Project
ELG3106 – Fall 2025

Section: A00
Name: Robert Gauvreau

Date Assigned: Sunday October 4th, 2025
Due Date: Saturday November 12th, 2025

Table of Contents

1. Introduction:

This design study examines how broadband antireflection (AR) coatings on a solar cell affect overall electrical power production. This experiment will compare double-layer and triple-layer coatings, and show how layer refractive indices and quarter-wave thicknesses change the wavelength-dependent reflectivity/transmissivity and, through the solar spectrum, the integrated power delivered to the cell. Beyond photovoltaics, the same multilayer-coating principles are used in camera lenses, optical sensors, displays, laser cavities, and telecommunications filters, where minimizing reflection for specific wavelengths is critical.

Incident sunlight is modeled as a normally-incident uniform plane wave with an idealized solar spectral irradiance ("blackbody-like"):

$$I(\lambda) = \frac{6.16 \times 10^{15}}{\lambda^5 \left(\exp\left(\frac{2484}{\lambda}\right) - 1\right)} (\mathrm{W/m^2/nm}),$$

With λ in nm. The cell is idealized to convert all transmitted light to electricity, so electrical power density is

$$P = \int_{\lambda_1}^{\lambda_2} T(\lambda)\, I(\lambda)\, d\lambda. \qquad , \lambda_1 = 200 \text{ nm}, \qquad \lambda_2 = 2200 \text{ nm}.$$

$$\text{For } T(\lambda) = 1, \qquad P = 1000 \frac{W}{m^2}$$

In all designs, each coating layer uses a quarter-wave optical thickness at the center wavelength $\lambda_C = 650$ nm:

$$d_m = \frac{\lambda_C}{4\, n_m} (\text{normal incidence}).$$

Theory: Transfer Matrix Method (TMM) at normal incidence

Consider N lossless dielectric layers between semi-infinite media n_0 (incidence) and n_{N+1} (substrate). Let E_m^+ and E_m^- be forward/backward field amplitudes just to the left of interface m.

Interface ("dynamical") matrix:

At the interface between media m and $m + 1$,

$$\begin{bmatrix} E_m^+ \\ E_m^- \end{bmatrix} = Q_{m,m+1} \begin{bmatrix} E_{m+1}^+ \\ E_{m+1}^- \end{bmatrix}, \qquad Q_{m,m+1} = \frac{1}{\tau_{m,m+1}} \begin{bmatrix} 1 & \Gamma_{m,m+1} \\ \Gamma_{m,m+1} & 1 \end{bmatrix},$$

With Fresnel coefficients (normal incidence)

$$\Gamma_{m,m+1} = \frac{n_m - n_{m+1}}{n_m + n_{m+1}}, \qquad \tau_{m,m+1} = \frac{2n_m}{n_m + n_{m+1}}.$$

Propagation matrix (layer m)

Within layer m of thickness d_m and refractive index n_m, the forward/backward waves acquire a phase δ_m:

$$\begin{bmatrix} E_m^{+'} \\ E_m^{-'} \end{bmatrix} = P_m \begin{bmatrix} E_m^+ \\ E_m^- \end{bmatrix}, \qquad P_m = \begin{bmatrix} e^{\,j\delta_m} & 0 \\ 0 & e^{-\,j\delta_m} \end{bmatrix}, \delta_m = \frac{2\pi n_m d_m}{\lambda}.$$

(Primes denote fields on the layer’s far side.)

Total transfer matrix and global coefficients:

Cascading the N layers and N +1 interfaces gives the system matrix

$$\begin{bmatrix} E_0^+ \\ E_0^- \end{bmatrix} = T \begin{bmatrix} E_{N+1}^+ \\ E_{N+1}^- \end{bmatrix}, \quad T = \prod_{m=0}^{N} (Q_{m,m+1} P_{m+1}) \qquad \text{(with } P_{N+1} := I \text{ for the substrate).}$$

With an incident wave $E_0^+ = E_i$, reflected$E_0^- = E_r$, and transmitted $E_{N+1}^+ = E_t$ into the substrate (no backward wave there), the overall reflection and transmission coefficients are obtained from $T = \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix}$:

$$\Gamma = \frac{E_r}{E_i} = \frac{T_{21}}{T_{11}}, \qquad \tau = \frac{E_t}{E_i} = \frac{1}{T_{11}}.$$

Power reflectance and transmittance (for nonmagnetic, lossless media) are

$$R(\lambda) = |\,\Gamma\,|^2, T(\lambda) = \frac{n_{N+1}}{n_0} \;|\,\tau\,|^2\,, \qquad R(\lambda) + T(\lambda) = 1.$$

These relations are the basis for computing spectra and the power integral P.

Part 1

We frame the optical problem at normal incidence: an incident plane wave is shined on a two-layer anti-reflective coating on a solar cell. Using the transfer-matrix method (TMM), we derive closed-form expressions for reflection Γ, transmission τ, and the resulting power reflectance/transmittance $R(\lambda) =| \Gamma |^2$, $T(\lambda) = \frac{n_{\text{cell}}}{n_0} | \tau |^2$. At the design wavelength λ_C (quarter-wave condition), we work out the analytical reflectivity for the double-layer coating and the semiconductor interface to see how the coating changes the boundary conditions. We then translate that theory into a short MATLAB routine that evaluates $R(\lambda)$ near $\lambda_C$vand confirms the algebra numerically. Finally, with a fixed n_1 (e.g., constrained by available material), we derive the closed-form relationship for n_2 that cancels reflection at λ_C and compare it to our simulation for consistency.

Part 2

Next, we move beyond a single wavelength and compute the full reflectivity spectrum over a visible/near-IR band (e.g., 400–1400 nm). Using the same TMM approach, we generate $R(\lambda)$ and convert it to transmitted power by weighting with the provided solar irradiance model $I(\lambda)$ and integrating $P = \int T(\lambda)\, I(\lambda)\, d\lambda$ over one or more wavelength ranges (e.g., 200-2200 nm and 400–1400 nm). We then experiment by adjusting n_1, n_2, and/or the layer thickness (kept quarter-wave at λ_C, or slightly detuned) to see whether the power increases. Using the plots for power and reflectance, we can observe and create a more optimized model by varying the refractive index and the layer thickness.

Part 3

We repeat the analytic treatment for a triple-layer coating. Under the quarter-wave condition, we cascade three interface matrices and three propagation matrices to form the total matrix T. Setting $\Gamma(\lambda_C) = 0$ yields a symbolic condition that relates the three film indices to n_0 and n_{cell}. With one or two indices fixed (material constraints), we solve for the remaining index (often n_2) and report its value (e.g., to two significant digits). We also discuss the “impedance-taper” choice (geometric progression of indices) that often broadens the low-reflectance band, and we contrast that with the fixed-material case used earlier.

For Part 3, we can repeat the same approach as for Part 1 but considering an additional layer. We must first describe the relationship for n_2 analytically at the center wavelength $\lambda_c = 650nm$

Part 4

We implement the three-layer TMM in code and wrap it with a sweep/optimization over the free index (commonly n_2) and, if allowed, small detunes in layer thicknesses while keeping n_0, n_1, n_3, and n_{cell} fixed. For each candidate, we compute the spectrum, convert it to transmitted power with the same $I(\lambda)$, and pick the arg-max design. We then compare the numerically optimal index with the analytic prediction from Part 3 at λ_C. If they differ, we explain the difference in terms of broadband weighting (the power integral depends on the whole spectrum, not just the single-wavelength null), manufacturing constraints, or dispersion.

2. Results and Discussion:

Part 1 – (a)

Given $n_{cell} = 3.5$, $n_0 = 1\ (air)$ and $\lambda_C = 650nm$, we can calculate the transmitted power with no anti-reflective coating specifically at the central wavelength.

$$\Gamma = \frac{n_{cell} - n_0}{n_{cell} + n_0} \quad (at\ normal\ incidence)$$

$$\Gamma = \frac{3.5 - 1}{3.5 + 1} \approx 0.556$$

$$R = |\Gamma|^2 = |0.556|^2 \approx 0.309. \quad (reflectivity)$$

$$P = T(\lambda)I(\lambda) = \frac{(1 - R)6.16 \times 10^{15}}{\lambda_C^5 \left(\exp\left(\frac{2484}{\lambda_C}\right) - 1\right)} \approx 0.8212 \frac{W}{m^2}$$

Table 1: Analytical Results for Reflectivity, Transmittivity and Transmitted Power

Reflectivity	0.309	31%
Transmittivity	0.691	69%
Transmitted Power	$.82\ W/m^2$	-

Part 1 – (b)

The Transfer Matrix Method (TMM) can be used to solve for the overall reflection and transmission of a multi-layer coating of thin dielectric layers at normal incidence as shown below in *Figure 2.1*.

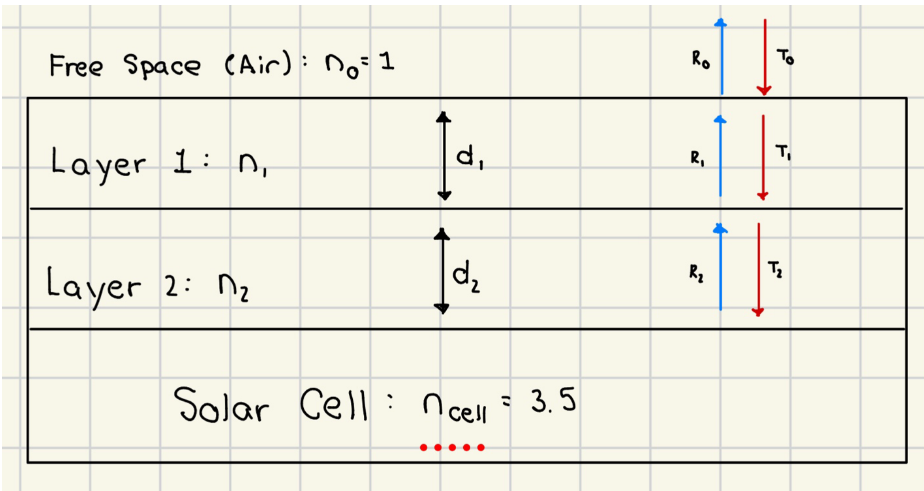

Figure 2.1: Double-layer coating 2-D model with specified parameters.

Each layer, and interface can be represented by a simple 2x2 matrix that relates the forward and backward electric field amplitudes on both sides of each layer. By multiplying these matrices in the correct order, the total system matrix T can be obtained. This allows us to compute the overall reflection and transmission coefficients.

For a multilayer system (e.g., air → layer 1 → layer 2 → cell), the total field relationship can be expressed as:

$$\begin{bmatrix} E_0^+ \\ E_0^- \end{bmatrix} = T \begin{bmatrix} E_{N+1}^+ \\ E_{N+1}^- \end{bmatrix}$$

where,

$$T = Q_{01} P_1 Q_{12} P_2 Q_{23}.$$

Important definitions:

- Q_{mn} - interface matrix between the medias *m* and *n*, containing the reflection and transmission coefficients Γ_{mn} and τ_{mn} pertaining to the particular layer:

$$Q_{mn} = \frac{1}{\tau_{mn}} \begin{bmatrix} 1 & \Gamma_{mn} \\ \Gamma_{mn} & 1 \end{bmatrix}$$

- P_i - propagation matrix for layer *i*, which also accounts for the phase delay across that particular layer:

$$P_i = \begin{bmatrix} e^{j\delta_i} & 0 \\ 0 & e^{-j\delta_i} \end{bmatrix}, \qquad \delta_i = \frac{2\pi n_i d_i}{\lambda}.$$

The overall reflection and transmission of the entire system can be calculated from the elements of the total matrix T:

$$\Gamma = \frac{T_{21}}{T_{11}}, \qquad \tau = \frac{1}{T_{11}},$$

the corresponding reflectivity and transmissivity are

$$R = |\Gamma|^2, \qquad T = \frac{n_{\text{cell}}}{n_0} |\tau|^2.$$

The in-depth mathematical solving's can be found in *Appendix A*.

MATLAB Implementation:

1. Define material parameters:

   Assign the refractive indices for each medium:
   $n_0 = 1.0$ (air), n_1, n_2, $n_{\text{cell}} = 3.5$.

2. Compute the interface matrices Q_{mn}:

   Using the Fresnel formulas for Γ_{mn} and τ_{mn}.

3. Compute the propagation matrices P_i:

   Each layer's phase delay $\delta_i = \frac{2\pi n_i d_i}{\lambda}$ is evaluated for the center wavelength (or later across the spectrum).

4. Compute *T* matrix:

   Multiply $Q_{01} P_1 Q_{12} P_2 Q_{23}$ in order to obtain the system matrix *T*.

5. Obtain coefficients:

   From *T* we can calculate $\Gamma = T_{21}/T_{11}$ and $\tau = 1/T_{11}$.

6. Compute reflectivity and transmissivity:

   Solve $R = |\Gamma|^2$ and $T = (n_{\text{cell}}/n_0)\,|\tau|^2$ to find how much power is reflected and transmitted.

Part 1 – (c)

Calculations to be found in *Appendix B*

$$n_2 = \sqrt{n_{\text{cell}} n_1^2}$$

$$n_2 = \sqrt{3.5 * 1.4^2} \approx 2.62$$

Part 1 – (d)

Using the formula obtained in Part 1 – (c), we can verify this result by plotting the Reflectivity (as a percentage) vs. different refractive indices for n_2. We can observe that our analytical theory is in-line with the experiment. This can be found in *Figure 2.2 and 2.3* below.

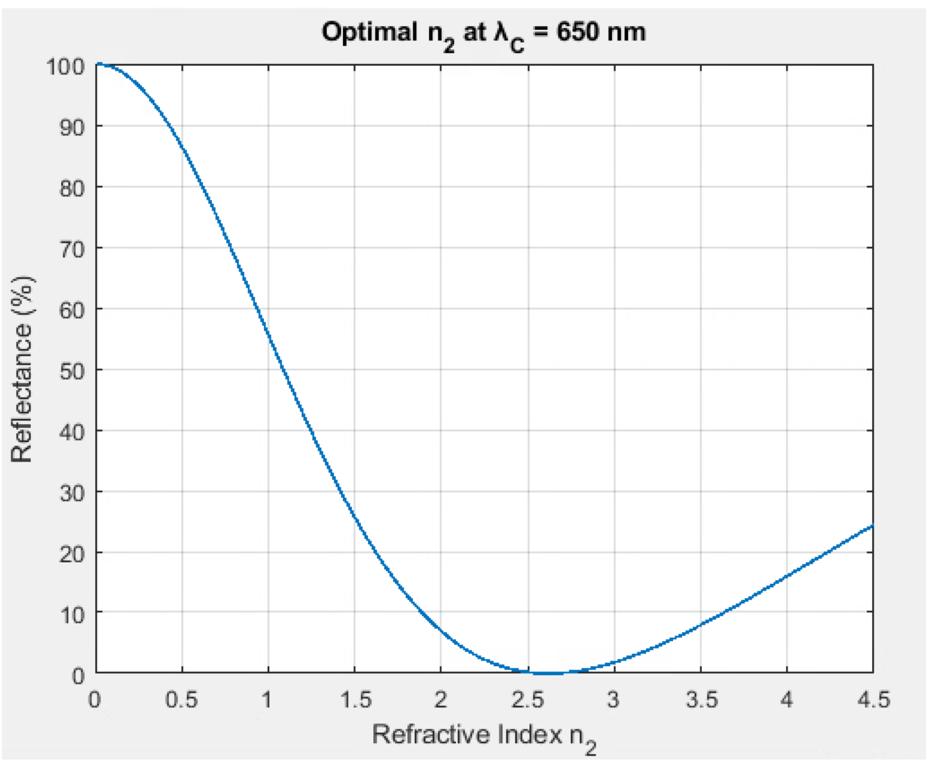

Figure 2.2: Plot of the Reflectivity (as percentage) vs. n_2 for optimal value of n_2 (double-layer).

```
n_1 = 1.40
Optimal n_2 = 2.62 (minimum reflectance)
```

Figure 2.3: Simulated optimal value of n_2 (double-layer)

Part 2 – (a)

Figure 2.4 below displays the relation of the Reflectivity (as a percentage) vs. the wavelength. We can observe that due to our choice of $n_2 = 2.62$ for the previous parts, we are able to obtain a a very low reflectivity at the center wavelength. This is beneficial as we can design our system to be more efficient depending on the wavelengths needed to be received.

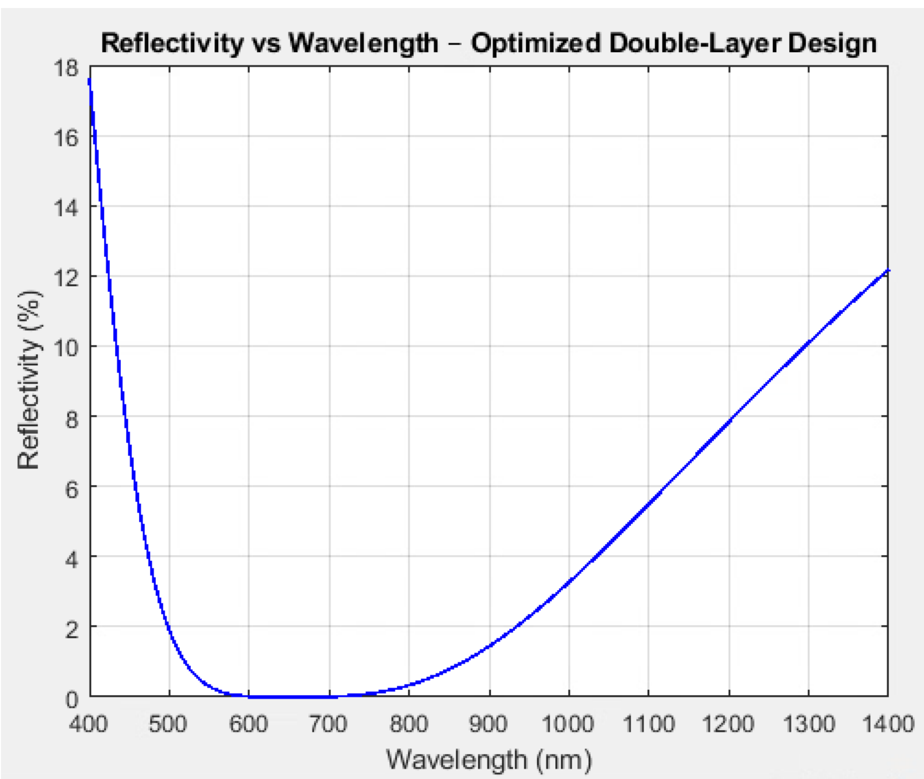

Figure 2.4: Plot of the reflectivity (%) vs. wavelength for the range 400-1400nm

Part 2 – (b)

Using the power equation defined in the introduction, we can utilize this to compute the total power over the defined wavelength interval. The total power for the range from $\lambda_1 = 400nm$ to $\lambda_2 = 1400nm$ was calculated to be 751.202W. The total power for the range from $\lambda_1 = 200nm$ to $\lambda_2 = 2200nm$ was calculated to be 929.592W. The plots of these can be found in *Figure 2.5* below and the values are displayed in *Figure 2.6 and 2.7*.

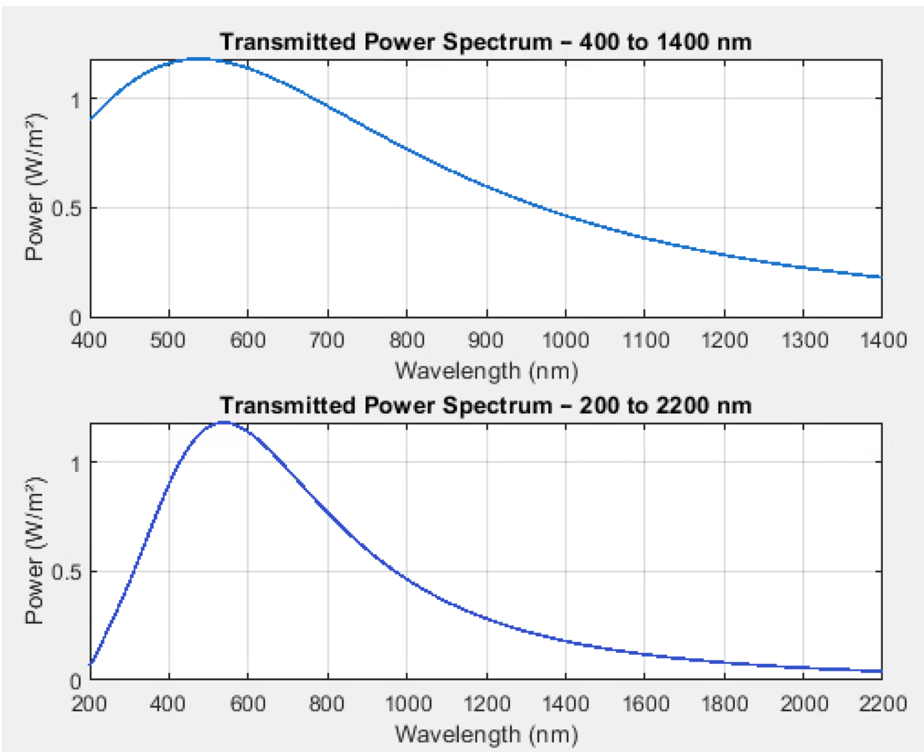

Figure 2.5: Plots of the Transmitted Power vs. Wavelength for 400-1400nm and 200-2200nm (double-layer).

```
Total Power in Watts = 751.202068
```

Figure 2.6: Simulated result of the Transmitted Power vs. Wavelength for 400-1400nm.

Total Power in Watts = 929.591599

Figure 2.7: Simulated result of the Transmitted Power vs. Wavelength for 200-2200nm.

Part 2 – (c)

Refractive indexes and center wavelength used in the optimized configuration *Part 2 – (b).*

$$n_1 = 1.4, \qquad n_2 = 2.62, \qquad n_3 = 3.5, \qquad \lambda_c = 650 \text{ nm},$$

Each coating behaves approximately as a quarter-wave plate which minimizes reflection at the designated wavelength ($\lambda_c = 650$ nm). As observed in *Figure 2.5.*, the design already yields a high-power transmission in the visible range ($\sim 400 - 700 nm$), but this can be further improved by adjusting either the refractive indices or the optical thicknesses of the layers.

1. Varying Refractive Indices

The ideal anti-reflective behavior occurs when the refractive index of each layer satisfies the geometric mean condition (Raut, Venkatesan, Nair, & Ramakrishna, 2011):

$$n_1 = (n_0 n_2)^{\frac{1}{2}}, \qquad n_2 = (n_1 n_3)^{\frac{1}{2}}.$$

This ensures that the impedances between interfaces match and that we have minimal reflection across each interface. In the optimized design, $n_1 = 1.4$ and $n_2 = 2.62$ already approximate this condition. However, if materials with slightly different indices are chosen—e.g.,

$$n_1 = 1.5, \qquad n_2 = 2.3,$$

The reflectivity spectrum slightly shifts, and the reflectivity minimum will broaden around the visible band. This broader bandwidth slightly increases the integrated power across the solar spectrum (400–1400 nm).

2. Adjusting Layer Thickness

The thickness d_i of each layer determines the path that light travels:

$$\delta_i = \frac{2\pi n_i d_i}{\lambda}.$$

In the original design, both coatings are quarter-wave layers ($\delta_i = \pi/2$) at 650 nm.
If we adjust the thicknesses so that the destructive interference condition occurs at slightly shorter wavelengths (e.g., $\lambda_c = 600$ nm), the transmittivity increases and centers itself closer to the EM waves of the solar spectrum, which contains more energy. This will then increase the overall transmitted power when integrated over the designated spectrum.

Simulation Results:

We were effectively able to optimize these parameters within our code for the wavelength interval from 400-1400nm which revealed interesting data. When comparing *Figure 2.8 to Figure 2.5*, we notice an increase in power of 7.686W. The behaviour of the optimized plot changes significantly around our center frequency. This is due to the effect of maximizing the power transmission. The sun doesn't uniformly emit these wavelengths and instead has specific wavelengths where its emissions are more concentrated. We can also observe that the reflectivity over the entire spectrum has slightly decreased for our optimized version which also leads to this additional power consumption.

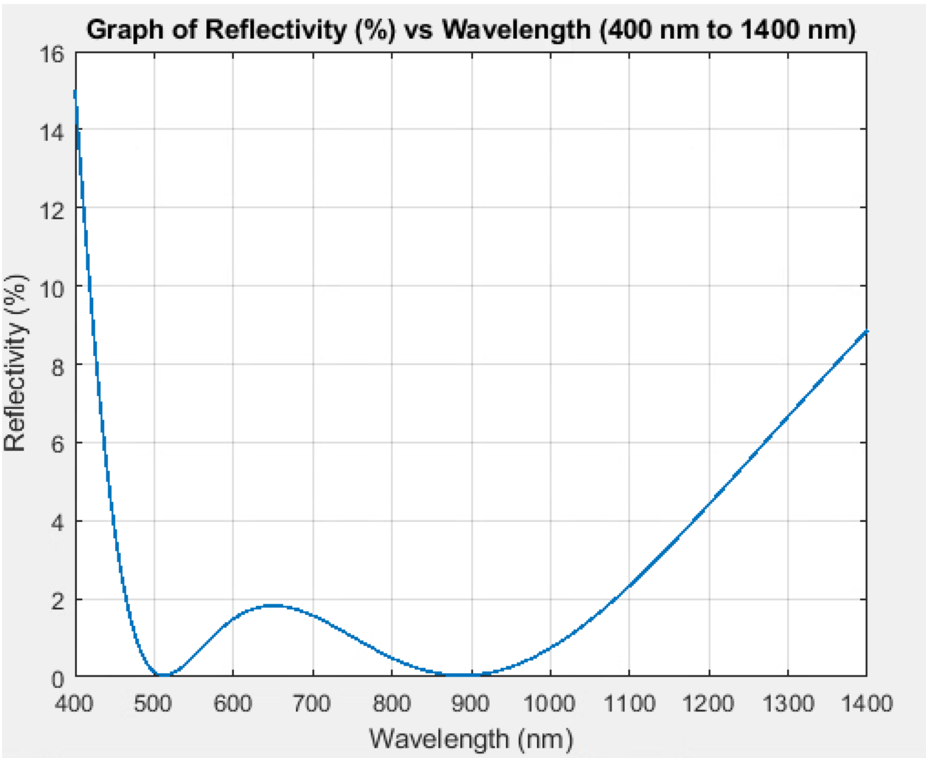

Figure 2.8: Plot of the Transmitted Power vs. Wavelength for 400-1400nm for optimized power (double-layer).

```
Part2_c (400-1400nm)
n_1 = 1.45
n_2 = 2.40
Total Power in Watts = 758.8876
```

Figure 2.9: Results of the Transmitted Power vs. Wavelength for 400-1400nm for optimized power (double-layer).

Part 3 – (a)

In Part 3, we begin implementing a third layer and observing how an additional layer can assist in optimizing the power and reflection. We can use a method similar to that of the 2-layer system and continue using the transfer matrix method to determine the overall transmission. The *Figure 2.10* below demonstrates the new model with the additional layer.

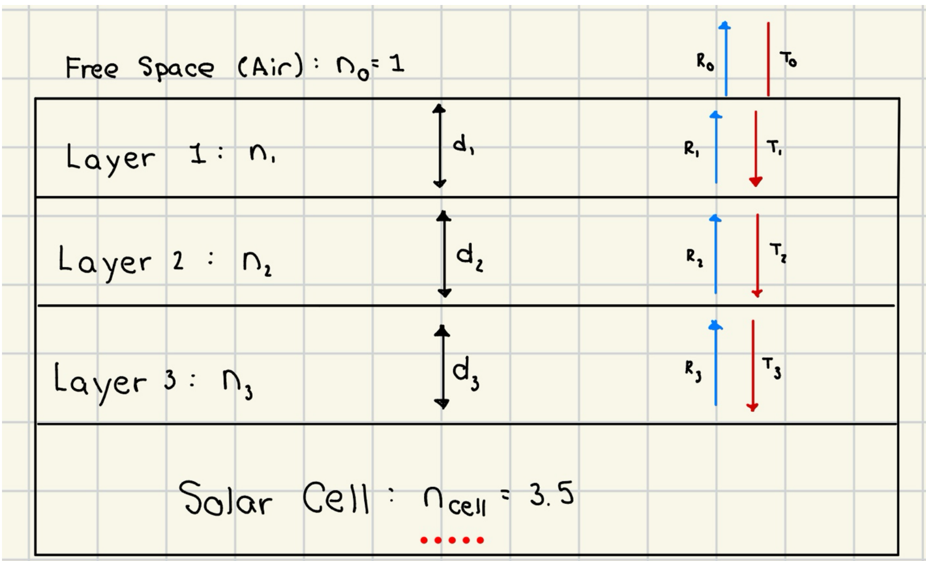

Figure 2.10: Triple-layer coating 2-D model with specified parameters.

The calculations to find n_2 can be found in *Appendix C*.

$$n_2 = \frac{n_1\, n_3}{\sqrt{n_0\, n_4}}.$$

Part 3 – (b)

We can use the derivations found in *Part 3 – (a)* to determine a numerical value for n_2 that optimizes reflectivity at the center wavelength. Given $n_0 = 1\ (air), n_1 = 1.4\ and\ n_3 = 3.15\ and\ n_{cell} = 3.5$:

$$n_2 = \frac{1.4 * 3.15}{\sqrt{1 * 3.5}} = 2.36$$

The value at which n_2 is minimized at the central wavelength is 2.36.

Part 3 – (c)

Using the derivation from *Part 3 – (a)* and the numerical value from *Part - 3 (c)*, we can verify this using a simulation that sweeps over all the values of n_2. The plot below in Figure 2.11 confirms our findings from the previous steps that our optimal refractive index at the center wavelength should be 2.4 (for 2 significant digits).

Commented [RG1]: format proeprly to 2 sig figs

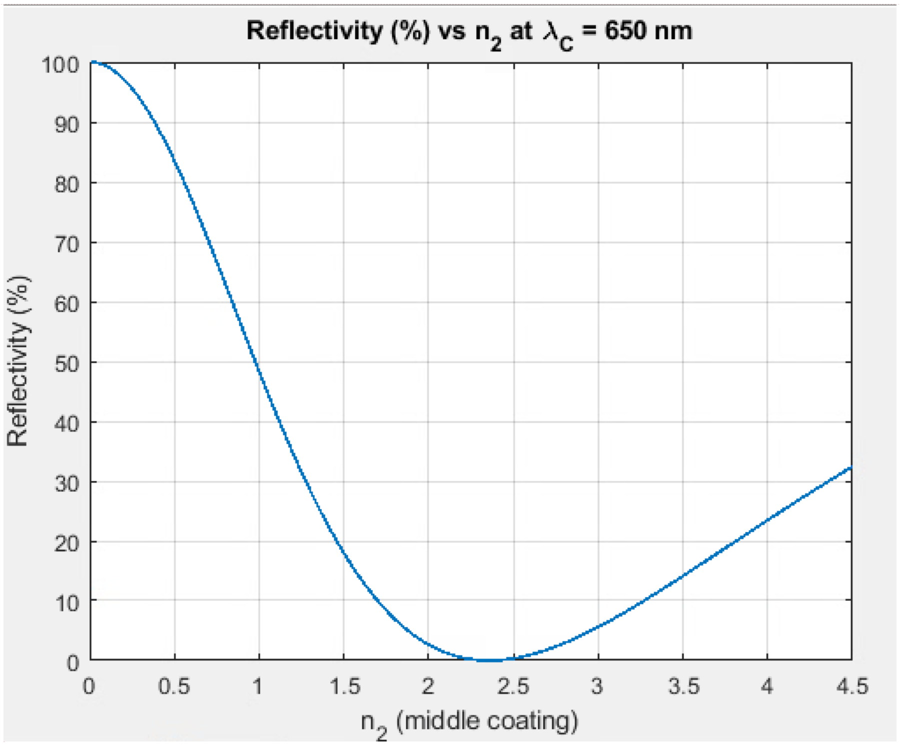

Figure 2.11: Plot of the Reflectivity (as a percentage) vs. n_2 for optimal value of n_2 (triple-layer).

```
Part3_c
n_1 = 1.40
n_2 = 3.15
Minimum reflectivity at n_2 = 2.36
```

Figure 2.12: Result of the Reflectivity (as a percentage) vs. n_2 for optimal value of n_2 (triple-layer).

Part 4 – (a)

Commented [RG2]: add answer

Using the refractive indices obtained in the previous parts, we will now sweep n_2 to determine where the maximum power occurs. After simulating, we notice a parabolic shape where the maximum power can be defined at a distinct value for n_2. These observations can be determined from *Figure 2.13* below

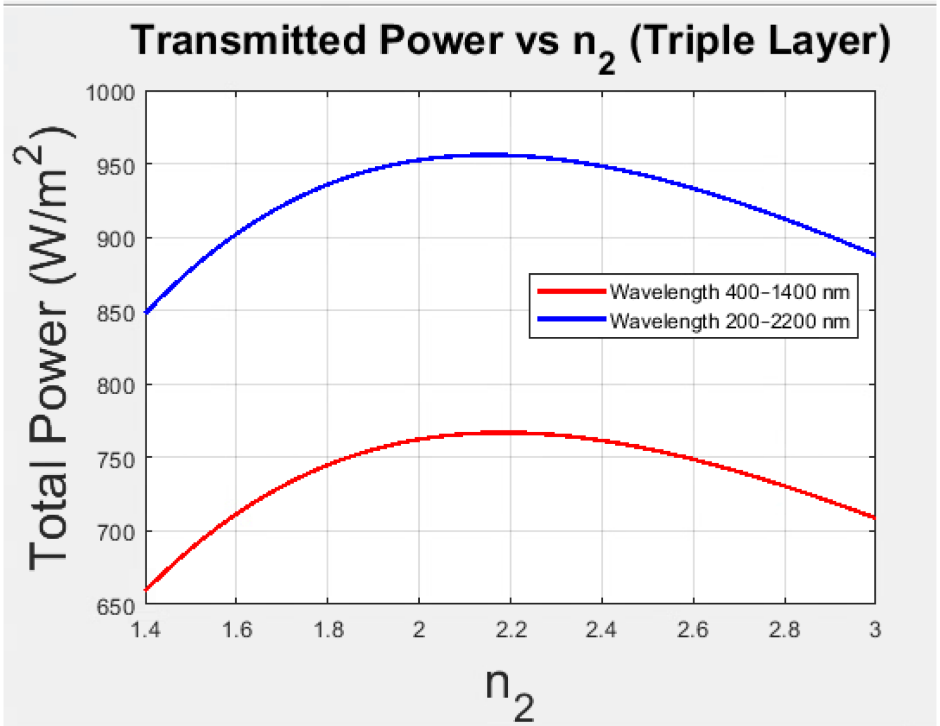

Figure 2.13: Plot of Total Transmitted Power vs. n_2 to determine maximum power (triple-layer)

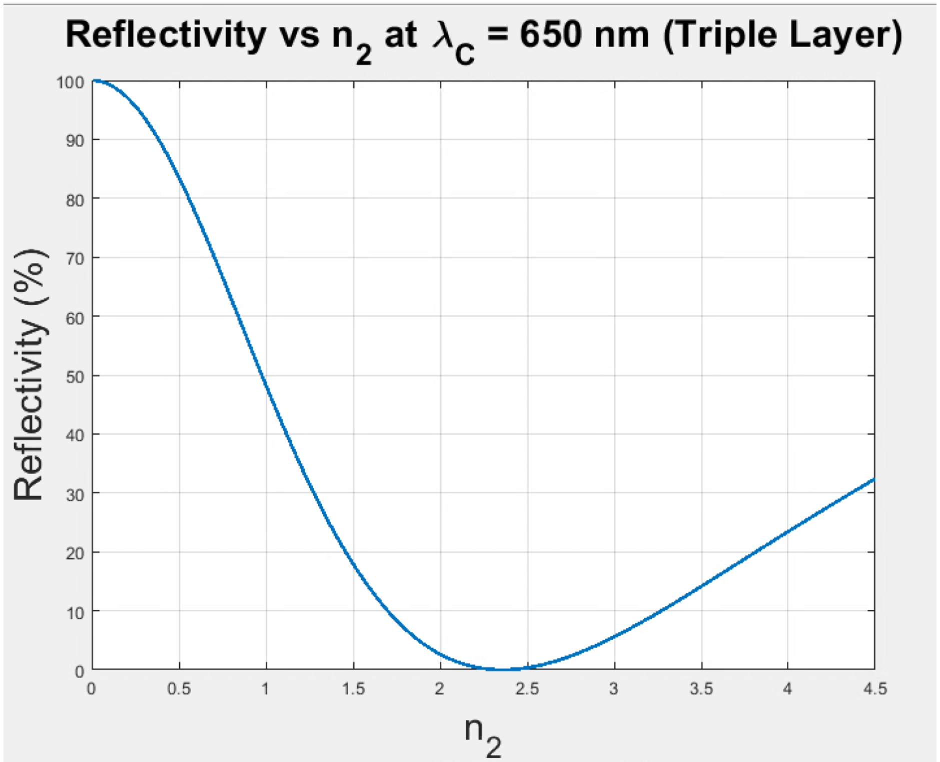

Figure 2.13: Plot of Reflectivity vs. n_2 to determine optimal n_2 (triple layer)

Part 4 – (b)

Based on *Figure 2.13* above and the results on *Figure 2.14* below, we can observe that this occurs at the 2.18 range for 400-1400nm and 2.15 for 200-2200nm.

```
n_1 = 1.40, n_3 = 3.15, n_4 = 3.50
Minimum Reflectivity at λ_C: n_2 = 2.36

Max Power (400-1400 nm): 766.95 W/m^2 at n_2 = 2.18
Max Power (200-2200 nm): 955.95 W/m^2 at n_2 = 2.15
```

Figure 2.14: Results of the maximum power obtained for different values of n_2 for 400-1400nm and 200-2200nm.

Part 4 – (c)

The values obtained for n_2 that yield the maximum transmitted power differ from those predicted by the analytical approach. This discrepancy arises from several factors. In the analytical method, the anti-reflective layer was optimized specifically for the central wavelength $\lambda_C = 650$ nm. While optimizing at this wavelength ensures strong transmission at that point, it does not represent the performance across the entire spectrum. In contrast, the simulated optimization program evaluates the full wavelength range and incorporates a significantly larger number of parameters in the power calculation, resulting in more comprehensive and accurate results.

3. Conclusion:

This project investigated the performance of anti-reflective coatings for solar cells using both double-layer and triple-layer configurations, analyzed through the Transfer Matrix Method (TMM) and MATLAB simulations.

In the initial single-interface model between air and the semiconductor, approximately 31% of incident light was reflected which confirmed the need for coatings to improve light transmission. When a double-layer anti-reflective coating was applied and optimized for the central wavelength of $\lambda_C = 650$ nm, reflectivity decreased significantly at that specific wavelength. However, when analyzing the entire solar spectrum, this improvement did not translate to maximum power transmission. Similarly, the triple-layer system produced even lower reflectivity around the design wavelength but did not maximize total transmitted power across all wavelengths.

For *Part 4(c)*, the refractive indices obtained from the analytical model and the MATLAB optimization were found to differ. The analytical method produced a minimum reflectivity at $n_2 = 2.36$ for $\lambda_C = 650$ nm, while the spectrum-based optimization yielded $n_2 = 2.18$ for 400–1400 nm and $n_2 = 2.15$ for 200–2200 nm. This discrepancy occurs because the analytical method optimizes only at the central wavelength under idealized conditions, whereas the MATLAB program integrates power over the entire wavelength range, accounting for the full solar spectrum. As a result, the simulation provides a more realistic evaluation of the coating's overall performance and better reflects real-world solar conditions.

These findings demonstrate that while wavelength-specific optimization can effectively minimize reflection at a single frequency, it does not necessarily maximize energy output over the solar spectrum. A full-spectrum approach is therefore essential when designing anti-reflective coatings for solar cells to ensure optimal power generation.

Although this study was primarily theoretical and computational, it highlights the importance of bridging the gap between analytical models and practical implementation. Future work should include non-ideal spectral irradiance, material absorption, temperature variations, and oblique incidents effects to further improve the accuracy and applicability of these designs.

In summary, this project establishes that the most effective anti-reflective coating design is achieved through spectrum-wide optimization rather than single-wavelength tuning. By considering the entire solar spectrum, coatings can be designed to significantly enhance solar cell efficiency and energy conversion in real-world conditions.

4. References

Raut, H. K., Venkatesan, A. G., Nair, A. S., & Ramakrishna, S. (2011). *Anti-reflective coatings: A critical, in-depth review*. Energy & Environmental Science, 4(10), 3779–3804. https://doi.org/10.1039/C1EE01297E

Schreimer, H. (2025). *ELG3106: Design study – impact of broadband antireflection coating design on solar power production*. University of Ottawa, Faculty of Engineering, School of Electrical Engineering and Computer Science.

Schriemer, H. (2025). *ELG3106: Module 5 — Transmission lines*. University of Ottawa, Faculty of Engineering, School of Electrical Engineering and Computer Science.

Schriemer, H. (2025). *ELG3106: Module 6 — The lossless transmission line*. University of Ottawa, Faculty of Engineering, School of Electrical Engineering and Computer Science.

5. Appendix:

Part 1 – (d) – Code, Flowchart and Virtual Simulations

Flow Chart for *Figure 5.1-5.3*

1. Define material refractive indices:
   - n_air (incident medium),
   - n_1 (first coating layer),
   - n_3 (second coating layer),
   - n_cell (substrate).
2. Define the center wavelength λ_C = 650 nm.
3. Compute reflection and transmission coefficients for **Interface 0–1 (Air → Layer 1)**:
   - Calculate r01 and t01.
   - Build the interface matrix Q01.
4. Compute reflection and transmission coefficients for **Interface 3–Substrate (Layer 3 → Cell)**:
   - Calculate r35 and t35.
   - Build the interface matrix Q35.
5. Define the **phase matrix P** for a quarter-wave layer:
   - Compute phase shift $\Delta = \pi / 2$.
   - Construct P = [exp(jΔ) 0; 0 exp(–jΔ)].
6. Initialize a sweep over possible refractive indices for the middle layer (n_2):
   - Define n_2_values = 0 : 0.01 : 4.5.
   - Preallocate an array “Reflectance” with zeros.
7. Begin loop: for each n_2 in n_2_values →
   a. Compute reflection and transmission coefficients for **Interface 1–2 (Layer 1 → Layer 2)**.
   b. Build corresponding matrix Q12.
   c. Compute reflection and transmission coefficients for **Interface 2–3 (Layer 2 → Layer 3)**.
   d. Build corresponding matrix Q23.
   e. Build the total transfer matrix for the system in order:
   Air → Layer 1 → Layer 2 → Layer 3 → Substrate, including propagation phases.
   T_total = Q01 × P × Q12 × P × Q23 × P × Q35.
   f. Compute overall reflection coefficient Γ = T_total(2,1)/T_total(1,1).
   g. Compute reflectance = $|\Gamma|^2$ and store in the array Reflectance(k).
8. End the loop when all n_2_values have been tested.
9. Identify the index where Reflectance is minimum.
10. Determine the optimal n_2 corresponding to that minimum reflectance.
11. Print the optimal refractive index n_2 and its corresponding parameters.
12. Plot Reflectance (%) versus refractive index n_2.

Code for *Figure 5.1-5.3*

```
clear; clc; close all;

%Material refractive indices
n_air   = 1.0;  % Refractive index of air (incident medium)
n_1     = 1.4;  % First coating layer
n_3     = 3.5;  % Second coating layer
n_cell  = 3.5;  % Solar cell (substrate) refractive index

lambdaC = 650;      % Center wavelength (nm)

%Interface 0-1: Air to first layer
r01 = (n_air - n_1) / (n_air + n_1);     % Reflection coefficient
t01 = 2 * n_air / (n_air + n_1); % Transmission coefficient
Q01 = (1 / t01) * [1 r01; r01 1]; % Interface (dynamical) matrix

%Interface 3-Substrate
r3S = (n_3 - n_cell) / (n_3 + n_cell);
t3S = 2 * n_3 / (n_3 + n_cell);
Q3S = (1 / t3S) * [1 r3S; r3S 1];

%Phase matrix for quarter-wave layer
Delta = pi / 2;  % Optical phase thickness (λ/4)
P = [exp(1j*Delta) 0; 0 exp(-1j*Delta)];

%Sweep n_2 to locate minimum reflectance
n2_values = 0:0.01:4.5;   % Candidate refractive indices
Reflectance = zeros(size(n2_values));% Allocate results

for k = 1:numel(n2_values)
    n_2 = n2_values(k);  % Current test refractive index

```

Figure 5.1: Code for optimal n_2 using double-layer – (1 of 3)

```
        %Interface 1-2
        r12 = (n_1 - n_2) / (n_1 + n_2);
        t12 = 2 * n_1 / (n_1 + n_2);
        Q12 = (1 / t12) * [1 r12; r12 1];

        %Interface 2-3
        r23 = (n_2 - n_3) / (n_2 + n_3);
        t23 = 2 * n_2 / (n_2 + n_3);
        Q23 = (1 / t23) * [1 r23; r23 1];

        % Build total transfer matrix for system
        % Order: Air→1→2→3→Substrate, including propagation phases
        T_total = Q01 * P * Q12 * P * Q23 * P * Q3S;

        %Calculate overall reflection coefficient (gamma)
        Gamma = T_total(2,1) / T_total(1,1);

        % Store reflectance (|gamma|²)
        Reflectance(k) = abs(Gamma)^2;
    end

    %Find index that gives minimum reflectivity
    [~, idx_min] = min(Reflectance);
    opt_n2 = n2_values(idx_min);

    %Display and plot results
    fprintf('n_1 = %.2f\nOptimal n_2 = %.2f (minimum reflectance)\n', n_1, opt_n2);

    figure;
    plot(n2_values, Reflectance * 100, 'LineWidth', 1.3);
    xlabel('Refractive Index n_2');
```

Figure 5.2: Code for optimal n_2 using double-layer – (2 of 3)

```
    ylabel('Reflectance (%)');
    title(sprintf('Optimal n_2 at λ_C = %d nm', lambdaC));
    grid on;
```

Figure 5.3: Code for optimal n_2 using double-layer – (3 of 3)

Virtual Simulations for *Figure 5.4 and 5.5*

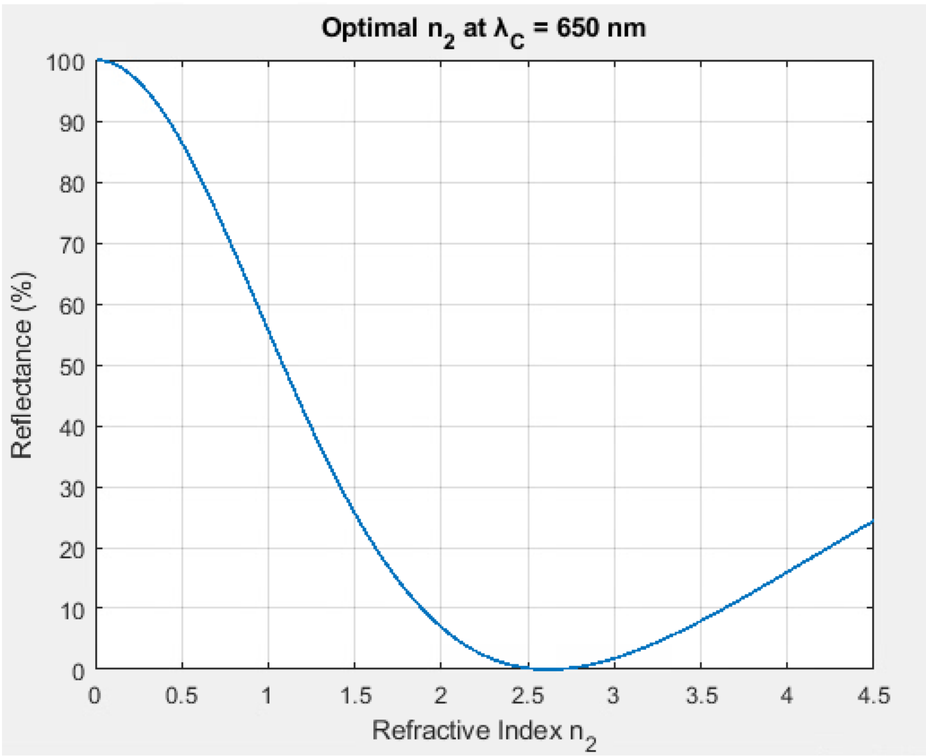

Figure 5.4: Plot of the Reflectivity (as percentage) vs. n_2 for optimal value of n_2 (double-layer).

```
n_1 = 1.40
Optimal n_2 = 2.62 (minimum reflectance)
```

Figure 5.5: Simulated optimal value of n_2 (double-layer)

Part 2 – (a) - Code, Flowchart and Virtual Simulations

Flow Chart for *Figure 5.6-5.8*

1. Define the imaginary constant j_const = 1j.
2. Define optical parameters and constants:
   - n_air (air or incident medium),
   - n_layer1 (first coating layer),

- n_layer2 (second coating layer, optimized),
- n_cell (solar cell substrate).

3. Define wavelength parameters:
   - λ_central = 650 nm (design wavelength),
   - λ_min = 400 nm (minimum wavelength),
   - λ_max = 1400 nm (maximum wavelength),
   - λ_range = λ_min : λ_max (vector of wavelengths),
   - num_lambda = number of wavelength samples.
4. Compute Fresnel reflection and transmission coefficients (independent of wavelength):
   - r_air_1, r_1_2, r_2_cell,
   - t_air_1, t_1_2, t_2_cell.
5. Initialize arrays to store results:
   - reflectivity (vs wavelength),
   - transmitted_power (vs wavelength).
6. Begin loop over each wavelength λ in λ_range:
   a. Set λ = current wavelength.
   b. Compute phase delay based on quarter-wave condition:
      $\Delta = (\pi/2) \times (\lambda\_central / \lambda)$.
   c. Build the propagation matrix P for the phase delay.
   d. Build interface matrices:
      Q_air_1, Q_1_2, Q_2_cell.
   e. Compute total transfer matrix T_total using the ordered multiplication sequence:
      T_total = ((Q_air_1 × P) × Q_1_2) × (P × Q_2_cell).
   f. Compute overall reflection and transmission coefficients:
      Γ = T_total(2,1) / T_total(1,1),
      τ = 1 / T_total(1,1).
   g. Compute reflectivity = $|\Gamma|^2$.
   h. Compute transmissivity = $|\tau|^2 \times$ (n_cell / n_air).
   i. Model solar irradiance using idealized spectral intensity formula:
      $I(\lambda) = (6.16\times10^{15}) / [\lambda^5 \times (\exp(2484 / \lambda) - 1)]$.
   j. Compute transmitted power into the cell:
      transmitted_power = transmissivity × irradiance.
   k. Store reflectivity and transmitted power for current wavelength.
7. End the loop after all wavelengths are processed.
8. Plot reflectivity (in %) versus wavelength.

Code for *Figure 5.6-5.8*

```
clear; clc; close all;

j_const = 1j;  % Imaginary unit

%Optical Parameters and Constants
n_air    = 1.0; % Air (incident medium)
n_layer1 = 1.4; % First coating layer
n_layer2 = 2.62; % Second coating layer (optimized)
n_cell   = 3.5; % Solar cell substrate

lambda_central = 650; % Design wavelength (nm)
lambda_min = 400;  % Lower wavelength limit (nm)
lambda_max = 1400; % Upper wavelength limit (nm)
lambda_range = lambda_min:lambda_max; % Wavelength vector
num_lambda = numel(lambda_range);

% Fresnel Reflection (Γ) and Transmission (τ) Coefficients
% Computed once (independent of wavelength)
r_air_1  = (n_air - n_layer1) / (n_air + n_layer1);
r_1_2    = (n_layer1 - n_layer2) / (n_layer1 + n_layer2);
r_2_cell = (n_layer2 - n_cell) / (n_layer2 + n_cell);

t_air_1  = 2 * n_air / (n_air + n_layer1);
t_1_2    = 2 * n_layer1 / (n_layer1 + n_layer2);
t_2_cell = 2 * n_layer2 / (n_layer2 + n_cell);

%Initialize Storage for Results
reflectivity = zeros(1, num_lambda); % Reflectivity vs wavelength
transmitted_power = zeros(1, num_lambda); % Transmitted power vs wavelength

% Compute Matrices and Results (Main Loop)
```

Figure 5.6: Code for the plot of reflectivity (%) vs. wavelength for 400-1400nm – (1 of 3)

```
% Compute Matrices and Results (Main Loop)

for idx = 1:num_lambda
    lambda = lambda_range(idx);

    %Phase Delay (quarter-wave condition)
    delta_val = (pi / 2) * (lambda_central / lambda);

    %Propagation Matrix
    P_matrix = [exp(j_const * delta_val) 0; 0 exp(-j_const * delta_val)];

    % Interface Matrices
    Q_air_1  = (1 / t_air_1)  * [1 r_air_1;  r_air_1 1];
    Q_1_2    = (1 / t_1_2)    * [1 r_1_2;    r_1_2   1];
    Q_2_cell = (1 / t_2_cell) * [1 r_2_cell; r_2_cell 1];

    % Compute Transfer Matrix (Reordered Multiplication Sequence)
    % Grouped differently to vary structure: ((Q_air_1 * P) * Q_1_2) * (P * Q_2_cell)
    T_total = ((Q_air_1 * P_matrix) * Q_1_2) * (P_matrix * Q_2_cell);

    %Reflection and Transmission Coefficients
    Gamma = T_total(2,1) / T_total(1,1);  % Overall reflection coefficient
    tau   = 1 / T_total(1,1); % Overall transmission coefficient

    %Reflectivity and Transmissivity
    reflectivity(idx) = abs(Gamma)^2;
    transmissivity = abs(tau)^2 / (n_air / n_cell);

    % Solar Irradiance Model
    % Idealized spectral intensity: I(λ) = 6.16x10^15 / [λ⁵(e^(2484/λ) - 1)]
    irradiance = (6.16e15) / ((lambda^5) * (exp(2484 / lambda) - 1));

    %Power Transmitted into the Cell
```

Figure 5.7: Code for the plot of reflectivity (%) vs. wavelength for 400-1400nm – (2 of 3)

```
    %Power Transmitted into the Cell
    transmitted_power(idx) = transmissivity * irradiance;
end

% Plot Results
figure;
plot(lambda_range, reflectivity * 100, 'b-', 'LineWidth', 1.3);
xlabel('Wavelength (nm)');
ylabel('Reflectivity (%)');
title('Reflectivity vs Wavelength - Optimized Double-Layer Design');
xlim([lambda_min, lambda_max]);
grid on;
```

Figure 5.8: Code for the plot of reflectivity (%) vs. wavelength for 400-1400nm – (3 of 3)

Virtual Simulations for *Figure 5.9*

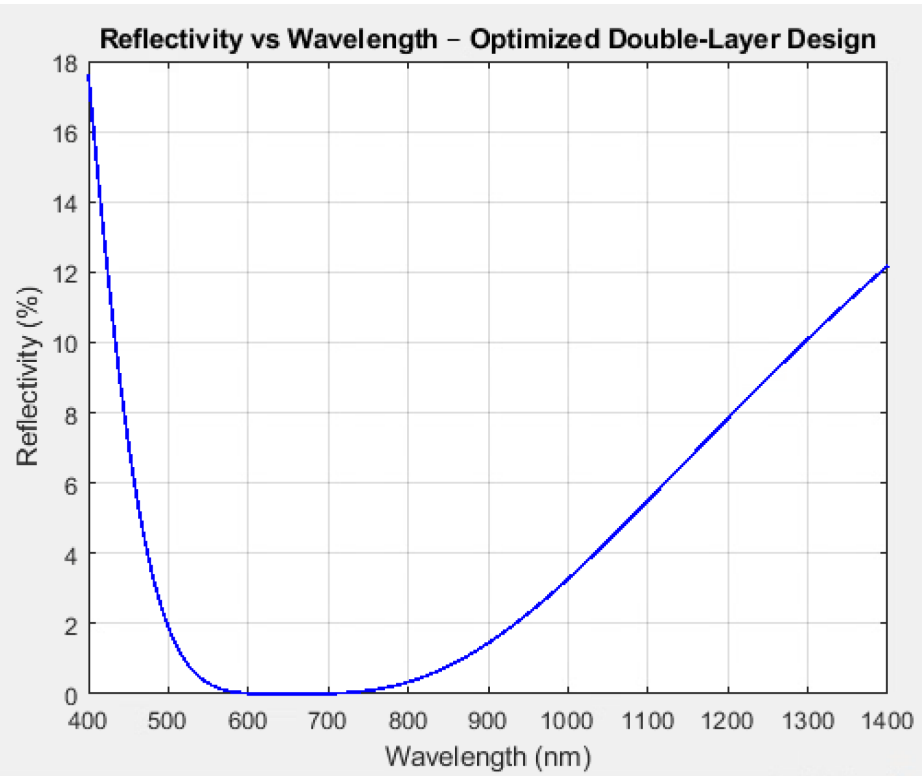

Figure 5.9: Plot of the reflectivity (%) vs. wavelength for the range 400-1400nm

Part 2 – (b) – Code, Flowchart and Virtual Simulations

Flow Chart for *Figure 5.10-5.12*

1. Define the imaginary constant j_const = 1j (for complex exponentials).
2. Define optical refractive indices (in reverse order from bottom to top):
   - n_cell = 3.5 (semiconductor substrate),
   - n_layer2 = 2.6192 (second coating layer, closest to cell),
   - n_layer1 = 1.4 (first coating layer, outer layer),
   - n_air = 1.0 (air, incident medium).
3. Define wavelength parameters:
   - λ_central = 650 nm (design wavelength),
   - λ_full = 200 : 1 : 2200 nm (entire wavelength range),

- λ_visible = 400 : 1 : 1400 nm (subrange of interest),
- num_points = number of wavelength samples.

4. Compute Fresnel reflection (Γ) and transmission (τ) coefficients:
   - Γ_{12}, Γ_{01}, Γ_{23} for the three interfaces,
   - τ_{12}, τ_{01}, τ_{23} corresponding to transmission coefficients.
5. Construct interface matrices (Q):
   - Q_{01}, Q_{12}, Q_{23}.
6. Initialize the result array transmitted_power with zeros (size = num_points).
7. Begin main loop over each wavelength λ in λ_full:
   a. Assign current λ = λ_full(k).
   b. Compute phase delay for both layers (quarter-wave scaling):
      $\Delta = (\pi/2) \times (\lambda\_central / \lambda)$.
   c. Build the propagation matrix for both layers:
      P_matrix = [exp(jΔ) 0; 0 exp(−jΔ)].
   d. Compute total transfer matrix with new grouping:
      T_total = (($Q_{01} \times Q_{12}$) × P_matrix) × (Q_{23} × P_matrix).
   e. Compute total reflection coefficient:
      Γ_total = T_total(2,1) / T_total(1,1).
   f. Compute reflectivity = $|\Gamma\_total|^2$.
   g. Compute irradiance using the spectral irradiance model:
      $I(\lambda) = (6.16\times10^{15}) / [\lambda^5 \times (\exp(2484/\lambda) - 1)]$.
   h. Compute transmitted power at this wavelength:
      P_transmitted = (1 − Reflectivity) × I(λ).
   i. Store transmitted_power(k) = P_transmitted.
8. End the loop after all wavelengths have been processed.
9. Integrate total transmitted power over two wavelength ranges:
   - Identify visible range indices ($400 \le \lambda \le 1400$).
   - Compute total_power_visible = sum(transmitted_power over visible range).
   - Compute total_power_full = sum(transmitted_power over full range).
10. Create a figure window for visualization.
11. Create Subplot 1 (400–1400 nm range):
    - Plot transmitted power vs wavelength.
12. Create Subplot 2 (200–2200 nm range):
    - Plot transmitted power vs wavelength.

Code for *Figure 5.10-5.12*

```
clear; clc; close all;

j_const = 1j;   % Imaginary unit for complex exponentials

%Optical Refractive Indices (Reordered Definition)
n_cell   = 3.5;       % Semiconductor substrate (bottom)
n_layer2 = 2.6192;    % Second coating layer (closest to cell)
n_layer1 = 1.4;       % First coating layer (outer layer)
n_air    = 1.0;       % Air (incident medium)

%Wavelength Parameters
lambda_central = 650; % Design (central) wavelength (nm)
lambda_full = 200:1:2200; % Full wavelength range (200-2200 nm)
lambda_visible = 400:1:1400; % Subrange of interest (400-1400 nm)
num_points = numel(lambda_full);

%Fresnel Coefficients: Reflection (gamma) & Transmission (tao)
Gamma_12 = (n_layer1 - n_layer2) / (n_layer1 + n_layer2);
Gamma_01 = (n_air - n_layer1) / (n_air + n_layer1);
Gamma_23 = (n_layer2 - n_cell) / (n_layer2 + n_cell);

tau_12 = 2 * n_layer1 / (n_layer1 + n_layer2);
tau_01 = 2 * n_air / (n_air + n_layer1);
tau_23 = 2 * n_layer2 / (n_layer2 + n_cell);

%Interface Matrices (Q)
Q_01 = (1 / tau_01) * [1 Gamma_01; Gamma_01 1];
Q_12 = (1 / tau_12) * [1 Gamma_12; Gamma_12 1];
Q_23 = (1 / tau_23) * [1 Gamma_23; Gamma_23 1];

```

Figure 5.10: Plots of the Power vs. Wavelength for 400-1400nm and 200-2200nm - (1 of 3)

```
%Initialize Storage for Results
transmitted_power = zeros(1, num_points); % Power spectrum (W/m^2)

%Main Loop: Compute Transmitted Power vs Wavelength

for k = 1:num_points
    lambda = lambda_full(k);

    %Phase delay for each layer (quarter-wave scaling)
    delta_phase = (pi / 2) * (lambda_central / lambda);

    %Propagation matrix for both layers
    P_matrix = [exp(j_const * delta_phase) 0; 0 exp(-j_const * delta_phase)];

    %Compute total transfer matrix (new sequence grouping)
    % Structure differs: ((Q_01 * Q_12) * P) * (Q_23 * P)
    T_total = ((Q_01 * Q_12) * P_matrix) * (Q_23 * P_matrix);

    %Reflection coefficient gamma_total
    Gamma_total = T_total(2,1) / T_total(1,1);
    Reflectivity = abs(Gamma_total)^2;

    %Irradiance computation
    irradiance = (6.16e15) / ((lambda^5) * (exp(2484 / lambda) - 1));

    %Transmitted total power at this wavelength
    transmitted_power(k) = (1 - Reflectivity) * irradiance;
end

% Power Integration
% Integrate total transmitted power in both given ranges
visible_idx = find(lambda_full >= 400 & lambda_full <= 1400);
```

Figure 5.11: Plots of the Power vs. Wavelength for 400-1400nm and 200-2200nm - (2 of 3)

```
total_power_visible = sum(transmitted_power(visible_idx));
total_power_full = sum(transmitted_power);

% Visualization using Subplots
figure('Name', 'Transmitted Power Spectra', 'NumberTitle', 'off');

% Subplot 1: 400-1400 nm range
subplot(2,1,1);
plot(lambda_full, transmitted_power, 'LineWidth', 1.3, 'Color', [0.1 0.45 0.8]);
xlim([400, 1400]);
xlabel('Wavelength (nm)');
ylabel('Power (W/m²)');
title('Transmitted Power Spectrum - 400 to 1400 nm');
grid on;

% Subplot 2: Full 200-2200 nm range
subplot(2,1,2);
plot(lambda_full, transmitted_power, 'LineWidth', 1.3, 'Color', [0.2 0.3 0.8]);
xlim([200, 2200]);
xlabel('Wavelength (nm)');
ylabel('Power (W/m²)');
title('Transmitted Power Spectrum - 200 to 2200 nm');
grid on;
```

Figure 5.12: Plots of the Power vs. Wavelength for 400-1400nm and 200-2200nm - (3 of 3)

Virtual Simulations for *Figure 5.13-5.15*

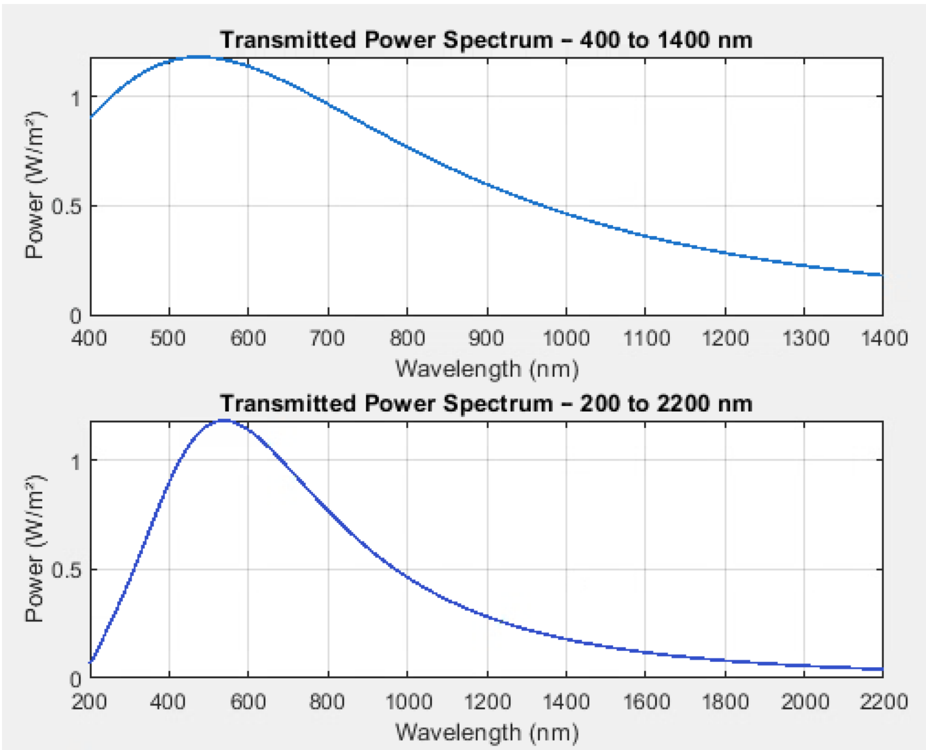

Figure 5.13: Plots of the Transmitted Power vs. Wavelength for 400-1400nm and 200-2200nm (double-layer).

```
Total Power in Watts = 751.202068
```

Figure 5.14: Result of the Transmitted Power vs. Wavelength for 400-1400nm and (double-layer).

```
Total Power in Watts = 929.591599
```

Figure 5.15: Result of the Transmitted Power vs. Wavelength for and 200-2200nm (double-layer).

Part 2 – (c) – Code, Flowchart and Virtual Simulations

Flow Chart for *Figure 5.16-5.20*

1. Define constants:
    - j = 1j (imaginary unit),
    - n_0 = 1.0 (air, incident medium),
    - n_3 = 3.5 (semiconductor substrate),
    - λ_C = 650 nm (center wavelength),
    - λ_range = 400 : 1 : 1400 nm (wavelength range),
    - num_lambda = number of wavelength samples.
2. Define irradiance and exponential constants:
    - IRRAD_Const = 6.16×10^{15},
    - Exp_Const = 2484.
3. Define search ranges for optimization:
    - n_1_min to n_1_max = [1.0, 3.0],
    - n_2_min to n_2_max = [1.0, 3.0],
    - Step_Size = 0.4,
    - Max_Iteration = 5.
4. Preallocate storage arrays:
    - Total_Power,
    - n_1_record,
    - n_2_record.
5. Begin **optimization loop** (Iter = 1 to Max_Iteration):
    a. Set index counter idx = 1.
    b. For each n_1 in [n_1_min : Step_Size : n_1_max]:
        i. For each n_2 in [n_2_min : Step_Size : n_2_max]:
            - Initialize R_vals and P_vals arrays (for each wavelength).
            - Compute **Fresnel reflection and transmission coefficients**:
                r_{01}, r_{12}, r_{23} and t_{01}, t_{12}, t_{23}.
            - Build **interface matrices** Q_{01}, Q_{12}, Q_{23}.
            - For each wavelength in λ_range:
                1. Compute phase shift $\Delta = (\pi/2) \times (\lambda\_C / \lambda)$.
                2. Construct phase matrix P = [exp(jΔ) 0; 0 exp(−jΔ)].
                3. Compute total transfer matrix $T = Q_{01} \times P \times Q_{12} \times P \times Q_{23}$.
                4. Calculate reflection and transmission coefficients:
                    $\Gamma = T(2,1) / T(1,1)$, $\tau = 1 / T(1,1)$.
                5. Compute reflectivity $R = |\Gamma|^2$ and transmittance Trans = $|\tau|^2 \times (n_0 / n_3)$.
                6. Compute irradiance I = IRRAD_Const / [λ^5 × (exp(Exp_Const / λ) − 1)].
                7. Compute transmitted power P = Trans × I.
                8. Store R_vals(i) and P_vals(i).
            - After wavelength loop ends, record total results:
                n_1_record(idx) = n_1,
                n_2_record(idx) = n_2,
                Total_Power(idx) = sum(P_vals).
            - Increment idx by 1.

ii. End of n_2 loop.
iii. End of n_1 loop.
c. Find **best combination** for current iteration:
- Identify maximum Total_Power and its position.
- Set Best_n_1 and Best_n_2 to the refractive indices at that position.
d. Refine search range for the next iteration (if Iter < Max_Iteration):
- Narrow n_1 and n_2 ranges around their best values,
- Reduce Step_Size by half (but not below 0.01).

6. End optimization loop.
7. Initialize arrays for final results:
   - R_final, P_final.
8. Using the optimized refractive indices (Best_n_1, Best_n_2), perform a **final sweep** across λ_range:
   a. For each wavelength:
   i. Compute Fresnel coefficients using Best_n_1 and Best_n_2.
   ii. Compute interface matrices Q_{01}, Q_{12}, Q_{23} and phase matrix P.
   iii. Calculate total transfer matrix T_total = $Q_{01} \times P \times Q_{12} \times P \times Q_{23}$.
   iv. Compute reflection coefficient Γ = T_total(2,1)/T_total(1,1).
   v. Compute reflectivity = $|\Gamma|^2$.
   vi. Compute transmittance and irradiance.
   vii. Compute final transmitted power P_final = Trans × I.
   viii. Store R_final and P_final for plotting.
9. End final wavelength sweep.
10. Subplot 1: Plot optimized reflectivity vs wavelength (400–1400 nm).
11. Subplot 2: Plot transmitted power vs wavelength (400–1400 nm).

Code for *Figure 5.16-5.20*

```
clear; clc; close all;

%Constant parameters
j = 1j; % Imaginary unit
n_0 = 1.0; % Air (incident medium)
n_3 = 3.5;  % Semiconductor substrate
lambda_C = 650; % Center wavelength [nm]
lambda_range = 400:1400; % Wavelength range [nm]
num_lambda = length(lambda_range);

IRRAD_Const = 6.16e15;% Irradiance constant
Exp_Const   = 2484;% Exponential term constant

%Search range
n1_min = 1.0;  n1_max = 3.0;
n2_min = 1.0;  n2_max = 3.0;
Step_Size = 0.4;
Max_Iteration = 5;

% Preallocation
Total_Power = [];
n1_record = [];
n2_record = [];

%OPTIMIZATION LOOP
for Iter = 1:Max_Iteration
    idx = 1;
```

Figure 5.16: Plots of the Power vs. Wavelength for 400-1400nm for optimized power - (1 of 5)

```
    for n_1 = n1_min:Step_Size:n1_max
        for n_2 = n2_min:Step_Size:n2_max

            R_vals = zeros(1, num_lambda);
            P_vals = zeros(1, num_lambda);

            % Fresnel coefficients for each interface
            r01 = (n_0 - n_1) / (n_0 + n_1);
            r12 = (n_1 - n_2) / (n_1 + n_2);
            r2S = (n_2 - n_3) / (n_2 + n_3);

            t01 = 2 * n_0 / (n_0 + n_1);
            t12 = 2 * n_1 / (n_1 + n_2);
            t2S = 2 * n_2 / (n_2 + n_3);

            % Interface matrices
            Q01 = (1/t01) * [1 r01; r01 1];
            Q12 = (1/t12) * [1 r12; r12 1];
            Q2S = (1/t2S) * [1 r2S; r2S 1];

            % Sweep through wavelength range
            for i = 1:num_lambda
                lambda = lambda_range(i);
                delta = (pi/2) * (lambda_C / lambda);
                P = [exp(j*delta) 0; 0 exp(-j*delta)];

                % Transfer matrix across layers
                T = Q01 * P * Q12 * P * Q2S;

                % Reflection (Γ) and transmission (τ)
                Gamma = T(2,1)/T(1,1);
                Tau   = 1/T(1,1);

```

Figure 5.17: Plots of the Power vs. Wavelength for 400-1400nm for optimized power - (2 of 5)

```
                % Reflectivity and transmitted power
                R_vals(i) = abs(Gamma)^2;
                Trans = abs(Tau)^2 / (n_0/n_3);
                Irrad = IRRAD_Const / ((lambda^5) * (exp(Exp_Const/lambda) - 1));
                P_vals(i) = Trans * Irrad;
            end

            % Store integrated results
            n1_record(idx) = n_1;
            n2_record(idx) = n_2;
            Total_Power(idx) = sum(P_vals);
            idx = idx + 1;
        end
    end

    % Best combination for this iteration
    [Best_Power, pos] = max(Total_Power);
    Best_n1 = n1_record(pos);
    Best_n2 = n2_record(pos);

    % Refinement
    if Iter < Max_Iteration
        n1_min = max(Best_n1 - Step_Size*2, 1.0);
        n1_max = min(Best_n1 + Step_Size*2, 3.0);
        n2_min = max(Best_n2 - Step_Size*2, 1.0);
        n2_max = min(Best_n2 + Step_Size*2, 3.0);
        Step_Size = max(Step_Size / 2, 0.01);
    end
end

%Final Results
R_final = zeros(1, num_lambda);
P_final = zeros(1, num_lambda);
```

Figure 5.18: Plots of the Power vs. Wavelength for 400-1400nm for optimized power - (3 of 5)

```
for k = 1:num_lambda
    lambda = lambda_range(k);
    delta = (pi/2) * (lambda_C / lambda);

    r01 = (n_0 - Best_n1)/(n_0 + Best_n1);
    r12 = (Best_n1 - Best_n2)/(Best_n1 + Best_n2);
    r2S = (Best_n2 - n_3)/(Best_n2 + n_3);

    t01 = 2 * n_0 / (n_0 + Best_n1);
    t12 = 2 * Best_n1 / (Best_n1 + Best_n2);
    t2S = 2 * Best_n2 / (Best_n2 + n_3);

    Q01 = (1/t01) * [1 r01; r01 1];
    Q12 = (1/t12) * [1 r12; r12 1];
    Q2S = (1/t2S) * [1 r2S; r2S 1];
    P = [exp(j*delta) 0; 0 exp(-j*delta)];

    T_total = Q01 * P * Q12 * P * Q2S;
    Gamma = T_total(2,1)/T_total(1,1);
    Tau = 1/T_total(1,1);

    R_final(k) = abs(Gamma)^2;
    Trans = abs(Tau)^2 / (n_0/n_3);
    Irrad = IRRAD_Const / ((lambda^5) * (exp(Exp_Const/lambda) - 1));
    P_final(k) = Trans * Irrad;
end

%Plotting
figure('Color','w');

subplot(2,1,1);
plot(lambda_range, R_final*100, 'r', 'LineWidth', 1.2);
title('Optimized Reflectivity vs Wavelength (400-1400 nm)');
```

Figure 5.19: Plots of the Power vs. Wavelength for 400-1400nm for optimized power - (4 of 5)

```
xlabel('Wavelength (nm)');
ylabel('Reflectivity (%)');
grid on; xlim([400 1400]);

subplot(2,1,2);
plot(lambda_range, P_final, 'b', 'LineWidth', 1.2);
title('Transmitted Power vs Wavelength (400-1400 nm)');
xlabel('Wavelength (nm)');
ylabel('Power (W/m^2)');
grid on; xlim([400 1400]);

sgtitle('Optimized Double-Layer Coating Performance (400-1400 nm)');
```

Figure 5.20: Plots of the Power vs. Wavelength for 400-1400nm for optimized power - (5 of 5)

Virtual Simulations for *Figure 5.21 and 5.22*

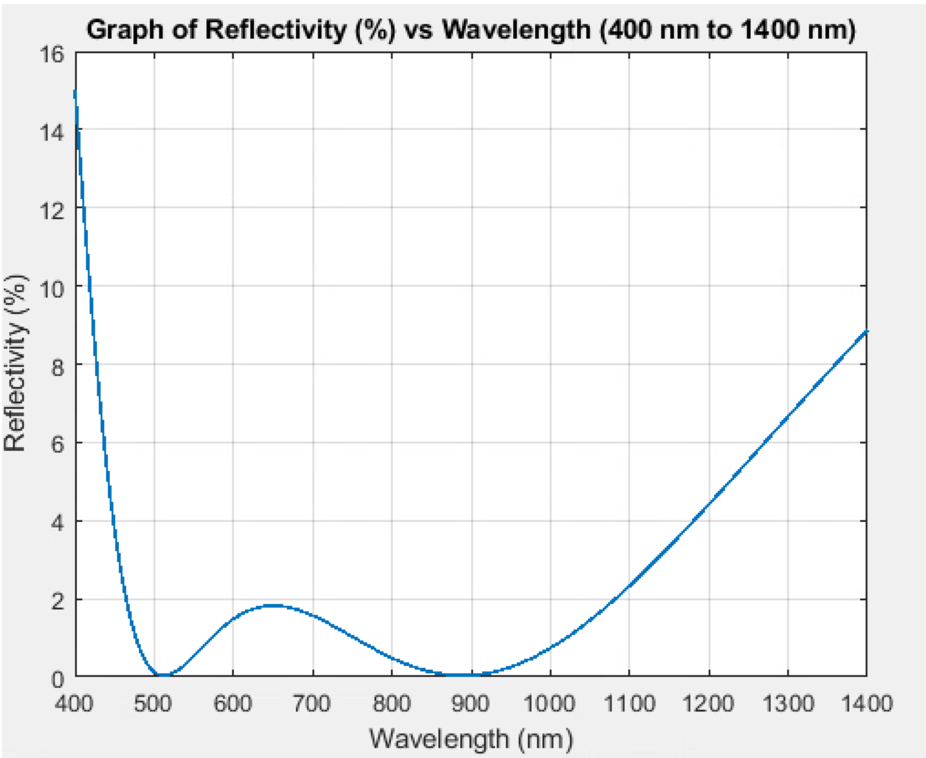

Figure 5.21: Plot of the Transmitted Power vs. Wavelength for 400-1400nm for optimized power (double-layer).

```
Part2_c (400-1400nm)
n_1 = 1.45
n_2 = 2.40
Total Power in Watts = 758.8876
```

Figure 5.22: Result of the Transmitted Power vs. Wavelength for 400-1400nm for optimized power (double-layer).

Flow Chart for *Figure 5.23-5.27*

1. Define constants and base parameters:
   - j = 1i (imaginary unit),
   - λ_c = 650 nm (center wavelength),
   - n_air = 1.0 (air refractive index),
   - n_sub = 3.5 (semiconductor substrate),
   - λ_span = 200 : 1 : 2200 nm (spectrum range),
   - L_count = number of wavelength samples.
2. Define irradiance constants:
   - C_irrad = 6.16×10^{15},
   - C_exp = 2484.
3. Define the **search grid** parameters:
   - n_1 range: $1.0 \rightarrow 3.0$,
   - n_2 range: $1.0 \rightarrow 3.0$,
   - step_init = 0.4,
   - refine_max = 5 (maximum refinement cycles).
4. Initialize data matrices for recording:
   - all_power, map_n_1, map_n_2 (empty arrays).
5. Begin **main optimization loop** (ref = 1 → refine_max):
   a. Set counter = 1.
   b. For each n_1 in search.n_1 range:
      For each n_2 in search.n_2 range:
         i. Initialize Rset and Pset arrays (size = L_count).
         ii. For each wavelength L in λ_span:
            1. Assign current wavelength L and compute phase shift:
               $\Delta\phi = (\pi/2) \times (\lambda\_c / L)$.
            2. Compute **Fresnel reflection and transmission coefficients**:
               r_{01}, r_{12}, r_{25} and t_{01}, t_{12}, t_{25}.
            3. Construct **interface matrices**:
               Q_{01}, Q_{12}, Q_{25}.
            4. Construct **phase matrix**:
               $P_m = [\exp(j\Delta\phi)\ 0;\ 0\ \exp(-j\Delta\phi)]$.
            5. Compute **total transfer matrix** through all layers:
               $T\_net = Q_{01} \times P_m \times Q_{12} \times P_m \times Q_{25}$.
            6. Compute **reflection and transmission coefficients**:
               $\Gamma = T\_net(2,1)/T\_net(1,1), \quad \tau = 1/T\_net(1,1)$.
            7. Compute **reflectance and transmitted power**:
               Rset(idx) = $|\Gamma|^2$,
               T_ratio = $|\tau|^2 \times$ (n_air / n_sub).
            8. Compute spectral irradiance at this wavelength:
               $I\_\lambda = C\_irrad / [L^5 \times (\exp(C\_exp / L) - 1)]$.
            9. Compute power transmitted:
               Pset(idx) = T_ratio × I_λ.
         iii. End wavelength loop.
         iv. Compute integrated total power for this n_1/n_2 pair:

map_n_1(counter) = n_1,
map_n_2(counter) = n_2,
all_power(counter) = sum(Pset).
v. Increment counter.
End of nested n_2 loop.
c. Find **best combination for this iteration**:
- Locate maximum(all_power).
- Extract corresponding best_n_1, best_n_2, and best_power.
d. Print iteration summary (optional).
e. If current refinement < refine_max:
- Narrow the n_1 and n_2 ranges around best values,
- Reduce step_init by half for finer resolution,
- Ensure bounds remain between 1 and 3.

6. End optimization loop.
7. Display final optimization results:
   - Print best_n_1, best_n_2, and best total transmitted power.
8. Initialize R_final array for storing reflectivity.
9. For each wavelength in λ_span (final spectrum computation):
   a. Compute updated Fresnel coefficients using best_n_1 and best_n_2.
   b. Build corresponding interface matrices and phase matrix.
   c. Compute total transfer matrix T_all = $Q_{01} \times P \times Q_{12} \times P \times Q_{25}$.
   d. Compute reflection coefficient G = T_all(2,1)/T_all(1,1).
   e. Compute reflectivity = $|G|^2$ and store in R_final(k).
10. Plot final reflectivity spectrum:.

Code for *Figure 5.23-5.27*

```
clc; clear; close all

%Base Parameters
j = 1i; % Imaginary unit
lambda_c = 650; % Center design wavelength [nm]
n_air  = 1.0; % Air refractive index
n_sub  = 3.5; % Substrate (semiconductor cell)
lambda_span = 200:1:2200; % Spectrum range
L_count = length(lambda_span);

% Irradiance constants
C_irrad = 6.16e15;
C_exp   = 2484;

%Searching grid
search.n1 = linspace(1.0, 3.0, 6);
search.n2 = linspace(1.0, 3.0, 6);
step_init = 0.4;
refine_max = 5;

% Data matrix
all_power = [];
map_n1 = [];
map_n2 = [];

%Main Loop
for ref = 1:refine_max
```

Figure 5.23: Plots of the Power vs. Wavelength for 200-2200nm for optimized power - (1 of 5)

```
    counter = 1;

    for n1 = search.n1
        for n2 = search.n2

            Rset = zeros(1, L_count);
            Pset = zeros(1, L_count);

            for idx = 1:L_count
                L = lambda_span(idx);
                dphi = (pi/2)*(lambda_c/L);

                % Fresnel terms (recomputed inside to allow per-layer tuning)
                r01 = (n_air - n1)/(n_air + n1);
                r12 = (n1 - n2)/(n1 + n2);
                r2S = (n2 - n_sub)/(n2 + n_sub);

                t01 = 2*n_air/(n_air + n1);
                t12 = 2*n1/(n1 + n2);
                t2S = 2*n2/(n2 + n_sub);

                % Interface matrices
                Q01 = (1/t01)*[1 r01; r01 1];
                Q12 = (1/t12)*[1 r12; r12 1];
                Q2S = (1/t2S)*[1 r2S; r2S 1];

                % Phase propagation
                Pm = [exp(j*dphi) 0; 0 exp(-j*dphi)];

                % Total transfer through all layers
                T_net = Q01 * Pm * Q12 * Pm * Q2S;
```

Figure 5.24: Plots of the Power vs. Wavelength for 200-2200nm for optimized power - (2 of 5)

```
                T_net = Q01 * Pm * Q12 * Pm * Q2S;

                % Reflection / Transmission coefficients
                Gamma = T_net(2,1)/T_net(1,1);
                Tau   = 1/T_net(1,1);

                % Reflectance and transmitted power
                Rset(idx) = abs(Gamma)^2;
                T_ratio   = abs(Tau)^2 / (n_air/n_sub);
                I_lambda  = C_irrad / ((L^5)*(exp(C_exp/L)-1));
                Pset(idx) = T_ratio * I_lambda;
            end

            % Store the integrated total power for this n1/n2 pair
            map_n1(counter) = n1;
            map_n2(counter) = n2;
            all_power(counter) = sum(Pset);
            counter = counter + 1;
        end
    end

    %Best combination for this iteration
    [best_power, pos] = max(all_power);
    best.n1 = map_n1(pos);
    best.n2 = map_n2(pos);

    % Adjust refinement window
    if ref < refine_max
        search.n1 = best.n1-0.8*step_init : step_init/2 : best.n1+0.8*step_init;
        search.n2 = best.n2-0.8*step_init : step_init/2 : best.n2+0.8*step_init;
        search.n1 = search.n1(search.n1>=1 & search.n1<=3);
        search.n2 = search.n2(search.n2>=1 & search.n2<=3);
        step_init = step_init/2;
```

Figure 5.25: Plots of the Power vs. Wavelength for 200-2200nm for optimized power - (3 of 5)

```
end

%Results and plotting
fprintf('\n[OPTIMIZATION RESULTS]\n');
fprintf('Best n1 = %.3f\n', best.n1);
fprintf('Best n2 = %.3f\n', best.n2);
fprintf('Maximum Total Power = %.4f W\n\n', best_power);

% Final reflectivity spectrum for best combination
R_final = zeros(1, L_count);
for k = 1:L_count
    lam = lambda_span(k);
    phi = (pi/2)*(lambda_c/lam);

    r01 = (n_air - best.n1)/(n_air + best.n1);
    r12 = (best.n1 - best.n2)/(best.n1 + best.n2);
    r2S = (best.n2 - n_sub)/(best.n2 + n_sub);

    t01 = 2*n_air/(n_air + best.n1);
    t12 = 2*best.n1/(best.n1 + best.n2);
    t2S = 2*best.n2/(best.n2 + n_sub);

    Q01 = (1/t01)*[1 r01; r01 1];
    Q12 = (1/t12)*[1 r12; r12 1];
    Q2S = (1/t2S)*[1 r2S; r2S 1];

    Pm = [exp(j*phi) 0; 0 exp(-j*phi)];
    T_all = Q01*Pm*Q12*Pm*Q2S;
    G = T_all(2,1)/T_all(1,1);
    R_final(k) = abs(G)^2;
end

% Plot final reflectivity
```

Figure 5.26: Plots of the Power vs. Wavelength for 200-2200nm for optimized power - (4 of 5)

```
xlabel('Wavelength (nm)');
ylabel('Reflectivity (%)');
xlim([200 2200]); grid on; box on;
```

Figure 5.27: Plots of the Power vs. Wavelength for 200-2200nm for optimized power - (5 of 5)

Virtual Simulations for *Figure 5.28-5.29*

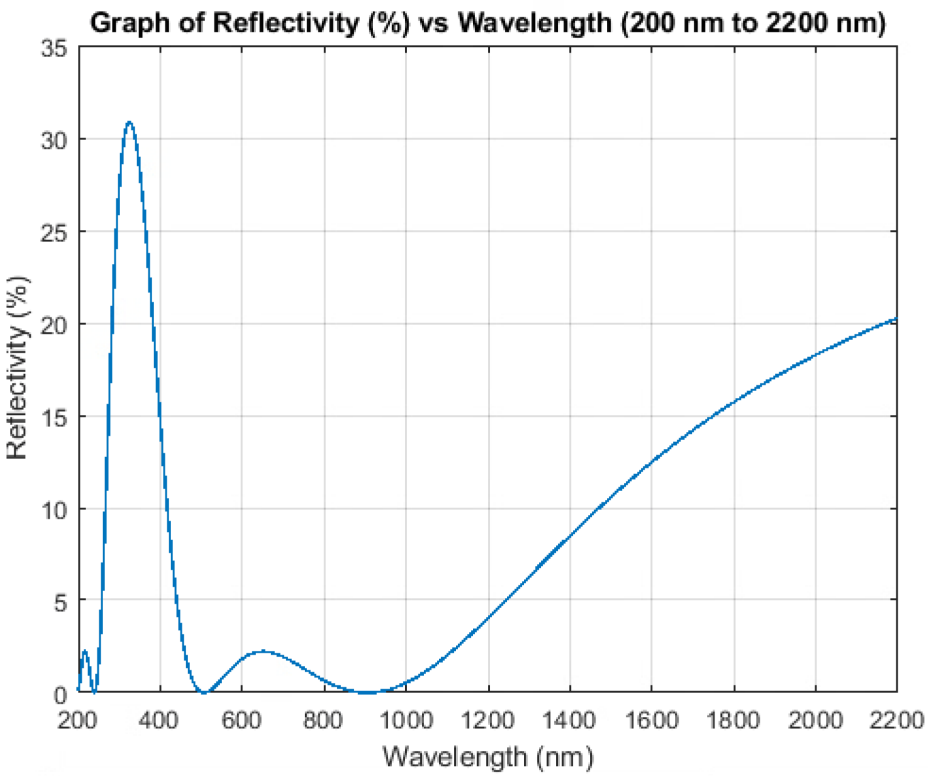

Figure 5.28: Plot of the Transmitted Power vs. Wavelength for 200-2200nm for optimized power (double-layer).

```
Part2_c (200-2200nm)
n1 = 1.475
n2 = 2.375
Maximum Total Power = 941.4174 W
```

Figure 5.29: Result of the Transmitted Power vs. Wavelength for 200-2200nm for optimized power (double-layer).

Part 3 – (c) – Code, Flowchart and Virtual Simulations

Commented [RG3]: Add code

Flow Chart for *Figure 5.30 and 5.31*

1. Define **material and wavelength parameters**:
   - n_inc = 1.0 (incident medium, air),
   - n_top = 1.4 (top fixed coating),
   - n_bot = 3.15 (bottom fixed coating adjacent to the cell),
   - n_cell = 3.5 (substrate),
   - λ_C = 650 nm (design wavelength).
2. Compute **fixed interface Fresnel terms** for the air–top and bottom–cell interfaces:
   - r_inc_top, t_inc_top, and matrix Q_inc_top,
   - r_bot_cell, t_bot_cell, and matrix Q_bot_cell.
3. Define **quarter-wave propagation matrix** for λ_C:
   - $\Delta = \pi/2$,
   - Pq = [exp(jΔ)   0;   0   exp(−jΔ)].
4. Define **n_2 sweep range** for the middle coating layer:
   - n_2_grid = 0.00 : 0.01 : 4.50.
5. Create an **anonymous helper function** reflectance_of that takes n_2 as input and returns its reflectivity using the local function.
6. Use arrayfun to evaluate reflectance over all n_2 values:
   - R_vec = arrayfun(reflectance_of, n_2_grid).
7. Determine **optimum n_2** (minimum reflectance):
   - [~, idx_min] = min(R_vec),
   - n_2_opt = n_2_grid(idx_min).
8. Create a **plot of reflectivity vs n_2**:
   a. Plot 100×R_vec versus n_2_grid.
   b. Label x-axis as "n_2 (middle coating)" and y-axis as "Reflectivity (%)".
   c. Add title: "Reflectivity (%) vs n_2 at λ_C = 650 nm".
   d. Enable grid and set limits based on n_2 range.
9. Display results in the command window:
   - Print n_top, n_bot, and the optimal n_2 with minimum reflectivity.
10. Define **local function R = function_R(...)** to calculate reflectivity for a given n_2:
    a. Compute **Fresnel terms** for:
       - top–middle interface (r_top_mid, t_top_mid),
       - middle–bottom interface (r_mid_bot, t_mid_bot).
    b. Build **interface matrices** Q_top_mid and Q_mid_bot.
    c. Compute **total transfer matrix** across all layers:
       T = Q_inc_top × Pq × Q_top_mid × Pq × Q_mid_bot × Pq × Q_bot_cell.
    d. Calculate **overall reflection coefficient**:
       $\Gamma = T(2,1) / T(1,1)$.
    e. Compute **reflectance** $R = |\Gamma|^2$.
    f. Return R.

Code for *Figure 5.30 and 5.31*

```
%Materials and design wavelength
n_inc    = 1.0; % incident medium (air)
n_top    = 1.4; % first coating (fixed)
n_bot    = 3.15; % second coating (fixed, adjacent to cell)
n_cell   = 3.5; % substrate (cell)
lambda_C = 650;% design wavelength [nm]

%Fixed interface Fresnel terms (air↔top, bot↔cell)
r_inc_top = (n_inc - n_top) / (n_inc + n_top);
t_inc_top = 2 * n_inc / (n_inc + n_top);
Q_inc_top = (1 / t_inc_top) * [1, r_inc_top; r_inc_top, 1];

r_bot_cell = (n_bot - n_cell) / (n_bot + n_cell);
t_bot_cell = 2 * n_bot / (n_bot + n_cell);
Q_bot_cell = (1 / t_bot_cell) * [1, r_bot_cell; r_bot_cell, 1];

%Quarter-wave propagation at lambda_C (same for each layer in this model)
Delta = pi/2;
Pq = [exp(1i*Delta), 0; 0, exp(-1i*Delta)];

% n2 sweep (middle coating index)
n2_grid = 0.00 : 0.01 : 4.50;

% Anonymous helper: reflectance for a given n2
reflectance_of = @(n2) ...
    ( ...
        function_R(n2, n_top, n_bot, n_inc, n_cell, Q_inc_top, Q_bot_cell, Pq) ...
    );

% Evaluate reflectance over the grid (vectorized via arrayfun)
R_vec = arrayfun(reflectance_of, n2_grid);

% Find optimum n2 (minimum reflectance)
```

Figure 5.30: Code of the Transmitted Power vs. Wavelengths for the optimized power – (1 of 2)

```
[~, idx_min] = min(R_vec);
n2_opt = n2_grid(idx_min);

% Plot
figure;
plot(n2_grid, 100*R_vec, 'LineWidth', 1.3);
grid on; xlim([min(n2_grid) max(n2_grid)]);
title('Reflectivity (%) vs n_2 at \lambda_C = 650 nm');
xlabel('n_2 (middle coating)');
ylabel('Reflectivity (%)');

% Report
fprintf('\nPart3_c\n');
fprintf('n_1 = %.2f\n', n_top);
fprintf('n_2 = %.2f\n', n_bot);
fprintf('Minimum reflectivity at n_2 = %.2f\n', n2_opt);

%Local function (kept in the same file)
function R = function_R(n2, n_top, n_bot, n_inc, n_cell, Q_inc_top, Q_bot_cell, Pq)
    % Fresnel terms for top↔n2 and n2↔bot
    r_top_mid = (n_top - n2) / (n_top + n2);
    t_top_mid = 2 * n_top / (n_top + n2);
    Q_top_mid = (1 / t_top_mid) * [1, r_top_mid; r_top_mid, 1];

    r_mid_bot = (n2 - n_bot) / (n2 + n_bot);
    t_mid_bot = 2 * n2 / (n2 + n_bot);
    Q_mid_bot = (1 / t_mid_bot) * [1, r_mid_bot; r_mid_bot, 1];

    % Total transfer matrix: inc → top → n2 → bot → cell
    T = Q_inc_top * Pq * Q_top_mid * Pq * Q_mid_bot * Pq * Q_bot_cell;

    % Overall reflection coefficient Γ = T21 / T11; R = |Γ|^2
    Gamma = T(2,1) / T(1,1);
    R = abs(Gamma)^2;
```

Figure 5.31: Code of the Transmitted Power vs. Wavelengths for the optimized power – (2 of 2)

Virtual Simulations for *Figure 5.32 and 5.33*

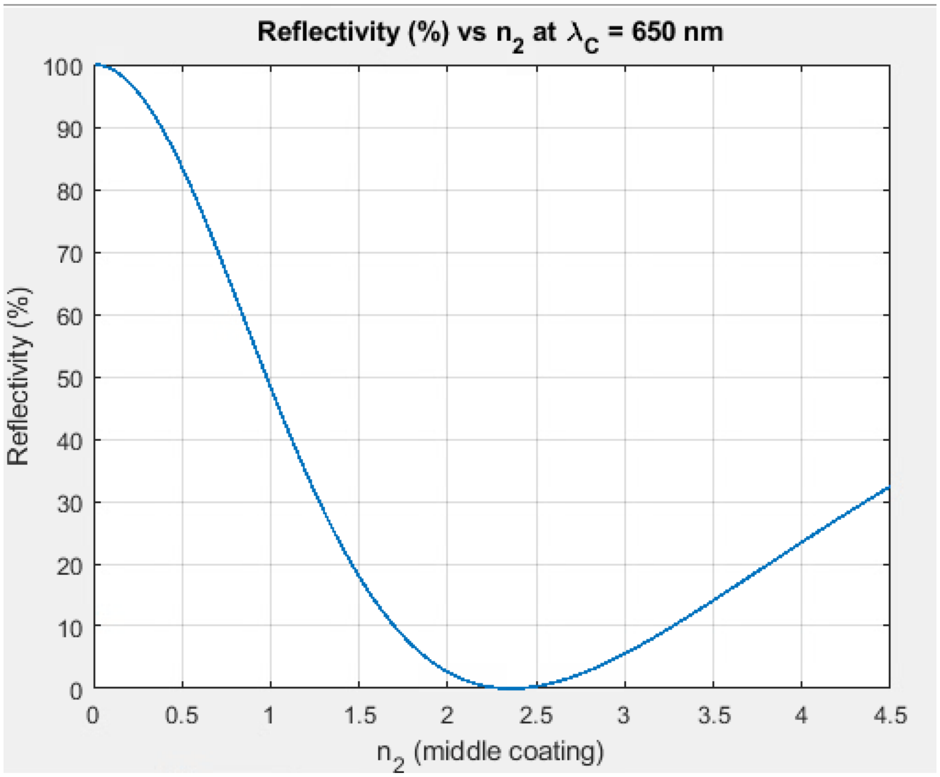

Figure 5.32: Plot of the Reflectivity (as a percentage) vs. n_2 for optimal value of n_2 (triple-layer).

```
Part3_c
n_1 = 1.40
n_2 = 3.15
Minimum reflectivity at n_2 = 2.36
```

Figure 5.33: Simulated optimal value of n_2 (triple-layer)

Part 4 – (a) and (b)– Code, Flowchart and Virtual Simulations

Flow Chart for *Figure 5.34-5.37*

1. Start the program.
2. Clear all variables, the command window, and any open figures.
3. Define constants:
    - Set the refractive indices for air, the first coating, the second coating, and the substrate.
    - Define the design wavelength (650 nm).
    - Assign the imaginary unit j for later use in complex calculations.
4. Calculate Fresnel coefficients for fixed boundaries:
    - Compute the reflection and transmission behavior at the air–first-coating interface.
    - Compute the same properties for the second-coating–substrate interface.
    - Store both results in 2×2 matrices that describe how light reflects and transmits at these surfaces.
5. Define quarter-wave propagation at the design wavelength:
    - Create a propagation matrix that represents how light travels through a single optical layer at $\lambda = 650$ nm.
6. Create the n_2 sweep range:
    - Generate a range of possible middle-layer refractive-index values (from 0 to 4.5).
    - Prepare an empty array to store the reflectivity results for each n_2 value.
7. Loop through all possible n_2 values:
    - For each n_2 in the range:
      a. Calculate how light reflects and transmits between each adjacent layer (first-to-middle and middle-to-second).
      b. Build interface matrices for those transitions.
      c. Combine all matrices (interfaces + propagation) to form a full system that represents the entire optical stack.
      d. Determine how much light is reflected overall.
      e. Save that reflectivity result in the array.
8. Identify the minimum reflectivity:
    - Scan through all stored values to find which n_2 gives the smallest reflectivity.
    - Save that optimal n_2 value for reporting.
9. Plot the results:
    - Display reflectivity (as %) versus the middle-layer index n_2.
    - Label the axes and show a clear title indicating $\lambda C = 650$ nm.
    - Add gridlines for readability.
10. Print numerical results:
    - Output the fixed layer indices (n_1, n_3, n_4).
    - Report the specific n_2 value where minimum reflectivity occurs.

11. Set parameters for transmission analysis:
    - Define how many samples of n_2 will be evaluated.
    - Create a linear range of n_2 values between 1.4 and 3.0.
12. Compute total transmitted power:
    - Run a helper function to calculate total transmitted power across all wavelengths for two cases:
        - One limited to 400–1400 nm (visible + near IR).

        - Another extended 200–2200 nm (broader solar spectrum).
    - Each computation determines how much light passes through the stack at each n_2 value.
13. Plot the transmitted-power results:
    - Plot both wavelength-range results on the same figure for comparison.
    - Use distinct colors and labels to differentiate the two ranges.
    - Add a title, axis labels, legend, and grid.
14. Print key results:
    - Display the maximum transmitted power and the n_2 value where it occurs for both wavelength ranges.

15. Set up wavelength and irradiance arrays:
    - Create a wavelength range between the start and end limits.
    - Compute a matching array of phase shifts and solar spectral irradiance.
    - Initialize storage for total-power results.
16. Loop through each n_2 value:
    - For every n_2:
      a. Calculate how light reflects and transmits between layers.
      b. Create propagation matrices for each wavelength to represent optical travel within the film.
      c. Combine all interface and propagation matrices to simulate the full system.
      d. Determine how much light passes through (transmission).
      e. Multiply by the solar irradiance at each wavelength to find total transmitted power.
      f. Store that total power for this n_2 value.
17. Return total-power data:
    - Send back the computed power values so they can be plotted and analyzed in the main script.

Code for *Figure 5.34-5.37*

```
clc; clear; close all;

%Constants
n_0 = 1.0;  % Incident medium (air)
n_1 = 1.4; % First coating (fixed)
n_3 = 3.15;% Second coating (fixed, adjacent to substrate)
n_4 = 3.5; % Substrate (cell)
Lambda_C = 650; % Center wavelength [nm]
j = 1i; % Imaginary unit

%  (A) Reflectivity vs. n₂ at λC = 650 nm
% Fresnel coefficients for fixed boundaries
r01 = (n_0 - n_1) / (n_0 + n_1);
t01 = 2 * n_0 / (n_0 + n_1);
Q01 = (1 / t01) * [1 r01; r01 1];

r3S = (n_3 - n_4) / (n_3 + n_4);
t3S = 2 * n_3 / (n_3 + n_4);
Q3S = (1 / t3S) * [1 r3S; r3S 1];

% Quarter-wave propagation at λC
Delta = pi / 2;
P = [exp(j * Delta) 0; 0 exp(-j * Delta)];

% Sweep n₂ from 0 to 4.5
n_2_range = 0:0.01:4.50;
Store_Reflectance = zeros(1, numel(n_2_range));

% Loop over all possible n₂ values
for i = 1:numel(n_2_range)
    n_2 = n_2_range(i);

    % Intermediate Fresnel terms
    r12 = (n_1 - n_2) / (n_1 + n_2);
```

Figure 5.34: Code for minimum reflectivity and maximum power transmission using a refractive index sweep – (1 of 4)

```
    t12 = 2 * n_1 / (n_1 + n_2);
    Q12 = (1 / t12) * [1 r12; r12 1];

    r23 = (n_2 - n_3) / (n_2 + n_3);
    t23 = 2 * n_2 / (n_2 + n_3);
    Q23 = (1 / t23) * [1 r23; r23 1];

    % Transfer matrix chain: air → n1 → n2 → n3 → substrate
    T = Q01 * P * Q12 * P * Q23 * P * Q3S;

    % Reflection coefficient Γ and reflectivity R = |Γ|²
    Gamma = T(2,1) / T(1,1);
    Store_Reflectance(i) = abs(Gamma)^2;
end

% Minimum reflectivity and corresponding n₂
[~, Min_Index] = min(Store_Reflectance);
min_n_2 = n_2_range(Min_Index);

% Plot Results
figure;
plot(n_2_range, Store_Reflectance * 100, 'LineWidth', 2);
grid on;
title('Reflectivity vs n_2 at \lambda_C = 650 nm (Triple Layer)', 'FontSize', 22);
xlabel('n_2', 'FontSize', 22);
ylabel('Reflectivity (%)', 'FontSize', 22);

%Print Results
fprintf('n_1 = %.2f, n_3 = %.2f, n_4 = %.2f\n', n_1, n_3, n_4);
fprintf('Minimum Reflectivity at λ_C: n_2 = %.2f\n\n', min_n_2);

%  (B) Total Transmitted Power vs. n₂ for two wavelength ranges
```

Figure 5.35: Code for minimum reflectivity and maximum power transmission using a refractive index sweep – (2 of 4)

```
num_N_2 = 300;  % Number of discrete n₂ samples
Store_n_2 = linspace(1.4, 3.0, num_N_2);

% Compute total transmitted power for each n₂
Store_Total_Power_400_1400 = computeTotalPower( ...
    400, 1400, Q01, Q3S, n_0, n_1, n_3, n_4, Lambda_C, num_N_2, Store_n_2);

Store_Total_Power_200_2200 = computeTotalPower( ...
    200, 2200, Q01, Q3S, n_0, n_1, n_3, n_4, Lambda_C, num_N_2, Store_n_2);

%Plot Results
figure;
plot(Store_n_2, Store_Total_Power_400_1400, 'r', 'LineWidth', 1.8); hold on;
plot(Store_n_2, Store_Total_Power_200_2200, 'b', 'LineWidth', 1.8);
grid on;
title('Transmitted Power vs n_2 (Triple Layer)', 'FontSize', 18);
xlabel('n_2', 'FontSize', 22);
ylabel('Total Power (W/m^2)', 'FontSize', 22);
legend('Wavelength 400-1400 nm', 'Wavelength 200-2200 nm', 'Location', 'best');

%Print Results
[max_Power_1, idx1] = max(Store_Total_Power_400_1400);
fprintf('Max Power (400-1400 nm): %.2f W/m^2 at n_2 = %.2f\n', ...
    max_Power_1, Store_n_2(idx1));

[max_Power_2, idx2] = max(Store_Total_Power_200_2200);
fprintf('Max Power (200-2200 nm): %.2f W/m^2 at n_2 = %.2f\n', ...
    max_Power_2, Store_n_2(idx2));


%  Helper Function – Compute Total Transmitted Power
function Store_Total_Power = computeTotalPower(Lambda_Start, Lambda_End, ...
    Q01, Q3S, n_0, n_1, n_3, n_4, Lambda_C, numN2, Store_n_2)
```

Figure 5.36: Code for minimum reflectivity and maximum power transmission using a refractive index sweep – (3 of 4)

```
    % Wavelength and irradiance arrays
    Lambda_Array = Lambda_Start:Lambda_End;
    Delta_Array  = (pi / 2) * (Lambda_C ./ Lambda_Array);
    IRRAD_Array  = 6.16e15 ./ (Lambda_Array.^5 .* (exp(2484 ./ Lambda_Array) - 1));

    Store_Total_Power = zeros(1, numN2);

    % Loop over each n₂ value
    for idx = 1:numN2
        n_2 = Store_n_2(idx);

        % Fresnel terms for layer transitions
        r12 = (n_1 - n_2) / (n_1 + n_2);
        r23 = (n_2 - n_3) / (n_2 + n_3);
        t12 = 2 * n_1 / (n_1 + n_2);
        t23 = 2 * n_2 / (n_2 + n_3);
        Q12 = (1 / t12) * [1 r12; r12 1];
        Q23 = (1 / t23) * [1 r23; r23 1];

        % Propagation phase matrix elements
        Pcol = [exp(1j * Delta_Array); exp(-1j * Delta_Array)];
        Tcol = zeros(2, numel(Lambda_Array));  % Holds field values per λ

        % Loop through wavelength spectrum
        for k = 1:numel(Lambda_Array)
            P = [Pcol(1, k) 0; 0 Pcol(2, k)];
            Tcol(:, k) = Q01 * P * Q12 * P * Q23 * P * Q3S * [1; 0];
        end

        % Transmission and total power calculation
        Tau   = 1 ./ Tcol(1, :);
        Trans = (abs(Tau).^2) * (n_4 / n_0);
        Store_Total_Power(idx) = sum(Trans .* IRRAD_Array);
    end
```

Figure 5.37: Code for minimum reflectivity and maximum power transmission using a refractive index sweep – (4 of 4)

Virtual Simulations for *Figure 5.38-5.40*

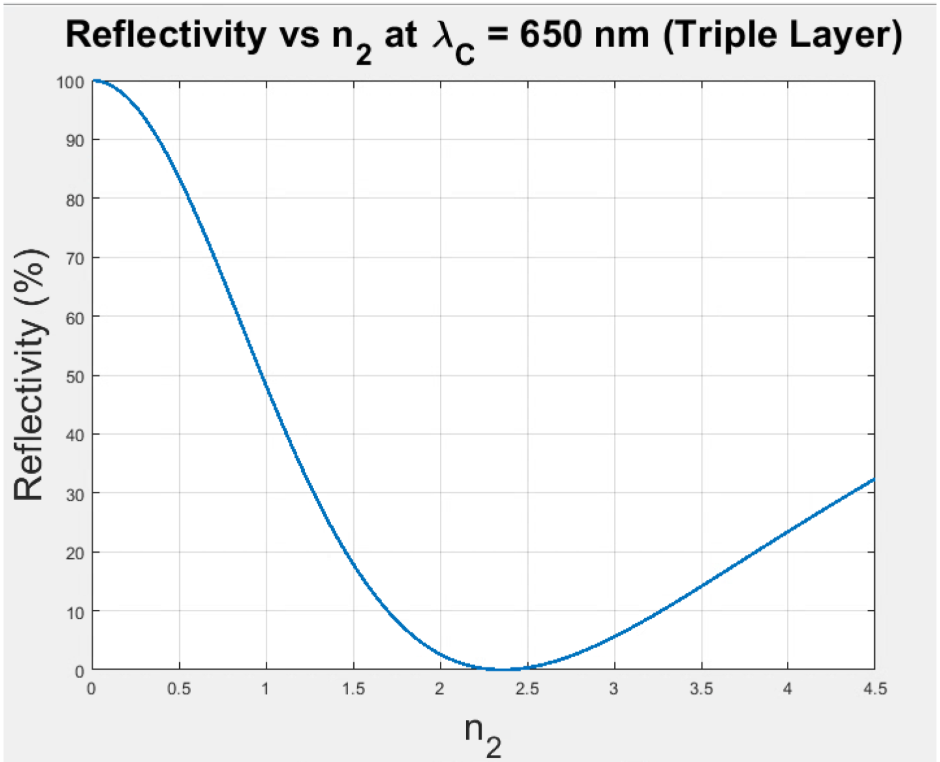

Figure 5.38: Plot of Reflectivity (%) vs. n_2 for triple-layer to identify minimum reflectivity.

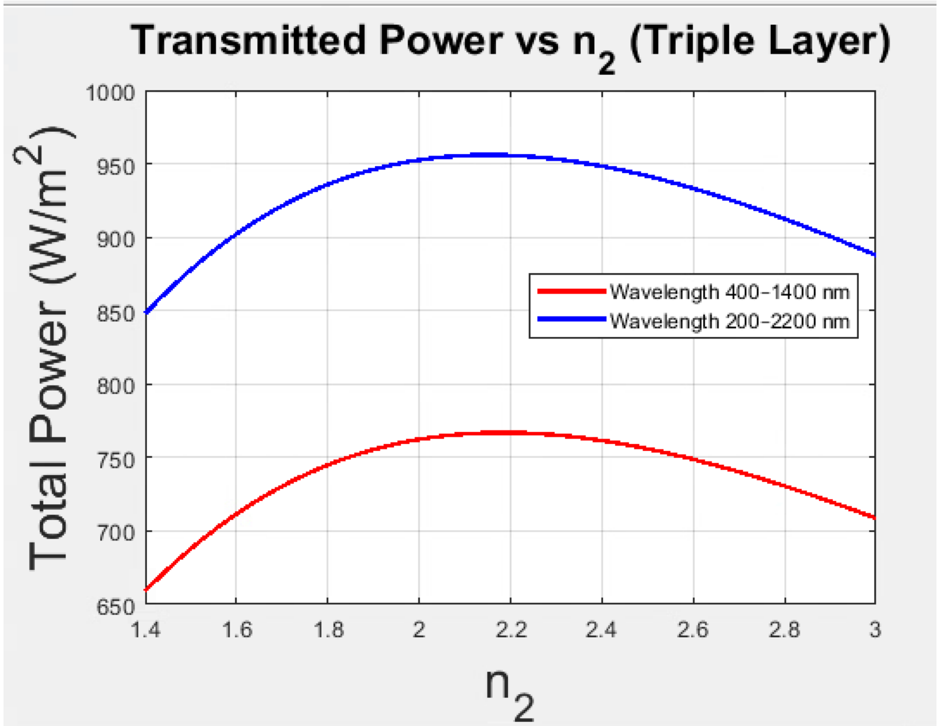

Figure 5.39: Plot of Total Transmitted Power vs. n_2 for triple-layer.

```
n_1 = 1.40, n_3 = 3.15, n_4 = 3.50
Minimum Reflectivity at λ_C: n_2 = 2.36

Max Power (400-1400 nm): 766.95 W/m^2 at n_2 = 2.18
Max Power (200-2200 nm): 955.95 W/m^2 at n_2 = 2.15
```

Figure 5.40: Result of Total Transmitted Power vs. n_2 for triple-layer.

Appendix A:

Hence, the transmissivity, Γ, and reflectivity, τ, coefficients across the double-layer system can be calculated using the refractive indices, as seen below:

$$\Gamma_{0,1} = \frac{n_0 - n_1}{n_0 + n_1}, \qquad \tau_{0,1} = \frac{2n_0}{n_0 + n_1}$$
$$\Gamma_{1,2} = \frac{n_1 - n_2}{n_1 + n_2}, \qquad \tau_{1,2} = \frac{2n_1}{n_1 + n_2}$$
$$\Gamma_{2,3} = \frac{n_2 - n_3}{n_2 + n_3}, \qquad \tau_{2,3} = \frac{2n_2}{n_2 + n_3}$$

The transmissivity, Γ, and reflectivity, τ, coefficients for the dynamic matrices Q can be calculated by:

$$Q_{0,1} = \frac{1}{\tau_{0,1}}\begin{bmatrix} 1 & \Gamma_{0,1} \\ \Gamma_{0,1} & 1 \end{bmatrix}$$
$$Q_{1,2} = \frac{1}{\tau_{1,2}}\begin{bmatrix} 1 & \Gamma_{1,2} \\ \Gamma_{1,2} & 1 \end{bmatrix}$$
$$Q_{2,3} = \frac{1}{\tau_{2,3}}\begin{bmatrix} 1 & \Gamma_{2,3} \\ \Gamma_{2,3} & 1 \end{bmatrix}$$

Additionally, the phase thickness δ_m is given by:

$$\delta_m = \frac{2\pi n_m d_m}{\lambda} \text{where} d_m = \frac{\lambda_c}{4n_m}$$

The thickness of each layer of the anti-reflective coating is given by d_m.
It is stated that:

$$\delta_m = \frac{\pi}{2} \text{at the center wavelength} \lambda_C = 650 \text{ nm.}$$

Hence, two propagation matrices P are calculated using:

$$P_m = \begin{bmatrix} \exp(j\delta_m) & 0 \\ 0 & \exp(-j\delta_m) \end{bmatrix}$$

P_1 and P_2 are given by the following matrices:

$$P_1 = \begin{bmatrix} \exp(j\delta_1) & 0 \\ 0 & \exp(-j\delta_1) \end{bmatrix} \qquad P_2 = \begin{bmatrix} \exp(j\delta_2) & 0 \\ 0 & \exp(-j\delta_2) \end{bmatrix}$$

We can implement these formulas into the transfer matrix T.

Appendix B:

Phase Thickness and Propagation Matrices

Taking the phase thickness δ_m at the central wavelength $\lambda_C = 650$ nm, we have:

$$\delta_m = \frac{2\pi n_m}{\lambda_C} \times \frac{\lambda_C}{4n_m} = \frac{\pi}{2}$$

We now calculate matrices P_1 and P_2:

$$P_1 = \begin{bmatrix} \exp(j\frac{\pi}{2}) & 0 \\ 0 & \exp(-j\frac{\pi}{2}) \end{bmatrix} = j\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$P_2 = \begin{bmatrix} \exp(j\frac{\pi}{2}) & 0 \\ 0 & \exp(-j\frac{\pi}{2}) \end{bmatrix} = j\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Dynamic Matrices Q

The dynamic matrices Q can be calculated by:

$$Q_{0,1} = \frac{1}{\tau_{0,1}}\begin{bmatrix} 1 & \Gamma_{0,1} \\ \Gamma_{0,1} & 1 \end{bmatrix}$$

$$Q_{1,2} = \frac{1}{\tau_{1,2}}\begin{bmatrix} 1 & \Gamma_{1,2} \\ \Gamma_{1,2} & 1 \end{bmatrix}$$

$$Q_{2,3} = \frac{1}{\tau_{2,3}}\begin{bmatrix} 1 & \Gamma_{2,3} \\ \Gamma_{2,3} & 1 \end{bmatrix}$$

We now calculate the transfer matrix T:

$$T = Q_{0,1}P_1Q_{1,2}P_2Q_{2,3}$$

Implementing P_1, P_2, $Q_{0,1}$, $Q_{1,2}$, and $Q_{2,3}$ into the transfer matrix T:

$$T = \frac{j^2}{\tau_{0,1}\tau_{1,2}\tau_{2,3}}\begin{bmatrix} 1 & \Gamma_{0,1} \\ \Gamma_{0,1} & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} 1 & \Gamma_{1,2} \\ \Gamma_{1,2} & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} 1 & \Gamma_{2,3} \\ \Gamma_{2,3} & 1 \end{bmatrix}$$

Simplifying:

$$T = \begin{bmatrix} 1-\Gamma_{0,1}\Gamma_{1,2} & \Gamma_{0,1}-\Gamma_{1,2} \\ \Gamma_{0,1}-\Gamma_{1,2} & 1-\Gamma_{0,1}\Gamma_{1,2} \end{bmatrix}\begin{bmatrix} 1 & \Gamma_{2,3} \\ \Gamma_{2,3} & 1 \end{bmatrix}$$

$$T = \begin{bmatrix} 1-\Gamma_{0,1}\Gamma_{1,2}+(\Gamma_{0,1}-\Gamma_{1,2})\Gamma_{2,3} & (1-\Gamma_{0,1}\Gamma_{1,2})\Gamma_{2,3}+\Gamma_{0,1}-\Gamma_{1,2} \\ \Gamma_{0,1}-\Gamma_{1,2}+(1-\Gamma_{0,1}\Gamma_{1,2})\Gamma_{2,3} & (\Gamma_{0,1}-\Gamma_{1,2})\Gamma_{2,3}+1-\Gamma_{0,1}\Gamma_{1,2} \end{bmatrix}$$

Reflection Coefficient and Reflectance

We now calculate the reflection coefficient R:

$$R = |\Gamma|^2 = \left|\frac{T_{2,1}}{T_{1,1}}\right|^2 = \left|\frac{\Gamma_{0,1}-\Gamma_{1,2}+(1-\Gamma_{0,1}\Gamma_{1,2})\Gamma_{2,3}}{1-\Gamma_{0,1}\Gamma_{1,2}+(\Gamma_{0,1}-\Gamma_{1,2})\Gamma_{2,3}}\right|^2$$

In order to find n_2, we extrapolate from the following:

$$T = \frac{T_{2,1}}{T_{1,1}}$$

Where:

$$T_{1,1} = 1-\Gamma_{0,1}\Gamma_{1,2}+(\Gamma_{0,1}-\Gamma_{1,2})\Gamma_{2,3}$$
$$T_{2,1} = \Gamma_{0,1}-\Gamma_{1,2}+(1-\Gamma_{0,1}\Gamma_{1,2})\Gamma_{2,3}$$

We can derive the following relationship:

$$n_0 n_2^2 = n_{\text{cell}} n_1^2$$
$$n_2 = \sqrt{\frac{n_{\text{cell}} n_1^2}{n_0}}$$

For $n_0 = 1$:

$$n_2 = \sqrt{n_{\text{cell}} n_1^2}$$

Appendix C:

Derivation for Analytical n_2 for 3-Layer Coating

Using our previous 2-layer model for the Total Matrix T, we can extend this out to a 3rd layer:

$$T = Q_{0,1}\, P_1\, Q_{1,2}\, P_2\, Q_{2,3}\, P_3\, Q_{3,4}$$

$$T = \frac{j^3}{\tau_{0,1}\tau_{1,2}\tau_{2,3}} \begin{bmatrix} 1 & \Gamma_{0,1} \\ \Gamma_{0,1} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \Gamma_{1,2} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \Gamma_{2,3} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \Gamma_{3,4} & 1 \end{bmatrix}$$

$$T = [\begin{matrix} 1 - \Gamma_{0,1}\Gamma_{1,2} & \Gamma_{0,1} - \Gamma_{1,2} \\ \Gamma_{0,1} - \Gamma_{1,2} & 1 - \Gamma_{0,1}\Gamma_{1,2} \end{matrix}][\begin{matrix} 1 & \Gamma_{2,3} \\ \Gamma_{2,3} & 1 \end{matrix}]$$

$$T = [\begin{matrix} 1 - \Gamma_{0,1}\Gamma_{1,2} + (\Gamma_{0,1} - \Gamma_{1,2})\Gamma_{2,3} & (\Gamma_{0,1} - \Gamma_{1,2})\Gamma_{2,3} + \Gamma_{0,1} - \Gamma_{1,2} \\ \Gamma_{0,1} - \Gamma_{1,2} + (1 - \Gamma_{0,1}\Gamma_{1,2})\Gamma_{2,3} & (\Gamma_{0,1} - \Gamma_{1,2})\Gamma_{2,3} + 1 - \Gamma_{0,1}\Gamma_{1,2} \end{matrix}]$$

$$T_{21} = (\Gamma_{0,1} - \Gamma_{1,2})(1 - \Gamma_{2,3}\Gamma_{3,4}) + (1 - \Gamma_{0,1}\Gamma_{1,2})(\Gamma_{2,3} - \Gamma_{3,4})$$

$$T_{11} = (\Gamma_{0,1}\Gamma_{1,2})(1 - \Gamma_{2,3}\Gamma_{3,4}) + (\Gamma_{0,1} - \Gamma_{1,2})(\Gamma_{2,3} - \Gamma_{3,4})$$

We can note that $\Gamma = 0$ when we want to maximize transmission:

$$\Gamma = \frac{T_{21}}{T_{11}} = \frac{(\Gamma_{0,1} - \Gamma_{1,2})(1 - \Gamma_{2,3}\Gamma_{3,4}) + (1 - \Gamma_{0,1}\Gamma_{1,2})(\Gamma_{2,3} - \Gamma_{3,4})}{(\Gamma_{0,1}\Gamma_{1,2})(1 - \Gamma_{2,3}\Gamma_{3,4}) + (\Gamma_{0,1} - \Gamma_{1,2})(\Gamma_{2,3} - \Gamma_{3,4})} = 0$$

Calculation for Γ for each layer:

$$\Gamma_{0,1} = \frac{n_0 - n_1}{n_0 + n_1}, \Gamma_{1,2} = \frac{n_1 - n_2}{n_1 + n_2}, \Gamma_{2,3} = \frac{n_2 - n_3}{n_2 + n_3}, \Gamma_{3,4} = \frac{n_3 - n_4}{n_3 + n_4}$$

$$\begin{aligned} T_{21} &= \Gamma_{0,1} - \Gamma_{0,1}\Gamma_{2,3}\Gamma_{3,4} - \Gamma_{1,2} + \Gamma_{1,2}\Gamma_{2,3}\Gamma_{3,4} + \Gamma_{2,3} - \Gamma_{3,4} \\ &\quad - \Gamma_{0,1}\Gamma_{1,2}\Gamma_{2,3} + \Gamma_{0,1}\Gamma_{1,2}\Gamma_{3,4} \\ &= \frac{n_0 - n_1}{n_0 + n_1} - \frac{n_0 - n_1}{n_0 + n_1}\frac{n_2 - n_3}{n_2 + n_3}\frac{n_3 - n_4}{n_3 + n_4} - \frac{n_1 - n_2}{n_1 + n_2} \\ &\quad + \frac{n_1 - n_2}{n_1 + n_2}\frac{n_2 - n_3}{n_2 + n_3}\frac{n_3 - n_4}{n_3 + n_4} + \frac{n_2 - n_3}{n_2 + n_3} - \frac{n_3 - n_4}{n_3 + n_4} \\ &\quad - \frac{n_0 - n_1}{n_0 + n_1}\frac{n_1 - n_2}{n_1 + n_2}\frac{n_2 - n_3}{n_2 + n_3} + \frac{n_0 - n_1}{n_0 + n_1}\frac{n_1 - n_2}{n_1 + n_2}\frac{n_3 - n_4}{n_3 + n_4} \end{aligned}$$

$$2\, n_0\, n_4\, n_2^2 - 2\, n_1^2\, n_3^2 = 0$$

$$n_2^2 = \frac{n_1^2\, n_3^2}{n_0\, n_4}$$

The above expression can be simplified to:

$$n_2 = \frac{n_1\, n_3}{\sqrt{n_0\, n_4}}.$$