ESA614: Computational Astrophysics
Assignment # 2

Name: Abraham Mathews
ID: SC22M077
Program: MS Astronomy and Astrophysics

---

1. The angular distance γ between two coordinates (α_1, δ_1) and (α_2, δ_2) is given by $\cos(\gamma) = \sin(\delta_1)\sin(\delta_2) + \cos(\delta_1)\cos(\delta_2)\cos(\alpha_1 - \alpha_2)$, where α and δ respectively are the right ascension and declination.

   The Fermi Gamma-ray Burst Monitor (GBM) detected a gamma-ray pulse at the position α=176.8deg, δ=-39.8deg (J2000), with an error region of 11.6 deg [*]. Near simultaneously, the Laser Interferometer Gravitational Observatory (LIGO) localised an event with the coordinates α=186.62deg, δ=-48.84deg (J2000) with an error radius of 17.45deg [†].

   Plot the two error regions in α vs δ plane. (Hint:- Use any of the root finding techniques we discussed in the class to obtain δ_2 corresponding to a chosen α_2 or vice versa.)

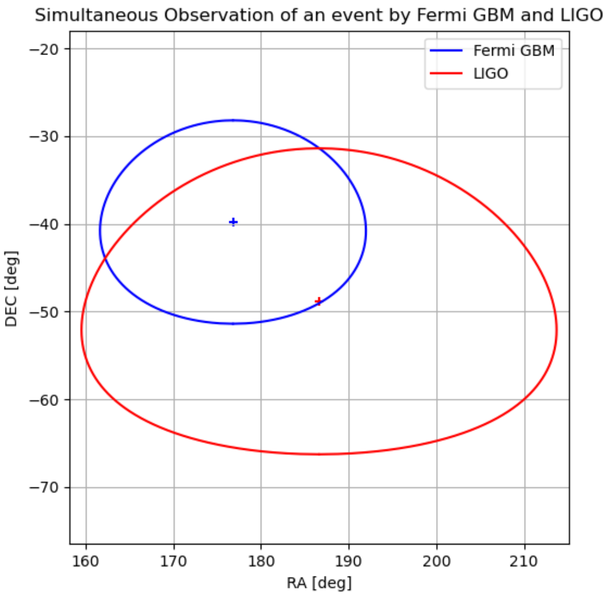
Python Code:

```
#Fermi and LIGO
import matplotlib.pyplot as plt
import numpy as np
import math
fig = plt.figure(figsize = (10,10))

# Data

# Fermi GBM

alphaFdeg = 176.8
decFdeg = -39.8
errorFdeg = 11.6
plt.scatter(alphaFdeg,decFdeg,marker='+',color = 'blue') # Plot the center
alphaFrad = alphaFdeg * (math.pi/180)
decFrad = decFdeg * (math.pi/180)
errorFrad = errorFdeg * (math.pi/180)

#LIGO

alphaLdeg = 186.62
decLdeg = -48.84
errorLdeg = 17.45
plt.scatter(alphaLdeg,decLdeg,marker='+',color = 'red') # Plot the center
alphaLrad = alphaLdeg * (math.pi/180)
decLrad = decLdeg * (math.pi/180)
```

```
errorLrad = errorLdeg * (math.pi/180)

#Function and Derivative for NR (Fermi)
def y_fermi(RA,DEC):
  y =
math.cos(errorFrad)-(math.sin(decFrad)*math.sin(DEC))-(math.cos(decFrad)*math.cos(DEC)*math.cos(RA-alphaFrad))
  return y

def y_fermi_dash(RA,DEC):
  y_dash = (math.cos(decFrad)*math.cos(DEC)*math.sin(RA-alphaFrad))
  return y_dash

#Function and Derivative for NR (LIGO)
def y_LIGO(RA,DEC):
  y =
math.cos(errorLrad)-(math.sin(decLrad)*math.sin(DEC))-(math.cos(decLrad)*math.cos(DEC)*math.cos(RA-alphaLrad))
  return y

def y_LIGO_dash(RA,DEC):
  y_dash = (math.cos(decLrad)*math.cos(DEC)*math.sin(RA-alphaLrad))
  return y_dash

#Root Finding for Fermi
DEC_Fermi = np.linspace((decFdeg-11.6)*(math.pi/180),(decFdeg+11.6)*(math.pi/180),1000)
RA_Fermi_left = []
RA_Fermi_right = []

#For Fermi's left half
for i in range(len(DEC_Fermi)):
  RA_initial = alphaFrad-0.4
  iter = 0
  NMAX = 100
  while(iter<=NMAX):
    RA_temp = RA_initial - (y_fermi(RA_initial,DEC_Fermi[i]))/(y_fermi_dash(RA_initial,DEC_Fermi[i]))
    iter = iter + 1
    RA_initial = RA_temp
    if(y_fermi(RA_initial,DEC_Fermi[i])<1e-7 and y_fermi(RA_initial,DEC_Fermi[i])>-1e-7):
      break
  RA_Fermi_left.append(RA_initial)


np_fermi_RAleft = np.array(RA_Fermi_left)
plt.plot(np_fermi_RAleft*(180/math.pi),DEC_Fermi*(180/math.pi),color ='blue')

#For Fermi's right half
for i in range(len(DEC_Fermi)):
  RA_initial = alphaFrad+0.4
  iter = 0
  NMAX = 100
  while(iter<=NMAX):
    RA_temp = RA_initial - (y_fermi(RA_initial,DEC_Fermi[i]))/(y_fermi_dash(RA_initial,DEC_Fermi[i]))
    iter = iter + 1
    RA_initial = RA_temp
    if(y_fermi(RA_initial,DEC_Fermi[i])<1e-7 and y_fermi(RA_initial,DEC_Fermi[i])>-1e-7):
```

```
        break
    RA_Fermi_right.append(RA_initial)


np_fermi_RAright = np.array(RA_Fermi_right)
plt.plot(np_fermi_RAright*(180/math.pi),DEC_Fermi*(180/math.pi),color='blue',label ="Fermi GBM")

#Root Finding for LIGO
DEC_LIGO = np.linspace((decLdeg-17.45)*(math.pi/180),(decLdeg+17.45)*(math.pi/180),1000)
RA_LIGO_left = []
RA_LIGO_right = []

#For LIGO's left half
for i in range(len(DEC_LIGO)):
    RA_initial = alphaLrad-0.4
    iter = 0
    NMAX = 100
    while(iter<=NMAX):
        RA_temp = RA_initial - (y_LIGO(RA_initial,DEC_LIGO[i]))/(y_LIGO_dash(RA_initial,DEC_LIGO[i]))
        iter = iter + 1
        RA_initial = RA_temp
        if(y_LIGO(RA_initial,DEC_LIGO[i])<1e-7 and y_LIGO(RA_initial,DEC_LIGO[i])>-1e-7):
            break
    RA_LIGO_left.append(RA_initial)


np_LIGO_RAleft = np.array(RA_LIGO_left)
plt.plot(np_LIGO_RAleft*(180/math.pi),DEC_LIGO*(180/math.pi),color ='red')

#For LIGO's right half
for i in range(len(DEC_LIGO)):
    RA_initial = alphaLrad+0.4
    iter = 0
    NMAX = 100
    while(iter<=NMAX):
        RA_temp = RA_initial - (y_LIGO(RA_initial,DEC_LIGO[i]))/(y_LIGO_dash(RA_initial,DEC_LIGO[i]))
        iter = iter + 1
        RA_initial = RA_temp
        if(y_LIGO(RA_initial,DEC_LIGO[i])<1e-7 and y_LIGO(RA_initial,DEC_LIGO[i])>-1e-7):
            break
    RA_LIGO_right.append(RA_initial)


np_LIGO_RAright = np.array(RA_LIGO_right)
plt.plot(np_LIGO_RAright*(180/math.pi),DEC_LIGO*(180/math.pi),color='red',label ="LIGO")

plt.title('Simultaneous Observation of an event by Fermi GBM and LIGO')
plt.axis('equal')
plt.xlabel("RA [deg]")
plt.ylabel("DEC [deg]")
plt.grid()
plt.legend()
plt.show()
```

Output:



Simultaneous Observation of an event by Fermi GBM and LIGO
Fermi GBM
LIGO
DEC [deg]
RA [deg]
−20
−30
−40
−50
−60
−70
160
170
180
190
200
210