



Arthur Fuchs, BSc

Experimental determination of dislocation densities in martensitic steels

MASTER'S THESIS

to achieve the university degree of
Diplom-Ingenieur

Master's degree programme: Advanced Materials Science

submitted to

Graz University of Technology

Supervisor

Univ.-Prof. Dipl.-Ing. Dr.techn. Bernhard Sonderegger

Institute of Materials Science, Joining and Forming

AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis dissertation.

Date

Signature

Acknowledgements

First of all, I would like to thank my supervisor Univ.-Prof. Dipl.-Ing. Dr.techn. Bernhard Sonderegger for his guidance. I really appreciate giving me my space but immediately being there when I needed directions.

In addition, I would like to thank all my colleagues from the IMAT for their support and their friendly attitude.

Furthermore, great appreciation is directed towards the XRD group of the Institute of Solid State Physics (TU Graz), especially Ao.Univ.-Prof. Dipl.-Ing. Dr.techn. Roland Resel, for the considerable amount of measurement time granted and for teaching me how to operate the X-ray diffractometer in the first place.

Special thanks go to the Institute of Materials Physics (TU Graz), especially Dipl.-Ing. BSc Robert Enzinger, for doing the dilatometer measurements and redoing them under argon atmosphere, as soon as we realized that evaporation of Cr is a problem under high vacuum.

I would also like to thank my family, especially my parents and my girlfriend, for their great support. And finally to all my friends: Thanks to those of you who studied with me for exams, those of you who prevented me from studying when I overdid it and those who did both!

Thank you very much, everyone!

Arthur Fuchs

Graz, March 2018.

1 Abstract

This work deals with the measurement of dislocation densities in complex martensitic alloys. In these materials, dislocation motion is responsible for plastic deformation and thus determines properties such as yield strength or creep resistance. Up to now, there has been no physically justified standard procedure to determine this property with good statistical significance and reasonable experimental effort. X-Ray diffraction (XRD) potentially is such a method, combining low experimental cost with sufficient sample volume. Nevertheless, the complexity of the microstructure interferes with a straightforward application of standard methods usually applied in other, less complex, materials.

In this work, conventional XRD-methods are adapted to a martensitic 9% Cr steel (P91). The basic physical concept is simple: An (ideally) monochromatic and parallel X-ray beam interacts with the crystal structure of the sample, the diffraction angle(s) Θ is a measure of the interatomic distance(s) according to Bragg's law. The broadening of the diffraction peak results from (i) properties of the experimental equipment and (ii) the microstructure of the sample. (i) is referred to as instrumental broadening and includes properties of the beam, the slits and the lenses, (ii) comprises any effect leading to local deviations from the ideal lattice constant, which is mostly dislocations and inner boundaries of any type.

For the measurement, a PANalytical Empyrean XRD with Cu K_α radiation was used. The samples were a NIST LaB₆ standard and seven P91 samples (austenitized (1), austenitized and heat treated (2-7)).

In a first step, the instrumental broadening was characterized by measurement of the LaB₆ standard. With this reference, the data from the steel samples could be de-convoluted into sample- and instrumental broadening. Finally, the sample broadening was split into microstrains and broadening due to coherent domain size. For this procedure, some established methods like "Williamson-Hall", "modified Williamson-Hall", "Warren-Averbach" and "double-Voigt" are compared qualitatively to each other. Furthermore, the frequently used "modified Warren Averbach" method and the most common mistake with this method is discussed in detail. A workaround, by introducing the contrast factor of dislocations into the double-Voigt method, is presented. In the modified Williamson-Hall and modified Warren-Averbach methods, the dislocation density is calculated assuming, that all microstrain-inducing defects except dislocations are negligible.

Exemplarily, the results are compared to findings from high-precision dilatometry tests and the results differ by less than a factor of two.

Contents

1	Abstract	II
2	Parameter List	V
3	Introduction	1
3.1	Task	1
4	Basics	2
4.1	General information about X-ray diffraction	2
4.1.1	Generation of X-rays	2
4.1.2	Bragg's law	3
4.1.3	Setup of an X-ray diffractometer	4
4.1.4	Peak characteristics	5
4.1.5	Peak-shapes in XRD-measurements	6
4.2	Instrumental broadening correction	7
4.2.1	Cauchy-/Gauss deconvolution	7
4.2.2	Voigt deconvolution	7
4.2.3	Stokes deconvolution	8
4.3	General information about size-/strain analysis	8
4.4	Williamson-Hall method	8
4.5	Warren-Averbach method	10
4.6	Modified Williamson-Hall and modified Warren-Averbach approach	12
4.7	Groma's method	12
4.8	Double-Voigt	13
4.9	Dilatometry	13
5	Experimental	14
5.1	Mathematical considerations for data processing	14
5.1.1	Combination of modified Williamson-Hall and modified Warren-Averbach	14
5.1.2	Combination of modified Warren-Averbach and Double-Voigt	16
5.2	Sample preparation	19
5.3	X-ray diffraction	19
5.4	Dilatometry	20
5.5	Data processing	20
5.5.1	Matlab programs for data processing	21
6	Results	25
6.1	X-ray diffraction	25
6.2	Dilatometry	30
7	Data Analysis and Discussion	31
7.1	Fitting of XRD peaks	31
7.2	Separation of size- and strainbroadening	34
7.2.1	Williamson-Hall method (also see chapter 4.4):	34
7.2.2	Modified Williamson-Hall technique (also see chapter 4.6):	35
7.2.3	Warren-Averbach method (also see chapter 4.5):	36
7.3	Modified Warren-Averbach method (also see chapter 4.6):	37
7.3.1	Double-Voigt method (also see chapter 4.8):	37

7.4	Calculation of dislocation densities	38
7.4.1	Calculation of dislocation densities according to the contrast-corrected double-Voigt method	39
7.5	Dilatometry	41
7.6	Comparison of results	42
8	Conclusion	44
9	Appendix	49
9.1	XRD measurement specifications	49
9.2	Determining γ in the modified Warren-Averbach method	53
9.3	Plots of data fits	53
9.3.1	LaB ₆ standard	53
9.3.2	Steel samples	60
9.4	Fit results	62
9.5	Determination of q	72
9.6	Matlab codes	73
9.6.1	Peak fitting	73
9.6.2	pseudoVoigtAsym_C_G_separate	73
9.6.3	pseudoVoigtAsym_separate	76
9.6.4	LaB ₆ _peak_fit	78
9.6.5	Steel_peak_fit	88
9.6.6	LaB ₆ _exp_fit	97
9.6.7	mWH_mWA	100

2 Parameter List

Table 1: Parameter List

Parameter	Meaning	Definition in:
α	Parameter: $\alpha = \left(\frac{K}{D}\right)^2$	eq. 34a
β	Integral breadth. Subscripts provide more information	eq. 2
β_S	Integral breadth due to crystallite size	eq. 13
β_D	Integral breadth due to microstrains	eq. 14
β_{sample}	Total integral breadth due to crystallite size and microstrains	eq. 8
$\beta_{instrument}$	Integral breadth due to instrumental broadening	eq. 8
$\beta_{observed}$	Integral breadth observed when measuring the strain-broadened sample	eq. 8
γ	Parameter: $\gamma = \frac{\pi M b^2 \rho}{2}$	eq. 34b
$\langle \epsilon^2(L) \rangle$	Mean-square-strain	eq. 23
η	Microstrains: $\eta = \frac{\Delta d}{d}$	eq. 14a
Θ	Diffraction angle	fig. 3
Θ_0	Bragg angle of K_α reflection maximum	eq. 3
λ	Used X-ray wavelength	eq. 1
ρ	Dislocation density	eq. 25
A	Real part of the curves' Fourier series	eq. 20
A_n^D	Distortion coefficient of A	eq. 20
A_n^S	Size coefficient of A	eq. 20
B	Parameter: $B = \frac{\pi b^2}{2}$	eq. 27a
C	Average dislocation contrast factor	eq. 25
C_{h00}	Dislocation contrast factor for $(h00)$ -planes	eq. 33
$C(2\Theta)$	Cauchy function	eq. 3
D	Crystallite size	eq. 13
F	Constant for dislocations interactions; in this work $F = 1$	eq. 45
$f(x), F(x)$	Pure-specimen (physically) broadened profile and its Fourier transform	eq. 7
$g(x), G(x)$	Instrumentally broadened profile and its Fourier transform	eq. 7
$G(2\Theta)$	Gauss function	eq. 4
$h(x), H(x)$	Observed broadened profile and its Fourier transform	eq. 7
H	Parameter: $H^2 = \frac{h^2 k^2 + h^2 l^2 + k^2 l^2}{h^2 + k^2 + l^2}$	eq. 33a
K	Shape factor; can be assumed as ≈ 1	eq. 13
L	Fourier length $L = n \cdot a$	eq. 21
M and M'	Parameters depending on the effective outer cutoff radius of dislocations	eq. 25
$P_{VII}(2\Theta)$	Pearson VII function	eq. 6
Q	A correlation factor	eq. 25
R_e	Outer cut-off radius for dislocations: $R_e \approx \frac{1}{2\rho}$	eq. 27
V and ΔV	Volume and volume change of dilatometry sample	eq. 31
$V(2\Theta)$	Voigt function	eq. 5
W	Full-width-at-half-maximum	sec. 4.1.4
X	Parameter: $X = \frac{(\Delta g)^2 - \alpha}{g^2}$	eq. 36
Y	Parameter: $Y = \rho B L^2 \ln\left(\frac{R_e}{L}\right)$	eq. 39

a	Unit of Fourier length in direction of the diffraction vector	eq. 21
g	Burgers Vector	eq. 25
d	Interatomic distance	fig. 3
g	Variable in reciprocal space: $g = \frac{2\sin(\Theta)}{\lambda} = \frac{1}{d}$	eq. 15a and 25
Δg	Parameter: $\Delta g = \frac{\beta_{sample} 2\cos(\Theta)}{\lambda}$	eq. 25
g_0	Some constant	eq. 28
h, k, l	Miller indices	
j	Constant that depends on the lattice; $j = 14.4$ for bcc	eq. 45
k	Slope for a linear $y = kx + d$ function	
l and Δl	Length and length change of dilatometry sample	eq. 31
m	Shape factor for Pearson VII function	eq. 6
n	Some integer	
q	Parameter depending on elastic constants and dislocation character	eq. 33
v_{disl}	Specific excess free volume of dislocations	eq. 32a
x	General variable, usually either $x = \Theta$ or $x = g$	eq. 7
\star	Convolution: $g(x) \star f(x) = \int g(z)f(x-z) dz$	
Subscripts		
f	Denotes the pure-specimen (physically) broadened profile	
$f(\eta)$	Wilkins function	eq. 26
g	Denotes the instrumentally broadened profile	
h	Denotes the observed profile (instrumental and physical broadened)	
C	Denotes the Cauchy component of the Voigt function	
D	Denotes the distortion relation of a parameter	
G	Denotes the Gaussian component of the Voigt function	
S	Denotes the crystallite size relation of a parameter	

3 Introduction

In most metals, plastic deformation is determined by the movement of dislocations and properties like yield strength and creep behaviour depend on them. For martensitic 9 % Cr steels (e.g. P91), especially the creep behaviour is crucial. Until now mostly TEM, EBSD and X-ray diffraction were used to determine dislocation densities. TEM has the disadvantages of very high costs combined with very bad statistical evidence and EBSD measures geometrically necessary- and wall-dislocations only. Thus, only X-ray diffraction combines moderate experimental costs with sufficient volumes for good statistics. This is why many papers already used this technique (e.g. [1] [2]).

The basics of dislocation density measurements via XRD go back to the well known Scherrer formula from 1918 ([3]): Scherrer noted that not only the position of an XRD peak is important (Bragg's law), but its broadness is influenced by the material's microstructure. He then denotes all peak broadening to the size of coherently diffracting domains. In 1944, Stokes and Wilson noted that peak broadening can be also caused by microstrains and developed a formula to quantify this "apparent strain" [4]. Later, Williamson and Hall combined these two approaches in their Williamson-Hall method to separate size- and strainbroadening [5]. This method is still very popular nowadays, even though its limitation to rough estimations has been stated multiple times (e.g. [6]). Later, the more fundamental but also more time-consuming Warren-Averbach approach [7] was published, which is also still used nowadays. The latest milestone for dislocation density measurements was done by T. Ungár at around 2000: He assumed all microstrains other than those of dislocations are negligible and thus published his modified Williamson-Hall- and modified Warren-Averbach approach [8]. Since that time, many authors used this techniques to evaluate dislocation densities and coherent domain sizes, often neglecting limitations and uncertainties. A critical review about using integral breath methods carelessly, especially the modified Williamson-Hall technique, is done in [9].

Nowadays there are two general approaches to apply the methods mentioned above to XRD data:

- **Rietveld, Whole-profile-fitting,...:** The first of this methods, the "Rietveld refinement" [10] was first introduced in the late 60s. In general, these methods share the same basic principle: All phenomena that change peak properties have to be described in mathematical models. Then, the whole profile measured is fitted according to these models. The differences between Rietveld, Whole-profile-fitting,... are solely the included models. These are very powerful techniques in general, but the operator needs to be very experienced because one wrong input afterwards may be "corrected" by other wrong results and controlling these programs is difficult ("black boxes").
- **Single peak fitting:** In these methods, every peak is fitted separately and the mathematical models describing the microstructures are applied afterwards. This has many advantages when testing different data processing models, because the CPU-intensive fitting procedure is done *a priori*. Furthermore, controlling the data analysis routines step-by-step is much easier. In this work, only single peak fitting methods are used.

3.1 Task

Aim of this work was to evaluate the capabilities and limitations of dislocation density measurements with X-ray diffraction on martensitic 9 % Cr steels (P91). To do so, methods to extract the instrumental broadening such as Cauchy-, Gauss-, Voigt- and Stokes deconvolution were tested and the most popular methods of data processing were evaluated: Williamson-Hall-, Warren-Averbach-, double-Voigt-, modified Williamson-Hall- and m. Warren-Averbach method. Groma's peak tail analysis [11] could not be used, as the measured data did not meet the required quality for this method.

To check the reliability of the XRD methods, dilatometry as an alternative method was also tested.

4 Basics

4.1 General information about X-ray diffraction

4.1.1 Generation of X-rays

The generation of X-rays is well-described in many books, e.g. [12] and [13]. Nevertheless, because many problems of this work are due to its specifics, a short overview is given in the following.

There are various ways of producing X-rays. The most important ones are synchrotron radiation, braking radiation ("Bremsstrahlung") and characteristic radiation. For this thesis, only braking radiation and characteristic radiation are important. The origin for both of them can be described by the bombardment of a metallic target with electrons:

- Braking radiation: The electrons are rapidly slowed down when they hit the target, which includes a loss in energy. Most of this energy is transferred into heat, but some of it is dispersed in form of radiation. This radiation is called "braking radiation" or "Bremsstrahlung". Its emission spectrum is continuous and the intensity increases with increasing acceleration voltage. At the same time, the minimum wavelength λ_{min} decreases with increasing acceleration voltage.
- Characteristic radiation: If an element is bombarded with high energy particles, in this case electrons, there is the possibility that a bound electron from an inner shell is excited and an electron from an outer shell takes its place. As the electron falls to the inner shell, it emits an X-ray with a very specific wavelength, corresponding to the energy gap between the two electron shells. This radiation can have several wavelengths, but its spectrum is discontinuous. Even though the intensities of these X-rays are usually much higher than the intensities of the braking radiation, there still is a contribution of the braking radiation. This results in a continuous background radiation.

A typical spectrum of a molybdenum anode is shown in fig. 1. The continuous background corresponds to braking radiation, the K_α and K_β peaks correspond to characteristic molybdenum radiation.

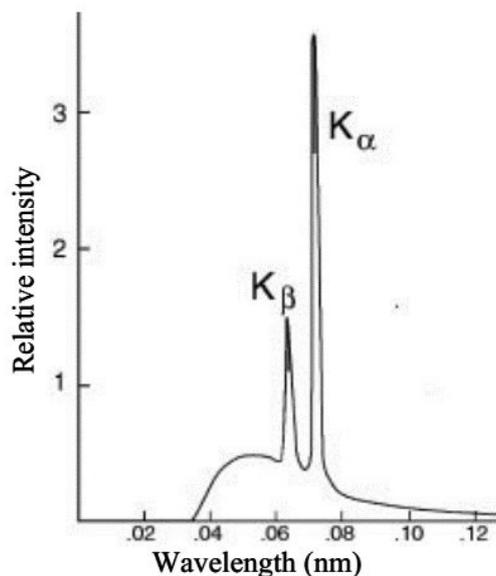


Figure 1: Example for the spectrum of a molybdenum anode. The continuous part corresponds to the braking radiation, the two sharp peaks are characteristic radiation. [Fig.1]

The setup for a typical X-ray tube (in this case a "Coolidge tube") is shown in fig. 2 .

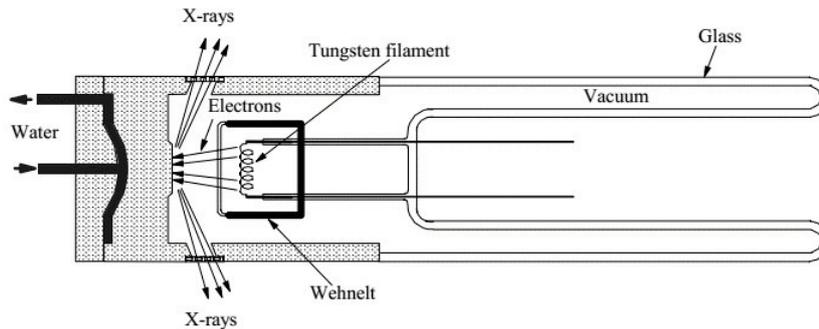


Figure 2: Setup for a "Coolidge tube" as typical example for an X-ray source. [Fig.2]

In most cases, the filament is made of tungsten. This together with the geometry shown in fig. 2 can explain two major problems when performing X-ray diffraction (XRD).

1. The source is not perfectly dot-like. This together with not perfect slits, lenses, monochromators,... lead to the so called "instrumental broadening".¹
2. Tungsten from the filament evaporates and condenses again, especially at the anode and walls of the tube. This deposition on the anode leads to absorption of some of the emitted X-rays and to the presence of parasitic X-rays.²

4.1.2 Bragg's law

When the X-rays from the tube get reflected by a material (sample), they first interact with this sample, then with each other (=interference). This can be described by Bragg's law: [15]

$$2d \sin(\Theta) = n\lambda \quad (1)$$

with the inter-atomic distance d , the X-ray wavelength λ and a positive integer n . Note that perfectly narrow peaks only occur for defect-free (includes not distorted!) crystals (no changes in the inter-atomic distance) A visualization of Bragg's law can be seen in fig. 3:

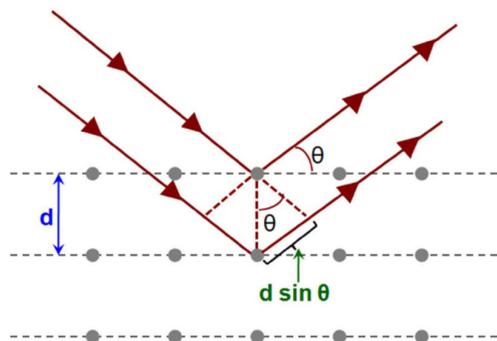


Figure 3: Bragg's law: Constructive interference is only possible, if wave maxima of both waves are at the same position. This is the case, when the path difference (= $2d \sin(\Theta)$) equals to $n\lambda$. [Fig.3]

¹A detailed evaluation of instrumental broadening factors is done in chapter 3.1 of [14]

²In general, all not coherent with the main source and not wanted X-rays are called "parasitic"

4.1.3 Setup of an X-ray diffractometer

The general setup of an X-ray diffractometer is shown in fig. 4 .

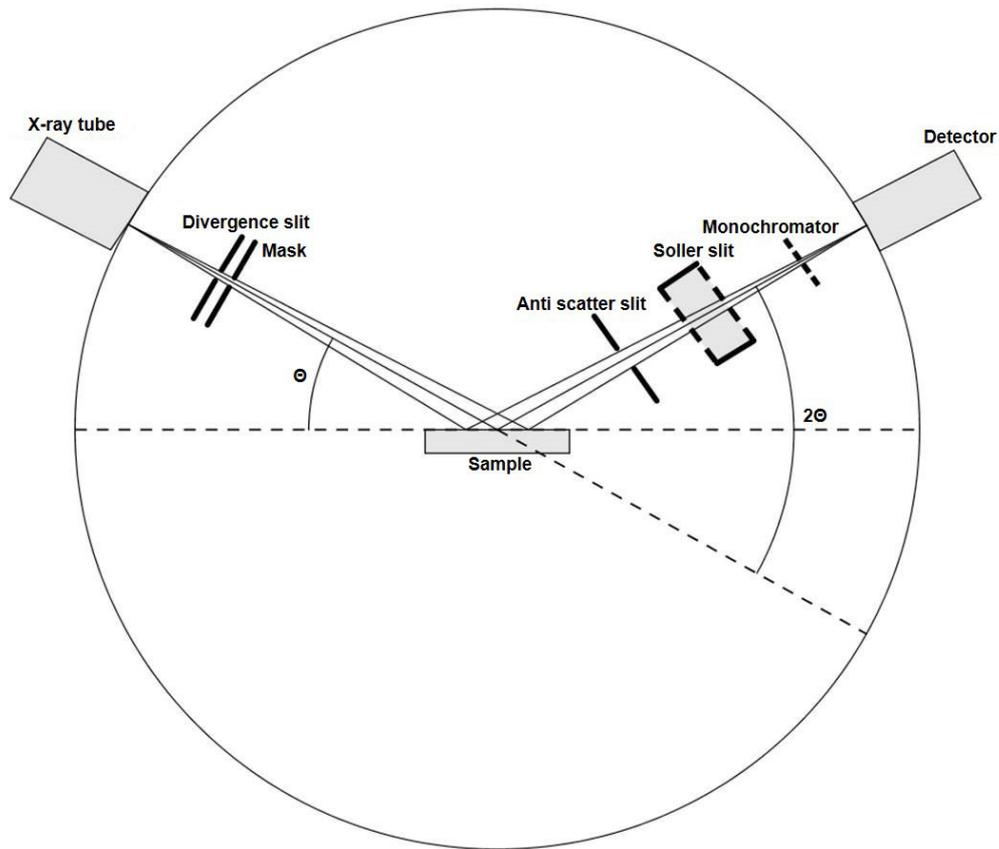


Figure 4: General setup of an X-ray diffractometer.

Divergence slit... limits horizontal divergence

Mask... guarantees, that the irradiated area is smaller than the sample

Anti-scatter-slit... filters stray light

Soller-slit... prevents vertical beam divergence

Monochromator... guarantees, that only selected wavelengths reach the detector

First, the X-rays have to pass a divergence slit, that limits horizontal beam divergence. Limiting the beam divergence too much results in very weak intensities. After the divergence slit, a mask guarantees that only the sample is irradiated, even at smaller 2θ angles. After the sample, an anti-scatter-slit filters stray light and a soller-slit prevents vertical beam divergence (it is quite common to have another soller-slit directly after the X-ray tube, before the beam passes the divergence slit). Finally, a monochromator suppresses all wavelengths except the K_{α} radiation, which also reduces background radiation originating within the sample. [12] [13]

A typical XRD-measurement detects angle-dependent diffracted intensities. There are several measurement methods that could be used described in [12]. Fig. 4 shows a normal $\theta/2\theta$ -measurement as used in this work.

More detailed descriptions on basics of X-ray diffraction can be found in books, e.g. [16].

4.1.4 Peak characteristics

As described in the next section, there are several ways to model peak shapes. To nevertheless be able to model universally applicable mathematical methods, some unique peak features were specified:

Half-width-at-half-maximum:

The half-width-at-half-maximum (HWHM) gives a good description of the peak broadness. One advantage is, that the method is applicable for any fixed, symmetric random peak shape. One disadvantage is that not the full information of the peak is used. This makes it inaccurate for comparisons of asymmetric and differently shaped peaks (e.g. if the Cauchy/Gauss ratio of Pearson VII or Voigt peaks change). In some cases, the peak broadness is described by the full-width-at-half-maximum (FWHM), that is two times the half-width-at-half-maximum. Both full- and half-width-at-half-maximum are used for rough estimations and quick data analysis.

Fig. 5 graphically shows the half-width-at-half-maximum:

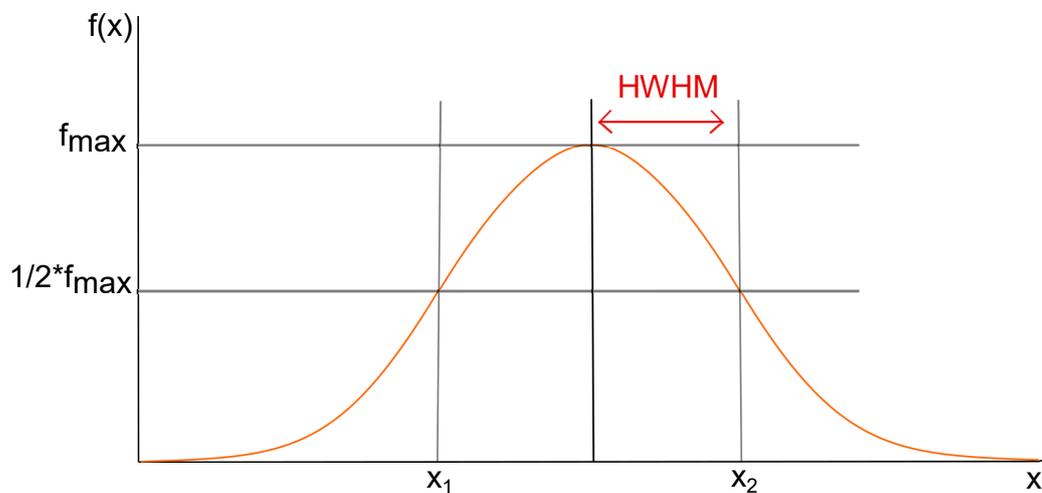


Figure 5: Graphical description of the half-width-at-half-maximum (HWHM)

Integral breadth:

For more precise analysis, the integral breadth (IB) should be used. In the following, the integral breadth is referred to as β and it is defined in eq. 2:

$$\beta = \frac{\text{Peak Area}}{\text{Peak height}} \quad (2)$$

As with the FWHM or the HWHM, the integral breadth method can be applied to any peak shape. If the background level is evaluated carefully, it is much more robust to comparing differently shaped and asymmetric peaks than FWHM/HWHM.

4.1.5 Peak-shapes in XRD-measurements

The peak shape of an XRD $\Theta/2\Theta$ -measurement does not fit into a single scheme and can only be determined empirically. For rough estimations, Cauchy³ (eq. 3) or Gaussian shapes (eq. 4) can be used. Usually, neither of these describes the measured peak accurately. Mostly acknowledged functions are the Voigt-function (eq. 5) and the Pearson VII function (eq. 6). [14]

Cauchy function: [14]

$$C(2\Theta) = \frac{2}{\pi W} \left[1 + \frac{4(\sqrt{2}-1)}{W^2} (2\Theta - 2\Theta_0)^2 \right]^{-1} \quad (3)$$

with the full-width-at-half-maximum (FWHM) W and the peak maximum (usually of $K_{\alpha 1}$) Θ_0 .

Gauss function: [14]

$$G(2\Theta) = \frac{2}{W} \sqrt{\frac{\ln(2)}{\pi}} \exp \left[\frac{-4 \ln(2)}{W^2} (2\Theta - 2\Theta_0)^2 \right] \quad (4)$$

Voigt function (is a convolution of a Cauchy- and a Gauss function): [14]

$$V(2\Theta) = \beta_G^{-1} \operatorname{Re} \left[\operatorname{erfc} \left(\frac{\sqrt{\pi}}{\beta_G} |2\Theta - 2\Theta_0| + i \frac{\beta_C}{\beta_G \sqrt{\pi}} \right) \right] \quad (5)$$

where:

- Re: real part of a complex function
- erfc: complex error function
- β_C and β_G : integral breadths of the Gaussian and Lorentzian components
- i : imaginary number

Since convolutions are quite CPU-intensive, most programs use the "pseudo-Voigt" function, which is a linear combination of a Cauchy- and a Gauss function⁴, instead of the Voigt function.

Pearson VII function: [14]

$$P_{VII}(2\Theta) = \frac{2\sqrt{m}\Gamma(2\frac{1}{m}-1)}{W\pi\Gamma(m-0,5)} \left[1 + \frac{4(2\frac{1}{m}-1)}{W^2} (2\Theta - 2\Theta_0)^2 \right]^{-m} \quad (6)$$

where m is a shape factor ($m < 1$ super-Lorentzian; $m = 1$ Lorentzian and $m \gg 1$ Gaussian) and Γ is the Gamma function.

³Lorentzian and Cauchy function are the same

⁴"pseudo-Voigt", because the linear combination is an approximation for the Voigt function's convolution

4.2 Instrumental broadening correction

In the last section, peak shapes were discussed. By fitting one of them to the measured data, the discontinuous sets of data are described by a continuous function. The next step in data analysis is to decompose the peak into different phenomena that contribute to its position, broadness and intensity (in this work, only broadness will play a major role). Note that the peak broadening is dependent on the diffraction angle 2θ .

In general the observed peak is a convolution of the instrumentally broadened peak and the sample broadened peak. [17]

$$h(x) = g(x) \star f(x) \quad (7)$$

with the observed broadened profile $h(x)$, the instrumentally broadened profile $g(x)$ and the physically broadened sample profile $f(x)$.

To determine the instrumental broadening only, there are two techniques used frequently:

- Simulation of the instrument with all slits and lenses used.
- Measurement of an external standard without sample broadening, e.g. some NIST standard ⁵. For reasons of convenience, this is the technique used most.

Depending on the data processing technique used, there are different methods to perform this instrumental-/sample broadening separation, the most important ones described in the following.

4.2.1 Cauchy-/Gauss deconvolution

If the measured XRD peaks can be fitted with either Cauchy- or Gauss peaks, the instrumental broadening can be extracted with eq. 8 and eq. 9:

$$\text{Cauchy: } \beta_{\text{sample}} = \beta_{\text{observed}} - \beta_{\text{instrument}} \quad (8)$$

$$\text{Gauss: } \beta_{\text{sample}}^2 = \beta_{\text{observed}}^2 - \beta_{\text{instrument}}^2 \quad (9)$$

with the integral breadth observed when measuring the strain-broadened sample β_{observed} , the integral breadth due to crystallite size and microstrains β_{sample} and the integral breadth due to instrumental broadening $\beta_{\text{instrument}}$.

Since peak shapes usually are neither pure Cauchy nor pure Gaussian, the fit quality suffers. Widely acknowledged peak shapes would be the Voigt profile (convolution of Cauchy and Gauss peaks), the pseudo-Voigt profile (the convolution is approximated by a linear combination of Cauchy and Gauss peaks) and the Pearson VII profile (see sec. 4.1.5). The deconvolution of Voigt- (and their approximation, the pseudo-Voigt) functions is discussed in the next chapter.

4.2.2 Voigt deconvolution

In the so called double-Voigt approach, both instrument and observed peaks are modelled via Voigt- or pseudo-Voigt functions. Since the convolution of two Voigt functions is also a Voigt function, the Cauchy-/Gaussian parts can be deconvoluted separately and the instrumental broadening can be extracted easily: [18]

$$\beta_{fC} = \beta_{hC} - \beta_{gC} \quad (10)$$

$$\beta_{fG}^2 = \beta_{hG}^2 - \beta_{gG}^2 \quad (11)$$

⁵NIST: National Institute of Standards and Technology. Commonly used standards for XRD applications are NIST SRM 660 (LaB₆) and NIST SRM 640 (Si).

with the Cauchy and Gaussian parts of the pure-specimen broadened integral breadths β_{fC} and β_{fG} , of the observed integral breadths β_{hC} and β_{hG} and of the instrumentally broadened integral breadths β_{gC} and β_{gG} .

4.2.3 Stokes deconvolution

Another deconvolution method is the method of Stokes [19]. From eq. 7 follows, that the deconvolution can be done quite easily in fourier-space: [19]

$$F(z) = \frac{H(z)}{G(z)} \quad (12)$$

$F(z)$, $H(z)$ and $G(z)$ are the Fourier transforms of $f(x)$, $h(x)$ and $g(x)$ (see eq. 7).

Transforming $F(z)$ back into real space gives the pure physically broadened profile $f(x)$. The big advantage of this method is, that no assumption concerning peak shape have been done so far.

After the pure sample-broadened peak is separated, the size-/strain analysis can be done.

4.3 General information about size-/strain analysis

Most formulas in diffraction theory (like Bragg's- and Laue's law) were developed for perfect mono-crystals. In real crystals, the perfect grid of atoms is tilted and bent through imperfections (e.g. clusters of dislocations and twins). This divides the material in many small volumes, where Bragg's law can be applied, which are called "coherent domains". The domain size is usually called "crystallite size", which is often confused with grain- or subgrain size. These two groups of microstructural features are not identical and should not be mixed up. There are however special cases like defect free nano powders (grain size) and fully recovered fine grained materials (subgrain size), where they are equal to each other.

The topic in size-/strain analysis is how to separate the pure specimen-broadened peak into size-effects (coherent domains) and microstrains.

4.4 Williamson-Hall method

Since the Williamson-Hall method assumes Cauchy- or Gaussian peak shapes, usually the method described in sec. 4.2.1 is used for separation of the instrumental broadening.

A general approach to calculate the crystallite size is the Scherrer-formula: [3]

$$\beta_S(\Theta) = \frac{K \cdot \lambda}{D \cdot \cos(\Theta)} \quad (13)$$

with the integral breadth due to crystallite size $\beta_S(\Theta)$, the X-ray wavelength λ , the crystallite size D and the diffraction angle Θ . The shape factor K can be assumed as ≈ 1 .

The microstrains can be described by the Stokes-Wilson-formula: [4]

$$\beta_D(\Theta) = \eta \cdot \tan(\Theta) \quad (14)$$

with the integral breadth due to microstrains $\beta_D(\Theta)$ and the microstrains

$$\eta = \frac{\Delta d}{d} \quad (14a)$$

in which d is the inter-atomic distance.

With

$$g = \frac{2\sin(\Theta)}{\lambda} \quad (15a)$$

the integral breadth in real space ($\beta(\Theta)$) can be transformed to reciprocal space ($\beta(g)$):

$$\beta(g) = \frac{\beta(\Theta)\cos(\Theta_0)}{\lambda} \quad (15b)$$

With the Bragg angle of K_α reflection maximum Θ_0 .

In general the easiest way to differentiate between size and strain broadening is that strain broadening in reciprocal space is dependent on the diffraction angle Θ and size broadening is not.

This implies different Θ dependences of size- and strain broadening in real space. The exact dependence of the strain broadening on Θ varies with the evaluation model one chooses. [18] Common models are: 1) The "apparent" strain defined by Stokes and Wilson [4] (which corresponds to a maximum strain) that is used in all simplified integral breadth methods e.g. the Williamson-Hall approach. 2) The mean-square strain defined by Warren [20] and used in the Warren-Averbach approach.

G.K. Williamson and W.H. Hall assumed, that either the integral breadths or their squares sum up to the measured peak:

$$\beta_{sample}^n(\Theta) = \beta_D^n(\Theta) + \beta_S^n(\Theta) \quad (16)$$

where $n=1$ for pure Cauchy and $n=2$ for pure Gaussian shapes. With this assumption the authors were able to combine eq. 13 and eq. 14⁶. Using $n = 1$ (Cauchy only!), eq. 17 is achieved: [5]

$$\beta_{sample}(\Theta) = \beta_D(\Theta) + \beta_S(\Theta) = \eta \cdot \tan(\Theta) + \frac{K \cdot \lambda}{D \cdot \cos(\Theta)} \quad (17)$$

By multiplying eq. 17 with $\cos(\Theta)$ one gets eq. 18 [5].

Note that this equation is only valid for pure Lorentzian (=Cauchy) peak shapes and that the size part of this equation is not dependent on Θ .

$$\beta_{sample}(\Theta) \cdot \cos(\Theta) = \eta \cdot \sin(\Theta) + \frac{K \cdot \lambda}{D} \quad (18)$$

By plotting $\beta_{sample}(\Theta) \cdot \cos(\Theta)$ vs. $\sin(\Theta)$, the system can be treated as a simple linear equation

$$y = kx + \bar{d} \quad (18a)$$

with

$$k = \eta \quad (18b)$$

$$\bar{d} = \frac{K \cdot \lambda}{D} \quad (18c)$$

By doing so, the crystallite size and microstrains can be fitted conveniently, as seen in fig. 6.

⁶Nowadays, also mixed types, especially $\beta_S(\Theta) \rightarrow cauchy$ and $\beta_D(\Theta) \rightarrow gauss$ are popular.

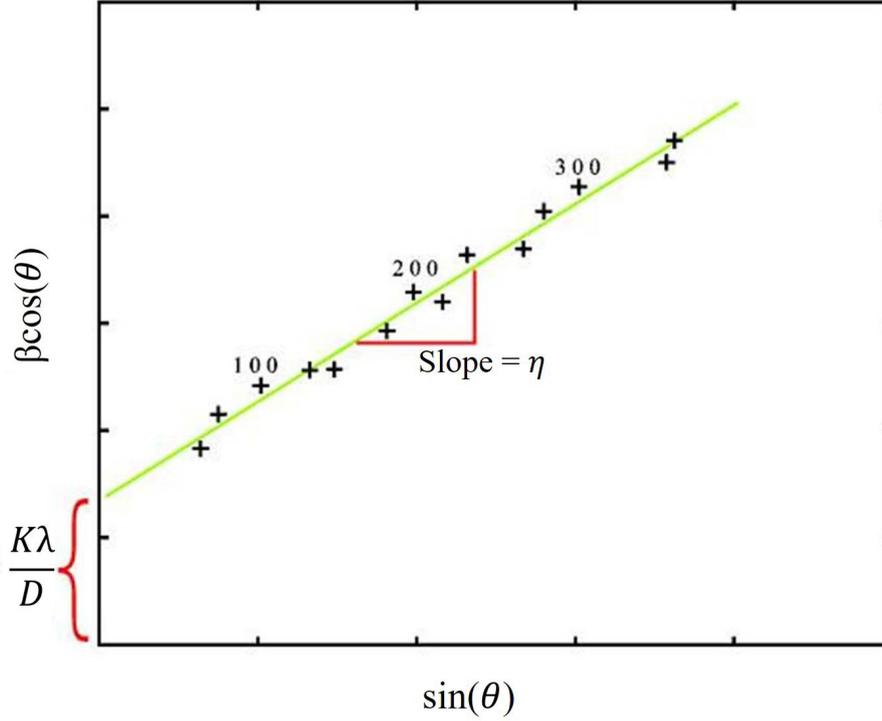


Figure 6: Example for a Williamson-Hall plot with $\beta = \beta_{sample}(\Theta)$ [Fig.6].

4.5 Warren-Averbach method

For the Warren-Averbach method, either the Stokes deconvolution described in sec. 4.2.3 (recommended frequently) or, for example if peak overlapping occurs, the Voigt deconvolution described in sec. 4.2.2 is used.

The following derivation (from eq. 20 to eq. 24) only contains the essential steps and can be looked up step by step in the original papers of B.E. Warren and B.L. Averbach [21] [22] [7].

The idea behind this method is the following: Since peak broadening is only due to two phenomena (size- and microstrain effects), the measured peak ($f(2\Theta)$) is a convolution of the size-broadened peak ($f_S(2\Theta)$) and the strain-broadened peak ($f_D(2\Theta)$):

$$f(2\Theta) = f_S(2\Theta) \star f_D(2\Theta) \quad (19)$$

Solving this problem in reciprocal space makes things much easier since the convolution (\star) reduces to a simple multiplication (see eq. 20). This transformation into Fourier space is done by expressing the measured intensities ($f(2\Theta)$) as Fourier series (a sum of cosine and sine components). According to the theory, the sine components of these Fourier sums are negligible and only the real parts of the Fourier series (the cosine components) contribute to the result. These real parts of the Fourier series (A_n) then are a simple multiplication of the size contributions (A_n^S) and the microstrain (=distortion) contributions (A_n^D) with the harmonic number n :

$$A_n = A_n^S \cdot A_n^D \quad (20)$$

This drastically reduces computational time for deconvolution compared to conventional Fourier transformation.

The authors further introduce a , which is the unit of Fourier length in direction of the diffraction vector g and the Fourier length L . One way to interpret this is: As written in chapter 4.3, in diffraction

theory materials can be considered consisting of coherent diffracting domains. According to Warren and Averbach, each of these domains again is considered consisting of columns of cells along a specific direction. This grid then has the length a and the distance between cell pairs (L) is defined in eq. 21:

$$L = n \cdot a \quad (21)$$

With this, $A_n = A(n)$ can be converted to $A(L)$ and eq. 22 is achieved:

$$\ln [A(L)] = \ln [A^S(L)] + \ln [A^D(L)] = \ln [A^S(L)] + \ln \left\{ \langle \exp [2\pi i g L \epsilon(L)] \rangle \right\} \quad (22)$$

with the microstrain $\epsilon(L)$.

For not too large L and a Gaussian strain distribution the following approximation can be made:

$$\langle \exp [2\pi i g L \epsilon(L)] \rangle \approx \exp \left[-2\pi^2 g^2 L^2 \langle \epsilon^2(L) \rangle \right] \quad (23)$$

Where $\langle \epsilon^2(L) \rangle$ is the so called "mean-square-strain" ⁷. It is the square of the non-uniform strain averaged over the distance L .

Inserting eq. 23 into eq. 22, the seminal Warren-Averbach relation is achieved: [21] [22] [7]

$$\ln [A(L)] = \ln [A^S(L)] - 2\pi^2 g^2 L^2 \langle \epsilon^2(L) \rangle \quad (24)$$

By plotting $\ln [A(L)]$ vs. the square of the diffraction vector, g^2 , one gets the Warren-Averbach plot shown in fig. 7.

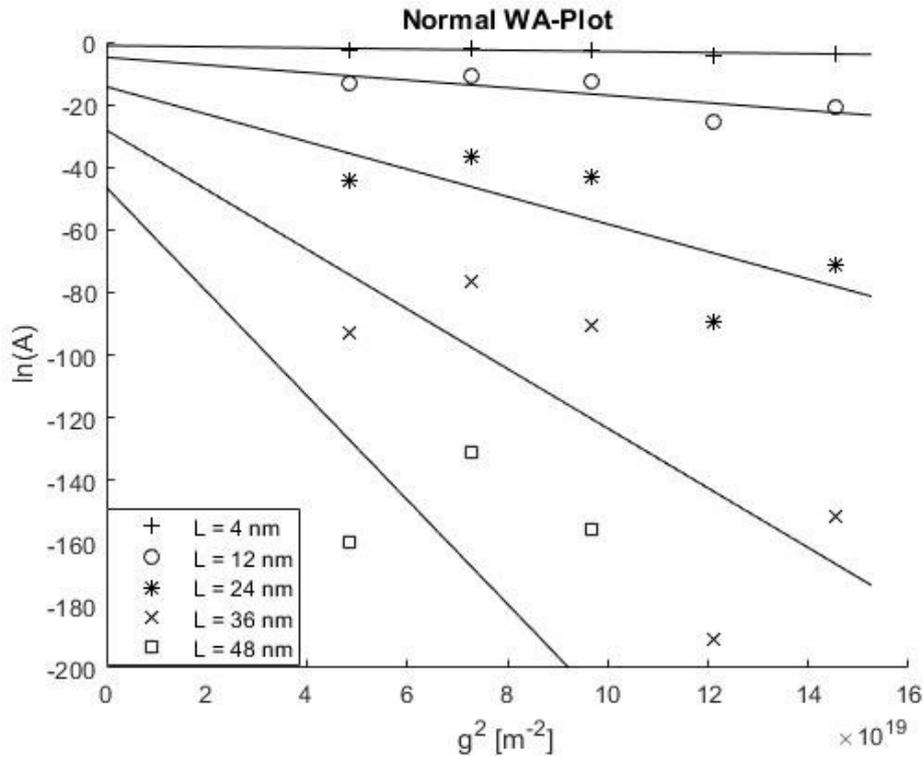


Figure 7: Example for a Warren-Averbach plot. The intercepts with the y-axis yield the size coefficients and the slopes the mean-square strains $\langle \epsilon^2(L) \rangle$ (compare with eq. 24).

⁷also referred to as e_{mss}

4.6 Modified Williamson-Hall and modified Warren-Averbach approach

T. Ungár noted, that most of the deviations from a straight line in the Williamson-Hall plot can be due to simple phenomena [8]. For example considering strain broadening solely due to dislocations, a contrast factor for dislocations (C) should be included to account for differences in intensity. Furthermore, a logarithmic series expansion of the Fourier coefficients should be done. Doing this, the modified Williamson-Hall equation is given in eq. 25: [8]

$$\Delta g \cong \frac{K}{D} + \sqrt{\frac{\pi M b^2}{2}} \cdot \rho \cdot g \sqrt{C} + \frac{\pi M' b^2}{2} \cdot \sqrt{Q} \cdot g^2 C \quad (25)$$

With shape factor K (≈ 1), the crystallite size D , the Burgers Vector b , the dislocation density ρ , a correlation factor Q and the average dislocation contrast factor C (see eq. 33). For a detailed description of the dislocation contrast factor see [8] at p. 5. M and M' are parameters depending on the effective outer cutoff radius of dislocations (R_e) and the dislocation density (for determination of dislocation densities, it is not necessary to evaluate these values). Parts of eq. 18 were merged to

$$\Delta g = \frac{\beta_{sample}(\Theta) 2 \cos(\Theta)}{\lambda} \quad (25a)$$

$$g = \frac{2 \sin(\Theta)}{\lambda} \quad (25b)$$

T. Ungár also applied this approach to the Warren-Averbach equation. For the microstrains, eq. 26 was used:

$$\langle \epsilon^2(L) \rangle = \left(\frac{b}{2\pi} \right)^2 \pi \rho f(\eta) C \quad (26)$$

where $f(\eta)$ is the Wilkens function. Approximating the Wilkens function in a series expansion, truncating it after the quadratic term and inserting eq. 26 into the Warren-Averbach equation (eq. 24), the modified Warren-Averbach equation (eq. 27) is achieved: [8]

$$\ln [A(L)] \cong \ln [A^S(L)] - \rho B L^2 \ln \left(\frac{R_e}{L} \right) (g^2 C) + Q B^2 L^4 \ln \left(\frac{R_2}{L} \right) \ln \left(\frac{R_3}{L} \right) (g^2 C)^2 \quad (27)$$

where

$$B = \frac{\pi b^2}{2} \quad (27a)$$

Note: Since the contrast factor of dislocations C accounts for differences in intensities for different crystallographic planes, the corresponding X-ray pattern looks exactly like a pattern with macrotexture. This can be fatal for all programs that fit the whole pattern at once and do not have the contrast factor of dislocations included (e.g. Rietveld programs). In this case, the pattern will never fit unless a (not existing) texture is included, subsequently leading to wrong results even though the fit quality can be mathematically good.

Compared to methods where all peaks are fitted separately, e.g. the Warren-Averbach method: Not including C results in very big scattering of data points, leading to very big errors that immediately indicate a problem.

4.7 Groma's method

I. Groma introduced a method to analyze the peak tails of single peaks. This method requires a very good peak-to-background ratio and well resolved peak tails. Since this requirements cannot be fulfilled in this work, this method is not used. A detailed description of the method is given in [11].

4.8 Double-Voigt

In the Double-Voigt approach, D. Balzar suggests to model the diffraction peaks as Voigt functions. After extracting the instrumental broadening (see section 4.2.2), the user can choose if the results should correspond to a simplified integral breadth method ("apparent" strain) or to Fourier methods (mean-square strain). [18]

In this work, the results will be compared with the results of the modified Warren-Averbach method. Thus, the following relations to calculate the integral breadth parts for the mean-square strain are used: [18]

$$\beta_C(g) = \beta_{CS}(g) + \beta_{CD}(g) \frac{g^2}{g_0^2} \quad (28)$$

$$\beta_G^2(g) = \beta_{GS}^2(g) + \beta_{GD}^2(g) \frac{g^2}{g_0^2} \quad (29)$$

With the reciprocal space variable g and $\beta_{CS}(g)$, $\beta_{CD}(g)$, $\beta_{GS}(g)$, $\beta_{GD}(g)$ and g_0 are constants.

The mean-square-strain is given by eq. 30 and can be evaluated by inserting the integral breadths from eq. 28 and eq. 29: [18]

$$\langle \epsilon^2(L) \rangle = \frac{1}{g^2} \left[\frac{\beta_{GD}^2(g)}{2\pi} + \frac{\beta_{CD}(g)}{\pi} \frac{1}{L} \right] \quad (30)$$

4.9 Dilatometry

In dilatometry, length changes due to pre-defined temperature programs are measured. Some special dilatometers measure volumes instead of lengths, but mostly the length changes have to be converted into volumes: [23]

$$\frac{\Delta V}{V} \approx \frac{3\Delta l}{l} \quad (31)$$

where ΔV and Δl are volume- and length changes and V and l are the total sample volume and length respectively.

The measured volume/length changes can be due to many phenomena like thermal expansion, phase transitions, grain growth and healing of crystal defects. For this work, especially the volume change due to recovery of dislocations given in eq. 32 is important. [23]

$$\left(\frac{\Delta V}{V} \right)_{disl} = v_{disl} \cdot \rho \quad (32)$$

with the dislocation density ρ and the specific excess free volume of dislocations [24]

$$v_{disl} = 0.5 \cdot b^2 \quad (32a)$$

Since this shrinkage due to recovery of dislocations is very small compared to other effects, the measurement is usually difficult or impossible. Even if all other effects can be excluded, the theory cannot distinguish between dislocation annihilation and formation of subgrain- or grain boundaries. Furthermore, there is no way to distinguish between dislocation annihilation in the grains and in the boundaries. This makes the comparison with XRD data difficult, since XRD only measures dislocations in the grains.

5 Experimental

5.1 Mathematical considerations for data processing

In the basics section (modified) Williamson-Hall-, (modified) Warren-Averbach- and double-Voigt methods were discussed separately. For data processing it can be convenient to combine some of these models, as done in the following.

5.1.1 Combination of modified Williamson-Hall and modified Warren-Averbach

Estimating the average dislocation contrast factor C for the modified Warren-Averbach/Williamson-Hall methods can be difficult. To avoid this, a combination of modified Williamson-Hall and modified Warren-Averbach method could be used, as done by [25].

The contrast factor C can be expressed as:

$$C = C_{h00}(1 - qH^2) \quad (33)$$

where C_{h00} is the average dislocation contrast factor for $(h00)$ reflections (can be looked up in books), q is a parameter depending on the elastic constants of the crystal and the character of the dislocations in the crystals and

$$H^2 = \frac{h^2k^2 + h^2l^2 + k^2l^2}{(h^2 + k^2 + l^2)^2}. \quad (33a)$$

The next steps aim at determining the constant q out of a set of experimental data.

With

$$\alpha = \left(\frac{K}{D}\right)^2 \quad (34a)$$

and

$$\gamma = \frac{\pi M b^2 \rho}{2} \quad (34b)$$

the modified Williamson-Hall equation (eq. 25) reduces to:

$$\frac{\Delta g - \sqrt{\alpha}}{g} \cong \sqrt{\gamma} \cdot \sqrt{C} + \frac{\pi M' b^2}{2} \cdot \sqrt{Q} \cdot g^2 C \quad (35)$$

Bringing eq. 35 to quadratic form and inserting eq. 33 into it, eq. 36 is obtained (the high order terms of eqs. 25/35 become negligible): [25] [26]

$$X = \frac{(\Delta g)^2 - \alpha}{g^2} \cong \gamma C_{h00}(1 - qH^2) \quad (36)$$

The crystallite size D (see eq. 34a) has to be implemented implicitly, so that it keeps the left side of eq. 36 linear.

As a result, q (together with other parameters as shown in section 9.2) and with eq. 33 C can be determined by a plot X vs. H^2 with linear extrapolation to $X = 0$ (see fig. 8) because

$$X = 0 \cong \gamma C_{h00}(1 - qH^2) \quad (37)$$

leads to

$$[H(X = 0)]^2 = \frac{1}{q} \quad (38)$$

When looking at eq. 27, the linear slope can be merged to eq. 39:

$$Y = \rho BL^2 \ln\left(\frac{R_e}{L}\right) \quad (39)$$

Eq. 39 can be determined in the modified Warren-Averbach plot (see eq. 27 and fig. 7) by a

$$y = ax^2 + bx + c$$

fit ⁸. By dividing eq. 39 through L^2 and separating the logarithm, eq. 40 is achieved: [25] [1]

$$\frac{Y}{L^2} = \rho B \ln(R_e) - \rho B \ln(L) \quad (40)$$

By plotting $\frac{Y}{L^2}$ vs. $\ln(L)$, the slope k

$$k = \rho B = \frac{\rho \pi b^2}{2} \quad (41a)$$

and

$$Y_0 = \rho B \ln(R_e) \quad (41b)$$

at $\ln(L) = 0$ can be evaluated as shown in fig. 9. Thus the dislocation density and the outer cutoff radius of dislocations R_e can be calculated.

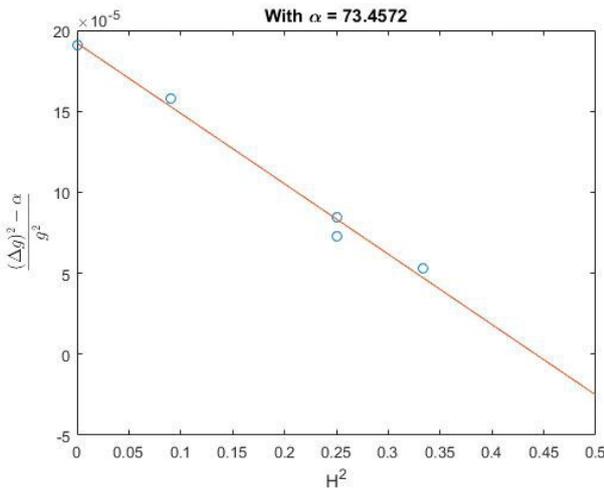


Figure 8: Determination of the parameter q

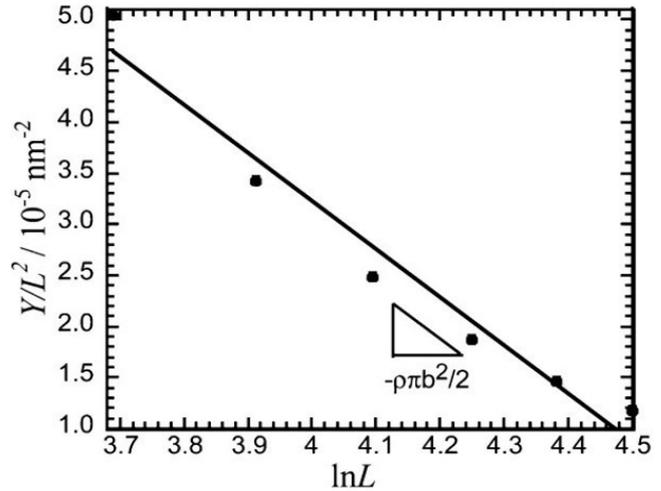


Figure 9: Determination of disl. density ρ [Fig.9]

In some cases, the determination of q and the contrast factor C as described above leads to an ill-conditioned problem. If that is the case, the modified Warren-Averbach plot can be done for several contrast factors C and an error minimization should reveal the "true" C -parameter. This obviously is very dangerous because errors might be "corrected" with a false contrast factor. Controlling the evaluated C is strongly advised!

Note: In the conventional Warren-Averbach analysis, the mean-square-strain is assumed to broaden only the Gaussian part of the peak. Later (around 1980), [27] and [28] showed that the mean-square-strain due to dislocations can be modeled by a sum of two strain-broadened profiles, one Cauchy and one Gaussian shaped. [29] even introduced a very similar expression to eq. 30 into the conventional Warren-Averbach method to correct this.

The modified Warren-Averbach method nevertheless introduced contrast factors for dislocations, but did not include the Cauchy part of microstrains into the model.

⁸It is quite common to do a linear fit instead, since a is usually very small

5.1.2 Combination of modified Warren-Averbach and Double-Voigt

If the Stokes deconvolution does not work e.g. because of Cu K_{alpha1} / K_{alpha2} overlapping, instrumental broadening can be extracted as suggested in the Double-Voigt-approach (section 4.2.2). Inserting the instrumental-broadening-free integral breadths into eq. 42 give the normalized Fourier transform of a Voigt function: [18]

$$A(L) = \exp \left[-2L\beta_C(g) - \pi L^2 \beta_G^2(g) \right] \quad (42)$$

Thus the (normal-/modified) Warren-Averbach routine can be continued.

As mentioned above, both (conventional- and modified-) Warren-Averbach methods assume that the peak-broadening due to microstrains only affect the Gaussian curve. This is done because many authors report Gaussian-like microstrain distributions, but it has never been physically justified. Even the inventor of the double-Voigt approach, D. Balzar, wrote: "Generally, it was shown that in the size-broadened profile the Cauchy part must dominate. No similar requirement for the strain-broadened profile exists. However, experience favors the assumption that it has to be more Gaussian." [18]

Lets assume that both Cauchy- and Gaussian parts contribute to the microstrains, as done in Balzar's Double-Voigt approach. D. Balzar achieved these microstrains $\langle \epsilon^2(L) \rangle$ by comparison with the Warren-Averbach method. For this reason, if the Cauchy-part of the microstrains are zero, the two definitions of "microstrains" are equal to each other. By substituting Balzar's microstrains (eq. 30) back into the Warren-Averbach-equation (eq. 24), eq. 43 is achieved: [18]

$$\ln [A(L)] = \ln [A^S(L)] - \frac{1}{g^2} \left[\pi \beta_{GD}^2(g) L^2 + 2\beta_{CD}(g) L \right] g^2 \quad (43)$$

By following the (modified) Warren-Averbach method (plot $\ln[A(L)]$ vs. g^2 (normal WA) or $\ln[A(L)]$ vs. $g^2 C$ (modified WA)) and determining

$$\frac{Y_{DV}}{L^2} = -\frac{1}{g^2} \left[\pi \beta_{GD}^2(g) - \frac{2\beta_{CD}(g)}{L} \right] \quad (44)$$

one can see the $\frac{1}{L}$ -contribution of the Cauchy-distortion parts. Since L is in the nm-region, this error can get enormous, despite very small Cauchy-contributions to the microstrains.

To determine $\beta_{GD}^2(g)$ and $\beta_{CD}(g)$ a plot $\frac{Y_{DV}}{L^2}$ vs. $\frac{1}{L}$ is useful, since $\beta_{CD}(g)$ is in the slope and $\beta_{GD}^2(g)$ is in the intercept with the y-axis.

If, as done in this work, the peaks were fitted as Voigt-peaks, the instrumental broadening was extracted as suggested in the Double-Voigt approach and the modified Warren-Averbach plot was fitted with straight lines, the fit will correspond to 100% to the data. In this case, the plot could be substituted by a simple equation.

Note: Let's call the intercept at $\frac{1}{L} = 0$ in fig. 10 $\frac{Y_{interc}}{L^2}$ and the slope $\frac{Y_{slope}}{L^2}$. In this case

$$-\frac{1}{g^2} \cdot \beta_{GD}^2 = \frac{Y_{interc}}{L^2 \pi} \quad (44a)$$

and

$$-\frac{1}{g^2} \cdot \beta_{CD} = \frac{Y_{slope}}{2L^2} * L \quad (44b)$$

which leads to a microstrain-term that is programmable much better when eq. 44a and eq. 44b are inserted back into eq. 30:

$$\langle \epsilon^2(L) \rangle = \frac{Y_{interc}}{2\pi^2 L^2} + \frac{Y_{slope}}{2\pi^2 L^2} \quad (44c)$$

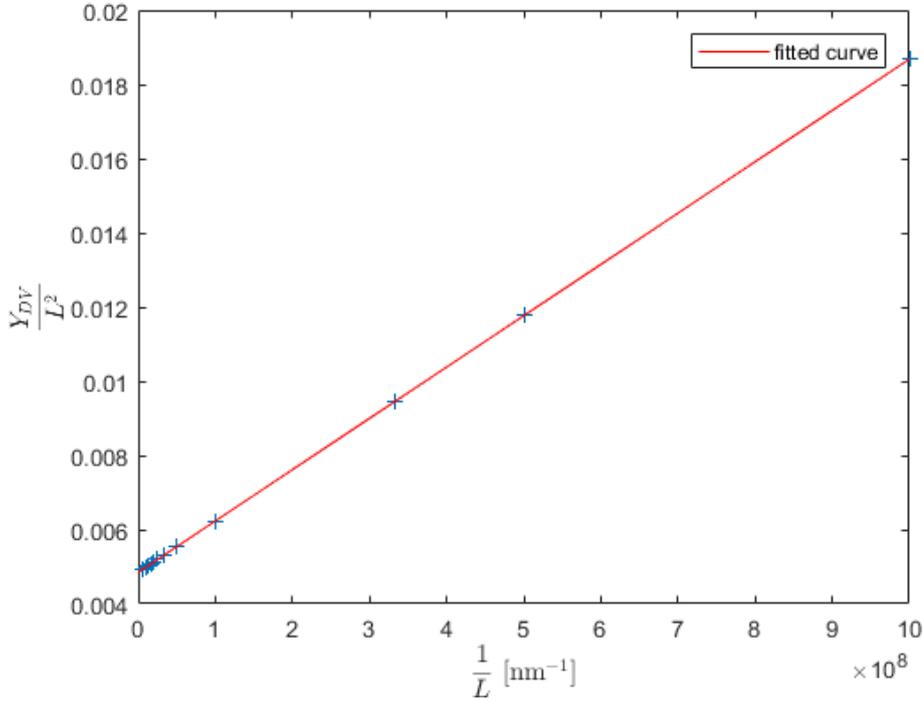


Figure 10: Determination of β_{CD} from the slope and β_{GD}^2 from the intercept with the y-axis

Afterwards, the $\frac{Y}{L^2}$ vs. $\ln(L)$ plot can be done as suggested in the modified Warren-Averbach method and the fit can be done according to eq. 44. Again, the fit will exactly correspond to the data if the peaks were fitted as Voigt-peaks, the instrumental broadening was extracted according to the Double-Voigt approach and the modified Warren-Averbach plot was fitted with straight lines. This case can be seen in fig. 11. The described method will be called "contrast-corrected double-Voigt method" for the rest of this work.

To convert the microstrains into dislocation densities, e.g. the Williamson-Smallman equation can be used: [30]

$$\rho = \frac{j \langle \epsilon^2 \rangle}{Fb^2} \quad (45)$$

j and F are constants that depend on the lattice and the interaction between dislocations respectively. In this work, $j = 14.4$ for bcc structure and $F = 1$ where used. [30]

But notice: In both cases (modified Warren-Averbach and the contrast-corrected double-Voigt method) it is assumed, that all microstrains are due to dislocations. Nevertheless, the modified Warren-Averbach method assumes a "internal dislocation density" (mobile + dipoles) and the Williamson-Smallman assumes only mobile dislocations. This means that the calculated results are not comparable one to one, but the trend should be the same.

Furthermore, the error can be evaluated: Comparing the terms for the microstrains of the conventional- and the modified Warren-Averbach method (comparing the 2nd terms of eq. 24 and eq. 27) leads to eq. 46:

$$\frac{\pi b^2 L^2 \rho_{mWA}}{2} \ln\left(\frac{R_e}{L}\right) g^2 C = 2\pi^2 L^2 g^2 \langle \epsilon^2(L) \rangle \quad (46)$$

which reduces to:

$$\rho_{mWA} [\ln(R_e) - \ln(L)] g^2 C = \frac{4\pi}{b^2} g^2 \langle \epsilon^2(L) \rangle \quad (47)$$

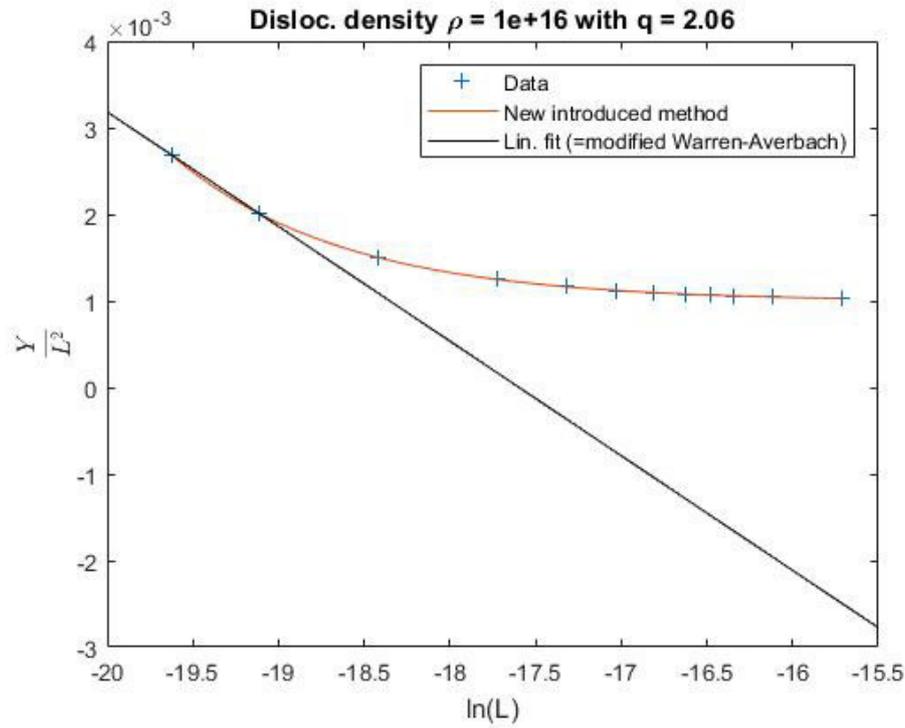


Figure 11: Modified Warren-Averbach method including microstrain-correction of cauchy-parts. The orange fit with the new introduced "contrast-corrected double-Voigt method" exactly corresponds to the data points, while the linear modified Warren-Averbach fit (black line) does not.

Inserting the Williamson-Smallman approach (eq. 45) and substituting g^2 in the conventional Warren-Averbach approach by $g^2 \cdot C$ leads to eq. 48:

$$\rho_{mWA} [\ln(R_e) - \ln(L)] g^2 C = \frac{4\pi}{b^2} \cdot \frac{b^2 F \rho_{WS}}{j} g^2 C \quad (48)$$

which reduces to:

$$\rho_{mWA} = \frac{4\pi \cdot F \cdot \rho_{WS}}{j [\ln(R_e) - \ln(L)]} \quad (49)$$

5.2 Sample preparation

All steel samples were P91 steel, austenitized at 1060 °C, air quenched to room temperature, tempered at 760 °C and again air quenched to room temperature. Table 2 clarifies the sample names and table 3 shows the tempering times.

Table 2: Sample names

Sample name	Meaning
P91	name of the steel used
BH	After austenitizing; no tempering at 760 °C was done
xxh_760C	tempered for xx h at 760 °C
XX_1 _to_ XX_2	measurement was done from XX_1 ° 2 Θ to XX_2 ° 2 Θ

Table 3: Heat treatments for steel samples:

Aust. t. [h]... Time for austenitization at 1060 °C in h

tempering t. [h]... Time for tempering at 760 °C in h

Sample	Aust. t. [h]	tempering t. [h]
P91_BH_vib_pol	1	0
P91_0p166h_760C	1	1/6
P91_0p5h_760C	1	0.5
P91_1p5h_760C	1	1.5
P91_12h_760C	1	12
P91_24h_760C	1	24
P91_36h_760C	1	36

For XRD measurement, all samples were grinded and polished. Especially polishing was tried to be kept as short as possible (nevertheless making sure to remove all scratches), because long polishing times tend to rip precipitates out, leaving back holes. Finally, vibration polishing was done for 1.5 h with the OP-AA suspension from Struers (Code: APACI) and no sample holder. At first, polishing for 3 h was intended but checking of the surface under SEM after 1.5 h showed this time was completely sufficient.

The dilatometry samples were austenitized and afterwards milled to square columns with 20x4x4 mm. No further surface treatment or tempering was done.

5.3 X-ray diffraction

For the X-ray measurements, a PANalytical Empyrean X-ray diffractometer with copper anode and monochromator in $\Theta/2\Theta$ mode on the Institute for Solid State Physics on the TU Graz was used. Detailed measurement specifications are in the appendix. For the alignment, a copper beam attenuator, the PIXcel alignment Anti-scatter slit 0,1 mm (on diffracted beam side) and the detector in alignment mode were used.

5.4 Dilatometry

For dilatometry, a self-assembled high precision laser dilatometer of the Institute of Materials Physics of TU Graz was used. Heating was done with 4 halogen spotlights, the thermo wires were spot-welded onto the samples. At first, only measurements under vacuum were possible. Later on, the machine was remodeled and measuring under argon atmosphere got included. Since a considerable amount of Cr evaporated at all measurements without argon (verified by EDX and EBSD measurements), only the measurement under argon atmosphere is considered in this work.

5.5 Data processing

In the following, the data processing routines are described as flowcharts. The MATLAB code was developed in this thesis and can be found in the appendix. Generally, the data processing is done in 3 steps and each step has its own Matlab program:

- **Processing of raw XRD data:** The measured data is fitted with pseudo-Voigt peaks and the corresponding integral breadths (=IBs) are calculated. This has to be done for the standard sample and the sample from which one wants to know the dislocation density and is done in the Matlab program "XXX_IB_ges_Auswertung_split.m". The subroutine that performs the data fit was not written by the author but was downloaded from the Net. [31]
- **Interpolation of the instrumental broadening:** The integral breadths of the standard sample are plotted and interpolated with an exponential fit. The instrumental broadenings at the peak positions of the to-be-evaluated sample are read out and stored into a file. This is done in the Matlab program "LaB6_IB_fit_aktuell.m".
- **Calculation of dislocation densities:** The contrast factor of dislocations is evaluated with the modified Williamson-Hall method. After that, the microstrains are qualitatively evaluated with the modified Warren-Averbach and the contrast-corrected double-Voigt method. Both microstrain-results are plotted into one figure and the internal dislocation density for the modified Warren-Averbach method is put out. Furthermore, the microstrains according to the contrast-corrected double-Voigt method are evaluated quantitatively and the according dislocation density is calculated using the Williamson-Smallman approach. This is done in the Matlab program "MWH_MWA.m".

Flowchart symbols

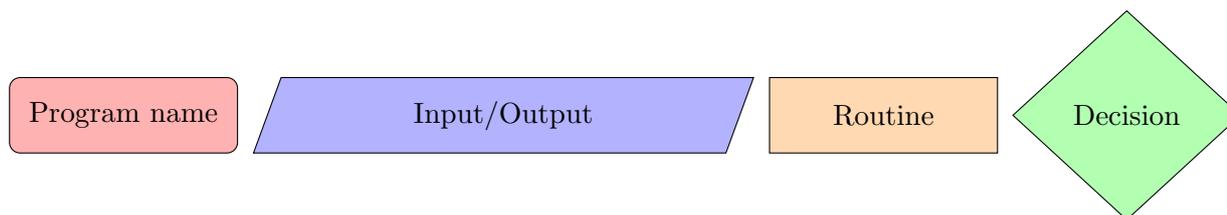


Figure 12: Flowchart symbols

5.5.1 Matlab programs for data processing

First of all, the XRD raw data (.xrdml files) have to be converted into .xy files. This can be easily done with the program PANalytical Data Viewer, that comes with the used XRD machine. The .xy files can then be imported into Matlab.

The following flowchart (fig. 13) shows how the Matlab routines work in general and how they are connected to each other.

Processing of raw XRD data: Flowchart fig. 14 shows the structure of the peak fitting programs XXX_IB_ges_Auswertung_split.m. In general, the mechanics of "LaB6_IB_ges_Auswertung_split.m" and "steel_IB_ges_Auswertung_split.m" are the same, the only difference are the presets.

"LaB6_IB_ges_Auswertung_split.m" will automatically search for the LaB₆ diffraction peaks and try to create a fit for them. Likewise, "steel_IB_ges_Auswertung_split.m" knows where the steel diffraction peaks should be and in which range reasonable starting parameter could be. The subroutine that performs the fitting itself (fitting ONE set of starting parameters to the data) was downloaded from [31].

Note: The 3 repetitions are done to get a feeling how stable the fitting process is. If all 4 fits have exactly the same minimum, the number of different starting parameter sets is more than sufficient. If all 4 give different minima, one should increase the number of starting parameter sets.

The data output has to be copied into a .txt file manually. Also take into account that the Matlab output often has ",," as comma, but only accepts "." as input for the next routine.

Interpolation of the instrumental broadening (no flowchart): The program for fitting the LaB₆ integral breadths is very simple. The data achieved with the peak fitting program (and manually stored into a .txt file) is read in for the LaB₆ sample and the to-be-evaluated sample. The integral breadths for the LaB₆ sample are plotted and interpolated with an exponential fit. The (fitted) LaB₆ integral breadths at the peak maxims of the to-be-evaluated sample are read out and considered as the "instrumental broadening". Since the instrumental broadening should not change too often⁹, the LaB₆ fitting program writes the results into a file which is automatically imported into "mWH_mWA.m" every time "mWH_mWA.m" is used. So when testing something with the LaB₆ fitting program, changing the name of the output file is strongly recommended!

Calculation of dislocation densities: Flowchart fig. 15 shows how the program "mWH_mWA.m" processes the integral breadths to calculate dislocation densities. A detailed, pure mathematical description is done in section 5.1.

Note: α is a variable that has to be determined implicitly, but the system is not very sensitive on it. In this code, the plot is fitted for ~ 5000 different values (can be changed) of α and the α that leads to the most accurate fit (mean square error \rightarrow min) is considered as correct.

⁹When evaluated carefully once, the instrumental broadening only changes when the measurement setup is changed.

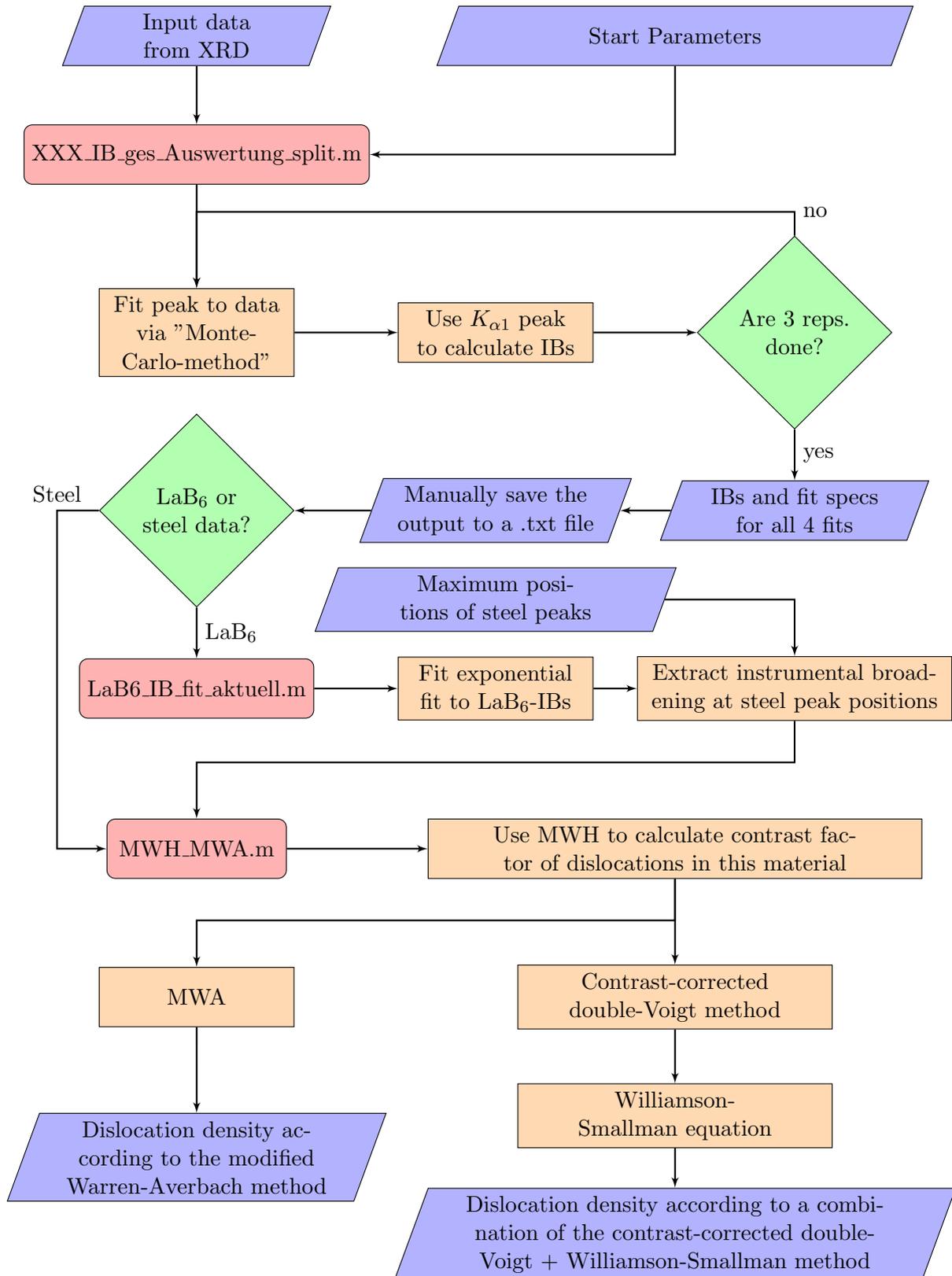


Figure 13: Flowchart of raw data processing routine.

IB... integral breadth

reps... repetitions

specs... specifications

MWH/MWA... modified Williamson-Hall/modified Warren-Averbach

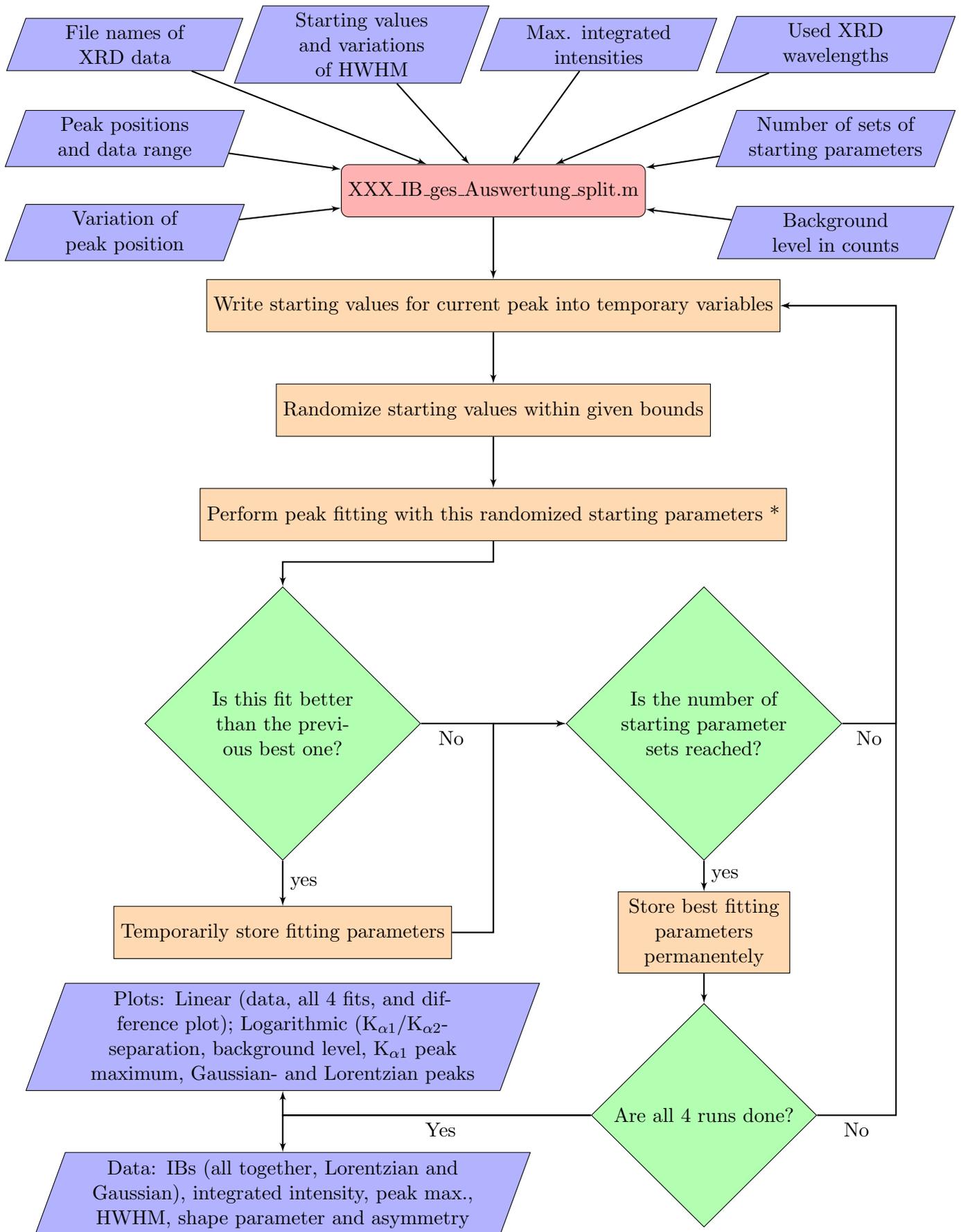


Figure 14: Flowchart of peak fitting routine.

*...Function downloaded from [31]

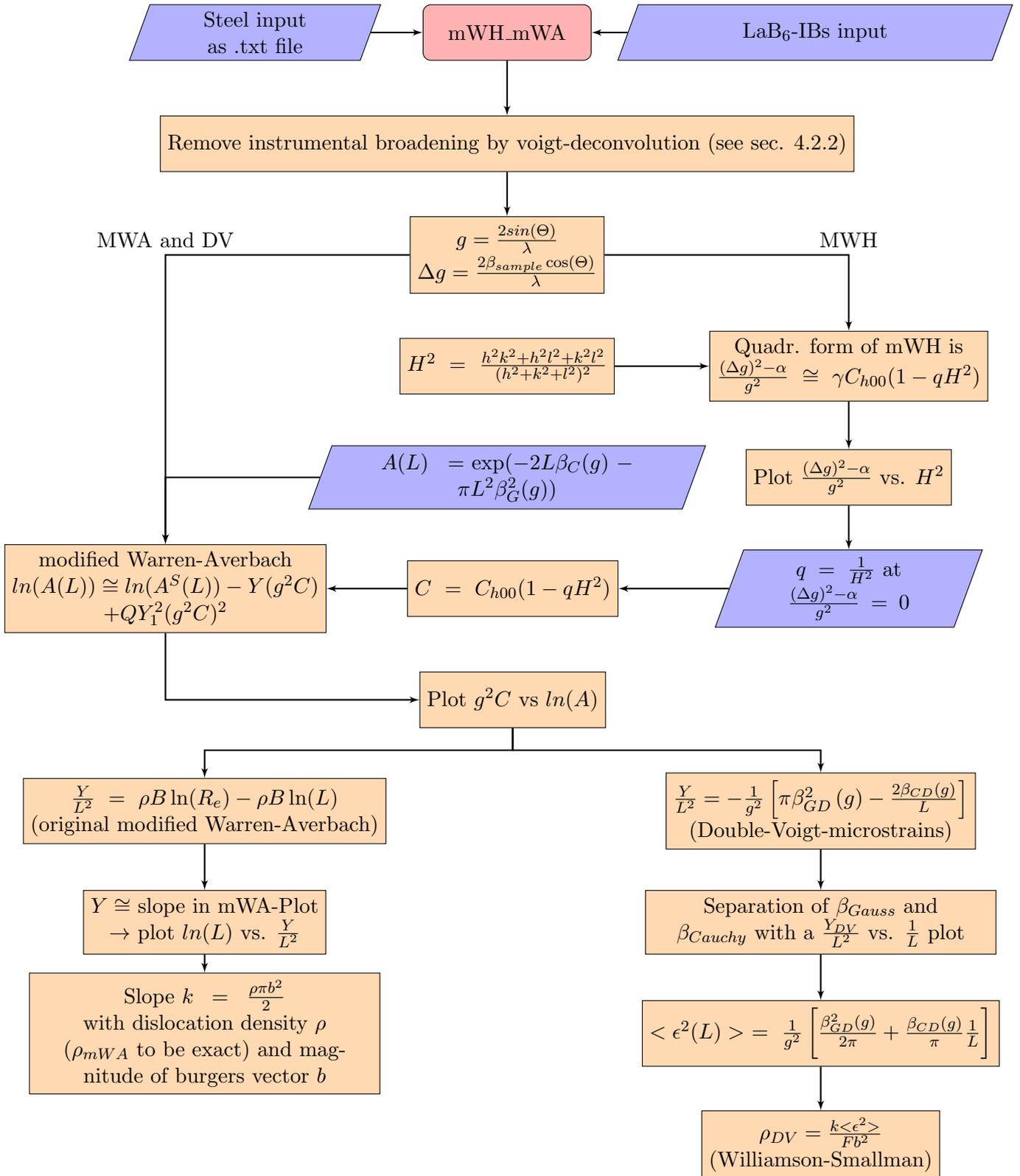


Figure 15: Flowchart to explain the combination of mWH with mWA and double-Voigt

6 Results

6.1 X-ray diffraction

In this work, a NIST SRM 660 ¹⁰ was spin-coated onto a Si wafer and used to determine the instrumental broadening (see sec. 4.2). From a previous (not published) work [32] it was known that strong parasitic peaks ¹¹ interfere in some 2Θ regions. In order to exclude these regions and to save machine time, the measurement was split into four parts and only usable regions were measured. The corresponding data are plotted in fig. 16.

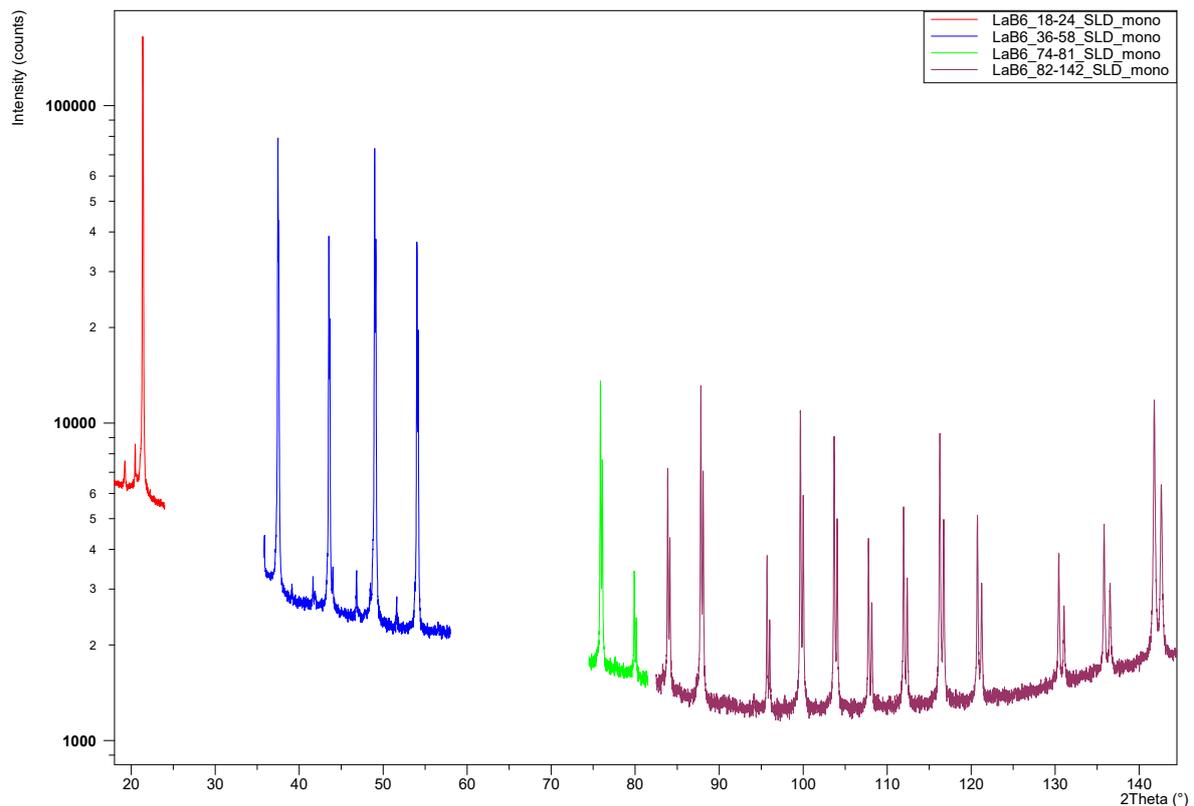


Figure 16: Measurement of the LaB_6 standard. The measurement was split into four parts and regions with too strong parasitic peaks were left out

For steel and a XRD machine with copper anode, there are 6 peaks (110, 200, 211, 220, 310 and 222) in the range between $40^\circ < 2\Theta < 140^\circ$. To save machine time, every peak was measured separately with a range of $\Delta 2\Theta = 10^\circ$. Due to difficult background levels (determination of the integral breadth β is highly unstable!), the 110 peak was left out.

¹⁰NIST: Nation Institute of Standards and Technology
SRM: Standard Reference Material
660: Reference material number for LaB_6

¹¹In general, all not coherent with the main source and not wanted X-rays are called "parasitic"

Figures 17 to 23 show the results of P91 steel measurements. Parasitic peaks can be seen very clearly, especially the tungsten K_{α} peaks for the 65 ° and 82 °. Also, the Cu $K_{\alpha 1}/K_{\alpha 2}$ -doublet can be distinguished when looking at the LaB₆ standard or the 0.5 h to 36 h tempered samples, see fig. 16 and fig. 19 to fig. 23. For the untempered or very short tempered samples, the $K_{\alpha 1}/K_{\alpha 2}$ peaks melt to one peak (broader peaks!), and the doublet can't be seen that clearly. When looking at the untempered 200 peak (fig. 17), one can clearly see a second peak beneath the main peak (marked with 200a). The author suspect it to be because of the bct-structure of untempered martensite. This would explain the peak vanishing after very short tempering times, since the carbon then leaves the octahedral sites to form carbides, allowing the steel to become (more or less) bcc.

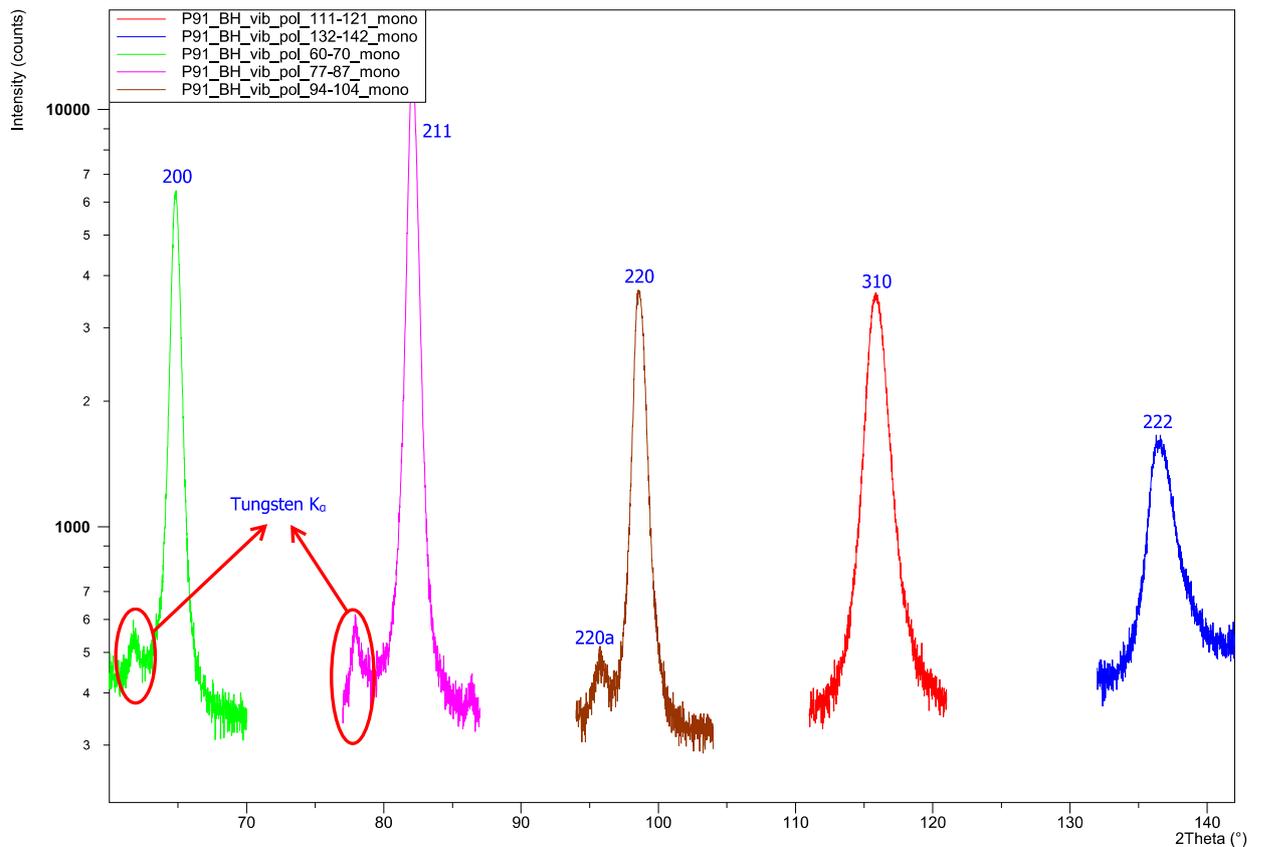


Figure 17: Measurement of the austenitized (at 1060 °C) steel sample prior to tempering (P91_BH)

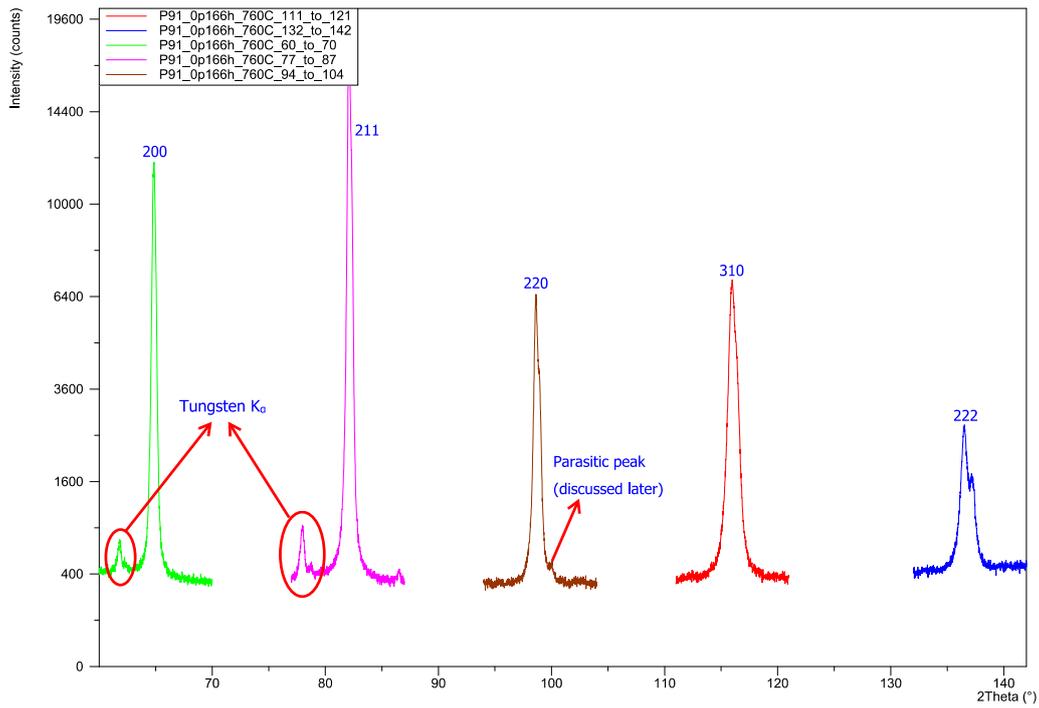


Figure 18: Measurement of the austenitized (at 1060 °C) and afterwards 10 min tempered (at 760 °C) steel sample (P91.0p166h.760C)

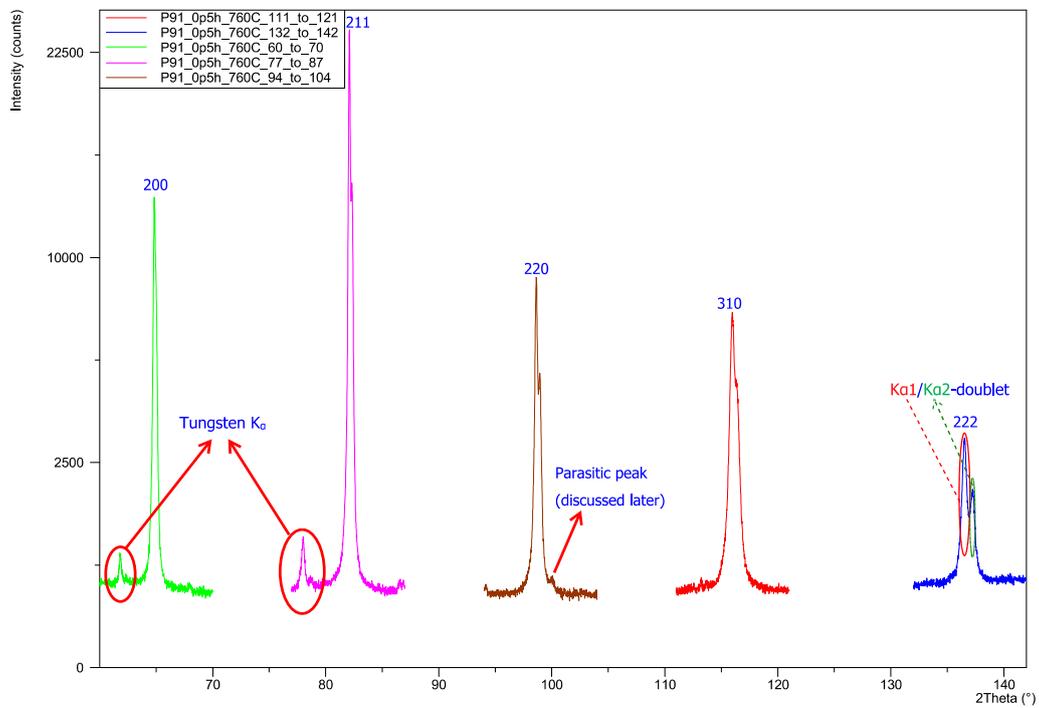


Figure 19: Measurement of the austenitized (at 1060 °C) and afterwards 30 min tempered (at 760 °C) steel sample (P91.0p5h.760C)

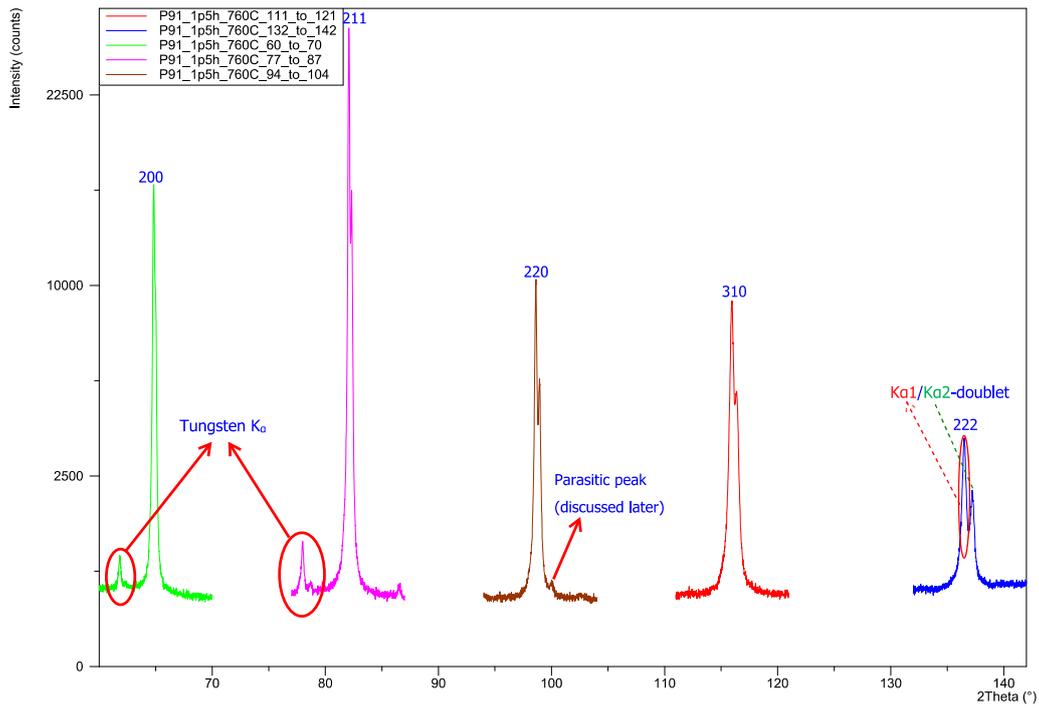


Figure 20: Measurement of the austenitized (at 1060 °C) and afterwards 1.5 h tempered (at 760 °C) steel sample (P91_1p5h.760C)

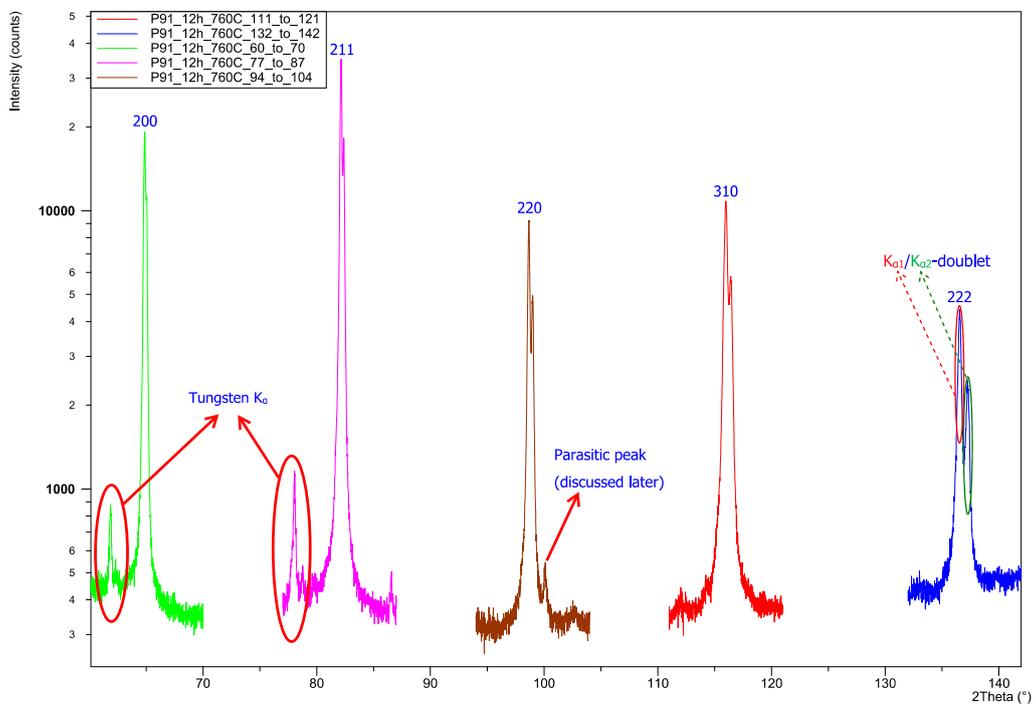


Figure 21: Measurement of the austenitized (at 1060 °C) and afterwards 12 h tempered (at 760 °C) steel sample (P91_12h.760C)

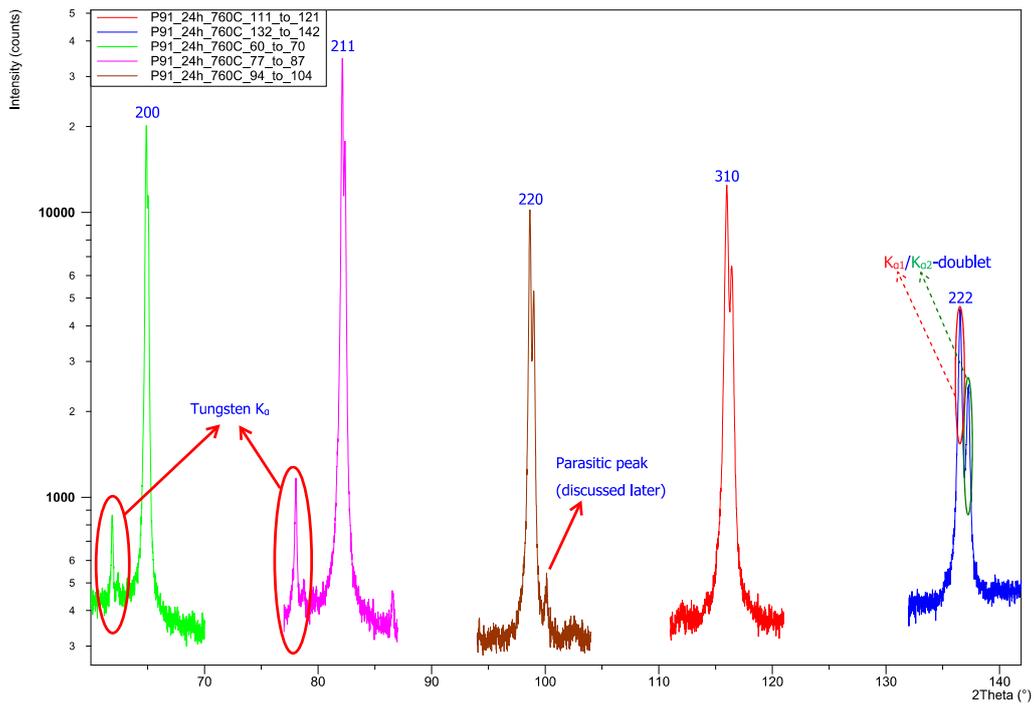


Figure 22: Measurement of the austenitized (at 1060 °C) and afterwards 24 h tempered (at 760 °C) steel sample (P91_24h.760C)

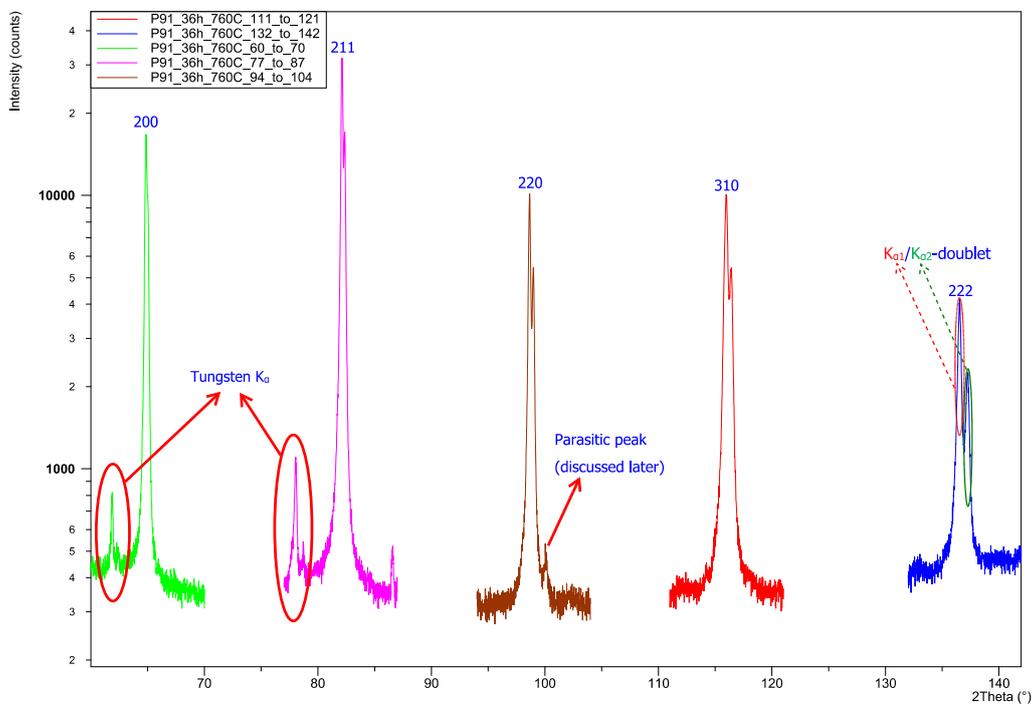


Figure 23: Measurement of the austenitized (at 1060 °C) and afterwards 36 h tempered (at 760 °C) steel sample (P91_36h.760C)

6.2 Dilatometry

The results of the dilatometry measurement under Argon atmosphere are plotted in fig. 24

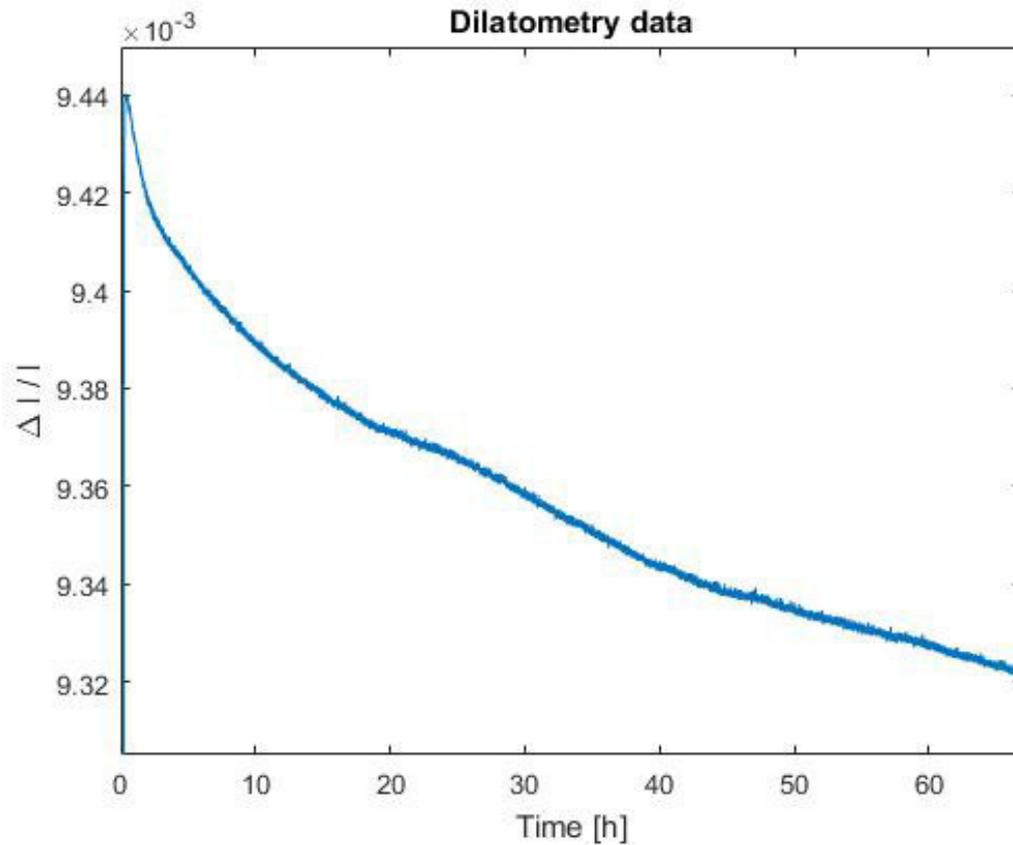


Figure 24: Dilatometry measurement under Argon atmosphere.

Δl ... length change

l ... total sample length

Between 15 h and 30 h, there appears to be some peak. The author does not know what is causing it but, unless one wants to know the dislocation density in this time-slot, it does not interfere with the dislocation density calculation.

7 Data Analysis and Discussion

7.1 Fitting of XRD peaks

There are many free-to-use peak fitting programs in the Internet. After some tests with "Fityk" in a previous project [32], it was clear that the fitting accuracy is not sufficient. For that reason, the routine described in the flowchart fig. 14 was programmed and used for fitting the XRD data. For the fitting procedure between 5 000 and 10 000 different randomized sets of starting parameters were used, depending on how fast the global minimum was found. After completing the fit, there are some mechanisms to qualitatively check the fit's quality. In fig. 25 XRD data and all 4 fits are plotted linearly. Beneath, there is a linear difference plot. If all 4 fits fall into the same minimum, the curves are identical and only the last one (green curve) can be seen.

In fig. 26 a semi-logarithmic plot of the fit is shown. In this plot, the contributions of $K_{\alpha 1}$ and $K_{\alpha 2}$ as well as the background level are separated and the $K_{\alpha 1}$ peak maximum can be checked. When looking at the $K_{\alpha 2}$ -curve, a kink can be seen. This is an artifact from fitting the peak as split-peak (left and right part of the peak are fitted separately).

Furthermore, plots with the following information can be enabled (mostly for searching bugs):

- Plot of the starting parameters that gave the best fit.
- Plot $K_{\alpha 1} + K_{\alpha 2}$ fit to see if separation worked properly.
- Plot of Lorentzian and Gaussian parts of the peak.

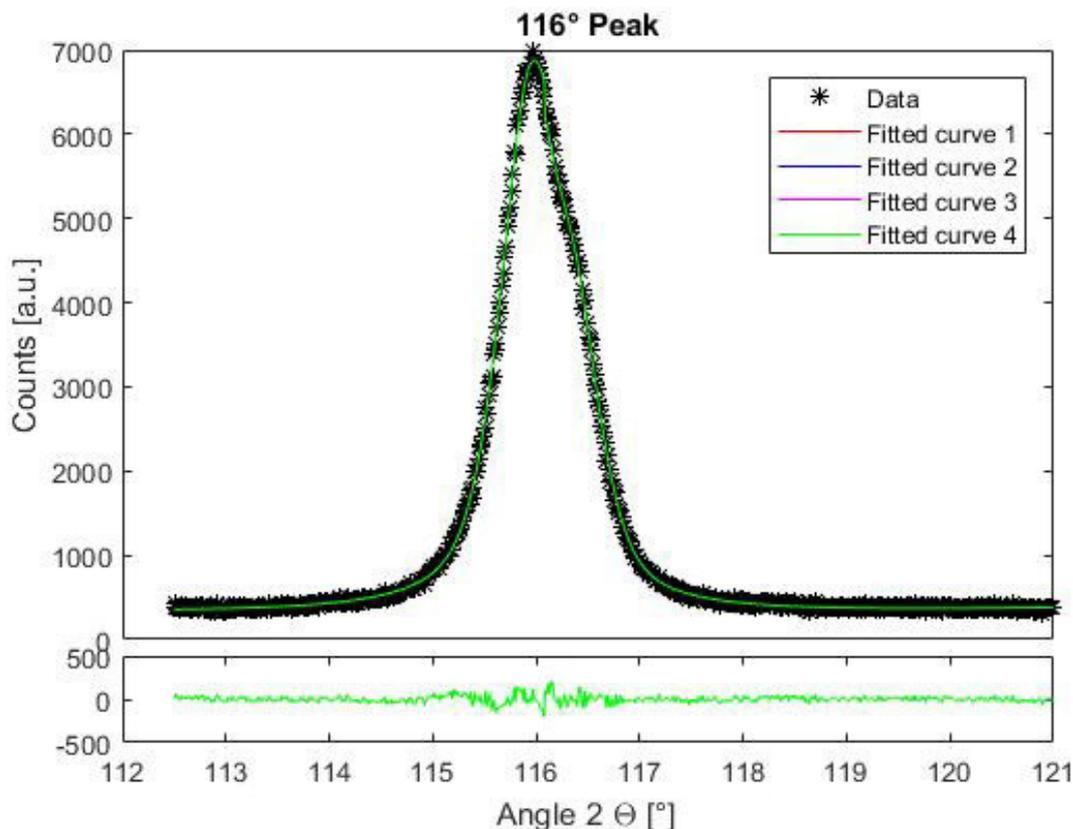


Figure 25: Plot to check if all 4 fits fall into the same minimum. Linear plot of data and corresponding fit(s) (upper plot) and difference plot (lower plot)

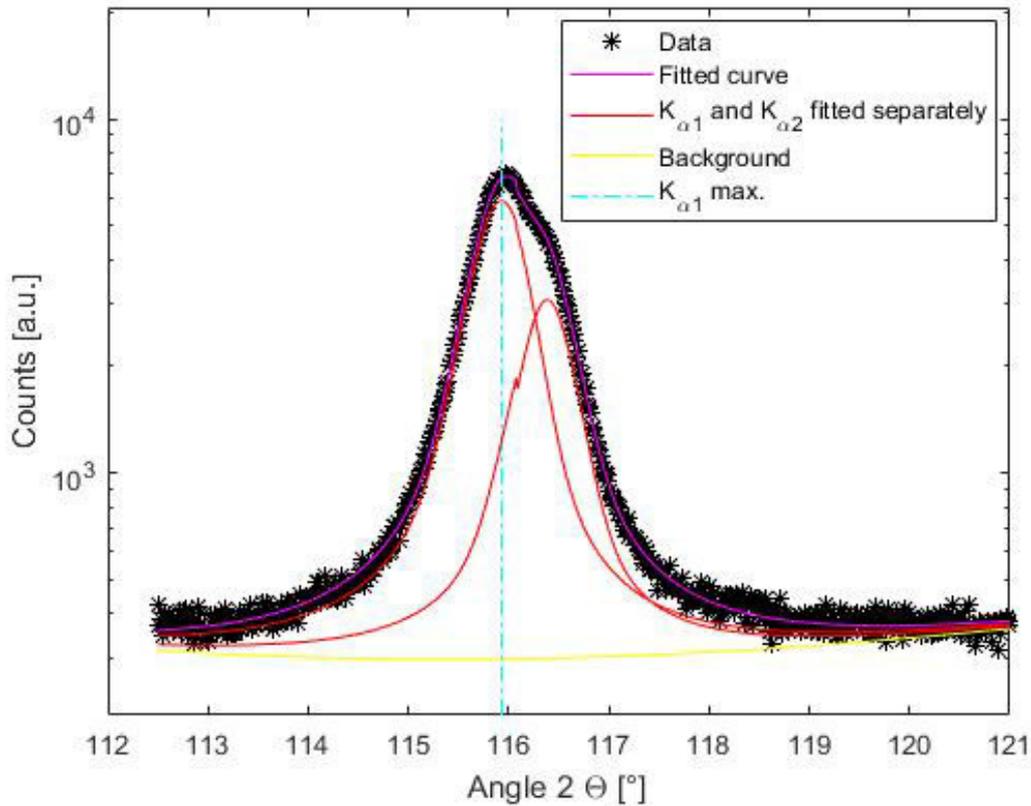


Figure 26: Example for a semi-logarithmic plot to check $K_{\alpha 1} / K_{\alpha 2}$ separation, background and $K_{\alpha 1}$ peak maximum.

As shown in figures 17 to 23, parasitic tungsten peaks appear a few degrees 2Θ beneath the 200 and the 211 peaks. Furthermore, another parasitic peak is a bit above the 220 peak. Excluding these parasitic peaks has to be done very carefully to keep potential errors as low as possible. For that reason, the main peaks were cut off prematurely where parasitic peaks appear in this work. This can be seen in figures 27 and 28 and is of course one source of errors.

For the 200 and the 211 peak, including the tungsten peak into the fitting procedure and fitting the whole $\Delta 2\Theta = 10^\circ$ was also tested. The result was the same as with the truncated fitting procedure, but needed disproportionately longer.

The fits for all measured specimen can be found in figures 45 to 68 and tables 6 to 19 in the appendix (pages 53 to 61 and 62 to 71).

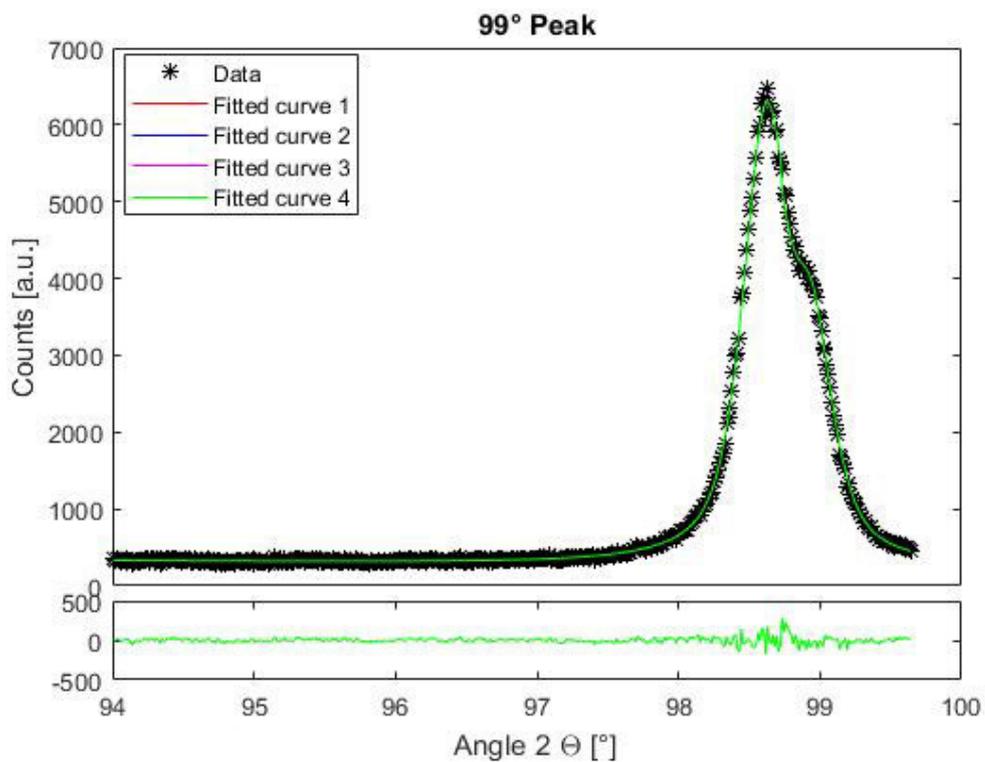


Figure 27: Same plot as fig. 25, but this time for a truncated peak.

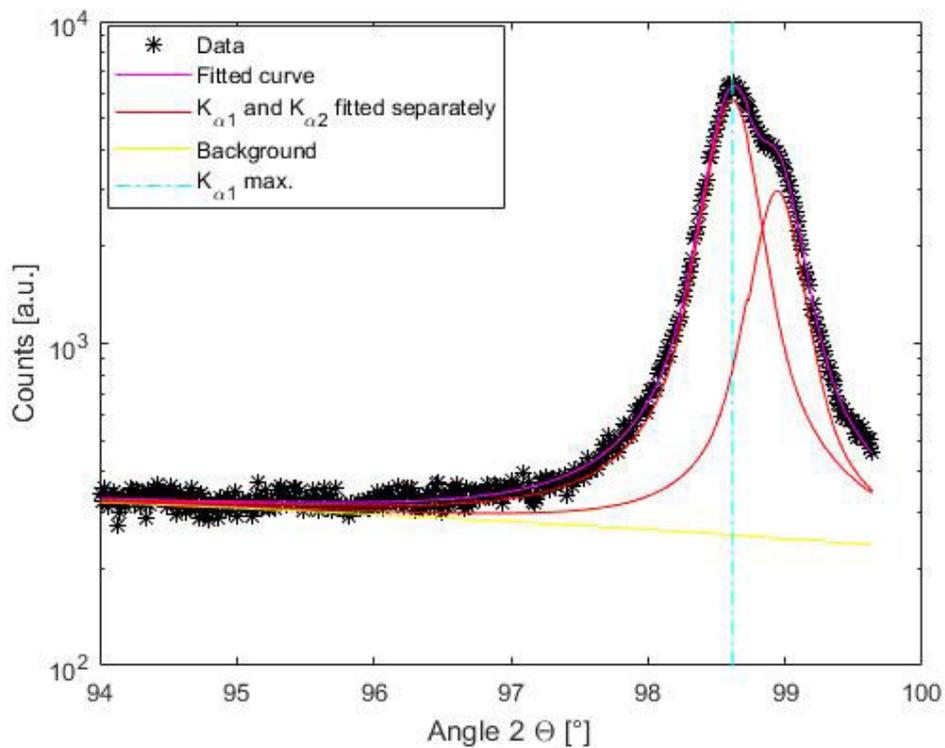


Figure 28: Same plot as fig. 26, but this time for a truncated peak.

7.2 Separation of size- and strainbroadening

Aim of this work was to evaluate the potential of X-ray diffraction in a standard laboratory diffractometer for measurements of dislocation densities in complex martensitic steels. To do so, the most used and/or most promising methods were discussed in the basics chapter (section 4). Some potentially good methods were excluded due to measurement or material restrictions, e.g. Groma's peak tail analysis [11] [33] [34] couldn't be done because of parasitic peaks and variations in the peak tails.

7.2.1 Williamson-Hall method (also see chapter 4.4):

This method is for very rapid coherent domain size and microstrain analysis. It can be used to have an impression if peak broadening is mostly due to coherent domain size or due to microstrains [9]. Even if the error bars are low, there is not more information to be extracted because the models used are too rough. The strain model used in this method is called "apparent strain" and represents a strain maximum (not a mean value!) and the data points in the Williamson-Hall plot can vary a lot. An example for a Williamson-Hall plot assuming size broadening solely affects Lorentzian and microstrains solely Gaussian curve is shown in fig. 29. According to the theory, the coherent domain size can be calculated from the intercept with the y-axis and the microstrains from the slope.

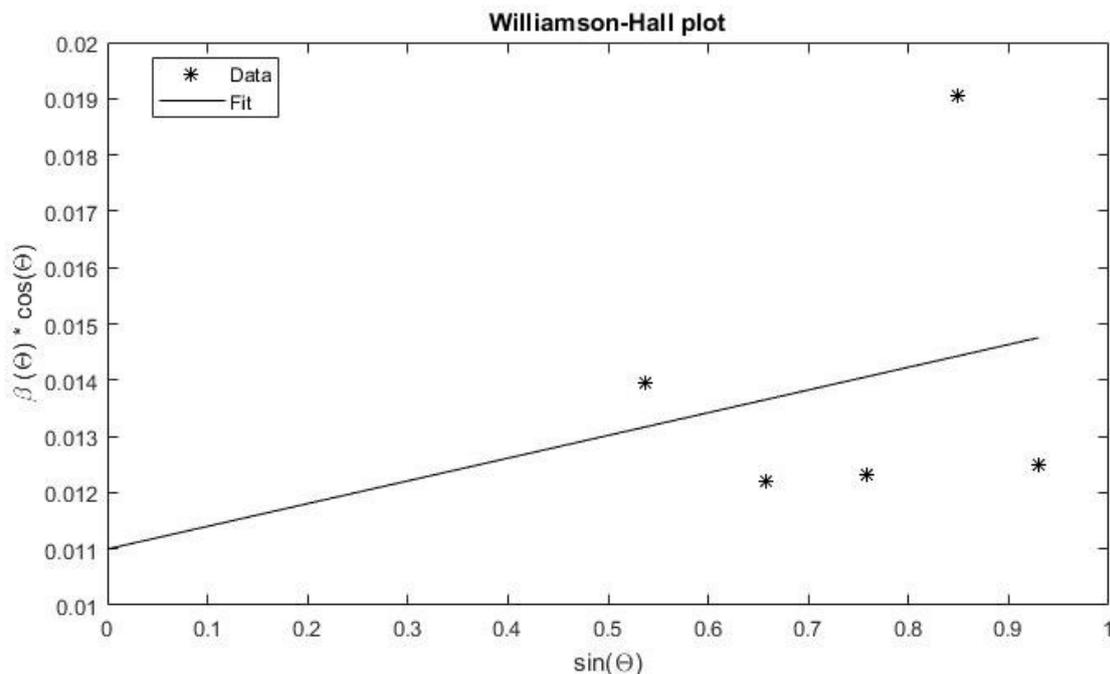


Figure 29: Williamson-Hall plot of sample P91_0p166h_760C with $\beta = \beta_{sample}$.

Fitting a straight line to the strongly varying six data points of ferrite and calculating the microstrains (and with them the dislocation density) from the slope usually give errors so huge, that presenting them would be unscientific. This also applies to this work, even though other, more advanced methods, do not have problems with scattering. This is why it is not used in this work.

7.2.2 Modified Williamson-Hall technique (also see chapter 4.6):

T. Ungár's **modified Williamson-Hall technique** improves the scattering in the Williamson-Hall plot, thus decreasing the errors. Nevertheless, considerable scattering and most problems and limitations of the conventional Williamson-Hall method remain. An example for a modified Williamson-Hall plot from another paper is shown in fig. 30.

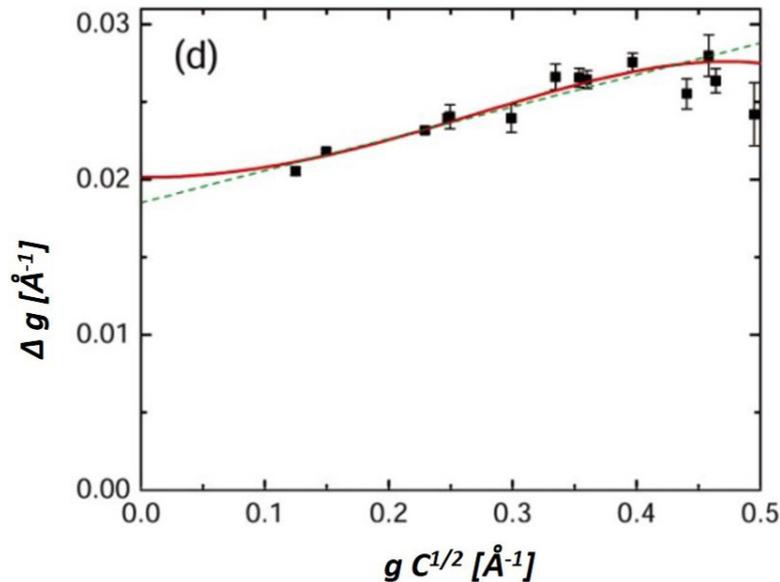


Figure 30: Example of a modified Williamson-Hall plot [Fig.30]

Please Note: As shown in section 5.1, the contrast factors for the modified Warren-Averbach method in this work are calculated with the modified Williamson-Hall (mWH) technique. This is done by optimizing C so the errors for the quadratic form of the modified Williamson-Hall method are minimized. To insert these contrast factors back into the mWH-equation would be mathematically incorrect, because all systematic errors would be "corrected" with a wrong C . For comparison of the same measurement with the other data evaluation methods, the mWH plot of sample P91_0p166h.760C is shown in fig. 31.

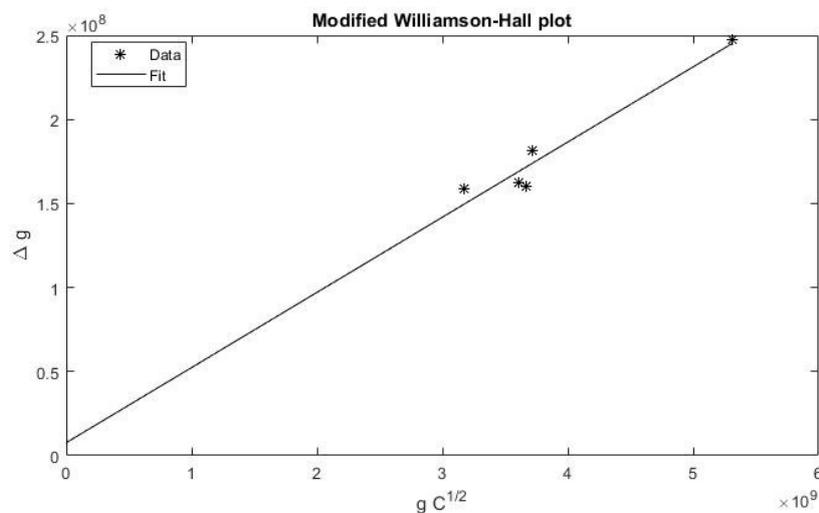


Figure 31: Modified Williamson-Hall plot of sample P91_0p166h.760C

Comparing fig. 30 and fig. 31, two things become evident:

- Fig. 30 has much more data points. This is because a different sample material and a different X-ray wavelength were used.
- In fig. 30 a quadratic and in fig. 31 a linear fit is used. This is because there are too few data points and a too small angular range for a quadratic fit in fig. 31. Because the quadratic part is small (according to the method!), the error by neglecting it is smaller than the one introduced by applying a quadratic fit to this data.

Furthermore, the very right part of fig. 30 indicates a negative slope, which is physically impossible. This shows, that even though the scattering is reduced, the limitations are mostly identical to those of the conventional Williamson-Hall method.

7.2.3 Warren-Averbach method (also see chapter 4.5):

Data processing takes a lot longer than for the Williamson-Hall method, but the data achieved have a better reproducibility due to a more fundamental physical and mathematical basis. With this method, a "mean-square-strain" is evaluated. An example for a Warren-Averbach plot is shown in fig. 32.

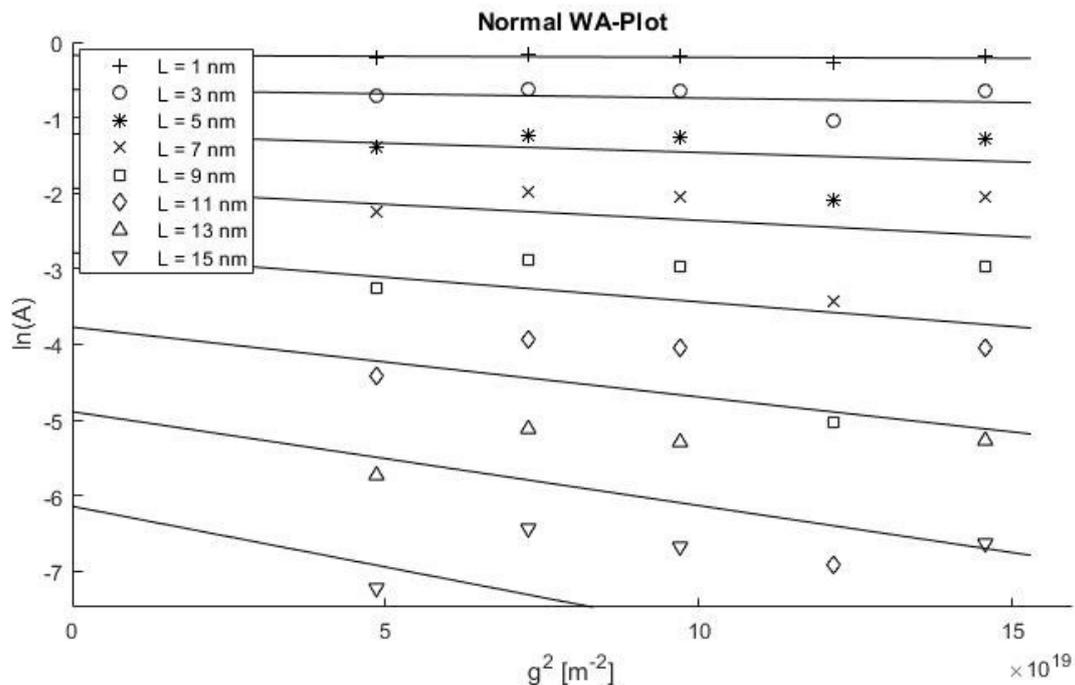


Figure 32: Warren-Averbach plot of sample P91_0p166h_760C

Like for the Williamson-Hall method, there are problems with scattering of data points. If at least two orders of the same reflection can be measured, only these peaks are used for data processing because scattering should be negligible in that case. In many measurements, this approach is not possible (like in this work). In these cases the scattering often can be reduced by using T. Ungár's modified Warren-Averbach method.

7.3 Modified Warren-Averbach method (also see chapter 4.6):

The **modified Warren-Averbach method** is evaluated in detail later in this section. It is used for data processing in this work and compared to the new method introduced in chapter 5.1.2. A comparison to D. Balzar's original double-Voigt method is not done because of the considerable scattering of data points in the double-Voigt method for the used material. An example for a modified Warren-Averbach plot is shown in fig. 33.

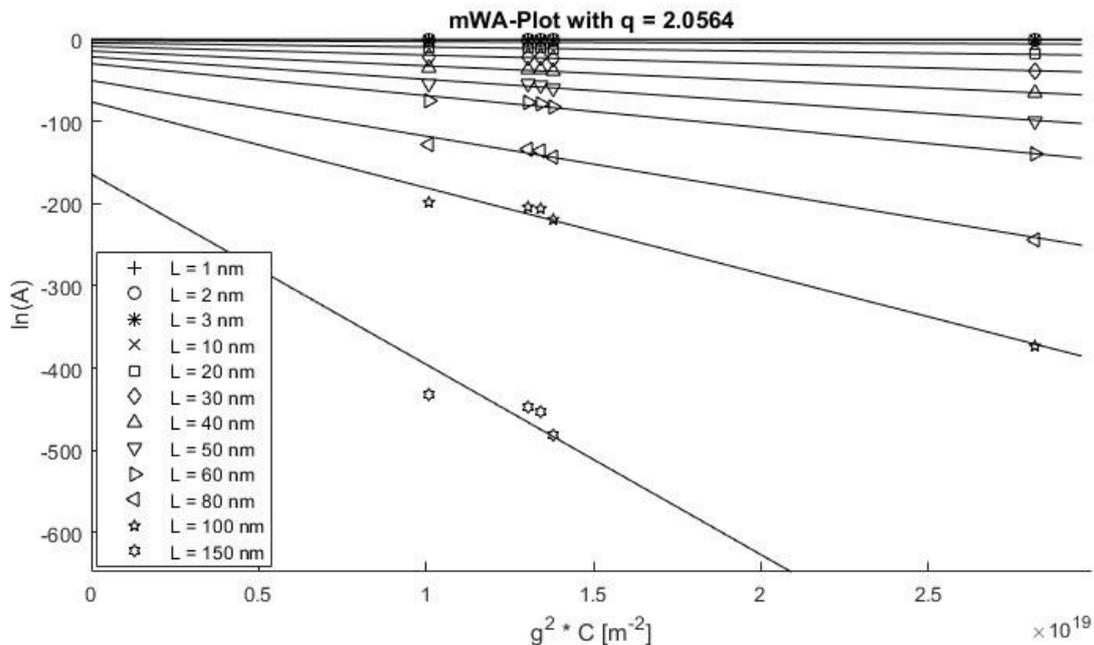


Figure 33: Modified Warren-Averbach plot of sample P91.0p166h.760C

7.3.1 Double-Voigt method (also see chapter 4.8):

It is, due to limitation of peak shapes to Voigt- and Pearson VII peaks (e.g. super-Lorentzian peaks appear to be a problem), mathematically more restricted than the conventional Warren-Averbach method. On the other hand, it has many improvements (as discussed later) and it is rather easy to check if the restrictions apply to the to-be-evaluated sample. Nevertheless, it is the least used of the evaluated models. Most authors just conclude that the evaluated microstrains are much higher than the microstrains achieved with other methods. This is most likely because D. Balzar also includes Lorentzian contribution to the microstrains (in the double-Voigt method), whereas microstrains are solely affected by the Gaussian contribution according to the Warren-Averbach method. In the double-Voigt method, one can decide if the "apparent strain" or the "mean-square-strain" should be evaluated. Furthermore, the extraction of the instrumental broadening is much easier than the Stokes deconvolution that is usually used for the Warren-Averbach method. Some material properties, most likely the high dislocation density, lead to considerable scattering of data points for the used material (P91 steel). If at least two orders of the same reflection can be measured, the same "trick" as for the Warren-Averbach method can be used. Because the contrast-corrected double-Voigt method is mathematically very similar to the conventional double-Voigt method, no example is shown here.

7.4 Calculation of dislocation densities

The dislocation densities were calculated as described in section 5.5 with the modified Warren-Averbach and the contrast-corrected double-Voigt method. The differences of these two methods in short:

The modified Warren-Averbach method is an improvement of the conventional Warren-Averbach method by including the contrast of dislocations into the model. This reduces scattering of data points. Furthermore, the method directly puts out dislocation densities, not microstrains like the conventional Warren-Averbach method. Even though [27], [28] and [29] prove that the strain field of dislocations generally cannot be modeled by pure Gaussian shapes, this is assumed in this method.

The contrast-corrected double-Voigt method is basically a double-Voigt method including the contrast factor C (as suggested by T. Ungár for the WH- and WA methods). This includes microstrains $\langle \epsilon^2 \rangle$ as output and not, like the modified Warren-Averbach method, dislocation densities. Nevertheless, the microstrains can be converted to dislocation densities, e.g. by the Williamson-Smallman approach. Notice: Even though the modified Warren-Averbach method puts out dislocation densities, it implicitly first calculates microstrains, then converting them to dislocation densities assuming all microstrains are due to dislocations.

The plots to determine q (see section 5.1, especially fig. 8) are shown in the appendix on pages 72 to 73 in figures 69 to 75. The fact, that the considerable scattering introduces an uncertainty into q is accounted for in the following calculations. Nevertheless, q (if kept in physical meaningful boundaries¹²) for the measured material mainly influences the fit quality but not so much the microstrains/dislocation densities.

In fig. 34 the plot to calculate the dislocation density according to the modified Warren-Averbach method is done. The linear fit (as introduced in the modified Warren-Averbach method) is done for several L -values and the according dislocation densities are calculated. In the title, the dislocation density according to the contrast-corrected double-Voigt method is shown.

In [25], Ungár wrote: "The straight line fitted to small values of L gives the dislocation density and the effective outer cut-off radius of dislocations". This would correspond to the fits with $\rho = 3.3 * 10^{16} \text{ m}^{-2}$ ($L = 1$ to $L = 3 \text{ nm}$) and $\rho = 1.9 * 10^{16} \text{ m}^{-2}$ ($L = 1$ to $L = 10 \text{ nm}$). In [25] he used values $1 \text{ nm} < L < 9 \text{ nm}$.

Other authors ([1] and [2]) used larger L -values, as it was done with the conventional Warren-Averbach method (it was convention to compare microstrains at about $L = 80 \text{ nm}$). This corresponds to the fit with $\rho = 9.5 * 10^{14} \text{ m}^{-2}$ for $L = 40$ to $L = 80 \text{ nm}$ (in [1] $L = 40 - 80 \text{ nm}$ and in [2] $L = 40 - 90 \text{ nm}$ were used).

This is different for the contrast-corrected double-Voigt method, since it is basically a conventional double-Voigt method with a correction for the reflectivity contrast: The data points correspond exactly to the fit and thus the microstrains do not depend on the evaluated range of L .

Many authors (e.g. [9]) correctly conclude, that XRD-measurements like this are solely to determine trends in a series of measurements. One may think that neglecting the Cauchy part of the microstrains is not a problem in this case, if the same range of L is used for all measurements. This assumption only holds, if the Cauchy part of microstrains stays constant, which cannot be assumed *a priori*. If this Cauchy contribution does change the division through L [nm], especially for small values of L , again make the errors explode and can subsequently lead to false trends.

¹² q depends on the elastic constants of a crystal, thus the type of dislocations and the crystal structure. Calculated q values for iron are: $q = 1.2$ for pure edge- and $q = 2.8$ for pure screw dislocations [26]. The experimental determined value should be between those values in most cases.

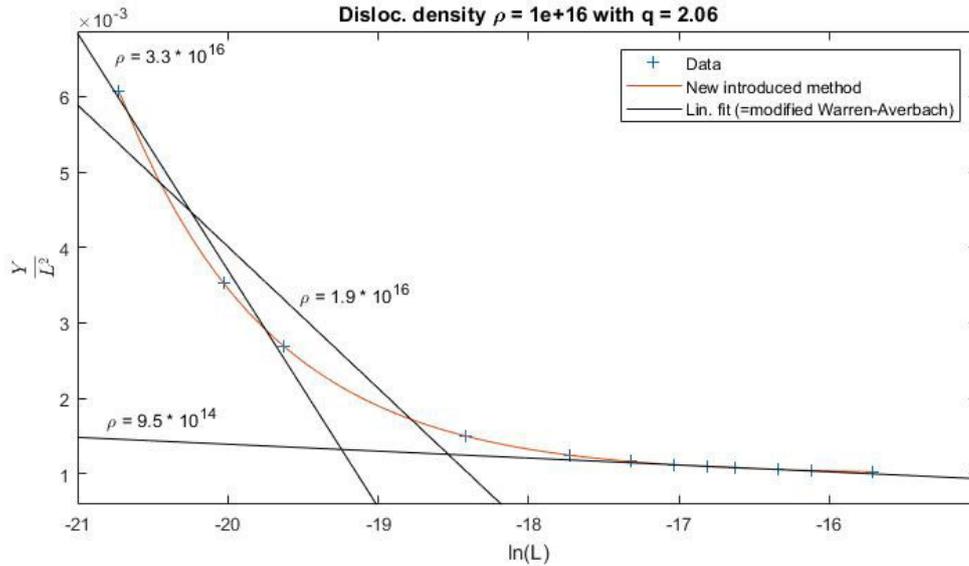


Figure 34: Example how the linear fit in the modified Warren-Averbach method introduces errors. All dislocation densities are given in $[\text{m}^{-2}]$

7.4.1 Calculation of dislocation densities according to the contrast-corrected double-Voigt method

Because of the problems with the modified Warren-Averbach method described in section 7.4, only the contrast-corrected double-Voigt method is used for data processing.

In figures 35 to 38 the modified Warren-Averbach plots of the un- and short tempered samples are shown (the modified Warren-Averbach plot is needed for the contrast-corrected double-Voigt method). The corresponding dislocation densities tabulated in tab. 4 should be seen relative to each other and not as absolute values. But note: The author expects the dislocation density of the untempered sample (P91_BH_vib_pol) to be too high, since the untempered martensite holds very high microstrains (e.g. due to carbon in the Fe octahedral sites) that do not come from dislocations. Nevertheless, the method converts all microstrains to dislocation density, leading to this high value.

Table 4: Dislocation densities of the un- and short tempered samples according to the contrast-corrected double-Voigt method

Sample	Dislocation density ρ $[\text{m}^{-2}]$
P91_BH_vib_pol	$6 \cdot 10^{16}$ ¹³
P91_0p166h_760C	$1 \cdot 10^{16}$
P91_0p5h_760C	$9 \cdot 10^{15}$
P91_1p5h_760C	$6 \cdot 10^{15}$

The modified Warren-Averbach plots of the long tempered samples are shown in figures 39 to 41. The dislocation densities for these samples are constant at $\rho \approx 5.5 \cdot 10^{15} \text{ m}^{-2}$ and are thus not tabulated. The author assumes this plateau is because the pure physically broadened integral breadth becomes too small compared to the instrumental broadening.

¹³The author expects this dislocation density to be wrong, since the untempered martensite holds very high microstrains (e.g. due to carbon in the Fe octahedral sites) that do not come from dislocations. Nevertheless, the method converts all microstrains to dislocation density.

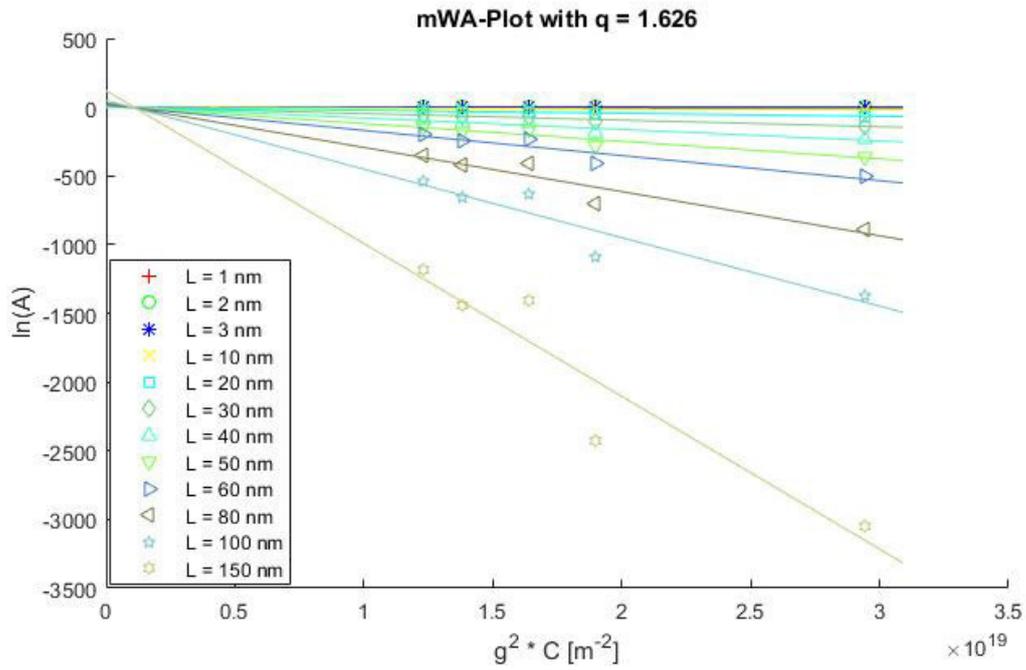


Figure 35: Modified Warren-Averbach plot of sample P91_BH_vib-pol

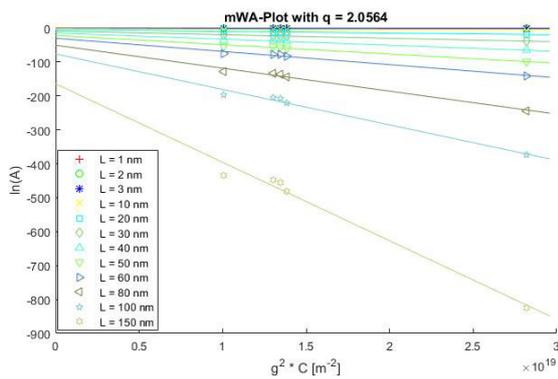


Figure 36: Modified Warren-Averbach plot of sample P91_0p166h.760C

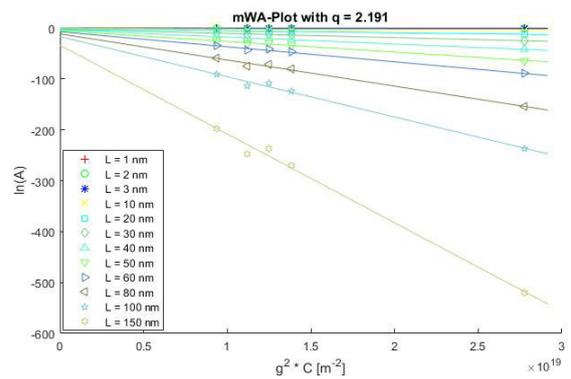


Figure 37: Modified Warren-Averbach plot of sample P91_0p5h.760C

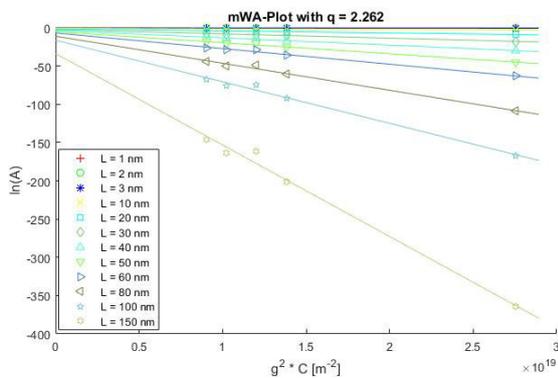


Figure 38: Modified Warren-Averbach plot of sample P91_1p5h.760C

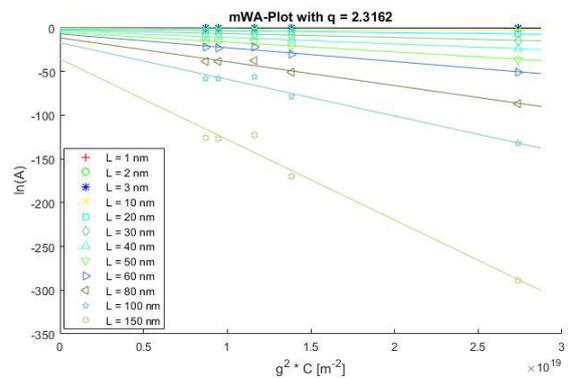


Figure 39: Modified Warren-Averbach plot of sample P91_12h.760C

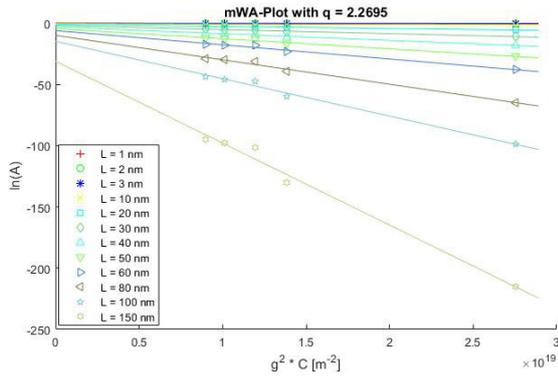


Figure 40: Modified Warren-Averbach plot of sample P91_24h_760C

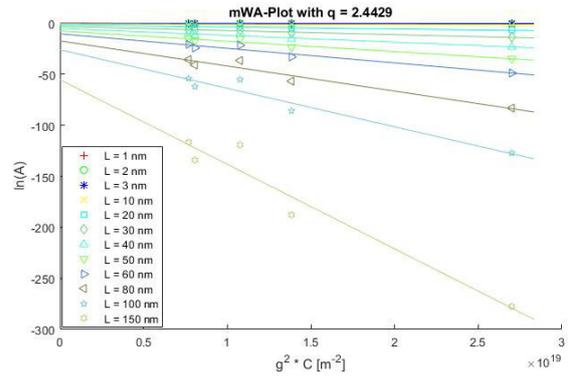


Figure 41: Modified Warren-Averbach plot of sample P91_36h_760C

7.5 Dilatometry

With the theory in section 4.9 and the data from fig. 24, the dislocation densities in fig. 42 are calculated. Since the sample length without any dislocations is not known, the length of the last data point (after more than 60 h) is taken instead. For the taken tempering times (up to 20 h), this introduces an error that is less than 10 %. Thus, the uncertainty by assuming all shrinkage is due to dislocation annihilation is much higher, especially because this assumption cannot be proven true.

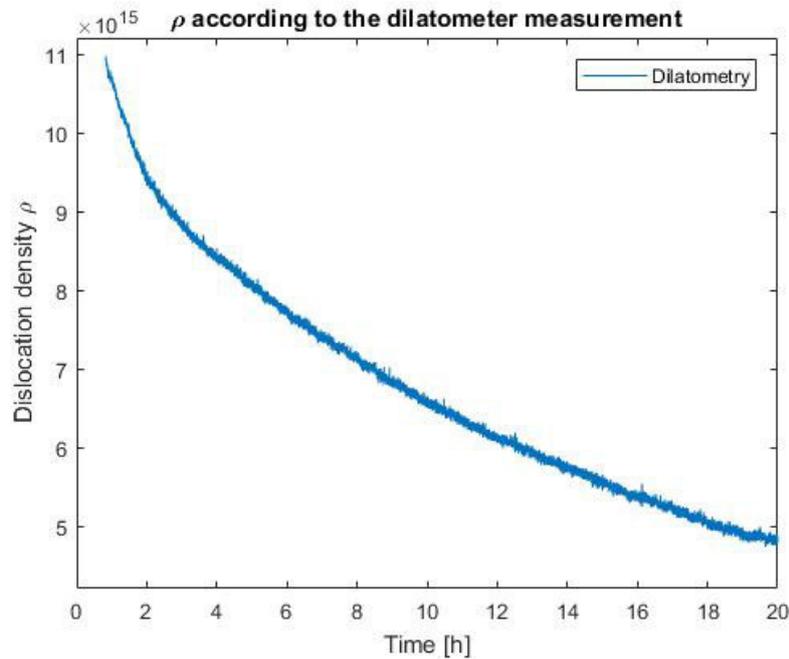


Figure 42: Dislocation densities according to the dilatometer measurement

Furthermore, the dislocation density for the first hour is not plotted since heating and precipitation naturally dominated the length change of the sample. To nevertheless achieve the dislocation density for this time frame, performing an exponential fit and extrapolating it to zero was tested. This was not successful since the data does not follow a simple exponential behavior, as shown on the next page.

7.6 Comparison of results

In fig. 43 the dislocation densities according to the dilatometer- and the XRD measurements are plotted.

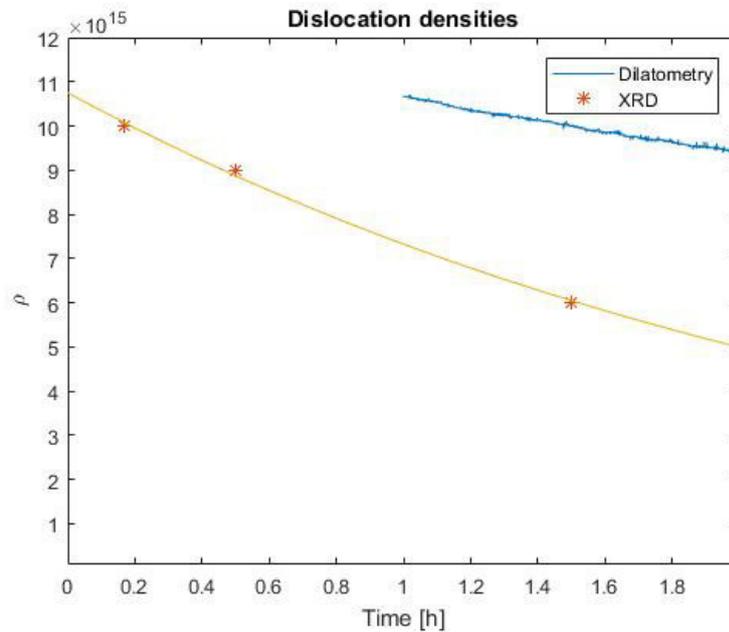


Figure 43: Comparison of dislocation densities according to dilatometry- and XRD measurements

The exponential fit for the three data points of the XRD measurements shows a nearly perfect exponential decay, but the decay is different from the dilatometer measurement. A closer look on the dilatometer measurement is done in fig. 44. By plotting $\ln(\rho)$ vs. time, all exponential decays appear as straight lines.

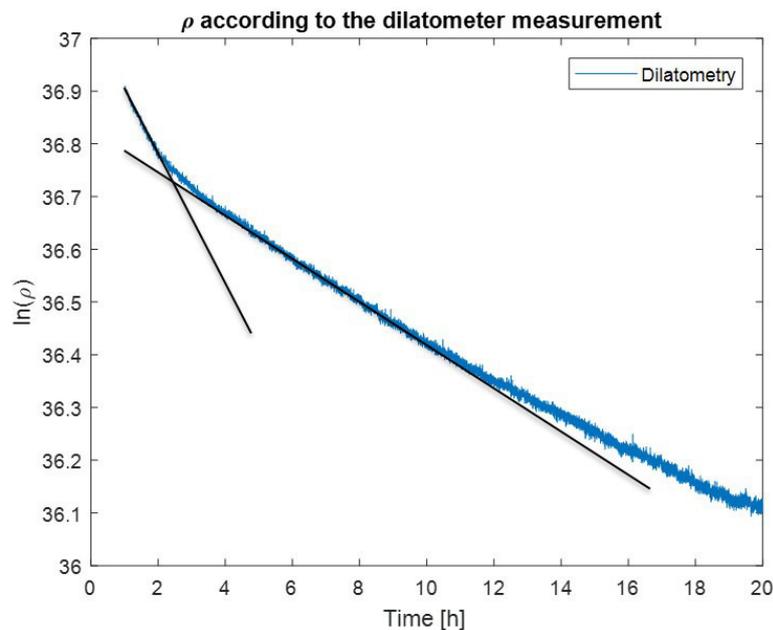


Figure 44: Plot of $\ln(\rho)$ according to dilatometry. Exponential decays appear as straight lines.

The two black lines indicate that two different exponential decays take place during the first 13 h. After that, a peak, most likely because of some late precipitation, deviates from the exponential decay.

This way, the dilatometer measurements don't prove the XRD measurements are wrong, but it is just not possible to correlate them because too many different phenomena take place simultaneously in the dilatometer. Another problem is that the time frame that is measurable by both techniques is very small. This is because there is no chance to measure dislocation densities with the dilatometer while precipitation takes place (measurement of ρ for short tempering times is not possible), but the XRD machine that was used (in this setup) is limited to very high dislocation densities (measurement of ρ for long tempering times is not possible).

Nevertheless comparing these two methods to see if they are in the same order of magnitude:

After 1.5 h tempering at 760 °C the dislocation density according to the dilatometer is $\approx 10^{16} \text{ m}^{-2}$, that is 5/3 times the dislocation density according to XRD ($=6 \cdot 10^{15} \text{ m}^{-2}$). Keeping in mind, that the used XRD method measures dislocations in the grains only, whereas dilatometry gives total dislocation densities, this is a good value.

Testing if the data achieved with XRD can be confirmed by positron annihilation measurements is planned for a future project.

8 Conclusion

Aim of this work was to test the capabilities of XRD dislocation density measurements for complex martensitic steels. A summary of the data processing methods discussed is shown in tab. 5.

Table 5: Comparison of data analysis methods

D ... "Apparent" crystallite size according to Scherrer

$\langle D \rangle$...surface-weighted average domain size according to Bertaud; Warren and Averbach

η ...(Upper-limit) Microstrain according to Stokes and Wilson

$\langle \epsilon^2 \rangle$...Mean-square-strain

ρ ...Dislocation density

Method	Data scattering	Output	Cauchy part included in microstrains	Achieved data quality
Williamson-Hall (WH)	huge	D, η	X	only for very rough comparisons
Warren-Averbach (WA)	high	$\langle D \rangle, \langle \epsilon^2 \rangle$	X (\checkmark^{14})	much better than WH
Modified Williamson-Hall	considerable	D, ρ	X	compared to WH, scattering is strongly reduced
Modified Warren-Averbach	small	$\langle D \rangle, \rho$	X	good, but Cauchy part of microstrains is still excluded
Double-Voigt	high	$\langle D \rangle, \langle \epsilon^2 \rangle$	\checkmark	scattering is the biggest problem
Contrast-corrected double-Voigt	small	$\langle D \rangle, \langle \epsilon^2 \rangle$	\checkmark	good

By introducing the contrast factor C into the double-Voigt method, it was possible to show how big the error to neglect the Cauchy part of microstrains in the modified Warren-Averbach method really is. Even when just comparing trends in a series of measurements (as it should be done in general), checking if the Cauchy part of microstrains remains constant is strongly advised. Otherwise a change in the Cauchy part might falsely change the calculated microstrains, leading to a false "trend".

The dislocation densities of a P91 steel after different tempering times then got evaluated with the contrast-corrected double-Voigt method and a dilatometry measurement. Both methods show decays and the dislocation density fraction ρ_{dil}/ρ_{XRD} is $\approx 5/3$, but the dilatometer measurement did not show the expected behavior. Instead of one exponential decay because of dislocation annihilation, at least two different exponential functions in the first 13 h were discovered. Furthermore, the used XRD machine setup only allows to measure very high dislocation densities, so the time frame covered by both measurement methods is very small.

So even though an XRD method that also includes the Cauchy part of microstrains got improved by greatly reducing the scattering of data points and the results look promising, the method is limited to very high dislocation densities with the available XRD machine in the used setup. Furthermore, it was not possible to fully confirm the achieved data with a complementary technique (dilatometry).

¹⁴Although the original Warren-Averbach method did not include the Cauchy part into the microstrains, the equation itself can be remodeled, so the Cauchy part is included

List of Figures

1	Example for the spectrum of a molybdenum anode	2
2	Example for a Coolidge tube	3
3	Visualization of Bragg's law	3
4	General setup of an X-ray diffractometer	4
5	Picture of the half-width-at-half-maximum	5
6	Example for a Williamson-Hall plot	10
7	Example for a Warren-Averbach plot	11
8	Determination of q	15
9	Determination of the dislocation density	15
10	Determination of β_{CD} and β_{GD}^2	17
11	Corrected modified Warren-Averbach method	18
12	Flowchart symbols	21
13	Flowchart of raw data processing routine	22
14	Flowchart of peak fitting routine	23
15	Flowchart to explain the combination of mWH with mWA and double-Voigt	24
16	Measurement of the LaB ₆ standard	25
17	Measurement of sample P91_BH_vib_pol	26
18	Measurement of sample P91_0p166h_760C	27
19	Measurement of sample P91_0p5h_760C	27
20	Measurement of sample P91_1p5h_760C	28
21	Measurement of sample P91_12h_760C	28
22	Measurement of sample P91_24h_760C	29
23	Measurement of sample P91_36h_760C	29
24	Dilatometry measurement under Argon atmosphere	30
25	Example for a linear plot of all 4 fits	31
26	Example for a semi-logarithmic plot to check $K_{\alpha 1} / K_{\alpha 2}$ separation, background and $K_{\alpha 1}$ peak maximum	32
27	Example for a linear plot (4 fits) with truncated peak	33
28	Example for a semi-logarithmic plot with truncated peak	33
29	Williamson-Hall plot of sample P91_0p166h_760C	34
30	Example of a modified Williamson-Hall plot	35
31	Modified Williamson-Hall plot of sample P91_0p166h_760C	35
32	Warren-Averbach plot of sample P91_0p166h_760C	36
33	Modified Warren-Averbach plot of sample P91_0p166h_760C	37
34	Example for problems with the modified Warren-Averbach method	39
35	Modified Warren-Averbach plot of sample P91_BH_vib_pol	40
36	Modified Warren-Averbach plot of sample P91_0p166h_760C	40
37	Modified Warren-Averbach plot of sample P91_0p5h_760C	40
38	Modified Warren-Averbach plot of sample P91_1p5h_760C	40
39	Modified Warren-Averbach plot of sample P91_12h_760C	40
40	Modified Warren-Averbach plot of sample P91_24h_760C	41
41	Modified Warren-Averbach plot of sample P91_36h_760C	41
42	Dislocation densities according to the dilatometer measurement	41
43	Comparison of dislocation densities according to dilatometry- and XRD measurements	42
44	Plot of $\ln(\rho)$ according to dilatometry	42
69	Determination of q for sample P91_BH_vib_pol	72
70	Determination of q for sample P91_0p166h_760C	72
71	Determination of q for sample P91_0p5h_760C	72

72	Determination of q for sample P91_1p5h_760C	73
73	Determination of q for sample P91_12h_760C	73
74	Determination of q for sample P91_24h_760C	73
75	Determination of q for sample P91_36h_760C	73

Image Source Directory

- [Fig.1] R. Guinebretière. *Instrumentation used for X-ray Diffraction*, page 40. ISTE, 2010. Fig. 2.1
- [Fig.2] R. Guinebretière. *Instrumentation used for X-ray Diffraction*, page 47. ISTE, 2010. Fig. 2.6
- [Fig.3] S. Baskaran. Structure and regulation of yeast glycogen synthase, page 28. 01 2010. Fig. 12
- [Fig.6] From <http://pd.chem.ucl.ac.uk/pdnn/peaks/sizedet.htm>; axis labeling was changed; 03.03.2017.
- [Fig.9] Z. Cong and Y. Murata. Dislocation density of lath martensite in 10Cr – 5W heatresistant steels. *Materials Transactions*, 52(12):2151-2154, 2011. Fig. 4
- [Fig.30] P. Scardi, M. Leoni, and R. Delhez. Line broadening analysis using integral breadth methods: A critical review. 37:381-390, 06 2004. Fig. 2. Axis labeling was changed.

References

- [1] T. Kunieda, M. Nakai, Y. Murata, T. Koyama, and M. Morinaga. Estimation of the system free energy of martensite phase in an fe-cr-c ternary alloy. *ISIJ International*, 45(12):1909–1914, 2005.
- [2] Z. Cong and Y. Murata. Dislocation density of lath martensite in 10cr-5w heat-resistant steels. *Materials Transactions*, 52(12):2151–2154, 2011.
- [3] P. Scherrer. Bestimmung der gröÙe und der inneren struktur von kolloidteilchen mittels röntgenstrahlen. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, 2, 1918.
- [4] A. R. Stokes and A. J. C. Wilson. The diffraction of x rays by distorted crystal aggregates - i. *Proceedings of the Physical Society*, 56(3):174, 1944.
- [5] G.K. Williamson and W.H. Hall. X-ray line broadening from filed aluminium and wolfram. *Acta Metallurgica*, 1(1):22 – 31, 1953.
- [6] D. Balzar and S. Popović. Reliability of the Simplified Integral-Breadth Methods in Diffraction Line-Broadening Analysis. *Journal of Applied Crystallography*, 29(1):16–23, Feb 1996.
- [7] B. E. Warren. X-ray studies of deformed metals. *Progress in Metal Physics*, 8, 1959. <http://www.sciencedirect.com/science/article/pii/0502820559900152> ; 6.03.2017.
- [8] T. Ungár. Strain broadening caused by dislocations. Technical report, Institute for General Physics, Eotvos University Budapest, H-1445 Muzeum krt. 6-8, Budapest VIII, P.O.B. 323, Hungary, 1997. Published at JCPDS-International Centre for Diffraction Data.
- [9] P. Scardi, M. Leoni, and R. Delhez. Line broadening analysis using integral breadth methods: A critical review. 37:381–390, 06 2004.

- [10] H. M. Rietveld. A profile refinement method for nuclear and magnetic structures. *Journal of Applied Crystallography*, 2(2):65–71, Jun 1969.
- [11] A. Borbély and I. Groma. Variance method for the evaluation of particle size and dislocation density from x-ray bragg peaks. *Applied Physics Letters*, 79(12):1772–1774, 2001.
- [12] R. Guinebretière. *Instrumentation used for X-ray Diffraction*, pages 39–126. ISTE, 2010.
- [13] Y. Leng. *X-Ray Diffraction Methods*, pages 47–82. Wiley-VCH Verlag GmbH and Co KGaA, 2013.
- [14] R. Guinebretière. *Data Processing, Extracting Information*, pages 127–153. ISTE, 2010.
- [15] W. H. Bragg and W. L. Bragg. The reflection of x-rays by crystals. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 88(605):428–438, 1913.
- [16] R. Guinebretière. *Kinematic and Geometric Theories of X-Ray Diffraction*, pages 1–38. ISTE, 2010.
- [17] D. Taupin. Automatic peak determination in X-ray powder patterns. *Journal of Applied Crystallography*, 6(4):266–273, Aug 1973.
- [18] D. Balzar and H. Ledbetter. Voigt-function modeling in Fourier analysis of size- and strain-broadened X-ray diffraction peaks. *Journal of Applied Crystallography*, 26(1):97–103, Feb 1993.
- [19] A. R. Stokes. A numerical fourier-analysis method for the correction of widths and shapes of lines on x-ray powder photographs. *Proceedings of the Physical Society*, 61(4):382, 1948.
- [20] B.E. Warren. *X-ray diffraction*, pages 251-314. Addison Wesley, 1969.
- [21] B. E. Warren and B. L. Averbach. The effect of cold-work distortion on x-ray patterns. *Journal of Applied Physics*, 21(6):595–599, 1950.
- [22] B. E. Warren and B. L. Averbach. The separation of cold-work distortion and particle size broadening in x-ray patterns. *Journal of Applied Physics*, 23(4):497–497, 1952.
- [23] B. Oberdorfer, B. Lorenzoni, K. Unger, W. Sprengel, M. Zehetbauer, R. Pippan, and R. Würschum. Absolute concentration of free volume-type defects in ultrafine-grained fe prepared by high-pressure torsion. *Scripta Materialia*, 63(4):452 – 455, 2010.
- [24] J. P. Hirth and J. Lothe. *Theory of Dislocations*. John Wiley & Sons, New York, 2. edition, 1982.
- [25] T. Ungár, I. Dragomir-Cernatescu, D. Louër, and N. Audebrand. Dislocations and crystallite size distribution in nanocrystalline ceo₂ obtained from an ammonium cerium(iv)-nitrate solution. *Journal of Physics and Chemistry of Solids*, 62(11):1935 – 1941, 2001.
- [26] T. Ungár, I. Dragomir, Á. Révész, and A. Borbély. The contrast factors of dislocations in cubic crystals: the dislocation model of strain anisotropy in practice. *Journal of Applied Crystallography*, 32(5):992–1002, Oct 1999.
- [27] C. R. Houska and T. M. Smith. Least-squares analysis of x-ray diffraction line shapes with analytic functions. *Journal of Applied Physics*, 52(2):748–754, 1981.
- [28] T. Adler and C. R. Houska. Simplifications in the x-ray line-shape analysis. *Journal of Applied Physics*, 50(5):3282–3287, 1979.
- [29] W. Yuming, L. Shansan, and L. Yenchin. X-ray line profile analysis of deformed Al. *Journal of Applied Crystallography*, 15(1):35–38, Feb 1982.

-
- [30] G. K. Williamson and R. E. Smallman. Iii. dislocation densities in some annealed and cold-worked metals from measurements on the x-ray debye-scherrer spectrum. *Philosophical Magazine*, 1(1):34–46, 1956.
- [31] Z. Matej and M. Dpita. <http://www.xray.cz/xrdmlread/Add-ons.htm>.
- [32] A. Fuchs, R. Resel, and B. Sonderegger. Report for project laboratory; not published.
- [33] I. Groma, T. Ungár, and M. Wilkens. Asymmetric X-ray line broadening of plastically deformed crystals. I. Theory. *Journal of Applied Crystallography*, 21(1):47–54, Feb 1988.
- [34] I. Groma. X-ray line broadening due to an inhomogeneous dislocation distribution. *Phys. Rev. B*, 57:7535–7542, Apr 1998.

9 Appendix

9.1 XRD measurement specifications

Data Viewer



LaB6_36-58_SLD_mono.xrml

Report

Created on: 16.10.2017 13:18:30

Operator: arthur

File

File name: W:\arthur_fuchs\Diplomarbeit\Rohdaten\LaB6_36-58_SLD_mono.xrml

Checksum: 87C80C2A05993F2D4534DD5E1CB595EB

File size (bytes): 13704

Time stamp

Started: 29.04.2017 16:47:08

Finished: 30.04.2017 16:27:21

Total time (h:m:s): 23:40:13

Comment

Configuration=Chi-Phi-Z-stage, Owner=User-1, Creation date=10.12.2010 13:37:09
Goniometer=Theta/Theta; Minimum step size 2Theta:0.0001; Minimum step size Omega:0.0001
Sample stage=Chi-Phi-z stage 240 mm (reflection); Minimum step size Phi:0.01; Minimum step size Chi:0.01; Minimum step size Z:0.001
Diffractometer system=EMPYREAN
Measurement program=C:\XRD Data\Arthur Fuchs\Programme\After_XRDGonioScanning1D_Pulver_BeamAttenuatorOpenForStandby.xrmp, Identifier={546C1ED7-DD1C-4D4A-93A3-096565A3964A}
Fine Calibration Offset for 2Theta = 0.0182 deg
Batch program=C:\XRD Data\Arthur Fuchs\Programme\Batch_XRD_standby.xrmp, Identifier={0426952A-C9D6-451D-A964-130B5DD96181}

Status

Completed

Sample

Sample type: To be analyzed

Measurement type

Single scan

Sample offsets

Omega (°): 0,0439

2Theta (°): -0,0050

LaB6_36-58_SLD_mono.xrdml

Used wavelength

Intended wavelength type:	K α
K α_1 (Å):	1,540598
K α_2 (Å):	1,544426
K α_2 /K α_1 intensity ratio:	0,50
K α (Å):	1,541874
K β (Å):	1,392250

Incident beam path

Radius (mm): 240,0

X-ray tube

Name:	Empyrean Cu LFF HR (9430 033 7310x) DK350568
Anode material:	Cu
Voltage (kV):	40
Current (mA):	40
Focus	
Focus type:	Line
Length (mm):	12,0
width (mm):	0,4
Take-off angle (°):	6,0

Soller slit

Name:	Soller slits 0.02 rad.
Opening (rad.):	0,02

Mask

Name:	Fixed mask 10 mm 15 mm
Distance to sample (mm):	140
Width (mm):	10,00

Divergence slit

Name:	Fixed slit 1/8°
Distance to sample (mm):	140
Type:	Fixed
Height (mm):	0,19
Angle (°):	0,1089

LaB6_36-58_SLD_mono.xrdml

Diffracted beam path

Radius (mm): 240,0**Anti-scatter slit**

Name: AS slit ~~8.0~~ **8,3 mm** (PIXcel)
Type: Fixed
Height (mm): 8,00

Soller slit

Name: Soller slits 0.02 rad.
Opening (rad.): 0,02

Monochromator

Name: 1D Standard diffracted beam monochr. Cu
Crystal:
Name: C
Type: Symmetric
Shape: Flat
No. of reflections: 1
hkl: 0 0 2

Detector

Name: PIXcel3D 1x1 detector
Type: Area detector
PHD - Lower level (%): 25,0
PHD - Upper level (%): 100,0
Mode: Scanning
No. of active channels: 255
Pitch (mm): 0,0550
Active length (°): 3,347

Source

Created by: Panalytical
Application SW: Data Collector
vs. 4.4a
Instrument control SW: EMPYREAN
vs. 7.6 20140701

LaB6_36-58_SLD_mono.xrdml

Instrument ID: 0000000011087517

Sample mode

Reflection

Scan

Start time stamp: 29.04.2017 16:47:08
End time stamp: 30.04.2017 16:27:21
Scan axis: Gonio
Scan range (°): 35,7967 - 58,0000
Start position (°): 35,8033
End position (°): 57,9934
Step size (°): 0,0131
No. of points: 1691
Scan mode: Continuous
Phi (°): 0,00
Chi (°): 0,20
Z (mm): 6,979
Counting time (s): 11158,545

Note: This is an automatic generated report. Since some used slits couldn't be set correctly in the software, they had to be corrected by hand in this report.

9.2 Determining γ in the modified Warren-Averbach method

For steels and the used experimental setup, γ can be determined by evaluating the (200) plane. In this case $H^2 = 0$ and eq. 36 reduces to

$$\frac{(\Delta g)^2 - \alpha}{g^2} \cong \gamma C_{h00}$$

$$\rightarrow \gamma \cong \frac{(\Delta g)^2 - \alpha}{g^2 C_{h00}}$$

and γ can be determined.

9.3 Plots of data fits

In the following, the plots for all fitted data are shown.

9.3.1 LaB₆ standard

In the following, the fits for the LaB₆ standard are shown. When the fits were done, the program was still in German. Afterwards, the language was switched and the same plots for all later measurements are shown in English.

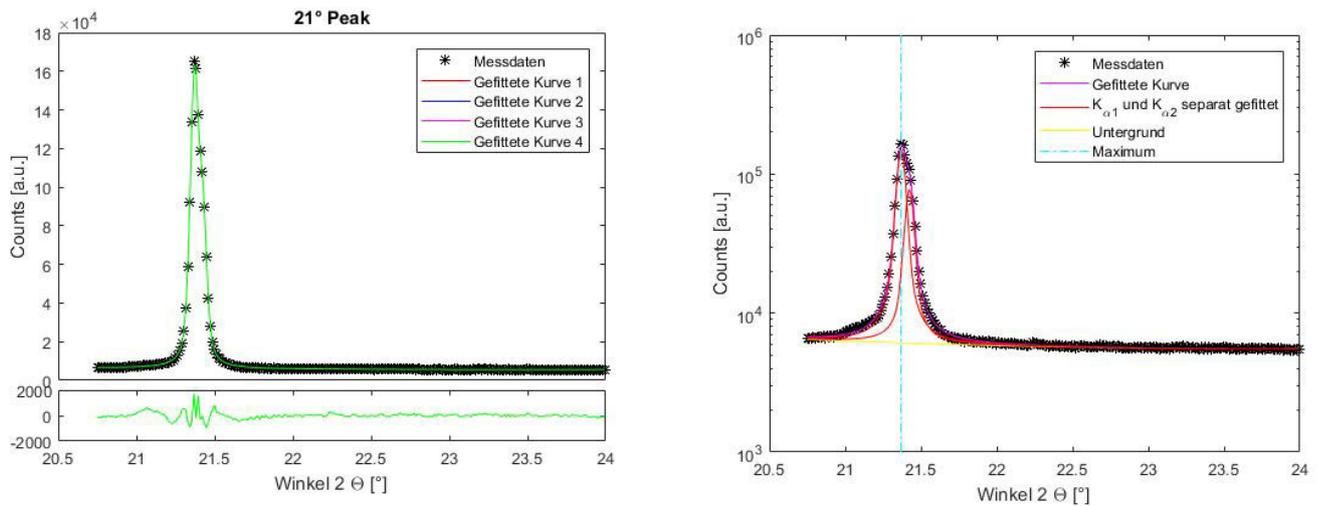


Figure 45: Linear- (left) and semi-logarithmic (right) fits of the 21 ° LaB₆ peak

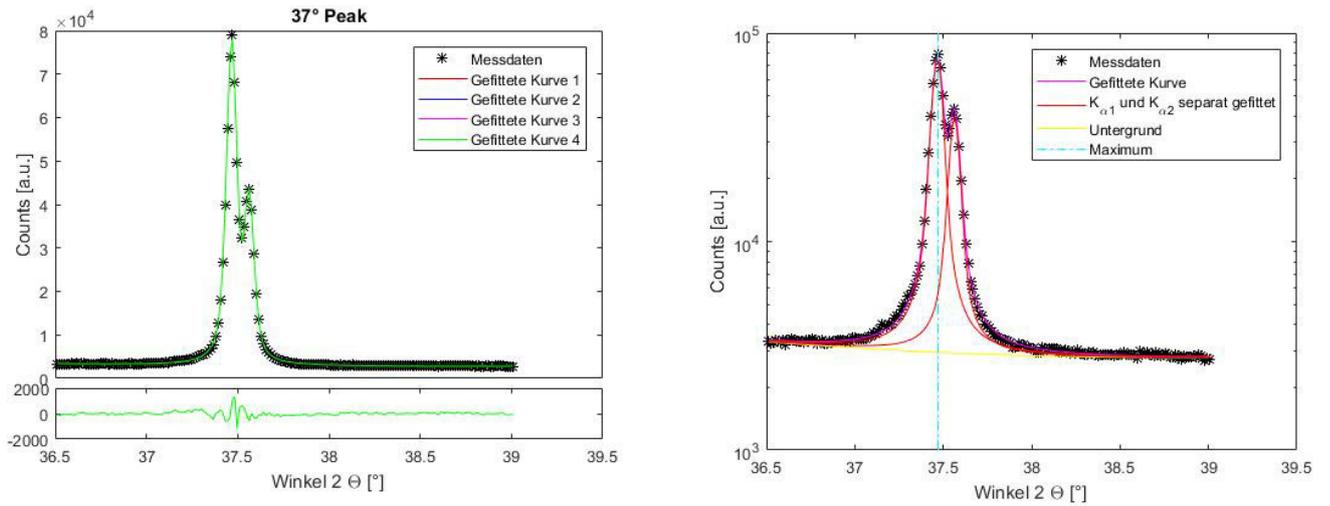


Figure 46: Linear- (left) and semi-logarithmic (right) fits of the 37 ° LaB₆ peak

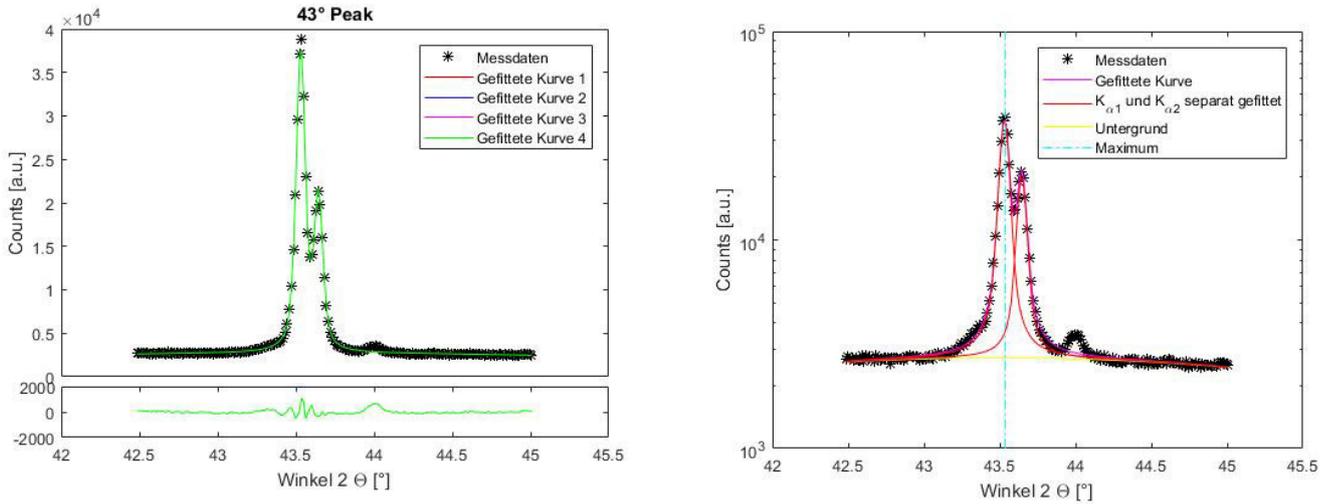


Figure 47: Linear- (left) and semi-logarithmic (right) fits of the 43 ° LaB₆ peak

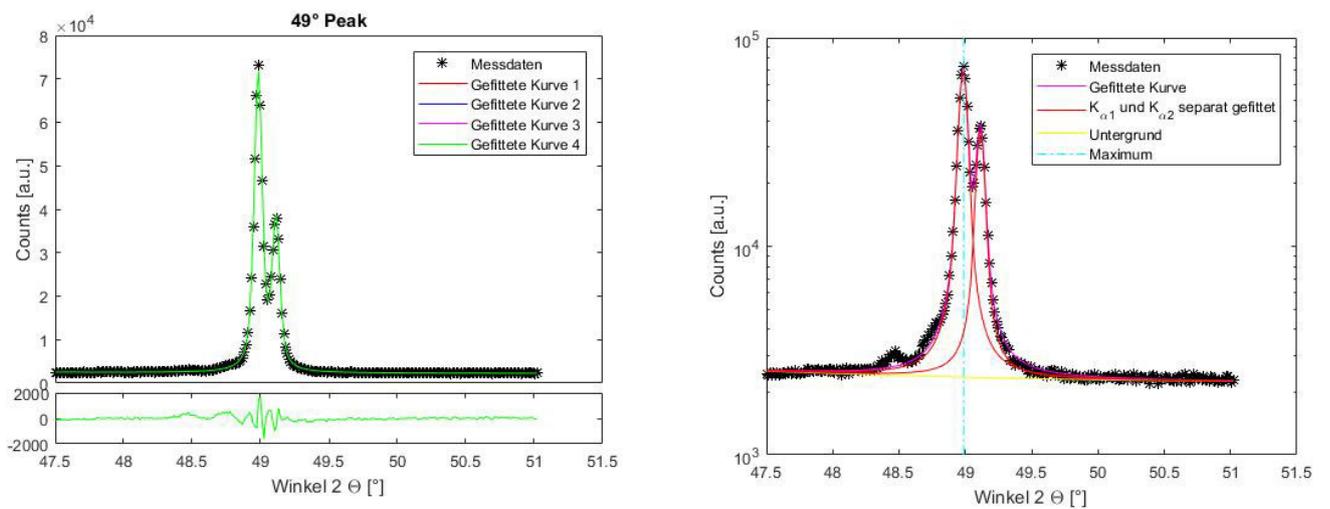


Figure 48: Linear- (left) and semi-logarithmic (right) fits of the 49 ° LaB₆ peak

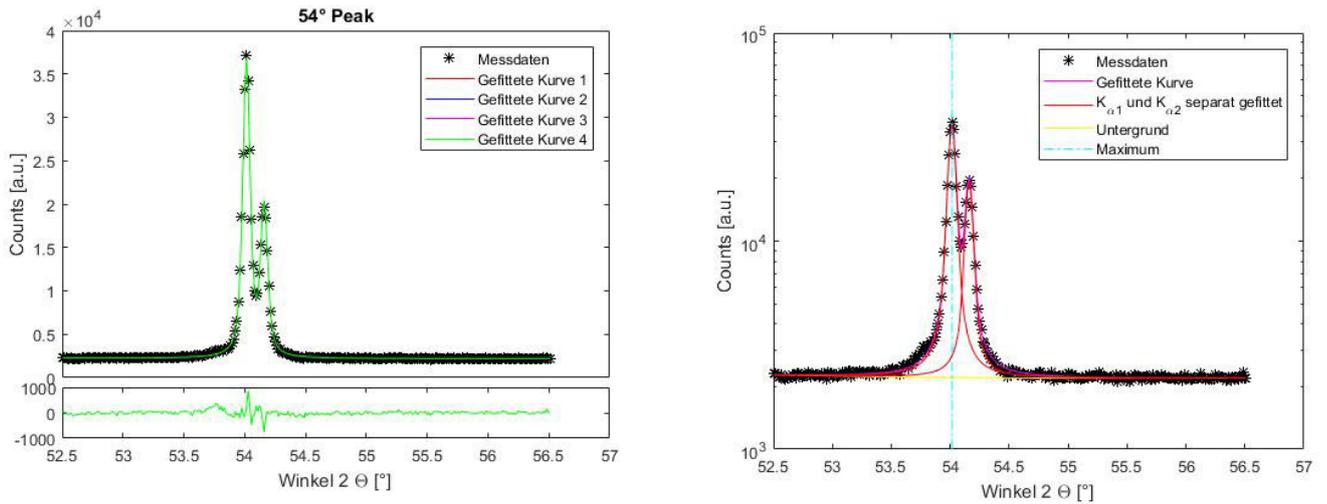


Figure 49: Linear- (left) and semi-logarithmic (right) fits of the 54 ° LaB₆ peak

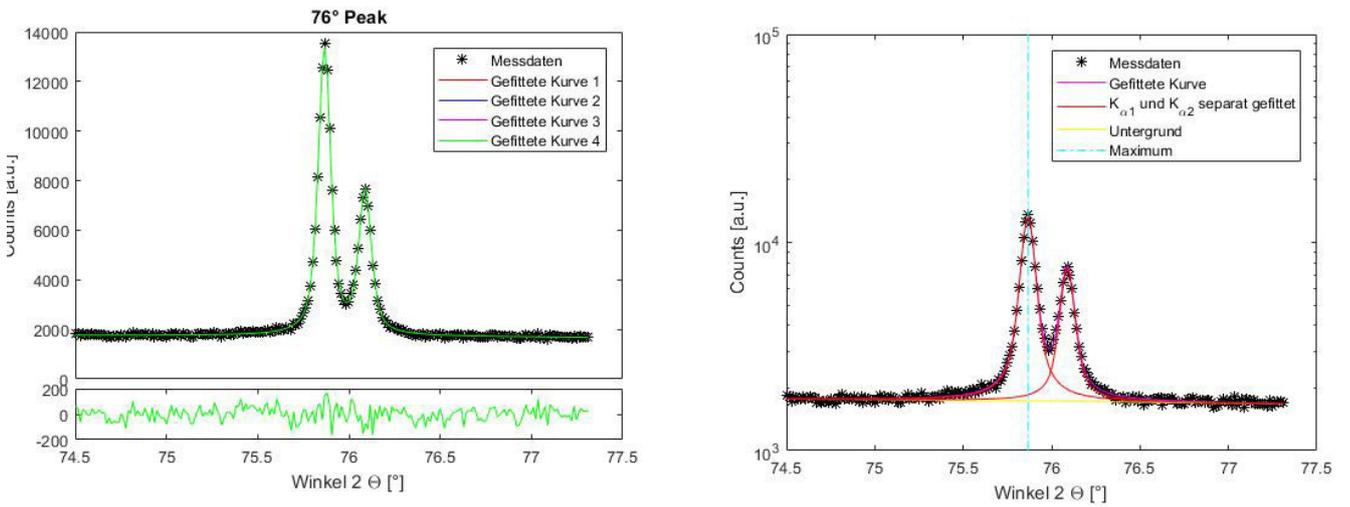


Figure 50: Linear- (left) and semi-logarithmic (right) fits of the 76 ° LaB₆ peak

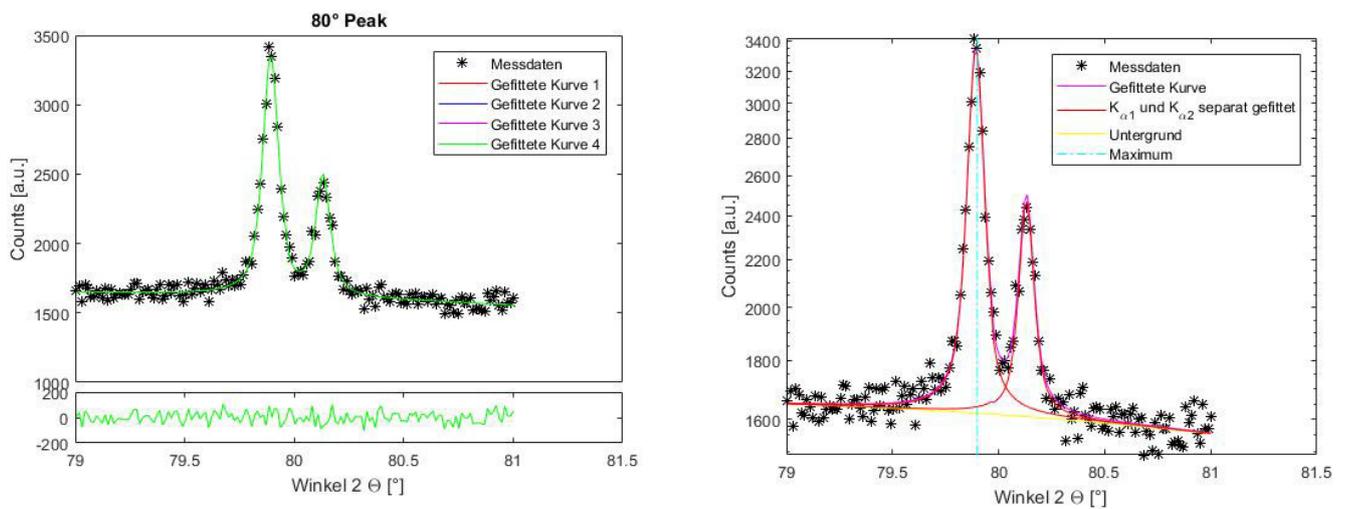


Figure 51: Linear- (left) and semi-logarithmic (right) fits of the 80 ° LaB₆ peak

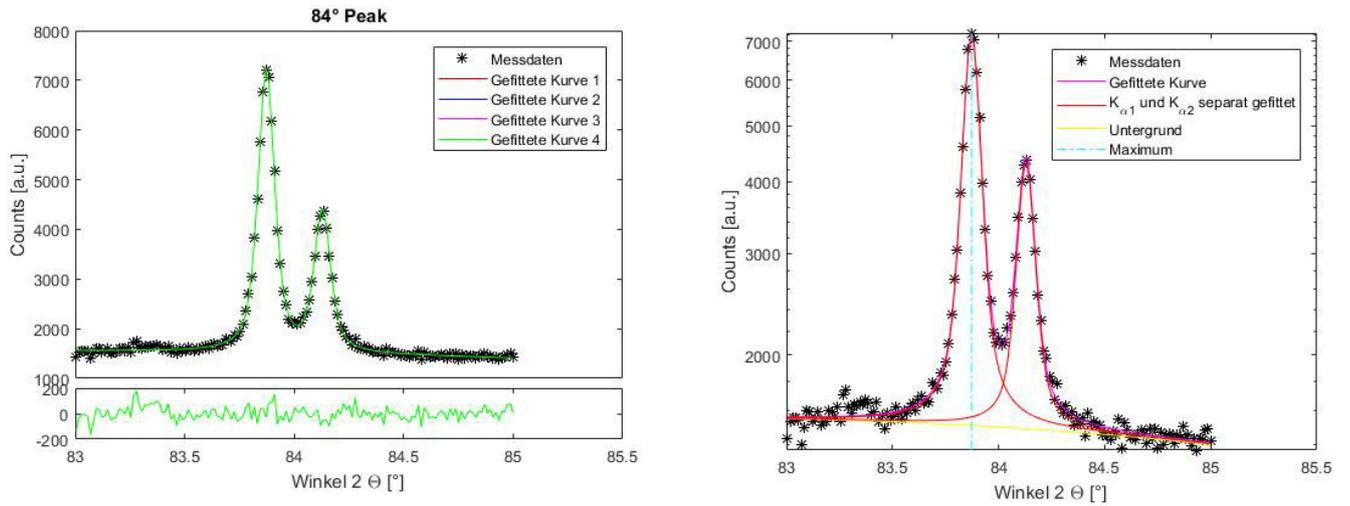


Figure 52: Linear- (left) and semi-logarithmic (right) fits of the 84 ° LaB₆ peak

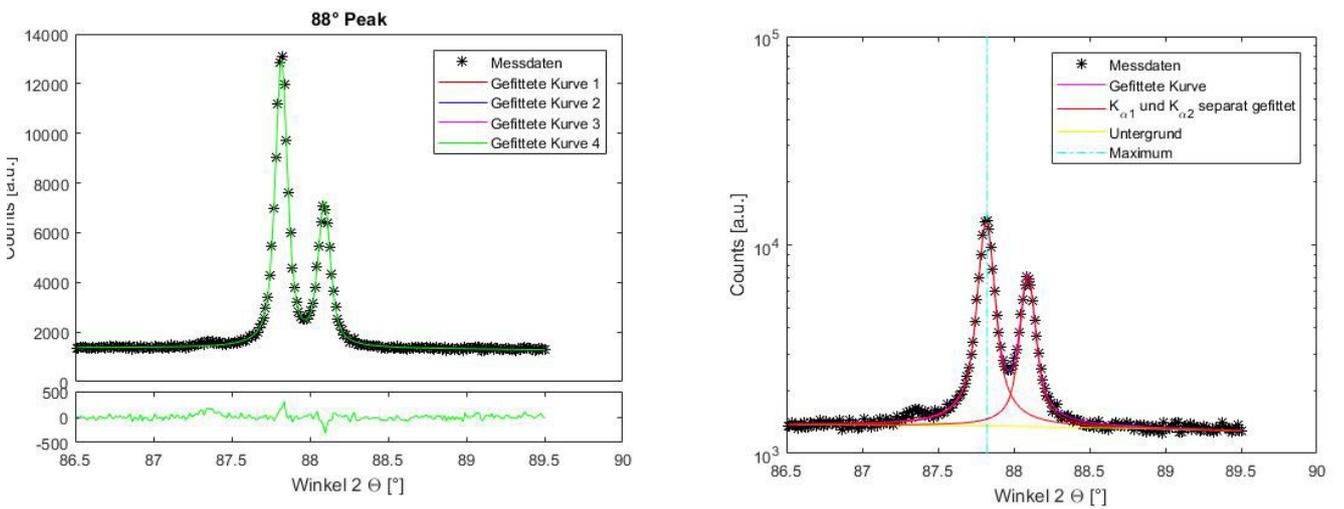


Figure 53: Linear- (left) and semi-logarithmic (right) fits of the 88 ° LaB₆ peak

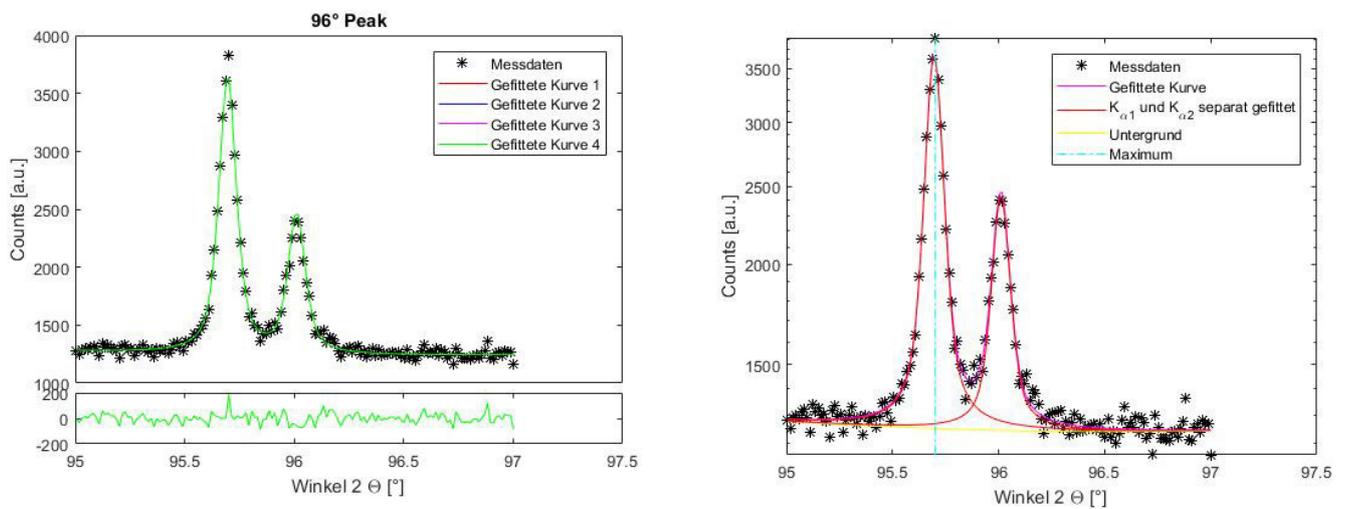


Figure 54: Linear- (left) and semi-logarithmic (right) fits of the 96 ° LaB₆ peak

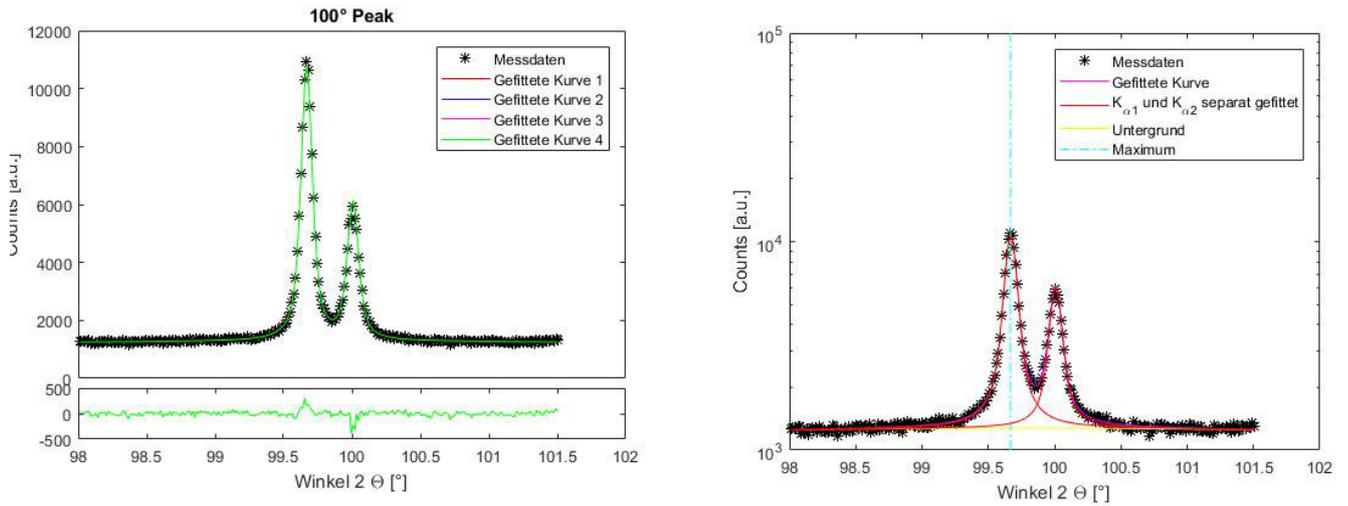


Figure 55: Linear- (left) and semi-logarithmic (right) fits of the 100 ° LaB₆ peak

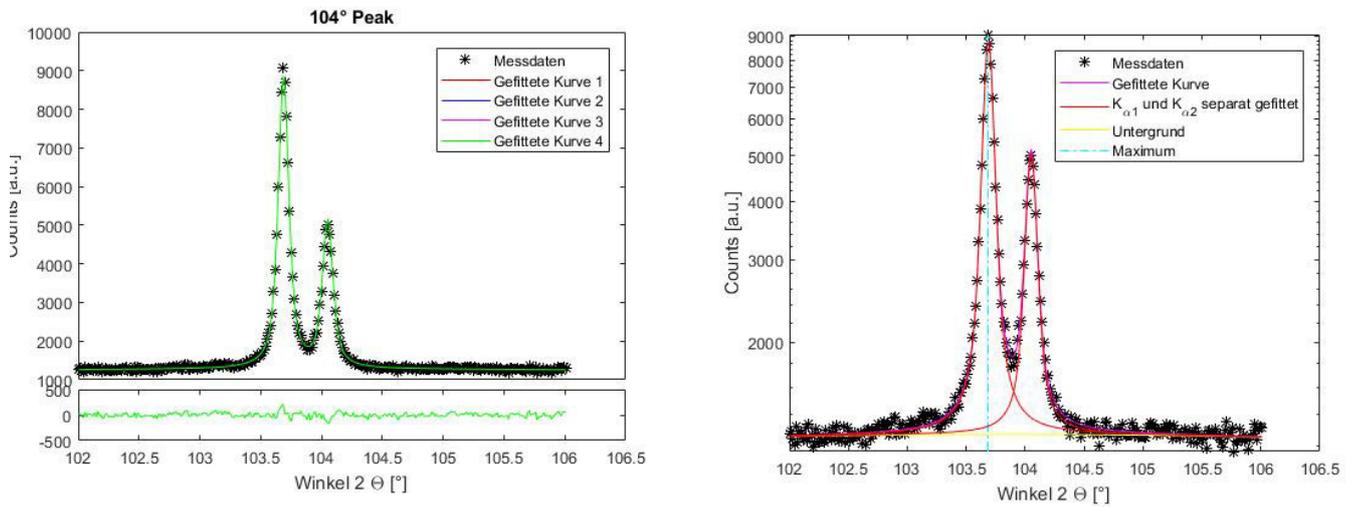


Figure 56: Linear- (left) and semi-logarithmic (right) fits of the 104 ° LaB₆ peak

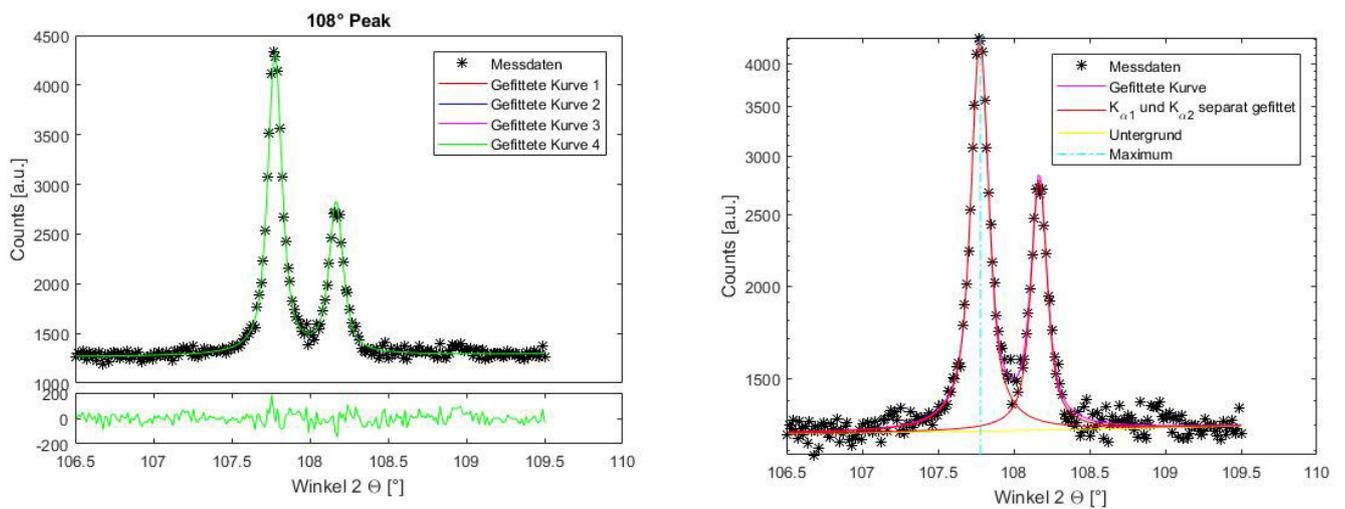


Figure 57: Linear- (left) and semi-logarithmic (right) fits of the 108 ° LaB₆ peak

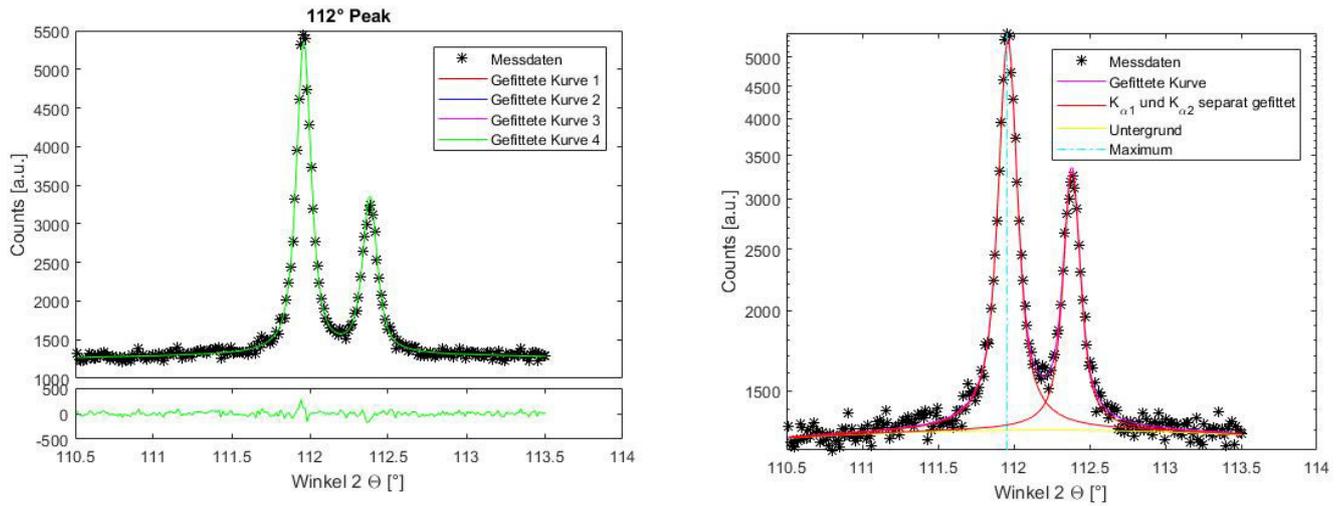


Figure 58: Linear- (left) and semi-logarithmic (right) fits of the 112 ° LaB₆ peak

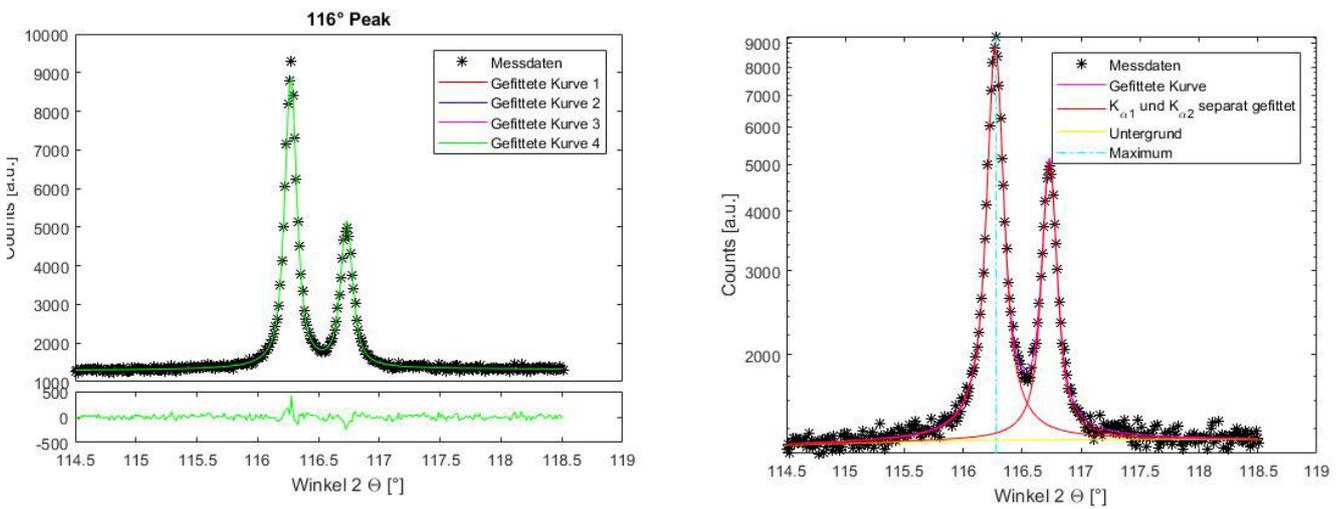


Figure 59: Linear- (left) and semi-logarithmic (right) fits of the 116 ° LaB₆ peak

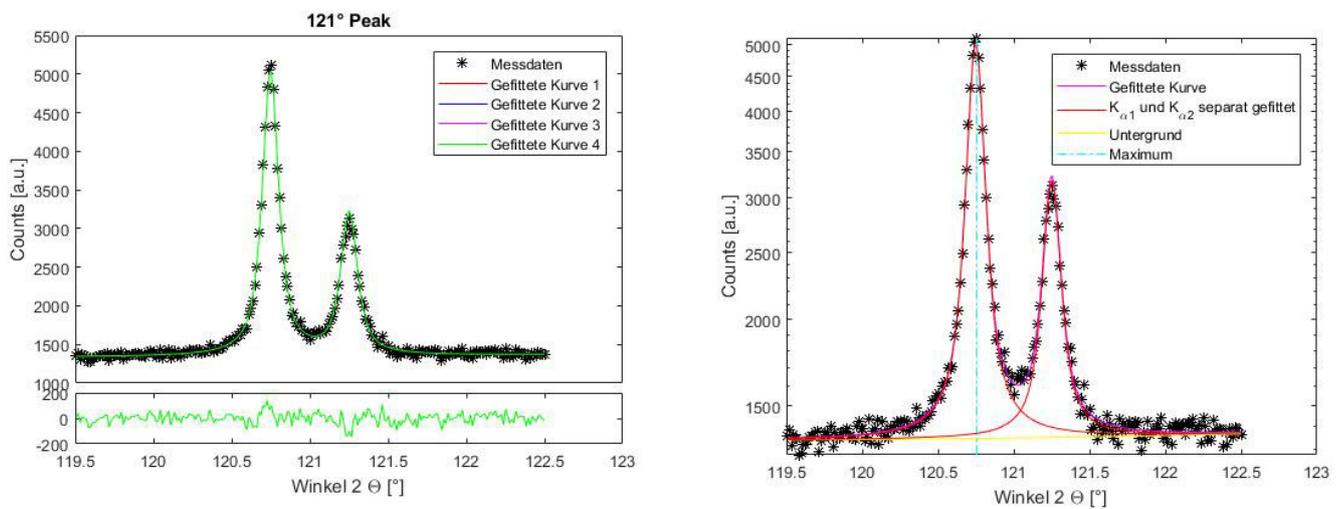


Figure 60: Linear- (left) and semi-logarithmic (right) fits of the 121 ° LaB₆ peak

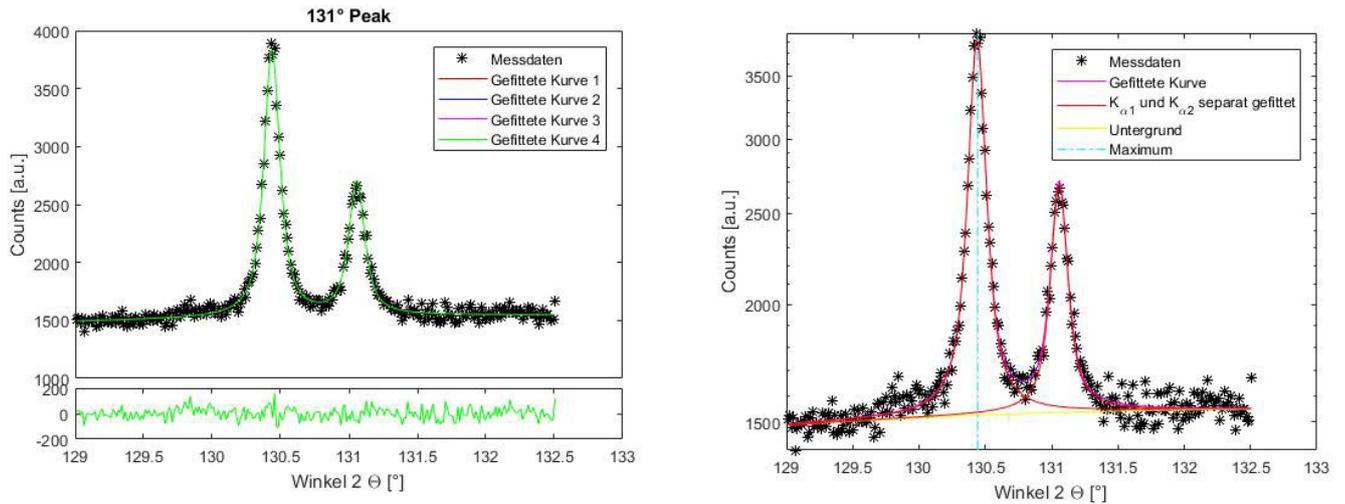


Figure 61: Linear- (left) and semi-logarithmic (right) fits of the 131 ° LaB₆ peak

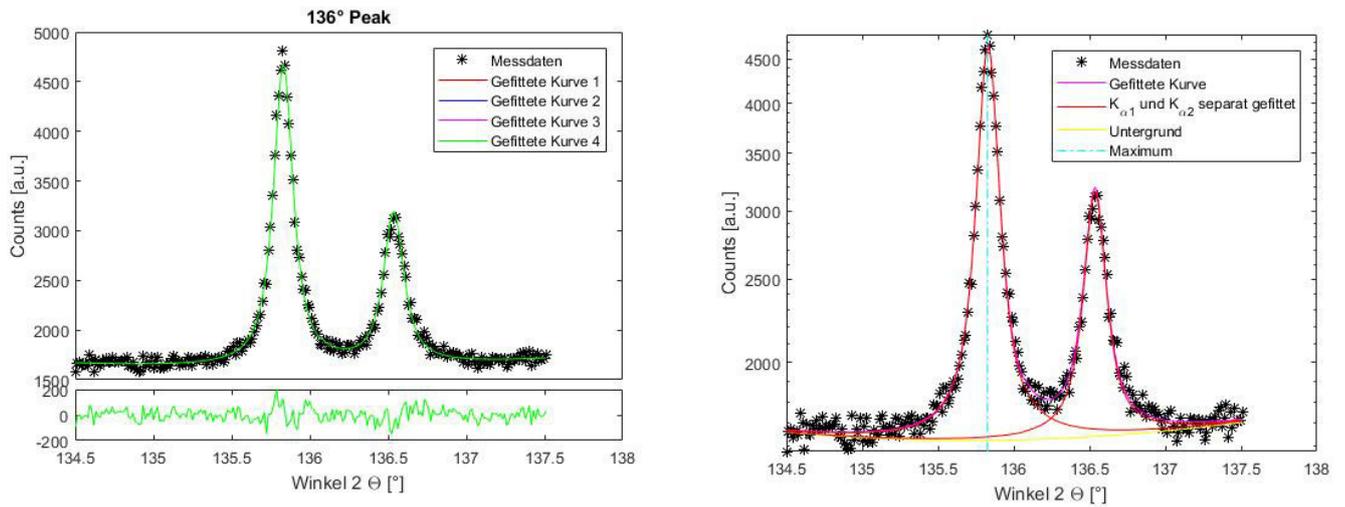


Figure 62: Linear- (left) and semi-logarithmic (right) fits of the 136 ° LaB₆ peak

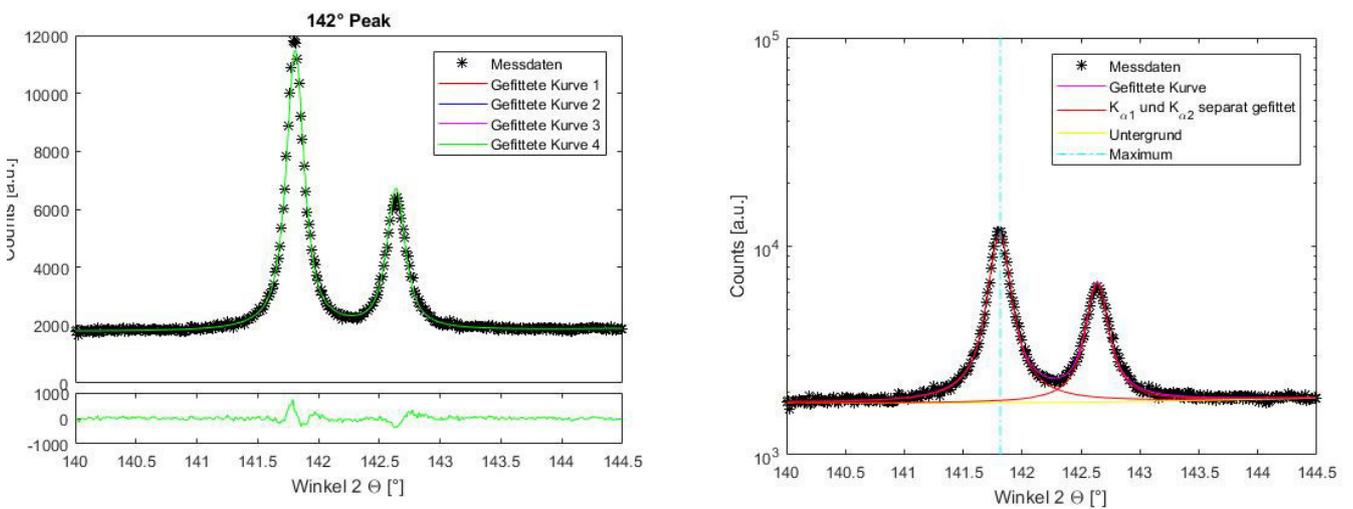


Figure 63: Linear- (left) and semi-logarithmic (right) fits of the 142 ° LaB₆ peak

9.3.2 Steel samples

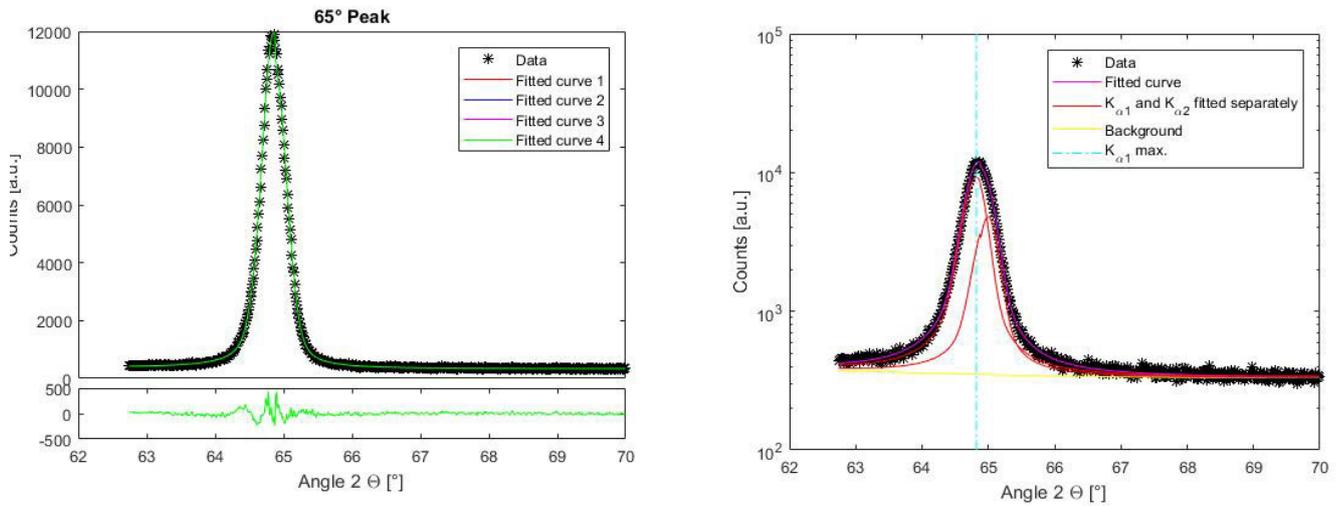


Figure 64: Linear- (left) and semi-logarithmic (right) fits of the 65 ° peak of the 1 h austenitized and 10 min at 760 °C tempered P91 sample

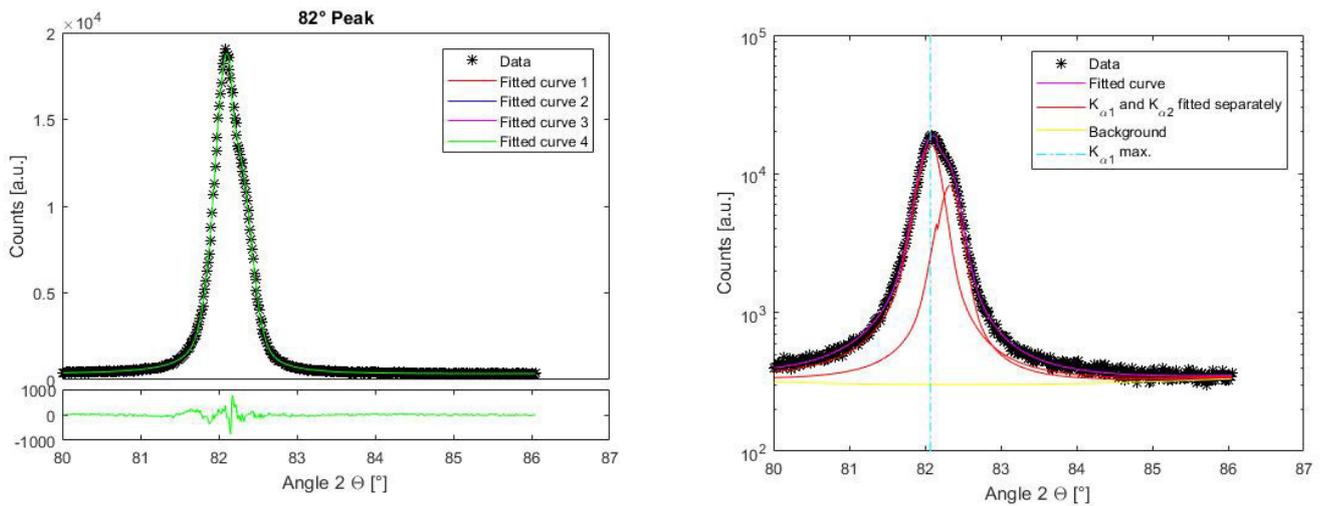


Figure 65: Linear- (left) and semi-logarithmic (right) fits of the 82 ° peak of the 1 h austenitized and 10 min at 760 °C tempered P91 sample

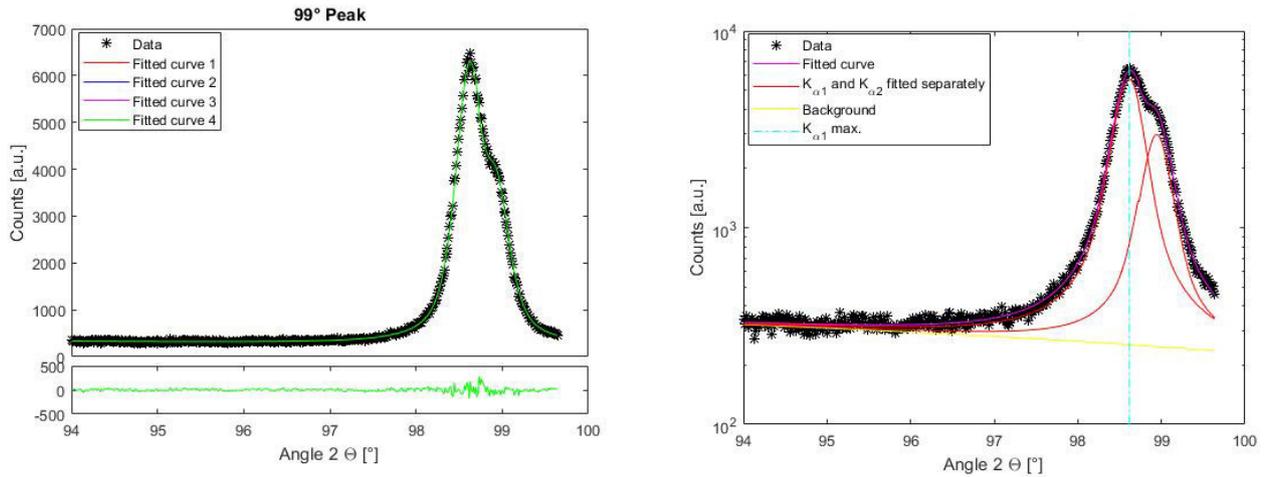


Figure 66: Linear- (left) and semi-logarithmic (right) fits of the 99 ° peak of the 1 h austenitized and 10 min at 760 °C tempered P91 sample

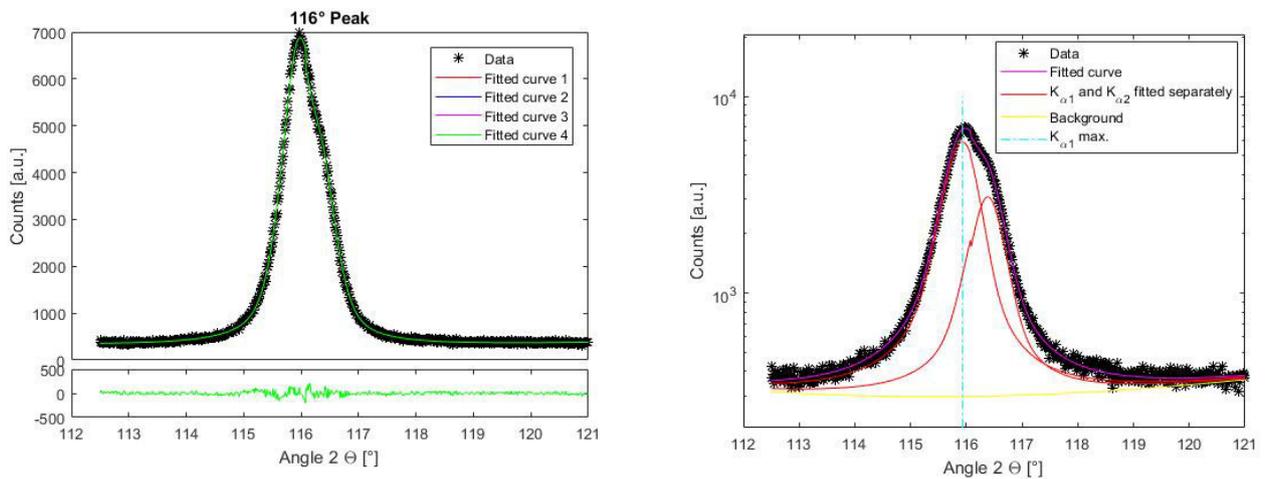


Figure 67: Linear- (left) and semi-logarithmic (right) fits of the 116 ° peak of the 1 h austenitized and 10 min at 760 °C tempered P91 sample

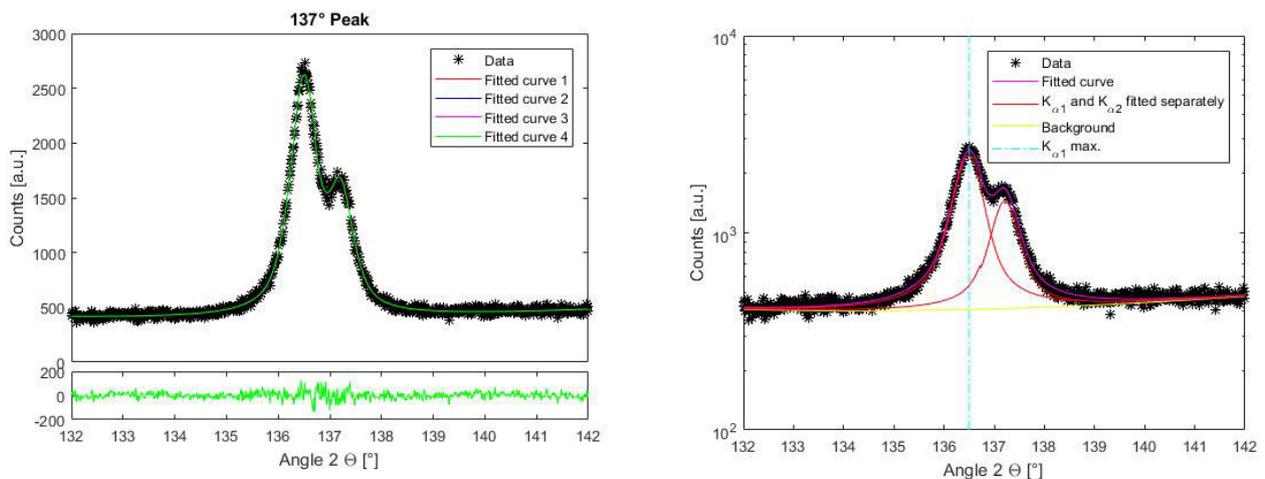


Figure 68: Linear- (left) and semi-logarithmic (right) fits of the 137 ° peak of the 1 h austenitized and 10 min at 760 °C tempered P91 sample

9.4 Fit results

In this subsection, the corresponding results to the plotted data fits of the previous subsection are tabulated.

Table 6: LaB₆ standard

Integral breadth pseudo-Voigt	Integral breadth Cauchy	Integral breadth Gauss	Integrated intensity
0.113740222749642	0.132367463441248	0.0948719506512459	17884.6106108726
0.113740735946802	0.132368544322303	0.0948726521361968	17884.6003329712
0.113740768398450	0.132368400030091	0.0948725760926827	17884.6037876325
0.113740486259267	0.132368031500252	0.0948723240548382	17884.6029126597
0.133271870503564	0.152885341852420	0.109296559093271	9965.81007169609
0.133271988633088	0.152885920234073	0.109297004449128	9965.80085004098
0.133271714474834	0.152885549699805	0.109296730699754	9965.79971850123
0.133271950704298	0.152885258429148	0.109296479612961	9965.81454011559
0.140173408384346	0.157835321766393	0.111158157373015	4929.12549159506
0.140173408384346	0.157835321766393	0.111158157373015	4929.12549159506
0.140173576607415	0.157835374986523	0.111158166412878	4929.12712313214
0.140173576607415	0.157835374986523	0.111158166412878	4929.12712313214
0.141473207373062	0.161060807358797	0.113001209595199	9805.79868795496
0.141473207373062	0.161060807358797	0.113001209595199	9805.79868795496
0.141473207373062	0.161060807358797	0.113001209595199	9805.79868795496
0.145413065789622	0.166328657539993	0.116103434165877	5033.40435122657
0.145413065789622	0.166328657539993	0.116103434165877	5033.40435122657
0.145413065789622	0.166328657539993	0.116103434165877	5033.40435122657
0.145413050716555	0.166328513793693	0.116103333767051	5033.40536460108
0.170133734479694	0.191159132408559	0.131679199577469	1987.88406646569
0.170133734479694	0.191159132408559	0.131679199577469	1987.88406646569
0.170133734479694	0.191159132408559	0.131679199577469	1987.88406646569
0.170133734479694	0.191159132408559	0.131679199577469	1987.88406646569
0.181505776933465	0.194974707545910	0.133673986523230	315.415862044294
0.181506011130462	0.194974093249444	0.133673529866432	315.416468350936
0.181505568655242	0.194972901548768	0.133672684789216	315.416587159285
0.181505568655242	0.194972901548768	0.133672684789216	315.416587159285
0.186542910916687	0.208975541622441	0.143641888253041	1076.00055761562
0.186542910916687	0.208975541622441	0.143641888253041	1076.00055761562
0.186542833705588	0.208976521444752	0.143642567020533	1075.99809972294
0.186543379046421	0.208976896410744	0.143642841344490	1075.99957373043
0.194655068450157	0.217751992026790	0.149267107342917	2251.23553144254
0.194654769678021	0.217752146524258	0.149267225997058	2251.23296275034
0.194654769678021	0.217752146524258	0.149267225997058	2251.23296275034
0.194654769678021	0.217752146524258	0.149267225997058	2251.23296275034
0.212656759768239	0.235180369246794	0.160919745479342	2027.79951968874
0.212656759768239	0.235180369246794	0.160919745479342	2027.79951968874
0.212656759768239	0.235180369246794	0.160919745479342	2027.79951968874
0.212656746095571	0.235180362277494	0.160919741015815	2027.79944118630

0.223180413724535	0.246487398442175	0.168492366826158	1696.18992124553
0.223180413724535	0.246487398442175	0.168492366826158	1696.18992124553
0.223181004179213	0.246489965473082	0.168494151964526	1696.18737255908
0.223181004179213	0.246489965473082	0.168494151964526	1696.18737255908
0.230920522732536	0.252578620302769	0.172462632764484	702.862040376335
0.230920522732536	0.252578620302769	0.172462632764484	702.862040376335
0.230920973016664	0.252578358992623	0.172462531197930	702.861781156520
0.230920973016664	0.252578358992623	0.172462531197930	702.861781156520
0.243385178705720	0.264674267334700	0.180575926312178	988.848941343270
0.243385178705720	0.264674267334700	0.180575926312178	988.848941343270
0.243386079054366	0.264676671323308	0.180577584982675	988.848179327951
0.243386079054366	0.264676671323308	0.180577584982675	988.848179327951
0.256783696639391	0.276855532829105	0.188875412390795	1940.17369656041
0.256783789801506	0.276855333381905	0.188875274103202	1940.17424813671
0.256783789801506	0.276855333381905	0.188875274103202	1940.17424813671
0.256783014013469	0.276853817814211	0.188874216377102	1940.17302964937
0.277152717209382	0.297376441065285	0.202685613493050	1014.78294741914
0.277153742020046	0.297380191526494	0.202688275140894	1014.78102583201
0.277153742020046	0.297380191526494	0.202688275140894	1014.78102583201
0.277153742020046	0.297380191526494	0.202688275140894	1014.78102583201
0.310026734930681	0.327717103846145	0.222987936329148	717.976725878305
0.310033603674938	0.327731431265359	0.222997966591581	717.980563018944
0.310033603674938	0.327731431265359	0.222997966591581	717.980563018944
0.310030106739735	0.327719001314929	0.222989335001603	717.984591185730
0.336815564349354	0.358606887166049	0.244265633328297	1014.56227250941
0.336815564349354	0.358606887166049	0.244265633328297	1014.56227250941
0.336815564349354	0.358606887166049	0.244265633328297	1014.56227250941
0.336814092416503	0.358603234778358	0.244263122689290	1014.56268691770
0.398144724617292	0.419700574736405	0.285587384915823	3849.57417445936
0.398144215738860	0.419699004498508	0.285586317368756	3849.57715262779
0.398142068868931	0.419693223692860	0.285582315050216	3849.57792380599
0.398142068868931	0.419693223692860	0.285582315050216	3849.57792380599

Table 7: LaB₆ standard

Peak maximum	HWHM	Shape parameter	Asymmetry
21.3884271649364	0.0313632071360818	0.585625535375824	1.06294256513561
21.3884272178622	0.0313635087818730	0.585615905556738	1.06294475406496
21.3884270338026	0.0313634998195822	0.585619533430704	1.06293612081357
21.3884271537451	0.0313633720765890	0.585620240787428	1.06294165522114
37.4988375385985	0.0340004458077274	0.630981855764799	1.06077175976268
37.4988375357265	0.0340005925117071	0.630973145957002	1.06077124111180
37.4988375731684	0.0340005078539366	0.630974304981516	1.06077311573567
37.4988374971210	0.0340004131234973	0.630985391306260	1.06076962658190
43.5692000307882	0.0341773284861275	0.699939387828430	1.03348367920340
43.5692000307882	0.0341773284861275	0.699939387828430	1.03348367920340
43.5691999444809	0.0341773110087495	0.699941961335800	1.03348185499016
43.5691999444809	0.0341773110087495	0.699941961335800	1.03348185499016
49.0266685090153	0.0352444719918997	0.674455781131947	1.03670006159017
49.0266685090153	0.0352444719918997	0.674455781131947	1.03670006159017
49.0266685090153	0.0352444719918997	0.674455781131947	1.03670006159017
49.0266685090153	0.0352444719918997	0.674455781131947	1.03670006159017
54.0630443811062	0.0363533658885596	0.667501328697057	1.01678451187657
54.0630443811062	0.0363533658885596	0.667501328697057	1.01678451187657
54.0630443811062	0.0363533658885596	0.667501328697057	1.01678451187657
54.0630444511687	0.0363533345552590	0.667503340328126	1.01678688056007
75.9400976243049	0.0412091721240406	0.726409824737839	0.996237539212653
75.9400976243049	0.0412091721240406	0.726409824737839	0.996237539212653
75.9400976243049	0.0412091721240406	0.726409824737839	0.996237539212653
75.9400976243049	0.0412091721240406	0.726409824737839	0.996237539212653
79.9720753028926	0.0412502698874813	0.838183061344497	1.10274786494301
79.9720752155603	0.0412501073862938	0.838193600665173	1.10274956988894
79.9720752461872	0.0412498447744054	0.838202315471030	1.10275012822668
79.9720752461872	0.0412498447744054	0.838202315471030	1.10275012822668
83.9582399896323	0.0449179967296640	0.735609638319486	1.01274440292375
83.9582399896323	0.0449179967296640	0.735609638319486	1.01274440292375
83.9582401346377	0.0449182007054309	0.735597039659949	1.01274442793164
83.9582400644587	0.0449182908830316	0.735599722989518	1.01274499004827
87.9098049823613	0.0464053328939600	0.741382559765623	1.01306730645094
87.9098050147889	0.0464053749670290	0.741377017302146	1.01306837572873
87.9098050147889	0.0464053749670290	0.741377017302146	1.01306837572873
87.9098050147889	0.0464053749670290	0.741377017302146	1.01306837572873
99.7820818351671	0.0502787778687344	0.770485810446388	1.00051642939354
99.7820818351671	0.0502787778687344	0.770485810446388	1.00051642939354
99.7820818351671	0.0502787778687344	0.770485810446388	1.00051642939354
99.7820818334116	0.0502787766169256	0.770485726004649	1.00051653484756

103.808955362336	0.0526012164788070	0.774397812939374	0.993771213101137
103.808955362336	0.0526012164788070	0.774397812939374	0.993771213101137
103.808955365673	0.0526017753469023	0.774379148818204	0.993766507296094
103.808955365673	0.0526017753469023	0.774379148818204	0.993766507296094
107.907019700709	0.0537520403254519	0.798101266216986	1.01157090992810
107.907019700709	0.0537520403254519	0.798101266216986	1.01157090992810
107.907020026577	0.0537520446713019	0.798108009786767	1.01158620296752
107.907020026577	0.0537520446713019	0.798108009786767	1.01158620296752
112.101950041901	0.0562978665510730	0.812182798618977	1.01365826009174
112.101950041901	0.0562978665510730	0.812182798618977	1.01365826009174
112.101950131723	0.0562983784602818	0.812170167331704	1.01365745264514
112.101950131723	0.0562983784602818	0.812170167331704	1.01365745264514
116.425244967491	0.0589347355400611	0.832192823241807	1.00042447264914
116.425244951207	0.0589346907409217	0.832195336642995	1.00042591629683
116.425244951207	0.0589346907409217	0.832195336642995	1.00042591629683
116.425244866173	0.0589343507129160	0.832201080976269	1.00043034334504
120.915982234103	0.0631626090984507	0.843808643238230	0.986217086080292
120.915982596282	0.0631634874533441	0.843787915041249	0.986215082174311
120.915982596282	0.0631634874533441	0.843787915041249	0.986215082174311
120.915982596282	0.0631634874533441	0.843787915041249	0.986215082174311
130.644149710737	0.0694355204122869	0.878506942911689	1.02015672660576
130.644150663703	0.0694388049654912	0.878457931303763	1.02012781399280
130.644150663703	0.0694388049654912	0.878457931303763	1.02012781399280
130.644150537171	0.0694360715834513	0.878518206133544	1.02012759409337
136.065721585844	0.0763954196957784	0.861786374024273	0.987965515895827
136.065721585844	0.0763954196957784	0.861786374024273	0.987965515895827
136.065721585844	0.0763954196957784	0.861786374024273	0.987965515895827
136.065721521154	0.0763946372139189	0.861799640294168	0.987967085376948
142.084550077718	0.0890976189806021	0.884709979329335	0.979576737016389
142.084550213997	0.0890972958018407	0.884715507417503	0.979575571273365
142.084550023153	0.0890960279821621	0.884734408594966	0.979582833471231
142.084550023153	0.0890960279821621	0.884734408594966	0.979582833471231

Table 8: P91.0p166h_760C

Integral breadth pseudo-Voigt	Integral breadth Cauchy	Integral breadth Gauss	Integrated intensity
0.527290119399441	0.649762786706706	0.437569658733183	5988.26509065383
0.527290119399441	0.649762786706706	0.437569658733183	5988.26509065383
0.527290119399441	0.649762786706706	0.437569658733183	5988.26509065383
0.527288732046739	0.649758277268281	0.437566772369075	5988.27209034689
0.570109437578331	0.664848017623619	0.467560666125258	10353.2844620845
0.570109437578331	0.664848017623619	0.467560666125258	10353.2844620845
0.570109437578331	0.664848017623619	0.467560666125258	10353.2844620845
0.570107268124486	0.664843120842662	0.467557657344037	10353.2865732479
0.700783444336066	0.770119323869001	0.552368331075112	4225.70818272576
0.700780044906837	0.770111575602760	0.552363114146670	4225.71230050969
0.700780044906837	0.770111575602760	0.552363114146670	4225.71230050969
0.700780044906837	0.770111575602760	0.552363114146670	4225.71230050969
1.12755603484098	1.29018831486239	0.899549248980856	7328.56745199267
1.12755603484098	1.29018831486239	0.899549248980856	7328.56745199267
1.12755603484098	1.29018831486239	0.899549248980856	7328.56745199267
1.12755603484098	1.29018831486239	0.899549248980856	7328.56745199267
1.22887486879813	1.32771006770420	0.959480432537655	2721.84802594968
1.22887486879813	1.32771006770420	0.959480432537655	2721.84802594968
1.22889176595041	1.32772412458233	0.959491093183323	2721.87410992779
1.22888530078116	1.32771822189133	0.959486761063414	2721.86815553721

Table 9: P91.0p166h_760C

Peak maximum	HWHM	Shape parameter	Asymmetry
64.8771566549783	0.173367167716769	0.521033180869180	1.24535863555926
64.8771566549783	0.173367167716769	0.521033180869180	1.24535863555926
64.8771566549783	0.173367167716769	0.521033180869180	1.24535863555926
64.8771557039396	0.173365872462965	0.521043626364064	1.24534854549737
82.1625179298473	0.163172742774584	0.606171295457613	1.17604168905240
82.1625179298473	0.163172742774584	0.606171295457613	1.17604168905240
82.1625179298473	0.163172742774584	0.606171295457613	1.17604168905240
82.1625165381724	0.163171502386559	0.606179899828647	1.17603082383996
98.7226773292644	0.183708979085349	0.749017857012887	1.07228168913006
98.7226759906612	0.183706843473346	0.749031834340486	1.07227973062992
98.7226759906612	0.183706843473346	0.749031834340486	1.07227973062992
98.7226759906612	0.183706843473346	0.749031834340486	1.07227973062992
116.080892625399	0.322325968860978	0.667862517551163	1.13173692223232
116.080892625399	0.322325968860978	0.667862517551163	1.13173692223232
116.080892625399	0.322325968860978	0.667862517551163	1.13173692223232
116.080892625399	0.322325968860978	0.667862517551163	1.13173692223232
136.732032735898	0.304415874047110	0.790433740193638	1.07664902032007
136.732032735898	0.304415874047110	0.790433740193638	1.07664902032007
136.732035393309	0.304419670663985	0.790442248452595	1.07667024585572
136.732037211124	0.304418121158347	0.790440005681302	1.07667730844157

Table 10: P91.0p5h_760C

Integral breadth pseudo-Voigt	Integral breadth Cauchy	Integral breadth Gauss	Integrated intensity
0.414509964063693	0.476055746089340	0.337190469465058	5089.89594739535
0.414509964063693	0.476055746089340	0.337190469465058	5089.89594739535
0.414509964063693	0.476055746089340	0.337190469465058	5089.89594739535
0.414509964063693	0.476055746089340	0.337190469465058	5089.89594739535
0.417279253308333	0.458618374699876	0.331081627643383	9882.56447785912
0.417280700620139	0.458620549731061	0.331083243237469	9882.56440167673
0.417280700620139	0.458620549731061	0.331083243237469	9882.56440167673
0.417280700620139	0.458620549731061	0.331083243237469	9882.56440167673
0.530345531235751	0.572628066351387	0.411936414654916	4595.77846186262
0.530345531235751	0.572628066351387	0.411936414654916	4595.77846186262
0.530347766352230	0.572629525165320	0.411937508166826	4595.78853216560
0.530347766352230	0.572629525165320	0.411937508166826	4595.78853216560
0.908176293148979	0.999701749542010	0.721908970340945	6395.28293836444
0.908176657013753	0.999702431987376	0.721909433917736	6395.28294506710
0.908176657013753	0.999702431987376	0.721909433917736	6395.28294506710
0.908176657013753	0.999702431987376	0.721909433917736	6395.28294506710
0.938254724756315	1.02768876155645	0.727966912531045	2487.08831486430
0.938254724756315	1.02768876155645	0.727966912531045	2487.08831486430
0.938254724756315	1.02768876155645	0.727966912531045	2487.08831486430
0.938254724756315	1.02768876155645	0.727966912531045	2487.08831486430

Table 11: P91.0p5h_760C

Peak maximum	HWHM	Shape parameter	Asymmetry
64.8993395019358	0.115643440559515	0.639467062477988	1.12198360607521
64.8993395019358	0.115643440559515	0.639467062477988	1.12198360607521
64.8993395019358	0.115643440559515	0.639467062477988	1.12198360607521
64.8993395019358	0.115643440559515	0.639467062477988	1.12198360607521
82.1725390032892	0.104799311241647	0.742821774547169	1.04539273867422
82.1725389989058	0.104799863963083	0.742818012435179	1.04538970116443
82.1725389989058	0.104799863963083	0.742818012435179	1.04538970116443
82.1725389989058	0.104799863963083	0.742818012435179	1.04538970116443
98.7394966636709	0.129756379261171	0.795619727231197	1.03301723862404
98.7394966636709	0.129756379261171	0.795619727231197	1.03301723862404
98.7394964296277	0.129756744875696	0.795624263042565	1.03301380596677
98.7394964296277	0.129756744875696	0.795624263042565	1.03301380596677
116.112323923778	0.234380357936342	0.738101369289730	1.02574903115413
116.112324235493	0.234380536013117	0.738100600774243	1.02575065130667
116.112324235493	0.234380536013117	0.738100600774243	1.02575065130667
116.112324235493	0.234380536013117	0.738100600774243	1.02575065130667
136.750048926640	0.228077316272671	0.768487038186277	1.00085151669058
136.750048926640	0.228077316272671	0.768487038186277	1.00085151669058
136.750048926640	0.228077316272671	0.768487038186277	1.00085151669058
136.750048926640	0.228077316272671	0.768487038186277	1.00085151669058

Table 12: P91_1p5h_760C

Integral breadth pseudo-Voigt	Integral breadth Cauchy	Integral breadth Gauss	Integrated intensity
0.366177261402393	0.414795953977922	0.297613320665316	5577.43057864836
0.366177261402393	0.414795953977922	0.297613320665316	5577.43057864836
0.366177261402393	0.414795953977922	0.297613320665316	5577.43057864836
0.366177261402393	0.414795953977922	0.297613320665316	5577.43057864836
0.379646949555851	0.408994791343633	0.293367629868689	10351.2273710940
0.379646949555851	0.408994791343633	0.293367629868689	10351.2273710940
0.379646654639386	0.408995508640324	0.293368192842500	10351.2185743290
0.379646891022587	0.408997226710955	0.293369515377593	10351.2085811920
0.464531687968059	0.492067573836691	0.350140861232518	4581.78271136102
0.464531881740614	0.492067641348398	0.350140906556368	4581.78387806017
0.464532174143415	0.492068491144420	0.350141551060399	4581.78236823877
0.464532174143415	0.492068491144420	0.350141551060399	4581.78236823877
0.770785959681230	0.845451279326925	0.610832425359612	6643.44832668966
0.770788976793776	0.845455831494055	0.610835800443739	6643.44946720184
0.770788126990665	0.845454672219675	0.610834938292847	6643.44852035973
0.770790966061206	0.845457961270153	0.610837349271493	6643.45242188789
0.836105572435249	0.865437631666447	0.606746763424706	2592.94557618829
0.836105572435249	0.865437631666447	0.606746763424706	2592.94557618829
0.836105572435249	0.865437631666447	0.606746763424706	2592.94557618829
0.836105572435249	0.865437631666447	0.606746763424706	2592.94557618829

Table 13: P91_1p5h_760C

Peak maximum	HWHM	Shape parameter	Asymmetry
64.8833797848310	0.0984230268271856	0.662789556821009	1.15285885641500
64.8833797848310	0.0984230268271856	0.662789556821009	1.15285885641500
64.8833797848310	0.0984230268271856	0.662789556821009	1.15285885641500
64.8833797848310	0.0984230268271856	0.662789556821009	1.15285885641500
82.1611315021426	0.0920186157537140	0.803867943373483	1.08180594982322
82.1611315021426	0.0920186157537140	0.803867943373483	1.08180594982322
82.1611315710677	0.0920187918095511	0.803860911761203	1.08180826224458
82.1611316833972	0.0920192141962710	0.803850918461782	1.08181019028618
98.7272618556633	0.109686661220961	0.853761285632002	1.05307821636404
98.7272618506488	0.109686676808984	0.853762021116452	1.05307766883353
98.7272618111474	0.109686884543075	0.853759095290150	1.05307664318315
98.7272618111474	0.109686884543075	0.853759095290150	1.05307664318315
116.086568355190	0.193511118250455	0.747800449897778	1.09423242308583
116.086568413151	0.193512299051580	0.747796123886198	1.09423096007994
116.086568386431	0.193511997746486	0.747796927713354	1.09423090862784
116.086568282595	0.193512845967704	0.747796285182399	1.09422523110658
136.731912583747	0.189816558014181	0.917717438072142	1.05330637473692
136.731912583747	0.189816558014181	0.917717438072142	1.05330637473692
136.731912583747	0.189816558014181	0.917717438072142	1.05330637473692
136.731912583747	0.189816558014181	0.917717438072142	1.05330637473692

Table 14: P91.12h_760C

Integral breadth pseudo-Voigt	Integral breadth Cauchy	Integral breadth Gauss	Integrated intensity
0.333548421020954	0.384901033130752	0.277566062080759	6200.47682767727
0.333548421020954	0.384901033130752	0.277566062080759	6200.47682767727
0.333547782671893	0.384899598044171	0.277565108034469	6200.47682427697
0.333547782671893	0.384899598044171	0.277565108034469	6200.47682427697
0.349288909262619	0.386681703261704	0.276383278901306	11737.2852119510
0.349286432446009	0.386676787421091	0.276379514487305	11737.2966194642
0.349286432446009	0.386676787421091	0.276379514487305	11737.2966194642
0.349286432446009	0.386676787421091	0.276379514487305	11737.2966194642
0.416341918816186	0.445192173386239	0.314569608900544	3695.57157896648
0.416341918816186	0.445192173386239	0.314569608900544	3695.57157896648
0.416341918816186	0.445192173386239	0.314569608900544	3695.57157896648
0.416341918816186	0.445192173386239	0.314569608900544	3695.57157896648
0.683743329439318	0.764123038774808	0.549638620158321	6900.68486618953
0.683743329439318	0.764123038774808	0.549638620158321	6900.68486618953
0.683743329439318	0.764123038774808	0.549638620158321	6900.68486618953
0.683743329439318	0.764123038774808	0.549638620158321	6900.68486618953
0.755124165093155	0.783143790457269	0.546262396160159	2864.44423345055
0.755119489001784	0.783135278365550	0.546256150507176	2864.44347093061
0.755119489001784	0.783135278365550	0.546256150507176	2864.44347093061
0.755119489001784	0.783135278365550	0.546256150507176	2864.44347093061

Table 15: P91.12h_760C

Peak maximum	HWHM	Shape parameter	Asymmetry
64.9245282900980	0.0897420400351990	0.601866455106988	1.18367724997587
64.9245282900980	0.0897420400351990	0.601866455106988	1.18367724997587
64.9245279525372	0.0897417033959266	0.601871454378399	1.18367064749979
64.9245279525372	0.0897417033959266	0.601871454378399	1.18367064749979
82.2143763265198	0.0863697079908372	0.731746388673641	1.11796340689875
82.2143765227169	0.0863684597019364	0.731762837293301	1.11797021384826
82.2143765227169	0.0863684597019364	0.731762837293301	1.11797021384826
82.2143765227169	0.0863684597019364	0.731762837293301	1.11797021384826
98.7736801548981	0.0983140428206316	0.833122391738689	1.05810069202491
98.7736801548981	0.0983140428206316	0.833122391738689	1.05810069202491
98.7736801548981	0.0983140428206316	0.833122391738689	1.05810069202491
98.7736801548981	0.0983140428206316	0.833122391738689	1.05810069202491
116.126483214169	0.172524798199945	0.698744613002953	1.12388631033446
116.126483214169	0.172524798199945	0.698744613002953	1.12388631033446
116.126483214169	0.172524798199945	0.698744613002953	1.12388631033446
116.126483214169	0.172524798199945	0.698744613002953	1.12388631033446
136.774341862071	0.170691267907944	0.914431422867559	1.09631184153677
136.774342959736	0.170689298086170	0.914442767358552	1.09631605089046
136.774342959736	0.170689298086170	0.914442767358552	1.09631605089046
136.774342959736	0.170689298086170	0.914442767358552	1.09631605089046

Table 16: P91.24h_760C

Integral breadth pseudo-Voigt	Integral breadth Cauchy	Integral breadth Gauss	Integrated intensity
0.305692509958350	0.346288860439689	0.250317257621689	5912.24864056680
0.305694016484607	0.346292444627294	0.250319862293378	5912.24205892435
0.305694016484607	0.346292444627294	0.250319862293378	5912.24205892435
0.305694016484607	0.346292444627294	0.250319862293378	5912.24205892435
0.323848006168151	0.352227226350850	0.249862598258243	10802.2933870654
0.323847497000257	0.35222720724925	0.249859116341985	10802.3235623512
0.323847497000257	0.35222720724925	0.249859116341985	10802.3235623512
0.323847497000257	0.35222720724925	0.249859116341985	10802.3235623512
0.390192721324623	0.417695456038596	0.293686726170199	3800.96551527486
0.390192668070817	0.417689718116537	0.293682447936976	3800.98701926415
0.390192668070817	0.417689718116537	0.293682447936976	3800.98701926415
0.390192668070817	0.417689718116537	0.293682447936976	3800.98701926415
0.613532669250457	0.676673954332981	0.482080263578520	7110.86743231929
0.613533313748809	0.676673287814297	0.482079652208242	7110.87435856377
0.613536798373287	0.676679310256803	0.482084200209241	7110.87148951862
0.613536798373287	0.676679310256803	0.482084200209241	7110.87148951862
0.695418230651589	0.712274325462526	0.494122933816402	2802.86967703749
0.695418230651589	0.712274325462526	0.494122933816402	2802.86967703749
0.695430585048691	0.712316451752918	0.494153179261912	2802.84303372179
0.695430585048691	0.712316451752918	0.494153179261912	2802.84303372179

Table 17: P91.24h_760C

Peak maximum	HWHM	Shape parameter	Asymmetry
64.9367094057584	0.0794152054440875	0.653622048349723	1.14683494039794
64.9367096818125	0.0794161161170344	0.653605959358552	1.14683731469033
64.9367096818125	0.0794161161170344	0.653605959358552	1.14683731469033
64.9367096818125	0.0794161161170344	0.653605959358552	1.14683731469033
82.2133188521652	0.0780424344617268	0.786100106256457	1.09946284615866
82.2133189070837	0.0780413370341047	0.786130733633185	1.09946083731188
82.2133189070837	0.0780413370341047	0.786130733633185	1.09946083731188
82.2133189070837	0.0780413370341047	0.786130733633185	1.09946083731188
98.7751517679573	0.0918418068009864	0.833072159664589	1.05193412173246
98.7751520111362	0.0918404664441726	0.833107106698686	1.05193849543373
98.7751520111362	0.0918404664441726	0.833107106698686	1.05193849543373
98.7751520111362	0.0918404664441726	0.833107106698686	1.05193849543373
116.136778683636	0.150900456514930	0.745043363951511	1.08698163250808
116.136778695893	0.150900288045291	0.745049176566665	1.08697596043069
116.136778315545	0.150901778368869	0.745039903781976	1.08696845150169
116.136778315545	0.150901778368869	0.745039903781976	1.08696845150169
136.771319937993	0.154461112829177	0.945098031378775	1.07171325190312
136.771319937993	0.154461112829177	0.945098031378775	1.07171325190312
136.771315650699	0.154470534146649	0.945001667283013	1.07168923566038
136.771315650699	0.154470534146649	0.945001667283013	1.07168923566038

Table 18: P91.36h_760C

Integral breadth pseudo-Voigt	Integral breadth Cauchy	Integral breadth Gauss	Integrated intensity
0.343028070226883	0.403058552871494	0.289045080036953	5486.52454228142
0.343033867511387	0.403063397451939	0.289048272936817	5486.55129473957
0.343033867511387	0.403063397451939	0.289048272936817	5486.55129473957
0.343033867511387	0.403063397451939	0.289048272936817	5486.55129473957
0.356876204258982	0.396270366617179	0.283868511294068	11001.6754057003
0.356876081310387	0.396269958688344	0.283868194625187	11001.6772065470
0.356876081310387	0.396269958688344	0.283868194625187	11001.6772065470
0.356874950251742	0.396272034176421	0.283869747479185	11001.6394059249
0.410995870380892	0.441637618614087	0.311784947287338	3974.85911125956
0.410995870380892	0.441637618614087	0.311784947287338	3974.85911125956
0.410992796298037	0.441636957734514	0.311784576978154	3974.83982089386
0.410992796298037	0.441636957734514	0.311784576978154	3974.83982089386
0.673590865003337	0.750461204553644	0.539297253509963	6325.09638943283
0.673590865003337	0.750461204553644	0.539297253509963	6325.09638943283
0.673590865003337	0.750461204553644	0.539297253509963	6325.09638943283
0.673590865003337	0.750461204553644	0.539297253509963	6325.09638943283
0.743381199155224	0.759410369022161	0.528272811792164	2649.35553279751
0.743384772039110	0.759410093746513	0.528272323936304	2649.36253287116
0.743383214150202	0.759380864155774	0.528250554751332	2649.39495572746
0.743383214150202	0.759380864155774	0.528250554751332	2649.39495572746

Table 19: P91.36h_760C

Peak maximum	HWHM	Shape parameter	Asymmetry
64.9181032810008	0.0956611495560143	0.556338664668804	1.17733605134046
64.9181033213498	0.0956625246525295	0.556354729771529	1.17733185656184
64.9181033213498	0.0956625246525295	0.556354729771529	1.17733185656184
64.9181033213498	0.0956625246525295	0.556354729771529	1.17733185656184
82.2011440593872	0.0887723453167592	0.721222307051086	1.12577564872915
82.2011440489761	0.0887722446454527	0.721224312354766	1.12577519585724
82.2011440489761	0.0887722446454527	0.721224312354766	1.12577519585724
82.2011439015788	0.0887727854573157	0.721200508427309	1.12576723814695
98.7642710093593	0.0975836804586582	0.820988744979926	1.05493558664940
98.7642710093593	0.0975836804586582	0.820988744979926	1.05493558664940
98.7642714329932	0.0975835370368478	0.820973119852965	1.05494965724386
98.7642714329932	0.0975835370368478	0.820973119852965	1.05494965724386
116.114371190709	0.168950883037129	0.708545315666624	1.14080126668979
116.114371190709	0.168950883037129	0.708545315666624	1.14080126668979
116.114371190709	0.168950883037129	0.708545315666624	1.14080126668979
116.114371190709	0.168950883037129	0.708545315666624	1.14080126668979
136.755722172212	0.164950587268971	0.950718127730553	1.09315091489441
136.755727135771	0.164950055907528	0.950730286560340	1.09317258513773
136.755733342297	0.164942771336678	0.950815699196104	1.09319978390023
136.755733342297	0.164942771336678	0.950815699196104	1.09319978390023

9.5 Determination of q

The following figures (69 to 75) show the plots to determine the parameter q to then calculate the contrast factor C .

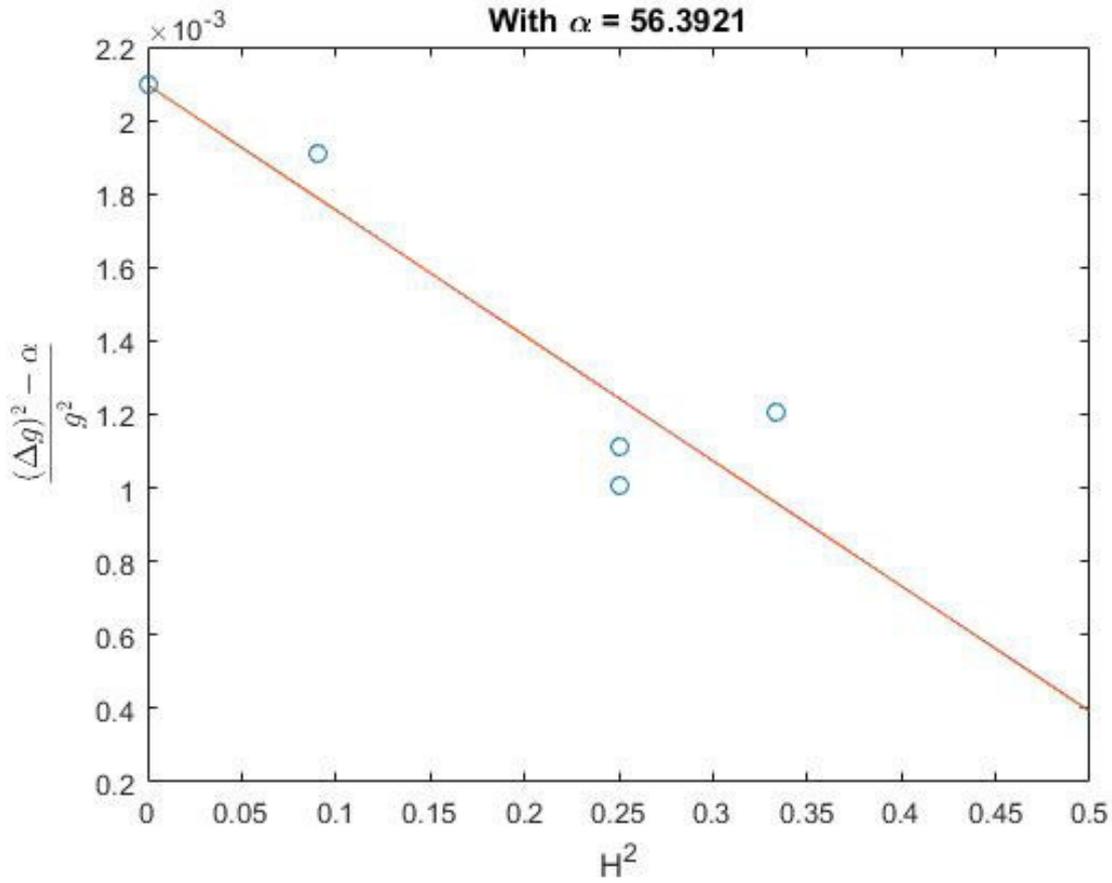


Figure 69: Determination of the parameter q for sample P91_BH_vib_pol

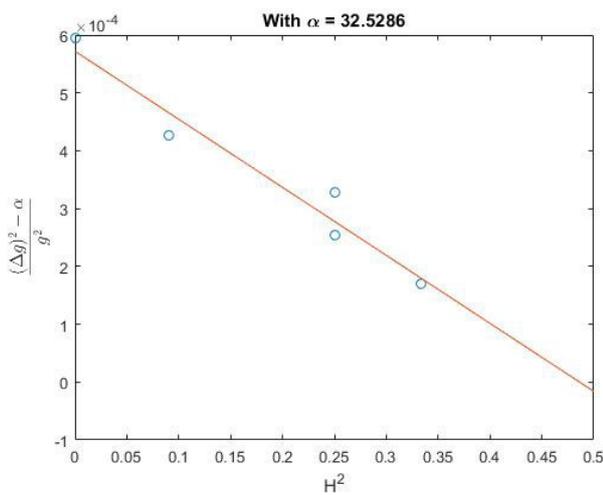


Figure 70: Determination of the parameter q for sample P91_0p166h.760C

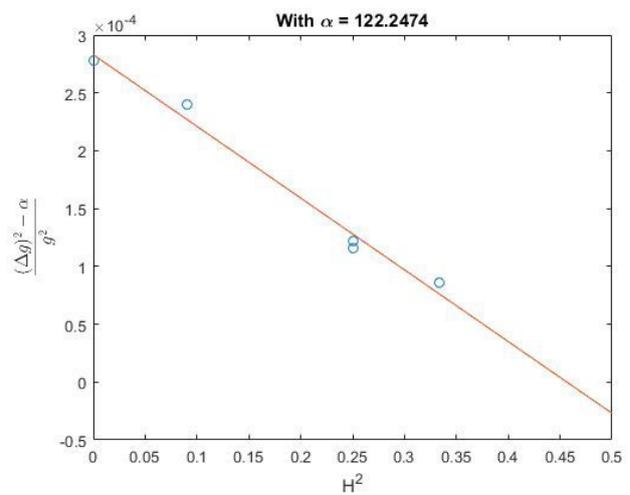


Figure 71: Determination of the parameter q for sample P91_0p5h.760C

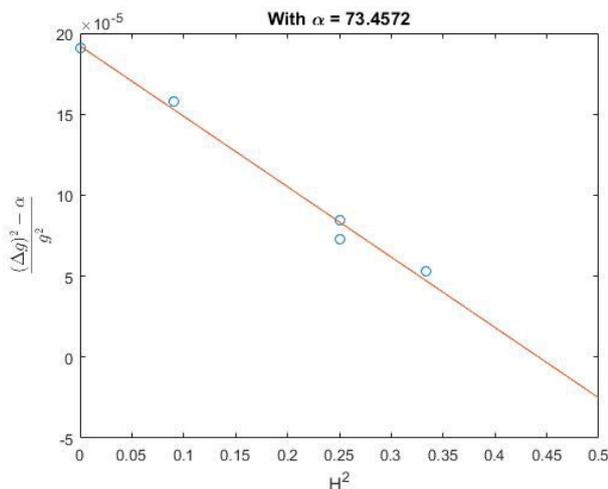


Figure 72: Determination of the parameter q for sample P91_1p5h_760C

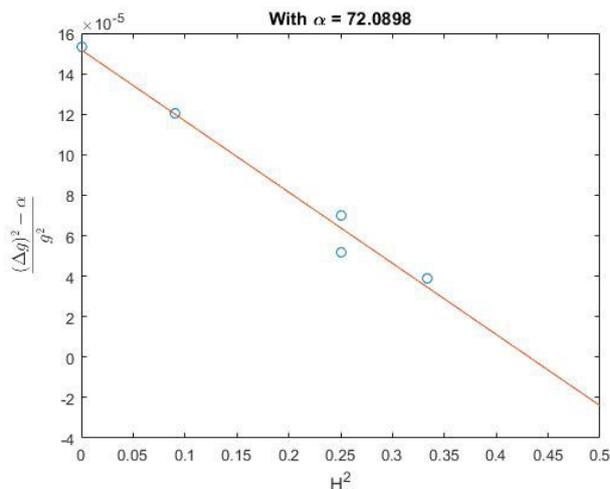


Figure 73: Determination of the parameter q for sample P91_12h_760C

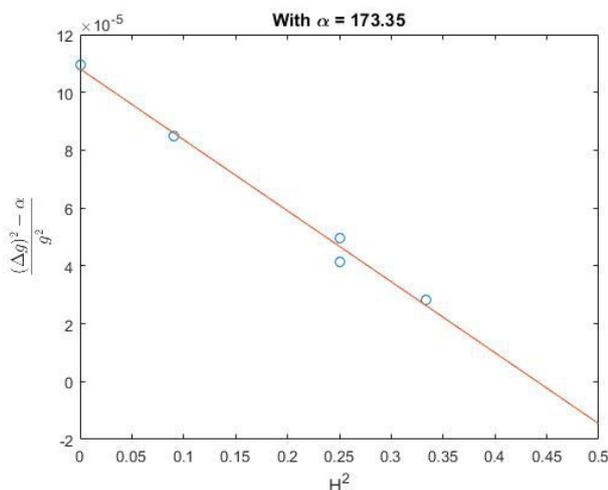


Figure 74: Determination of the parameter q for sample P91_24h_760C

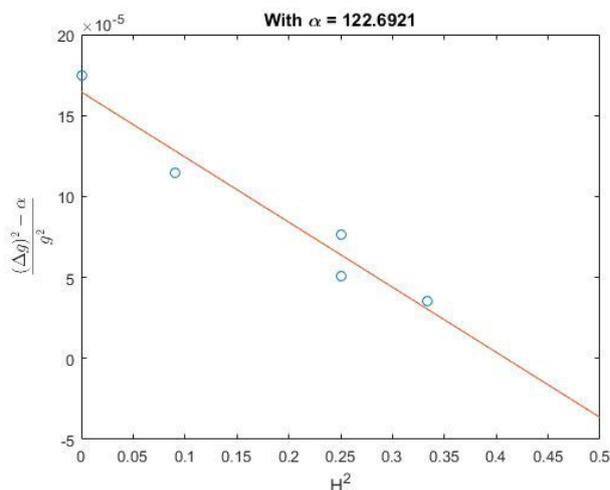


Figure 75: Determination of the parameter q for sample P91_36h_760C

9.6 Matlab codes

9.6.1 Peak fitting

To run the programs please also download the routines "pseudoVoigtAsym", "pseudoVoigtAsymDeriv" and "pseudoVoigtAsymFit" from <http://www.xray.cz/xrdmlread/Add-ons.htm>. Also the files "pseudoVoigtAsym_C_G_separate" and "pseudoVoigtAsym_separate" are needed (see below). These two files are mostly identical to "pseudoVoigtAsym" but have been adapted by the author of this work.

9.6.2 pseudoVoigtAsym_C_G_separate

The Matlab code for "pseudoVoigtAsym_C_G_separate" is:

```

function [y,y_c,y_g] = pseudoVoigtAsym(a,x)
%-----
%%%----This file was downloaded from
%%%http://www.xray.cz/xrdmlread/Add-ons.htm
%%%and the author only adapted minor changes!!!----
%-----
% PSEUDOVOIGTASYM    Calculate values of the asymmetric
%                    pseudo-Voigt function.
%
% [y] = pseudoVoigtAsym(a,x)
%
% a - parameters of the asymmetric pseudo-Voigt function
% x - a (row) vector of points where the asymmetric
%     pseudo-Voigt should be calculated.
%
% y - calculated values of the asym. pseudo-Voigt function
%
% example: a Gauss-like peak
%
%           x = [-10:0.1:10];
%           a = [1.0, 2.0, 1.5, 0.0, 1.0];
%           y = pseudoVoigtAsym(a,x);
%           plot(x,y)
%
% The integral of the asym. pseudo-Voigt function is normalized
% to one and hence the function is defined as:
%
% 
$$y = a_1 * \left\{ (1-a_4) * \sqrt{\log(2)/\pi/ss^2} * \exp[-\log(2)/s^2*(x-a_2)^2] \right.$$

% + 
$$a_4 * 1/\pi/ss * 1 / [ 1 + (x-a_2)^2/s^2 ] \left. \right\},$$

%
% where ss and s are defined as follows:
%
%           
$$ss = 1/4*(2 + a_5 + 1/a_5)*a_3$$

%
%           
$$s = 1/2*(1 + a_5)*a_3 \quad \text{for } x < a_2,$$

%           
$$s = 1/2*(1 + 1/a_5)*a_3 \quad \text{for } x \geq a_2.$$

%
% Parameters have this meaning:
%
% a1 - integrated intensity
% a2 - maximum position
% a3 - HWHM - half width in the half of maximum
%     (for symmetrical profiles)
% a4 - Cauchy/Gauss component weight (0 - Gauss, 1 - Cauchy)
% a5 - asymmetry (1 - symmetric)
%
% FWHM = 1/2*(2+a5+1/a5)*a3
%
% Function parametes (a) should be optionally stored in

```

```

% a vector of the size (1 x 5). If input parameters (a)
% represent a matrix of size (n x 5), n pseudo-Voigt
% functions are returned in n rows of the output vector y.
%
% X-ray diffraction: global WAVELENGTHS
%
% Function support a global variable WAVELENGTHS that
% represent an array of size (m x 2), where m is a number
% of spectral lines. (In scripts if for example sigle peaks
% are generated take caution that these variables were not
% previously defined by another script.) The WAVELENGTHS matrix
% has this interpretation:
%
% the 1st column - relative line intensity
% the 2st column - delta(Lambda)/Lambda ratio
%                   (Lambda is an averaged wavelength)
%
% example: CuKalpha1 + CuKalpha1 (Lambda = 1.5419)
%
%           WAVELENGTHS =
%           1.0000  -0.0008
%           0.5000   0.0017
%
% Ref.: The function has been adapted from
% [1] H. Toraya, J.Appl.Cryst 23 (1990), 485-491
% [2] ObjCryst/FOX - V. Favre-Nicolin, J.Appl.Cryst. 35 (2002), 734-743
%
% See also pseudoVoigtAsymDeriv, pseudVoigtAsymFit
%
% version 1.1, 19.11.2013, (c) Zdenek Matej
%-----
global WAVELENGTH_SEPARATE
global normierung
% check parameters
if ~exist('WAVELENGTH_SEPARATE','var') | isempty(WAVELENGTH_SEPARATE)
    WAVELENGTH_SEPARATE=[1.0 0.0];
end
if size(x,1)>1 x = reshape(x,1,[]); end
if size(a,2)~=5 a = reshape(a,[],5); end
% calc. all sets (sum over all spectral lines)
y = zeros(size(a,1),length(x));
y_c = zeros(size(a,1),length(x));
y_g = zeros(size(a,1),length(x));
for j=1:size(a,1)
    for k=1:size(WAVELENGTH_SEPARATE,1)
        a2 = a(j,2)+tan(a(j,2)*pi/360)*WAVELENGTH_SEPARATE(k,2)*360/pi;
        Lind = x < a(j,2);
        s1 = (1. + a(j,5))*a(j,3)/2;
        s2 = (1. + 1./a(j,5))*a(j,3)/2;
        ss = (s1 + s2)/2;
    end
end

```

```

c = (1-a(j,4))*sqrt(log(2)/pi/ss^2);
y_g(j, Lind) = y_g(j, Lind) + WAVELENGTH_SEPARATE(k,1)*c*...
    exp(-log(2)/s1^2*(x(Lind)-a2).^2);
y_g(j,~Lind) = y_g(j,~Lind) + WAVELENGTH_SEPARATE(k,1)*c*...
    exp(-log(2)/s2^2*(x(~Lind)-a2).^2);
c = a(j,4)/pi/ss;
y_c(j, Lind) = y_c(j, Lind) + WAVELENGTH_SEPARATE(k,1)*...
    c./(1+(x(Lind)-a2).^2/s1^2);
y_c(j,~Lind) = y_c(j,~Lind) + WAVELENGTH_SEPARATE(k,1)*...
    c./(1+(x(~Lind)-a2).^2/s2^2);

y(j,:) = y_c(j,:) + y_g(j,:);
end
y_c(j,:) = a(j,1)*y_c(j,)/normierung;
y_g(j,:) = a(j,1)*y_g(j,)/normierung;
y(j,:) = a(j,1)*y(j,)/normierung;
end
return;

```

9.6.3 pseudoVoigtAsym_separate

The Matlab code for "pseudoVoigtAsym_separate" is:

```

function [y] = pseudoVoigtAsym(a,x)
%-----
%%%----This file was downloaded from
%%%http://www.xray.cz/xrdmlread/Add-ons.htm
%%%and the author only adapted minor changes!!!----
%-----
% PSEUDOVOIGTASYM    Calculate values of the asymmetric
%                    pseudo-Voigt function.
%
% [y] = pseudoVoigtAsym(a,x)
%
% a - parameters of the asymmetric pseudo-Voigt function
% x - a (row) vector of points where the asymmetric
%     pseudo-Voigt should be calculated.
%
% y - calculated values of the asym. pseudo-Voigt function
%
% example: a Gauss-like peak
%
%         x = [-10:0.1:10];
%         a = [1.0, 2.0, 1.5, 0.0, 1.0];
%         y = pseudoVoigtAsym(a,x);
%         plot(x,y)
%
% The integral of the asym. pseudo-Voigt function is normalized
% to one and hence the function is defined as:
%

```

```

% y = a1 * { (1-a4) * sqrt[log(2)/pi/ss^2]
%           * exp[-log(2)/s^2*(x-a2)^2]
%   +      a4 * 1/pi/ss * 1 / [ 1 + (x-a2)^2/s^2 ] },
%
% where ss and s are defined as follows:
%
%           ss = 1/4*(2 + a5 + 1/a5)*a3
%
%           s = 1/2*(1 + a5)*a3           for x < a2,
%           s = 1/2*(1 + 1/a5)*a3       for x >= a2.
%
% Parameters have this meaning:
%
% a1 - integrated intensity
% a2 - maximum position
% a3 - HWHM - half width in the half of maximum
%       (for symmetrical profiles)
% a4 - Cauchy/Gauss component weight (0 - Gauss, 1 - Cauchy)
% a5 - asymmetry (1 - symmetric)
%
% FWHM = 1/2*(2+a5+1/a5)*a3
%
% Function parametes (a) should be optionally stored in
% a vector of the size (1 x 5). If input parameters (a)
% represent a matrix of size (n x 5), n pseudo-Voigt
% functions are returned in n rows of the output vector y.
%
% X-ray diffraction: global WAVELENGTHS
%
% Function support a global variable WAVELENGTHS that
% represent an array of size (m x 2), where m is a number
% of spectral lines. (In scripts if for example sigle peaks
% are generated take caution that these variables were not
% previously defined by another script.) The WAVELENGTHS matrix
% has this interpretation:
%
% the 1st column - relative line intensity
% the 2st column - delta(Lambda)/Lambda ratio
%                 (Lambda is an averaged wavelength)
%
% example: CuKalpha1 + CuKalpha1 (Lambda = 1.5419)
%
%           WAVELENGTHS =
%           1.0000   -0.0008
%           0.5000    0.0017
%
% Ref.: The function has been adapted from
% [1] H. Toraya, J.Appl.Cryst 23 (1990), 485-491
% [2] ObjCryst/FOX - V. Favre-Nicolin, J.Appl.Cryst. 35 (2002), 734-743
%

```

```

% See also pseudoVoigtAsymDeriv, pseudVoigtAsymFit
%
% version 1.1, 19.11.2013, (c) Zdenek Matej
%-----
global WAVELENGTH_SEPARATE
global normierung
% check parameters
if ~exist('WAVELENGTH_SEPARATE','var') | isempty(WAVELENGTH_SEPARATE)
    WAVELENGTH_SEPARATE=[1.0 0.0];
end
if size(x,1)>1 x = reshape(x,1,[]); end
if size(a,2)~=5 a = reshape(a,[],5); end
% calc. all sets (sum over all spectral lines)
y = zeros(size(a,1),length(x));
for j=1:size(a,1)
    for k=1:size(WAVELENGTH_SEPARATE,1)
        a2 = a(j,2)+tan(a(j,2)*pi/360)*WAVELENGTH_SEPARATE(k,2)*360/pi;
        Lind = x < a(j,2);
        s1 = (1. + a(j,5))*a(j,3)/2;
        s2 = (1. + 1./a(j,5))*a(j,3)/2;
        ss = (s1 + s2)/2;
        c = (1-a(j,4))*sqrt(log(2)/pi/ss^2);
        y(j, Lind) = y(j, Lind) + WAVELENGTH_SEPARATE(k,1)*c*...
            exp(-log(2)/s1^2*(x(Lind)-a2).^2);
        y(j,~Lind) = y(j,~Lind) + WAVELENGTH_SEPARATE(k,1)*c*...
            exp(-log(2)/s2^2*(x(~Lind)-a2).^2);
        c = a(j,4)/pi/ss;
        y(j, Lind) = y(j, Lind) + WAVELENGTH_SEPARATE(k,1)*...
            c./(1+(x(Lind)-a2).^2/s1^2);
        y(j,~Lind) = y(j,~Lind) + WAVELENGTH_SEPARATE(k,1)*...
            c./(1+(x(~Lind)-a2).^2/s2^2);
    end
    y(j,:) = a(j,1)*y(j,:)/normierung;
    %y(j,:) = a(j,1)*y(j,:)/sum(WAVELENGTH_SEPARATE(:,1));
end
return;

```

9.6.4 LaB₆ peak fit

The Matlab code for the LaB₆ peak fitting routine is:

```

close all
clear all
clc
%-----Specify x-ray radiation-----
Cu_K_alpha1 = 1.540598*10^-10; % Cu_K_alpha1 wavelength
Cu_K_alpha2 = 1.544426*10^-10; % Cu_K_alpha2 wavelength
intensity_Kalpha1 = 1; % Cu_K_alpha1 intensity
intensity_Kalpha2 = 0.5; % Cu_K_alpha2 intensity

```

```

global WAVELENGTH_SEPARATE; % For separate K_alpha1/K_alpha2 calculations
global normierung          % For Scaling area under peaks with intensity
normierung = intensity_Kalpha1 + intensity_Kalpha2;

global WAVELENGTHS        %For combined K_alpha1/K_alpha2 calculations
Cu_K_alpha = (intensity_Kalpha1*Cu_K_alpha1 + intensity_Kalpha2...
              * Cu_K_alpha2) / normierung;
delta_Cu_K_alpha1 = (Cu_K_alpha1 - Cu_K_alpha);
delta_Cu_K_alpha2 = (Cu_K_alpha2 - Cu_K_alpha);
WAVELENGTHS = [intensity_Kalpha1 delta_Cu_K_alpha1/Cu_K_alpha;...
              intensity_Kalpha2 delta_Cu_K_alpha2/Cu_K_alpha];

% Import data: When measuring the whole data at once, use the M=... ,
% otherwise import the data with M_xx=... in separate files.

%M = load('LaB6_standard_parallel_beam_20h.xy','-ascii');

M_21 = load('LaB6_18-24_SLD_mono.xy','-ascii');
M_36_58 = load('LaB6_36-58_SLD_mono.xy','-ascii');
M_74_81 = load('LaB6_74-81_SLD_mono.xy','-ascii');
M_82_142 = load('LaB6_82-142_SLD_mono.xy','-ascii');

%peaknamen = [21 30 37 43 49 54 76 80 84 88 96 100 104 108 112 121 ...
% 131 136 142]; % If all peaks can be analysed

peaknamen = [21 37 43 49 54 76 80 84 88 96 100 104 108 112 116 121 131 ...
136 142]; % If the 30° peaks is too small to be analysed. Often
% happens when measuring with the physics department's PANalytical
% x-ray-machine with monochromator.

% %Define evaluation range: Use this if all data was importet in one file:
% x_data_LaB6_21 = M(610:1050,1).';
% y_data_LaB6_21 = M(610:1050,2).';
% x_data_LaB6_30 = M(1350:1715,1).';
% y_data_LaB6_30 = M(1350:1715,2).';
% x_data_LaB6_37 = M(1905:2290,1).';
% y_data_LaB6_37 = M(1905:2290,2).';
% x_data_LaB6_43 = M(2360:2745,1).';
% y_data_LaB6_43 = M(2360:2745,2).';
% x_data_LaB6_49 = M(2820:3125,1).';
% y_data_LaB6_49 = M(2820:3125,2).';
% x_data_LaB6_54 = M(3200:3505,1).';
% y_data_LaB6_54 = M(3200:3505,2).';
% x_data_LaB6_76 = M(4875:5180,1).';
% y_data_LaB6_76 = M(4875:5180,2).';
% x_data_LaB6_80 = M(5180:5485,1).';
% y_data_LaB6_80 = M(5180:5485,2).';
% x_data_LaB6_84 = M(5485:5790,1).';

```

```

% y_data_LaB6_84 = M(5485:5790,2).';
% x_data_LaB6_88 = M(5790:6095,1).';
% y_data_LaB6_88 = M(5790:6095,2).';
% x_data_LaB6_96 = M(6400:6705,1).';
% y_data_LaB6_96 = M(6400:6705,2).';
% x_data_LaB6_100 = M(6705:7010,1).';
% y_data_LaB6_100 = M(6705:7010,2).';
% x_data_LaB6_104 = M(7010:7315,1).';
% y_data_LaB6_104 = M(7010:7315,2).';
% x_data_LaB6_108 = M(7315:7620,1).';
% y_data_LaB6_108 = M(7315:7620,2).';
% x_data_LaB6_112 = M(7620:7925,1).';
% y_data_LaB6_112 = M(7620:7925,2).';
% x_data_LaB6_121 = M(8300:8610,1).';
% y_data_LaB6_121 = M(8300:8610,2).';
% x_data_LaB6_131 = M(9065:9300,1).';
% y_data_LaB6_131 = M(9065:9300,2).';
% x_data_LaB6_136 = M(9445:9750,1).';
% y_data_LaB6_136 = M(9445:9750,2).';
% x_data_LaB6_142 = M(9830:10285,1).';
% y_data_LaB6_142 = M(9830:10285,2).';

%Define evaluation range: Use this if data was importet in separated files:
x_data_LaB6_21 = M_21(210:end,1).';
y_data_LaB6_21 = M_21(210:end,2).';
x_data_LaB6_37 = M_36_58(55:245,1).';
y_data_LaB6_37 = M_36_58(55:245,2).';
x_data_LaB6_43 = M_36_58(510:702,1).';
y_data_LaB6_43 = M_36_58(510:702,2).';
x_data_LaB6_49 = M_36_58(892:1160,1).';
y_data_LaB6_49 = M_36_58(892:1160,2).';
x_data_LaB6_54 = M_36_58(1273:1578,1).';
y_data_LaB6_54 = M_36_58(1273:1578,2).';
x_data_LaB6_76 = M_74_81(1:215,1).';
y_data_LaB6_76 = M_74_81(1:215,2).';
x_data_LaB6_80 = M_74_81(344:496,1).';
y_data_LaB6_80 = M_74_81(344:496,2).';
x_data_LaB6_84 = M_82_142(39:191,1).';
y_data_LaB6_84 = M_82_142(39:191,2).';
x_data_LaB6_88 = M_82_142(306:534,1).';
y_data_LaB6_88 = M_82_142(306:534,2).';
x_data_LaB6_96 = M_82_142(953:1105,1).';
y_data_LaB6_96 = M_82_142(953:1105,2).';
x_data_LaB6_100 = M_82_142(1182:1448,1).';
y_data_LaB6_100 = M_82_142(1182:1448,2).';
x_data_LaB6_104 = M_82_142(1486:1791,1).';
y_data_LaB6_104 = M_82_142(1486:1791,2).';
x_data_LaB6_108 = M_82_142(1829:2057,1).';
y_data_LaB6_108 = M_82_142(1829:2057,2).';
x_data_LaB6_112 = M_82_142(2134:2362,1).';

```

```

y_data_LaB6_112 = M_82_142(2134:2362,2).';
x_data_LaB6_116 = M_82_142(2438:2743,1).';
y_data_LaB6_116 = M_82_142(2438:2743,2).';
x_data_LaB6_121 = M_82_142(2819:3047,1).';
y_data_LaB6_121 = M_82_142(2819:3047,2).';
x_data_LaB6_131 = M_82_142(3543:3809,1).';
y_data_LaB6_131 = M_82_142(3543:3809,2).';
x_data_LaB6_136 = M_82_142(3961:4190,1).';
y_data_LaB6_136 = M_82_142(3961:4190,2).';
x_data_LaB6_142 = M_82_142(4380:end,1).';
y_data_LaB6_142 = M_82_142(4380:end,2).';

% Give approximate positions of peak maxima in 2Theta:
peakmax_data_21 = 21.4;
%peakmax_data_30 = 30.4;
peakmax_data_37 = 37.4;
peakmax_data_43 = 43.5;
peakmax_data_49 = 49.0;
peakmax_data_54 = 54.0;
peakmax_data_76 = 75.9;
peakmax_data_80 = 79.9;
peakmax_data_84 = 83.9;
peakmax_data_88 = 87.8;
peakmax_data_96 = 95.7;
peakmax_data_100 = 99.7;
peakmax_data_104 = 103.7;
peakmax_data_108 = 107.8;
peakmax_data_112 = 112.0;
peakmax_data_116 = 116.4;
peakmax_data_121 = 120.9;
peakmax_data_131 = 130.6;
peakmax_data_136 = 136.0;
peakmax_data_142 = 142.0;

fitting_iterations = 3000; % Define how many sets of starting parameters
% for "Monte-Carlo-method" should be given.

%-----Begin of peak evaluation-----
%!!!!Important:!!!!
% a(1) - integrated intensity
% a(2) - maximum position
% a(3) - HWHM - half width in the half of maximum
%           (for symmetrical profiles)
% a(4) - Cauchy/Gauss component weight (0 - Gauss, 1 - Cauchy)
% a(5) - asymmetry (1 - symmetric)

% For-loop over all peaks:
for ges = 1 : numel(peaknamen)
    % Get the right peak maximum for the peak:

```

```

    peakmax_data = eval(['peakmax_data_' int2str(peaknamen(ges))]);
    % Get right data range for the peak:
    x_data = eval(['x_data_LaB6_' int2str(peaknamen(ges))]);
    y_data = eval(['y_data_LaB6_' int2str(peaknamen(ges))]);

% Estimate starting parameters:
background = 1000; % Variable for background estimation in counts
x = 0.5;          % Chauchy/Gauss (see a(4))
II_max = 120000; % Estimated maximum of integrated intensity in counts
HWHM_max = 0.1;  % Estimated Half-Width-at-Half-Maximum in 2Theta
HWHM_var = 0.1;  % Variation around the estimated HWHM in 2Theta
peakvar = 0.3;   % Variation of the estimated peak maxima

mu(1) = (x_data(1)+x_data(end))/2; % (see function: pseudoVoigtAsym.m)
mu(2) = x_data(end)-x_data(1);     % (see function: pseudoVoigtAsym.m)

Linda = [1 1 1 1 1]; % Determine which parts of a(1-5) should be fitted:
                    % 1-Perform fit, 0-do not fit
b0 = [0 0 background]; % Guess background function y=b0(1)x^2+b0(2)x+b0(3)
Lindb = [1 1 1]; % Determine which parts of background b(1-3) should
                % be fitted: 1-Perform fit, 0-do not fit
w = 1./y_data; % (see function: pseudoVoigtAsym.m)

% Define variables to be filled later:
rwp_final = 1000;
a_final = [0,0,0,0,0];
b_final = NaN;
a0_final = [0,0,0,0,0];
counter = 1;

a_final_faul = zeros(4,5);
IB_ges_faul = zeros(4,1);
IB_cauchy_faul = zeros(4,1);
IB_gauss_faul = zeros(4,1);
b_final_faul = zeros(4,3);

% For-loop to do each data processing 4 times to get a feeling of the fit
% stability:
for faul = 1:4

% Perform fitting with "Monte-Carlo-Simulation":
for counter = 1:fitting_iterations

    % Randomize starting parameters:
    h = rand * II_max; % Integrated intensity
    j = HWHM_max + (2*rand -1) * HWHM_var; % HWHM
    i = peakmax_data + (2*rand -1) * peakvar; % Peak maximum in 2Theta
    k = x + (2*rand -1) * 0.5; % Chauchy/Gauss weight
                                % (0 - Gauss, 1 - Cauchy)
    asym = 1 + (2*rand -1) * 0.1; % Asymmetry (1 - symmetric)

```

```

a0 = [h, i, j, k, asym];

% Perform fitting:
[a_fit,b_fit,~,~,~,rwp] =...
    pseudoVoigtAsymFit(x_data,y_data,w,a0,b0,mu,Linda,Lindb);

if 0<=a_fit(4) % Make sure peak shape is exact or between Gauss/Cauchy
if a_fit(4)<=1 % Make sure peak shape is exact or between Gauss/Cauchy

    if rwp < rwp_final % Only save if fit is better than
        a_final = a_fit; % previously best one.
        b_final = b_fit;
        a0_final = a0;
        rwp_final = rwp;
        intensity_Kalpha2_final = intensity_Kalpha2;
    end
end
end
counter = counter +1
end

%-----Define parameters for plotting:-----
% Draw background function:
background = polyval(b_final,x_data,[],mu);
% Complete fit including Gauss & Cauchy and K_alpha1 & K_alpha2 parts:
yc_fit = sum(pseudoVoigtAsym(a_final,x_data),1)...
    +background; % Convert data fit into curve
yc_schaetz = sum(pseudoVoigtAsym(a0_final,x_data),1)...
    +polyval(b0,x_data,[],mu); % Convert starting parameters for best fit
    % into curve

%Create curve for Gauss & Cauchy and K_alpha1 (NO K_alpha2!):
WAVELENGTH_SEPARATE = [intensity_Kalpha1 delta_Cu_K_alpha1/Cu_K_alpha];
yc_fit_Kalpha1 = sum(pseudoVoigtAsym_separate(a_final,x_data),1)...
    +background;
%Also separate Gauss and Cauchy parts (of K_alpha1 peak):
[~,ycauchy1,ygauss1] = pseudoVoigtAsym_C_G_separate(a_final,x_data);
%Find maximum of K_alpha1 peak:
ind_ges1 = find(pseudoVoigtAsym_separate(a_final,x_data)>=...
    max(pseudoVoigtAsym_separate(a_final,x_data)));

%Create curve for Gauss & Cauchy and K_alpha2 (NO K_alpha1!):
WAVELENGTH_SEPARATE = [intensity_Kalpha2 delta_Cu_K_alpha2/Cu_K_alpha];
yc_fit_Kalpha2 = sum(pseudoVoigtAsym_separate(a_final,x_data),1)...
    +background;
%Again separate Gauss and Cauchy parts (of K_alpha2 peak):
[y,ycauchy2,ygauss2] = pseudoVoigtAsym_C_G_separate(a_final,x_data);
%Combine Cauchy_K_alpha1 and Cauchy_K_alpha2:

```

```

ycauchy = ycauchy1 + ycauchy2;
%Combine Gauss_K_alpha1 and Gauss_K_alpha2:
ygauss = ygauss1 + ygauss2;
%Combine Kalpha1 and Kalpha2 fit to make sure they give the same result as
%the combined fit:
yc_fit_Kalpha_test = yc_fit_Kalpha1 + yc_fit_Kalpha2...
    -background;
%Check if Gauss/Cauchy separation worked correctly. If it works for Kalpha2
%peak, it is assumed that it also worked for Kalpha1:
if y ~= pseudoVoigtAsym_separate(a_final,x_data)
    errormessage=['Der Funktionierer ist kaputt!(Gauss+Lorentz-Kurve '...
        '~= Gesamtfit) \nThis usually happes if the fit function did '...
        'not find a minimum'];
    error(errormessage,[ ])
end

% Get 2Theta angle of overall peak maximum (K_alpha1&2 together):
x1 = x_data(ind_ges1);

% Sum Cauchy parts and background for plotting:
ycauchy_plot = sum(ycauchy,1)+background;
% Find maximum position of Cauchy K_alpha1 peak:
ind_cauchy = find(ycauchy1>=max(ycauchy1));
% Sum Gauss parts and background for plotting:
ygauss_plot = sum(ygauss,1)+background;
% Find maximum position of Gauss K_alpha1 peak:
ind_gauss = find(ygauss1>=max(ygauss1));

%---Difference plot-----
y_diff = y_data - yc_fit;

%!!!!!!!!!!!!!!IMPORTANT!!!!!!!!!!!!!!
% The plots here are done for every of of the 4 repetitions, one can
% visually analyse each peak and search for problems if something big
% goes wrong! For normal data analysis the two plots at the end should do.

% % This Plot is to compare the data with the fit, the K_alpha1 and
% % K_alpha2 peaks separately, the background function and the peak
% % maximum:

% figure
% semilogy(x_data,y_data,'k*')
% hold on
% semilogy(x_data,yc_fit,'m')
% semilogy(x_data,yc_fit_Kalpha1,'r')
% semilogy(x_data,background,'y')
% line([x1 x1], get(gca, 'ylim'),'color','c','LineStyle','-.');
% semilogy(x_data,yc_fit_Kalpha2,'r')
% xlabel ('Winkel 2 \Theta [°]')

```

```

% ylabel ('Counts [a.u.]')
% legend ('Messdaten', 'Gefittete Kurve', 'K_{\alpha1} und K_{\alpha2}...
% separat gefittet', 'Untergrund', 'Maximum')
% hold off

% % The next Plot is to compare the data with the fit, the starting
% % parameters that gave the best fit results, the background function, the
% % peak maximum and to check if separation has worked properly (The fittet
% % curve should be overplottet by a curve that is the sum of all
% % previously separated K_alpha1+K_alpha2 curves).
%
% figure
% semilogy(x_data,y_data,'k*')
% hold on
% semilogy(x_data,yc_schaetz,'g')
% semilogy(x_data,yc_fit,'m')
% semilogy(x_data,yc_fit_Kalpha_test,'b')
% semilogy(x_data,background,'y')
% line([x1 x1], get(gca, 'ylim'),'color','c','LineStyle','-.');
% xlabel ('Winkel 2 \Theta [°]')
% ylabel ('Counts [a.u.]')
% legend ('Messdaten', 'Geschaetzte Anfangsparameter', 'Gefittete Kurve',...
% 'K_{\alpha1} + K_{\alpha2}', 'Untergrund', 'Maximum')
% hold off

% % The next Plot is to compare the data with the fit, the background
% % function, the peak maximum, Cauchy and Gauss peaks separately and to
% % check if separation has worked properly (The fittet curve should be
% % overplottet by a curve that is the sum of the previously separated
% % Cauchy- and Gauss-curves).

% figure
% semilogy(x_data,y_data,'k*')
% hold on
% semilogy(x_data,yc_fit,'m')
% %semilogy(x_data,yc_fit_Kalpha1,'r')
% semilogy(x_data,background,'y')
% %semilogy(x_data,yc_fit_Kalpha_test,'b')
% line([x1 x1], get(gca, 'ylim'),'color','c','LineStyle','-.');
% semilogy(x_data,ygauss_plot,'g')
% semilogy(x_data,ycauchy_plot,'c')
% semilogy(x_data,ygauss_plot+ycauchy,'r')
% %semilogy(x_data,yc_fit_Kalpha2,'r')
% xlabel ('Winkel 2 \Theta [°]')
% ylabel ('Counts [a.u.]')
% legend ('Messdaten', 'Gefittete Kurve', 'Untergrund', 'Maximum',...
% 'Gaussanteil', 'Cauchyanteil', 'Cauchy+Gaussanteil')

```

```

% hold off

% % Next is a LINEAR plot with just the data and the fit together with a
% % difference plot to evaluate the fit quality.

% figure
% pos1 = [0.1 0.25 0.8 0.68];
% subplot('Position',pos1)
% plot(x_data,y_data,'k*')
% hold on
% plot(x_data,yc_fit,'m')
% ylabel ('Counts [a.u.]')
% legend ('Messdaten','Gefittete Kurve')
% hold off
% set(gca,'xtick',[])
% pos2 = [0.1 0.13 0.8 0.10];
% subplot('Position',pos2)
% plot(x_data,y_diff,'b')
% xlabel ('Winkel 2 \Theta [°]')

%-----Calculate integral breadths-----
%Calculate height of the peak as seen in the plot (K_alpha1 and K_alpha2
%usually are not separated):
height_ges = yc_fit(ind_ges1) - background (ind_ges1);
%Calculate height of Cauchy peak:
height_cauchy = ycauchy(ind_cauchy);
%Calculate height of Gauss peak:
height_gauss = ygauss(ind_gauss);
%Calculate area under Cauchy peak:
A_cauchy = a_final(1) * a_final(4);
%Calculate area under Gauss peak:
A_gauss = a_final(1) * (1 - a_final(4));

%Calculate integral breadth of peak as seen in the plot (K_alpha1 and
%K_alpha2 usually are not separated):
IB_standard_ges = a_final(1)/height_ges;
%Calculate integral breadth of Cauchy peak:
IB_standard_cauchy = A_cauchy ./ height_cauchy;
%Calculate integral breadth of Gauss peak:
IB_standard_gauss = A_gauss ./ height_gauss;
%If Cauchy or Gauss peaks have no maximum, show a message:
if numel(IB_standard_cauchy)>1
    IB_standard_cauchy = max(IB_standard_cauchy)
    disp('IB_standard_cauchy hat kein maximum')
elseif numel(IB_standard_gauss) > 1
    IB_standard_gauss = max(IB_standard_gauss)

```

```

    disp('IB_standard_gauss hat kein maximum')
end

% Save the results before restarting the fitting procedure:
a_final_faul(faul,:) = a_final;
IB_ges_faul(faul,:) = IB_standard_ges;
IB_cauchy_faul(faul,:) = IB_standard_cauchy;
IB_gauss_faul(faul,:) = IB_standard_gauss;
b_final_faul(faul,:) = b_final;

faul = faul + 1

end
%----After all 4 repetitions are done, plot the results to compare fit----
%-----stability:-----

% Create variables to be filled in loop:
yc_fit_faul = zeros(4,numel(y_data));
y_diff_faul = zeros(4,numel(y_data));
y_data1 = [y_data;y_data;y_data;y_data];
farbe = ['r','b','m','g'];
%----Plotten-----
figure
pos3 = [0.1 0.25 0.8 0.68]; % Define position and height for upper plot
                        % [x y width height]
subplot('Position',pos3)
plot(x_data,y_data,'k*')          %Plot data
hold on
for i = 1:4
% Save fit data for all 4 repetitions:
yc_fit_faul = sum(pseudoVoigtAsym(a_final_faul(i,:),x_data),1)...
    +polyval(b_final_faul(i,:),x_data,[],mu);
% Create difference plot for all 4 repetitions:
y_diff_faul(i,:) = y_data1(i,:) - yc_fit_faul;

plot(x_data,yc_fit_faul,farbe(i))
ylabel ('Counts [a.u.]')
set(gca,'xtick',[])
end
legend ('Messdaten','Gefittete Kurve 1','Gefittete Kurve 2',...
    'Gefittete Kurve 3','Gefittete Kurve 4')
title([int2str(peaknamen(ges)) '° Peak'])

pos4 = [0.1 0.13 0.8 0.10]; % Define position and height for lower plot
                        % [x y width height]
subplot('Position',pos4)

%Plot the difference plot:
for i = 1:4

```

```

plot(x_data,y_diff_faul(i,:),farbe(i))
hold on
end
xlabel ('Winkel 2 \Theta [°]')
hold off

%Also plot data, fit, K_alpha1, K_alpha2, background and peam maximum:
figure
semilogy(x_data,y_data,'k*')
hold on
semilogy(x_data,yc_fit,'m')
semilogy(x_data,yc_fit_Kalpha1,'r')
semilogy(x_data,background,'y')
line([x1 x1], get(gca, 'ylim'),'color','c','LineStyle','-');
semilogy(x_data,yc_fit_Kalpha2,'r')

xlabel ('Winkel 2 \Theta [°]')
ylabel ('Counts [a.u.]')
legend ('Messdaten','Gefittete Kurve',...
        'K_{\alpha1} und K_{\alpha2} separat gefittet','Untergrund','Maximum')
hold off

% Create output variable that can be copied into a .txt file and contains
% all important information. If this is done, the mWH/mWA data processing
% routine can import needed data easily:
if ges == 1
    daten_exportieren =...
        [IB_ges_faul IB_cauchy_faul IB_gauss_faul a_final_faul];
else
daten_exportieren =...
    [daten_exportieren;IB_ges_faul IB_cauchy_faul IB_gauss_faul a_final_faul];
end
end
end

```

9.6.5 Steel_peak_fit

The Matlab code for the steel peak fitting routine is:

```

close all
clear all
clc
%-----Specify x-ray radiation-----
Cu_K_alpha1 = 1.540598*10^-10; % Cu_K_alpha1 wavelength
Cu_K_alpha2 = 1.544426*10^-10; % Cu_K_alpha2 wavelength
intensity_Kalpha1 = 1;          % Cu_K_alpha1 intensity
intensity_Kalpha2 = 0.5;       % Cu_K_alpha2 intensity

global WAVELENGTH_SEPARATE;    % For separate K_alpha1/K_alpha2 calculations
global normierung              % For Scaling area under peaks with intensity

```

```

normierung = intensity_Kalpha1 + intensity_Kalpha2;

global WAVELENGTHS      %For combined K_alpha1/K_alpha2 calculations
Cu_K_alpha = (intensity_Kalpha1*Cu_K_alpha1 + intensity_Kalpha2...
              * Cu_K_alpha2) / normierung;
delta_Cu_K_alpha1 = (Cu_K_alpha1 - Cu_K_alpha);
delta_Cu_K_alpha2 = (Cu_K_alpha2 - Cu_K_alpha);
WAVELENGTHS = [intensity_Kalpha1 delta_Cu_K_alpha1/Cu_K_alpha;...
               intensity_Kalpha2 delta_Cu_K_alpha2/Cu_K_alpha];

% Import data: When measuring the whole data at once, use the M=... ,
% otherwise import the data with M_xx=... in separate files.
%M = load('BM_AR_vib_pol_parallel_beam_20h.xy','-ascii');
%M_45 = load('XXXXXXXXXX.xy','-ascii');
M_65 = load('P91_36h_760C_60_to_70.xy','-ascii');
M_82 = load('P91_36h_760C_77_to_87.xy','-ascii');
M_99 = load('P91_36h_760C_94_to_104.xy','-ascii');
M_116 = load('P91_36h_760C_111_to_121.xy','-ascii');
M_137 = load('P91_36h_760C_132_to_142.xy','-ascii');

%peaknamen = [45 65 82 99 116 137]; %If 45° Peak is not corrupted
peaknamen = [65 82 99 116 137]; %If 45° Peak is corrupted. Often happens
% when measuring with the physics department's PANalytical x-ray-machine
% with monochromator.

%Define evaluation range: Use this if all data was importet in one file:
% x_data_45 = M(2360:2895,1).';
% y_data_45 = M(2360:2895,2).';
% x_data_65 = M(4080:4342,1).';
% y_data_65 = M(4080:4342,2).';
% x_data_82 = M(5370:5637,1).';
% y_data_82 = M(5370:5637,2).';
% x_data_99 = M(6665:6950,1).';
% y_data_99 = M(6665:6950,2).';
% x_data_116 = M(7850:8300,1).';
% y_data_116 = M(7850:8300,2).';
% x_data_137 = M(9400:9900,1).';
% y_data_137 = M(9400:9900,2).';

%Define evaluation range: Use this if data was importet in separated files:
% x_data_45 = M_45(1:end,1).';
% y_data_45 = M_45(1:end,2).';
% x_data_65 = M_65(1:end,1).';
% y_data_65 = M_65(1:end,2).';
% x_data_82 = M_82(1:end,1).';
% y_data_82 = M_82(1:end,2).';
% x_data_99 = M_99(1:end,1).';
% y_data_99 = M_99(1:end,2).';
% x_data_116 = M_116(1:end,1).';
% y_data_116 = M_116(1:end,2).';

```

```

% x_data_137 = M_137(1:end,1).';
% y_data_137 = M_137(1:end,2).';

x_data_65 = M_65(210:end,1).';
y_data_65 = M_65(210:end,2).';
x_data_82 = M_82(230:690,1).';
y_data_82 = M_82(230:690,2).';
x_data_99 = M_99(1:430,1).';
y_data_99 = M_99(1:430,2).';
x_data_116 = M_116(115:end,1).';
y_data_116 = M_116(115:end,2).';
x_data_137 = M_137(1:end,1).';
y_data_137 = M_137(1:end,2).';

% Give approximate positions of peak maxima in 2Theta:
peakmax_data_45 = 44.6;
peakmax_data_65 = 64.9;
peakmax_data_82 = 82.2;
peakmax_data_99 = 98.8;
peakmax_data_116 = 116.2;
peakmax_data_137 = 136.8;

fitting_iterations = 20; % Define how many sets of starting parameters
% for "Monte-Carlo-method" should be given.

%-----Begin of peak evaluation-----
%!!!!Important!!!!
% a(1) - integrated intensity
% a(2) - maximum position
% a(3) - HWHM - half width in the half of maximum
%           (for symmetrical profiles)
% a(4) - Cauchy/Gauss component weight (0 - Gauss, 1 - Cauchy)
% a(5) - asymmetry (1 - symmetric)

% For loop over all peaks:
for ges = 1 : numel(peaknamen)
    % Get the right peak maximum for the peak:
    peakmax_data = eval(['peakmax_data_' int2str(peaknamen(ges))]);
    % Get right data range for the peak:
    x_data = eval(['x_data_' int2str(peaknamen(ges))]);
    y_data = eval(['y_data_' int2str(peaknamen(ges))]);

% Estimate starting parameters:
background = 14000; % Variable for background estimation in counts
x = 0.5;           % Chauchy/Gauss (see a(4))
II_max = 120000;  % Estimated maximum of integrated intensity in counts
HWHM_schaetz = 0.7; % Estimater Half-Width-at-Half-Maximum in 2Theta
HWHM_var = 0.7;   % Variation around the estimated HWHM in 2Theta
peakvar = 0.5;    % Variation of the estimated peak maxima

```

```

mu(1) = (x_data(1)+x_data(end))/2;  %(see function: pseudoVoigtAsym.m)
mu(2) = x_data(end)-x_data(1);      %(see function: pseudoVoigtAsym.m)

Linda = [1 1 1 1 1];    % Determine which parts of a(1-5) should be fitted:
                        % 1-Perform fit, 0-do not fit
b0 = [0 0 background];  % Guess background function y=b0(1)x^2+b0(2)x+b0(3)
Lindb = [1 1 1];       % Determine which parts of background b(1-3) should
                        % be fitted: 1-Perform fit, 0-do not fit
w = 1./y_data;         %(see function: pseudoVoigtAsym.m)

% Define variables to be filled later:
rwp_final = 1000;
a_final = [0,0,0,0,0];
b_final = NaN;
a0_final = [0,0,0,0,0];
counter = 1;

a_final_faul = zeros(4,5);
IB_ges_faul = zeros(4,1);
IB_cauchy_faul = zeros(4,1);
IB_gauss_faul = zeros(4,1);
b_final_faul = zeros(4,3);

% For-loop to do each data processing 4 times to get a feeling of the fit
% stability:
for faul = 1:4

% Perform fitting with "Monte-Carlo-Simulation":
for counter = 1:fitting_iterations

    % Randomize starting parameters:
    h = rand * II_max;          % Integrated intensity
    j = HWHM_schaetz + (2*rand -1) * HWHM_var; % HWHM
    i = peakmax_data + (2*rand -1) * peakvar; % Peak maximum in 2Theta
    k = x + (2*rand -1) * 0.5; % Chauchy/Gauss weight
                                % (0 - Gauss, 1 - Cauchy)
    asym = 1 + (2*rand -1) * 0.1; % Asymmetry (1 - symmetric)

    a0 = [h, i, j, k, asym];

% Perform fitting:
[a_fit,b_fit,~,~,~,rwp] =...
    pseudoVoigtAsymFit(x_data,y_data,w,a0,b0,mu,Linda,Lindb);

if 0<=a_fit(4) % Make sure peak shape is exact or between Gauss/Cauchy
if a_fit(4)<=1 % Make sure peak shape is exact or between Gauss/Cauchy

    if rwp < rwp_final % Only save if fit is better than
        a_final = a_fit; % previously best one.
        b_final = b_fit;

```

```

        a0_final = a0;
        rwp_final = rwp;
    end
end
end
counter = counter +1
end

%-----Define parameters for plotting:-----
% Draw background function:
background = polyval(b_final,x_data,[],mu);
% Complete fit including Gauss & Cauchy and K_alpha1 & K_alpha2 parts:
yc_fit = sum(pseudoVoigtAsym(a_final,x_data),1)...
    +background;          % Convert data fit into curve
yc_schaetz = sum(pseudoVoigtAsym(a0_final,x_data),1)...
    +polyval(b0,x_data,[],mu); % Convert starting parameters for best fit
                                % into curve

%Create curve for Gauss & Cauchy and K_alpha1 (NO K_alpha2!):
WAVELENGTH_SEPARATE = [intensity_Kalpha1 delta_Cu_K_alpha1/Cu_K_alpha];
yc_fit_Kalpha1 = sum(pseudoVoigtAsym_separate(a_final,x_data),1)...
    +background;
%Also separate Gauss and Cauchy parts (of K_alpha1 peak):
[~,ycauchy1,ygauss1] = pseudoVoigtAsym_C_G_separate(a_final,x_data);
%Find maximum of K_alpha1 peak:
ind_ges1 = find(pseudoVoigtAsym_separate(a_final,x_data)>=...
    max(pseudoVoigtAsym_separate(a_final,x_data)));

%Create curve for Gauss & Cauchy and K_alpha2 (NO K_alpha1!):
WAVELENGTH_SEPARATE = [intensity_Kalpha2 delta_Cu_K_alpha2/Cu_K_alpha];
yc_fit_Kalpha2 = sum(pseudoVoigtAsym_separate(a_final,x_data),1)...
    +background;
%Again separate Gauss and Cauchy parts (of K_alpha2 peak):
[y,ycauchy2,ygauss2] = pseudoVoigtAsym_C_G_separate(a_final,x_data);
%Combine Cauchy_K_alpha1 and Cauchy_K_alpha2:
ycauchy = ycauchy1 + ycauchy2;
%Combine Gauss_K_alpha1 and Gauss_K_alpha2:
ygauss = ygauss1 + ygauss2;
%Combine Kalpha1 and Kalpha2 fit to make sure they give the same result as
%the combined fit:
yc_fit_Kalpha_test = yc_fit_Kalpha1 + yc_fit_Kalpha2...
    -background;
%Check if Gauss/Cauchy separation worked correctly. If it works for Kalpha2
%peak, it is assumed that it also worked for Kalpha1:
if y ~= pseudoVoigtAsym_separate(a_final,x_data)
    errormessage=['Der Funktionierer ist kaputt!(Gauss+Lorentz-Kurve '...
        '~= Gesamtfite) \nThis usually happens if the fit function did '...
        'not find a minimum'];
    error(errormessage,[ ])
end

```

```

end

% Get 2Theta angle of overall peak maximum (K_alpha1&2 together):
x1 = x_data(ind_ges1);

% Sum Cauchy parts and background for plotting:
ycauchy_plot = sum(ycauchy,1)+background;
% Find maximum position of Cauchy K_alpha1 peak:
ind_cauchy = find(ycauchy1>=max(ycauchy1));
% Sum Gauss parts and background for plotting:
ygauss_plot = sum(ygauss,1)+background;
% Find maximum position of Gauss K_alpha1 peak:
ind_gauss = find(ygauss1>=max(ygauss1));

%---Difference plot-----
y_diff = y_data - yc_fit;

%!!!!!!!!!!!!!!IMPORTANT!!!!!!!!!!!!!!
% The plots here are done for every of of the 4 repetitions, one can
% visually analyse each peak and search for problems if something big
% goes wrong! For normal data analysis the two plots at the end should do.

% % This Plot is to compare the data with the fit, the K_alpha1 and
% % K_alpha2 peaks separately, the background function and the peak
% % maximum:

% figure
% semilogy(x_data,y_data,'k*')
% hold on
% semilogy(x_data,yc_fit,'m')
% semilogy(x_data,yc_fit_Kalpha1,'r')
% semilogy(x_data,background,'y')
% line([x1 x1], get(gca, 'ylim'),'color','c','LineStyle','-');
% semilogy(x_data,yc_fit_Kalpha2,'r')
% xlabel ('Winkel 2 \Theta [°]')
% ylabel ('Counts [a.u.]')
% legend ('Messdaten','Gefittete Kurve','K_{\alpha1} und K_{\alpha2}...
% separat gefittet','Untergrund','Maximum')
% hold off

% % The next Plot is to compare the data with the fit, the starting
% % parameters that gave the best fit results, the background function, the
% % peak maximum and to check if separation has worked properly (The fittet
% % curve should be overplottet by a curve that is the sum of all
% % previously separated K_alpha1+K_alpha2 curves).
%
% figure
% semilogy(x_data,y_data,'k*')

```

```

% hold on
% semilogy(x_data,yc_schaetz,'g')
% semilogy(x_data,yc_fit,'m')
% semilogy(x_data,yc_fit_Kalpha_test,'b')
% semilogy(x_data,background,'y')
% line([x1 x1], get(gca, 'ylim'),'color','c','LineStyle','-');
% xlabel ('Winkel 2 \Theta [°]')
% ylabel ('Counts [a.u.]')
% legend ('Messdaten','Geschaetzte Anfangsparameter','Gefittete Kurve',...
% 'K_{\alpha1} + K_{\alpha2}','Untergrund','Maximum')
% hold off

```

```

%% The next Plot is to compare the data with the fit, the background
%% function, the peak maximum, Cauchy and Gauss peaks separately and to
%% check if separation has worked properly (The fittet curve should be
%% overplottet by a curve that is the sum of the previously separated
%% Cauchy- and Gauss-curves).

```

```

% figure
% semilogy(x_data,y_data,'k*')
% hold on
% semilogy(x_data,yc_fit,'m')
% %semilogy(x_data,yc_fit_Kalpha1,'r')
% semilogy(x_data,background,'y')
% %semilogy(x_data,yc_fit_Kalpha_test,'b')
% line([x1 x1], get(gca, 'ylim'),'color','c','LineStyle','-');
% semilogy(x_data,ygauss_plot,'g')
% semilogy(x_data,ycauchy_plot,'c')
% semilogy(x_data,ygauss_plot+ycauchy,'r')
% %semilogy(x_data,yc_fit_Kalpha2,'r')
% xlabel ('Winkel 2 \Theta [°]')
% ylabel ('Counts [a.u.]')
% legend ('Messdaten','Gefittete Kurve','Untergrund','Maximum',...
% 'Gaussanteil','Cauchyanteil','Cauchy+Gaussanteil')
% hold off

```

```

%% Next is a LINEAR plot with just the data and the fit together with a
%% difference plot to evaluate the fit quality.

```

```

% figure
% pos1 = [0.1 0.25 0.8 0.68];
% subplot('Position',pos1)
% plot(x_data,y_data,'k*')
% hold on
% plot(x_data,yc_fit,'m')
% ylabel ('Counts [a.u.]')

```

```

% legend ('Messdaten', 'Gefittete Kurve')
% hold off
% set(gca, 'xtick', [])
% pos2 = [0.1 0.13 0.8 0.10];
% subplot('Position', pos2)
% plot(x_data, y_diff, 'b')
% xlabel ('Winkel 2 \Theta [°]')

%-----Calculate integral breadths-----
%Calculate height of the peak as seen in the plot (K_alpha1 and K_alpha2
%usually are not separated):
height_ges = yc_fit(ind_ges1) - background (ind_ges1);
%Calculate height of Cauchy peak:
height_cauchy = ycauchy(ind_cauchy);
%Calculate height of Gauss peak:
height_gauss = ygauss(ind_gauss);
%Calculate area under Cauchy peak:
A_cauchy = a_final(1) * a_final(4);
%Calculate area under Gauss peak:
A_gauss = a_final(1) * (1 - a_final(4));

%Calculate integral breadth of peak as seen in the plot (K_alpha1 and
%K_alpha2 usually are not separated):
IB_standard_ges = a_final(1)/height_ges;
%Calculate integral breadth of Cauchy peak:
IB_standard_cauchy = A_cauchy ./ height_cauchy;
%Calculate integral breadth of Gauss peak:
IB_standard_gauss = A_gauss ./ height_gauss;
%If Cauchy or Gauss peaks have no maximum, show a message:
if numel(IB_standard_cauchy)>1
    IB_standard_cauchy = max(IB_standard_cauchy)
    disp('IB_standard_cauchy hat kein maximum')
elseif numel(IB_standard_gauss) > 1
    IB_standard_gauss = max(IB_standard_gauss)
    disp('IB_standard_gauss hat kein maximum')
end

% Save the results before restarting the fitting procedure:
a_final_faul(faul,:) = a_final;
IB_ges_faul(faul,:) = IB_standard_ges;
IB_cauchy_faul(faul,:) = IB_standard_cauchy;
IB_gauss_faul(faul,:) = IB_standard_gauss;
b_final_faul(faul,:) = b_final;

faul = faul + 1

end

```

```

%----After all 4 repetitions are done, plot the results to compare fit----
%-----stability:-----

% Create variables to be filled in loop:
yc_fit_faul = zeros(4,numel(y_data));
y_diff_faul = zeros(4,numel(y_data));
y_data1 = [y_data;y_data;y_data;y_data];
farbe = ['r','b','m','g'];
%----Plotten-----
figure
pos3 = [0.1 0.25 0.8 0.68]; % Define position and height for upper plot
                                % [x y width height]
subplot('Position',pos3)
plot(x_data,y_data,'k*')          %Plot data
hold on
for i = 1:4
% Save fit data for all 4 repetitions:
yc_fit_faul = sum(pseudoVoigtAsym(a_final_faul(i,:),x_data),1)...
    +polyval(b_final_faul(i,:),x_data,[],mu);
% Create difference plot for all 4 repetitions:
y_diff_faul(i,:) = y_data1(i,:) - yc_fit_faul;

plot(x_data,yc_fit_faul,farbe(i))
ylabel ('Counts [a.u.]')
set(gca,'xtick',[])
end
legend ('Data','Fitted curve 1','Fitted curve 2',...
    'Fitted curve 3','Fitted curve 4')
title([int2str(peaknamen(ges)) '° Peak'])

pos4 = [0.1 0.13 0.8 0.10]; % Define position and height for lower plot
                                % [x y width height]
subplot('Position',pos4)

%Plot the difference plot:
for i = 1:4
plot(x_data,y_diff_faul(i,:),farbe(i))
hold on
end
xlabel ('Angle 2 \Theta [°]')
hold off

%Also plot data, fit, K_alpha1, K_alpha2, background and peam maximum:
figure
semilogy(x_data,y_data,'k*')
hold on
semilogy(x_data,yc_fit,'m')
semilogy(x_data,yc_fit_Kalpha1,'r')
semilogy(x_data,background,'y')
line([x1 x1], get(gca, 'ylim'),'color','c','LineStyle','-.');

```

```

semilogy(x_data,yc_fit_Kalpha2,'r')

xlabel ('Angle 2 \Theta [°]')
ylabel ('Counts [a.u.]')
legend ('Data','Fitted curve',...
        'K_{\alpha1} and K_{\alpha2} fitted separately','Background',...
        'K_{\alpha1} max.>')
hold off

% Create output variable that can be copied into a .txt file and contains
% all important information. If this is done, the mWH/mWA data processing
% routine can import needed data easily:
if ges == 1
    daten_exportieren =...
        [IB_ges_faul IB_cauchy_faul IB_gauss_faul a_final_faul];
else
daten_exportieren =...
    [daten_exportieren;IB_ges_faul IB_cauchy_faul IB_gauss_faul a_final_faul];
end
end
end

```

9.6.6 LaB₆_exp_fit

The Matlab code for creating an exponential fit for the LaB₆ integral breadths and reading out the instrumental broadening at the steel peak positions is:

```

close all
clear all
clc

%Import data
IB_LaB6 = load('LaB_6_1.txt','-ascii');

%-----Calculate peak positions-----

% Inculding the 30 ° peak:
% position_21 = mean(IB_LaB6(1:4,5));
% position_30 = mean(IB_LaB6(5:8,5));
% position_37 = mean(IB_LaB6(9:12,5));
% position_43 = mean(IB_LaB6(13:16,5));
% position_49 = mean(IB_LaB6(17:20,5));
% position_54 = mean(IB_LaB6(21:24,5));
% position_76 = mean(IB_LaB6(25:28,5));
% position_80 = mean(IB_LaB6(29:32,5));
% position_84 = mean(IB_LaB6(33:36,5));
% position_88 = mean(IB_LaB6(37:40,5));
% position_96 = mean(IB_LaB6(41:44,5));
% position_100 = mean(IB_LaB6(45:48,5));

```

```

% position_104 = mean(IB_LaB6(49:52,5));
% position_108 = mean(IB_LaB6(53:56,5));
% position_112 = mean(IB_LaB6(57:60,5));
% position_121 = mean(IB_LaB6(61:64,5));
% position_131 = mean(IB_LaB6(65:68,5));
% position_136 = mean(IB_LaB6(69:72,5));
% position_142 = mean(IB_LaB6(73:76,5));
% position = [position_21;position_30;position_37;position_43;...
% position_49;position_54;position_76;position_80;position_84;...
% position_88;position_96;position_100;position_104;position_108;...
% position_112;position_121;position_131;position_136;position_142];

% Without the 30 ° peak:
position_21 = mean(IB_LaB6(1:4,5));
position_37 = mean(IB_LaB6(5:8,5));
position_43 = mean(IB_LaB6(9:12,5));
position_49 = mean(IB_LaB6(13:16,5));
position_54 = mean(IB_LaB6(17:20,5));
position_76 = mean(IB_LaB6(21:24,5));
position_80 = mean(IB_LaB6(25:28,5));
position_84 = mean(IB_LaB6(29:32,5));
position_88 = mean(IB_LaB6(33:36,5));
position_100 = mean(IB_LaB6(37:40,5));
position_104 = mean(IB_LaB6(41:44,5));
position_108 = mean(IB_LaB6(45:48,5));
position_112 = mean(IB_LaB6(49:52,5));
position_116 = mean(IB_LaB6(53:56,5));
position_121 = mean(IB_LaB6(57:60,5));
position_131 = mean(IB_LaB6(61:64,5));
position_136 = mean(IB_LaB6(65:68,5));
position_142 = mean(IB_LaB6(69:72,5));

position = [position_21;position_37;position_43;position_49;...
position_54;position_76;position_80;position_84;position_88;...
position_100;position_104;position_108;position_112;position_116;...
position_121;position_131;position_136;position_142];

%-----Calculate IBs-----
IB_ges = zeros(numel(IB_LaB6(:,1))/4,1);
IB_cauchy = IB_ges;
IB_gauss = IB_ges;

i=1;
ii = 1;
while i < numel(IB_LaB6(:,1))
IB_ges(ii) = mean(IB_LaB6(i:i+3,1));
IB_cauchy(ii) = mean(IB_LaB6(i:i+3,2));
IB_gauss(ii) = mean(IB_LaB6(i:i+3,3));
i = i + 4;
ii = ii+1;

```

```

end

pos_rad = position / 180 * pi;
fit_ges = fit(position,IB_ges,'exp2');
fit_gauss = fit(position,IB_gauss,'exp2');
fit_cauchy = fit(position,IB_cauchy,'exp2');

%plot(fit_ges,position,IB_ges,'*')
%hold on
figure
plot(fit_cauchy,position,IB_cauchy,'*')
title('Cauchy')

figure
plot(fit_gauss,position,IB_gauss,'*')
title('Gauss')
% hold off
% legend('IB-gesamt Daten','IB-gesamt Fit','IB-Cauchy-Anteil Daten',...
% 'IB-Cauchy-Anteil Fit','IB-Gauss-Anteil Daten','IB-Gauss-Anteil Fit')

%-----Read data from LaB_6 fit at steel peak positions-----
% With 45 ° peak:
% P91 = load('P91_1.txt','-ascii');
% pos_steel_45 = mean(P91(1:4,5));
% pos_steel_65 = mean(P91(5:8,5));
% pos_steel_82 = mean(P91(9:12,5));
% pos_steel_99 = mean(P91(13:16,5));
% pos_steel_116 = mean(P91(17:20,5));
% pos_steel_137 = mean(P91(21:24,5));
% pos_steel = [pos_steel_45 pos_steel_65 pos_steel_82 pos_steel_99...
% pos_steel_116 pos_steel_137];

% Without 45 ° peak:
P91 = load('P91_1 - Kopie.txt','-ascii');
pos_steel_65 = mean(P91(1:4,5));
pos_steel_82 = mean(P91(5:8,5));
pos_steel_99 = mean(P91(9:12,5));
pos_steel_116 = mean(P91(13:16,5));
pos_steel_137 = mean(P91(17:20,5));
pos_steel = [pos_steel_65 pos_steel_82 pos_steel_99 pos_steel_116...
pos_steel_137];

LaB6_cauchy_export = zeros(1,numel(pos_steel));
LaB6_gauss_export = zeros(1,numel(pos_steel));

for export_lauf = 1:numel(pos_steel)
    LaB6_cauchy_export(export_lauf) = fit_cauchy(pos_steel(export_lauf));
    LaB6_gauss_export(export_lauf) = fit_gauss(pos_steel(export_lauf));
end
g=1;

```

```
save('LaB6_export', 'LaB6_cauchy_export', 'LaB6_gauss_export', '-ascii')
```

9.6.7 mWH_mWA

The file that finally calculates the dislocation densities via modified Williamson-Hall method combined with modified Warren-Averbach/contrast-corrected double-Voigt method is:

```
close all
clear all
clc
%-----Import Data-----

% %-----Data import WITH 45° Peak-----
%
% % Load the data file for the (modified) Warren-Averbach analysis:
% IB_P91 = load('BM_AR_vib_pol_parallel_beam.txt', '-ascii');
% LaB6_import = load('LaB6_parallel_export', '-ascii');
%
% % Define integral breadths (IBs) for cauchy parts and extract the
% % instrumental broadening.
% IB_C_45 = mean(IB_P91(1:4,2)) - LaB6_import(1,1);
% IB_C_65 = mean(IB_P91(5:8,2)) - LaB6_import(1,2);
% IB_C_82 = mean(IB_P91(9:12,2)) - LaB6_import(1,3);
% IB_C_99 = mean(IB_P91(13:16,2)) - LaB6_import(1,4);
% IB_C_116 = mean(IB_P91(17:20,2)) - LaB6_import(1,5);
% IB_C_137 = mean(IB_P91(21:24,2)) - LaB6_import(1,6);
% IB_C = [IB_C_45 IB_C_65 IB_C_82 IB_C_99 IB_C_116 IB_C_137];
%
% % Convert real space IBs to IBs in reciprocal space (IB(2\Theta) to
% % IB(g)):
% IB_C1 = [IB_C(1)*2*cos(mean(IB_P91(1:4,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_C(2)*2*cos(mean(IB_P91(5:8,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_C(3)*2*cos(mean(IB_P91(9:12,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_C(4)*2*cos(mean(IB_P91(13:16,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_C(5)*2*cos(mean(IB_P91(17:20,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_C(6)*2*cos(mean(IB_P91(21:24,5))/2*pi/180)/(0.1540598*10^-9)];
%
% % Define integral breadths (IBs) for gaussian parts and extract the
% % instrumental broadening.
% IB_G_45 = sqrt(mean(IB_P91(1:4,3)).^2 - LaB6_import(2,1).^2);
% IB_G_65 = sqrt(mean(IB_P91(5:8,3)).^2 - LaB6_import(2,2).^2);
% IB_G_82 = sqrt(mean(IB_P91(9:12,3)).^2 - LaB6_import(2,3).^2);
% IB_G_99 = sqrt(mean(IB_P91(13:16,3)).^2 - LaB6_import(2,4).^2);
% IB_G_116 = sqrt(mean(IB_P91(17:20,3)).^2 - LaB6_import(2,5).^2);
% IB_G_137 = sqrt(mean(IB_P91(21:24,3)).^2 - LaB6_import(2,6).^2);
% IB_G = [IB_G_45 IB_G_65 IB_G_82 IB_G_99 IB_G_116 IB_G_137];
%
% % Convert real space IBs to IBs in reciprocal space (IB(2\Theta) to
% % IB(g)):
```

```

% IB_G1 = [IB_G(1)*2*cos(mean(IB_P91(1:4,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_G(2)*2*cos(mean(IB_P91(5:8,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_G(3)*2*cos(mean(IB_P91(9:12,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_G(4)*2*cos(mean(IB_P91(13:16,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_G(5)*2*cos(mean(IB_P91(17:20,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_G(6)*2*cos(mean(IB_P91(21:24,5))/2*pi/180)/(0.1540598*10^-9)];
%
%
% if numel(IB_C) ~= numel(IB_G)
%     error('There must be the same amount of Lorentz- and Gauss-peaks!')
% end
%
% IB_ges = IB_C + IB_G;
%
% % Define variables:
% deltag = ...
% [IB_ges(1)*2*cos(mean(IB_P91(1:4,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_ges(2)*2*cos(mean(IB_P91(5:8,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_ges(3)*2*cos(mean(IB_P91(9:12,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_ges(4)*2*cos(mean(IB_P91(13:16,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_ges(5)*2*cos(mean(IB_P91(17:20,5))/2*pi/180)/(0.1540598*10^-9)...
% IB_ges(6)*2*cos(mean(IB_P91(21:24,5))/2*pi/180)/(0.1540598*10^-9)];
%
% g = [2*sin(mean(IB_P91(1:4,5))/2*pi/180)/(0.1540598*10^-9)...
% 2*sin(mean(IB_P91(5:8,5))/2*pi/180)/(0.1540598*10^-9)...
% 2*sin(mean(IB_P91(9:12,5))/2*pi/180)/(0.1540598*10^-9)...
% 2*sin(mean(IB_P91(13:16,5))/2*pi/180)/(0.1540598*10^-9)...
% 2*sin(mean(IB_P91(17:20,5))/2*pi/180)/(0.1540598*10^-9)...
% 2*sin(mean(IB_P91(21:24,5))/2*pi/180)/(0.1540598*10^-9)];
%
% C_h00 = 0.285;          % Here, C_h00 of pure iron is used

% % -----End of data import WITH 45° Peak-----

%-----Data import WITHOUT 45° Peak-----
% When using the physics department's PANalytical x-ray-machine with
% monochromator, the 45° peak is usually corrupted. Use this import
% option then.

% Load the data file for the (modified) Warren-Averbach analysis:
%IB_P91 = load('P91_1p5h_760C.txt','-ascii');
IB_P91 = load('P91_36h_760C.txt','-ascii');
%IB_P91 = load('P91_1.txt','-ascii');
LaB6_import = load('LaB6_export','-ascii');
% Define inequal breadths (IBs) for cauchy parts and extract the
% instrumental broadening.
IB_C_65 = mean(IB_P91(1:4,2)) - LaB6_import(1,1);
IB_C_82 = mean(IB_P91(5:8,2)) - LaB6_import(1,2);
IB_C_99 = mean(IB_P91(9:12,2)) - LaB6_import(1,3);

```

```

IB_C_116 = mean(IB_P91(13:16,2)) - LaB6_import(1,4);
IB_C_137 = mean(IB_P91(17:20,2)) - LaB6_import(1,5);
IB_C = [IB_C_65 IB_C_82 IB_C_99 IB_C_116 IB_C_137]*pi/180;

% Convert real space IBs to IBs in reciprocal space (IB(2\Theta) to
% IB(g)):
IB_C1 = [IB_C(1)*2*cos(mean(IB_P91(1:4,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_C(2)*2*cos(mean(IB_P91(5:8,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_C(3)*2*cos(mean(IB_P91(9:12,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_C(4)*2*cos(mean(IB_P91(13:16,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_C(5)*2*cos(mean(IB_P91(17:20,5))/2*pi/180)/(0.1540598*10^-9)];

% Define integral breadths (IBs) for gaussian parts and extract the
% instrumental broadening.
IB_G_65 = sqrt(mean(IB_P91(1:4,3)).^2 - LaB6_import(2,1).^2);
IB_G_82 = sqrt(mean(IB_P91(5:8,3)).^2 - LaB6_import(2,2).^2);
IB_G_99 = sqrt(mean(IB_P91(9:12,3)).^2 - LaB6_import(2,3).^2);
IB_G_116 = sqrt(mean(IB_P91(13:16,3)).^2 - LaB6_import(2,4).^2);
IB_G_137 = sqrt(mean(IB_P91(17:20,3)).^2 - LaB6_import(2,5).^2);
IB_G = [IB_G_65 IB_G_82 IB_G_99 IB_G_116 IB_G_137]*pi/180;

% Convert real space IBs to IBs in reciprocal space (IB(2\Theta) to
% IB(g)):
IB_G1 = [IB_G(1)*2*cos(mean(IB_P91(1:4,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_G(2)*2*cos(mean(IB_P91(5:8,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_G(3)*2*cos(mean(IB_P91(9:12,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_G(4)*2*cos(mean(IB_P91(13:16,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_G(5)*2*cos(mean(IB_P91(17:20,5))/2*pi/180)/(0.1540598*10^-9)];

if numel(IB_C) ~= numel(IB_G)
    error('There must be the same amount of Lorentz- and Gauss-peaks!')
end
IB_ges = IB_C + IB_G;

% Define variables:
deltag =...
  [IB_ges(1)*2*cos(mean(IB_P91(1:4,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_ges(2)*2*cos(mean(IB_P91(5:8,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_ges(3)*2*cos(mean(IB_P91(9:12,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_ges(4)*2*cos(mean(IB_P91(13:16,5))/2*pi/180)/(0.1540598*10^-9)...
  IB_ges(5)*2*cos(mean(IB_P91(17:20,5))/2*pi/180)/(0.1540598*10^-9)];

g = [2*sin(mean(IB_P91(1:4,5))/2*pi/180)/(0.1540598*10^-9)...
  2*sin(mean(IB_P91(5:8,5))/2*pi/180)/(0.1540598*10^-9)...
  2*sin(mean(IB_P91(9:12,5))/2*pi/180)/(0.1540598*10^-9)...
  2*sin(mean(IB_P91(13:16,5))/2*pi/180)/(0.1540598*10^-9)...
  2*sin(mean(IB_P91(17:20,5))/2*pi/180)/(0.1540598*10^-9)];

C_h00 = 0.285;          % Here, C_h00 of pure iron is used

```

```

% -----End of data import WITHOUT 45° Peak-----

deltag_cauchy = deltag;
g_cauchy = g;

%-----End of data import-----

% Define H2=(h^2*k^2 + h^2*l^2 + k^2*l^2) / (h^2+k^2+l^2)^2
% with the Miller's indices (hkl) of the analysed peaks:
H2_110 = (1^2*1^2 + 1^2*0^2 + 1^2*0^2) / (1^2+1^2+0^2)^2;
H2_200 = (2^2*0^2 + 2^2*0^2 + 0^2*0^2) / (2^2+0^2+0^2)^2;
H2_211 = (2^2*1^2 + 2^2*1^2 + 1^2*1^2) / (2^2+1^2+1^2)^2;
H2_220 = (2^2*2^2 + 2^2*0^2 + 2^2*0^2) / (2^2+2^2+0^2)^2;
H2_310 = (3^2*1^2 + 3^2*0^2 + 1^2*0^2) / (3^2+1^2+0^2)^2;
H2_222 = (2^2*2^2 + 2^2*2^2 + 2^2*2^2) / (2^2+2^2+2^2)^2;

% If the Data import WITH 45° peak is activated, use the following H2:
%H2 = [H2_110 H2_200 H2_211 H2_220 H2_310 H2_222];

% If the Data import WITHOUT 45° peak is activated, use the following H2:
H2 = [H2_200 H2_211 H2_220 H2_310 H2_222];

% % Use the modified Williamson-Hall method to determine the reflection
% % constant of dislocations C: C = C_h00 .* (1 - q .* H2);

%-----Guess alpha = (0.9/D)^2 and calculate q-----
%-----with the crystallite size D-----
% Set a random alpha and only save it, if the fit results are better
% than the previously best one:
goffinal = 1000;
for i = 1:500
    alpha = rand*200;

% Define parameters for (deltag^2-alpha)./(g^2) vs. H^2 plot:
y = (deltag_cauchy.^2-alpha)./(g_cauchy.^2);
x = H2;

% Make a linear fit and evaluate it's quality:
[fitq, gof] = fit(x.',y.', 'poly1');
if gof.sse<goffinal
    fitq_final = fitq;
    goffinal = gof.sse;
    y_final = y;
    alphafinal = alpha;
end
end

%-----Find Root of the best fit and calculate q-----
fitq_final_vals = coeffvalues(fitq_final);
qq = roots(fitq_final_vals);

```

```

q = 1 / qq;

%-----Plot-----
fitq_final_plot = fitq_final(0:0.001:0.5);
figure
plot (x,y_final,'o')
hold on
plot(0:0.001:0.5,fitq_final_plot)
hold off
title(['Mit \alpha = ' num2str(alpha)])
xlabel('H^2')
ylabel('$\displaystyle\frac{(\Delta g)^2 - \alpha}{g^2}$',...
'interpreter','latex')

% %-----modified Warren-Averbach-----
% % If one doesn't want to use the modified Williamson-Hall method to
% % determine q, the following for-loop can be used instead. To do so,
% % scroll down ~100 lines and also uncomment the marked 5 lines there! The
% % loop will create a modified Warren-Averbach plot for each q and save
% % the estimated standard deviation. The last iteration is again done with
% % the "best q" (=lowest estimated standard deviation) and the program
% % continues with this value.
% delta_A0 = 10000;
% iter_max = 18; %By default, q starts at zero und increases with every
% % iteration by 0.1 . For steel q~1.2 (pure edge
% % dislocations) to q~2.8 (pure screw dislocations) is
% % realistic
%
% for ii=0:iter_max
% q = 1 + 0.1*ii;
% if ii == iter_max
% % close all %Uncomment if one does NOT want to controll the mWA-plots
% q = q_final;
% end

% If you already know your q (and do NOT use the for loop to evaluate it),
% please insert it here. Otherwise comment it out.
%q = 0;

% % If one wants to know gamma to determine the outer cutoff-radius and did
% % use the modified Williamson-Hall technique to determine q (see
% % PDF-document) please uncomment. Otherwise leave it commented.
%gamma = (deltag_cauchy(2)^2-alphafinal)/(g_cauchy(2)^2*C_h00);

%Calculate the contrast factor C:
if q == 0

```

```

    C=1;
else
    C = C_h00 .* (1 - q .* H2);
end

%-----Warren-Averbach ueber Double-Voigt-----

% Often used n ranges: (max. length of n is numel(n)=12 )
%n = [1 3 5 7 9 11 13 15];
%n = [1 2 3 5 10 15 20 25 30 35 40];
n = [1 2 3 10 20 30 40 50 60 80 100 150];

% The calculation of a3 is commented out, because n can be chosen so that
% L is a natural number. This would come to the same result we do, but
% would take some calculation time.
%a3(1) = (0.1540598*10^-9) / (2 * (sin(70/2*pi/180) - sin(60/2*pi/180)));
%a3(1) = (0.1540598*10^-9) / (2 * (sin(87/2*pi/180) - sin(77/2*pi/180)));
%a3(1) = (0.1540598*10^-9) / (2 * (sin(104/2*pi/180) - sin(94/2*pi/180)));
%a3(1) = (0.1540598*10^-9) / (2 * (sin(121/2*pi/180) - sin(111/2*pi/180)));
%a3(1) = (0.1540598*10^-9) / (2 * (sin(142/2*pi/180) - sin(132/2*pi/180)));
%n = n ./ a3 * 10^(-9);

%-----Calculate ln(A(L)) for Warren-Averbach-plot-----
% Define variables to be filled in loop:
A_L = zeros(numel(n),numel(IB_C));
ln_A_L = A_L;
L = zeros (size(n));

%Loop to calculate ln(A(L)):
for n_lauf = 1 : numel(n)
    %L(n_lauf) = n(n_lauf) * a3;    %Is used if a3 is calculated
    L(n_lauf) = n(n_lauf) * 10^(-9);
    for IB_lauf = 1 : numel(IB_C)
        ln_A_L(n_lauf, IB_lauf) = -2 * L(n_lauf) * IB_C1(IB_lauf) -...
            pi * L(n_lauf)^2 * IB_G1(IB_lauf)^2;
    end
end

% Calculate g^2*C for modified Warren-Averbach plot
ss = g(1:end).^2 .*C(1:end);
%ss = ss([1 2 4 5]);
%ln_A_L = ln_A_L(:, [1 2 4 5]);

%Define colors and shapes for modified Warren-Averbach plot:
farben = [[1,0,0];[0,1,0];[0,0,1];[1,1,0];[0,1,1];[0.5,0.8,0.5];...
    [0.3 1 0.8];[0.5 1 0.3];[0.3 0.5 0.8];[0.5 0.5 0.3];[0.5 0.8 0.8];...
    [0.8 0.8 0.5]];
formen = ['+', 'o', '*', 'x', 's', 'd', '^', 'v', '>', '<', 'p', 'h'];

%-----Calculate and plot fits for modified Warren-Averbach method-----

```

```

% Define variables to be filled in loop:
Y_val = zeros(numel(n),2); %Y_val(:,1)=p1;Y_val(:,2)=p2 for y=p1x+p2 fit.

% Begin plot:
figure
hold on
plots = size(n); % Save plots to variable to insert correct legend later

for n_lauf = 1 : numel(n)
    % Perform fit with moving zero-point to ensure convergence:
    [A_fit,S_fit,mu_fit] = polyfit(ss.', ln_A_L(n_lauf,1:end).',1);
    % Calculate y-values for this fit over wide range to evaluate slope
    % later:
    Y_val(n_lauf,:) =...
        polyval(A_fit,10^19,S_fit,mu_fit) - polyval(A_fit,0,S_fit,mu_fit);
    % Calculate y-values for this fit for plotting:
    [A_fit_plot,delta_A] =...
        polyval(A_fit,linspace(0,max(ss)+max(ss)*0.05,10),S_fit,mu_fit);
    % Plot data and corresponding fits:
    plots(n_lauf) = plot(ss,ln_A_L(n_lauf,:),formen(n_lauf), 'color',...
        farben(n_lauf,:), 'DisplayName', ['L = ' int2str(L(n_lauf)*10^9) ' nm']);
    plot(linspace(0,max(ss)+max(ss)*0.05,10),A_fit_plot,...
        'color',farben(n_lauf,:));
end
% Insert plot title, labels and legend:
if q==0
    title('Normal WA-Plot')
    xlabel('g^2 [m^{-2}]')
else
    title(['mWA-Plot with q = ' num2str(q)])
    xlabel('g^2 * C [m^{-2}]')
end
ylabel('ln(A)')
legend(plots)
hold off

%% !!!!As mentioned at around line 280/290: !!!!
%% If one does not want to calculate q from the modified Williamson-Hall
%% technique, one can also fit the data for various q values and evaluate
%% the fit quality both numerically (best q = q_final) and graphically
%% (plots are done for all q-values). To do so, uncomment the 5 lines
%% below.
% if delta_A0 > delta_A
%     q_final = q;
% end
% delta_A0 = delta_A;
% end

```

```

% Continue with modified Warren-Averbach analysis:

Y = -Y_val(:,1)/10^19; % Get slope of mWA-plot for Y/L^2 vs. ln(L)-plot
YY = Y.' ./ (L(1:end)).^2; % = Y/L^2
durchL = 1./L; %Create variable for cauchy-/gauss-separation
Y_durchL_fit = fit (durchL.',YY.', 'poly1'); %Create Y/L^2 vs. 1/L fit
% Cauchy-/gauss-separation: Plot Y/L^2 vs. 1/L
% --> beta_gauss_distortion = Y/L^2 at 1/L = 0;
% --> beta_cauchy_distortion = slope of this plot

% Plot data and fit:
figure
plot(1./L,YY,'+');
hold on
plot(Y_durchL_fit);
hold off
xlabel('1/L [nm^-1]')
xlabel('\displaystyle\frac{1}{L} [nm^{-1}]','interpreter','latex')
ylabel('Y_{DV}/L^2')
ylabel('\displaystyle\frac{Y_{DV}}{L^2}','interpreter','latex')

% Continue calculation with cauchy-/gauss-separated data:
Y_bereinigt = coeffvalues(Y_durchL_fit);
propto_beta_GD2 = Y_bereinigt(2); % = beta_gauss_distortion^2 * pi * 1/g^2
propto_beta_CD = Y_bereinigt(1); % = beta_cauchy_distortion * 2 * 1/g^2

e_G_ss = propto_beta_GD2./(2*pi^2); % calculate mean-squared-strain
% from gauss only
e_C_ss = propto_beta_CD ./ (2*pi^2); % calculate mean-squared-strain
% from cauchy only
e_mss = e_G_ss+e_C_ss; % calculate combined mean-squared-strain

% Calculate the dislocation density according to this kind-of double-Voigt-
% method with the Williamson-Smallman-equation:
rho_double_voigt = (14.4 .* e_mss) ./ ((0.248.*10.^(-9)).^2)

% Plot data according to kind-of double-Voigt-method:
L_plot = linspace(L(1),L(end),1000);
YY_new = propto_beta_GD2 + Y_bereinigt(1)./L_plot;
% Fit data according to the conventional modified Warren-Averbach method:
Y_fit1 = fit(log(L([1 2 3])).', YY([1 2 3])).', 'poly1');

% Use conventional modified Warren-Averbach method to determine
% dislocation density:
k_val = coeffvalues(Y_fit1); % Get values from fit

rho1 = -(k_val(1)*2)/(pi*(0.248*10.^(-9))^2) % Calculate disloc. density

% Plot data, modified double-Voigt fit and linear fit according to mWA:
figure

```

```
plot(log(L(1:end)),YY,'+')
hold on
plot(log(L_plot),YY_new)
plot(Y_fit1,'k')
hold off
xlabel ('ln(L)')
ylabel('$\displaystyle\frac{Y}{L^2}$','interpreter','latex')
legend('Data','New introduced method',...
       'Lin. fit (=modified Warren-Averbach)')
title(['Disloc. density \rho = ' num2str(rho_double_voigt,1)...
       ' with q = ' num2str(q,3)])
```