Finite-difference wave-equation migration

René-Edouard Plessix, Wim Mulder

Shell Exploration & Production

Rijswijk, The Netherlands

Outline

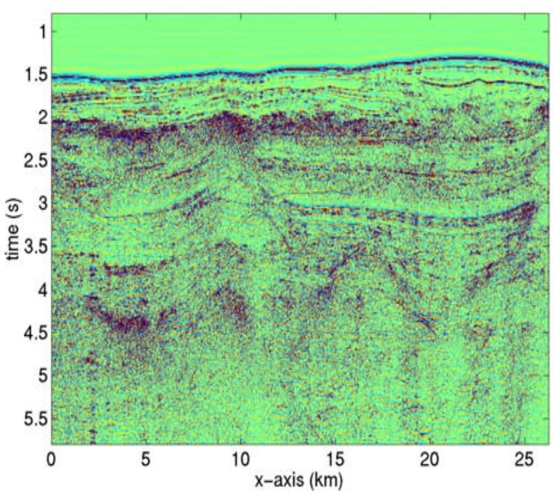
- Introduction
  - Imaging/migration
  - Ray-based approach and its limitations
- Finite difference wave equation migration
  - complexity
  - "one-way" wave equation migration
  - "full" (two-way) wave equation migration in 2D
  - examples
- Conclusions

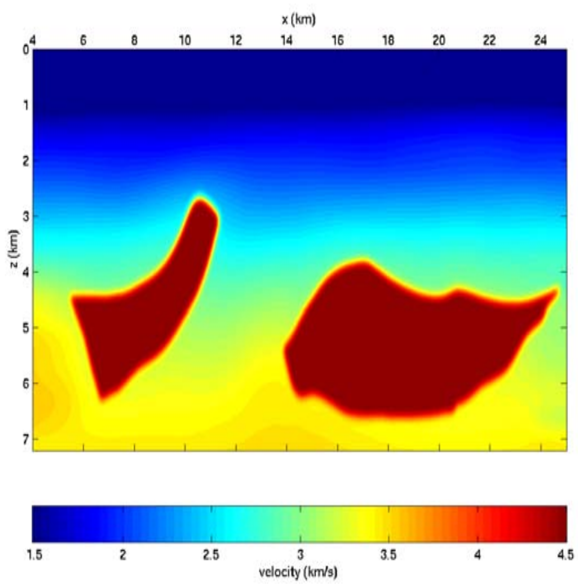
Outline

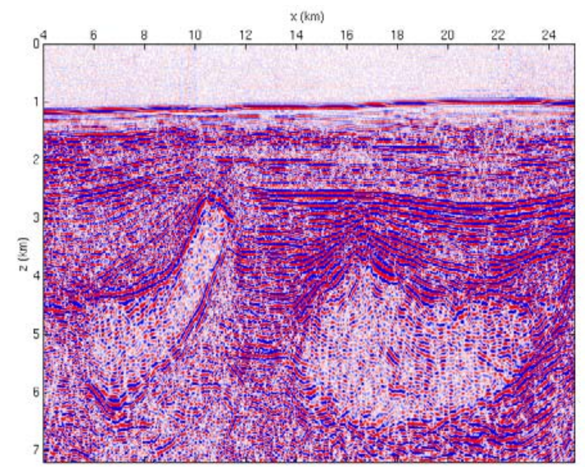
- Introduction
  - Imaging/migration
  - Ray-based approach and its limitations
- Finite difference wave equation migration
  - complexity
  - “one-way” wave equation migration
  - “full” (two-way) wave equation migration in 2D
  - examples
- Conclusions

Introduction: imaging problem

Goal of the migration: given the propagation model of the earth, retrieve the locations and the amplitudes of the reflectors







Introduction: physics

Wave equation

$$-\frac{\omega^2}{v^2}u-\frac{\partial^2 u}{\partial x^2}-\frac{\partial^2 u}{\partial y^2}-\frac{\partial^2 u}{\partial z^2}=f$$

u is the pressure, f is the source

General form of the wave equation: $A(x,v)\ u(x,x_s,v) = f(x,x_s)$

Data: $c(x_s,x_r) = R(x,x_s,x_r)\ u(x,x_s)$

Introduction: mathematics

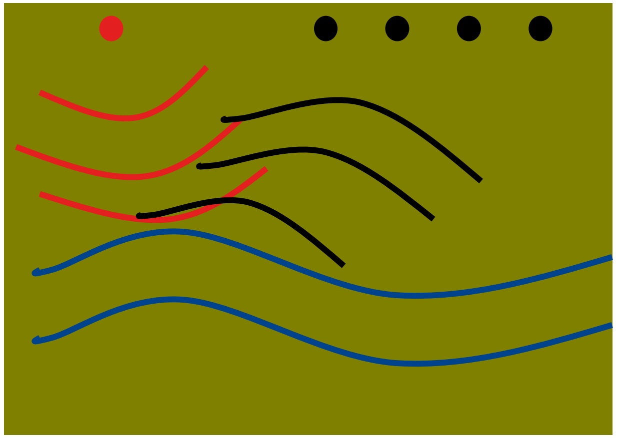
Find *v** such as:

$$J(v^*) = \min J(v) = \frac{1}{2} \sum_{x_s, x_r} \left\| c(x_s, x_r, v) - d(x_s, x_r) \right\|^2$$

Migration-gradient:

$$\begin{cases} A(x,v)\,u(x,x_s,v) = f(x,x_s); \\ c(x_s,x_r,v) = R(x_s,x_r,x)\,u(x,x_s,v) \\ A^*(x,v)\,\lambda(x,x_s,v) = \sum_{x_r} R^*(x_s,x_r,x)\big(c(x_s,x_r,v) - d(x_r,x_s)\big) \\ m(x) = -K(x)\nabla_v J(v) = \sum_{x_s} w(x,x_s)\,\lambda^*(x,x_s,v)\,u(x,x_s,v) \end{cases}$$

Introduction: migration



For all the shot gathers:

1. Compute the incident field from the source
2. Compute the backpropagated field of the shot gather
3. Cross-corrolate the two fields to obtain the shot migrated image

Stack over all the shots the migrated images

Need to solve efficiently the wave equation for a large number of shot-receiver positions and a large enough domain

High frequency solution: ray method

$$c(x_s, x_r) = \sum_x \sum_l A_l(x, x_s, x_r)\, e^{i\omega\tau_l(x, x_s, x_r)}$$

τ obeys the Eikonal equation: $$|\nabla\tau|^2 = \frac{1}{v^2}$$

A obeys the transport equation: $$2\nabla A \cdot \nabla\tau + A\Delta\tau = 0$$

- Efficiently solved with a wavefront construction algorithm
- The approximation reaches its limits with complex earth structure (multipathing, irregular boundaries, …)
- Need to go back to the finite-difference solution

Outline

- Introduction
  - Imaging/migration
  - Ray-based approach and its limitations
- Finite difference wave equation migration
  - complexity
  - “one-way” wave equation migration
  - “full” (two-way) wave equation migration in 2D
  - examples
- Conclusions

Forward modeling

Time domain

$$\frac{1}{v^2}\frac{\partial^2 u}{\partial t^2}-\frac{\partial^2 u}{\partial x^2}-\frac{\partial^2 u}{\partial y^2}-\frac{\partial^2 u}{\partial z^2}=f$$

Complexity:

2D: $n_t O(n^2)$

3D: $n_t O(n^3)$

Frequency domain

$$-\frac{\omega^2}{v^2}u-\frac{\partial^2 u}{\partial x^2}-\frac{\partial^2 u}{\partial y^2}-\frac{\partial^2 u}{\partial z^2}=f$$

Complexity (direct solver)

2D: $n_\omega O(n^3)+ n_\omega O(n^2 \log(n))$

2D: $n_\omega O(n^6)+ n_\omega O(n^4 \log(n))$

Time domain is more appropriate for the modeling of one shot and it is achievable in 3D because it is parallelizable

migration

Time domain

$$\frac{1}{v^2}\frac{\partial^2 u}{\partial t^2}-\frac{\partial^2 u}{\partial x^2}-\frac{\partial^2 u}{\partial y^2}-\frac{\partial^2 u}{\partial z^2}=f$$

Complexity:

2D: $n_s n_t O(n^2)$

3D: $n_s n_t O(n^3)$

Frequency domain

$$-\frac{\omega^2}{v^2}u-\frac{\partial^2 u}{\partial x^2}-\frac{\partial^2 u}{\partial y^2}-\frac{\partial^2 u}{\partial z^2}=f$$

Complexity (direct solver)

2D: $n_\omega O(n^3)+ n_s n_\omega O(n^2 \log(n))$

2D: $n_\omega O(n^6)+ n_s n_\omega O(n^4 \log(n))$

$n_t = O(n)$ but $n_w = O(1)$

In 2D: the frequency domain is preferable when n_s is large

In 3D: not yet achievable in time domain; impossible in frequency domain with direct solver

“one-way” finite-difference migration

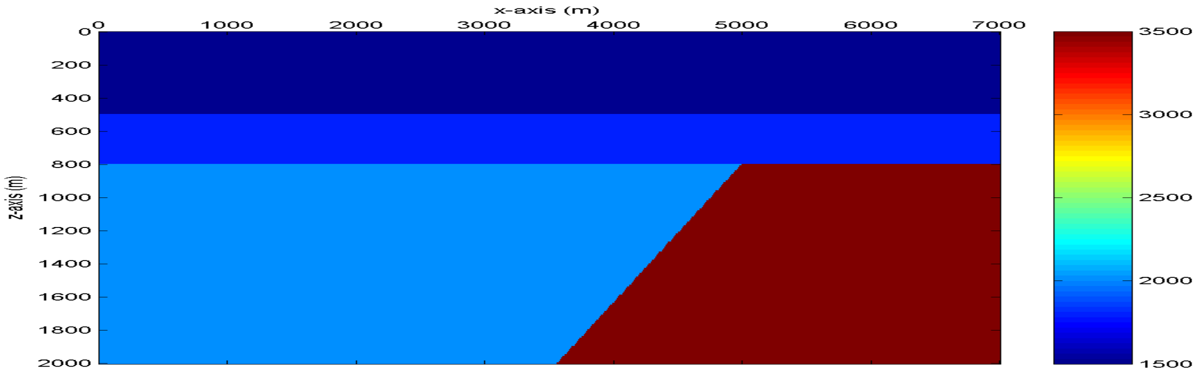
- Paraxial approximation of the wave equation
  - Choice of a preferred direction, z-direction
  - Assume not too large lateral variation
  - Assume not too wide angle propagation from the preferred direction
- Marching approach: $$i\frac{\partial}{\partial z} = \sqrt{k^2 + \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}}$$
- Complexity in 2D: $n_s n_\omega O(n^2)$
- Complexity in 3D: $n_s n_\omega O(n^3)$
- Feasible even in 3D

Synthetic example no. 1
dipped interface

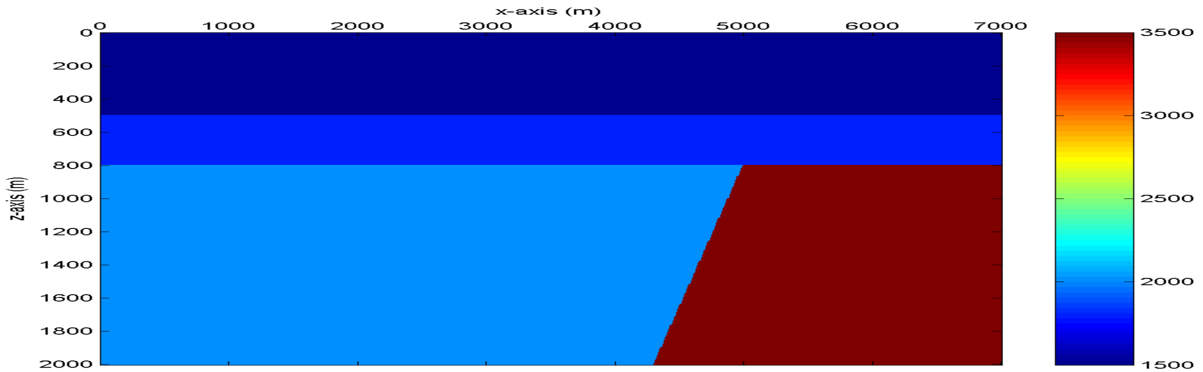
- Interface dip: 40°, 60°, and 80°
- Synthetic data, marine type acquisition, cable length 2 km
- Frequencies from 8 to 20 Hz
- One-way migration scheme with 70° Padé approximation

dip angle models

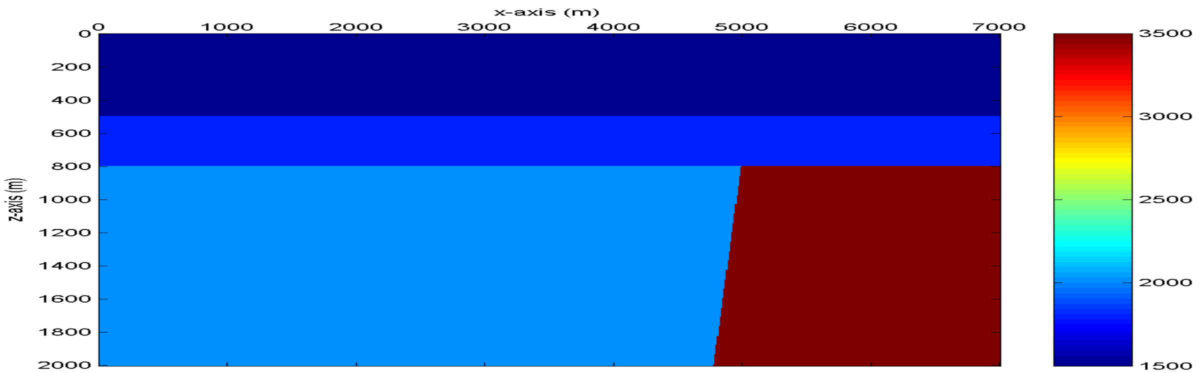
40° dip



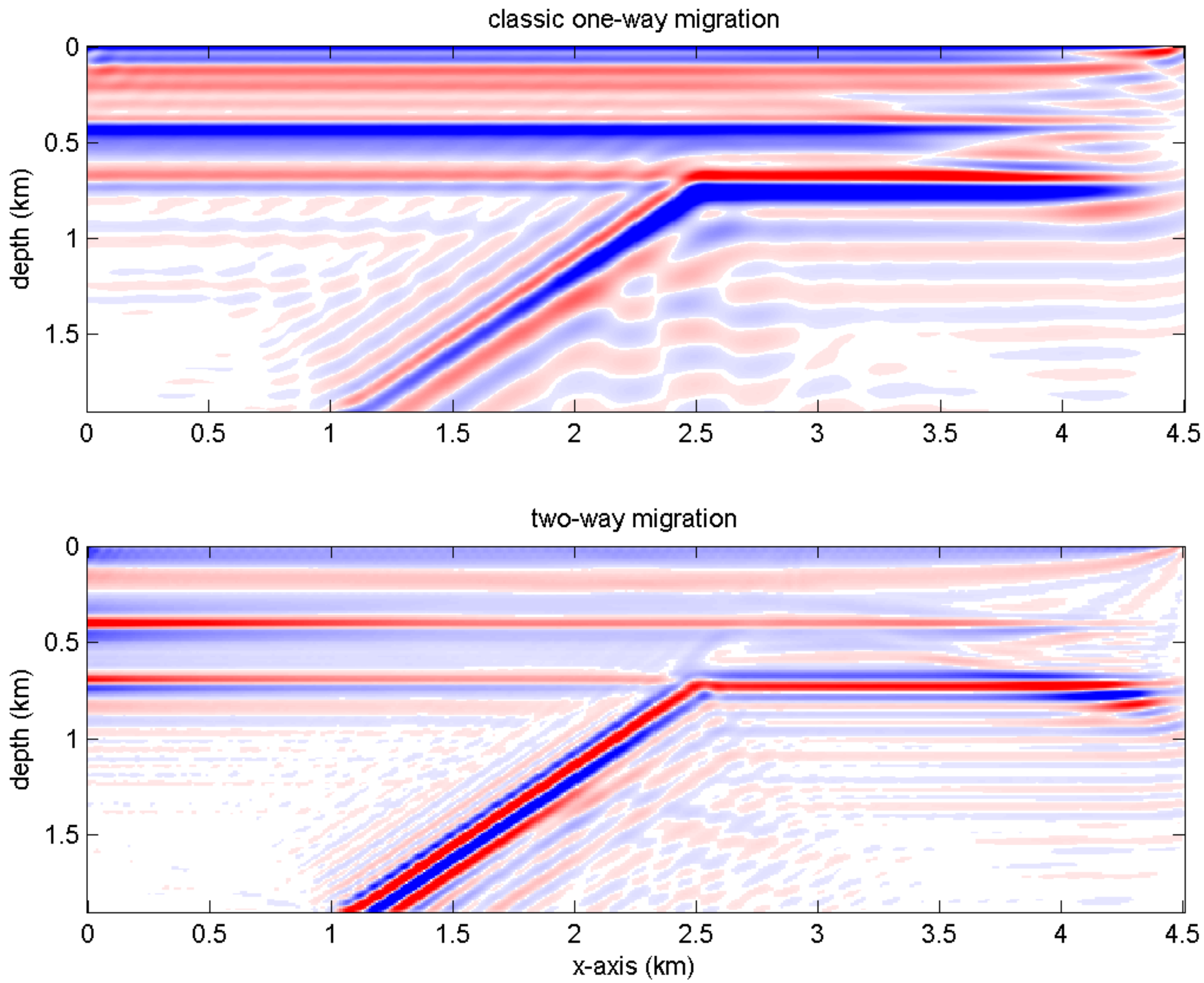
60° dip



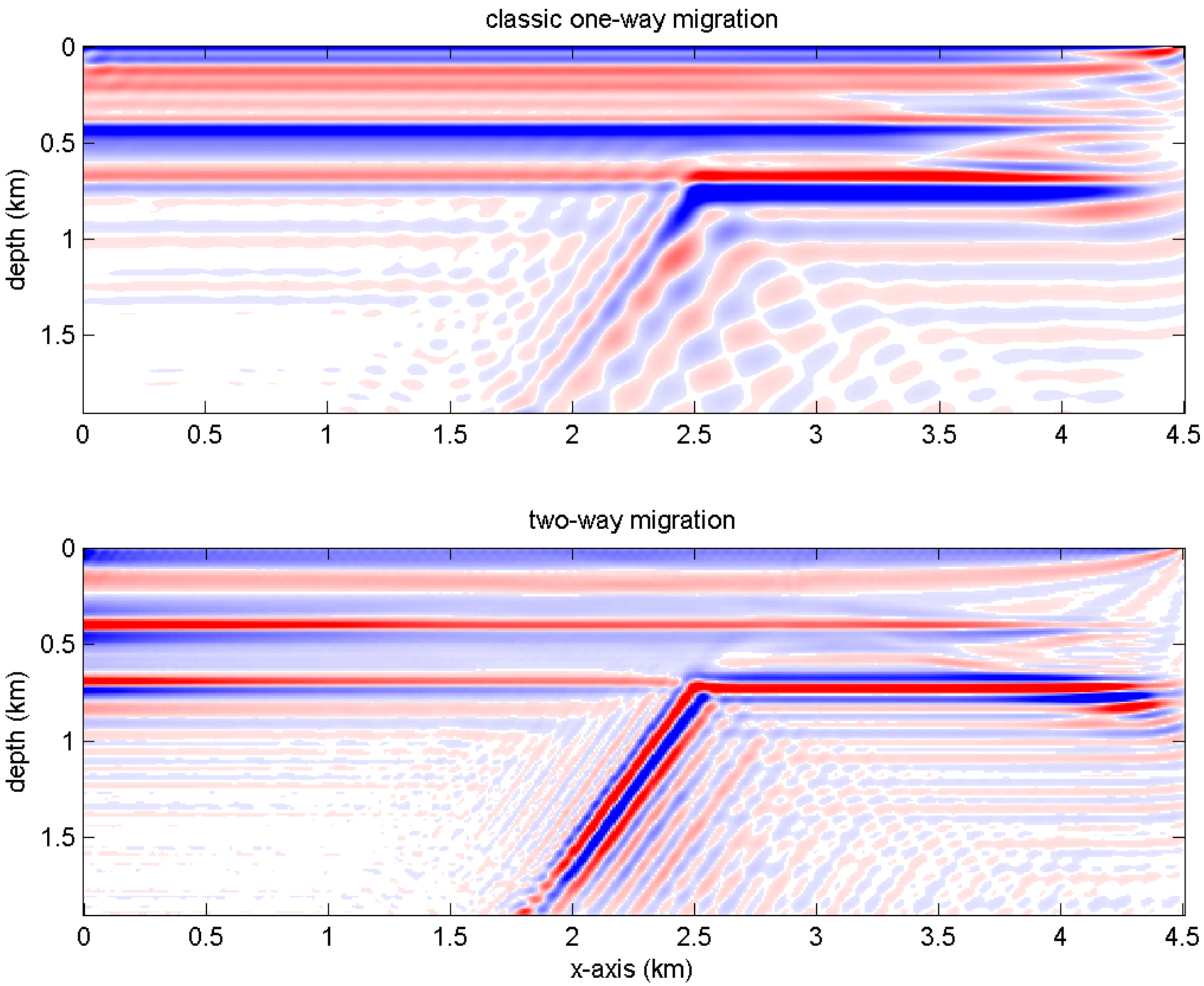
80° dip



40° dip angle

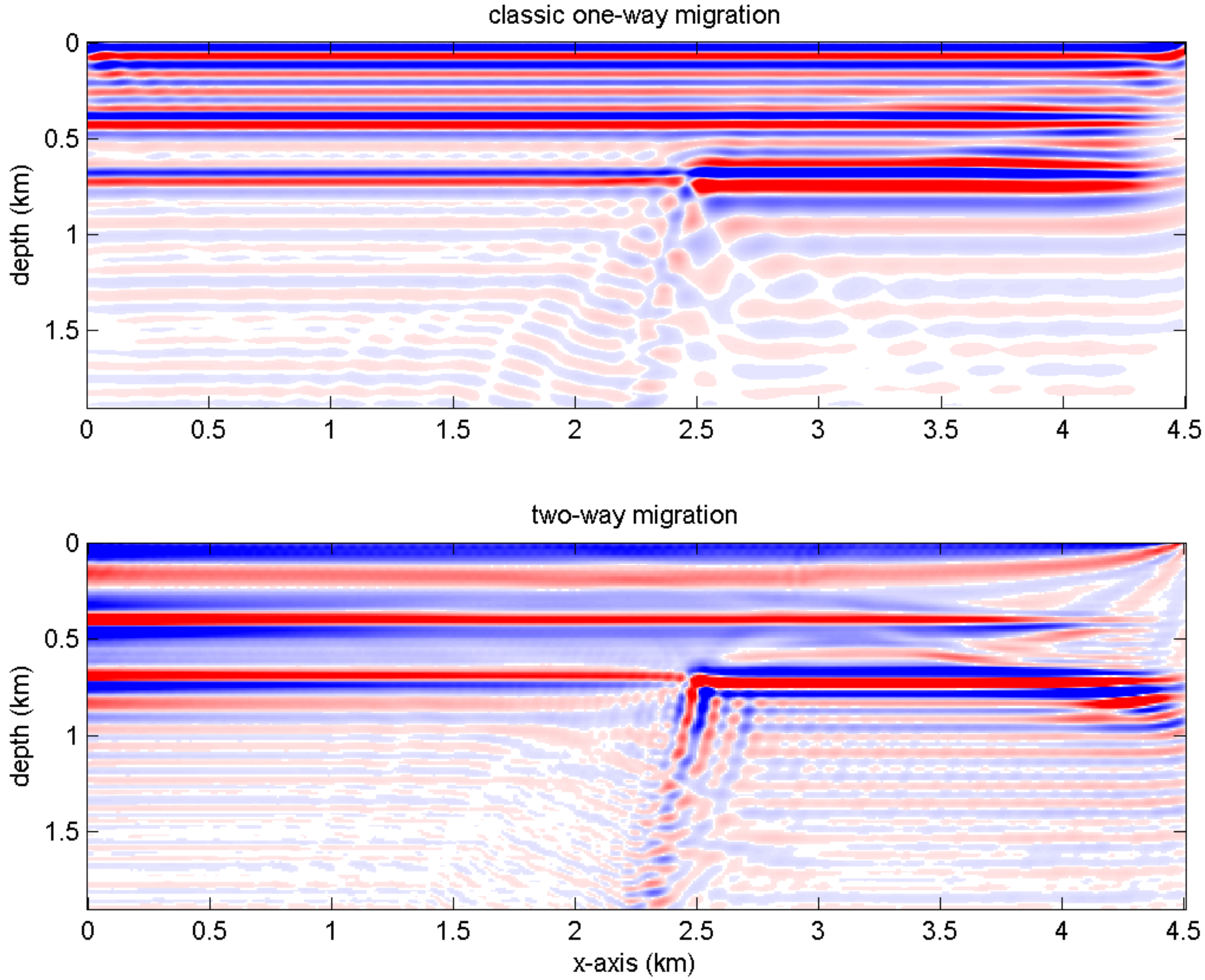

classic one-way migration
depth (km)
0
0.5
1
1.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
two-way migration
depth (km)
0
0.5
1
1.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
x-axis (km)

60° dip angle



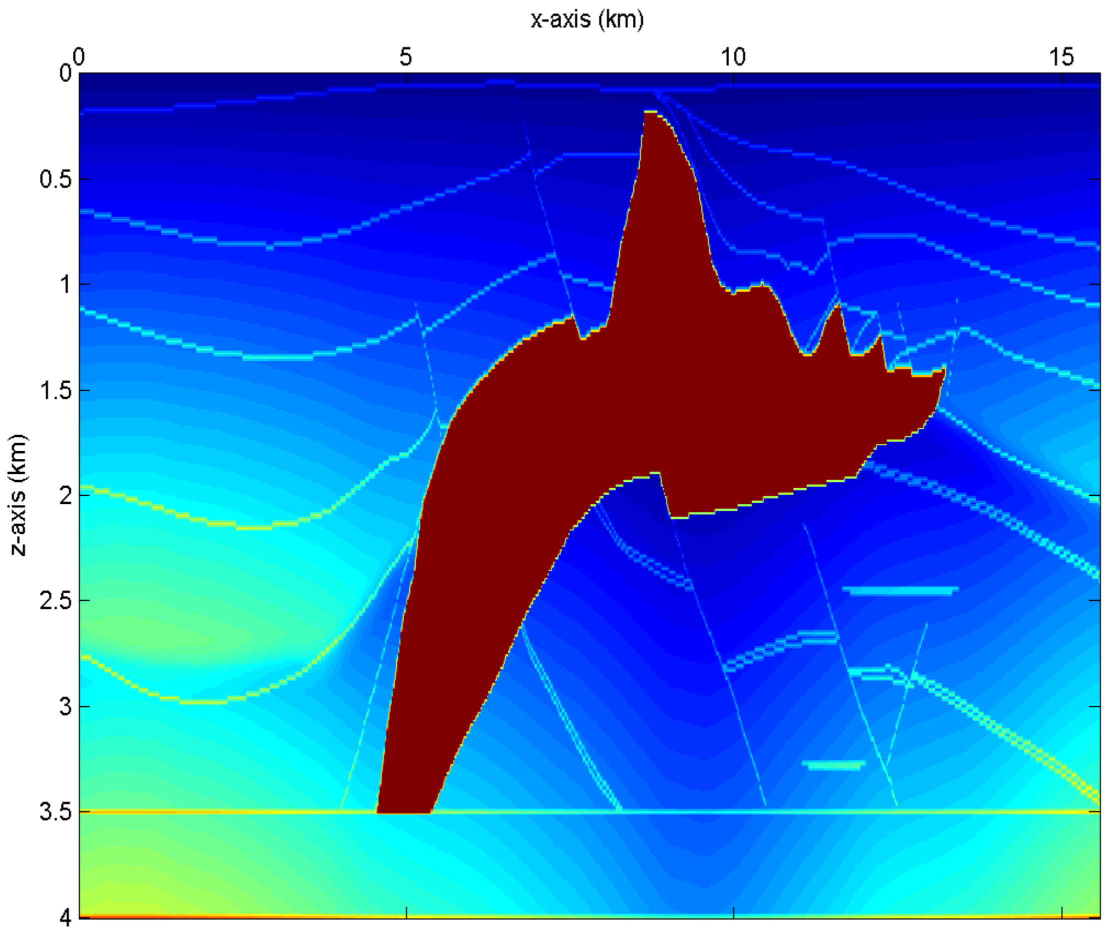
classic one-way migration
depth (km)
0
0.5
1
1.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
two-way migration
depth (km)
0
0.5
1
1.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
x-axis (km)

80° dip angle


classic one-way migration
depth (km)
0
0.5
1
1.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
two-way migration
depth (km)
0
0.5
1
1.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
x-axis (km)

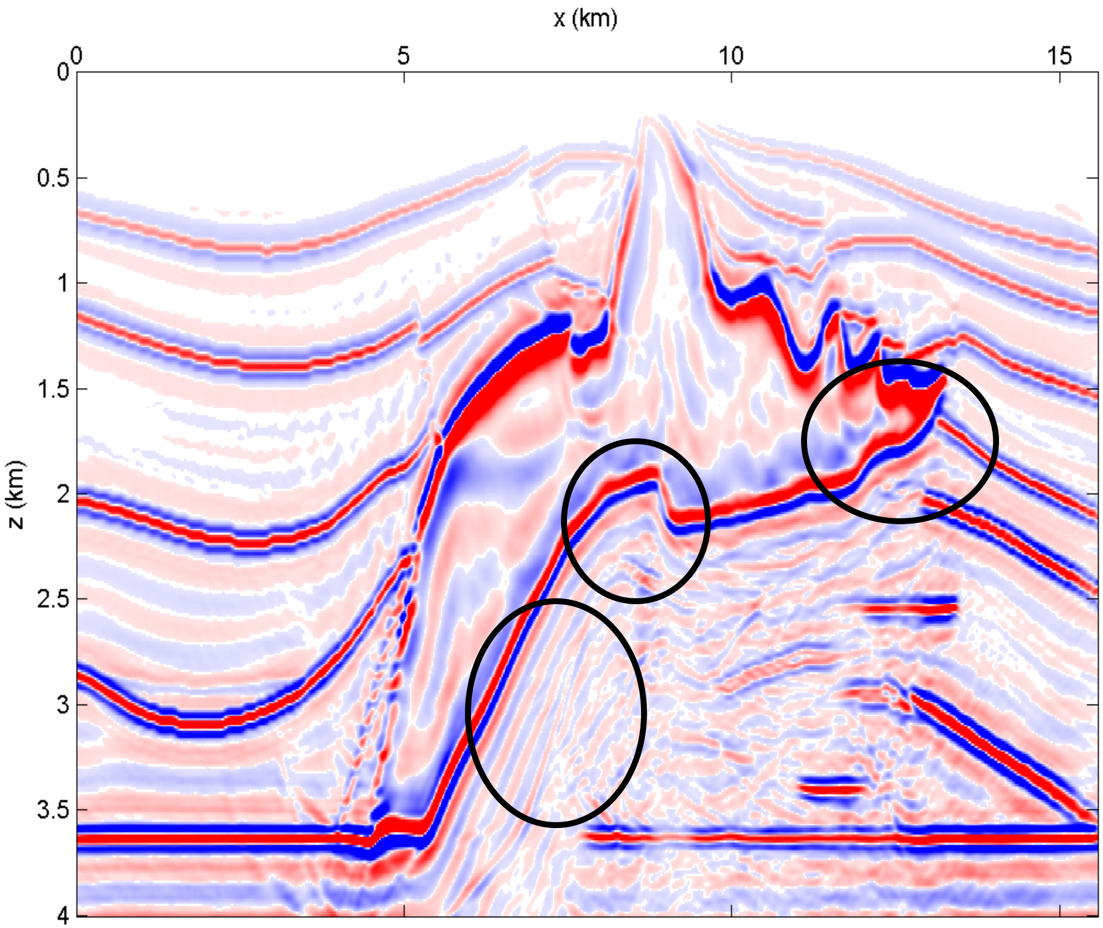
Synthetic example no. 2: SEG/EAGE salt model

Data reshot in 2D with a time domain method



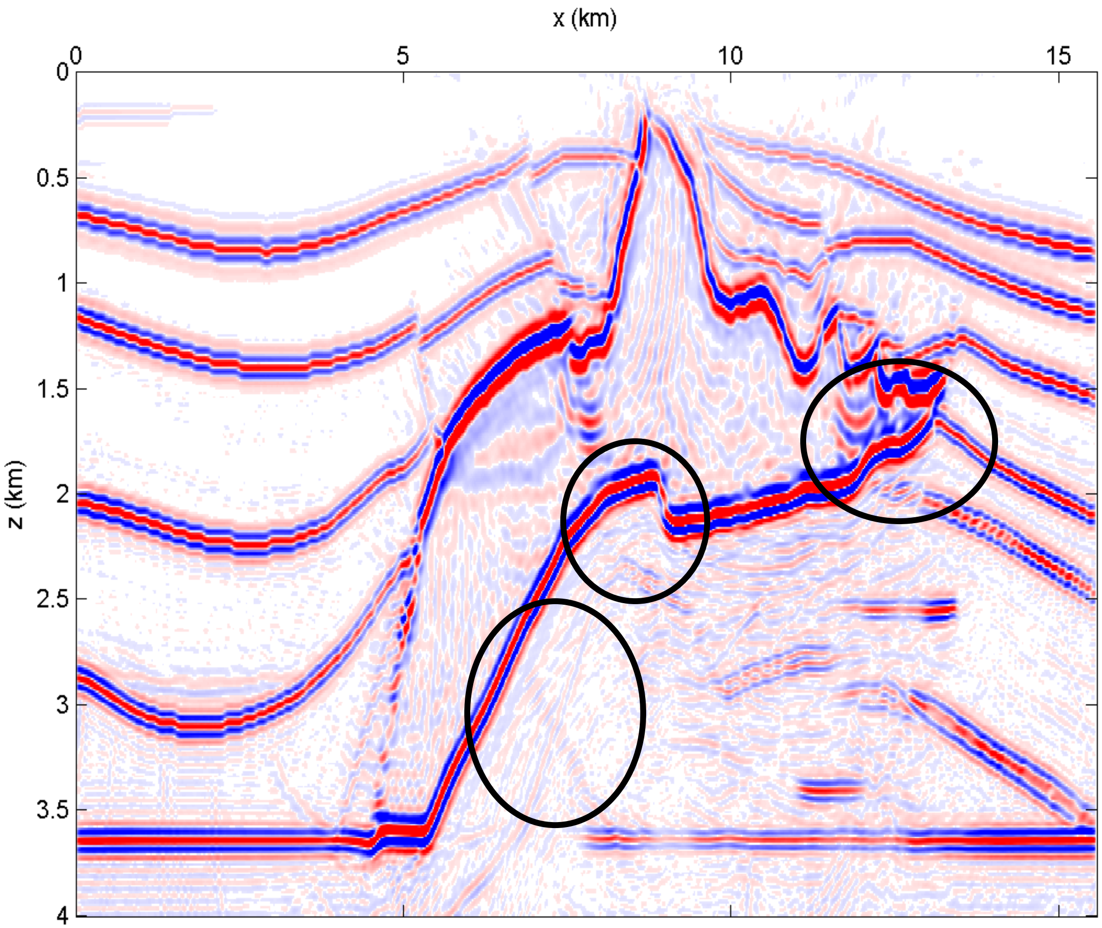
velocity

SEG/EAGE salt model



one-way wave-equation

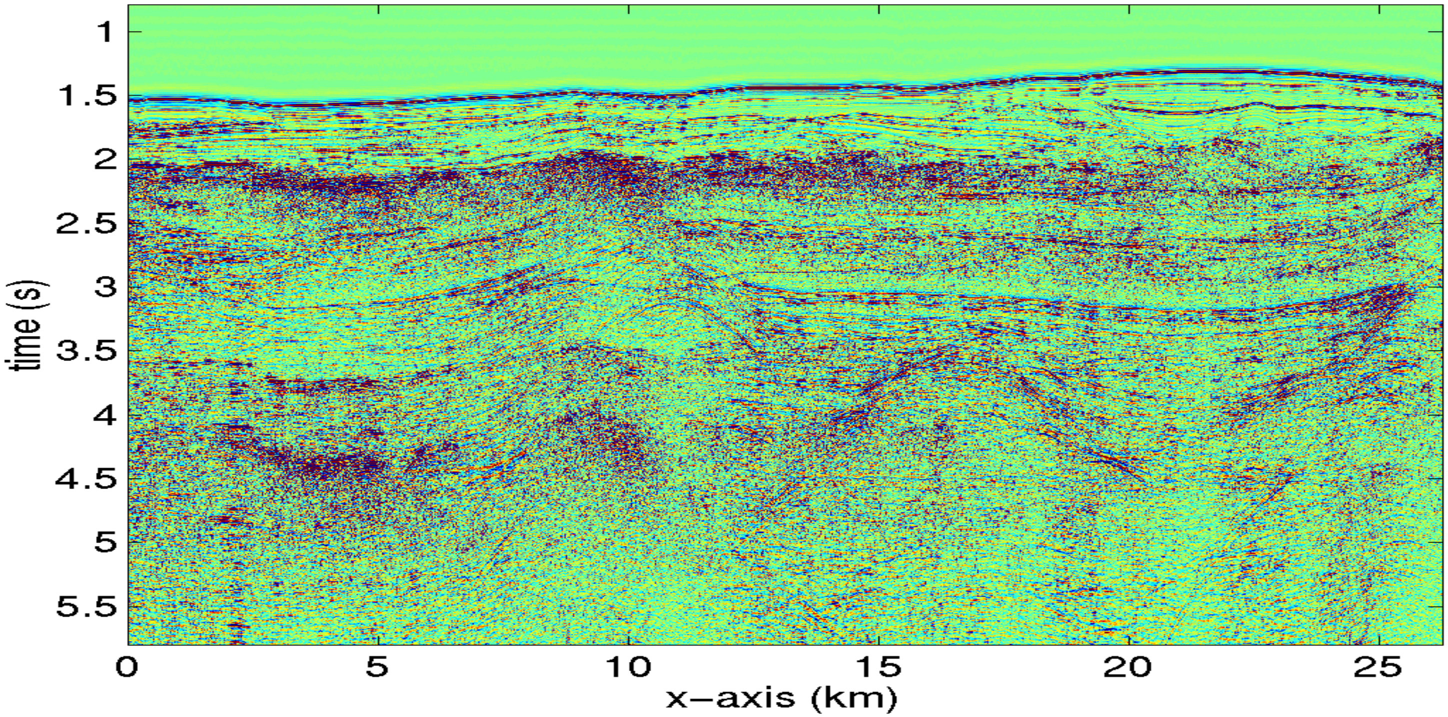
SEG/EAGE salt model



two-way wave-equation (high-pass filtered)

Golf of Mexico data set

Near offset traces

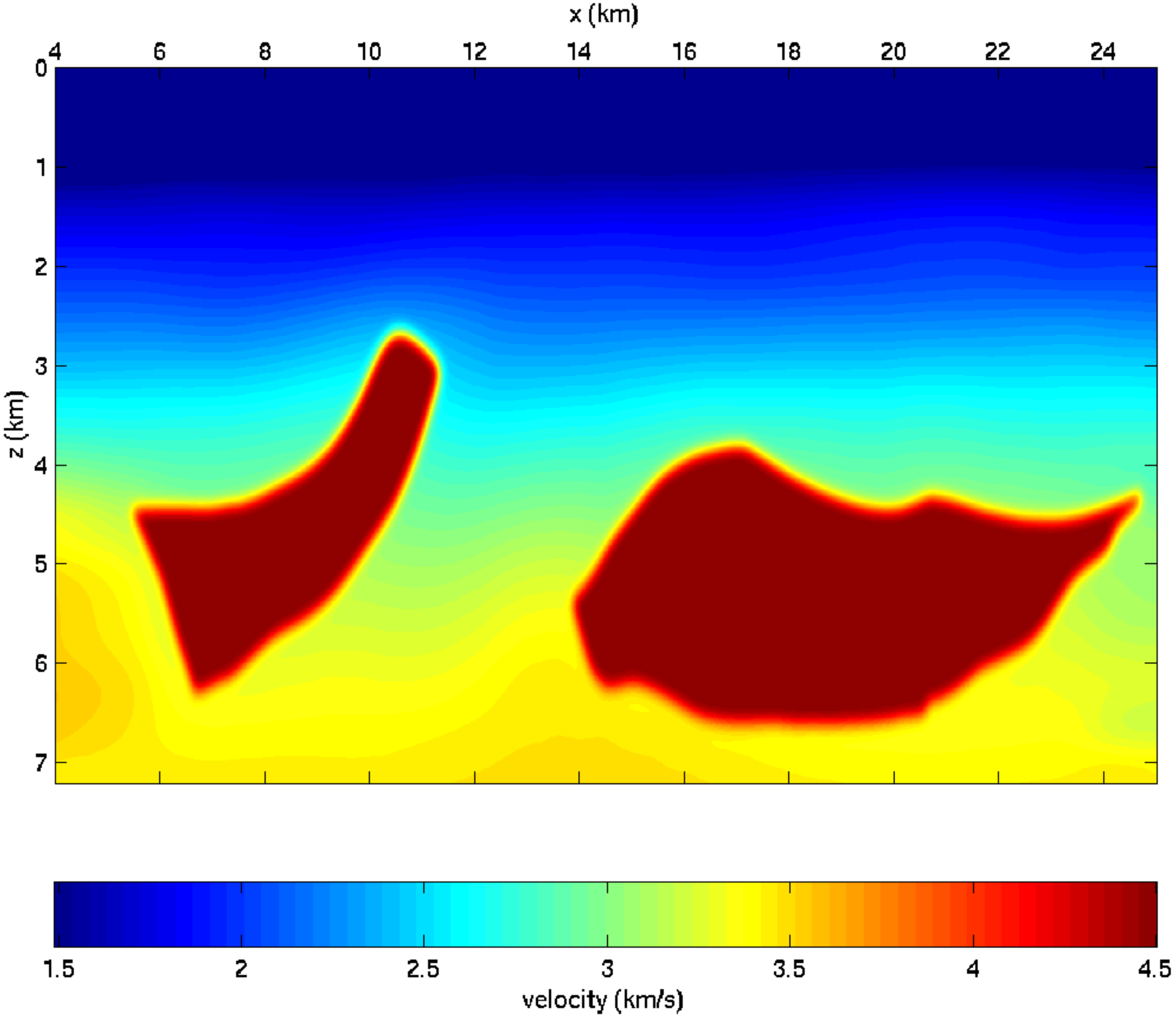


1055 shots of 320 traces, largest offset: ~8 km

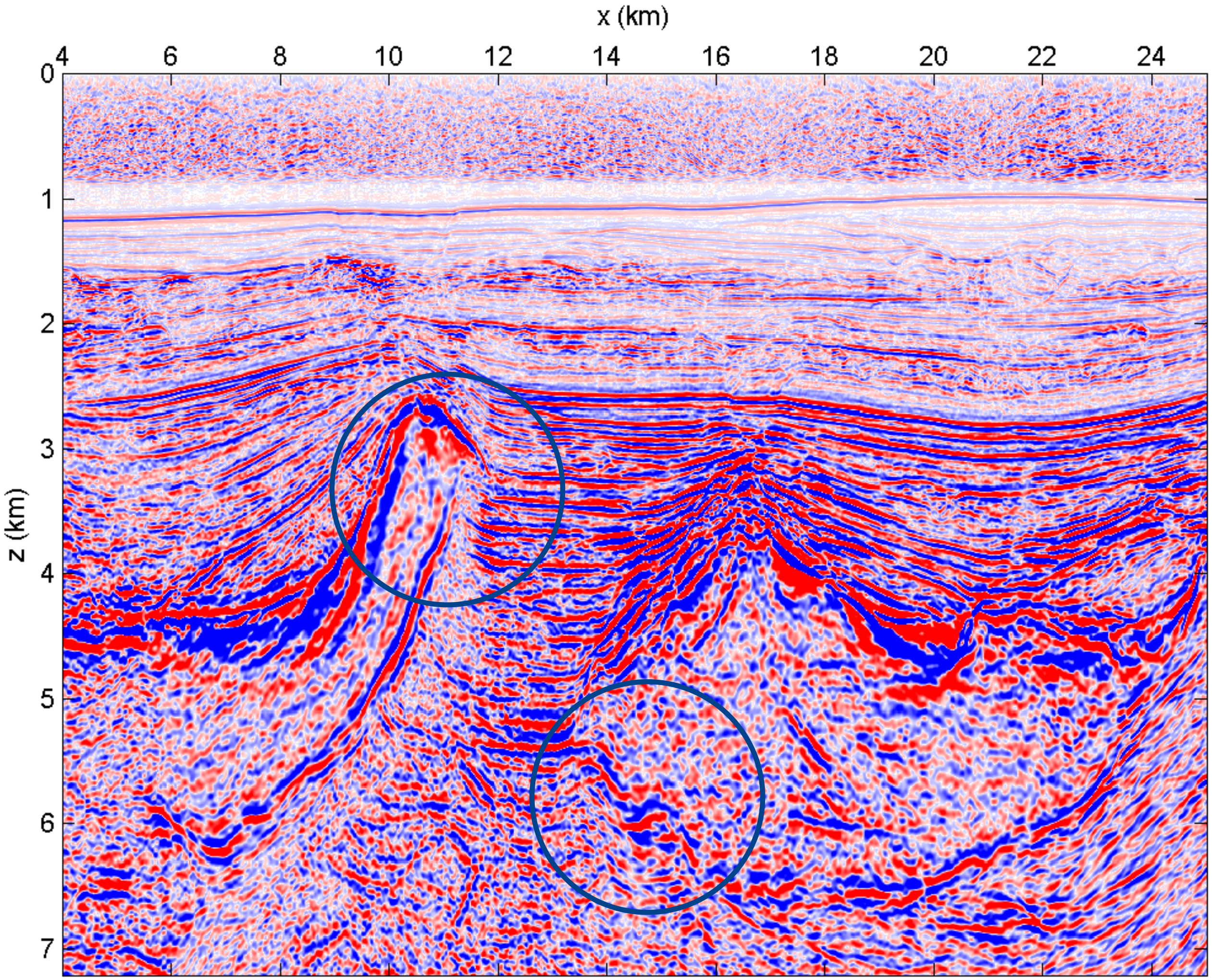
For the processing, the data have been divided in 5 sets

For each subset, the model contained 1829 by 881 points with a spancing of 10 m (this leads to a sparse matrix of $1.6\ 10^6$ by $1.6\ 10^6$).

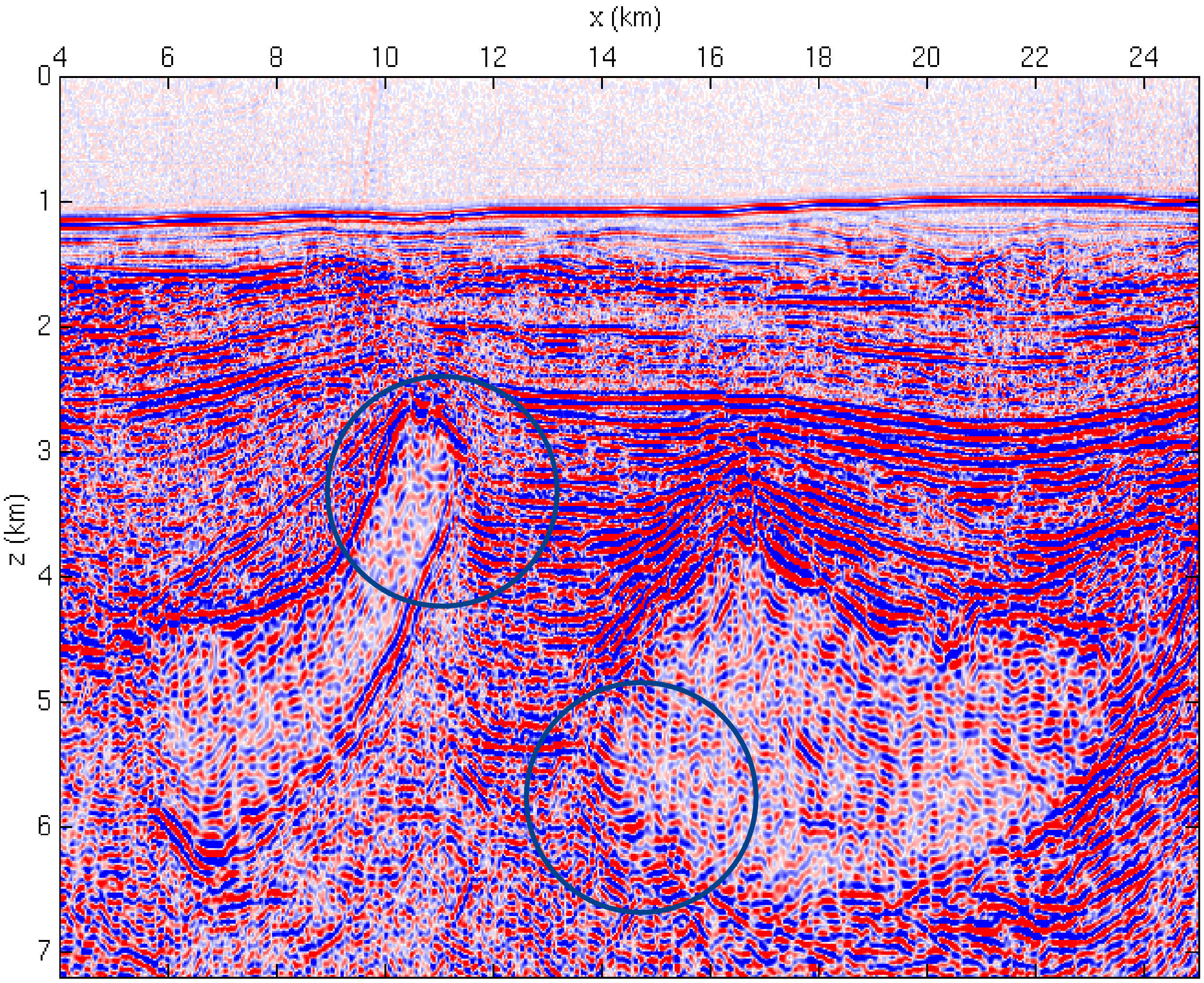
Velocity model



One-way, 70°



Two-way



Conclusions

- One-way wave-equation migration is less accurate at steep dips and amplitudes.
- In 2D, the two-way wave-equation migration is not much more expensive than one-way migration. In 3D, this is not true.
- In 3D, the one-way wave-equation migration is the only affordable solution with finite-difference type of migration

Conclusions

- The 3D time domain “two-way” wave equation migration requires a peta flop computer

- The challenge:
  - A 3D iterative Helmholtz solver faster than $n_s O(n^4)$
  - Can we process simultaneous 100’s of right hand sides ?
  - What is the memory requirement when n=1000 ?
  - How efficient would the parallelization be ?