ELSEVIER

LibCPIXE: A PIXE simulation open-source library for multilayered samples

C. Pascual-Izarra [a,b,*], N.P. Barradas [a], M.A. Reis [a]

[a] *Instituto Tecnológico e Nuclear, E.N. 10, 2686-953 Sacavém Codex, Portugal*
[b] *CMAM, Universidad Autónoma de Madrid, E-28049 Madrid, Spain*

Available online 11 May 2006

Abstract

Most particle induced X-ray emission (PIXE) data analysis codes are not focused on handling multilayered samples. We have developed an open-source library called "LibCPIXE", for PIXE data analysis. It is written in standard C and implements functions for simulating X-ray yields of PIXE spectra taken from arbitrary samples, including multilayered targets. The library is designed to be fast, portable, modular and scalable, as well as to facilitate its incorporation into any existing program. In order to demonstrate the capabilities of the library, a program called CPIXE was developed and used to analyze various real samples involving both bulk and layered samples. Just as the library, the CPIXE source code is freely available under the General Public License. We demonstrate that it runs both under GNU/Linux systems as well as under MS Windows. There is in principle no limitation to port it to other platforms.
© 2006 Elsevier B.V. All rights reserved.

1. Introduction

The PIXE technique constitutes an excellent tool for quantitative characterization of samples. This has been traditionally done with the aid of computer simulation and fitting programs, which are able to solve the *inverse* problem of determining the composition of the sample from an iterative fitting of a PIXE spectrum. In the currently available codes [1], the fitting routines pose limitations on the type of samples that can be analyzed in order to guarantee a reasonably non-ambiguous solution. This is the case, e.g. in GUPIX, where layered samples cannot contain the same element in more than one layer [2]. Such a situation is unsatisfactory, specially in fields like materials science where many interesting samples require a more flexible definition.

In order to address this problem, we decided to make use of more advanced techniques for inverse problem solving [3] for which we needed a way of simulating the X-ray yields from a given arbitrary sample under general experimental conditions. In this work, we present a collection of computer routines (the "LibCPIXE" library) that can be used to perform such simulations. The models and databases used in LibCPIXE (based on those from the DATT-PIXE code [4]) are discussed in the next section and the computational details are treated in Section 3. After that, some examples are presented of the performance of the library by comparing simulations and selected experimental results.

2. Simulation model

The primary objective of the LibCPIXE library is to calculate the characteristic X-ray emissions produced by the

---

* Corresponding author. Address: CFMAC, Consejo Superior de Investigaciones Científicas, IEM-CFMAC, CSIC, Instituto de Estructura de la Materia, C/ Serrano 113bis, 28006 Madrid, Spain. Tel.: +34 91 5616800; fax: +34 91 5859413.

E-mail address: carlos.pascual@iem.cfmac.csic.es (C. Pascual-Izarra).
URL: http://cpixe.sourceforge.net (C. Pascual-Izarra).

irradiation of a sample with an ion beam. The number of detected characteristic X-rays, N_{ij}, emitted by an element, i, due to an electronic transition, j, can be written for a layered sample as:

$$N_{ij} = \frac{\Omega}{4\pi} \epsilon_j N \sum_l \left[T_{jl} X_{il} \int_{E_{l,\text{out}}}^{E_{l,\text{in}}} \frac{\tau_{lj}(x(E))}{S_l(E)} \sigma_{ij}(E) \mathrm{d}E \right], \qquad (1)$$

where the l index accounts for each layer in the sample, Ω is the detection solid angle, ϵ_j is the efficiency of the detector for the energy of the j emission, N is the number of beam particles and X_{il} is the concentration of the i element in the layer l. T_{jl} is the transmittance for the j X-ray energy due to absorbers located between the surface of the layer l and the active area of the detector while $\tau_{lj}(x)$ accounts for the transmittance between the position, x, inside the l layer and its surface. Finally, S_l is the stopping force for the beam particles in the layer l, and σ_{ij} is the X-ray production cross section. Note that S_l, σ_{ij} and x depend on the energy of the particle, E and that the integral covers the range between $E_{l,\text{in}}$ and $E_{l,\text{out}}$, the energies of the particles when entering and exiting the l layer, respectively.

In the current LibCPIXE code, the X-ray production cross sections for protons are calculated using the ionization cross sections from the ECPSSR theory [5] with semi-empirical corrections from Paul [6] and Reis [7] for the K and L subshells, respectively, in combination with Coster–Kronig fluorescence yields [8] and the Scofield K and L line transition ratios [9,10]. Ions other than protons are not yet supported but implementing this should not be complicated since previsions were taken when coding this first version.

In order to calculate the transmittances, absorption coefficients for X-rays in the energy range from 1 to 30 keV have been obtained from [11].

As for the stopping forces (*aka* "stopping powers"), the ZBL85 values [12] are used by default and Bragg's rule is applied for compound targets. However, if the default values are not precise enough, any other values (including user-defined ones) can be easily incorporated.

LibCPIXE supports arbitrary samples (including thin, bulk, intermediate and layered specimens). In the case of non-thin samples, enhancement of the X-ray yield may occur due to photoionization by the X-rays emitted in any other part of the sample. The treatment of such effect (the so-called *secondary fluorescence correction*) is not trivial in the case of multilayered samples. A paper giving all the details about how this is treated in LibCPIXE is under preparation and will be published elsewhere.

The detector efficiency, ϵ, is obtained experimentally by analyzing known samples and using the code to determine ϵ_j for the main emission lines. The full efficiency curve is then obtained by interpolation on these experimental values.

It must be noted that, currently, LibCPIXE is focused on calculating the yields of the relevant lines of a PIXE spectrum but does not try to actually simulate a real spectrum and, hence, background is not calculated and no convolutions to simulate finite detector resolutions are made.

In order to compare the simulation results with real data, the experimental spectra can be processed to eliminate background, identify the peaks an extract their areas. Work is in progress to incorporate into LibCPIXE a tool for automating this process, but meanwhile, the publicly available code QXAS [13] can be used to that purpose.

3. Computational details

The LibCPIXE library, which was originally based on the DATTPIXE v5.3 code [4], is written in standard C. A Fortran 90 compatibility layer has also been developed so that existing Fortran programs can be linked against LibC-PIXE too.

Routines are included in LibCPIXE to facilitate the interfacing: sample and experimental parameter descriptions can be read from ASCII files or passed directly as function parameters. Similarly, results can be directly returned to the caller program and/or be written in ASCII files. The samples and the absorbers are defined in a layer-by-layer basis using either atomic or mass areal densities for the thicknesses.

To build a fast library has been a priority and this was achieved by using several coding optimization techniques. In the first place, all the databases are pre-cached and stopping force interpolation tables are built to save computation time. The thickness of the internal sub-layers required for numerical integration of Eq. (1) is not taken to be constant but is dynamically adjusted to guarantee small enough variations of τ_{lj}, S_l and σ_{ij}. Finally, dynamic memory allocation and intensive use of C pointers are done to improve efficiency.

The LibCPIXE library is easily modifiable and scalable thanks to its modular design and the fact of being open-source (it is freely available under the GPL license [14] from http://cpixe.sourceforge.net). A simple program called CPIXE has been developed to test and demonstrate the capabilities of the library. CPIXE links against LibC-PIXE and has been compiled and tested under both GNU/Linux and MS Windows (porting to other systems/platforms should not pose difficulties).

4. Examples with the CPIXE demonstration code

The CPIXE program has been used to simulate three example cases for which experimental spectra were acquired. The spectra used for these examples are: (a) 1090 MeV H^+ beam on an pure Au bulk target, (b) same beam on a stoichiometric Fe_2O_3 bulk sample and (c) 1220 MeV H^+ beam on a Mn_4Ir thin film (562×10^{15} at/cm^2) deposited over a Si substrate. The simulated and experimental yields are compared in Table 1. A further demonstration of the practical capabilities of the LibC-PIXE (beyond the limits of the CPIXE simple example code) is presented in [3].

From Table 1 it can be noted that the experimental results – deconvoluted from the raw spectra using QXAS

Table 1
Comparison of simulated and experimental X-ray yields

Sample	Line	Simulation	Experiment
Au	Au–L$_{\alpha_{1,2}}$	7720	7874
	Au–L$_{\beta_1}$	1880	2535
	Au–L$_{\beta_2}$	1546	1470
	Au–L$_{\gamma_1}$	323	499
	Au–L$_\ell$	325	394
	Au–L$_{\beta_3}$	155	186
	Au–L$_\eta$	46	69
	Au–L$_{\gamma_3}$	51	50
Fe$_2$O$_3$	Fe–K$_{\alpha_{1,2}}$	1.04×10^5	1.09×10^5
	Fe–K$_{\beta_1}$	15,079	14,891
Mn$_4$Ir/Si	Si–K$_{\alpha_{1,2}}$	1.76×10^6	1.77×10^6
	Mn–K$_{\alpha_{1,2}}$	41,158	41,473
	Mn–K$_{\beta_1}$	5929	6027
	Ir–L$_{\alpha_{1,2}}$	2511	2508
	Ir–L$_{\beta_1}$	634	665
	Ir–L$_{\beta_2}$	423	460
	Ir–L$_{\gamma_1}$	127	128
	Ir–L$_\ell$	119	122
	Ir–L$_\eta$	17	18

– are reasonably well reproduced by the simulations, specially the K$_{\alpha_{1,2}}$ and L$_{\alpha_{1,2}}$ lines. One would typically choose these lines for a fitting procedure in the case that LibCPIXE is integrated into an automated fitting program (as in [3]).

5. Conclusions

LibCPIXE is, to our knowledge, the first truly free [14] and open-source library to be made publicly available to the IBA community for PIXE simulations. It is our aim that developers of existing IBA simulation/analysis codes may consider to make use of this library to extend their codes into supporting the PIXE technique too. LibCPIXE is already functional – as demonstrated in this paper and in [3] – but more capabilities and extensions are being developed at the moment of writing this.

References

[1] S. Fazinić, Intercomparison of PIXE Spectrometry Software Packages TECDOC 1342, International Atomic Energy Agency (IAEA), 2003.

[2] J.A. Maxwell, W.J. Teesdale, J.L. Campbell, The Guelph PIXE software package II, Nucl. Instr. and Meth. B 95 (1995) 407.

[3] C. Pascual-Izarra, M.A. Reis, N.P. Barradas, Simultaneous PIXE and RBS data analysis using Bayesian Inference with the DataFurnace code, Nucl. Instr. and Meth. B, Proceedings of the IBA2005 conference, in press, doi:10.1016/j.nimb.2006.03.190.

[4] M.A. Reis, L.C. Alves, DATTPIXE, a computer package for TTPIXE data analysis, Nucl. Instr. and Meth. B 68 (1992) 300.

[5] W. Brandt, G. Lapicki, Energy-loss effect in inner-shell coulomb ionization by heavy charged particles, Phys. Rev. A 23 (1981) 1717.

[6] H. Paul, An analytical cross-section formula for K X-ray production by protons, Nucl. Instr. and Meth. B 3 (1984) 5.

[7] M.A. Reis, A.P. Jesus, Semi-empirical approximation cross sections for L X-ray production by proton impact, Atomic Data Nucl. Data Tables 63 (1996) 1.

[8] M.O. Krause, Atomic radiative and radiationless yields for K and L shells, J. Phys. Chem. Ref. Data 8 (1979) 307.

[9] J.H. Scofield, Exchange corrections of K X-ray emission rates, Phys. Rev. A 9 (1974) 1041.

[10] J.H. Scofield, Relativitic Hartree–Slater values for K and L X-ray emission rates, Atomic Data Nucl. Data Tables 14 (1974) 121.

[11] E.F. Plechaty, D.E. Cullen, R.J. Howeton, Tables and Graphs of Photon Interaction Cross Section from 1.0 keV to 100 MeV Derived from LLL Evaluated Nuclear Data Library, Report DLC-7, Lawrence Radiation Laboratory, Livermore, CA, 1975.

[12] J.F. Ziegler, J.P. Biersack, U. Littmark, Stopping and Ranges of Ions in Solids, Pergamon, New York, 1985.

[13] QXAS Seibersdorf Laboratories, IAEA, URL: Available from: <http://www.iaea.org/OurWork/ST/NA/NAAL/pci/ins/xrf/pciXRFdown.php>.

[14] GNU General Public License, URL: Available from: <http://www.gnu.org/licenses/gpl.html>.