



HAL
open science

DxTools: Processing large data files recorded with the Bruker D8 diffractometer

Alexandre Boulle

► **To cite this version:**

Alexandre Boulle. DxTools: Processing large data files recorded with the Bruker D8 diffractometer. *Journal of Applied Crystallography*, 2017, 50 (3), pp.967-974. 10.1107/S1600576717005192. hal-02193112

HAL Id: hal-02193112

<https://hal.science/hal-02193112>

Submitted on 24 Jul 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

DxTools: Processing large data files recorded with the Bruker D8 diffractometer

A. Boulle

Science des Procédés Céramiques et Traitements de Surfaces, CNRS UMR 7315, Centre Européen de la Céramique, 12 rue Atlantis, 87068 Limoges Cedex, France.

Abstract

Data processing is the daily reality for crystallographers who often seek to automate this repetitive task. With modern commercial diffractometers allowing for more elaborate experiments to be performed, the use of commercial software is almost unavoidable to process the data which is often provided in a manufacturer-specific format. This is especially the case when the intensity is recorded as a function of more than one single scanning motor, and/or when recording data with a one-dimensional detector. A simple, free and open-source alternative is here presented to process that type of diffraction data recorded with the Bruker D8 diffractometer. It allows handling of data files (in uxd and brml format) corresponding to reciprocal space maps [in the (Q_x, Q_z) and (Q_x, Q_y) planes], temperature-dependent scans, spatial scans, $\sin^2\psi$ measurements and pole figures. The program is easy to use and can be extended to any file format. It can be downloaded from <https://aboulle.github.io/DxTools/>

1. Introduction

The two last decades witnessed huge improvements in the performance of commercial high-resolution X-ray diffractometers devoted to the analysis of thin film systems and epitaxial heterostructures. The main innovations that have led to this situation are (i) the development of multilayer mirrors (Schuster & Göbel, 1995) yielding an increase in incident X-ray beam intensity

of about one order of magnitude and, (ii) the development of one-dimensional and two-dimensional solid-state detectors (*e.g.* the PANalytical X'Celerator, PIXcel and PIXcel^{3D} detectors or the Bruker LYNXEYE, HI-STAR and VÅNTEC detectors) that further reduced the acquisition time by several orders of magnitude. Such improved capabilities opened the way to studies that were previously restricted to synchrotron facilities, like for instance in-situ experiments or the study of ultra-thin films while keeping a monochromatic and well collimated beam.

An immediate side effect is that the amount of data that the users have to deal with has scaled by the same amount. Obviously, all manufacturers provide dedicated software programs to read and handle the data generated with their equipment. However, this often comes with a relatively high price. Moreover, for those users curious to know how their data is manipulated within these programs this can be quite disappointing since such software programs are in general closed-source. Fortunately, there is always the possibility to export the data in ASCII format. However, the handling of these data files is often not user friendly, especially in the case of two-dimensional data, *i.e.* when two (or more) motors are scanned, which is the case in most experiments (reciprocal space mapping, spatial mapping, temperature experiments, pole figure measurements, $\sin^2\psi$ measurements, etc.). Crystallographers around the world are hence often led to develop *ad hoc* scripts or short programs to analyze the data from a particular piece of equipment and for a particular type of measurement.

In the present article we present a collection of scripts, gathered within a simple graphical user interface, that allows for the display and processing of two-dimensional data recorded with the Bruker D8 diffractometer. The program is written in Python and runs both in Windows and Linux environments. The program can be freely downloaded from <https://abouille.github.io/DxTools/>. As will be shown below, this program is by no means restricted to the D8 diffractometer, and it can be extended to any data formats. This task should be facilitated by the fact that the code is open source and it can therefore be copied and modified by anyone with some basic programming knowledge (within the terms the CeCILL license, a French license similar to the General Public License).

DxTools has been developed to fulfill the needs encountered in our laboratory. Therefore, although it might be useful to other users (which justifies its public distribution), it certainly does not cover all possible types of experiments that can be carried out with the D8 diffractometer. For example, we here do not consider the case of data recorded with two-dimensional detectors.

The DxTools program is described below. In section 2, the data file format and the geometry of the different experiments supported by DxTools are described, and the underlying theoretical concepts are briefly recalled. In section 3, we give a detailed description the different data treatments and data analysis options that are available in the program. Supportive examples are also given in this section.

2. Program description

2.1 General principle

DxTools (which stands for Data eXtraction Tools) has been developed with three leading ideas in mind. First of all, the program must be easily extensible: the support for new data formats and new type of experiments should easily be added. This led to a two-step procedure where the reading of the input data is independent of the data processing, Fig. 1 (section 2.3 and 2.4). Secondly, the data should not be altered by the processing: all output files must contain the raw intensity values reshaped in a format suitable for plotting or simulation programs (section 3). Finally, the program should be easy to use and modify. The program has been developed using the Python 3.5 programming language (www.python.org) using only a minimal amount of external libraries, here the SciPy library (www.scipy.org) to benefit from array programming capabilities and advanced scientific functions, such as interpolation or least-squares fitting (van de Walt, 2011; Oliphant, 2007), and the Matplotlib library (www.matplotlib.org) for data plotting. Note that these libraries are part of the so-called SciPy stack and are available in a number of scientific Python distributions, such as Anaconda (<https://www.continuum.io/>), Canopy (<https://www.enthought.com/>) or

Python(x,y) (<https://python-xy.github.io/>), just to cite a few. These libraries can of course be installed independently from any distribution. The graphical user interface is built using the tkinter library which is included in Python.

2.2 Diffraction geometry and motor designation

The diffraction geometry is schematically represented in Fig. 2, where \mathbf{K}_0 and \mathbf{K}_h are the incident and diffracted wave vectors, and \mathbf{Q} is scattering vector with $Q = 4\pi \sin \theta / \lambda$ (where λ is the X-ray wavelength). The goniometer of the D8 diffractometer is based on a standard Eulerian cradle where the Euler angles ω , χ and φ , correspond, respectively, to the incidence angle of the X-ray beam on the surface, the tilt of the surface normal out of the plane defined by the incident and diffracted beams (the scattering plane, in gray in Fig. 2), and the rotation of the crystal around its surface normal¹. When these three angles are set to zero, the X, Y and Z axes, which constitute the crystal reference frame, correspond to the χ , ω and φ rotation axes, respectively. Notice that it is common to use ω instead of θ since the latter correspond to half the scattering angle, 2θ , and is therefore not necessarily equal to ω . The quantity $\omega - \theta$ corresponds to the deviation of the scattering vector from the specular (*i.e.* symmetrical) geometry and is referred to as the offset. DxTools supports experiments carried out in the coplanar geometry ($\chi=0$) and in the skew geometry ($\chi \neq 0$). In these geometries, for a given reflection (*i.e.* a given 2θ) the scattering vector is entirely defined with the three ω , χ and φ angles. For the moment, DxTools does not support the in-plane diffraction geometry ($\chi \approx 90^\circ$) for which four angles are required to define \mathbf{Q} : the incidence and emergence angle with respect to the surface and with respect to the diffracting planes, *i.e.* the incidence and emergence angle with respect to the surface are not necessarily equal (Pietsch, Holý & Baumbach, 2004; Schmidbauer, 2004).

¹ Strictly speaking, instead of referring to the “crystal surface”, we should here refer to the “crystallographic planes parallel to the surface”. Indeed, in practice, most crystals exhibit a small surface miscut so that the corresponding crystallographic planes are never exactly parallel to the surface. For conciseness we use “crystal surface” but the difference should be borne in mind.

It should also be noticed that, depending on the diffractometer's generation and the corresponding software, the tilt angle χ can also be referred to as ψ (which is the convention used in stress analysis [Noyan & Cohen, 1987]). In addition to the ϕ rotation, two orthogonal translations, noted t_x and t_y , allow the sample to be scanned horizontally. In addition to these motors, a t_z translation and, optionally, a compact double tilt stage allow adjustments to the height of the sample and corrections to sample misalignments (due to a miscut or to a bad sample positioning on the goniometer, for example), respectively. These last three motors are in general not used for scanning and won't be discussed any further below.

2.3 Input data format (data_reader.py)

Historically, Bruker was storing the data in binary *.raw files, which could be converted to a text format called *.uxd. With the last generation of diffractometers, the default file format for the data is now *.brml, which will be further discussed below. For the moment, DxTools provides support for the uxd and brml data formats. The algorithm to import the data is implemented in the data_reader.py file (Fig. 1). It is briefly presented below.

The uxd data format is a plain text format which is probably encoded in latin-1 (ISO/IEC 8859-1), (or a derivative of latin-1 like latin-9 [ISO/IEC 8859-15] or Windows-1252), although it is not possible to determine exactly the encoding from the file itself. Latin-1 (and its derivatives) differs from plain ASCII by using 8 bits instead of 7 thereby allowing support for special European characters. Nowadays, to avoid compatibility issues, the best practice is to use the Universal Character Set Transformation Format – 8bits (utf-8). This is what is done in this program. The uxd file is read line by line and all motor positions, or sensor values such as the temperature, as well as the intensity values are stored in variables which are finally written in a temporary text file over multiple columns (in a similar fashion to the SPEC data format, popular at synchrotron facilities). The ordering of the columns is the following: temperature, χ , ϕ , t_x , t_y , ω , offset, 2θ , primary

scanning motor and intensity. The values of temperature, χ , φ , t_x , t_y , ω , offset and 2θ either correspond to static motors, or to the starting value of scanning motors. The primary scanning motor is the fast scanning motor, for instance, the 2θ motor in the case of a series of 2θ scans recorded at, say, different χ values. In this example, χ is the secondary scanning motor. There is a redundancy in the file between the primary scanning motor and the starting position of this same motor: this allows for discrimination between the primary and the secondary scanning motor without requiring input from the user.

The brml file format is actually a *.zip archive containing a series of *.xml files, one for each secondary scanning motor position. The Python programming language natively handles zip archives and the xml file format. DxTools uncompresses the zip archive to the hard drive (in a temporary file called “unzip”) and parses the xml files to extract, as above, the motor positions and intensity values. The data is stored in a temporary file with the same structure as for the uxd files.

2.4 Supported experiments (data_processor.py)

DxTools is compatible with several common experiments carried out with point- and one-dimensional detectors. The data processing scripts are implemented in the data_processor.py file.

Reciprocal space mapping:

Reciprocal space maps (RSMs) correspond to the two-dimensional intensity distribution recorded in a particular plane of the reciprocal space². In the (Q_x, Q_z) plane (either in the coplanar or the skew geometry) RSMs are usually obtained by varying the orientation and length of the scattering vector, with the ω and 2θ angles, respectively, so as to scan two-dimensional area (Fig. 3a). This is usually achieved by recording a series of ω scans for various 2θ values, or a series of $2\theta/\theta$ scans for various offset values (Fig. 3b). The series of scans can be recorded simultaneously by using a one-

2 With the advent of two-dimensional detectors, three-dimensional RSMs can also be recorded in short amount of times. However the data processing and visualization are significantly more complicated than for conventional (two-dimensional) maps and are therefore not implemented in the present program.

dimensional detector, which reduces the counting time by several orders of magnitude. The (Q_x, Q_z) coordinates can be deduced from the ω and 2θ angles (Fig. 3a, see also Pietsch, Holý, & Baumbach, 2004):

$$\begin{aligned} Q_x &= 4\pi \sin(\theta) \sin(\omega - \theta) / \lambda \\ Q_z &= 4\pi \sin(\theta) \cos(\omega - \theta) / \lambda \end{aligned} \quad (1)$$

In the skew geometry, $\chi \neq 0$, the Q_z coordinate given by Eq. 1 actually corresponds to the vertical coordinate in the scattering plane, Q_z' (Fig. 2). From the previous equation it is straightforward to notice that the grid sampled in reciprocal space is neither orthogonal, nor does it exhibit a constant step size. For display purposes, the data has to be interpolated on a square and regular grid (Fig. 3b). The interpolation is carried-out with the 2D interpolation function of the SciPy library. This is the only case where the intensity values are modified. In all other experiments, the intensity values are not altered.

DxTools also supports RSMs recorded in the (Q_x, Q_y) plane where a series of ω scans is carried out for different azimuthal ϕ values (Conchon et al., 2007). The (Q_x, Q_y) coordinates can be deduced from the ω , 2θ and ϕ angles according to:

$$\begin{aligned} Q_x &= 4\pi \sin(\theta) \sin(\omega - \theta) \cos(\phi) / \lambda \\ Q_y &= 4\pi \sin(\theta) \sin(\omega - \theta) \sin(\phi) / \lambda \end{aligned} \quad (2)$$

Temperature-dependent scans:

The Bruker D8 diffractometer can easily be equipped with a furnace or a low-temperature chamber to perform XRD measurements at non-ambient temperatures. DxTools is compatible with any type of one-dimensional scan recorded for different temperatures.

Spatial scans:

In a similar fashion to temperature-dependent scans, one-dimensional scans can be recorded for various positions of the X-ray beam on the sample surface. This type of experiment is relevant, for

instance, in the case of inhomogeneous samples. The position of the sample is controlled *via* the t_x and t_y translations. For the moment, only experiments with one single varying translation (either t_x or t_y) are supported. The reason for this is that, because of the large beam size of the D8 diffractometer, especially when using a linear focus, it is in general irrelevant to scan the sample in the direction parallel to the long dimension of the beam.

Sin²ψ measurements:

In the framework of the determination of residual stresses from XRD data, the $\sin^2\psi$ technique takes its name from the relationship linking the planar spacing to the tilt angle ψ (which, as explained above, is equivalent to χ). For example, for the bi-axial state of strain frequently encountered in thin films (Noyan & Cohen, 1987):

$$\frac{d_{\phi\psi} - d_0}{d_0} = \frac{S_2}{2} \sigma_\phi \sin^2(\psi) + S_1(\sigma_1 + \sigma_2)$$

where d_0 is the stress-free planar spacing, σ_1 and σ_2 are the in-plane stresses at $\phi=0^\circ$ and 90° , S_1 and S_2 are the so-called X-ray elastic constants (Noyan & Cohen, 1987) which, in the case of elastic isotropy for instance, are given by $-\nu/E$ and $2(1+\nu)/E$ (ν and E being the Poisson's ratio and Young's modulus) and σ_ϕ is the in-plane stress at angle ϕ , that is $\sigma_\phi = \sigma_1 \cos^2(\phi) + \sigma_2 \sin^2(\phi)$. The planar spacing $d_{\phi\psi}$ corresponding to a particular (ϕ, ψ) pair can be obtained by recording a series of $2\theta/\theta$ scans or 2θ scans at different ϕ and ψ values (Fig. 4). The in-plane residual stress can then be obtained from the slope of the plot of $d_{\phi\psi}$ [or $(d_{\phi\psi} - d_0)/d_0$] vs. $\sin^2\psi$, for at least two ϕ values.

$\sin^2\psi$ measurements are preferably carried out with a point focus X-ray source. With a linear focus, (i) for large ψ values, the footprint of the beam on the sample plane largely exceeds the sample size, which results in a loss of intensity, and (ii) because of the poor collimation of the beam in the direction perpendicular to the scattering plane, the divergence of the beam significantly increases for increasing ψ values. These issues can be partly circumvented by scanning the offset angle, $\omega - \theta$, instead of ψ . In this geometry (see Fig. 1), the scattering vector is tilted away from the sample

surface by varying the incidence angle ω . However, for small incidence angles, the beam's footprint increases significantly which also causes a loss of intensity and a loss of resolution due to the increased beam cross section (Boulle *et al.*, 2002). Additionally, because the incidence angle ω has to remain positive, the maximum accessible tilt angle is equal to θ , whereas there is no such restriction when the ψ tilting technique is used. DxTools is compatible with both type of measurements.

Pole figures:

Pole figures are usually carried out to assess the texture of materials (Kocks, Tomé & Wenk, 1998), including the quantification of epitaxial relations in thin films. For this purpose, a series of χ scans for different φ values (or alternatively, a series of φ scans for different χ values) is recorded using a point detector with an angular acceptance wide enough to include all the diffracted intensity from the investigated reflection. The corresponding geometry is given in Fig. 4. A complete pole figure requires the χ and φ angles to be scanned from 0 to 90° and from 0 to 360°, respectively.

Custom:

In principle there is no restriction on the number and type of motors than can be scanned together. DxTools supports any of such measurement, but its role is then limited to extracting individual scans from the data file.

3. Usage and examples

The DxTools folder contains four Python source files:

- dxtools.py, which is the main program with the graphical user interface,
- data_reader.py, which contains the uxd and brml importing functions,
- data_processor, which contains the processing functions (one for each type of experiment),

- misc.py, which contains various functions (like peak fitting functions) that couldn't fit in any of the above.

With Python 3.5 (together with the SciPy and Matplotlib libraries) installed on the system, DxTools can be launched from a command line window with: `python dxtools.py`, or (for Linux operating systems), `python3 dxtools.py`. DxTools has been successfully tested in Windows 10 and different Linux distributions (Debian and Ubuntu). The typical work-flow with DxTools is as follows:

- import data from the file menu
- select the experiment type from the tab bar
- select the export options and, when relevant, the data analysis options
- process the data

The data is automatically plotted and the extracted data files, as well as the data-analysis files, are exported in text format (with a recognizable name) in the same folder as the input data file. The import and processing time are of course function of the size of the data; with an up-to-date computer, typical times are around 1 sec or less. The different export and data-analysis options are described below with the help of illustrative examples. The corresponding data files are provided in the "Example" folder. All the plots below are directly generated by DxTools and, in the case of surface plots, the intensity is plotted on a logarithmic scale in order to enhance low-intensity details.

3.1 Reciprocal space mapping

A screenshot of the user interface and a RSM generated by DxTools are given in Fig. 5. It corresponds to the 002 reflection of an irradiated 3C-SiC single crystal (Debelle *et al.*, 2010). The data has been recorded using a stationary one-dimensional detector while scanning the incidence angle. Let us first detail the export options (Fig. 5a):

- "Log(intensity) scale": if checked, all the exported data files contain the logarithm of the intensity

instead of the intensity itself.

- “Angular-space intensity matrix”: exports the data in a two-dimensional array where the first line and first column contain the ω and 2θ angles, respectively.

- “Q-space intensity matrix”: exports the data in two-dimensional array where the first line and first column contain the Q_x and Q_z coordinates, respectively. Because the data has to be interpolated, the interpolation step should be provided by the user.

- “3-column format”: exports the data in a 3 column format containing the Q_x , Q_z and intensity values, respectively.

- “Skip points at start/stop”: this option allows the scanning range of the experiment to be reduced directly from the interface, by specifying the number of points to be removed at the beginning and at the end of the scan, for both scanning motors. This is especially useful to remove useless data points as for instance at the edges of a one-dimensional detector where a few channels often exhibit wrong intensity values, or to simply remove data points in an otherwise too large experiment. In the case presented here, the primary motor is 2θ and the secondary motor is ω .

In the RSM, Fig. 5b, the intense peak located at $Q_z = 2.883 \text{ \AA}$ corresponds to the 002 reflection from 3C-SiC, whereas the vertical intensity streak spreading towards lower Q_z values is the signal emanating from the irradiated surface of the crystal (Debelle *et al.*, 2014; Boulle & Debelle, 2016). Inclined streaks (making a $\sim 55^\circ$ angle with Q_z) originating from stacking faults can also be distinguished (Boulle *et al.*, 2009)

3.2 Temperature-dependent scans

Fig. 6 displays temperature-dependent $2\theta/\theta$ scans recorded from a VO_2 thin film grown on a (001)-oriented TiO_2 substrate. VO_2 exhibits a metal-insulator transition around 68°C which is accompanied by a structural phase transition from a monoclinic (insulator) to a tetragonal rutile-like (metal) phase upon heating. The data shown in Fig. 6 corresponds to the 002 reflection of VO_2

(indexed in the tetragonal unit-cell).

There are two export options (Fig. 6a):

- “Individual scans”: the scans recorded at different temperatures are extracted from the file and saved to the hard drive.
- “Temperature/angle intensity matrix”: exports the data in a two-dimensional array where the first line and first column contain the temperature and angles, respectively.

There are two data analysis options that can be used for a rapid evaluation of the data:

- “Numerical intensity integration”: the intensity of each scan is integrated and plotted as a function of temperature. This option is selected in Fig. 6.
- “Peak fitting (one peak only)”: this options performs a fit of the intensity distribution with a pseudo-Voigt function (Wertheim *et al.*, 1974), plus a linear background function. The fitting parameters (peak position, peak area and the full-width at half-maximum [FWHM] of the peak) are plotted as function of temperature. The fit is performed with the least-squares fitting function of the SciPy library.

The point skipping option acts on the angular range and on the temperature range. The phase transition of the material is clearly observed around 60° which gives rise to the appearance of the XRD peak of the high temperature phase at higher angles (around 65.4°) and an overall increase of the intensity as a result of the increased structure factor. The shift of the transition temperature towards lower temperatures is likely due to strain (Théry *et al.*, 2016). This will be subject of a dedicated study.

3.3 Spatial scans

The export and data analysis options are exactly the same as those present in the temperature-dependent experiments (above), with the temperature being replaced with the value of the t_x or t_y translation (Fig. 7a). Fig. 7b shows the data corresponding to a (010)-VO₂ film grown on a (001)-

oriented Al₂O₃ substrate. The film has been positioned on double Peltier heating stage. One Peltier element (for positive translations) is set below the transition temperature (50°C), whereas the other (for negative translations) is set above the transition (90°C). The temperature gradient induces a spatial spread of the phase transition. The upper left panel in Fig. 5b shows the evolution of 2 θ / θ scans of the 020 reflection of VO₂ as a function of the beam position. The phase transition is clearly observed around -2 mm. The increase in intensity at negative translations is due to the increased structure factor in the high-temperature phase. The anomalies in the evolution of the peak position (upper right panel) and peak width (lower right panel) are due to the appearance of an intermediate phase at low angles, and the co-existence of those three phases in the same angular domain, respectively (Théry *et al.*, 2016).

3.4 *Sin*² ψ measurements

The only export option is “Individual scans”, whereas a rapid data analysis can be conducted using either an estimate of the peak position from the data (“d-spacing from peak maximum”) or from a fit of the data with a pseudo-Voigt function using the same approach as above (“d-spacing from peak fitting”). The latter option is chosen in Fig. 8a. The upper part of Fig. 8b shows the data (black circles) and the fit (red lines). The data correspond to the 110 reflection of a thick Mo film. The lower part is a d vs. $\sin^2\psi$ plot which essentially shows, from the lack of a clear visible slope, that the material is free of residual strain (which agrees well with the fact that all peaks are located at the same position in the upper panel).

3.5 Pole figures

The last example correspond to the analysis of micro-twins in epitaxial (001) GaP films epitaxially grown on (001) Si (Wang *et al.*, 2015). Similarly to the previous cases the export options are (Fig. 9a):

- “Individual scans”: the χ scans (respectively φ scans) recorded at different φ (respectively χ) values are extracted from the file and saved to the hard drive.
- “Chi/Phi intensity matrix”: exports the data in a two-dimensional array where the first line and first column contain the χ and φ angles, respectively.
- “3-column format”: exports the data in a 3 column format containing the χ and φ angles, and intensity values, respectively.

Fig. 9b is a pole figure of the (111) lattice planes. The spots observed at $\chi \approx 55^\circ$ exhibit the correct four-fold symmetry expected for the (001) orientation. The additional streak-like spots $\chi \approx 15^\circ$ correspond to micro-twins (Wang *et al.*, 2015). It can be noted that the pole figure is shifted 6° along the χ axis because of a surface miscut that could not be corrected during the sample alignment procedure.

4. Conclusions

This article presents the first version of the program DxTools which is aimed at handling large data files recorded with the Bruker D8 diffractometer. For the moment, DxTools supports the following experiments: reciprocal space mapping, temperature-dependent measurements, spatial scans, $\sin^2\psi$ measurements and pole figures. It should be noted that DxTools only supports data recorded with point and one-dimensional detectors. The support for additional data formats (including additional hardware such as two-dimensional detectors or diffractometers from other manufacturers) or additional experiments can be added by modifying the `data_reader.py` and `data_processor.py` files respectively.

DxTools requires Python 3.5 as well as the SciPy and Matplotlib libraries to be installed on the computer. The program can be used by anyone even without any programming knowledge. The source code of DxTools is open and it can therefore legally be copied and modified (within the terms the CeCILL license), although some basic programming skills are required, especially to

modify the graphical user interface. The program can be downloaded from <https://abouille.github.io/DxTools/> . The DxTools program will be useful for those users willing to keep control on their data.

Acknowledgements

I would like to warmly thank C. Grygiel (CIMAP-GANIL, Caen) for testing the program. I am also grateful to A. Létoublon (INSA, Rennes) for providing the pole figure data and N. Lemée (LPMC, Amiens) for providing other testing data sets. I would also like to thank my colleagues (V. Théry, A. Crunteanu, J. C. Orlianges) for the VO₂ studies.

References

- Boulle, A., Masson, O., Guinebretiere, R., Lecomte, A., Dauger, A. (2002). *J. Appl. Cryst.* 35, 606-614.
- Boulle, A., Aube, J., Galben-Sandulache, I. G., Chaussende, D. (2009). *Appl. Phys. Lett.* 94, 201904.
- Boulle, A., Debelle, A. (2016). *Phys. Rev. Lett.* 116, 245501.
- Conchon, F., Boulle A., Guinebretiere, R. (2007). *Phys. Stat. Sol. (a)* **204**, 2535-2541.
- Debelle, A., Thomé, L., Dompont, D., Boulle, A., Garrido, F., Jagielski, J., Chaussende, D. (2010). *J. Phys. D: Appl. Phys.* 43, 455408.
- Debelle, A., Boulle, A., Chartier, A., Gao, F., Weber, W. J. (2014). *Phys. Rev. B*, 90, 174112.
- Kocks, U. F., Tomé, C. N., Wenk, H. R. (1998). *Texture and Anisotropy*, Cambridge University Press.
- Noyan, I. C. & Cohen, J. B. (1987). *Residual Stress - Measurement by Diffraction and Interpretation* New York: Springer Science+Business Media.
- Oliphant, T. E. (2007). *Comput. Sci. Eng.* **9**, 10-20.
- Pietsch, U., Holý, V., Baumbach, T. (2004). *High-Resolution X-ray Scattering – From Thin Films to Lateral Nanostructures*. New York: Springer-Verlag.
- Schmidbauer, M. (2004). *X-Ray Diffuse Scattering from Self-Organized Mesoscopic Semiconductor Structures*. New York: Springer-Verlag.
- Schuster, M. & Göbel, H. (1995). *J. Phys. D: Appl. Phys.* 28, A270.
- Théry, V., Boulle, A., Crunteanu, A., Orlianges, J. C., Beaumont, A., Mayet, R., Mennai, A., Cosset, F., Bessaudou, A., Fabert, M. (2016). *Phys. Rev. B* 93, 184106.
- Wertheim, G. K., Butler, M. A., West, K. W., Buchanan, D. N. E. (1974). *Rev. Sci. Instr.* 45, 1369-1371.

van de Walt, S., Colbert, S. C., Varoquaux, G. (2011). *Comput. Sci. Eng.* **13**, 22-30.

Wang, Y. P., Letoublon, A., Nguyen Thanh, T., Bahri, M., Largeau, L., Patriarche, G., Cornet, C.,
Bertru, N., Le Corre, A., Durand, O. (2015). *J. Appl. Cryst.* **48**, 702-710.

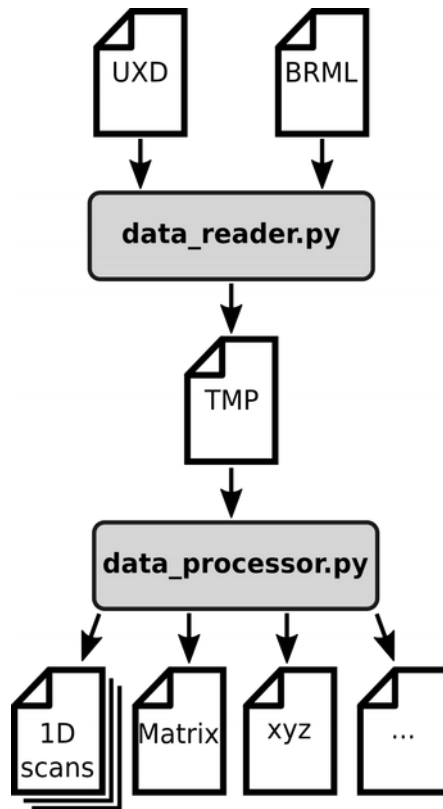


Figure 1: the two-step procedure implemented in DxTools. Data files (in uxd or brml format) are imported with `data_reader.py` routine, stored in a temporary file and then processed with the `data_processor.py` routine.

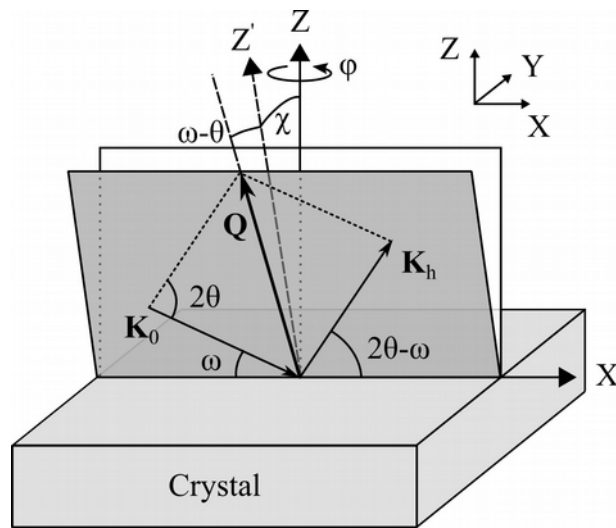


Figure 2: diffraction geometry of the Bruker D8 diffractometer. See text for definitions.

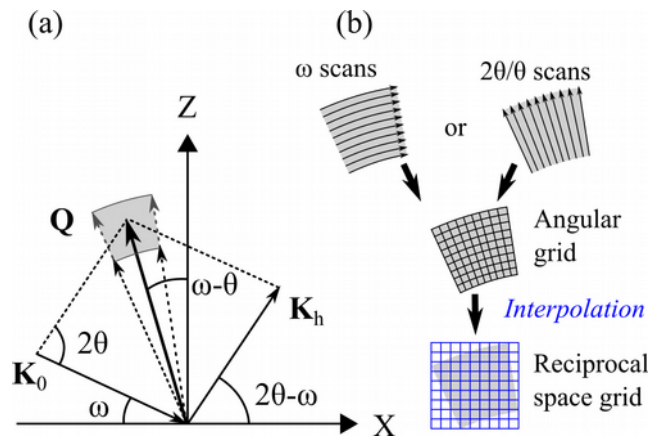


Figure 3: geometry used for reciprocal space mapping. (a) a region of the reciprocal space (gray area) is scanned by varying the length (2θ) and orientation ($\omega - \theta$) of the scattering vector \mathbf{Q} . (b) a series of ω scans or $2\theta/\theta$ scans form a non-orthogonal grid in reciprocal space which has to be interpolated on a square grid.

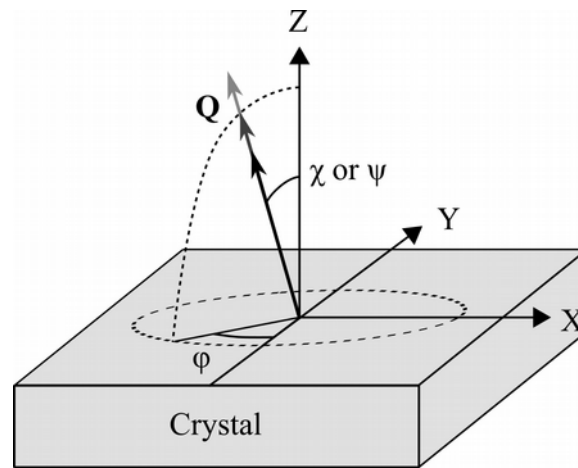


Figure 4: geometry used for $\sin^2\psi$ measurements and pole figures. For $\sin^2\psi$ measurements, $2\theta/\theta$ scans (gray arrows along \mathbf{Q}) are recorded for various tilt values (ψ or χ) and azimuthal orientations (ϕ). For pole figures, a point-detector (with an angular opening large enough to integrate all the intensity diffracted from the lattice planes) is preferred.

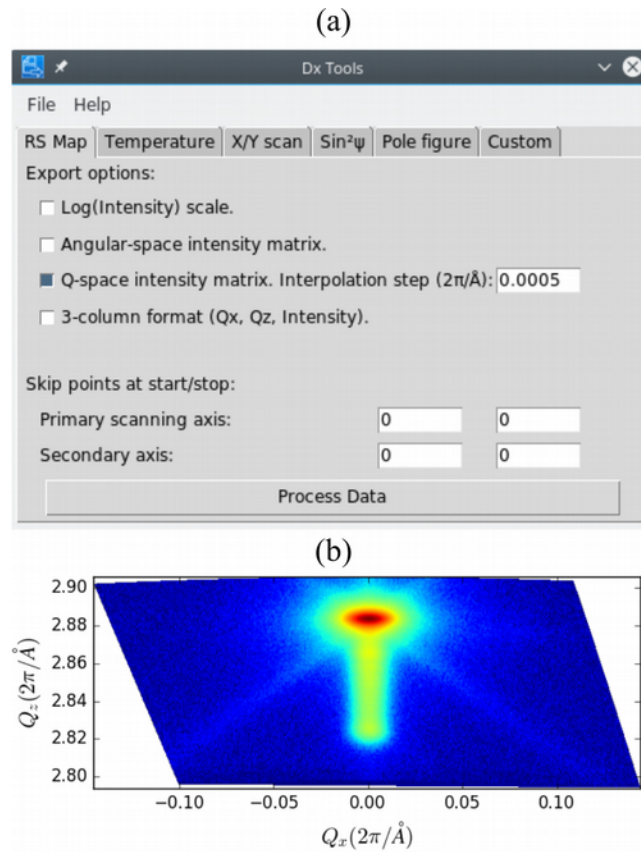


Figure 5: (a) screenshot of the reciprocal space mapping section. See text for definitions. (b) RSM of the 002 reflection of irradiated 3C-SiC.

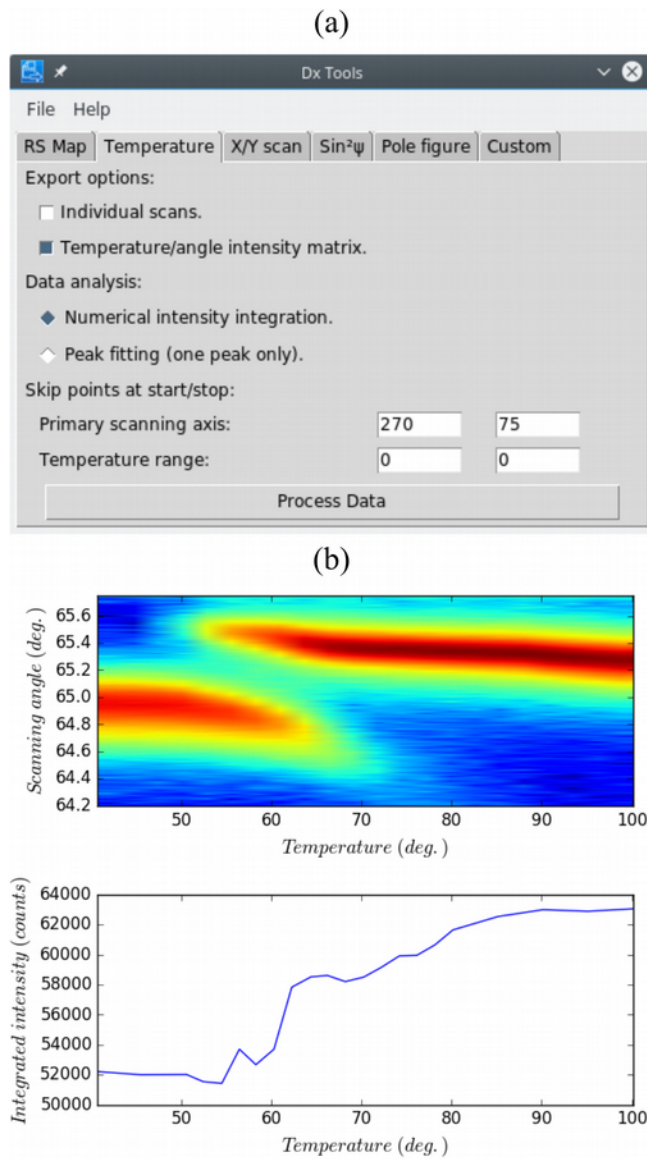


Figure 6: (a) screenshot of the temperature-dependent XRD section. See text for definitions. (b) evolution of the intensity distribution and the integrated intensity with temperature.

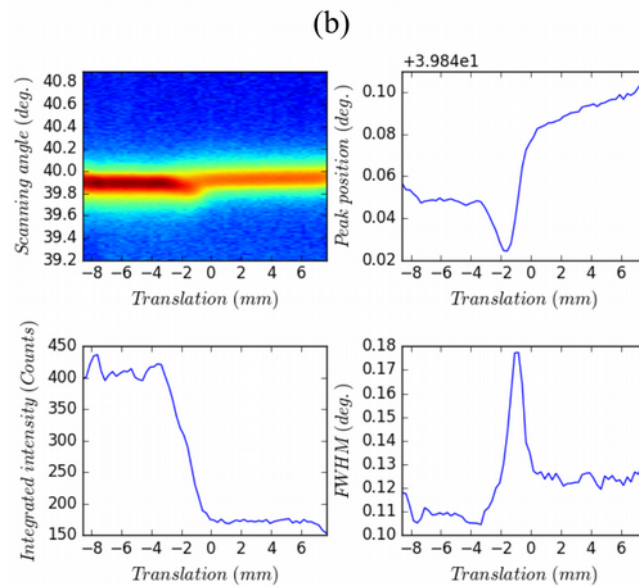
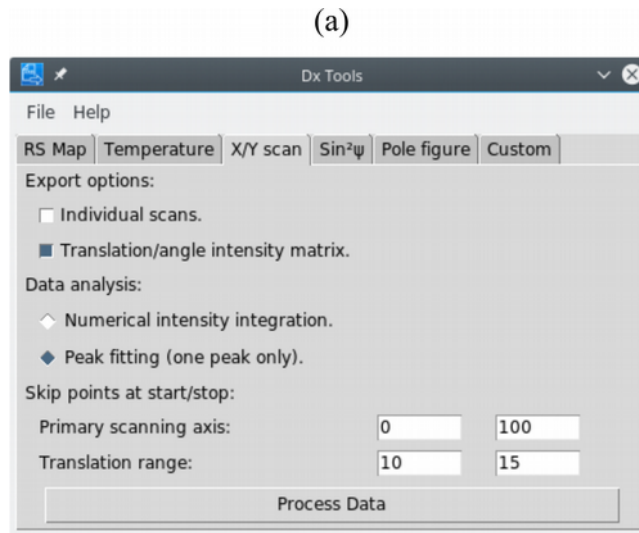


Figure 7: (a) screenshot of the spatial-scan section. See text for definitions. (b) evolution of the intensity distribution and pseudo-Voigt fitting parameters as a function of the sample position.

Figure 8: (a) screenshot of the $\sin^2\psi$ section. See text for definitions. (b) experimental (black circles) and fitted (red lines) curves (upper part) and d vs. $\sin^2\psi$ plot (lower part).

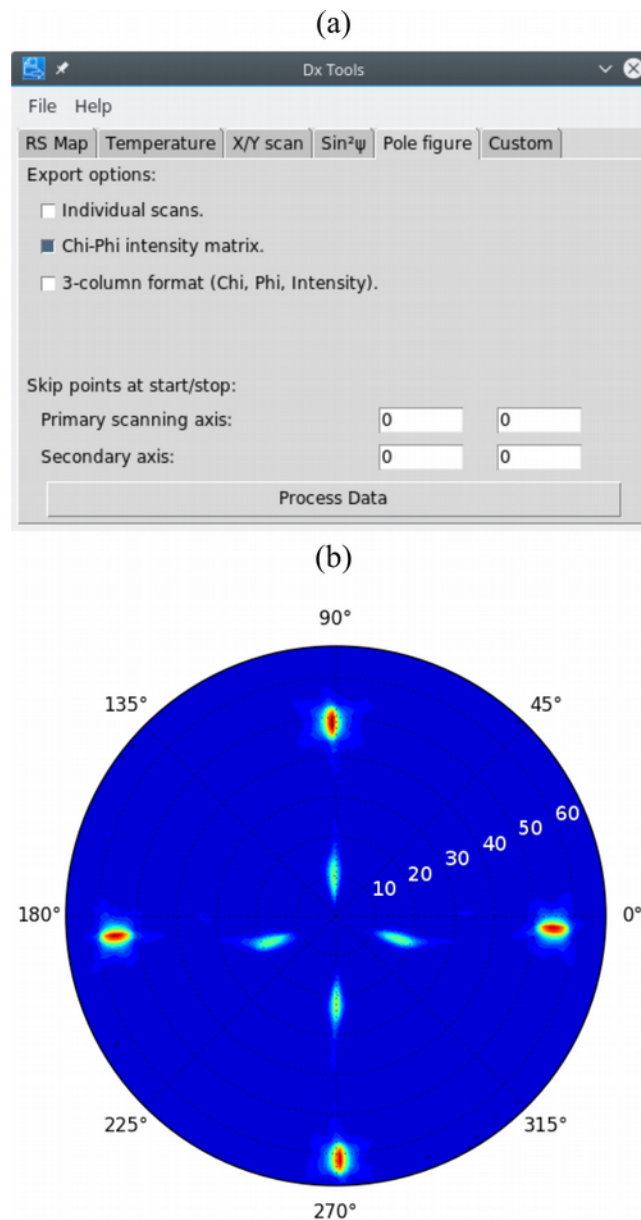


Figure 9: (a) screenshot of the pole figure section. See text for definitions. (b) (111) pole figure of GaP.