DECal, a Python tool for the efficiency calculation of thermal neutron detectors based on thin-film converters

Álvaro Carmona Basáñez\textsuperscript{a,*}, Kalliopi Kanaki\textsuperscript{a}, Francesco Piscitelli\textsuperscript{a}

\textsuperscript{a}European Spallation Source ERIC, P.O. Box 176, SE-221 00 Lund, Sweden

Abstract

The Detector Efficiency Calculator (DECal) is a series of Python functions and tools designed to analytically calculate, visualise and optimise the detection efficiency of thermal neutron detectors, which are based on thin-film converters. The implementation presented in this article concerns\textsuperscript{10}B-based detectors in particular. The code can be run via a graphical user interface, as well as via the command line. The source code is openly available to interested users via a GitHub repository.

PROGRAM SUMMARY

Program Title: DECal

\textit{Licensing provisions: Free for non-commercial use (as per terms in LICENSE file)}

\textit{Programming language:} Python Version 2.7 and 3.3

\textit{Operating system:} OS X (preferred) and Linux

\textit{Version:} 1.0.0

\textit{Keywords:}

Detector efficiency calculator, neutron scattering, Boron-10, neutron detector, efficiency, thin film, converter, Python, PyQt5,

\textit{Nature of problem:}

Implementation of a Python-based tool to calculate and optimise detection effi-

\textsuperscript{*}Corresponding author.

\textit{E-mail address:} alvaro.carmonabasanez@esss.se
ciency for thin-film thermal neutron detectors via a graphical user interface and a command line interface.

1. Introduction

The European Spallation Source (ESS) ERIC is a joint European organisation committed to the construction and operation of the world’s leading facility for research using thermal neutrons. The ESS is designed to be the world’s brightest neutron source and the instantaneous neutron flux on detectors at ESS full power will be without precedent. In general, neutron scattering facilities are going toward higher fluxes and this translates into higher demands on instrument and detector performance: higher counting rate capability, better timing and finer spatial resolution are requested among others.

The current thermal neutron detector technology is reaching fundamental limits and most of the neutron sources in the world, including ESS, are pushing the development of their detector technologies, to tackle the increased flux available and the scarcity of $^3$He, the so-called “Helium-3 crisis”. $^{10}$B along with the $^3$He and $^6$Li isotopes are the main actors in thermal neutron detection due to their large neutron absorption cross sections. Due to its favourable properties, $^3$He (a rare isotope of He) has been dominating thermal neutron detection for decades.

Nowadays, the importance of solid conversion layers is increasing as this technology appears to be a viable and promising alternative to $^3$He. Recently, high-quality, low-cost production of square metres of $^{10}$B$_4$C-coated substrates became possible. The detection efficiency of a single thin layer of $^{10}$B$_4$C is limited to a few percent at thermal neutron energies compared to the very high efficiencies of $^3$He-based detectors. The Multi-Blade, the Jalousie detector, the A1CLD and many others, are examples of the detector developments which exploit solid neutron converters operated at a grazing angle in order to increase the detection efficiency. A different way to increase the efficiency is to arrange many layers in sequence. Examples of detector
developments arranging several layers are the Multi-Grid [22, 23, 24, 25, 26], CASCADE [27, 28] and many others [29, 30, 31, 32, 33, 34, 35, 36, 37].

The difference between the physical processes of a gas-based (such as $^3$He) and a solid-converter-based detector (such as $^{10}$B$_4$C) is described in detail in [38, 39]; a detector based on solid converter has several parameters that can be tuned to increase the detection efficiency. A simple model of single and multi-layer thermal neutron detectors is explored mainly analytically to help optimize the design in different circumstances. Several theorems are deduced that help guide the design. Using powerful simulation software has the advantage of including many effects and potentially results with high accuracy [40]. On the other hand it does not always give the insight an equation can deliver. The equations in [39, 41] and the tool described in this manuscript focus on $^{10}$B$_4$C-based thermal neutron detectors but can be extended to any detector based on thin-film converters.

The theoretical details of the calculations which this software implements in Python are described in [39, 41] and more specifically in the Ph.D. thesis [42]. The authors explain a series of equations to calculate the efficiency of detectors based on thin film neutron converters given the specific geometry of the detector and the neutron beam characteristics.

2. Detector geometry and neutron beam configuration

A detector can be made of a single or multiple layers. A layer consists of a substrate material, usually Aluminium, on which the $^{10}$B$_4$C converter is coated, either on one or both sides. A double-coated layer is referred to as blade. The number of converter layers, their thickness and their composition matter. In addition, a detector can contain layers of the same or varying converter thickness. This parameter can be optimised accordingly for a single neutron wavelength or for a distribution of neutron wavelengths, a scenario which is closer to reality in a neutron scattering instrument. The material and thickness of the substrate are not considered in the calculations presented here and will
be the topic of a future improvement.

The incoming neutrons can enter this geometrical arrangement at an angle or perpendicularly, depending on the needs of the application (see Fig. 1). Both the neutron incident angle and its wavelength (energy) affect the efficiency and thus enter the respective calculation function.

Fig. 1 summarizes all parameters that impact the detection efficiency. The neutron beam hits the detector with an incident angle $\Theta$. A numbered series of layers follow. A layer is called back-scattering when neutrons are incident from the gas-converter interface and the escaping particles are emitted backwards into the gas volume; it is called a transmission layer when neutrons are incident from the substrate-converter interface and the escaping fragments are emitted in the forward direction in the sensitive volume. In Fig. 1 the detection efficiency of every blade is the sum of the back-scattering and transmission layer efficiencies, as the substrate holds two converter layers, one in back-scattering mode and one in transmission mode.

![Diagram](image_url)

Figure 1: Depiction of detector geometry and neutron beam arrangement. The beam hits the parallel blades with an incident angle $\Theta$. The blades consist of a substrate on which the converter is coated on both sides.
3. Software overview

DECal has been developed as a cross-platform software for Mac OS X (version 10.11 and higher) and Linux (tested on CentOS 7 and Ubuntu 17.4). It has been written in Python 3. The objective of this project has been to develop an open source tool that allows the user to calculate the detection efficiency of a $^{10}$B-based multi-blade neutron detector and optimise the respective geometry parameters.

The software features allow the user to set the parameters of the neutron beam and detector with intuitive visualisation features. The package containing the efficiency calculation functions is distributed via Python Package Index (pip).

The configurations can be exported in JSON format, and this JSON file can be imported with graphical interface as well as in the functions of the library as input.

3.1. Software availability

The software is divided in two different repositories. DECal is available in https://github.com/DetectorEfficiencyCalculator/dg.efficiencyCalculator In the repository https://github.com/alvcarmona/neutronDetectorEffFunctions the users can find the library that holds the calculation functions. It is also available via easy install via the following command:

```bash
$ pip install neutron_detector_eff_functions
```

3.2. Requirements and dependencies

This is a list of hardware and software requirements for the users to access all the features of the tool:

- OS X ≥10.10, Linux Ubuntu 17.4 and CentOS 7 are the OS where the software has been tested
- Multi-core CPU
• 1200×610 screen resolution or higher
• Python v2.7 or 3.3 and higher
• Qt5 for the DECal application

The Python libraries available via pip needed to execute the DECal software are included in the requirements.txt file found in the repository. These are some of them:

• SciPy [43]
• NumPy [44]
• Matplotlib ≥ 1.5.0 [45]
• PySide ≥ 1.2.4
• PyQt5 (not available via pip in some systems)

For more information about the environment needed the user can refer to the requirements.txt file.

4. Software Design

The classes used to represent the Detector and its parts are the following: Detector, Blade, B10. The Detector class is the object that represents a detector configuration and has the plot functions. To deal with the construction of the Detector entity with the different parameters there is a class that is built with the Builder pattern [46]. The builder pattern builds a complex object using simple objects and using a step by step approach. The builder function builds a detector entity depending on the parameters given.
4.1. File structure

Script functions are in `scripts.py`. In this folder there are other basic Python files like the requirements file and the license. The `exports` folder contains some examples of exported neutron configurations and wavelengths. The `Detector` class is written in `Detector.py`. `Efftools` holds the core efficiency calculation functions and the metadata functions for plotting. The entire tool is built around `efficiency4boron` and `efficiency2particles` functions. These functions are the ones used to calculate the theoretical efficiency of a neutron detector using B10 as converter.

-----------
neutronDetectorEffFunctions library file structure
-----------

`README.md`

`LICENSE.txt`

`neutron_detector_eff_functions/`

   `Aluminium.py`
   `B10.py`
   `Blade.py`
   `Converter.py`
   `Detector.py`
   `Detector_meta.py`

`data/`

   `Aluminium/`
     `AlCrossSect_(n,g).py`
   `B10/`
     `10B4C220/`
       `IONIZ_Alpha06.py`
The data subpackage contains the files the application uses to calculate the cross-section of different converter materials. In the current version of the software there are two available configurations of B10. These files have been generated with a tool called SRIM (The Stopping and Range of Ions in Matter) [47]. This data is the stopping power of the materials. The user can find the description of the materials in the header lines of the files efficiencyCalculator/data/B10/. This is an example of this information:

============= TARGET MATERIAL ================
Layer 1 : Layer 1
Layer Width = 50000.00 Å

Layer #1 - Density = 12.74E22 atoms/cm³ = 2.207 g/cm³
Layer #1 - B = 77.6 Atomic Percent = 74.4 Mass Percent
Layer #1 - C = 20 Atomic Percent = 23.0 Mass Percent
Layer #1 - B = 2.4 Atomic Percent = 2.53 Mass Percent

====================================================================
Total Ions calculated =000000.00
====================================================================

Ionization Energy Units are >>>> eV/(Angstrom-Ion) <<<

Fig. 2 represents the class diagram of the DECal project.

Figure 2: Class diagram of the DECal project.
5. High level functions

A set of functions has been developed for simplifying the use of the classes included in the project. The functions can be called using the `scripts.py` file located in the root folder of the library. The explanations listed in the following serve documentation purposes.

`calculate_eff_multiblade(nb, converterThickness, substrateThickness, wavelength, angle, threshold, single, converter)`

The function returns a list of two positions. The first contains a list of the efficiency for each blade in depth order (the last is the deepest) and the second value is the total efficiency of the detector. This is an example of usage:

```
calculate_eff_multiblade(10, 1, 0, [[1.8,100]], 90, 100, False, '10B4C 2.24g/cm3')
```

The function call returns the efficiency of a neutron detector configuration of 10 blades with a converter thickness of 1 micrometre, 0 micrometres of substrate, a neutron wavelength of 1.8 Å (monoenergetic), 90 degrees of incident neutron angle, an applied threshold of 100 keV on the total energy deposition from the ionized conversion products, double-coated layers and a $^{10}$B$_4$C converter with a density of 2.24 g/cm$^3$ [10].

`calculate_eff_json(path)`

The function calculates the efficiency of a configuration loaded from a JSON file. The user can generate this configuration via the graphical user interface.

`plot_eff_vs_thick(path)`

This function plots the detector efficiency as a function of different converter thicknesses. The argument is the path to a JSON configuration file. A typical result is presented in Fig. 3.

`plot_eff_vs_wave(path)`
Figure 3: Detector efficiency vs. converter thickness for a set of default beam and detector parameters, e.g. 1.8 Å wavelength, 1.3 μm converter thickness, read from a typical JSON file in GitHub.

Providing a path to a JSON file as parameter, this function plots the efficiency as a function of a monochromatic neutron wavelength. The resulting plot is displayed in Fig. 4.

optimize_config_same_thick(originPath, destinyPath)

The function returns a list of converter thicknesses optimised to maximise detector efficiency, with the condition that all converter thicknesses are identical. originPath points to a configuration from a JSON file. destinyPath is the destination where the new configuration is saved in JSON format.

for each blade the same converter thickness is applied on both substrate sides.

optimize_config_diff_thick(originPath, destinyPath)

Same as before but with the condition that different blades can have different converter thicknesses.
Figure 4: Detector efficiency vs. neutron wavelength, produced from a typical JSON file in GitHub.

6. Graphical user interface tool

A graphical user interface (GUI) has been developed for easy access to the features of the software. The GUI can be executed running the launch.py script, located in the root folder of the project. This invokes the main window as shown in Fig. 5. The detector configuration list is empty to start with. To create a new configuration click on the New configuration button and a new window pops up (see Fig. 6). Here the user can configure the neutron detector and neutron beam characteristics. When all the parameters are set, the user can proceed to calculate the efficiency, get an optimised configuration, export the data from the plots or export the configuration to a JSON file. The full list of use cases can be found in the use cases section.
Figure 5: Main GUI window, where detector configurations are listed. New configurations can be created from here or existing ones can be edited, duplicated and imported.

Figure 6: Detector configurator view. This is the window allowing the user to define the parameters, display plots and detector efficiency values.
7. **Use cases**

The following list describes the different functionalities available in the software.

*Create a configuration:* There are different parameters that can be changed with the user input: name, substrate, converter, and energy threshold. Then the user can *Add Wavelength* and *Add Blades*. The configuration can be saved for later use in this session by clicking *Save*, and can be exported by clicking on *Export Configuration*.

*Add wavelength:* The user can add wavelength ($\lambda$) in two ways, either by manually setting a value and its respective ratio (0–100%) or via an imported text file. The latter should have a two-column format as `in file 2gaussdistr.txt`. The tool will display a list of wavelengths and weights (see Fig. 7), and a plot (see Fig. 8).

![Figure 7: Example of an imported wavelength list with respective weights.](image)

*Add blades:* The user can set the converter thickness and the number of substrate layers. The GUI provides visualisation aid, i.e. a list with the added blades and plots of the thickness vs. the blade number. The user can set a single layer detector to separately see the backscattering and transmission efficiency values. To change the blade configuration the *Delete Blades* button has to be pushed or the desired value in the blade list can be modified (see Fig. 9).
Figure 8: Example of a plot from an imported wavelength list with a double Gaussian shape. The red line is at the barycenter of the wavelength distribution.

*Calculate efficiency:* When the configuration is complete, the user can calculate the efficiency pressing the *Calculate Efficiency* button or calling the function from the *scripts.py* file. When the efficiency is calculated, the window displays the total efficiency for double-coated layers or both backscattering and transmission values for single layer configurations. The window displays the respective efficiency of a blade in a multi-blade configuration. When a multi-blade detector has all the layers with the same thickness, a plot of the efficiency vs. the blade number and another plot with the total efficiency vs. converter thickness are displayed (see Fig. 3).

*Optimisation:* The list of converter thicknesses of detector can be optimised for maximum efficiency. The optimisation is done by calling the *Calculate Efficiency* function changing the thickness until the maximum is found. In a multi-blade detector, which has to be optimised for any distribution of neutron wavelengths or for a single wavelength, each blade has to hold two converter layers of the same thickness ([39], page 11). Naturally converter thicknesses on different blades can be distinct.
The most complex operation is the optimisation of a multi-blade detector with different coating thicknesses and polychromatic wavelength. This operation could take several minutes for a multi-core CPU but as it has been demonstrated ([39], page 16), it is a sufficient approximation to optimise using the barycenter of the wavelength distribution.

![Figure 9: Example of a list of blade thicknesses optimised as a function of depth with the aim of maximizing detector efficiency.](image)

| N | Conv thick (µm) | Substrate thick (µm) | Efficiency (%) |
|---|----------------|----------------------|----------------|
| 1 | 1.0            | 0.0                  | 17.8%          |
| 2 | 1.0            | 0.0                  | 12.3%          |
| 3 | 1.0            | 0.0                  | 8.90%          |
| 4 | 1.0            | 0.0                  | 6.55%          |
| 5 | 1.0            | 0.0                  | 4.92%          |
| 6 | 1.0            | 0.0                  | 3.77%          |
| 7 | 1.0            | 0.0                  | 2.94%          |
| 8 | 1.0            | 0.0                  | 2.33%          |
| 9 | 1.0            | 0.0                  | 1.88%          |

Exporting data: The active neutron detector configuration can be exported from the detector dialog tab. The configuration is exported in the JSON format. This JSON file can be imported in the main window or used as a parameter for a script function. The plots are exported with the desired format using the Save feature in the toolbar under the plots. Data from the plot is exported with the Export Data button under some of the plots. This is an example of a configuration exported as a JSON file:
Figure 10: Example of blade thicknesses optimised as a function of depth with the aim of maximizing detector efficiency.

```json
{
    "angle": 90,
    "blades": [
        {
            "backscatter": 4.0,
            "inclination": 0,
            "substrate": 0.0,
            "transmission": 0
        }
    ]
}
```
8. Conclusions

Based on prior analysis of the thin-film neutron detector characteristics an open source software is developed, in order to calculate and optimise detector efficiency. The focus of this work is on B10-based neutron detectors. The Python code produced to this end is accessible via a GitHub repository for the neutron scattering community to access, use and modify according to the users’ needs. To this end, the calculations are made available as a separate Python library but can be run via a GUI application as well. Various utilities are offered that make the software quick to start with and intuitive to use.

The software is available at
https://github.com/DetectorEfficiencyCalculator/dg_efficiencyCalculator and
https://github.com/alvcarmona/neutronDetectorEffFunctions. Part of the DE-Cal functionality is implemented as a web application and can be found in
https://github.com/alvcarmona/efficiencycalculatorweb.

Acknowledgments

This work was supported by the EU Horizon 2020 framework, BrightnESS project 676548. Álvaro Carmona Basáñez would like to thank the Basque gov-
ernment and Novia Salcedo for their partial financial support via the Global training grant.

References

[1] S. Peggs, et al., ESS Technical Design Report, ESS 2013-001, http://esss.se/scientific-technical-documentation, 2013.

[2] R. Cooper, et al., A program for neutron detector research and development - A White Paper based on a workshop held at Oak Ridge National Laboratory, Proceedings of the workshop held at Oak Ridge National Laboratory (2003).

[3] B. Gebauer, Nuclear Instruments and Methods in Physics Research Section A 535 (2004) 65 – 78. doi:10.1016/j.nima.2004.07.266

[4] K. Zeitelhack, Neutron News 23 (2012) 10–13. doi:10.1080/10448632.2012.725325

[5] O. Kirstein, et al., arXiv:1411.6194 Proceedings of the 23rd International Workshop on Vertex Detectors, 15-19 September, Macha Lake, The Czech Republic (2014).

[6] R. T. Kouzes, The $^3$He Supply Problem, Technical Report, Pacific Northwest National Laboratory, Richland, WA, 2009.

[7] D. Kramer, Physics Today 64 (2011) 20–23.

[8] D. A. Shea, D. Morgan, Congressional Research Service (2010).

[9] A.-J. Dianoux, G. Lander, Neutron Data Booklet, Second Edition, 2013.

[10] C. Höglund, J. Birch, K. Andersen, T. Bigault, J.-C. Buffet, J. Correa, P. van Esch, B. Guerard, R. Hall-Wilton, J. Jensen, A. Khaplanov, F. Piscitelli, C. Vettier, W. Vollenberg, L. Hultman, Journal of Applied Physics 111 (2012). doi:10.1063/1.4718573
[11] S. Schmidt, C. Höglund, J. Jensen, L. Hultman, J. Birch, R. Hall-Wilton, Journal of Materials Science 51 (2016) 10418–10428. doi:10.1007/s10853-016-0262-4

[12] C. Höglund, K. Zeitelhack, P. Kudejova, J. Jensen, G. Greczynski, J. Lu, L. Hultman, J. Birch, R. Hall-Wilton, Radiation Physics and Chemistry 113 (2015) 14–19. doi:10.1016/j.radphyschem.2015.04.006

[13] F. Piscitelli, The European Physical Journal Plus 130 (2015) 1–9. doi:10.1140/epjp/i2015-15027-3

[14] F. Piscitelli, J. C. Buffet, J. F. Clergeau, S. Cuccaro, B. Guérand, A. Kaplanov, Q. L. Manna, J. M. Rigal, P. V. Esch, Journal of Instrumentation 9 (2014) P03007.

[15] F. Piscitelli, F. Messi, M. Anastasopoulos, T. Bryš, F. Chicken, E. Dian, J. Fuzi, C. Höglund, G. Kiss, J. Orban, P. Pazmandi, L. Robinson, L. Rosta, S. Schmidt, D. Varga, T. Zsiros, R. Hall-Wilton, Journal of Instrumentation 12 (2017) P03013.

[16] M. Henske, M. Klein, M. Köhli, P. Lennert, G. Modzel, C. Schmidt, U. Schmidt, Nuclear Instruments and Methods in Physics Research Section A 686 (2012) 151 – 155. doi:10.1016/j.nima.2012.05.075

[17] G. Modzel, M. Henske, A. Houben, M. Klein, M. Köhli, P. Lennert, M. Meven, C. Schmidt, U. Schmidt, W. Schweika, Nuclear Instruments and Methods in Physics Research Section A 743 (2014) 90 – 95. doi:10.1016/j.nima.2014.01.007

[18] R. Kampmann, IEEE NSS, Anaheim N1-109 (2012).

[19] R. Kampmann, First Tests of Novel Neutron Detectors with Thin Conversion Layers in Inclined Geometry at REFSANS, Oral presentation at the Second International $^{10}$B-BF$_3$ Detectors Workshop, 13-14 March, ILL, Grenoble (2012). URL: https://www.ill.eu/10bbf3
[20] E. P. Cippo, G. Croci, A. Muraro, A. Menelle, G. Albani, M. Cavenago, C. Cazzaniga, G. Claps, G. Grosso, F. Murtas, M. Rebai, M. Tardocchi, G. Gorini, Journal of Instrumentation 10 (2015) P10003.

[21] G. Croci, et al., EPL 107 (2014) 12001. doi:10.1209/0295-5075/107/12001.

[22] M. Anastasopoulos, R. Bebb, K. Berry, J. Birch, T. Brys, J.-C. Buffet, J.-F. Clergeau, P. Deen, G. Ehlers, P. van Esch, S. Everett, B. Guérard, R. Hall-Wilton, K. Herwig, L. Hultman, C. Högland, I. Iruretagoiena, F. Issa, J. Jensen, A. Khablanov, O. Kirstein, I. L. Higuera, F. Piscitelli, L. Robinson, S. Schmidt, I. Stefanescu, Journal of Instrumentation 12 (2017) P04030.

[23] K. Andersen, T. Bigault, J. Birch, J. Buffet, J. Correa, R. Hall-Wilton, L. Hultman, C. Högland, B. Guérard, J. Jensen, A. Khaplanov, O. Kirstein, F. Piscitelli, P. V. Esch, C. Vettier, Nuclear Instruments and Methods in Physics Research Section A 720 (2013) 116 – 121. doi:10.1016/j.nima.2012.12.021.

[24] J. Birch, J.-C. Buffet, J.-F. Clergeau, J. Correa, P. van Esch, M. Ferraton, B. Guérard, J. Hallwachs, R. Hall-Wilton, L. Hultman, C. Högland, A. Khablanov, M. Koza, F. Piscitelli, M. Zbiri, Journal of Physics: Conference Series 528 (2014) 012040.

[25] J. Birch, J. C. Buffet, J. Correa, P. van Esch, B. Guérard, R. Hall-Wilton, C. Högland, L. Hultman, A. Khaplanov, F. Piscitelli, IEEE Transactions on Nuclear Science 60 (2013) 871–878. doi:10.1109/TNS.2012.2227798.

[26] B. Guérard, J. Buffet, Ionizing radiation detector - patent no. 20110215251., 2011. URL: https://www.google.com/patents/US20110215251, US Patent App. 13/038,915.

[27] M. Klein, C. J. Schmidt, Nuclear Instruments and Methods in Physics Research Section A 628 (2011) 9–18. doi:10.1016/j.nima.2010.06.278.
Proceedings of the 12th International Vienna Conference on Instrumentation.

[28] M. Klein, Experimente zur Quantenmechanik mit ultrakalten Neutronen und Entwicklung eines neuen Detektors zum ortsaufgelösten Nachweis von thermischen Neutronen auf grossen Flächen, Ph.D. thesis, University of Heidelberg, 2000.

[29] I. Stefanescu, Y. Abdullahi, J. Birch, I. Defendi, R. Hall-Wilton, C. Höglund, L. Hultman, D. Seiler, K. Zeitelhack, Nuclear Instruments and Methods in Physics Research Section A 727 (2013) 109 – 125. doi:10.1016/j.nima.2013.06.003.

[30] I. Stefanescu, Y. Abdullahi, J. Birch, I. Defendi, R. Hall-Wilton, C. Höglund, L. Hultman, M. Zee, K. Zeitelhack, Journal of Instrumentation 8 (2013) P12003.

[31] D. Pfeiffer, F. Resnati, J. Birch, R. Hall-Wilton, C. Hglund, L. Hultman, G. Iakovidis, E. Oliveri, E. Oksanen, L. Ropelewski, P. Thuiner, Journal of Instrumentation 10 (2015) P04004.

[32] J. L. Lacy, L. Sun, C. S. Martin, A. Athanasiades, T. D. Lyons, in: 2009 IEEE Nuclear Science Symposium Conference Record (NSS/MIC), pp. 1117–1121. doi:10.1109/NSSMIC.2009.5402421.

[33] J. L. Lacy, A. Athanasiades, N. N. Shehad, R. A. Austin, C. S. Martin, IEEE NSS, Norfolk 1 (2002) 392–396.

[34] J. Lacy, IEEE Nuclear Science Symposium Conference Record 1 (2006) 20–26.

[35] J. L. Lacy, A. Athanasiades, L. Sun, C. S. Martin, T. D. Lyons, M. A. Foss, H. B. Haygood, Nuclear Instruments and Methods in Physics Research Section A 652 (2011) 359 – 363. doi:10.1016/j.nima.2010.09.011 symposium on Radiation Measurements and Applications (SORMA) {XII} 2010.