

HyperCFD
Fertinaz Yazılım Ltd.

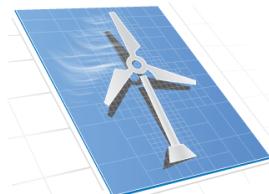
OpenFOAM Workshop Teknopark İstanbul

03 Meshing
January 2017



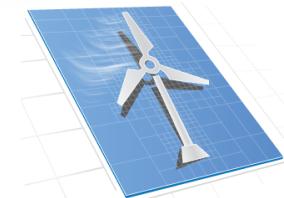
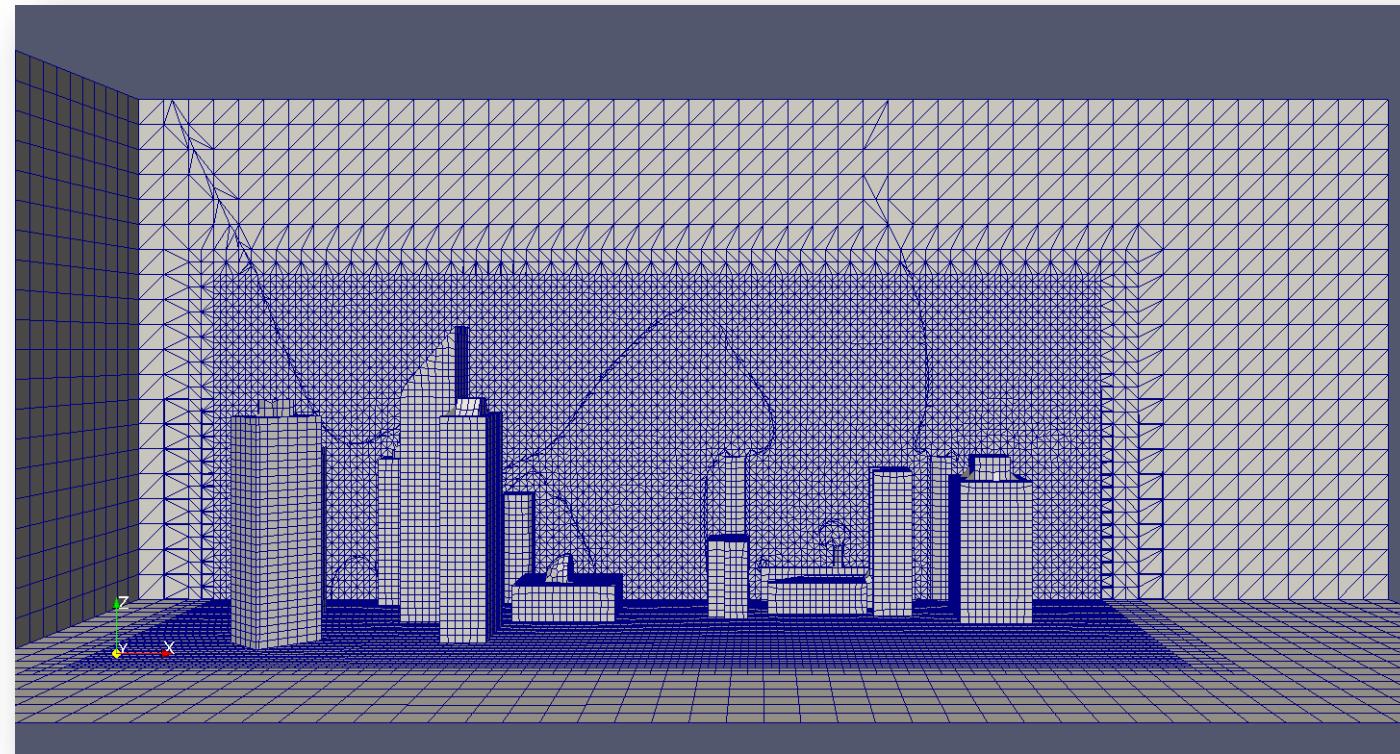
HyperCFD – Disclaimer

This offering is not approved or endorsed by OpenCFD Limited, the producer of the OpenFOAM software and owner of the OpenFOAM and OpenCFD trademarks.



HyperCFD – OpenFOAM Meshing

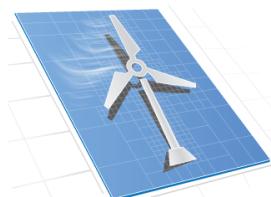
- Meshing is indeed very important
- Good mesh may lead good results
- Bad mesh definitely leads bad results



HyperCFD – OpenFOAM Meshing

- Mesh generators
 - blockMesh
 - snappyHexMesh
- Mesh converters

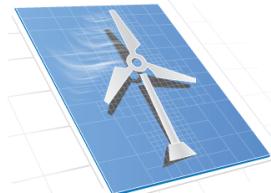
```
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities/mesh/conversion
fertinaz@fertinaz-M4800:~$ OF40
fertinaz@fertinaz-M4800:~$ cd $FOAM → Using double tab here to see environment variables starting with FOAM
$FOAM_APP           $FOAM_INST_DIR    $FOAM_RUN          $FOAM_SITE_LIBBIN   $FOAM_USER_APPBIN
$FOAM_APPBIN        $FOAM_JOB_DIR     $FOAM_SETTINGS    $FOAM_SOLVERS      $FOAM_USER_LIBBIN
$FOAM_ETC           $FOAM_LIBBIN     $FOAM_SIGFPE       $FOAM_SRC         $FOAM_UTILITIES
$FOAM_EXT_LIBBIN    $FOAM_MPI        $FOAM_SITE_APPBIN $FOAM_TUTORIALS   $FOAM_HEX_MESH
fertinaz@fertinaz-M4800:~$ cd $FOAM_UTILITIES
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities$ ls
mesh  miscellaneous  parallelProcessing  postProcessing  preProcessing  surface  thermophysical
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities$ cd mesh/
advanced/  conversion/  generation/  manipulation/
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities$ cd mesh/conversion/
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities/mesh/conversion$ ls
ansysToFoam        fluentMeshToFoam  gambitToFoam      mshToFoam        sammToFoam      vtkUnstructuredToFoam
cfx4ToFoam         foamMeshToFluent  gmshToFoam      netgenNeutralToFoam star3ToFoam      writeMeshObj
datToFoam          foamToStarMesh   ideasUnvToFoam  Optional        star4ToFoam
fluent3DMeshToFoam foamToSurface   kivaToFoam      plot3dToFoam   tetgenToFoam
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities/mesh/conversion$ █
```



HyperCFD – OpenFOAM Meshing

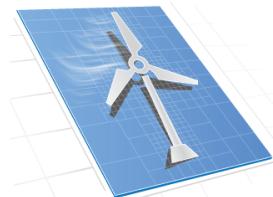
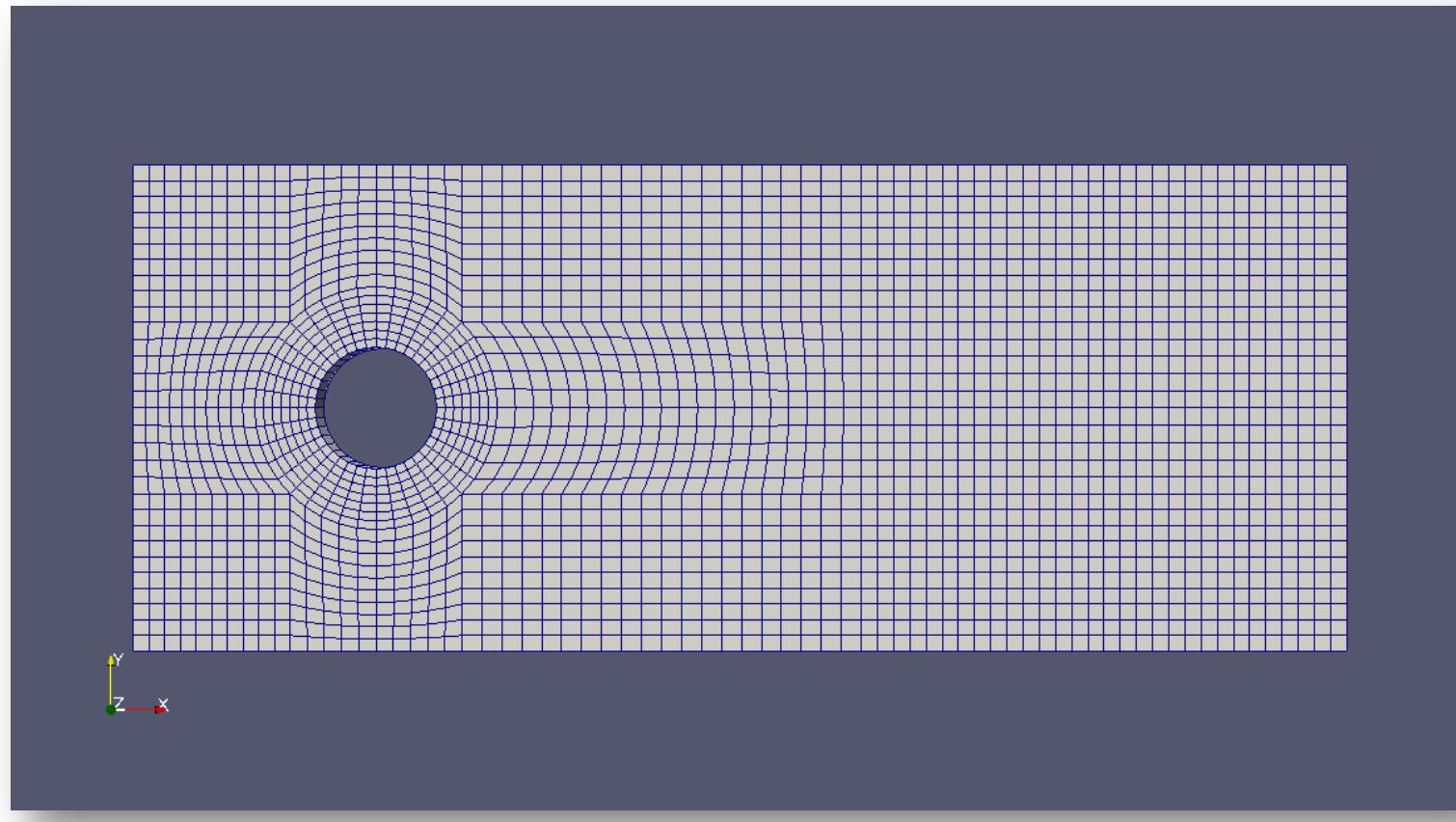
- Mesh manipulation

```
fertinaz@fertinaz-M4800: ~/OpenFOAM/OpenFOAM-4.0/applications/utilities/mesh/manipulation
mesh miscellaneous parallelProcessing postProcessing preProcessing surface thermophysical
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities$ cd mesh/
advanced/ conversion/ generation/ manipulation/
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities$ cd mesh/conversion/
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities/mesh/conversion$ ls
ansysToFoam      fluentMeshToFoam  gambitToFoam    mshToFoam      sammToFoam    vtkUnstructuredToFoam
cfx4ToFoam       foamMeshToFluent gmshToFoam     netgenNeutralToFoam star3ToFoam   writeMeshObj
datToFoam        foamToStarMesh   ideasUnvToFoam Optional      star4ToFoam
fluent3DMeshToFoam foamToSurface   kivaToFoam    plot3dToFoam  tetgenToFoam
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities/mesh/conversion$ cd ../
advanced/ conversion/ generation/ manipulation/
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities/mesh/conversion$ cd ../manipulation/
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities/mesh/manipulation$ ls
attachMesh      deformedGeom      mirrorMesh      orientFaceZone setSet      stitchMesh
autoPatch       flattenMesh       moveDynamicMesh polyDualMesh  setsToZones  subsetMesh
checkMesh       insideCells       moveEngineMesh  refineMesh    singleCellMesh topoSet
createBaffles   mergeMeshes      moveMesh       renumberMesh  splitMesh   transformPoints
createPatch     mergeOrSplitBaffles objToVTK      rotateMesh   splitMeshRegions zipUpMesh
fertinaz@fertinaz-M4800:~/OpenFOAM/OpenFOAM-4.0/applications/utilities/mesh/manipulation$ █
```



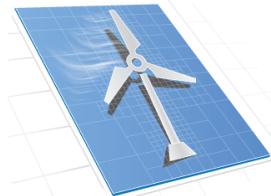
HyperCFD – OpenFOAM Meshing: blockMesh

- `blockMesh`: Utility for generating simple meshes of blocks of hexahedral cells



HyperCFD – OpenFOAM Meshing: blockMesh

- blockMesh
 - Input parameters are in
 - system/blockMeshDict
 - Results are stored in different files
 - constant/polyMesh/boundary: List of patches
 - constant/polyMesh/faces: Ordered list of points
 - constant/polyMesh/neighbour: List of neighbour cell labels
 - constant/polyMesh/owner: List of owner cell labels
 - constant/polyMesh/points: List of point locations in 3D space

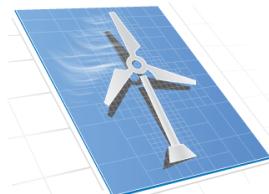


HyperCFD – OpenFOAM Meshing: blockMesh

- blockMesh: How the blockMeshDict file looks

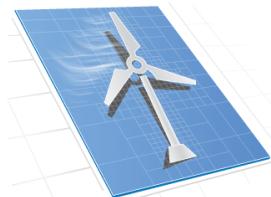
```
1 /*----- C++ -----*/
2 | ====== / Field      | OpenFOAM: The Open Source CFD Toolbox
3 | \  / Operation    | Version: 4.0
4 | \ / And           | Web: www.OpenFOAM.org
5 | \|/ Manipulation |
6 /*
7 */
8 FoamFile
9 {
10    version      2.0;
11    format       ascii;
12    class        dictionary;
13    object       blockMeshDict;
14 }
15 // * * * * *
16
17 convertToMeters 1;
18
19 vertices
20 (
21    ( 700000 4195000 400)
22    ( 750000 4195000 400)
23    ( 750000 4235000 400)
24    ( 700000 4235000 400)
25    ( 700000 4195000 6000)
26    ( 750000 4195000 6000)
27    ( 750000 4235000 6000)
28    ( 700000 4235000 6000)
29
30 );
31
32 blocks
33 (
34    hex (0 1 2 3 4 5 6 7) (125 50 14) simpleGrading (1 1 1)
35 );
36
```

```
37 edges
38 (
39 );
40
41 boundary
42 (
43    outlet
44    {
45        type patch;
46        faces
47        (
48            (2 6 5 1)
49        );
50    }
51    sides
52    {
53        type patch;
54        faces
55        (
56            (1 5 4 0)
57            (3 7 6 2)
58        );
59    }
60    inlet
61    {
62        type patch;
63        faces
64        (
65            (0 4 7 3)
66        );
67    }
68    ground
69    {
70        type wall;
71        faces
72        (
73            (0 3 2 1)
74        );
75    }
76    top
77    {
78        type patch;
79        faces
80        (
81            (4 5 6 7)
82        );
83    }
84 );
85
86 mergePatchPairs
87 (
88 );
89
90
91 // *****
```



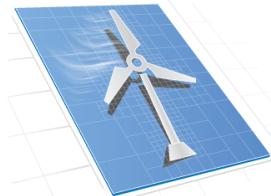
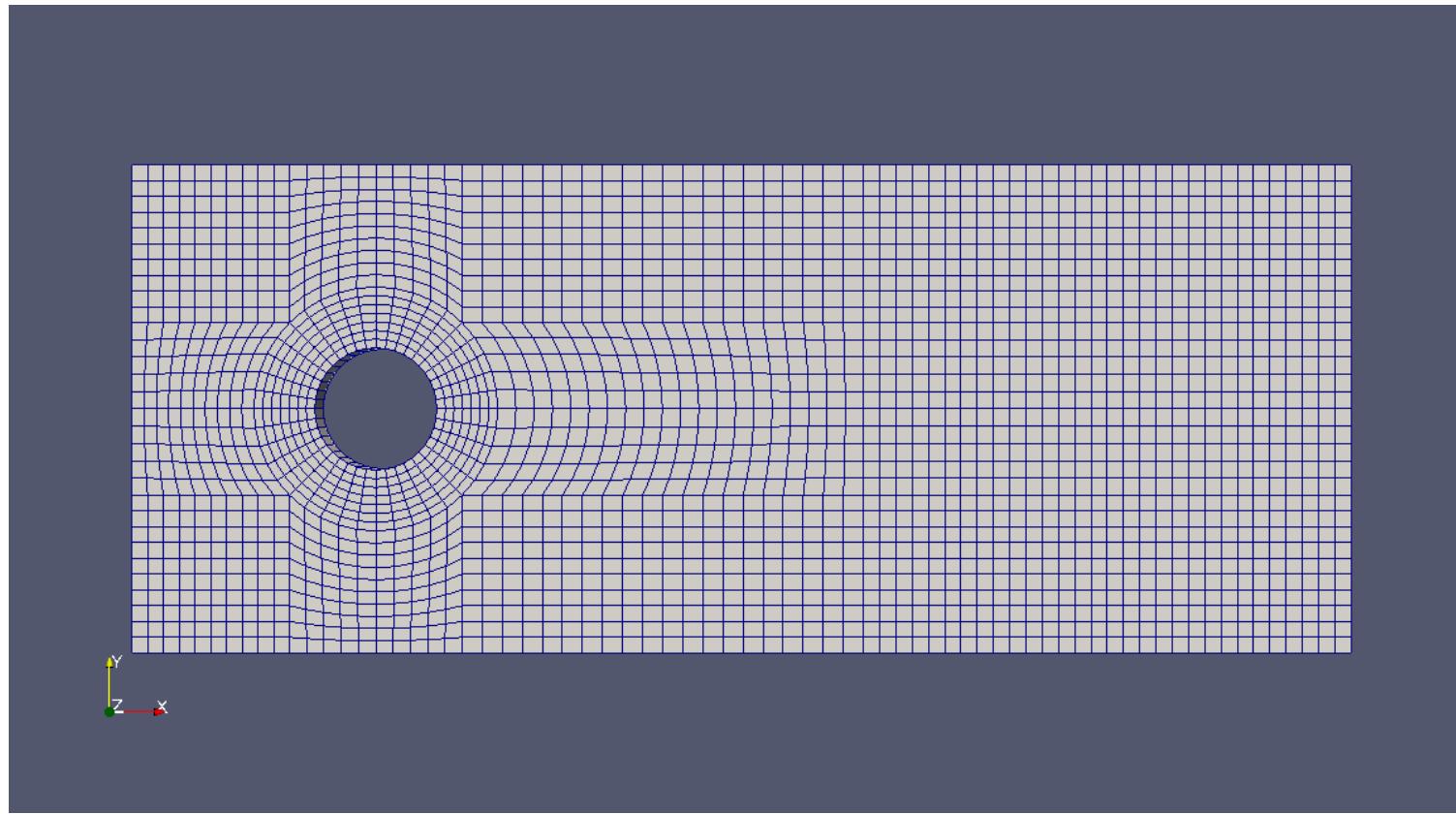
HyperCFD – OpenFOAM Meshing: blockMesh

- `blockMeshDict` can be parameterized using `m4`
- `m4` is a very handy GNU preprocessor very handy however might be a bit difficult to understand if one is reading it for the first time. There are nice documents about it on the internet.



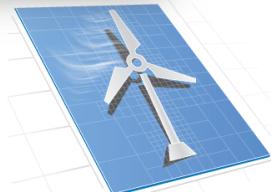
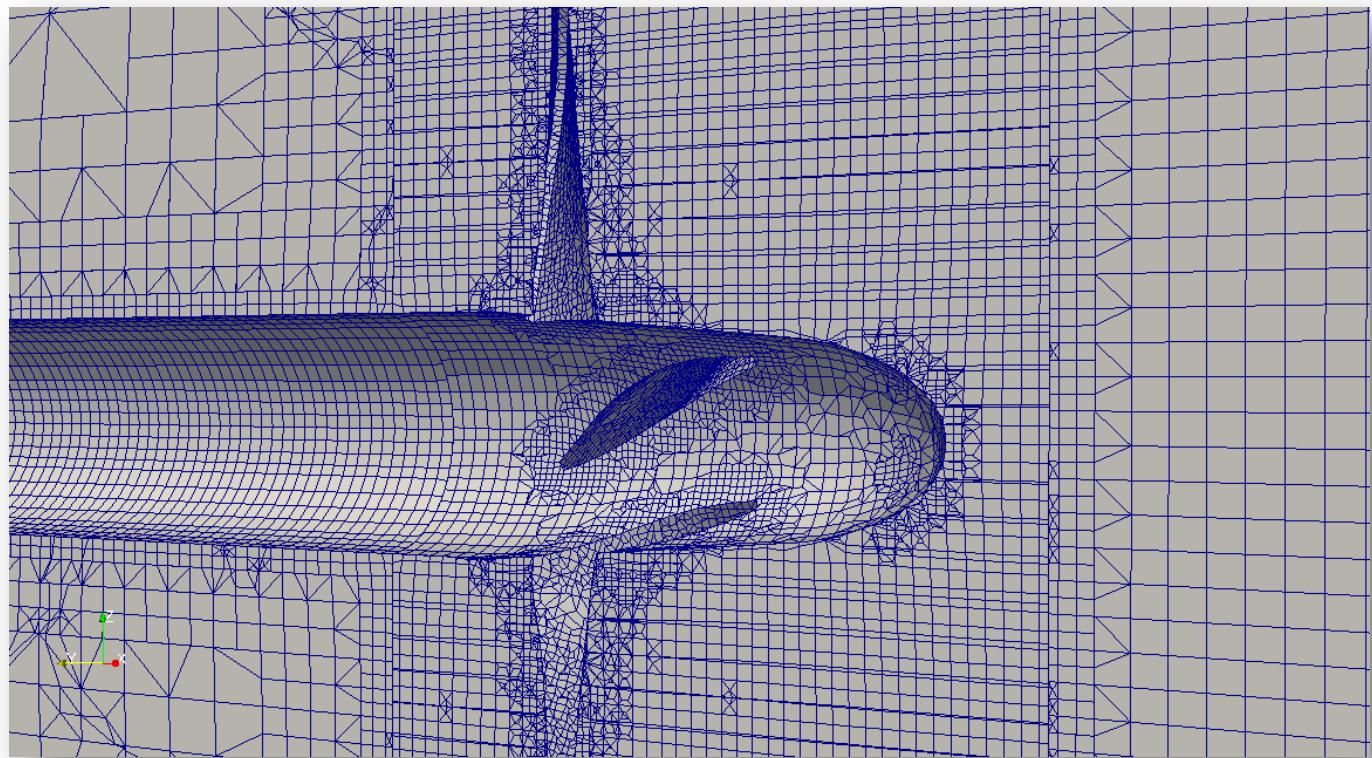
HyperCFD – OpenFOAM Meshing: snappyHexMesh

- `snappyHexMesh`: Utility for generating 3D meshes containing hex and split-hex automatically from triangulated surface geometries



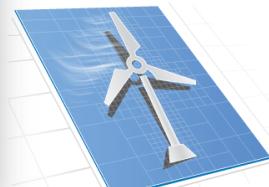
HyperCFD – OpenFOAM Meshing: snappyHexMesh

- snappyHexMesh:
 - Comes after blockMesh
 - Reads input from snappyHexMeshDict which is located in system under the case directory
 - Consists of 3 steps
 - Castellation
 - Surface snapping
 - Boundary layers



HyperCFD – OpenFOAM Meshing: snappyHexMesh

- snappyHexMesh : How snappyHexMeshDict file is configured

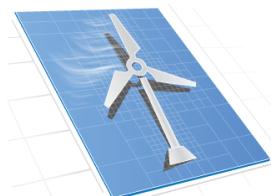


The image shows a screenshot of a code editor displaying an OpenFOAM snappyHexMeshDict file. The file contains configuration parameters for mesh generation, including geometry definitions for turbines and a terrain mesh.

```
*snappyHexMeshDict (~/OpenFOAM/fertinaz-4.0/run/turbineSiting...turbineSiting.yahyali.basis.coar
[OpenFOAM] Open Save Undo Redo Cut Copy Paste Find Replace
*snappyHexMeshDict x
1 /*----- C++ -----*/
2 // Field Operation OpenFOAM: The Open Source CFD Toolbox
3 // And Web: www.OpenFOAM.org
4 // Manipulation
5
6 FoamFile
7 {
8     version    2.0;
9     format      ascii;
10    class       dictionary;
11    object      snappyHexMeshDict;
12 }
13
14 // * * * * *
15
16 // Which of the steps to run
17 castellatedMesh true;
18 snap         true;
19 addLayers    false;
20
21
22
23
24 // Geometry. Definition of all surfaces. All surfaces are of class
25 // searchableSurface.
26 // Surfaces are used
27 // - to specify refinement for any mesh cell intersecting it
28 // - to specify refinement for any mesh cell inside/outside/near
29 // - to 'snap' the mesh boundary to the surface
30 geometry
31 {
32     windTurbine1
33     {
34         type searchableBox;
35         min (721571 4226122 1506);
36         max (721871 4226422 1688);
37     }
38
39     windTurbine2
40     {
41         type searchableBox;
42         min (721813 4226174 1513);
43         max (722286 4226474 1695);
44     }
45
46     windRegion_allTurbines
47     {
48         type searchableBox;
49         min (710000 4197500 200);
50         max (740000 4232500 3000);
51     }
52
53     terrain.ofw.stl
54     {
55         type triSurfaceMesh;
56         name terrain;
57     }
58 }
```

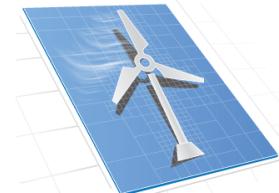
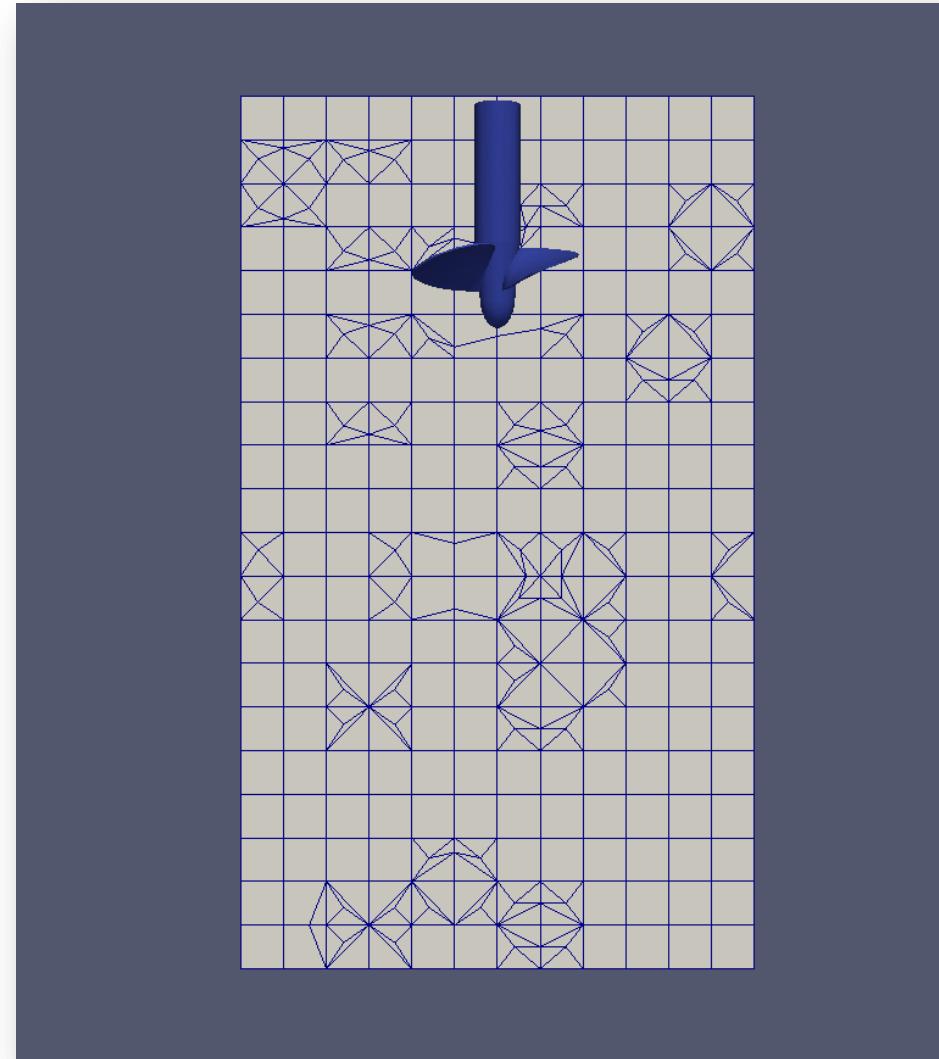
HyperCFD – OpenFOAM Meshing: snappyHexMesh

- castellatedMeshControls
 - maxLocalCells
 - maxGlobalCells
 - nCellsBetweenLevels
 - refinementSurfaces
 - refinementRegions
- snapControls
 - nSmoothPatch
 - Tolerance
 - nSolvIter
 - nRelaxIter
- addLayerControls
 - relativeSizes
 - expansionRatio
 - finalLayerThickness
 - nGrow
 - featureAngle
 - nRelaxIter
 - nSmoothSurfaceNormals
 - nSmoothNormals
 - nSmoothThickness



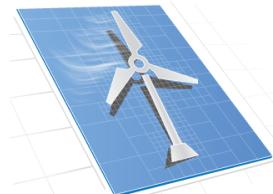
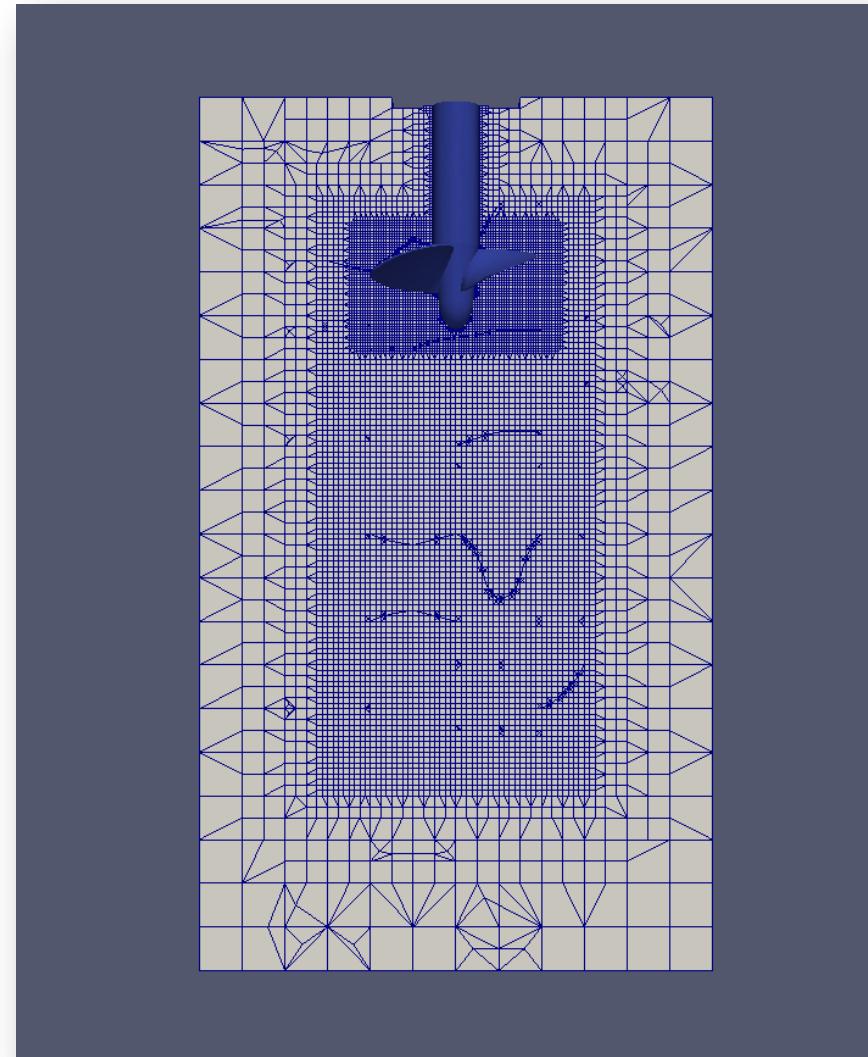
HyperCFD – OpenFOAM Meshing Workflow

- A sample meshing procedure
 - blockMesh with propellers
 - Structured hex cells
 - An initial background grid is compulsory before snappy
 - Cell aspect ratio should be around 1 at least near surfaces otherwise snapping may fail



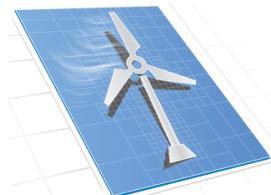
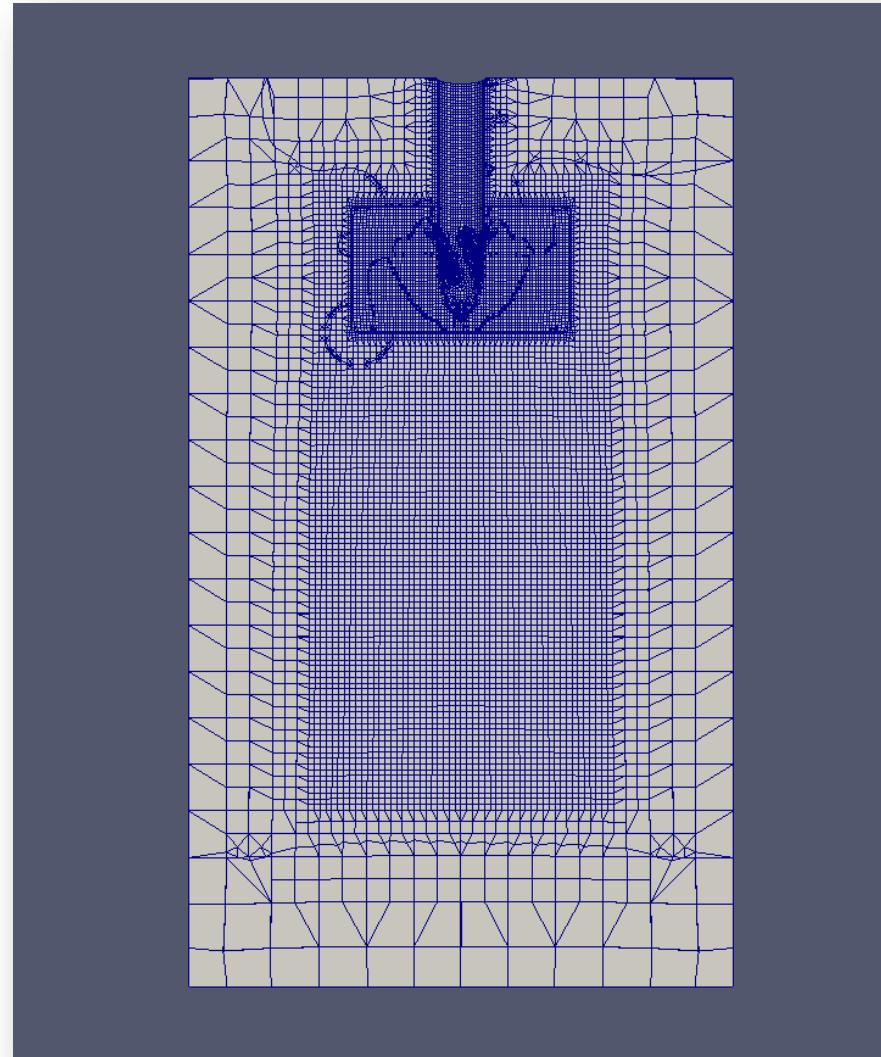
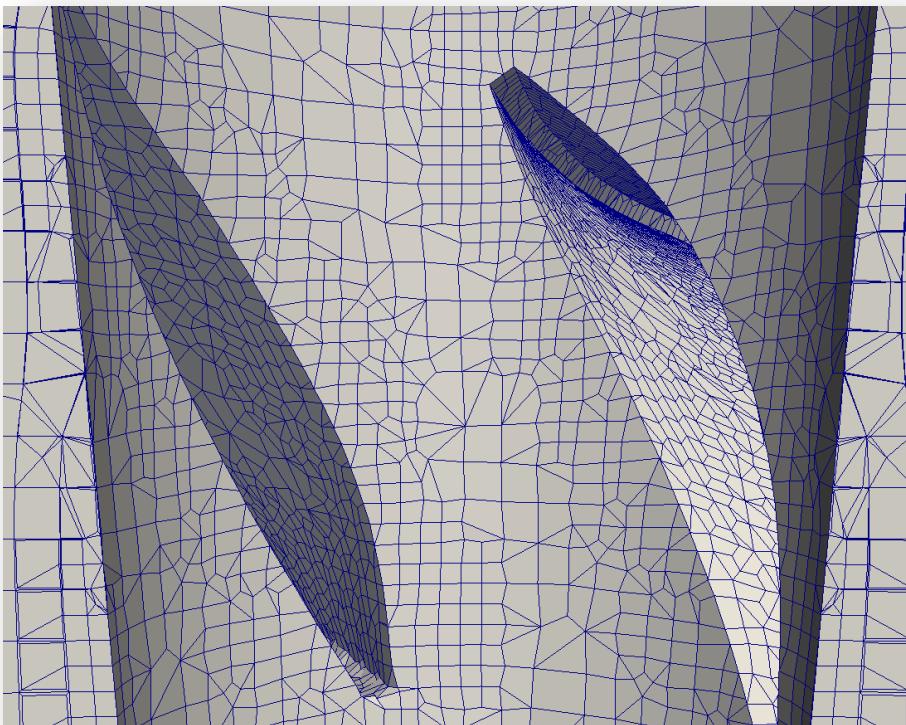
HyperCFD – OpenFOAM Meshing Workflow

- A sample meshing procedure
 - snappyHexMesh after blockMesh
 - Only castellation
 - Regions of interest are refined



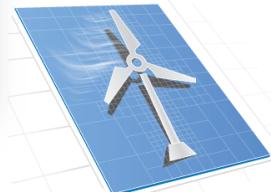
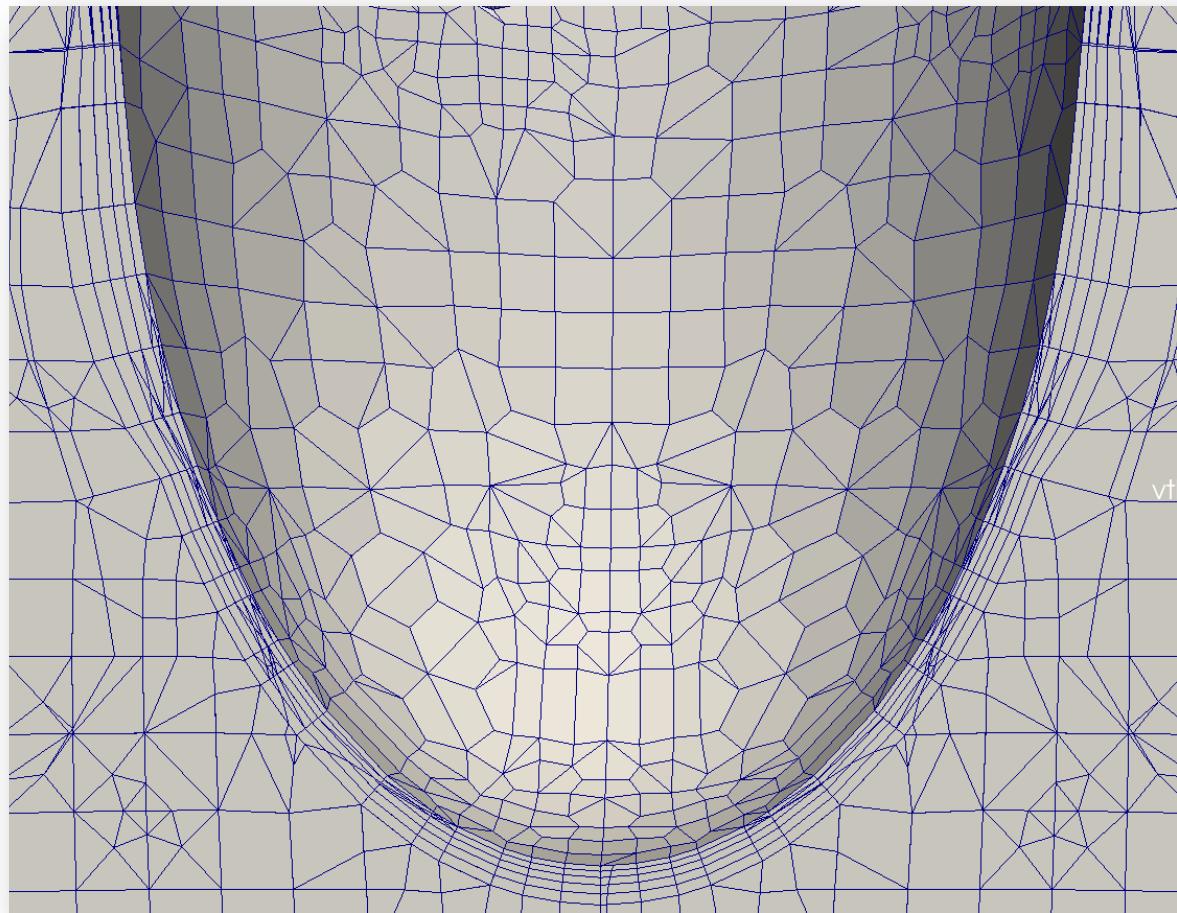
HyperCFD – OpenFOAM Meshing Workflow

- A sample meshing procedure
 - snappyHexMesh after blockMesh
 - Castellation + snapping



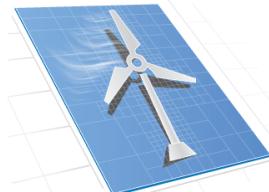
HyperCFD – OpenFOAM Meshing Workflow

- A sample meshing procedure
 - snappyHexMesh after blockMesh
 - Boundary layer addition
 - 5 layers are added around the propellerTip patch
 - Expansion ratio is 1.2



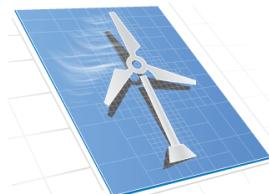
HyperCFD – OpenFOAM Mesh Quality Control

- Mesh quality criteria
 - Non-Orthogonality: Measure of angular deviation between a line vector connects two cell centers and the vector at the center of the connecting face
 - In OpenFOAM nonorthogonality is scaled between 0 to 90. 0 means a perfectly orthogonal mesh while 90 means re-meshing should be considered seriously
 - Usually any value above 70 is considered bad and above 80 is not tolerable
 - Cell skewness: Measure of distance between a line connects two cell centers and the center of the connecting face
 - Same scale as non-orthogonality
 - For these two types of issues OpenFOAM uses correction algorithms. It is usually very difficult to generate industrial meshes with very low nonorthogonalities, therefore you can either avoid it if it is around 60 or consider applying additional `nonOrthogonalCorrectors` which significantly increases the computational time
 - Negative volume cells: Even one of these can yield divergence. Strictly avoid them



HyperCFD – OpenFOAM Mesh Quality Control

- OpenFOAM command: `checkMesh`
 - Reads the `polyMesh` directory
 - Beware in parallel cases. If you have a decomposed case, and want to run `checkMesh` make sure that you run parallel `checkMesh`. Otherwise serially executed `checkMesh` ends up reading the `polyMesh` directory under the case's root which will produce irrelevant results.
 - Complains if there are any nonorthogonal, skewed, negative volume cells.
 - It complains too much
 - You can visualize them using `foamToVTK` as we did
 - Usually you can ignore a few non-orthogonal or skewed cells
 - But you can never ignore a negative volume cell or `wrongOrientedFaces`



HyperCFD – OpenFOAM Meshing Remarks

- Closing remarks:
 - Start with a sample geometry from OpenFOAM tutorials when possible
 - Try to ignore sharp edges on your geometry
 - Make a manual check after each step on ParaView
 - Run also `checkMesh` to be sure
 - Be careful with the `blockMesh` settings, especially total cells and aspect ratio
 - A moderate size `blockMesh` can lead extremely fine final meshes
 - A finer mesh does not always mean a better mesh
 - Try to follow log files
 - Use `-overwrite` option when you're sure what you're doing

