```
#numericalBerryphaseparticlein magnetic field

import numpy as np

def upeigenvector(theta,phi):
    nz=np.cos(theta)
    nx=np.sin(theta)*np.cos(phi)
    ny=np.sin(theta)*np.sin(phi)


    H11=nz
    H22=-nz
    H12=nx-1j*ny
    H21=nx+1j*ny

    H=np.array([[H11,H12],[H21,H22]])

    eigvals,eigvecs=np.linalg.eig(H)
    idx=np.argmax(eigvals.real)
    v=eigvecs[:,idx]
    v=v/np.linalg.norm(v)

    return(v)

#print(upeigenvector(78,908))

N=700
theta1=np.linspace(0,np.pi/2,N)
#print(theta1)
phi1=np.linspace(0,np.pi/2,N)
#print(phi1[0],phi1[N])
prod=1
for i in range(N-1):
    a=np.vdot(upeigenvector(theta1[i],0),upeigenvector(theta1[i+1],0))
    prod=prod*a#from north pole to x=1,y=0,z=0

#print(prod)
m=prod
for i in range(1,N-1):
    b=np.vdot(upeigenvector(np.pi/2,phi1[i]),upeigenvector(np.pi/2,phi1[i+1]))
    m=m*b#from x=1,y=0,z=0 to x=0,y=1,z=0

#print(m)
p=m
phi1=np.linspace(np.pi/2,0,N)

for i in range(N-1):
    c=np.vdot(upeigenvector(phi1[i],np.pi/2),upeigenvector(phi1[i+1],np.pi/2))
    p=p*c#from x=0,y=1,z=0 to x=0,y=0,z=1 aka the north pole
```

```
#print(p)
totalprod=p
lntotalprod=np.log(p)
berryphase=-lntotalprod.imag
print(berryphase/np.pi)
#result -0.2503576537911302 expected -0.25
```