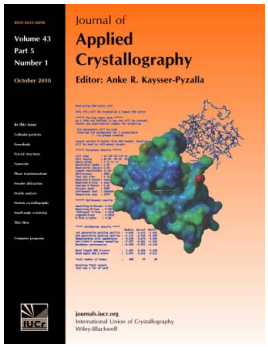
Journal of
Applied Crystallography
ISSN 0021-8898
Editor: **Anke R. Kaysser-Pyzalla**

Fityk: a general-purpose peak fitting program

Marcin Wojdyr

J. Appl. Cryst. (2010). **43**, 1126–1128



Journal of
Applied
Crystallography

ISSN 0021-8898

Received 22 June 2010
Accepted 30 July 2010

Fityk: a general-purpose peak fitting program

Marcin Wojdyr

Institute of High Pressure Physics, ulica Sokolowska 29/37, 01-142 Warsaw, Poland. Correspondence e-mail: wojdyr@gmail.com

Fityk is portable, open-source software for nonlinear curve fitting and data analysis. It specializes in fitting a sum of bell-shaped functions to experimental data. In particular, it enables Pawley refinement of powder diffraction data and size–strain analysis.

1. Introduction

Popular methods of powder pattern analysis, such as Rietveld (1969), Pawley (1981) and Le Bail (Le Bail *et al.*, 1988) whole profile refinement, are optimization procedures that serve for derivation of a set of parameters that provide the best fit of a mathematical function to the diffraction data. Similar refinement is used in many other experimental techniques. Usually a mathematical model is fitted to the data by minimizing the weighted sum of squared residuals, using one of the well known iterative algorithms, such as the Levenberg–Marquardt method.

Fityk can analyze different types of experimental data providing the possibility of quantifying selected parameters. This paper focuses on using the software for analysis of powder diffraction data. Besides powder diffraction, *Fityk* is used in chromatography, photoluminescence and photoelectron spectroscopy, infrared and Raman spectroscopy, and in other techniques. Using a more generic program provides greater flexibility. While this is not an advantage in routine analysis, the flexibility is important when one wants to modify the refinement procedure or introduce a custom data analysis scheme.

The underlying idea of *Fityk* is to let the user build a model (a function that is to be fitted to the data) from components such as bell-shaped functions (*e.g.* Pearson VII or Voigt), polynomials and other types of mathematical functions. The model may contain a so-called x correction, the function that describes errors on the measured x values. The parameters of the component functions can be constrained by any equality constraints, *e.g.* the positions of peaks in the diffraction pattern can be bound with the radiation wavelength and lattice parameters. When the model is prepared, it can be refined using one of the three fitting methods implemented in the program.

The software has its own mini-language, used to perform all operations, but also comes with a graphical interface that allows users to perform the most common tasks without learning the language. In particular, the interface simplifies the Pawley refinement of powder diffraction data.

2. Program description

2.1. Architecture

Fityk has two user interfaces: a graphical one and a text-based one. Both are built on top of the libfityk library (Fig. 1).

The graphical user interface (GUI) (Fig. 2) is written using the cross-platform wxWidgets library. The user interacts with the GUI in the way typical for modern computer programs and the interface translates the user's actions to text – a sequence of *Fityk* commands that are executed by the engine in the libfityk library. Recently, we developed an extension to the GUI, called the *XRPD add-on*, which allows powder pattern analysis to be performed. This add-on is described in §3.2.

The command line interface (CLI) provides light-weight access to the software functions. The CLI is useful for running scripts and performing automated tasks.

Reading of experimental data files is handled by the xylib library. The library can read 14 file formats, including nine formats for

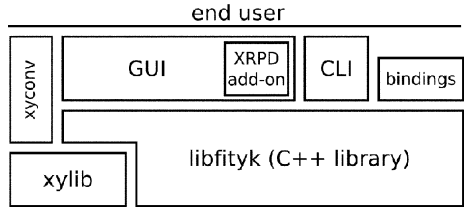


Figure 1
Software architecture. The three programs [*fityk* (GUI), *cfityk* (CLI) and *xyconv*] are based on the libfityk and xylib libraries. All the components are described in the text.

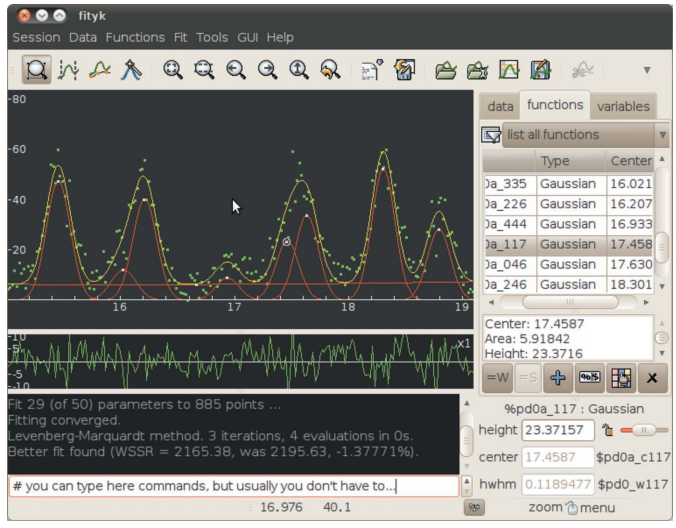


Figure 2
A screenshot of the graphical user interface. The main panel shows data (green), fitted model (yellow) and component Gaussian functions (red). The auxiliary panel below shows the difference between data and model. The text input at the bottom of the window allows use of the *Fityk* mini-language commands. The parameters of the component functions can be inspected and changed in the sidebar on the right-hand side.

powder diffraction data. The code for data file reading has been put into a separate library to allow its use in other programs. The library is distributed with a small program, *xyconv*, which converts supported formats to plain text.

The libfityk library can process the data, construct the mathematical model and optimize the model. All the operations are managed using commands of the *Fityk* mini-language, which is a domain-specific language, tailored toward data processing and model composition. Having a textual representation of all the actions performed by the user allows the actions to be logged to a file. It also makes it possible to automate common tasks with scripts.

The libfityk library is written in the C++ language and can be used natively in C++ programs. Using it from other programming languages requires bindings to the library. We provide bindings to the Python and Lua languages, and bindings to other programming languages, including Java and Ruby, can be easily generated.

2.2. Features

The data processing is based on parsing and executing mathematical expressions. For example, the command `x = 4*pi*sin(x/2*pi/180) / 1.5405` transforms the x coordinate of the data points according to the specified formula, converting 2θ data to the Q space.

A few data operations have dedicated support. For example, it is possible to manually subtract the data background in the GUI with the mouse. When the user clicks the 'Strip background' button to finish the operation, the GUI passes a command, *e.g.* `y = y - Spline(1.7,7.4, 8.8,4.5, 37.9,3.1)`, to the engine.

The mathematical model F that is to be fitted to the data is defined as

$$F(x; \mathbf{a}) = \sum_i f_i[x - z(x; \mathbf{a}); \mathbf{a}], \tag{1}$$

where $\mathbf{a}$ is the vector of parameters that are to be refined, f_i are component functions and z is the so-called x correction. The component functions have either built-in or user-defined type. The built-in types include functions commonly used for modeling of diffraction peaks: Gaussian, Lorentzian, Voigt, pseudo-Voigt, Pearson VII and the bifurcated versions of these functions. An example of user-defined type is given in §3.1. The x correction is used to model errors of the measured x values, such as errors caused by instrumental zero shift and sample displacement.

To make construction of the model easier, libfityk provides a simple peak detection algorithm. The user chooses the type and the approximate position of the peak that is to be added to the model, and the program determines the initial parameters of the peak function.

Once the model is defined, the next step is to apply one of the well known optimization algorithms. Two alternatives to the popular Levenberg–Marquardt method (L–M) were investigated: the Nelder–Mead downhill simplex method (N–M) and genetic algorithms (GA). They both search for the minimum without using the gradient of a merit function; this makes them much slower than the L–M method, but they may avoid (especially GA) a local minimum and find a global one. Although in our studies neither N–M nor GA were superior to L–M, the user can still switch between these methods.

It is also worth noting that *Fityk* allows the user to work with multiple data sets and can thus be used to process data series or to simultaneously fit patterns from the same sample obtained in different experiments.

```
$lambda = 1.54051 # definition of constant (Cu Ka1)
$a = ~4.36 # definition of variable (lattice parameter)

# add peak (111), $c111 is a peak position variable
$c111 = 360/pi * asin(sqrt(1^2+1^2+1^2) / $a / 2 * $lambda)
%p111 = guess Voigt center=$c111

# peak hkl=(220)
$c220 = 360/pi * asin(sqrt(2^2+2^2+0) / $a / 2 * $lambda)
%sic220 = guess Voigt center=$c220

fit  # fit the data
```

Figure 3
A minimal example of the Pawley method in a *Fityk* script. The first lines set up the model consisting of two Voigt peaks with positions constrained by the wavelength and lattice parameter. In the last line, the model is fitted to the data.

3. Usage examples

3.1. Custom peak shape

If the built-in functions are not sufficient, the user can define a new function by providing its analytical form. A good example of the usefulness of this feature for powder diffraction is the implementation of an analytical function derived by Pielaszek and co-workers (Pielaszek, 2003; Wejrzanowski *et al.*, 2006). This function describes a physical line profile under the assumptions that there is no strain, grains are spherical and the grain sizes obey the Gamma distribution, which can be regarded as an approximation of the lognormal distribution usually assumed for the grain sizes. The derived line profile has the following formula:

$$\begin{aligned}\left\langle Lp(q; \langle R\rangle, \sigma)\right\rangle &= \frac{\langle R\rangle^3}{(2\pi)^{1/2} q^4\sigma^6\left(1+q^2\sigma^4/\langle R\rangle^2\right)^{\langle R\rangle^2/2\sigma^2}} \\ &\times \frac{1}{\left(\langle R\rangle^2/\sigma^2-1\right)\left(\langle R\rangle^2/\sigma^2-2\right)\left(\langle R\rangle^2/\sigma^2-3\right)} \\ &\times \left\{3\left(1+\frac{q^2\sigma^4}{\langle R\rangle^2}\right)^{\langle R\rangle^2/2\sigma^2}\left[2+\frac{q^2\sigma^4}{\langle R\rangle^2}\left(\frac{\langle R\rangle^2}{\sigma^2}-2\right)\left(\frac{\langle R\rangle^2}{\sigma^2}-3\right)\right]\right. \\ &- 6\left(1+\frac{q^2\sigma^4}{\langle R\rangle^2}\right)^{3/2}\cos\left[\left(\frac{\langle R\rangle^2}{\sigma^2}-3\right)\arctan\left(\frac{q\sigma^2}{\langle R\rangle}\right)\right] \\ &\left.- 6\frac{q\sigma^2}{\langle R\rangle}\left(1+\frac{q^2\sigma^4}{\langle R\rangle^2}\right)\left(\frac{\langle R\rangle^2}{\sigma^2}-3\right)\sin\left[\left(\frac{\langle R\rangle^2}{\sigma^2}-2\right)\arctan\left(\frac{q\sigma^2}{\langle R\rangle}\right)\right]\right\},\end{aligned} \tag{2}$$

where q is the scattering vector, $\langle R\rangle$ is the average grain size and σ is the standard deviation of the grain size.

In *Fityk*, we define this function type with the command `define Pielaszek(h, center, r, s) = h * r^3 / (q^4*s^6 * (1 + q^2*s^4/r^2)^(r^2/(2*s^2)) * (r^2/s^2 - 1) * (r^2/s^2 - 2) * (r^2/s^2 - 3)) * (3 * (1 + q^2*s^4/r^2)^(r^2/(2*s^2)) * (2 + q^2*s^4/r^2 * (r^2/s^2 - 2) * (r^2/s^2 - 3)) - 6 * (1 + q^2*s^4/r^2)^1.5 * \cos((r^2/s^2 - 3) * atan(q*s^2/r)) - 6 * q*s^2/r*(1 + q^2*s^4/r^2) * (r^2/s^2 - 3) * \sin((r^2/s^2 - 2) * atan(q*s^2/r))) where q = x - center`. Fitting this function to a diffraction line readily gives the parameters $\langle R\rangle$ and σ.

3.2. Pawley refinement

Fig. 3 shows a small *Fityk* script that uses two Voigt functions to model the (111) and (220) lines in the diffraction pattern. The positions of the diffraction lines are given as functions of the cubic SiC lattice parameter and radiation wavelength. The tilde before the lattice parameter value, in the second line, indicates that this value is to be refined. Other parameters of the Voigt functions are not constrained and are also refined.

Since setting up a complex model by manually writing a script like this would be tedious, we wrote the *XRPD add-on* (Fig. 4), which allows the model to be prepared by filling in information about radiation, corrections, phases present in the sample, the function chosen to model reflections and constraints on the parameters. The positions of reflections are constrained by lattice parameters and wavelengths. The peak breadth and peak shape parameters can depend on the diffraction angle according to commonly used formulas. Intensities of each reflection are refined independently. Optionally, if the user provides positions of atoms in the unit cell, the initial intensities are calculated using the structural information.

The part of the code that handles space groups is based on the Computational Crystallography Toolbox (CCTBX; Grosse-Kunstleve *et al.*, 2002). Because of the large size of the CCTBX library, we do not use it directly in our program. Instead, we wrote a small program to export data from CCTBX and we embedded the exported data in our code.

4. History and outlook

Fityk has been publicly available since 2004. It was originally developed to analyze powder diffraction patterns with highly overlapping peaks (Wojdyr *et al.*, 2005), using the Pawley method, and then evolved toward a more general tool. Our survey of papers published in 2009 has shown that only a few out of 28 articles that mention using *Fityk* are related to powder diffraction.

In 2010 we started developing the *XRPD add-on*. In the latest version of the program (0.9.3), the add-on can perform Pawley whole pattern refinement. The current work is focused on microstructure (size–strain) analysis.

5. Distribution

Fityk is distributed under the terms of the GNU General Public License, version 2. Source code (in the C++ language) and Windows and Linux binaries are available at the *Fityk* web site (http://www.unipress.waw.pl/fityk/). The web site also contains documentation and a link to unofficial Mac OS X binaries. For most of the popular Linux distributions (including Debian, Ubuntu and Fedora) and for FreeBSD the program is readily available from their respective package repositories.

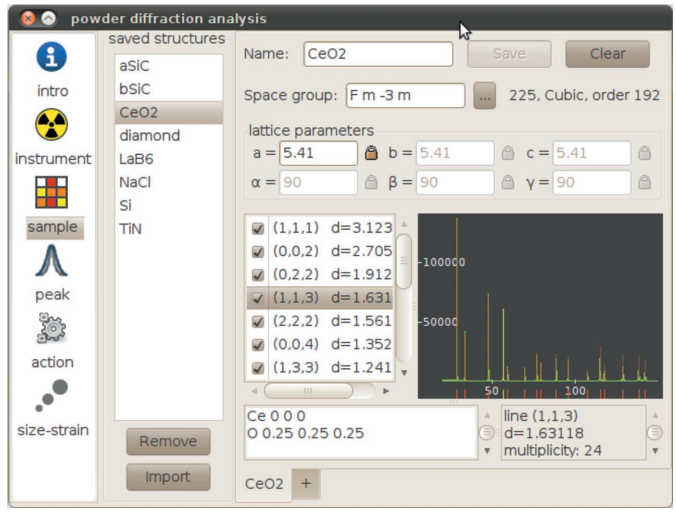


Figure 4
A dialog that is a part of the *XRPD add-on*. It prepares a model for a powder pattern. The buttons with padlock icons next to the lattice parameters are used to indicate whether the parameter should be refined or not.

The author thanks Bogdan Palosz and Stanislaw Gierlotka for helpful discussions. During the six years of development, the software has been improved by contributions and suggestions from many users. In particular, the work of Peng Zhang on the xylib library is acknowledged.

References

Grosse-Kunstleve, R. W., Sauter, N. K., Moriarty, N. W. & Adams, P. D. (2002). *J. Appl. Cryst.* **35**, 126–136.
Le Bail, A., Duroy, H. & Fourquet, J. L. (1988). *Mater. Res. Bull.* **23**, 447–452.
Pawley, G. S. (1981). *J. Appl. Cryst.* **14**, 357–361.
Pielaszek, R. (2003). PhD thesis, Faculty of Physics, Warsaw University, Poland.
Rietveld, H. M. (1969). *J. Appl. Cryst.* **2**, 65–71.
Wejrzanowski, T., Pielaszek, R., Opalińska, A., Matysiak, H., Łojkowski, W. & Kurzydłowski, K. J. (2006). *Appl. Surf. Sci.* **253**, 204–208.
Wojdyr, M., Gierlotka, S., Ivanisenko, Y., Lojkowski, W. & Fecht, H. (2005). *Solid State Phenom.* **101–102**, 97–100.