

基于 python 的平面四边形单元

本文处理的问题为：10*10 的正方形，厚度为 0.025，左边加约束，右边加上均匀载荷 204pa。其中弹性模型 E=200000，泊松比 u=0.3，共划分了 50*50 共计 2500 个网格，为四边形单元，如图 1 所示。

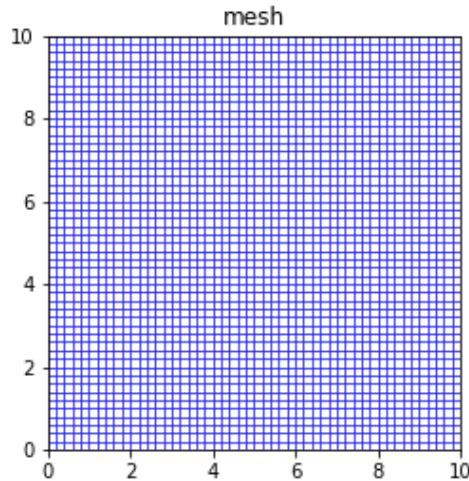
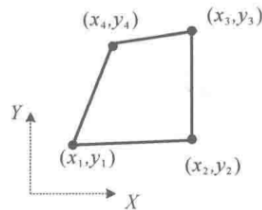


图 1. 网格划分图（2500 个四边形单元）

一. 单元基本理论（平面四边形）



四边形单元如上图所示，共有四个节点，每个节点有两个自由度(Ux 和 Uy)，一个单元有 8 个自由度，则一次四边形实体单元的单元刚度矩阵为 8*8 阶。该单元局部坐标系和整体坐标系已知，则整体坐标系中的刚度矩阵和局部坐标系中的刚度矩阵相同。假设四边形四个顶点坐标分别为 (x1,y1)，(x2,y2)，(x3,y3)，(x4,y4)，需要注意的是节点顺序为逆时针。单元材料弹性模量为 E，厚度 t，泊松比 u,则一个单元的刚度矩阵表示为：

$$K_e = t \int_{-1}^1 \int_{-1}^1 B^T D B |J| d\xi d\eta$$

其中矩阵 B 也称为应变矩阵，表示为：

$$B = \frac{1}{|J|} \begin{bmatrix} B_1 & B_2 & B_3 & B_4 \end{bmatrix}$$

Bi 表示为：

$$B_i = \begin{bmatrix} a \frac{\partial N_i}{\partial x} & b \frac{\partial N_i}{\partial y} & 0 \\ 0 & c \frac{\partial N_i}{\partial x} & d \frac{\partial N_i}{\partial y} \\ c \frac{\partial N_i}{\partial x} & d \frac{\partial N_i}{\partial y} & a \frac{\partial N_i}{\partial x} & b \frac{\partial N_i}{\partial y} \end{bmatrix}$$

上式中的 N_i 为单元形函数，表示为：

$$\begin{aligned} N_1 &= \frac{1}{4}(1-x)(1-y) \\ N_2 &= \frac{1}{4}(1-x)(1+y) \\ N_3 &= \frac{1}{4}(1+x)(1-y) \\ N_4 &= \frac{1}{4}(1+x)(1+y) \end{aligned}$$

系数 a,b,c,d 表示为：

$$\begin{aligned} a &= \frac{1}{4}[y_1(1-x) - y_2(1-x) + y_3(1-x) - y_4(1-x)] \\ b &= \frac{1}{4}[y_1(1-x) - y_2(1-x) + y_3(1-x) - y_4(1-x)] \\ c &= \frac{1}{4}[x_1(1-y) - x_2(1-y) + x_3(1-y) - x_4(1-y)] \\ d &= \frac{1}{4}[x_1(1-y) - x_2(1-y) + x_3(1-y) - x_4(1-y)] \end{aligned}$$

J 为雅可比矩阵，其行列式表示为：

$$|J| = \frac{1}{8} \begin{vmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \end{vmatrix}$$

矩阵 D 也称为本构矩阵或弹性矩阵，对于平面应力问题，其表达式为：

$$D = \frac{E}{1-\mu^2} \begin{bmatrix} 1 & \mu & 0 \\ \mu & 1 & 0 \\ 0 & 0 & \frac{1-\mu}{2} \end{bmatrix}$$

如果是独立四边形实体单元系统，则系统总体刚度矩阵为 $8n \times 8n$ 阶， n 为系统节点数量，用 K 表示，U 代表整体坐标系中系统的节点位移列阵，F 表示整体

坐标系中力矩阵，均为 $8n \times 1$ 阶，有：

$$K * U = F$$

求解处理过的方程组可得到整体坐标系中的节点位移，然后通过下式计算单元应力：

$$f_e = K_e U_e$$

二. 程序架构（四边形单元）

本文基于 python 编写的有限元四边形单元程序，其程序构架如图 2 所示（从右边看起）：

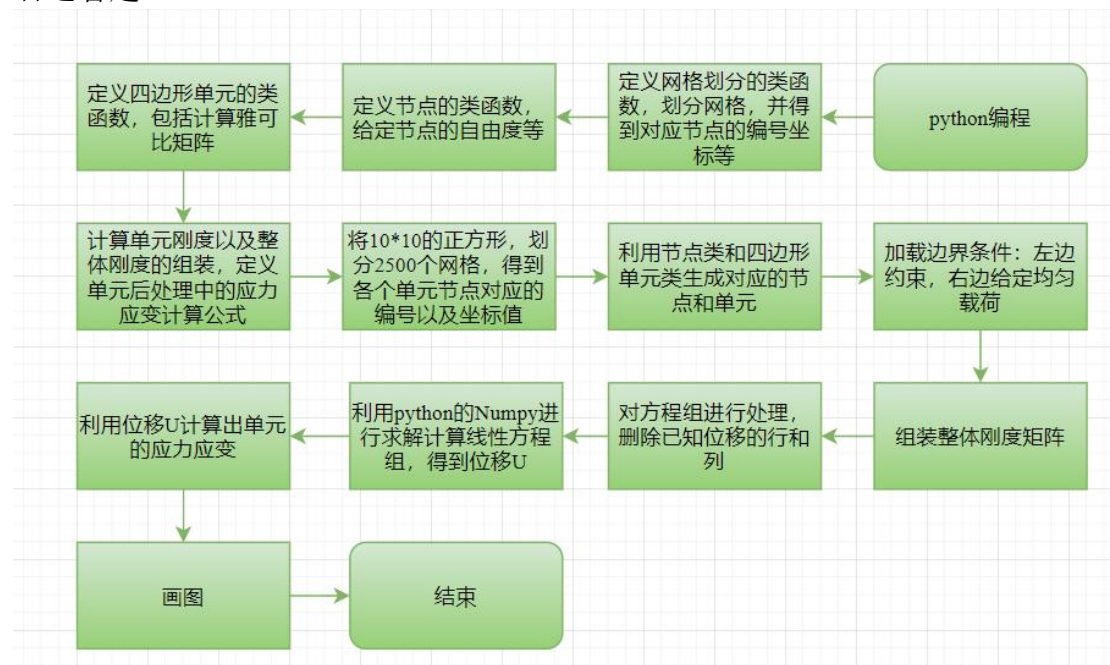


图 2 四边形单元程序架构图

三. 程序结果对比（Feon）

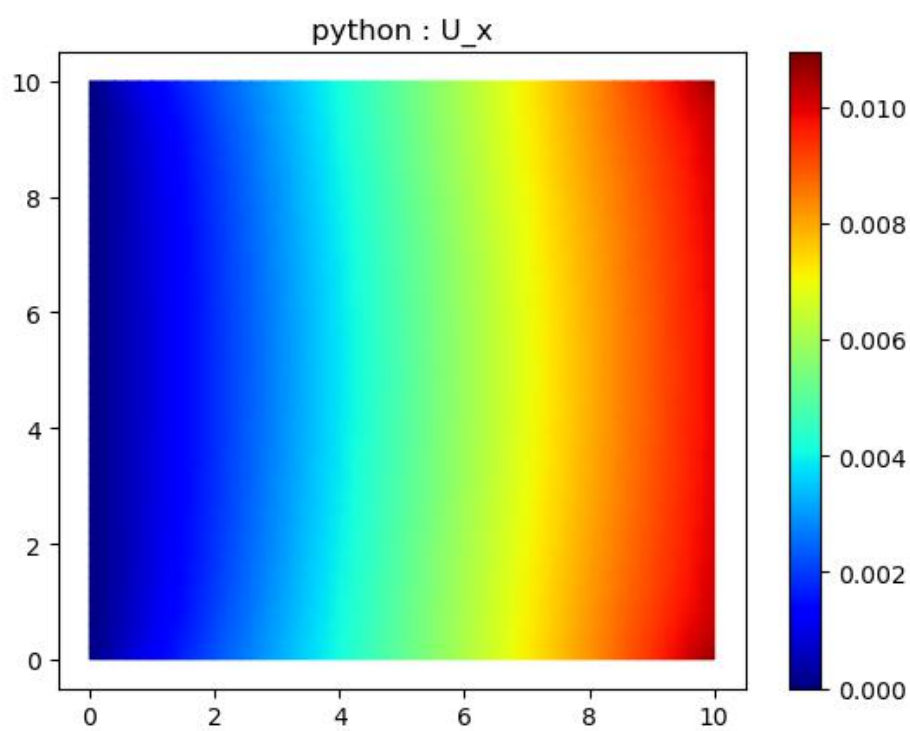
本文所使用的对比程序为 Feon。

Feon 是湖北工业大学土木建筑与环境学院教师裴尧尧基于 python 开发的一个开源免费的有限元计算框架。这是一个致力于有限元编程教学和有限元理论研究的框架。著有：《python 和有限元》

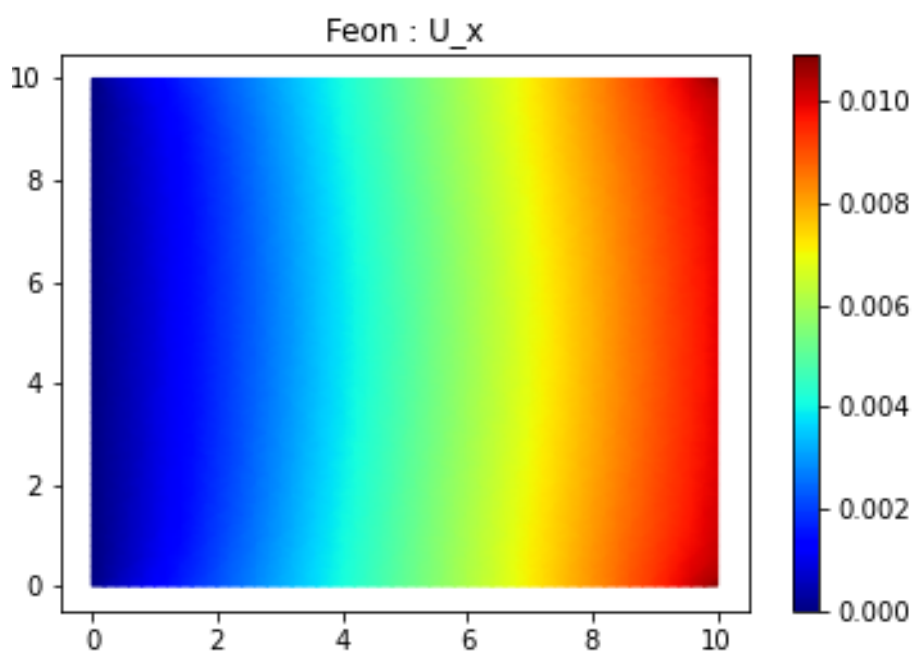
计算结果	Python 程序		Feon	
	最大值	最小值	最大值	最小值
x 轴方向上位移	0.010934359	0	0.010934359	0
y 轴方向上位移	0.002072748	-0.002072748	0.002072748	-0.002072748
x 轴方向上单元应力	316.4008122	191.0123743	316.4008122	191.0123743
y 轴方向上单元应力	58.83211337	-36.67803717	58.83211337	-36.67803717
xy 平面方向上剪切力	59.07063377	-59.07063377	59.07063377	-59.07063377
x 轴方向上单元应变	0.00157111	0.000871715	Feon 没有计算应变	
y 轴方向上单元应变	-8.69E-06	-0.000546067		
xy 平面方向上剪切应变	0.000767918	-0.000767918		

从上表中可以看出，本文基于 python 所写有限元四边形单元程序，其计算得到的位移以及应力，可以达到与 Feon 开源代码包差不多的精度。

Python 程序与 Feon 包具体的对比云图如图 3—图 10 所示。

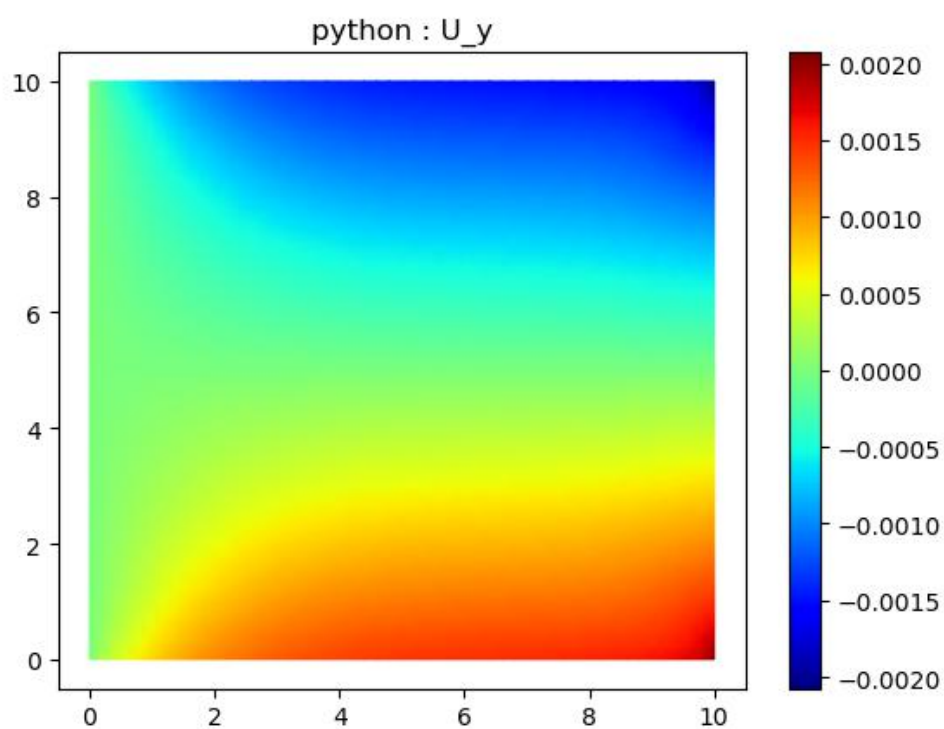


(a) python 程序 x 轴位移

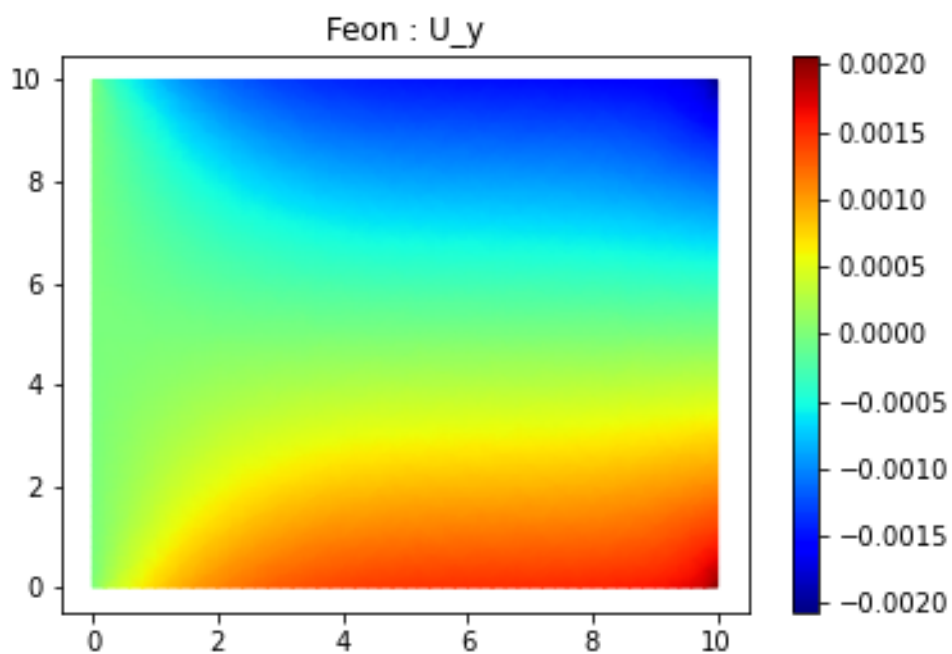


(b) Feon 架构的 x 轴位移

图 3. x 轴位移对比图

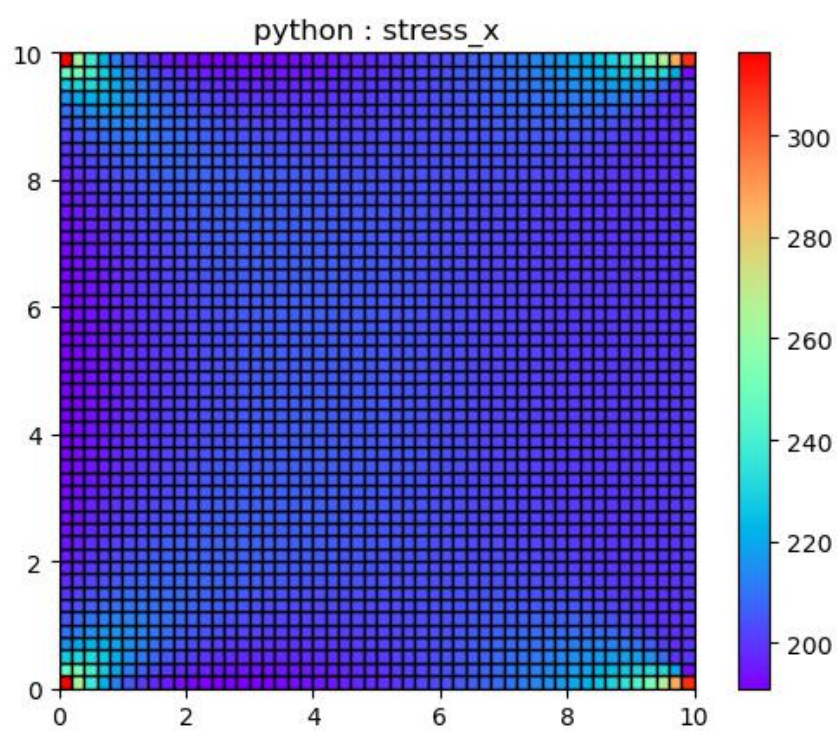


(a) python 程序 y 轴位移

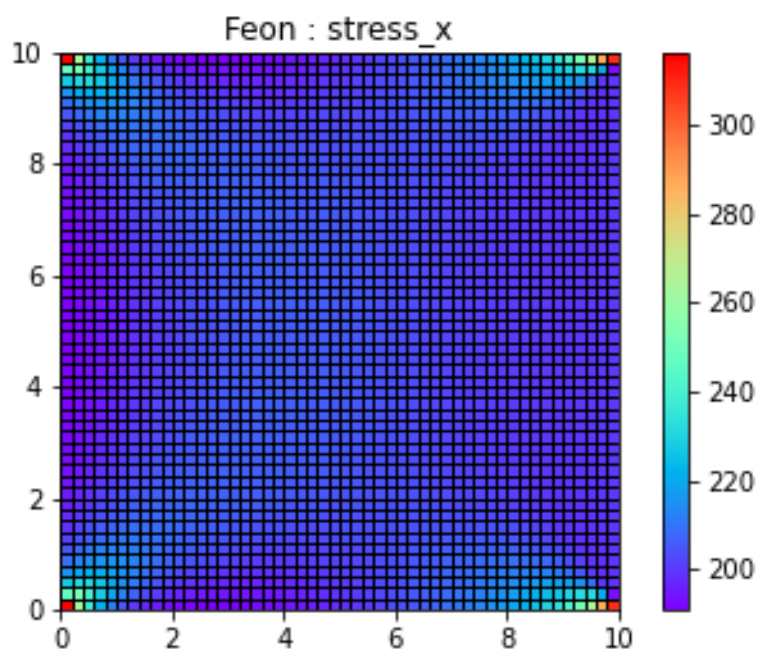


(b) Feon 架构的 y 轴位移

图 4. y 轴位移对比图

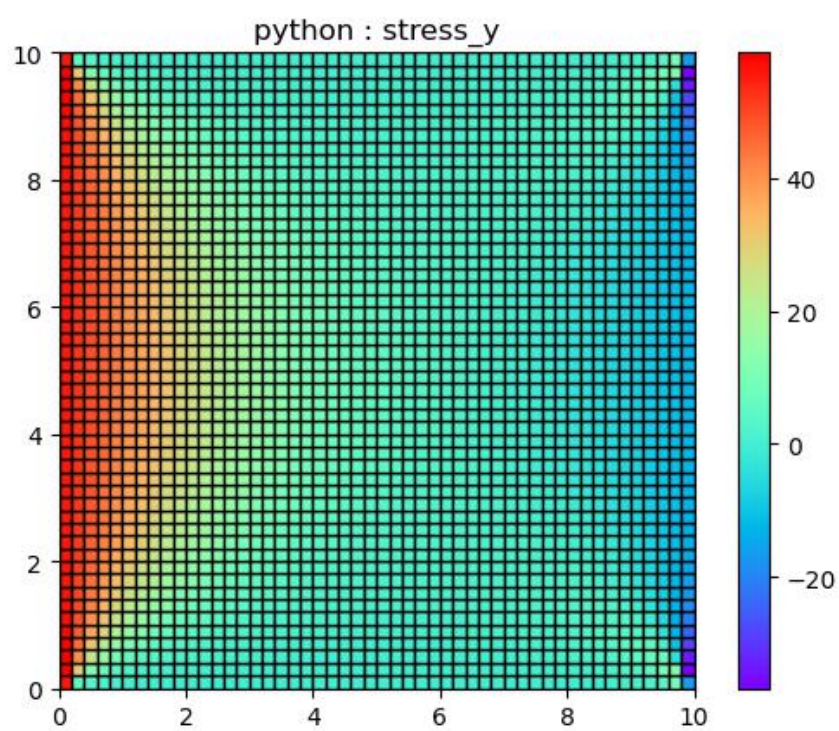


(a) python 程序 x 轴方向正应力

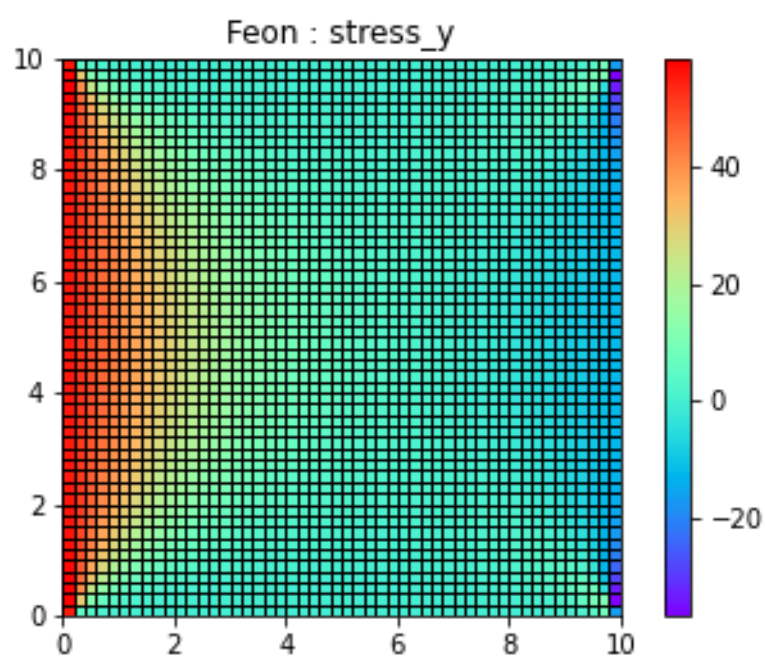


(b) Feon 架构的 x 轴方向正应力

图 5. x 轴方向正应力对比图

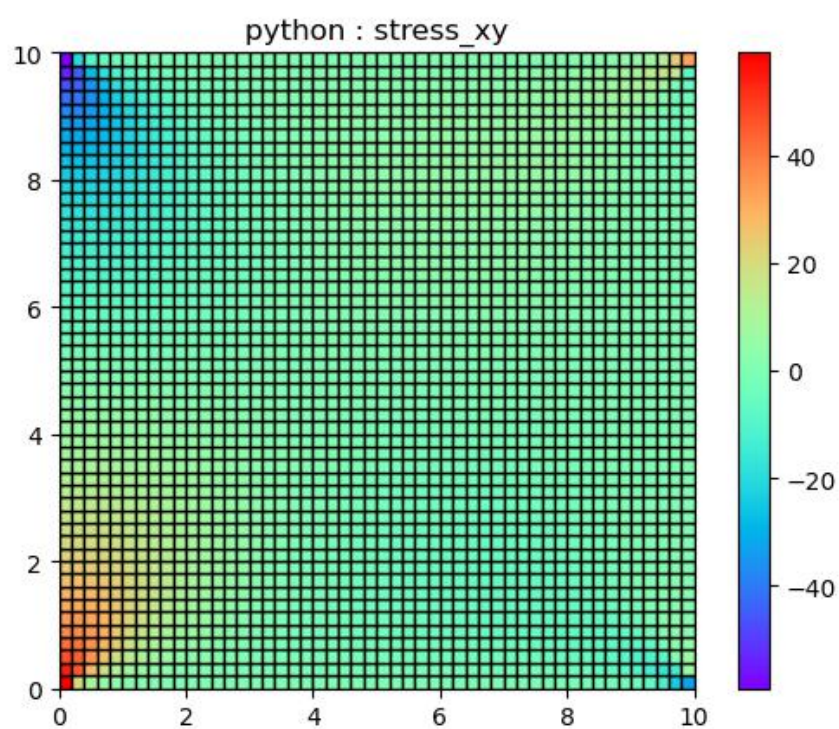


(a) python 程序 y 方向正应力

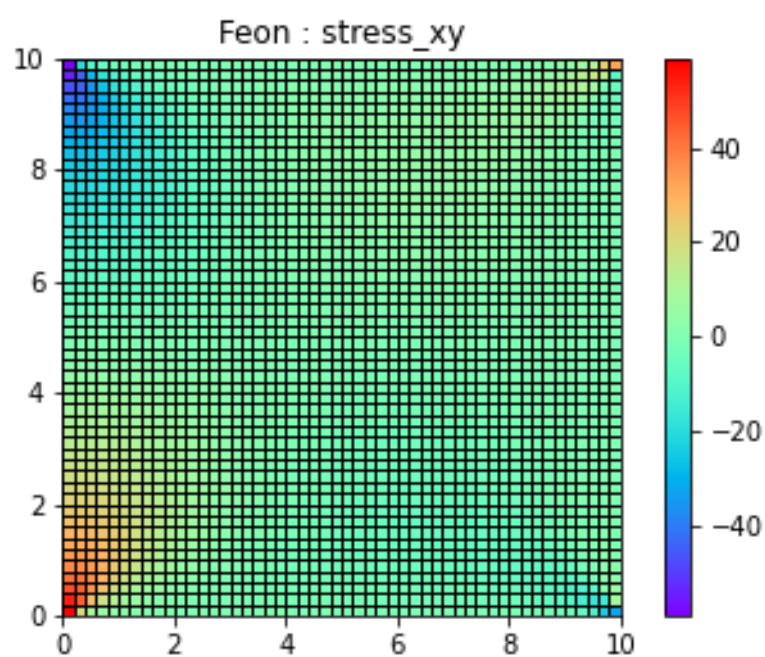


(b) Feon 架构的 y 方向正应力

图 6. y 方向正应力对比图

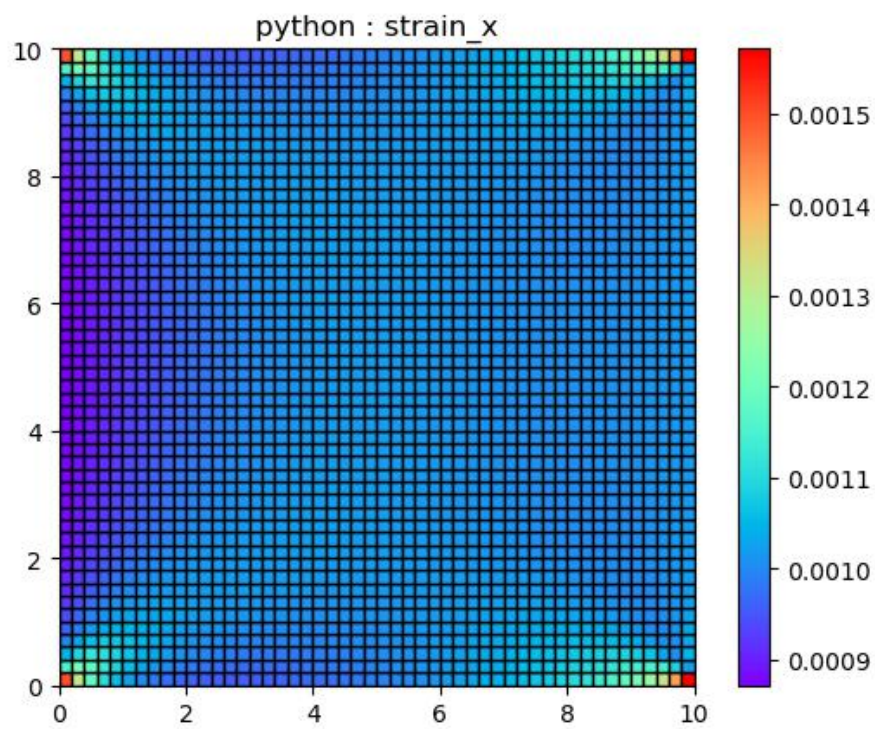


(a) python 程序 xy 方向切应力



(b) Feon 架构的 xy 方向切应力

图 7. xy 平面方向剪切应力对比图

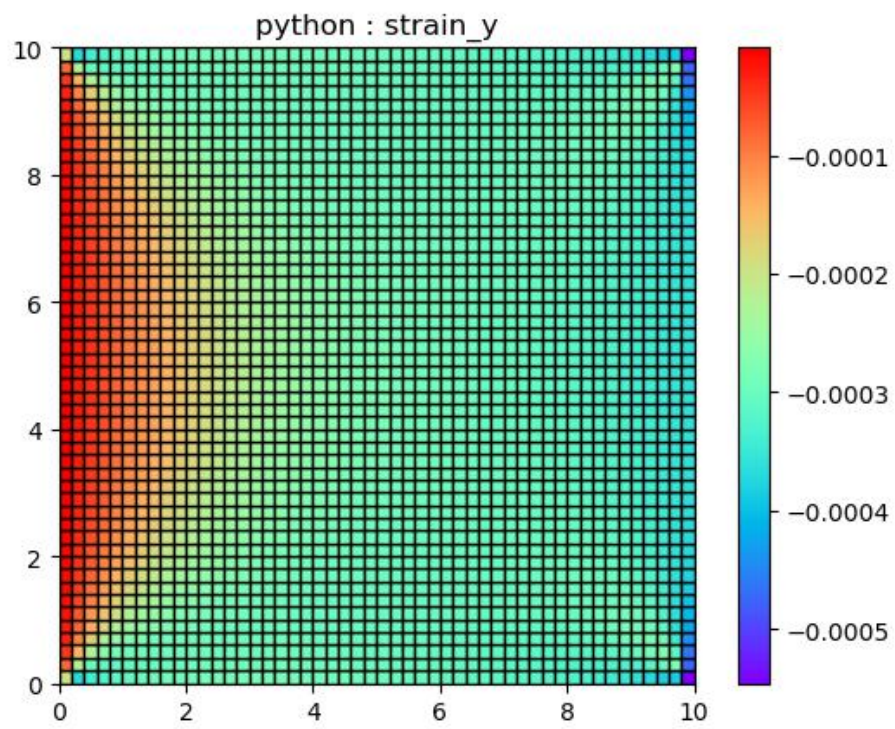


(a) python 程序 x 方向正应变

Feon 没有计算应变

(b) Feon 架构的 x 方向正应变

图 8. x 方向正应变对比图

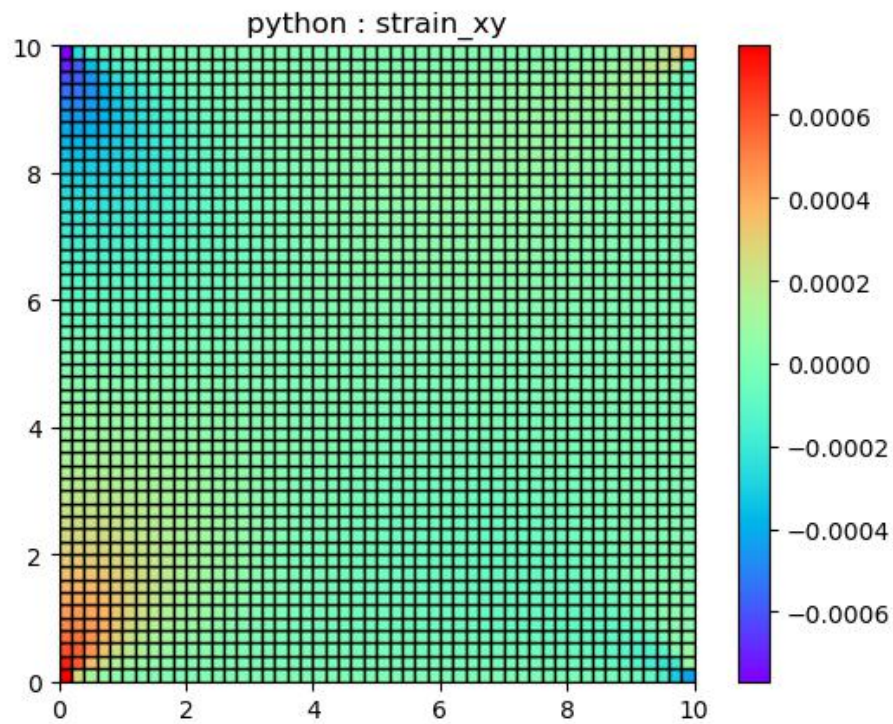


(a) python 程序 y 方向正应变

Feon 没有计算应变

(b) Feon 架构的 y 方向正应变

图 9. y 方向正应变对比图



(a) python 程序 xy 方向切应变

Feon 没有计算应变

(b) Feon xy 方向切应变

图 10. xy 方向切应变对比图

如上图所示，本文所设计的基于 python 的有限元四边形单元程序，可以较好的处理计算下 x 轴，y 轴位移，正应力、剪切应力，正应变、剪切应变等问题。