Massively parallel fluid-structure interaction simulation of blast and explosions impacting on realistic building structures with a block-structured AMR method

Ralf Deiterding & Stephen Wood

Computer Science and Mathematics Division
Oak Ridge National Laboratory
P.O. Box 2008 MS6367, Oak Ridge, TN 37831, USA
E-mail: deiterdingr@ornl.gov

SIAM Conference on Parallel Processing for Scientific Computing
February 16, 2012

Outline

This work was sponsored by the Office of Advanced Scientific Computing Research; U.S. Department of Energy (DOE) and was performed at the Oak Ridge National Laboratory, which is managed by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725.

The Virtual Test Facility

- Developed for first DOE ASC Center at the Caltech under Dan Meiron
- Overall idea: Use Cartesian embedded boundary approach based on level sets in combination with AMR to enable generic fluid-structure interaction coupling to numerous explicit solid mechanics solvers
- Targets strongly driven problems (shocks, blast, detonations)
- http://www.cacr.caltech.edu/asc
- Papers: [Deiterding, 2011, Deiterding et al., 2009, Deiterding et al., 2007, Deiterding et al., 2006], etc: http://www.csm.ornl.gov/~r2v
- AMROC V2.0 plus some solid mechanics solvers (SFC, beam solver)
- $\sim 430,000$ lines of code total in C++, C, Fortran-77, Fortran-90
- autoconf / automake environment with support for typical parallel high-performance system
- Used in here AMROC V2.1 (not released yet)
- New interface to DYNA3D - first prototype by Patrick Hung, Julian Cummings (CACR)

Outline

Introduction

Parallel SAMR
- Structured adaptive mesh refinement
- Complex geometry embedding
- Parallelization
- Performance data from AMROC

Fluid-structure interaction
- Coupling to a solid mechanics solver
- Verification and validation configurations
- Blast-driven deformation
- Detonation-driven deformations

Conclusions

Block-structured adaptive mesh refinement (SAMR)

For simplicity $\partial_t \mathbf{q}(\mathbf{x},t) + \nabla \cdot \mathbf{f}(\mathbf{q}(\mathbf{x},t)) = 0$

- Refined blocks overlay coarser ones
- Refinement in space *and time* by factor r_l
- Block (aka patch) based data structures

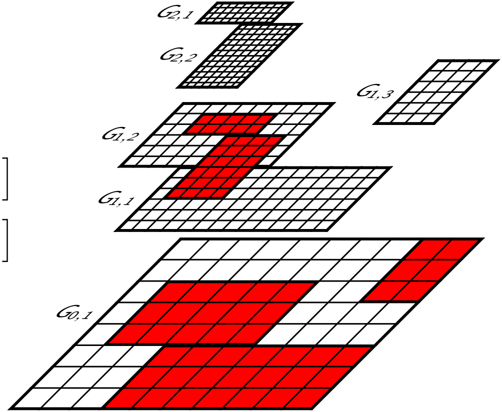
\+ Numerical scheme

$$\mathbf{Q}_{jk}^{n+1} = \mathbf{Q}_{jk}^{n} - \frac{\Delta t}{\Delta x_1}\left[\mathbf{F}^1_{j+\frac{1}{2},k} - \mathbf{F}^1_{j-\frac{1}{2},k}\right] - \frac{\Delta t}{\Delta x_2}\left[\mathbf{F}^2_{j,k+\frac{1}{2}} - \mathbf{F}^2_{j,k-\frac{1}{2}}\right]$$

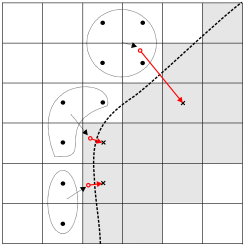
only for single patch necessary

\+ Efficient cache-reuse / vectorization possible

\- Cluster-algorithm necessary



Level-set method for boundary embedding



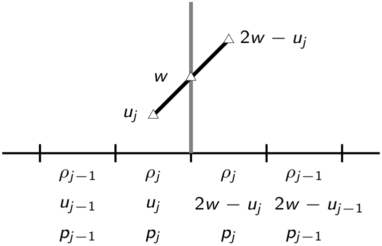
- Implicit boundary representation via distance function φ, normal $\mathbf{n} = \nabla\varphi/|\nabla\varphi|$
- Complex boundary moving with local velocity $\mathbf{w}$, treat interface as moving rigid wall
- Construction of values in embedded boundary cells by interpolation / extrapolation

Interpolate / constant value extrapolate values at

$$\tilde{\mathbf{x}} = \mathbf{x} + 2\varphi\mathbf{n}$$

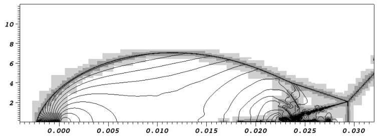
Velocity in ghost cells

$$\mathbf{u}' = (2\mathbf{w}\cdot\mathbf{n} - \mathbf{u}\cdot\mathbf{n})\mathbf{n} + (\mathbf{u}\cdot\mathbf{t})\mathbf{t}$$
$$= 2\left((\mathbf{w} - \mathbf{u})\cdot\mathbf{n}\right)\mathbf{n} + \mathbf{u}$$

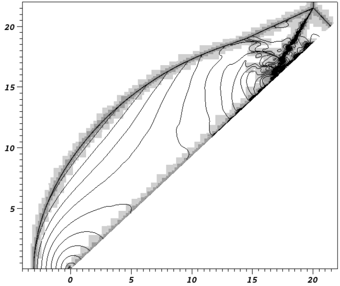


Verification: shock reflection

- Reflection of a Mach 2.38 shock in nitrogen at 43^o wedge
- 2nd order MUSCL scheme with Roe solver, 2nd order multidimensional wave propagation method

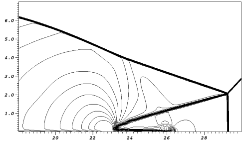


Cartesian base grid 360 × 160 cells on domain of $36\,\mathrm{mm} \times 16\,\mathrm{mm}$ with up to 3 refinement levels with $r_l = 2, 4, 4$ and $\Delta x_{1,2} = 3.125\mu m$, $38\,\mathrm{h}$ CPU

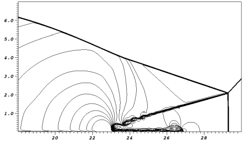


GFM base grid 390 × 330 cells on domain of $26\,\mathrm{mm} \times 22\,\mathrm{mm}$ with up to 3 refinement levels with $r_l = 2, 4, 4$ and $\Delta x_{e,1,2} = 2.849\mu m$, $200\,\mathrm{h}$ CPU

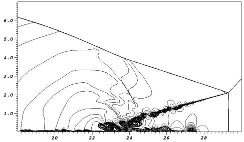
Shock reflection: SAMR solution for Euler equations



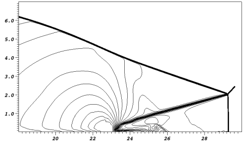
$\Delta x = 25\,\mathrm{mm}$



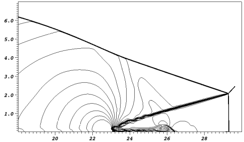
$\Delta x = 12.5\,\mathrm{mm}$



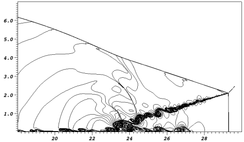
$\Delta x = 3.125\,\mathrm{mm}$



$\Delta x_e = 22.8\,\mathrm{mm}$

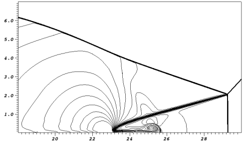


$\Delta x_e = 11.4\,\mathrm{mm}$

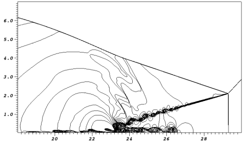


$\Delta x_e = 2.849\,\mathrm{mm}$

2nd order MUSCL scheme with Van Leer FVS, dimensional splitting



$\Delta x = 12.5\,\mathrm{mm}$



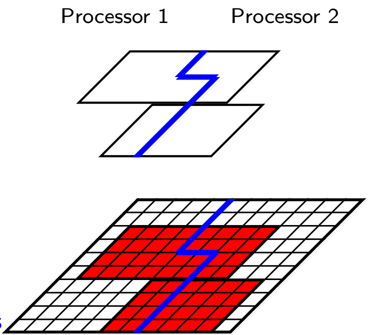
$\Delta x = 3.125\,\mathrm{mm}$

Parallelization strategies

- Data of all levels resides on same node
- Grid hierarchy defines unique "floor-plan"
- Workload estimation

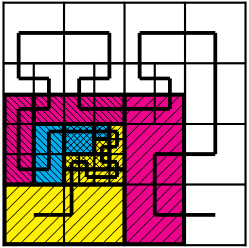
$$\mathcal{W}(\Omega) = \sum_{l=0}^{l_{\max}} \left[\mathcal{N}_l(G_l \cap \Omega) \prod_{\kappa=0}^{l} r_\kappa \right]$$

- Parallel operations
  - Synchronization of ghost cells
  - Redistribution of data blocks within regridding operation
  - Flux correction of coarse grid cells



- Clip grid lists with properly chosen quadratic bounding box before using $\cap$, $\setminus$
- All topological operations in `Recompose(l)` involving global lists can be reduced to local ones
- Present code still uses `MPI_allgather()` to communicate global lists to all nodes
- Global view useful to evaluate new local portion of hierarchy and for data redistribution

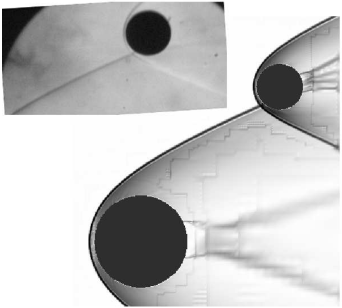
Parallelized construction of space-filling curve

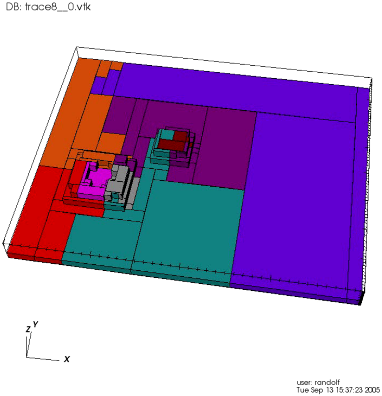


Computation of space filling curve

- Partition-Init
  1. Compute aggregated workload for new grid hierarchy and project result onto level 0
  2. Construct recursively SFC-units until work in each unit is homogeneous, GuCFactor defines minimal coarseness relative to level-0 grid
- Partition-Calc
  1. Compute entire workload and new work for each processor
  2. Go sequentially through SFC-ordered list of partitioning units and assign units to processors, refine partition if necessary and possible
- Ensure scalability of Partition-Init by creating SFC-units strictly local
- Currently still use of `MPI_allgather()` to create globally identical input for Partition-Calc (can be a bottleneck for weak scalability)

Partitioning example





- Cylinders of spheres in supersonic flow
- Predict force on secondary body
- Right: 200x160 base mesh, 3 Levels, factors 2,2,2, 8 CPUs

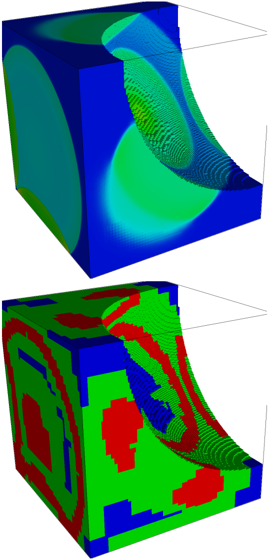
[Laurence et al., 2007]

First performance assessment

- Test run on 2.2 GHz AMD Opteron quad-core cluster connected with Infiniband
- Cartesian test configuration
- Spherical blast wave, Euler equations, 3rd order WENO scheme, 3-step Runge-Kutta update
- AMR base grid 64^3, $r_{1,2} = 2$, 89 time steps on coarsest level
- With embedded boundary method: 96 time steps on coarsest level
- Redistribute in parallel every 2nd base level step
- Uniform grid $256^3 = 16.8 \cdot 10^6$ cells

Level	Grids	Cells
0	115	262,144
1	373	1,589,808
2	2282	5,907,064

Grid and cells used on 16 CPUs



Cost of SAMR and ghost-fluid method

- Flux correction is negligible
- Clustering is negligible (already local approach). For the complexities of a scalable global clustering algorithm see [Gunney et al., 2007]
- Costs for GFM constant around $\sim 36\%$
- Main costs: `Regrid(l)` operation and ghost cell synchronization

CPUs	16	32	64
Time per step	32.44s	18.63s	11.87s
Uniform	59.65s	29.70s	15.15s
Integration	73.46%	64.69%	50.44%
Flux Correction	1.30%	1.49%	2.03%
Boundary Setting	13.72%	16.60%	20.44%
Regridding	10.43%	15.68%	24.25%
Clustering	0.34%	0.32%	0.26%
Output	0.29%	0.53%	0.92%
Misc.	0.46%	0.44%	0.47%

CPUs	16	32	64
Time per step	43.97s	25.24s	16.21s
Uniform	69.09s	35.94s	18.24s
Integration	59.09%	49.93%	40.20%
Flux Correction	0.82%	0.80%	1.14%
Boundary Setting	19.22%	25.58%	28.98%
Regridding	7.21%	9.15%	13.46%
Clustering	0.25%	0.23%	0.21%
GFM Find Cells	2.04%	1.73%	1.38%
GFM Interpolation	6.01%	10.39%	7.92%
GFM Overhead	0.54%	0.47%	0.37%
GFM Calculate	0.70%	0.60%	0.48%
Output	0.23%	0.52%	0.74%
Misc.	0.68%	0.62%	0.58%

AMROC scalability tests

Basic test configuration

- Spherical blast wave, Euler equations, 3D wave propagation method
- AMR base grid 32^3 with $r_{1,2} = 2, 4$. 5 time steps on coarsest level
- Uniform grid $256^3 = 16.8 \cdot 10^6$ cells, 19 time steps
- Flux correction deactivated
- No volume I/O operations
- Tests run IBM BG/P (mode VN)

Weak scalability test

- Reproduction of configuration each 64 CPUs
- On 1024 CPUs: $128 \times 64 \times 64$ base grid, $> 33,500$ Grids, $\sim 61 \cdot 10^6$ cells, uniform $1024 \times 512 \times 512 = 268 \cdot 10^6$ cells

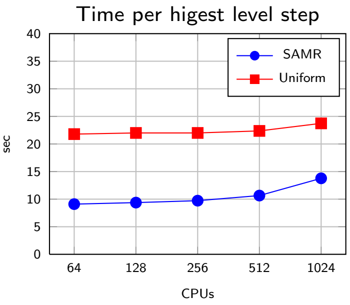
Level	Grids	Cells
0	606	32,768
1	575	135,312
2	910	3,639,040

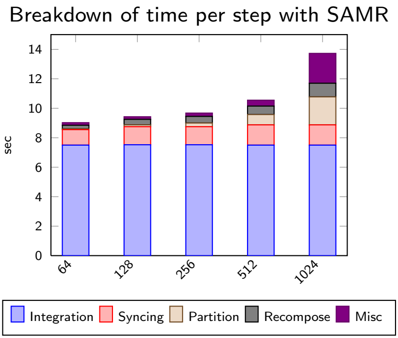
Strong scalability test

- $64 \times 32 \times 32$ base grid, uniform $512 \times 256 \times 256 = 33.6 \cdot 10^6$ cells

Level	Grids	Cells
0	1709	65,536
1	1735	271,048
2	2210	7,190,208

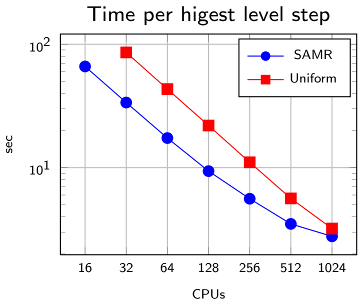
Weak scalability test

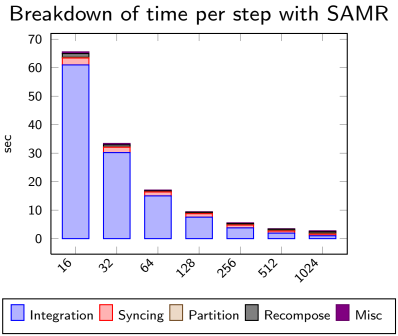




- Costs for Syncing basically constant
- Partitioning, Recompose, Misc (origin not clear) increase
- 1024 required usage of -DUAL option due to usage of global lists data structures in Partition-Calc and Recompose

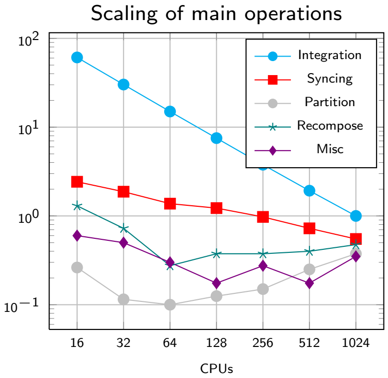
Strong scalability test

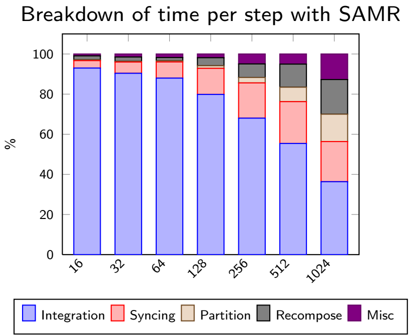




- Uniform code has basically linear scalability (explicit method)
- SAMR visibly looses efficiency for > 512 CPU, or $15,000$ finite volume cells per CPU

Strong scalability test - II





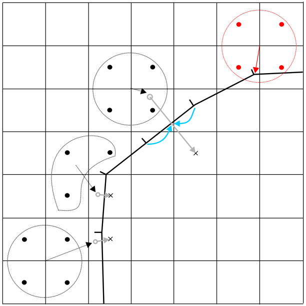
- Perfect scaling of Integration, reasonable scaling of Syncing
- Strong scalability of Partition needs to be addressed (eliminate global lists)

Outline

Construction of coupling data

- Moving boundary/interface is treated as a moving contact discontinuity and represented by level set [Fedkiw, 2002][Arienti et al., 2003]
- Efficient construction of level set from triangulated surface data with closest-point-transform (CPT) algorithm [Mauch, 2003]
- One-sided construction of mirrored ghost cell and new FEM nodal point values
- FEM ansatz-function interpolation to obtain intermediate surface values
- Explicit coupling possible if geometry and velocities are prescribed for the more compressible medium [Specht, 2000]

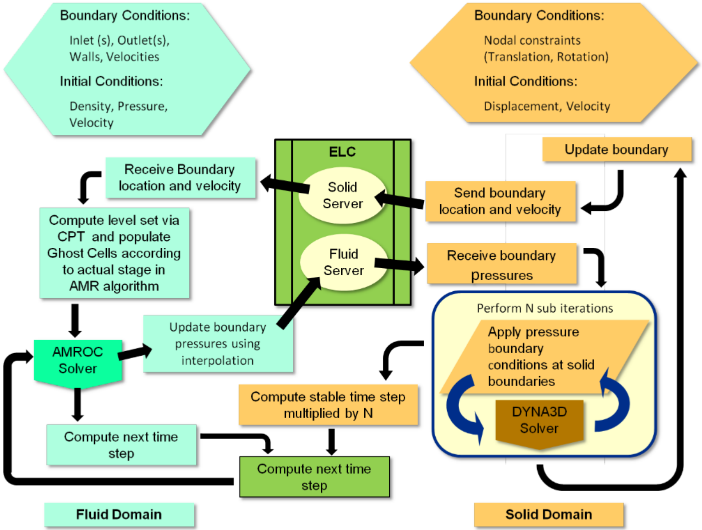
$$u_n^F := u_n^S(t)|_{\mathcal{I}}$$
UpdateFluid(Δt)
$$\sigma_{nn}^S := -p^F(t+\Delta t)|_{\mathcal{I}}$$
UpdateSolid(Δt)
$$t := t + \Delta t$$



Coupling conditions on interface

$$\left.\begin{array}{rcl} u_n^S & = & u_n^F \\ \sigma_{nn}^S & = & -p^F \\ \sigma_{nm}^S & = & 0 \end{array}\right|_{\mathcal{I}}$$

Coupling elements



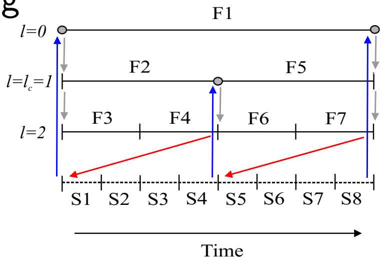
Usage of SAMR

- Eulerian SAMR + non-adaptive Lagrangian FEM scheme
- Exploit SAMR time step refinement for effective coupling to solid solver
  - Lagrangian simulation is called only at level $l_c \le l_{\max}$
  - SAMR refines solid boundary at least at level l_c
  - Additional levels can be used resolve geometric ambiguities
- Nevertheless: Inserting sub-steps accommodates for time step reduction from the solid solver within an SAMR cycle
- Communication strategy:
  - Updated boundary info from solid solver must be received before regridding operation
  - Boundary data is sent to solid when highest level available
- Inter-solver communication (point-to-point or globally) managed on-the-fly by special Eulerian-Lagrangian coupling (ELC) module [Deiterding et al., 2006]

SAMR algorithm for FSI coupling

`AdvanceLevel(`l`)`

Repeat r_l times
- Set ghost cells of $\mathbf{Q}^l(t)$
- CPT(φ^l, C^l, $\mathcal{I}$, δ_l)
- If time to regrid?
  - Regrid(l)
- UpdateLevel(l)
- If level $l+1$ exists?
  - Set ghost cells of $\mathbf{Q}^l(t+\Delta t_l)$
  - AdvanceLevel($l+1$)
  - Average $\mathbf{Q}^{l+1}(t+\Delta t_l)$ onto $\mathbf{Q}^l(t+\Delta t_l)$
- If $l = l_c$?
  - SendInterfaceData($p^F(t+\Delta t_l)|_{\mathcal{I}}$)
  - If $(t+\Delta t_l) < (t_0+\Delta t_0)$?
    - ReceiveInterfaceData($\mathcal{I}$, $\mathbf{u}^S|_{\mathcal{I}}$)
- $t := t + \Delta t_l$



- Call CPT algorithm before Regrid(l)
- Include also call to CPT(·) into Recompose(l) to ensure consistent level set data on levels that have changed
- Communicate boundary data on coupling level l_c

Fluid and solid update / exchange of time steps

```
FluidStep( )
```

$\Delta\tau_F := \min_{l=0,\cdots,l_{\max}} (R_l \cdot \texttt{StableFluidTimeStep}(l),\ \Delta\tau_S)$
$\Delta t_l := \Delta\tau_F / R_l$ for $l = 0, \cdots, L$
$\texttt{ReceiveInterfaceData}(\mathcal{I},\ \mathbf{u}^S|_{\mathcal{I}})$
$\texttt{AdvanceLevel(0)}$

```
SolidStep( )
```

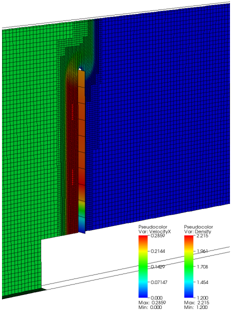
$\Delta\tau_S := \min(K \cdot R_{l_c} \cdot \texttt{StableSolidTimeStep}(),\ \Delta\tau_F)$
Repeat R_{l_c} times
  $t_{\mathrm{end}} := t + \Delta\tau_S / R_{l_c}$, $\Delta t := \Delta\tau_S / (K R_{l_c})$
  While $t < t_{\mathrm{end}}$
    $\texttt{SendInterfaceData}(\mathcal{I}(t),\ \vec{u}^S|_{\mathcal{I}}(t))$
    $\texttt{ReceiveInterfaceData}(p^F|_{\mathcal{I}})$
    $\texttt{UpdateSolid}(p^F|_{\mathcal{I}},\ \Delta t)$
    $t := t + \Delta t$
    $\Delta t := \min(\texttt{StableSolidTimeStep}(),\ t_{\mathrm{end}} - t)$

with $R_l = \prod_{\iota=0}^{l} r_\iota$

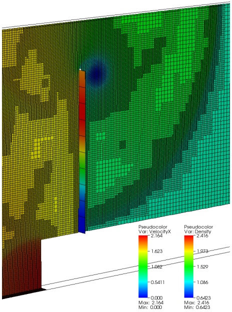
- Time step stays constant for R_{l_c} steps, which correponds to one fluid step at level 0

Shock-driven elastic panel motion

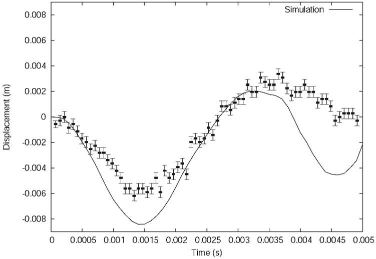
- Thin steel plate (thickness $h = 1\,\mathrm{mm}$, length $50\,\mathrm{mm}$), clamped at lower end
- $\rho_s = 7600\,\mathrm{kg/m^3}$, $E = 220\,\mathrm{GPa}$, $I = h^3/12$, $\nu = 0.3$ [Giordano et al., 2005]
- SAMR base mesh $320 \times 64(\times 2)$, $r_{1,2} = 2$, $l_c = 2$, 4 solid sub-iterations
- Intel 3.4GHz Xeon dual processors, GB Ethernet interconnect
  - $\sim 450\,\mathrm{h}$ CPU on 15 fluid CPU + 1 solid CPU for DYNA3D [Hallquist and Lin, 2005]



$t \approx 0.06\,\mathrm{ms}$



$t \approx 0.24\,\mathrm{ms}$

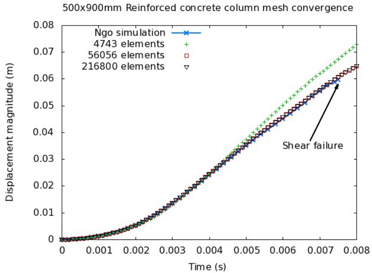


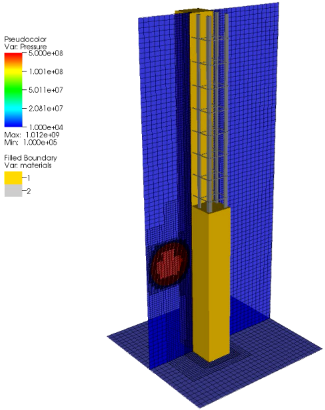
Tip displacement in simulation and experiment

[Deiterding, 2010]

Plastic deformation of reinforced concrete column

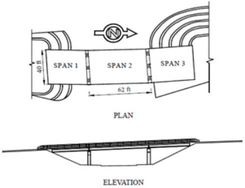
- Column of 6.4 m and 500×900 mm cross-section as in [Ngo et al., 2007]
- DYNA3D elastic-plastic concrete model: strength $\sigma_{\text{max}} = 80\,\mathrm{MPa}$
  - $\rho_s = 2010\,\mathrm{kg/m^3}$, $E = 21.72\,\mathrm{GPa}$, $\nu = 0.2$, yield stress $\sigma_y = 910\,\mathrm{kPa}$, $E_T = 11.2\,\mathrm{GPa}$, $\beta = 0.03$
- Spherical energy deposition $\equiv 150\,\mathrm{kg}$ TNT, 0.5 m distance, 2 m above the ground
- 297 h CPU on 33+1 CPU 3.4 GHz Intel-Xeon

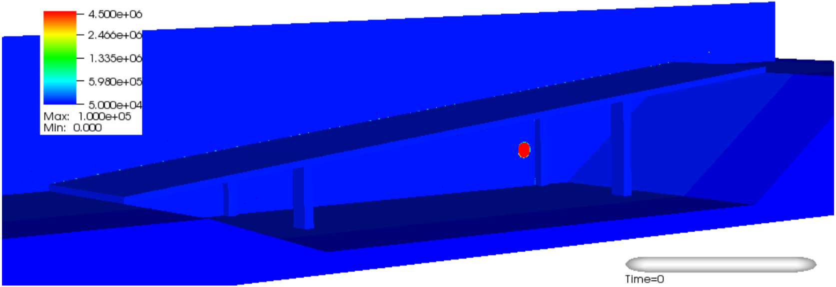




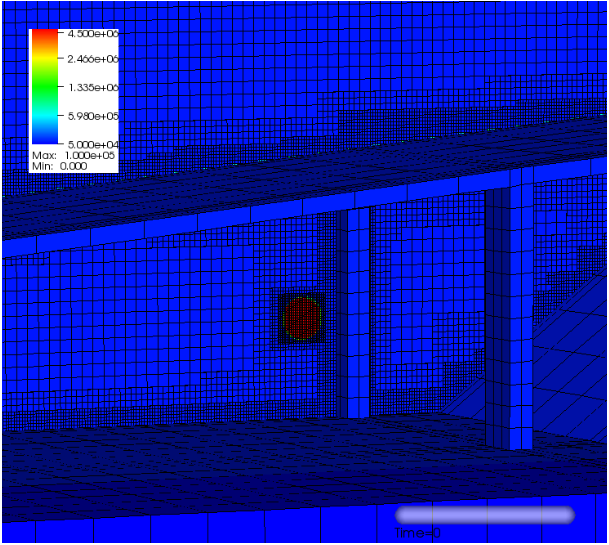
Highway bridge

- Case follows [Agrawal and Yi, 2009]: 150 kg TNT 0.5 m in front of the high middle column, 2 m above the ground
- Concrete modeled with DYNA3D plastic concrete model, 3365 solid hexahedron elements
- SAMR: $240 \times 40 \times 80$ base level, three additional levels $r_{1,2,3} = 2$, $l_c = 2$, $R_{l_c} = 1$
- 487 h CPU on 63+1 CPU 3.4 GHz Intel-Xeon, 1504 coupled time steps to $t_{\text{end}} = 20$ ms





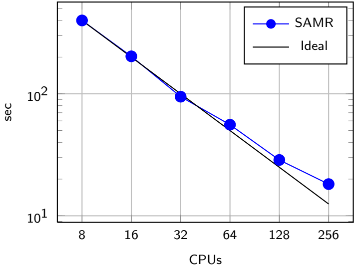
Highway bridge - meshing detail



Coupled FSI - strong scalability

- SAMR: $240 \times 40 \times 80$, two levels: $r_1 = 2$, $r_2 = 4$; coupling: $l_c = 2$, $R_{l_c} = 1$
- Timing done on fluid side for 24 steps on finest level
- $\sim 56,500,000$ cells instead $393,216,000$

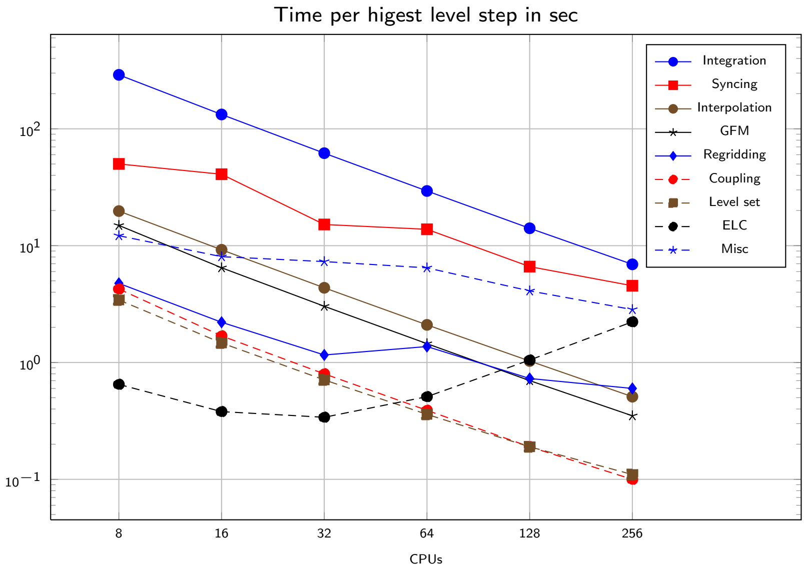
Time per higest level step in sec



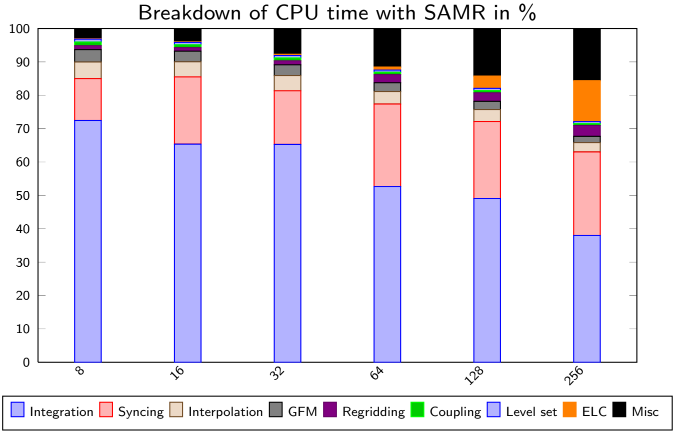
CPUs	8	16	32	64	128	256
Integration	290	133	62	29	14	6.9
Syncing	50	41	15	14	6.6	4.5
Interpolation	20	9.2	4.4	2.1	1.0	0.5
GFM	15	6.5	3.0	1.5	0.7	0.4
Regridding	4.8	2.2	1.1	1.4	0.7	0.6
Coupling	4.3	1.7	0.8	0.4	0.2	0.1
Level set	3.4	1.5	0.7	0.4	0.2	0.1
ELC	0.7	0.4	0.3	0.5	1.1	2.2
Misc	12	8.1	7.3	6.5	4.1	2.8

- *Interpolation:* 1/3 SAMR interpolation, 2/3 GFM extrapolation/interpolation
- *Regridding:* Partition (negligible in this case) + Recompose + Clustering
- *Coupling:* Computation of coupling data on fluid side
- *Level set:* Overhead + CPT algorithm
- *ELC:* waiting to receive solid data on fluid side

Coupled FSI scalability - main operations



Coupled FSI scalability - main operations II



- ELC (waiting for solid data) increases unproportionally in strong scalability test
- Problem: only serial DYNA3D version easily available → change splitting approach slightly and evaluate fluid and solid simultaenously for same time step

Prototypical hydrogen explosion in nuclear reactor

Chapman-Jouguet detonation in hydrogen-air mixture at atmospheric pressure. Euler equations with single exothermic reaction $A \longrightarrow B$

$$\partial_t \rho + \partial_{x_n}(\rho u_n) = 0 \;, \quad \partial_t(\rho u_k) + \partial_{x_n}(\rho u_k u_n + \delta_{kn} p) = 0 \;, k = 1, \dots, d$$

$$\partial_t(\rho E) + \partial_{x_n}(u_n(\rho E + p)) = 0 \;, \quad \partial_t(Y\rho) + \partial_{x_n}(Y\rho u_n) = \psi$$

with

$$p = (\gamma - 1)(\rho E - \frac{1}{2}\rho u_n u_n - \rho Y q_0) \quad \text{and} \quad \psi = -kY\rho \exp\left(\frac{-E_{\rm A}\rho}{p}\right)$$

modeled with heuristic detonation model by [Mader, 1979]

$V := \rho^{-1},\; V_0 := \rho_0^{-1},\; V_{\rm CJ} := \rho_{\rm CJ}$
$Y' := 1 - (V - V_0)/(V_{\rm CJ} - V_0)$
If $0 \le Y' \le 1$ and $Y > 10^{-8}$ then
  If $Y < Y'$ and $Y' < 0.9$ then $Y' := 0$
  If $Y' < 0.99$ then $p' := (1 - Y')p_{\rm CJ}$
    else $p' := p$
  $\rho_{\rm A} := Y'\rho$
  $E := p'/(\rho(\gamma - 1)) + Y' q_0 + \frac{1}{2} u_n u_n$

Used parameters for H_2-Air, stoichiometry 0.5, induction length 3.2 mm, $d_{\rm CJ} \approx 1620\,\mathrm{m/s}$

ρ_0	$0.985\,\mathrm{kg/m^3}$
p_0	$100\,\mathrm{kPa}$
$\rho_{\rm CJ}$	$1.951\,\mathrm{kg/m^3}$
$p_{\rm CJ}$	$1378\,\mathrm{kPa}$
γ	1.266

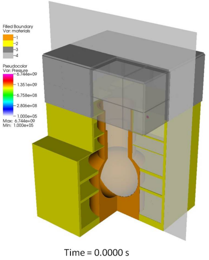
H_2-Air detonation in reactor building

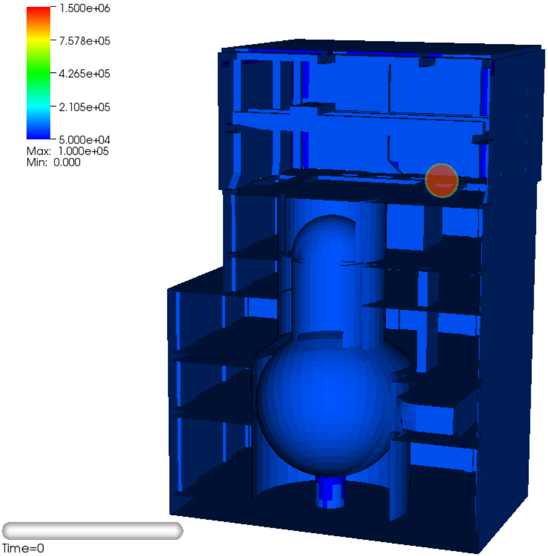
Four materials used

- orange: high strength
- yellow: low strength
- dark gray: concrete, girders
- light gray: paneling

19502 solid hexahedron elements

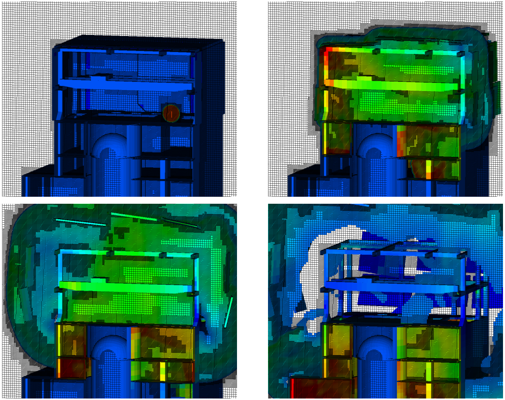
Exemplary ignition in center plane





H_2-Air detonation in reactor building - meshing details

- SAMR base mesh: $120 \times 120 \times 180$, two levels $r_{1,2} = 2$
- coupling level $l_c = 2$, $R_{l_2} = 15$ sub-iterations, 2852 coupled time steps to $t_{\rm end} = 50\,{\rm ms}$
- $3742\,{\rm h}$ CPU on 63+1 CPU 3.4 GHz Intel-Xeon
- $\sim 16,300,000$ ($t = 0$) to $\sim 50,300,000$ ($t_{\rm end}$) instead of $165,888,000$ cells



Conclusions

- Developed and demonstrated the parallel coupling of AMROC with DYNA3D for sophisticated, real-world engineering scenarios
- Future directions
  - Increase level of detail and realism on structural side
  - Investigate more sophisticated detonation models on fluid side
- Parallelization
  - Rigorous domain decomposition scales acceptably for hierarchies that are not too deep and will scale fully in the weak sense
  - Improved strong and weak scalability requires complete elimination of global data for recomposition and partitioning
  - Recomposition and partitioning bottlenecks will be reduced by implementing hybrid MPI-OpenMP parallelization in AMROC
  - Improved scalability for FSI coupled application requires slight algorithmic change to enable overlapping of computation on fluid and solid side

References I

[Agrawal and Yi, 2009] Agrawal, A. K. and Yi, Z. (2009). Blast load effects on highway bridges. Technical report, University Transportation Research Center, City College of New York.

[Arienti et al., 2003] Arienti, M., Hung, P., Morano, E., and Shepherd, J. E. (2003). A level set approach to Eulerian-Lagrangian coupling. *J. Comput. Phys.*, 185:213–251.

[Deiterding, 2010] Deiterding, R. (2010). Parallel adaptive cartesian upwind methods for shock-driven multiphysics simulation. In *Anais do CNMAC - 33rd Brazilian National Congress for Applied and Computational Mathematics*, pages 1048–1057.

[Deiterding, 2011] Deiterding, R. (2011). Block-structured adaptive mesh refinement - theory, implementation and application. *European Series in Applied and Industrial Mathematics: Proceedings*, 34:97–150.

[Deiterding et al., 2009] Deiterding, R., Cirak, F., and Mauch, S. P. (2009). Efficient fluid-structure interaction simulation of viscoplastic and fracturing thin-shells subjected to underwater shock loading. In Hartmann, S., Meister, A., Schäfer, M., and Turek, S., editors, *Int. Workshop on Fluid-Structure Interaction. Theory, Numerics and Applications, Herrsching am Ammersee 2008*, pages 65–80. kassel university press GmbH.

References II

[Deiterding et al., 2007] Deiterding, R., Cirak, F., Mauch, S. P., and Meiron, D. I. (2007). A virtual test facility for simulating detonation- and shock-induced deformation and fracture of thin flexible shells. *Int. J. Multiscale Computational Engineering*, 5(1):47–63.

[Deiterding et al., 2006] Deiterding, R., Radovitzky, R., Mauch, S. P., Noels, L., Cummings, J. C., and Meiron, D. I. (2006). A virtual test facility for the efficient simulation of solid materials under high energy shock-wave loading. *Engineering with Computers*, 22(3-4):325–347.

[Fedkiw, 2002] Fedkiw, R. P. (2002). Coupling an Eulerian fluid calculation to a Lagrangian solid calculation with the ghost fluid method. *J. Comput. Phys.*, 175:200–224.

[Giordano et al., 2005] Giordano, J., Jourdan, G., Burtschell, Y., Medale, M., Zeitoun, D. E., and Houas, L. (2005). Shock wave impacts on deforming panel, an application of fluid-structure interaction. *Shock Waves*, 14(1-2):103–110.

[Gunney et al., 2007] Gunney, B. T., Wissink, A. M., and Hysoma, D. A. (2007). Parallel clustering algorithms for structured AMR. *J. Parallel and Distributed Computing*, 66(11):1419–1430.

References III

[Hallquist and Lin, 2005] Hallquist, J. and Lin, J. I. (2005). A nonlinear explicit three-dimensional finite element code for solid and structural mechanics. Technical Report UCRL-MA-107254, Lawrence Livermore National Laboratory. Source code (U.S. export controlled) available for licensing fee from http://www.osti.gov/estsc.

[Laurence et al., 2007] Laurence, S. J., Deiterding, R., and Hornung, H. G. (2007). Proximal bodies in hypersonic flows. *J. Fluid Mech.*, 590:209–237.

[Mader, 1979] Mader, C. L. (1979). *Numerical modeling of detonations*. University of California Press, Berkeley and Los Angeles, California.

[Mauch, 2003] Mauch, S. P. (2003). *Efficient Algorithms for Solving Static Hamilton-Jacobi Equations*. PhD thesis, California Institute of Technology.

[Ngo et al., 2007] Ngo, T., Mendis, P., Gupta, A., and Ramsay, J. (2007). Blast loading and blast effects on structures - an overview. *Electronic Journal of Structual Engineering*.

[Specht, 2000] Specht, U. (2000). *Numerische Simulation mechanischer Wellen an Fluid-Festkörper-Mediengrenzen*. Number 398 in VDI Reihe 7. VDU Verlag, Düsseldorf.