

Układy belkowe i ramowe

Konwencja zwrotów dodatnich – belki i ramy

W programach MES, konwencje zwrotów dodatnich mogą się różnić, dlatego dobrze jest sprawdzić w dokumentacji, jak należy interpretować poszczególne wielkości.

Dodatnie zwroty przemieszczeń, obrotów i reakcji w węźle w programie OpenSees.

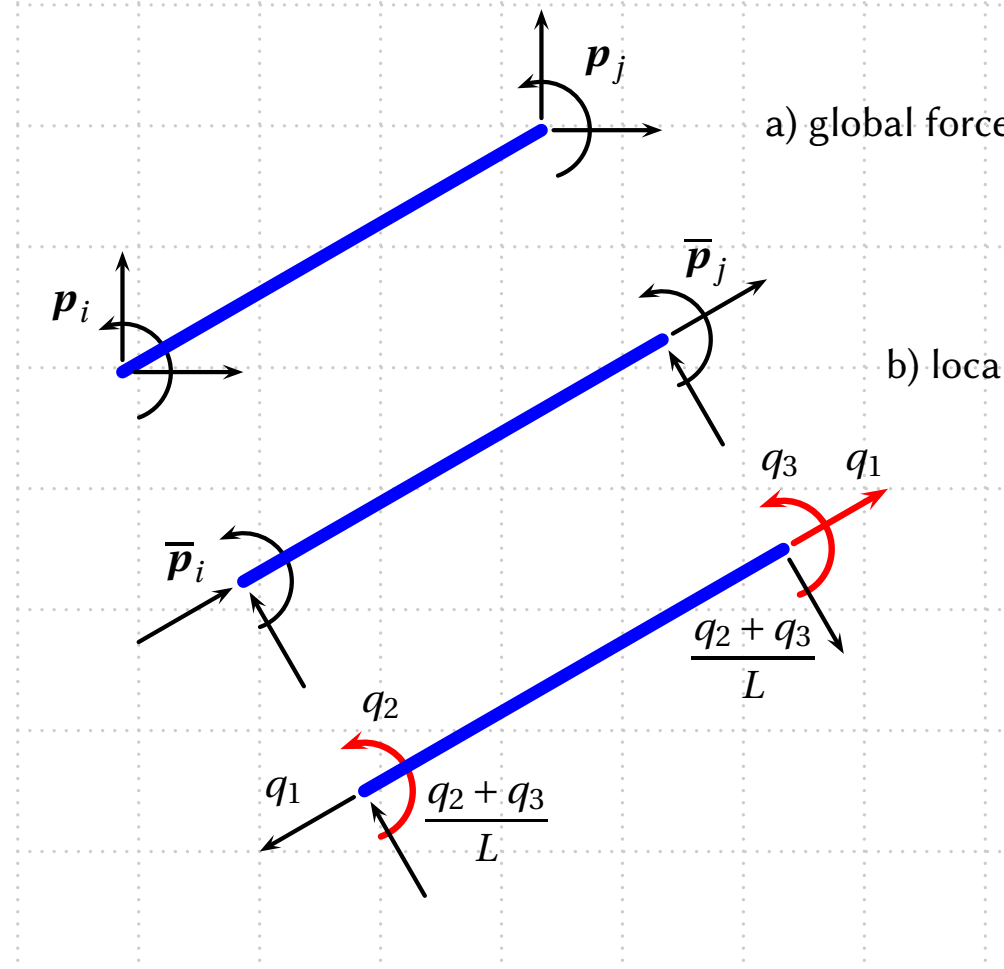
- globalne stopnie swobody - przemieszczenia i obroty
 - ◆ u_x – w prawo; u_y – do góry
 - ◆ r_z – przeciwnie do ruchu wskazówek zegara
- lokalne stopnie swobody - przemieszczenia i obroty
 - ◆ \bar{u}_x – w prawo (wzdłuż pręta)
 - ◆ \bar{u}_y – do góry (prostopadle do pręta)
 - ◆ $\bar{r}_z = r_z$ – przeciwnie do ruchu wskazówek zegara
- reakcje w węźle w ukł. globalnym
 - ◆ R_x – w prawo; R_y – do góry
 - ◆ M_z – przeciwnie do ruchu wskazówek zegara

Przykłady dla belki ukośnej obustronnie zamocowanej i wolnopodpartej poziomej pokazano poniżej.

Konwencja zwrotów dodatnich el. ramowego

Przyjęto, że dodatnie wartości sił przekrojowych w programie OpenSees mają następujące zwroty dla lewego (i) i prawego (j) końca el.

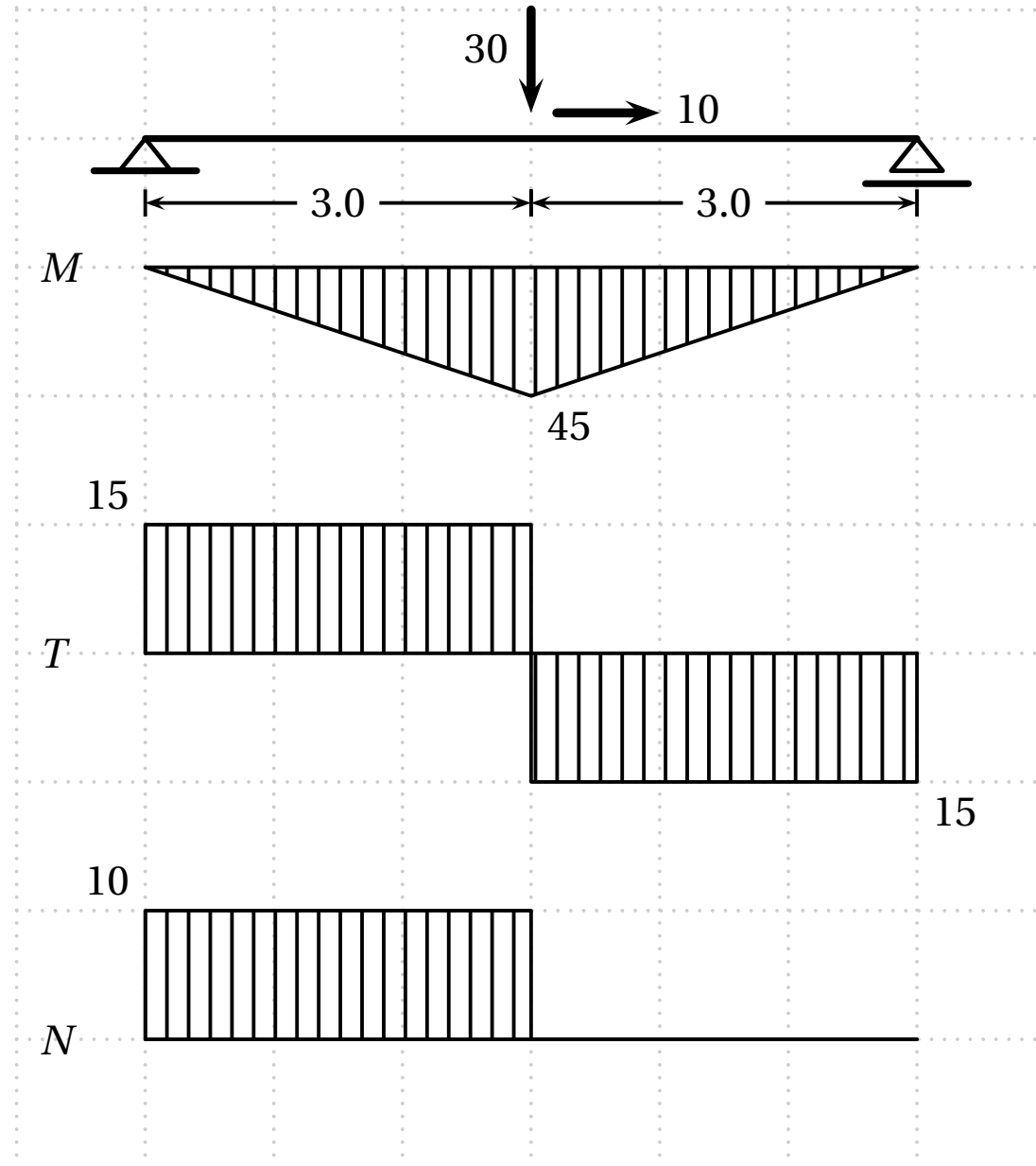
- siły globalne
 - ◆ F_{xi} [p_1], F_{xj} [p_4] – w prawo
 - ◆ F_{yi} [p_2], F_{yj} [p_5] – do góry
 - ◆ M_{zi} [p_3], M_{zj} [p_6] – przeciwnie do ruchu wskazówek zegara
- siły lokalne
 - ◆ N_i [\bar{p}_1], N_j [\bar{p}_4] – w prawo
 - ◆ T_i [\bar{p}_2], T_j [\bar{p}_5] – do góry
 - ◆ M_{zi} [\bar{p}_3], M_{zj} [\bar{p}_6] – przeciwnie do ruchu wskazówek zegara
- siły w układzie podstawowym (3 niezależne składowe)
 - ◆ N [q_1] – w prawo
 - ◆ M_{zi} [q_2], M_{zj} [q_3] – przeciwnie do ruchu wskazówek zegara



W nawiasach [...] podano alternatywne oznaczenia wielkości.

Konwencja znaków - belka wolnopodparta

ssb-test-konwencja.tcl



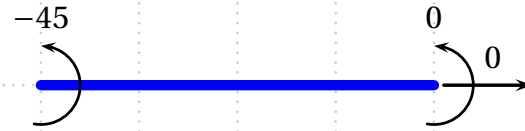
basicForces

bF-el1: 10 0 45



ssb-test-konwencja.tcl

bF-el2: 0 -45 0



localForces = globalForces

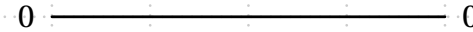
lF-el1: -10 15 0; 10 -15 45



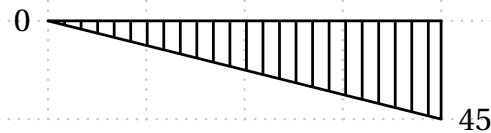
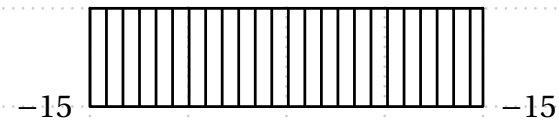
lF-el2: 0 -15 -45; 0 15 0



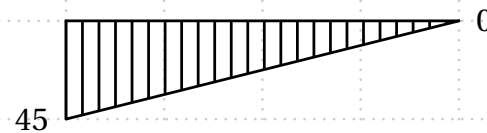
N [kN]



T [kN]



M [kNm]

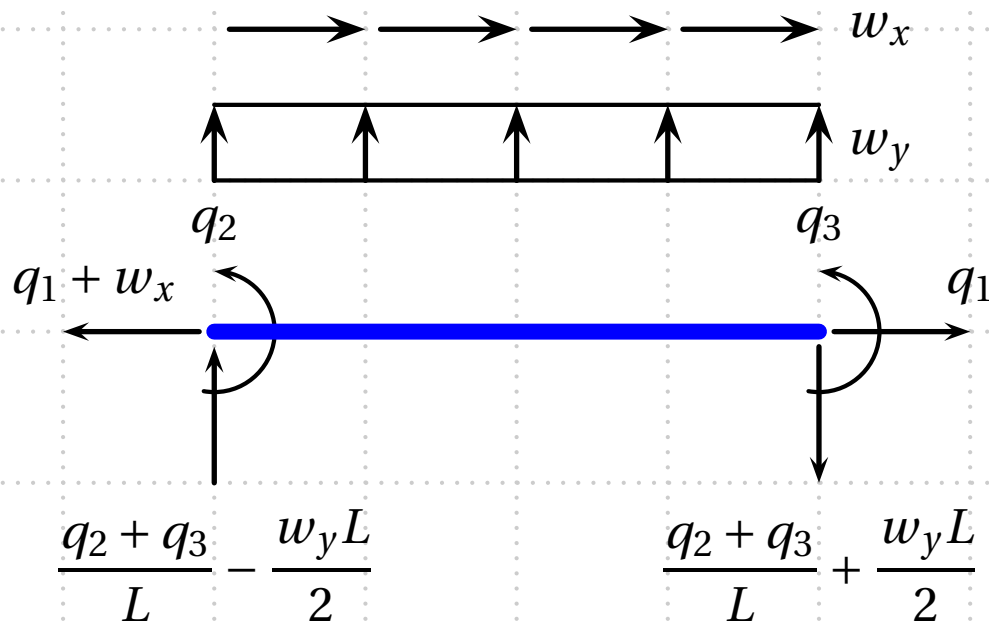


Obciążenie równomierne elementu

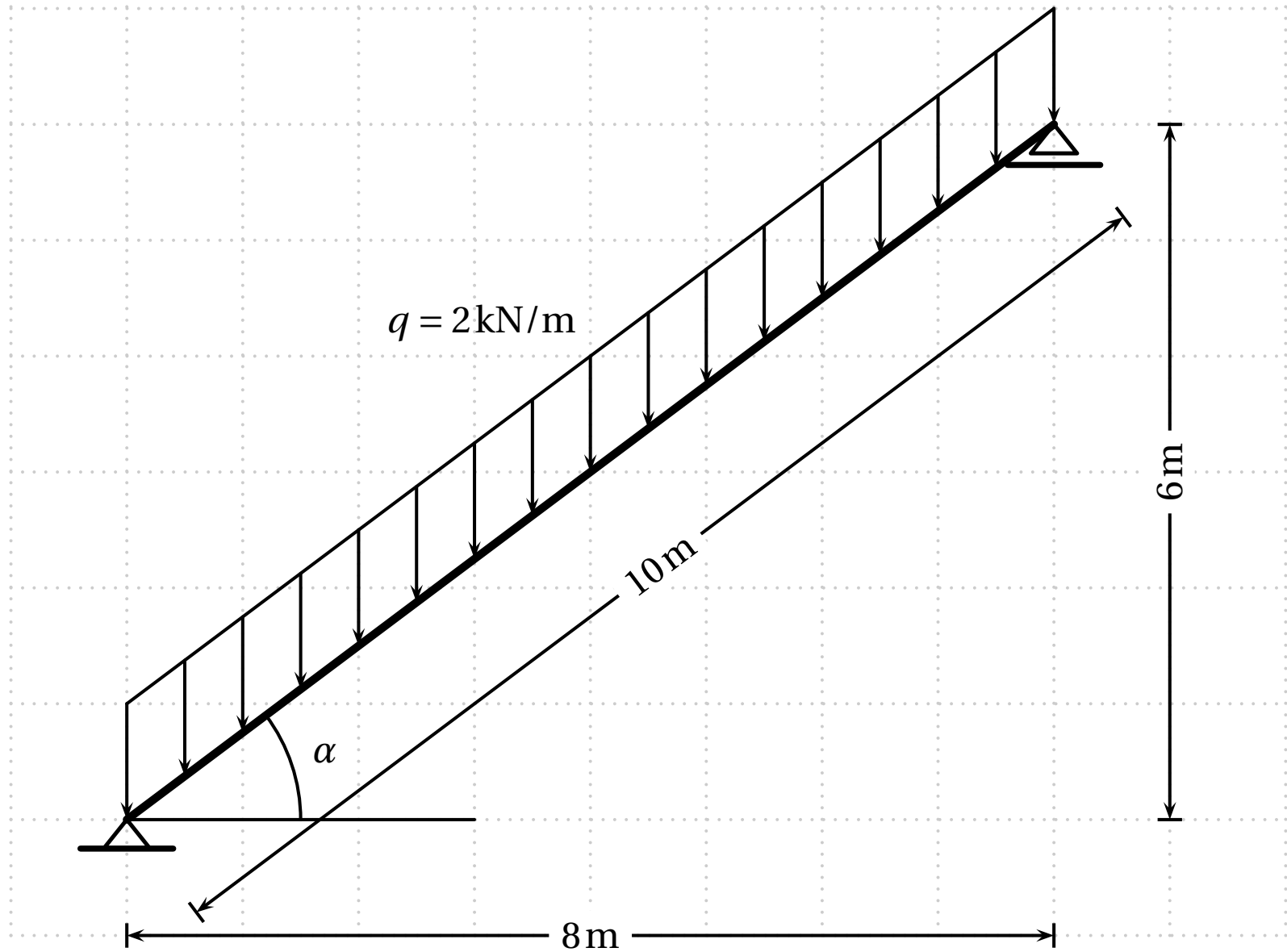
Komenda OpenSeesPy:

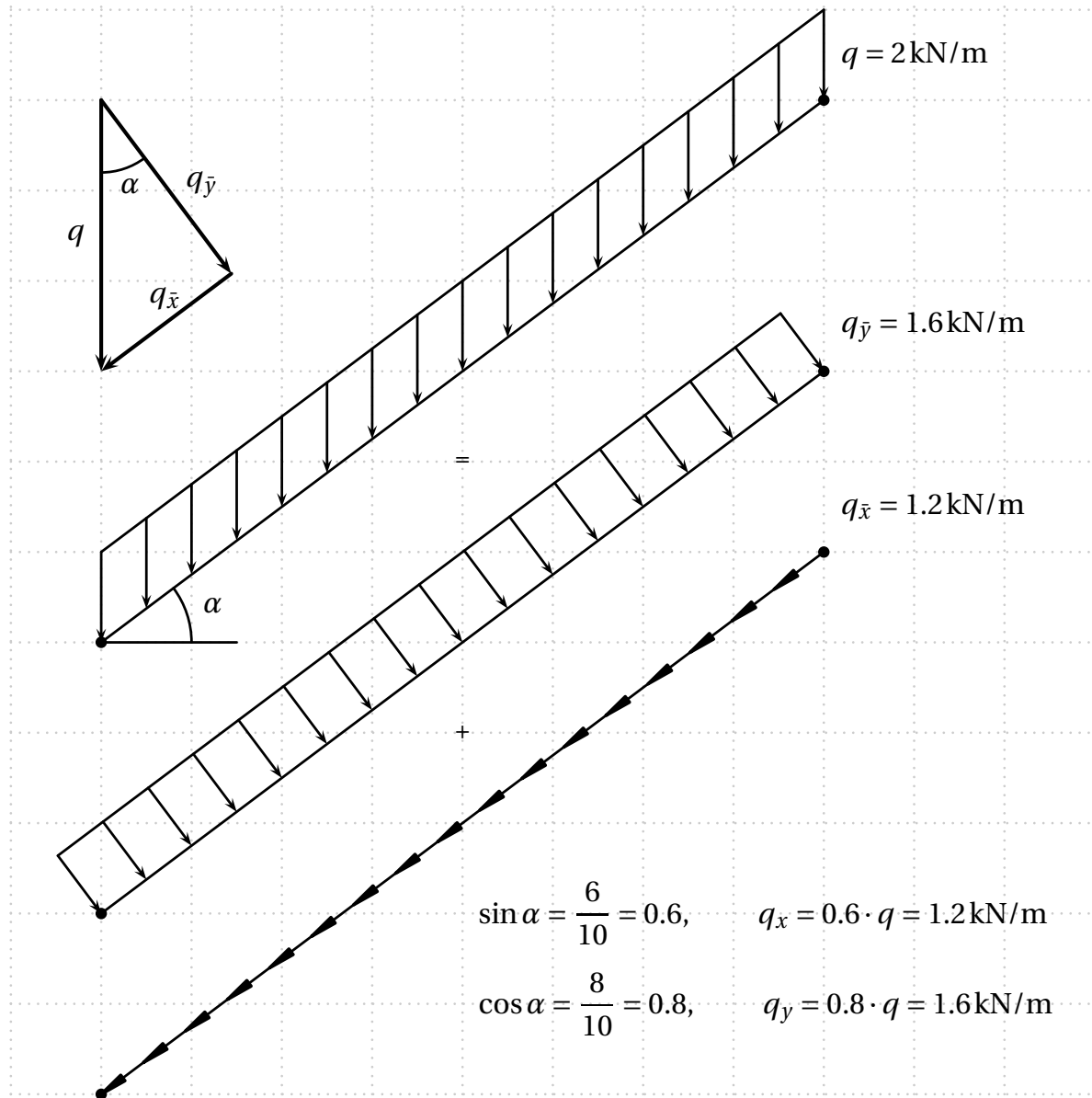
```
eleLoad('-ele', *eleTags, '-type', '-beamUniform', wy, wx=0.0)
```

wymaga podania składowych obciążenia równomiernego prostopadłego do belki i wzdłuż belki.



Przykład obc. rozłożonego na belce ukośnej





Plik wejściowy do OpenSeesPy

```
# import openseespy.opensees as ops
import opensees as ops
import ops_visu as opsv
import matplotlib.pyplot as plt

ops.wipe()
ops.model('basic', '-ndm', 2, '-ndf', 3)

Lx, Ly = 4., 3.
b, h, E = 0.2, 0.3, 30.e6
A, Iz = b*h, b*h*h*h/12.

ops.node(1, 0., 0.)
ops.node(2, Lx, Ly)
ops.node(3, 2*Lx, 2*Ly)

ops.fix(1, 1, 1, 0)
ops.fix(3, 0, 1, 0)

ops.geomTransf('Linear', 1)
ops.element('elasticBeamColumn', 1, 1, 2, A, E, Iz
    ↪ , 1)
ops.element('elasticBeamColumn', 2, 2, 3, A, E, Iz
    ↪ , 1)

ops.timeSeries('Constant', 1)
ops.pattern('Plain', 1, 1)
```

```
# downward element load -2.0 kN/m
Wy, Wx = -1.6, -1.2
ops.eleLoad('-ele', 1, '-type', '-
    ↪ beamUniform', Wy, Wx)
ops.eleLoad('-ele', 2, '-type', '-
    ↪ beamUniform', Wy, Wx)

ops.recorder('Node', '-file', f'out/uAll.
    ↪ out', '-node', 1, 2, 3,
        '-dof', 1, 2, 3, 'disp')
ops.recorder('Node', '-file', f'out/rAll.
    ↪ out', '-node', 1, 2, 3,
        '-dof', 1, 2, 3, 'reaction')

ops.recorder('Element', '-file', f'out/
    ↪ gF_all.out', '-ele', 1, 2,
        'globalForce')
ops.recorder('Element', '-file', f'out/
    ↪ lF_all.out', '-ele', 1, 2,
        'localForce')

ops.analysis('Static')
ops.analyze(1)

ops.reactions()

ops.printModel()

opsv.plot_model()
```

Plik wejściowy do OpenSeesPy c.d

```
plt.figure()
opsv.plot_defo()

sfacN, sfacV, sfacM = 0.5, 0.1, 0.1

plt.figure()
minY, maxY = opsv.section_force_diagram_2d('M',
    ↪ sfacM)
plt.title(f'Momenty_zginajace, _max_={maxY:.2f}, _
    ↪ min_={minY:.2f}')

plt.figure()
minY, maxY = opsv.section_force_diagram_2d('T',
    ↪ sfacV)
plt.title(f'Sily_tnace, _max_={maxY:.2f}, _min_={
    ↪ minY:.2f}')

plt.figure()
minY, maxY = opsv.section_force_diagram_2d('N',
    ↪ sfacN)
plt.title(f'Sily_osiowe, _max_={maxY:.2f}, _min_={
    ↪ minY:.2f}')

plt.show()

exit()

# plt.savefig('kurs_belka_ukosna_1_model.png')
# plt.savefig('kurs_belka_ukosna_1_defo.png')
# plt.savefig('kurs_belka_ukosna_1_M.png')
# plt.savefig('kurs_belka_ukosna_1_T.png')
```

```
# plt.savefig('kurs_belka_ukosna_1_N.png')
```

Wyniki z programu OpenSeesPy

Po uruchomieniu pliku .py, najważniejsze dane wyjściowe z `ops.printModel()` są następujące

```
Node: 1
  Coordinates : 0 0
  Disps: 0 0 -0.00493827
  reaction: -3.12372e-13 10 1.77636e-15

Node: 2
  Coordinates : 4 3
  Disps: 0.00925259 -0.0123507 -3.94567e-20
  reaction: 6.86509e-13 1.06233e-12 7.10543e-15

Node: 3
  Coordinates : 8 6
  Disps: -3.19745e-19 0 0.00493827
  reaction: -3.73124e-13 10 9.76996e-15

ElasticBeam2d: 1
  Connected Nodes: 1 2
  End 1 Forces (P V M): 6 8 1.77636e-15
  End 2 Forces (P V M): 5.54223e-13 2.17604e-13 20

ElasticBeam2d: 2
  Connected Nodes: 2 3
  End 1 Forces (P V M): 6.32383e-13 2.21156e-13 -20
  End 2 Forces (P V M): 6 8 9.76996e-15
```

Pliki z wynikami

plik gF_all.out

```
0 10 0 0 0 20 0 0 -20 0 10 0
```

plik lF_all.out

```
6 8 0 0 0 20 0 0 -20 6 8 0
```

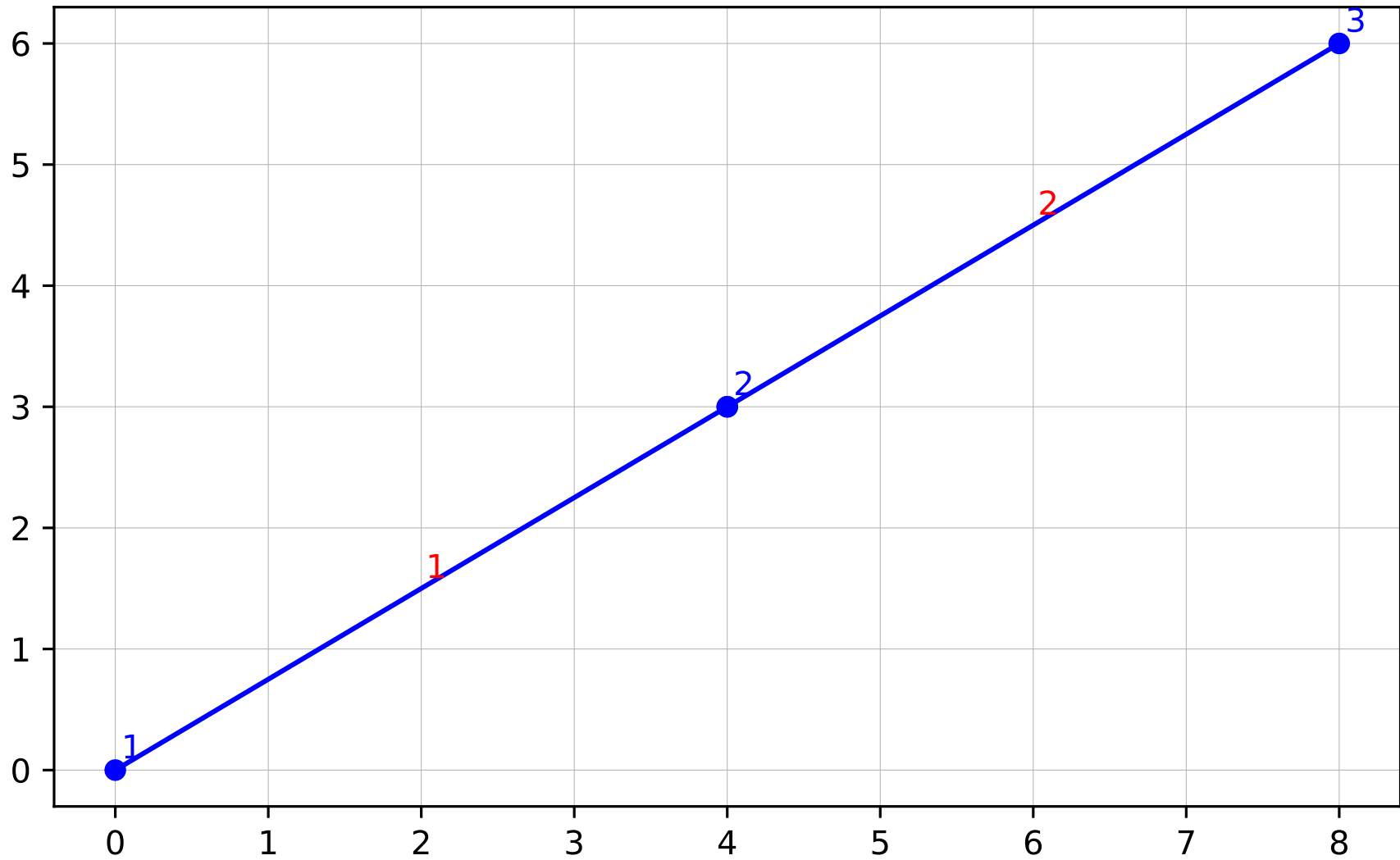
plik rAll.out

```
0 10 0 0 0 0 0 10 0
```

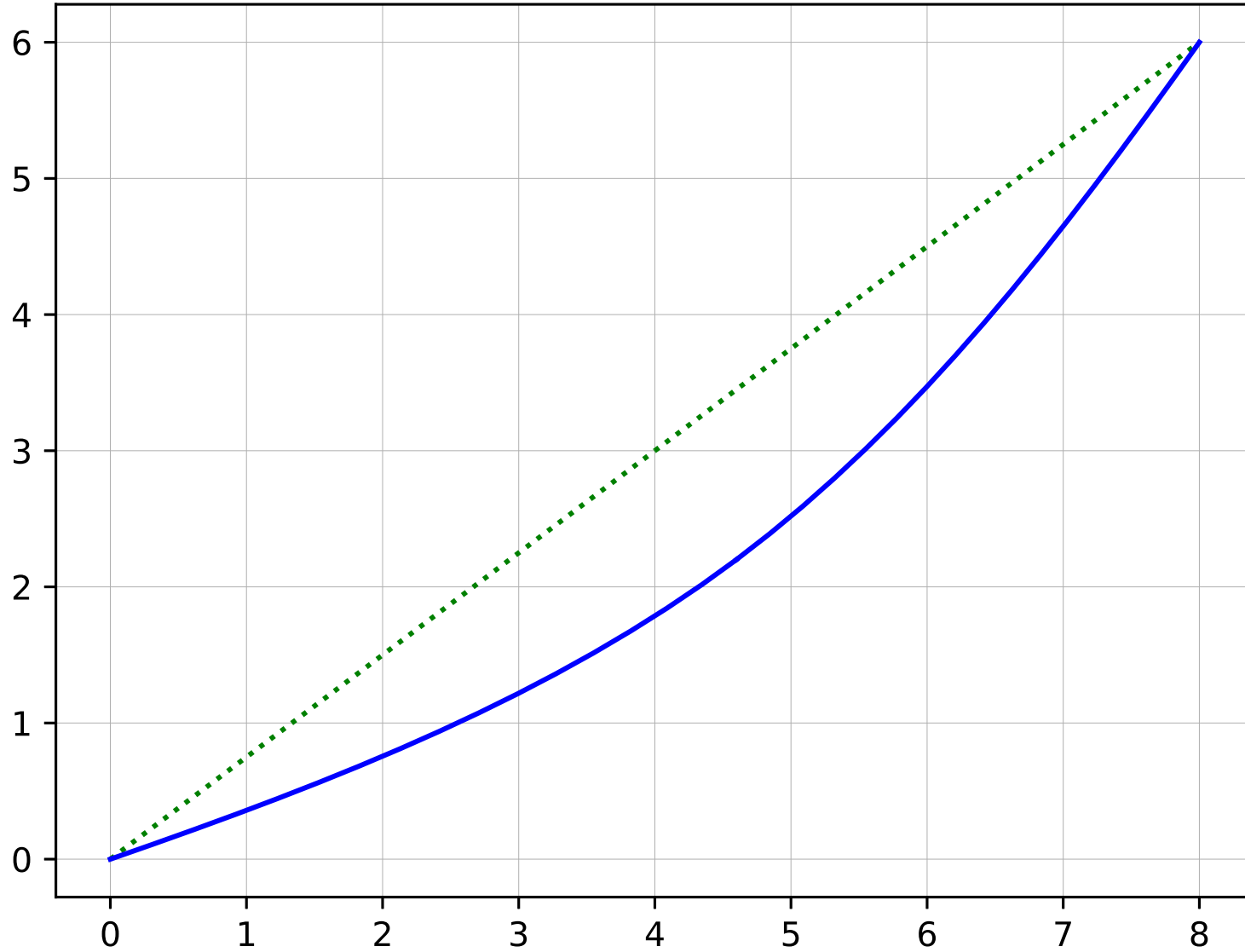
plik uAll.out

```
0 0 -0.004938 0.00925 -0.01235 0 0 0 0.004938
```

Model

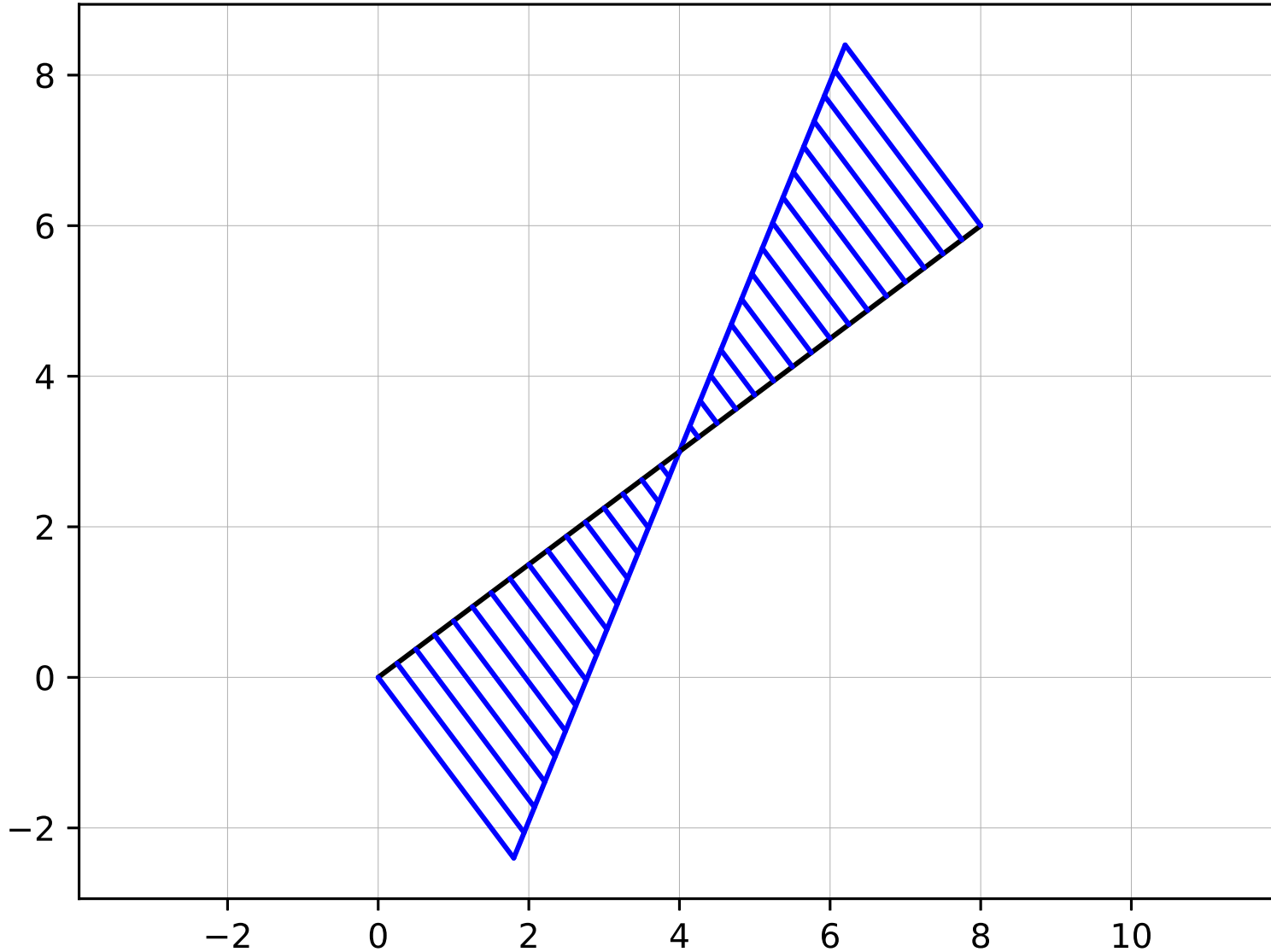


Belka odkształcona



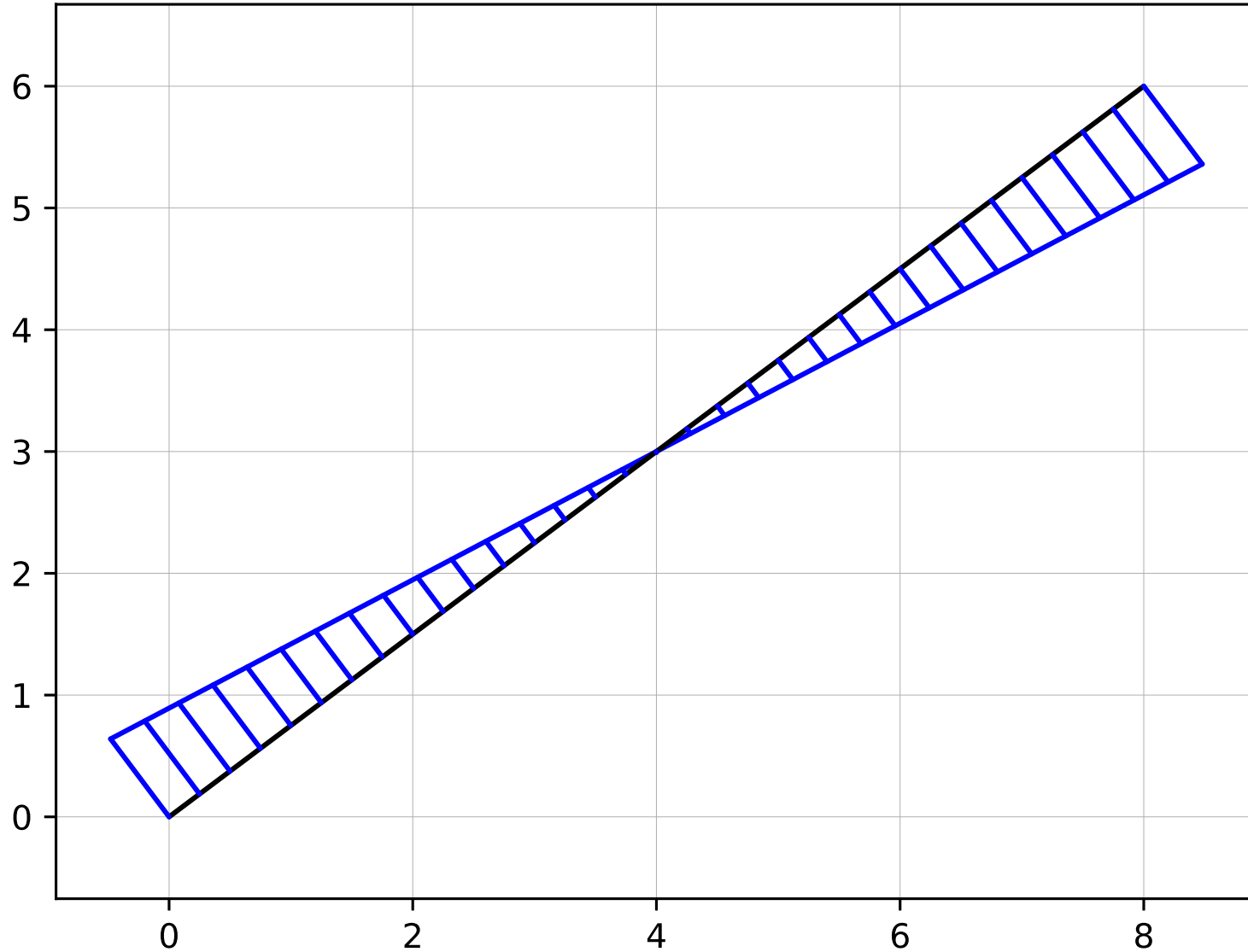
Siły osiowe

Siły osiowe, max = 6.00, min = -6.00



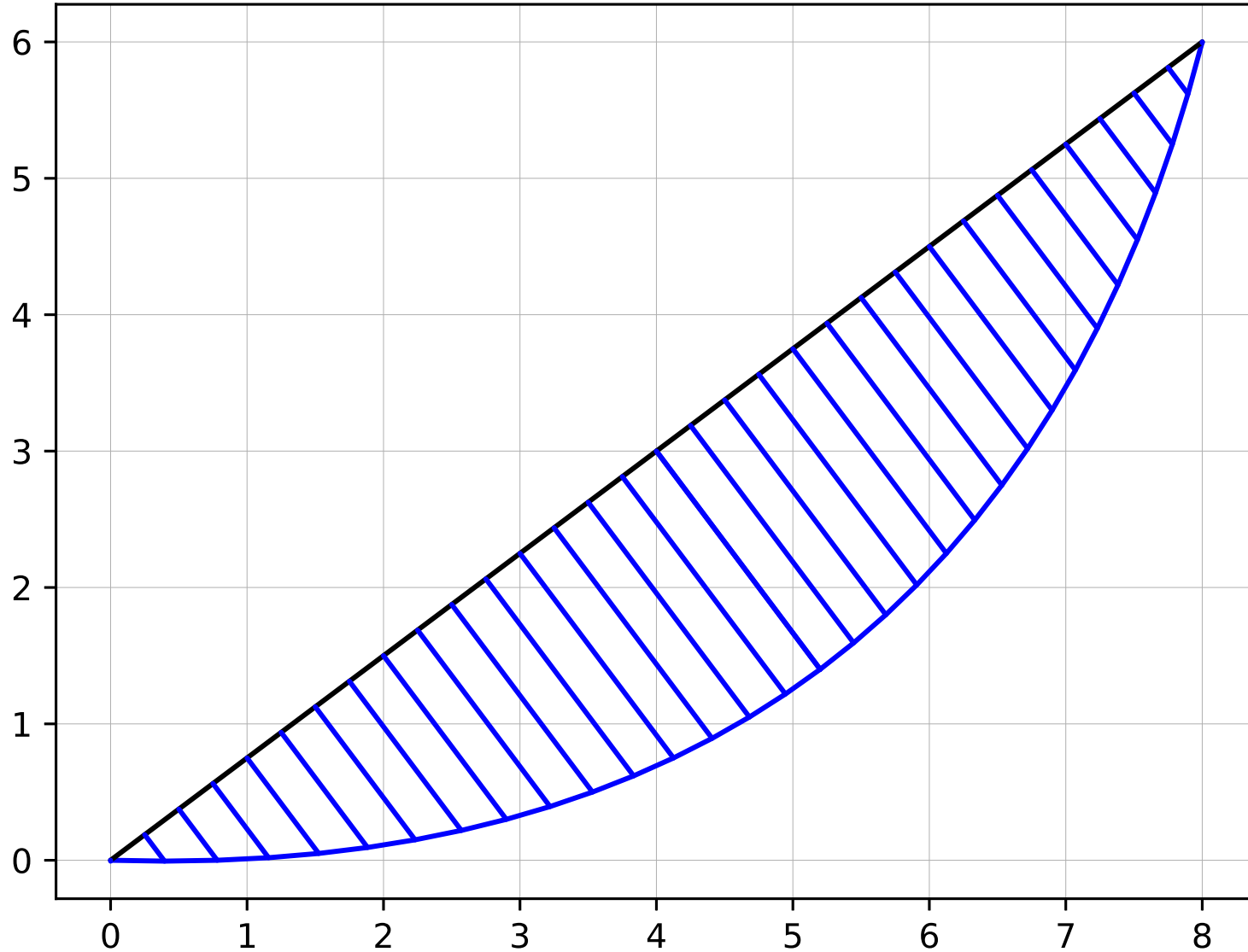
Siły tnące

Siły tnące, max = 8.00, min = -8.00

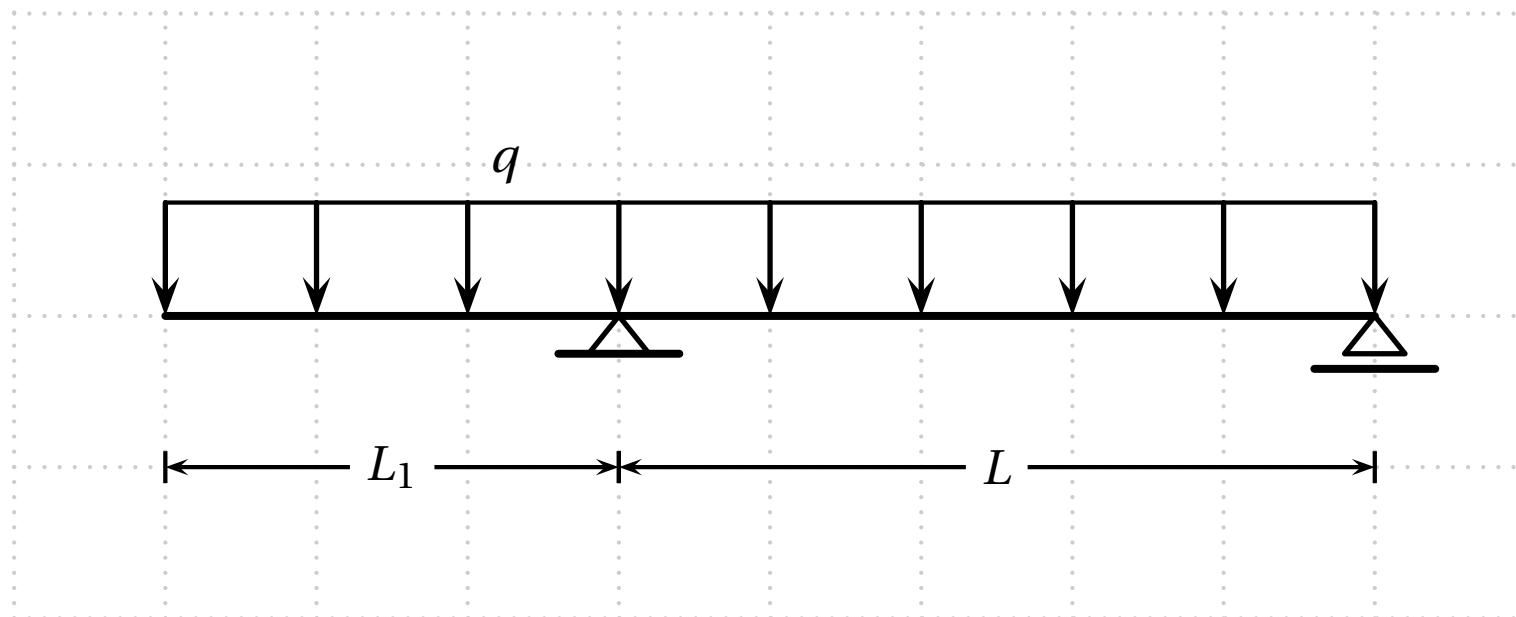


Momenty zginające

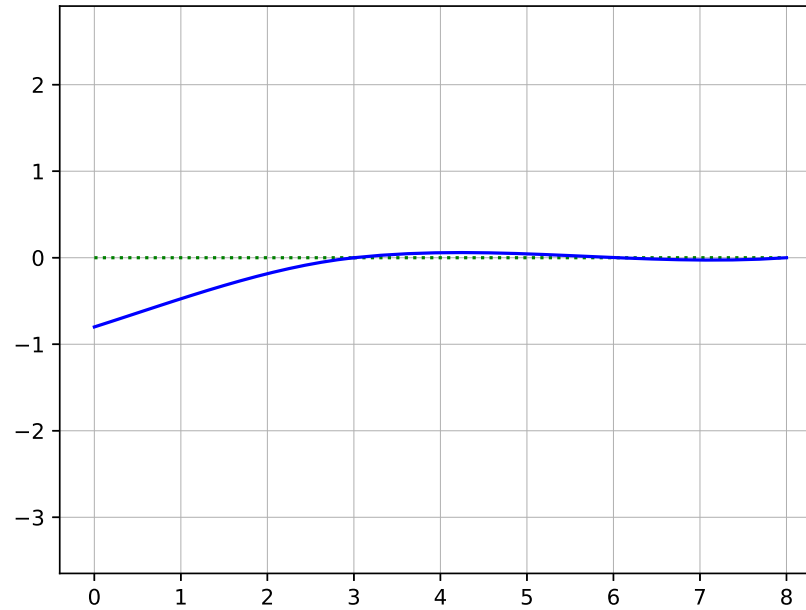
Momenty zginajace, max = 20.00, min = -0.00



Dla schematu belki obciążonej równomiernie wyznaczyć: reakcje, siły przekrojowe (momenty zginające i siły tnące) i pokazać belkę odkształconą i wykresy sił przekrojowych. Dane: długość wspornika $L_1 = 3\text{ m}$, główne przęsło $L = 5\text{ m}$, obciążenie równomiernie rozłożone $q = 4\text{ kN/m}$, $E = 25\text{ GPa}$, przekrój belki prostokątny $b = 0.2\text{ m}$ i $h = 0.3\text{ m}$.



Belka ze wspornikiem – deformacja



Belka ze wspornikiem – wykres sił przekrojowych

