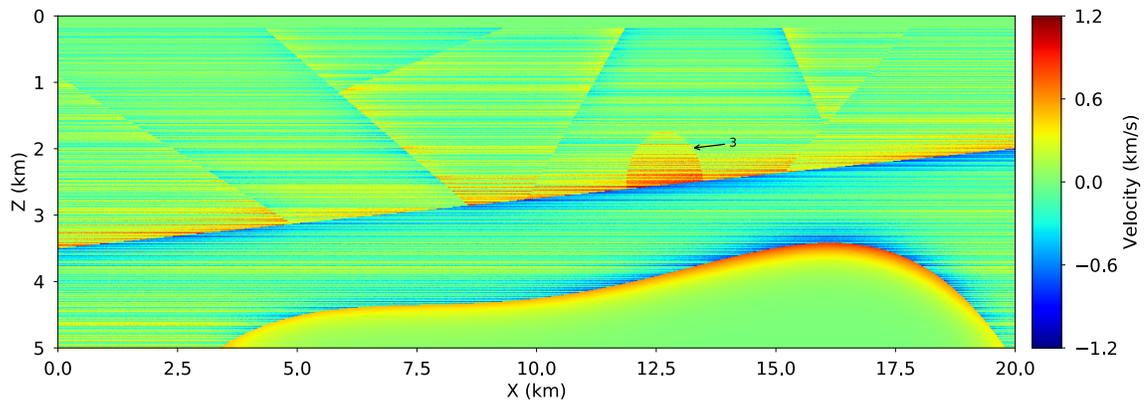
In this section I show the results of experiments with two different sets of parameters:  $(v_z, \epsilon, \delta)$  and  $(v_h, \epsilon, \delta)$ . The setup is similar to that of the previous example in terms of sources, receivers, wavelet, and solver. Figure 3.6(a) and 3.7(a) plot the differences between the true and initial models for vertical and horizontal velocities. Figure 3.6(b) plots the final vertical velocity update from an inversion using  $(v_z, \epsilon, \delta)$ , while Figure 3.7(b) plots the final model update in horizontal velocity after an inversion using  $(v_h, \epsilon, \delta)$ . The two velocity model updates are very similar to their corresponding ground truths. In contrast, the final model updates and their true-initial differences of the Thomsen parameters  $\epsilon$  and  $\delta$  are shown in Figures 3.8 and 3.9 respectively. Compared with the final velocity updates, the updates in Thomsen parameters are not as good, which is expected because they are less sensitive to seismic data than velocity.

In addition to being the most influential parameter, velocity is also sensitive at all scattering angles (having circular radiation patterns, Figure 3 in Gholami et al. (2013)). As a result, we can expect FWI to recover a wide range of wavenumber for velocity (Figures 3.6(b), 3.7(b), and their corresponding wavenumber domain

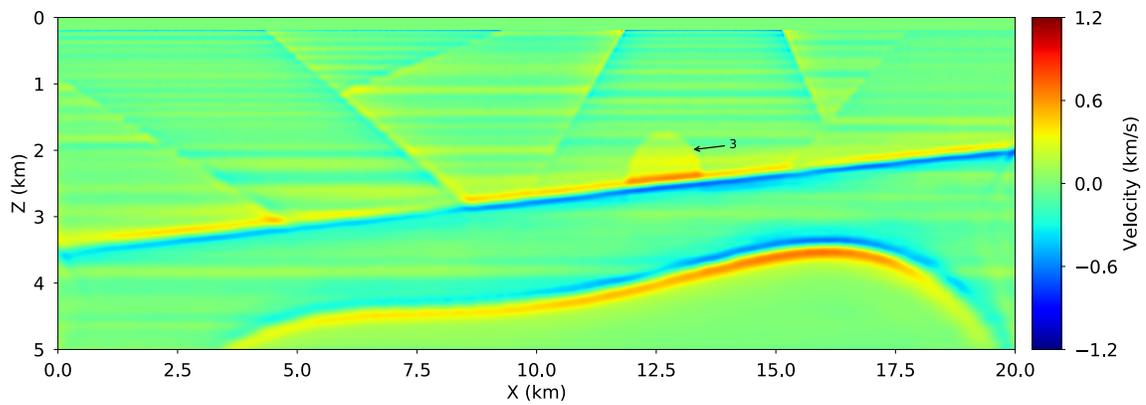
plots). The scattering angle coverages of  $\epsilon$  and  $\delta$  change with parameterization. In  $(v_z, \epsilon, \delta)$ ,  $\epsilon$  is sensitive at large angles while  $\delta$  is sensitive at intermediate angles. Consequently, an inversion of this set of parameters will recover a smooth low-wavenumber  $\epsilon$  and a intermediate-wavenumber  $\delta$  (Figures 3.8(b), 3.9(b), and their corresponding wavenumber domain plots, Figures 3.10(b) and 3.10(c)). On the other hand, in  $(v_h, \epsilon, \delta)$ , the influence of  $\epsilon$  reverts, being sensitive at small scattering angles and basically playing the role of reflectivity. An inversion with this parameterization will recover a high-wavenumber  $\epsilon$  and a smoother intermediate-wavenumber  $\delta$  (Figures 3.8(c), 3.9(c), and their corresponding wavenumber domain plots, Figures 3.11(b) and 3.11(c)).

Parameterization furthermore determines the correlation between model updates. Comparing the final updates in vertical velocity, Figure 3.6(b),  $\epsilon$ , Figure 3.8(b), and  $\delta$ , Figure 3.9(b), I observe that these model update have a positive correlation, i.e. a increase in velocity collocates with increases in  $\epsilon$  and  $\delta$ . When horizontal velocity is used instead of vertical velocity,  $\epsilon$  is updated in the opposite direction from the other two parameters (compare polarity of Figure 3.8(c) versus Figures 3.7(b) and 3.9(c)). Neither positive nor negative correlation indicates accurate updating. Notice the two features pointed out by the arrows in Figure 3.8: (1) is a fault and (2) is an anticline. Comparing with the ground truth, Figure 3.8(a), positive correlation between model parameters in  $(v_z, \epsilon, \delta)$  leads to wrong update at the fault but correct update at the anticline, Figure 3.8(b). On the other hand, negative correlation between  $(v_h, \epsilon, \delta)$  leads to correct update at the fault but wrong update at the anticline, Figure 3.8(c).

The correlation in parameter updates is also observed by Gholami et al. (2013) in acoustic anisotropic FWI and by Kamath and Tsvankin (2016) in elastic FWI. The correlation changes with acquisition, parameterization, and starting models. In order to understand this correlation between parameters, I look at the gradients at the initial models and observe that while the gradients in  $(v_z, \epsilon, \delta)$  have the same polarity (Figure 3.12), the gradient of  $\epsilon$  in  $(v_h, \epsilon, \delta)$  has an opposite polarity from the other two parameters (Figure 3.13). I also observe this correlation in the point spread functions (PSF). The PSFs in  $(v_z, \epsilon, \delta)$  share the same polarity (Figure 3.14) while

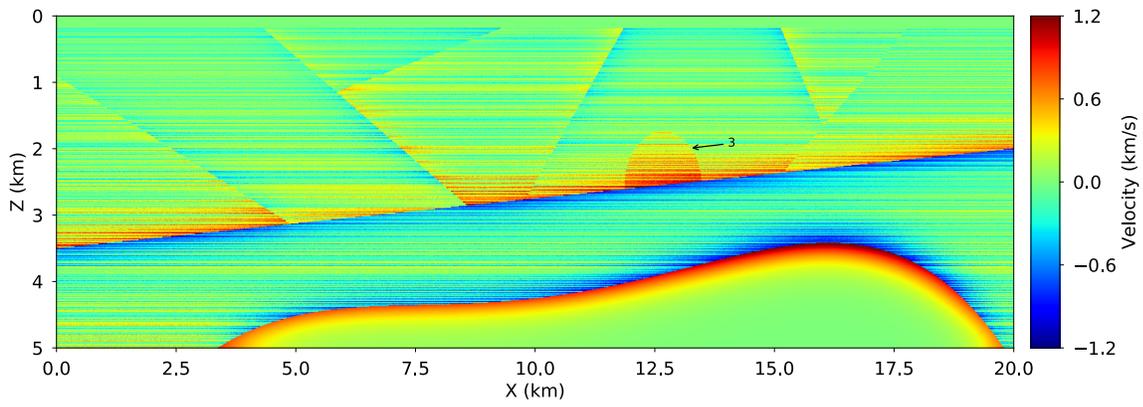


(a)

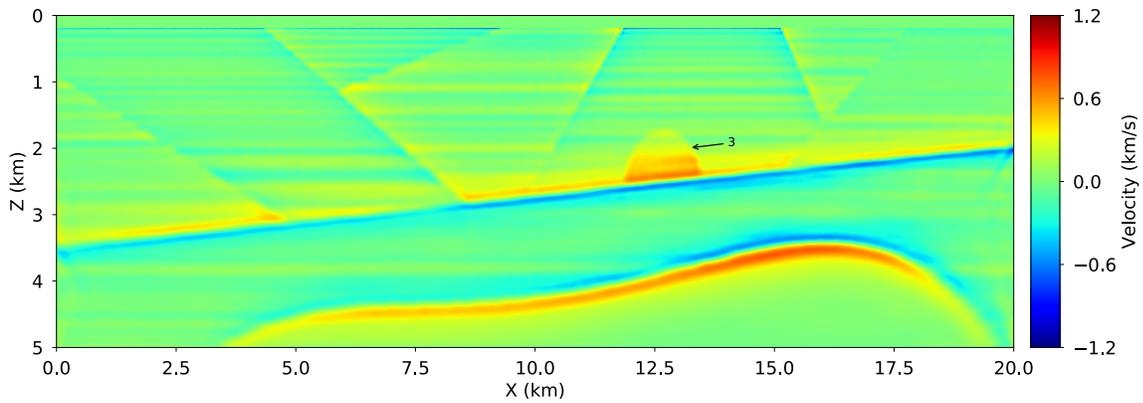


(b)

Figure 3.6: Inversion with  $(v_z, \epsilon, \delta)$  parameters. Panel (a) is the difference between the true and initial vertical velocity model while panel (b) is the model update.



(a)



(b)

Figure 3.7: Inversion with  $(v_h, \epsilon, \delta)$  parameters. Panel (a) is the difference between the true and initial horizontal velocity model while panel (b) is the model update.

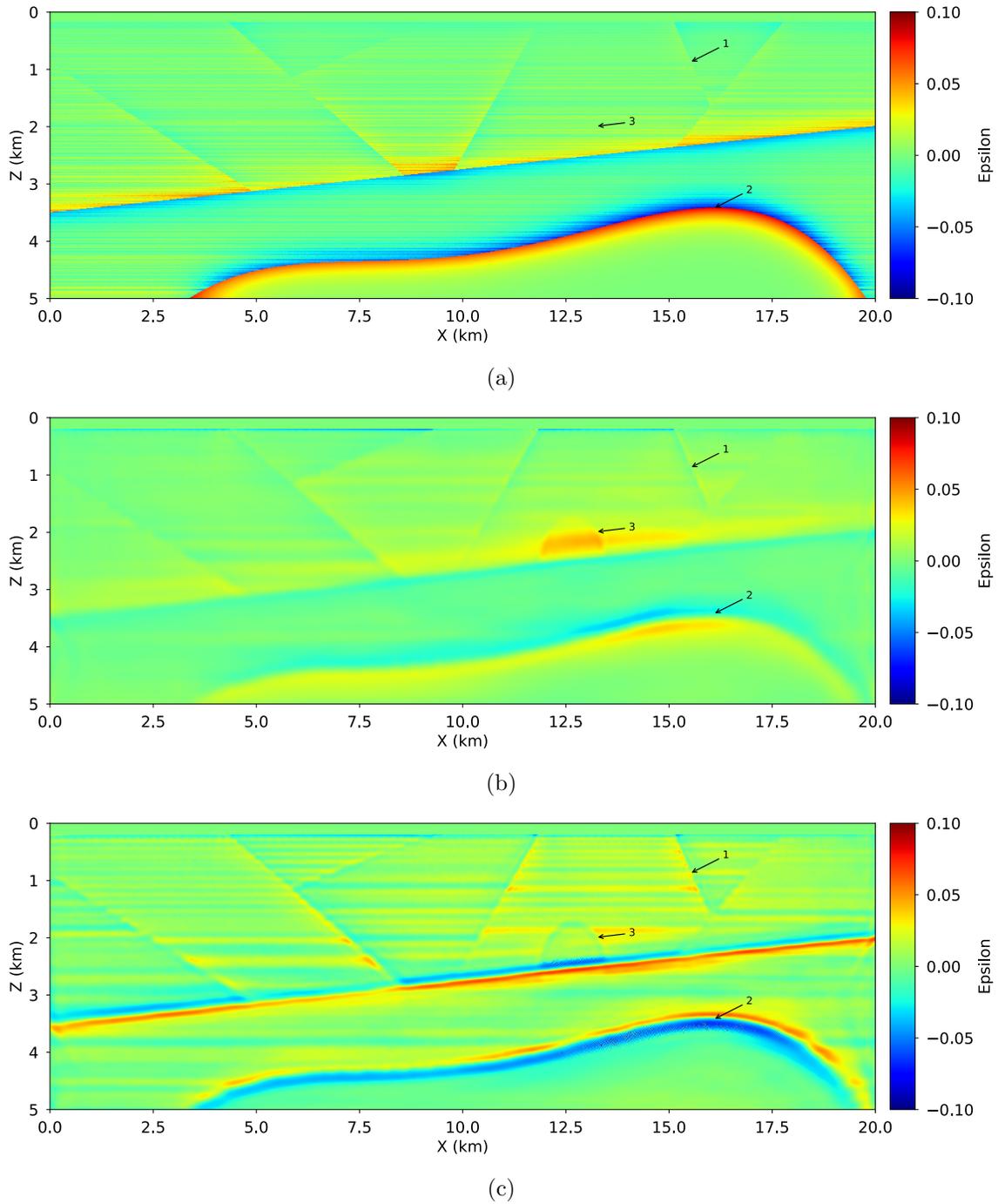
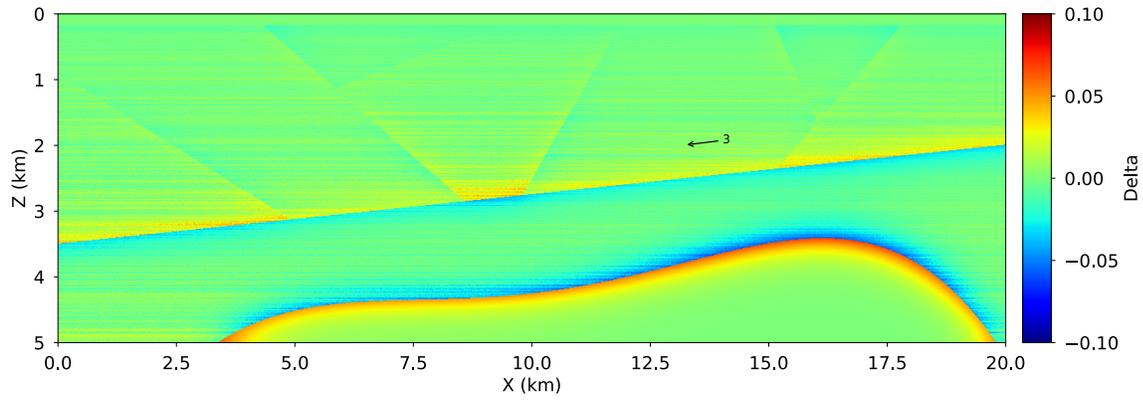
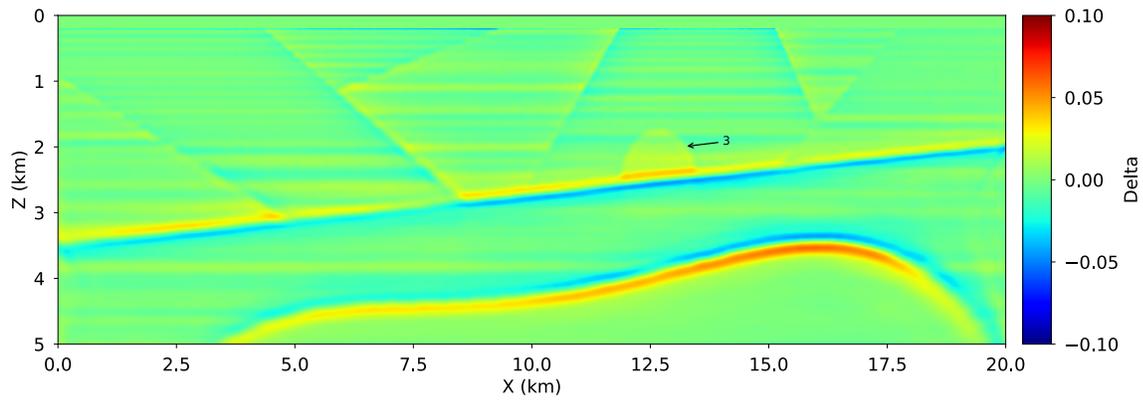


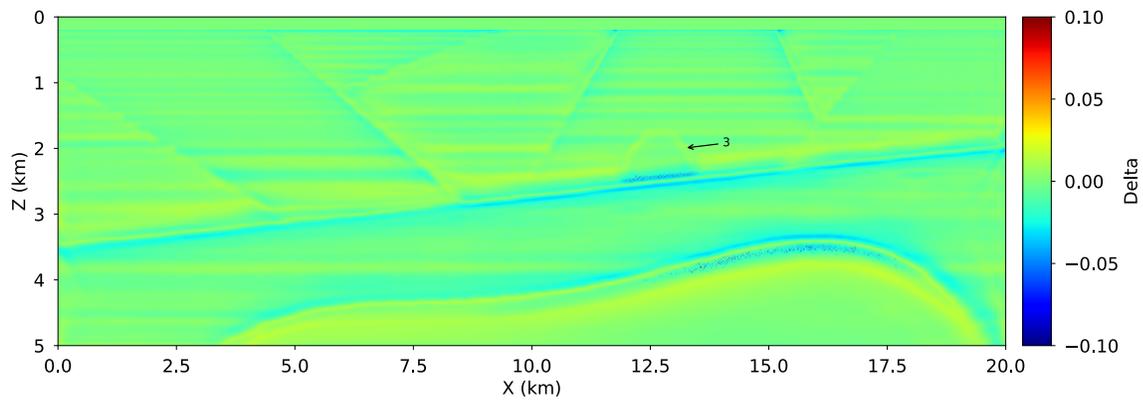
Figure 3.8: Panel (a) is the difference between the true and initial  $\epsilon$ , panel (b) is the model update using  $(v_z, \epsilon, \delta)$  parameterization, and panel (c) is model update using  $(v_h, \epsilon, \delta)$  parameterization.



(a)



(b)



(c)

Figure 3.9: Panel (a) is the difference between the true and initial  $\delta$ , panel (b) is the model update using  $(v_z, \epsilon, \delta)$  parameterization, and panel (c) is model update using  $(v_h, \epsilon, \delta)$  parameterization.

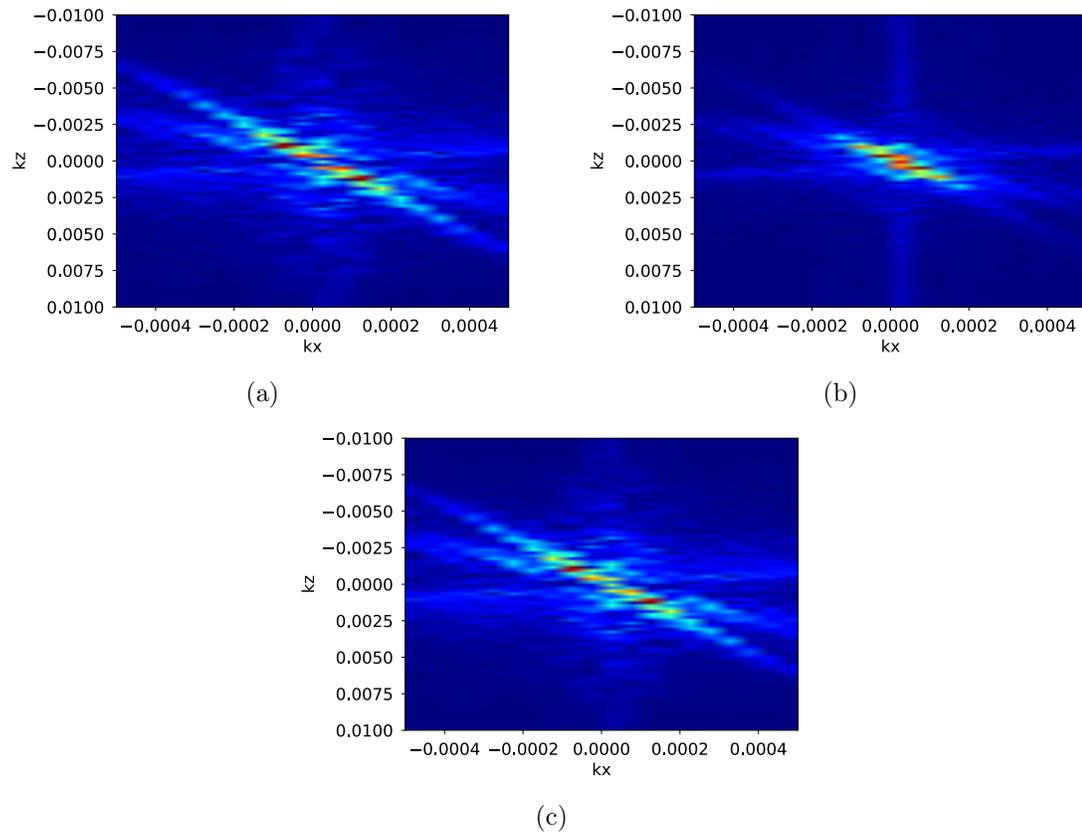


Figure 3.10: Fourier spectra of final model updates in (a)  $v_z$ , (b)  $\epsilon$ , and (c)  $\delta$  using  $(v_z, \epsilon, \delta)$ .

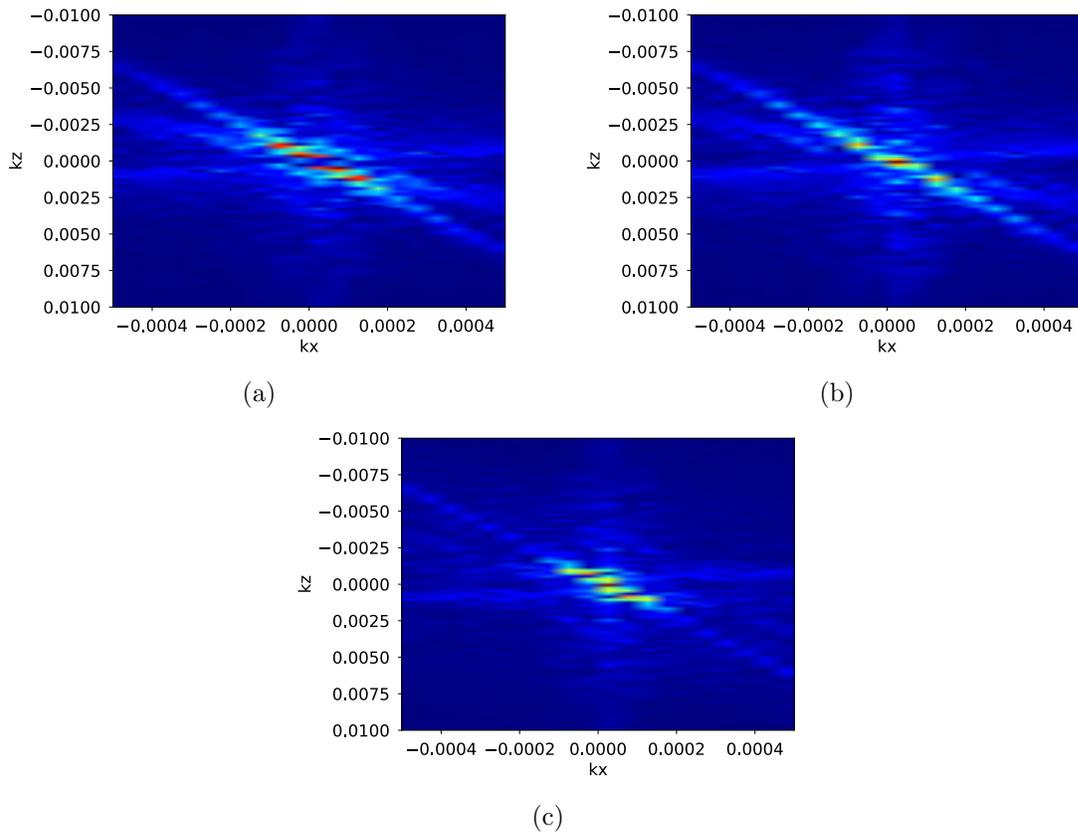


Figure 3.11: Fourier spectra of final model updates in (a)  $v_h$ , (b)  $\epsilon$ , and (c)  $\delta$  using  $(v_h, \epsilon, \delta)$ .

the PSFs in  $\epsilon$  in  $(v_h, \epsilon, \delta)$  has the opposite (Figure 3.15). These PSFs are computed with only a positive perturbation in velocity in a homogeneous isotropic background. This indicates that block (2,1) in the Hessian matrix of  $(v_h, \epsilon, \delta)$  also has the opposite polarity from other blocks (Figure 3.16(b)).

Mathematically, the correlation between model updates and gradients is a result of the starting models and the Jacobian matrices that control the transformation from one set of parameters to another. Recall that the gradients are first computed in  $(c_{ij})$ , and then chain rules are applied with the Jacobian matrices

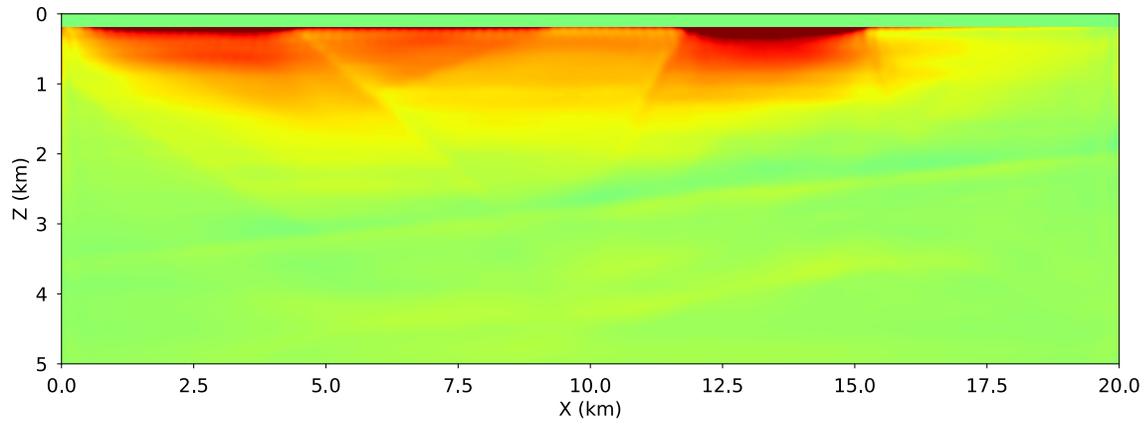
$$J_{(v_z, \epsilon, \delta) \rightarrow (c_{ij})} = \begin{bmatrix} \frac{\partial c_{11}}{\partial v_z} & \frac{\partial c_{11}}{\partial \epsilon} & \frac{\partial c_{11}}{\partial \delta} \\ \frac{\partial c_{13}}{\partial v_z} & \frac{\partial c_{13}}{\partial \epsilon} & \frac{\partial c_{13}}{\partial \delta} \\ \frac{\partial c_{33}}{\partial v_z} & \frac{\partial c_{33}}{\partial \epsilon} & \frac{\partial c_{33}}{\partial \delta} \end{bmatrix} = \begin{bmatrix} 2v_z(1+2\epsilon) & 2v_z^2 & 0 \\ 2v_z\sqrt{1+2\delta} & 0 & \frac{v_z^2}{\sqrt{1+2\delta}} \\ 2v_z & 0 & 0 \end{bmatrix} \quad (3.3)$$

$$J_{(v_h, \epsilon, \delta) \rightarrow (c_{ij})} = \begin{bmatrix} \frac{\partial c_{11}}{\partial v_h} & \frac{\partial c_{11}}{\partial \epsilon} & \frac{\partial c_{11}}{\partial \delta} \\ \frac{\partial c_{13}}{\partial v_h} & \frac{\partial c_{13}}{\partial \epsilon} & \frac{\partial c_{13}}{\partial \delta} \\ \frac{\partial c_{33}}{\partial v_h} & \frac{\partial c_{33}}{\partial \epsilon} & \frac{\partial c_{33}}{\partial \delta} \end{bmatrix} = \begin{bmatrix} 2v_h & 0 & 0 \\ \frac{2v_h\sqrt{1+2\delta}}{1+2\epsilon} & -\frac{2v_h^2\sqrt{1+2\delta}}{(1+2\epsilon)^2} & \frac{v_h^2}{(1+2\epsilon)\sqrt{1+2\delta}} \\ \frac{2v_h}{1+2\epsilon} & -\frac{2v_h^2}{(1+2\epsilon)^2} & 0 \end{bmatrix} \quad (3.4)$$

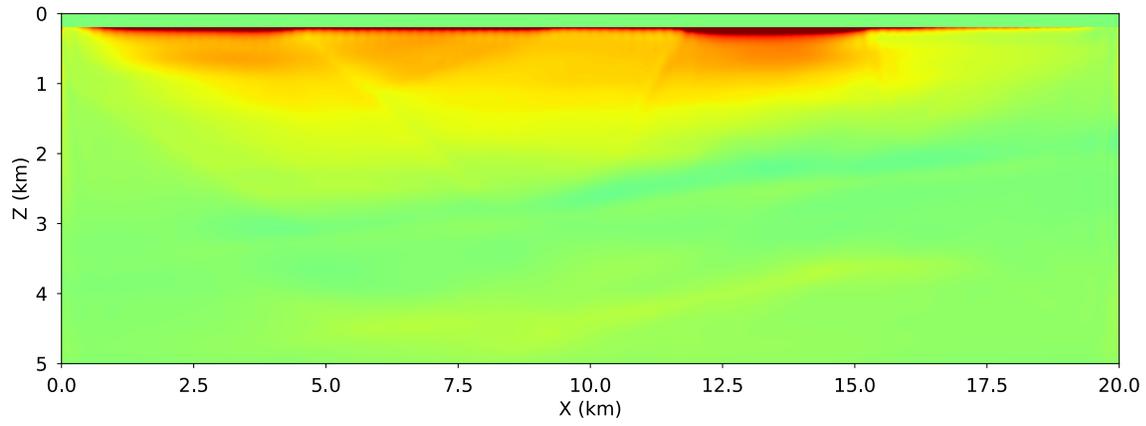
Notice the negative signs in the second column of the Jacobian matrix transformation between  $(v_h, \epsilon, \delta)$  and  $(c_{ij})$  in Equation 3.4. This is the reason why the  $\epsilon$  update is negatively correlated with updates in the horizontal velocity  $v_h$  and  $\delta$ .

## PARAMETER CROSSTALK AND NEWTON'S METHOD

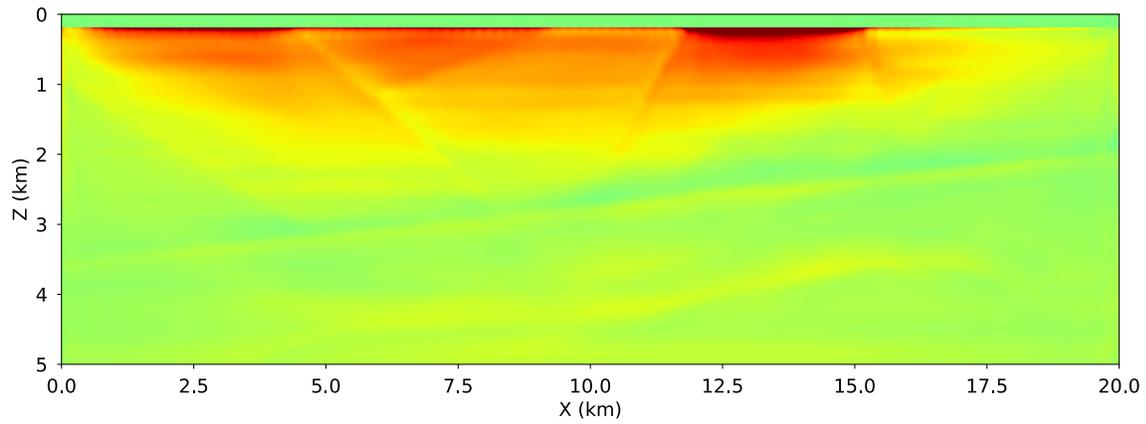
The synthetic example in the previous section also reveals the crosstalk between parameters. Comparing the differences between the true and initial models in vertical velocity, Figure 3.6(a), in horizontal velocity, Figure 3.7(a), in  $\epsilon$ , Figure 3.8(a), and in  $\delta$ , Figure 3.9(a), one can notice the dome-like structure, indicated by the arrow number (3), appears only in velocities. In the final models, this structure, however, is present in both Thomsen parameters, Figures 3.8(b) and 3.8(c) for  $\epsilon$ , and Figures 3.9(b) and 3.9(c) for  $\delta$ , regardless of parameterization. Because correlation and crosstalk between parameters are reflected in the Hessian matrix, I employ a Newton's



(a)

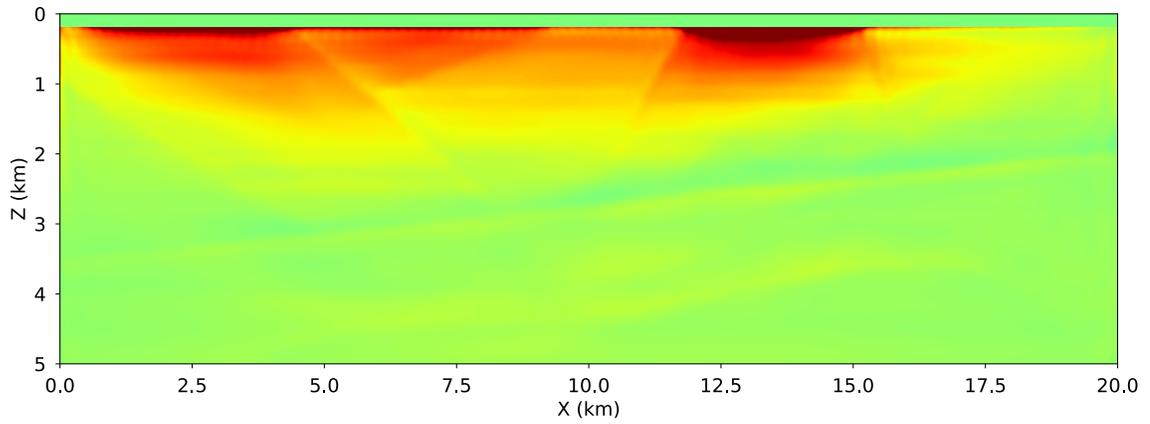


(b)

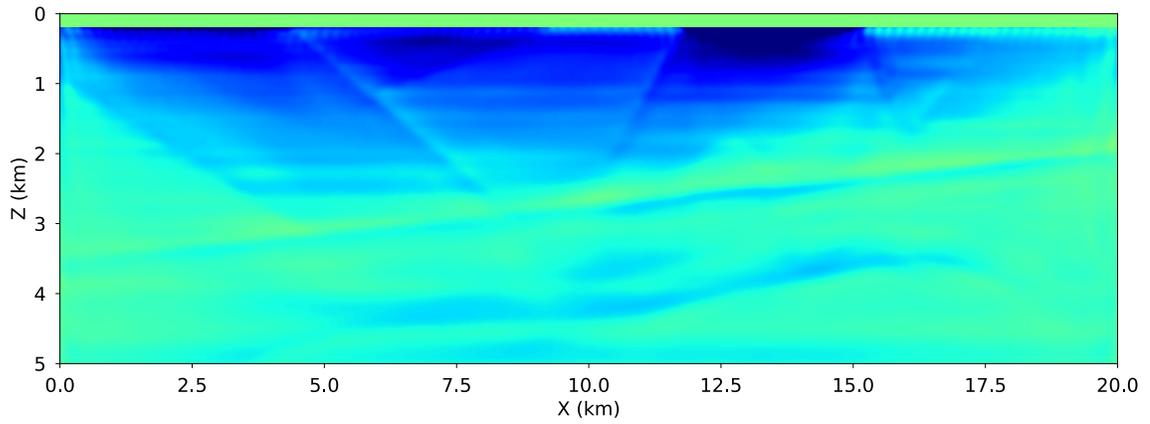


(c)

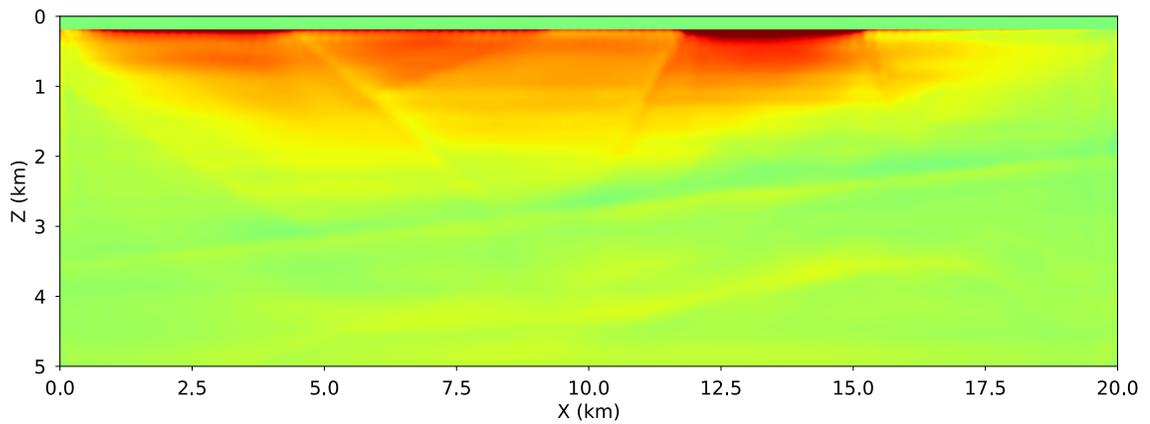
Figure 3.12: Initial gradients in (a)  $v_z$ , (b)  $\epsilon$ , and (c)  $\delta$  using  $(v_z, \epsilon, \delta)$ .



(a)



(b)



(c)

Figure 3.13: Initial gradients in (a)  $v_h$ , (b)  $\epsilon$ , and (c)  $\delta$  using  $(v_h, \epsilon, \delta)$ .

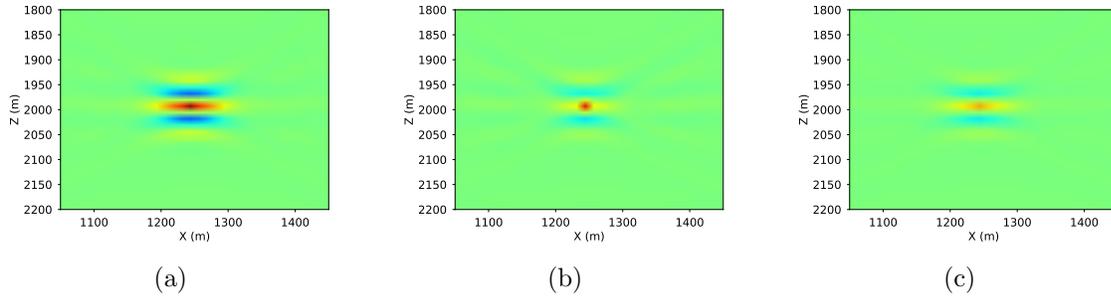


Figure 3.14: PSFs in (a)  $v_z$ , (b)  $\epsilon$ , and (c)  $\delta$  using  $(v_z, \epsilon, \delta)$ .

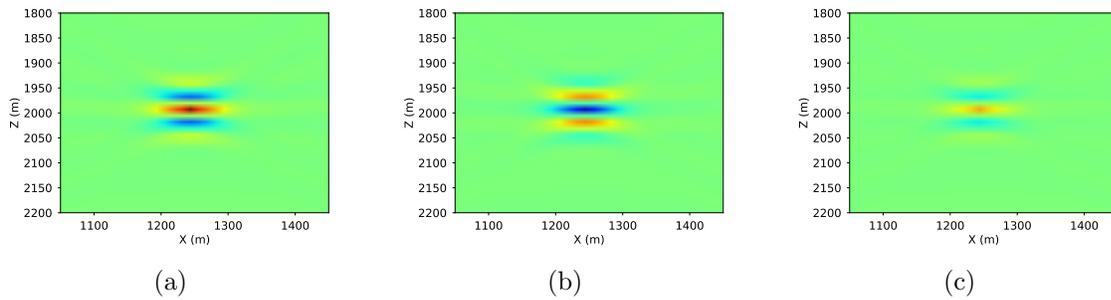


Figure 3.15: PSFs in (a)  $v_h$ , (b)  $\epsilon$ , and (c)  $\delta$  using  $(v_h, \epsilon, \delta)$ .

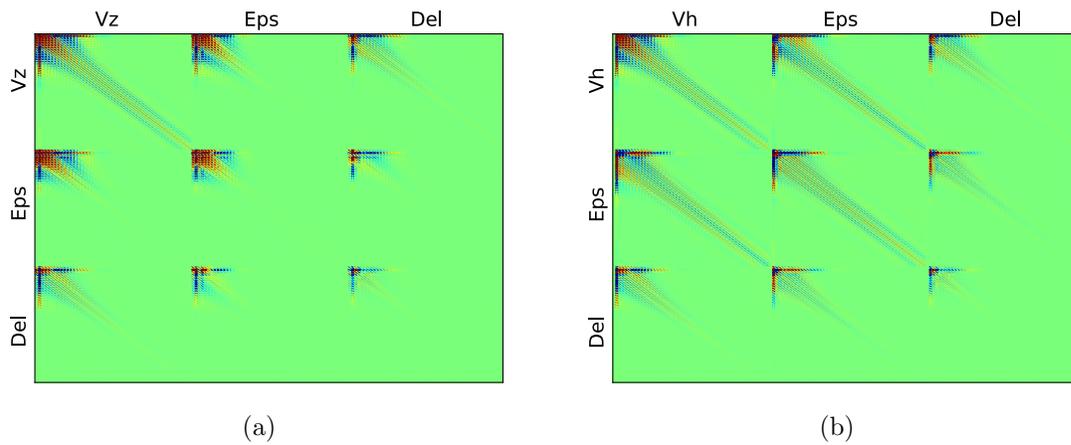


Figure 3.16: Hessian matrices: (a) for  $(v_z, \epsilon, \delta)$  and (b)  $(v_h, \epsilon, \delta)$ .

method in an attempt to remove this crosstalk. The LBFGS solver chosen before also tries to capture some second-order information in the Hessian matrix, but only in a low-rank type of approximation by a small number of gradient vectors, and therefore unsurprisingly fails to reduce the crosstalk.

A Newton search direction,  $p$ , is computed as the solution of

$$Hp = -g, \tag{3.5}$$

where  $H$  is the Hessian matrix and  $g$  is the gradient vector. The Hessian matrix of the FWI objective function is most of the time too computationally expensive to form explicitly, but only the multiplication with a model-space vector is affordable. As a result, Newton Equation 3.5 can only be solved by iterative methods, such as conjugate gradients (CG). This means a Newton solver usually consists of two nested loops. The inner loop iterates CG to find a Newton search direction by solving Equation 3.5. Once a search direction is returned, each outer loop iteration performs a line search and update the models. I implement a Newton solver based on Algorithm 7.1 in Nocedal and Wright (2006) (1). The inner loop is a modified linear CG with early termination criteria and safeguards against indefinite matrices. The line search routine is reused from the LBFGS solver.

---

**Algorithm 1** Pseudocode for Newton’s methods.

---

- 1: **for**  $k = 0$  to  $N$  **do**
  - 2:   Solve  $H_k p_k = -g_k$  with linear CG
  - 3:   Terminate inner loop if  $\|H_k p_k + g_k\| < \varepsilon_k$  or  $p_k^T H_k p_k \leq 0$
  - 4:   Line search for  $\alpha_k$
  - 5:   Update  $x_{k+1} = x_k + \alpha_k p_k$
  - 6: **end for**
- 

The inversion results with Newton solver are shown in Figure 3.18 where  $H$  is the full Hessian matrix and in Figure 3.19 where  $H$  is the approximate Gauss-Newton matrix. Both of these inversions are in  $(v_z, \epsilon, \delta)$ . I observe that the dome structure, pointed out by arrow number (3), is still present in all of the final models. This seems to illustrate that Newton’s method, with either full or approximate Hessians, fails

to reduce the crosstalk. Furthermore, the computation performed by the Newton solver actually exceeds the LBFGS solver. Although it only takes 20 iterations of the outer loop for Newton’s methods to reach the same level of objective function (Figure 3.17(a)), the number of function evaluations in Newton’s methods actually doubles (Figure 3.17(b)). This increase in computational cost of Newton’s methods is also observed by Métivier et al. (2014).

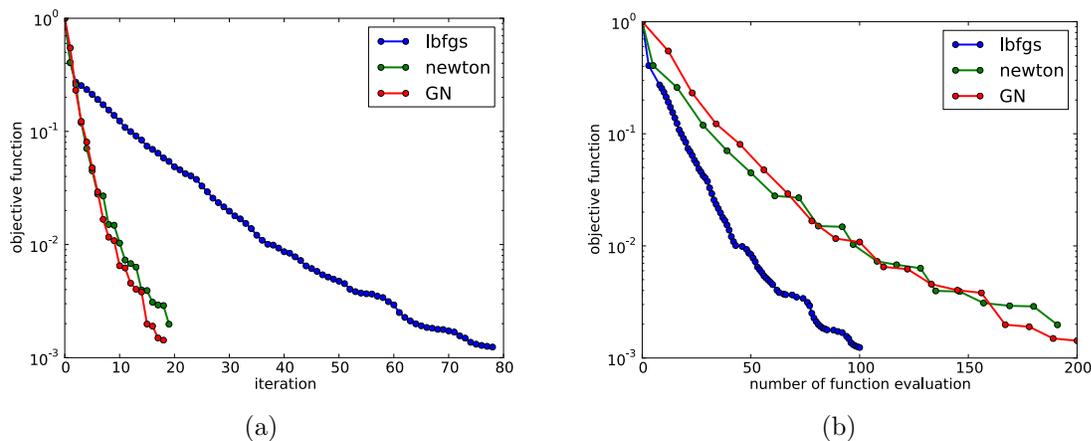


Figure 3.17: Comparisons of LBFGS, Newton, and Gauss-Newton solvers in: (a) iteration and (b) number of function and gradient evaluations.

Newton’s methods in conjunction with the second-order adjoint state method to compute the action of the FWI Hessian have been studied in the context of multi-parameter inversion. Baumstein (2014) modified subspace methods (Kennett et al., 1988) using the Hessian to find coefficients of linear combinations of the gradients that most reduce the objective function. The advantage of this technique is that it requires a small number of Hessian applications (square of number of parameter classes), which saves a lot of computation compared to Newton’s methods. Baumstein (2014) shows that for parameters with different wavenumber spectra, this technique does reduce inter-parameter crosstalk. My experiment with synthetic models in which different parameters have the same wavenumber content, subspace methods seem to struggle. Taking a step back, I realize the connection between parameter scaling, subspace methods, and Newton’s methods. While heuristic scaling attempts to approximate the inverse Hessian block matrix by a diagonal matrix, subspace methods do that

with a non-blocked matrix.

Métivier et al. (2015) use Newton’s methods in visco-acoustic variable density FWI and show that estimation of attenuation seems to improve. The inverted models they obtain, however, are still contaminated by crosstalk artifacts. Pan et al. (2016) successfully apply Newton’s methods to eliminate the crosstalk in elastic FWI for a horizontal transverse isotropic (HTI) medium. Their synthetic models are, however, simple and small enough that the Hessian matrix is constructed explicitly and the Newton Equation 3.5 is solved accurately by SVD. Pan et al. (2017) try to reduce parameter crosstalks in an adaptive subtraction manner in isotropic elastic FWI and obtain some improvements. Their method can be considered a simplified version of subspace methods.

## PARAMETERIZATION ANALYSIS

There are many different ways to formulate an acoustic anisotropic FWI problem. In this section I evaluate eight different parameterizations that have been studied and suggested by Plessix and Cao (2011); Gholami et al. (2013); Guitton and Alkhalifah (2017):  $(v_z, \epsilon, \delta)$ ,  $(v_h, \epsilon, \delta)$ ,  $(v_n, \eta, \delta)$ ,  $(v_h, \epsilon, \eta)$ ,  $(v_z, v_h, \delta)$ ,  $(v_n, v_h, \delta)$ ,  $(v_z, v_n, v_h)$ , and  $(c_{ij})$ . I perform inversions using these parameters starting from the same initial models with two different acquisitions: short offset streamer, 5 km, and uniform with maximum offset up to 20 km. Figure 3.20(a) plots the objective functions with number of evaluation. At a first glance, it seems that all parameterizations perform equally well and all inversions converge. Figure 3.20(b), which shows the objective functions in log scale, classifies these parameterizations into two groups. The first four parameterizations are those with one velocity and two Thomsen parameters, and the second group includes parameterizations with two velocities, three velocities, and stiffnesses. This figure shows that parameterizing with one velocity and two Thomsen parameters results in a slightly lower objective function. The difference between two groups in terms of objective function reduction is small, albeit noticeable.

I also compute and compare the normalized model residuals in  $v_z$ ,  $v_h$ , and  $v_n$  as

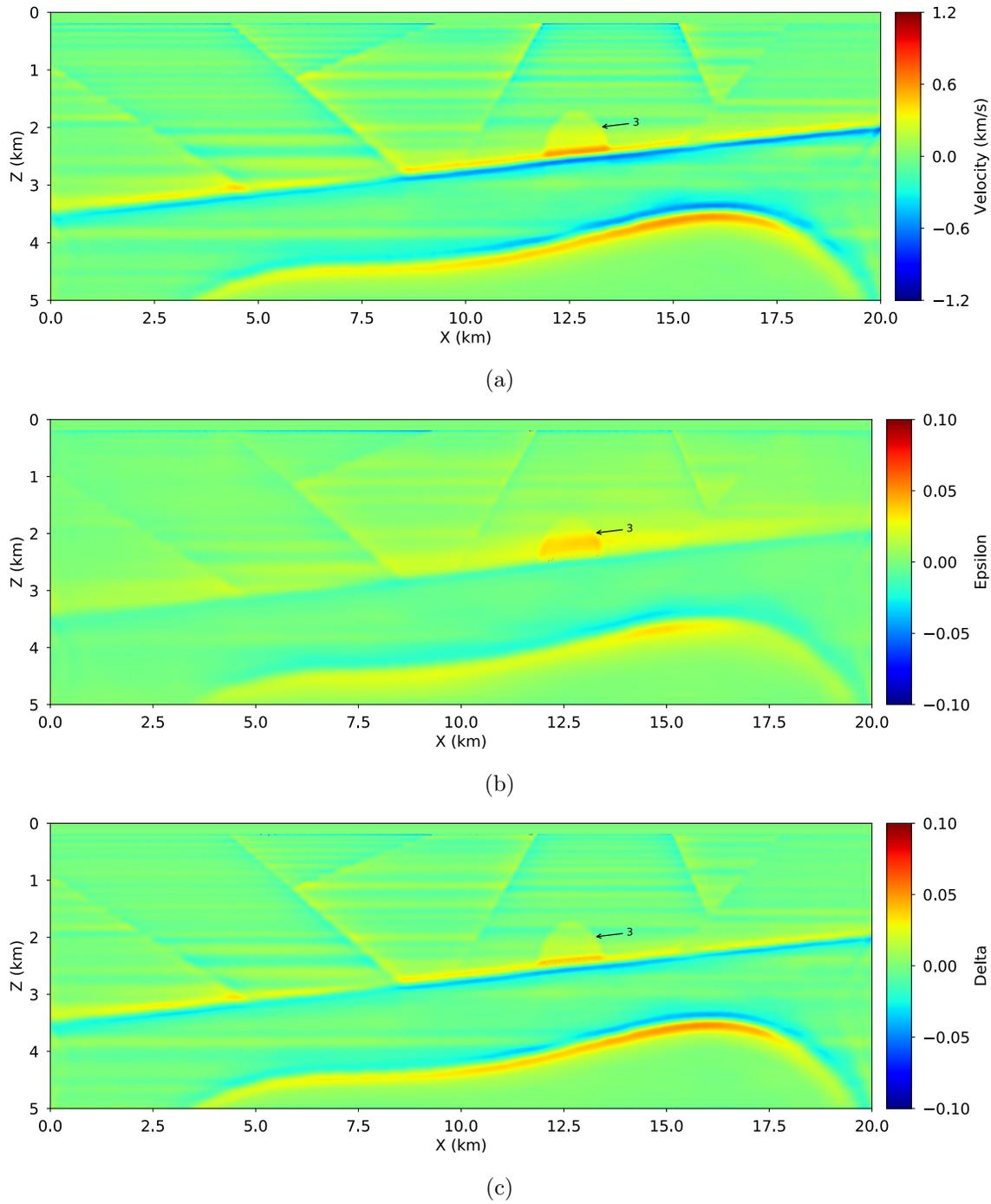


Figure 3.18: Final model updates in (a)  $v_z$ , (b)  $\epsilon$ , and (c)  $\delta$  using a Newton solver with full Hessian and  $(v_z, \epsilon, \delta)$  parameters.

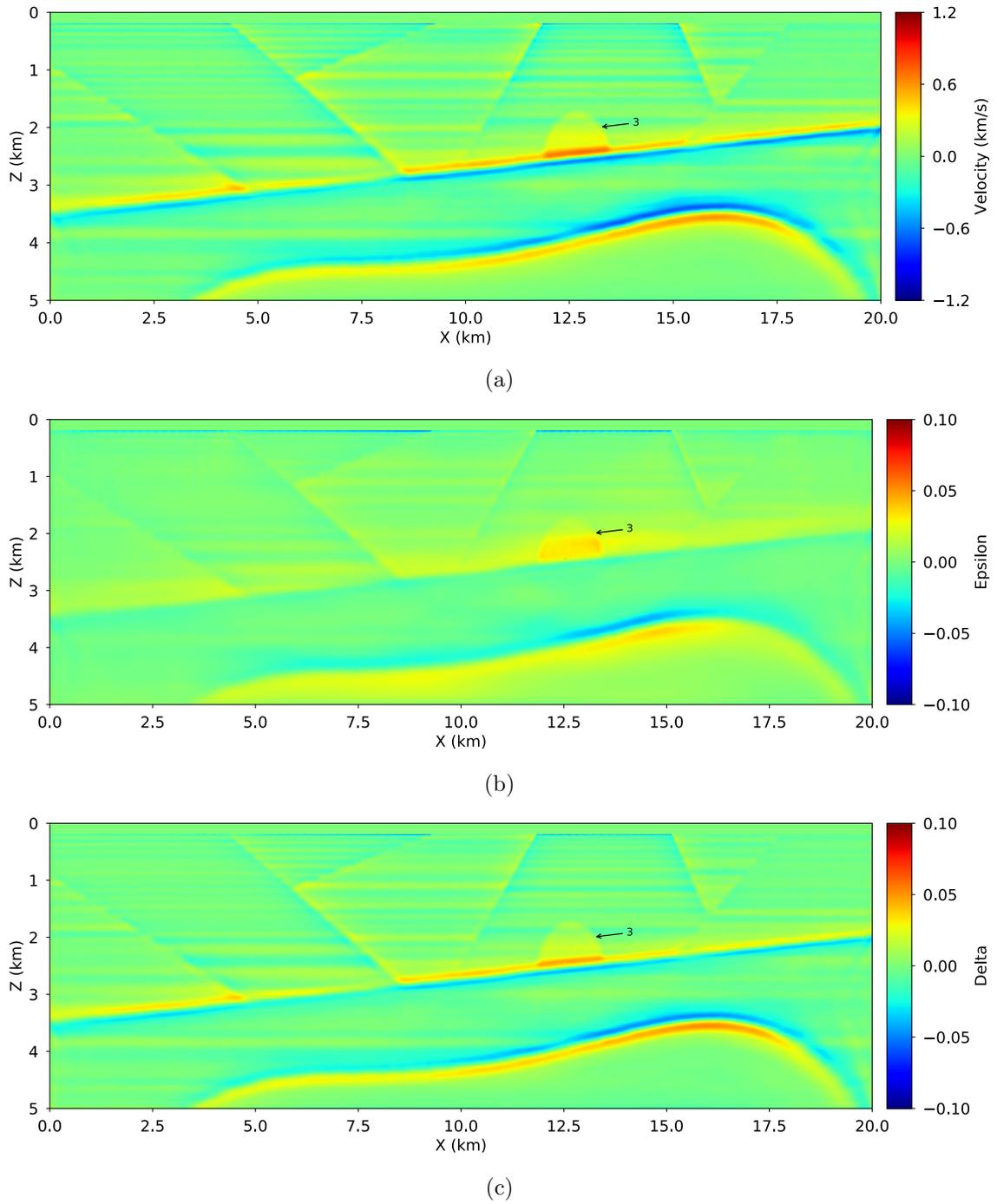


Figure 3.19: Final model updates in (a)  $v_z$ , (b)  $\epsilon$ , and (c)  $\delta$  using a Newton solver with Gauss-Newton Hessian and  $(v_z, \epsilon, \delta)$  parameters.

a function of evaluation. This is defined as the  $\ell_2$  norm of the normalized difference between the  $k^{\text{th}}$  iterate and the true model. For  $v_z$  it is, for example

$$r_{v_z}^k = \left\| \frac{v_z^k - v_z^{\text{true}}}{v_z^{\text{true}}} \right\|_2^2. \quad (3.6)$$

Figure 3.21 shows the model residuals in velocities for different parameterizations. Similar to Figure 3.20(b), the final model residuals of the inversions with one velocity and two Thomsen parameters are lower than those of the other parameterizations. Similar plots for uniform acquisition, Figures 3.22 and 3.23, illustrate the same observation: parameterizing acoustic anisotropic FWI with one velocity and two Thomsen parameters leads to a better convergence. The “bumps” on the model residual curves are where the line search routine tries out different step lengths to find a satisfactory one. Comparing the final RTM images, Figures 3.24-3.31, I also notice that parameterizations with one velocity and two Thomsen parameters result in better focusing and higher quality images, particularly at the fault indicated by the arrow. This is also observed in elastic VTI FWI (Kamath and Tsvankin, 2017). Among the four parameterizations with one velocity and two Thomsen parameters, it is hard to draw a definite comparison given the band-limited nature of the data.

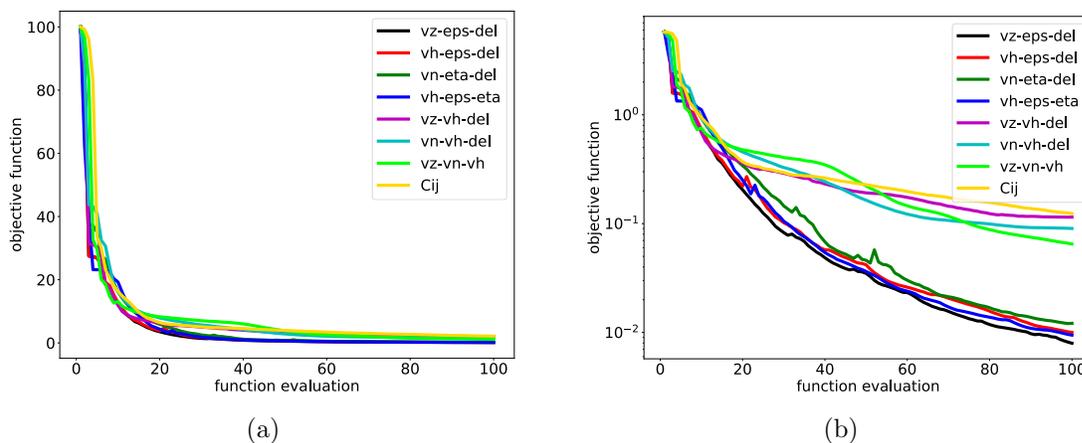


Figure 3.20: (a) Objective function with number of evaluation for different parameterization using short offset data. (b) Final objective functions.

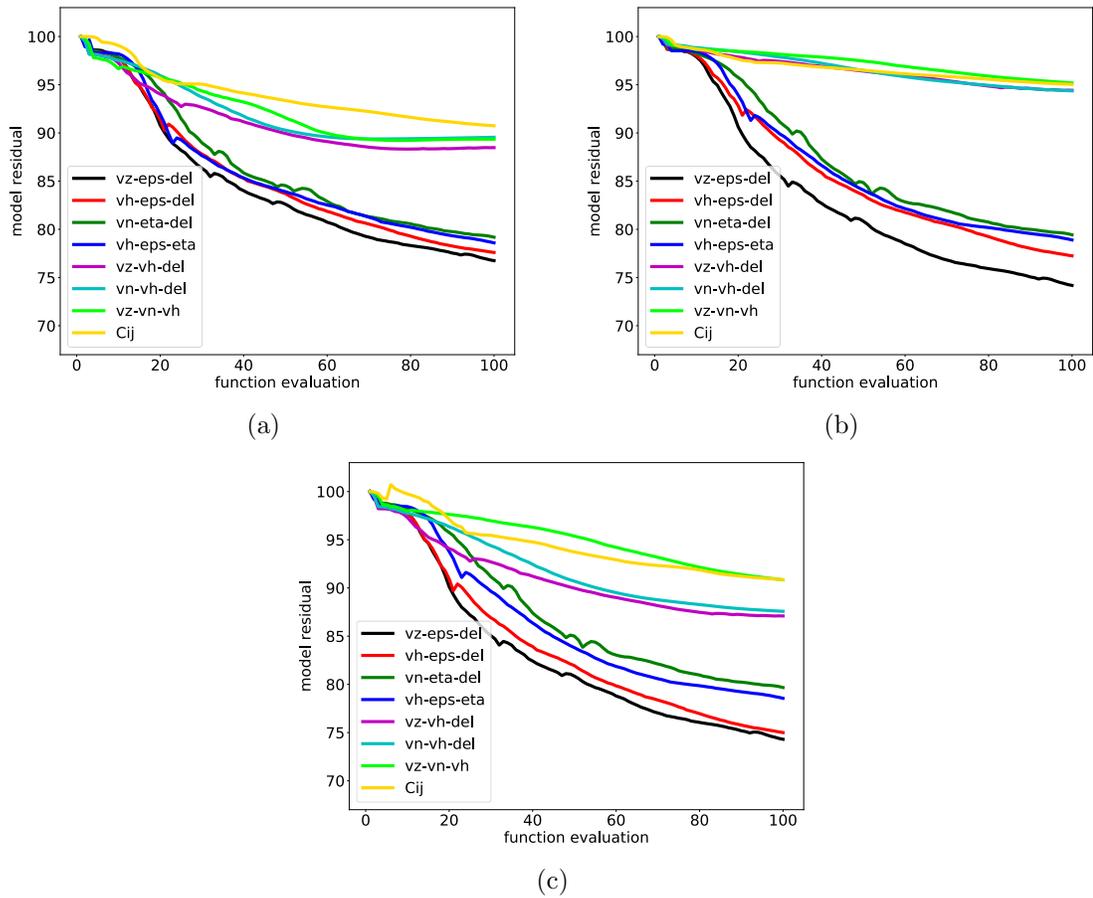


Figure 3.21: Model residuals in (a)  $v_z$ , (b)  $v_h$ , and (c)  $v_n$  for different parameterization using short offset data. .

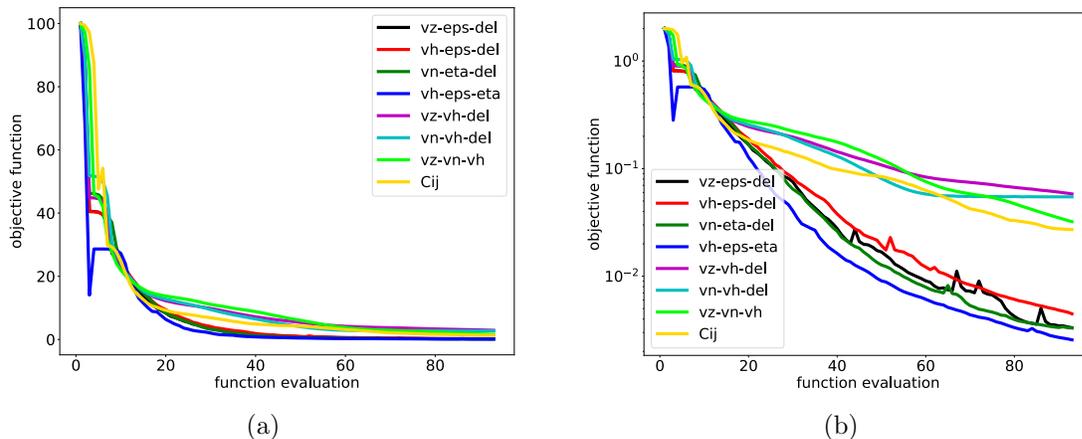


Figure 3.22: (a) Objective function with number of evaluation for different parameterization using short offset data. (b) Final objective functions.

## MONO-PARAMETER VERSUS MULTI-PARAMETER

The previous example seems to indicate that using a combination of one velocity and two anisotropic parameters gives the best inversion results. In such parameterization, velocity is influential at all scattering angles and the inversion is expected to recover a wide range of wavenumbers in velocity. On the other hand, velocity leaves very strong imprints on the other less sensitive parameters, making their final inverted models unreliable. Now one question rises is whether it is necessary to invert for the two Thomsen parameters. Not doing so would reduce the parameter space and save some algebraic calculation. An argument based on computational cost would, however, not justify it because the most expensive component in FWI is the wave propagation, which does not depend on the number of parameters being inverted for. In this section I try to quantify the performance of mono-parameter versus multi-parameter waveform inversions using  $(v_z, \epsilon, \delta)$ .

Figure 3.32 plots the histograms of initial and final velocity errors for three inversions, in which only velocity, velocity and  $\epsilon$ , and all three parameters are inverted for, starting from the initial models shown in Figure 3.2. The data acquisition is short offset streamer. All three inversions show improvements and the final velocity models

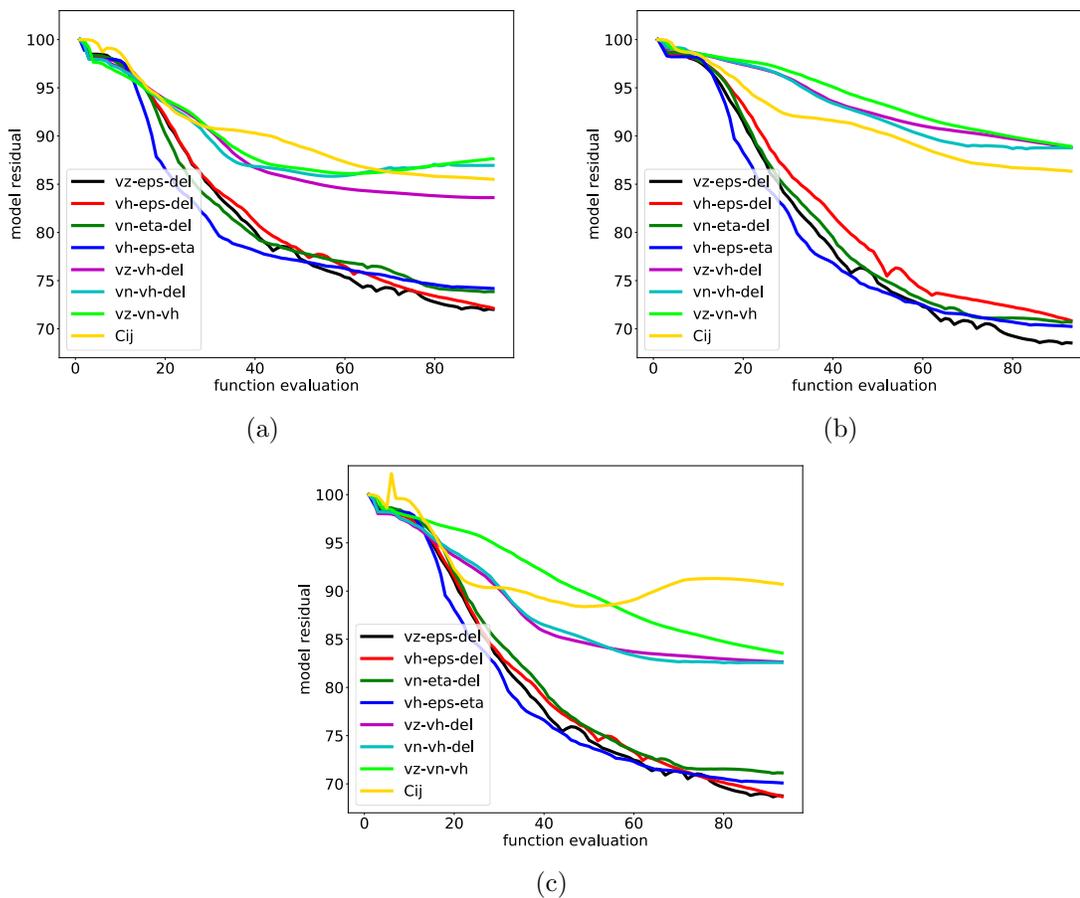


Figure 3.23: Model residuals in (a)  $v_z$ , (b)  $v_h$ , and (c)  $v_n$  for different parameterization using short offset data. .

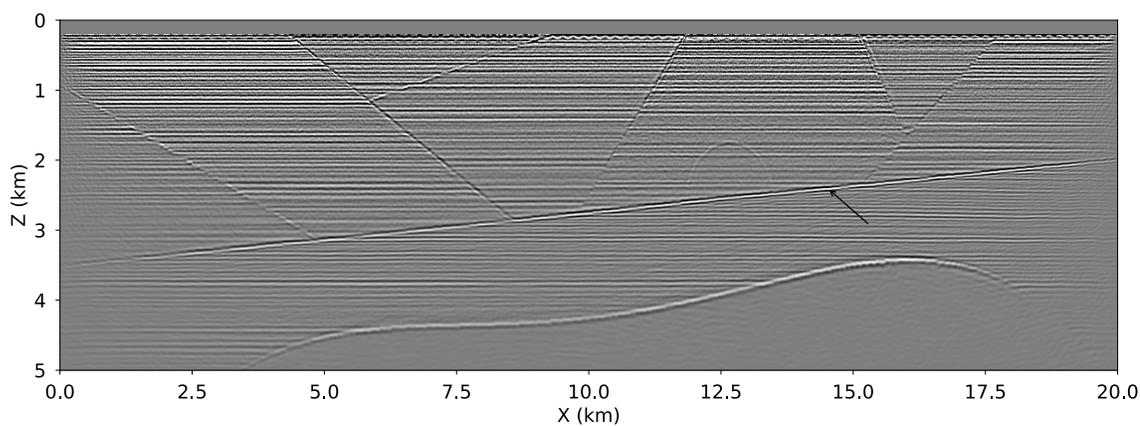


Figure 3.24: Final RTM image of inversion with  $(v_z, \epsilon, \delta)$  parameters.

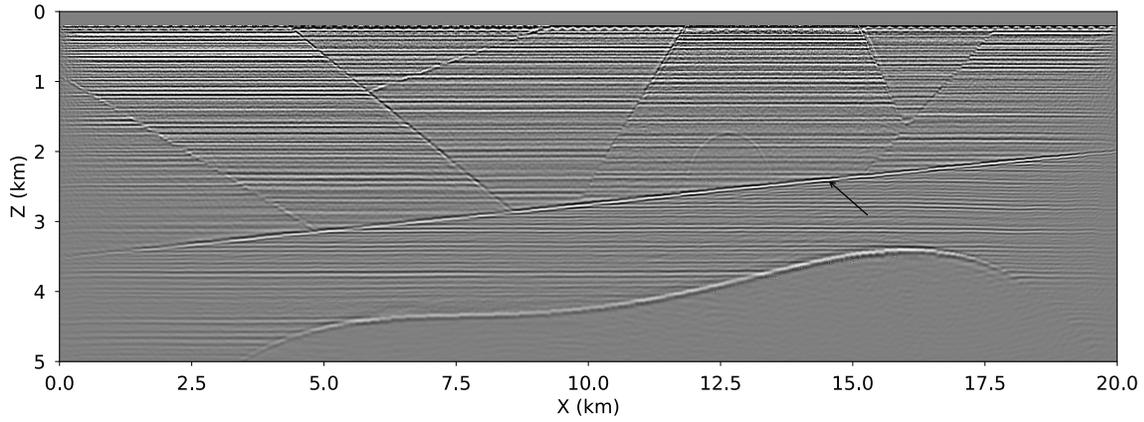


Figure 3.25: Final RTM image of inversion with  $(v_h, \epsilon, \delta)$  parameters.

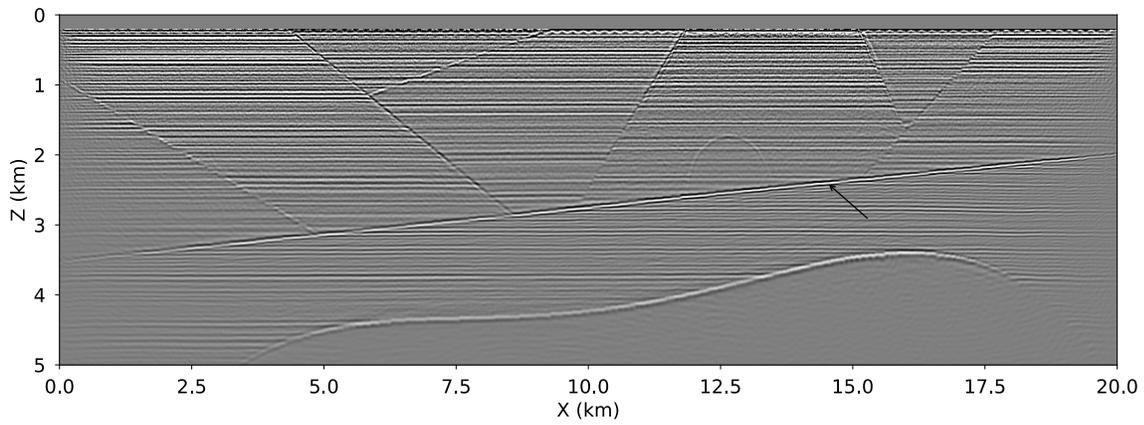


Figure 3.26: Final RTM image of inversion with  $(v_n, \eta, \delta)$  parameters.

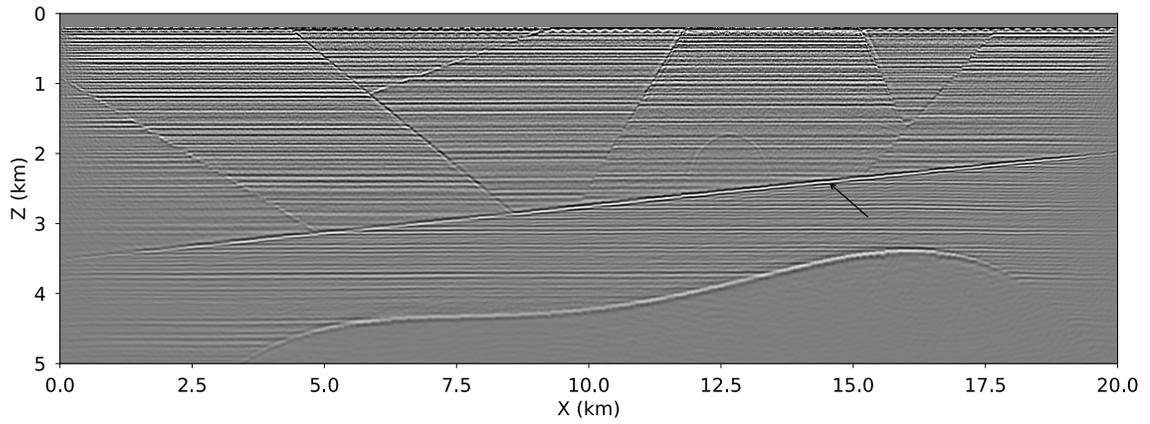


Figure 3.27: Final RTM image of inversion with  $(v_h, \epsilon, \eta)$  parameters.

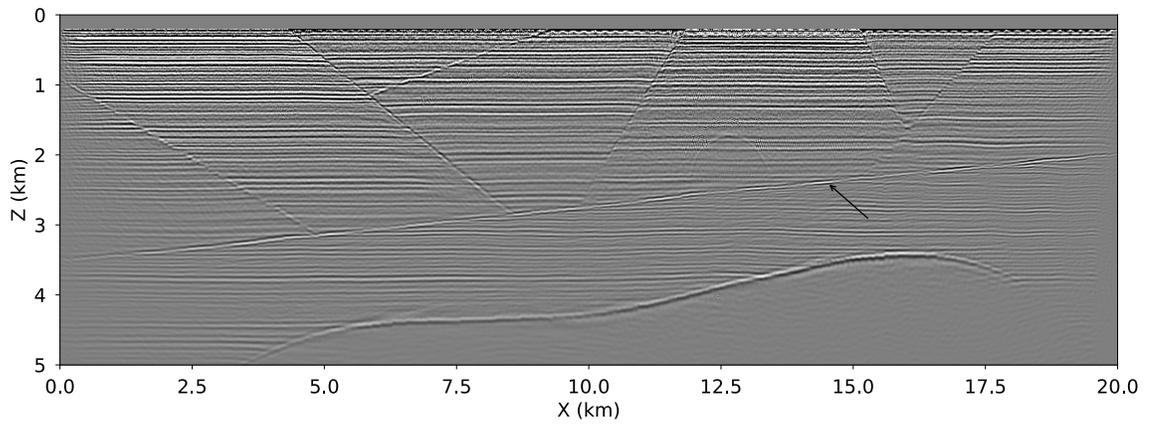


Figure 3.28: Final RTM image of inversion with  $(v_z, v_h, \delta)$  parameters.

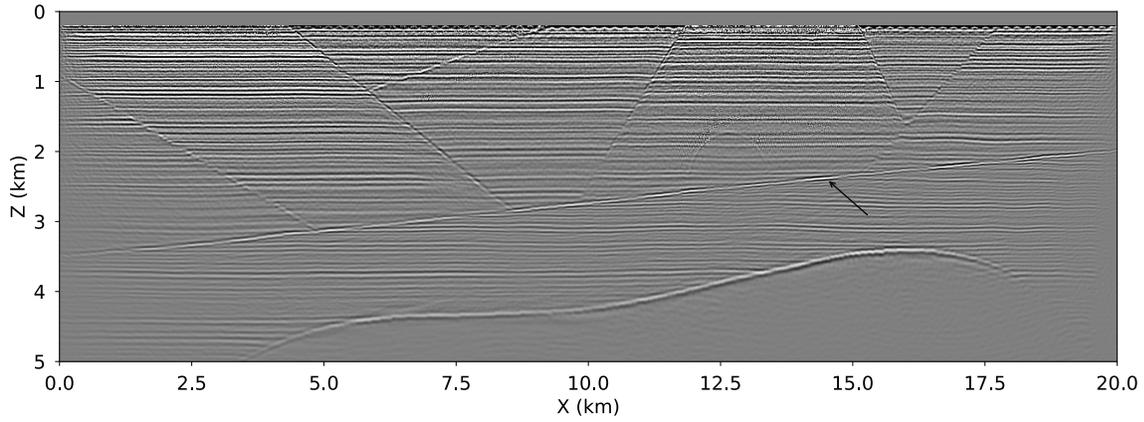


Figure 3.29: Final RTM image of inversion with  $(v_n, v_h, \delta)$  parameters.

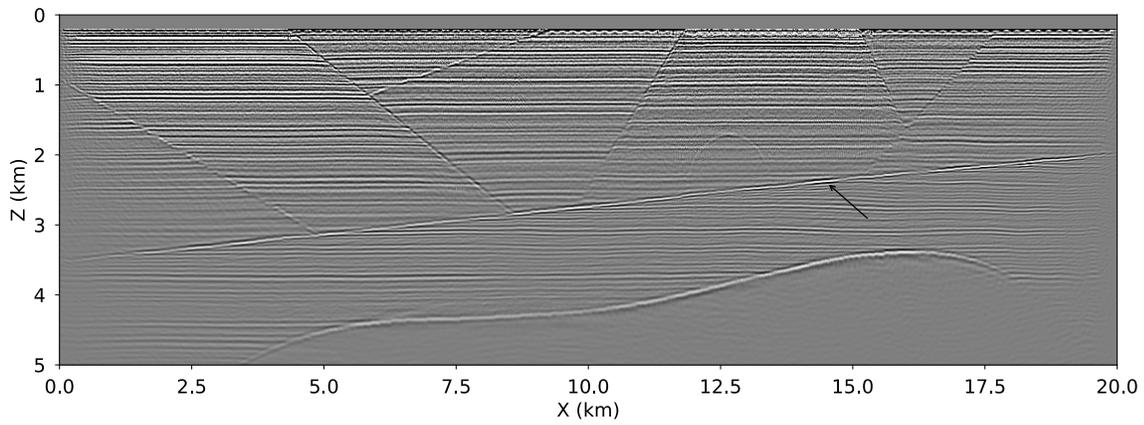


Figure 3.30: Final RTM image of inversion with  $(v_z, v_n, v_h)$  parameters.

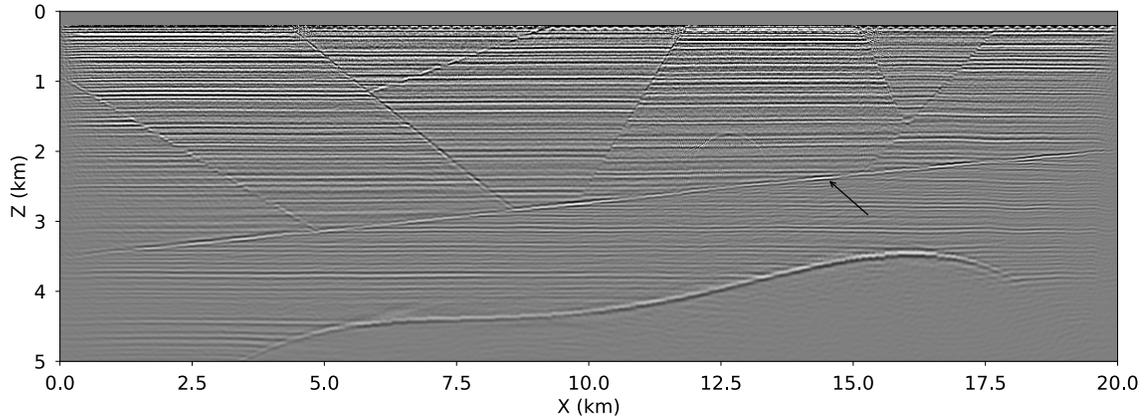


Figure 3.31: Final RTM image of inversion with  $(c_{ij})$  parameters.

are closer to the true velocity model than the initial model, depicted by high concentrations around zero velocity errors. Additionally, this figure shows that inverting for three parameters, Figure 3.32(c), gives better results than a two-parameter inversion, Figure 3.32(b), which in turn is slightly better than a mono-parameter inversion, Figure 3.32(a). The improvement from simultaneously inverting for multiple parameters is more significant for long offset data, Figure 3.33.

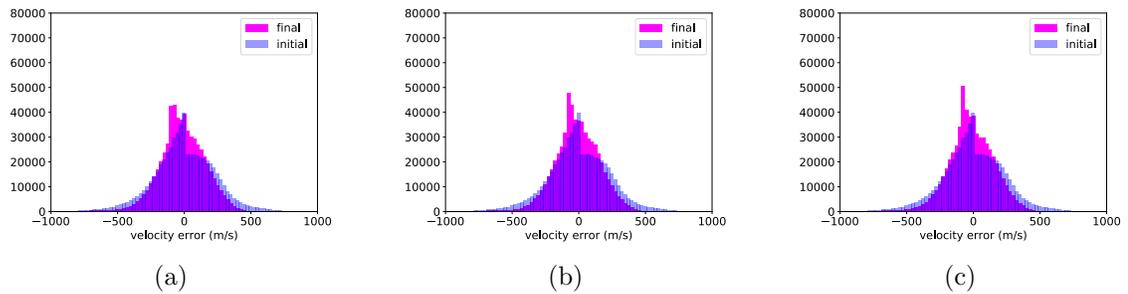


Figure 3.32: Histograms of velocity error: (a) inversion of velocity only, (b) inverting for  $v_z$  and  $\epsilon$ , and (c) inverting for all three parameters with short offset streamer data.

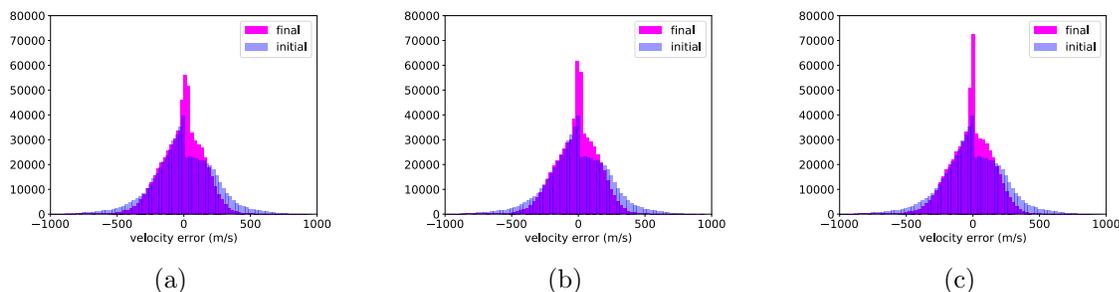


Figure 3.33: Histograms of velocity error: (a) inversion of velocity only, (b) inverting for  $v_z$  and  $\epsilon$ , and (c) inverting for all three parameters with long offset data.

## CONCLUSIONS

Rescaling is necessary and important in multi-parameter FWI. Even though my choice of the normalization coefficients is heuristic, it improved the inversion result. In terms of parameterization, I show that it is better to combine one sensitive parameter, for example velocity, with two less influential anisotropic parameters. This type of combinations induces severe crosstalk from velocity onto the other two parameters. Crosstalk patterns are embedded in the Hessian matrix of the FWI objective function. The use of second-order adjoint state methods allows for the application of this matrix on a model perturbation to be computed efficiently at a cost of two gradients. This gives rise to a truncated Newton's method in which the Newton equation is solved approximately in an iterative manner. However, my experiments show that early termination of the inner-loop iteration due to the negative definiteness of the Hessian matrix can disrupt the ability of Newton's methods to reduce parameter crosstalk. Synthetic examples also reveal that velocity estimation benefits from having a large model search space by incorporating all three parameters rather than updating only the most sensitive parameter. The benefit is more noticeable with long offset data than short offsets.