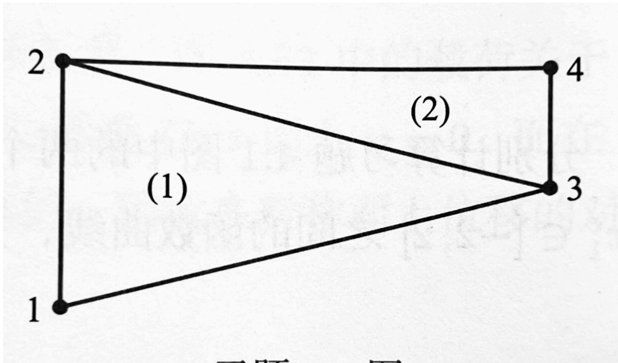
1 习题 4.4

采用习题 4.4 图所示的两个由 2 个三角形单元组成的网格，重新手工求解例题 4-1 中的问题。



习题 4.4 图

1.1 例题 4-1

考虑图 4.25a 所示的线弹性平面应力问题，弹性模量 $E = 3 \times 10^7 pa$, 泊松比 $\nu = 0.3$。求解域为梯形，尺寸如图所示，厚度为 1. 梯形左端固定，右端和下部边界自由（即 $\bar{\boldsymbol{t}} = 0$），在上部边界受均布力 $\bar{t_y} = -20Nm^{-1}$ 作用。试采用一个四边形单元求解位移场和应力场。

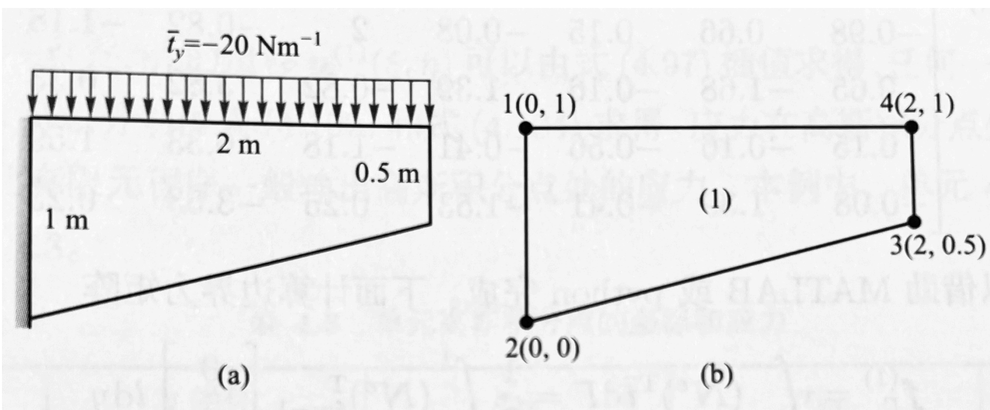


图 4.25 梯形线弹性平面应力问题

1.2 前处理

网格划分习题 4.4 图所示。单元的弹性矩阵为：

$$\boldsymbol{D}^e = \frac{E}{1-\nu^2}\begin{bmatrix}1 & \nu & 0\\ \nu & 1 & 0\\ 0 & 0 & \frac{1-\nu}{2}\end{bmatrix} = 3.3\times 10^7\begin{bmatrix}1 & 0.3 & 0\\ 0.3 & 1 & 0\\ 0 & 0 & 0.35\end{bmatrix}$$

单元的坐标矩阵为:

$$\begin{bmatrix}\boldsymbol{x}^{(1)} & \boldsymbol{y}^{(1)}\end{bmatrix} = \begin{bmatrix}0 & 0\\ 2 & 0.5\\ 0 & 1\end{bmatrix}, \quad \begin{bmatrix}\boldsymbol{x}^{(2)} & \boldsymbol{y}^{(2)}\end{bmatrix} = \begin{bmatrix}0 & 1\\ 2 & 0.5\\ 2 & 1\end{bmatrix}$$

三角形单元是等参单元，利用母单元内形函数进行坐标变换。三角形单元的形函数为：

$$\boldsymbol{N}_{\boldsymbol{I}}^{\boldsymbol{e}} = \frac{1}{2A^e}(a_I + b_I x + c_I y)$$

$$a_1 = x_2^e y_3^e - x_3^e y_2^e$$

$$b_1 = y_2^e - y_3^e$$

$$c_1 = x_3^e - x_2^e$$

$$2A^e = a_1 + a_2 + a_3$$

轮换指标，将 1 换为 2，2 换为 3，3 换为 1，可以得到 a_2, b_2, c_2。同理，进一步对换指标可以得到 a_3, b_3, c_3。因此从母空间（面积坐标 ξ_I）到物理空间有如下的坐标变换：

$$\xi_I = \frac{1}{2A^e}(a_I + b_I x + c_I y)$$

$$x = \Sigma_{I=1}^3 \xi_I x_I^e$$

$$y = \Sigma_{I=1}^3 \xi_I y_I^e$$

对于母空间内的 T3 单元, 带入各个节点 (对于单元 1, 1->1, 2->3, 3->1; 对于单元 2, 2->1, 3->2, 4->3 以保证逆时针顺序) 的坐标 (ξ, η)：(0,0), (1,0), (0,1)：

$$N_1^e = \eta$$

$$N_2^e = 1 - \xi - \eta$$

$$N_3^e = \xi$$

最终得到母单元坐标和物理空间坐标之间的 Jacobi 矩阵：

$$\boldsymbol{J^e}=\begin{bmatrix}\frac{\partial x}{\partial\xi} & \frac{\partial y}{\partial\xi}\\ \frac{\partial x}{\partial\eta} & \frac{\partial y}{\partial\eta}\end{bmatrix}=\begin{bmatrix}\sum_{I=1}^{3}\frac{\partial N_I^{T3}}{\partial\xi}x_I^e & \sum_{I=1}^{3}\frac{\partial N_I^{T3}}{\partial\xi}y_I^e\\ \sum_{I=1}^{3}\frac{\partial N_I^{T3}}{\partial\eta}x_I^e & \sum_{I=1}^{3}\frac{\partial N_I^{T3}}{\partial\eta}y_I^e\end{bmatrix}=\begin{bmatrix}x_3^e-x_2^e & y_3^e-y_2^e\\ x_1^e-x_2^e & y_1^e-y_2^e\end{bmatrix}$$

带入单元的坐标信息和形函数并微分，两个单元的 Jacobi 矩阵分别为：

$$\boldsymbol{J^1}=\begin{bmatrix}-2 & 0.5\\ -2 & -0.5\end{bmatrix},\quad \boldsymbol{J^2}=\begin{bmatrix}0 & 0.5\\ -2 & 0.5\end{bmatrix}$$

但由于三节点三角形单元的形函数 $\boldsymbol{N_I^e}$ 的形式，我们也可以不使用 Jacobi 矩阵和链式法则，先在母空间中求导再得到物理空间内形函数的梯度。我们可以直接对单元形函数矩阵进行操作。

1.3 单元分析

每个单元的试探函数，

$$u^e(x,y)=\boldsymbol{N^e}\boldsymbol{d^e}=\begin{bmatrix}N_1^e & N_2^e & N_3^e\end{bmatrix}\boldsymbol{d^e}$$

$\boldsymbol{N^e}$ 为单元形函数矩阵，$\boldsymbol{N_I^e}$ 为节点 I 的形函数矩阵，

$$\boldsymbol{N_I^e}=\begin{bmatrix}N_I^{T3} & 0\\ 0 & N_I^{T3}\end{bmatrix}$$

单元形函数矩阵的梯度为单元应变矩阵，

$$\boldsymbol{B^e}=\boldsymbol{\nabla_S}\boldsymbol{N^e}=\frac{1}{2A^e}\begin{bmatrix}b_1 & 0 & b_2 & 0 & b_3 & 0\\ 0 & c_1 & 0 & c_2 & 0 & c_3\\ c_1 & b_1 & c_2 & b_2 & c_3 & b_3\end{bmatrix}$$

其中,

$$a_1^1=2,b_1^1=-0.5,c_1^1=-2$$

$$a_2^1=0,b_2^1=1,c_2^1=0$$

$$a_3^1=0,b_3^1=-0.5,c_3^1=2$$

$$a_1^2=1,b_1^2=-0.5,c_1^2=0$$

$$a_2^2=2,b_2^2=0,c_2^2=-2$$

$$a_3^2=-2,b_3^2=0.5,c_3^2=2$$

因此，

$$\boldsymbol{B^1} = \frac{1}{2}\begin{bmatrix} -0.5 & 0 & 1 & 0 & -0.5 & 0 \\ 0 & -2 & 0 & 0 & 0 & 2 \\ -2 & -0.5 & 0 & 1 & 2 & -0.5 \end{bmatrix}$$

$$\boldsymbol{B^2} = \frac{1}{1}\begin{bmatrix} -0.5 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & -2 & 0 & 2 \\ 0 & -0.5 & -2 & 0 & 2 & 0.5 \end{bmatrix}$$

利用弱形式得到了单元刚度矩阵，

$$K^e = \int_{\Omega^e} (\boldsymbol{B^e})^{\boldsymbol{T}} \boldsymbol{D^e} \boldsymbol{B^e} d\Omega = A^e t^e (\boldsymbol{B^e})^{\boldsymbol{T}} \boldsymbol{D^e} \boldsymbol{B^e}$$

A^e 为物理空间中的单元的面积，t^e 为单元的厚度，在该题目中为 1，$\boldsymbol{D^e}$ 为单元的弹性矩阵。带入数据得到两个单元的单元刚度矩阵，

$$\boldsymbol{K^1} = 10^6\begin{bmatrix} 13.6125 & 5.3625 & -4.125 & -5.775 & -9.4875 & 0.4125 \\ 5.3625 & 33.721875 & -4.95 & -1.44375 & -0.4125 & -32.278125 \\ -4.125 & -4.95 & 8.25 & 0 & -4.125 & 4.95 \\ -5.775 & -1.44375 & 0 & 2.8875 & 5.775 & -1.44375 \\ -9.4875 & -0.4125 & -4.125 & 5.775 & 13.6125 & -5.3625 \\ 0.4125 & -32.278125 & 4.95 & -1.44375 & -5.3625 & 33.721875 \end{bmatrix}$$

$$\boldsymbol{K^2} = 10^6\begin{bmatrix} 2.0625 & 0 & 0 & 2.475 & -2.0625 & -2.475 \\ 0 & 0.721875 & 2.8875 & 0 & -2.8875 & -0.721875 \\ 0 & 2.8875 & 11.55 & 0 & -11.55 & -2.8875 \\ 2.475 & 0 & 0 & 33 & -2.475 & -33 \\ -2.0625 & -2.8875 & -11.55 & -2.475 & 13.6125 & 5.3625 \\ -2.475 & -0.721875 & -2.8875 & -33 & 5.3625 & 33.721875 \end{bmatrix}$$

在单元 2 的 1-3（局部节点号）边上有均布的本质边界条件，单元 1 的 1-3（局部节点号）边有固支的约束力。对于单元 2 上的均匀分布的面力，有 $t_{x1} = t_{x_2} = t_x = 0, t_{y_1} = t_{y_2} = t_y = -20Nm^{-1}$，对应的单元边界力列阵 $\boldsymbol{f_\Gamma^1}$ 为

$$\boldsymbol{f_\Gamma^1} = \frac{lt^e}{2}\begin{bmatrix} t_x & t_y & 0 & 0 & t_x & t_y \end{bmatrix}^T = \begin{bmatrix} 0 & -20 & 0 & 0 & 0 & -20 \end{bmatrix}^T$$

1.4 单元组装

局部节点编号和总体节点编号之间有对号矩阵 $\boldsymbol{LM}$，其每一列对应一个单元，各行对应该单元的各个自由度在全局中的编号，

$$\boldsymbol{LM} = \begin{bmatrix} 1 & 3 \\ 2 & 4 \\ 5 & 5 \\ 6 & 6 \\ 3 & 7 \\ 4 & 8 \end{bmatrix}$$

在使用 Mathemetica 计算时，我们希望使用矩阵乘法以简化操作，因此有两个单元的提取矩阵，

$$\boldsymbol{L^1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \boldsymbol{L^2} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

使用提取矩阵进行矩阵散步求和，得到总体刚度矩阵

$$\boldsymbol{K} = (\boldsymbol{L^1})^{\boldsymbol{T}} \boldsymbol{K^1} \boldsymbol{L^1} + (\boldsymbol{L^2})^{\boldsymbol{T}} \boldsymbol{K^2} \boldsymbol{L^2}$$

$$\boldsymbol{K} = 10^6 \begin{bmatrix} 13.6125 & 5.3625 & -9.4875 & 0.4125 & -4.125 & -5.775 & 0 & 0 \\ 5.3625 & 33.721875 & -0.4125 & -32.278125 & -4.95 & -1.44375 & 0 & 0 \\ -9.4875 & -0.4125 & 17.7375 & -5.3625 & -4.125 & 10.725 & -4.125 & -4.95 \\ 0.4125 & -32.278125 & -5.3625 & 35.165625 & 10.725 & -1.44375 & -5.775 & -1.44375 \\ -4.125 & -4.95 & -4.125 & 10.725 & 31.35 & 0 & -23.1 & -5.775 \\ -5.775 & -1.44375 & 10.725 & -1.44375 & 0 & 68.8875 & -4.95 & -66 \\ 0 & 0 & -4.125 & -5.775 & -23.1 & -4.95 & 27.225 & 10.725 \\ 0 & 0 & -4.95 & -1.44375 & -5.775 & -66 & 10.725 & 67.44375 \end{bmatrix}$$

1.5 方程求解

使用缩减法施加单元 1 的 1-3 边（局部编号）上的固支边界条件，求解刚度方程

$$\boldsymbol{K}\boldsymbol{d} = \boldsymbol{f} + \boldsymbol{r}$$

$$\boldsymbol{d} = \begin{bmatrix} 0 & 0 & 0 & 0 & u_{x3} & u_{y3} & u_{x4} & u_{y4} \end{bmatrix}^T$$

$$\boldsymbol{f} = (\boldsymbol{L^2})^{\boldsymbol{T}} \boldsymbol{f_\Gamma^1} = \begin{bmatrix} 0 & 0 & 0 & -20 & 0 & 0 & 0 & -20 \end{bmatrix}^T$$

$$\boldsymbol{r} = \begin{bmatrix} r_{x1} & r_{y1} r_{x2} & r_{y2} & 0 & 0 & 0 & 0 \end{bmatrix}^T$$

缩减后的刚度方程为

$$\boldsymbol{K_F d_F} = \boldsymbol{f_F} - \boldsymbol{K_{FE} d_E}$$

$$\boldsymbol{K_F} = 10^6 \begin{bmatrix} 31.35 & 0 & -23.1 & -5.775 \\ 0 & 68.8875 & -4.95 & -66 \\ -23.1 & -4.95 & 27.225 & 10.725 \\ -5.775 & -66 & 10.725 & 67.44375 \end{bmatrix}$$

$$\boldsymbol{K_{FE}} = 10^6 \begin{bmatrix} -4.125 & -4.95 & -4.125 & 10.725 \\ -5.775 & -1.44375 & 10.725 & -1.44375 \\ 0 & 0 & -4.125 & -5.775 \\ 0 & 0 & -4.95 & -1.44375 \end{bmatrix}$$

$$\boldsymbol{d_F} = \begin{bmatrix} u_{x3} & u_{y3} & u_{x4} & u_{y4} \end{bmatrix}$$

$$\boldsymbol{d_E} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{f_F} = \begin{bmatrix} 0 & 0 & 0 & -20 \end{bmatrix}$$

得到位移:

$$\boldsymbol{d_F} = 10^{-6} \begin{bmatrix} -0.386714096 & -6.65018257 & 1.23358547 & -7.03364697 \end{bmatrix}^T$$

计算固支处的约束力:

$$\boldsymbol{r_E} = \boldsymbol{K_E d_E} + \boldsymbol{K_{EF} d_F} - \boldsymbol{f_E}$$

$$\boldsymbol{K_E} = 10^6 \begin{bmatrix} 13.6125 & 5.3625 & -9.4875 & 0.4125 \\ 5.3625 & 33.721875 & -0.4125 & -32.278125 \\ -9.4875 & -0.4125 & 17.7375 & -5.3625 \\ 0.4125 & -32.278125 & -5.3625 & 35.165625 \end{bmatrix}$$

$$\boldsymbol{d_E} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{K_{EF}} = 10^6 \begin{bmatrix} -4.125 & -5.775 & 0 & 0 \\ -4.95 & -1.44375 & 0 & 0 \\ -4.125 & 10.725 & -4.125 & -4.95 \\ 10.725 & -1.44375 & -5.775 & -1.44375 \end{bmatrix}$$

$$\boldsymbol{f_E} = \begin{bmatrix} 0 & 0 & 0 & -20 \end{bmatrix}$$

计算得到约束力为:

$$\boldsymbol{r_E} = \begin{bmatrix} 40 & 11.51543586 & -40 & 28.48456414 \end{bmatrix}^T$$

1.6 后处理

计算两个单元的应力场,

$$\boldsymbol{\sigma^e} = \boldsymbol{D^e B^e d^e}$$

$$\boldsymbol{\sigma^1} = \begin{bmatrix} -62.64641615 \\ -218.49890742 \\ 14.73585436 \end{bmatrix}, \boldsymbol{\sigma^2} = \begin{bmatrix} 12.76156514 \\ -19.20240232 \\ -3.19039128 \end{bmatrix}$$

2 习题 4.9

编写三角形常应变单元程序,并利用习题 4.4 的计算结果或者用程序 elasticity2d-python 使用密网格(至少 64 个单元)计算例题 4-1 所给问题的结果来验证程序的正确性。三角形单元的刚度矩阵为常数,程序中无需使用高斯积分。

2.1 三角形常应变单元程序编写

利用 elasticity2d-python 中提供的四节点四边形单元 Elast2DElem.py 进行修改,得到了三角形常应变单元 Tri2DElem.py 程序。该程序使用广义坐标法的结论,直接计算三角形常应变单元的形函数,单元应变矩阵,以及单元刚度矩阵。其源代码如下:

```
    #!/usr/bin/env python3

import FEData as model
from utitls import gauss
import numpy as np


```

```python
def Tri2DElem(e):
  """
  Calculate element stiffness matrix and element nodal body force vector

  Args:
    e : (int) element number

  Returns: ke, fe
    ke : (numpy(nen,nen)) element stiffness matrix
    fe : (numpy(nen,1)) element nodal force vector
  """
  ke = np.zeros((model.nen*model.ndof, model.nen*model.ndof))
  fe = np.zeros((model.nen*model.ndof, 1))

  # get coordinates of element nodes
  je = model.IEN[:, e] - 1
  C = np.array([model.x[je], model.y[je]]).T
  a, b, c, Ae=ParamCalc(C)

  # derivative of shape function
  B = BmatTri2D(C)
  ke = Ae * B.T @ model.D @ B

  # compute element nodal force vector
  fe = Ae * model.b[:, e].reshape((-1, 1))
  fe = fe / 3

  return ke, fe


def NmatTri2D(x, y, C):
  """
  Calculate element shape function matrix N at coordinate xt

  Args:
    Ae: Area of the element in physical coordinates
  Returns:
    Element shape function matrix N
  """
  a, b, c, Ae = ParamCalc(C)

```

```
    N1 = (a[0] + b[0] * x + c[0] * y) / (2 * Ae)
    N2 = (a[1] + b[1] * x + c[1] * y) / (2 * Ae)
    N3 = (a[2] + b[2] * x + c[2] * y) / (2 * Ae)

    return np.array([[N1, 0, N2, 0, N3, 0],
                     [0, N1, 0, N2, 0, N3]])

def BmatTri2D(C):
    """
    Calcualte derivative of element shape function matrix B at coordinate xt

    Args:
      C   : The physical coordinates

    Returns:
      Derivative of element shape function matrix B and Jacobian determination
    """
    a, b, c, Ae = ParamCalc(C)

    B = np.array([[b[0], 0, b[1], 0, b[2], 0],
                  [0, c[0], 0, c[1], 0, c[2]],
                  [c[0], b[0], c[1], b[1], c[2], b[2]]]) / (2*Ae)

    return B

def ParamCalc(C):
    '''
    Calculate a b and c in Shape function N and B

    Args:
      C: coordinate vetor of the element in physical coordnates
    '''
    a = np.zeros(3)
    b = np.zeros(3)
    c = np.zeros(3)

    a[0]=C[1][0]*C[2][1]-C[2][0]*C[1][1]
    a[1]=C[2][0]*C[0][1]-C[0][0]*C[2][1]
    a[2]=C[0][0]*C[1][1]-C[1][0]*C[0][1]

    b[0]=C[1][1]-C[2][1]
```

```
b[1]=C[2][1]-C[0][1]
b[2]=C[0][1]-C[1][1]

c[0]=C[2][0]-C[1][0]
c[1]=C[0][0]-C[2][0]
c[2]=C[1][0]-C[0][0]

Ae=0.5 * np.sum(a)

return a, b, c, Ae
```

2.2 使用三角形单元计算例 4-1

按照 elasticity2d-python 程序的规则编写使用三角形单元计算例题 4-1 问题的.json 文件如下：

```
  {
"Title": "Exercise 4-9 (2 Triangle elements)",

"nsd": 2,
"ndof": 2,
"nnp": 4,
"nel": 2,
"nen": 3,

"E": 30e6,
"nu": 0.3,

"ngp": 2,
"nd": 4,

"flags": [2, 2, 2, 2, 0, 0, 0, 0],

"nbe": 1,
"n_bc": [
  [2],
  [4],
      [0],
      [-20],
      [0],
      [-20]
```

```
    ],

  "x": [0.0, 0.0, 2.0, 2.0],
  "y": [0.0, 1.0, 0.5, 1.0],

  "IEN": [
    [1, 2],
    [3, 3],
    [2, 4]
  ],

  "plane_strain": 0,
  "plot_mesh": "no",
  "plot_nod" : "no",
  "plot_disp": "no",
  "compute_stress": "no",
  "plot_stress_xx": "no",
  "plot_mises": "no",
  "plot_tex": "no",
  "fact": "9.221e3",
  "print_disp":"no"
}
```

使用三角形单元进行计算需要修改主程序中 FERun() 函数中计算单元刚度矩阵的函数。修改后计算各节点的应变以及约束力如下：

```
       Mesh Params
No. of Elements  2
No. of Nodes     4
No. of Equations 8

Condition number of stiffness matrix:  92.97542813753554

solution d
[ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
 -3.87100810e-07 -6.65683275e-06  1.23481905e-06 -7.04068062e-06]

reaction f =
 [[ 40.        ]
 [ 11.51543586]
 [-40.        ]
 [ 28.48456414]]
```

2.3 计算结果对比

将三角形单元计算结果 $\boldsymbol{d_F^h}$ 与习题 4-4 中手工求解的结果进行对比，将手工求解的位移 $\boldsymbol{d_F}$ 视作精确值，计算有限元解对精确值的相对误差，

$$\boldsymbol{Error} = \frac{|\boldsymbol{d_F^h} - \boldsymbol{d_F}|}{\boldsymbol{d_F}} = \begin{bmatrix} 0.001 & 0.001 & 0.001 & 0.001 \end{bmatrix}^T$$

相对误差均在 0.1% 左右，可以认为程序正确。