

## Appendix C: Application Programs

### APPLICATIONS FOR WBS 6 - DIFFRACTION

#### 6.1 *DiffLAB*

This will be a modular configurable structure modeling code, as opposed to traditional monolithic structure refinement and modeling codes. Monolithic applications are not flexible enough for modern materials structure problems – the classes of models employed are becoming more diverse and structural features often require specialized software implementations. Increasingly we need a model that depends on the specific problem under study: e.g., for a molecular system that is best expressed as a  $Z$ -matrix, or for a discrete nanoparticle with no boundary conditions, a periodic system, and so on. Incorporating other types of structural data such as EXAFS and x-ray data in the analysis is also important for DiffLAB. We may want to employ a fast local-search regression algorithm such as Levenberg-Marquardt, or increase convergence using a global Monte-Carlo or genetic algorithm, or in general combine both approaches in a basin-hopping scheme. We seek the flexibility to adapt the modeling software to the needs of a particular problem.

DiffLAB will allow a modeling code to be configured at run-time from components such as function calculators (that calculate different data spectra), regression algorithms and models. The target function being optimized can be specified by the user by adding terms for more than one function calculator depending on the data available. DiffLAB will be extensible: as new opportunities and methods arise new modules can be written in the future. DiffLAB will be an application that makes use of the components in DiffPy, the library of diffraction component modules, and the DANSE libraries in general, to build a configurable, operating, regression modeling program on the fly. This will allow expert users to quickly and straightforwardly to build customized regression codes.

#### 6.2 *SrRietveld*

SrRietveld will be a complete package for most of the tasks currently supported in existing Rietveld programs, but with the speed to match data acquisition rates from POWGEN3 (this will require distributed operation), extensibility to grow in the future to meet future needs, and the ease and speed of use that has been demonstrated by the PDFgui application. The precise requirements are under development now, but an important feature will be regression fitting directly to parametric equations such as expected temperature dependences of multiple data-sets. It will also have versatile built-in plotting and structure visualization capabilities.

#### 6.3 *SrReal*

This will have similar requirements to SrRietveld but will be an application for carrying out fits to real-space PDF data and total scattering data using Monte-Carlo approaches to obtain local and nanostructure information.

## APPLICATIONS FOR WBS 7 - ENGINEERING DIFFRACTION

*7.1 Data Analysis*

This application will generate information crucial for engineering calculations, such as orientation dependent lattice strain, peak broadening and texture from neutron diffraction data. It will use Rietveld (full-pattern) analysis to fit a crystallographic model to diffraction data or single peak fitting to obtain  $hkl$ -specific lattice strain and peak broadening. It will then employ appropriate averaging schemes (e.g., based on mechanics models) and calculate specimen-specific engineering data such as strain tensors and orientational distribution functions. Some data analysis functions will be performed in real time for comparison to the expected outcome. It will then help the user find new strategies if results and expectations differ significantly.

*7.2 Mechanics Modeling I: Finite Element Analysis*

*purpose:* This application will first perform mechanics modeling of materials using the ABAQUS finite element software. It will then allow comparison of model predictions with experimental data. In this comparison an optimization analysis will be performed so that material parameters can be refined to fit the data.

*input:*

- Lattice strain data from diffraction (after Rietveld refinement)
- Macro strain data from extensometer (or strain gauges)
- Macro stress data from load cell and sample dimensions
- Material parameters (elastic constants, dimensions, strength, etc.)
- Sample models: Sample geometry: Cylindrical, dog-bone shaped, compact tension, weld
- Microstructure morphology: Unidirectional fibers, spherical particles, concentric single fiber, laminate composite

*output:*

- Phase dependent and macroscopic stresses/strains
- Optimized material parameters (e.g., plastic flow curve variables) after comparison with data

*methods:*

- FEA using ABAQUS as a Pyre component
- Optimization algorithms: leastsq, fmin, fmin\_powell, genetic and artificial neural networks

*7.3 Mechanics Modeling II: Self-Consistent Analysis*

This application will concentrate on mechanics modeling of materials using the EPSC self-consistent model (SCM) software. It will also allow comparison of model predictions with experimental data. In this comparison an optimization analysis will be performed so that material parameters can be refined to fit the data. A similar approach will also be followed with other SCM codes, e.g., one developed at ISU for ferroelectrics. Finally, where it is appropriate, a comparison with FEA predictions will be performed.

#### 7.4 *Experiment Design and Simulation*

This application will guide the user in planning an optimum experiment by (i) collecting information on specimens and scientific goals; (ii) simulating the responses of instrument and specimen based on conditions defined by the user; (iii) performing various sensitivity studies to optimize experimental parameters; (iv) offering expert advice to the user based on past data and experience in the field.

### APPLICATIONS FOR WBS 8 - SMALL ANGLE NEUTRON SCATTERING

#### 8.1 *Model-Independent Analysis*

This application will read 1D or 2D reduced SANS data and allow various manipulations by the user including simple math, Inversion to the pair density distribution function [ $P(R)$ ], so called ab-initio fitting, and include simple linear analysis (e.g. fractal power law ( $\log(I)$  vs  $\log(Q)$ ), Guinier ( $\log(I)$  vs  $Q^2$ ), Debye ( $I$  vs  $Q^{-2}$ ), etc), peak shape analysis, correlation length analysis, etc. Batch mode operation for parametric analysis (e.g. peak intensity as a function of time) will also be included.

#### 8.2 *Model Fitting Analysis*

This application will allow the user specify objects in a 3D modeling space, either rotationally averaged or partially oriented and produce 2D and 1D scattering patterns. The Application will provide simple models with analytical functions to produce the scattering patterns as well as allow the user to build more complex low-resolution models or import structures from PDB files which can then be inverted to  $q$  space. The objects may contain pieces with different scattering length densities and portions of the structures may be manually moved and oriented. Optimization against 1D or 2D reduced data will include standard parameter optimizations as well as constrained conformational searches. Some capabilities for exploring interaction potentials through  $g(r)$  will also be included.

#### 8.3 *Experimental Planning Tools*

Given source and instrument parameters, this application will allow simulation of the raw data collected as a function of experimental parameters (e.g. time, sample thickness, and size) including some system dependent background such as incoherent scattering from the sample. This will make use of the models developed in the previous applications and convolute them with the instrument specific parameters (e.g. beam flux). MC simulation of the instrument may also be possible.

### APPLICATIONS FOR WBS 9 - REFLECTOMETRY

#### 9.1 *Reduction*

*purpose:* Transform the data into scattering intensity in  $Q_x$ - $Q_z$ , allowing observation and control of every step in the process. Further reduction to intensity versus  $Q_z$  allows the data to be fit with traditional reflectometry fitting programs. If enough

information is available, phase determination and direct inversion of the complex amplitude can be performed. Comparisons between related datasets will be part of the general viewing capabilities of the reduction application.

*input:* NeXus files containing - sample data - intensity - detector efficiency

*input:*  $Q$  resolution - background selection

*output:*  $R(Q_z)$ ,  $R(Q_x, Q_z)$ ,  $r(Q_z)$ ,  $\rho(z)$

*method:* detector corrections - rotation - dead time - pixel width

*method:* monitor corrections - dead time - rebinning - intensity scaling

*method:* monitor normalization

*method:* specular/background decomposition from detector image

*method:* back reflectivity corrections - absorption - absolute  $Q$

*method:* polarization efficiency estimation and correction

*method:* footprint corrections for scanned measurements with fixed slits

*method:* phase reconstruction and direct inversion

*method:* contrast plots for comparing data sets

### 9.2 1-D Analysis

*purpose:* Fit specular reflectivity data to 1-D models of scattering length density, absorption, magnetic scattering length density and angle. Dynamic observation and control of the search space is important since reflectometry is a poorly defined inverse problem.

*input:*  $R(Q_z)$

*input:*  $\text{real}(r(Q_z))$  if available

*output:*  $\rho(z)$ ,  $\mu(z)$ ,  $P(z)$ ,  $\theta(z)$

*output:* model parameters such as SLD of the layer  $\rho(k)$ , roughness of the layer  $\sigma(k)$ , volume fraction, *etc.* depending on the model

*method:* modeling of nuclear and magnetic parameters

*method:* constrained assignment of parameters shared amongst models

*method:* automatic adjustment of parameters using combinations of global optimization (genetic algorithms, simulated annealing, restarts) and local optimization (Nelder-Mead, Levenberg-Marquardt)

*method:* uncertainty estimation for parameters

### 9.3 3-D Analysis

*purpose:* Fit off-specular reflectivity data to 3-D models of scattering.

*input:*  $R(Q_x, Q_z)$

*output:*  $\rho(\bar{z})$ ,  $\mu(\bar{z})$ ,  $P(\bar{z})$ ,  $\theta(\bar{z})$

*output:* model parameters depending on the model

*method:* modeling of nuclear and magnetic parameters

*method:* constrained assignment of parameters shared amongst models

*method:* automatic adjust of parameters using global and local optimization

*method:* uncertainty estimation for parameters

## APPLICATIONS FOR WBS 10 - INELASTIC SCATTERING

*10.1 Reduction*

*purpose:* Reduction will take experimental data files in NeXus or other formats from Direct Geometry Chopper Spectrometers, including LRMECS, PHAROS, ARCS, and SEQUOIA, and transform the data into scattering intensity, with outputs in physical units such as barns, meV and  $\text{\AA}^{-1}$ . Standard corrections for instrument and background will be performed. The functionality of Reduction will include the capabilities of the IDL Pharos code, and the basic functions of Open Genie used for data reduction on LRMECS. Reduction is available today as release 1.1. A release 1.2 of Reduction will also include multiphonon and multiple scattering corrections. These may be incorporated as a second package that can be run independently of Reduction.

*input:* NeXus (or other formats) data files containing - sample data - empty can - vanadium calibration - flux monitor - time independent background - instrument characteristics such as moderator-sample distance, detector positions

*input:* data file selections - output directory selection - detector masks - parameters of axes of output histograms - parameters for time-independent background selection and vanadium calibration - UB matrix information

*output:* selection of neutron-weighted  $S(Q)$ ,  $S(\vec{Q})$ ,  $S(E)$ ,  $S(Q, E)$ ,  $S(\vec{Q}, E)$

*output:* data files in mslice format

*output:*  $g_{\text{nw}}(E)$  for phonons or magnons

*output:* calibration results (detector efficiencies)

*method:* Standard Corrections: - detector efficiency - auto determination of additional masks - incident energy - background - empty can - black absorber - normalization

*method:* single crystal reduction (TBA)

*method:* transformations to  $S(Q)$ ,  $S(\vec{Q})$ ,  $S(E)$ ,  $S(Q, E)$ ,  $S(\vec{Q}, E)$ , with input on  $\vec{Q}$

*method:* scattering corrections - multiple scattering - multiphonon scattering (incoherent) to give  $g_{\text{nw}}(E)$

*method:* data provenance

*10.2 Dynamics Modeling*

*purpose:* Dynamics Modeling will optimize the parameters of a physical model of sample dynamics to fit reduced data. The optimized model can then provide other properties of the sample, such as the scattering after correction for neutron weights, or the separation of spin and phonon scattering.

Functionalities planned for this application include:

- Fits to experimental data of results from a Born – von Kármán lattice dynamics model or a Heisenberg spin dynamics model,
- Calculating experimental data with a Van Hove function  $G(r, t)$ , or a magnetic susceptibility function  $\chi(q, E)$ ,
- Neutron weight correction (magnetic or nuclear),
- Separation of nuclear and magnetic scattering by  $Q$ -dependence.

*input:* neutron-weighted  $S(E)$ ,  $S(Q, E)$ ,  $S(\vec{Q}, E)$   
*input:* initial parameters for model (eventually from database)  
*input:* selection of optimizer and its numerical criteria  
*output:* optimized parameters of the model  
*output:*  $g(E)$ , for phonons or magnons, corrected for neutron weighting  
*output:*  $E(\vec{q})$ ,  $S(Q, E)$ ,  $S(\vec{Q}, E)$  corrected for neutron weighting in incoherent approx.  
*method:* fits of model to experimental  $g(E)$   
*method:* fits of model to experimental  $S(Q, E)$   
*method:* neutron weight correction (magnetic or nuclear)  
*method:* neutron weight correction (magnetic or nuclear)  
*method:* separation of nuclear and magnetic scattering  
*method:* models for forward calculations - BvK lattice dynamics model - Heisenberg spin dynamics model - Van Hove  $G(r, t)$  - Magnetic susceptibility function  $\chi(q, E)$   
*method:* CLIMAX  
*method:* optimizers for model parameters - Levenberg-Marquardt - Powell - Simulated annealing - TBA

### 10.3 Experiment Simulation

*purpose:* Experiment Simulation will include sample scattering within Monte Carlo simulations of instruments, and generate NeXus files of simulated neutron scattering. This package is extending the simulation framework and pyre-Mcstas to inelastic scattering, developed in WBS 5. Extensions to inelastic scattering include the development of inelastic scattering kernels and molecular dynamics modeling.

*input:* instrument geometry and configuration  
*input:* sample characteristics - geometry (shape) - physics (scattering kernel selections, scattering kernel characteristics) - tentative inputs to scattering kernels using dynamic model: - BvK - Heisenberg -  $G(r, t)$  function -  $\chi(q, E)$  function - TBA: Molecular Dynamics (may be too demanding on computing resources, when combined with instrument simulation)  
*input:* sample environment - temperature - magnetic fields - furnace - sample holder (container)  
*input:* user parameter selections for simulations - number of neutron packets - size of neutron packet - computing nodes selection (if parallel) - output directory  
*output:* NeXus file  
*output:* simulated sample properties from dynamics model and optimization  
*output:* simulated instrument properties - flux as a function of  $Q, E$  - resolution in  $Q, E$   
*method:* DANSE instrument simulation - DANSE new - McStas component - Vitess component  
*method:* neutron recorder with data manager  
*method:* phonon scattering from  $E(\vec{q})$   
*method:* phonon scattering from  $G(r, t)$   
*method:* phonon scattering from BvK  
*method:* phonon scattering from molecular dynamics  
*method:* magnetic scattering from  $\chi(q, E)$

*method:* magnetic scattering from Heisenberg model

*method:* simulation of intermediate results - user selection of  $S(Q)$ ,  $S(\vec{Q})$ ,  $S(E)$ ,  $S(Q, E)$ ,  $S(\vec{Q}, E)$ , with input on  $\vec{Q}$

*method TBA:* Molecular Dynamics (may be too demanding on computing resources. prototype needed before design.)

#### 10.4 *Ab Initio Lattice Dynamics*

*purpose:* Ab Initio Lattice Dynamics will generate the phonon properties of crystal structures of moderate complexity from first-principles electronic structure codes such as VASP. The quantities generated will include the phonon density of states (DOS)  $g(E)$ , dispersions  $E(\vec{q})$  in selected directions, and the scattering functions  $S(Q, E)$  and  $S(\vec{Q}, E)$ . The scattering function  $S(\vec{Q}, E)$  can then be used in a scattering kernel to generate the sample scattering with a Monte Carlo simulation.

*input:* Crystal structure

*input:* Simulation parameters for *ab initio* electronic structure code

*input:* User selection of quantity to calculate

*output:*  $g(E)$ , with or without neutron-weighting in case non-monatomic crystals

*output:*  $E(\vec{q})$  in selected directions

*output:*  $S(Q, E)$ ,  $S(\vec{Q}, E)$

*method:* VASP to calculate the atomic forces corresponding to atomic displacements from first principles

*method:* Symmetry analysis to automatically generate complete set of atomic displacements

*method:* Fitting of forces to generate interatomic force-constant tensor

*method:* Linear response method for dispersions in selected crystallographic directions

*method:* BvK lattice dynamics to generate  $g(E)$ ,  $E(\vec{q})$ ,  $S(Q, E)$ , and  $S(\vec{Q}, E)$

*method:* neutron weighting of  $g(E)$  or  $S(Q, E)$  or  $S(\vec{Q}, E)$

#### 10.5 *Science Package – Contingency Item*

*purpose:* Science Package will be one example of a data analysis network built from modules developed in the DANSE project. This application and its documentation will provide a template for how scientists could use the components of the DANSE system to quickly build novel applications for data analysis. The functionality of this package will be decided after the DANSE beta release. The example here is representative – take a set of data files acquired at different temperature, and construct an anharmonic thermodynamic partition function.

*input:* NeXus files with parametric variation in  $T$

*input:* weights of different input files

*output:* partition function  $Z(T)$

*method:* script to run multiple cases of reduction and model to get  $g(E)$

*method:* calculate  $Z(T)$  from  $g(E)$