Joe Stanley

ECE522 - EXAM1

In [2]:

```python
# Import Necessary Libraries
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import fsolve
import electricpy as ep
from electricpy.constants import *

# Set Boolean Control for Report Style
debug = True
```

Problem III:

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

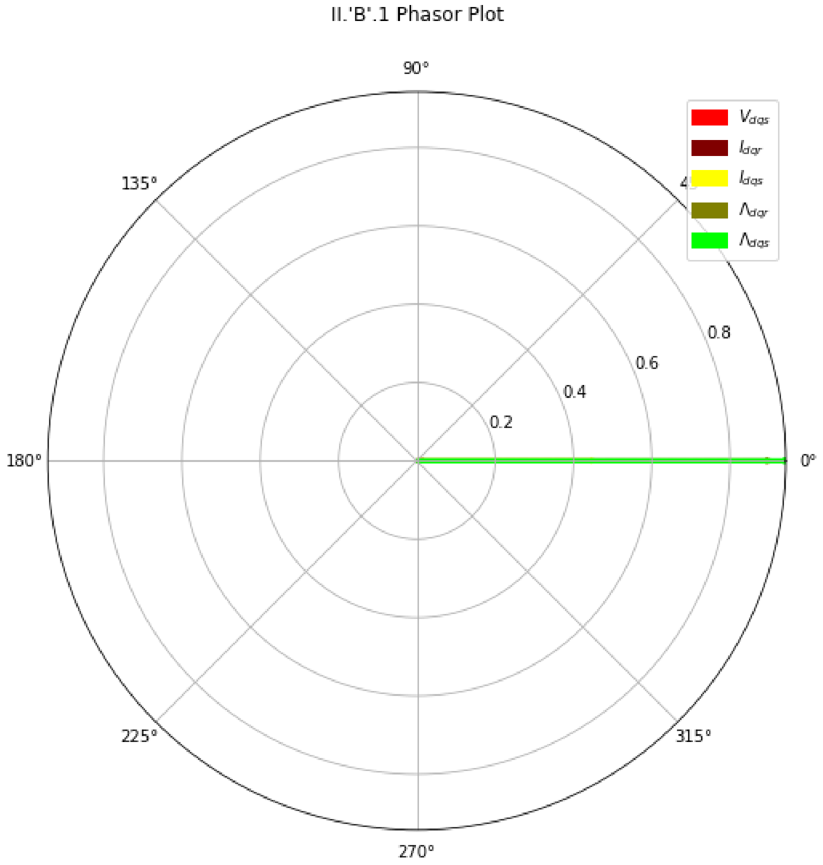
Comment on the effect on steady state performance of such "detuning" of the controller.

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.

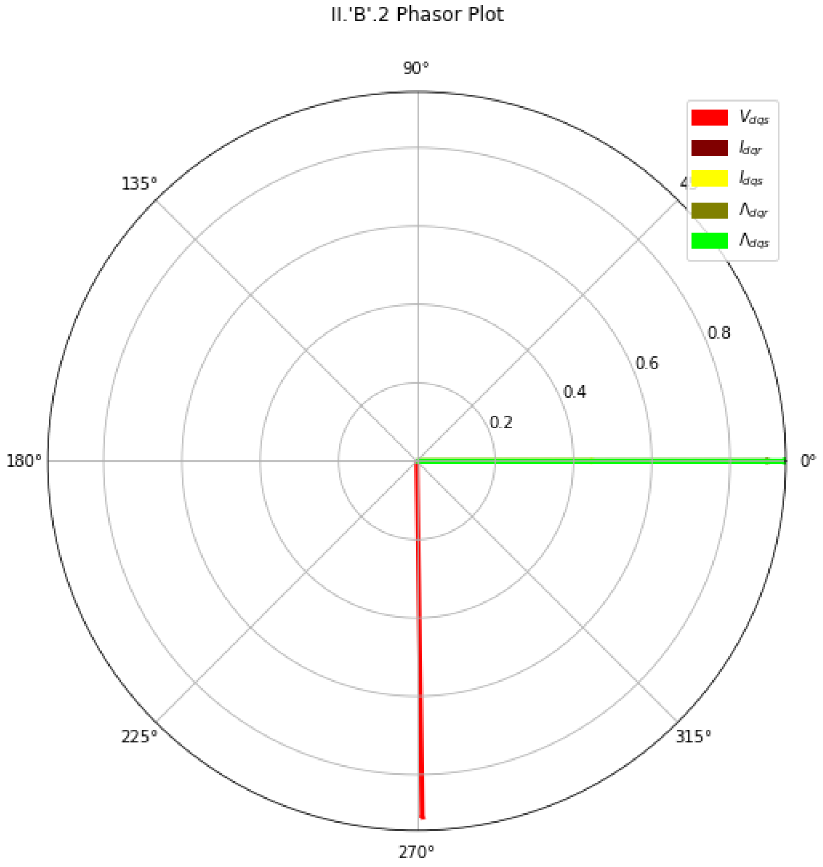
Part 'B' (since we're only repeating parts B and C of problem 1):

In [3]:

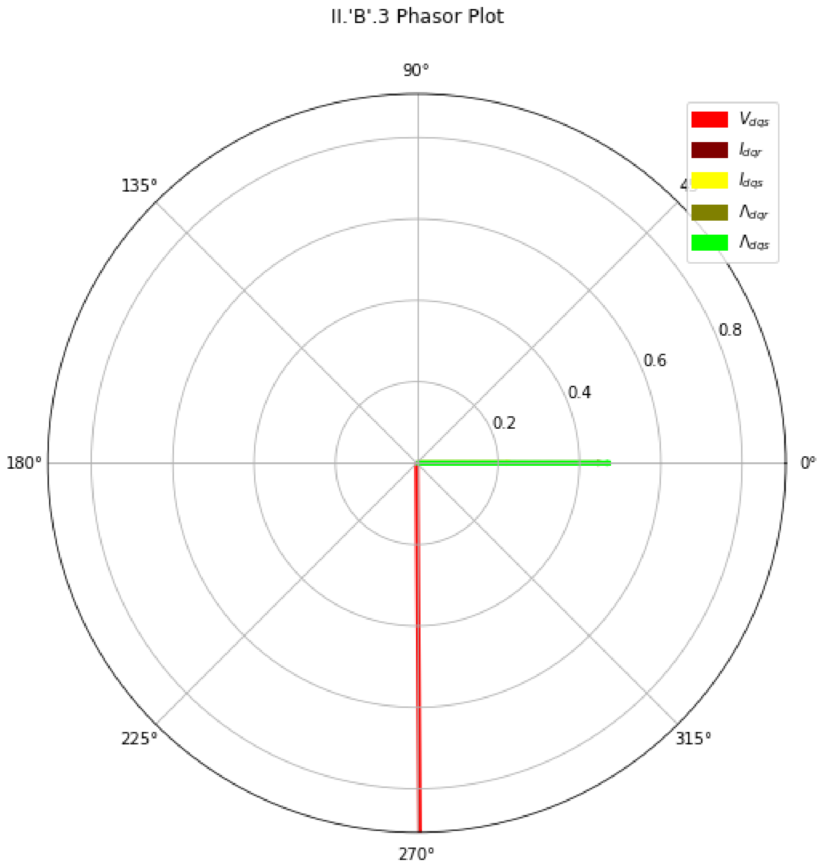
```python
# Define Provided Machine Parameters
rs = 0.03 #pu
LLs = 0.1 #pu
Lm = 2.0 #pu
LLr = 0.1 #pu
rr = 0.03 #pu

# Define Rated Criteria
VdqsMag = 1
wes = 1
Tem = 0

####################################################
# Read Data Calculated from Problem 1
with open("constants.txt",'r') as file:
    s_rated = float(file.readline())
    w_rated = float(file.readline())
    lamdr_rated = float(file.readline())
print("S-rated:",s_rated,"\tw-rated:",w_rated,"\tLambda-rated:",lamdr_rated)
####################################################

texlabels = [
    "$V_{dqs}$",
    "$I_{dqr}$",
    "$I_{dqs}$",
    "$\\Lambda_{dqr}$",
    "$\\Lambda_{dqs}$",
]
labels = [
    "Vdqs:",
    "Idqr:",
    "Idqs:",
    "λdqr:",
    "λdqs:",
]

######################################################################
# "B".1)
wr = 0.0
LAMdr = lamdr_rated

# Generate Phasor Plot
Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
print("w-slip:",wslip,"\tw-es",wes)
clist *= ep.phs(-np.angle(clist[3],deg=True))
ep.phasorplot(clist,"III.'B'.1 Phasor Plot",texlabels,filename="III-B-1",size=8,linewidth=3,plot=deb

######################################################################
# "B".2)
wr = w_rated
LAMdr = lamdr_rated

# Generate Phasor Plot
Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
print("w-slip:",wslip,"\tw-es",wes)
clist *= ep.phs(-np.angle(clist[3],deg=True))
ep.phasorplot(clist,"III.'B'.2 Phasor Plot",texlabels,filename="III-B-2",size=8,linewidth=3,plot=deb

######################################################################
# "B".3)
wr = 2*w_rated
LAMdr = lamdr_rated/2

# Generate Phasor Plot
Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
print("w-slip:",wslip,"\tw-es",wes)
clist *= ep.phs(-np.angle(clist[3],deg=True))
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
```



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

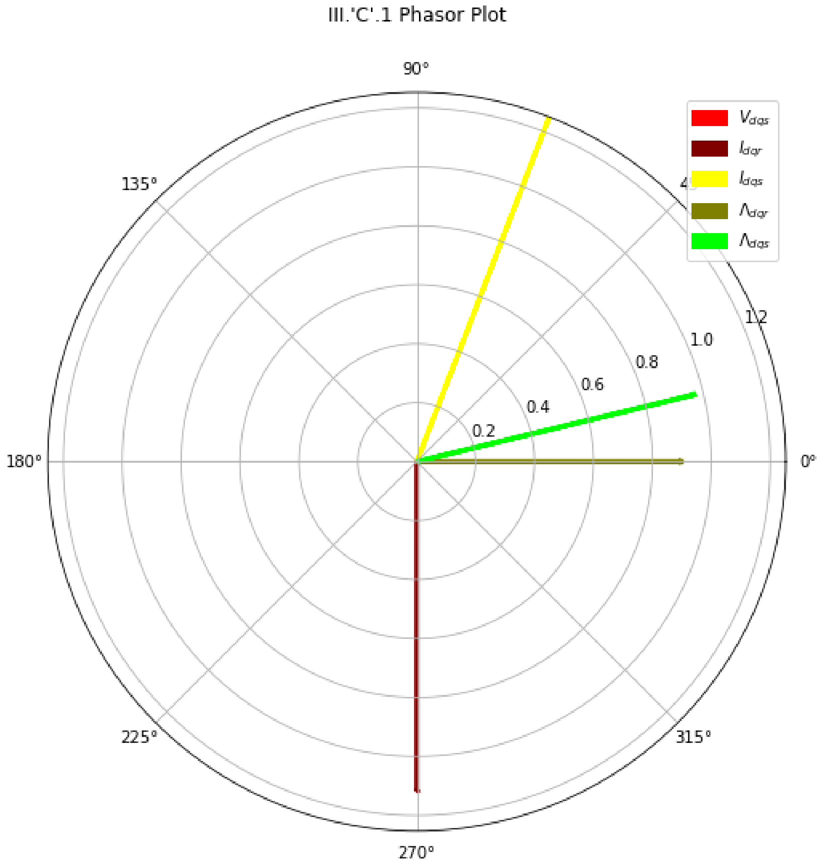
Comments and Analysis:

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.

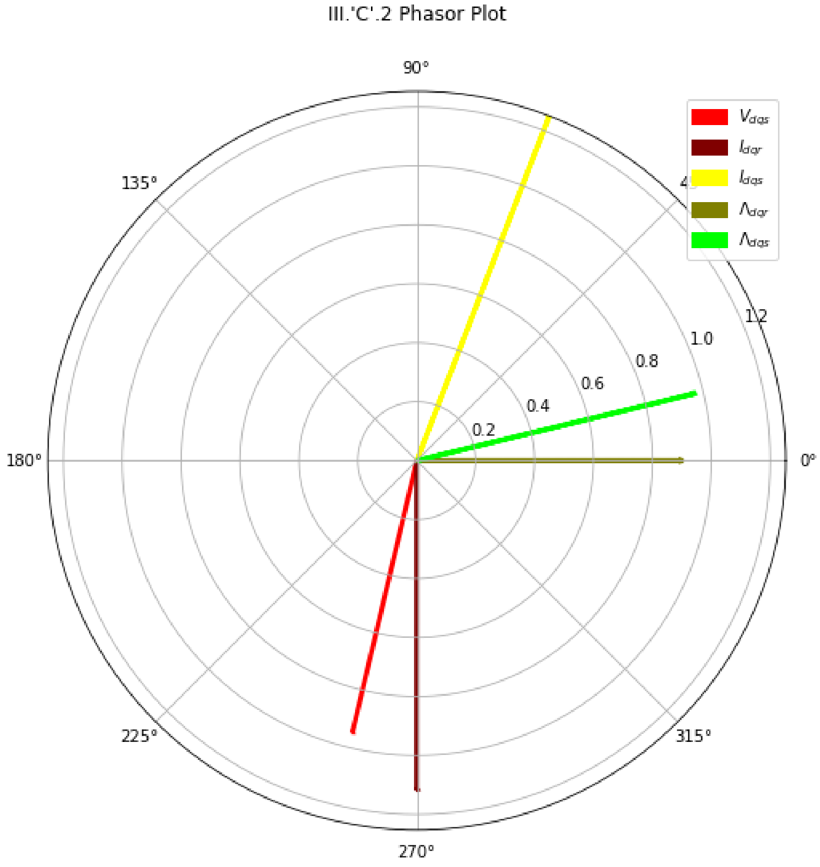
In [4]:

```
######################################################################
# "C".1)
Tem = 1.0
wr = 0.0
LAMdr = lamdr_rated

# Generate Phasor Plot
Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
print("w-slip:",wslip,"\tw-es",wes)
clist *= ep.phs(-np.angle(clist[3],deg=True))
ep.phasorplot(clist,"III.'C'.1 Phasor Plot",texlabels,filename="III-C-1",size=8,linewidth=3,plot=deb

######################################################################
# "C".2)
Tem = 1.0
wr = w_rated
LAMdr = lamdr_rated

# Generate Phasor Plot
Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
print("w-slip:",wslip,"\tw-es",wes)
clist *= ep.phs(-np.angle(clist[3],deg=True))
ep.phasorplot(clist,"III.'C'.2 Phasor Plot",texlabels,filename="III-C-2",size=8,linewidth=3,plot=deb

######################################################################
# "C".3)
Tem = 0.5
wr = 2*w_rated
LAMdr = lamdr_rated/2

# Generate Phasor Plot
Vdqs,Idqr,Idqs,LAMdqr,LAMdqs,wslip,wes = ep.imfoc_control(Tem,LAMdr,wr,rr,rs,Lm,LLr,LLs,s_err=-0.25)
clist = np.array([Vdqs,Idqr,Idqs,LAMdqr,LAMdqs])
print("w-slip:",wslip,"\tw-es",wes)
clist *= ep.phs(-np.angle(clist[3],deg=True))
ep.phasorplot(clist,"III.'C'.3 Phasor Plot",texlabels,filename="III-C-3",size=8,linewidth=3,plot=deb
```

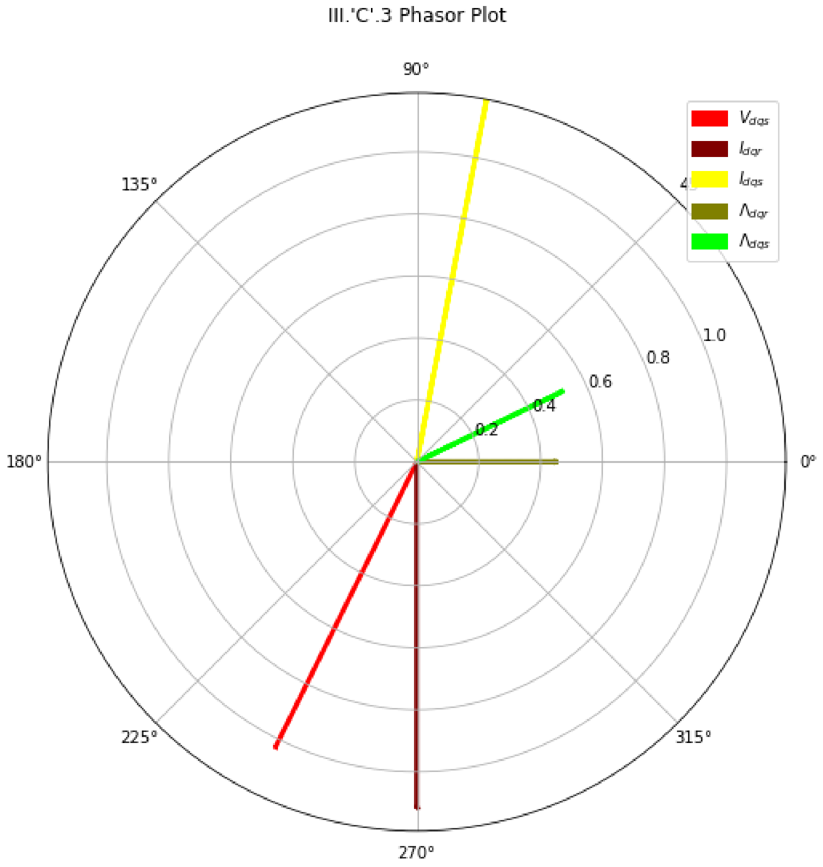
```
w-slip: 0.04964105996636662     w-es 0.04964105996636662
```



```
w-slip: 0.04964105996636662     w-es 1.0124102649914142
```



```
w-slip: 0.09928211993273324     w-es 2.0248205299828284
```



Comments:

We see the same behavior as the first problem!

In [ ]:

```
```