Fakultät für Physik

Physikalisches Praktikum P2 für Studierende der Physik

Versuch P2-13 (Stand: April 2023)

Raum F1-09

Name: Schwarz Vorname: Felix E-Mail: unuhz@student.kit.edu

Name: Steib Vorname: Lukas E-Mail: lukas.steib@student.kit.edu

Gruppennummer: Mo-21

Betreuer: Lukas Gülzow

Versuch durchgeführt am: 10.07.2023

Beanstandungen:

Testiert am: ________________ Vermerk: ________________

Interferenz

Motivation

Bei diesem Versuch geht es um Beugungs- und Interferenzerscheinungen des Lichts, also um Phänomene deren Deutung auf dem Wellencharakter des Lichts beruht. Interferenzerscheinungen können in der Natur oft beobachtet werden, z.B. als Farben dünner Schichten (bei Seifenblasen oder Ölfilmen auf Wasseroberflächen), als störende Farbschlieren bei der Diaprojektion oder als Beugung an Kanten. Die technischen Anwendungen sind vielfältig. Beispiele dafür sind Interferometer für diverse Messzwecke, Interferenzfilter und holographische Verfahren.

Es gibt verschiedene Möglichkeiten zur Erzeugung monochromatischen Lichts. Abgesehen von Lasern, denen zwei eigene Praktikumsversuche gewidmet sind, werden hier Spektrallampen eingesetzt, deren charakteristische Linienspektren durch die Atomphysik erklärt werden. Eine viel weniger aufwändige (und sehr viel billigere) Lichtquelle stellt die LED dar, die im ersten Teil dieses Versuchs zur Anwendung kommt. Durch den Vergleich einer farbigen LED mit einer weißen oder gar der Na-Spektrallampe können Sie sich ein Bild davon machen, wie „monochromatisch" ein solches Bauteil wirklich ist.

Lernziele

Wir listen im Folgenden die wichtigsten **Lernziele** auf, die wir Ihnen mit dem Versuch **Interferenz** vermitteln möchten:

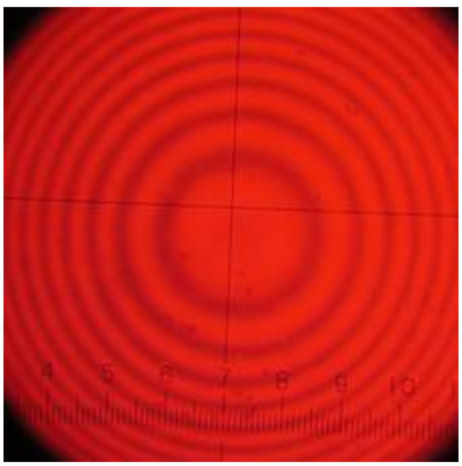
- Sie üben sich in der Verwendung optischer Geräte.
- Sie vertiefen Ihr Verständnis für die Phänomene der Beugung und Interferenz an Spalt und Gitter.
- Sie untersuchen die Spektren von Na und Zn.

Versuchsaufbau

Im Folgenden finden Sie einige Bilder der für diesen Versuch verwendeten Aufbauten:

Stereomikroskope für Beobachtungen mit dem Licht verschiedener LEDs (Aufgabe 1): (Link)

Mit dem LED Licht aufgenommene Newtonsche Ringe (hier mit rotem LED Licht): (Link)



Anordnung zur Bestimmung der Autokollimation (Aufgabe 1.3): (Link)

Spektrometer mit Spektrallampen (Aufgabe 2): (Link)

Für die verschiedenen Versuchsteile stehen Ihnen die folgenden Geräte zur Verfügung:

- Auflichtmikroskope, Objektiv ($1\times$), Okular ($10\times$) mit Fadenkreuz, mit Zoom-Faktoren von 1 bis 4. Seitlich einfallendes Licht wird durch einen teildurchlässigen, planen 45° -Reflektor von oben auf das Objekt gelenkt. Der Objekttisch (Kreuztisch) ist in beiden Richtungen verschiebbar. Die Verschiebungen sind an Millimeterskalen mit einem Nonius ablesbar.
- LED-Leuchten in den folgenden Wellenlängenbereichen: $465\,\mathrm{nm}$ (blau), $520\,\mathrm{nm}$ (grün), $590\,\mathrm{nm}$ (gelb), $625\,\mathrm{nm}$ (rot).
- Symmetrische plankonvexe Linsen mit verschiedenen Brennweiten zwischen $5-35\,\mathrm{cm}$.
- Linsenhalter mit Wechselfassungen.
- Eine Mattscheibe mit undurchsichtigem Muster.
- Ein justierbarer Planspiegel.
- Eine Zeißschiene mit Reitern.
- Eine Glühlampe in einem Lampengehäuse mit zugehörigem Transformator.
- Gitterspektrometer bestehend aus:
  - Einem feststehendes Spaltrohr mit symmetrisch einstellbarem Spalt (S) am äußeren und Achromat (L1, mit Brennweite $f = 18\,\mathrm{cm}$ und Durchmesser $d = 17.5\,\mathrm{mm}$) am

und Achromat (L1, mit Brennweite $f = 18\,\mathrm{cm}$ und Durchmesser $d = 17,5\,\mathrm{mm}$) am inneren Ende. Der Abstand SL1 ist im Bereich $17,5 - 18,5\,\mathrm{cm}$ einstellbar.
    - Einem dreh- und arretierbaren Tisch mit Teilkreis von $0 - 360^\circ$, mit 1°-Teilung. Darauf justierbar bezüglich der Teilung befindet sich ein Halter für eine entsprechendes Gitter G. Der Abstand L1G beträgt $\approx 6\,\mathrm{cm}$.
    - Ein um die Tischachse schwenkbares Fernrohr mit Nonius am Teilkreis des Tisches, achromatischem Objektiv (L2, mit Brennweite $f = 17\,\mathrm{cm}$ und Durchmesser $d = 17,5\,\mathrm{mm}$), Okular (L3, Vergößerung $8\times$, Brennweite $f = 3\,\mathrm{cm}$ und Fadenkreuz; verschiebbar im Fernrohrtubus). Der Abstand GL2 beträgt $\approx 6\,\mathrm{cm}$.
    - Das Fernrohr hat noch eine weitere Feinverstellmöglichkeit bezüglich des Teilkreises. Die zugehörige Mikrometerschraube trägt eine zusätzliche Skala, die mit Hilfe des Nonius geeicht werden kann und die Messung von Winkeldifferenzen ermöglicht, die am Nonius direkt nicht mehr abgelesen werden können.
- Verschiedene Gitter (in Form von Kollodiumfolien o.ä.) zwischen Diagläsern. Die nutzbare Breite beträgt $\approx 36\,\mathrm{mm}$, die nutzbare Höhe $\approx 24\,\mathrm{mm}$ und die Liniendichte $\approx 140\,\mathrm{mm}^{-1}$ und $\approx 600\,\mathrm{mm}^{-1}$.
- Spiegel (passend zum Gitterhalter).
- Na und Zn Spektrallampen. Dabei handelt es sich um Wechselstrombetriebene Niederdruck-Gasentladungslampen mit nur mäßiger Erwärmung und beidseitigen Glühkathoden. Die Na-Lampe funktioniert (bis auf die fehlende Leuchtschicht und die andersartige Füllung) wie eine übliche Leuchtstoffröhre. Der Glimm-Bimetall-Zünder ist in den Lampenkolben integriert. Die Vorschaltdrossel befindet sich in einem Universal-Vorschaltgerät. Sehen Sie sich die Lampe ruhig genauer an und diskutieren Sie ihre funktionsweise mit dem/der Tutor:in, falls Ihnen noch nicht bekannt sein sollte, wie das Zünden und Brennen einer Leuchtstoffröhre funktioniert. Der Mittelwert der Wellenlängen der beiden gelben Na-Linien beträgt $\langle\lambda\rangle = 589,3\,\mathrm{nm}$. Die Zn-Lampe ist anders aufgebaut. Sie enthält temperaturabhängige Widerstände.

Durchführung

Vorstudie

Im Praktikum geht es auch um das Kennenlernen von unterschiedlichen Geräten als Handwerkszeug eines Physikers. Obwohl ein schlichtes, einäugiges Gerät für die Vermessung der Newtonschen Ringe ebenso gut geeignet wäre, steht Ihnen für den ersten Versuchsteil ein Stereo-Zoom-Mikroskop zur Verfügung.

- Machen Sie sich mit diesem Gerät vertraut.
- Überzeugen Sie sich durch Zukneifen je eines Auges, das beide Augen ein leicht unterschiedliches, scharfes Bild des zu untersuchenden Objekts sehen.

Aufgabe 1: Newtonsche Ringe

Aufgabe 1.1

Bestimmen Sie den Krümmungsradius R einer symmetrischen, sphärischen, plankonvexen Linse aus der Beobachtung der Newtonschen Ringe unter dem Mikroskop. Gehen Sie dabei wie folgt vor:

Auf dem verschiebbaren Objekttisch des Mikroskops liegt ein planer Objektträger und darauf die Linse. Als Auflichtquelle dient eine einfarbige LED, dere Licht von vorn über einen Strahlteiler eingekoppelt wird. Reflexionen gibt es unter anderem an der unteren Linsenfläche (Glas-Luft-Übergang) und an der oberen Objektträgerfläche (Luft-Glas-Übergang). Je nach Länge des zusätzlichen Lichtweges $2d$ ergibt sich konstruktive oder destruktive Interferenz. Bei der Herleitung des Zusammenhanges zwischen Durchmesser $2r_k$ des k-ten dunklen Newtonringes, Wellenlänge λ des benutzten Lichts, Brechungsindex $n_\mathrm{L} \approx 1$ der Luft und R müssen Sie ein zusätzlichen Phasensprung um π bei der Reflexion am optisch dichteren Medium berücksichtigen.

- Verifizieren Sie die Formel

$$\frac{r_k^2}{R} = \frac{k\,\lambda}{n_\mathrm{L}}.$$

- Bestimmen Sie R aus der Steigung einer Regressionsgeraden durch mehrere Messpunkte (in manchen Fällen können Sie Newtonsche Ringe bis zur Ordnung $k = 30$ beobachten; oft kann jedoch nur bis $k = 20$ gemessen werden). Bei der Anpassung der Geraden können Sie nicht à priori davon ausgehen, dass es sich um eine Ursprungsgerade handelt. Es könnte z.B. Staub zwischen Linse und Objektträger geraten sein, oder die Linse könnte in ihrem Scheitel abgenutzt sein.
- Führen Sie die Messung erst mit der gelben, dann mit der blauen LED durch.
- Diskutieren Sie für Ihre Auswertung die folgenden Fragen:
  - Wieso spielen die übrigen Reflexionen keine Rolle für das Auftreten von Interferenzerscheinungen?
  - Welchen wesentlichen Nachteil hätte eine Durchlicht- im Gegensatz zu einer Auflichtbeobachtung?

Lösung:

Es werden die Positionen der Newtonringe gemessen. Dann wird mit einem linearen Fit die Formel für die Newtonringe verifiziert.

Die benutzten Lichtquellen haben kein diskretes Spektrum, sondern es ist gaußverteilt. Deshalb wird eine Standardabweichung von $0.1nm$ angenommen.

Die Newtonringe treten auf, wenn Licht auf eine dünne Schicht oder einen dünnen Film fällt und teilweise reflektiert und teilweise transmittiert wird. Das reflektierte und das transmittierte Licht interferieren miteinander und erzeugen ein Interferenzmuster in Form von konzentrischen Ringen.

Die gegebene Formel

$$\frac{r_k^2}{R} = \frac{k\,\lambda}{n_\mathrm{L}}$$

beschreibt das Verhältnis zwischen dem Quadrat des Ringradius (r_k) und dem Abstand zur Lichtquelle (R) in Bezug auf die Wellenlänge des Lichts (λ), die Kreisordnung (k) und den Brechungsindex des umgebenden Mediums (n_L).

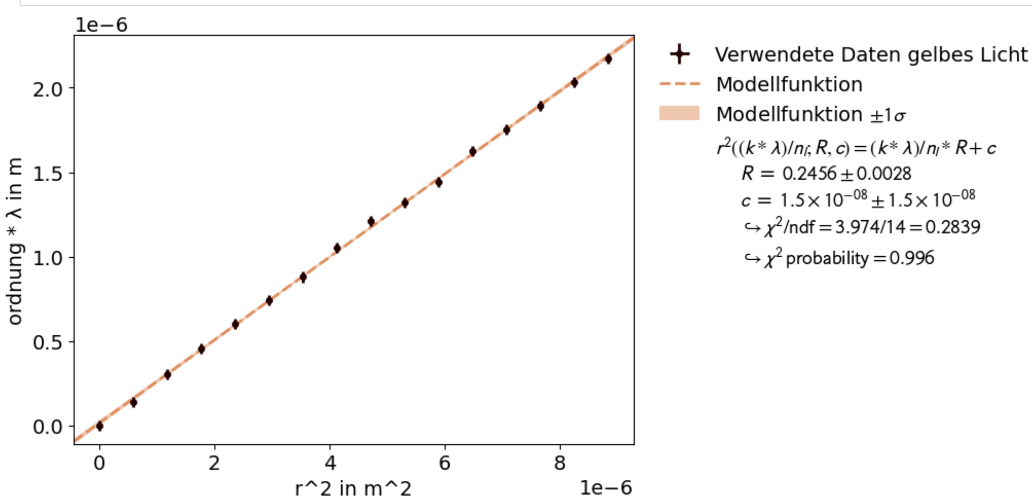
Die Radien der Newtonschen Ringe werden durch Vermessen ihrer Position sowohl auf der einen als auch auf der anderen Seite des Mittelpunkts bestimmt. Es wird hierbei ein Fehler von $0.7 \cdot 0.00025m$ auf das Ablesen der Minima angenommen, da die Minima nicht gut abzulesen waren.

In [ ]:
```
import numpy as np
import matplotlib.pyplot as plt
from PhyPraKit import readPicoScope, resample
from scipy import signal
from scipy import interpolate
import sys
from kafe2 import  XYContainer, Fit, XYFit, Plot, ContoursProfiler
from uncertainties import ufloat, unumpy
from math import log10, floor
import scipy.integrate as integrate
import pandas as pd
import csv
```

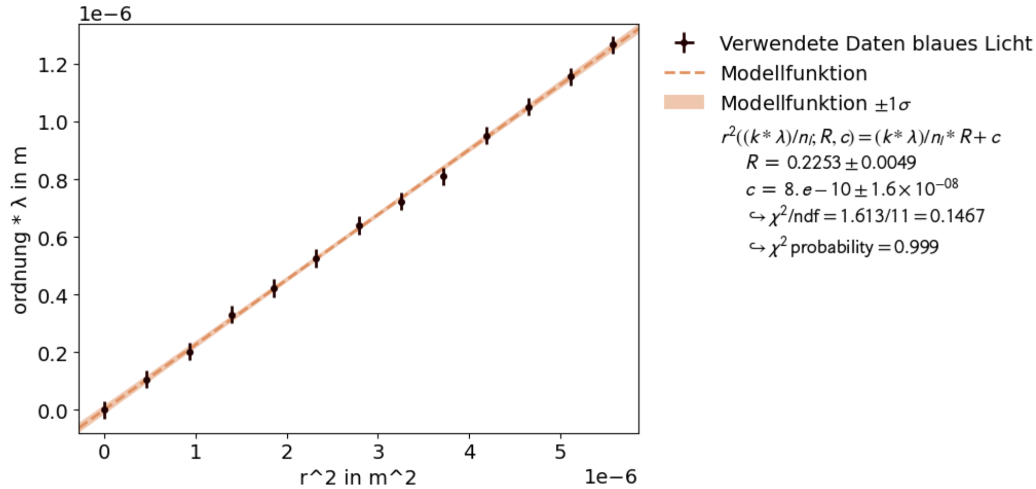
In [ ]:

```
lambda_gelb = ufloat(590*10**(-9),0.1*10**(-9)) #gelbes Licht in m
lambda_gelb_wahr=unumpy.nominal_values(lambda_gelb)
lambda_gelb_fehler=unumpy.std_devs(lambda_gelb)

lambda_blau = ufloat(465*10**(-9),0.1*10**(-9)) #blaue Licht in m
lambda_blau_wahr=unumpy.nominal_values(lambda_blau)
lambda_blau_fehler=unumpy.std_devs(lambda_blau)

ordnung_gelb=np.array([0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15]) #gemessene Ordnung
radius_gelb = unumpy.uarray(np.array([0,1.5,2.2,2.7,3.1,3.45,3.75,4.1,4.4,4.6,4.8,5
radius_gelb_wahr=unumpy.nominal_values(radius_gelb)
radius_gelb_fehler=unumpy.std_devs(radius_gelb)

ordnung_blau=np.array([0,1,2,3,4,5,6,7,8,9,10,11,12]) #gemessene Ordnung
radius_blau = unumpy.uarray(np.array([0,1.3,1.8,2.3,2.6,2.9,3.2,3.4,3.6,3.9,4.1,4.3
radius_blau_wahr=unumpy.nominal_values(radius_blau)
radius_blau_fehler=unumpy.std_devs(radius_blau)

#Gelbes Licht
x= ordnung_gelb*lambda_gelb_wahr
y= radius_gelb_wahr**2
x_error = ordnung_gelb*lambda_gelb_fehler
y_error = radius_gelb_fehler**2
xy_data = XYContainer(x,y)
def linear(x,a=0.0005,b=0):
    return x*a+b
line_fit=Fit(data=xy_data,model_function=linear)
line_fit.add_error(axis='x', err_val=x_error,relative=False)  # add the x-error to
line_fit.add_error(axis='y', err_val=y_error,relative=False)

line_fit.do_fit()
R_gelb = ufloat(line_fit.parameter_values[0],line_fit.parameter_errors[0])
R_gelb_wahr=unumpy.nominal_values(R_gelb)
R_gelb_fehler=unumpy.std_devs(R_gelb)
c_gelb = ufloat(line_fit.parameter_values[1],line_fit.parameter_errors[1])
c_gelb_wahr=unumpy.nominal_values(c_gelb)
c_gelb_fehler=unumpy.std_devs(c_gelb)
line_fit.assign_parameter_latex_names(x='(k * λ) / n_l', a='R',b='c')
line_fit.assign_model_function_latex_name('r^2')
line_fit.assign_model_function_latex_expression(' {x} * {a} + {b}')

line_fit.data_container.label = "Verwendete Daten gelbes Licht"
line_fit.model_label = "Modellfunktion"


line_fit.data_container.axis_labels = ['r^2 in m^2',"ordnung * λ in m"]
plot = Plot(fit_objects= line_fit)

plot.plot()
plt.show()
print('Aus dem Fit folgt R = ',R_gelb,' in m und c = ',c_gelb)


#Blau Licht
x= ordnung_blau*lambda_blau_wahr
y= radius_blau_wahr**2
```

```python
x_error = ordnung_blau*lambda_blau_fehler
y_error = radius_blau_fehler**2
xy_data = XYContainer(x,y)
def linear(x,a=0.0005,b=0):
    return x*a+b
line_fit=Fit(data=xy_data,model_function=linear)
line_fit.add_error(axis='x', err_val=x_error,relative=False)  # add the x-error to
line_fit.add_error(axis='y', err_val=y_error,relative=False)

line_fit.do_fit()
R_blau = ufloat(line_fit.parameter_values[0],line_fit.parameter_errors[0])
c_blau = ufloat(line_fit.parameter_values[1],line_fit.parameter_errors[1])
R_blau_wahr=unumpy.nominal_values(R_blau)
R_blau_fehler=unumpy.std_devs(R_blau)
c_blau_wahr=unumpy.nominal_values(c_blau)
c_blau_fehler=unumpy.std_devs(c_blau)
line_fit.assign_parameter_latex_names(x='(k * λ) / n_l', a='R',b='c')
line_fit.assign_model_function_latex_name('r^2')
line_fit.assign_model_function_latex_expression(' {x} * {a} + {b}')

line_fit.data_container.label = "Verwendete Daten blaues Licht"
line_fit.model_label = "Modellfunktion"

line_fit.data_container.axis_labels = ['r^2 in m^2',"ordnung * λ in m"]
plot = Plot(fit_objects= line_fit)

plot.plot()
plt.show()
print('Aus dem Fit folgt R = ',R_blau,' in m und c = ',c_blau)
```




```
Aus dem Fit folgt R =  0.2456+/-0.0028  in m und c =  (1.5+/-1.5)e-08
```




```
Aus dem Fit folgt R =  0.225+/-0.005  in m und c =  (0.1+/-1.6)e-08
```

Die übrigen Reflexionen, die an anderen Oberflächen auftreten können, spielen normalerweise keine wesentliche Rolle für das Auftreten der Interferenzerscheinungen bei den Newtonschen Ringen. Das liegt daran, dass diese Reflexionen in der Regel keine konstruktive Interferenz erzeugen, die für die Ausbildung der charakteristischen Ringe notwendig ist. Stattdessen tragen sie zu einer allgemeinen Hintergrundbeleuchtung bei.
Ein wesentlicher Nachteil der Durchlichtbeobachtung im Vergleich zur Auflichtbeobachtung besteht darin, dass die Interferenzringe bei der Durchlichtbeobachtung weniger deutlich sichtbar sind. Dies liegt daran, dass das transmittierte Licht durch Streuung und Absorption in der Probe abgeschwächt wird. Zudem können störende Effekte wie Reflexionen an den Oberflächen der Probe auftreten, die die Beobachtung der Ringe erschweren können.

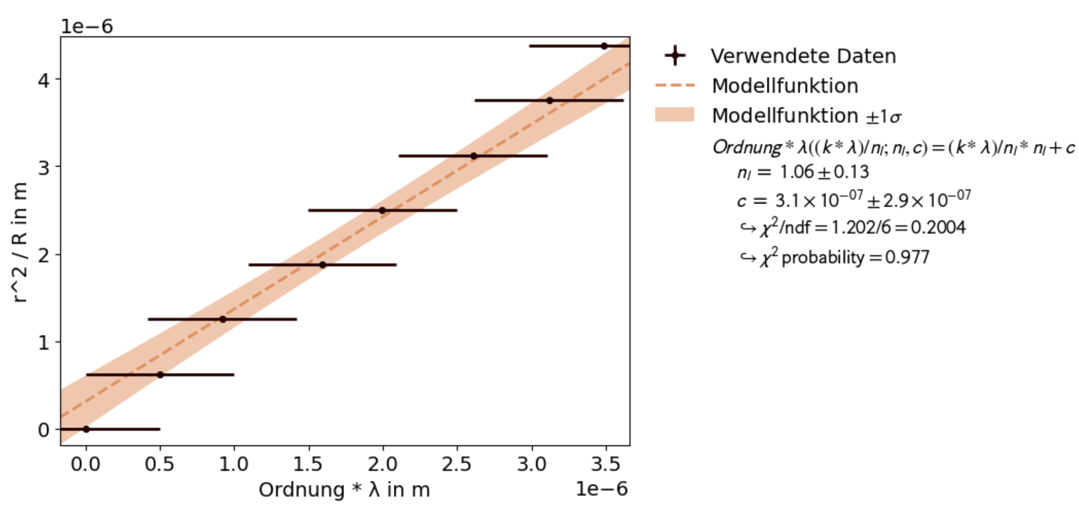
Aufgabe 1.2

Bestimmen Sie den Brechungsindex von Wasser aus den veränderten Durchmessern der Newtonschen Ringe (unter Verwendung der gelben LED), wenn sich zwischen Linse und Objektträger Wasser statt Luft befindet.

Lösung:
Es wird analog zu Aufgabe 1.1 die Messung durchgeführt, jedoch wird der Zwischenraum zwischen der Linse und dem Objektträger mit Wasser gefüllt. Da jetzt ein anderer Brechungsindex vorliegt ist der Radius ein anderer.
Aufgrund dieser Veränderung kann der Brechungsindex bestimmt werden. Für diese Messung wird ausschließlich die gelbe LED verwendet. Mit einem linearen Fit wird der Brechungsindex bestimmt.
Hier wird eine Unsicherheit von $0.1 \cdot 0.00025m$ auf den Radius angenommen. Hier ist der Fehler kleiner als bei der Aufgabe 1.1, da die Ringe mit Wasser näher beieinander sind und somit mit starker Konzentration die Werte bestimmt werden.

In [ ]:

```python
lambda_rot_wahr = 625*10**(-9)
lambda_rot_fehler = 5*10**(-9)
ordnung_rot_wasser=np.array([0,1,2,3,4,5,6,7]) #gemessene Ordnung
radius_rot_wasser = unumpy.uarray(np.array([0,1.4,1.9,2.5,2.8,3.2,3.5,3.7])*0.00025
radius_rot_wasser_wahr=unumpy.nominal_values(radius_rot_wasser)
radius_rot_wasser_fehler=unumpy.std_devs(radius_rot_wasser)

y= ordnung_rot_wasser*lambda_rot_wahr
x= radius_rot_wasser_wahr**2/R_gelb_wahr
y_error = ordnung_rot_wasser*lambda_rot_fehler
x_error = radius_rot_wasser_fehler**2/R_gelb_fehler
xy_data = XYContainer(x,y)
def linear(x,a=0.0005,b=0):
    return x*a+b
line_fit=Fit(data=xy_data,model_function=linear)
line_fit.add_error(axis='x', err_val=x_error,relative=False)  # add the x-error to
line_fit.add_error(axis='y', err_val=y_error,relative=False)

line_fit.do_fit()
n_l = ufloat(line_fit.parameter_values[0],line_fit.parameter_errors[0])
n_l_wahr=unumpy.nominal_values(n_l)
n_l_fehler=unumpy.std_devs(n_l)
c = ufloat(line_fit.parameter_values[1],line_fit.parameter_errors[1])
c_wahr=unumpy.nominal_values(c)
c_fehler=unumpy.std_devs(c)
line_fit.assign_parameter_latex_names(x='(k * λ) / n_l', a='n_l',b='c')
line_fit.assign_model_function_latex_name('Ordnung * λ')
line_fit.assign_model_function_latex_expression(' {x} * {a} + {b}')

line_fit.data_container.label = "Verwendete Daten"
line_fit.model_label = "Modellfunktion"


line_fit.data_container.axis_labels = ["Ordnung * λ in m" ,'r^2 / R in m',]
plot = Plot(fit_objects= line_fit)

plot.plot()
plt.show()
print('Aus dem Fit folgt Brechungsindex n_l = ',n_l,' und c = ',c)
```




```
Aus dem Fit folgt Brechungsindex n_l =  1.06+/-0.13  und c =  (3.1+/-2.9)e-07
```

Der Brechungsindex von Wasser besitzt einen Literaturwert von 1.33.
(https://de.wikipedia.org/wiki/Brechungsindex)
Der von uns ermittelte Wert und der Literaturwert passen nicht überein. Dies kann an verunreinigtem Wasser liegen. Außerdem kann auch von uns ausversehen ein falscher Messwert abgelesen worden sein.

Aufgabe 1.3

Bestimmen Sie die Brennweite f einer Linse durch Autokollimation. Gehen Sie dabei wie folgt vor:

Verschieben Sie eine beleuchtete Mattscheibe mit scharfen Schatten (als "selbstleuchtenden" Gegenstand) vor der zu untersuchenden Linse so, dass er seitenverkehrt wieder scharf auf sich selbst abgebildet wird, wenn hinter der Linse ein Planspiegel das Licht reflektiert. Dabei ist der Abstand zwischen Spiegel und Linse unerheblich.

Lösung:
Bei dieser Teilaufgabe ist der Abstand der Scheibe zur Linse die Brennweite.

In [ ]:
```
marke_1= ufloat(1.04,0.005)
marke_2= ufloat(1.524,0.005)
abstand=marke_2-marke_1
f=abstand
print('Die Brennweite beträgt ',abstand,' m')
```

```
Die Brennweite beträgt  0.484+/-0.007  m
```

Aufgabe 1.4

Bestimmen Sie den Brechungsindex n des Linsenglases aus Radius R der Linse und der Brennweite f. Verifizieren Sie die hierzu benötigte Formel

$$R = (n-1)\,f$$

auf einfache Weise mit der Näherung für eine dünne Linse und der paraxialen Näherung für einen achsennahen Strahlengang.

Lösung:

In der Nähe der optischen Achse und für dünne Linsen kann die paraxiale Näherung angewendet werden, d.h. der Einfallswinkel des Lichtstrahls auf die Linse ist sehr klein. Unter dieser Näherung gilt die Beziehung $\frac{1}{f} = (n-1)\left(\frac{1}{R_1} - \frac{1}{R_2}\right)$.

Man kann annehmen, dass die Krümmungsradien r_1 und r_2 im Vergleich zur Dicke der Linse sehr groß sind. Dies ermöglicht es uns, die Näherung $r_1 \approx \infty$ und $r_2 \approx \infty$ zu verwenden. Unter Verwendung dieser Näherungen vereinfacht sich die obige Gleichung zu $\frac{1}{f} = (n-1)\left(\frac{1}{\infty} - \frac{1}{\infty}\right)$. Da $\frac{1}{\infty}$ als Null definiert ist, ergibt sich $\frac{1}{f} = 0$, was darauf hinweist, dass die Brennweite f unendlich groß ist, was einer idealen dünnen Linse entspricht. Wenn wir diese Beziehung in die gegebene Formel $R = (n-1)\ \ f$ einsetzen, erhalten wir $R = (n-1)\ \ (0) = 0$. Dies bedeutet, dass der Radius RR in diesem Fall ebenfalls Null ist, was darauf hindeutet, dass die Ausdehnung der Newtonschen Ringe nicht vorhanden ist. Daher wird die gegebene Formel $R = (n-1)\, f$ durch die Verwendung der Näherung für eine dünne Linse und die paraxiale Näherung für einen achsennahen Strahlengang verifiziert.

In [ ]:
```
n_Linsenglas= R_gelb/(f)+1
print('Der Brechungsindex des Linsenglases beträgt ',n_Linsenglas)
```

```
Der Brechungsindex des Linsenglases beträgt  1.507+/-0.009
```

Der Brechungsindex von Glas besitzt einen Literaturwert von 1.45 bis 2.14. (https://de.wikipedia.org/wiki/Brechungsindex)
Der von uns ermittelte Wert und der Literaturwert passen überein. Daher ist unser experimenteller Wert gut.

Aufgabe 2: Beugung am Gitter

Aufgabe 2.1

Justieren Sie das Gitterspektrometer. Gehen Sie dabei wie folgt vor:

1. Stellen Sie die Brennweite des Fernrohres (durch Verschieben des Okulars) auf "unendlich", indem Sie einen weit entfernten Gegenstand scharf einstellen. Bei richtiger Einstellung darf dabei bei Kopfbewegung keine Verschiebung (Parallaxe) von einem beobachteten Detail und dem Fadenkreuz zu sehen sein. Normalsichtige beobachten mit entspanntem (d.h. auf "Unendlich" gerichtetem) Auge. Kurzsichtige können Parallellicht nicht auf die Netzhaut fokussieren und benötigen (ohne Kontaktlinse oder Brille) eine etwas abweichende Einstellung.
2. Beleuchten Sie den Spalt mit der Na-Spektrallampe. Überlegen Sie, ob ein Kondensor (Abbildung der Lichtquelle in die Apparatur) von Vorteil sein kann. Beobachten Sie den schmal eingestellten Spalt durch das Fernrohr. Stellen Sie ihn durch Verschieben des Spaltes scharf und bringen Sie ihn dann durch Schwenken des Fernrohres mit dem Fadenkreuz zur Deckung. Stellen Sie den Spektrometertisch mit Teilkreis geeignet ein (Nullstellung) und arretieren Sie ihn. Setzen Sie dann den Spiegel in den Gitterhalter ein (verwenden Sie hierzu einen weit geöffneten Spalt). Stellen Sie zwischen Spalt und Lampe einen Objektträger so unter 45° gegen die Achse auf, dass von der Seite her über diesen "Strahlteiler" das vom Spiegel reflektierte Licht sichtbar wird, falls es durch den Spalt zurücktrifft. Der Spiegel (und damit der Gitterhalter) ist also senkrecht zur Achse justiert (Justierung am Rändelrand des Gitterhalters). Stellen Sie dann den Spalt wieder schmal ein. Tauschen Sie schließlich den Spiegel gegen das Gitter aus.

Aufgabe 2.2

Bestimmen Sie die Gitterkonstante eines Gitters. Gehen Sie dabei wie folgt vor:

Das Gitter ist nur mit "Gitter" bezeichnet. Es hat etwa 600 Striche pro Millimeter. Das Öffnungsverhältnis des Gitters ist mit $b/g \approx 0,9$ sehr groß. Damit wird erreicht, dass die Spaltbreite nicht im Bereich von $\lambda/2$ liegt. Die Breite des Gitters ist genügend groß, um den ganzen Querschnitt des Parallellichtbündels im Spektrometer auszunutzen. Zur Messung wird das gelbe Licht der Na-Spektrallampe verwendet. Legen Sie für Ihre Messungen nur die mittlere Wellenlänge $\langle\lambda\rangle$ der Doppellinie des Na-Spektrums zugrunde; der Abstand der beiden Linien ($\approx\ 0,5\,\mathrm{nm}$) soll in der nächsten Aufgabe genauer bestimmt werden.

Überlegen Sie sich schon bei der Vorbereitung die Antworten auf die folgenden Fragen:

- Unter welchen Winkeln sind etwa Maxima (Hauptmaxima) zu erwarten?
- Sind diese vielleicht durch Minima der Beugungsbilds des Einzelspaltes ausgelöscht?
- Wie breit sind die Maxima in etwa?
- Welches Intensitätsverhältnis haben die Maxima?
- Wie gut ist das theoretische Auflösungsvermögen?
- Wie schmal müssten Sie den Spalt einzustellen um dieses zu nutzen?
- Ist die Doppellinie getrennt beobachtbar und ihr Abstand messbar?

Lösung:

In diesem Aufgabenteil wird die Gitterkonstante eines Gitters bestimmt, unter der Annahme das nur die mittlere Wellenlänge der Natriumdoppellinie bei $589,3$ nm bekannt ist.

Zuerst berechnen wir allerdings die zu erwartenden Winkel der Maxima, mit allen bekannten Werten. Ist die Spaltbreite des Gitters vernachlässigbar, sind die Maxima der Intensitätsverteilung gegeben durch:

$$\sin \alpha_n = \frac{n\lambda}{g}$$

Die Öffnungsverhältnis des Gitters ist gegeben als $b/g = 0.9$. Die Minima der Intensitätsverteilung lassen sich somit berechnen durch

$$\sin \alpha_n = \frac{n\lambda}{b}$$

Damit ergeben sich die Extrema:

Extremazahl	Winkel der Maxima in °	Winkel der Minima in °
1	20,70	23,13
2	45,00	51,788

Es ist zu erkennen, das sich die Minima und Maxima nicht überlagern. Weitere Ordnungen sind nicht sichtbar.

Um das Intensitätsverhältnis zu berechnen, verwenden wir die gegebene Intensitätverteilung

$$I = \frac{\sin \beta}{\beta}^2 \cdot \frac{\sin (N\phi)^2}{\sin \phi}$$

mit
$\beta = \frac{\pi b}{\lambda} \cdot \sin \alpha, \phi = \frac{\pi g}{\lambda} \cdot \sin \alpha$
Dabei ist N die Anzahl der beleuchteten Spalten, N = l/g mit l = 36mm ergibt sich N = 21600
Damit ergeben sich für die Maxima folgende Intensitäten:

Ordnung	Winkel der Maxima in °	Intensität
1	20,70	3890

| 2 | 45,00 | 3791 |\

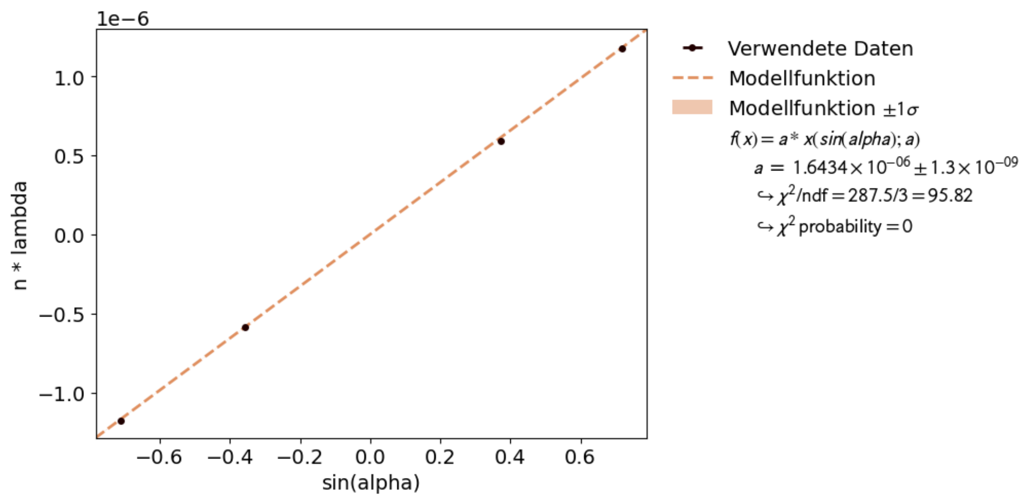
Wodurch sich das Verhältnis $I_1/I_2 = 1.026$ ergibt. Beide Ordnungen werden also ungefähr gleich hell beleuchtet. Das theoretische Auflösungsvermögen ist gegeben durch $\frac{\lambda}{\Delta\lambda} = nN$ um die erste Ordnung aufzulösen, ist es also nötig $N >= \frac{589{,}3}{0{,}5} = 1179$ Spalte zu beleuchten, was etwa 2mm Gitter entspricht.

Die gemessenen Winkel der Maxima sind:

Ordnung	Winkel der Maxima in °
1	20,70
2	45,00

Die Gitterkonstante lässt sich nun mit einem linearen Fit von $n\lambda$ über $\sin\alpha$ ermitteln, dabei ist die Gitterkonstante die Steigung.

In [ ]:
```
n = np.array([1,2,-1,-2])
alpha = unumpy.uarray(np.array([21.82,45.91,-20.82,-45.33]),np.full(4,0.05))
n2= np.array([1,2])
alpha = alpha*np.pi/180
alpha2= np.array([159.18,134.67])
lambda1 = 589.3*10**(-9)
x_error = unumpy.std_devs(alpha)
xy_data = XYContainer(np.sin(unumpy.nominal_values(alpha)),n*lambda1)
def exp(x,a=1):
    return a*x
line_fit=Fit(data=xy_data,model_function=exp)
line_fit.add_error(axis='x', err_val=x_error,relative=False)  # add the x-error to
#line_fit.add_error(axis='y', err_val=y_error,relative=False)

line_fit.do_fit()
line_fit.assign_parameter_names(x='sin(alpha)',a='a' )
line_fit.assign_model_function_expression('f(x) = {a} * {x}')
#line_fit.report()

line_fit.assign_parameter_latex_names(x='sin(alpha)', a='a' )
line_fit.assign_model_function_latex_name("f(x) = a * x ")


line_fit.data_container.label = "Verwendete Daten"
line_fit.model_label = "Modellfunktion"

line_fit.data_container.axis_labels = ["sin(alpha)","n * lambda"]

plot = Plot(fit_objects= line_fit)
plot.plot()
plt.show()


print(1/line_fit.parameter_values[0])
```




```
608512.5584445699
```

Aufgabe 2.3

Bestimmen Sie den Wellenlängenabstand der gelben Na-Spektrallinien. Verwenden Sie hierzu das Gitter von Aufgabe 2.2 und die zu eichende Feineinstellung mit Skala am Spektrometer.

Lösung:

Um den Wellenlängenabstand der Natrium Doppellinie zu bestimmen, messen wir den Winkel der auftretenden Maxima und bestimmen durch die Formel $\Delta\lambda = \frac{n}{g}(\sin\alpha_2 - \sin\alpha_1)$ um die Differenz zu bestimmen.

In [ ]:

```
alpha_1 = unumpy.uarray(np.array([20.83,45.83,180-159.27,180-135.27]),np.full(4,0.0
alpha_2 = unumpy.uarray(np.array([20.92,45.88,180-159.22,180-135.18]),np.full(4,0.0
alpha_1 = alpha_1 *np.pi/180
alpha_2 = alpha_2 *np.pi/180
n = np.array([1,2,1,2])
d_lambda = 10**(-3)/(600*n)*(unumpy.sin(alpha_2) - unumpy.sin(alpha_1))
print(d_lambda)
print(np.mean(unumpy.nominal_values(d_lambda))*10**9,"nm")
```

```
[2.4461486175333945e-09+/-3.8437511048510137e-10
 5.064920581059525e-10+/-1.432575929461642e-10
 1.3600540136854411e-09+/-3.8468134718913655e-10
 9.292281953923732e-10+/-1.4601413836108206e-10]
1.3104807211792904 nm
```

Aufgabe 2.4

Bestimmen Sie die Gitterkonstante eines zweiten Gitters. Gehen Sie hierzu wie folgt vor:

Das Gitter ist mit dem Grobwert $140\,\mathrm{mm}^{-1}$ für die Strichdichte bezeichnet. Details der Gitterstruktur sind nicht bekannt. Als Lichtquelle dient wieder die Na-Spektrallampe. Beantworten Sie die folgenden Fragen:

- Das Gitter kann bis bis zu 6. Ordnung mit monoton abnehmender Intensität in jeder Ordnung beobachtet werden. Haben Sie eine Erklärung dafür?
- Von welcher Ordnung ab ist die Na-Doppellinie getrennt beobachtbar?
- Entspricht dies dem theoretischen Auflösungsvermögen?

Lösung:

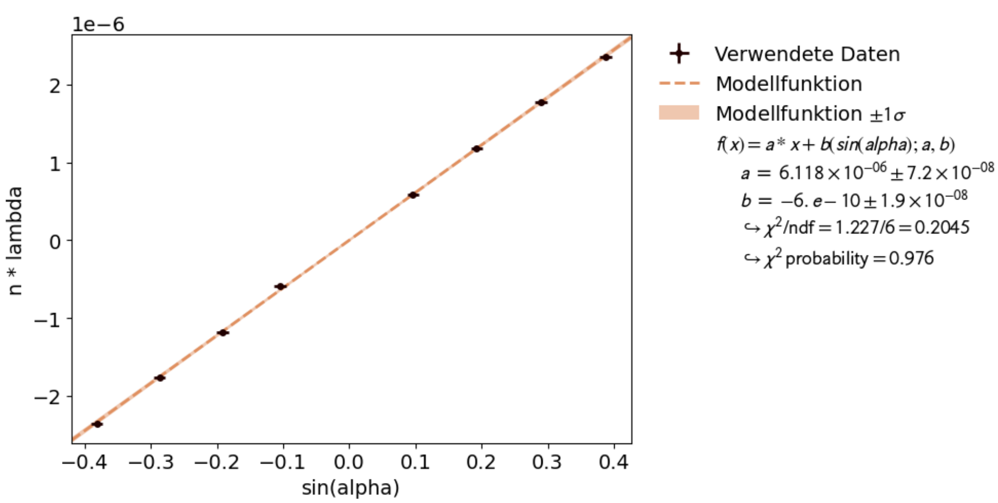
Eine monoton abfallende Intensität bedeutet, das kein Maximum auf einem Minimum der Einhüllenden liegt.
Die Spaltbreite des Gitters ist vermutlich ähnlich groß wie die Wellenlänge des Lichts, sodass die Einhüllende sehr breit ist und alle Maxima umschließt.

Mit dem theoretischen Auflösungsvermögen ergibt sich bei Beleuchtung des gesamten Gitters, das sich die Doppellinie schon ab dem erstem Maximum auflösen lässt.

Durch einen linearen Fit von $n\lambda$ über $\sin\alpha$ der Maxima lässt sich die Gitterkonstante g direkt aus der Steigung ablesen.

In [ ]:

```
n = np.array([1,2,3,4,-1,-2,-3,-4])
alpha_wahr = np.array([5.51,11.08,16.83,22.8,-(180-174.0),-(180-168.95),-(180-163.3
alpha_wahr = alpha_wahr* np.pi/180
alpha_fehler = np.full(8,0.50)* np.pi/180
lambda3_wahr = 589.3*10**(-9)
lambda3_fehler = 0.1*10**(-9)

x_error = np.abs(np.sin(alpha_fehler))
y_error = np.abs(lambda3_fehler*n)
xy_data = XYContainer(np.sin(alpha_wahr),n*lambda3_wahr)
def exp(x,a=0,b=0):
    return a*x + b
line_fit=Fit(data=xy_data,model_function=exp)
line_fit.add_error(axis='x', err_val=x_error,relative=False)  # add the x-error to
line_fit.add_error(axis='y', err_val=y_error,relative=False)
line_fit.do_fit()
line_fit.assign_parameter_names(x='sin(alpha)',a='a' , b='b')
line_fit.assign_model_function_expression('f(x) = {a} * {x} + {b}')
#line_fit.report()

line_fit.assign_parameter_latex_names(x='sin(alpha)', a='a' , b='b')
line_fit.assign_model_function_latex_name("f(x) = a * x + b")


line_fit.data_container.label = "Verwendete Daten"
line_fit.model_label = "Modellfunktion"

line_fit.data_container.axis_labels = ["sin(alpha)","n * lambda"]

plot = Plot(fit_objects= line_fit)
plot.plot()
plt.show()

print(1/(6.1*10**-6))
```




```
163934.42622950822
```

Aufgabe 2.5

Messen Sie möglichst genau die Wellenlängen der vier deutlich erkennbaren Linien einer Zn -Spektrallampe. Die Farben sind violettblau, blau, blaugrün und rot. Begründen Sie Ihre Wahl der Mittel und der Methode in Ihrer Auswertung.

Lösung:

Um eine möglichst hohe Auflösung zu haben, wählen wir das Gitter mit der größten Gitterkonstante, g= 140/mm

Es werden die Ablenkwinkel in beide Richtungen gemessen und deren Mittel im folgenden dargestellt. Die entsprechenden Wellenlängen ergeben sich dann wieder aus:

$$\sin\alpha = \frac{n\lambda}{g}$$

In [ ]:

```
g=1/140000
alpha_1_db = unumpy.uarray(np.array([4.33,8.83,13.4]),np.full(3,0.01))
alpha_2_db = unumpy.uarray(np.array([180-175.65,180-171.25,180-166.88]),np.full(3,0
alpha_1_db = alpha_1_db *np.pi/180
alpha_2_db = alpha_2_db *np.pi/180
alpha_db = (alpha_1_db + alpha_2_db)/2
n = [1,2,3]
lamda_db = unumpy.sin(alpha_db)*g/n
lamda_db_wahr = np.mean(unumpy.nominal_values(lamda_db))
lamda_db_fehler = np.mean(unumpy.std_devs(lamda_db))
lamda_db = ufloat(lamda_db_wahr,lamda_db_fehler)
print(lamda_db)

alpha_1_hdb = unumpy.uarray(np.array([4.5,8.98,13.67]),np.full(3,0.01))
alpha_2_hdb = unumpy.uarray(np.array([180-175.58,180-171.17,180-166.72]),np.full(3,
alpha_1_hdb = alpha_1_hdb *np.pi/180
alpha_2_hdb = alpha_2_hdb *np.pi/180
alpha_hdb = (alpha_1_hdb + alpha_2_hdb)/2
lamda_hdb = unumpy.sin(alpha_hdb)*g/n
lamda_hdb_wahr = np.mean(unumpy.nominal_values(lamda_hdb))
lamda_hdb_fehler = np.mean(unumpy.std_devs(lamda_hdb))
lamda_hdb = ufloat(lamda_hdb_wahr,lamda_hdb_fehler)
print(lamda_hdb)

alpha_1_hb = unumpy.uarray(np.array([4.55,9.17,13.9]),np.full(3,0.01))
alpha_2_hb = unumpy.uarray(np.array([180-175.53,180-171,180-166.48]),np.full(3,0.01
alpha_1_hb = alpha_1_hb *np.pi/180
alpha_2_hb = alpha_2_hb *np.pi/180
alpha_hb = (alpha_1_hb + alpha_2_hb)/2
lamda_hb = unumpy.sin(alpha_hb)*g/n
lamda_hb_wahr = np.mean(unumpy.nominal_values(lamda_hb))
lamda_hb_fehler = np.mean(unumpy.std_devs(lamda_hb))
lamda_hb = ufloat(lamda_hb_wahr,lamda_hb_fehler)
print(lamda_hb)

alpha_1_r = unumpy.uarray(np.array([5.97,12.17,18.58]),np.full(3,0.01))
alpha_2_r = unumpy.uarray(np.array([180-174.03,180-168.08,180-162.03]),np.full(3,0.
alpha_1_r = alpha_1_r *np.pi/180
alpha_2_r = alpha_2_r *np.pi/180
alpha_r = (alpha_1_r + alpha_2_r)/2
lamda_r = unumpy.sin(alpha_r)*g/n
lamda_r_wahr = np.mean(unumpy.nominal_values(lamda_r))
lamda_r_fehler = np.mean(unumpy.std_devs(lamda_r))
lamda_r = ufloat(lamda_r_wahr,lamda_r_fehler)
print(lamda_r)
```

```
(5.441+/-0.005)e-07
(5.544+/-0.005)e-07
(5.633+/-0.005)e-07
(7.449+/-0.005)e-07
```

In [ ]: