
Versuch 255

21. November 2021

Röntgenspektrometer

Physikalisches Anfängerpraktikum II

Juan Provencio

Betreuer/in: Felix Waldherr

Inhaltsverzeichnis

1	Ziel des Versuches	2
2	Grundlagen	2
2.1	Funktionsweise einer Röntgenröhre	2
2.2	Bragg-Reflexion	3
2.3	Kristalle	5
3	Versuchsaufbau	6
3.1	Materialen und Geräte	6
3.2	Aufbau	6
4	Messung und Auswertung	7
4.1	Messprotokoll	7
4.2	Auswertung	8
5	Zusammenfassung und Diskussion	13
5.1	Zusammenfassung	13
5.2	Diskussion	14
6	Quellen	15

1 Ziel des Versuches

Im folgenden Versuch werden wir mithilfe zweier Kristallen, Lithiumfluorid LiF und Natriumchlorid NaCl die Röntgenstrahlung untersuchen. Mit ihnen ist es möglich das Spektrum einer Röntgenquelle zu untersuchen, mit welchem wir beispielsweise die Planck-Konstante h bestimmen werden und auch die Wellenlängen der K_α und K_β Übergänge. Zusätzlich werden wir die Gitterkonstante, d.h., der Abstand der Atome in den Kristallen und die Avogadro-Zahl bestimmen.

2 Grundlagen

2.1 Funktionsweise einer Röntgenröhre

Eine Kathode und eine Anode werden in einem evakuierten Glasrohr reingebracht. Zwischen der Kathode und der Anode wird dann eine Hochspannung angelegt, typischerweise in einem Bereich zwischen 10 kV und 100 kV. An der Kathode werden dann Elektronen emittiert, die in Richtung der Anode hinbeschleunigt werden. Beim Aufprallen gegen die Anode verlieren die Elektronen Energie, welche in Form elektromagnetischer Wellen ausgestrahlt wird. Dieses ist ein kontinuierliches Spektrum, welches wir als Bremspektrum bezeichnen. Die Elektronen besitzen aufgrund des elektrischen Feldes die Energie $E = eU$. Beim Bremsvorgang lässt sich diese Energie in Strahlung übersetzen, weshalb es eine minimale Wellenlänge λ_{gr} gibt, die wir am Spektrum erkennen können, diese lässt sich mittels Umformungen herleiten als:

$$E = eU = h\nu_{\text{gr}} = h \frac{c}{\lambda_{\text{gr}}} \quad (1)$$

$$\iff \lambda_{\text{gr}} = \frac{hc}{eU} \quad (2)$$

Hier bezeichnen h das Planck'sche Wirkungsquantum, c die Lichtgeschwindigkeit, e die Elementarladung und U die Beschleunigungsspannung.

Außerdem können bei ausreichend hohen Beschleunigungsspannungen die Elektronen beim Kontakt mit dem Anodematerial durch Ionisation Energie verlieren, wie wir schon im PAP 1 untersucht haben ergibt dieser Prozess ein diskretes Spektrum, weshalb man zusätzlich zum kontinuierlichen Bremspektrum auch ein Linienspektrum beobachten kann. Die Energie hier entsteht, wenn ein beschleunigtes Elektron ein Elektron aus einer inneren Schale des Anodenmaterials rausschlägt, dann entsteht eine Lücke die von

einem Elektron einer höheren Schale gefüllt sein muss. Die dabei frei werdende Energie wird dann als ein Röntgenquant ausgestrahlt. Je nachdem, von welcher Schale das Elektron rausgeschlagen wurde, und aus welcher Schale das neue Elektron seine Lücke füllt wird unterschiedlich viel Energie frei. Die Energie, die beim Übergang von der n -ten auf die m -ten Schale frei wird lässt sich grob berechnen als

$$E_{n \rightarrow m} = hcR_{\infty}(Z - A)^2 \left(\frac{1}{m^2} - \frac{1}{n^2} \right) \quad (3)$$

Hier ist R_{∞} die Rydberg Konstante, Z die Kernladungszahl und A eine Abschirmungskonstante. Für unsere Zwecke ist $A \approx 1$. Dieses Gesetz ist als das Moseley'sche Gesetz bekannt.

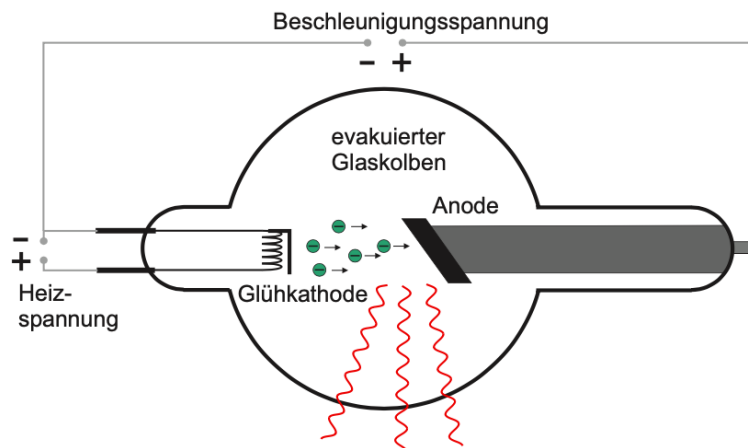


Abbildung 1: Aufbau einer Röntgenröhre

2.2 Bragg-Reflexion

Als man versucht hat, das Spektrum von Röntgenstrahlung zu untersuchen ist man gegen ein großes Hinderniss gestoßen. Normalerweise würde man auf Effekte der Dispersion und Beugung, wie zum Beispiel bei einem Prismen- oder Gitterspektrometer. Im Prismenspektrometer hatte man das Problem, dass der Brechungsindex für die zu untersuchenden Wellenlängen zu klein war, bzw. zu nah an der Eins war und deswegen ließ sich das Spektrum nicht beobachten. Mit dem Gitter hätte man eine Gitterkonstante gebraucht, die technisch nicht realisierbar gewesen wäre.

Mit besonderen Kristallen ließ sich aber ein Beugungsmuster erkennen, und kam zur Schlussfolgerung, dass die Atomstruktur dieser Kristalle periodisch

und regelmäßig sein musste. Der Abstand der Atome zueinander diente also wie die Gitterkonstante. William Henry Bragg und sein Sohn William Lawrence Bragg erklärten 1913 den Zusammenhang zwischen dieser Kristallstruktur und der Beugung, weshalb diese Methode der Röntgenbeugung nach ihnen benannt ist.

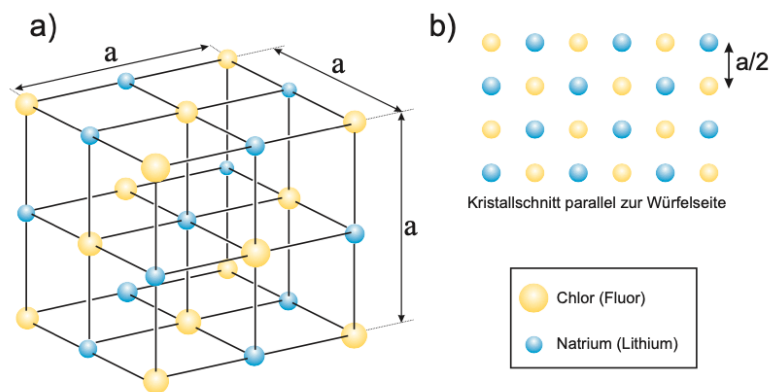


Abbildung 2: Aufbau eines Kristalls

Die Beugung am Kristall folgt den gleichen Gesetzen wie die Beugung an einem Gitter, mit den üblichen Winkelverhältnissen. Dabei wird ein Intensitätsmaximum an den Stellen

$$2d \sin \vartheta = n\lambda \quad (4)$$

detektiert. Hier ist $d/2$ die Gitterkonstante, ϑ der Einfallswinkel und λ die Wellenlänge. Der Gangunterschied kann bestimmt werden als

$$\Delta s = 2d \sin \vartheta \quad (5)$$

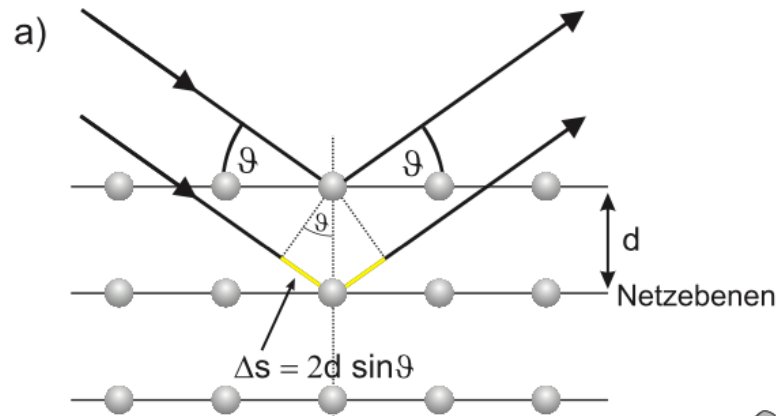


Abbildung 3: Bragg-Reflexion an einem Kristall

Bei bekannter Wellenlänge kann man somit die Struktur eines Kristalles untersuchen und bei bekannter Struktur kann man das Spektrum einer Röntgenquelle untersuchen. Dies werden wir machen, indem wir den Einfallswinkel ϑ ändern und die Intensität der Strahlung am Detektor dabei beobachten.

2.3 Kristalle

Die zu erforschenden Kristallen LiF und NaCl besitzen jeweils eine kubische Struktur, das heißt, dass die drei Gitterkonstanten a gleich groß sind. Bei diesen Kristallen ist es einfach die Avogadro-Konstante N_A zu bestimmen, für welche das molare Volumen und das Volumen einer Zelle gebraucht sind. Man überlegt sich, dass dadurch dass einige Atome von mehreren Elementarzellen geteilt werden, insgesamt jede Zelle 4 "eigene" Atomen besitzt. Somit kann man N_A bestimmen als

$$N_A = 4 \frac{V_{\text{mol}}}{V} \quad | \quad V = (a)^3 = (2d)^3 \quad (6)$$

$$= 4 \frac{V_{\text{mol}}}{(2d)^3} \quad | \quad V_{\text{mol}} = \frac{M_{\text{mol}}}{\rho} \quad (7)$$

$$= \frac{M_{\text{mol}}}{2\rho d^3} \quad (8)$$

M_{mol} entspricht hier die molare Masse und ρ die Dichte, welche im Praktikumsskript angegeben werden.

3 Versuchsaufbau

3.1 Materialien und Geräte

- Röntgengerät mit Röntgenröhre (Molybdän Anode)
- Goniometer
- Zählrohr
- LiF-Kristall
- NaCl-Kristall
- Computer
- Leuchtschirm mit CCD-Kamera

3.2 Aufbau



Abbildung 4: Placeholder

4 Messung und Auswertung

4.1 Messprotokoll

Messprotokoll V255: Röntgenspektrometer 22.11.21
Mike Brand
Juan Provencio

Funktionstest: $I = 1\text{ mA}$, $U = 5\text{ kV}$

1. Messung Röntgenspektrum mit LiF-Kristall

(a) Röntgenspektrum Molybdän soll bestimmt werden. Dazu wird LiF-Kristall montiert.

Dazu werden $U = 35\text{ kV}$, $I = 1\text{ mA}$, $t = 5\text{ sec}$, sowie $\Delta\beta = 0.2^\circ$ eingestellt.

Gemessen wird der Bereich $3^\circ - 22^\circ$

\Rightarrow Protokoll per Computer generiert

(b) K_α und K_β werden am (a) bestimmt und mit $\Delta\beta = 0.1^\circ$ und $t = 20\text{ sec}$ genauer vermessen

$K_\alpha^{(1)} =$, $K_\beta^{(1)} =$, $K_\alpha^{(2)} =$, $K_\beta^{(2)} =$

(c) Neue Zählrate bei $\beta = 7.5^\circ$ (fest) für $U = 20$ bis 35 kV in 1 kV -Schritten und 20 sec Messzeit.

Spannung [kV]	Zählrate [1/sec]
20	1,2
21	2,0
22	1,0
23	5,5
24	45,55
25	107,2
26	157,2
27	201,5
28	240,1
29	285,1
30	329,5
31	366,4
32	404,6
33	441,1
34	487,8
35	520,5

Aut. Wah.

Abbildung 5: Messprotokoll

4.2 Auswertung

Im Folgenden wird mit Mathematica's Werkzeugen zur Fehlerrechnung der Fehler der angegebenen Größen nach der Gaußschen Fehlerfortpflanzung angegeben. Es wird dabei angenommen, dass jede Variable voneinander unabhängig ist, und dass der relative Fehler klein ist, sodass eine Näherung erster Ordnung durchgeführt werden kann. Genauere Informationen sind hier zu finden.

4.2.1 Kurzwelliges Ende des Spektrums

Das kurzwellige Ende des Spektrums besitzt einen einigermaßen geraden Anstieg. Der wird hier auf Diagramm 1 abgebildet

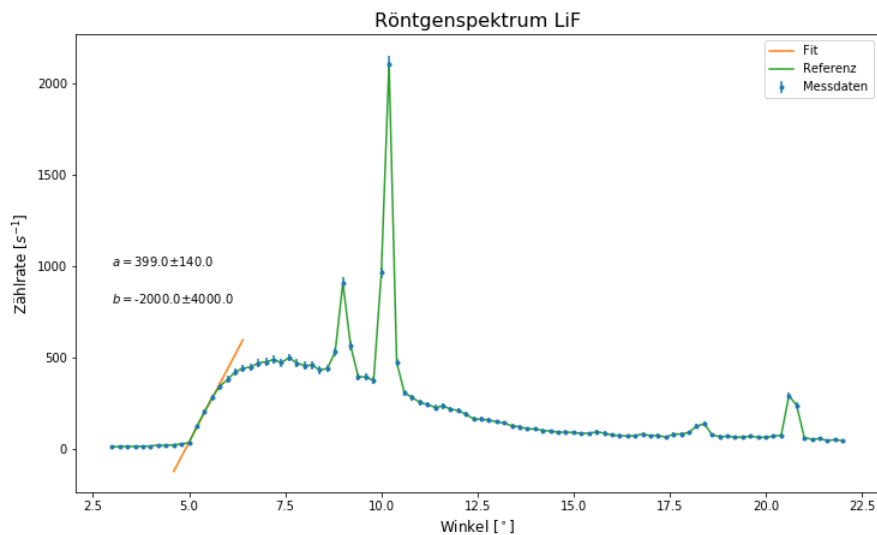


Diagramm 1: Extrapolierung des linearen Anstiegs am kurzwelligen Ende des LiF Spektrums

Daraus wurde die Steigung a und den Achsenabschnitt b mit der Ordinate bestimmt. Mithilfe dieser beiden Werte lassen sich der Grenzwinkel und die Grenzwellenlänge bestimmen, indem man die Zählrate gleich Null setzt:

$$0 = ax + b \quad (9)$$

$$x = -\frac{b}{a} \quad (10)$$

$$= 5,01(0,27)^\circ \quad (11)$$

Daraus lässt sich die Grenzwellenlänge nach (4) bestimmen als:

$$\lambda_{\text{gr}} = \frac{2d \sin x}{n} \quad | d = 201,4 \text{ pm}; n = 1 \quad (12)$$

$$= 35,2(1,9) \text{ pm} \quad (13)$$

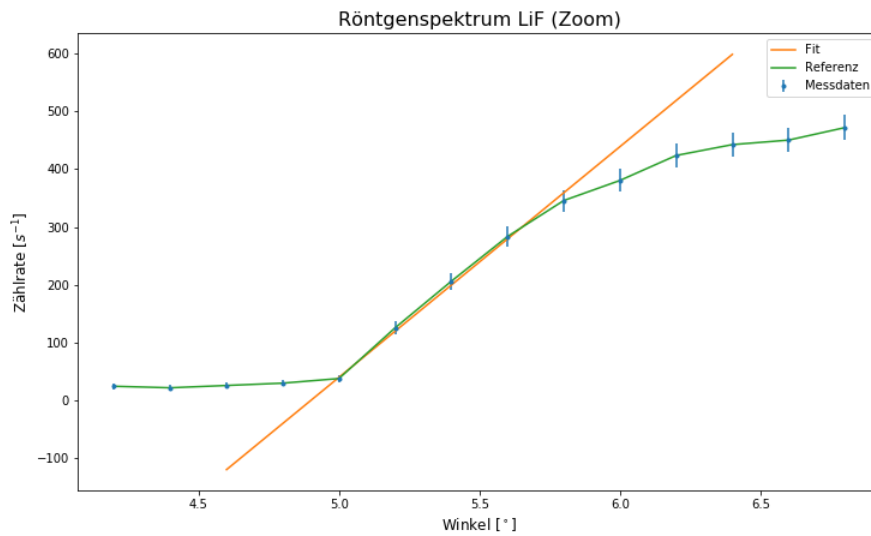


Diagramm 2: Nahaufnahme der linearen Anpassung

Außerdem hat man die Spannung $U = 35 \text{ kV}$ und die Konstanten e, c gegeben, womit wir nach (2) die Planck-Konstante bestimmen können:

$$h = 6,6(0,4) \cdot 10^{-34} \text{ J s} \quad (14)$$

Das Spektrum 2. Ordnung tritt nach (4) um den Winkel

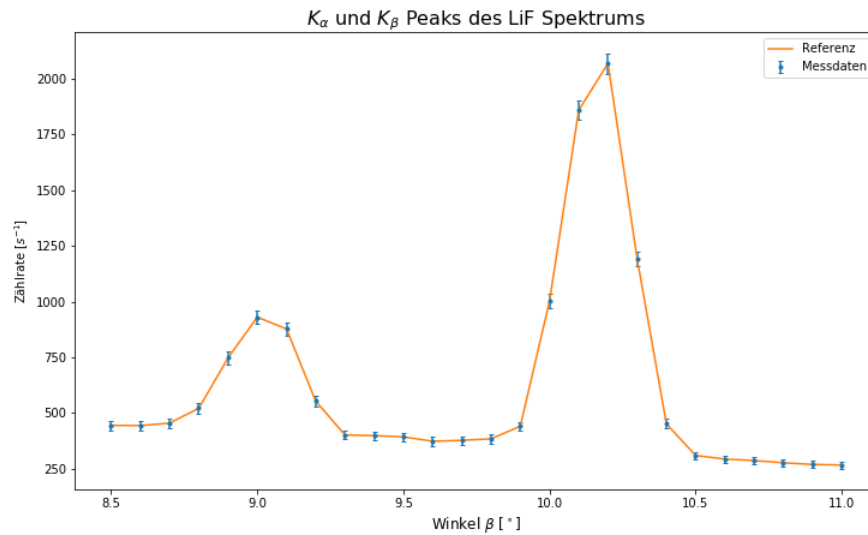
$$x_2 = \arcsin \left(\frac{\lambda_{2 \text{ gr}}}{2d} \right) \quad (15)$$

$$= 10,1(0,6)^\circ \quad (16)$$

auf.

4.2.2 Peaks

Wir tragen die Zählrate als Funktion des Winkels β für die vier verschiedenen Peaks auf und bestimmen daraus die Wellenlänge für K_α und K_β .

Diagramm 3: K_α und K_β Peaks 1. Ordnung

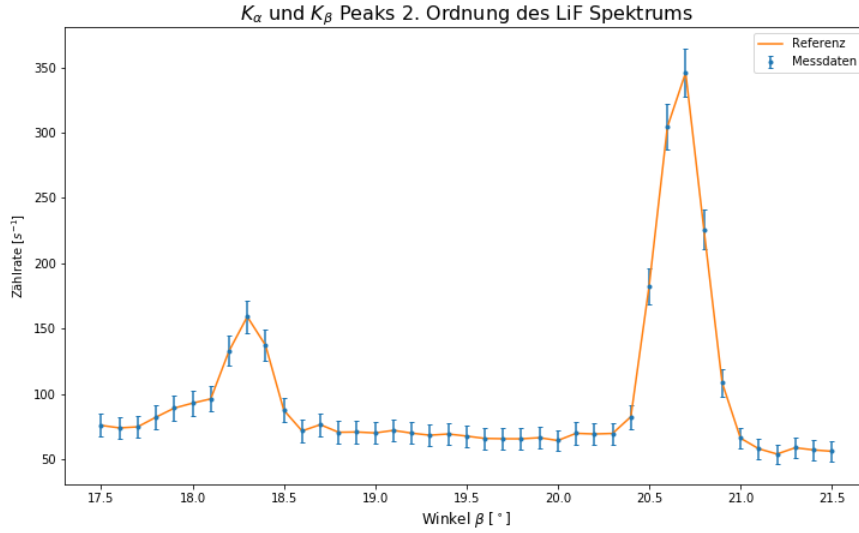
Aus den Werten für die Peaks erhalten wir eine Wellenlänge¹ von

$$\lambda_\alpha = 71,3(2,8) \text{ pm} \quad (17)$$

$$\lambda_\beta = 63,0(2,8) \text{ pm} \quad (18)$$

Für den Fehler haben wir grob die Halbwertsbreite am Diagramm abgeschätzt.

¹ebenfalls mit (2)

Diagramm 4: K_α und K_β Peaks 2. Ordnung

Für die Peaks zweiter Ordnung erhält man eine Wellenlänge von

$$\lambda_{\alpha_2} = 71,2(1,4) \text{ pm} \quad (19)$$

$$\lambda_{\beta_2} = 63,3(2,0) \text{ pm} \quad (20)$$

Aus diesem Wert und dem Wert aus den Peaks 1. Ordnung berechnen wir einen groben Mittelwert:

$$\lambda_{\alpha, \text{Mo}} = 71,3(3) \text{ pm} \quad (21)$$

$$\lambda_{\beta, \text{Mo}} = 63,3(3) \text{ pm} \quad (22)$$

Für den K_α Peak 1. Ordnung bestimmen wir die Halbwertsbreite mithilfe eines Python Codes. Da der Gauß-Fit gut genug ist lässt sich die Halbwertsbreite als

$$\text{FWHM} = 2\sqrt{2 \ln 2} \sigma \quad (23)$$

ausrechnen. Wir brauchen jedoch die Werten der Winkel um diese Halbwertsbreite, also gucken uns den halben Wert der Halbwertsbreite um das Maximum²:

$$\beta_{\text{FWHM}/2} = \beta_\mu \pm \sqrt{2 \ln 2} \sigma \quad (24)$$

²Werte sind auf dem Diagramm aufgetragen

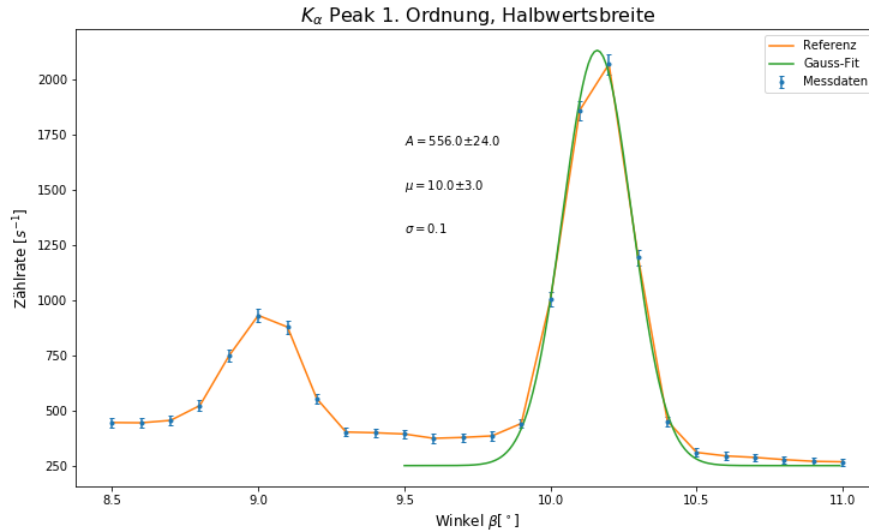


Diagramm 5: Halbwertsbreite

$$\lambda_{\text{FWHM}} \approx 1,9 \text{ pm} \quad (25)$$

4.2.3 Einsatzspannung

Wir bestimmen die Einsatzspannung, d.h. die Spannung ab welche es Quanten mit einem konstanten Winkel $\beta = 7,5^\circ$ gibt. Hieraus lässt sich wieder die Planck-Konstante bestimmen mit (2)

$$h = \frac{\lambda e U}{c} \quad (26)$$

Die Messwerte entnehmen wir der Tabelle im Messprotokoll. Wir erhalten eine Einsatzspannung von

$$U = 22,56 \text{ kV} \quad (27)$$

und schließlich eine Planck-Konstante von

$$h = 6,3(1,3) \cdot 10^{-34} \text{ J s} \quad (28)$$

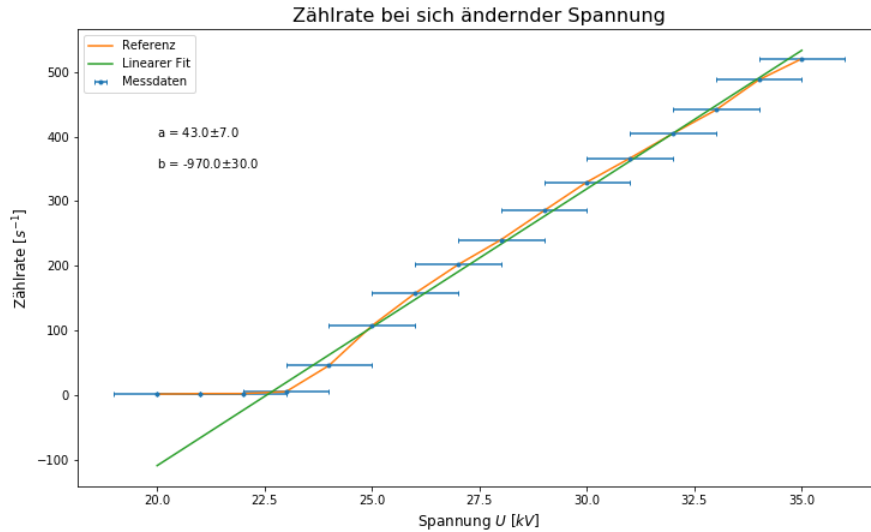


Diagramm 6: Zählrate bei sich ändernder Spannung

4.2.4 Röntgenspektrum am NaCl-Kristall

Als nächstes wollen wir die Gitterkonstante mit den im Teil Peaks 4.2.2 ermittelten Wellenlängen. Die Gitterkonstante lässt sich nach dem Bragg-Gesetz aus (4) bestimmen als:

$$g = \frac{n\lambda}{\sin \vartheta} \quad (29)$$

Und mit dem Python-Skript erhalten wir einen Wert von

$$g = 571(5) \text{ pm} \quad (30)$$

Damit lässt sich die Avogadro-Konstante bestimmen als

$$N_A = \frac{4M_{\text{mol}}}{\rho g^3} = 5,78(17) \text{ mol}^{-1} \quad | M_{\text{Mol}} = 58,44 \text{ g}, \rho = 2,164 \text{ g cm}^3 \quad (31)$$

5 Zusammenfassung und Diskussion

5.1 Zusammenfassung

In diesem Versuch wurde mithilfe eines Röntgenspektrometers das Spektrum von Molybdän mit einem LiF und mit einem NaCl Kristall als Gitter untersucht. Mit dem ersten ließen sich die Wellenlängen der $K_{\alpha,\beta}$ Peaks bestimmen.

Mit dem LiF-Kristall haben wir zusätzlich die Planck-Konstante ausgerechnet. Die Wellenlängen der Peaks waren später nützlich, um die Gitterkonstante des NaCl-Kristalls zu bestimmen, und damit und andere bekannte Werte ließ sich die Avogadro-Konstante herleiten. Insgesamt haben wir Werte über einen Bereich größer als 50 Größenordnungen³ untersucht. Wir haben die ganz feine Planck-Konstante und die ganz große Avogadro-Zahl bestimmt, und dies alle mit einer ähnlichen Herangehensweise.

5.2 Diskussion

Aus dem Anhang des Praktikumsskripts sind die Literaturwerte für die K -Linien von Molybdän, die Planck- und die Avogadro-Konstante entnommen. Die Gitterkonstante von NaCl wurde aus (Wikipedia, 2002) entnommen. Wir stellen unsere Ergebnisse, die Literaturwerte und die Sigma-Abweichung tabellarisch dar. Die Sigma-Abweichung einer Größe G wird folgendermaßen berechnet:

$$\sigma = \frac{|G - G_{\text{Lit}}|}{\sqrt{\Delta_G^2 + \Delta_{G_{\text{Lit}}}^2}} \quad (32)$$

Größe	Messung	Rel. Fehler	Literatur	σ
h [10^{-34} J s] ((14))	6,6(0,4)	0,06	6,6261	0,07
(28)	6,3(1,3)	0,2		0,26
λ_α [10^{-12} m]	71,3(3)	0,04	71,1	0,7
λ_β [10^{-12} m]	63,3(3)	0,05	63,1	0,7
Gitterkonstante g [10^{-12} m]	572(5)	0,009	564	1,6
N_A [10^{23} mol ⁻¹]	5,78(17)	0,03	6,0221	1,4

Tabelle 1: Vergleich zu Literaturwerte

Aus den Sigma-Abweichungen lässt sich keinen signifikanten Fehler beobachten. Außerdem sind die relativen Fehler in den meisten Fällen ziemlich klein, weshalb die geringe Abweichung sich nicht auf eine Überschätzung des absoluten Fehlers zurückführen lässt. Dennoch lassen sich einige Verbesserungen erkennen, nämlich mit dem Umgang der Fehlerrechnung mit "gefitteten" Werte. Beispielsweise ist für Gaußkurven die Halbwertsbreite eine verlässliche Methode um den Fehler zu bestimmen, aber bei anderen Fits wird der Fehler oft unrealistisch angezeigt.

³in unterschiedlichen Einheiten natürlich

Bemerkenswert ist der Unterschied der zwei Methoden zur Berechnung der Planck-Konstante. Mit der zweiten Methode, wo wir die Einsatzspannung extrapoliert haben, haben wir einen relativen Fehler und eine Abweichung vom Literaturwert über 3 Mal so groß wie mit der Methode der Extrapolation der Grenzwellenlänge.

Mögliche Fehler liegen im Allgemeinen in der Messung. Insbesondere bei der Bestimmung der Peaks wäre eine genauere Messung vorteilhaft zur konkreteren Bestimmung des Peaks. Diese Fehlerquelle haben wir reduziert, indem wir uns auf dem zweiten Teil der Aufgabe uns auf die Umgebung der Peaks 1. Ordnung und 2. Ordnung konzentriert haben und dort in feineren Schritte gemessen haben. Trotzdem würde eine Verfeinerung bis zu einem gewissen Grad noch vorteilhaft sein, denn die Auswertung genauso viel Zeit fordert. Der Unterschied daran wären einige Minuten längere Messung.

6 Quellen

Wagner, J., Universität Heidelberg (2021). Physikalisches Praktikum PAP 2.2 für Studierende der Physik B.Sc..

Wikipedia-Autoren. (2002, 2. August). Natriumchlorid. Wikipedia. Abgerufen am 29. Dezember 2021, von <https://de.wikipedia.org/wiki/Natriumchlorid>

Anhang

VII.1

In [1]:

```
import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np
from scipy.optimize import curve_fit
from scipy.signal import find_peaks

# Konstanten
e = 1.6022e-19 # C
c = 3e8 # m s^-2
d = 201.4e-12 # m
```

Kurzwelliges Ende des LiF Spektrums

In [2]:

```
# Werte
deg = np.loadtxt('data/V255LiF3-22.txt', dtype=str, usecols=[0])
deg = np.char.replace(deg, ',', '.').astype(float)

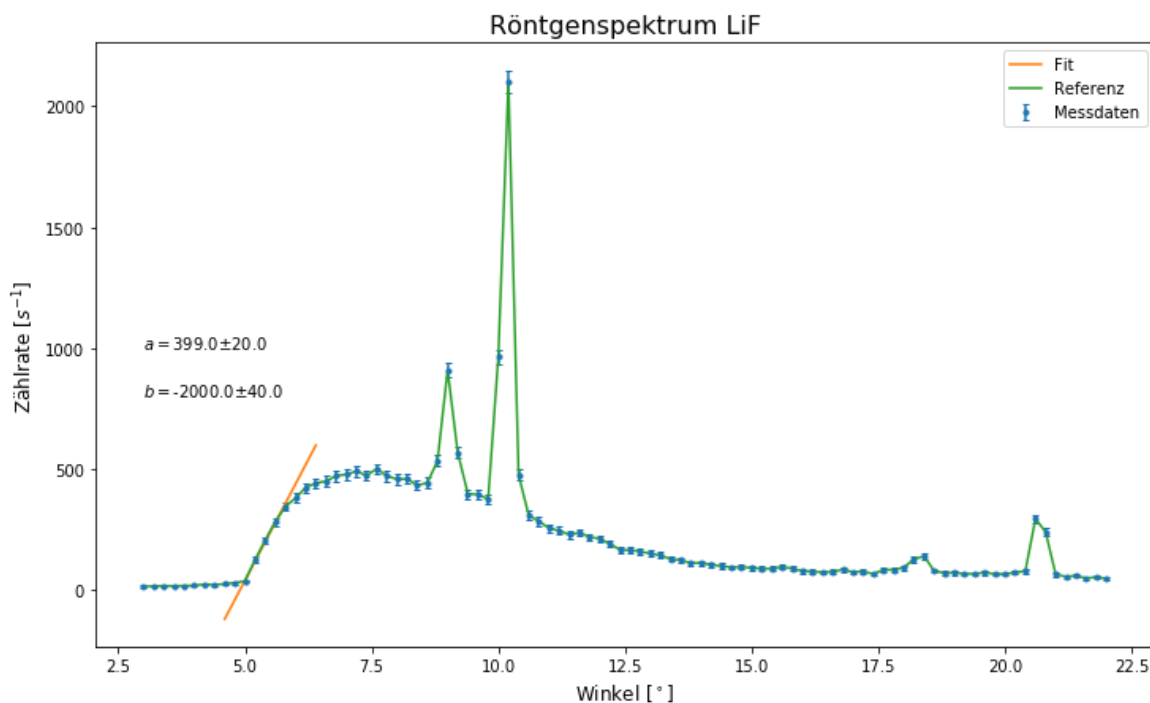
n = np.loadtxt('data/V255LiF3-22.txt', dtype=str, usecols=[1])
n = np.char.replace(n, ',', '.').astype(float)
sig_n = np.sqrt(n)

# Define fit
def line(x, a, b):
    return a * x + b

# Fit
popt, pcov = curve_fit(line, deg[10:15], n[10:15], sigma=sig_n[10:15])
```

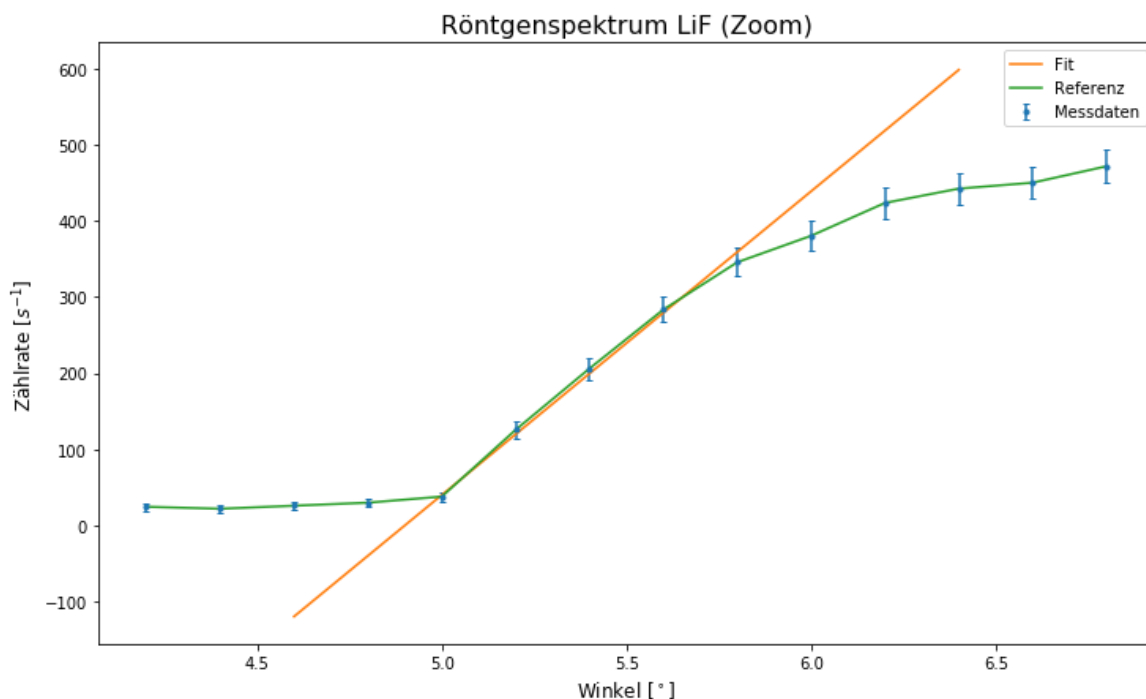

In [3]:

```
# Plot
plt.figure(figsize = (12,7))
plt.errorbar(deg, n, yerr = sig_n, fmt='.', label = 'Messdaten', capsize = 2)
plt.plot(deg[8:18], line(deg[8:18],*popt), label = 'Fit')
plt.plot(deg, n, label = 'Referenz')
plt.title('Röntgenspektrum LiF', size = 16)
plt.xlabel('Winkel' + '  $[\text{\textcircled{~}}]$ ', size = 12)
plt.ylabel('Zählrate' + '  $[\text{s}^{-1}]$ ', size = 12)
plt.text(3, 1000, '$a = $' + str(np.round(popt[0],0)) + '$\pm$' + str(np.round(np.sqrt(popt[0]),0)))
plt.text(3, 800, '$b = $' + str(np.round(popt[1],-3)) + '$\pm$' + str(np.round(np.sqrt(-popt[1]),-1)))
plt.legend(loc = 'upper right')
plt.savefig('images/V255Diagramm1.png')
plt.show()
```



In [4]:

```
# Plot
plt.figure(figsize = (12,7))
plt.errorbar(deg[6:20], n[6:20], yerr = sig_n[6:20], fmt='.', label = 'Messdaten',
capsize = 2)
plt.plot(deg[8:18], line(deg[8:18],*popt), label = 'Fit')
plt.plot(deg[6:20], n[6:20], label = 'Referenz')
plt.title('Röntgenspektrum LiF (Zoom)', size = 16)
plt.xlabel('Winkel' + ' [ $^{\circ}$ ]', size = 12)
plt.ylabel('Zählrate' + ' [ $s^{-1}$ ]', size = 12)
plt.legend(loc = 'upper right')
plt.savefig('images/V255Diagramm1b.png')
plt.show()
```



Peaks

In [5]:

```
# Werte 1. Ordnung
deg_2 = np.loadtxt('data/V255LiF8-11.txt', dtype=str, usecols=[0])
deg_2 = np.char.replace(deg_2, ',', '.').astype(float)

n_2 = np.loadtxt('data/V255LiF8-11.txt', dtype=str, usecols=[1])
n_2 = np.char.replace(n_2, ',', '.').astype(float)
sig_n_2 = np.sqrt(n_2)
```

1. Ordnung

In [6]:

```
# Plot
plt.figure(figsize = (12,7))
plt.errorbar(deg_2, n_2, yerr = sig_n_2, fmt='.', label = 'Messdaten', capsize = 2)
plt.plot(deg_2, n_2, label = 'Referenz')
plt.title('$K_{\alpha}$' + ' und ' + '$K_{\beta}$' + ' Peaks des LiF Spektrums', size
= 16)
plt.xlabel('Winkel  $\beta$  [°]', size = 12)
plt.ylabel('Zählrate [ $s^{-1}$ ]')
plt.legend(loc = 'upper right')
plt.savefig('images/V255Diagramm2.png')
plt.show()

# Lokale Maxima
peaks, _ = find_peaks(n_2, height=250)

n_max_alpha = n_2[peaks[1]]
deg_max_alpha = deg_2[peaks[1]]
n_max_beta = n_2[peaks[0]]
deg_max_beta = deg_2[peaks[0]]

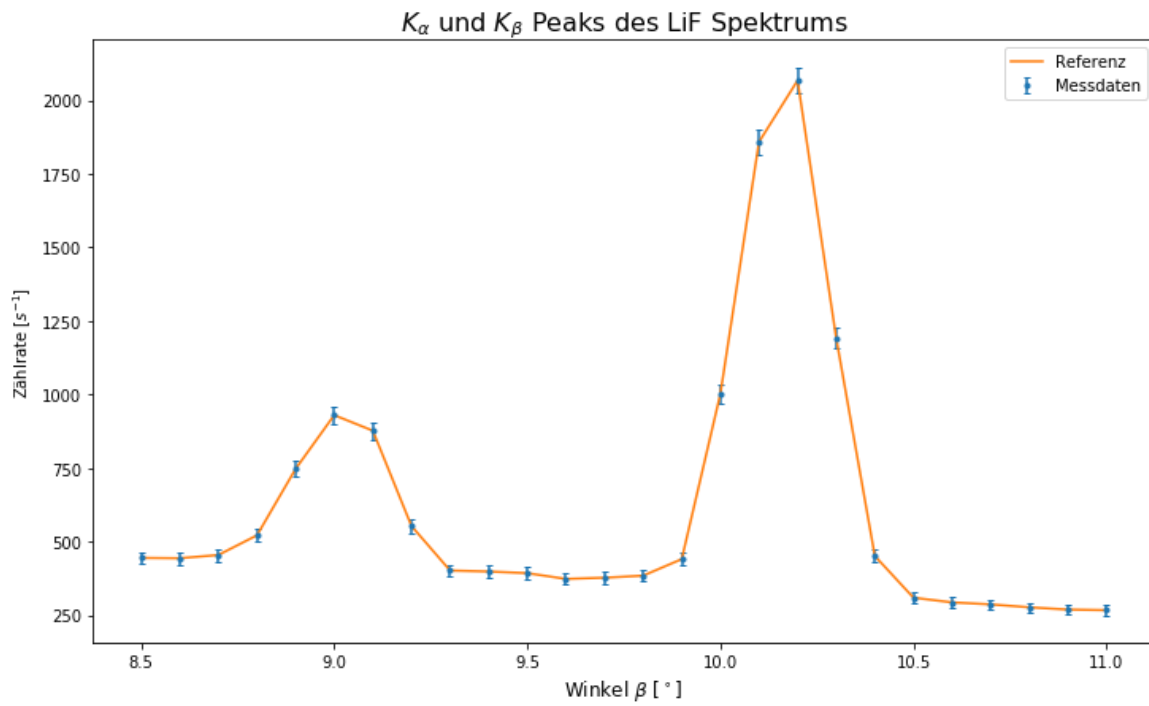
print('\nn_max_alpha = ', n_max_alpha, '\ndeg_max_alpha = ', deg_max_alpha)
print('\nn_max_beta = ', n_max_beta, '\ndeg_max_beta = ', deg_max_beta)

# Wellenlängen
def wlange(deg, d, n):
    return 2 / n * d * np.sin(np.deg2rad(deg))

l_alpha = wlange(deg_max_alpha, d, 1) # d in 10-12 m
l_beta = wlange(deg_max_beta, d, 1) # d in 10-12 m

print('\nl_alpha = ', l_alpha)
print('l_beta = ', l_beta)

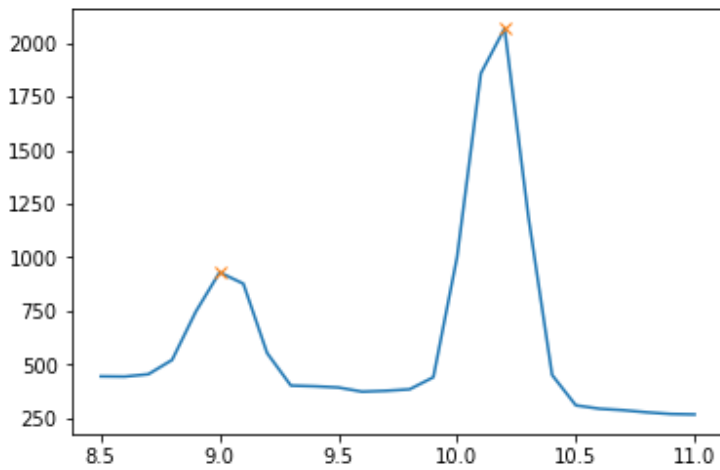
#Sanity check
plt.plot(deg_2, n_2)
plt.plot(deg_2[peaks], n_2[peaks], 'x')
plt.show()
```



```
n_max_alpha = 2067.55  
deg_max_alpha = 10.2
```

```
n_max_beta = 930.05  
deg_max_beta = 9.0
```

```
l_alpha = 7.132973340072815e-11  
l_beta = 6.301180251820499e-11
```



1. Ordnung

In [7]:

```
# Werte 2. Ordnung
deg_3 = np.loadtxt('data/V255LiF17-22.txt', dtype=str, usecols=[0])
deg_3 = np.char.replace(deg_3, ',', '.').astype(float)

n_3 = np.loadtxt('data/V255LiF17-22.txt', dtype=str, usecols=[1])
n_3 = np.char.replace(n_3, ',', '.').astype(float)
sig_n_3 = np.sqrt(n_3)
```

In [8]:

```
# Plot
plt.figure(figsize = (12,7))
plt.errorbar(deg_3, n_3, yerr = sig_n_3, fmt='.', label = 'Messdaten', capsize = 2)
plt.plot(deg_3, n_3, label = 'Referenz')
plt.title('$K_{\alpha}$' + ' und ' + '$K_{\beta}$' + ' Peaks 2. Ordnung des LiF Spekt  
rums', size = 16)
plt.xlabel('Winkel  $\beta$  [°]', size = 12)
plt.ylabel('Zählrate [ $s^{-1}$ ]')
plt.legend(loc = 'upper right')
plt.savefig('images/V255Diagramm3.png')
plt.show()

# Lokale Maxima
peaks, _ = find_peaks(n_3, height=100)

n_max_alpha_2 = n_3[peaks[1]]
deg_max_alpha_2 = deg_3[peaks[1]]
n_max_beta_2 = n_3[peaks[0]]
deg_max_beta_2 = deg_3[peaks[0]]

print('\nn_max_alpha_2 = ', n_max_alpha_2, '\ndeg_max_alpha_2 = ', deg_max_alpha_2)
print('\nn_max_beta_2 = ', n_max_beta_2, '\ndeg_max_beta_2 = ', deg_max_beta_2)

l_alpha_2 = wlange(deg_max_alpha_2, d, 2) # d in 10-12 m
l_beta_2 = wlange(deg_max_beta_2, d, 2) # d in 10-12 m

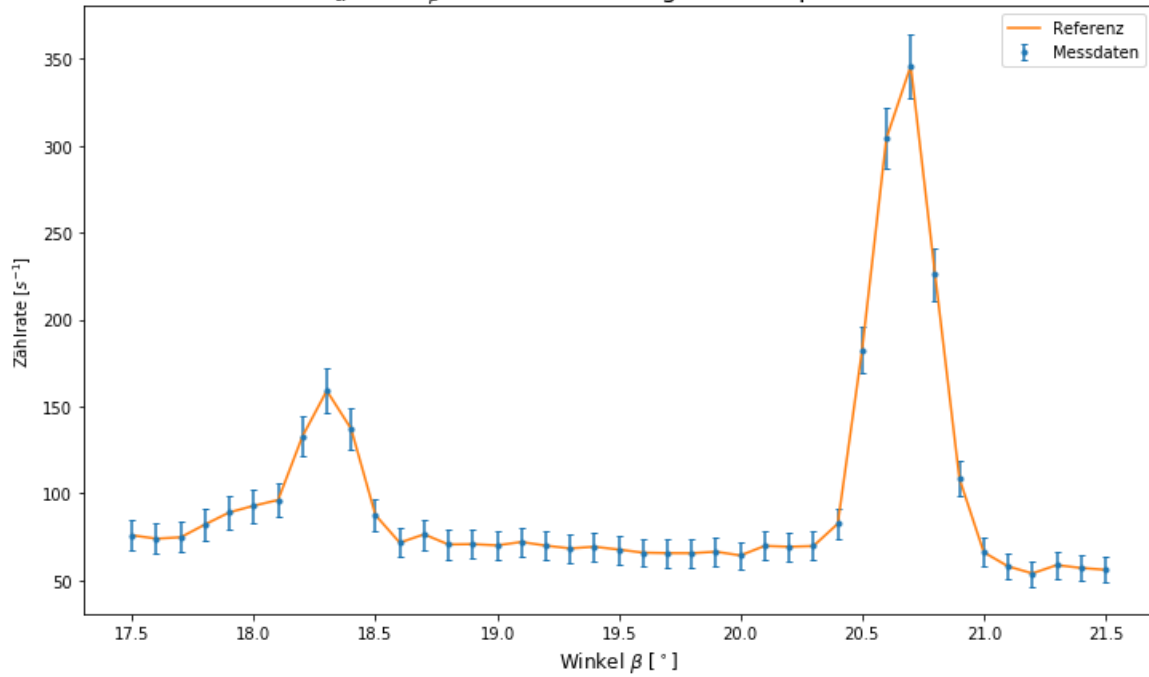
print('\nl_alpha_2 = ', l_alpha_2)
print('l_beta_2 = ', l_beta_2)

# Sanity check
plt.plot(deg_3, n_3)
plt.plot(deg_3[peaks], n_3[peaks], 'x')
plt.show()

# Durchschnitt
Mo_l_alpha = (l_alpha + l_alpha_2) / 2
Mo_l_beta = (l_beta + l_beta_2) / 2

print('Durchschnitt:\n')
print('Mo_l_alpha = ', Mo_l_alpha)
print('Mo_l_beta = ', Mo_l_beta)
```

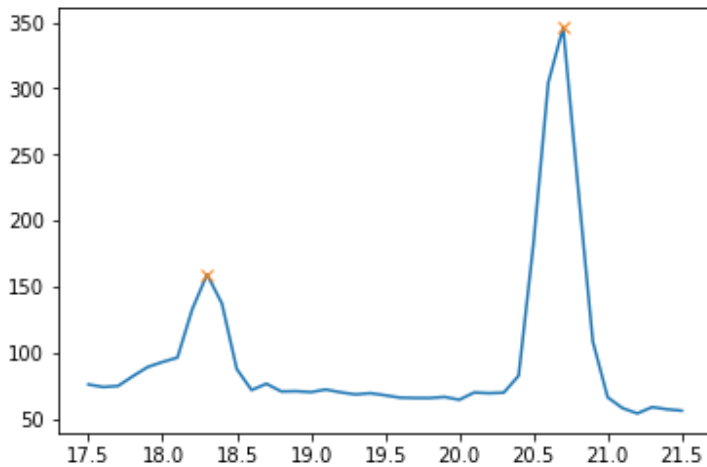
K_α und K_β Peaks 2. Ordnung des LiF Spektrums



n_max_alpha_2 = 345.95
deg_max_alpha_2 = 20.7

n_max_beta_2 = 159.05
deg_max_beta_2 = 18.3

l_alpha_2 = 7.118983353714239e-11
l_beta_2 = 6.323808063183535e-11



Durchschnitt:

Mo_l_alpha = 7.125978346893526e-11

Mo_l_beta = 6.312494157502017e-11

Halbwertsbreite

In [9]:

```
# Halbwertsbreite 1. Ordnung
from numpy import exp, pi, sqrt

def gauss(x, A, mu, sig):
    return A / (sqrt(2 * pi) * sig) * exp(- (x - mu) ** 2 / (2 * sig ** 2)) + 250

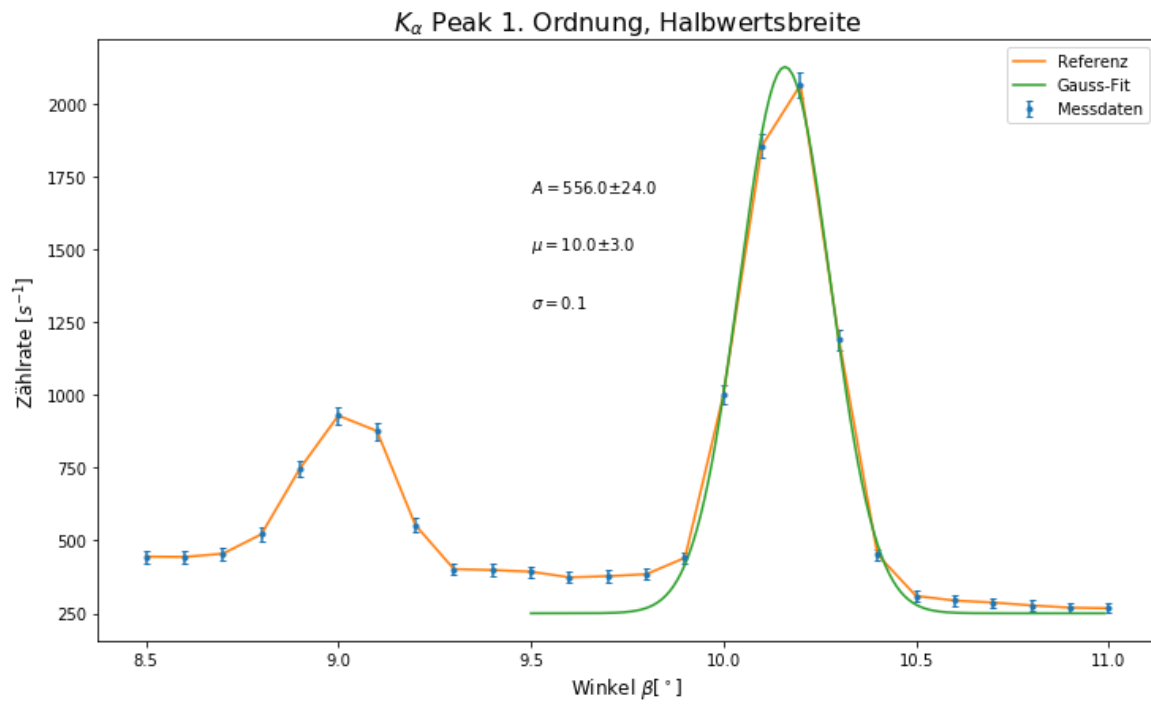
p0 = [500, 10, 0.4]
popt, pcov = curve_fit(gauss, deg_2[14:-4], n_2[14:-4], p0, sigma = sig_n_2[14:-4])

degx = np.arange(9.5, 11, 0.01)

plt.figure(figsize = (12, 7))
plt.title('$K_{\alpha}$ Peak 1. Ordnung, Halbwertsbreite', size = 16)
plt.xlabel('Winkel $\beta$ [^\circ]', size = 12)
plt.ylabel('Zählrate [$s^{-1}$]', size = 12)
plt.errorbar(deg_2, n_2, yerr = sig_n_2, label = 'Messdaten', fmt = '.', capsize = 2)
plt.plot(deg_2, n_2, label = 'Referenz')
plt.plot(degx, gauss(degx, *popt), label = 'Gauss-Fit')
plt.text(9.5, 1700, '$A = $' + str(np.round(popt[0], 0)) + '$\pm$' + str(np.round(np.sqrt(popt[0]), 0)))
plt.text(9.5, 1500, '$\mu = $' + str(np.round(popt[1], 0)) + '$\pm$' + str(np.round(np.sqrt(popt[1]), 0)))
plt.text(9.5, 1300, '$\sigma = $' + str(np.round(popt[2], 1)))
plt.legend(loc = 'best')
plt.savefig('images/V255Diagramm5.png')
plt.show()

y = gauss(degx, *popt)
beta_halb1 = popt[1] - np.sqrt(2 * np.log(2)) * popt[2]
beta_halb2 = popt[1] + np.sqrt(2 * np.log(2)) * popt[2]
print('Winkel beta_1/2 = ', popt[1])
print('Winkel beta_halb1 = ', beta_halb1)
print('Winkel beta_halb2 = ', beta_halb2)

lambda_halb1 = wlange(beta_halb1, d, 1)
lambda_halb2 = wlange(beta_halb2, d, 1)
print('Halbwertsbreite = ', np.abs(lambda_halb2 - lambda_halb1))
```



```
Winkel beta_1/2 = 10.159666873946383
Winkel beta_halb1 = 10.02061545650315
Winkel beta_halb2 = 10.298718291389617
Halbwertsbreite = 1.92445667493146e-12
```

Einsatzspannung

In [10]:

```
# Werte Tabelle
U = np.arange(20, 36, 1)
dU = np.ones(16)
n_U = np.array([1.2, 2.0, 2.0, 5.0, 45.55, 107.2, 157.2, 201.5, 240.1, 285.1, 329.5,
, 366.4, 404.6, 441.1, 487.8, 520.5])

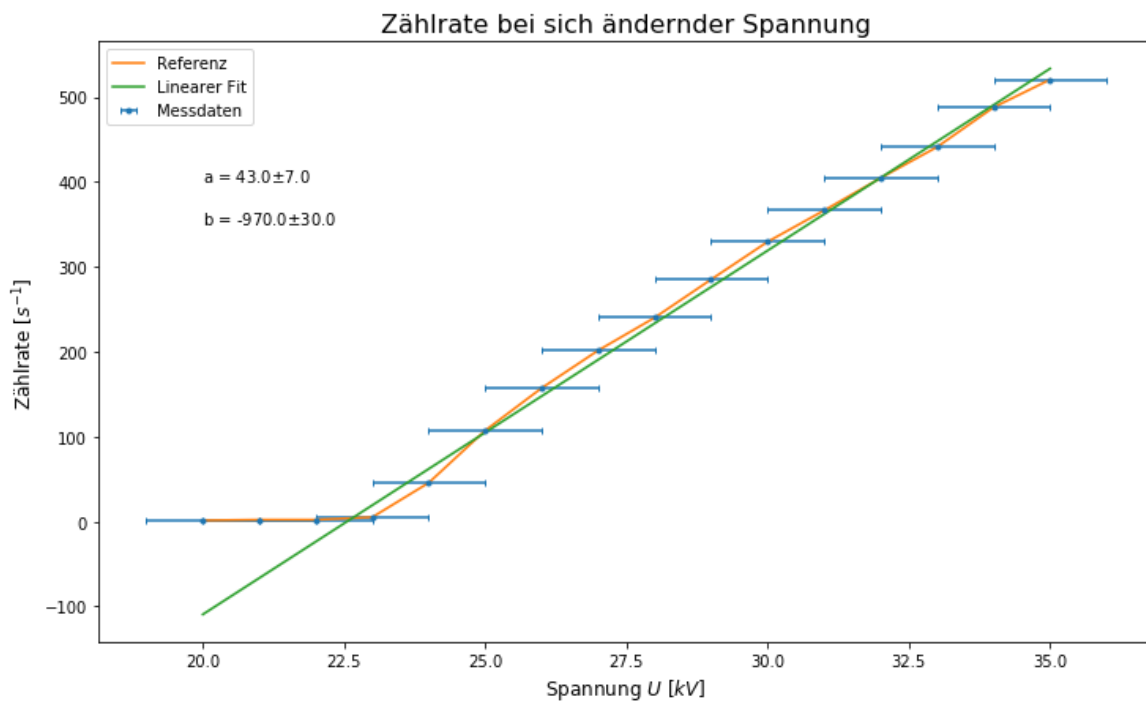
# Fit
popt, pcov = curve_fit(line, U[3:], n_U[3:])
x = np.arange(20, 36, 1)

# Plot
plt.figure(figsize = (12, 7))
plt.title('Zählrate bei sich ändernder Spannung', size = 16)
plt.xlabel('Spannung $U$ [$kV$]', size = 12)
plt.ylabel('Zählrate [$s^{-1}$]', size = 12)
plt.errorbar(U, n_U, xerr = dU, label = 'Messdaten', fmt = '.', capsize = 2)
plt.plot(U, n_U, label = 'Referenz')
plt.plot(x, line(x, *popt), label = 'Linearer Fit')
plt.text(20, 400, 'a = ' + str(np.round(popt[0],0)) + '$\pm$' + str(np.round(sqrt(p
opt[0]),0)))
plt.text(20, 350, 'b = ' + str(np.round(popt[1],-1)) + '$\pm$' + str(np.round(sqrt(
-popt[1]),-1)))
plt.legend(loc = 'best')
plt.savefig('images/V255Diagramm4.png')
plt.show()

# Einsatzspannung
val = np.arange(0, 100, 0.01)
for i in range(0, 10000):
    if line(val[i], *popt) > 0:
        U_E = val[i] #kV
        break

print('Einsatzspannung U_E =', U_E)

# Planck Konstante
h = (wlange(7.5, d, 1)) * e * U_E / c
print('Planck-Konstante h =', h)
```



Einsatzspannung $U_E = 22.56$

Planck-Konstante $h = 6.334636496426238e-37$

Röntgenspektrum am NaCl Kristall

In [11]:

```
# Werte 2. Ordnung
deg = np.loadtxt('data/V255NaCl3-18.txt', dtype=str, usecols=[0])
deg = np.char.replace(deg, ',', '.').astype(float)

n = np.loadtxt('data/V255NaCl3-18.txt', dtype=str, usecols=[1])
n = np.char.replace(n, ',', '.').astype(float)
sig_n = np.sqrt(n)
```

In [12]:

```
# Plot
plt.figure(figsize = (12,7))
plt.errorbar(deg, n, yerr = sig_n, fmt='.', label = 'Messdaten', capsize = 2)
plt.plot(deg, n, label = 'Referenz')
plt.title('Röntgenspektrum NaCl', size = 16)
plt.xlabel('Winkel  $\beta$  [°]', size = 12)
plt.ylabel('Zählrate [ $s^{-1}$ ]\n')
plt.legend(loc = 'upper right')
plt.savefig('images/V255Diagramm6.png')
plt.show()

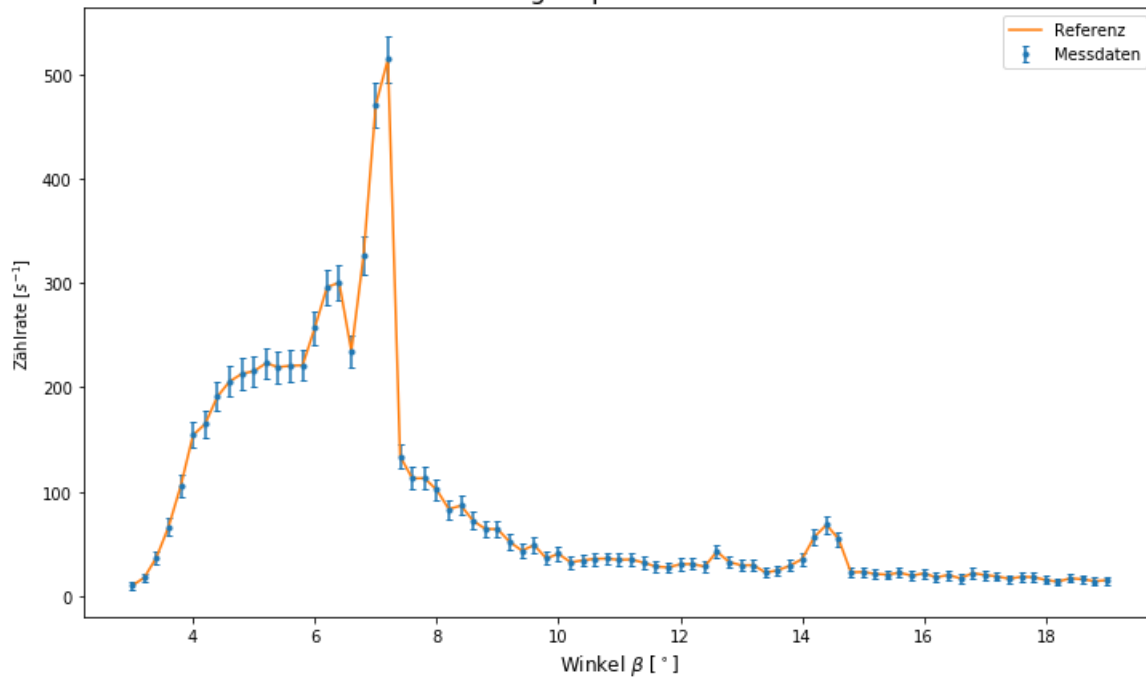
# Lokale Maxima 1. Ordnung
peaks, _ = find_peaks(n, height=250)

n_max_alpha = n[peaks[1]]
deg_max_alpha = deg[peaks[1]]
n_max_beta = n[peaks[0]]
deg_max_beta = deg[peaks[0]]

print('\nn_max_alpha = ', n_max_alpha, '\ndeg_max_alpha = ', deg_max_alpha)
print('\nn_max_beta = ', n_max_beta, '\ndeg_max_beta = ', deg_max_beta)

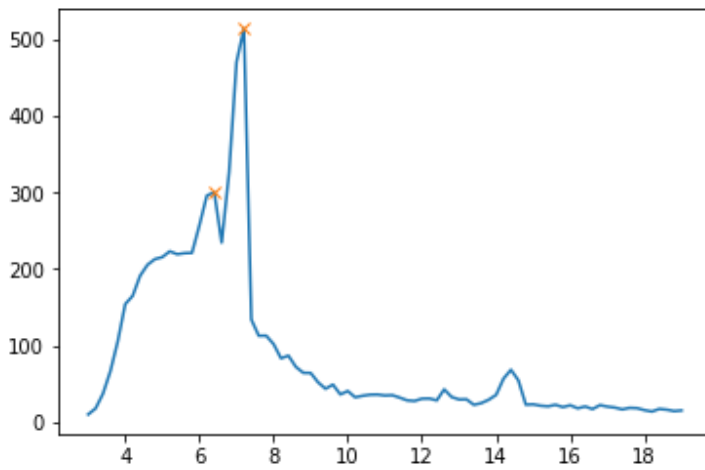
#Sanity check
plt.plot(deg,n)
plt.plot(deg[peaks], n[peaks], 'x')
plt.show()
```

Röntgenspektrum NaCl



n_max_alpha = 514.2
deg_max_alpha = 7.2

n_max_beta = 300.2
deg_max_beta = 6.4



In [13]:

```
# Lokale Maxima 2. Ordnung
deg_2 = deg[45:]
n_2 = n[45:]
peaks, _ = find_peaks(n_2, height=35)

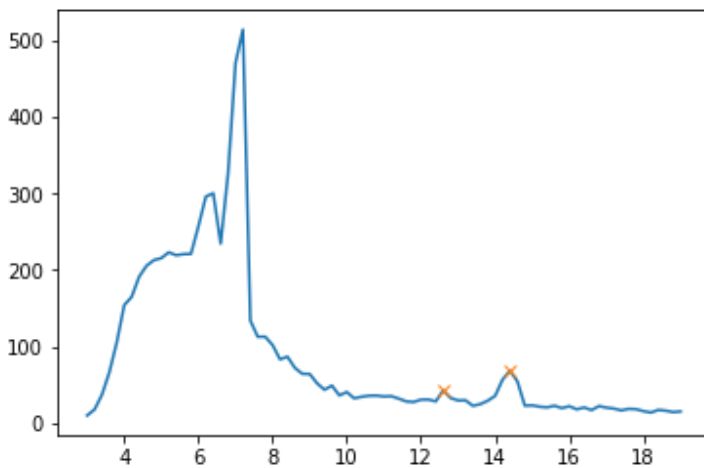
n_max_alpha_2 = n_2[peaks[1]]
deg_max_alpha_2 = deg_2[peaks[1]]
n_max_beta_2 = n_2[peaks[0]]
deg_max_beta_2 = deg_2[peaks[0]]

print('\nn_max_alpha_2 = ', n_max_alpha_2, '\ndeg_max_alpha_2 = ',
      deg_max_alpha_2)
print('\nn_max_beta_2 = ', n_max_beta_2, '\ndeg_max_beta_2 = ',
      deg_max_beta_2)

#Sanity check
plt.plot(deg, n)
plt.plot(deg_2[peaks], n_2[peaks], 'x')
plt.show()
```

```
n_max_alpha_2 = 68.2
deg_max_alpha_2 = 14.4
```

```
n_max_beta_2 = 42.6
deg_max_beta_2 = 12.6
```



In [16]:

```
# Gitterkonstanten
def gitterkonst(deg, l_peak, n):
    return n * l_peak / (np.sin(np.deg2rad(deg)))

da1 = gitterkonst(deg_max_alpha, Mo_l_alpha, 1)
db1 = gitterkonst(deg_max_beta, Mo_l_beta, 1)
da2 = gitterkonst(deg_max_alpha_2, Mo_l_alpha, 2)
db2 = gitterkonst(deg_max_beta_2, Mo_l_beta, 2)
dlist = np.array([da1, db1, da2, db2])

dg = np.mean(dlist)
sig_dg = np.std(dlist)

print('Mittelwert der Gitterkonstante: dg = ' + str(dg))
print('Fehler des Mittelwerts: sig_dg = ' + str(sig_dg))
```

Mittelwert der Gitterkonstante: dg = 5.716731048233624e-10
Fehler des Mittelwerts: sig_dg = 4.7584113304708365e-12

In [15]:

```
# Avogadro Konstante
Mmol = 58.44 # g mol^-1
rho = 2.164 # g cm^-3

NA = 4 * Mmol / (rho * dg ** 3)
print(NA)
```

5.7818878044529996e+29

