# SINE2020 WP10 Workshop in Grenoble (24-25 April)

Feedback from the meeting

1. GUI
    a. Overall postive feedback
    b. Context menu for ploting - request for Silx feature
    c. Integrated jupyter notebook
2. CLI and Sasmodels
    a. numba http://numba.pydata.org/ on-the-fly compilation (llvm), cython
    b. we were advised to keep python models
    c. smarter fitting strategy; start fitting with least complicated models but also looked into models if they can be simplified
    d. For the core-shell ellipsoid, it would be great to have 3 different axis (instead of 2).
3. Tools for structural biology
    a. PDB-SANS: Read a pdb file and calculate its SANS curve in a given buffer (% D2O) and at a given concentration (mg/mL), and with a given H-D exchange rate (in % of exchangeable H) and a given deuteration level of the protein (in % of the non-exchangeable H), with the possibility to add a hydration shell and a smearing. Input: pdb file; concentration (mg/mL); deuterate the molecule (y/n, if yes, which % of non-exchangeable H); % D2O of buffer; % exchange; hydration layer (y/n); Instrumental smearing (y/n, if yes, 2 columns text-file: Q (A-1)/smearing; can be taken from measured data or from the resolution estimator); Output: 4-column text file: Q/I/errI=0/smearing. That would be extremely useful to prepare experiments. That would let enough flexibility in writing your own pdb, with fancy deuteration if you like.
    b. PDB-SANS fit: Compare (Fit) the curve calculated from a PDB file (as described above) to experimental curves. Input: theoretical curve (4-column text file); experimental curve (4-column text file); Output: Chi square. That would be extremely useful to analyse experiments.
    c. Multi-PDB-SANS fit: Fit experimental curve with curves calculated from several pdb. Output: proportion of each structure, the best mixture curve and a chi square of the total fit. That would be extremely useful in the case of several structures being represented in the sample (partial oligomerization; conformational equilibrium).
    d. MD-SANS: That is probably much more complicated... From a pdb file and all the inputs of the 1st point, plus a file specifying the flexible regions (in a text file, as in MultiFOXS), create a large number of conformations (inspired from rrt-sampler), fit them to experimental curve and choose the optimal combination (of a minimum number of structures) to define the sample. I imagine the amount of work that this represents, but if we can get Sali's lab interested, they can reuse their algorithm in SASview maybe..
    e. I tried the "generic scattering calculator" with a pdb file. It is not as well adapted to biologists needs as what I proposed above: the polarization stuff is confusing, the deuteration must be entered manually in the pdb before, it is not written in the help if the "scale" is the concentration or something else... And also, I do not know why, but it is much faster than the estimated computing time...
    f. freesas https://github.com/kif/freesas for CorMap and dummy models generation in future
4. 2D data treatment
    a. need to define typical workflow
    b. potentailly different figure of merit
5. Licensing
    a. License GUI part as GPL so as to not need a commercial license for PyQT. In theory we will want to release sasgui, sascalc and sasmodels all as separate packages on PyPi and so separate licensing should be clean.
6. Documentation and tutorial
    a. Jupyter notebook for writting tutorial as new functionality is developed. This notebook can be saved as rst and then used to build documentation
    b. Video tutorials is a good idea. Keep them modularized

Next workshop in spring 2018. Ready for user testing.