Software update

Update 1.5 to “Takin: An open-source software for experiment planning, visualisation, and data analysis”, (PII: S2352711016300152)

Tobias Weber
Physik-Department E21, Technische Universität München, James-Franck-Str. 1, 85748 Garching, Germany

A R T I C L E   I N F O

Article history:
Received 4 May 2017
Accepted 8 June 2017

Keywords:
Triple-axis spectroscopy
Instrument control
Reciprocal and real space visualisation
Brillouin zones
Instrumental resolution

A B S T R A C T

We present an updated version of our inelastic neutron scattering software package Takin, which is a programme for neutron triple-axis experiment planning and evaluation. The new version features several additional programme modules mainly concerning three-dimensional calculations and visualisations. In addition, existing modules have been improved and extended.

© 2017 The Author. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

Several updates to the free and open-source software (FOSS) Takin [1,2] have been made over the course of the last year. Apart from general performance optimisations and improvements of the existing programme modules, new features have been introduced. Of these features, we present two selected ones in the present short paper.

2. 3D Brillouin zones

The previous version of our software included code to calculate a two-dimensional (2D) slice of the first Brillouin zone relevant...
Fig. 1. Example calculation of the first Brillouin zone in the fcc crystal MgV₂O₄. (a) The two-dimensional cuts of the Brillouin zone in the (hk0) plane can only be correctly calculated using the full three-dimensional information. In a 2D calculation, the diamond-shaped borders of the zone along the [110] directions – marked with (c) – would be missing. (b) The full results of the 3D calculations are displayed using OpenGL. Here, the red points mark positions of high symmetry. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Acknowledgements

This work was supported by the German Research Foundation (DFG) and the Technische Universität München within the funding programme Open Access Publishing. We are mainly indebted to Peter Böni and Robert Georgii for their continuing support of these scientific computing projects. Additionally, we want to thank Petr Čermák, Astrid Schneidewind, Jitae Park, Greg Tucker, and Bertrand Roessli for helpful discussions and for reporting bugs in the previous version of the software. Many thanks go to Alain Filhol for his continuous help on improving the OS X build. Also thanks to Petr Čermák for his website at MLZ, which hosts the binaries.

References


3. Scripting interfaces

In its resolution convolution code, Takin uses externally supplied scripts for the calculation of the dynamical structure factor \( S(Q, E) \). In addition to Python [5], we now support the novel scripting language Julia [5] for this task. As Julia is just-in-time compiled, it considerably speeds up the calculation of the convolution integral.

In order to circumvent threading limitations in the Python interpreter and for stability reasons, both the Python and Julia scripting interfaces are now separated from the main programme. \( S(Q, E) \) script modules are now spawned in their own process space and communicate with Takin using a shared-memory messaging system.