Exercício 6

December 8, 2020

1 Exercício PA3-6

Exercício com data de entrega para 9 de dezembro de 2020.

Aluno: Noé de Lima Bezerra

noe_lima@id.uff.br

[1]:
```
import numpy as np
import sympy as sp
import pandas as pd
import matplotlib.pyplot as plt
from IPython.display import display, Math, Image, IFrame
from sympy.abc import x, y, z
sp.init_printing(use_latex='mathjax',latex_mode='equation*')
```

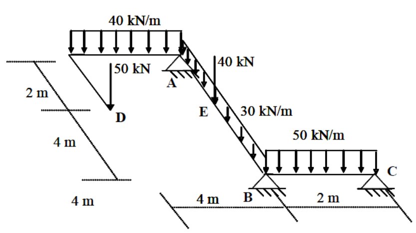
[2]:
```
Image("Figuras/PA3-6.png")
```

[2]:

AVALIAÇÃO 6-PA3 – PRINCÍPIO DOS TRABALHOS VIRTUAIS– EXERCÍCIO 3

Calcular o deslocamento vertical da seção D da grelha da figura. $EI_c = 10^5$ kN.m² e $EI_C/GJ_T = 2$

Observação: apresentar de forma detalhada as equações de equilíbrio para o cálculo das reações de apoio e a obtenção dos momentos fletores nas seções A e E.



2 Formulário

2.1 Princípio dos Trabalhos Virtuais

$$\delta = \sum_{i=1}^{n} \int_{l_i} \frac{\bar{M}_i M_i}{EI_{Ci}} \mathrm{d}x + \sum_{j=1}^{m} \int_{l_j} \frac{\bar{T}_j T_j}{GJ_{Tj}} \mathrm{d}x$$

2.2 Temperatura

2.2.1 Barras Com Seção Transversal Constante

$$\delta = \alpha \cdot \delta t_g \cdot A_{\bar{N}} + \frac{\alpha \cdot \Delta t}{h} A_{\bar{M}}$$

Onde,

- $A_{\bar{N}}$ é a área dos diagramas de esforço normal;
- $A_{\bar{M}}$ é a área dos diagramas de momento fletor;
- δt_g é a variação média de temperatura;
- Δt é a diferença de temperatura $\Delta t = \delta t_i - \delta t_e$ (temperatura interna menos a temperatura externa).

2.2.2 Barras Com Seção Transversal Variável

$$\delta = \alpha \int_l \bar{N} \delta t_g \mathrm{d}s + \alpha \cdot \Delta t \int_l \frac{\bar{M}}{h} \mathrm{d}s$$

2.3 Recalque

$$\delta = -\sum \bar{R} \cdot \rho$$

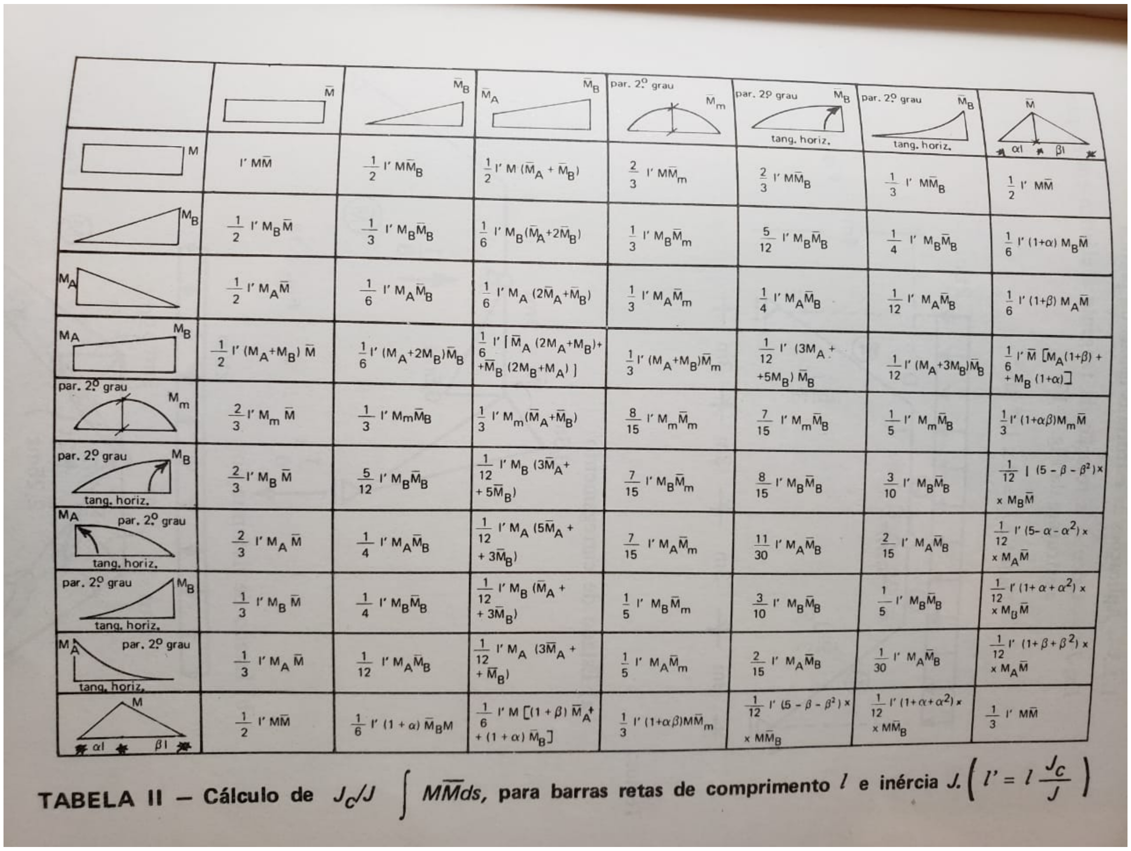
2.4 Utilização

Calcular o deslocamento para os esforços no Estado de Deformação, em seguida, calcular o deslocamento em função da Temperatura e, em seguida, o deslocamento em função dos recalques. Por último, devido à linearidade, utilizar a superposição de efeitos e somar os resultados.

Para fins práticos, temos uma tabela com diversas integrais comuns já calculadas e disponíveis para utilização.

[3]:
```
Image("Figuras/tabela integrais.jpg")
```

[3]:



	$\bar{M}$	$\bar{M}_B$	$\bar{M}_A$ $\bar{M}_B$	par. 2º grau $\bar{M}_m$	par. 2º grau $\bar{M}_B$ tang. horiz.	par. 2º grau $\bar{M}_B$ tang. horiz.	$\bar{M}$ αl βl
M	$l' M\bar{M}$	$\frac{1}{2} l' M\bar{M}_B$	$\frac{1}{2} l' M(\bar{M}_A+\bar{M}_B)$	$\frac{2}{3} l' M\bar{M}_m$	$\frac{2}{3} l' M\bar{M}_B$	$\frac{1}{3} l' M\bar{M}_B$	$\frac{1}{2} l' M\bar{M}$
M_B	$\frac{1}{2} l' M_B\bar{M}$	$\frac{1}{3} l' M_B\bar{M}_B$	$\frac{1}{6} l' M_B(\bar{M}_A+2\bar{M}_B)$	$\frac{1}{3} l' M_B\bar{M}_m$	$\frac{5}{12} l' M_B\bar{M}_B$	$\frac{1}{4} l' M_B\bar{M}_B$	$\frac{1}{6} l'(1+\alpha) M_B\bar{M}$
M_A	$\frac{1}{2} l' M_A\bar{M}$	$\frac{1}{6} l' M_A\bar{M}_B$	$\frac{1}{6} l' M_A(2\bar{M}_A+\bar{M}_B)$	$\frac{1}{3} l' M_A\bar{M}_m$	$\frac{1}{4} l' M_A\bar{M}_B$	$\frac{1}{12} l' M_A\bar{M}_B$	$\frac{1}{6} l'(1+\beta) M_A\bar{M}$
M_A M_B	$\frac{1}{2} l'(M_A+M_B)\bar{M}$	$\frac{1}{6} l'(M_A+2M_B)\bar{M}_B$	$\frac{1}{6} l'[\bar{M}_A(2M_A+M_B)+\bar{M}_B(2M_B+M_A)]$	$\frac{1}{3} l'(M_A+M_B)\bar{M}_m$	$\frac{1}{12} l'(3M_A+5M_B)\bar{M}_B$	$\frac{1}{12} l'(M_A+3M_B)\bar{M}_B$	$\frac{1}{6} l'\bar{M}[M_A(1+\beta)+M_B(1+\alpha)]$
par. 2º grau M_m	$\frac{2}{3} l' M_m\bar{M}$	$\frac{1}{3} l' M_m\bar{M}_B$	$\frac{1}{3} l' M_m(\bar{M}_A+\bar{M}_B)$	$\frac{8}{15} l' M_m\bar{M}_m$	$\frac{7}{15} l' M_m\bar{M}_B$	$\frac{1}{5} l' M_m\bar{M}_B$	$\frac{1}{3} l'(1+\alpha\beta)M_m\bar{M}$
par. 2º grau M_B tang. horiz.	$\frac{2}{3} l' M_B\bar{M}$	$\frac{5}{12} l' M_B\bar{M}_B$	$\frac{1}{12} l' M_B(3\bar{M}_A+5\bar{M}_B)$	$\frac{7}{15} l' M_B\bar{M}_m$	$\frac{8}{15} l' M_B\bar{M}_B$	$\frac{3}{10} l' M_B\bar{M}_B$	$\frac{1}{12} l'(5-\beta-\beta^2)\times M_B\bar{M}$
M_A par. 2º grau tang. horiz.	$\frac{2}{3} l' M_A\bar{M}$	$\frac{1}{4} l' M_A\bar{M}_B$	$\frac{1}{12} l' M_A(5\bar{M}_A+3\bar{M}_B)$	$\frac{7}{15} l' M_A\bar{M}_m$	$\frac{11}{30} l' M_A\bar{M}_B$	$\frac{2}{15} l' M_A\bar{M}_B$	$\frac{1}{12} l'(5-\alpha-\alpha^2)\times M_A\bar{M}$
par. 2º grau M_B tang. horiz.	$\frac{1}{3} l' M_B\bar{M}$	$\frac{1}{4} l' M_B\bar{M}_B$	$\frac{1}{12} l' M_B(\bar{M}_A+3\bar{M}_B)$	$\frac{1}{5} l' M_B\bar{M}_m$	$\frac{3}{10} l' M_B\bar{M}_B$	$\frac{1}{5} l' M_B\bar{M}_B$	$\frac{1}{12} l'(1+\alpha+\alpha^2)\times M_B\bar{M}$
M_A par. 2º grau tang. horiz.	$\frac{1}{3} l' M_A\bar{M}$	$\frac{1}{12} l' M_A\bar{M}_B$	$\frac{1}{12} l' M_A(3\bar{M}_A+\bar{M}_B)$	$\frac{1}{5} l' M_A\bar{M}_m$	$\frac{2}{15} l' M_A\bar{M}_B$	$\frac{1}{30} l' M_A\bar{M}_B$	$\frac{1}{12} l'(1+\beta+\beta^2)\times M_A\bar{M}$
M αl βl	$\frac{1}{2} l' M\bar{M}$	$\frac{1}{6} l'(1+\alpha)\bar{M}_B M$	$\frac{1}{6} l' M[(1+\beta)\bar{M}_A+(1+\alpha)\bar{M}_B]$	$\frac{1}{3} l'(1+\alpha\beta)M\bar{M}_m$	$\frac{1}{12} l'(5-\beta-\beta^2)\times M\bar{M}_B$	$\frac{1}{12} l'(1+\alpha+\alpha^2)\times M\bar{M}_B$	$\frac{1}{3} l' M\bar{M}$

TABELA II — Cálculo de $J_c/J \int M\bar{M}ds$, para barras retas de comprimento l e inércia J. $\left(l' = l\,\frac{J_c}{J} \right)$

3 Solução

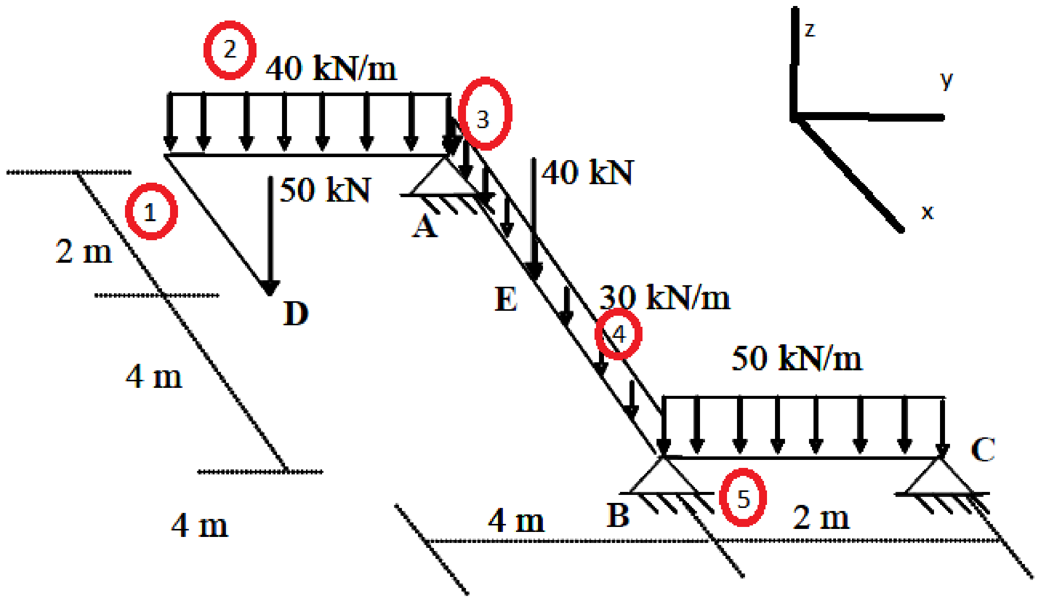
```
[4]: kN,m,C = sp.symbols('kN,m,C',positive=True)
     M,M_b = sp.symbols('M,M_b',cls=sp.Function)
     EI,l = sp.symbols('EI,l',positive=True)
     delta_i = sp.Integral(M(x)*M_b(x)/EI,(x,0,l))
     display(delta_i)
```

$$\int\limits_0^l \frac{M(x)\,\mathrm{M_b}(x)}{EI}\,dx$$

Para integrar os momentos fletores, as barras seguirão a numeração da figura abaixo:

```
[5]: Image("Figuras/PA3-6-0.png")
```

[5]:



3.1 Reações de Apoio

3.1.1 Estado de Deformação

Neste sistema temos cargas solicitantes e reações de apoio apenas na direção z. Temos, portanto, três variáveis a calcular, sendo elas:

- F_A
- F_B
- F_C

Para tanto, podemos utilizar três equações de equilíbrio:

- $\sum_{M_{Ay}} = 0$
- $\sum_{M_{By}} = 0$
- $\sum_{M_{Bx}} = 0$

$$\sum_{M_{Ay}} = 0 \therefore$$

$$50 \times 2 + 40 \times 2 + 30 \times \frac{6^2}{2} + 50 \times 2 \times 6 = (F_B + F_C) \times 6 \therefore$$

$$F_B + F_C = 220 \;\; kN$$

$$\sum_{M_{By}} = 0 \therefore$$

$$F_A \times 6 = 30 \times \frac{6^2}{2} + 40 \times 4 + 50 \times 4 + 40 \times 4 \times 6 \therefore$$

$$F_A = 310 \ \ kN$$

$$\sum_{M_{Bx}} = 0 \therefore$$

$$50 \times 4 + 40 \times \frac{4^2}{2} + F_C \times 2 = 50 \times \frac{2^2}{2}$$

$$F_C = -210 \ \ kN$$

Como $F_B + F_C = 220 \ \ kN$,

$$F_B = 430 \ \ kN$$

Assim, temos as reações no Estado de Deformação:

- $F_A = 310 \ \ kN$
- $F_B = 430 \ \ kN$
- $F_C = -210 \ \ kN$

3.1.2 Estado de Carregamento

Como estamos interessados no deslocamento vertical na seção do ponto D, vamos inserir uma força virtual $P = 1$ no ponto D orientada para baixo. Assim, temos as seguintes reações de apoio, obtidas pelas mesmas equações de equilíbrio:

$$\sum_{M_{Ay}} = 0 \therefore$$

$$1 \times 2 = \left(F'_B + F'_C\right) \times 6 \therefore$$

$$F'_B + F'_C = \frac{1}{3}$$

$$\sum_{M_{By}} = 0 \therefore$$

$$F'_A \times 6 = 1 \times 4 \therefore$$

$$F'_A = \frac{2}{3}$$

$$\sum_{M_{Bx}} = 0 \therefore$$

$$1 \times 4 + F'_C \times 2 = 0 \therefore$$

$$F'_C = -2$$

Como $F'_B + F'_C = \frac{1}{3}$, temos:

$$F'_B = \frac{1}{3} + 2 = \frac{7}{3}$$

Assim, temos as reações no Estado de Carregamento:

- $F'_A = \frac{2}{3}$
- $F'_B = \frac{7}{3}$
- $F'_C = -2$

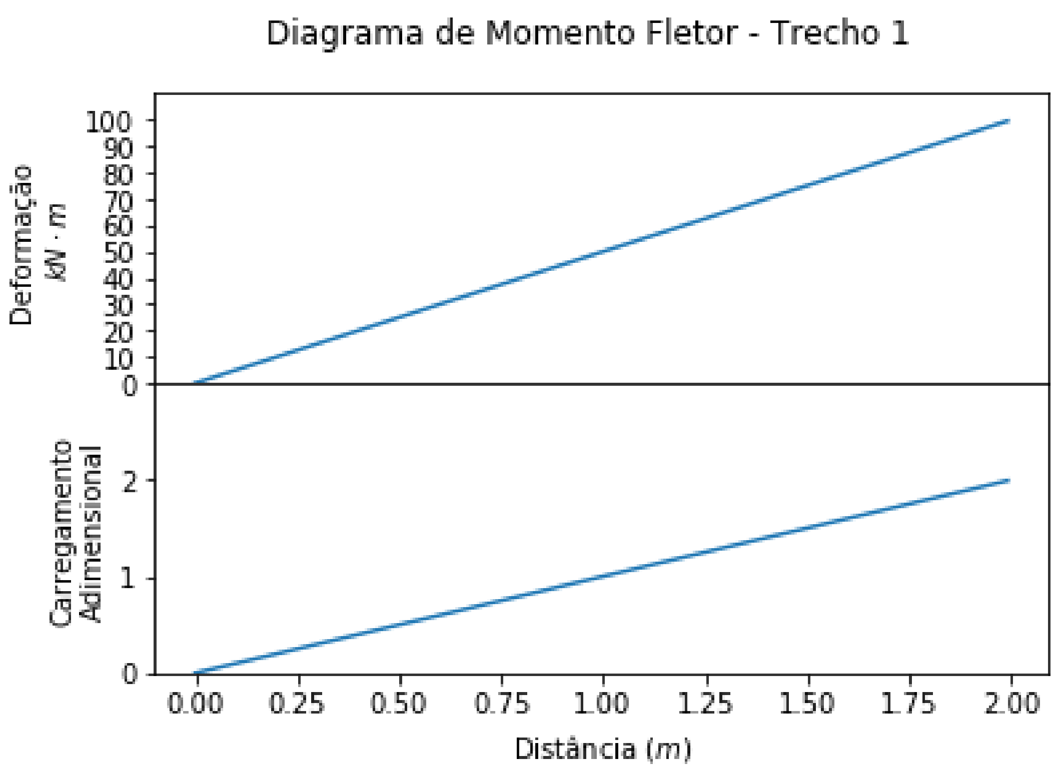
Daqui em diante, os diagramas serão plotados em eixos locais, sempre percorrendo a estrutura no sentido de D para C. Os sinais foram adaptados para que os gráficos fiquem no sentido correto, pois mantendo a posição relativa entre eles, o sinal do produto não se altera.

3.2 Trecho 1

[6]:
```
t = np.arange(0.0, 2.0, 0.01)

s1 = 50*t
s2 = t

fig, axs = plt.subplots(2, 1, sharex=True)
# Remove horizontal space between axes
fig.subplots_adjust(hspace=0)
fig.suptitle('Diagrama de Momento Fletor - Trecho 1')


# Plot each graph, and manually set the y tick values
axs[0].plot(t, s1)
axs[0].set_yticks(np.arange(0., 110.0, 10))
axs[0].set_ylim(0, 110)
```

```python
axs[0].set_ylabel('Deformação\n$kN\cdot m$')

axs[1].plot(t, s2)
axs[1].set_yticks(np.arange(0., 3.0, 1))
axs[1].set_ylim(0, 3)
axs[1].set_ylabel('Carregamento\nAdimensional')
axs[1].set_xlabel('Distância ($m$)')

plt.show()
```

```
<>:16: DeprecationWarning: invalid escape sequence \c
<>:16: DeprecationWarning: invalid escape sequence \c
<ipython-input-6-2a9a7d82e72f>:16: DeprecationWarning: invalid escape sequence
\c
  axs[0].set_ylabel('Deformação\n$kN\cdot m$')
```



Este trecho apresenta apenas área triangular, em ambos os estados, sendo as duas no mesmo sentido. Assim, a equação para este caso fica:

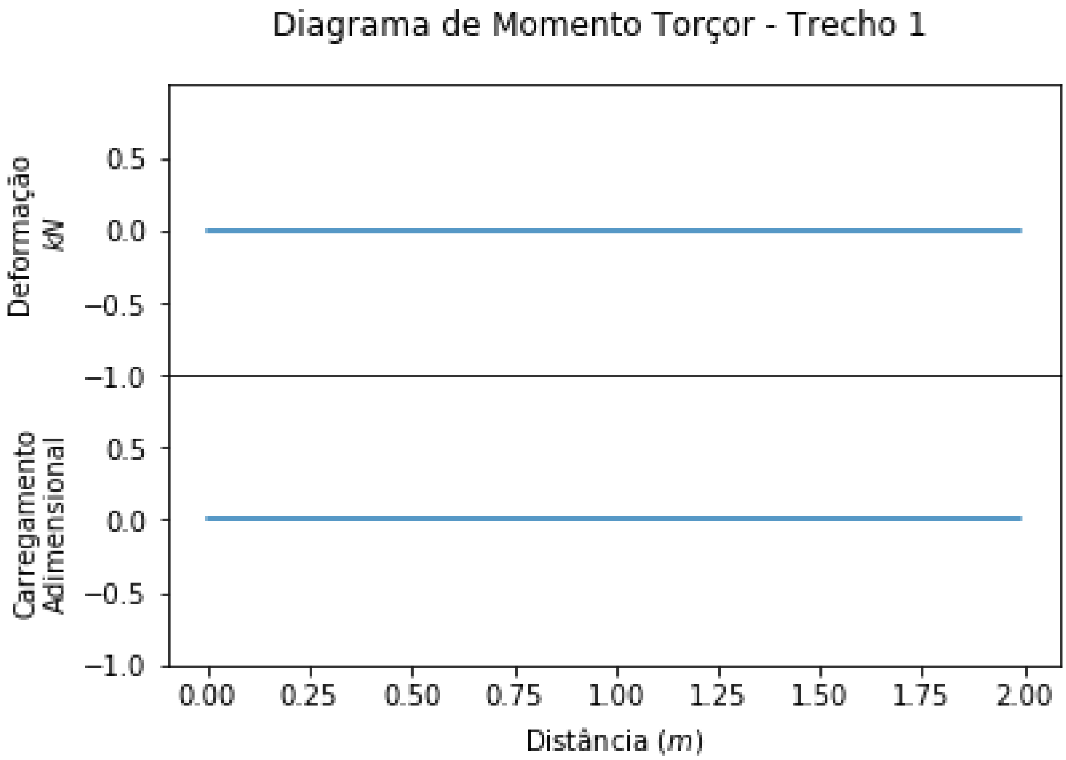
$$\frac{1}{3}l'M_B\bar{M}_B$$

Onde $l' = l\frac{J_C}{J} = \frac{l}{EI}$

[7]:
```
EIdelta_M1 = 2*100*2/3
display(EIdelta_M1)
```

133.33333333333334

[8]:
```
t = np.arange(0.0, 2.0, 0.01)

s1 = 0*t
s2 = 0*t

fig, axs = plt.subplots(2, 1, sharex=True)
# Remove horizontal space between axes
fig.subplots_adjust(hspace=0)
fig.suptitle('Diagrama de Momento Torçor - Trecho 1')


# Plot each graph, and manually set the y tick values
axs[0].plot(t, s1)
axs[0].set_yticks(np.arange(-1., 1., 0.5))
axs[0].set_ylim(-1, 1)
axs[0].set_ylabel('Deformação\n$kN$')

axs[1].plot(t, s2)
axs[1].set_yticks(np.arange(-1., 1., 0.5))
axs[1].set_ylim(-1, 1)
axs[1].set_ylabel('Carregamento\nAdimensional')
axs[1].set_xlabel('Distância ($m$)')

plt.show()
```



[9]:
```
EIdelta_T1 = 0
display(EIdelta_T1)
```

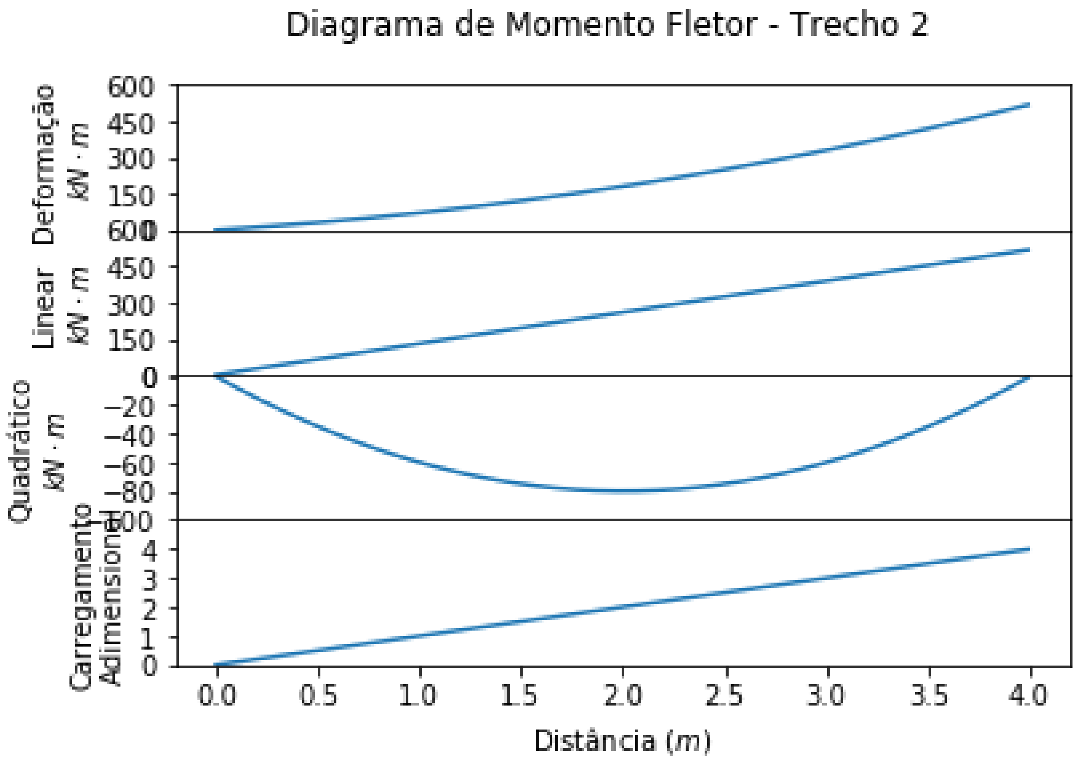
0

3.3 Trecho 2

[10]:
```
t = np.arange(0.0, 4.0, 0.01)

s1 = 50*t+40*(t**2)/2
s2 = 520*t/4
s3 = 40*(t**2 - 4*t)/2
s4 = t

fig, axs = plt.subplots(4, 1, sharex=True)
# Remove horizontal space between axes
fig.subplots_adjust(hspace=0)
fig.suptitle('Diagrama de Momento Fletor - Trecho 2')


# Plot each graph, and manually set the y tick values
```

```python
axs[0].plot(t, s1)
axs[0].set_yticks(np.arange(0., 601., 150.))
axs[0].set_ylim(0, 601)
axs[0].set_ylabel('Deformação\n$kN\cdot m$')

axs[1].plot(t, s2)
axs[1].set_yticks(np.arange(0., 601., 150.))
axs[1].set_ylim(0, 601)
axs[1].set_ylabel('Linear\n$kN\cdot m$')

axs[2].plot(t, s3)
axs[2].set_yticks(np.arange(-100., 1., 20.))
axs[2].set_ylim(-100, 1)
axs[2].set_ylabel('Quadrático\n$kN\cdot m$')

axs[3].plot(t, s4)
axs[3].set_yticks(np.arange(0., 5.0, 1.))
axs[3].set_ylim(0, 5)
axs[3].set_ylabel('Carregamento\nAdimensional')
axs[3].set_xlabel('Distância ($m$)')

plt.show()
```

```
<>:18: DeprecationWarning: invalid escape sequence \c
<>:23: DeprecationWarning: invalid escape sequence \c
<>:28: DeprecationWarning: invalid escape sequence \c
<>:18: DeprecationWarning: invalid escape sequence \c
<>:23: DeprecationWarning: invalid escape sequence \c
<>:28: DeprecationWarning: invalid escape sequence \c
<ipython-input-10-85279ea530c0>:18: DeprecationWarning: invalid escape sequence
\c
  axs[0].set_ylabel('Deformação\n$kN\cdot m$')
<ipython-input-10-85279ea530c0>:23: DeprecationWarning: invalid escape sequence
\c
  axs[1].set_ylabel('Linear\n$kN\cdot m$')
<ipython-input-10-85279ea530c0>:28: DeprecationWarning: invalid escape sequence
\c
  axs[2].set_ylabel('Quadrático\n$kN\cdot m$')
```



Temos, portanto, duas componentes a integrar separadamente e somar por superposição. Uma componente linear, e outra quadrática.

Temos as seguintes equações:

3.3.1 Linear

$$\frac{1}{3}l' M_B \bar{M}_B$$

3.3.2 Parábola

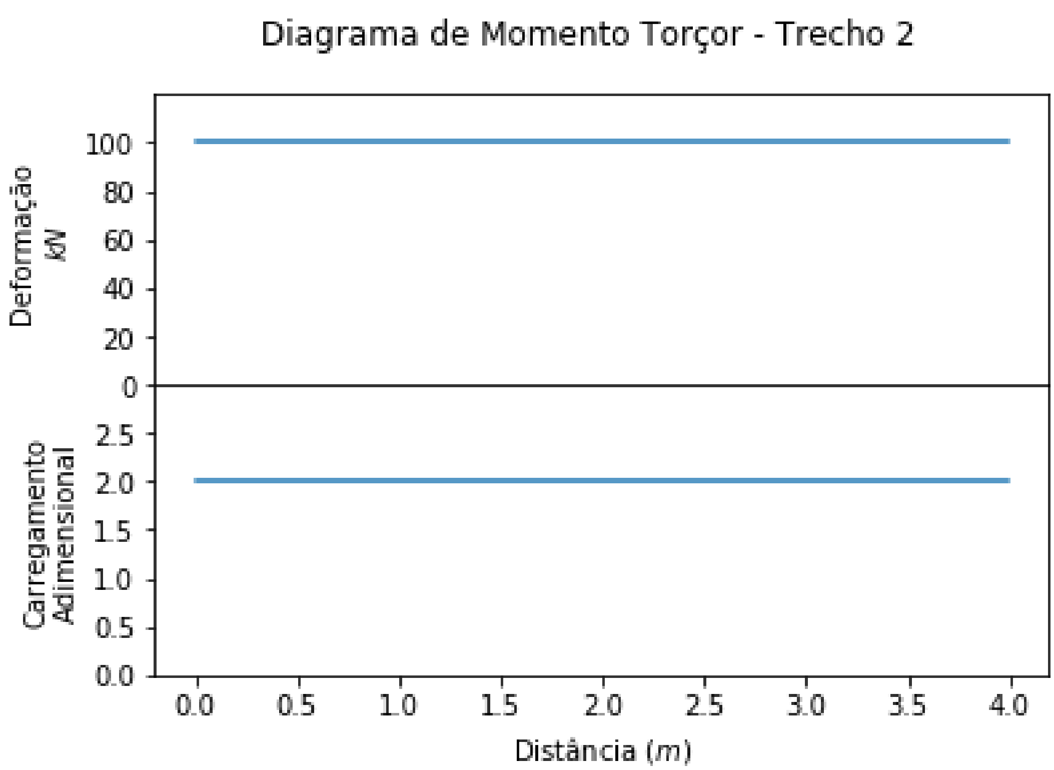
$$\frac{1}{3}l' M_m \bar{M}_B$$

[11]:
```
EIdelta_M2 = 4*520*4/3 - 4*80*4/3
display(EIdelta_M2)
```

2346.666666666667

[12]:
```
t = np.arange(0.0, 4.0, 0.01)

s1 = 100+0*t
s2 = 2+0*t
```

```python
fig, axs = plt.subplots(2, 1, sharex=True)
# Remove horizontal space between axes
fig.subplots_adjust(hspace=0)
fig.suptitle('Diagrama de Momento Torçor - Trecho 2')


# Plot each graph, and manually set the y tick values
axs[0].plot(t, s1)
axs[0].set_yticks(np.arange(0., 120., 20))
axs[0].set_ylim(0, 120)
axs[0].set_ylabel('Deformação\n$kN$')

axs[1].plot(t, s2)
axs[1].set_yticks(np.arange(0., 3., 0.5))
axs[1].set_ylim(0, 3)
axs[1].set_ylabel('Carregamento\nAdimensional')
axs[1].set_xlabel('Distância ($m$)')

plt.show()
```



[13]:
```python
EIdelta_T2 = 2*4*100*2
display(EIdelta_T2)
```

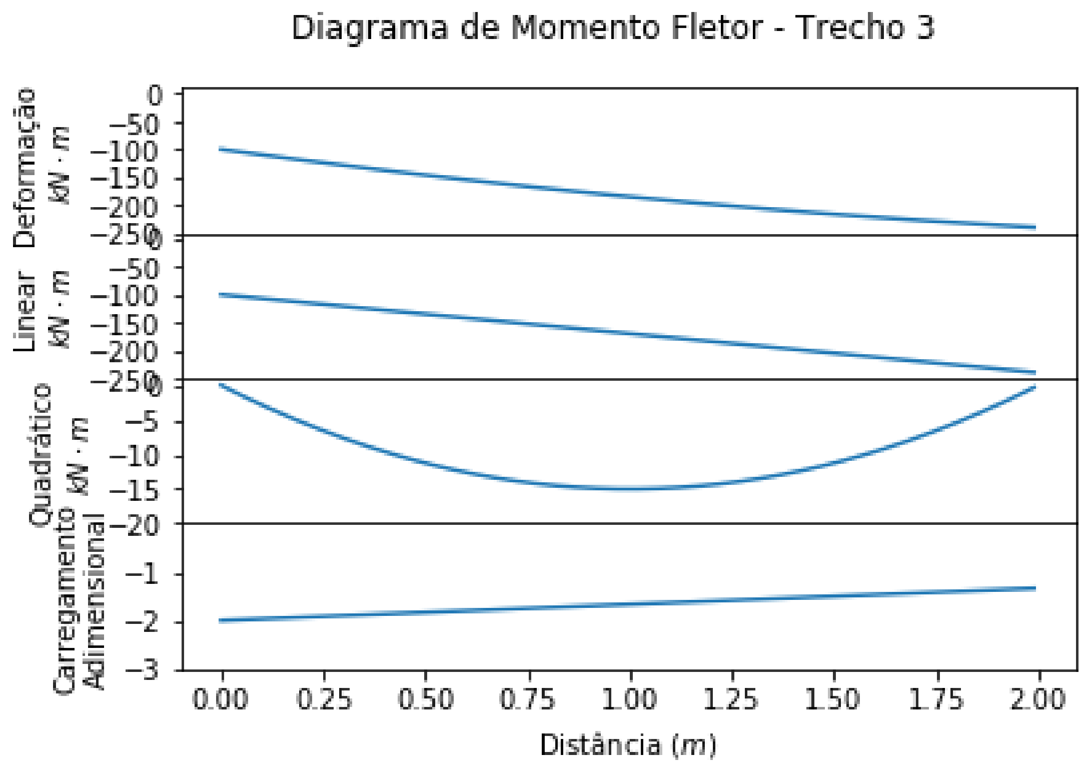
1600

3.4 Trecho 3

[14]:
```python
t = np.arange(0.0, 2.0, 0.01)

s1 = -(310-50-40*4)*t + 30*(t**2)/2 - 100
s2 = -(240-100)*t/2 - 100
s3 = 30*(t**2 - 2*t)/2
s4 = t/3 - 2

fig, axs = plt.subplots(4, 1, sharex=True)
# Remove horizontal space between axes
fig.subplots_adjust(hspace=0)
fig.suptitle('Diagrama de Momento Fletor - Trecho 3')


# Plot each graph, and manually set the y tick values
axs[0].plot(t, s1)
axs[0].set_yticks(np.arange(-250., 10., 50.))
axs[0].set_ylim(-250, 10)
axs[0].set_ylabel('Deformação\n$kN\cdot m$')

axs[1].plot(t, s2)
axs[1].set_yticks(np.arange(-250., 10., 50.))
axs[1].set_ylim(-250, 10)
axs[1].set_ylabel('Linear\n$kN\cdot m$')

axs[2].plot(t, s3)
axs[2].set_yticks(np.arange(-20., 1., 5.))
axs[2].set_ylim(-20, 1)
axs[2].set_ylabel('Quadrático\n$kN\cdot m$')

axs[3].plot(t, s4)
axs[3].set_yticks(np.arange(-3., 0., 1.))
axs[3].set_ylim(-3, 0)
axs[3].set_ylabel('Carregamento\nAdimensional')
axs[3].set_xlabel('Distância ($m$)')

plt.show()
```

```
<>:18: DeprecationWarning: invalid escape sequence \c
<>:23: DeprecationWarning: invalid escape sequence \c
<>:28: DeprecationWarning: invalid escape sequence \c
```

```
<>:18: DeprecationWarning: invalid escape sequence \c
<>:23: DeprecationWarning: invalid escape sequence \c
<>:28: DeprecationWarning: invalid escape sequence \c
<ipython-input-14-ad2969629e3c>:18: DeprecationWarning: invalid escape sequence
\c
  axs[0].set_ylabel('Deformação\n$kN\cdot m$')
<ipython-input-14-ad2969629e3c>:23: DeprecationWarning: invalid escape sequence
\c
  axs[1].set_ylabel('Linear\n$kN\cdot m$')
<ipython-input-14-ad2969629e3c>:28: DeprecationWarning: invalid escape sequence
\c
  axs[2].set_ylabel('Quadrático\n$kN\cdot m$')
```



Temos, portanto, duas componentes a integrar separadamente e somar por superposição. Uma componente linear, e outra quadrática.

Temos as seguintes equações:

3.4.1 Linear

$$\frac{1}{6}l' \left[\bar{M}_A \left(2M_A + M_B\right) + \bar{M}_B \left(2M_B + M_A\right)\right]$$

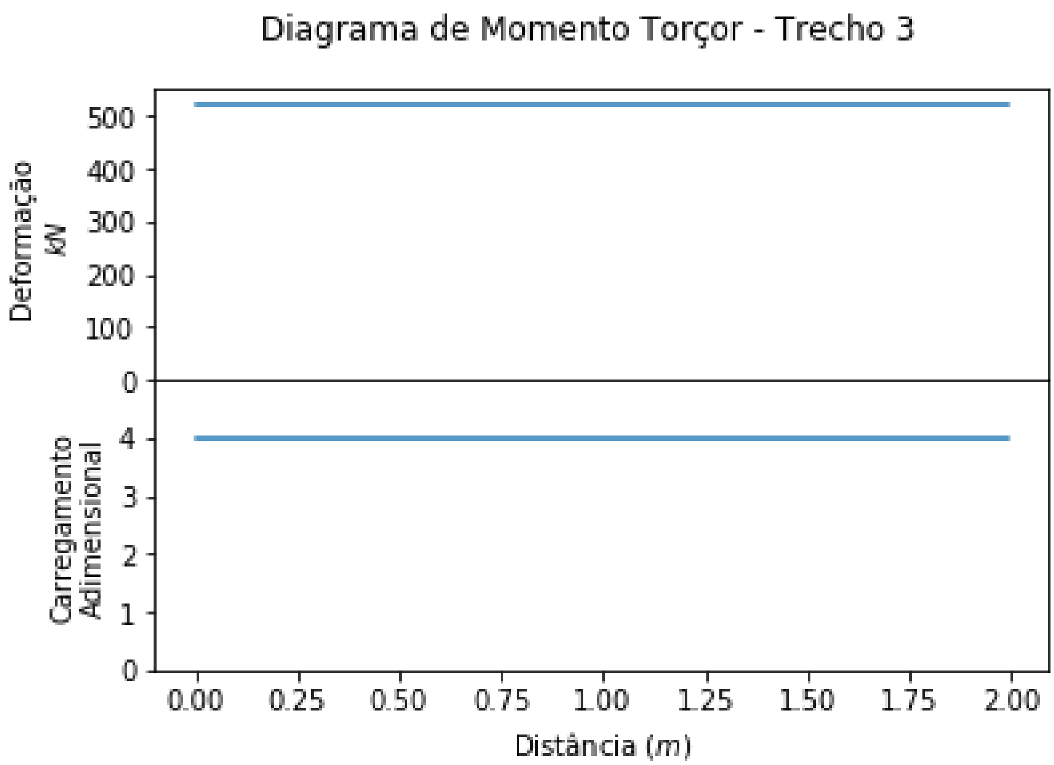
3.4.2 Parábola

$$\frac{1}{3}l'M_m\left(\bar{M}_A + \bar{M}_B\right)$$

[15]:
```
EIdelta_M3 = 2*(2*(2*100+240)+(4/3)*(2*240+100))/6 + 2*15*(2+4/3)/3
display(EIdelta_M3)
```

584.4444444444445

[16]:
```
t = np.arange(0.0, 2.0, 0.01)

s1 = 520+0*t
s2 = 4+0*t

fig, axs = plt.subplots(2, 1, sharex=True)
# Remove horizontal space between axes
fig.subplots_adjust(hspace=0)
fig.suptitle('Diagrama de Momento Torçor - Trecho 3')


# Plot each graph, and manually set the y tick values
axs[0].plot(t, s1)
axs[0].set_yticks(np.arange(0., 550., 100.))
axs[0].set_ylim(0, 550)
axs[0].set_ylabel('Deformação\n$kN$')

axs[1].plot(t, s2)
axs[1].set_yticks(np.arange(0., 5., 1.))
axs[1].set_ylim(0, 5)
axs[1].set_ylabel('Carregamento\nAdimensional')
axs[1].set_xlabel('Distância ($m$)')

plt.show()
```



[17]:
```
EIdelta_T3 = 2*2*520*4
display(EIdelta_T3)
```

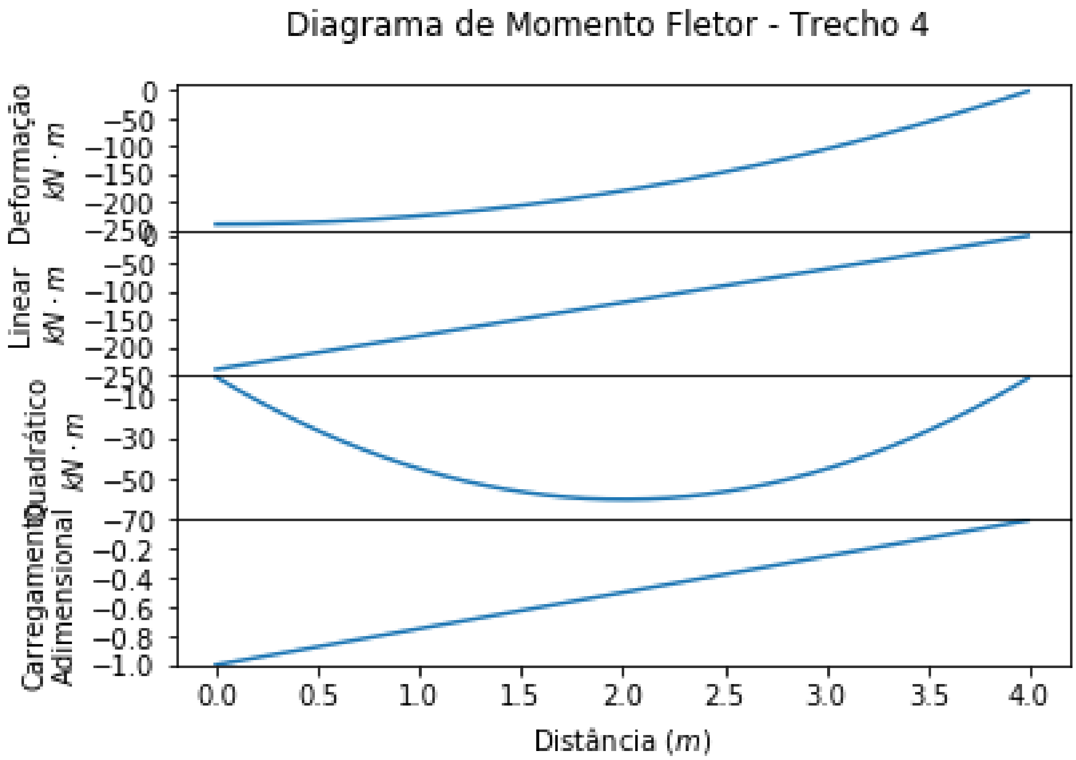
8320

3.5 Trecho 4

[18]:
```
t = np.arange(0.0, 4.0, 0.01)

s1 = (50+40*4-310+30*2+40)*t + 30*(t**2)/2 - 240
s2 = 240*t/4 - 240
s3 = 30*(t**2 - 4*t)/2
s4 = t/4 - 1

fig, axs = plt.subplots(4, 1, sharex=True)
# Remove horizontal space between axes
fig.subplots_adjust(hspace=0)
fig.suptitle('Diagrama de Momento Fletor - Trecho 4')


# Plot each graph, and manually set the y tick values
```

```
axs[0].plot(t, s1)
axs[0].set_yticks(np.arange(-250., 10., 50.))
axs[0].set_ylim(-250, 10)
axs[0].set_ylabel('Deformação\n$kN\cdot m$')

axs[1].plot(t, s2)
axs[1].set_yticks(np.arange(-250., 10., 50.))
axs[1].set_ylim(-250, 10)
axs[1].set_ylabel('Linear\n$kN\cdot m$')

axs[2].plot(t, s3)
axs[2].set_yticks(np.arange(-70., 1., 20.))
axs[2].set_ylim(-70, 1)
axs[2].set_ylabel('Quadrático\n$kN\cdot m$')

axs[3].plot(t, s4)
axs[3].set_yticks(np.arange(-1., 0., 0.2))
axs[3].set_ylim(-1, 0)
axs[3].set_ylabel('Carregamento\nAdimensional')
axs[3].set_xlabel('Distância ($m$)')

plt.show()
```

```
<>:18: DeprecationWarning: invalid escape sequence \c
<>:23: DeprecationWarning: invalid escape sequence \c
<>:28: DeprecationWarning: invalid escape sequence \c
<>:18: DeprecationWarning: invalid escape sequence \c
<>:23: DeprecationWarning: invalid escape sequence \c
<>:28: DeprecationWarning: invalid escape sequence \c
<ipython-input-18-f070308a096b>:18: DeprecationWarning: invalid escape sequence
\c
  axs[0].set_ylabel('Deformação\n$kN\cdot m$')
<ipython-input-18-f070308a096b>:23: DeprecationWarning: invalid escape sequence
\c
  axs[1].set_ylabel('Linear\n$kN\cdot m$')
<ipython-input-18-f070308a096b>:28: DeprecationWarning: invalid escape sequence
\c
  axs[2].set_ylabel('Quadrático\n$kN\cdot m$')
```



Temos, portanto, duas componentes a integrar separadamente e somar por superposição. Uma componente linear, e outra quadrática.

Temos as seguintes equações:

3.5.1 Linear

$$\frac{1}{3}l' M_A \bar{M_A}$$

3.5.2 Parábola

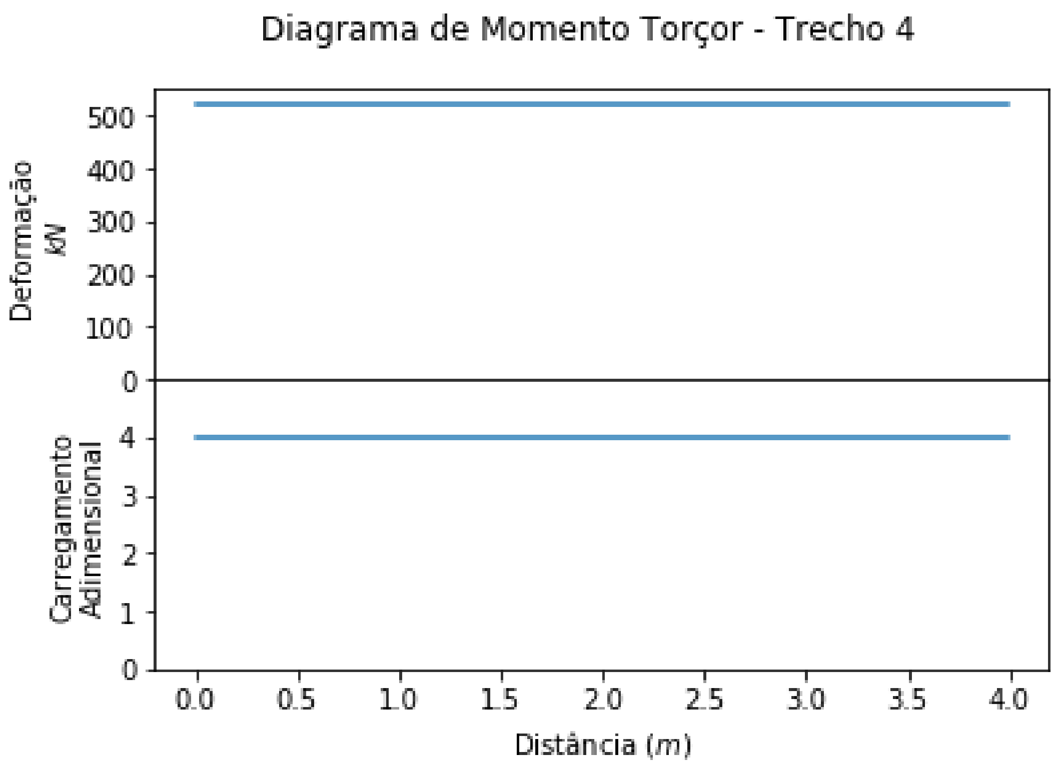
$$\frac{1}{3}l' M_m \bar{M_A}$$

[19]:
```
EIdelta_M4 = 4*240*(4/3)/3 + 4*60*(4/3)/3
display(EIdelta_M4)
```

533.3333333333334

[20]:
```
t = np.arange(0.0, 4.0, 0.01)

s1 = 520+0*t
s2 = 4+0*t
```

```python
fig, axs = plt.subplots(2, 1, sharex=True)
# Remove horizontal space between axes
fig.subplots_adjust(hspace=0)
fig.suptitle('Diagrama de Momento Torçor - Trecho 4')


# Plot each graph, and manually set the y tick values
axs[0].plot(t, s1)
axs[0].set_yticks(np.arange(0., 550., 100.))
axs[0].set_ylim(0, 550)
axs[0].set_ylabel('Deformação\n$kN$')

axs[1].plot(t, s2)
axs[1].set_yticks(np.arange(0., 5., 1.))
axs[1].set_ylim(0, 5)
axs[1].set_ylabel('Carregamento\nAdimensional')
axs[1].set_xlabel('Distância ($m$)')

plt.show()
```



[21]:
```
EIdelta_T4 = 2*4*520*4
display(EIdelta_T4)
```

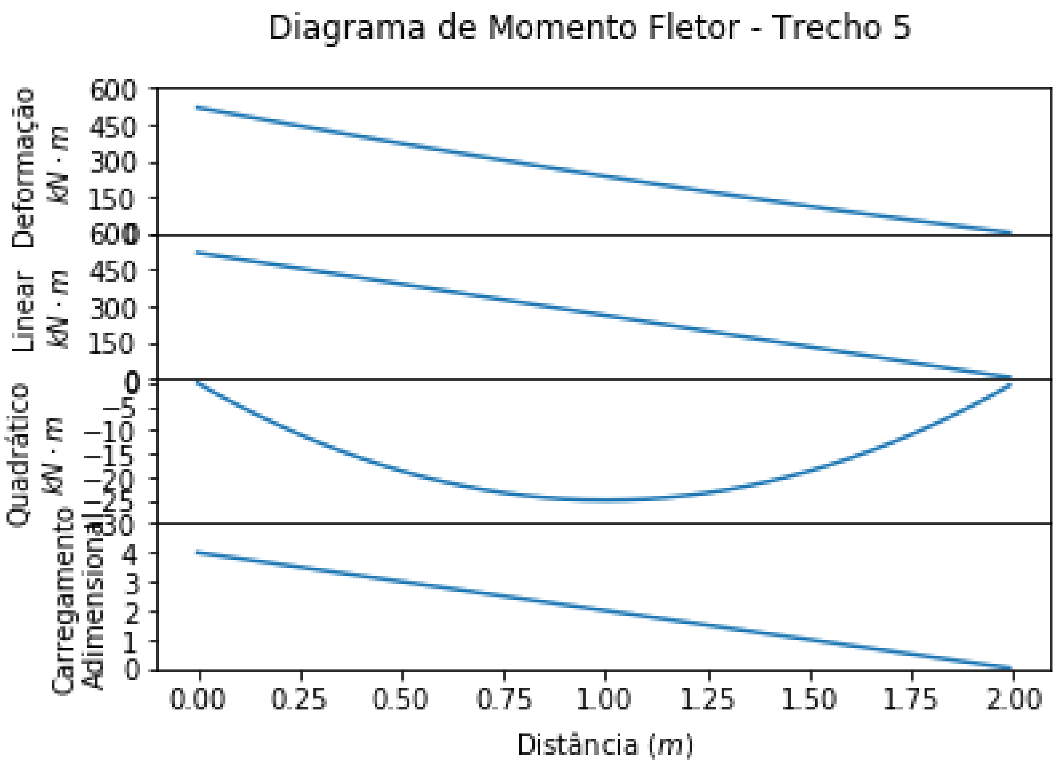
16640

3.6 Trecho 5

[22]:
```
t = np.arange(0.0, 2.0, 0.01)

s1 = -310*t + 50*(t**2)/2 + 520
s2 = 520 - 520*t/2
s3 = 50*(t**2 - 2*t)/2
s4 = 4 - 2*t

fig, axs = plt.subplots(4, 1, sharex=True)
# Remove horizontal space between axes
fig.subplots_adjust(hspace=0)
fig.suptitle('Diagrama de Momento Fletor - Trecho 5')


# Plot each graph, and manually set the y tick values
axs[0].plot(t, s1)
axs[0].set_yticks(np.arange(0., 601., 150.))
axs[0].set_ylim(0, 601)
axs[0].set_ylabel('Deformação\n$kN\cdot m$')

axs[1].plot(t, s2)
axs[1].set_yticks(np.arange(0., 601., 150.))
axs[1].set_ylim(0, 601)
axs[1].set_ylabel('Linear\n$kN\cdot m$')

axs[2].plot(t, s3)
axs[2].set_yticks(np.arange(-30., 1., 5.))
axs[2].set_ylim(-30, 1)
axs[2].set_ylabel('Quadrático\n$kN\cdot m$')

axs[3].plot(t, s4)
axs[3].set_yticks(np.arange(0., 5.0, 1.))
axs[3].set_ylim(0, 5)
axs[3].set_ylabel('Carregamento\nAdimensional')
axs[3].set_xlabel('Distância ($m$)')

plt.show()
```

```
<>:18: DeprecationWarning: invalid escape sequence \c
<>:23: DeprecationWarning: invalid escape sequence \c
<>:28: DeprecationWarning: invalid escape sequence \c
```

```
<>:18: DeprecationWarning: invalid escape sequence \c
<>:23: DeprecationWarning: invalid escape sequence \c
<>:28: DeprecationWarning: invalid escape sequence \c
<ipython-input-22-97bf54f9cc90>:18: DeprecationWarning: invalid escape sequence
\c
  axs[0].set_ylabel('Deformação\n$kN\cdot m$')
<ipython-input-22-97bf54f9cc90>:23: DeprecationWarning: invalid escape sequence
\c
  axs[1].set_ylabel('Linear\n$kN\cdot m$')
<ipython-input-22-97bf54f9cc90>:28: DeprecationWarning: invalid escape sequence
\c
  axs[2].set_ylabel('Quadrático\n$kN\cdot m$')
```



Temos, portanto, duas componentes a integrar separadamente e somar por superposição. Uma componente linear, e outra quadrática.

Temos as seguintes equações:

3.6.1 Linear

$$\frac{1}{3}l'M_A\bar{M}_A$$

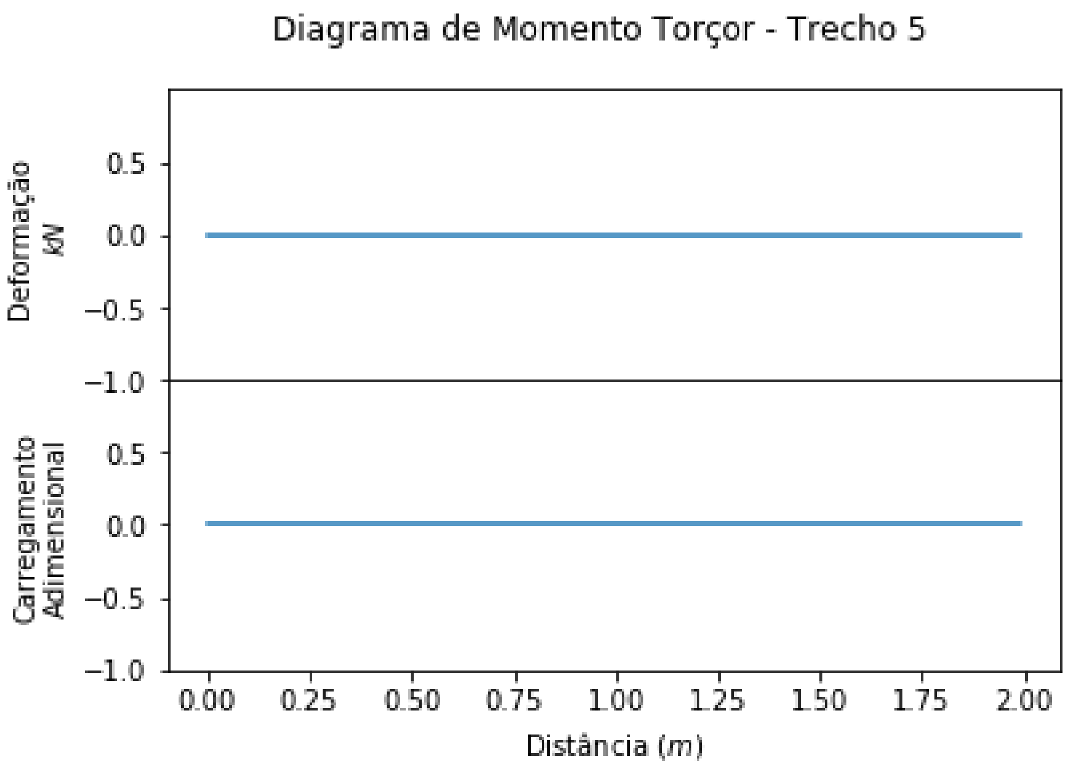
3.6.2 Parábola

$$\frac{1}{3}l' M_m \bar{M}_A$$

[23]:
```
EIdelta_M5 = 2*520*4/3 - 2*25*4/3
display(EIdelta_M5)
```

1320.0

[24]:
```
t = np.arange(0.0, 2.0, 0.01)

s1 = 0*t
s2 = 0*t

fig, axs = plt.subplots(2, 1, sharex=True)
# Remove horizontal space between axes
fig.subplots_adjust(hspace=0)
fig.suptitle('Diagrama de Momento Torçor - Trecho 5')


# Plot each graph, and manually set the y tick values
axs[0].plot(t, s1)
axs[0].set_yticks(np.arange(-1., 1., 0.5))
axs[0].set_ylim(-1, 1)
axs[0].set_ylabel('Deformação\n$kN$')

axs[1].plot(t, s2)
axs[1].set_yticks(np.arange(-1., 1., 0.5))
axs[1].set_ylim(-1, 1)
axs[1].set_ylabel('Carregamento\nAdimensional')
axs[1].set_xlabel('Distância ($m$)')

plt.show()
```



[25]:
```
EIdelta_T5 = 0
display(EIdelta_T5)
```

$$0$$

3.7 Deslocamento no Ponto D

Deslocamento causado por momento fletor:

[26]:
```
delta_M = (EIdelta_M1 + EIdelta_M2 + EIdelta_M3 + EIdelta_M4 + EIdelta_M5)/1e5
display(delta_M)
```

$$0.0491777777777777786$$

Deslocamento causado por momento torçor:

[27]:
```
delta_T = (EIdelta_T1 + EIdelta_T2 + EIdelta_T3 + EIdelta_T4 + EIdelta_T5)/1e5
display(delta_T)
```

$$0.2656$$

[28]:
```
delta_T/4
```

[28]:

$$0.0664$$

Deslocamento total:

[29]:
```
delta_C = delta_M + delta_T
display(delta_C)
```

$$0.3147777777777777$$

Ou seja,

$$\delta_C = 31,48 \ \ cm \downarrow$$