Re-meshing techniques on Kratos Multiphysics with the MMG library

Kratos Workshop 2019

Vicente Mataix Ferrándiz [1]   Riccardo Rossi [1]   Rubén Zorrilla Martínez [1]
Carlos Roig Pina[1]   Alejandro Cornejo Velázquez [1]   Marc Núñez Corbacho [1]
Eugenio Oñate Ibañez de Navarra [1]
vmataix@cimne.upc.edu

[1]CIMNE. International Center for Numerical Methods in Engineering, Technical University of Catalonia (UPC). Barcelona. Spain

March 26, 2019

Overview

Section 1

Introduction

MMG library

(a) **MMG** (b) QR code (I) (c) QR code (II)

MMG is an open source software for simplicial remeshing. You can download the code in GitHub.

Features

- It uses a **LGPL** license and it has been integrated in *Kratos* via the `mmg_process.h` in the `MeshingApplication`
- It is used like a process, using the `mmg_process.py` in the `MeshingApplication`
- Can be used to remesh in 3D/2D and 3D with surfaces (shells/membrames for example).

Data structures classes

`Model`

`Model` stores the whole model to be analyzed. All `Nodes`, `Properties`, `Elements`, `Conditions` and solution data

`ModelPart`

`ModelPart` holds all data related to an arbitrary part of model. It stores all existing components and data like `Nodes`, `Properties`, `Elements`, `Conditions` and solution data related to a part of model

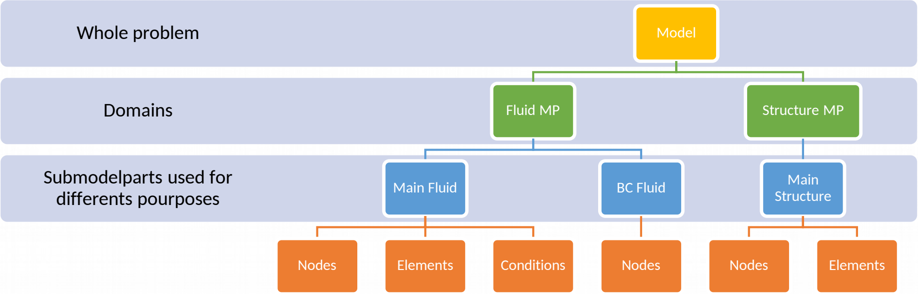
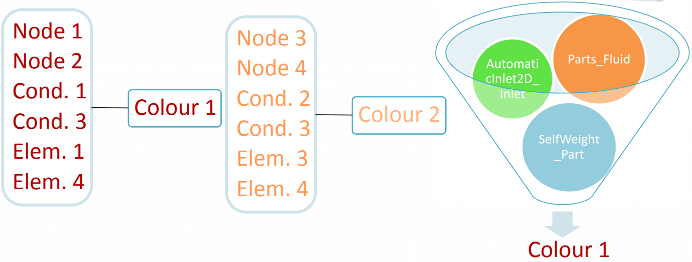


Figure 1: Example

Colours identification

In in our implementations we use a processes to set the **BC** (both *Neumann* or *Dirichlet*)



Processes and submodelparts relationship

```
"python_module" : "apply_inlet_process",
"kratos_module" : "KratosMultiphysics.FluidDynamicsApplication",
"help"          : [],
"process_name"  : "ApplyInletProcess",
"Parameters"    : {
    "model_part_name" : "AutomaticInlet2D_Inlet",
    "variable_name"   : "VELOCITY",
    "modulus"         : 1.0,
    "direction"       : "automatic_inwards_normal",
    "interval"        : [0,"End"]
}
```

Figure 2: Example of BC in *json* format

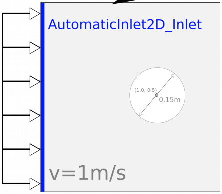


Figure 3: Subm. BC

Section 2

Theory

Level set remeshing

We compute the gradient (1) of a scalar variable f in order to compute an anisotropic metric to remesh, using the procedure from (2)

$$\nabla f = \frac{\partial f}{\partial x}\mathrm{i} + \frac{\partial f}{\partial y}\mathrm{j} + \frac{\partial f}{\partial z}\mathrm{k} \tag{1}$$

Level set metric computation

Calling h the element size and ρ the anisotropic ratio
The scalar value f and ∇f the gradient from that scalar. $\mathcal{M}$ is the metric
We compute the following auxiliar coefficients:

$$\begin{cases} c_0 = \frac{1.0}{h^2} & \text{Isotropic metric} \\ c_1 = \frac{c_0}{\rho^2} & \text{Applying anisotropic ratio} \end{cases} \tag{2a}$$

For **2D**:

$$\mathcal{M} = \begin{pmatrix} c_0(1-\nabla f_x^2) + c_1 \nabla f_x^2 & (c_1 - c_0)\nabla f_x \nabla f_y \\ (c_1 - c_0)\nabla f_x \nabla f_y & c_0(1-\nabla f_y^2) + c_1 \nabla f_y^2 \end{pmatrix} \tag{2b}$$

For **3D**:

$$\mathcal{M} = \begin{pmatrix} c_0(1-\nabla f_x^2) + c_1 \nabla f_x^2 & (c_1 - c_0)\nabla f_x \nabla f_y & (c_1 - c_0)\nabla f_x \nabla f_z \\ (c_1 - c_0)\nabla f_x \nabla f_y & c_0(1-\nabla f_y^2) + c_1 \nabla f_y^2 & (c_1 - c_0)\nabla f_y \nabla f_z \\ (c_1 - c_0)\nabla f_x \nabla f_z & (c_1 - c_0)\nabla f_y \nabla f_z & c_0(1-\nabla f_z^2) + c_1 \nabla f_z^2 \end{pmatrix} \tag{2c}$$

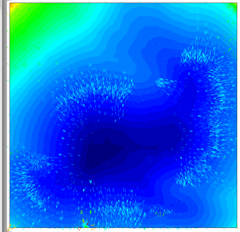


Figure 4: Scalar and its gradient

Hessian remeshing

Following a similar procedure like in the case of the level set, we can compute the hessian matrix (3) of a scalar variable f

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \, \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \, \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \text{ or, just: } H_{i,j} = \frac{\partial^2 f}{\partial x_i \partial x_j} \quad (3)$$

Hessian metric computation

Once the *Hessian* matrix has been computed we can compute the corresponding anisotropic metric by the following

$$\mathcal{M} = \mathcal{R}^t \tilde{\Lambda}^t \mathcal{R} \text{ where } \tilde{\Lambda} = (\tilde{\lambda}_i) \text{ being } \tilde{\lambda}_i = \min\left(\max\left(\frac{c_d |\lambda_i|}{\epsilon}, \frac{1}{h_{max}^2} \right), \frac{1}{h_{min}^2} \right) \quad (4a)$$

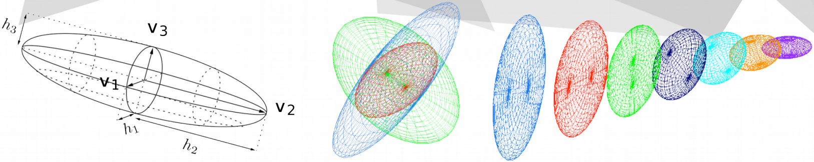
Being ϵ the error threshold and c_d a constant ratio of a mesh constant and the interpolation ratio
For an isotropic mesh the metric will be:

$$\mathcal{M}_{iso} = diag(\max(\tilde{\lambda}_i)) = \begin{pmatrix} \max(\tilde{\lambda}_i) & 0 & 0 \\ 0 & \max(\tilde{\lambda}_i) & 0 \\ 0 & 0 & \max(\tilde{\lambda}_i) \end{pmatrix} \quad (4b)$$

For anisotropic mesh will be (Being $R_{\lambda rel} = |\tilde{\lambda}_{max} - R_\lambda|$ where $R_\lambda = (1-\rho)|\tilde{\lambda}_{max} - \tilde{\lambda}_{min}|$):

$$\mathcal{M}_{aniso} = \mathcal{R}^t \begin{pmatrix} \max(\min(\tilde{\lambda}_1, \tilde{\lambda}_{max}), R_{\lambda rel}) & 0 & 0 \\ 0 & \max(\min(\tilde{\lambda}_2, \tilde{\lambda}_{max}), R_{\lambda rel}) & 0 \\ 0 & 0 & \max(\min(\tilde{\lambda}_3, \tilde{\lambda}_{max}), R_{\lambda rel}) \end{pmatrix} \mathcal{R} \quad (4c)$$

Metric intersection



(a) Metric analogy (b) Interpolation

The metric intersection consists in keeping the most restrictive size constraint in all directions imposed by this set of metrics[2]

Procedure

The simultaneous reduction enables to find a common basis such that $\mathcal{M}_1$ and $\mathcal{M}_2$ are congruent to a diagonal matrix, in this basis then $\mathcal{N}$ is introduced

$$\mathcal{N} = \mathcal{M}_1^{-1}\mathcal{M}_2 \text{ considering that can be decomposed in } \lambda_i = \mathrm{e}_i^t \mathcal{M}_1 \mathrm{e}_i \text{ and } \mu_i = \mathrm{e}_i^t \mathcal{M}_2 \mathrm{e}_i \tag{5a}$$

Considering $\mathcal{P} = (\mathrm{e}_1 \mathrm{e}_2 \mathrm{e}_3)$ be the matrix the columns of which are the eigenvectors of $\mathcal{N}$ (common basis)

$$\mathcal{M}_1 = \mathcal{P}^{-t} \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} \mathcal{P}^{-1} \text{ and } \mathcal{M}_2 = \mathcal{P}^{-t} \begin{pmatrix} \mu_1 & 0 & 0 \\ 0 & \mu_2 & 0 \\ 0 & 0 & \mu_3 \end{pmatrix} \mathcal{P}^{-1} \tag{5b}$$

Computing the metric intersection as:

$$\mathcal{M}_{1\cap 2} = \mathcal{M}_1 \cap \mathcal{M}_2 = \mathcal{P}^{-t} \begin{pmatrix} \max(\lambda_1, \mu_1) & 0 & 0 \\ 0 & \max(\lambda_2, \mu_2) & 0 \\ 0 & 0 & \max(\lambda_3, \mu_3) \end{pmatrix} \mathcal{P}^{-1} \tag{5c}$$

Internal variables transfer

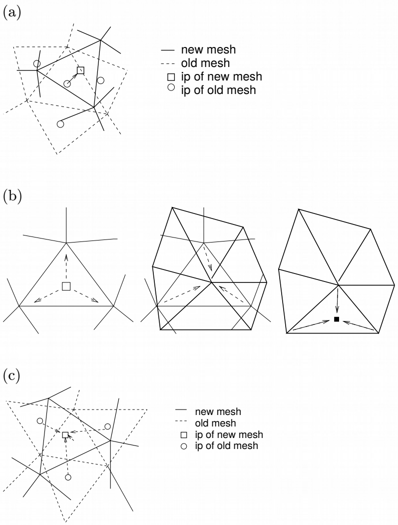


Figure 5: Transfer operators

The Figure 5 shows graphically how each one of the transfer methods work

Techniques

- *CPT*: Closest Point Transfer. **(a)**

  It just takes the value from the closest point

  It provides acceptable results at low cost

- *SFT*: Shape Function Projection transfer. **(b)**

  It interpolates the values using the standard **FEM** shape functions

  Leads to an artificial damage diffusion, but preserves the original shape of the damage profile

- *LST*: Least-Square Projection transfer. **(c)**

  It uses an *least-square* transfer across the closest points

  It is probably the most accurate technique but computationally more expensive

Some example will be shown following

Numerical contact remeshing (I)

NOTE: Work of *Anna Rehr* from **TUM**

Error estimation[8]

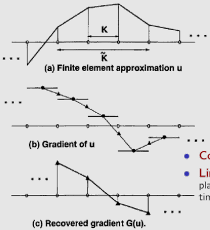
Residual based methods

- **Internal residual** (r): Error in the differential equation
- **Boundary error** (R):
  - *Traction boundaries*: Difference stress and traction
  - *Interelement boundaries*: Stress jumps
  - *Contact boundary*: Difference stress and contact pressure

$$\|e\|_{E,K} \approx \|\hat{e}\|_{E,K} = C[h_k^2\|r\|_{L_2(K)} + h_K^2\|R\|_{L_2(\partial K)}]$$

- Sound mathematical error bounds
- Determination of the constant C not trivial
- Quality depends heavily on the chosen constant

Recovery based methods



- Constant stress FE solution
- Linear recovered solution: Replaces the exact solution in the error estimation

$$\|e\|_{E,K} \approx \|\hat{e}\|_{E,K} = \left[\int_{\Omega_K} \left(\boldsymbol{\sigma}^* - \boldsymbol{\sigma}_h\right)^T D^{-1} \left(\boldsymbol{\sigma}^* - \boldsymbol{\sigma}_h\right)\right]^{\frac{1}{2}} d\Omega_K$$

- Robust
- Easy implementation, no unknown constant, easy extensibility
- No sound mathematical error bounds

Numerical contact remeshing (II)

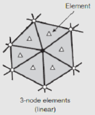
NOTE: Work of *Anna Rehr* from **TUM**

The present work introduces a modified version of the **SPR**[8] method

Modified SPR

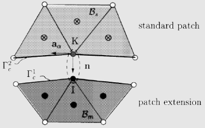
Element patch:

All elements that are neighboring one node



- **Concept**:
  - Stresses at the integration points show superior convergence behavior
  - Use these points to compute the superior stress field
- **Procedure**:
  - Execute a polynomial least square fit with the integration points
  - Compute with this polynomial recovered stress at the center node
  - Compute the stress field by interpolation with the shape functions

Extension for contact mechanics



- **Existing approach (penalty formulation)**:
  - Couple patches at the contact boundary
  - Enforce stress continuity in the recovery procedure by a penalty formulation
- **Anna Rehr's work**:
  - Patch coupling not necessary: contact pressure is known (Lagrange Multiplier)
  - Contact BC are regarded in the recovered stress calculation by a penalty formulation which forces the stresses to coincide with the contact pressure

Section 3

Cases

Coarse sphere

In this problem we remesh using the gradient of the distance function, which is the distance to the plane contained in the sphere center.

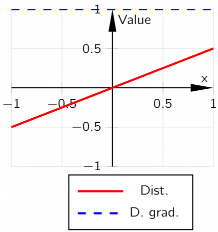


Figure 6: Distance function

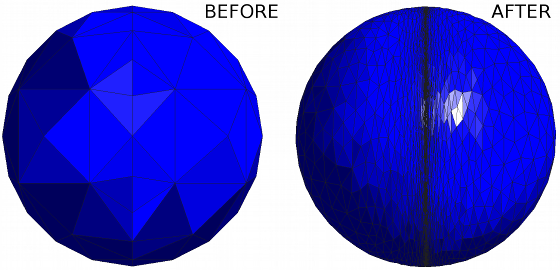


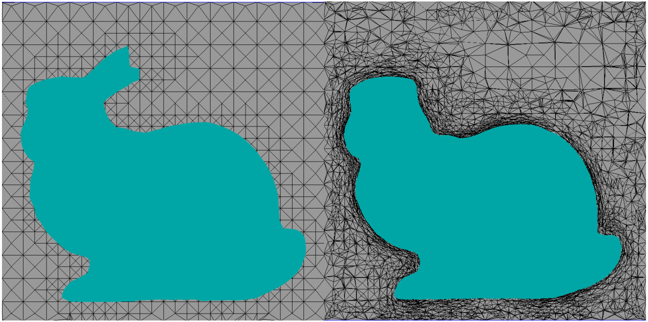
Figure 7: Mesh before and after remeshing

The function can be seen in the Figure6

Stanford's bunny

Geometry of the *Standford*'s bunny[4].

Figure 8: Standford's bunny



(a) Octree mesh (b) Anisotropic mesh

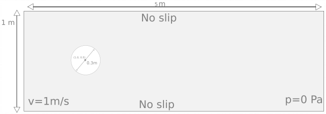
Figure 9: Mesh before and after remeshing

Anisotropically remesh the geometry using the distance gradient as error measure . Previously meshed with an embedded octree mesher (*GiD*).

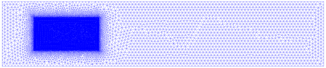
Embedded fluid channel 2D

Adaptative anisotropic remeshing of 2D fluid channel with sphere using as level set the distance function. The problem is solved using an embedded formulation

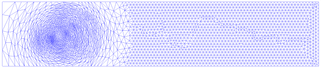
It consists in a channel 5x1, a sphere of 0.3 diameter and with a velocity of 1 m/s in the inlet an zero pressure in the outlet. The resulting flow has *Reynolds* number of **100**.



(a) Problem

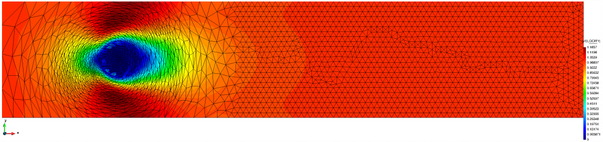


(b) Initial mesh

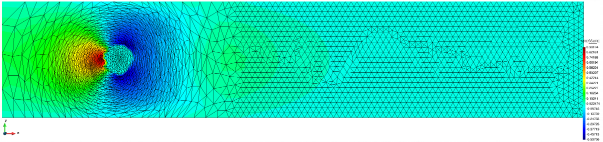


(c) Remeshed

Figure 10: Setup



(a) Velocity

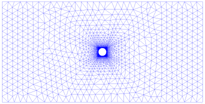


(b) Pressure

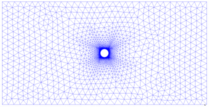
Potential fluid simulation 2D. Sphere

NOTE: Work of *Marc Núñez Corbacho* from **CIMNE**

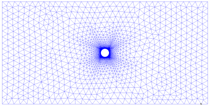
Adaptative anisotropic remeshing of 2D fluid channel with sphere using as level set the distance function. The problem is solved using a potential fluid formulation



(c) No remesh

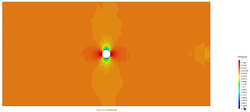


(d) Remesh 0.01

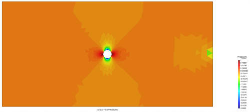


(e) Remesh 0.005

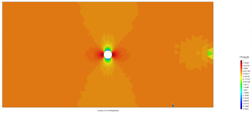
Figure 11: Meshes



(a) No remesh

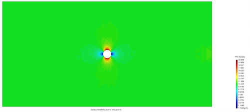


(b) Remesh 0.01

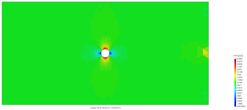


(c) Remesh 0.005

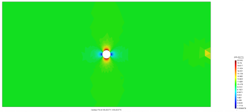
Figure 12: Pressure



(a) No remesh



(b) Remesh 0.01



(c) Remesh 0.005

Figure 13: Velocity

Potential fluid simulation 2D. NACA 12 (I)

NOTE: Work of *Marc Núñez Corbacho* from **CIMNE**

Adaptative anisotropic remeshing of 2D fluid channel with a NACA 12 aerofoil using as level set the distance function. The problem is solved using a potential fluid formulation

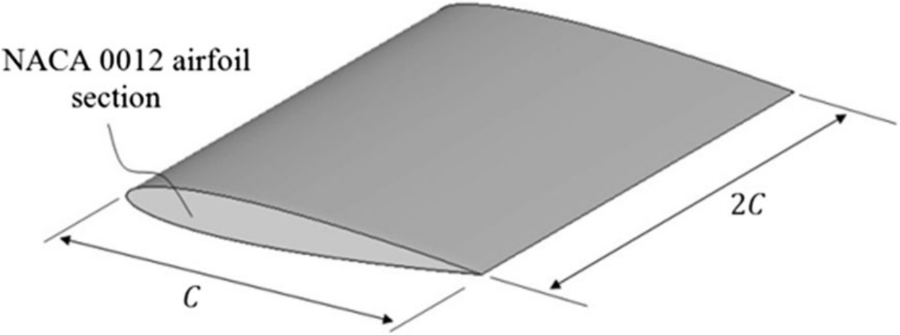
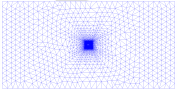
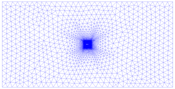
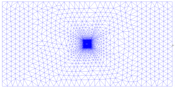


Figure 14: NACA 12 aerofoil

Potential fluid simulation 2D. NACA 12 (II)


(a) No remesh 0°


(b) Remesh 0°


(c) No remesh 5°

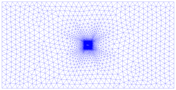
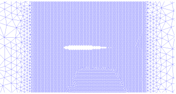
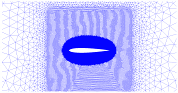
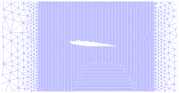

(d) Remesh 5°

Figure 15: Meshes


(a) No remesh 0°


(b) Remesh 0°


(c) No remesh 5°

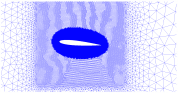
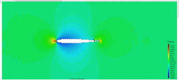
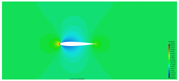
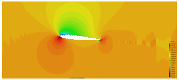

(d) Remesh 5°

Figure 16: Detail


(a) No remesh 0°


(b) Remesh 0°


(c) No remesh 5°

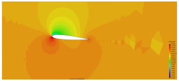
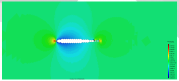
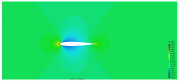
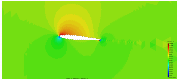

(d) Remesh 5°

Figure 17: Pressure


(a) No remesh 0°


(b) Remesh 0°


(c) No remesh 5°

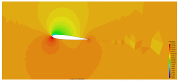
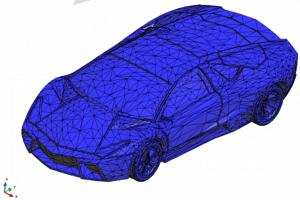
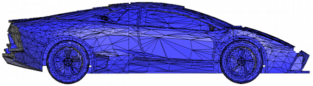

(d) Remesh 5°

Figure 18: Velocity

Lamborghini



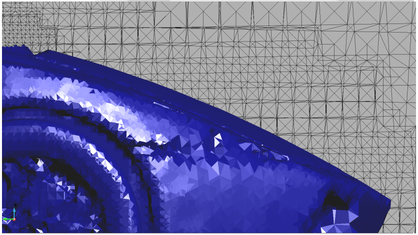
(a) View 1



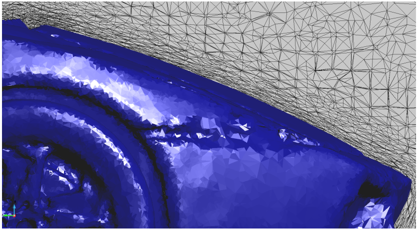
(b) View 2

Figure 19: Lamborghini

In this test case we want to remesh anisotropically the geometry of *Lamborghini*, more complex that the previous bunny.
Anisotropically remesh the geometry using the distance gradient as error measure.
Previously meshed with an embedded octree mesher (*GiD*).



(a) Octree mesh



(b) Anisotropic mesh

Figure 20: Mesh before and after remeshing

Hessian 2D

The problem corresponds with the example proposed in reference[2]

The objective is to remesh the structured 1x1 mesh with the error function from Figure 21 and equation (6)

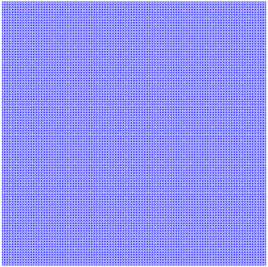
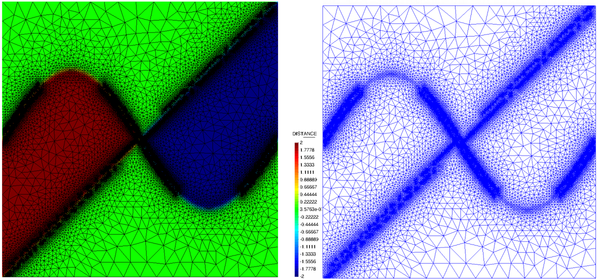


Figure 21: Initial mesh



(a) Error estimation

(b) New mesh

Figure 22: Solution

The χ shaped function:

$$\begin{aligned} f(x,y) = &\tanh(-100(y - 0.5 - 0.25\sin(2\pi x))) \\ &+ \tanh(100(y - x)) \end{aligned} \tag{6}$$

KRATOS MULTI-PHYSICS

Hessian 3D

An extension of the previous problem to 3D in several remeshing iterations

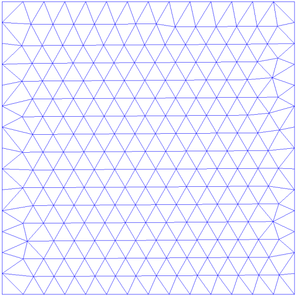
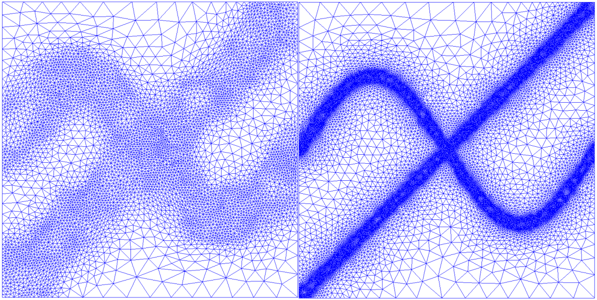
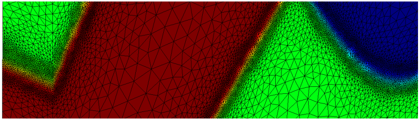


Figure 23: Initial mesh



(a) Iteration 1

(b) Iteration 2

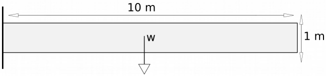


(c) Error estimation

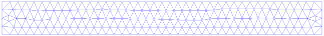
Figure 24: Solution

Beam 2D

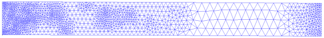
The simulation considers 100 time steps of 0.01s. The problem will be remeshed each ten steps considering the Hessian of the displacement



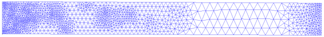
(a) Problem



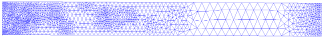
(b) Initial mesh



(c) Mesh 1

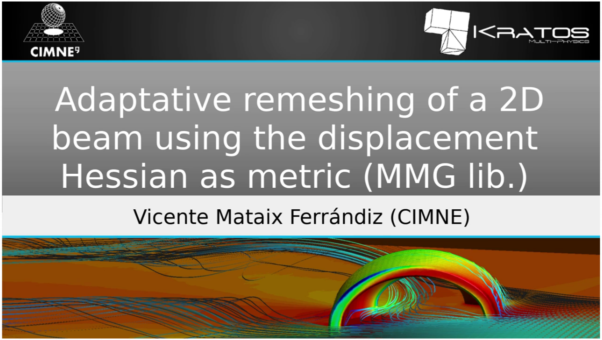


(d) Mesh 2



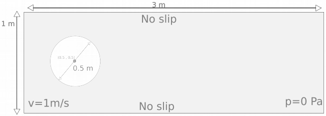
(e) Mesh 3

Figure 25: Setup

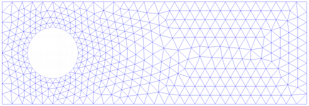


Fluid channel 2D

Adaptative remeshing of 2D fluid channel with sphere using Hessian of velocity as metric measure. It consists in a channel 3x1, a sphere of 0.5 diameter and with a velocity of 1 m/s in the inlet an zero pressure in the outlet.

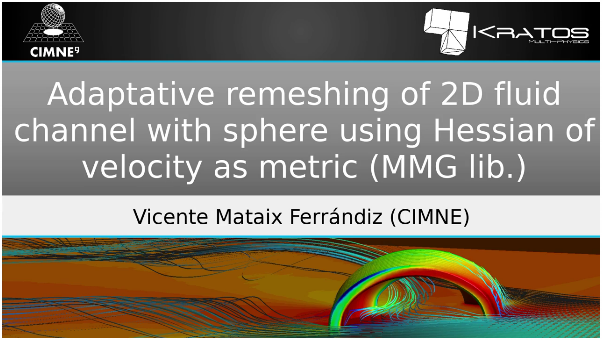


(a) Problem



(b) Initial mesh

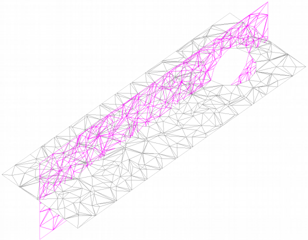
Figure 26: Setup



Fluid channel 3D

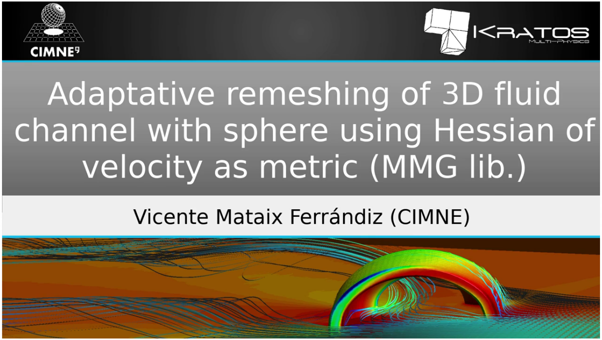
Adaptative remeshing of 3D fluid channel with sphere using Hessian of velocity as metric measure. It consists in a channel 3x1x1, a sphere of 0.5 diameter and with a velocity of 1 m/s in the inlet an zero pressure in the outlet.

(a) Problem



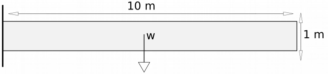
(b) Initial mesh

Figure 27: Setup

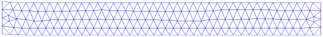


Beam 2D

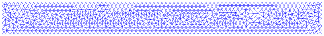
The simulation considers 10 time steps of 0.01s
The problem will be remeshed each ten steps considering the Hessian of the displacement
A J2-plasticity law has been considered



(a) Problem

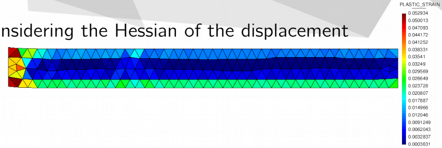


(b) Initial mesh

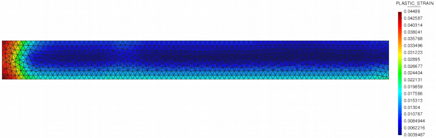


(c) After remeshing

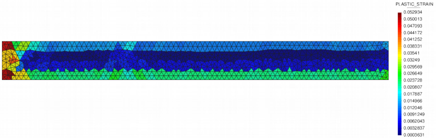
Figure 28: Setup



(a) Initial mesh



(b) LST



(c) CPT

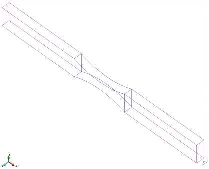
Figure 29: Plastic strain

Tensile test 3D

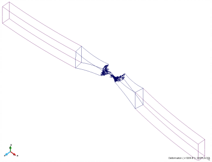
NOTE: Work of *Alejandro Cornejo Velázquez* from **CIMNE**

The problem will be remeshed each ten steps considering the Hessian of the equivalent stress at nodes. A damage law has been considered

(a) Problem



(b) Initial conf.



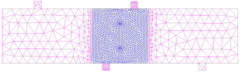
(c) Final conf.

Figure 30: Tensile test

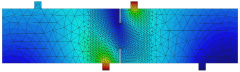
Concrete 4 point bending test

NOTE: Work of *Alejandro Cornejo Velázquez* from **CIMNE**

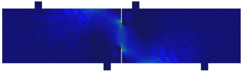
The problem will be remeshed each ten steps considering the Hessian of the equivalent stress at nodes. A damage law has been considered



(a) Problem

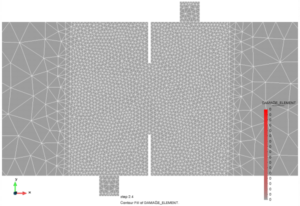


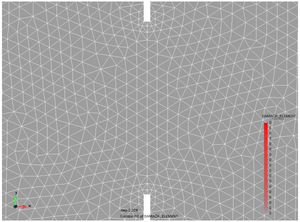
(b) Displacements



(c) Final conf.

Figure 31: Concrete 4 point bending test

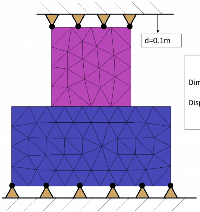




Patch test

NOTE: Work of *Anna Rehr* from **TUM**

The patch test is the most absic test to pass to verify a contact formulation
It has been solved in 2D using the modification of the **SPR**



$E = 2.1 \cdot 10^5\ N/m^2$
$\nu = 0.29$
Dimensions lower block: $2 \times 1m$
upper block: $1 \times 1m$
Displacement upper edge: $0.1m$

Figure 32: Setup

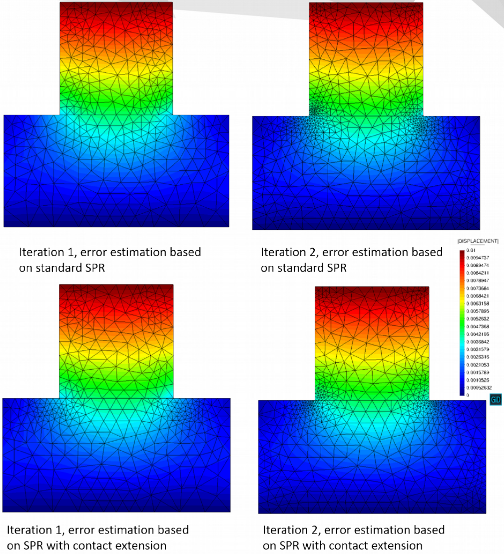


Figure 33: Solution

Hertz problem

NOTE: Work of *Anna Rehr* from **TUM**

The *Hertz* test is a very used benchmark for contact mechanics

It has been solved both in 2D and 3D using the modification of the **SPR**

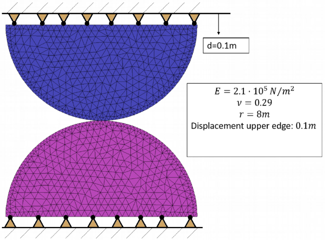
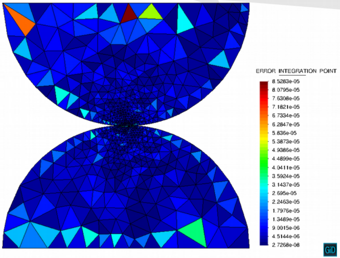
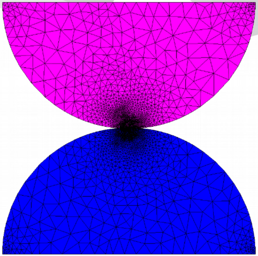


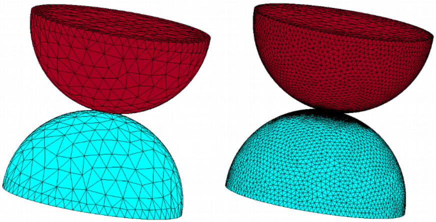
Figure 34: Setup



(a) Error



(b) Final mesh 2D



(c) Final mesh 3D

Now with shell remeshing!

Recently was added the compatibility with the sublibrary *MMGS*, for surface meshes.

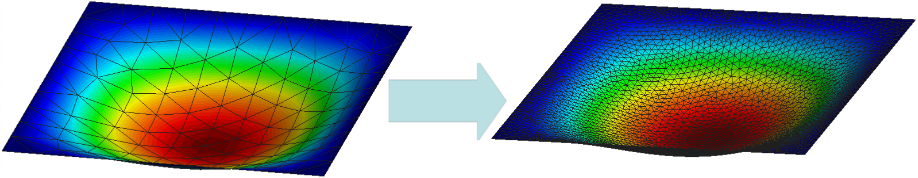


Figure 35: Remeshing of shells

Section 4

Conclusions

Conclusions and future work

Conclusions

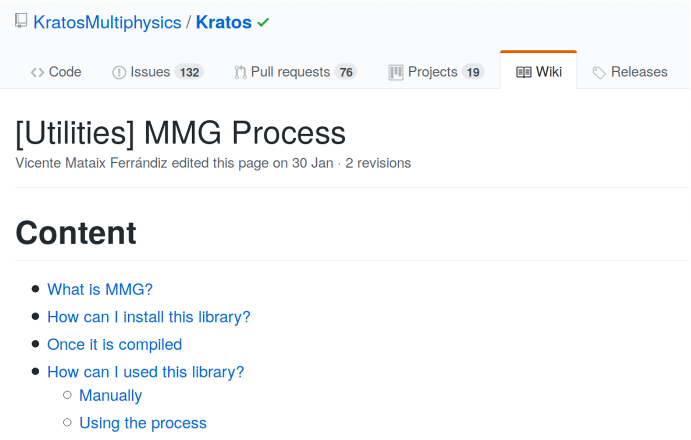
- *MMG:* We have implemented into *Kratos* the *MMG* **API**
- *Metrics*: We have implemented several metric measures and utilities
- *Internal values*: We have implemented several utilities in order to interpolate internal values
- *Contact*: We have implemented an extended version of **SPR** in order to be able to compute contact remeshing
- *Problems*: We have used the library in a set of different problems (fluid, structural analysis, potential flow)

Future works

- *New problems*: Use the process for new problems not computed yet (for example *FSI*)
- *Extend*: Extend the *Kratos*/*MMG* integration
- *Paralellization*: Using *ParMMG*. Part of the *ExaQUte* project

How to use it?

In order to access to the tutorial you can follow the following link:



(a) https://goo.gl/urzHRz

(b) QR code

References

P. Dadvand, R. Rossi, E. Oñate: *An Object-oriented Environment for Developing Finite Element Codes for Multi-disciplinary Applications.* Computational Methods in Engineering. 2010

F. Alauzet: *Metric-Based Anisotropic Mesh Adaptation.* Course material, CEA-EDF-INRIA Schools. Numerical Analysis Summer School. 2007

P. Tremblay: *2-D, 3-D and 4-D Anisotropic Mesh Adaptation for the Time-Continuous Space-Time Finite Element Method with Applications to the Incompressible Navier-Stokes Equations.* PhD thesis Ottawa-Carleton Institute for Mechanical and Aerospace Engineering, Department of Mechanical Engineering, University of Ottawa. 2007

G.Turk, M. Levoy: *The Stanford 3D Scanning Repository.*

P.J. Frey, F. Alauzet: *Anisotropic mesh adaptation for CFD computations.* Comput. Methods Appl. Mech. 2004

P.J. Frey, F. Alauzet: *Anisotropic mesh adaptation for transient flows simulations*

M. Bellet: *Adaptive mesh technique for thermal–metallurgical numerical simulation of arc welding processes*, International Journal for Numerical Methods in Engineering, 2008.

Zienkiewicz, O. C. and Zhu, J.Z. and Taylor, Robert L.: *The Finite Element Method: its Basis and Fundamentals*, Butterworth-Heinemann, 2013.

P.J. Frey, F. Alauzet: *Estimateur d'erreur ge´ome´trique et me´triques anisotropes pour l'adaptation de maillage. Partie I : aspects the´oriques*, HAL Id: inria-00071827, 2006

Thank you very much for your attention