Adaptive Cartesian methods
Combustion modeling
Detonation simulation
Summary

Detonation and hypersonics simulation with AMROC – Part I

Ralf Deiterding

Aerodynamics and Flight Mechanics Research Group
University of Southampton
Highfield Campus
Southampton SO17 1BJ, UK
Email: r.deiterding@soton.ac.uk

Xiamen
23rd July, 2019

Outline

Adaptive Cartesian finite volume methods
Block-structured AMR with complex boundaries
Parallelization approach

Combustion modeling
Governing equations
Finite volume schemes

Detonation simulation
Shock induced combustion from projectile flight
Thermal ignition
Propagation of regular detonations in 2d
Cellular structures in 3d and their ignition
Detonation-boundary layer interaction

Summary
Conclusions

Collaboration with

Detonations
► Bok Jik Lee (Gwangju Institute of Science and Technology, South Korea)
► Xiaodong Cai, Jiang Liang, Zhiyong Lin (National University of Defense Technology, Changsha)
► Jack Ziegler (now Northrop Grumman), Dale Pullin, Joe Shepherd (Graduate Aeronautical Laboratory, California Institute of Technology)
► Yong Sun, Matthias Ihme (Stanford University)

Hypersonics simulation
► Chay Atkins, Adriano Cerminara, Neil Sandham (University of Southampton)

Block-structured adaptive mesh refinement (SAMR)

For simplicity \( \partial_t q(x, y, t) + \partial_j f(q(x, y, t)) + \partial_j g(q(x, y, t)) = 0 \)

► Refined blocks overlay coarser ones
► Refinement in space and time by factor \( r \) [Berger and Colella, 1988]
► Block (aka patch) based data structures
  + Numerical scheme
  
  \[
  Q_{jk}^{n+1} = Q_{jk}^n - \frac{\Delta t}{\Delta x} \left[ F_{j+\frac{1}{2},k} - F_{j-\frac{1}{2},k} \right] \\
  - \frac{\Delta t}{\Delta y} \left[ G_{j,k+\frac{1}{2}} - G_{j,k-\frac{1}{2}} \right]
  \]

  only for single patch necessary
  + Efficient cache-reuse / vectorization possible
  - Cluster-algorithm necessary
► Papers: [Deiterding, 2011a, Deiterding et al., 2009b, Deiterding et al., 2007]
Level-set method for boundary embedding

- Implicit boundary representation via distance function $\varphi$, normal $\mathbf{n} = \nabla \varphi / |\nabla \varphi|
- Complex boundary moving with local velocity $\mathbf{w}$, treat interface as moving rigid wall [Deiterding et al., 2007]
- Construction of values in embedded boundary cells by interpolation / extrapolation [Deiterding, 2009, Deiterding, 2011a]
- Creation of level set from triangulated surface data with closest-point-transform (CPT) algorithm [Mauch, 2003, Deiterding et al., 2006]
AMROC framework and most important patch solvers

- Implements described algorithms and facilitates easy exchange of the block-based numerical scheme
- Hybrid WENO methods for LES and DNS: [Pantano et al., 2007, Lombardini and Deiterding, 2010, Ziegler et al., 2011, Cerminara et al., 2018]
- FSI deformation from water hammer: [Cirak et al., 2007, Feldhaus et al., 2009a, Perotti et al., 2013, Wan et al., 2017]
- Level-set method for Eulerian solid mechanics: [Barton et al., 2013]
- Ideal magneto-hydrodynamics: [Gomes et al., 2015, Souza Lopes et al., 2018]

Parallelization

- Rigorous domain decomposition
  - Data of all levels resides on same node
  - Grid hierarchy defines unique “floor-plan”
  - Workload estimation
    \[ \mathcal{W}(\Omega) = \sum_{l=0}^{L_{\text{max}}} \mathcal{N}(G_l \cap \Omega) \prod_{k=0}^{l} r_{nk} \]
  - Parallel operations
    - Synchronization of ghost cells
    - Redistribution of data blocks within regridding operation
    - Flux correction of coarse grid cells
  - Dynamic partitioning with space-filling curve

[Deiterding, 2005, Deiterding, 2011a]

AMROC strong scalability tests

- 3D wave propagation method with Roe scheme: spherical blast wave
  - Tests run IBM BG/P (mode VN)
- 3D SRT-lattice Boltzmann scheme: flow over rough surface of 19 × 13 × 2 spheres
  - Tests run Cray XC30m (Archer)

Axisymmetric Navier-Stokes equations with chemical reaction

\[
\frac{\partial q}{\partial t} + \frac{\partial (f - f_v)}{\partial x} + \frac{\partial (g - g_v)}{\partial y} = \frac{\alpha_y}{y} (c - g + g_v) + s
\]

\[ q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho \rho u^2 \rho v \\ \rho v \rho u \rho \rho v^2 \rho \rho \rho E \end{bmatrix}, \quad f = \begin{bmatrix} \rho \rho u \\ \rho u^2 + \rho p \\ \rho v \rho u \\ u (\rho E + p) \end{bmatrix}, \quad g = \begin{bmatrix} \rho \rho v \\ \rho v u \\ \rho v v + \rho E \rho v + \rho v E \end{bmatrix}, \quad c = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad s = \begin{bmatrix} \omega_i \\ 0 \\ 0 \end{bmatrix}
\]

\[ f_v = \begin{bmatrix} \rho D_x \frac{\partial T}{\partial x} + \rho \sum h_j D_j \frac{\partial Y_j}{\partial x} + u \tau_{xx} + \nabla \cdot \mathbf{v} \\ \rho D_y \frac{\partial T}{\partial y} + \rho \sum h_j D_j \frac{\partial Y_j}{\partial y} + v \tau_{yy} + \nabla \cdot \mathbf{v} \\ \rho D_\theta \frac{\partial T}{\partial \theta} + \rho \sum h_j D_j \frac{\partial Y_j}{\partial \theta} + \tau_{x\theta} + \nabla \cdot \mathbf{v} \\ \tau_{xx} = -\frac{2}{3} \mu (\nabla \cdot \mathbf{v}) + 2 \mu \frac{\partial u}{\partial x} \\ \tau_{yy} = -\frac{2}{3} \mu (\nabla \cdot \mathbf{v}) + 2 \mu \frac{\partial v}{\partial y} \\ \tau_{x\theta} = -\frac{2}{3} \mu (\nabla \cdot \mathbf{v}) + 2 \mu \frac{\partial \theta}{\partial \theta} \\ \tau_{x\theta} = \mu \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial \theta}{\partial \theta} \\ \nabla \cdot \mathbf{v} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial \theta}{\partial \theta} 
\]
Equation of state

Ideal gas law and Dalton’s law for gas-mixtures

\[ p(\rho_1, \ldots, \rho_K, T) = \sum_{i=1}^{K} \rho_i = \sum_{i=1}^{K} \rho_i \frac{R}{W_i} T = \rho \frac{R}{W} T \quad \text{with} \quad \sum_{i=1}^{K} \rho_i = \rho, \quad Y_i = \frac{\rho_i}{\rho} \]

Caloric equation

\[ h(Y_1, \ldots, Y_K, T) = \sum_{i=1}^{K} Y_i h_i(T) \quad \text{with} \quad h_i(T) = h_i^0 + \int_0^T c_{pi}(s) \, ds \]

Computation of \( T = T(\rho_1, \ldots, \rho_K, e) \) from implicit equation

\[ \sum_{i=1}^{K} \rho_i h_i(T) - R T \sum_{i=1}^{K} \rho_i \frac{R}{W_i} - \rho e = 0 \]

for thermally perfect gases with \( \gamma_i(T) = c_{pi}(T)/c_v(T) \) using an iterative Newton or bisection method

Chemistry and transport properties

Arrhenius-kinetics:

\[ \omega_i = \sum_{j=1}^{M} (w_{ij} - w_{ij}^p) \left[ k_i^j \prod_{n=1}^{K} \left( \frac{\rho_n}{W_n} \right)^{v_{jn}} - k_i^j \prod_{n=1}^{K} \left( \frac{\rho_n}{W_n} \right)^{v_{jn}} \right] \quad i = 1, \ldots, K \]

- Parsing of mechanisms and evaluation of \( \omega_i \) with Chemkin-II
- \( c_{pi}(T) \) and \( h_i(T) \) tabulated, linear interpolation between values

Mixture viscosity \( \mu = \mu(T, Y_i) \) with Wilke formula

\[ \mu = \sum_{i=1}^{K} \frac{Y_i \mu_i}{W_i} \quad \text{with} \quad \Phi_{im} = \frac{1}{\sqrt{8}} \left( 1 + \frac{W_i}{W_m} \right)^{-\frac{1}{2}} \left( 1 + \left( \frac{\mu_i}{\mu_m} \right)^{\frac{1}{2}} \left( \frac{W_m}{W_i} \right)^{\frac{1}{2}} \right) \]

Mixture thermal conductivity \( k = k(T, Y_i) \) following Mathur

\[ k = \frac{1}{2} \left( W \sum_{i=1}^{K} \frac{Y_i k_i}{W_i} + \frac{1}{W \sum_{i=1}^{K} Y_i / (W_i k_i)} \right) \]

Mixture diffusion coefficients \( D_i = D_i(T, p, Y_i) \) from binary diffusion \( D_{mi}(T, p) \) as

\[ D_i = \frac{1 - Y_i}{W \sum_{m=1}^{K} Y_m / (W_m D_m)} \]

- Evaluation with Chemkin-II Transport library

Finite volume discretization

Time discretization \( t_n = n \Delta t \), discrete volumes \( V_{jk} = [x_j - \frac{1}{2} \Delta x, x_j + \frac{1}{2} \Delta x] \times [x_k - \frac{1}{2} \Delta y, x_k + \frac{1}{2} \Delta y] \times [x_l - \frac{1}{2} \Delta z, x_l + \frac{1}{2} \Delta z] \)

Approximation \( Q_{jk}(t) \approx \frac{1}{V_{jk}} \int_{V_{jk}} q(x, t) \, dx \) and numerical fluxes

\[ F \left( Q_{jk}(t), Q_{j+1,k}(t) \right) \approx f(q(x_{j+1/2}, y_k, t)) \]

\[ F_v \left( Q_{jk}(t), Q_{j+1,k}(t) \right) \approx f_v(q(x_{j+1/2}, y_k, t), \nabla q(x_{j+1/2}, y_k, t)) \]

yield (for simplicity)

\[ Q_{jk}^{n+1} = Q_{jk}^n - \Delta t \frac{\Delta x}{\Delta x} \left[ F \left( Q_{jk}^n, Q_{j+1,k}^n \right) - F \left( Q_{j-1,k}^n, Q_{jk}^n \right) \right] + \frac{\Delta t}{\Delta z} \left[ F_v \left( Q_{jk}^n, Q_{jk+1}^n \right) - F_v \left( Q_{jk-1}^n, Q_{jk}^n \right) \right] \]

- Riemann solver to approximate \( F \left( Q_{jk}^n, Q_{j+1,k}^n \right) \)
- 1st-order finite differences for \( F_v \left( Q_{jk}^n, Q_{jk+1}^n \right) \) yield 2nd-order accurate central differences in \( x \)

Stability condition used:

\[ \max_{i,j,k} \left\{ \frac{\Delta t}{\Delta x} \left( |u_{jk} + c_k| + c_p \right) + \frac{8}{3} \frac{\mu_k \Delta t}{\rho_{jk} \Delta x^2} + \frac{\Delta t}{\Delta z} \left( |u_{jk} + c_z| + c_p \right) + \frac{2k \Delta t}{c_{zjk} \Delta y^2} + \frac{\Delta t}{\Delta z} \left( |u_{jk} + c_y| + c_p \right) + D_{jk} \frac{\Delta t}{\Delta y^2} \right\} \leq 1 \]

Splitting methods

Dimensional splitting for PDE

\[ \chi^{(\Delta t)}: \quad \partial_t q + \chi f(q) = 0, \quad \text{IC: } Q(t_m) \rightarrow \chi^1/2 Q \]

\[ \gamma^{(\Delta t)}: \quad \partial_t q + \chi g(q) = 0, \quad \text{IC: } Q \rightarrow \gamma^1/2 Q \]

Treat right-hand side as source term

\[ C^{(\Delta t)}: \quad \partial_t q = \alpha \chi (q - g(q) + g_v(q)), \quad \text{IC: } \chi \rightarrow \chi^1/2 Q \]

Chemical source term

\[ S^{(\Delta t)}: \quad \partial_t q = s(q), \quad \text{IC: } \chi \rightarrow \chi^1/2 Q(t_m + \Delta t) \]

Formally 1st-order algorithm

\[ Q(t_m + \Delta t) = S^{(\Delta t)} C^{(\Delta t)} \gamma^{(\Delta t)} (Q(t_m)) \]

but all sub-operators 2nd-order accurate or higher.
Finite volume discretization – cont.

Symmetry source term \( C(\Delta t) \): Use

\[
Q_{ik}^{n+1} = Q_{ik}^{n} + \Delta t \left( \frac{\alpha}{2} (c(Q_{ik}^{n}) - g(Q_{ik}^{n})) + \frac{1}{2} (G_v(Q_{ik}^{n}, Q_{ik+1}^{n}) + G_v(Q_{ik-1}^{n}, Q_{ik}^{n})) \right)
\]

within explicit 2nd-order accurate Runge-Kutta method

- Gives 2nd-order central difference approximation of \( G_v \)
- Transport properties \( \mu, k, D_i \) are stored in vector of state \( Q \) and kept constant throughout entire time step

Chemical source term \( S(\cdot) \):

- 4th-order accurate semi-implicit ODE-solver subcycles within each cell
- \( \rho, e, u, v \) remain unchanged!

\[
\partial_t \rho_i = W_i \dot{\omega}_i (\rho_1, \ldots, \rho_K, T) \quad i = 1, \ldots, K
\]

Riemann solver for combustion

(S1) Calculate standard Roe-averages \( \hat{\rho} = \frac{\sqrt{\gamma(\rho \hat{v}_i - \hat{\rho})}}{\sqrt{1 + \rho}}, \hat{\rho} = \frac{\sqrt{\gamma(\rho \hat{v}_i - \hat{\rho})}}{\sqrt{1 + \rho}} \) for \( \hat{u}, \hat{v}, \hat{H}, \hat{Y}_i, \hat{T} \).

(S2) Compute \( \hat{\gamma} = \frac{c}{c_s} \) and \( \hat{\gamma} = \frac{c}{c_s} + \hat{\gamma} \).

(S3) Calculate \( \hat{v}_i := (\hat{V}_i - 1) \) for \( \hat{R}, \hat{T} \) with standard Roe-averages \( \hat{e}_i \) and \( \hat{H}_i \).

(S4) Calculate \( \hat{\epsilon} := \frac{\sum_{i=1}^K \hat{v}_i - (\hat{V}_i - 1)\hat{u}_i^2}{(\hat{V}_i - 1)} \).

(S5) Use \( \Delta q = \hat{q}_m - q_i \) and \( \Delta \hat{p} \) to compute the wave strengths \( a_m \).

(S6) Calculate \( \hat{W}_i = a_i \hat{f}_i, \hat{W}_j = \sum_{i=2}^{K+1} a_i \hat{f}_i, \hat{W}_K = a_{K+1} \hat{f}_{K+1} \).

(S7) Evaluate \( \hat{s}_i = \hat{u} - \hat{e}, \hat{s}_j = \hat{u}, \hat{s}_j = \hat{u} + \hat{\epsilon} \).

(S8) Evaluate \( \hat{P}_{1/R}, \hat{e}_{1/R}, \hat{c}_{1/R} \) from \( q_i^{n+1} = q_i + \hat{W}_i \) and \( q_i^{n+1} = q_i - \hat{W}_i \).

(S9) If \( \hat{P}_{1/R} \leq 0 \) or \( \hat{e}_{1/R} \leq 0 \) use \( \hat{P}_{1/R} = \hat{q}_{1/R} \) and go to (S12).

(S10) Entropy correction: Evaluate \( |\epsilon_i| \).

(S11) Positivity correction: Replace \( F \) by \( F = F_P \), \( F \geq 0 \).

(S12) Evaluate maximal signal speed by \( S = \max(|s_i|, |s_j|) \).

Shock-induced combustion around a sphere

- Spherical projectile of radius 1.5 mm travels with constant velocity \( v_i = 2170.6 \text{ m/s} \) through \( H_2 : O_2 : Ar \) mixture (molar ratios 2:1:7) at 6.67 kPa and \( T = 298 \text{ K} \)
- Mechanism by [Westbrook, 1982]: 34 forward reactions, 9 species
- Axisymmetric Euler simulation on AMR base mesh of 70 × 40 cells
- Comparison of 3-level computation with refinement factors 2.2 (≈ 5 Pts/\( L_g \)) and a 4-level computation with refinement factors 2.2, 4 (≈ 19 Pts/\( L_g \)) at \( t = 350 \mu s \)
- Higher resolved computation captures combustion zone visibly better and at slightly different position (see below)

Iso-contours of \( \rho \) (black) and \( Y_{H_2} \) (white) on refinement domains for 3-level (left) and 4-level computation (right)

Lehr’s ballistic range experiments

- Spherical-nosed projectile of radius 1.5 mm travels with constant velocity through stoichiometric \( H_2 : O_2 : N_2 \) mixture (molar ratios 2:1:3.76) at 42.66 kPa and \( T = 293 \text{ K} \) [Lehr, 1972]
- Chapman Jouguet velocity \( \sim 957 \text{ m/s} \)
- Axisymmetric Navier-Stokes and Euler simulations on AMR base mesh of 400 × 200 cells, physical domain size 6 cm × 3 cm
- 4-level computations with refinement factors 2.2, 4 to final time \( t = 170 \mu s \). Refinement downstream removed.

Main configurations

- Velocity \( v_i = 1931 \text{ m/s} \) (\( M = 4.79 \)), \( \sim 40 \text{ Pts/} L_g \)
- Velocity \( v_i = 1806 \text{ m/s} \) (\( M = 4.48 \)), \( \sim 60 \text{ Pts/} L_g \)

- Various previous studies with not entirely consistent results. E.g. [Yungster and Radhakrishnan, 1996]. [Axdahl et al., 2011]

- Stagnation point location and pressure tracked in every time step
- All computations were on 32 cores requiring \( \sim 1500 \text{ h CPU each} \)
Viscous case – $M = 4.79$ – mesh adaptation

- 5619 iterations with CFL=0.9 to $t = 170 \mu s$
- Oscillation frequency in last 20 $\mu s$: $\sim 722$ kHz (viscous), $\sim 737$ kHz (inviscid)
- Experimental value: $\sim 720$ kHz

Comparison of temperature field

Viscous case – $M = 4.48$

- 5432 iterations with CFL=0.9 to $t = 170 \mu s$
- Oscillation frequency in last 20 $\mu s$: $\sim 417$ kHz
- Experimental value: $\sim 425$ kHz
Oscillation mechanism

Inviscid case – $M = 4.48$

- 4048 iterations with CFL=0.9 to $t = 170 \mu s$
- Oscillation frequency in last 20 $\mu s$: $\sim 395$ kHz
- Experimental value: $\sim 425$ kHz

Schlieren plot of density

Deflagration to detonation transition in 2d

Simulation of regular structures

- CJ detonation for $H_2 : O_2 : Ar$ (2:1:7) at $T_0 = 298$ K and $p_0 = 10$ kPa, cell width 1.6 cm
- Perturb 1d solution with unreacted high-pressure pocket behind front
- Triple point trajectories by tracking $|\omega|$ on auxiliary mesh shifted through grid with CJ velocity. $\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$
- SAMR simulation with 4 additional levels (2,2,2,4), 67.6 Pts/lg
- Configuration similar to Oran et al., J. Combustion and Flame 113, 1998.
Triple point analysis

Detailed Mach reflection structure shortly before the next collision

<table>
<thead>
<tr>
<th>Angle (°)</th>
<th>P/p₀</th>
<th>T/K</th>
<th>u [m/s]</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1.00</td>
<td>298</td>
<td>1775</td>
<td>5.078</td>
</tr>
<tr>
<td>B</td>
<td>31.45</td>
<td>2248</td>
<td>447</td>
<td>0.477</td>
</tr>
<tr>
<td>C</td>
<td>31.69</td>
<td>1775</td>
<td>965</td>
<td>1.153</td>
</tr>
<tr>
<td>D</td>
<td>19.17</td>
<td>1487</td>
<td>1178</td>
<td>1.533</td>
</tr>
<tr>
<td>E</td>
<td>35.61</td>
<td>1856</td>
<td>901</td>
<td>1.053</td>
</tr>
<tr>
<td>F</td>
<td>40.61</td>
<td>1987</td>
<td>777</td>
<td>0.880</td>
</tr>
</tbody>
</table>

On coarse meshes, the high energy release in triple points cannot be captured
Under sufficient resolution, the oscillation frequency is recovered after the bend

The effect of resolution - \( \varphi = 15^° \)

\[ \varphi = 15^° \text{ (left, top), } \varphi = 30^° \text{ (left, bottom), and } \varphi = 60^° \text{ (right)} \]
Triple point structures – $\varphi = 15^\circ$

- Triple point re-initiation after bend with change from transitional to Double Mach reflection

Triple point structures – $\varphi = 30^\circ$

- Triple point quenching and failure as single Mach reflection

Detonation cell structure in 3D

- 44.8 Pts/I$^4$ for $H_2 : O_2 : Ar$ CJ detonation
- SAMR base grid 400x24x24 for one quadrant, 2 additional refinement levels (2, 4)
- Simulation uses ~ 18 M cells instead of ~ 118 M (unigrid)
- ~ 51,000 h CPU on 128 CPU Compaq Alpha. $H$: 37.6 %, $S$: 25.1 %

Temporal Development of Detonation Velocity

- Schematic front view of the periodic triple point line structure right plot at the same time.

Comparison with 2D Simulation
Detonation ignition by a hot jet in 3d

- 3d Euler simulation on AMR base mesh of $64 \times 32 \times 16$ cells
- Domain size $3.2 \text{cm} \times 1.6 \text{cm} \times 0.8 \text{cm}$
- Inflow of $\text{H}_2 : \text{O}_2 : \text{Ar}$ mixture (molar ratios 2:1:7) at 10 kPa and $T = 298$ K at CJ velocity $V_{\text{CJ}} = 1627 \text{ m/s}$
- Hot jet inflow with fully reacted CJ conditions, i.e., $T = 3296$ K, $p = 172.7$ kPa and $\rho = 0.0893 \text{ kg/m}^3$
- Mechanism by [Westbrook, 1982]: 34 forward reactions, 9 species
- Computations on 1024 cores Intel E5-2692 2.20 GHz (Tianhe-2)

Isosurfaces of $\rho$ at $t = 18.85 \mu s$ Isosurfaces of $\rho$ at $t = 224.34 \mu s$ Isosurfaces of $\rho$ at $t = 323.07 \mu s$ Isosurfaces of $\rho$ at $t = 334.10 \mu s$

- Continuous jet injection overdrives the detonation to $\mu s 
- Number of triple point lines is increased compared to CJ case
- Rectangular domain straightens triple point lines
- Primarily TMR triple point line structures visible as in previous case
Dynamic mesh refinement

- Mesh adaptation with 4 additional levels refined by factors 2, 2, 2, 2 → ~ 30.85 Pts/l
- Adaptation indicators similar as before

\[ t = 234.10 \mu s \quad t = 253.32 \mu s \quad t = 272.78 \mu s \quad t = 292.46 \mu s \]

Shock-boundary layer interaction

Non-reactive case

Reactive case: \( \text{H}_2 : \text{O}_2 : \text{Ar} = 15 : 17.85 : 67.15 \)
Detonation establishment in a scramjet combustor

Setup 1 – Experiment $\phi = 0.28$

\[
\begin{align*}
H_2 : O_2 : N_2 & = 0.56 : 1.0 : 2.9, \quad p_0 = 36.1\, \text{kPa}, \quad T_0 = 581\, \text{K}, \quad \text{inflow} \, V_I = 1532\, \text{m/s}, \\
V_{CJ} &= 1431\, \text{m/s}
\end{align*}
\]

Setup 2 – Experiment $\phi = 0.29$

\[
\begin{align*}
H_2 : O_2 & = 0.58 : 1.0 : 2.9, \quad p_0 = 36.1\, \text{kPa}, \quad T_0 = 581\, \text{K}, \quad \text{inflow} \, V_I = 1532\, \text{m/s}
\end{align*}
\]
Numerical simulation $\phi = 0.29$

Conclusions – Detonations

- For small mechanisms, detailed detonation structure simulations and accurate DNS are nowadays possible for realistic 2d geometries
- Accurate studies for idealized 3d configurations feasible
- Resolution down to the scale of secondary triple points can be provided on parallel capacity computing systems
- Enabling components:
  - Splitting methods combined with high-resolution FV schemes for hyrodynamic transport
  - SAMR provides a sufficient spatial and temporal resolution. Savings from SAMR for pipe bend simulations: up to >680x
- Future work will concentrate on non-Cartesian and higher order schemes with low numerical dissipation geared to DNS.

References I


References II


References III


References IV


References VI


Riemann solver for combustion: carbuncle fix

Entropy corrections [Harten, 1983]
[Harten and Hyman, 1983]

1. $|\bar{s}_i| = \begin{cases} s_i & \text{if } |s_i| \geq 2\eta \\ \frac{s_i^2}{4\eta^2} + \eta & \text{otherwise} \end{cases}$

$\eta = \frac{1}{2} \max \{ |s_i(q_{R}) - s_i(q_{L})| \}$

2. Replace $|\bar{s}_i|$ by $|\bar{s}_i|$ only if $s_i(q_{L}) < 0 < s_i(q_{R})$

$\bar{\eta}_{j+1/2,k} = \max \{ \eta_{j+1/2,k}, |\eta_{j,k+1/2} - \eta_{j+1/2,k}|, \eta_{j+1,k-1/2}, |\eta_{j+1,k+1/2} - \eta_{j+1,k+1/2}| \}$

Carbuncle phenomenon

- [Quirk, 1994]
- Test from [Deiterding, 2003]

2D modification of entropy correction [Sanders et al., 1998]:

R. Deiterding – Detonation and hypersonics simulation with AMROC – Part I

Clustering by signatures

$\bar{T}$  Flagged cells per row/column
$\Delta$  Second derivative of $\bar{T}$, $\Delta = \bar{T}_{\nu+1} - 2\bar{T}_{\nu} + \bar{T}_{\nu-1}$

Technique from image detection: [Bell et al., 1994], see also [Berger and Rigoutsos, 1991], [Berger, 1986]

Recursive generation of $\tilde{C}_{i,m}$

1. 0 in $\bar{T}$
2. Largest difference in $\Delta$
3. Stop if ratio between flagged and unflagged cell $> \eta_{tol}$

Detonation and hypersonics simulation with AMROC – Part II

Ralf Deiterding

Aerodynamics and Flight Mechanics Research Group
University of Southampton
Highfield Campus
Southampton SO17 1BJ, UK
Email: r.deiterding@soton.ac.uk

Xiamen
23rd July, 2019
Outline

Two-temperature solver
Thermodynamic model
Cartesian results

Two-temperature mapped mesh solver
Mapped mesh treatment
Non-cartesian results and comparison

DNS with a hybrid method
Higher-order hybrid methods

Summary
Conclusions

Thermodynamic Model

The two temperature thermodynamic model is used to model the thermodynamic nonequilibrium,

$$e_s(T_{tr}, T_{ve}) = e_s^0(T_{tr}) + e_s^* (T_{tr}) + e_s^v (T_{ve}) + e_s^{el} (T_{ve})$$

- Computationally efficient,
- Widely used,
- Integrated into the open source library Mutation++ [Scoggins and Magin, 2014].

The internal energies are calculated within the Mutation++ library using the Rigid-Rotator Harmonic-Oscillator (RRHO) model.

Source Terms

The net species production rates,

$$\dot{w}_s = M_s \sum_{r=1}^{N_r} \left[ \beta_{tr} - \alpha_{tr} \right] \left[ k_{fr} \sum_{i=1}^{N_i} \left( \frac{\rho_i}{M_i} \right)^{\alpha_{sr}} - k_{br} \prod_{i=1}^{N_i} \left( \frac{\rho_i}{M_i} \right)^{\beta_{sr}} \right],$$

$$k_{fr} (T_c) = A_{fr} T_c^{n_{fr}} \exp \left[ - \eta_{fr}/T_c \right],$$

and the energy transfer rate (neutral mixture),

$$Q_{ve} = \sum_i Q_{e}^{T-V} + Q_{e}^{v-x} + Q_{e}^{C-\text{el}},$$

$$Q_{e}^{T-V} = \rho e_s^*(T_{tr}) - e_s^v,$$

$$Q_{e}^{v-x} = c_i \dot{w}_s e_s^{v}, \quad Q_{e}^{C-\text{el}} = c_i \dot{w}_s e_s^{el},$$

are both calculated using the Mutation++ library.
Numerical Integration

Finite volume method with two flux schemes implemented,
- Van Leer’s flux vector splitting method [van Leer, 1982].
- The AUSM scheme [Liou and Steffen Jr, 1993].

Second order in space and time,
- The MUSCL-Hancock scheme is used for the fluxes.
- Strang splitting is used to integrate the source term.

Double Wedge

Simulation of a double wedge in a high enthalpy flow of air [Pezzella et al., 2015].

<table>
<thead>
<tr>
<th>$T_\infty$</th>
<th>$p_\infty$</th>
<th>$U_\infty$</th>
<th>$M_\infty$</th>
<th>$L_1$</th>
<th>$\theta_1$</th>
<th>$L_2$</th>
<th>$\theta_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>710 K</td>
<td>0.78 kPa</td>
<td>3812 m/s</td>
<td>7.14</td>
<td>50.8 mm</td>
<td>30°</td>
<td>25.4 mm</td>
<td>55°</td>
</tr>
</tbody>
</table>

Table: Double wedge geometry and experimental conditions.

- Five species mixture of air.
- Initial $200 \times 200$ cell mesh, with 3 levels of refinement.
- Embedded boundary used to define geometry.
- Van Leer flux scheme.
- Physical time of 242 $\mu$s.
Double Wedge

Dynamic load balancing distributes the cells across the processors.

\[ t = 242 \mu s. \]

The AMR enables the flow features to be captured in detail.

The schlieren image is taken from [Pezzella et al., 2015].

**Mapped Solution Update**

Within the AMROC-Clawpack framework, the solution is stored in physical \((x, y)\) space and the fluxes are mapped from computational \((\xi, \eta)\) space.

Using dimensional splitting the solution update is given by:

\[
Q^{n+1}_{i,j} = Q^n_{i,j} - \frac{\Delta t}{\Delta \xi} \left[ \left( \hat{F} - \hat{F}^v \right)_{i+1,j} - \left( \hat{F} - \hat{F}^v \right)_{i,j} \right] \frac{\Delta \eta \Delta \xi}{V_{i,j}},
\]

\[
Q^{n+1}_{i,j} = Q^n_{i,j} - \frac{\Delta t}{\Delta \eta} \left[ \left( \hat{G} - \hat{G}^v \right)_{i,j+1} - \left( \hat{G} - \hat{G}^v \right)_{i,j} \right] \frac{\Delta \eta \Delta \xi}{V_{i,j}},
\]

where \(V_{i,j}\) is the volume of cell \(i,j\) in physical space. \(\hat{F}, \hat{F}^v, \hat{G}, \hat{G}^v\) are the physical fluxes per computational unit length.

**Mapped Mesh Computation**

In the mapped mesh computations, the flux is transformed to align with the cell face,

\[
\hat{F} = T^{-1}F_n(TQ_l, TQ_r),
\]

where \(T\) is the transformation matrix,

\[
T = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \tilde{\eta} & \tilde{\eta} & 0 & 0 \\
0 & 0 & 0 & -\tilde{\eta} & \tilde{\eta} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 
\end{bmatrix}.
\]
Mapped Inviscid Fluxes

The inviscid fluxes per computational unit length are found by:

- Rotating the momentum components to be normal to the face,
- Calculating the flux with the rotated solution vectors,
- Rotating the solution vector back,
- Scaling the flux using the ratio of the computational face to the mapped face

In the $\xi$ directional sweep, this gives

$$F_{i-1/2,j} = T_{i-1/2,j}^{-1} F_n(T_{i-1/2,j} Q_{i-1,j}, T_{i-1/2,j} Q_{i,j}).$$

where $T$ is the rotation matrix used to rotate the momentum components, and $F_n$ is the normal flux through the face. The scaling is given by:

$$\hat{F}_{i,j} = \frac{|n_{i-1/2,j}|}{\Delta \eta} F_{i-1/2,j},$$

Mapped Viscous Fluxes

The physical viscous flux per computational unit length in the $\xi$ directional sweep is given by,

$$\hat{F}^\nu_{i-1/2,j} = \frac{|n_{i-1/2,j}|}{\Delta \eta} \left[ (F^\nu \hat{n}^\nu)_{i-1/2,j} + (G^\nu \hat{n}^\nu)_{i-1/2,j} \right],$$

To calculate the derivatives needed for $F^\nu$ and $G^\nu$, one must use

$$\frac{\partial \phi}{\partial x} = \left( \frac{\partial \phi}{\partial \xi} \right) \left( \frac{\partial \xi}{\partial x} \right) + \left( \frac{\partial \phi}{\partial \eta} \right) \left( \frac{\partial \eta}{\partial x} \right),$$

and,

$$\frac{\partial \phi}{\partial y} = \left( \frac{\partial \phi}{\partial \xi} \right) \left( \frac{\partial \xi}{\partial y} \right) + \left( \frac{\partial \phi}{\partial \eta} \right) \left( \frac{\partial \eta}{\partial y} \right).$$

Boundary Conditions

For wall boundary conditions the ghost cell values are set by first transforming the domain variables,

$$\hat{Q} = T_w Q_{\text{dom}}.$$

Then setting the ghost cell variables using interpolation,

$$Q_{\text{gc}} = \frac{d_{xw}}{d_{xw}} \hat{Q}_{\text{gw}}$$

and

$$Q_{\text{gc}} = Q_{\text{gw}} \text{ slip,} \quad Q_{\text{gc}} = \frac{d_{xw}}{d_{xw}} \hat{Q}_{\text{gw}} \text{ no slip},$$

Then rotating the ghost cell values using the inverse transformation,

$$Q_{\text{gc}} = T_w^{-1} \hat{Q}_{\text{gc}}.$$

CFL condition

The time step must be adjusted to account for the changes in mesh size. The Courant-Friedrichs-Lewy (CFL) condition can be written as [Moukalal et al., 2015],

$$\sum_f \left[ \lambda_\nu |n_f| + \lambda_\phi |n_f| \right] - \frac{V_c}{\Delta t} \leq 0,$$

where $\lambda_\nu$ and $\lambda_\phi$ are the viscous and convective spectral radii, respectively, and $d_f$ is the distance between the cell centres either side of the face. Rearranging the above equation gives,

$$\frac{\Delta t}{V_c} \sum_f \left[ \lambda_\nu |n_f| + \lambda_\phi |n_f| \right] |n_f| \leq 1.$$
CFL Condition

With dimensional splitting, the CFL condition must be evaluated in each dimension separately, giving,

$$\max \left( \frac{\lambda_i^{j-1/2} \Delta t}{d_i^{j-1/2}}, \lambda_i^{j-1/2} \right) \left| \frac{n_i}{\rho_i} \right|_{i-1/2,j} + \left( \frac{\lambda_i^{j+1/2} \Delta t}{d_i^{j+1/2}}, \lambda_i^{j+1/2} \right) \left| \frac{n_i}{\rho_i} \right|_{i+1/2,j} \leq 1.$$
Experiments of a cylinder in hypersonic flow [Hornung, 1972] were simulated with the mapping and initial conditions given by,

\[ x = \xi \cos(\eta), \quad y = -\xi \sin(\eta). \]

<table>
<thead>
<tr>
<th>Radius</th>
<th>( Y_{N_1} )</th>
<th>( Y_W )</th>
<th>( T_\infty )</th>
<th>( \rho_\infty )</th>
<th>( U_\infty )</th>
<th>( M_\infty )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0127 m</td>
<td>0.927</td>
<td>0.073</td>
<td>1833 K</td>
<td>2.91 kPa</td>
<td>5590 m/s</td>
<td>6.14</td>
</tr>
</tbody>
</table>

\textbf{Table:} Cylinder geometry and freestream conditions

The implementation was verified by comparing a mapped computation with an embedded boundary computation.

\[ t = 100 \mu\text{sec} \]

Preliminary results have been obtained for computations including the viscous flux vectors,

\[ \frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial (\mathbf{F} - \mathbf{F}^v)}{\partial x} + \frac{\partial (\mathbf{G} - \mathbf{G}^v)}{\partial y} = \mathbf{W} \]

where,

\[ \mathbf{F}^v = \begin{bmatrix} -J_{x_1} \\ \vdots \\ -J_{x_N} \\ \tau_{x_1} \\ \tau_{y_1} \\ \kappa_{ve} \frac{\partial T_{ve}}{\partial x} - \sum_{s=1}^{N_s} J_{x_s} \theta_{vb} \\ \kappa_{tr} \frac{\partial T_{tr}}{\partial x} + \kappa_{ve} \frac{\partial T_{ve}}{\partial x} + u \tau_{x_1} + v \tau_{y_1} - \sum_{s=1}^{N_s} J_{x_s} h_{es} \end{bmatrix} \]

and a similar expression is obtained for \( \mathbf{G}^v \).
Viscous Computations

The species diffusion uses a modified version of Fick's diffusion law [Sutton and Gnoatto, 1998],

\[ J_{k,s} = -\rho D_s \frac{\partial Y_s}{\partial x} - Y_s \sum_{r=1}^{N_s} (-\rho D_r \frac{\partial Y_r}{\partial x}). \]

The viscous stress tensor, \( \tau_{ij} \) is given by,

\[ \tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \delta_{ij} \frac{2}{3} \mu \nabla \cdot \mathbf{u}, \]

where \( \delta_{ij} \) is the Kronecker delta.

The diffusion coefficients, the viscosity and the thermal conductivities are all calculated within the Mutation++ library.

Flat Plate Comparison

To test the implementation of the viscous fluxes a comparison between the mapped AMROC solver and the SU2 solver was completed. A hyperbolic tangent mapping to stretch the grid away from the wall, with an initial spacing of 1e-5 m. A Mach 3 flow over a 0.3 m flat plate was simulated using both an isothermal and adiabatic wall using the same mesh in each solver.

Figure: A comparison of the velocity boundary layers at 0.2 m is shown below,
Cylinder Heat Flux Computation

The mapped mesh solver has been validated by simulating a cylinder in a nonequilibrium, high enthalpy flow.

The inflow conditions and results were taken from [Degrez et al., 2009].

<table>
<thead>
<tr>
<th>$T_{\infty}$</th>
<th>$\rho_{\infty}$</th>
<th>$U_{\infty}$</th>
<th>$Y_{N_2}$</th>
<th>$Y_{N_2}$</th>
<th>$Y_{O_2}$</th>
<th>$Y_{NO}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>694 K</td>
<td>3.26 g/m$^3$</td>
<td>4776 m/s</td>
<td>0.7356</td>
<td>0.0</td>
<td>0.1340</td>
<td>0.07955</td>
</tr>
</tbody>
</table>

Table: Freestream conditions for the HEG cylinder simulation.

A cylinder mesh was generated with hyperbolic tangent stretching away from the wall using a $1e^{-6}$ initial spacing.

Hybrid method

Convective numerical flux is defined as

$$ F_{\text{inv}}^n = \begin{cases} F^n_0 \quad \text{in } \mathcal{C} \\ F^n_0 - \text{WENO}^+ \quad \text{in } \mathcal{C}^+ \\ F^n_0 - \text{CD} \quad \text{in } \mathcal{C}^- \end{cases} $$

- For LES: 3rd order WENO method, 2nd order TCD [Hill and Pullin, 2004]
- For DNS: Symmetric 6th order WENO, 6th-order CD scheme

Use WENO scheme to only capture shock waves but resolve interface between species. Shock detection based on using two criteria together:

1. Lax-Liu entropy condition $|u_R \pm a_R| < |u_s \pm a_s| < |u_L \pm a_L|$ tested with a threshold to eliminate weak acoustic waves. Used intermediate states at cell interfaces:
   $$ u_s = \frac{\sqrt{p_L u_L^2 + \gamma p_L R}}{\sqrt{p_L + \gamma p_L}}, \quad a_s = \sqrt{(\gamma + 1)(h_s - \frac{1}{2}u^2)}, \ldots $$

2. Limiter-inspired discontinuity test based on mapped normalized pressure gradient
   $$ \phi(\theta_j) = \frac{2\theta_j}{(1 + \theta_j)^2} \quad \text{with} \quad \theta_j = \frac{|p_{j+1} - p_j|}{|p_{j+1} + p_j|}, \quad \phi(\theta_j) > \alpha_{\text{map}} $$

Results for shear layer in Mach reflection pattern

- WENO/CD - 6 levels
- WENO/CD - 7 levels
- WENO/CD - 8 levels

<table>
<thead>
<tr>
<th>$\Delta x_{\text{min}}$</th>
<th>$\Delta x_{\text{min}}$</th>
<th>$\Delta x_{\text{min}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$3.91 \cdot 10^{-6}$ m</td>
<td>$1.95 \cdot 10^{-6}$ m</td>
<td>$9.77 \cdot 10^{-7}$ m</td>
</tr>
</tbody>
</table>

- MUSCL - 7 levels
- MUSCL - 7 levels - Euler

Usage of WENO for WENO/CD - 8 levels

- WENO/CD/RK3 gives results comparable to 4x finer resolved optimal 2nd-order scheme, but CPU times with SAMR 2-3x larger
- Gain in CPU time from higher-order scheme roughly one order
Detonation ignition by hot jet in 2d

(a) Detailed structure, (b) WENO usage

R. Deiterding

References

Aerodynamics and fluid-structure interaction simulation with AMROC

Part I

Ralf Deiterding

Aerodynamics and Flight Mechanics Research Group
University of Southampton
Highfield Campus
Southampton SO17 1BJ, UK
Email: r.deiterding@soton.ac.uk

Xiamen
24th July, 2019
Outline

Fluid-structure coupling
  Approach
  Rigid body motion
  Thin elastic and deforming thin structures
  Real-world example

Train-tunnel aerodynamics
  Validation
  Passing trains in open space
  Passing trains in a double track tunnel

Summary
  Conclusions

Collaboration with

Finite volume methods
  ▶ Jose M. Garro Fernandez (University of Southampton)
  ▶ Stuart Laurence (Department of Aerospace Engineering, University of Maryland, College Park)
  ▶ Fehmi Cirak (Cambridge University)
  ▶ Sean Mauch, Joe Shepherd, Dan Meiron (California Institute of Technology)

Lattice Boltzmann methods
  ▶ Christos Gkoudesnes, Juan Antonio Reyes Barraza (University of Southampton)
  ▶ Stephen Wood (NASA)
  ▶ Kai Feldhusen, Claus Wagner (German Aerospace Center – DLR)
  ▶ Moritz Fragner (University of Applied Sciences Hannover, Germany)
  ▶ Cinar Laloglu (Marmara University, Turkey)

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]

\[
\begin{align*}
  \Delta t & := t - \Delta t \\
  \sigma_{nm} & := -\rho F (t + \Delta t) \delta_{nm} \\
  \text{UpdateSolid} & (\Delta t) \\
  \text{UpdateFluid} & (\Delta t) \\
  u_n^F & := u_n^S \\
\end{align*}
\]

Coupling conditions on interface

Inviscid fluid:

\[
\begin{align*}
  u_n^F & = u_n^S \\
  p^F & = p^S \\
\end{align*}
\]

Closest point transform algorithm

The signed distance \( \varphi \) to a surface \( I \) satisfies the eikonal equation [Sethian, 1999]

\[
|\nabla \varphi| = 1 \quad \text{with} \quad \varphi|_I = 0
\]

Solution smooth but non-differentiable across characteristics.

Distance computation trivial for non-overlapping elementary shapes but difficult to do efficiently for triangulated surface meshes:

  ▶ Geometric solution approach with closest-point-transform algorithm [Mauch, 2003]
The characteristic / scan conversion algorithm

1. Build the characteristic polyhedrons for the surface mesh
2. For each face/edge/vertex
   2.1 Scan convert the polyhedron.
   2.2 Compute distance to that primitive for the scan converted points
3. Computational complexity.
   ▶ $O(m)$ to build the b-rep and the polyhedra.
   ▶ $O(n)$ to scan convert the polyhedra and compute the distance, etc.
4. Problem reduction by evaluation only within specified max. distance

[Mauch, 2003], see also [Deiterding et al., 2006]

Eulerian/Lagrangian communication module

1. Put bounding boxes around each solid processors piece of the boundary and around each fluid processors grid
2. Gather, exchange and broadcast of bounding box information
3. Optimal point-to-point communication pattern, non-blocking

Coupling elements

Boundary Conditions:
- Inlet (a), Outlet (b), Walls, Velocities
- Initial Conditions:
  - Density, Pressure, Velocity

Compute level set via CPT and populate Ghost Cells according to actual shape in AMR algorithm

Fluid Domain

AMROC Solver

Compute next time step

Send boundary location and velocity

Receive boundary pressures

Update boundary

Apply pressure boundary conditions at solid boundaries

Perform N sub iterations

Compute stable time step multiplied by N

Update boundary locations using interpolation

Update boundary location and velocity

ELC

Solid Domain

Send boundary location and velocity

Receive boundary pressures

Compute stable time step multiplied by N

Proximal bodies in hypersonic flow

Flow modeled by Euler equations for a single polytropic gas with $p = (\gamma - 1) \rho e$

\[
\partial_t \rho + \partial_x (\rho u) = 0, \quad \partial_t (\rho u) + \partial_x (\rho u^2 + p) = 0, \quad \partial_t (\rho E) + \partial_x (u (\rho E + p)) = 0
\]

Numerical approximation with
- Finite volume flux-vector splitting scheme with MUSCL reconstruction, dimensional splitting
- Spherical bodies, force computation with overlaid latitude-longitude mesh to obtain drag and lift coefficients $C_D, C_L$
- Inflow $M = 10$, $C_D$ and $C_L$ on secondary sphere, lateral position varied, no motion
Verification and validation

Static force measurements, $M = 10$:
[Laurence et al., 2007]

- Refinement study: $40 \times 40 \times 32$ base grid, up to without AMR up to $209.7 \times 10^4$ cells, largest run $\sim 35,000$ h CPU

<table>
<thead>
<tr>
<th>$n_{\text{max}}$</th>
<th>$C_D$</th>
<th>$\Delta C_D$</th>
<th>$C_L$</th>
<th>$\Delta C_L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.264</td>
<td>-0.176</td>
<td>1.176</td>
<td>0.157</td>
</tr>
<tr>
<td>2</td>
<td>1.442</td>
<td>0.178</td>
<td>0.019</td>
<td>0.052</td>
</tr>
<tr>
<td>3</td>
<td>1.423</td>
<td>-0.019</td>
<td>0.052</td>
<td>0.071</td>
</tr>
<tr>
<td>4</td>
<td>1.408</td>
<td>-0.015</td>
<td>0.087</td>
<td>0.035</td>
</tr>
</tbody>
</table>

- Comparison with experimental results: $3$ additional levels, $\sim 2000$ h CPU

$C_D$ Experimental | $C_D$ Computational
1.11 ± 0.08 | 1.01
0.29 ± 0.05 | 0.28

Dynamic motion, $M = 4$:
- Base grid $150 \times 125 \times 90$, two additional levels with $n_{1,2} = 2$
- $24,704$ time steps, $36,808$ h CPU on $256$ cores IBM BG/P

Schlieren graphics on refinement regions

Treatment of thin structures

- Thin boundary structures or lower-dimensional shells require “thickening” to apply embedded boundary method
- Unsigned distance level set function $\varphi$
- Treat cells with $0 < \varphi < d$ as ghost fluid cells
- Leaving $\varphi$ unmodified ensures correctness of $\nabla \varphi$
- Use face normal in shell element to evaluate in $\Delta \rho = \rho^+ - \rho^-$
- Utilize finite difference solver using the beam equation
  \[
  \rho_h \frac{\partial^2 w}{\partial t^2} + EI \frac{\partial^4 w}{\partial x^4} = \rho^e
  \]
to verify FSI algorithms

FSI verification by elastic vibration

- Thin steel plate (thickness $h = 1\, \text{mm}$, length $50\, \text{mm}$), clamped at lower end
- $\rho_s = 7600\, \text{kg/m}^3$, $E = 220\, \text{GPa}$, $I = h^3/12$, $\nu = 0.3$
- Modeled with beam solver (101 points) and thin-shell FEM solver (325 triangles) by F. Cirak
- Left: Coupling verification with constant instantenous loading by $\Delta \rho = 100\, \text{kPa}$
- Right: FSI verification with Mach 1.21 shockwave in air ($\gamma = 1.4$)
**Detonation-driven plastic deformation**

Chapman-Jouguet detonation in a tube filled with a stoichiometric ethylene and oxygen ($C_2H_4 + 3O_2, 295K$) mixture. Euler equations with single exothermic reaction $A \rightarrow B$

\[
\frac{\partial}{\partial t}(\rho V) + \frac{\partial}{\partial x_i}(\rho V^i) = 0, \quad \frac{\partial}{\partial t}(\rho V^i) + \frac{\partial}{\partial x_j}(\rho V^j V^i) = \frac{1}{2} \frac{\partial}{\partial x_j}(\sum_{l} V^j V^l p_{ij}) + \delta_{ij}\frac{\partial p}{\partial x_j} + \rho \frac{\partial f^i}{\partial x_j}
\]

with

\[
p = (\gamma - 1)\left(\frac{1}{2}p(u_{n}E + p)\right) - \rho Y_0 \exp\left(-\frac{E_{\Delta \rho}}{\rho}\right)
\]

modeled with heuristic detonation model by Mader, 1979,

\[
V := \rho^{-1}, \quad V_0 := \rho_0^{-1}, \quad V_{CJ} := \rho_{CJ}
\]

\[
\gamma := 1 - (V - V_0)/(V_{CJ} - V_0)
\]

If $0 < Y' < 1$ and $Y' > 10^{-8}$ then

If $Y < Y'$ and $Y' < 0.9$ then $Y' := 0$

If $Y' < 0.9$ then $p' := (1 - \gamma)Y_{CJ}$

else $p' := p$

\[
\rho_A := Y'\rho
\]

\[
E := p'/(\rho(\gamma - 1)) + Y'q_0 + \frac{1}{2}u_{n}u_{n}
\]

**Detonation-driven plastic deformation**

Chapman-Jouguet detonation in a tube filled with a stoichiometric ethylene and oxygen ($C_2H_4 + 3O_2, 295K$) mixture. Euler equations with single exothermic reaction $A \rightarrow B$

\[
\frac{\partial}{\partial t}(\rho V) + \frac{\partial}{\partial x_i}(\rho V^i) = 0, \quad \frac{\partial}{\partial t}(\rho V^i) + \frac{\partial}{\partial x_j}(\rho V^j V^i) = \frac{1}{2} \frac{\partial}{\partial x_j}(\sum_{l} V^j V^l p_{ij}) + \delta_{ij}\frac{\partial p}{\partial x_j} + \rho \frac{\partial f^i}{\partial x_j}
\]

with

\[
p = (\gamma - 1)\left(\frac{1}{2}p(u_{n}E + p)\right) - \rho Y_0 \exp\left(-\frac{E_{\Delta \rho}}{\rho}\right)
\]

modeled with heuristic detonation model by Mader, 1979,

\[
V := \rho^{-1}, \quad V_0 := \rho_0^{-1}, \quad V_{CJ} := \rho_{CJ}
\]

\[
\gamma := 1 - (V - V_0)/(V_{CJ} - V_0)
\]

If $0 < Y' < 1$ and $Y' > 10^{-8}$ then

If $Y < Y'$ and $Y' < 0.9$ then $Y' := 0$

If $Y' < 0.9$ then $p' := (1 - \gamma)Y_{CJ}$

else $p' := p$

\[
\rho_A := Y'\rho
\]

\[
E := p'/(\rho(\gamma - 1)) + Y'q_0 + \frac{1}{2}u_{n}u_{n}
\]
Coupled fracture simulation

Blast explosion in a multistory building

- 20 m × 40 m × 25 m seven-story building similar to [Luccioni et al., 2004]
- Spherical energy deposition ≡ 400 kg TNT, \( r = 0.5 \text{ m} \) in lobby of building
- SAMR: 80 × 120 × 90 base level, three additional levels \( n_2 = 2, \ l_{fsi} = 1, \ k = 1 \)
- Simulation with ground: 1,070 coupled time steps, 830 h CPU (~25.9 h wall time) on 31+1 cores
- ~ 8,000,000 cells instead of 55,296,000 (uniform)
- 69,709 hexahedral elements and with material parameters. [Deiterding and Wood, 2013]

<table>
<thead>
<tr>
<th>Columns</th>
<th>( \rho_s ) [kg/m(^3)]</th>
<th>( \sigma_0 ) [MPa]</th>
<th>( E_T ) [GPa]</th>
<th>( \beta )</th>
<th>( K ) [GPa]</th>
<th>( G ) [GPa]</th>
<th>( \bar{\epsilon}_p )</th>
<th>( \rho_T ) [MPa]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Walls</td>
<td>2010</td>
<td>25</td>
<td>11.2</td>
<td>1.0</td>
<td>21.72</td>
<td>4.67</td>
<td>0.02</td>
<td>-30</td>
</tr>
</tbody>
</table>

验证

- Two-dimensional axi-symmetric computation
- \( \rho_0 = 100 \text{ kPa}, \ \rho_b = 1.225 \text{ kg/m}^3, \ \gamma = 1.4 \)
- Roe shock-capturing scheme blended with HLL
- 2nd order accuracy achieved with MUSCL-Hancock method

实验室隧道模拟器 [Zonglin et al., 2002]
Basic phenomena – $v_0 = 100 \text{ m/s}$
- $800 \times 25$ mesh with Cartesian cut-out (200, 5) to (800, 25)
- 2 level of additional refinement by factor 2

Comparison with experiment – I
Pressure record at location (1020 mm, 20 mm) inside tunnel

R. Deiterding – Aerodynamics and fluid-structure interaction simulation with AMROC Part I

Pressure record at (1020 mm, 20 mm) for $v_0 = 75 \text{ m/s}$. Experiment (left) and AMROC (right)

Comparison with experiment – I
Pressure record at (40 mm, 20 mm) for model velocity $v_0 = 100 \text{ m/s}$. Experiment (left) and AMROC (right)

Variation of velocity and nose half angle

Pressure sensor at (420 mm, 20 mm)

- Dependence on $v_0^2$ is the dynamic pressure influence (left)
- For constant blockage ratio and body velocity, using more pointed noses alleviates the maximal pressure level (right, nose half angle varied)
- For $v_0 \approx 140 \text{ m/s}$ a shock wave (tunnel boom) can be observed. Sharper noses also delay this phenomenon.
NGT2 prototype setup

- Next Generation Train 2 (NGT2) geometry by the German Aerospace Centre (DLR) [Fragner and Deiterding, 2016, Fragner and Deiterding, 2017]
- Mirrored train head of length $\sim 60$ m, no wheels or tracks, train models $0.17$ m above ground above the ground level.
- Train velocities $100$ m/s and $-100$ m/s, middle axis $6$ m apart, initial distance between centers $200$ m
- Base mesh of $360 \times 40 \times 30$ for domain of $360 \times 40 \times 30 \times 30$ m
- Two/three additional levels, refined by $r_{1,2,3} = 2$. Refinement based on pressure gradient and level set and regenerated at every coarse time step. Parallel redistribution at every level-0 time step.
- On 96 cores Intel Xeon E5-2670 2.6 GHz a final $t_e = 3$ sec was reached after $12,385$ sec / $43,395$ sec wall time, i.e., $330$ h and $1157$ h CPU

Passing in open space – AMR and dynamic distribution

Domains of three-level refinement

Distribution to 96 processors

Enlargement of domain center shown

Pressure isosurfaces

Pressure transects
Setup with realistic tunnel shape

- Two NGT2 trains again at velocities 100 m/s and ~100 m/s
- Prototype straight double track tunnel of 640 m length, initial distance between centers of trains 820 m
- Base mesh of $1060 \times 36 \times 24$ for domain of $1060 \times 36 \times 24 \times 24$, three levels refined by $r_{1,2,3}=2$
- On 96 cores Intel Xeon E5-2670 2.6 GHz a final $t_f = 5 \text{ sec}$ was reached after 84,651 sec wall time, i.e., 2257 h CPU

Pressure transects

![Pressure record at 1.39875 seconds for trains intersecting inside a double-track tunnel](image)

Conclusions – compressible flow aerodynamics

- A Cartesian embedded boundary method for compressible flows with block-based adaptive mesh refinement is an efficient and scalable prediction tool for pressure and shock waves created by moving bodies
- Multi-resolution and fluid-structure coupling problems can be tackled without expensive mesh regeneration
  - Level set approach easily handles large motions, element failure and removal
  - Dynamic adaptation ensures high resolution at embedded boundaries and essential flow features
- Aerodynamics of bodies with large motion are easily accessible
  - Current inviscid approach predicts maximal overpressure in front of trains reliably
  - For predicting the flow around entire trains, the boundary layer growing over the train body needs to be considered.
- AMROC solvers for the compressible Navier-Stokes equations and even LES are already available, however, for this particular application a turbulent wall function on the embedded boundary first needs to be implemented. Such a wall function is currently work-in-progress for the LBM-LES solver.

References I

References II


Aerodynamics and fluid-structure interaction simulation with AMROC Part II

Ralf Deiterding

Aerodynamics and Flight Mechanics Research Group
University of Southampton
Highfield Campus
Southampton SO17 1BJ, UK
Email: r.deiterding@soton.ac.uk

Xiamen
24th July, 2019

Outline

Adaptive lattice Boltzmann method
- Construction principles
- Verification and validation
- Thermal LBM

Large-eddy simulation
- LES models
- Verification for homogeneous isotropic turbulence

Realistic aerodynamics computations
- Vehicle geometries
- Wind turbine benchmark
- Wake interaction prediction

Non-Cartesian lattice Boltzmann method
- Construction principles
- Verification and validation for 2d cylinder

Summar y

Approximation of Boltzmann equation

Is based on solving the Boltzmann equation with a simplified collision operator

\[ \partial_t f + \mathbf{u} \cdot \nabla f = \omega (f^{eq} - f) \]

- \( \mathbf{Kn} = l_i/L \ll 1 \), where \( l_i \) is replaced with \( \Delta x \)
- Weak compressibility and small Mach number assumed
- Assume a simplified phase space

Equation is approximated with a splitting approach.

1.) Transport step solves \( \partial_t f_\alpha + \mathbf{e}_\alpha \cdot \nabla f_\alpha = 0 \)

Operator: \( \mathcal{T}: \tilde{f}_\alpha(x + \mathbf{e}_\alpha \Delta t, t + \Delta t) = f_\alpha(x, t) \)

\[ \rho(x, t) = \sum_{\alpha=0}^{18} f_\alpha(x, t), \quad \rho(x, t)\mathbf{u}(x, t) = \sum_{\alpha=0}^{18} \mathbf{e}_\alpha f_\alpha(x, t) \]

Discrete velocities:

\[ \mathbf{e}_\alpha = \begin{cases} 0, & \alpha = 0, \\ (\pm 1, 0, 0)c, (0, \pm 1, 0)c, (0, 0, \pm 1)c, & \alpha = 1, \ldots, 6, \\ (\pm 1, \pm 1, 0)c, (\pm 1, 0, \pm 1)c, (0, \pm 1, \pm 1)c, & \alpha = 7, \ldots, 18, \end{cases} \]
Approximation of equilibrium state

2.) Collision step solves \( \partial_t f_\alpha = \omega (f_\alpha^{eq} - f_\alpha) \)
Operator \( \mathcal{C} \):
\[
f_\alpha(\cdot, t + \Delta t) = \tilde{f}_\alpha(\cdot, t + \Delta t) + \omega_\alpha \Delta t \left( \tilde{f}_\alpha^{eq}(\cdot, t + \Delta t) - \tilde{f}_\alpha(\cdot, t + \Delta t) \right)
\]
with equilibrium function
\[
f_\alpha^{eq}(\rho, u) = \rho_0 \left[ 1 + \frac{3 e_{\alpha} u}{c^2} + \frac{9 (e_{\alpha} u)^2}{2 c^4} - \frac{3 u^2}{2 c^2} \right]
\]
with \( \rho_0 = \frac{1}{3} \left( \frac{1}{2} + \frac{1}{3} \cdot \frac{1}{4} + \frac{1}{2} \cdot \frac{1}{4} + \frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{4} + \frac{1}{2} \cdot \frac{1}{2} \right) \)
Pressure \( \delta p = \sum_\alpha f_\alpha^{eq} c^2 = \rho c^2 \).
Dev. stress \( \Sigma_\alpha = \left( 1 - \omega_\alpha \Delta t \right) \sum_\alpha e_{\alpha} e_{\alpha} (f_\alpha^{eq} - f_\alpha) \)
Is derived by assuming a Maxwell-Boltzmann distribution of \( f_\alpha^{eq} \) and approximating the involved \( \exp() \) function with a Taylor series to second-order accuracy.
Using the third-order equilibrium function
\[
f_\alpha^{eq}(\rho, u) = \rho_0 \left[ 1 + \frac{3 e_{\alpha} u}{c^2} + \frac{9 (e_{\alpha} u)^2}{2 c^4} - \frac{3 u^2}{2 c^2} + e_{\alpha} / 3 \left( \frac{9 (e_{\alpha} u)^2}{2 c^4} - \frac{3 u^2}{2 c^2} \right) \right]
\]
allows higher flow velocities.

Relation to Navier-Stokes equations

Inserting a Chapman-Enskog expansion, that is,
\[
f_\alpha = f_\alpha(0) + \epsilon f_\alpha(1) + \epsilon^2 f_\alpha(2) + ...
\]
and using
\[
\frac{\partial}{\partial t} = \epsilon \frac{\partial}{\partial t_1} + \epsilon^2 \frac{\partial}{\partial t_2} + ..., \quad \nabla = \epsilon \nabla_1 + \epsilon^2 \nabla_2 + ... \]
into the LBM and summing over \( \alpha \) one can show that the continuity and moment equations are recovered to \( O(\epsilon^2) \) [Hou et al., 1996]
\[
\partial_t p + \nabla \cdot (\rho \mathbf{u}) = 0
\]
\[
\partial_t \mathbf{u} + \nabla \cdot \mathbf{u} = -\nabla p + \nu \nabla^2 \mathbf{u}
\]
Kinematic viscosity and collision time are connected by
\[
\nu = \frac{1}{3} \left( \frac{\tau_1}{\Delta t} - \frac{1}{2} \right) c \Delta x
\]
from which one gets with \( \sqrt{3} c_e = \frac{\Delta x}{\Delta t} \) [Hähnel, 2004]
\[
\omega_L = \tau_{\epsilon}^{-1} = \frac{c^2}{\nu + \Delta tc^2/2}
\]

Initial and boundary conditions

- Initial conditions are constructed as \( f_\alpha^{eq}(\rho(t = 0), u(t = 0)) \)
Boundary conditions (applied before streaming step)

No-slip

Slip

Symmetry

Outlet basically copies all distributions (zero gradient)
Inlet and pressure boundary conditions use \( f_\alpha^{eq} \)

Complex geometry:
- Use level set method as before to construct macro-values in embedded boundary cells by interpolation / extrapolation [Deiterding, 2011].
- Distance function \( \varphi \), normal \( \mathbf{n} = \nabla \varphi / |\nabla \varphi| \). Triangulated meshes use CPT algorithm [Mauch, 2003].
- Construct macro-velocity in ghost cells for no-slip BC as \( \mathbf{u}^f = 2\mathbf{w} - \mathbf{u} \)
- Then use \( f_\alpha^{eq}(\rho, u') \) or interpolated bounce-back [Bouzidi et al., 2001] to construct distributions in embedded ghost cells

Normalization

The method is implemented on the unit lattice with \( \Delta \tilde{x} = \Delta \tilde{t} = 1 \)
\[
\Delta \tilde{x} / \tilde{b}_0 = 1, \quad \Delta \tilde{t} / \tilde{t}_0 = 1 \rightarrow c = 1
\]
Lattice viscosity \( \tilde{\nu} = \frac{1}{3} \left( \tau - \frac{1}{2} \right) \) and lattice sound speed \( \tilde{c}_s = \frac{1}{\sqrt{\tilde{\nu}}} \) yield again
\[
\omega_L = \frac{c^2}{\tilde{\nu} + \tilde{c}_s^2/2} = \frac{c^2}{\tilde{\nu} + \Delta \tilde{t}c^2/2}
\]
Velocity normalization factor: \( \tilde{u}_0 = \frac{b_0}{\tilde{\nu}} \), density \( \rho_0 \)
\[
Re = \frac{uL}{\tilde{\nu}} = \frac{u/\tilde{u}_0 \cdot 1/\tilde{b}_0}{\tilde{\nu}/(\tilde{u}_0 \tilde{b}_0)} = \frac{\tilde{u}L}{\tilde{\nu}}
\]
Trick for scheme acceleration: Use \( \tilde{u} = Su \) and \( \tilde{v} = S \tilde{v} \) which yields
\[
\tilde{w}_k = \frac{c^2_L}{S \tilde{v} + \Delta \tilde{t}/S \tilde{c}_s^2/2}
\]
For instance, the physical hydrodynamic pressure is then obtained for a caloric gas as
\[
p = (\tilde{\rho} - 1) c^2_L \frac{\tilde{b}_0^2}{\tilde{c}_s^2} \rho_0 + \frac{c^2_s \rho_0}{\gamma}
\]
Adaptive LBM

1. Complete update on coarse grid: 
   \[ f_{\alpha,in}^{C,n+1} := C \mathcal{T}(f_{\alpha,in}^{C,n}) \]
2. Interpolate \( f_{\alpha,in}^{C,n} \) onto \( f_{\alpha,in}^{C,n} \) to fill fine halos. Set physical boundary conditions.
3. \( \tilde{f}_{\alpha,n}^{C} := \mathcal{T}(f_{\alpha,n}^{C}) \) on whole fine mesh. 
   \[ f_{\alpha,n+1/2}^{C} := C(f_{\alpha,n}^{C}) \] in interior.
4. \( \tilde{f}_{\alpha,n+1/2}^{C} := \mathcal{T}(f_{\alpha,n+1/2}^{C}) \) on whole fine mesh. 
   \[ f_{\alpha,n+1/2}^{C} := C(f_{\alpha,n+1/2}^{C}) \] in interior.
5. Average \( \tilde{f}_{\alpha,n+1/2}^{C} \) (inner halo layer), \( \tilde{f}_{\alpha,out}^{C} \) (outer halo layer) to obtain \( \tilde{f}_{\alpha,out}^{C,n} \).
6. Revert transport into halos: 
   \[ \tilde{f}_{\alpha,out}^{C,n} := \mathcal{T}^{-1}(f_{\alpha,n}^{C}) \]
7. Parallel synchronization of \( f_{\alpha,in}^{C,n} \), \( \tilde{f}_{\alpha,n}^{C} \)
8. Cell-wise update where correction is needed: 
   \[ f_{\alpha,n+1} := C \mathcal{T}(f_{\alpha,n+1/2}^{C}, f_{\alpha,in}^{C,n}) \]

Algorithm equivalent to [Chen et al., 2006].

Oscillating cylinder – Setup

- Motion imposed on cylinder
  \[ y(t) = A_s \sin(2\pi f_t t), \quad \theta(t) = A_\theta \sin(2\pi f_\theta t) \]
- Setup follows [Nazarinia et al., 2012]. Here \( A_d = 1 \) for all cases.
- Natural frequency of cylinder \( f_d \approx 0.6154 \text{s}^{-1} \).
- Strouhal number \( S_t = \frac{f_t D}{U_\infty} \approx 0.198 \) for all cases.
- Chosen here \( D = 20 \text{mm} \)
- Fluid is water with \( c_s = 1482 \text{m/s}, \nu = 9.167 \cdot 10^{-7} \text{m}^2/\text{s}, \)
  \( \rho = 1016 \text{kg/m}^3 \)
- Constant coefficient model deactivated for Case 1, active for Case 2 with \( C_{\alpha} = 0.2 \)

Flow over 2D cylinder, \( d = 2 \text{cm} \)

- Air with 
  \( \nu = 1.61 \cdot 10^{-5} \text{m}^2/\text{s}, \quad \rho = 1.205 \text{kg/m}^3 \)
- Domain size 
  \([-8d, 24d] \times [-8d, 8d] \)
- Dynamic refinement based on velocity. Last level to refine structure further.
- Base lattice \( 320 \times 160, 3 \) additional levels with factors 2, 4, 4.
- Resolution: \( \approx 320 \) points in diameter \( d \)
- Computation of \( C_D \) on 400 equidistant points along circle and averaged over time. Comparison above with [Henderson, 1995].

Case 1b, \( V_R = 1, \text{Re} = 1322 \)

- Visualization enlargement of cylinder region
- Base mesh is discretized with \( 320 \times 160 \) cells, \( 3 \) additional levels with factor \( r = 2, 2, 2 \)
- 80 cells within \( D \) on highest level
- Speedup \( S = 2000 \)
- Basically identical setup in commercial code XFlow for comparison
Case 1b, $V_R = 1$, $f_t = f_\theta = 0.6$, $Re = 1322$

**AMROC**

- Increase of rotational velocity leads to formation of a vortex pair plus single vortex. Drag and lift amplitude roughly doubled.
- Laminar results in good agreement with experiments of [Nazarinia et al., 2012].

**XFlow**

- Configuration by [Toomey and Eldredge, 2008].
- Vortex. Drag and lift amplitude roughly doubled.

### Computational performance

<table>
<thead>
<tr>
<th>Flow type</th>
<th>Case</th>
<th>$\Delta t_0$ [s]</th>
<th>Total cells</th>
<th>$\Delta t_e$ [s]</th>
<th>$\text{Re}$</th>
<th>$y^+$</th>
<th>CPU time [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laminar</td>
<td>1a</td>
<td>0.0015</td>
<td>AMROC 85982</td>
<td>84778</td>
<td>3.33</td>
<td>1322</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1b</td>
<td>0.0015</td>
<td>91774</td>
<td>90488</td>
<td>3.33</td>
<td>1322</td>
<td>0</td>
</tr>
<tr>
<td>Turbulent</td>
<td>2a</td>
<td>0.00031</td>
<td>232840</td>
<td>216452</td>
<td>1.66</td>
<td>6310</td>
<td>2.4</td>
</tr>
<tr>
<td></td>
<td>2b</td>
<td>0.00031</td>
<td>255582</td>
<td>246366</td>
<td>1.66</td>
<td>6310</td>
<td>2.6</td>
</tr>
</tbody>
</table>

- Intel-Xeon-3.50-GHz desktop workstation with 6 cores, communication through MPI
- Same base mesh and always three additional refinement levels
- AMROC: single-relaxation time LBM, block-based mesh adaptation
- XFlow: slightly more multi-relaxation time LBM, cell-based mesh adaptation
- AMROC uses $\sim 7.5\%$ more cells on average more cells
- Normalized on cell number Case 2a is $50\%$ more expensive for XFlow than for AMROC-LBM
- Case 2b is $42\%$ more expensive in CPU time alone

Case 2a, $V_R = 0.5$, $f_t = f_\theta = 3$, $Re = 6310$

**AMROC**

- Oscillation period: $T = 1/f_t = 0.33s$. 10 regular vortices in 1.67 s.
- CPU time on 6 cores for AMROC: 635.8 s, XFlow $\sim 50\%$ more expensive when normalized based on number of cells

**XFlow**

- Configuration by [Toomey and Eldredge, 2008].
- Manufactured bodies in tank filled with water.
- Prescribed translation and rotation

\[
X_\ell(t) = \frac{A_0}{2} \frac{G_1(f_t)}{\max G_\ell} C(f_t), \quad \alpha_\ell(t) = -\beta \frac{G_1(f_t)}{\max G_\ell}
\]

with $G_\ell(t) = \tanh[\sigma_\ell \cos(2\pi t + \Phi)]$.

\[
G_\ell(t) = \int_0^t \tanh[\sigma_\ell \cos(2\pi t')] dt'
\]

- 7 cases constructed by varying $\sigma_\ell$, $\sigma_t$, $\Phi$
- Rotational Reynolds number $Re_\ell = 2\pi \beta \sigma_\ell f^2 / (\tanh(\sigma_\ell) \nu)$ varied between 2200 and 7200 in experiments
- [Toomey and Eldredge, 2008] reference simulations with a viscous particle method are for $Re_\ell = \{100, 500\}$

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_0$ (cm)</td>
<td>7.1</td>
</tr>
<tr>
<td>$c$ (cm)</td>
<td>5.1</td>
</tr>
<tr>
<td>$d$ (cm)</td>
<td>0.25</td>
</tr>
<tr>
<td>$\rho_0$ (kg/m$^3$)</td>
<td>5080</td>
</tr>
<tr>
<td>$f$ (Hz)</td>
<td>0.15</td>
</tr>
</tbody>
</table>
Case 1 - $\sigma_r = \sigma_t = 0.628$, $\Phi = 0$, $Re_r = 100$

- Quiescent water
  $\rho_r = 997 \text{kg/m}^3$
  $c_s = 1497 \text{m/s}$
- No-slip boundaries in $y$, periodic in $x$-direction
- Base level: $100 \times 100$ for $[-0.5, 0.5] \times [-0.5, 0.5]$ domain
- 4 additional levels with factors $2, 2, 2, 4$
- Coupling to rigid body motion solver on 4th level

Right: computed vorticity field (enlarged)

An LBM for thermal transport

Consider the Navier-Stokes equations under Boussinesq approximation

\[ \nabla \cdot \mathbf{u} = 0 \]
\[ \frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \mathbf{u}) = -\nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{F} \]
\[ \frac{\partial T}{\partial t} + \nabla \cdot (\mathbf{u} T) = D \nabla^2 T \]

with $\mathbf{F} = g \beta (T - T_{ref})$.

An LBM for this system needs to use two distribution functions $f_\alpha$ and $g_\alpha$.

1.) Transport step $T^t$:

\[ \tilde{f}_\alpha(x + \mathbf{e}_\alpha \Delta t, t + \Delta t) = f_\alpha(x, t) \]
\[ \tilde{g}_\alpha(x + \mathbf{e}_\alpha \Delta t, t + \Delta t) = g_\alpha(x, t) \]

2.) Collision step $C$:

\[ f_\alpha(x, t + \Delta t) = \tilde{f}_\alpha(x, t + \Delta t) + \omega_{L,\nu} \Delta t \left( \tilde{f}_\alpha(x, t + \Delta t) - \tilde{f}_\alpha(x, t) \right) + \Delta t \mathbf{F}_\alpha \]
\[ g_\alpha(x, t + \Delta t) = \tilde{g}_\alpha(x, t + \Delta t) + \omega_{L,D} \Delta t \left( \tilde{g}_\alpha(x, t + \Delta t) - \tilde{g}_\alpha(x, t) \right) \]

with collision frequencies

\[ \omega_{L,\nu} = \frac{c_s^2}{\nu + c_s^2 \Delta t/2} \]
\[ \omega_{L,D} = \frac{\frac{1}{2} c_s^2}{D + \frac{3}{2} c_s^2 \Delta t/2} \]

Quantitative comparison

- Evaluate normalized force $F_{x,y} = 2 F_{x,y}^* / (\rho L^2 c^4)$ and moment $M = 2 M^* / (\rho L^4 c^6)$ over 3 periods
- [Wood and Deiterding, 2015] Used finest spatial resolution $\Delta x/c = 0.0122$
- [Toomey and Eldredge, 2008]: $\Delta x/c = 0.013$ ($Re_r = 100$), $\Delta x/c = 0.0032$ ($Re_r = 500$)
- Temporal resolution $\sim 113$ and $\sim 28$ times finer

Right: Hinge deflection angle over time

Case 1
- $\sigma_r = 0.628$
- $\sigma_t = 0.628$
- $\Phi = 0$

Case 2
- $\sigma_r = 1.885$
- $\sigma_t = 1.885$
- $\Phi = 0$

Experimental results (-): Current (-)

Equilibrium operators

This incompressible method uses in 2D [Guo et al., 2002]

\[ f_{\alpha}^{(eq)} = \begin{cases} -4 \sigma_0 p - s_\alpha \mathbf{u}, & \text{for } \alpha = 0, \\ \sigma_\alpha \mathbf{u} + s_\alpha \mathbf{u}, & \text{for } \alpha = 1, \ldots, 8, \end{cases} \]

where

\[ s_\alpha \mathbf{u} = t_\alpha \left[ 3 e_\alpha \mathbf{u} \frac{g_\alpha \mathbf{u}^2}{c^2} + \frac{9 (e_\alpha \mathbf{u})^2}{2 c^4} - \frac{3 \mathbf{u}^2}{2 c^2} \right] \]

with $t_\alpha = \frac{1}{3} \{4, 1, 1, 1, \frac{1}{2}, \frac{1}{2}, 1, \frac{1}{2}, \frac{1}{2}\}$ and $\sigma_\alpha = \frac{1}{3} \{-5, 1, 1, 1, \frac{1}{2}, \frac{1}{2}, 1, \frac{1}{2}, \frac{1}{2}\}$

\[ g_{\alpha}^{(eq)} = \frac{T}{4} [1 + 2 e_\alpha \cdot \mathbf{u}] \text{ for } \alpha = 1, \ldots, 4 \]

Forces are applied in $y$-direction only:

\[ F_\alpha = \frac{1}{2} (\delta_{13} - \delta_{6}) e_i \cdot \mathbf{F} \]

Moments: $\mathbf{u} = \sum_{\alpha} e_i f_\alpha$, $p = 1/4 \rho \left( \sum_{\alpha} f_\alpha + s_\alpha \mathbf{u} \right)$, $T = \sum_{\alpha=1}^{4} g_\alpha$
Heated rotating cylinder

- \( R = 15 \), domain: \([-6R, 16R] \times [-8R, 8R]\)
- \( Re = 2U_{\infty}R / \nu = 200 \), \( U_{\infty} = 0.01 \)
- Peripheral velocity \( V = \Omega R \), \( V / U_{\infty} = 0.5 \)
- Base grid \( 288 \times 240 \) refined by three levels with \( n_1 = 2 \), \( n_2, n_3 = 4 \) using scaled gradients of \( u, v, T \)

\[
\begin{align*}
u & = 0, \quad \frac{\Delta u}{\Delta t} = 0, \quad \frac{\Delta T}{\Delta t} = 0 \\
\frac{\Delta u}{\Delta t} & = 0, \quad \frac{\Delta v}{\Delta t} = 0, \quad \frac{\Delta T}{\Delta t} = 0 \\
\frac{\Delta u}{\Delta t} & = 0, \quad \frac{\Delta v}{\Delta t} = 0, \quad \frac{\Delta T}{\Delta t} = 0
\end{align*}
\]

Turbulence modeling

Pursue a large-eddy simulation approach with \( \bar{T}_\alpha \) and \( \bar{T}_\alpha^{eq} \), i.e.

1. \( \bar{T}_\alpha(x + e_\alpha \Delta t, t + \Delta t) = \bar{T}_\alpha(x, t) \)
2. \( \bar{T}_\alpha(\cdot, t + \Delta t) = \bar{T}_\alpha(\cdot, t + \Delta t) + \frac{1}{2} \Delta t \left( \bar{T}_\alpha^{eq}(\cdot, t + \Delta t) - \bar{T}_\alpha(\cdot, t + \Delta t) \right) \)

Effective viscosity: \( \nu^* = \nu + \nu_t = \frac{1}{2} \bar{T}_\alpha \frac{\Delta x}{\Delta t} \) with \( \bar{T}_\alpha = \bar{T}_\alpha + \nu_t \)

Use Smagorinsky model to evaluate \( \nu_t \), e.g., \( \nu_t = (C_m \Delta x)^2 |S| \), where

\[
|S| = \sqrt{2 \sum_{ij} S_{ij} S_{ij}}
\]

The filtered strain rate tensor \( S_{ij} = (\partial_i \bar{u}_j + \partial_j \bar{u}_i) / 2 \) can be computed as a second moment as

\[
S_{ij} = \frac{1}{2 \rho c^2 \nu_t} \sum_{\alpha} e_{\alpha h} e_{\alpha h} (\bar{T}_{\alpha} - \bar{T}_\alpha)
\]

\( \nu_t \) can be obtained as [Yu, Hou et al., 2004]

\[
\nu_t = \frac{1}{2} \left( \sqrt{\nu^2 + 18 \nu^2} (\nu_0 c^2) - C_m \Delta x |S| - \nu_t \right)
\]

Natural convection

Characterized by

\[
Ra = \frac{g \beta \Delta T H^3}{\nu D}
\]

Further LES models

Dynamic Smagorinsky model (DSMA)

\[
C_{sm}(x, t)^2 = -\frac{1}{2} \left( \frac{L_{ij} M_{ij}}{M_{ij}} \right)
\]

No van Driest damping implemented yet!

Wall-Adapting Local Eddy-viscosity model (WALE)

\[
\nu_t = (C_w \Delta x)^2 O P_{WALE}, \quad \text{where} \quad C_w = 0.5
\]

WALE turbulence time-scale

\[
O P_{WALE} = \frac{(J_{ij} J_{ij})^2}{(S_{ij} S_{ij})^2 + (J_{ij} J_{ij})^2}
\]

Effective relaxation time (see previous slide):

\[
\tau_e^* = \frac{(\nu + \nu_t) + \Delta t c^2 / 2}{C_t^2}
\]
Forced homogeneous isotropic turbulence

- Fourier representation
- Periodic boundaries, uniform mesh
- Use of external forcing term, i.e., result independent of initial conditions

Forcing:

\[ F_x = 2A \left( \frac{K_x K_z}{|K|^2} \right) G(K_x, K_y, K_z) \]
\[ F_y = -A \left( \frac{K_y K_z}{|K|^2} \right) G(K_x, K_y, K_z) \]
\[ F_z = -A \left( \frac{K_x K_y}{|K|^2} \right) G(K_x, K_y, K_z) \]

with phase

\[ G(K_x, K_y, K_z) = \sin \left( \frac{2\pi x}{L} K_x + \frac{2\pi y}{L} K_y + \frac{2\pi z}{L} K_z + \phi \right) \]

for \( 0 < K_i \leq 2 \) and \( \phi \) being a random phase value.

LES model spectra

Time-averaged energy spectra normalised by the turbulent kinetic energy \( k \) and the integral length scale \( L_{11} \) of LBM DNS and LES for two resolutions and DNS of the highest resolution for the viscosity value \( \nu = 5 \cdot 10^{-5} \)

Decaying homogeneous isotropic turbulence

- Restart DNS of 512³ resolution without forcing. Volume-averaging to 128³ cells gives DSMA and WALE initial conditions

Evolution of the turbulent kinetic energy \( k \) (left) and energy spectra at \( t = 68.72 \) (right) for DNS of 512³ against DSMA and WALE of 128³ cells resolution.
Flow field comparison

Contours of vorticity magnitude ($|\omega| = 0.18$) at $t = 4.91$ (left) and $t = 68.72$ (right) for DNS (thin blue lines) of $512^3$ against DSMA (dotted black lines) and WALE (thick red lines) of $128^3$ cells resolution.

Wind tunnel simulation of a prototype car

Fluid velocity and pressure on vehicle

Mesh adaptation

- Inflow $40 \text{ m/s}$. LES model active. Characteristic boundary conditions.
- To $t = 0.5\text{s}$ ($\sim 4$ characteristic lengths) with $31,416$ time steps on finest level in $\sim 37\text{ h}$ on 200 cores ($7389\text{ h CPU}$). Channel: $15\text{ m} \times 5\text{ m} \times 3.3\text{ m}$

- Use refinement blocks and levels (indicated by color)
Flow over a motorcycle
- Inflow 40 m/s. Bouzidi pressure boundary conditions at outflows. CSMA LES model active.
- SAMR base grid 200 × 80 × 80 cells, \( r_{1,2,3} = 2 \) yielding finest resolution of \( \Delta x = 6.25 \text{ mm} \). 23560 time steps on finest level
- Forces in AMROC-LBM are time-averaged over interval [0.5s, 1s]
- Unstructured STAR-CCM+ mesh has significantly finer as well as coarser cells

Mexico experimental turbine – 0° inflow
- Setup and measurements by Energy Research Centre of the Netherlands (ECN) and the Technical University of Denmark (DTU) [Schepers and Boorsma, 2012]
- Inflow velocity 14.93 m/s in wind tunnel of 9.5 m × 9.5 m cross section.
- Rotor diameter \( D = 4.5 \text{ m} \). Prescribed motion with 424.35 rpm: tip speed 100 m/s, \( R_c \approx 75839 \) TSR 6.70
- Simulation with three additional levels with factors 2, 2, 4. Resolution of rotor and tower \( \Delta x = 1.6 \text{ cm} \)
- 149.5 h on 120 cores Intel-Xeon (17490 h CPU) for 10s
- Data collected as average during \( t \in [5, 10] \). Load on blade 1 as it passes through \( \theta = 0^\circ \) (pointing vertically upwards), 35 rotations

Mexico experimental turbine – 30° yaw
- 157.6 h on 120 cores Intel-Xeon for 10 s (70.75 revolutions) \( \sim \sim \sim 22.25 \text{ h CPU/1M cells/revolution} \)
- \( \sim \sim \sim 12 \text{ M cells in total – level 0: 760,000, level 1:} \sim \sim \sim 1.5 \text{ M, level 2:} \sim \sim \sim 6.8 \text{ M, level 3:} \sim \sim \sim 3.0 \text{ M} \)
- For comparison [Schepers and Boorsma, 2012]:
- Wind Multi-Block Liverpool University (14 M cells): 209 h CPU/1M cells/revolution
- EllipSys3D (28.3 M cell mesh): \( \sim \sim \sim 40 \text{ h CPU/1M cells/revolution, but} \sim \sim \sim 15\% \text{ error in } F_z \) and \( T_z \) already for 0° inflow [Sørensen et al., 2014]
Single Vestas V27

- Inflow velocity $u_{\infty} = 8 \text{ m/s}$. Prescribed motion of rotor with $n_{\text{rpm}} = 33$, $r = 14.5 \text{ m}$: tip speed $46.7 \text{ m/s}$, $Re \approx 919,700$ TSR 5.84
- Simulation with three additional levels with refinement factors 2, 2, 4.
- Refinement based on vorticity and level set.
- Sampled rotor and circular regions ($r_c = 1.5 r$) every 0.034 s over $t = [8, 18] \text{s}$
- Computing 84,806 highest level iterations to $t_e = 18 \text{s}$.
- $\sim 24$ time steps for $1^\circ$ rotation

Simulation of the SWIFT array

- Three Vestas V27 turbines (geometric details prototypical). 225 kW power generation at wind speeds 14 to 25 m/s (then cut-off)
- Prescribed motion of rotor with 33 and 43 rpm. Inflow velocity 8 and 25 m/s
- TSR: 5.84 and 2.43. $Re \approx 919,700$ and 1,208,000
- Simulation domain $448 \times 240 \times 100 \text{ m}$
- Base mesh $448 \times 240 \times 100$ cells with refinement factors 2, 2.4. Resolution of rotor and tower $\Delta x = 6.25 \text{ cm}$
- 94,224 highest level iterations to $t_e = 40 \text{s}$ computed, then statistics are gathered for 10 s [Deiterding and Wood, 2016]

Vorticity – inflow at $30^\circ$, $u = 8 \text{ m/s}$, 33 rpm

- Top view in plane $z$-direction at 30 m (hub height)
- Turbine hub and inflow at $30^\circ$ yaw leads to off-axis wake impact.
- 160 cores Intel Xeon E5 2.6 GHz, $33.03 \text{ h}$ wall time for internal $[50, 60] \text{s}$ (including gathering of statistical data)
- $\sim 6.01 \text{ h per revolution (961 h CPU) } \rightarrow \sim 5.74 \text{ h CPU/1M cells/revolution}$

Levels – inflow at $30^\circ$, $u = 8 \text{ m/s}$, 33 rpm

- At 63.8 s approximately 167M cells used vs. 44 billion (factor 264)
- $\sim 6.01 \text{ h per revolution (961 h CPU) } \rightarrow \sim 5.74 \text{ h CPU/1M cells/revolution}$
Vorticity development – inflow at 0°, \( u = 8 \text{ m/s}, 33 \text{ rpm} \)

Refinement – inflow at 0°, \( u = 8 \text{ m/s}, 33 \text{ rpm} \)

Mean point values – inflow at 0°,

- Turbines located at \((0, 0, 0), (135, 0, 0), (-5.65, 80, 80, 0)\)
- Lines of 13 sensors with \( \Delta y = 5 \text{ m}, z = 37 \text{ m} \) (approx. center of rotor)
- \( u \) and \( p \) measured over \([40, 50] \) s (1472 level-0 time steps) and averaged

Lattice Boltzmann equation in mapped coordinates

Consider mapping from Cartesian to non-Cartesian coordinates

\[ \xi = \xi(x, y), \eta = \eta(x, y) \]

with

\[ \frac{\partial}{\partial x} = \frac{\partial \xi}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial \eta}{\partial \xi} \frac{\partial \xi}{\partial y} = \frac{\partial \xi}{\partial \xi} + \frac{\partial \eta}{\partial \eta} \frac{\partial \xi}{\partial y} \]

Under this transformation the convection term reads

\[ e_\alpha \cdot \nabla f_\alpha = e_{\alpha x} \frac{\partial f_\alpha}{\partial x} + e_{\alpha y} \frac{\partial f_\alpha}{\partial y} \]

\[ = e_{\alpha x} \left( \frac{\partial f_\alpha}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial f_\alpha}{\partial \eta} \frac{\partial \eta}{\partial x} \right) + e_{\alpha y} \left( \frac{\partial f_\alpha}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial f_\alpha}{\partial \eta} \frac{\partial \eta}{\partial y} \right) \]

\[ = \left( e_{\alpha x} \frac{\partial f_\alpha}{\partial \xi} + e_{\alpha y} \frac{\partial f_\alpha}{\partial \eta} \right) \frac{\partial f_\alpha}{\partial \xi} + \left( e_{\alpha x} \frac{\partial f_\alpha}{\partial \xi} + e_{\alpha y} \frac{\partial f_\alpha}{\partial \eta} \right) \frac{\partial f_\alpha}{\partial \eta} \]

and hence the lattice Boltzmann equation becomes

\[ \frac{\partial f_\alpha}{\partial t} + e_{\alpha x} \frac{\partial f_\alpha}{\partial \xi} + e_{\alpha y} \frac{\partial f_\alpha}{\partial \eta} = -\frac{1}{\tau} (f_\alpha - f_\alpha^e) . \]
Scheme construction

Currently using the explicit 4th-order Runge-Kutta scheme

\[ f_{i}^{n+1} = f_{i}^{n} + \frac{\Delta t}{4} R_{i}, \]
\[ f_{i}^{n+2} = f_{i}^{n} + \frac{\Delta t}{3} R_{i}, \]
\[ f_{i}^{n+3} = f_{i}^{n} + \frac{\Delta t}{2} R_{i}, \]
\[ f_{i}^{n+4} = f_{i}^{n} + \Delta t R_{i}. \]

with

\[ R_{i} = -\left( \frac{\bar{C}_{i}^{(1)} - C_{i}^{(0)}}{2\Delta t} + \bar{C}_{i}^{(2)} - C_{i}^{(1)} \right) \]

for the solution, 2nd-order central differences to approximate derivatives. A 4th-order dissipation term

\[ D = -\epsilon \left( (\Delta \xi)^{4} \frac{\partial f_{i}}{\partial \xi} + (\Delta \eta)^{4} \frac{\partial f_{i}}{\partial \eta} \right) \]

is added for stabilization [Hejranfar and Hajihassanpour, 2017]. Prototype implementation is presently on finite difference meshes!

2d cylinder study – unsteady flow case

Conclusions – subsonic aerodynamics with LBM

- Cartesian LBM is a very efficient low-dissipation method for subsonic aerodynamic simulation and especially suitable for DNS and LES
- Cartesian CFD with block-based AMR is faster than cell-cased AMR and tailored for modern massively parallel computer systems
- Fast dynamic mesh adaptation in AMROC makes FSI problems with complex motion easily accessible. Time-explicit approach leads to very tight coupling
- For high Reynolds number flows around complex bodies an LES turbulence model is vital for stability (so are higher-order in- and outflow boundary conditions)
- Currently validating and extending (dynamic) Smagorinsky with wall-near damping and WALE model for realistic problems
- Turbulent wall function boundary condition model under development
- Accurate simulation of thin, wall-resolved boundary layers is dramatically more efficient with the non-Cartesian LBM approach, despite the availability of AMR in AMROC
  - Develop non-Cartesian version of AMROC-LBM as near-term goal
  - Chimera technique within AMROC-LBM might be long-term goal
References I


References II


Motion solver

Based on the Newton-Euler method solution of dynamics equation of kinetic chains

\[ \begin{aligned}
F &= \left( \frac{m}{m_c^\alpha \times I_{cm}^\alpha} \right) a_p + \left( \frac{m}{m_c^\alpha \times I_{cm}^\alpha} \right) \omega \\
\tau &= \left( \frac{\omega \times \tau_{prescribed}}{m_c^\alpha} \right) \times c
\end{aligned} \]

Where \( m \) is the mass of the body, \( I_{cm} \) is the 4x4 homogeneous identity matrix, \( a_p \) is the acceleration of link frame with origin at \( p \) in the preceding link’s frame, \( I_{cm} \) is the moment of inertia about the center of mass, \( \omega \) is the angular velocity of the body, \( \alpha \) is the angular acceleration of the body, \( c \) is the location of the body’s center of mass, and \( [c] \times \) and \( [\omega] \times \) denote skew-symmetric cross product matrices.

Here, we additionally define the total force and torque acting on a body,

\[ \begin{aligned}
F &= (F_{FSI} + F_{prescribed}) \cdot C_{xyz} \\
\tau &= (\tau_{FSI} + \tau_{prescribed}) \cdot C_{\alpha\beta\gamma}
\end{aligned} \] respectively.

Where \( C_{xyz} \) and \( C_{\alpha\beta\gamma} \) are the translational and rotational constraints, respectively.