Applying the truncated Newton method in anacoustic full waveform inversion

Scott Keating and Kris Innanen

ABSTRACT

In this report, the truncated Newton (TN) method and its application to anacoustic FWI is investigated. Larger problems are shown to be efficiently solvable when using TN FWI instead of full Newton FWI. It is also found that stricter forcing terms are required in anacoustic FWI to achieve results comparable to acoustic TN FWI, due mostly to cross-talk effects. This additional cost can partly mitigated by using effective preconditioning and a CG inner solver, or better mitigated using a BFGS inner solver. Even with these strategies, significant cost is incurred to accurately recover Q. If only velocity is desired, it can be effectively recovered for a lower cost.

INTRODUCTION

Full Newton methods provide a high rate of convergence per iteration compared with other local optimization methods, but they are impractical in realistic full waveform inversion (FWI). This is largely due to the excessive memory and computational requirements associated with storing and inverting the Hessian matrix. Unfortunately, the information provided by the Hessian matrix is often important in FWI, and especially crucial in multi-parameter FWI (Operto et al. (2013), Innanen (2014)). The infeasibility of dealing directly with the Hessian matrix leads to consideration of methods which approximate the Newton update without explicitly manipulating the Hessian. One of these, the BFGS method, generates and updates an estimate of the inverse Hessian, whose product with a vector can be quickly and efficiently calculated and requires only a limited number of previous model updates. This method is applied in FWI increasingly often. Another approximation of the Newton update is the truncated Newton method. In this method, the equations for the exact Newton update are solved by an approximate numerical method, such as the conjugate gradient method, which requires only Hessian-vector product information. These products can be efficiently calculated using the adjoint state method. This report focuses on the application of the truncated Newton method in anacoustic FWI, as well as the associated challenges.

THEORY

The truncated Newton method

Many local optimization methods are based on approximations of the Newton method. The Newton method of optimization calculates a model update by minimizing a quadratic approximation of the objective function. The Newton update is calculated by solving

$$Hu = -g \quad , \tag{1}$$

where u is the update, g is the gradient of the objective function with respect to the model parameters, and H is the Hessian, or second derivative of the objective function with respect to the model parameters. This can be impractical, and usually is in FWI due to the large number of model parameters rendering H exceedingly difficult to store and costly to invert.

The truncated Newton method is an approximation of the Newton method which attempts to solve equation 1 by an iterative numerical optimization method. Solving equation 1 is equivalent to minimizing the objective function given by

$$\phi(u) = \frac{1}{2}u^T H u + g^T u \quad . \tag{2}$$

The evaluation of this objective function does not require that we explicitly form the Hessian matrix H, only the product Hu is needed. This product can be efficiently calculated using the adjoint state method, described in a later section. The gradient of this objective function is given by

$$\nabla\phi = Hu + g \quad . \tag{3}$$

Once again, there is no need to explicitly evaluate the Hessian in order to determine $\nabla\phi$. The minimization of the objective function in 2 can be done by any means, but in order to preserve the efficiency that comes with avoiding an explicit evaluation of the Hessian, it is important that the minimization used employs only ϕ and $\nabla\phi$. In this report we focus on the truncated Newton method using a conjugate gradient inner solver. The truncated Newton update is then given by the approximate minimizer of equation 2.

Overall then, in the truncated Newton method there are two optimization problems. The first is the 'outer loop', a nonlinear optimization problem where we minimize the objective function of the problem we are trying solve using the model updates calculated using the TN method. In our case, this is the objective function used in FWI, often given by

$$\Phi(\mathbf{m}) = \frac{1}{2}||\mathbf{d}_{obs} - \mathbf{d}_{mod}||_2^2 \quad , \tag{4}$$

where Φ, the misfit, is a function of the subsurface model $\mathbf{m}$ which measures the discrepancy between the measured data $\mathbf{d}_{obs}$ and the modeled data $\mathbf{d}_{mod}$. The other problem is the 'inner loop', a linear optimization problem in which we minimize the objective in equation 2 to find the model update that will be applied at each step in the outer loop.

By changing the number of iterations used for the CG solver in the inner loop, the cost and accuracy of the truncated Newton method can be altered. When many iterations are employed, the approximate minimizer of 2 will be very close to the solution of equation 1, and so the truncated Newton step will be very close to the Newton step, but the cost of many iterations may be prohibitive. By lowering the number of iterations used the computational cost can be reduced in exchange for a less accurate approximation of the Newton step.

The adjoint state method

The computational savings and reduced memory requirements of the truncated Newton method hinge on the ability to efficiently evaluate the product of the Hessian matrix with a vector without explicitly forming the Hessian matrix. This can be achieved via the adjoint state method. To solve for the Hessian-vector product under the Gauss Newton approximation using the adjoint state method, three wave propagation problems must be solved, one of which is also required for the calculation of the gradient. This means that in an optimization context, evaluation of a Hessian-vector product will require two additional wave propagation problems to be solved. The details of applying the adjoint state method to a truncated Gauss Newton FWI problem are outlined in Metivier et al. (2013). A brief description of the truncated Newton method is given in the appendix.

Convergence rate improvements

There are several means by which the convergence rate for the solution of equation 1 can be increased. Two of the most obvious approaches are to utilize a more efficient optimization method in solving the inner system, and to precondition the inner system to increase the convergence rate of the CG algorithm.

In choosing a more efficient optimization method, it is important that only the gradient of the objective in equation 2 is employed, in order to preserve the efficiency of the TN method. Amongst the methods which consider only gradient information is the BFGS method, which generates an approximation of the inverse Hessian based on previous model updates. The BFGS method is faster than the nonlinear conjugate gradient method in many cases, and can be applied to linear problems. A linear BFGS algorithm should not incur significant extra cost as compared to the conjugate gradient method in the context of FWI. While there is some extra cost, this takes the form of gradient vector sized products, not extra wavefield propagation problems. Most of the cost in FWI is due to such wavefield modeling problems, so it is to be expected that the extra costs incurred by an inner BFGS solver will be negligible relative to these larger costs. The BFGS method does impose a memory requirement of one gradient per iteration performed. This can grow large with a large number of iterations, but can be reduced by employing a limited memory BFGS optimization (L-BFGS).

In lieu of finding a faster solver for inner system, preconditioning can be applied in the conjugate gradient method to help improve convergence speed. Preconditioning of equation 1 requires a means of estimating the product of an approximate H^{-1} with a vector. While there exist numerous methods of generating such estimates for preconditioning, many are unavailable in this problem, due to the lack of access to the Hessian matrix, and the necessity that preconditioner require significantly less storage than the Hessian matrix. The available preconditioning strategies under these restrictions are relatively limited. One option is to use the inverse of a sparse estimate of the Hessian matrix, such as the main diagonal. While this is inexpensive to implement, the effects of such a simple preconditioner are unlikely to significantly improve the convergence rate. Another option is to efficiently generate an approximation of the inverse Hessian using the BFGS method. This allows for the product of an approximate inverse Hessian with a given vector to be efficiently calculated, with the storage requirement of just a small number of previous model updates and gradients.

One straightforward method of applying a BFGS preconditioner uses the previous gradients and updates of the outer loop objective function (eq 4) to generate the preconditioner (Metivier et al. (2013)). This approach faces several implementation challenges. One is the fact that the objective function is generally not the same at each iteration, as in the multiscale approach of FWI the frequency band considered changes at iterations progress. This means that the BFGS approximation of the inverse Hessian will be based on curvature information from points on several objective functions. Similar problems are confronted in applying the BFGS method to FWI, which has been done successfully, but it is reasonable to assume that this is still a source of error. Another problem is the relative paucity of outer iterations with which to generate the BFGS approximation. Crosstalk is undesirable at early iterations as it is at later ones, but outer BFGS preconditioning will have little effect when there are few outer iterations to use, as the inverse Hessian approximation will be poor. This approach will be referred to as outer BFGS preconditioning in this report, following Metivier et al. (2013).

Morales and Nocedal (2000) suggest using instead the information from minimizing the inner problem, equation 2 (which has the same Hessian as the outer problem), in BFGS in order to generate a preconditioner for the next outer iteration. Provided there are enough inner CG iterations to provide a good estimate in BFGS, lack of data with which to generate the approximation is not a problem, but in this approach the BFGS estimate of the inverse Hessian at a given iteration is used to precondition the next iteration. This approach hinges on the assumption that the Hessian matrix at each iteration is similar to that on the previous iteration. If this assumption fails, this method of preconditioning faces serious challenges. This approach will be referred to as inner BFGS preconditioning in this report, following Metivier et al. (2013).

Both of the preconditioning methods discussed above face challenges, and most of the preconditioning challenges outlined thus far in this report have a common cause: the BFGS updates used to generate the preconditioner are not drawn from the same problem as the one being preconditioned. The inverse Hessian being estimated by these preconditioners is not the same as the inverse Hessian in the problem being solved. This limits the efficacy that the preconditioning can have. An alternate approach would be to adopt the inner preconditioned method, but instead of employing updates from the inner problem of the previous outer iteration, using updates from the current problem. This can be done by applying the non-preconditioned CG method to equation 2 for a limited number of steps, then using these steps to generate a BFGS preconditioner for use with the preconditioned conjugate gradient method. This approach adopts the additional cost of the preliminary non-preconditioned CG steps to better estimate an effective preconditioner for the inner problem. In order for this method to be cost effective, this extra cost must be offset by the increase in convergence rate provided by the preconditioning. This approach will be referred to as two step inner BFGS preconditioning in this report.

In this research report, all of the BFGS preconditioning strategies outlined above were tested.

NUMERICAL EXAMPLES

In these experiments, frequency domain finite difference modeling was used, both for generating the "measured data" and internally within the FWI algorithm. In both the truncated Newton and exact Newton methods discussed, the Gauss-Newton approximation was made. A multiscale approach was adopted, wherein six evenly spaced frequencies were

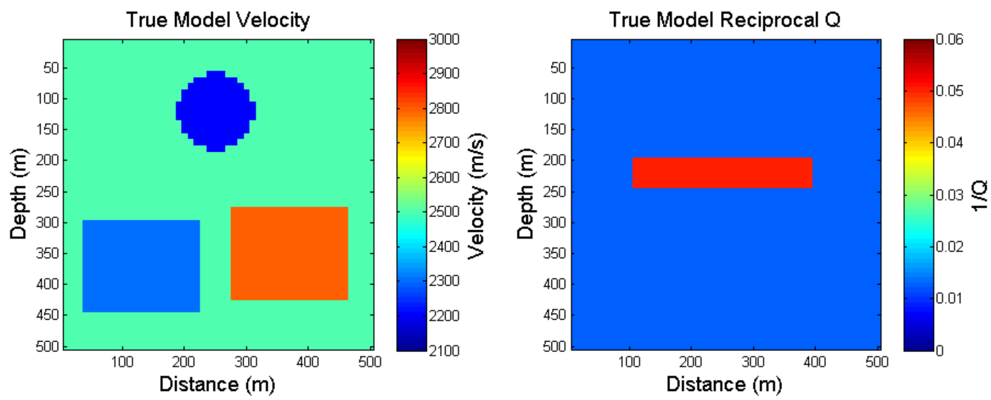

FIG. 1: Velocity model for both the acoustic and anacoustic examples (right). Reciprocal Q model for the anacoustic examples (left). The acoustic model used a reciprocal Q model of zero at every point.

inverted at each iteration, starting with a 1-2 Hz band, with the upper band increasing by 1Hz at each iteration, up to a maximum of 25 Hz at the final iteration. The regularization factor used penalized the gradient of the model update at each iteration. The forward modeling, frequency updating strategy and regularization used are all the same as employed in Keating and Innanen (2016), where they are described in greater detail.

In applying the TN method, a convergence criterion has to be chosen for the solution of the inner system, equation 1. This is specified by a forcing term, η. Convergence is considered to be satisfied when

$$|Hu + g|_2 \leq \eta |g|_2 \quad . \tag{5}$$

By lowering η, we increase the accuracy of the approximate Newton update, but incur additional cost in the inner optimization. A maximum number of allowed inner iterations was also enforced. As the focus of this report is on the comparison of the convergence rates of different approaches, this method was set to an extremely large value, 500 iterations.

An example of acoustic TN FWI

The model used in the acoustic numerical tests done here is shown in figure 1 (right). This model is 50 samples in both width and depth, with a spacing of 10m in both the x and z directions. This small size was chosen to allow for tractable calculation of the exact Gauss-Newton update. The background velocity for this model is 2500m/s, the upper anomaly has a velocity of 2200m/s, the lower left anomaly has a velocity of 2300m/s, and the lower right anomaly has a velocity of 2800m/s. Given the simplicity of the model used, the starting model for each FWI problem was a uniform 2500m/s. Figure 2 (right) shows the result of an acoustic TN FWI applied to the acoustic model, using a forcing term of 10^{-2}. An average of 26 inner iterations per outer iteration were necessary to achieve this level of accuracy. The model is largely well-recovered, with some regions of poor reconstruction on the lower anomalies. Figure 2 (left) demonstrates for comparison the result of a full GN FWI applied to the same model, using the same inversion parameters. While the results are superior to those in the TN result, the difference is not dramatic, and it can be reasonably

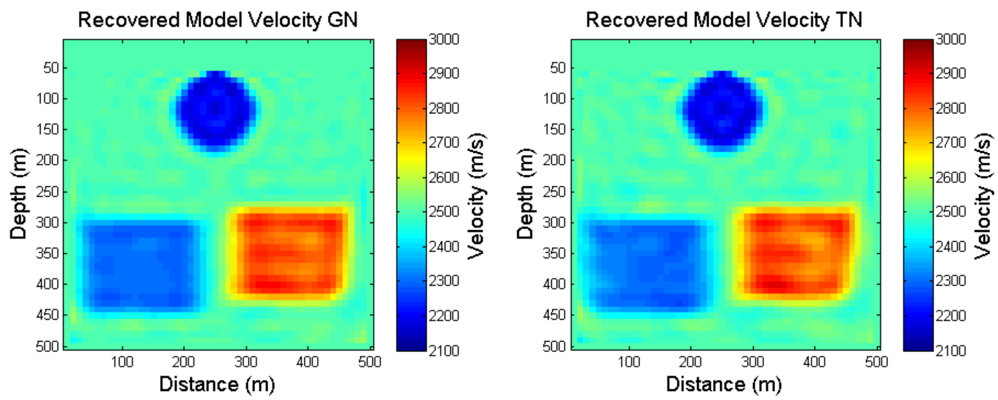

FIG. 2: Recovered velocity from Gauss Newton FWI (left) and TN FWI (right). A forcing term of 10^{-2} was used. The TN result is very similar to the GN one, the forcing used here is acceptable.

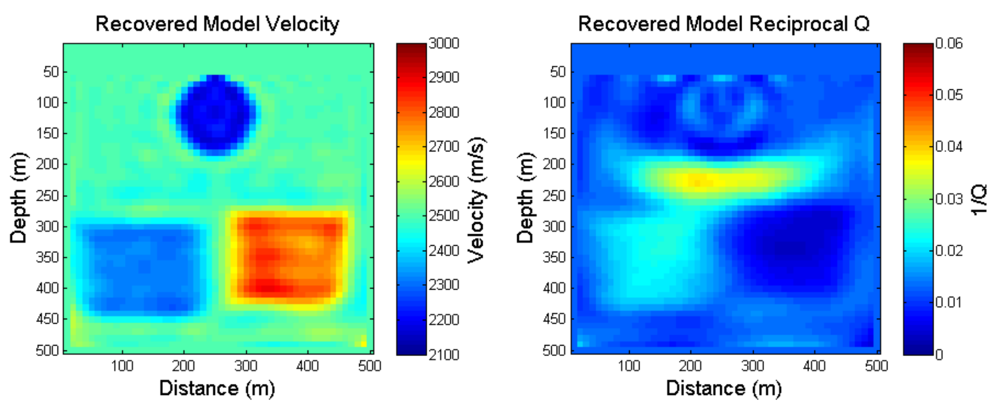

FIG. 3: Result of Truncated Newton FWI with forcing term 10^{-2} for recovered reciprocal Q (left) and slowness squared (right). Cross-talk is pervasive, especially in the recovered Q model.

expected that an increased number of outer FWI iterations would be a viable, cost-effective alternative to increasing the number of inner conjugate gradient iterations if it was desired to improve the TN result to better match the GN one. While the TN method is not exactly replicating the GN results here, it is evident that it is generating update steps of comparable quality, and for very large models the computational cost savings of the TN method are very large. Thus, the $\eta = 10^{-2}$ form of the TN method appears very viable in this case.

An example of anacoustic TN FWI

A more challenging problem is to attempt to extend the successes of the acoustic TN FWI method to the multiparameter anacoustic case. An extended model was used for this example, employing again the reference velocity model shown in figure 1 (right), but including also the Q model on the left of that figure. This model has a background Q of 80, and a small Q anomaly partially obscuring the lower two velocity anomalies with a Q value of 20. This Q anomaly does not overlap spatially with any of the velocity anomalies, which we later use to facilitate the identification of cross-talk effects. Figure 3 shows the result

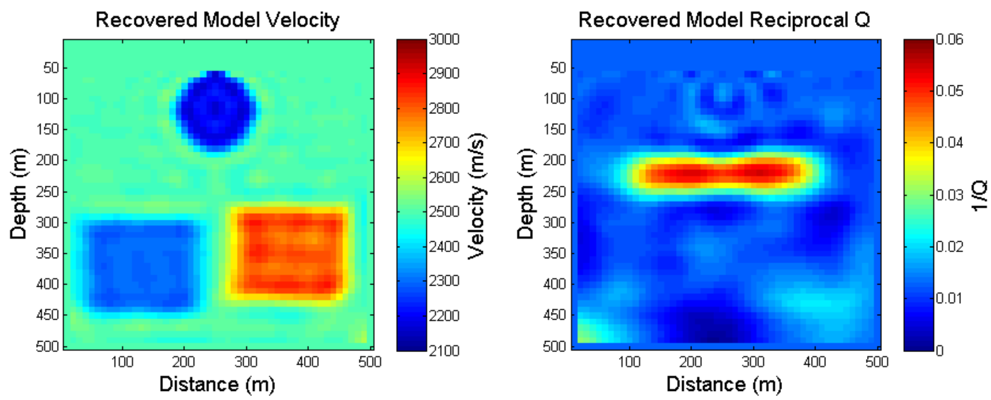

FIG. 4: Result of Gauss Newton FWI for recovered velocity (left) and reciprocal Q (right). The velocity model recovered closely matches the true model (figure 1). The Q anomaly is recovered well, but errors are present at the edges of the Q model. Limited evidence of crosstalk is present at the position of the upper velocity anomaly.

of an anacoustic TN FWI applied to the anacoustic model, using a forcing of 10^{-2}. Here, again, the velocity model is reasonably well reconstructed (though there is some evidence of possible cross-talk), but the recovered Q model suffers from major errors. The significant spatial overlap between the recovered Q anomalies and the velocity anomalies present in the true model strongly suggests that cross-talk has a significant role in this reconstruction. Perhaps the most compelling reason for attempting to account for the action of the Hessian in multiparameter FWI is the mitigation of cross-talk, so in this case the truncated Newton method is not performing as well as is required.

The result of an exact GN FWI is shown in figure 4. Significantly better results are evident here, especially in the recovered Q model. This reinforces the notion that the errors evident in the TN case are not inevitable in this example, and are likely caused by cross-talk, which is well accounted for in the GN FWI. It is to be expected here that an increased number of outer FWI iterations would not significantly improve the results of the TN FWI if cross-talk is the origin of the errors, as this would persist for even a very large number of iterations. Unlike the acoustic case investigated in figure 2, the TN approximation of the Newton update is insufficient here. To improve the performance of the TN method, we can either solve equation more exactly, or we can reformulate the problem such that the recovery of important aspects of the update is more heavily weighted. A means of doing the latter is difficult to determine, while the former suggests either an increased number of CG iterations in solving equation 1 or increasing the rate of convergence in solving 1.

Convergence rate for the inner problem

Figure 5 demonstrates the effect of dramatically decreasing the forcing term for the inner problem. The forcing term in the inner problem has been reduced here to $\eta = 10^{-5}$. The result is a considerable improvement in the result, which closely matches that of the exact GN result in figure 4. This large improvement in the results is associated with an increased computational cost, however. The average number of inner CG iterations per outer FWI iteration has increased to () with this forcing, as compared to 31 iterations needed

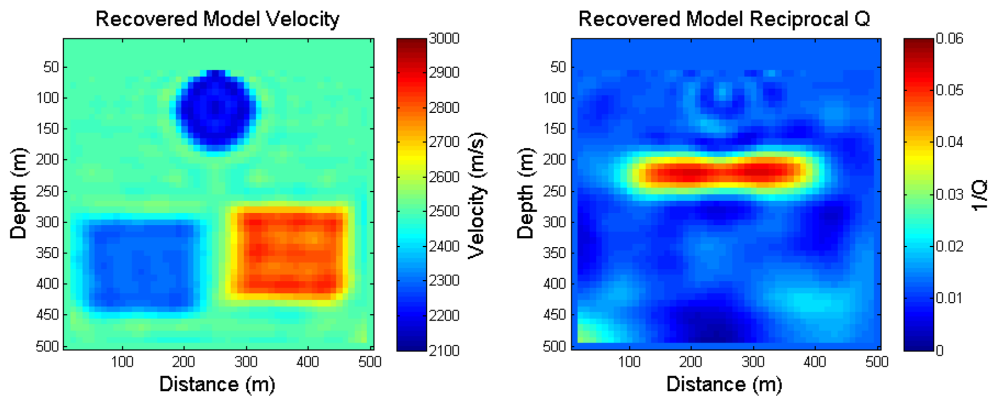

FIG. 5: Result of Truncated Newton FWI with forcing term 10^{-5} for recovered velocity (left) and reciprocal Q (right). Results closely match the GN result from figure 4. Most evident errors originate in the GN update, not the TN approximation of it here.

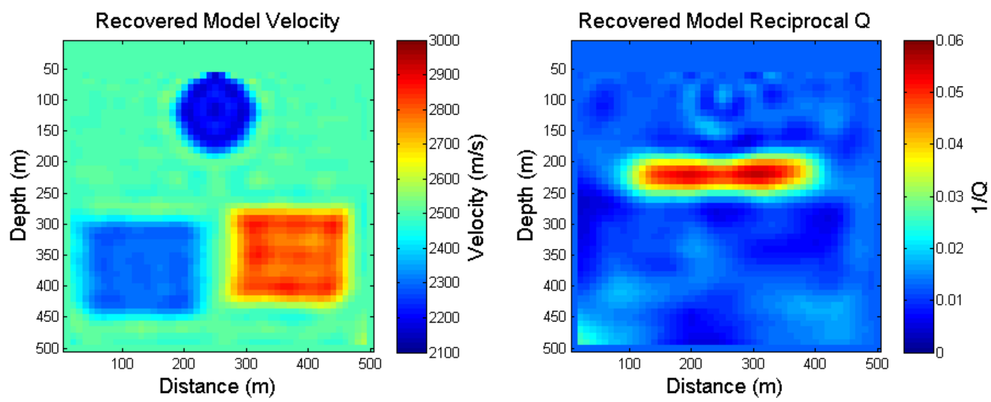

FIG. 6: Result of Truncated Newton FWI with forcing term 10^{-4} for recovered velocity (left) and reciprocal Q (right). Cross-talk present in figure 3 is largely mitigated, though some remains in the Q model at the location of the lower right velocity anomaly.

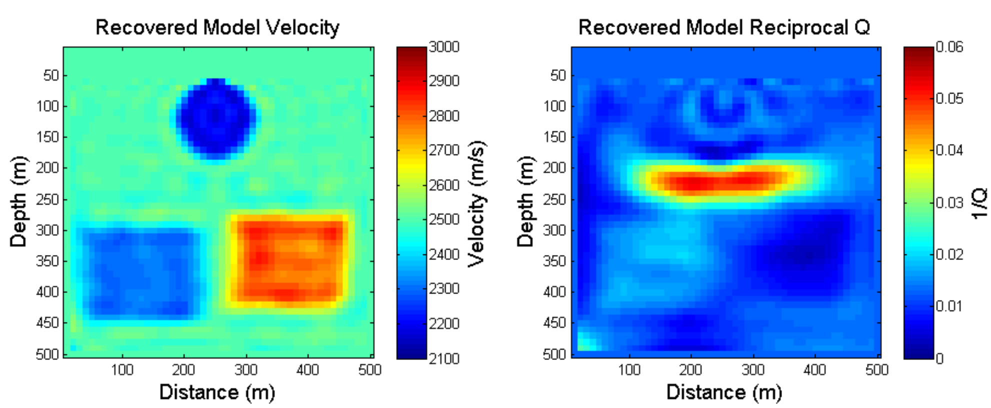

FIG. 7: Result of Truncated Newton FWI with forcing term 10^{-3} for recovered velocity (left) and reciprocal Q (right). Improvements over figure 3 are evident, but significant evidence of cross-talk remains.

for the forcing of 10^{-2} in figure 3. As this level of cost is almost certainly outside the range of any reasonable implementation, it is worth looking at larger forcing terms as well. Figures 6 and 7 show results for forcing terms of 10^{-4} and 10^{-3} respectively, and require an average of 285 and 93 CG iterations per outer iteration. Both results are inferior to those in figure 5, but represent an improvement over the results in figure 3. While figure 6 can be argued to be largely free of significant cross-talk, this is not the case in figure 7. The desirable forcing term then lies somewhere between 10^{-4} and 10^{-3}. The computational cost associated with achieving this level of accuracy is still very high however. This leads to the consideration of preconditioning and alternative inner solvers to increase the convergence rate. In the following examples, a forcing term of 10^{-4} is used to achieve the desired accuracy of the truncated Newton update.

Preconditioning

Tests conducted using the inner BFGS preconditioned TN Newton FWI reached the maximum number of inner iterations before achieving convergence in every outer iteration after the first, that is, for every preconditioned iteration. This result is in keeping with the reported failure of this preconditioning approach in Metivier et al. (2013). This suggests that the assumption made in the inner preconditioning, namely that the Hessian does not vary much from iteration to iteration, fails badly in the FWI problem studied here.

Outer BFGS preconditioning offered more stable results. An average number of 374 inner iterations was necessary in order to achieve convergence, including 13 of 25 outer iterations which reached the maximum number of iterations before achieving this accuracy. While this preconditioning was an improvement over the inner preconditioning, it is still outperformed by the case without preconditioning. The factors discussed above; a lack of previous updates with which to generate a reliable BFGS estimate and a changing objective function are likely both significant contributors to this failure. Altering the frequency updating strategy to perform several outer iterations at the same frequency band would likely help to alleviate the latter problem, but the lack of updates with which to generate a reliable BFGS estimate seems to be unavoidable.

In the first example of two step inner preconditioned TN, 20 preliminary CG iterations were used to generate the preconditioner, and a forcing term of 10^{-4} was used in the inner loop. The average number of total inner iterations used (including those used to precondition) to achieve convergence was 216. This represents a marked improvement over the case with no preconditioning, unlike the other preconditioners investigated. One problem that needs to be approached when using this method for preconditioning TN FWI is the determination of the number of preliminary CG iterations to be used. Increasing this number improves the efficacy of the BFGS approximation, and thus the preconditioner, but increases the number of iterations which do not benefit from any preconditioning at all. It is not obvious which of these factors plays a bigger role. In this research, several values for the number of preliminary iterations were tested. When 50 preliminary CG iterations were employed, an average of 177 inner iterations were required. With 100 preliminary CG iterations, an average of 162 iterations were required. The shift from 20 to 100 preliminary iterations is associated with a five fold increase in the memory requirements for preconditioning, but the cost per iteration remains nearly constant and the number of iterations

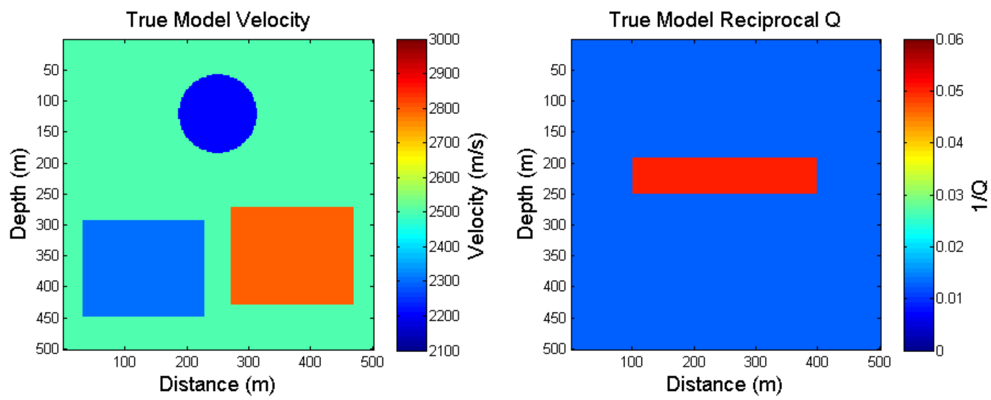

FIG. 8: Velocity model for the 2.5m resolution anacoustic example (left), and reciprocal Q model (right).

decreases significantly.

BFGS inner solver

An alternative method of improving the convergence rate in the truncated Newtons method is to change to a more efficient inner solver. When a BFGS inner solver was employed, and a forcing term of 10^{-4} was used, an average of 98 inner iterations were required per outer iteration. These iterations require the same number of wave propagation problems as conjugate gradient iterations, which make up the majority of the computational cost. The additional computational cost associated with generating the approximate inverse Hessian-vector product was negligible. Consequently, this method demonstrated a considerable increase in convergence rate, and offers a significantly more efficient approach than the CG or preconditioned CG approach.

Scaleability

As the rationale for employing the truncated Newton algorithm in FWI is based on the unmanageable costs associated with the exact Newton method for very large models, it is important to verify that the results discussed above extend to larger models. In order to investigate the effects of a larger model, a scaled up version of the model in figure 1 was used. This model used an x and z spacing of 2.5m instead of the 10m spacing used in figure 1. The resulting model is shown in figure 8. As the purpose of this example is to demonstrate the scaleability of the method, the sources, which occupied one 10m by 10m gridpoint each in the low resolution model were still modeled as being 10m by 10m in the high resolution model. The anacoustic TN FWI formulation as described above was then applied to this system. An inner BFGS solver was used, with a forcing term of 10^{-4}. This required an average of 111 inner iterations per outer iteration, a small increase in iterations required over the smaller model. The results of this example are shown in figure 8.

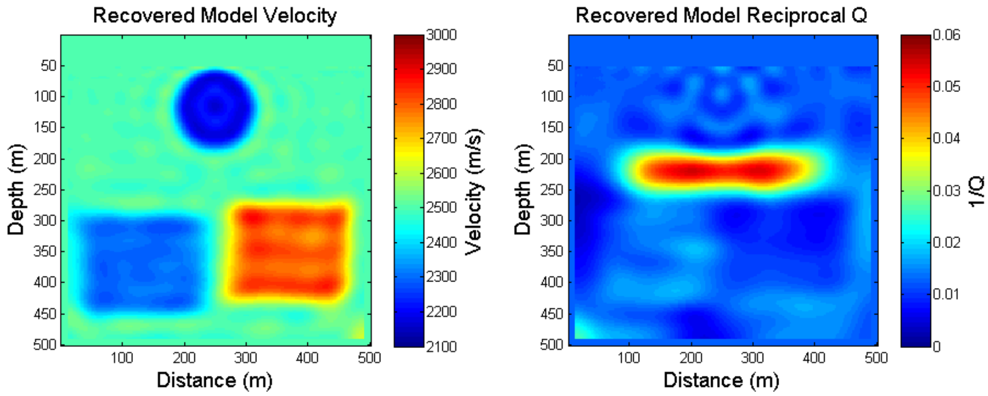

FIG. 9: Result of Truncated Newton FWI with forcing term 10^{-4} for recovered velocity (left) and reciprocal Q (right) using 2.5m resolution model. Considerable similarities exist with the equivalent 10m resolution case (figure 6).

DISCUSSION

While the convergence rate of the TN method can be significantly improved, it is evident that avoiding cross-talk in the anacoustic problem requires a significant increase in computation over similar acoustic problems. This cross-talk is dominant in the recovered attenuation model, at least in the example studied. This is true even though the Q value at the anomaly is quite low (20), and the velocity contrasts are relatively small (less than 12%). Attenuation effects are unlikely to have as large an effect on measured data relative to velocity effects as they do here in most real subsurface environments. It seems then to be a fair question to ask whether the extra cost is worth the associated mitigation of cross-talk in this example. The answer would seem to rely on the goals of the inversion. If the goal of including attenuative and dispersive effects is merely to improve the recovered velocity model, it can be easily argued that the level of precision used in the acoustic TN case is adequate, and the cross-talk effects on velocity are relatively minor considering the cost associated with removing them. If instead the goal of including anelastic effects is to recover a usable Q model, these extra costs are likely a necessity to be able to perceive the distinction between true Q effects and velocity associated cross-talk.

CONCLUSIONS

Multiparameter, anacoustic FWI can be performed using the truncated Newton method. Accurately recovering attenuation information requires a far better approximation of the Newton update than is necessary in the single parameter acoustic case. This is associated with a significant additional cost of computation. This computational cost can be reduced by using a preconditioned conjugate gradient or BFGS inner solver instead of the non-preconditioned conjugate gradient. While outer and inner BFGS preconditioning have been reported as being effective for other truncated Newton problems, they were not found to be effective in this FWI approach. A two step inner BFGS preconditioning was found to increase convergence rate, but the greatest improvement was obtained using the BFGS inner solver. The BFGS inner solver incurs negligible additional cost per iteration, but has an increased memory requirement. An L-BFGS inner solver would help to relax this requirement. Even with convergence rate improvements, the cost of accurately recovering a

Q model remains very high. Lower accuracy truncated Newton methods may offer a means of partially correcting for Q effects in recovering the velocity model, but are not sufficient to recover a Q model.

ACKNOWLEDGMENTS

The authors thank the sponsors of CREWES for continued support. This work was funded by CREWES industrial sponsors and NSERC (Natural Science and Engineering Research Council of Canada) through the grant CRDPJ 461179-13. Scott Keating was also supported by the Queen Elizabeth II scholarship.

REFERENCES

Innanen, K., 2014, Seismic avo and the inverse hessian in precritical reflection full waveform inversion: Geophys. J. Int., **199**.

Keating, S., and Innanen, K., 2016, Challenges in anacoustic full waveform inversion: CREWES Annual Report, **28**.

Metivier, L., Brossier, R., Virieux, J., and Operto, S., 2013, Full waveform inversion and the truncated newton method: Siam J. Sci. Comput., **35**.

Morales, J., and Nocedal, J., 2000, Automatic preconditioning by limited memory quasi-newton updating: Siam J. Optim., **10**.

Operto, S., Gholami, Y., Prieux, V., Ribodetti, A., Brossier, R., Metivier, L., and Virieux, J., 2013, A guided tour of multiparameter full-waveform inversion with multicomponent data: From theory to practice: The Leading Edge, 1040–1054.

APPENDIX

The adjoint state method is important to ensuring computational savings in TN FWI. In the following, a summary of the truncated Newton method based on Metivier et al. (2013) is provided. Given a optimization problem where we seek to minimize

$$\phi\big(u(p)\big) \text{ subject to } F\big(p, u(p)\big) = 0 \quad , \tag{6}$$

the Lagrangian function is given by

$$\mathcal{L}(p, u(p), \lambda) = \phi\big(u(p)\big) + \Big(F\big(p, u(p)\big), \lambda\Big) \quad . \tag{7}$$

Here p is a model parameter, and (,) denotes the inner product. The adjoint state method is helpful when we are interested in finding the derivative of ϕ with respect to p while still maintaining the constraint, and where directly calculating $\frac{\partial u}{\partial p}$ is prohibitively expensive. If for example, u is the wavefield in a seismic problem, and p are model parameters, computing $\frac{\partial u}{\partial p}$ is often impractical. We then define

$$\bar{u}(p) \text{ such that } F\big(p, \bar{u}(p)\big) = 0 \quad . \tag{8}$$

Then,

$$\mathcal{L}(p, \bar{u}(p), \lambda) = \phi\big(\bar{u}(p)\big) = \phi(p) \quad . \tag{9}$$

The derivative is then given by

$$\frac{d}{dp}\phi(p) = \frac{d}{dp}\left[\phi\big(\bar{u}(p)\big) + \Big(F\big(p, \bar{u}(p)\big), \lambda\Big)\right] \quad . \tag{10}$$

This reduces to

$$\frac{d}{dp}\phi(p) = \Big(\frac{\partial}{\partial p}F\big(p, \bar{u}(p)\big), \lambda\Big) + \frac{\partial}{\partial u}\mathcal{L}(p, \bar{u}(p), \lambda)\frac{\partial}{\partial p}\bar{u}(p) \quad . \tag{11}$$

If we then define $la\bar{m}bda$ such that

$$\frac{\partial}{\partial u}\mathcal{L}(p, \bar{u}(p), \bar{\lambda}) = 0 \quad , \tag{12}$$

the second term of 11 vanishes, and we are left with

$$\frac{d}{dp}\phi(p) = \Big(\frac{\partial}{\partial p}F\big(p, \bar{u}(p)\big), \lambda\Big) \quad . \tag{13}$$

Consequently, calculating $\frac{d}{dp}\phi(p)$ requires only that we solve for $\bar{u}(p)$ and $\bar{\lambda}$ from equations 8 and refA2, then evaluate equation 13. The specific case of a GN Hessian-vector product is outlined in Metivier et al. (2013), but involves an additional constraint, and thus requires solution for another Lagrange multiplier. This brings the total cost to three wavefield propagation problems, one of which is already solved in finding the gradient. This total cost of two additional forward problems to find the Hessian-vector product is what allows the computational savings of the truncated Newton method in FWI.