Accelerating an Iterative Helmholtz Solver with FPGAs

Art Petrenko[1], Tristan van Leeuwen[2], Diego Oriato[3], Simon Tilbury[3] and Felix J. Herrmann[1]

June 18, 2014

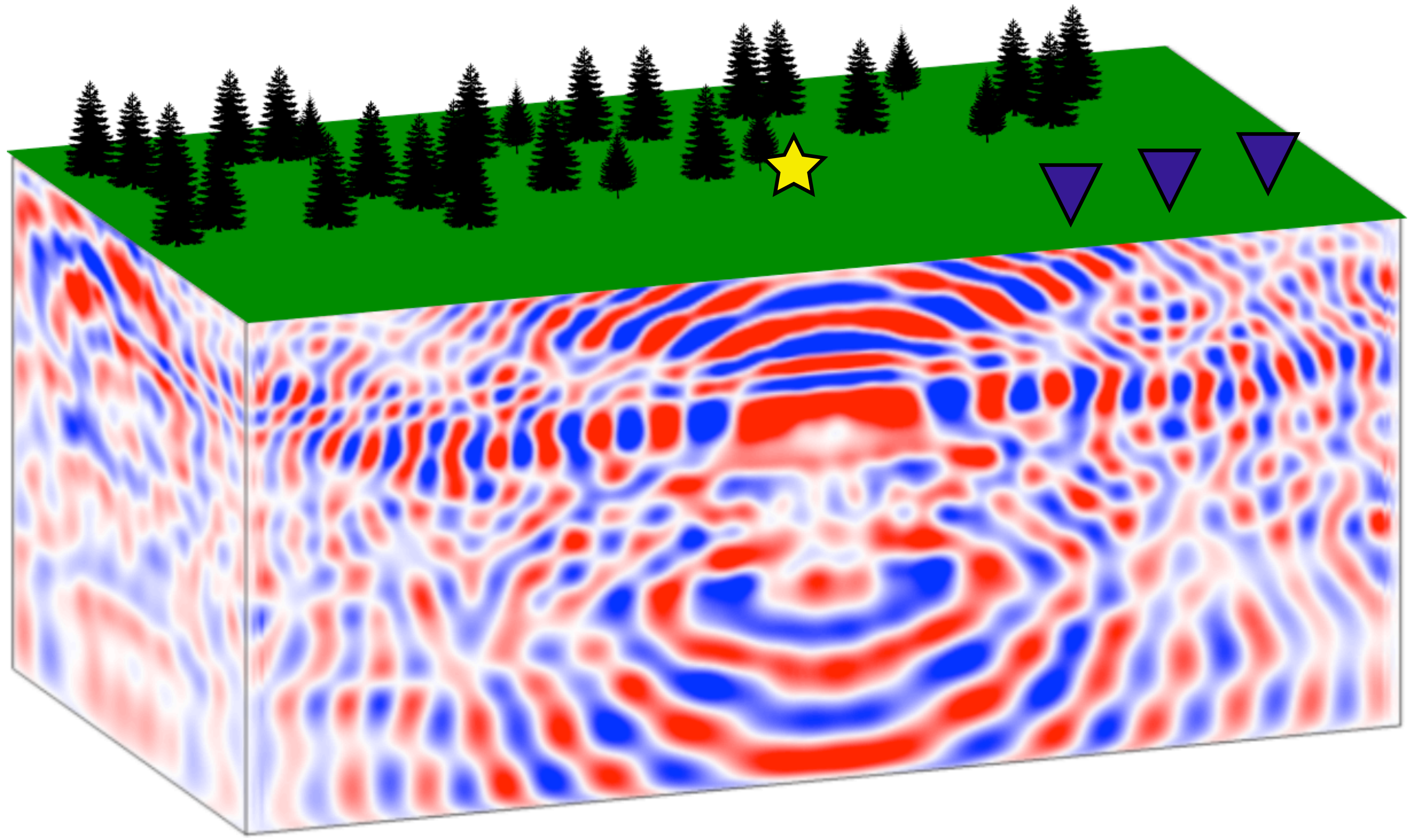
[1]University of British Columbia (UBC)
[2]Centrum Wiskunde & Informatica, formerly with UBC
[3]Maxeler Technologies

SLIM
University of British Columbia

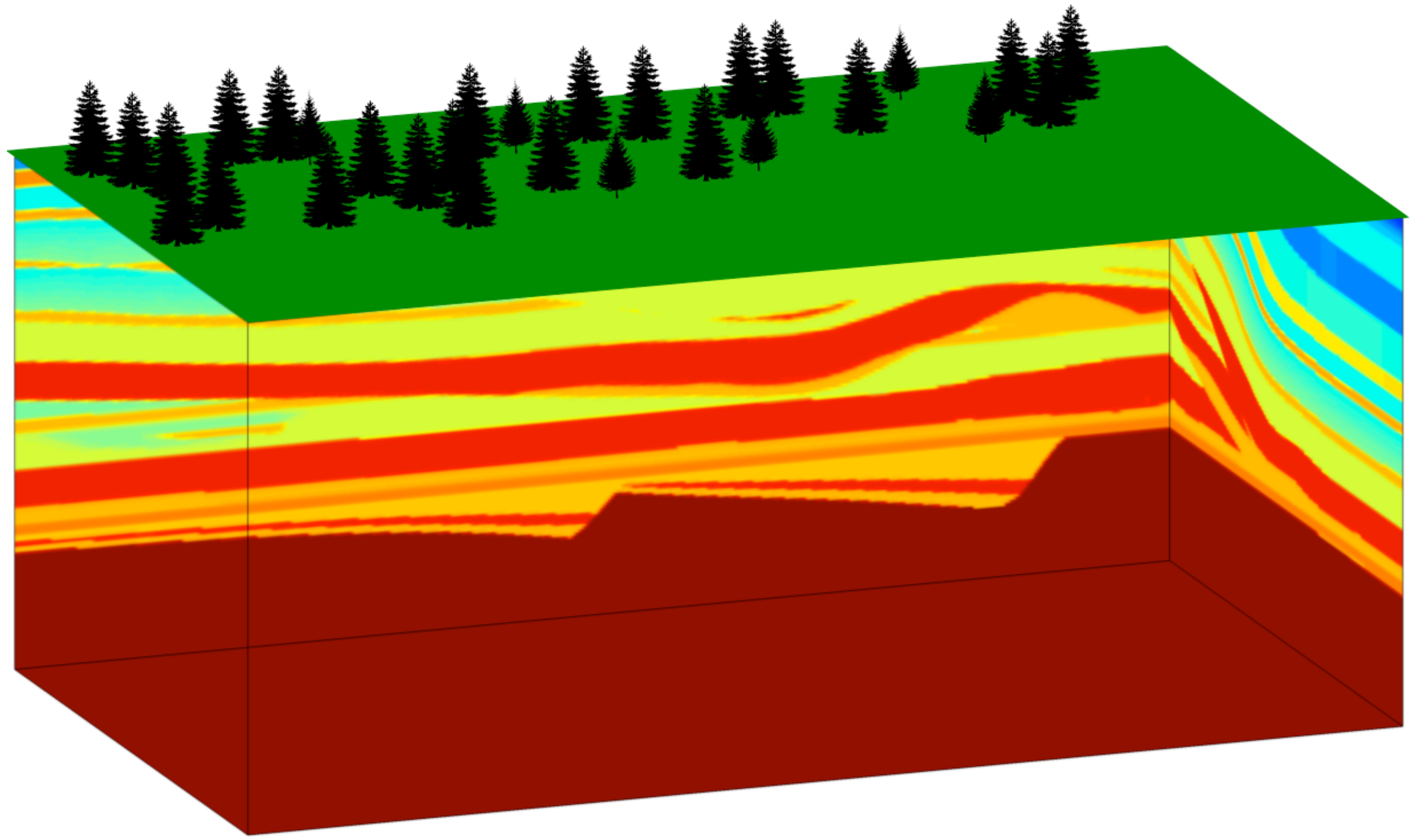
Oh by the way: I have a stutter.

Seismic Wave Simulation

Full-waveform Inversion

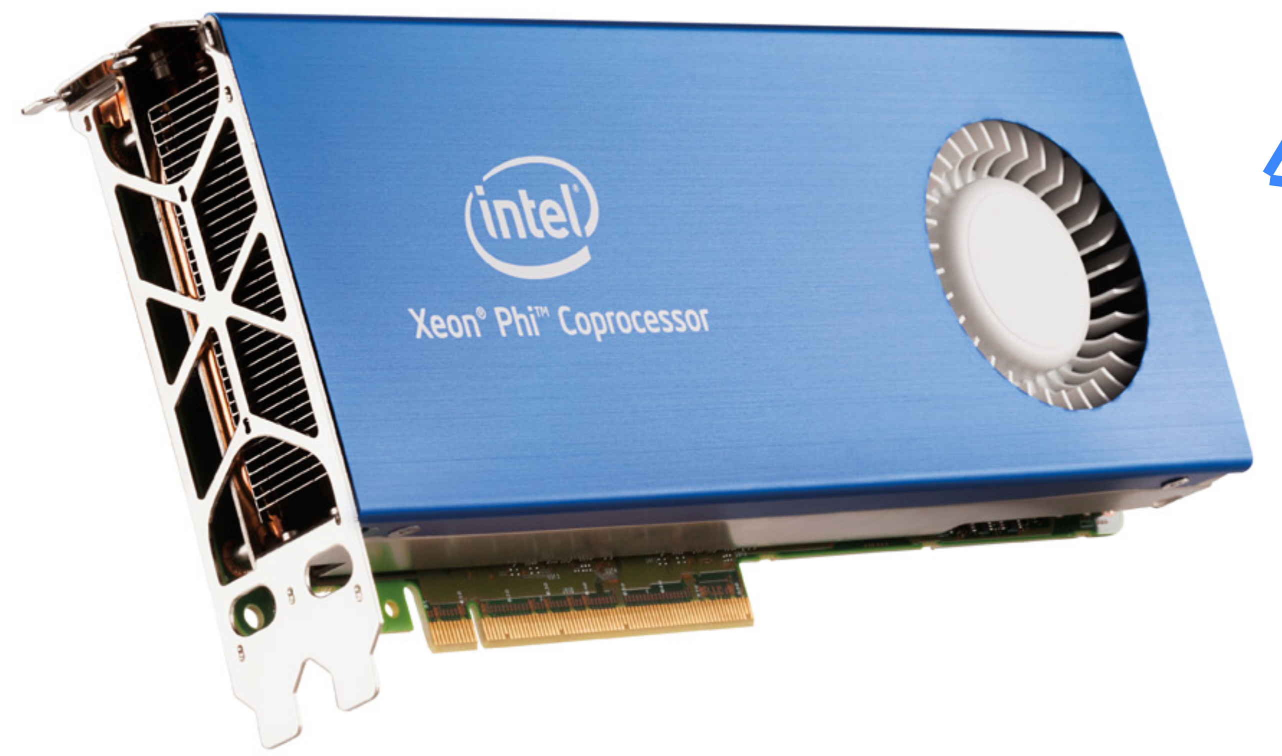


Seismic Wavefield (**u**)



Earth model (**m**)

The Accelerators Have Arrived



Top 10 of "Top 500" Supercomputers

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	National Super Computer Center in Guangzhou China	**Tianhe-2 (MilkyWay-2)** - TH-IVB-FEP Cluster, Intel Xeon E5-2692 12C 2.200GHz, TH Express-2, Intel Xeon Phi 31S1P NUDT	3,120,000	33,862.7	54,902.4	17,808
2	DOE/SC/Oak Ridge National Laboratory United States	**Titan** - Cray XK7 , Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x Cray Inc.	560,640	17,590.0	27,112.5	8,209
3	DOE/NNSA/LLNL United States	**Sequoia** - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM	1,572,864	17,173.2	20,132.7	7,890
4	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect Fujitsu	705,024	10,510.0	11,280.4	12,660
5	DOE/SC/Argonne National Laboratory United States	**Mira** - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	786,432	8,586.6	10,066.3	3,945
6	Swiss National Supercomputing Centre (CSCS) Switzerland	**Piz Daint** - Cray XC30, Xeon E5-2670 8C 2.600GHz, Aries interconnect , NVIDIA K20x Cray Inc.	115,984	6,271.0	7,788.9	2,325
7	Texas Advanced Computing Center/Univ. of Texas United States	**Stampede** - PowerEdge C8220, Xeon E5-2680 8C 2.700GHz, Infiniband FDR, Intel Xeon Phi SE10P Dell	462,462	5,168.1	8,520.1	4,510
8	Forschungszentrum Juelich (FZJ) Germany	**JUQUEEN** - BlueGene/Q, Power BQC 16C 1.600GHz, Custom Interconnect IBM	458,752	5,008.9	5,872.0	2,301
9	DOE/NNSA/LLNL United States	**Vulcan** - BlueGene/Q, Power BQC 16C 1.600GHz, Custom Interconnect IBM	393,216	4,293.3	5,033.2	1,972
10	Leibniz Rechenzentrum Germany	**SuperMUC** - iDataPlex DX360M4, Xeon E5-2680 8C 2.70GHz, Infiniband FDR IBM	147,456	2,897.0	3,185.1	3,423

FPGAs: Reconfigurable Hardware Accelerators

The Punchline

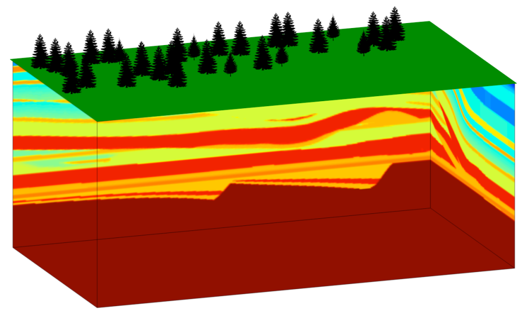
Modelling Seismic Waves
Mathematical Formulation

Modelling Seismic Waves: The Wave Equation

$$\left(\omega^2 \mathbf{m} + \Delta\right)\mathbf{u} = \mathbf{q}$$

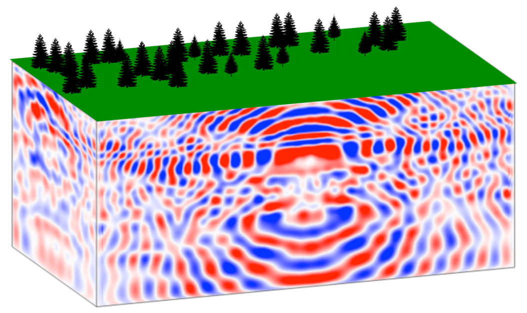
Frequency

Earth Model



Laplacian

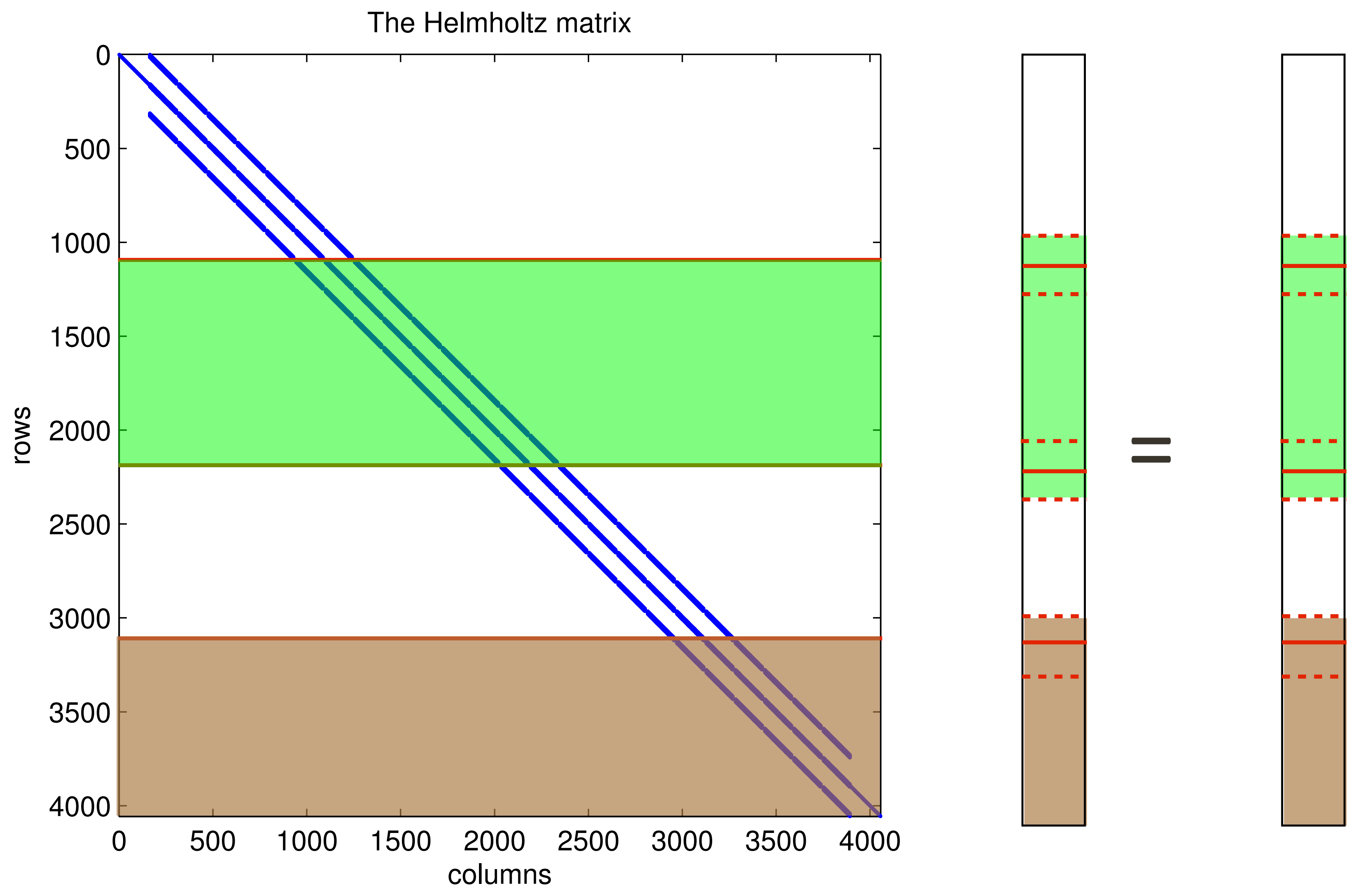
Wavefield (iterate)



Seismic Source (right-hand side)

Modelling Seismic Waves: Discretization
[Operto, 2007]

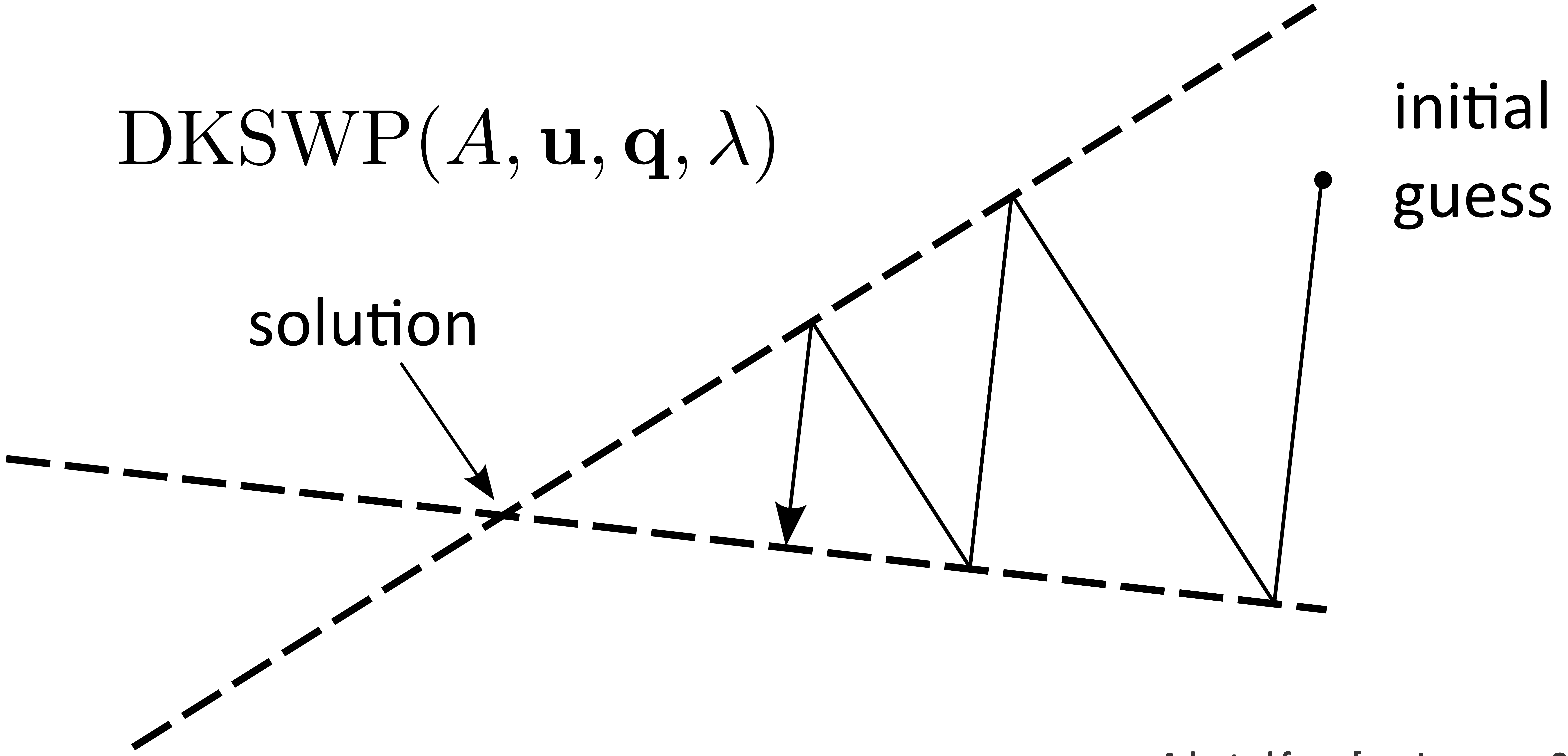
$$A(\mathbf{m}, \omega)\mathbf{u} = \mathbf{q}$$



Solving the Helmholtz System

The Kaczmarz Algorithm
[Kaczmarz, 1937]

$$\mathrm{DKSWP}(A, \mathbf{u}, \mathbf{q}, \lambda)$$



Adapted from [van Leeuwen, 2012]

The Kaczmarz Algorithm: Equivalent to SSOR-NE
[Björck and Elfving, 1979]

Double Kaczmarz sweep on the original system:

$$A\mathbf{u} = \mathbf{q}$$

↔

One iteration of SSOR on the normal equations:

$$AA^*\mathbf{y} = \mathbf{q}$$
$$A^*\mathbf{y} = \mathbf{u}$$

Both are computed as:

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \lambda(b_i - \langle \mathbf{a}_i, \mathbf{u}_k \rangle) \frac{\mathbf{a}_i^*}{\|\mathbf{a}_i\|^2}$$

$$k : 1 \rightarrow 2N$$
$$i : 1 \rightarrow N, N \rightarrow 1$$

Kaczmarz + CG = CGMN

[Björck & Elfving 1979]

CGMN: Solves for Fixed Point of Kaczmarz Row Projections

$$\begin{aligned} \mathrm{DKSWP}(A, \mathbf{u}, \mathbf{q}, \lambda) &= Q_1 \cdots Q_N Q_N \cdots Q_1 \mathbf{u} + R\mathbf{q} \\ &= Q\mathbf{u} + R\mathbf{q}. \end{aligned}$$

Assume **u** is a solution and re-arrange:

$$(I - Q)\mathbf{u} = R\mathbf{q}$$

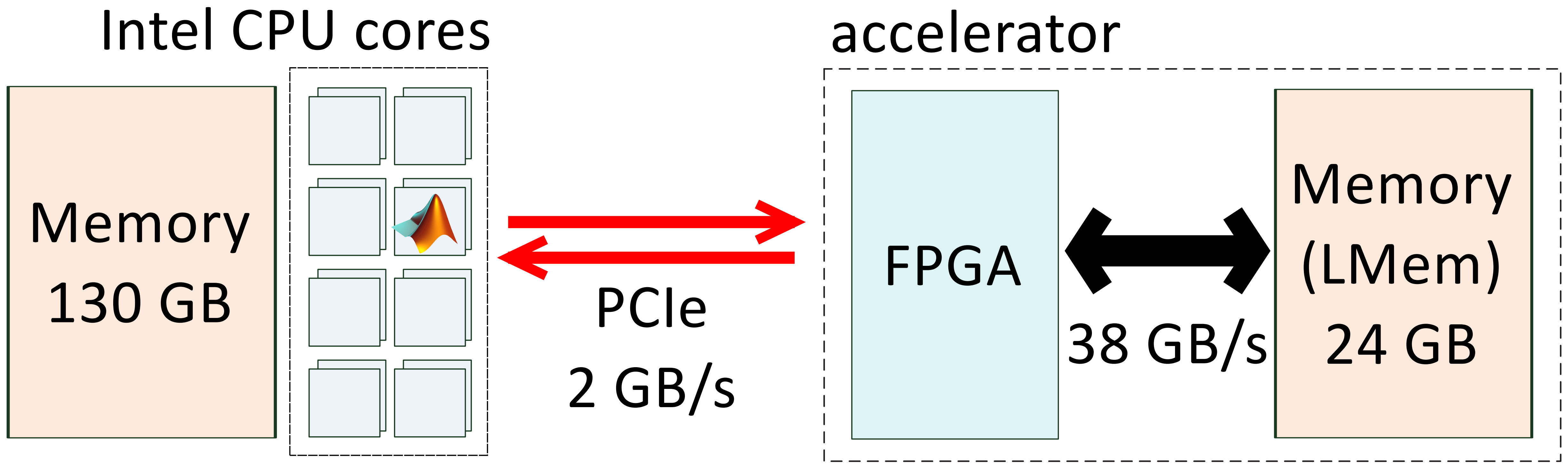
Block parallelization of Kaczmarz + CG = CARP-CG

[Gordon & Gordon, 2010]

Contribution of This Work

Compute Node Overview

[Maxeler Technologies, 2011]



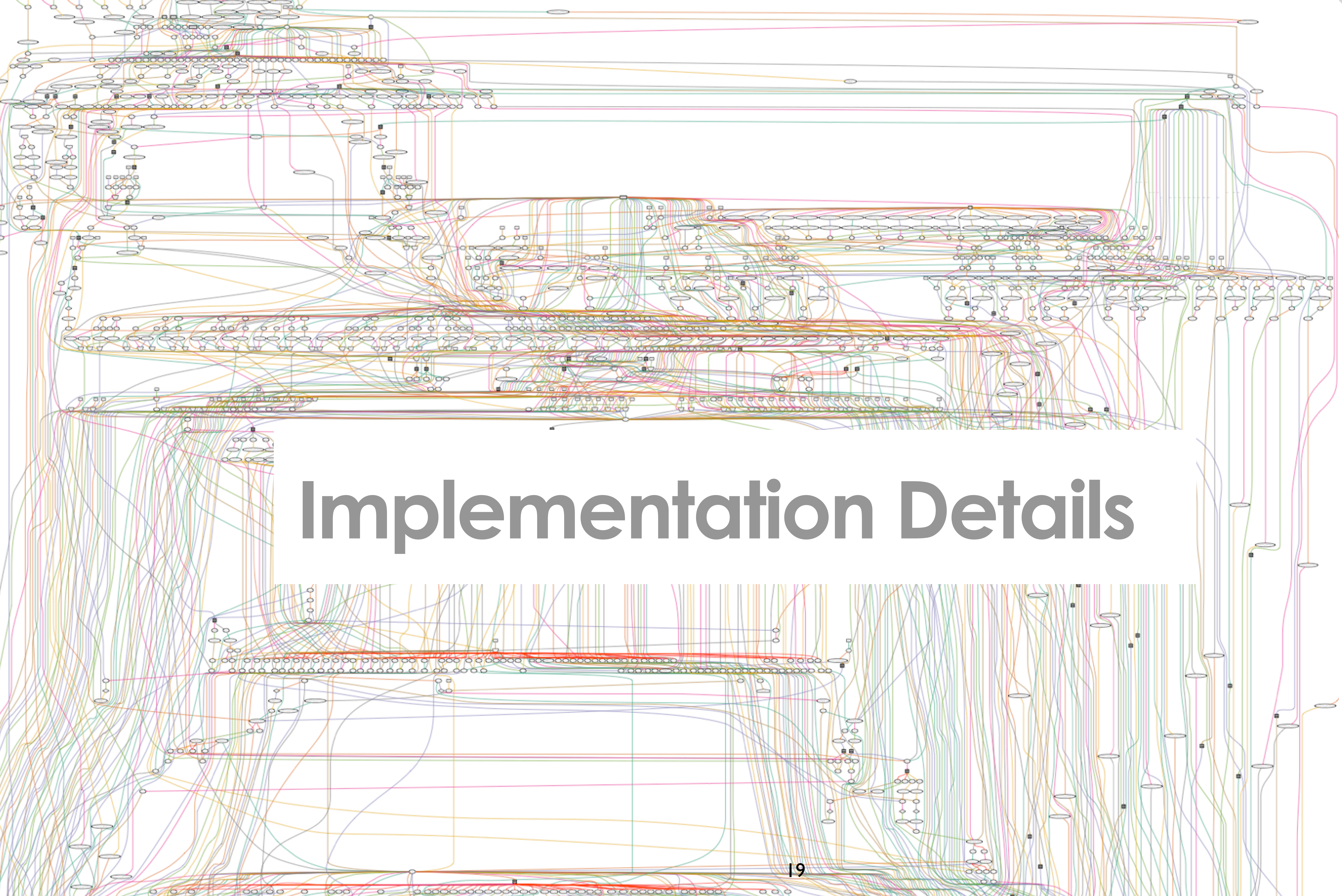
Algorithm 1 CGMN (Björck and Elfving [4])

Input: A, $\mathbf{u}$, $\mathbf{q}$, λ

$R\mathbf{q} \leftarrow \mathrm{DKSWP}(A, \mathbf{0}, \mathbf{q}, \lambda)$
$\mathbf{r} \leftarrow R\mathbf{q} - \mathbf{u} + \mathrm{DKSWP}(A, \mathbf{u}, \mathbf{0}, \lambda)$
$\mathbf{p} \leftarrow \mathbf{r}$
while $\|\mathbf{r}\|^2 > tol$ **do**
  $\mathbf{s} \leftarrow (I - Q)\mathbf{p} = \mathbf{p} - \mathrm{DKSWP}(A, \mathbf{p}, \mathbf{0}, \lambda)$
  $\alpha \leftarrow \|\mathbf{r}\|^2 \,/\, \langle \mathbf{p}, \mathbf{s} \rangle$
  $\mathbf{u} \leftarrow \mathbf{u} + \alpha \mathbf{p}$
  $\mathbf{r} \leftarrow \mathbf{r} - \alpha \mathbf{s}$
  $\beta \leftarrow \|\mathbf{r}\|^2_{\mathrm{curr}} \,/\, \|\mathbf{r}\|^2_{\mathrm{prev}}$
  $\|\mathbf{r}\|^2_{\mathrm{prev}} \leftarrow \|\mathbf{r}\|^2_{\mathrm{curr}}$
  $\mathbf{p} \leftarrow \mathbf{r} + \beta \mathbf{p}$
end while

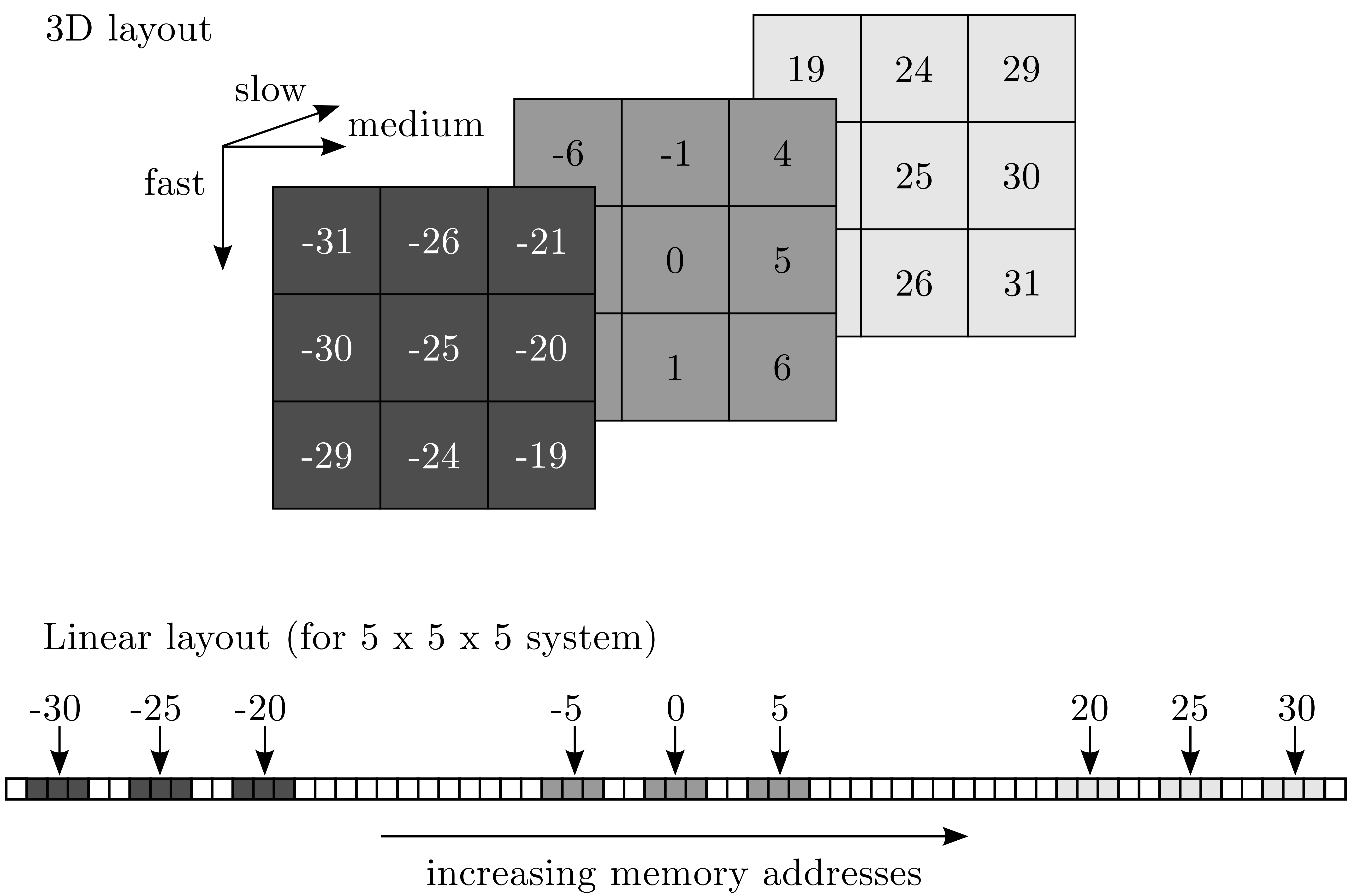
Output: $\mathbf{u}$

Kernel: running on accelerator

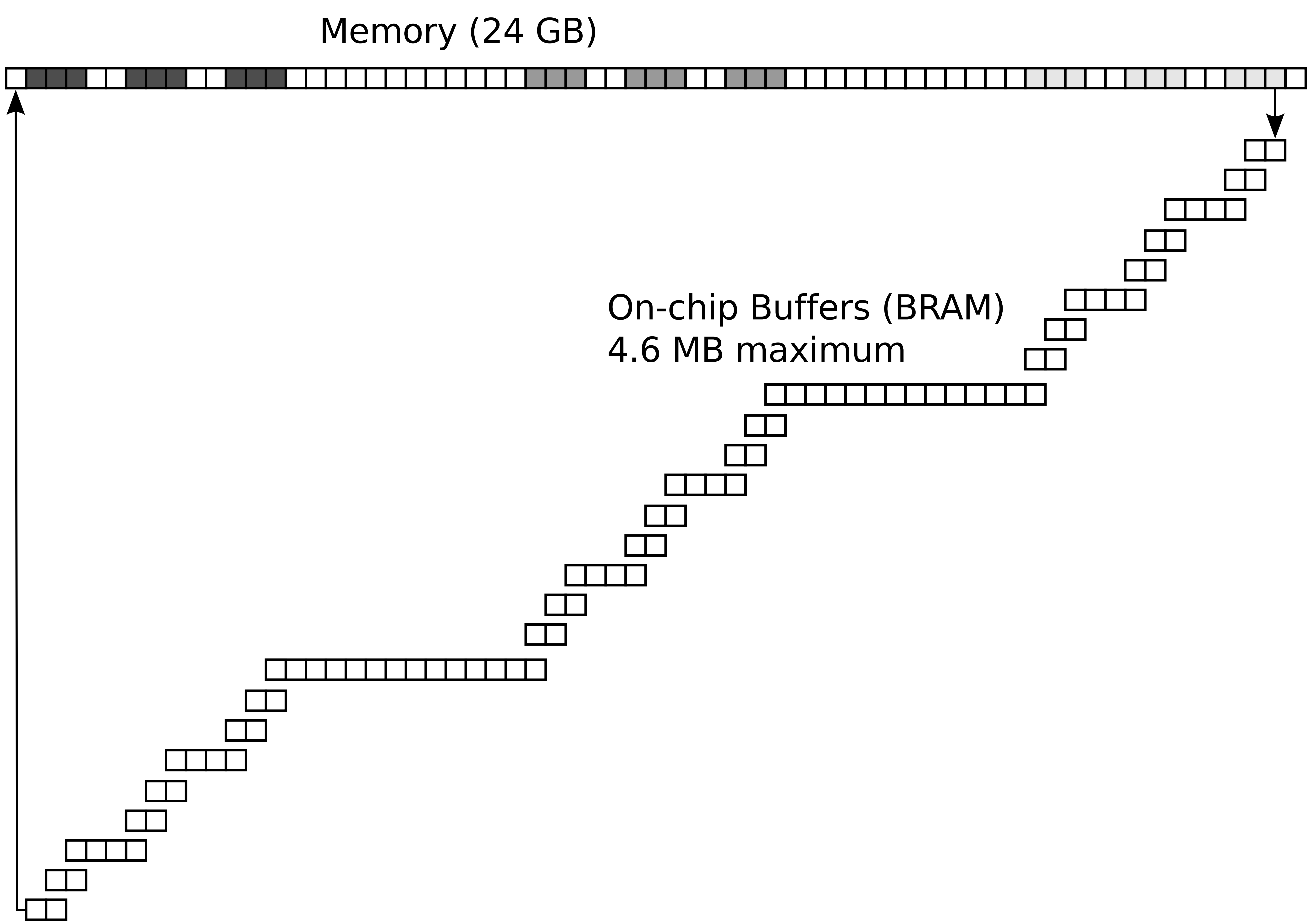


Implementation Details

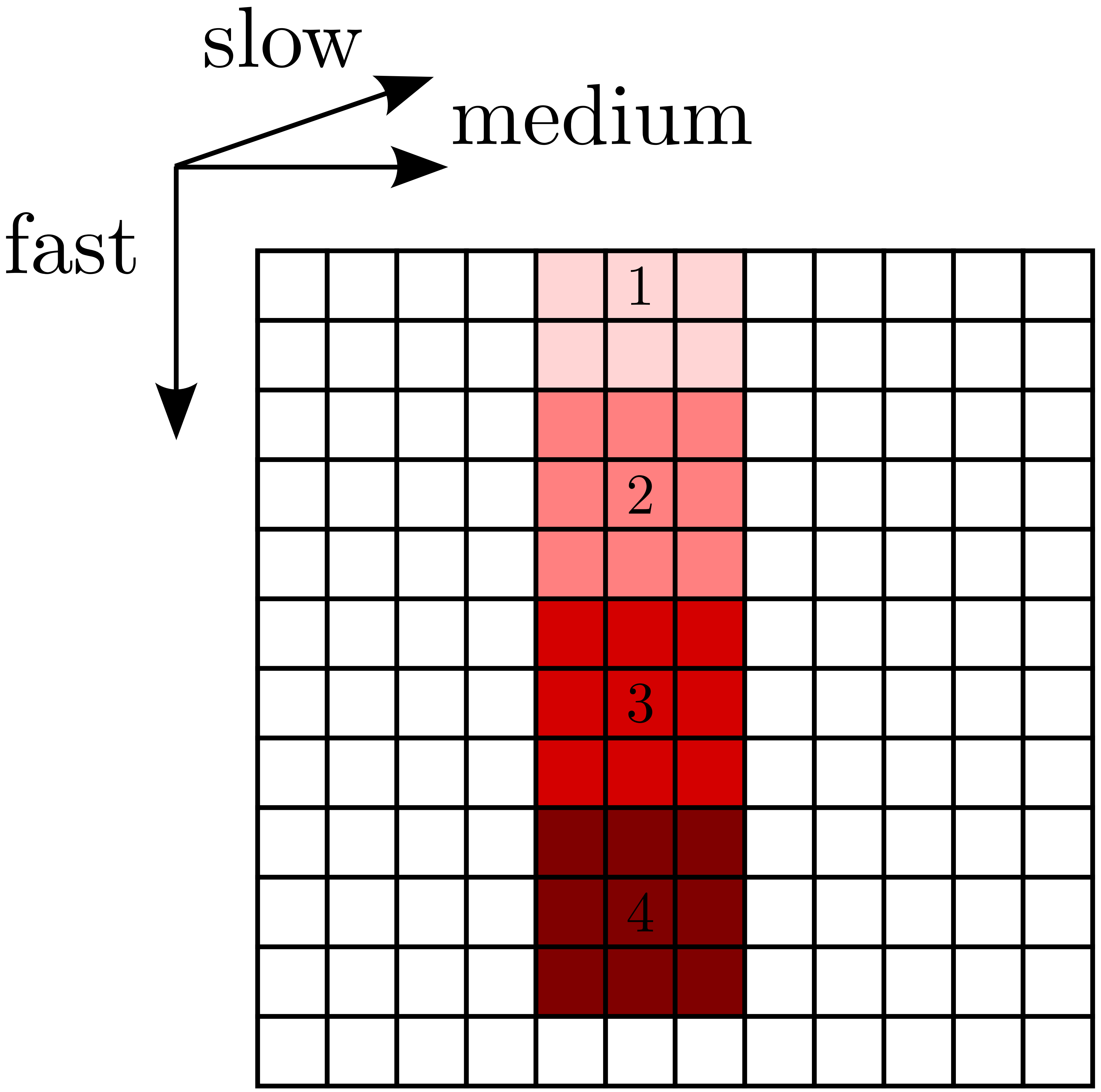
Layout of 3D Wavefields in 1D Memory



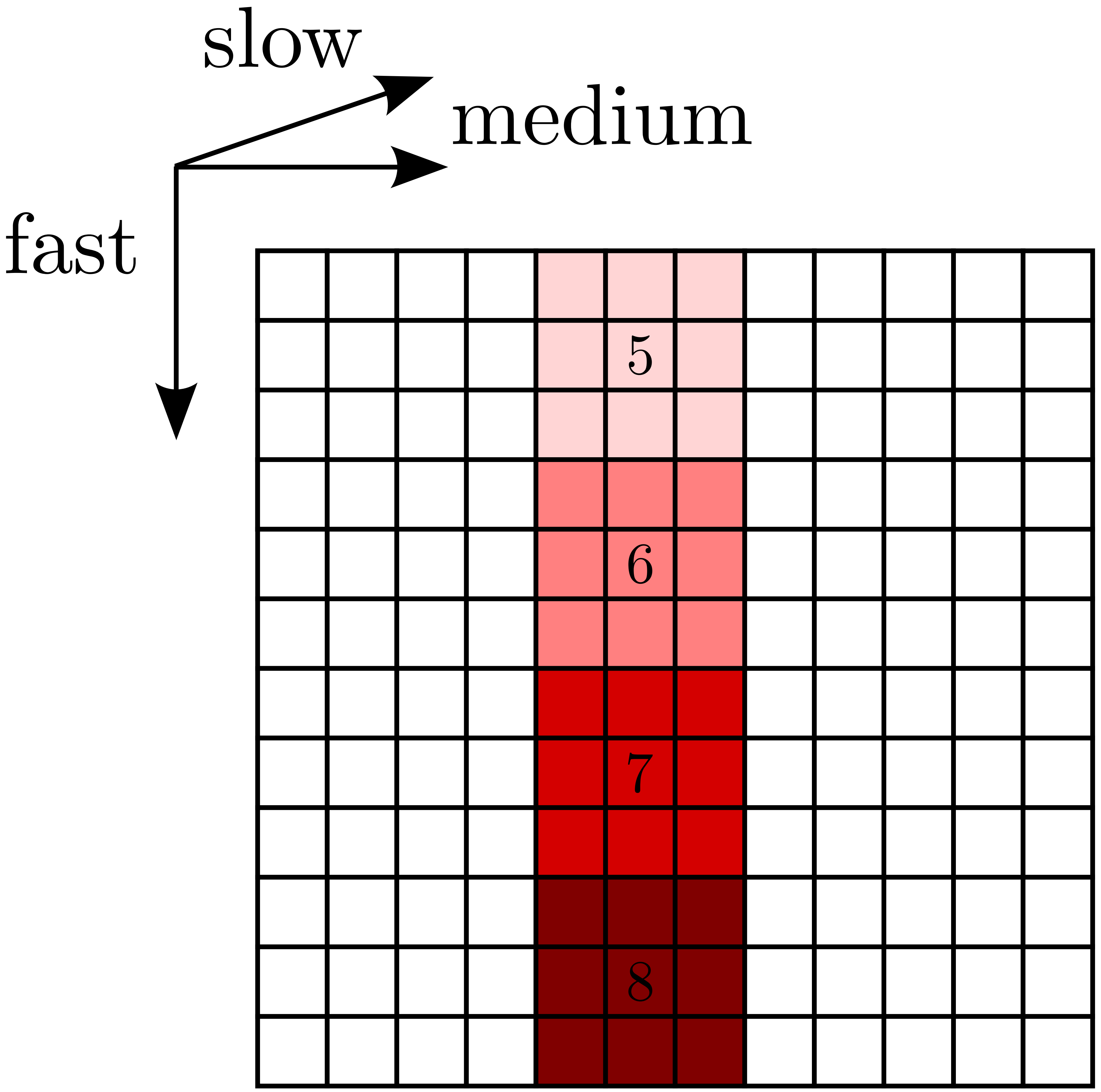
Buffering: Overcoming Latency of Memory Access



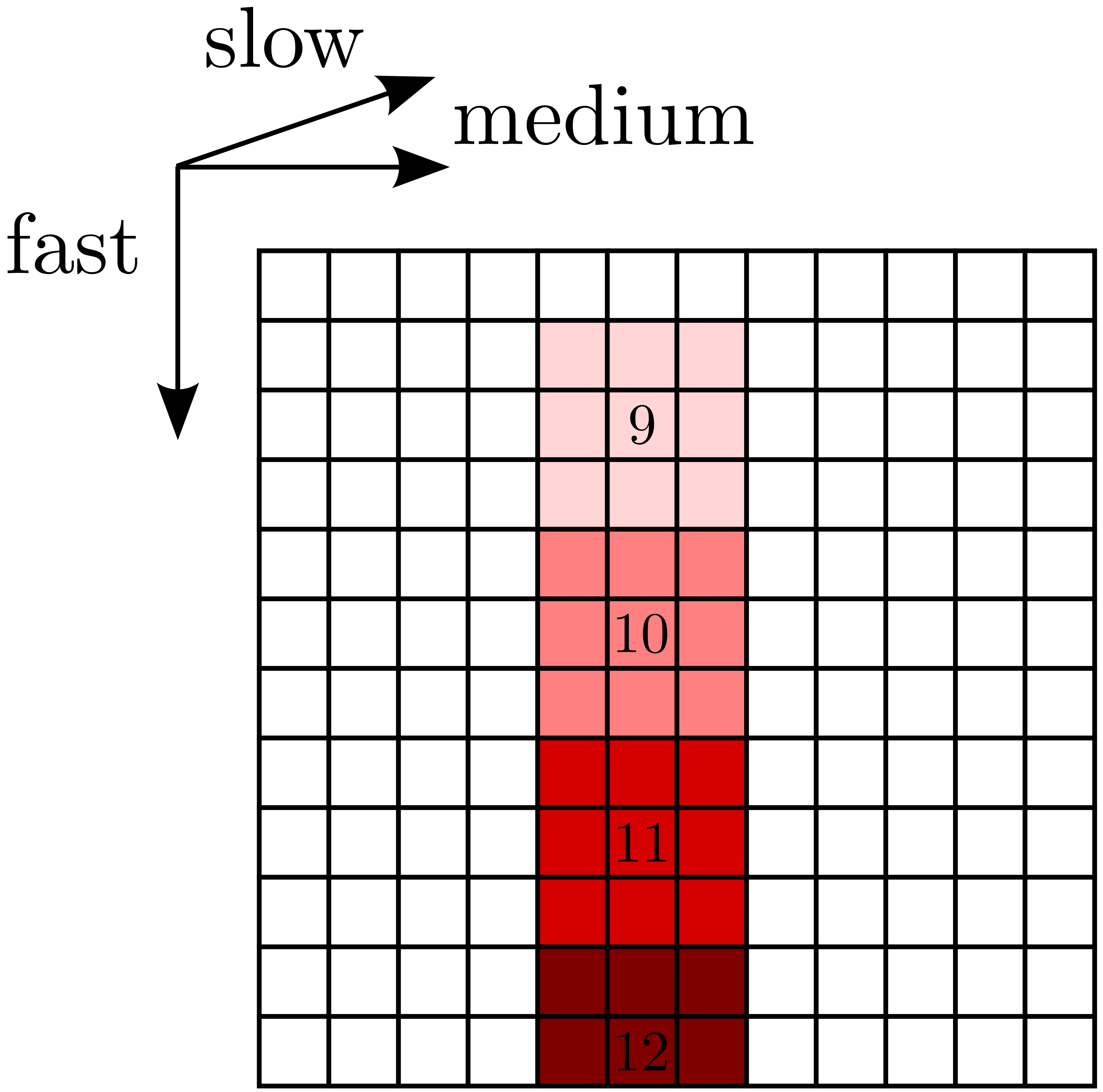
Pipelining: Overcoming Latency of Computation



Pipelining: Overcoming Latency of Computation



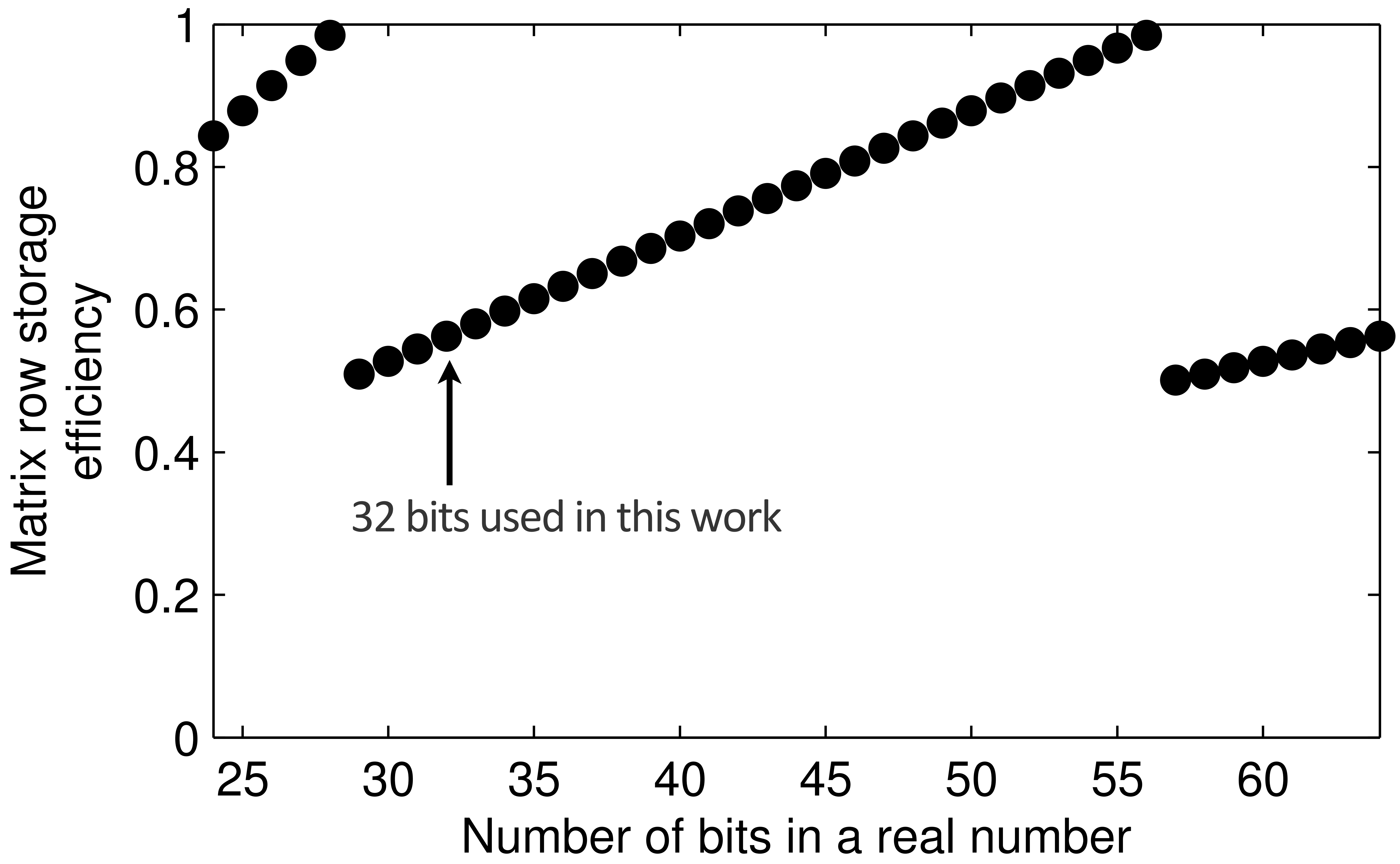
Pipelining: Overcoming Latency of Computation



Granular Memory Access: 384 bytes / burst

Number of bits in a real number	Number of bits in a complex number	Complex numbers per burst
24	48	64
32 (single precision)	64	48
48	96	32
64 (double precision)	128	24

Matrix Coefficient Representation



Results & Discussion

What is being compared?

Reference implementation:

- Solution Algorithm
  - CARP-CG
  - written in MATLAB
  - single-threaded
  - double precision
  - running on Intel Xeon E5-2670
- Computational Kernel
  - CARP sweeps
  - written in C
  - 32 threads
  - double precision
  - running on Intel Xeon E5-2670

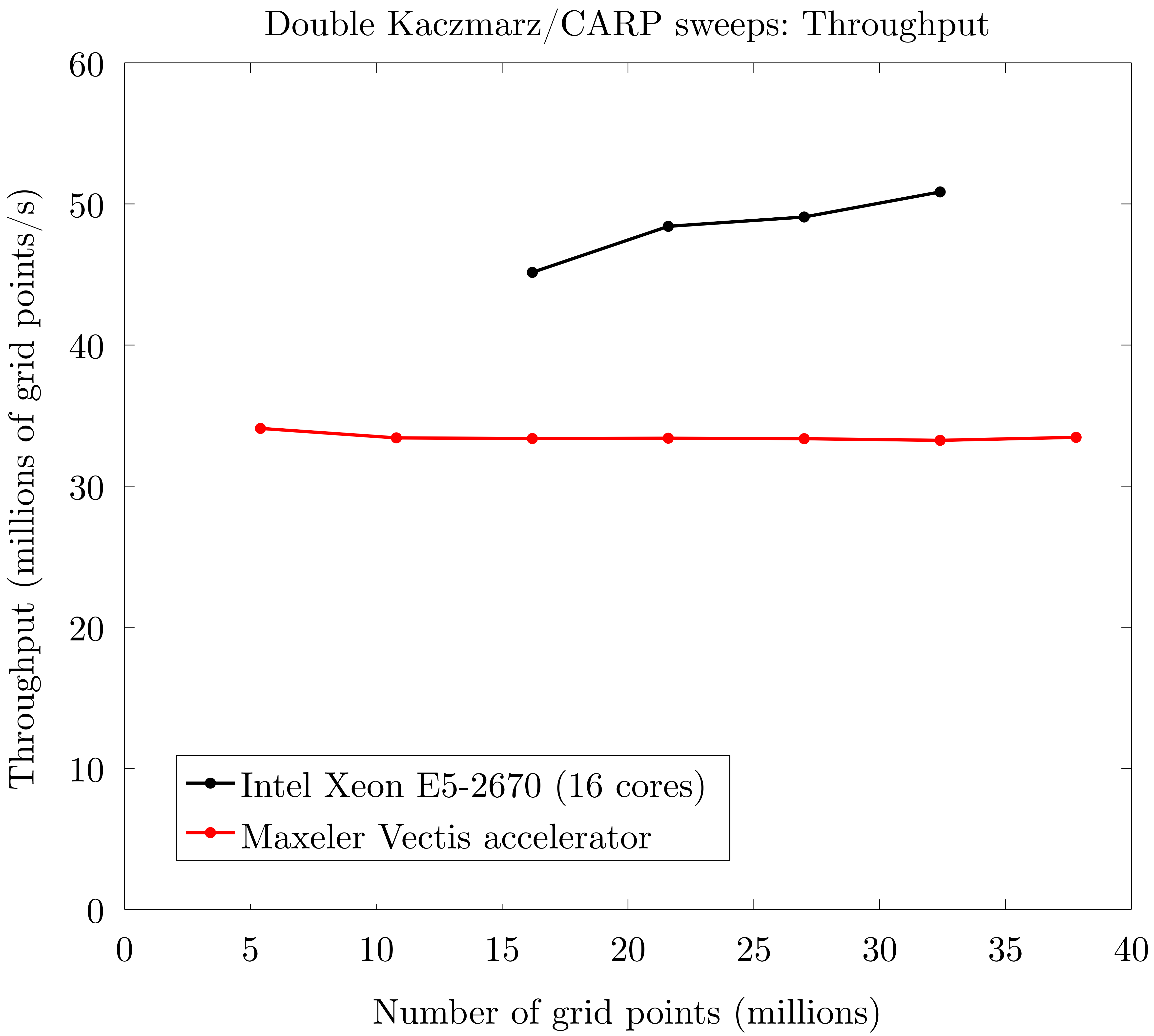
Accelerator implementation:

- Solution Algorithm
  - CGMN
  - written in MATLAB
  - single-threaded
  - single precision
  - running on Intel Xeon E5-2670
- Computational Kernel
  - Kaczmarz sweeps
  - single precision
  - running on Maxeler Vectis accelerator at 100 MHz
  - Memory clock at 303 MHz

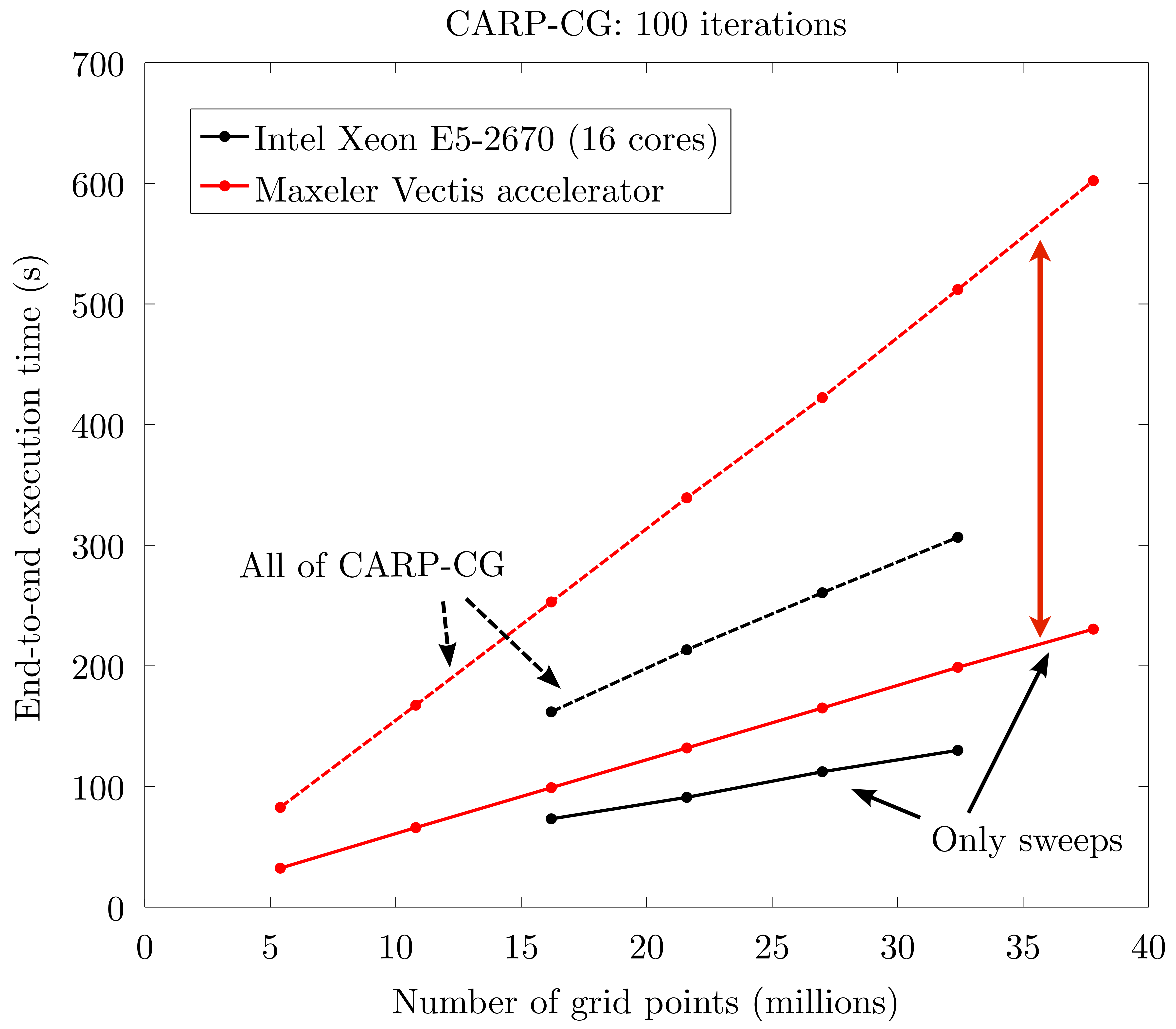
Experimental Set-up

- Solve part of the SEG/EAGE Overthrust System: 432 x 500 x Z.
- Point source.
- Zero initial guess.
- Run 100 CARP-CG/CGMN iterations.

Throughput of Kaczmarz/CARP sweeps



CARP-CG: End-to-end execution time

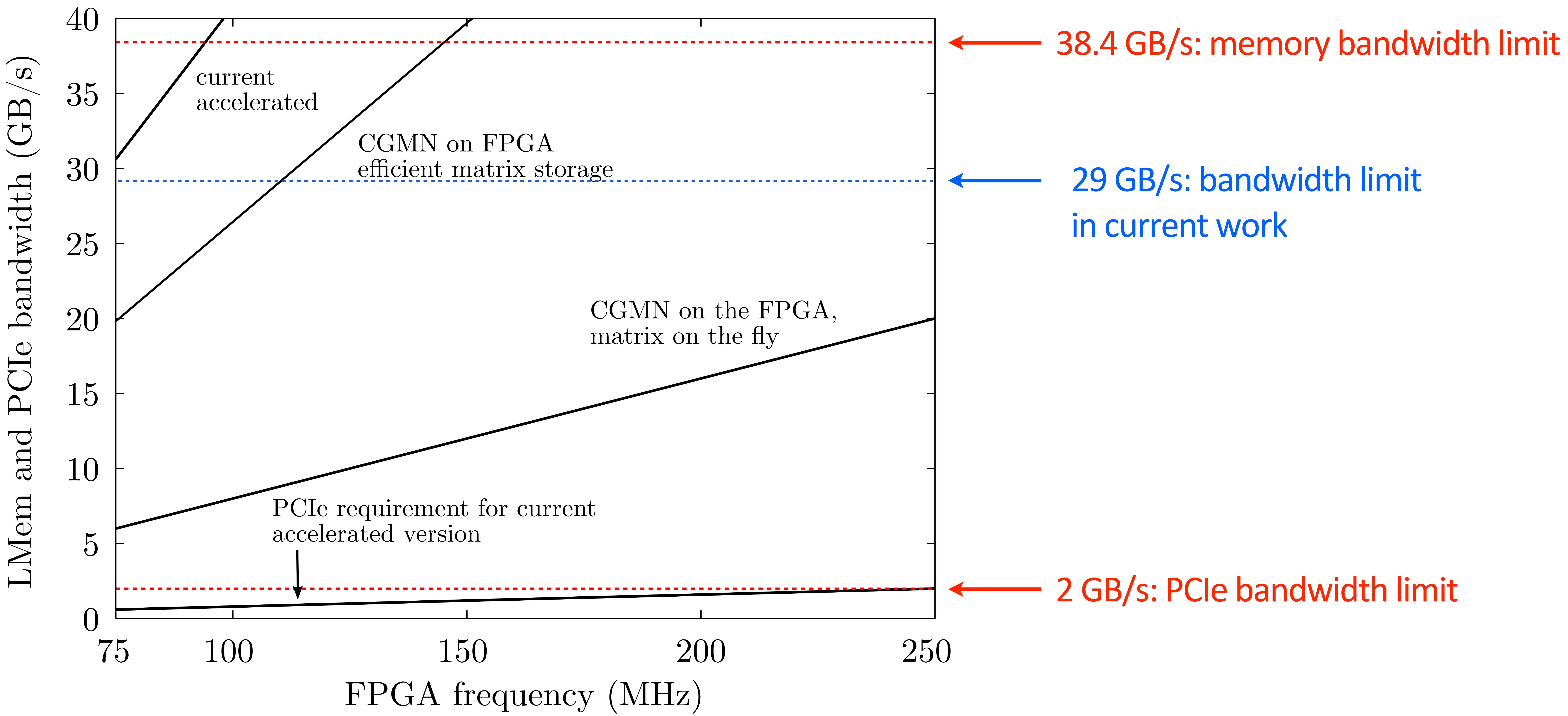


Lots of overhead!

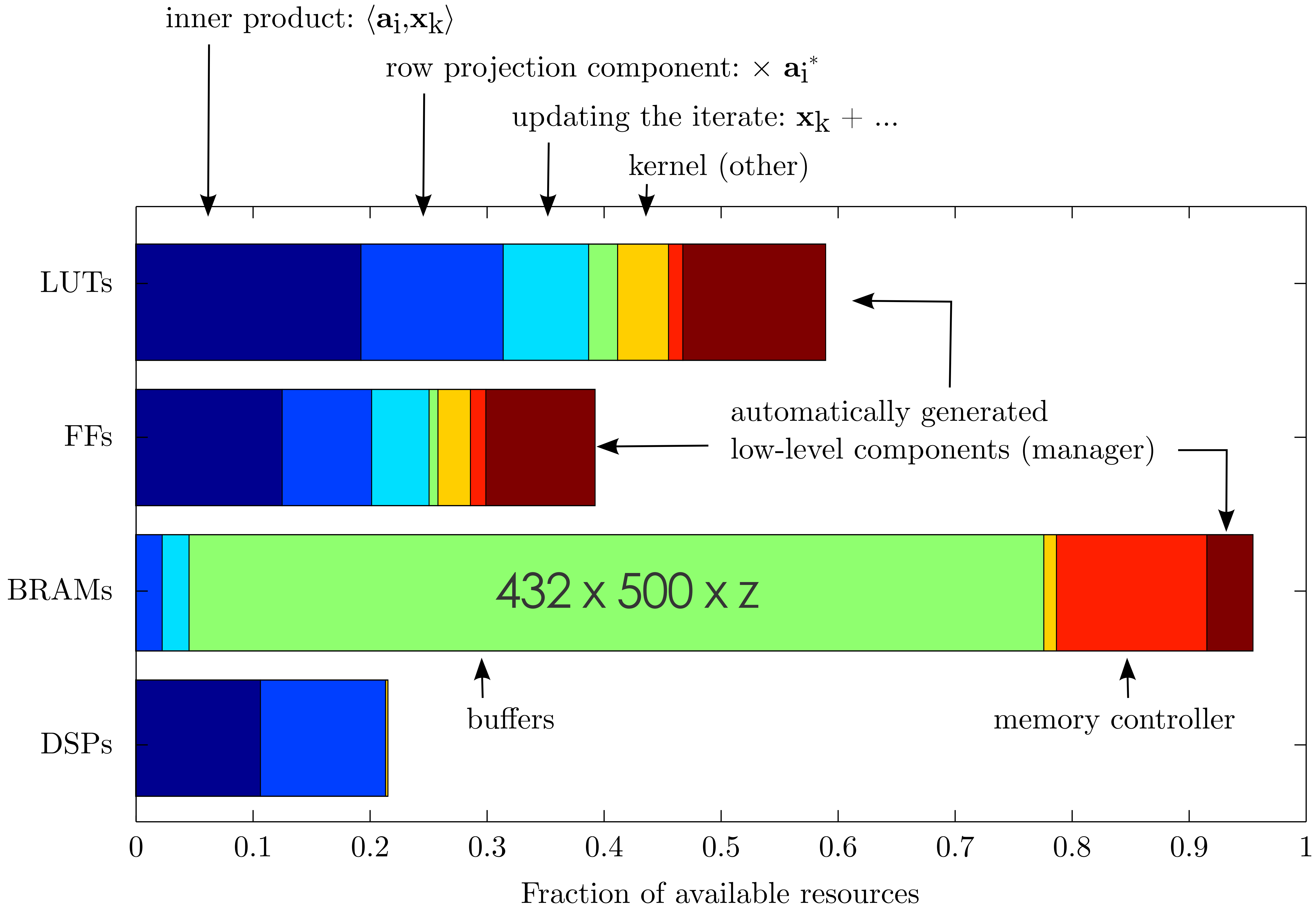
Kaczmarz sweeps are 39% of run time.

Future Solution: Port all of CGMN to accelerator.

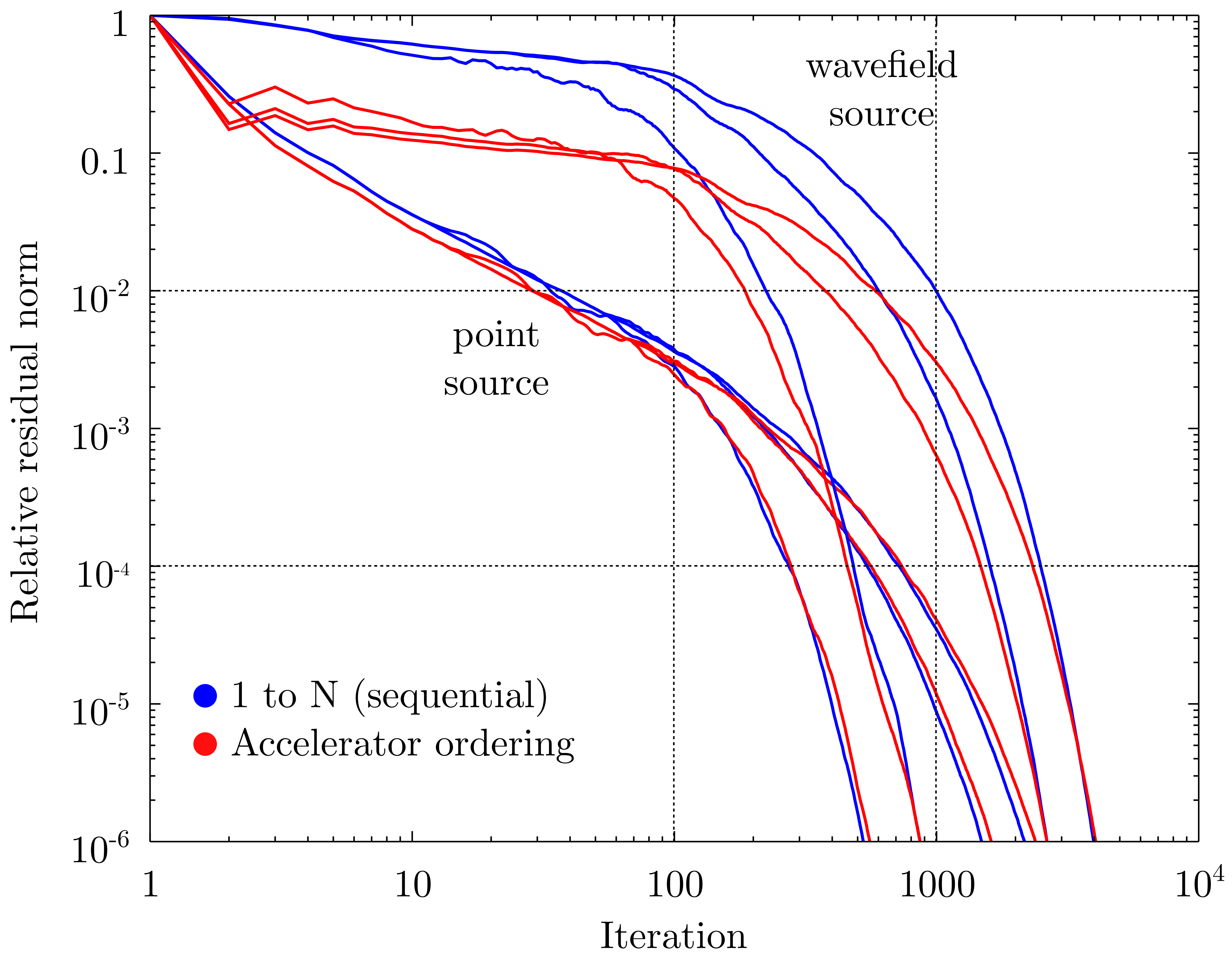
Avoiding Future Communication Bottlenecks



FPGA Resource Usage: Room for parallelism?



Effect of matrix row ordering on CGMN convergence



Conclusion

Have **implemented** frequency-domain wave simulation using reconfigurable hardware.

More work needed to realize full potential of accelerator system.

SLIM

Acknowledgements

Thank you to:

- Felix Herrmann, Henryk Modzelewski, Diego Oriato, Simon Tilbury, Tristan van Leeuwen, Eddie Hung, Rafael Lago, Maxeler Technologies, and everyone in the SLIM group!

SINBAD

NSERC CRSNG

This work was financially supported in part by the Natural Sciences and Engineering Research Council of Canada Discovery Grant (RGPIN 261641-06) and the Collaborative Research and Development Grant DNOISE II (CDRP J 375142-08). This research was carried out as part of the SINBAD II project with support from the following organizations: BG Group, BGP, BP, Chevron, ConocoPhillips, CGG, ION GXT, Petrobras, PGS, Statoil, Total SA, WesternGeco, Woodside.

References

Å. Björck and T. Elfving. Accelerated projection methods for computing pseudoinverse solutions of systems of linear equations. BIT Numerical Mathematics, 19(2):145–163, 1979. ISSN 0006-3835. doi: 10.1007/BF01930845. URL http://dx.doi.org/10.1007/BF01930845.

D. Gordon and R. Gordon. Component-averaged row projections: A robust, block-parallel scheme for sparse linear systems. SIAM Journal on Scientific Computing, 27(3):1092–1117, 2005. doi: 10.1137/040609458. URL http://epubs.siam.org/doi/abs/10.1137/040609458.

D. Gordon and R. Gordon. CARP-CG: A robust and efficient parallel solver for linear systems, applied to strongly convection dominated PDEs. Parallel Computing, 36(9): 495–515, 2010. ISSN 0167-8191. doi: 10.1016/j.parco.2010.05.004. URL http://www.sciencedirect.com/science/article/pii/S0167819110000827.

F. Grüll, M. Kunz, M. Hausmann, and U. Kebschull. An implementation of 3D electron tomography on FPGAs. In Reconfigurable Computing and FPGAs (ReConFig), 2012 International Conference on, pages 1–5, 2012. doi: 10.1109/ReConFig.2012.6416732.

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

S. Kaczmarz. Approximate solution of systems of linear equations. International Journal of Control, 57(6):1269–1271, 1993. doi: 10.1080/00207179308934446. (translation)

T. van Leeuwen, D. Gordon, R. Gordon, and F. J. Herrmann. Preconditioning the Helmholtz equation via row-projections. In EAGE technical program. EAGE, 2012. URL https://www.slim.eos.ubc.ca/Publications/Public/Conferences/EAGE/2012/vanleeuwen2012EAGEcarpcg/vanleeuwen2012EAGEcarpcg.pdf.

H. Meuer, E. Strohmaier, J. Dongarra, and H. Simon. Top 500 supercomputer sites, November 2013. URL https://www.top500.org.

S. Operto, J. Virieux, P. Amestoy, J.-Y. L'Excellent, L. Giraud, and H. B. H. Ali. 3D finite-difference frequency-domain modeling of visco-acoustic wave propagation using a massively parallel direct solver: A feasibility study. Geophysics, 72(5):SM195–SM211, 2007. doi: 10.1190/1.2759835. URL http://geophysics.geoscienceworld.org/content/72/5/SM195.abstract.

O. Pell, J. Bower, R. Dimond, O. Mencer, and M. J. Flynn. Finite-difference wave propagation modeling on special-purpose dataflow machines. Parallel and Distributed Systems, IEEE Transactions on, 24(5):906–915, 2013. ISSN 1045-9219. doi: 10.1109/TPDS.2012.198.