https://www.reflectometry.org/

Analysis Working Group

ORSO Annual General Meeting
2021-06-14

Chairs: Brian Maranville and Andrew Nelson

Introduction

Goals: establishing cross-facility collaboration on

- standardized data analysis “best practices”
  - Strategies for multi-modal (X-ray and neutron, etc) fitting
- making it easier to build and use reflectometry analysis software
  - Available standard test problems to verify calculations
  - Calculation kernels available to the community
  - Development of shared model language (easier swapping between programs + reproducible research)
  - Facilitating use of shared computation platforms: Binder, Google Colab, etc

Multi-modal analysis

E.g. Fitting X-ray and neutron reflectometry together; the techniques have different sources of error, and/or different weights on the errors

- X-ray reflectometry has much higher count rate, lower statistical (Poisson) error
- Neutron instruments (typically) are longer and have better-known angular resolution

See presentation in Analysis session Thursday (21:00 2021-06-17) by A. Caruana:

“Fitting laboratory XRR data – to use with NR”

Still to do:

- Ellipsometry (uncertainties?)
- Detailed catalog of systematic + statistical uncertainties (per-instrument)

Standard test problems

These are made available for verification of the calculations in software packages

https://github.com/reflectivity/analysis/tree/master/validation

Automated scripts run validation tests on major packages on a scheduled basis

https://github.com/reflectivity/analysis/blob/master/.github/workflows/validation.yml

(e.g. on 2021-06-10 this system caught an error in one of those packages, and it was fixed immediately by the package maintainer)

Still to do:

- Polarized validation tests

Calculation kernels

The kernels (lowest-level calculation implementation) for multiple packages are gathered for *Refl1D* and *refnx*:

https://github.com/reflectivity/analysis/tree/master/kernel

Implemented in C, and for refnx in cython and pyopencl

Community has requested wrappers for common languages, to help with development of further packages.

$$c_n = \begin{bmatrix} \exp(\beta_n) & r_{n,n+1}\exp(\beta_n) \\ r_{n,n+1}\exp(-\beta_n) & \exp(-\beta_n) \end{bmatrix}$$

The resultant matrix is defined as the product of these characteristic matrices

$$M = \prod_n c_n$$

from which the reflectivity is calculated as:

$$R = \left| \frac{M_{10}}{M_{00}} \right|^2$$

From https://en.wikipedia.org/wiki/Transfer-matrix_method_(optics)

Shared model language

- Simplification of cross-package testing and validation
- Portability:
  - Knowledge (users learn one thing)
  - Tools (developers can work work together)
  - Models themselves
- Standardised definition of “what is a model”
  - Can be referenced or attached to publications
  - Discoverable/reusable

First steps:

https://github.com/reflectivity/analysis/blob/master/model_language/refl1d.schema.json

See presentation 22:00 Thursday, 2021-06-17 (B. Maranville)

Lowering the barrier: online tools for learning

Resources available for providing interactive tutorials and demonstrations with real, live code

- Binder.org
- Google colab

Example binder setup:

https://github.com/reflectivity/analysis/tree/master/tutorials

Contributing

Efforts are managed through github: https://github.com/reflectivity/analysis

Tasks are converted to “issues”

Anyone can join and work on issues or add new ones

We’re going to need new chairs someday!

See you on Thursday!