McStas: Liquids and Powders Dynamics

E. Farhi, *ILL*

Isotropic density samples in McStas

Outline

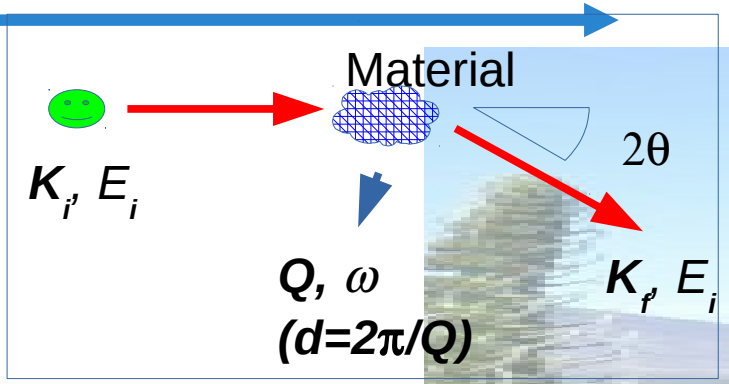
- Scattering law, a bit of theory
- The *Isotropic_Sqw* component and input data files
- How to get S(q,w) data sets:
  - From MD
  - From lattice dynamics
  - From experiments
- Exercise: a “liquid” TOF spectrometer
- Exercise: separate contributions

Disclaimer: in case of errors and uncertainties, please correct me...

Neutron-matter interaction

Selections rules $K_f = K_i + Q$

$E_f = E_i + \omega$



Bragg's law (diffraction on structure – atoms separated by distance *d*)

$$n\lambda = 2\pi / K_i = 2d \sin\theta$$

Scattering law (intensity per solid angle and energy, dynamics)

Holy Book (Squires)

$$\frac{d^2\sigma}{d\Omega\, dE_f} = \frac{K_f}{K_i}\left[\frac{\sigma}{4\pi} S(Q,\omega)\right]$$

Dynamical structure factor *S(Q,ω)* is characteristic of each material

Reflects ordering of matter (atom/molecule positions – movements - domains)

Scattering law

The double differential scattering cross section gives the probability for a neutron to scatter for a given solid angle and energy

The total intensity is the full integral over all scattering possibilities.

Effective total scattering cross section

$$\hat{\sigma} = \iint \frac{d^2\sigma}{d\Omega dE_f} d\Omega dE_f$$

V.F. Sears. Adv. Phys., 24, 1, 1975.

Computing the total scattering probability

The total scattering cross section is given in (Ω,E_f) space, but *S* is given in (q,ω)
A variable change must be done for the integration (Jacobian).

We like to play games in (q,ω) space

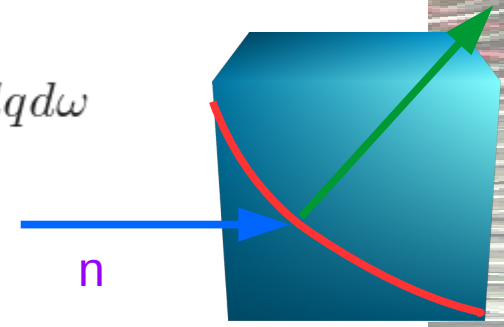
$$\frac{d\Omega}{d\theta} = -2\pi sin\theta$$
$$\frac{dq}{d\theta} = -\frac{k_i k_f sin\theta}{q}$$

Effective cross section in (q,ω) space

$$\hat{\sigma} = \sigma \iint \frac{S(q,\omega)q}{2k_i^2} dq d\omega$$

Probability to transmit

$$p = e^{-\rho \hat{\sigma} x}$$

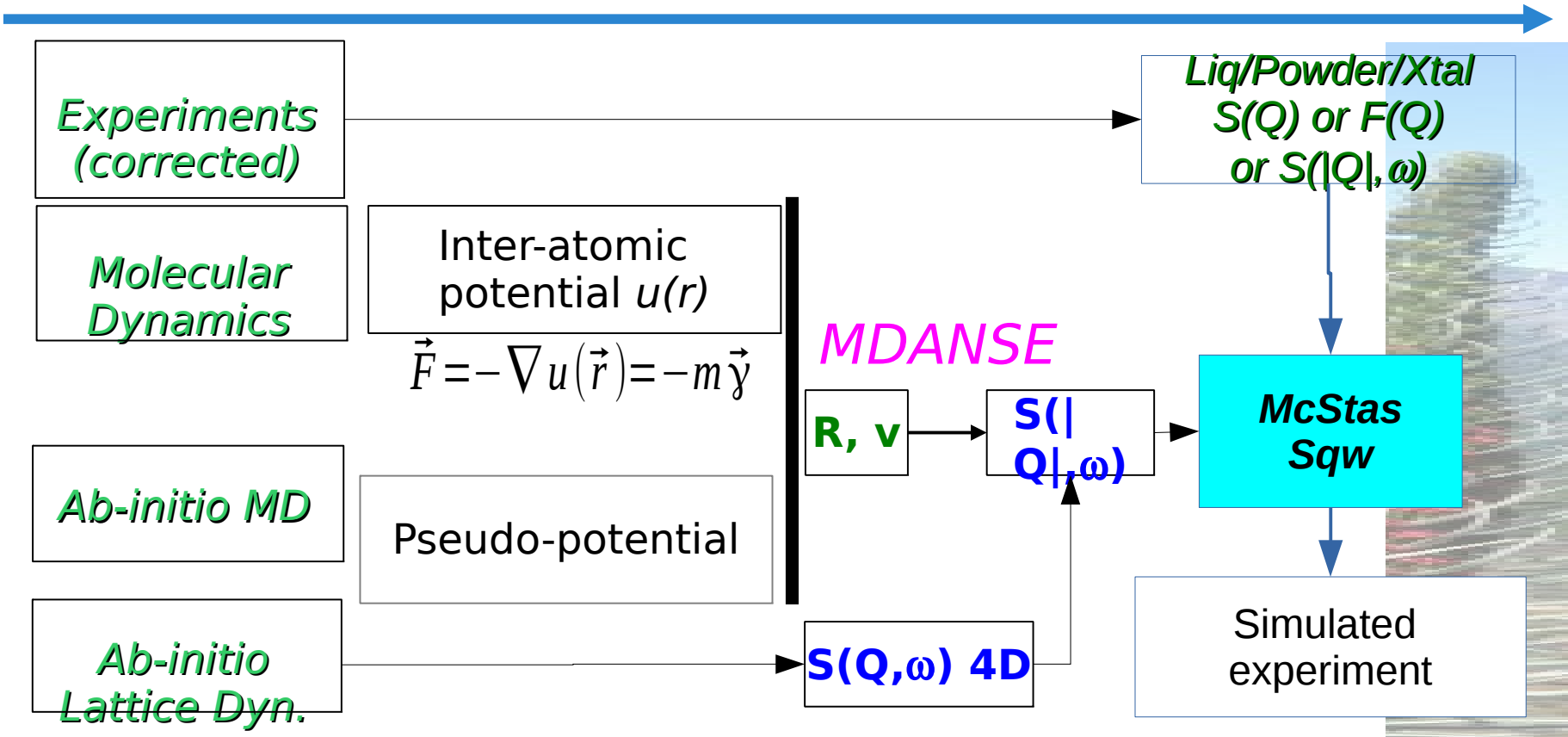


Scattering distribution

S(q,ω)

with importance sampling to scatter preferably where *S* is large

How to get S(q,w) data sets



MD step is done prior to the virtual experiment (NAMD, VASP, GROMACS, ...). Computationally intensive (e.g. use clusters). Then use FFT(r,t)

McStas provides a few sample S(Q,w): Rb, Ge, H2O, D2O, D2, ...

Isotropic_Sqw: Handles **elastic** and **inelastic** for both **coherent** and **incoherent** channels

Isotropic_Sqw syntax

- Isotropic_Sqw(

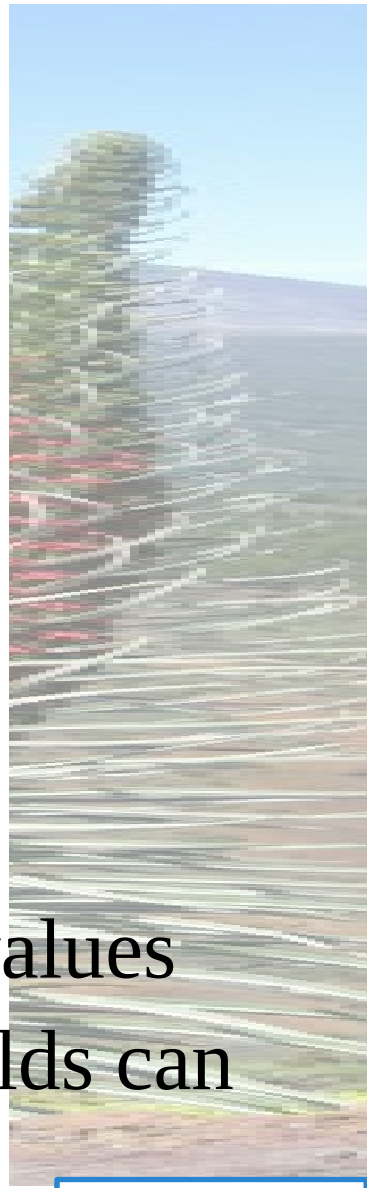
  Sqw_coh=FILE_COH,

  Sqw_inc=FILE_INC,

  radius=R,

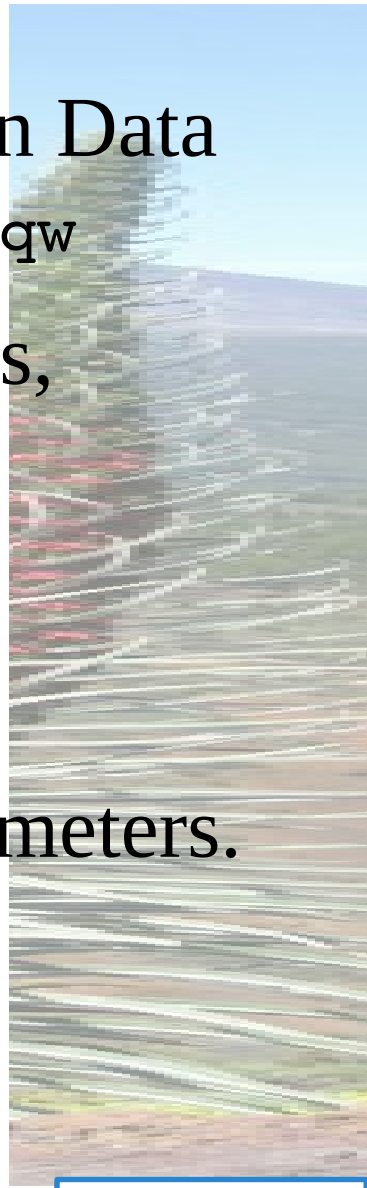
  height=H)

- More component parameters can specify geometry, physical properties, ...
- The data files specify the $S(Q,\omega)$ or $S(Q)$ values as a matrix with Q,ω extent. Additional fields can be included as meta data (# lines).



Isotropic_Sqw data format

- Open McStas, select Help / Comp Ref, then Data files. Select a .sqw file, e.g. `He4_liq_coh.sqw`
- Look at format: Header and meta data, axes, matrix
- Used # fields:
  - density, weight, sigma, Temperature
- Meta data can be given as component parameters.



Isotropic_Sqw data format

```
# Sqw data file for Isotropic_Sqw
# liquid He4: coherent part, no incoherent, atomic number 2
# Elementary Excitation Data by R.J. Donnelly et al., J. Low Temp. Phys., 44 (1981) 471
# WARNING: line width is constant, intensity is not right
#
# Physical parameters:
# V_rho        0.072    atom density per Angs^3
# weight       4.002    in [g/mol]
# density      0.4784   in [g/cm^3]
# sigma_abs    0.00747 absorption scattering cross section in [barn]
# sigma_coh    1.34     coherent scattering cross section in [barn]
# sigma_inc    0        incoherent scattering cross section in [barn]
# Temperature  2        in [K]
# classical    0        experimental, contains Bose factor
#
# q axis values
# vector of m values in Angstroem-1
0.001000 0.011000 0.02 …
# w axis values
# vector of n values in meV
0.001391 0.011391 0.021391 0.0313 …
# sqw values (one line per q axis value)
# matrix of S(q,w) values (m rows x n values), one line per q value
9.721422 10.599145 11.344954  …
```

He4_liq_coh.sqw

Such files can be written from an iFit iData_Sqw2D object:

- sqw2d = iData_Sqw2D(...);
- saveas(sqw2d, 'filename.sqw','mcstas')

Sqw file from experiments

- Reduce the experimental data, correct it for e.g. empty cell, parallax, detector efficiency, …
- Correct for absorption, incoherent scattering, multiple-scattering, ...
- Integrate over |Q| as *Isotropic_Sqw* is isotropic.
- Write a text file with [Q,ω] and S(Q,ω) vectors/matrices. Add meta-data.

Sqw file from MD

- Set the system box, (pseudo) potential/FF
- Equilibrate, and couple to thermostat (NVT)
- Run calculator in MD mode (NVE)
- Convert trajectory (**R**,t) into S(|Q|,ω) [MDANSE]
  - Analysis/Scattering/DCSF and DISF → .nc file
- Export into .sqw text files.
  - sqw2d = iData_Sqw2D('DCSF.nc');
  - saveas(sqw2d, 'DCSF.sqw','mcstas');

Sqw file from Lattice Dynamics

- Create a crystal, calculate displacements or so.
- Assemble the ‘phonon’ representation
- Evaluate S(Q,w) on powder average
- Export to McStas Sqw

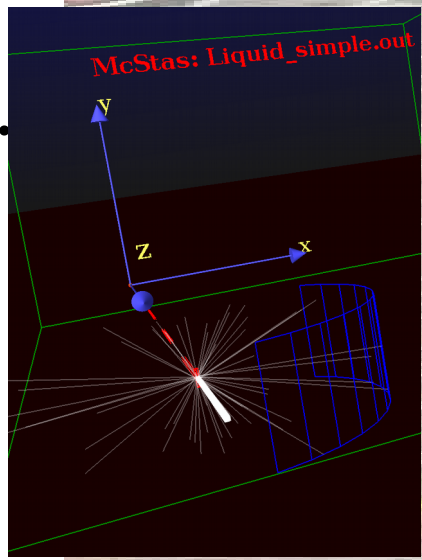
Sqw: Exercise 1: a liquid TOF

Aim: A simple spectrometer (and diffractometer)

- Create a new instrument from 'template (test)'.
- Call it Liquid_simple and define input parameters
```
(lambda=2.36, string coh="Rb_liq_coh.sqw", string
inc="Rb_liq_inc.sqw")
```
- Insert a Source_simple ϕ1cm sending λ=lambda with $d\lambda/\lambda$=1%. Focus onto a 1x1cm^2 area.
- Insert an Isotropic_Sqw 3m away, using σ_{coh}=coh, σ_{inc}=inc with ϕ1cm x 5cm.

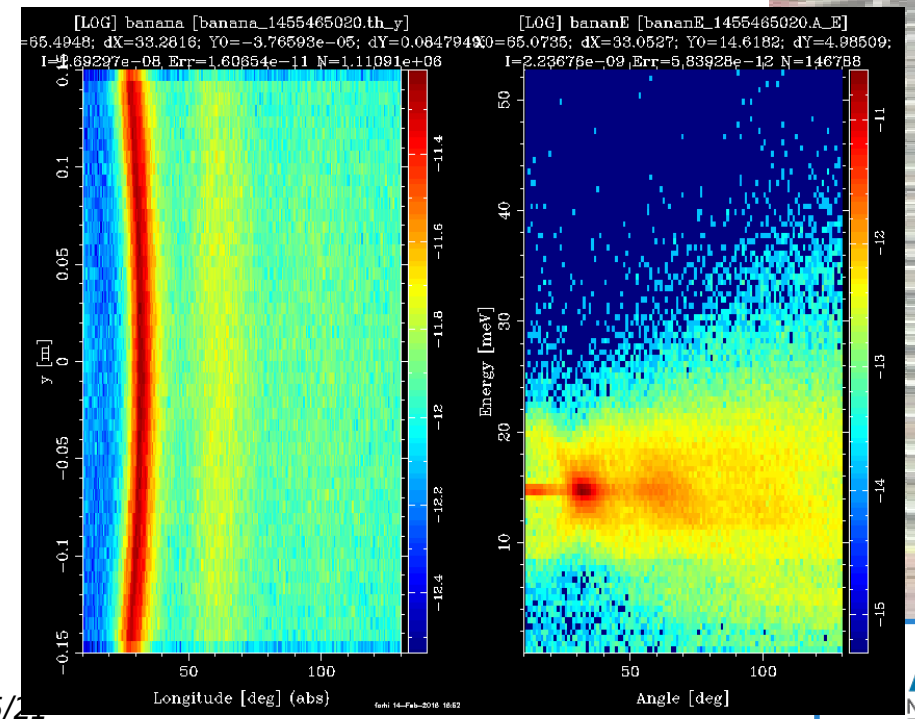
Sqw: Exercise 1: a liquid TOF

- Add a Monitor_nD cylindrical detector ϕ1m x 30cm, sensitive to ***(θ,y)*** for diffraction, centred on the sample, with 100 bins.
- Add the same, but sensitive to **(angle,energy)** with automatic energy limits.
- Save, run in *Trace 3D* to check geometry.



Sqw: Exercise 1: a liquid TOF

- Run in Simulation/PGPLOT mode with 1e8 neutron events.
- Plot results !
- Comment on the diffraction pattern and the inelastic one.



Sqw: Exercise 2: contributions

- Insert an instrument variable in the DECLARE block, as 'flag_scat'.

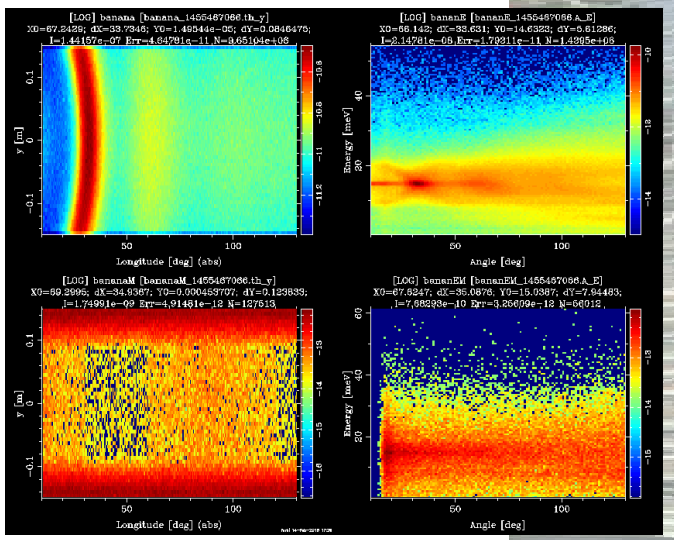
```
DECLARE %{
int flag_scat=0;
%}
```

- After the AT token of the 'sample', insert an EXTEND block that sets flag_scat to the number of SCATTERED events.

```
EXTEND %{
flag_scat=SCATTERED; // nb of scattered events
%}
```

Sqw: Exercise 2: contributions

- Duplicate the 2 monitors, and make them sensitive to **multiple-scattering** only. Add a WHEN(flag_scat) between the COMPONENT and the AT keywords:




```
COMPONENT …

WHEN(flag_scat > 1)

AT ...
```

- Save and Run !
- How much multiple scattering ratio at λ=2.36 Å
- Make the sample ϕ5cm and repeat.

Sqw: Exercise 3: from iFit: build

- McStas can be controlled from within iFit.
- Open Matlab/iFit
- Create the Liquid_simple model with:
  - model = mccode('Liquid_simple')
- Plot the geometry with:
  - plot(model) % has contextual menus
- Then run it with (default 1e6 event and pars)
  - data = iData(model, [], nan);
  - subplot(model)

Sqw: Exercise 3: from iFit: eval

- Specify parameters
  - data = iData(model, ‘lambda=2.36; coh=Cu.laz’)
- Do a scan:
  - data = iData(model, ‘lambda=[1.2 2.4 3.6]’)
- Change neutron events #
  - model.UserData.options.ncount = 1e7;

Sqw: Exercise 3: from iFit: optim

- Fix all parameters but lambda, Maximize model value:
  - mlock(model, ‘all’); munlock(model, ‘lambda’)
  - xlim(model, ‘lambda’, [1 3]); % bounds
  - fmax(model, ‘lambda=2.36’, ‘’, nan)
  - fmax(model, ‘lambda=2.36’, ‘OutputFcn=fminplot’, nan)

Sqw: Exercise 3: from iFit: crazy

- You can add two McStas models:
  - model=mccode('instr1')+mccode('instr2')
- Edit the instrument and re-compile
  - edit(model)
- ...