



**Politecnico  
di Torino**

INDUSTRIAL PHOTONICS

## Design of ARCs

*Ariel Priarone*

s274149

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Assignment</b>	<b>2</b>
<b>3</b>	<b>Initialize the script</b>	<b>2</b>
<b>4</b>	<b>Reflectivity without ARC</b>	<b>3</b>
4.1	$\theta = 0^\circ$ . . . . .	3
4.2	$\theta = 30^\circ$ . . . . .	3
<b>5</b>	<b>ARC design - single layer</b>	<b>4</b>
5.1	Wavelength . . . . .	4
5.2	Design procedure . . . . .	4
5.3	Compute the reflectivity - neglect chromatic dispersion . . . . .	5
5.4	Compute the reflectivity - consider chromatic dispersion . . . . .	7
<b>6</b>	<b>DLARC design - double layer</b>	<b>8</b>
6.1	Silica/Titania layers . . . . .	8
6.2	Other materials layers . . . . .	9
6.3	Comparison of designs . . . . .	11
<b>7</b>	<b>Effective reflectivity</b>	<b>12</b>
<b>8</b>	<b>Tolerances</b>	<b>13</b>

# List of Figures

1	Typical irradiation spectrum of the sun . . . . .	4
2	Equivalent TL . . . . .	6
3	Different ARCs considering a single $n$ . . . . .	6
4	$n$ dependency on $\lambda$ . . . . .	8
5	Different ARCs considering $n$ dependency on $\lambda$ . . . . .	9
6	Equivalent TL - DLARC . . . . .	9
7	Different materials $n$ ratios . . . . .	11
8	Equivalent TL - DLARC . . . . .	11
9	Reflectivity for different designs . . . . .	12
10	Reflectivity for different manufacture deviations . . . . .	14

# List of Tables

1	Common materials refractive indeces . . . . .	10
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 principle of work of ARCs is that if the reflected light that is reflected back at the first air/material interface has a phase shift of  $180^\circ$ , 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  $\theta = 0^\circ$  and at  $\theta = 30^\circ$
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. The script is available at <https://github.com/arielpriarone/IndustrialPhotonics/tree/main/Assignment1>.

```
1 import matplotlib.pyplot as plt
2 import matplotlib.colors as color
3 import numpy as np
4 import matplotlib
5 import tikzplotlib
6
7 def tikzplotlib_fix_ncols(obj):
8     """
9     workaround for matplotlib 3.6 renamed legend's _ncol to _ncols, which breaks tikzplotlib
10    """
11    if hasattr(obj, "_ncols"):
12        obj._ncol = obj._ncols
13    for child in obj.get_children():
14        tikzplotlib_fix_ncols(child)
```

## 4 Reflectivity without ARC

### 4.1 $\theta = 0^\circ$

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

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

To compute that, i used python:

```

1  # %% compute the reflectivity
2  n_si=3.8      # refraction coefficient of silicon
3  n_air=1       # refraction coefficient of air
4
5  R_{\theta=0^\circ} = ((n_air-n_si)/(n_air+n_si))**2    # Normal incidence

```

the result is:

$$R = 0.3403$$

### 4.2 $\theta = 30^\circ$

In this case, we have to consider the two components (polarization) of the light. The reflection coefficients and the reflectivity are:

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

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

$$R^p = |\Gamma^p|^2 = 0.3921 \quad (4)$$

$$R^s = |\Gamma^s|^2 = 0.2883 \quad (5)$$

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

the result are calculated using the following:

```

1  theta=30/360*2*np.pi      # Angle of incidence, in radians
2  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
3  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
4  R_mean=np.mean([R_te,R_tm])

```

One thing to notice is that, on average, if we consider unpolarized light, at  $\theta = 30^\circ$ , 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 optimize 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:

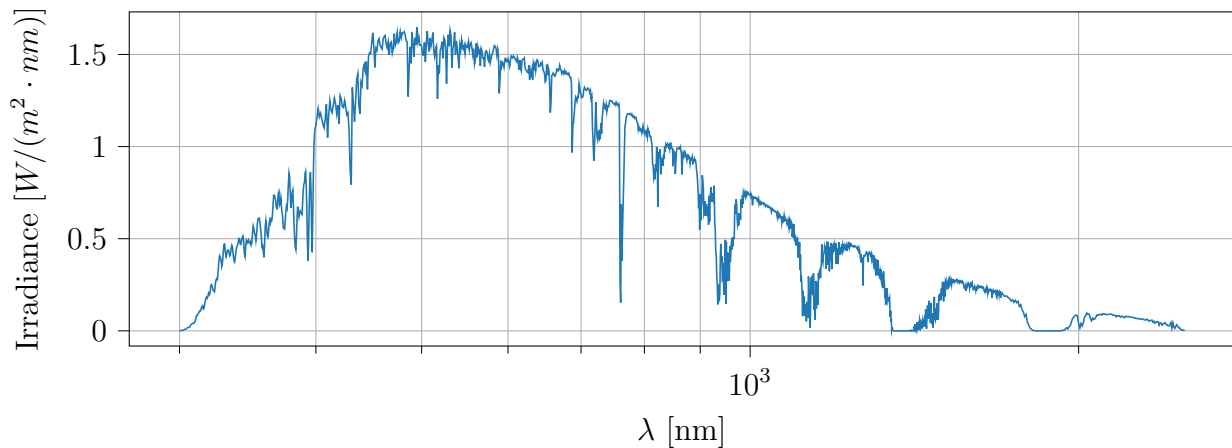
```

1  # %% Sun tipical irradiance
2  AM15_lambda=np.array([300, ... ])      # wavelength
3  AM15_irr=np.array([0.0010205, ... ])   # irradiance
4
5  fig, axs=plt.subplots()
6  fig.tight_layout()
7  axs.semilogx(AM15_lambda,AM15_irr)
8  axs.set_xlabel('$\lambda$ [nm]')
9  axs.set_ylabel('Irradiance [W/(m^2\cdot nm)]')
10 axs.grid(True, 'both')
11 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  $\lambda = 495\text{nm}$ .

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



**Figure 1:** *Typical irradiance spectrum of the sun*

### 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  $\lambda_0$ , 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.

$$n_{ar} = \sqrt{n_{air}n_{si}} \quad (7)$$

$$\lambda_{ar} = \frac{\lambda_0}{n_{ar}} \quad (8)$$

$$d = \frac{\lambda_{ar}}{4} = 63.48\text{nm} \quad (9)$$

$n_{ar}$  ideal refractive index,  $n_{si}$  refractive index of silicon,  $\lambda_{ar}$  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:  $Al_2O_3$ ,  $Si_3N_4$ ,  $SiO_2$ ,  $TiO_2$ .

```

1  # %% SLARC design
2  lambda0=495 # design wavelength [nm]
3  n_ideal=np.sqrt(n_air*n_si) # ideal refractive index at lambda0
4  d_ideal=lambda0/4/n_ideal # ideal thickness
5  n_TiO2= 2.7193 # at 495 nm
6  d_TiO2= lambda0/4/n_TiO2 # optimized for lambda0
7  n_Si3N4 = 2.0647 # at 495 nm
8  d_Si3N4=lambda0/4/n_Si3N4 # optimized for lambda0
9  n_Al2O3 = 1.7747 # at 495 nm
10 d_Al2O3=lambda0/4/n_Al2O3 # optimized for lambda0
11 n_SiO2 = 1.4626 # at 495 nm
12 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  $Z_\infty^{ar} = Z_0/n_{ar}$ .

$$\Gamma_B^- = \frac{Z_\infty^{si} - Z_\infty^{ar}}{Z_\infty^{si} + Z_\infty^{ar}} \quad (10)$$

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

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

$$\Gamma_A^- = \frac{Z_A - Z_\infty^{air}}{Z_A + Z_\infty^{air}} \quad (13)$$

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

```

1  # %% check reflectivity - no cromatic dispersion
2  fig, axs=plt.subplots()
3  fig.tight_layout()

```

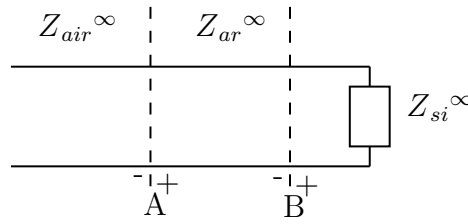
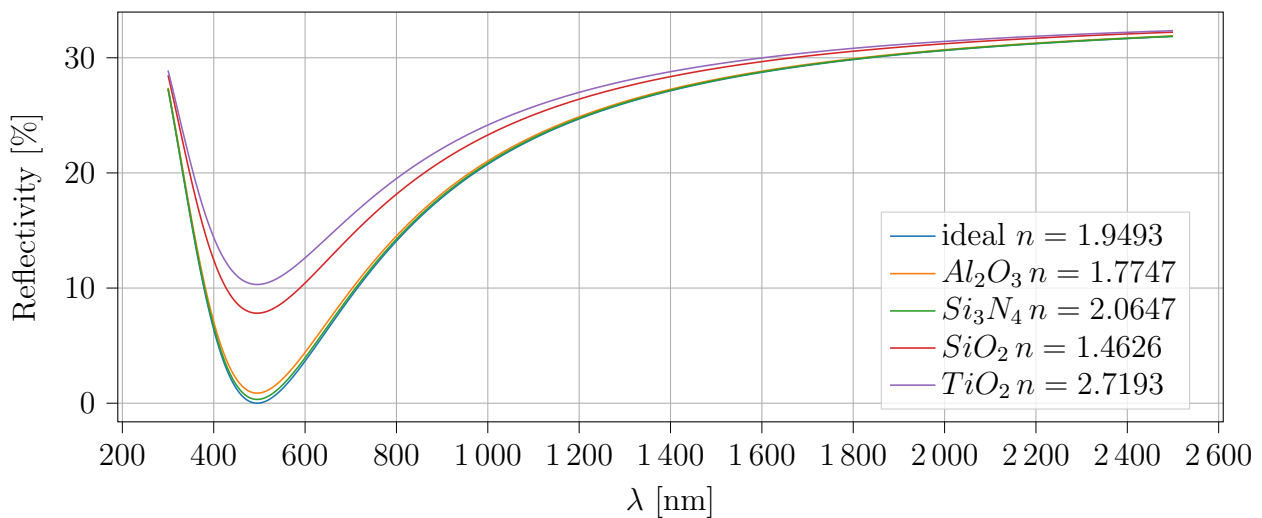


Figure 2: Equivalent TL

```

4  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]):
5      Z0=120*np.pi                # free space impedance
6      Z_inf_ar=Z0/n_ar             # impedance of ARC
7      Z_inf_Si=Z0/n_si            # impedance of bulk silicon
8      Gbminus=(Z_inf_Si-Z_inf_ar)/(Z_inf_Si+Z_inf_ar) # reflection coefficient
9      K0=2*np.pi/lambda_s
10     k=K0*n_ar
11     Gaplus=(Gbminus*np.exp(-2j*k*d))
12     Za=Z_inf_ar*(1+Gaplus)/(1-Gaplus)
13     Gaminus=(Za-Z0)/(Za+Z0)
14     R_vect=np.abs(Gaminus)**2*100
15     axs.plot(lambda_s,R_vect)
16 axs.set_xlabel('$\lambda$ [nm]')
17 axs.set_ylabel('Reflectivity [%]')
18 axs.grid(True,'both')
19 axs.minorticks_on()
20 axs.legend(['ideal $n=1.9493$', '$Al_2O_3 \setminus, n=1.7747$', '$Si_3N_4 \setminus, n=2.0647$', '$SiO_2 \setminus, n=1.4626$', '$TiO_2 \setminus, n=2.7193$'])
21

```

Figure 3: Different ARCs considering a single  $n$ 

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

## 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/>.

$$n_{Al_2O_3}^2 - 1 = \frac{1.4313493\lambda^2}{\lambda^2 - 0.0726631^2} + \frac{0.65054713\lambda^2}{\lambda^2 - 0.1193242^2} + \frac{5.3414021\lambda^2}{\lambda^2 - 18.028251^2} \quad (14)$$

$$n_{Si_3N_4}^2 - 1 = \frac{3.0249\lambda^2}{\lambda^2 - 0.1353406^2} + \frac{40314\lambda^2}{\lambda^2 - 1239.842^2} \quad (15)$$

$$n_{SiO_2}^2 - 1 = \frac{0.6961663\lambda^2}{\lambda^2 - 0.0684043^2} + \frac{0.4079426\lambda^2}{\lambda^2 - 0.1162414^2} + \frac{0.8974794\lambda^2}{\lambda^2 - 9.896161^2} \quad (16)$$

$$n_{TiO_2}^2 = 5.913 + \frac{0.2441}{\lambda^2 - 0.0803} \quad (17)$$

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

```
1  # %% compute the chromatic dispersion
2  n_ideal_v=n_ideal*lamdas/lambdas # ideal refractive index at lambda0
3  lamdas_um=lamdas/1000
4  n_TiO2_v = (4.99+1/96.6*lamdas_um**(-1.1)+1/4.6*lamdas_um**(-1.95))**.5 # at every
   ↪ lambda
5  n_Si3N4_v = (1+3.0249/(1-(0.1353406/lambdas_um)**2)+40314/(1-(1239.842/lambdas_um)**2))**.5
   ↪ # at every lambda
6  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
7  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
8
9  fig, axs=plt.subplots()
10 fig.tight_layout()
11 for n_ar in [n_ideal_v,n_Al2O3_v,n_Si3N4_v,n_SiO2_v,n_TiO2_v]:
12     axs.plot(lambdas,n_ar)
13     axs.set_xlabel('$\lambda$ [nm]')
14     axs.set_ylabel('$n$')
15     axs.grid(True,'both')
16     axs.minorticks_on()
17     axs.legend(['ideal', '$Al_2O_3$', '$Si_3N_4$', '$SiO_2$', '$TiO_2$'])
```

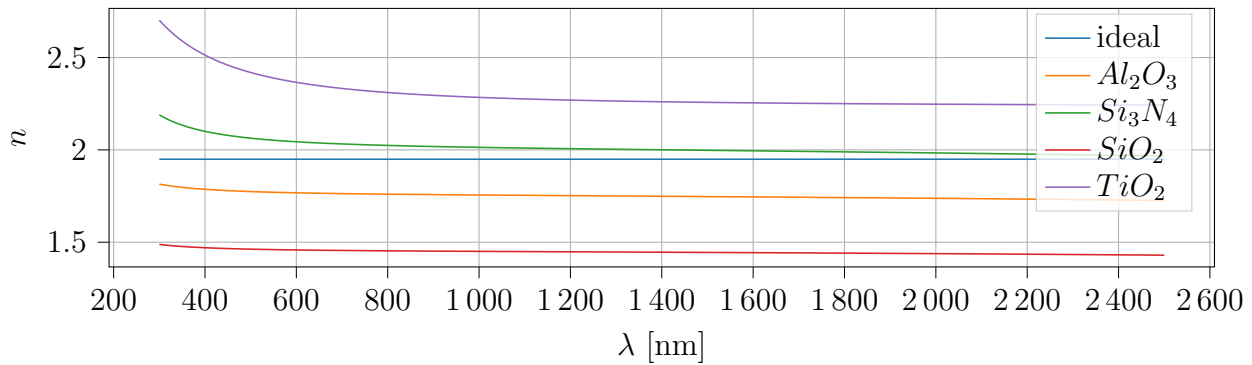
And also computed the input reflectivity considering the chromatic dispersion. As evident in **Figure 5**, the effect of the varying refraction coefficient is negligible for  $Al_2O_3$  and  $Si_3N_4$ , but is quite visible for  $SiO_2$  and  $TiO_2$ .

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 it is:

```
1  # %% check reflectivity - with dispersion
2  fig, axs=plt.subplots()
```





**Figure 4:**  $n$  dependency on  $\lambda$

```

3 fig.tight_layout()
4 for n_ar, d in zip([n_ideal_v, n_Al2O3_v, n_Si3N4_v, n_SiO2_v, n_TiO2_v],
5   ↪ [d_ideal, d_Al2O3, d_Si3N4, d_SiO2, d_TiO2]):
6     Z0=120*np.pi           # free space impedance
7     Z_inf_ar=Z0/n_ar        # impedance of ARC
8     Z_inf_Si=Z0/n_si        # impedance of bulk silicon
9     Gbminus=(Z_inf_Si-Z_inf_ar)/(Z_inf_Si+Z_inf_ar) # reflection coefficient
10    k=K0*n_ar
11    Gaplus=(Gbminus*np.exp(-2j*k*d))
12    Za=Z_inf_ar*(1+Gaplus)/(1-Gaplus)
13    Gaminus=(Za-Z0)/(Za+Z0)
14    R_vect=np.abs(Gaminus)**2*100
15    if n_ar is n_Si3N4_v:
16        R_SLARC=R_vect #save the best design
17    axs.plot(lambdas, R_vect)
18
19 axs.set_xlabel('$\lambda$ [nm]')
20 axs.set_ylabel('Reflectivity [%]')
21 axs.grid(True, 'both')
22 axs.minorticks_on()
23 axs.legend(['ideal', '$Al_{20}3$', '$Si_{3N}4$', '$SiO_2$', '$TiO_2$'])

```

## 6 DLARC design - double layer

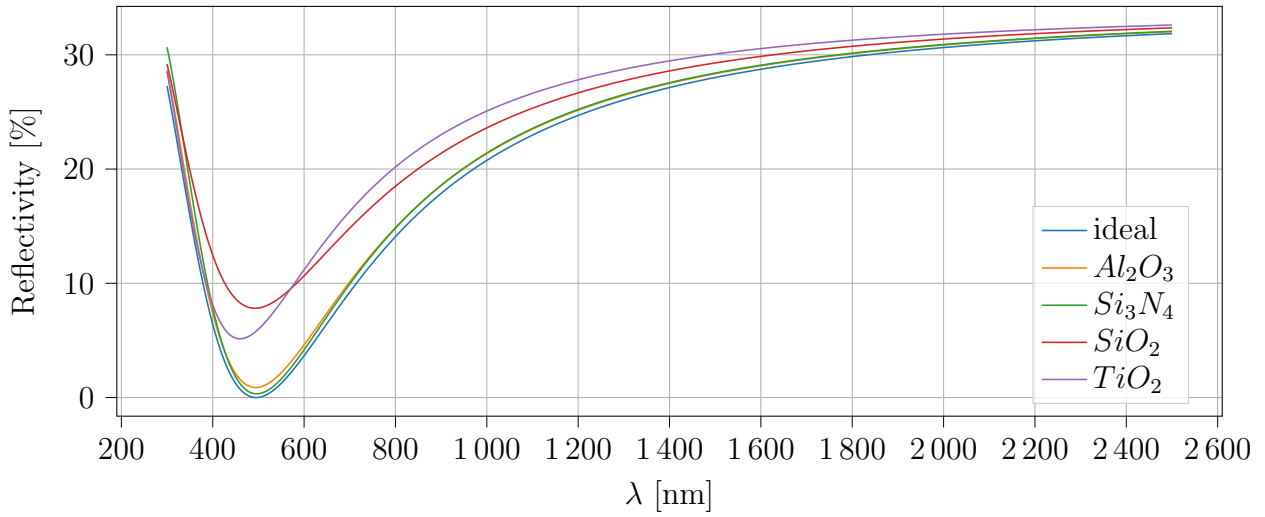
### 6.1 Silica/Titania layers

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.

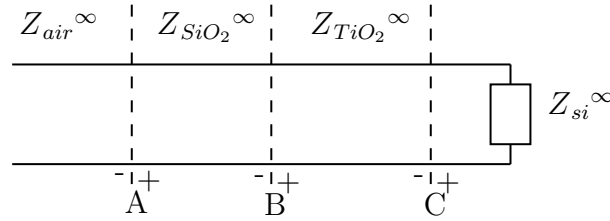
$$d_{TiO_2} = 45.5\text{nm} \quad d_{SiO_2} = 84.6\text{nm}$$

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:



**Figure 5:** Different ARCs considering  $n$  dependency on  $\lambda$



**Figure 6:** Equivalent TL - DLARC

```

1  # %% DLARC - SILICA/TITANIA
2  Z_inf_SiO2=Z0/n_SiO2
3  Z_inf_TiO2=Z0/n_TiO2
4  Gcminus=(Z_inf_Si-Z_inf_TiO2)/(Z_inf_Si+Z_inf_TiO2)
5  k=K0*n_dict['$TiO_2$']
6  Gbplus=Gcminus*np.exp(-2j*k*d_TiO2)
7  Zb=Z_inf_TiO2*(1+Gbplus)/(1-Gbplus)
8  Gbminus=(Zb-Z_inf_SiO2)/(Zb+Z_inf_SiO2)
9  k=K0*n_dict['$SiO_2$']
10 Gaplus=Gbminus*np.exp(-2j*k*d_SiO2)
11 Za=Z_inf_SiO2*(1+Gaplus)/(1-Gaplus)
12 Gaminus=(Za-Z0)/(Za+Z0)
13 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:

```

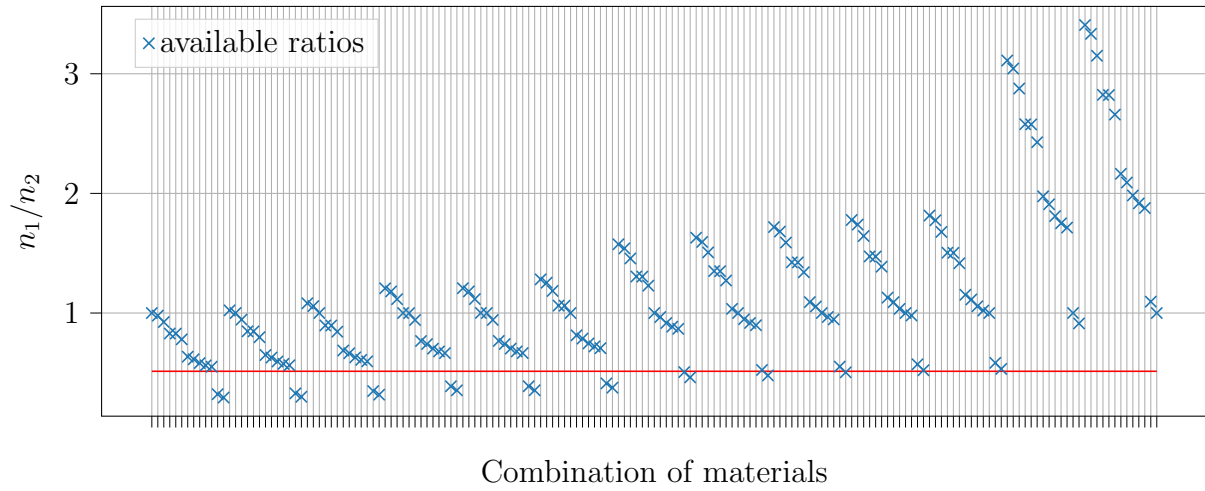
1  # %% DL ARC - some other materials
2  n_dict={"$Na_3AlF_6$": 1.35,
3         "$MgF_2$": 1.38,
4         "$SiO_2$": 1.46,
5         "$Al_2O_3$": 1.629,
6         "$CeF_3$": 1.63,
7         "$PbF_2$": 1.73,
8         "$Ta_2O_5$": 2.126,
9         "$ZrO_2$": 2.20,
10         "$ZnS$": 2.32,
11         "$TiO_2$": 2.40,
12         "$Bi_2O_3$": 2.45,
13         "$Ge$": 4.2,
14         "$Te$": 4.60}
15
16  legend=[]; val=[]
17  for keys_e, values_e in n_dict.items():
18      for keys_i, values_i in n_dict.items():
19          val.append(values_e/values_i)
20          legend.append(keys_e + "/" + keys_i)
21  fig, axs=plt.subplots()
22  fig.tight_layout()
23  axs.plot(legend,val,'x')
24  axs.hlines([(n_air/n_si)**0.5],legend[0],legend[-1],'red')
25  axs.set_xlabel('Combination of materials')
26  axs.set_ylabel('$n_1/n_2$')
27  axs.grid(True,'both')
28  axs.legend(['available ratios', 'reference'])

```

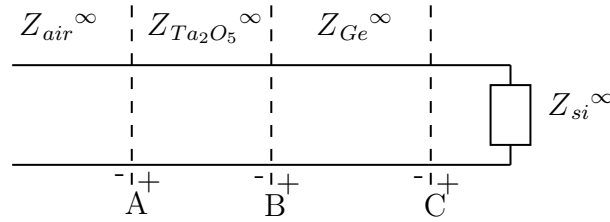
material	$n$	material	$n$
$Na_3AlF_6$	1.35	$ZrO_2$	2.20
$MgF_2$	1.38	$ZnS$	2.32
$SiO_2$	1.46	$TiO_2$	2.40
$Al_2O_3$	1.62	$Bi_2O_3$	2.45
$CeF_3$	1.63	$Ge$	4.20
$PbF_2$	1.73	$Te$	4.60
$Ta_2O_5$	2.12	-	-

**Table 1:** Common materials refractive indeces

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:



**Figure 7:** Different materials  $n$  ratios



**Figure 8:** Equivalent TL - DLARC

```

1  # %% DLARC - Ta2O5 and Ge
2  d_Ge=lambda0/4/n_dict['$Ge$']
3  d-Ta=lambda0/4/n_dict['$Ta_2O_5$']
4  Z_inf_Ge=Z0/n_dict['$Ge$']
5  Z_inf-Ta=Z0/n_dict['$Ta_2O_5$']
6  Gcminus=(Z_inf_Si-Z_inf_Ge)/(Z_inf_Si+Z_inf_Ge)
7  k=K0*n_dict['$Ge$']
8  Gbplus=Gcminus*np.exp(-2j*k*d_Ge)
9  Zb=Z_inf_Ge*(1+Gbplus)/(1-Gbplus)
10 Gbminus=(Zb-Z_inf-Ta)/(Zb+Z_inf-Ta)
11 k=K0*n_dict['$Ta_2O_5$']
12 Gaplus=Gbminus*np.exp(-2j*k*d-Ta)
13 Za=Z_inf-Ta*(1+Gaplus)/(1-Gaplus)
14 Gaminus=(Za-Z0)/(Za+Z0)
15 R_DLARC-Ta2O5Ge=np.abs(Gaminus)**2*100

```

### 6.3 Comparison of designs

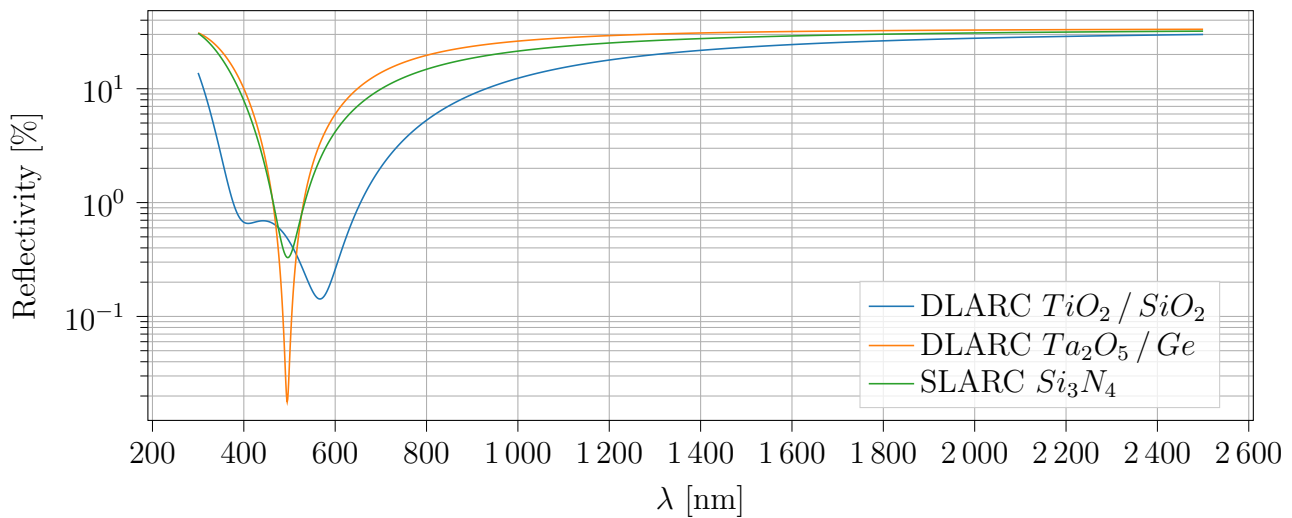
At this point we can compare the reflectivity of the best SLARC design with Silica/Titania and  $Ta_2O_5/Ge$  DLARC designs. The script is the following, and the result are shown in **Figure 9**.

It's evident that the  $Ta_2O_5/Ge$  DLARC outperform both  $TiO_2/SiO_2$  and  $Si_3N_4$  coating at the design wavelength  $\lambda_0$ , because it reaches almost zero reflectivity, but the reflectivity drop at  $\lambda_0$  in a really narrow manner, w.r.t  $TiO_2/SiO_2$ . This may imply that even is  $Ta_2O_5/Ge$  is best at  $\lambda_0$ ,  $TiO_2/SiO_2$  could be best in average considering the whole useful sunlight spectrum.

```

1  # %% check reflectivity - SLARC / DLARCs
2  fig, axs=plt.subplots()
3  fig.tight_layout()
4  axs.semilogy(lambdas,R_DLARC_SiO2TiO2)
5  axs.semilogy(lambdas,R_DLARC-Ta2O5Ge)
6  axs.semilogy(lambdas,R_SLARC)
7  axs.set_xlabel('$\lambda$ [nm]')
8  axs.set_ylabel('Reflectivity [%]')
9  axs.grid(True,'both')
10 axs.minorticks_on()
11 axs.legend(['DLARC $TiO_2 \backslash, / \backslash, SiO_2$', 'DLARC $Ta_{20_5} \backslash, / \backslash, Ge$', 'SLARC $Si_{3N_4}$'])

```



**Figure 9:** *Reflectivity for different designs*

## 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:

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

$R(\lambda)$  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**.

```

1  # %% check effective reflectivity
2  indx=0
3  R_eff=[]; R_min=[]
4  for R in [R_DLARC_SiO2TiO2, R_DLARC-Ta2O5Ge, R_SLARC]:
5      NUM=0; DEN=0
6      for _ in lambdas:

```

```

7         if indx is (len(lambdas)-2):
8             break
9         NUM+=R[indx]*AM15_irr[indx]*(lambdas[indx+1]-lambdas[indx])
10        DEN+=AM15_irr[indx]*(lambdas[indx+1]-lambdas[indx])
11        R_eff.append(NUM/DEN)
12        R_min.append(np.min(R))
13    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:

```

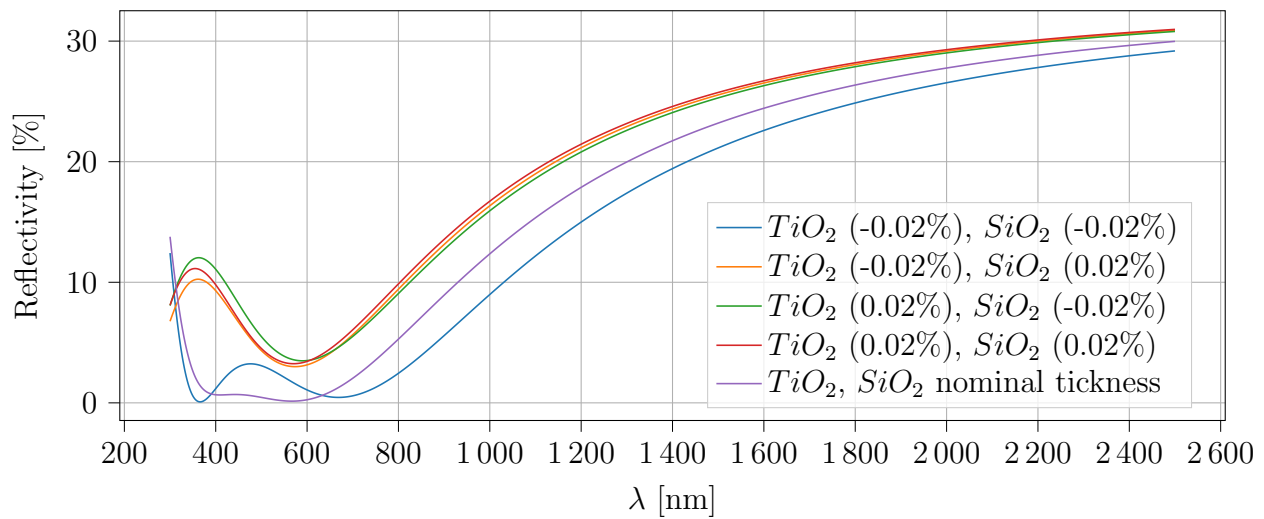
1  # %% DLARC - tolerance effect
2  Z_inf_SiO2=Z0/n_dict['$SiO_2$']
3  Z_inf_TiO2=Z0/n_dict['$TiO_2$']
4  Gcminus=(Z_inf_Si-Z_inf_TiO2)/(Z_inf_Si+Z_inf_TiO2)
5  k=K0*n_dict['$TiO_2$']
6  fig, axs=plt.subplots()
7  fig.tight_layout()
8  legend=[]
9  for err1 in [-0.02, 0.02]:
10     for err2 in [-0.02, 0.02]:
11         Gbplus=Gcminus*np.exp(-2j*k*d_TiO2*(1-err1))
12         Zb=Z_inf_TiO2*(1+Gbplus)/(1-Gbplus)
13         Gbminus=(Zb-Z_inf_SiO2)/(Zb+Z_inf_SiO2)
14         k=K0*n_dict['$SiO_2$']
15         Gaplus=Gbminus*np.exp(-2j*k*d_SiO2*(1-err2))
16         Za=Z_inf_SiO2*(1+Gaplus)/(1-Gaplus)
17         Gaminus=(Za-Z0)/(Za+Z0)
18         R_DLARC_tolerance=np.abs(Gaminus)**2*100
19         axs.plot(lambdas,R_DLARC_tolerance)
20         legend.append('$TiO_2$ ('+str(err1)+'\%'), $SiO_2$ ('+str(err2)+'\%')')
21     axs.plot(lambdas,R_DLARC_SiO2TiO2)
22     legend.append('$TiO_2$', '$SiO_2$ nominal tickness')
23     axs.minorticks_on()

```

```

24  axs.set_xlabel('$\lambda$ [nm]')
25  axs.set_ylabel('Reflectivity [%]')
26  axs.grid(True, 'both')
27  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 Lukowiak, 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 Si<sub>3</sub>N<sub>4</sub> microresonator. *Optics Letters*, 40(21), 2015.