

Joe Stanley

ECE522 - EXAM1

In [2]:

```
1 # Import Necessary Libraries
2 import numpy as np
3 import matplotlib.pyplot as plt
4 from scipy.optimize import fsolve
5 import electricpy as ep
6 from electricpy.constants import *
7
8 # Set Boolean Control for Report Style
9 debug = True
```

Problem III:

Repeat Parts B and C of Problem I for the situation where the parameter L/r in the "slip calculator" is in error by -25%.

Comment on the effect on steady state performance of such "detuning" of the controller.

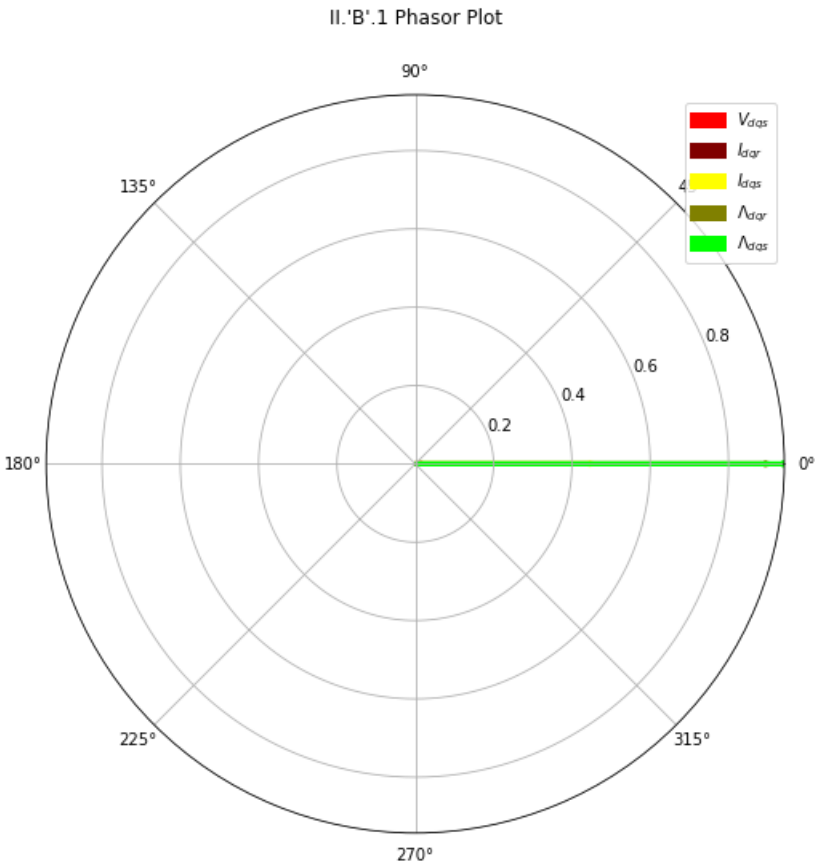
Just like Problem II, since we know: $s = \frac{\omega_{es} - \omega_r}{\omega_{es}}$, we can manipulate the equation into the form: $(\omega_{es} - \omega_r) = s \cdot \omega_{es}$. In this form, we can substitute it into our equations to solve.

Part 'B' (since we're only repeating parts B and C of problem 1):

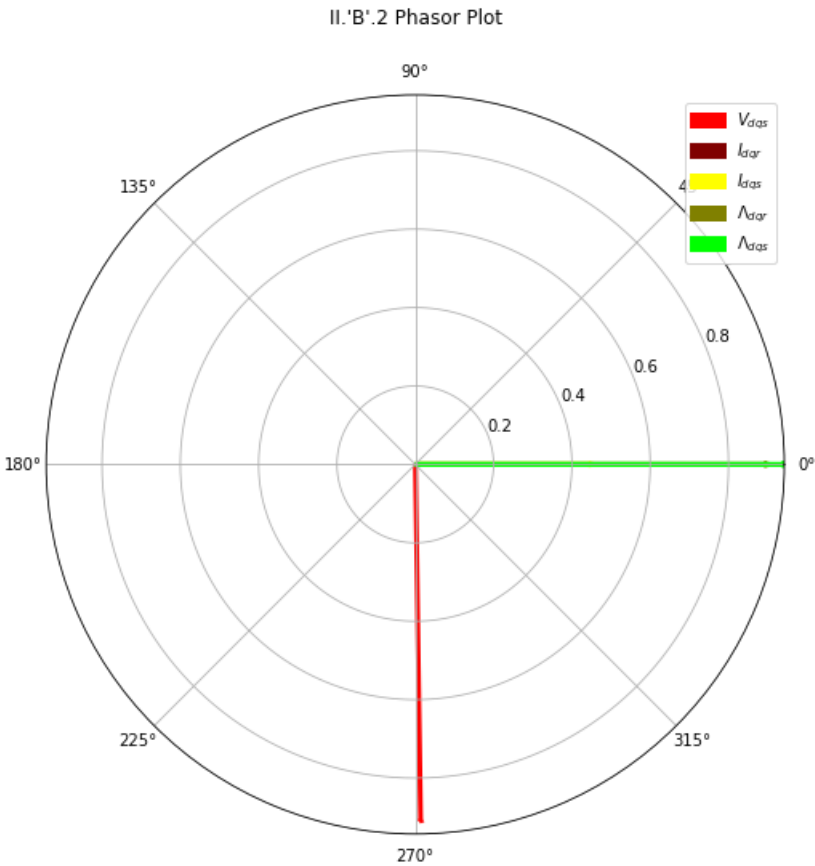
In [3]:

```
1  # Define Provided Machine Parameters
2  rs = 0.03 #pu
3  LLs = 0.1 #pu
4  Lm = 2.0 #pu
5  LLr = 0.1 #pu
6  rr = 0.03 #pu
7
8  # Define Rated Criteria
9  VdqsMag = 1
10 wes = 1
11 Tem = 0
12
13 #####
14 # Read Data Calculated from Problem 1
15 with open("constants.txt",'r') as file:
16     s_rated = float(file.readline())
17     w_rated = float(file.readline())
18     lamdr_rated = float(file.readline())
19 print("S-rated:",s_rated,"tw-rated:",w_rated,"tLambda-rated:",lamdr_rated)
20 #####
21
22 texlabels = [
23     "$V_{dqs}$",
24     "$I_{dqr}$",
25     "$I_{dqs}$",
26     "$\\Lambda_{dqr}$",
27     "$\\Lambda_{dqs}$",
28 ]
29 labels = [
30     "Vdqs:",
31     "Idqr:",
32     "Idqs:",
33     "λdqr:",
34     "λdqs:",
35 ]
36
37 #####
38 # "B".1)
39 wr = 0.0
40 LAMdr = lamdr_rated
41
42 # Generate Phasor Plot
43 Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
44 clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
45 print("w-slip:",wslip,"tw-es",wes)
46 clist *= ep.phs(-np.angle(clist[3],deg=True))
47 ep.phasorplot(clist,"III.'B'.1 Phasor Plot",texlabels,filename="III-B-1",size=8,linewidth=3,plot=deb)
48
49 #####
50 # "B".2)
51 wr = w_rated
52 LAMdr = lamdr_rated
53
54 # Generate Phasor Plot
55 Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
56 clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
57 print("w-slip:",wslip,"tw-es",wes)
58 clist *= ep.phs(-np.angle(clist[3],deg=True))
59 ep.phasorplot(clist,"III.'B'.2 Phasor Plot",texlabels,filename="III-B-2",size=8,linewidth=3,plot=deb)
60
61 #####
62 # "B".3)
63 wr = 2*w_rated
64 LAMdr = lamdr_rated/2
65
66 # Generate Phasor Plot
67 Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
68 clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
69 print("w-slip:",wslip,"tw-es",wes)
70 clist *= ep.phs(-np.angle(clist[3],deg=True))
71 ep.phasorplot(clist,"III.'B'.3 Phasor Plot",texlabels,filename="III-B-3",size=8,linewidth=3,plot=deb)
```

S-rated: 0.03723079497495241 w-rated: 0.9627692050250476 Lambda-rated: 0.8976550377456242
w-slip: 0.0 w-es 0.0



w-slip: 0.0 w-es 0.9627692050250476

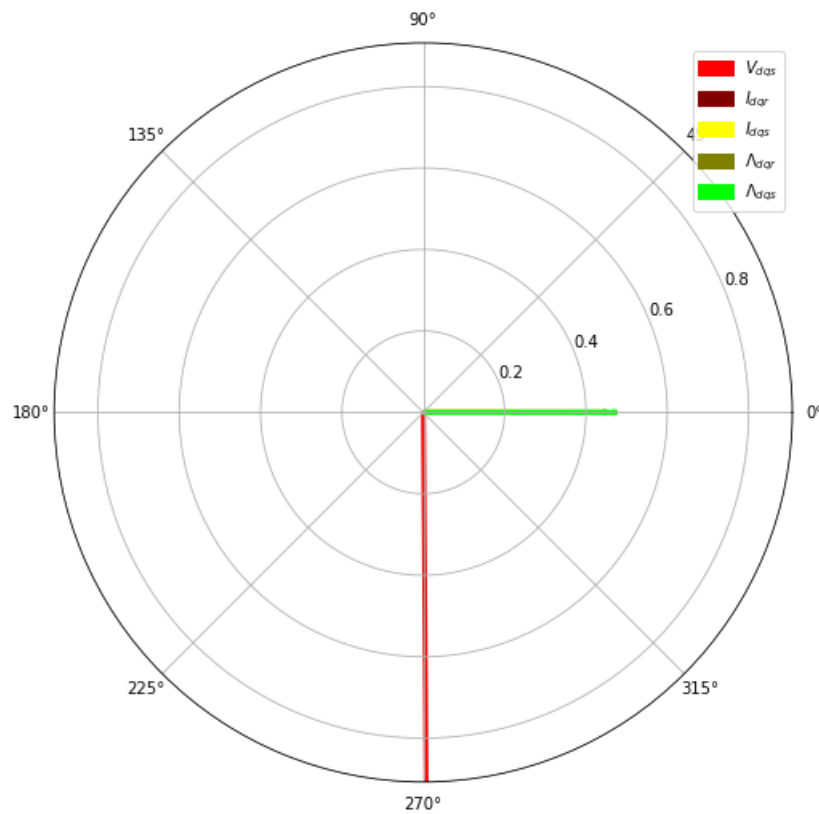


w-slip: 0.0

w-es 1.9255384100500952



II.'B'.3 Phasor Plot



Part 'C' (since we're only repeating parts B and C of problem 1):

Comments and Analysis:

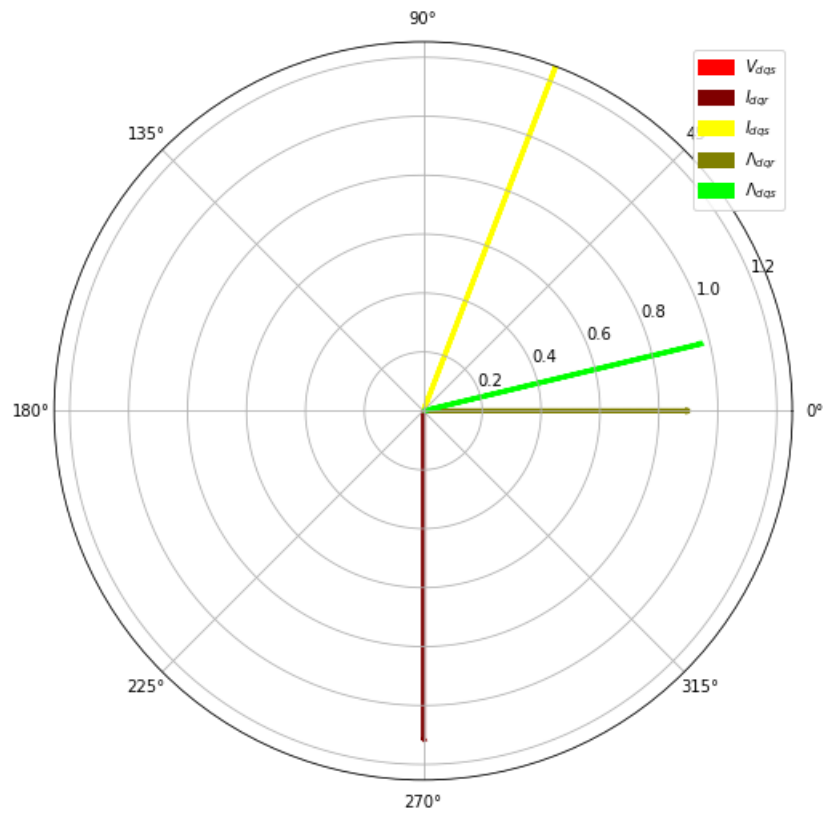
From comparison, it seems clear that these results are not too dissimilar from those found in the first problem (problem I). Perhaps the only truly notable difference is that magnitude difference between the results. Angle differences and general relations between the vectors appear to be largely the same between the Problem I results and these Problem III results. It is interesting to see that the magnitudes again seem to reflect the error in slip. It seems that perhaps it could be drawn that slip is directly proportional to these terms.

In [4]:

```
1 #####
2 # "C".1)
3 Tem = 1.0
4 wr = 0.0
5 LAMdr = lamdr_rated
6
7 # Generate Phasor Plot
8 Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
9 clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
10 print("w-slip:",wslip,"\tw-es",wes)
11 clist *= ep.phs(-np.angle(clist[3],deg=True))
12 ep.phasorplot(clist,"III.'C'.1 Phasor Plot",texlabels,filename="III-C-1",size=8,linewidth=3,plot=deb
13
14 #####
15 # "C".2)
16 Tem = 1.0
17 wr = w_rated
18 LAMdr = lamdr_rated
19
20 # Generate Phasor Plot
21 Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
22 clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
23 print("w-slip:",wslip,"\tw-es",wes)
24 clist *= ep.phs(-np.angle(clist[3],deg=True))
25 ep.phasorplot(clist,"III.'C'.2 Phasor Plot",texlabels,filename="III-C-2",size=8,linewidth=3,plot=deb
26
27 #####
28 # "C".3)
29 Tem = 0.5
30 wr = 2*w_rated
31 LAMdr = lamdr_rated/2
32
33 # Generate Phasor Plot
34 Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
35 clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
36 print("w-slip:",wslip,"\tw-es",wes)
37 clist *= ep.phs(-np.angle(clist[3],deg=True))
38 ep.phasorplot(clist,"III.'C'.3 Phasor Plot",texlabels,filename="III-C-3",size=8,linewidth=3,plot=deb
```

w-slip: 0.04964105996636662 w-es 0.04964105996636662

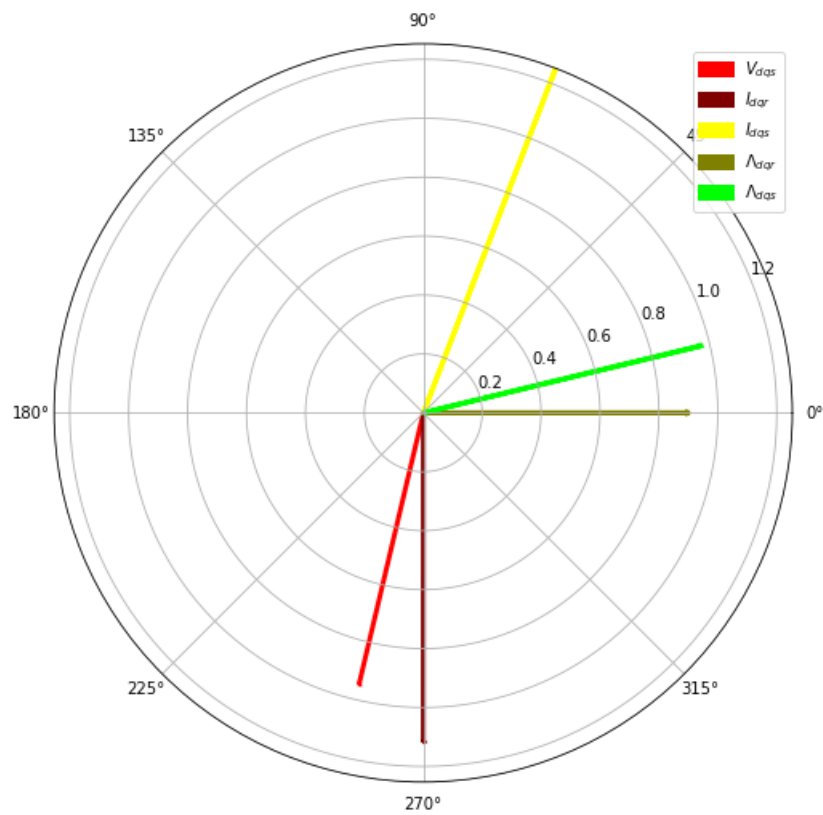
III.'C'.1 Phasor Plot



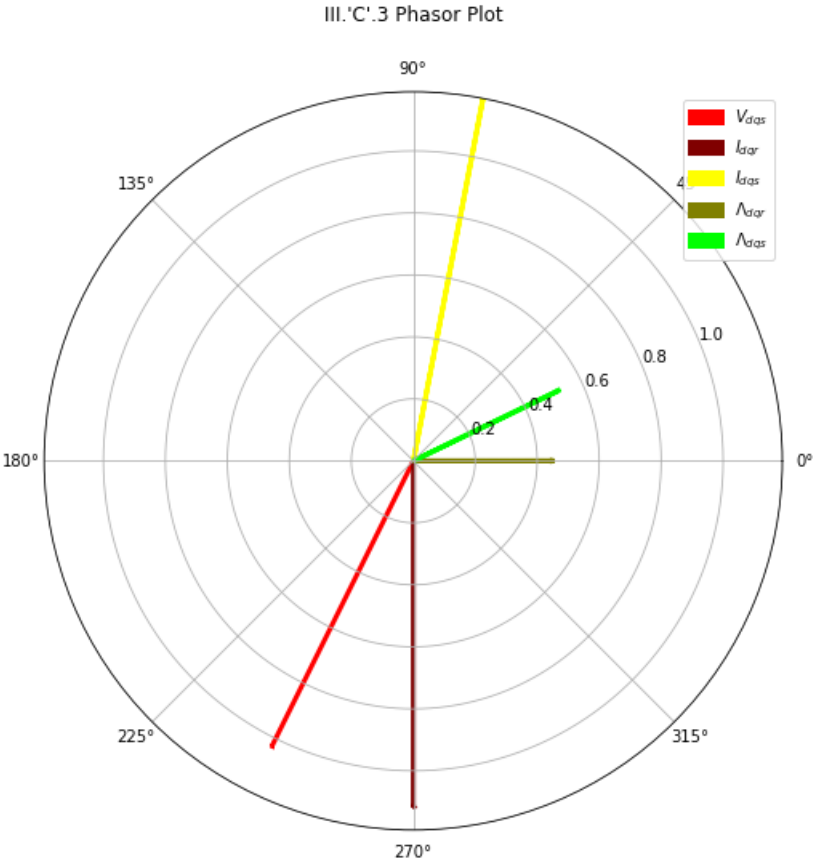
w-slip: 0.04964105996636662

w-es 1.0124102649914142

III.'C'.2 Phasor Plot



w-slip: 0.09928211993273324 w-es 2.0248205299828284



Comments:

We see the same behavior as the first problem!

In []:

1