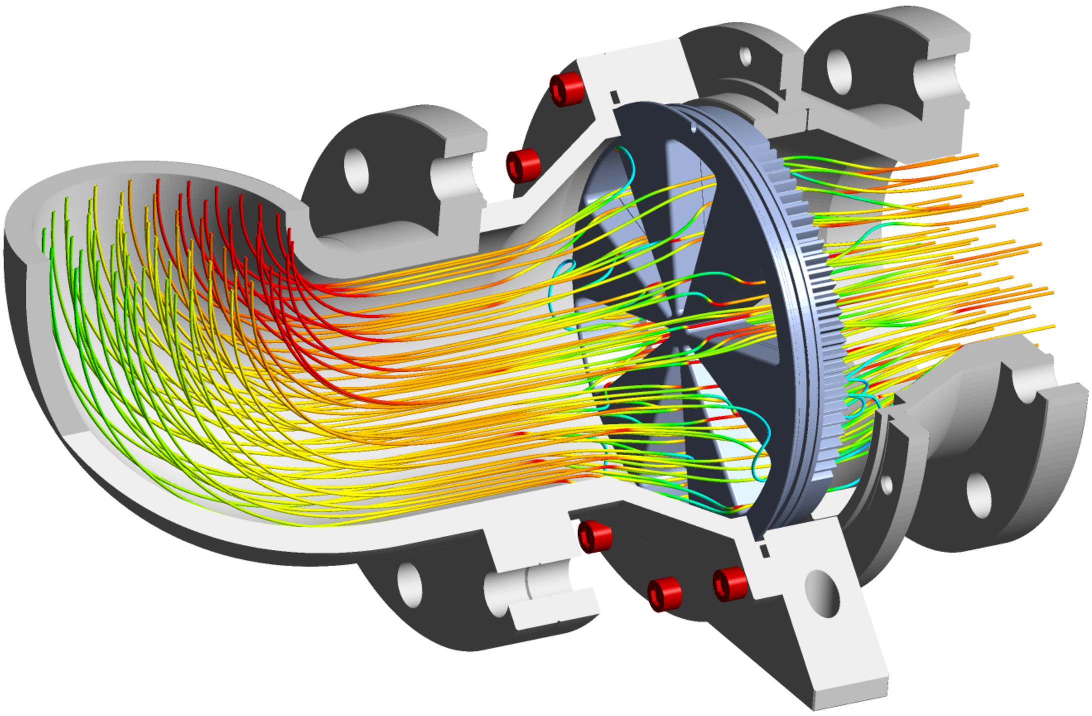
CFD Mapper

Manual (v1.0.0)



Michal Galuszka

December 2020

Contents

1 Introduction **2**

2 Getting started **2**

2.1 CFD Pressure 3

2.2 Finite Element model 3

2.2.1 MSC.Marc 3

2.2.2 LS-Dyna 5

2.3 Using CFD Mapper 5

2.4 Export output file 7

3 Mapping algorithm **8**

3.1 Face centroid 8

3.2 Influence zone 8

3.3 Pressure smoothing 9

4 Code description and technical issues **10**

4.1 Structure 10

4.2 Output file 10

4.2.1 MSC.Marc 10

4.2.2 LS-Dyna 10

5 Examples **10**

6 Copyright and contribution **11**

1 Introduction

CFD Mapper is a stand-alone application developed to convert fluid pressure into a boundary condition of Finite Element model. It is an open-source software, supports 3D data visualization and parallel computing. CFD simulation is a commonly used tool to identify pressure distribution along the structure walls. Usually FE mesh does not match with CFD cell grid. In this case mapping procedure needs to be involved to transfer load across the domains.

First reason of creating this program was to couple independent CFD and FEA solvers. Advanced simulation environments like ANSYS offer similar tools with, frankly speaking, quite limited features. The problem starts if you want to mix different software. Second reason was a mapping outcome visualization. Obviously, everyone wants to check what is the actual distribution of the load applied to the structure, but not all solvers create this kind of output. Therefore, I put great effort into efficient graphics rendering using third party packages.

Main idea behind the code is distance based pressure mapping from CFD points to finite element faces with optional averaging called *pressure smoothing*. Detailed description of mapping algorythm is presented in Chapter 3.

Application is compiled to single file assembly (*.exe*) with all *.dll* files embedded - no additional packages or installation is needed to run it.

2 Getting started

The contents of the main window are (Figure 1): top toolbar, feature panel, viewport and message window. To create new project use *New* button in toolbar. View properties such as background and label color, mesh wireframe and transparency mode could be changed in *View* section. All controls related to mapping procedure and data visualization are located in *Feature panel*. Note that initially most of the contents are disabled. Therefore, input files needs to be loaded in the first step.

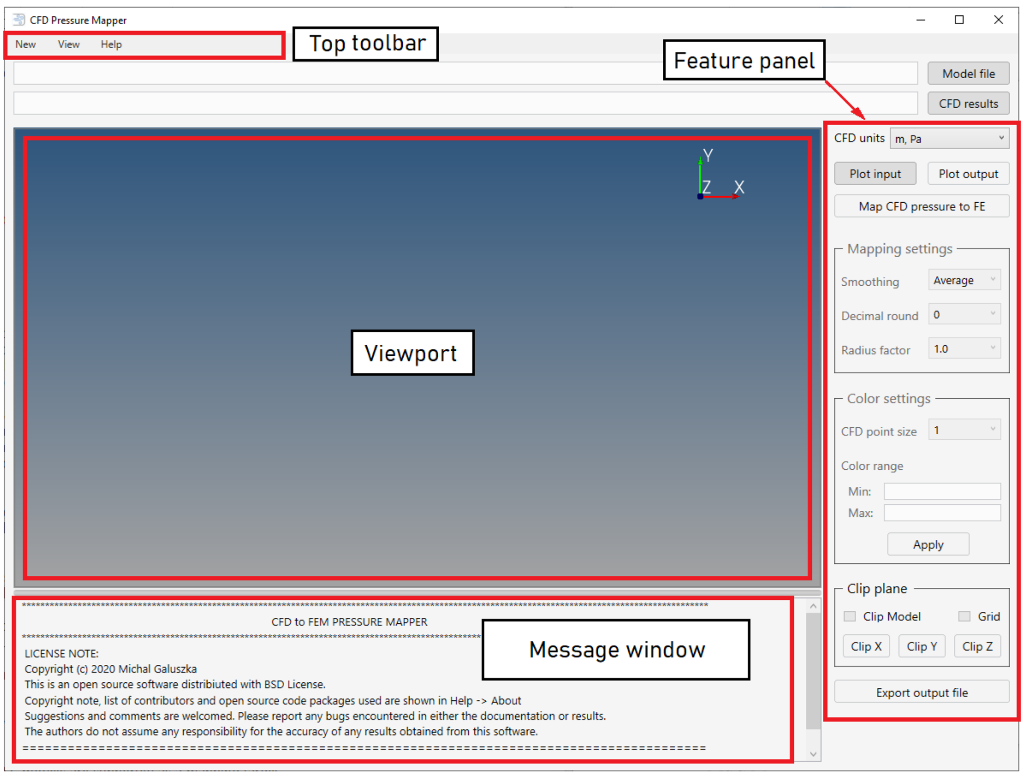


Figure 1: Main window

Two input files are required by mapping procedure (formatting described in section 2.1 and 2.2):

- *Model file* - Structural model input deck (solver dependent format),
- *CFD results* - Fluid pressure file with point coordinates.

2.1 CFD Pressure

Fluid flow analysis can be performed with any solver (Fluent, CFX, OpenFOAM, etc.). You can use any post-processor to export pressure data at the specified fluid domain boundary. The only requirement is to keep 4 column ASCII format:

- Column 1 - x coordinate,
- Column 2 - y coordinate,
- Column 3 - z coordinate,
- Column 4 - pressure value.

An example of CFD input format is presented in Figure 2. The following separators are supported: **comma, tab, white space**. Program reads each text line, splits it and tries to convert all 4 strings to double variables. If conversion fails, the line will be skipped and reported in *Message window*. For instance all comment and header lines are treated in such a way.

```
X [ m ]          Y [ m ]          Z [ m ]          Total Pressure [ Pa ]
3.57407220E-02, 1.02973819E-01, 4.19999994E-02, 1.81089188E+06
3.59905586E-02, 1.03221782E-01, 4.18881550E-02, 1.88604000E+06
3.58534865E-02, 1.03381477E-01, 4.18508537E-02, 1.92273563E+06
3.56597863E-02, 1.03336625E-01, 4.18881252E-02, 1.90487150E+06
3.54660861E-02, 1.03291780E-01, 4.19253968E-02, 1.92067700E+06
3.54095735E-02, 1.03087559E-01, 4.19999994E-02, 1.83991725E+06
3.70608494E-02, 1.02506094E-01, 4.19999994E-02, 1.69788113E+06
3.73098999E-02, 1.02751710E-01, 4.18883525E-02, 1.76713238E+06
3.71735692E-02, 1.02913447E-01, 4.18511182E-02, 1.82763400E+06
```

Figure 2: CFD input file format

2.2 Finite Element model

Currently only LS-Dyna and MSC.Marc FE models are supported as a mapping target. Input file format depends on solver, but in general all data necessary to create mapping target (loaded faces) are needed. FE model input should be built in the following way:

1. Create the mesh.
2. Define faces loaded by pressure - solver dependent procedure.
3. Export input deck.

Detailed description of input file reading in presented in Section 4.1.

2.2.1 MSC.Marc

Marc solver input is used by CFD Mapper to create FE mapping target. Only the following 3D element types are supported: 1st order TRIA, QUAD, TETRA, PENTA, HEXA and 2nd order TRIA, QUAD, TETRA. In MSC.Marc pressure is applied by *Face Load* boundary condition. Therefore, set of faces *"Pressure_faces"* **must** be included in input file. It can be easily created in Marc.Mentat as presented below.

1. Select faces:
   (a) *Select → Selection Control...*
   (b) Change *Method* to *Face Flood* and adjust *Limit angle*
   (c) Click *...Faces* and select element face on a wall
2. Create new structural boundary condition *Face Load* named *Pressure*.
3. Add *All Selected* faces to load.
4. Activate pressure load in *Loadcase*.

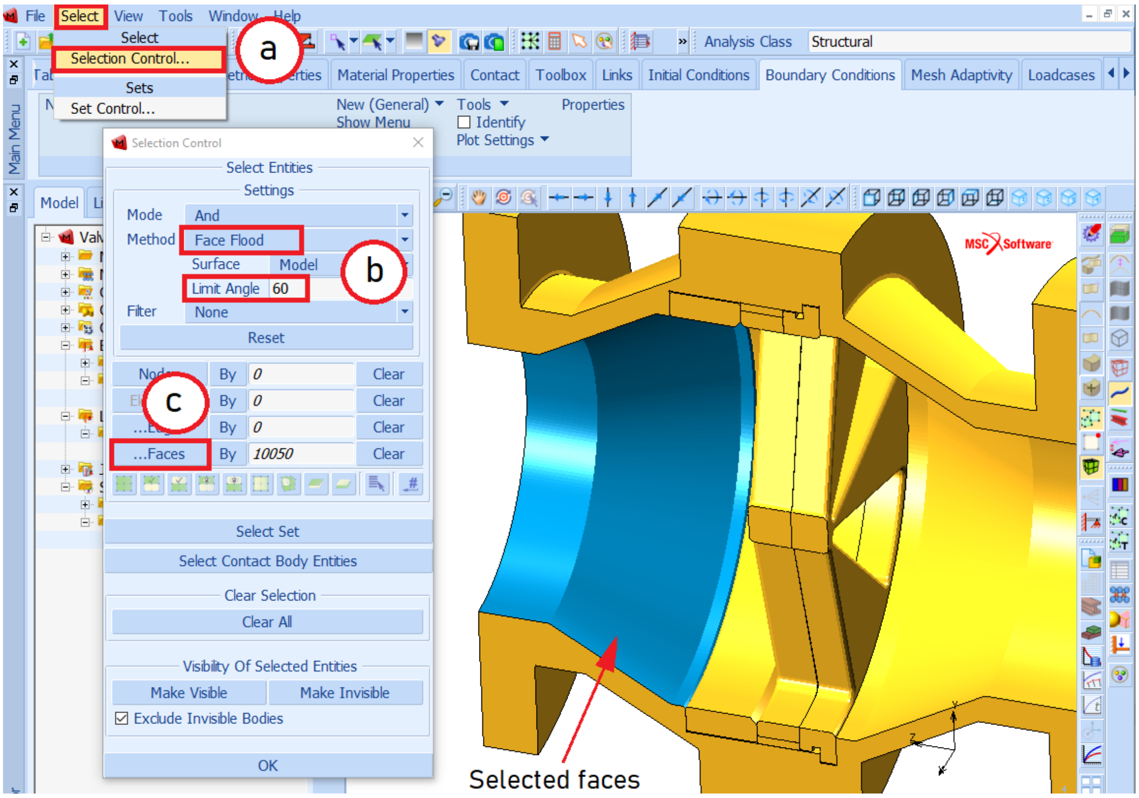


Figure 3: Face selection in Marc Mentat

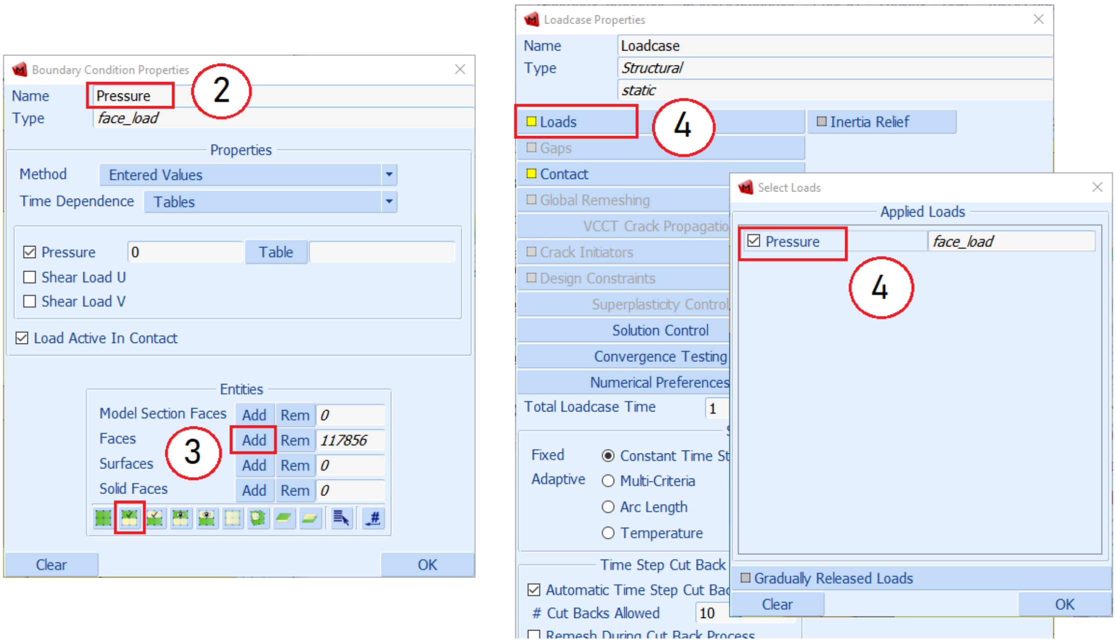


Figure 4: Create new *Face Load* in Marc Mentat

Finally export Marc Input (*.dat*). If *Face Load* was created properly, text line shown in Figure 5 should exist in input file. The following lines specify the element and its face number. It is mandatory to use **Pressure_faces** set name, otherwise CFD mapper will omit it.

```
define              facemt              set                 Pressure_faces
            9224:3              9225:3              9226:3              9227:3
            9232:3              9233:3              9234:3              9235:3
```

Figure 5: Pressure face set line in Marc input file

2.2.2 LS-Dyna

In LS-Dyna it is very convenient to apply pressure load using ***LOAD_SEGMENT** keyword. In this case ***SET_SEGMENT** is needed to define loaded zones. You can use any pre-processor to do it, for instance LS-PrePost. Simple example is presented in Figure 6.

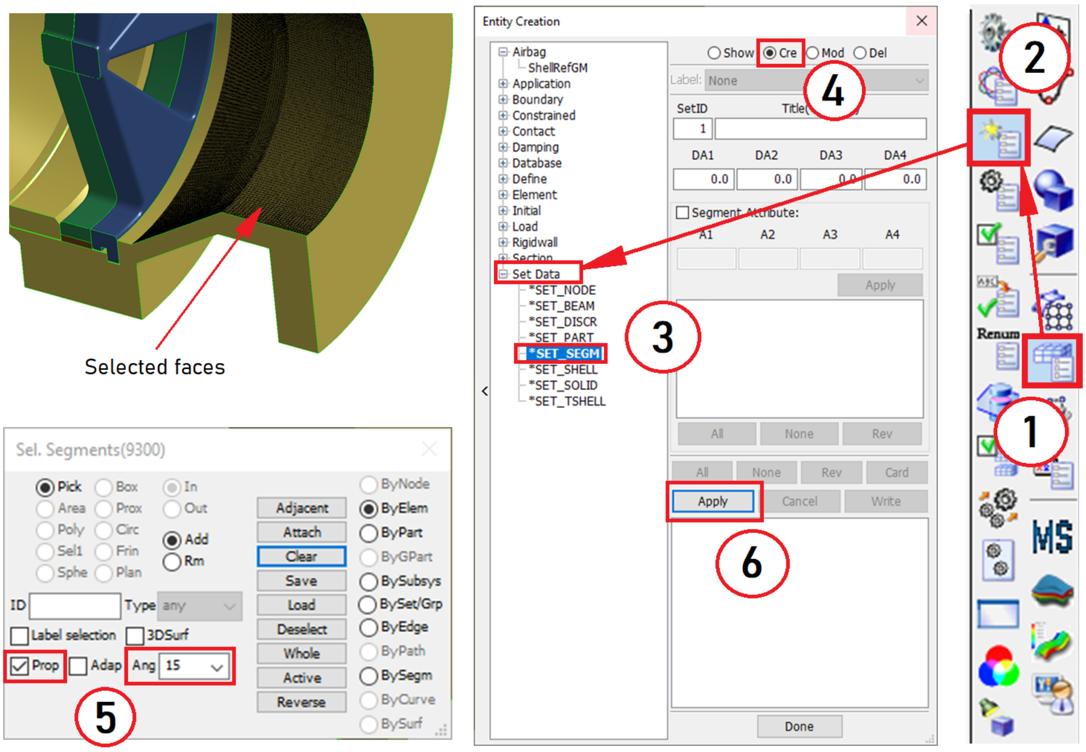


Figure 6: Create *SET_SEGMENT in LS PrePost

Structure of ***SET_SEGMENT** is shown in Figure 7. Note that only Node ID is used to define segment, so ***ELEMENT** keyword is not required in LS-Dyna input file. Therefore segment could be bound with both SOLID and SHELL elements.

```
*SET_SEGMENT_TITLE
Pressure faces
$#      sid       da1       da2       da3       da4    solver
          1       0.0       0.0       0.0       0.0MECH
$#       n1        n2        n3        n4        a1        a2        a3        a4
     530274    530892    530894    530276       0.0       0.0       0.0       0.0
     136550    136548    139448    139450       0.0       0.0       0.0       0.0
     692342    692340    692334    692334       0.0       0.0       0.0       0.0
```

Figure 7: *SET_SEGMENT keyword

2.3 Using CFD Mapper

To select input files use *Model file* and *CFD results* buttons at the top of the window. If selected files do not exist warning window will be shown. In case of different unit system used in fluid and structural analyses, choose the right *CFD units* to convert CFD input data. Afterwards, press *Plot input* button to build visualization. Input file reading should take less then a minute. The lines that could not be converted to numeric data are reported in *Message window*. An example of input data visualization is shown in Figure 9.

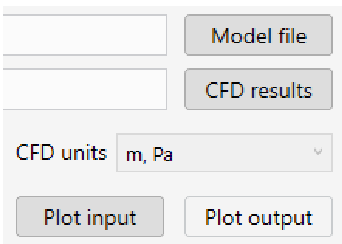


Figure 8: Input and Plot buttons

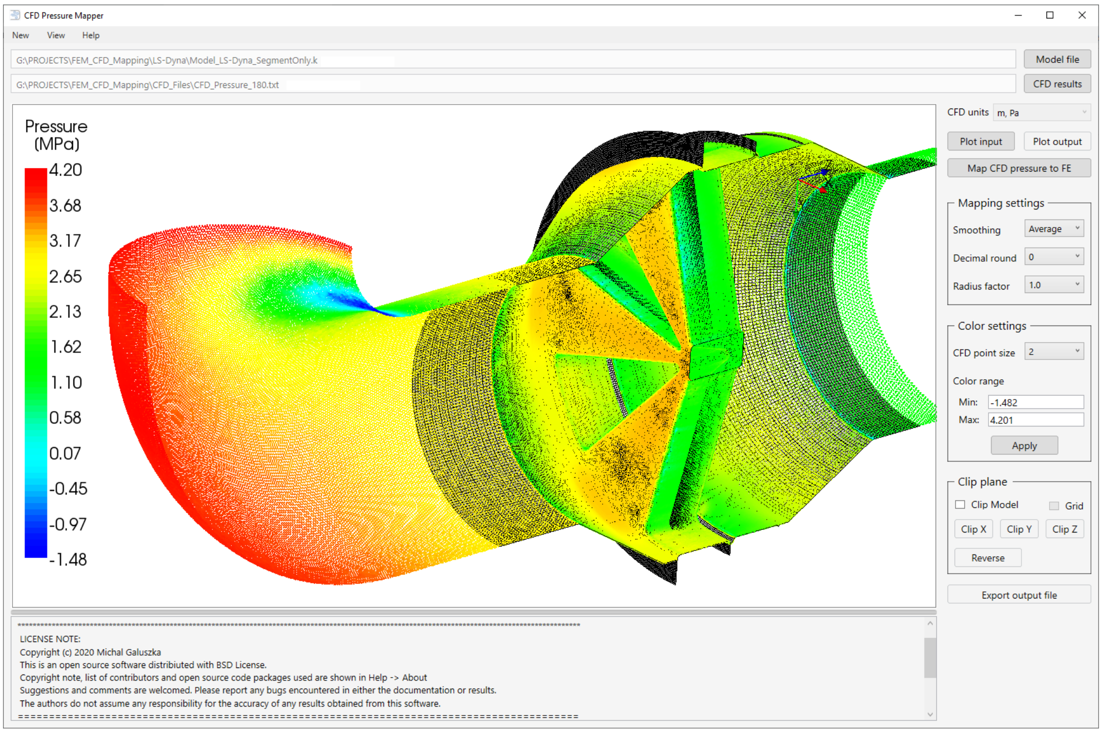


Figure 9: Input data visualization in CFD Mapper

CFD data are represented by points with color specified by default scale (min/max). CFD data appearance modification is provided by *Color settings* subpanel. Use *CFD point size* box to increase/decrease vertex size. Colorbar scale could be adjusted with *Color range* boxes.

FE faces have grey color in *Plot input* mode. Wireframe and transparency could be turn on/off in *View* section in Top toolbar. You can also create clip plane normal to X, Y or Z axis using *Clip plane* subpanel.

At this stage you may verify whether fluid domain boundaries fit to structural model. Note that small gap/overlap is allowed since mapping procedure uses so called *influence zone* surrounding FE face (see Chapter 3). Moreover, fluid domain may extend beyond structural model as shown in Figure 9. Nonetheless, data resolution is crucial. In general best mapping quality is provided when CFD grid resolution in lower then Finite Element size - multiple pressure points inside the FE face (Figure 10). It is not mandatory, but recommended.

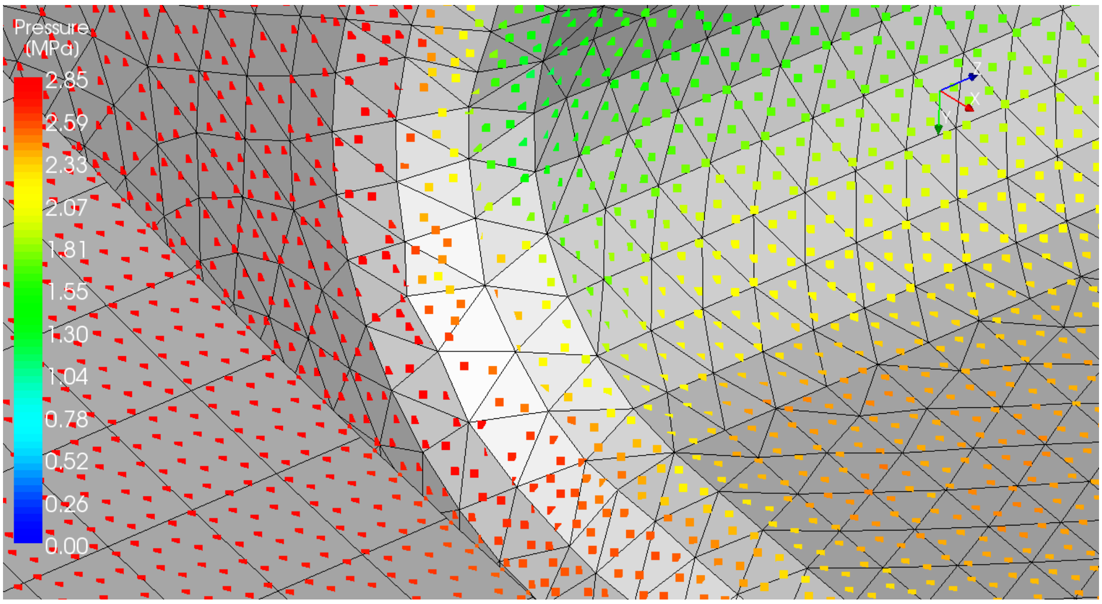


Figure 10: An example of recommended input data resolution

Last step before mapping is to check *Mapping settings*. *Smoothing* and *Radius factor* options are explained in Chapter 3. Briefly, if CFD grid resolution in greater then Finite Element size, *Min, Max* or *Average* smoothing is enough. Otherwise, when CFD grid is coarse, it highly recommended to use *Linear* or *Gaussian* smoothing. On the first try *Radius factor* 1.0 might be taken. Decimal round allows to round off mapped value and should be chosen depending on pressure range. It is mostly relevant for Marc inputs to gather large groups of faces with the same boundary condition (to speed up .proc reading by Mentat).

Afterwards, when all set, click *Map CFD pressure to FE* button. Running time is mostly dependent on number of faces, in fact number of pressure points has much smaller impact. For large scale models, calculation might take a while, but it should be no more than few minutes (around 3 min for benchmark with 1 million CFD points and 110 000 FE faces).

When computing is finished, message about wall time and mapping result will be shown (Figure 11). As you can see, pressure may not be assigned to all of the faces if there is no CFD point in their *influence zones*. This message is also printed in *Message window* at the bottom. Therefore mapping should be rerun with bigger value of *Radius factor*.

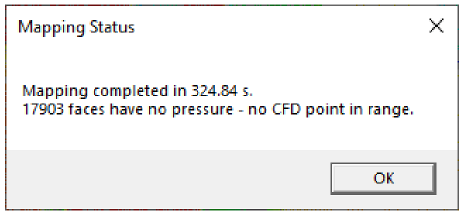


Figure 11: Mapping Status message

Plot output button is enabled now, click it to evaluate mapping outcome. You can quickly switch between input and output data to compare them, color scale remains unchanged. Example of mapping input/output is shown in Figure 12. In output mode CFD points are hidden and faces with no pressure assignment have gray color.

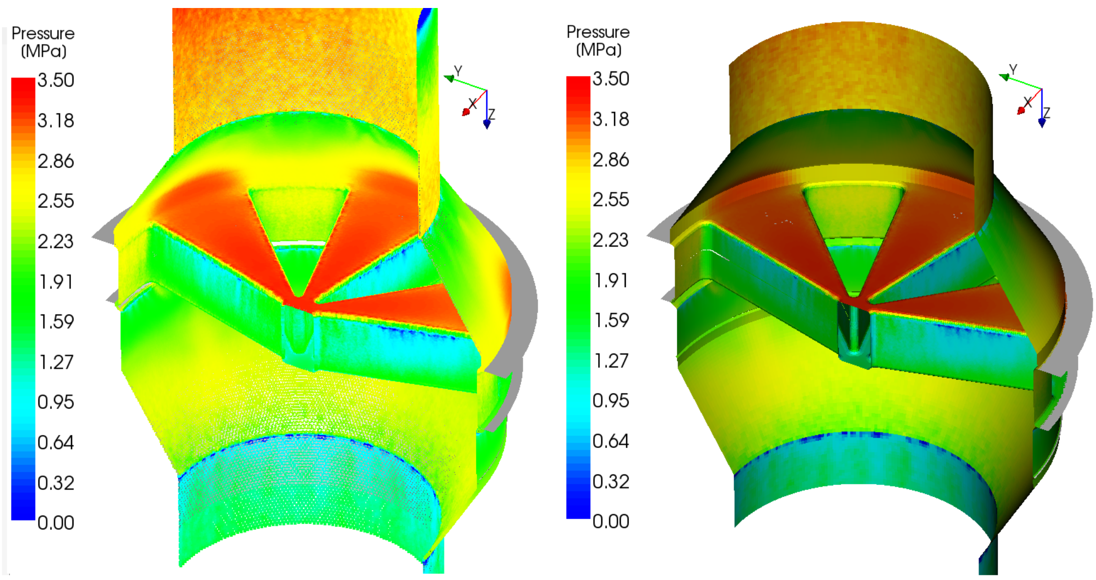


Figure 12: Mapping input (left) and output (right)

2.4 Export output file

When satisfying pressure mapping is achieved output file with structural load can be exported. Output file format is automatically matched based on FE input file. The file is always saved in FE input file directory with extension *"_PressureMapCFD"*.

For MSC.Marc the output is a *.proc* file which can be directly executed in Mentat. For LS-Dyna the output is *.k* file with keyword *LOAD_SEGMENT, which would be an include in your master input deck. In both cases pressure rise is defined by linear curve with abscissa in the range 0 to 1. Each face (or group of faces in Marc) has individual pressure scale factor assigned related to mapping result.

3 Mapping algorithm

CFD Mapper uses distance based pressure mapping from CFD points to finite element faces with optional averaging called *pressure smoothing*. Basic assumption is that pressure is uniform over the face. Therefore, the most representative pressure value is the one located in face center. The smaller distance between face centroid and CFD point, the better pressure representation. However CFD point may, obviously, occupied any place in space. To overcome this issue, *influence zone* is introduced.

3.1 Face centroid

There are to kind of face shape that may appear for supported FE type: triangular and quadrilateral. Triangle center is a simple arithmetic average of vertex coordinates:

$$x_O = \frac{x_A + x_C + x_C}{3}; \qquad y_O = \frac{y_A + y_B + y_C}{3}; \qquad z_O = \frac{z_A + z_B + z_C}{3}$$

In case of quadrangle, it is split into 2 triangles and centroid is calculated as a midpoint between their centers (Figure 13).

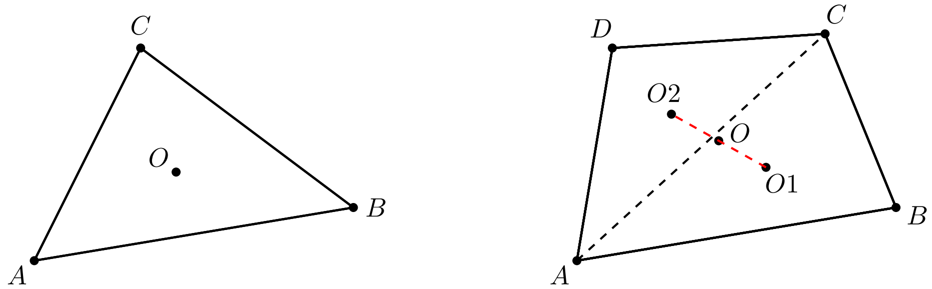


Figure 13: Centroid location for triangular (left) and quadrilateral (right) FE face

3.2 Influence zone

The purpose of influence zone is to limit number of CFD point associated with considered Finite Element face only to its closest neighbors. Since the distance to CFD point is measured from face center, the most reasonable shape of influence zone is obviously a sphere. The sphere should be big enough to cover whole face, therefore its radius is an average of distance between each vertex and centroid.

For quadrilateral face:

$$R = \frac{|OA| + |OB| + |OC| + |OD|}{4}$$

Radius R can be scaled by *Radius factor* variable to expand influence zone (usefull if CFD grid is coarse). To exclude CFD points that are far away from centroid, "box" is defined around the face. Box has an orgin located in face centroid and default size 2 times R. Note that distance calculation is much more expensive than boxing procedure (simple x, y, z comparing).

Finally, CFD points inside the box are checked whether they belong to influence zone. If so, they contribute the resultant pressure value in accordance with selected smoothing technique.

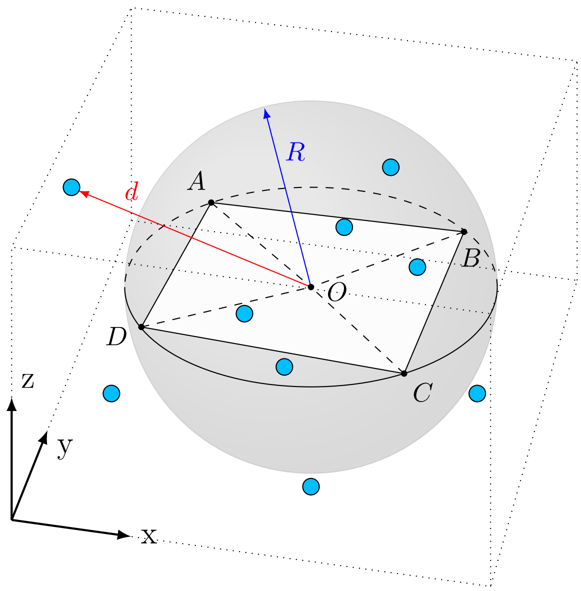


Figure 14: Influence zone surrounding FE face

3.3 Pressure smoothing

One can imagine that influence zone of any face might contain multiple CFD points. Thus, it needs to be handled somehow to assign one, most representative value of pressure to single FE face. Here comes an idea for so called *smoothing* procedure, which is some sort of approximation technique. Several smoothing types have been implemented in CFD Mapper:

- *Min* - minimal value of pressure is taken from set of CFD points inside influence domain:

$$P = min(P_i)$$

- *Max* - maximal value of pressure is taken from set of CFD points inside influence domain:

$$P = max(P_i)$$

- *Average* - arithmetic average value of pressure is taken from set of CFD points inside influence domain:

$$P = \frac{\sum\limits_{i=1}^{n} P_i}{n} \text{ ; where } n - \text{number of CFD points}$$

- *Linear* - weighted average value of pressure is taken from set of CFD points inside influence domain. Weight is a linear function of CFD point distance from face centroid:

$$P = \frac{\sum\limits_{i=1}^{n} P_i w_i}{\sum\limits_{i=1}^{n} w_i} \text{ ; where } w - \text{linear weight (Figure 15)}$$

- *Gaussian* - weighted average value of pressure is taken from set of CFD points inside influence domain. Weight is a nonlinear (Gaussian) function of CFD point distance from face centroid:

$$P = \frac{\sum\limits_{i=1}^{n} P_i w_i}{\sum\limits_{i=1}^{n} w_i} \text{ ; where } w - \text{nonlinear weight (Figure 15)}$$

Weight functions are shown in Figure 15. Note that in fact *Average* smoothing has constant weight equal to 1.

Linear weight is expressed by:

$$w(x) = 1 - \frac{x}{R}$$

Gaussian weight uses normalized Gaussian distribution with $\sigma = 0.20$:

$$w(x) = \exp\left(-\left(\frac{x}{R}\right)^2 / 2\sigma^2\right)$$

For influence zone radius factor less or equal to 1 *Average* smoothing is sufficient since we do not exceed face boundaries very much. However for factor 2 or higher it is desirable to emphasize CFD points located closer to centroid with some nonconstant weight. In case of very nonuniform mesh Gaussian weight might be more accurate then Linear, but note that it actually reduce the influence zone size by almost half.

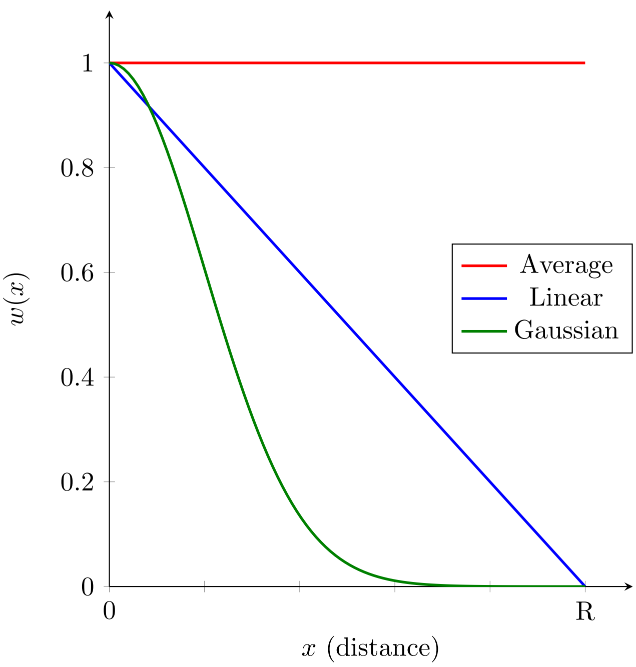


Figure 15: Weight functions for different smoothing type

4 Code description and technical issues

To do...

4.1 Structure

To do...

4.2 Output file

To do...

4.2.1 MSC.Marc

For Marc output is a *.proc* file which can be directly executed in Mentat. A piece of created procedure file is presented on right hand side. Firstly, Mentat window *Update* and *Undo memory* is turned off to speed-up execution. Secondly, the two point table [0,0 1,1] is created to define pressure vs. time dependency. Afterwords, a single boundary condition is created with name *"Pressure_"* plus loading value (e.g. *Pressure_307*). Table is assigned and *Load Active in Contact* disabled. Next, load is assigned to the list of faces. Finally, the boundary condition is activated in every *Loadcase* and job's *Initial Loads*. At the end of procedure, *Update* and *Undo* are switched on back again.

4.2.2 LS-Dyna

To do...

5 Examples

To do...

6 Copyright and contribution

Third party packages used:

- ActiViz .NET x64 (v5.8.0): https://kitware.eu/activiz
- Costura.Fody (v4.1.0): https://github.com/Fody/Costura

Download source code and binary: https://github.com/galuszkm/CFD_Mapper.git
Please report any bugs or comments at: michal.galuszka1@gmail.com