CDP†
Large Eddy Simulation

Eric Darve
Frank Ham

† named after late Charles David Pierce (1969-2002)
## LES Background

<table>
<thead>
<tr>
<th>method</th>
<th>turbulence</th>
<th>solved velocity</th>
</tr>
</thead>
<tbody>
<tr>
<td>RANS</td>
<td>completely modeled</td>
<td>$\langle u_i \rangle (x)$</td>
</tr>
<tr>
<td>LES</td>
<td>$E$ vs $k$</td>
<td>$\bar{u}_i (x, t)$</td>
</tr>
<tr>
<td>DNS</td>
<td>$E$ vs $k$</td>
<td>$u_i (x, t)$</td>
</tr>
</tbody>
</table>
Mathematical Model

- Navier-Stokes equations: conservation of momentum

\[
\frac{\partial u_i}{\partial t} + \frac{u_j u_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j} \left( \nu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right)
\]

- Continuity equation for incompressible flows:

\[
\frac{\partial u_i}{\partial x_i} = 0
\]
More complex models are actually used in CDP

- Low Mach number compressible reacting flows.
- Combustion.
- Lagrangian particles.
- ...
Filtering

- We don’t want to resolve all the scales in the flow. All scales smaller than $\Delta$ are modeled.
- Filtered NS:

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_j \bar{u}_i}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j} \left( \nu \left( \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right)$$

- Convective term is difficult to calculate: we are going to model it!
Closure problem

\[
\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial u_j \bar{u}_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j} \left( \nu \left( \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right) - \frac{\partial \tau_{ij}}{\partial x_j}
\]

\[
\tau_{ij} = u_j \bar{u}_i - u_j \bar{u}_i
\]

- Closure problem: can we model the stress term \( \tau_{ij} \)?
- Simplest model is the Smagorinsky model:

\[
\tau_{ij} = 2C \Delta^2 \left| \bar{S} \right| \bar{S}_{ij}
\]

\[
\bar{S}_{ij} = \frac{1}{2} \left( \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right)
\]

“Stress is proportional to strain”
End result

• Final equation often written as an additional viscosity term:

\[
\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial \overline{u}_j \overline{u}_i}{\partial x_j} = - \frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j} \left( (\nu + \nu_{sgs}) \left( \frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i} \right) \right)
\]

\[
\nu_{sgs} = C \Delta^2 |\overline{S}|
\]

• What value should be used for the constant \(C\)? You can either use a tuned value based on homogeneous turbulence (\(C = 0.15\)), or a more appropriate case (channel, etc.).

• The dynamic procedure uses the fluctuations in the resolved velocity to estimate \(C\) locally based on some scale similarity assumptions.
How do we solve those equations: collocated fractional step
Idea: we need to satisfy both the conservation of momentum and the continuity equations.
Solution:
  – Advance in time using conservation of momentum
  – Satisfy the continuity equation for solving a Poisson equation for the pressure.
Advance in time using momentum equation

$$\frac{\hat{u}_i - u_i^n}{\Delta t} + \text{Conv}^{n+\frac{1}{2}}_i = -\frac{\delta p^{n-\frac{1}{2}}}{\delta x_i} + \text{Visc}^{n+\frac{1}{2}}_i$$

- Solution at $n+1/2$ usually obtained by taking the average of solution at $n$ and $n+1$.
- Non linear equation in $\hat{u}_i$.
- Fast solution procedure based on non linear Gauss-Seidel with successive over-relaxation.
Enforce continuity equation

- Add back the pressure at step $n-1/2$:
  \[
  \frac{u_i^* - \tilde{u}_i}{\Delta t} = + \frac{\delta p^{n-1/2}}{\delta x_i}
  \]

- At this point we just need to solve for:
  \[
  \frac{u_i^{n+1} - u_i^*}{\Delta t} = - \frac{\delta p^{n+1/2}}{\delta x_i}
  \]

- Assume we have solved the previous equation. Then add the last 3 equations to get:
  \[
  \frac{u_i^{n+1} - u_i^n}{\Delta t} + \text{Conv}_{i}^{n+1/2} = - \frac{\delta p^{n+1/2}}{\delta x_i} + \text{Visc}_{i}^{n+1/2}
  \]
Poisson equation for the pressure

• How do we solve for $u_i^{n+1}$? Take divergence of equation. Since

$$\nabla u_i^{n+1} = 0$$

we get an equation for the pressure only.
• Once we have the pressure we can update $u_i^{n+1}$.
• Poisson equation for pressure:

$$\sum_f \frac{u_f^*}{\Delta t} = \sum_f \frac{\delta p^{n+\frac{1}{2}}}{\delta n}$$

Solved using Multigrid.
• Un-structured grid.
Data structure

- Ordering of nodes is critical to get good performance.

Types of boundary conditions
CV ordering

1. Inner nodes
2. Inner boundary nodes
3. Periodic boundary nodes
4. Assigned boundary nodes
5. Ghost nodes

- Fast access to elements and fast operations.
- Use of Fortran90 vector operations.
Face and node ordering
Connectivity information

Two types of connectivity in CDP:

1. When number of elements is fixed, a direct “array” format is used. For example, faces always have 2 neighboring CVs.
   
do ifa = 1, gp%nfa_ib
       icv1 = gp%cvofa(1,ifa)
       icv2 = gp%cvofa(2,ifa)
       write(*,*) 'face ',ifa,' has cvs ',icv1,icv2
   
end do

2. When number of elements vary, a compact row storage is used, i.e. all the neighboring CVs of a given CV.
   
do icv = 1, gp%ncv_ib
       foc_f = gp%faocv_i(icv)
       foc_l = gp%faocv_i(icv+1)-1
       do foc = foc_f, foc_l
           ifa = gp%faocv_v(foc)
           write(*,*) 'cv ',icv,' has face ',ifa
        
       end do
   
end do
Parallel Implementation

Latest version of the unstructured LES code focused on scalability:

- Parallel pre-processing
- Dynamic load balancing

Facilitated first Large Scale Simulation on 1024 processors and 100M CVs.
Parallel preprocessing

Steps:
• Sub-domain mesh generation
• Parallel reassembly of global mesh using fast oct-tree searches
• Parallel partitioning (ParMETIS)
• Parallel writing of partitions (MPI-I/O)

Results:
Preprocessing of 100M CV mesh in < 3 hrs
CDP-α Scalability

Scalability of the 35M control volume full combustor simulation with AMG on Frost:
<table>
<thead>
<tr>
<th>Table 1.3: Some computer science indices for CDP-α</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Total memory requirement</strong></td>
</tr>
<tr>
<td>- with PCG solver for pressure</td>
</tr>
<tr>
<td>- with AMG solver for pressure</td>
</tr>
<tr>
<td><strong>I/O (MPI-2):</strong></td>
</tr>
<tr>
<td>- input partition files</td>
</tr>
<tr>
<td>- input restart or output result files</td>
</tr>
<tr>
<td><strong>Floating point performance (Frost):</strong></td>
</tr>
<tr>
<td>- sustained Megaflops per processor</td>
</tr>
<tr>
<td>- percentage of peak</td>
</tr>
<tr>
<td>- flops to memory references</td>
</tr>
<tr>
<td><strong>Sample wall clock time (35 M cv on 480 processors of Frost, cold flow):</strong></td>
</tr>
<tr>
<td>- startup (read partition &amp; restart files)</td>
</tr>
<tr>
<td>- 20,000 time steps at 11 s/step</td>
</tr>
<tr>
<td>- writing result/restart files</td>
</tr>
<tr>
<td>- cooperative writing 2D plane data file</td>
</tr>
</tbody>
</table>
Particle-Mesh Load Balancing

**Challenge:** Geometric localization of particles can create load imbalance

**A Solution:** Parallel multi-constraint repartitioning
Test problem: balancing \#CV’s only

Dual constraint: \#CV’s and \#Particles

- Speedup: 4x
- Increase in edge-cut: 30%
Scalable Multigrid Solvers

Multigrid solvers are being developed and/or integrated into CDP-α to solve the Poisson system and other transport equations:

<table>
<thead>
<tr>
<th>Poisson Solver</th>
<th>Cost</th>
<th>Scalable</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCG</td>
<td>~2000 N</td>
<td>No</td>
<td>Existing solver</td>
</tr>
<tr>
<td>Algebraic MG</td>
<td>~200 N</td>
<td>Yes</td>
<td>LLNL AMG in both old and new codes</td>
</tr>
<tr>
<td>Geometric MG</td>
<td>~20-50 N*</td>
<td>Yes</td>
<td>Under development</td>
</tr>
</tbody>
</table>

*estimated cost based on small cold flow simulations
LLNL’s CASC group AMG solver has now replaced PCG for the Poisson system in both old and new codes:

**CDP, multi-physics**
(2M CV on 80 proc: 25,000 CV/proc)

- Over 2x speedup

**CDP-α, cold flow**
(35M CV on 480 proc: 73,000 CV/proc)

- Over 4x speedup
Algebraic Multigrid

**Challenge:** memory requirement can limit maximum problem size.

<table>
<thead>
<tr>
<th>Operator Matrix Information:</th>
</tr>
</thead>
<tbody>
<tr>
<td>lev</td>
</tr>
<tr>
<td>-----</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>12</td>
</tr>
<tr>
<td>13</td>
</tr>
<tr>
<td>14</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>16</td>
</tr>
<tr>
<td>17</td>
</tr>
</tbody>
</table>
Solver Plan:

1. Tune AMG parameters (e.g. increase coarsening rate) to allow solution of larger problems without significantly degrading convergence rate.

2. Develop custom geometric multigrid solver
   + 5-10x faster than AMG
   + Much lower memory requirement
   - Increased sensitivity to geometric quality of the coarse grids
CDP-α summary

• Currently being phased in as ASCI production code
• Already used exclusively for RANS/LES integration simulations.

Basis for a growing body of research:
  • Advanced combustion models (G-equation)
  • Primary atomization
  • Local mesh refinement and immersed boundary
  • Multiscale LES method of Hughes et al.
  • Porting of algorithms to Brook/Merrimac
  • Transfer to industry (P&W) - drives combustion capabilities, flow interrogation, turn-around
  • …