Unione Europea

Repubblica Italiana

Regione Autonoma della Sardegna

Re-meshing strategies in CFD simulations of moving meshes

MANUELA PROFIR
manuela@crs4.it

August 16, 2012

Computing & Research in Sardinia

Approach to Mesh Morphing in CFD framework

Morphing in Computer Graphics

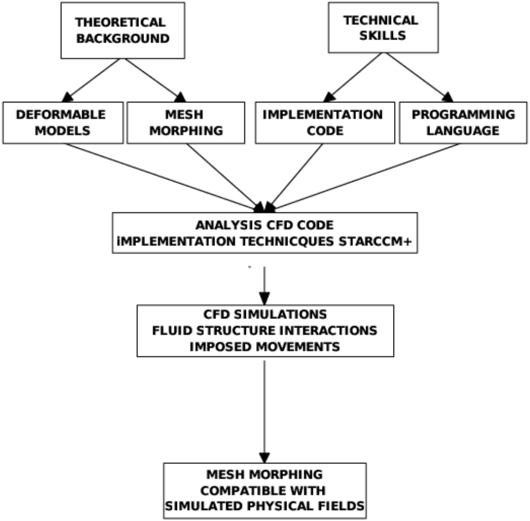
- gradually changes a source shape into a target shape
- medical imaging, scientific visualization, special effects in movies

- Volume mesh = mathematical description of the geometry: vertices, faces, cells
- Surface deformations in moving fluids/solids simulations: warped, self-intersected faces lead to negative volume cells

Research's objectives

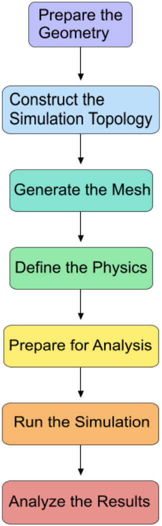
- control mesh deformations
- develop re-meshing strategies
- use Java programming for automation

Research work-flow



Computing & Research in Sardinia

Implementation code: CD-adapco's Star-ccm+



Tools and Novel Concepts

- Numerical algorithms: Cell-based discretization (arbitrary polyhedra), AMG solver, Convergence
- Physical models: Motions (Rigid, Morphing, Solid displacement), Turbulence, Solid Stress
- Flexible mesh manipulation

- Multi-physics, continuum-based modelling
- Separation of physics, geometry and mesh
- Generalized interfaces

- 3D-CAD modeller - design parameters
- Client-Server Architecture - Java Scripting

Mesh Morphing procedure in Star-ccm+

- **Morphing Motion** model (deforming mesh) $\longrightarrow$ **Mesh Morpher**
- Morpher collects control points and specified displacements from **boundaries** $\longrightarrow$ morpher boundary conditions:
  - **Displacement, Grid Velocity, Solid Stress, Floating**.
- Control points x_i and displacements $d(x_i)$ are used to generate an interpolation field:

$$d(x_i) = \sum_{j=1}^{n} \lambda_j \sqrt{\|x_i - x_j\|^2 + c_j^2} + \alpha, \quad \sum_{j=1}^{n} \lambda_j = 0 \quad \Rightarrow \lambda_j, \alpha$$

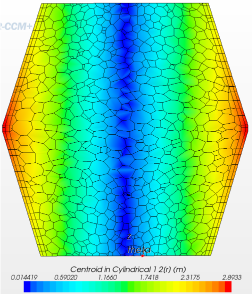
- The interpolation field applies to **all** the mesh vertices:

$$d(x) = \sum_{j=1}^{n} \lambda_j \sqrt{\|x - x_j\|^2 + c_j^2} + \alpha$$

- Final adjustments to mesh vertices on or near boundaries.

Morpher boundary conditions

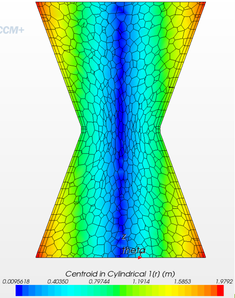
Displacement



$$\delta_r = \begin{cases} 0.00625 x_z, & x_z < 3 \\ 0.00625(6 - x_z), & x_z \geq 3 \end{cases}$$

Grid Velocity

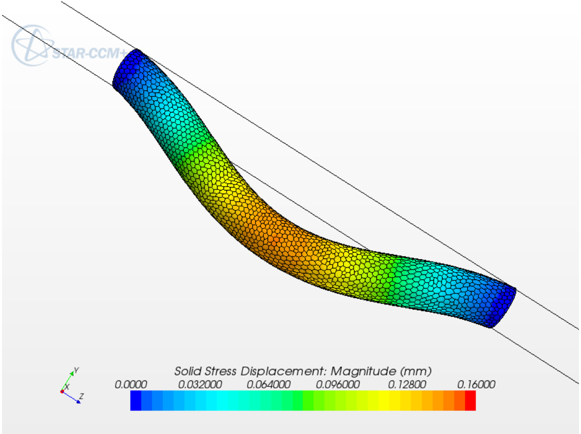
$$v_r = \begin{cases} -0.003 x_z, & x_z < 3 \\ -0.003(6 - x_z), & x_z \geq 3 \end{cases}$$

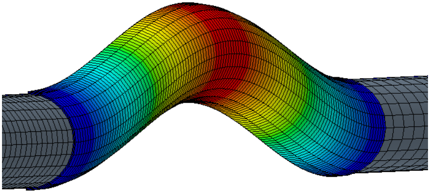


Morpher boundary conditions: Solid Stress in FSI

Pipe deformation/oscillation

- **Morphing**: fluid mesh deforms under external load
- **Solid displacement**: body load applied to the solid pipe





Imposed solid displacement

$$\delta(t) = [0.0, ((0 < z < l)?(2\sin t \sin^2(\frac{z}{l}\pi)) : 0), 0.0]$$

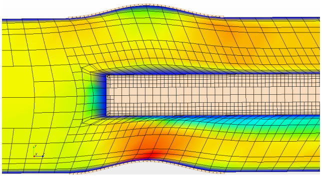
Pipe/water system	Cells	Solid stress	Imposed
Structured-trimmed	30 K	$\delta y = 0.2mm$	$\delta y = 1cm$
Unstructured-poly	15 K	$\delta y = 0.15mm$	$\delta y = 0.5cm$

Computing & Research in Sardinia

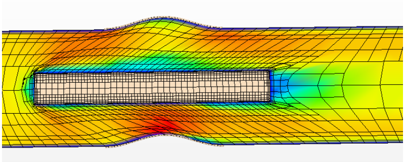
Morphing and rigid motions

- "Piston" subject to a periodical translation motion
- Behaviour of surrounding fluid mesh when the piston is moving?
- Keep under control the expected deformations of the fluid mesh!

$$\delta y = 0.01 \sin t \sin^2(\frac{z}{l}\pi), \quad 0 \leq z \leq l$$

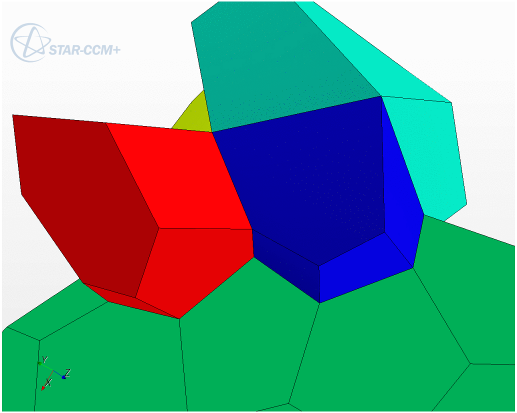


$$v_z(t) = A\omega \sin(\omega t)$$



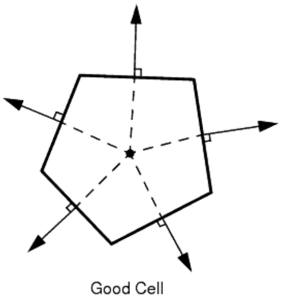
Bad mesh deformations - Negative volume cells

- Large motions determine cells degeneracy! Re-mesh needed!

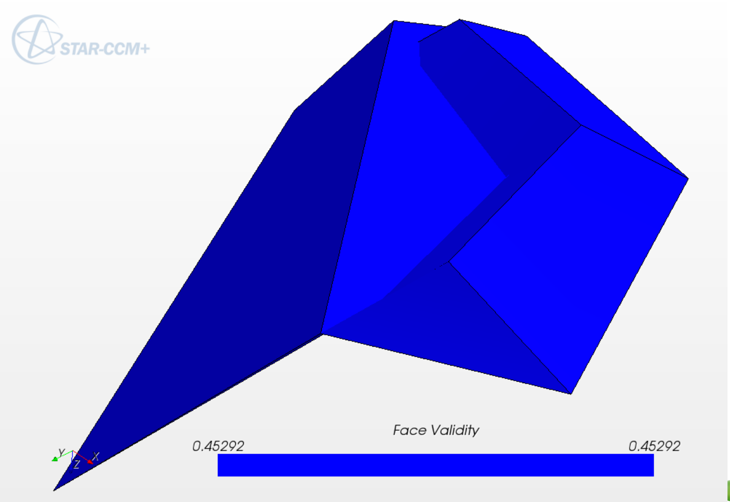


Mesh quality metrics: **Face Validity**

- Face normals must point away from cell centroid.
- Perfect cells: FV = 1. Minimum acceptable value: FV = 0.51.
- Face Validity metric description (reprinted from User Guide [2]):

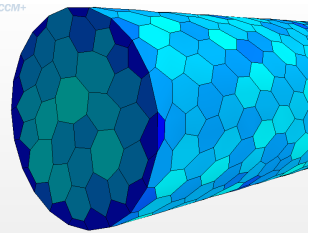


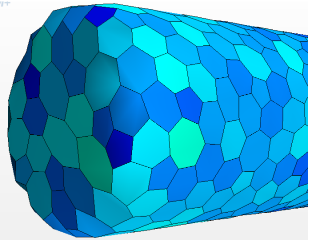
Negative volume cell with low face validity



Re-meshing strategy: surface mesh extraction

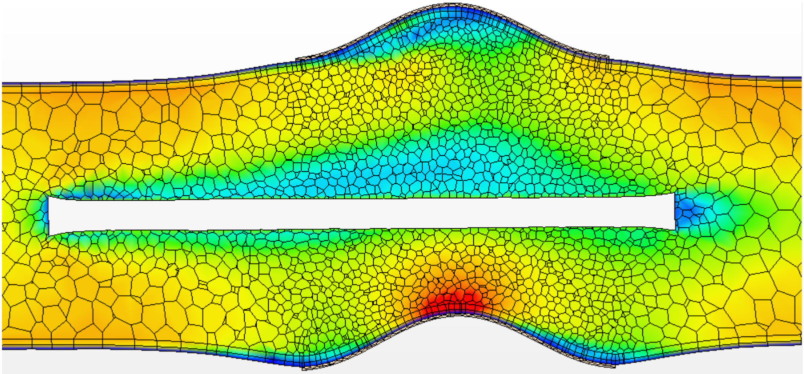
- Problems arising: Non-planar faces loose feature curves!
- Geometry is not preserved!





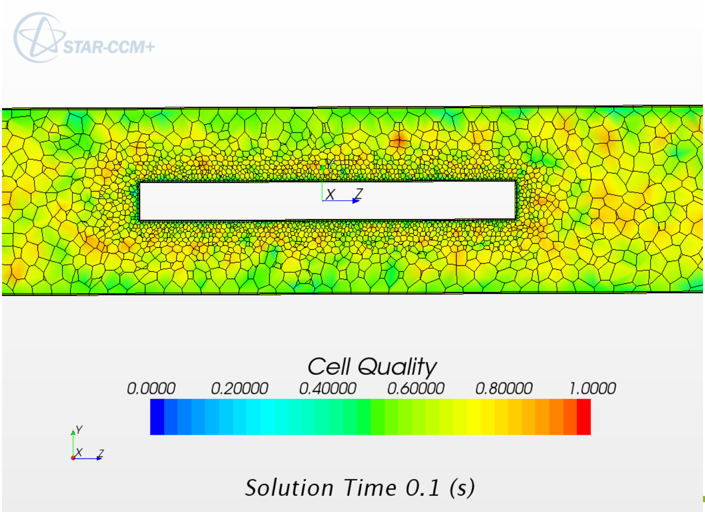
Computing & Research in Sardinia

Re-meshing strategy: surface mesh extraction



Computing & Research in Sardinia

Re-meshing strategy: update CAD geometry



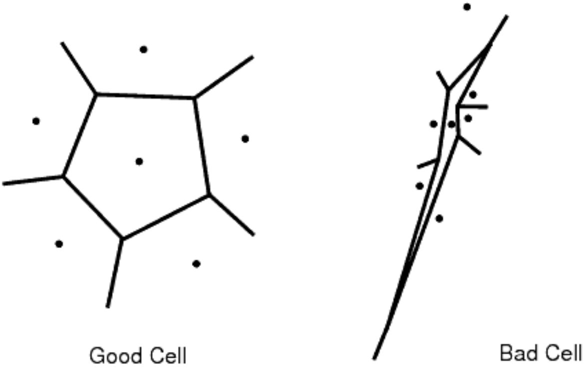
Re-meshing strategy: CAD geometry update

- Automation of re-meshing with external loops in a Java macro:

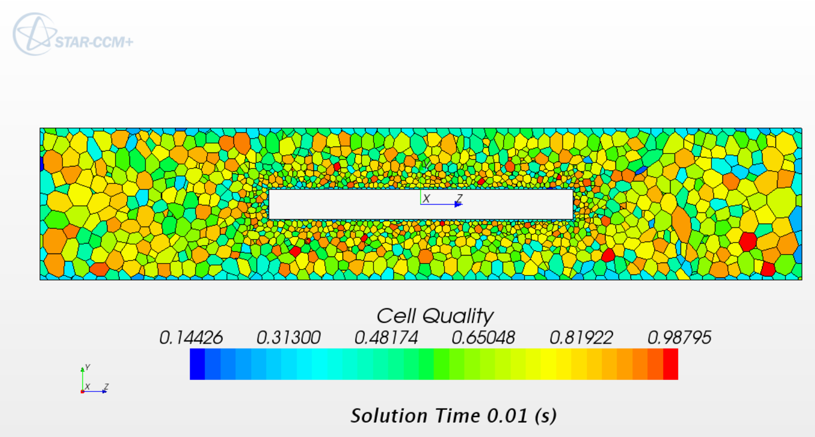
1. Run the simulation for one time step.
2. Control the condition about the mesh quality criterion.
3. If the criterion is satisfied, take one time step forward; otherwise:
4. Regenerate the volume mesh, including:
   I. prepare a report measuring the exact performed displacement;
   II. in the CAD construction, apply a translation to the piston body and expose as design parameter the translation vector;
   III. the macro reads the displacement's value and assigns it to the translation vector, updating the initial position of the piston;
   IV. regenerate the surface and volume meshes, taking the most recent boundaries as starting point (i.e. the updated piston position).

Mesh quality metrics: **Cell Quality**

- Depends on the relative geometric distribution of the centroids of cells neighbours of a face and also on the cell faces orientation
- Perfect cells: CQ = 1.0. Degenerate cells: CQ = 0.
- Cell Quality metric description (reprinted from User Guide [2]):



Translational periodical motion

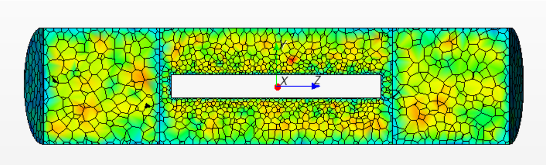


Grid Velocity for Morphing motion

$$v(t) = A\omega \sin(\omega t)$$

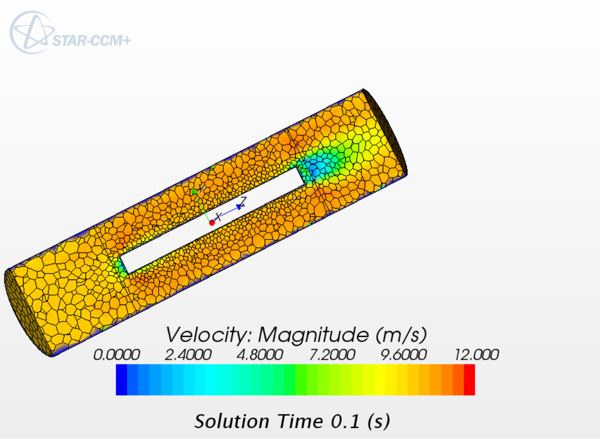
Improved strategy: optimised re-meshing

- Divide domain into three regions, by subtraction, intersection and imprinting boolean operations
- Central region: **Translation motion**
- Lateral regions: **Morphing motion**
- MBC: **Floating** for Wall boundaries, **Grid Velocity** for interfaces
- Update geometry for re-meshing
- Re-mesh only lateral regions



- faster mesh generation
- less re-meshing

Update lateral bodies through translations



- Translate lateral bodies: $\Delta z = p_1 - p_0$
- p_0 = Init. Position
- p_1 = Max. Position of central region

Computing & Research in Sardinia

Extrusion distances: design parameters for up-dating

- Central body: 2-way asymmetric extrusion distances

$$d = d_0 + \Delta z = p_1$$

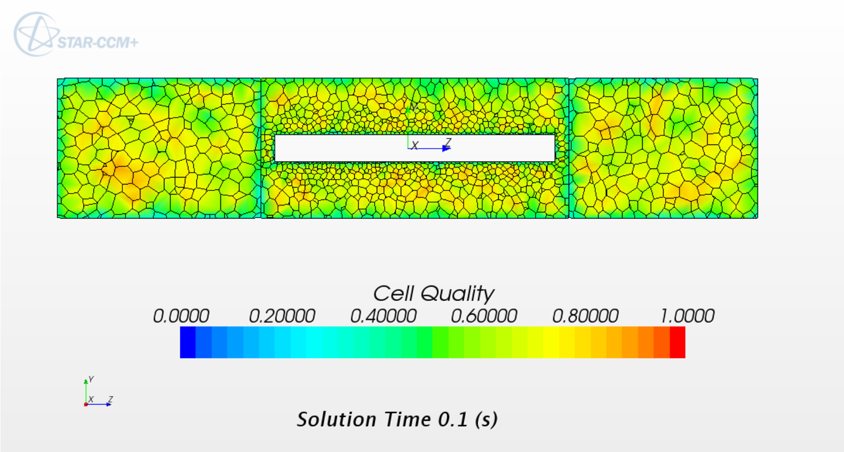
$$d' = d_0 - \Delta z = 2p_0 - p_1$$

Java implementation

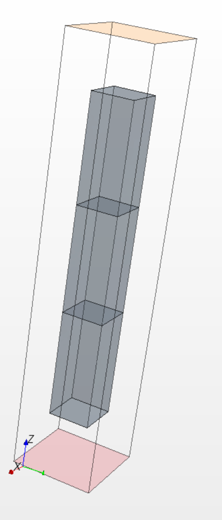
```
ExtrusionMerge extrusionMerge_0 = ((ExtrusionMerge)
cadModel_0.getFeatureManager().getObject("Central"));
extrusionMerge_0.getDistance().setValue(d);
extrusionMerge_0.getDistanceAsymmetric().setValue(d’)
```

Computing & Research in Sardinia

Update lateral bodies with extrusion distances



Morphing and re-meshing compatibility with physics

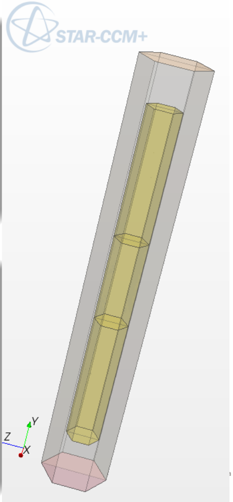


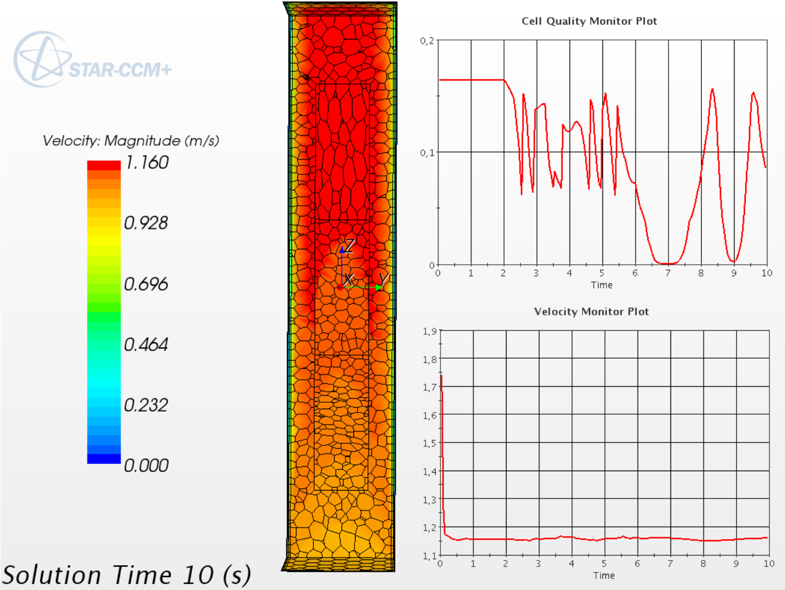
- **Floating** - Sliding interfaces
- **Grid Velocity** - Interfaces with Central region
- **Fixed** - Interfaces with external region

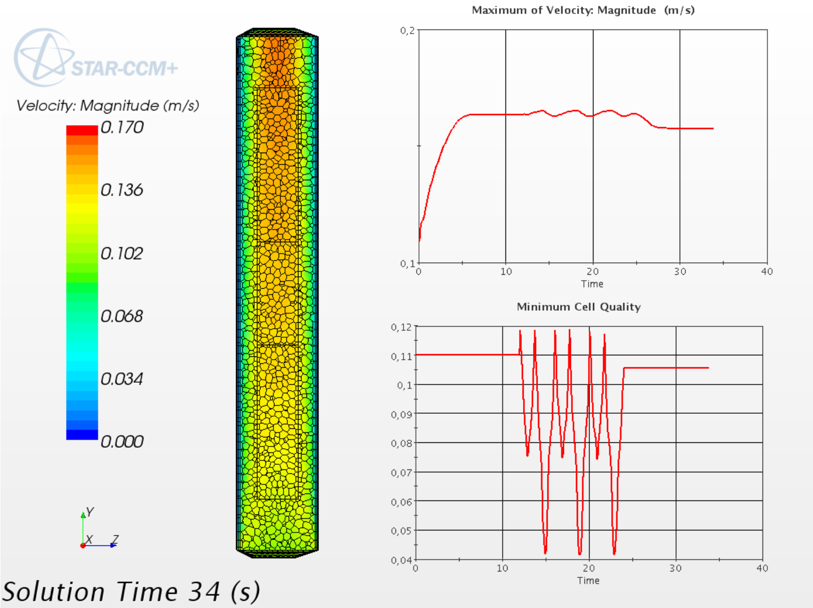
Translation velocity/Grid velocity

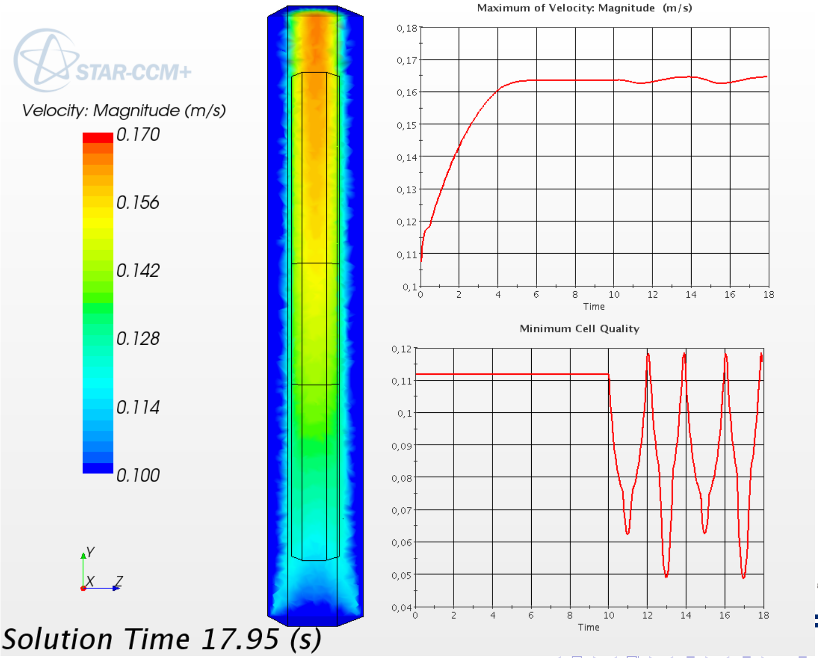
$$v(t) = [0.0, A\frac{\pi}{2} sin(\frac{\pi}{2} t + \frac{\pi}{2}), 0.0],$$

$$A \leqq \Delta z$$

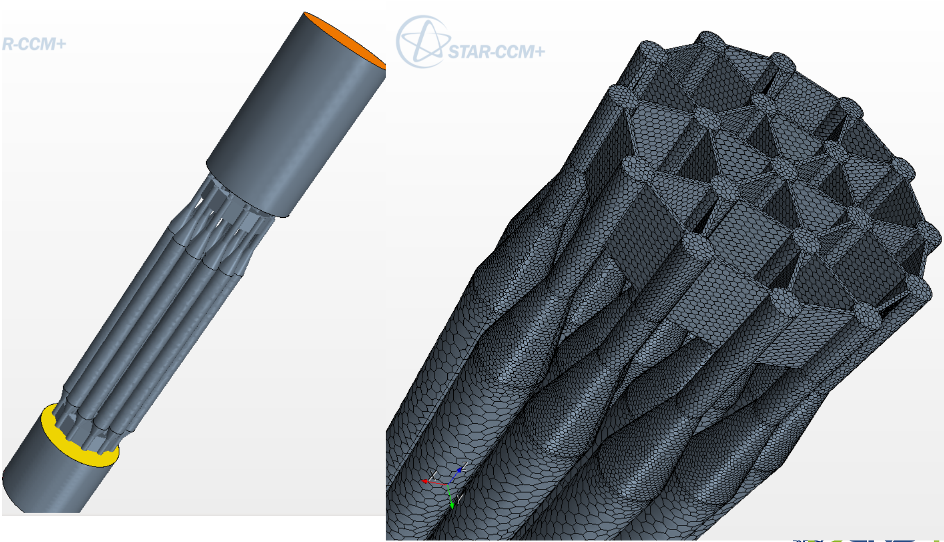



STAR-CCM+
Velocity: Magnitude (m/s)
1.160
0.928
0.696
0.464
0.232
0.000
Solution Time 10 (s)
Z
X
Y
Cell Quality Monitor Plot
0,2
0,1
0
0 1 2 3 4 5 6 7 8 9 10
Time
Velocity Monitor Plot
1,9
1,8
1,7
1,6
1,5
1,4
1,3
1,2
1,1
0 1 2 3 4 5 6 7 8 9 10
Time

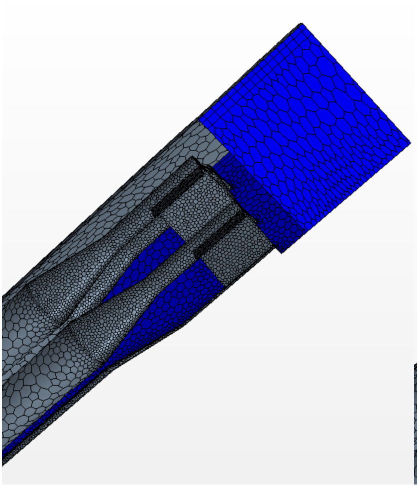

STAR-CCM+
Velocity: Magnitude (m/s)
0.170
0.136
0.102
0.068
0.034
0.000
Y
X
Z
Solution Time 34 (s)
Maximum of Velocity: Magnitude (m/s)
0,2
0,1
0
10
20
30
40
Time
Minimum Cell Quality
0,12
0,11
0,1
0,09
0,08
0,07
0,06
0,05
0,04
0
10
20
30
40
Time

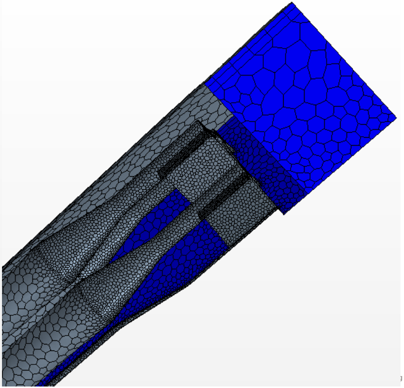

STAR-CCM+
Velocity: Magnitude (m/s)
0.170
0.156
0.142
0.128
0.114
0.100
Y
X
Z
Solution Time 17.95 (s)
Maximum of Velocity: Magnitude (m/s)
0,18
0,17
0,16
0,15
0,14
0,13
0,12
0,11
0,1
0 2 4 6 8 10 12 14 16 18
Time
Minimum Cell Quality
0,12
0,11
0,1
0,09
0,08
0,07
0,06
0,05
0,04
0 2 4 6 8 10 12 14 16 18
Time

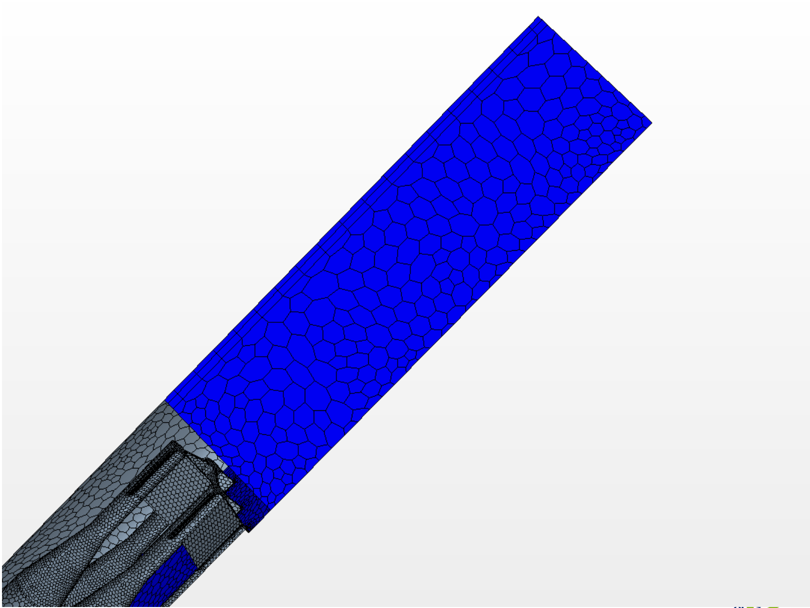
Re-meshing strategies on complex geometries



Surface extraction strategy on complex geometries







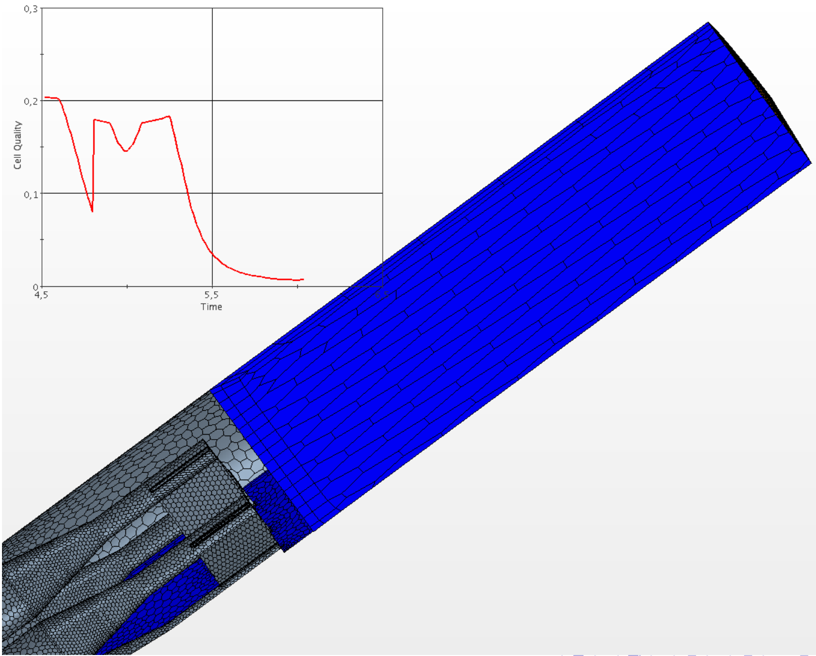
Update CAD with two design parameters

Implementation in Java

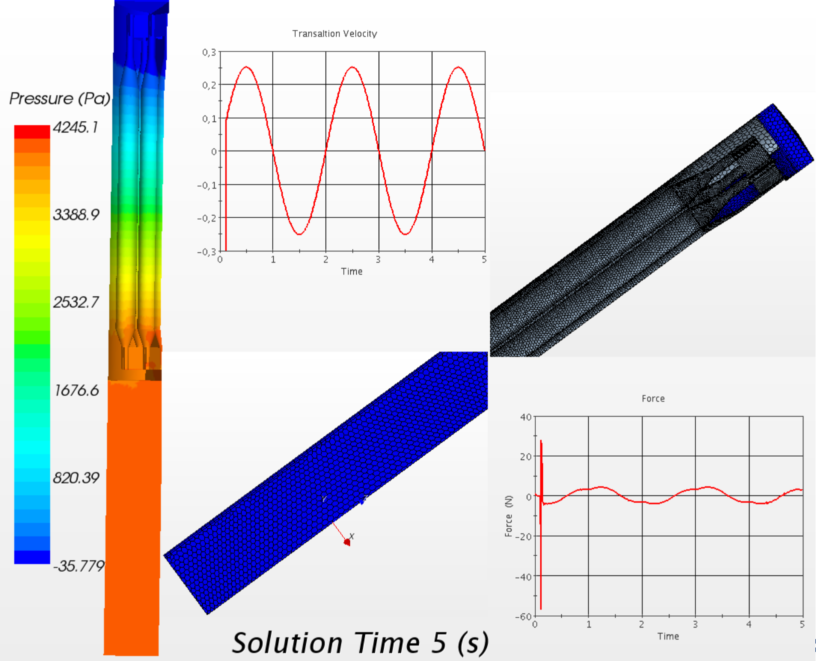
```
double d = maxReport_0.getReportMonitorValue();
ExtrusionMerge extrusionMerge_0 = ((ExtrusionMerge)
cadModel_0.getFeatureManager().getObject("B"));

extrusionMerge_0.getDistance().setValue(d);
extrusionMerge_0.getDistanceAsymmetric()
.setValue(0.34-d);

MoveBodyFeature moveBodyFeature_0 =
((MoveBodyFeature) cadModel_0.getFeatureManager().
getObject("MoveBody 5"));
CadModelCoordinate cadModelCoordinate_0 =
moveBodyFeature_0.getTranslationVector();
cadModelCoordinate_0.setCoordinate(
new DoubleVector(new double[] {0.0, 0.0, d-0.33}));
```



& Research in Sardinia


Pressure (Pa)
4245.1
3388.9
2532.7
1676.6
820.39
-35.779
Transaltion Velocity
0,3
0,2
0,1
0
−0,1
−0,2
−0,3
0
1
2
3
4
5
Time
Force
40
20
0
−20
−40
−60
Force (N)
Time
Solution Time 5 (s)

Conclusions and further work

- Control of the mesh deformations coherently with the imposed displacements.
- Optimised computation/mesh regeneration time, through a selective (region-wise) re-meshing.
- Extend the software's capabilities, by the coupling with Java .
- Automation of the re-meshing procedures.
- Regular development of the fluid flow, even in presence of morphing and re-meshing operations.
- Combine sliding interfaces with moving/deforming domains, maintaining or quickly re-creating good quality mesh.

- Integration of the moving parts in a larger closed loop
- Coupling with the physics of nuclear experimental facilities
- Reproduce the movement of the control/safety rods system in relation to applied physical forces (fluid drag, buoyancy, gravity).

Acknowledgments & References

RAS

This project is supported by RAS (Regione Autonoma della Sardegna) through a grant co-funded by PO Sardegna FSE 2007-2013, L.R.7/2007 *Promozione della ricerca scientifica e dell'innovazione tecnologica in Sardegna*.

CRS4

- *Energy & Environment* Program
- Process Engineering and Combustion
- Vincent Moreau

- CD-adapco website, http://www.cd-adapco.com/ (Last access March 2012)
- USER GUIDE STAR-CCM+ Version 7.02.011.

- ALEXA M. *Recent Advances in Mesh Morphing*, Computer Graphics Forum (9/2002).
- BOER A., SCHOOT M.S., BIJL H., *Mesh morphing based on radial function interpolation*, Elsevier, 2007.
- HARDY, R. L., *Theory and applications of the multi-quadric biharmonic method*, Comput. Math. Applic. (1990), Vol. 19, pp. 163-208
- NEALEN, MULLER, BOXERMAN, CARLSON *Physically Based Deformable Models in Computer Graphics*, Eurographics 2005
- PROFIR M. M., *Mesh morphing implementation methods with examples*, Atti Semin. Mat. Fis. Univ. Modena Reggio Emilia pages 141-157 vol. 57, 2010
- PROFIR M. M., *Automated moving mesh techniques and re-meshing strategies in CFD applications using morphing and rigid motions*, CRS4 Technical Report, http://publications.crs4.it/pubdocs/2012/Pro12c.

Computing & Research in Sardinia