

Projektarbeit: Optische Dünnschichtsysteme

Oğuzhan Aygün, Abdul-Malik Jakupi

WiSe 2025

Inhaltsverzeichnis

1	Das Prinzip optischer Dünnschichtsysteme	3
2	Mathematisch-Physikalisches Modell	4
2.1	Brechungsindex	4
2.2	Reflexion und Transmission	4
2.3	Die Transfermatrixmethode	5
3	Algorithmische Umsetzung	6
3.1	Materialmodellierung	6
3.2	Berechnung der Grenzflächenkoeffizienten	7
3.3	Aufbau der Transfermatrix	7
3.4	Berechnung des Reflexionsgrades	8
4	Benutzerhandbuch	9
5	Simulationsergebnisse	10

1 Das Prinzip optischer Dünnschichtsysteme

Der Begriff optische Dünnschichtsysteme bezeichnet üblicherweise ein System aus oft nur wenigen Nanometer dünnen Schichten, dessen Zweck darin liegt, Wellen und ihre Interferenz gezielt zu nutzen, um einen gewünschten Reflexions- und Transmissionsgrad zu erreichen. Dünnschichtsysteme sind ein wichtiger Bestandteil im Bereich der Optik und haben auch in anderen Fachgebieten einige Anwendungen, z.B.:

- Antireflexbeschichtungen in Brillengläsern, Hochreflexschichten wie in Spiegeln oder Kameraobjektiven in der Optik,
- Beschichtung von Geräten in der Medizintechnik,
- Herstellung von Prozessoren bei Mikroelektronik.

Dafür bedient man sich meist verschiedenster Materialien wie z.B. Titanoxid, Magnesiumfluorid oder Aluminiumoxid, die alle einen individuellen Brechungsindex besitzen. Diese Eigenschaften werden in allen Anwendungen als das essenzielle Werkzeug genutzt, um das jeweilige Ziel im gewünschten Wellenbereich zu erreichen. Dies kann z.B. im Fall von Brillengläsern destruktive Interferenz sein, bei anderen Anwendungen, wie Spiegeln, aber auch konstruktive Interferenz.

Im Folgenden wird Weg zur Berechnung des Reflexionsgrades beschrieben, um ein möglichst realistisches Programm zur Erfassung eben dieses zu entwickeln. Das größte Problem, das es hier zu lösen gilt, ist die Berücksichtigung aller Teilwellen die bei jedem Grenzübergang, wie in Abbildung 1 zu sehen ist, innerhalb des Schichtsystems entstehen.

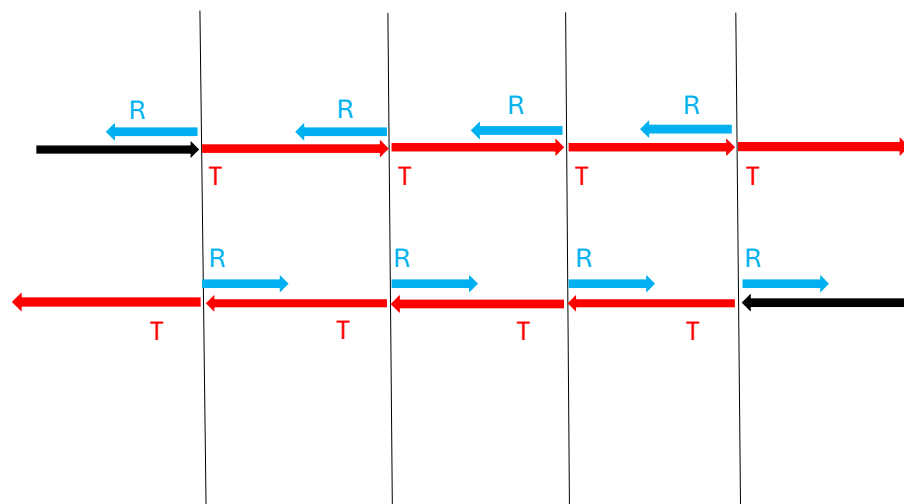


Abbildung 1: Darstellung der Reflexion und Transmission im Mehrschichtsystem

2 Mathematisch-Physikalisches Modell

Um das erwähnte Problem zu lösen wird die Transfermatrixmethode angewandt, welche es ermöglicht auch unter Berücksichtigung der Teilwellen einen Reflexionsgrad zu bestimmen. Bevor die Transfermatrixmethode angewandt werden kann, müssen noch andere Eigenschaften ermittelt werden, auf welche in den folgenden Unterkapiteln eingegangen wird.

2.1 Brechungsindex

Der komplexe Brechungsindex einer Schicht ist definiert als:

$$\tilde{n} = n + i\kappa. \quad (1)$$

Dabei beschreibt der Realteil n die Phasengeschwindigkeit und der Imaginärteil κ die Absorption innerhalb des Mediums. Die Wellenausbreitung in z -Richtung lässt sich durch Einsetzen von Gleichung (1) im folgenden Ansatz darstellen:

$$E(z) \propto e^{i\tilde{n}\frac{\omega}{c}z} = \underbrace{e^{i\frac{n\omega}{c}z}}_{\text{Phase}} \cdot \underbrace{e^{-\frac{\kappa\omega}{c}z}}_{\text{Dämpfung}}. \quad (2)$$

Diese Aufteilung ist fundamental für die Aufstellung der Propagationsmatrix innerhalb der Transfermatrixmethode.

Des Weiteren ist es wichtig den Brechungsindex als Funktion der Wellenlänge $n(\lambda)$ zu betrachten, um präzise Resultate zu erlangen, da Dispersion berücksichtigt werden muss. Dafür wird im Laufe der Arbeit, wenn möglich, die Sellmeier-Gleichung verwendet [3]:

$$n^2(\lambda) = 1 + \sum_i \frac{B_i \lambda^2}{\lambda^2 - C_i}. \quad (3)$$

Für manche Materialien werden zum Zweck eines präziseren Ergebnisses von der Sellmeier-Gleichung abweichende Formeln oder Interpolation von Messwertetabellen genutzt. [2][4].

2.2 Reflexion und Transmission

Reflexions- und Transmissionskoeffizienten von elektromagnetischen Wellen an einer Grenzfläche zwischen zwei Schichten werden mit den fresnelschen Formeln ermittelt [1]. θ_1 ist der Einfallswinkel des Lichts auf die Grenzfläche und θ_2 der Brechungswinkel. Der Brechungswinkel lässt sich durch das snelliussche Brechungsgesetz definieren [1]:

$$n_1 \sin \theta_1 = n_2 \sin \theta_2. \quad (4)$$

Durch Auflösen nach θ_2 entsteht folgende Gleichung:

$$\theta_2 = \arcsin\left(\frac{n_1}{n_2} \sin \theta_1\right) \quad (5)$$

Zusätzlich muss die Polarisierung berücksichtigt werden. Abhängig davon, ob das Licht senkrecht oder parallel zur Einfallsebene schwingt, variieren die Formeln.

Senkrechte Polarisation:

$$r_s = \frac{n_1 \cos \theta_1 - n_2 \cos \theta_2}{n_1 \cos \theta_1 + n_2 \cos \theta_2}, \quad t_s = \frac{2n_1 \cos \theta_1}{n_1 \cos \theta_1 + n_2 \cos \theta_2} \quad (6)$$

Parallele Polarisation:

$$r_p = \frac{n_2 \cos \theta_1 - n_1 \cos \theta_2}{n_2 \cos \theta_1 + n_1 \cos \theta_2}, \quad t_p = \frac{2n_1 \cos \theta_1}{n_2 \cos \theta_1 + n_1 \cos \theta_2} \quad (7)$$

n_1 und n_2 stellen die Brechungsindizes der linken und rechten Schicht beim Grenzübergang dar. Dieser Prozess liefert nun die Koeffizienten an der Grenzfläche zwischen zwei Schichten und muss für jede Schicht, bzw. jeden Grenzübergang erfolgen. Die Resultate werden in der Transfermatrixmethode verwertet.

2.3 Die Transfermatrixmethode

Es handelt sich bei der Transfermatrixmethode um eine elegante mathematische Lösung für das Problem der unendlichen Teilwellen [1]. Anstatt den Verlauf des Lichtstrahls zu verfolgen, wird das gesamte Mehrschichtsystem betrachtet. Die Grundlage hierfür sind die fresnelschen Formeln, da sie das Verhältnis der Amplituden auf der linken und rechten Seite der Grenzfläche beschreiben und dadurch folgendes Gleichungssystem bilden:

$$\begin{aligned} R'_{i+1} &= R_i \cdot t_{i,i+1} + L'_{i+1} \cdot r_{i+1,i} \\ L_i &= R_i \cdot r_{i,i+1} + L'_{i+1} \cdot t_{i+1,i} \end{aligned} \quad (8)$$

Das resultierende Gleichungssystem kann auch in der Matrixschreibweise

$$\begin{pmatrix} t_{i,i+1} & 0 \\ -r_{i,i+1} & 1 \end{pmatrix} \begin{pmatrix} R_i \\ L_i \end{pmatrix} = \begin{pmatrix} 1 & -r_{i+1,i} \\ 0 & t_{i+1,i} \end{pmatrix} \begin{pmatrix} R'_{i+1} \\ L'_{i+1} \end{pmatrix} \quad (9)$$

geschrieben werden. Wird nach $\begin{pmatrix} R_i \\ L_i \end{pmatrix}$ aufgelöst erhält man

$$\begin{pmatrix} R_i \\ L_i \end{pmatrix} = \frac{1}{t_{i,i+1}} \begin{pmatrix} 1 & -r_{i+1,i} \\ r_{i,i+1} & t_{i,i+1}t_{i+1,i} - r_{i,i+1}r_{i+1,i} \end{pmatrix} \begin{pmatrix} R'_{i+1} \\ L'_{i+1} \end{pmatrix} \quad (10)$$

was das Muster der benötigten Transfermatrix liefert. In Abbildung 2 befindet sich eine grafische Darstellung der Herleitung.

Es werden immer jeweils die zwei bereits erwähnten Arten von Matrizen gebildet, dabei bestimmt die Grenzflächenmatrix D die Reflexion und Transmission an der Grenzfläche, während die Propagationsmatrix P die Phasenverschiebung innerhalb einer Schicht beschreibt. Die Grenzflächenmatrix besitzt die Form

$$D = \frac{1}{t} \begin{pmatrix} 1 & r \\ r & 1 \end{pmatrix}, \quad (11)$$

wobei r dem berechneten Reflexions- und t dem Transmissionskoeffizienten aus den fresnelschen Formeln entsprechen. Die Propagationsmatrix hat die Form

$$P = \begin{pmatrix} e^{-i\varphi} & 0 \\ 0 & e^{i\varphi} \end{pmatrix} \quad (12)$$

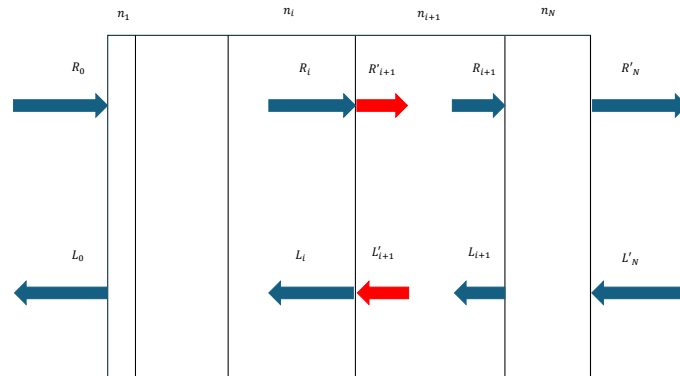


Abbildung 2: Abbildung der Herleitung des Gleichungssystems

mit

$$\varphi = \frac{2\pi n d \cos(\theta_2)}{\lambda}, \quad (13)$$

wobei d die Dicke, n der Brechungsindex und θ_2 der Brechungswinkel der Schicht sind. Daraufhin werden wie in der vorherigen Herleitung alle Matrizen der Reihe nach miteinander multipliziert

$$M = D_0 \cdot P_0 \cdot D_1 \cdot P_1 \dots D_{n-1} \cdot P_n \cdot D_n \quad (14)$$

und anschließend genutzt um den Reflexionsgrad R , nach der Definition der Intensität einer elektromagnetischen Welle,

$$R = \left| \frac{M_{21}}{M_{11}} \right|^2 \quad (15)$$

zu bestimmen.

3 Algorithmische Umsetzung

Die mathematischen Beziehungen, die in Kapitel 2 beschrieben werden, finden in einer modular strukturierten Softwarearchitektur Umsetzung. Diese teilt den Berechnungsprozess in klar voneinander getrennte Komponenten auf. Das Programm umfasst die Materialverwaltung, die Berechnung der Fresnel-Koeffizienten, die konsistente Konstruktion der Transfermatrix und darauf basierende Reflexionsberechnungen.

Alle Schritte sind durch eigene Funktionen oder Klassen voneinander abgegrenzt, was das Programm wartbar und erweiterbar macht und es ermöglicht, dass es mit beliebigen Schichtsystemen verwendet werden kann

3.1 Materialmodellierung

Zur Repräsentation einer Schicht dient die Klasse `Material`. Sie speichert den Namen `name` des verwendeten Materials, eine Schichtdicke `d` und einen Berechnungstyp `n_type`. Dieser Berechnungstyp entscheidet dann welche optionalen Parameter der Klasse festgelegt werden, die später zur Berechnung von Brechungsindizes genutzt wird. Bei diesen optionalen Parametern handelt es sich um:

- einen festen komplexen Brechungsindex n ,
- die Sellmeier-Koeffizient B und C und dem Parameter $A = 0$ für leichten Abweichungen,
- eine benutzerdefinierte Funktion in Form von `stringformula`, welche später mit `eval(formula)` verwertet wird,
- oder eine Messwerttabelle `table` mit den Spalten Wellenlänge | Realteil- n | Imaginärteil- n

Für die gegebene Wellenlänge λ liefert sie den komplexen Brechungsindex der entsprechenden Schicht. Je nach Material kommen verschiedene bewährte Formeln zum Einsatz. Dadurch wird eine realistische Nachbildung der Wellenlängenabhängigkeit möglich, ohne dass Anpassungen an den späteren optischen Berechnungen erforderlich sind. Die Klasse ist zudem dafür zuständig, die Materialparameter aus der JSON-Datei zu laden. Die Schichtfolge kann flexibel über die Benutzeroberfläche festgelegt werden, ohne dass eine Anpassung des Programmcodes erforderlich ist.

3.2 Berechnung der Grenzflächenkoeffizienten

Die Funktion

```
fresnel_coefficients(n1, n2, theta1, polarization)
```

übernimmt die numerische Umsetzung der Fresnel-Gleichungen an einer Grenzfläche zwischen zwei Schichten. Sie berechnet zuerst den Brechungswinkel θ_2 mit dem Snelliusschen Gesetz [1], die Reflexions- und Transmissionskoeffizienten r und t und den für die Transfermatrix benötigten Phasenverlauf. Die Funktion unterscheidet streng zwischen den beiden Polarisierungen

'Senkrecht' (s-polarisiert) und 'Parallel' (p-polarisiert)

und gibt die zugehörigen Koeffizienten zurück. Damit ist die Polarisierung vollständig als Parameter steuerbar, was eine Simulation beider physikalischen Fälle ermöglicht.

3.3 Aufbau der Transfermatrix

Die Funktion

```
transfer_matrix(material_list, d_list, wavelength, polarization, theta0)
```

durchläuft alle Schichten und führt für jede Grenzfläche die bekannten Schritte aus dem mathematisch-physikalischen Modell aus. Sie ermittelt die Fresnelschen Koeffizienten aus den Brechungsindizes der angrenzenden Schichten, wobei für die Brechungsindizes für jede einzelne Wellenlänge die Material-Methode

```
reflective_index(self, wavelength)
```

genutzt wird. Daraufhin wird mit den Reflexions- und Transmissionskoeffizienten die Grenzflächenmatrix D generiert und anschließend für alle endlichen Schichten in Übereinstimmung mit der Dicke, dem Einfallswinkel und dem Brechungsindex, auch die Propagationsmatrix P . Zuletzt werden beide Matrizen multiplikativ in die Gesamtmatrix eingebaut. Die Transfermatrix wird iterativ aufgebaut durch

$$M \leftarrow M \cdot D \cdot P, \quad (16)$$

was sicherstellt, dass der aktuelle optische Zustand des Lichtfelds im Mehrschichtsystem erhalten bleibt. Dabei werden Schichten mit unendlicher Dicke (Luft und Substrat) automatisch identifiziert, und ihnen wird keine Propagationsmatrix zugewiesen. So wird das physikalische Modell exakt wiedergegeben

3.4 Berechnung des Reflexionsgrades

Die Funktion

```
reflectance(material_list, wavelengths, polarization, theta)
```

fasst alle vorher beschriebenen Schritte zu einer vollständigen Simulation zusammen. Sie übernimmt, die Umrechnung der Schichtdicken in SI-Einheiten, berechnet beliebige spektrale Abtastungen, berücksichtigt dabei mehrere oder einen Einfallswinkel, ruft die Transfermatrix auf und kümmert sich zuletzt um die Herausarbeitung des Reflexionskoeffizienten. Der Amplitudenreflexionskoeffizient wird aus der Transfermatrix bestimmt:

$$r = M[1,0] / M[0,0]$$

woraus letztlich der Reflexionsgrad resultiert:

$$R = |r|^2. \quad (17)$$

Weil die Berechnung vollständig vektorisiert ist, können ganze Winkel- oder Wellenlängenbereiche mit nur einem Funktionsaufruf evaluiert werden, ohne dass der Nutzer zusätzliche Einstellungen vornehmen muss.

Programmausschnitt 1: Funktion zur Berechnung des Reflexionsgrades

```
def reflectance(material_list, wavelengths, polarization, theta):
    R = []

    d_list = [i.d * 1e-9 for i in material_list if i.d != np.
               inf]
    wls = np.atleast_1d(wavelengths)
    thetas = np.atleast_1d(theta)

    for wl, theta0 in np.broadcast(wls, thetas):
        M = transfer_matrix(
            material_list, d_list, float(wl), polarization,
            float(theta0)
        )
        r = M[1, 0] / M[0, 0]
        R.append(np.abs(r) ** 2)
    return np.array(R)
```


4 Benutzerhandbuch

Hier wird die grafische Nutzeroberfläche der Software präsentiert, zusammen mit einzelnen Funktionen, denen es eine Erläuterung bedarf.

Zu aller erst müssen die Abhängigkeiten installiert werden. Da die Software auf Python basiert, kann dies am einfachsten über das Paketverwaltungsprogramm pip getätigt werden. In der Repository ist eine `requirements.txt` dafür zu finden, welche über pip mit dem Befehl

```
pip install -r requirements.txt
```

zu installieren sind. Beachtet werden sollte hierbei, dass eine globale Installation auf Linux Distributionen nicht auf denselben Weg möglich ist. Hierfür sollte vorher mit

```
python -m venv .venv
```

eine virtuelle Umgebung für das Programm erstellt werden. Die Installation ist daraufhin nach Aktivieren dieser virtuellen Umgebung wie vorher beschrieben auszuführen. Anschließend wird die Software nach erfolgreicher Installation mit

```
Python GUI.py
```

ausgeführt.

Alternativ kann das `Start_GUI.bat` Skript für Windows und das `Start_GUI.sh` Skript für Linux genutzt werden, um das Programm und seine Abhängigkeiten zu installieren und zu starten. Beide Skripte erstellen wie oben beschrieben eine virtuelle Umgebung für Python, installieren automatisch die Abhängigkeiten in `requirements.txt` und starten danach das Programm.

Nun sollte eine GUI zu sehen sein, welche aus einer Tabelle, einem Graphen und einzelnen zu bestimmenden Parametern unter beiden Flächen besteht. Dies ist zusätzlich in Abbildung 3 zusehen. Es stehen in der Tabelle bereits 2 Zeilen vor angefertigt zur Verfügung. Sie stellen in der Software das Einfallsmittel und Substrat dar und können entsprechend nicht entfernt werden. Am Einfallsmittel befindet sich in der letzten Spalte ein Button, welcher eine weitere Zeile, bzw. Schicht, direkt nach ihr selbst generiert und einfügt. Alle Zeilen, die eine Schicht repräsentieren, welche weder das Einfallsmittel, noch das Substrat sind, besitzen zwei Buttons in ihrer letzten Spalte. Das Pluszeichen hat dieselbe Funktion, wie beim Einfallsmittel, während das Minuszeichen die Schicht, in der sie beinhaltet ist, aus der Simulation löscht.

In der ersten Spalte unter `Material` ist ein Dropdown-Menü mit der Aufschrift `Preset` zu sehen. Benutzer können hier eine Auswahl von mehreren vorgefertigten Schichten wählen, welche bereits alle Parameter belegt haben. Es steht dem Nutzer jedoch offen die Dicke der Schicht anzupassen. Alternativ besteht auch die Möglichkeit mit dem Button unter der Tabelle ein neues Material anzulegen. Das Betätigen öffnet ein zusätzliches Fenster, in dem der Nutzer einen Namen für das Material und einen Berechnungstypen angeben kann. Die gegebenen Optionen sind:

- Berechnung durch die Sellmeier-Gleichung,

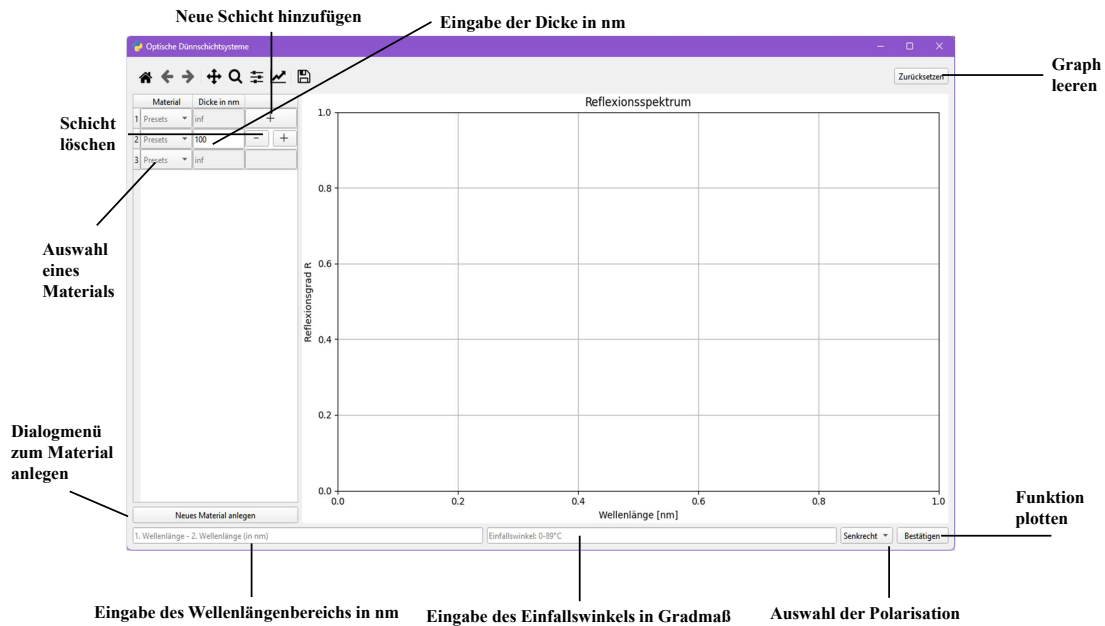


Abbildung 3: Beschreibung der Benutzeroberfläche

- Angabe eines festen Brechungsindex,
- Nutzen einer benutzerdefinierten Formel und
- Interpolation von Daten einer Messwertetabelle.

Das Bestätigen der Eingabe erstellt ein neues Materialobjekt und fügt dessen Daten in das `Material.json` im Root ein, damit sie zusammen mit den vorgefertigten Materialien für den Nutzer langfristig gespeichert sind. In Abbildung 4 ist das Dialogfenster abgebildet.

Bestätigen plottet den Reflexionsgrad in Abhängigkeit der gegebenen Parameter in den Graphen. Wichtig ist hierbei die Definition des Wellenlängenbereichs und des Einfallswinkels, da sich die berechnete Funktion sich je nach Angabe ändert. Hierbei muss eins der folgenden Muster eingehalten werden:

$$(1.\text{Wellenlänge} - 2.\text{Wellenlänge}, \text{Einfallswinkel}) \rightarrow R(\lambda)$$

$$(\text{Wellenlänge}, 1.\text{Einfallswinkel} - 2.\text{Einfallswinkel}) \rightarrow R(\theta)$$

Sollten die Parameter in einem ungültigen Bereich liegen, wird stattdessen eine Fehlermeldung geliefert. Weiteres Nutzen des Bestätigen-Buttons fügt die gewünschten Funktionen in den Graphen hinzu, bis der Nutzer sich entscheidet, diesen mit Zurücksetzen zu leeren. Eine Übersicht zu den Parametern und validen Eingaben befindet sich in Tabelle 1

5 Simulationsergebnisse

In diesem Kapitel werden zwei der Extremen unserer Anwendung beobachtet. Zuerst wird eine Antireflexbeschichtung betrachtet. Sie ist aufgeteilt in die jeweiligen Mate-

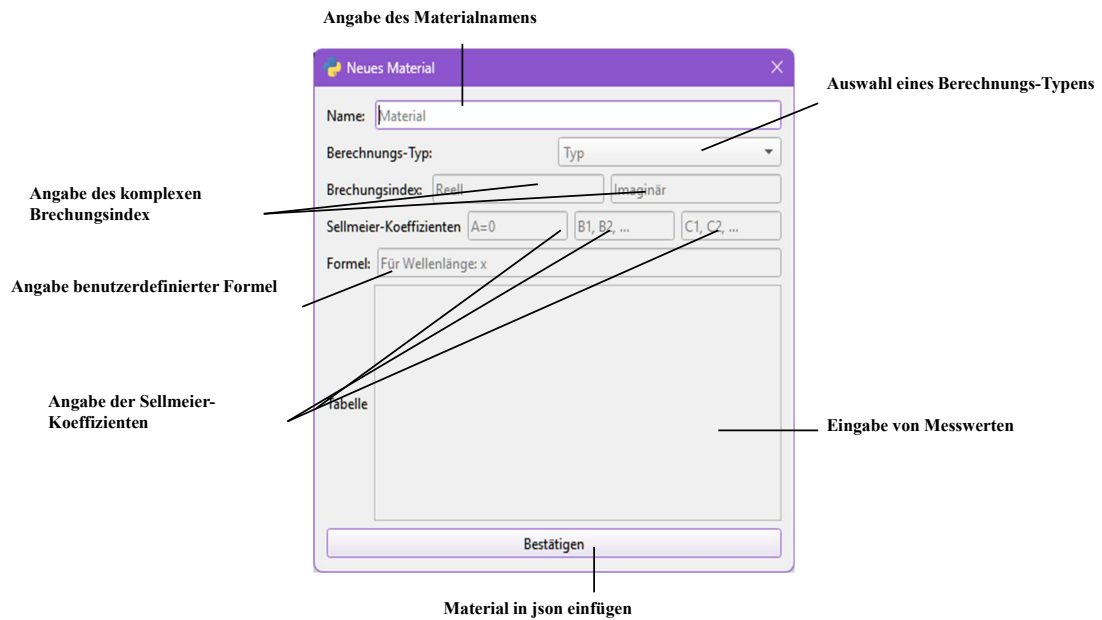


Abbildung 4: Dialogfenster zum Hinzufügen neuer Materialien

Tabelle 1: Beschreibung der Eingabeparameter im Programm

Parameter	Beschreibung
Dicke	Dicke der entsprechenden Schicht in Nanometern
Wellenlängenbereich	Gewünschter Wellenlängenbereich > 0 , der im Graphen geplottet werden soll. Kann auch angegeben werden in der Form: 1.Wellenlänge-2.Wellenlänge
Einfallswinkel	Einfallswinkel in Gradmaß im Bereich $0^\circ - 89^\circ$
Polarisation	Art der Polarisation mit Auswahl zwischen senkrecht und parallel

rialien und ihren individuellen Dicken, welche in Tabelle 2 zu sehen sind und soll im sichtbaren Spektrum simuliert werden. Hierfür nehmen wir einen Wellenlängenbereich von 400 nm bis 800 nm mit einem senkrechten Einfallswinkel. Anschließend betrachten wir den Reflexionsgrad in Abhängigkeit des Einfallswinkels. Das Ergebnis ist in Abbil-

Tabelle 2: Zu simulierendes Schichtsystem

Material	Dicke in nm
Luft	∞
MgF ₂	102,0
TiO ₂	105,0
AL ₂ O ₃	79,0
Glas	∞

dung 5 dargestellt und zeigt keine physikalischen Auffälligkeiten, die auf einen Fehler deuten würden. Die Funktion ist im ausgewählten Bereich stetig, was auf mathematische Stabilität deutet. Zusätzlich kann der Reflexionsgrad mit dem eines Lichtstrahls der von Luft auf bloßes Glas trifft, im Graph verglichen werden, um die Verbesserung der Reflexivität durch die Antireflexbeschichtung zu bestätigen.

Es wird außerdem, wie in Abbildung 6 zu sehen ist, der Reflexionsgrad in Abhängigkeit vom Einfallswinkel bestimmt. Es wird bis etwa 40° Reflexivität von nahezu 0 beobachtet, was bestätigt, dass die Entspiegelung optimal simuliert wird. Ab 60° steigt der Reflexionsgrad steil an, was bei flachen Winkeln zu erwarten ist und dadurch in der Norm liegt.

Die zweite zu simulierende Schicht befindet sich in Tabelle 3. Betrachtet wird hierbei ein Hochreflektor im EUV-Bereich. Es wurde versucht für dieses Schichtsystem bei extremem Ultraviolettlicht optimale Reflexivität zu garantieren. Wichtig ist hierbei, dass hier 2 Schichten mit jeweils einem hohen und einem niedrigen Brechungsindex gepaart sind. Um einen hohen Reflexionsgrad zu erreichen werden meist bis zu 40 bis 60 dieser Schichtpaare aneinander gereiht, es wird sich im Anschluss aber nur auf 15 Paare beschränkt, da dies bereits ein repräsentatives Ergebnis liefert.

Tabelle 3: Zu simulierendes EUV-Schichtsystem

Material	Brechzahl bei 13.5 nm
Mo	$1 - 7,6044 \cdot 10^{-2} - 6,4100 \cdot 10^{-3}i$
Si	$1 - 9,3781 \cdot 10^{-4} - 1,7260 \cdot 10^{-3}i$
SiO ₂	$1 - 0,0218338 - 0,010721i$

Tabelle 4: Schichtdicke für EUV-Schicht

Material	Dicke°
Luft	∞
Mo	2,8nm
Si	4,1nm
SiO ₂	∞

Die gewählten Schichtdicken sind in Tabelle 4 zu sehen. Das Ergebnis in Abbildung 7 ist die gewünschte Funktion für die EUV-Schicht. Der Hochpunkt bei etwa 13.5 nm lässt vermuten, dass die Simulation, genau wie im ersten Simulationsergebnis, realitätsgetreu

ist. In Abbildung 8 ist dasselbe Schichtsystem in Abhängigkeit vom Einfallswinkel in Grad dargestellt, dessen steigender Reflexionsgrad bei 0° und ab 75° ebenfalls eine realistische Simulation vermuten lässt.

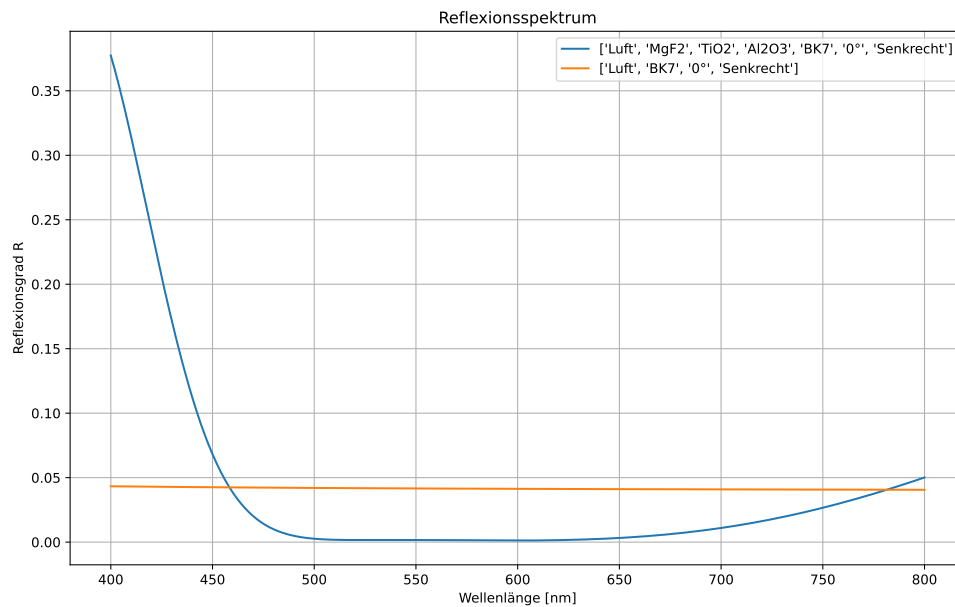


Abbildung 5: Darstellung von $R(\lambda)$ für die Antireflexbeschichtung bei senkrechtem Einfall

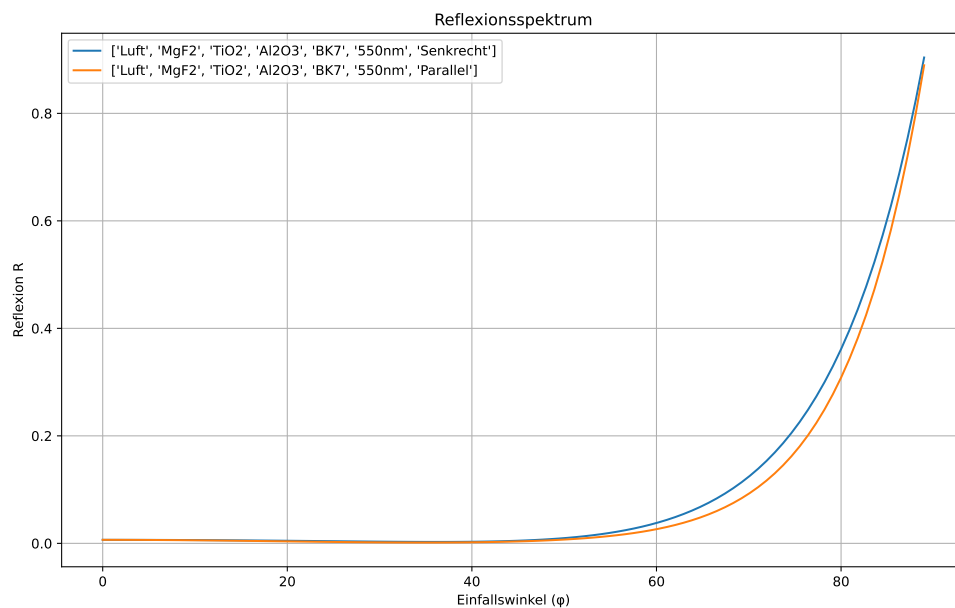


Abbildung 6: Darstellung von $R(\varphi)$ für die Antireflexbeschichtung bei einer Wellenlänge von 550 nm

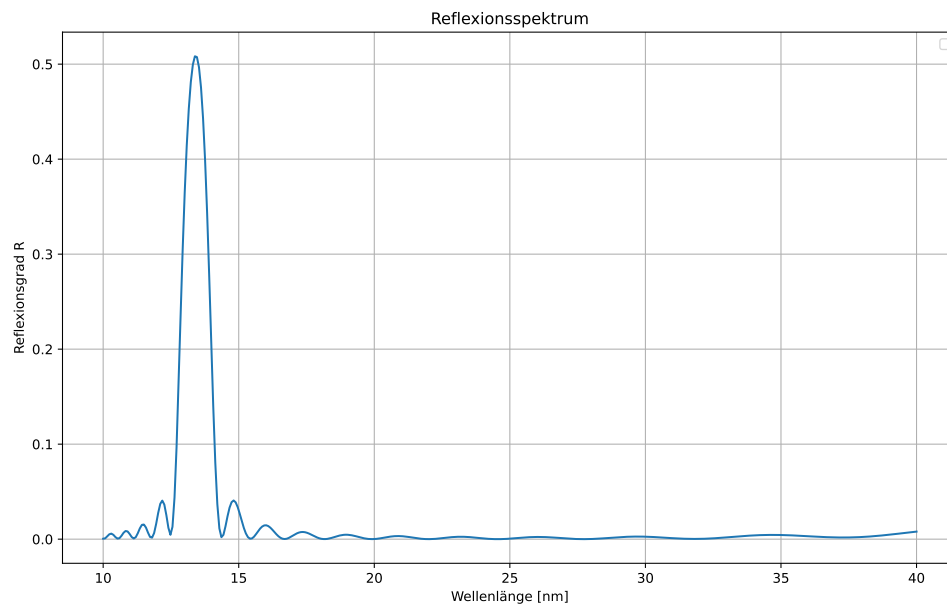


Abbildung 7: Darstellung von $R(\lambda)$ für die EUV-Schicht bei Senkrechtem Einfall

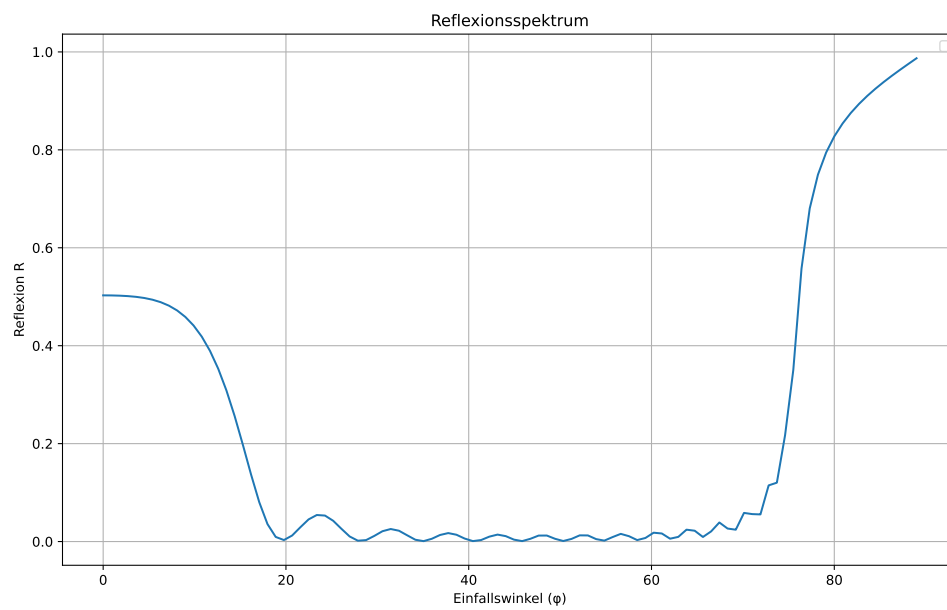


Abbildung 8: Darstellung von $R(\varphi)$ für die EUV-Schicht bei einer Wellenlänge von 13.5 nm

Literatur

- [1] Eugene Hecht. *Optik*. Berlin, Boston: De Gruyter, 2018. ISBN: 9783110526653. DOI: doi : 10 . 1515 / 9783110526653. URL: [https : // doi . org / 10 . 1515 / 9783110526653](https://doi.org/10.1515/9783110526653).
- [2] Petr Křen. “Comment on “Precision refractive index measurements of air, N2, O2, Ar, and CO2 with a frequency comb””. In: *Appl. Opt.* 50.35 (Dez. 2011), S. 6484–6485. DOI: 10 . 1364 / AO . 50 . 006484. URL: [https : // opg . optica . org / ao / abstract . cfm ? URI = ao - 50 - 35 - 6484](https://opg.optica.org/ao/abstract.cfm?URI=ao-50-35-6484).
- [3] H. H. Li. “Refractive index of alkaline earth halides and its wavelength and temperature derivatives”. In: *Journal of Physical and Chemical Reference Data* 9.1 (Jan. 1980), S. 161–290. ISSN: 0047-2689. DOI: 10 . 1063 / 1 . 555616. eprint: [https : // pubs . aip . org / aip / jpr / article - pdf / 9 / 1 / 161 / 11502934 / 161 _ 1 _ online . pdf](https://pubs.aip.org/aip/jpr/article-pdf/9/1/161/11502934/161_1_online.pdf). URL: [https : // doi . org / 10 . 1063 / 1 . 555616](https://doi.org/10.1063/1.555616).
- [4] I. H. Malitson. “Interspecimen Comparison of the Refractive Index of Fused Silica*,†”. In: *J. Opt. Soc. Am.* 55.10 (Okt. 1965), S. 1205–1209. DOI: 10 . 1364 / JOSA . 55 . 001205. URL: [https : // opg . optica . org / abstract . cfm ? URI = josa - 55 - 10 - 1205](https://opg.optica.org/abstract.cfm?URI=josa-55-10-1205).