Full Waveform Inversion of Crosswell Seismic Data Using Automatic Differentiation

Wenyuan Liao
Department of Math. & Stat., University of Calgary

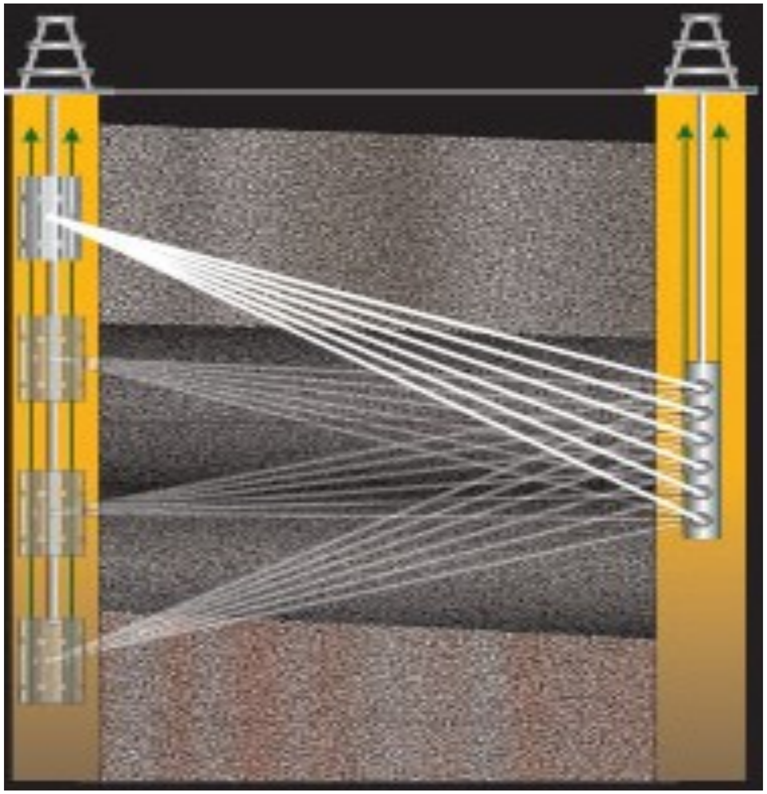
Danping Cao
School of Geosciences, China University of Petroleum

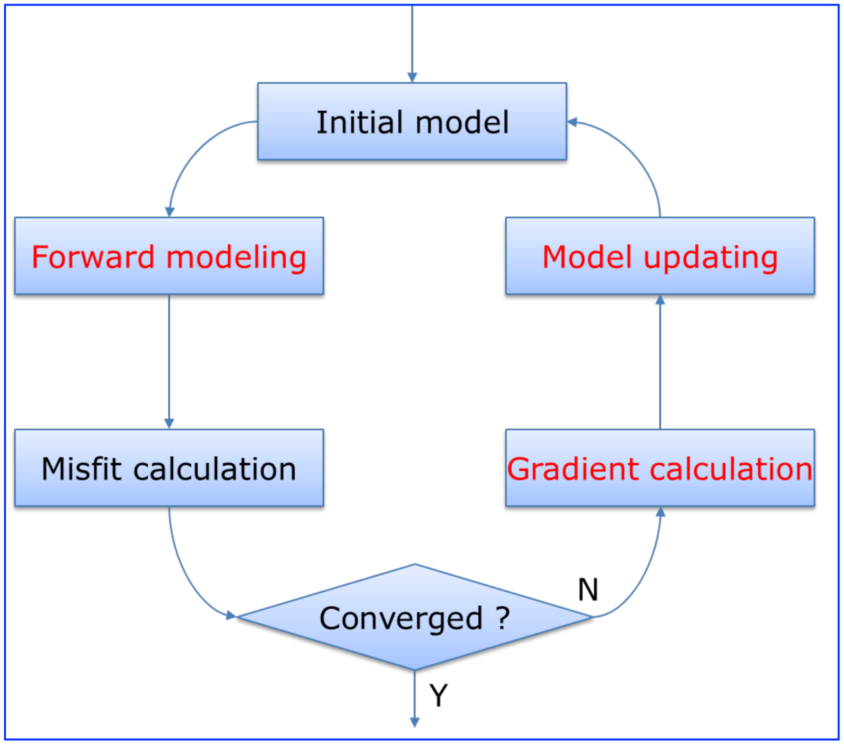
CREWES Tech Talk
February 1, 2013

Outline

1. Introduction
2. Adjoint State Method
3. Automatic Differentiation(AD)
4. FWI using AD
5. Model Test
6. Conclusions

Introduction





Fixed Receivers – varying sources

Workflow of FWI

Introduction

- Mathematical Formulation: PDE-constrained Optimization

$$\mathcal{J}(m) = \frac{1}{2}\int_0^{t_f}\sum_{i=1}^{N_r}(d^i_{obs} - d^i_{cal}(m))^2\,dt + \kappa\|m\|$$

where

m : Model parameter(wave velocity)

d^i_{obs} : Observational data

$d^i_{cal}(m)$: Synthetic seismogram based on m through the wave eq.

$\kappa\|m\|$: Regularity term (Optional, depending on prior knowledge)

The inverse problem is solved through

$$\min_{m\in\mathbb{H}(\Omega)} \mathcal{J}(m)$$

Introduction

- PDE-Constrained Optimization: Gradient Calculation

$$\frac{\partial \mathcal{J}}{\partial m} = -\int_0^{t_f} \sum_{i=1}^{N_r} \left((d^i_{obs} - d^i_{cal}) \cdot \frac{\partial d^i_{cal}}{\partial u} \cdot \frac{\partial u}{\partial m} \right) dt + \kappa \frac{\partial \|m\|}{\partial m}$$

Direct computation of $\frac{\partial u}{\partial m}$ is difficult and expensive!

Adjoint-state method is an effective way to resolve this issue

Outline

1. Introduction
2. Adjoint State Method
3. Automatic Differentiation(AD)
4. FWI using AD
5. Model Test
6. Conclusions

Adjoint State Method

- Define the cost functional as

$$\mathcal{J}(m) = \mathcal{J}(u(m), m)$$

which may depend on the model parameter implicitly if no regularity term.

The governing PDE(acoustic wave equation in this case) is stated as

$$\mathbb{L}(u(m), m) = 0$$

Here $\mathbb{L}$ is an operator defining the initial-boundary value problem of the wave equation.

Adjoint State Method: Perturbation Theory

Introduce a perturbation to the parameter:

$$\delta m \Rightarrow \delta u \Rightarrow \delta \mathcal{J}$$

$$\mathbb{L}(u, m) = 0 \qquad + \qquad \mathbb{L}(u + \delta u, m + \delta m) = 0$$

$$\delta \mathcal{J} = \left(\frac{\partial \mathcal{J}(u, m)}{\partial m} - \left\langle \xi, \frac{\partial \mathbb{L}(u, m)}{\partial m} \right\rangle \right) \delta m$$

where the adjoint-state variable is defined as

$$\left[\left(\frac{\partial \mathbb{L}(u, m)}{\partial u} \right)^* \right] \xi = \left[\frac{\partial \mathcal{J}(u, m)}{\partial u} \right] \Rightarrow \xi = \left[\left(\frac{\partial \mathbb{L}(u, m)}{\partial u} \right)^* \right]^{-1} \left[\frac{\partial \mathcal{J}(u, m)}{\partial u} \right]$$

Adjoint State Method: Lagrange Multipliers

Redefine a new cost functional as

$$\tilde{\mathcal{J}}(u, m, \xi) = \mathcal{J}(u, m) - \langle \xi, \mathbb{L}(u, m) \rangle$$

Solving the unconstrained optimization problem we obtain the gradient as

$$\frac{\partial \tilde{\mathcal{J}}(u, m, \xi)}{\partial m} = \frac{\partial \mathcal{J}(u, m)}{\partial m} - \left\langle \xi, \frac{\partial \mathbb{L}(u, m)}{\partial m} \right\rangle$$

where

$$\frac{\partial \tilde{\mathcal{J}}(u, m, \xi)}{\partial u} = 0 \Rightarrow \frac{\partial \mathcal{J}(u, m)}{\partial u} - \left(\frac{\partial \mathbb{L}(u, m)}{\partial u} \right)^* \xi = 0 \Rightarrow \xi = \left[\left(\frac{\partial \mathbb{L}(u, m)}{\partial u} \right)^* \right]^{-1} \frac{\partial \mathcal{J}(u, m)}{\partial u}$$

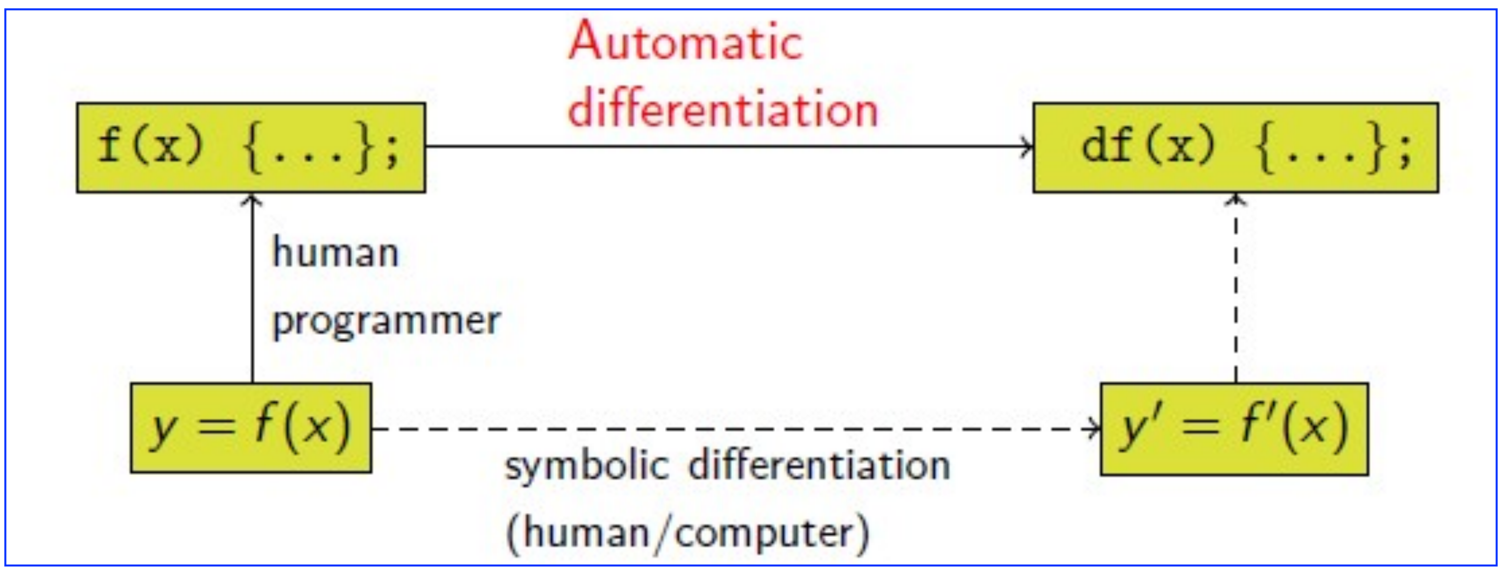
Adjoint –> Discretization or Discretization -> Adjoint

Outline

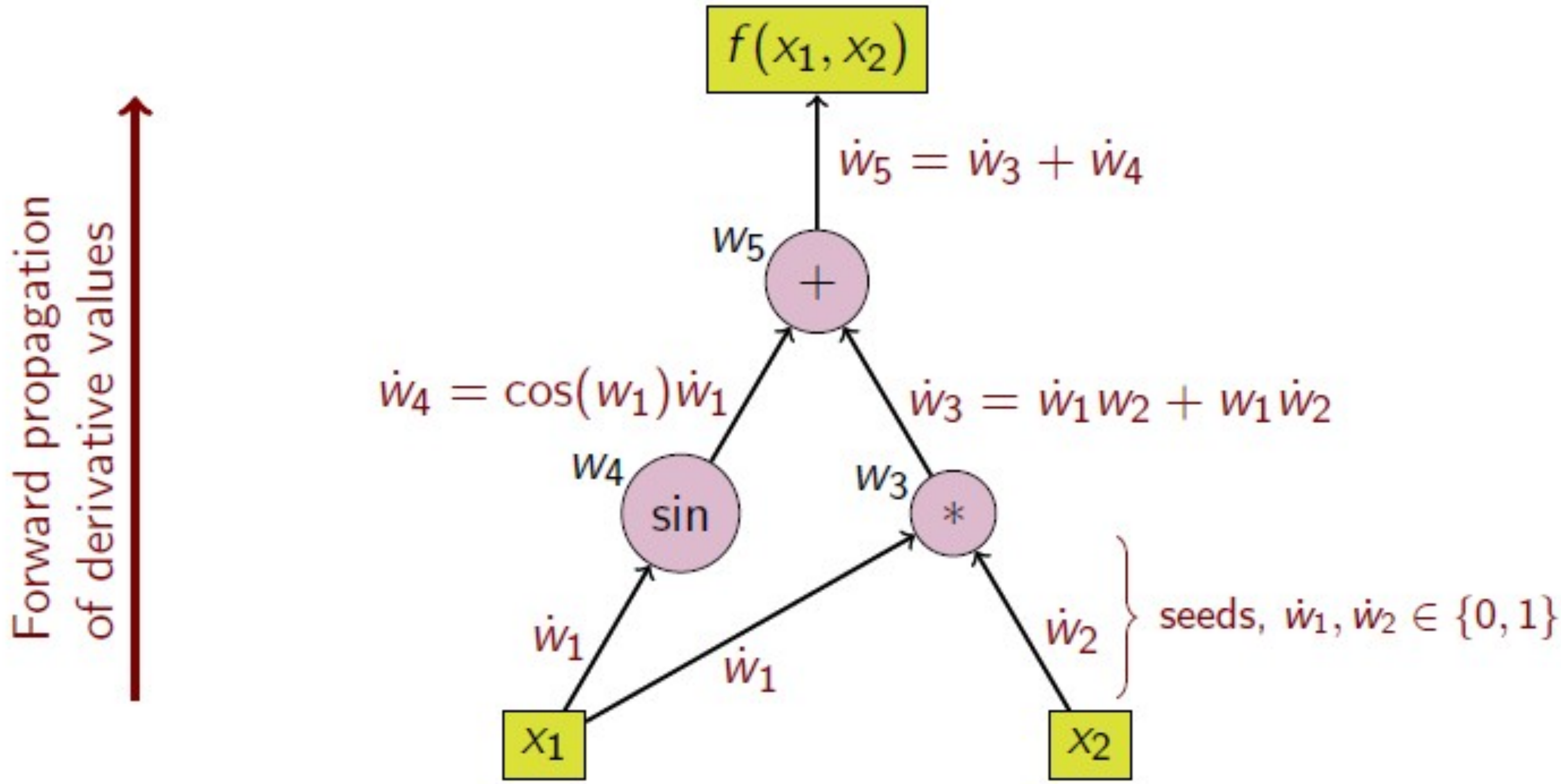
1. Introduction
2. Adjoint State Method
3. Automatic Differentiation(AD)
4. FWI using AD
5. Model Test
6. Conclusions

Automatic differentiation

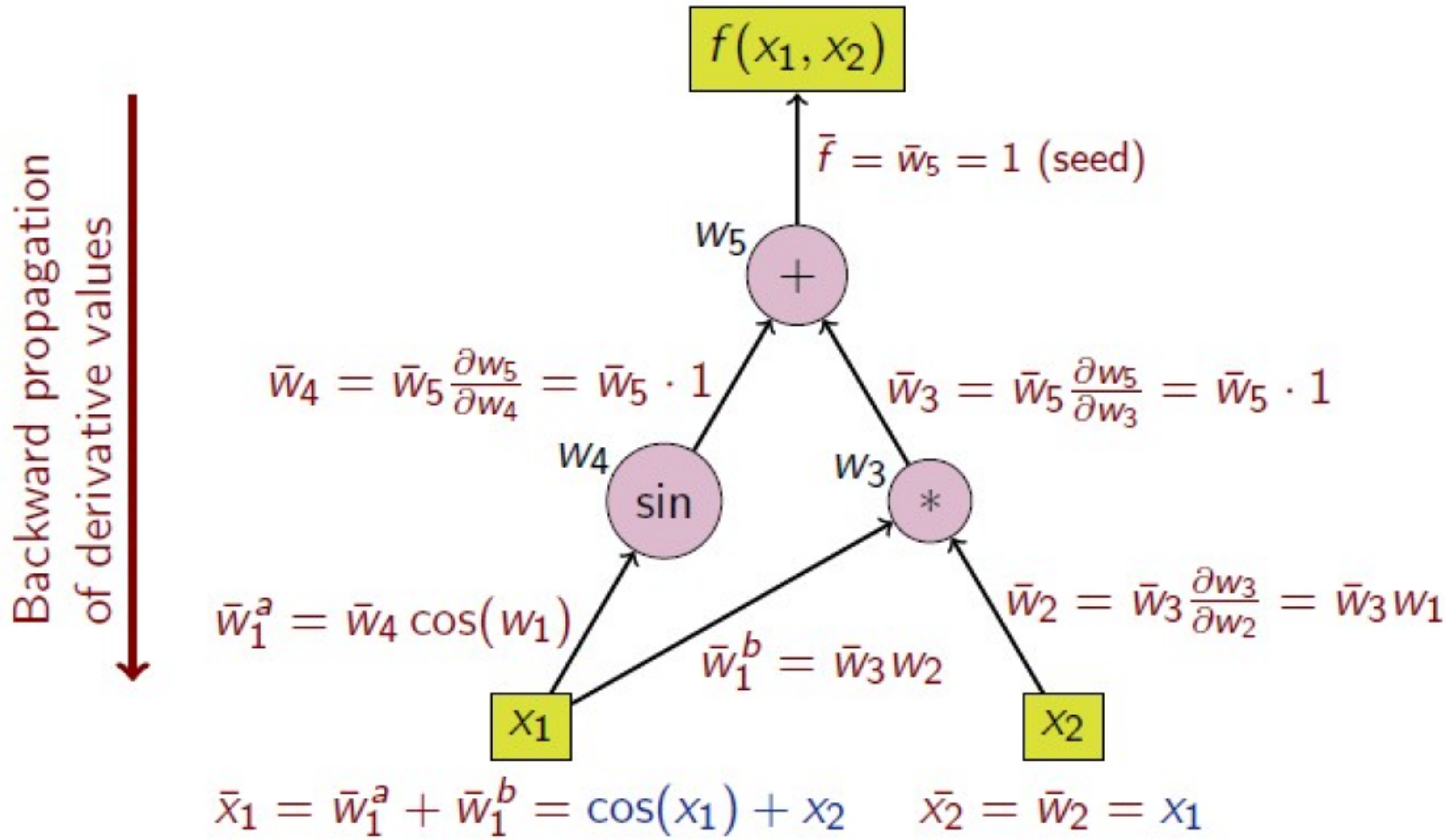
- Automatic Differentiation (AD), sometimes alternatively called algorithmic differentiation, is a set of techniques to numerically evaluate the derivative of a function specified by a computer program.



Forward mode AD



Reverse mode AD



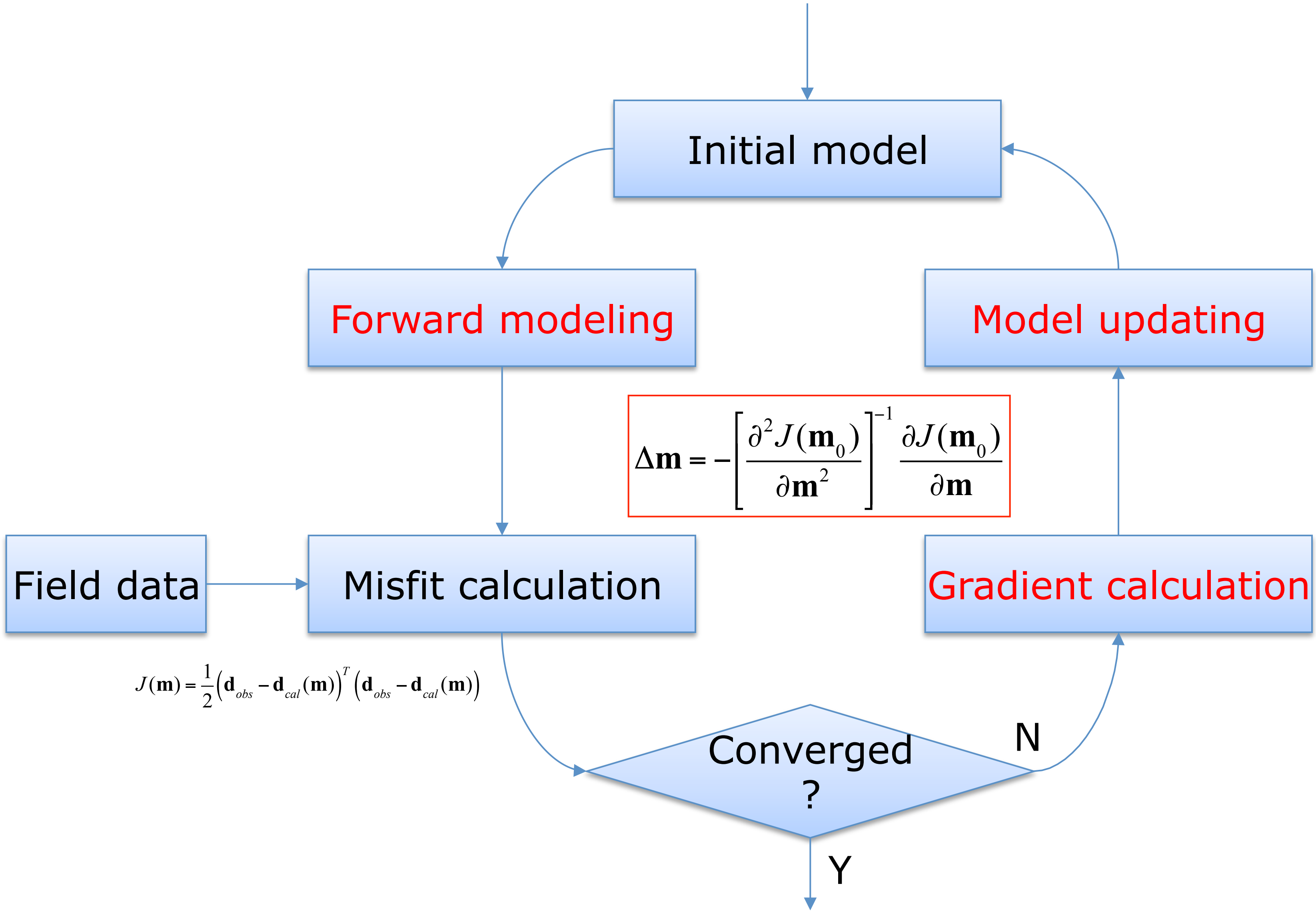
AD tools

- AD Model Builder (C/C++)
- ADC (C/C++)
- ADF (Fortran77,Fortran95)
- ADIC (C/C++)
- ADIFOR (Fortran77)
- ADiMat (MATLAB)
- ADMAT / ADMIT (MATLAB)
- ADOL-C (C/C++)
- ADOL-F (Fortran95)
- APMonitor (Interpreted)
- AUTODIF (C/C++)
- AutoDiff .NET (.NET)
- AUTO_DERIV (Fortran77/95)
- ColPack (C/C++)
- COSY INFINITY (Fortran77/95,C/C++)
- CppAD (C/C++)
- CTaylor (C/C++)
- FAD (C/C++)
- FADBAD/TADIFF (C/C++)
- FFADLib (C/C++)
- GRESS (Fortran77)
- HSL_AD02 (Fortran95)
- INTLAB (MATLAB)
- NAGWare Fortran 95 (Fortran77,Fortran95)
- OpenAD (C/C++,Fortran77/95)
- PCOMP (Fortran77)
- pyadolc (python)
- pycppad (Interpreted,python)
- Rapsodia (C/C++,Fortran95)
- Sacado (C/C++)
- TAF (Fortran77,Fortran95)
- TAMC (Fortran77)
- TAPENADE (C/C++,Fortran77/95)
- TaylUR (Fortran95)
- The Taylor Center (independent)
- TOMLAB /MAD (MATLAB)
- TOMLAB /TomSym (MATLAB)
- Treeverse / Revolve (C/C++,Fortran77/95)
- YAO (C/C++)

Outline

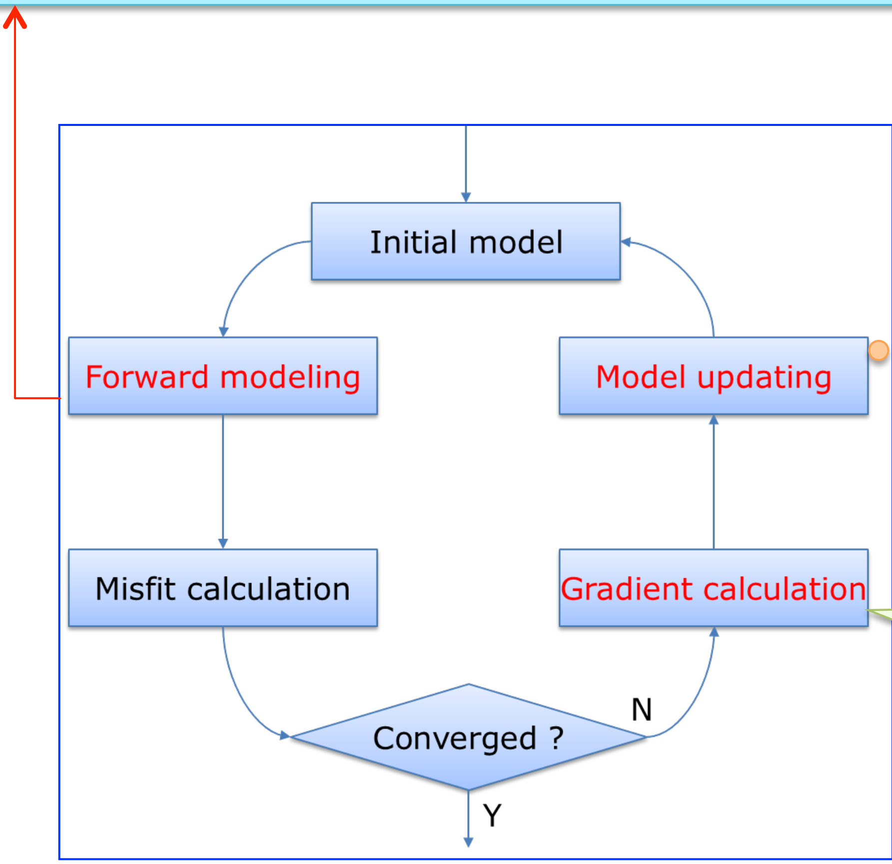
1. Introduction
2. Adjoint State Method
3. Automatic Differentiation(AD)
4. FWI using AD
5. Model Test
6. Conclusions

Workflow of FWI



FWI solution one by one

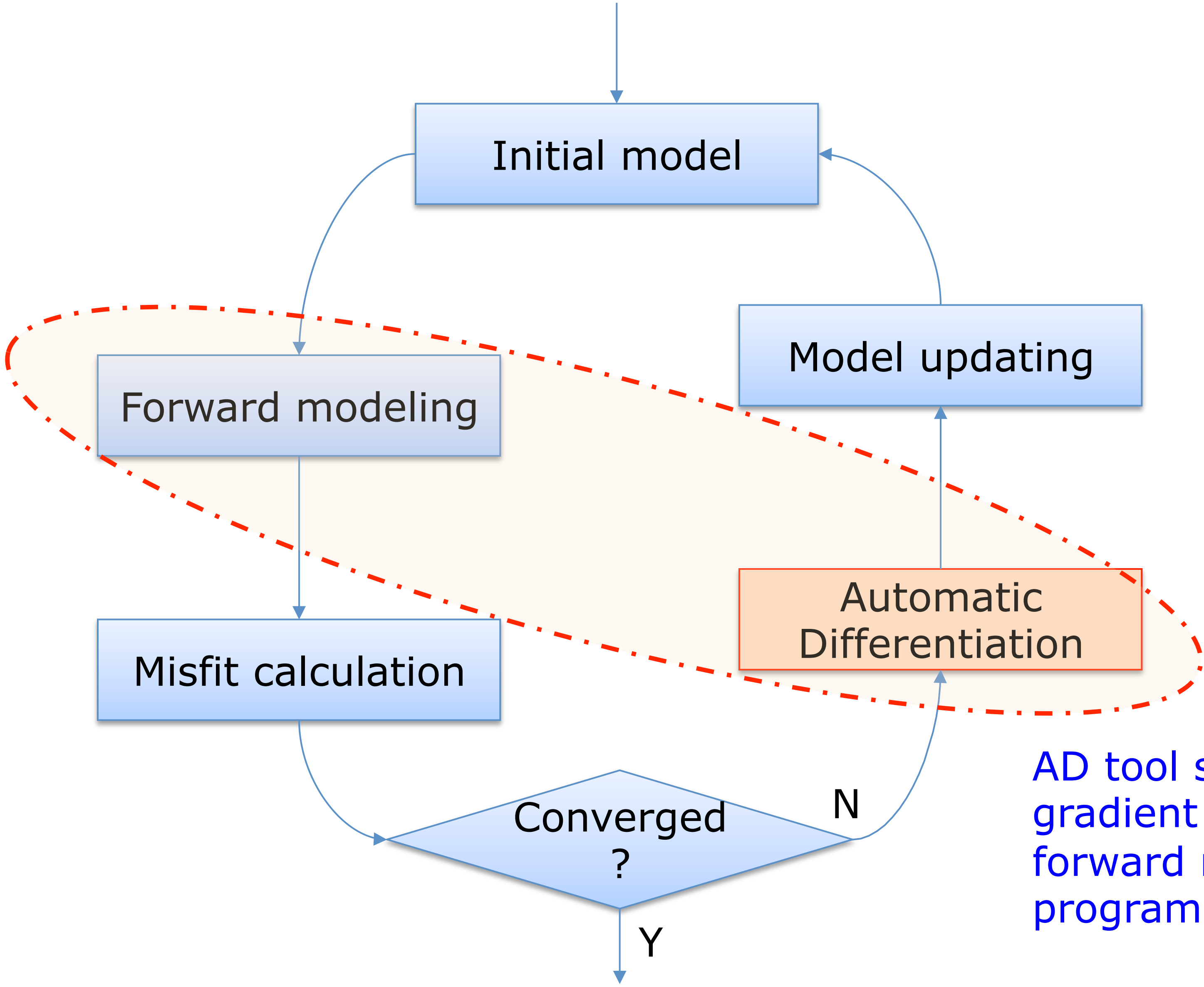
Forward Modeling:
- 2^{nd}-order in time and 4^{th}-order in space
- Stagger-grid finite difference
- PML absorbing boundary



Model Updating:
- **L-BFGS selected**
- **Limited memory required**
- **Quasi-Newton method**

Gradient Calculation:
- **Adjoint state method**
- **AD tools used**
- **TAPENADE test**

FWI workflow with AD

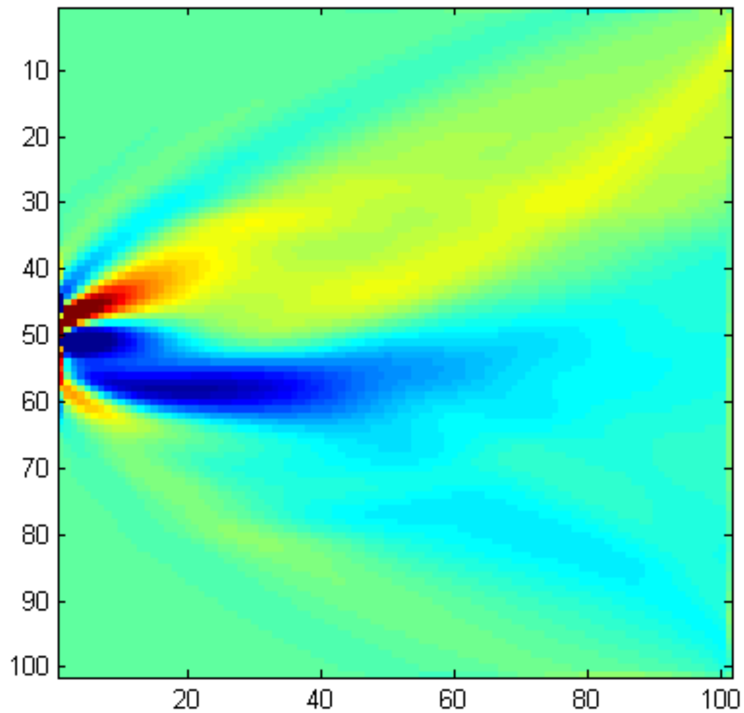


AD tool solve the gradient according to the forward modeling program

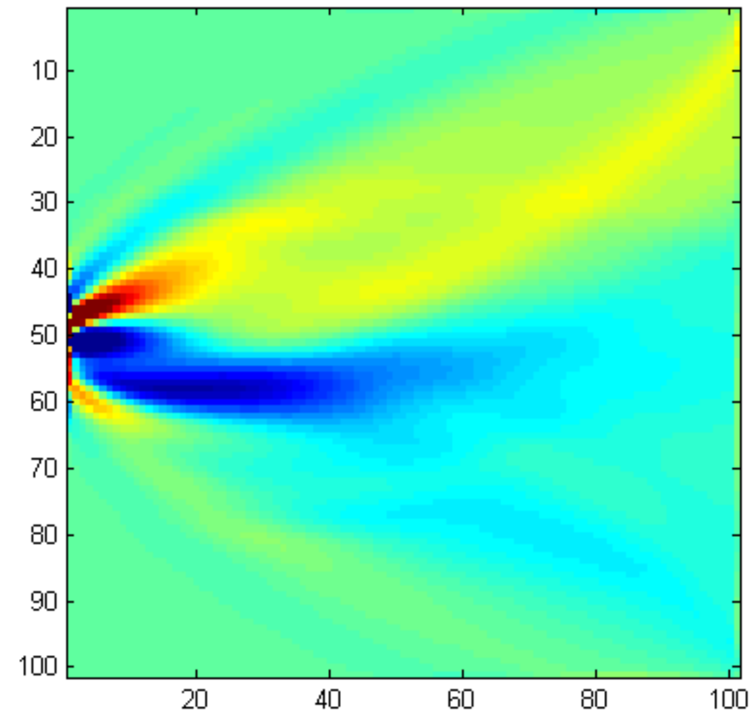
Benefit of FWI with AD

- Simplify the gradient calculation
- Focus on forward modeling and optimization method
- High efficiency forward modeling program will lead to high efficiency gradient calculation code
- FWI workflow is simplified

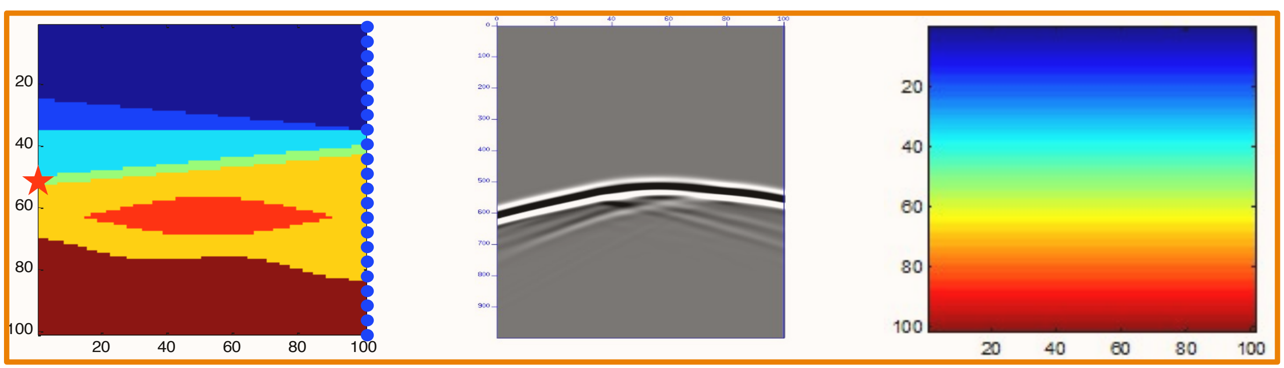
Accuracy of Gradient calculation



Gradient by TAPENADE



Gradient by central difference quotient

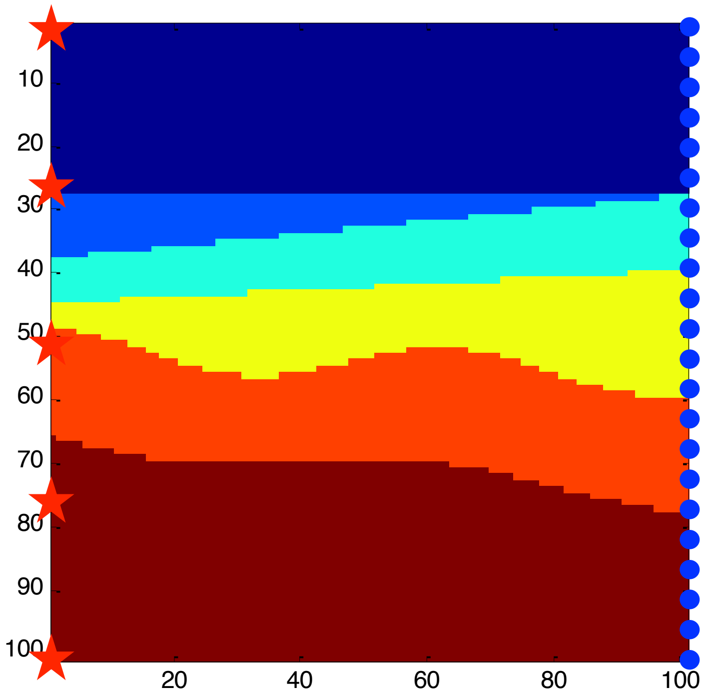


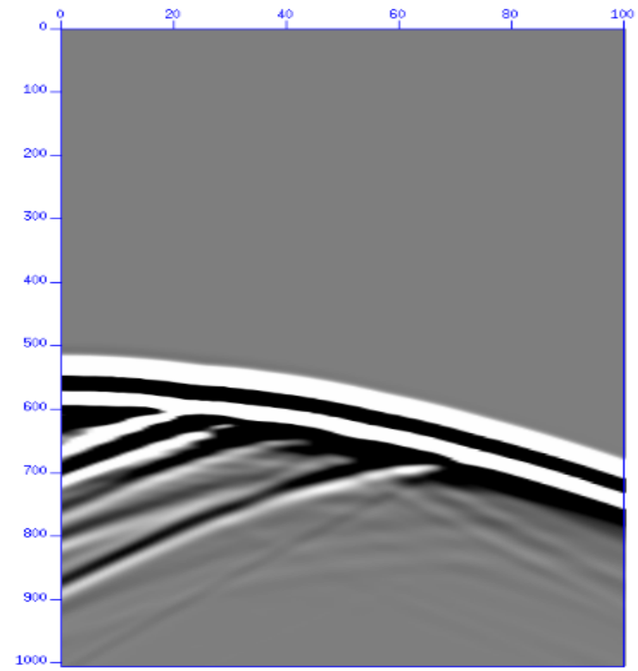
Gradient calculation:
-True model
-Synthetic record
-Initial model

Outline

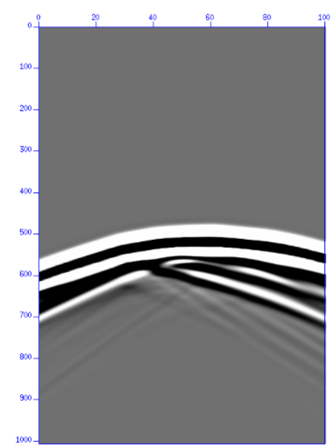
1. Introduction
2. Adjoint State Method
3. Automatic Differentiation(AD)
4. FWI using AD
5. Model Test
6. Conclusions

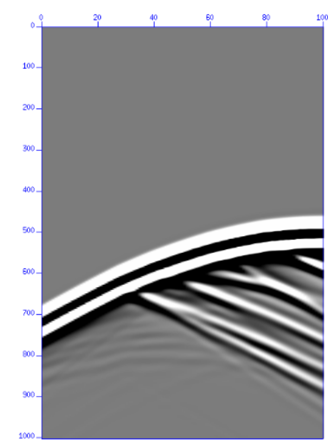
Model test 1



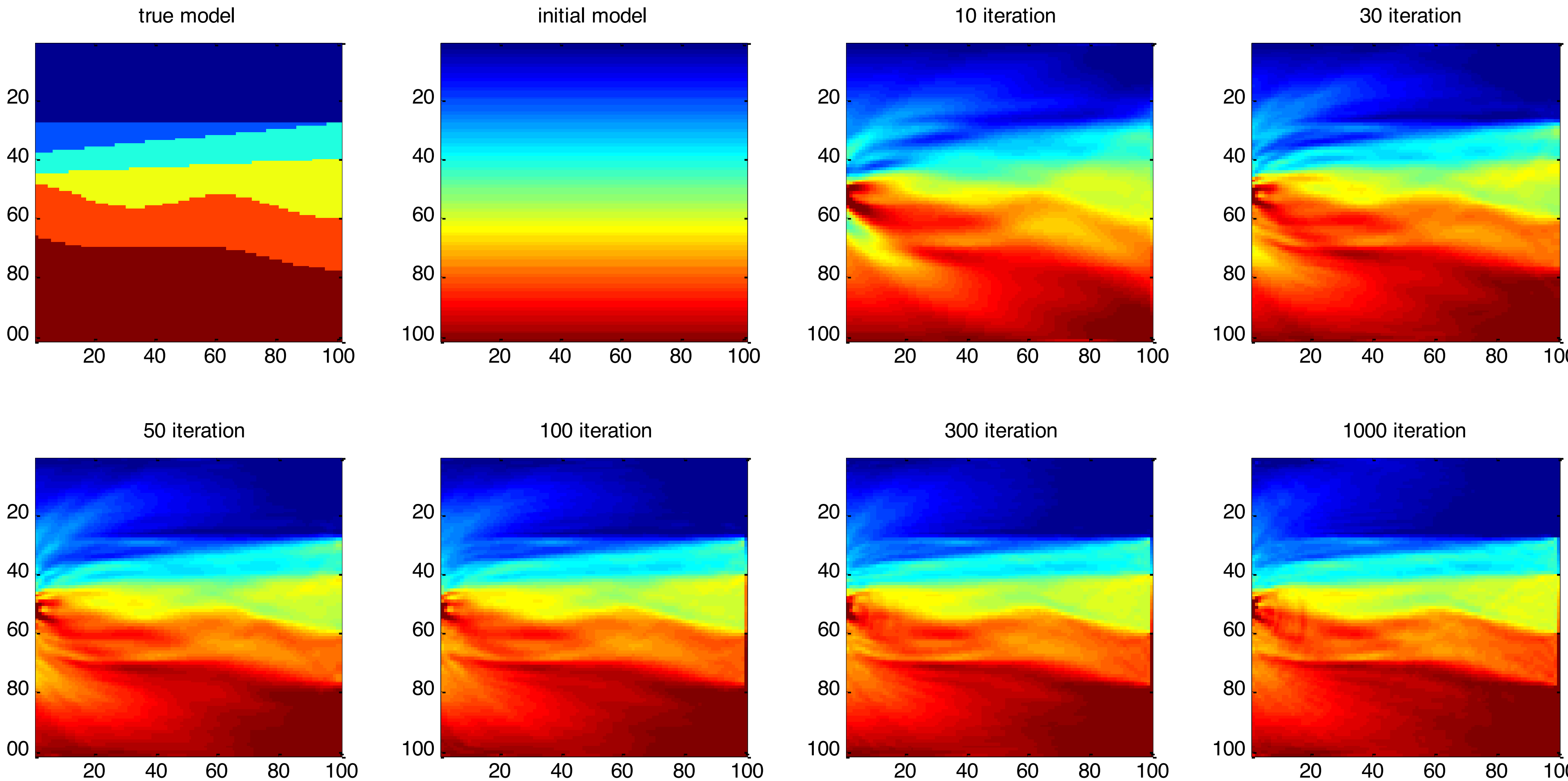


Model:101 X 101
Spatial sample:1m
Time sample:0.1ms
Source： ricker wavelet
Main frequency: 180 Hz
Boundary: PML

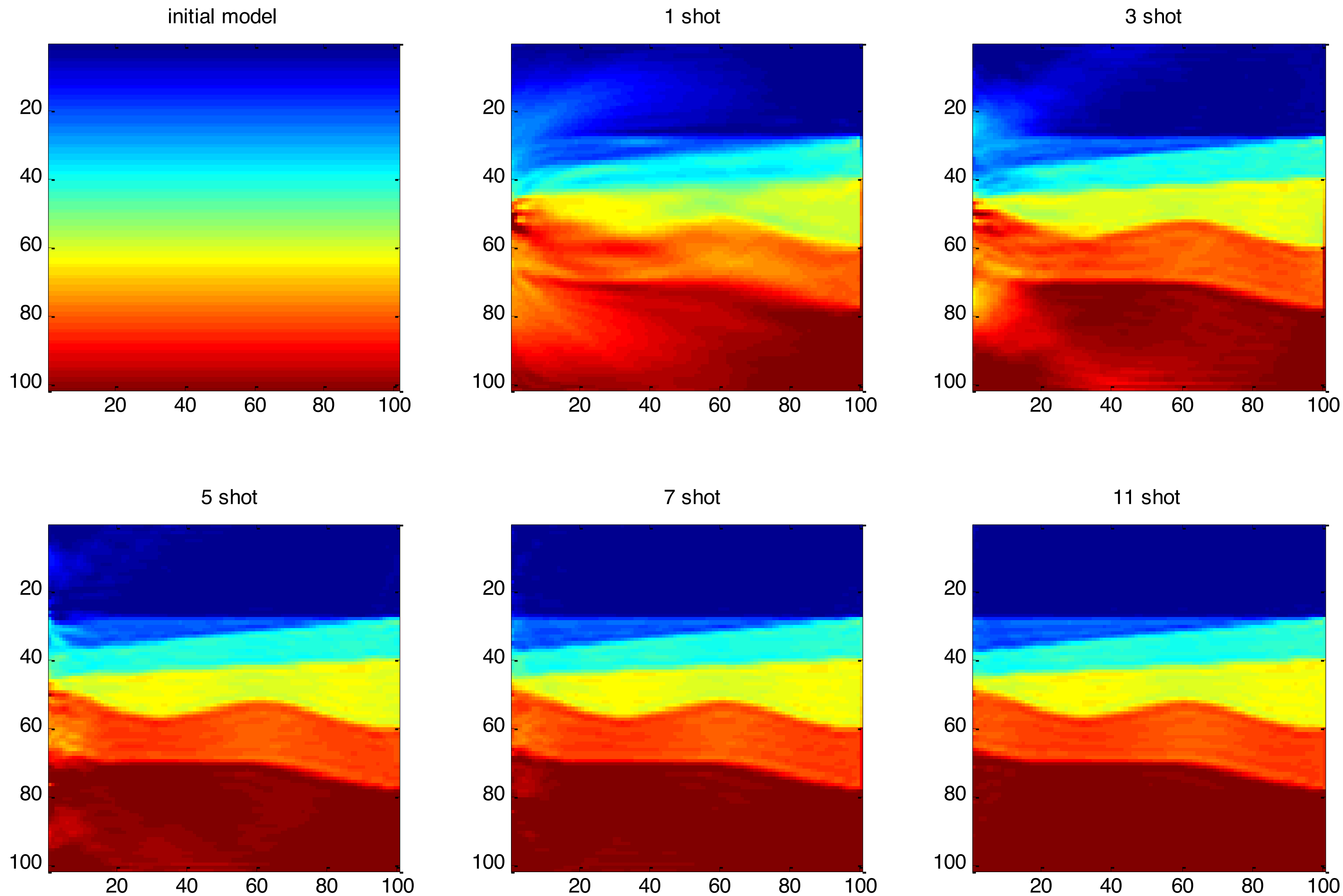




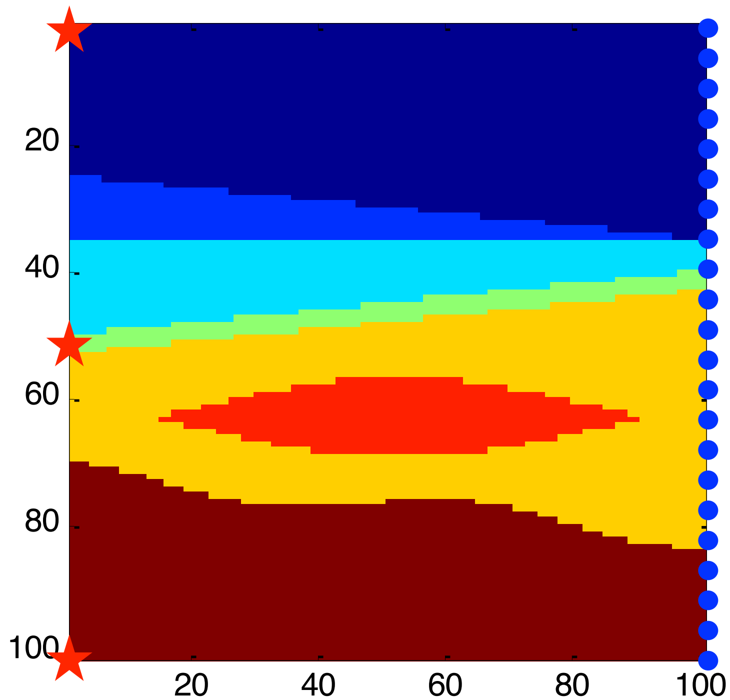
Inversion result - 1 shot

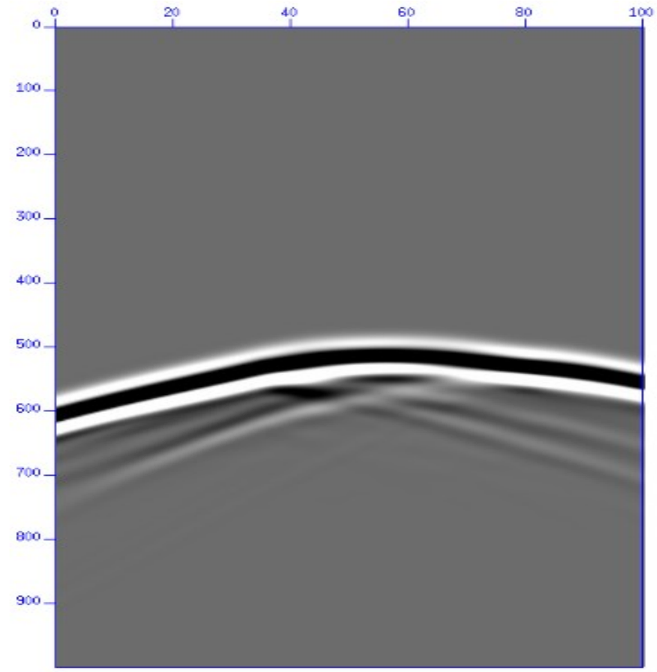


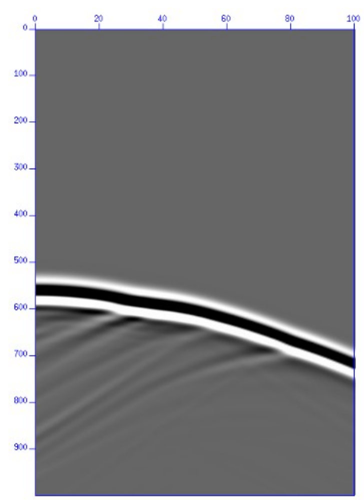
Inversion result - different shot

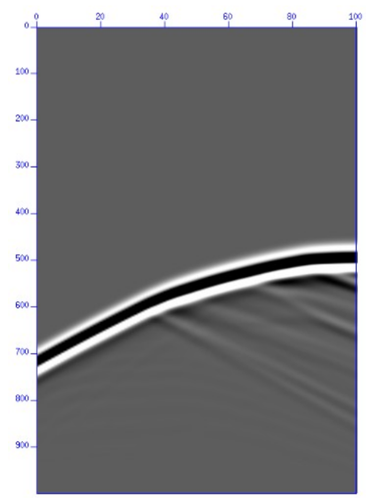


Model test 2



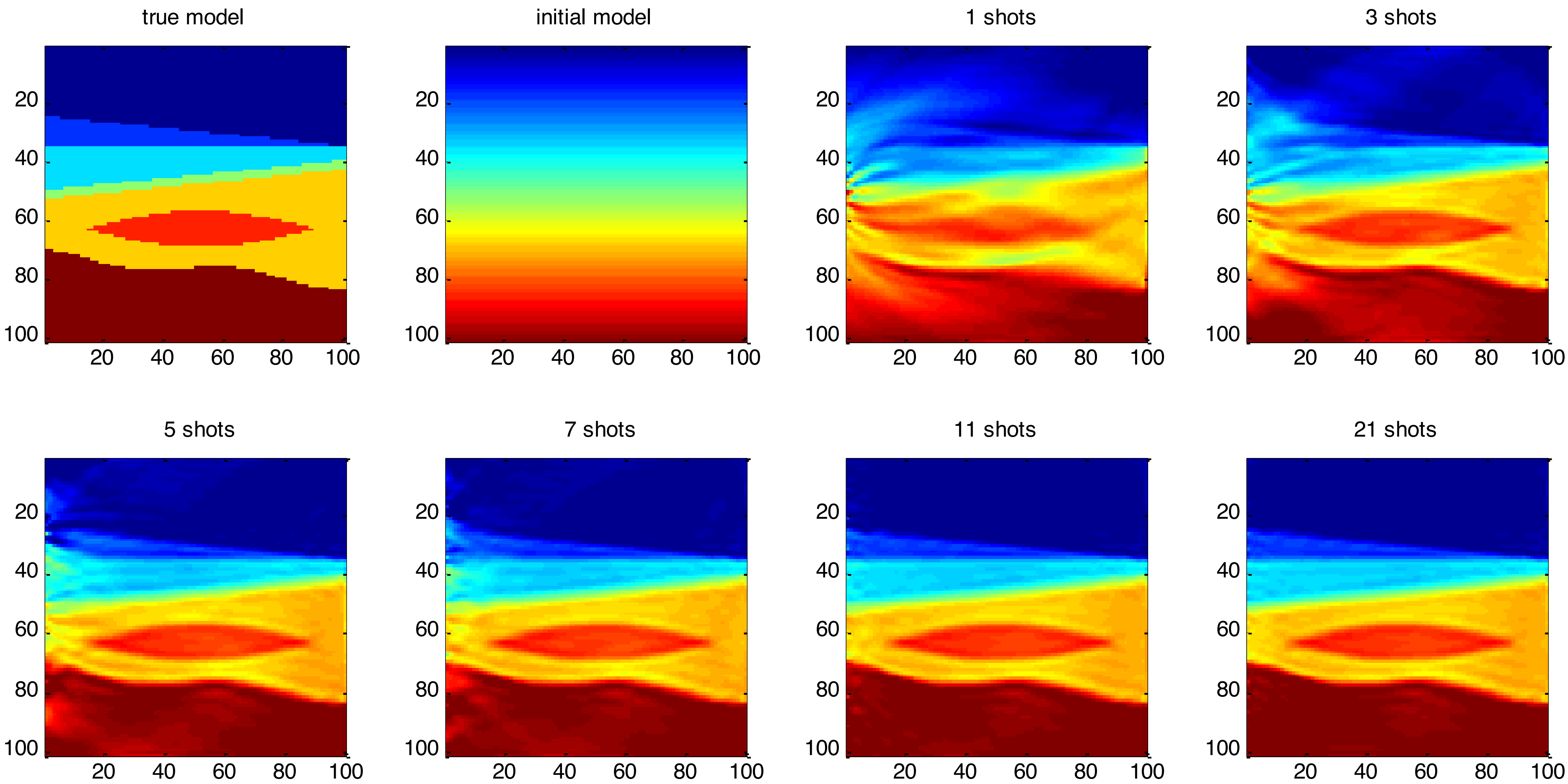




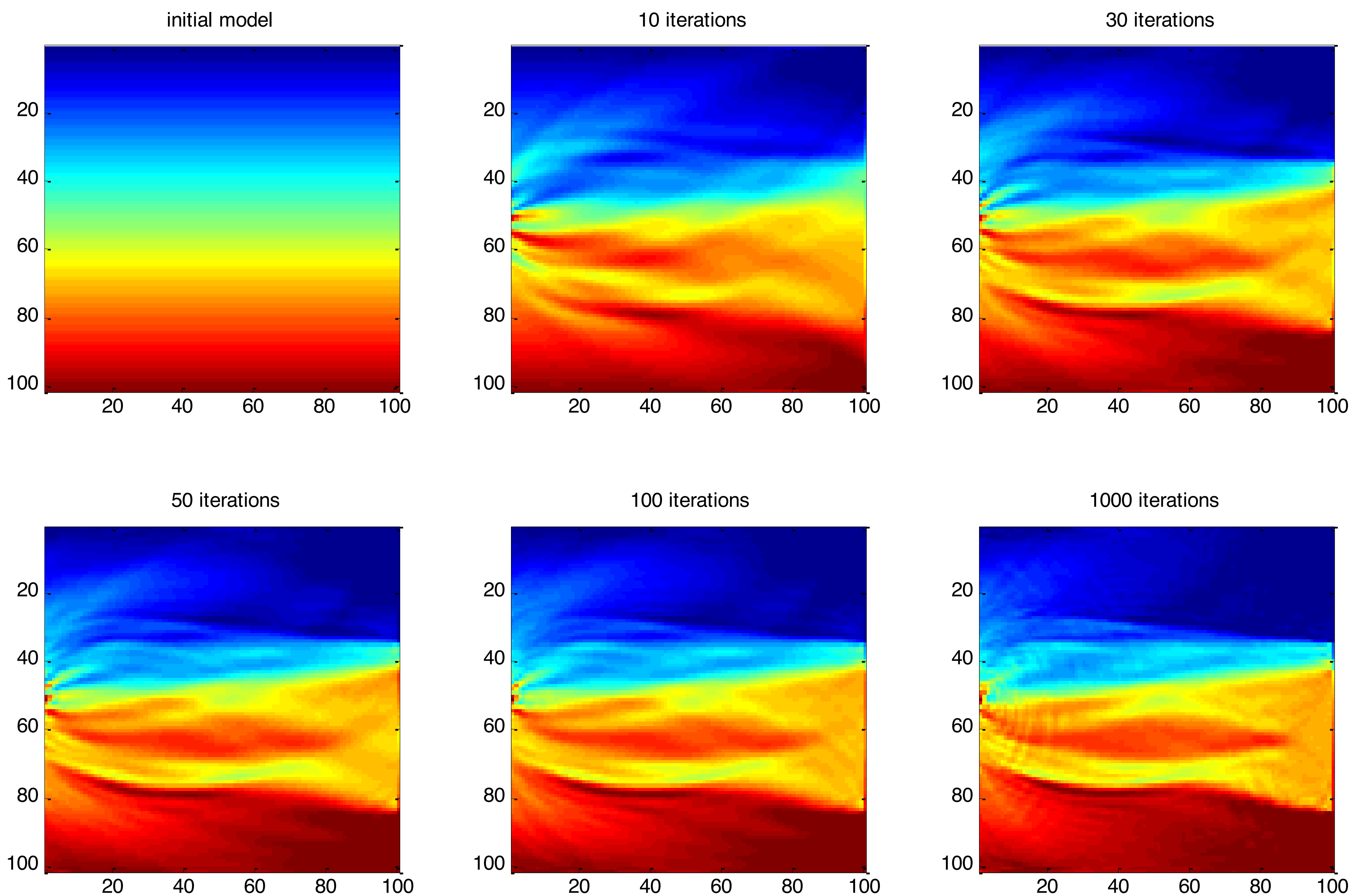


Model:101 X 101
Spatial sample:1m
Time sample:0.1ms
Source： ricker wavelet
Main frequency: 180 Hz
Boundary: PML

Inversion result - different shot



Inversion result - 1 shot



Outline

1. Introduction
2. Adjoint State Method
3. Automatic Differentiation(AD)
4. FWI using AD
5. Model Test
6. Conclusions

Conclusion

- Automatic differentiation (AD) is a promising yet not popular approach in Geoscience.
- The gradient calculated through AD is accurate.
- The full waveform inversion workflow is simplified with the usage of the AD tool.
- Model tests show that the full waveform inversion method with AD is effective and efficient in the inversion of the crosswell seismic data.

Future work

- Improve the forward modeling: finite difference 4th-order in time
- Test the large-scale data inversion using checkpoint technology
- Test with other AD tools, and Optimization algorithms
- Test the surface seismic inversion
- Address inverse modeling issues under the current framework
- Test on other types of wave equations

Acknowledgement

- Gary Margrave
- CREWES
- POTSI
- NSERC
- Department of Math. & Stat.
- Chinese Scholarship Council, National Nature Science Foundation of China