Proceedings of FEniCS 2021

22–26 March 2021

Editors

Igor Baratta

Jørgen S. Dokken

Chris Richardson

Matthew W. Scroggs

Working with complex meshes: The mesh processing pipeline

U Meenu Krishnan
(https://computationalmechanics.in/rajib_teams/u-meenu-krishnan/), Indian Institute of Technology, Roorkee, India

Abhinav Gupta, Indian Institute of Technology, Roorkee, India

Rajib Chowdhury, Indian Institute of Technology, Roorkee, India

24 March 2021

In any finite element analysis, it is very important to have an efficient tool to generate the mesh. In our work, we use Gmsh as a meshing software, and it is observed that the main problem with the current implementation of the mesh processing pipeline is that it is facing difficulties to handle models with complex geometries comprising of many boundaries and loading conditions; this means it has more number of physical groups. We use these physical groups to assign material properties, loads, or boundary conditions to the model.

The main objective of this talk is to explain the complete mesh processing pipeline for complex geometry. Starting from modeling of a mesh geometry and marking different boundary conditions in Gmsh, then will discuss about the benefits of using Mesh functions and Mesh Value Collections in FEniCS, and we will also discuss how to use the number tags from the XDMF file for defining various subdomains required for load and boundary conditions applications, and at the end how to visualize the XDMF file for validating if all the boundary conditions and loads are applied appropriately.

This talk will help the community to understand the complete mesh processing pipeline for a complex mesh geometry with increased efficiency.

You can cite this talk as:

U Meenu Krishnan, Abhinav Gupta, and Rajib Chowdhury. "Working with complex meshes: The mesh processing pipeline". In: *Proceedings of FEniCS 2021, online, 22–26 March* (eds Igor Baratta, Jørgen S. Dokken, Chris Richardson, Matthew W. Scroggs) (2021), 430–438. DOI: 10.6084/m9.figshare.14495403.

BibTeX for this citation can be found at https://mscroggs.github.io/fenics2021/talks/krishnan.html.

Working with Complex Meshes : The Mesh Processing Pipeline

FEniCS-2021

U. Meenu Krishnan
umeenukrishnan@ce.iitr.ac.in
Indian Institute of Technology, Roorkee

Abhinav Gupta,
Indian Institute of Technology, Roorkee
Dr. Rajib Chowdhury,
Indian Institute of Technology, Roorkee

Motivation

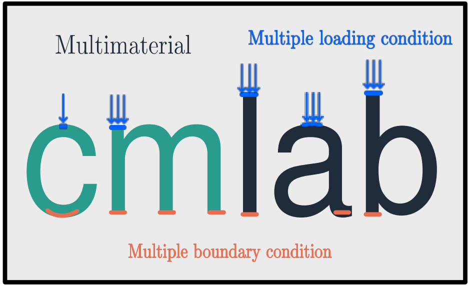
We wish to use FEniCS with complex geometries

In practice, a real world engineering structure could have :

1. Multiple loading areas
2. Multiple boundary conditions
3. Multiple materials

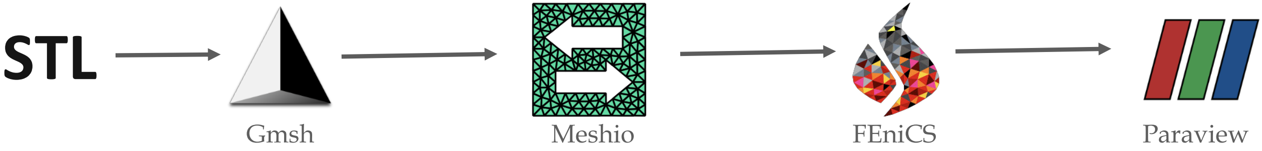
Thus can have 10 – 100's of marked regions in the mesh

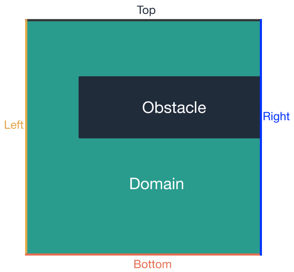
Problem: This could result in human error in the process of modelling



"Output of a simulation is as good as the accuracy in the mathematical modelling"

Preferred mesh processing pipeline





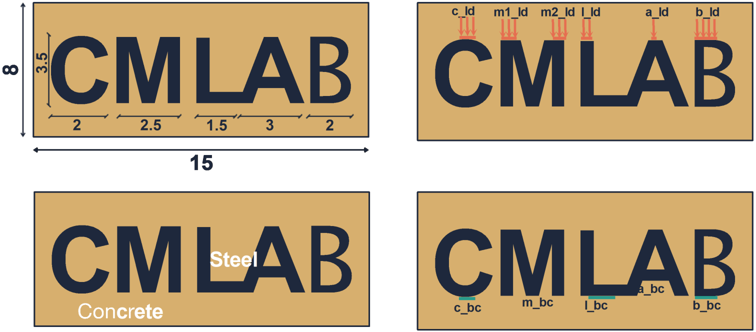

```
$PhysicalNames
6
1 3 "Top"
1 4 "Right"
1 5 "Left"
1 6 "Bottom"
2 1 "Domain"
2 2 "Obstacle"
$EndPhysicalNames
```

```
# Define Dirichlet boundary conditions at top and bottom boundaries
bcs = [DirichletBC(V, 5.0, boundaries, 2),
       DirichletBC(V, 0.0, boundaries, 4)]

# Define new measures associated with the interior domains and
# exterior boundaries
dx = Measure("dx")[domains]
ds = Measure("ds")[boundaries]

# Define variational form
F = (inner(a0*grad(u), grad(v))*dx(0) + inner(a1*grad(u), grad(v))*dx(1)
     - g_L*v*ds(1) - g_R*v*ds(3)
     - f*v*dx(0) - f*v*dx(1))
```

Basic design approach

Engineering structures are accompanied with schematic drawings

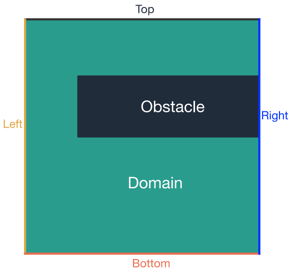
1. Layout of the structure
2. Details of boundary conditions
3. Details of loading condition
4. Details about material properties



Aim : To use the same tag names which is in the schematic drawing in the FEniCS implementation

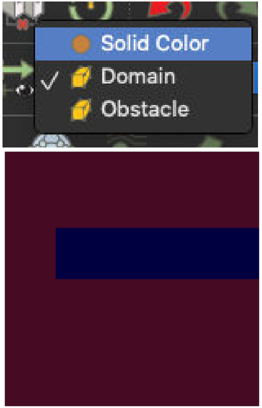
Desired mesh processing pipeline

Gmsh → Meshio + Meshx → FEniCS → Paraview




```
$PhysicalNames
6
1 3 "Top"
1 4 "Right"
1 5 "Left"
1 6 "Bottom"
2 1 "Domain"
2 2 "Obstacle"
$EndPhysicalNames
```

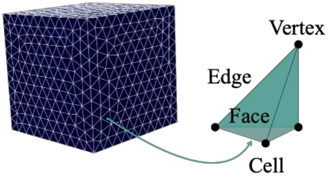
```python
# Define Dirichlet boundary conditions at top and bottom boundaries
bcs = [DirichletBC(V, 5.0, boundaries, tags['Top']),
       DirichletBC(V, 0.0, boundaries, tags['Bottom'])]

# Define new measures associated with the interior domains and
# exterior boundaries
dx = Measure("dx")[domains]
ds = Measure("ds")[boundaries]

# Define variational form
F = (inner(a0*grad(u), grad(v))*dx(tags['Domain'])
     + inner(a1*grad(u), grad(v))*dx(tags['Obstacle'])
     - g_L*v*ds(tags['Left']) - g_R*v*ds(tags['Right'])
     - f*v*dx(tags['Domain']) - f*v*dx(tags['Obstacle']))
```



Basis for Meshx

Mesh entities

FEniCS	Vertex	Edge	Face	Cell
Gmsh	Point	Line	Triangle	Tetra
Dimension	0	1	2	3
	point.xdmf	line.xdmf	triangle.xdmf	tetra.xdmf




```
{
    "Top": 3,
    "Right": 4,
    "Left": 5,
    "Bottom": 6,
    "Domain": 1,
    "Obstacle": 2
}
```

tags.json

```
domain_mvc = MeshValueCollection("size_t", mesh, dim)
with XDMFFile("mesh/triangle.xdmf") as infile:
    infile.read(domain_mvc, "tag")
domain = cpp.mesh.MeshFunctionSizet(mesh, domain_mvc)

f = open('mesh/tags.json')
tags = json.load(f)
```

```
# Define variational form
F = (inner(a0*grad(u), grad(v))*dx(tags['Domain'])
     + inner(a1*grad(u), grad(v))*dx(tags['Obstacle'])
     - g_L*v*ds(tags['Left']) - g_R*v*ds(tags['Right'])
     - f*v*dx(tags['Domain']) - f*v*dx(tags['Obstacle']))
```

Example:

```
→  root cd Codes/
→  Codes cd poisson/
→  poisson ls
main.py  plate.geo  plate.msh
→  poisson meshx plate.msh
String tag      Num Tag   Dim
------------   ---------  -----
Top                   3     1
Right                 4     1
Left                  5     1
Bottom                6     1
Domain                1     2
Obstacle              2     2
Creating line mesh
Creating triangle mesh
XDMF created! 🎉🥳
→  poisson ls
main.py  mesh  plate.geo  plate.msh  sub_domains
→  poisson python3 main.py
```

GitHub repository for meshx:
https://github.com/iitrabhi/meshx

You can use this Docker image:
https://github.com/iitrabhi/fenics-docker

Thank You….

(computationalmechanics.in)