FC-HYPERMESH Matlab toolbox, User's Guide *

François Cuvelier†

December 24, 2016

Abstract

This object-oriented Matlab toolbox allows to mesh any d-orthotopes (hyperrectangle in dimension d) and their m-faces by simplices or orthotopes. It was created to show the implementation of the algorithms of the report Vectorized algorithms for regular tessellations of d-orthotopes and their faces. The FC-HYPERMESH toolbox uses Matlab objects and is provided with meshes visualisation tools for dimension leather or equal to 3.

Contents

*Compiled with Matlab 2015b

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

This work was partially supported by ANR Dedales.

1 Introduction

The FC-HYPERMESH toolbox contains a simple class object OrthMesh which permits, in any dimension $d \geqslant 1$, to obtain a simplicial mesh or orthotope mesh with all their m-faces, $0 \leqslant m < d$. It is also possible with the method function plot of the class object OrthMesh to represent a mesh or its m-faces for $d \leqslant 3$.

In the folowing section, the class object OrthMesh is presented. Thereafter some warning statements on the memory used by these objects in high dimension are given. Finally computation times for orthotope meshes and simplicial meshes are given in dimension $d \in [\![1, 5]\!]$.

2 Installation of the fc-hypermesh toolbox

For a better presentation, the following notation is used in various linux command

<HREF> equal to
http://www.math.univ-paris13.fr/~cuvelier/software/codes/

2.1 Automatic installation, all in one (recommanded)

For this method, one just have to get/download the install file mfc_hypermesh_install.m and run it under Matlab. This command download, extract and configure the *fc-hypermesh* and the required *fc-tools* toolbox in the current directory.

For example, to install this toolbox in directory ~/Matlab/toolboxes, in a terminal one can do:

Terminal
```
mkdir -p ~/Matlab/toolboxes
cd ~/Matlab/toolboxes
wget <HREF>/fc-hypermesh/0.0.2/mfc_hypermesh_install.m
```

Then in a Matlab terminal run the following commands

Command Window
```
>> cd ~/Matlab/toolboxes
>> mfc_oogmsh_install
```

There is the output of the mfc_oogmsh_install command:

```
1- Downloading and extracting the toolboxes
   -> <fc-tools>[0.0.14] ... OK
   -> <fc-hypermesh>[0.0.2] ... OK
2- Setting the <fc-hypermesh> toolbox
Write in ~/Matlab/toolboxes/fc-hypermesh-full/mfc-hypermesh-0.0.2/configure_loc.m ...
  -> done
3- Using the <fc-hypermesh> toolbox
   Under Matlab:
```

```
  addpath('~/Matlab/toolboxes/fc-hypermesh-full/mfc-hypermesh-0.0.2')
  fc_hypermesh.init()

See ~/Matlab/toolboxes/mfc_hypermesh_set.m
```

The complete toolbox (i.e. with the FC-TOOLS toolbox included) is stored in the directory `~/Matlab/toolboxes/fc-hypermesh-full` and, for each Matlab session, one have to set the toolbox by:

Command Window

```
▷ addpath('~/Matlab/toolboxes/fc-hypermesh-full/mfc-hypermesh-0.0.1')
▷ fc_hypermesh.init()
```

For **uninstalling**, one just have to delete directory `~/Matlab/toolboxes/fc-hypermesh-full`

2.2 Manual installation

- Installation of the FC-TOOLS toolbox by downloading/extracting in `~/Toolboxes` directory

  Terminal

  ```
  cd ~/Toolboxes
  wget <HREF>/fc-tools/0.0.14/mfc-tools-0.0.14.tar.gz
  tar zxf mfc-tools-0.0.14.tar.gz
  ```

  Then the FC-TOOLS toolbox directory is `~/Toolboxes/mfc-tools-0.0.14`

- Installation of the FC-HYPERMESH toolbox by downloading/extracting in `~/Toolboxes` directory

  Terminal

  ```
  cd ~/Toolboxes
  wget <HREF>/fc-hypermesh/0.0.2/mfc-hypermesh-0.0.2.tar.gz
  tar zxf mfc-hypermesh-0.0.2.tar.gz
  ```

  Then the FC-HYPERMESH toolbox directory is `~/Toolboxes/mfc-hypermesh-0.0.1`

- Configuration of the FC-HYPERMESH toolbox: one must specify it where to found the FC-TOOLS toolbox. So under Matlab we do:

  Command Window

  ```
  ▷ cd ~/Toolboxes/mfc-hypermesh-0.0.2
  ▷ fc_hypermesh.configure('fc_tools_dir','~/Toolboxes/mfc-tools
      -0.0.14')
  Write in /home/cuvelier/tmp/Matlab/mfc-hypermesh-0.0.2/configure_loc.m ...
  -> done
  ```

- Using the FC-HYPERMESH toolbox: in any Matlab session, one must :

  Command Window

  ```
  >> addpath('~/Toolboxes/mfc-hypermesh-0.0.2)
  >> fc_hypermesh.init()
  ```

3 Using the fc-hypermesh toolbox

First of all, the main class object `OrthMesh` is presented. A complete report was written to described the algorithms used []. Thereafter some usage samples are given.

3.1 Class object `OrthMesh`

The aim of the class object `OrthMesh` is to efficiently create an object which contains a mesh of a d-orthotope and all its m-face meshes. An elementary mesh class object `EltMesh` is used to store only one mesh, the main mesh as well as any of the m-face meshes. This class `EltMesh` also simplify (for me) the codes writing. Its fields are the following:

- d, space dimension
- m, kind of mesh ($m = \mathrm{d}$ for the main mesh)
- type, 0 for simplicial mesh or 1 for orthotope mesh
- $\mathrm{n_q}$, number of vertices
- q, vertices array of dimension d-by-$\mathrm{n_q}$
- $\mathrm{n_{me}}$, number of mesh elements
- me, connectivity array of dimension $(\mathrm{d}+1)$-by-$\mathrm{n_{me}}$ for simplices elements or 2^{d}-by-$\mathrm{n_{me}}$ for orthotopes elements
- `toGlobal`, index array linking local array q to the one of the main mesh
- label, name/number of this elementary mesh
- color, color of this elementary mesh (for plotting purpose)

Let the d-orthotope defined by $[a_1, b_1] \times \cdots \times [a_{\mathrm{d}}, b_{\mathrm{d}}]$. The class object `OrthMesh` corresponding to this d-orthotope contains the main mesh and all its m-face meshes, $0 \leqslant m < \mathrm{d}$. Its Fields are the following

- d: space dimension
- type: string 'simplicial' or 'orthotope' mesh
- Mesh: main mesh as an `EltMesh` object
- Faces: list of arrays of `EltMesh` objects such that Faces(1) is an array of all the $(\mathrm{d}-1)$-face meshes, Faces(2) is an array of all the $(\mathrm{d}-2)$-face meshes, and so on
- box: a d-by-2 array such that $\mathrm{box}(i,1) = a_i$ and $\mathrm{box}(i,2) = b_i$.

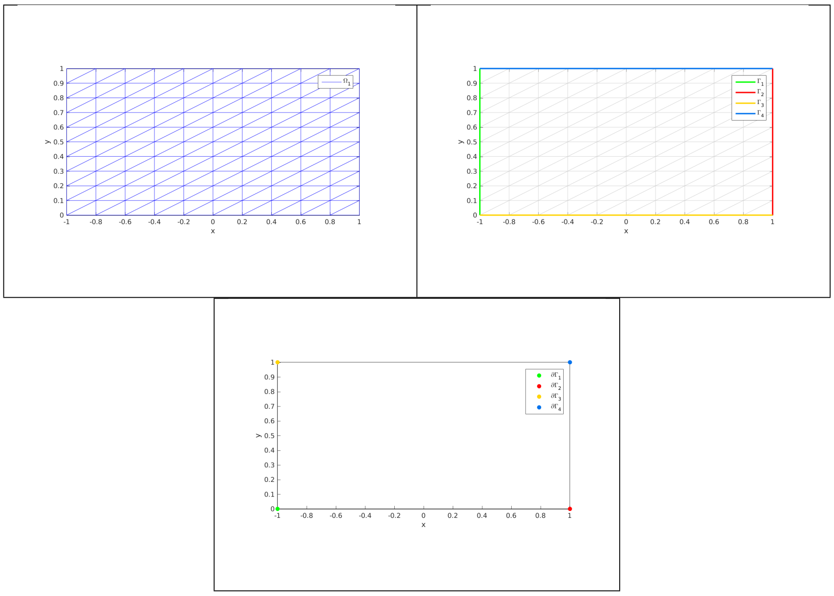
The `OrthMesh` constructor is

$$Oh \leftarrow \text{ORTHMESH}(\text{d}, \mathbf{N}, < \text{box} >, < \text{type} >)$$

where $\mathbf{N}$ is either a 1-by-d array such that $\mathbf{N}(i)$ is the number of discretization for $[a_i, b_i]$ or either an integer if the the number of discretization is the same in all space directions. The optional parameter box previously described as for default value $a_i = 0$ and $b_i = 1$. The default value for optional parameter type is 'simplicial' and ortherwise 'orthotope' can be used.

3.2 2d-orthotope meshing by simplices

In Listing 22, an `OrthMesh` object is built under Matlabfor the orthotope $[-1, 1] \times [0, 1] \times [0, 2]$ with simplicial elements and $\mathbf{N} = (10, 5, 10)$. The main mesh and all the m-face meshes of the resulting object are plotted.

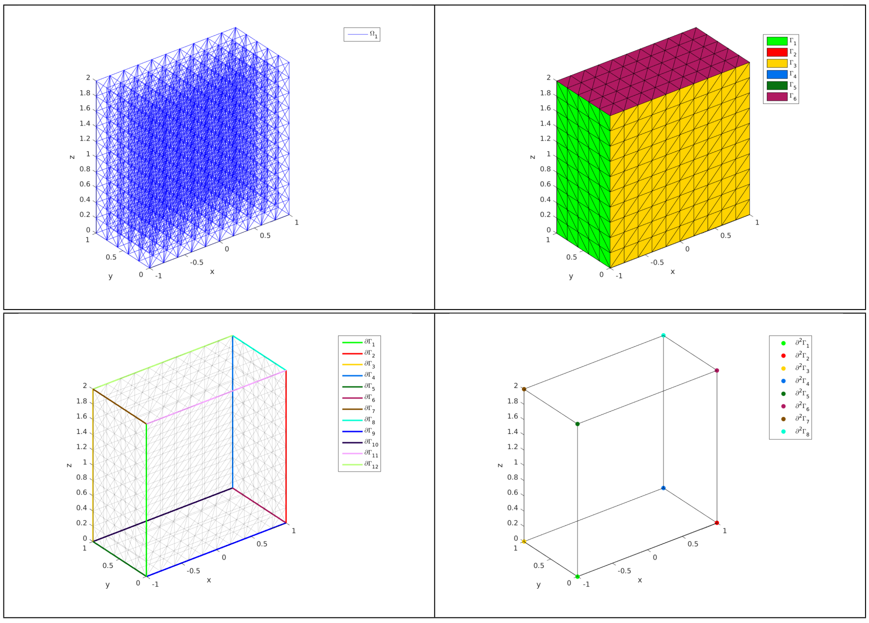


```
Oh=OrthMesh(2,10,'box',[-1,1;0,1])
% plot the main mesh
figure(1)
Oh.plot('legend',true)
axis equal;xlabel('x');ylabel('y')
% plot the 1-face meshes
figure(2)
Oh.plot('color',[0.8,0.8,0.8])
hold on
Oh.plot('m',1,'Linewidth',2,'legend',true)
axis equal;xlabel('x');ylabel('y');
% plot the 0-face meshes
figure(3)
Oh.plot('m',1,'color','k')
hold on
Oh.plot('m',0,'legend',true)
axis equal;xlabel('x');ylabel('y');
```

Listing 1: 2D simplicial `OrthMesh` object with Matlab 2015b, main mesh (upper left), 1-face meshes (upper right), and 0-face meshes (bottom)

3.3 3d-orthotope meshing by simplices

In Listing 22, an `OrthMesh` object is built under Matlabfor the orthotope $[-1, 1] \times [0, 1] \times [0, 2]$ with simplicial elements and $\mathbf{N} = (10, 5, 10)$. The main mesh and all the m-face meshes of the resulting object are plotted.

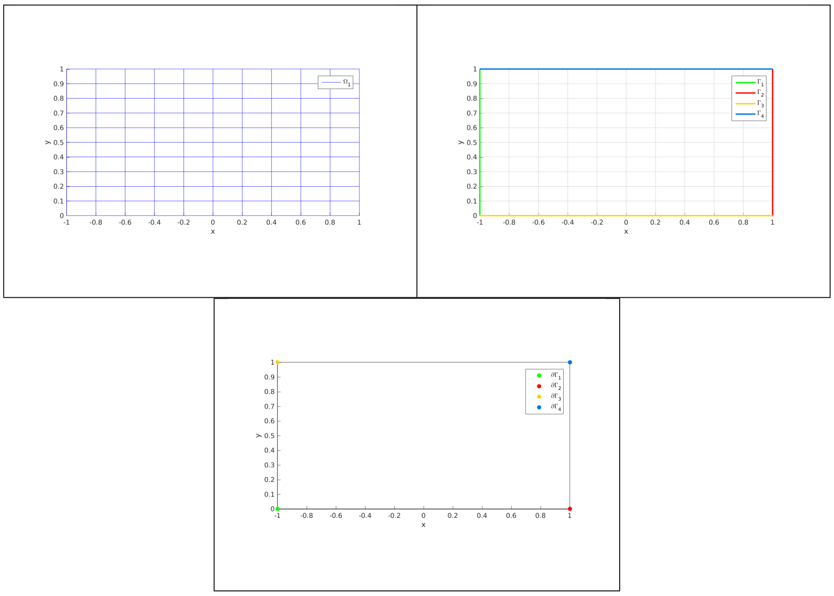


```
Oh=OrthMesh(3,[10,5,10],'box',[-1,1;0,1;0,2])
% plot the main mesh
figure(1)
Oh.plot('legend',true)
axis equal;xlabel('x');ylabel('y');zlabel('z')
% plot the 2-face meshes
figure(2)
Oh.plot('m',2,'legend',true)
axis equal;xlabel('x');ylabel('y');zlabel('z')
% plot the 1-face meshes
figure(3)
Oh.plot('m',2,'color',[0.8,0.8,0.8],'EdgeAlpha',0.2, ...
    'FaceColor','none')
hold on
Oh.plot('m',1,'Linewidth',2,'legend',true)
axis equal;xlabel('x');ylabel('y');zlabel('z')
% plot the 0-face meshes
figure(4)
Oh.plot('m',1,'color','k')
hold on
Oh.plot('m',0,'legend',true)
axis equal;xlabel('x');ylabel('y');zlabel('z')
```

Listing 2: 3D simplicial `OrthMesh` object with Matlab 2015b, main mesh (upper left), 2-face meshes (upper right), 1-face meshes (bottom left) and 0-face meshes (bottom right)

3.4 2d-orthotope meshing by orthotopes

In Listing 22, an `OrthMesh` object is built under Matlabfor the orthotope $[-1,1]\times[0,1]\times[0,2]$ with orthotope elements and $\mathbf{N}=(10,5,10)$. The main mesh and all the m-face meshes of the resulting object are plotted.

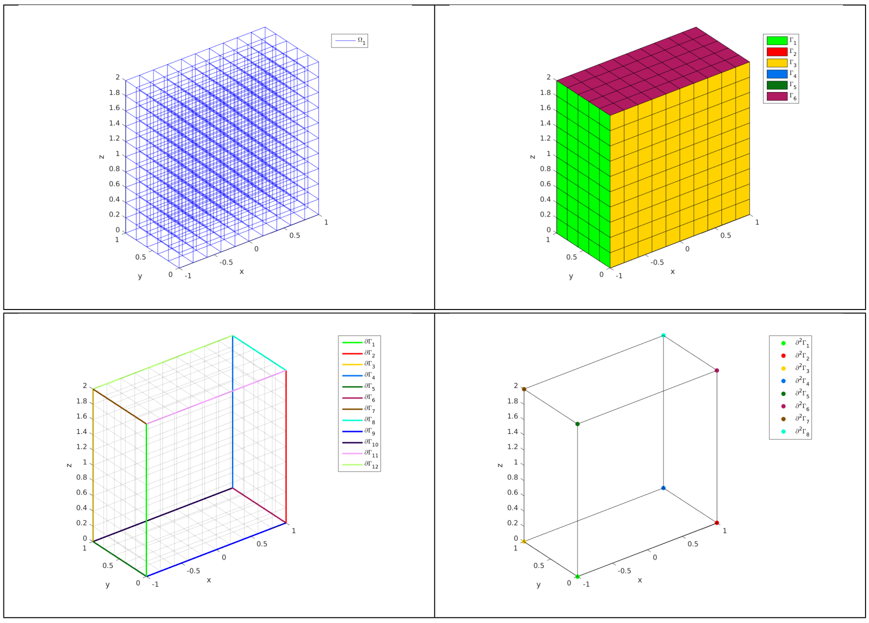


```
Oh=OrthMesh(2,10,'box',[-1,1;0,1],'type','orthotope')
% plot the main mesh
figure(1)
Oh.plot('legend',true)
axis equal;xlabel('x');ylabel('y')
% plot the 1-face meshes
figure(2)
Oh.plot('color',[0.8,0.8,0.8])
hold on
Oh.plot('m',1,'Linewidth',2,'legend',true)
axis equal;xlabel('x');ylabel('y');
% plot the 0-face meshes
figure(3)
Oh.plot('m',1,'color','k')
hold on
Oh.plot('m',0,'legend',true)
axis equal;xlabel('x');ylabel('y');
```

Listing 3: 2D orthotope `OrthMesh` object with Matlab 2015b, main mesh (upper left), 1-face meshes (upper right), and 0-face meshes (bottom)

3.5 3d-orthotope meshing by orthotopes

In Listing 22, an `OrthMesh` object is built under Matlabfor the orthotope $[-1,1]\times[0,1]\times[0,2]$ with orthotope elements and $\mathbf{N}=(10,5,10)$. The main mesh and all the m-face meshes of the resulting object are plotted.

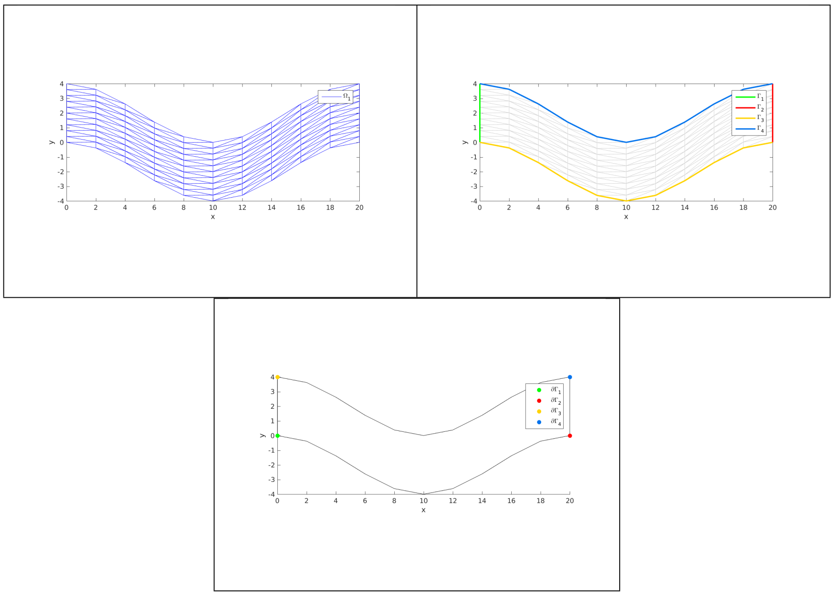


```
Oh=OrthMesh(3,[10,5,10],'box',[-1,1;0,1;0,2], ...
    'type','orthotope')
% plot the main mesh
figure(1)
Oh.plot('legend',true)
axis equal;xlabel('x');ylabel('y');zlabel('z')
% plot the 2-face meshes
figure(2)
Oh.plot('m',2,'legend',true)
axis equal;xlabel('x');ylabel('y');zlabel('z')
% plot the 1-face meshes
figure(3)
Oh.plot('m',2,'color',[0.8,0.8,0.8],'EdgeAlpha',0.2, ...
    'FaceColor','none')
hold on
Oh.plot('m',1,'Linewidth',2,'legend',true)
axis equal;xlabel('x');ylabel('y');zlabel('z')
% plot the 0-face meshes
figure(4)
Oh.plot('m',1,'color','k')
hold on
Oh.plot('m',0,'legend',true)
axis equal;xlabel('x');ylabel('y');zlabel('z')
```

Listing 4: 3D orthotope `OrthMesh` object with Matlab 2015b, main mesh (upper left), 2-face meshes (upper right), 1-face meshes (bottom left) and 0-face meshes (bottom right)

For example, the following 2D geometrical transformation allows to deform the reference unit hypercube.

$$\begin{aligned} [0,1]\times[0,1] &\longrightarrow \mathbb{R}^2 \\ \begin{pmatrix} x \\ y \end{pmatrix} &\longrightarrow F(x,y) = \begin{pmatrix} 20\,x \\ 2\,(2\,y-1+\cos(2\pi x)) \end{pmatrix} \end{aligned}$$



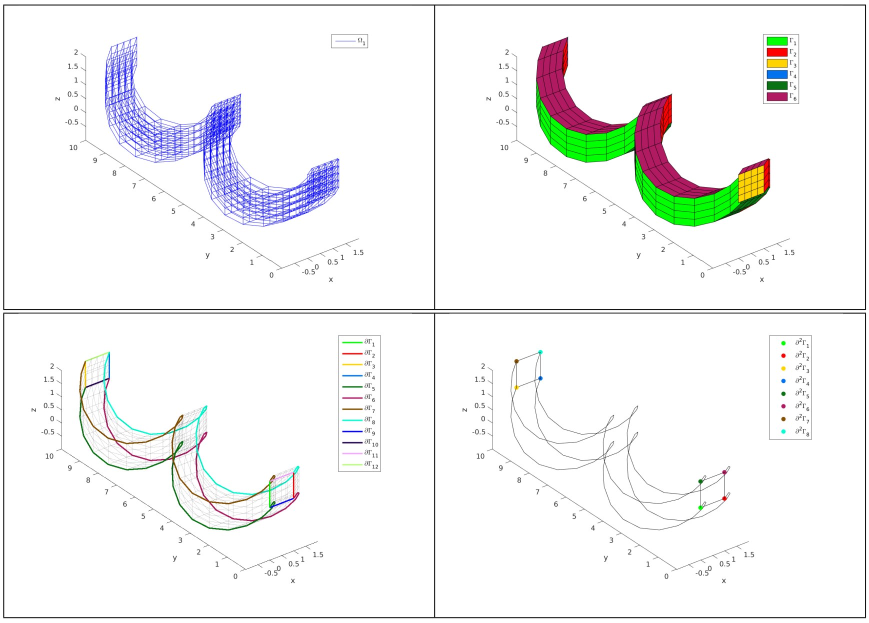
```
Oh=OrthMesh(2,10,'mapping',@(q) [20*q(1,:); ...
    2*(2*q(2,:)-1+cos(2*pi*q(1,:)))]);
% plot the main mesh
figure(1)
Oh.plot('legend',true)
axis equal;xlabel('x');ylabel('y')
% plot the 1-face meshes
figure(2)
Oh.plot('color',[0.8,0.8,0.8])
hold on
Oh.plot('m',1,'Linewidth',2,'legend',true)
axis equal;xlabel('x');ylabel('y');
% plot the 0-face meshes
figure(3)
Oh.plot('m',1,'color','k')
hold on
Oh.plot('m',0,'legend',true)
axis equal;xlabel('x');ylabel('y');
```

Listing 5: Mapping of a 2D simplicial `OrthMesh` object with Matlab 2015b, main mesh (upper left), 1-face meshes (upper right), and 0-face meshes (bottom)

3.7 3d-orthotope meshing by orthotopes

For example, the following 3D geometrical transformation allows to deform the reference unit hypercube.

$$\begin{array}{ccc} [0,1]\times[0,1]\times[0,1] & \longrightarrow & \mathbb{R}^2 \\ \begin{pmatrix} x \\ y \\ z \end{pmatrix} & \longrightarrow & F(x,y,y) = \begin{pmatrix} x+\sin(4\pi y) \\ 10y \\ z+\cos(4\pi y) \end{pmatrix} \end{array}$$



```
Map=@(q) [q(1,:)+ sin(4*pi*q(2,:)); 10*q(2,:); ...
     q(3,:)+cos(4*pi*q(2,:))];
Oh=OrthMesh(3,[4,25,4],'mapping',Map,'type','orthotope');
% plot the main mesh
figure(1)
Oh.plot('legend',true)
xlabel('x');ylabel('y');zlabel('z');axis image;axis equal;
% plot the 2-face meshes
figure(2)
Oh.plot('m',2,'legend',true)
xlabel('x');ylabel('y');zlabel('z');axis image;axis equal;
% plot the 1-face meshes
figure(3)
Oh.plot('m',2,'color',[0.8,0.8,0.8],'EdgeAlpha',0.2, ...
     'FaceColor','none')
hold on
Oh.plot('m',1,'Linewidth',2,'legend',true)
xlabel('x');ylabel('y');zlabel('z');axis image;axis equal;
% plot the 0-face meshes
figure(4)
Oh.plot('m',1,'color','k')
hold on
Oh.plot('m',0,'legend',true)
xlabel('x');ylabel('y');zlabel('z');axis image;axis equal;
```

Listing 6: Mapping of a 3D orthotope `OrthMesh` object with Matlab 2015b, main mesh (upper left), 2-face meshes (upper right), 1-face meshes (bottom left) and 0-face meshes (bottom right)