Industrial Photonics

Design of ARCs

Ariel Priarone

s274149

(a.y. 2022/2023)

Contents

1 Introduction 2

2 Assignment 2

3 Initialize the script 2

4 Reflectivity without ARC 3
4.1 $\theta = 0°$ . . . 3
4.2 $\theta = 30°$ . . . 3

5 ARC design - single layer 4
5.1 Wavelength . . . 4
5.2 Design procedure . . . 5
5.3 Compute the reflectivity - neglect chromatic dispersion . . . 5
5.4 Compute the reflectivity - consider chromatic dispersion . . . 7

6 DLARC design - double layer 9
6.1 Silica/Titania layers . . . 9
6.2 Other materials layers . . . 10
6.3 Comparison of designs . . . 12

7 Effective reflectivity 12

8 Tolerances 13

List of Figures

1 Typical irradiation spectrum of the sun . . . 4
2 Equivalent TL . . . 6
3 Different ARCs considering a single n . . . 7
4 n dependency on λ . . . 8
5 Different ARCs considering n dependency on λ . . . 9
6 Equivalent TL - DLARC . . . 9
7 Different materials n ratios . . . 11
8 Equivalent TL - DLARC . . . 11
9 Reflectivity for different designs . . . 13
10 Reflectivity for different manufacture deviations . . . 14

List of Tables

1 Common materials refractive indeces . . . 11
2 Min and effective reflectivity . . . 13

1 Introduction

In this assignment it is asked to study the reflectivity of a solar cell and design some anti-reflection coating (ARC) to improve the performance of the photovoltaic cell (PV). I expect a pretty high reflection, without coating, because the refractive index of the silicon is quite big compared with the one of air. The working principle of ARCs is that if the reflected light that is reflected back at the first air/material interface has a phase shift of 180°, and the same amplitude as the reflected light at that same interface, the two wave cancel out, so that, at a certain wavelength, a null value of reflectivity can be achieved.

2 Assignment

The points to accomplish in this assignment are

1. compute the reflectivity for unpolarized light at $\theta = 0°$ and at $\theta = 30°$
2. design a single layer ARC (SLARC)
3. design a double layer ARC (DLARC) using SiO_2 and TiO_2
4. compare the reflectivity with the effective reflectivity accounting for the spectral irradiance of the typical sunlight.
5. analyze the impact of 2% tolerance on the layer thicknesses

3 Initialize the script

In order to compute the solutions of this assignment, i used a python script, that i will include in this document. The following is just the preamble of that script:

```python
import matplotlib.pyplot as plt
import matplotlib.colors as color
import numpy as np
import matplotlib
import tikzplotlib

def tikzplotlib_fix_ncols(obj):
    """
    workaround for matplotlib 3.6 renamed legend's _ncol to_ncols, which breaks tikzplotlib
    """
    if hasattr(obj, "_ncols"):
        obj._ncol = obj._ncols
    for child in obj.get_children():
        tikzplotlib_fix_ncols(child)
```

4 Reflectivity without ARC

4.1 $\theta = 0°$

With normal incidence of the light, and considering the bulk silicon as infinite length transmission line, the reflectivity can be calculated as

$$R = \left(\frac{z_\infty^{si} - z_\infty^{air}}{z_\infty^{si} - z_\infty^{air}}\right)^2 = \left(\frac{z_0/n_{si} - z_0/n_{air}}{z_0/n_{si} + z_0/n_{air}}\right)^2 = \left(\frac{n_{air} - nsi}{n_{air} + nsi}\right)^2 \tag{1}$$

To compute that, i used python:

```
# %% compute the reflectivity
n_si=3.8     # refraction coefficient of silicon
n_air=1      # refraction coefficient of air

R_{\theta=0^\circ}=((n_air-n_si)/(n_air+n_si))**2     # Normal incidence
```

the result is:

$$R = 0.3403$$

4.2 $\theta = 30°$

In this case, we have to consider the two components (polarization) of the light. This is possible because the s and p polarization are orthogonal, so they form a basis in which any wave can be projected. The reflection coefficients and the reflectivity are:

$$\Gamma^p = \frac{n_{air}\sqrt{n_{si}^2 - n_{air}^2 \sin^2\theta} - n_{si}^2\cos\theta}{n_{air}\sqrt{n_{si}^2 - n_{air}^2 \sin^2\theta} + n_{si}^2\cos\theta} \tag{2}$$

$$\Gamma^s = \frac{n_{air}\cos\theta - \sqrt{n_{si}^2 - n_{air}^2 \sin^2\theta}}{n_{air}\cos\theta + \sqrt{n_{si}^2 - n_{air}^2 \sin^2\theta}} \tag{3}$$

$$R^p = |\Gamma^p|^2 = 0.3921 \tag{4}$$

$$R^s = |\Gamma^s|^2 = 0.2883 \tag{5}$$

$$R = \text{mean}(R^p, R^s) = 0.3402 \tag{6}$$

the result are calculated using the following:

```
theta=30/360*2*np.pi                    # Angle of incidence, in radiants
R_tm=((n_air*np.sqrt(n_si**2-n_air**2*np.sin(theta)**2)-n_si**2*np.cos(theta))/
↪  (n_air*np.sqrt(n_si**2-n_air**2*np.sin(theta)**2)+n_si**2*np.cos(theta)))**2
R_te=((n_air*np.cos(theta)-np.sqrt(n_si**2-n_air**2*np.sin(theta)**2))/
↪  (n_air*np.cos(theta)+np.sqrt(n_si**2-n_air**2*np.sin(theta)**2)))**2
R_mean=np.mean([R_te,R_tm])
```

One thing to notice is that, on average, if we consider unpolarized light, at $\theta = 30°$, we have almost the same reflectivity as with normal incidence.

5 ARC design - single layer

5.1 Wavelength

In order to design an ARC, a wavelength at which that ARC will be optimized is needed. Ideally, at that wavelength, the reflectivity will be exactly null. To decide at which wavelength perform the design, we can analyze where the most of the solar energy is concentrated. To do that, i plotted the AM1.5 standard irradiation:

```
# %% Sun tipical irradiance
AM15_lambda=np.array([300,  ...    ])       # wavelength
AM15_irr=np.array([0.0010205, ...  ])       # irradiance

fig, axs=plt.subplots()
fig.tight_layout()
axs.semilogx(AM15_lambda,AM15_irr)
axs.set_xlabel('$\lambda$ [nm]')
axs.set_ylabel('Irradiance [${W}/{(m^2\cdot nm)}$]')
axs.grid(True,'both')
tikzplotlib.save('irradiance.tex',axis_width='0.9\\textwidth',axis_height ='6cm')
```

Looking at the standard sunlight irradiance spectrum in **Figure 1**, is possible to see that the most irradiance is at $\lambda = 495$nm.

The simplest choice is to design the ARC at $\lambda_0 = 495$nm

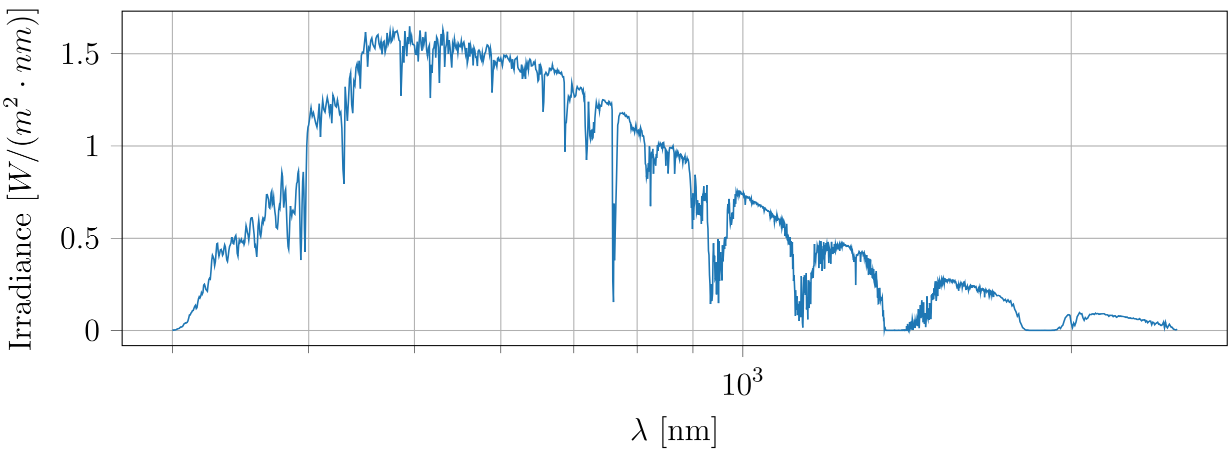

Figure 1: *Typical irradiation spectrum of the sun*

Another approach, considering that the reflectivity will have a bell shaped graph, would be to consider the “ center of mass ” of the irradiance distribution, and design the ARC at that wavelength, so that an ARC centered in that point will be, on average, near to the wavelength with the most irradiance:

$$\lambda_c = \frac{\sum_{i=1}^{n} I(\lambda_i)\lambda_i}{\sum_{i=1}^{n} I(\lambda_i)} = 767\text{nm}$$

```
# center of mass, since both positive, we can use norm1
com=(np.dot(AM15_irr,AM15_lambda))/np.linalg.norm(AM15_irr,1)
```

Looking at the plot in **Figure 1**, it seem more reasonable to stick to the first consideration and design the ARC for the peack irradiance wavelength $\lambda = 495\text{nm}$.

5.2 Design procedure

The working principle of a SLARC is to match the thickness of the layer with the quarter of the design wavelength λ_0, in the layer medium. The ideal refractive index depends on the two mediums on the sides of the ARC, in this case air and silicon.

$$n_{ar} = \sqrt{n_{air} n_{si}} \tag{7}$$

$$\lambda_{ar} = \frac{\lambda_0}{n_{ar}} \tag{8}$$

$$d = \frac{\lambda_{ar}}{4} = 63.48\text{nm} \tag{9}$$

n_{ar} ideal refractive index, n_{si} refractive index of silicon, λ_{ar} design wavelength in the ARC medium.

Unfortunately, there is no material with the ideal refractive index, so the procedure is repeated considering some materials with similar refractive indeces: Al_2O_3, Si_3N_4, SiO_2, TiO_2.

```
# %% SLARC design
lambda0=495 # design wavelength [nm]
n_ideal=np.sqrt(n_air*n_si) # ideal refractive index at lambda0
d_ideal=lambda0/4/n_ideal # ideal thickness
n_TiO2= 2.7193              # at 495 nm
d_TiO2= lambda0/4/n_TiO2    # optimized for lambda0
n_Si3N4 = 2.0647            # at 495 nm
d_Si3N4=lambda0/4/n_Si3N4   # optimized for lambda0
n_Al2O3 = 1.7747            # at 495 nm
d_Al2O3=lambda0/4/n_Al2O3   # optimized for lambda0
n_SiO2  = 1.4626            # at 495 nm
d_SiO2=lambda0/4/n_SiO2     # optimized for lambda0
```

5.3 Compute the reflectivity - neglect chromatic dispersion

At this point we can study how the ARC is affecting the reflectivity. Neglecting the chromatic dispersion means to consider just one refractive index for all the wavelength. The problem can be formulated in computing the input reflection coefficient for the transmission line shown in **Figure 2**. Since we consider the silicon to be infinitely thick, $\Gamma_B^+ = 0$, $Z_B = Z_\infty^{si}$. And for normal incidence, $Z_\infty^{si} = Z_0/n_{si}$ and $Z_\infty^{ar} = Z_0/n_{ar}$.

$$\Gamma_B^- = \frac{Z_\infty^{si} - Z_\infty^{ar}}{Z_\infty^{si} + Z_\infty^{ar}} \tag{10}$$

$$\Gamma_A^+ = \Gamma_B^- e^{-2j\cdot k\cdot d} \tag{11}$$

$$Z_A = Z_\infty^{ar} \frac{1+\Gamma_A^+}{1-\Gamma_A^+} \tag{12}$$

$$\Gamma_A^- = \frac{Z_A - Z_\infty^{air}}{Z_A - Z_\infty^{air}} \tag{13}$$

This calculation are performed in the following script, and the result shown in **Figure 3**

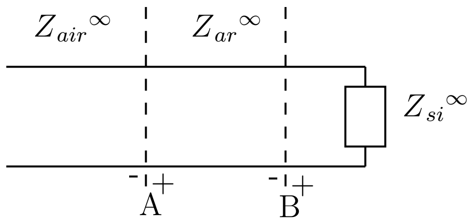

Figure 2: *Equivalent TL*

```
# %% check reflectivity - no cromatic dispersion
fig, axs=plt.subplots()
fig.tight_layout()
for n_ar, d in zip([n_ideal,n_Al2O3,n_Si3N4,n_SiO2,n_TiO2], [d_ideal,d_Al2O3,d_Si3N4,d_SiO2,d_TiO2]):
    Z0=120*np.pi                # free space inpedance
    Z_inf_ar=Z0/n_ar            # impedence of ARC
    Z_inf_Si=Z0/n_si            # impedence of bulk silicon
    Gbminus=(Z_inf_Si-Z_inf_ar)/(Z_inf_Si+Z_inf_ar) # reflection coefficient
    K0=2*np.pi/lambdas
    k=K0*n_ar
    Gaplus=(Gbminus*np.exp(-2j*k*d))
    Za=Z_inf_ar*(1+Gaplus)/(1-Gaplus)
    Gaminus=(Za-Z0)/(Za+Z0)
    R_vect=np.abs(Gaminus)**2*100
    axs.plot(lambdas,R_vect)
axs.set_xlabel('$\lambda$ [nm]')
axs.set_ylabel('Reflectivity [%]')
axs.grid(True,'both')
axs.minorticks_on()
axs.legend(['ideal $n=1.9493$', '$Al_2O_3 \, n=1.7747$', '$Si_3N_4 \, n=2.0647$', '$SiO_2 \, n=1.4626$', '$TiO_2 \, n=2.7193$'])

```

Looking at **Figure 3** it is evident how the best match is the Si_3N_4 coating.

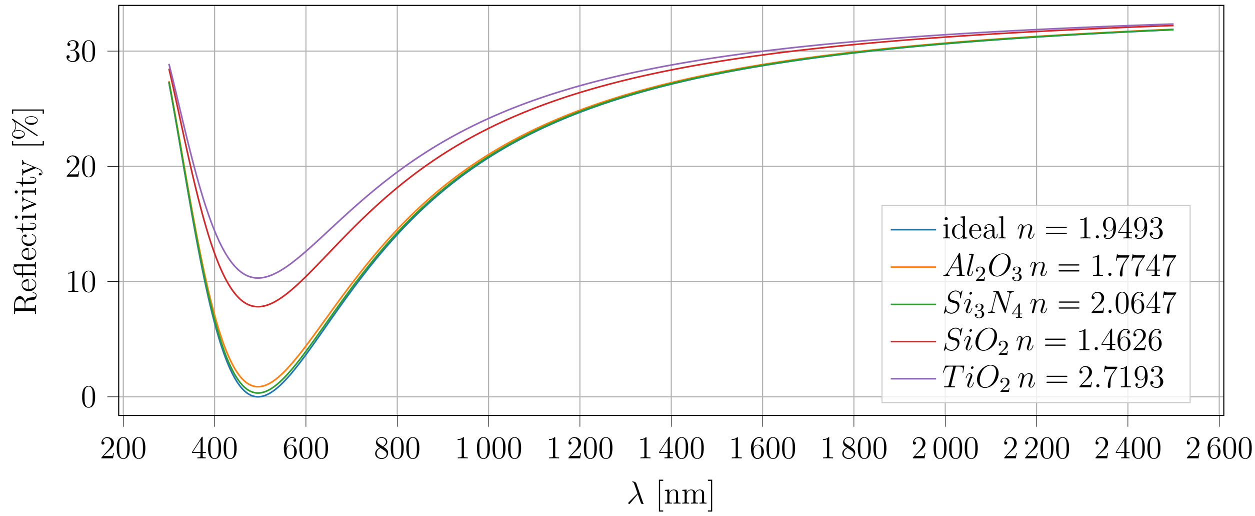

Figure 3: *Different ARCs considering a single n*

5.4 Compute the reflectivity - consider chromatic dispersion

In the assignment it was not required to account for the chromatic dispersion, but i wanted to see if it would have made a big difference. To do that i used the Sellmeier equations taken from [1][2], and from the website https://refractiveindex.info/.

$$n^2_{Al_2O_3} - 1 = \frac{1.4313493\lambda^2}{\lambda^2 - 0.0726631^2} + \frac{0.65054713\lambda^2}{\lambda^2 - 0.1193242^2} + \frac{5.3414021\lambda^2}{\lambda^2 - 18.028251^2} \tag{14}$$

$$n^2_{Si_3N_4} - 1 = \frac{3.0249\lambda^2}{\lambda^2 - 0.1353406^2} + \frac{40314\lambda^2}{\lambda^2 - 1239.842^2} \tag{15}$$

$$n^2_{SiO_2} - 1 = \frac{0.6961663\lambda^2}{\lambda^2 - 0.0684043^2} + \frac{0.4079426\lambda^2}{\lambda^2 - 0.1162414^2} + \frac{0.8974794\lambda^2}{\lambda^2 - 9.896161^2} \tag{16}$$

$$n^2_{TiO_2} = 5.913 + \frac{0.2441}{\lambda^2 - 0.0803} \tag{17}$$

with this i considered the dependency of n on λ, and plotted the dispersion an shown in **Figure 4**. The script is the following:

```
# %% compute the chromatic dispersion
n_ideal_v=n_ideal*lambdas/lambdas # ideal refractive index at lambda0
lambdas_um=lambdas/1000
n_TiO2_v = (4.99+1/96.6*lambdas_um**(-1.1)+1/4.6*lambdas_um**(-1.95))**.5            # at every
↪ lambda
n_Si3N4_v = (1+3.0249/(1-(0.1353406/lambdas_um)**2)+40314/(1-(1239.842/lambdas_um)**2))**.5
↪ # at every lambda
n_Al2O3_v = (1+1.4313493/(1-(0.0726631/lambdas_um)**2)+0.65054713/(1-(0.1193242/lambdas_um)**2)
↪ +5.3414021/(1-(18.028251/lambdas_um)**2))**.5            # at every lambda
n_SiO2_v  = (1+0.6961663/(1-(0.0684043/lambdas_um)**2)+0.4079426/(1-(0.1162414/lambdas_um)**2)
↪ +0.8974794/(1-(9.896161/lambdas_um)**2))**.5            # at every lambda

fig, axs=plt.subplots()
fig.tight_layout()
for n_ar in [n_ideal_v,n_Al2O3_v,n_Si3N4_v,n_SiO2_v,n_TiO2_v]:
```

```
    axs.plot(lambdas,n_ar)
axs.set_xlabel('$\lambda$ [nm]')
axs.set_ylabel('$n$')
axs.grid(True,'both')
axs.minorticks_on()
axs.legend(['ideal', '$Al_2O_3$', '$Si_3N_4$', '$SiO_2$', '$TiO_2$'])
```

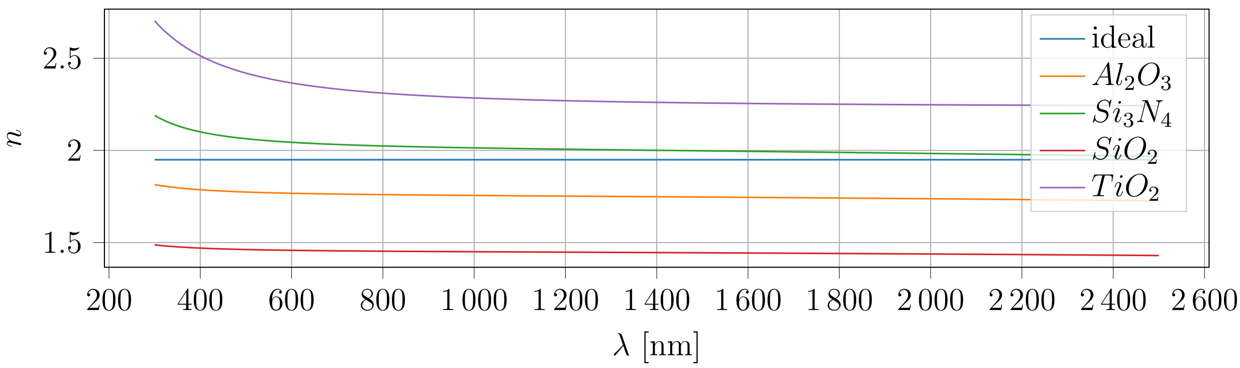

Figure 4: *n dependency on λ*

And also computed the input reflectivity considering the chromatic dispersion. As evident in **Figure 5**, the effect of the varying refraction coefficient is negligible for Al_2O_3 and Si_3N_4, but is quite visible for SiO_2 and TiO_2.

The chosen design will be to use Si_3N_4, with a thickness of 59.93nm, which gives a minimum reflectivity of about 0.33%.

The code to compute the reflectivity is:

```
# %% check reflectivity - with dispersion
fig, axs=plt.subplots()
fig.tight_layout()
for n_ar, d in zip([n_ideal_v,n_Al2O3_v,n_Si3N4_v,n_SiO2_v,n_TiO2_v], [d_ideal,d_Al2O3,d_Si3N4,d_SiO2,d_TiO2]):
    Z0=120*np.pi                 # free space inpedance
    Z_inf_ar=Z0/n_ar             # impedence of ARC
    Z_inf_Si=Z0/n_si             # impedence of bulk silicon
    Gbminus=(Z_inf_Si-Z_inf_ar)/(Z_inf_Si+Z_inf_ar) # reflection coefficient
    k=K0*n_ar
    Gaplus=(Gbminus*np.exp(-2j*k*d))
    Za=Z_inf_ar*(1+Gaplus)/(1-Gaplus)
    Gaminus=(Za-Z0)/(Za+Z0)
    R_vect=np.abs(Gaminus)**2*100
    if n_ar is n_Si3N4_v:
        R_SLARC=R_vect #save the best design
    axs.plot(lambdas,R_vect)

axs.set_xlabel('$\lambda$ [nm]')
axs.set_ylabel('Reflectivity [%]')
axs.grid(True,'both')
```

```
axs.minorticks_on()
axs.legend(['ideal', '$Al_2O_3$', '$Si_3N_4$', '$SiO_2$', '$TiO_2$'])
```

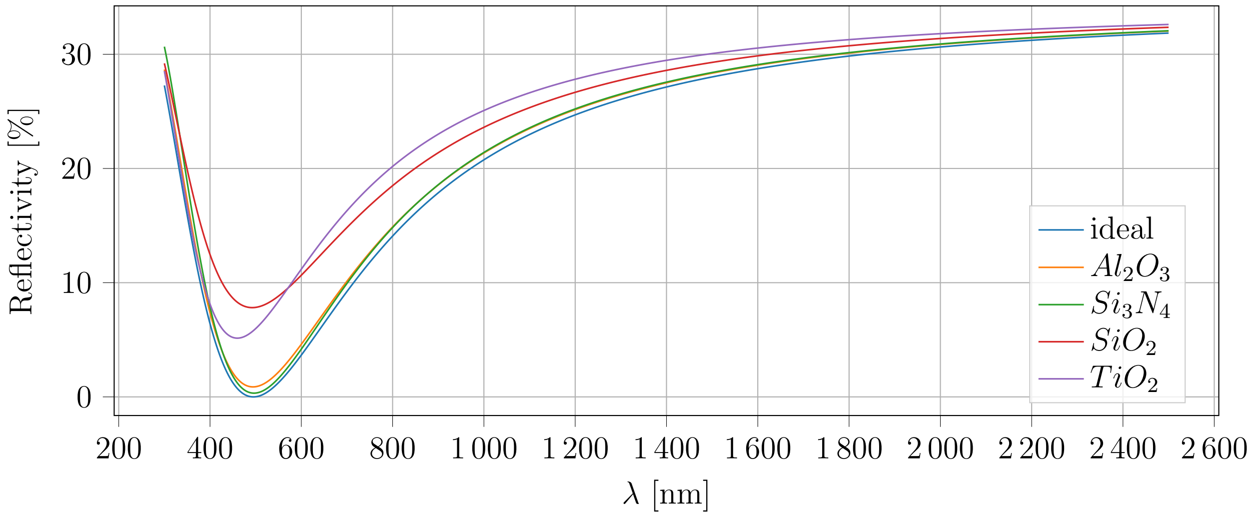

Figure 5: *Different ARCs considering n dependency on* λ

6 DLARC design - double layer

6.1 Silica/Titania layers

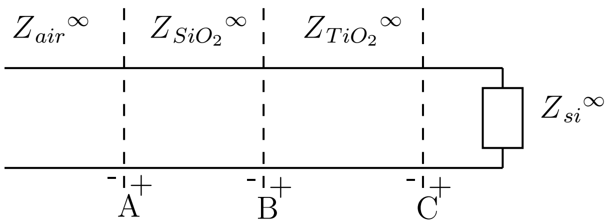

Figure 6: *Equivalent TL - DLARC*

At this point the assignment ask to design a double layer ARC with silica and titania. I will follow the QWOT approach, fixing before the layer thicknesses. This were already calculated in the previous design.

$$d_{TiO_2} = 45.5\text{nm} \quad d_{SiO_2} = 84.6\text{nm}$$

The remaining step is to compute the reflectivity with this new DLARC, that is equivalent to the TL shown in **Figure 6**.

The script is the following:

```
# %% DLARC - SILICA/TITANIA
Z_inf_SiO2=Z0/n_SiO2
Z_inf_TiO2=Z0/n_TiO2
Gcminus=(Z_inf_Si-Z_inf_TiO2)/(Z_inf_Si+Z_inf_TiO2)
k=K0*n_dict['$TiO_2$']
Gbplus=Gcminus*np.exp(-2j*k*d_TiO2)
```

```
Zb=Z_inf_TiO2*(1+Gbplus)/(1-Gbplus)
Gbminus=(Zb-Z_inf_SiO2)/(Zb+Z_inf_SiO2)
k=K0*n_dict['$SiO_2$']
Gaplus=Gbminus*np.exp(-2j*k*d_SiO2)
Za=Z_inf_SiO2*(1+Gaplus)/(1-Gaplus)
Gaminus=(Za-Z0)/(Za+Z0)
R_DLARC_SiO2TiO2=np.abs(Gaminus)**2*100
```

6.2 Other materials layers

It was not requested from the assignment, but i was curious about if some materials were available with a more suitable refractive index ratios:

$$\frac{n_1}{n_2} = \sqrt{\frac{n_{air}}{n_{si}}} = 0.510$$

In order to understand that, i considered the materials in the **Table 1**, and computed the ratio for any combinations. The results are reported in **Figure 7**. It turned out that the pair of material with most suitable ratio (red line reference) is Ta_2O_5 and Ge. The code to perform that is the following:

```
# %% DL ARC - some other materials
n_dict={"$Na_3AlF_6$": 1.35,
        "$MgF_2$": 1.38,
        "$SiO_2$": 1.46,
        "$Al_2O_3$": 1.629,
        "$CeF_3$": 1.63,
        "$PbF_2$": 1.73,
        "$Ta_2O_5$": 2.126,
        "$ZrO_2$": 2.20,
        "$ZnS$": 2.32,
        "$TiO_2$": 2.40,
        "$Bi_2O_3$": 2.45,
        "$Ge$": 4.2,
        "$Te$": 4.60}

legend=[]; val=[]
for keys_e, values_e in n_dict.items():
    for keys_i, values_i in n_dict.items():
        val.append(values_e/values_i)
        legend.append(keys_e + "//" + keys_i)
fig, axs=plt.subplots()
fig.tight_layout()
axs.plot(legend,val,'x')
axs.hlines([(n_air/n_si)**0.5],legend[0],legend[-1],'red')
axs.set_xlabel('Combination of materials')
axs.set_ylabel('$n_1/n_2$')
```

```python
axs.grid(True,'both')
axs.legend(['available ratios', 'reference'])
```

material	n	material	n
Na_3AlF_6	1.35	ZrO_2	2.20
MgF_2	1.38	ZnS	2.32
SiO_2	1.46	TiO_2	2.40
Al_2O_3	1.62	Bi_2O_3	2.45
CeF_3	1.63	Ge	4.20
PbF_2	1.73	Te	4.60
Ta_2O_5	2.12	-	-

Table 1: *Common materials refractive indeces*

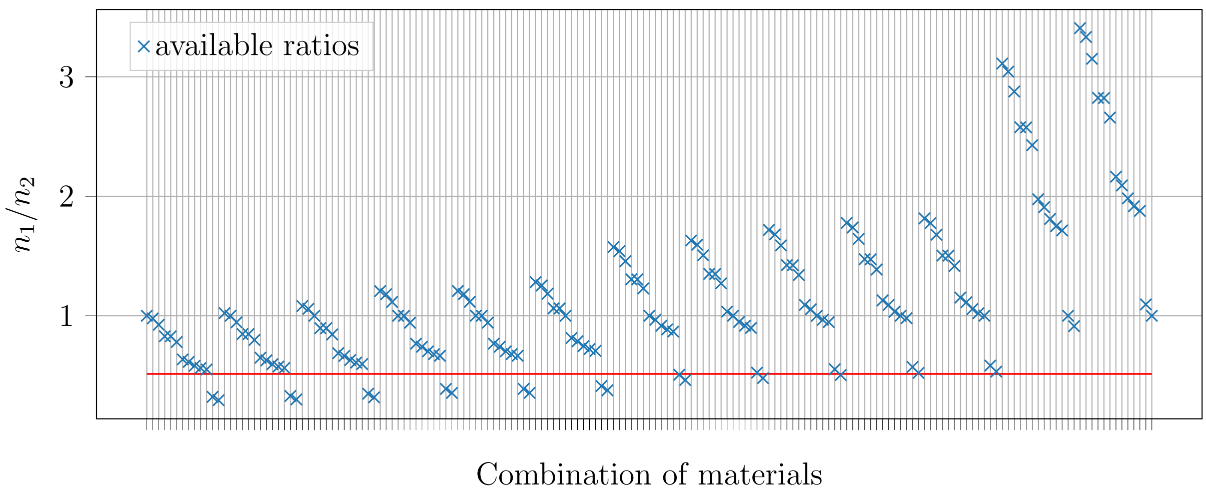

Figure 7: *Different materials n ratios*

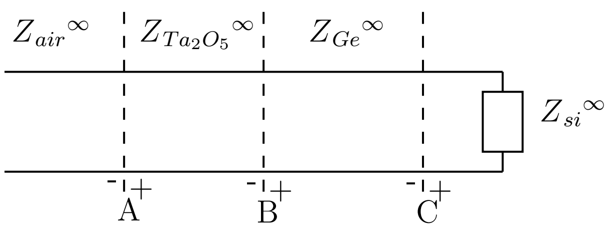

Figure 8: *Equivalent TL - DLARC*

At this point the problem become similar to the previous one: design a DLARC with Ta_2O_5 and Ge, using QWOT approach, and than solve the TL in **Figure 8** for input reflectivity. The script is the following:

```python
# %% DLARC - Ta2O5 and Ge
d_Ge=lambda0/4/n_dict['$Ge$']
d_Ta=lambda0/4/n_dict['$Ta_2O_5$']
Z_inf_Ge=Z0/n_dict['$Ge$']
```

```
Z_inf_Ta=Z0/n_dict['$Ta_20_5$']
Gcminus=(Z_inf_Si-Z_inf_Ge)/(Z_inf_Si+Z_inf_Ge)
k=K0*n_dict['$Ge$']
Gbplus=Gcminus*np.exp(-2j*k*d_Ge)
Zb=Z_inf_Ge*(1+Gbplus)/(1-Gbplus)
Gbminus=(Zb-Z_inf_Ta)/(Zb+Z_inf_Ta)
k=K0*n_dict['$Ta_20_5$']
Gaplus=Gbminus*np.exp(-2j*k*d_Ta)
Za=Z_inf_Ta*(1+Gaplus)/(1-Gaplus)
Gaminus=(Za-Z0)/(Za+Z0)
R_DLARC_Ta2O5Ge=np.abs(Gaminus)**2*100
```

6.3 Comparison of designs

At this point we can compare the reflectivity of the best SLARC design with Silica/Titania and Ta_2O_5/Ge DLARC designs. The script is the following, and the result are shown in **Figure 9**.

It's evident that the Ta_2O_5/Ge DLARC outperform both TiO_2/SiO_2 and $Si3N_4$ coating at the design wavelength λ_0, because it reaches almost zero reflectivity, but the reflectivity drop at λ_0 in a really narrow manner, w.r.t. TiO_2/SiO_2. This may imply that even if Ta_2O_5/Ge is best at λ_0, TiO_2/SiO_2 could be best in average considering the whole useful sunlight spectrum.

```
# %% check reflectivity - SLARC / DLARCs
fig, axs=plt.subplots()
fig.tight_layout()
axs.semilogy(lambdas,R_DLARC_SiO2TiO2)
axs.semilogy(lambdas,R_DLARC_Ta2O5Ge)
axs.semilogy(lambdas,R_SLARC)
axs.set_xlabel('$\lambda$ [nm]')
axs.set_ylabel('Reflectivity [%]')
axs.grid(True,'both')
axs.minorticks_on()
axs.legend(['DLARC $TiO2 \,/\, SiO2$','DLARC $Ta_2O_5 \,/\, Ge$','SLARC $Si_3N_4$'])
```

7 Effective reflectivity

In the evaluation of the performance, up to now, we neglected the actual power distribution of the sunlight. To account for that it is possible to compute the effective reflectivity, defined as:

$$R_{eff} = \frac{\int_{\lambda_{min}}^{\lambda_{max}} R(\lambda) \cdot N(\lambda) \mathrm{d}\lambda}{\int_{\lambda_{min}}^{\lambda_{max}} N(\lambda) \mathrm{d}\lambda} \tag{18}$$

$R(\lambda)$ is the spectral reflectivity already calculated, and $N(\lambda)$ is the standard solar spectrum already considered in **Figure 1**. As usual the calculation is carried out in python, and the result are collected in the **Table 2**.

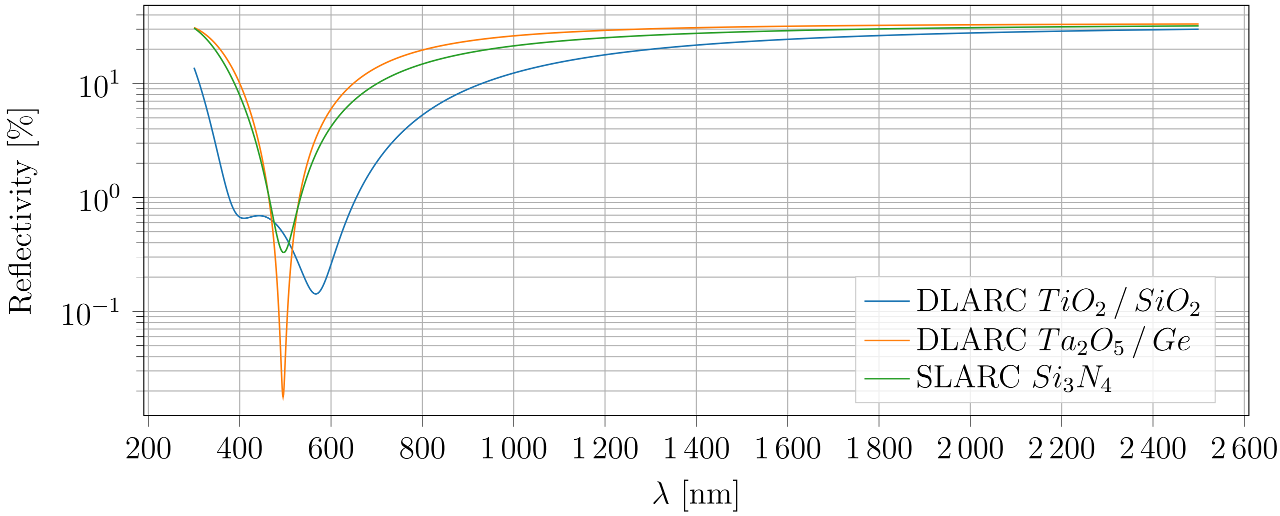

Figure 9: *Reflectivity for different designs*

```
# %% check effective reflectivity
indx=0
R_eff=[]; R_min=[]
for R in [R_DLARC_SiO2TiO2, R_DLARC_Ta2O5Ge, R_SLARC]:
    NUM=0; DEN=0
    for _ in lambdas:
        if indx is (len(lambdas)-2):
            break
        NUM+=R[indx]*AM15_irr[indx]*(lambdas[indx+1]-lambdas[indx])
        DEN+=AM15_irr[indx]*(lambdas[indx+1]-lambdas[indx])
    R_eff.append(NUM/DEN)
    R_min.append(np.min(R))
print(R_eff);print(R_min)
```

ARC	SiO_2/TiO_2	Ta_2O_5/Ge	Si_3N_4
R_{eff}	13.775 %	30.979 %	30.664 %
R_{min}	0.142 %	0.017 %	0.328 %

Table 2: *Min and effective reflectivity*

Now, with this new data, it is clear that the best design is the SiO_2/TiO_2 DLARC, because even if it doesn't have the best minimum reflectivity, on average, it has the lowest where the most solar power is distributed.

8 Tolerances

The SiO_2/TiO_2 DLARC has to be manufactured with some technique, so the problem of the tolerance of manufacturing become relevant. In the **Figure 10**, we consider how the two different deviation from the nominal value affect the reflectivity. As usual this is the code for the computations:

```python
# %% DLARC - tolerance effect
Z_inf_SiO2=Z0/n_dict['$SiO_2$']
Z_inf_TiO2=Z0/n_dict['$TiO_2$']
Gcminus=(Z_inf_Si-Z_inf_TiO2)/(Z_inf_Si+Z_inf_TiO2)
k=K0*n_dict['$TiO_2$']
fig, axs=plt.subplots()
fig.tight_layout()
legend=[]
for err1 in [-0.02, 0.02]:
    for err2 in [-0.02, 0.02]:
        Gbplus=Gcminus*np.exp(-2j*k*d_TiO2*(1-err1))
        Zb=Z_inf_TiO2*(1+Gbplus)/(1-Gbplus)
        Gbminus=(Zb-Z_inf_SiO2)/(Zb+Z_inf_SiO2)
        k=K0*n_dict['$SiO_2$']
        Gaplus=Gbminus*np.exp(-2j*k*d_SiO2*(1-err2))
        Za=Z_inf_SiO2*(1+Gaplus)/(1-Gaplus)
        Gaminus=(Za-Z0)/(Za+Z0)
        R_DLARC_tolerance=np.abs(Gaminus)**2*100
        axs.plot(lambdas,R_DLARC_tolerance)
        legend.append('$TiO_2$ ('+str(err1)+'\%), $SiO_2$ ('+str(err2)+'\%)')
axs.plot(lambdas,R_DLARC_SiO2TiO2)
legend.append('$TiO_2$, $SiO_2$ nominal tickness')
axs.minorticks_on()
axs.set_xlabel('$\lambda$ [nm]')
axs.set_ylabel('Reflectivity [%]')
axs.grid(True,'both')
axs.legend(legend,loc='lower right')
```

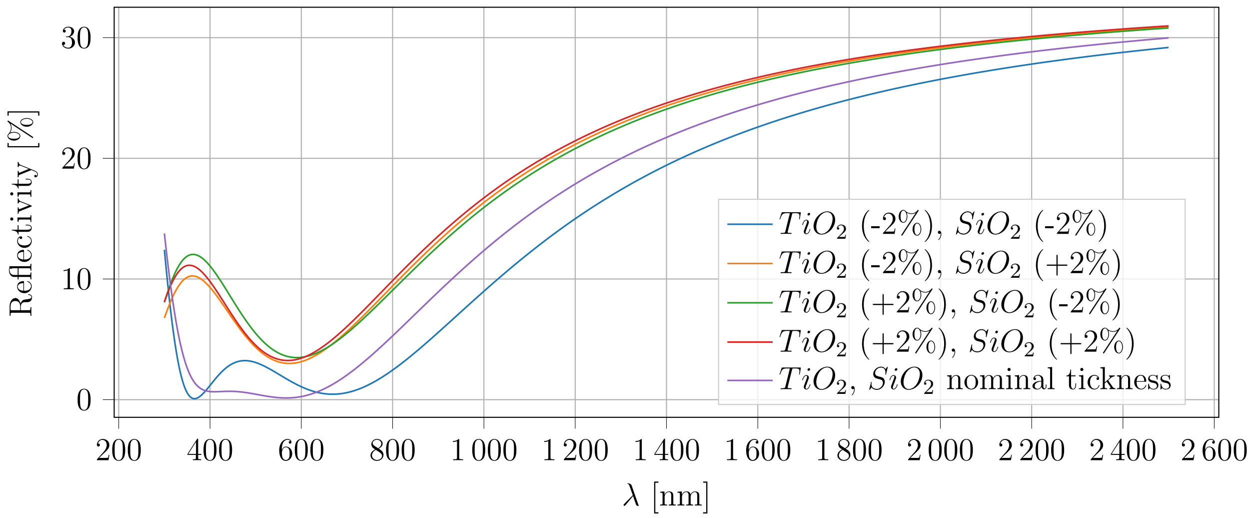

Figure 10: *Reflectivity for different manufacture deviations*

References

[1] Francesco Scotognella, Alessandro Chiasera, Luigino Criante, Eduardo Aluicio-Sarduy, Stefano Varas, Stefano Pelli, Anna Łukowiak, Giancarlo C. Righini, Roberta Ramponi, and Maurizio Ferrari. Metal oxide one dimensional photonic crystals made by rf sputtering and spin coating. *Ceramics International*, 41(7):8655–8659, 2015.

[2] KEVIN LUKE, YOSHITOMO OKAWACHI, MICHAEL R. E. LAMONT, ALEXANDER L. GAETA, and MICHAL LIPSON. Broadband mid-infrared frequency comb generation in a Si3N4 microresonator. *Optics Letters*, 40(21), 2015.