Geometric Immersed Boundaries (GIB): A New Framework For Applying Boundary Conditions in OpenFOAM®

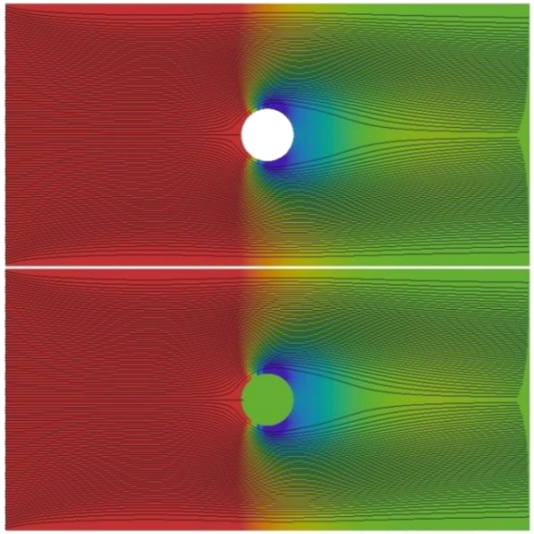
OpenFOAM Workshop 2015

29 June – 2 July 2015

University of Michigan, Ann Arbor

Georgios Karpouzas – ENGYS Ltd. /NTUA

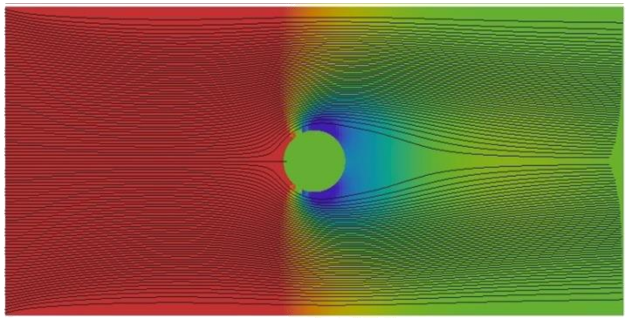
Eugene de Villiers – ENGYS Ltd.



info@engys.com | Tel: +44 (0)20 32393041 | Fax: +49 (0)20 3357 3123 | www.engys.com

Contents

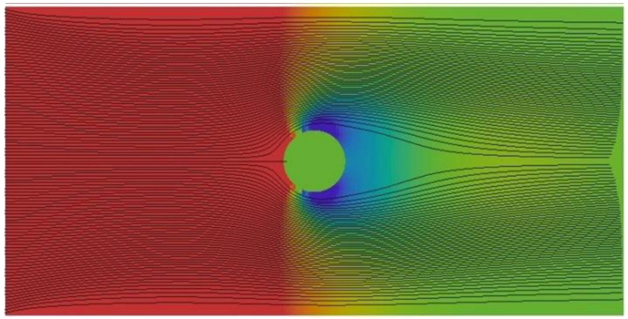
- Motivation
- Methodology
- Validation
- Applications
- Closing Comments



Motivation

- AboutFlow
  - Adjoint-based Optimisation of Industrial and Unsteady Flows
  - Initial Training Network (ITN) funded under FP7
- Improve immersed boundaries in context of topology optimization
  - Level-set coupled with the continuous adjoint method
  - Currently modelled immersed boundaries are applied on the fluid-solid interface (Tukovic)
  - Lacks of accuracy/robustness especially in turbulent cases
- Solution: Implement immersed boundaries with the same accuracy as a real boundary

Contents

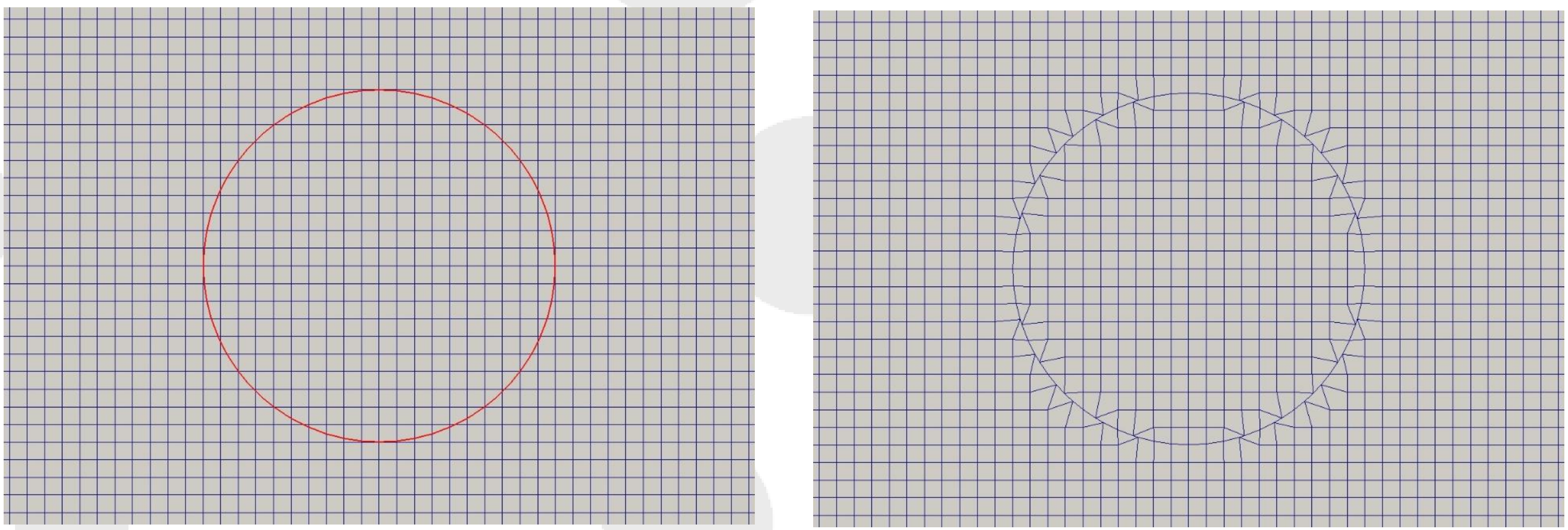


Methodology | Goals

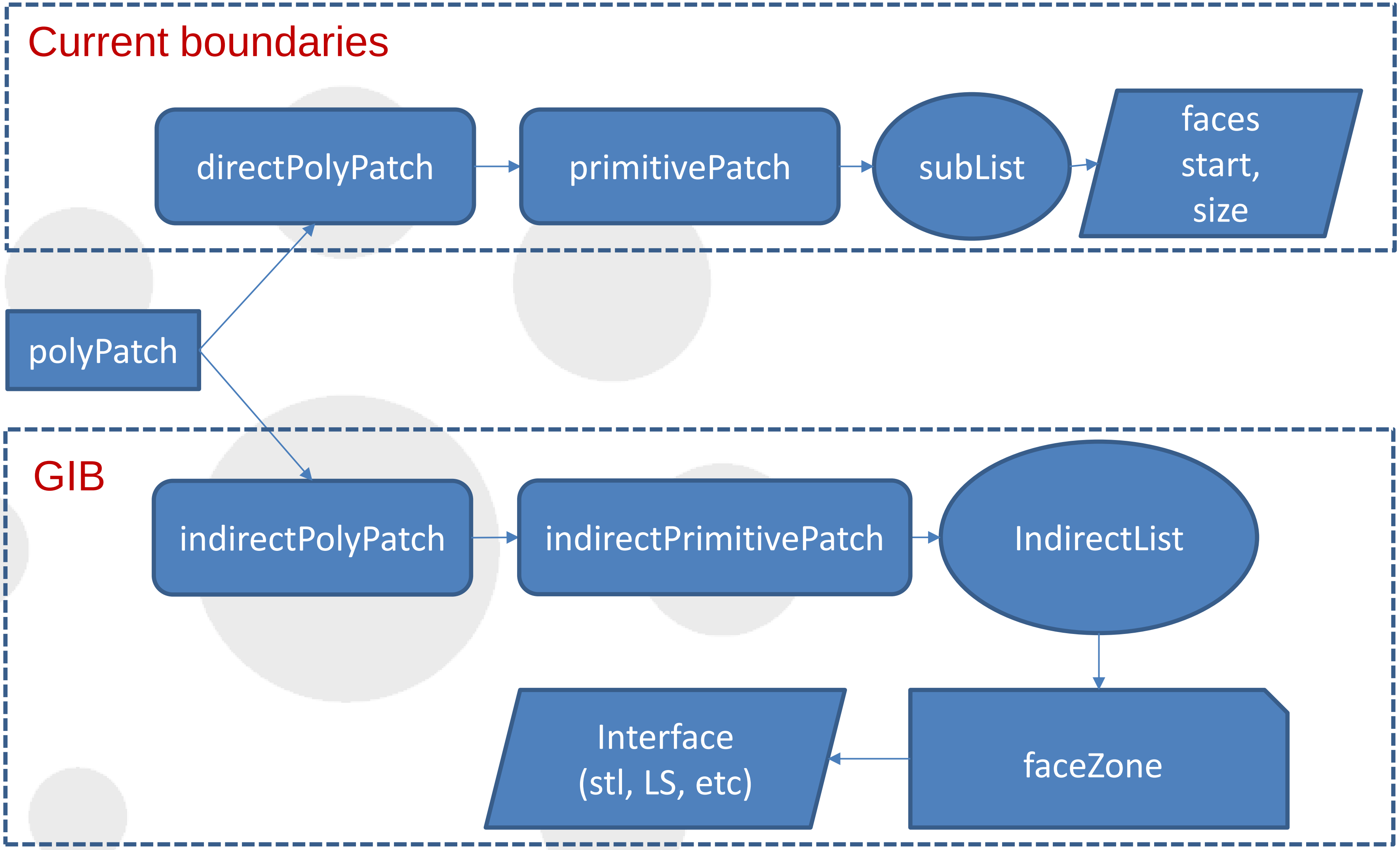
- Same accuracy as body fitted meshes
  - Conservative, implicit
- Automation: compatible with existing solvers and operations
- Same interface as the other boundaries
  - No extra files or non-local setup
- Re-use existing boundary conditions on the immersed boundaries without alteration
  - (fixedValue, zeroGradient…)

Methodology | Concept

- Conform internal faces to interface location (LS, .stl, etc.)
  - Mesh motion + optimisation required
- Update all finite volume quantities affected by interface faces
- Problem: There is not a code structure in OpenFOAM® to apply boundary conditions in internal faces.



Methodology | Implementation



Methodology | Implementation

- Two new boundaries are constructed
  - One for each side of interface
  - Based on faceZone and flipMap
- Uses existing boundary conditions
- GIB boundary faces give appropriate contributions to matrix and other FV operators
- The GIB can behave like a pass-through, normal boundary or hybrid combination of the two
- More than 200 files modified or added (so far)

Methodology | Implementation

Boundary file sample

```
7
(
    Inlet
    {
        type patch;
        physicalType inlet;
        nFaces 100;
        startFace 39700;
    }
    …
    Ib1
    {
        type indirectWall;
        neighbourPatch ib2;
        faceZone ib;
        indirectPolyPatchType master;
        startFace 80300;
    }
    Ib2
    {
        type indirectWall;
        neighbourPatch ib1;
        faceZone ib;
        indirectPolyPatchType slave;
        startFace 80300;
    }
)
```

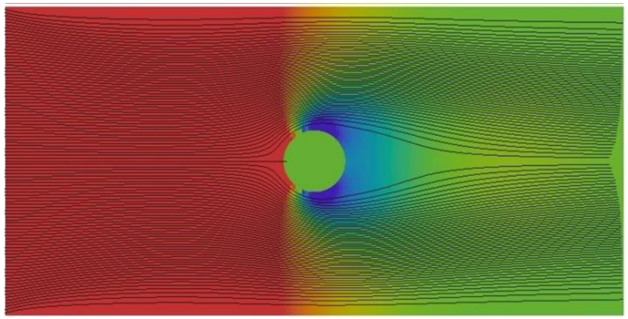
U Boundary Conditions sample

```
…
boundaryField
{
    Inlet
    {
        surfaceNormalFixedValue;
        redValue uniform -1;
    }
    …
    Ib1
    {
        type fixedValue;
        value uniform (0 0 0);
    }
    Ib2
    {
        type fixedValue;
        value uniform (0 0 0);
    }
}
```

p Boundary Conditions sample

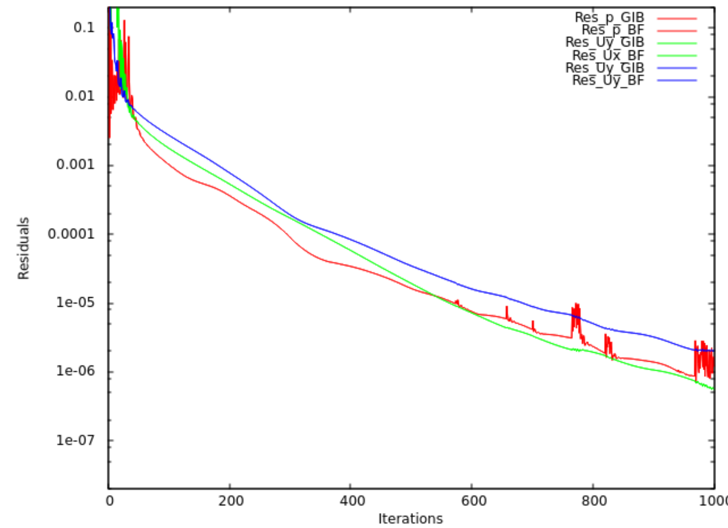
```
…
boundaryField
{
    Inlet
    {
        type zeroGradient;
    }
    …
    Ib1
    {
        type zeroGradient;
    }
    Ib2
    {
        type zeroGradient;
    }
}
```

Contents



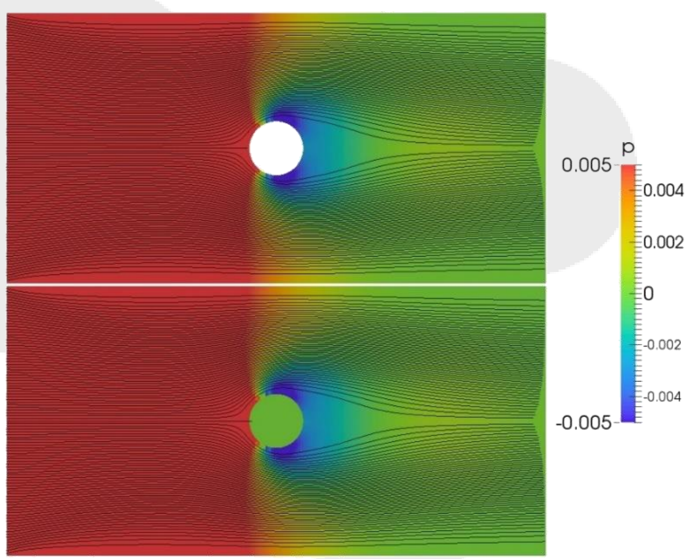
Validation | cylinder

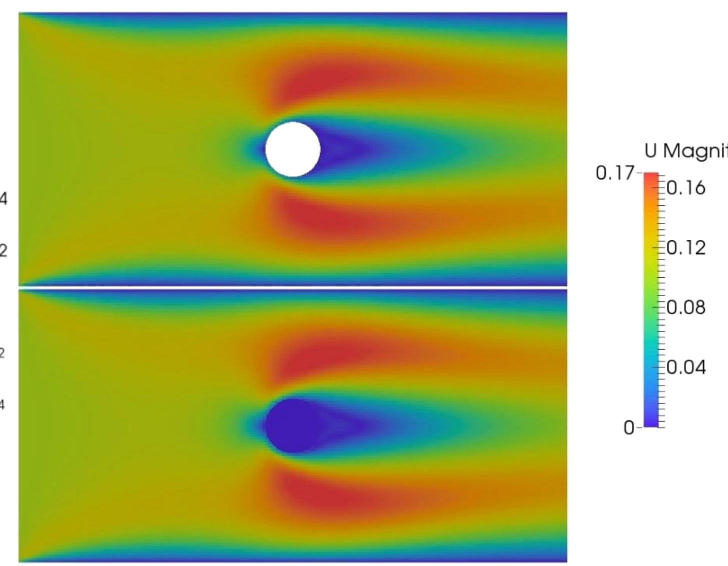
- Body-fitted vs GIB cylinder results
- Identical residuals-results



Body-Fitted

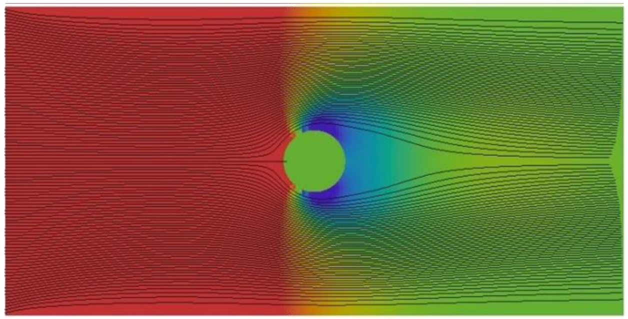
GIB





Contents

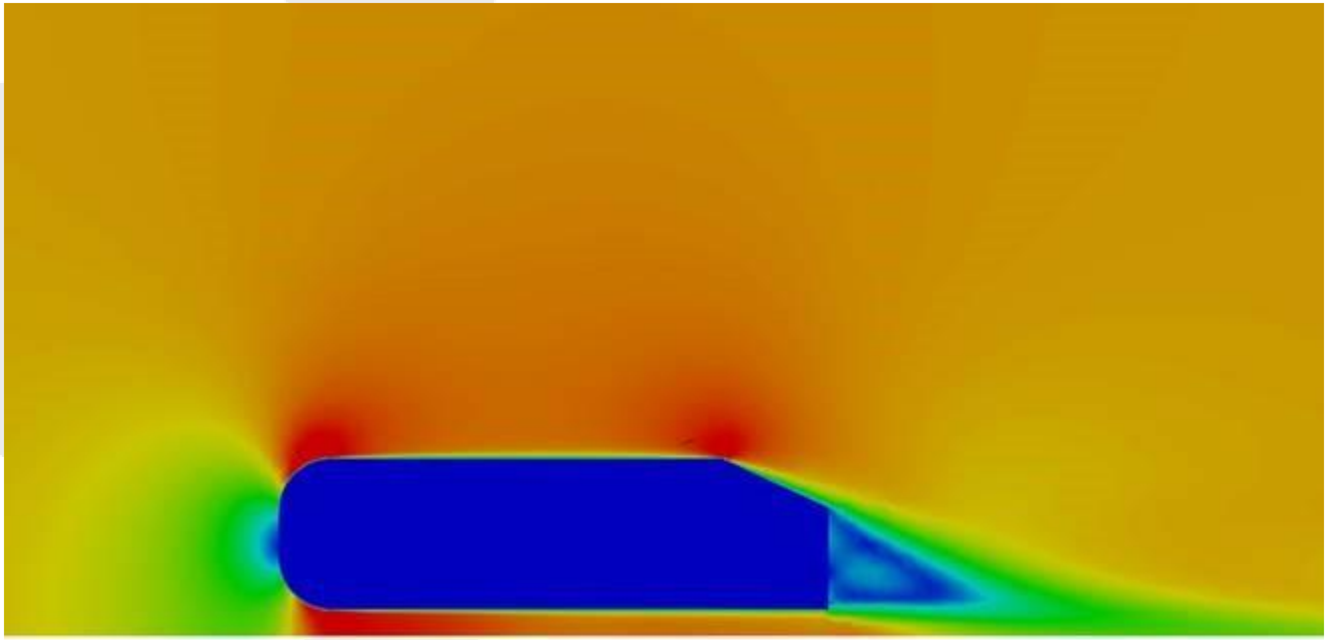
- Motivation
- Methodology
- Validation
- Applications
- Closing Comments



Applications | Ahmed Example

- Fully parallel
- Works with turbulence
- No top level change is required in the standard solvers

simpleFoam

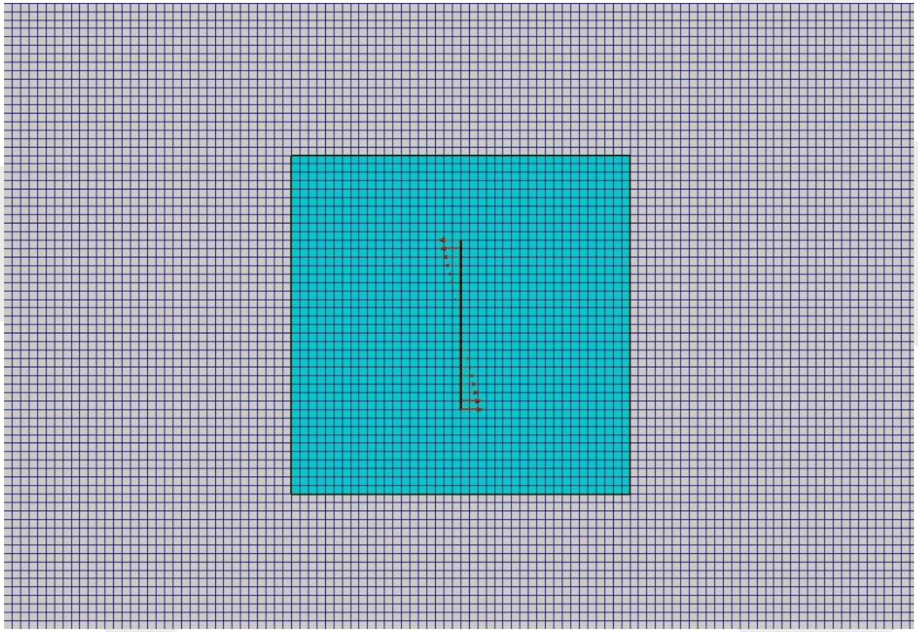


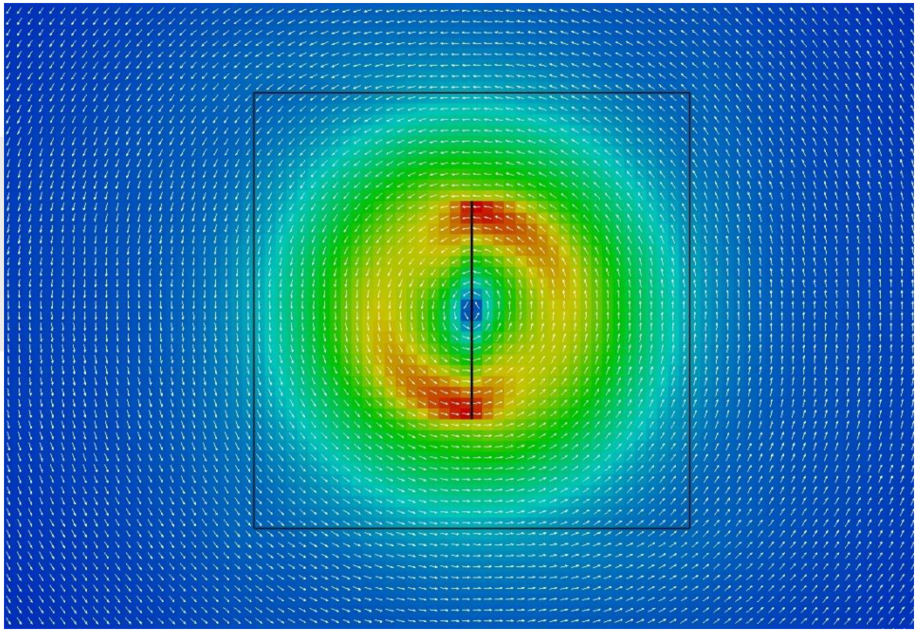
Applications | MRF | Concept

- Current constraint:
  - The cellZone must be circular.
  - Reason: The relative and absolute fluxes should be the same at the interface of stationary and rotating part
- Apply GIB on the interface:
  - The pressure boundary is a pass-through
  - The velocity and the derived fields (phi, …) takes the value of the GIB wherever needed for FV operations
  - The relative flux is added only in one part of the GIB

Applications | MRF | simpleMixer

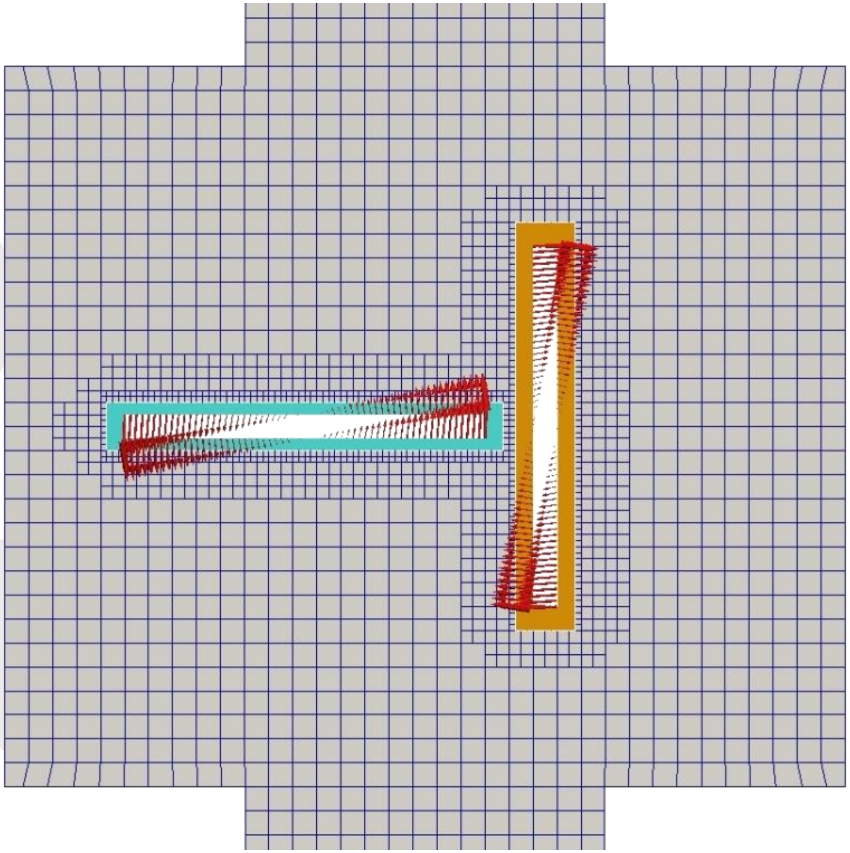
- Simple blockMesh geometry with a blade and a cellZone.
- GIB applied at the perimeter of the cellZone



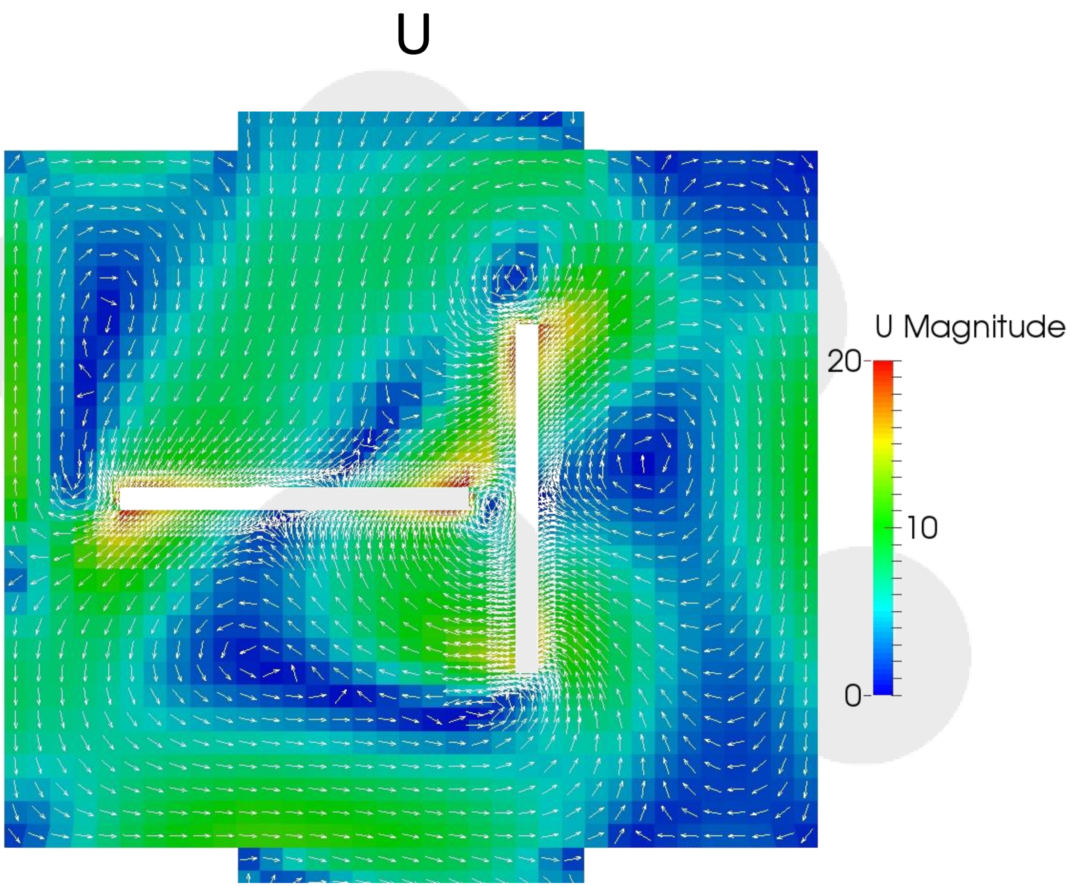


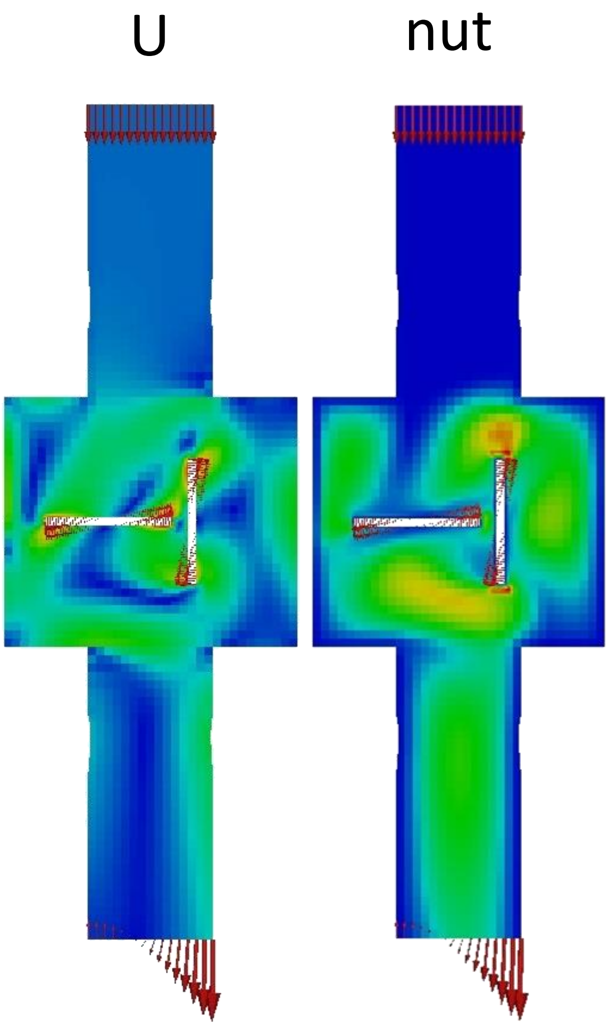
Applications | MRF | “Gear Pump”

- Pressure-pressure boundaries at the top and bottom
- Cyan and orange areas are two cellZones
- Cannot be simulated by standard MRF method in OpenFOAM®
- Independent GIBs applied on cellZone interfaces.
- GIB boundary conditions are coupled (communication is required)



Applications | MRF | “Gear Pump”



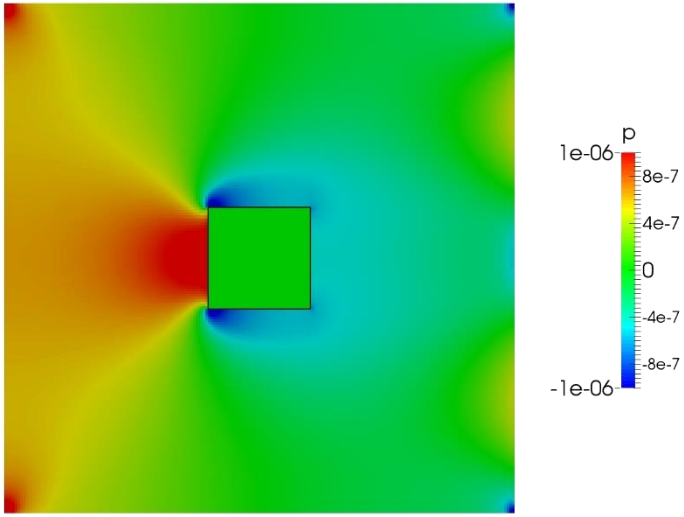


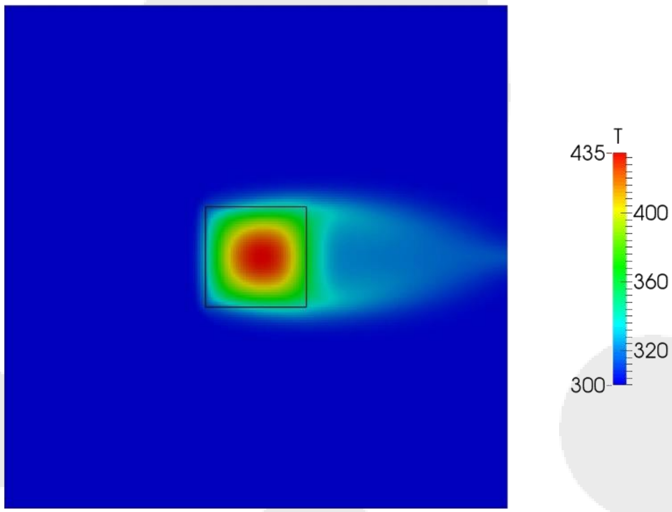
Applications | CHT | Current Technique

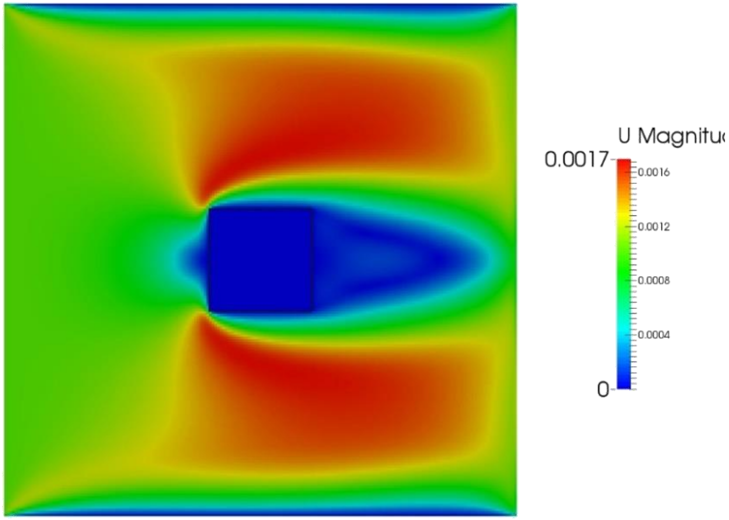
- Segregated multi-region solution
- Basic equations:
  - For fluid:
    - Pressure
    - Velocity
    - Energy (enthalpy or temperature)
  - For solid:
    - Energy (enthalpy or temperature)
- Result: slow convergence

Applications | CHT | GIB

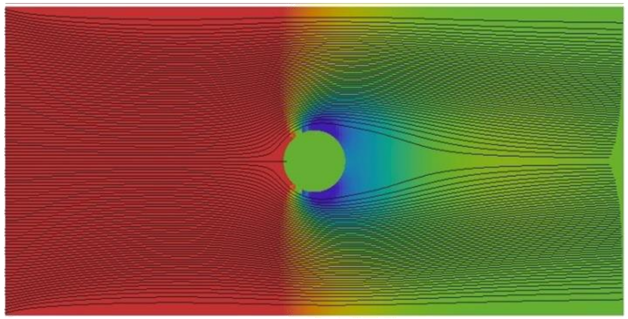
- Heated square in a cross-flow
- Single region CHT
- Solid and fluid communicate via GIB (black line)
  - Coupled thermal boundary conditions
- Heat source is applied on the solid
- 1 matrix -> faster convergence







Contents

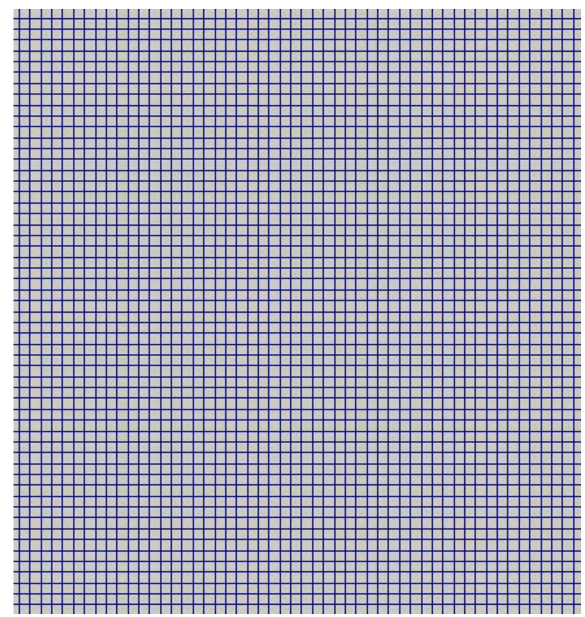


Closing comments

- New framework for applying boundary conditions on internal faces implemented
  - Accuracy equivalent to normal boundary conditions
- Extreme ease of use and full integration with existing infrastructure
  - Parallel, GAMG
- Can be applied to variety of applications
  - CHT, MRF, FSI, multiphase, topology optimization & combinations
- Project goal: adjoint optimisation of CHT

Closing comments

- Next step: unsteady GIB
  - Moving/deforming solids and solid-fluid interfaces
- Several outstanding components remain
  - faceZone & polyPatches addressing update
  - Boundary value mapping
  - Field transport & flux consistency
  - Reliable local mesh deformation
  - Dynamic interface refinement



The end

Thanks for your time! Any questions?