INRIA

Thirty Month Review Meeting
23rd- 24th November 2017, INRIA Bordeaux Sud-Ouest

Presented by Andrea Ferrero
Development of high-fidelity aero-elastic models

Fluid dynamics

• **Cartesian immersed boundary code**
  Used for preliminary testing. In the last months:
  • Improvement of the scalability on large scale machines (BlueGene, Occigen)
  • Validation of the implementation of the Vreman LES turbulent model

• **Adaptive octree immersed boundary code**
  • Devoted to full scale simulations
  • Under development during the PhD of Claire Taymans as a collaboration between VALEOL and INRIA
  • Discretisation and parallelisation based on the BitPit library developed by Optimad
Parallel scalability on the incompressible NS cartesian solver

The large scale simulations are carried out on two different machines:

• **Occigen**
  Section 1: 50544 cores on 2106 nodes
  Each node has 2 Intel 12-cores Haswell CPUs @ 2.6 GHz
  Each node can use 64/128 GB of RAM (2.67-5.33 GB/core)

  Section 2: 35280 cores on 1260 nodes
  Each node has 2 Intel 14-cores Broadwell CPUs @ 2.6 GHz
  Each node can use 64 GB of RAM (2.67 GB/core)

• **Turing**
  IBM Blue gene/Q, 98304 cores PowerPC A2 @ 1.6 GHz
  RAM 1 GB/core
Parallel scalability on the incompressible NS cartesian solver

The parallel profiling of the code was performed by using Scalasca in order to identify the main bottlenecks and unbalancing.

- Reduction of the number of collective operations
- Balanced update of the level-set function: compute it everywhere at low cost by keeping into the memory the indexes of the closest lagrangian marker
- The code instrumented with Scalasca is 30-50% slower
Parallel scalability on the incompressible NS cartesian solver

Strong scalability results on Occigen

- 96 millions of points
- Measure the time required to perform 10 time steps
- 1 process per core
- 2 different exit conditions for Poisson (100 iterations or relative tolerance 10^{-4})
- Reference time for scalability with 96 cores
- Measure with 96, 192, 384, 768, 1536 cores
- 75% and 82% of efficiency at 768 cores with the two exit conditions
- The cost is dominated by the Poisson solver (Time Poisson/time prediction =10)
Parallel scalability on the incompressible NS cartesian solver

Strong scalability results on Turing

- 96 millions of points
- Measure the time required to perform 10 time steps
- 1 process per core
- 2 different exit conditions for Poisson (100 iterations or relative tolerance $10^{-4}$)
- Reference time for scalability with 1024 cores
- Measure with 1024, 2048, 4096, 8192 cores
- Almost the same results with different exit conditions for the Poisson solver
- Most of the CPU time in the prediction step and not in the Poisson solver
Parallel scalability on the incompressible NS cartesian solver

Using more than 1 MPI process per core on Turing

<table>
<thead>
<tr>
<th>MPI proc. per core</th>
<th>Comp. Time [s]</th>
<th>Speed up</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>687.30</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td>438.32</td>
<td>1.57</td>
<td>0.78</td>
</tr>
<tr>
<td>4</td>
<td>405.11</td>
<td>1.70</td>
<td>0.42</td>
</tr>
</tbody>
</table>

Test done with 64 cores

<table>
<thead>
<tr>
<th>MPI proc. per core</th>
<th>Comp. Time [s]</th>
<th>Speed up</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>105.46</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td>79.83</td>
<td>1.32</td>
<td>0.66</td>
</tr>
<tr>
<td>4</td>
<td>73.96</td>
<td>1.43</td>
<td>0.36</td>
</tr>
</tbody>
</table>

Test done with 512 cores
Parallel scalability on the incompressible NS cartesian solver

Comparison between Occigen and Turing in terms of CPU time vs number of cores

- Strong scalability with 96 millions of points
- Test done with a Poisson exit condition based on a fixed number of iterations
- 1 process per core on both machines
- There is a factor 20 in the computational time
Parallel scalability on the incompressible NS cartesian solver

Comparison between Occigen and Turing in terms of CPU time vs number of cores

- Turing’s CPU works at low frequency for reducing electric power consumption
- One hour of CPU on Occigen is 4 times more expensive than one hour on Turing
- We are using a full distributed memory parallelization: especially on Turing, mixed shared/distributed memory approaches are recommended
Validation of the implementation of the Vreman LES model

• The section of the wind turbine blade is a circular cylinder close to the nacelle
  • The wind turbine tower has a circular section

• Need to simulate correctly separated turbulent flows around circular sections

The incompressible flow around a circular cylinder at Reynolds=3900 is studied and the results are compared with the experiments (drag coefficient and wake profile). The flow shows laminar separation and transition to turbulence in the wake.

This Reynolds number is significantly lower than the maximum Reynolds number evaluated on the wind turbine blade’s chord in the full scale configuration (10^6).

The full scale configuration will be studied by means of the Octree code under development in collaboration with Valeol.
Flow around a circular cylinder at Re=3900

- Immersed boundary code (penalisation + linear correction of velocity close to the solid wall)
  - Circular cylinder with diameter D
  - Spanwise extension of the domain = 4D
    - Inlet at 8D from the cylinder
    - Outlet at 16D from the cylinder
    - Lateral boundaries at 8D

- BCs for velocity: Dirichlet at inlet, periodic in the axial direction, homogeneous Neumann on lateral boundaries
- BCs for pressure: periodic in the axial direction and homogeneous Neumann everywhere else

- Grid 100x800x1200 = 96 millions of points (50 points along D)
- Simulation on Turing with 4096 cores (8192 MPI processes).
- 20 wall clock hours are required to simulate 30 convective times
Flow around a circular cylinder at Re=3900

Magnitude of vorticity (10 levels)
Flow around a circular cylinder at $Re=3900$

Vreman eddy viscosity
Flow around a circular cylinder at Re=3900

Comparison of the predicted drag coefficient with data in the literature

- An initial simulation of 100 convective times is performed to remove the initial transient
- The simulation is restarted with this solution and the Cd is computed on a period of 30 convective times

<table>
<thead>
<tr>
<th>Source</th>
<th>Average Cd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Present work</td>
<td>0.98</td>
</tr>
<tr>
<td>LES of Hansen and Long (2002)</td>
<td>1.31</td>
</tr>
<tr>
<td>LES of Snyder and Degrée (2003)</td>
<td>1.09</td>
</tr>
<tr>
<td>LES of Kravchenko and Moin (2000)</td>
<td>1.04</td>
</tr>
<tr>
<td>LES (C2) of Breuer (1998)</td>
<td>1.10</td>
</tr>
<tr>
<td>LES (C3) of Breuer (1998)</td>
<td>1.07</td>
</tr>
<tr>
<td>LES of Blackburn and Schmidt (2001)</td>
<td>1.01</td>
</tr>
<tr>
<td>LES (Nc = 10) of Franke and Frank (2002)</td>
<td>1.01</td>
</tr>
<tr>
<td>LES (Nc = 42) of Franke and Frank (2002)</td>
<td>0.98</td>
</tr>
<tr>
<td>LES DM of (Park et al. 2006)</td>
<td>1.04</td>
</tr>
<tr>
<td>LES DM1 of (You and Moin 2006)</td>
<td>1.01</td>
</tr>
<tr>
<td>LES WALE of (Ouvrard et al. 2010)</td>
<td>1.02</td>
</tr>
<tr>
<td>VMS-LES WALE of (Ouvrard et al. 2010)</td>
<td>0.94</td>
</tr>
</tbody>
</table>

Table 1: Results from the literature (data from N. Rajan et al. / JAFM, Vol. 9, No. 3, pp. 1421-1435, 2016.)
Flow around a circular cylinder at Re=3900

Comparison of the wake profile at 3 different locations with exp. data in the literature

- Two sets of exp. data are considered:
  - Parraud et al. 2008 (Exp. A)
  - Lourenco and Shih 1994 (Exp. B)

- The experiments gave very different results

- After the initial simulation of 100 convective times, the average profile is computed over 30 convective times

- The averaging period seems to be not sufficiently long because the profiles are not exactly symmetric
Flow around a circular cylinder at Re=3900

The comparison with the data of Parnaudeau et al. 2008 is quite good
Flow around a circular cylinder at Re=140000

Testing the code with strongly underresolved structures.
A simulation was performed with the same mesh used for Re=3900

Magnitude of vorticity (10 levels)
Flow around a circular cylinder at $\text{Re}=140000$

Some oscillations appear in the velocity field: they disappear when the LES eddy viscosity is increased or when the mesh is refined.

Magnitude of vorticity (100 levels)
Flow around a circular cylinder at $Re=140000$

Normal and streamwise average velocity distribution: comparison with the experiment

Preliminary results: average performed only on 5 convective times
Finite element blade model

• The structural behaviour of the blade is described by a linear beam model with flexion, torsion and axial deformation

• The CAD of the blade was processed in order to generate a valid level set for the immersed boundary method

• The structural code was written as a C library and was linked to the Octree CFD code which is under development in collaboration with Valeol: the two way coupling between the fluid and the structure was implemented

• Centrifugal and Coriolis terms were implemented in the structural code and in the Octree CFD code
A zonal POD approach for Reduced Order Modelling
Work done in collaboration with Optimad

• The POD model is used to define the boundary conditions for a reduced CFD domain
• The non linearities which are difficult to describe by POD and the phenomena not included in the database are directly taken into account by the CFD solver

\[ U_{ROM}(x, t) = U_{avg}(x) + \sum_{i=1}^{N} a_i(t)\Phi_i(x) \]

The coefficients of the POD expansion \((a_i)\) are computed by minimizing the distance between the CFD solution and the POD expansion in the overlapping region at each time step: \( min \left( \| U_{CFD} - U_{ROM} \|_{\Omega_o} \right) \)

Introduction of forcing terms for gust simulations:

\[ U_{ROM}(x, t) = U_{avg}(x) + U_g(x, t) + \sum_{i=1}^{N} a_i(t)\Phi_i(x) \]

Approximated forcing term: POD introduces corrections

Updates after previous meeting:
• Convergence test w.r.t. the sampling of the database
A zonal POD approach for Reduced Order Modelling
Convergence test w.r.t. the sampling in the database

• Consider the prediction error indicator map obtained by the leave-one-out strategy
• Increase the sampling of the database in the parameter space
• The error in the map decreases when the database is improved

• This brings two consequences:
  • **If you fix a threshold on the prediction error**, you can reduce the size of the domain in the zonal simulation (the more you pay in the offline sampling, the cheaper will be the online zonal simulation)
  • **If you fix the size of the domain in the zonal simulation**, the prediction error decreases when the database is improved
A zonal POD approach for Reduced Order Modelling

Fix a threshold $\sigma_R$ on the prediction error

The size of the domain (and the cost) of the zonal simulation decreases when the database is enriched.
A zonal POD approach for Reduced Order Modelling
Fix the size of the domain in the zonal simulation

The prediction error (evaluated a-posteriori) is reduced when the database is improved and the size of the domain in the zonal simulation is fixed.

Without POD modes

Database 3x3

Database 5x5
The research leading to this work has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement number 636053.