Implementation of Anisotropic Mesh Refinement in OpenFOAM

Jonas Karlsson

Hydrodynamics Group
Department of Shipping and Marine Technology
Chalmers University of Technology
Göteborg, Sweden

November 13, 2012
About this work

- MSc. thesis in Computer Science
- Employee at Shipping and Marine Technology
Adaptive Mesh Refinement (AMR)

- Refine (split) cells with greatest error
- Anisotropic refinement. Refine in one direction
OpenFOAM

- Supports isotropic (not anisotropic) AMR
- Provides:
  - Face remover (for unrefinement)
  - Synchronization (sending/receiving parts of mesh)
  - Topology tools (polyTopoChange for adding/removing points, cells and faces)
Application components

**Refinement-engine**
Selects cells to refine

**Mesh-cutter**
Executes topological changes for anisotropic split

**Refinement-tree**
Stores the refinement changes

Note: Only for hexahedra.
Refinement-tree

- Enables unrefinement
- Criteria – store the refinement-changes and be redistributable.
Refinement-tree example
Refinement-tree example
Mesh Cutter

- Used for cutting/splitting cells anisotropically
- For every cell store its corner points
- Split between predetermined pairs of points
Mesh Cutter

- Used for cutting/splitting cells anisotropically
- For every cell store its corner points
- Split between predetermined pairs of points
Mesh Cutter

- Used for cutting/splitting cells anisotropically
- For every cell store its corner points
- Split between predetermined pairs of points
Mesh Cutter (Consistency)

Hanging node:

![Diagram of a mesh with a hanging node]
Mesh Cutter (Consistency)

Hanging node:

"Two-to-one" rule.

- 2D: every side of a cell has at most two edges,
- 3D: every side of a cell has at most two faces.
Mesh Cutter (Consistency)

Hanging node:

"Two-to-one" rule.
- 2D: every side of a cell has at most two edges,
- 3D: every side of a cell has at most two faces.

Refered to as a *Consistent mesh*. 
Mesh Cutter (Splits)

Allowed:
Mesh Cutter (Splits)

Allowed:

Disallowed:
Mesh Cutter (Splits)

Constraint:
For a face to be splittable it needs to contain four corner points from its neighbour
Mesh Cutter (Spreading)

- Original domain
- Shaded cell marked for refinement
- Spreading algorithm marks big cell
- Refining all cells not restricted
- Repeat
Mesh Cutter (Spreading)

- Original domain
- **Shaded cell marked for refinement**
- Spreading algorithm marks big cell
- Refining all cells not restricted
- Repeat
Mesh Cutter (Spreading)

- Original domain
- Shaded cell marked for refinement
- Spreading algorithm marks big cell
- Refining all cells not restricted
- Repeat
Mesh Cutter (Spreading)

- Original domain
- Shaded cell marked for refinement
- Spreading algorithm marks big cell
- Refining all cells not restricted
- Repeat
Mesh Cutter (Spreading)

- Original domain
- Shaded cell marked for refinement
- Spreading algorithm marks big cell
- Refining all cells not restricted

Repeat
Refinement-engine flow graph

Refine?

Y

(1) Select cells for refinement
(2) Cut cells with mesh-cutter
(3) Update refinement-tree

N

Solving

Distribute?

Y

(1) Partition mesh
(2) Distribute corner points
(3) Distribute refinement-tree

N
Refinement-engine collaboration diagram

- refinementTree
- refTree
- dynamicFvMesh
- hexRef2
- meshCutter
- dynamicRefineFvMeshHexRef2
Spreading example
Spreading example
Spreading example
Spreading example
Spreading example
Spreading example
Parallel example
Parallel example
Parallel example
Parallel example
Parallel example
Parallel example
Parallel example
Parallel example
Parallel example
Parallel example
Parallel example
Parallel example
Parallel example
Parallel example
Questions and comments