# Escola Politécnica da Universidade de São Paulo

# Análise matricial de estruturas não-lineares usando o Método de Newton

Exercício Computacional - MAP3121

Professor: Claudio Hirofume Asano

Turma 5

Danilo Akira Matsuyoshi 9267196

Felipe Sung Hyon Kim 9837094

Maio de 2018

São Paulo - SP

# Sumário

## 1. Introdução

Para esse primeiro exercício programável, foi utilizado o método de Newton, que através de um sistema de matrizes, itera-se afim de aproximarmo-nos da solução real do problema, podendo haver um erro mensurável da solução obtida com a solução real. Inicialmente, nos problemas de 1 a 2, foi solucionado duas diferentes estruturas, que geraram sistemas, os quais eram lineares ou não. Foi aplicado o método de Newton para obtenção dos deslocamentos das barras, bem como suas tensões e normais, e também as reações de apoio. Em Testes Iniciais, foram testadas as funções criadas, calculando os pontos de uma função onde o gradiente se anula e obtendo a solução de sistemas não lineares de dimensões variáveis.

## 2. Funções criadas:

### a) Main

Para o usuário, foi criado uma caixa de texto com opções para os problemas de 1 a 3, bem como as opções para obtenção dos gráficos. Deixando o usuário escolher por qual problema ele irá analisar. Na construção dessa função, foram criadas condições de acordo com o número do problema. Através das matrizes jacobiana, matriz de rigidez, combinando-se com as funções LU e Sistemalinear, foram determinados os deslocamentos de cada barra da estrutura nos problemas 1 e 2. Da mesma forma, foram utilizadas ambas funções para a realização dos testes iniciais, afim de obtermos o ponto onde o gradiente se anulava, e as soluções dos sistemas lineares e não lineares de dimensão variável. Para os problemas, foi utilizado um erro de $10^{-6}$ para encerrar o looping de iterações.

### b) LU

Recebe matriz A e a decompõe em LU.
É criada a matriz permutação, matriz L e matriz U.
Foi seguido o algoritmo do roteiro disponibilizado pela disciplina.
Tem como saída as matrizes L, U e p.

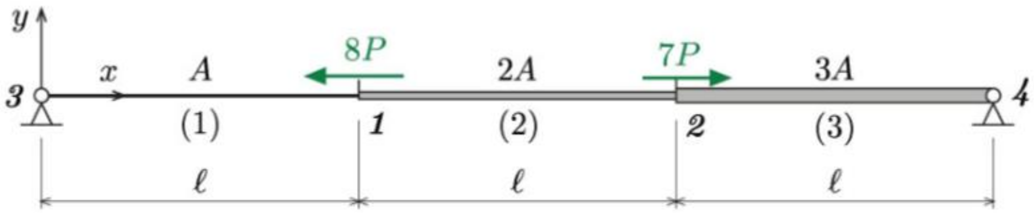
### c) Sistemalinear

Calcula Ly=B e depois Ux=y.
Tem como entrada L, U, p, B.
Primeiro permuta-se B através da matriz de permutação p, a matriz y é criada e temos o algoritmo propriamente da resolução do sistema linear já escalonado (Ly=B). Analogamente para o caso Ux=y.

## 3. Primeiro problema

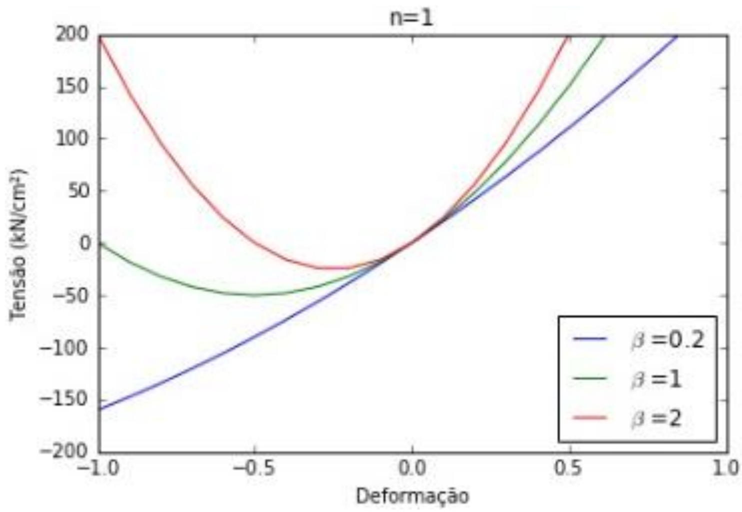
Tarefas computacionais

Segundo os dados do problema: A = 0.5 $cm^2$, E10 = E20 = E30 = E = 200.0 kN $cm^2$, P = 5.0 kN, l= 50.0 cm, n=1. Foi computacionado a resolução da estrutura apresentada na figura a seguir.

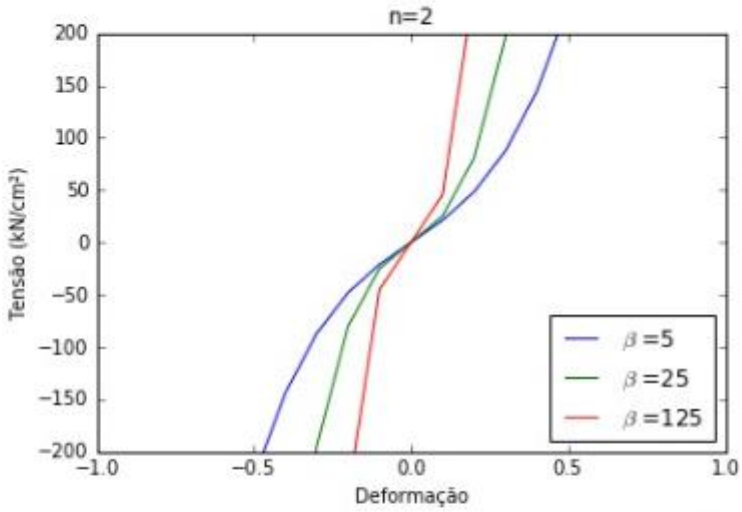



Foi utilizado um erro de $10^{-6}$ para encerrar o looping de iterações.

1. Para n = 1, foram traçadas as curvas tensão-deformação (1) para m = 1 e β = 0.2, β = 1 e β = 2 no gráfico a seguir:




Para n = 2, foram traçadas as curvas tensão-deformação (1) para m = 1 e β = 5, β = 25 e β = 125 no gráfico a seguir:




2. Começando com $u^0$= (0,0), foi implementada a iteração do método de Newton.

Primeiro teste: β = 0 (caso linear)
Foram calculadas as reações de apoio $R_3$, $R_4$, $N_m^{(k)}$, $\varepsilon_m$(d), $\sigma_m$(t); m=1, 2, 3.
Os deslocamentos são dados em cm. As reações de apoio e as forças normais, são dadas em kN.
As tensões (t) são dadas em kN/cm$^2$.

```
beta= 0

n= 1
       u1       u2       R3       R4         N1       N2         N3
1    -5.909   1.136   11.818   -6.818   -11.818   28.182   -6.818

         d1        d2        d3        t1        t2        t3
       -0.118    0.141    -0.023   -23.636    28.182    -4.545
```

3. A seguir, foi resolvido o sistema não-linear pelo método de Newton para n = 1 e β = 0.2. Em cada iteração (k) do método de Newton, foram calculados os deslocamentos $u^{(k)}{}_1$ , $u^{(k)}{}_2$ , as forças normais $N^{(k)}{}_m$ , m = 1, 2, 3, e o resíduo $R_{(k)} := |\Phi(u^{(k)}) - F|$. Foram calculadas também as reações de apoio $R_3$, $R_4$, $N_m^{(k)}$, $\varepsilon_m$(d), $\sigma_m$(t); m=1, 2, 3.

Os deslocamentos são dados em cm. As reações de apoio e as forças normais são dadas em kN.
As tensões (t) são dadas em kN/cm$^2$.

```
beta= 0.2

n= 1
      u1       u2       N1        N2        N3        Resíduo
1   -5.909   1.136   -11.539   28.976   -6.787   [[0.51], [0.76]]
2   -5.864   1.080   -11.453   28.547   -6.453   [[0.0], [0.0]]

R3        R4        d1        d2       d3        t1         t2        t3
11.453    -6.453    -0.117    0.139    -0.022    -22.905    28.547    -4.302
```

A seguir, verificamos que a matriz $D\Phi(u^{(k)})$ é simétrica:

Coluna 1:

$$\begin{pmatrix} -\frac{2AE_{20}}{l}\left(1+\beta(n+1)\left(\frac{u_2-u_1}{l}\right)^n\right)-\frac{AE_{10}}{l}\left(1+\beta(n+1)\left(\frac{u_1}{l}\right)^n\right) \\ \frac{2AE_{20}}{l}\left(1+\beta(n+1)\left(\frac{u_2-u_1}{l}\right)^n\right) \end{pmatrix}$$

Coluna 2:

$$\begin{pmatrix} \frac{2AE_{20}}{l}\left(1+\beta(n+1)\left(\frac{u_2-u_1}{l}\right)^n\right) \\ -\frac{3AE_{30}}{l}\left(1+\beta(n+1)\left(\frac{-u_2}{l}\right)^n\right)-\frac{2AE_{20}}{l}\left(1+\beta(n+1)\left(\frac{u_2-u_1}{l}\right)^n\right) \end{pmatrix}$$

4. A seguir, foi resolvido o sistema não-linear pelo método de Newton para n = 1 e $\beta$ = 1. Em cada iteração (k) do método de Newton, foram calculados os deslocamentos $u^{(k)}_1$ , $u^{(k)}_2$ , as forças normais $N^{(k)}_m$ , m = 1, 2, 3, e o resíduo $R_{(k)} := |\Phi(u^{(k)}) - F|$.

Foram calculadas também as reações de apoio $R_3$, $R_4$, $N_m^{(k)}$, $\varepsilon_m(d)$, $\sigma_m(t)$; m=1, 2, 3.

Os deslocamentos são dados em cm. As reações de apoio e as forças normais são dadas em kN.
As tensões (t) são dadas em kN/cm$^2$.

```
beta= 1

n= 1
      u1       u2       N1        N2        N3        Resíduo
1   -5.909   1.136   -10.421   32.153   -6.663   [[2.57], [3.82]]
2   -5.727   0.871   -10.142   29.873   -5.135   [[0.01], [0.01]]
3   -5.724   0.871   -10.138   29.862   -5.138   [[0.0], [0.0]]

R3        R4        d1        d2       d3        t1         t2        t3
10.138    -5.138    -0.114    0.132    -0.017    -20.275    29.862    -3.425
```

O mesmo para β = 2:

```
beta= 2

n= 1
       u1      u2      N1      N2      N3      Resíduo
1    -5.909   1.136   -9.025   36.124   -6.508   [[5.15], [7.63]]
2    -5.639   0.629   -8.734   31.357   -3.678   [[0.09], [0.03]]
3    -5.621   0.635   -8.714   31.286   -3.714   [[0.0], [0.0]]

R3      R4      d1      d2      d3      t1      t2      t3
8.714   -3.714   -0.112   0.125   -0.013   -17.428   31.286   -2.476
```

5. Executando a mesma tarefa dos 2 itens anteriores, porém, agora com expoente de não linearidade para: n = 2 e os valores β = 5, β = 25 e β = 125. Os deslocamentos são dados em cm. As reações de apoio e as forças normais são dadas em kN. As tensões (t) são dadas em kN/cm$^2$.

```
beta= 5

n= 2
       u1      u2      N1      N2      N3      Resíduo
1    -5.909   1.136   -12.644   30.980   -6.836   [[3.62], [2.82]]
2    -5.463   1.092   -11.579   28.474   -6.566   [[0.05], [0.04]]
3    -5.457   1.091   -11.563   28.437   -6.563   [[0.0], [0.0]]

R3      R4      d1      d2      d3      t1      t2      t3
11.563   -6.563   -0.109   0.131   -0.022   -23.126   28.437   -4.375
```

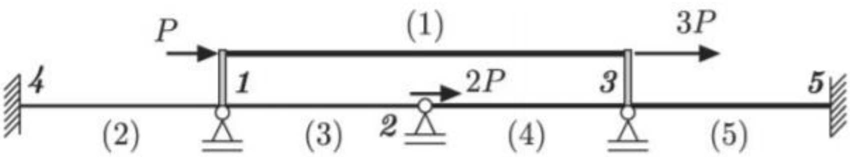
```
beta= 25

n= 2
       u1      u2      N1      N2      N3      Resíduo
1    -5.909   1.136   -15.945   42.171   -6.906   [[18.12], [14.08]]
2    -4.716   1.001   -11.530   30.339   -6.063   [[1.87], [1.4]]
3    -4.557   0.990   -11.007   29.019   -6.000   [[0.03], [0.02]]
4    -4.555   0.990   -11.000   29.000   -6.000   [[0.0], [0.0]]

R3      R4      d1      d2      d3      t1      t2      t3
11.000   -6.000   -0.091   0.111   -0.020   -21.999   29.000   -4.000
```

```
beta= 125

n= 2
       u1      u2       N1       N2        N3        Resíduo
1   -5.909   1.136   -32.451   98.127   -7.258   [[90.58], [70.39]]
2   -4.141   0.876   -15.383   45.327   -5.459   [[20.71], [15.79]]
3   -3.396   0.816   -10.706   31.790   -5.060   [[2.5], [1.85]]
4   -3.274   0.813   -10.055   29.999   -5.040   [[0.05], [0.04]]
5   -3.271   0.813   -10.041   29.959   -5.041   [[0.0], [0.0]]

R3       R4       d1       d2       d3       t1        t2       t3
10.041   -5.041   -0.065   0.082   -0.016   -20.081   29.959   -3.360
```

Comparando os resultados para n=1 e n=2, percebe-se que há semelhança no comportamento dos deslocamentos, reações, deformações e tensões quando há um aumento no valor de beta. Nota-se que, para n=1, conforme se aumenta o valor de beta, o deslocamento u2 vai diminuindo. Mas, aumentando beta de 2 para 5 e n de 1 para 2, há um aumento no deslocamento u2.

## 3. Segundo problema

Considerando que para a estrutura a seguir, a (m)-ésima barra tem a área mA; m = 1, 2, 3, 4, 5.A barra (1) tem o comprimento 2l, enquanto o comprimento das outras barras é l. Os graus de liberdade 4 e 5 são bloqueados, enquanto os graus de liberdade 1, 2, 3 são livres. Usamos os dados seguintes: $A = 0.1\ cm^2$ , $l = 20.0\ cm$, $P = 2.0\ kN$, e $E_{m0} = E = 200.0\ kN\ cm^2$ para m = 1, . . . , 5.




Foi utilizado um erro de $10^{-6}$ para encerrar o looping de iterações.

Tarefas computacionais

1. Foram efetuadas as mesmas tarefas que as tarefas 2-5 do primeiro problema para os dois casos seguintes: n = 1, β = 3; e n = 2, β = 125. Temos agora m = 1, ..., 5, e três deslocamentos $u^{(k)}_1$, $u^{(k)}_2$, $u^{(k)}_3$ que não são zeros, então os vetores $\Phi(u)$ e F têm três entradas, e a matriz de rigidez $D\Phi(u^{(k)})$ é uma matriz 3 × 3.

Os deslocamentos são dados em cm. As reações de apoio e as forças normais são dadas em kN. As tensões (t) são dadas em $kN/cm^2$.

Para n = 1, β = 3

```
beta= 3

n= 1
      u1      u2      u3      N1       N2       N3       N4       N5       Resíduo
1   1.770   2.297   1.692   -0.039   4.481    1.705   -2.201   -6.313   [[0.815], [0.095], [1.927]]
2   1.808   2.545   2.135   0.167    4.597    2.456   -1.541   -7.256   [[0.026], [0.003], [0.118]]
3   1.831   2.578   2.177   0.177    4.669    2.491   -1.509   -7.330   [[0.0], [0.0], [0.001]]
4   1.831   2.578   2.177   0.177    4.669    2.492   -1.508   -7.331   [[0.0], [0.0], [0.0]]

 R4       R5       d1      d2      d3      d4       d5       t1      t2       t3      t4       t5
-4.669   -7.331   0.009   0.092   0.037   -0.020   -0.109   1.774   23.346   8.306   -3.770   -14.662
```

Para n = 2, β = 125

```
beta= 125

n= 2
      u1      u2      u3      N1       N2       N3       N4       N5        Resíduo
1   1.770   2.297   1.692   -0.039   7.008    1.717   -2.697   -16.026   [[3.331], [0.414], [7.29]]
2   1.334   1.829   1.279   -0.027   4.151    1.600   -2.408   -9.669    [[0.579], [0.008], [1.233]]
3   1.228   1.727   1.180   -0.024   3.614    1.611   -2.389   -8.470    [[0.027], [0.0], [0.057]]
4   1.223   1.721   1.175   -0.024   3.588    1.612   -2.388   -8.412    [[0.0], [0.0], [0.0]]

 R4       R5       d1       d2      d3      d4       d5       t1       t2       t3      t4       t5
-3.588   -8.412   -0.001   0.061   0.025   -0.027   -0.059   -0.238   17.941   5.373   -5.970   -16.824
```

A seguir é verificado que a matriz de rigidez $D\Phi(u^{(k)})$ é simétrica:

Coluna 1:

$$\begin{pmatrix} -\frac{3AE_{30}}{l}\left(1+\beta(n+1)\left(\frac{u_2-u_1}{l}\right)^n\right)-\frac{AE_{10}}{2l}\left(1+\beta(n+1)\left(\frac{u_3-u_1}{2l}\right)^n\right)-\frac{2AE_{20}}{l}\left(1+\beta(n+1)\left(\frac{u_1}{l}\right)^n\right) \\ \frac{3AE_{30}}{l}\left(1+\beta(n+1)\left(\frac{u_2-u_1}{l}\right)^n\right) \\ \frac{AE_{10}}{2l}\left(1+\beta(n+1)\left(\frac{u_3-u_1}{2l}\right)^n\right) \end{pmatrix}$$

Coluna 2:

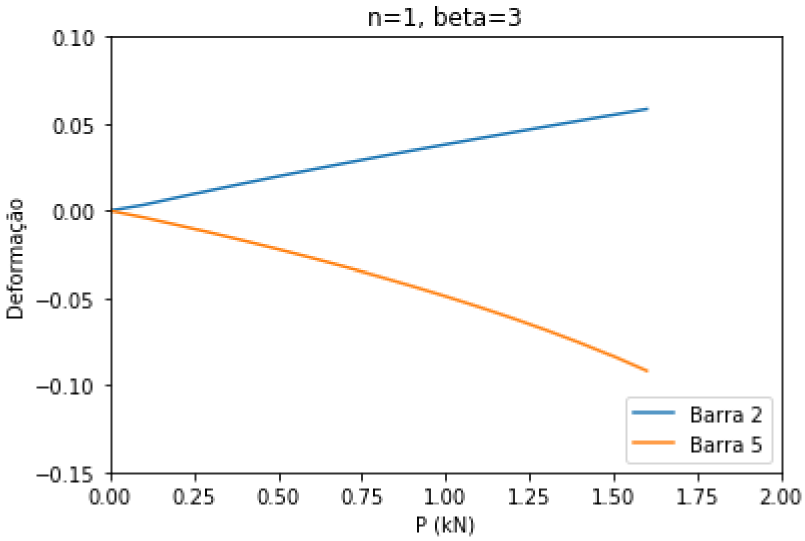
$$\begin{pmatrix} \frac{3AE_{30}}{l}\left(1+\beta(n+1)\left(\frac{u_2-u_1}{l}\right)^n\right) \\ -\frac{4AE_{40}}{l}\left(1+\beta(n+1)\left(\frac{u_3-u_2}{l}\right)^n\right)-\frac{3AE_{30}}{l}\left(1+\beta(n+1)\left(\frac{u_2-u_1}{l}\right)^n\right) \\ \frac{4AE_{40}}{l}\left(1+\beta(n+1)\left(\frac{u_3-u_2}{l}\right)^n\right) \end{pmatrix}$$

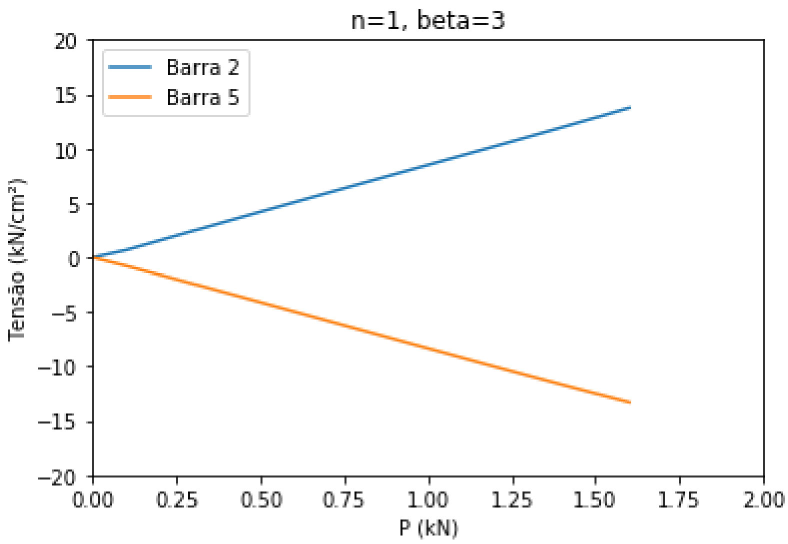
Coluna 3:

$$\begin{pmatrix} \frac{AE_{10}}{2l}\left(1+\beta(n+1)\left(\frac{u_3-u_1}{2l}\right)^n\right) \\ \frac{4AE_{40}}{l}\left(1+\beta(n+1)\left(\frac{u_3-u_2}{l}\right)^n\right) \\ -\frac{5AE_{50}}{l}\left(1+\beta(n+1)\left(\frac{-u_3}{l}\right)^n\right)-\frac{AE_{10}}{2l}\left(1+\beta(n+1)\left(\frac{u_3-u_1}{2l}\right)^n\right)-\frac{4AE_{40}}{l}\left(1+\beta(n+1)\left(\frac{u_3-u_2}{l}\right)^n\right) \end{pmatrix}$$

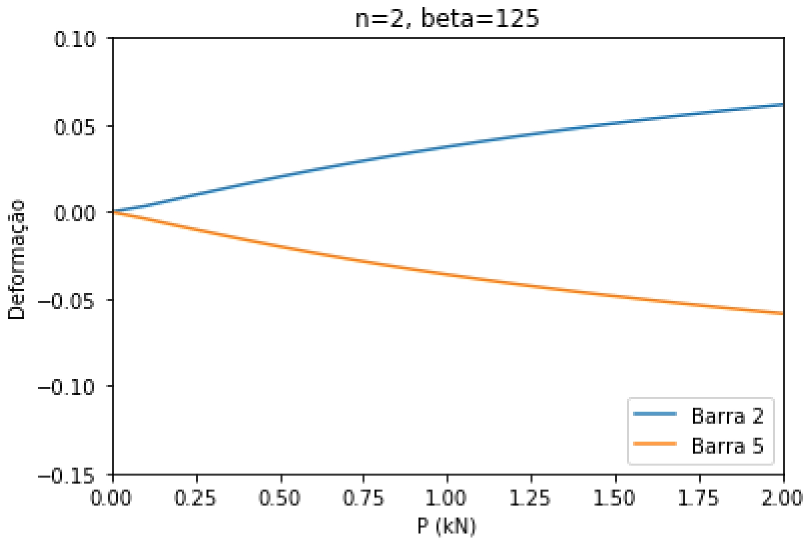
2. Foram traçados os gráficos das curvas das tensões e das deformações em função da carga P, nas barras com as maiores tensões de tração e de compressão, correspondendo aos maiores alongamentos e encurtamentos. Os gráficos foram traçados apenas para os dois casos seguintes: n = 1, β = 5 (para P = 0, . . . , 1.6 kN); e n = 2, β = 125 (para P = 0, . . . , 4.0 kN).

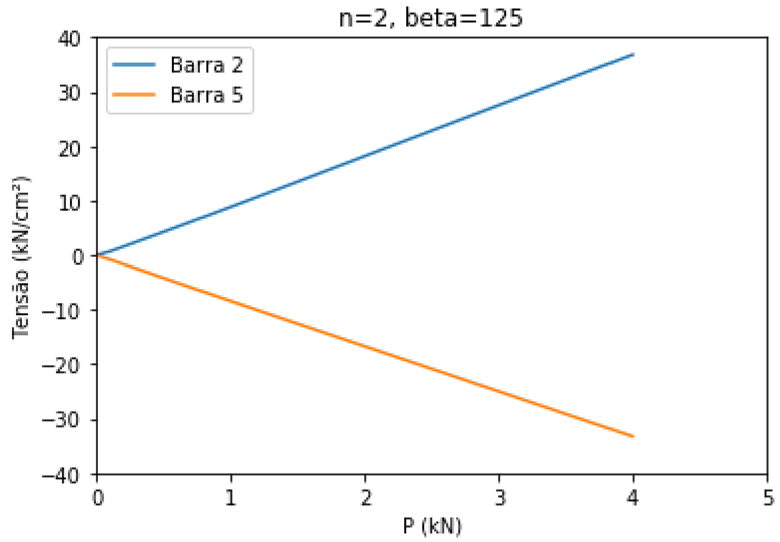
n = 1, β = 5 (para P = 0, . . . , 1.6 kN)



n = 2, β = 125 (para P = 0, . . . , 4.0 kN)




## 4. Testes iniciais

Testes iniciais

- Usando o código desenvolvido para determinar o ponto de mínimo da função $F(x, y) = (x - 2)^2 + (y - 3)^2$ ,
  Ponto de mínimo:

  Onde o seu gradiente se anula ou para esse caso,

  ```
  O ponto de mínimo da função F(x,y) = (x-2)² + (y-3)² é:
  (x, y) = (2.00, 3.00)
  ```

  Número de iterações do método de Newton necessárias para convergência: 2

- Dada a função $F(x_1, x_2, x_3, x_4) = (4x_1 - x_2 + x_3 - x_1x_4, -x_1 + 3x_2 - 2x_3 - x_2x_4, x_1 - 2x_2 + 3x_3 - x_3x_4, x_1^2 + x_2^2 + x_3^2 - 1)$, a raiz que se obtém pelo método de Newton tomando x = (1, 1, 1, 1) como valor inicial é:

```
x1 = 0.00 , x2 = 0.71 , x3 = 0.71 , x4 = 1.00
```

- O método de Newton foi utilizado para determinar a solução do sistema $n - 1 \times n - 1$, cujas equações são $-x_{i-1} + 2x_i - x_{i+1} = e^{x_i}/n^2$, i = 1, 2, ..., n-1.
  Com $x_0 = x_n = 0$, a partir da aproximação inicial nula.

Para n=20

| n = 20 |
|---|
| x1 = 0.026060 |
| x2 = 0.049555 |
| x3 = 0.070426 |
| x4 = 0.088621 |
| x5 = 0.104094 |
| x6 = 0.116807 |
| x7 = 0.126728 |
| x8 = 0.133832 |
| x9 = 0.138102 |

| |
|---|
| x10 = 0.139526 |
| x11 = 0.138102 |
| x12 = 0.133832 |
| x13 = 0.126728 |
| x14 = 0.116807 |
| x15 = 0.104094 |
| x16 = 0.088621 |
| x17 = 0.070426 |
| x18 = 0.049555 |
| x19 = 0.026060 |

Para n=40

| n = 40 |
|---|
| x1 = 0.013344 |
| x2 = 0.026055 |
| x3 = 0.038125 |
| x4 = 0.049546 |
| x5 = 0.060311 |
| x6 = 0.070414 |
| x7 = 0.079847 |
| x8 = 0.088605 |
| x9 = 0.096683 |
| x10 = 0.104076 |
| x11 = 0.110778 |
| x12 = 0.116787 |
| x13 = 0.122097 |
| x14 = 0.126706 |
| x15 = 0.130611 |
| x16 = 0.133809 |
| x17 = 0.136299 |
| x18 = 0.138078 |
| x19 = 0.139146 |

| |
|---|
| x20 = 0.139502 |
| x21 = 0.139146 |
| x22 = 0.138078 |
| x23 = 0.136299 |
| x24 = 0.133809 |
| x25 = 0.130611 |
| x26 = 0.126706 |
| x27 = 0.122097 |
| x28 = 0.116787 |
| x29 = 0.110778 |
| x30 = 0.104076 |
| x31 = 0.096683 |
| x32 = 0.088605 |
| x33 = 0.079847 |
| x34 = 0.070414 |
| x35 = 0.060311 |
| x36 = 0.049546 |
| x37 = 0.038125 |
| x38 = 0.026055 |
| x39 = 0.013344 |

Para n=80

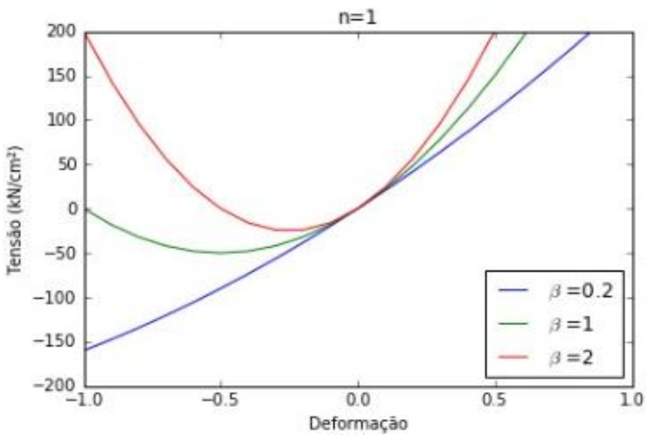
| n = 80 |
|---|
| x1 = 0.006751 |
| x2 = 0.013344 |
| x3 = 0.019779 |
| x4 = 0.026054 |
| x5 = 0.032170 |
| x6 = 0.038124 |
| x7 = 0.043915 |
| x8 = 0.049544 |
| x9 = 0.055009 |
| x10 = 0.060309 |
| x11 = 0.065443 |
| x12 = 0.070411 |
| x13 = 0.075211 |
| x14 = 0.079843 |
| x15 = 0.084307 |
| x16 = 0.088601 |
| x17 = 0.092726 |
| x18 = 0.096679 |
| x19 = 0.100461 |
| x20 = 0.104071 |
| x21 = 0.107509 |
| x22 = 0.110774 |
| x23 = 0.113865 |
| x24 = 0.116782 |
| x25 = 0.119524 |
| x26 = 0.122092 |
| x27 = 0.124484 |
| x28 = 0.126700 |
| x29 = 0.128741 |
| x30 = 0.130605 |
| x31 = 0.132293 |
| x32 = 0.133803 |
| x33 = 0.135137 |
| x34 = 0.136293 |
| x35 = 0.137271 |
| x36 = 0.138072 |
| x37 = 0.138695 |
| x38 = 0.139140 |
| x39 = 0.139407 |

| |
|---|
| x40 = 0.139496 |
| x41 = 0.139407 |
| x42 = 0.139140 |
| x43 = 0.138695 |
| x44 = 0.138072 |
| x45 = 0.137271 |
| x46 = 0.136293 |
| x47 = 0.135137 |
| x48 = 0.133803 |
| x49 = 0.132293 |
| x50 = 0.130605 |
| x51 = 0.128741 |
| x52 = 0.126700 |
| x53 = 0.124484 |
| x54 = 0.122092 |
| x55 = 0.119524 |
| x56 = 0.116782 |
| x57 = 0.113865 |
| x58 = 0.110774 |
| x59 = 0.107509 |
| x60 = 0.104071 |
| x61 = 0.100461 |
| x62 = 0.096679 |
| x63 = 0.092726 |
| x64 = 0.088601 |
| x65 = 0.084307 |
| x66 = 0.079843 |
| x67 = 0.075211 |
| x68 = 0.070411 |
| x69 = 0.065443 |
| x70 = 0.060309 |
| x71 = 0.055009 |
| x72 = 0.049544 |
| x73 = 0.043915 |
| x74 = 0.038124 |
| x75 = 0.032170 |
| x76 = 0.026054 |
| x77 = 0.019779 |
| x78 = 0.013344 |
| x79 = 0.006751 |

## 5. Conclusões

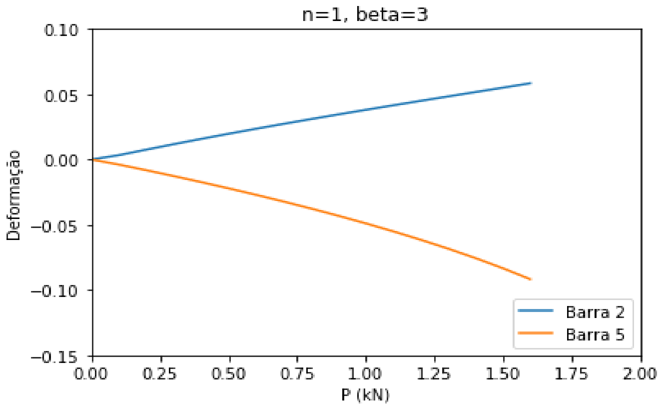
O exercício programável foi importante para a dupla ver uma das aplicações do cálculo numérico, seja o método de Newton ou do escalonamento de matrizes. Por meio da ferramenta Python foi possível obtermos a solução de diferentes estruturas de barras, com seus respectivos deslocamentos, tensões, reações e normais. Foi verificado que o problema é solucionável através de matrizes, aplicando-se o método de Newton.

No problema 1, como mostra o gráfico abaixo, já enunciado relatório do exercício computacional, verificamos na barra da estrutura, a variação da tensão com a deformação.




Para a dupla, que está cursando a Engenharia Civil, o exercício foi de grande interesse e aplicabilidade, visto que constantemente calculamos todas as variáveis de uma estrutura, afim de obtermos o equilíbrio estável de uma edificação, com um coeficiente de segurança. Desta forma, vimos que estruturas que aparentam ter alta complexidade de resolução, podem ser sistematizadas e solucionadas através de algoritmos estudados por exemplo no cálculo numérico. Vale citar que na matéria de resistência dos materiais, por exemplo, o método de Newton será utilizado para analisar o equilíbrio de uma estrutura (problema de Euler) em que a solução será aproximada através de iterações a partir de um valor próximo da solução real.

No problema 2, em uma estrutura de maior complexidade em comparação com a estrutura do problema 1, foi verificada a variação da deformação em função de um carregamento, como mostra o gráfico abaixo, bem como a variação da tensão em função do carregamento. Novamente, um problema que não é facilmente mensurável, através de algoritmos de aproximações, pode-se obter a relação entre tais parâmetros e solicitações.




Além disso, foi obtido os valores em que o gradiente de uma função se anula, bem como as raízes de uma função obtidas pelo método de Newton, iterando-se a partir de um valor inicial. Foi solucionado também um sistema na ferramenta Python que recebe valores que determinam a dimensão da matriz quadrada bem como os elementos da matriz. Sistemas com a matriz quadrada de dimensão 79, puderam ser solucionados através dos algoritmos montados no exercício. Através deles, analisamos como são calculadas soluções de um sistema linear ou não, com precisão determinada.