High-resolution fast microseismic source collocation and source-time function estimation

Shashin Sharan, Rongrong Wang and Felix J. Herrmann

SEG 2017

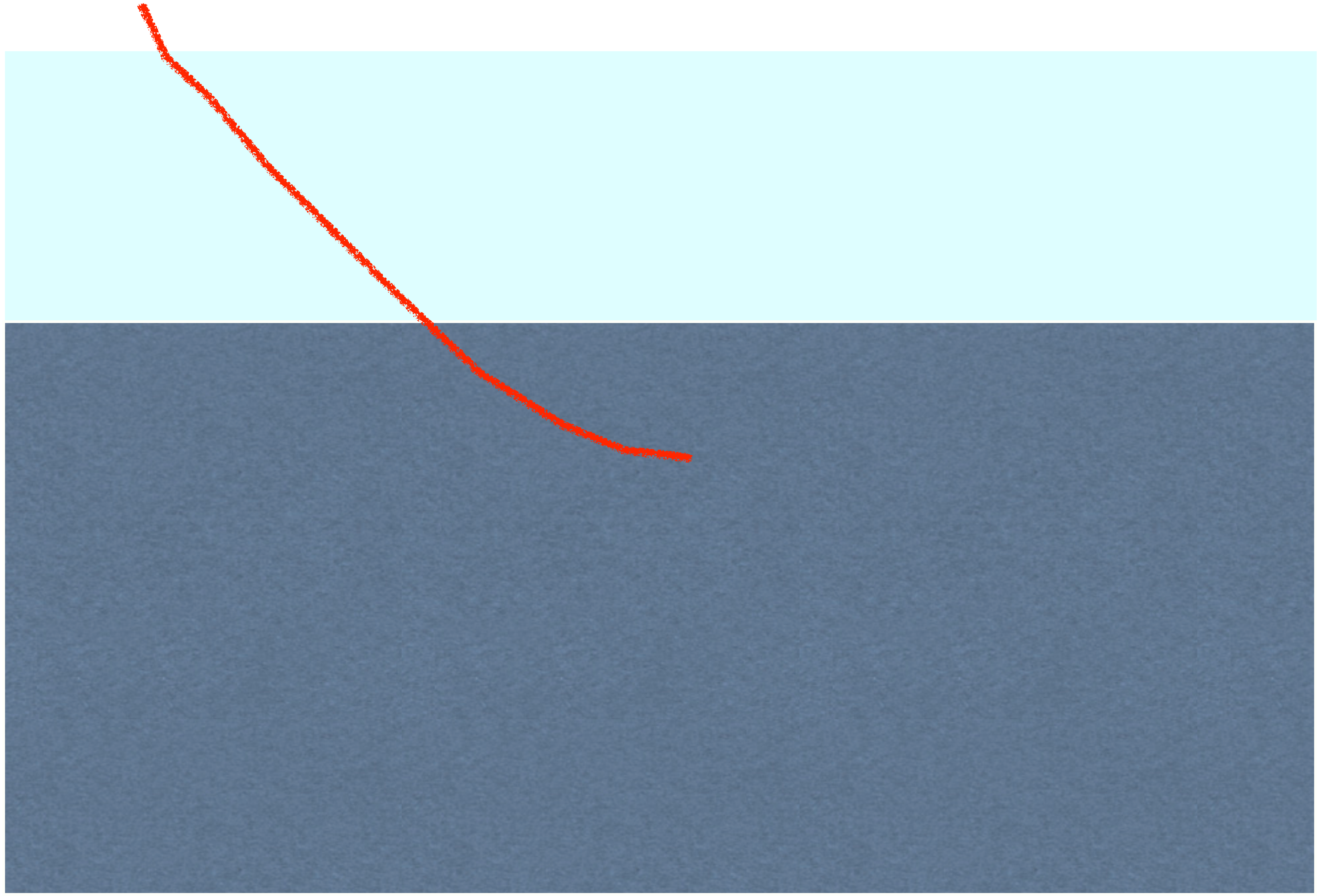


SLIM
University of British Columbia

Motivation

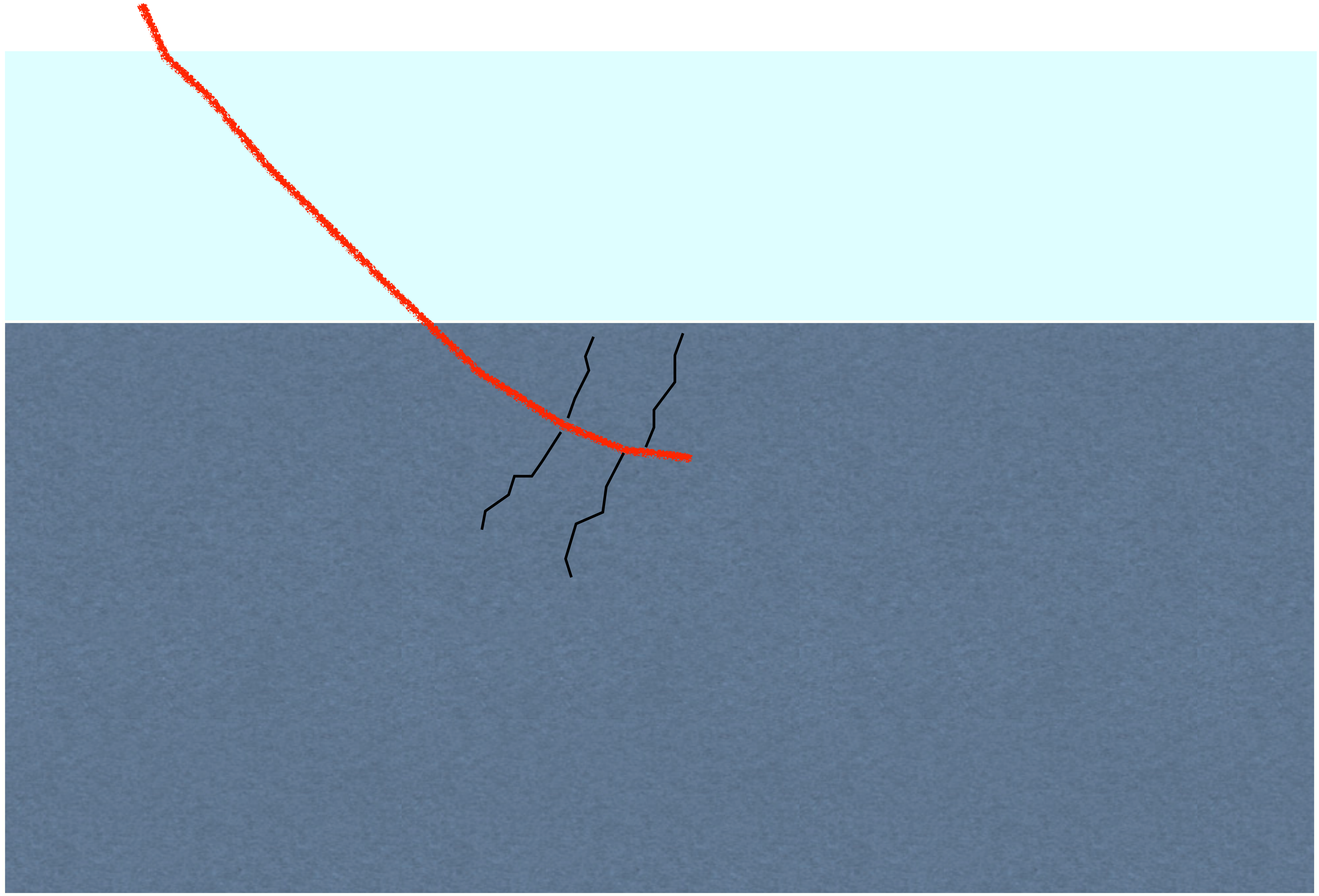
Unconventional Reservoir Schematic

Motivation



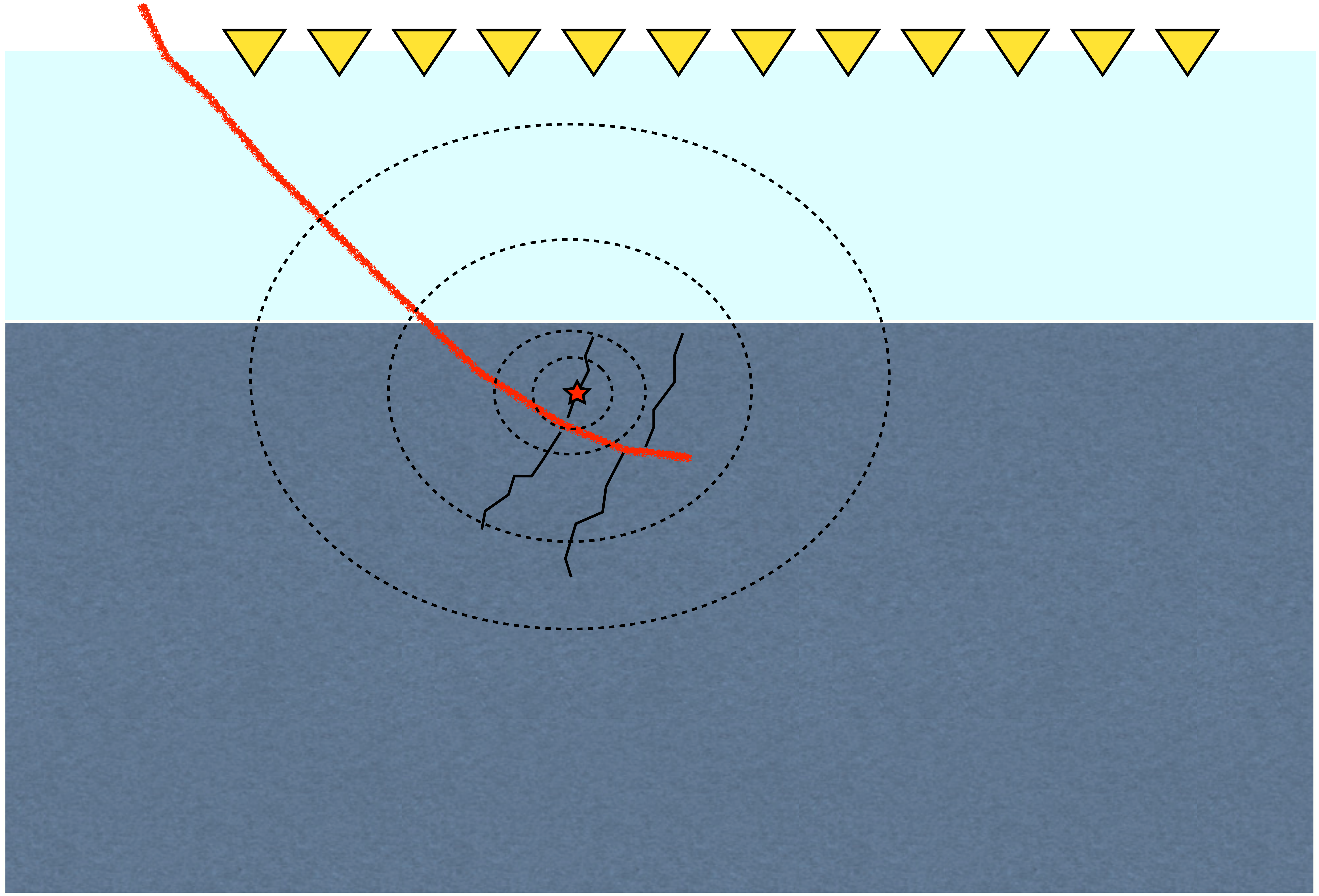
Unconventional Reservoir Schematic

Motivation



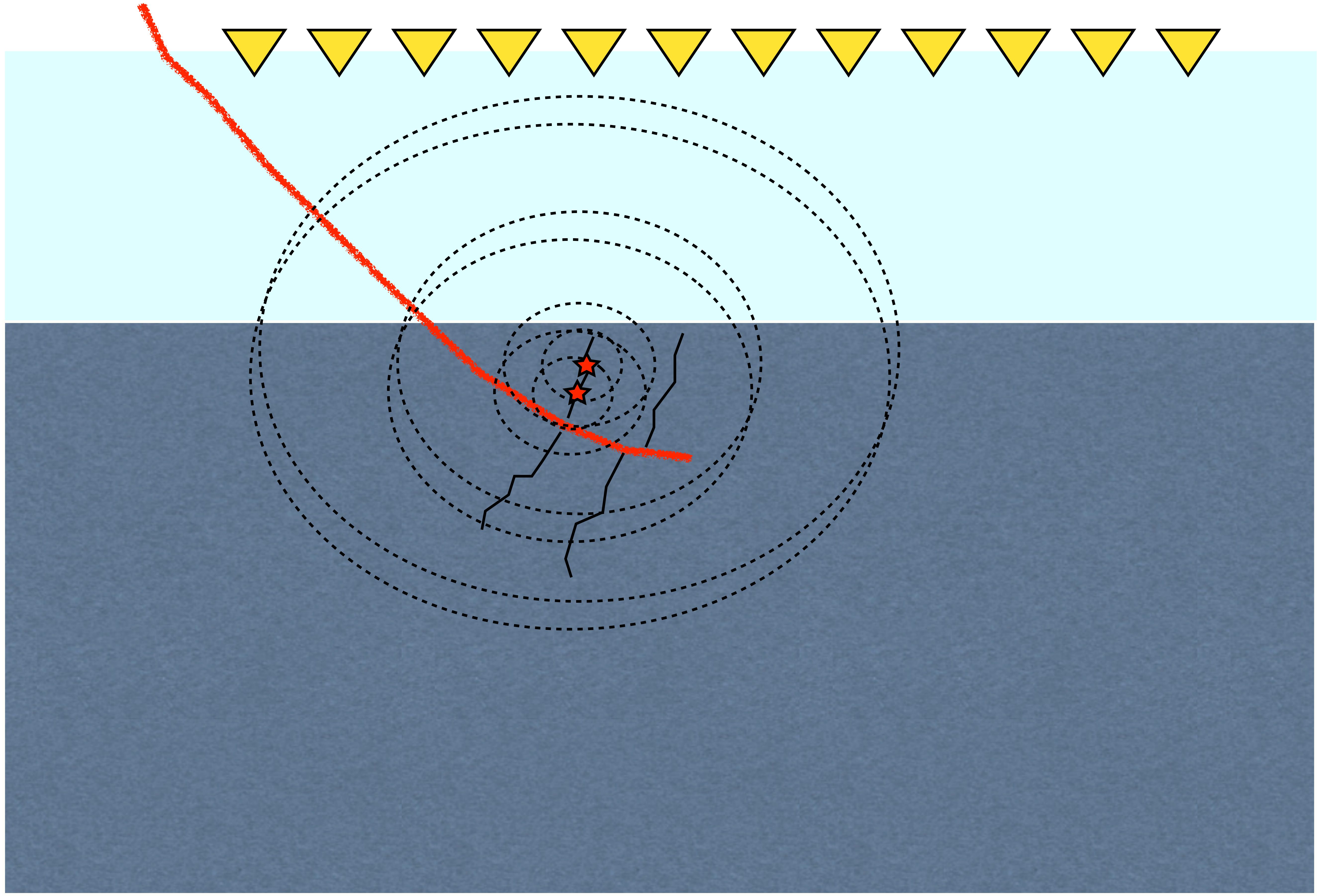
Unconventional
Reservoir Schematic

Motivation



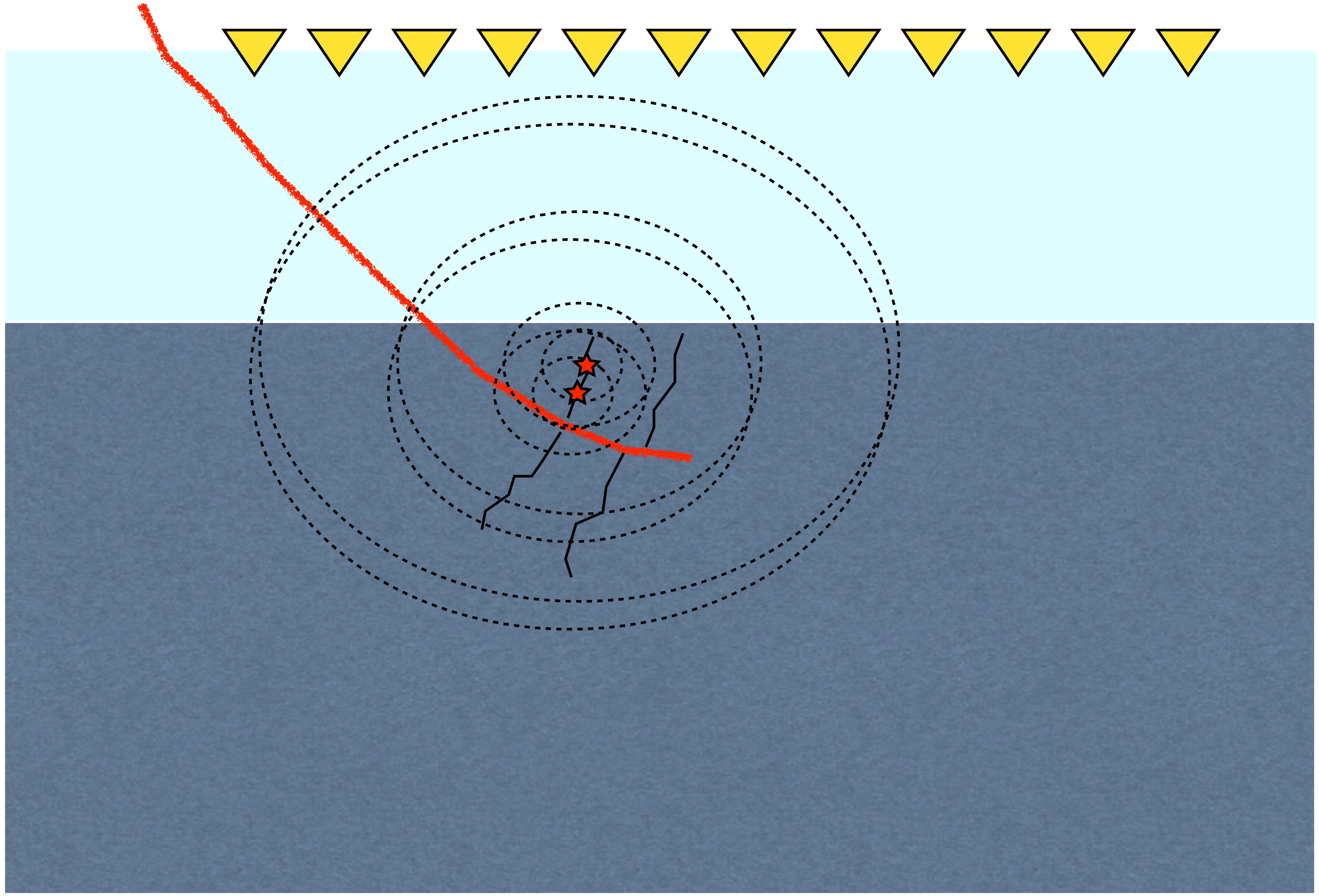
Unconventional Reservoir Schematic

Motivation



Unconventional Reservoir Schematic

Motivation



Unconventional Reservoir Schematic

Objectives

- detection of microseismic events in space and time
- estimation of source time function

Motivation

Accurate fracture detection

- by locating closely spaced point microseismic sources along a fracture
  - within half a dominant wavelength

Estimation of fracture evolution in time

- by estimating the source-time function of point microseismic sources
  - without any assumptions on the shape of source-time function
  - source-time functions can be used for estimating the source mechanism

[Rentsch et al., ’07; McMechan, ’82; Gajewski et al., ’05; Nakata et al.,’16; Bazargani et al.,’16]
[Thurber et al., ’00; Waldhauser et al.,’00]

SLIM

Pre-existing methods

Arrival time picking based methods:

- estimate the location and origin time
- can be challenging in the presence of noise

Imaging based methods:

- do not require arrival time picking
- based on back propagation
- estimate the location and origin time
- require scanning of complete 4D volume (3D in space and 1D in time)

[Sjögreen et al.,'14; Wu et al.,'96; Kim et al.,'11; Michel et al.,'14; Kaderli et al.,'15]
[Rodriguez et al.,'12; Ely et al., '13]

Pre-existing methods

Dictionary learning based methods:

- simultaneously estimate location, origin time and source mechanism
- require forming large dictionaries based on
  - number of sources, number of receivers and number of time samples
- require prior knowledge of source-time function

Full-waveform inversion (FWI) based methods:

- invert for source parameters
- some of these methods assume
  - prior knowledge of source-time function
  - source-time function to be a gaussian function

[Sharan et al.,'16]

Proposed method w/ sparsity promotion

Estimates complete source field in:

- space and
- time

[Sharan et al.,’16]

Proposed method w/ sparsity promotion

Estimates complete source field in:

- space and
- time

No assumptions on:

- shape of source-time function
- prior knowledge of source-time function

[Sharan et al.,'16]

SLIM

Proposed method w/ sparsity promotion

Estimates complete source field in:

- space and
- time

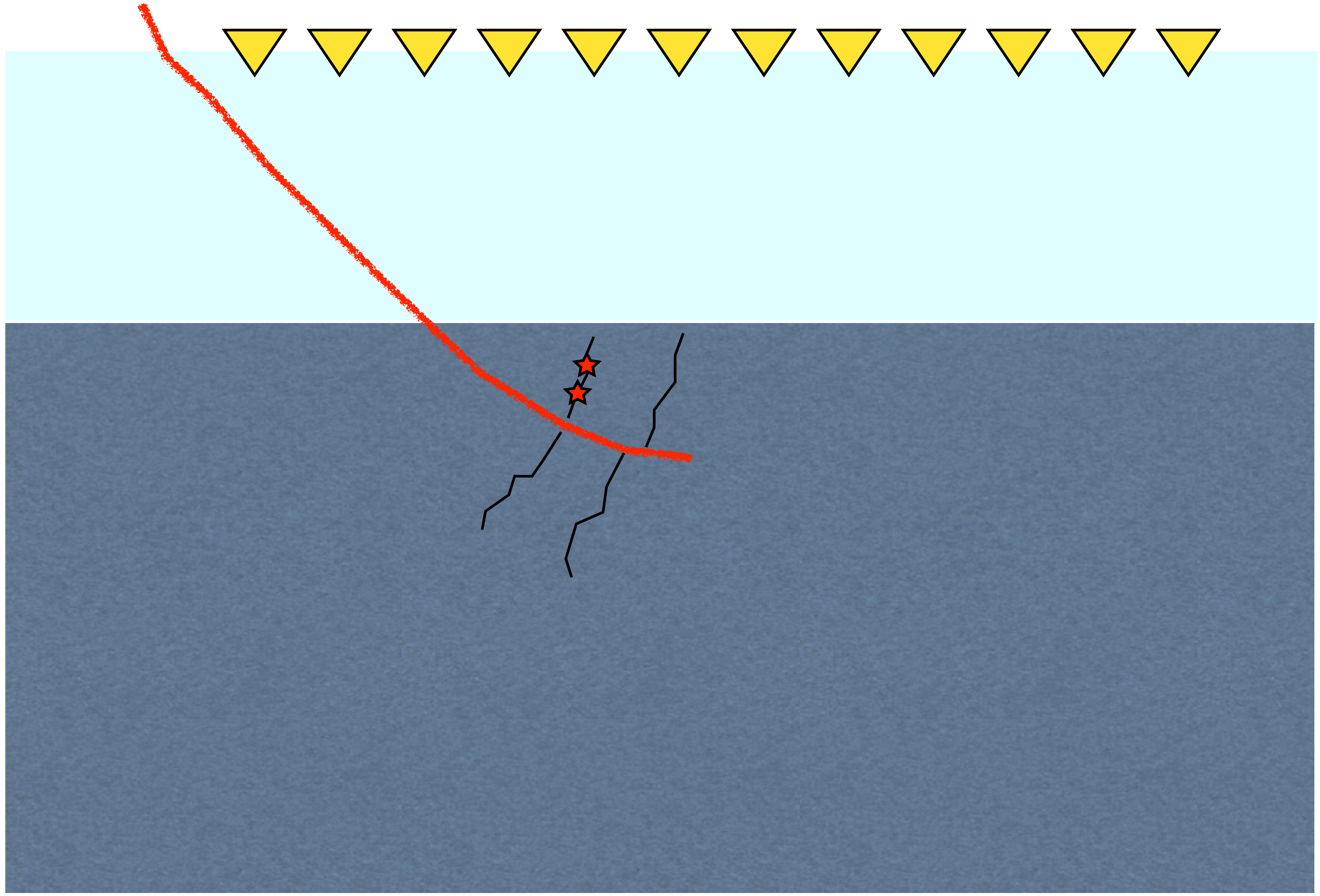
No assumptions on:

- shape of source-time function
- prior knowledge of source-time function

Needs:

- sufficiently accurate medium velocity model
- position of receivers

Proposed method w/ sparsity promotion

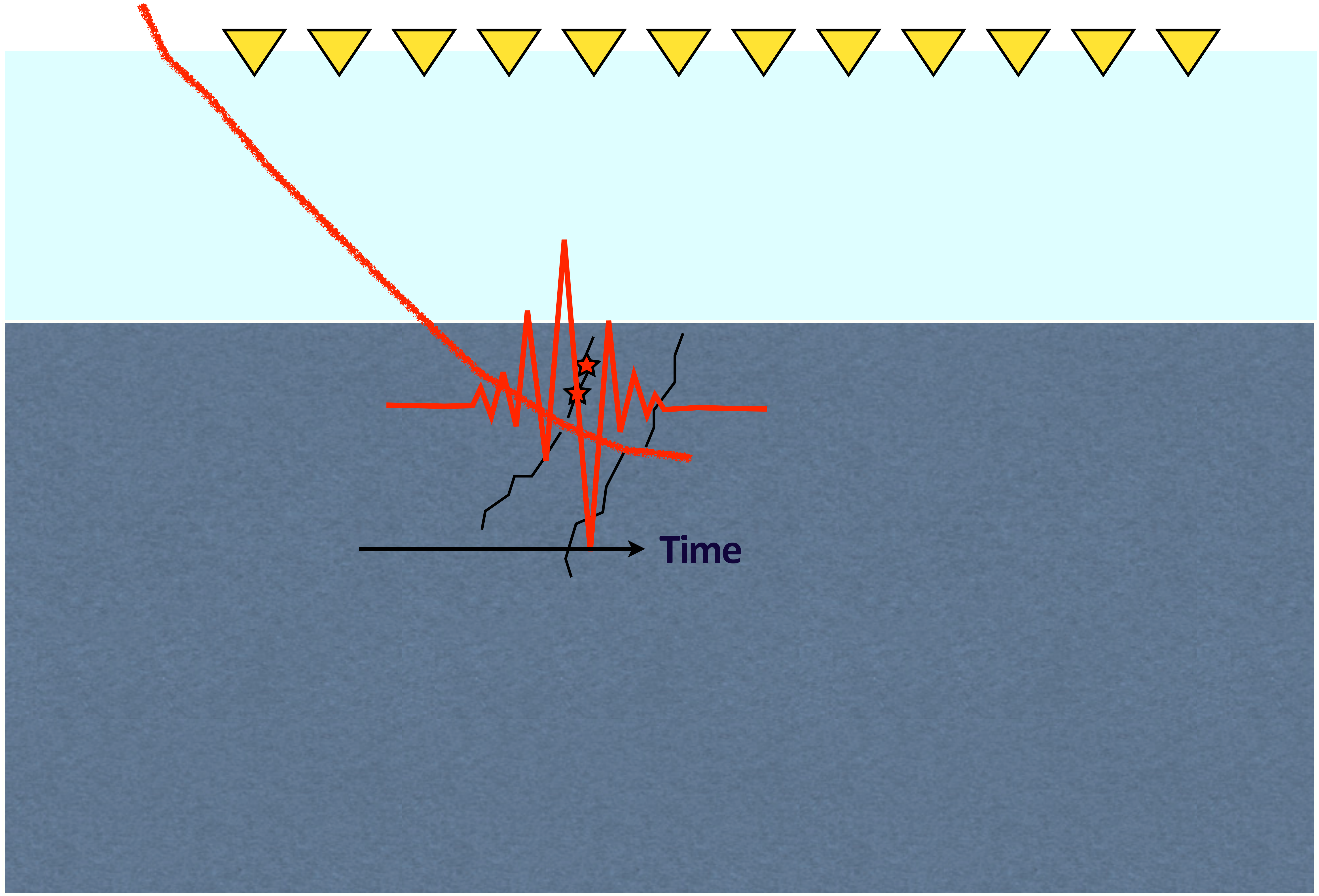


Unconventional Reservoir Schematic

Assumptions

- localized in space

Proposed method w/ sparsity promotion



Unconventional Reservoir Schematic

Assumptions

- localized in space
- finite energy along time

[Van Den Berg et al.,'08]

SLIM

Proposed method w/ sparsity promotion

Source field **Noise level**

$$\underset{\mathbf{Q}}{\text{minimize}} \ \|\mathbf{Q}\|_{2,1}$$
$$\text{subject to } \|\mathcal{F}[\mathbf{m}](\mathbf{Q}) - \mathbf{d}\|_2 \leq \epsilon$$

Forward modeling operator **Slowness square** **Observed data**

$\mathbf{Q} \in \mathbb{R}^{n_x \times n_t}$

n_x: number of grid points
n_t: number of time samples

Similar to classic Basis pursuit denoising (BPDN)

[Van Den Berg et al.,'08]

SLIM

Proposed method w/ sparsity promotion

Source field

Noise level

$$\underset{\mathbf{Q}}{\text{minimize}} \quad \|\mathbf{Q}\|_{2,1}$$

$$\text{subject to } \|\mathcal{F}[\mathbf{m}](\mathbf{Q}) - \mathbf{d}\|_2 \leq \epsilon$$

Forward modeling operator

Slowness square

Observed data

$\mathbf{Q} \in \mathbb{R}^{n_x \times n_t}$

n_x: number of grid points

n_t: number of time samples

[Lorentz et al.,'14; Herrmann et al.,'15; Sharan et al.,'16]

Solving w/ Linearized Bregman

$$\underset{\mathbf{Q}}{\text{minimize}} \;\; \|\mathbf{Q}\|_{2,1} + \frac{1}{2\mu}\|\mathbf{Q}\|_F^2$$

$$\text{subject to } \|\mathcal{F}[\mathbf{m}](\mathbf{Q}) - \mathbf{d}\|_2 \leq \epsilon$$

*where $\|.\|_F$ is the Frobenius norm

- Recent successful application to seismic imaging problem
- Three-step algorithm simple to implement
- Choice of μ controls the trade off between sparsity and the Frobenius norm
- $\mu \uparrow \infty$ corresponds to solving original BPDN problem

[Lorentz et al.,'14; Combettes et al.,'11]

Linearized Bregman algorithm

1. **Data d, slowness square m** //Input
2. **for** $k = 0, 1, \cdots$
3. $\mathbf{V}_k = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d}))$ //adjoint solve
4. $\mathbf{Z}_{k+1} = \mathbf{Z}_k - t_k \mathbf{V}_k$ //auxiliary variable update
5. $\mathbf{Q}_{k+1} = \mathrm{Prox}_{\mu\ell_{2,1}}(\mathbf{Z}_{k+1})$ //sparsity promotion
6. **end**
7. $\mathbf{I}(\mathbf{x}) = \sum_t \mid \mathbf{Q}(\mathbf{x}, t) \mid$ //Intensity plot

[Lorentz et al.,'14; Combettes et al.,'11]

SLIM

Linearized Bregman algorithm

1. $\quad$ **Data d, slowness square m** $\quad$ //Input
2. $\quad$ **for** $k = 0, 1, \cdots$
3. $\quad\quad \mathbf{V}_k = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d}))$ $\quad$ //adjoint solve
4. $\quad\quad \mathbf{Z}_{k+1} = \mathbf{Z}_k - t_k \mathbf{V}_k$ $\quad$ //auxiliary variable update
5. $\quad\quad \mathbf{Q}_{k+1} = \mathrm{Prox}_{\mu\ell_{2,1}}(\mathbf{Z}_{k+1})$ $\quad$ //sparsity promotion
6. $\quad$ **end**
7. $\quad \mathbf{I}(\mathbf{x}) = \sum_t \mid \mathbf{Q}(\mathbf{x}, t) \mid$ $\quad$ //Intensity plot

* $\Pi_\epsilon(\mathbf{x}) = \max\{0, 1 - \frac{\epsilon}{||\mathbf{x}||}\}.(\mathbf{x})$ the projection on to ℓ_2 norm ball

[Lorentz et al.,'14; Combettes et al.,'11]

SLIM

Linearized Bregman algorithm

1. **Data d, slowness square m** //Input
2. **for** $k = 0, 1, \cdots$
3. $\mathbf{V}_k = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d}))$ //adjoint solve
4. $\mathbf{Z}_{k+1} = \mathbf{Z}_k - t_k \mathbf{V}_k$ //auxiliary variable update
5. $\mathbf{Q}_{k+1} = \mathrm{Prox}_{\mu \ell_{2,1}}(\mathbf{Z}_{k+1})$ //sparsity promotion
6. **end**
7. $\mathbf{I}(\mathbf{x}) = \sum_t \mid \mathbf{Q}(\mathbf{x}, t) \mid$ //Intensity plot

* $\Pi_\epsilon(\mathbf{x}) = \max\{0, 1 - \frac{\epsilon}{\|\mathbf{x}\|}\}.(\mathbf{x})$ the projection on to ℓ_2 norm ball

*where $t_k = \frac{\|\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d}\|^2}{\|\mathcal{F}^\top[\mathbf{m}](\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d})\|^2}$ is the dynamic step length

[Lorentz et al.,'14; Combettes et al.,'11]

SLIM

Linearized Bregman algorithm

1. **Data d, slowness square m** //Input
2. **for** $k = 0, 1, \cdots$
3. $\mathbf{V}_k = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d}))$ //adjoint solve
4. $\mathbf{Z}_{k+1} = \mathbf{Z}_k - t_k \mathbf{V}_k$ //auxiliary variable update
5. $\mathbf{Q}_{k+1} = \mathrm{Prox}_{\mu\ell_{2,1}}(\mathbf{Z}_{k+1})$ //sparsity promotion
6. **end**
7. $\mathbf{I}(\mathbf{x}) = \sum_t \mid \mathbf{Q}(\mathbf{x}, t) \mid$ //Intensity plot

* $\Pi_\epsilon(\mathbf{x}) = \max\{0, 1 - \frac{\epsilon}{\|\mathbf{x}\|}\}.(\mathbf{x})$ the projection on to ℓ_2 norm ball

*where $t_k = \frac{\|\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d}\|^2}{\|\mathcal{F}^\top[\mathbf{m}](\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d})\|^2}$ is the dynamic step length

* $\mathrm{Prox}_{\mu\ell_{2,1}}(\mathbf{C}) := \arg\min_{\mathbf{B}} \|\mathbf{B}\|_{2,1} + \frac{1}{2\mu}\|\mathbf{C} - \mathbf{B}\|_F^2$ is the proximal mapping of the $\ell_{2,1}$ norm

[Lorentz et al.,’14; Combettes et al.,’11]

Linearized Bregman algorithm

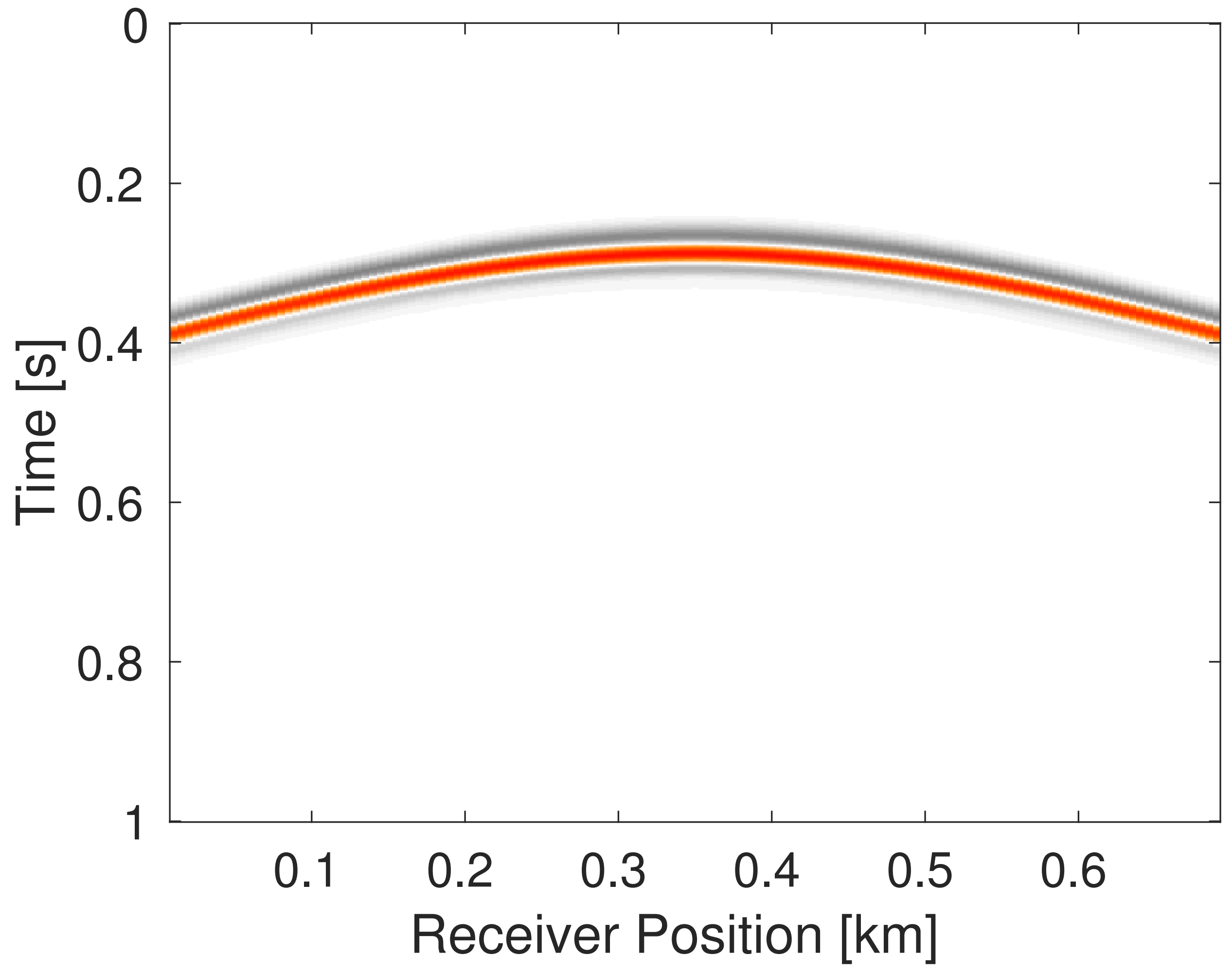
1. **Data d, slowness square m** //Input
2. **for** $k = 0, 1, \cdots$
3. $\mathbf{V}_k = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d}))$ //adjoint solve
4. $\mathbf{Z}_{k+1} = \mathbf{Z}_k - t_k \mathbf{V}_k$ //auxiliary variable update
5. $\mathbf{Q}_{k+1} = \mathrm{Prox}_{\mu \ell_{2,1}}(\mathbf{Z}_{k+1})$ //sparsity promotion
6. **end**
7. $\mathbf{I}(\mathbf{x}) = \sum_t \mid \mathbf{Q}(\mathbf{x}, t) \mid$ //Intensity plot

* $\Pi_\epsilon(\mathbf{x}) = \max\{0, 1 - \frac{\epsilon}{\|\mathbf{x}\|}\}.(\mathbf{x})$ the projection on to ℓ_2 norm ball

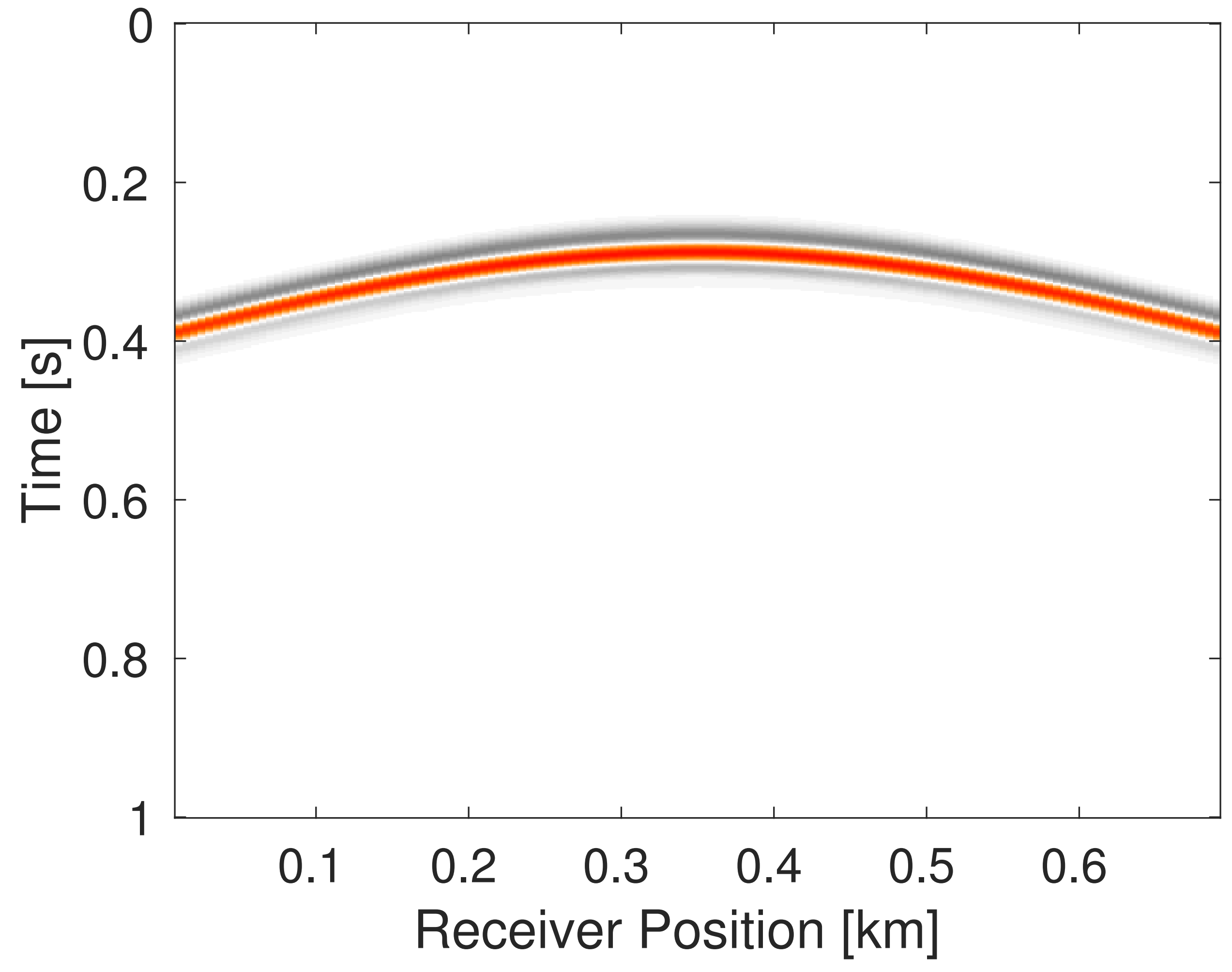
*where $t_k = \frac{\|\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d}\|^2}{\|\mathcal{F}^\top[\mathbf{m}](\mathcal{F}[\mathbf{m}](\mathbf{Q}_k) - \mathbf{d})\|^2}$ is the dynamic step length

* $\mathrm{Prox}_{\mu \ell_{2,1}}(\mathbf{C}) := \arg\min_{\mathbf{B}} \|\mathbf{B}\|_{2,1} + \frac{1}{2\mu}\|\mathbf{C} - \mathbf{B}\|_F^2$ is the proximal mapping of the $\ell_{2,1}$ norm

- **Source location**: estimated as outlier in intensity plot
- **Source-time function**: temporal variation of wavefield at estimated source location

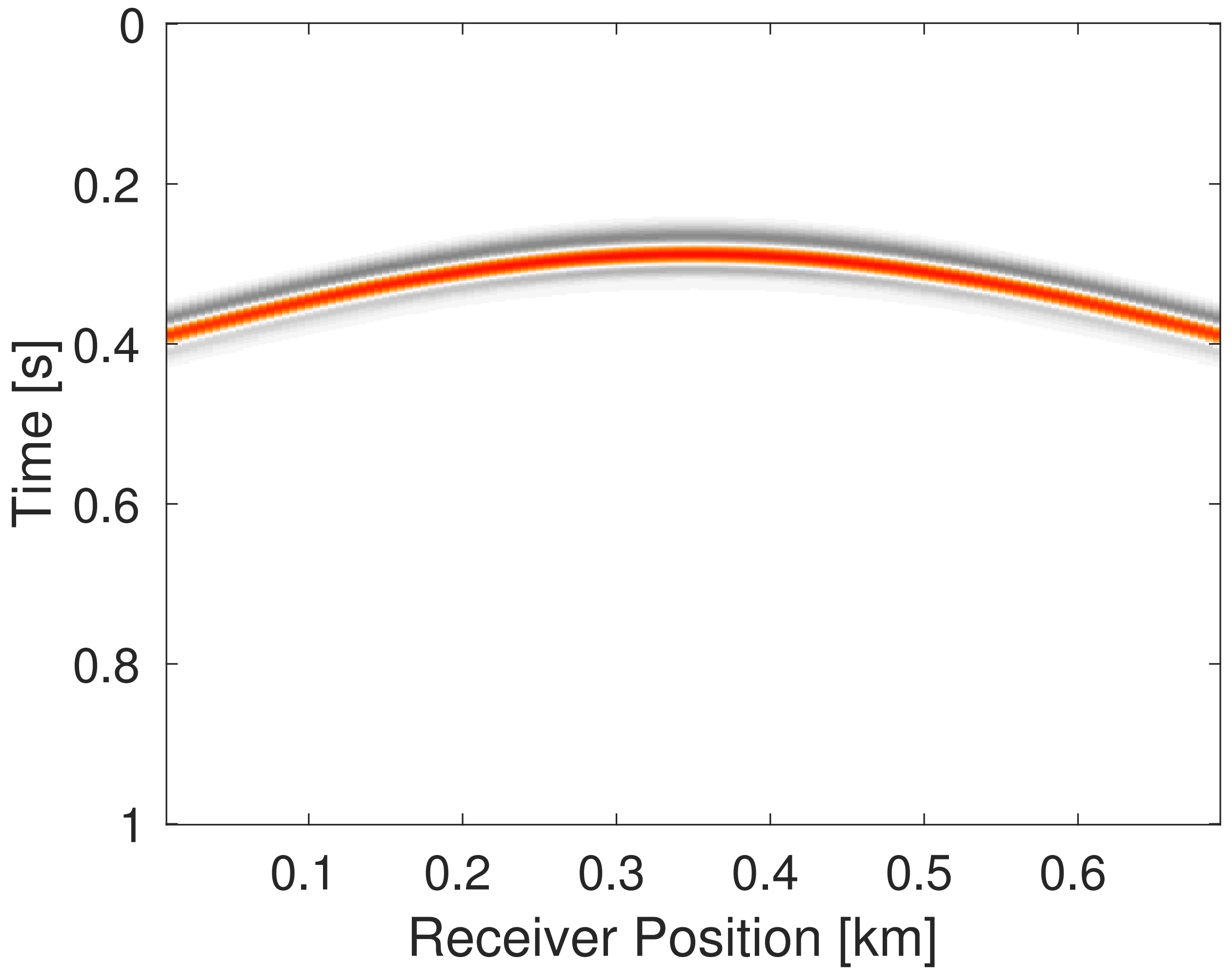


Time [s]
0
0.2
0.4
0.6
0.8
1
0.1
0.2
0.3
0.4
0.5
0.6
Receiver Position [km]



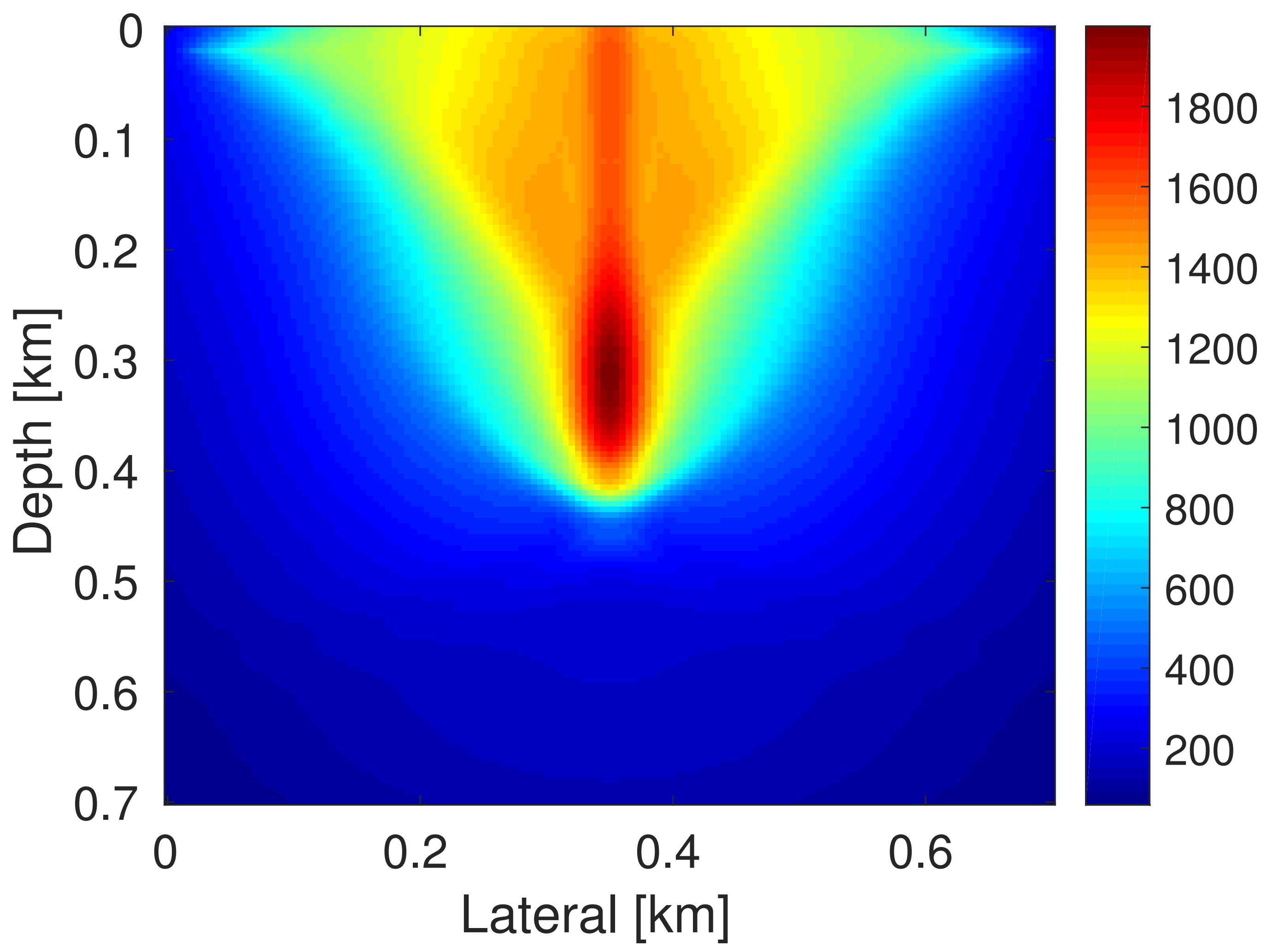
$$\mathbf{V}_1 = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_0) - \mathbf{d}))$$

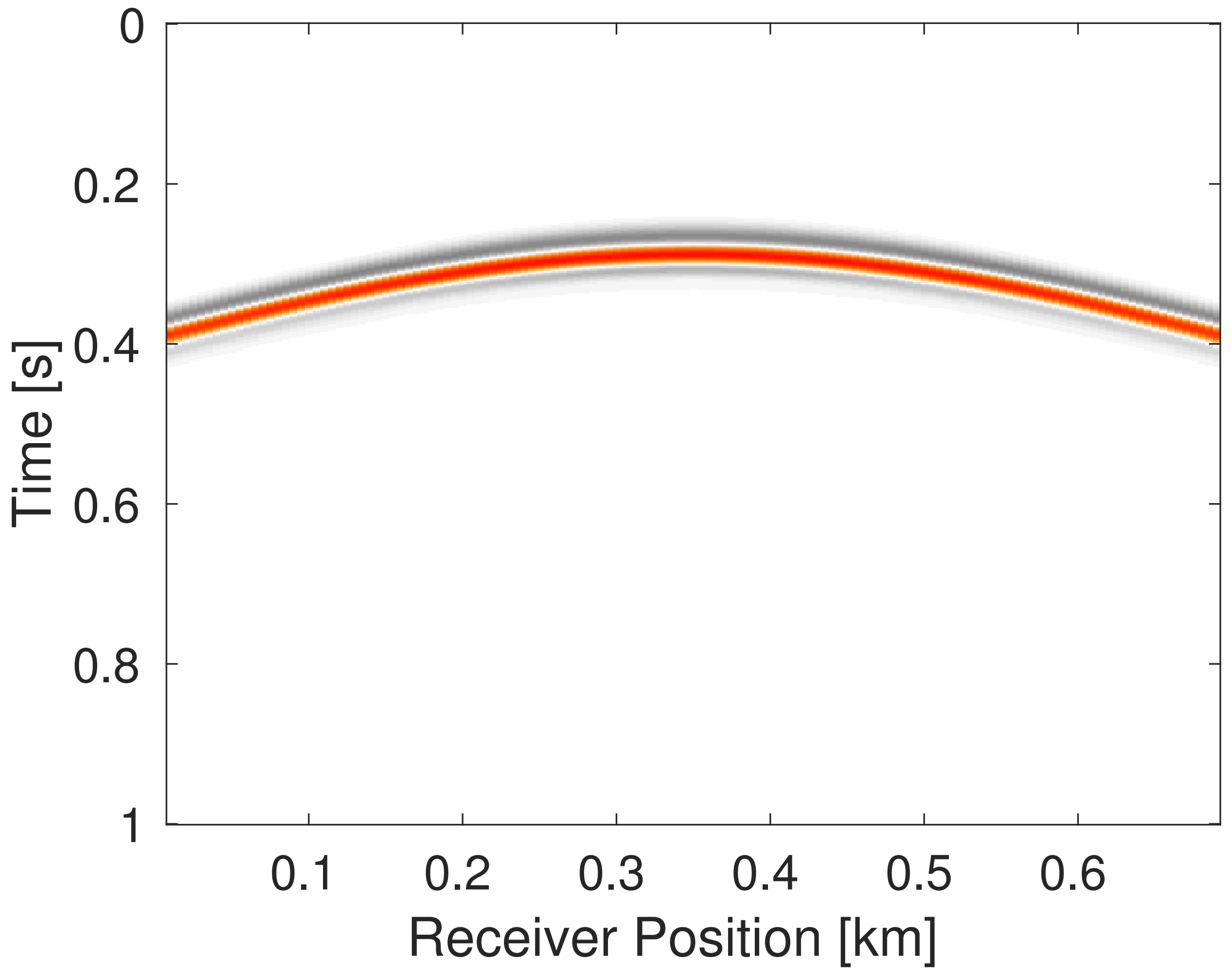
Adjoint solve



$$\mathbf{V}_1 = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_0) - \mathbf{d}))$$

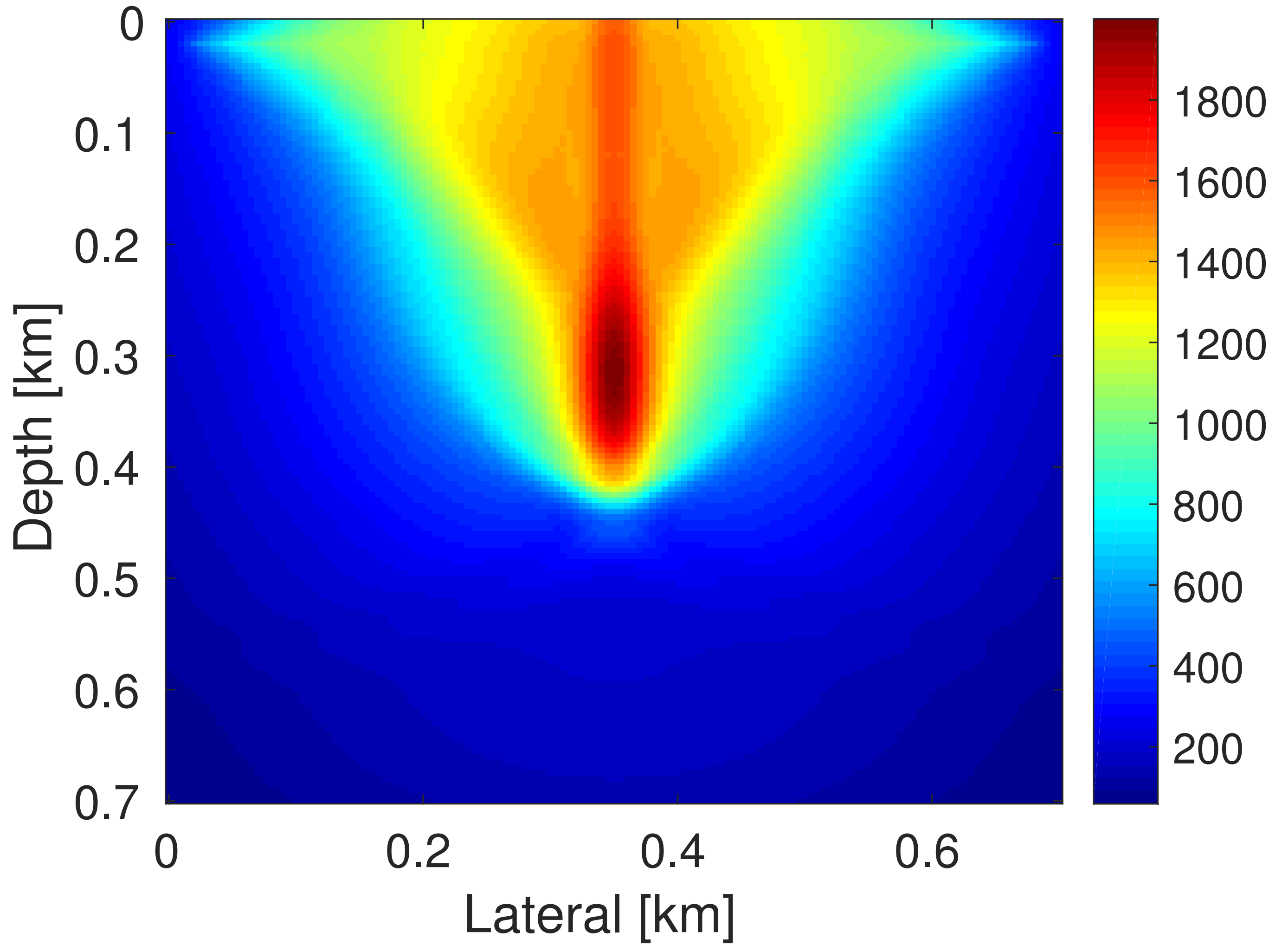
Adjoint solve





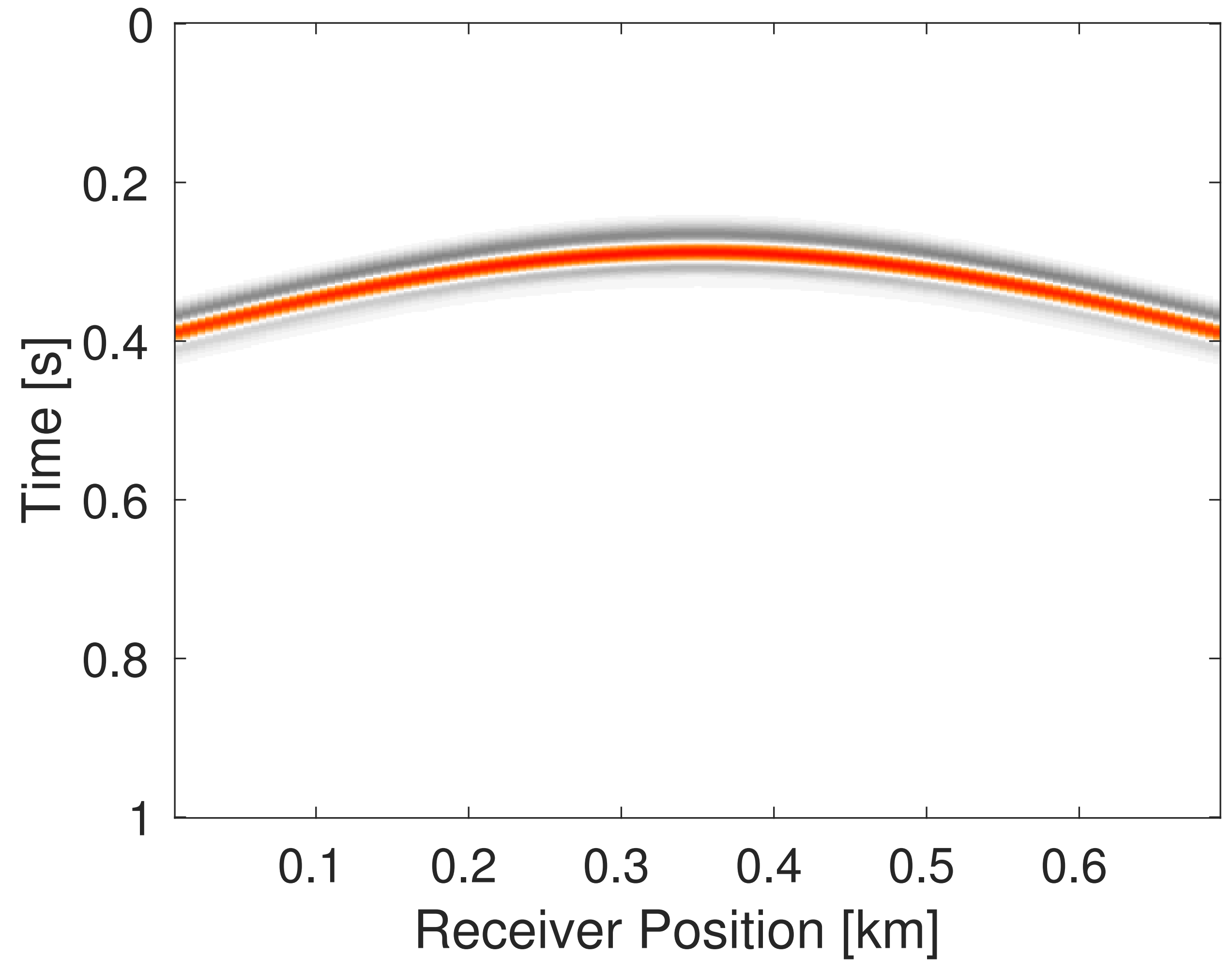
$$\mathbf{V}_1 = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_0) - \mathbf{d}))$$

Adjoint solve



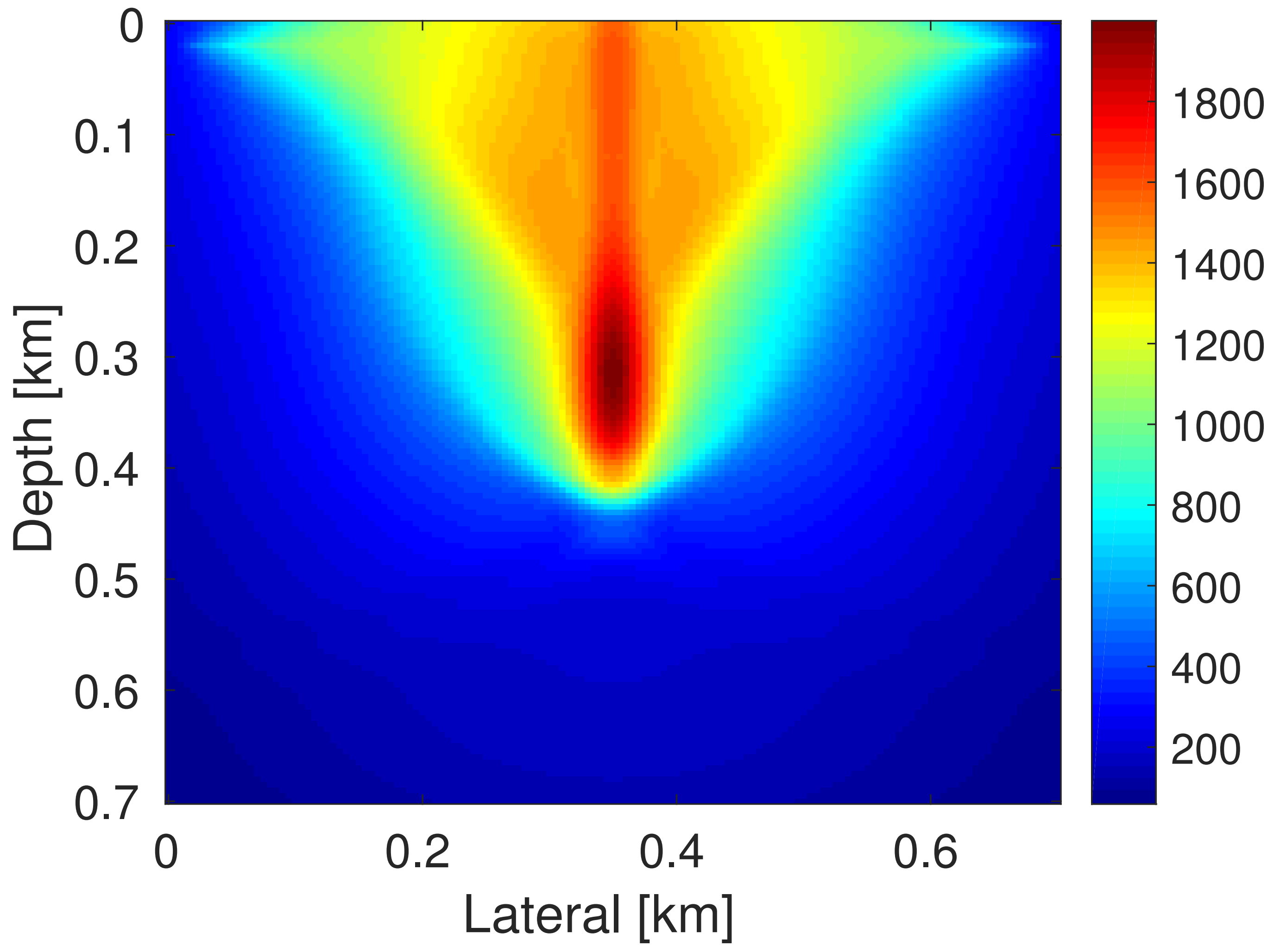
Auxiliary variable update

$$\mathbf{Z}_1 = \mathbf{Z}_0 - t_1 \mathbf{V}_1$$



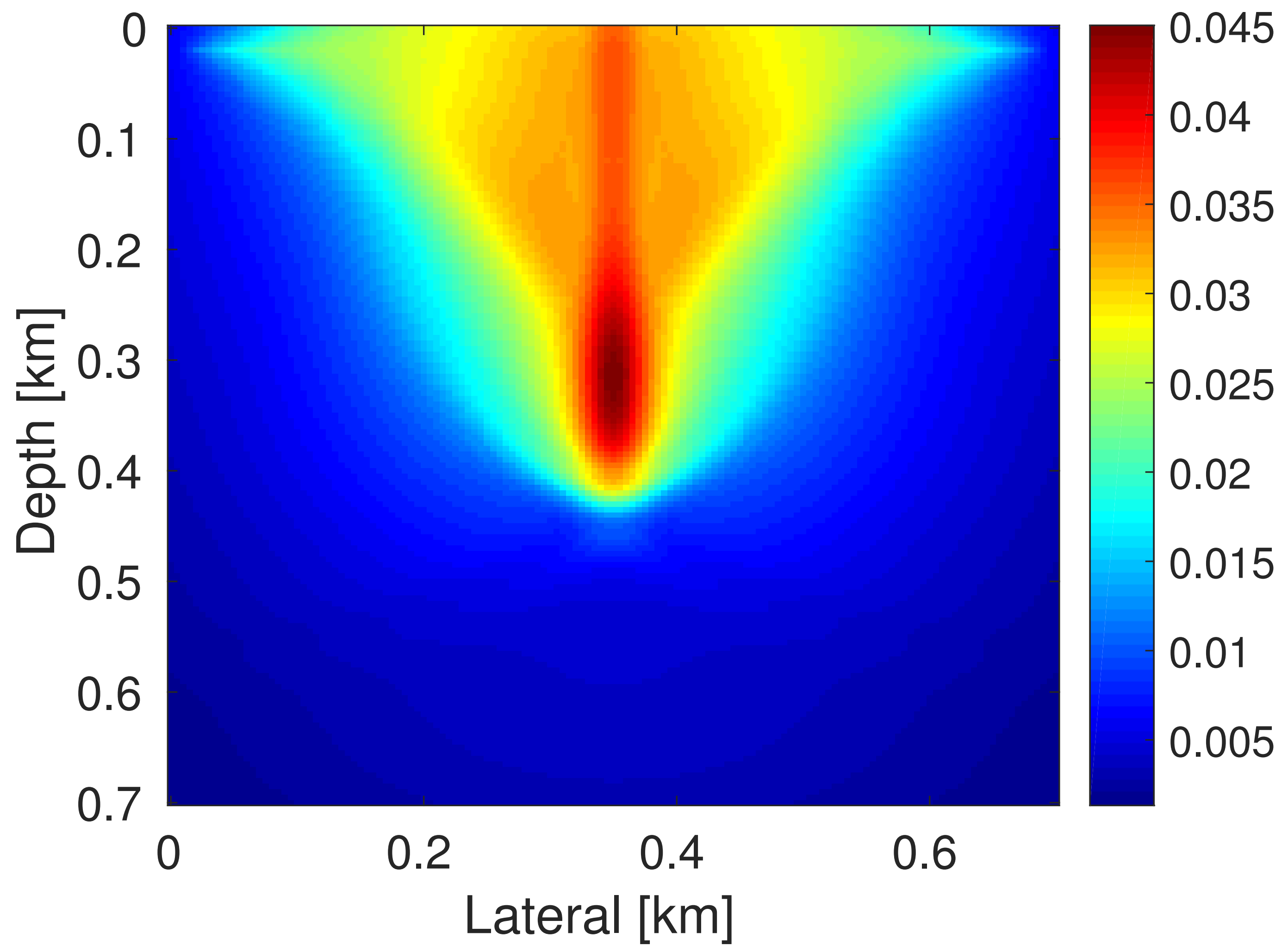
$$\mathbf{V}_1 = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_0) - \mathbf{d}))$$

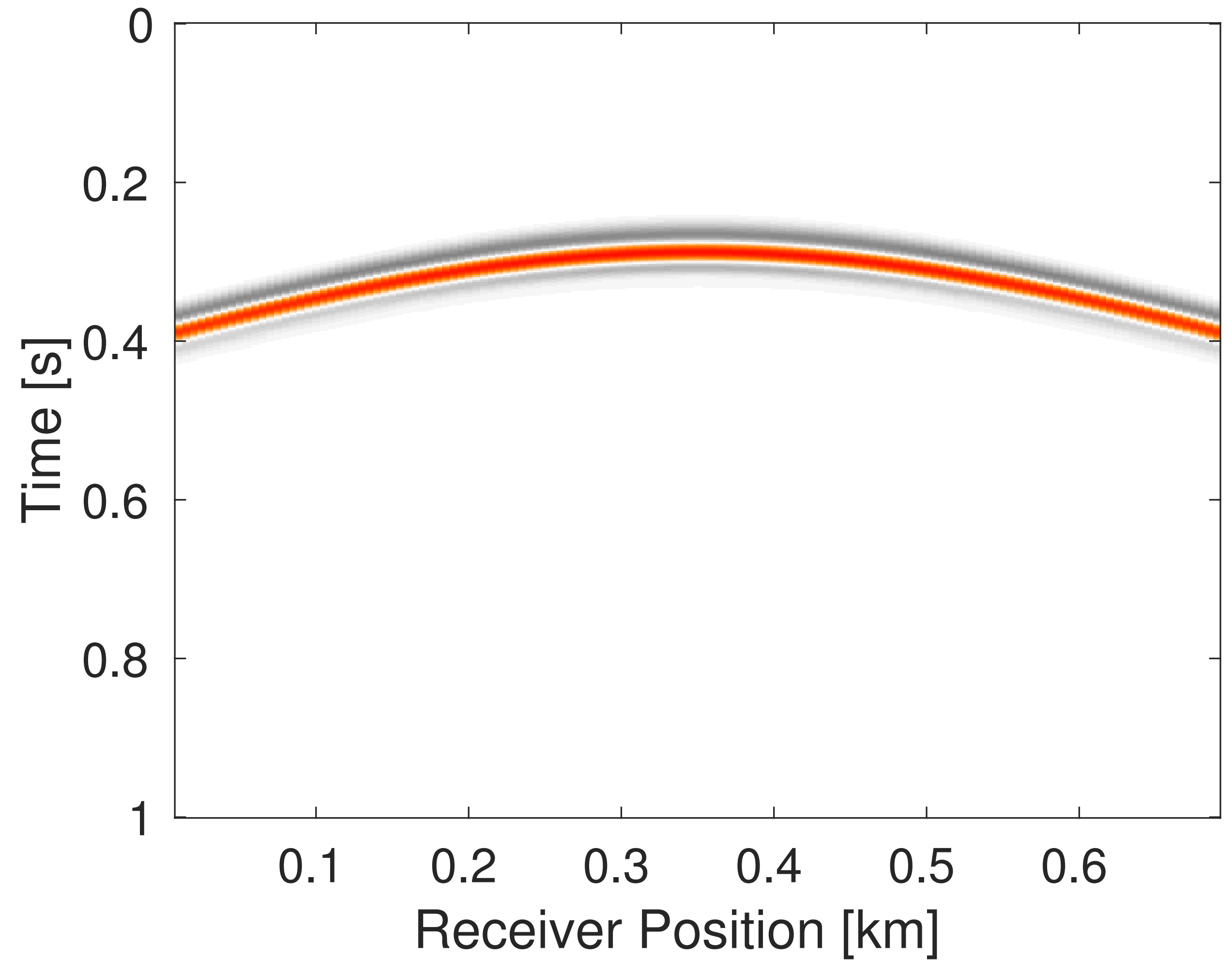
Adjoint solve



Auxiliary variable update

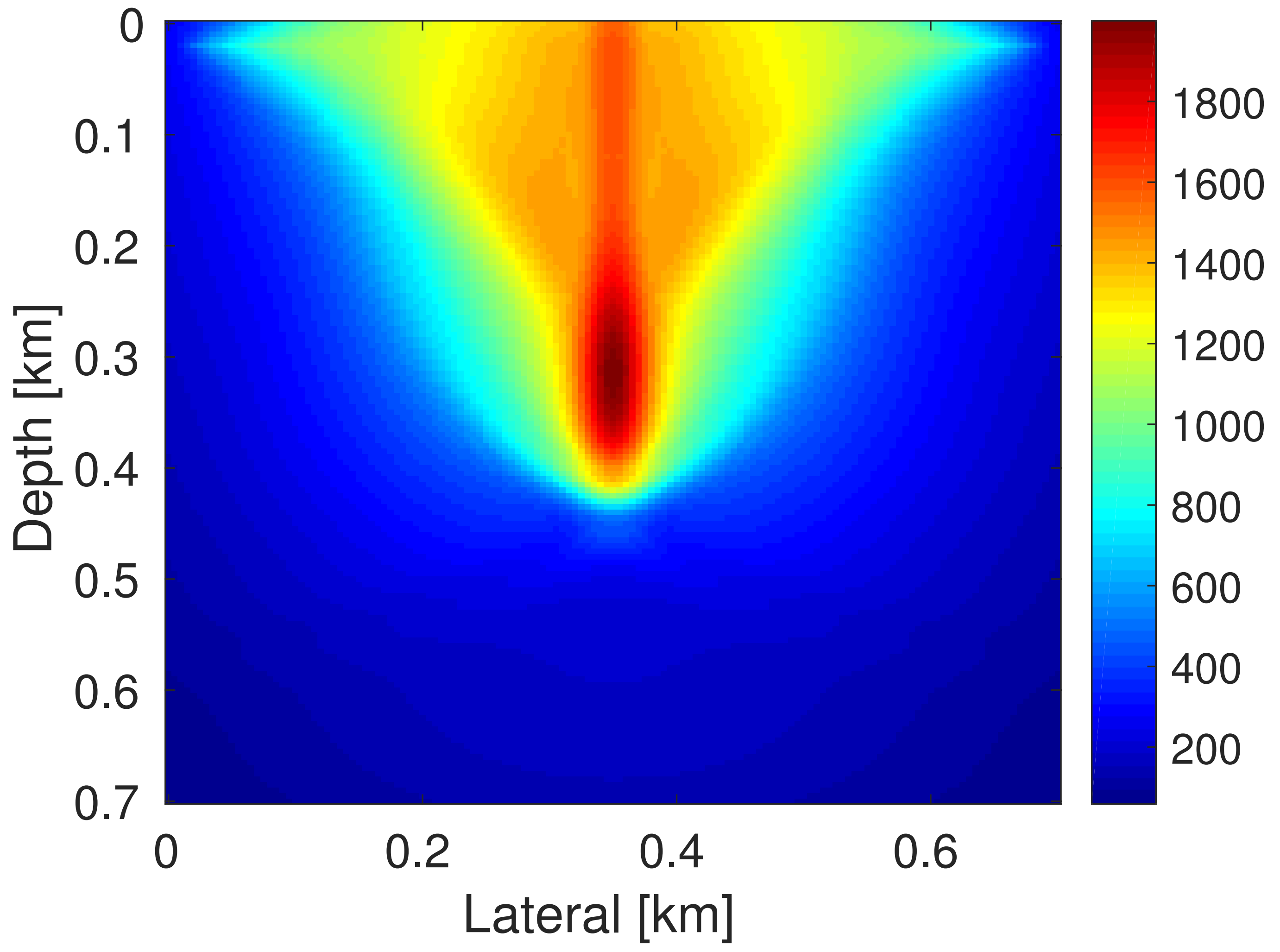
$$\mathbf{Z}_1 = \mathbf{Z}_0 - t_1 \mathbf{V}_1$$





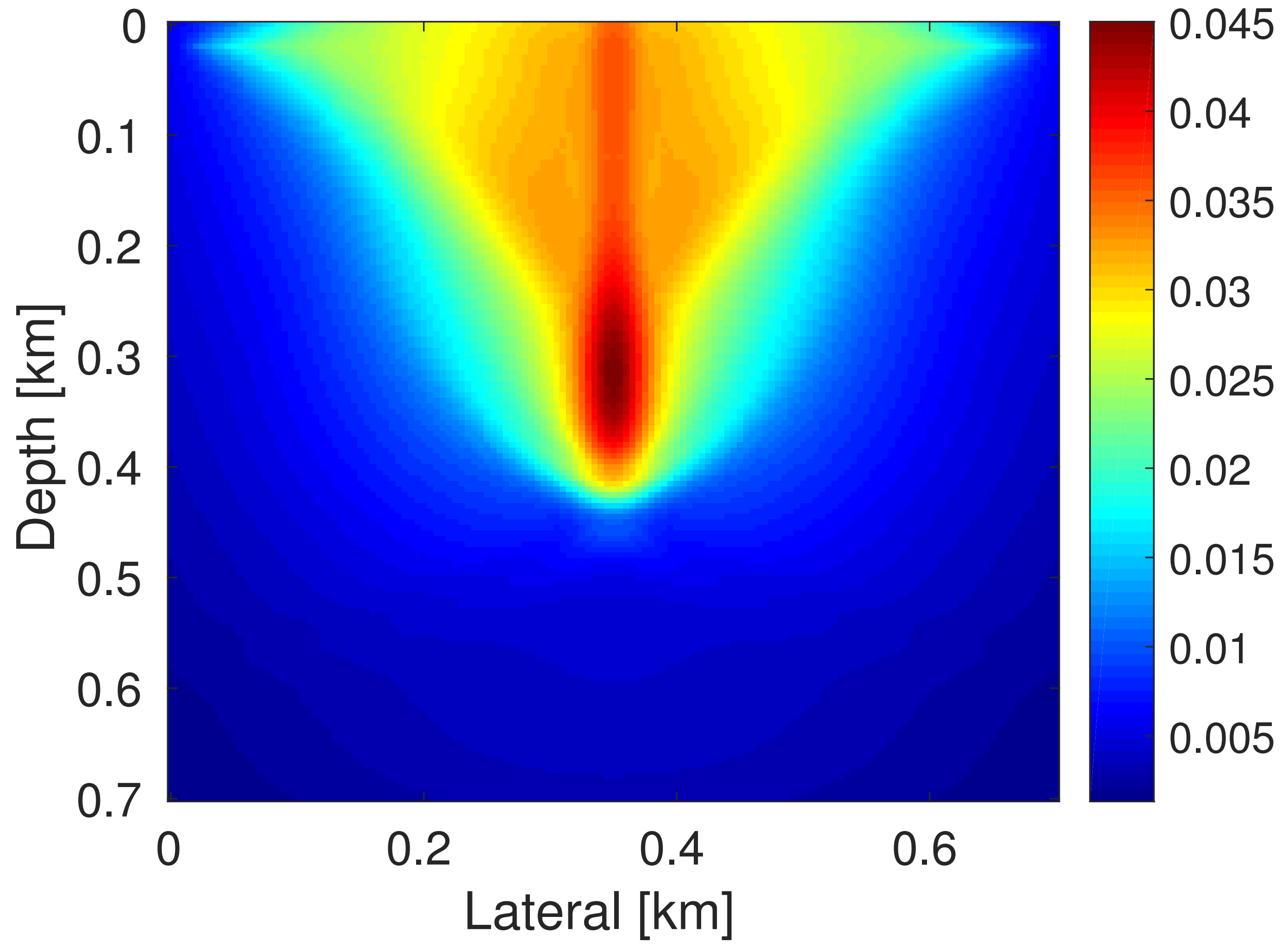
$$\mathbf{V}_1 = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_0) - \mathbf{d}))$$

Adjoint solve



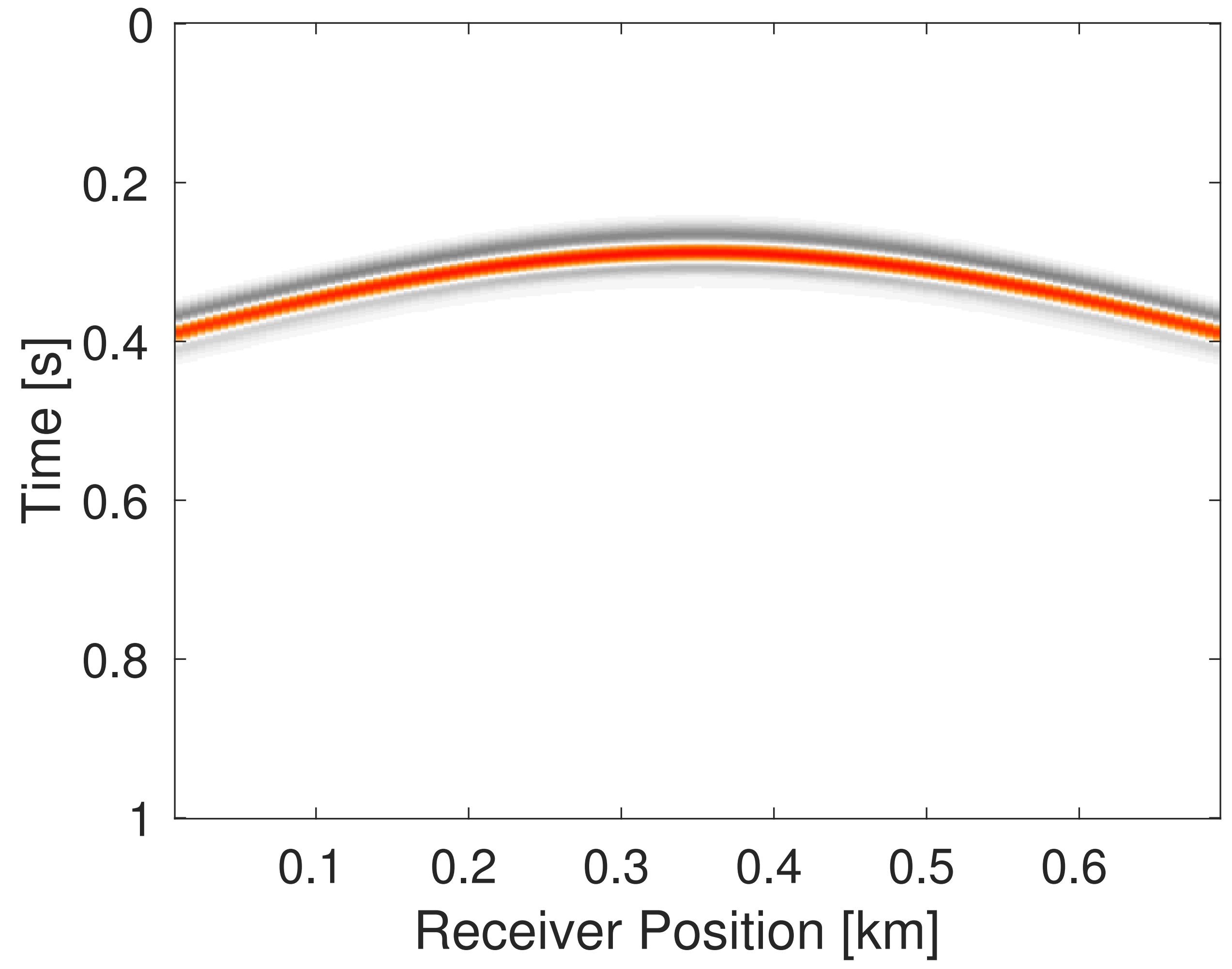
Auxiliary variable update

$$\mathbf{Z}_1 = \mathbf{Z}_0 - t_1\mathbf{V}_1$$



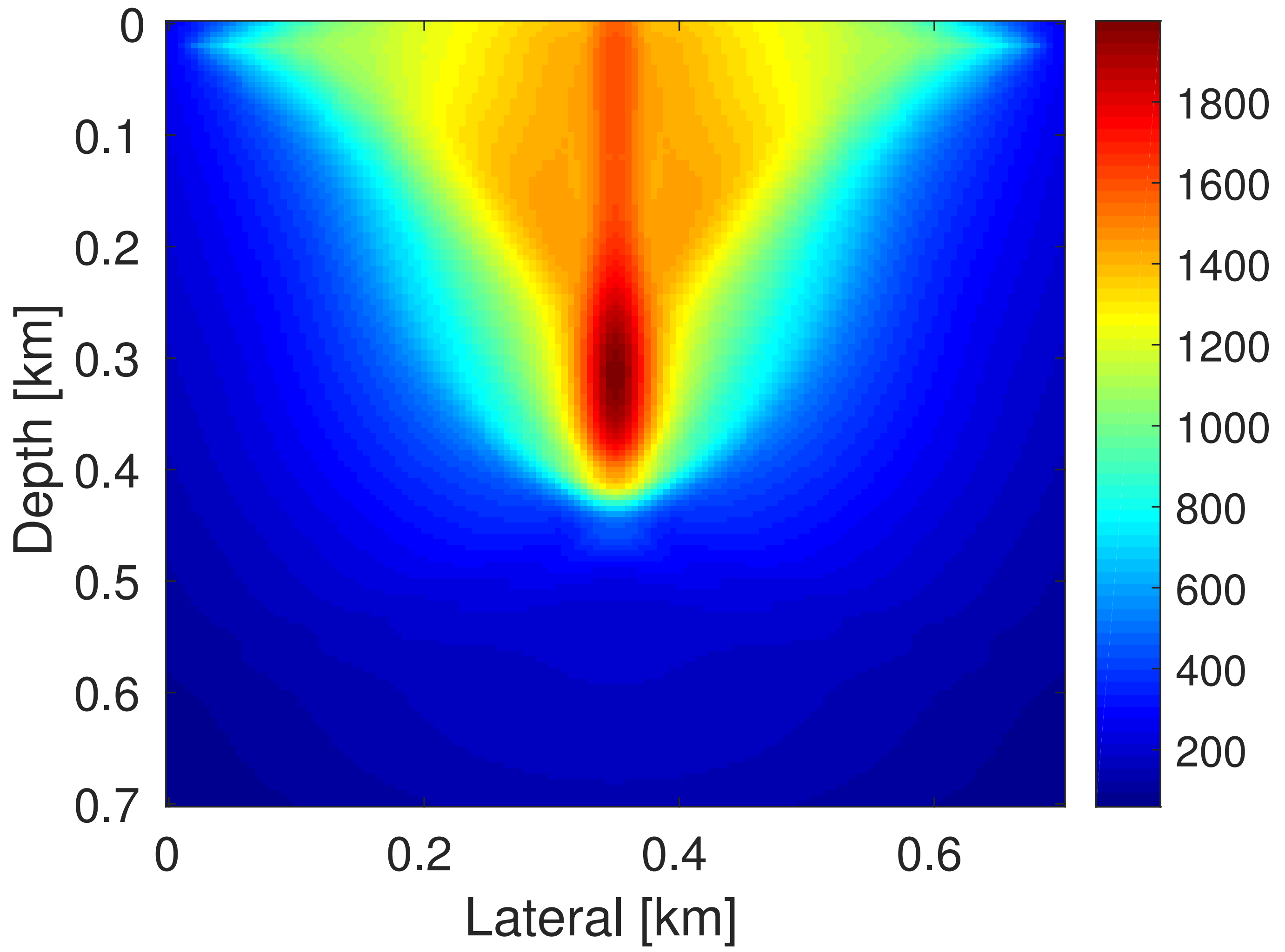
$$\mathbf{Q}_1 = \mathrm{Prox}_{\mu\ell_{2,1}}(\mathbf{Z}_1)$$

Sparsity promotion



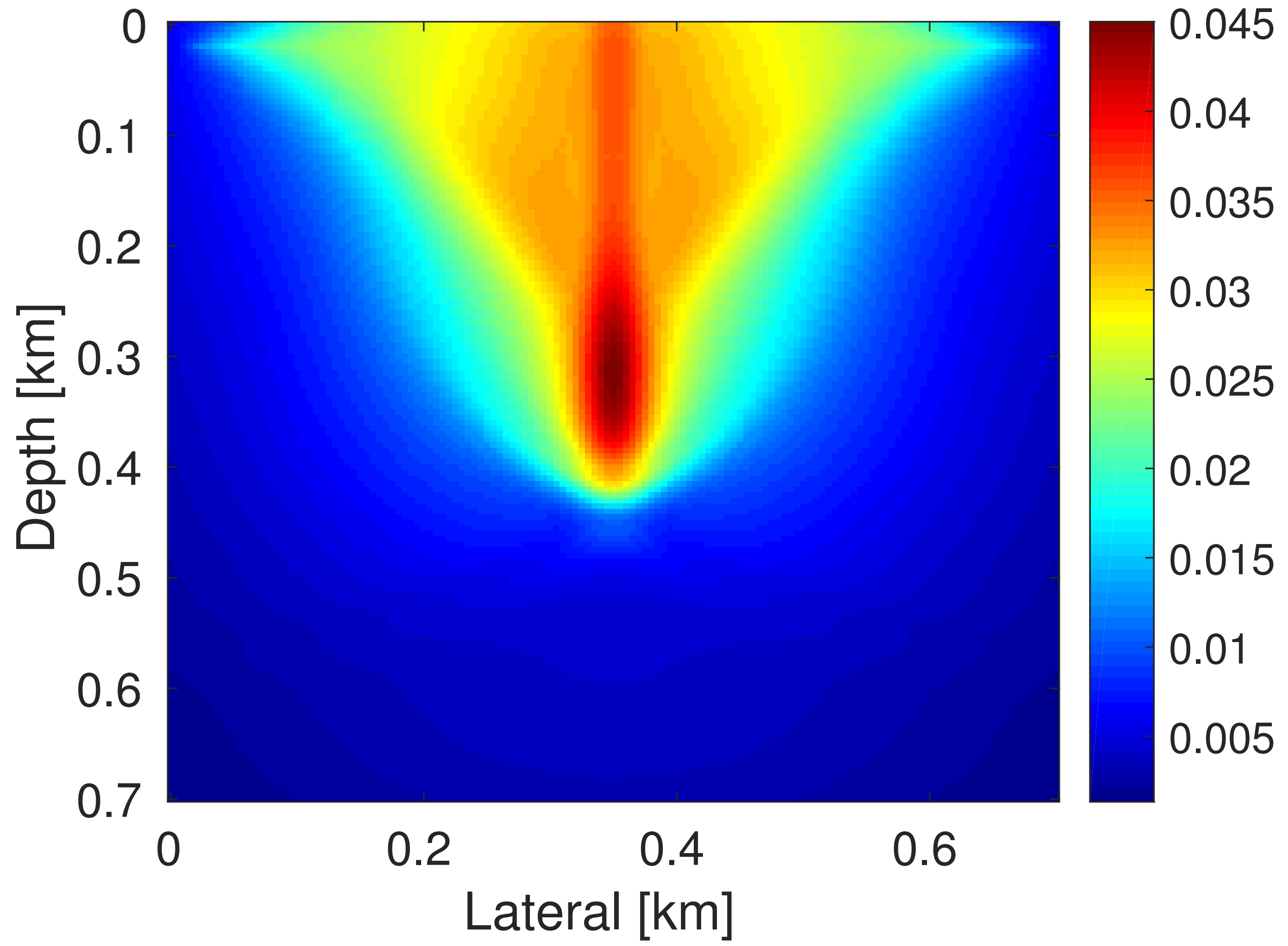
$$\mathbf{V}_1 = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_0) - \mathbf{d}))$$

Adjoint solve



Auxiliary variable update

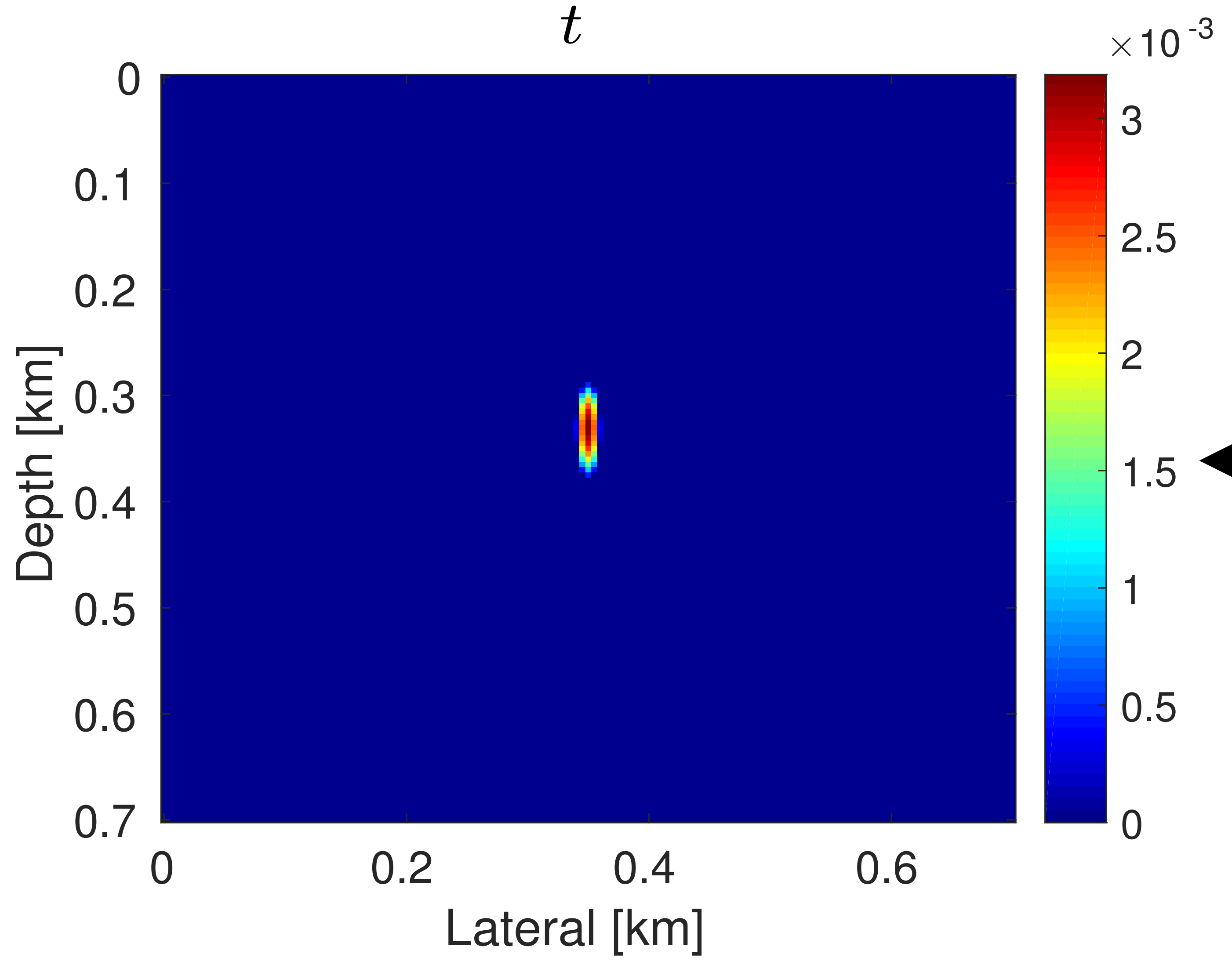
$$\mathbf{Z}_1 = \mathbf{Z}_0 - t_1 \mathbf{V}_1$$

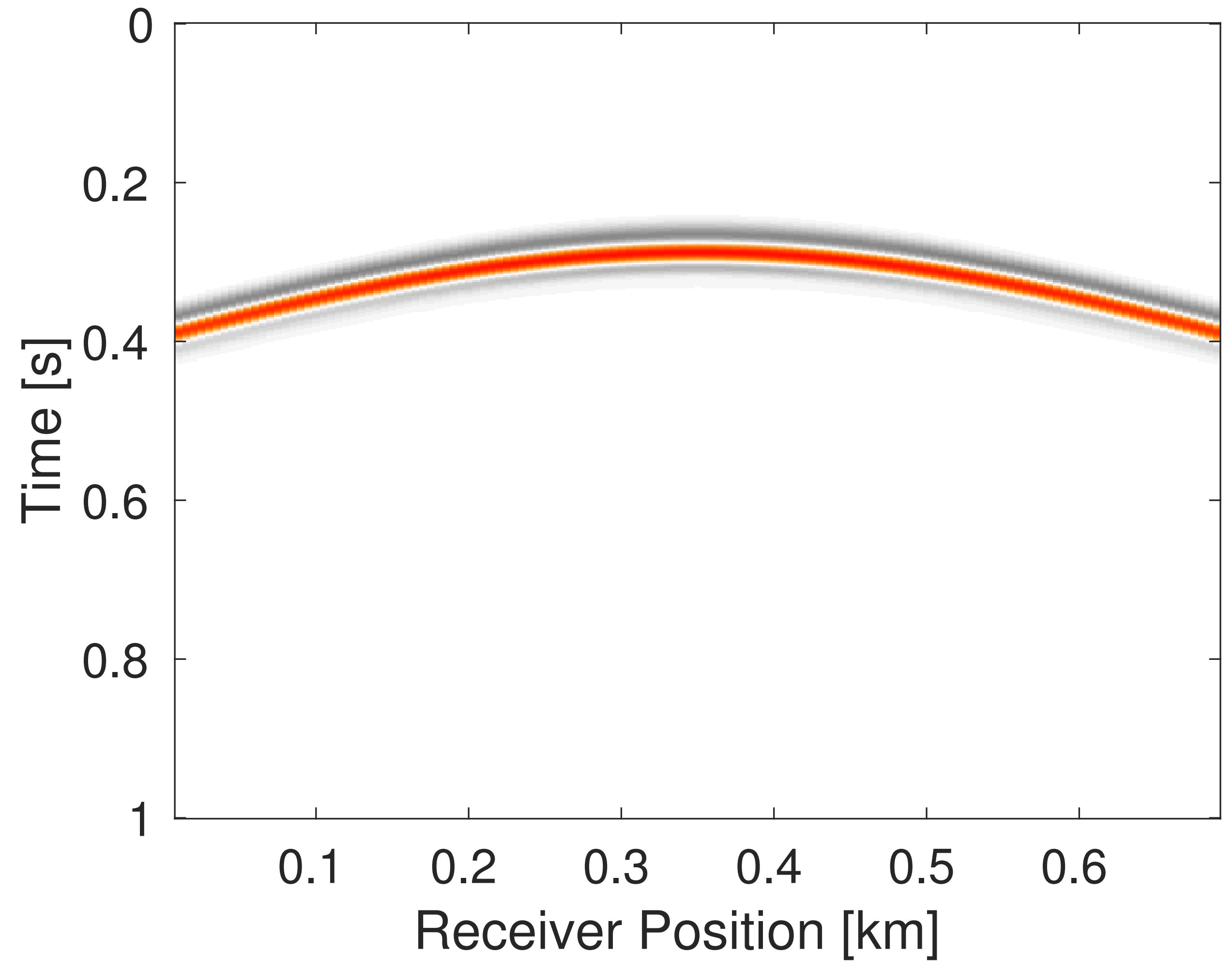


$$\mathbf{Q}_1 = \mathrm{Prox}_{\mu \ell_{2,1}}(\mathbf{Z}_1)$$

Sparsity promotion

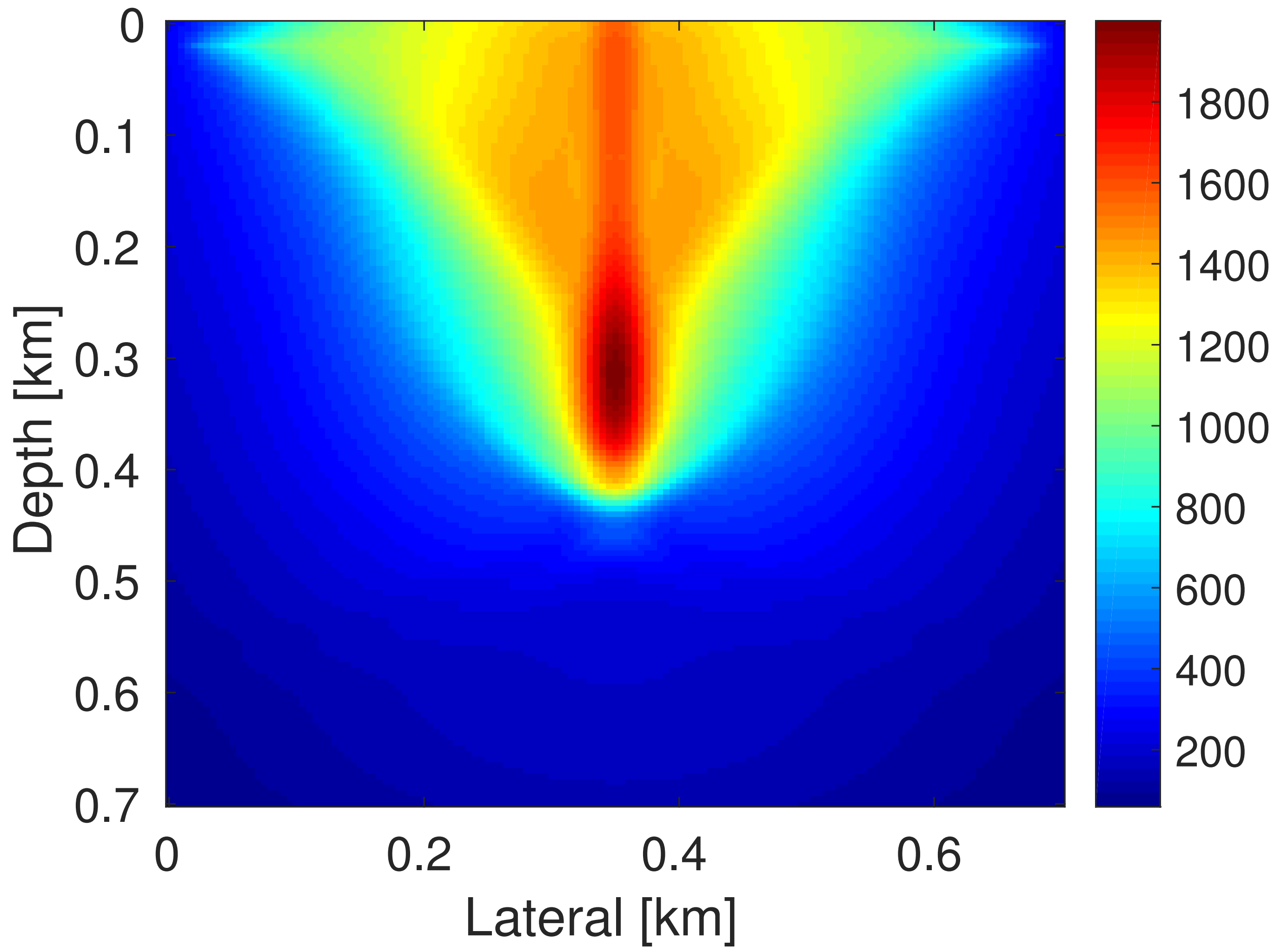
$$\mathbf{I}(\mathbf{x}) = \sum_t \mid \mathbf{Q}_1(\mathbf{x}, t) \mid$$





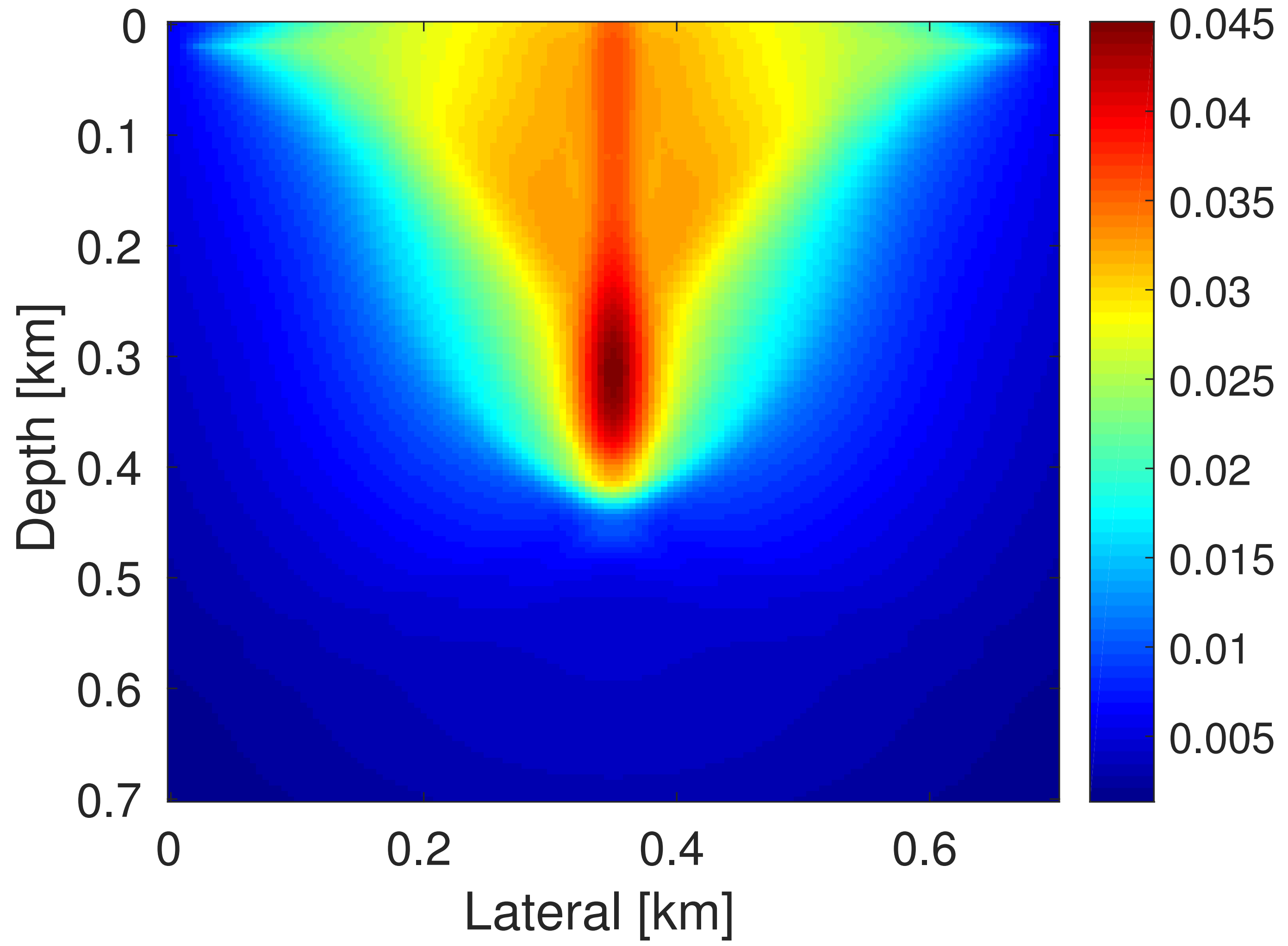
$$\mathbf{V}_1 = \mathcal{F}^\top[\mathbf{m}](\Pi_\epsilon(\mathcal{F}[\mathbf{m}](\mathbf{Q}_0) - \mathbf{d}))$$

Adjoint solve



Auxiliary variable update

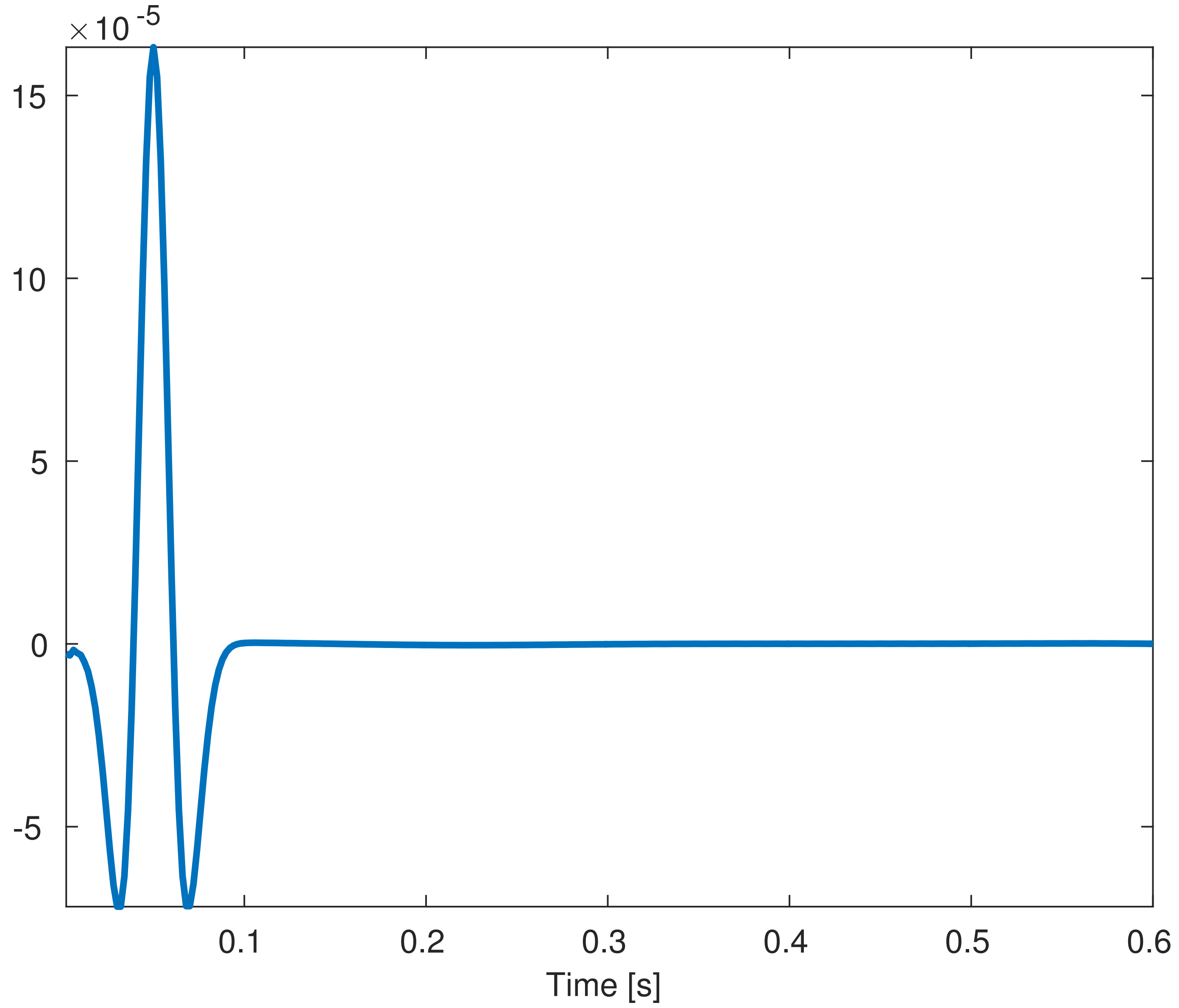
$$\mathbf{Z}_1 = \mathbf{Z}_0 - t_1\mathbf{V}_1$$



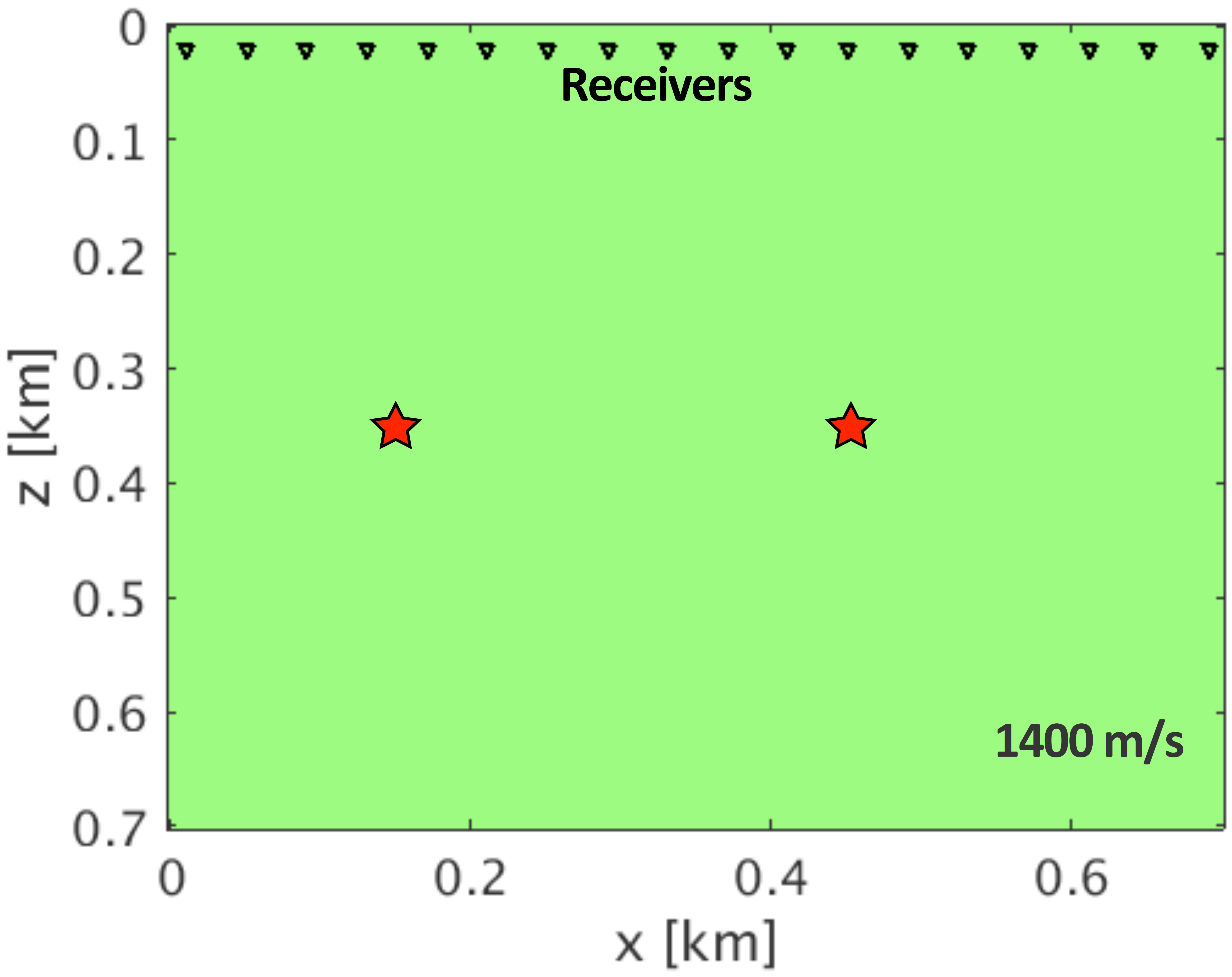
$$\mathbf{Q}_1 = \mathrm{Prox}_{\mu\ell_{2,1}}(\mathbf{Z}_1)$$

Sparsity promotion

Source-time function



Case study: two far sources

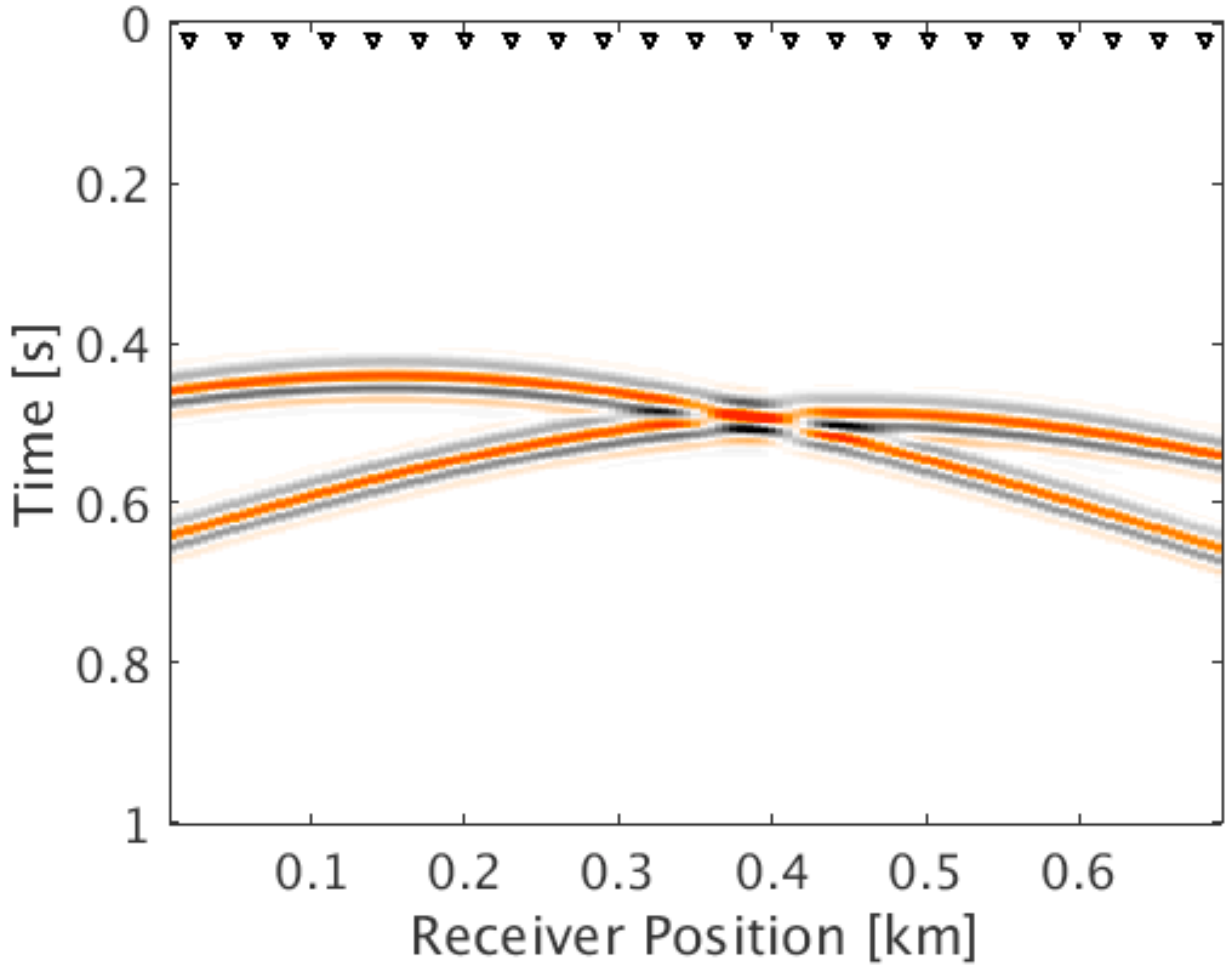


Acquisition setup

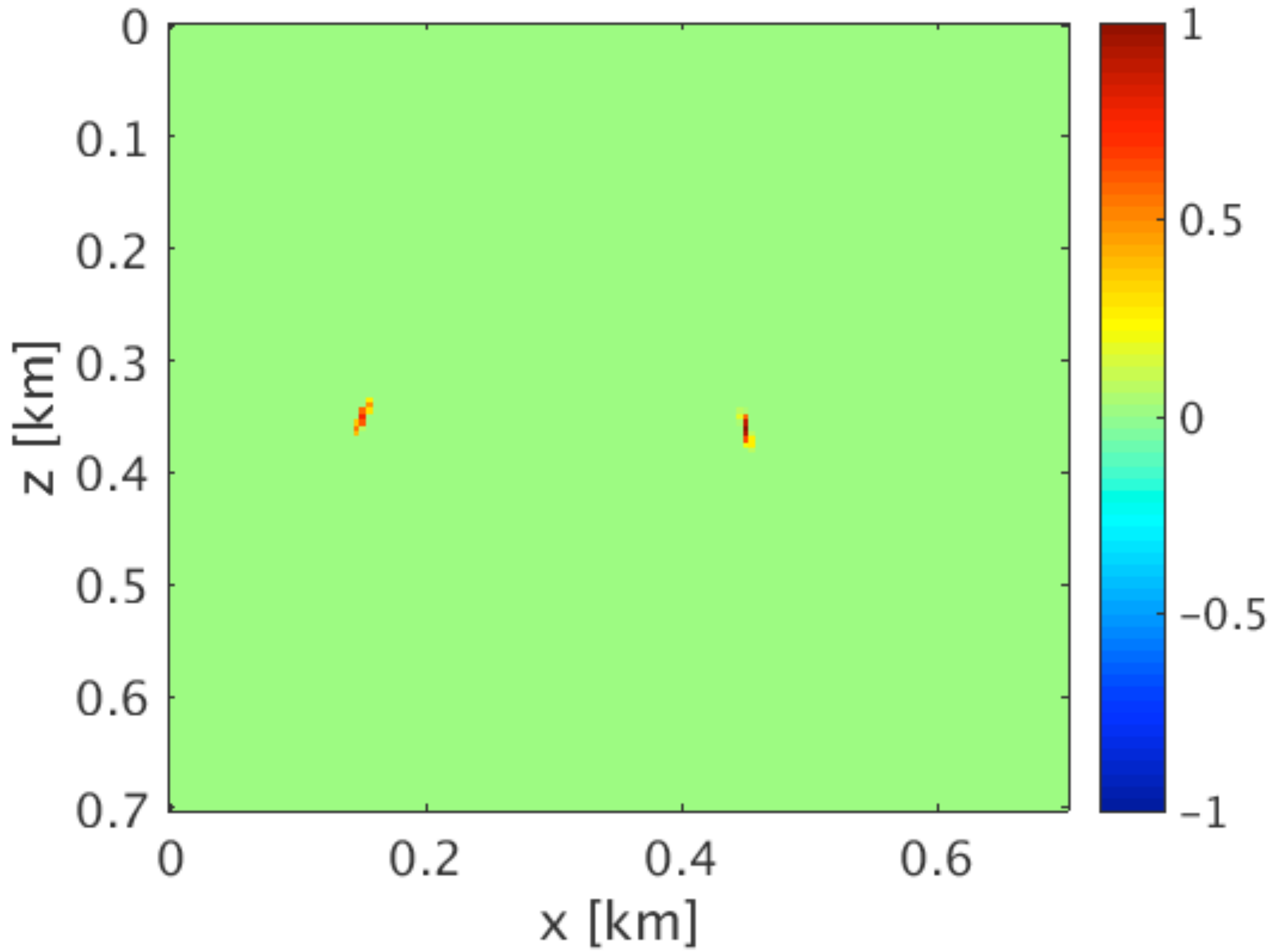
Modeling information:

Model size: 0.7 km x 0.7 km
Grid spacing: 5m
Receiver spacing: 10m
Receiver depth: 20m
Fixed spread: 0.69km
Sampling interval: 2 ms
Recording length: 1s
Peak frequency : 30 Hz
Dominant wavelength: 46 m

Data and estimated location

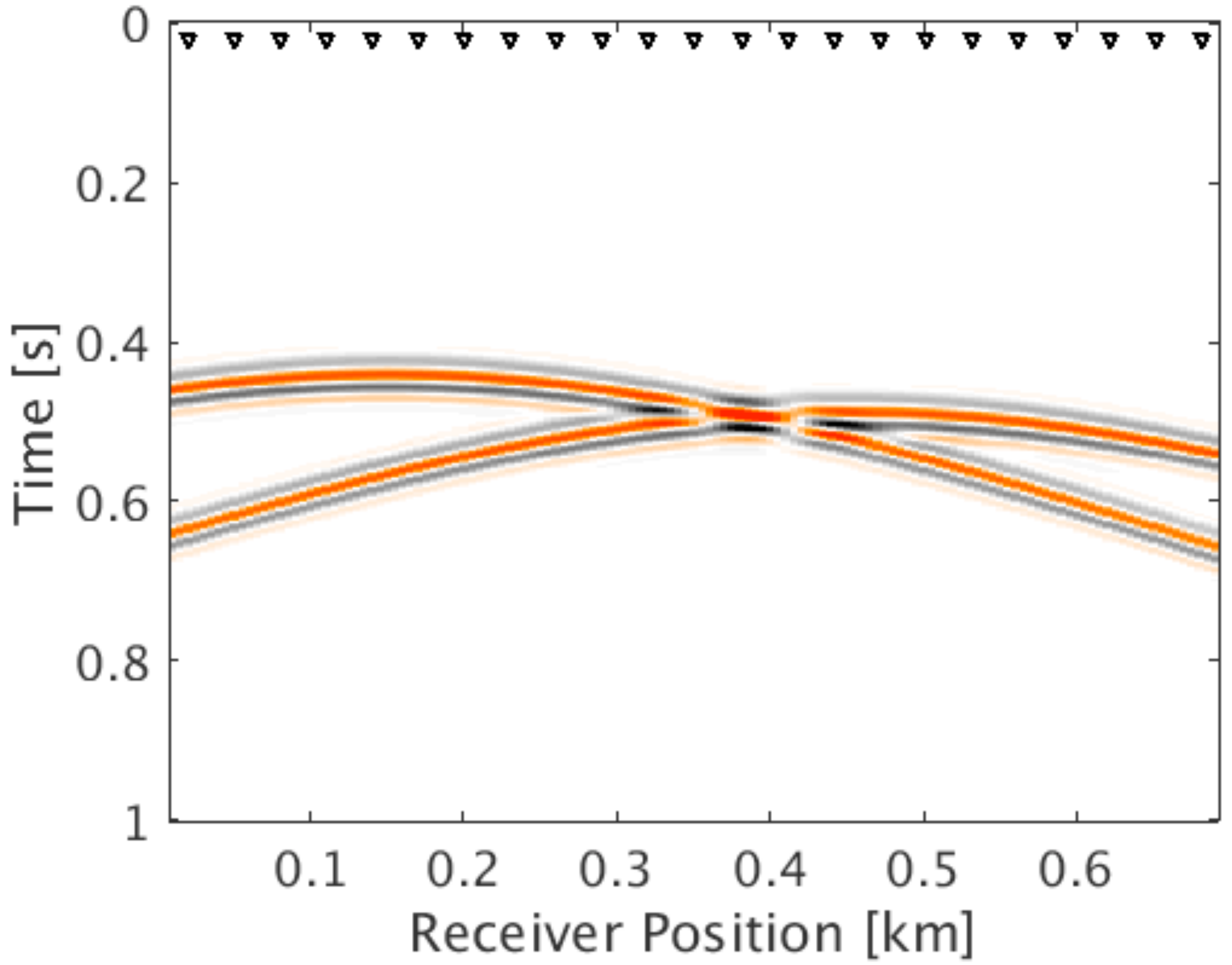


Microseismic data

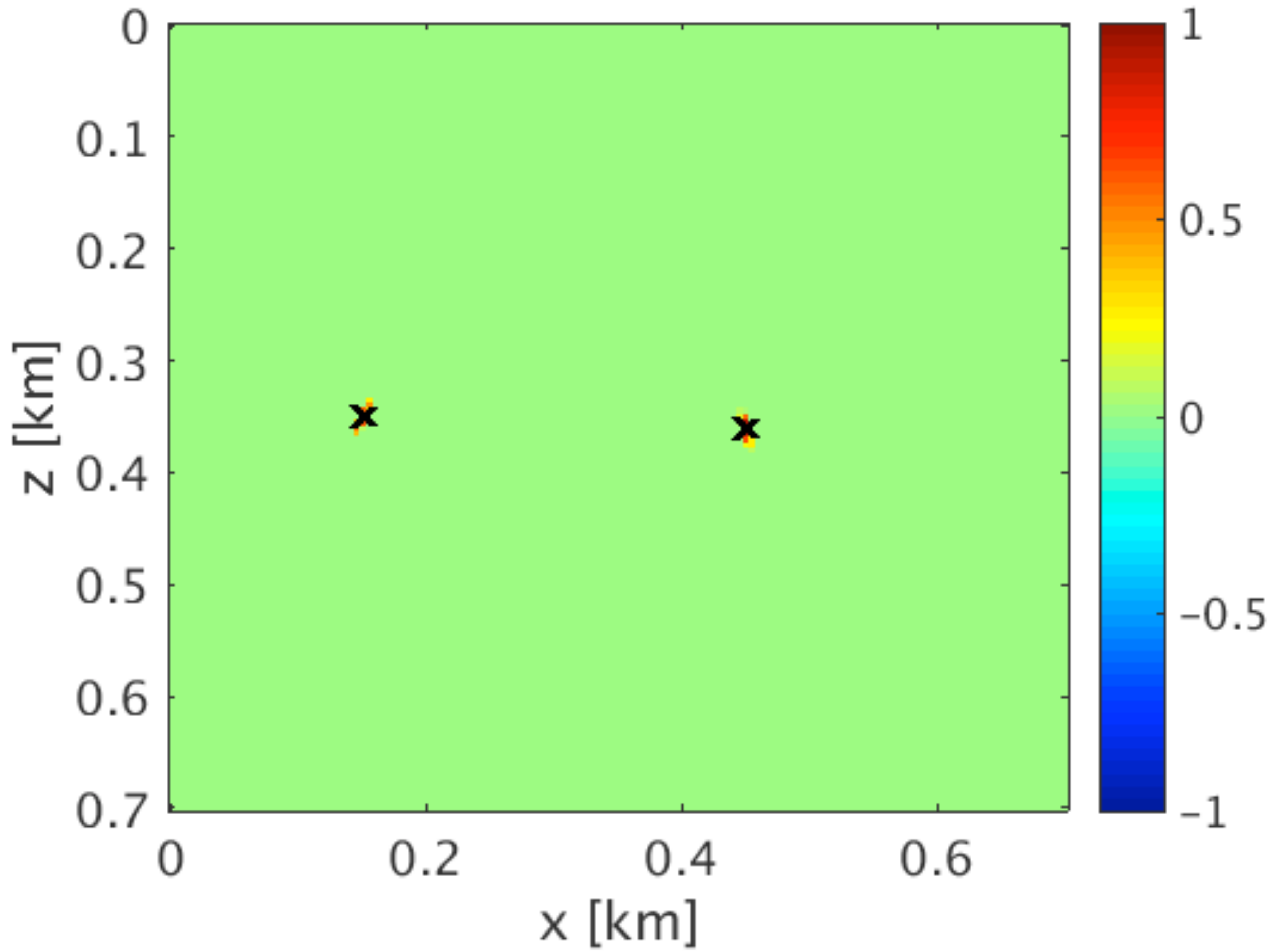


Estimated location

Data and estimated location

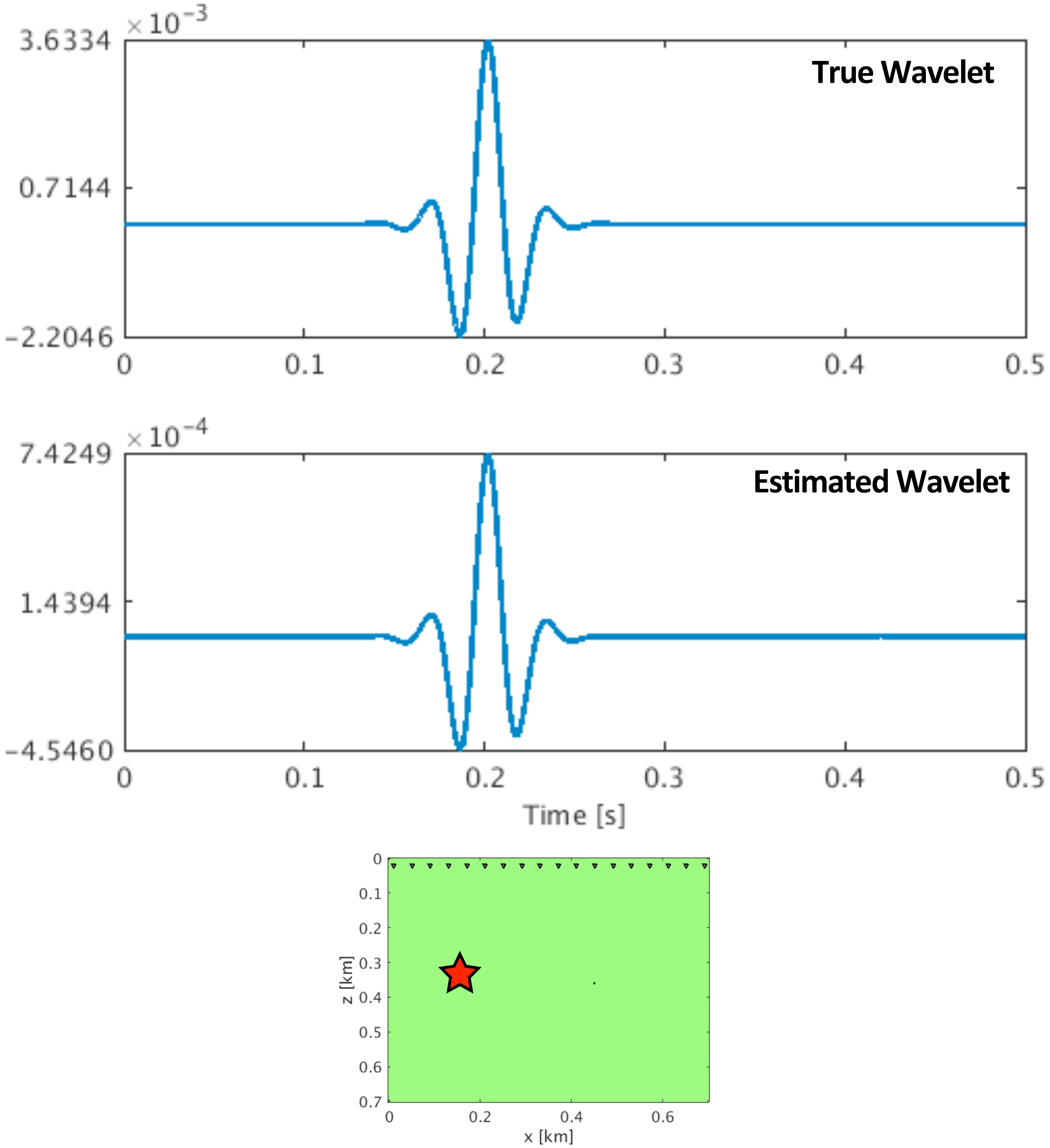


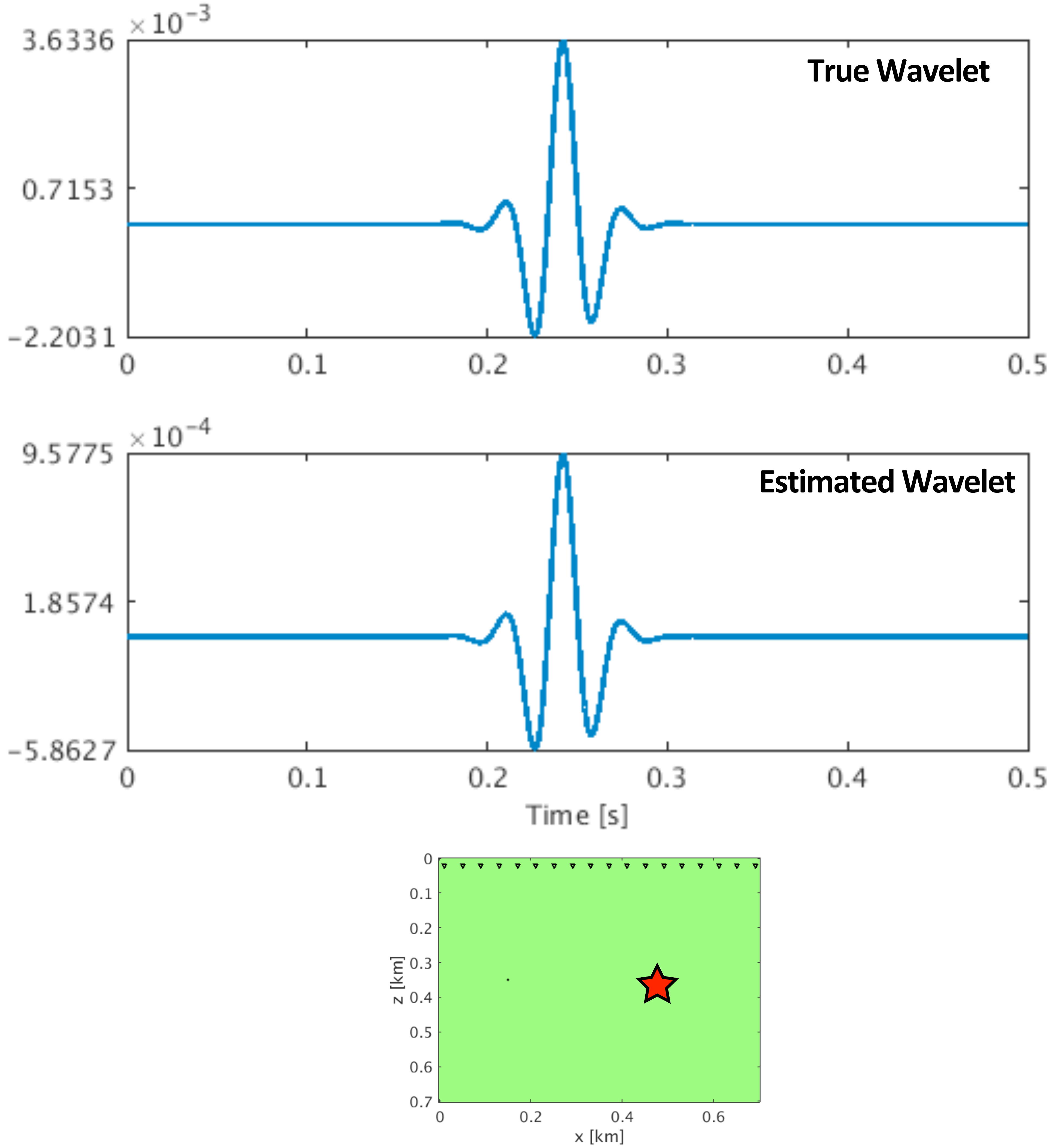
Microseismic data



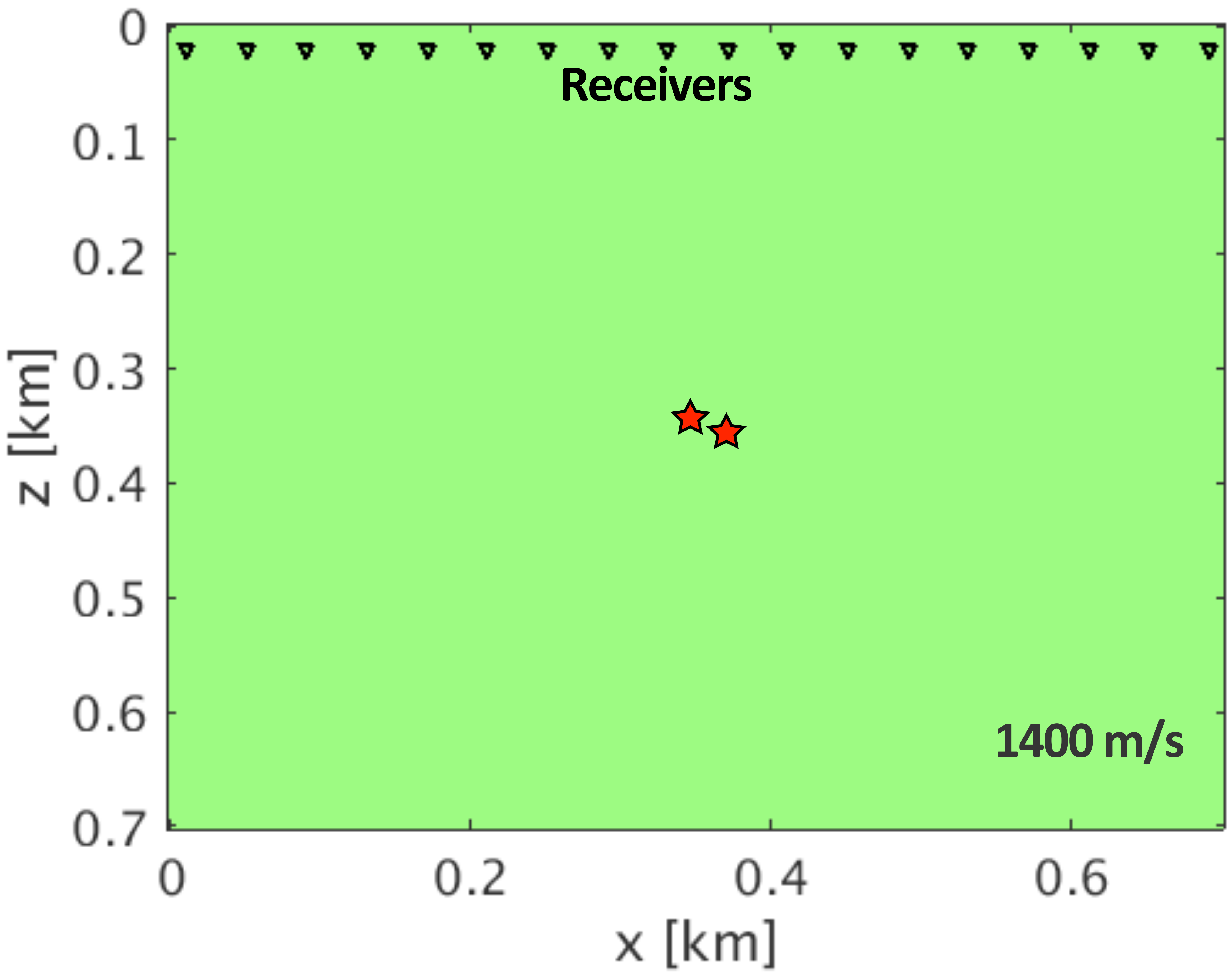
Estimated location

Estimated wavelet

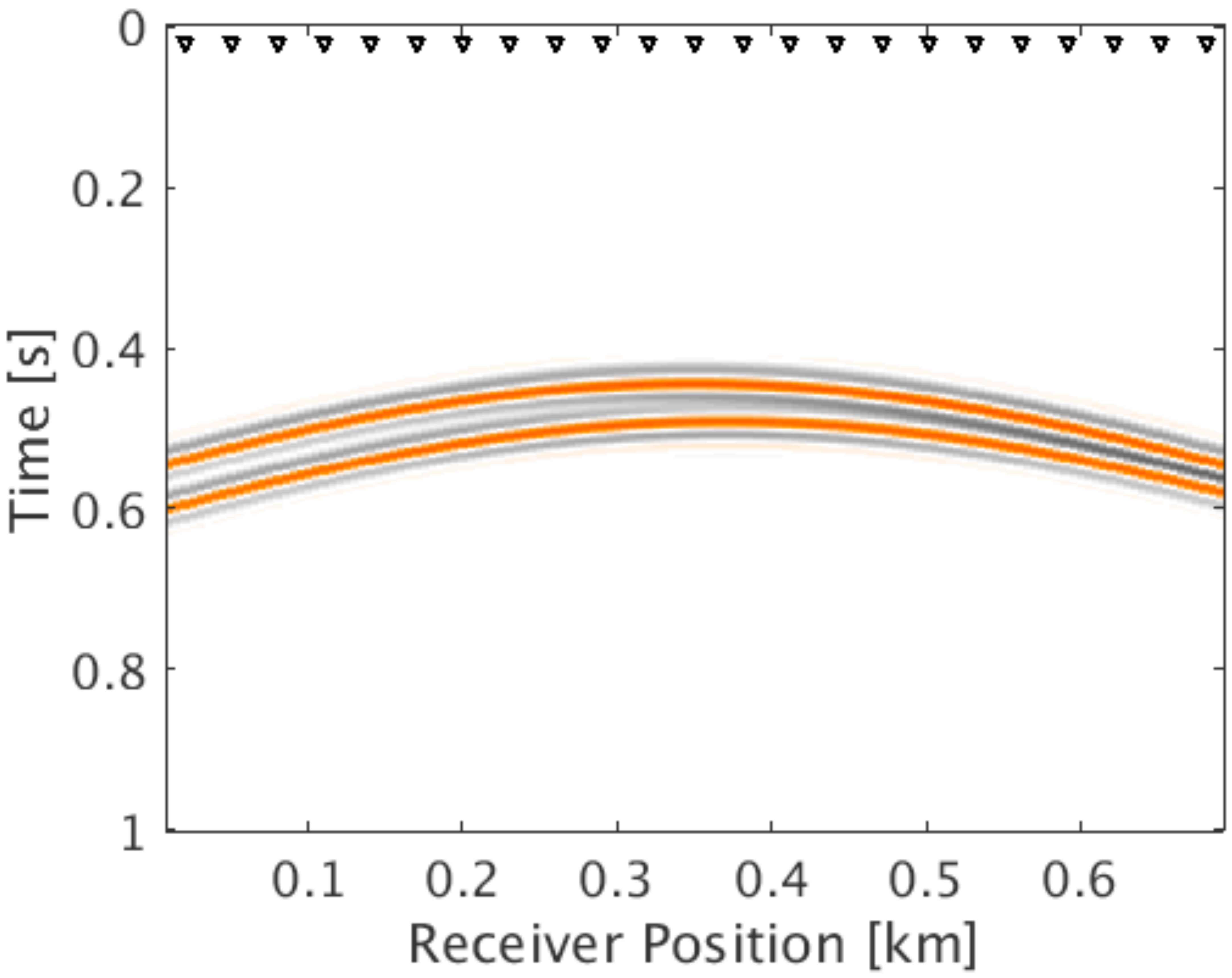




What happens when sources are very close?

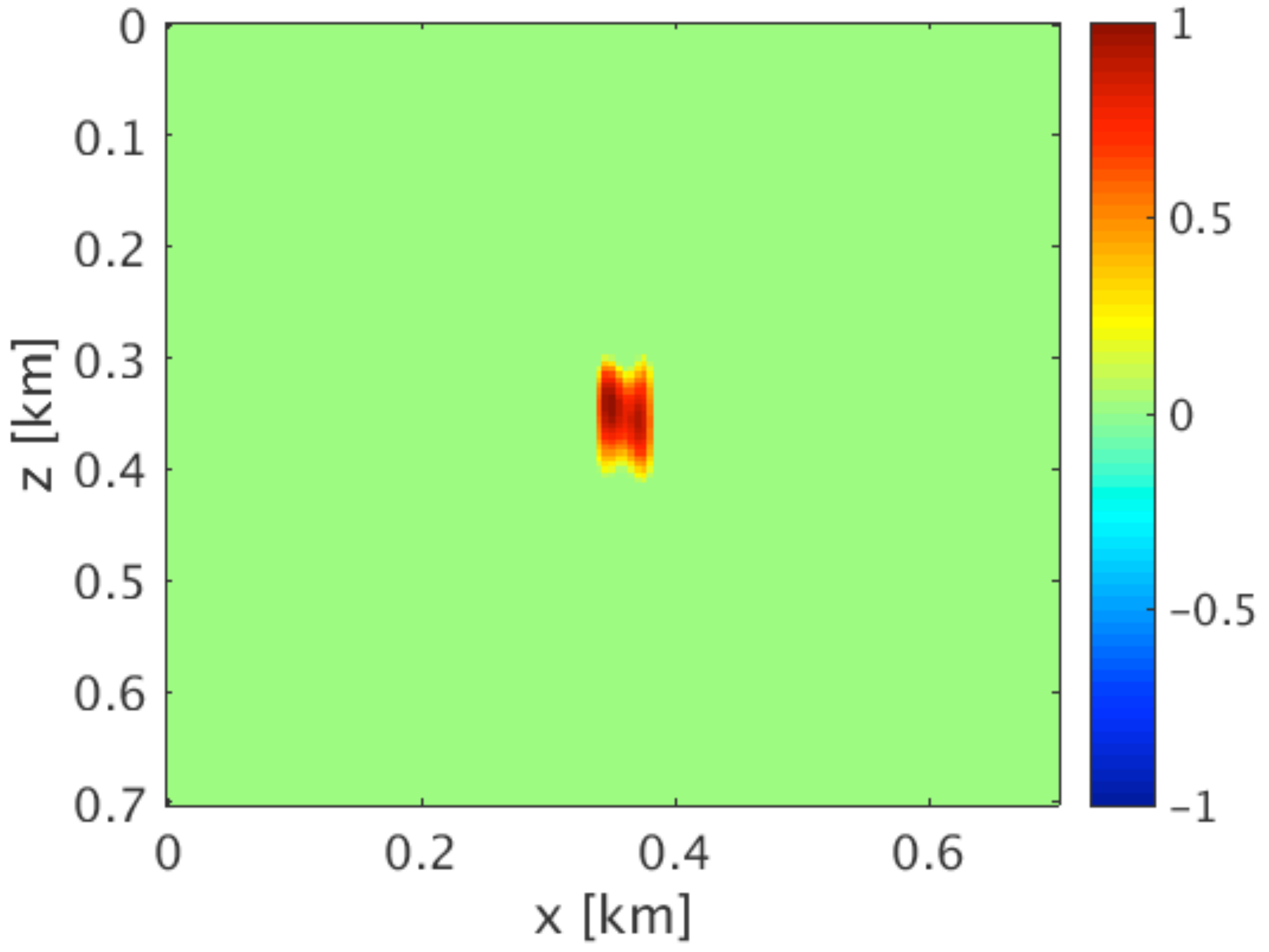


Acquisition setup



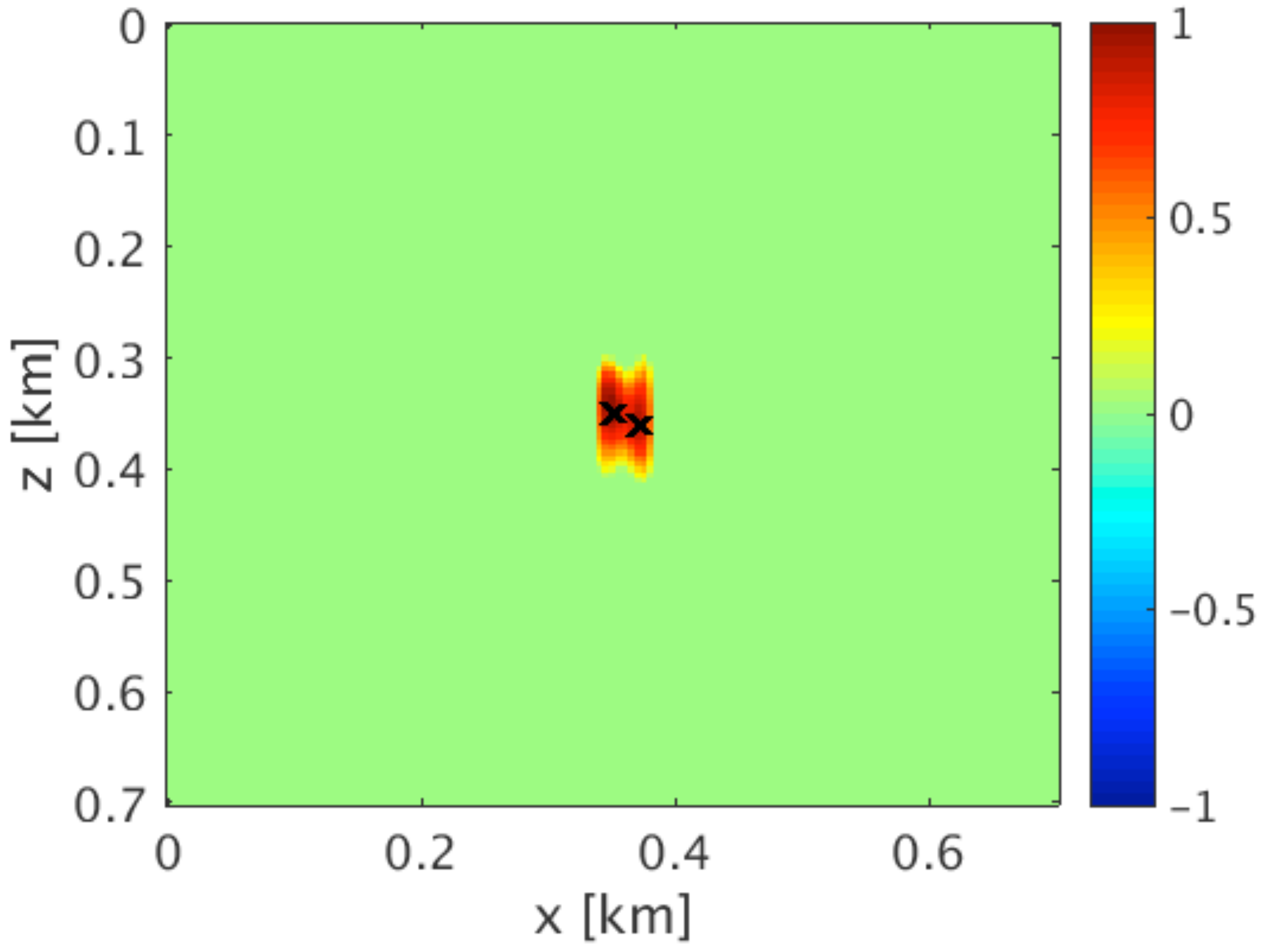
Microseismic data

Estimated location after 150 iterations



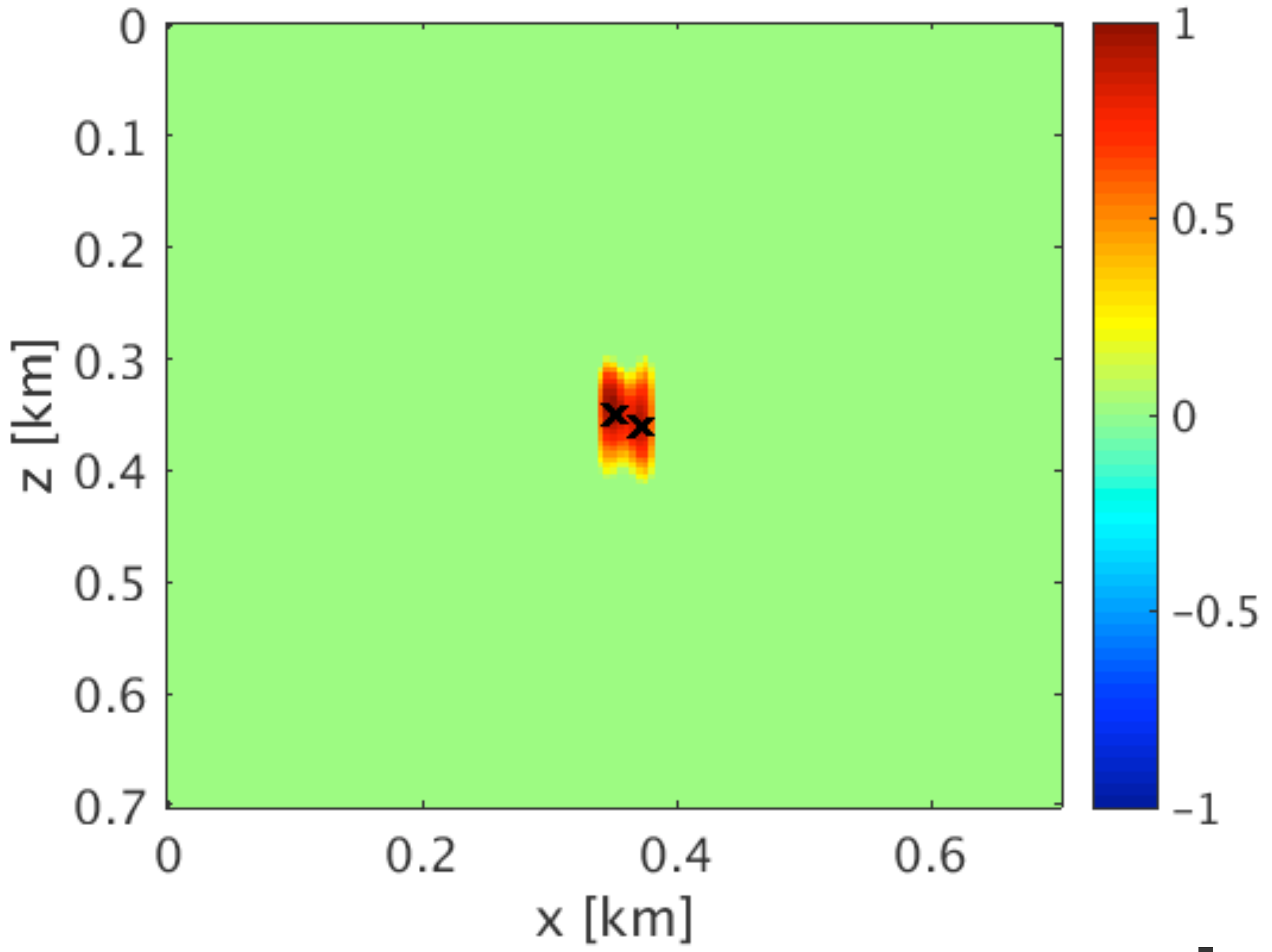
Estimated location, μ = 8e-4

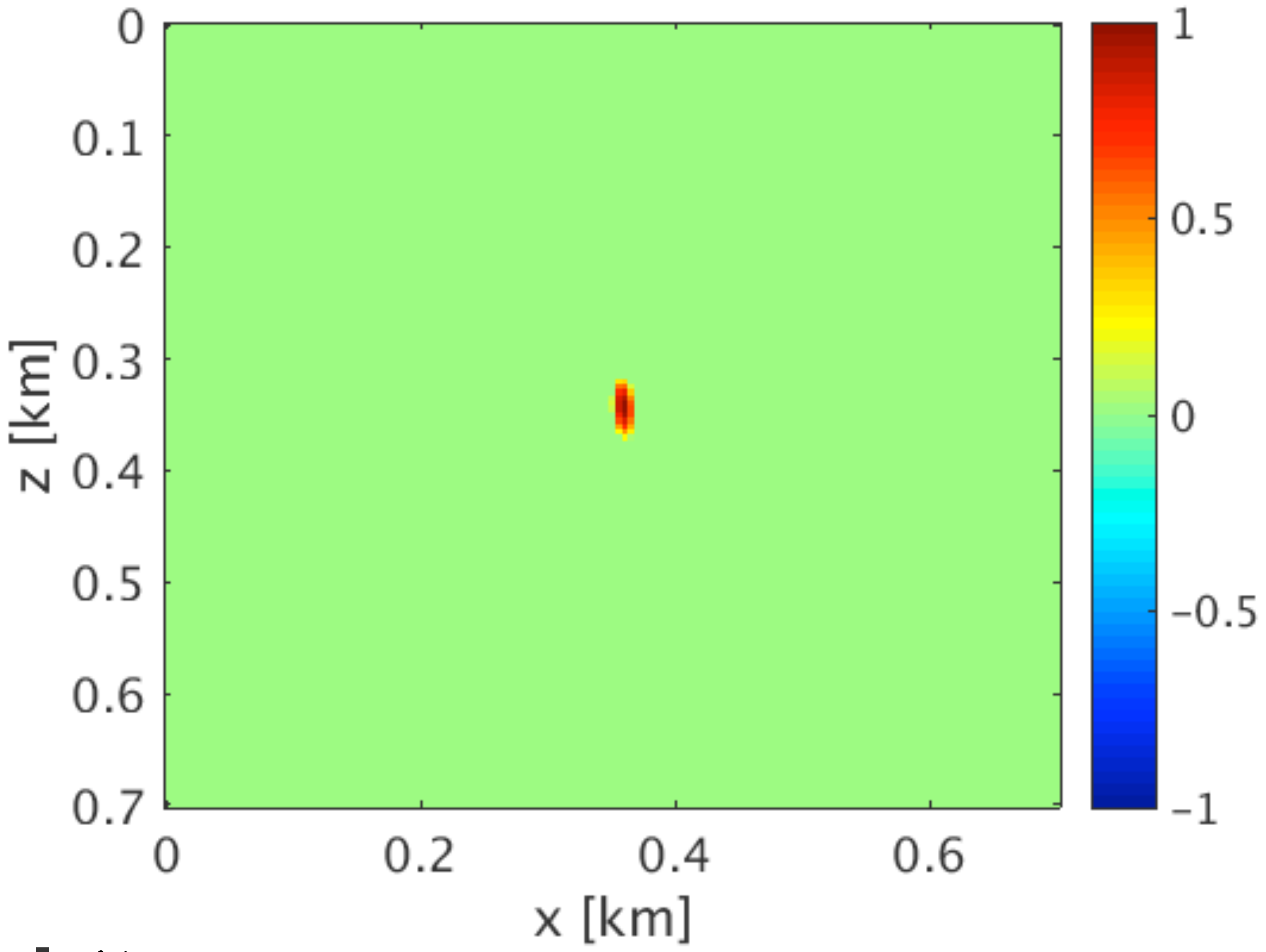
Estimated location after 150 iterations



Estimated location, $\mu = 8e\text{-}4$

Estimated location after 150 iterations





Estimated location, μ= 8e-4

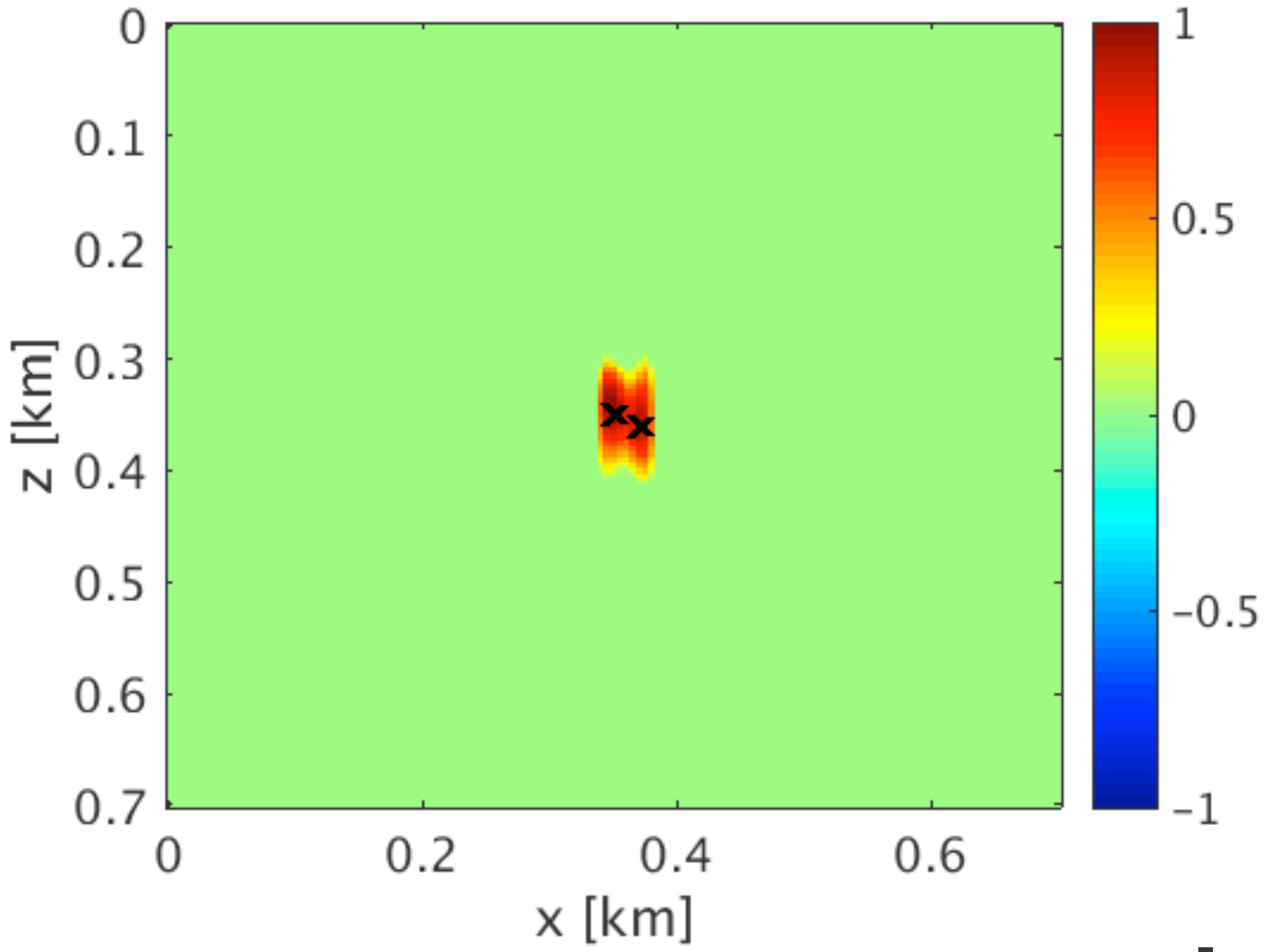
Increased μ

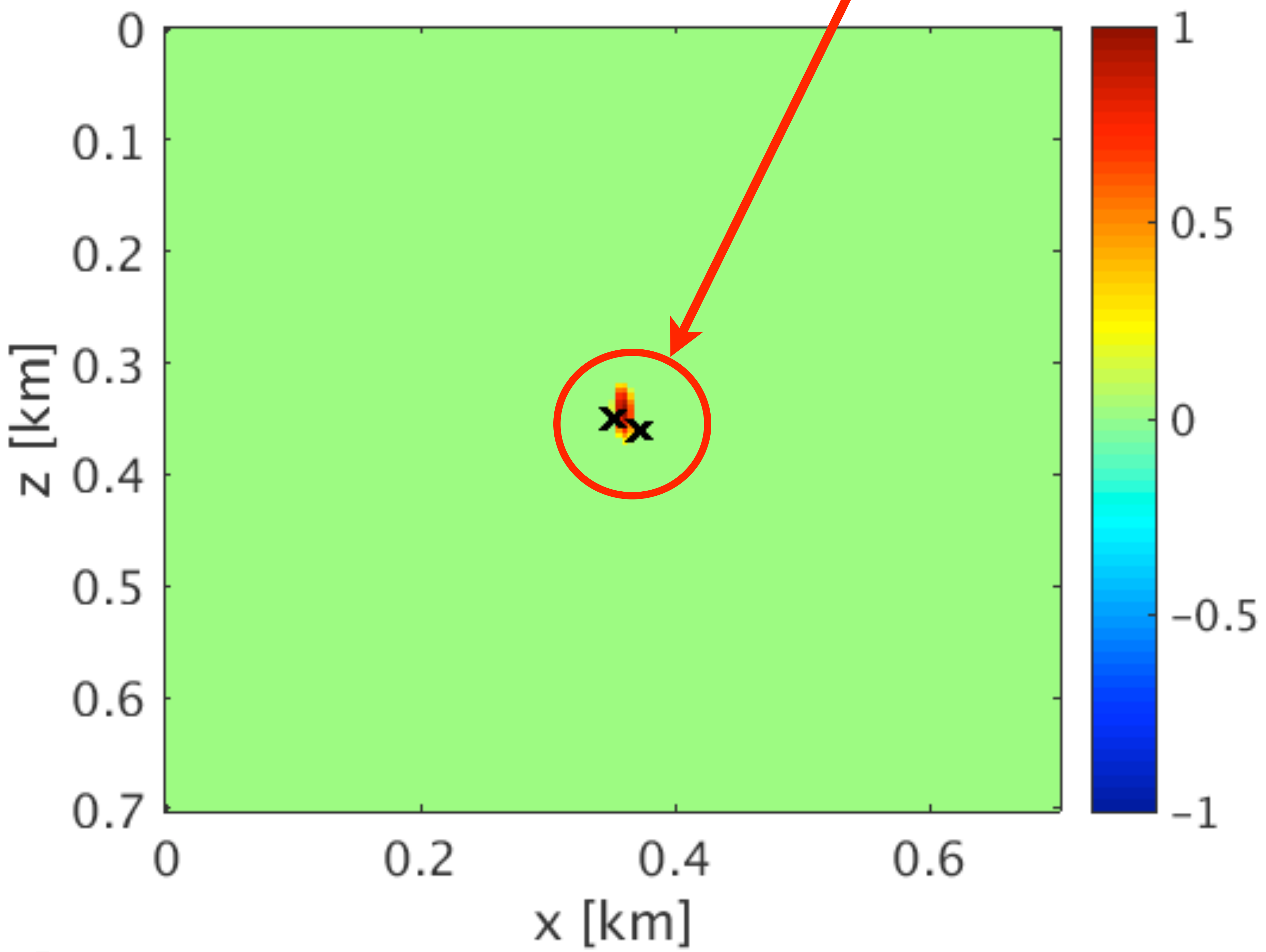
For high resolution

Estimated location, μ = 8e-3

Estimated location after 150 iterations

Fails to resolve due to slow convergence



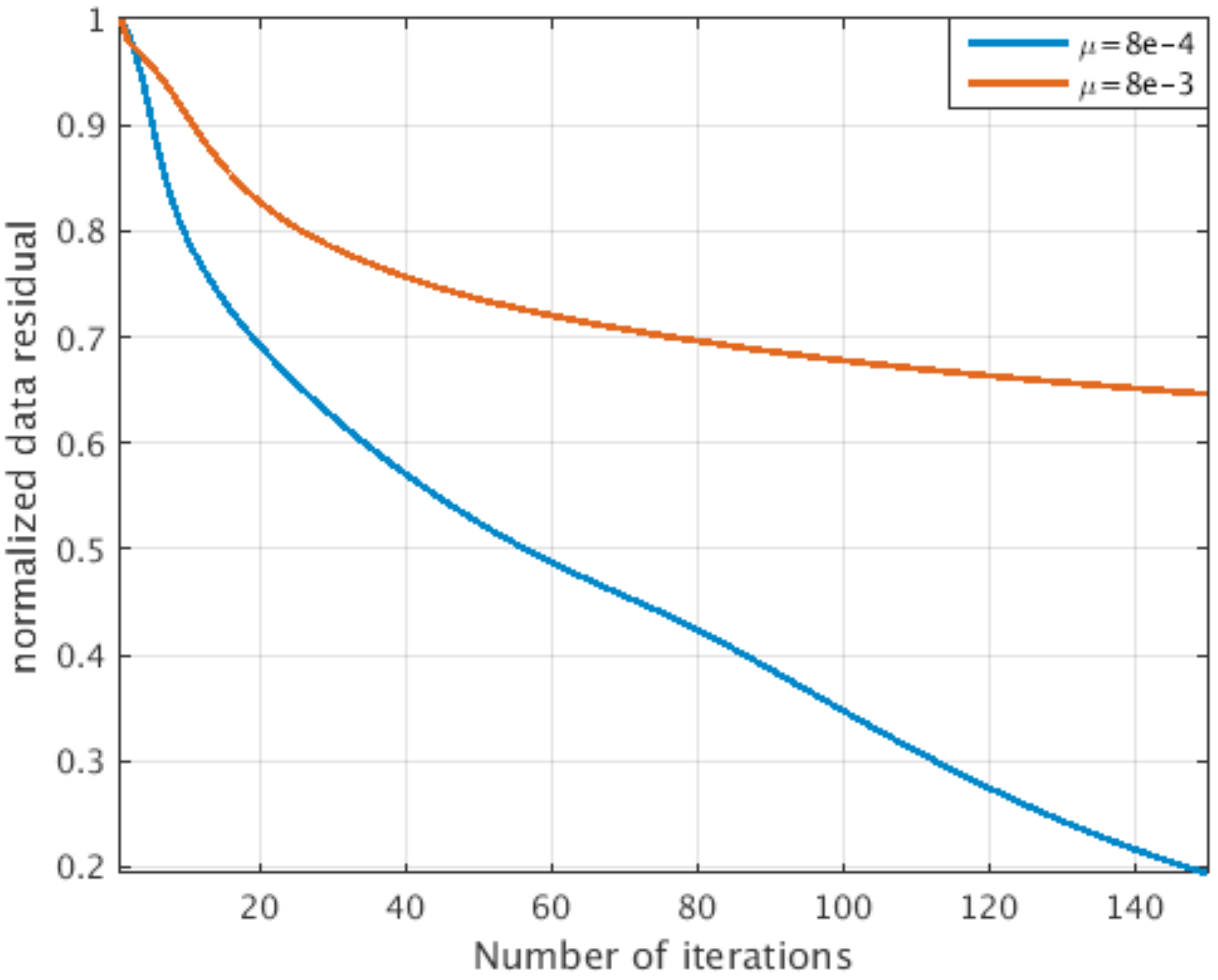


Increased μ

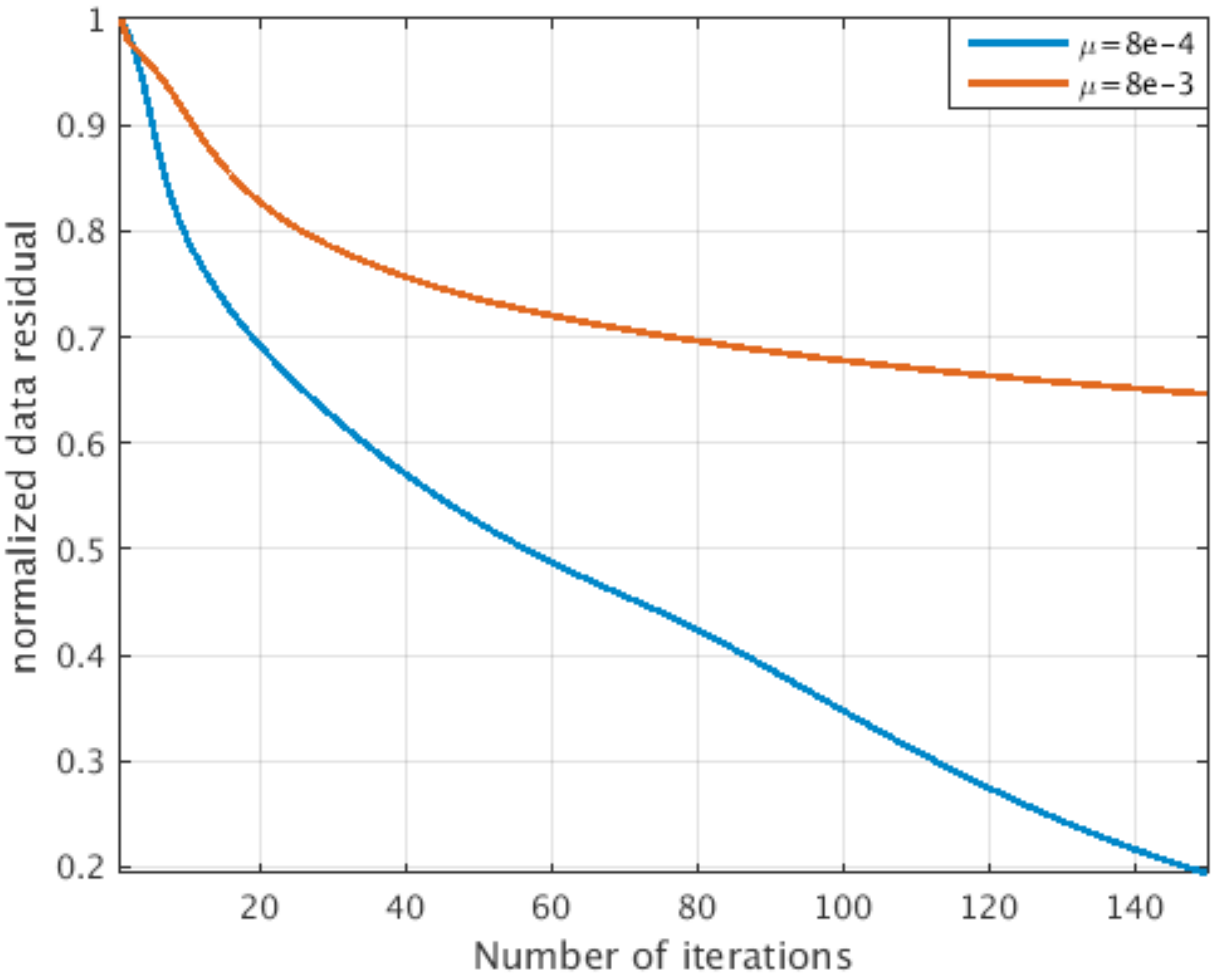
Estimated location, μ = 8e-4 → **Estimated location, μ = 8e-3**

For high resolution

Convergence comparison

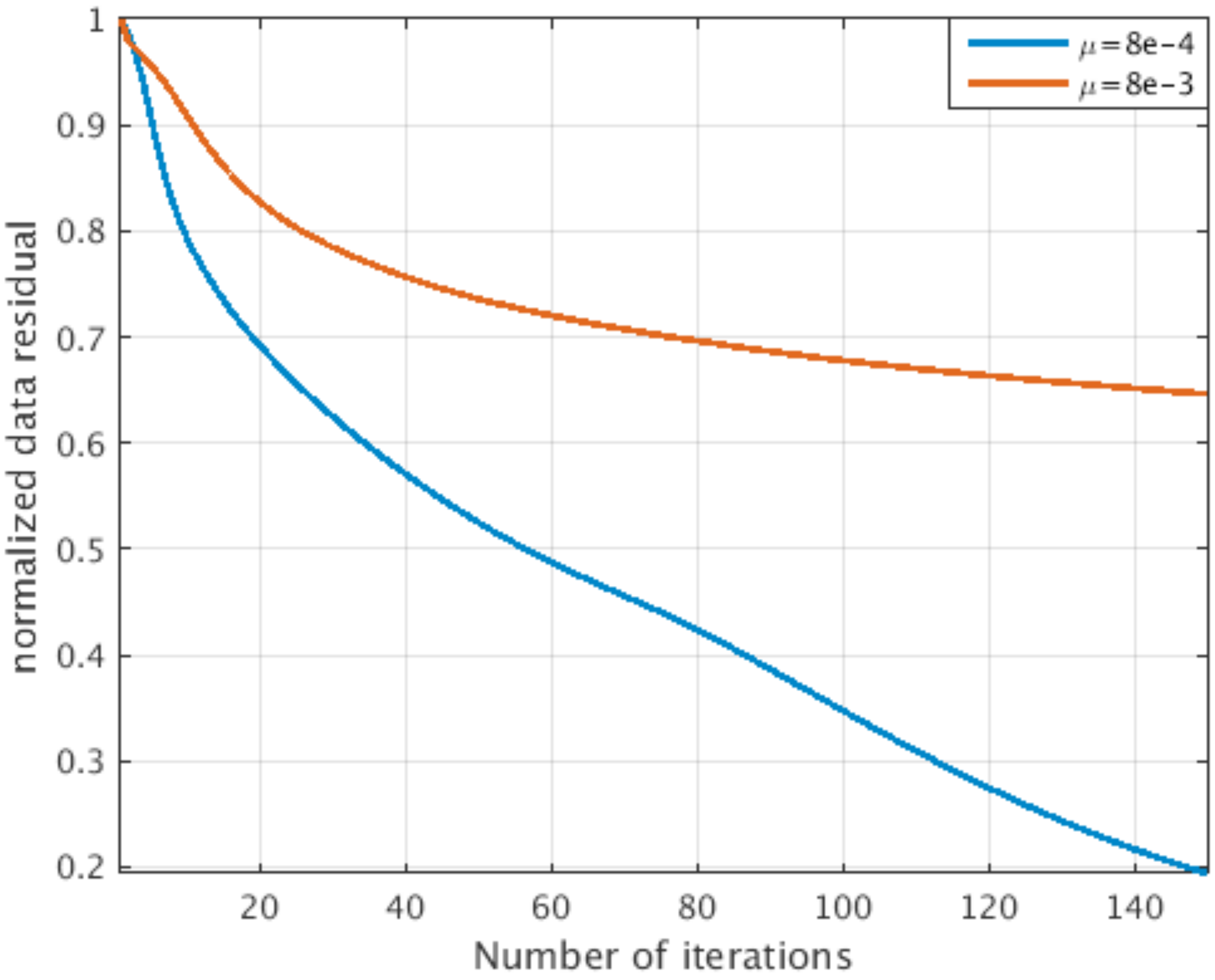


Convergence comparison



Motivation for locating closely spaced sources: for accurate fracture mapping

Convergence comparison



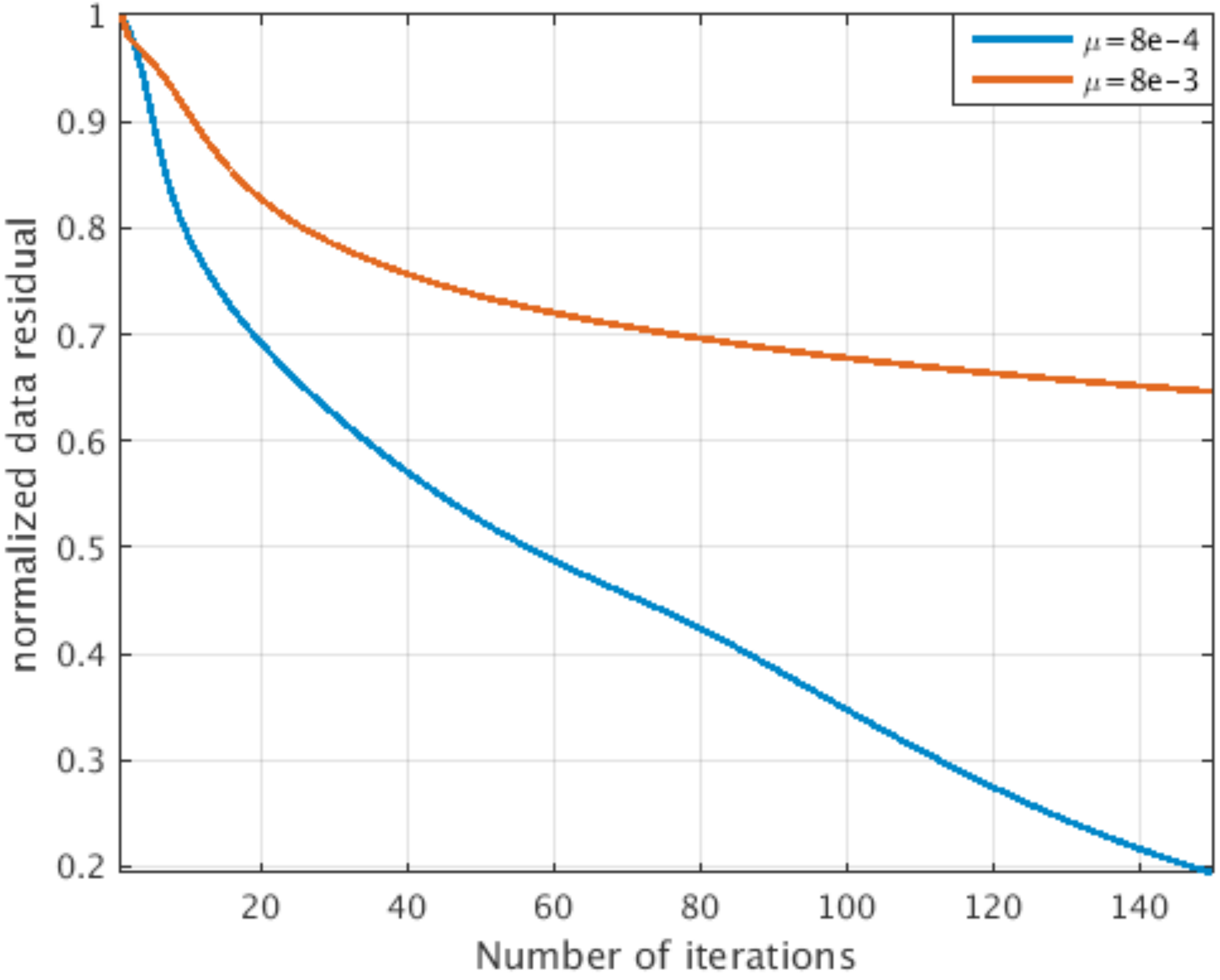
Motivation for locating closely spaced sources: for accurate fracture mapping

Challenges with Linearized Bregman algorithm:

- need higher value of μ to resolve closely spaces sources
- higher values of μ needs more iterations

[Huang et al.,'13; Yin ,'10]

Convergence comparison



Motivation for locating closely spaced sources: for accurate fracture mapping

Challenges with Linearized Bregman algorithm:

- need higher value of μ to resolve closely spaces sources
- higher values of μ needs more iterations

Acceleration with quasi-Newton:

- Linearized Bregman algorithm is equivalent to solving dual problem by gradient descent
- we accelerate the dual problem using quasi-Newton

SLIM

Acceleration with quasi-Newton: Algorithm

1. **Data d, slowness square m, number of iterations** k //Input
2. **Initialize dual variable** $\mathbf{y} = 10^{-3}\mathbf{d}$
3. $\hat{\mathbf{y}} = \text{L-BFGS}(f(\mathbf{y}), g(\mathbf{y}), \mathbf{y}, k)$ //Dual solution

where $f(\mathbf{y}) = \Psi(\mathbf{y}) - \epsilon\|\mathbf{y}\|_2$ //L-BFGS objective

and $g(\mathbf{y}) = \Psi'(\mathbf{y}) - \epsilon\mathbf{y}/\|\mathbf{y}\|_2$ //L-BFGS gradient

4. $\hat{\mathbf{Q}} = \text{Prox}_{\mu\ell_{2,1}}(\mu\mathcal{F}[\mathbf{m}]^\top(\hat{\mathbf{y}}))$ //Primal solution
5. $\mathbf{I}(\mathbf{x}) = \sum_t |\hat{\mathbf{Q}}(\mathbf{x}, t)|$ //Intensity plot

*where $\Psi(\mathbf{y}) = \min_{\mathbf{Q}} \quad \|\mathbf{Q}\|_{2,1} + \frac{1}{2\mu}\|\mathbf{Q}\|_F - \mathbf{y}^\top(\mathcal{F}[\mathbf{m}](\mathbf{Q}) - \mathbf{d})$

* $\Psi'(\mathbf{y}) = \mathbf{d} - \mathcal{F}[\mathbf{m}](\text{Prox}_{\mu\ell_{2,1}}(\mu\mathcal{F}[\mathbf{m}]^\top(\mathbf{y})))$ is the gradient of $\Psi(\mathbf{y})$

[Huang et al.,’13; Yin ,’10]

SLIM

Acceleration with quasi-Newton: Algorithm

1. Data $\mathbf{d}$, slowness square $\mathbf{m}$, number of iterations k //Input
2. Initialize dual variable $\mathbf{y} = 10^{-3}\mathbf{d}$
3. $\hat{\mathbf{y}} = \text{L-BFGS}(f(\mathbf{y}), g(\mathbf{y}), \mathbf{y}, k)$ //Dual solution

where $f(\mathbf{y}) = \Psi(\mathbf{y}) - \epsilon\|\mathbf{y}\|_2$ //L-BFGS objective

and $g(\mathbf{y}) = \Psi'(\mathbf{y}) - \epsilon\mathbf{y}/\|\mathbf{y}\|_2$ //L-BFGS gradient

4. $\hat{\mathbf{Q}} = \text{Prox}_{\mu\ell_{2,1}}(\mu\mathcal{F}[\mathbf{m}]^\top(\hat{\mathbf{y}}))$ //Primal solution
5. $\mathbf{I}(\mathbf{x}) = \sum_t |\, \hat{\mathbf{Q}}(\mathbf{x}, t) \,|$ //Intensity plot

lives in much smaller space :

- dimensions equals that of observed data
- better approximation of inverse Hessian by storing more and more dual variable updates

*where $\Psi(\mathbf{y}) = \min_{\mathbf{Q}} \quad \|\mathbf{Q}\|_{2,1} + \frac{1}{2\mu}\|\mathbf{Q}\|_F - \mathbf{y}^\top(\mathcal{F}[\mathbf{m}](\mathbf{Q}) - \mathbf{d})$

* $\Psi'(\mathbf{y}) = \mathbf{d} - \mathcal{F}[\mathbf{m}](\text{Prox}_{\mu\ell_{2,1}}(\mu\mathcal{F}[\mathbf{m}]^\top(\mathbf{y})))$ is the gradient of $\Psi(\mathbf{y})$

[Herrmann et al.,'09;]

Further acceleration w/ 2D Preconditioning

Each iteration of L-BFGS requires solving at least one

- wave equation and
- its adjoint

[Herrmann et al.,’09;]

Further acceleration w/ 2D Preconditioning

Each iteration of L-BFGS requires solving at least one

- wave equation and
- its adjoint

Requires further reduction in the total number of iterations due to:

- the problem size and
- computational costs

SLIM

Further acceleration w/ 2D Preconditioning

Each iteration of L-BFGS requires solving at least one

- wave equation and
- its adjoint

Requires further reduction in the total number of iterations due to:

- the problem size and
- computational costs

Left preconditioner:

- reduces the condition number of 2D forward modeling operator $\mathcal{F}$
- accelerates the convergence

[Herrmann et al.,'09; Song et al. ,'95]

SLIM

Further acceleration w/ 2D Preconditioning

In 2D, a point source implicitly assumes

- a line source
- extending infinitely in the out of plain direction

This causes wavefields to have:

- amplitude and
- phase differ from the wavefields of a true point source

We introduce:

- a symmetric half differentiation correction along time
- corrects for the amplitude and phase of 2D wavefield
- which acts as a left preconditioner

SLIM

Modified problem w/ 2D Preconditioning

$$\underset{\mathbf{Q}}{\text{minimize}} \ \ \|\mathbf{Q}\|_{2,1} + \frac{1}{2\mu}\|\mathbf{Q}\|_F^2$$
$$\text{subject to } \|\mathcal{M}_L\mathcal{F}[\mathbf{m}](\mathbf{Q}) - \mathcal{M}_L\mathbf{d}\|_2 \leq \gamma$$

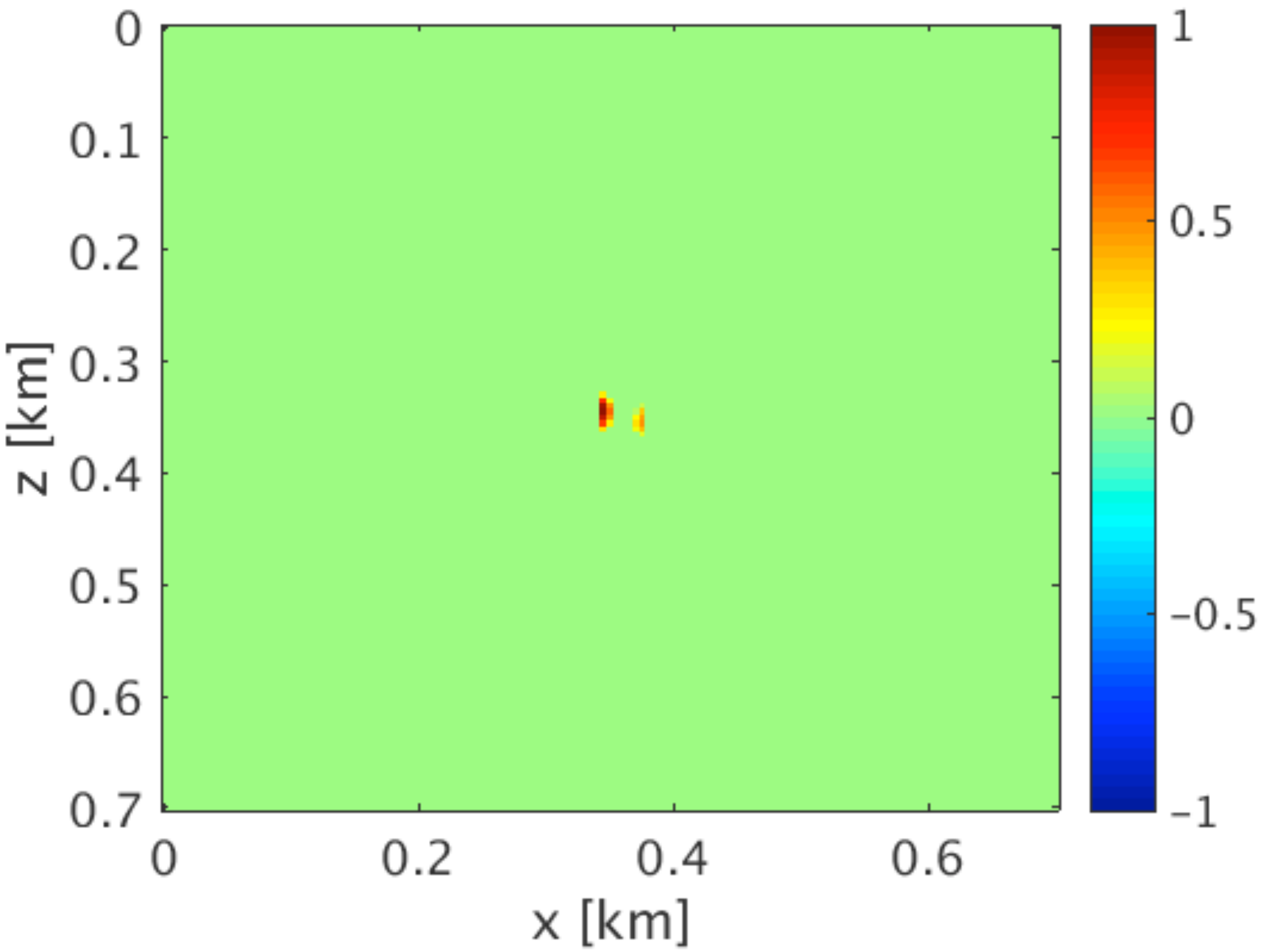
*with $\mathcal{M}_L := \partial_{|t|}^{1/2}$ is the half differentiation correction

*where $\partial_{|t|}^{1/2} = \mathbf{F}^{-1}|\omega|^{1/2}\mathbf{F}$

*$\mathbf{F}$ is the Fourier transform and ω is the frequency

*γ is the noise level

Result for two close sources

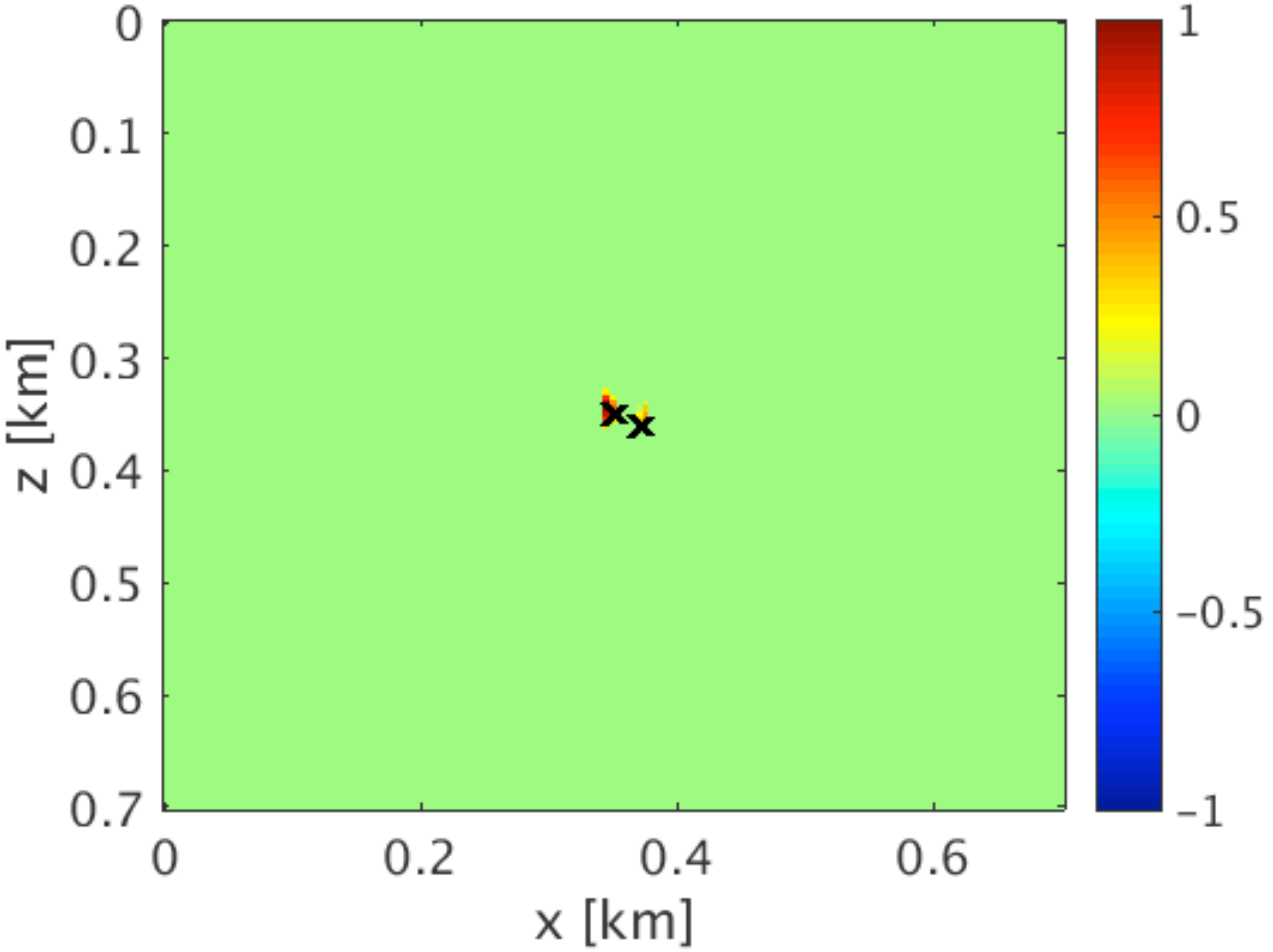


Estimated location

With L-BFGS and 2D preconditioner

10 iterations

Result for two close sources

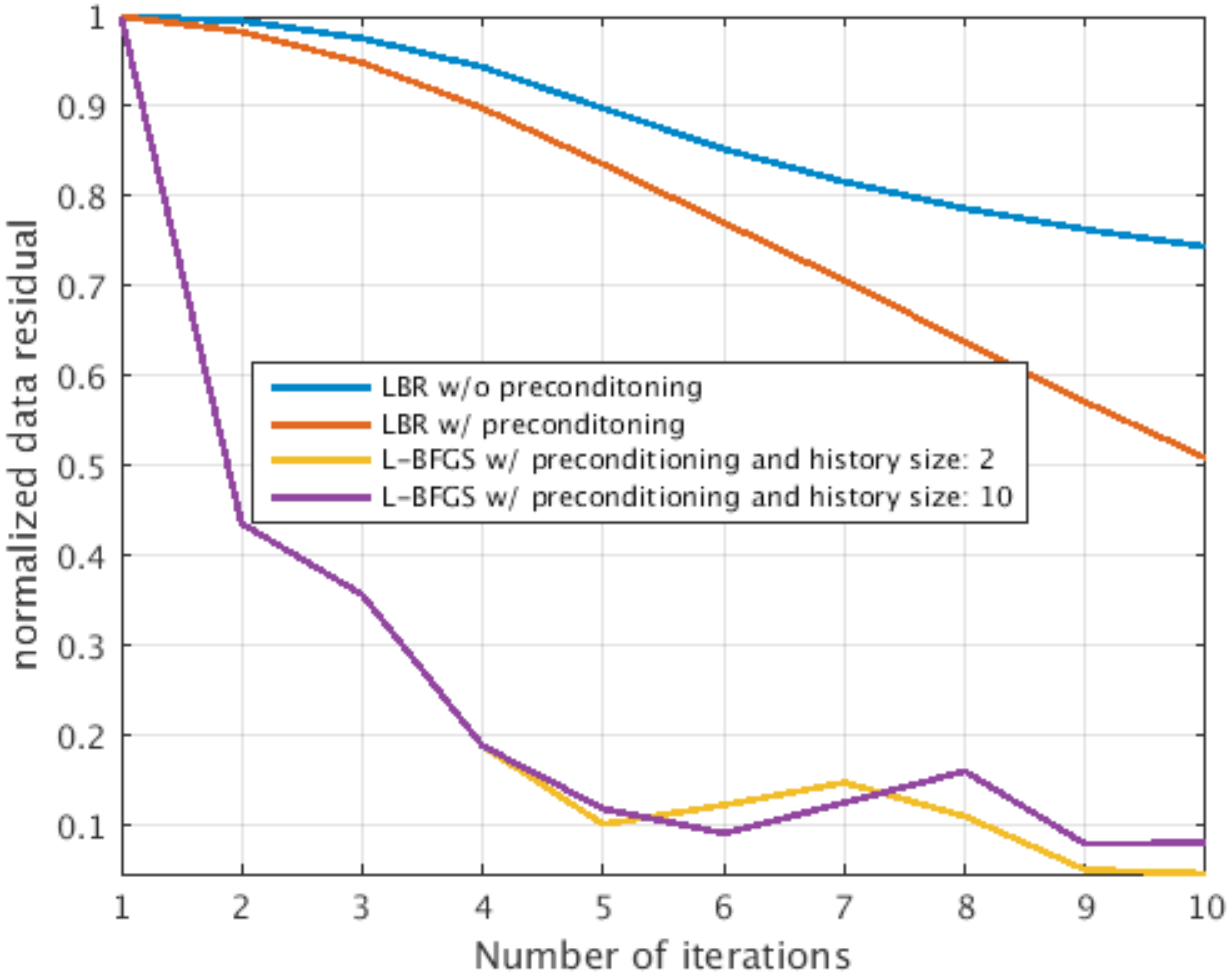


Estimated location

With L-BFGS and 2D preconditioner

10 iterations

Convergence comparison: LBR vs L-BFGS



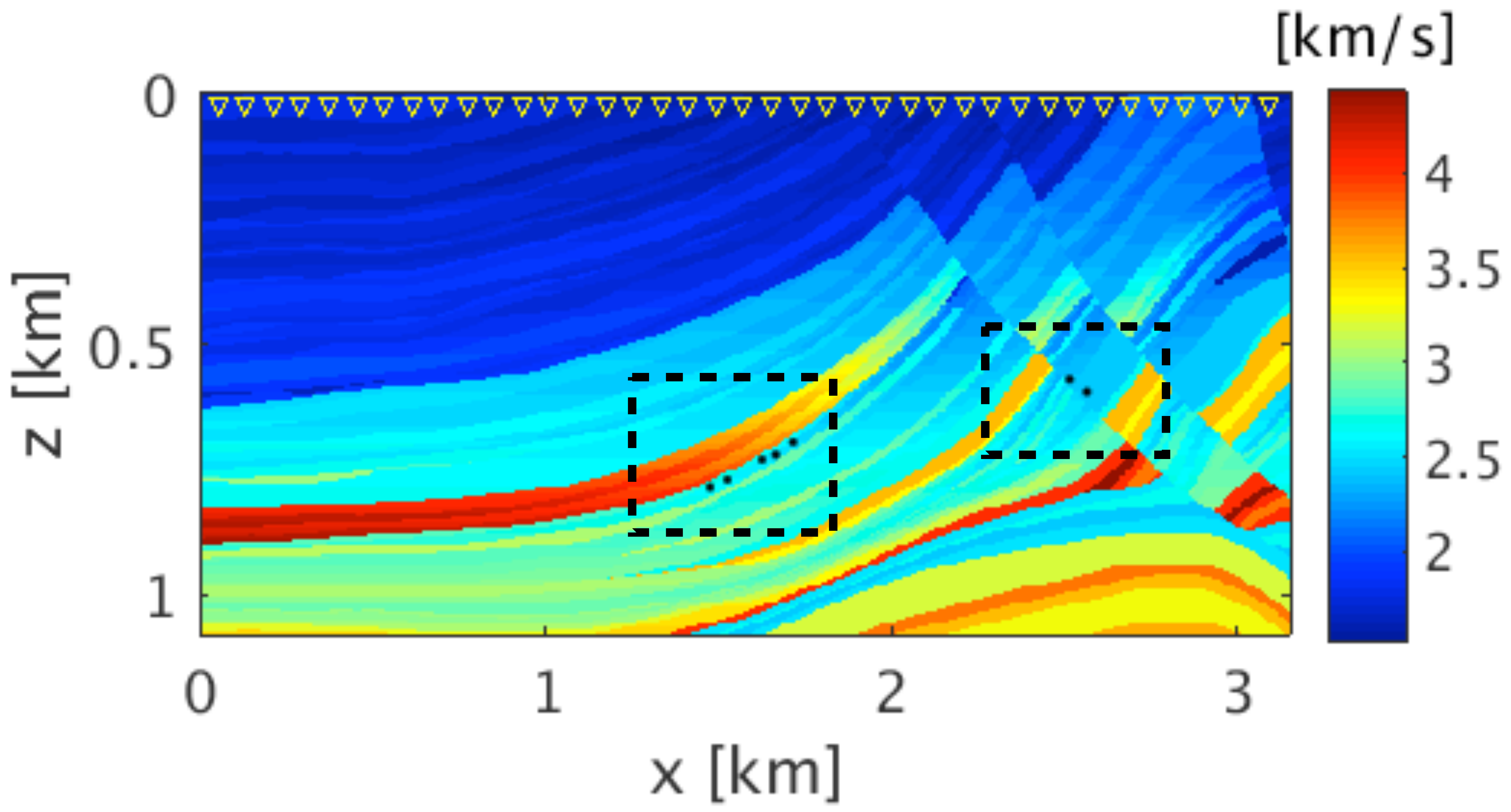
Convergence comparison

- Using same value of μ

Improvement in convergence with

- Dual formulation and
- 2D Preconditioning

Numerical Experiment: Marmousi model

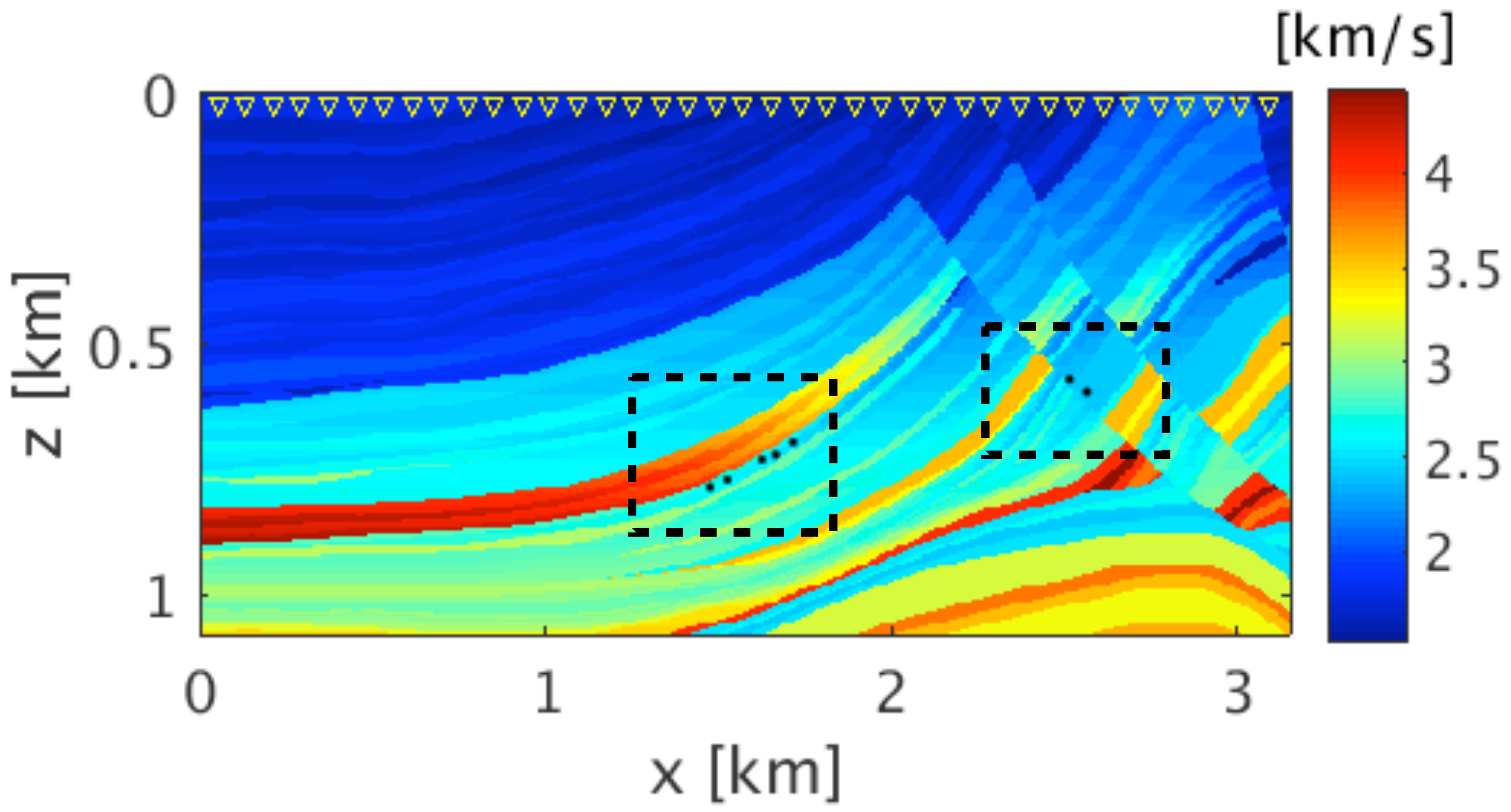


Marmousi model

Modeling information:

Model size: 3.15 km x 1.08 km
Grid spacing: 5 m
Total number of sources: 7
Peak frequency : 22 Hz, 25 Hz & 30 Hz
Receiver spacing: 10m
Receiver depth: 20m
Sampling interval: 0.5 ms
Recording length: 1 s
Free surface: No

Numerical Experiment: Marmousi model



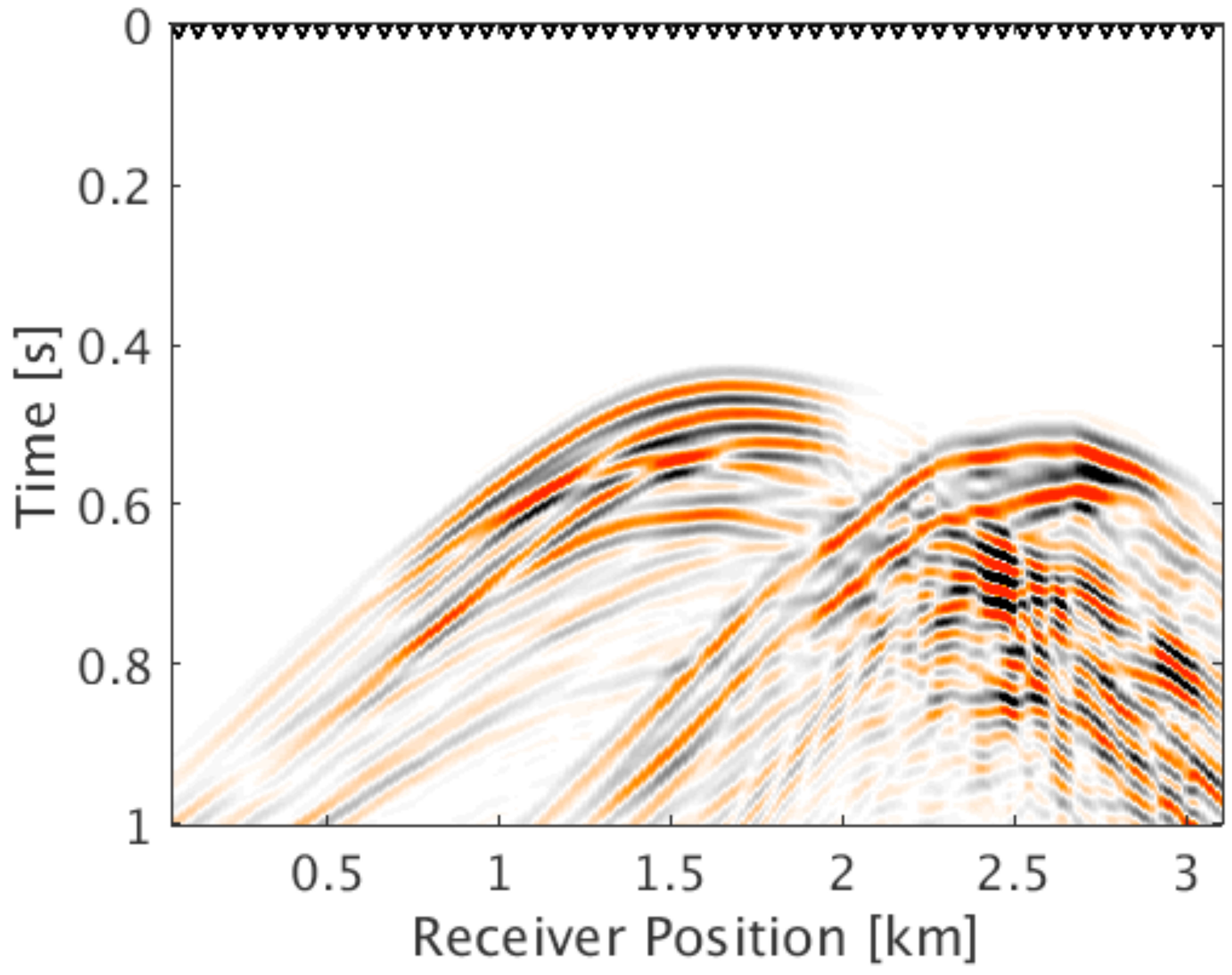
Marmousi model

Modeling information:

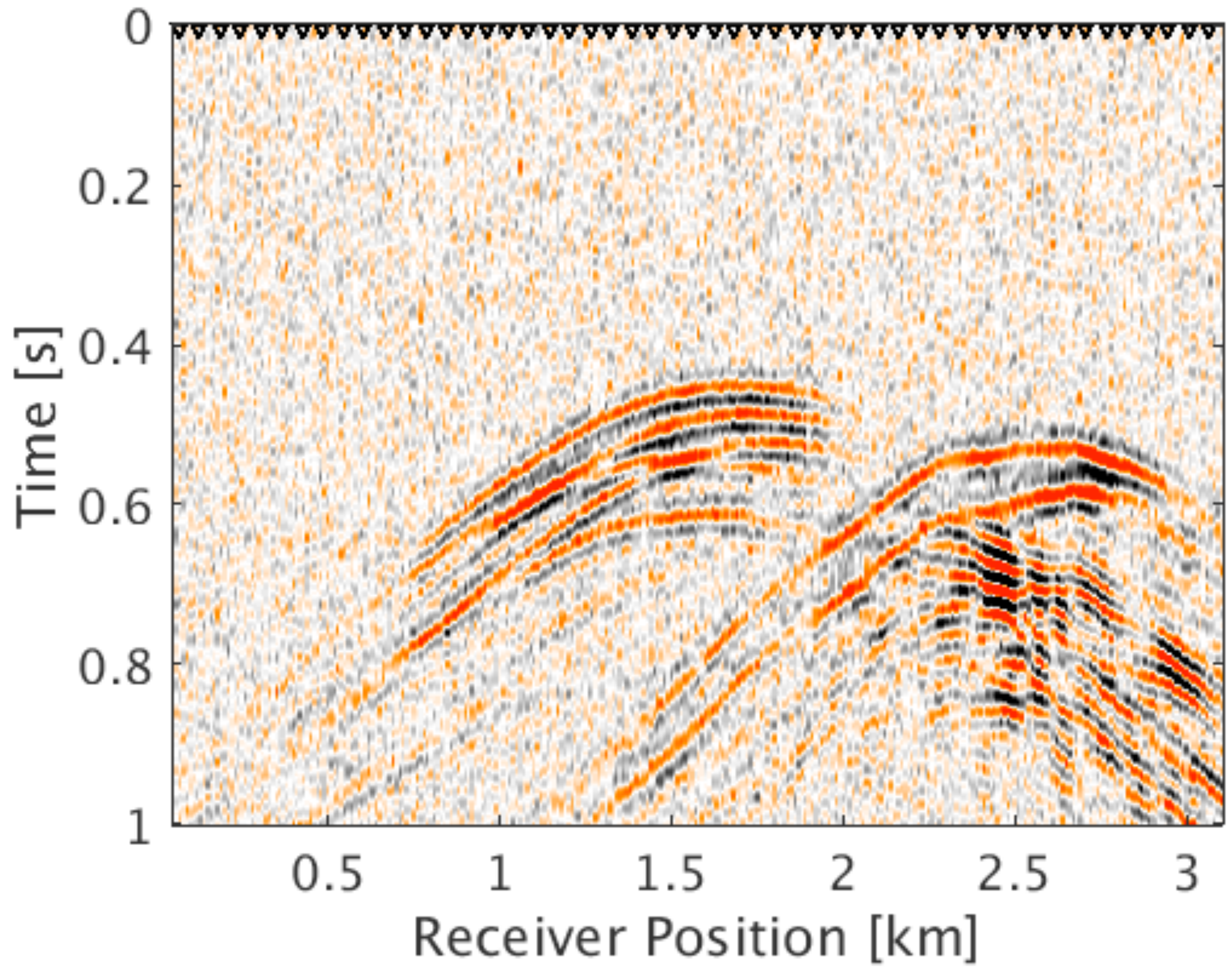
Model size: 3.15 km x 1.08 km
Grid spacing: 5 m
Total number of sources: 7
Peak frequency : 22 Hz, 25 Hz & 30 Hz
Receiver spacing: 10m
Receiver depth: 20m
Sampling interval: 0.5 ms
Recording length: 1 s
Free surface: No

Adjacent sources are located within half a wavelength with overlapping source-time functions

Data



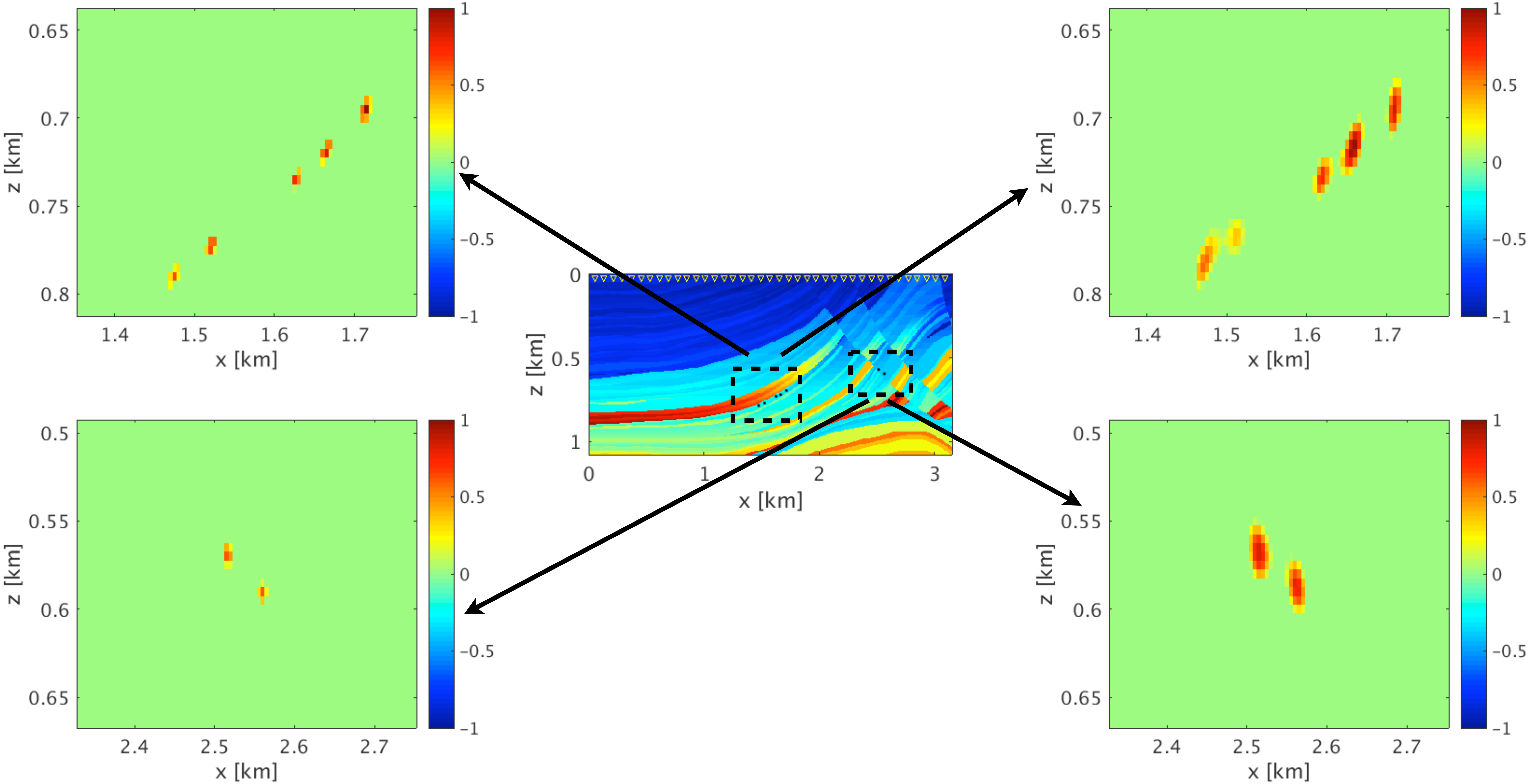
Noise free microseismic data



Noisy Microseismic data, SNR = 2.9

Estimated source location in 10 iterations

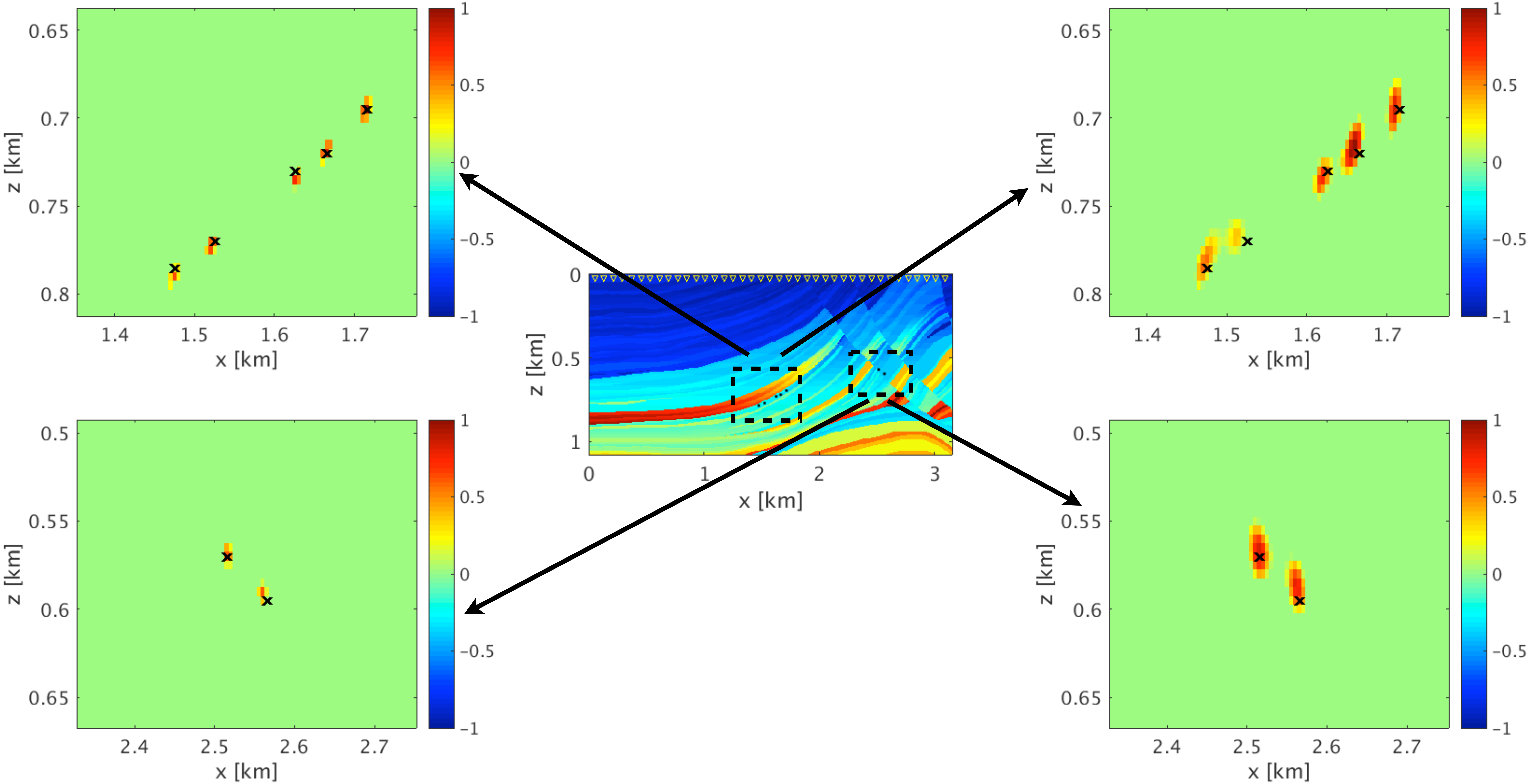
w/noise free data and true velocity model **w/noisy data and smooth velocity model**



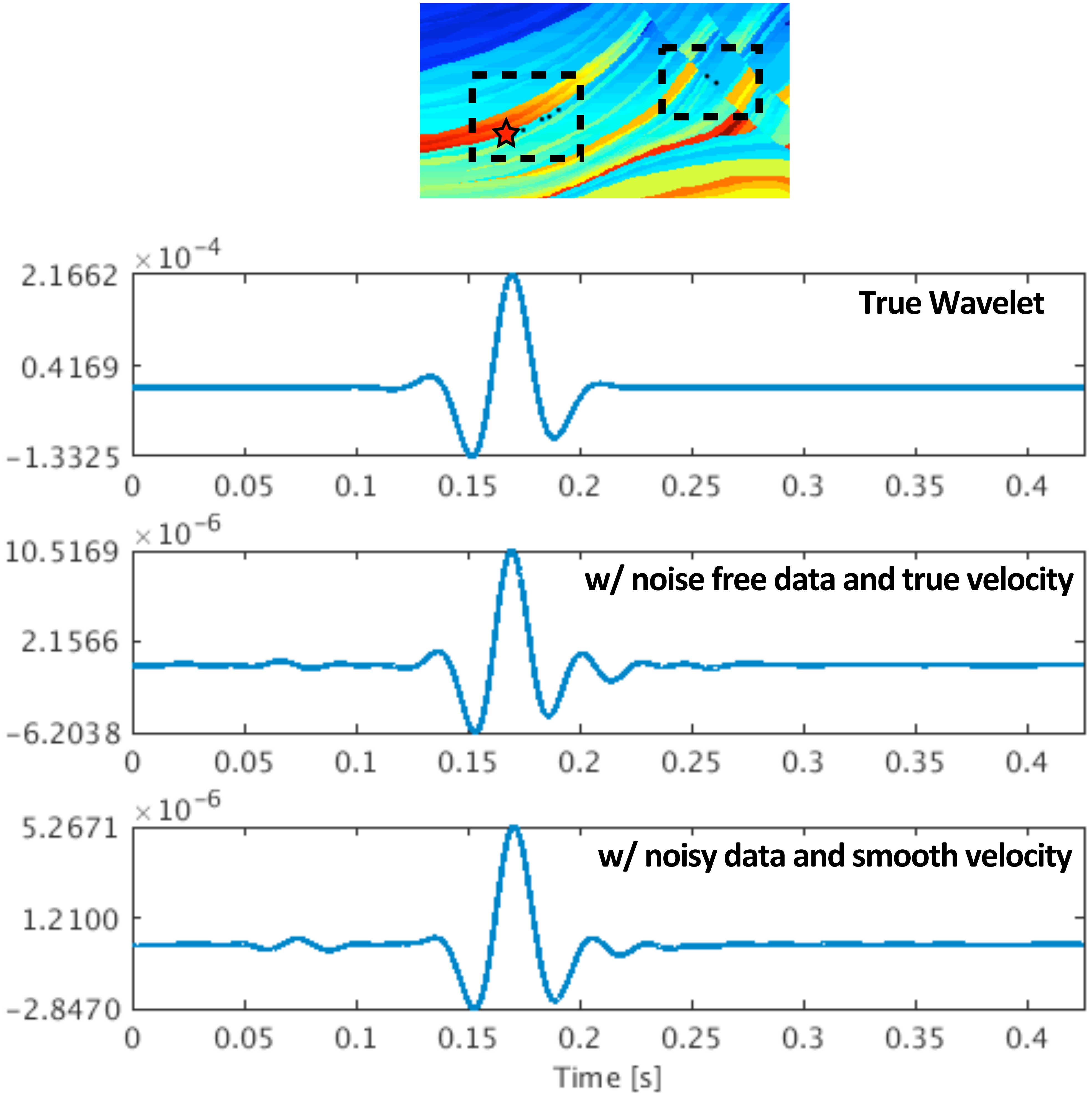
Estimated source location in 10 iterations

w/noise free data and true velocity model

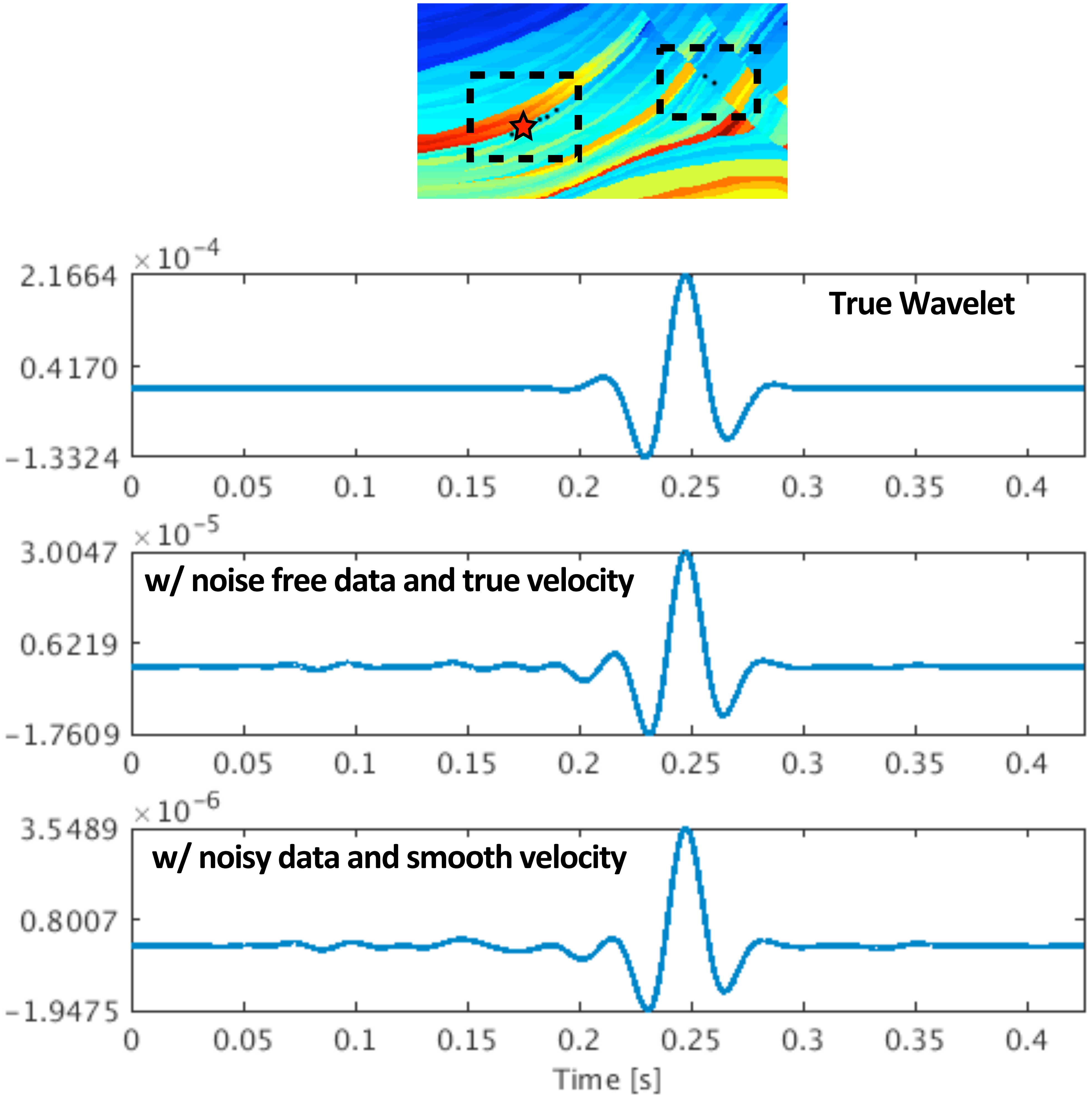
w/noisy data and smooth velocity model



Wavelet comparison

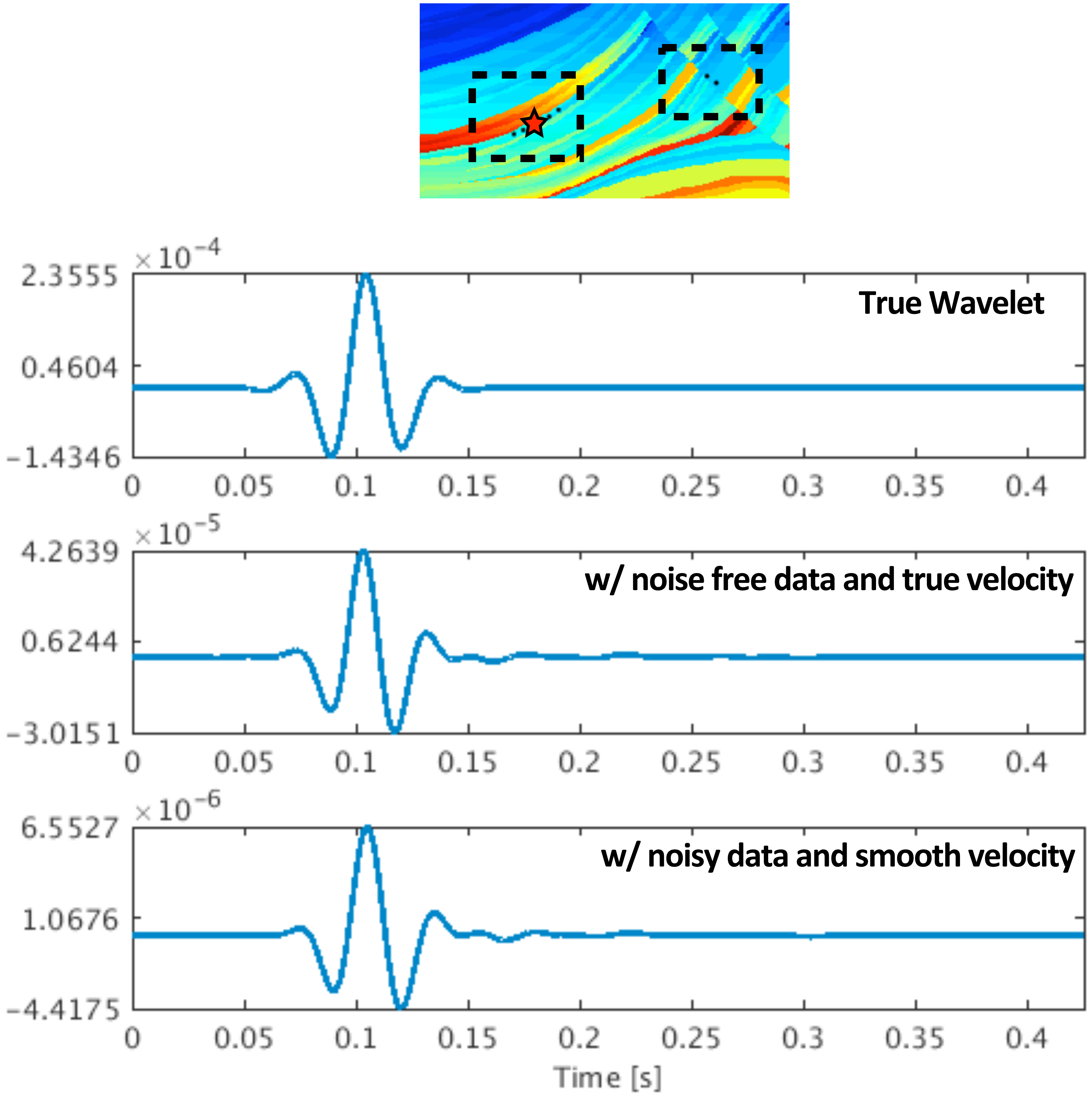


Peak frequency: 25 Hz

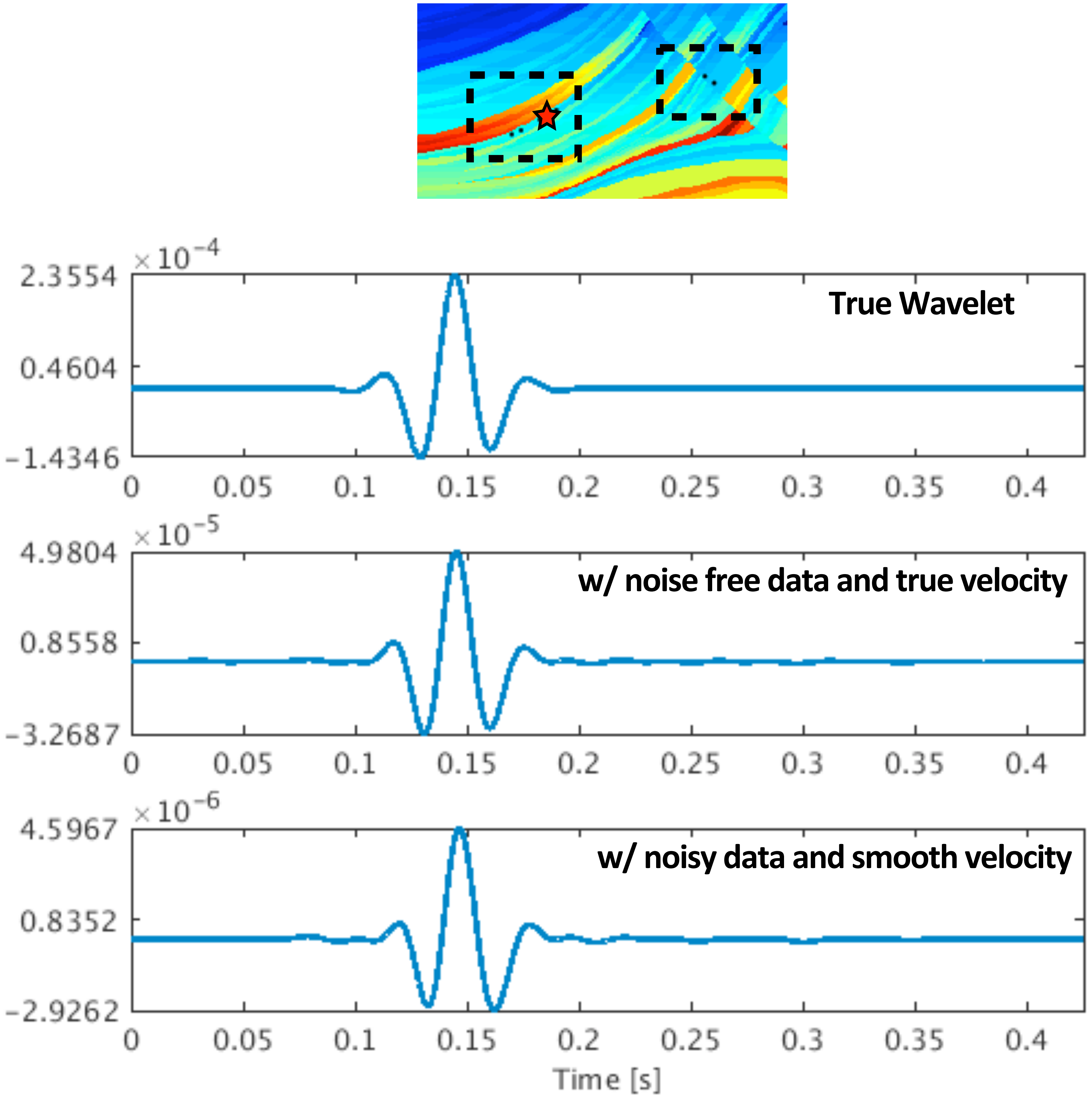


Peak frequency: 25 Hz

Wavelet comparison

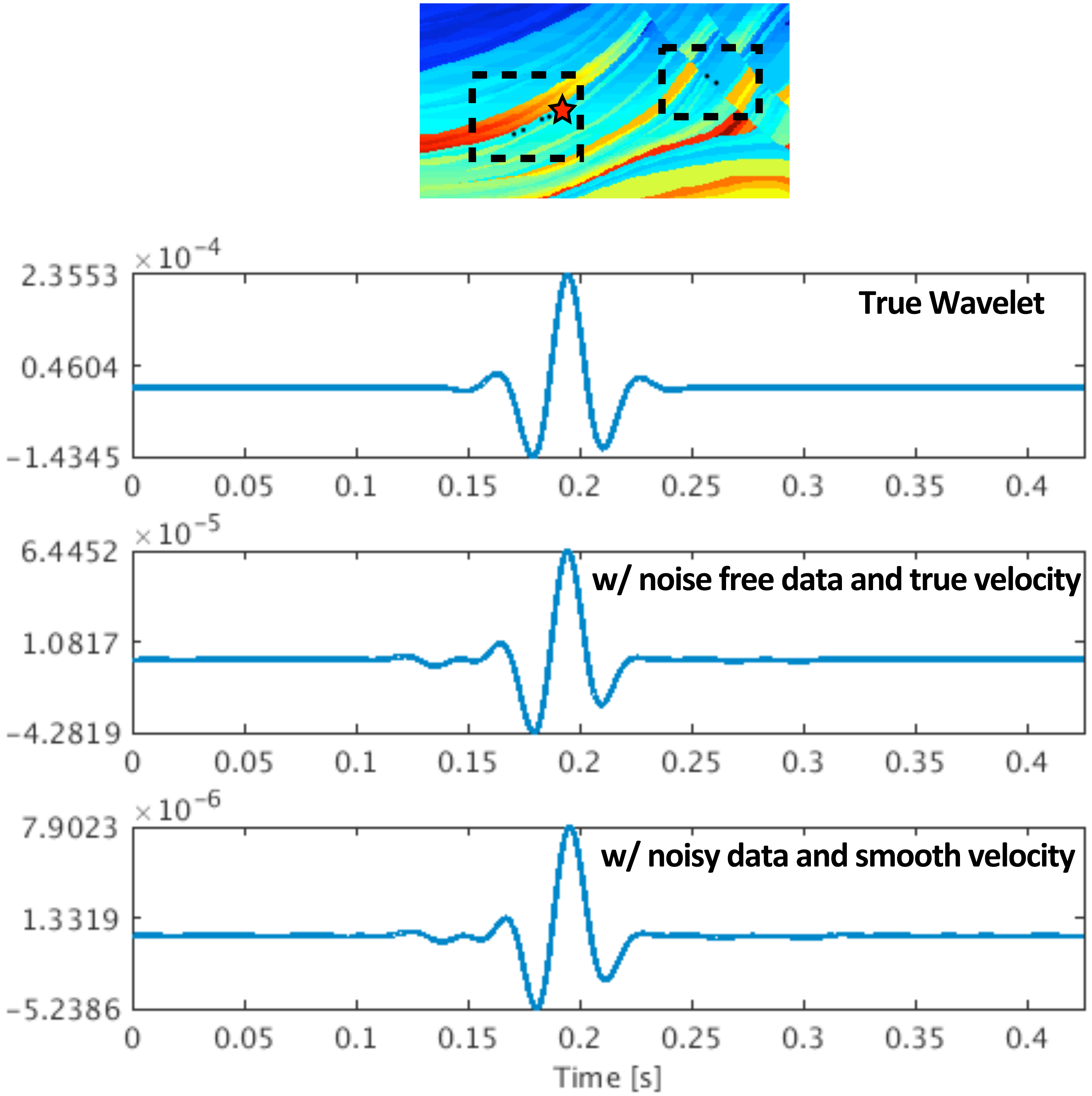


Peak frequency: 30 Hz

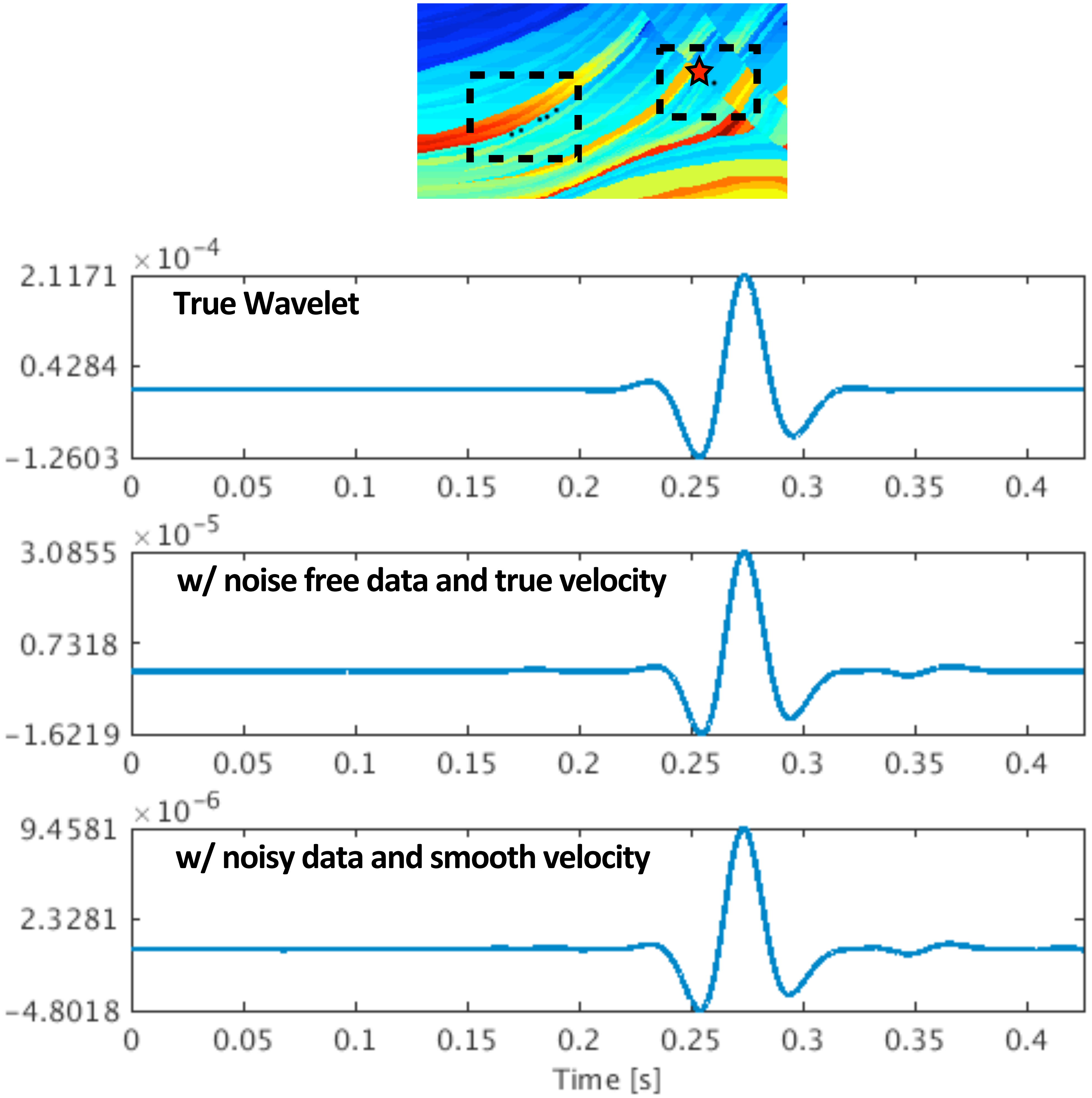


Peak frequency: 30 Hz

Wavelet comparison

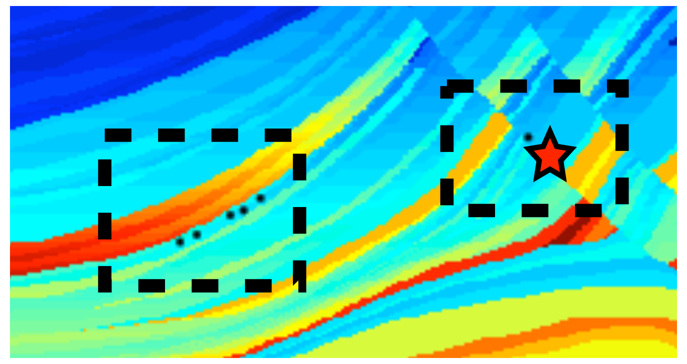


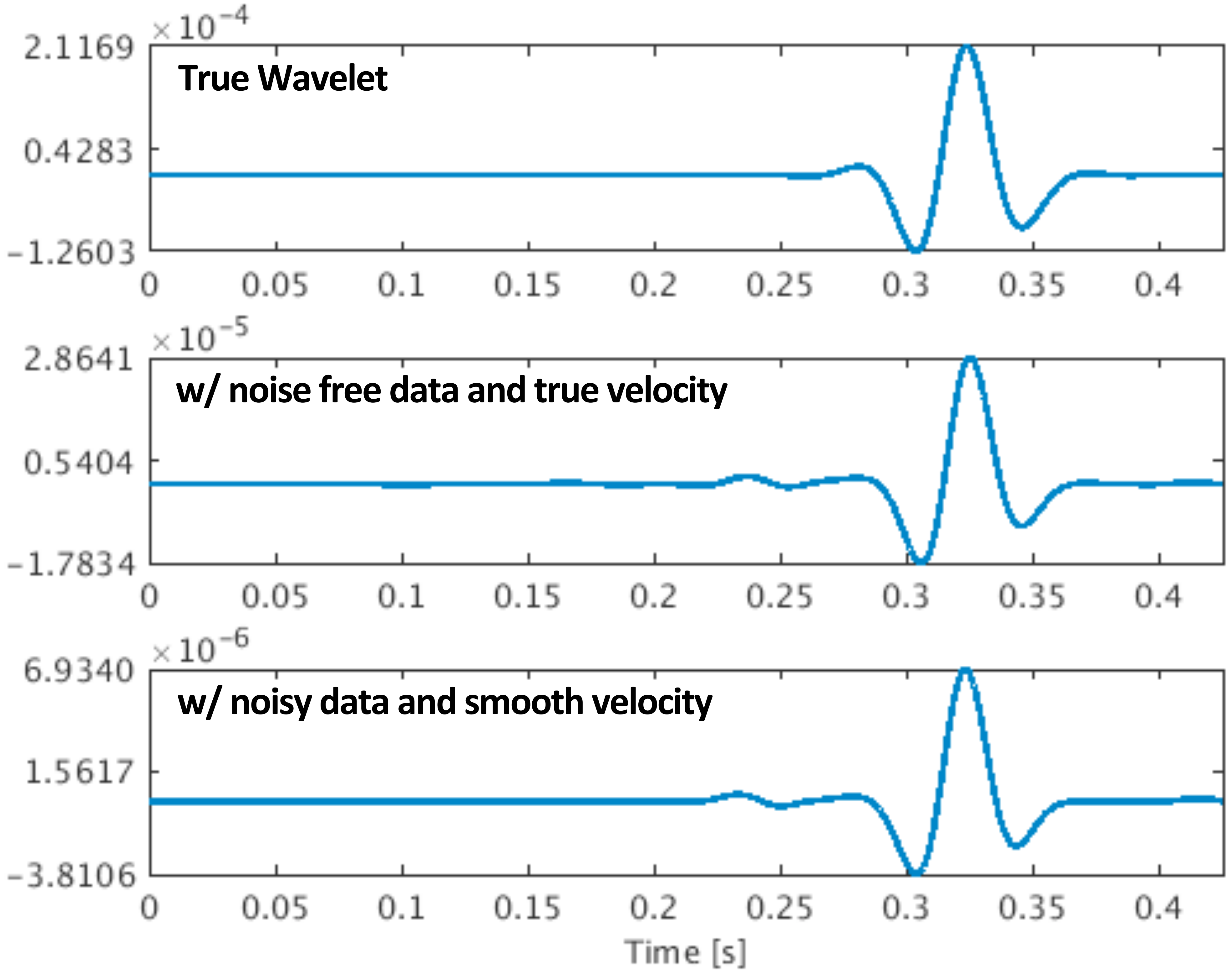
Peak frequency: 30 Hz



Peak frequency: 22 Hz

Wavelet comparison





Peak frequency: 22 Hz

SLIM

Conclusions

Sparsity promotion based method:

- can simultaneously estimate multiple source locations & source-time functions
- can provide locations of fractures by resolving microseismic sources within half a wavelength
- works w/ sources of different frequencies & origin times

Conclusions

Sparsity promotion based method:

- can simultaneously estimate multiple source locations & source-time functions
- can provide locations of fractures by resolving microseismic sources within half a wavelength
- works w/ sources of different frequencies & origin times

Dual formulation provides a computationally efficient scheme.

Acknowledgement

This research was carried out as part of the SINBAD project with the support of the member organizations of the SINBAD Consortium.

Thank you !!