Problem Set 6 problems

Question 1: Constructing hadrons states using SU(3) raising and lowering operators (25 points)

Learning objectives

In this question you will:

- Learn how to find the SU(3) wave functions for baryons using SU(3) raising and lowering operators

If we know the SU(3) wave function for one member of an SU(3) multiplet, we can construct the other members of the multiplet using SU(3) raising and lowering operators. Each multiplet has well defined properties under the interachange of two quarks. For example, the spin-3/2 baryon decouplet that contains the Δ is symmetric under interchange, eg $\Delta^+ = \frac{1}{\sqrt{3}}(uud + udu + duu)$, while the spin-1/2 octet that contains the proton and neutron is antisymmetric under the interchange of the first two quarks but has no symmetry with respect to the third, eg $p = \frac{1}{\sqrt{2}}(ud - du)\,u$.

1a.

For the spin-3/2 decouplet, start with the $\Omega^- = sss$ baryon with strangeness $S = -3$. Use raising and/or lowering operators to construct the members of the decouplet with $S = -2$, known as the the Ξ, and with $S = -1$, known as the Σ^*.

Write your answer here

1b.

Starting from the proton, use raising and/or lowering operators to construct the members if the octet with $S = -1$, known as the Σ.

Write your answer here

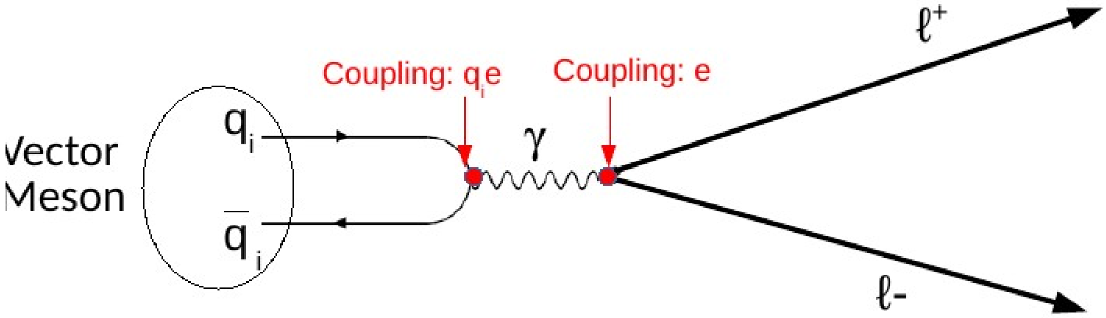
Question 2: Leptonic decays of vector mesons (25 points)

Learning objectives

In this question you will:

- Learn how neutral mesons couple to the photon

Neutral vector mesons can decay into lepton pairs through $q\overline{q}$ annihiliation via a virtual photon.



This annihilation requires that the wave function of the q and $\bar{q}$ overlap ($\psi(r=0) \neq 0$), so annihilation is only possible for states with $\ell = 0$. Because the photon has spin 1 and angular momentum must be conserved at the interaction vertices, the sum of the quark spins must have spin 1 (since $\ell = 0$). The lowest lying octet vector meson states have exactly this quark configuration

In general, these decays have small branching ratios, since they are occur via the electromagnetic rather than the strong interaction. We can calculate the partial width for electromagnetic decays of vector mesons in terms of the quark content of the mesons and the $q\bar{q}$ wave function overlap $|\psi(0)|^2$. The result is:

$$\Gamma(V \to \ell^+\ell^-) \propto \frac{\alpha^2 Q^2}{M_V^2} |\psi(0)|^2$$

where α is the fine structure constant, M_V is the mass of the meson and $\psi(0)$ is the wave function at the origin. The Q^2 abover represents the value of $\left|qe^2\right|^2$ and is discussed below.

Because annihilation into a photon can only occur if the quark and anti-quark in the meson are of the same flavor ($u\bar{u}$, $d\bar{d}$ or $s\bar{s}$) we can write the relevant meson wave function $|\psi\rangle$ in terms of the quarks as:

$$|\psi\rangle = \sum_i a_i \, |q_i \bar{q}_i\rangle$$

With this definition, and noting that our initial $q\bar{q}$ is annihilating into a photon $\langle\gamma|$, expectation value of Q^2 can be written:

$$\begin{aligned} Q^2 &= |\langle \gamma \,|\mathcal{Q}|\, \psi \rangle|^2 \\ &= \left| \sum_i a_i \, \langle \gamma \,|\mathcal{Q}|\, q_i \bar{q}_i \rangle \right|^2 \end{aligned}$$

is the squared sum of the charges of the quarks in the meson (weighted by a_i, the appropriate SU(3) Clebsch-Gordon Coefficients), The charge operator $\mathcal{Q}$ has the following effect on quark-antiquark pairs:

$$\begin{aligned} \langle\gamma| \, \mathcal{Q} \, |u\bar{u}\rangle &= \frac{2}{3} \\ \langle\gamma| \, \mathcal{Q} \left|d\bar{d}\right\rangle &= -\frac{1}{3} \\ \langle\gamma| \, \mathcal{Q} \, |s\bar{s}\rangle &= -\frac{1}{3} \end{aligned}$$

Using the quark model assignments for the vector mesons

$$\begin{aligned} \rho^0 &= \frac{1}{\sqrt{2}} \left(u\bar{u} - d\bar{d}\right) \\ \omega^0 &= \frac{1}{\sqrt{2}} \left(u\bar{u} + d\bar{d}\right) \\ \phi^0 &= s\bar{s} \end{aligned}$$

calculate the ratios of the leptonic decay widths for these 3 mesons. For this part of the problem, use SU(3) symmetry to make the approximation that $\psi(0)$ is the same for all 3 states. Use the Particle Data Book to compare your result to the measured values of these ratios.

Question 3: Hyperfine splitting in the meson multiplets (20 points)

Learning objectives

In this question you will:

- Use concepts first learned in the context of the hydrogen atom to understand mass splittings between vector and pseudoscalar mesons

The lowest lying pseudoscalar and vector multiplets are both SU(3) nonets (octet plus singlet) with no orbital angular momentum. The mass splittings within a multiplet come from the difference in mass between the u, d and s quarks and to a smaller extent from electromagnetic corrections due to the differences in quark charges. Mass splitting between the pseudoscalar and vector multiplets is due to QCD hyperfine splitting. The origin of this term is the same as hyperfine splitting in hydrogen: the interaction between the magnetic moments of the two particles in the bound state.

The hyperfine mass splitting for mesons with quark content $q_1\overline{q_2}$ has the form:

$$\Delta M(\text{meson}) = A\frac{\mathbf{S}_1 \cdot \mathbf{S}_2}{m_1 m_2}|\psi(0)|^2$$

where A is a constant, $\mathbf{S}_i$ is the spin of quark i, m_i is the mass of quark i and $|\psi(0)|$ is the wave function at the origin. We cannot calculate $\psi(0)$ from first principles but we can use experimental measurements to determine the combination $A|\psi(0)|^2$. For this problem, use $m_u = m_d = 336$ MeV and $m_s = 509$ MeV.

3a.

By comparing the masses of the K^0 and K^{0*} mesons, estimate the value of $A|\psi(0)|^2$.

3b.

Use this value of $A|\psi(0)|^2$ and the difference between the up and strange quark masses to predict the splitting between the η and ω. Note: you may assume that the η is a pure SU(3) octet state:

$$\eta = \frac{1}{\sqrt{6}}\left(u\overline{u} + d\overline{d} + 2s\overline{s}\right)$$

In the case of the ω, use the wave function given in problem 2.

Question 4: Rutherford Scattering (30 points)

Learning objectives

In this question you will:

- -Review the relationship between impact parameter and scattering angle for non-relativistic Rutherford Scattering in Classical Mechanics - Use the Ruthering Scattering formulae to calculate the cross section into a defined solid angle

4a.

This problem describes scattering in the non-relativistic limit using Classical Mechanics. Although most of our problems in particle physics need Relativistic Quantum Mechanics, starting with a classical problem provides us with much needed intuition.

Consider a beam of particles of charge e moving in the $+z$-direction. Each particle has kinetic energy E. The beam is incident on a target consisting of a single nucleus of charge Ze. As shown on slide 6 of the pdf file for lecture 11, the angle of deflection θ that the particle undergoes is related to its impact parameter b by the expression:

$$b = \frac{Ze^2}{8\pi\epsilon_0 E}\cot\frac{\theta}{2}$$

A real Rutherford scattering experiment would have a finite sized target. To keep things simple, for this problem we will focus on a single nucleus. That menas the beam size and intensities given below have been adjusted to match this artificial situation.

Assume that the beam is centered on the target and has a circular cross sectional area with radius R. The beam particles are uniformly distributed within the beam. Take the beam current to be 10^{-12} Amps. We will use a gold target ($Z = 79$), and a beam with $E = 6$ MeV. We'll use an unrealistically small beam size, comparable to the size of a gold atom: $R = 10^{-12}$ m. Assume that the beam is turned on for 1 second.

Calculate how many particles are incident on the target and make a histogram with this number of entries that shows the distribution of the scattering angle θ (in radians). Display the histogram with a logrithmic y-axis.

When solving this problem you are free to use the code below, but are not required to do so. The following function returns an array of $N \quad [x, y]$-positions of the particles within the beam. (The impact parameter is $b = \sqrt{x^2 + y^2}$.)

In [3]:

```
import numpy as np
import random
import math
import matplotlib.pyplot as plt

def makeData(N,R,seed):
    np.random.seed(seed)
    xout = []
    yout = []
    numToGen=int(1.1*(math.pi)*N)
    i = 0
    xin = np.random.uniform(-1.0,1.0,numToGen)
    yin = np.random.uniform(-1.0,1.0,numToGen)
    rsq = xin*xin+yin*yin
    while len(xout)<N  and i<numToGen:
        if rsq[i] < 1:
            xout.append(R*xin[i])
            yout.append(R*yin[i])
        i += 1
    return np.array(xout), np.array(yout)
```

You may use following definitions of physical constants and the parameters:

In [4]:

```
# Constants
# atomic number for gold
Z = 79

# Charge of the electron in Coulombs
e = 1.602e-19

#Kinetic energy of a beam particle in eV
EineV = 6.0e6

# Translation of the kinetic  energy to Joules
E = EineV*e

# E&M epsilon_0
epsilon0 = 8.854e-12
```

In [5]:

```
# Noted by the the announcment, use current = 10^-13 Amps instead
I = 10**(-13)  # A

R = 10**(-12)  # m

# I = C/s, so to get# particle:
time = 1 #s
N_electron = int(I * time/e)
```

In [6]:

```
xout, yout = makeData(N_electron, R, seed = 23456)

b = np.sqrt(xout**2 + yout**2)
```

In [7]:

```
# using formula for b, getting theta:

# assuming (x,y) was given in meters:

def get_theta(Z, e, epsilon, E, b):
    inside = Z*(e**2)/(8*np.pi*epsilon*E*b)
    return 2*np.arctan(inside)
```

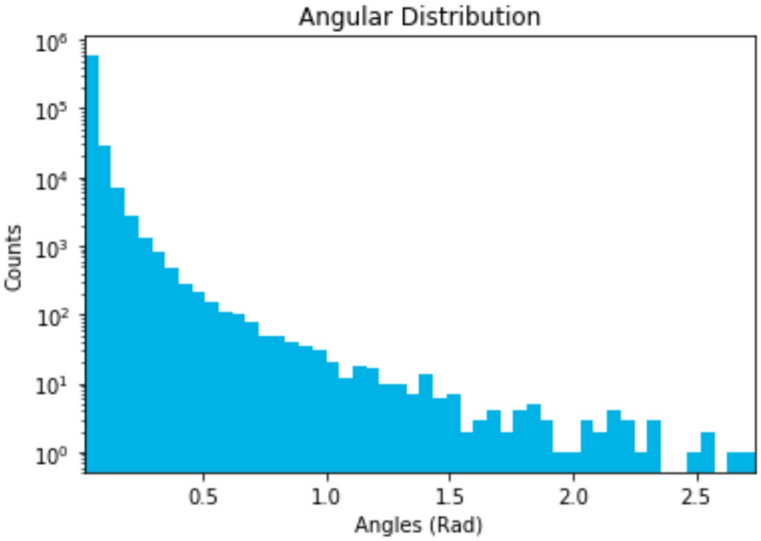
In [8]:

```
theta = get_theta(Z, e, epsilon0, E, b)
```

In [17]:

```
def plot_histogram(data,xtitle,ytitle, title, nbins, den, limits):

    #Plot the histogram of the sampled data with nbins and a nice color
    n, bins, patches =plt.hist(data, bins=nbins, range=limits, density
= den, color=(0,0.7,0.9))  #Set the color using (r,g,b) values or
                                                                #  us
e a built-in matplotlib color"""
    bincenters = 0.5*(bins[1:]+bins[:-1])
    #errs = np.sqrt(n)

    #plt.errorbar(bincenters, n, yerr=errs, fmt='none')
    #Add some axis labels and a descriptive title
    plt.xlabel(xtitle)
    plt.ylabel(ytitle)
    plt.title(title)
    plt.yscale('log')

    #Get rid of the extra white space on the left/right edges (you can
delete these two lines without a problem)
    xmin, xmax, ymin, ymax = plt.axis()
    plt.axis([limits[0],limits[1],ymin,ymax])

    #Not necessarily needed in a Jupyter notebook, but it doesn't hurt
    plt.show()
    return n, bins, patches
```

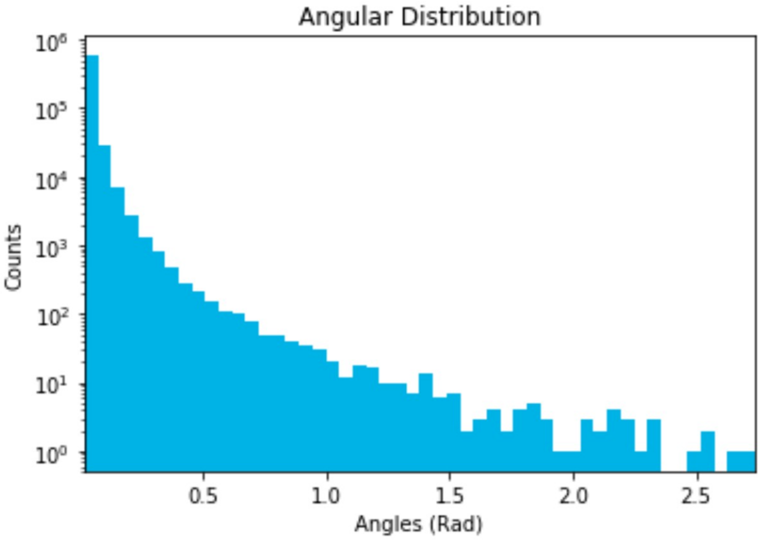
```
In [31]: nbins = 50
         lims = [np.min(theta), np.max(theta)]
         n, bins, patches = plot_histogram(theta, "Angles (Rad)", "Counts", "Angular Distribution", nbins, den = False, limits = lims)
```



4b.

A detector that subtends a solid angle $\Delta\theta = \frac{\pi}{8}$ and $\Delta\phi = \frac{\pi}{2}$ centered at $\theta = \frac{\pi}{4}$ and $\phi = \pi$. How many particles hit the detector? Note: you are expected to answer this question using your code from part (a); you are not expected to do an analytic calculation.

```
In [38]: n, bins, patches = plot_histogram(filter1, "Angles (Rad)", "Counts", "Angular Distribution", nbins, den = False, limits = lims)
```



In [33]:

```
filter1
```

Out[33]:

```
array([[0.01962886],
       [0.03946534],
       [0.01948528],
       ...,
       [0.02113531],
       [0.06212197],
       [0.02115912]])
```

In [42]:

```
# dOmega = d(cos(theta)) dPhi
dPhi = np.pi/2
dtheta = np.pi/8

phi0 = np.pi
theta0 = np.pi/4

theta_range = [(theta0 - dtheta/2), (theta0 + dtheta/2)]


filter1 = theta[np.argwhere(theta <= theta_range[1])]
filter2 = filter1[np.argwhere(filter1 >= theta_range[0])]

# 1/2 comes from the phi angle

final = len(filter2)/2

print("Number = ", final)
```

```
Number =  215.0
```

4c.

Using the expression for cross section, find the cross section

$$\sigma = \int \frac{d\sigma}{d\Omega} d\Omega$$

where the integral corresponds to the solid angle of the detector.

In [43]:

```
# dtheta/dOmega equation

def get_diff_angle(Z, e, epsilon, E, theta):
    numerator = (Z*(e**2))**2
    denom = (16*np.pi* epsilon * E * (np.sin(theta/2))**2 )**2

    return numerator/denom
```

In [46]:

```
import scipy.integrate as integrate
import scipy.special as special
```

In [52]:

```
# dOmega = sin(theta) * dtheta * dphi
dphi = np.pi/2

integral_result = integrate.quad(lambda x: np.sin(x)*get_diff_angle(Z,
e, epsilon0, E, x)*dphi,
                                 theta_range[0], theta_range[1])

cross_section = integral_result[0]

print("Cross section = ", cross_section, "m^2")
```

```
Cross section =  5.198848859486806e-28 m^2
```

In [51]:

```
# Compare with total cross section:
total_cross_section = np.pi*(R**2)
print('Total cross section:', total_cross_section, 'm^2')
```

```
Total cross section: 3.141592653589793e-24 m^2
```

In [50]:

```
print("As we can see, cross section < total cross section; this sanity
check helps confirm the possibility of the cross section value")
```

```
As we can see, cross section < total cross section; this sanity check
helps confirm the possibility of the cross section value
```

In [ ]: