Solving the Helmholtz equation via row-projections

Tristan van Leeuwen (Univ. of BC, Canada)

Dan Gordon (Univ. of Haifa, Israel)

Rachel Gordon (Technion, Israel)

Felix J. Herrmann (Univ. of BC, Canada)

SLIM

University of British Columbia

EAGE

Copenhagen '12

SLIM

Modelling engine for
3D Frequency-domain FWI:

- *work with few sources/ frequencies at each iteration*
- *flexibility in type of equation*
- *robust*
- *parallel*

3D Helmholtz equation:

- *large, sparse, indefinite system*
- *direct factorization not feasible*
- *`standard' preconditioners often fail*
- *successful preconditioners often tailored to specific wave equation*

[Ernst & Martin, 2011]

SLIM

fast, complicated,..

VS.

simple, robust, ...

Overview

- Kaczmarz preconditioning
- Examples
- Parallelization
- 3D Benchmark
- Inversion
- Conclusions

Kaczmarz

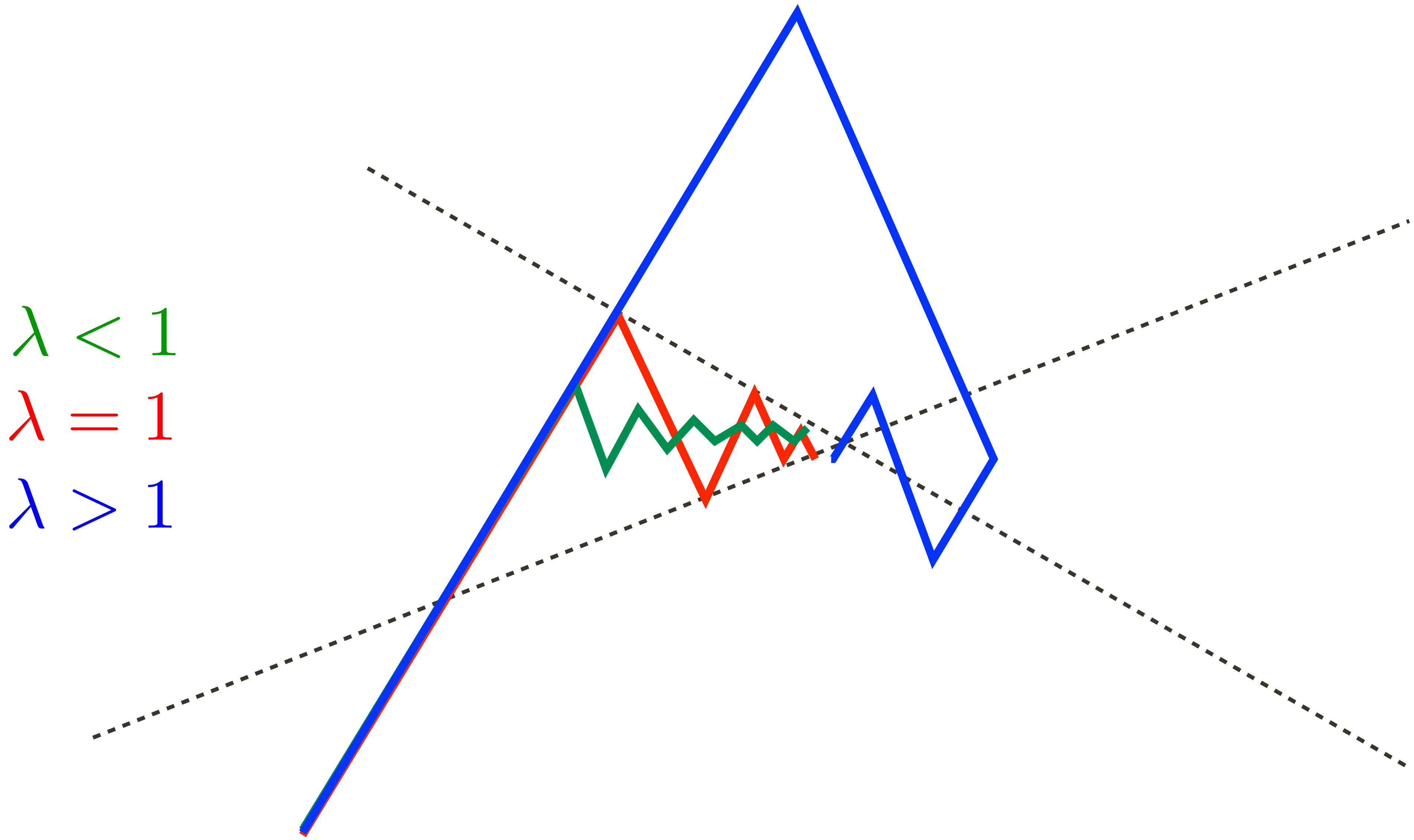
The Kaczmarz method solves a system $A\mathbf{x} = \mathbf{b}$ by successive row projections

$$\mathbf{x} := \mathbf{x} + \frac{\lambda_i}{||\mathbf{a}_i||_2^2} \left(b_i - \mathbf{a}_i^T \mathbf{x}\right) \mathbf{a}_i,$$

with relaxation parameter $0 < \lambda_i < 2$

[Kaczmarz, ’37]

Kaczmarz



Kaczmarz

rewrite:

$$\mathbf{x} := \underbrace{\left(I - \frac{\lambda_i}{||\mathbf{a}_i||_2^2}\mathbf{a}_i\mathbf{a}_i^T\right)}_{Q_i}\mathbf{x} + \frac{\lambda_i}{||\mathbf{a}_i||_2^2} b_i \mathbf{a}_i$$

a double sweep yields

$$\mathbf{x} := \underbrace{(Q_1 Q_2 \ldots Q_n Q_n \ldots Q_1)}_{Q}\mathbf{x} + \underbrace{(\ldots)}_{R}\mathbf{b}$$

Kaczmarz

Find a fixed point by solving

$$(I - Q)\mathbf{x} = R\mathbf{b}$$

where $I - Q$ is symmetric and positive semidefinite, so we can use CG (CGMN).

[Bjork & Elfving, ’79]

Kaczmarz

We never form the matrix explicitly, but compute its action:

Algorithm 1 DKSWP$(A, \mathbf{x}, \mathbf{b}, \lambda) = Q\mathbf{x} + R\mathbf{b}$

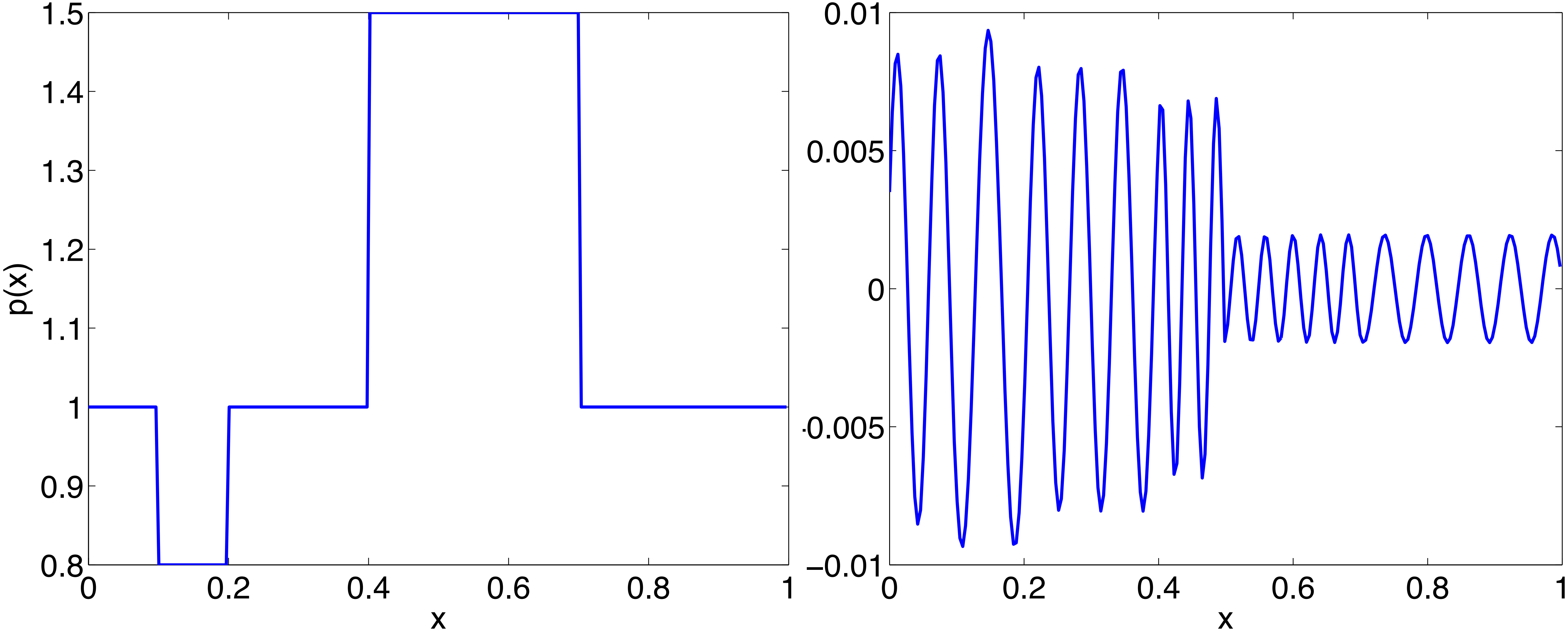
forward sweep
for $i = 1$ to n **do**
  $\mathbf{x} := \mathbf{x} + \lambda(b_i - \mathbf{a}_i^T\mathbf{x})\mathbf{a}_i/||\mathbf{a}_i||_2^2$
end for
backward sweep
for $i = n$ to 1 **do**
  $\mathbf{x} := \mathbf{x} + \lambda(b_i - \mathbf{a}_i^T\mathbf{x})\mathbf{a}_i/||\mathbf{a}_i||_2^2$
end for
return x

Kaczmarz

- low complexity
- low memory (same as original matrix)
- no setup time

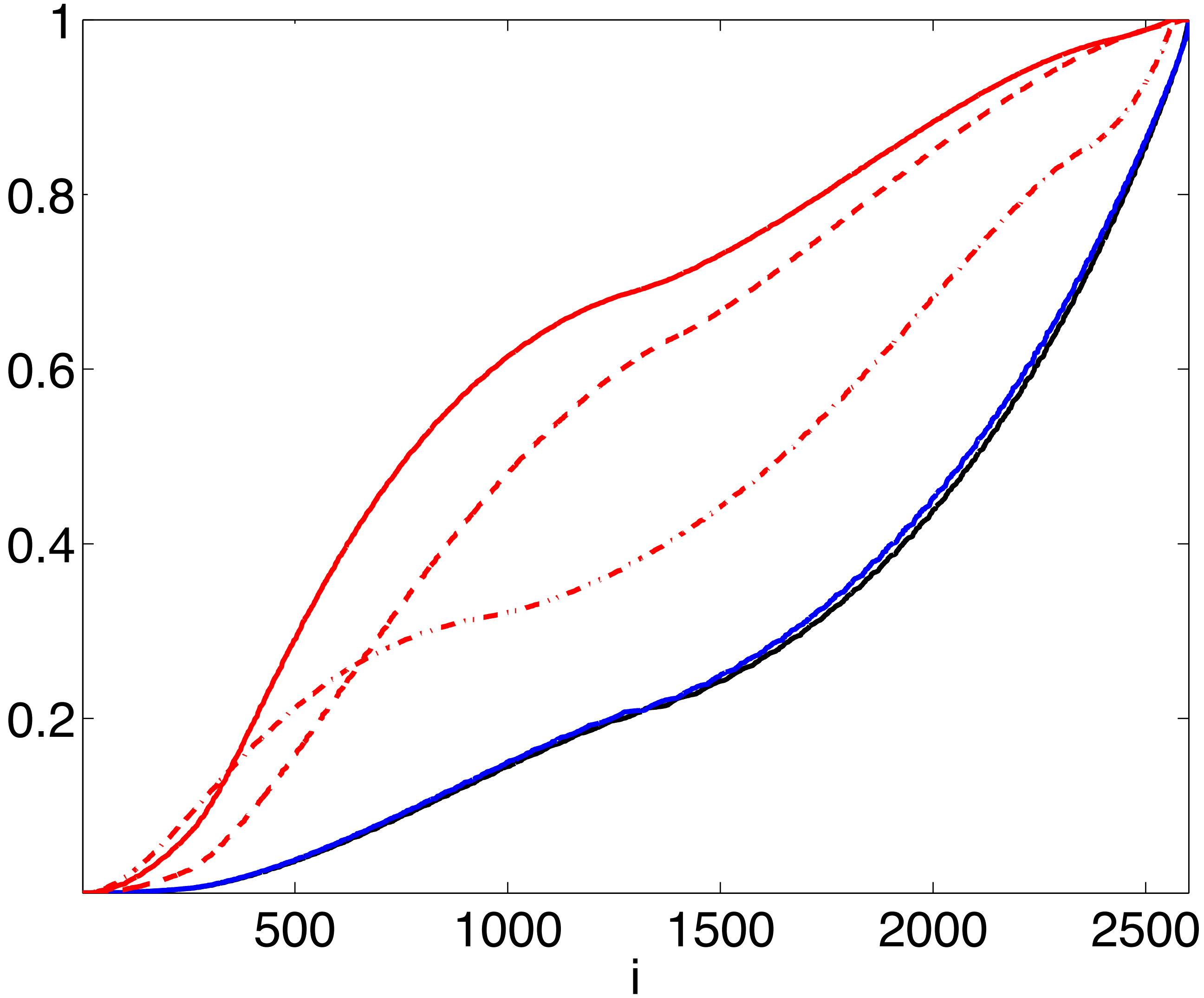
1D results

1D profile, varying k, 10 p/wavelength

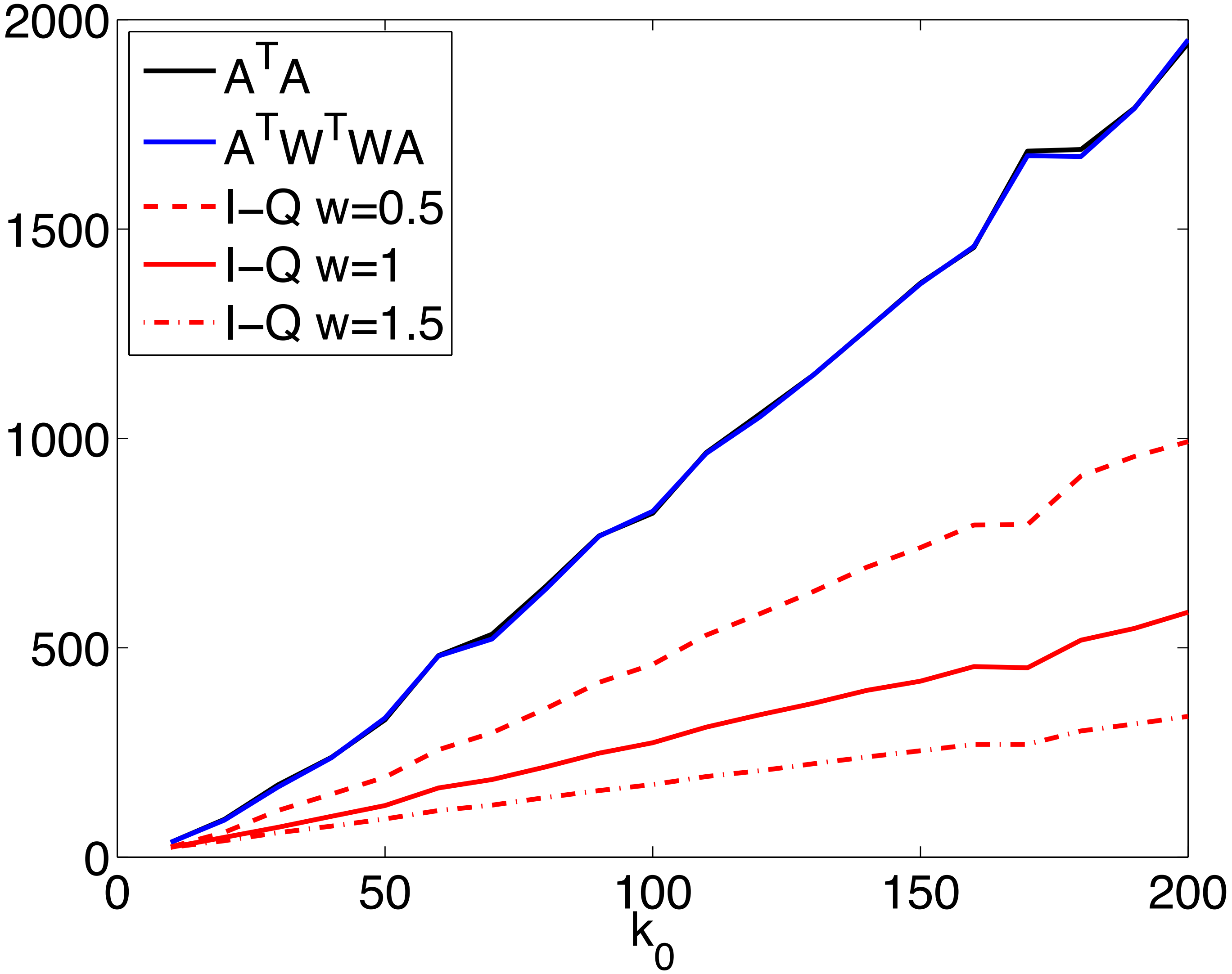


1D results

eigenvalues



# of CG iterations



2D results

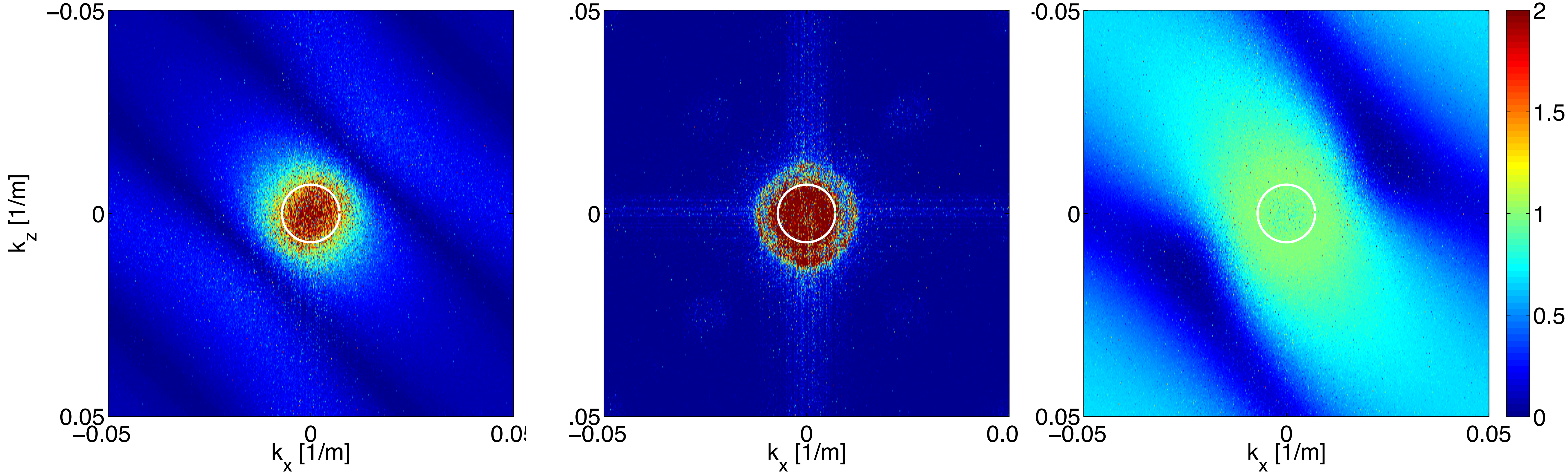
Marmousi, 304 x 1100, f=20, h=10

- CG + Kaczmarz (CGMN)
- BiCGstab + ILU(0)
- SQMR + ML [Bollhofer et al, ’08]

2D results

solve $A\mathbf{r} = 0$ starting from random vector

$$\mathbf{r}_1 = (I - M^{-1}A)\mathbf{r}_0 \qquad\qquad \mathbf{r}_1 = Q\mathbf{r}_0$$

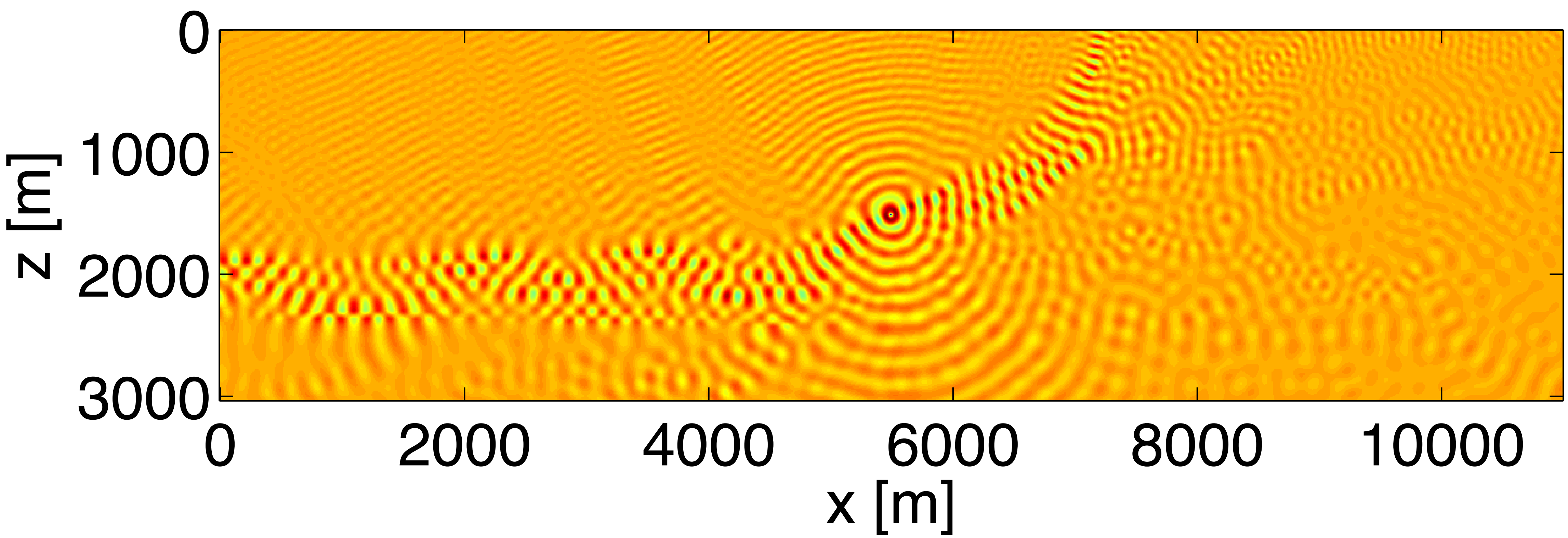


ILU(0) ML Kaczmarz

SLIM

2D results

	iterations	time [s]*
CG + Kaczmarz	5542	603
BiCGstab + ILU(0)	div.	div.
SQMR + ML	514	379



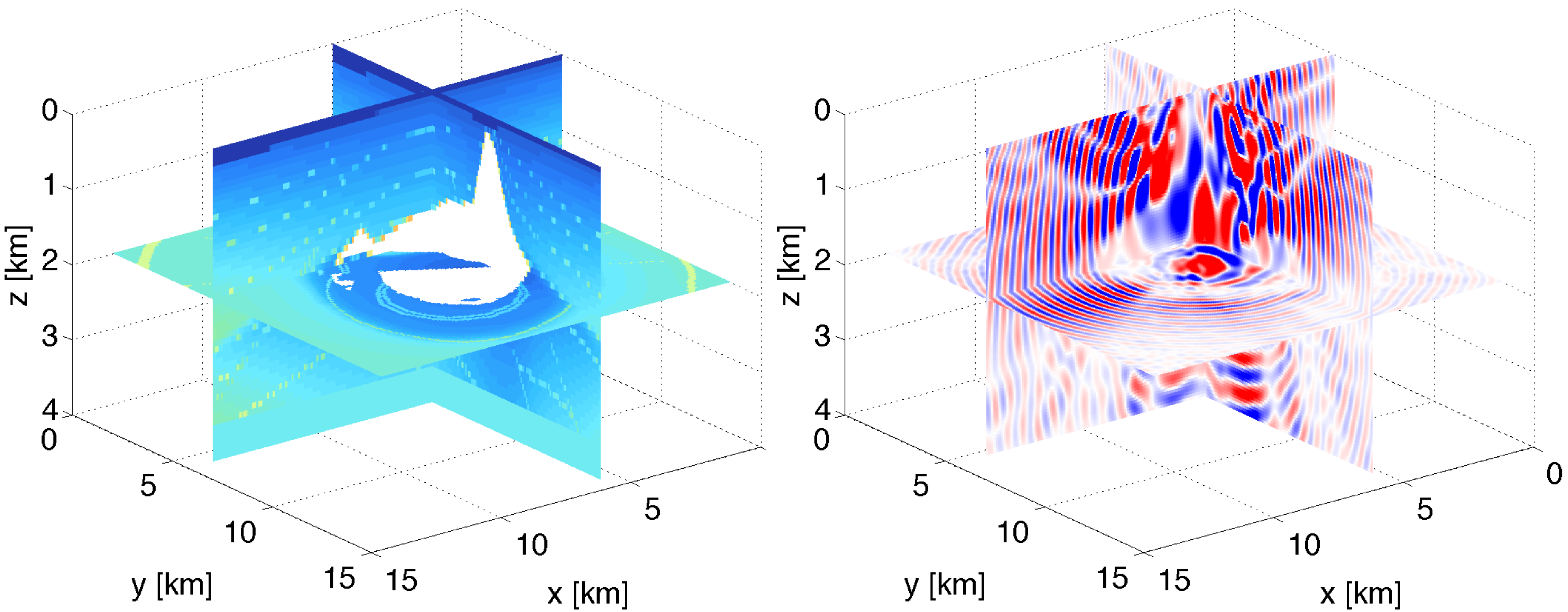
Parallelization

- divide domain in blocks
- Kaczmarz sweeps on blocks are done in parallel (CARP)
- average boundary elements between each sweep
- convergence guaranteed

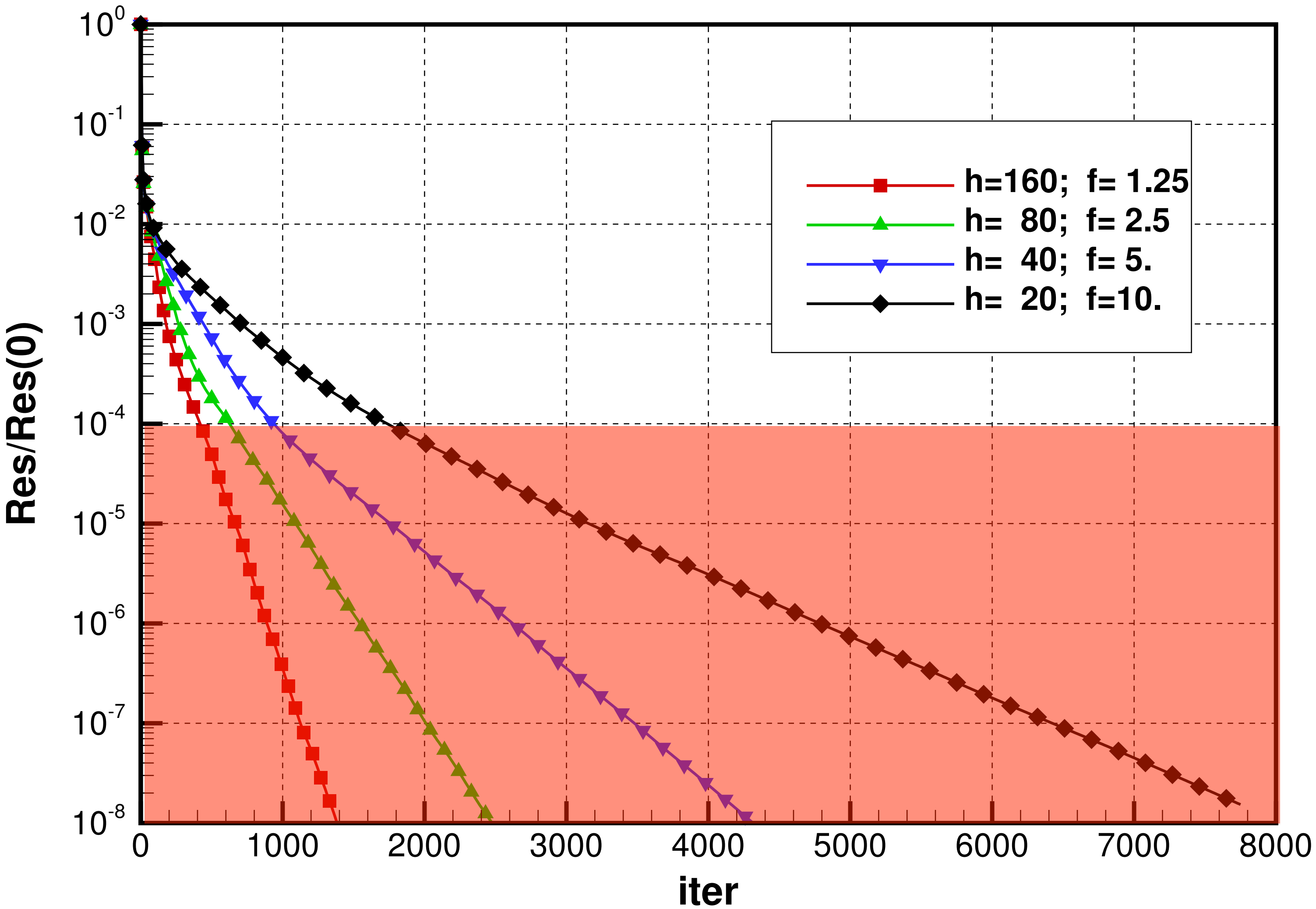
[Gordon & Gordon, ’10]

SEG/EAGE salt

7-point stencil, ABC



SEG/EAGE salt



on 12 processors

f	h	iterations
1.25	160	310
2.5	80	510
5	40	760
10	20	1780

on 1 processor, $\epsilon = 10^{-4}$

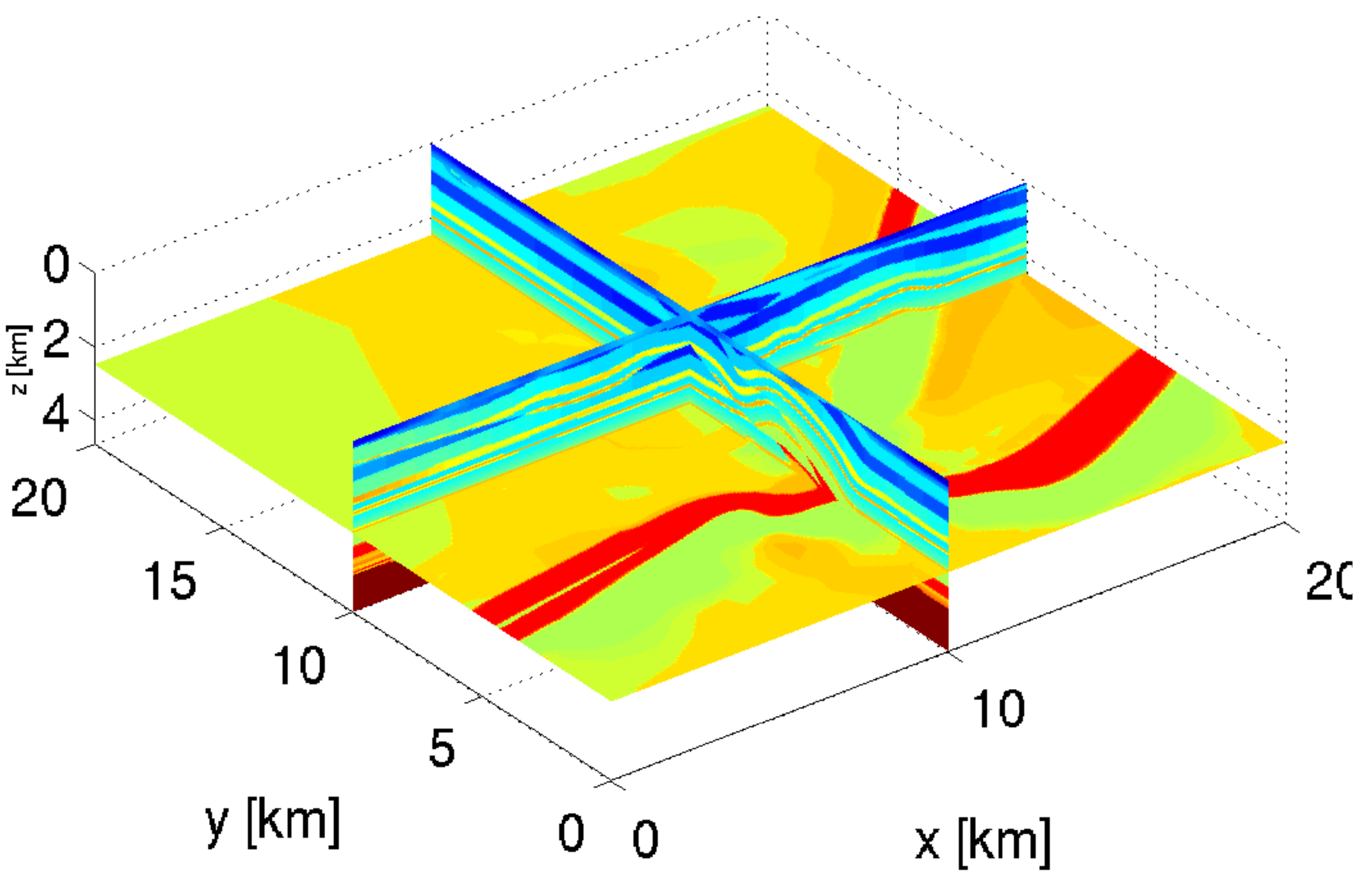
SEG/EAGE salt

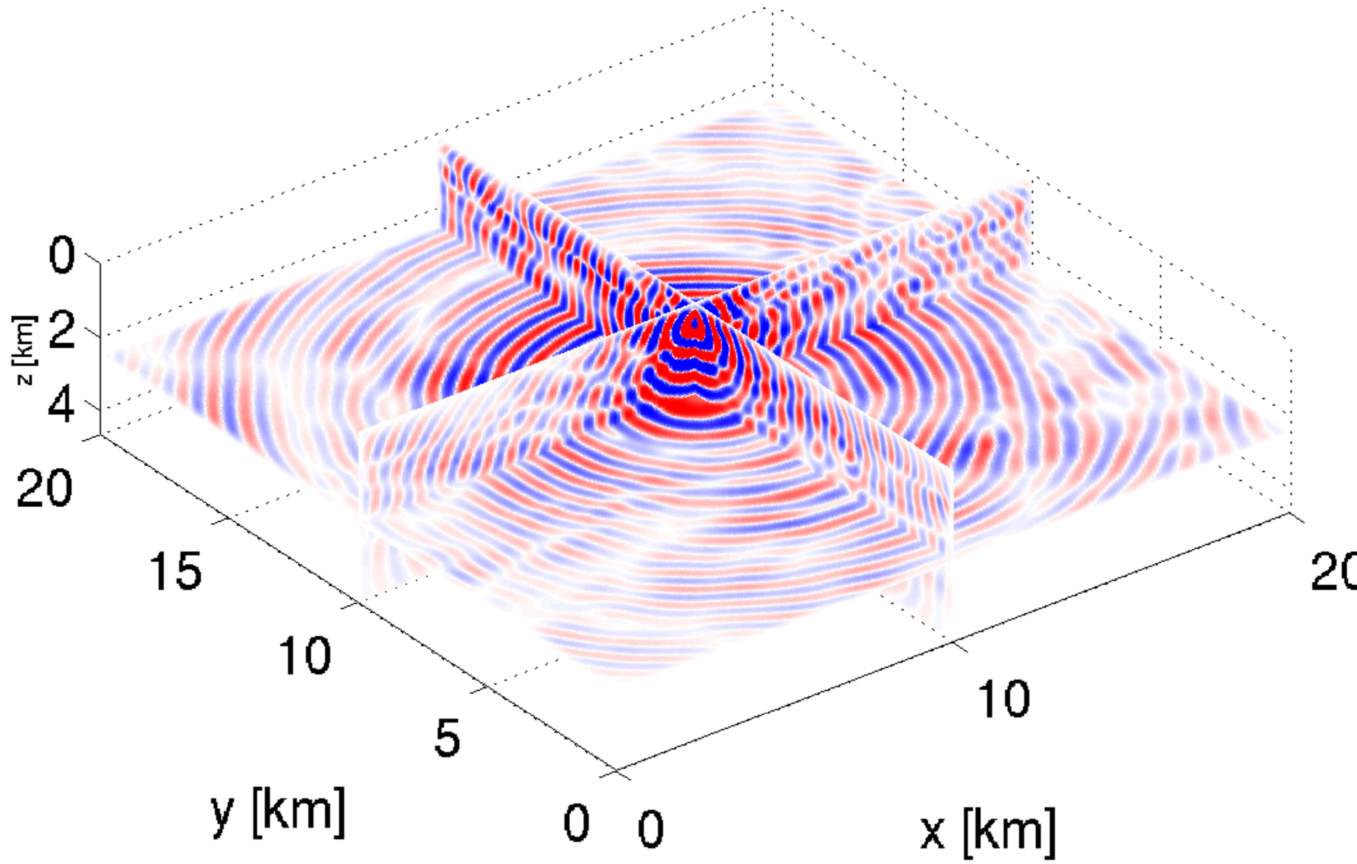
grid: 105 x 338 x 338, h=40, f=5, $\epsilon = 10^{-4}$

np	iter	time (s)	efficiency
1	621	4444.90	1.00
2	619	3091.10	0.72
4	593	1335.00	0.83
8	599	737.90	0.75

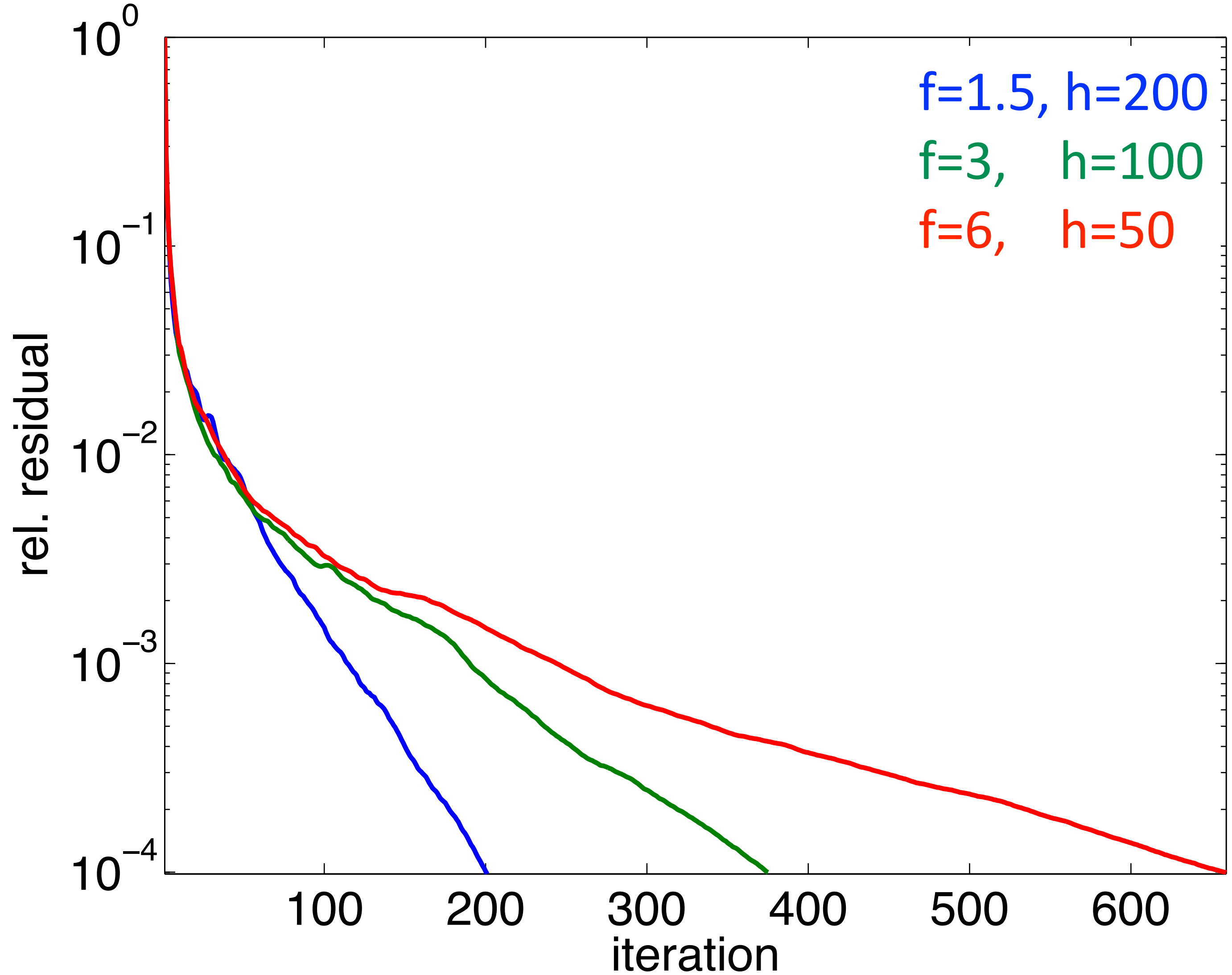
Overthrust

27 point stencil (2^{nd} order), PML





Overthrust



on 1 processor

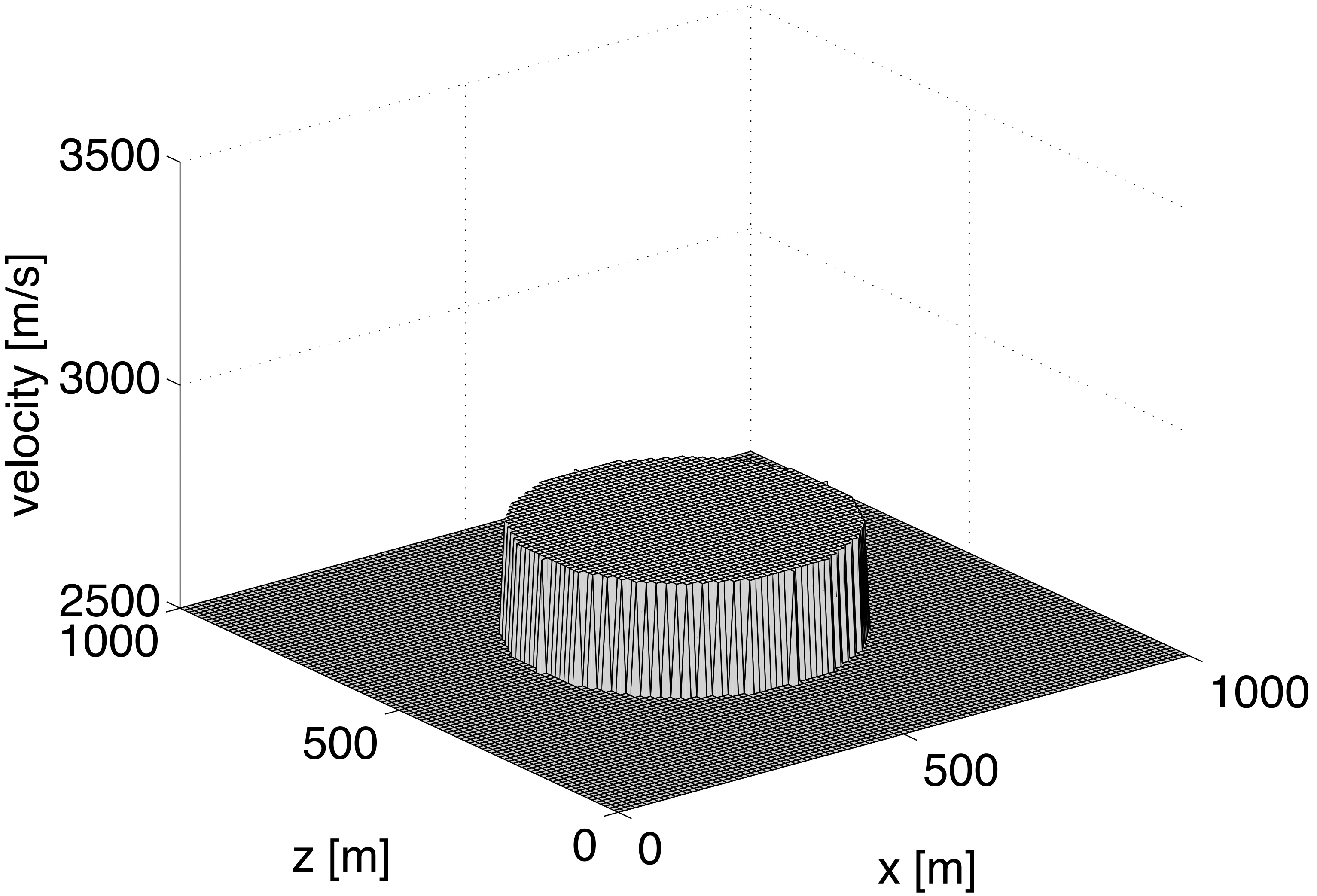
Overthrust

grid: 47x201x201, h=100, f=3 Hz, $\epsilon = 10^{-4}$

np	iter	time	efficiency
1	659	20785.40	1.00
2	657	11306.90	0.92
4	596	4882.50	0.96
8	603	3960.10	0.60

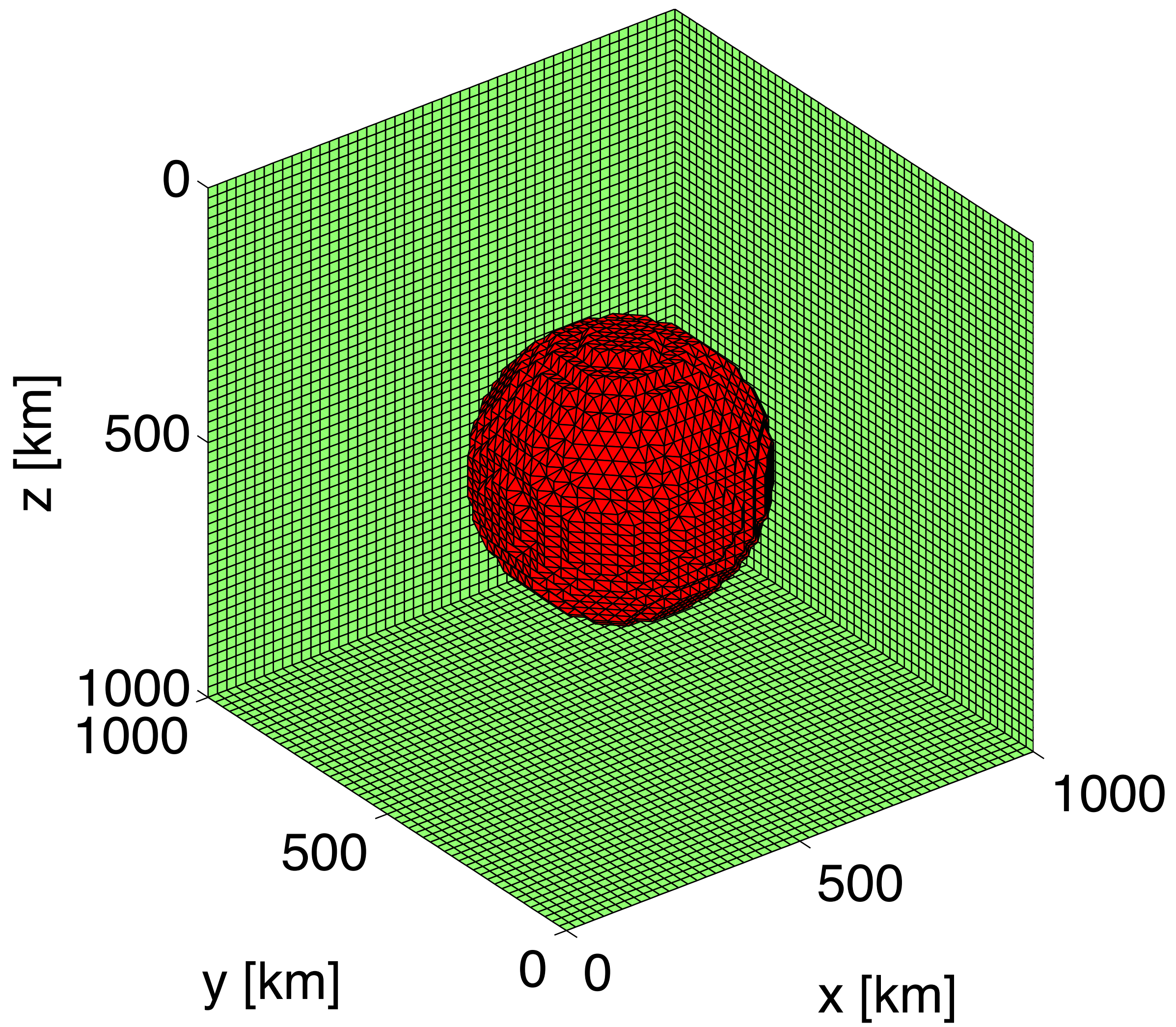
Inversion

Camembert model in 2D ...



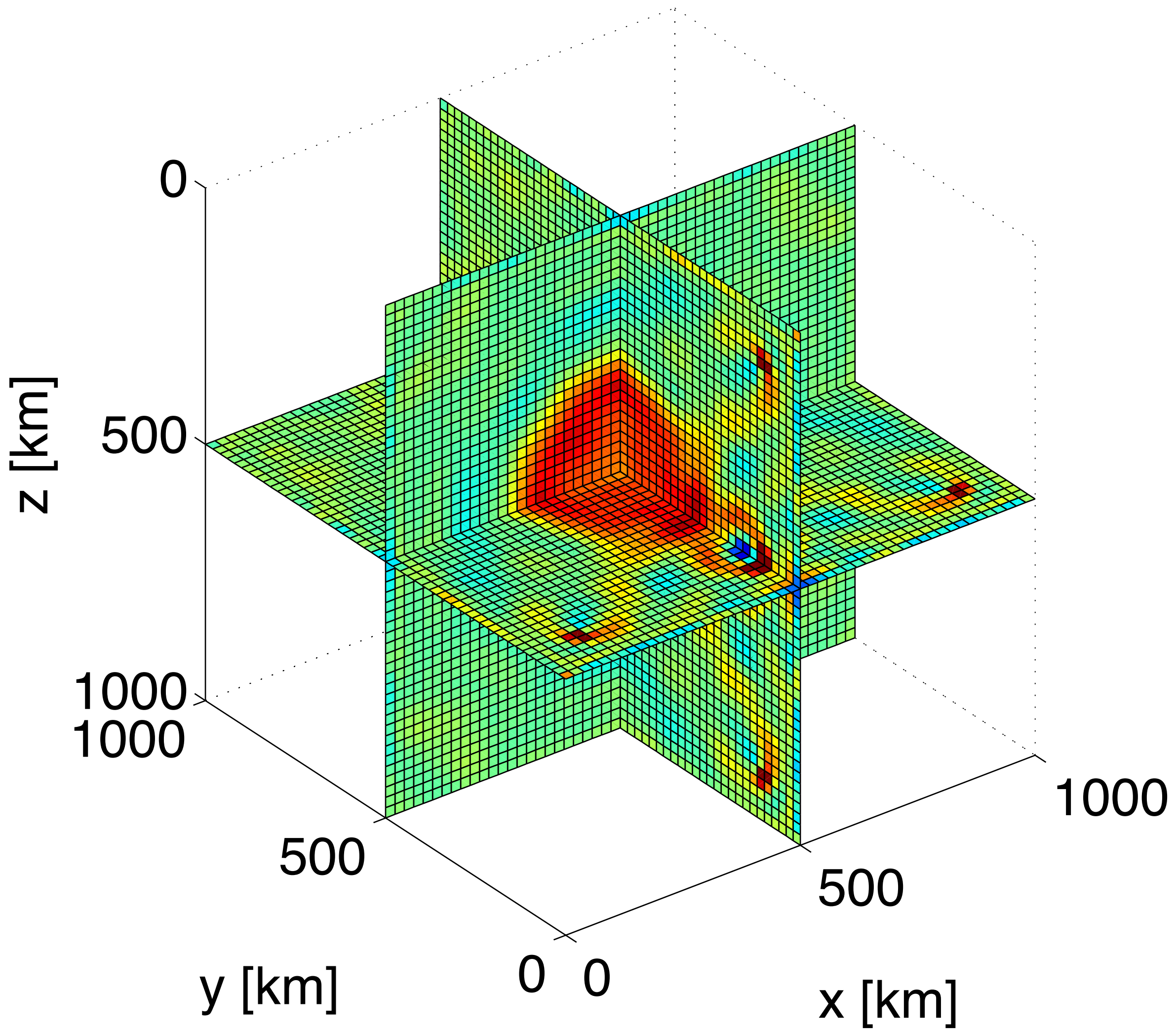
Inversion

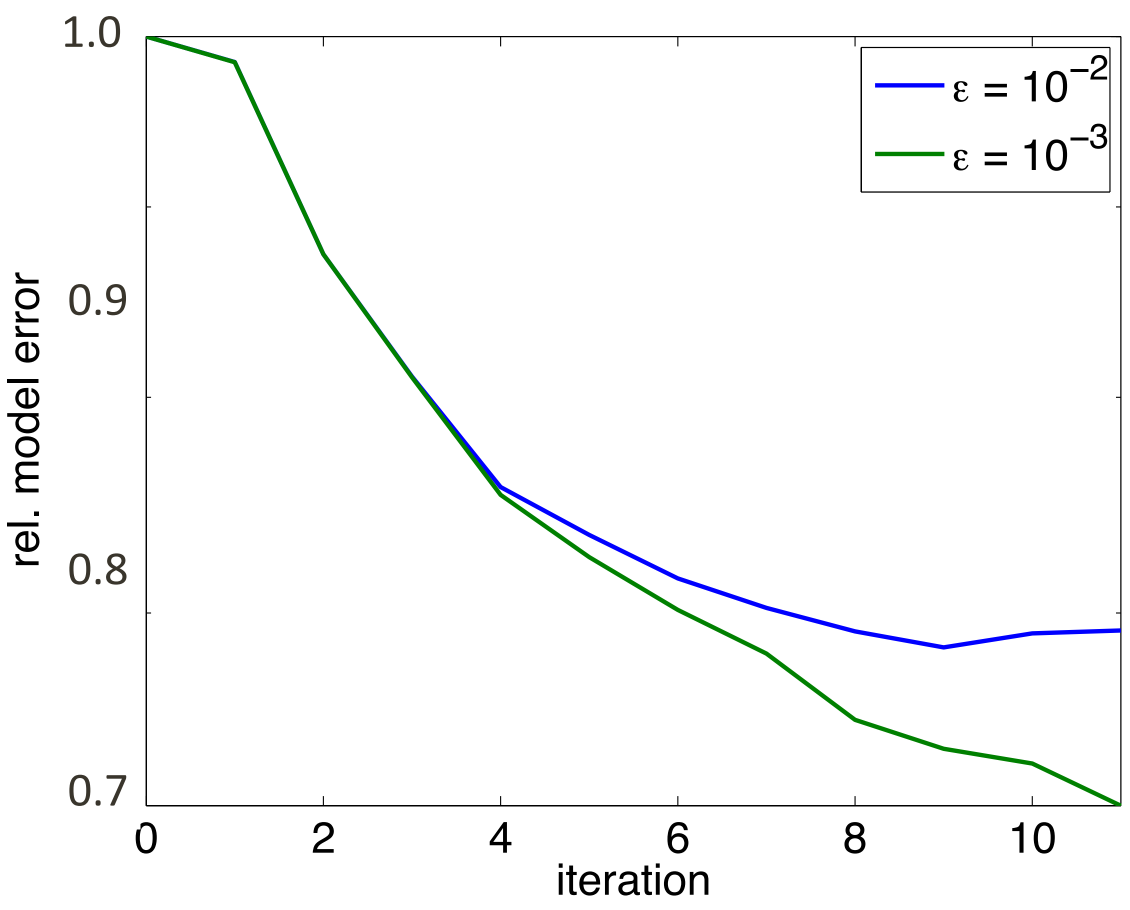
EDAM model in 3D !



Inversion

transmission setup, 9 sources, 3 frequencies





Conclusions

- simple, robust and generic preconditioner
- no overhead, cheap to apply
- easy to parallelize

Future plans

- efficient implementation of sweeps using multi-threading
- investigate BlockCG
- incorporate in inversion
- high-order schemes

http://cs.haifa.ac.il/~gordon/soft.html

Acknowledgements

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, BP, Chevron, ConocoPhillips, Petrobras, Total SA, and WesternGeco.

References

Bollhöfer, M., Grote, M. J., & Schenk, O. (2009). Algebraic Multilevel Preconditioner for the Helmholtz Equation in Heterogeneous Media. *SIAM J. on Sc. Comp.*, *31*(5), 3781.

Gordon, D. and Gordon, R. (2010). CARP-CG: A robust and efficient parallel solver for linear systems, applied to strongly convection dominated PDEs. *Parallel Computing, 36*, 495–515.

Björck, Å. and Elfving, T. (1979) Accelerated projection methods for computing pseudoinverse solutions of systems of linear equations. *BIT, 19*, 145–163.

Kaczmarz, S. [1937] Angenäherte auflösung von systemen linearer gleichungen. *Bulletin International de l'Académie Polonaise des Sciences et des Lettres, 35*, 355–357.

Ernst, O.G. and Martin J. (2011) Why it is Difficult to Solve Helmholtz Problems with Classical Iterative Methods. *Unpublished.*