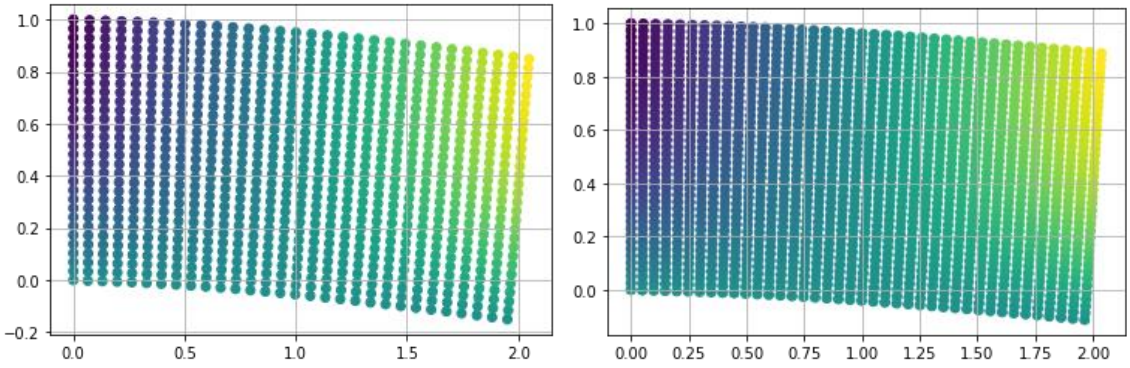
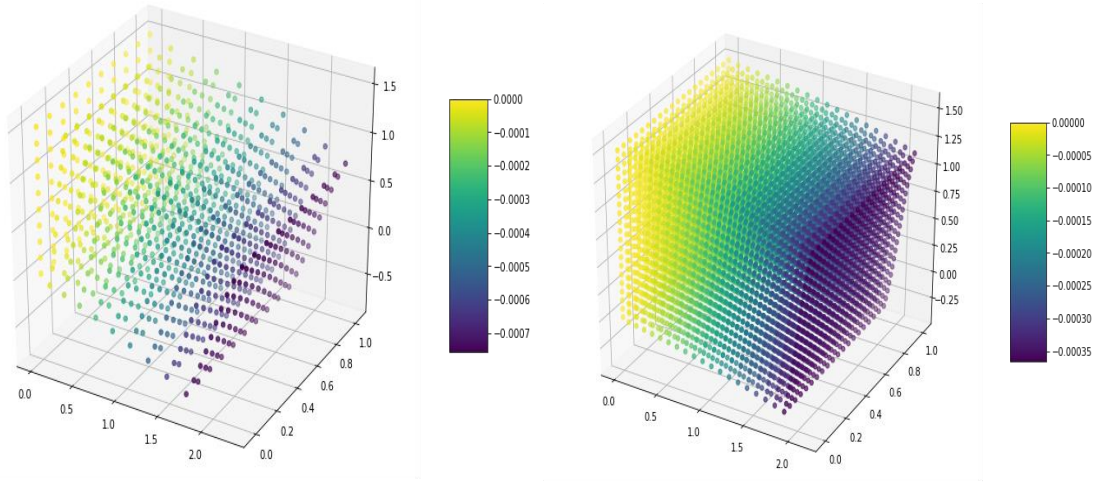
1. Mesh description



<Figure1. 2D result of 900 nodes and 1600 nodes>



<Figure2. 3D result of 1000 nodes and 8000 nodes>

Rectangle mesh is used for 2D model and hexahedron mesh is used for 3D model. Since I make an 2D and 3D FEM algorithm through Python, it is difficult for me to use different type of element such as triangular or tetrahedron mesh. In addition, it is impossible for me to use different meshing strategies such as partitioning and local refinement. Above figures are example result of FEM algorithm. Left edge/surface is fixed and right edge/surface is applied down direction loads. In 2D, 4 nodes (points) are one element. In 3D, 8 nodes (points) are one element.

2. Quantities of interest

Deformation, strain, and stress are the quantities of interest in the model. Since, goal of the FEM analysis is getting displacement and stress, they are important.

First, unknow deformation must be calculated:

$$d_f = K_{ff}^{-1}\left(f_f - K_{fs}d_s\right)$$

Then, unknown deformation vector is combined with known deformation vector. Now we have whole deformation vector named d.

Second, strain vector must be calculated:

$$\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} = [B][d]$$

Then, stress vector will be calculated:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = [D] \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix}$$

$$Where, D\ is\ elasticity\ matrix$$

For the convergence study, maximum and minimum value of displacement, and maximum and minimum value of stress value will be used.

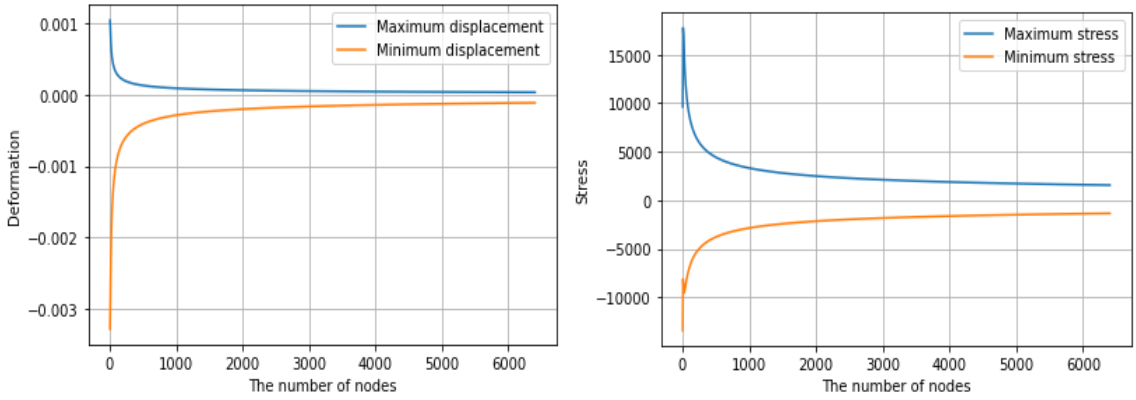
3. **Convergence study**

Since I build my own algorithm for FEM, I don't have any special method about refinement method. I just increase the number of nodes to refine mesh. In order to make mesh, I defined generate_mesh function:

(6). Mesh

```
def generate_mesh(x,y,xNode,yNode):
    xmin,xmax = 0,x
    ymin,ymax = 0,y
    xl = np.linspace(xmin,xmax,xNode)
    yl = np.linspace(ymin,ymax,yNode)
    xm, ym = np.meshgrid(xl,yl)
    return xl,yl,xm,ym
```

When I want to refine mesh, I just increase xNode, yNode variable in the function. Then, there would be more nodes according to np.linspace algorithm.



<Figure3. Convergence of deformation and stress>

As the number of nodes increase, difference between maximum and minimum displacement/stress decrease. That is, maximum value decrease and minimum value increase, as the number of node increase. Left figure of figure 3 is convergence study of displacement. And right figure of figure 3 is convergence study of stress. In displacement case, maximum displacement's convergence speed is faster than that of minimum one. And they are all converged specific value at 4000 nodes. In stress case, they are almost converged to specific value at 5000 nodes.