

questao01

February 15, 2022

Questão 01 - Lista 3

```
[50]: import sympy as sp
import numpy as np
from sympy import *
from utils import symdisp, symplot
```

```
[51]: #valor dos três ultimos digitos da matricula somado mais 1
m = input('Digite os três últimos dígitos de sua matrícula:')
m1 = round(int(m)/100)
m2 = round((int(m)%100)/10)
m3 = round(int(m)%10)
```

Digite os três últimos dígitos de sua matrícula:814

```
[52]: x = m1+1
y = m2+1
z = m3+1
#Valores das indutâncias L1, L2 e L3
L1 = 4*(10**-3)
L2 = 1*(10**-3)
L3 = 5*(10**-3)
symdisp('L_1 = ', L1*(10**3), 'mH')
symdisp('L_2 = ', L2*(10**3), 'mH')
symdisp('L_3 = ', L3*(10**3), 'mH')
```

$$L_1 = 4.0 \text{ mH}$$

$$L_2 = 1.0 \text{ mH}$$

$$L_3 = 5.0 \text{ mH}$$

```
[53]: #variável simbólica
t = sp.symbols('t', real=True)
v1 = -0.05*x*exp(-x*t)
v2 = -0.02*y*exp(-y*t)*cos(20*t)
v3 = 0.01*z*exp(-z*t)
symdisp('v_1 = ', v1, 'V')
symdisp('v_2 = ', v2, 'V')
symdisp('v_3 = ', v3, 'V')
```

$$v_1 = -0.45e^{-9t} \text{ V}$$

$$v_2 = -0.04e^{-2t} \cos(20t) \text{ V}$$

$$v_3 = 0.05e^{-5t} \text{ V}$$

```
[54]: #A partir do método da superposição, foi-se obtido as expressões para as
      → tensões em cada indutor
      # Tensão no indutor = Contribuição de v1 + Contribuição de v2 + Contribuição de
      → v3
      vL1 = -v1 + (L1/(L3+L2))*v2 - v3
      vL2 = 0 + v2 - v3
      vL3 = -v1 - (L3/(L1+L3))*v2
      symdisp('v_{L1} = ', vL1.simplify(), 'V')
      symdisp('v_{L2} = ', vL2.simplify(), 'V')
      symdisp('v_{L3} = ', vL3.simplify(), 'V')
```

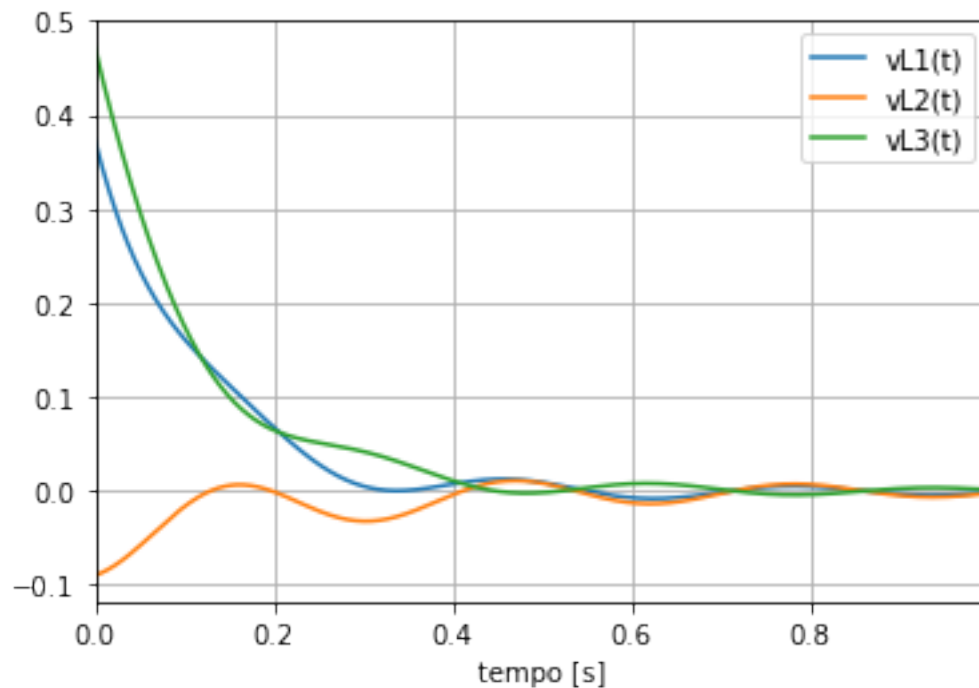
$$v_{L1} = \left(-0.0266666666666667e^{7t} \cos(20t) - 0.05e^{4t} + 0.45 \right) e^{-9t} \text{ V}$$

$$v_{L2} = - \left(0.04e^{3t} \cos(20t) + 0.05 \right) e^{-5t} \text{ V}$$

$$v_{L3} = \left(0.0222222222222222e^{7t} \cos(20t) + 0.45 \right) e^{-9t} \text{ V}$$

```
[55]: #plotagem das tensões
      intervalo = np.arange(0,1,0.001)
      symplot(t,[vL1,vL2,vL3],intervalo,['vL1(t)', 'vL2(t)', 'vL3(t)'])
```

[55]:



```
[56]: #Determinando as correntes em cada indutor
int_vL1 = sp.integrate(vL1,(t,0,t))
int_vL2 = sp.integrate(vL2,(t,0,t))
int_vL3 = sp.integrate(vL3,(t,0,t))
```

Letra A

```
[57]: print('Letra A: Correntes nos indutores L1, L2 e L3')
iL1 = (L1**-1)*int_vL1; symdisp('i_{L1} (t) = ', iL1.simplify(), 'A')
iL2 = (L2**-1)*int_vL2; symdisp('i_{L2} (t)= ', iL2.simplify(), 'A')
iL3 = (L3**-1)*int_vL3; symdisp('i_{L3} (t)= ', iL3.simplify(), 'A')
```

Letra A: Correntes nos indutores L1, L2 e L3

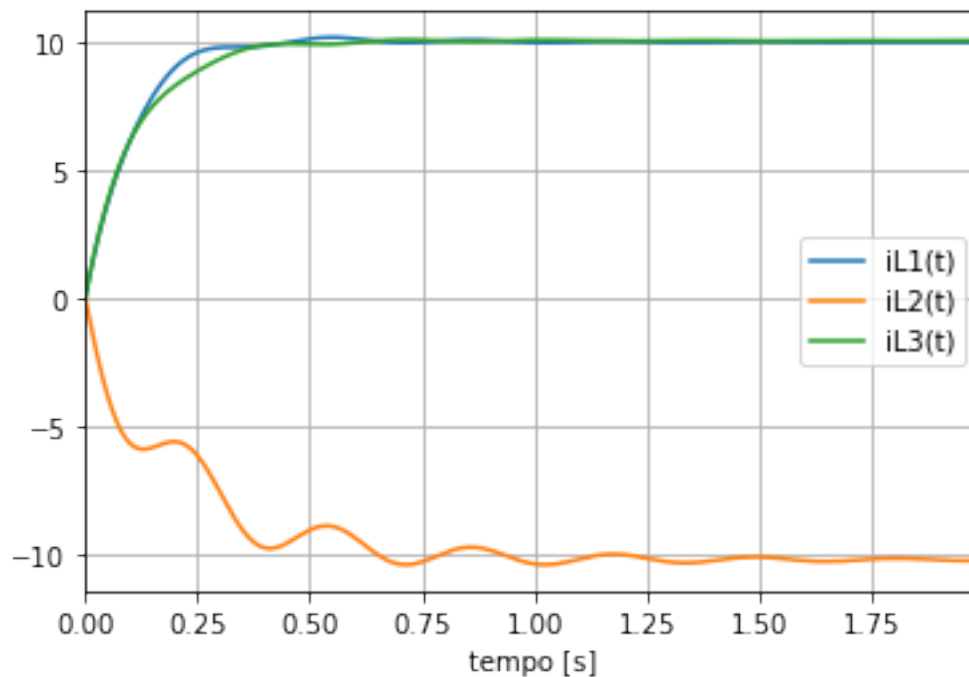
$$i_{L1}(t) = ((-0.33003300330033 \sin(20t) + 0.033003300330033 \cos(20t)) e^{7t} + 9.96699669966996 e^{9t} + 2.5 e^{4t} - 12.5)$$

$$i_{L2}(t) = ((-1.98019801980198 \sin(20t) + 0.198019801980198 \cos(20t)) e^{3t} - 10.1980198019802 e^{5t} + 10.0) e^{-5t} \text{ A}$$

$$i_{L3}(t) = ((0.22002200220022 \sin(20t) - 0.022002200220022 \cos(20t)) e^{7t} + 10.02200220022 e^{9t} - 10.0) e^{-9t} \text{ A}$$

```
[58]: #plotagem da Correntes
intervalo = np.arange(0,2,0.01)
sympplot(t,[iL1,iL2,iL3],intervalo,['iL1(t)','iL2(t)','iL3(t)'])
```

[58]:



Letra B

```
[59]: print('Letra B: Potência nos indutores L1, L2 e L3')
      pL1 = vL1*iL1
      pL2 = vL2*iL2
      pL3 = vL3*iL3
      symdisp('p_{L1}(t) = ', pL1.simplify(), 'W')
      symdisp('p_{L2}(t) = ', pL2.simplify(), 'W')
      symdisp('p_{L3}(t) = ', pL3.simplify(), 'W')
```

Letra B: Potência nos indutores L1, L2 e L3

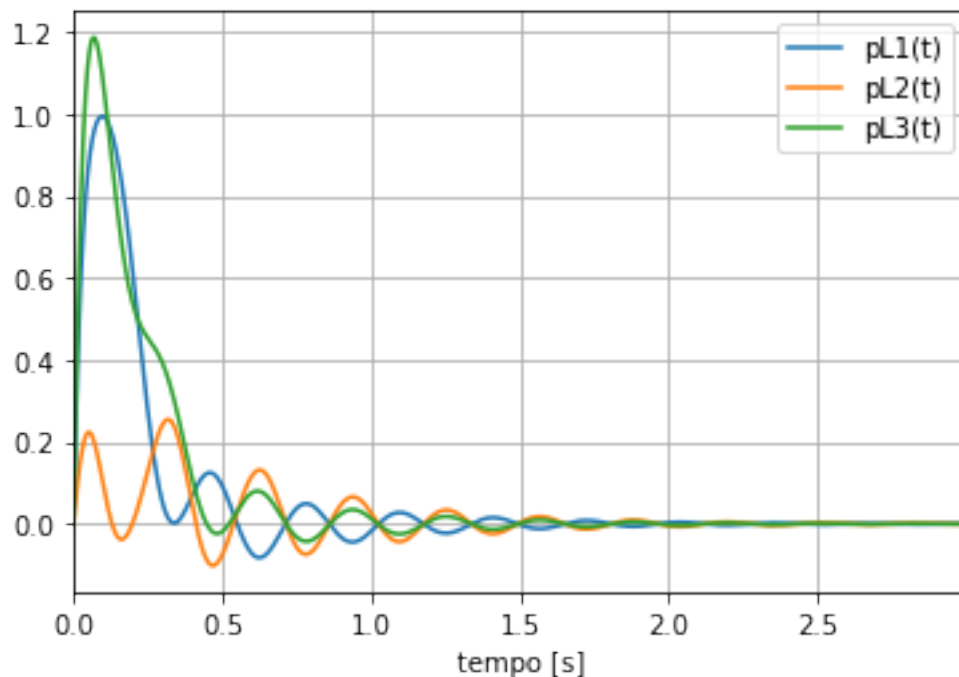
$$p_{L1}(t) = \left(0.02666666666666667e^{7t} \cos(20t) + 0.05e^{4t} - 0.45\right) \left((0.33003300330033 \sin(20t) - 0.033003300330033 \cos(20t))e^{3t} + 10.198019801980198\right)$$

$$p_{L2}(t) = (0.04e^{3t} \cos(20t) + 0.05) \left((1.98019801980198 \sin(20t) - 0.198019801980198 \cos(20t))e^{3t} + 10.198019801980198\right)$$

$$p_{L3}(t) = (0.02222222222222222e^{7t} \cos(20t) + 0.45) \left((0.22002200220022 \sin(20t) - 0.022002200220022 \cos(20t))e^{7t} + 10.198019801980198\right)$$

```
[60]: #plotagem da potencias
      intervalo = np.arange(0,3 ,0.001)
      symplot(t,[pL1,pL2,pL3],intervalo,['pL1(t)', 'pL2(t)', 'pL3(t)'])
```

[60]:



Letra C

```
[61]: print('Letra C: Potência Máxima nos indutores L1, L2 e L3')
      eqpL1 = sp.Eq(sp.diff(pL1,t),0)
      eqpL2 = sp.Eq(sp.diff(pL2,t),0)
      eqpL3 = sp.Eq(sp.diff(pL3,t),0)
```

```

# encontra numericamente o valor de t para o qual dp(t)/dt = 0
tL1max = sp.nsolve(eqpL1,t,(0,0.2), solver = 'bisect')
tL2max = sp.nsolve(eqpL2,t,(0.2,0.4), solver = 'bisect')
tL3max = sp.nsolve(eqpL3,t,(0,0.2), solver = 'bisect')
pL1max = pL1.evalf(subs={t:tL1max})
pL2max = pL2.evalf(subs={t:tL2max})
pL3max = pL3.evalf(subs={t:tL3max})

pmax = [pL1max,pL2max,pL3max]
print('Indutor que atinge o maior pico de potência:')

if (pL1max > pL2max) & (pL1max > pL3max):
    print('\n\tIndutor L1 atinge maior pico de potência.\n')
    print('Máximo valor de Potência:')
    symdisp('P_{LMax} = ', round(max(pmax),3), 'W')
    print('Instante em que a Potência é máxima:')
    symdisp('t_{LMax} = ', round(tL1max,3), 's')

elif (pL2max > pL1max) & (pL2max > pL3max):
    print('\n\tIndutor L2 atinge maior pico de potência.\n')
    print('Máximo valor de Potência:')
    symdisp('P_{LMax} = ', round(max(pmax),3), 'W')
    print('Instante em que a Potência é máxima:')
    symdisp('t_{LMax} = ', round(tL2max,3), 's')

elif (pL3max > pL1max) & (pL3max > pL2max):
    print('\n\tIndutor L3 atinge maior pico de potência.\n')
    print('Máximo valor de Potência:')
    symdisp('P_{LMax} = ', round(max(pmax),3), 'W')
    print('Instante em que a Potência é máxima:')
    symdisp('t_{LMax} = ', round(tL3max,3), 's')

```

Letra C: Potência Máxima nos indutores L1, L2 e L3
Indutor que atinge o maior pico de potência:

Indutor L3 atinge maior pico de potência.

Máximo valor de Potência:

$$P_{LMax} = 1.187 \text{ W}$$

Instante em que a Potência é máxima:

$$t_{LMax} = 0.068 \text{ s}$$

Letra D

```
[62]: print('Letra D: Energia em cada Indutor')
      #w = (1/2)*L*(i**2)
      print('Energia acumulada no Indutor L1')
      wL1 = sp.integrate(pL1,(t,0,100))
      symdisp('W_{L1} = ', round(wL1*(10**3),3), 'mJ')

      print('Energia acumulada no Indutor L2')
      wL2 = sp.integrate(pL2,(t,0,100))
      symdisp('W_{L2} = ', round(wL2*(10**3),3), 'mJ')

      print('Energia acumulada no Indutor L3')
      wL3 = sp.integrate(pL3,(t,0,100))
      symdisp('W_{L3} = ', round(wL3*(10**3),3), 'mJ')
```

Letra D: Energia em cada Indutor

Energia acumulada no Indutor L1

$$W_{L1} = 198.682 \text{ mJ}$$

Energia acumulada no Indutor L2

$$W_{L2} = 52.0 \text{ mJ}$$

Energia acumulada no Indutor L3

$$W_{L3} = 251.101 \text{ mJ}$$