Problem 1. Anderson 5.9

A narrow beam of gamma rays passes through 2.0 cm of lead. The incident beam consists of 30% 0.4 MeV photons and 70% 1.5 MeV photons. What fraction of the incident fluence is transmitted? Use Figure 5.5.

In addition to what's asked for in the question, find the effective attenuation coefficient.

Solution

$$MAC = \begin{cases} 3\times10^{-2}, & E_\gamma = 0.4\,MeV \\ 5\times10^{-3}, & E_\gamma = 1.5\,MeV \end{cases} \qquad \rho = 1.134\times10^4\ kg/m^3$$

$$\phi_a' = \phi_a e^{-\mu_a x} = \phi_a e^{-MAC_a \rho x} = 0.3 \cdot \phi_0 e^{-MAC_a \rho x}$$

$$\frac{\phi_a'}{\phi_0} = 0.3\, e^{-(3\times10^{-2}\frac{m^2}{kg}\cdot 1.134\times10^4\frac{kg}{m^3}\cdot 2cm\cdot 10^{-2}\frac{cm}{m})}$$

$$= 0.000333$$

$$\frac{\phi_b'}{\phi_0} = 0.7\, e^{-(5\times10^{-3}\frac{m^2}{kg}\cdot(1.134\times10^4\frac{kg}{m^3}\cdot 0.02m)}$$

$$= 0.225221$$

$$\frac{\phi_0'}{\phi_0} = 0.22553$$

$$\mu_{eff} = \frac{\ln\frac{\phi'}{\phi}}{-x} = \boxed{74.46\,m^{-1}}$$

Problem 2. Anderson 5.10

A narrow beam of neutrons passes through 2.0 cm of cadmium. The incident beam consists of 60% 0.02 MeV neutrons and 40% 0.5 MeV neutrons. What fraction of the incident fluence is transmitted? Use the information on Figure 5.6.

Solution $\rho_{Cd} = 8.65\ g/cm^3 * 10^3 \frac{cm^3}{m^3}\frac{kg}{g} = 8.65\times10^3\ kg/m^3$

$x = 2cm = 0.02m$

	E(MeV)	f	MAC (m^2/kg)
a	0.02	0.6	5×10^{-3}
b	0.5	0.4	5×10^{-3}

$$\phi = \phi_0 e^{-MAC\cdot\rho\cdot x}$$

$$\frac{\phi}{\phi_0} = e^{-(5\times10^{-3}\ m^2/kg \cdot 8.65\times10^{3}\ kg/m^3 \cdot 0.02m)}$$

$$\boxed{= 0.421}$$

Problem 3. Anderson 5.14

Calculate the dose for a 100 R exposure measured in muscle tissue and bone at 18 keV (Mo – K_α), 140 keV (^{99m}Tc), and 1.25 MeV (^{60}Co) from the information on Figure 5.14. Assume that electronic equilibrium holds at the point of consideration.

Solution

$D = bK$

	b (18 keV)	b (140 keV)	b (1.25 MeV)
muscle	1.0	1.0	1.0
bone	4.0	1.0	1.0

D	muscle	bone
18 keV	100	400
140 keV	100	100
1.25 MeV	100	100

Problem 4.

Calculate the flux of epithermal neutrons needed to deliver a dose rate of 0.1 Gy s^{-1} to muscle (tissue). Use an energy of 0.1 MeV to represent the average energy of epithermal neutrons.

Solution

$T = 0.1$ MeV

Tissue

$\phi = ?$

from fig 5.9

$$\left(\frac{K}{\phi}\right)_{10^{-1}\,MeV} * \phi = D$$

$$\frac{D}{K/\phi} = \frac{0.1\ \frac{Gy}{s}}{500 \times 10^{-18}\ Gy \cdot m^2}$$

$$\boxed{\phi = 2 \times 10^{14}\ m^{-2} s^{-1}}$$

Problem 5. Anderson 6.4

What is the angle of scatter and the energy of a Compton electron when the incident photon energy is 140 keV and the angle of scatter of the photon is 60°?

Solution

Please see attached code for calculations

$\theta_e' = 53.7°$

$T_e' = 16.87\ \text{keV}$

Problem 6.

Calculate the Compton edge energies (max scattered electron energy) for the following isotopes:

(a) ^{54}Mn

(b) ^{137}Cs

(c) ^{22}Na (ignore the positron annihilation gammas)

Solution

Please see attached code for calculations

	T_e^{max} (keV)
^{54}Mn	639.2
^{137}Cs	477.3
^{22}Na	1062

Problem 7.

Using the photon energy from a ^{137}Cs decay, calculate the following:

(a) The Klein Nishina total scattering cross section

(b) The total atomic cross section for Compton scattering in lead

(c) The Compton scattering attenuation coefficient in lead

Solution

Please see attached code for work

a] $$\sigma_{KN} = \pi r_0^2 \left[\frac{2(1+\alpha)}{\alpha^2}\left(\frac{2(1+\alpha)}{1+2\alpha} - \frac{\ln(1+2\alpha)}{\alpha} \right) + \frac{\ln(1+2\alpha)}{\alpha} - \frac{2(1+3\alpha)}{(1+2\alpha)^2} \right]$$

$$\sigma_{KN} = 0.256 \text{ b}$$

b] $$\sigma_{Compton} = Z * \sigma_{KN}$$

$$\boxed{= 21 \text{ b}}$$

c] $$\left(\frac{\mu}{\rho}\right)_{is} = n_m \sigma_{is} \Rightarrow \mu_{is} = \frac{N_A Z}{M_m} \rho \sigma_{is}$$

$$\mu \boxed{= 56.78 \text{ cm}^{-1}}$$

Problem 8.

Calculate the values for $\frac{d\sigma_{KN}}{dT_e}$ versus T_e assuming an incoming photon energy of 0.5 MeV. Calculate the values between $T_e = 0$ and $T_e = T_{max}$ in step sizes of 0.02 MeV. Plot your results and compare with figure 6.7

Solution

Please see attached code & graph

Strongly resembles fig. 6.7, though without drawing vertical edge (due to plot difficulty)

In [16]:
```
import scipy.constants as const
import numpy as np
```

In [47]:
```
def alpha(E_gamma=None, nu=None):
    """ Should specify E_gamma in eV
    """
    mecsquared = const.value('electron mass energy equivalent in MeV')
* 10**6
    h = const.value('Planck constant in eV s')
    if E_gamma is None and nu is None:
        raise exception("Must specify photon properties")
    elif nu is None:
        a = E_gamma / mecsquared
    elif E_gamma is None:
        a = h * nu / mecsquared
    else:
        raise exception('Please only specify one photon property')

    return a
```

In [48]:
```
def T_e_max(E_gamma=None, nu=None):
    """ Specify E_gamma in eV
    """
    h = const.value('Planck constant in eV s')
    if E_gamma is None and nu is None:
        raise exception("Must specify photon properties")
    elif nu is None:
        a = alpha(E_gamma=E_gamma)
        t = E_gamma * 2.0 * a / (1.0 + 2.0 * a)
    elif E_gamma is None:
        a = alpha(nu=nu)
        t = h * nu * 2.0 * a / (1.0 + 2.0 * a)
    else:
        raise exception('Please only specify one photon property')

    return t
```

In [49]:
```
def theta_e(T_gamma, theta_gamma):
    a = alpha(E_gamma=T_gamma)
    inner = (1.0 + a) * np.tan(theta_gamma / 2.0)
    return np.arctan(1.0 / inner)
```

In [50]:
```
def T_e(T_gamma, theta):
    a = alpha(E_gamma=T_gamma)
    numerator = T_gamma * a * (1.0 - np.cos(theta))
    denominator = 1.0 + a * (1.0 - np.cos(theta))

    return numerator / denominator
```

Problem 5: Anderson 6.4

In [51]:
```
theta_prime = 60.0 * np.pi / 180.0
```

In [52]:
```
T_g = 140000
```

In [53]:
```
alpha(E_gamma=T_g)
```

Out[53]: 0.27397316778927894

Calculate the angle of scatter of the Compton electron

In [56]:
```
theta_e(T_g, theta_prime) * 180.0 / np.pi
```

Out[56]: 53.664449190568014

In [57]:
```
T_e(T_g, theta_prime)
```

Out[57]: 16867.500476176854

Problem 6

References:

- Mn-54 (http://www.nucleide.org/DDEP_WG/Nuclides/Mn-54_tables.pdf) gamma energy 834.855 keV
- Cs-137 (http://www.nucleide.org/DDEP_WG/Nuclides/Cs-137_tables.pdf) gamma energy 661.659 keV
- Na-22 (http://www.nucleide.org/DDEP_WG/Nuclides/Na-22_tables.pdf) gamma energy 1274.577 keV

Mn-54:

In [58]:
```
T_e_max(E_gamma=834855)
```

Out[58]: 639225.9473887982

In [59]:
```
T_e_max(E_gamma=661659)
```

Out[59]: 477335.86413384555

In [60]:
```
T_e_max(E_gamma=1274577)
```

Out[60]: 1061742.0485465585

Problem 7

In [61]:
```
def sigma_kn(E_gamma):
    r = const.value('classical electron radius') * 100 # convert from
m to cm

    a = alpha(E_gamma=E_gamma)

    answer = np.pi * r * r

    term1 = 2.0 * (1.0 + a) / a ** 2
    term1 *= (2.0 * (1.0 + a) / (1.0 + 2.0 * a) - np.log(1.0 + 2.0 * a
) / a)
    term1 += np.log(1.0 + 2.0 * a) / a
    term1 -= 2.0 * (1.0 + 3.0 * a) / (1.0 + 2.0 * a) ** 2

    return answer * term1
```

Part (a)

In [62]:
```
E = 661659.0 # eV
```

In [63]:
```
sigma_kn(E)
```

Out[63]: 2.5619923244178853e-25

Convert from cm^2 to b

In [64]:
```
sigma_kn(E) * 10**24
```
Out[64]: 0.25619923244178855

Part (b)

From the notes, $\sigma_{atomic} = \sigma_{Compton} = Z * \sigma_{KN}$

In [67]:
```
sigma_kn(E) * 10**24 * 82.0
```
Out[67]: 21.008337060226662

Part (c)

$(\mu/\rho)_{is} = n_m \sigma_{is}, n_m = \frac{N_A Z}{M_m}$

In [68]:
```
rho = 11.34 # g/cm^3
n_m = const.Avogadro * 82.0 / 207.2
```

In [70]:
```
n_m * sigma_kn(E) * 82.0 * rho
```
Out[70]: 56.77795443140117

Problem 8

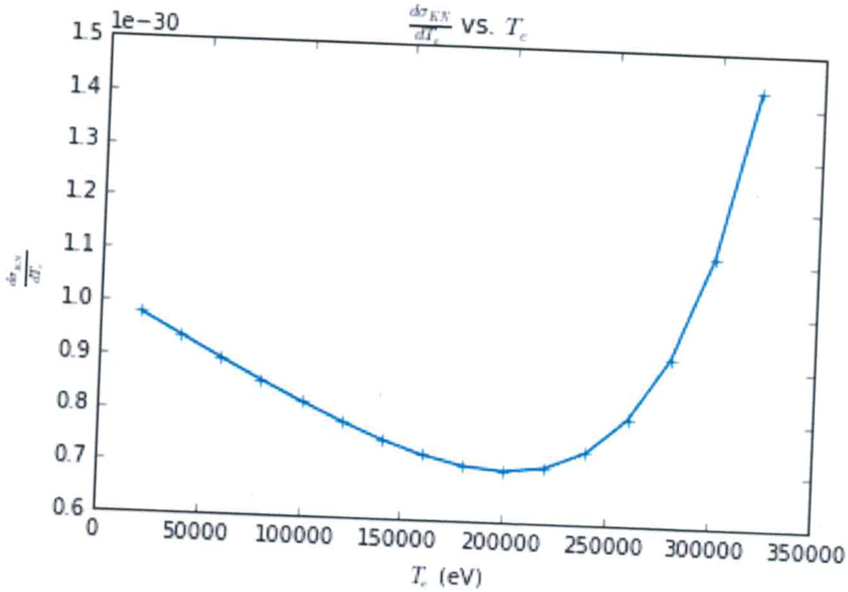
In [72]:
```
def sigma_prime(T_e, T_gamma=500000):
    r = const.value('classical electron radius') * 100 # convert to cm
    a = alpha(E_gamma=T_gamma)

    constant_part = np.pi * r * r / (a * T_gamma)

    other_part = T_e ** 2 / (T_gamma - T_e) ** 2
    other_part *= (1 / a ** 2 + (T_gamma - T_e) / T_gamma - 2 * (T_gam
ma - T_e) / (a * T_e))
    other_part += 2.0

    return other_part * constant_part
```

In [73]:
```
%matplotlib inline
```

In [74]:
```
import matplotlib.pyplot as plt
```

In [75]:
```
T_in = np.arange(20000, T_e_max(E_gamma=500000), 20000)
```

In [81]:
```
plt.plot(T_in, sigma_prime(T_in), 'b-+')
plt.ylabel(r'$\frac{d\sigma_{KN}}{dT_e}$')
plt.xlabel(r'$T_e$ (eV)')
plt.title(r'$\frac{d\sigma_{KN}}{dT_e}$ vs. $T_e$')
plt.show()
```



In [ ]: