HW2_EE_5601

September 27, 2024

[2]:
```
import skrf as rf
import matplotlib.pyplot as plt
import numpy as np
import cmath as cm
import math
import sympy as sp
from sympy.solvers import solve
pi = math.pi

#Prints out numbers without "np.flat64" displaying
np.set_printoptions(legacy='1.25')
```

0.1 Problem 2.19

[2]:
```
ZL = complex(80,-40)
Zo = complex(100,0)
Zg = 100

Gamma = (ZL-Zo)/(ZL+Zo)
print(Gamma)
Gamma_angle = np.angle(Gamma)
Gamma_mag = abs(Gamma)
Gamma_polar =[Gamma_mag,Gamma_angle]
print(Gamma_polar)
Vg = 10
Vo = Vg*(Zg/(Zg+Zo))
```

```
(-0.058823529411764705-0.23529411764705882j)
[0.24253562503633297, -1.8157749899217608]
```

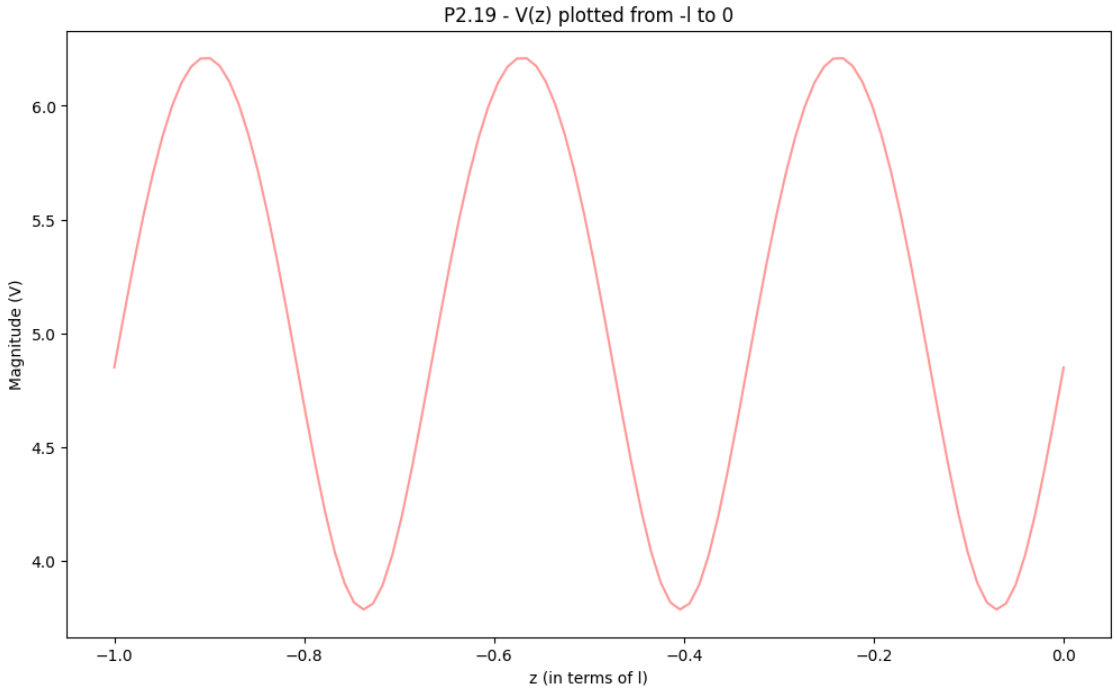
[3]:
```
#sanity check that .exp handles complex() correctly
print(np.exp(complex(1,1)))
print(abs(complex(1,1)))
print(np.angle(complex(1,1)))
```

```
(1.4686939399158851+2.2873552871788423j)
1.4142135623730951
0.7853981633974483
```

[4]:
```
#the values of Bz we input into the equation that satisfy -1.5l<=z<=0
#100 points from -3pi to 0
input = np.linspace((-3*pi),0,num=100)
#print(f'input={input}')

Vz = [0] * len(input) #somewhere to store the calculated results
k=0 #Vz index to be incremented.

#calculate Vz(z) for -1.5l<=z<=0 and store in vector "Vz"
for Bz in input :
    Vz[k] = Vo*(np.exp(complex(0,-Bz))+Gamma_mag*np.
 ↪exp((complex(0,Bz+Gamma_angle))))
    k+=1

#print(f'output={Vz}')



x_array = [element / (3*pi) for element in input] #scale Bz to be fraction of␣
 ↪l, -l<z<0 (Bz = -3pi when z=-l) for x-axis
y_array = [abs(element) for element in Vz] #magnitude of Vz for y-axis
```

[5]:
```
fig = plt.figure(figsize = (12,7))

plt.plot(x_array,y_array, alpha = 0.4, label='V(z)', color='red')
plt.title('P2.19 - V(z) plotted from -l to 0')
plt.xlabel('z (in terms of l)')
plt.ylabel('Magnitude (V)')
```

[5]:
```
Text(0, 0.5, 'Magnitude (V)')
```



[23]:
```
# for i in range(0, len(x_array)) :
#     print('Bl = {:<25} |V(z)| = {:<40}'.format(x_array[i],  y_array[i]))

# for i in range(0, len(x_array)) :
#     print(f'Bl = {x_array[i]}')

# for i in range(0, len(y_array)) :
#     print(f'|V(z)| = {y_array[i]}')
```

```
Bl = -1.0
Bl = -0.9898989898989898
Bl = -0.9797979797979799
Bl = -0.9696969696969697
Bl = -0.9595959595959596
Bl = -0.9494949494949496
Bl = -0.9393939393939394
Bl = -0.9292929292929293
Bl = -0.9191919191919191
Bl = -0.9090909090909092
Bl = -0.898989898989899
Bl = -0.8888888888888888
Bl = -0.8787878787878789
Bl = -0.8686868686868687
Bl = -0.8585858585858586
Bl = -0.8484848484848485
```

```
Bl = -0.8383838383838385
Bl = -0.8282828282828283
Bl = -0.8181818181818182
Bl = -0.8080808080808082
Bl = -0.797979797979798
Bl = -0.787878787878788
Bl = -0.7777777777777778
Bl = -0.7676767676767677
Bl = -0.7575757575757576
Bl = -0.7474747474747475
Bl = -0.7373737373737373
Bl = -0.7272727272727273
Bl = -0.7171717171717171
Bl = -0.7070707070707071
Bl = -0.6969696969696969
Bl = -0.6868686868686869
Bl = -0.6767676767676768
Bl = -0.6666666666666666
Bl = -0.6565656565656565
Bl = -0.6464646464646465
Bl = -0.6363636363636364
Bl = -0.6262626262626262
Bl = -0.6161616161616161
Bl = -0.6060606060606061
Bl = -0.5959595959595959
Bl = -0.5858585858585859
Bl = -0.5757575757575758
Bl = -0.5656565656565656
Bl = -0.5555555555555556
Bl = -0.5454545454545454
Bl = -0.5353535353535354
Bl = -0.5252525252525253
Bl = -0.5151515151515151
Bl = -0.5050505050505051
Bl = -0.4949494949494949
Bl = -0.48484848484848486
Bl = -0.4747474747474748
Bl = -0.46464646464646464
Bl = -0.4545454545454546
Bl = -0.44444444444444453
Bl = -0.43434343434343436
Bl = -0.4242424242424243
Bl = -0.41414141414141414
Bl = -0.4040404040404041
Bl = -0.393939393939394
Bl = -0.38383838383838387
Bl = -0.37373737373737376
Bl = -0.36363636363636365
```

```
Bl = -0.35353535353535354
Bl = -0.3434343434343435
Bl = -0.3333333333333333
Bl = -0.32323232323232326
Bl = -0.3131313131313131
Bl = -0.30303030303030304
Bl = -0.292929292929293
Bl = -0.2828282828282828
Bl = -0.27272727272727276
Bl = -0.2626262626262626
Bl = -0.25252525252525254
Bl = -0.2424242424242425
Bl = -0.23232323232323232
Bl = -0.22222222222222227
Bl = -0.21212121212121218
Bl = -0.20202020202020204
Bl = -0.19191919191919196
Bl = -0.18181818181818182
Bl = -0.17171717171717174
Bl = -0.16161616161616169
Bl = -0.15151515151515152
Bl = -0.14141414141414146
Bl = -0.1313131313131313
Bl = -0.12121212121212133
Bl = -0.11111111111111117
Bl = -0.10101010101010102
Bl = -0.09090909090909086
Bl = -0.08080808080808088
Bl = -0.07070707070707073
Bl = -0.06060606060606057
Bl = -0.0505050505050506
Bl = -0.04040404040404044
Bl = -0.030303030303030287
Bl = -0.020202020202020315
Bl = -0.010101010101010157
Bl = 0.0
|V(z)| = 4.850712500726662
|V(z)| = 5.080260611708739
|V(z)| = 5.3021310614709805
|V(z)| = 5.509699932309544
|V(z)| = 5.69741479244118
|V(z)| = 5.860698336885376
|V(z)| = 5.995859593089853
|V(z)| = 6.100021055360177
|V(z)| = 6.171063270217249
|V(z)| = 6.207585636043595
|V(z)| = 6.208881503463191
|V(z)| = 6.174925961722788
```

```
|V(z)| = 6.106375412671366
|V(z)| = 6.004578899773368
|V(z)| = 5.871602030001632
|V(z)| = 5.710265060547235
|V(z)| = 5.524197065901182
|V(z)| = 5.317907524514072
|V(z)| = 5.096874113048223
|V(z)| = 4.86763904654876
|V(z)| = 4.637892888022077
|V(z)| = 4.4165006231738335
|V(z)| = 4.213388459235438
|V(z)| = 4.03917022945626
|V(z)| = 3.904382556725633
|V(z)| = 3.818279843472876
|V(z)| = 3.787360865701637
|V(z)| = 3.81408265340437
|V(z)| = 3.8963177496902577
|V(z)| = 4.027825977594335
|V(z)| = 4.199503891339053
|V(z)| = 4.4008629649537845
|V(z)| = 4.621263618031345
|V(z)| = 4.850712500726661
|V(z)| = 5.080260611708738
|V(z)| = 5.302131061470981
|V(z)| = 5.509699932309544
|V(z)| = 5.697414792441181
|V(z)| = 5.860698336885377
|V(z)| = 5.995859593089852
|V(z)| = 6.100021055360177
|V(z)| = 6.171063270217247
|V(z)| = 6.207585636043595
|V(z)| = 6.208881503463191
|V(z)| = 6.174925961722789
|V(z)| = 6.106375412671366
|V(z)| = 6.004578899773368
|V(z)| = 5.871602030001634
|V(z)| = 5.710265060547235
|V(z)| = 5.524197065901182
|V(z)| = 5.317907524514072
|V(z)| = 5.096874113048223
|V(z)| = 4.867639046548761
|V(z)| = 4.637892888022078
|V(z)| = 4.4165006231738335
|V(z)| = 4.213388459235437
|V(z)| = 4.039170229456259
|V(z)| = 3.9043825567256336
|V(z)| = 3.8182798434728764
|V(z)| = 3.7873608657016375
```

```
|V(z)| = 3.81408265340437
|V(z)| = 3.896317749690257
|V(z)| = 4.027825977594336
|V(z)| = 4.199503891339054
|V(z)| = 4.4008629649537845
|V(z)| = 4.621263618031344
|V(z)| = 4.85071250072666
|V(z)| = 5.080260611708735
|V(z)| = 5.302131061470983
|V(z)| = 5.509699932309544
|V(z)| = 5.6974147924411795
|V(z)| = 5.860698336885377
|V(z)| = 5.995859593089852
|V(z)| = 6.100021055360176
|V(z)| = 6.171063270217247
|V(z)| = 6.207585636043594
|V(z)| = 6.208881503463191
|V(z)| = 6.174925961722789
|V(z)| = 6.106375412671367
|V(z)| = 6.004578899773368
|V(z)| = 5.871602030001634
|V(z)| = 5.710265060547235
|V(z)| = 5.524197065901182
|V(z)| = 5.317907524514075
|V(z)| = 5.096874113048223
|V(z)| = 4.867639046548762
|V(z)| = 4.63789288802208
|V(z)| = 4.416500623173835
|V(z)| = 4.213388459235438
|V(z)| = 4.039170229456259
|V(z)| = 3.9043825567256327
|V(z)| = 3.818279843472877
|V(z)| = 3.787360865701637
|V(z)| = 3.8140826534043697
|V(z)| = 3.8963177496902563
|V(z)| = 4.027825977594336
|V(z)| = 4.199503891339054
|V(z)| = 4.400862964953781
|V(z)| = 4.621263618031344
|V(z)| = 4.85071250072666
```

0.2 Problem 2.24

[19]:
```
Zo = 75
ZL = 40
Z1 = np.sqrt(Zo*ZL)
```

```
input = np.linspace(0.5,2,num=100) # vector of f from 0.5 to 2 in over 100 slices
SWR = [999] * len(input)

#Calculate SWR for every f value
index=0
for f in input :
    Zin = Z1*((ZL+complex(0,Z1*np.tan((pi/2)*f)))/(Z1+complex(0,ZL*np.tan((pi/2)*f))))
    ref_mag = abs((Zin-Zo)/(Zin+Zo)) #reflection coefficient magnitude
    SWR[index] = (1+ref_mag)/(1-ref_mag)

    index+=1

x_array = [element for element in input] #frequency = x-axis
y_array = [element for element in SWR] #SWR = y-axis
```

[22]:
```
#Plotting the data
fig = plt.figure(figsize = (12,7))

plt.plot(x_array,y_array, alpha = 0.4, label='V(z)', color='blue')
plt.title('P2.24, SWR plotted from 0.5 <= f/fo <= 2.0')
plt.xlabel('f/fo')
plt.ylabel('SWR')
```

[22]:
```
Text(0, 0.5, 'SWR')
```