Matrix Method – Solution to the Seminar Tasks

The task was to implement the matrix method for a stratified medium. The implemented functions should be tested with a Bragg mirror made of a sequence of alternating high and low refractive index layers.

1 Task 1 – Transfer Matrix of a Bragg Mirror

For this task a function `transfermatrix` had to be implemented and applied to the layer sequence of a Bragg mirror. The parameters of the Bragg mirror are defined as follows:

```
# %% task 1: transfer matrix %%
n1 = np.sqrt(2.25)
n2 = np.sqrt(15.21)
d1 = 0.13
d2 = 0.05
N = 5
epsilon, thickness = bragg(n1, n2,d1, d2, N)
```

Please note that both layers have a quarter-wave optical thickness at a wavelength of $\lambda = 780\,\mathrm{nm}$. Hence, this is the center wavelength of the high-reflectance region of the mirror. The permittivity and thickness vectors of the Bragg mirror are defined in a separate function named `bragg` for reusability:

```
def bragg(n1, n2, d1, d2, N):
'''Generates the stack parameters of a Bragg mirror
 The Bragg mirror starts at the incidence side with layer 1 and is
 terminated by layer 2

Parameters
----------
n1, n2 : float or complex
Refractive indices of the layers of one period
d1, d2 : float
Thicknesses of layers of one period
N : in
Number of periods
Returns
-------
epsilon : 1d-array
Vector containing the permittivities
thickness : 1d-array
Vector containing the thicknesses
'''
# find suitable type for epsilon
dt = np.common_type(np.array([n1]), np.array([n2]))
epsilon = np.zeros((2*N,), dtype = dt)
epsilon[::2] = n1**2
epsilon[1::2] = n2**2
thickness = np.zeros((2*N,))
thickness[::2] = d1
thickness[1::2] = d2
return epsilon, thickness
```

The command `epsilon = np.zeros((2*N,), dtype = dt)` allocates a row vector of length $2N$ for the layer permittivities. The same is done for the thicknesses. The index vectors `::2` and `1::2` are then used to set the even and the odd elements of the vectors to the desired values, respectively. The thickness and epsilon vectors are passed to the `transfermatrix` function. The remaining parameters of the function were given by $\lambda = 780\,\mathrm{nm}$, $k_z = 0$ (i.e. normal incidence) and TE-polarization:

```
# %% task 1: transfer matrix %%
wavelength = 0.78
kz = 0.0
polarisation = 'TE'
M = transfermatrix(thickness, epsilon, polarisation , wavelength, kz)
print('M = {0}'.format(M))
```

The result of the last command should be similar to

```
M =

   -0.0084    0.0000
   -0.0000 -118.8138
```

The matrix is extremely asymmetric. But the determinant is unity as expected for a system without absorption:

```
print('det(M) = {0}'.format(np.linalg.det(M)))
```

```
det(M) = (1+0j)
```

The eigenvalues of the transfer matrix are real:

```
print('eig(M) = {0}'.format(np.linalg.eig(M)[0]))
```

```
eig(M) = [-0.0084+0.j  -118.8138+0.j]
```

This indicates that the fields are heavily damped along their passage through the structure.

For a wavelength of $\lambda = 1200\,\mathrm{nm}$ the situation changes drastically:

```
M =
    1.0336    0.0033
   -0.5301    0.9658
```

```
det(M) = (0.999+0j)
```

Again the determinant is unity.

```
eig(M) = [0.999+0.0246j 0.999-0.0246j]
```

But the eigenvalues are now complex and their magnitude is unity. The wavelength of $\lambda = 1200\,\mathrm{nm}$ lies outside the photonic band gap of the mirror and hence light is transmitted.

2 Task 2 – Reflection and Transmission Coefficients of a Bragg Mirror

A separate function `spectrum` had to be implemented for the calculation of the spectral response of a layer stack. Using this function, the reflection and transmission coefficients of a Bragg mirror with 5 periods are calculated with the following commands:

```
epsilon, thickness = bragg(n1, n2, d1, d2, N)
n_in = 1
n_out = 1.5
angle_inc = 0
wavelength_vector = np.linspace(0.5, 1.5, 1001)
t, r, T, R = spectrum(thickness, epsilon, polarisation ,
wavelength_vector , angle_inc, n_in, n_out)
```

```
plt.figure()
plt.plot(wavelength_vector , T,
wavelength_vector , R)
plt.xlim(wavelength_vector[[0,-1]])
plt.ylim([0, 1])
plt.xlabel('wavelength  [m]')
plt.ylabel('reflectance, transmittance')
plt.legend(['transmittance', 'reflectance'], loc='center', frameon=False)
```

The commands create a plot of the power transmission an reflection coefficients similar to the one shown on the left of figure 1. Note the large reflection in the spectral domain between $\lambda = 600\,\mathrm{nm}$ and $\lambda =$

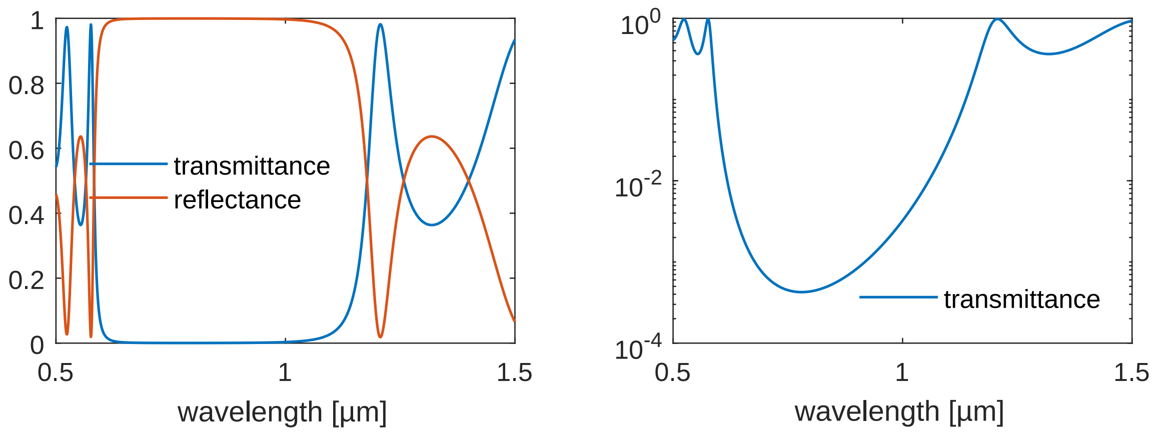

Figure 1: Spectral response of a Bragg mirror made of 5 layer pairs. The left plot show the reflectance and transmittance spectra. The right plot shows the transmittance spectrum on a logarithmic scale.

$1100\,\mathrm{nm}$. This region corresponds to the photonic band gap of the periodic Bragg structure for normal incidence. A logarithmic plot of the transmittance, that can be produced with the commands

```
plt.figure()
plt.semilogy(wavelength_vector , T)
plt.xlim(wavelength_vector[[0,-1]])
plt.xlabel('wavelength  [m]')
plt.ylabel('transmittance')
plt.legend(['transmittance'], loc='lower right', frameon=False)
```

is shown on the right of figure 1. The transmittance in the center of the gap is less than 10^{-3}. With a larger number of periods this value can be made even smaller.

The Bragg mirror can be modified to a Fabry-Perot resonator by introducing a defect that acts as a cavity. In this case twice as many layers are used, while the defect is created by increasing the thickness of a central layer to $\lambda/2$:

```
epsilon, thickness = bragg(n1,n2, d1, d2, 2*N)
thickness[2*N] = 2*thickness[2*N]
```

The defect causes a transmission peak in the middle of the gap of the mirror as shown in figure 2.

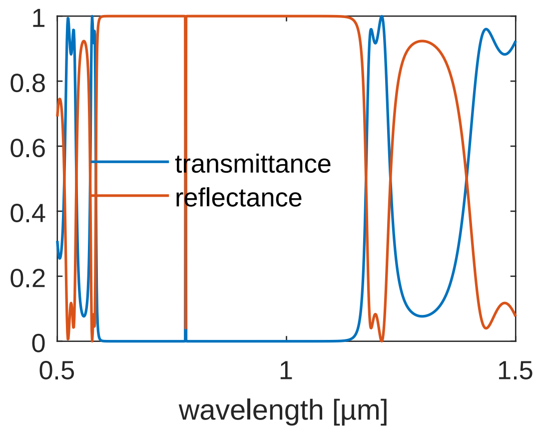

Figure 2: Spectral response of a Fabri-Perot resonator made of two Bragg mirrors with a half-wave low index layer in the center.

3 Task 3 – Field inside a Bragg Mirror

This task included the implementation of a function `field` to calculate the electromagnetic field inside the layer structure. Using this function, the field distribution of the Bragg mirror is calculated with the commands

```
epsilon, thickness = bragg(n1, n2, d1, d2, N)
l_in = 1.0
l_out = 1.0
Nx = 1000
wavelength = 0.78
kz = 0
f, index, x = field(thickness, epsilon, polarisation , wavelength, kz,
n_in ,n_out, Nx, l_in, l_out)
```

A plot of the magnitude of the electromagnetic field inside the layer stack is shown on the left of figure 3. It can be created with

```
plt.figure()
plt.plot(x, np.abs(f)/np.abs(f).max()*index.real.max(), x, index.real)
plt.xlim(x[[0,-1]])
plt.ylim([0, 1.1*index.real.max()])
plt.xlabel('x  [m]')
plt.ylabel('normalized field (magnitude)')
plt.legend(['EM field', 'refr. index'], loc='lower right', frameon=False)
```

The magnitude of the field is scaled to the same range as the refractive index to allow plotting of both in the same axes. In front of the mirror and within the layer stack the interference of the incident and the reflected wave creates a standing wave pattern. As the incident wavelength is within the photonic gap of

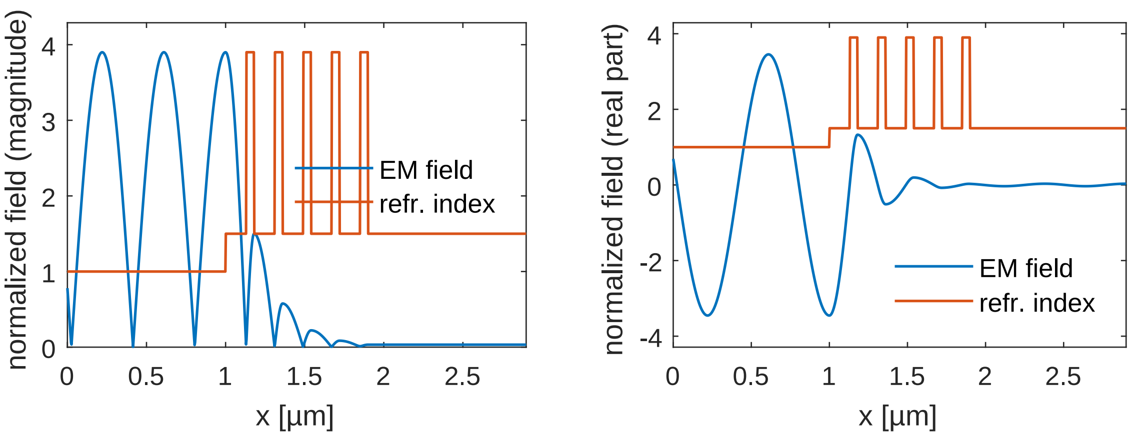

Figure 3: Field distribution inside a Bragg mirror made of 5 layer pairs. The left plot shows magnitude of the complex field. The right plot shows the real part of the complex field.

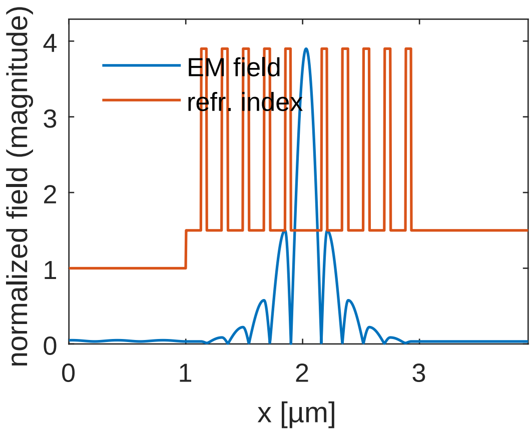

Figure 4: Field distribution inside a Fabri-Perot resonator made of two Bragg mirrors with a half-wave low index layer in the center.

the Bragg stack, the field decays exponentially within the structure. Only a small fraction is transmitted due to the finite number of layers.

Please remember that the observable field corresponds to the real part of the complex field and oscillates harmonically in time and along z:

$$f_{\mathrm{r}}(x,z,t) = \Re\left\{f(x)\cdot\exp\left(ik_z z - i\omega t\right)\right\}. \tag{1}$$

The real part of the field inside the Bragg mirror is shown on the right of figure 3.

In a similar fashion the field inside the Fabri-Perot resonator can be calculated. The magnitude of the field is shown in figure 4. The excitation wavelength of $\lambda = 780\,\mathrm{nm}$ corresponds to the resonance wavelength of the cavity. Figure 4 shows that this causes a strong field enhancement inside the cavity and a transmission through the structure.

4 Task 4 – Temporal Animation of the Field

The final task was to create a temporal animation of the field by implementing a function `timeanimation` that multiplies the field with the temporal phase factor $\exp(i\omega t)$ and plots the real part. In the test script the function is invoked with the field of the Fabri-Perot resonator:

```
steps = 200
periods = 10
ani = timeanimation(x, f, index, steps, periods)
```

The animation shows that a standing wave oscillates ("breathes") inside the cavity while the field propagates into the structure on the front side and is emitted from the rear side.