Swift FWI

Swift Team(Felix J. Herrmann, Henryk Modzelewski, Tristan van Leeuwen,
Thomas Lai, Harsh Juneja, Bas Peters, Zhilong Fang)

2013. 12. 04

SLIM
University of British Columbia

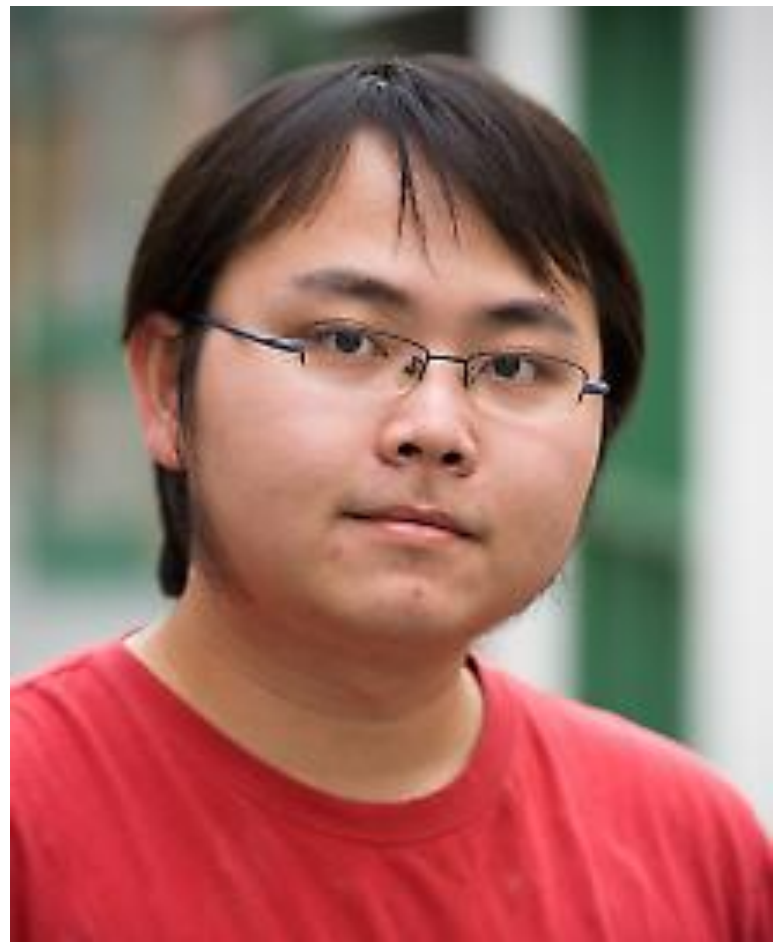
Zhilong Fang

SWIFT TEAM

Felix J. Herrmann

Henryk Modzelewski

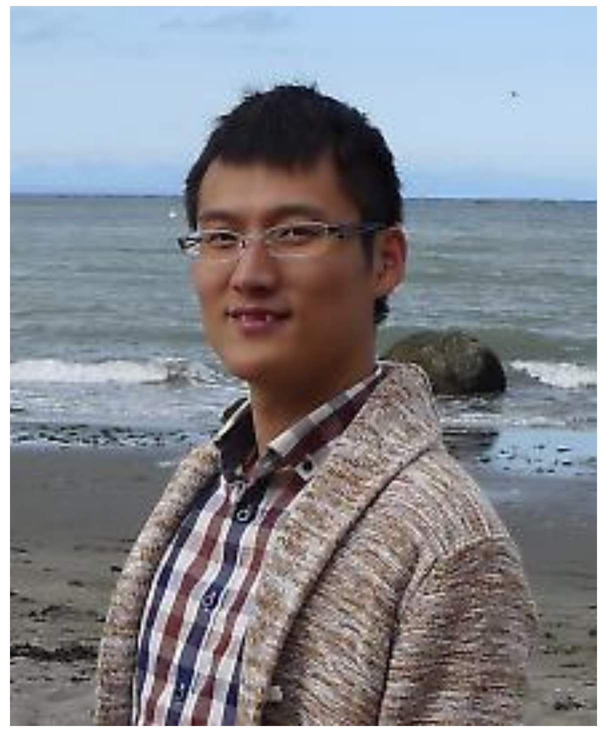
Tristan van Leeuwen



Thomas Lai

Harsh Juneja

Bas Peters



Zhilong Fang

Motivation

Matlab Parallel Computing Toolbox

- forward modeling
- linearized modeling
- full waveform inversion
- ……

Bottleneck licenses

Domain Decomposition

Parallel Over Shots

price = $$$ * nlabs

Motivation

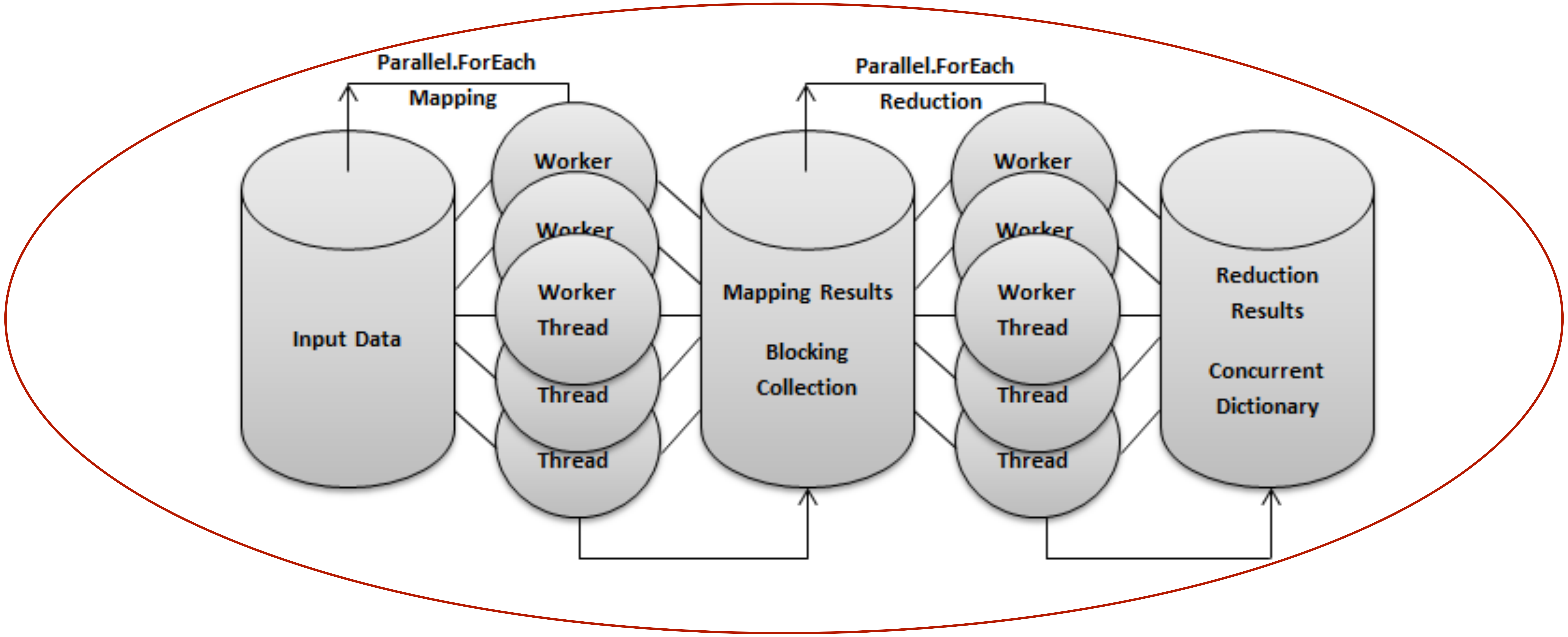
Solution:

+ MapReduce

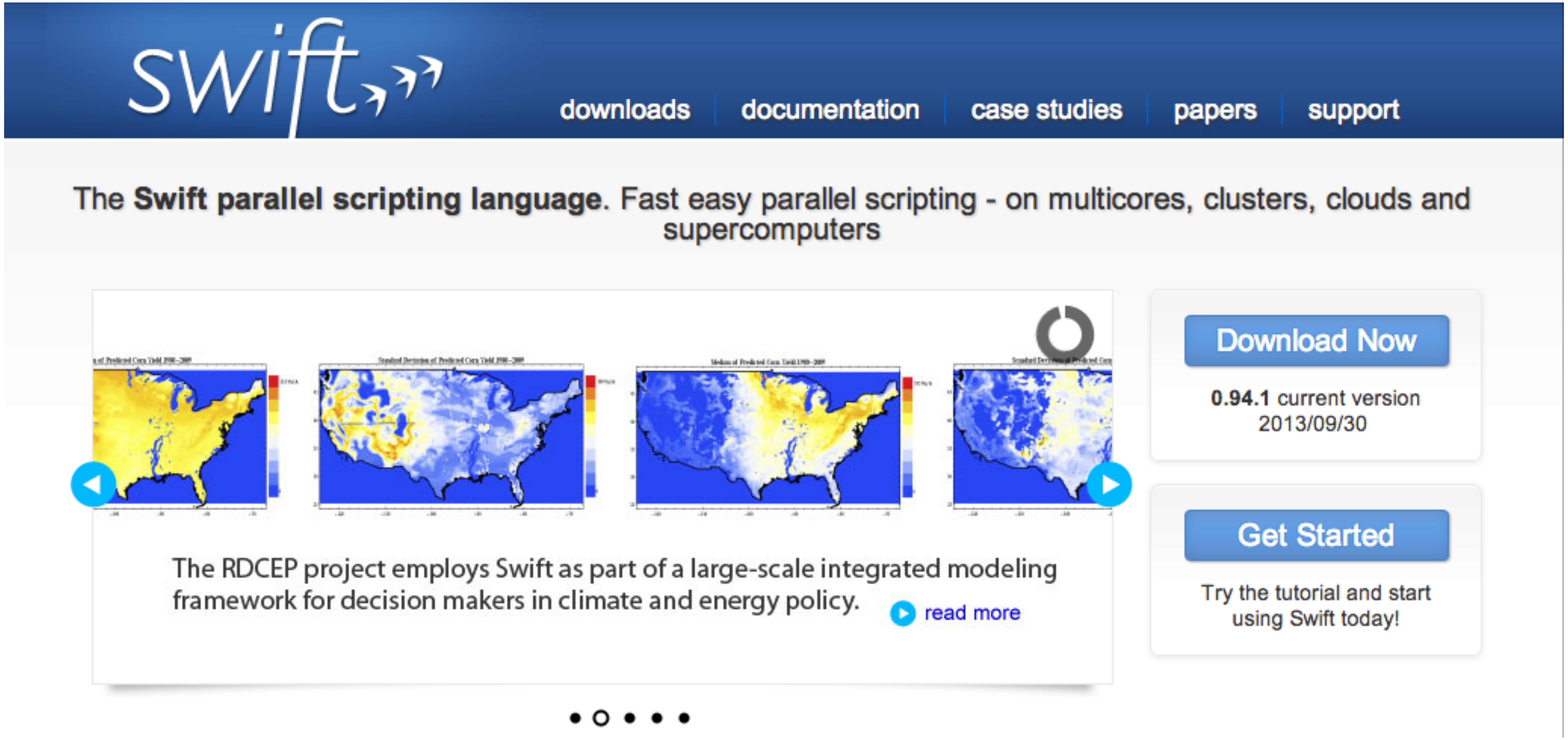
MapReduce

MapReduce: MapReduce is a programming model for processing large data sets with a parallel, distributed algorithm on a cluster.

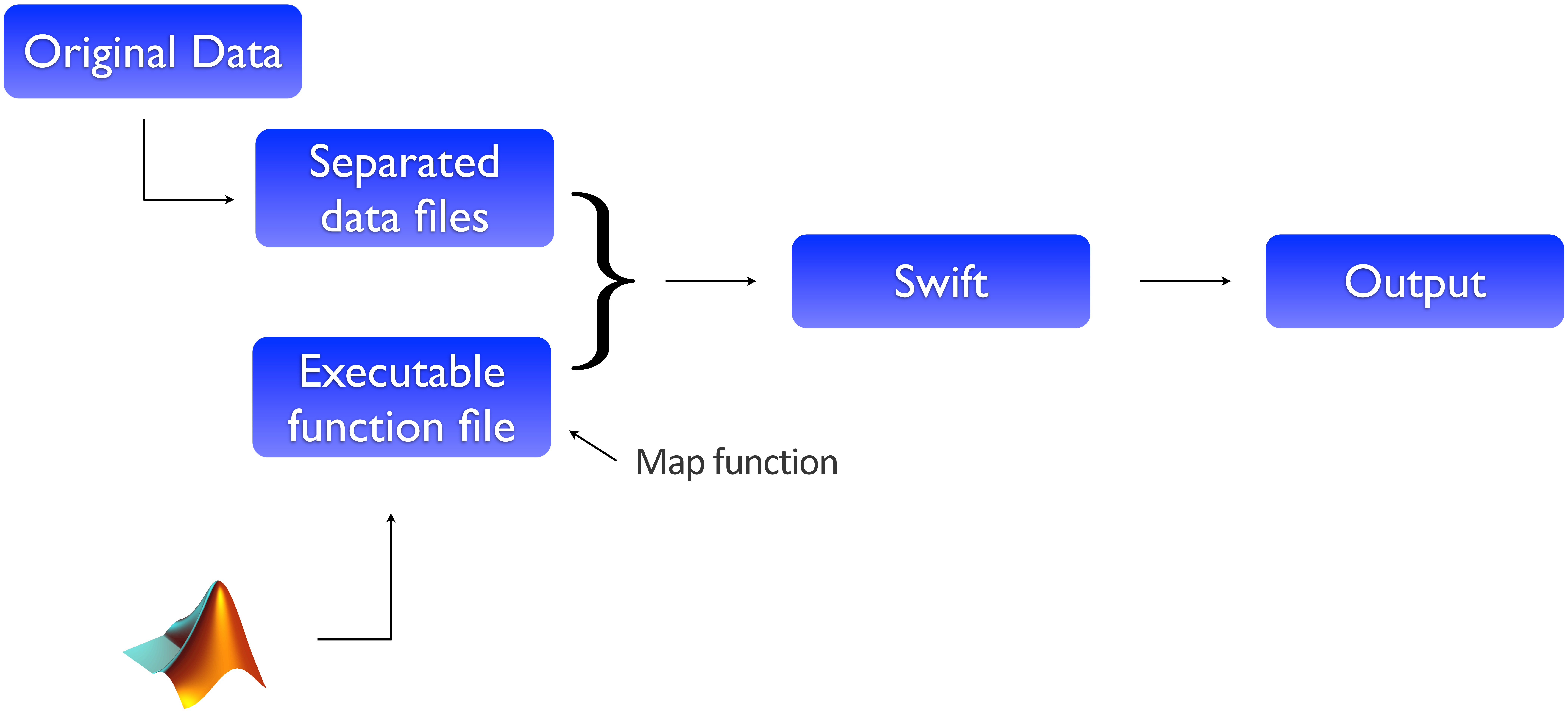
Swift



Swift


swift
downloads
documentation
case studies
papers
support
The Swift parallel scripting language. Fast easy parallel scripting - on multicores, clusters, clouds and supercomputers
The RDCEP project employs Swift as part of a large-scale integrated modeling framework for decision makers in climate and energy policy.
read more
Download Now
0.94.1 current version
2013/09/30
Get Started
Try the tutorial and start using Swift today!

Swift


Original Data
Separated data files
Executable function file
Swift
Output
Map function

3D Full waveform inversion

Optimization problem:

$$\min_{\mathbf{m}} f(\mathbf{m}) = \sum_{i=1}^{M} f_i(\mathbf{m}) = \sum_{i=1}^{M} \|F_i(\mathbf{m}) - \mathbf{d}_i\|_2^2$$

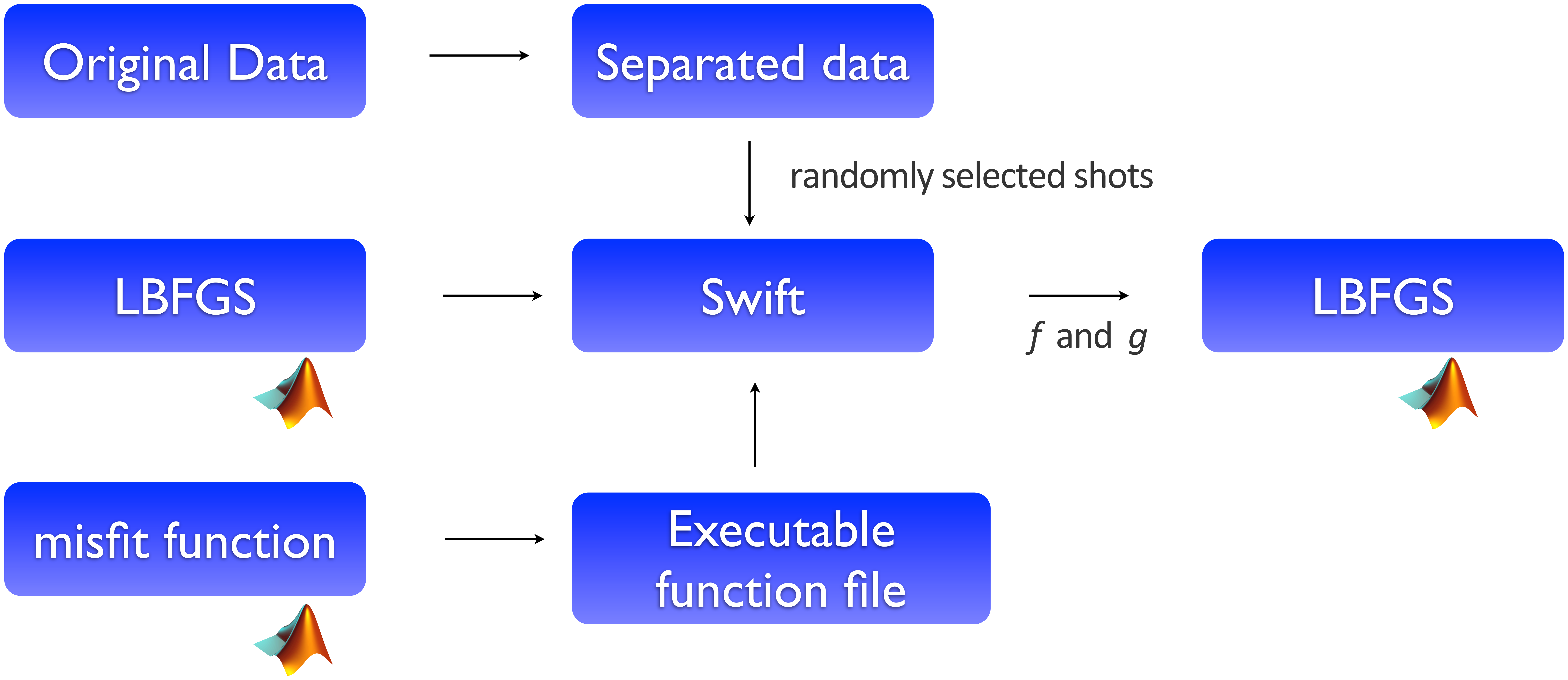
Gradient: $g(\mathbf{m}) = \sum_{i=1}^{M} g_i(\mathbf{m})$

Map: $(\mathbf{m}, \mathbf{d}_i) \rightarrow (f_i, g_i)$

Reduce: $(f_i, g_i)_{1 \leq i \leq M} \rightarrow f(\mathbf{m}) = \sum_{i=1}^{M} f_i(\mathbf{m}), g(\mathbf{m}) = \sum_{i=1}^{M} g_i(\mathbf{m})$

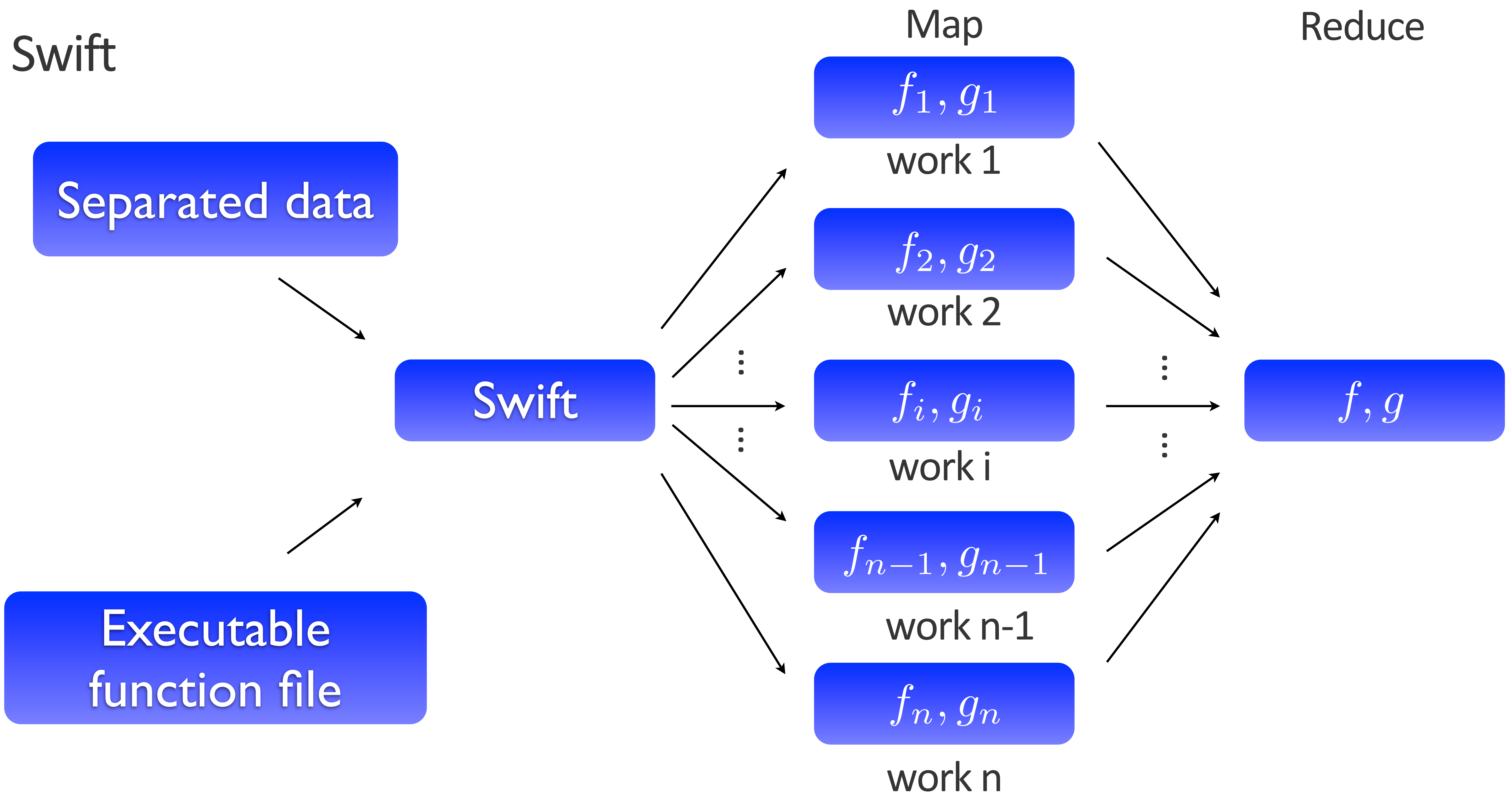
3D full waveform inversion with Swift

Workflow



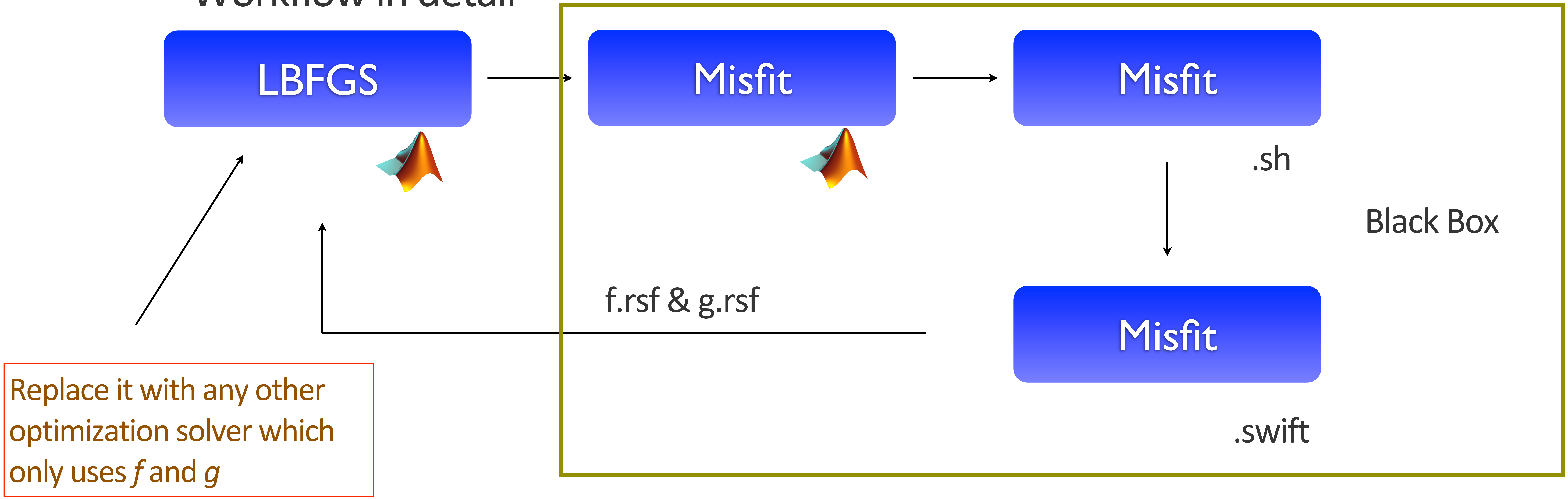
3D full waveform inversion with Swift

Swift



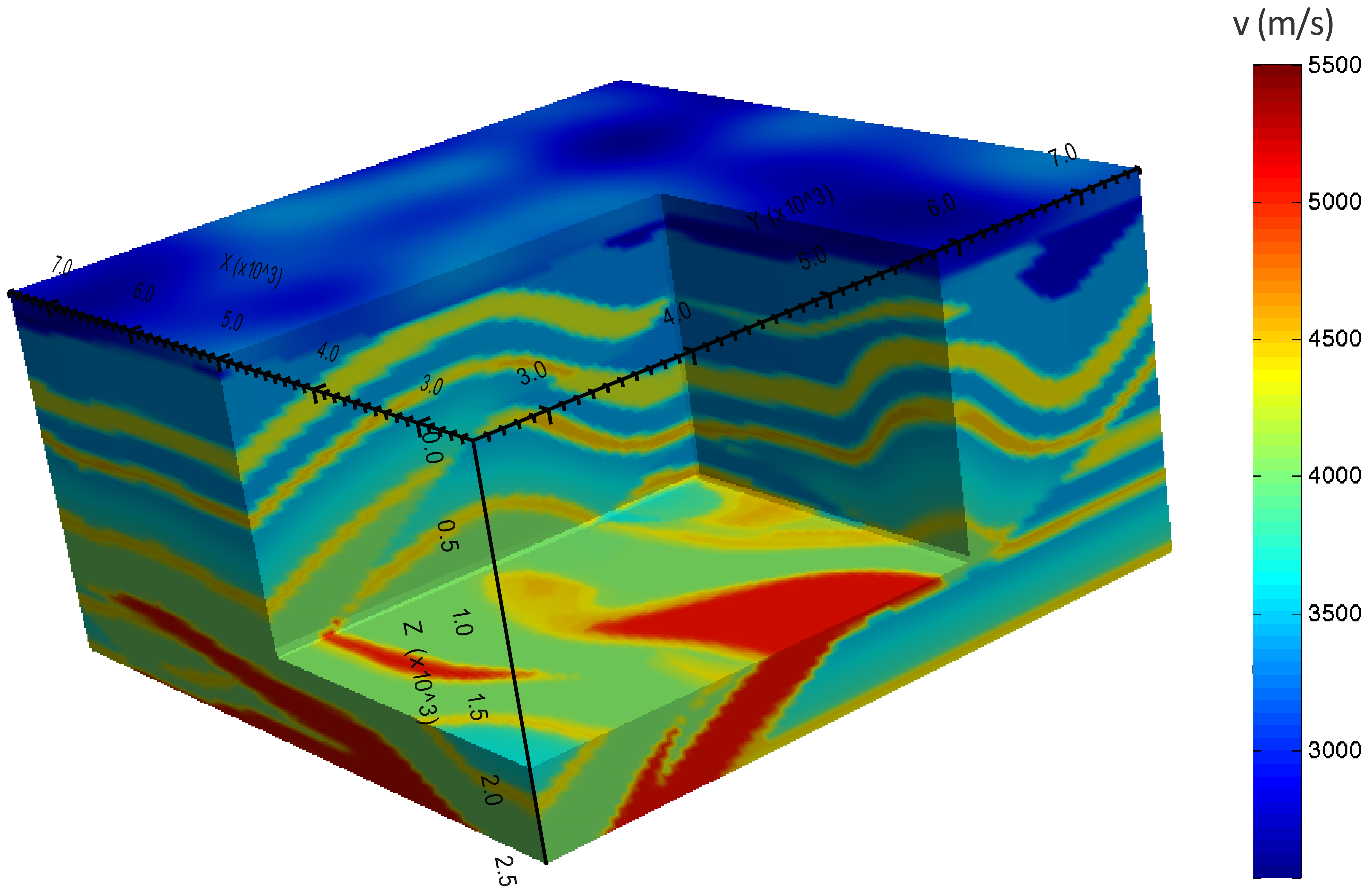
3D full waveform inversion with Swift

Workflow in detail



Numerical Experiment

Overthrust model



Model Infomation:

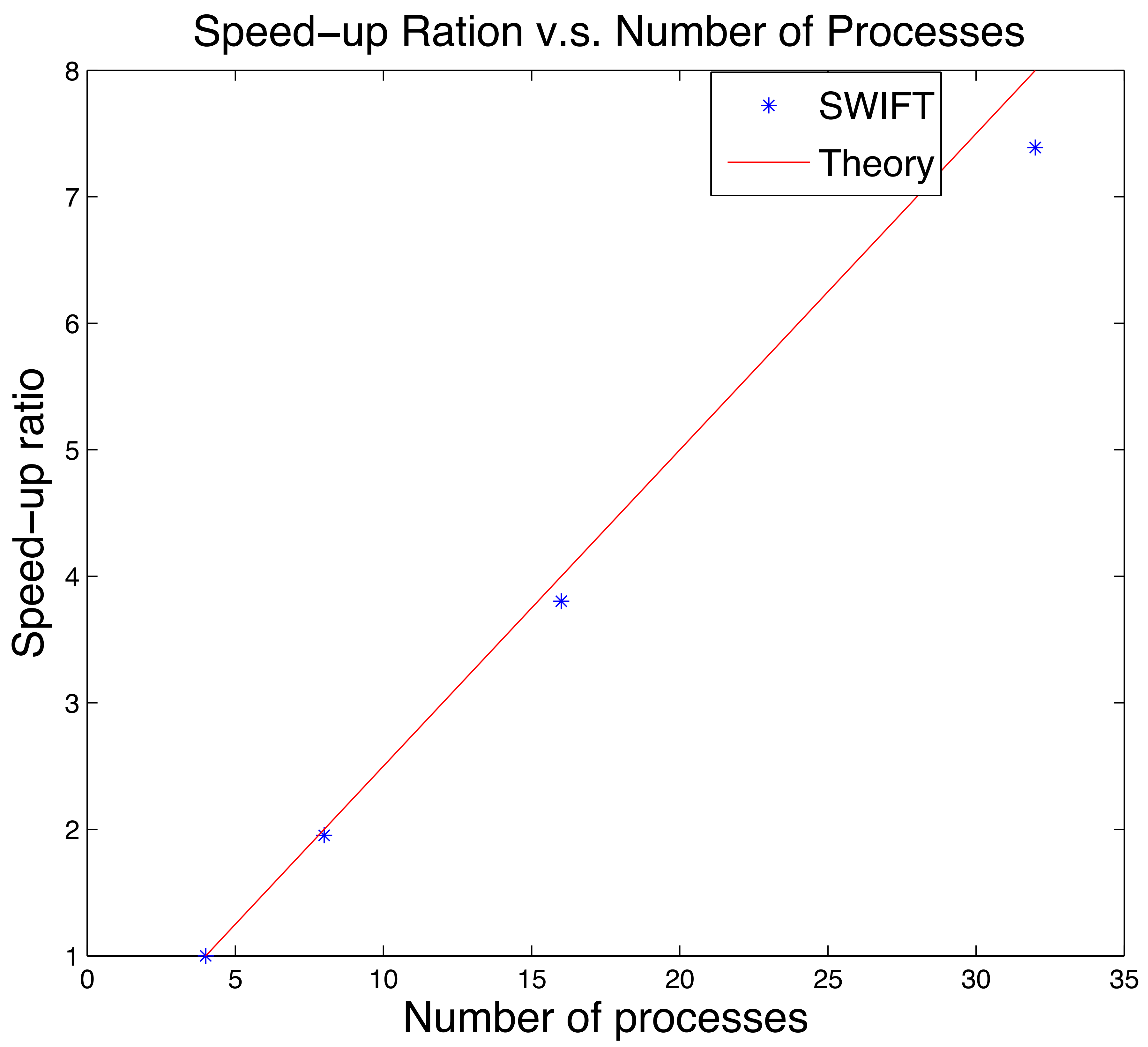
model size	- 26*51*51
	2.5* 5 *5 km
d	- [100 100 100] m
# of shots	- 121
# of receivers	- 2601
frequency	- 4 Hz
source depth	- 100 m
receiver depth	- 100 m

Parallel performance

Number of shots: 121

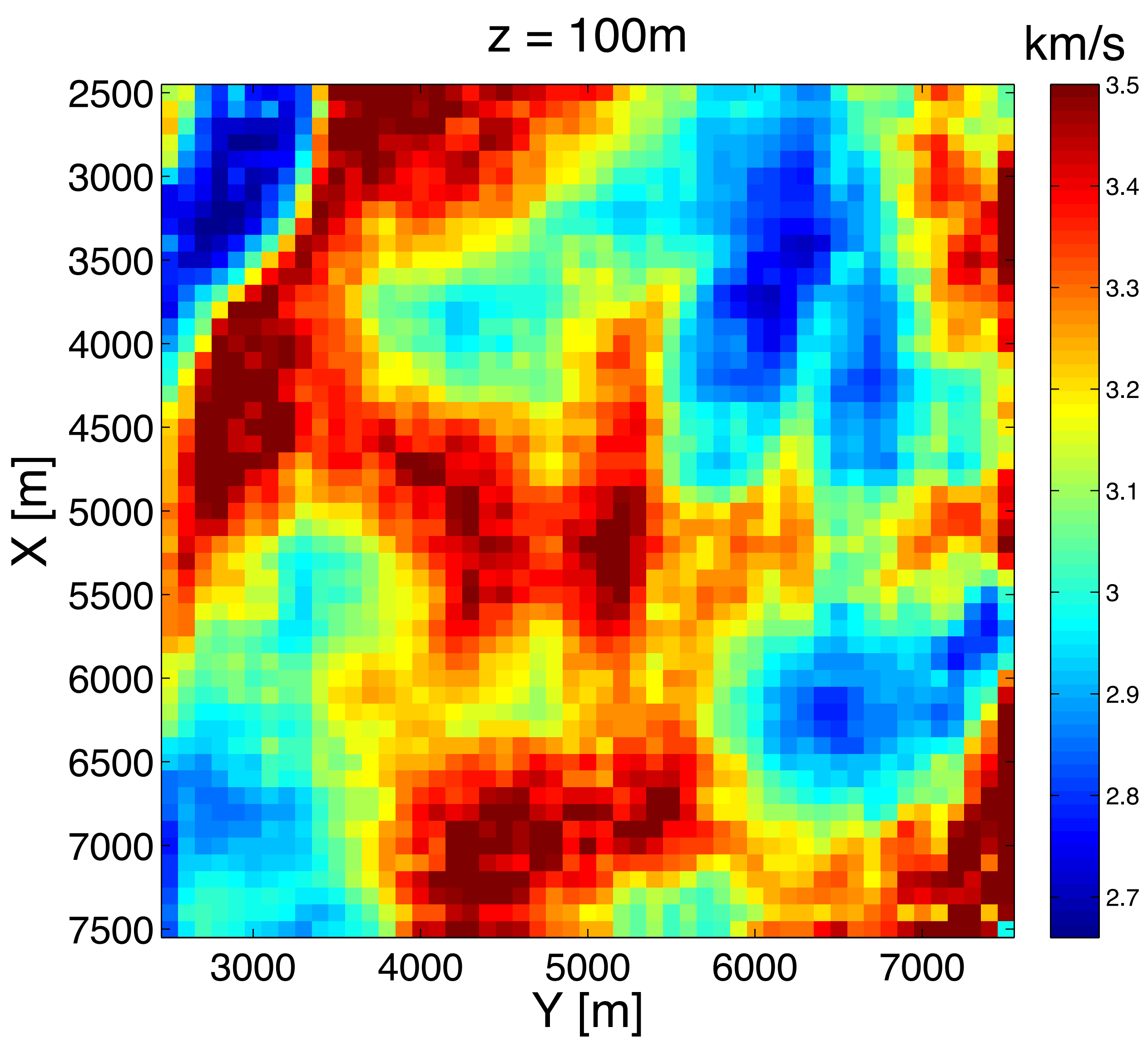
Number of processes:

4, 8, 16, 32

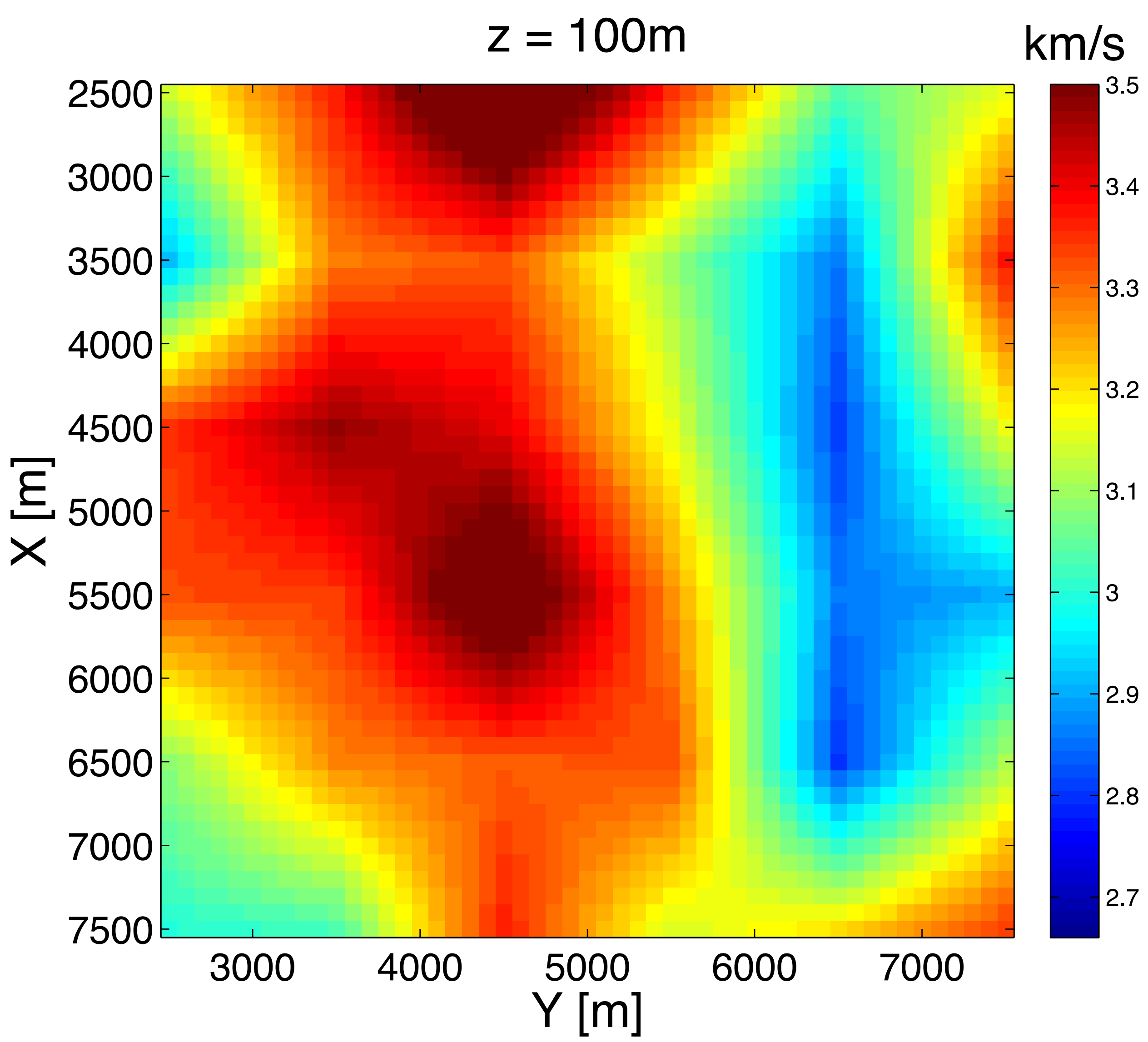


z = 100m

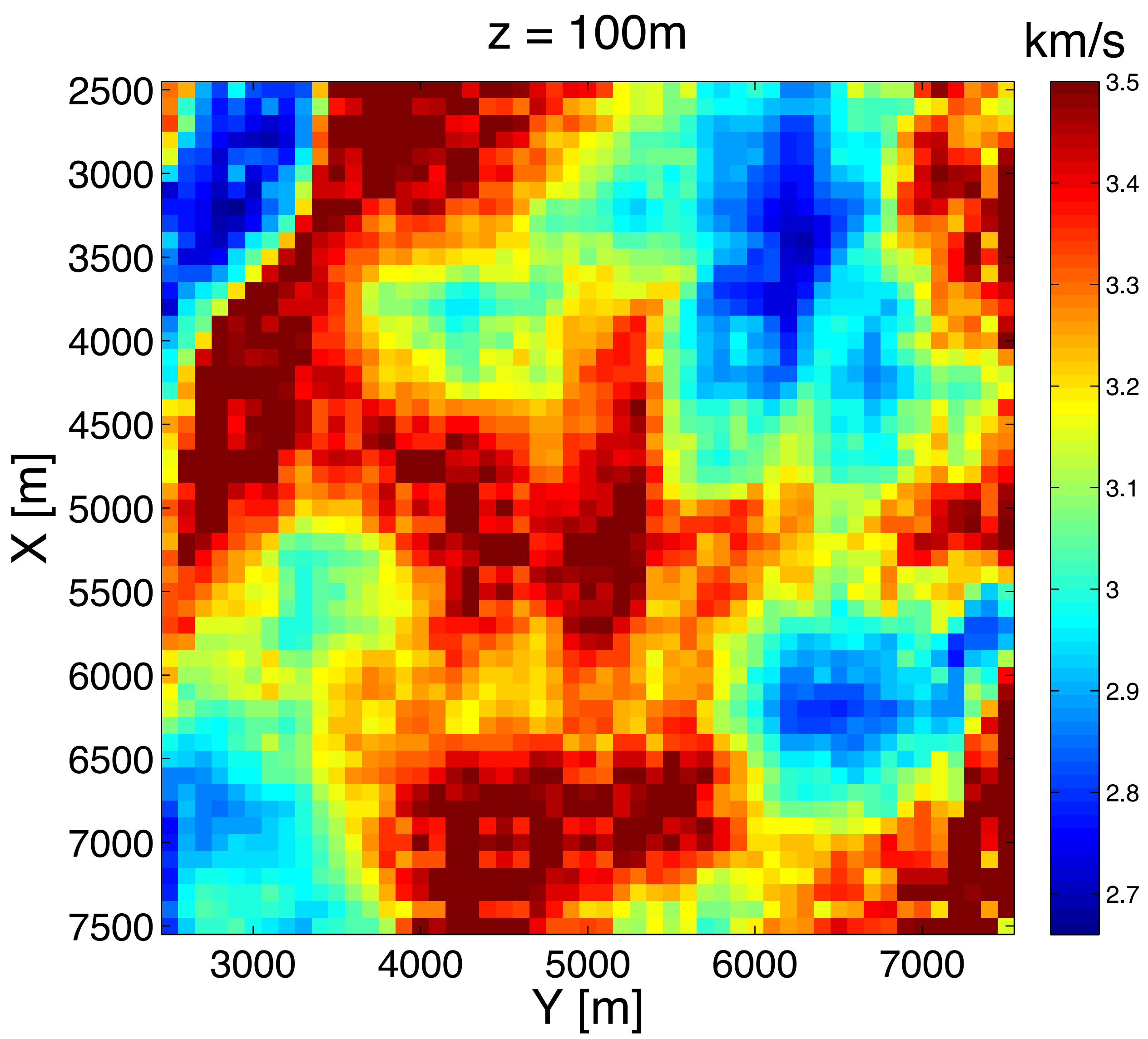
Randomly Selected Shots



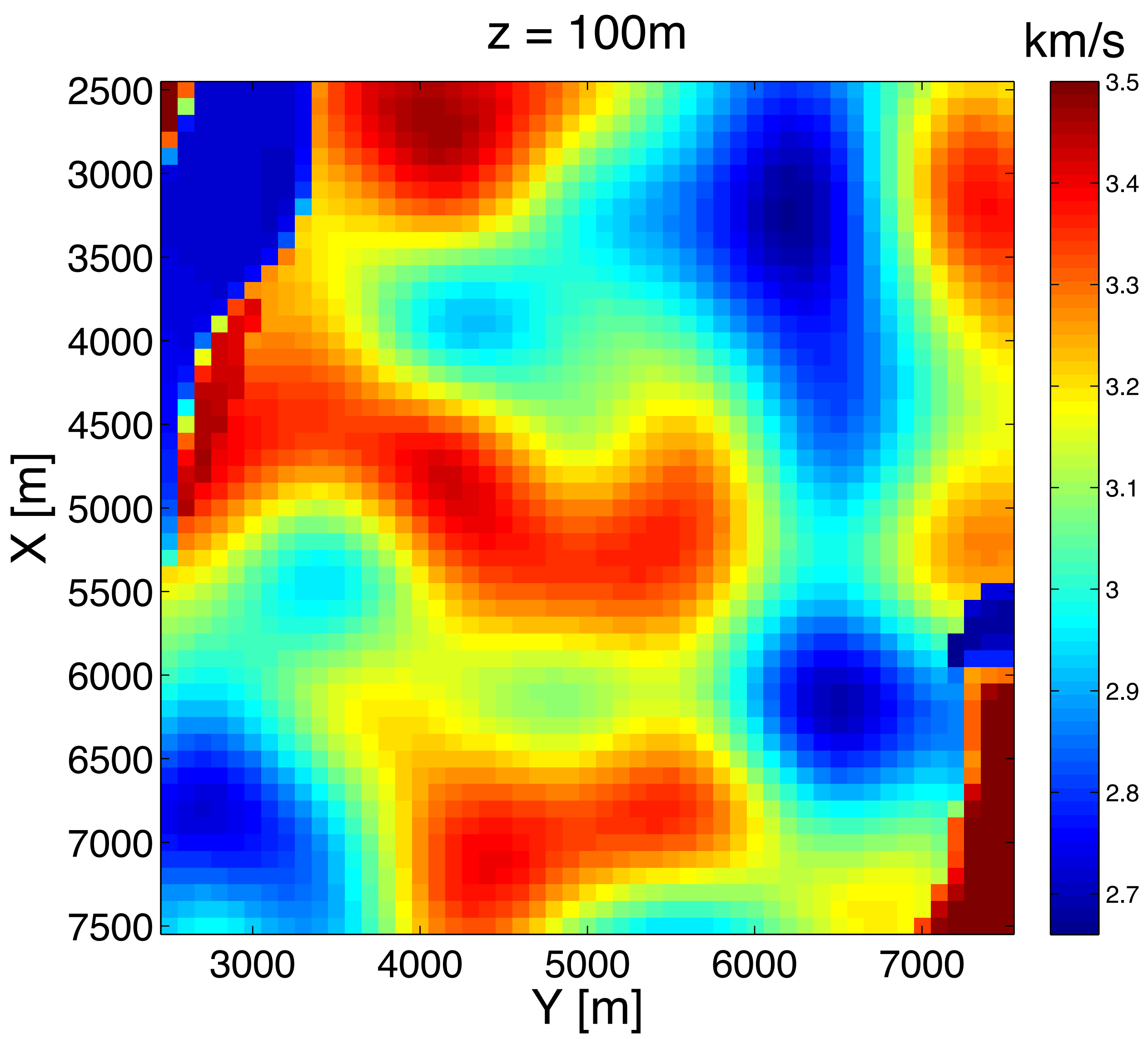
Initial model



All Shots

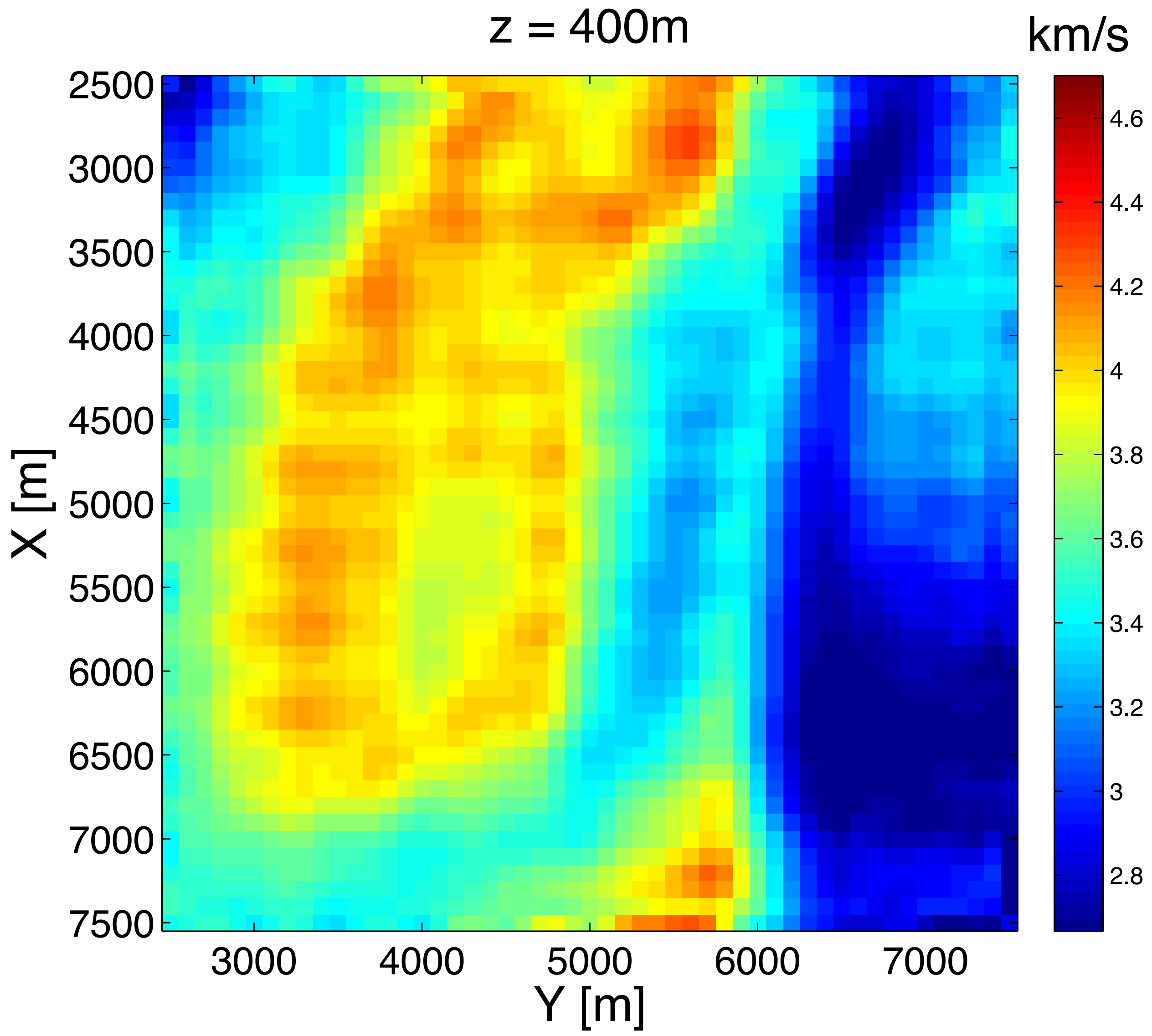


True model

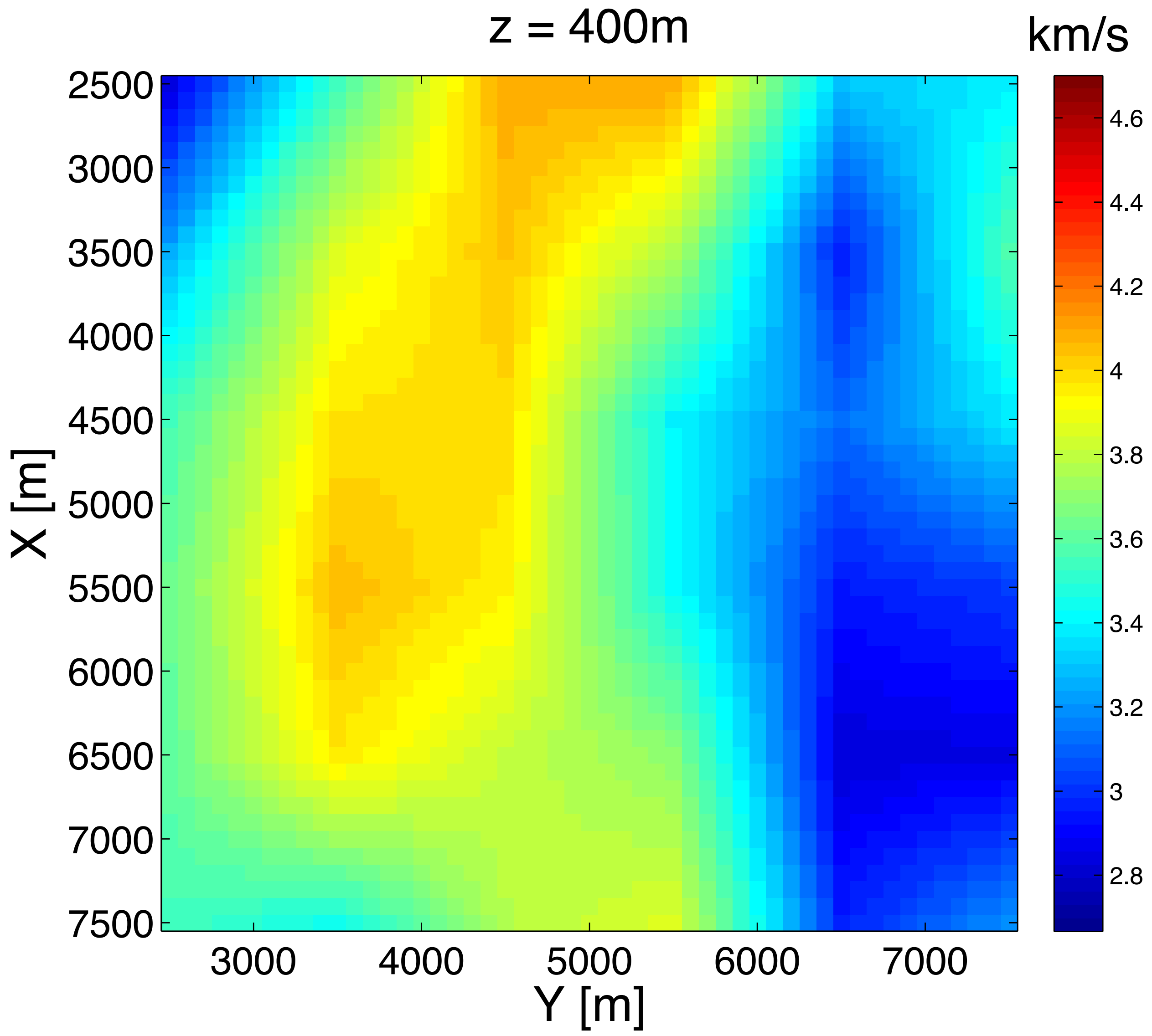


z = 400m

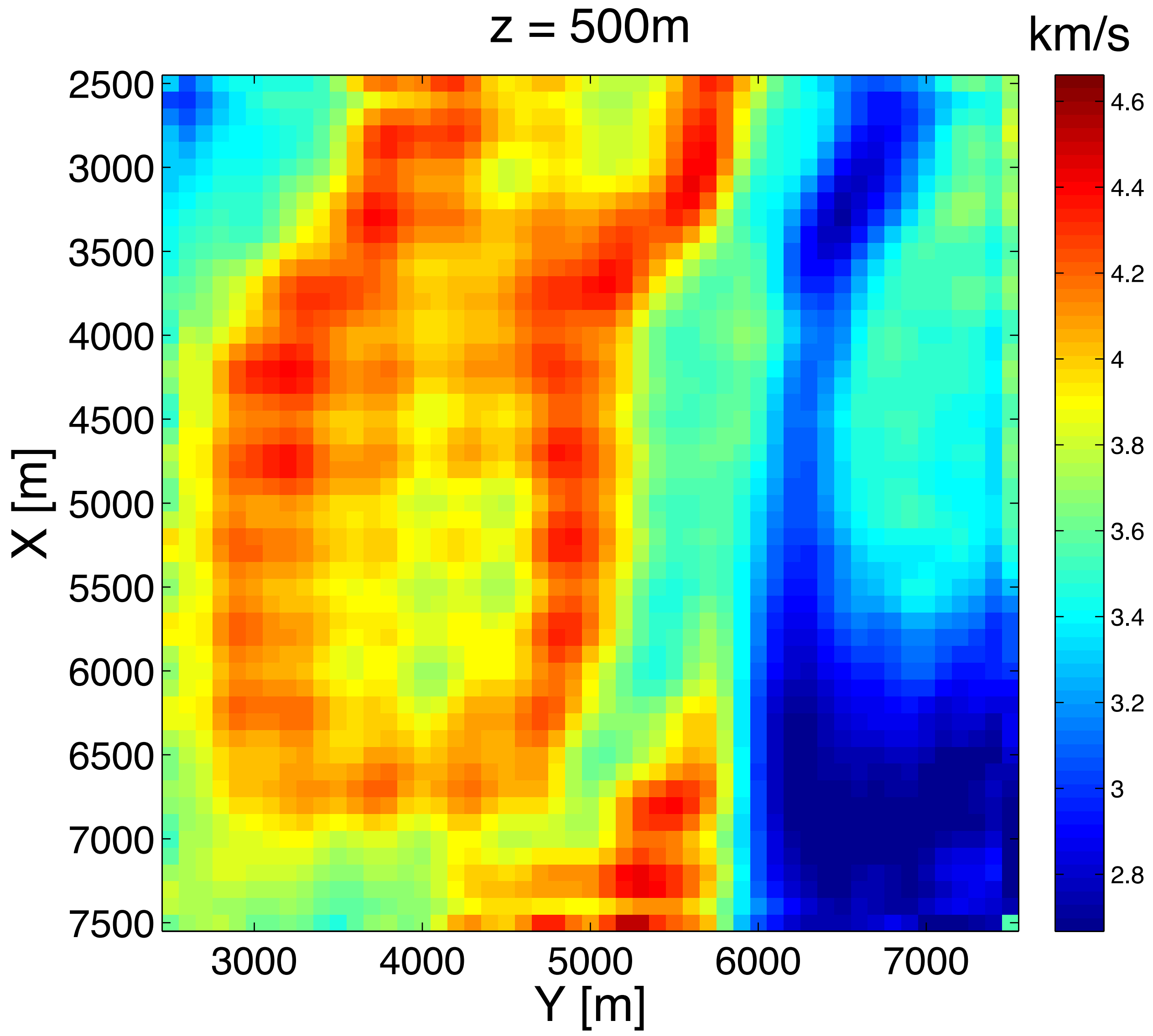
Randomly Selected Shots



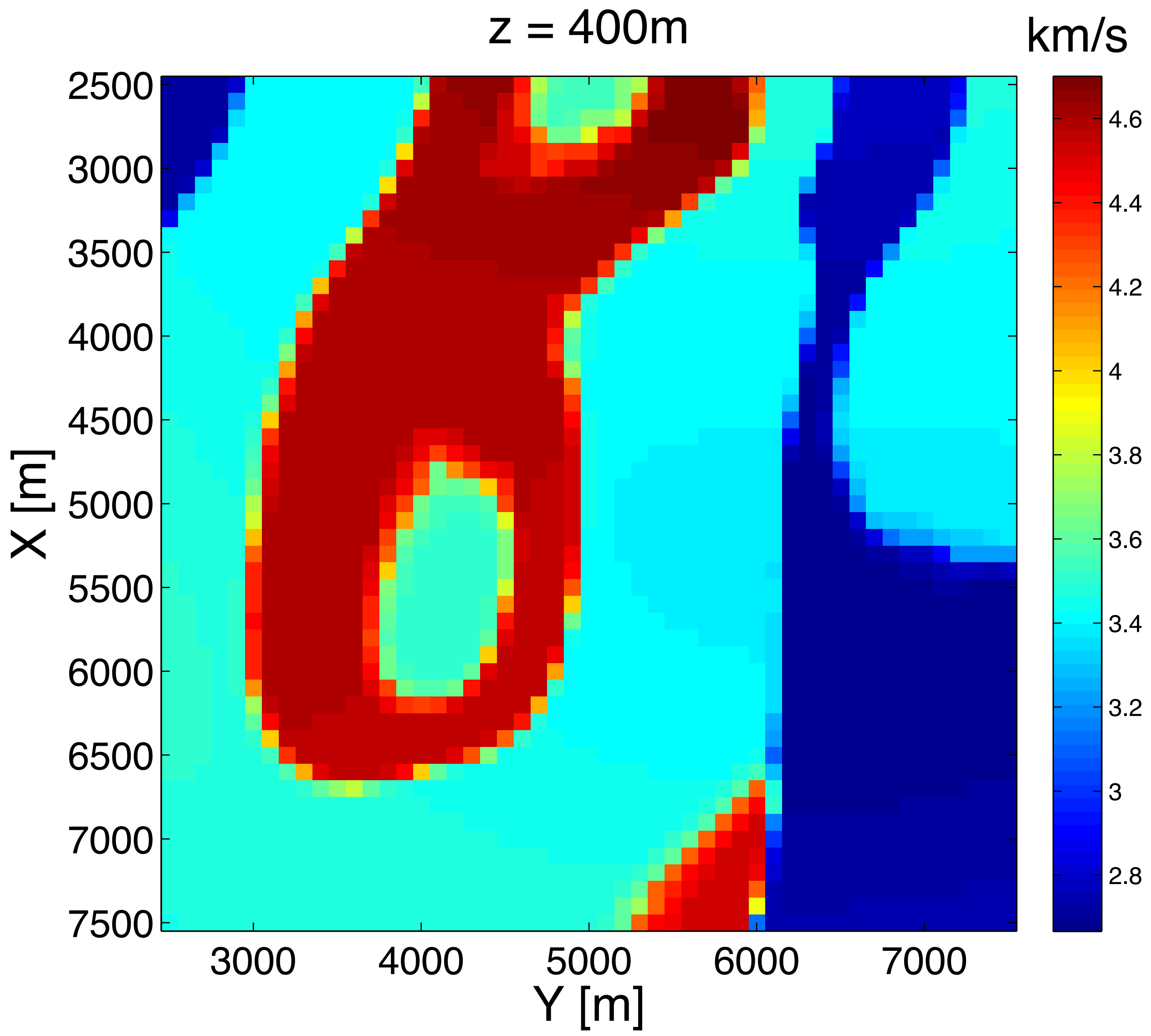
Initial model



All Shots

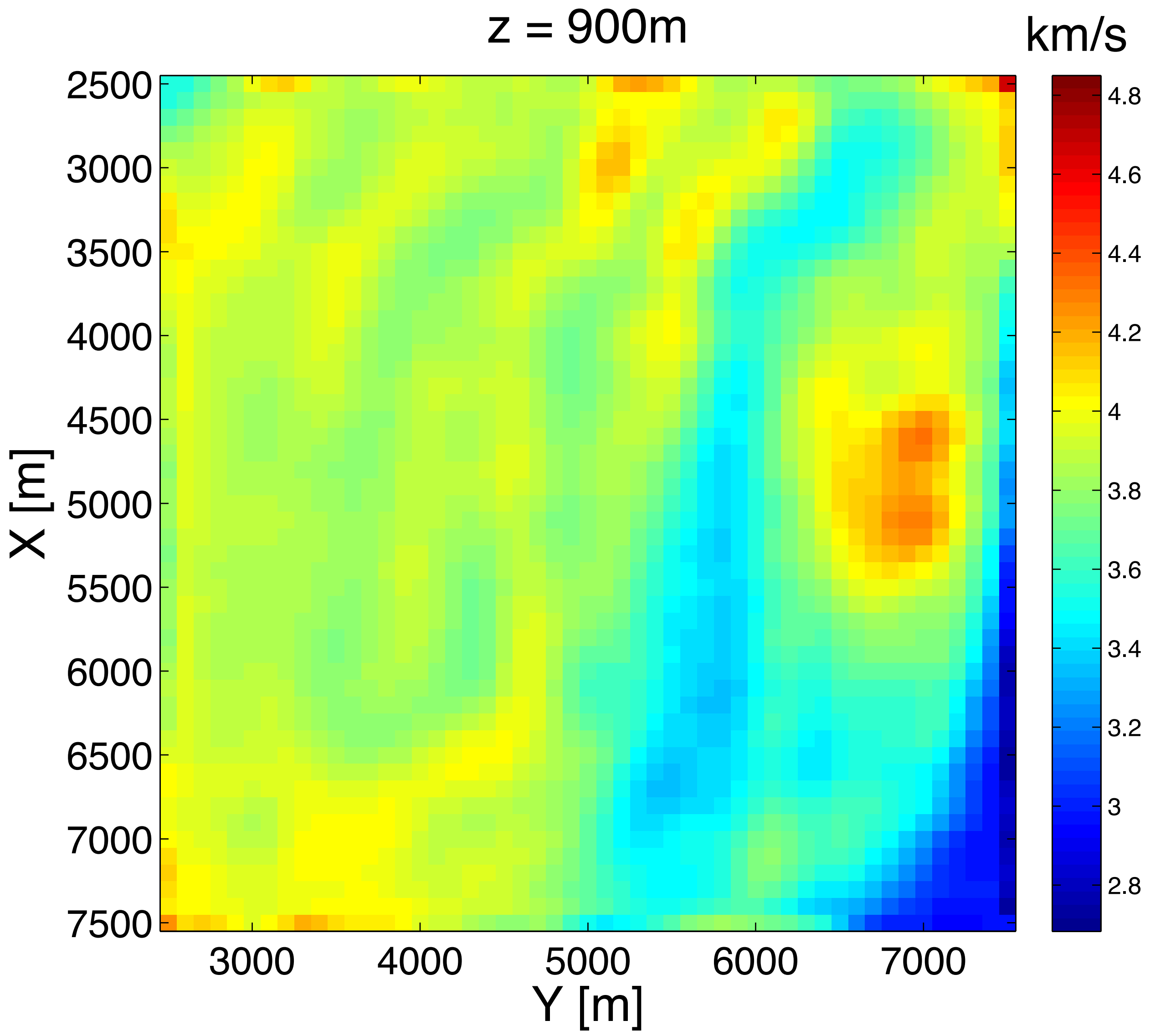


True model

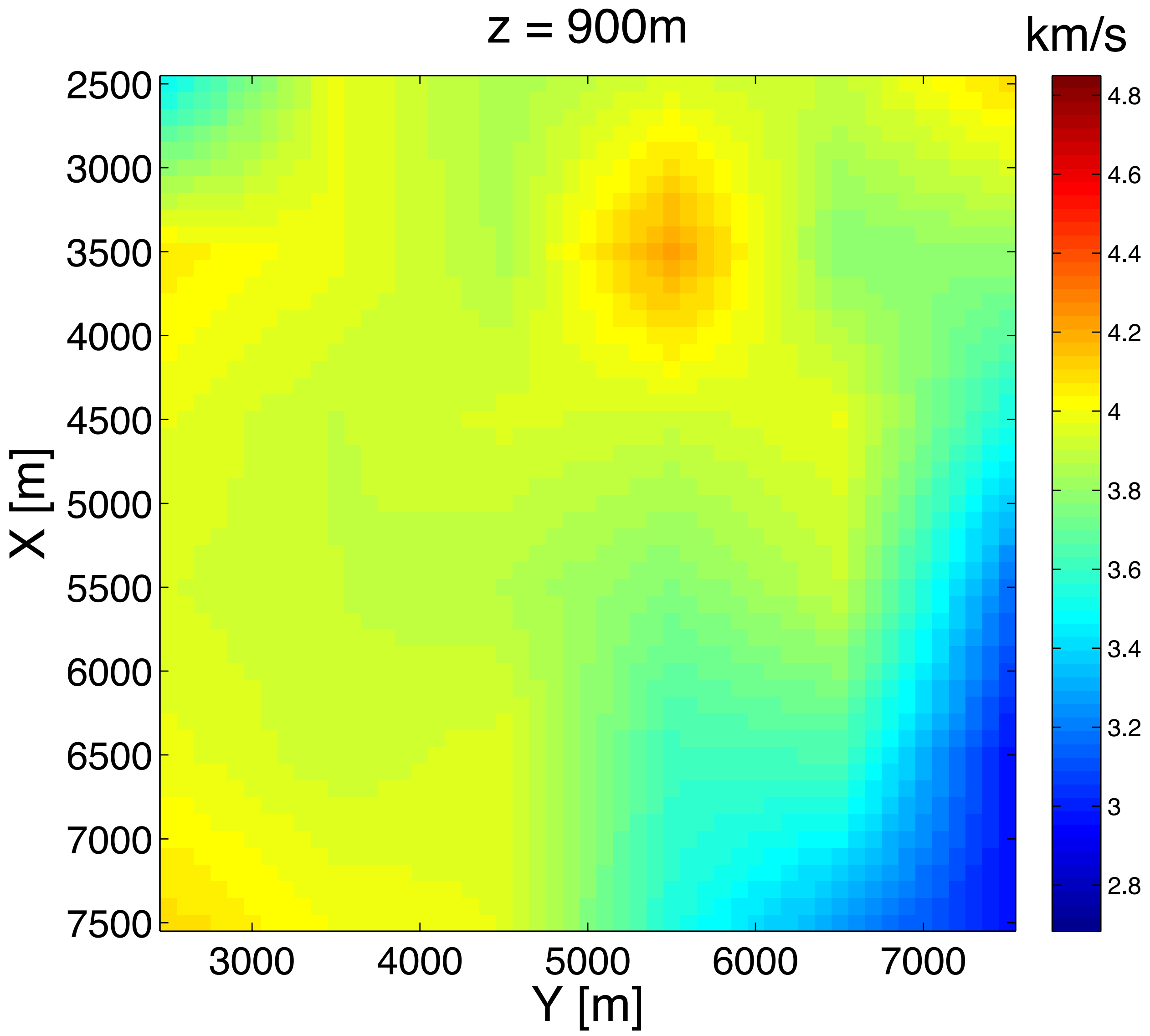


z = 900m

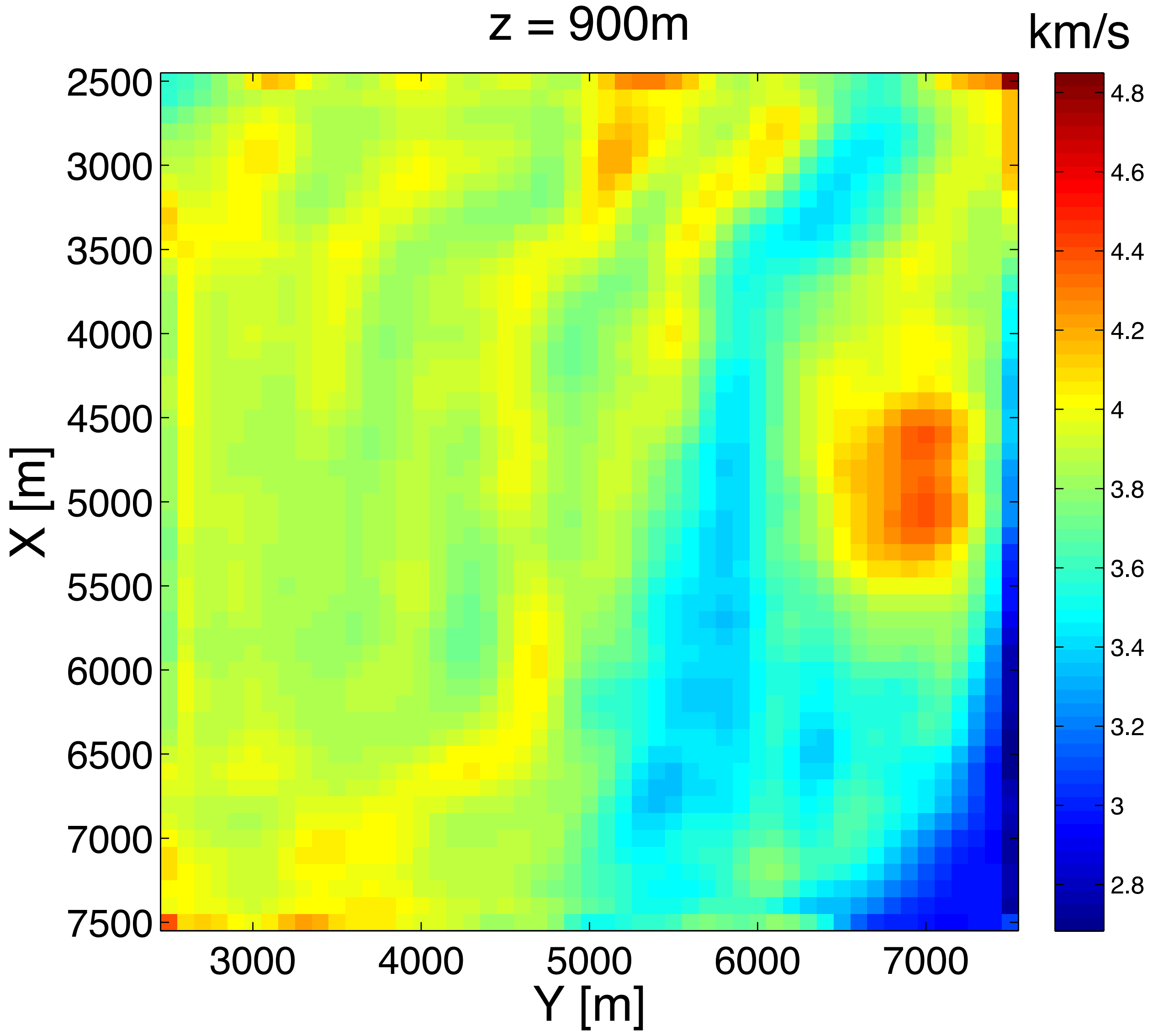
Randomly Selected Shots



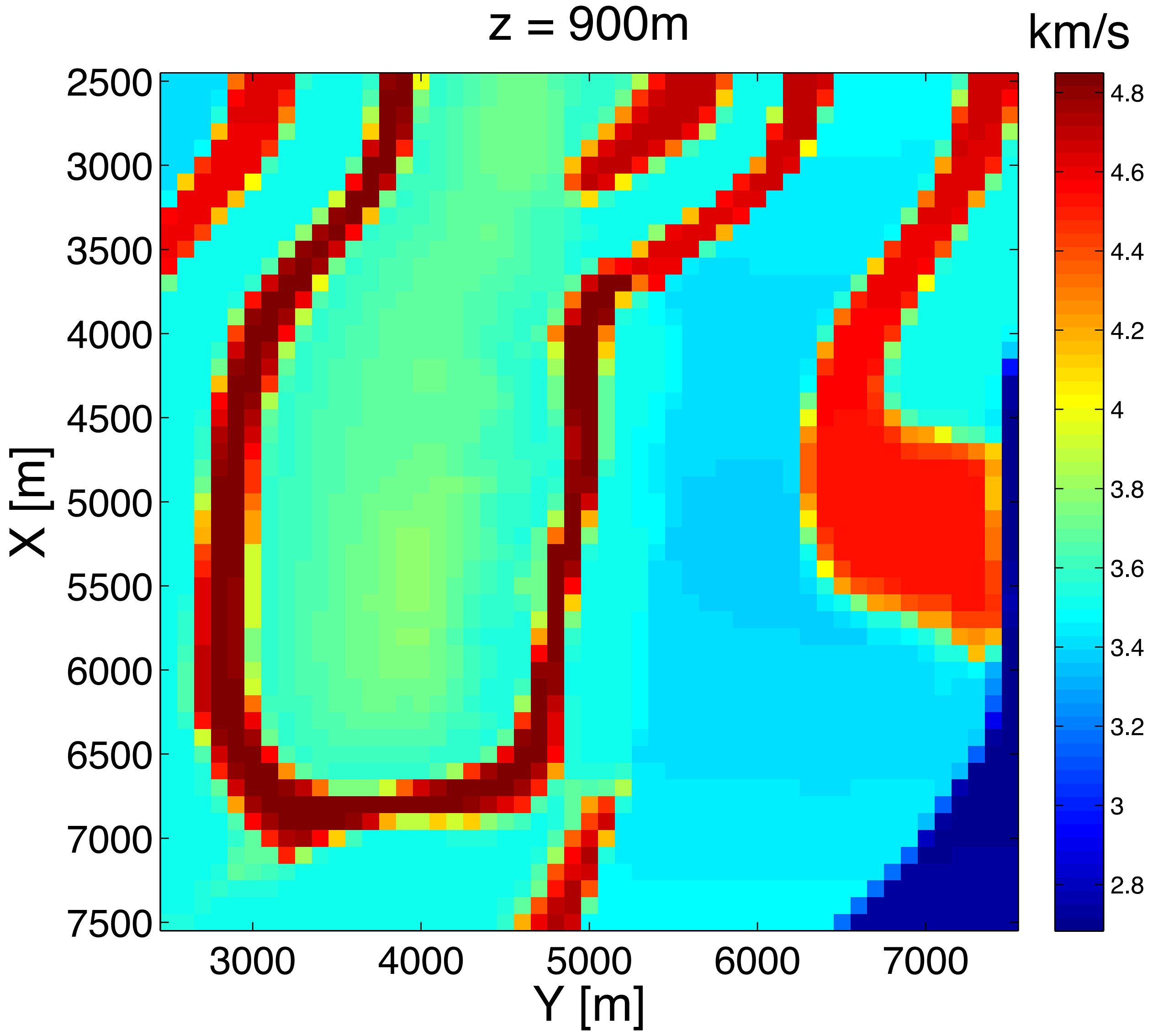
Initial model



All Shots

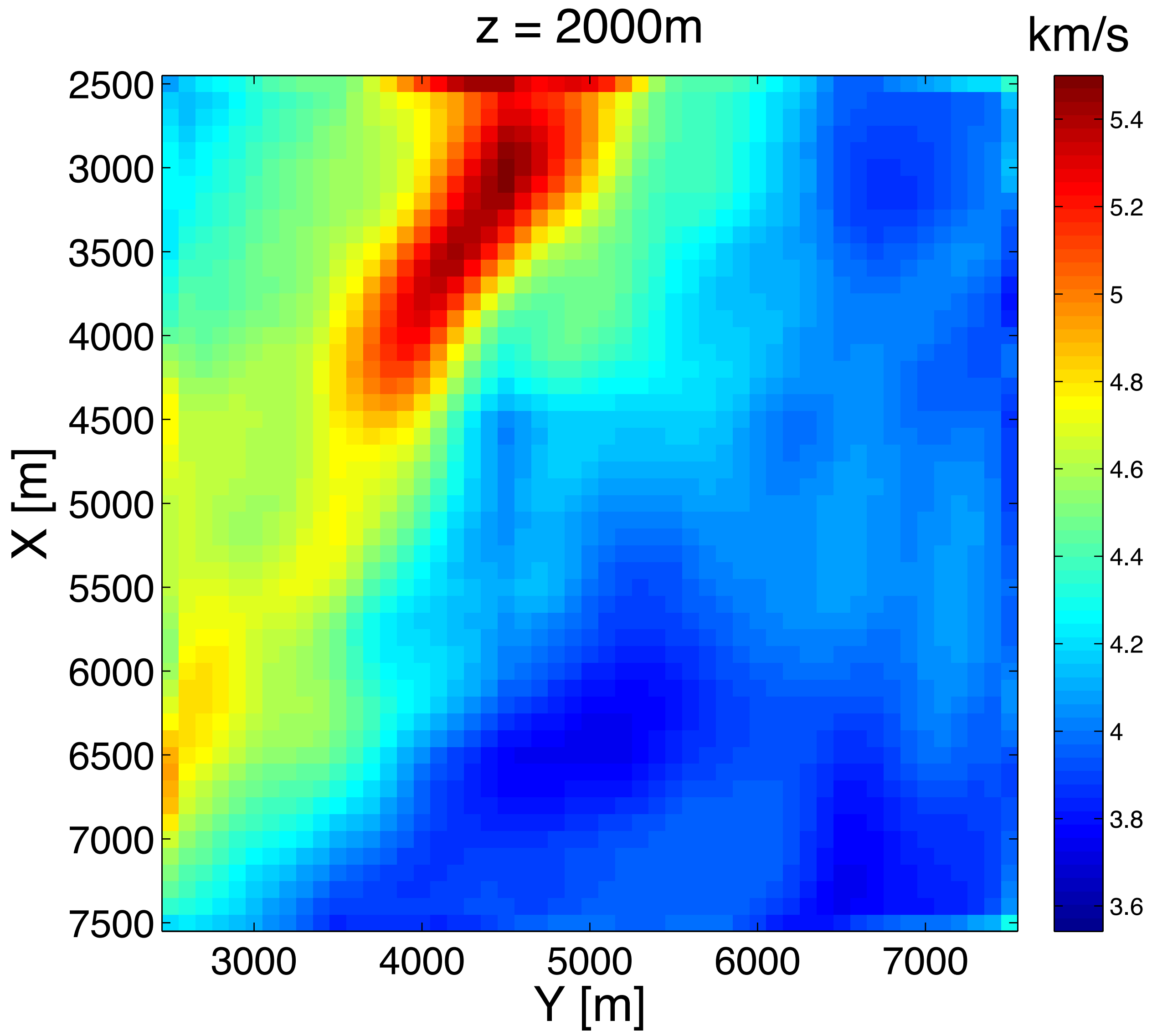


True model

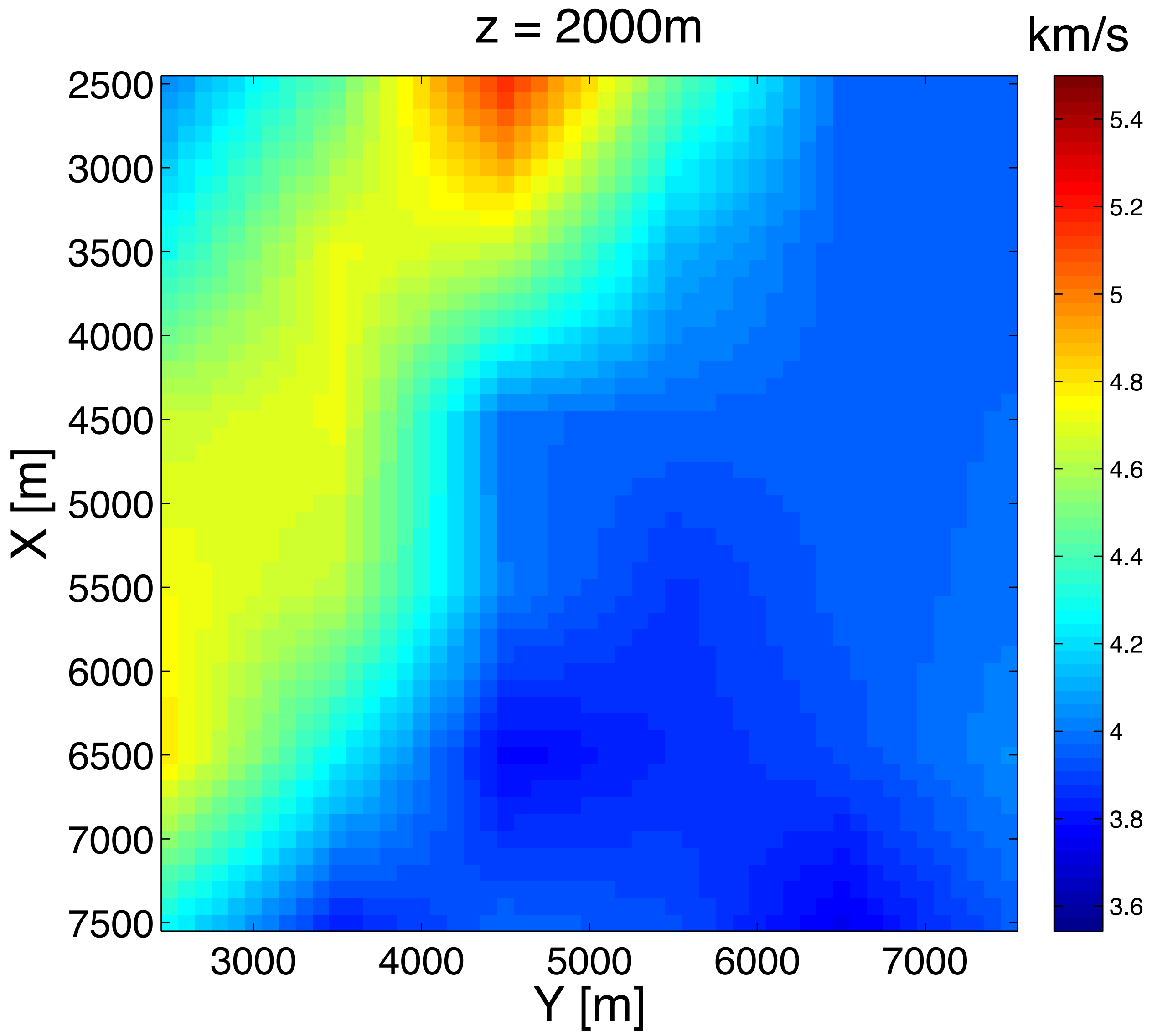


z = 2000m

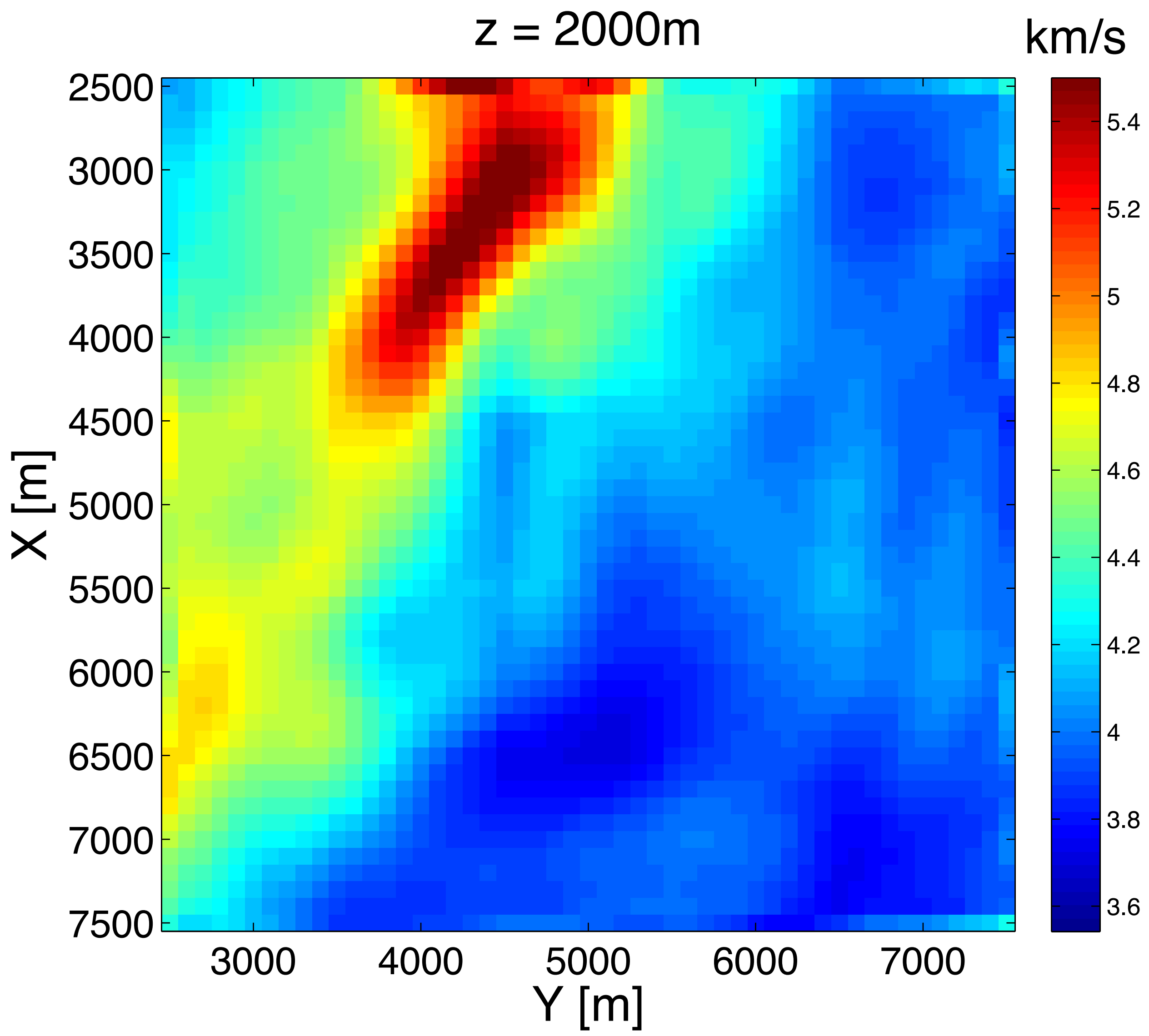
Randomly Selected Shots



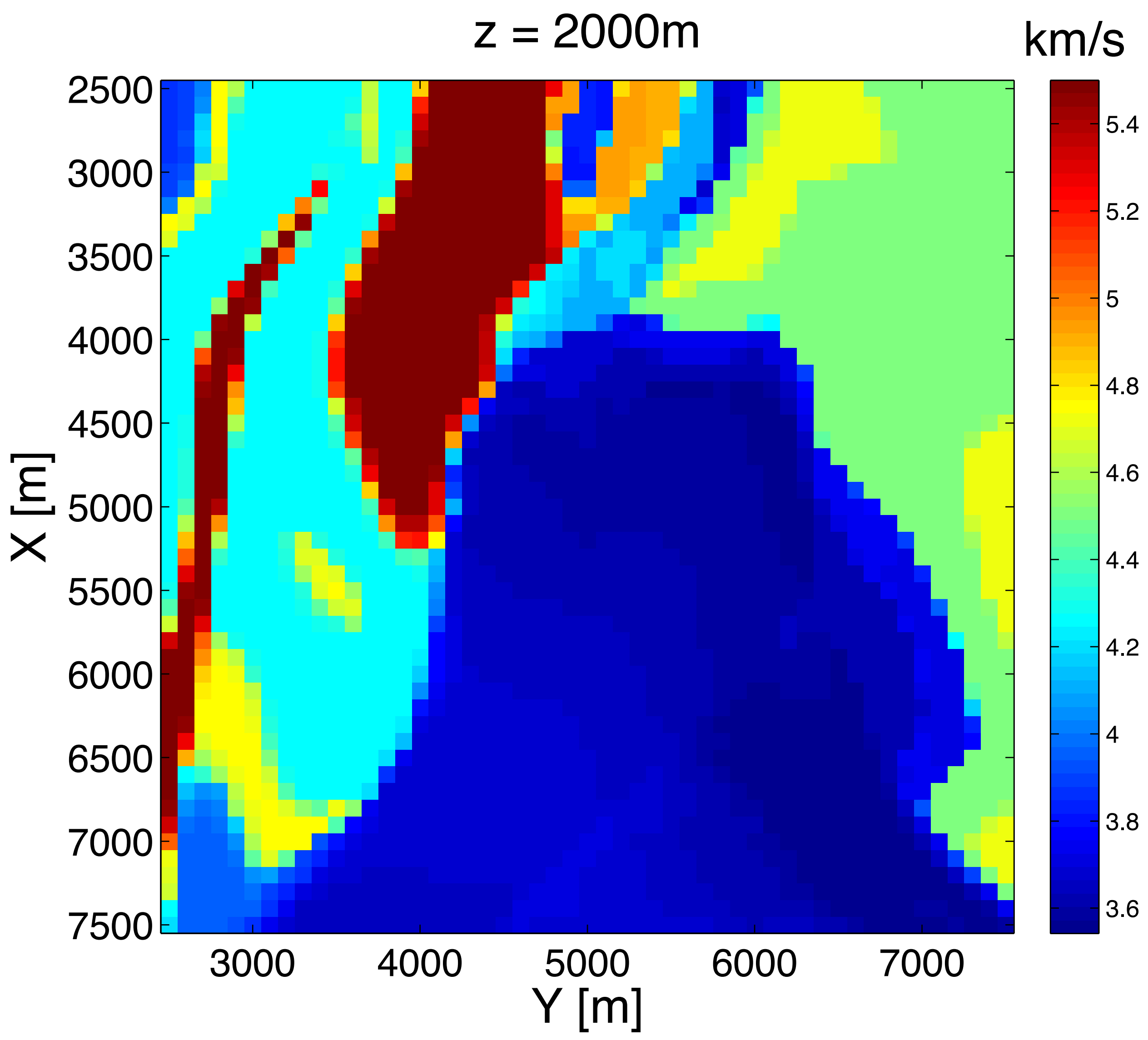
Initial model



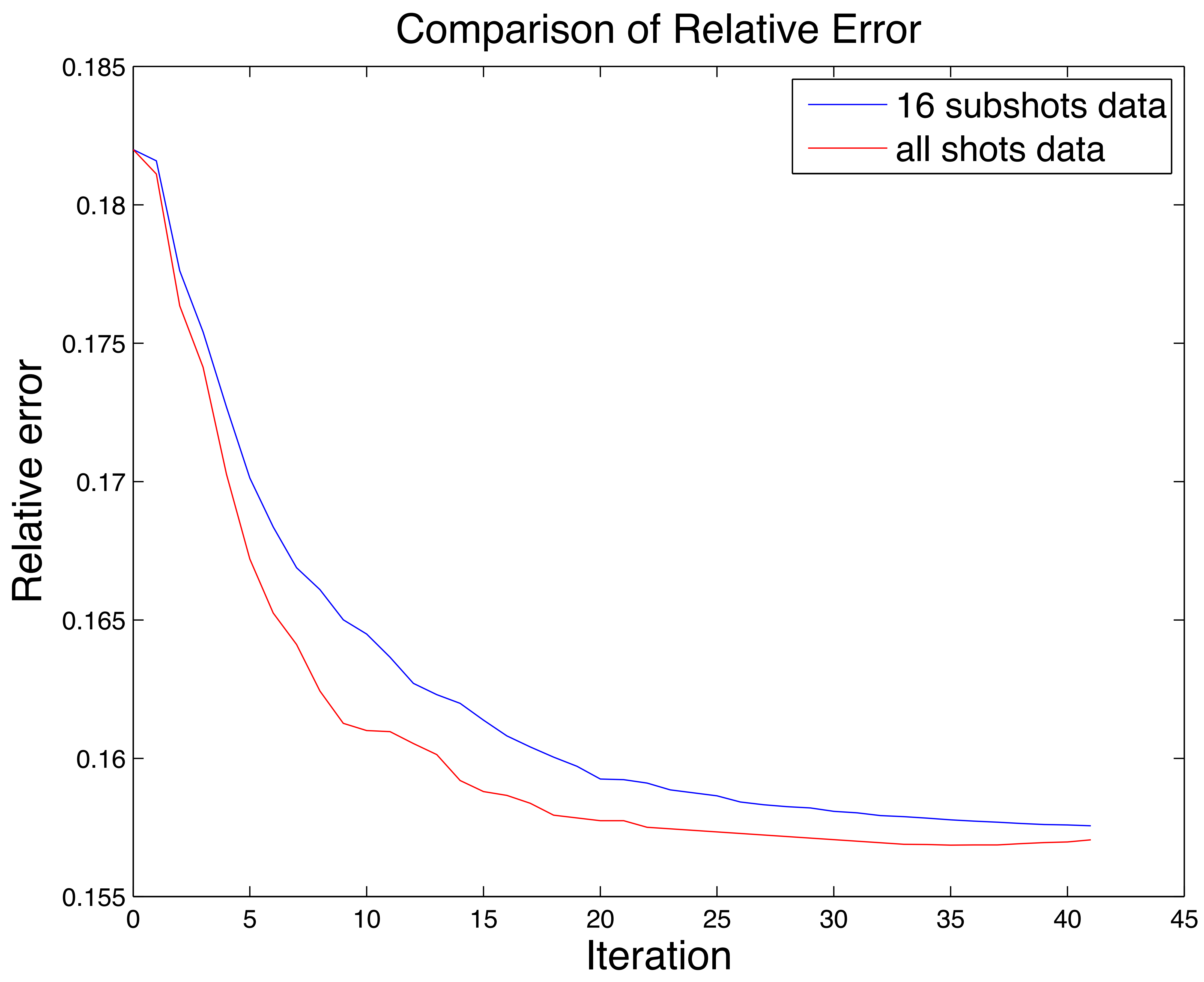
All Shots



True model



Comparison of Model Relative Error



Conclusion

- Swift is a powerful parallel computing tool which can be combined with matlab perfectly.

- Using swift, we can run matlab code on large cluster without the requirement of huge number of licenses.

Future work

- Test 3D FWI using swift with larger problem;
- Apply simultaneous shots for 3D FWI using swift;
- Combine adaptive 3D FWI with swift.

Acknowledgements

Thank you for your attention !

https://www.slim.eos.ubc.ca/

This work was in part financially supported by the Natural Sciences and Engineering Research Council of Canada Discovery Grant (22R81254) and the Collaborative Research and Development Grant DNOISE II (375142-08). This research was carried out as part of the SINBAD II project with support from the following organizations: BG Group, BGP, BP, CGG, Chevron, ConocoPhillips, ION, Petrobras, PGS, Total SA, WesternGeco, and Woodside.