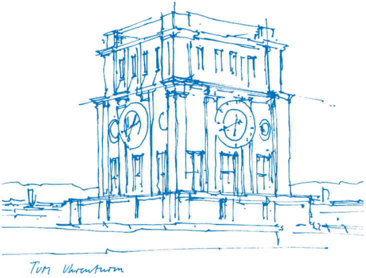
Extreme-scale Multi-physics Simulation of the 2004 Sumatra Earthquake

Intel MIC Programming Workshop

Michael Bader (and many others!)
Technical University of Munich

LRZ, 28 June 2017



Co-Authors – Current SeisSol Group

LMU Munich – Geophysics:


Alice-Agnes Gabriel


Elizabeth Madden


Stephanie Wollherr


Thomas Ulrich

Technical University of Munich – HPC:


Sebastian Rettenberger


Carsten Uphoff

Further/former members:

Alexander Breuer (TUM → San Diego)
Alexander Heinecke (Intel)
Christian Pelties (LMU → MunichRe)
Leonhard Rannabauer (TUM)

Part I

Dynamic Rupture and Earthquake Simulation with SeisSol

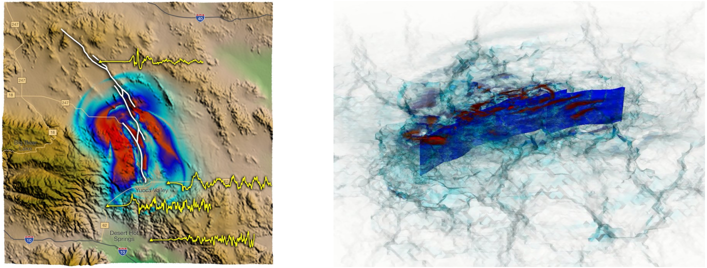
`http://www.seissol.org/`

Dumbser, Käser et al. [9]
An arbitrary high-order discontinuous Galerkin method . . .

Pelties, Gabriel et al. [11]
Verification of an ADER-DG method for complex dynamic rupture problems

Heinecke, Breuer, Rettenberger, Gabriel, Pelties et al. [4]:
Petascale High Order Dynamic Rupture Earthquake Simulations on Heterogeneous Supercomputers (Gordon Bell Prize Finalist 2014)

Dynamic Rupture and Earthquake Simulation

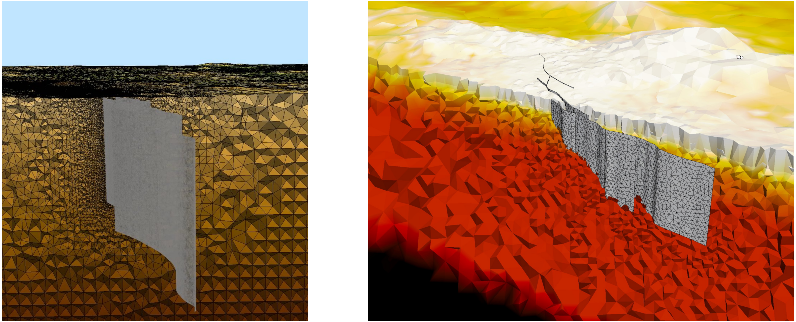


Landers fault system: simulated ground motion and seismic waves [4]

SeisSol – ADER-DG for seismic simulations:

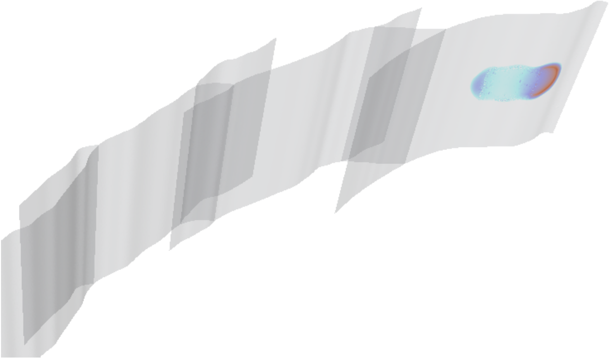
- adaptive tetrahedral meshes
  → complex geometries, heterogeneous media, multiphysics
- complicated fault systems with multiple branches
  → non-linear multiphysics dynamic rupture simulation
- ADER-DG: high-order discretisation in space and time

Example: 1992 Landers M7.2 Earthquake



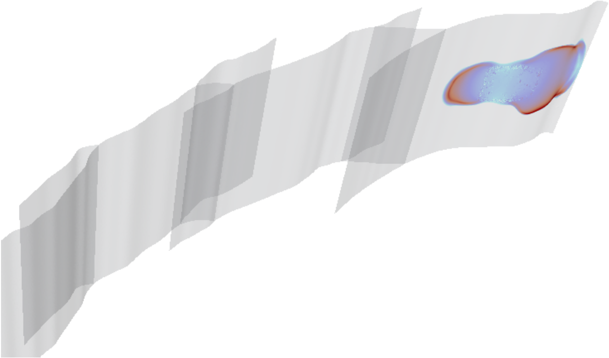
- multiphysics simulation of dynamic rupture and resulting ground motion of a M7.2 earthquake
- fault inferred from measured data, regional topography from satellite data, physically consistent stress and friction parameters
- static mesh refinement at fault and near surface

Multiphysics Dynamic Rupture Simulation



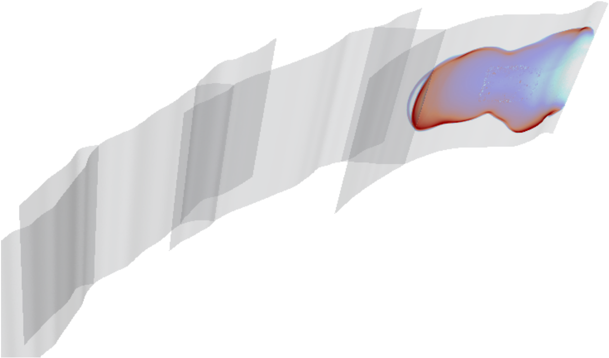
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



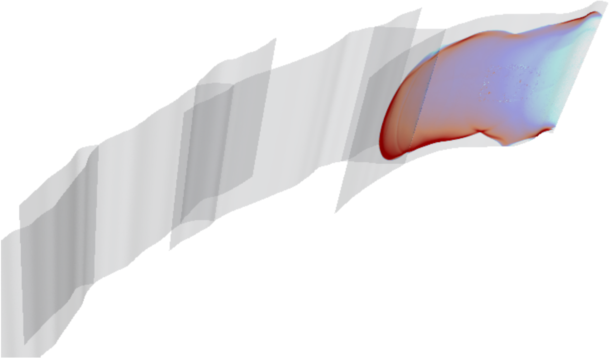
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



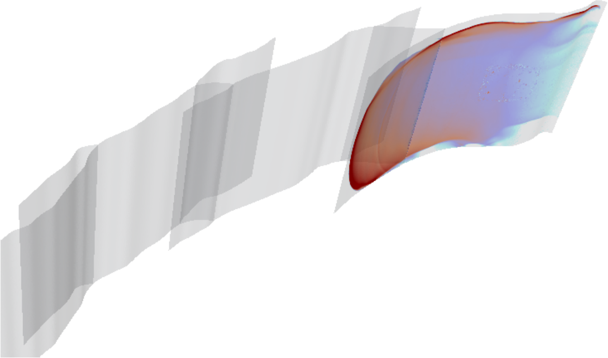
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



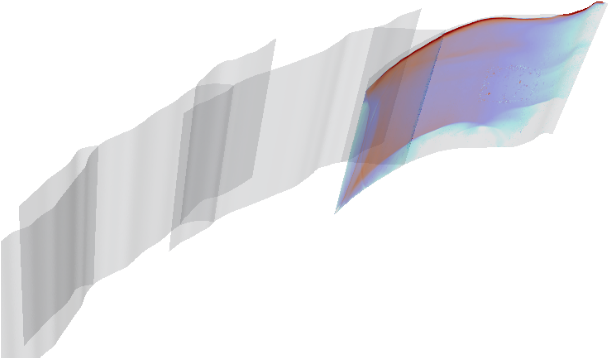
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



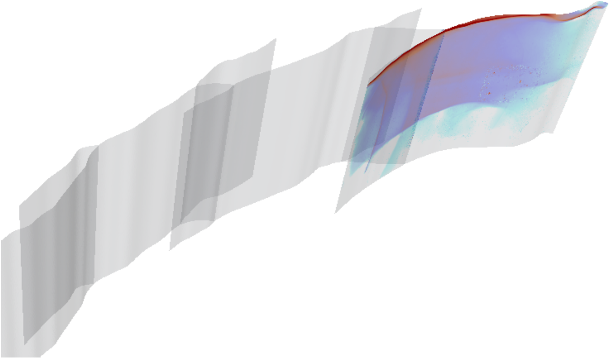
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



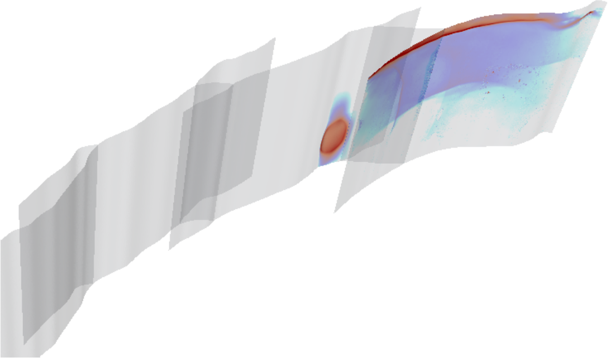
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



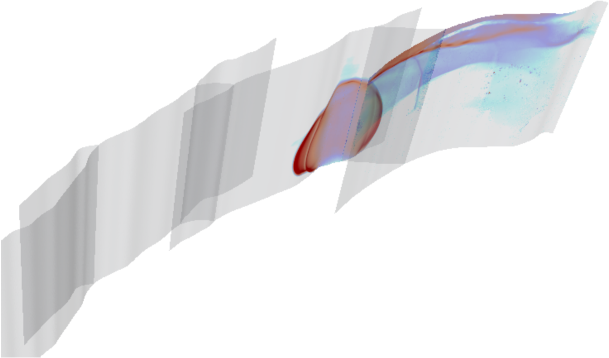
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



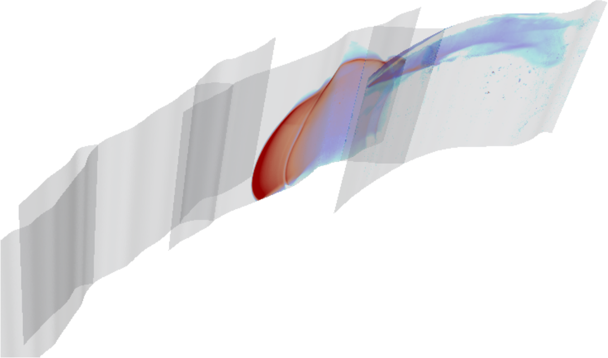
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



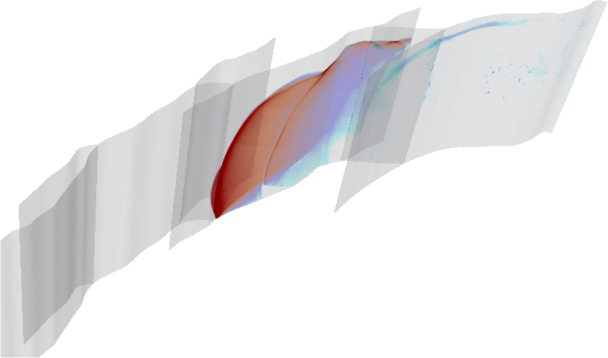
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



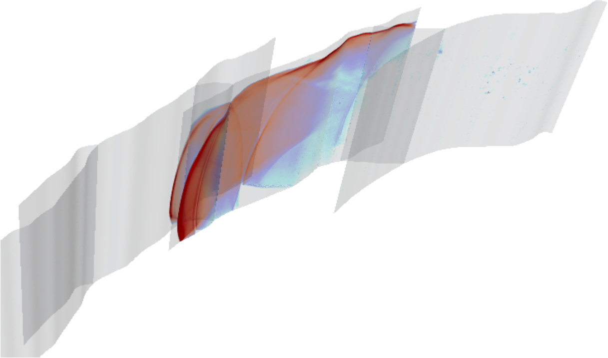
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



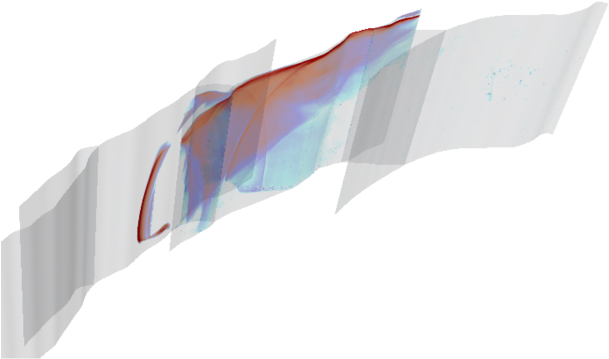
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



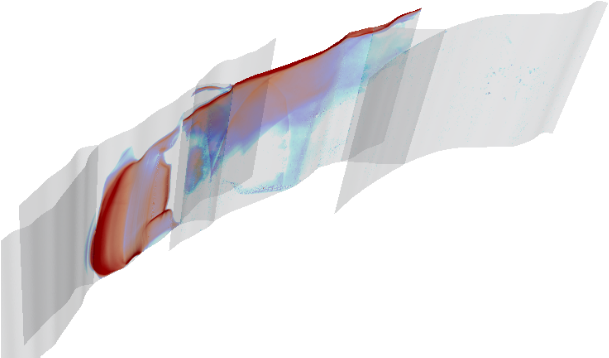
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



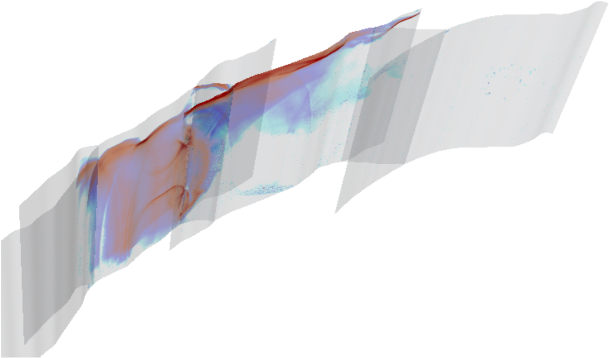
- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Multiphysics Dynamic Rupture Simulation



- spontaneous rupture, non-linear interaction with wave-field
- featuring rupture jumps, fault branching, etc.
- tackles fundamental questions on earthquake dynamics
- realistic rupture source for seismic hazard assessment

Part II

SeisSol as a Compute-Bound Code: Code Generation for Matrix Kernels

Breuer, Heinecke, Rannabauer, Bader [2]: High-Order ADER-DG Minimizes Energy- and Time-to-Solution of SeisSol (ISC'15)

Uphoff, Bader [6]: Generating high performance matrix kernels for earthquake simulations with viscoelastic attenuation (HPCS 2016)

Seismic Wave Propagation with SeisSol

Elastic Wave Equations: (velocity-stress formulation)

$$q_t + Aq_x + Bq_y + Cq_z = 0$$

with $q = (\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{13}, u, v, w)^T$

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -\lambda - 2\mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu \\ -\rho^{-1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\rho^{-1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\rho^{-1} & 0 & 0 & 0 \end{pmatrix} \quad B = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\lambda & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\lambda - 2\mu & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\lambda & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\rho^{-1} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\rho^{-1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\rho^{-1} & 0 & 0 & 0 & 0 \end{pmatrix}$$

- high order discontinuous Galerkin discretisation
- **ADER-DG**: high approximation order in space and time:
- additional features: local time stepping, high accuracy of earthquake faulting (full frictional sliding)

→ Dumbser, Käser et al., e.g. [8]

Discontinous Galerkin Discretisation in SeisSol

Weak Form of the elastic wave equations:

$$\int_{T_k} q_t \phi_m d\vec{x} + \int_{T_k} (Aq_x + Bq_y + Cq_z)\phi_m d\vec{x} = 0$$

Apply chain rule and divergence theorem:

$$\int_{T_k} q_t \phi_m d\vec{x} = \int_{T_k} Aq(\phi_m)_x + Bq(\phi_m)_y + Cq(\phi_m)_z \; d\vec{x} - \int_{\partial T_k} F\phi_m d\vec{s}$$

Further choices:

- modal basis ϕ_m; ϕ_m orthogonal to obtain diagonal mass matrix
- hierachical (w.r.t polynomial degree) basis ϕ_m,
  leads to staircase pattern in stiffness matrices
- exact Riemann solver for linear flux F

SeisSol in a Nutshell – ADER-DG

Update scheme

$$\begin{aligned}
Q_k^{n+1} = Q_k - \frac{|S_k|}{|J_k|} M^{-1} \Bigg(& \sum_{i=1}^{4} F^{-,i} I(t^n, t^{n+1}, Q_k^n) N_{k,i} A_k^{+} N_{k,i}^{-1} \\
& + \sum_{i=1}^{4} F^{+,i,j,h} I(t^n, t^{n+1}, Q_{k(i)}^n) N_{k,i} A_{k(i)}^{-} N_{k,i}^{-1} \Bigg) \\
& + M^{-1} K^{\xi} I(t^n, t^{n+1}, Q_k^n) A_k^{*} \\
& + M^{-1} K^{\eta} I(t^n, t^{n+1}, Q_k^n) B_k^{*} \\
& + M^{-1} K^{\zeta} I(t^n, t^{n+1}, Q_k^n) C_k^{*}
\end{aligned}$$

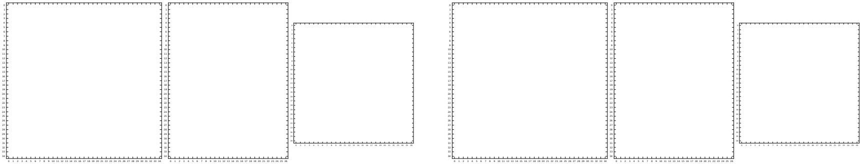
Cauchy Kovalewski

$$I(t^n, t^{n+1}, Q_k^n) = \sum_{j=0}^{J} \frac{(t^{n+1} - t^n)^{j+1}}{(j+1)!} \frac{\partial^j}{\partial t^j} Q_k(t^n)$$

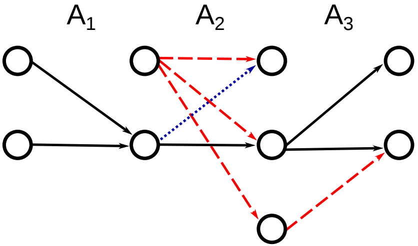
$$(Q_k)_t = -M^{-1} \left((K^{\xi})^T Q_k A_k^{*} + (K^{\eta})^T Q_k B_k^{*} + (K^{\zeta})^T Q_k C_k^{*} \right)$$

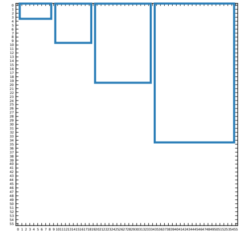
Sparse, Dense → Block-Sparse

Consider equivalent sparsity patterns: (Uphoff, [6])



Graph representation and block-sparse memory layouts





Code Generator for Matrix Chain Products

Programming Interface:

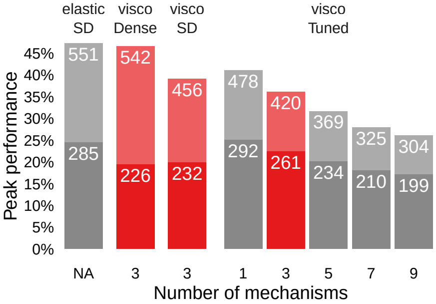
```
db = Tools.parseMatrixFile('matrices.xml')
Tools.memoryLayoutFromFile('layout.xml', db)
arch = Arch.getArchitectureByIdentifier('dhsw')
volume = db['kXiDivM'] * db['timeIntegrated'] * db['AstarT']
       + db['timeIntegrated'] * db['ET']
kernels = [('volume', volume)]
Tools.generate(
  'path/to/output', db, kernels,
  'path/to/libxsmm_gemm_generator', arch
)
```

Code Generation:

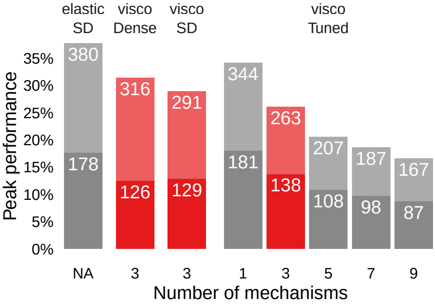
- auto-tuning to chose dense/sparse/blocked-sparse matrices
- automatically determine best order to evaluate matrix chain products
- efficient matrix multiplication backend: **libxsmm** library [10]

Floating-Point Performance (Haswell vs. KNC)

Single-node, 65,000 elements, 1000 timesteps, 6-th order (Uphoff, [6])



Dual-socket Xeon E5-2697 v3, 28 cores

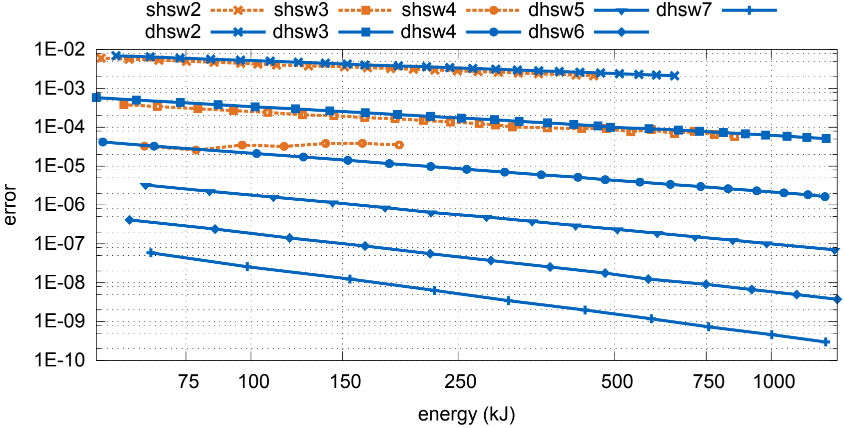


Xeon Phi 5110P, 60 cores

Non-zero flops increase by 13% due to matrix partitioning.

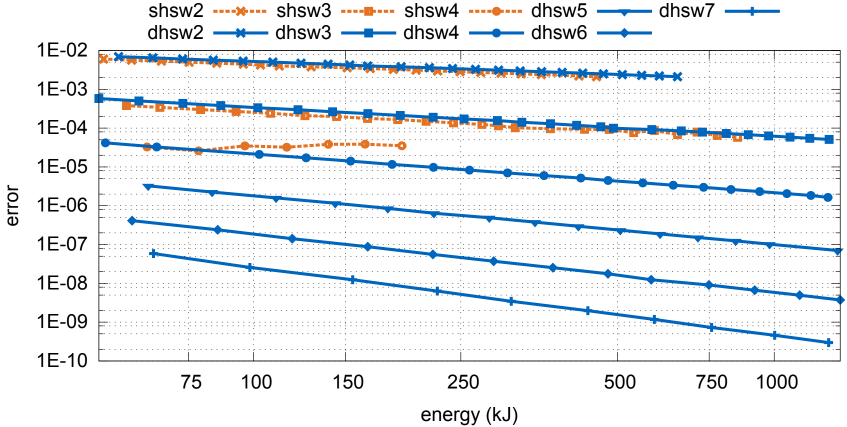
Non-zero flops increase by 7% due to matrix partitioning.

Benefit of High Order ADER-DG – Energy-Efficient



- mesasure maximum error vs. consumed energy
- for increasing discretisation order on regular meshes
- here: dual-socket “Haswell” server, 36 cores @1.9 GHz

Benefit of High Order ADER-DG – Energy-Efficient



- high order (“compute”) beats high resolution (“memory”)
- ≈ 35% gain in energy-to-solution for single precision, but only for low order

SeisSol – Recent Extensions

“Multiphysics” Simulations:

- viscoelastic attenuation; implementation based on new matrix-based code generator (C. Uphoff, [6])
- off-fault plasticity (current work by S. Wollherr)

Workflow and HPC:

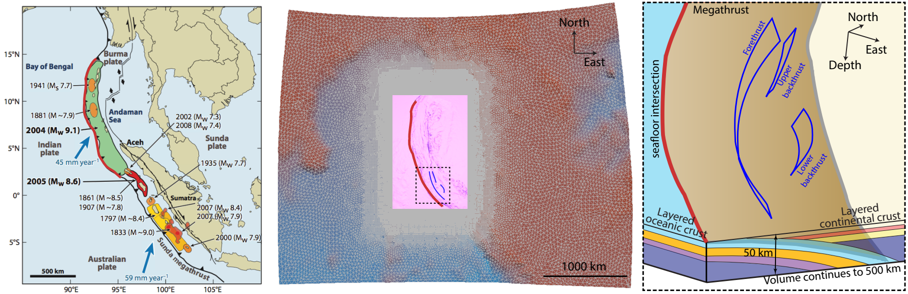
- asynchronous parallel IO using staging nodes or writer cores (S. Rettenberger, [13])
- input of 3D velocity models from data files via parallel library ASAGI (S. Rettenberger, [14])
- simplified CAD generation and close-to-automatic meshing using SimModeler and Simulation Modeling Suite by Simmetrix

Part III

Simulation of the 2004 Sumatra Megathrust Earthquake

Sebastian Rettenberger, Carsten Uphoff,
Alice Gabriel, Betsy Madden, Stephanie Wollherr, Thomas Ulrich:
Extreme Scale Multi-Physics Simulations of the Tsunamigenic 2004 Sumatra Megathrust Earthquake
SC17

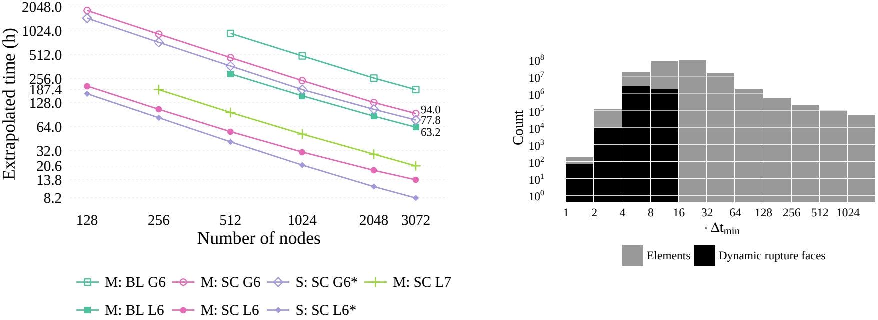
Sumatra Earthquake – Seismology Challenges



Domain, mesh and geometry of the Sumatra scenario

- multiscale: rupture extends of 1500 km, but happens on meter scale
- complex geometry: shallow angles in subduction zone; splay faults, topography, multiple material layers
- extremely long duration of earthquake: 500 s simulated time (over 3 Mio smallest time steps) → **local time stepping imperative**

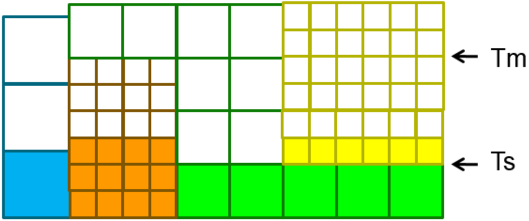
Sumatra Earthquake – HPC Challenges



Sumatra: histogram of LTS clusters and extrapolated runtimes

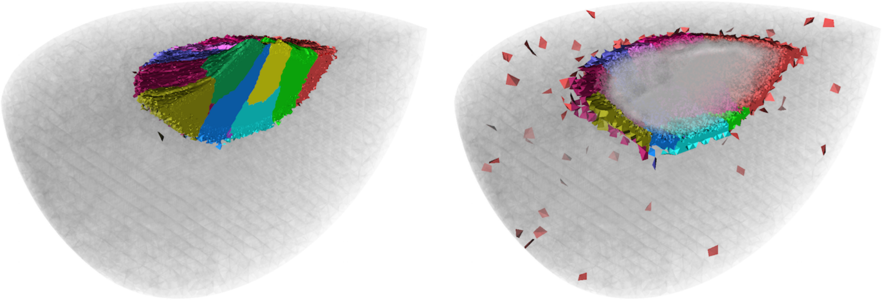
- target manycore CPUs (Knights Landing → Cori supercomputer)
  → available cache/local memory per core → new flux computation
  → dynamic rupture became bottleneck → matrix-based code generation
- dynamic rupture plus local time stepping with strong(!) scalability required

ADER Local Time Stepping



- ADER time stepping scheme allows straightforward extension to local time stepping
- implemented for SeisSol in 2007 (Dumbser et al. [9])
  → experienced severe scalability problems
  → better with (explicitly declared) clusters, but never really solved
- new approach by Alex Breuer [1]:
  settle for multi-rate time stepping and (arbitrary!) clusters
  ⇝ 4–5× speedup in time-to-solution for Landers scenario

Clusters for Local Time Stepping



- what we hoped for (but don't get): compact clusters of uniform time steps
- therefore: implemented bins of arbitrarily located grid cells
- bins defined from smallest time step Δt (a.k.a. global time step) $\rightarrow [\Delta t, 2\Delta t), [2\Delta t, 4\Delta t), [4\Delta t, 8\Delta t), \ldots$
- needed to re-organise data structures (ghost layers, element buffers, etc.) and data exchange (introduced communication threads)

Optimizing SeisSol for Xeon Phi (Knights Landing)

Step 1: Memory Optimization (Heinecke, Breuer et al., ISC 16 [5])

- profit from Knights Landing optimization of libxsmm library [10]
- examine impact of DRAM-only, CACHE and FLAT mode
- FLAT mode: careful placement of element-local matrices in MCDRAM:

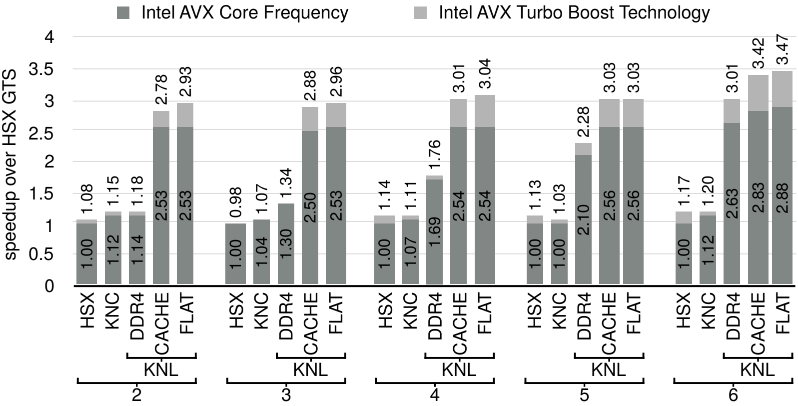
order	Q_k	$\mathcal{B}_k, \mathcal{D}_k$	$A_k^{\xi_c}, \hat{A}_k^{-,i}, \hat{A}_k^{+,i}$	$\hat{K}^{\xi_c}, \tilde{K}^{\xi_c}, \hat{F}^{-,i}, \hat{F}^{+,i,j,h}$
2	MCDRAM	MCDRAM	MCDRAM	MCDRAM
3	MCDRAM	MCDRAM	MCDRAM	MCDRAM
4	DDR4	MCDRAM	MCDRAM	MCDRAM
5	DDR4	MCDRAM	DDR4	MCDRAM
6	DDR4	MCDRAM	DDR4	MCDRAM

Step 2: Improved Flux Computation and Dynamic Rupture (C. Uphoff)

- exploit code generation based on matrix chain products
- fluxes: Riemann solvers expressed via matrix chain product $\rightarrow$ reformulate via smaller matrices (slightly fewer ops; much fewer cache)
- dynamic rupture: derive new scheme based on chain products

Performance Results on Knights Landing

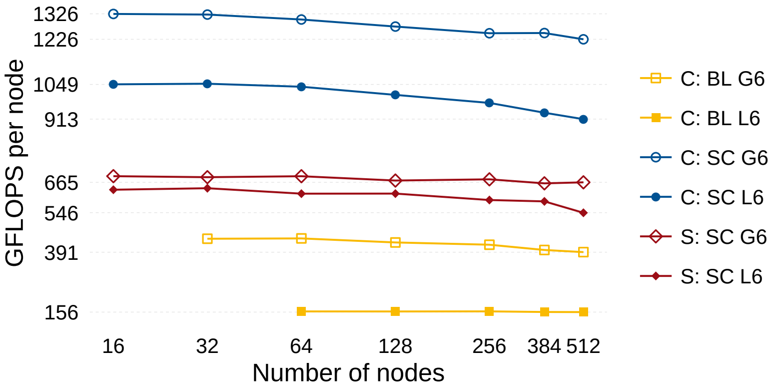
Phase 1: Heinecke et al., ISC 16 [5]



Landers scenario, 466,574 elements

Performance Results on Knights Landing

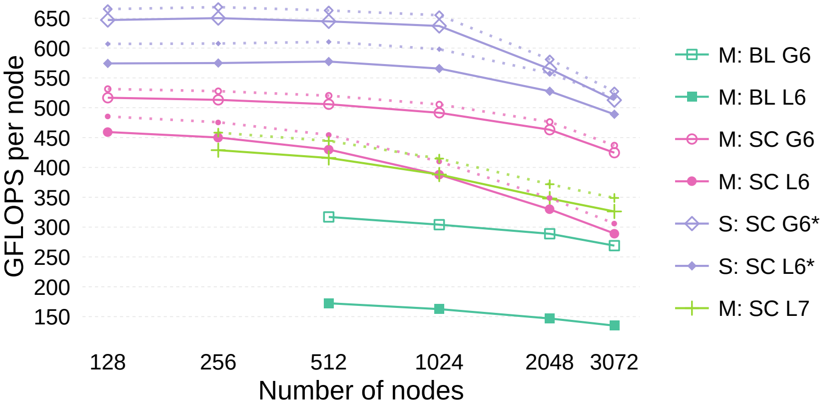
Phase 2: New Results on Cori (C. Uphoff et al.)



Sumatra scenario, mesh with 51 Mio elements

Performance Results on Haswell

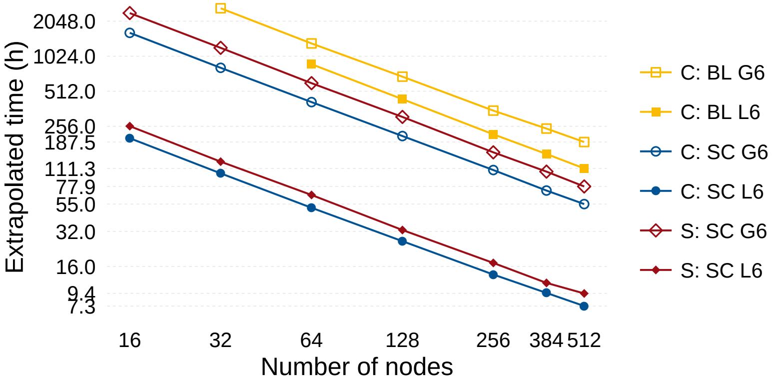
Phase 2: New Results on SuperMUC and Shaheen-II (C. Uphoff et al.)



Sumatra scenario, production mesh with 220 Mio elements

Performance Results on Haswell

Phase 2: New Results on SuperMUC and Shaheen-II (C. Uphoff et al.)



Sumatra scenario, production mesh with 220 Mio elements

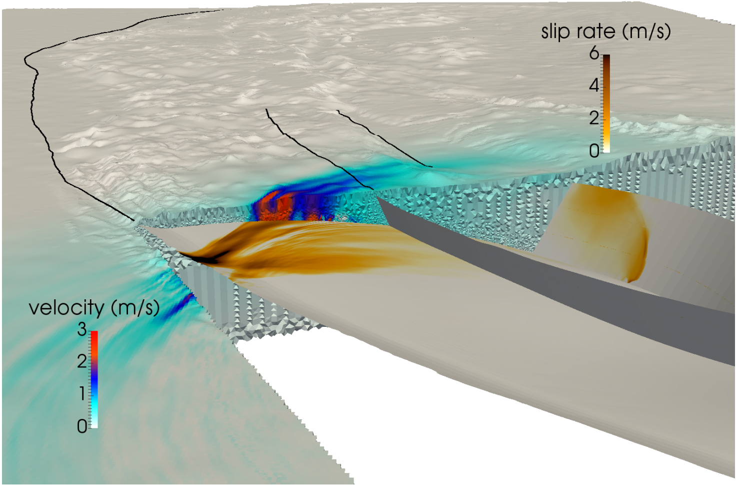
Sumatra 2004: 220 Mio Elements on SuperMUC

HPC Facts – 13.9 Hours Production Run:

- 221 million elements with order 6 accuracy
- 111 billion degrees of freedom
- 11 LTS clusters: “smallest” elements performed 3.3 Mio time steps
- 500 s simulated time
- 1500km fault size; 400 m geometrical resolution;
- 2.2 Hz frequency content of the seismic wave field
- 0.94 PFLOPS sustained performance (86,016 Haswell cores 2.2 GHz)
- 13 TB checkpoint data, 2.8 TB for post-processing
  (asynchronous IO; costs entirely overlapped by computation)

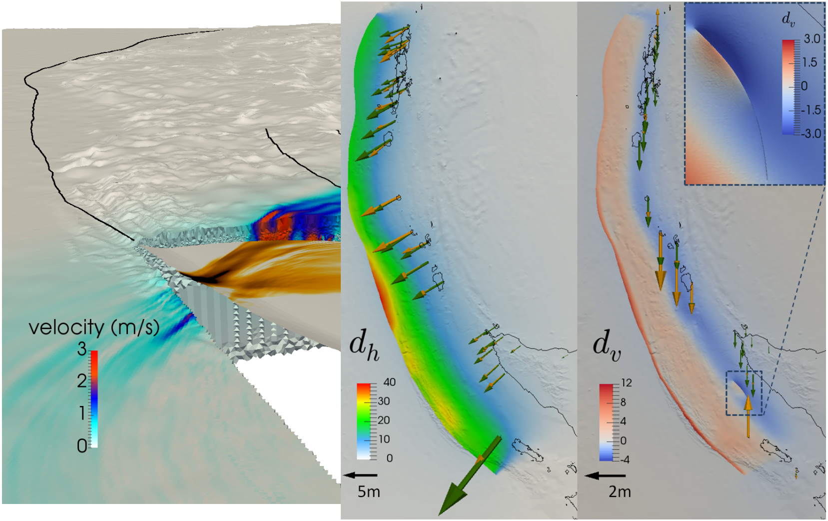
Sumatra 2004 – Results

Splay Fault Activation and Ocean Floor Displacements



Sumatra 2004 – Results

Splay Fault Activation and Ocean Floor Displacements



Conclusions – Earthquake Simulation with SeisSol

Compute-Bound Simulations at Petascale:

- high convergence order and high computational intensity of ADER-DG
  → compute-bound performance on current and imminent CPUs
- code generation based on matrix chain products to accelerate all element kernels
- careful tuning and parallelisation of the entire simulation pipeline (scalable mesh input, output and checkpointing)
- offload scheme scaled to 1.5 million cores (Tianhe-2, Stampede)
  → latest work tackled KNL and heterogeneous KNC platforms
  (Cori, Stampede, Salomon)

Multiphysics Earthquake Simulation:

- dynamic rupture coupled to seismic wave propagation
- recent/current work: visco-elastic attenuation, off-fault plasticity
- Sumatra 2004: first dynamic rupture simulation at this level of detail

Special thanks go to . . .

- the entire SeisSol team and all contributors, esp.:
    - Sebastian Rettenberger, Carsten Uphoff (TUM)
    - Alice Gabriel, Christian Pelties, Stephanie Wolherr (LMU)
    - Alex Breuer (SDSC, former: TUM)
    - Alex Heinecke (Intel, former: TUM)

- Leibniz Supercomputing Centre (esp. Nicolay Hammer): 30 Mio CPUh; 30-hour block operation on SuperMUC
- KAUST (esp. Martin Mai): access to Shaheen-II
- NERSC, Berkeley Lab (Rich Gerber, Jack Deslippe): access to Cori
- Intel: IPCC ExScaMIC –“Extreme Scaling on MIC-KNL”
- Volkswagen Foundation (project ASCETE)

Publications

[1] A. Breuer, A. Heinecke, M. Bader: Petascale local time stepping for the ADER-DG Finite Element method. Proc. IPDPS16.

[2] A. Breuer, A. Heinecke, L. Rannabauer, M. Bader: *High-Order ADER-DG Minimizes Energy- and Time-to-Solution of SeisSol*. In: High Performance Computing, Proceedings of ISC 15, LNCS 9137, p. 340–357, 2015.

[3] A. Breuer, A. Heinecke, S. Rettenberger, M. Bader, A.-A. Gabriel, C. Pelties: *Sustained Petascale Performance of Seismic Simulations with SeisSol on SuperMUC.* In: Supercomputing, LNCS 8488, p. 1–18. PRACE ISC Award 2014.

[4] A. Heinecke, A. Breuer, S. Rettenberger, M. Bader, A.-A. Gabriel, C. Pelties, A. Bode, W. Barth, X.-K. Liao, K. Vaidyanathan, M. Smelyanskiy, P. Dubey: *Petascale High Order Dynamic Rupture Earthquake Simulations on Heterogeneous Supercomputers*. Gordon Bell Prize Finalist 2014.

[5] A. Heinecke, A. Breuer, M. Bader, P. Dubey: *High Order Seismic Simulations on the Intel Xeon Phi Processor (Knights Landing)*. ISC High Performance, 2016.

[6] C. Uphoff, M. Bader: *Generating high performance matrix kernels for earthquake simulations with viscoelastic attenuation*. The 2016 International Conference on High Performance Computing & Simulation (HPCS 2016), p. 908–916. IEEE, 2016.

[7] C. Uphoff, S. Rettenberger, M. Bader, E. H. Madden, T. Ulrich, S. Wollherr and A.-A. Gabriel: *Extreme Scale Multi-Physics Simulations of the Tsunamigenic 2004 Sumatra Megathrust Earthquake.* SC '17.

Publications and References

[8] M. Dumbser, M. Käser: *An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes – II. The three-dimensional isotropic case.* Geophys. J. Int. 167(1), 2006.

[9] M. Dumbser, M. Käser, E. Toro: *An Arbitrary High Order Discontinuous Galerkin Method for Elastic Waves on Unstructured Meshes – V. Local Time Stepping and p-Adaptivity*, Geophys. J. Int. 171(2), 2007

[10] A. Heinecke, G. Henry, M. Hutchinson, H. Pabst: *LIBXSMM: Accelerating Small Matrix Multiplications by Runtime Code Generation*, SC16.

[11] C. Pelties, A.-A. Gabriel, J.-P. Ampuero: *Verification of an ADER-DG method for complex dynamic rupture problems*, Geoscientific Model Development, 7(3), p. 847–866.

[12] C. Pelties, J. de la Puente, J.-P. Ampuero, G. B. Brietzke, M. Käser: *Three-dimensional dynamic rupture simulation with a high-order discontinuous Galerkin method on unstructured tetrahedral meshes.* J. Geophys. Res.: Solid Earth, 117(B2), 2012.

[13] S. Rettenberger, M. Bader: *Optimizing Large Scale I/O for Petascale Seismic Simulations on Unstructured Meshes* 2015 IEEE International Conference on Cluster Computing (CLUSTER), p. 314–317. IEEE Xplore, 2015.

[14] S. Rettenberger, O. Meister, M. Bader, A.-A. Gabriel: *ASAGI – A Parallel Server for Adaptive Geoinformation.* Proceedings of the Exascale Applications and Software Conference 2016 (EASC '16), p. 2:1–2:9. ACM, 2016.