meshtools Octave package, User's Guide*

version 0.1.4

François Cuvelier†

December 21, 2022

Abstract

The experimental meshtools Octave package contains some simplicial meshes given by their vertices array **q** and connectivity array **me**. Theses meshes can be easily used in other Octave codes for debugging or testing purpose.

0 Contents

*LaTeX manual, revision 0.1.4.a, compiled with Octave 7.3.0, and packages `fc-meshtools`[0.1.4], `fc-tools`[0.0.34], `fc-bench`[0.1.3], `fc-amat`[0.1.3]

†LAGA, UMR 7539, CNRS, Université Paris 13 - Sorbonne Paris Cité, Université Paris 8, 99 Avenue J-B Clément, F-93430 Villetaneuse, France, cuvelier@math.univ-paris13.fr.

1 Introduction

A simplicial mesh is given by its vertices array **q** and its connectivity array **me**. For demonstration purpose, some simplicial meshes are given in this package and stored in the `+fc_meshtools/data` directory. They can be load by using the functions fc_meshtools.simplicial.getMesh2D, fc_meshtools.simplicial.getMesh3D or fc_meshtools.simplicial.getMesh3Ds. Here are the kind of simplicial meshes present in this package:

- a triangular mesh in dimension 2, made with 2-simplices (ie. triangles),
- a tetrahedral mesh in dimension 3, made with 3-simplices (ie. tetrahedron),
- a triangular mesh in dimension 3 (surface mesh), made with 2-simplices,
- a line mesh in dimension 2 or 3 made with 1-simplices (ie. lines).

This package was only tested on Ubuntu 22.04.1 with Octave 7.3.0.

2 Installation

2.1 Installation automatic, all in one (recommanded)

For this method, one just have to get/download the install file

ofc_meshtools_install.m

or get it on the dedicated web page. Thereafter, one run it under Octave. This command download, extract and configure the *fc-meshtools* and the required *fc-tools* package in the current directory.

For example, to install this package in `~/Octave` directory, one have to copy the file ofc_meshtools_install.m in the `~/Octave` directory. Then in a Octave terminal run the following commands to install the fc meshtools package

```
>> cd ~/Octave
>> ofc_meshtools_install
```

There is the output of the `ofc_meshtools_install` command on a Linux computer:

```
Parts of the <fc-meshtools> Octave package.
Copyright (C) 2018-2022 F. Cuvelier <cuvelier@math.univ-paris13.fr>

1- Downloading and extracting the packages
2- Setting the <fc-meshtools> package
Write in ~/Octave/fc-meshtools-full/fc_meshtools-0.1.4/configure_loc.m
    ...
3- Using packages :
   ->            fc-tools : 0.0.34
   ->            fc-bench : 0.1.3
   ->             fc-amat : 0.1.3
 with        fc-meshtools : 0.1.4
*** Using instructions
   To use the <fc-meshtools> package:
   addpath('~/Octave/fc-meshtools-full/fc_meshtools-0.1.4')
   fc_meshtools.init()

   See ~/Octave/ofc_meshtools_set.m
```

The complete package (which included the `fc-tools` package) is stored in the directory `~/Octave/fc-meshtools-full` and, for each Octave session, one have to set the package by:

```
>> addpath('~/Octave/fc-meshtools-full/fc-meshtools-0.1.4')
>> fc_meshtools.init()
```

If it's the first time the `fc_meshtools.init()` function is used, then its output is

```
Try to use default parameters!
 Use fc_tools.configure to configure.
Write in ~/Octave/fc-meshtools-full/fc_tools-0.0.34/configure_loc.m ...
Try to use default parameters!
 Use fc_bench.configure to configure.
Write in ~/Octave/fc-meshtools-full/fc_bench-0.1.3/configure_loc.m ...
Try to use default parameters!
 Use fc_amat.configure to configure.
Write in ~/Octave/fc-meshtools-full/fc_amat-0.1.3/configure_loc.m ...
Using fc_meshtools[0.1.4] with fc_tools[0.0.34], fc_bench[0.1.3], fc_amat
    [0.1.3].
```

Otherwise, the output of the `fc_meshtools.init()` function is

```
Using fc_meshtools[0.1.4] with fc_tools[0.0.34], fc_bench[0.1.3], fc_amat
    [0.1.3].
```

For **uninstalling**, one just have to delete directory

`~/Octave/fc-meshtools-full`

3 Simplicial meshes

The functions fc_meshtools.simplicial.getMesh2D, gfc_meshtools.simplicial.etMesh3D and fc_meshtools.simplicial.getMesh3Ds return a mesh vertices array $\mathbf{q}$, a mesh elements connectivity array associated with the input argument d (simplex dimension) and the indices array toGlobal. The vertices array $\mathbf{q}$ is a dim-by-n_{q} array where dim is the space dimension (2 or 3) and n_{q} the number of vertices. The connectivity array **me** is a $(d+1)$-by-n_{me} array where n_{me} is the number of mesh elements and $0 \leqslant d \leqslant dim$ is the simplicial dimension:

- $d = 0$: points,
- $d = 1$: lines,
- $d = 2$: triangle,
- $d = 3$: tetrahedron.

So we can use theses functions to obtain

- 3D mesh: getMesh3D(3) (*main* mesh), getMesh3D(2), getMesh3D(1), getMesh3D(0),
- 3D surface mesh: getMesh3Ds(2) (*main* mesh), getMesh3Ds(1) , getMesh3Ds(0),
- 2D mesh: getMesh2D(2) (*main* mesh), getMesh2D(1), getMesh2D(0).

For example,

- [q3,me3,toGlobal3]=fc_meshtools.simplicial.getMesh3D(3) return a 3-simplicial mesh (main mesh) in space dimension $dim = 3$,
- [q2,me2,toGlobal2]=fc_meshtools.simplicial.getMesh3D(2) return a 2-simplicial mesh in space dimension $dim = 3$.

The third output are indices of the vertices in the *main* mesh:
q3 (:, toGlobal2) == q2

4 Functions

4.1 getMesh functions

Returns a vertices array q, a connectivity array me and an indices array toGlobal.

Description

```
[q,me,toGlobal]=fc_meshtools.simplicial.getMesh3D(d)
```

```
[q,me,toGlobal]=fc_meshtools.simplicial.getMesh3Ds(d)
```

```
[q,me,toGlobal]=fc_meshtools.simplicial.getMesh2D(d)
```

Returns a vertices array q, a connectivity array me and an indices array toGlobal depending on the value of the d. For a 3D mesh, d $\in [\![0, 3]\!]$, and for a 2D or 3Ds mesh, d $\in [\![0, 2]\!]$.

In Listing 2, some examples are provided.

Listing 1: : examples of fc_meshtools.simplicial.getMesh3D function usage

```
[q2,me2,toG2]=fc_meshtools.simplicial.getMesh3D(2);
[q3,me3,toG3]=fc_meshtools.simplicial.getMesh3D(3);
whos('q2','me2','toG2','q3','me3','toG3')
fprintf('Error: %.5e\n',norm(q3(:,toG2) - q2,Inf))
```

Output

```
Variables visible from the current scope:

variables in scope: top scope

  Attr  Name       Size                     Bytes  Class
  ====  ====       ====                     =====  =====
        q2         3x7533                  180792  double
        me2        3x15074                 361776  double
        toG2       1x7533                   60264  double
        q3         3x17416                 417984  double
        me3        4x84302                2697664  double
        toG3       1x17416                 139328  double

Total is 482226 elements using 3857808 bytes

Error: 0.00000e+00
```

4.2 Volumes function

Syntaxe Returns all the element volumes of a mesh given by a vertices array q and a connectivity array me. One can refer to [1] for computationnal details.

Description

vols=fc_meshtools.simplicial.Volumes(q,me)

vols(k) is the volume of the k-th mesh element where its vertices are the columns of q(:,me(:,k)).

In Listing 2, some examples are provided.

Listing 2: : examples of fc_meshtools.simplicial.Volumes function usage

```
[q,me,toG]=fc_meshtools.simplicial.getMesh3D(2);
vols=fc_meshtools.simplicial.Volumes(q,me);
whos('q','me','toG','vols')
```

Output

```
Variables visible from the current scope:

variables in scope: top scope

  Attr  Name       Size                     Bytes  Class
  ====  ====       ====                     =====  =====
        q          3x7533                  180792  double
        me         3x15074                 361776  double
        toG        1x7533                   60264  double
        vols       1x15074                 120592  double

Total is 90428 elements using 723424 bytes
```

4.3 Gradient of barycentric coordinates

Syntaxe Returns all the gradients of barycentric coordinates of each element of a mesh given by a vertices array q and a connectivity array me. One can refer to [1] for computationnal details.

Description

G=fc_meshtools.simplicial.GradBaCo(q,me)

G(k,i,:) is the gradient of the i-th barycentric coordinate of the k-th mesh element.

In Listing 3, some examples are provided.

Listing 3: : examples of fc_meshtools.simplicial.GradBaCo function usage

```
[q,me,toG]=fc_meshtools.simplicial.getMesh3D(3);
G=fc_meshtools.simplicial.GradBaCo(q,me);
whos('q','me','toG','G')
```

Output

```
Variables visible from the current scope:

variables in scope: top scope

  Attr   Name        Size                     Bytes  Class
  ====   ====        ====                     =====  =====
         q           3x17416                 417984  double
         me          4x84302                2697664  double
         toG         1x17416                 139328  double
         G       84302x4x3                  8092992  double

Total is 1418496 elements using 11347968 bytes
```

4 References

[1] F. Cuvelier. Exact integration for products of power of barycentric coordinates over d-simplexes in R^n. http://hal.archives-ouvertes.fr/hal-00931066v1, June 2018. preprint.

Informations for git maintainers of the fc meshtools Octave package

git informations on the packages used to build this manual

```
-------------------------------------------------
   name : fc-meshtools
    tag : 0.1.4.a
 commit : 57826462a8280dec5aa1eff0d424877097187b41
   date : 2022-12-19
   time : 08-27-27
 status : 0
-------------------------------------------------
   name : fc-tools
    tag : 0.0.34
 commit : 34edf9393425222cdb72914e51e6465f50aa6760
   date : 2022-12-21
   time : 09-28-20
 status : 0
-------------------------------------------------
   name : fc-bench
    tag : 0.1.3
 commit : 6c8969bb49b90c7cae34e70c88dd2f968766376b
   date : 2022-12-17
   time : 13-58-43
 status : 0
-------------------------------------------------
   name : fc-amat
    tag : 0.1.3
 commit : 90dbaa2839188cf8e01629817e5f8ba1b7188052
   date : 2022-12-19
   time : 07-41-33
 status : 0
-------------------------------------------------
```

git informations on the LaTeX package used to build this manual

```
-------------------------------------------------
   name : fctools
    tag :
 commit : c9a33ce7b4dacf90f66e5e49856e69afa1dac0a3
   date : 2022-12-17
   time : 07:57:20
 status : 1
-------------------------------------------------
```

Using the remote configuration repository:

url	ssh://lagagit/MCS/Cuvelier/Matlab/fc-config
commit	1ddb965ef261410ecee4ce178be4ceb85939de04