Developing a graphical user interface for the FELIX suite of electron diffraction visualisation codes

The symmetry of a crystal normally determines its functional properties. This is equally true on the nano-scale as it is at the macro-scale. Whilst for bulk material the structure and symmetry can routinely be solved by X-ray diffraction, there is no comparable technique for nanostructured materials. Electron diffraction has the required nano-scale resolution and sensitivity, but overlapping data from multiple diffracted beams has limited its use to date. We recently demonstrated that computer control of beam tilt and image capture in a conventional transmission electron microscope (TEM) can be used to overcome this problem, quickly providing very rich diffraction datasets [1]. The technique requires no new hardware, no more expertise than conventional electron diffraction and takes only a few minutes to acquire and process a complete data set. The vast improvement in the detail and quality of data that results from this procedure allows rapid and unambiguous determination of local symmetry at the nanometre scale. However, the method is most productive when using in conjunction with computational simulation of expected diffraction results based on the structure and symmetries of the input crystal. We have recently developed such a computer code, based on what is known as the Bloch-wave method. The code is fully parallelised for MPI-type architectures and has been shown to have linear speed-up. It consists of two parts, (1) FELIXSIM for simulating diffraction patterns from a given input crystal structure (described via standard crystal information files .cif) and (2) FELIXREFINE which uses FELIXSIM iteratively together with an minimisation algorithm in order to refine a .cif file based on experimental diffraction images. Both codes are available as open source via github [2].

The goal of the MSc project will be to write a graphical user interface (GUI) to both FELIXSIM and FELIXREFINE. This GUI should work across various compute platforms and best be written using a modern scripting language such as Python. While the emphasis of the project is most for a graphical input system, a system which additionally visualises the output images would be great as well. This project will give you a flavour of numerical techniques in computing and physics, and so will be ideal for a student contemplating research in this area. A strong background in programming is recommended. For more information please contact Rudolf Roemer (R.Roemer@warwick.ac.uk) or see http://www.warwick.ac.uk/go/DisQS.

2. https://github.com/RudoRoemer/Felix