

Nr. 44

Röntgenreflektometrie

Sara Krieg
sara.krieg@udo.edu

Marek Karzel
marek.karzel@udo.edu

Durchführung: 10.12.2020

Abgabe: 05.01.2021

TU Dortmund – Fakultät Physik

Inhaltsverzeichnis

1	Theorie	3
1.1	Grundlagen	3
1.2	Fresnelsche Formeln	3
1.3	Mehrschichtsysteme	4
1.4	Rauigkeit	5
1.5	Geometriefaktor und Geometriewinkel	5
2	Aufbau	6
3	Durchführung	8
3.1	Justage	8
3.2	Messung	8
4	Auswertung	9
4.1	Vorbereitende Berechnungen	9
5	Scananwendung zur Vermessung eines mit Polystyrol beschichteten Silizium-Wafers	11
6	Diskussion	12
7	Anhang	13
	Literatur	18

1 Theorie

Mithilfe der Röntgenreflektometrie sollen Dichte, Rauigkeit und Schichtdicke eines Polystyrolfilms bestimmt werden.

1.1 Grundlagen

Brechung findet statt, wenn eine elektromagnetische Welle von einem Medium mit Brechungsindex n_1 in ein anderes Medium mit Brechungsindex n_2 eintritt. Dabei muss gelten, dass $n_1 \neq n_2$. In diesem Versuch wird Röntgenstrahlung mit einer Wellenlänge λ im Bereich $0,1 \text{ \AA} - 10 \text{ \AA}$ betrachtet.

Für den Brechungsindex gilt

$$n = 1 - \delta + i\beta,$$

wobei δ ein Korrekturterm und β die Absorption ist. Für Röntgenstrahlung gilt, dass der Brechungsindex kleiner als eins ist.

Mit dem Snelliusschen Berechnungsgesetz

$$\frac{n_1}{n_2} = \frac{\cos(\alpha_2)}{\cos(\alpha_1)}$$

und der Annahme, dass die Grenzfläche der Medien eine homogene Ebene ist, ergibt sich ein kritischer Winkel α_C , bei dem eine Totalreflexion auftritt. Unter Vernachlässigung der Absorption folgt für kleine Winkel näherungsweise

$$\alpha_C \approx \sqrt{2\delta} = \lambda \sqrt{\frac{r_e \rho}{\pi}}.$$

Dabei ist r_e der klassische Elektronenradius und ρ die Elektronendichte des betrachteten Materials.

1.2 Fresnelsche Formeln

Bei der Betrachtung von Reflexion und Transmission elektromagnetischer Wellen muss im Allgemeinen die Polarisierung dieser berücksichtigt werden. Dies tun die Fresnelschen Formeln. Für s-Polarisation des Lichtes ergeben sich die Koeffizienten zu

$$r = \frac{n_1 \cos(\alpha_1) - n_2 \cos(\alpha_2)}{n_1 \cos(\alpha_1) + n_2 \cos(\alpha_2)},$$
$$t = \frac{2n_1}{n_1 \cos(\alpha_1) + n_2 \cos(\alpha_2)}.$$

Dabei ist hier eine Unterscheidung zwischen p - und s -Polarisation nicht notwendig, weil $n_1 \approx n_2$ gilt.

Die Fresnelreflektivität für Röntgenstrahlung ist für $\alpha_i > 3\alpha_C$ näherungsweise gegeben als

$$R_f = \frac{a_C}{2\alpha_i}.$$

1.3 Mehrschichtsysteme

In diesem Versuch wird ein Polystyrolfilm auf einem Siliziumsubstrat betrachtet, weswegen der Umgang mit Mehrschichtsystemen erläutert werden muss. Ein Beispiel für die Reflektivität eines solchen Systems ist in Abbildung 1 zu sehen.

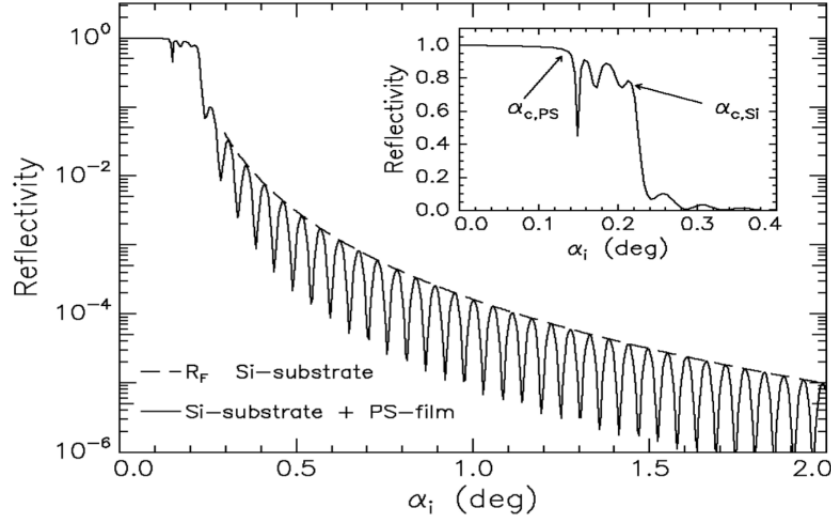


Abbildung 1: Beispielhafte Reflektivität eines Mehrschichtsystems aufgetragen gegen den Einfallswinkel α_i . [2]

Dabei sind in dem vergrößerten Ausschnitt zwei Totalreflexionen zu erkennen. Diese sind dem Siliziumsubstrat und dem Polystyrolfilm zuzuordnen. Die beobachteten Oszillationen beim Abfall der Reflektivität treten aufgrund von Interferenzeffekten an der Oberfläche auf. Durch diese Oszillationen sind demnach Rückschlüsse auf den Schichtabstand möglich, da bei einer destruktiven Interferenz ein Gangunterschied von einem ungeraden Vielfachen von $\frac{\lambda}{2}$ vorliegen muss. Damit folgt der Zusammenhang

$$d = \frac{2\pi}{\delta q_z} = \frac{\lambda}{2\delta\alpha_1},$$

wobei $\vec{q} = \vec{k}_2 - \vec{k}_1$ und $q_2 = 2k \sin(\alpha_1)$ gilt.

Wenn das betrachtete System $N + 1$ Schichten hat, kann die Reflektivität mithilfe des rekursiven Parratt-Algorithmus berechnet werden. Ein solches System ist in Abbildung 2 dargestellt.

Bei dem Parratt-Algorithmus wird die Annahme getroffen, dass die unterste Schicht unendlich dick ist. An dieser kann demnach keine Transmission stattfinden. Mathematisch wird der Algorithmus beschrieben durch

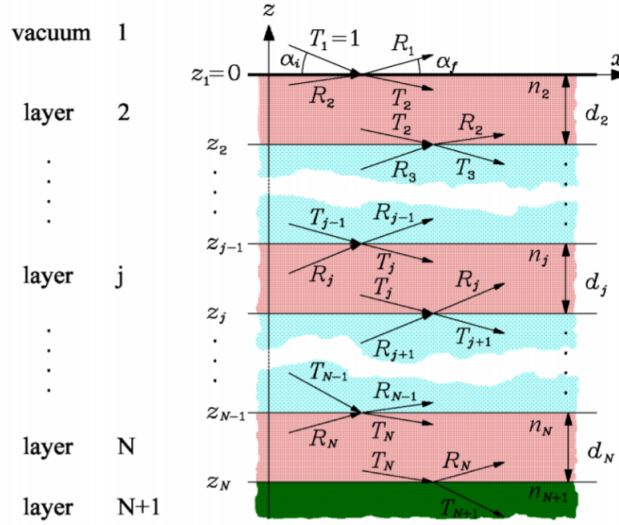


Abbildung 2: Beispielhafte Darstellung eines Mehrschichtsystems mit $N + 1$ Schichten.[2]

$$X_j = \frac{R_j}{T_j} = \exp(-2ik_{z,j}z_j) \cdot \frac{r_{j,j+1} + X_{j+1} \exp(2ik_{z,j+1}z_j)}{1 + r_{j,j+1}X_{j+1} \exp(2ik_{z,j+1}z_j)}.$$

Dabei beschreibt $r_{j,j+1}$ die Fresnelreflektivität an der j -ten Grenzfläche.

Mit der Startbedingung, dass an der untersten Schicht keine Reflexion stattfindet $R_{N+1} = 0$, können die Verhältnisse der reflektierten und transmittierten Anteile rekursiv von unten nach oben berechnet werden.

1.4 Rauigkeit

Bisher wurde angenommen, dass die Oberfläche der Grenzfläche perfekt glatt ist. Dies ist in der Realität nicht der Fall, weswegen diese Unperfektion in der Berechnung der Reflektivität berücksichtigt werden muss. Dies geschieht über eine Modifikation der Fresnelkoeffizienten:

$$\begin{aligned} \tilde{r}_{j,j+1} &= r_{j,j+1} \exp(-2k_{z,j}k_{z,j+1}\sigma_j^2), \\ \tilde{t}_{j,j+1} &= t_{j,j+1} \exp\left(\left(k_{z,j}-k_{z,j+1}\right)^2 \cdot \frac{\sigma_j^2}{2}\right). \end{aligned}$$

1.5 Geometriefaktor und Geometriewinkel

In Abbildung 3 ist zu sehen, dass der verwendete Strahl eine größere Fläche überstreicht als die Probenoberfläche.

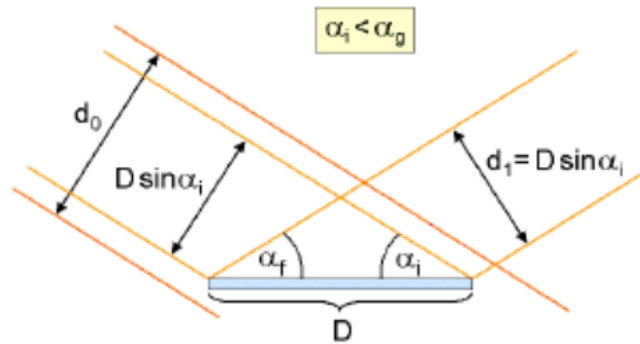


Abbildung 3: Veranschaulichung des Geometriewinkels.[1]

Aus diesem Grund kann nur ein Teil der Intensität I reflektiert und später detektiert werden. Dies wird durch den Geometriefaktor G berücksichtigt:

$$G = \frac{D \sin(\alpha_i)}{d_0} \quad \text{mit} \quad \alpha_i < \alpha_g, \quad (1)$$

$$G = 1 \quad \text{mit} \quad \alpha_i > \alpha_g. \quad (2)$$

Dabei ist $D \sin(\alpha_i)$ die Strahlbreite, die die Probenoberfläche trifft und d_0 die gesamte Strahlbreite.

2 Aufbau

Der Versuch wird mit einem D8-Diffraktometer durchgeführt, welches in Abbildung 4 zu sehen ist.

Dieses besteht aus einer Röntgenröhre mit Kupferanode, welche mit einer Spannung von 40 kV und 40 mA betrieben wird. Die austretende Strahlung trifft auf einen Göbelspiegel, der die Strahlung bündelt und monochromatisiert. Der dann resultierende Strahl besitzt eine Wellenlänge von $\lambda = 1,54 \text{ \AA}$. Die verwendete Röntgenröhre ist in Abbildung 5 zu sehen.



Abbildung 4: Das verwendete D8-Diffraktometer.[1]

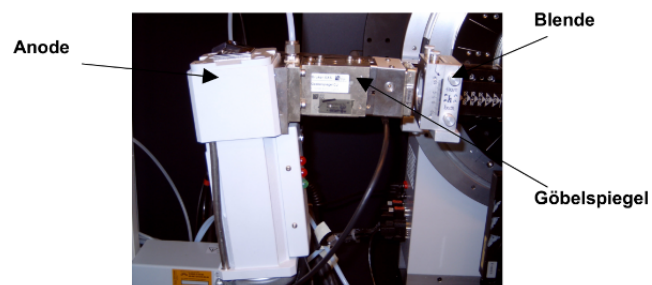


Abbildung 5: Röntgenröhre des D8-Diffraktometers.[2]

3 Durchführung

Vor der eigentlichen Messung muss der Aufbau zunächst justiert werden.

3.1 Justage

Für die Justage des Aufbaus sind sechs Schritte notwendig.

Detektorscan:

Zunächst wird die Probe aus dem Strahlengang entfernt. Es werden der Detektor und die Röntgenröhre auf Position 0° gefahren. Um nun die tatsächliche Nulllage des Detektors zu finden, wird nun seine Lage variiert, bis die Intensität des Primärstrahls das Maximum durchläuft. Diese Position ist die neue Nullposition.

Erster Z-Scan:

In diesem Schritt wird die Probenjustage angepasst. Um dabei die z -Position der Probe zu variieren, wird diese zunächst wieder in den Strahlengang geschoben. Dann wird die Intensität I gemessen. Die Position der Probe wird variiert, bis I auf den Wert $I = \frac{1}{2}I_{\max}$ gesunken ist. Der z -Wert wird notiert und die Motoren in die entsprechende Position gebracht.

Erster Rockingscan:

Nun werden der Detektor und die Röntgenröhre um die Probe bewegt. Dabei wird ein konstanter Winkel zwischen der Probe und dem Detektor mit $2\theta = 0^\circ$ beibehalten. Die Drehung erfolgt dabei in einem Winkelbereich von -1° bis 1° . Aus dem resultierenden Intensitätsverlauf wird dann das Maximum abgelesen und für die weiteren Schritte verwendet.

Zweiter Z-Scan:

Nach dem Rockingscan befindet sich die Probe nun nicht mehr in der gewollten Position. Aus diesem Grund wird ein erneuter Z-Scan durchgeführt.

Zweiter Rockingscan:

Um eine Erhöhung der Präzession zu erreichen, wird ein erneuter Rockingscan mit $2\theta = 0,3^\circ$ durchgeführt. Es wird ein Scanbereich von $-0,5\text{ mm}$ bis $0,5\text{ mm}$ gewählt. Danach wird erneut das Maximum abgelesen und die Motoren entsprechend ausgerichtet.

Dritter Rockingscan:

Abschließend wird ein letzter Rockingscan mit $2\theta = 1^\circ$ durchgeführt. Der Scanbereich wird als $0,45^\circ$ bis $0,55^\circ$ gewählt, da das erwartete Maximum bei $0,5^\circ$ liegt. Das Maximum wird erneut abgelesen und die Motoren entsprechend ausgerichtet.

3.2 Messung

Nun wird ein Reflektivitätsscan durchgeführt. Bei diesem sind der Einfallswinkel α_i und der Winkel zwischen Probe und Detektor α_f gleich. Es wird ein Scanbereich von 0° bis 5°

und eine Schrittweite von $0,05^\circ$ gewählt. Die Messzeit beträgt dabei pro Datenpunkt 5 s. Außerdem wird ein diffuser Scan durchgeführt, bei dem der Anteil der gestreuten Intensität an der Reflektivität bestimmt wird. Dabei gilt hier $\Delta a = |\alpha_i - \alpha_f| = 0,1^\circ$.

4 Auswertung

4.1 Vorbereitende Berechnungen

Der erste Detektorscan wird mittels Funktion `scipy.optimize.curve_fit` aus der Python-Bibliothek SciPy nach der Gaußfunktion

$$f(\alpha_i) = \frac{a}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\alpha_i - \mu)^2}{2\sigma^2}\right) + b \quad (3)$$

approximiert und ist in Abbildung 6 dargestellt. Die Näherungsparameter ergeben sich zu

$$\mu = (-4,1 \pm 0,5) \cdot 10^{-3^\circ},$$

$$\sigma = (45,9 \pm 0,6) \cdot 10^{-3^\circ},$$

$$a = (18,55 \pm 0,22) \cdot 10^4,$$

$$b = (1,6 \pm 0,4) \cdot 10^4.$$

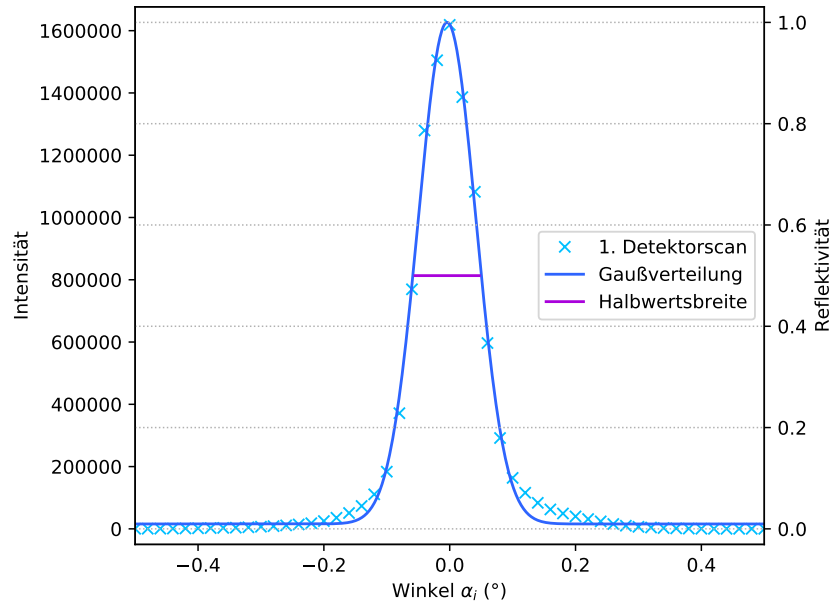


Abbildung 6: Gaußgenäherte Intensität des ersten Detektorscans in Abhängigkeit des Einfallswinkels α_i .

Aus den aufgenommenen Daten und des Maximums der Gaußnäherung ergeben sich die maximalen Intensitäten von

$$I_{\max, \text{detec}} = 1\,618\,610 ,$$

$$I_{\max, \text{gauss}} = 1\,626\,579,26$$

und die Halbwertsbreite der Näherung von

$$d_{1/2} = 0,12^\circ .$$

In Abbildung 7 ist der erste Z-Scan dargestellt. Aus diesem lässt sich anhand der Differenz der Höhen rechts und links der negativen Flanke der Intensität die Strahldicke als etwa

$$d_0 = 0,28 \text{ mm}$$

bestimmen. Zudem übersteigt hier die maximale Intensität die des Detektorscans mit

$$I_{\max, z} = 1\,636\,650 , \quad (4)$$

mit welcher die Reflektivitäten der folgenden Scans normiert werden sollen.

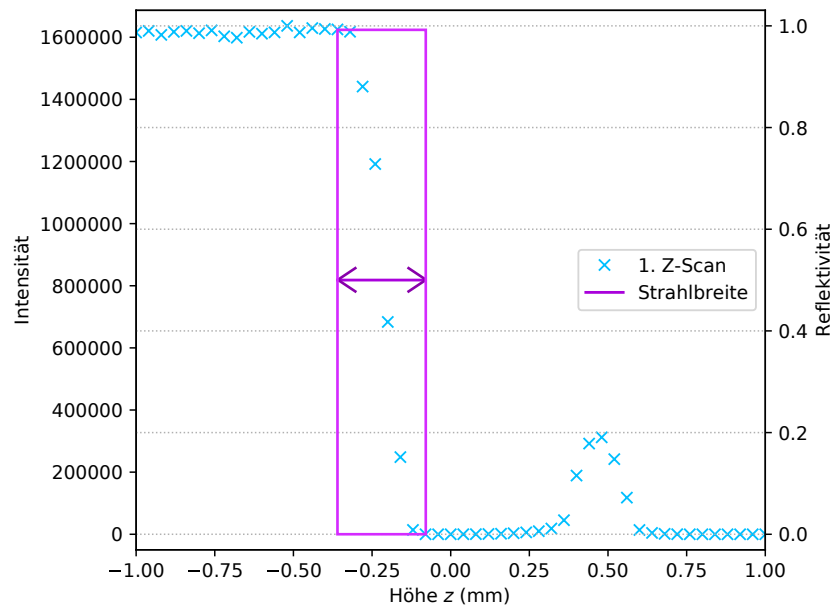


Abbildung 7: Intensität des Z-Scans in Abhängigkeit der Probenhöhe z .

Aus dem in Abbildung 7 dargestellten ersten Rockingscans können links und rechts des Intensitätspeaks die beiden Geometriewinkel $\alpha_{g,i}$ abgelesen werden, deren Intensitäten gerade nicht verschwindend sind. Zusammen mit dem gemitteltem Winkel α_g ergeben sich diese zu

$$\begin{aligned}\alpha_{g,1} &= -0,84^\circ, \\ \alpha_{g,2} &= 0,64^\circ, \\ \alpha_g &= 1/2|\alpha_{g,1} - \alpha_{g,2}| = 0,74^\circ.\end{aligned}$$

Alternativ kann der Geometriewinkel auch über Strahlbreite $d_0 = 0,28\text{ mm}$ und die Probendicke $D = 20\text{ mm}$ bestimmt werden zu

$$\alpha'_g = \arcsin(d_0/D) \approx 0,8^\circ. \quad (5)$$

5 Scananwendung zur Vermessung eines mit Polystyrol beschichteten Silizium-Wafers

Für die eigentliche Messung wurden ein Reflektivitäts- und ein Diffuser Scan angewandt und deren Daten aufbereitet, sodass sich die in Abbildung 8 dargestellten Ergebnisse zeigen. Zur Bestimmung der Reflektivitäten werden die Intensitäten auf die fünffache maximale Intensität des Z-Scans (4) normiert. Damit wird berücksichtigt, dass die Messdauer im Vergleich zu den Justagemessungen von 1 s auf 5 s erhöht wurde. Weiterhin werden die Daten statt in Winkelabhängigkeit, in Abhängigkeit des Wellenvektorübertrags $q = \frac{4\pi}{\lambda} \cdot \sin\left(\frac{\pi}{180}\alpha_i\right)$ dargestellt.

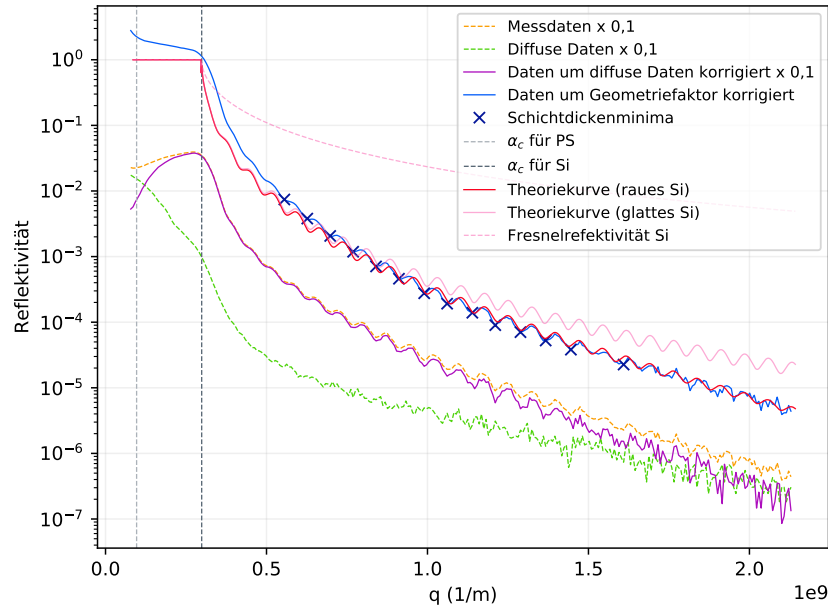


Abbildung 8: Vermessungsdaten der Reflektivität PS-Si-Wafers in Abhängigkeit des Wellenvektorübertrages q . Zur Übersichtlichkeit sind einige Messdaten auf der logarithmischen Skala um 0,1 tiefer geschoben.

Schließlich werden zur Eliminierung von Rückstreuungseffekten im Schichtsystem, die Messdaten des Diffusen Scans von den eigentlichen abgezogen. Es erfolgt eine weitere Korrektur um den Geometriefaktor, welcher sich aus der Strahlbreite d_0 und dem Geometriewinkel α_g nach den Gleichungen (1) bestimmen lässt. Diese verhindert die Unterrepräsentation sehr kleiner Winkel, für die nicht die gesamte Strahlbreite am Wafer reflektiert wird.

Anschließend werden die Positionen der Minima der Kiessig-Oszillationen mit Hilfe der Funktion `scipy.signal.find_peaks` aus der Python-Bibliothek `SkiPy` bestimmt. Aus deren Abständen kann die Schichtdicke des Polysterol(PS)-Films auf dem Silizium(Si)-Wafer bestimmt werden

$$d_{\text{PS}} = (882,43 \pm 0,04) \cdot 10^{-10} \text{ m} .$$

Weiterhin wird mithilfe des Parrat-Algorithmus eine Theoriekurve an die korrigierten Messdaten durch Variation einiger Parameter gefittet. Diese Parameter beinhalten Brechungsindizes n der drei Materialien, die Rauigkeiten σ der beiden Grenzflächen und die Schichtdicke d_{PS} .

$$\begin{aligned} n_{\text{Luft}} &= 1 , \\ n_{\text{PS}} &= 1 - 0,7 \cdot 10^{-6} , \\ n_{\text{Si}} &= 1 - 6,7 \cdot 10^{-6} , \\ \sigma_{\text{Luft-PS}} &= 7,9 \cdot 10^{-10} \text{ m} , \\ \sigma_{\text{PS-Si}} &= 5,7 \cdot 10^{-10} \text{ m} , \\ d_{\text{PS}} &= 855 \cdot 10^{-10} \text{ m} . \end{aligned}$$

Die kritischen Winkel der Totalreflexion der beiden Materialien lassen sich aus den Korrekturen δ der Brechungsindizes $n = 1 - \delta$ berechnen zu

$$\begin{aligned} \alpha_{\text{c,PS}} &= 0,068^\circ , \\ \alpha_{\text{c,Si}} &= 0,210^\circ . \end{aligned}$$

Zuletzt werden, durch den Parrat-Algorithmus berechnet, eine Theoriekurve der Reflektivität mit ideal glatten Grenzflächen und die Fresnelreflektivität einer Siliziumoberfläche an der Luft eingezeichnet.

6 Diskussion

Zuerst lässt sich feststellen, dass die Theoriekurven des Wafers beide Kiessig-Oszillationen im Verlauf aufzeigen. Sie gleichen sich im Verlauf stark, spalten sich jedoch für einen größer werdenden Wellenvektorübertrag q . Die Reflektivität des Wafers mit theoretisch glatten Oberflächen ist dabei größer als mit realistisch rauhen Oberflächen. Außerdem liegen beide Theoriekurven signifikant unter der Fresnelreflektivität einer einzelnen Siliziumoberfläche. Diese Zusammenhänge sind physikalisch logisch. Glatte Oberflächen reflektieren besser als raue. Für die nach verschiedenen Methoden bestimmten Geometriewinkel

$$\alpha_g = 0,74^\circ \qquad \alpha'_g = 0,8^\circ$$

ergibt sich eine relative Abweichung von 7,5 %. Die durch die Oszillationsminima abgeschätzte Schichtdicke

$$d_{\text{PS,est}} = (882,43 \pm 0,04) \cdot 10^{-10} \text{ m} .$$

weicht um etwa 3,2 % von der durch die Theoriekurve bestimmte Schichtdicke

$$d_{\text{PS,theo}} = 855 \cdot 10^{-10} \text{ m}$$

ab. Diese geringe Abweichung ist dadurch zu erklären, dass die Theoriekurve nicht optimal an die Messdaten angenähert werden kann.

Die Korrekturterme konnten zu

$$\begin{aligned} \delta_{\text{PS}} &= 0,7 \cdot 10^{-6} , \\ \delta_{\text{Si}} &= 6,7 \cdot 10^{-6} . \end{aligned}$$

bestimmt werden und weichen um 80 % und 11,8 % von den Literaturwerten [3]

$$\begin{aligned} \delta_{\text{PS}} &= 0,7 \cdot 10^{-6} , \\ \delta_{\text{Si}} &= 6,7 \cdot 10^{-6} \end{aligned}$$

ab. Diese Abweichungen ergeben sich möglicherweise aus einer ungenauen Bestimmung der Parameter. Die kritischen Winkel

$$\begin{aligned} \alpha_{\text{c,PS}} &= 0,068^\circ , \\ \alpha_{\text{c,Si}} &= 0,210^\circ \end{aligned}$$

weichen um 54,7 % und 4,5 % von den Literaturwerten [3]

$$\begin{aligned} \alpha_{\text{c,PS,lit}} &= 0,15^\circ , \\ \alpha_{\text{c,Si,lit}} &= 0,22^\circ \end{aligned}$$

ab. Hier liegt die Abweichung für Polystyrol erneut deutlich höher als für Silizium. Insgesamt halten sich die Abweichungen des Versuches allerdings im Rahmen und liefert nah an der Theorie liegende Werte.

7 Anhang

\caption{Programmcode für zweiten Auswertungsteil}

```
import numpy as np
import matplotlib.pyplot as plt
from scipy import optimize
from scipy.optimize import curve_fit
from scipy import signal as sig
from uncertainties import ufloat
import uncertainties.unumpy as unp
```

```

#Versuchsgrößen
l = 1.54e-10 # Wellenlänge
ai = np.pi/180 * np.arange(6e-2, 1.505, 5e-4)
#Winkel -> -xWerte der Theoriekurve
k = 2*np.pi / l #Wellenvektor
qz = 2*k * np.sin(ai)
#Wellenvektorübertrag -> y-Werte der Theoriekurve

#Parameter des Parratt-Algorithmus

#Brechungsindizes
d1 = 0.7e-6 #Polysterol -> Amplitude vergrößert + negativer Offset
d2 = 6.7e-6 #Silizium -> Amplitude verkleinert + positiver Offset
#
n1 = 1 #Luft
n2 = 1 - d1 #Polysterol
n3 = 1 - d2 #Silizium

#Rauigkeit
s1 = 7.9e-10
#Polysterol -> Amplitude verkleinert bei hinteren Oszillationen
s2 = 5.7e-10
#Silizium -> Senkung des Kurvenendes und
#Amplitudenverkleinerung der Oszillationen

#Schichtdicke
z = 855e-10 # verkleinert Oszillationswellenlänge

#####

#Messdaten
x, y = np.genfromtxt('data/reflektivitätsscan.UXD', unpack=True)
y = y / (5 * 1636650)
#Normierung auf die 5fache maximale Intensität des Z-Scans

def plotten(phi, I):
    fig, ax1 = plt.subplots()
    #
    q = 2*k * np.sin(np.pi/180 * phi) #Winkel -> Wellenvektorübertrag
    phi_d, I_d = np.genfromtxt('data/difscan.UXD', unpack=True)
    #Messdaten des diffusen Scans
    I_d = I_d / 1636650 #Normierung
    q_d = 2*k * np.sin(np.pi/180 * phi_d)

```

```

#
I_d_k = diff_k(I, I_d) #Korrektur der Messdaten um Diffusen Scan
I_res = geofaktor() #Korrektur der Messdaten um Geometriefaktor
#
q_min , I_min , peak_array = min(I_res , q)
#findet Oszillationsminima
d_layer = layer(q, peak_array) #Bestimmt Schichtdicke
print(d_layer)
#
u = 40 #Ende des Plateaus
beg = 11 # Anfang sinnvoller Messdaten
end = 300 # Ende sinnvoller Messdaten
#
unkorr = I[beg:end] * 1e-1
#Übersichtshalber werden die Daten nach unten verschoben
diff_data = I_d[beg:end] * 1e-1
#Plots
ax1.plot(q[beg:end], unkorr, linestyle='dashed', linewidth=0.7,
color='#FB9D02', label='Messdaten_x_0,1')

ax1.plot(q_d[beg:end], diff_data, linestyle='dashed',
linewidth=0.7, color='#4DD30A', label='Diffuse_Daten_x_0,1')
ax1.plot(q_d[beg:end], I_d_k[beg:end], linewidth=0.7,
color='#AD0EBD', label='Daten_um_diffuse_Daten_korrigiert_x_0,1')
ax1.plot(q[beg:end], I_res[beg:end], linewidth=0.7,
color='#045DF9', label='Daten_um_Geometriefaktor_korrigiert')
ax1.plot(q_min, I_min, 'x', color='#06189A',
label='Schichtdickenminima')
#
phicrit_PS , qcrit_PS = phi_crit(d1)
#Bestimmung der kritischen Winkel
ax1.axvline(x=qcrit_PS, ymin=0, ymax=1, linestyle='dashed',
linewidth = 0.7, color='#ABB2B9', label=r '$\alpha_c$ für PS')
#Vertikales Plotten des Winkels

phicrit_Si , qcrit_Si = phi_crit(d2)
ax1.axvline(x=qcrit_Si, ymin=0, ymax=1, linestyle='dashed',
linewidth = 0.7, color='#566573', label=r '$\alpha_c$ für Si')
#
par = parratt(z) #Aufruf des Parrat-Algorithmus
par2, r13 = parratt2(z)
plt.plot(qz, par, linewidth=0.7, color='#F90421',
label='Theoriekurve_(raues_Si)')
plt.plot(qz, par2, linewidth=0.7, color='#FA58AC',

```

```

label='Theoriekurve_(glattes_Si)', alpha=0.5)
plt.plot(qz, r13, linewidth=0.7, color='#FA58AC',
linestyle='dashed', label='Fresnelrefektivität_Si', alpha=0.5)
#
#
plt.yscale('log')
plt.grid(alpha = 0.2)
ax1.legend(fancybox=True, ncol = 1, loc='upper_right')
ax1.set_xlabel('q_(1/m)')
ax1.set_ylabel('Reflektivität')
fig.tight_layout()
plt.savefig('plot4.pdf')

def diff_k(I, I_d) :
    I_d_k = np.zeros(np.size(I))
    for i in np.arange(np.size(I)):
        I_d_k[i] = 1e-1 * (I[i] - I_d[i])
    return I_d_k

def geofaktor():
    ag = 0.74
    I_res = np.zeros(np.size(x))
    for i in np.arange(np.size(x)):
        if(x[i] <= ag and x[i] > 0):
            I_res[i] = y[i] * np.sin(ag) / np.sin(x[i])
        else :
            I_res[i] = y[i]
    return I_res

def min(I_res, q):
    I_log = np.zeros(np.size(I_res)) #logarithmierte Daten
    for i in np.arange(np.size(I_res)):
        if (I_res[i] != 0):
            I_log[i] = np.log10(I_res[i])
        else :
            I_log[i] = 0
    #
    I_log_n = -1 * I_log #Negatives Vorzeichen
    #-> Minima werden zu Maxima -> Zur Arbeit mit find_peaks
    peakfinder = sig.find_peaks(I_log_n, prominence = 7e-2)
    #

```



```

peak_array = [78, 88, 98, 108, 118, 128, 139, 149,
160, 170, 181, 192, 203, 226] #... Manuell angepasst
#
peak_q = []
peak_I = []
#
for i in peak_array:
    peak_q.append(q[i])
    peak_I.append(I_res[i])
#
return peak_q , peak_I , peak_array

def layer(q, peak_array):
    dist_peaks = [] #Peakabstand
    for i in peak_array:
        if (i < 277):
            dist_peaks.append(q[i+1] - q[i])
        else:
            pass
    mean = np.mean(dist_peaks)
    std = np.std(dist_peaks)
    d = 2*np.pi / ufloat(mean, std)
    return d

def phi_crit(delta) :
    phicrit = 180/np.pi * np.sqrt(2 * delta)
    qcrit = 2*k * np.sin(np.pi/180 * phicrit)
    return (phicrit , qcrit)

def parratt(z):
    kz1 = k * np.sqrt(np.abs(n1**2 - np.cos(ai)**2))
    kz2 = k * np.sqrt(np.abs(n2**2 - np.cos(ai)**2))
    kz3 = k * np.sqrt(np.abs(n3**2 - np.cos(ai)**2))
    #
    r12 = (kz1 - kz2) / (kz1 + kz2) * np.exp(-2 * kz1 * kz2 * s1**2)
    r23 = (kz2 - kz3) / (kz2 + kz3) * np.exp(-2 * kz2 * kz3 * s2**2)
    #
    x2 = np.exp(0 - (kz2 * z) * 2j) * r23
    x1 = (r12 + x2) / (1 + r12 * x2)
    par = np.abs(x1)**2
    #Strecke vor Beginn der Oszillationen auf 1 setzen

```

```

for i in np.arange(np.size(par)):
    if (i <= 296): #296 manuell angepasst
        par[i] = 1
    else:
        pass
return par

def parratt2(z):
    kz1 = k * np.sqrt(np.abs(n1**2 - np.cos(ai)**2))
    kz2 = k * np.sqrt(np.abs(n2**2 - np.cos(ai)**2))
    kz3 = k * np.sqrt(np.abs(n3**2 - np.cos(ai)**2))
    #
    r12 = (kz1 - kz2) / (kz1 + kz2)
    r23 = (kz2 - kz3) / (kz2 + kz3)
    r13 = (kz1 - kz3) / (kz1 + kz3)
    #
    x2 = np.exp(0 - (kz2 * z) * 2j) * r23
    x1 = (r12 + x2) / (1 + r12 * x2)
    par = np.abs(x1)**2
    #Strecke vor Beginn der Oszillationen auf 1 setzen
    for i in np.arange(np.size(par)):
        if (i <= 296): #296 manuell angepasst
            par[i] = 1
            r13[i] = 1
        else:
            pass
    return par, r13

plotten(x, y)

```

Literatur

- [1] TU Dortmund. *V44 Röntgenreflektometrie*. 2019. URL: https://moodle.tu-dortmund.de/pluginfile.php/1278137/mod_resource/content/1/V44.pdf.
- [2] Lehrstuhl für Experimentelle Physik I. *Röntgenreflektometrie Versuch*. 2007. URL: http://e1.physik.tu-dortmund.de/cms/Medienpool/Downloads/Roentgenreflektometrie_Versuch.pdf.
- [3] Metin Tolan. *X-Ray Scattering from Soft Matter Thin Films, Reflectivity of X-Rays from Surfaces - Multiple Interfaces*. Springer-Verlag, 1999.