TORRICELLI: A software to determine atomic spatial distribution from normal incidence x-ray standing wave data

– Manual of version 3.8 –

See Ref. [1] for a review of the used theory and approximations.

The latest updates are available on www.torricelli-software.com.

F. C. Bocquet, G. Mercurio and M. Franke

SEPTEMBER 14, 2018

(Image in the Public Domain)

Preface

TORRICELLI is a software designed for the analysis of x-ray standing wave (XSW) data. While the XSW technique has been employed for several decades, to our knowledge no free, open source, user-friendly and well-documented program for conducting XSW data analysis exists to date. TORRICELLI is therefore an attempt to fill this gap. The spatial distribution of atomic species with respect to the atomic planes of a single crystal can be described by two parameters, the coherent position P_c and the coherent fraction F_c. The main target of TORRICELLI is to determine this pair of parameters in the most accurate way, and also provide the corresponding statistical errors [2, 1].

We encourage the readers to make suggestions that could improve the program as well as the present manual. If you can program in python you are also welcome to take part in the programming. TORRICELLI is distributed under the GNU General Public License v3. You should have received a copy of the GNU General Public License along with TORRICELLI. If not, see `https://www.gnu.org/licenses`. This manual is distributed under the creative commons Attribution-ShareAlike license (CC BY-SA 4.0), see `https://creativecommons.org/licenses/by-sa/4.0/legalcode`.

We would like to thank particularly Tien-Lin Lee for discussing the fitting equations and their implementation. Of course, all the TORRICELLI users are also thanked for reporting small bugs, and encouraging development.

Contents

— 1 —

Getting started

1.1 Installation

TORRICELLI is an interpreted program. That means first you need to install Python and a few modules in order to be able to launch TORRICELLI (which itself does not require any installation). TORRICELLI was programmed in `Python2.7` and `PyQt4`. Compatibility to `Python3` and/or `PyQt5` is not supported at the moment. If you already have a scientific `Python2.7` installation, you can test your installation by trying to import the following packages in a python console:

```
>>> import scipy, numpy, cmath, lmfit
>>> import pyqtgraph
>>> from pyqtgraph.Qt import QtGui, QtCore
>>> from PyQt4.QtCore import pyqtSignal, pyqtSlot
>>> from PyQt4 import QtCore, QtGui
>>> import ConfigParser, csv, sip, colorsys
>>> import sys, string, os, datetime, glob, ast
>>> from distutils.version import StrictVersion
>>> from operator import itemgetter
```

That will load one by one all the necessary modules required by TORRICELLI. *None* of these lines should return *any* error message. If it does, that means there are missing packages that should be installed, as explained in the following. You may see a 'FutureWarning' message depending on the version you use. Also there is a small bug in one of the packages that is needed to run Torricelli. There is a workaround explained after the installation instructions.

Linux

Simply install the following packages with your preferred package manager: `Pyhton2.7`, `PyQt4`, `pyqtgraph`, `numpy`, `scipy`, and `lmfit`

Note Once `Pyhton2.7` is installed, you can also use `pip` to install the rest (useful to get the latest `pyqtgraph` version on some distributions).

Windows

We recommend to install the package manager `Miniconda` for `Python 2.7` (https://conda.io/miniconda.html). During the installation, check the box *add Anaconda to the PATH* (if not, you will have to use the anaconda prompt for Torricelli, which is fine but less practical). After installation, execute the following commands:

- `conda install pyqt=4 numpy scipy`
- `pip install lmfit pyqtgraph`

Mac

We recommand to install the package manager `Homebrew` (https://brew.sh). After installation, execute the following commands in a console:

- `brew install python`
- `sudo easy_install pip` (may not be necessary with the latest homebrew version)
- `pip install pyqt numpy scipy lmfit pyqtgraph`

Small bug There is a tiny bug in the latest version of pyqtgraph (0.10.0). The problematic line number and file name will display in the console (`ImageExporter.py` at line 70) when you use Torricelli. A couple of **int()** must be added:

```
bg = np.empty((int(self.params['width']),
               int(self.params['height']), 4), dtype=np.ubyte)
```

You will find the file in the directory `...\pyqtgraph\exporters` in your site-packages directory. If you do not know where your site-package folder is, type
`import site; site.getsitepackages()` in the python console and it will be displayed.

Upgrading packages

To upgrade any of the packages, one can simply run the following command:
`conda update <package_name>`
or
`pip install <package_name> -U`
depending on how you installed the package.

1.2 Conventions used in this manual

- 🗁 `‣Here‣folder`: designates the sub-path in the folder containing the Torricelli program.

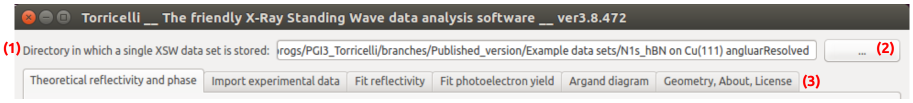
- 🗋▸`Here`▸`the`▸`file.txt`: designates a file in a sub-folder of the TORRICELLI program.
- 🗁𝒟▸`Some`▸`folder`: designates the sub-path in the folder containing the data to be analyzed.
- 🗋𝒟▸`All`▸`data1.dat`: designates a file in a sub-folder of the folder containing the data to be analyzed.
- The coming references in bracket, like (n), indicate fields or buttons marked in the region of interest of the program screenshot shown in the figures.

1.3 Launching Torricelli

There are several possibilities to start TORRICELLI.

- We recommend to open a console, change directory (command `cd` on Linux, Mac and Windows) to the program folder (🗁▸) and execute `python Torricelli.py`.
- One also can double click on the file 🗋▸`TORRICELLI.py` if it is executable on your system.
- Do not start TORRICELLI from an interactive python shell.

Before starting an analysis, one must first define the directory that contains your experimental files (🗁𝒟▸) (1), by clicking on the [...] button (2). In this folder, a 🗁𝒟▸`results` sub-folder is created and will contain all results (as ASCII files and pictures).

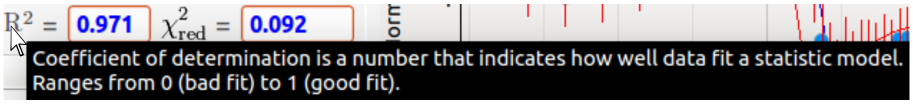


Directory in which a single XSW data set is stored: rogs/PGI3_Torricelli/branches/Published_version/Example data sets/N1s_hBN on Cu(111) angluarResolved

Figure 1.1: The tab structure of TORRICELLI.

TORRICELLI is organized in tabs (3) (Fig. 1.1). There is a separate tab for each important step of the data analysis. One can scroll from tab to tab with the mouse wheel, or using the shortcuts [Ctrl]+[PageUp] and [Ctrl]+[PageDown]. In this manual, a chapter is dedicated to each tab necessary to the data analysis. The details of the algorithm are described in Ref. [1] and should be read together with this manual.

We tried to add as much information as possible in 'tooltips', see Fig. 1.2. Rest the mouse cursor on a given object for one second, and a box will appear with some explanations if there is anything to clarify.



R^2 = 0.971 χ^2_{red} = 0.092

Coefficient of determination is a number that indicates how well data fit a statistic model.
Ranges from 0 (bad fit) to 1 (good fit).

Figure 1.2: Example of a tooltip on R^2.

— 2 —

Theoretical reflectivity and phase

The calculation of the structure factors requires the knowledge of the (sample and monochromator) crystallographic parameters: lattice constants, and list of atoms in the unit cell. Besides, the Debye-Waller factor can also be added if known.

There exist two databases of crystallographic parameters, for elemental (🗎▸`CrystallographicData_Elemental.csv`) and for compound (🗎▸`CrystallographicData_Compound.csv`) samples. They can be found in the folder 🗁▸`imports`▸`Databases`▸`Lattices`. Each crystal has a dedicated line, with comma separated values, e.g.,

🗋▸`imports`▸`Databases`▸`Lattices`▸`CrystallographicData_Elemental.csv`

Contains the lattice constants of all elemental crystal:

```
Z,Name,cell_type,a,b,c,alpha,beta,gamma,checked_values
14,Si,diamond,543.09,543.09,543.09,90,90,90,yes
28,Ni,faceCentered,352.4,352.4,352.4,90,90,90,no
29,Cu,faceCentered,361.49,361.49,361.49,90,90,90,yes
⋮
```

First comes the atomic number of the atom(s), the element name, the type of unit cell, the three lattice lengths (a, b, c) in [pm], the three lattice angles (α, β, γ) in [°]. The last argument (checked_values) specifies if the values has already been used in the analysis of NIXSW data. This is then displayed in the graphical user interface to draw the user's attention on crystallographic parameters that were taken from various references but not tested (see file 🗋▸`imports`▸`Databases`▸`Database references.txt` for the sources). If your sample has a complex unit cell that is not in the database, you can create a file named 🗋▸`imports`▸`Databases`▸`Lattices`▸`AtomCoordinates_name-of-unit-cell.csv` that lists the relative position of all atoms. For example, for the face-centered unit cell we have:

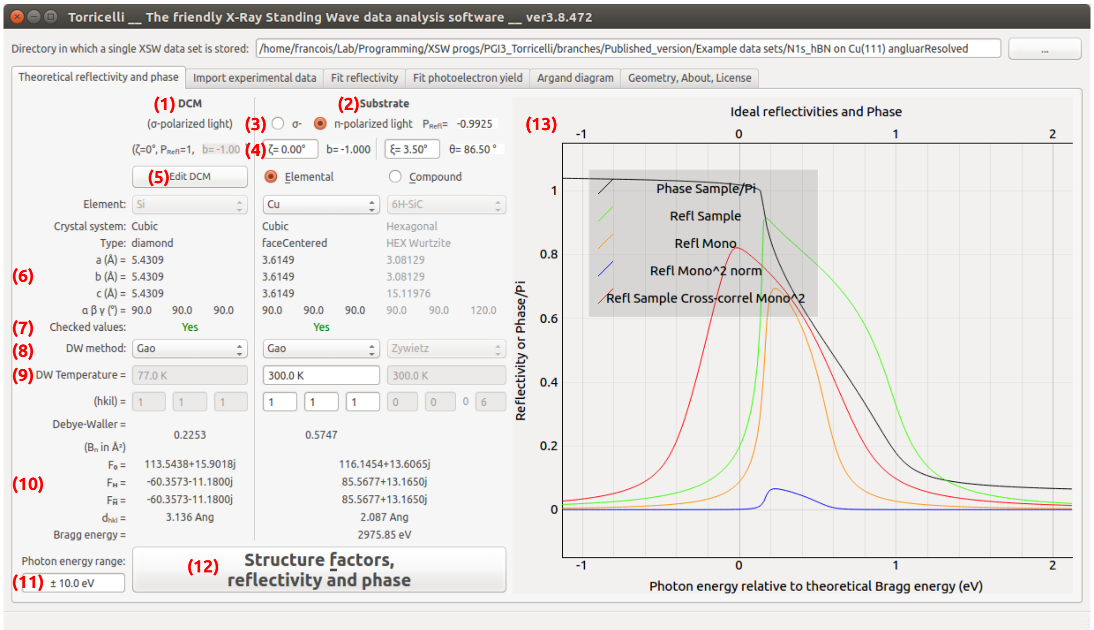
`▸imports▸Databases▸Lattices▸AtomCoordinates_faceCentered.csv`

Atomic positions in the face centered unit cell:

```
Element,x (a),y (b),z (c)
A,0,0,0
A,.5,.5,0
A,.5,0,.5
A,0,.5,.5
```

Then just add a line in the `▸imports▸Databases▸Lattices▸CrystallographicData_*.csv` file that indicates `name-of-unit-cell` in the column `cell_type` as well as the atomic mass and the crystal parameters. After restarting Torricelli, the new crystal will be included in the database.

Note If you successfully used not-checked-yet values in experiments, or included new samples/unit cells into the database, please contact the developers to include your values to the next version of Torricelli.



Torricelli __ The friendly X-Ray Standing Wave data analysis software __ ver3.8.472

Directory in which a single XSW data set is stored: /home/francois/Lab/Programming/XSW progs/PGI3_Torricelli/branches/Published_version/Example data sets/N1s_hBN on Cu(111) angluarResolved

(1) DCM (σ-polarized light)

(2) Substrate

(3) ○ σ- ● π-polarized light P_{Refl}= -0.9925

(ζ=0°, P_{Refl}=1, b= -1.00 (4) ζ= 0.00° b= -1.000 ξ= 3.50° θ= 86.50 °

(5) Edit DCM ● Elemental ○ Compound

	DCM	Substrate	
Element:	Si	Cu	6H-SiC
Crystal system:	Cubic	Cubic	Hexagonal
Type:	diamond	faceCentered	HEX Wurtzite
a (Å) =	5.4309	3.6149	3.08129
(6) b (Å) =	5.4309	3.6149	3.08129
c (Å) =	5.4309	3.6149	15.11976
α β γ (°) =	90.0 90.0 90.0	90.0 90.0 90.0	90.0 90.0 120.0
(7) Checked values:	Yes	Yes	
(8) DW method:	Gao	Gao	Zywietz
(9) DW Temperature =	77.0 K	300.0 K	300.0 K
(hkil) =	1 1 1	1 1 1	0 0 0 6
Debye-Waller = (B_n in Å²)	0.2253	0.5747	
F_0 =	113.5438+15.9018j	116.1454+13.6065j	
(10) F_H =	-60.3573-11.1800j	85.5677+13.1650j	
$F_{\bar{H}}$ =	-60.3573-11.1800j	85.5677+13.1650j	
d_{hkl} =	3.136 Ang	2.087 Ang	
Bragg energy =		2975.85 eV	

Photon energy range: (11) ± 10.0 eV

(12) Structure factors, reflectivity and phase

Figure 2.1: Screenshot of the *Theoretical reflectivity and phase* tab of Torricelli.

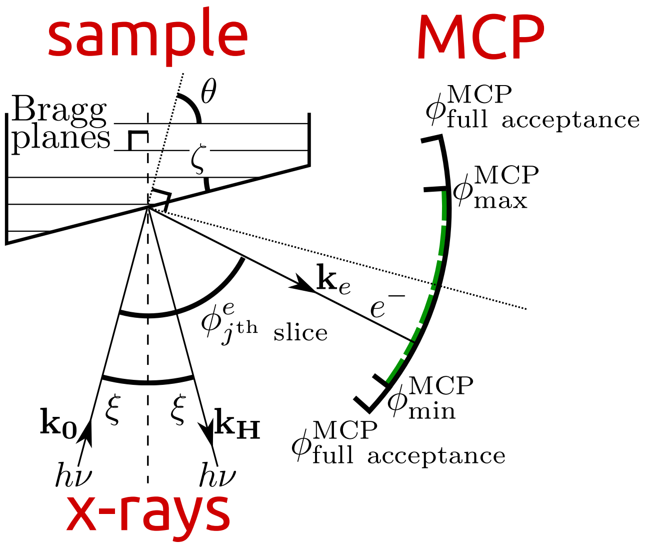


Figure 2.2: Definition of all angles used by TORRICELLI (ζ and ξ have positive values in the picture). This picture can also be found in the *Geometry, About, License* tab of TORRICELLI.

In order to calculate structure factors necessary to simulate the theoretical reflectivity and phase, the following steps need to be taken. First define the photon energy range for which you wish to compute the theoretical curves (11) (see Fig. 2.1). This must be larger than the experimental range. Then one needs to specify the double crystal monochromator (DCM) (1) and sample (2) parameters. The two beam-lines on which we have some experience (I09 at the Diamond Light Source and ID32 at the ESRF) are both equipped with a Si(111) DCM. In order to correctly calculate a theoretical reflectivity curve corresponding to the experiments, TORRICELLI takes the DCM into account. Typically, Si crystals are cooled with liquid nitrogen and have a temperature of 77 K, and the (111) Bragg reflection is used. DCM parameters may be changed by clicking the Edit DCM button (5). The light polarization is chosen in (3). Now one needs to set up the sample. ζ (4) is the the angle between the sample surface and the Bragg planes. ξ (4) is the deviation from perfect normal incidence on the Bragg planes ($\xi = 3.5°$ at the I09 beamline). See all angles depicted in Fig. 2.2. The parameters extracted from the database are displayed in (6). If those values were used in an actual XSW experiment, Yes will appear in (7). If No, please take the given values with care.

> **Important** In the present state of the program, the *DW Temperature* (9) only influences the calculation of the Debye-Waller factor. It does not affect the lattice parameter of your crystal. If you want to use a different crystal temperature, we advise to create a new line in the 🗋▸`imports`▸`Databases`▸`Lattices`▸`CrystallographicData_*.csv` file, use an explicit name, and adapt the lattice constant values.

Once you defined the sample (2), the *hkl* Miller indices of the wanted Bragg planes reflection (10) and the type of Debye-Waller factor (8), click on [Structure factors, reflectivity and phase] (12) to first compute the structure factors. If the Bragg reflection is allowed, the structure factor values will be displayed (11) and the sample reflectivity $R_{\mathrm{S}}^{\mathrm{theo}}(h\nu)$, sample phase $\Phi_{\mathrm{S}}^{\mathrm{theo}}(h\nu)/\pi$, monochromator crystal reflectivity $R_{\mathrm{M}}^{\mathrm{theo}}(h\nu)$, the reflectivity of the double crystal monochromator $R_{\mathrm{DCM}}^{\mathrm{theo}}(h\nu) = \left(R_{\mathrm{M}}^{\mathrm{theo}}\right)^2(h\nu)$, and cross-correlation between the sample reflectivity and the double crystal monochromator reflectivity $R_{\mathrm{S+DCM}}^{\mathrm{theo}}(h\nu) = \left(R_{\mathrm{S}}^{\mathrm{theo}} \star \left(R_{\mathrm{M}}^{\mathrm{theo}}\right)^2_{\mathrm{Norm}}\right)(h\nu)$ will be computed and displayed in (13). Here and in the rest of the program, all curves are always shifted in photon energy such that 0 eV corresponds to the theoretical Bragg energy.

Note From now on, Torricelli will remember the theoretical reflectivities and phases, and one does not need to redo this step unless Torricelli is restarted, or if data from a different sample are to be analyzed.

Auto-save After clicking on (12), the structural factors are be saved in 🗁$\mathcal{D}$▸`results`▸`Structure Factor.dat`, the, theoretical curves are saved in 🗁$\mathcal{D}$▸`results`▸`Theoretical values.dat` and a screenshot of the plot is saved in 🗁$\mathcal{D}$▸`results`▸`Theoretical values.png`. If these files already exist, they will be overwritten.

— 3 —

Import experimental data

Note For users of the I09 beam-line of the Diamond synchrotron, we advise to use the I09DataBrowser, a small software that converts the raw data to ASCII files ready to be used by CasaXPS and TORRICELLI. I09DataBrowser is developed and maintained by the TORRICELLI team.

In the following, the steps to be taken to import and normalize reflectivity and yield data are explained.

1. Choose the text files containing both the experimental reflectivity and I_0^{exp} (the intensity of the incident x-ray beam) (1) in Fig. 3.1 and electron yield (2) curves (standard CasaXPS output), by clicking on the corresponding ... button (3). They should have the following formats:

🗋𝒟▸`*.refl`:
Contains the reflectivity and the beam intensity:

```
Energy I_refl I0
2460.92 224.0 215042.0
2460.99 225.0 214766.0
2461.05 258.0 214328.0
⋮
```

🗋𝒟▸`*.txt` or 🗋𝒟▸`*.ey`:
Contains the yield of each components as well as the standard deviations:

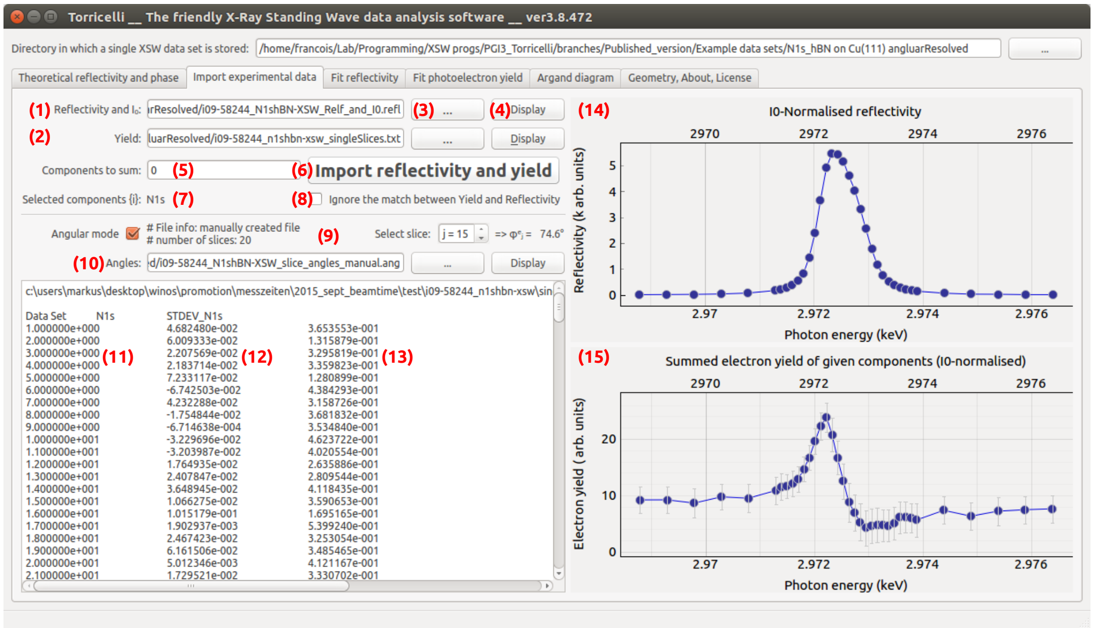
```
Path of the file containing the full CasaXPS analysis (.vms)

Photon energy   Reg_Area(0) CPSeV  StDev_Reg_Area CPSeV ···
2.460920e+003  3.558958e+005        8.186899e+002 ···
2.460990e+003  3.595590e+005        7.855856e+002 ···
2.461050e+003  3.510015e+005        7.930356e+002 ···
⋮
```

Note You can use the Display buttons (4) to quickly check the content of a file, for instance, compatible number of points, or presence of some aberrant points (11-13). If you decide to remove some points from the data set, simply remove the corresponding lines *in both files* using a text editor, and load again.

2. Choose which component (5), or which set of components $\mathscr{S}$ you wish to analyze. In the field `Fit components` you can list the components numbers, separated by a space (e.g. '1' or '3 2 5'), that will be analyzed. Typically, a photoemission spectrum is fitted with one or more peaks, here also called components. If there is more than one, they will be summed. If nothing is given, the component 0 is the default. The component number start from 0 for the leftmost column, and it increments for columns going to the right. Every column with a component area (12) should be immediately followed by a column with the corresponding statistical errors (13).

3. By clicking on the Import reflectivity and yield button (6), the data are normalized by I_0^{exp}, and displayed (14 and 15). TORRICELLI will automatically check that both files have the same photon energies in the first column (111). If the photon energies are not defined as shown in the instance chosen for Fig. 3.1, this option can be removed by checking `Ignore the match between EY and Refl` (8). This check does not influence the data analysis, it only makes sure you are not mixing different data files. In angular mode, this check is not performed.



Directory in which a single XSW data set is stored: /home/francois/Lab/Programming/XSW progs/PGI3_Torricelli/branches/Published_version/Example data sets/N1s_hBN on Cu(111) angluarResolved

Reflectivity and I₀: ırResolved/i09-58244_N1shBN-XSW_Relf_and_I0.refl

Yield: luarResolved/i09-58244_n1shbn-xsw_singleSlices.txt

Components to sum: 0

Selected components {i}: N1s

☐ Ignore the match between Yield and Reflectivity

Angular mode ☑ # File info: manually created file
number of slices: 20

Select slice: j = 15 => φᵉⱼ = 74.6°

Angles: ·d/i09-58244_N1shBN-XSW_slice_angles_manual.ang

```
c:\users\markus\desktop\winos\promotion\messzeiten\2015_sept_beamtime\test\i09-58244_n1shbn-xsw\sin

Data Set        N1s             STDEV_N1s
1.000000e+000                   4.682480e-002           3.653553e-001
2.000000e+000                   6.009333e-002           1.315879e-001
3.000000e+000                   2.207569e-002           3.295819e-001
4.000000e+000                   2.183714e-002           3.359823e-001
5.000000e+000                   7.233117e-002           1.280899e-001
6.000000e+000                   -6.742503e-002          4.384293e-001
7.000000e+000                   4.232288e-002           3.158726e-001
8.000000e+000                   -1.754844e-002          3.681832e-001
9.000000e+000                   -6.714638e-004          3.534840e-001
1.000000e+001                   -3.229696e-002          4.623722e-001
1.100000e+001                   -3.203987e-002          4.020554e-001
1.200000e+001                   1.764935e-002           2.635886e-001
1.300000e+001                   2.407847e-002           2.809544e-001
1.400000e+001                   3.648945e-002           4.118435e-001
1.500000e+001                   1.066275e-002           3.590653e-001
1.600000e+001                   1.015179e-001           1.695165e-001
1.700000e+001                   1.902937e-003           5.399240e-001
1.800000e+001                   2.467423e-002           3.253054e-001
1.900000e+001                   6.161506e-002           3.485465e-001
2.000000e+001                   5.012346e-003           4.121167e-001
2.100000e+001                   1.729521e-002           3.330702e-001
```

Figure 3.1: Screenshot of the *Import experimental data* tab of TORRICELLI.

4. In the case of angular-resolved acquisition of photoelectrons, we recommend to save each angular-resolved data set (slice) sequentially in the yield file (for the same reflectivity file). It is then possible to choose the slice number (9) to be analyzed by checking `Angular mode`. The corresponding ϕ_j^e must be given in the file `⋆.ang` file (10) so that it can be displayed in this tab, and also copied in the Fit Yield tab (See Fig. 2.2) in view of the calculation of the photoemission correction parameters.

🗋$\mathcal{D}$▸`⋆.ang`: Contains the ϕ_j^e value corresponding to each slice j

```
# this file contains the angles corresponding to the centers of the slices
# File info: manually created file
# number of slices: 20

slice angle
0     116.6
1     113.8
2     111.0
3     108.2
4     105.4
5     102.6
⋮       ⋮
```

In CasaXPS, you just have to put all spectra in the same folder before you load. The file name is used to order the files. Note that by simply scrolling on the slice number, the data are automatically imported (you do not need to press the button (6)).

Note If the working directory (see (1) in Fig. 1.1) is changed, TORRICELLI will try to find both the reflectivity, yield and angle files in this directory. They should have the extensions `.refl`, `.txt` or `.ey` and `.ang` respectively. If several files of the same type are present, the user has to choose the correct file manually.

Important Do not forget that the definition of ϕ_j^e includes ξ.

Important If you are using CasaXPS (or a similar program) to fit your spectra you have to pay attention to the statistical error calculation. CasaXPS uses the Monte Carlo method to fit the data [2]. To estimate the statistical errors CasaXPS introduces a theoretical noise and assumes it to follow a Poisson distribution. This is a reasonable assumption for pulse counted data. However in modern detectors which rather use MCPs than channeltrons it can happen that your experimental noise is not Poisson distributed. In that case the statistical errors from the fits of your spectra will be wrongly estimated. To check in CasaXPS if your noise is Poisson distributed and to correct it, open a spectrum and fit a region on a flat area (just background intensity) of your spectrum using `regression` as background type. Than activate the display of the residuals. CasaXPS should show the `Residual STD` of your region fit. If your noise is Poisson distributed this value should be around 1. If that is not the case there is an option for a correction. First select all your spectra. In the `Processing` window, go to the tab `Calculator` and press the button Poisson Adjust Selection. When you now check the residual STD they should be around 1. To have the option `Poisson Adjust` you have to use at least version `2.3.17 PR 1.1` of CasaXPS.

Auto-save At this point nothing new is saved.

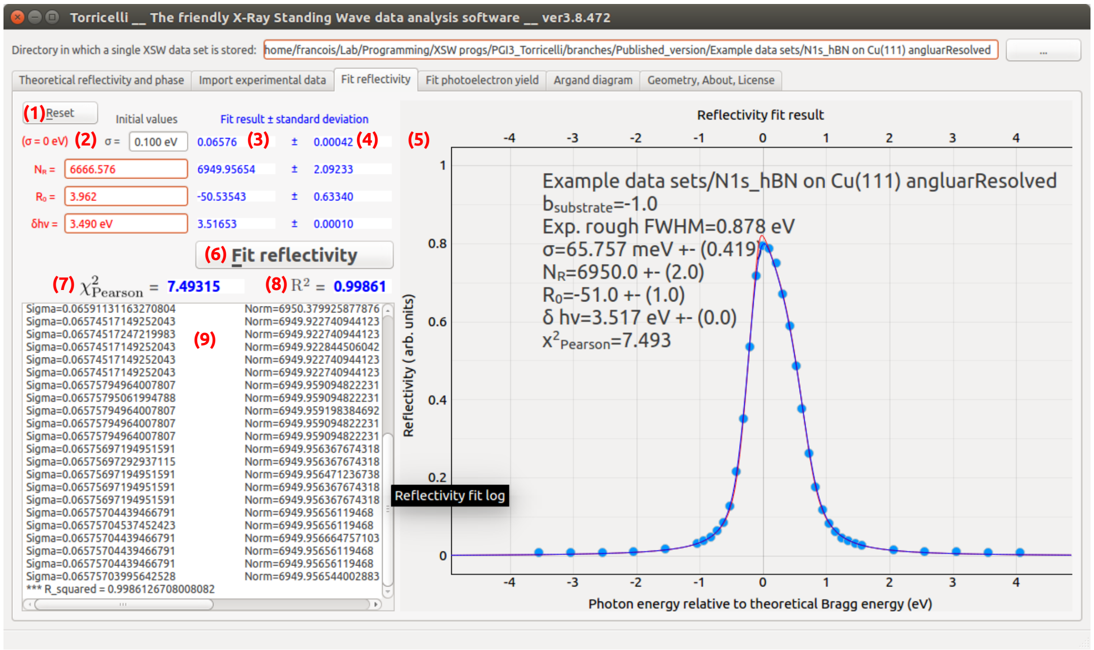
— 4 —

Fit reflectivity

The fit is performed by minimizing

$$\mathcal{R}_R(\underline{\delta h\nu}, \underline{\sigma}, \underline{R_0}, \underline{N_R}) = R^{\text{model}}(h\nu, \underline{\sigma}) - \frac{R^{\text{exp}}(h\nu + \underline{\delta h\nu}) - \underline{R_0}}{\underline{N_R}} \tag{4.1}$$

using the Levenberg-Marquardt method [3, 4, 5]. By default, the data points in the residuals are not weighted by a standard error because the later is not known. If one wishes to weight each reflectivity data point differently (e.g., if the standard errors are known), the code needs to be modified. The fitting parameters are: the background $\underline{R_0}$, the normalization $\underline{N_R}$, the photon energy shift $\underline{\delta h\nu}$ and the gaussian broadening $\underline{\sigma}$. The initial guess of $\underline{N_R}$ and $\underline{R_0}$ are



Directory in which a single XSW data set is stored: home/francois/Lab/Programming/XSW progs/PGI3_Torricelli/branches/Published_version/Example data sets/N1s_hBN on Cu(111) angluarResolved

	Initial values	Fit result	±	standard deviation
σ =	0.100 eV	0.06576	±	0.00042
N_R =	6666.576	6949.95654	±	2.09233
R_0 =	3.962	-50.53543	±	0.63340
δhν =	3.490 eV	3.51653	±	0.00010

χ^2_{Pearson} = 7.49315 $\quad$ R^2 = 0.99861

```
Sigma=0.0659113116327080 4                Norm=6950.379925877876
Sigma=0.06574517149252043                 Norm=6949.922740944123
Sigma=0.06574517247219983                 Norm=6949.922740944123
Sigma=0.06574517149252043                 Norm=6949.922844506042
Sigma=0.06574517149252043                 Norm=6949.922740944123
Sigma=0.06574517149252043                 Norm=6949.922740944123
Sigma=0.06575794964007807                 Norm=6949.959094822231
Sigma=0.06575795061994788                 Norm=6949.959094822231
Sigma=0.06575794964007807                 Norm=6949.959198384692
Sigma=0.06575794964007807                 Norm=6949.959094822231
Sigma=0.06575794964007807                 Norm=6949.959094822231
Sigma=0.06575697194951591                 Norm=6949.956367674318
Sigma=0.06575697292937115                 Norm=6949.956367674318
Sigma=0.06575697194951591                 Norm=6949.956471236738
Sigma=0.06575697194951591                 Norm=6949.956367674318
Sigma=0.06575697194951591                 Norm=6949.956367674318
Sigma=0.06575704439466791                 Norm=6949.95656119468
Sigma=0.06575704537452423                 Norm=6949.95656119468
Sigma=0.06575704439466791                 Norm=6949.956664757103
Sigma=0.06575704439466791                 Norm=6949.95656119468
Sigma=0.06575704439466791                 Norm=6949.95656119468
Sigma=0.06575703995642528                 Norm=6949.956544002883
*** R_squared = 0.9986126708008082
```

Example data sets/N1s_hBN on Cu(111) angluarResolved
$b_{\text{substrate}}$=-1.0
Exp. rough FWHM=0.878 eV
σ=65.757 meV +- (0.419)
N_R=6950.0 +- (2.0)
R_0=-51.0 +- (1.0)
δ hv=3.517 eV +- (0.0)
x^2_{Pearson}=7.493

Figure 4.1: Screenshot of the *Fit reflectivity* tab of TORRICELLI.

such that the reflectivity difference between maximum and minimum as well as origin are the same between the data points and the theoretical curve. $\underline{\delta h\nu}$ is chosen such that the highest reflectivity point of the data and of the theoretical curve lie at the same photon energy. The default initial value of $\underline{\sigma}$ is 0.1. All initial values are displayed in red and fit results in blue.

Note The lower the value σ, the better the crystal quality. The value of σ for SiC is typically 0.05 eV, and for coinage metals 0.1-0.2 eV.

Press the Fit reflectivity (6) button (see Fig. 4.1). If the fit converges, the normalized experimental data together with the fitting theoretical curve are displayed as blue points and line (5), respectively. The experimental points correspond to the first term of Eq. 4.1. The red line corresponds to $R_{\mathrm{S}}^{\mathrm{theo}} \star \left(R_{\mathrm{M}}^{\mathrm{theo}}\right)^2_{\mathrm{Norm}}$, that is *without* gaussian broadening ($\sigma = 0$).

If the fit does not converge, you can try to modify the initial parameters (2) (e.g., increasing $\underline{\sigma}$ usually helps to find convergence, the other values are usually very good). The fit results are saved as text file and picture, see Chap. 10. It is possible to reset the initial values to the guess made by TORRICELLI by clicking on Reset (1).

In the absence of standard errors, the standard χ^2_{red} is not defined. Therefore, the Pearson's chi-squared ($\chi^2_{\mathrm{Pearson}}$, see Eq. 1.8 in Ref. [7]) and the coefficient of determination (R^2, see Sec. 1.3 and 11.2 in [8]) are displayed (7, 8) in order to evaluate the quality of the fit [6, 8] without the knowledge of the variance. They are defined as follows:

$$\chi^2_{\mathrm{Pearson}} \equiv \frac{1}{\mathcal{DF}} \sum_{\mathrm{exp.h}\nu} \frac{\Big(R^{\mathrm{exp}}(h\nu) - R^{\mathrm{model}}(h\nu)\Big)^2}{R^{\mathrm{model}}(h\nu)}$$

and

$$R^2 \equiv 1 - \frac{SS_{\mathrm{res}}}{SS_{\mathrm{tot}}}, \tag{4.2}$$

with $\mathcal{DF}$ the number of degree of freedom and

$$SS_{\mathrm{res}} = \sum_{\mathrm{exp.\ h}\nu} \left(R^{\mathrm{exp}}(h\nu) - R^{\mathrm{model}}(h\nu)\right)^2 \tag{4.3}$$

$$SS_{\mathrm{tot}} = \sum_{\mathrm{exp.\ h}\nu} \left(R^{\mathrm{exp}}(h\nu) - \bar{R}^{\mathrm{exp}}\right)^2 \tag{4.4}$$

$$\bar{R}^{\mathrm{exp}} = \frac{1}{n} \sum_{\mathrm{exp.\ h}\nu}^{n} R^{\mathrm{exp}}(h\nu) \tag{4.5}$$

where the sums run over all experimental $h\nu$ values, and n is the total the number of experimental points. R^2 values range from 0 (bad fit) to 1 (good fit).

Important If this fit is not good, it is of no use to continue the analysis! Check the parameters you used for the sample crystal.

Auto-save After clicking (6), the fit results are saved as ASCII file and pictures in 🗁$\mathcal{D}$▸`results`▸`Exp_refl_norm_centred.dat`, 🗁$\mathcal{D}$▸`results`▸`Fit_refl.log`, 🗁$\mathcal{D}$▸`results`▸`Fit_result_refl.png` and 🗁$\mathcal{D}$▸`results`▸`Fit_result_refl.dat`.

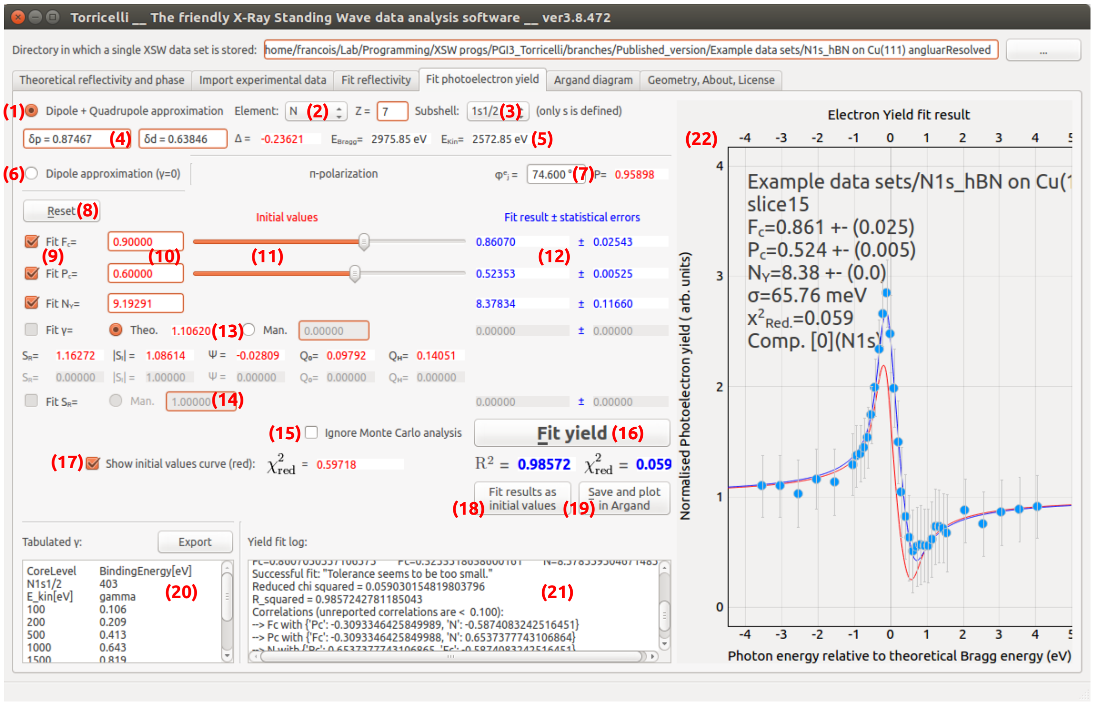
— 5 —

Fit yield

The fit is performed by minimizing

$$\mathcal{R}_Y(\underline{P_\mathrm{c}^{\mathscr{S}}},\underline{F_\mathrm{c}^{\mathscr{S}}},\underline{N_\mathrm{Y}^{\mathscr{S}}}) = \Bigg[Y_{\mathscr{S}}^{\mathrm{model}}\left(h\nu,\underline{P_\mathrm{c}^{\mathscr{S}}},\underline{F_\mathrm{c}^{\mathscr{S}}},\underline{N_\mathrm{Y}^{\mathscr{S}}},\gamma,\phi\right) - \frac{Y_{\mathscr{S}}^{\mathrm{exp}}(h\nu+\delta h\nu,\phi)}{\underline{N_\mathrm{Y}^{\mathscr{S}}}}\Bigg] \times \frac{\underline{N_\mathrm{Y}^{\mathscr{S}}}}{\sigma_{Y_{\mathscr{S}}^{\mathrm{exp}}}(h\nu,\phi)} \tag{5.1}$$

using the Levenberg-Marquardt method. The values of $\delta h\nu$ and σ are the result of the reflectivity fit and are fixed to fit the yield. In the simplest case, there



Directory in which a single XSW data set is stored: home/francois/Lab/Programming/XSW progs/PGI3_Torricelli/branches/Published_version/Example data sets/N1s_hBN on Cu(111) angluarResolved

● Dipole + Quadrupole approximation Element: N Z = 7 Subshell: 1s1/2 (only s is defined)

δp = 0.87467 δd = 0.63846 Δ = -0.23621 E_{Bragg}= 2975.85 eV E_{Kin}= 2572.85 eV

○ Dipole approximation (γ=0) π-polarization φ^{e}_{j} = 74.600 ° P= 0.95898

☑ Fit F_c= 0.90000 — 0.86070 ± 0.02543

☑ Fit P_c= 0.60000 — 0.52353 ± 0.00525

☑ Fit N_Y= 9.19291 — 8.37834 ± 0.11660

☐ Fit γ= ● Theo. 1.10620 ○ Man. 0.00000 — 0.00000 ± 0.00000

S_R= 1.16272 $|S_I|$ = 1.08614 Ψ = -0.02809 Q_0= 0.09792 Q_H= 0.14051

S_R= 0.00000 $|S_I|$ = 1.00000 Ψ = 0.00000 Q_0= 0.00000 Q_H= 0.00000

☐ Fit S_R= ○ Man. 1.00000 — 0.00000 ± 0.00000

☐ Ignore Monte Carlo analysis

☑ Show initial values curve (red): χ^2_{red} = 0.59718

R^2 = 0.98572 χ^2_{red} = 0.059

Tabulated γ:

```
CoreLevel     BindingEnergy[eV]
N1s1/2        403
E_kin[eV]     gamma
100           0.106
200           0.209
500           0.413
1000          0.643
1500          0.819
```

Yield fit log:

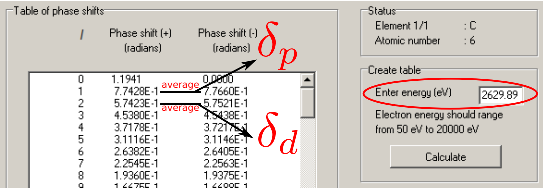
```
Successful fit: "Tolerance seems to be too small."
Reduced chi squared = 0.059030154819803796
R_squared = 0.9857242781185043
Correlations (unreported correlations are <  0.100):
--> Fc with {'Pc': -0.3093346425849989, 'N': -0.5874083242516451}
--> Pc with {'Fc': -0.3093346425849988, 'N': 0.6537377743106864}
```

Figure 5.1: Screenshot of the *Fit yield* tab of Torricelli.

are only three fitting parameters: the normalization factor $\underline{N_{\mathrm{Y}}^{\mathscr{S}}}$, the coherent position $\underline{P_{\mathrm{c}}}$ and fraction $\underline{F_{\mathrm{c}}}$. If the statistical error $\sigma_{\mathrm{Y}^{\exp}_{\mathscr{S}}}(h\nu, \phi)$ are not known or not reliable, it is possible to replace them by $\underline{N_{\mathrm{Y}}^{\mathscr{S}}}$ in Eq. 5.1 by checking Ignore Monte Carlo analysis (15) (see Fig. 5.1). It is possible to independently fix/fit each of the fit variables by un/checking them (9).

The level of approximation to be used to treat the photoemission process can be chosen: dipole (6) or dipole-quadrupole (1). Within the dipole-quadrupole approximation, Δ and the non-dipolar parameter γ need to be calculated. To do so, the user must provide the element subject to photoemission (2) and the sub-shell under consideration (3). The corresponding photoelectron kinetic energy is then calculated and displayed E_{Kin} (5). This calculation ignores the sample work function. E_{Kin} is used to interpolate the γ (13) value from the database [9, 10], that is displayed in (20). Second, the value of E_{Kin} must then be given in the NIST Electron Elastic-Scattering Cross-Section Database version 3.2 (available on-line) to obtain δp and δd (4). A screenshot of the program `Elastic32` is given in Fig. 5.2. Afterwards, the user must copy the δp and δd in TORRICELLI so that Δ can be calculated. Finally, ϕ_j^{e} (7) is used to calculate the polarization factor P (in case of π-polarization). The value of ϕ_j^{e} is automatically updated if an angle file is given in the import tab, if not this value must be given by hand. Once ξ, ζ, P, γ and Δ are known, TORRICELLI can calculate the photoemission correction parameter to the yield S_R, S_I and ψ. Even within the dipole approximation, the ϕ_j^{e} is needed (in case of π-polarization) to properly calculate S_R, S_I and ψ. All initial values are displayed in red.

Press the Fit yield button (16). If the fit converges, the normalized experimental data together with the fitting theoretical curve are displayed as blue points and line (22), respectively. The normalized statistical error are displayed as vertical red bars. Nonphysical negative or greater than unity $\underline{F_{\mathrm{c}}}$ or $\underline{P_{\mathrm{c}}}$ can still produce a very good fit. To overcome this, one has to set better initial values. After having fitted once, it is possible to display the theoretical curve with initial values in red (17). The initial values can be changed by typing the value (10), or by moving the cursor (11). The red curve in (22) will instantly



l	Phase shift (+) (radians)	Phase shift (-) (radians)
0	1.1941	0.0000
1	7.7428E-1	7.7660E-1
2	5.7423E-1	5.7521E-1
3	4.5380E-1	4.5438E-1
4	3.7178E-1	3.7217E-1
5	3.1116E-1	3.1146E-1
6	2.6382E-1	2.6405E-1
7	2.2545E-1	2.2563E-1
8	1.9360E-1	1.9375E-1

Element 1/1 : C
Atomic number : 6
Enter energy (eV) 2629.89
Electron energy should range from 50 eV to 20000 eV

Figure 5.2: Screenshot of the Elastic32 program, the NIST Electron Elastic-Scattering Cross-Section Database to calculate δp and δd (Menu: Database/Phase shifts). Enter the adequate kinetic energy of the photoemitted electrons as calculated by TORRICELLI, and click Calculate.

update. Press the Fit yield button (16) again once the initial values are set. The fit results together with their standard deviation (12) are displayed, as well as the coefficient of determination R^2 (see Chap. 4) and the χ^2_{red}. Also, each step of the optimization are displayed in (21), as well as further fit results details. All fit result values are displayed in blue. It is possible to fit the $\underline{\gamma}$ (or $\underline{S_R}$) if your sample is completely disordered and you are confident to fix $\overline{F}_{\mathrm{c}}$ to 0.

Once satisfied with the fit results, click on Save and plot in Argand (19) to store the results (including the standard deviations and all other relevant parameters) in a list of all gathered results (See Chap. 6).

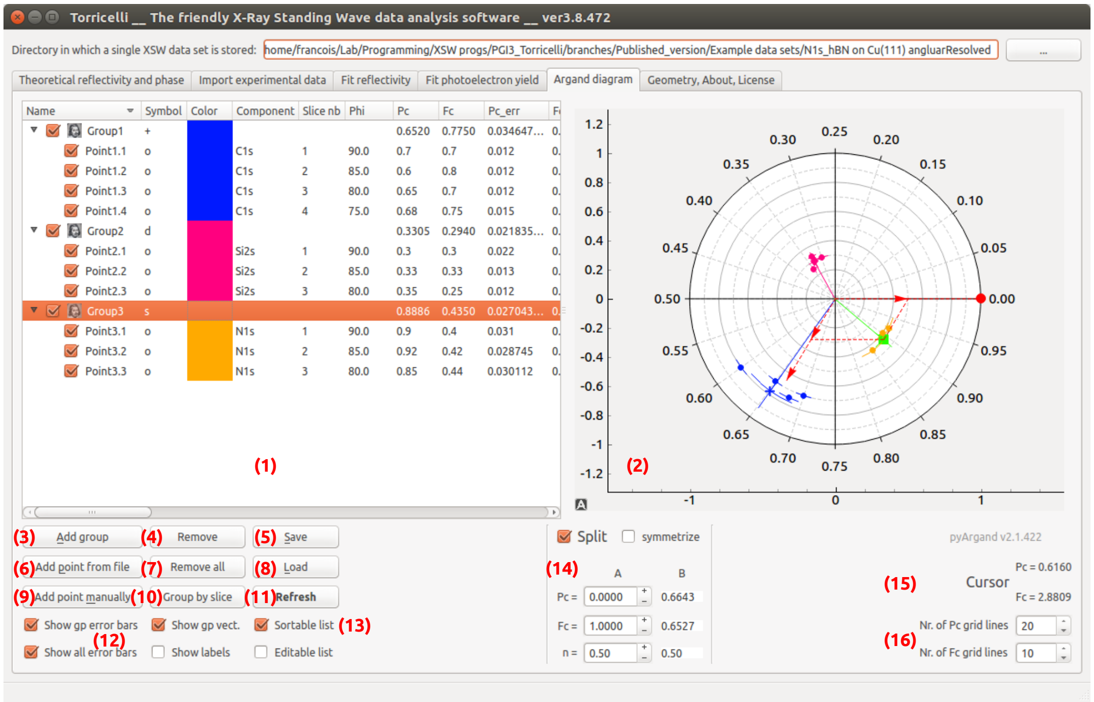
Note The Monte-Carlo analysis in CasaXPS is usually slow. If you are in a hurry (during a beam-time), you may want to avoid this step and tell Torricelli not to read in the error bars and click Ignore Monte Carlo analysis (15). This will remove the error bar weight on the data points, by effectively setting $\sigma_{Y^{\mathrm{exp}}_{\mathscr{S}}}(h\nu, \phi) = \underline{N^{\mathscr{S}}_{Y}}$. As a consequence, the error bars given by the fit procedure are not defined anymore and should not be used. Also, χ^2_{red} is not defined anymore.

Auto-save In the 🗁$\mathcal{D}$▸`results` folder, pictures, normalized experimental data, fitted theoretical data as well as fit results are saved/updated automatically each time the Fit Yield button is pressed. The created `RESULTS_*.csv` file, that summarize the fit results and all used parameters, can be easily loaded by another program to consult the data.

— 6 —

Argand diagram

In this tab it is possible to display the fit results of the electron yield curve on an Argand diagram (2), a polar plot where each point is defined by a vector with angle P_c and length F_c. The data are all listed in a list (1) and can be grouped (see Fig. 6.1). The vector average position and fraction of each group is automatically calculated and displayed (by default a + symbol and a line from the origin). One can display/hide independently every single data point (circle symbol by default, with the color of the group) or entire groups. One can then move the data points from one group to another or within a group by



Directory in which a single XSW data set is stored: home/francois/Lab/Programming/XSW progs/PGI3_Torricelli/branches/Published_version/Example data sets/N1s_hBN on Cu(111) angluarResolved

Name	Symbol	Color	Component	Slice nb	Phi	Pc	Fc	Pc_err	F
▼ ☑ Group1	+	🟦				0.6520	0.7750	0.034647...	0.
☑ Point1.1	o	🟦	C1s	1	90.0	0.7	0.7	0.012	0.
☑ Point1.2	o	🟦	C1s	2	85.0	0.6	0.8	0.012	0.
☑ Point1.3	o	🟦	C1s	3	80.0	0.65	0.7	0.012	0.
☑ Point1.4	o	🟦	C1s	4	75.0	0.68	0.75	0.015	0.
▼ ☑ Group2	d	🟥				0.3305	0.2940	0.021835...	0.
☑ Point2.1	o	🟥	Si2s	1	90.0	0.3	0.3	0.022	0.
☑ Point2.2	o	🟥	Si2s	2	85.0	0.33	0.33	0.013	0.
☑ Point2.3	o	🟥	Si2s	3	80.0	0.35	0.25	0.012	0.
▼ ☑ Group3	s					0.8886	0.4350	0.027043...	0.
☑ Point3.1	o	🟧	N1s	1	90.0	0.9	0.4	0.031	0.
☑ Point3.2	o	🟧	N1s	2	85.0	0.92	0.42	0.028745	0.
☑ Point3.3	o	🟧	N1s	3	80.0	0.85	0.44	0.030112	0.

☑ Show gp error bars ☑ Show gp vect. ☑ Sortable list

☑ Show all error bars ☐ Show labels ☐ Editable list

☑ Split ☐ symmetrize

	A	B
Pc =	0.0000	0.6643
Fc =	1.0000	0.6527
n =	0.50	0.50

Cursor Pc = 0.6160 Fc = 2.8809

Nr. of Pc grid lines 20

Nr. of Fc grid lines 10

Figure 6.1: Screenshot of the *Argand diagram* tab of TORRICELLI.

a simple drag and drop. The selected point/s is/are highlighted in (2). Press Esc to clear the selection. Points and groups can be removed by pressing Del on the keyboard, or clicking the Remove button in TORRICELLI. Double-click or F2 on any value-field permits to amend the values (only possible is Editable list is selected). If several points are selected, a right-click on the name, color or symbol permit to apply the new property on the whole selection.

To add a group, just click on Add group (3). To insert values in the selected group, one can either type-in the values manually (9), or choose the log file of a fit (6), or by clicking the Save and plot in Argand button after having obtained a satisfying fit ((19) in the previous chapter). In the latter case, a new point is created in a group named like the working directory name. Furthermore, all relevant values are also saved and displayed in the list (1). The content of the list can be saved (5) in a text `.csv` file and reloaded at a later point (8). The format is chosen so that it can be easily loaded by another program (see Chap. 10 for the content details).

One can also decompose any data point $\mathscr{D}$ in the sum of two vectors ($\mathscr{A}$ and $\mathscr{B}$) (13). The $\mathscr{A}$ vector can be modified by moving it with the mouse (big red circle in (2)), or by typing values. The $\mathscr{B}$ will be updated automatically. n corresponds to the respective amount of atoms populating each species. The *Symmetrize* option forces $\mathscr{A}$ and $\mathscr{B}$ to have the same fraction.

By right-clicking in the diagram, one can export the picture by choosing 'Entire scene'. Several formats are possible, we advise though the `.svg` vector format which allows the easy modification of the picture with a dedicated program, like INKSCAPE.

Auto-save Additionally, the full content of the Argand diagram is automatically saved regularly in the 🗋$\mathcal{D}$▸`autosave_time_description.csv` file in the working directory when the whole list is cleared, when the list is grouped by slices, after inserting a new point, or when TORRICELLI is closed. In each case, an explicit name is used.

The 🗋$\mathcal{D}$▸`results`▸`RESULTS_DataName_TorricelliVersion.csv` file is created/updated each time the fit yield button is pressed in a given working directory. It summarizes all fit results obtained from a single data set for each $\mathscr{S}$ and each slice. If the fit button is pressed several times for the same $\mathscr{S}$ and slice, only the last fit results are saved.

— 7 —

Example data sets

In the directory 📂▸`Examples data sets`, two data sets are given. In the following section, we give the fit results produced by TORRICELLI for a specific set of parameters.

7.1 C1s_H-QFMLG on SiC(0001)

The first is a C 1s core level on quasifreestanding monolayer graphene on 6H-SiC(0001), using the reflection (006). The data are angle integrated. Using π-polarization, $\zeta = 0$, $\xi = 3.5°$, the Zywietz DW method, the core level component C1s_QFMLG ($i = 1$), $\phi_j^e = 90°$ and the dipole approximation, the reflectivity fit result gives $\sigma = 45.7$ meV, $N_R = 14396$, $R_0 = 154$ and $\delta h\nu = 2.67$ eV. The yield fit result provides $F_\mathrm{c} = 0.871$, $P_\mathrm{c} = 0.688$, $N_Y = 128891.17$.

7.2 N1s_hBN on Cu(111) angluarResolved

The second is a N 1s of hexagonal boron nitride monolayer grown on Cu(111) using the (111) reflection. The data are angle-resolved. Using π-polarization, $\zeta = 0$, $\xi = 3.5°$, the Gao DW method, the core level component N1s ($i = 0$), the slice $j = 15$, $\phi_j^e = 74.6°$ and the dipole-quadrupole approximation ($\delta p = 0.87467$, $\delta d = 0.63846$, $\gamma = 1.10620$), the reflectivity fit result gives $\sigma = 65.8$ meV, $N_R = 6950$, $R_0 = -51$ and $\delta h\nu = 3.517$ eV. The yield fit result provides $F_\mathrm{c} = 0.861$, $P_\mathrm{c} = 0.524$, $N_Y = 8.38$.

— 8 —

Useful shortcuts

- General ones:

  Alt+**F** Presses the big button you always want to press on the active tab

  Ctrl+**PageDown** Moves to the next tab

  Ctrl+**PageUp** Moves to the previous tab

  Mouse Wheel on tab Scroll through tabs

- In graphical sub-windows:

  Mouse wheel zoom in/out

  Left click + mouse move x/y translation

  Right click + mouse move x/y re-scaling

  Right click menu with more options, including export options

- In the Argand list of group and points:

  Double click on value change the value in the selected field (Right-click is several items are selected)

  F2 **on value** change the value in the selected field

  Double click on name change the name. If several points are selected, a right-click will rename the selection with with increasing N: `name_N`

  Esc clear the item selection

— 9 —

The folder structure of TORRICELLI

In the following we describe briefly all files present in the TORRICELLI program folder tree.

-  .... TORRICELLI home folder
  - `Torricelli.py` .... execute this file to start TORRICELLI
  - `COPYRIGHT` .... copy of the GNU General Public License v3
  - `README` .... Brief description of the files and folders
  - `Example data sets`
    - `ArgandTest.csv` .... few (F_c,P_c) points for the Argand diagram
    - `Argand_BigFileTest.csv` .. many (F_c,P_c) points for the Argand diagram
    - `C1s_H-QFMLG on SiC(0001)`
    - `N1s_hBN on Cu(111) angluarResolved`
  - `imports`
    - `user_settings` .... last settings used before closing TORRICELLI
    - `GUI_⋆` .... gui modules
    - `⋆.png and ⋆.svg` .... images
    - `pyArgand.py` .... Argand diagram library
    - `Databases`
      - `Database references.txt` ..... references for all files in the database
      - `Nondipolar_parameters_of_angular_distribution_Z1to100.ini`
      - `f0.csv` .... f0 atomic scattering factor for all elements
      - `f1 and f2` .... all f1 and f2 atomic scattering factor
        - `si.nff` .... f1 and f2 for Si
      - `DW` .... database necessary for the Debye-Waller factors
      - `Lattices`
        - `AtomCoordinates_name-of-unit-cell.csv` .. relative atomic position in this unit cell
        - `CrystallographicData_Compound.csv` crystallographic parameters for compounds
        - `CrystallographicData_Elemental.csv` crystallographic parameters for elemental crystals

— 10 —

The folder structure of an analyzed data set

Example of directory in which a single XSW data set is stored. In the folder 📂▸sum (depicted as 🗁$\mathcal{D}$▸), here chosen as working directory, the folder 🗁$\mathcal{D}$▸results will be created by TORRICELLI during the analysis.

- 📂▾C1s_H-QFMLG on SiC(0001) *or* 🗁$\mathcal{D}$▸
  - 🗎▸i09-22758_C1sXSW_Relf_and_I0.refl … reflectivity, TORRICELLI *input*
  - 🗎▸i09-22758_c1sxsw_allregions_2Comp_ShirleyBgd.txt electron yield (CasaXPS analysis output), TORRICELLI *input*
  - 🗎▸autosave_time_description.csv ….. Content of the Argand list automatically saved after pressing the `Save and Plot in Argand` button
  - ■▸slice1 ……………… all data from the first angle-resolved data set
  - ■▸slicen ……………… all data from the nth angle-resolved data set
  - 📂▾sum …………… all data from the sum of all angle-resolved data sets
    - 🗎▸i09-22758_C1sXSW_allRegions_**hnu**.dat . core level measured with a photon energy of **hnu** (CasaXPS input)
    - 🗎▸i09-22758_c1sxsw_allregions_2Comp_ShirleyBgd.vms CasaXPS analysis file
  - 📂▾results
    - 🗎▸Structure Factor.dat ……………… all structure factors
    - 🗎▸Exp_ey_norm_centred[0].dat …. electron yield of component 0 normalised to I_0 and relative to E_{Bragg}
    - 🗎▸Exp_refl_norm_centred.dat … reflectivity normalised to I_0 and relative to E_{Bragg}
    - 🗎▸Fit_ey_comp[0].dat … best fit electron yield curve, component 0
    - 🗎▸Fit_ey_comp[0].log ………… log of the electron yield fitting
    - 🗎▸Fit_ey_comp[0].png ……… image from electron yield fit result
    - 🗎▸Fit_refl.log …………………… log of the reflectivity fitting
    - 🗎▸Fit_result_refl.dat ……………… best fit reflectivity curve
    - 🗎▸Fit_result_refl.png ………. image from reflectivity fit result
    - 🗎▸RESULTS_C1s_H-QFMLG on SiC(0001)_Torricelli_verx.x.x.csv list of results obtained from this working directory, updated after each click on `Fit yield`

`Theoretical values.dat` .... theoretical curves
`Theoretical values.png` ...... image of the theoretical curves

Few `.png` files simply record the graph displayed in TORRICELLI, for a quick printout. The `.dat` files include actual data in ASCII that can be used in a scientific plotting program to produce high quality figures. The exact content of each of the `.dat` is now detailed (for programmers, we also included the respective variable names.)

1. **`Theoretical values.dat`**

   col1 Photon energy relative to the Bragg energy (`self.Theory_photonEnergy`)

   col2 Reflectivity of the sample crystal (`self.Theory_Refl_sample`)

   col3 Phase of the sample crystal (`self.Theory_Phase_Sample`)

   col4 Reflectivity of the monochromator crystal (`self.Theory_Refl_Monochromator`)

   col5 Phase of the monochromator crystal (`self.Theory_Phase_Monochromator`)

   col6 Final convoluted reflectivity of both crystal and monochromator (`self.Theory_ReflSample_cc_ReflMono2`)

2. **`Exp_refl_norm_centred.dat`**

   col1 Photon energy shifted by $\delta h\nu$ to fit the theoretical values (`self.Exp_photonEnergy_BraggCentered` $+\delta h\nu$)

   col2 Reflectivity normalized by I_0^{exp} for each photon energy, and multiplied by the average of all I_0^{exp}, then shifted by R_0 and multiplied by N_R to fit the theory ((`self.Exp_Refl_Normalised` $- R_0)/N_R$)

   col3 Estimated statistical error on the reflectivity (square root of the previously normalized intensity) (`self.Exp_Refl_Estimated_Error`$/N_R$)

3. **`Fit_result_refl.dat`**

   col1 Photon energy relative to the Bragg energy (`self.Theory_photonEnergy`)

   col2 Theoretical reflectivity curve best fit, including the gaussian broadening (`bestFit_theo_refl`)

4. **`Fit_refl.log`** : List of all tested parameters configurations tested during the fitting procedure. The last one corresponds to the best fit.

5. **`Exp_ey_norm_centred[`$\mathscr{S}$`].dat`**

   col1 Photon energy shifted by $\delta h\nu$ (`self.xsw_energies_BraggCentered` $+ \delta h\nu$)

   col2 Electron yield of the component set $\mathscr{S}$ provided (for example) by CasaXPS. It is normalized by I_0^{exp} for each photon energy, and multiplied by the average of all I_0^{exp} and divided by the $N_Y^{\mathscr{S}}$ (`self.xsw_ey_normalised`$/N_Y^{\mathscr{S}}$)

col3 Statistical error of the electron yield of the component set $\mathscr{S}$ provided by CasaXPS normalized in the same way as the electron yield itself (`self.xsw_ey_error_casaXPS`/$N_Y^{\mathscr{S}}$) (NOTE that I09_DataBrowser does not perform any normalisation)

6. 🗎‣**`Fit_ey_comp[`$\mathscr{S}$`].dat`** :

   col1 Photon energy relative to the Bragg energy (`self.Theory_photonEnergy`)

   col2 Electron yield best fit (`self.Fit_Result_EY`)

7. 🗎‣**`Fit_ey_comp[`$\mathscr{S}$`].log`** : List of all tested parameters configurations tested during the fitting procedure. The last one corresponds to the best fit. Each different value of the initial curve that is displayed will also be saved in this file.

8. **`*.csv` files**: Contains the fit results parameters separated by a comma. Each line correspond to a different data set. In the case of `.csv` file created by the Argand diagram tab, there is also a line corresponding to the average of all points contained in the group. The 🗎‣`autosave.csv` is rewritten each time the Save and plot in Argand button is clicked and contains full content of the list visible in the Argand tab. The 🗎‣`RESULT_*.csv` file is updated each time you click the Fit yield button, and contains only fit results from the data present in the working directory. It remembers only the fit result of resulting from the last time you clicked Fit yield for each set of component or angle chosen.

— 11 —

Miscellaneous

11.1 Reading or modifying the code

Reading or modifying a function that is related to a specific object (Button, integer field, etc.) in the graphical user interface works as follows. One opens the `GUI_MainWindow.py` file with the `QtDesigner` free software[1] to obtain the object name. Then one can search the `Torricelli.py` file using a standard text editor to find which function is connected to this object. Both the command connecting an object to a function and the function itself are in the `Torricelli.py` file.

11.2 Few advices for efficiency

- Use the shortcuts! Once you chose the working directory and set all your parameters/angles, you basically just have to press Alt+F and Ctrl+PageDown few times... And you are done!
- Use the folder structure created by the DataBrowser (as explained in Chap. 10).
- Save the electron yield file in the folder created by the DataBrowser, and use either a .txt or .ey extension.
- Select the new folder for each new measurement (top of the window), then TORRICELLI will find all it needs automatically.
- It is a good habit to keep an eye on the console window. Typically unexpected warnings or errors are displayed there.
- It is an open source software: it can be modified but the modifications must maintain the same license.

Note Many file paths, folder paths and other settings are saved in the file 🗁‣`imports`‣`user_settings`. So when you close TORRICELLI, and open it again, the program is already pre-configured for you.

---

[1] https://doc.qt.io/archives/qt-4.8/designer-manual.html

Bibliography

[1] F. C. Bocquet, G. Mercurio, M. Franke, G. van Staaten, S. Weiß, S. Soubatch, C. Kumpf and F.S. Tautz – TORRICELLI*: A software to determine atomic spatial distributions from normal incidence x-ray standing wave data.*
Computer Physics Communications **vol**, page (2018).

[2] Mercurio, G. and Bauer, O. and Willenbockel, M. and Fairley, N. and Reckien, W. and Schmitz, C. H. and Fiedler, B. and Soubatch, S. and Bredow, T. and Sokolowski, M. and Tautz, F. S. – *Adsorption height determination of nonequivalent C and O species of PTCDA on Ag(110) using x-ray standing waves.*
Phys. Rev. B **87**, 045421 (2013).

[3] K. Levenberg – *A Method for the Solution of Certain Non-Linear Problems in Least Squares.*
Quart. Appl. Math **2**, 164 (1944).

[4] D. W. Marquardt – *An Algorithm for Least-Squares Estimation of Nonlinear Parameters.*
J. Soc. Indust. Appl. Math. **11**, 431 (1963).

[5] M. Newville, T. Stensitzki, D. B. Allen and A. Ingargiola – LMFIT*: Non-Linear Least-Square Minimization and Curve-Fitting for Python.*
Zenodo, (2014).

[6] A. Glantz and B. K. Slinker – *Primer of applied regression and analysis of variance.*
2nd edition, McGraw-Hill (2001).

[7] P. E. Greenwood and M. S. Nikulin – *A guide to chi-squared testing.*
John Wiley & Sons (1996).

[8] N. R. Draper and H. Smith – *Applied Regression Analysis.*
Wiley Series in Probability and Statistics, 3rd edition, Wiley-Interscience (1998).

[9] M. B. Trzhaskovskaya, V. I. Nefedov, V. G. Yarzhemsky – *Photoelectron angular distribution parameters for elements Z=1 to Z=54 in the photoelectron energy range 100-5000 eV.*
Atom. Data Nucl. Data Tables **77**, 97 (2001).

[10] M. B. Trzhaskovskaya, V. I. Nefedov, V. G. Yarzhemsky – *Photoelectron angular distribution parameters for elements Z=55 to Z=100 in the photoelectron energy range 100-5000 eV.*
Atom. Data Nucl. Data Tables **82**, 257 (2002).