Well-log derived step lengths in Full Waveform Inversion

Tunde Arenrin

Gary Margrave

Motivation

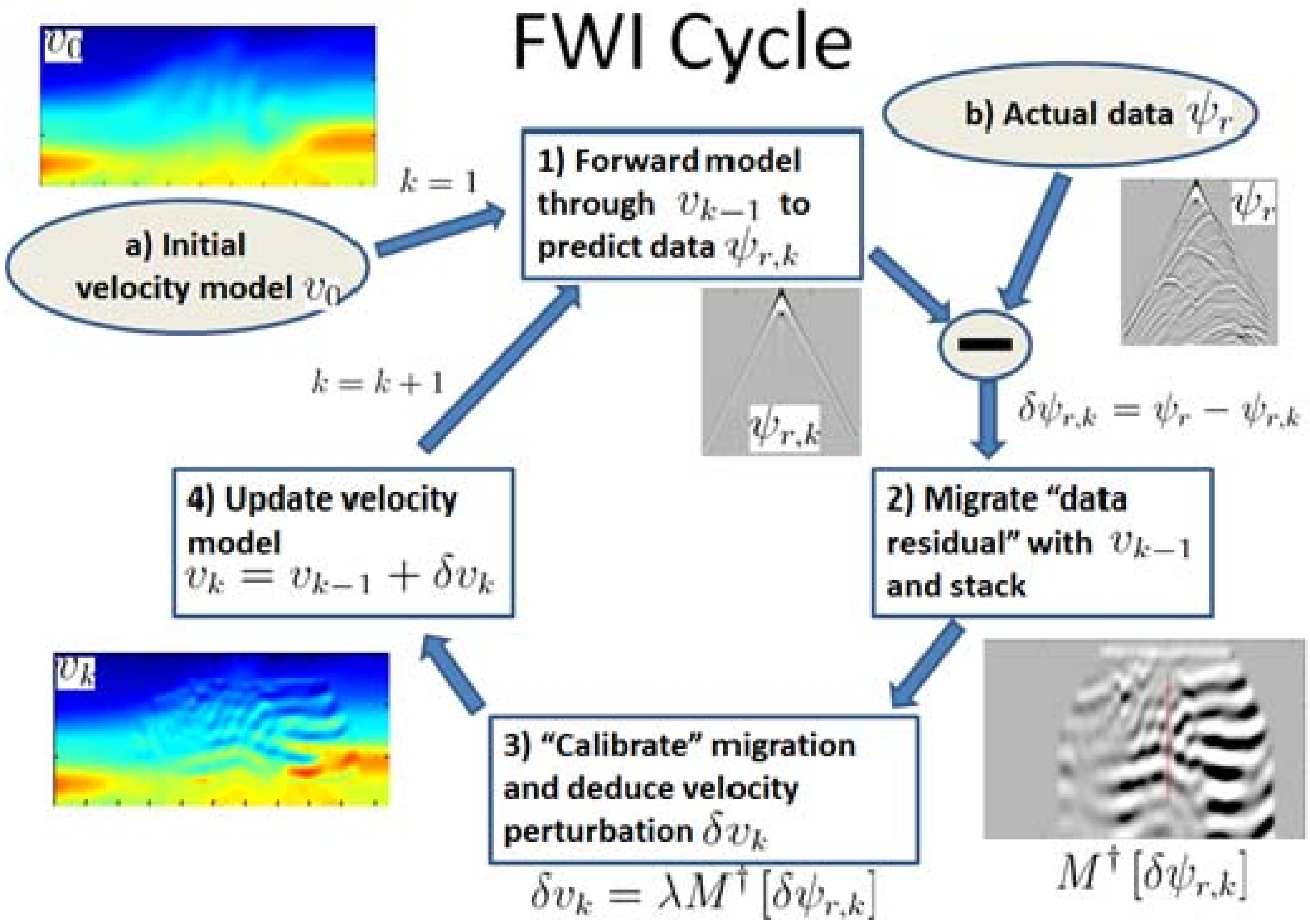
»Being able to make FWI a part of the seismic processing work flow under current computational capability.

»Incorporating as much data as can be gathered would always be advantageous as is the case of joint inversion, or any field development programme. So why not do same with FWI if we have other sources of data!

»Information from well logs can help with improving the wavelet estimate which is essential for proper updates.

» Due to the computational cost of Newton or Newton-like methods.

Full Waveform Inversion is an optimization technique that seeks to find a model of the subsurface that best matches the observed field data at every receiver location. The method begins from a best guess of the true model, which is iteratively improved using linearized inversions methods.



Margrave et al, 2012

Minimizing objective functions

1) $$\phi_k = \sum_{s,r} (\psi - \psi_k)^2$$

Conventional FWI, **validates** the predicted data ψ_k against the observed data ψ

2) $$\beta_k = \left\| \lambda G_k - (V_{well} - V_{BG})_k \right\|^2$$

Validates the migration velocity model V_{BG} against the known velocity model V_{well} at the well. Using well control (IMMI) as an alternative to line search methods.

$$\delta v_k(x,z) = \lambda \int \sum_{s,r} \omega^2 \hat{\psi}_s(x,z,\omega) \delta\hat{\psi}^*_{r(s),k}(x,z,\omega) d\omega$$

$$\lambda = \frac{\sum_j \delta V_j G_j}{\sum_j G_j^{\,2}}$$ $\delta V_j = (V_{well} - V_{BG})_j$ j indicates sample number.

Optimization Schemes

- Steepest Descent $\lambda^k \, \mathrm{g}^k$
- Conjugate Gradient $p_k = \mathrm{g}^k + \beta_k \, p_{k-1}$
- Newton/Newton based Methods $H^{-1k} \, \mathrm{g}^k$

In our previous work (Arenrin et al, 2014 and Arenrin et al, 2015), we applied well-derived step-length to the steepest descent method. In this paper however, we apply well-derived step-length to CG method.

Conjugate Gradient (CG) Method

$$\alpha_{k-1} = \frac{\left|\nabla \phi_{k-1}\right|^2}{\left(p_{k-1}, Ap_{k-1}\right)}$$

$$v_k = v_{k-1} + \alpha_{k-1} p_{k-1}$$

$$\beta_k = \frac{\left|\nabla \phi_k\right|^2 - \left(\nabla \phi_k, \nabla \phi_{k-1}\right)}{\left|\nabla \phi_{k-1}\right|^2}$$

$$p_k = \nabla \phi_k + \beta_k p_{k-1}$$

Hestenes and Stiefel, CG algorithm (1952)

$$\alpha_{k-1} = \frac{\left|\nabla \phi_{k-1}\right|^2}{\left(p_{k-1}, Ap_{k-1}\right)}$$

$$v_k = v_{k-1} + \alpha_{k-1} p_{k-1}$$

$$\beta_k = \frac{\nabla \phi_k^{\;T} \left(\nabla \phi_k - \nabla \phi_{k-1}\right)}{\left|\nabla \phi_{k-1}\right|^2}$$

$$p_k = \nabla \phi_k + \beta_k p_{k-1}$$

Polak and Ribiere, CG algorithm (1969)

Computation of the step length

Our misfit function is

$$S(\mathbf{m}_{n+1}) = S(\mathbf{m}_n + \alpha_n \boldsymbol{\phi}_n) \tag{A-3}$$

$$= [\mathbf{d}_0 - \mathbf{f}(\mathbf{m}_n + \alpha_n \boldsymbol{\phi}_n)]^t [\mathbf{d}_0 - \mathbf{f}(\mathbf{m}_n + \alpha_n \boldsymbol{\phi}_n)], \tag{A-4}$$

where $\mathbf{d}_0$ are the observed seismic records, and $\mathbf{f}(\mathbf{m})$ is the computed field (synthetic traces) in the model $\mathbf{m}$, and we have

$$\underset{\sim}{\mathbf{F}}\delta\mathbf{m} = \lim_{\varepsilon \to 0} \frac{\mathbf{g}(\mathbf{m} + \varepsilon\delta\mathbf{m}) - \mathbf{g}(\mathbf{m})}{\varepsilon}, \tag{A-5}$$

where $\underset{\sim}{\mathbf{F}}$ is a linear operator that takes the derivative of $\mathbf{f}$ at point $\mathbf{m}$.

The expression $S(\mathbf{m}_{n+1})$ can thus be written

$$S(\mathbf{m}_{n+1}) = [\mathbf{d}_0 - \mathbf{f}(\mathbf{m}_n) + \alpha_n \underset{\sim}{\mathbf{F}}_n \boldsymbol{\phi}_n)]^t \times [\mathbf{d}_0 - \mathbf{f}(\mathbf{m}_n) + \alpha_n \underset{\sim}{\mathbf{F}}_n \boldsymbol{\phi}_n)]. \tag{A-6}$$

The optimal α_n is that which minimizes $S(\mathbf{m}_{n+1})$ and satisfies

$$\frac{\partial S(\mathbf{m}_{n+1})}{\partial \alpha_n} = 0. \tag{A-7}$$

Then, obtaining the coefficient

$$\alpha_n = \frac{[\underset{\sim}{\mathbf{F}}_n \boldsymbol{\phi}_n]^t [\mathbf{d}_0 - \mathbf{f}(\mathbf{m}_n)]}{[\underset{\sim}{\mathbf{F}}_n \boldsymbol{\phi}_n]^t [\underset{\sim}{\mathbf{F}}_n \boldsymbol{\phi}_n]} \tag{A-8}$$

requires the computation of a new forward problem in a medium $\mathbf{m} = \mathbf{m}_n + \varepsilon \boldsymbol{\phi}_n$, with ε sufficiently small. Practically, we choose an ε such that

$$\max(\varepsilon \boldsymbol{\phi}_n) \leq \frac{\max(\mathbf{m}_n)}{100}. \tag{A-9}$$

Note that the operator $\mathbf{F}$ corresponds to a Born approximation, but that in our nonlinear algorithms we approximate it by a finite-difference relation.

A. Pica, J. P. Diet, and A. Tarantola (Geophysics, vol. 55, March, 1990)

Conjugate Gradient (CG) Method

$$\lambda = \frac{\sum_j \delta V_j G_j}{\sum_j G_j^{\,2}}$$

$$v_k = v_{k-1} + \lambda_{k-1} p_{k-1}$$

$$\beta_k = \frac{\left|\nabla \phi_k\right|^2 - (\nabla \phi_k, \nabla \phi_{k-1})}{\left|\nabla \phi_{k-1}\right|^2}$$

$$p_k = (1+\beta_k)^{-1} (\nabla \phi_k + \beta_k p_{k-1})$$

Arenrin and Margrave, Algorithm (2015)

$$\alpha_{k-1} = \frac{\left|\nabla \phi_{k-1}\right|^2}{(p_{k-1}, Ap_{k-1})}$$

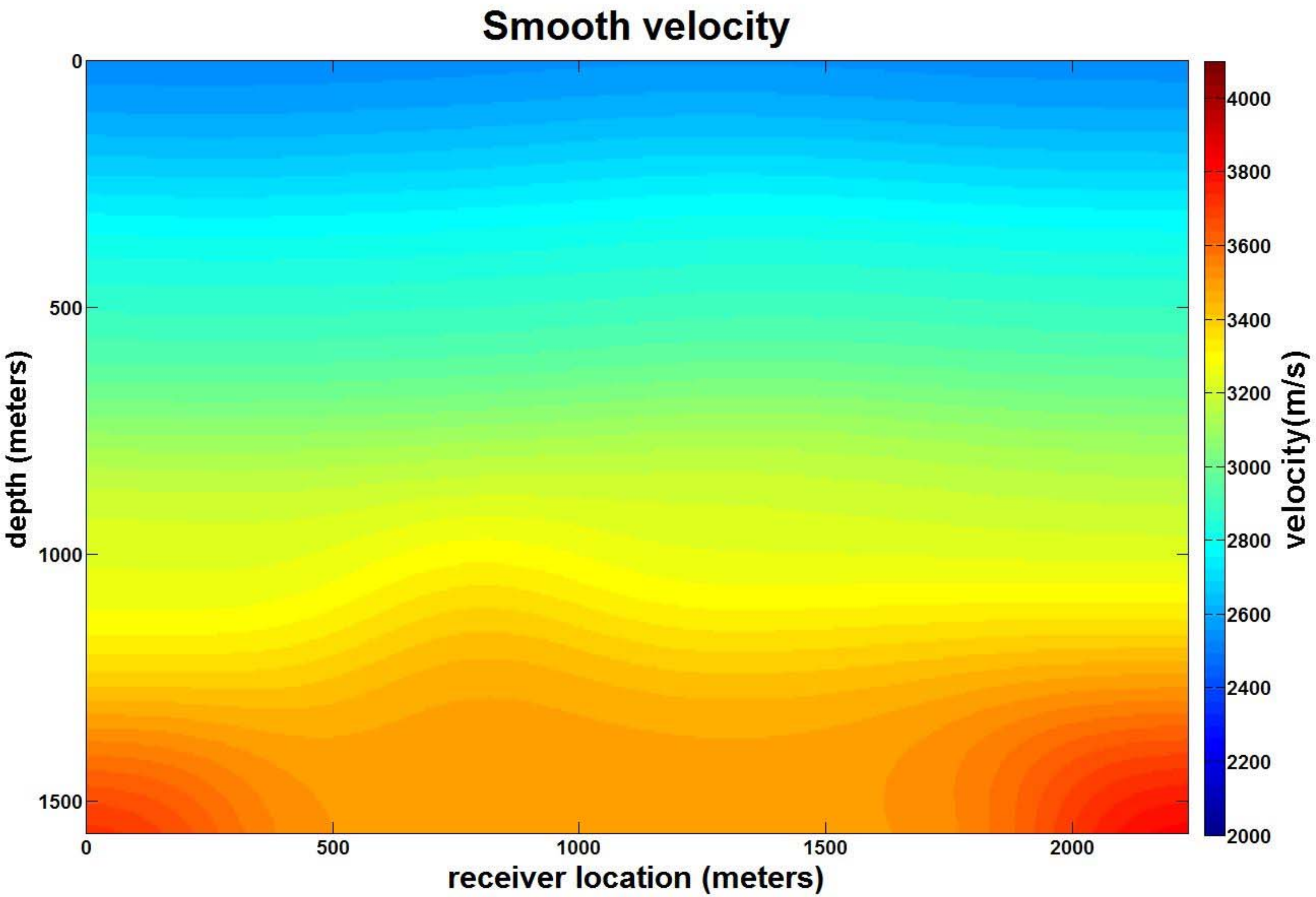
$$v_k = v_{k-1} + \alpha_{k-1} p_{k-1}$$

$$\beta_k = \frac{\left|\nabla \phi_k\right|^2 - (\nabla \phi_k, \nabla \phi_{k-1})}{\left|\nabla \phi_{k-1}\right|^2}$$

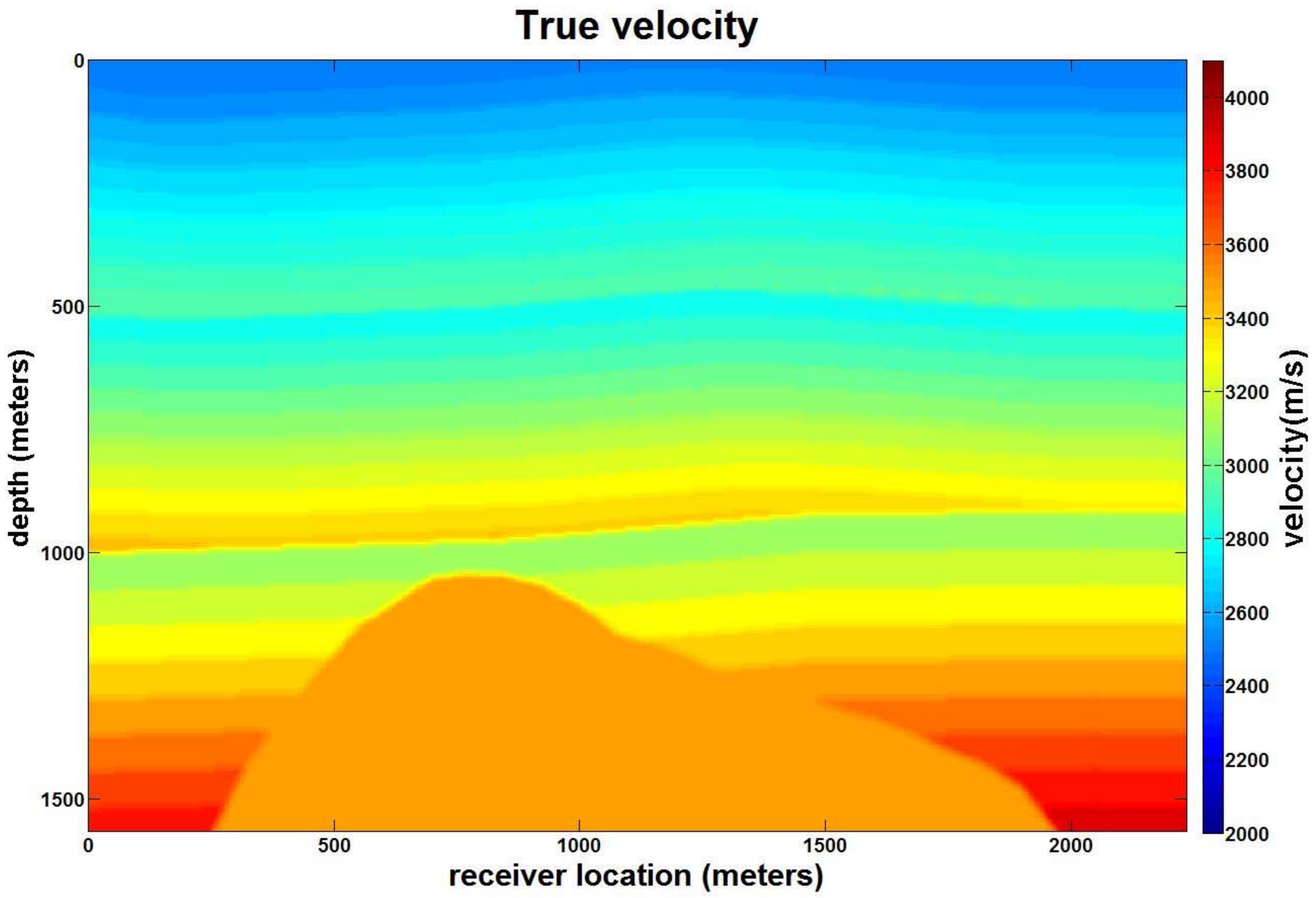
$$p_k = (1+\beta_k)^{-1} (\nabla \phi_k + \beta_k p_{k-1})$$

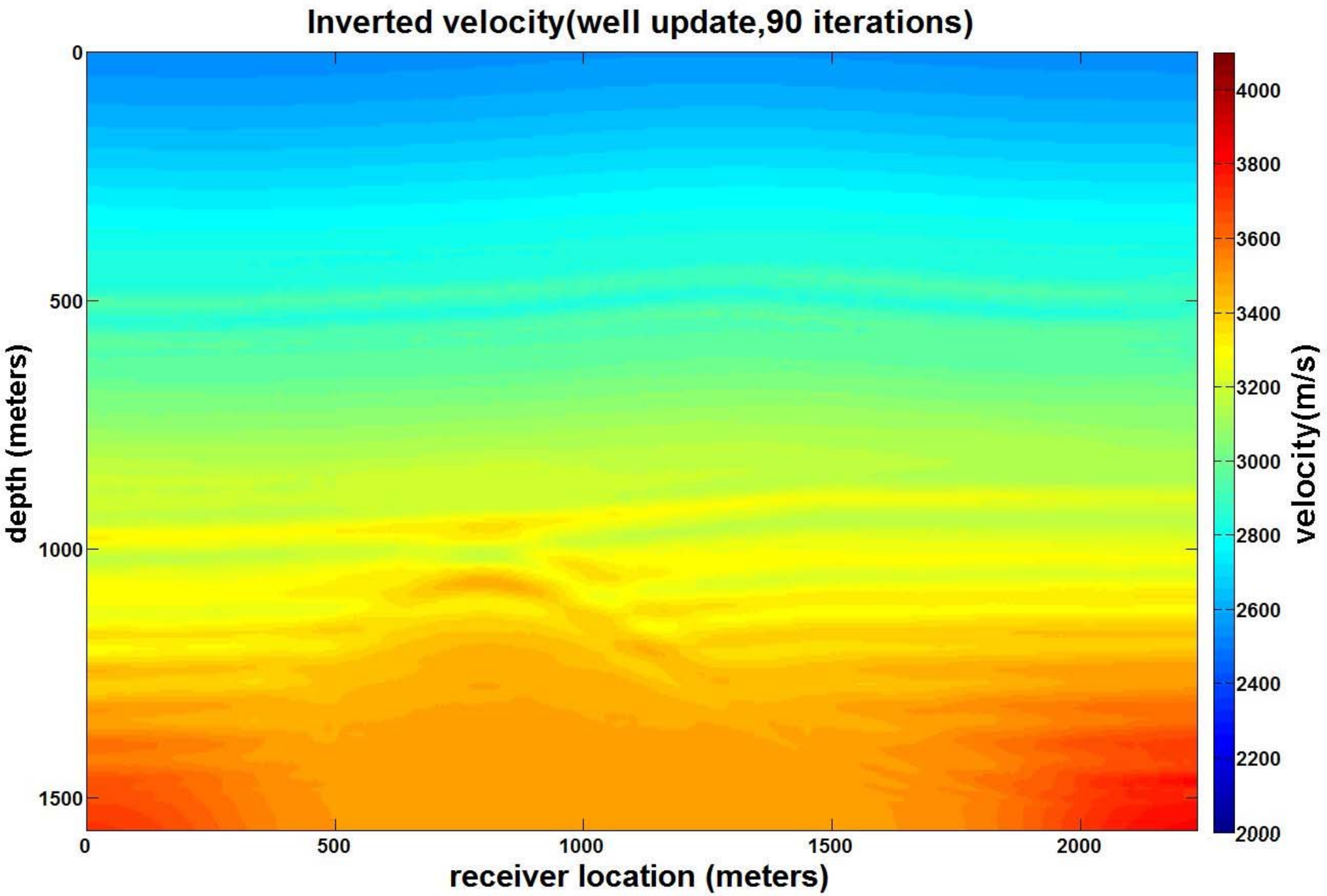
Hestenes, CG algorithm (1990)

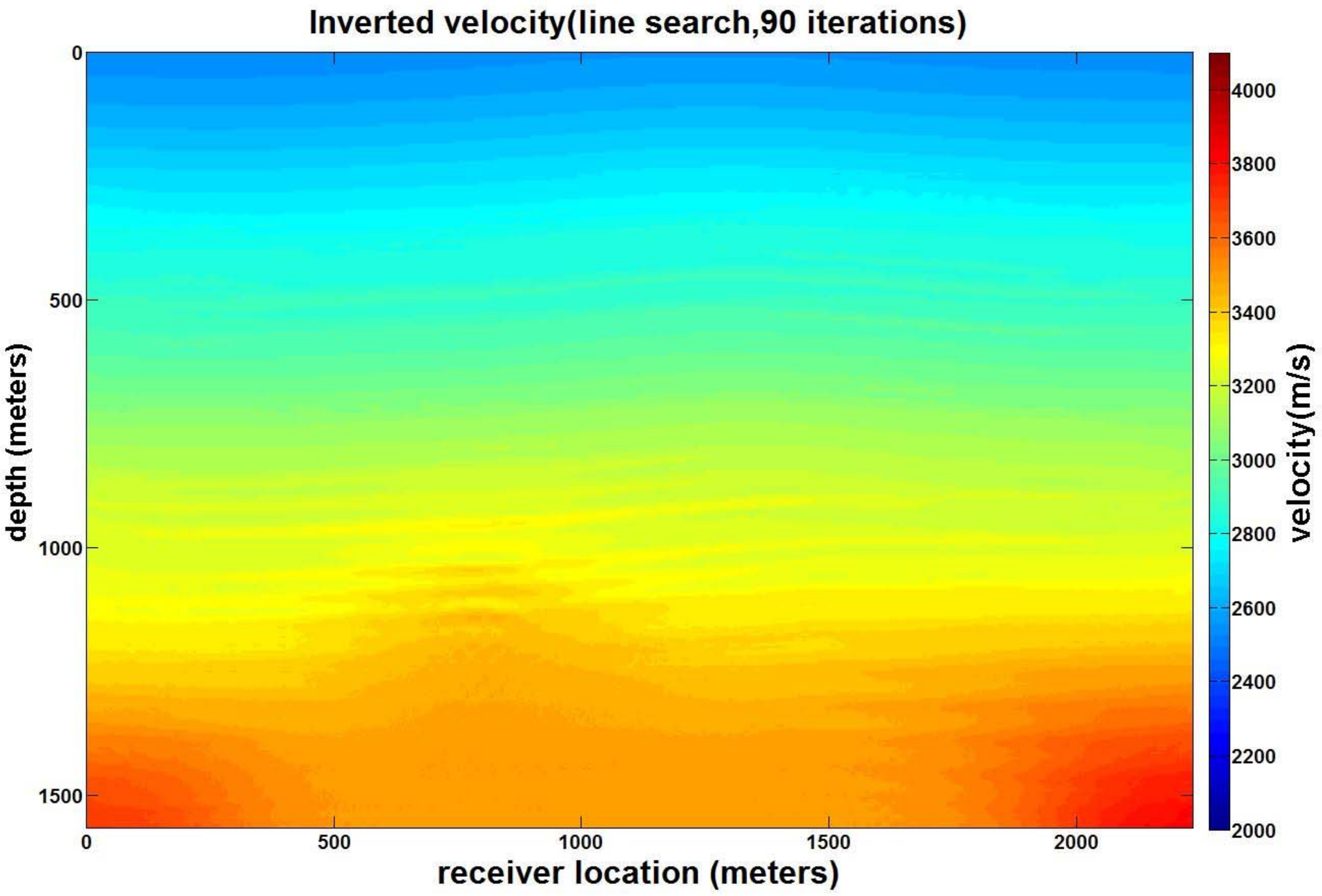
Quickly let's revisit our results of last year (2014)

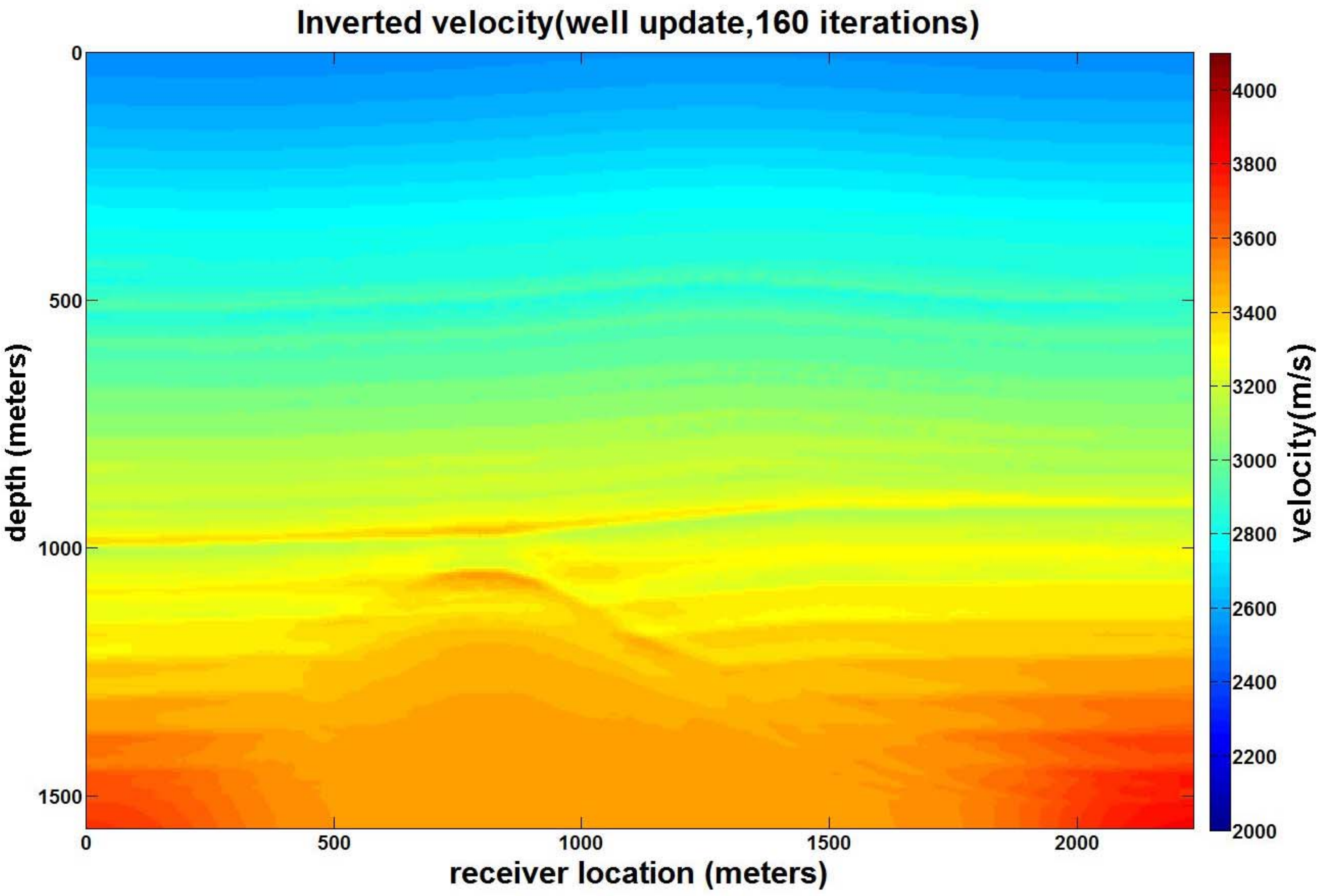

Smooth velocity
depth (meters)
0
500
1000
1500
receiver location (meters)
0
500
1000
1500
2000
velocity(m/s)
4000
3800
3600
3400
3200
3000
2800
2600
2400
2200
2000

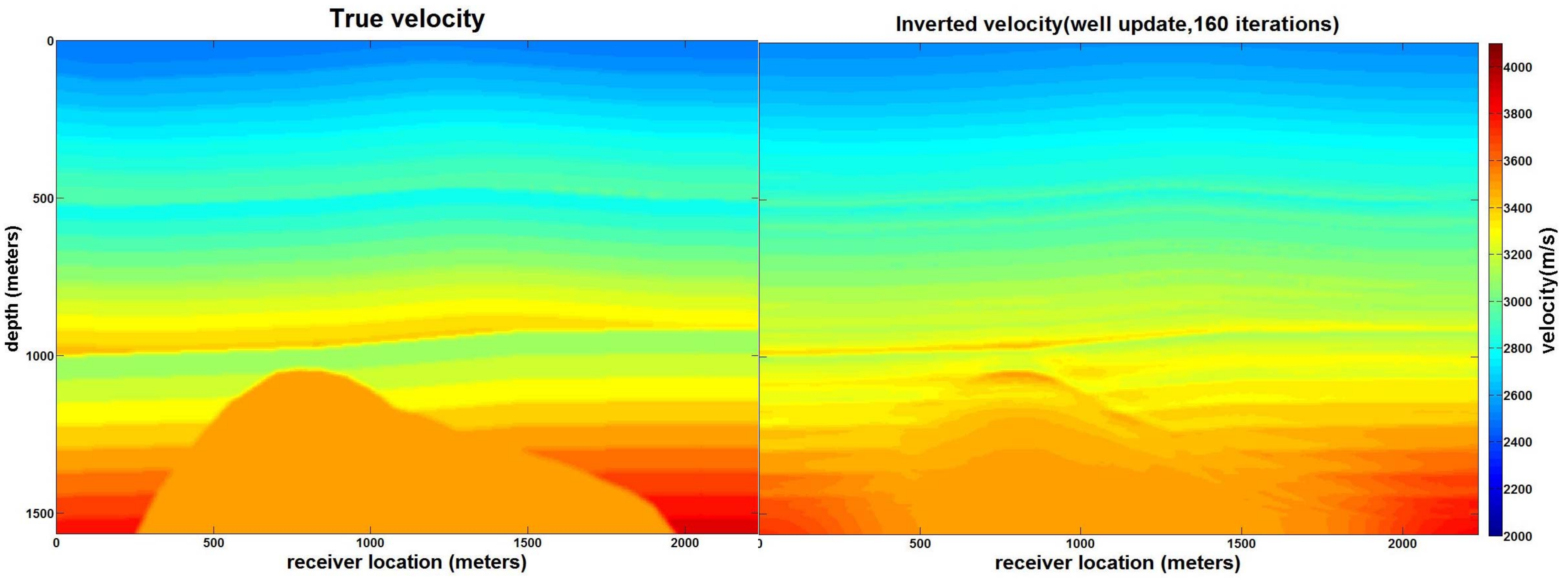
Last year's results

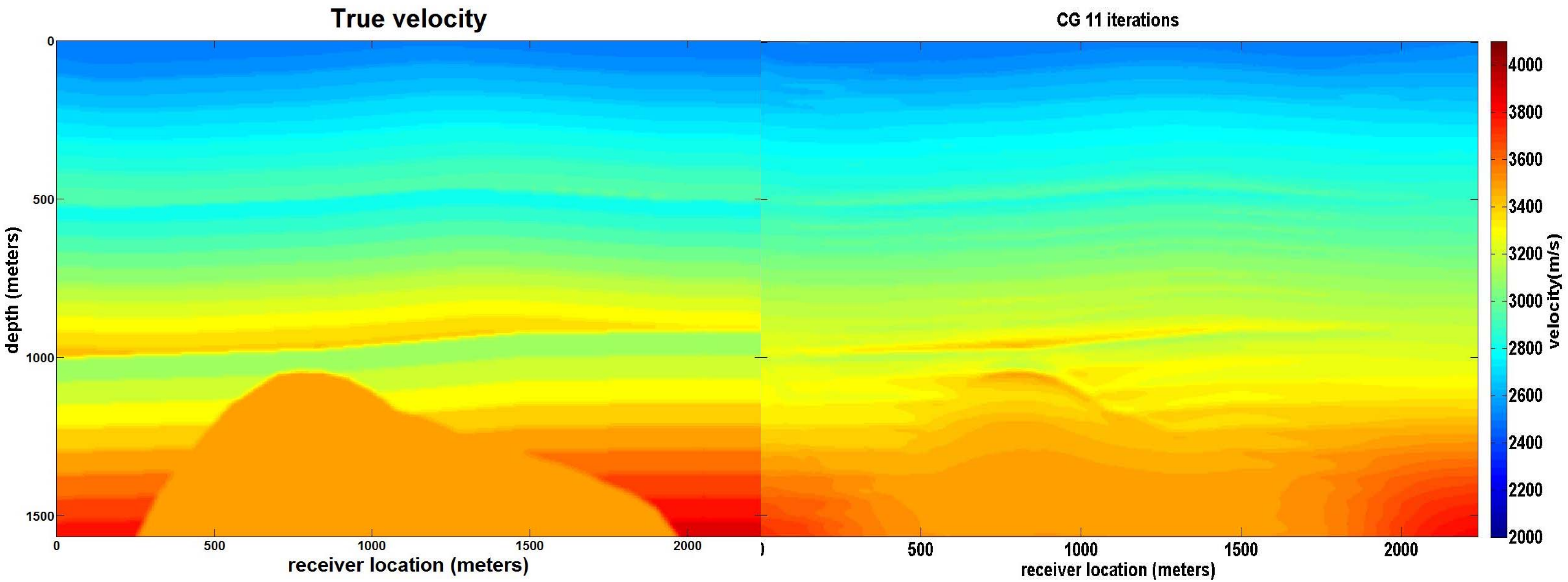


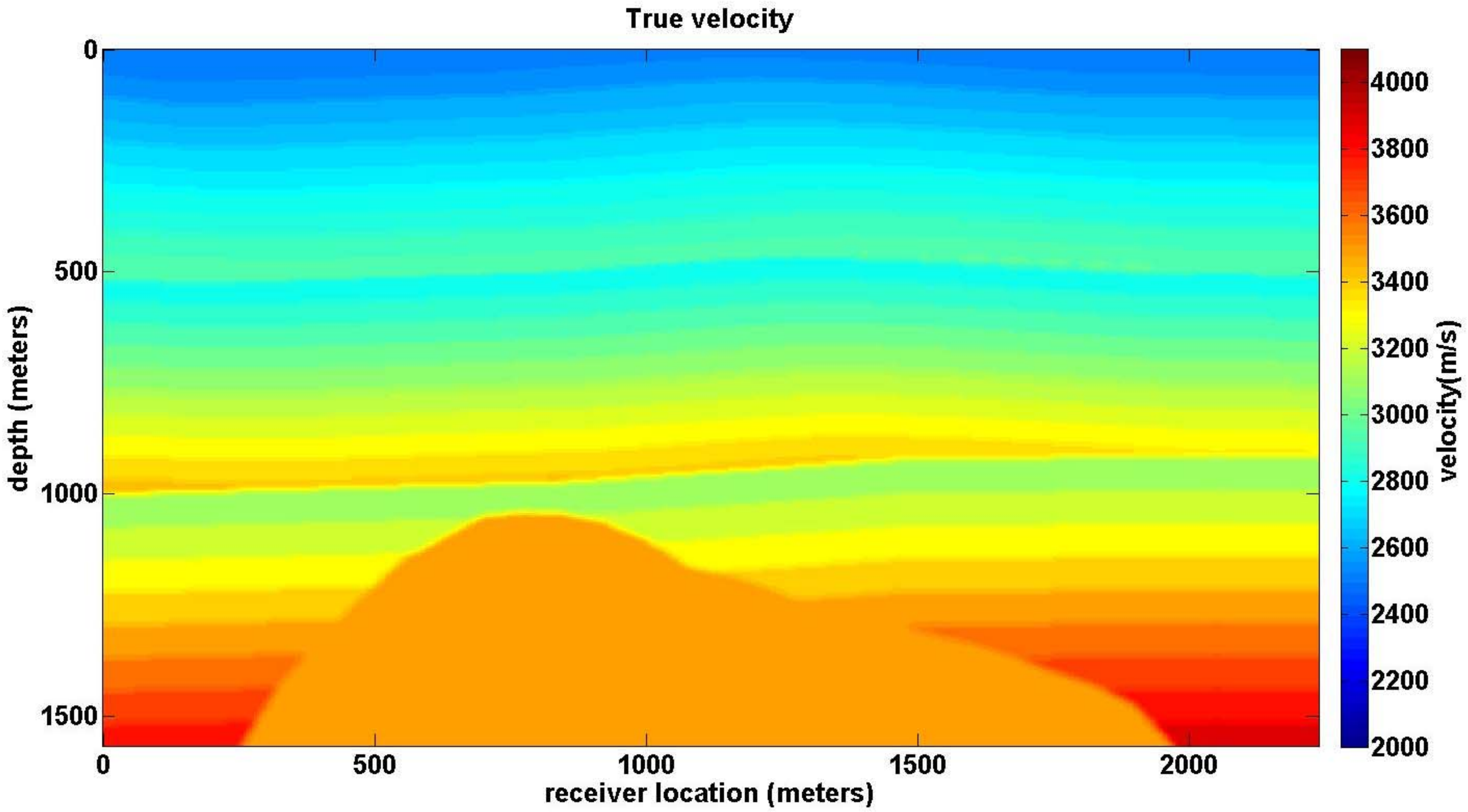

Inverted velocity(well update,90 iterations)
depth (meters)
0
500
1000
1500
receiver location (meters)
0
500
1000
1500
2000
velocity(m/s)
4000
3800
3600
3400
3200
3000
2800
2600
2400
2200
2000

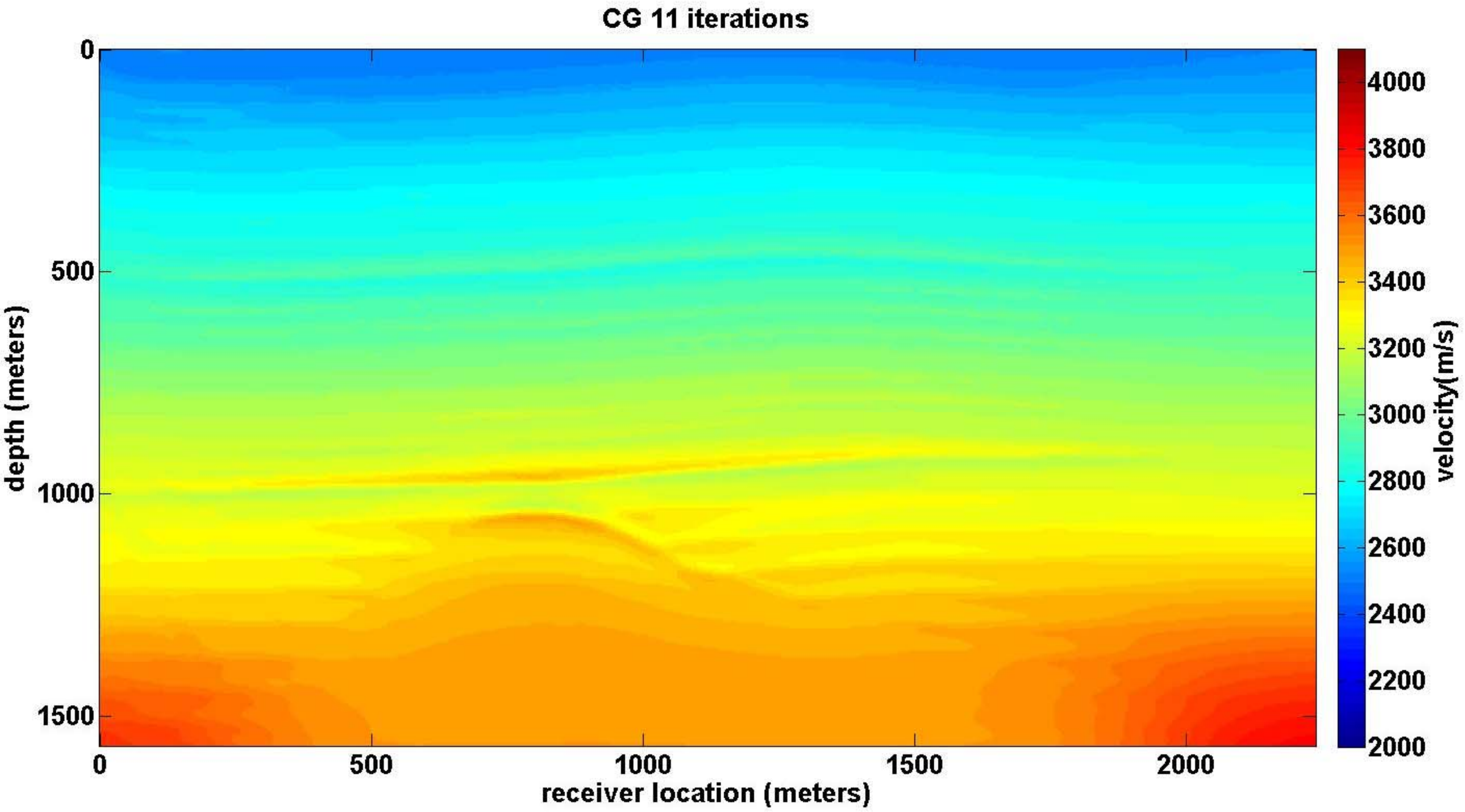

Inverted velocity(line search,90 iterations)
depth (meters)
0
500
1000
1500
0
500
1000
1500
2000
receiver location (meters)
velocity(m/s)
4000
3800
3600
3400
3200
3000
2800
2600
2400
2200
2000

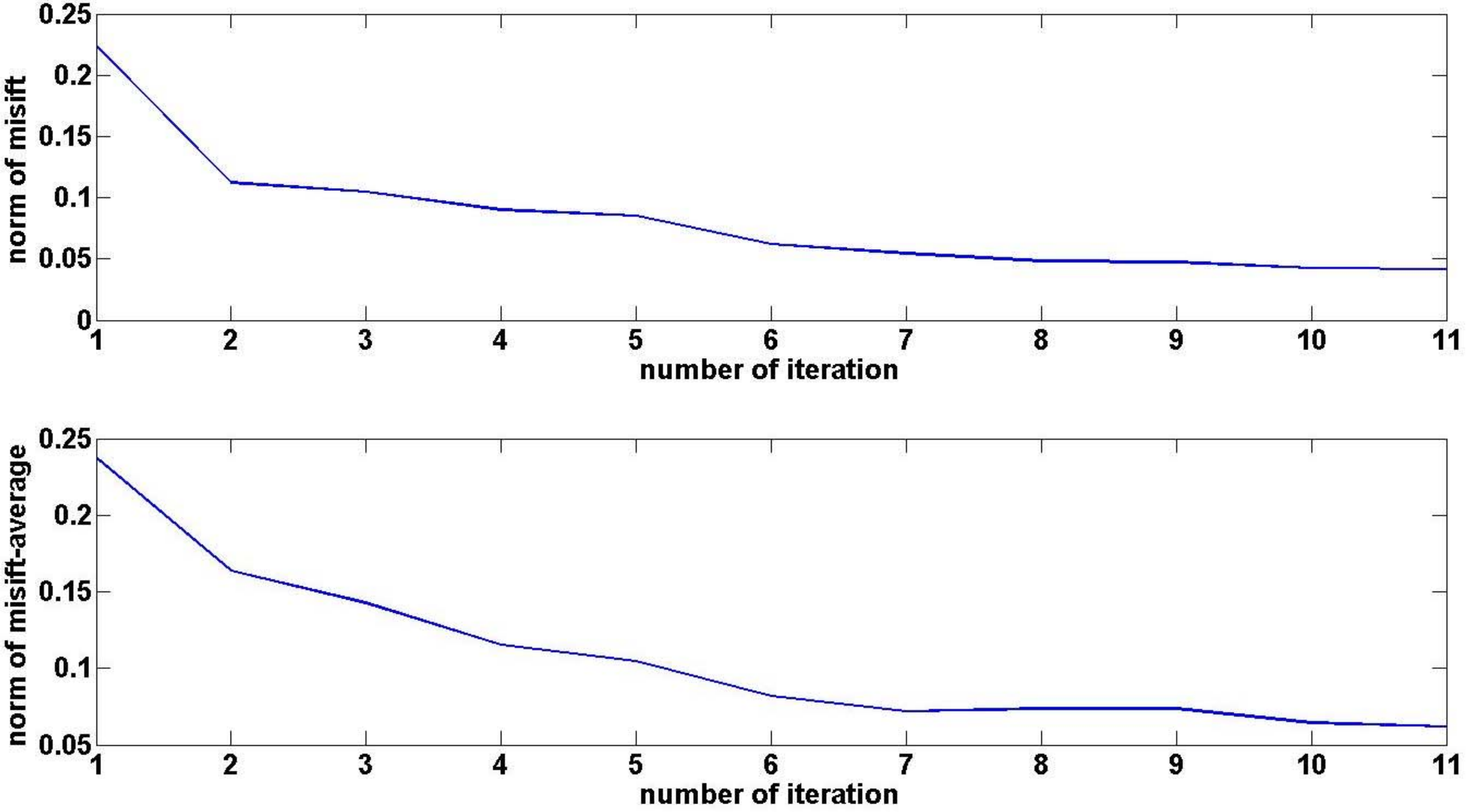

Inverted velocity(well update,160 iterations)
depth (meters)
0
500
1000
1500
0
500
1000
1500
2000
receiver location (meters)
velocity(m/s)
4000
3800
3600
3400
3200
3000
2800
2600
2400
2200
2000

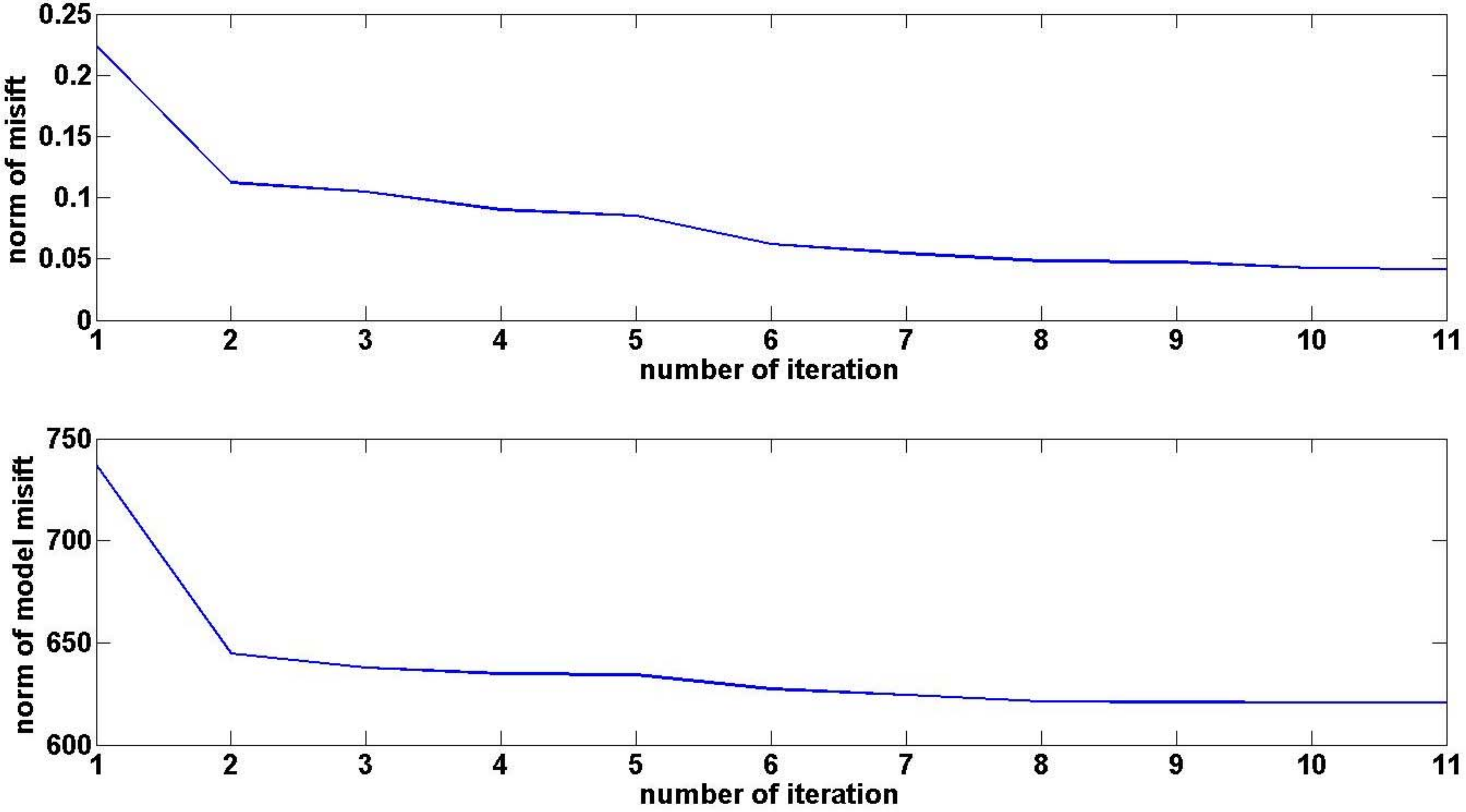

True velocity
Inverted velocity(well update,160 iterations)
depth (meters)
0
500
1000
1500
receiver location (meters)
0
500
1000
1500
2000
velocity(m/s)
4000
3800
3600
3400
3200
3000
2800
2600
2400
2200
2000


True velocity
CG 11 iterations
depth (meters)
0
500
1000
1500
receiver location (meters)
2000
4000
3800
3600
3400
3200
3000
2800
2600
2400
2200
velocity(m/s)


True velocity
depth (meters)
0
500
1000
1500
receiver location (meters)
0
500
1000
1500
2000
velocity(m/s)
4000
3800
3600
3400
3200
3000
2800
2600
2400
2200
2000


CG 11 iterations
depth (meters)
0
500
1000
1500
receiver location (meters)
0
500
1000
1500
2000
velocity(m/s)
4000
3800
3600
3400
3200
3000
2800
2600
2400
2200
2000


norm of misfit
0.25
0.2
0.15
0.1
0.05
0
1 2 3 4 5 6 7 8 9 10 11
number of iteration
norm of misfit-average
0.25
0.2
0.15
0.1
0.05
1 2 3 4 5 6 7 8 9 10 11
number of iteration



0.25
0.2
0.15
0.1
0.05
0
norm of misift
1 2 3 4 5 6 7 8 9 10 11
number of iteration
750
700
650
600
norm of model misift
1 2 3 4 5 6 7 8 9 10 11
number of iteration

Conclusions

»We have been able to reduce the number of iterations from 90 iterations to 11 iterations using the algorithm described.

»The data misfit norm and the model misfit norm both decreased as we ran more iterations. This could serve as a good QC tool when incorporating well information into FWI.

»Information from well logs to calculate a scalar for the model update shows encouraging results and could save us some computational time compared with the line search method.

» Deriving step-lengths from well logs seem to work better with the CG directions to steepest descent directions.

Example: Hussar synthetic

- **Modelling Parameters**

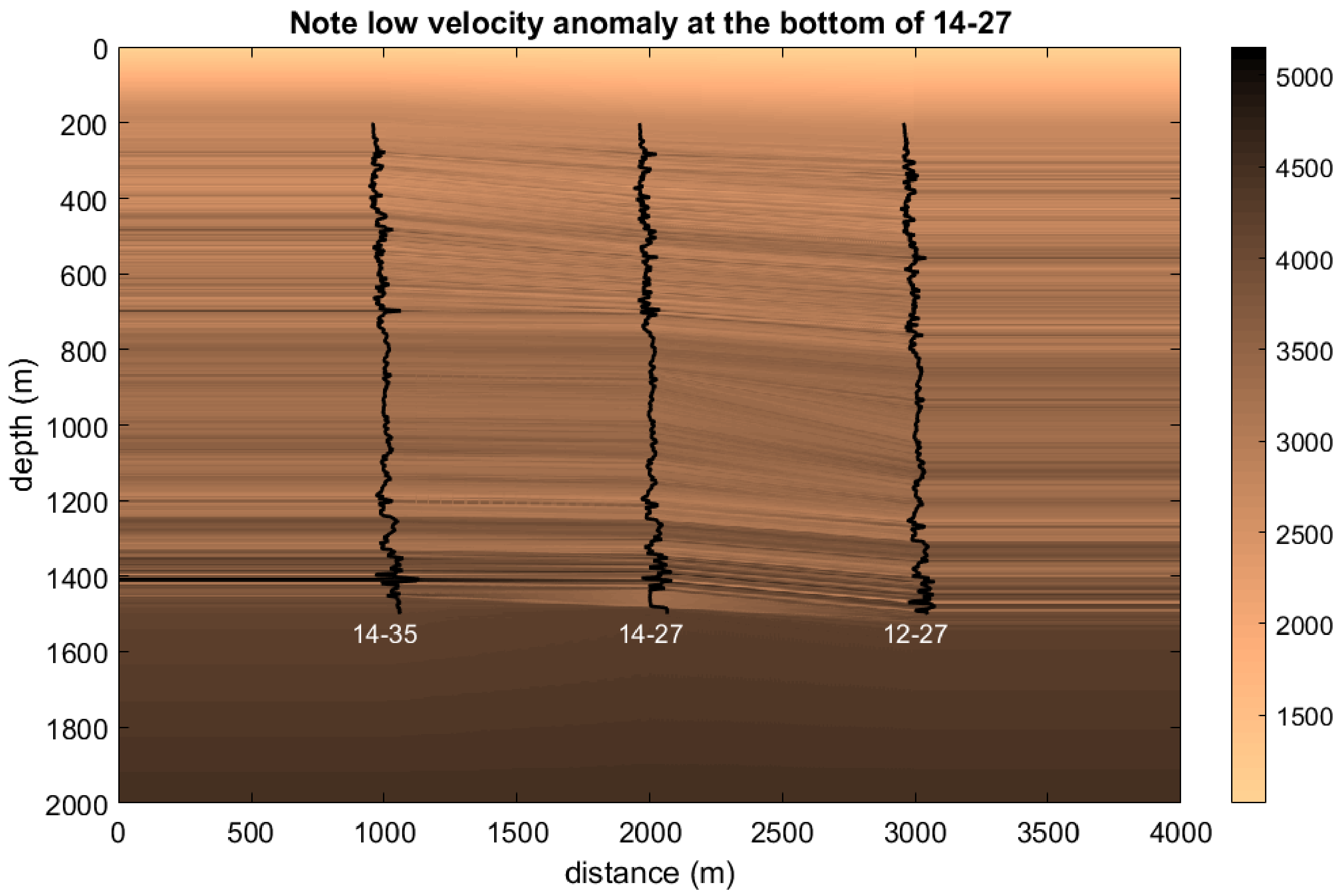
Shot spacing	67 meters
Receiver spacing	2.5 meters
Dominant frequency	50 hertz (min phase)
Number of Shots	61
Number of Receivers	1600
Sampling Interval	2ms
Discrete parameters	1.28x10^6

Marghuin Velocity Model (from Hussar sonic logs)

The three wells (14-35, 14-27, and 12-27) in the area had sonic logs

-The velocity model was made by spatially interpolating the sonic logs from the three wells at Hussar. The interpolation was guided using the formation tops.

-The three well logs all start near 200m depth and extend to about 1550m depth.



Marghuin Velocity Model

Inversion

- Reverse Time Migration algorithm with a Cross Correlation IC

-The Cross Correlation IC provides an image amplitude that is the product of source and receiver wavefields and has the unit of amplitude square.

Output from the RTM algorithm is 'source-normalized', thus it has the same unit as the reflectivity

- Frequency strategy: Expanding bandwidth iteration. (Margrave, Crewes Research Report, 2015)

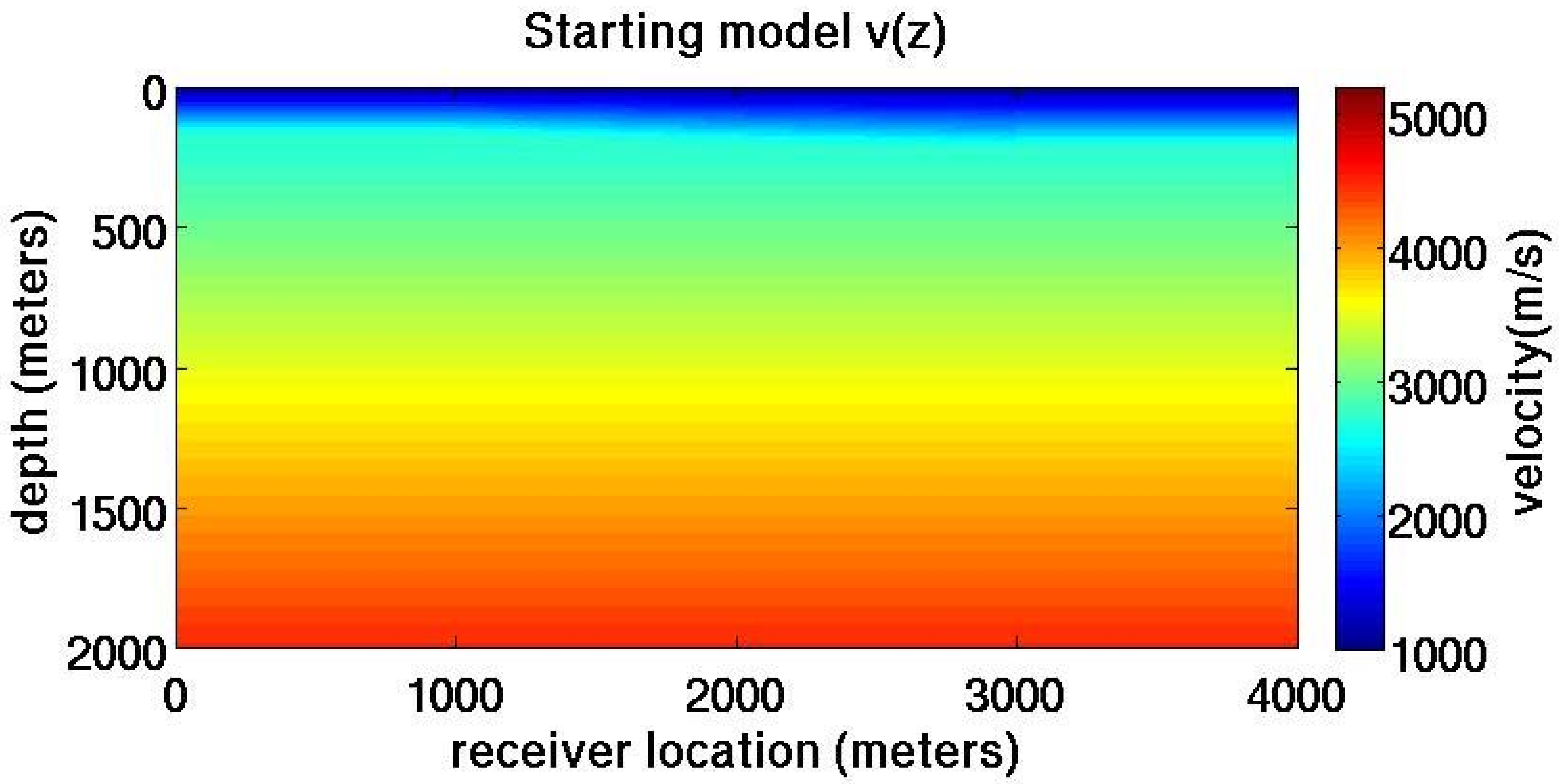
Iteration 1	[4Hz, 6Hz, 8Hz, 15Hz]
Iteration 2	[4Hz, 6Hz, 13Hz, 20Hz]
Iteration 3	[4Hz, 6Hz, 18Hz, 25Hz]
Iteration 4	[4Hz, 6Hz, 23Hz, 30Hz]
Iteration 5	[4Hz, 6Hz, 28Hz, 35Hz]
Iteration 6	[4Hz, 6Hz, 33Hz, 40Hz]
Iteration 7	[4Hz, 6Hz, 38Hz, 45Hz]
Iteration 8	[4Hz, 6Hz, 43Hz, 50Hz]
Iteration 9	[4Hz, 6Hz, 48Hz, 55Hz]

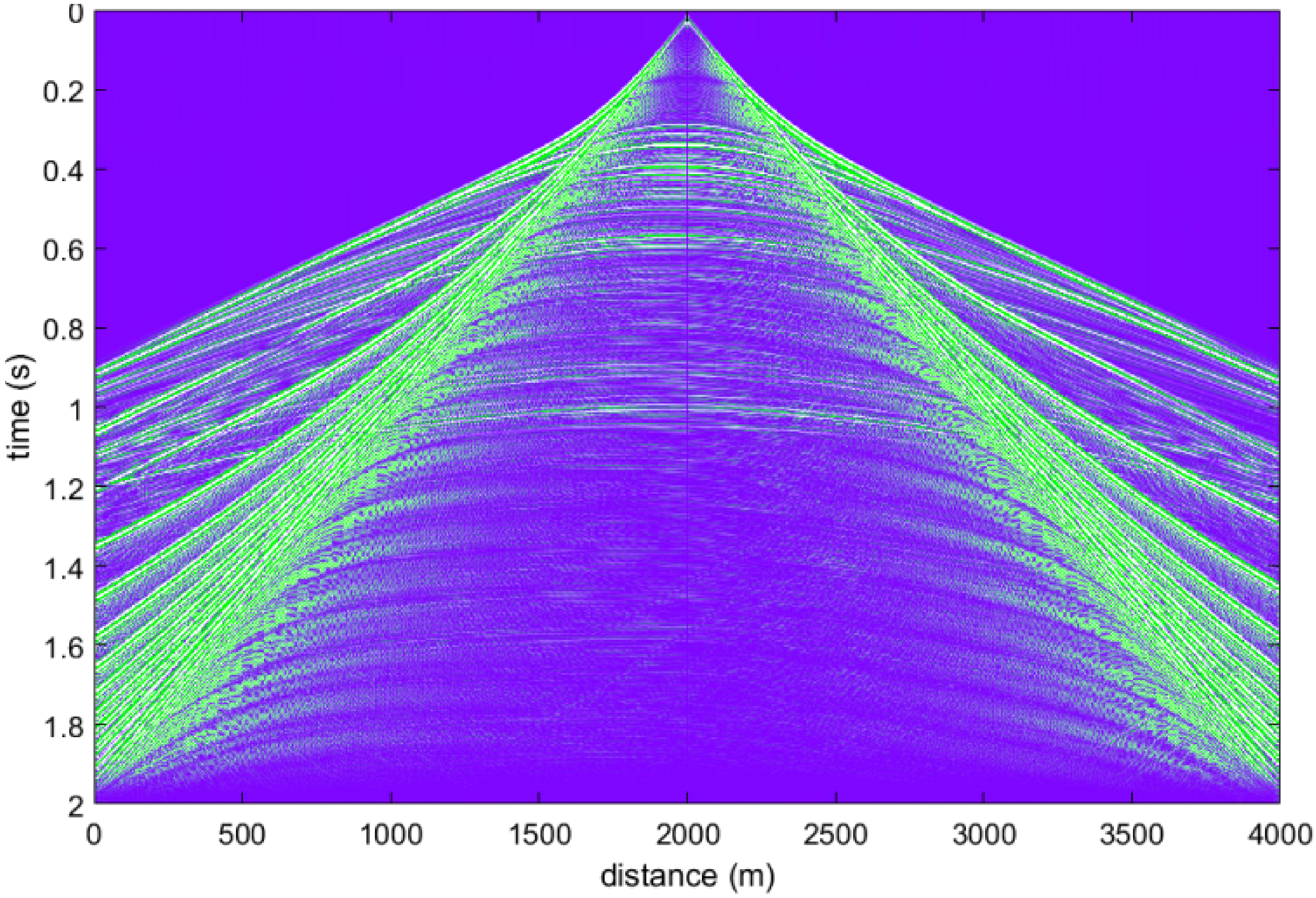
"…A multiplicative scalar is perhaps the simplest possible form of matching to a well, and there are many more possible adjustments including phase rotation, gain adjustment, match filtering, wavelet estimation and deconvolution, dynamic time warping, and more.

Most of these methods are routinely used in industry and most are time-domain methods while the gradient is typically estimated in depth. Thus a depth to time conversion may be useful for optimal matching".

(Margrave, Crewes Research Report, 2015)

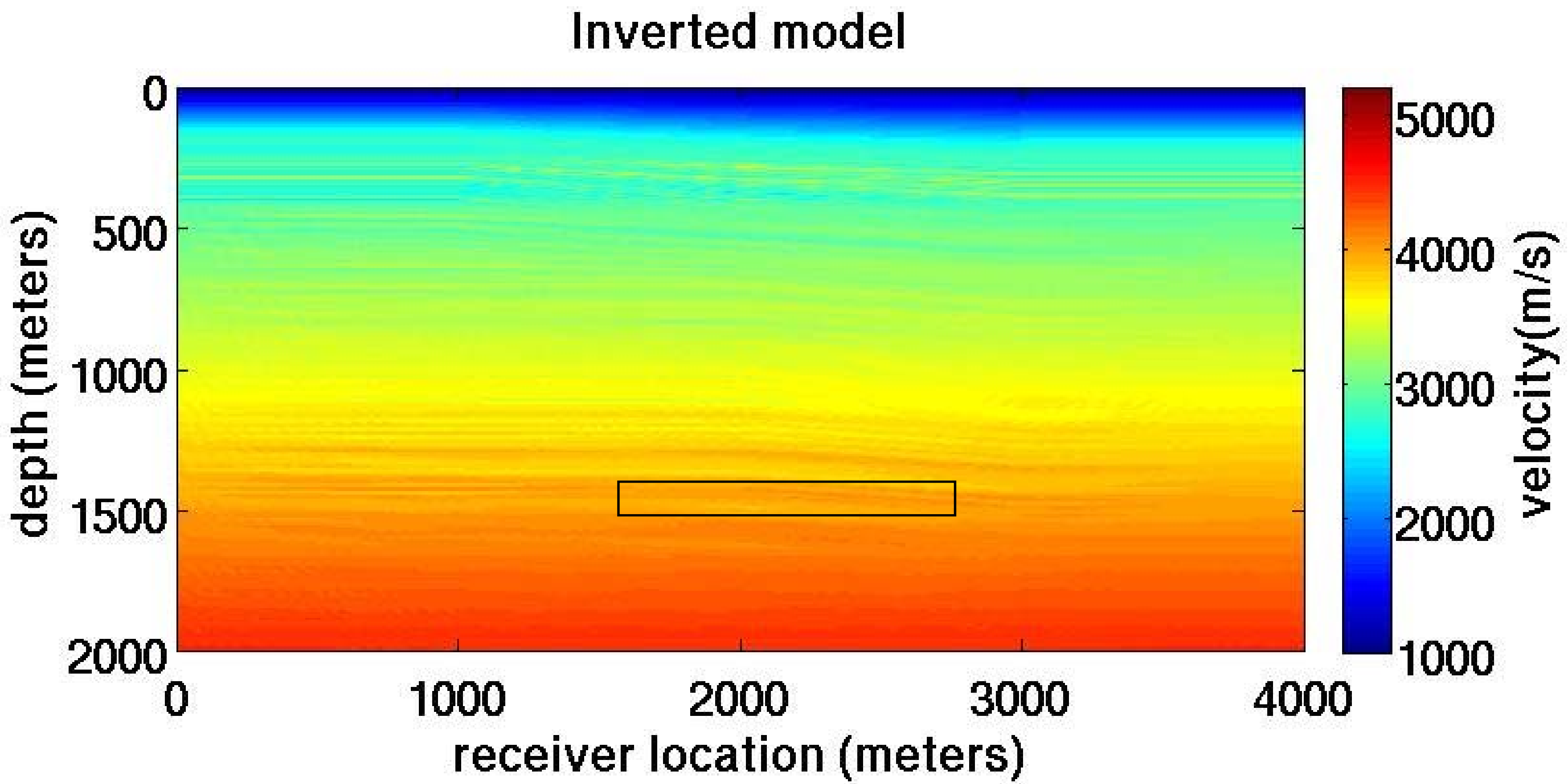
Linear v(z) model, turning-rays preserved

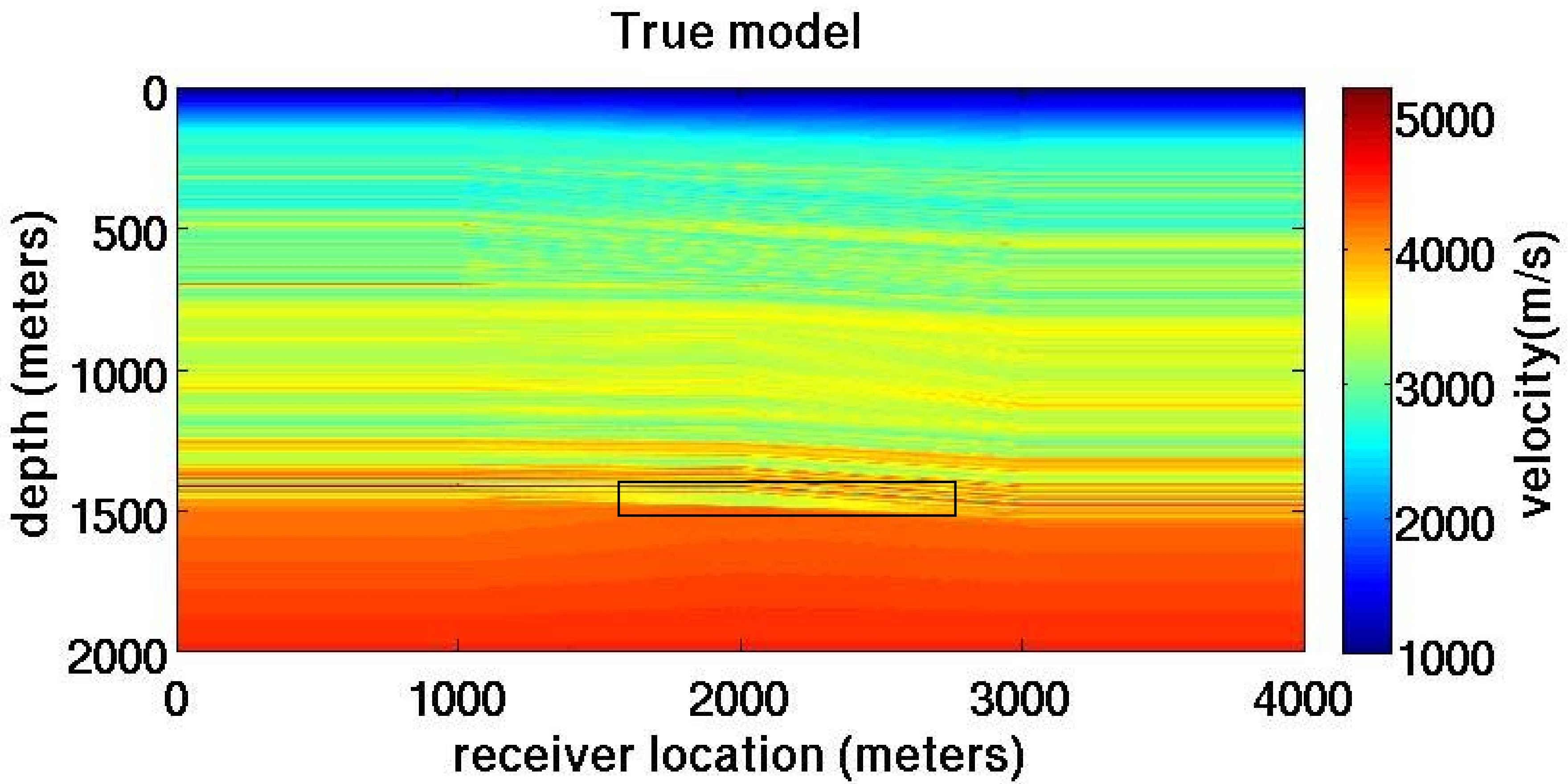




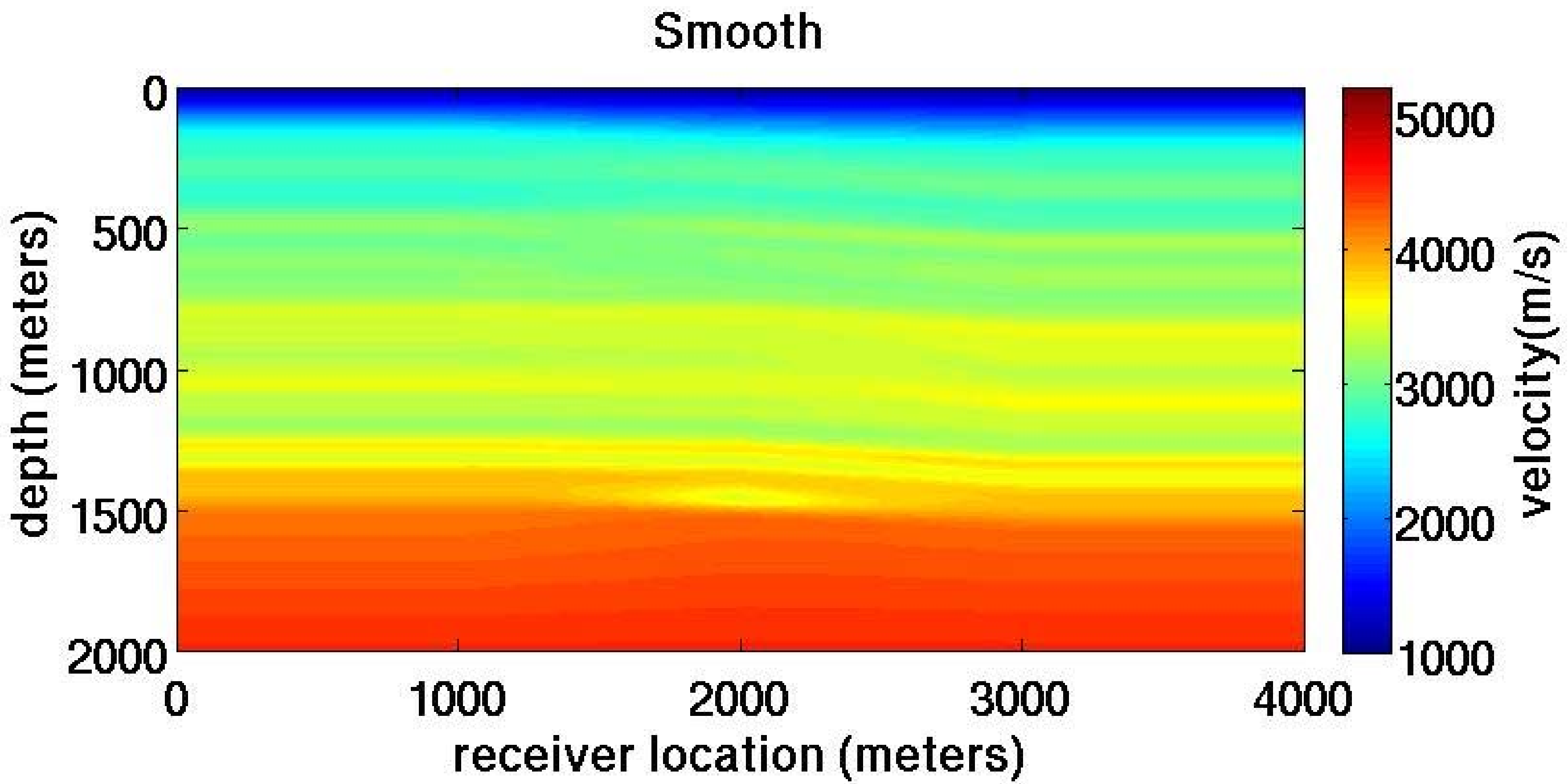
A shot gather from the centre of the line. (Margrave, 2015)

Inverted velocity model after 9 iterations

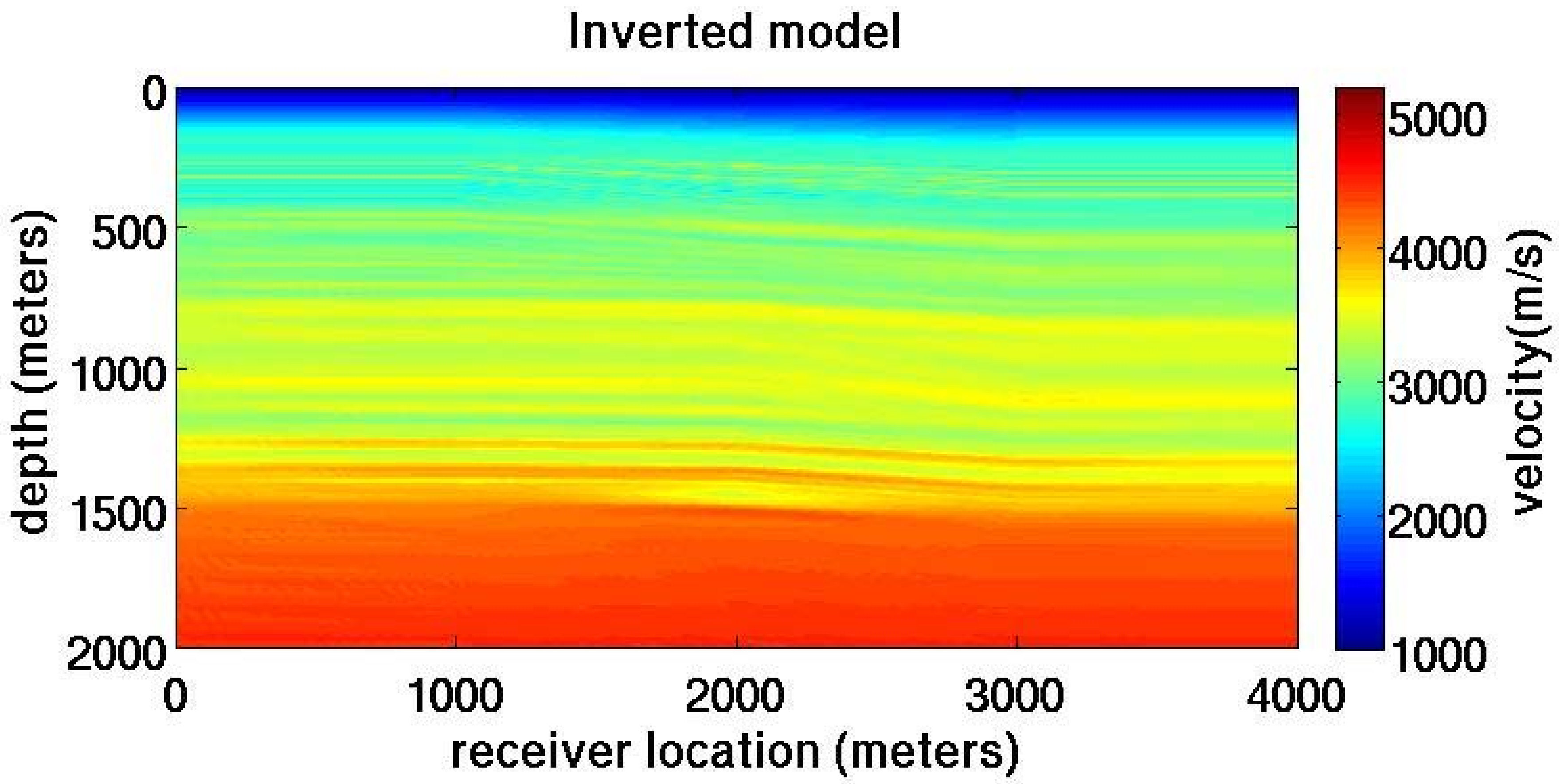


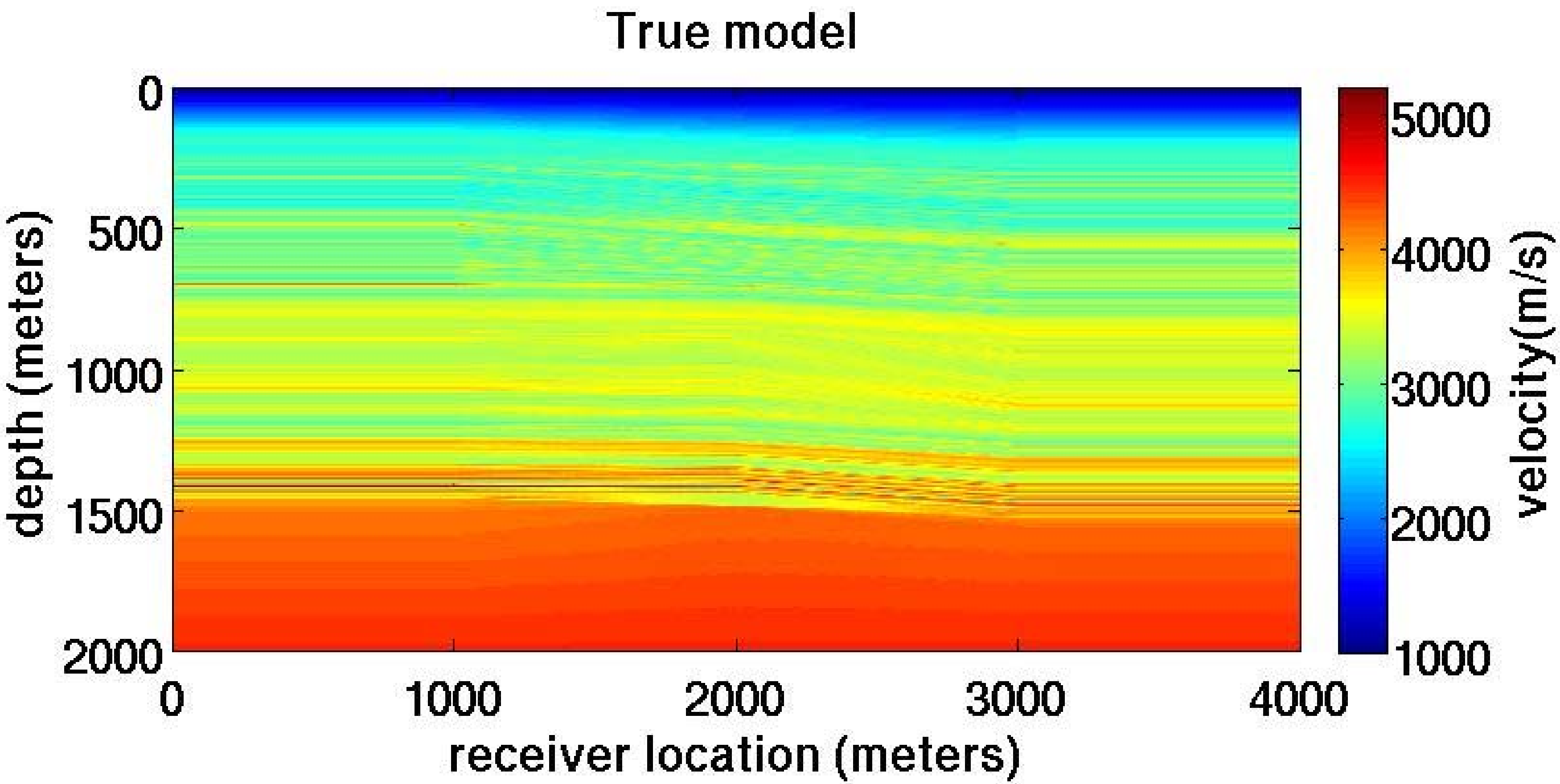

True model
depth (meters)
0
500
1000
1500
2000
0
1000
2000
3000
4000
receiver location (meters)
5000
4000
3000
2000
1000
velocity(m/s)

By convolving the true model with a Gaussian smoother

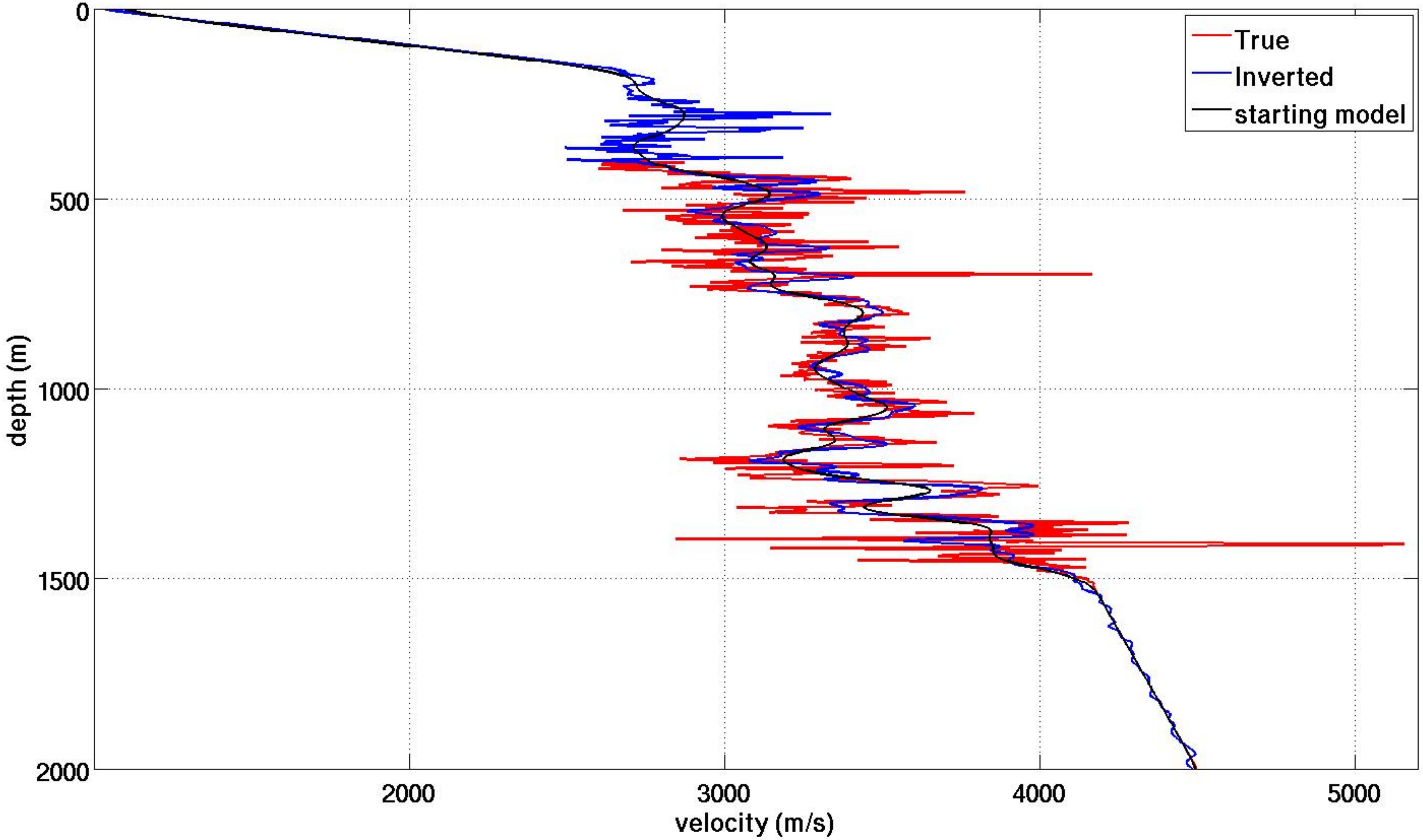


Inverted velocity model after 8 iterations

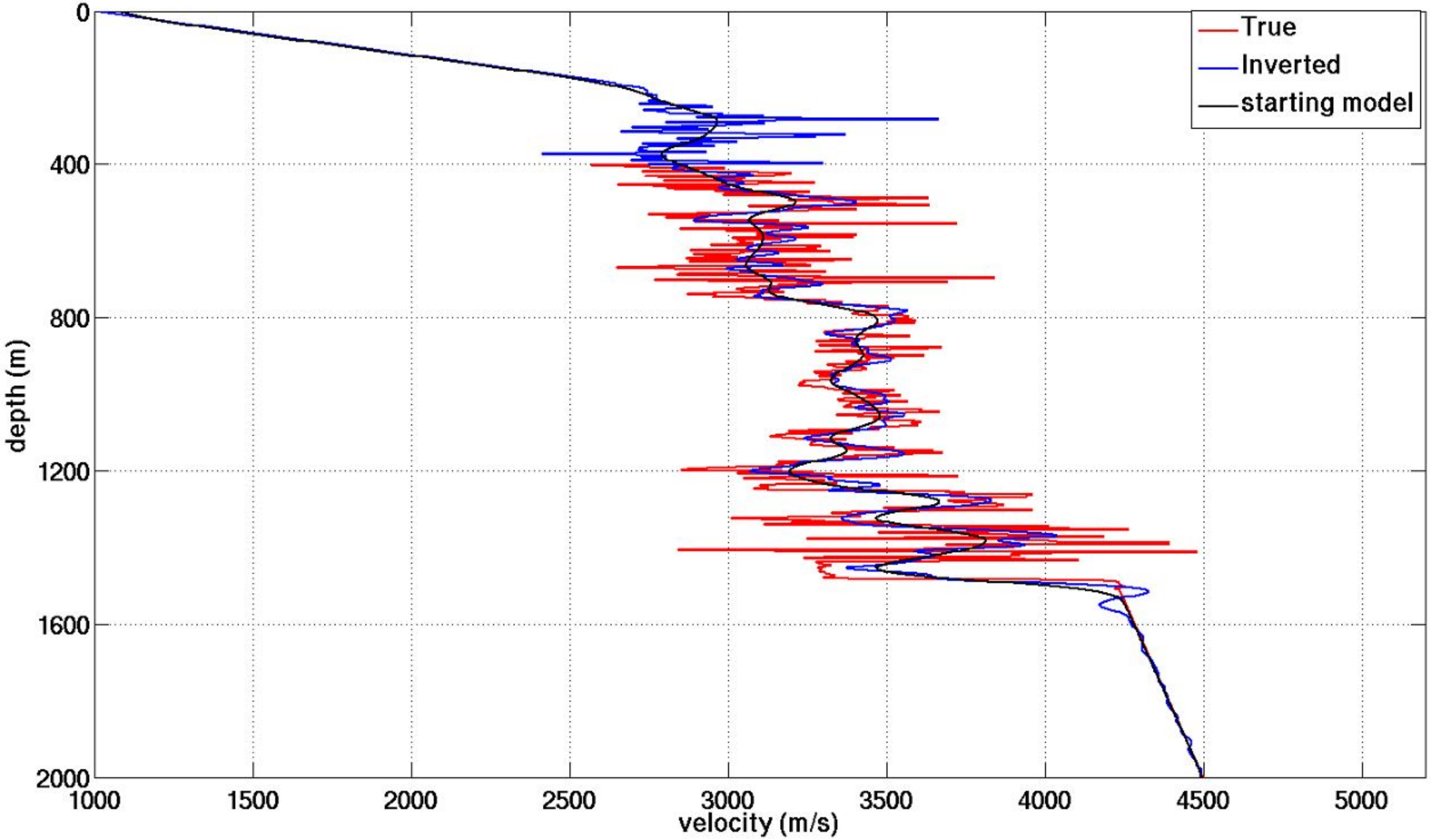



True model
0
500
1000
1500
2000
depth (meters)
0
1000
2000
3000
4000
receiver location (meters)
5000
4000
3000
2000
1000
velocity(m/s)

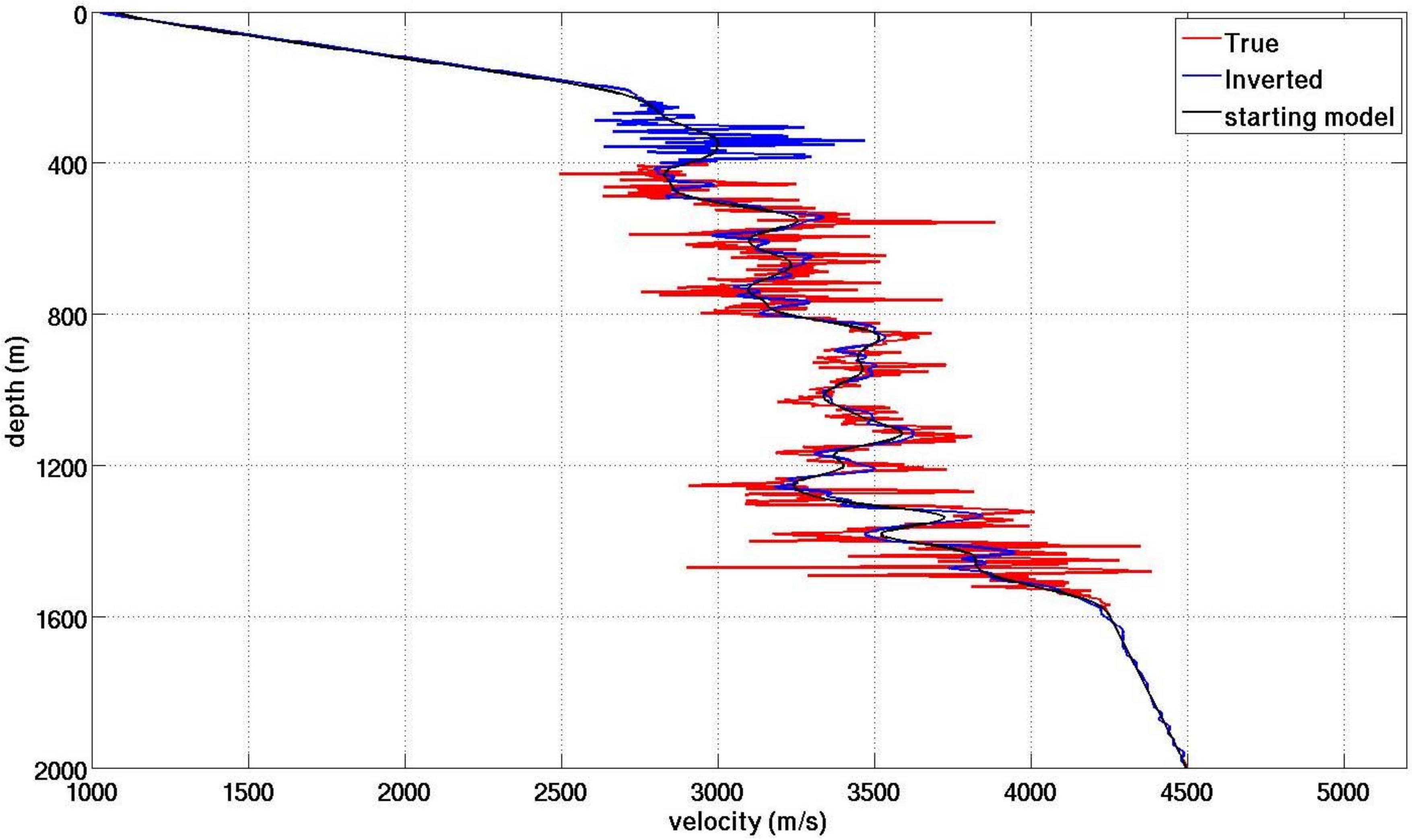
Vertical Profile of well 14-35



Vertical Profile of well 14-27



Vertical Profile of well 12-27



Conclusions

»We have been able to recover inverted models in about 9 iterations for the linear v(z) starting model and in about 8 iterations for the smoothed version of the true model as starting model.

»Same again as before, information from well logs to calculate a scalar for the model update shows encouraging results and could save us some computational time compared with the line search method.

» We had better inverted result when the stating model is close to the true model as is the case using a slightly smoothed version of the true model as the starting model.

»Ultimately, better inverted results are got when the starting model is close enough, thus other methods of estimating subsurface velocity such as TRT, RTTT, MVA, should serve as sources of starting model for FWI.

Future Work

»**Exploit the full possibilities of incorporating into the CG algorithm, wavelet estimation, phase rotation, and a depth to time conversion**

» **Performing this experiment again but using the Pica et al, 1990 line search code in lieu of the scalar derived from well information in the CG algorithm in order to see if there will be improved results in 8 iterations as in the case of the Marghuin model or 11 iterations from previous studies (2014)**

»**Testing on real data**

Acknowledgements

- CREWES Sponsors
- NSERC through the grant CRDPJ 461179-13
- Acceleware for providing the RTM codes
- Marcelo Guarido
- CREWES staff and students