CROWN: Applied tool for CARS-to-Raman spectrum decomposition

Ildar Galikov [a], Alytis Gruodis
[1] Faculty of Physics, Vilnius University, Saulėtekio 9, korp. 3, Vilnius, Lithuania

Received 2 Februar 2011, accepted 10 May 2011

Abstract. Novel advanced tool *CROWN* was created for CARS spectrum decomposition purposes: in order to extract Gauss- or Lorentz-shaped components containing Raman spectrum. *Win32* type application was created in object-oriented programming (OOP) manner using Visual Studio 8.0 package. C++ language was used for programming GDI interface using standard windows.h. library.

Citations: Ildar Galikov, Alytis Gruodis. *CROWN*: Applied Tool for CARS-to-Raman Spectrum Decomposition – *Innovative Infotechnologies for Science, Business and Education,* ISSN 2029-1035 – **2(11)** 2011 – Pp. 22-26.

Keywords: *CROWN*; CARS; CARS-to-Raman; Spectrum decomposition.
PACS: 42.65.Dr
Short title: *CROWN*: CARS-to-Raman.

Introduction

Coherent anti-Stokes (AS) Raman scattering (CARS) technique [1] is popular as a unique macroscopic as well as microscopic [2] tool in material sciences [3], biology [4], biophysics [5], medicinal physics [6], etc. The benefit of CARS technique could be described as sensitivity to intramolecular changes and versality due to χ^3 behaviour [7]. CARS signal according to the experimental schema is blue-shifted from laser excitation frequencies – it means that CARS signal could be easily detected in the presence of strong luminescence [8].

Visualization of the digital data containing spectral dependencies belongs to the one of the most important computer tasks in computational physics. Information graphics software allows to users the fast manipulation in order to create the suitable visualization form from equipment-provided data-set. Computer-algebra systems (such as Maple [9] , MathCad [10] and Matematica[11]), numerical-software systems (LabView [12] , MatLAb [13] , SciLAb [14]) are quite useful for creating the two-dimensional or three-dimensional space projection containing object-oriented dependencies.

However, sometimes all functional possibilities of mentioned packages are not partially or fully required due to specificity of the task. Reordering and pre-manipulation (including sorting, fitting etc.) – such non-trivial operations are possible using Origin [15] package which, however, requires many non-automated operations. Using of Origin tools - worksheets and data fitting wizards – complicates the fast search of solution, but is very useful for slowly routine operations only. This work is devoted to create the novel fast tool for express estimation of CARS and Raman spectrum content without necessity to use routine, time-wasting fitting procedure in the some wizards.

[a]Corresponding author, email: *Ildar.Galikov@gmail.com*

1. Physical behaviour of CARS spectrum

Project *CROWN* was created as original graphical user-friendly tool for fast Gauss/Lorenz-shaped decomposition of CARS spectra $S(\omega)$ into Raman components (CARS2Raman). Adapted formula from Ref. [16] was used as follows:

$$S(\omega_{AS}) = [I_{BCG}(\omega_{AS}) + I_R(\omega_{AS}) \cdot \cos\phi]^2 \tag{4}$$

I_R and I_{BCG} represent the terms of Raman bands and backgrounds, respectively. Due to experimental conditions, such two terms represent different origin (changes of induced dipole moment for I_R(pure nuclear movement) and changes of nuclear dipole moment in the fast electrostatic field for I_{BCG} (nuclear movement in the electron environment), respectively). Factor $\cos(\phi)$, when ϕ=0 deg or ϕ=180 deg, represents the increasing or decreasing of intensity S. Angle ϕ represents the difference of phases between Raman term and background term.

Single Raman bands I_R could be described using classical Gauss I^G, Lorenz I^L or Voight I^V dependencies according to physical model of the task:

$$I^V = x \cdot I^G + (1 - x) \cdot I^L, x \subset [0 \div 1] \tag{5}$$

$$I^G = \frac{A_0}{\sigma}\sqrt{\frac{2}{\pi}}\exp\left(-\frac{2(\omega-\omega_0)^2}{\sigma^2}\right) \tag{6}$$

$$I^L = \frac{2A_0}{\pi}\frac{\sigma}{4(\omega-\omega_0)^2 + \sigma^2} \tag{7}$$

A_0 represents the altitude – intensity at ω_0, and σ represents the halbwidth of band.

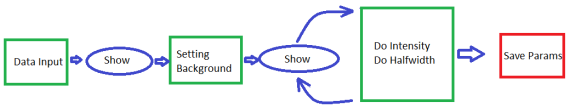

Fig. 1. *CROWN*. Setup of package.

2. Structuric scheme of *CROWN*

CROWN is written using *object-oriented programming* (OOP) language well known JAVA [17]. Java programming language was selected, due to it's portability between different operation systems. Different classes of program are responsible for data storage, calculating, visualization and *Graphical User Interface* (GUI). Therefore it creates the possibility for rapid updating of program.

Fig. 1 shows the principal scheme of the program. Firstly, input data as two-dimensional distribution is read and visualized. After that, user sets the background values. Next, user changes half-width or intensity of bands, in order to fit experimental and theoretically calculated from bands and half-width CARS spectra. When the purpose is achieved, bands parameters are writing to file. Used libraries and their description are shown in Table 1. The most important classes and sub-classes are shown and described in Table 2.

2. Description of graphical User Interface

Program Crown is written to give a possibility to do easy, comfortable and fast CARS spectra decomposition. CARS spectrum decomposition proceeds from opening data file with experimentally calculated values, presented as X, Y matrix.

Fig. 2 shows file choosing dialog. The same result could be achieved by pressing "alt+F3" keyboard keys combination. After reading file, user must select background values, by pressing mouse at two points. The background is a linear function that goes through selected two points.

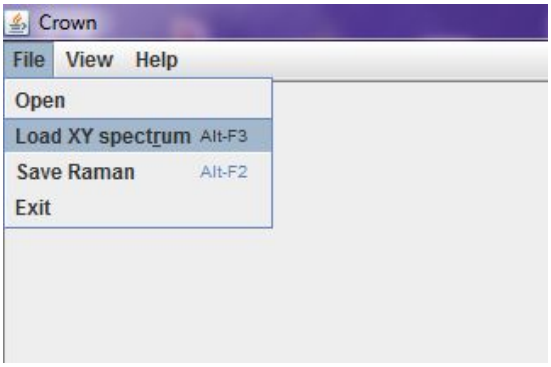

Fig. 2. *CROWN*: initial dialog for file input/output.

Table 1. List of used JAVA libraries.

GUI objects and their extended options
javax.swing.*
java.awt.*
javax.swing.border.*
javax.swing.JFileChooser
javax.swing.filechooser.FileNameExtensionFilter
javax.swing.event.DocumentListener
javax.swing.event.DocumentEvent
java.awt.event.*
File data management: Input/Output
java.io.BufferedInputStream.*
java.text.DecimalFormat
java.util.Scanner
java.io.File
java.io.FileInputStream
java.io.FileNotFoundException

Fig. 3-10 illustrates visualization and fitting processes. Fig. 3, 4, 5 show the whole background defining process. Green line shows the place, where the band would be added, if the left mouse button be pressed. Fig. 6 shows experimental CARS spectra data, selected background and 3 bands. Bands can be deleted by pressing right mouse button. The closest to the mouse band (by x coordinate) will be deleted.

Table 2. The most important classes and sub-classes.

Class name	Sub-class name	Description
WindowClass		GUI objects creation, response to user interactions
	MouseMoved	Mouse motion events
	MouseClicked	Mouse pressed events
	JInfoPanel	Output of important information to the screen(the right part of the workspace)
	PaintPanel	Spectrum data visualization, alongside other painting work.
	MenuLoadXY	Reading data from file.
	SaveAs	Writing data to file.
Bands		Data of a list of bands, for Raman and CARS spectrums calculation
Spectrum		Spectra (CARS, RAMAN, calculated CARS) data.
ChooseDialog		Dialog window, for changing bands data(x,y, phase).

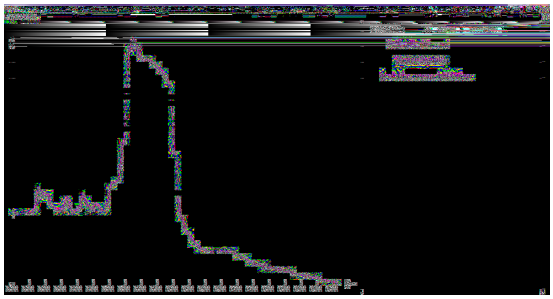

Fig. 3. Selecting of the first point of the background. Red curve corresponds the input file values.

Fig. 4. Selecting the second point of background.

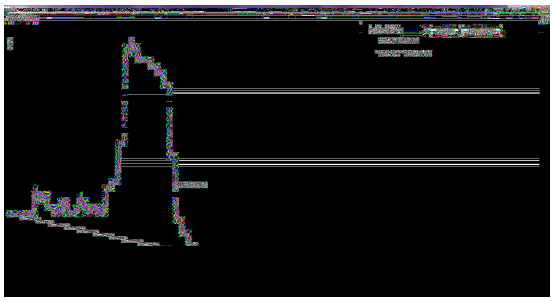

Fig. 5. After selecting both background points. At the right side of the workspace is written background equation. The black line corresponds to the background.

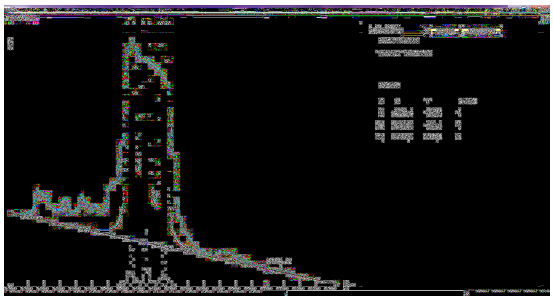

Fig. 6. Blue lines corresponds to the selected bands. Their values are shown in the information part of the workspace. Pink curve corresponds to theoretically calculated CARS spectrum. Black curve corresponds to the Raman spectrum.

Fig. 7 shows the result after pressing the right mouse button on the right side of all bands, presented Fig. 6. There is a possibility to change x, y and phase of any bond. To do that, user could either press middle mouse button(this selects the nearest band) or press the left button directly on the band (blue line). 8 picture shows the dialog window, of changing values of the middle band.

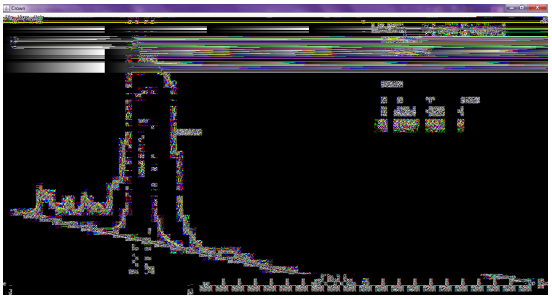

Fig. 7. Result after pressing right mouse button on the right side of all bonds.

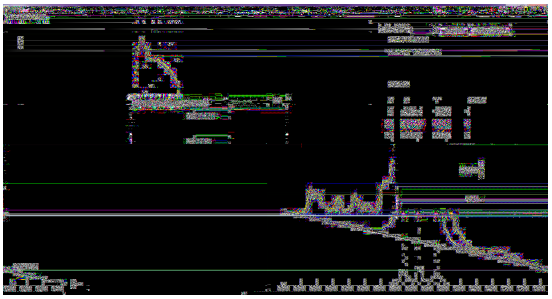

Fig. 8. The second band values changing. At the information part of the workspace , orange colour shows the band values are going to change.

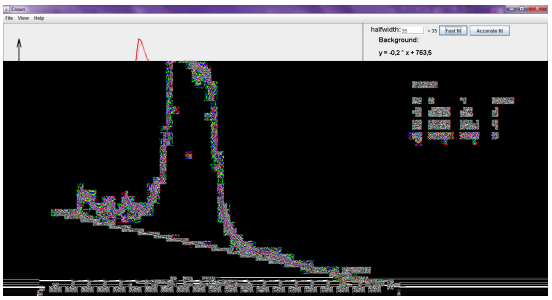

Fig. 9. results, after setting half-width = 35, and pressing “fast fit” two times.

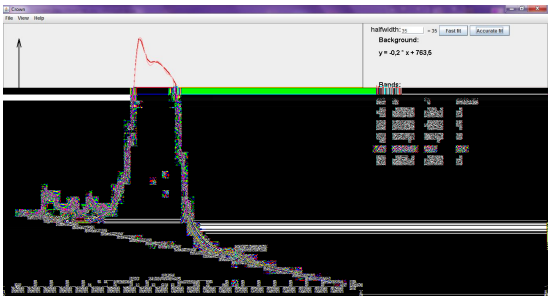

Fig. 10. Theoretical and experimentally calculated CARS spectrums after fitting.

Band values could be fitted by pressing “fast fit”, “accurate fit” buttons or changing halfwidth values. 9 picture shows the results, after setting half-width = 35 and pressing “fast fit” two times. In purpose to get less difference between experimentally and theoretically calculated CARS spectrums, use “accurate fit” button. 10 picture shows the better fitting, after adding two more bands and using “accurate fit” algorithm.

Raman spectrum information could be exported by pressing "alt+F2" key combination, or by selecting the corresponding menu. 11 picture shows exported data. The first column shows X values, the second – Raman spectrum values. Other columns have values of Raman spectrum components each component place an accentual row in the Raman spectrum.

3. Fitting algorithm

It is important to understand, that CARS spectra changes, whenever band's phase or intensity are changed. The fitting is done, by changing Y values of bands, depending on difference between experimental and theoretically calculated CARS spectre values. It would take a huge amount of computer resources to recalculate CARS and Raman spectrums after changing any bands value. To solve that problem were created two fitting algorithms – fast fitting and accurate fitting. Algorithms start to work after pressing „do fit" or „accurate fit" buttons.

Fast fitting algorithm. This alhorithm is dedicated to fit approximately values. By running this algorithm every bands value changing by 5%, depending on difference of theoretically calculated value and value from inputted file at the bands X value. Theoretical CARS and RAMAN spectrums recalculated, after changing every bands value. This process is repeated thirty times.

Accurate fitting algorithm. This algorithm is dedicated to make theoretically calculated CARS spectrum as much closer to experimentally obtained spectrum value. It is obvious that bands with higher value have stronger influence on whole spectrum, so all bands are sorted by their value. CARS spectra recalculated right after changing any bands Y value. Therefore each other consequent calculations include corrective from all others. That is why this algorithm is much more accurate. Beginning from band with highest Y value and finishing with the lowest, band's value is changing by 1% and Raman and CARS spectrums are recalculated. This process is repeated ten times.

Conclusions

Novel tool was created for express estimation of CARS and Raman spectrum content without necessity to use routine, time-wasting fitting procedure in the some wizards.

Acknowledgements

This work was partly supported by the Lithuanian State Science and Studies Foundation (project B-07013 "KARS-KOPAS").

References

1. G. Knopp, Iddo Pinkas and Yehiam Prior. Two-dimensional time-delayed coherent anti-Stokes Raman spectroscopy and wavepacket dynamics of high ground-state vibrations. – *Journal of Raman spectroscopy* 31 (2000) 51–58.
2. Mamoru Hashimoto, Tsutomu Araki, and Satoshi Kawata Multi-focus coherent anti-Stokes Raman scattering microscopy. – *Microsc. Microanal.* 9(Suppl 2) (2003) 1090-1091.
3. Josefa R Baena1 and Bernhard Lendl. Raman spectroscopy in chemical bioanalysis. – *Current Opinion in Chemical Biology* 8 (2004) 534–539.
4. M. O. Scully, G. W. Kattawar, R. P. Lucht, T. Opatrny, H. Pilloff, A. Rebane, A. V. Sokolov, and M. S. Zubairy. Fast CARS: Engineering a laser spectroscopic technique for rapid identification of bacterial spores. – *PNAS* Vol. 99 N.17 (2002) 10994–11001.
5. Haifeng Wang, Yan Fu, Phyllis Zickmund, Riyi Shi, and Ji-Xin Cheng. Coherent Anti-Stokes Raman Scattering Imaging of Axonal Myelin in Live Spinal Tissues. – *Biophysical Journal* Vol.89 (2005) 581–591.
6. Hilde A. Rinia, Mischa Bonn, Erik M. Vartiainen, Chris B. Schaffer, and Michiel Müller. Spectroscopic analysis of the oxygenation state of hemoglobin using coherent anti-Stokes Raman scattering. – *Journal of Biomedical Optics* Vol.11-5 (2006) 050502-1,3.
7. Schluker S., Schaeberle M.D., Huffman S.W., Levin I.W. Raman microspectroscopy: a comparison of point, line, and widefield imaging methodologies. – *Anal Chem* 75 (2003) 4312-4318.
8. Arenas JF, Lopez-Tocon I, Centeno SP, Soto J, Otero JC: How a resonant charge transfer mechanism determines the relative intensities in the SERS spectra of 4-methylpyridine. – *Vib Spectrosc* 29 (2002) 147-154.
9. Maple13, math and engineering software, <http://www.maplesoft.com/], accessed 2009 11 15.
10. MathCad, engineering calculation software, <http://www.ptc.com/products/mathcad/], accessed 2009 11 15.
11. Matematica 7.0.1, computational software program for technical computing, [http://www.wolfram.com/products/mathematica/latestversion/], accessed 2009 11 15.
12. LabView, graphical programming for measurement and automation, <http://www.ni.com/labview/], accessed 2009 11 15.
13. MatLAb, interactive environment to perform computationally intensive tasks, [http://www.mathworks.com/products/matlab/], accessed 2009 11 15.
14. SciLAb 5.1.1, open source platform for numerical computation, <http://www.scilab.org/], accessed 2009 11 15.
15. Origin 8.1, analysis tools for Spectroscopy, Statistics, Signal Processing, 3D Surface Fitting and Image Processing, [http://www.originlab.com/], accessed 2009 11 15.

16. Erik M. Vartiainen, Hilde A. Rinia, Michiel Müller, Mischa Bonn. Direct extraction of Raman line-shapes from congested CARS spectra. – *Optics express* Vol.14 N.8 (2006) 3622.
17. Java SE Development Kit 6 (JDK 6) 25 update.
23. SP, PerkinElmer IR DataManager format, [http://las.perkinelmer.com] , accessed 2009 11 15.
24. JCAMP-DX, spectroscopic data exchange format, [http://www.jcamp.org], accessed 2009 11 15.
25. OPUS viever 6.5, Bruker Optics spectroscopic software, [http://www.brukeroptics.com], accessed 2009 11 15.