POLITECNICO DI MILANO





Mesh motion strategies for the simulation of external gear pumps

J. Martínez, F. Piscaglia, A. Montorfano

Internal Combustion Engine Group
Department of Energy

OFW11, Guimarães (Portugal), June 27th 2016

Outline

- **Motivation and state of the art**
- **Mesh motion strategy**

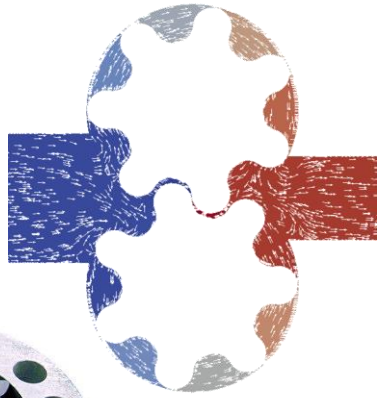
  Profile definition and mesh generation
  Interface calculation
  Projection algorithm
  2D-3D mesh motion
  Parallelization
  Topological changes

- **Implementation**
- **Test cases**
- **Conclusions and future work**

Motivation and state of the art

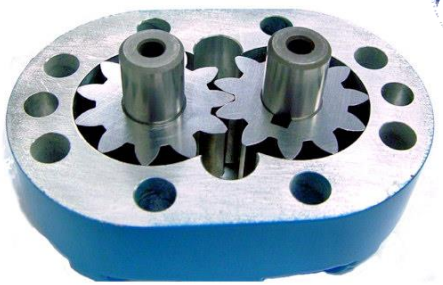
External gear pumps:

- High viscosity fluids
- Working pressure up to 210 bar
- Rotation speed up to 3000 rpm
- Mechanical clearance $\approx 10\ \mu m$



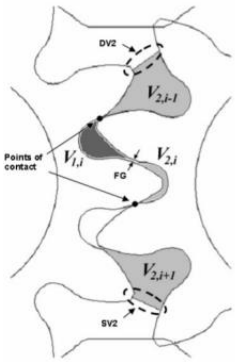
Crucial predictions:

- Pressure ripple
- Flow rate pulsations
- Max/min pressure

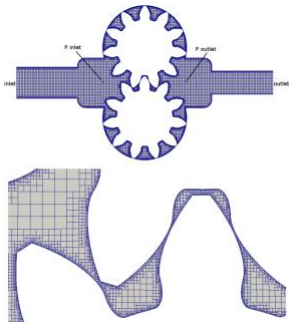


Simulation strategies:

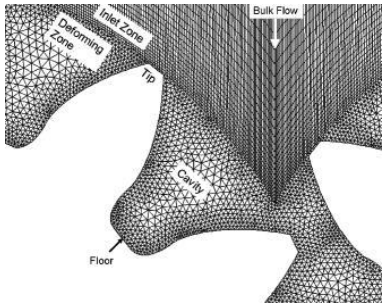
1D-2D simulations

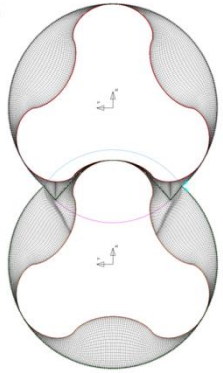


Multiple-mesh approach



User defined dynamic mesh



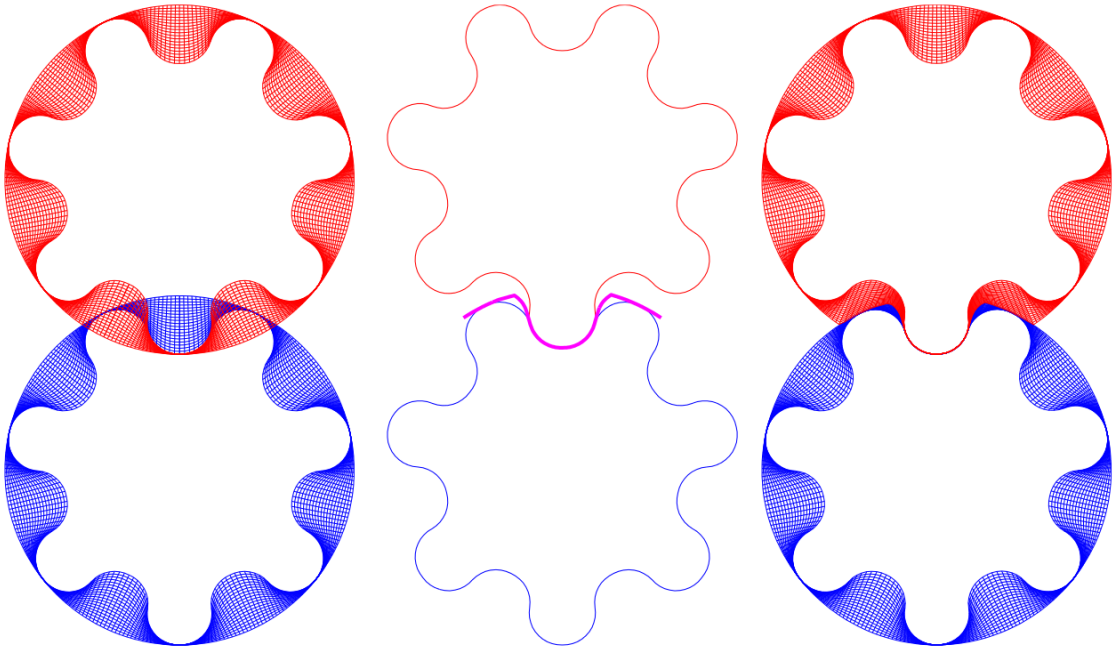


Mesh motion strategy

Overview of the proposed strategy:

Single dynamically changing 2D/3D mesh

- Constant number of cells/points (speed)
- No mesh-to-mesh interpolation (accuracy)
- Automatic mesh generation (extendibility)



1.- Solid rotation of “non projected points”

2.- Calculation of an “interface” for the given angle

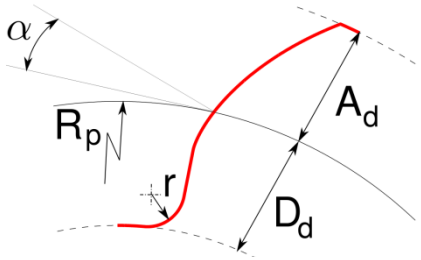
3.- Projection of points to the calculated interface

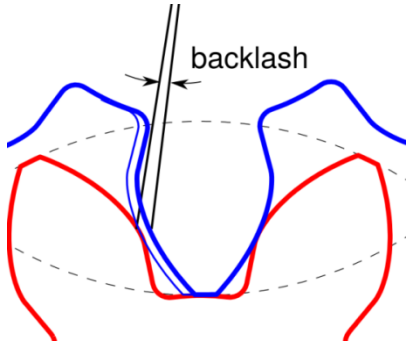
Mesh motion strategy

Profile definition

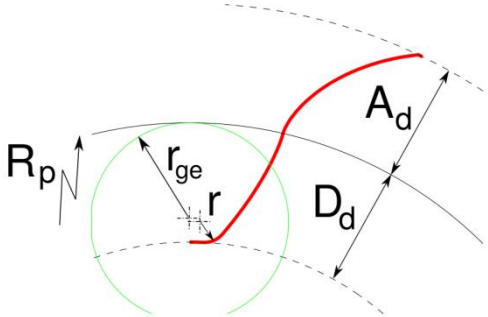
The user can choose a predefined geometry or provide a list of points

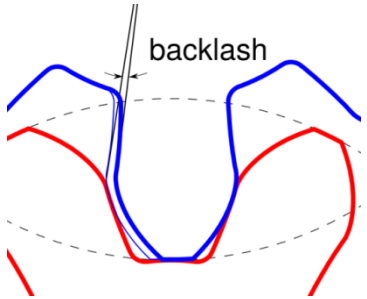
Involute
- Pressure angle (α)
- Number of teeth (Z)
- Backlash angle



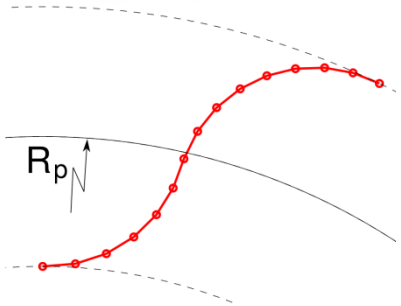


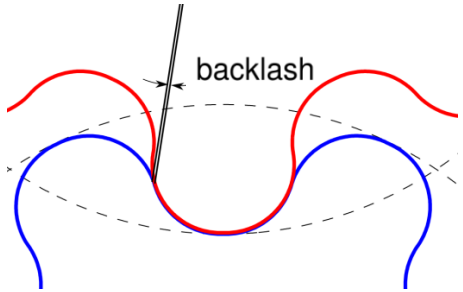
Cycloidal
- Generating circle radius (r_{ge})
- Number of teeth (Z)
- Backlash angle





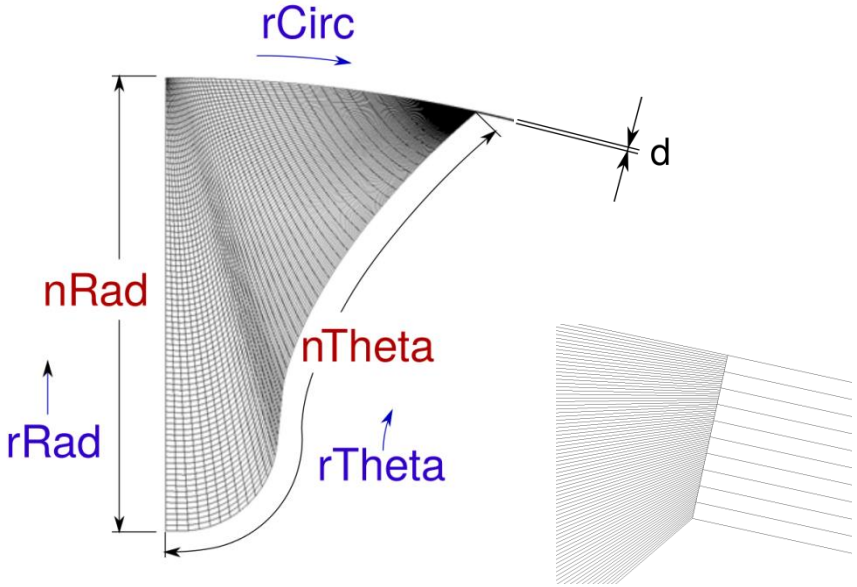
User-given pointList
- List of points
- Backlash angle

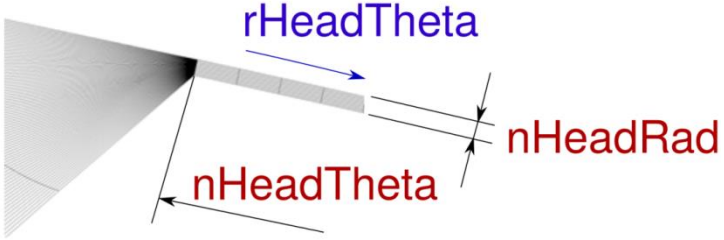




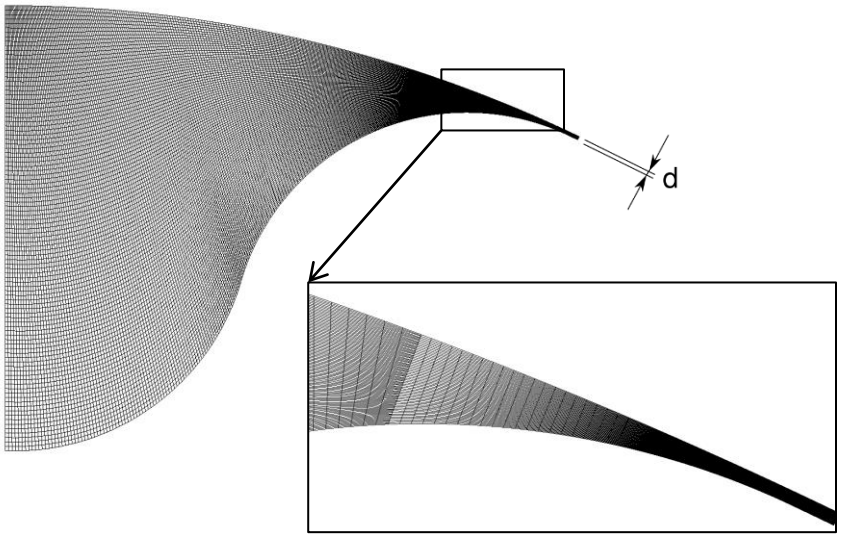
Mesh motion strategy

Generation of initial “non-projected” mesh

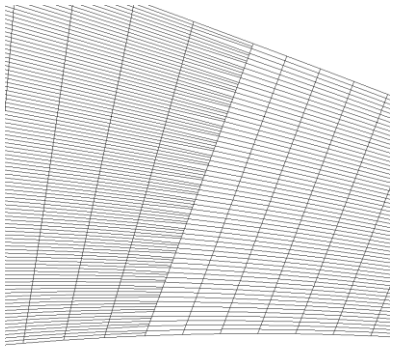




User defined number of cells and grading ratios:



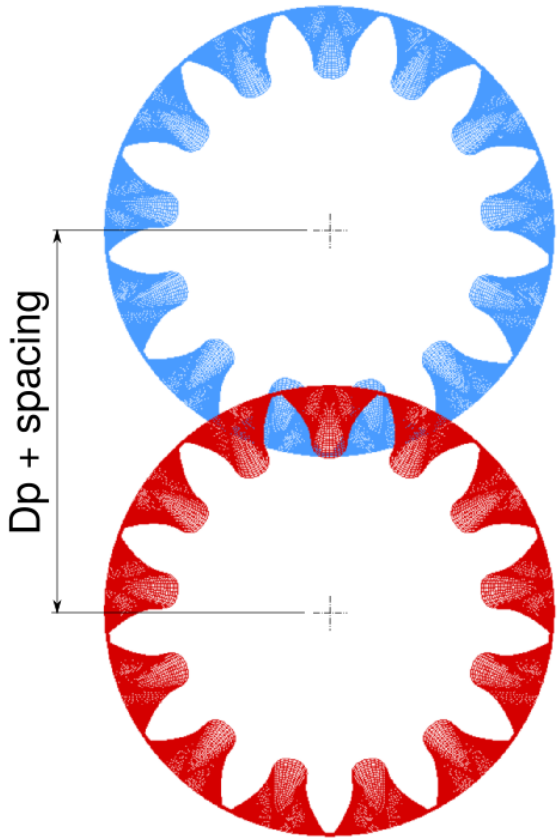
Block splitting with non-conformal interfaces

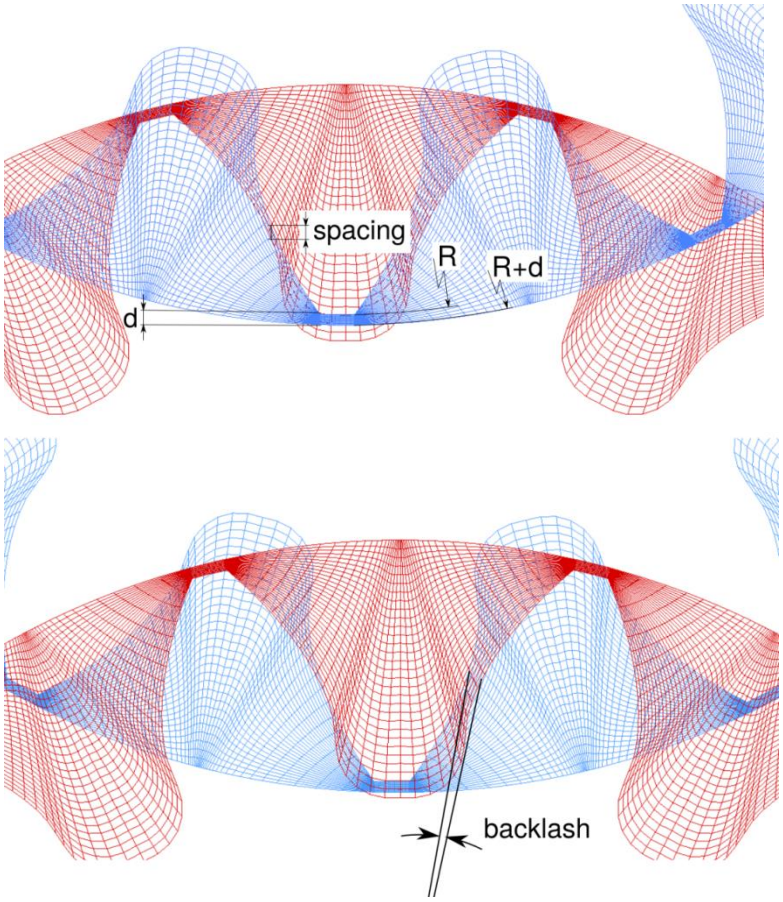


Mesh motion strategy

Generation of initial “non-projected” mesh

The initial block is repeated to fill both gears.
A tolerance for the distance between centers is introduce



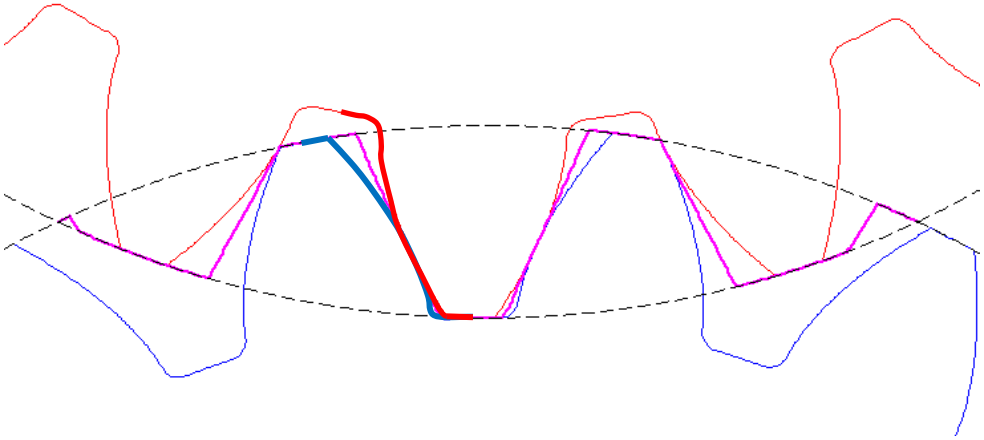


Mesh motion strategy

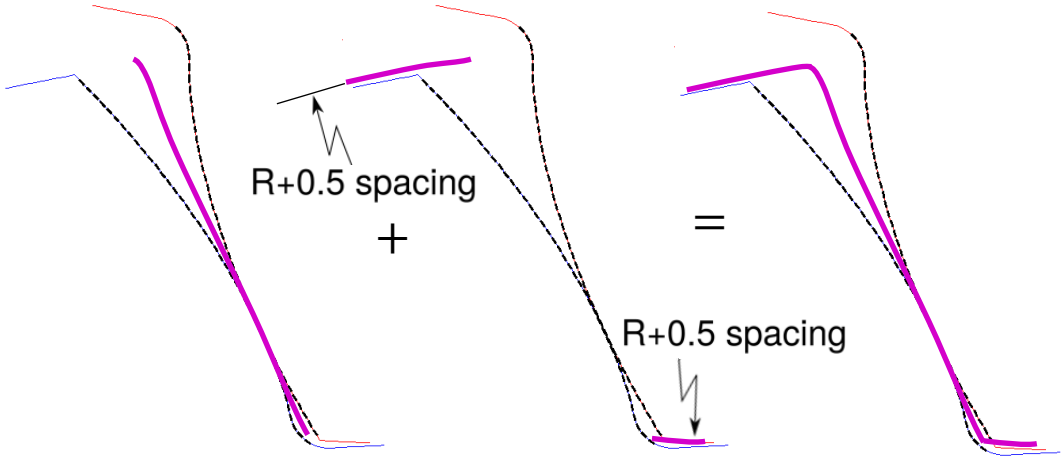
Interface calculation

Method1 - Separated profiles

The gearing zone is separated in “half-tooth” couples, which are used to determine “single interfaces”.



Single interfaces are joined with arcs to complete the whole interface.

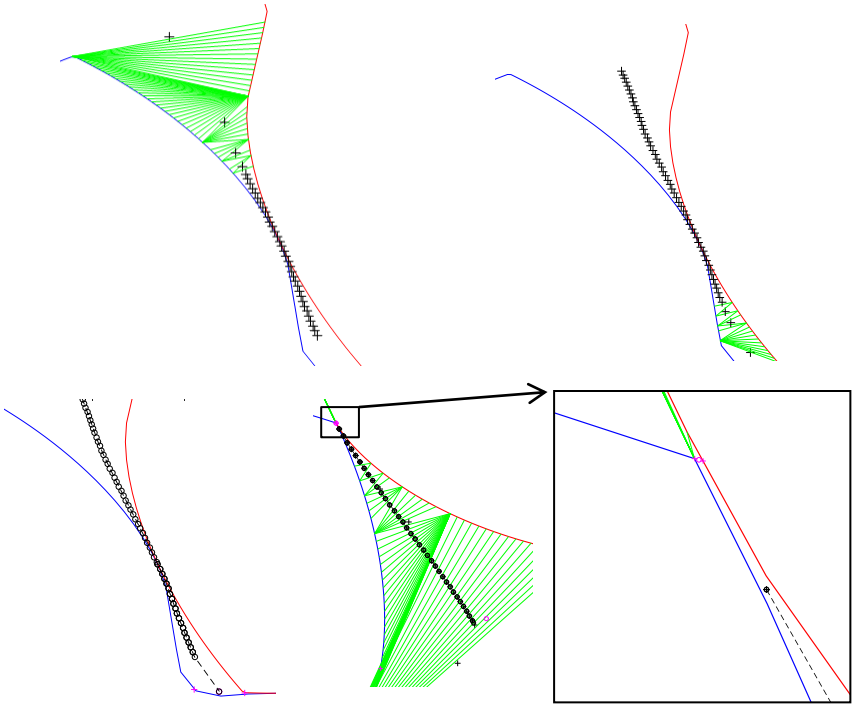


Mesh motion strategy

Method1 - Separated profiles

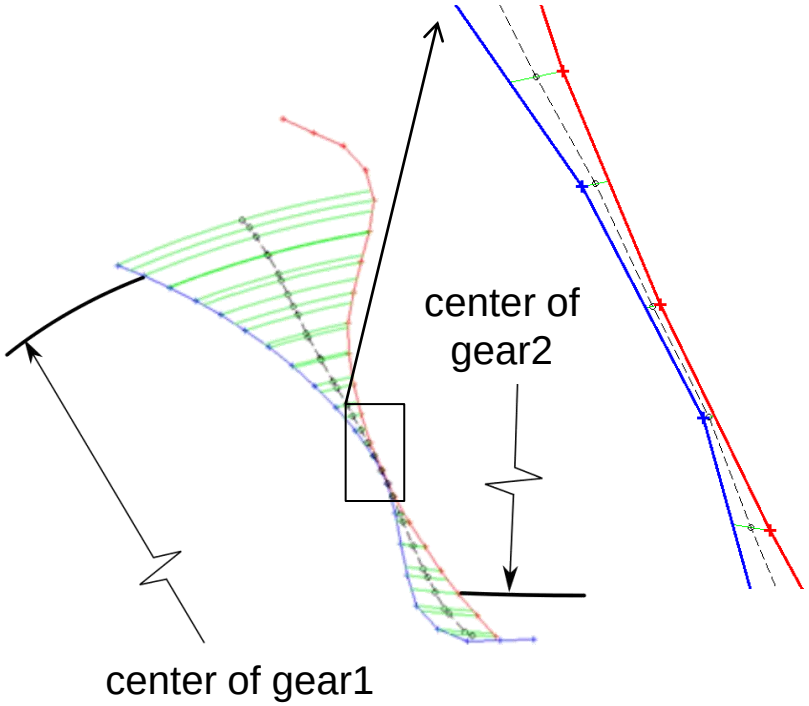
Two possible ways of calculating the “single interfaces”

Fréchet



Interface is calculated using profiles with *nPointsInterface* number of points, which is independent of the number of mesh points

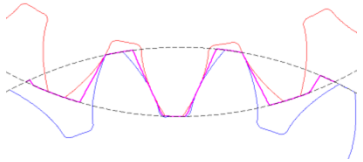
Arcs



Interface is calculated using the same points as those defining the mesh

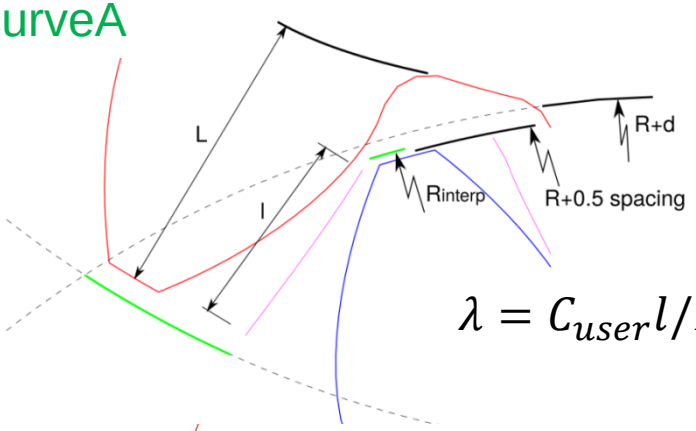
Mesh motion strategy

Method1 - Separated profiles



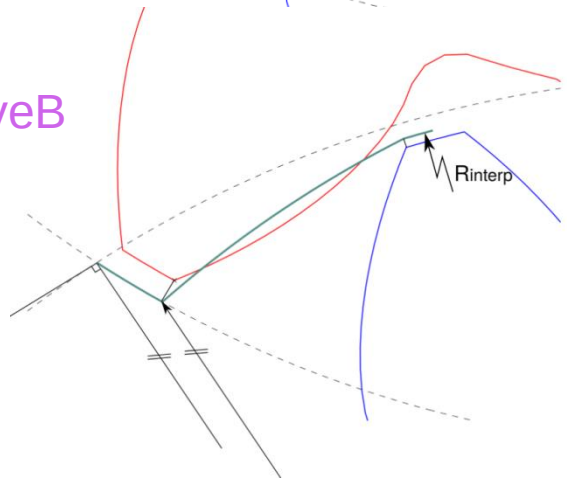
First and last “single interfaces” need to be completed in a different way

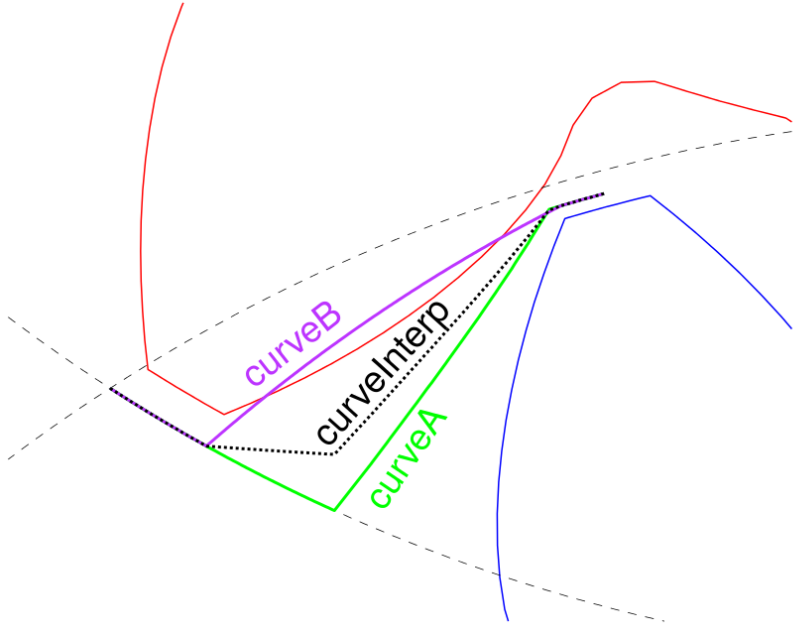
curveA



$$\lambda = C_{user} l/L$$

curveB





Final curve is a blending

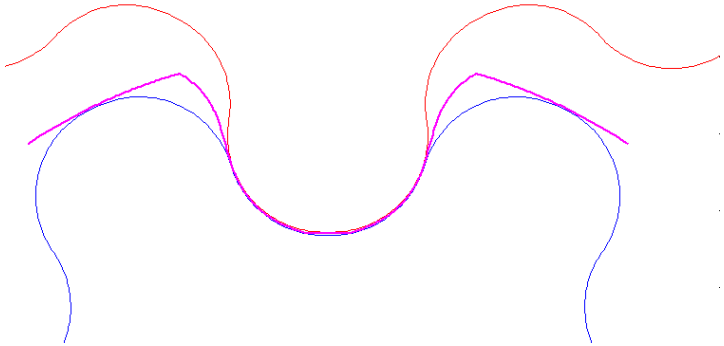
$$curveInterp = \lambda\, curveA + (1 - \lambda)\, curveB$$

Mesh motion strategy

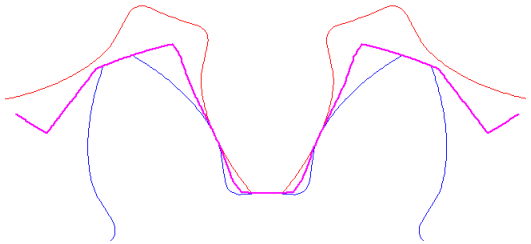
Method1 - Separated profiles

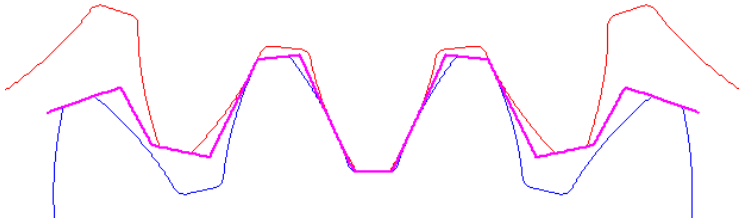
Some examples:

Lobe profile:

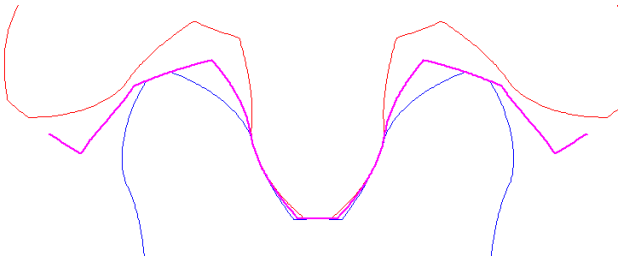


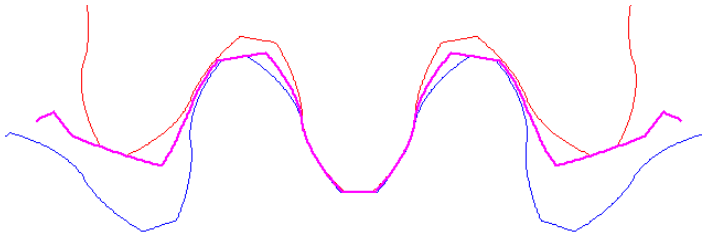
Involute profiles:





Cycloid profiles:

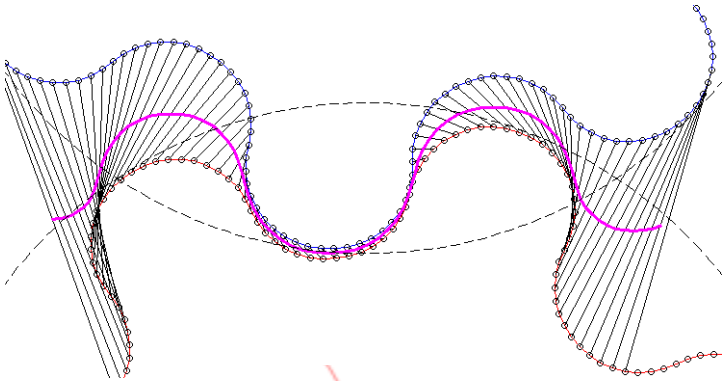




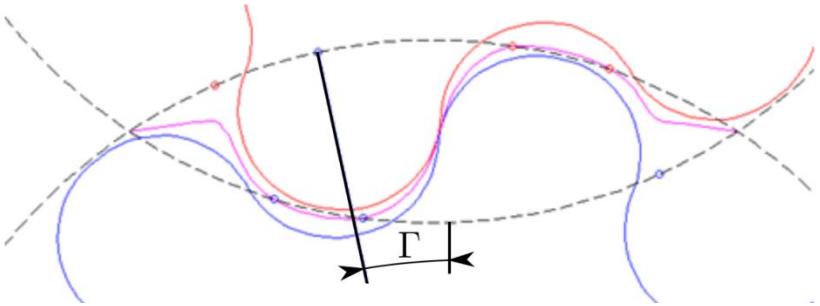
Mesh motion strategy

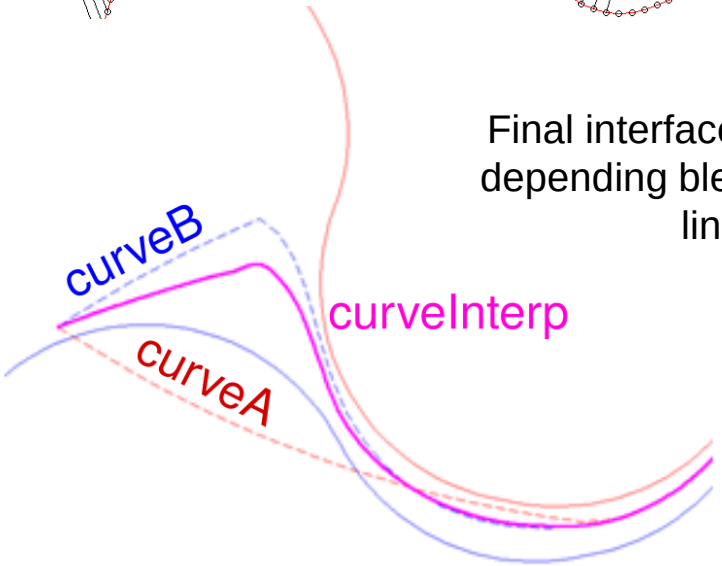
Interface calculation

Method2 – Complete profile

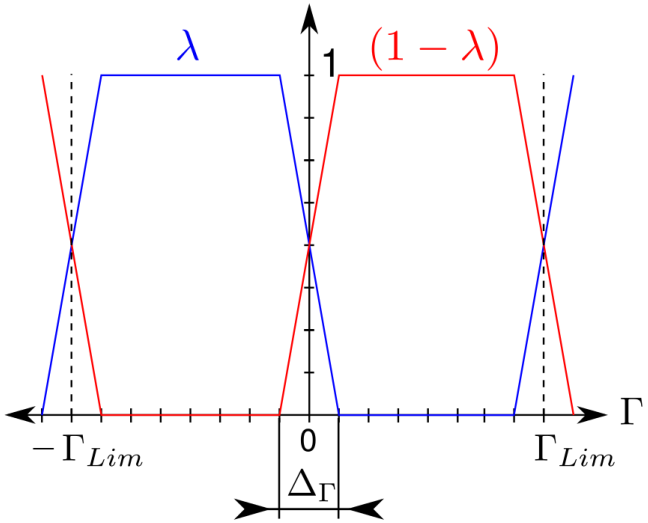


The whole interface is calculated at once





Final interface is an angle-depending blend of dashed lines

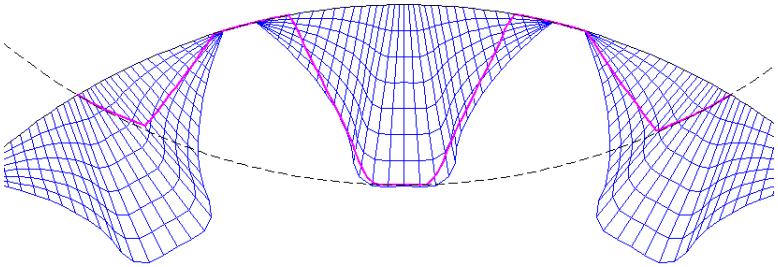


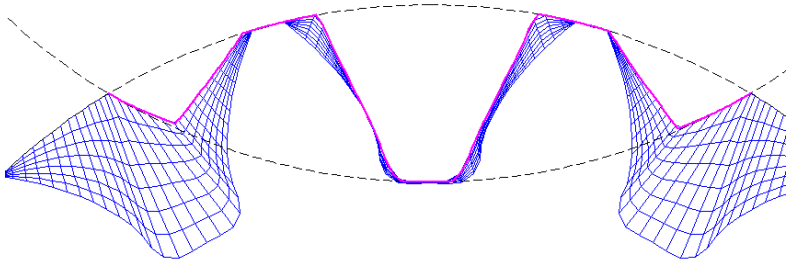
$$curveInterp = \lambda\, curveB + (1 - \lambda)\, curveA$$

Mesh motion strategy

Projection algorithm

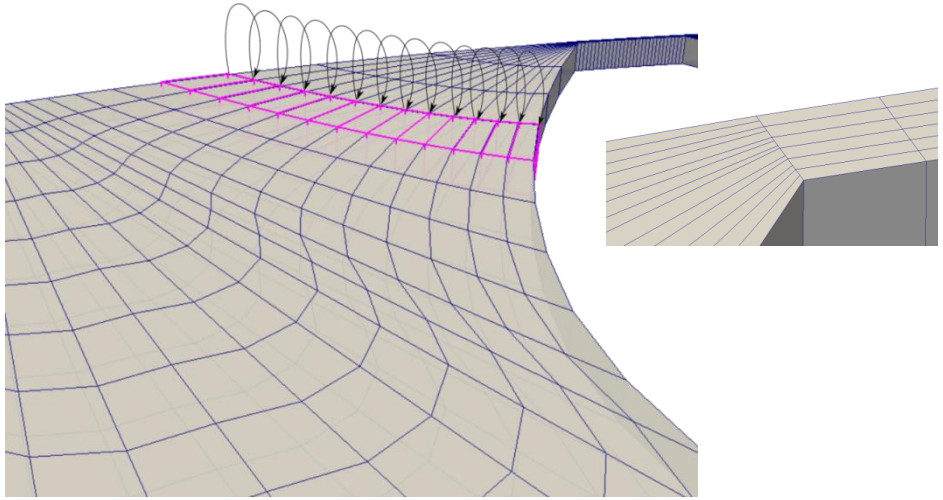
Initial non-projected points are moved to their final projected position

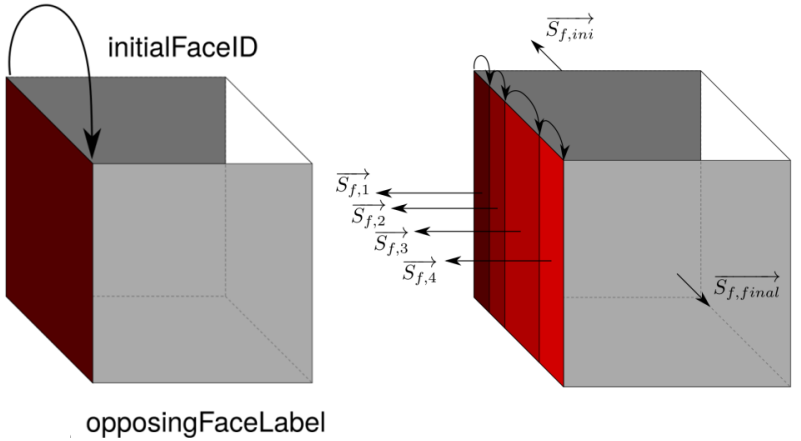




At the beginning of the calculation a matrix containing all ordered point labels of the gears is created

matrixOfPoints_[zoneI][zLevel][circPointI][interiorPI]



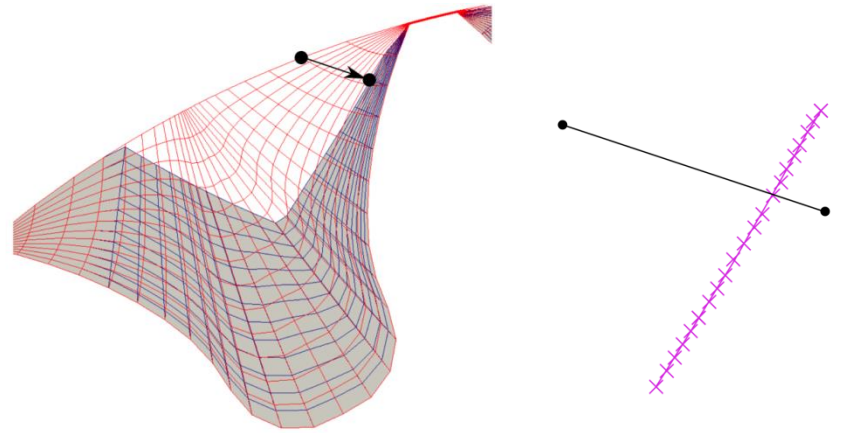


Mesh motion strategy

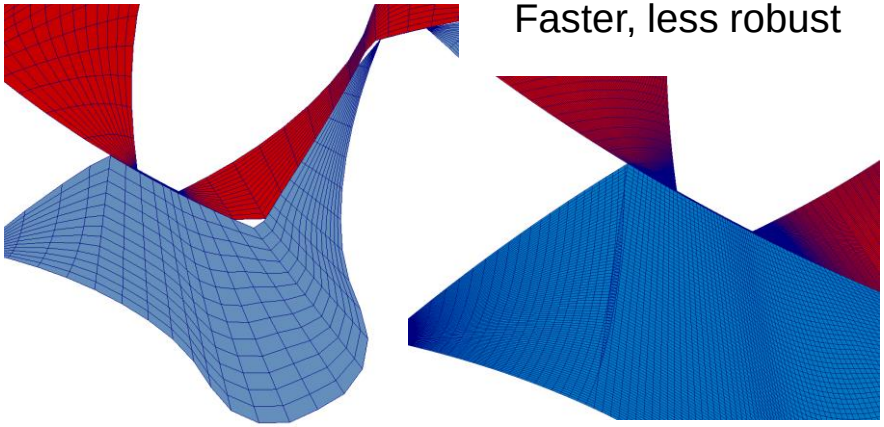
Projection algorithm

Intersection between the edges of the mesh and the interface is found and points are projected using any of the two following strategies

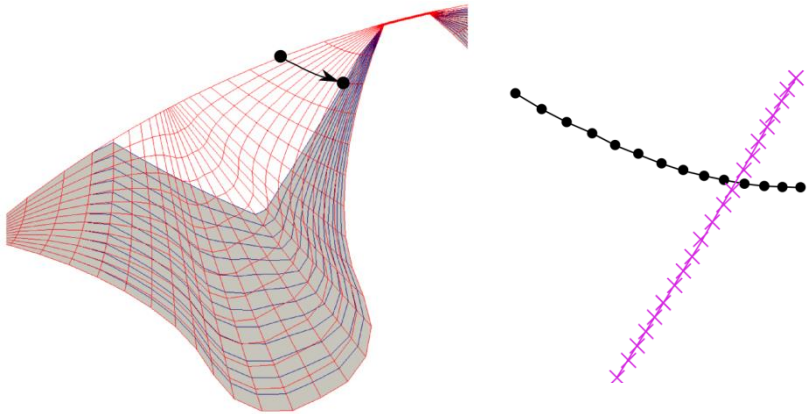
linearProjection



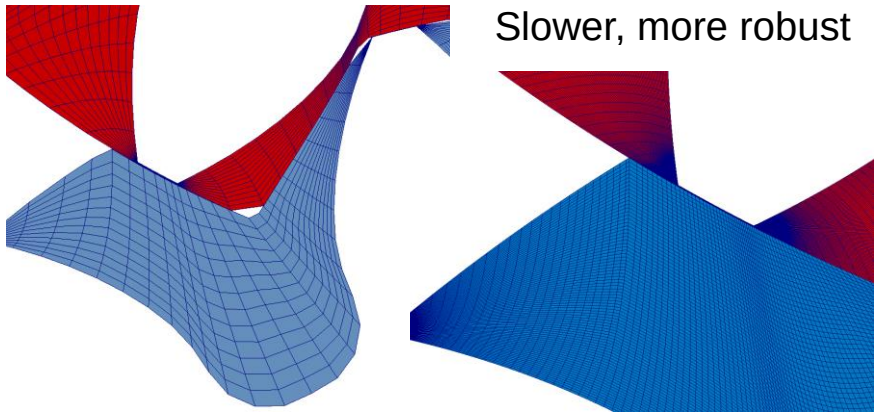
Faster, less robust



polyLineProjection



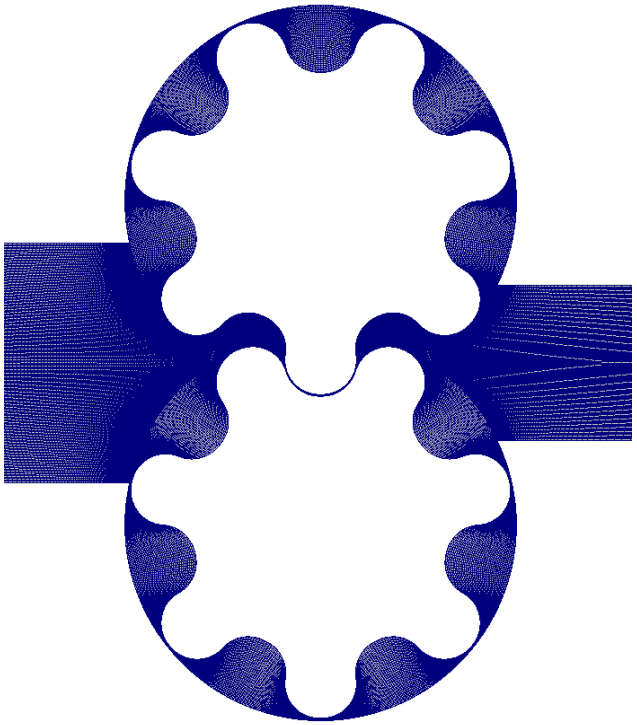
Slower, more robust

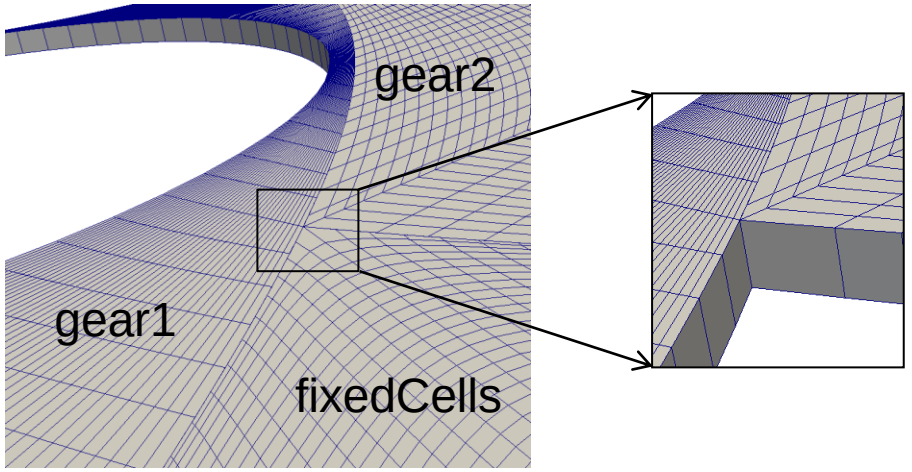


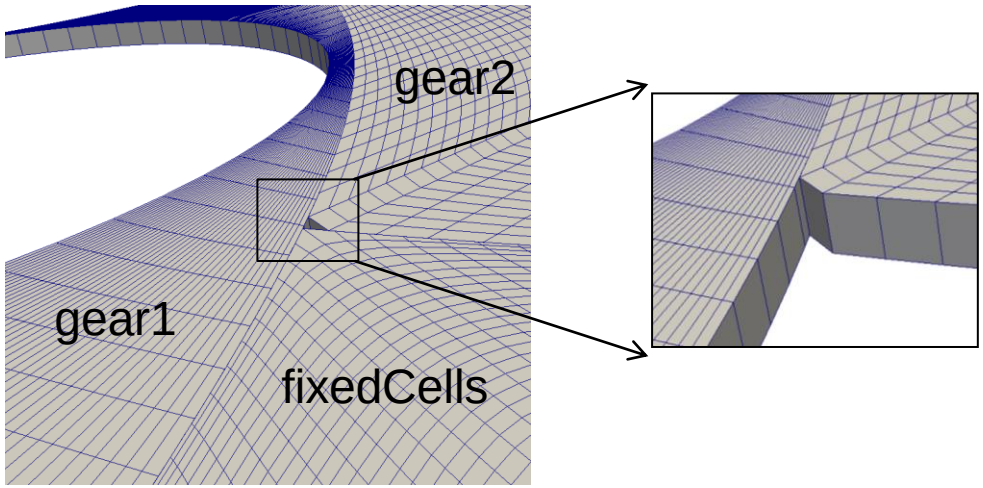
Mesh motion strategy

2D mesh motion

Limit points can be further projected to avoid wholes and facilitate boundary condition treatment





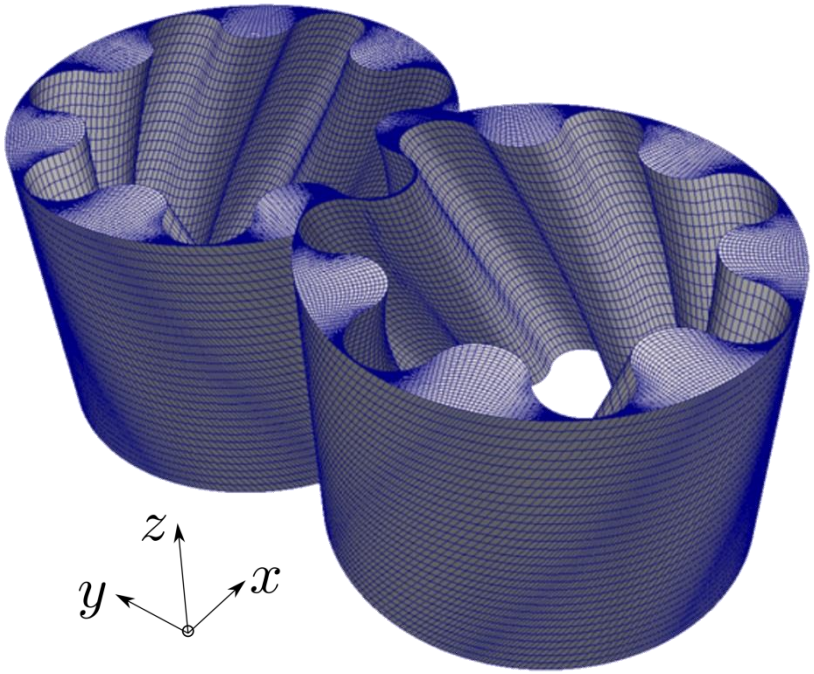


Mesh motion strategy

3D mesh motion The same strategy can be applied in the 3D case

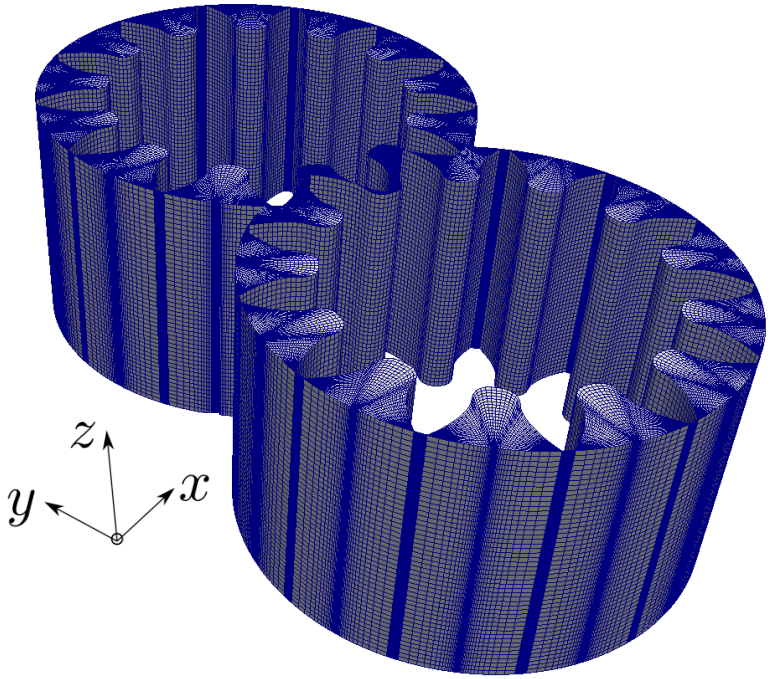
Helical gears

Points are contained in constant *z* planes. Interface calculation and point projection is determined for the points in each *z-Level*



Straight-cut gears

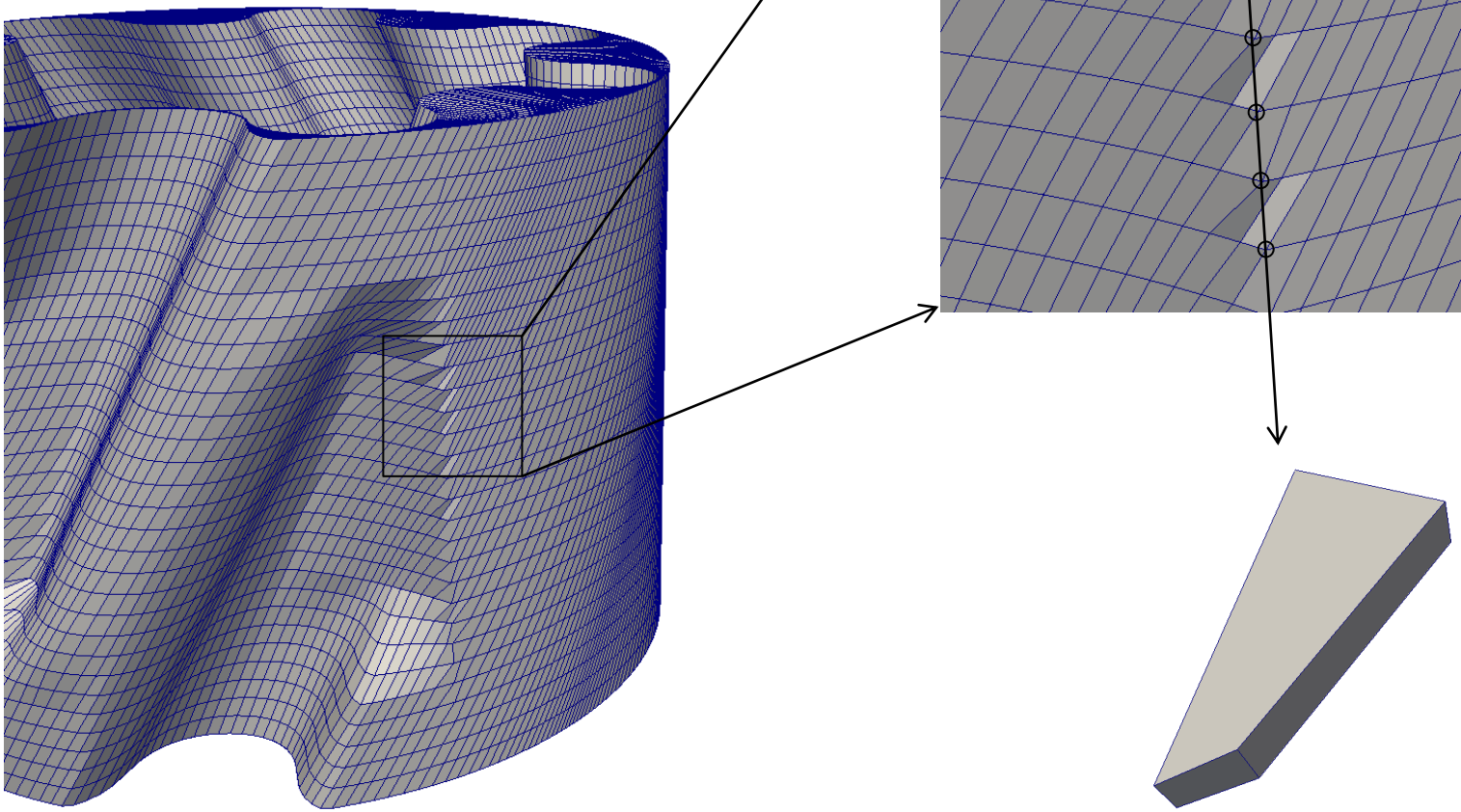
Interface calculation and point projection (final coordinates determination) need to be done just for one *z-Level*



Mesh motion strategy

3D mesh motion

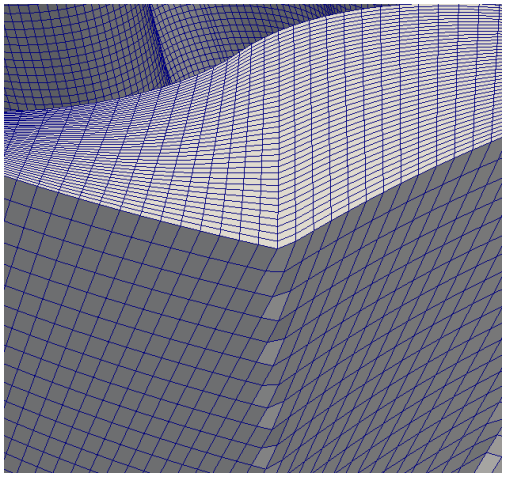
Projection of points in limit edges can generate concave cells

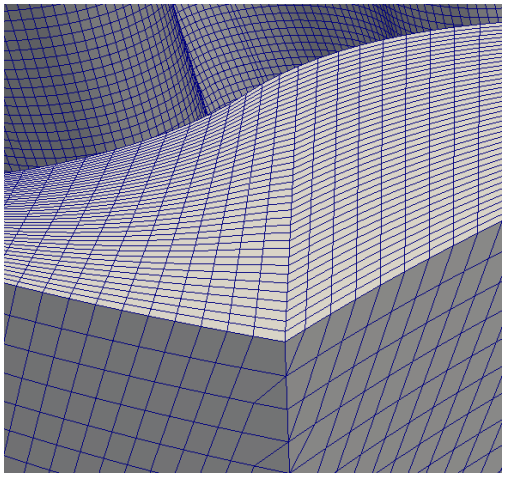


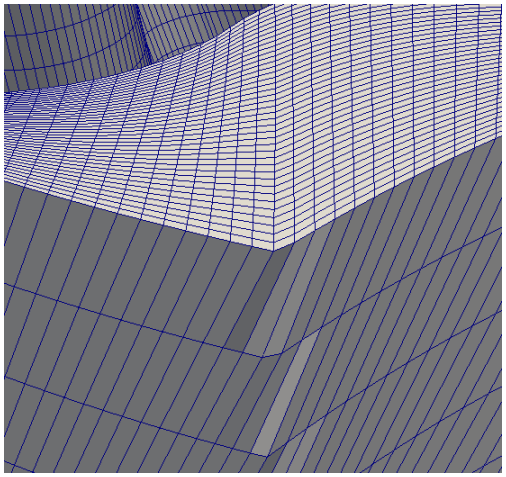
Mesh motion strategy

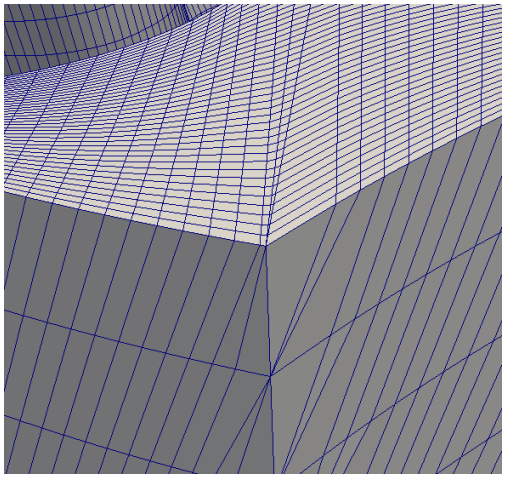
3D mesh motion

The motion class can optionally project some additional points are projected to the limit edge, generating wedge-like cells

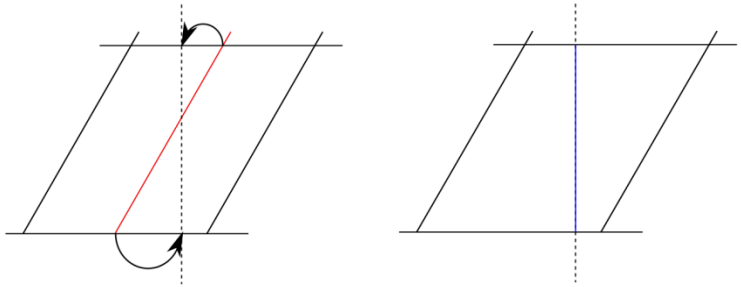




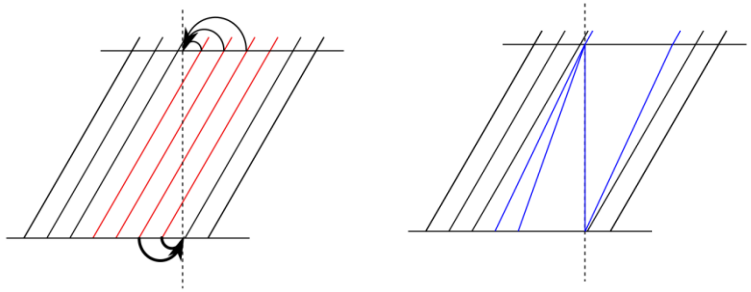




No “*zeroArea*” faces are created



We choose one edge that will become vertical

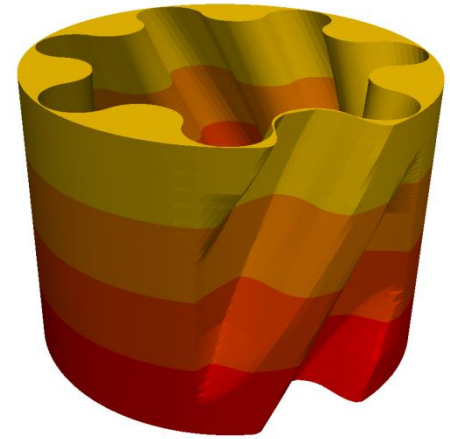


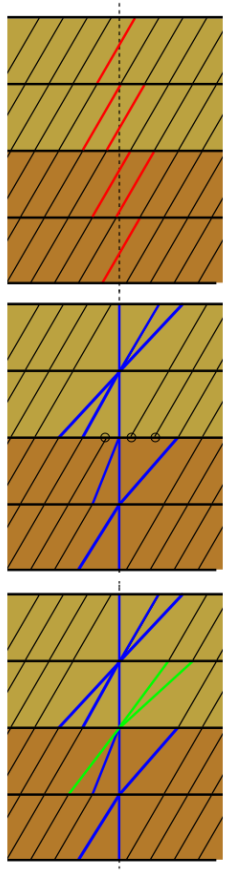
Mesh motion strategy

Parallelization

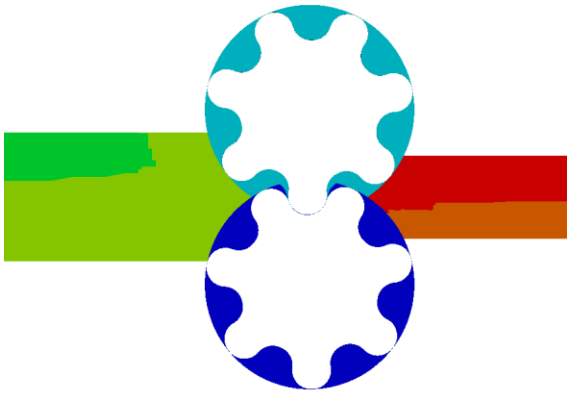
Cells of each single gear are kept in one processor. Rest of the domain can be decomposed with any other strategy (e.g. scotch)

Synchronization of points

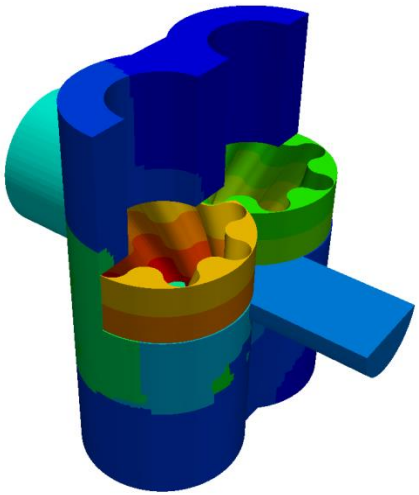


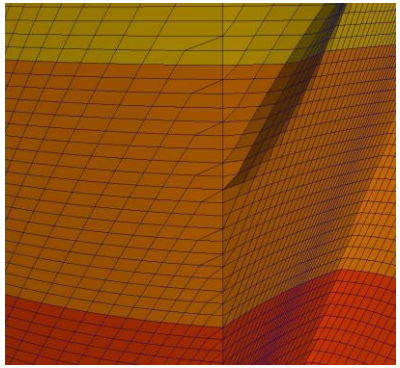


2D case



3D case



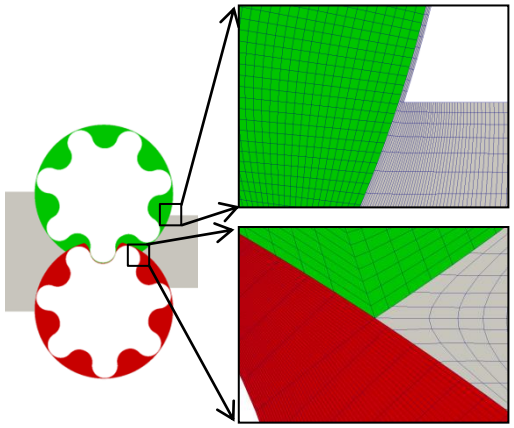


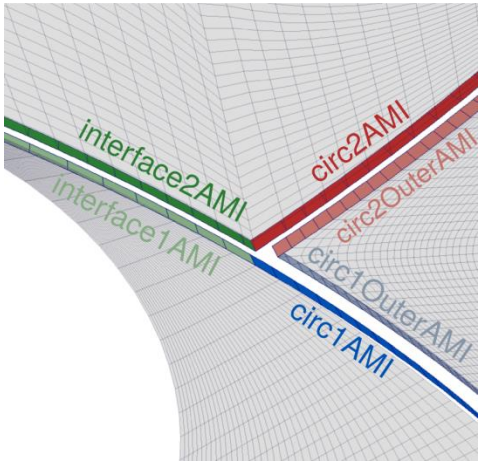
Mesh motion strategy

Topological changes

Two possible strategies to treat the boundaries between regions

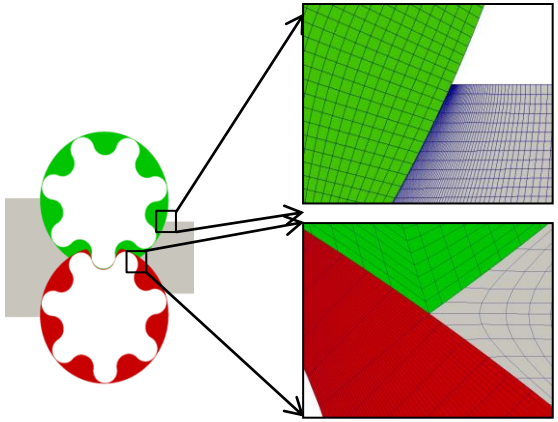
Using AMI conditions

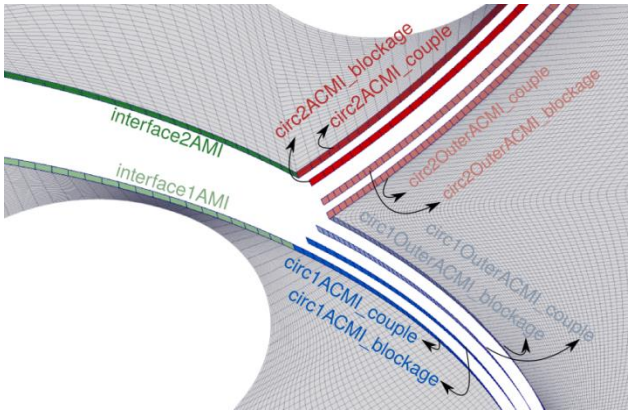




When required a face is moved from one patch to another

Using AMI and ACMI conditions



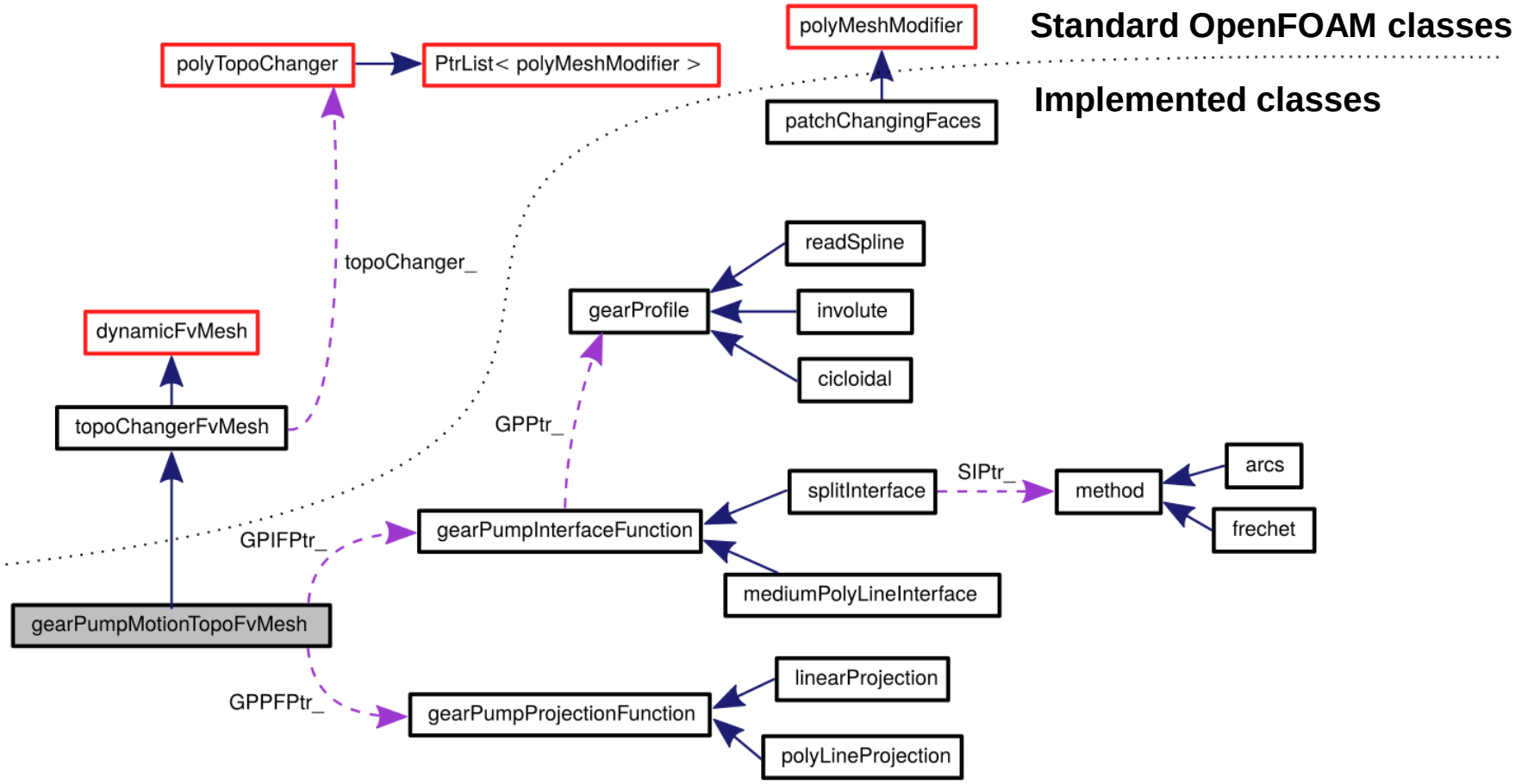


In this case new boundary faces need to be created/deleted

Implementation

Collaboration diagram

Implementation is done in the updated release of OpenFOAM-dev



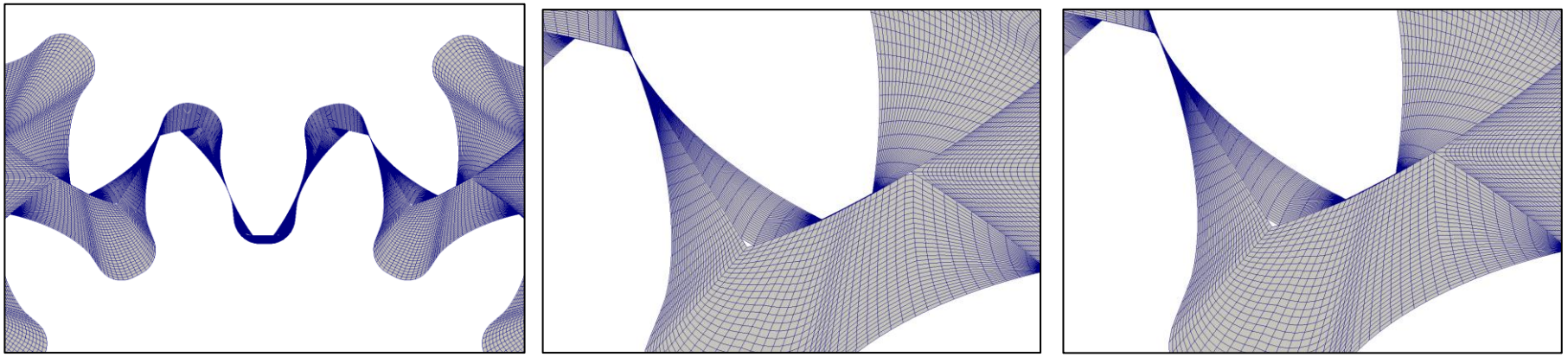
Mesh motion: `dynamicMeshDict`

Mesh generation: `createGearPumpMeshDict` ⟶ *createGearPumpMesh*

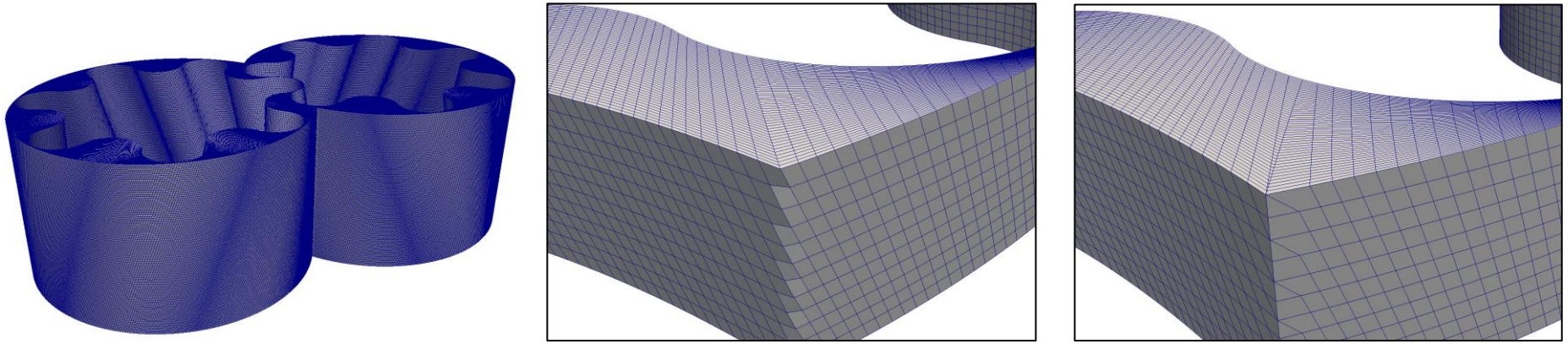
Test cases

Mesh motion examples

2D involute profile

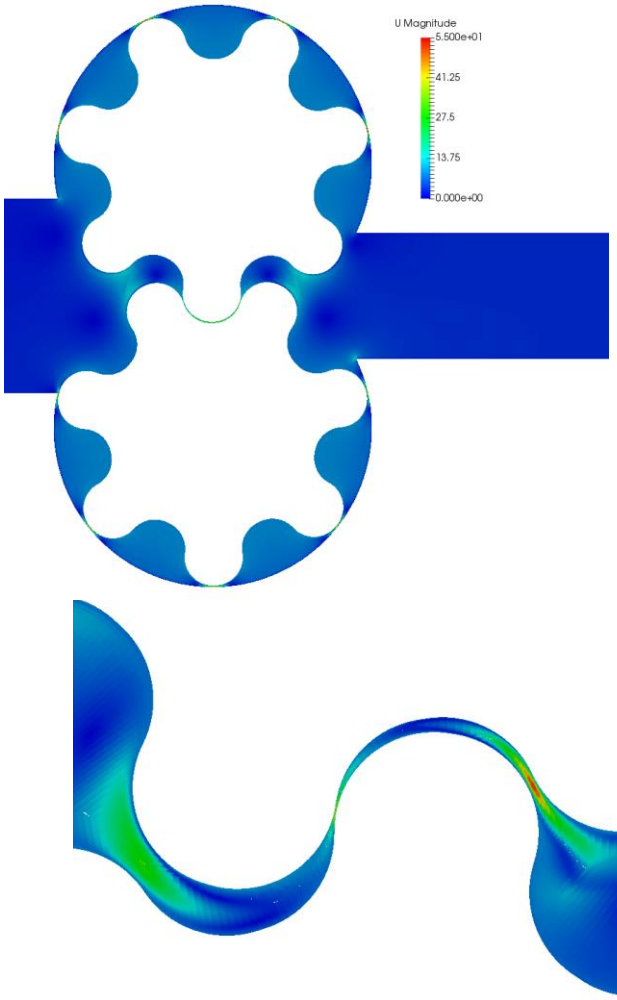


3D Lobe profile

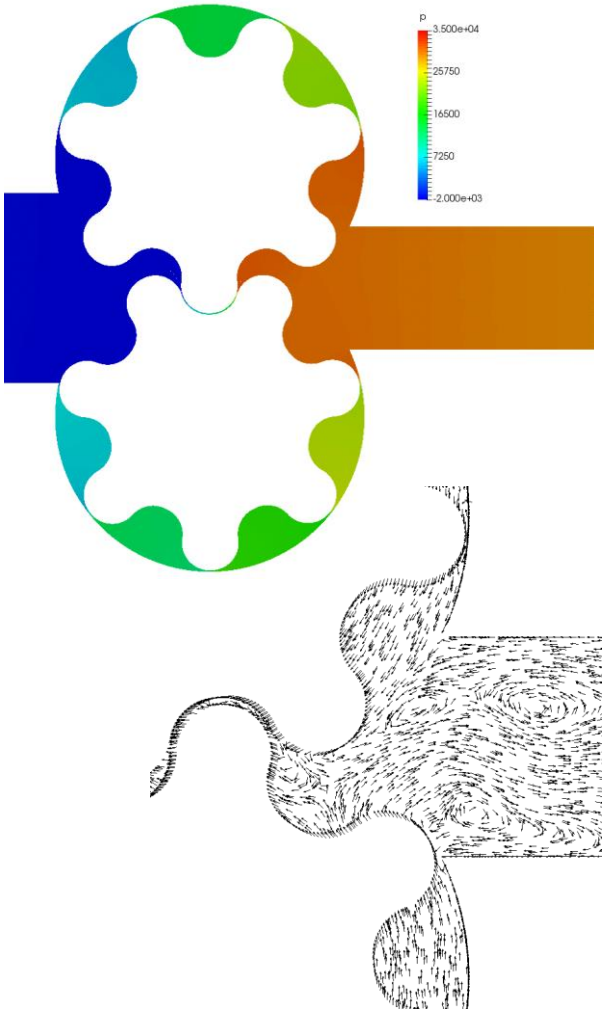


Test cases

Simulation example



2D lobe-profile simulation



Case setup

$\rho = 878\ kg/m^3$
$\mu = 5.97\ 10^{-3} kg/ms$
$tol \approx 100\ \mu m$
$n = 3500\ rpm$
$\Delta p \approx 3000\ psi$

pimpleDyMFoam

(with additional correctPhi iterations)

≈150000 cells
kOmegaSST model

Conclusions and future work

Conclusions

- Development and implementation of a new mesh motion strategy for 2D and 3D simulations of external gear pumps
- Appropriate parallelization and topological changes
- Mesh generation application

Future work

- Run-time mesh smoothing
- Identification of contact region and appropriate treatment (porous media)
- Testing the influence of tolerances and working conditions in 2D-3D cases

Javier Martínez Rubio

javier.martinez@polimi.it