TP2: Mesh adaptation with the Mmg platform

Algiane Froehly

12 décembre 2017

1   Goals :

- Adapt a mesh to a given size map ;
- Compute an isotropic/anisotropic size map based on the interpolation error.

2   Mmg platform in short

The Mmg platform gathers software dedicated to simplicial mesh modifications :
- mmg2d : for 2D meshes ;
- mmgs : for 3D surface meshes ;
- mmg3d : for 3D volume meshes.

All this software allow quality improvement, mesh adaptation to a size map (isotropic/anisotropic) and level-set discretization.

Additional documentation can be founded here :
http://www.mmgtools.org/mmg-remesher-try-mmg/mmg-remesher-tutorials
and here :
http://www.mmgtools.org/mmg-remesher-try-mmg/mmg-remesher-options

2.1   Installation

1. Download and untar the last stable binaries :
   - For Linux users :
     ```
     $ wget https://www.mmgtools.org/files/2017/11/mmg-5.3.6-Linux-4.9.0-2-
     amd64-appli.tar.gz
     $ tar -xvzf mmg-5.3.6-Linux-4.9.0-2-amd64-appli.tar.gz
     ```

   - for Mac users :
     ```
     $ wget https://www.mmgtools.org/files/2017/11/mmg-5.3.6-Darwin-16.7.0-
     appli.tar.gz
     $ tar -xvzf mmg-5.3.6-Darwin-16.7.0-appli.tar.gz
     ```

2. Add the path of the `bin` folder to you `PATH` variable to be able to run the applications from your terminal without adding the full binary path :
   ```
   $ echo "PATH=$PATH_TO_BIN:$PATH" >> ~/.bashrc
   $ source ~/.bashrc
   ```

3   A first run of the remesher

To run the remesher, you must give the application name followed by the path and mesh name (the naca mesh of the previous TP is provided in the `TP/Data` directory of this repository) :
```
$ cd TP
$ mmg2d_O3 Data/naca_embedded.mesh
```

FIGURE 1 – Default Mmg output.

FIGURE 2 – Detailed quality histogram.

By default, Mmg creates a mesh in the same path and of the same extension than the input mesh (.mesh here) with the .o prefix to the extension so here : `Data/naca_embedded.o.mesh`.

3.1 The Mmg output

You can see the Mmg output in the konsole. By default, Mmg prints (see figure 1) :

- The different phases of the algorithm (analysis step, remeshing step...) and the time spent in each of this step ;
- some info about the input/output qualities histogram ;
- the final mesh statistics (number of nodes, of elements and of edges).

You can change the default verbosity of Mmg with the -v option. By default, the verbosity value is setted to 1. If you set the verbosity to 5, `mmg2d_O3 Data/naca_embedded.mesh -v 5`, you will obtain :

- detailed quality histograms (see figure 2) ;
- detailed remeshing steps ;
- edge length histogram (see figure 3).

3.2 Mesh improvement with edge length preservation : -optim option

Open your output mesh in Gmsh : by default, Mmg tries to create a mesh that respect the asked boundary approximation (-hausd option, 0.1 by default), the maximal ratio between two adjacent edges
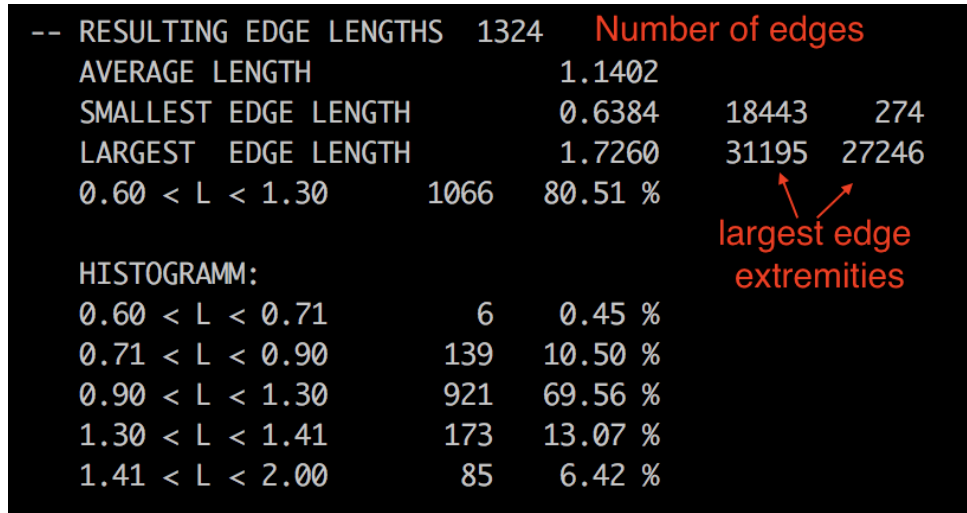
FIGURE 3 – Edge length histogram.

(`-hgrad` option, 1.3 by default) and that contains the smallest possible number of points.

If you want to preserve the edge length of the input mesh, you can run Mmg with the `-optim` option :

```
$ mmg2d_O3 Data/naca_embedded.mesh -optim -v 5
```

Open your mesh in Gmsh and check that the edge length ar preserved. You can compare the input and output quality histograms.

If you want, you can play with the some other options : try for example to disable the gradation (`-hgrad -1`) or to ask for a better boundary approximation (`-hmax 0.5 -hausd 0.0005`).

4 Mesh adaptation to a size map

You can provide to Mmg a size map in a `.sol` file. This file lists, for each node, the prescribed edge length. In isotropic case, the file contains 1 scalar data per node (s). In anisotropic case, it contains a metric tensor :

$$M = R \, \Lambda \, R^T$$

With :

- $\Lambda = (\lambda_i)_i$ a diagonal matrix such as $\lambda_i = 1/s^2$ with s the wanted edge length.
- $R = (r_{ij})_{ij}$, an orthonormal matrix such as the r_j vectors gives the direction in which we want the $1/s^2$ length.

See figure 4 to see an explanation of the .sol file format for an isotropic and an anisotropic size map.

You will find in the `Data` directory 2 size maps, `naca_iso.sol` and `naca_aniso.sol`. Try to adapt your mesh to each map. For example, for the isotropic map :

```
$ mmg2d_O3 Data/naca_embedded.mesh -sol naca_iso.sol -hausd 0.001 -v 5
```

Again, you can play with the gradation parameter.

5 Size map computation to control the error of interpolation of an analytic function over the mesh

5.1 Computation of the nodal values of a 2D analytic function

1. Choose a function, for example, a Gaussian function : $f(x, y) = 10 * exp(x * x/30 + y * y/30)$ or $f(x, y) = sin(x)$;
2. Compute its nodal values on the mesh nodes (you can load the Gmsh mesh in Octave) ;
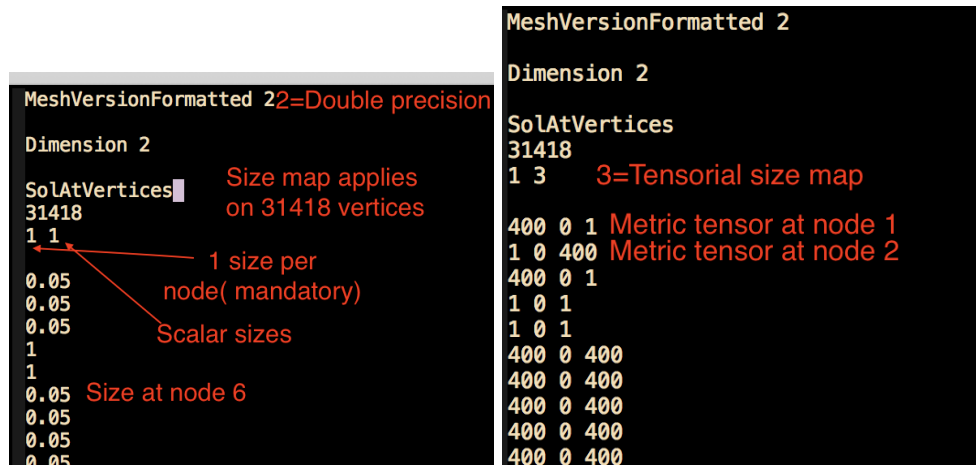
3

FIGURE 4 – Isotropic (left) and anisotropic (right) size maps at `Medit` file format.

3. Save this values in a .sol file;

4. Convert your .mesh/.sol file at Gmsh format with the following command :

```
$ mmg2d_O3 Data/naca_embedded.mesh -sol gauss.sol -noinsert -noswap
   -nomove Data/vizu_gmsh.msh
```

5. Now you can open your mesh inside Gmsh to check the solution data.

5.2 Computation of an anisotropic size map to control the interpolation error over the mesh

1. compute $M_V = \frac{2}{9\epsilon} H_u(V)$ with V a mesh node;

2. save M_V in a `.sol file`.

Adapt your mesh and check the results.

5.3 Computation of an isotropic size map to control the interpolation error over the mesh

1. Compute M_V;

2. compute the eigenvalues of M_V;

3. compute the maximal eigenvalue λ;

4. Compute $s_v = \frac{1}{\sqrt{\lambda}}$;

5. save s_v in a `.sol file`.

Adapt your mesh and check the results.