tmm_final

December 18, 2020

[1]:
```python
import numpy as np
import matplotlib.pyplot as plt
from tmm_final import *
```

0.1 Single Layer Coating

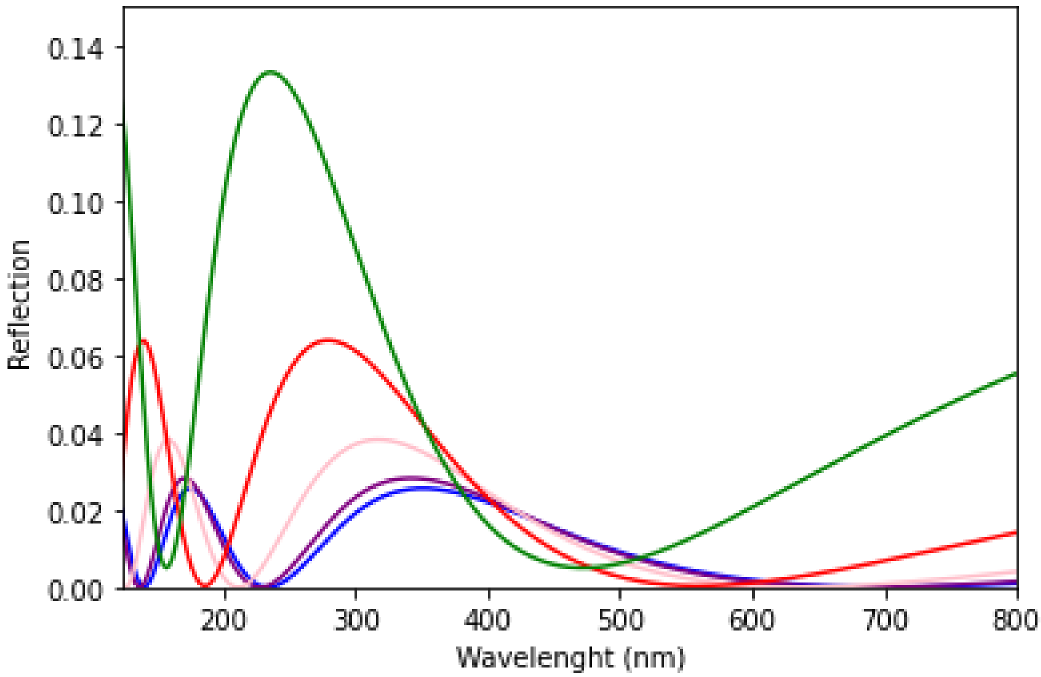
For the case of a single layer of anti-reflecting coating, we want to have a quarter-wave thickness as well as [1]:

$$n_{film} = \sqrt{n_{substrate}}$$

Thus, if we want to eliminate 600nm light and use, for example, a MgF_2 substrate with $n = 1.38$ our film should have an index of refraction of $n = 1.17$:

[2]:
```python
degree = np.pi/180

lamb = 600

n_list = [1, 1.17, 1.38]

d_list = [np.inf, lamb/4, np.inf]

wl = np.linspace(50,1000,num=400)

R0=[]
R15=[]
R30=[]
R45=[]
R60=[]

for w in wl:

    R0.append(transfer_matrix_method('s', n_list, d_list, 0, w)['R'])
    R15.append(transfer_matrix_method('s', n_list, d_list, 15 * degree, w)['R'])
    R30.append(transfer_matrix_method('s', n_list, d_list, 30 * degree, w)['R'])
    R45.append(transfer_matrix_method('s', n_list, d_list, 45 * degree, w)['R'])
    R60.append(transfer_matrix_method('s', n_list, d_list, 60 * degree, w)['R'])
```

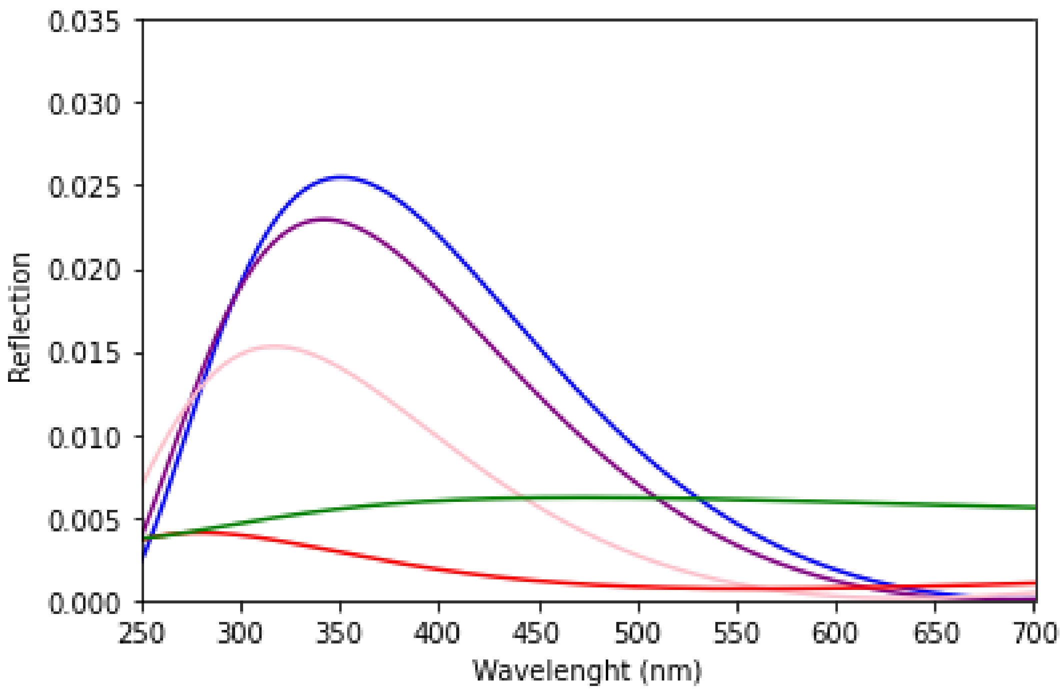
```
plt.figure()
plt.xlim(125, 800)
plt.ylim(0, 0.15)
plt.plot(wl,R0,'blue',wl,R15,'purple',wl,R30,'pink',wl,R45,'red',wl,R60,'green')
plt.xlabel('Wavelenght (nm)')
plt.ylabel('Reflection')
plt.show()
```



As we can see, the minima shifts with a changing angle of incidence, and light is nearly extinguished around the 600nm mark. We now repeat this process for TM or p-wave radiation:

[3]:
```
degree = np.pi/180

lamb = 600

n_list = [1, 1.17, 1.38]

d_list = [np.inf, lamb/4, np.inf]

wl = np.linspace(50,1000,num=400)

R0=[]
R15=[]
R30=[]
R45=[]
```

```
R60=[]

for w in wl:

    R0.append(transfer_matrix_method('p', n_list, d_list, 0, w)['R'])
    R15.append(transfer_matrix_method('p', n_list, d_list, 15 * degree, w)['R'])
    R30.append(transfer_matrix_method('p', n_list, d_list, 30 * degree, w)['R'])
    R45.append(transfer_matrix_method('p', n_list, d_list, 45 * degree, w)['R'])
    R60.append(transfer_matrix_method('p', n_list, d_list, 60 * degree, w)['R'])

plt.figure()
plt.xlim(250, 700)
plt.ylim(0, 0.035)
plt.plot(wl,R0,'blue',wl,R15,'purple',wl,R30,'pink',wl,R45,'red',wl,R60,'green')
plt.xlabel('Wavelenght (nm)')
plt.ylabel('Reflection')
plt.show()
```



Although there is a clear drop in reflectance around 600nm, the overall reflectance is around seven times less than for s-waves.

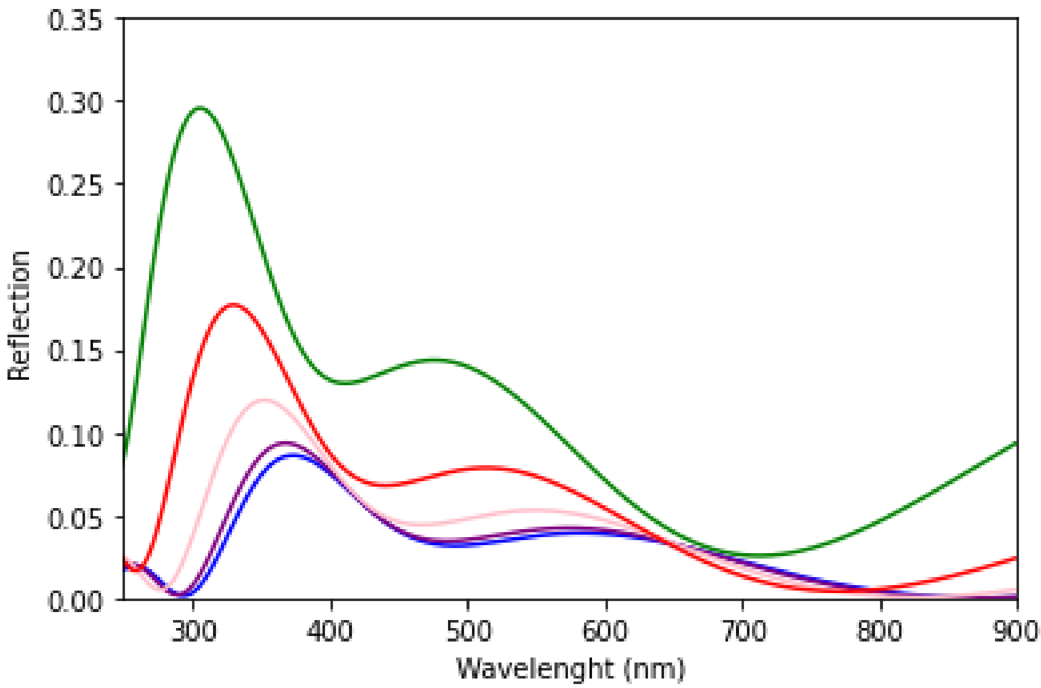
0.2 Double Layer Coating

Let us now examine double layer anti-reflection coatings. As before, we will examine quarter-quarter-wave layer thicknesses that also obey:

$$\left(\frac{n_1}{n_2}\right)^2 = \frac{n_{air}}{n_{substrate}}$$

If we use glass with and index of refraction of 1.5 as a substrate, assume that $n_{air} = 1$, and re-use MgF_2 a our first layer, the second material must have $n_2 = 1.69$. Lets see how this reflects 600nm light:

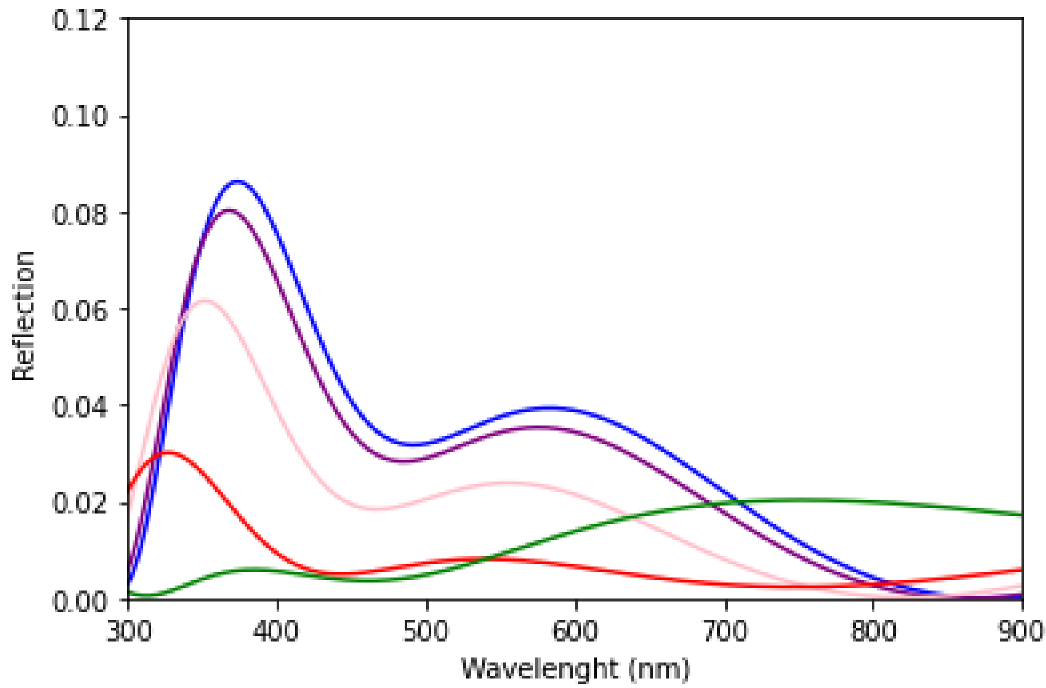
[4]:
```
degree = np.pi/180

lamb = 600

n_list = [1, 1.38, 1.69, 1.50]

d_list = [np.inf, lamb/4, lamb/4, np.inf]

wl = np.linspace(50,1000,num=400)

R0=[]
R15=[]
R30=[]
R45=[]
R60=[]

for w in wl:

    R0.append(transfer_matrix_method('s', n_list, d_list, 0, w)['R'])
    R15.append(transfer_matrix_method('s', n_list, d_list, 15 * degree, w)['R'])
    R30.append(transfer_matrix_method('s', n_list, d_list, 30 * degree, w)['R'])
    R45.append(transfer_matrix_method('s', n_list, d_list, 45 * degree, w)['R'])
    R60.append(transfer_matrix_method('s', n_list, d_list, 60 * degree, w)['R'])

plt.figure()
plt.xlim(250, 900)
plt.ylim(0, 0.35)
plt.plot(wl,R0,'blue',wl,R15,'purple',wl,R30,'pink',wl,R45,'red',wl,R60,'green')
plt.xlabel('Wavelenght (nm)')
plt.ylabel('Reflection')
plt.show()
```



For two layers, the reflectance drops to zero around 780nm rather than around the 600nm mark, signifying a shift of around 200nm.

[5]:
```
degree = np.pi/180

lamb = 600

n_list = [1, 1.38, 1.69, 1.50]

d_list = [np.inf, lamb/4, lamb/4, np.inf]

wl = np.linspace(50,1000,num=400)

R0=[]
R15=[]
R30=[]
R45=[]
R60=[]

for w in wl:

    R0.append(transfer_matrix_method('p', n_list, d_list, 0, w)['R'])
    R15.append(transfer_matrix_method('p', n_list, d_list, 15 * degree, w)['R'])
    R30.append(transfer_matrix_method('p', n_list, d_list, 30 * degree, w)['R'])
    R45.append(transfer_matrix_method('p', n_list, d_list, 45 * degree, w)['R'])
```

```
    R60.append(transfer_matrix_method('p', n_list, d_list, 60 * degree, w)['R'])

plt.figure()
plt.xlim(300, 900)
plt.ylim(0, 0.12)
plt.plot(wl,R0,'blue',wl,R15,'purple',wl,R30,'pink',wl,R45,'red',wl,R60,'green')
plt.xlabel('Wavelenght (nm)')
plt.ylabel('Reflection')
plt.show()
```



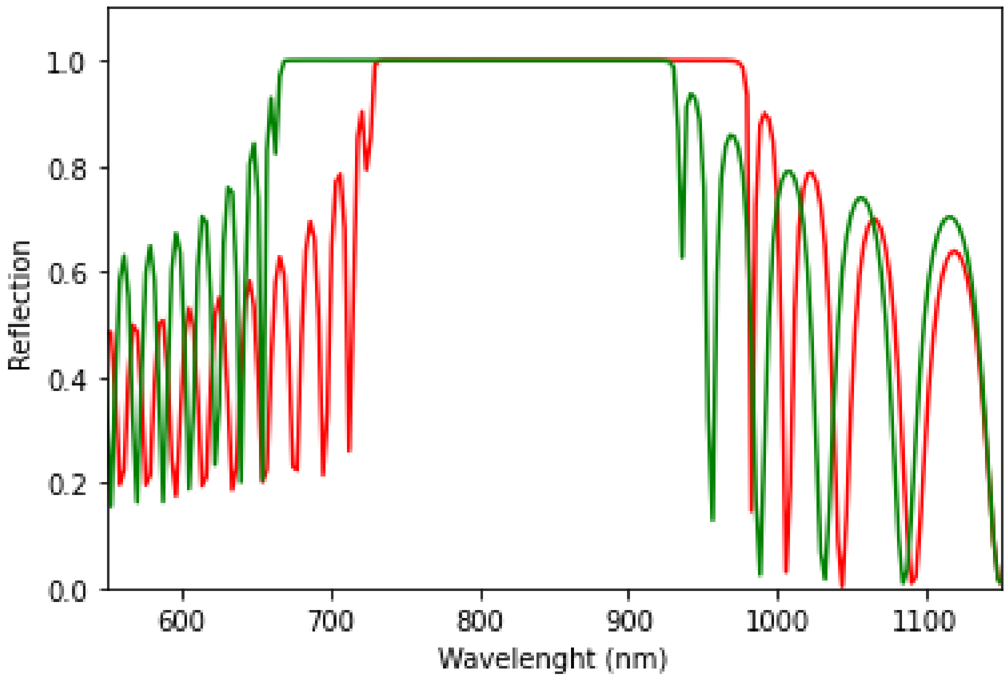
Similar to the single layer coating, the reflectance is much lower for p-waves than for s-wave, though the 200nm shift is still present.

0.3 Distributed Bragg Reflectors

Lets now try to reproduce a reflectance plot for a Distributed Bragg Reflector (DBR) using some available data. We will consider 20 layers, each one composed out of two materials such that $n_1 = 1.86$ and $n_2 = 1.27$. Furhtermore, we will be using quarter-wave layers on a 600nm beam once again, and a substratue with $n_{substrate} = 4.1$:

[6]:
```
degree = np.pi/180

lamb = 600
```

```
n_list = [1,1.86,1.27,1.86,1.27,1.86,1.27,1.86,1.27,1.86,1.27,1.86,1.27,1.86,1.
 ↪27,1.86,1.27
          ,1.86,1.27,1.86,1.27,1.86,1.27,1.86,1.27,1.86,1.27,1.86,1.27,1.86,1.27
          ,1.86,1.27,1.86,1.27,1.86,1.27,1.86,1.27,1.86,1.27,4.1]

d_list = [np.inf, lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/
 ↪4, lamb/4, lamb/4, lamb/4, lamb/4,
          lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/
 ↪4, lamb/4, lamb/4, lamb/4,
          lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/4, lamb/
 ↪4, lamb/4, lamb/4, lamb/4,
          lamb/4, lamb/4, lamb/4, lamb/4, np.inf]

wl = np.linspace(50,1500,num=500)

R0=[]
R15=[]
R30=[]
R45=[]
R60=[]

for w in wl:

    R0.append(transfer_matrix_method('s', n_list, d_list, 0, w)['R'])
    R15.append(transfer_matrix_method('s', n_list, d_list, 15 * degree, w)['R'])
    R30.append(transfer_matrix_method('s', n_list, d_list, 30 * degree, w)['R'])
    R45.append(transfer_matrix_method('s', n_list, d_list, 45 * degree, w)['R'])
    R60.append(transfer_matrix_method('s', n_list, d_list, 60 * degree, w)['R'])

plt.figure()
plt.xlim(550, 1150)
plt.ylim(0, 1.1)
plt.plot(wl,R45,'red',wl,R60,'green')
plt.xlabel('Wavelenght (nm)')
plt.ylabel('Reflection')
plt.show()
```



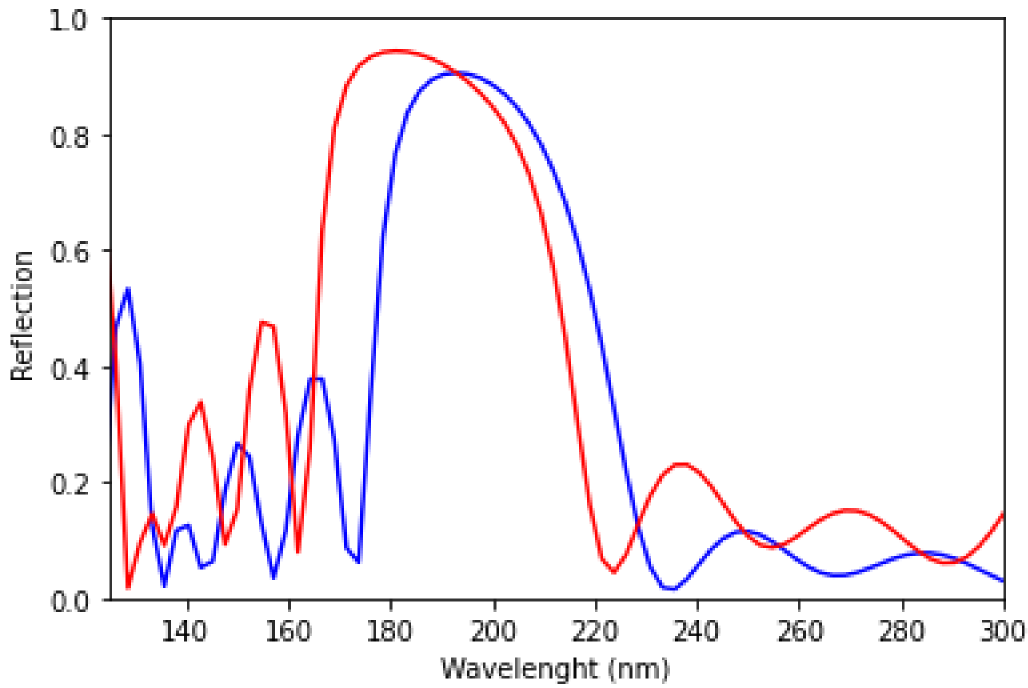
Even if the perfect reflectance region is not centered around 600nm as expected, the plot does exhibit the long flat region that is the hallmark of a DBR.

0.4 Additional DBR Plotting

As a final test of our program, we plot another DBR graph with some experimental data. The layers are not exactly quarter-wave plates, however seeing as how the curves seem to be centered around 200nm and some of the layers are 50nm, we can conclude that this is the target wavelenght.

[7]:
```
degree = np.pi/180

n_list = [1, 1.63, 2.74, 1.63, 2.74, 1.63, 2.74, 1.63, 2.74, 1.5]

d_list = [np.inf, 38, 50, 38, 50, 38, 50, 38, 50, np.inf]

wl = np.linspace(50,1000,num=400)

R0=[]
R45=[]
R60=[]

for w in wl:

    R0.append(transfer_matrix_method('s', n_list, d_list, 0, w)['R'])
    R45.append(transfer_matrix_method('s', n_list, d_list, 45 * degree, w)['R'])
```

```
    R60.append(transfer_matrix_method('s', n_list, d_list, 60 * degree, w)['R'])

plt.figure()
plt.xlim(125, 300)
plt.ylim(0,1)
plt.plot(wl,R0,'blue',wl,R45,'red')
plt.xlabel('Wavelenght (nm)')
plt.ylabel('Reflection')
plt.show()
```




```
[ ]:
```