FoamedOver: A Dynamic Overset Grid Implementation in OpenFOAM

David Boger       Eric Paterson       Ralph Noack

Penn State University, Applied Research Lab
State College, PA USA

10th Symposium on Overset Composite Grids
and Solution Technology
20 - 23 September 2010
NASA Ames Research Center
Moffet Field, California USA

DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited.
Acknowledgements

- Development of foamedOver has been sponsored by: CREATE-Ships, Myles Hurwitz, Project Manager
- Supercomputing resources have been provided by: DOD High–Performance Computing Modernization Program (HPCMP) and the Army Research Laboratory (ARL) DOD Supercomputing Resource Center (DSRC)
Outline

1. Objectives

2. Background
   - OpenFOAM
   - Suggar++
   - DiRTlib

3. foamedOver

4. Code Verification and Example Applications
   - Static Meshes
   - Dynamic Meshes

5. Validation

6. Summary
Objectives

- Develop a library that provides dynamic–overset mesh capability for NavyFOAM (and other naval applications/solvers).
  - NavyFOAM is an OpenFOAM–based incompressible, multiphase, free–surface hydrodynamics solver.
- Adhere to OOP standards of OpenFOAM
- Library design goals:
  - Build on top of Suggar++ and DiRTlib.
  - Instrumentation of solvers with overset capability should be simple.
  - Dynamic–mesh capability for a variety of motion types: table look–up; analytical functions; 6DOF equations of motion.
  - Utilization of the run–time selection mechanism for motion type and algebraic solvers.
  - Simulation control through dictionaries.
Objectives

Intended Use

- Intended use is naval hydrodynamics: surface ships; submarines; weapons systems.
- CREATE-Ships: Develop CFD tools which can impact design and acquisition
  - Rapid Design Capability: Automation of CFD Process
  - Hull–form optimization and ship motions in ocean waves
OpenFOAM: Executive Summary

Overview

- OpenFOAM is a free-to-use Open Source numerical simulation software with extensive CFD and multi-physics capabilities.
- Free-to-use means using the software without paying for license and support, including massively parallel computers.
- Software under active development, capabilities mirror those of commercial CFD.
- Substantial user base in industry, research labs, and universities.
- Possibility of extension to non-traditional, complex or coupled physics.
- Physics model implementation through equation mimicking.
OpenFOAM: Executive Summary

Main components

- Discretization: General-polyhedral finite–volume method. Numerous schemes are available.
- Lagrangian particle tracking.
- Finite Area Method: 2-D FVM on curved surface in 3-D
- Libraries for turbulence modeling (RANS, DES, LES); thermophysical properties; combustion; . . .
- Automatic mesh motion, support for topological changes
- Parallelism via domain decomposition. Methods include metis, scotch, simple, and hierarchial
OpenFOAM: Executive Summary

Equation Mimicking

- Flexible handling of arbitrary equations sets
- Natural language of continuum mechanics: partial differential equations
- Example: turbulence kinetic energy equation

\[
\frac{\partial k}{\partial t} + \nabla \cdot (Uk) - \nabla \cdot [(\nu + \nu_t) \nabla k] = \nu_t \left[ \frac{1}{2} \left( \nabla U + \nabla U^T \right) \right]^2 - \frac{\epsilon_0}{k_0} k
\]

- Objective: Represent differential equations in their natural language

```cpp
solve
{
    fvm::ddt(k)
    + fvm::div(phi, k)
    - fvm::laplacian(nu()+nut(), k)
    == nut*magSqr(symm(fvc::grad(U)))
    - fvm::Sp(epsilon/k, k)
};
```

- Correspondence between implementation and the original equation is clear
Applications

- Libraries encapsulate interchangeable models with run-time selection
- New models provide functionality by adhering to a common interface
- Custom top-level solvers written for a class of physics, e.g. compressible combusting LES or VOF free-surface flow
- Code clarity is paramount: existing solvers act as examples for further development or customization
Utilities

- Pre-processing, data manipulation, mesh-to-mesh mapping etc.
- Mesh import and export, mesh generation and manipulation
- Parallel processing tools: decomposition and reconstruction
- Post processor hook-up (Paraview) and data export (EnSight, Tecplot, Fieldview)
- Solution analysis, PyFoam

Customized Data Extraction and Analysis

- User-defined on-the-fly data extraction: function objects

This is just a “standard set”: *Users write their own applications using the library*
PSU OpenFOAM community has significantly grown to include faculty and students from:

- Applied Research Laboratory
- Mechanical Engineering
- Bioengineering
- Nuclear Engineering
- Aerospace Engineering
- Research Computing and Cyberinfrastructure

Application areas include:

- Naval hydrodynamics
- Fluid–structure interaction
- Wind– and hydro–turbines
- Atmospheric turbulence and LES
- Explosives detection
- Cardio–vascular hemodynamics and blood pumps
- Electron beam-physical vapor deposition
- Nuclear–reactor dynamics
- Rotorcraft icing
- CFD education
Suggar++

- Overset grid assembly software
- Performs hole-cutting, donor search, overlap minimization
- Static or dynamic (moving body) assemblies
- Structured and **unstructured** grids
- Node-centered and **cell-centered** flow solvers
- Stand-alone executable (static) or library calls for dynamic grids (libsuggar.so)

---

DiRTlib is a solver-neutral library that simplifies the addition of an overset capability to a flow solver by encapsulating the required operations:

- Acquire interpolation stencils via file I/O or direct communication with libSuggar++
  
- Provides higher-level methods to transfer field data from donors to receptors and interpolate
  
- Provides lower-level access to donor member indices and weights to help build implicit global matrix

FoamedOver is an interface between OpenFOAM and other specialized libraries:

- **DiRTlib** – Library to facilitate the addition of overset capability to any flow solver
- **Suggar++** – Overset grid assembly software
- **PETSc** – Library of data structures and routines for the parallel solution of large systems of linear and non-linear equations
FoamedOver is a collection of custom classes, solvers, and applications.
FoamedOver is a **stand-alone library** that provides a dynamic overset grid capability to any OpenFOAM solver

- No changes are required to the OpenFOAM library itself
- Any OpenFOAM solver is made overset-capable by the insertion of a half-dozen lines of code . . .
- . . . and the use of custom run-time selectable objects

```cpp
#include "oversetObject.H"
#include "createOverset.H"

U *= cellMask;
overset.updateFringeValues(U);

dynamicFvMesh oversetFvMesh;
mesh.update();
```
Run–Time Selectable Objects

- `oversetPETScSolver (IduMatrix::solver)`
- `oversetSmoothSolver (IduMatrix::solver)`
- `oversetFvMesh (dynamicFvMesh)`
Code verification on static meshes
Potential Foam

- cylinder
- steady
- potential flow
Overset dictionary for static meshes
constant/oversetDict

// ************************************************************************* //

isOverset yes;
readFromDisk yes;
isDynamicOverset no;
dciFileName "SUGGAR/output++.dci";
clipInterpolation yes;
bodies ();

// ************************************************************************* //
Code verification on static meshes

interFoam

- submerged hydrofoil\textsuperscript{a}
- steady
- incompressible multiphase

Code verification on static meshes
interFoam

- extension of damBreak tutorial
- unsteady
- incompressible multiphase

figures/damBreak.avi
Code verification on static meshes
compressibleInterFoam

- extension of depthCharge tutorial
- unsteady
- compressible multiphase
Code verification on dynamic meshes

Unsteady potential flow

- cylinder
- **prescribed** mesh motion
- unsteady
- unsteady potential flow is a series of steady solutions

figures/plungingCylinder-p.jpg
Dictionaries for dynamic mesh motion
constant/dynamicMeshDict and constant/oversetDict

%-----------------------------------------------------------------
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dynamicFvMesh oversetFvMesh;
// ************************************************************************* //
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
isOverset yes;
readFromDisk no;
isDynamicOverset yes;
useLibSuggar yes;
motionType multibody;

bodies
{
  projectile
  {
    oversetMotion specifiedTrajectoryOversetMotion;
    translationTable "constant/translations";
    rotationTable "constant/rotations";
  }
};
Code verification on dynamic meshes
icoDyMFoam

- pitching foil
- **prescribed** mesh motion
- incompressible, laminar, multiphase flow

figures/animatePitchingFoil.avi
Code verification on dynamic meshes
interDyMFoam with 6DOF

- falling cylinder
- **6DOF** mesh motion
- incompressible, multiphase, laminar flow
- Compare to simplified analytical solution for free-falling body with constant CD

figures/6dof_sphere_vy_mesh.png
Overset dictionary for dynamic meshes with 6DOF motion

constant/oversetDict

```plaintext
// ************************************************************************* //

isOverset   yes;
readFromDisk no;
isDynamicOverset yes;
useLibSuggar yes;
motionType   multibody;

bodies
{
  ship
  {
    oversetMotion sixDofOversetMotion;
  }
};

// ************************************************************************* //
```
Body dictionary for sixDofOversetMotion

```plaintext
0/ship

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

mass m [1 0 0 0 0 0 0] 0.500;
momentOfInertia J [1 2 0 0 0 0 0] (10 10 10);
equilibriumPosition x [0 1 0 0 0 0 0] (0 0.1144 0);
linearSpring k [1 0 -2 0 0 0 0] (0 0 0);
linearDamping d [1 0 -1 0 0 0 0] (0 0 0);

// Xabs = Xeq + Xrel
Xrel Xrel [0 1 0 0 0 0 0] (0 0 0);
U U [0 1 -1 0 0 0 0] (0 0 0);
Uold Uold [0 1 -1 0 0 0 0] (0 0 0);

rotationVector (0 0 1);
rotationAngle rotationAngle [0 0 0 0 0 0 0] 0;
omega rotUrel [0 0 -1 0 0 0 0] (0 0 0);

force f [1 1 -2 0 0 0 0] (0 0 0);
moment m [1 2 -2 0 0 0 0] (0 0 0);

forceRelative fRel [1 1 -2 0 0 0 0] (0 0 0);
momentRelative mRel [1 2 -2 0 0 0 0] (0 0 0);

// ************************************************************************* //
```
Code verification on dynamic meshes

Complex mesh motion: interlacing paddles

- interlacing rotating paddles
- motion would be very difficult to resolve with GGI, RBF, or Laplacian dynamicFvMesh methods.

figures/twoPaddles.avi
Code verification on dynamic meshes
Water entry of projectiles

[Image: Water entry of projectiles video]
Validation for intended–use applications
Model 5415: an international benchmark for a naval surface combatant

- Gothenburg 2010 A Workshop on CFD in Ship Hydrodynamics, Dec 8-10, 2010
- 49th AIAA Aerospace Sciences Meeting, 4-7 January 2011
  - Steady resistance
  - Dynamic sinkage and trim (2DOF)
  - Ships in waves (diffraction problem)
  - Roll damping with bilge keels (1DOF)

Steady wave field at Fr = 0.28
Validation for intended–use applications
Model 5415: an international benchmark for a naval surface combatant

- Gothenburg 2010 A Workshop on CFD in Ship Hydrodynamics, Dec 8-10, 2010
- 49th AIAA Aerospace Sciences Meeting, 4-7 January 2011
  - Steady resistance
  - Dynamic sinkage and trim (2DOF)
  - Ships in waves (diffraction problem)
  - Roll damping with bilge keels (1DOF)

Comparison to experiment: dynamic sinkage and trim vs. speed
Validation for intended–use applications
Model 5415: an international benchmark for a naval surface combatant

- Gothenburg 2010 A Workshop on CFD in Ship Hydrodynamics, Dec 8-10, 2010
- 49th AIAA Aerospace Sciences Meeting, 4-7 January 2011
  - Steady resistance
  - Dynamic sinkage and trim (2DOF)
  - Ships in waves (diffraction problem)
  - Roll damping with bilge keels (1DOF)
Validation for intended-use applications
Model 5415: an international benchmark for a naval surface combatant

- Gothenburg 2010 A Workshop on CFD in Ship Hydrodynamics, Dec 8-10, 2010
- 49th AIAA Aerospace Sciences Meeting, 4-7 January 2011
  - Steady resistance
  - Dynamic sinkage and trim (2DOF)
  - Ships in waves (diffraction problem)
  - Roll damping with bilge keels (1DOF)
Summary

- **foamedOver** is a collection of custom classes, solvers, and applications which adhere to OpenFOAM object-oriented programming practices, and which utilizes the existing tools Suggar++, DiRTlib, and PETSc.

- It is a **stand-alone library** that provides a dynamic overset grid capability to any OpenFOAM solver.

- **motionObjects** have been developed which permit table-lookup, analytical, and 6DOF motions.