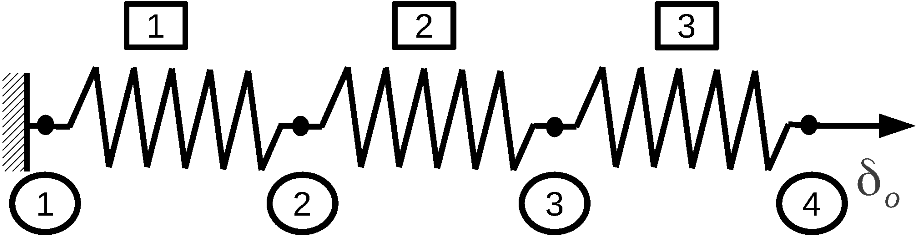
Resumen de la Guía 2 - Introducción a los elementos Finitos

24 de abril de 2020

INSTITUTO DE TECNOLOGÍA SABATO
Comisión Nacional de Energía Atómica
UNIVERSIDAD NACIONAL DE SAN MARTÍN

Problema 1

Problema 1 : Resortes



Resolución (Python)

```
r = [1, 2]
s = [0, 3]
U = np.zeros((4,1))
F = np.zeros((4,1))
U[s] = [0, 0.002]
F[R] = [0, 0]
U[r] = numpy.linalg.solve(K[np.ix_(r,r)], F[r] - K[ np.ix_[r, s].dot(U[s]))
```

Resolución (Matlab)

```
r = [1, 2]
s = [0, 3]
U = zeros(4,1)
F = zeros(4,1)
U(s) = [0, 0.002]
F(R) = [0, 0]
U(r) = K(r,r)\(F(r) - K(r, s)*U(s))
```

Resultado

Desplazamientos (por nodo)

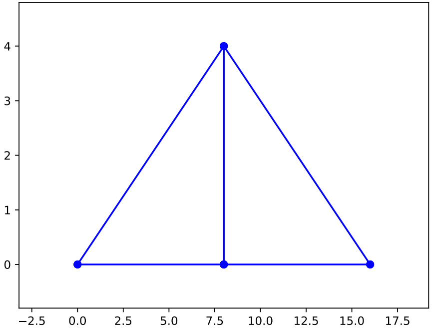
```
0.000000e+00
6.666667e-04
1.333333e-03
2.000000e-03
```

Fuerzas (por nodo)

```
-1.333333e-01
0.000000e+00
0.000000e+00
1.333333e-01
```

Problema 2

Problema 2



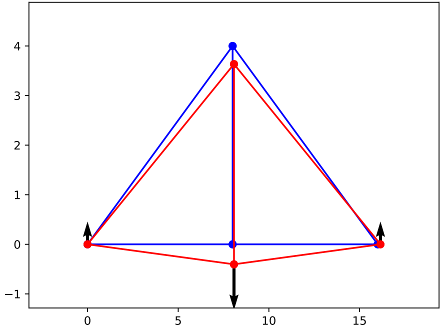

```
GL
2
NODES
4
 0.  0. 0
 8.  4. 0
 16. 0. 0
 8.  0.  0
ELEMENTS
5 2
0  1 10e-4  210e9
3  0 10e-4  210e9
3  1 10e-4  210e9
1  2 10e-4  210e9
2  3 10e-4  210e9
VINS
3
0 1 1 0 0
2 -1 1 0 0
3 -1 -1 0 -20e3
```

Problema 2: Matrices Elementales

```
Elemento 0, scle = 1.878297e+07
[[ 1.    0.5  -1.   -0.5 ]
 [ 0.5   0.25 -0.5  -0.25]
 [-1.   -0.5   1.    0.5 ]
 [-0.5  -0.25  0.5   0.25]]
Elemento 1, scle = 2.625000e+07
[[ 1.  0. -1.  0.]
 [ 0.  0.  0.  0.]
 [-1.  0.  1.  0.]
 [ 0.  0.  0.  0.]]
Elemento 2, scle = 5.250000e+07
[[ 0.  0.  0.  0.]
 [ 0.  1.  0. -1.]
 [ 0.  0.  0.  0.]
 [ 0. -1.  0.  1.]]
Elemento 3, scle = 1.878297e+07
[[ 1.   -0.5  -1.    0.5 ]
 [-0.5   0.25  0.5  -0.25]
 [-1.    0.5   1.   -0.5 ]
 [ 0.5  -0.25 -0.5   0.25]]
```

```
Elemento 4, scle = 2.625000e+07
[[ 1.  0. -1.  0.]
 [ 0.  0.  0.  0.]
 [-1.  0.  1.  0.]
 [ 0.  0.  0.  0.]]

Matriz Global , scale factor = 6.189149e+07
[[ 0.73  0.15 -0.3  -0.15  0.    0.   -0.42  0.  ]
 [ 0.15  0.08 -0.15 -0.08  0.    0.    0.    0.  ]
 [-0.3  -0.15  0.61  0.   -0.3   0.15  0.    0.  ]
 [-0.15 -0.08  0.    1.    0.15 -0.08  0.   -0.85]
 [ 0.    0.   -0.3   0.15  0.73 -0.15 -0.42  0.  ]
 [ 0.    0.    0.15 -0.08 -0.15  0.08  0.    0.  ]
 [-0.42  0.    0.    0.   -0.42  0.    0.85  0.  ]
 [ 0.    0.    0.   -0.85  0.    0.    0.    0.85]]

 su determinante: 0.000000e+00
```

Problema 2: Resortes



Desplazamientos

```
(0,0)  x 0.000000e+00
       y 0.000000e+00
(8,4)  x 7.619048e-04
       y -3.653398e-03
(16,0) x 1.1523810e-03
       y 0.000000e+00
(8,0   x 7.619048e-04
       y -4.034350e-03
```

Fuerzas

```
1.455192e-11
1.000000e+04
0.000000e+00
0.000000e+00
0.000000e+00
1.000000e+04
0.000000e+00
-2.000000e+04
```

Problema 3

Problema 3: Fuerzas Distribuidas

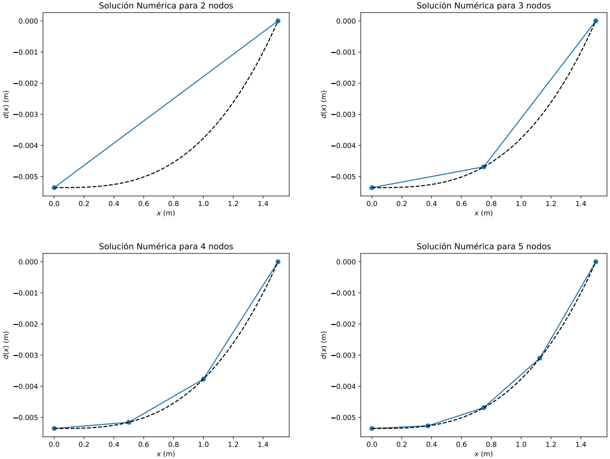
$$F_e = \int_{elemento} \begin{bmatrix} N_1 \\ N2 \end{bmatrix} f(x) dx$$

Calculo de fuerzas Directo

```
def NI(x, l):
    return 1. - x/l

def ND(x, l):
    return x/l

def Tx(x):
    return -C*x

def dfI(x, a, l):
    return NI(x-a, l) * Tx(x)

def dfD(x, a, l):
    return ND(x-a, l)*Tx(x)

def dfI(x, a, l):
    return NI(x-a, l) * Tx(x)

def dfD(x, a, l):
    return ND(x-a, l)*Tx(x)
```

```
def fuerza():
    """
    define las fuerzas equivalentes sobre los nodos
    distribucuón de fuerzas T(x) = - C*x
    """
    F = np.zeros((len(MN), 1))
    for i in range(len(MN)-1):
        a = MN[i, 0]
        b = MN[i+1, 0]
        li = b - a
        F[i] += quad(lambda x: dfI(x, a, li), a, b)
```

Desplazamientos



Tensiones

$$\tau = \frac{\Delta l_e}{l_e} E$$