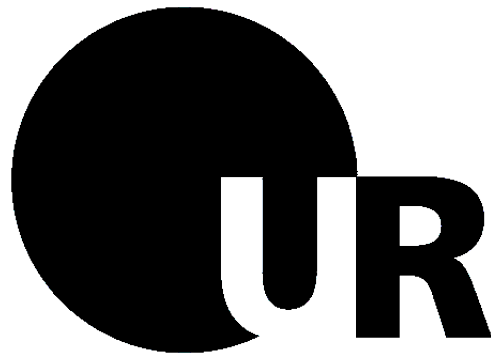


# Entwicklung von Komponenten für einen Chiral - STEM Betrieb



Diplomarbeit  
von

**Andreas Hasenkopf**  
aus Bayerisch Eisenstein

durchgeführt am

Institut für Experimentelle und Angewandte Physik  
Universität Regensburg

unter der Anleitung von  
Prof. Dr. Josef Zweck

September 2008

# Entwicklung von Komponenten für einen Chiral - STEM Betrieb



Andreas Hasenkopf



---

## Zusammenfassung

Der Effekt des magnetischen Chiraldichroismus bei inelastischer Elektronenstreuung (EMCD) wurde von Schattschneider et. al. in Analogie zu XMCD postuliert aufgrund Ähnlichkeiten in den Streuquerschnitten. Dieser Effekt wurde erstmals von Rubino beobachtet.

Die momentane Methode zur Aufnahme von EMCD-Spektren wird als „intrinsische Methode“ bezeichnet. Dafür müssen drei exakte Bedingungen (Monokristall, Dicke, Ausrichtung) durch die Probe erfüllt werden aufgrund der Verwendung der Probe als Strahlteiler. Um diese Bedingungen zu umgehen wird eine Doppellochblende in Kombination mit einer Boersch-Phasenplatte vorgeschlagen um zwei kohärente Elektronenwellen mit einer Phasendifferenz von  $\pi/2$  zu erzeugen. Chiraler Kontrast wird erzielt aufgrund der Verwendung einer magnetischen Probe.

Um zu untersuchen, ob eine Doppellochblende für EMCD-Messungen – insbesondere im STEM-Betrieb – geeignet ist, werden die Wellenfunktion und die Intensitätsverteilung in der Probenebene numerisch berechnet und die optimalen Parameter zum Bau einer Doppellochblende und Phasenplatte bestimmt. Abschließend wird eine Doppellochblende im STEM-Betrieb getestet.

## Summary

The effect of Energy-loss Magnetic Chiral Dichroism (EMCD) was proposed by Schattschneider et al. in analogy to XMCD due to similar scattering cross section expressions. The effect was measured for the first time by Rubino.

The state of the art method for recording EMCD-spectra is the "intrinsic method" requiring the specimen to fulfill three strict properties (mono crystal, thickness, orientation) due to its use as a beam splitter. In order to avoid these requirements an alternative setup with a twin aperture and Boersch phase plate for generating two coherent electron waves with a phase difference of  $\pi/2$  is suggested. Chiral contrast in energy-loss spectra is achieved due to a magnetic sample.

To determine whether a twin aperture is suitable for EMCD-measurements – in particular in STEM-mode – the wave function and intensity distribution in the sample plane are calculated numerically and the optimum dimensions for the aperture and phase plate are determined. Finally a STEM operation with a twin aperture is realized.



---

---

## Inhaltsverzeichnis

<b>Zusammenfassung</b>	<b>3</b>
<b>Inhaltsverzeichnis</b>	<b>5</b>
<b>1 Einführung</b>	<b>7</b>
<b>2 Elektronenmikroskopie</b>	<b>11</b>
2.1 Transmissionselektronenmikroskopie . . . . .	11
2.2 Elektronen-Energieverlust-Spektrometrie . . . . .	12
2.2.1 Ein Spektrum . . . . .	13
2.2.2 Das Spektrometer . . . . .	14
2.3 Rasterelektronenmikroskopie . . . . .	16
2.4 Rastertransmissionselektronenmikroskopie . . . . .	17
2.4.1 Das Reziprozitätsprinzip . . . . .	17
2.4.2 Aufbau und Funktion . . . . .	18
<b>3 Zirkulardichroismus</b>	<b>20</b>
3.1 Dichroismus und Chiralität . . . . .	20
3.2 Chiralität und Zirkulardichroismus . . . . .	21
3.3 Röntgenzirkulardichroismus . . . . .	22
3.4 Chiraldichroismus . . . . .	25
3.5 Analogie Photonen – Elektronen . . . . .	29
<b>4 Zum „2 Blenden“-Betrieb</b>	<b>33</b>
<b>5 Grundlagen – Fourieroptik und Phasenplatte</b>	<b>37</b>
5.1 Fourieroptik . . . . .	37
5.2 Phasenplatte . . . . .	41
5.2.1 Prinzip der Potentialbarriere . . . . .	42
5.2.2 Einzellinse . . . . .	43
5.2.3 Konstanz der Phasenverschiebung . . . . .	44
<b>6 Simulationen</b>	<b>52</b>
6.1 Mathematische Beschreibung . . . . .	52
6.2 Wellenfunktion in BFP . . . . .	57
6.3 Propagierte Wellenfunktion . . . . .	59
6.4 Variation der Parameter . . . . .	60
<b>7 Experimente</b>	<b>69</b>
7.1 Test der Kohärenzlänge . . . . .	69
7.2 Herstellung . . . . .	72
7.2.1 Doppellochblende . . . . .	73
7.2.2 Phasenplatte . . . . .	74

## INHALTSVERZEICHNIS

---

7.3	STEM-Betrieb mit Doppellochblende . . . . .	77
<b>8</b>	<b>Ausblick</b>	<b>81</b>
<b>A</b>	<b>STEM - Detektoren</b>	<b>83</b>
<b>B</b>	<b>Fouriertransformation und Faltung</b>	<b>85</b>
B.1	Fouriertransformation . . . . .	85
B.2	Faltung . . . . .	87
B.3	Faltungstheorem . . . . .	87
<b>C</b>	<b>Software</b>	<b>89</b>
C.1	Python . . . . .	89
C.2	MySQL . . . . .	90
<b>D</b>	<b>Programme</b>	<b>91</b>
D.1	Anmerkungen . . . . .	91
D.1.1	plot_interface.py . . . . .	91
D.1.2	bruteforce.py . . . . .	94
D.1.3	sql1.py und sql2.py . . . . .	95
D.2	Quelltext . . . . .	96
	<b>Abbildungsverzeichnis</b>	<b>129</b>
	<b>Literatur</b>	<b>130</b>

## 1 Einführung

Nach der Formulierung des Welle-Teilchen-Dualismus im Jahre 1924 durch De Broglie [Bro24] wurden die typischen Eigenschaften elektromagnetischer Wellen auch für Elektronen – und andere Partikel, die mit einer Ruhemasse behaftet sind – nachgewiesen, z.B. die Möglichkeit zur Interferenz durch Möllenstedt und Düker (1955)[MD56].

Chirale Kristalle absorbieren Licht mit unterschiedlicher Polarisierung unterschiedlich stark. Diese Eigenschaft wird in der Optik als Dichroismus bezeichnet. Dieses dichroische Verhalten lässt sich z.B. in Eisen, Nickel und Cobalt durch Anlegen eines magnetischen Feldes hervor rufen. Ein Maß für den Dichroismus erhält man aus Röntgenabsorptionsspektren (XAS – „X-ray absorption spectrometry“, siehe Abbildung 1(a): Spektrum von Eisen), wenn man diese für die unterschiedlichen Polarisierungen (z.B. links- und rechts-zirkular) oder entgegengesetzte Magnetisierungsrichtungen aufnimmt und die Differenz daraus nimmt. Dies wird als magnetischer Röntgenzirkulardichroismus (XMCD – „X-ray magnetic circular dichroism“) bezeichnet im Falle von unterschiedlichen zirkularen Polarisierungen.

In Analogie zu diesem Effekt wurde von P. Schattschneider et al. [HS03] ein ähnlicher Effekt für Elektronen postuliert: der magnetische Chiraldichroismus (EMCD – „Energy-loss magnetic chiral dichroism“), welcher bei inelastischer Elektronenstreuung beobachtet werden kann.

Die Elektronen wechselwirken beim Passieren der Probe mit den in der Probe gebundenen Elektronen (und Kernen) und werden aufgrund dieser Wechselwirkung von ihrer Bahn abgelenkt. Dies wird als Streuung bezeichnet. Bei dieser Wechselwirkung kann zwischen einem schnellen Elektron des Strahls und einem gebundenen Elektron in der Probe Energie übertragen werden. Mittels eines Spektrometers lässt sich die Änderung in der Energie der schnellen Elektronen in einem sogenannten Elektronen-Energieverlust-Spektrum (EELS – „Electron energy loss spectrometry“) darstellen.

Da für Elektronen im Gegensatz zu Photonen keine Polarisierung definiert ist, betrachtet man an Stelle der Polarisierung den Impulsübertrag während eines inelastischen Streuprozesses. Diese Analogie stammt aus einer theoretischen Betrachtung der Streuquerschnitte für Röntgenabsorption und inelastische Elektronenstreuung. Die Streuquerschnitte in diesen beiden Fällen weisen eine sehr ähnliche Form auf, wenn man den Impulsübertrag als Analogon zur Polarisierung betrachtet. Auf diese Betrachtung wird in Kapitel 3 näher eingegangen.

In der Mathematik lässt sich eine zirkular polarisierte Welle beschreiben als eine Superposition aus zwei linear polarisierten Wellen. Dafür müssen die Polarisationsvektoren der beiden linear polarisierten Wellen senkrecht aufeinander stehen und eine Phasendifferenz von  $\pi/2$  aufweisen (siehe Ab-

bildung 10(b) auf Seite 25).

In Analogie dazu betrachtet man im Falle der Elektronenstreuung zwei kohärente Teilwellen, deren Impulsüberträge diese Eigenschaften (senkrecht zueinander, phasenverschoben) erfüllen. In der Praxis werden diese durch Beugung an der Probe erzeugt. Man sagt, die Probe dient als Strahlteiler. Diese Methode wird als intrinsische Methode bezeichnet. Damit dies funktioniert muss die Probe die richtige Dicke und Kristallorientierung aufweisen und monokristallin sein. Nimmt man in der Beobachtungsebene an zwei geeigneten Stellen (siehe Kapitel 3.4) Energieverlustspektren auf und betrachtet deren Differenz erhält man ein Spektrum wie in Abbildung 1(b).

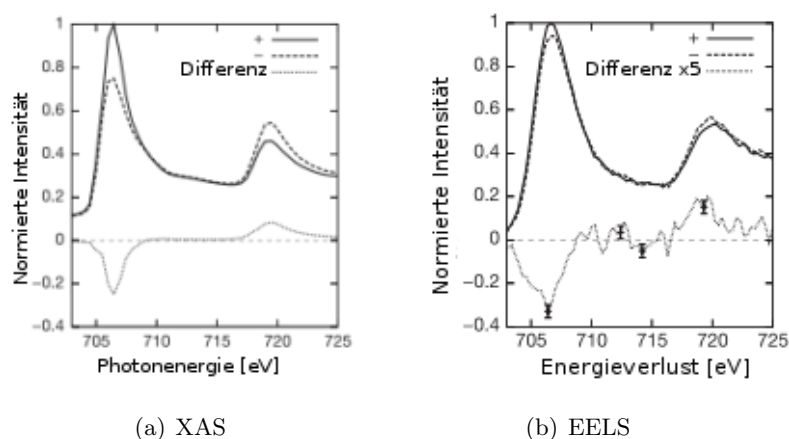


Abbildung 1: Vergleich zwischen Röntgenabsorptions- (a) und Elektronenenergieverlust-Spektrum (b) an der  $L_{2,3}$ -Kante von Eisen. Quelle: [SRH<sup>+</sup>06]

Abbildung 1 zeigt die Spektren an der  $L_{2,3}$ -Kante von Eisen. Abbildung 1(a) zeigt die Röntgenabsorptions-Spektren (XAS) und das Differenzsignal (XMCD) und Abbildung 1(b) zeigt die Elektronenenergieverlust-Spektren (EELS) und das Differenzsignal (EMCD). Ein solches Spektrum wurde erstmals von S. Rubino [SRH<sup>+</sup>06] in einem Transmissionselektronenmikroskop (TEM) aufgenommen.

Diese Arbeit ist motiviert durch die Frage, ob solche EMCD-Messungen in einer vorteilhafteren Methode durchführbar sind. Ein wesentlicher Nachteil der bekannten intrinsischen Methode ist durch die Verwendung der Probe als Strahlteiler gegeben, da dafür eine monokristalline Probe auf eine exakte Dicke präpariert und im TEM exakt ausgerichtet werden muss.

Daher soll in dieser Arbeit eine Möglichkeit untersucht werden, besagte zwei Teilstrahlen bereits vor der Probe durch zwei Kondensorblenden – diese Anordnung wird im folgenden als Doppellochblende bezeichnet – zu

erzeugen, so dass keine Strahlteilung durch die Probe nötig ist. Erwies sich dies als möglich, ist keine exakte Ausrichtung und Präparation auf eine exakte Dicke zur Erzeugung zweier Teilstrahlen mit einer Phasendifferenz von  $\pi/2$  in der Probenebene nötig. Eine solche Phasendifferenz kann durch eine sogenannte Phasenplatte, wie sie in Kapitel 5.2 beschrieben wird, eingeführt werden.

Der erste Schritt zur Untersuchung dieser Möglichkeit ist die numerische Berechnung der Intensitätsverteilung in der Probenebene. Dabei soll der Effekt einer Doppellochblende und einer Phasenplatte auf die Intensitätsverteilung untersucht werden. Anhand dieser Ergebnisse wird für den Bau einer solchen Blende der optimale Satz an Parametern (Abstand, Radien, Phasenschiebung) bestimmt.

Bei der gängigen, intrinsischen Methode ist aufgrund der notwendigen exakten Ausrichtung der Probe fraglich, ob Messungen in einem STEM-Betrieb möglich sind, da die geforderten Bedingungen zur Erzeugung der Phasendifferenz nicht notwendigerweise an mehreren Punkten erfüllt sind. Aufgrund des in dieser Arbeit präsentierten Aufbaus – Doppellochblende und Phasenplatte – stellt sich die Frage, ob diese Bedingungen (siehe Kapitel 3.4) bei Verschiebung des fokussierten Strahles in der Probenebene erfüllt bleiben und somit EMCD-Messungen in einem „Scanning“-Modus durchführbar sind.

Zur Untersuchung dieser Möglichkeit sollen in dieser Arbeit bei Verwendung einer solchen Doppellochblende als Kondensorblende das Mikroskop im Scanning-Modus betrieben und Abbildungen geeigneter Proben gemacht werden.

In dieser Arbeit stehen folgende vier Fragen im Vordergrund:

- Können mit einer Doppellochblende und einer Phasenplatte zwei Teilstrahlen derart erzeugt werden, dass ein Phasenunterschied von  $\pi/2$  in der Probenebene an einem Messpunkt (Maximum der Wellenfunktion) gegeben ist?
- Welche Blenden- und Phasenplattenparameter werden dafür benötigt?
- Existieren Beschränkungen aufgrund der lateralen Kohärenz des Elektronenstrahls?
- Ist mit einer solchen Anordnung ein STEM-Betrieb möglich? Und was ist die dabei erzielbare Auflösung?

Um diese Fragen im Detail zu beleuchten, sollen die Kapitel 2 und 3 einen Einblick in die für diese Arbeit nötigen Grundlagen liefern: Kapi-

tel 3 führt in die Terminologie des Dichroismus und der Chiralität ein. Des weiteren werden die Grundlagen von EMCD und dem analogen Effekt für Photonen (XMCD – „X-ray Magnetic Circular Dichroism“) präsentiert. Der prinzipielle Aufbau und die Funktionsweise von Elektronenmikroskopen (Transmissions- und Rastertransmission-) werden in Kapitel 2 erklärt. Dabei wird auch auf eine wichtige Detektionsmethode eingegangen: die Elektronen-Energieverlust-Spektrometrie.

In Kapitel 4 wird ausgehend von der Motivation den experimentellen Aufbau für EMCD-Messungen zu verbessern, die Idee zur Verwendung einer Doppellochblende eingeführt. Dabei wird auf die erhofften Verbesserungen gegenüber der bisher gängigen intrinsischen Methode eingegangen und der dazu konzipierte experimentelle Aufbau schematisch erklärt. Dazu wird dieser dem schematischen Aufbau für die intrinsische Methode gegenüber gestellt.

Für die anschließenden numerischen Simulationen präsentiert Kapitel 5 die wichtigsten Grundlagen der Wellenoptik für Elektronen. Das Augenmerk liegt dabei auf den Fragen, wie sich eine Welle entlang der optischen Achse ausbreitet und wie die Wirkung von Linsen in der Elektronenoptik beschrieben wird. Ergänzt wird dieser Abschnitt durch Anhang B, in dem wichtige mathematische Hilfsmittel (Fouriertransformation und Faltung) zur Verfügung gestellt werden. Abgeschlossen wird dieses Kapitel durch die Präsentation einer Phasenplatte sowie der Beschreibung ihrer Wirkungsweise.

In Kapitel 6 werden die physikalischen Zusammenhänge und Größen, die für die Simulationen von Bedeutung sind, aus der Wellenoptik hergeleitet und neue Größen definiert, welche im Rahmen der Simulationen dazu verwendet werden, um die Nutzbarkeit der Kombinationen aus Blenden- und Phasenplattenparameter abzuschätzen. Des weiteren werden die Ergebnisse der Simulationen sowie deren Auswertungen präsentiert. Im Anhang D befindet sich der Quelltext der vier verwendeten Programme, welche in der Programmiersprache *Python* geschrieben wurden. Jedes Programmskript wird im Anhang ebenfalls in kompakter Form beschrieben.

Abschließend wird in Kapitel 7 abgeschätzt, ob die Kohärenzlängen der zur Verfügung stehenden Mikroskope ausreichen um die Doppellochblende kohärent zu beleuchten. Anhand der Ergebnisse der Simulationen in Kapitel 6 wird eine Doppellochblende produziert und in das TEM als Kondensorblende eingebaut. Mit diesem Aufbau wird im STEM-Betrieb eine geeignete Probe betrachtet. Diese Aufnahmen werden verglichen mit Bildern, welche bei Verwendung einer normalen kreisrunden Kondensorblende aufgenommen wurden.

## 2 Elektronenmikroskopie

Ein Elektronenmikroskop ist ein Mikroskop, das das Innere oder die Oberfläche einer Probe mit Elektronen abbilden kann. Informationen über die Oberfläche gewinnt man mittels eines Rasterelektronenmikroskops, während man für Untersuchungen des Inneren einer Probe ein Transmissionselektronenmikroskop verwendet.

Teilchen	Energie	Wellenlänge
Photon	ca. 1.6 - 3.3eV (sichtbares Licht)	ca. 400 - 700nm
Photon	ca. 100 - 250eV (Röntgen)	ca. 0.01 - 15nm
Elektron	E = 100keV	ca. 4pm
Elektron	E = 300keV	ca. 2pm

Tabelle 1: Wellenlängen von Elektronen und Photonen

Tabelle 1 zeigt für einige Photonen und Elektronen unterschiedlicher Energien die Wellenlängen bzw. De Broglie Wellenlängen.

Da schnelle Elektronen eine sehr viel kleinere Wellenlänge als sichtbares Licht haben und die maximal erzielbare Auflösung eines Mikroskops durch die Wellenlänge begrenzt ist, kann mit einem Elektronenmikroskop eine deutlich höhere Auflösung erreicht werden als mit einem Lichtmikroskop (vergleiche Tabelle 1).

Während bei optischen Mikroskopen die Auflösung tatsächlich nahezu die von der Lichtwellenlänge gesetzte physikalische Grenze erreicht, wird diese Grenze bei Elektronenmikroskopen aufgrund der Aberrationen der elektronenoptischen Bauteile nicht erreicht. Diese Aberrationen sind aber zu einem gewissen Grad notwendig, da ohne Aberrationen kein Kontrast in einem Transmissionselektronenmikroskop erzielt wird. Dies liegt daran, dass die Kontrasttransferfunktion unter anderem von  $C_s$ , dem Koeffizienten der sphärischen Abberation, abhängt. Eine genauere Beschreibung dieses Umstandes kann der Literatur zur Elektronenmikroskopie, z.B. [Spe03, WC96], entnommen werden.

### 2.1 Transmissionselektronenmikroskopie

Ein Transmissionselektronenmikroskop (TEM) ist eine Variante eines Elektronenmikroskops, bei welchem ein Strahl von Elektronen mit einer dünnen



Probe wechselwirkt, während der Strahl die Probe durchdringt. Im einfachsten Fall wird das Bild aus den transmittierten Elektronen durch Vergrößerung und Fokussierung durch eine Objektivlinse auf dem Beobachtungsschirm erzeugt.

Das erste in der Praxis anwendbare Transmissionselektronenmikroskop wurde gebaut von A. Prebus und J. Hillier an der Universität von Toronto (1938) unter Verwendung von Konzepten, welche M. Knoll und E. Ruska bei der Entwicklung des ersten TEM aufstellten [KR32]. E. Ruska erhielt 1986 für seine fundamentale Arbeit auf dem Gebiet der Elektronenoptik und den Bau des ersten Elektronenmikroskops den Nobelpreis für Physik.

Im Aufbau ist das Transmissionselektronenmikroskop einem Durchlichtmikroskop nachempfunden, d.h. es besteht aus hintereinander geschalteten vergrößernden Linsen. Das TEM erzeugt ein Durchlicht-Elektronenbild mit einer Vergrößerung bis  $10^6$  und größer. Das Auflösungsvermögen beträgt in etwa 0.2nm, kann aber mittels Aberrationskorrektoren auf unter 0.1nm gesenkt werden. So lassen sich beispielsweise Netzebenen von Kristallgittern wie z.B. von Graphit abbilden.

Zur Erzeugung des Elektronenstrahls verwendet man eine Haarnadelkathode aus Wolfram, eine Kristallkathode aus Lanthanhexaborid ( $\text{LaB}_6$ ) oder eine Feldemissionskathode (FEG – „Field Emission Gun“) [CIJ69]. Die Elektronen werden nach der Erzeugung durch einen sogenannten Wehnelt-Zylinder gebündelt und in Richtung Anode beschleunigt (Beschleunigungsspannungen: zwischen 50 und 300kV am CM30). Da allerdings die Anwesenheit reaktiver Gase (Luftsauerstoff) zu Wechselwirkungen mit dem Elektronenemitter und den Strahlelektronen führen würde wird im Elektronenmikroskop ein Vakuum benötigt. Linsen in der Elektronenoptik nutzen zur Lenkung und Bündelung des Elektronenstrahls inhomogene magnetische (oder elektrische) Felder.

## 2.2 Elektronen-Energieverlust-Spektrometrie

In der Elektronen-Energieverlust-Spektroskopie (EELS – „Electron Energy loss spectroscopy“) wird eine Probe mit einem Elektronenstrahl mit einem bekannten, schmalen Band an kinetischer Energie durchstrahlt. Einige der Elektronen werden dabei inelastisch gestreut. Die Energieänderung kann mittels eines Spektrometers gemessen werden. Inelastische Wechselwirkungen umfassen unter anderem neben Phononen- und Plasmonenanregungen auch Inter- und Intrabandübergänge.

Erste Messungen der Geschwindigkeitsverteilung von Elektronen, welche sich durch eine Membran aus Kollodium ausbreiteten, wurden 1939 von Voges und Ruthemann durchgeführt [VR39]. Weiter entwickelt wurde diese Technik von J. Hillier und R. F. Baker [Hil43, HB44] Mitte der 1940er Jahre um Spektren an schwereren Elementen aufnehmen zu können. Einer

der ersten modernen Energiefilter zur Aufnahme von EELS-Spektren wurde entwickelt von Castaing und Henry [CH62]. Andere Filterformen wurden von Boersch und Möllenstedt vorgeschlagen [KH03].

### 2.2.1 Ein Spektrum

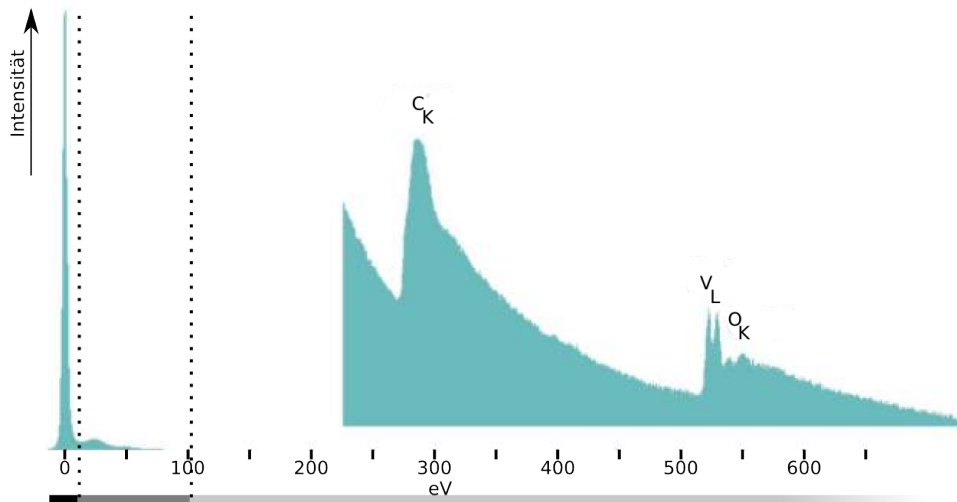


Abbildung 2: Elektronen-Energieverlust-Spektrum von Vanadiumoxid ( $VO_x$ ) Nanoröhren. Zwischen den Schichten aus Vanadiumoxid ist ein Templat – ein langkettiges Amin – eingelagert. Zu sehen sind die L-Kante von Vanadium, die K-Kante des Sauerstoffs und die K-Kante des Kohlenstoffs, welcher in dem Amin enthalten ist. Durch die drei unterschiedlich dunklen grauen Balken, werden drei Regionen des Spektrums hervor gehoben. Der dritte Bereich ist stark vergrößert dargestellt. Eine Beschreibung der Charakteristika dieser Regionen ist im Text zu finden. Quelle: [Kru, KMN<sup>+</sup>00]

Als Beispiel zeigt Abbildung 2 ein typisches EELS-Spektrum von Vanadiumoxid Nanoröhren. Es besteht aus drei Teilen [Kru]:

**Null-Verlust-Peak bei 0eV** Dieser besteht hauptsächlich aus Elektronen mit der ursprünglichen Energie, d.h. dass sie an einer elastischen Streuung oder an keiner Wechselwirkung beteiligt waren. Des weiteren enthält dieser Peak Elektronen, welche quasi-elastisch mit der Probe wechselwirkten, d.h. diese Elektronen haben einen Energieverlust erlitten, der aufgrund der Energieauflösung des Spektrometers nicht messbar ist. Zu den quasi-elastisch wechselwirkenden Elektronen zählen u.a. Bragg gebeugte Elektronen und solche, welche Phononen angeregt haben.

**Region mit geringem Verlust (kleiner 100eV)** In diesem Bereich regen die Elektronen Plasmonenoszillationen an. Da die Plasmonenge-

neration die häufigste Wechselwirkung von Elektronen mit einer Probe ist bei der die Elektronen Energie verlieren, ist die Intensität in dieser Region relativ hoch. Intensität und Anzahl der Plasmonenpeaks nimmt mit Probendicke zu.

**Region mit hohem Verlust (größer 100eV)** Für die Ionisation von Atomen muss eine elementspezifische Mindestenergie, die kritische Ionisierungsenergie  $E_c$ , vom einfallenden Elektron an ein Rumpfelektron übergeben werden. Dies führt zu Ionisierungskanten im Energieverlustspektrum, welche charakteristisch für jedes Element sind.

Im Vergleich zur Plasmonengeneration ist die Ionisation von Rumpfelektronen ein weniger häufig auftretender Effekt, was zu einer geringen Intensität der Peaks führt. In dieser Region nimmt die Zahl der inelastisch gestreuten Elektronen drastisch mit zunehmendem Energieverlust ab. Da beim Ionisierungsprozess mehr Energie als  $E_c$  aufgenommen werden kann, ist auch hinter der entsprechenden Kante Intensität vorhanden. Daher sind kleine Peaks mit einem schnell abfallenden Hintergrund überlagert. Der Bereich dieses Hintergrunds wird als ELNES („Energy-Loss Near-Edge Structure“) bezeichnet und spiegelt die Zustandsdichte wieder und enthält Informationen über die Bindungsverhältnisse [Kru]. Aufgrund dieser geringen Intensität ist die Region mit hohem Verlust meist stark vergrößert dargestellt. In dem gezeigten Spektrum (Abbildung 2) wird ab einer Verlustenergie von 220eV die Intensität vergrößert dargestellt.

Die kritische Ionisierungsenergie  $E_c$  ist abhängig von der chemischen Situation des Elements, z.B. die L-Kanten von elementarem Kupfer und Kupfer in Kupferoxid sind gegeneinander verschoben (chemische Verschiebung) [Kru].

### 2.2.2 Das Spektrometer

Ein Spektrometer für Elektronen-Energieverlust-Spektrometrie nutzt aus, dass Elektronen mit unterschiedlicher kinetischer Energie in einem homogenen magnetischen (oder alternativ elektrischen) Feld unterschiedlich stark abgelenkt werden. Wenn ein nahezu monochromatischer Elektronenstrahl eine Probe durchdringt, verlieren einige Elektronen Energie aufgrund inelastischer Wechselwirkungen und können somit dispergiert – d.h. räumlich separiert und detektiert – werden.

Üblicherweise geschieht diese räumliche Separation in einem Spektrometer mittels eines sektorförmigen Magneten (siehe Abbildung 3), welcher ein Magnetfeld  $\vec{B}$  senkrecht zum Elektronenpfad erzeugt. Die Lorentzkraft auf ein Elektron (Ladung  $e$ , Masse  $m_e$  und Geschwindigkeit  $\vec{v}$ ) aufgrund dieses

Feldes ist:

$$\vec{F} = e\vec{v} \times \vec{B}. \quad (2.1)$$

Da diese Kraft immer senkrecht zur Elektronengeschwindigkeit ist, wird das Elektron auf eine Kreisbahn gezwungen. Der Radius  $r$  dieser Kreisbahn ist gegeben durch:

$$evB = \gamma m_e \frac{v^2}{r} \quad (2.2)$$

$$\Rightarrow r = \gamma \frac{m_e v}{eB} \quad (2.3)$$

$$\gamma = \frac{1}{\sqrt{1 - \left(\frac{v}{c}\right)^2}}, \quad (2.4)$$

wobei  $\gamma$  ein relativistischer Faktor ist.

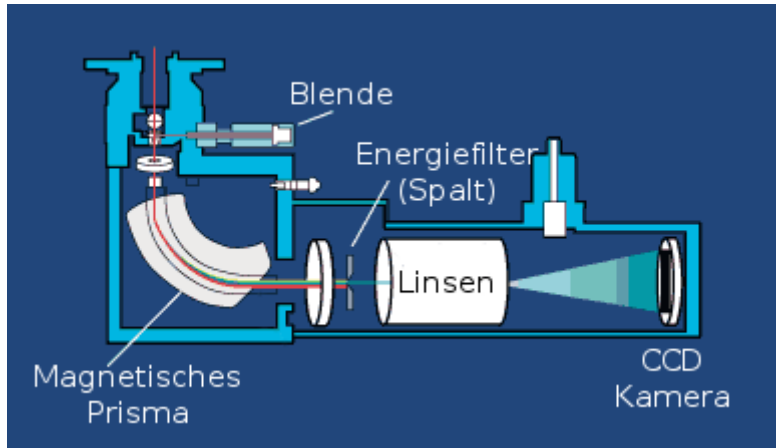


Abbildung 3: Schematische Darstellung des EELS-Spektrometers „Post-Column TEM Energy Filter Model 850 GIF 2002“.

Der Elektronenstrahl fällt durch die Aperturblende in das Spektrometer ein. Durch einen sektorförmigen Magneten (magnetisches Prisma) wird der Strahl nach Energie gefiltert. Mittels eines Spaltes werden Elektronen in einem kleinen Energieintervall ausgewählt und durch eine CCD-Kamera detektiert. Quelle: [Inc07]

Aufgrund dieser Abhängigkeit des Radius  $r$  von der Elektronengeschwindigkeit (bzw. kinetischen Energie) werden die Elektronen aufgrund des magnetischen Feldes räumlich separiert. Durch die Platzierung eines Spaltes an unterschiedlichen Radien und die Verwendung einer CCD-Kamera (siehe Abbildung 3) lässt sich die Anzahl der Elektronen, welche einen bestimmten Energieverlust erlitten haben, bestimmen. Somit ergibt sich ein Spektrum wie es z.B. in Abbildung 2 zu sehen ist.

Die Abbildung 3 zeigt schematisch ein Spektrometer der Firma Gatan (Modell 850 GIF 2002). Der Elektronenstrahl fällt von oben durch eine Blende in das Spektrometer ein und wird durch den sektorförmigen Magneten (magnetisches Prisma) abgelenkt und gemäß (2.4) aufgespalten. Mittels eines Spalts (Energiefilter – „Piezo-controlled Energy-Selecting Slit“) werden nur Elektronen mit einer bestimmten Energie ausgewählt und durch eine Optik (Linsen) auf eine CCD-Kamera gelenkt, wo sie detektiert werden.

### 2.3 Rasterelektronenmikroskopie

Das erste Bild mit einem Rasterelektronenmikroskop (SEM – „Scanning Electron Microscope“) wurde 1935 von Max Knoll aufgenommen [Kno35].

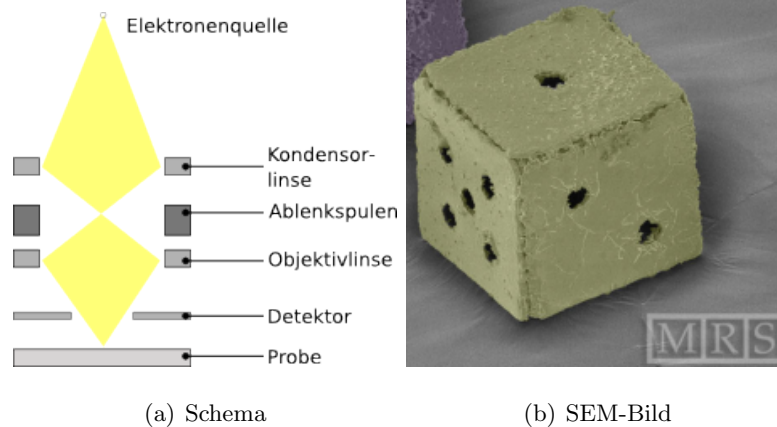


Abbildung 4: Schematischer Aufbau eines SEM und SEM-Aufnahme.  
 (a) Schematischer Aufbau eines SEM mit Linsen, Ablenkspulen und Detektor. Nachbildung einer Grafik aus [Kru]  
 (b) SEM-Aufnahme eines  $200\mu\text{m}$  großen Nickel-Würfels in Falschfarben. Quelle: [Leo07]

Im SEM wird ein fokussierter Elektronenstrahl über eine Probe gerastert und die „Sekundärelektronen“, welche durch die Wechselwirkung der Strahlelektronen („Primärelektronen“) mit den Atomen der Probe erzeugt werden, detektiert. Das Bild wird Punkt für Punkt aufgenommen. Da sich der Detektor auf der selben Seite der Probe wie die Elektronenquelle befindet, können mit einem SEM kompakte Proben untersucht werden im Gegensatz zum TEM, in dem nur dünne Proben durchstrahlt werden können. Mittels Rasterelektronenmikroskopie können wertvolle Informationen über Morphologie, Oberflächentopologie und Zusammensetzung gewonnen werden. In einem SEM können Auflösungen unter  $1\text{nm}$  erreicht werden.

## 2.4 Rastertransmissionselektronenmikroskopie

Das erste Rastertransmissionselektronenmikroskop (STEM – „Scanning Transmission Electron Microscope“) wurde 1938 von Baron Manfred von Ardenne gebaut [vA38b, vA38a]. Die Technik wurde in den 1970er Jahren verfeinert durch die Entwicklung der Feldemissionskathode (FEG – „Field Emission Gun“) durch Albert Crewe [CIJ69] und hochqualitativer Objektivlinsen, so dass einzelne Atome erstmals aufgelöst werden konnten [CWL70].

Der STEM-Modus ist ein Betriebsmodus eines Transmissionselektronenmikroskops, bei welchem der Elektronenstrahl auf die Probe fokussiert wird. Dieser Strahl wird dann mittels Ablenkspulen über die Probe gerastert ähnlich zu einem Rasterelektronenmikroskop. In der Beobachtungsebene befinden sich Detektoren, welche das STEM-Signal erzeugen.

### 2.4.1 Das Reziprozitätsprinzip

Um ausgehend vom schematischen Aufbau eines konventionellen Transmissionselektronenmikroskops (TEM) den Aufbau und die Funktionsweise eines STEM zu verstehen bietet sich die Argumentation mittels des Reziprozitätsprinzips an [Cow69, RP77, Spe03].

Die Umkehrbarkeit bzw. die Reziprozität bei der Ausbreitung von Lichtstrahlen – und Elektronenwellen – ist bereits durch das Fermat'sche Prinzip gegeben. Danach wird lediglich gefordert, dass ein Weg auf die schnellst mögliche Art zurückgelegt wird. In welche Richtung der Lichtstrahl – oder hier der Elektronenstrahl – verläuft spielt dabei keine Rolle.

Dabei ist zu beachten, dass dies in der Praxis bei magnetischen Linsen aufgrund der Lorentzkraft nicht erfüllt ist. Hier muss die Magnetfeldrichtung geändert werden, wenn die Strahlrichtung geändert wird.

Abbildung 5(a) zeigt ein vereinfachtes Diagramm eines Transmissionselektronenmikroskops (TEM), wobei L die Objektivlinse, S eine Punktquelle am Punkt A und P ein Punkt in der Detektorebene sind. Die sphärische Welle, die vom Punkt A ausgeht, kann in der Probenebene, welche weit von der Quelle entfernt ist, als ebene Welle beschrieben werden. Die Linse L bildet alle Punkte in der Probenebene in die Detektorebene ab.

Das Reziprozitätsprinzip besagt, dass eine Intensität  $I_p$ , welche am Punkt P aufgrund einer Punktquelle am Ort A gemessen wird (siehe Abbildung 5(a)), identisch zu der Intensität ist, die man am Punkt A detektiert, wenn man die Punktquelle und den Detektor vertauscht und das restliche System dabei unverändert lässt.

Abbildung 5(b) zeigt das Strahlendiagramm eines TEM aus Abbildung 5(a), wobei die Punktquelle S und der Detektor, welcher den Punkt P enthält, vertauscht sind. Dies entspricht nun dem Strahlendiagramm eines

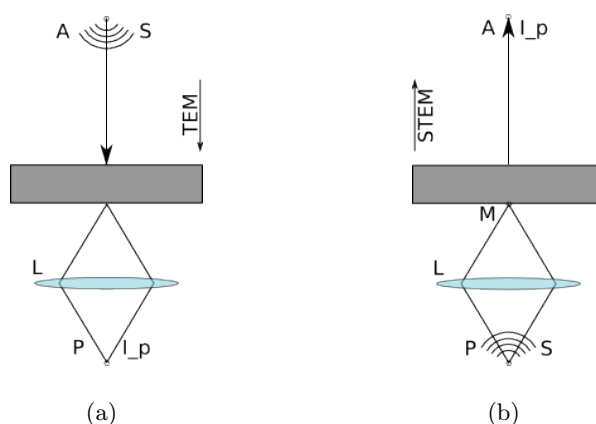


Abbildung 5: Das Reziprozitätsprinzip. Die Figuren sind Nachbildungen aus Referenz [Spe03]

(a) Strahlendiagramm für ein vereinfachtes TEM-Instrument. Ein Punkt S in der beleuchtenden Blende emittiert eine Kugelwelle, welche am Ort der entfernten Probe als ebene Welle betrachtet werden kann. Die Linse L bildet die untere Fläche der Probe auf den Detektor ab, auf welchem der Punkt P liegt.

(b) Strahlendiagramm für ein STEM-Instrument. Quelle (S) und Detektor sind gegenüber Diagramm (a) vertauscht. Die Quelle befindet sich am Ort P und der Detektor am Ort A. Die Linse L fokussiert die von P ausgehende Welle auf den Punkt M in der unteren Fläche der Probe.

STEM. Die vom Punkt S ausgehende Kugelwelle wird durch die Linse L auf den Punkt M in der Probenebene fokussiert. Am Ort A befindet sich ein (Hellfeld-) Detektor. Eine kompakte Übersicht über Detektoren wird in Anhang A gegeben.

Das Reziprozitätsprinzip lässt sich erweitern auf ausgedehnte Quellen und Detektoren. Darauf soll aber an dieser Stelle nicht eingegangen werden. Dies wird z.B. ausführlich beschrieben in Referenz [Spe03].

### 2.4.2 Aufbau und Funktion

Abbildung 6 zeigt den schematischen Aufbau eines STEM mit Linsen, Blenden und Ablenkungspulen. Elastisch gestreute Elektronen (Streuwinkel  $> \beta_{el}$ ) werden mit einem ringförmigen Detektor aufgenommen und erzeugen das Dunkelfeldsignal (DF – „dark field“). Ein Spektrometer lenkt die Elektronen, welche bei einer inelastischen Streuung (Streuwinkel  $< \beta_{in}$ ) einen Energieverlust erlitten haben, stärker ab, als ungestreute Elektronen. Das Hellfeldsignal (BF – „bright field“) wird erzeugt aus ungestreuten und quasi-elastisch (inelastisch gestreut, aber Energieverlust kleiner als Auflösungsvermögen des Spektrometers) gestreuten Elektronen. Dazu wird eine kleine Blende auf der optischen Achse platziert. Die BF- und DF-Signale können

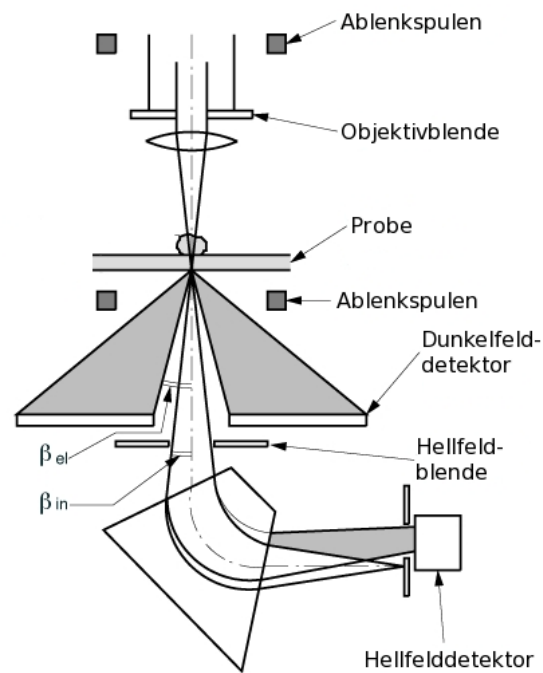


Abbildung 6: Schematische Darstellung der elektronenoptischen Komponenten eines Rastertransmissionselektronenmikroskops. Quelle: [MSAE]

parallel aufgenommen und ausgewertet werden, während der Strahl über die Probe gerastert wird. Weitere Informationen zum Thema Detektoren und Abbildungen zur Verdeutlichung der Charakteristika der Detektoren finden sich in Anhang A.



### 3 Zirkulardichroismus für Röntgenstrahlung und Elektronen

#### 3.1 Dichroismus und Chiralität

Polarisation ist eine Eigenschaft elektromagnetischer Wellen, welche die Richtung des Amplitudenvektors des elektrischen Feldes beschreibt, und zwar im Vakuum oder in optisch-isotropen Medien in Bezug auf den Wellenvektor  $\vec{k}$ . Sie beschränkt sich auf Transversalwellen. In nicht-kubischen Kristallen gibt es Abweichungen, die zu dem Phänomen der Doppelbrechung führen. Bei longitudinalen Wellen tritt kein Polarisationsphänomen auf, da die Schwingung in Ausbreitungsrichtung erfolgt.

Eine Transversalwelle ist durch zwei Richtungen charakterisiert: Den Wellenvektor und den Amplitudenvektor des elektrischen Feldes, der unter den angegebenen Voraussetzungen immer senkrecht auf dem Wellenvektor steht. Dies lässt im dreidimensionalen Raum noch einen Rotationsfreiheitsgrad offen, nämlich die Rotation um den Wellenvektor. Man unterscheidet drei Arten von Polarisation:

- **Elliptische Polarisation** – Der Amplitudenvektor rotiert mit konstanter Winkelgeschwindigkeit um den Wellenvektor und ändert dabei periodisch den Betrag. Die Spitze des Feldvektors beschreibt dabei eine Ellipse.
- **Zirkulare Polarisation** – Der Amplitudenvektor rotiert mit konstanter Winkelgeschwindigkeit um den Wellenvektor und ändert seinen Betrag dabei nicht.
- **Lineare Polarisation** – Der Amplitudenvektor zeigt immer in eine feste Richtung und die Auslenkung ändert bei Propagation der Welle ihren Betrag und ihr Vorzeichen periodisch.

Der Begriff Dichroismus (vom lateinischen „di“ = doppelt und vom griechischen „chrome“ = Farbe, also Zweifarbigkeit) bezeichnet in der Optik die Eigenschaft eines Materials, Lichtstrahlen mit unterschiedlicher Polarisation verschieden stark zu absorbieren. Ein solches Material wird demnach als dichroisch bezeichnet [SZ82].

Dichroismus wird beobachtet in uniaxialen Kristallen, d.h. in Kristallen, die eine optische Achse haben. Typische Vertreter uniaxialer Kristalle sind u.a. Kristalle mit einer tetragonalen oder hexagonalen Kristallstruktur (siehe Abbildung 3.1). An ihnen kann außerdem Doppelbrechung beobachtet werden, da diese Kristalle für Licht, welches entlang der optischen Achse oder senkrecht dazu einfällt, unterschiedliche Brechungsindizes aufweist [Bri]. In

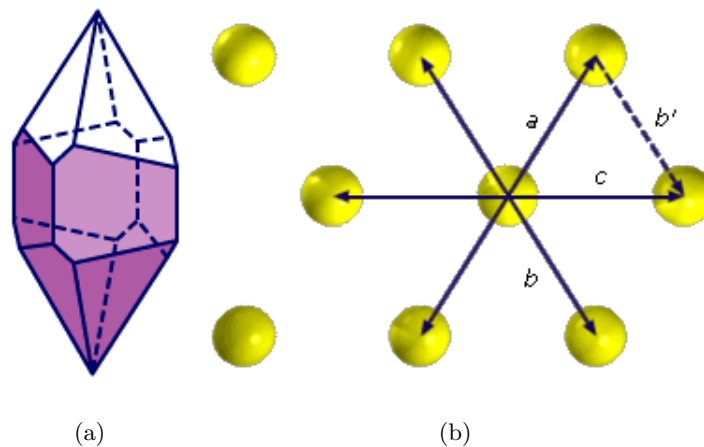


Abbildung 7: Zur Verdeutlichung der Doppelbrechung und Dichroismus aufweisenden Kristallstrukturen. Quelle: [Bri]

(a) Tetragonaler Kristall: Drei zueinander senkrecht stehende Kristallachsen, von denen zwei gleich lang sind. Die tetragonale Einheitszelle weist eine vierfache Symmetrie auf. Die Elemente Bor und Zinn sowie einige Kristalle wie Zirkon können in tetragonaler Form kristallisieren.

(b) Hexagonaler Kristall: Drei gleich lange Kristallachsen in einer Ebene mit einem Winkel von  $120^\circ$  und einer dazu senkrecht stehenden Achse. Die hexagonale Einheitszelle weist eine sechsfache Symmetrie auf. In hexagonaler Form kristallisieren z.B. Arsen, Kalkspat, Quarz und Graphit. Die Pfeile deuten Translationssymmetrien des Gitters an. Kombiniert man zwei dieser Symmetrien (z.B.  $a$  und  $b$ ), erhält man eine dritte ( $c$ ).

der Mineralogie wird daher mit Dichroismus die Eigenschaft von Mineralien bezeichnet bei Betrachtung aus verschiedenen Blickwinkeln verschiedene Farben zu zeigen.

Es wird unterschieden, welche Art von Polarisation das einfallende Licht hat. Es gibt den linearen und den zirkularen Dichroismus. Bei ersterem hängt die Absorption linear polarisierten Licht durch ein chirales Material von der Orientierung des Polarisationsvektors ab. Analog dazu absorbieren andere Materialien Licht entgegengesetzter zirkularer Polarisation (links. bzw. rechts-zirkular) verschieden stark; dies wird als zirkularer Dichroismus oder Zirkulardichroismus bezeichnet (siehe Abbildung 8).

### 3.2 Chiralität und Zirkulardichroismus

Die Chiralität ist die Eigenschaft bestimmter Gegenstände oder Systeme, dass ihr Spiegelbild durch Drehung nicht mit dem Original zur Deckung gebracht werden kann. Objekte mit dieser Eigenschaft werden chiral genannt. Ein chirales Objekt und sein Spiegelbild werden auch als Spiegelbildisomerenpaar oder Enantiomerenpaar bezeichnet. Im Gegensatz dazu wird ein

Objekt als achiral oder amphichiral bezeichnet, wenn es mit seinem Spiegelbild durch Drehung zur Deckung gebracht werden kann.

Allgemein ist ein Objekt genau dann chiral, wenn es keine Drehspiegelachse besitzt. Andere Symmetrieelemente können aber durchaus vorhanden sein, d.h. ein chirales Objekt ist nicht zwangsläufig asymmetrisch.

Gängige Beispiele sind rechte und linke Hand oder rechts- bzw. links gewundene Schneckenhäuser. Zur Verdeutlichung zeigt Abbildung 9 das Enantiomerenpaar eines typischen chiralen Moleküls. Ein solches besteht aus einem Kohlenstoffatom, welches vier Reste trägt. In der Chemie wird das Kohlenstoffatom als Stereozentrum, Asymmetriezentrum oder Chiralitätszentrum bezeichnet. [MPG07]

Ein chirales Objekt, das unterschiedlich mit den Enantiomeren eines chiralen Verbundes wechselwirkt, ist zirkular polarisiertes Licht. Ein Enantiomer weist für links- bzw. rechts-zirkular polarisiertes Licht abweichende Transmissionsverhalten auf. Dies ist die Grundlage für optische Aktivität und die Spektrometrie an zirkulardichroischen Materialien. [Rub07]

Optische Aktivität ist die Rotation der Polarisationssebene von linear polarisiertem Licht beim Durchgang durch bestimmte Medien, z.B. Lösungen chiraler Moleküle wie Saccharose (Zucker). Dies ist möglich, da die lineare Polarisation eine Superposition von links- und rechts-zirkular polarisiertem Licht ist und ein chirales Objekt links- und rechts-zirkular polarisiertes Licht unterschiedlich stark absorbiert. Daraus resultiert eine Verkipfung der linearen Polarisationsrichtung.

### 3.3 Röntgenzirkulardichroismus

In der Anwesenheit eines magnetischen Feldes sind alle Moleküle optisch aktiv. Ist das Feld parallel zur Ausbreitungsrichtung von Licht, so wird eine Drehung der Polarisationssebene des linear polarisierten Lichts bewirkt. Dies ist bekannt als Faraday- und magnetooptischer Kerr-Effekt. Diese Effekte unterscheiden sich lediglich durch die Messung des transmittierten bzw. reflektierten Lichts.

Magnetisch induzierter Zirkulardichroismus (MCD) ist die unterschiedlich starke Absorption von links- und rechts-zirkular polarisiertem Licht durch ein magnetisiertes Material. Absorption findet nur bei ausgewählten Wellenlängen statt, welche den Energiedifferenzen zwischen zwei Zuständen – Anfangs- und Endzustand – des absorbierenden Atoms oder Moleküls entsprechen. Findet die Absorption (und die Detektion) im Röntgenbereich statt, so spricht man von XMCD („X-ray magnetic circular dichroism“). [SWW<sup>+</sup>87, Rub07]

1975 sagten Erskine und Stern [ES75] voraus, dass Röntgenabsorption mit zirkular polarisiertem Licht magnetische Informationen über den anfäng-

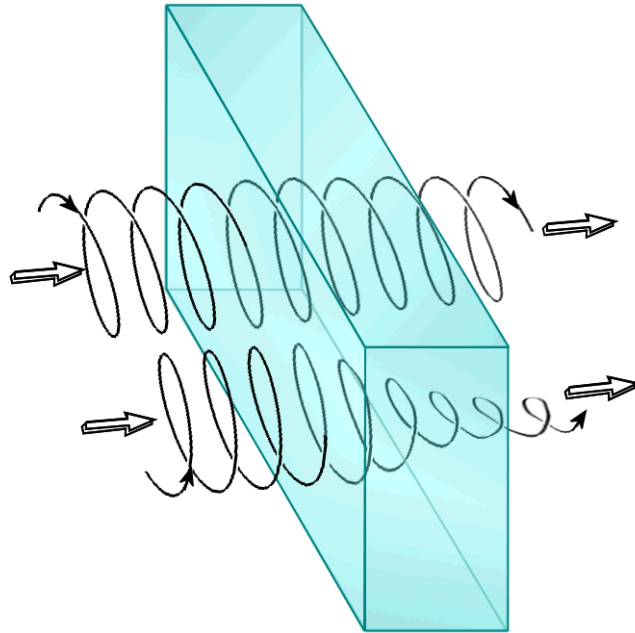


Abbildung 8: Zirkularer Dichroismus – Trajektorien der Polarisationsvektoren einer links und rechts zirkular polarisierten elektromagnetischen Welle, die sich durch ein dichroisches Medium ausbreiten. Dabei werden diese unterschiedlich stark absorbiert. Bild wurde entnommen aus [Wik]

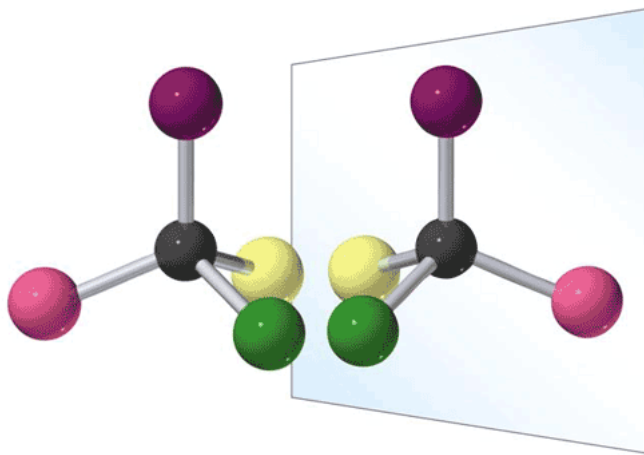


Abbildung 9: Chirale Moleküle können durch Spiegelung nicht auf sich selbst, aber auf ihr Gegenstück, das sogenannte Enantiomer, abgebildet werden. Quelle: [MPG07]

lichen Zustand des Absorptionsprozesses liefern kann. 1985 zeigte Thole [TvdLS85], dass der Verlauf der  $M_{4,5}$ -Absorptionskanten von Metallionen der Seltenen Erden, deren Grundzustand durch ein Magnetfeld aufgespalten wird, abhängt von der relativen Orientierung zwischen der Magnetisierungsrichtung und dem Polarisationsvektor der Röntgenstrahlung.

Chen nahm 1990 XMCD-Spektren an den  $L_{2,3}$ -Kanten ( $2p \rightarrow 3d$  Übergänge) von Nickel [CSMM90], Kobalt und Eisen auf. Der Unterschied in der Absorption betrug dabei bis zu 20% der gesamten Absorption. Für diese Kanten findet der Übergang in leere  $3d$  Zustände, welche stark polarisiert sind, direkt statt. [NEE]

XMCD lässt sich verstehen anhand der Betrachtung der Übergangswahrscheinlichkeiten von einem gebundenen Zustand in einen freien Zustand, welcher oberhalb der Fermi-Energie liegt. Ein Atom erhält ein Drehimpulsquantum bei der Absorption eines zirkular polarisierten Photons, wenn dessen Helizität parallel zur Magnetisierung ist, d.h. es sind nur Übergänge erlaubt, welche die Auswahlregel  $\Delta J = \pm 1$  erfüllen.

Die Helizität ist eine Quantenzahl, die den Spin und den Impuls eines Teilchens miteinander verbindet. Sie ist eine innere Eigenschaft von Teilchen mit der Ruhemasse Null (Photonen, Neutrinos) und charakterisiert einen Schraubensinn. Wenn der Impuls- und Spinvektor parallel zueinander liegen, ergibt sich die Helizität  $+1$ ; liegen die Vektoren antiparallel, ist die Helizität  $-1$ . Die Helizität  $H$  ist definiert als die Projektion des Spinvektors eines (Elementar-)Teilchens auf dessen Impulsvektor.

$$H = \frac{\vec{s} \cdot \vec{p}}{|\vec{s}| \cdot |\vec{p}|}, \quad (3.1)$$

wobei  $\vec{s}$  der Spin- und  $\vec{p}$  der Impulsvektor sind. Im Rahmen der Relativitätstheorie ist die Helizität nur für masselose Teilchen (die sich stets mit Lichtgeschwindigkeit bewegen) eindeutig bestimmt. Für alle anderen massebehafteten Teilchen lässt sich immer ein Bezugssystem wählen, das das Teilchen „überholt“, wodurch sich die Richtung seines Impulses und damit seine Helizität umkehrt. [Mey]

Gemäß [Rub07] legen Photonen mit der Helizität  $\pm 1$  die zusätzliche Auswahlregel  $\Delta m = \pm 1$  nahe. Des weiteren wird aufgrund des magnetischen Feldes die  $l$  - Entartung des Endzustandes aufgehoben (Zeeman-Effekt). Folglich können damit Photonen mit entgegengesetzten Helizitäten ( $+1$  und  $-1$ ) während ihrer Absorption den Übergang in einen bestimmten dieser aufgespaltenen Endzustände erzwingen. Diese Übergänge können unterschiedliche Übergangswahrscheinlichkeiten aufgrund einer Abhängigkeit von der Magnetisierungsrichtung und der Helizität des zu absorbierenden Photons aufweisen. Die resultierenden Spektrallinien in Absorption weisen

somit verschiedene Intensitäten auf und verdeutlichen somit den Einfluss der Polarisation auf Absorptionsspektren. [Rub07]

### 3.4 Chiraldichroismus bei inelastischer Elektronenstreuung

Die ersten Schritte zur Erforschung der Analogie zwischen Elektronen-Streuung und Röntgenabsorption unternahm Hitchcock 1993 [Hit93]. Der Lineardichroismus an magnetischen Materialien für Elektronen wurde 1997 von Yuan und Menon untersucht [YM97]. Es besteht eine formelle Analogie zwischen dem Polarisationsvektor  $\vec{\epsilon}$  in Röntgenabsorption und dem Impulsübertrag  $\hbar\vec{q}$  bei Elektronenstreuung. Dabei ist  $\vec{q}$  der Wellenvektorübertrag. Auf diese Analogie soll im Unterkapitel 3.5 näher eingegangen werden.

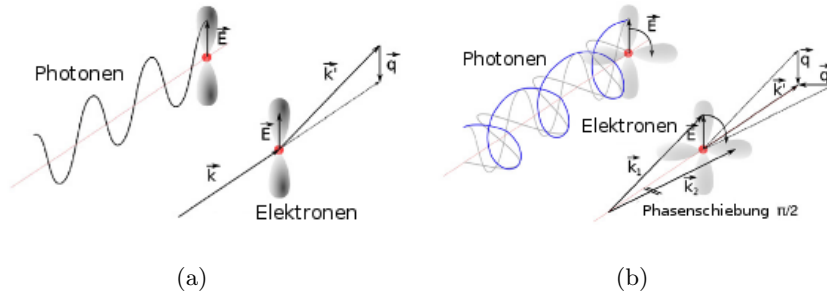


Abbildung 10: Analogie zwischen Polarisationsvektor  $\vec{\epsilon}$  (identisch mit dem Vektor des elektrischen Feldes  $\vec{E}$ ) bei Röntgenabsorption und Impulsübertrag  $\hbar\vec{q}$  bei inelastischer Elektronenstreuung.  $\vec{q}$  ist dabei der Wellenvektorübertrag. Quelle: [HSR<sup>+</sup>08]

(a) Lineare Polarisation – Bei Photonenabsorption ist das elektrische Feld  $\vec{E}$  parallel zur Polarisation  $\vec{\epsilon}$ . Bei inelastischer Elektronenstreuung wird für ein Elektron, welches einen Impulsübertrag  $\hbar\vec{q}$  erfährt, das die Elektronenwelle beschreibende Feld  $\vec{E}$  derart definiert, dass es antiparallel zum Impulsübertrag  $\hbar\vec{q}$  ist.

(b) Zirkulare Polarisation – Superposition zweier linear polarisierter elektromagnetischer Wellen liefert eine zirkular polarisierte Welle. Analog dazu werden zwei Elektronenwellen mit Impulsüberträgen  $\hbar\vec{q}$  und  $\hbar\vec{q}'$  überlagert. Da die Elektronenstreuung statistisch erfolgt, müssen die beiden Impulsüberträge durch eine Blende ausgewählt werden, so dass die Impulsüberträge senkrecht aufeinander stehen. Mit einer Phasendifferenz von  $\pi/2$  zwischen den beiden Wellen ergibt sich das Analogon zur Zirkularpolarisation elektromagnetischer Wellen.

Diese Analogie zwischen Elektronen und Photonen wird zuerst für lineare Polarisation beschrieben und im Anschluss auf zirkulare Polarisation erweitert, die als eine Superposition von zwei linear polarisierten Wellen betrachtet wird. Dabei sollen – um die zirkular polarisierte Welle zu erzeugen – die linear polarisierten Wellen um  $90^\circ$  phasenverschoben sein. Für Elektronen wird dies in Analogie zu Photonen ( $\vec{\epsilon} \pm i\vec{\epsilon}'; \vec{\epsilon} \perp \vec{\epsilon}'$ : Polarisationsvektoren) geschrieben als  $\vec{q} \pm i\vec{q}'$ , wobei  $\hbar\vec{q}$  und  $\hbar\vec{q}'$  Impulsüberträge im Prozess der Elektronenstoßionisation darstellen. Dabei bezeichnen  $\vec{q}$  und  $\vec{q}'$

die Differenzen (Wellenvektorüberträge) der Wellenvektoren von End- und Anfangszuständen (siehe dazu Abbildung 10(a) für lineare Polarisierung und Abbildung 10(b) für zirkuläre Polarisierung). Für diese soll gelten:  $\vec{q} \perp \vec{q}'$ ,  $|\vec{q}| = |\vec{q}'|$ . In der Praxis werden diese beiden Überträge durch eine Blende ausgewählt (siehe Abbildung 12(b) ( $\rightarrow$  Seite 34)). [SRH<sup>+</sup>06]

Zur Erzeugung dieser Superposition aus zwei ebenen Wellen in der Praxis sind bereits zwei Verfahren bekannt: die Verwendung eines Biprismas [EFH<sup>+</sup>06] und die Verwendung der Probe als Strahlteiler. Letztere Methode wird als „intrinsischer Weg“ bezeichnet und wurde erstmals von Nelhiebel [NSJ00] vorgeschlagen. Bei der Biprismenmethode limitieren technische Probleme die Intensität auf ein unzureichendes Maß [HSR<sup>+</sup>08]. Wie in der Einleitung bereits erwähnt, soll in dieser Arbeit eine dritte Möglichkeit – die Verwendung einer Doppelochblende mit Phasenplatte – vorgestellt und erste Versuche eines STEM-Betriebs mit einer solchen Blende durchgeführt werden (siehe Kapitel 4 und 7). Im weiteren Verlauf dieses Abschnittes soll zunächst auf die intrinsische Methode eingegangen werden, bevor die Doppelochblende eingeführt wird.

Um solche Chiraldichroismus-Messungen in inelastischer Elektronenstreuung (EMCD – „Energy-loss Magnetic Chiral Dichroism“) durchführen zu können werden eine Reihe von Anforderungen an die Proben und den Versuchsaufbau gestellt. Erstere werden in Referenz [Hur08] näher beschrieben und lauten wie folgt:

- Die Probe muss ferromagnetisch sein.
- Die Magnetisierung sollte möglichst (anti-) parallel zur optischen Achse gesättigt sein.
- Die Probe muss flach und eben sein.
- Die Probe darf kein Substrat haben.

Speziell bei der intrinsischen Methode sind noch zwei weitere Bedingungen aufgrund der Verwendung der Probe als Strahlteiler gegeben. Diese zusätzlichen Bedingungen, auf welche in Referenz [Hur08] näher eingegangen wird, lauten wie folgt:

- Die Probe muss monokristallin sein oder – im Falle einer polykristallinen Probe – Kristallite aufweisen, die größer als der beleuchtete Bereich der Probe sind.
- Die Probe muss eine optimale, elementspezifische Dicke aufweisen (siehe Tabelle 2).
- Die Probe muss exakt ausgerichtet werden.

Element	Opt. Dicke
Eisen	8 - 22 nm
Nickel	6 - 10 nm
Cobalt	15 - 22 nm

Tabelle 2: Optimale Dicke für EMCD-Messungen von drei Elementen. Quelle: [RRS07]

Durch die Verwendung der Probe als Strahlteiler und die Erfüllung dieser Bedingungen soll in der Beugungsebene eine Reihe von Bragg-Reflexen mit einer Phasendifferenz angeregt werden. Die Anregung von zwei Reflexen ist in diesem Betriebsmodus des TEM vorteilhaft [RhLL<sup>+</sup>08]. Der optimale Wert für die Phasendifferenz ist  $\pi/2$ . Die Abhängigkeit der Phasendifferenz von der Kristallorientierung und -dicke ergibt sich aus einer dynamischen Streutheorie. Diese wird in Referenzen [Rub07, Hur08] ausführlich erörtert.

Die Anforderungen an den Versuchsaufbau werden in Referenz [Rub07] beschrieben und lauten wie folgt:

- Es müssen zwei kohärente Elektronenwellen existieren, damit zwei simultane Impulsüberträge und Interferenz stattfinden können
- Die Impulsüberträge  $\vec{q}$  und  $\vec{q}'$  dürfen nicht parallel sein; ein Winkel von  $90^\circ$  zwischen diesen ist optimal, damit der analoge Fall zur zirkularen Polarisierung eintritt
- Die Wellen dürfen nicht in Phase sein; eine Phasendifferenz von  $\pi/2$  am Ort der Probe ist optimal

Zur Veranschaulichung zeigt Abbildung 11 schematisch einen experimentellen Aufbau für die intrinsische Methode.

Als nächstes müssen zwei Impulsüberträge, die senkrecht zueinander stehen, ausgewählt werden. Dies kann in der Beugungsebene durch die Verwendung einer Blende geschehen. Eine detaillierte Beschreibung dieses Vorgehens in verschiedenen Betriebsmodi wird in Referenz [HSR<sup>+</sup>08, MHR<sup>+</sup>08] gegeben. Nach dieser Auswahl – die Position (+) oder (-) in Abbildung 11 – werden Elektronen-Energieverlust-Spektren aufgenommen. Um bei der Messung dieser Spektren ein maximales Signal zu erhalten, sollten große Blenden verwendet werden, die nicht auf dem Thaleskreis zentriert sind [VHR<sup>+</sup>08a, VHR<sup>+</sup>08b].

Der abschließende Schritt besteht in der Umkehrung der Helizität des anregenden Elektrons relativ zur Magnetisierung der Probe, d.h. Änderung des Vorzeichens des Interferenzterms in (3.12). Dies kann geschehen durch:



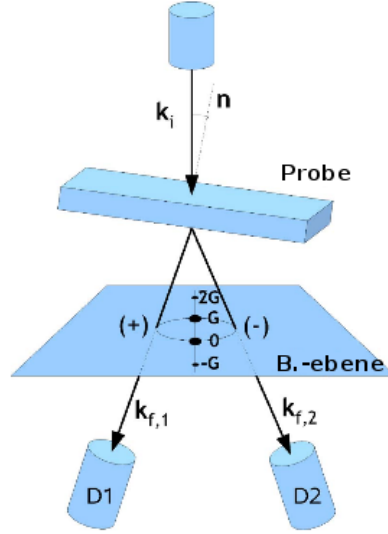


Abbildung 11: Experimentelle Anordnung für die intrinsischen Methode – Die monokristalline, ferromagnetische Probe ist geneigt, so dass ihre Flächennormale  $\mathbf{n}$  gegenüber der Richtung des einfallenden Strahls um einen Winkel von ca.  $10^\circ$  geneigt ist. Dies regt eine Reihe von Bragg-Reflexen  $-\mathbf{G}$ ,  $\mathbf{0}$ ,  $\mathbf{G}$ ,  $2\mathbf{G}$ , ... an (Ein Zwei-Strahl-Fall, d.h. die Anregung von nur zwei Bragg-Reflexen, ließe sich realisieren durch setzen des Mittelpunkts des Laue Kreises auf  $\tilde{G}/2$  (siehe dazu: [Rub07, RRS07])). An den Positionen **D1** und **D2** werden mit einem Detektor Elektronen-Energieverlust-Spektren aufgenommen. Durch die beiden am stärksten angeregten Bragg-Reflexe (hier  $\mathbf{0}$  und  $\mathbf{G}$ ) wird ein Thaleskreis definiert, wobei der Abstand  $\overline{OG}$  den Durchmesser des Thaleskreises vorgibt. Die Messpunkte (+) und (–), welche den Richtungen  $\mathbf{k}_{f,1}$  bzw.  $\mathbf{k}_{f,2}$  entsprechen, befinden sich auf dem Thaleskreis. Quelle: [RRS07]

- die Platzierung der Blende auf eine andere Position (–) oder (+), so dass das elektrische Feld, welches die Elektronenwelle beschreibt, in die entgegengesetzte Richtung rotiert. Dies entspricht einer Umkehr der Helizität analog zur Änderung der zirkularen Polarisierung bei einer elektromagnetischen Welle.
- die Umkehr der Stromrichtung in der Objektivlinse, was die Magnetisierungsrichtung in der Probe umkehrt [EFH<sup>+</sup>06, Hur08, HSR<sup>+</sup>08], so dass eine Vorzeichenänderung der Energiedifferenz zweier benachbarter und aufgrund des Zeeman-Effekts nicht mehr entarteter Energieniveaus erzielt wird.
- die „vorsichtige“ Variation [HSR<sup>+</sup>08] der Kristallausrichtung und Probendicke, so dass eine Vorzeichenänderung der Phasendifferenz erzielt wird

Ein EMCD-Spektrum ist somit die Differenz zweier EELS-Spektren.

Dazu wird eines dieser Spektren vor der Umkehr der Helizität aufgenommen und das zweite danach. Bei der ersten der drei oben beschriebenen Möglichkeiten werden die Spektren an unterschiedlichen Positionen in der Beugungsebene aufgenommen. In den anderen beiden Möglichkeiten werden sie an derselben Stelle aufgenommen.

### 3.5 Analogie Photonen – Elektronen

Um die Analogie zwischen Elektronen bei inelastischer Streuung und Photonen bei Röntgenabsorption zu zeigen werden im Folgenden die Streuquerschnitte für die beiden Fälle in gekürzter Fassung nach Referenz [Rub07, HS03] beschrieben und anhand eines Vergleiches dieser Streuquerschnitte die formale Analogie gezeigt. Eine detaillierte Herleitung des DDSCS wird in Referenz [NLS<sup>+</sup>99] gegeben.

Der Streuquerschnitt für die Streuung eines schnellen Elektrons an einem gebundenen Elektron ist in der Born-Näherung erster Ordnung nach den Referenzen [HS03, Rub07, RRS07, Sch86] gegeben durch:

$$\frac{\partial^2 \sigma}{\partial \Omega \partial E} = \frac{4\gamma^2 k_f}{a_0^2 k_i} \frac{S(\vec{q}, E)}{q^4} \quad (3.2)$$

$$S(\vec{q}, E) = \sum_{i, f} \left| \langle f | e^{i\vec{q} \cdot \vec{R}} | i \rangle \right|^2 \delta(E_f - E_i - E), \quad (3.3)$$

wobei das gebundene Elektron von einem Anfangszustand  $|i\rangle$  in einen Endzustand  $|f\rangle$  aufgrund der Wechselwirkung mit dem schnellen einlaufenden Elektron übergeht. Die Wellenfunktion des schnellen Elektrons geht aufgrund der Wechselwirkung von  $|\psi_i\rangle$  zu  $|\psi_f\rangle$  über.  $\vec{q} = \vec{k}_f - \vec{k}_i$  ist die Differenz (Wellenvektorübertrag) der Wellenvektoren von End- und Anfangszustand des schnellen Elektrons.  $E_f$  und  $E_i$  sind die Energien des End- bzw. Anfangszustandes des gebundenen Elektrons.  $\gamma = 1/\sqrt{1 - v^2/c^2}$  ist ein relativistischer Faktor,  $a_0$  ist der Bohr-Radius,  $S(\vec{q}, E)$  ist der dynamische Formfaktor (DFF),  $E$  ist der Energieverlust und  $\vec{R}$  ist der Ortsoperator.

(3.2) ist nur gültig, wenn die Wellenfunktionen des schnellen Elektrons ebene Wellen sind. In Kristallen ist die volle Translationssymmetrie gebrochen. Deshalb wird die Wellenfunktion des Elektrons zu einer Superposition von Bloch-Wellen. Jede Bloch-Welle kann in eine Linearkombination aus ebenen Wellen entwickelt werden. Eine detaillierte Beschreibung mit dieser Superposition aus Linearkombinationen wird gegeben in [RRS07].

Zum Vergleich der Absorptionsstreuquerschnitte für Elektronen (3.2) und für Photonen (3.4) wird der Streuquerschnitt für die Absorption eines Photons mit Energie  $E$  in Röntgenabsorptionsspektrometrie (XAS –

„X-ray Absorption Spectrometry“) nach Referenz [HS03, Rub07] wie folgt geschrieben:

$$\sigma = \sum_{i,f} 4\pi^2 \hbar \alpha \omega \left| \langle f | \vec{\epsilon} \cdot \vec{R} e^{i\vec{q} \cdot \vec{R}} | i \rangle \right|^2 \delta(E_f - E_i - E), \quad (3.4)$$

wobei  $\alpha = 1/137$  die Feinstrukturkonstante,  $\vec{\epsilon}$  der Polarisationsvektor und  $\hbar \vec{q}$  der Impuls des absorbierten Photons sind.

Nun soll die Dipolnherung

$$e^{i\vec{q} \cdot \vec{R}} = 1 + i\vec{q} \cdot \vec{R} - O(q^2 R^2) \quad (3.5)$$

sowohl auf den Streuquerschnitt fr Elektronen (3.2) als auch fr Photonen (3.4) angewandt werden. Man erhlt somit durch Ersetzen gem (3.5) fr (3.2)

$$\frac{\partial^2 \sigma}{\partial \Omega \partial E} = \sum_{i,f} \frac{4\gamma^2}{a_0^2 q^4} \frac{k_f}{k_i} \left| \overbrace{\langle f | i \rangle}^{=0} + \langle f | i \vec{q} \cdot \vec{R} | i \rangle \right|^2 \delta(E_f - E_i - E), \quad (3.6)$$

wobei der erste Term im Betragsquadrat in (3.2) aufgrund der Orthogonalitt der Zustnde  $|i\rangle$ ,  $|f\rangle$  verschwindet. (3.6) lsst sich daher in einfacherer Weise schreiben als

$$\frac{\partial^2 \sigma}{\partial \Omega \partial E} = \sum_{i,f} \frac{4\gamma^2}{a_0^2 q^4} \frac{k_f}{k_i} \left| \langle f | \vec{q} \cdot \vec{R} | i \rangle \right|^2 \delta(E_f - E_i - E). \quad (3.7)$$

Die Anwendung der Dipolnherung auf (3.4) ergibt

$$\sigma = \sum_{i,f} 4\pi^2 \hbar \alpha \omega \left| \langle f | \vec{\epsilon} \cdot \vec{R} | i \rangle + \langle f | i \vec{\epsilon} \cdot \vec{R} \vec{q} \cdot \vec{R} | i \rangle \right|^2 \delta(E_f - E_i - E). \quad (3.8)$$

Der zweite Term im Betragsquadrat in (3.8) liefert einen vernachlssigbaren Beitrag [HS03, Rub07]. Daher wird dieser weg gelassen und es ergibt sich somit der Streuquerschnitt fr XAS zu:

$$\sigma = \sum_{i,f} 4\pi^2 \hbar \alpha \omega \left| \langle f | \vec{\epsilon} \cdot \vec{R} | i \rangle \right|^2 \delta(E_f - E_i - E) \quad (3.9)$$

Gem den Ausfhrungen in Referenz [HS03, Rub07] zeigt der Vergleich der Absorptionsstreuquerschnitte fr Elektronen (3.7) und Photonen (3.9) in der verwendeten Dipolnherung nun eine formelle Analogie zwischen dem Polarisationsvektor  $\vec{\epsilon}$  in Photonenabsorption und dem Impulsbertrag  $\hbar \vec{q}$  in inelastischer Elektronenstreuung (siehe Abbildung 10(a)). Dies ist der Grund, warum linearer Dichroismus in (Elektronen-) Energieverlustspektren (EELS – „Electron Energy Loss Spectrometry“) beobachtet werden kann.

Die formelle Analogie zwischen Photonen und Elektronen soll im Weiteren auf zirkuläre Polarisation erweitert werden. Dazu wird diese als eine Superposition von zwei linear polarisierten Wellen mit einer relativen Phasenverschiebung von  $\pi/2$  betrachtet. D.h. wir ersetzen  $\vec{q}$  in (3.7) durch  $\vec{q} \pm i\vec{q}'$ , wobei  $\vec{q} \perp \vec{q}'$ ,  $|\vec{q}| = |\vec{q}'|$ , und ersetzen  $\vec{\epsilon}$  in (3.9) durch  $\vec{\epsilon} \pm i\vec{\epsilon}'$ , wobei  $\vec{\epsilon} \perp \vec{\epsilon}'$ ,  $|\vec{\epsilon}| = |\vec{\epsilon}'|$  (siehe Abbildung 10(b)).

Aufgrund dieser Ersetzung in (3.7) erhält man für den Elektronenstreuquerschnitt

$$\begin{aligned} \frac{\partial^2 \sigma}{\partial \Omega \partial E} = & \frac{4\gamma^2 k_f}{a_0^2 k_i} \sum_{i,f} \left[ \frac{1}{q^4} \left| \langle f | \vec{q} \cdot \vec{R} | i \rangle \right|^2 + \frac{1}{q'^4} \left| \langle f | \vec{q}' \cdot \vec{R} | i \rangle \right|^2 \right. \\ & \mp \frac{1}{q^2 q'^2} \langle f | \vec{q} \cdot \vec{R} | i \rangle \langle i | \vec{q}' \cdot \vec{R} | f \rangle \\ & \left. \pm \frac{1}{q^2 q'^2} \langle f | i \vec{q}' \cdot \vec{R} | i \rangle \langle i | \vec{q} \cdot \vec{R} | f \rangle \right] \delta(E_f - E_i - E) \quad (3.10) \end{aligned}$$

Die ersten beiden Terme in der eckigen Klammer sind dynamische Formfaktoren (vergleiche (3.3)), welche die inelastische Streuung je einer Elektronenwelle beschreiben. Sie beschreiben die inelastische Streuung einer einfallenden ebenen Welle in eine auslaufende Welle mit Wellenvektorübertrag  $\vec{q}$  und Energieverlust  $E$ . Die letzten beiden sind inelastische Interferenzterme, welche als gemischte dynamische Formfaktoren  $S(\vec{q}, \vec{q}', E)$  (MDFF – „Mixed Dynamic Form Factor“) bezeichnet werden.

$$S(\vec{q}, \vec{q}', E) = \sum_{i,f} \langle f | \vec{q} \cdot \vec{R} | i \rangle \langle i | \vec{q}' \cdot \vec{R} | f \rangle \delta(E_f - E_i - E) \quad (3.11)$$

Diese sind gekennzeichnet durch zwei Wellenvektorüberträge. Der MDFF ist eine wichtige Größe in der Beschreibung des Chiraldichroismus bei inelastischer Elektronenstreuung (EMCD). Der MDFF fand bereits Anwendung bei der Beschreibung der Interferenz von inelastisch gestreuten Elektronen [KR85, SNSJ00, SW05]. Somit lässt sich (3.10) in kompakterer Form schreiben

$$\frac{\partial^2 \sigma}{\partial \Omega \partial E} = \frac{4\gamma^2 k_f}{a_0^2 k_i} \left[ \frac{S(\vec{q}, E)}{q^4} + \frac{S(\vec{q}', E)}{q'^4} - \frac{S(\vec{q}, \vec{q}', E)}{q^2 q'^2} + \frac{S(\vec{q}', \vec{q}, E)}{q^2 q'^2} \right] \quad (3.12)$$

Diese Situation ist sehr ähnlich zum Doppelspaltexperiment. Bei diesem wird das Interferenzmuster beschrieben durch einen Kopplungsterm zwischen ebenen Wellen mit Wellenvektoren  $\vec{k}$  und  $\vec{k}'$  [Rub07, SNSJ00].

Zur Vollständigkeit der Darstellung der Analogie ergibt sich bei Röntgenabsorption durch die Ersetzung  $\vec{\epsilon} = \vec{\epsilon} + i\vec{\epsilon}'$  für zirkulare Polarisation für den Streuquerschnitt

$$\begin{aligned} \sigma = & 4\pi^2 \hbar \alpha \omega \sum_{i,f} \left[ \left| \langle f | \vec{\epsilon} \cdot \vec{R} | i \rangle \right|^2 + \left| \langle f | \vec{\epsilon}' \cdot \vec{R} | i \rangle \right|^2 \right. \\ & - \langle f | \vec{\epsilon} \cdot \vec{R} | i \rangle \langle i | i\vec{\epsilon}' \cdot \vec{R} | f \rangle \\ & \left. + \langle f | i\vec{\epsilon}' \cdot \vec{R} | i \rangle \langle i | \vec{\epsilon} \cdot \vec{R} | f \rangle \right] \times \delta(E_f - E_i - E) \end{aligned} \quad (3.13)$$

Abschließend zeigt der Vergleich der Streuquerschnitte mit zwei senkrecht zueinander stehenden Polarisations- (3.13) bzw. Impulsübertragsvektoren (3.10), dass die Analogie zwischen inelastischer Elektronenstreuung und Röntgenabsorption bei linearen Polarisation, welche beim Vergleich der Gleichungen (3.7) und (3.9) entdeckt wurde, sich auf zirkulare Polarisation erweitern lässt.

## 4 Zum „2 Blenden“-Betrieb

Wie in Abschnitt 3.4 beschrieben, werden EMCD-Messungen mit der intrinsischen Methode durchgeführt, bei der eine monokristalline Probe durch exakte Orientierung und optimale Dicke (siehe Tabelle 2, [Hur08, RRS07]) den parallelen oder konvergenten Strahl derart teilt, dass in der Beobachtungsebene zwei Bragg-Reflexe angeregt werden, die eine Phasendifferenz von  $\pi/2$  zueinander aufweisen (siehe Unterkapitel 3.4 und Abbildung 12(a)).

Für EMCD-Messungen mit der intrinsischen Methode werden an die Proben eine Reihe von Anforderungen gestellt, welche bereits in Unterkapitel 3.4 eingeführt wurden und ausführlich in der Referenz [Hur08] erläutert werden. In Abbildung 12(a) ist schematisch die experimentelle Anordnung für eine intrinsische Messung dargestellt. Die einfallende Elektronenwelle wird vom Einkristall derart gebeugt, dass in der Beugungsebene zwei Bragg-Reflexe (rot) angeregt werden, welche eine Phasendifferenz von  $\pi/2$  aufweisen. Der Abstand dieser beiden Punkte definiert den Durchmesser des Thaleskreis. Die Elektronen-Energieverlust-Spektren (EELS – „Electron Energy Loss Spectrometry“) werden an den Positionen **A** (grün) und **B** (blau) aufgenommen. In allen Punkten auf dem Thaleskreis gilt für die Wellenvektorüberträge:  $\vec{q}_1 \perp \vec{q}_2$  und  $\vec{q}'_1 \perp \vec{q}'_2$ .

Ausgehend von der Überlegung eine zirkular polarisierte Welle als Superposition zweier linear polarisierter Wellen zu schreiben würde man optimale Ergebnisse erwarten, wenn zusätzlich zur Orthogonalität der  $q_i$  und  $q'_i$  folgendes gelten würde:  $|\vec{q}_1| = |\vec{q}_2|$ ,  $|\vec{q}'_1| = |\vec{q}'_2|$ . Wie allerdings in [VHR<sup>+</sup>08a, VHR<sup>+</sup>08b] gezeigt wird, wird ein maximales Signal, d.h. ein maximales Signal/Rausch-Verhältnis unter anderen Bedingungen erhalten. Darauf wird in dieser Arbeit nicht näher eingegangen.

Eine Möglichkeit ein EMCD-Signal zu erhalten ist die Differenz der an den Punkten **A** und **B** aufgenommenen Spektren (siehe Abschnitt 3.4).

Abbildung 12(b) zeigt den in dieser Arbeit vorgeschlagenen und zu untersuchenden neuen Aufbau. Eine ebene Elektronenwelle (gelber Strahl) fällt auf die Doppellochblende, so dass zwei kohärente Teilwellen entstehen. Diese sollen ebenfalls als ebene Wellen betrachtet werden. Eine dieser Blenden ist mit einer Phasenplatte ausgestattet; im konkreten Fall von Abbildung 12(b) wird der Teilstrahl, welcher die rechte Blende passiert, durch die Phasenplatte (rot) in der Phase verschoben. Dazu hat das Potential der Ringelektrode der Phasenplatte auf einem vordefinierten Potential zu liegen, welches durch die Simulationen in Kapitel 6 und den Formalismus aus Kapitel 5.2 zu bestimmen ist. Die beiden Strahlen werden durch eine Linse auf die Probe fokussiert und erzeugen in der Beugungsebene zwei Punkte – vergleichbar zu den Bragg-Reflexen in Abbildung 12(a).

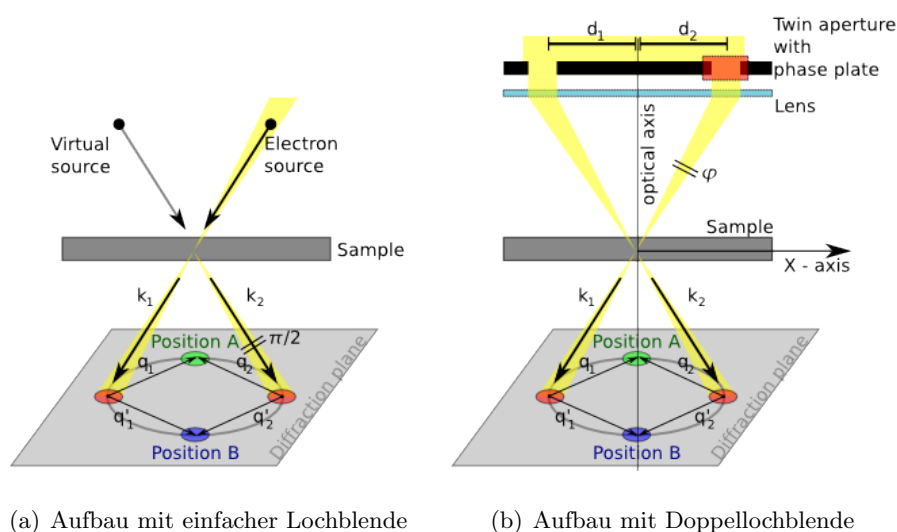


Abbildung 12: Schematischer Aufbau für ein EMCD-Experiment

(a) – Intrinsische Methode (wie beschrieben in Unterkapitel 3.4) mit einem einfallenden Elektronenstrahl und der Probe, welche als Strahlteiler dient. In der Streuebene befinden sich zwei angeregte Bragg-Reflexe (rote Punkte), durch welche ein Thaleskreis definiert wird. Aufgrund der korrekten Orientierung und Dicke weisen die beiden Bragg-Reflexe eine Phasendifferenz von  $\pi/2$  auf. Auf dem Thaleskreis sind die beiden Messpunkte derart platziert, dass die vier Punkte die Ecken eines Quadrates bilden. Nachbildung einer Graphik aus [Hur08]

(b) – In dieser Arbeit angestrebter Aufbau mit Doppellochblende und Phasenplatte. Durch die Phasenplatte wird ein Strahl um  $\varphi$  phasenverschoben, so dass sich in der Probenebene am Ort des Maximums eine Phasendifferenz von  $\pi/2$  ergibt.

Es stellt sich die Frage, ob eine Doppellochblende (siehe Abbildung 12(b) und Abbildung 13) mit einer Phasenplatte die gewünschten Strahleigenschaften für einen fokussierten Strahl in der Probenebene liefert. Ein fokussierter Strahl ist von Interesse, da dadurch die Anzahl der Atome, an denen inelastische Stöße stattfinden, minimiert wird. Mit anderen Worten: Die Intensität wird auf eine minimale Fläche auf der Probe konzentriert, d.h. das gemessene EELS-Spektrum ist über weniger Atome bzw. Atomsäulen der Probe gemittelt im Vergleich zu einem parallelen, nicht-konvergierenden Strahl. Des weiteren ist ein fokussierter Strahl für einen STEM („Scanning Transmission Electron Microscope“)-Betrieb notwendig.

Werden die beiden benötigten kohärenten Teilstrahlen durch eine solche Doppellochblende bereit gestellt und die benötigte Phasendifferenz durch eine Phasenplatte eingestellt, so ist es nicht mehr erforderlich die Probe als Strahlteiler zu verwenden. D.h. der Strahl muss nicht mehr durch einen Einkristall geteilt werden. Damit wäre die Untersuchung nicht mehr auf

monokristalline Proben mit speziellen Eigenschaften (Orientierung, Dicke) beschränkt. Durch den Wegfall der Notwendigkeit einer Präparation auf eine exakte Probendicke, einer exakten Orientierung der Kristallausrichtung und aufgrund des fokussierten Strahls, sind EMCD-Messungen ebenfalls im STEM-Betrieb denkbar.

Zur Verifikation dieser Vermutungen sollen im weiteren Verlauf dieser Arbeit die Wellenfunktion und die Phasendifferenz in der hinteren Brennebene theoretisch betrachtet werden (Kapitel 6) mit dem Ziel, an einem Messpunkt auf der Probe die benötigte Phasendifferenz von  $\pi/2$  bei maximaler Intensität zu erhalten. Dazu sollen insbesondere anhand numerischer Berechnungen die optimalen Parameter für die Blenden (Radius, Abstände) und die Phasenplatte (Phasenverschiebung  $\varphi$ ) bestimmt werden.

Des weiteren soll eine solche Doppellochblende hergestellt und als Kondensorblende in ein Transmissionselektronenmikroskop (TEM) eingebaut werden um die Möglichkeit eines STEM-Betriebs zu untersuchen (Kapitel 7). Dazu sollen kontrastreiche Proben – Goldpartikel auf einer Kohlenstoffmembran – im STEM-Modus betrachtet werden.

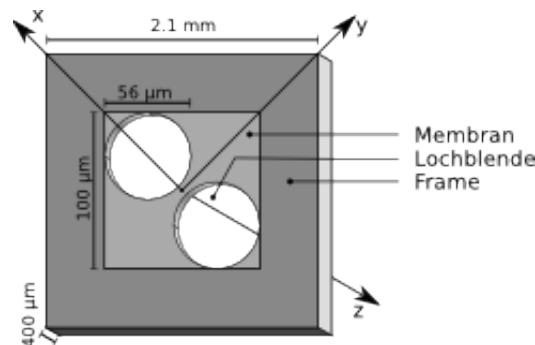


Abbildung 13: Doppellochblende – Perspektivische Darstellung, wobei die  $x$ - und  $y$ -Achse die Blendenebene aufspannen, während die  $z$ - (optische) Achse senkrecht zu dieser steht. Die beiden kreisrunden Blendenlöcher sind entlang der  $x$ -Achse verschoben und befinden sich in einer  $100\mu\text{m} \times 100\mu\text{m}$  messenden Membran mit Dicke  $30\text{nm}$ , welche von einem Rahmen („Frame“) gehalten wird. Der Frame misst  $2\text{mm} \times 2\text{mm}$  und hat eine Dicke von  $400\mu\text{m}$ .

Abbildung 13 zeigt eine perspektivische Darstellung einer Doppellochblende ohne Phasenplatte zur Verdeutlichung der geometrischen Verhältnisse. Die Mittelpunkte der Lochblenden sind entlang der  $x$ -Achse vom Zentrum der Membran verschoben. Per Definition wird in den folgenden Kapiteln die optische Achse als  $z$ -Achse bezeichnet, während die  $x$ - und  $y$ -Achsen senkrecht dazu stehen und die Blendenebene und die hintere Brennebene definieren. Informationen zur Membran, zur Herstellung der Doppellochblende



und qualitative Bewertung derselben folgen in Unterkapitel 7.2.

## 5 Grundlagen – Fourieroptik und Phasenplatte

In diesem Kapitel werden die benötigten theoretischen Beschreibungen der Elektronen- bzw. Fourieroptik geschildert, welche für die Berechnungen der Wellenfunktionen in Kapitel 6 benötigt werden. Außerdem werden in kompakter Form Aussehen einer elektrostatischen Phasenplatte – auch Boersch-Phasenplatte genannt – und wichtige praktische und theoretische Erkenntnisse über solche Phasenplatten rekapituliert, welche in den Kapiteln 6 für theoretische Betrachtungen und in Kapitel 7 für den Bau einer Phasenplatte von Interesse sind.

### 5.1 Fourieroptik

In der Fourieroptik wird das Bild von einem Gegenstand durch eine Linse in zwei Schritten geformt: eine Fouriertransformation der den Gegenstand beschreibenden Wellenfunktion ergibt die Wellenfunktion in der hinteren Brennebene (BFP – „Back Focal Plane“). Eine weitere Transformation erzeugt das Bild. Dies soll im Verlauf dieses Unterkapitels näher beschrieben werden. Zur Verdeutlichung zeigt Abbildung 14 die Anordnung von Gegenstand, Bild, Linse und Brennebenen. In diesem Abschnitt ist in erster Linie die Wellenfunktion in der hinteren Brennebene von Interesse. Diese Funktion ist die Fouriertransformierte der Wellenfunktion in der Gegenstandsebene, in welcher sich die in Kapitel 4 eingeführte Doppellochblende befinden wird.

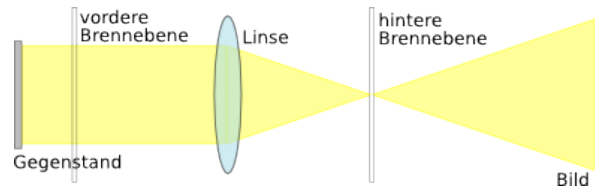


Abbildung 14: Die Wellenfunktion, welche den Gegenstand verlässt, wird durch die Linse in der hinteren Brennebene fokussiert und propagiert frei weiter bis zur Bildebene. Dies entspricht zwei Fouriertransformationen.

Dazu sollen in diesem Unterkapitel wichtige Erkenntnisse aus der Wellenoptik anhand von Referenz [Spe03] präsentiert werden. Dabei werden Elektronenlinsen als idealisierte dünne Linsen beschrieben, deren wellenoptische Eigenschaften diskutiert werden. Das Betragsquadrat der komplexen Wellenfunktion  $\psi(x, y)$  wird als Intensität interpretiert.

Die freie Ausbreitung einer Welle mit Wellenfunktion  $\psi(x, y)$  entlang der  $z$ -Achse über die Distanz  $\Delta z$  ist gegeben durch [Spe03]

$$\psi'(x, y) = A \psi(x, y) \otimes \mathcal{P}_{\Delta z}(x, y), \quad (5.1)$$

wobei  $A$  eine komplexe Konstante,  $\otimes$  der Faltungsoperator und

$$\mathcal{P}_{\Delta z}(x, y) = e^{-\left(\frac{i\pi(x^2+y^2)}{\lambda\Delta z}\right)} \quad (5.2)$$

der Fresnelpropagator ist.  $i$  ist dabei die imaginäre Einheit und  $\lambda$  die Elektronenwellenlänge.

Die Ausbreitung einer Elektronenwelle ist quantitativ beschrieben durch (5.1). Eine qualitative Beschreibung liefert das Huygens'sche Prinzip.

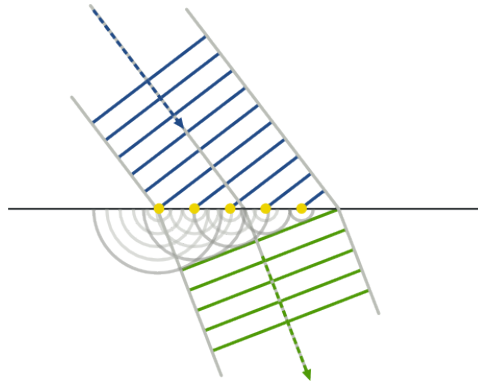


Abbildung 15: Brechung einer ebenen Wellenfront nach dem Huygens'schen Prinzip. Bild entnommen aus [Wik]

Das Huygens'sche Prinzip besagt, dass jeder Punkt einer Wellenfront als Ausgangspunkt einer neuen Kugelwelle, einer so genannten Elementarwelle, betrachtet werden kann. Dadurch ergibt sich eine neue Wellenfront durch die Überlagerung all dieser Elementarwellen. Die sich weiter ausbreitende Wellenfront kann als äußere Einhüllende der Elementarwellen betrachtet werden, da sich die Elementarwellen im selben Medium und mit gleicher Geschwindigkeit ausbreiten wie die ursprüngliche Welle. Beim Übergang in ein optisch dichteres oder dünneres Medium ändert sich die Ausbreitungsgeschwindigkeit und damit auch die Ausbreitungsrichtung der Welle, was sich als Brechung bemerkbar macht. Die Brechung gemäß diesem Prinzip wird in Abbildung 15 schematisch dargestellt. Am Rand eines Hindernisses (z. B. Spalt) führt das Huygens'sche Prinzip zur Beugung.

Bei der Abbildung eines Punktes durch eine Linse verursacht eine limitierende Blende eine Unschärfe im Bild. Dies wird als Impulsansprechverhalten („lens impulse response“) bezeichnet [Spe03, Goo04]. Die Wellenfunktion in der Eintrittsebene einer Linse, welche eine Distanz  $U$  von einer Punktquelle entfernt ist, ist gegeben durch

$$\psi'(x, y) = A\delta(x, y) \otimes \mathcal{P}_U(x, y) = Ae^{-\frac{i\pi(x^2+y^2)}{\lambda U}} \quad (5.3)$$

Dies ist eine quadratische Beschreibung für eine vom Objektpunkt divergierende Kugelwelle. Für eine konvergierende Welle wird das Vorzeichen im Exponenten umgekehrt.

In Analogie zwischen Elektronen- und Lichtoptik wird für die Wirkung einer magnetischen Linse eine fokussierende Phasenverschiebung eingeführt. Trifft eine ebene Welle auf eine solche Linse, ist die resultierende auslaufende Welle proportional zu

$$\psi_{aus} \propto e^{\frac{\pi i x^2}{\lambda f}}, \quad (5.4)$$

wobei die fokussierende Phasenverschiebung  $(2\pi/\lambda)(x^2/2f)$  beträgt,  $f$  die Brennweite und  $x$  eine kartesische Koordinate in der Linsenebene ist (siehe Abbildung 16).

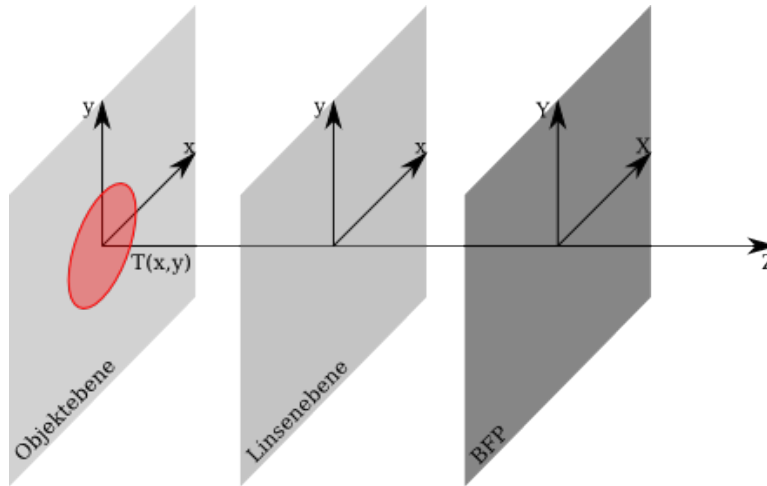


Abbildung 16: Die Objekt- und Linsenebene werden aufgespannt durch kartesische  $(x, y)$ -Systeme. In der Objektebene soll sich ein Objekt mit Transmissionsfunktion  $T(x, y)$  befinden. Die hintere Brennebene (BFP) wird aufgespannt durch das kartesische  $(X, Y)$ -System. Dieses System ist durch (5.10) verbunden mit dem  $(u, v)$ -System des Fouriertraums durch  $X = u\lambda f$  und  $Y = v\lambda f$ , wobei  $\lambda$  die Wellenlänge und  $f$  die Brennweite sind.

Die Abbildung durch eine einfache Linse lässt sich durch die Anwendung des Fresnelpropagators und Verfolgen der Wellenfunktion durch das Linsensystem verstehen. Dazu zeigt Abbildung 16 die  $(x, y)$ -Koordinatensysteme der betrachteten Ebenen. Bei kohärenter Beleuchtung eines Objekts mit Transmissionsfunktion  $T(x, y)$  ergibt sich die Wellenfunktion in der Linsenebene nach [Spe03] mit  $A = i/(\lambda\Delta_z)$  zu

$$\psi_1(x, y) = \frac{i}{\lambda U} \psi_0(x, y) \otimes \mathcal{P}_U(x, y), \quad (5.5)$$

wobei  $\psi_0(x, y) = \psi_{ein}(x, y) \cdot T(x, y)$  und die Welle über die Gegenstandsweite  $U$  frei propagiert wurde.  $\psi_{ein}(x, y)$  ist dabei die das Objekt beleuchtende Wellenfunktion. Für eine kreisrunde Blende mit Mittelpunkt auf der optischen Achse ( $x = 0, y = 0$ ) und Radius  $R$  z.B. ergibt sich die Transmissionsfunktion (vergleiche Abbildung 16) zu

$$T(x, y) = \begin{cases} 1, & x^2 + y^2 \leq R^2 \\ 0, & \text{sonst} \end{cases}. \quad (5.6)$$

Direkt hinter der Linse ergibt sich die Wellenfunktion zu

$$\psi_2(x, y) = \psi_1(x, y) e^{\frac{i\pi(x^2+y^2)}{\lambda f}}. \quad (5.7)$$

Dies ist die Wellenfunktion  $\psi_1$  in der Linsenebene multipliziert mit dem die Linsenwirkung beschreibenden Term aus (5.4). Die komplexe Amplitude in der hinteren Brennebene (BFP) ist schließlich gegeben durch die Propagation über die Strecke  $f$

$$\psi_{BFP}(X, Y) = \frac{i}{\lambda f} \psi_2(x, y) \otimes \mathcal{P}_f(x, y) \quad (5.8)$$

$$\begin{aligned} \psi_{BFP}(X, Y) &= \frac{i}{\lambda f} e^{\left(-\frac{i\pi(X^2+Y^2)}{\lambda} \left[\frac{1}{f} - \frac{U}{f^2}\right]\right)} \\ &\times \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \psi_0(x, y) e^{2\pi i(ux+vy)} dx dy, \end{aligned} \quad (5.9)$$

wobei  $u = X/\lambda f = \theta/\lambda$  und  $v = Y/\lambda f = \beta/\lambda$  ist. Es wurde in (5.8) der Propagator gemäß (5.2) eingesetzt und die Welle über die Brennweite  $f$  frei propagiert. Da in Elektronenmikroskopen die Gegenstandsweite in etwa der Brennweite der Objektivlinse entspricht, wird in Rechnungen meist die Näherung  $U = f$  verwendet. Dies hat den Vorteil, dass der quadratische Phasenterm verschwindet. Dieses wichtige Ergebnis zeigt, dass die Wellenfunktion in der Brennebene  $\psi_{BFP}$  proportional zur Fouriertransformierten der Wellenfunktion  $\psi_0$  ist, welche das Objekt beschreibt. Somit lässt sich (5.9) einfacher schreiben als

$$\psi_{BFP}(X, Y) = \frac{i}{\lambda f} F(u, v) \quad (5.10)$$

$$\begin{aligned} F(u, v) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \psi_0(x, y) e^{2\pi i(ux+vy)} dx dy \\ &= \mathcal{F}(\psi_0(x, y)), \end{aligned} \quad (5.11)$$

wobei  $\theta$  und  $\beta$  Winkel zur optischen Achse bezeichnen, und  $\mathcal{F}(\psi_0(x, y))$  ist die Fouriertransformierte der Objektwellenfunktion  $\psi_0$ .

## 5.2 Phasenplatte

Die Notwendigkeit einer  $\pi/2$  Phasendifferenz zwischen den beiden Teilstrahlen legt die Verwendung eines phasenschiebenden Elementes hinter einer der beiden Blendenlöcher nahe. Solche Phasenplatten sind bereits bekannt. Erste Realisierungen [PJ72, BR70] bestanden aus dünnen, amorphen Filmen mit einem Loch in der Mitte, so dass ein Teilstrahl ohne Phasenverschiebung passieren kann. Parsons et. al. erzielten 1972 mit einem Kohlenstofffilm der Dicke  $220\text{\AA}$  und einem Lochdurchmesser von  $6\mu\text{m}$  eine Phasenschiebung von nahezu  $\pi/2$  [PJ72].

Solche filmbasierenden Phasenplatten sind mit Bildstörungen verbunden wegen Variationen in der Filmdicke, Aufladungen des Films und Streuprozessen im Film. Stattdessen finden elektrostatische (Boersch) Phasenplatten [Boe47, MOT98, MBS<sup>+</sup>07, AHA<sup>+</sup>08] (siehe Abbildung 17) oder anamorphotische Phasenplatten [SBS<sup>+</sup>08, SBRB07] in der Phasenkontrastelektronenmikroskopie [MOT98] Verwendung, die nach Möglichkeit die Anwesenheit von Material im Strahlengang der Elektronen vermeiden. Phasenplatten, welche den Aharonov-Bohm-Effekt ausnutzen, wurden ebenfalls vorgeschlagen [EB08].

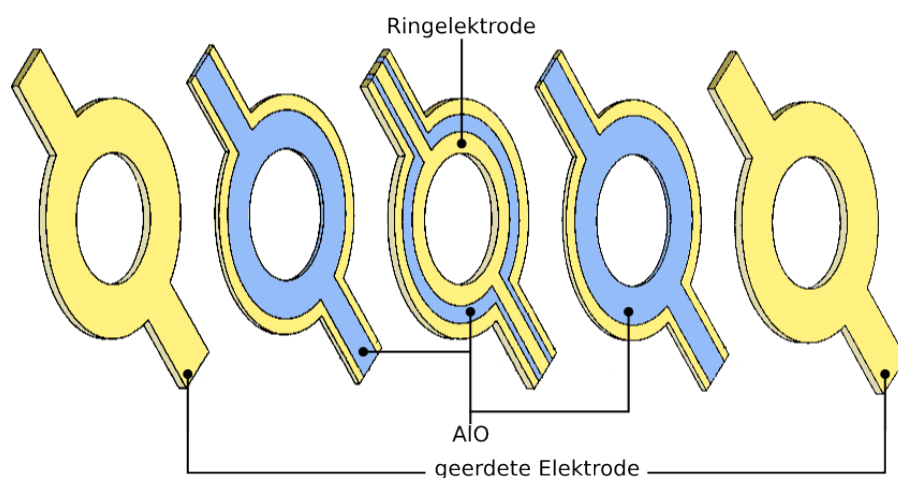


Abbildung 17: Explosionsdarstellung einer Boersch-Phasenplatte bestehend aus fünf Schichten. Das Potential der Ringelektrode (Aluminium oder Gold, Dicke  $250\text{nm}$ , Durchmesser  $1\mu\text{m}$ ) ist nach außen hin abgeschirmt aufgrund der umgebenden geerdeten Elektroden (Gold, Dicke  $250\text{nm}$ ), welche durch isolierende Schichten (AIO, Dicke  $100\text{nm}$ ) von der Ringelektrode getrennt sind. Die Abschirmung wird vervollständigt durch den Zylindermantel aus Gold, welcher die geerdeten Elektroden verbindet. Quelle: [MOT98]

Boersch-Phasenplatten sind experimentell erprobt [AHA<sup>+</sup>08] und haben sich als mechanisch stabil erwiesen [MBS<sup>+</sup>07] und verringern nicht die Möglichkeit an einem Mikroskop hochaufgelöste Bilder zu betrachten.

### 5.2.1 Prinzip der Potentialbarriere

Die zugrunde liegende Idee hinter einer Phasenplatte für Elektronen entspringt der Quantenmechanik. Eine Elektronenwelle, welche eine Potentialbarriere passiert, hat vor und nach der Barriere dieselbe Wellenlänge. Im Bereich der Barriere hängt diese allerdings von der Höhe der Barriere ab. Somit ergibt sich im Vergleich zu einer Welle, welche keine Barriere überwinden muss, eine Phasenverschiebung.

Zur Verdeutlichung dieser Idee, beschränkt sich dieser Abschnitt auf eine nicht-relativistische Beschreibung. In Abschnitt 5.2.3 wird eine exakte, relativistische Beschreibung der Wirkung einer Phasenplatte gegeben.

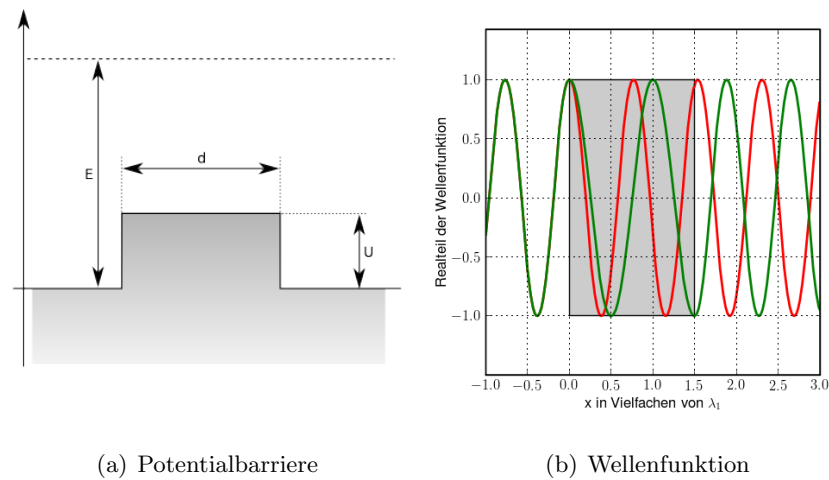


Abbildung 18:

(a) Querschnitt durch eine Potentialbarriere (vgl. [Sch02])

(b) Realteile der ungestörten Wellenfunktion (ohne Barriere, rot) und der Wellenfunktion, welche über die Barriere (grün) läuft. Die Barriere ist durch eine graue Hinterlegung angedeutet und hat eine Breite von  $1.5\lambda_1$ . Die rote Welle hat durchgehend die Wellenlänge  $\lambda_1 = 2\pi/k_1$ , wobei  $k_1 = 1$ . Die grüne Welle hat vor und nach der Barriere ebenfalls die Wellenlänge  $\lambda_1$ , aber im Bereich der Barriere die Wellenlänge  $\lambda_2 = 2\pi/k_2$ , wobei  $k_2 = 1.3$ .

In der Quantenmechanik werden die Wellenfunktionen vor, im Bereich und nach der Barriere durch die Lösung der Schrödinger-Gleichung bestimmt. Dies kann z.B. in Referenz [Sch02] nachgelesen werden. Die Wellenzahlen bzw. Wellenlängen in diesen Bereichen sind (vgl. [Sch02]) gegeben durch:

$$\text{vor/nach Barriere : } k_1 = \frac{\sqrt{2mE}}{\hbar} \quad (5.12)$$

$$\lambda_1 = \frac{2\pi}{k} = \hbar\pi\sqrt{\frac{2}{mE}} \quad (5.13)$$

$$\text{in Barriere : } k_2 = \frac{\sqrt{2m(E-U)}}{\hbar} \quad (5.14)$$

$$\lambda_2 = \hbar\pi\sqrt{\frac{2}{m(E-U)}}, \quad (5.15)$$

wobei  $m$  die Elektronenmasse,  $E$  die kinetische Energie des Elektrons und  $U$  die Höhe der Potentialbarriere sind.

In diesem einfachen Modell ergibt sich die Phasendifferenz zwischen einer ebenen Welle, welche sich über eine solche Barriere ausbreitet, und einer ebenen Welle, welche sich frei (neben der Barriere) ausbreitet aus der Differenz der Wellenzahlen multipliziert mit der Breite  $d$  der Barriere:

$$\Delta\phi = k_1d - k_2d = (k_1 - k_2)d = \sqrt{2}d \frac{\sqrt{mE} - \sqrt{m(E-U)}}{\hbar} \quad (5.16)$$

Abbildung 18(a) zeigt einen schematischen Querschnitt durch eine Potentialbarriere mit Breite  $d$  und Höhe  $U$ . Abbildung 18(b) zeigt beispielsweise den Realteil der Wellenfunktionen für ein Elektron, welches sich über die gesamte Breite frei ausbreitet (rote Kurve) und für ein Elektron, welches eine Potentialbarriere überwindet (grüne Kurve). Die Barriere ist angedeutet durch den grauen Quader.

Wie zu erkennen ist, wird im Bereich der Potentialbarriere die grüne Welle gegenüber der roten „verzögert“, so dass nach der Barriere die beiden Wellen gegeneinander phasenverschoben sind.

Im folgenden soll nun die Phasenverschiebung aufgrund einer Phasenplatte im korrekten Formalismus der Elektronenoptik betrachtet und ein Ausdruck für die Phasenverschiebung durch eine solche Platte angegeben werden.

### 5.2.2 Einzellinse

In der Elektronenoptik bezeichnet man eine Anordnung von drei hintereinander angeordneten Lochblenden, deren äußeren beiden Elektroden auf gleichem Potential liegen, als Einzellinse. Die Elektronen besitzen vor und hinter einer solchen Einzellinse die gleiche Geschwindigkeit [BS04]. Den Potentialverlauf einer solchen Einzellinse zeigt Abbildung 19 anhand von



Äquipotentiallinien (Abbildung 19(a)) und eines Potentialgebirges (Abbildung 19(b)), welches in der Mitte einen Sattelpunkt hat. Eine Umdeutung von Linien gleichen elektrischen Potentials zu Höhenlinien erlaubt ein anschauliches mechanisches Bild der Verhältnisse in elektrischen Linsen, indem man das Elektron als Kugel auffasst, die sich nach den Gesetzen der Mechanik im Schwerkraftfeld bewegt [BS04].

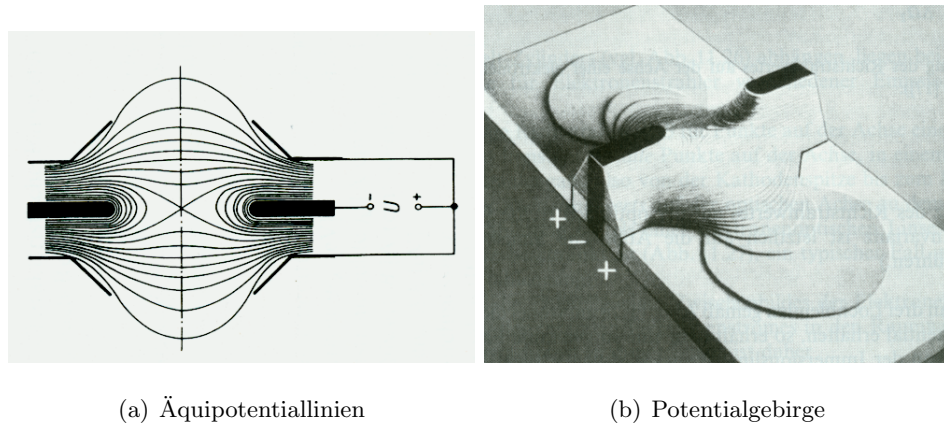


Abbildung 19: Potential einer elektrostatischen Einzellinse. Ein ähnliches, aber schwächeres Feld wird durch die Ringelektrode der Phasenplatte erzeugt. Quelle: [BS04]

- (a) Darstellung der Äquipotentiallinien, der drei Ringelektroden im Querschnitt und einer Spannungsquelle mit Spannung  $U$   
 (b) Potentialgebirge

Vergleicht man die Abbildungen 17 und 19 so lässt sich erkennen, dass die Boersch-Phasenplatte mit der auf einem definierten Potential liegenden Ringelektrode und den beiden abschirmenden, auf Masse liegenden Elektroden im Sinne der Definition des vorherigen Absatzes eine Einzellinse darstellt. Die zu erwartende Linsenwirkung ist dabei laut Referenz [MT96] minimal: Für eine 50keV Elektronenwelle, die am Rand einer Ringelektrode (Radius: 500nm, Potential: 0.54V) einfällt, wird ein Ablenkungswinkel von  $4.0 \times 10^{-9}$  rad angegeben. Bei einer Erhöhung der Elektronenenergie auf 300keV erwartet man, dass sich dieser Winkel verkleinert.

### 5.2.3 Konstanz der Phasenverschiebung

Im Folgenden soll ein Ausdruck für die Phasenschiebung durch die Phasenplatte in Abhängigkeit von der Dicke und dem Potential an der Ringelektrode hergeleitet werden. Des weiteren soll gezeigt werden, dass die Phasenverschiebung, welche ein Elektron beim Passieren einer Ringelektrode erfährt, unabhängig vom Abstand des Elektrons zum Mittelpunkt des

Ringes ist und dass die Linsenwirkung der Phasenplatte vernachlässigbar ist („negligible lens approximation“).

In Referenz [MT96] wird eine konstante Phasenverschiebung durch eine Boersch-Phasenplatte theoretisch begründet. Diese ist unabhängig vom Radius und gilt nur für geringe angelegte Spannungen. Diese theoretische Begründung soll im Folgenden in kompakter Weise betrachtet werden.

Die Phasenverschiebung einer Elektronenwelle, welche sich durch das Potential  $\Phi$  innerhalb einer Ringelektrode einer Boersch-Phasenplatte (PP – Phase Plate) ausbreitet, ist nach Referenz [Rei84] gegeben durch

$$\Delta\varphi_{PP} = \frac{\pi e}{\lambda E_{kin}} \frac{E_0 + E_{kin}}{E_0 + E_{kin}/2} \int_s \Phi(s) ds, \quad (5.17)$$

wobei  $e$  die Ladung,  $\lambda$  die Wellenlänge,  $E_{kin}$  die kinetische Energie und  $E_0 = m_e c^2$  die Ruheenergie des Elektrons sind. Dabei ist  $m_e$  die Ruhemasse des Elektrons und  $c$  die Lichtgeschwindigkeit im Vakuum.  $s$  ist der Pfad, entlang welchem die Elektronen das Potential durchqueren. Dieser Pfad soll parallel zur optischen ( $z$ -) Achse (vergleiche Abbildung 20) verlaufen.

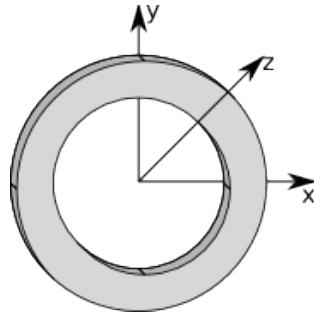


Abbildung 20: Schematische Darstellung einer Ringelektrode und des verwendeten Koordinatensystems. Die Mittelachse des Rings fällt im gezeigten Fall mit der  $z$ -Achse zusammen.

Abbildung 20 zeigt das verwendete Koordinatensystem. Das elektrostatische Skalarpotential  $\Phi$  innerhalb des Rings genügt einer drei-dimensionalen Laplace-Gleichung

$$\left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) \Phi(x, y, z) = 0. \quad (5.18)$$

Diese Gleichung wird nach  $z$  integriert. Durch die Permutation dieser Integration und der Differentiation nach  $z$  und die Definition eines integrierten Potentials  $\Psi$  entlang der  $z$ -Achse ergibt sich aus (5.18) eine zwei-dimensionale Laplace-Gleichung

$$\left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) \Psi(x, y) = 0 \quad (5.19)$$

$$\Psi(x, y) = \int_{-\infty}^{\infty} \Phi(x, y, z) dz, \quad (5.20)$$

wobei das integrierte Potential  $\Psi$  durch (5.20) definiert wird. Damit das integrierte Potential  $\Psi$  der Laplace-Gleichung genügt, muss es lateral in der  $xy$ -Ebene auf die Ringelektrode limitiert sein. Des weiteren muss das Potential bei unendlichen Abständen in  $z$ -Richtung gegen Null gehen. Eine Lösung von (5.19) mit den Randbedingungen

$$\begin{aligned} \Psi(x, y)_c &= \Psi_0 = \text{const.} \\ c &: x^2 + y^2 = a^2. \end{aligned}$$

ist gegeben durch

$$\begin{aligned} \Psi(x, y)_s &= \Psi_0 \\ S &: x^2 + y^2 \leq a^2, \end{aligned} \quad (5.21)$$

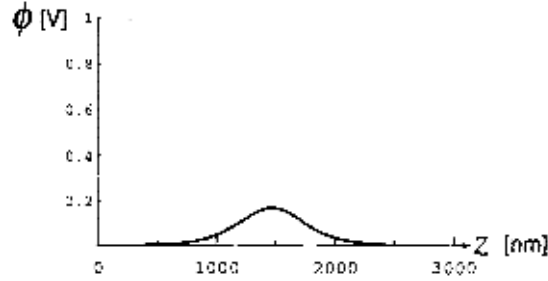
wobei  $a$  der Radius des Ringes ist. Dies ist das zwei-dimensionale Analogon zum konstanten Potential innerhalb einer uniform geladenen Kugel. Zur Verdeutlichung zeigt Abbildung 21 den Verlauf des Potentials  $\Phi$  entlang einer Achse, welche zur Mittelachse des Ringes parallel verläuft, für verschiedene Abstände dieser Achse zur Mittelachse des Ringes. Die Flächeninhalte der beiden Kurven sind gleich.

Da die Linsenwirkung einer Phasenplatte sehr schwach ist, wie zu Beginn des Unterkapitels bereits erwähnt, wäre es wünschenswert diese zu vernachlässigen. Deshalb soll im folgenden Absatz die Vernachlässigung der Linsenwirkung gerechtfertigt werden [MBS<sup>+</sup>07].

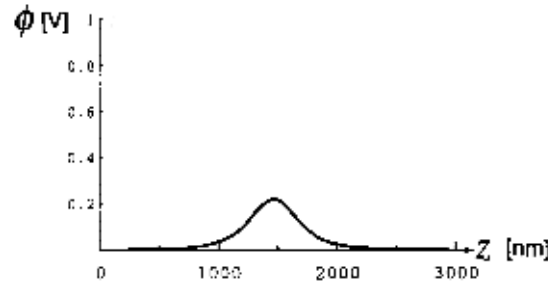
Dazu wird die resultierende Phasenverschiebung  $\Delta\varphi_{Linse}$  in Abhängigkeit des Abstandes  $r$  von der optischen Achse, in welchem ein Elektron die Ringelektrode passiert, betrachtet. Die Phasenverschiebung lässt sich anhand des Gangunterschiedes  $\Delta s$  in Abbildung 22 abschätzen. Die Phasenverschiebung aufgrund des Gangunterschiedes ist

$$\Delta\varphi_{Linse}(r) = \frac{2\pi}{\lambda} \Delta s. \quad (5.22)$$

Die beiden in Abbildung 22 eingezeichneten Winkel sind jeweils  $\alpha$ , da die Schenkel, welche die beiden Winkel bilden, paarweise aufeinander senkrecht



(a) Abstand: 250nm



(b) Abstand: 0nm

Abbildung 21: Verteilung des elektrostatischen Potentials  $\Phi$  aufgrund einer Ringelektrode entlang einer Achse parallel zur Mittelachse ( $z$ -Achse) des Rings. Dabei beträgt der Abstand dieser Achse zur Mittelachse des Ringes 0nm (a) bzw. 250nm (b). Die Flächen unter den Kurven sind gleich und entsprechen dem integrierten Potential  $\Psi$ . Quelle: [MT96]

stehen. Unter Verwendung dieser Tatsache und der Kleinwinkelnäherung ( $\tan \alpha \approx \sin \alpha$ ) erhält man für die Phasenverschiebung folgenden Ausdruck:

$$\sin \alpha = \frac{\Delta s}{r} \quad (5.23)$$

$$\tan \alpha = \frac{r}{F} \quad (5.24)$$

$$\Rightarrow \Delta s = \frac{r^2}{F} \quad (5.25)$$

$$\Rightarrow \Delta \varphi_{Linse}(r) = \frac{2\pi}{\lambda} \frac{r^2}{F}, \quad (5.26)$$

wobei  $F$  die Brennweite der Linse (Ringelektrode) ist. Befindet sich die Ringelektrode in der hinteren Brennebene (BFP) einer vorgeschalteten Linse (z.B. Kondensorlinse), kann man die Ortskoordinate  $r$  in der BFP als Koordinate im Fourierraum  $u = r/\lambda f$  betrachten, wobei  $f$  die Brennweite der

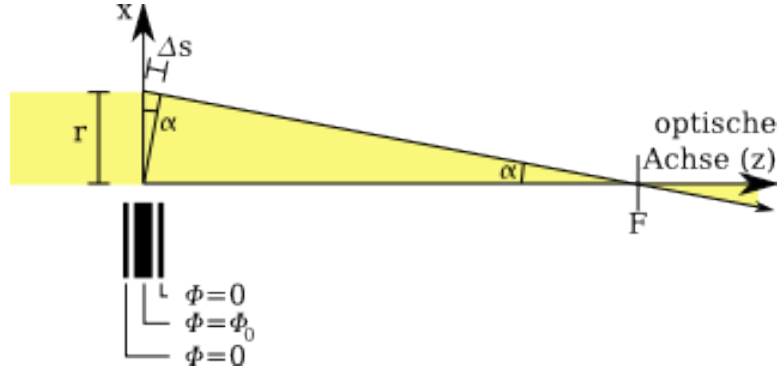


Abbildung 22: Linseneffekt einer Ringelektrode – Bei einer gegebenen endlichen Brennweite  $F$  ergibt sich aus dem Wegunterschied  $\Delta s$  eine Phasenverschiebung in Abhängigkeit der Koordinate  $r$  in der Ringebene. Nachbildung einer Abbildung aus [MBS<sup>+</sup>07]

vorgeschalteten Linse ist. somit ergibt sich für  $\Delta\varphi_{Linse}$

$$\Delta\varphi_{Linse}(u) = 2\pi\lambda \frac{f^2}{F} u^2. \quad (5.27)$$

Diese  $u^2$ -abhängige Linsenwirkung der Phasenplatte kann in der Wellenoptik im Defokusterm  $\exp(i\chi(u))$  mit  $\chi(u) = \pi/2(C_s\lambda^3u^4 - 2\lambda\Delta zu^2)$  durch eine Änderung der Defokusstrecke  $\Delta z$  berücksichtigt werden [MBS<sup>+</sup>07]. Somit ist eine Vernachlässigung der Linsenwirkung gerechtfertigt.

Vernachlässigt man nun die Linsenwirkung einer solchen Phasenplatte in (5.17) so erhält man für die Phasenverschiebung

$$\begin{aligned} \Delta\varphi_{PP} &= \frac{\pi e}{\lambda E_{kin}} \frac{E_0 + E_{kin}}{E_0 + E_{kin}/2} \int_{-\infty}^{\infty} \Phi(x, y, z) dz \\ &= \frac{\pi e}{\lambda E_{kin}} \frac{E_0 + E_{kin}}{E_0 + E_{kin}/2} \Psi_0 \end{aligned} \quad (5.28)$$

wobei  $e$  die Elektronenladung,  $\lambda$  die Elektronenwellenlänge und  $E_{kin} = Ve$  die kinetische Energie des Elektrons sind.  $V$  ist die Beschleunigungsspannung.

Das heißt die Elektronenwelle erfährt eine konstante Phasenverschiebung beim Passieren des elektrischen Feldes der Ringelektrode.  $\Psi_0$  aus (5.28) kann nach Referenz [MT96] als rechtecksförmiger Potentialwall abgeschätzt werden zu

$$\Psi_0 = U \cdot d, \quad (5.29)$$

wobei  $U$  das an die Ringelektrode angelegte Potential und  $d$  die Dicke der Ringelektrode sind.

$d \backslash V$	200nm	300nm	400nm	500nm	600nm	700nm
50kV	0.64V	0.43V	0.32V	0.26V	0.21V	0.18V
100kV	0.85V	0.57V	0.42V	0.34V	0.28V	0.24V
150kV	0.98V	0.66V	0.49V	0.39V	0.33V	0.28V
200kV	1.08V	0.72V	0.54V	0.43V	0.36V	0.31V
250kV	1.15V	0.77V	0.57V	0.46V	0.38V	0.33V
300kV	1.20V	0.80V	0.60V	0.48V	0.40V	0.34V

Tabelle 3: Abhängigkeit des Ringelektrodenpotentials  $U$  von deren Dicke  $d$  (horizontal in der obersten, grau hinterlegten Zeile) und von der Beschleunigungsspannung  $V$  (vertikal in der ersten, grau hinterlegten Spalte) um eine Phasenverschiebung von  $\pi/2$  gemäß Gleichung (5.31) zu erzeugen.

Unter Verwendung der de Broglie Wellenlänge ergibt sich für die Phasenverschiebung  $\Delta\varphi_{PP}$  (siehe (5.28)) aufgrund der Phasenplatte folgender Ausdruck:

$$\lambda = \frac{hc}{pc} = \frac{hc}{\sqrt{E^2 - E_0^2}} \quad (5.30)$$

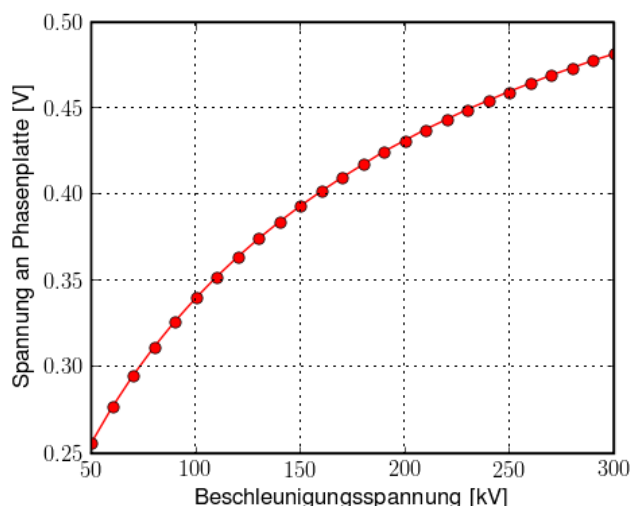
$$\Delta\varphi_{PP} = \frac{2\pi e U d}{hc} \frac{E_0/e + V}{\sqrt{V^2 + 2V E_0/e}}, \quad (5.31)$$

wobei für  $E = E_0 + Ve$  eingesetzt wurde.  $V$  ist die Beschleunigungsspannung.  $E_0/e = 511\text{kV}$  ist die Ruheenergie eines Elektrons dividiert durch die Elementarladung  $e$ .

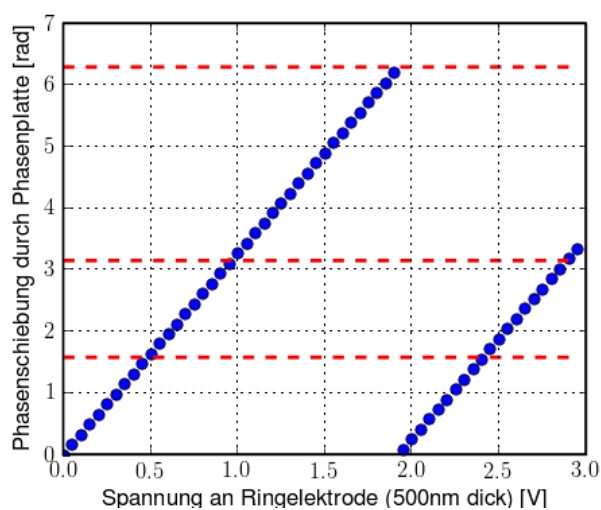
Tabelle 3 zeigt das Potential, welches an der Ringelektrode angelegt werden muss, um eine Phasendifferenz von  $\pi/2$  zwischen den beiden Teilstrahlen zu erzeugen. Diese Potential wurde für unterschiedliche Beschleunigungsspannungen  $V$  und Ringdicken  $d$  berechnet gemäß (5.31). Für höhere Beschleunigungsspannungen wird ein höheres Potential an der Ringelektrode benötigt um die gewünschte Phasenschiebung einzustellen. Während konsistent dazu bei dickeren Ringelektroden ein geringeres Potential benötigt wird. Dieses Ergebnis deckt sich mit den prinzipiellen Überlegungen aus Kapitel 5.2.1.

Dieser Zusammenhang soll ebenfalls anhand von Abbildung 23 verdeutlicht werden. In Abbildung 23(a) ist für eine Ringelektrode der Dicke 500nm

und Beschleunigungsspannungen zwischen 50 und 300kV das zugehörige Potential aufgetragen, welches angelegt werden muss um eine Phasenverschiebung von  $\pi/2$  zu erzielen. Abbildung 23(b) zeigt für Elektronen mit einer kinetischen Energie von 300keV die durch eine 500nm dicke Ringelektrode bewirkte Phasenverschiebung in Abhängigkeit des angelegten Potentials.



(a) Konstante Phasenschiebung von  $\pi/2$



(b) Konstante  $e^-$ -Energie von 300keV

Abbildung 23: Abhängigkeit der Phasenverschiebung durch eine Phasenplatte von der Elektronenenergie und dem an die Ringelektrode angelegtem Potential

**a** – Benötigte Spannung (bzw. Potential) an einer 500nm dicken Ringelektrode in Abhängigkeit der Beschleunigungsspannung um eine Phasenverschiebung von  $\pi/2$  zu erzielen

**b** – Phasenverschiebung, welche mit 300keV beschleunigte Elektronen an einer 500nm dicken Ringelektrode erfahren, in Abhängigkeit der Spannung der Ringelektrode. Durch die roten gestrichelten Linien sind die Werte  $\pi/2$ ,  $\pi$  und  $2\pi$  hervorgehoben.



## 6 Simulationen

Im Versuchsaufbau gemäß Abbildung 12(b) ( $\rightarrow$  Seite 34) befindet sich die Doppellochblende vor der Objektivlinse. Wie in der Elektronenmikroskopie üblich wird angenommen, dass sich diese Blende näherungsweise in der vorderen Brennebene der Objektivlinse befindet, während sich die zu untersuchende Probe in der hinteren Brennebene befindet. Durch die Phasenplatte soll eine Elektronenteilwelle derart phasenverschoben werden, dass in der Probenebene ein möglichst kleiner Spot bei einem Phasenunterschied der beiden Teilwellen von  $\pi/2$  auftritt. Ein kleiner Spot ist von Interesse, da bei einem kleineren Strahldurchmesser bei EMCD-Messungen weniger Atome bzw. Atomsäulen zum Signal beitragen und somit das Signal/Rausch-Verhältnis erhöht wird.

Bei den folgenden Simulationen soll untersucht werden, ob am Ort eines Maximums der Intensitätsverteilung (Spot) in der hinteren Brennebene eine Phasendifferenz von  $\pi/2$  erreicht werden kann. Dazu soll ein möglichst breiter Satz an Parametern (Blendenradius, Blendenabstände, Phasenverschiebung) betrachtet werden. Dies soll auch durch Abbildungen visualisiert werden. Dabei werden aber zunächst Beschränkungen aufgrund der lateralen Kohärenz des Elektronenstrahls nicht berücksichtigt, d.h. die Betrachtung beschränkt sich auf einen ideal kohärenten Strahl. In dem experimentellen Abschnitt 7.1 soll aber untersucht werden, ob die laterale Kohärenz ausreicht um die aufgrund dieser Simulation erwarteten Intensitätsverteilungen zu beobachten.

### 6.1 Mathematische Beschreibung

Ausgehend von den Überlegungen in Kapitel 5.1 wird die Wellenfunktion in der hinteren Brennebene  $\psi_{BFP}$  berechnet durch die Fouriertransformation der Wellenfunktion  $\psi_{ap}$  in der Blendenebene. Diese ist gegeben durch das Produkt aus der einfallenden Wellenfunktion  $\psi_{ein}$  und der Summe der Blendentransmissionsfunktionen  $P_i$ . Somit ist die Wellenfunktion in der Blendenebene

$$\psi_{ap}(x, y) = \psi_{ein}(x, y) \cdot (P_{d_1}(x, y) + P_{d_2}(x, y)) \quad (6.1)$$

$$P_i(x, y) = \begin{cases} 1 & : (x - d_i)^2 + y^2 \leq R \\ 0 & : (x - d_i)^2 + y^2 > R \end{cases}, i = 1, 2, \quad (6.2)$$

wobei  $x$  und  $y$  Koordinaten in der Blendenebene sind.  $R$  ist der Blendenradius und  $d_1$  und  $d_2$  sind die Verschiebungen der Mittelpunkte der einzelnen Blenden weg von der optischen Achse ( $z$ -Achse) entlang der  $x$ -Achse. Anhand der Abbildung 24 und Abbildung 13 ( $\rightarrow$  Seite 35) kann man sich die Orientierung der Achsen und die Anordnung der Löcher veranschaulichen.

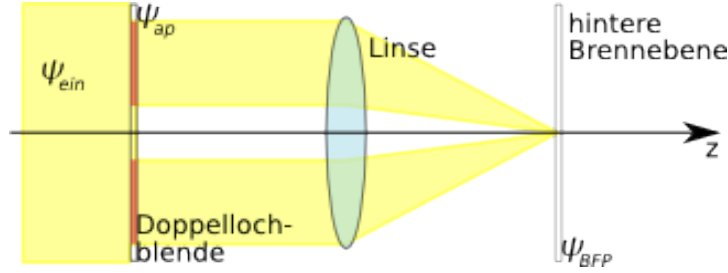


Abbildung 24: Wellenfunktionen in der Blenden- und in der hinteren Brennebene sowie die einfallende ebene Wellenfunktion.

Bei Verwendung ebener Wellen ( $\psi_{ein}(x, y) = e^{-ikz}$ ) ergibt sich die Wellenfunktion  $\psi_{BFP}$  in der hinteren Brennebene (BFP – „Back Focal Plane“) gemäß (5.9) als Fouriertransformierte der Objektwellenfunktion  $\psi_{ap}$  aus (6.1) zu

$$\begin{aligned}\psi_{BFP}(X, Y) &= \mathcal{F}(\psi_{ap}(x, y)) \\ &= \mathcal{F}\left(e^{-ikz} \cdot (P_{d1}(x, y) + P_{d2}(x, y))\right)\end{aligned}\quad (6.3)$$

Um diesen Ausdruck zu vereinfachen wendet man das Faltungstheorem (siehe dazu Anhang B.3) an. Dieses besagt, dass die Fouriertransformierte eines Produktes zweier Funktionen  $f$  und  $g$  gleich der Faltung der Fouriertransformierten von  $f$  und  $g$  ist. Somit vereinfacht sich (6.3) zu

$$\psi_{BFP}(X, Y) = \mathcal{F}\left(e^{-ikz}\right) \otimes \mathcal{F}(P_1(x, y) + P_2(x, y)) \quad (6.4)$$

Dieser Ausdruck lässt sich weiter vereinfachen, wenn man für die Fouriertransformierten einsetzt. Für die ebene Welle  $\exp(-ikz)$  ergibt die Fouriertransformation eine Delta-Funktion  $\delta(z)$ . Der zweite Term ist die Fouriertransformierte der Blendenfunktion; diese ist die sogenannte Airy-Funktion, welche mit Phasentermen multipliziert wird. Diese Terme berücksichtigen die Verschiebung der Blenden von der optischen Achse und eine zusätzliche Phase aufgrund einer Phasenplatte. Somit ergibt sich für die Wellenfunktion in der hinteren Brennebene

$$\begin{aligned}\psi_{BFP}(X, Y) &= \delta(z) \otimes \left[ \text{Airy}(X, Y) \cdot \left( e^{-\frac{id_1 X}{\lambda f}} + e^{-\frac{id_2 X}{\lambda f} - i\varphi} \right) \right] \\ &= \text{Airy}(X, Y) \cdot \left( e^{-\frac{id_1 X}{\lambda f}} + e^{-\frac{id_2 X}{\lambda f} - i\varphi} \right)\end{aligned}\quad (6.5)$$

$$\text{Airy}(X, Y) = J_1\left(\frac{2\pi R\sqrt{X^2 + Y^2}}{\lambda f}\right) \frac{\lambda f}{\pi R\sqrt{X^2 + Y^2}}, \quad (6.6)$$

wobei  $\lambda$  die Wellenlänge und  $k$  die Wellenzahl der einfallenden ebenen Welle sind.  $X$  und  $Y$  sind Koordinaten in der BFP und  $f$  ist die Brennweite der Linse.  $\varphi$  ist die durch die Phasenplatte verursachte Phasenverschiebung einer Teilwelle, welche gemäß (5.31) eingestellt werden kann. Die Airy-Funktion ist die Fouriertransformierte einer kreisrunden Blende, welche durch  $P_i(x, y)$  beschrieben wird, und  $J_1$  ist die Besselfunktion erster Ordnung.

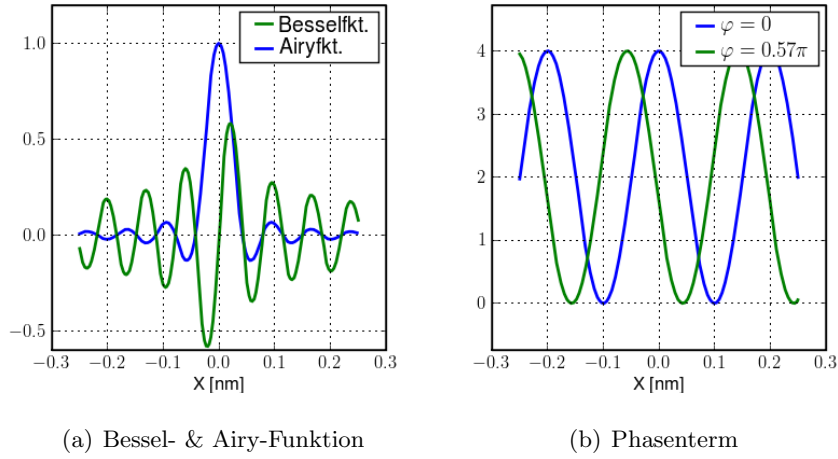


Abbildung 25: Aussehen der Faktoren in der Wellenfunktion in (6.5).  
 (a) Bessel- und Airy-Funktion. Verwendete Plotparameter:  $R = 28\mu\text{m}$ ,  $\lambda = 1.97\text{pm}$ ,  $f = 1\text{mm}$ .  
 (b) Betragsquadrat der Exponentialfunktionen. Verwendete Plotparameter:  $d_1 = -31\mu\text{m}$ ,  $d_2 = 31\mu\text{m}$ ,  $\lambda = 1.97\text{pm}$ ,  $f = 1\text{mm}$ ,  $\varphi = 0$  (blau),  $\varphi = 0.57\pi$  (grün).

Abbildung 25 zeigt zur Verdeutlichung das Verhalten der Wellenfunktion in der hinteren Brennebene, welche durch (6.5) gegeben ist. Abbildung 25(a) zeigt die Bessel- und die Airy-Funktion. Diese kommt als Faktor in (6.5) vor. Abbildung 25(b) zeigt das Betragsquadrat der Exponentialterme aus (6.5) für unterschiedliche Werte von  $\varphi$ . Anhand dieser Abbildung lässt sich das Verhalten der Wellenfunktion in Abhängigkeit der Phasenverschiebung  $\varphi$  im nächsten Abschnitt (siehe Abbildung 28 ( $\rightarrow$  Seite 59)) verstehen.

Da die Airy-Funktion rein reelle Werte besitzt, ergibt sich der Phasenunterschied zwischen den beiden Teilwellen ausschließlich aus der Verschiebung der Blenden von der optischen Achse und aus dem an der Ringelektrode angelegtem Potential  $\varphi$  zu

$$\Delta\phi(X) = \left( (d_2 - d_1) \frac{X}{\lambda f} + \varphi + \pi \right) \bmod(2\pi) - \pi, \quad (6.7)$$

wobei durch den Modulo-Operator die Phasenverschiebung auf das Intervall

$[0, 2\pi]$  reduziert wird. Die Addition und Subtraktion von  $\pi$  bewirkt die Verschiebung des Intervalls, so dass der Phasenunterschied in der Brennebene im Intervall  $[-\pi, \pi]$  angegeben werden kann.

Für die spätere Bewertung der Nutzbarkeit der Intensitätsverteilung für EMCD-Messungen werden drei Größen zur Auswertung heran gezogen:

- der Anteil der Intensitätsverteilung mit einer günstigen Phasendifferenz ( „Usable Signal“  $\mathfrak{U}$  definiert in (6.8)),
- die Halbwertsbreite („FWHM“ – „Full Width at Half Maximum“) und
- das Signal-Rausch-Verhältnis („S/N“ – „Signal/Noise“, definiert in (6.11)).

Die Halbwertsbreite wird aus der numerisch berechneten Intensitätsverteilung bestimmt (siehe Anhang: plot\_interface.py und bruteforce.py), während die Größen  $\mathfrak{U}$  und „S/N“ separat numerisch berechnet werden.  $\mathfrak{U}$  soll ein Maß dafür sein, wie viel Prozent der Intensitätsverteilung zur Erzeugung eines dichroischen Signals geeignet sind. Dazu wird das Integral über das Produkt der Intensitätsverteilung und einer Gewichtsfunktion berechnet und durch das Integral über die Intensitätsverteilung dividiert

$$\mathfrak{U} = \frac{\int_{-\infty}^{\infty} \Upsilon(X) |\psi_{BFP}(X, 0)|^2 dX}{\int_{-\infty}^{\infty} |\psi_{BFP}(X, 0)|^2 dX} \quad (6.8)$$

$$\Upsilon(X) = 2 \operatorname{sgn}(\Delta\phi(X)) \left| \left( \frac{\Delta\phi(X)}{\pi} + \frac{1}{2} \right) \bmod 1 - \frac{1}{2} \right| \quad (6.9)$$

wobei  $\Upsilon(X)$  die Gewichtsfunktion ist. Diese ist über die Phasendifferenz in der BFP gemäß (6.7) derart definiert, dass sie für die wichtigsten Werte der Phasendifferenz folgende Werte annimmt:

$$\Upsilon(X) = \begin{cases} 1 & \text{für } \Delta\phi(X) = \frac{\pi}{2} \\ 0 & \text{für } \Delta\phi(X) = i\pi, i = 0, \pm 1 \\ -1 & \text{für } \Delta\phi(X) = -\frac{\pi}{2} \end{cases} \quad (6.10)$$

Da eine Phasendifferenz von  $\pi/2$  in der BFP das Analogon zur zirkularen Polarisation eines Photons und den Idealwert darstellt, liefert  $\Upsilon$  den Wert 1 zurück, um anzudeuten, dass die Wellenfunktion an diesem Punkt  $X$  zu 100% zum EMCD-Signal beiträgt. Für eine Differenz von  $-\pi/2$  ergibt sich ein gegenteiliger Effekt, bei dem das EMCD-Signal ein anderes Vorzeichen trägt, was durch den Funktionswert -1 berücksichtigt wird. Dies entspricht in der Analogie zum Photon einem entgegengesetzten Drehsinn

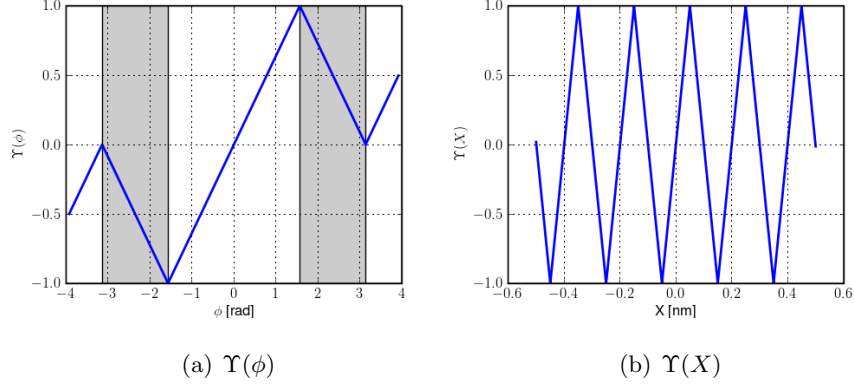


Abbildung 26: Verhalten der Gewichtsfunktion  $\Upsilon$  in Abhängigkeit von  $\phi$  und  $X$ :  
 (a)  $\Upsilon(\phi)$  ist durch die Punkte  $\phi = 0, \pm\pi/2, \pm\pi$  gemäß (6.10) festgelegt und verläuft dazwischen linear.  
 (b)  $\Upsilon(X)$  ist gemäß (6.9) definiert, und entspricht  $\Upsilon(\phi)$ , wobei für  $\phi$  gemäß (6.7)  $\phi(X)$  eingesetzt wird mit folgenden Parametern:  $R = 28\mu m$ ,  $d_1 = 31\mu m$ ,  $d_2 = -31\mu m$ ,  $\lambda = 1.97\mu m$ ,  $f = 1mm$ ,  $\varphi = 0$

der zirkularen Polarisation. Bei einer Differenz von  $i\pi$  ( $i = 0, \pm 1$ ) ist die Analogie zur linearen Polarisation gegeben, so dass kein dichroisches Signal beobachtbar ist, was durch den Rückgabewert 0 angedeutet wird. Zwischen diesen Punkten wird ein linearer Verlauf der Funktion angenommen, da dies durch den Ausdruck (6.9) am einfachsten in einer numerischen Simulation zu realisieren ist.

Die Gewichtsfunktion  $\Upsilon(\phi)$  ist in Abbildung 26 dargestellt in Abhängigkeit von  $\phi$ . Der Graph verläuft zwischen den Extremalwerten  $\pm 1$  gemäß (6.9) und (6.10). Durch die grau hinterlegten Bereiche in Abbildung 26(a) sind die Intervalle  $[-\pi, -\pi/2]$ ,  $[-\pi/2, \pi/2]$ ,  $[\pi/2, \pi]$  hervorgehoben. Ebenfalls in Abbildung 26 ist die Gewichtsfunktion  $\Upsilon(X)$  für einen realistischen Satz von Parametern zu sehen. Dafür wurde für die Phasendifferenz  $\phi$  gemäß (6.7) eingesetzt, so dass die Gewichtsfunktion nach der Wahl der Parameter ( $R = 28\mu m$ ,  $d_1 = 31\mu m$ ,  $d_2 = -31\mu m$ ,  $\lambda = 1.97\mu m$ ,  $f = 1mm$ ,  $\varphi = 0$ ) eine Funktion der Koordinate  $X$  in der hinteren Brennebene ist. Dies ist zu sehen in Abbildung 26(b).

Die letzte für die Auswertung relevante Größe ist das Signal/Rausch-Verhältnis  $S/N$ . Dieses ist das Verhältnis zwischen dem Flächeninhalt innerhalb der Halbwertsbreite (FWHM) und dem Flächeninhalt unter der ge-

samen Intensitätsverteilung

$$S/N = \frac{\int_a^b |\psi_{BFP}(X, 0)|^2 dX}{\int_{-\infty}^{\infty} |\psi_{BFP}(X, 0)|^2 dX}, [a, b] : \text{FWHM} \quad (6.11)$$

## 6.2 Wellenfunktion in der hinteren Brennebene

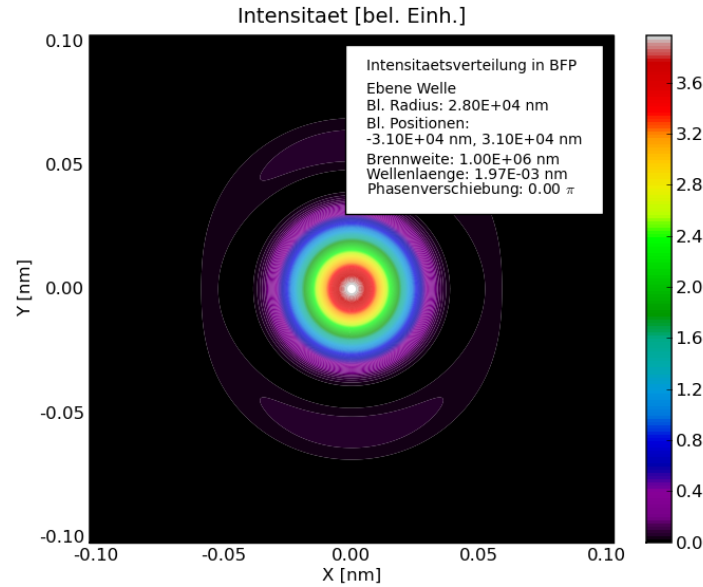
Abbildung 27 zeigt als gefüllte Contourlinienplots die Intensitätsverteilungen ebener Wellen in der hinteren Brennebene der Linse, in welcher sich die Probe befindet. Diese wurden gemäß (6.5) berechnet für Blendenradien von  $28\mu\text{m}$ , Blendenverschiebungen von  $-31\mu\text{m}$  bzw.  $+31\mu\text{m}$  entlang der  $x$ -Achse, Brennweite von  $1\text{mm}$  und einer Elektronenwellenlänge von  $1.97\text{pm}$  (entspricht einer Beschleunigungsspannung von ca.  $300\text{kV}$ ).

Ohne Phasenschiebung ist die Rotationssymmetrie der Airy-Funktion in Abbildung 27(a) aufgrund der Superposition der beiden Teilwellen gebrochen. Aber es ist noch ein vollständiger Ring des Maximums 1. Ordnung zu erkennen. Bei einer durch die Phasenplatte verursachten Phasenschiebung von  $0.57\pi$  ist in Abbildung 27(b) das Fehlen der Rotationssymmetrie deutlicher zu erkennen. Der Ring des 1. Maximums ist nur noch in der  $X < 0$  - Halbebene präsent und die Intensität des zentralen Maximums (Maximum 0. Ordnung) ist stark reduziert. Des weiteren ist die Position des zentralen Maximums gegenüber dem zentralen Maximum im Falle einer deaktivierten Phasenplatte etwas in Richtung der negativen  $X$ -Achse verschoben.

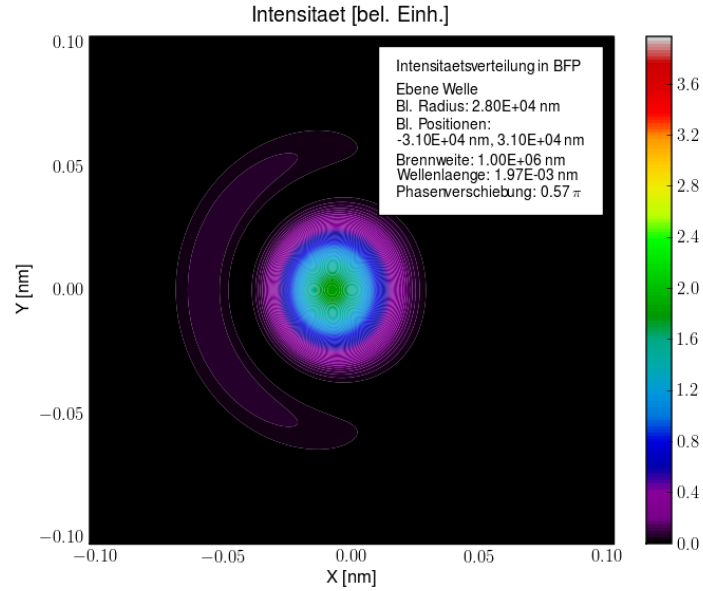
Zur besseren Analyse der Daten beschränkt sich die weitere Betrachtung der Daten auf einen Linescan entlang der  $Y = 0$  - Achse. Dazu zeigt Abbildung 28 die Intensitätsverteilung in der hinteren Brennebene (BFP) in einem Linescan entlang dieser Achse. Gezeigt sind die Fälle mit Phasenschiebungen von  $\varphi = 0$  (grüne Kurve) und  $\varphi = 0.57\pi$  (rote Kurve). Durch eine graue Hinterlegung wird der Teil der resultierenden Intensitätsverteilung hervor gehoben, der mit einer Toleranz von 5% die gewünschte Phasendifferenz von  $\pi/2$  aufweist. Dabei ist die Airy-Funktion (siehe (6.5)) normiert auf  $Airy(0, 0) = 1$ . Des weiteren ist der Phasenunterschied (6.7) für beide Fälle in derselben Abbildung separat aufgetragen.

Das Maximum der roten Intensitätsverteilung ist um einen Faktor von  $\approx 2$  kleiner im Vergleich zur grünen Kurve. Es ist deutlich zu sehen, dass das Maximum der roten Kurve leicht verschoben ist und an diesem Ort die Phasendifferenz in der Beugungsebene den gewünschten, optimalen Wert von  $\pi/2$  annimmt.

Die Abbildungen 27 und 28 wurden mit dem Programm `plot_interface.py` (Quelltext und Beschreibung im Anhang D) erzeugt.



(a) Intensitätsverteilung in BFP ohne Phasenplatte



(b) Intensitätsverteilung mit Phasenplatte

Abbildung 27: Betragsquadrat der Wellenfunktion in der Proben- bzw. hinteren Brennebene (BFP) nach Gleichung (6.5) für ebene Wellen. Blendenradien und -abstände zur optischen (z-) Achse betragen  $28 \mu\text{m}$  bzw.  $31 \mu\text{m}$ . Brennweite und Wellenlänge betragen  $1 \text{mm}$  bzw.  $1.97 \text{pm}$ . Abbildungen erzeugt mit `plot_interface.py`

(a) Phasenschiebung aufgrund Phasenplatte: 0  
 (b) Phasenschiebung aufgrund Phasenplatte:  $0.57 \pi$

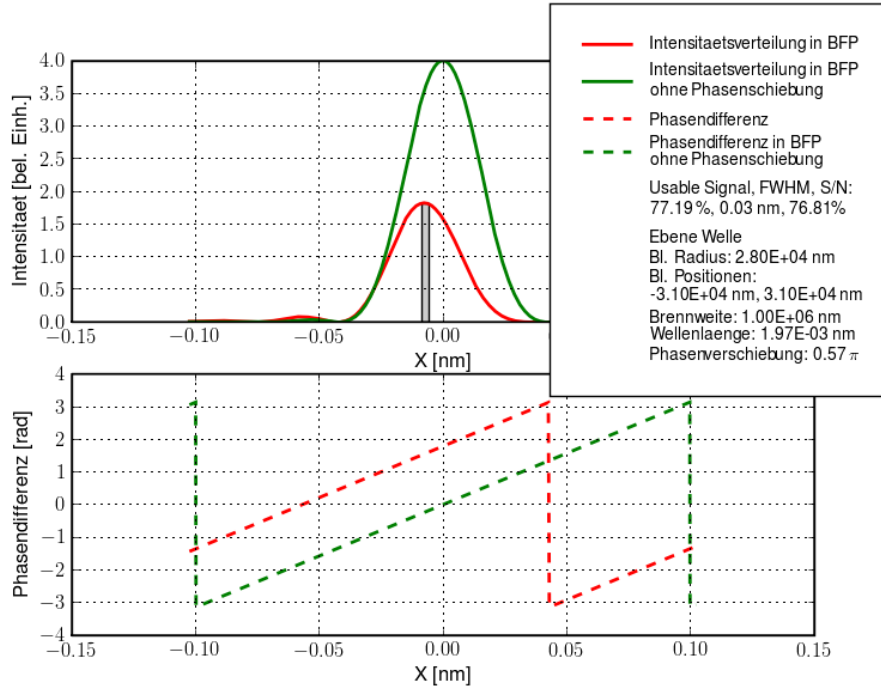


Abbildung 28: Linescan in der BFP entlang der  $Y = 0$  -Achse. Die **grüne und rote durchgezogene Kurve** beschreiben die Intensitätsverteilungen bei deaktivierter bzw. aktivierter Phasenplatte. Durch das grau hinterlegte Polygon wird der Teil der Intensitätsverteilung hervorgehoben, welcher eine Phasenverschiebung von  $\pi/2$  mit einer Toleranz von  $\pm 5\%$  aufweist. Die dazugehörigen Phasendifferenzen sind darunter als **grün bzw. rot gestrichelte Kurven** gezeigt.

### 6.3 Wellenfunktion außerhalb der hinteren Brennebene

Nachdem im voran gegangenen Abschnitt die Intensitätsverteilung bzw. die Wellenfunktion in der hinteren Brennebene (BFP) betrachtet wurde (Abbildungen 27 und 28), wird nun ein kurzer Blick auf die Wellenfunktion in beliebigen Ebenen geworfen.

Dazu wird zuerst der Nullpunkt der  $Z$ -Achse (optische Achse) in die BFP gelegt. Zur Verdeutlichung der Geometrie (Achsen und Ebenen) siehe Abbildung 12(b) ( $\rightarrow$  Seite 34) und Abbildung 16 ( $\rightarrow$  Seite 39). Ausgehend von der Wellenfunktion in der BFP (6.5) lässt sich durch Verwendung des Propagators (5.2) die Wellenfunktion in jeder Ebene parallel zur BFP numerisch berechnen.

Abbildung 29 zeigt als gefüllten Contourlinienplot die Intensitätsverteilungen entlang der  $X$ -Achse bei  $Y = 0$  während der Propagation entlang der  $Z$ -Achse. Bei  $Z = 0$  befindet sich definitionsgemäß die hintere Brennebene (vergleiche Abbildung 28). Für unterschiedliche Propagationsstrecken



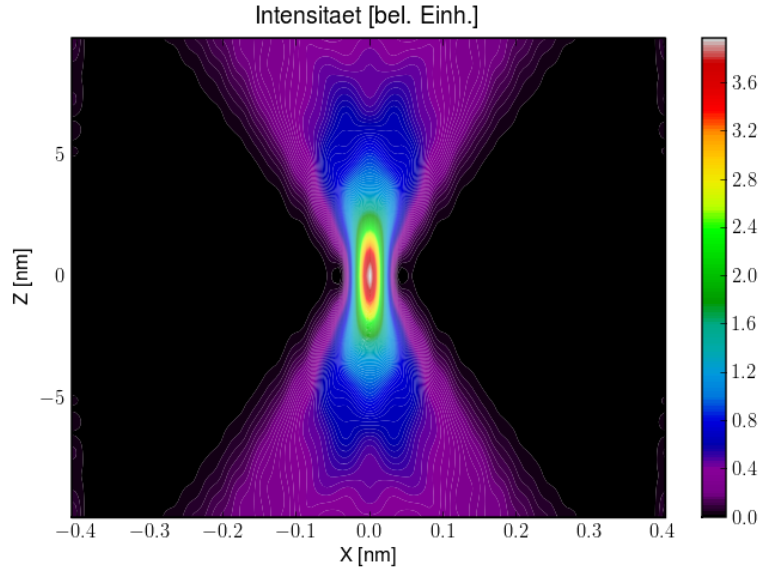


Abbildung 29: Querschnitt in Falschfarben durch die Intensitätsverteilung entlang der X-Achse (ähnlich zu Abbildung 28) für unterschiedliche Propagationsstrecken entlang der Z-Achse. Es wurden dieselben Parameter wie in den Abbildungen 27(a) und 28 verwendet:  $\lambda = 1.97\text{nm}$ ,  $R = 28\mu\text{m}$ ,  $d_{1,2} = \pm 31\mu\text{m}$ ,  $\varphi = 0$ ,  $f = 1\text{mm}$

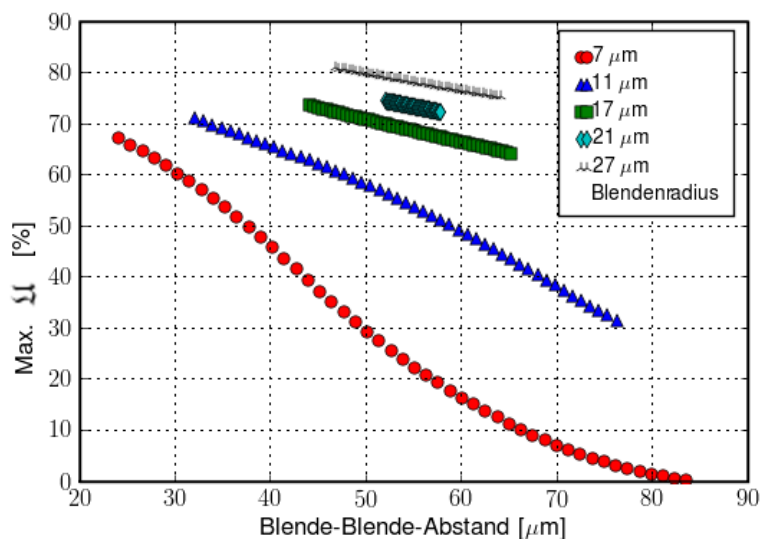
entlang der Z-Achse sind die Intensitätsverteilungen entlang der X-Achse (analog zu Abbildung 28) aufgetragen. Im Bereich der hinteren Brennebene ( $Z = 0$ ) bleibt für geringe Propagationsstrecken (ca. 1 - 1.5nm) das Profil der Intensitätsverteilung nahezu unverändert.

#### 6.4 Variation der Parameter

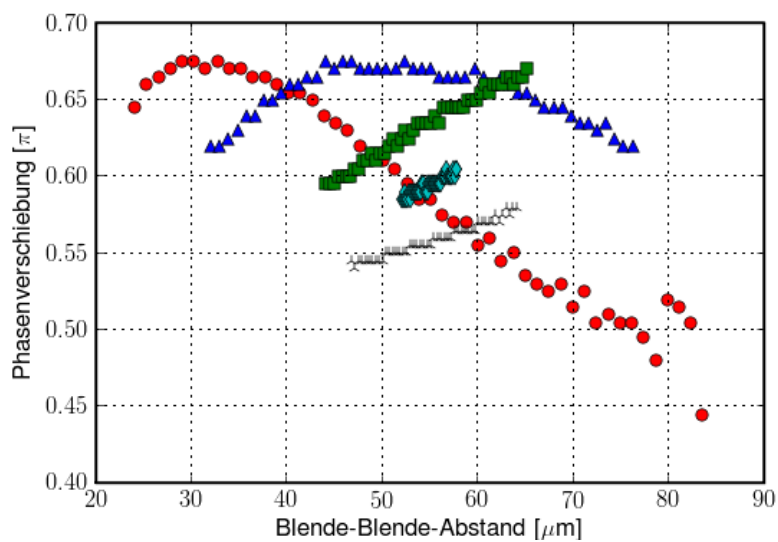
Um den optimalen Parametersatz zu finden, werden für alle Parameterkombinationen die in (6.8) und (6.11) definierten Größen und die Halbwertsbreite berechnet und in einer *MySQL*-Datenbank gespeichert (Programm: brute-force.py, Quelltext im Anhang D). Dabei werden die Wertebereiche für die Größen Blendenabstände und Blendenradius bereits so gewählt, dass limitierende Herstellungsfaktoren (Quadratische Fläche mit  $100\mu\text{m}$  Kantenlänge, Abstand zwischen Blenden in der Größenordnung einiger  $\mu\text{m}$ ) berücksichtigt werden. Auf diese wird in Kapitel 7.2 näher eingegangen.

Anhand von graphischen Auswertungen dieser numerisch gewonnenen Datensätze wird abschließend ein passender Parametersatz ausgewählt (Programm: sql1.py, Code im Anhang D enthalten).

Die graphische Ausgabe des Auswertungsprogramms sql1.py ist in den

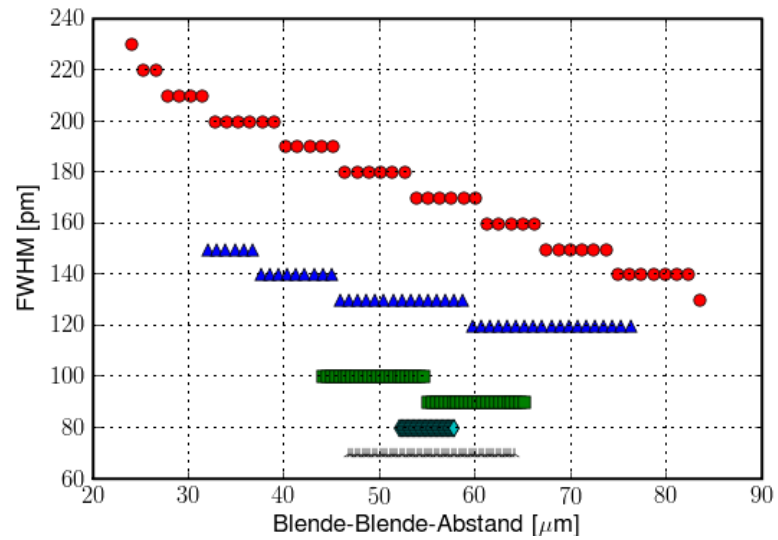


(a)  $U$

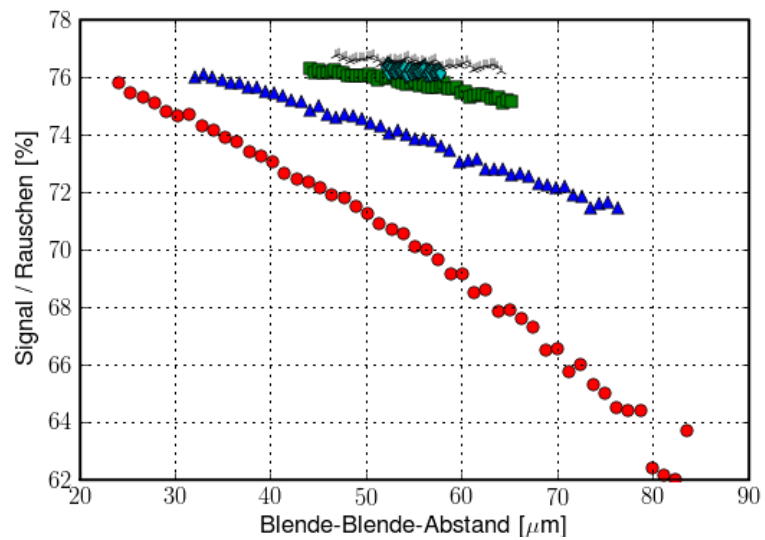


(b) Phasenschiebung

Abbildung 30: Ergebnis des Programms sql1.py: „ $U$ “ und Phasenschiebung sind aufgetragen gegen den Blende-Blende-Abstand. Brennweite: 1mm, Wellenlänge: 4pm



(a) Halbwertsbreite



(b) Signal/Rausch - Verhältnis

Abbildung 31: Ergebnis des Programms sql1.py: Halbwertsbreite und Signal/Rausch-Verhältnis sind aufgetragen gegen den Blende-Blende-Abstand. Brennweite: 1mm, Wellenlänge: 4pm. Legende: siehe Abbildung 30(a)

Abbildung 30 und 31 zu sehen für die Blendenradien  $7\mu\text{m}$  (rote Kreise),  $11\mu\text{m}$  (blaue Dreiecke),  $17\mu\text{m}$  (grüne Quadrate),  $21\mu\text{m}$  (cyan-farbige Rauten) und  $27\mu\text{m}$  (schwarze dreizackige Sterne). Dabei wurden die Parameter Brennweite ( $f = 1\text{mm}$ ) und Wellenlänge ( $\lambda = 4\text{pm}$ ) konstant gehalten.

Abbildung 30(a) zeigt den maximal erzielbaren Wert für die in (6.8) definierte Größe „ $\mathfrak{U}$ “ in Abhängigkeit vom Abstand der beiden Einzelblenden, der im folgenden als Blende-Blende-Abstand bezeichnet werden soll. Da jeder Wert für den Blende-Blende-Abstand durch mehrere Kombinationen von  $d_1$  und  $d_2$ , den Abständen der Einzelblenden zur optischen Achse, realisiert werden kann, gibt es möglicherweise mehrere Werte für  $\mathfrak{U}$  zu einem Blende-Blende-Abstand. Daher wird aus allen Datensätzen, die diesen Blende-Blende-Abstand haben, der maximal erreichbare all dieser Werte zur Auftragung gewählt. Für eine detailliertere Beschreibung dieser Idee anhand des Programmquelltextes siehe Anhang D.1.3.

Die Größe  $\mathfrak{U}$  zeigt für die Blendenradien 17, 21 und  $27\mu\text{m}$  einen linearen Abfall bei steigendem Blende-Blende-Abstand. Bei den kleinen Blendenradien ( $7$  und  $11\mu\text{m}$ ) verläuft die Kurve nicht linear und nimmt stark ab. Insbesondere für den  $7\mu\text{m}$  Radius folgen die Punkte dem Verlauf der rechten Flanke einer Gaußkurve.

Für den Bau einer Doppellochblende empfehlen sich daher zwei möglichst große Blendenlöcher mit minimalen Abstand.

In Abbildung 30(b) ist die Phasenschiebung gegen den Blende-Blende-Abstand aufgetragen. Dabei handelt es sich um die durch die Phasenplatte verursachte Phasenschiebung, mit welcher die Maxima in Diagramm 30(a) erhalten werden.

Die Werte für die benötigte Phasenverschiebung bewegen sich im Intervall  $[0.45\pi, 0.7\pi]$ . Die dafür benötigten Potentiale (im Bereich  $0.2 - 1.2\text{V}$ ) können an einer Phasenplatte problemlos angelegt werden; für einige Parameterkombinationen aus Beschleunigungsspannung und Phasenplattendicke sind die anzulegenden Potentiale, welche benötigt werden um eine Phasenverschiebung von  $\pi/2$  zu erhalten, in Tabelle 3 ( $\rightarrow$  Seite 49) aufgeführt. Ein Elektron mit einer kinetischen Energie von  $300\text{keV}$  erfährt z.B. an einer Ringelektrode mit  $300\text{nm}$  Dicke eine Phasenverschiebung von  $\pi/2$  für ein angelegtes Potential von  $0.8\text{V}$ .

Diagramm 31(a) zeigt die Halbwertsbreite (FWHM), welche zu besagtem Maximalwert von  $\mathfrak{U}$  gehört, ebenfalls gegen den Blende-Blende-Abstand aufgetragen. Die diskrete Abstufung der FWHM ist zu erklären durch die Bestimmung der FWHM innerhalb der numerischen Rechnung und spiegelt nicht eine experimentell zu erwartende Quantisierung wider.

Die FWHM nimmt mit wachsendem Blende-Blende-Abstand ab. Diese Abnahme folgt bei den Blendenradien  $7$  und  $11\mu\text{m}$  einem exponentiellen

Abfall, wie man ihn z.B. vom radioaktiven Zerfall kennt. Diese Abhängigkeit nimmt in dem hier betrachteten Rahmen mit wachsendem Blendenradius ab, bis für einen Radius von  $21\ \mu\text{m}$  und größer im Rahmen der Genauigkeit der numerischen Berechnung eine Unabhängigkeit der FWHM vom Abstand erreicht ist.

Diese Abhängigkeit der FWHM vom Blende-Blende-Abstand bestätigt die Vermutung, dass der Radius der zu verwendenden Blende maximal sein sollte. Durch diese „Minimierung“ der FWHM erhält man ein zentrales Maximum der Airy-Funktion mit minimaler Breite, was für die Durchführung eines EMCD-Experimentes besonders wünschenswert ist: Wird der beleuchtete Bereich der Probe kleiner, so tragen weniger Partikel, über welche gemittelt wird, zum EMCD-Signal bei. Ebenfalls lässt sich die Ortsauflösung im STEM-Betrieb steigern aufgrund der kleineren FWHM.

In Diagramm 31(b) wird das in (6.11) definierte Signal-Rausch-Verhältnis gegen den Blende-Blende-Abstand aufgetragen.

Das Signal-Rausch-Verhältnis (S/N) scheint linear mit wachsendem Blende-Blende-Abstand abzufallen. Dies ist vor allem für die kleinen Blendenradien ( $7\ \mu\text{m}$ ,  $11\ \mu\text{m}$ ) gut zu sehen. Nimmt der Radius zu, so wird die Steigung des Abfalls geringer.

Dieses Verhalten des S/N bestätigt die Eignung eines kleinen Blende-Blende-Abstands. Somit steht der Wahl eines Maximalwerts für die Blendenradien und eines Minimalwerts für den Blende-Blende-Abstand nichts im Weg.

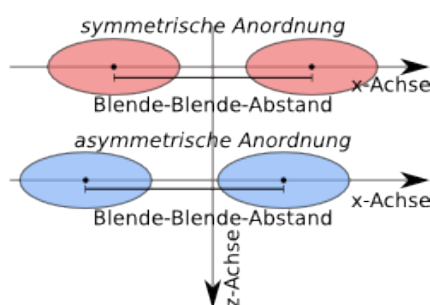


Abbildung 32: Symmetrische Anordnung der Blendenöffnungen (rot, oben): die Mittelpunkte der Lochblenden haben denselben Abstand zur optischen (z-) Achse. Werden die beiden Blenden entlang der x-Achse verschoben, so dass der *Blende-Blende-Abstand* konstant bleibt, wird die Anordnung als asymmetrisch (blau, unten) bezeichnet.

Anhand der Abbildungen 33 und 34 soll überprüft werden, ob die ausgewerteten Größen von einer Asymmetrie in der Blendenanordnung beeinflusst werden. Als symmetrisch soll die Anordnung der zwei Blenden bezeichnet

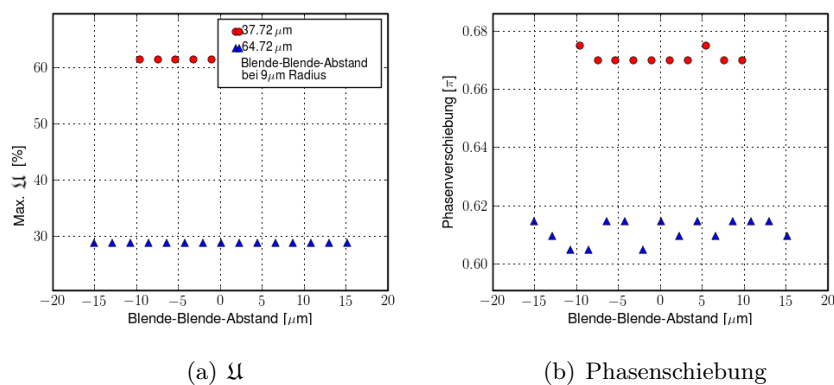


Abbildung 33: Ergebnis des Programms `sql2.py`:  $\mathfrak{U}$  und Phasenschiebung sind aufgetragen gegen die Verschiebung der symmetrisch angeordneten Doppellochblende entlang der  $X$ -Achse. Radius 9̑m, Brennweite: 1mm, Wellenlänge: 4pm

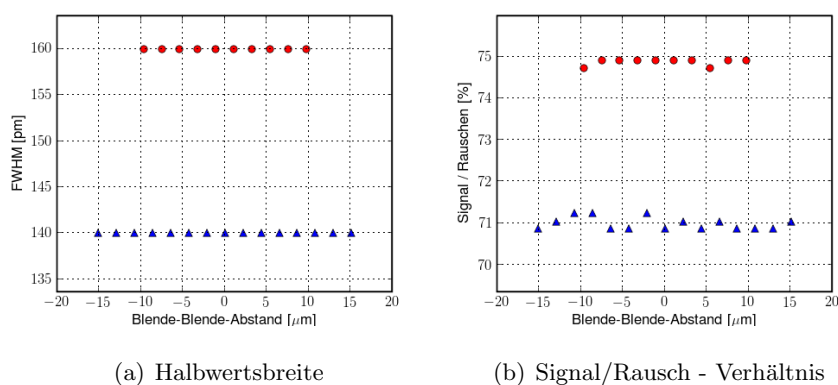


Abbildung 34: Ergebnis des Programms `sql2.py`:  $\mathfrak{U}$  und Phasenschiebung sind aufgetragen gegen die Verschiebung der symmetrisch angeordneten Doppellochblende entlang der  $X$ -Achse. Radius 9̑m, Brennweite: 1mm, Wellenlänge: 4pm. Legende: siehe Abbildung 33(a)

werden, wenn die Blenden entlang der  $X$ -Achse von der optischen Achse verschoben sind und zur optischen Achse denselben Abstand haben. Als asymmetrisch soll die Anordnung bezeichnet werden, wenn eine symmetrische Anordnung entlang der  $X$ -Achse verschoben wird und dabei der Abstand zwischen den beiden Blenden unverändert bleibt. Dies kann man sich anhand von Abbildung 32 verdeutlichen.

In Abhängigkeit dieser Verschiebung sind die Größen  $\mathfrak{U}$  (Abbildung 33(a)), Phasenverschiebung durch Phasenplatte (Abbildung 33(b)), FWHM (Abbildung 34(a)) und S/N (Abbildung 34(b)) aufgetragen für einen Blendenradius von  $9\mu\text{m}$  und Blendenabstände von  $37.72\mu\text{m}$  (rote Punkte) und  $64.72\mu\text{m}$  (blaue Dreiecke).

Die Größe  $\mathfrak{U}$  und die Halbwertsbreite weisen keinerlei Abhängigkeit gegenüber einer Verschiebung der symmetrischen Blendenanordnung auf.

Die Phasenverschiebung und das Signal-Rausch-Verhältnis weisen leichte Schwankungen für unterschiedliche Verschiebungen der Blenden aus der symmetrischen Anordnung auf. Die Phasenschiebung schwankt dabei um maximal  $0.01\pi$ . Das Signal/Rausch-Verhältnis weist Variationen von 1% (Absolutwert) auf.

Folglich wird bei einer asymmetrischen Anordnung keine Einwirkung auf die Nutzbarkeit der Doppellochblende für EMCD-Messungen beobachtet. Da eine asymmetrische Anordnung keinen vorteilhaften Einfluss auf die Auswertungsgrößen hat, kann in der Praxis eine symmetrische Anordnung gewählt werden. Da während der Justierung eines Elektronenmikroskops der Elektronenstrahl immer entlang der optischen Achse ausgerichtet werden soll ist eine symmetrische Anordnung der Blenden ebenfalls aus diesem Grund eine gute Wahl.

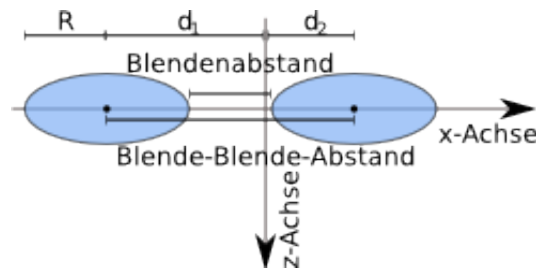


Abbildung 35: Zur Verdeutlichung des Unterschiedes zwischen den Größen *Blendenabstand* und *Blende-Blende-Abstand*, welche in Abbildung 36 bzw. in den Abbildungen 30 und 31 Verwendung finden. Der *Blende-Blende-Abstand* ist die Summe der Abstände der Einzelblenden zur optischen Achse ( $d_1$  und  $d_2$ ). Der *Blendenabstand* ist der *Blende-Blende-Abstand* subtrahiert mit dem doppelten Blendenradius  $R$ .

In einer alternativen Darstellung zeigt Abbildung 36 die Abhängigkeit von  $\mathfrak{U}$ , der Phasenschiebung, der Halbwertsbreite und des Signal/Rausch-Verhältnisses vom *Blendenabstand*. Dabei soll der *Blendenabstand* die Dicke des Stegs zwischen den beiden Blenden sein

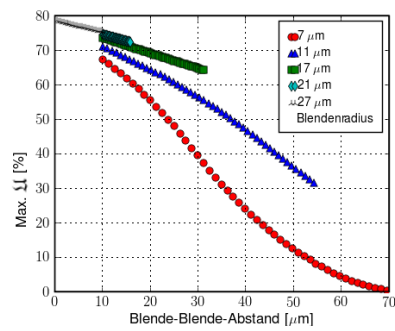
$$\text{Blendenabstand} = \overbrace{d_1 + d_2}^{\text{Blende-Blende-Abstand}} - 2 \cdot R. \quad (6.12)$$

$d_1$  und  $d_2$  sind die Abstände der Mittelpunkte der beiden Einzelblenden von der optischen Achse.  $R$  ist der Radius der Blenden. Dazu zeigt Abbildung 35 anhand einer asymmetrischen Blendenanordnung die geometrischen Verhältnisse ( $d_1$ ,  $d_2$  und  $R$ ) sowie die Unterschiede zwischen den beiden definierten Größen *Blendenabstand* und *Blende-Blende-Abstand*.

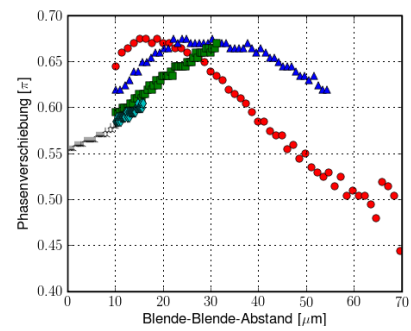
In Abbildung 36 sind die Kurven für die selben fünf Radien wie in den Abbildungen 30 und 31 aufgetragen. Es ist zu erkennen, dass der Maximalwert für  $\mathfrak{U}$  und S/N gegen eine Art Grenzwert streben für *Blendenabstand*  $\rightarrow 0$  und Radius  $\rightarrow \infty$ .

Die Kurven zeigen bei der Auftragung gegen den *Blendenabstand* dieselben Verläufe wie bei der Auftragung gegen den *Blende-Blende-Abstand*. Sie sind lediglich um  $2R$  auf der horizontalen Achse verschoben.

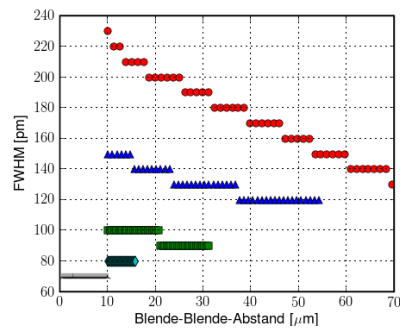




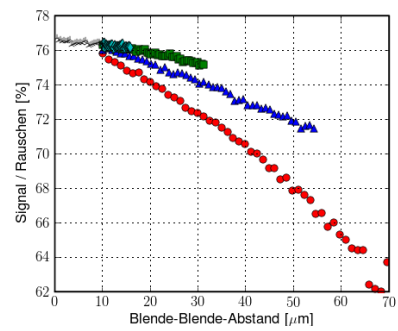
(a)



(b)



(c)



(d)

Abbildung 36: Alternative Darstellung der Diagramme aus den Abbildungen 30 und 31. Es wurden dieselben Parameter (Brennweite: 1mm, Wellenlänge: 4pm) verwendet.

## 7 Experimente

Nach den theoretischen Betrachtungen in Kapitel 6 befasst sich dieses Kapitel mit der Frage der experimentellen Verifikation der berechneten Intensitätsverteilungen und Realisierbarkeit eines STEM-Betriebes mit einer Doppellochblende.

Zuerst wird anhand eines Linien- und Lochgitters abgeschätzt, ob die lateralen Kohärenzlängen der zur Verfügung stehenden Elektronenmikroskope ausreichen um in der Beugungsebene die erwarteten Interferenzerscheinungen beobachten zu können. In der Arbeitsgruppe von Prof. Dr. Josef Zweck an der Universität Regensburg stehen ein Philips CM30 sowie ein FEI Tecnai F30 Transmissionselektronenmikroskop zur Verfügung. Danach wird eine Doppellochblende auf möglichst einfache Weise hergestellt und als Kondensorblende in ein Mikroskop eingebaut. Mit einer solchen Anordnung sollen Beobachtungen im STEM-Betrieb gemacht werden.

### 7.1 Test der Kohärenzlänge

Während für die Simulationen in Kapitel 6 ein kohärenter Elektronenstrahl angenommen wurde, muss in der Praxis natürlich die reale laterale Kohärenzlänge und damit ein inkohärenter Hintergrund berücksichtigt werden, da inkohärente Teilwellen das Interferenzsignal deutlich reduzieren.

Zu diesem Zweck wird ein Kohlenstoff-Abdruck eines optischen Liniengitters mit 1200 Linien pro mm in einem Philips CM30 betrachtet.

Abbildung 37 zeigt die berechnete Fouriertransformierte einer Aufnahme des Liniengitters, welches mit einer CCD-Kamera im Philips CM30 aufgenommen wurde. Dies ist die erwartete Beugungsfigur im Falle einer ideal kohärenten Elektronenwelle.

Zu sehen ist eine Reihe von punktförmigen Maxima, welche von links unten nach rechts oben verlaufen. In einem spitzen Winkel dazu verlaufen einige dunkle Linien vor einem flächigen grauen Fleck, welcher aufgrund des amorphen Kohlenstoffs zu sehen ist.

Das tatsächlich beobachtete Beugungsbild (siehe Abbildung 38) zeigt ein intensives zentrales, punktförmiges Maximum, welches mit einer dunklen Ellipse überlagert ist. Die beobachtete Beugungsfigur scheint somit eine „verschmierte“ Version der erwarteten Figur zu sein. Durch eine in das Mikroskop einführbare Spitze wird ein Großteil des Maximums verdeckt um Beschädigungen aufgrund von Überbelichtung an der CCD-Kamera zu vermeiden. Somit ist im CM30 keine ausreichende Kohärenz gegeben, um die Details des erwarteten Beugungsbildes (siehe Abbildung 37) zu erkennen. Dieser Sachverhalt war auch zu erwarten, da im CM30 eine LaB<sub>6</sub>-Kathode und im F30 eine Feldemissionskathode zum Einsatz kommt.

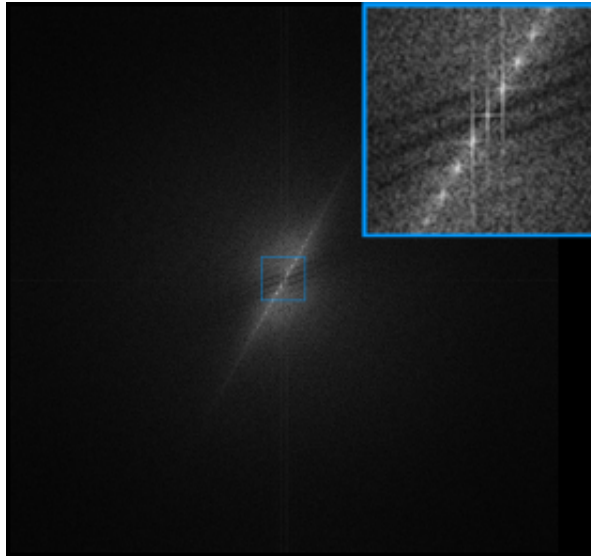


Abbildung 37: Berechnete Fouriertransformierte des beleuchteten Liniengitters und Vergrößerung. Dies ist die erwartete Beugungsfigur für ideal kohärente Elektronen. Das Beugungsbild besteht aus einer Reihe von punktförmigen Maxima vor einem grauen Hintergrund. In einem spitzen Winkel zu dieser Reihe verlaufen einige dunkle Linien vor diesem Hintergrund.

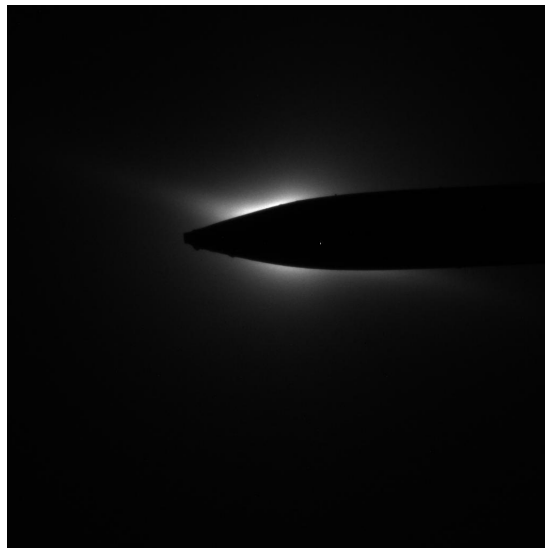


Abbildung 38: Beobachtete Beugungsfigur eines Liniengitters aufgenommen am Philips CM30 mit einer CCD-Kamera. Ein intensives zentrales, punktförmiges Maximum ist mit einer dunkleren Ellipse überlagert. Die beobachtete Beugungsfigur ist somit eine „verschmierte“ Version der erwarteten Figur.

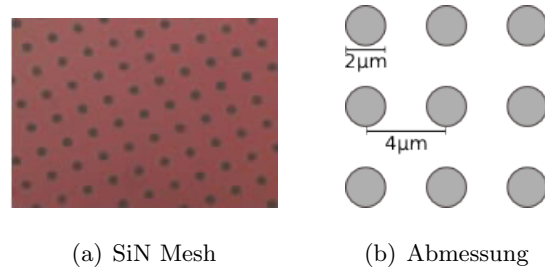


Abbildung 39: DuraSiN Lochgitter.

(a) Aufnahme eines DuraSiN Meshes. Quelle: [SiN]

(b) Abmessung des verwendeten DuraSiN Meshes DTM-25233

Zur Beobachtung einer Beugungsfigur am Tecnai F30 wird ein SiN-Mesh der Firma DuraSiN verwendet. Bei dem verwendeten SiN-Mesh (Bezeichnung bei DuraSiN: DTM-25233) handelt es sich um einen quadratischen, starren Siliziumrahmen (Kantenlänge: 2.65mm, Dicke:  $300\mu\text{m}$ ), welcher eine dünne SiN-Schicht (Kantenlänge:  $500\mu\text{m}$ , Dicke: 200nm) stützend trägt. In dieser Schicht befinden sich regelmäßig angeordnete Löcher im Abstand von  $4\mu\text{m}$  mit Durchmesser  $2\mu\text{m}$  [SiN] (siehe Abbildung 39). Auf dieses SiN-Mesh wird eine Goldschicht (Dicke ca.  $1\mu\text{m}$ ) durch Sputtern in einer Balzer MED10 aufgebracht. Das so erhaltene Lochgitter wird als Probe eingeschleust und im Beugungsmodus betrachtet.

Am Tecnai F30 werden mehrere beleuchtete Löcher zur Abbildung verwendet. In der Beugungsebene ist das Betragsquadrat der Fouriertransformierten der die Löcher passierenden Wellenfunktion beobachtbar. In Abbildung 40(a) ist die Intensität der Beugungsfigur logarithmisch aufgetragen. Des weiteren zeigt Abbildung 40(b) einen Linescan durch diese Beugungsfigur entlang des türkis-umrandeten Balkens, wobei über die Breite des Balkens gemittelt wird.

Es ist zu erkennen, dass die Intensitätsverteilung starke Ähnlichkeit zu einer Airy-Funktion aufweist, welche mit einem inkohärenten Hintergrund überlagert ist. Ein zentrales, intensives Maximum ist umgeben von ringförmigen Nebenmaxima, deren Intensität mit dem Abstand von diesem zentralen Maximum schnell abnimmt. Dies lässt sich im Linescan erkennen anhand des gaußförmigen inkohärenten Untergrundes, aus welchem eine Vielzahl von Spitzen heraus ragen.

Daraus ist ersichtlich, dass am CM30 aufgrund der beschränkten Kohärenz der LaB<sub>6</sub>-Kathode keine EMCD-Messung möglich sind, da für die kohärente Beleuchtung der im folgenden Kapitel 7.2 hergestellten Doppellochblende eine laterale Kohärenzlänge von ca.  $140\mu\text{m}$  (Länge der Diagonalen der Membran; siehe Abbildung 13 (→ Seite 35)) erforderlich ist. Somit sind am

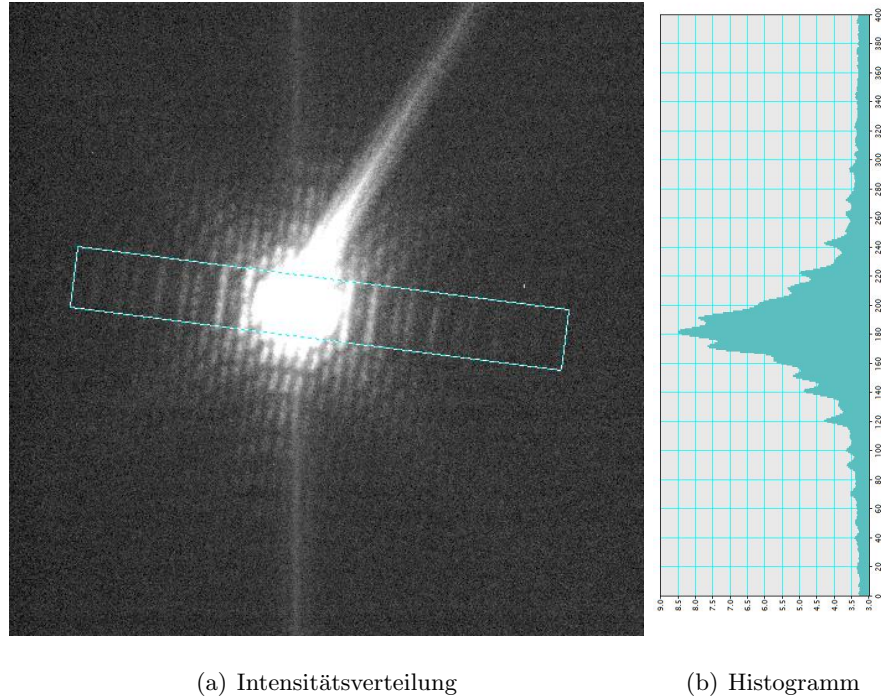


Abbildung 40: Beugungsfigur am F30.

(a) Intensitätsverteilung in der Beugungsebene verursacht durch mehrere Lochblenden mit Abstand  $4\mu\text{m}$  und Durchmesser  $2\mu\text{m}$ . Die Intensität wurde logarithmisch in Grauwerte umgerechnet.

(b) Histogramm entlang des türkis umrandeten Balkens in (a). Über die Breite des Balkens wird gemittelt.

CM30 keine EMCD-Messungen, aber zumindest Betrachtungen im STEM-Betrieb (siehe Kapitel 7.3) möglich. Deshalb werden die ersten Versuche auch am CM30 durchgeführt werden. Spätere EMCD-Messungen werden am F30 durchgeführt werden müssen.

## 7.2 Herstellung

In den vorangegangenen Kapiteln wurde bereits mehrmals Bezug genommen auf eine Doppellochblende, wobei eines der Löcher mit einer Phasenplatte ausgestattet sein soll. In diesem Unterkapitel wird nun diese Doppellochblende hergestellt und vorgeschlagen, wie diese mit einer Phasenplatte ausgerüstet werden kann.

### 7.2.1 Doppellochblende

Die Doppellochblende wird hergestellt aus einer  $\text{Si}_3\text{N}_4$ -Membran, deren Kantenlänge  $100\mu\text{m}$  beträgt. Dazu werden durch Elektronenlithographie und Trockenätzen zwei Löcher in der Membran erzeugt. Um diese Membran als Kondensorblende in einem TEM (im Rahmen dieser Arbeit: Philips CM30) zu verwenden, wird auf diese eine Goldschicht der Dicke  $200\text{nm}$  aufgedampft. Somit befinden sich zwei Blendenlöcher in einer für Elektronen nicht transparenten Goldschicht.

Aufgrund der Ergebnisse der Auswertung der Simulation (vergleiche Abbildung 30 ( $\rightarrow$  Seite 61)) ist ein maximaler Blendenradius und minimaler Blendenabstand wünschenswert. Um auf der begrenzten zur Verfügung stehenden Fläche ( $100\mu\text{m} \times 100\mu\text{m}$ ) den Radius zu maximieren werden die Löcher – wie in Abbildung 13 ( $\rightarrow$  Seite 35) gezeigt – auf der Diagonalen der Membran platziert. Um die Stabilität der Membran zwischen den Löchern zu bewahren und um Platz für die Ringelektroden der Phasenplatte zu reservieren wurde für den „Steg“ zwischen den Löchern eine Größe von einigen  $\mu\text{m}$  angenommen.

Unter diesen numerischen Bedingungen wurden zwei Sätze von Doppellochblenden mit unterschiedlichen Radien hergestellt.

Beim ersten Satz beträgt der Blendenradius  $28\mu\text{m}$ , so dass sich der Abstand zwischen den beiden Blenden zu ca.  $6.2\mu\text{m}$  ergibt. Eine solche Doppellochblende ist schematisch dargestellt in Abbildung 13.

Der zweite Blendensatz wurde hergestellt mit einem gewählten Blendenradius von  $28.5\mu\text{m}$ . Dies reduziert den „Steg“ zwischen den Löchern auf eine Breite von  $3.7\mu\text{m}$ . Die Anordnung entspricht wieder der Geometrie von Abbildung 13.

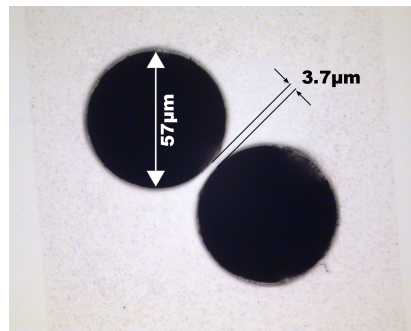


Abbildung 41: Doppellochblende im Lichtmikroskop. Die beiden geätzten Löchern haben einen Radius von  $28.5\mu\text{m}$ . Die Breite des Stegs zwischen den beiden Löchern beträgt ca.  $3.7\mu\text{m}$ .

Eine lichtmikroskopische Aufnahme einer solchen Doppellochblende mit

Radius  $28.5\mu\text{m}$  ist zu sehen in Abbildung 41. Die beiden Löcher (schwarz dargestellt) weisen einen scharfen Rand auf und der Steg dazwischen ist intakt. Im weiteren Verlauf der Arbeit wird deshalb diese Doppellochblende verwendet.

Vor dem Einbau der Doppellochblende mit dem Kondensorblendenhalter in das Elektronenmikroskop, wird diese auf einen Aluminiumring geklebt und der Kleber ausgehärtet. Bei dem Kleber handelt es sich um einen M-Bond-Kleber. Der Aluminiumring wird in einer Führungsschiene des Blendenhalters aufgenommen.

### 7.2.2 Phasenplatte

Um die Implementation einer Phasenplatte auf der Doppellochblende zu realisieren zeigen sich zwei Methoden in der Elektronenlithographie in Anlehnung an das Herstellungsverfahren in [MOT98] als geeignet:

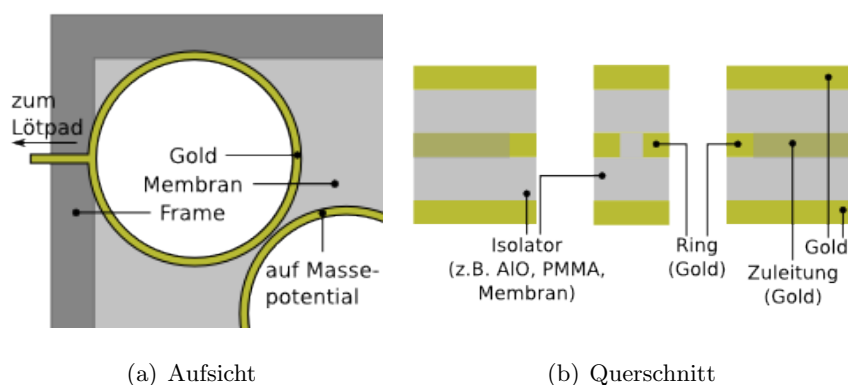


Abbildung 42: Schematische Darstellung der Ringelektroden auf der Membran.  
**(a)** Ringe und Zuleitungen, welche durch Elektronenlithographie erzeugt werden sollen.  
**(b)** Querschnitt durch diese Phasenplatte. Die Gold-Strukturen sind in derselben Farbe wie in (a) dargestellt (solid: Ring, halbtransparent: Zuleitung). Eine Ringelektrode wird über ein Kontaktpad (hier nicht gezeigt) mit einer Spannungsquelle verbunden. Die zweite Ringelektrode wird gemeinsam mit den abschließenden Goldschichten auf Masse gelegt. Zwischen den Goldelementen wird ein isolierendes Material benötigt.

Die erste Möglichkeit ist die elektronenlithographische Aufbringung zweier Ringe und Zuleitungen aus leitendem Material an der nicht vergoldeten Seite der Membran. Diese Ringe sollen die Löcher umschließen. Während der erste Ring auf ein Potential  $U$  gelegt wird, ist der andere Ring auf Masse zu legen. Auf diese Schicht aus Goldstrukturen werden eine isolierende Schicht (z.B. AlO) und eine abschließende Goldschicht aufgebracht. Abbil-

Abbildung 42 zeigt eine schematische Darstellung der ersten Möglichkeit einer Phasenplatte.

Durch den auf Potential  $U$  liegenden Ring erleiden Elektronen, welche durch diesen fliegen, eine Phasenschiebung (vgl. Kapitel 5.2). Allerdings sollen die Elektronen, welche durch den anderen Ring gehen, keine oder eine möglichst geringe Phasenschiebung erleiden. Deshalb liegt dieser Ring auf Masse-Potential um das Potential des ersten Rings abzuschirmen. Die beiden abschließenden Goldschichten liegen ebenfalls auf Masse.

Bei Vergleich der Phasenplatte in Abbildung 17 mit der vorgeschlagenen, vereinfachten Phasenplatte fällt auf, dass die Abschirmung des Potentials nicht durch eine geschlossene Zylinderwand, sondern lediglich durch den auf Masse liegenden Ring erfüllt wird. Daher stellt sich die Frage, ob dieses Potential im auf Masse liegenden Ring verschwindet. Die Abwesenheit des Potentials ist wichtig, da einer der beiden Strahlen keine Phasenschiebung erleiden soll.

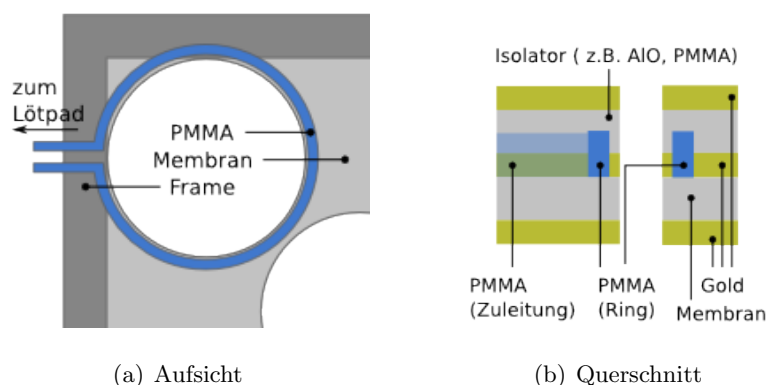


Abbildung 43: Schematische Darstellung der PMMA-Strukturen, welche als Negativlack auf die SiN-Membran (mit Löchern) und den Träger geschrieben werden.

(a) PMMA-Struktur, welche nach der Lithographie und dem Ätzen stehen bleiben soll

(b) Querschnitt durch die geplante Phasenplatte. Die PMMA-Strukturen aus (a) sind wieder blau dargestellt (solid: Ring, halbtransparent: Zuleitung). Durch diese wird eine Goldschicht unterbrochen, so dass eine Ringelektrode, eine Leitung und ein Kontaktpad (hier nicht gezeigt) entstehen. Darauf wird ein isolierendes Material aufgebracht. Abgeschlossen wird die Phasenplatte durch zwei Goldschichten „oben“ und „unten“.

Die zweite Möglichkeit ist die Aufbringung eines nicht geschlossenen Ringes und eines nicht geschlossenen Quaders aus PMMA auf der nicht vergoldeten Seite. Der Ring und der Quader werden durch zwei Geraden (Begrenzung der späteren Zuleitung) verbunden. Anschließend wird eine Goldschicht aufgedampft, so dass ein leitender Ring mit Zuleitung und ein Kontaktpad erzeugt werden. Dabei dient das PMMA als Isolator und bei



der Lithographie als Negativlack [HWB97]. Zur Verdeutlichung zeigt Abbildung 43 schematisch die Strukturen aus PMMA, die auf die Membran bzw. den Träger „geschrieben“ werden. Während der so erhaltene Ring auf ein definiertes Potential  $U$  gelegt wird, ist die restliche Goldschicht in dieser Ebene mit Masse verbunden. Anschließend wird eine Schicht aus isolierendem Material – z.B. AlO, PMMA – aufgebracht. Abgeschlossen wird die Phasenplatte durch Goldschichten. Diese werden später auf Masse-Potential gelegt. Dadurch soll ebenfalls eine Phasenschiebung innerhalb des Rings und eine Abschirmung des Potentials außerhalb des Ringes realisiert werden.

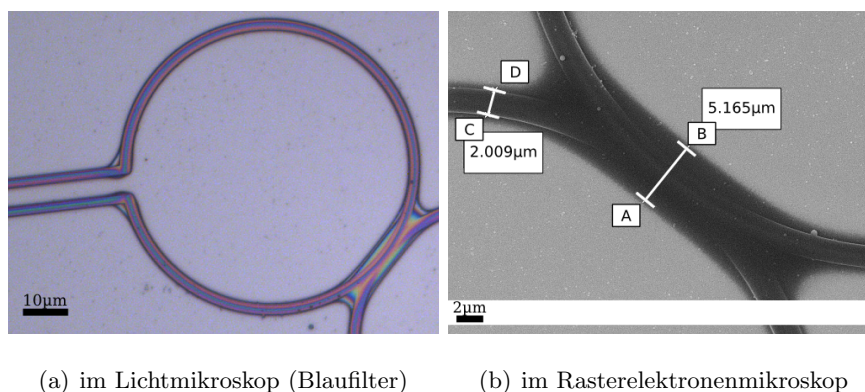


Abbildung 44: Aufnahmen von Ring- und Zuleitungsstrukturen aus PMMA als Negativlack vor dem Bedampfen mit Gold.

(a) PMMA-Ring und zwei PMMA-Geraden im Lichtmikroskop

(b): Aufnahme derselben Stelle im Rasterelektronenmikroskop. Dicke des Rings (C → D)  $2.009\mu\text{m}$ , Dicke der Überlappung (A → B):  $5.165\mu\text{m}$ , Beschleunigungsspannung =  $10\text{kV}$

Für die ersten Versuche zum Bau einer Phasenplatte wurden die Strukturen der zweiten Möglichkeit mit einem zusätzlichen Ring geschrieben. Die Resultate dieser Versuche zur Aufbringung dieser Strukturen sind gezeigt in Abbildung 44. Abbildung 44(a) zeigt zwei nach dem Belichten stehen gebliebene überlappende Ringe und Begrenzungen der Zuleitung aus PMMA. Abbildung 44(b) zeigt eine Vergrößerung der Überlappung der beiden Ringe im Rasterelektronenmikroskop. In dieser Aufnahme wurde die Dicke der Ringe zu  $2\mu\text{m}$  und die Dicke der Überlappung zu  $5.2\mu\text{m}$  bestimmt.

Anhand dieser Versuche ist zu sehen, dass der Verwendung von PMMA als Negativlack nichts im Wege steht. Es ist in weiteren Schritten zu untersuchen, ob PMMA im Betrieb einer Phasenplatte als Isolationsmaterial geeignet ist und wie hoch die geschriebenen PMMA-Strukturen sind.

### 7.3 STEM-Betrieb mit Doppellochblende

Vor einer jeden Messung an einem Elektronenmikroskop – in diesem Fall ein Philips CM30 – ist es empfehlenswert eine Justierung des Mikroskops (Alignment) durchzuführen.

Durch das Alignment führen schrittweise Anleitungen am Monitor des Mikroskops. Durchzuführen sind die Prozeduren „Gun Alignment“ und „Column Alignment“. Die letztere Prozedur unterteilt sich weiter in die Subprozeduren „TEM Bright Field“, „Nano Probe“ und „Scanning“.

Für die Prozedur des „Gun Alignment“ wird die Probe aus dem Strahlengang entfernt. Durch Verkippen und anschließendes Verschieben der Kathode wird der Elektronenstrahl zuerst parallel zur optischen Achse ausgerichtet und dann so verschoben, dass er möglichst gut mit der optischen Achse zur Deckung kommt.

Die Prozedur „Column Alignment“ umfasst die Justierungsschritte für die Betriebsmodi „TEM Bright Field“, „Nano Probe“ und „Scanning“. Mit diesen Schritten sollen folgende Ziele erreicht werden:

- Beugungsmuster sind in der Ebene der Objektivblende fokussiert
- der Elektronenstrahl wird bei Verschiebung auf der Probe nicht verkippt und umgekehrt
- der Strahl verläuft entlang der optischen Achse der Objektivlinse
- (Beugungs-) Bilder bleiben bei Änderung der Kameralänge/Vergrößerung zentriert
- Beugungsbilder und Bilder sind aufeinander abgestimmt, so dass beim Umschalten vom Bild- in den Beugungsmodus und umgekehrt die selbe Fläche auf der Probe betrachtet wird

Um Beschädigungen an der dünnen Goldschicht (Dicke: 200nm) der Doppellochblende, die durch den sehr intensiven Elektronenstrahl verursacht werden könnten, soweit wie möglich auszuschließen, wird zuerst das Alignment mit einer normalen Kondensorblende durchgeführt und anschließend wird erst auf die Doppellochblende umgestellt und diese auf der optischen Achse zentriert.

Für diesen „ersten Test“ des STEM-Betriebes mit einer Doppellochblende ist zunächst keine Phasendifferenz zwischen den beiden Teilstrahlen notwendig. Daher wird vorerst eine einfache Doppellochblende ohne Phasenplatte (siehe Abbildung 41) verwendet.

Nachdem das Mikroskop justiert wurde und nun betriebsbereit ist, kann in den STEM-Modus gewechselt werden. In diesem soll eine kontrastreiche

Probe betrachtet werden. Dazu eignet sich eine Kohlenstoffmembran auf welcher einzelne oder kleine Gruppen von mikroskopisch kleinen Goldpartikeln aufgebracht sind.

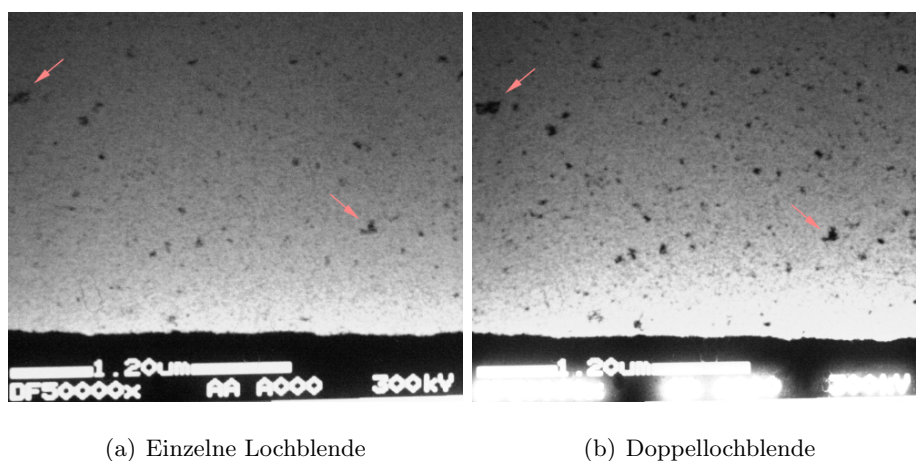


Abbildung 45: STEM-Hellfeldaufnahmen von Goldpartikeln bei 50,000-facher Vergrößerung. Diese wurden mit einer digitalen Spiegelreflexkamera vom Monitor des CM30 aufgenommen.

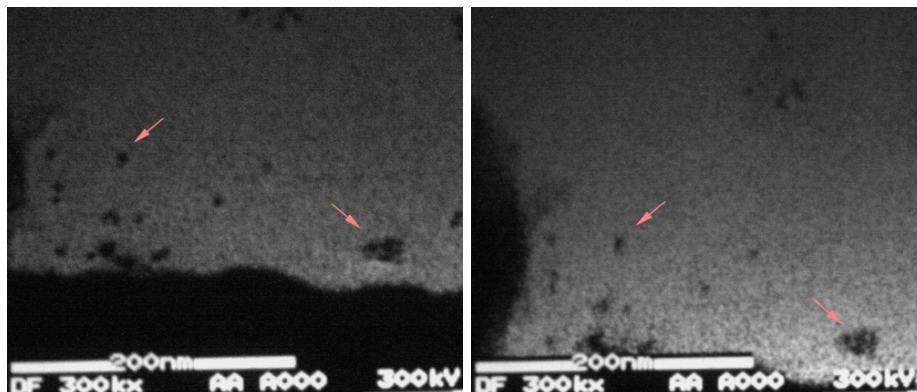
Die Figuren (a) und (b) zeigen denselben Bereich der Probe für eine Standard-einzel-lochblende (a) und die selbst hergestellte Doppellochblende (b). Am unteren Rand ist jeweils der stützende Rahmen der Probe zu sehen.

Goldpartikel und Gruppen von Goldpartikeln können auf beiden Aufnahmen identifiziert werden.

Die Abbildungen 45 und 46 zeigen vier Hellfeldaufnahmen im STEM-Betrieb. Der Hellfelddetektor wurde dazu mit einem der beiden hellen Spots, welche von den Blendenlöchern her rühren, beleuchtet. Mittels einer Digitalkamera (Canon EOS 350D) wurden die Aufnahmen vom Monitor des Mikroskops aufgenommen.

Abbildungen 45(a) und 45(b) zeigen denselben Ausschnitt der Probe bei einer Vergrößerung von 50,000 und Spotgröße „5“ bei Verwendung einer einzelnen Blende (a) und bei Verwendung der Doppellochblende (b). Auf beiden Aufnahmen ist am unteren Rand der Stützrahmen der Probe zu sehen und darüber eine Vielzahl von Goldpartikeln. Auch bei näherer Betrachtung der beiden Aufnahmen ist durch die Verwendung der Doppellochblende keine Beeinträchtigung der Bildqualität zu erkennen. Durch rote Pfeile in den Aufnahmen werden einige identische Partikel hervorgehoben.

Die Aufnahmen für 300,000-fache Vergrößerung und Spotgröße „8“ im STEM-Modus sind in den Abbildungen 46(a) und 46(b) zu sehen. Die beiden Abbildungen zeigen denselben Bereich der Probe, wobei am unteren Rand jeweils der Stützrahmen der Probe zu erkennen ist. Auf beiden Aufnahmen ist ebenfalls ein großes dunkles Objekt am linken Rand zu sehen. In beiden



(a) Einzelne Lochblende

(b) Doppel Lochblende

Abbildung 46: STEM-Hellfeldaufnahmen von Goldpartikeln bei 300,000-facher Vergrößerung. Diese wurden mit einer digitalen Spiegelreflexkamera vom Monitor des CM30 aufgenommen.

Die Figuren (a) und (b) zeigen denselben Bereich der Probe für eine Standardeinzel Lochblende (a) und die selbst hergestellte Doppel Lochblende (b). Am unteren Rand ist jeweils der stützende Rahmen der Probe zu sehen.

Goldpartikel und Gruppen von Goldpartikeln können auf beiden Aufnahmen identifiziert werden.

Abbildungen sind zwei große Gruppen von Goldpartikeln (rechts unten; links unten) sowie eine Reihe von einzelnen Goldpartikeln zu erkennen. Durch rote Pfeile werden einige Partikel, welche in beiden Abbildungen zu sehen sind, markiert.

Beim Vergleich der beiden Abbildungen 46(a) und 46(b) lässt sich aufgrund der Verwendung der Doppel Lochblende keine Beeinträchtigung der Bildqualität feststellen.

Für die Belichtung des Sensors der Digitalkamera steht leider nur ein kurzes Zeitintervall zur Verfügung, welches der Lesedauer für eine Zeile entspricht. Dies liegt daran, dass der Monitor zwei phosphorisierende Schichten hat: die erste leuchtet intensiv blau für eine kurze Zeit und die zweite leuchtet mit geringerer Intensität gelb/grün und vergleichsweise lang. Zur Aufnahme mit der Digitalkamera eignet sich nur das Signal, welches durch die gelb/-grün leuchtende Schicht erzeugt wird, da diese Schicht derart leuchtet um den ganzen Monitor so auszufüllen, dass auf dem Bild der Digitalkamera „kontrastreiche“ Strukturen erkennbar sind. Die blau leuchtende Schicht erzeugt einen über den Monitor laufenden extrem intensiven Balken. Dieser bewirkt eine starke Überbelichtung des Sensors. So sieht man in der Aufnahme der Digitalkamera nur eine bläulich weiße Fläche, wenn man eine kleine Blende und eine Belichtungszeit, welche der Zeit zum Darstellen des

gesamten Bildes am Monitor entspricht, verwendet. In Abbildung 45(b) sind die Auswirkungen einer teilweisen Belichtung des Kamerasensors durch die blaue Phosphoreszenzschicht am unteren Rand zu erkennen: Die am unteren Bildschirmrand ausgegebenen Informationen (Längenangabe, Vergrößerung, Beschleunigungsspannung) sind stark überbelichtet und kaum noch zu erkennen.

Diese ersten Ergebnisse im STEM-Betrieb mit einer Doppellochblende zeigen wie erwartet keine Auswirkung auf die Bildqualität. Um im Weiteren die Möglichkeit von EMCD-Messungen im STEM-Betrieb zu untersuchen muss der Detektor anstatt auf einen „Bragg Strahl“ auf eine der beiden Positionen **A** oder **B** (siehe Abbildung 12(b)) gesetzt werden. Danach wäre zu überlegen, an Stelle des Detektors ein Spektrometer (EELS) zu verwenden.

## 8 Ausblick

Im Rahmen dieser Arbeit wurde die Doppellochblende in Form einer einfachen Blende (siehe Abbildung 47(a)) gebaut und verwendet. Für zukünftige EMCD-Messungen im STEM-Betrieb ist zu überlegen, ob die Ausführung der Doppellochblende als Hutblende (siehe Abbildung 47(b)) vorteilhaft ist, da eine solche Hutblende aufgrund ihrer Bauform stark divergierende Elektronen blockiert und üblicherweise auch Hutblenden Verwendung finden. Insbesondere ließen sich dadurch mechanische Belastungen auf die Membran verringern. Abbildung 47 zeigt schematisch eine Hutblende sowie eine einfache Blende. In beiden Fällen ist durch einen roten Quader ein Bereich markiert, in dem sich die beiden Blendenlöcher und die Phasenplatte befinden sollen.

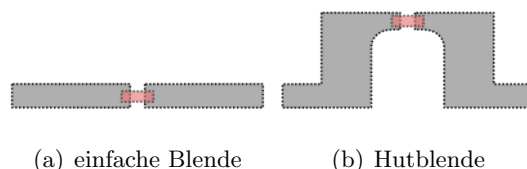


Abbildung 47: Querschnitt durch eine schematische einfachen Blende (a) und eine schematische Hutblende (b)

Des weiteren wurden bereits die ersten Schritte zum Bau einer Phasenplatte auf der vorgestellten Doppellochblende gemacht.

Im Anschluss an diese Arbeit ist zu erforschen, ob PMMA als Isolationsmaterial bei Dicken in der Größenordnung  $2\mu\text{m}$  ausreichend ist unter den gegebenen Umständen (Potential an der Ringelektrode in der Größenordnung 1V und schnelle Elektronen der Energie 300keV). Des weiteren muss eine Phasenplatte, wie sie in Kapitel 5.2 vorgestellt wurde, vollständig in die Doppellochblende integriert und mit Verkabelung im Elektronenmikroskop installiert werden.

Wie in Kapitel 5.2 bereits Erwähnung fand, wurde vor Kurzem auf dem „14<sup>th</sup> European Microscopy Congress“ (2008) eine Phasenplatte beruhend auf dem Aharonov-Bohm-Effekt vorgeschlagen [EB08]. Die Vorzüge einer solchen Phasenplatte wären, dass man keine elektrische Installation am Mikroskop anbringen muss und dass keine Spannungsschwankungen der Spannungsquelle befürchtet werden müssen. Daher erscheint es sinnvoll, die Entwicklungen auf diesem Gebiet im Auge zu behalten.

Bevor EMCD-Messungen mit dem vorgeschlagenen Aufbau durchgeführt werden können, wäre ein Testlauf (Beobachtung der Interferenzfiguren) erforderlich.

Nachdem in dieser Arbeit die ersten STEM-Aufnahmen mit einer Doppellochblende (siehe Abbildungen 45 und 46) präsentiert wurden, wird der nächste Schritt sein, STEM-Bilder an den Positionen **A** und **B** in der Beugungsebene (vergleiche Abbildung 12(b)) aufzunehmen. Vor dem letzten Schritt zu einem ChiralSTEM-Betrieb sollen EMCD-Spektren mit der intrinsischen Methode am F30 zu Vergleichszwecken aufgenommen werden.

Anschließend muss eine solche Doppellochblende mit Phasenplatte in ein Mikroskop mit höherer lateralen Kohärenzlänge ( $> 140\mu\text{m}$ ) – hier in Regensburg ein FEI Tecnai F30 – eingebaut werden. Der letzte Schritt in Richtung zu dem angestrebten „ChiralSTEM“-Betrieb, wäre die Ersetzung des Detektors durch ein Spektrometer (EELS) um somit während eines STEM-Betriebs EMCD-Signale aufzunehmen.

# Anhang

## A STEM - Detektoren

Im Rastertransmissionselektronenmikroskop finden drei Arten von Detektoren, welche in Abbildung 48 schematisch dargestellt sind, Verwendung. Diese werden im Folgenden kurz beschrieben:

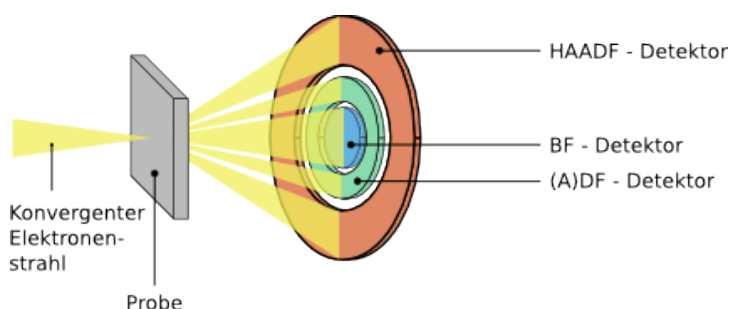


Abbildung 48: Schematische Darstellung der Anordnung der unterschiedlichen Detektoren: Hellfeld- (blau), Dunkelfeld- (grün) und Großwinkeldunkelfeld-Detektor (rot). Nachbildung einer Grafik aus [Kru]

**BF-Detektor** Der kreisförmige Hellfeld-Detektor (BF – „bright field“) befindet sich hinter der Probe und misst die Intensität des direkten Strahls von einem Punkt in der Probe. In diesem Fall resultiert das Bild aus einer Abschwächung des direkten Strahls aufgrund der Wechselwirkung mit der Probe. Daher erscheinen dicke Bereiche der Probe, Bereiche mit schweren Atomen und kristalline Bereiche mit dunklem Kontrast.

**(A)DF-Detektor** Der ringförmige Dunkelfeld-Detektor (ADF – „annular dark field“) kann in seinem Loch den BF-Detektor aufnehmen und formt sein Bild aus gestreuten Elektronen. Der beobachtbare Kontrast resultiert hauptsächlich aus an Kristalliten gestreuten Elektronen und ist mit einem Hintergrund aus inkohärenter Rutherford-Streuung überlagert.

Das DF-Bild enthält wichtige Informationen über die Probe [Kru], z.B. Flächendefekte, Stapelfehler oder Partikelgröße.

**HAADF-Detektor** Der ringförmige Detektor (HAADF – „high angle annular dark field“) hat einen größeren Innendurchmesser als der ADF-Detektor, so dass er diesen umschließen kann. Somit registriert er Elektronen, die in einen sehr großen Winkel gestreut werden. Größtenteils



trägt inkohärente Rutherford-Streuung zum Bild bei. Dadurch wird ein Kontrast in Abhängigkeit der Ladungszahl (Z-Kontrast) aufgrund der Coulomb-Wechselwirkung erreicht.

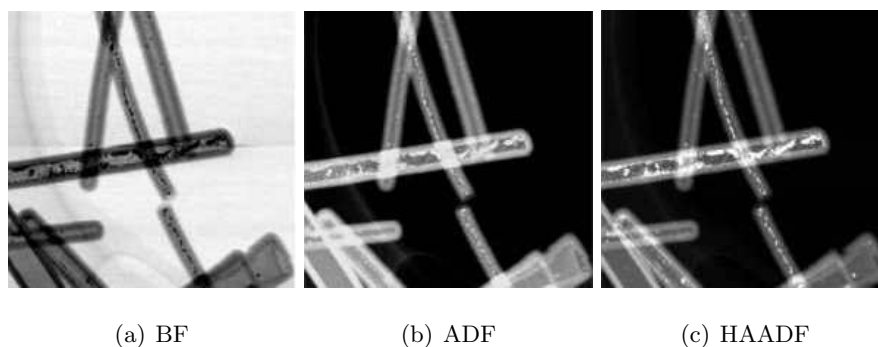


Abbildung 49: STEM-Aufnahmen von SiO<sub>2</sub>-Nanoröhrchen mit Platinfüllung.  
Quelle: [Kru]

Ein Beispiel zur Verdeutlichung der Unterschiede zwischen Hellfeld- und Dunkelfeldbild zeigt Abbildung 49. Dabei handelt es sich um STEM-Aufnahmen von SiO<sub>2</sub>-Nanoröhrchen, welche mit Platin gefüllt sind. Im Hellfeldbild (a) erscheinen die Platin-Partikel mit einem dunklen Kontrast, da sie kristallin und die schwersten Streuzentren im betrachteten System sind. Im Dunkelfeldbild (b) erscheinen die Platin-Teilchen mit einem hellen Kontrast aufgrund der Beugung und inkohärenten Streuung. Im großwinkligen Dunkelfeldbild (c) tragen nur inkohärent gestreute Elektronen zum Bild bei.

## B Fouriertransformation und Faltung

### B.1 Fouriertransformation

Die Fouriertransformation ist eine Integraltransformation, die einer gegebenen Funktion eine andere Funktion – ihre Fourier-Transformierte – zuordnet. In vielen Einsatzgebieten wird sie dazu verwendet, um für zeitliche Signale (z. B. ein Sprachsignal oder einen Spannungsverlauf) das Frequenzspektrum zu berechnen (Fourier-Analyse).

Die Fouriertransformation ist in vielen Wissenschafts- und Technikzweigen von außerordentlicher praktischer Bedeutung. Sie findet Anwendung in der Akustik, Optik, vielen Teilgebieten der Mathematik, Signalverarbeitung, Kryptographie und vielen mehr. Je nach Anwendungsbereich erfährt die Zerlegung vielerlei Interpretationen. In der Akustik ist sie beispielsweise die Frequenz-Transformation des Schalls in Oberschwingungen.

Die Fourier-Transformation wurde von dem französischen Mathematiker Jean Baptiste Joseph Fourier 1822 in seiner „Théorie analytique de la chaleur“ entwickelt. Die Fouriertransformation und -rücktransformation sind folgendermaßen definiert:

$$F(k) = \int_{-\infty}^{\infty} f(x) \cdot e^{-ikx} dx \quad (\text{B.1})$$

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(k) \cdot e^{ikx} dk \quad (\text{B.2})$$

Eigenschaften der Fouriertransformation  $\mathcal{F}$ :

$$\mathcal{F}(af(x) + bg(x)) = aF(k) + bG(k) \quad (\text{B.3})$$

$$\mathcal{F}(f(x)) = F(k) \quad (\text{B.4})$$

$$\mathcal{F}(f(-x)) = F(-k) \quad (\text{B.5})$$

$$\mathcal{F}(f^*(x)) = F^*(-k) \quad (\text{B.6})$$

$$\Rightarrow F(-k) = F^*(k) \quad (\text{B.7})$$

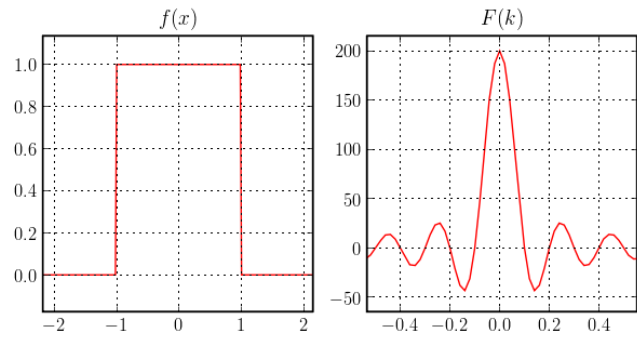
$$\mathcal{F}(f^*(-x)) = F^*(k) \quad (\text{B.8})$$

$$\mathcal{F}f(x \mp \zeta) = e^{\mp ik\zeta} F(k) \quad (\text{B.9})$$

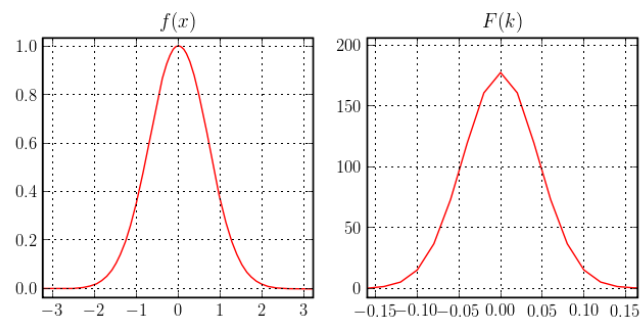
$$\mathcal{F}(F(k \mp \kappa)) = e^{\pm i\kappa x} f(x) \quad (\text{B.10})$$

$$\mathcal{F}\left(\frac{d}{dx}f(x)\right) = ikF(k) \quad (\text{B.11})$$

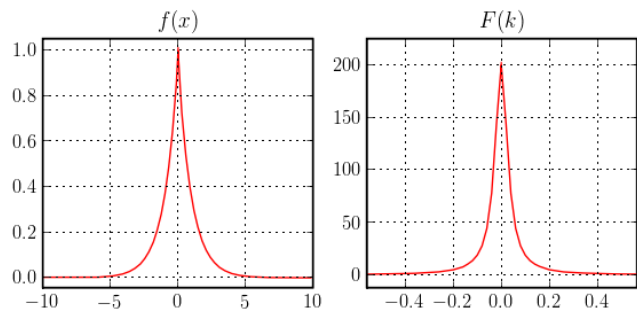
Abbildungung 50 zeigt einige Beispiele von Fouriertransformierten  $F(k)$  von Funktionen  $f(x)$ .



(a)  $f(x) = \text{Rechtecksfunktion}$



(b)  $f(x) = e^{-x^2}$



(c)  $f(x) = e^{-|x|}$

Abbildung 50: Beispiele einiger Fouriertransformationen

## B.2 Faltung

In der Mathematik (besonders in der Funktionalanalysis) und der Physik beschreibt die Faltung einen mathematischen Operator, der für zwei Funktionen  $f$  und  $g$  eine dritte Funktion liefert, welche typischerweise als eine modifizierte Version einer der beiden ursprünglichen Funktionen betrachtet wird. Die Faltung gibt eine Art „Überlappung“ zwischen  $f$  und einer gespiegelten und verschobenen Version von  $g$  an. Die Faltung ist ein mathematisches Werkzeug, welches Anwendung findet in der Statistik, Bild- und Signalverarbeitung und bei der Lösung von Differentialgleichungen. Die Faltung  $\otimes$  ist wie folgt definiert:

$$f(t) \otimes g(t) = \int_{-\infty}^{\infty} f(\zeta) \cdot g(t - \zeta) d\zeta \quad (\text{B.12})$$

Eigenschaften der Faltung:

$$f \otimes g = g \otimes f \quad (\text{B.13})$$

$$f \otimes (g \otimes h) = (f \otimes g) \otimes h \quad (\text{B.14})$$

$$f \otimes (g + h) = f \otimes g + f \otimes h \quad (\text{B.15})$$

$$f \otimes \delta = f, \quad (\text{B.16})$$

wobei  $\delta$  die Diracfunktion ist.

Abbildung 51 zeigt als Beispiel das Ergebnis der Faltung  $f(x) \otimes g(x)$  von zwei Rechtecksfunktionen  $f(x)$  und  $g(x)$ .

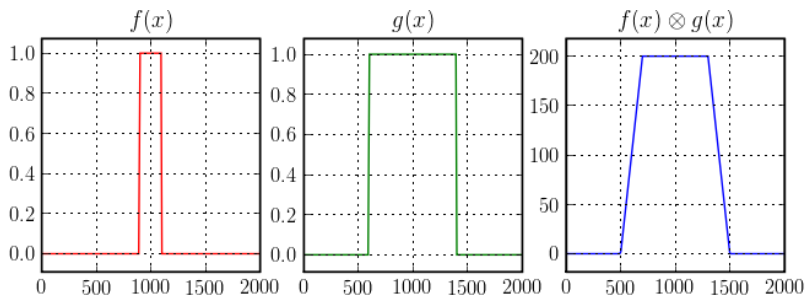


Abbildung 51: von links: Die Rechtecksfunktionen  $f(x)$  und  $g(x)$  und das Ergebnis ihrer Faltung  $f(x) \otimes g(x)$

## B.3 Faltungstheorem

Das Faltungstheorem besagt, dass die Fouriertransformierte einer Faltung zweier Funktionen  $f$  und  $g$  gleich dem Produkt der Fouriertransformierten

von  $f$  und  $g$  ist. Das heißt, die Faltung in einer Domäne (z.B. Ortsraum) entspricht einer Multiplikation in der anderen Domäne (z.B. reziproker Raum).

Seien  $f$  und  $g$  zwei Funktionen mit Faltung  $f \otimes g$  und  $\mathcal{F}$  die Fouriertransformation. Dann gilt:

$$\mathcal{F}(f \otimes g) = \mathcal{F}(f) \cdot \mathcal{F}(g) \quad (\text{B.17})$$

$$\mathcal{F}(f \cdot g) = \mathcal{F}(f) \otimes \mathcal{F}(g) \quad (\text{B.18})$$

$$f \otimes g = \mathcal{F}^{-1}(\mathcal{F}(f) \cdot \mathcal{F}(g)), \quad (\text{B.19})$$

wobei  $\mathcal{F}^{-1}$  die inverse Fouriertransformation darstellt.

## C Software

In diesem Anhang sollen in kompakter Form Informationen zu verwendeter Software und Links zu hilfreichen Dokumentationen angeboten werden, damit der interessierte Leser schnell und problemlos Änderungen an den geschriebenen Programm vornehmen kann.

### C.1 Python

Python ist eine Skriptsprache, die Anfang der 1990er-Jahre von Guido van Rossum am Centrum voor Wiskunde en Informatica [⇒] in Amsterdam entwickelt wurde und heute auf einer Vielzahl von Betriebssystem-Plattformen (Unix/Linux, Windows, MacOS, etc.) verfügbar ist. Sie eignet sich zur funktionalen und objektorientierten Programmierung. Ihre leicht lesbare Syntax und umfangreiche Standard-Bibliothek sowie eine Vielzahl von Erweiterungen aus den verschiedensten Bereichen (GUI, Netzwerke, Datenbanken, Graphik, 3D, Audio, Video, Web, GIS, Numerik, Spiele, etc.) haben sie zum Mittel der Wahl bei vielen Open-Source-Projekten und in namhaften Unternehmen und Organisationen gemacht.

Ein Python-Programm übersetzt seinen Quellcode transparent in Bytecode, der dann von einem Interpreter ausgeführt wird.

**NumPy** ist ein fundamentales Paket für Python, welches bietet:

- Unterstützung von  $N$ -dimensionalen Arrays
- Grundlegende Funktionen der linearen Algebra
- Grundlegende Funktionen zur Fouriertransformation
- Integrationsmöglichkeiten für Fortran, C und C++ Code

**SciPy** ist ein OpenSource-Paket für Python für Mathematik, Naturwissenschaft und Ingenieurwesen. Die SciPy Bibliothek benötigt das Paket NumPy. Die Bibliothek arbeitet mit NumPy-Arrays und bietet eine Vielzahl von nützlichen Funktionen, z.B. zur numerischen Integration oder Optimierung.

**Matplotlib / PyLab** ist eine Bibliothek für 2D- und 3D-Plots für Python. Die Plotfunktionen im PyLab-Interface weisen ein hohes Maß an Kompatibilität zu MatLab auf.



Offizielle Python Homepage	⇒
Das „Python OpenBook“	⇒
Offizielle NumPy Homepage	⇒
Offizielle SciPy Homepage	⇒
Offizielle Matlab homepage	⇒
Python Scripting for Computational Science	[Lan06]
Tutorial on Threads Programming with Python	⇒

Tabelle 4: Weiterführende Links zum Thema *Python*, NumPy und SciPy

## C.2 MySQL

MySQL ist ein relationales Datenbankverwaltungssystem und die populärste Open-Source-Datenbank der Welt.

Einer Datenbankmanagementsystem-Engine können mehrere Datenbanken zugeordnet werden. In einer Datenbank können mehrere Tabellen angelegt werden. Die Tabellen können von unterschiedlichem Typ sein. Die maximale Größe der Tabellen wird im Prinzip nur durch das Betriebssystem limitiert.

Mittels separater Bibliotheken steht für fast alle gängigen Programmiersprachen eine MySQL-Unterstützung zur Verfügung.

Offizielle MySQL Homepage	⇒
MySQL-Bibliothek für Python	⇒

Tabelle 5: Weiterführende Links zum Thema *MySQL*


## D Programme

In diesem Anhang befinden sich Anmerkungen zu den geschriebenen Python-Skripten sowie der vollständige Quelltext.

### D.1 Anmerkungen

Bei den Programmen handelt es sich um:

- `plot_interface.py` – Code benutzt für sämtliche Plots
- `bruteforce.py` – Code benutzt zur Berechnung sämtlicher Parameterkombinationen
- `sql1.py` und `sql2.py` – Code zur graphischen Auswertung der Datensätze, die durch `bruteforce.py` erhalten wurden

#### D.1.1 `plot_interface.py`

Um dieses Skript auszuführen werden eine Reihe von zusätzlichen Programmen und Python-Erweiterungen benötigt. Diese sind:

- Python (mit Tcl/Tk-Extensions)
- NumPy
- SciPy
- Matplotlib in der Version 0.91. In der neueren Version 0.98 kommt es zu Problemen mit dem Objekt „matplotlib.Axes3D“
- $\text{\LaTeX}$ – die Bibliothek Matplotlib greift auf zahlreiche Programme aus der TeX-Installation zu.

Hinweise zur Installation dieser Programme/Erweiterungen finden sich auf den jeweiligen Internetseiten dieser Projekte.

Abbildung 52 zeigt die grafische Oberfläche (GUI) des Programms. Neben der Eingabe sämtlicher numerischer Parameter, kann der Anwender zwischen den Berechnungen ebener und sphärischer Wellen wählen. Erstere folgt den Gleichungen aus Kapitel 6. Letztere (die Simulation sphärischer Wellen) ist zwar im Programmcode implementiert, war aber in Tests meist sehr problematisch. Die Anforderungen an Rechenzeit und Speicherplatz sind sehr hoch. Daher wird ausschließlich die Verwendung ebener Wellen zur Simulation empfohlen. Aufgrund der Schwierigkeiten im Umgang mit sphärischen Wellen, stehen einige Funktionen für diesen Wellentyp gar nicht



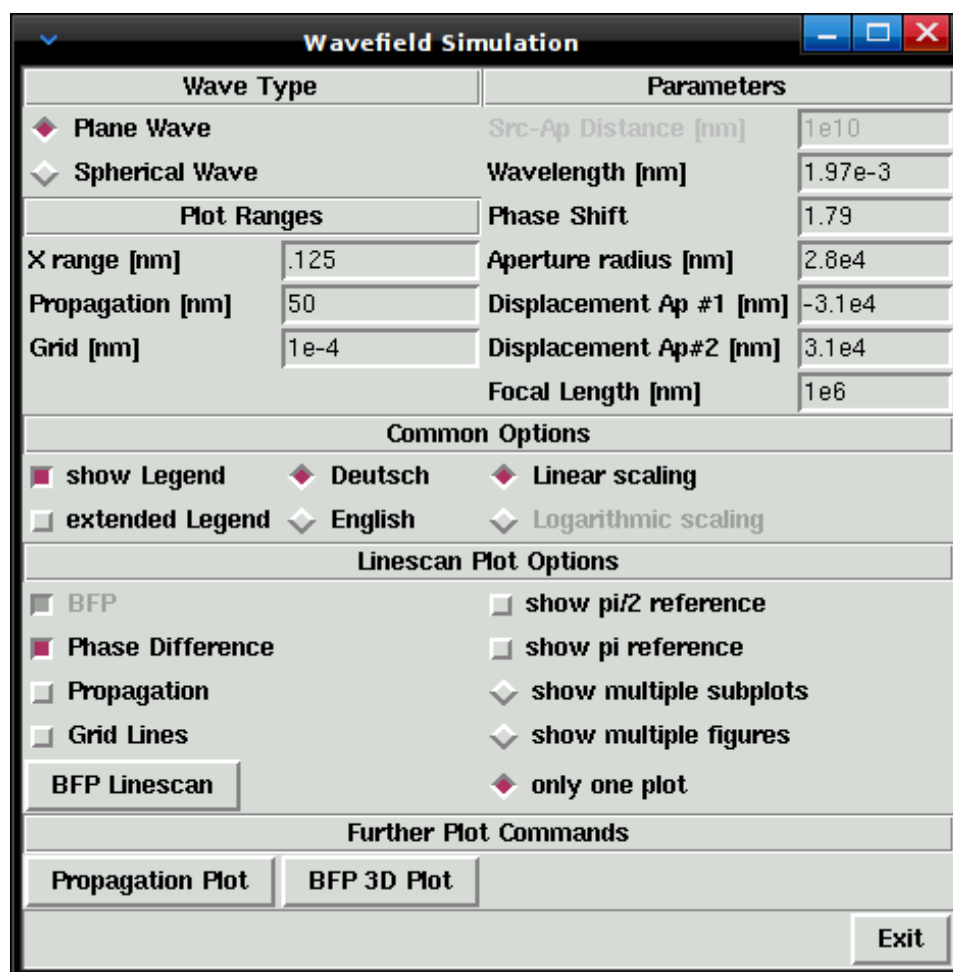


Abbildung 52: Screenshot des Hauptfensters der graphischen Oberfläche des Plot-tools

zur Verfügung (z.B. Plot des Phasenunterschiedes) oder liefern keine verwertbaren Ergebnisse.

Im oberen Bereich des Interfaces („Wave Type“, „Plot Range“, „Parameters“) wird der Anwender gebeten, den Wellentyp, den zu plottenden Bereich und die numerischen Parameter einzugeben.

**X range** Gewünschtes Intervall  $[-Wert, Wert]$  entlang der X-Achse in der BFP, das für den Plot verwendet werden soll. Aufgrund des verwendeten Romberg-Integrationsalgorithmus kann die Größe des Intervalls durch das Programm verändert werden.

**Propagation** Gewünschtes Intervall  $[-Wert, Wert]$  entlang der z-Achse

(Nullpunkt in BFP) für Propagationsdarstellungen oder gewünschtes Intervall entlang der Y-Achse in der BFP für 2D-Plots.

**Grid** Punktabstand in den beiden zuvor eingeführten Intervallen.

Im mittleren Bereich („Common Options“) der GUI wählt der Anwender zwischen linearer und logarithmischer Skalierung. Des weiteren kann er eine Legende erstellen („Show Legend“) und in dieser die aus Kapitel 6 bekannten Auswertungsgrößen ( $\mathcal{U}$ , FWHM, S/N) ausgeben lassen („extended Legend“). Mittels der Radiobuttons kann die Ausgabe in der Legende in einer gewünschten Sprache (Englisch oder Deutsch) erfolgen.

Darunter befindet sich der „Linescan Plot Options“ Bereich. Hier wählt der Anwender zwischen den unterschiedlichen Linien, die geplottet werden sollen:

**BFP** Plot des Betragsquadrats der Wellenfunktion in der BFP (siehe Gleichung 6.5)

**Phase Difference** Plot des Phasenunterschiedes in der BFP (siehe Gleichung 6.7)

**Propagation** Plot des Betragsquadrats der Wellenfunktion hinter der BFP (siehe Gleichung 5.1 und 6.5)

**Grid Lines** In den Plot werden schwarz gestrichelte horizontale und vertikale Linien eingefügt

Hat sich der Anwender für die Plot Option „Phase Difference“ entschieden, so stehen ihm in diesem Modus weitere Optionen zur Verfügung:

**show pi & pi/2 reference** Plot von Linien bei  $\pm\pi/2$  bzw.  $\pm\pi$

**show multiple subplots** Es werden die Intensitätsverteilungen und dazugehörigen Phasenverschiebungen in eine Figure aber in zwei separaten Subplotfenstern (siehe z.B. Abbildung 28 auf Seite 59) dargestellt

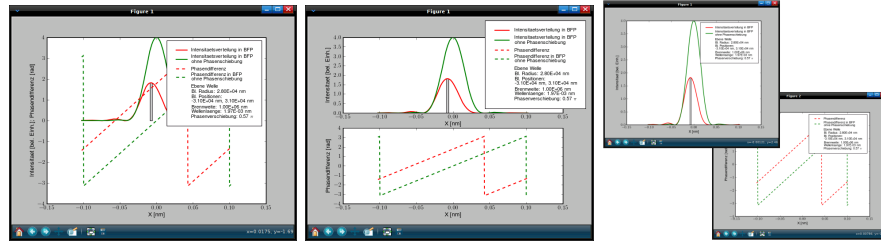
**show multiple figures** Es werden die Intensitätsverteilungen und dazugehörigen Phasenverschiebungen in zwei separate Figuren gezeichnet

**only one plot** Beides wird in einem Plotfenster dargestellt.

Die Ausgabe gemäß den letzten drei Optionen ist in Abbildung 53 gezeigt.

Abschließend hat der Anwender die Auswahl zwischen drei Plotbefehlen:

**BFP Linescan** Linescan entlang der X-Achse bei  $Y = 0$  in der BFP (siehe Abbildung 28)



(a) „only one plot“

(b) „subplots“

(c) „figures“

Abbildung 53: Screenshots der Ausgabe am Bildschirm bei der Plot Option „Phase Difference“. Die Situation in Abbildung (b) ist identisch zur Situation in Abbildung 28 (→ Seite 59)

**Propagation Plot** Darstellung von Linescans entlang der  $X$ -Achse bei  $Y = 0$  für unterschiedliche Propagationen entlang der optischen Achse

**BFP 3D Plot** Plot des Betragsquadrats der Wellenfunktion in der BFP

Die Plotbefehle „Propagation Plot“ und „BFP 3D Plot“ bieten nach Fertigstellung der Berechnung verschiedene Darstellungsstile (z.B. 3D Plot, Contour) und Farbschemata.

Bei den Befehlen „Propagation Plot“ und „BFP 3D Plot“ werden sehr viele Datenpunkte benötigt. Die Berechnung dieser ist sehr zeitaufwendig und erfordert eine große Menge an Arbeitsspeicher ( $>1\text{GB}$ ). Daher sind diese Funktionen mit Vorsicht zu genießen.

### D.1.2 bruteforce.py

Zur korrekten Ausführung dieses Skripts wird zusätzlich zu den Anforderungen des Skripts plot\_interface.py die MySQL-Bibliothek für Python („MySQLdb“) benötigt.

Dieses Skript berechnet die in Kapitel 6 eingeführten Auswertungsgrößen ( $\mathcal{U}$ , FWHM, S/N) und speichert diese mit den verwendeten Parametern (Radius, Verschiebungen der beiden Blenden, Phasenverschiebung, FWHM) in einer MySQL-Datenbank ab.

Dieses Programm wurde im Rahmen dieser Arbeit auf einem AMD Quadcore Rechner ausgeführt und lieferte 1.5 Mio Datensätze zur Auswertung innerhalb eines Tages zurück.

In diesem Skript und in dem Skript plot\_interface.py wird die FWHM durch folgendes Codesegment bestimmt:

```

1 # A ist ein numerisches Array, welches das Betragsquadrat
   der Wellenfunktion enthaelt
2 AM = argmax ( A )
3 # argmax() liefert den Index des Maximums
4 A1, A2 = A[:AM], A[AM:]
5 # diese Zuweisung spaltet das Array A in zwei Arrays A1 und
   A2
6 F1 = argmin ( abs ( A1 - .5 * A[AM] ) )
7 F2 = argmin ( abs ( A2 - .5 * A[AM] ) ) + AM
8 # argmin() liefert den Index des Minimums
9 # F1 und F2 sind die Indices der beiden Elemente von A, die
   am naechsten am Wert A[AM] sind
10 F = X [ F2 ] - X [ F1 ]
11 # durch diese Zuweisung enthaelt F die FWHM

```

### D.1.3 sql1.py und sql2.py

Das Skript sql1.py erzeugt aus den Datensätzen in der Datenbank graphische Auswertungen, wie sie in Abbildung 30 bzw. Abbildung 31 zu sehen sind. Dazu werden zwei SQL-Abfragen verwendet. Für jeden Blendenradius (angedeutet durch `--R--`) erzeugt die erste Abfrage eine Liste mit allen Blende-Blende-Abständen.

```

1 SELECT ( 'space1' - 'space2' ) AS 'sep'
2 FROM 'bruteforce' . 'values'
3 WHERE 'radius' = '--R--'
4 GROUP BY 'sep'
5 ORDER BY 'sep'

```

Da jeder Wert für den Blende-Blende-Abstand (`'space1' - 'space2'`; `space1` und `space2` sind die Verschiebungen der Lochmittelpunkte entlang der  $x$ -Achse) durch mehrere Kombinationen von `space1` und `space2` realisiert werden kann, gibt es möglicherweise mehrere Werte für  $\mathfrak{U}$  (`'signal'`) zu einem Blende-Blende-Abstand. Daher wird für jeden Blende-Blende-Abstand, welche die erste Abfrage zurück gab, nur der Datensatz mit dem maximalen  $\mathfrak{U}$  ausgewählt. Dies geschieht in der zweiten SQL-Abfrage. Diese wird für jeden Blende-Blende-Abstand durchgeführt (angedeutet durch `--S--`).

```

1 SELECT 'signal', 'phase', 'fwhm' * 1000, 's/n'
2 FROM 'bruteforce' . 'values'
3 WHERE
4 'radius' = '--R--'
5 AND
6 'space1' - 'space2' = '--S--'
7 ORDER BY 'signal' DESC
8 LIMIT 1

```

Durch die Kombination dieser beiden Abfragen erhält man für jeden Blende-Blende-Abstand (erste Abfrage) den Maximalwert von  $\mathfrak{U}$  und die anderen Werte, welche von Interesse sind (zweite Abfrage). Das Ergebnis ist in den Abbildungen 30 bis 34 zu sehen.

Im Skript `sql2.py` wird nach demselben Prinzip vorgegangen. Es wird zusätzlich zum Radius der Blende-Blende-Abstand konstant gehalten. Im Gegensatz zu `sql1.py` wird hier der Datensatz mit maximalem  $\mathfrak{U}$  für jede Verschiebung (`'space1'` + `'space2'`) gewählt. Siehe SQL-Abfragen im Quelltext: Zeilen 13–22 bzw. 27–36.

## D.2 Quelltext

Listing 1: `plot_interface.py`

```
1 #!/usr/bin/env python
2
3 from scipy.special import jv, jn_zeros
4 from scipy.integrate.quadrature import quadrature, romb
5 from scipy.integrate.quadpack import dblquad, quad
6 from scipy import fft, ifft, arange, zeros, ones, exp, sin,
   log, concatenate
7 from pylab import plot, semilogy, draw, show, legend,
   xlabel, ylabel, figure, ion, title, savefig, axis, hot,
   colorbar, imshow, contour, contourf, clabel, pcolormesh,
   subplot, grid, figlegend
8 from matplotlib.font_manager import FontProperties
9 from matplotlib import rc
10 from math import pi, log
11 from numpy import array, append, outer, insert, convolve,
   piecewise, where, fliplr, flipud, sign, sqrt, real,
   float32, argmin, argmax, max, min, sort
12 from operator import mod
13 from matplotlib.axes3d import Axes3D
14 from matplotlib.patches import Polygon
15 import matplotlib.cm as cm
16 from Tkinter import *
17 from time import sleep
18 import tkinterMessageBox
19 import threading
20
21 DropPoint = 20
22 MaxThreads = 300
23 MaxSubDiv = 1000
24 ThreadUnsafe = 1
25
26 String_DEU = [ \
27     'Intensitätsverteilung in BFP', \
28     'Intensitätsvert. %.2f nm hinter BFP', \
29     'Intensitätsverteilung in BFP\ohne
   Phasenschiebung', \
30     'Phasendifferenz', \
31     'Referenzlinie', \
32     'Intensität [bel. Einh.]', \
33     'Phasendifferenz [rad]', \
34     'Intensitätsverteilung bei\nterschiedenen
   Propagationen Z', \
35     'Ebene Welle', \
36     'Kugelwelle\nt Abstand Quelle-Blende:', \
```

```

37      'Bl. Radius:', \
38      'Bl. Positionen:', \
39      'Brennweite:', \
40      'Wellenlaenge:', \
41      'Phasenverschiebung:', \
42      'Phasendifferenz in BFP\ohne Phasenschiebung' ]
43 String_ENG = [ \
44      'Intensity distribution in BFP', \
45      'Intensity distr. %.2f nm behind BFP', \
46      'Intensity distribution in BFP\nwithout phase shift
      ', \
47      'Phase difference', \
48      'reference line', \
49      'Intensity [a.u.]', \
50      'Phase difference [rad]', \
51      'Intensity distribution for\ndifferent propagations
      Z', \
52      'Plane Wave', \
53      'Spherical Wave propagated till aperture:', \
54      'Ap. Radius:', \
55      'Ap. Positions:', \
56      'Focal Length:', \
57      'Wave Length:', \
58      'Phase Shift:', \
59      'Phase difference in BFP\nwithout phase shift' ]
60
61 def SWaves ( x, R, a1, a2, src, l, f, phi, dim ):
62     # this function returns the absolute square
63     # of a spherical wave regarding the
64     # dimension of the result array z:
65     # 1 dimension — Linescan in BFP
66     # 2 dimensions — Plot of whole BFP
67     global z
68     u = x / l / f
69     LenU = len ( u )
70     a = array ( [ a1, a2 ] )
71     Jobs = []
72
73     if not dim:
74         z = zeros ( 2 * LenU, 'Complex64' )
75         for U in range ( 0, LenU ):
76             job1 = sphere1 ( U, 0, u[U], 0, src
77                             , R, a.max(), 2 * pi / l )
78             Jobs.append ( job1 )
79
79             job2 = sphere1 ( U + LenU, 0, u[U],
80                             0, src, R, a.min(), 2 * pi / l
81                             )
82             Jobs.append ( job2 )
83
83             while ThreadUnsafe and threading.
84                 activeCount() > MaxThreads:
85                 sleep ( 1e-3 )
86
86             job1.start()

```

```
86         job2.start()
87
88     for Job in Jobs:
89         Job.join()
90
91     return z[:LenU] / sqrt ( (abs ( z[:LenU] )
92         **2).max() ) + z[LenU:] / sqrt ( (abs (
93         z[LenU:] )**2).max() ) * exp ( - 1j *
94         phi )
95
96 else:
97     o = zeros ( len ( u ), 'Complex64' )
98     z = outer ( o, o )
99     for U in range ( 0, len ( u ) ):
100         for V in range ( 0, len ( u ) ):
101             job = sphere2 ( U, V, u[U],
102                 u[V], src, R, a.max(),
103                 2 * pi / l )
104             Jobs.append ( job )
105             while ThreadUnsafe and
106                 threading.activeCount()
107                 > MaxThreads:
108                 sleep ( 1e-3 )
109             job.start()
110
111     for Job in Jobs:
112         Job.join()
113
114     del Jobs[:len(Jobs)]
115     tmp = z
116     for U in range ( 0, len ( u ) ):
117         for V in range ( 0, len ( u ) ):
118             job = sphere2 ( U, V, u[U],
119                 u[V], src, R, a.min(),
120                 2 * pi / l )
121             Jobs.append ( job )
122             while ThreadUnsafe and
123                 threading.activeCount()
124                 > MaxThreads:
125                 sleep ( 1e-3 )
126             job.start()
127
128     for Job in Jobs:
129         Job.join()
130
131     return tmp / sqrt ( (abs ( tmp )**2).max()
132         ) + z / sqrt ( (abs ( z )**2).max() ) *
133         exp ( - 1j * phi )
134
135 class sphere1 ( threading.Thread ):
136
137     def fkt ( self, x, y ):
138         r = sqrt ( x**2 + y**2 + self.src**2 )
139         return self.src / r * exp ( -1j * self.k *
140             r ) * exp ( 2j * pi * x * self.u + 2j *
```

```

        pi * y * self.v )
128
129     def run ( self ):
130         # this function returns the fourier
            transform
131         # of the wave function in the aperture
            plane
132         # for a 1D result array z
133         z[ self.ix ] = quadrature ( self.fkt , self.
            a - self.R, self.a + self.R, ( 0, ) )[0]
134     def __init__ ( self, ix, iy, u, v, src, R, a, k ):
135         threading.Thread.__init__ ( self )
136         self.ix = ix
137         self.iy = iy
138         self.u = u
139         self.v = v
140         self.src = src
141         self.R = R
142         self.a = a
143         self.k = k
144
145     class sphere2 ( threading.Thread ):
146
147         def integrand ( self, y, x ):
148             r = sqrt ( x**2 + y**2 + self.src**2 )
149             return self.src / r * exp ( -1j * self.k *
                r ) * exp ( 2j * pi * x * self.u + 2j *
                pi * y * self.v )
150
151         def q1 ( self, x ):
152             return quad ( self.integrand, -self.R, self
                .R, ( x, ), limit = MaxSubDiv )[0]
153
154         def run ( self ):
155             # this function returns the fourier
                transform
156             # of the wave function in the aperture
                plane
157             # for a 2D result array z
158             z[self.ix, self.iy] = quad ( self.q1, self.
                a - self.R, self.a + self.R, limit =
                MaxSubDiv )[0]
159
160         def __init__ ( self, ix, iy, u, v, src, R, a, k ):
161             threading.Thread.__init__ ( self )
162             self.ix = ix
163             self.iy = iy
164             self.u = u
165             self.v = v
166             self.src = src
167             self.R = R
168             self.a = a
169             self.k = k
170
171     # simple definitions of plane wave functions, phase shift,
172     # and Fresnel propagator
173     def Wave ( x, y, R, a, l, f, phi ):
174         r = sqrt ( x**2 + y**2 )

```



```

175         return jv ( 1, 2 * pi * R * r / l / f ) * exp ( - 1
                j * a * x / l / f - 1j * phi ) / pi / R / r * l
                * f
176
177 def Waves ( x, y, R, a1, a2, l, f, phi ):
178     return Wave ( x, y, R, a1, l, f, 0 ) + Wave ( x, y,
                R, a2, l, f, phi )
179
180 def Phase ( x, R, a1, a2, z, l, f, phi ):
181     return mod ( 1 * ( a2 - a1 ) * x / l / f + phi + pi
                , 2 * pi ) - pi
182
183 def FFTprop ( x, y, l, z, f ):
184     return exp ( 1j * pi * z * ( x**2 + y**2 ) / l / f
                **2 )
185
186 def Prop ( x, y, l, z ):
187     return exp ( - 1j * pi * ( x**2 + y**2 ) / l / z )
188
189 class Cal3D ( threading.Thread ):
190     def __init__ ( self, index, x, y, a, l, grid ):
191         threading.Thread.__init__ ( self )
192         self.i = index
193         self.x = x
194         self.y = y
195         self.a = a
196         self.A = fft ( a )
197         self.l = l
198         self.grid = grid
199
200     def run ( self ):
201         # this function calculates linescans for
202         # different propagation distances
203         global z
204         if abs(self.y) > 10 * self.l:
205             tmp = abs ( ifft ( self.A * fft (
                Prop ( self.x, 0, self.l, self.y )
                ) ) * 1j / self.l / self.y )
                **2
206             L = int ( .5 * len ( tmp ) )
207             tmp = concatenate ( ( tmp[L:], tmp[:
                L] ) )
208         else:
209             tmp = abs ( ifft ( self.A ) )**2
210
211             tmp = tmp / romb ( tmp, self.grid ) * romb
                ( abs ( self.a )**2, self.grid )
212             z[self.i] = tmp[:, DropPoint]
213
214 # the classes NormA and NormB contain simple functions
215 # for normalizing the wave function and return its absolute
    square
216 class NormA ( threading.Thread ):
217     def __init__ ( self, dx ):
218         threading.Thread.__init__ ( self )
219         self.x = dx
220

```

```

221     def run ( self ):
222         global A
223         A = abs(A)**2 / romb ( abs(A)**2, self.x )
224
225 class NormB ( threading.Thread ):
226     def __init__ ( self, dx ):
227         threading.Thread.__init__ ( self )
228         self.x = dx
229
230     def run ( self ):
231         global B
232         B = abs(B)**2 / romb ( abs(B)**2, self.x )
233
234 class GUI:
235     def CrArrayX ( self ):
236         # this function returns an array with X
237         # values
238         # regarding the length requirements for the
239         # integration algorithm
240         N = int ( log ( self.xr / self.st - 1, 2 )
241                 ) + 1
242         Xrange = .5 * ( 1 + 2**N ) * self.st
243         self.X = arange ( -Xrange, Xrange, self.st,
244                         'Float32' )
245         return 0
246
247     def CrArrayY ( self ):
248         # does the same as CrArrayX
249         self.Y = arange ( -self.yr, self.yr, 2. *
250                         self.yr / len ( self.X[:, DropPoint] ), '
251                         Float32' )
252         return 0
253
254     def Propagate ( self, a ):
255         # this function return the propagated wave
256         # function
257         return ifft ( fft ( a ) * fft ( Prop ( self
258         .X, 0, self.l, self.yr ) ) ) * 1j / self
259         .l / self.yr
260
261     def CheckVar ( self ):
262         # this function converts all entry field
263         # values
264         # into float variables and sets the
265         # language
266
267         if self.lang.get() == 0:
268             self.Dict = String_DEU
269         elif self.lang.get() == 1:
270             self.Dict = String_ENG
271         else:
272             tkinter.messagebox.showwarning ( 'No
273             language chosen', 'Will use
274             german language pack.' )
275             self.Dict = String_DEU

```

```
265         try:
266             self.l = float ( self.Wavelength.
267                             get() )
268             self.xr = float ( self.XRange.get()
269                             )
270             self.yr = float ( self.YRange.get()
271                             )
272             self.st = float ( self.Step.get() )
273             self.R = float ( self.DiaAp.get() )
274             self.d1 = float ( self.DisA1.get()
275                             )
276             self.d2 = float ( self.DisA2.get()
277                             )
278             self.f = float ( self.Focus.get() )
279             self.phi = float ( self.PhSh.get()
280                             )
281             self.src = float ( self.SrAp.get()
282                             )
283             if self.R <= 0:
284                 tkinter.messagebox.showerror ( '
285                                     Invalid radius', 'Radius
286                                     needs to be greater
287                                     than zero.' )
288                 return 1
289             if self.R > abs ( self.d1 - self.d2
290                             ):
291                 tkinter.messagebox.showerror ( '
292                                     Radius greater than
293                                     separation', '''Radius
294                                     has to be smaller than
295                                     the
296                                     separation of the apertures
297                                     .''' )
298                 return 1
299             return 0
300         except ValueError:
301             tkinter.messagebox.showerror ( 'No
302                                     numeric Value', 'At least one
303                                     quantity is no integer or float.
304                                     ' )
305             return 1
306
307     def plotLL ( self, sb, x, y, style ):
308         # called by the plotting functions;
309         # causes linear or logarithmic plots
310         if self.scaling.get():
311             return sb.semilogy ( x, y, style,
312                                 linewidth = 2 )
313         return sb.plot ( x, y, style, linewidth = 2
314                         )
315
316     def plotbfp ( self ):
317         # this function return a plot of the back
318         # focal plane
```

```

300         if self.CheckVar(): return 1
301
302         self.CrArrayX()
303         self.Y = self.X
304         L = len ( self.X )
305         Xmin = argmin ( abs ( self.X ) )
306         o = ones ( L, 'Complex64' )
307
308         if not self.Wave.get():
309             self.z = outer ( o, o )
310
311             for i in range ( 0, L ):
312                 self.z[i] = abs ( Waves (
313                     self.X, self.X[i], self.
314                     R, self.d1, self.d2,
315                     self.l, self.f, self.phi
316                     ) )**2
317
318         else:
319             self.z = abs ( SWaves ( self.X,
320                 self.R, self.d1, self.d2, self.
321                 src, self.l, self.f, self.phi, 1
322                 ) )**2
323
324         self.Data = []
325         self.Labels = []
326
327         self.Data.append ( self.Dict[0] )
328
329         # the following lines perform calculations
330         # for FWHM, S/N
331         # Usable Signal is not calculated due to
332         # problems
333         # with limitation in recursion depth of
334         # integration algorithms
335         if self.legend[1].get():
336             AM = argmax ( self.z[Xmin] )
337             A1, A2 = self.z[Xmin][:AM], self.z[
338                 Xmin][AM:]
339             F1 = argmin ( abs ( A1 - .5 * self.
340                 z[Xmin][AM] ) )
341             F2 = argmin ( abs ( A2 - .5 * self.
342                 z[Xmin][AM] ) )
343             F = self.X [ AM + F2 ] - self.X [
344                 F1 ]
345
346             G = dblquad ( lambda y,x: abs (
347                 Waves ( x, y, self.R, self.d1,
348                 self.d2, self.l, self.f, self.
349                 phi ) )**2, self.X[F1], self.X[
350                 AM + F2], lambda x: - sqrt ( .25
351                 * F**2 + ( x - self.X[AM] )**2
352                 ), lambda x: sqrt ( .25 * F**2 +
353                 ( x - self.X[AM] )**2 ) ) [0] \
354             / dblquad ( lambda y,x: abs ( Waves
355                 ( x, y, self.R, self.d1, self.

```

```

334         d2, self.l, self.f, self.phi ))
335         **2, self.X[0], self.X[ len (
            self.X ) - 1], lambda x: self.X
            [0], lambda x: self.X[ len (
            self.X ) - 1])[0]
336
337     self.Data.append ( ''.join (( 'FWHM
338         , S/N: \n', '%.2f nm, %.2f' % (
339         F, G * 100 ), '%\n' )) )
340
341     self.Labels.append ( 'X [nm]' )
342     self.Labels.append ( 'Y [nm]' )
343     self.Labels.append ( self.Dict[5] )
344     self.style3D()
345     return 0
346
347 def plot2d ( self ):
348     # this function returns linescans
349
350     # check whether all variables are properly
351     # set
352     if self.CheckVar(): return 1
353     rc ( 'text', usetex = True )
354
355     # define X axis
356     self.CrArrayX()
357     co = .5 * len ( self.X )
358
359     global A
360     global B
361
362     if self.subplots.get() == 1:
363         sb1 = subplot ( 211 )
364         ylabel ( self.Dict[5] )
365     elif self.subplots.get() == 2:
366         fig1 = figure()
367         sb1 = subplot ( 111 )
368         ylabel ( self.Dict[5] )
369     else:
370         sb1 = subplot ( 111 )
371         if self.opt[1].get():
372             ylabel ( ''.join (( self.
373                 Dict[5], '; ', self.Dict
374                 [6] )) )
375         else:
376             ylabel ( self.Dict[5] )
377
378     xlabel ( 'X [nm]' )
379
380     # printing the grid lines in linescans
381     grid ( self.opt[3].get() )
382
383     if self.opt[0].get():
384         if not self.Wave.get():
385             A = Waves ( self.X, 0, self
386                 .R, self.d1, self.d2,
387                 self.l, self.f, self.phi

```

```

379         )
380     else:
381         A = SWaves ( self.X, self.R
382                     , self.d1, self.d2, self
383                     .src, self.l, self.f,
384                     self.phi, 0 )
385
386     else:
387         tkMessageBox.showerror ( 'No wave
388                                 function in BFP', 'Wave function
389                                 in BFP is required...' )
390
391     return 1
392
393     self.Lines, self.Data = [], []
394     self.Lines.append ( self.plotLL ( sb1, self
395                                     .X, abs(A)**2, 'r-' ) )
396     self.Data.append ( self.Dict[0] )
397
398     if self.opt[2].get():
399         # propagating distance is not equal
400         # zero and
401         # plot of phase difference is not
402         # desired
403         if self.yr:
404             # Wave A is propagated and
405             # stored in B
406             B = self.Propagate ( A )
407             normF = romb ( abs(A)**2,
408                           self.st ) / romb ( abs(B
409                                               )**2, self.st )
410             B = concatenate (( B[co:],
411                               B[:co] ))
412             B = B * sqrt ( normF )
413         else:
414             tkMessageBox.showerror ( '
415                                     No Propagation Distance',
416                                     'Non-zero propagation
417                                     distance required...' )
418             return 1
419
420     self.Lines.append ( self.plotLL (
421                             sb1, self.X, abs(B)**2, 'g-' ) )
422     self.Data.append ( self.Dict[1] %
423                       self.yr )
424
425     if self.opt[1].get():
426         # Plot phase difference with or
427         # without pi/2 reference line
428         A2 = abs ( Waves ( self.X, 0, self.
429                             R, self.d1, self.d2, self.l,
430                             self.f, 0 ) )**2
431
432     self.Lines.append ( self.plotLL (
433                             sb1, self.X, A2, 'g-' ) )
434     self.Data.append ( self.Dict[2] )
435
436     B = Phase ( self.X, self.R, self.d1
437                , self.d2, 0, self.l, self.f,
438                self.phi )

```

```
413         percent = .05
414         C = where ( ( B < .5 * pi + percent
415                     ) & ( B > .5 * pi - percent ) )
416         # Plot of a grey polygon for
417         # highlighting the
418         # part of the wave with a phase
419         # difference
420         # of pi/2
421         xx = ones( len ( C[0] ), 'Float32' )
422         aa = ones( len ( C[0] ), 'Complex64' )
423         ref = .5 * pi * ones ( len ( self.X
424                               ), 'Float32' )
425
426         for i in range ( 0, len ( C[0] ) ):
427             xx[i] = self.X[C[0][i]]
428             aa[i] = A[C[0][i]]
429
430         am = argmin ( abs ( xx ) )
431         bm = am
432
433         while bm > 0 :
434             if C[0][bm] == C[0][bm-1] +
435                1:
436                 bm -= 1
437             else:
438                 break
439
440         while am < len ( C[0] ) - 1 :
441             if C[0][am] == C[0][am+1] -
442                1:
443                 am += 1
444             else:
445                 break
446
447         xxx = arange ( self.X[C[0][bm]],
448                       self.X[C[0][am]], self.st )
449         if not self.Wave.get():
450             yyy = abs ( Waves ( xxx, 0,
451                                self.R, self.d1, self.
452                                d2, self.l, self.f, self
453                                .phi ) )**2
454         else:
455             yyy = abs ( SWaves ( xxx,
456                                 self.R, self.d1, self.d2
457                                 , self.src, self.l, self
458                                 .f, self.phi, 0 ) )**2
459
460         verts = [ ( self.X[C[0][bm]], 0 ) ]
461         + zip ( xxx, yyy ) + [ ( self.X[C[0][am]], 0 ) ]
462         verts = [ ( xxx[0], 0 ) ] + zip (
463             xxx, yyy ) + [ ( xxx[len(xxx)
464                               -1], 0 ) ]
465         poly = Polygon ( verts, facecolor =
466                         '0.8', edgecolor = 'k' )
467         sb1.add_patch ( poly )
```

```

451
452         if self.subplots.get() == 1:
453             sb2 = subplot ( 212 )
454             grid ( self.opt[3].get() )
455             xlabel ( 'X [nm]' )
456             if self.opt[1].get():
457                 ylabel ( self.Dict
458                     [6] )
459         elif self.subplots.get() == 2:
460             self.mkLegend( 0 )
461             del self.Lines[:]
462             del self.Data[:]
463             fig2 = figure()
464             grid ( self.opt[3].get() )
465             sb2 = subplot ( 111 )
466             xlabel ( 'X [nm]' )
467             if self.opt[1].get():
468                 ylabel ( self.Dict
469                     [6] )
470         else:
471             sb2 = sb1
472
473         self.Lines.append ( self.plotLL (
474             sb2, self.X, B, 'r—' ) )
475         self.Data.append ( self.Dict[3] )
476         self.Lines.append ( self.plotLL (
477             sb2, self.X, Phase ( self.X,
478                 self.R, self.d1, self.d2, 0,
479                 self.l, self.f, 0 ), 'g—' ) )
480         self.Data.append ( self.Dict[15] )
481
482         if self.pi2.get():
483             self.Lines.append ( self.
484                 plotLL ( sb2, self.X,
485                     ref, 'b:' ) )
486             self.Data.append ( ''.join
487                 (( r '$\frac{\pi}{2}$ ',
488                     self.Dict[4] )) )
489             self.Lines.append ( self.
490                 plotLL ( sb2, self.X, -
491                     ref, 'b:' ) )
492             self.Data.append ( ''.join
493                 (( r '$-\frac{\pi}{2}$ ',
494                     self.Dict[4] )) )
495
496         if self.pi.get():
497             self.Lines.append ( self.
498                 plotLL ( sb2, self.X, 2
499                     * ref, 'r:' ) )
500             self.Data.append ( ''.join
501                 (( r '$\pi$ ', self.Dict
502                     [4] )) )
503             self.Lines.append ( self.
504                 plotLL ( sb2, self.X, -2
505                     * ref, 'r:' ) )

```



```

486         self.Data.append ( ''.join
                               (( r '$-\pi$ ', self.Dict
                                [4] )) )
487
488     # Calculating Usable Signal, FWHM, S/N if
        desired
489     if self.legend[1].get():
490         if self.opt[1].get():
491             D = 2 * abs ( mod ( B / pi
                                + .5, 1 ) - .5 ) * sign
                                ( B / pi )
492             E = romb ( D * abs(A)**2,
                          self.st ) / romb ( abs(A)
                                                **2, self.st )
493             A = abs ( A )**2
494             AM = argmax ( A )
495             A1, A2 = A[:AM], A[AM:]
496             F1 = argmin ( abs ( A1 - .5 * A[AM]
                                ) )
497             F2 = argmin ( abs ( A2 - .5 * A[AM]
                                ) )
498             F = self.X [ AM + F2 ] - self.X [
                                F1 ]
499
500             G = quad ( lambda x: abs ( Waves (
                                x, 0, self.R, self.d1, self.d2,
                                self.l, self.f, self.phi ))**2,
                          self.X[F1], self.X[ AM + F2 ])
                                [0]\
501             / quad ( lambda x: abs ( Waves ( x,
                                0, self.R, self.d1, self.d2,
                                self.l, self.f, self.phi ))**2,
                          self.X[0], self.X[ len ( self.X
                                ) - 1 ] ) [0]
502
503         if not self.opt[1].get():
504             self.Data.append ( ''.join
                                (( 'FWHM, S/N:\n', '%.2f
                                    nm, %.2f' % ( F, G *
                                    100 ), '\%' )) )
505
506         else:
            self.Data.append ( ''.join
                                (( 'Usable Signal, FWHM,
                                    S/N:\n', '%.2f' % ( E
                                    * 100 ), '\%', ', ', %.2f
                                    nm' % F, ', ', %.2f' % ( G
                                    * 100 ), '\%' )) )
507
508     # draw a legend
509     self.mkLegend( self.subplots.get() )
510
511     show()
512     return 0
513
514     def plot3d ( self ):

```

---

```

515         # this function returns a plot of the
516         propagated wave
517     # Check whether all variables are valid
518     format
519     # and whether propagation not equal zero
520     if self.CheckVar(): return 1
521     if self.yr == 0 :
522         tkMessageBox.showerror ( 'Zero
523         Propagation', '''Integer or
524         float greater than zero
525         required for 3D plot.''' )
526     return 1
527
528     self.Data = []
529     self.Labels = []
530
531     self.Data.append ( self.Dict[7] )
532     self.Labels.append ( 'X [nm]' )
533     self.Labels.append ( 'Z [nm]' )
534     self.Labels.append ( self.Dict[5] )
535     # Define X and Y range
536     self.CrArrayX()
537     global DropPoint
538     DropPoint = int ( len ( self.X ) / 150 )
539     self.CrArrayY()
540     if not self.Wave.get():
541         A = Waves ( self.X, 0, self.R, self
542         .d1, self.d2, self.l, self.f,
543         self.phi )
544     else:
545         A = SWaves ( self.X, self.R, self.
546         d1, self.d2, self.src, self.l,
547         self.f, self.phi, 0 )
548         Aref = SWaves ( self.X, self.R,
549         self.d1, self.d2, self.src, self
550         .l, self.f, 0, 0 )
551         A = A * 2 / sqrt ( max ( abs(Aref)
552         **2 ) )
553     co = int ( .5 * len ( self.X[:, DropPoint] )
554     )
555     o = ones ( len(self.X[:, DropPoint]), '
556     Float32' )
557
558     # Define 2D-arrays for each axis
559     global z
560     z = outer ( o, o )
561     Jobs = []
562
563     # insert values into 2D-array for different
564     propagations
565     for i in range ( 0, len ( z ) ):
566         Job = Cal3D ( i, self.X, self.Y[i],
567         A, self.l, self.st )
568         Jobs.append ( Job )
569         while ThreadUnsafe and threading.
570         activeCount() > MaxThreads:

```

---

```
556                 sleep ( 1e-3 )
557                 Job.start()
558
559         for job in Jobs:
560             job.join()
561
562         self.X = self.X[:, DropPoint]
563         self.z = z
564         self.style3D()
565         return 0
566
567     def style3D ( self ):
568         # this function is part of the GUI and asks
569         # for specifications of the plotting style
570         self.d3style = IntVar()
571         self.d3color = IntVar()
572         colormaps = [ 'autumn', 'bone', 'cool', ' ',
573                     'copper', 'flag', 'gray', 'hot', 'hsv', ' ',
574                     'jet', 'pink', 'prism', 'spring', 'summer',
575                     ' ', 'winter', 'spectral' ]
576
577         self.top = Toplevel ( self.root )
578         frame = Frame ( self.top )
579         frame.pack()
580         label = Label ( frame, text = 'Plotting
581                        Style', relief = 'groove' )
582         label.grid ( row = 0, column = 0, sticky =
583                     'wens' )
584         button = Radiobutton ( frame, text = '
585                        Colormesh', variable = self.d3style,
586                        value = 0 )
587         button.grid ( row = 1, column = 0,
588                     columnspan = 2, sticky = 'w' )
589         button = Radiobutton ( frame, text = '
590                        Colormesh & Contour', variable = self.
591                        d3style, value = 1 )
592         button.grid ( row = 2, column = 0,
593                     columnspan = 2, sticky = 'w' )
594         button = Radiobutton ( frame, text = '3D
595                        Plot', variable = self.d3style, value =
596                        2 )
597         button.grid ( row = 3, column = 0,
598                     columnspan = 2, sticky = 'w' )
599         button = Radiobutton ( frame, text = '
600                        Contour', variable = self.d3style, value
601                        = 3 )
602         button.grid ( row = 4, column = 0,
603                     columnspan = 2, sticky = 'w' )
604         button = Radiobutton ( frame, text = '
605                        Contour ( Filled )', variable = self.
606                        d3style, value = 4 )
607         button.grid ( row = 5, column = 0,
608                     columnspan = 2, sticky = 'w' )
```

```

591         label = Label ( frame, text = 'Color Maps:'
592                        , relief = 'groove' )
593         label.grid ( row = 6, column = 0,
594                    columns = 2, sticky = 'wens' )
595         for i in range ( 0, len ( colormaps ) ):
596             button = Radiobutton ( frame, text
597                                = colormaps[i], variable = self.
598                                d3color, value = i, state = '
599                                normal' )
600             button.grid ( row = 7 + i - i%2,
601                        column = i%2, sticky = 'w' )
602             if colormaps[i] == 'jet':
603                 button.select()
604         button = Button ( frame, text = 'OK',
605                        command = self.style3Dplot )
606         button.grid ( row = 0, column = 1, sticky =
607                    'nswe' )
608         return 0
609
610     def style3Dplot ( self ):
611         # this function returns the final figures
612         # for the
613         # plotting commands that are callable in
614         # the main
615         # window of the GUI (Propagation Plot, BFP
616         # 3D Plot)
617
618         colormap = int ( self.d3color.get() )
619         colormaps = [ cm.autumn, cm.bone, cm.cool,
620                    cm.copper, cm.flag, cm.gray, cm.hot, cm.
621                    hsv, cm.jet, cm.pink, cm.prism, cm.
622                    spring, cm.summer, cm.winter, cm.
623                    spectral ]
624         self.top.destroy()
625
626         figure ( figsize = ( 8, 5 ) )
627
628         self.Lines = []
629         if self.d3style.get() == 4:
630             xlabel ( self.Labels[0] )
631             ylabel ( self.Labels[1] )
632             title ( self.Labels[2] )
633             Extent = ( self.X[0], self.X[len(
634                 self.X)-1], self.Y[0], self.Y[
635                 len(self.Y)-1] )
636             contourf ( self.z.real, arange ( 0,
637                    4, .025 ), origin = 'lower',
638                    cmap = colormaps[colormap],
639                    extent = Extent )
640             colorbar()
641
642         elif self.d3style.get() == 3:
643             xlabel ( self.Labels[0] )
644             ylabel ( self.Labels[1] )

```

```

626         Extent = ( self.X[0], self.X[len(
                    self.X)-1], self.Y[0], self.Y[
                    len(self.Y)-1] )
627         contour ( self.z.real, arange ( 0,
                    4, .025 ), origin = 'lower',
                    linewidth = 1, cmap = colormaps[
                    colormap], extent = Extent )
628         colorbar()
629     elif self.d3style.get() == 2:
630         o = ones ( len ( self.X ), 'Float32'
                    , )
631         x = outer ( o, self.X )
632         y = outer ( self.Y, o )
633         fig = figure()
634         ax = Axes3D ( fig )
635         ax.set_xlabel ( self.Labels[0] )
636         ax.set_ylabel ( self.Labels[1] )
637         ax.set_zlabel ( self.Labels[2] )
638         self.Lines.append ( ax.
                    plot_wireframe ( x, y, self.z.
                    real ) )
639     elif self.d3style.get() == 1:
640         xlabel ( self.Labels[0] )
641         ylabel ( self.Labels[1] )
642         Extent = ( self.X[0], self.X[len(
                    self.X)-1], self.Y[0], self.Y[
                    len(self.Y)-1] )
643         pcolormesh ( self.X, self.Y, self.z
                    .real, cmap = cm.gray, vmin = 0,
                    vmax = 4 )
644         cset = contour ( self.z.real,
                    arange ( 0, 4, .025 ), origin =
                    'lower', linewidth = 2, cmap =
                    colormaps[colormap] , extent =
                    Extent )
645 #         label ( cset, inline = 1, fmt =
                    '%.1f', fontsize = 10 )
646         colorbar()
647     else:
648         xlabel ( self.Labels[0] )
649         ylabel ( self.Labels[1] )
650         self.Lines.append ( pcolormesh (
                    self.X, self.Y, self.z.real,
                    cmap = colormaps[colormap], vmin
                    = 0, vmax = 4 ) )
651         colorbar()
652
653         self.mkLegend( 0 )
654         show()
655         return 0
656
657     def mkLegend ( self, fl ):
658         # this function generates the basic legend
659
660         if self.legend[0].get():

```

```

661         if not self.Wave.get():
662             wType = ''.join(( self.
                Dict[8], '\n' ))
663         else:
664             wType = ''.join(( self.
                Dict[9], '%.2E nm' %
                self.src ))
665
666         self.Data.append ( ''.join (( wType
                , self.Dict[10], ' %.2E nm\n' %
                ( self.R ), \
667         self.Dict[11], '\n%.2E nm, %.2E nm\
                n' % ( self.d1, self.d2 ), \
668         self.Dict[12], ' %.2E nm\n' % self.
                f, self.Dict[13], ' %.2E nm\n' %
                self.l, \
669         self.Dict[14], ' %.2f $\pi$' % (
                self.phi / pi ) )) )
670
671         if fl == 1:
672             figlegend ( self.Lines,
                self.Data, 1, numpoints
                = 3, prop =
                FontProperties ( size =
                'small' ) )
673         else:
674             legend ( self.Lines, self.
                Data, 1, numpoints = 3,
                prop = FontProperties (
                size = 'small' ) )
675
676         return 0
677     def ToggleWave ( self ):
678         # this function en- and disables the entry
679         # field
680         # for source-aperture distance
681         # and the buttons for plotting phase
682         # differences and swapping
683
684         if self.Wave.get():
685             act1 = 'normal'
686             act2 = 'disabled'
687             col = 'black'
688             self.P2P.select()
689             self.PPD.deselect()
690         else:
691             act1 = 'disabled'
692             act2 = 'normal'
693             col = 'grey'
694
695         self.SrAp.configure ( state = act1 )
696         self.SAL.configure ( fg = col )
697         self.PPD.configure ( state = act2 )
698         self.SrAp.update_idletasks()
699         self.SAL.update_idletasks()
700         self.PPD.update_idletasks()
701         self.TogglePO()

```

```

700         return 0
701
702     def TogglePO ( self ):
703         # this function en- and disables the
704         # checkbuttons
705         # for plotting the pi/2 reference line
706         if self.opt[1].get():
707             act1 = 'normal'
708             act2 = 'disabled'
709             self.plotLin.select()
710         else:
711             act1 = 'disabled'
712             act2 = 'normal'
713             self.RefLine1.deselect()
714             self.RefLine2.deselect()
715             self.RefLine1.configure ( state = act1 )
716             self.RefLine1.update_idletasks()
717             self.RefLine2.configure ( state = act1 )
718             self.RefLine2.update_idletasks()
719             self.Multi1.configure ( state = act1 )
720             self.Multi2.configure ( state = act1 )
721             self.Multi3.configure ( state = act1 )
722             self.Multi1.update_idletasks()
723             self.Multi2.update_idletasks()
724             self.Multi3.update_idletasks()
725             self.plotLog.configure ( state = act2 )
726             self.plotLog.update_idletasks()
727         return 0
728
729     def ToggleLeg ( self ):
730         # this function en- and disables the
731         # extended legend checkbutton
732
733         if self.legend[0].get():
734             act = 'normal'
735         else:
736             act = 'disabled'
737             for i in range ( 1, len ( self.legb
738                                     ) ):
739                 self.legb[i].deselect()
740
741             for i in range ( 1, len ( self.legb ) ):
742                 self.legb[i].configure ( state =
743                                         act )
744                 self.legb[i].update_idletasks()
745
746     def Shutdown ( self ):
747         # this function writes numeric parameters
748         # to the hard drive
749         # before shutdown
750         ConfFile = open ( '.WafeFieldConf', 'w' )
751         ConfFile.write ( '\t'.join (( self.XRange.
752                                     get(), \
753                                     self.YRange.get(), self.Step.get(), \

```

```
750         self.SrAp.get(), self.Wavelength.get(),
751         self.PhSh.get(), \
752         self.DiaAp.get(), \
753         self.DisA1.get(), self.DisA2.get(), self.
754         Focus.get() )) )
755     ConfFile.close()
756     root.destroy()
757     return 0
758
759 def StartUp ( self ):
760     # this function reads a file saved by the
761     # funtion Shutdown,
762     # if present, and inserts these values into
763     # the entry fields.
764     # If not present, a set of default values
765     # is used
766
767     try:
768         ConfFile = open ( '.WafeFieldConf',
769                           'r' )
770         self.Parameters = ConfFile.readline
771         ().split()
772     except:
773         self.Parameters = [ '1', '50', '4e
774                             -4', '1e10', '4e-3', '1.9', '2e4
775                             ', '-1e4', '4e4', '1e6' ]
776
777     return 0
778
779 def __init__ ( self, root ):
780     # this function defines the main window of
781     # the GUI
782
783     self.opt = []
784     # these variables determine the different
785     # plot options
786     self.opt.append ( IntVar() )
787     self.opt.append ( IntVar() )
788     self.opt.append ( IntVar() )
789     self.opt.append ( IntVar() )
790     # this variable determines the wave type
791     self.Wave = IntVar()
792     # these variables control plotting of pi/2
793     # and pi reference line and subplots
794     self.pi2 = IntVar()
795     self.pi = IntVar()
796     self.subplots = IntVar()
797     # this variable switches between linear and
798     # logarithmic scaling
799     self.scaling = IntVar()
800     # this variable swaps left and right halves
801     # of the plot
802     self.swap = IntVar()
803     # this variable controls output of the
804     # legend
805     self.legend, self.legb = [], [ '', '' ]
806     self.legend.append ( IntVar() )
```



```
792         self.legend.append ( IntVar() )
793         # this variable changes the language of
           legends and titles
794         self.lang = IntVar()
795
796         self.root = root
797
798         self.StartUp()
799
800         frame = Frame ( self.root, relief = 'sunken
           ' )
801         frame.pack ( side = 'top' )
802
803         label = Label ( frame, text = 'Wave Type',
           relief = 'groove' )
804         label.grid ( row = 0, column = 0, sticky =
           'we', columnspan = 2 )
805
806         button = Radiobutton ( frame, text = 'Plane
           Wave', variable = self.Wave, value = 0,
           command = self.ToggleWave )
807         button.select()
808         button.grid ( row = 1, column = 0,
           columnspan = 2, sticky = 'w' )
809         button = Radiobutton ( frame, text = '
           Spherical Wave',
810         variable = self.Wave, value = 1, command =
           self.ToggleWave )
811         button.grid ( row = 2, column = 0,
           columnspan = 2, sticky = 'w' )
812
813         label = Label ( frame, text = 'Plot Ranges',
           relief = 'groove' )
814         label.grid ( row = 3, column = 0, sticky =
           'we', columnspan = 2 )
815         label = Label ( frame, text = 'X range [nm]
           ' )
816         label.grid ( row = 4, column = 0, sticky =
           'w' )
817         label = Label ( frame, text = 'Propagation
           [nm]' )
818         label.grid ( row = 5, column = 0, sticky =
           'w' )
819         label = Label ( frame, text = 'Grid [nm]' )
820         label.grid ( row = 6, column = 0, sticky =
           'w' )
821
822         self.XRange = Entry ( frame, width = 10 )
823         self.XRange.insert ( END, self.Parameters
           [0] )
824         self.XRange.grid ( row = 4, column = 1,
           sticky = 'we' )
825         self.YRange = Entry ( frame, width = 10 )
826         self.YRange.insert ( END, self.Parameters
           [1] )
```

```

827         self.YRange.grid ( row = 5, column = 1,
828                             sticky = 'we' )
829         self.Step = Entry ( frame, width = 10 )
830         self.Step.insert ( END, self.Parameters[2]
831                             )
832         self.Step.grid ( row = 6, column = 1,
833                         sticky = 'we' )
834
835         label = Label ( frame, text = 'Parameters',
836                         relief = 'groove' )
837         label.grid ( row = 0, column = 2,
838                     columnspan = 2, sticky = 'we' )
839         self.SAL = Label ( frame, text = 'Src-Ap
840                             Distance [nm]', fg = 'grey' )
841         self.SAL.grid ( row = 1, column = 2, sticky
842                         = 'w' )
843         label = Label ( frame, text = 'Wavelength [
844                             nm]' )
845         label.grid ( row = 2, column = 2, sticky =
846                     'w' )
847
848         self.SrAp = Entry ( frame, width = 10 )
849         self.SrAp.insert ( END, self.Parameters[3]
850                             )
851         self.SrAp.configure ( state = 'disabled' )
852         self.SrAp.grid ( row = 1, column = 3 )
853         self.Wavelength = Entry ( frame, width = 10
854                                 )
855         self.Wavelength.insert ( END, self.
856                                 Parameters[4] )
857         self.Wavelength.grid ( row = 2, column = 3
858                                 )
859
860         label = Label ( frame, text = 'Phase Shift'
861                             )
862         label.grid ( row = 3, column = 2, sticky =
863                     'w' )
864         label = Label ( frame, text = 'Aperture
865                             radius [nm]' )
866         label.grid ( row = 4, column = 2, sticky =
867                     'w' )
868         label = Label ( frame, text = 'Displacement
869                             Ap #1 [nm]' )
870         label.grid ( row = 5, column = 2, sticky =
871                     'w' )
872         label = Label ( frame, text = 'Displacement
873                             Ap#2 [nm]' )
874         label.grid ( row = 6, column = 2, sticky =
875                     'w' )
876         label = Label ( frame, text = 'Focal Length
877                             [nm]' )
878         label.grid ( row = 7, column = 2, sticky =
879                     'w' )

```

---

```

858         self.PhSh = Entry ( frame, width = 10 )
859         self.PhSh.insert ( END, self.Parameters[5]
860                             )
860         self.PhSh.grid ( row = 3, column = 3 )
861         self.DiaAp = Entry ( frame, width = 10 )
862         self.DiaAp.insert ( END, self.Parameters[6]
863                             )
863         self.DiaAp.grid ( row = 4, column = 3 )
864         self.DisA1 = Entry ( frame, width = 10 )
865         self.DisA1.insert ( END, self.Parameters[7]
866                             )
866         self.DisA1.grid ( row = 5, column = 3 )
867         self.DisA2 = Entry ( frame, width = 10 )
868         self.DisA2.insert ( END, self.Parameters[8]
869                             )
869         self.DisA2.grid ( row = 6, column = 3 )
870         self.Focus = Entry ( frame, width = 10 )
871         self.Focus.insert ( END, self.Parameters[9]
872                             )
872         self.Focus.grid ( row = 7, column = 3 )
873
874         label = Label ( frame, text = 'Common
875                             Options', relief = 'groove' )
875         label.grid ( row = 11, column = 0,
876                     columnspan = 4, sticky = 'we' )
876         self.legb[0] = Checkbutton ( frame, text =
877                                     'show Legend', variable = self.legend
878                                     [0], command = self.ToggleLeg )
877         self.legb[0].select()
878         self.legb[0].grid ( row = 12, column = 0,
879                             sticky = 'w' )
879         self.legb[1] = Checkbutton ( frame, text =
880                                     'extended Legend', variable = self.
881                                     legend[1] )
880         self.legb[1].grid ( row = 13, column = 0,
881                             sticky = 'w' )
881
882         button = Radiobutton ( frame, text = 'Deutsch', variable = self.lang, value =
883                                     0 )
883         button.grid ( row = 12, column = 1, sticky
884                     = 'w' )
884         button = Radiobutton ( frame, text = 'English', variable = self.lang, value =
885                                     1 )
885         button.grid ( row = 13, column = 1, sticky
886                     = 'w' )
886
887         self.plotLin = Radiobutton ( frame, text =
888                                     'Linear scaling', variable = self.
889                                     scaling, value = 0 )
888         self.plotLin.grid ( row = 12, column = 2,
889                             columnspan = 2, sticky = 'w' )
889         self.plotLog = Radiobutton ( frame, text =
890                                     'Logarithmic scaling', variable = self.

```

```

890         scaling , value = 1 )
891     self.plotLog.grid ( row = 13, column = 2,
892         columnspan = 2, sticky = 'w' )
893     label = Label ( frame, text = 'Linescan
894         Plot Options', relief = 'groove' )
895     label.grid ( row = 15, column = 0,
896         columnspan = 4, sticky = 'we' )
897     self.bfp = Checkbutton ( frame, text = 'BFP
898         ', variable = self.opt[0], command =
899         self.TogglePO )
900     self.bfp.select()
901     self.bfp.configure ( state = 'disabled' )
902     self.bfp.update_idletasks()
903     self.bfp.grid ( row = 16, column = 0,
904         columnspan = 2, sticky = 'w' )
905     self.PPD = Checkbutton ( frame, text = '
906         Phase Difference', variable = self.opt
907         [1], command = self.TogglePO )
908     self.PPD.grid ( row = 17, column = 0,
909         columnspan = 2, sticky = 'w' )
910     self.RefLine1 = Checkbutton ( frame, text =
911         'show  $\pi/2$  reference', variable = self.
912         pi2, state = 'disabled' )
913     self.RefLine1.grid ( row = 16, column = 2,
914         columnspan = 2, sticky = 'w' )
915     self.RefLine2 = Checkbutton ( frame, text =
916         'show  $\pi$  reference', variable = self.pi
917         , state = 'disabled' )
918     self.RefLine2.grid ( row = 17, column = 2,
919         columnspan = 2, sticky = 'w' )
920     self.Multi1 = Radiobutton ( frame, text = '
921         show multiple subplots', variable = self.
922         subplots, value = 1, state = 'disabled'
923         )
924     self.Multi1.grid ( row = 18, column = 2,
925         columnspan = 2, sticky = 'w' )
926     self.Multi2 = Radiobutton ( frame, text = '
927         show multiple figures', variable = self.
928         subplots, value = 2, state = 'disabled'
929         )
930     self.Multi2.grid ( row = 19, column = 2,
931         columnspan = 2, sticky = 'w' )
932     self.Multi3 = Radiobutton ( frame, text = '
933         only one plot', variable = self.subplots
934         , value = 0, state = 'disabled' )
935     self.Multi3.grid ( row = 20, column = 2,
936         columnspan = 2, sticky = 'w' )
937     self.prop = Checkbutton ( frame, text = '
938         Propagation', variable = self.opt[2],
939         command = self.TogglePO )
940     self.prop.grid ( row = 18, column = 0,
941         columnspan = 2, sticky = 'w' )
```

```

914         self.gitter = Checkbutton ( frame, text = '
           Grid Lines', variable = self.opt[3],
           onvalue = True, offvalue = False )
915         self.gitter.grid ( row = 19, column = 0,
           colspan = 2, sticky = 'w' )
916         button = Button ( frame, text = 'BFP
           Linescan', command = self.plot2d )
917         button.grid ( row = 20, column = 0,
           colspan = 2, sticky = 'w' )
918
919         label = Label ( frame, text = 'Further Plot
           Commands', relief = 'groove' )
920         label.grid ( row = 21, column = 0,
           colspan = 4, sticky = 'nws' )
921         self.PropButton = Button ( frame, text = '
           Propagation Plot', command = self.plot3d
           )
922         self.PropButton.grid ( row = 22, column =
           0, sticky = 'w' )
923         self.BFPButton = Button ( frame, text = '
           BFP 3D Plot', command = self.plotbfp )
924         self.BFPButton.grid ( row = 22, column = 1,
           sticky = 'w' )
925
926         status = Frame ( root, relief = 'sunken',
           borderwidth = 1 )
927         status.pack ( side = 'bottom', fill = 'x' )
928         self.statusbar = Label ( status, text = ''
           )
929         self.statusbar.pack ( side = 'left' )
930         button = Button ( status, text = 'Exit',
           command = self.Shutdown )
931         button.pack ( side = 'right' )
932
933 root = Tk( className = ' Wavefield Simulation ' )
934 Interface = GUI ( root )
935 root.mainloop()

```

Listing 2: bruteforce.py

```

1 #!/usr/bin/env python
2
3 from scipy.special import jv, jn_zeros
4 from scipy.integrate.quadrature import quadrature, romb
5 from scipy.integrate.quadpack import quad, dblquad
6 from scipy import fft, ifft, arange, zeros, ones, exp, sin,
   log, concatenate
7 from numpy import array, append, outer, insert, convolve,
   piecewise, where, \
8   fliplr, flipud, sign, sqrt, real, argmin, argmax
9 from math import pi, log
10 from operator import mod
11 from time import sleep
12 import threading
13 import MySQLdb as sql
14

```

```

15 def Wave ( x, y, R, a, l, f, phi ):
16     r = sqrt ( x**2 + y**2 )
17     return jv ( 1, 2 * pi * R * r / l / f ) * exp ( - 1
        j * a * x / l / f - 1j * phi ) / pi / R / r * l
        * f
18
19 def Waves ( x, y, R, a1, a2, l, f, phi ):
20     return Wave ( x, y, R, a1, l, f, 0 ) + Wave ( x, y,
        R, a2, l, f, phi )
21
22 def Phase ( x, R, a1, a2, l, f, phi ):
23     return mod ( 1 * ( a2 - a1 ) * x / l / f + phi + pi
        , 2 * pi ) - pi
24
25 class fCalc ( threading.Thread ):
26     def __init__ ( self, r, d1, d2, phi ):
27         threading.Thread.__init__ ( self )
28         self.r = r
29         self.d1 = d1
30         self.d2 = -d2
31         self.phi = phi
32
33     def integrand0 ( self, x ):
34         return 2 * abs ( mod ( Phase ( x, self.r,
            self.d1, self.d2, l, f, self.phi * pi )
            / pi + .5, 1 ) - .5 ) * sign ( Phase ( x
            , self.r, self.d1, self.d2, l, f, self.
            phi * pi ) / pi ) * abs ( Waves ( x, 0,
            self.r, self.d1, self.d2, l, f, self.phi
            * pi ))**2
35
36     def run ( self ):
37         A = abs ( Waves ( X, 0, self.r, self.d1,
            self.d2, l, f, self.phi * pi ))**2
38         B = Phase ( X, self.r, self.d1, self.d2, l,
            f, self.phi * pi ) / pi
39         D = 2 * abs ( mod ( B + .5, 1 ) - .5 ) *
            sign ( B )
40         ee = romb ( A, step )
41         E = romb ( D * A, step ) / ee
42
43         AM = argmax ( A )
44         A1, A2 = A[:AM], A[AM:]
45         F1 = argmin ( abs ( A1 - .5 * A[AM] ) )
46         F2 = argmin ( abs ( A2 - .5 * A[AM] ) ) +
            AM
47         F = X [ F2 ] - X [ F1 ]
48
49         FF1, FF2 = F1, F2
50         index = int ( log ( F2 - F1 - 1, 2 ) ) + 1
51         while F2 - F1 < 2**index + 1:
52             F2 += 1
53             if F2 - F1 == 2**index + 1:
54                 break
55             F1 -= 1
56
57         A [ F1 : FF1 ] = 0

```

```

58         A [ FF2 : F2 ] = 0
59         G = romb ( A [ F1 : F2 ], step ) / ee
60
61         job = PutPut ( 'INSERT INTO %s'. 'values '
                        'VALUES ( 0, %.3e, %.3e, %.3e, %.4f, %.2f
                        , %.2f, %.2f );\n' % ( db, self.r, self.
                        d1, self.d2, self.phi, 100 * E, F, 100 *
                        G ) )
62         Jobs.append ( job )
63         job.start()
64         return 0
65
66 class PutPut ( threading.Thread ):
67     Handle = threading.Lock()
68
69     def __init__ ( self, something ):
70         threading.Thread.__init__ ( self )
71         self.string = something
72
73     def run ( self ):
74         self.Handle.acquire()
75         Cur.execute ( self.string )
76         self.Handle.release()
77         return 0
78
79 l, f = 4e-3, 1e6
80 xr, step = 2., 1e-4
81 N = int ( log ( xr / step - 1, 2 ) ) + 1
82 Range = .5 * ( 1 + 2**N ) * step
83 X, R = arange ( -Range, Range, step ), arange ( 19e3, 28e3,
            2e3 )
84 Phi = arange ( 0, 1.01, .005 )
85 Jobs = []
86
87 host, user, passw, db = 'localhost', 'andi', 'oaschloch', '
            bruteforce'
88 conn = sql.connect ( host, user, passw, db )
89 Cur = conn.cursor()
90
91 for r in R:
92     D = arange ( r + 5e3, 50e3 - r, ( 45e3 - 2 * r ) /
            25 )
93     for d1 in D:
94         for d2 in D:
95             for phi in Phi:
96                 job = fCalc ( r, d1, d2,
                                phi )
97                 Jobs.append ( job )
98                 while threading.activeCount
99                     () > 100:
100                     sleep ( .01 )
101                 job.start()
102 for job in Jobs:
103     job.join()
104
105 Cur.close()

```

Listing 3: sql1.py

```
1 #!/usr/bin/env python
2
3 from MySQLdb import connect
4 from pylab import plot, subplot, grid, xlabel, ylabel,
    title, show, figlegend, figure, legend
5 from matplotlib.font_manager import FontProperties
6 from matplotlib import rc
7 from numpy import arange, ones, where, any
8 from matplotlib.font_manager import FontProperties
9
10 Cur = connect ( 'localhost', 'andi', 'oaschloch', '
    bruteforce' ).cursor()
11
12 def SQLquery ( radius ):
13     Cur.execute ( ''.join ( ( """SELECT ( 'space1' - '
        space2' ) AS 'sep'
14         FROM 'bruteforce'.'values'
15         WHERE 'radius' = """ , str ( radius ), """
16         GROUP BY 'sep'
17         ORDER BY 'sep'
18         """ ) ) )
19     return Cur.fetchall()
20
21 def SQLquery2 ( radius, sep ):
22     Cur.execute ( ''.join ( ( """SELECT 'signal', '
        phase', 'fwhm' * 1000, 's/n'
23         FROM 'bruteforce'.'values'
24         WHERE
25         'radius' = """ , str ( radius ), """
26
27         AND
28         'space1' - 'space2' = """ , str ( sep
29         ), """ ,
30         ORDER BY 'signal' DESC
31         LIMIT 1
32         """ ) ) )
33     return Cur.fetchall()
34
35 arg = [ 7000, 11000, 17000, 21000, 27000 ]
36
37 Lines1 = SQLquery ( arg[0] );
38 Lines2 = SQLquery ( arg[1] );
39 Lines3 = SQLquery ( arg[2] );
40 Lines4 = SQLquery ( arg[3] );
41 Lines5 = SQLquery ( arg[4] );
42
43 # subplots or figures?
44 # 0: subplots
45 # 1: figures
46 PlotStyle = 1
47 FS = ( 8, 4 )
48
49 L1, L2, L3 = len ( Lines1 ), len ( Lines2 ), len ( Lines3 )
50 L4, L5 = len ( Lines4 ), len ( Lines5 )
51 L1 = len ( Lines1 )
```



```

51 X1 = ones ( L1, 'Float32' )
52 Sig1 = ones ( L1, 'Float32' )
53 Phase1 = ones ( L1, 'Float32' )
54 FWHM1 = ones ( L1, 'Float32' )
55 SN1 = ones ( L1, 'Float32' )
56
57 X2 = ones ( L2, 'Float32' )
58 Sig2 = ones ( L2, 'Float32' )
59 Phase2 = ones ( L2, 'Float32' )
60 FWHM2 = ones ( L2, 'Float32' )
61 SN2 = ones ( L2, 'Float32' )
62
63 X3 = ones ( L3, 'Float32' )
64 Sig3 = ones ( L3, 'Float32' )
65 Phase3 = ones ( L3, 'Float32' )
66 FWHM3 = ones ( L3, 'Float32' )
67 SN3 = ones ( L3, 'Float32' )
68
69 X4 = ones ( L4, 'Float32' )
70 Sig4 = ones ( L4, 'Float32' )
71 Phase4 = ones ( L4, 'Float32' )
72 FWHM4 = ones ( L4, 'Float32' )
73 SN4 = ones ( L1, 'Float32' )
74
75 X5 = ones ( L5, 'Float32' )
76 Sig5 = ones ( L5, 'Float32' )
77 Phase5 = ones ( L5, 'Float32' )
78 FWHM5 = ones ( L5, 'Float32' )
79 SN5 = ones ( L5, 'Float32' )
80
81 print, 'Data Points: ', L1, ' - ', L2, ' - ', L3, ' - ', L4,
    -, L5
82
83 for i in range ( 0, L1 ):
84     X1[i] = Lines1[i][0] * 1e-3
85     data1 = SQLquery2 ( arg[0], Lines1[i][0] )
86     Sig1[i], Phase1[i] = data1[0][0], data1[0][1]
87     FWHM1[i], SN1[i] = data1[0][2], data1[0][3]
88
89 for i in range ( 0, L2 ):
90     X2[i] = Lines2[i][0] * 1e-3
91     data1 = SQLquery2 ( arg[1], Lines2[i][0] )
92     Sig2[i], Phase2[i] = data1[0][0], data1[0][1]
93     FWHM2[i], SN2[i] = data1[0][2], data1[0][3]
94
95 for i in range ( 0, L3 ):
96     X3[i] = Lines3[i][0] * 1e-3
97     data1 = SQLquery2 ( arg[2], Lines3[i][0] )
98     Sig3[i], Phase3[i] = data1[0][0], data1[0][1]
99     FWHM3[i], SN3[i] = data1[0][2], data1[0][3]
100
101 for i in range ( 0, L4 ):
102     X4[i] = Lines4[i][0] * 1e-3
103     data1 = SQLquery2 ( arg[3], Lines4[i][0] )
104     Sig4[i], Phase4[i] = data1[0][0], data1[0][1]

```

```

105         FWHM4[i], SN4[i] = data1[0][2], data1[0][3]
106
107 for i in range ( 0, L5 ):
108     X5[i] = Lines5[i][0] * 1e-3
109     data1 = SQLquery2 ( arg[4], Lines5[i][0] )
110     Sig5[i], Phase5[i] = data1[0][0], data1[0][1]
111     FWHM5[i], SN5[i] = data1[0][2], data1[0][3]
112
113 rc ( 'text', usetex = True )
114 if PlotStyle:
115     figure( figsize = FS )
116 else:
117     figure ( figsize = ( 11.692913, 8.2677165 ) )
118     subplot ( 221 )
119 line1, line2 = plot ( X1, Sig1, 'ro' ), plot ( X2, Sig2, 'b
120 line3, line4 = plot ( X3, Sig3, 'gs' ), plot ( X4, Sig4, '
121 line5 = plot ( X5, Sig5, 'k2' )
122 grid ( True )
123 #xlabel ( 'Blende-Blende-Abstand [ $\mu\text{m}$ ]' )
124 xlabel ( 'Aperture spacing [ $\mu\text{m}$ ]' )
125 #ylabel ( 'Max. Usable Signal [%]' )
126 ylabel ( ' $\frac{U}{U_0}$  [%]' )
127 #title ( '(a) Usable Signal' )
128 legend ( ( line1, line2, line3, line4, line5 ), ( '7  $\mu\text{m}$ 
129     ', '11  $\mu\text{m}$ ', '17  $\mu\text{m}$ ', '21  $\mu\text{m}$ ', '27  $\mu\text{m}$ ', '
130     Aperture radius' ), numpoints = 2, handlelen = 0.01,
131     handletextsep = 0.01, prop = FontProperties ( size = '
132     small' ) )
133
134 if PlotStyle:
135     figure( figsize = FS )
136 else:
137     subplot ( 222 )
138 plot ( X1, Phase1, 'ro', X2, Phase2, 'b^', X3, Phase3, 'gs'
139     , X4, Phase4, 'cd', X5, Phase5, 'k2' )
140 grid ( True )
141 xlabel ( 'Blende-Blende-Abstand [ $\mu\text{m}$ ]' )
142 #xlabel ( 'Aperture spacing [ $\mu\text{m}$ ]' )
143 ylabel ( r 'Phasenverschiebung [ $\pi$ ]' )
144 #ylabel ( 'Phase shift [ $\pi$ ]' )
145 #title ( '(c) Phasenverschiebung' )
146
147 if PlotStyle:
148     figure( figsize = FS )
149 else:
150     subplot ( 223 )
151 plot ( X1, FWHM1, 'ro', X2, FWHM2, 'b^', X3, FWHM3, 'gs',
152     X4, FWHM4, 'cd', X5, FWHM5, 'k2' )
153 grid ( True )
154 xlabel ( 'Blende-Blende-Abstand [ $\mu\text{m}$ ]' )
155 #xlabel ( 'Aperture spacing [ $\mu\text{m}$ ]' )
156 ylabel ( 'FWHM [pm]' )

```

---

```

152 #title ( '(c) Halbwertsbreite' )
153
154
155 if PlotStyle:
156     figure( figsize = FS )
157 else:
158     subplot ( 224 )
159 plot ( X1, SN1, 'ro ', X2, SN2, 'b^', X3, SN3, 'gs ', X4, SN4
        , 'cd ', X5, SN5, 'k2' )
160 grid ( True )
161 #xlabel ( 'Blende-Blende-Abstand [ $\mu\text{m}$ ]' )
162 xlabel ( 'Aperture spacing [ $\mu\text{m}$ ]' )
163 ylabel ( 'Signal / Noise [%]' )
164 #title ( '(d) Signal / Rausch - Verhaeltnis' )
165
166 show()

```

Listing 4: sql2.py

```

1 #!/usr/bin/env python
2
3 from MySQLdb import connect
4 from pylab import plot, subplot, grid, xlabel, ylabel,
    title, show, figlegend, figure, legend
5 from matplotlib.font_manager import FontProperties
6 from matplotlib import rc
7 from numpy import arange, ones, where, any
8 from matplotlib.font_manager import FontProperties
9
10 Cur = connect ( 'localhost', 'andi', 'oaschloch', '
    bruteforce' ).cursor()
11
12 def SQLquery ( radius, sep ):
13     Cur.execute ( ''.join ( ( """SELECT
14         ( 'space1' - 'space2' ) AS 'sep',
15         ( 'space1' + 'space2' ) AS 'shift '
16         FROM 'bruteforce'.'.values '
17         WHERE
18         'radius' = """ , str ( radius ), """ ,
19         AND
20         'space1' - 'space2' = """ , str ( sep ), """
21
22         GROUP BY 'shift '
23         ORDER BY 'shift '
24     """ ) ) )
25     return Cur.fetchall()
26
27 def SQLquery2 ( radius, sep, shift ):
28     Cur.execute ( ''.join ( ( """SELECT 'signal', '
29         phase', 'fwhm' * 1000, 's/n'
30         FROM 'bruteforce'.'.values '
31         WHERE
32         'radius' = """ , str ( radius ), """
33
34         AND
35         'space1' - 'space2' = """ , str (
36             sep ), """ ,

```

```

33                                     AND
34                                     'space1' + 'space2' = ''", str (
                                     shift ), "",
35                                     ORDER BY 'signal' DESC
36                                     LIMIT 1
37                                     "" ) ) )
38     return Cur.fetchall()
39
40 arg = [ 9000 ]
41 arg2 = [ 37720, 64720 ]
42
43 Lines1 = SQLQuery ( arg[0], arg2[0] );
44 Lines2 = SQLQuery ( arg[0], arg2[1] );
45
46 # subplots or figures?
47 # 0: subplots
48 # 1: figures
49 PlotStyle = 1
50 FS = ( 8, 4 )
51
52 L1, L2 = len ( Lines1 ), len ( Lines2 )
53
54 X1 = ones ( L1, 'Float32' )
55 Sig1 = ones ( L1, 'Float32' )
56 Phase1 = ones ( L1, 'Float32' )
57 FWHM1 = ones ( L1, 'Float32' )
58 SN1 = ones ( L1, 'Float32' )
59
60 X2 = ones ( L2, 'Float32' )
61 Sig2 = ones ( L2, 'Float32' )
62 Phase2 = ones ( L2, 'Float32' )
63 FWHM2 = ones ( L2, 'Float32' )
64 SN2 = ones ( L2, 'Float32' )
65
66 print 'Data Points: ', L1, ' - ', L2
67
68 for i in range ( 0, L1 ):
69     X1[i] = Lines1[i][1] * 1e-3
70     data1 = SQLQuery2 ( arg[0], arg2[0], Lines1[i][1] )
71     Sig1[i], Phase1[i] = data1[0][0], data1[0][1]
72     FWHM1[i], SN1[i] = data1[0][2], data1[0][3]
73
74 for i in range ( 0, L2 ):
75     X2[i] = Lines2[i][1] * 1e-3
76     data1 = SQLQuery2 ( arg[0], arg2[1], Lines2[i][1] )
77     Sig2[i], Phase2[i] = data1[0][0], data1[0][1]
78     FWHM2[i], SN2[i] = data1[0][2], data1[0][3]
79
80 rc ( 'text', usetex = True )
81 if PlotStyle:
82     figure( figsize = FS )
83 else:
84     figure ( figsize = ( 11.692913, 8.2677165 ) )
85     subplot ( 221 )
86 line1, line2 = plot ( X1, Sig1, 'ro' ), plot ( X2, Sig2, 'b
    ^, )

```

```

87 grid ( True )
88 #xlabel ( 'Blende-Blende-Abstand [ $\mu\text{m}$ ]' )
89 xlabel ( 'Aperture spacing [ $\mu\text{m}$ ]' )
90 #ylabel ( 'Max. Usable Signal [%]' )
91 ylabel ( ' $\mathit{U}$  [%]' )
92 #title ( '(a) Usable Signal' )
93 legend ( ( line1, line2 ), ( '37.72  $\mu\text{m}$ ', '64.72  $\mu\text{m}$ ',
    , 'Aperture spacing \nat 9 $\mu\text{m}$  radius' ), numpoints =
    2, handlelen = 0.01, handletextsep = 0.01, prop =
    FontProperties ( size = 'small' ) )
94
95 if PlotStyle:
96     figure( figsize = FS )
97 else:
98     subplot ( 222 )
99 plot ( X1, Phase1, 'ro', X2, Phase2, 'b^' )
100 grid ( True )
101 xlabel ( 'Blende-Blende-Abstand [ $\mu\text{m}$ ]' )
102 #xlabel ( 'Aperture spacing [ $\mu\text{m}$ ]' )
103 ylabel ( 'r Phasenverschiebung [ $\pi$ ]' )
104 #ylabel ( 'Phase shift [ $\pi$ ]' )
105 #title ( '(c) Phasenverschiebung' )
106
107
108 if PlotStyle:
109     figure( figsize = FS )
110 else:
111     subplot ( 223 )
112 plot ( X1, FWHM1, 'ro', X2, FWHM2, 'b^' )
113 grid ( True )
114 xlabel ( 'Blende-Blende-Abstand [ $\mu\text{m}$ ]' )
115 #xlabel ( 'Aperture spacing [ $\mu\text{m}$ ]' )
116 ylabel ( 'FWHM [pm]' )
117 #title ( '(c) Halbwertsbreite' )
118
119
120 if PlotStyle:
121     figure( figsize = FS )
122 else:
123     subplot ( 224 )
124 plot ( X1, SN1, 'ro', X2, SN2, 'b^' )
125 grid ( True )
126 #xlabel ( 'Blende-Blende-Abstand [ $\mu\text{m}$ ]' )
127 xlabel ( 'Aperture spacing [ $\mu\text{m}$ ]' )
128 ylabel ( 'Signal / Noise [%]' )
129 #title ( '(d) Signal / Rausch - Verhaeltnis' )
130
131 show()

```

## Abbildungsverzeichnis

1	XMCD- und EMCD-Spektrum . . . . .	8
2	Elektronen-Energieverlust-Spektrum . . . . .	13
3	EELS-Spektrometer . . . . .	15
4	Rasterelektronenmikroskop . . . . .	16
5	Zum Reziprozitätsprinzip . . . . .	18
6	Rastertransmissionselektronenmikroskop . . . . .	19
7	Kristallstrukturen . . . . .	21
8	Zirkularer Dichroismus . . . . .	23
9	Chirale Moleküle . . . . .	23
10	Analogie Elektronen $\leftrightarrow$ Photonen . . . . .	25
11	Experimenteller Aufbau für EMCD . . . . .	28
12	Schematischer Aufbau für ein EMCD-Experiment . . . . .	34
13	Perspektivische Darstellung: Doppellochblende . . . . .	35
14	Linsenabbildung . . . . .	37
15	Huygens'sches Prinzip . . . . .	38
16	Unterschiedliche Koordinaten . . . . .	39
17	Explosionsdarstellung: Phasenplatte . . . . .	41
18	Potentialbarriere . . . . .	42
19	Potential einer Einzellinse . . . . .	44
20	Ringelektrode . . . . .	45
21	Elektrostatisches Potential einer Ringelektrode . . . . .	47
22	Linseneffekt einer Ringelektrode . . . . .	48
23	Abhängigkeit der Phasenverschiebung . . . . .	51
24	Wellenfunktionen . . . . .	53
25	Bessel- und Airy-Funktion . . . . .	54
26	Gewichtsfunktion . . . . .	56
27	Intensitätsverteilungen in BFP . . . . .	58
28	Intensitätsverteilung auf Achse in BFP . . . . .	59
29	Propagation: ebene Welle . . . . .	60
30	Auswertung mit sql1.py . . . . .	61
31	Auswertung mit sql1.py . . . . .	62
32	Symmetrie der Blendenanordnung . . . . .	64
33	Auswertung mit sql2.py . . . . .	65
34	Auswertung mit sql2.py . . . . .	65
35	Blendenabstand / Blende-Blende-Abstand . . . . .	66
36	Alternative Darstellung . . . . .	68
37	CM30: Erwartetes Beugungsmuster . . . . .	70
38	CM30: Beobachtetes Beugungsbild . . . . .	70
39	SiN Mesh . . . . .	71
40	F30: Airy-Scheibe in Beugungsbild . . . . .	72
41	Blende im Lichtmikroskop . . . . .	73
42	Zwei Ringelektroden . . . . .	74

43	PMMA als Negativlack: schematisch . . . . .	75
44	PMMA als Negativlack: im Mikroskop . . . . .	76
45	Goldpartikel im STEM . . . . .	78
46	Goldpartikel im STEM . . . . .	79
47	Blenden . . . . .	81
48	Detektoren . . . . .	83
49	STEM Bilder: Platingefüllte Nanoröhrchen . . . . .	84
50	Beispiel: Fouriertransformation . . . . .	86
51	Beispiel: Faltung . . . . .	87
52	Screenshot . . . . .	92
53	Screenshots . . . . .	94

## Literatur

- [AHA<sup>+</sup>08] D. Alloyeau, W. K. Hsieh, E. H. Anderson, G. Benner, M. J. Park, E. D. Gomez, N. P. Balsara, and C. Kisielowski. Contrast transfer function design by an electrostatic phase plate for contrast enhancement in transmission electron microscopy. In 14<sup>th</sup> *European Microscopy Congress – Late Breaking Posters*, number P800, 2008.
- [Boe47] H. Boersch. Über die kontraste von atomen im elektronenmikroskop. *Z. Naturforsch.*, 2a:615ff, 1947.
- [BR70] H. G. Badde and L. Reimer. Der einfluss einer streuenden phasenplatte auf das elektronenmikroskopische bild. *Z. Naturforsch.*, 25a:760ff, 1970.
- [Bri] Encyclopaedia Britannica. Britannica online encyclopedia.
- [Bro24] L. V. De Broglie. *Recherches sur la Theorie des Quanta*. PhD thesis, Sorbonne, 1924.
- [BS04] Bergmann and Schaefer, editors. *Optik – Wellen- und Teilchenoptik*. de Gruyter, 2004.
- [CH62] H. Castaing and L. Henry. Filtrage magnetique des vitesses en microscopie electronique. *C.R. Acad. Sci., Paris*, B255:76–78, 1962.
- [CIJ69] A. Crewe, M. Isaacson, and D. Johnson. A simple scanning electron microscope. *Rev. Sci. Inst.*, 40:241–246, 1969.
- [Cow69] J. M. Cowley. Image contrast in a transmission scanning electron microscope. *Appl. Phys. Lett.*, 15(2):58, 1969.

- [CSMM90] C. T. Chen, F. Sette, Y. Ma, and S. Modesti. Soft-x-ray magnetic circular dichroism at the  $l_{2,3}$  edges of nickel. *Phys. Rev. B*, 42:7262–7265, 1990.
- [CWL70] A. Crewe, J. Wall, and J. Langmore. Visibility of a single atom. *Science*, 168(3937):1338–1340, 1970.
- [EB08] C. J. Edgcombe and J. A. C. Bland. A phase plate for electron microscopy using the aharonov-bohm effect. In *14<sup>th</sup> European Microscopy Congress – Late Breaking Posters*, number P802, 2008.
- [EFH<sup>+</sup>06] B. Eienkela, P. Formanek, C. Hèbert, C. Hurm, H. Lichte, S. Rubino, P. Schattschneider, and J. Zweck. Instrumental developments for the chiraltem project. The 16<sup>th</sup> International Microscopy Congress, 2006.
- [ES75] J. L. Erskine and E. A. Stern. Calculation of the  $m_{2,3}$  magneto-optical absorption spectrum of ferromagnetic nickel. *Phys. Rev. B*, 12:5016–5024, 1975.
- [Goo04] J. W. Goodman. *Introduction to Fourier Optics*. Roberts and Company Publishers, 3rd edition, 2004.
- [HB44] J. Hillier and R. F. Baker. Microanalysis by means of electrons. *Journal of Applied Physics*, 15:663, 1944.
- [Hil43] J. Hillier. On microanalysis by electrons. *Phys. Rev.*, 64:318, 1943.
- [Hit93] A. P. Hitchcock. Near edge electron energy loss spectroscopy: Comparison to x-ray absorption. *Jpn. J. Appl. Phys*, 32(2):176, 1993.
- [HS03] C. Hébert and P. Schattschneider. A proposal for dichroic experiments in the electron microscope. *Ultramicroscopy*, 96(3-4):463–468, 2003.
- [HSR<sup>+</sup>08] C. Hébert, P. Schattschneider, S. Rubino, P. Novák, J. Ruzs, and M. Stöger-Pollach. Magnetic curcular dichroism in electron energy loss spectrometry. *Ultramicroscopy*, 108:277–284, 2008.
- [Hur08] C. Hurm. *Electron Magnetic Chiral Dichroism in a Transmission Electron Microscope*. PhD thesis, U Regensburg, 2008.
- [HWB97] A. C. F. Hoole, M. E. Welland, and A. N. Broers. Negative pmma as a high-resolution resist – the limits and possibilities. *Semicond. Sci. Technol.*, 12(9), 1997.



## LITERATUR

---

- [Inc07] Gatan Inc. Post-column tem energy filters – model 850 gif 2002 (<http://www.gatan.com>), 2007.
- [KH03] G. Kothleitner and F. Hofer. Eels performance measurements on a new high energy resolution imaging filter. *Micron*, 34:211–218, 2003.
- [KMN<sup>+</sup>00] F. Krumeich, H.-J. Muhr, M. Niederberger, F. Bieri, and R. Nesper. The cross-sectional structure of vanadium oxide nanotubes studied by transmission electron microscopy and electron spectroscopic imaging. *Z. Anorg. Allg. Chem.*, 626:2208–2216, 2000.
- [Kno35] M. Knoll. Aufladepotential und sekundäremission elektronenbestrahlter körper. *Z. tech. Phys.*, 16:467–475, 1935.
- [KR32] M. Knoll and E. Ruska. Das elektronenmikroskop. *Zeitschrift für Physik*, 78:318–339, 1932.
- [KR85] H. Kohl and H. Rose. Theory of image formation by inelastically scattered electrons in the electron microscope. *Advances in electronics and electron physics*, 65, 1985.
- [Kru] F. Krumeich. Electron microscopy (<http://www.microscopy.ethz.ch>).
- [Lan06] H. P. Langtangen. *Python Scripting for Computational Science*. Springer, 2nd edition, 2006.
- [Leo07] T. Leong. Dirty dice. In *2007 MRS FALL MEETING "SCIENCE AS ART" IMAGES*. Material Research Society (<http://www.mrs.org>), 2007.
- [MBS<sup>+</sup>07] E. Majorovits, B. Barton, K. Schultheiß, F. Perez-Willard, D. Gerthsen, and R. R. Schröder. Optimizing phase contrast in transmission electron microscopy with an electrostatic (boersch) phase plate. *Ultramicroscopy*, 107:213–226, 2007.
- [MD56] G. Möllenstedt and H. Düker. Beobachtungen und messungen an biprisma-interferenzen mit elektronenwellen. *Zeits. für Physik A*, 145(3):377–397, 1956.
- [Mey] Meyers lexikon online 2.0 (<http://lexikon.meyers.de>).
- [MHR<sup>+</sup>08] J. P. Morniroli, F. Houdellier, C. Roucau, J. Puiggali, S. Gesti, and A. Redjaimia. Lacdif, a new electron diffraction technique obtained with the laced configuration and a  $c_s$  corrector: Comparison with electron precession. *Ultramicroscopy*, 108(2):100–115, January 2008.

- [MOT98] T. Matsumoto, N. Osakabe, and A. Tonomura. *US Patent 5814815 – Phase-contrast electron microscope and phase plate therefor*, 1998.
- [MPG07] C. Remenyi Max-Planck-Gesellschaft. Max wissen (<http://www.max-wissen.de>), 2007.
- [MSAE] S. A. Mueller, M. Steinmetz, U. Aebi, and Andreas Engel. Imaging and measuring biomolecules and their assemblies by scanning transmission electron microscopy (<http://www.mih.unibas.ch/booklet/booklet96/>).
- [MT96] T. Matsumoto and A. Tonomura. The phase constancy of electron waves traveling through boersch’s electrostatic phase plate. *Ultramicroscopy*, 63:5–10, 1996.
- [NEE] Institut NEEL. X-ray magnetic circular dichroism (<http://neel.cnrs.fr>).
- [NLS<sup>+</sup>99] M. Nelhiebel, P.-H. Louf, P. Schattschneider, P. Blaha, K. Schwarz, and B. Jouffrey. Theory of orientation-sensitive near-edge fine-structure core-level spectroscopy. *Phys. Rev. B*, 59(20):12807–12814, May 1999.
- [NSJ00] M. Nelhiebel, P. Schattschneider, and B. Jouffrey. Observation of ionization in a crystal interferometer. *Phys. Rev. Lett.*, 85(9):1847–1850, Aug 2000.
- [PJ72] D. F. Parsons and H. M. Johnson. Possibility of a phase contrast electron microscope. *Appl. Opt.*, 11(12):2840–2843, 1972.
- [Rei84] L. Reimer. *Transmission Electron Microscopy*. Springer, 1984.
- [RhLL<sup>+</sup>08] J. Rusz, h. Lidbaum, A. Liebig, P. Novák, P. M. Oppeneer, O. Eriksson, K. Leifer, and B. Hjörvarsson. Role of asymmetries for emcd sum rules. In *14<sup>th</sup> European Microscopy Congress*, volume 1, pages 423–424, 2008.
- [RP77] L. Reimer and G. Pfefferkorn. *Rasterelektronenmikroskopie*. Springer, 2 edition, 1977.
- [RRS07] J. Rusz, S. Rubino, and P. Schattschneider. First-principles theory of chiral dichroism in electron microscopy applied to 3d ferromagnets. *Physical Review Letters*, B 75(214425), 2007.
- [Rub07] S. Rubino. *Magnetic Circular Dichroism in the Transmission Electron Microscope*. PhD thesis, TU Wien, 2007.

- [SBRB07] R. R. Schröder, B. Barton, H. Rose, and G. Benner. Contrast enhancement by anamorphic phase plates in an aberration corrected tem. *Micros. Microanalysis*, 13:8–9, 2007.
- [SBS<sup>+</sup>08] R. R. Schröder, B. Barton, K. Schultheiß, B. Gamm, and D. Gerthsen. In-focus phase contrast: Present state and future developments. In 14<sup>th</sup> *European Microscopy Congress*, pages 69–70, 2008.
- [Sch86] P. Schattschneider. *Fundamentals of Inelastic Electron Scattering*. Springer, 1986.
- [Sch02] F. Schwabl. *Quantenmechanik*. Springer, 6 edition, 2002.
- [SiN] Electron microscopy sciences (<http://www.emsdiasum.com>).
- [SNSJ00] P. Schattschneider, M. Nelhiebel, H. Souchay, and B. Jouffrey. The physical significance of the mixed dynamic form factor. *Micron*, 31(4):333–345, 2000.
- [Spe03] J. C. H. Spence. *High-Resolution Electron Microscopy*. Oxford University Press, 3 edition, 2003.
- [SRH<sup>+</sup>06] P. Schattschneider, S. Rubino, C. Hébert, J. Rusz, J. Kunes, P. Novák, E. Carlino, M. Fabrizioli, G. Panaccione, and G. Rossi. Detection of magnetic circular dichroism using a transmission electron microscope. *Nature*, 441(7092):486–488, May 2006.
- [SW05] P. Schattschneider and W. S. M. Werner. Coherence in electron energy loss spectrometry. *Journal of Electron Spectroscopy and Related Phenomena*, 143:83–97, 2005.
- [SWW<sup>+</sup>87] G. Schütz, W. Wagner, W. Wilhelm, P. Kienle, R. Zeller, R. Frahm, and G. Materlik. Absorption of circularly polarized x rays in iron. *Phys. Rev. Lett.*, 58(7):737–740, Feb 1987.
- [SZ82] F. W. Sears and M. W. Zemansky. *University Physics 6. Ed.* Addison-Wesley, 1982.
- [TvdLS85] B. T. Thole, G. van der Laan, and G. A. Sawatzky. Strong magnetic dichroism predicted in the  $m_{4,5}$  x-ray absorption spectra of magnetic rare-earth materials. *Phys. Rev. Lett.*, 55:2086–2088, 1985.
- [vA38a] M. v. Ardenne. Das elektronen-rastermikroskop. praktische ausführung. *Z. tech. Phys.*, 19:407–416, 1938.
- [vA38b] M. v. Ardenne. Das elektronen-rastermikroskop. theoretische grundlagen. *Z. Phys.*, 109:553–572, 1938.

- [VHR<sup>+</sup>08a] J. Verbeeck, C. Hébert, S. Rubino, P. Novák, J. Rusz, F. Houdellier, C. Gatel, and P. Schattschneider. Optimal aperture sizes and positions for emcd experiments. *Ultramicroscopy*, 108(9):865–872, 2008.
- [VHR<sup>+</sup>08b] J. Verbeeck, C. Hébert, S. Rubino, P. Novák, J. Rusz, F. Houdellier, C. Gatel, and P. Schattschneider. Optimal aperture sizes and positions for emcd experiments. In 14<sup>th</sup> *European Microscopy Congress*, volume 1, pages 453–454, 2008.
- [VR39] H. Voges and G. Ruthemann. Steigerung des auflösungsvermögens der magnetischen halbkreis-methode zur geschwindigkeitsanalyse von kathodenstrahlen. *Zeits. für Physik*, 114(11-12):709–718, 1939.
- [WC96] D. B. Williams and C. B. Carter. *Transmission electron microscopy: A Textbook for Materials Science*. Springer, 1996.
- [Wik] Wikimedia commons (<http://commons.wikimedia.org>).
- [YM97] J. Yuan and N. K. Menon. Magnetic linear dichroism in electron energy loss spectroscopy. *Journal of Applied Physics*, 81:5087, 1997.

## Danksagung

Abschließend möchte ich mich bedanken

bei JOSEF ZWECK für die Vergabe dieses interessanten Themas, die fachliche Hilfe und das Korrekturlesen dieser Arbeit,

bei CHRISTIAN DIETRICH für die Hilfe bei der Herstellung der Doppellochblende & bei den ersten Schritten zur Realisierung einer Phasenplatte und das Korrekturlesen dieser Arbeit,

bei CHRISTIAN HURM für die Unterstützung bei der Aufbereitung der Theorie und das Korrekturlesen dieser Arbeit.

Ebenfalls möchte ich mich bei der Arbeitsgruppe von Josef Zweck und dem gesamten Lehrstuhl für die gute Atmosphäre bedanken.

---

# Erklärung

Hiermit versichere ich, dass ich diese Diplomarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Die Stellen meiner Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen sind, habe ich in jedem Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht. Dasselbe gilt für Tabellen und Abbildungen. Diese Arbeit hat in dieser oder einer ähnlichen Form noch nicht im Rahmen einer anderen Prüfung vorgelegen.

Regensburg, 30.09.2008

---

(Ort, Datum)

---

(Unterschrift)