



Jurnal Teknologi Reaktor Nuklir

Tri Dasa Mega

Journal homepage: jurnal.batan.go.id/index.php/tridam

Development of Automatic Data Processing for BATAN's HRPD and FCD/TD Using *Python* Code

Muzakkiy Putra M. Akhir, Rina Kamila*

Center for Science and Technology of Advanced Material (PSTBM) – BATAN, Kawasan Nuklir Serpong, Tangsel, 15314, Indonesia

ARTICLE INFO

Article history:

Received: 27 August 2020

Received in revised form: 24 September 2020

Accepted: 25 September 2020

Keywords:

HRPD

FCD/TD

Automatic Data Processing

Neutron Diffraction

Python

ABSTRACT

High Resolution Powder Diffractometer (HRPD) and Four Circle Diffractometer/Texture Diffractometer (FCD/TD) are two BATAN-owned neutron diffractometers which have been fully operational since 1992. These are used to investigate structure and texture of crystalline materials, respectively. Before analyzing, the acquired raw neutron diffraction data should first be processed in a specific way to achieve the suitable data format required by the analysis software. This data processing step is a repetitive task for every single experiment which is previously done manually and very time-consuming. The purpose of this development project was to optimize this step to be fully automatic and executable by a code. This work was performed by means of *Python* code utilizing the array manipulation in re-arranging and re-formatting the raw data. The resulted *Python* codes were named as *hrpd.py* and *fcdd.py*. These have been successfully done and validated, making data processing step easier, simpler, and significantly faster with only 20 seconds or less required.

© 2020 Tri Dasa Mega. All rights reserved.

1. INTRODUCTION

Neutron diffractometers are rarely available in the world. BATAN is one of few institution which possess this sophisticated instrument. Three neutron diffractometers are fully functional in BATAN under the Neutron Scattering Laboratory (NSL). Two of them are High Resolution Powder Diffractometer (HRPD), denoted as DN3, and Four Circle Diffractometer/Texture Diffractometer (FCD/TD), denoted as DN2. Both instruments use the same neutron source which originated from GA-Siwabessy multipurpose reactor (RSG-GAS) through S5 radial neutron guide tube. RSG-GAS is the only research reactor in Southeast Asia that was designed to be able to operate at maximum power of 30 MW with neutron flux of $2.5 \times 10^{14} \text{ n/cm} \cdot \text{s}$.

However, normal operation was usually set to 15 MW reactor power.

Main components of HRPD and FCD/TD are the same, which consist of monochromator as the neutron wavelength selector, sample table, neutron detector, and some collimators as the neutron beam aligner. To stop the direct neutron beam, beam stopper made from neutron absorbing material (such as B_4C) is installed behind the detector at the direction of direct beam.

HRPD is utilized to investigate the structure, nuclear phases, even magnetic phases of crystalline materials which cannot be done by XRD [1, 2]. The high resolution feature of this instrument comes from 3 times collimation from collimator mounted before monochromator, between monochromator and sample table, as well as between sample table and detector. HRPD has 32 neutron detectors, one of each covers 5 degree of 2θ . Smallest 2θ step of 0.02 degree can be obtained. These detectors are

* Corresponding author. Tel./Fax.: +62-877-7566-7720

E-mail: rinakamila14@batan.go.idDOI: [10.17146/tdm.2020.22.3.5998](https://doi.org/10.17146/tdm.2020.22.3.5998)

extremely heavy, thus air cushion is used to lift them from the floor.

FCD/TD, on the other hand, can obtain texture data to analyze the crystalline orientation. A recent example is the texture of Stainless Steel SS420 [3]. This is possible due to the four-circle configuration that refer to four rotation axis consisting of two instrument rotation axis, which are 2-theta (2θ) and theta (θ) or omega (ω), and 2 sample angular rotation axis which are pi (π) and chi (χ). Eulerian-cradle goniometer mounted on the top of the sample table made this angular rotation possible. Smallest 2θ step of 0.1 degree can be obtained.

Before analyzing the data using software analysis such as *FullProf* and *MAUD*, the acquired raw neutron diffraction data should be processed in a specific way to achieve the suitable data format. For HRPD, not only the format, the content of raw data itself needs to be corrected first. This is because the actual spacing between the center of two neighboring detectors in HRPD may be slightly deviated and not exactly 5 degree. Moreover, each detector has different efficiency. The detector spacing correction data and detector efficiency data of HRPD was prepared in advance [4]. FCD/TD, having only 1 detector, has much simpler data processing with only background correction needed and it depends on instrument performance which sometimes has stochastic behavior. The background correction data was also prepared in advance.

This data processing is a repetitive task on every single experimental data, which previously done manually using “copy-paste-edit” method, manipulating each data column separately one by one via Microsoft Excel. This method has been done for years and remarkably time-consuming with high risk of human error.

Programming can turn manual task into automatic, making it more effective and efficient. *Python* [5] is one of programming languages that is widely used in academic and scientific communities. It is considered superior in terms of simple syntax, so one can focus more on the problem solving than the syntax [9]. Furthermore, *Python* is designed to be easy to learn and focuses on its reliability. In particular, *Python* is very powerful for data manipulation (acquisition, editing, processing, etc) and computation that in Radboud University, it is used instead of C++ [6]. *Python* is also very useful in various research field such as simplifying genomic simulation and mass spectrometric data, automatic wavelength calibration on spectroscopic data, etc [7–9]. In 2010, Zolnierczuk *et al* carried out the automation of neutron scattering experiments on one of the

scientific instrument at the Oak Ridge National Laboratory using *Python* [10].

This work was aimed to automate and optimize the processing of raw neutron diffraction data of HRPD and FCD/TD using *Python*. As the input data for the *Python* code, raw neutron diffraction data of Silicon Si640d standard sample taken from both HRPD and FCD/TD are used.

2. METHODOLOGY

2.1 Instrumentation Specifications

Two of neutron diffractometers in BATAN, HRPD (Fig. 1) and FCD/TD (Fig. 2), were installed in different location. HRPD is located in neutron guide hall (NGH) while FCD/TD located in experimental hall reactor (XHR) (see Fig. 3) [11]. Since the installation of both instruments in 1992, many developments have been done either in its control system or in data acquisition system. Latest specification and parameter of each instrument is shown in Table 1 and 2.



Fig. 1. High Resolution Powder Diffractometer



Fig. 2. Four Circle Diffractometer / Texture Diffractometer (FCD/TD)

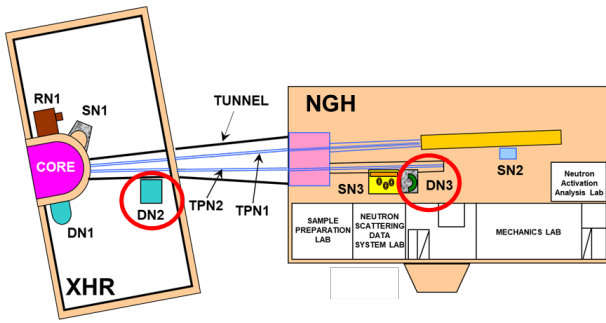


Fig. 3. Layout of NSL - BATAN [11]

Table 1. Specification and parameter of HRPD [1, 12]

Beam tube	S5 radial, NGT-2, 33 x 90 mm ²
Take-off Angle	41.5°, 89° (present), 130°
Monochromator	Hot-pressed Ge (311)
Wavelength	1.821399 Å
Beam size	40 x 100 mm ²
Collimator	$\alpha_1 = 16.2'$ (0.27°) $\alpha_2 = 20'$ (0.33°, 25 x 100 mm ²) $\alpha_3 = 12'$ (0.2°, 20 x 100 mm ²)
Scattering angle	-170° < 2θ _S < +10°
Detector	3He, XERAM 30NH15
Control system	Rigaku
Data acquisition	MESYDAQ

Table 2. Specification and parameter of FCD/TD [13]

Beam tube	S5 radial, NGT-2, 33 x 90 mm ²
Take-off Angle	± 46°
Monochromator	Si (311)
Wavelength	1.2799 Å
Beam Size	20 x 20 mm ²
Collimator	$\alpha_2 = 20', 30'$, or open $\alpha_3 = 20', 30'$
Scattering angle	-20° < 2θ < +105°, -30° < θ < +70°
Angular range	-40° < χ < +100°, -180° < π < +180°
Detector	BF ₃ point detector
Control system	Rigaku
Data acquisition	Visual Basics

2.2 Automatic data processing for HRPD

Three files are required; raw neutron diffraction data, detector spacing correction data, and detector efficiency data. The contain layout of each file is shown in Table 3-5 (within the red box). In normal experiment, initial value and step of 2θ is set to 2.50 and 0.05 respectively. However, the actual step may slightly vary due to limitation of the motor stepper resulting in irregular increment of 2θ.

Table 3. Contain layout of raw data of HRPD

2θ	Count on the detector number-						
	1	2	3	...	32		
2.50	00	xx	xx	xx	...	xx	00
2.55	00	xx	xx	xx	...	xx	00
2.60	00	xx	xx	xx	...	xx	00
2.65	00	xx	xx	xx	...	xx	00
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
7.45	00	xx	xx	xx	...	xx	00

Table 4. Contain layout of detector spacing correction

Det. No-	1	2	3	...	32
Spacing (ϑ)	xx	xx	xx	...	xx

Table 5. Contain layout of detector efficiency data

Det. No-	1	2	3	...	32
Efficiency (η)	xx	xx	xx	...	xx

Detector efficiency data is used to correct the diffracted neutron intensity, or count, on each detector. The corrected count on the nth detector (C_n^{*}) can be calculated using Eq. (1).

$$C_{n,m}^* = \frac{C_{n,m}}{\eta_n} \quad (1)$$

C_{n,m} is the uncorrected count on the nth detector. The value of subscript m depends on the step. Here, since the step is 0.05, m has value from 1 to 100 indicating that each detector has 100 value of 2θ. The result is then rounded to zero expected value (no digit after comma, i.e. integer).

The 2θ scanning in HRPD is only done in 5 degree and this covers 160 degree since 32 detector move simultaneously. Detector spacing correction data is used to correct the value of 2θ. First detector has zero correction as it acts as the reference point. The corrected 2θ for the nth detector, 2θ_n^{*}, can be obtained through Eq. (2).

$$2\theta_{n,m}^* = 2\theta_{1,m} + \sum_{i=1}^n \vartheta_i \quad (2)$$

2θ_{1,m} is the mth value of 2θ from the 1st detector. For example, the 3rd value of 2θ on the 5th detector, 2θ_{5,3}^{*}, is 2θ_{1,3} + ϑ₁ + ϑ₂ + ϑ₃ + ϑ₄ + ϑ₅. The result is then rounded to 2 digit after comma.

Two kinds of data format are targeted as the result; one is called the XY format with two columns data (2θ and count) and one other is called the intensity data series, or the free format with only one column data (intensity or count). Table 6 shows the content layout of XY format (red box) and free format (blue box).

All file can be treated as array by using the readline command, then stripping and splitting the

component based on the delimiter. The detector spacing correction and detector efficiency data will become 1-dimensional array while raw data will become 2-dimensional one. 2θ data is taken from first component of each line in raw data file while count data is taken from 3rd to 34th component. 2nd and last (35th) component which only contain zeros (00) in each line are neglected since both are only dummy values. Last line of the raw data should be popped out because the last value of 2θ on a detector is the first value of 2θ on the next detector. Each component, either 2θ or count, is corrected based on Eq. (1) and (2). The corrected values ($2\theta^*$ and C^*) are collected in two new array named “thetaneu” and “countneu” respectively and then stacked together in two new files, one for the XY and the other for free format, with specific layout arrangement as in Table 6. Both new files are having .dat extension.

For the free format file, there are header and some constants in the first two rows. Header is taken from the name of raw neutron diffraction data file which is input by the user. The constants are the neutron wavelength (λ), 2θ initial, ($2\theta_{1,1}$), 2θ step (0.05), and the number of data (Ndata) which can be calculated using Eq. (3).

$$Ndata = \frac{5 \times 32}{step} \tag{3}$$

Table 6. XY format (red box) and free format (blue box)

	$2\theta^*$	C^*
1 st Detector	xx	xx
	xx	xx
	xx	xx
	xx	xx
	⋮	⋮
2 nd Detector	xx	xx
	xx	xx
	xx	xx
	xx	xx
	⋮	⋮
⋮	xx	xx
	⋮	⋮
	⋮	⋮
	⋮	⋮
	⋮	⋮
32 th Detector	xx	xx
	xx	xx
	xx	xx
	xx	xx
	⋮	⋮
	xx	xx

By using *Matplotlib* modul [14], graph of diffraction or diffractogram ($2\theta^*$ vs C^*) could be displayed. With yes/no option, user can decide whether to display the diffractogram or not. The diffractogram can be manually saved as picture.

2.3. Automatic data processing for FCD/TD

Since the only correction for the raw data in FCD/TD is the background correction of the diffracted neutron count, the *Python* code is simpler than one for HRPD. The contain layout of raw neutron diffraction data and background correction data are shown together in Table 7 with the initial value and step of 2θ are set to 5.00 and 0.10, respectively. In actual experiment, range of 2θ can be customized and the increment of 2θ may also be slightly vary due to the limitation of motor stepper as in the case of HRPD. C_i^* count is corrected through Eq. (4).

$$C^* = \frac{C}{Corr} \tag{4}$$

Table 7. Contain layout of raw data of FCD/TD (red box) and background correction data (blue box)

θ or ω	2θ	Count	Corr
2.50	5.00	xx	xx
2.55	5.10	xx	xx
2.60	5.20	xx	xx
2.65	5.30	xx	xx
⋮	⋮	⋮	⋮
52.50	105	xx	xx

With array treatment, raw data and background normalization data will become 2 dimensional and 1 dimensional array respectively. 2θ data are taken from 2nd component of each line in raw data while count data is taken from its 3rd component and directly corrected by dividing it with the normalization value taken from the other file which corresponds to the same 2θ value. Two new arrays named “theta” and “count” are prepared to collect the value of 2θ and corrected counts C^* , respectively, with the rest (stacking in two new files and viewing the diffractogram) are exactly the same as in HRPD.

To validate these codes, the diffractogram resulted from both *hrpd.py* and *fcdd.py* were directly compared with the diffractogram resulted by manual method using Microsoft Excel. Identical diffractogram indicates the correctness of the codes.

3. RESULTS AND DISCUSSION

The *Python* code for automatic data processing of HRPD and FCD/TD is named as *hrpd.py* and

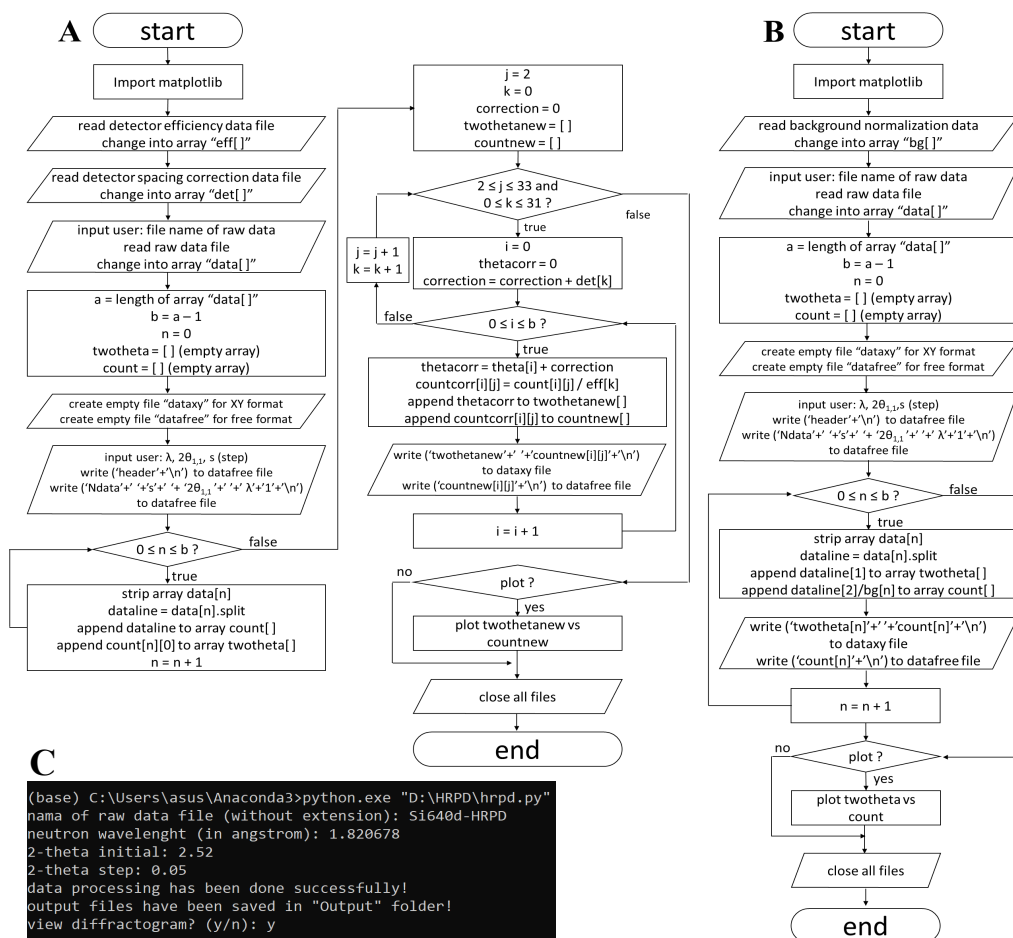


Fig. 4. Flow chart of *Python* code for HRPD (A), FCD/TD (B), and example of *hrpd.py* code running through Command Prompt (C)

fcdd.py respectively. The flow chart is shown in Fig. 4 (A) and (B). Both can be run through the command prompt (cmd.exe). Graphical User Interface (GUI) is not needed due to the low complexity of the code. Even a complex code might also be run through the command prompt [10]. *Anaconda* platform [15] for *Python* ver.2.7.8 is used. The base activation should be conducted first using 'conda activate base' command. The code and input file must be located in the same directory while the output files are saved inside a sub folder in that directory. Using cmd, after directing to that directory where the *Python* is installed, the code is run by 'python.exe' command space quotation marks containing the code location and name. Next, an instruction to input the name of raw data (without extension), neutron wavelength, 2θ initial and 2θ step will appear. Fill it properly, then press enter and the code will be running until an option to whether to view diffractogram comes up (see Fig. 4 (C) for the example).

Until this step, raw data processing into ready-to-refine data has been completed. It is very practical compared to the manual "copy-paste-edit"

method using Microsoft Excel. Two output files, the XY and free format file, will emerge in the directory of output file. The user should simply type 'y' and enter to view the diffractogram or 'n' and enter to end the code without it.

The computer equipment used in this work has Intel Core i3 processor with 4 GB of RAM. Table 8 shows the comparison of average time for data processing between using manual and automatic method. Using this *Python* code, the average time needed to perform the data processing step is significantly reduced to only 20 seconds or less for both HRPD and FCD/TD, which is 120 times faster for HRPD and 30 times faster for FCD/TD using manual method. Automating the repetitive and time-consuming actions is clearly achieved in this work as in other works by other researchers [7, 8]. Fig. 5 shows the diffractogram of Si640d acquired from HRPD (A) and FCD/TD (B). There is no difference between the result from manual method via Microsoft Excel (red) and the result from *Python* code via *Matplotlib* module (blue). This clearly indicates that the codes are working properly and the results are correct. These

automatic *Python* codes have successfully made the data processing become much easier, simpler, more effective and efficient with nearly no risk of human error.

Table 8. Comparison of required time (average) for data processing between using manual and automatic method

	HRPD	FCD/TD
Manual	40 minutes	10 minutes
Automatic	20 seconds	20 seconds

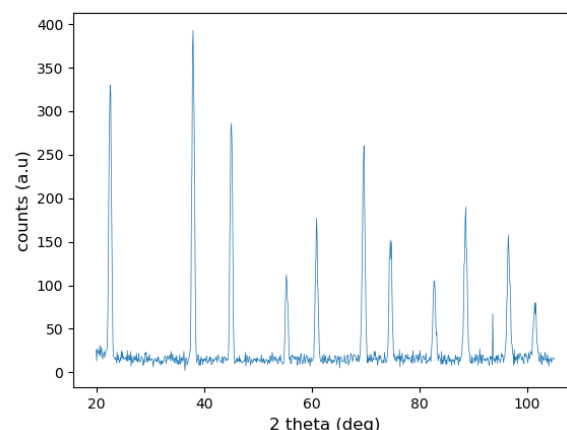
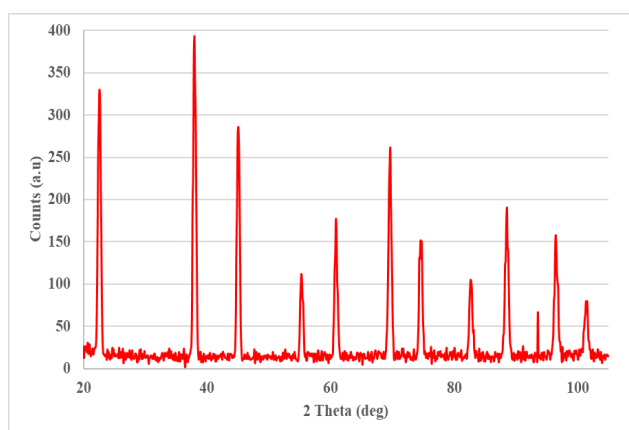
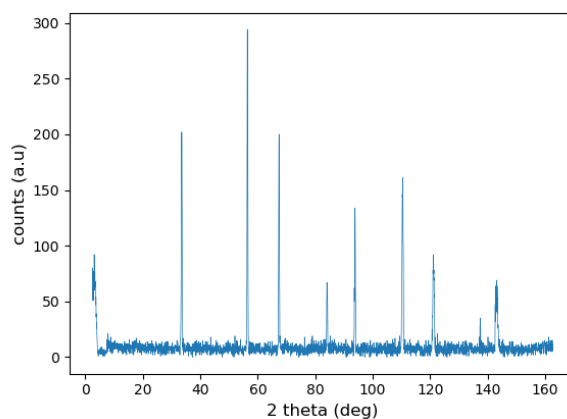
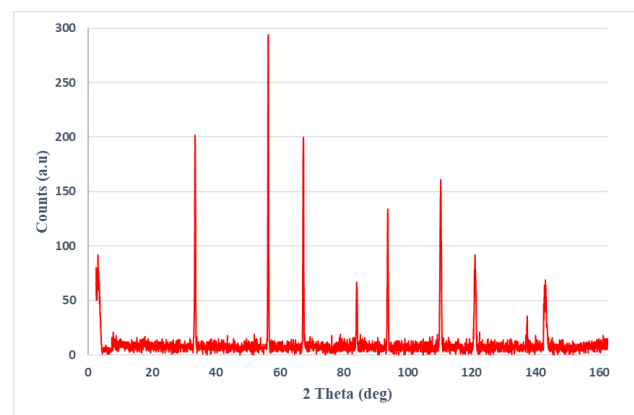


Fig. 5. Comparison between diffractogram of Si 640d using HRPD (A) and FCD/TD (B) with manual method (red) and Python code (blue)

30 times faster for FCD/TD. The codes have also been validated. The data processing has become significantly simpler, effective, and highly efficient.

ACKNOWLEDGMENT

We would likely to thank Dr. Eng. Iwan Sumirat as the Head of Neutron Beam Technology Division, Dr. Andon Insani, M.Eng and Drs. Tri Hardi Priyanto, M.T as the person in charge of HRPD and FCD for the uncountable support.

5. CONCLUSION

Automatic data processing system for HRPD and FCD/TD has been successfully created using *Python* code. It processes raw neutron diffraction data into corrected and formatted data which is ready to be refined and analyzed. This code reduces the data processing time significantly into only around 20 seconds which is 120 times faster for HRPD and

AUTHOR CONTRIBUTION

Muzakkiy P. M. Akhir and Rina Kamila equally contributed as the main contributors of this paper. All authors approved the final version of the paper.

REFERENCES

1. Bharoto, Insani A. Development of Data Acquisition and Control Software for High Resolution Powder Diffractometer. INKOM. 2014. **8**:47–51.
2. Insani A., Mugirahardjo H. Effect of Collimator 3 Degree Divergence Modification

- of Neutron Diffractometer DN3 on Ni and TiO₂ Scattering Quality (Pengaruh Modifikasi Sudut Divergensi Kolimator 3 Difraktometer Neutron DN3 terhadap Kualitas Hamburan Ni dan TiO₂). in: *Prosiding Seminar Hamburan Neutron dan Sinar X ke 7*. 2015. pp. 90–4.
3. Priyanto T.H., Muslih R., Mugirahardjo H., Insani A. Effects of the Preheating Temperature on the Crystal Structure and Texture of Martensitic Stainless Steel. *Makara J. Technol.* 2018. **22**(2):79–83.
 4. Mugirahardjo H., Insani A., Bharoto, Muslih R., Santoso E. Counting System Characterization of High Resolution Neutron Powder Diffractometer (HRPD-DN3) (Karakterisasi Sistem Pencacah Difraktometer Neutron Serbuk Resolusi Tinggi (HRPD-DN3)). in: *Prosiding Seminar Penelitian dan Pengelolaan Perangkat Nuklir*. 2013. pp. 261–7.
 5. *Welcome to Python* [Accessed: 2 July 2020]. Available from: <https://www.python.org/>.
 6. Huy N. *Exploring Python as a replacement for C++ in Imperative Programming for Computing Science at Radboud University*. Radboud University; 2019.
 7. Yunker L.P.E., Ting M., Yeung D., Mcindoe J.S. PythoMS: A Python Framework To Simplify and Assist in the Processing and Interpretation of Mass Spectrometric Data. *J. Chem. Inf. Model.* 2019. **59**:1295–300.
 8. Gladstein A.L., Quinto-cortés C.D., Pistorius J.L., Christy D., Gantner L., Joyce B.L. SimPrily: A Python framework to simplify high-throughput genomic simulations. *SoftwareX.* 2018. **7**:335–40.
 9. Cardiel N. Using Python to Simplify Spectrometri Data.pdf. in: *ASP Conference Series*. 2019. pp. 499–502.
 10. Zolnierczuk P.A., Riedel R.A. *Neutron Scattering Experiment Automation with Python*. 2010.
 11. Abarrul I. Neutron Scattering Activities in Serpong (Kegiatan Hamburan Neutron di Serpong). in: *Prosiding Seminar Hamburan Neutron dan Sinar X*. 2001. pp. 36–40.
 12. Fajar A., Mugirahardjo H. The Performance of Fine Resolution Neutron Powder Diffractometer at PTBIN-BATAN. *Atom Indones.* 2010. **36**:1–9.
 13. Priyanto T.H., Mugirahardjo H., Muslih R., Ramadhani A., Priyanto T.H., Mugirahardjo H., et al. Development of BATAN 's texture diffractometer (current status). in: *AIP Conference Proceeding*. 2015.
 14. *Matplotlib Python plotting – Matplotlib 3* [Accessed: 2 July 2020]. Available from: <http://matplotlib.sourceforge.net/>.
 15. *Anaconda - The World's Most Popular Data Science Platform Title* [Accessed: 2 July 2020]. Available from: <https://www.anaconda.com/>.