

# Sasview and SAS Data Analysis

Dr Stephen KING

TNT 2024, San Giovanni, Italy

17 Jun 2024





SasView is an international, **open-source, collaborative, community-driven** SAS data analysis development project

- 2006; originates in the NSF DANSE project
- 2013; transitions into a community project
- 2020; some EU Horizon programme funding
- over 50 contributors to date
- ~15 active at any one time
- twice monthly Zoom calls
- regular 'contributor camps' & 'hackathons'
- small leadership team



SasView Contributor Camp, Delaware, USA, Jan 2024

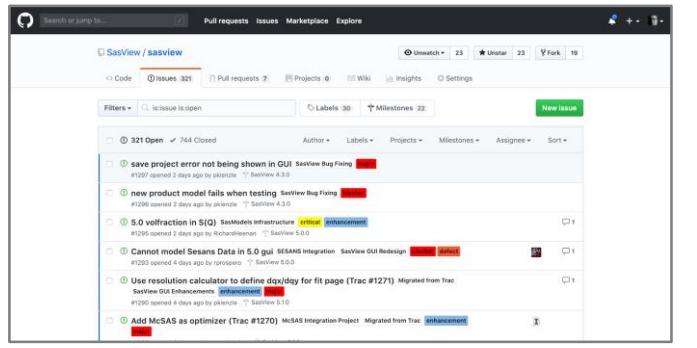
- 15 different facilities or universities represented
  - 1/3 participants from outside the USA
    - 2/3 attending for the first time



# Open, Collaborative, Community Development

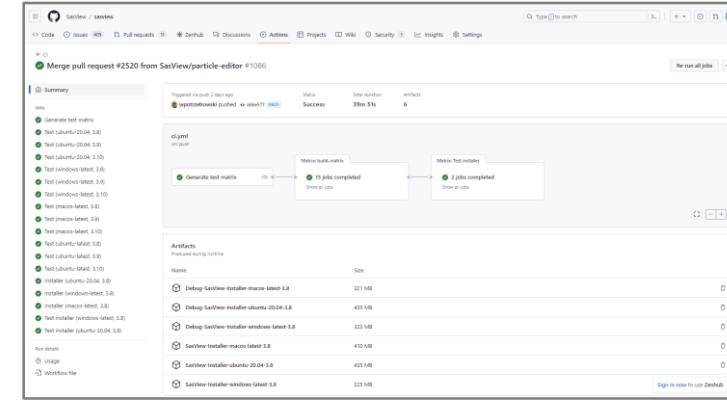
Code is open source and publicly hosted at Github  
Released under BSD 3-clause license

Code Hosting, Issue Tracking & Developer Wiki on Github



<https://github.com/SasView>

Automated Builds with Github Actions



<https://www.sasview.org>

Model Marketplace for Users to share their fitting models

Name	Description	Category	Upload Date	Author	Score	Verified Categories:
correlated_spheres	The 1D scattering length of two correlated spheres can be written as $g(r) = F_1 r^2 + F_2 r^2 + 2F_1 r^2 \cdot 1/r^2 \sin(\theta)/\theta D_S$ , where $S_F$ , $F_1$ and $F_2$ are the parameters of the model.	Sphere	30 Mar 2019	Tianfu	0	Cylinder Ellipsoid
WoodSAS	This model is tailored for fitting the equatorial intensity profile from wood samples (Penttila et al. 2018). The model consists of three independent contributions: 1) Scattering in the plane per...	Cylinder	15 Mar 2019	penttila	0	Lamellae Other
Nanodisc	This is a simple re-parameterization of the core-shell bicelle model such that it can be more easily applied to the fitting of a phospholipid nanodisc.	Cylinder	02 Dec 2018	arm61	0	Paracrystal Parallelepiped
TestModel	Something	Other	12 Oct 2018	tsm.snow	0	ShapeIndependent
Core Shell Bicelle	Definition This model provides the form factor for an elliptical cylinder with a core-shell scattering length	Cylinder	08 Oct 2018	penttila	0	Sphere

<http://marketplace.sasview.org>

# Open, Collaborative, Community Development

Traceability with a DOI for each release

The screenshot shows the Zenodo software page for SasView version 5.0.6. At the top, there's a navigation bar with the Zenodo logo, a search bar, and links for 'Communities' and 'My dashboard'. On the right are 'Log in' and 'Sign up' buttons. Below the header, it says 'Published June 6, 2023 | Version v1'. The main content area has a title 'SasView version 5.0.6' and a list of authors. It includes sections for 'Data managers', 'Others', 'Researchers', 'Supervisors', and 'Work package leaders'. A 'Show affiliations' button is located near the bottom of the author list. To the right, there are stats: 97 views and 41 downloads. Below that is a 'Versions' section showing 'Version v1' (Jun 6, 2023) with the DOI 10.5281/zenodo.7581379. A note explains that all versions can be cited using this DOI. Further down are sections for 'External resources' (Indexed in OpenAIRE) and 'Details' (DOI 10.5281/zenodo.7581379, Resource type: Software, Publisher: Zenodo). At the bottom left is a '5.0.6 release' link, and at the bottom right is a 'Files' section listing two files: 'SasView-5.0.6-MacOSX.dmg' (283.9 MB) and 'setupSasView-5.0.6-Win64.exe' (181.2 MB), each with a 'Download' button.

<https://zenodo.org/communities/sasview-analysis>

# Open, Collaborative, Community Learning

- website
- documentation
  - in-program & online
- written tutorials
- video tutorials (YouTube)
- taught courses
  - scattering schools/workshops
  - university courses
- bootcamps & regional workshops
- e-learning
- Slack
- X (Twitter)

- [help@sasview.org](mailto:help@sasview.org)
- [users@sasview.org](mailto:users@sasview.org)
- <http://lists.sasview.org/cgi-bin/mailman/listinfo/users>

The collage consists of four screenshots:

- Screenshot 1:** A screenshot of the SasView Slack interface. It shows the Slack desktop app with the SasView Slack workspace selected. The sidebar lists channels like #general, #code\_camp, #manuscript\_chat, and #random. The main pane shows messages from users Michael Zhang and wpotrzebowksi in the #general channel.
- Screenshot 2:** A screenshot of the SasView website's "Working with SasView" page. It features a "Table of Contents" sidebar with sections like "Loading Data", "The data explorer", and "Loading data". The main content area provides detailed instructions on how to load data into SasView, mentioning the Data Explorer panel and context menu options.
- Screenshot 3:** A screenshot of the SasView website's "FAQ" page. It lists several frequently asked questions about SasView, such as "What is SasView?", "What platforms does SasView run on?", and "Is there a SasView Manual?". Each question has a blue link to its answer.
- Screenshot 4:** A screenshot of the SasView website's homepage. It features a large circular image of a scattering experiment, the text "Software for analysis of small angle scattering modelling #SANS #SAXS.", and social media links for GitHub, LinkedIn, and X (Twitter). At the bottom, it shows the site's footer with links to sasview.org and the date "Joined February 2015".

# Open, Collaborative, Community Learning

- website
- documentation
  - in-program & online
- **written tutorials**
- **video tutorials (YouTube)**
- taught courses
  - scattering schools/workshops
  - university courses
- bootcamps & regional workshops
- e-learning
- Slack
- X (Twitter)
- [help@sasview.org](mailto:help@sasview.org)
- [users@sasview.org](mailto:users@sasview.org)
- <http://lists.sasview.org/cgi-bin/mailman/listinfo/users>

The image shows two screenshots of the SasView website. The top screenshot is the homepage, featuring the SasView logo, navigation links (ABOUT, LINKS & DOWNLOADS, DOCUMENTATION, CONTENT, HELP), and a search bar. Below the navigation is a section titled "Written Tutorials:" with a list of PDFs: "Getting started with SasView", "Basic 1D Fitting in SasView", "Simultaneous 1D Fitting in SasView", "Creating Custom Fitting Models in SasView", "P(r) Inversion Analysis in SasView", "Correlation Function Analysis in SasView", "Subtracting a Model Calculation from Data in SasView", and "for earlier versions of SasView". The bottom screenshot shows a YouTube channel page for "sasview" with 5 subscribers. It displays four video thumbnails: "Using the P(r) calculator in SasView" (65 views, 5 months ago), "Scattering Length Density Calculator in SasView" (43 views, 5 months ago), "Introduction to applying the beta approximation in..." (23 views, 5 months ago), and "Calculating the Scattering Invariant in SasView" (29 views, 5 months ago). To the right of the channel page are two screenshots of the SasView software interface. The left screenshot shows a graph with multiple curves and a constraint setup window. The right screenshot shows a "Fit Constraints" dialog box with several parameters listed. A tip at the bottom right of the software interface says: "Tip: if you need to scale parameter values between FitPage's then use the free-form constraint box below Easy Setup. The right-hand side of the equality can be of the form: scalar \* [M]parameter\_name". Below the software screenshots, there is explanatory text: "As we are assuming the volume fraction of droplets in each sample was different, remove the two scale constraints. And because the different datasets represent samples containing different amounts of H & D (and therefore have different incoherent scattering contributions), also remove the two background constraints. But as it is the 'shell' contrast (M1) which should provide the most sensitivity to the thickness parameter, we need to change the constraint equalities dealing with thickness to read: M2 thickness = M1.thickness M3 thickness = M1.thickness So we now have constraints that look like this:

# Open, Collaborative, Community Learning

- website
- documentation
  - in-program & online
- written tutorials
- video tutorials (YouTube)
- taught courses
  - **scattering schools/workshops**
  - university courses
- bootcamps & regional workshops
- **e-learning**
- Slack
- X (Twitter)

- [help@sasview.org](mailto:help@sasview.org)
- [users@sasview.org](mailto:users@sasview.org)
- <http://lists.sasview.org/cgi-bin/mailman/listinfo/users>



The work of ISIS Sandwich Student  
Michael Oakley

The screenshot shows the homepage of the PaN-learning platform. At the top, there is a navigation bar with the PaN-learning logo and a search bar containing the URL <https://pan-learning.org/>. Below the header, a welcome message and course descriptions are displayed. A sidebar on the right lists various available courses. Two specific courses are highlighted with orange borders: "Introduction to muon spin spectroscopy" and "SasView: Analysis of SAS Data (Swedness)".

Welcome to PaN-learning

PaN-learning is a Photon and Neutron e-learning platform, which aims to bring free education and training to scientists and students.

Below you will find courses both on the theory of photon and neutron scattering along with how to use python code or software for data reduction and modelling.

If there is a course you are interested in but self enrolment is not possible or if you wish to get in contact with the teacher, please email [stella.dambrumenil@ess.eu](mailto:stella.dambrumenil@ess.eu).

Find out more about our partners, **PaNOSC** and **ExPANDS**, on their websites.

If you are interested in contributing to PaN-learning, please send an email to [stella.dambrumenil@ess.eu](mailto:stella.dambrumenil@ess.eu).

Search courses

Available courses

- Neutron Scattering Library
- Introduction to neutron scattering
- Advanced topics in neutron scattering
- Quasielastic Neutron Scattering Course
- Introduction to muon spin spectroscopy**
- Muons in Semiconductors
- Muons in Magnetism
- Muons in Superconductivity
- IKON Python Workshop
- Illumidesk: Python Course
- SasView: Analysis of SAS Data (Swedness)**

# Outline

## 1 Data

What do we mean by data?

## 2 Data Formats

A possibly dull, but important, consideration!

## 3 Options for Data Analysis

What SasView can/cannot do for you!

## 4 Model-Fitting

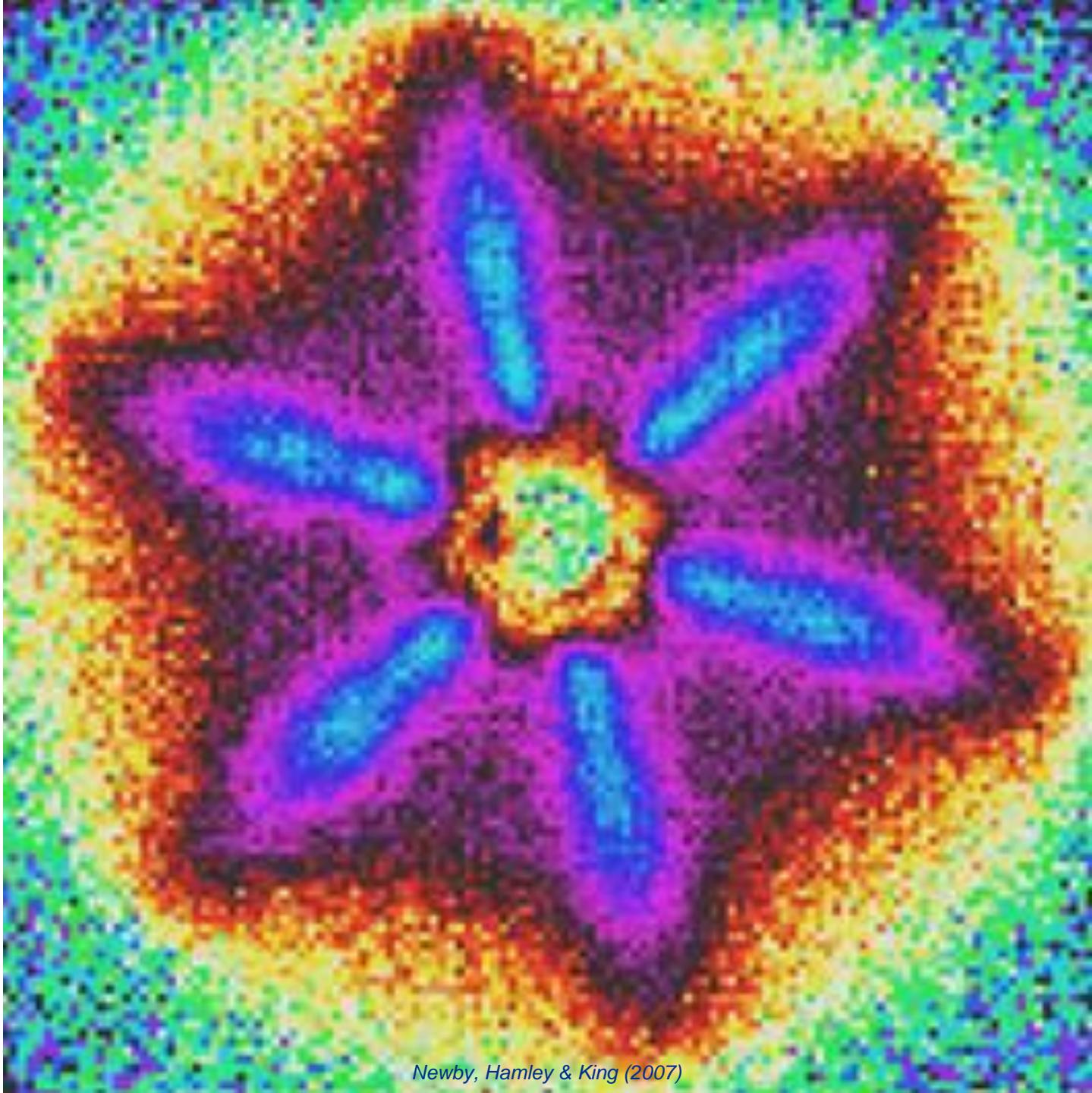
Objectives, pitfalls, limiting behavior...

## 5 Real-Space Methods

Performing ‘magic’!

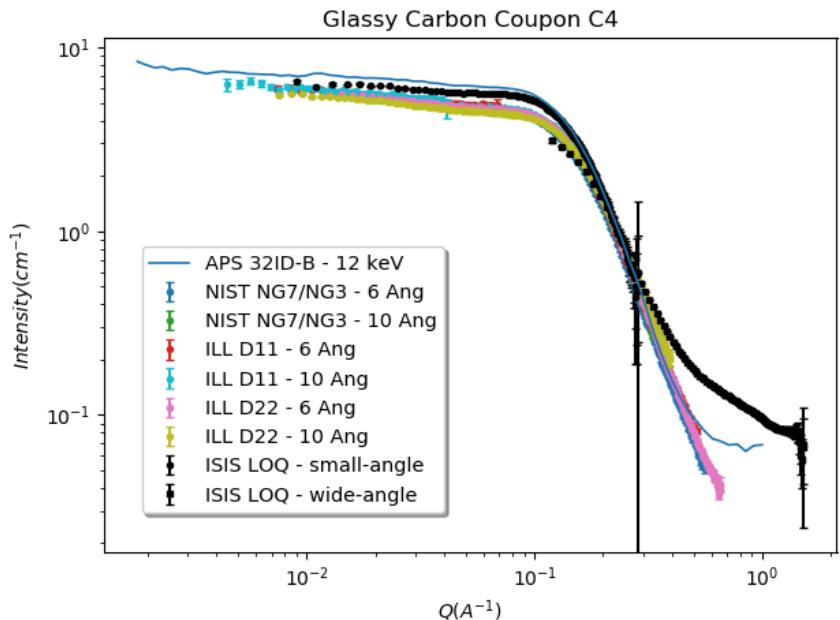
## 6 Other Methods

Invariant, Kratky plot, ...



# Measurement Data Flow

Reduced data from the same sample should be the same no matter where it is measured!

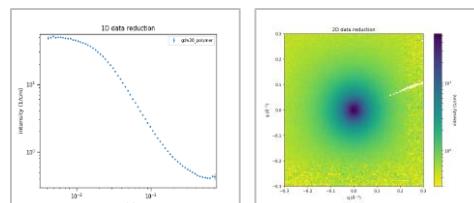
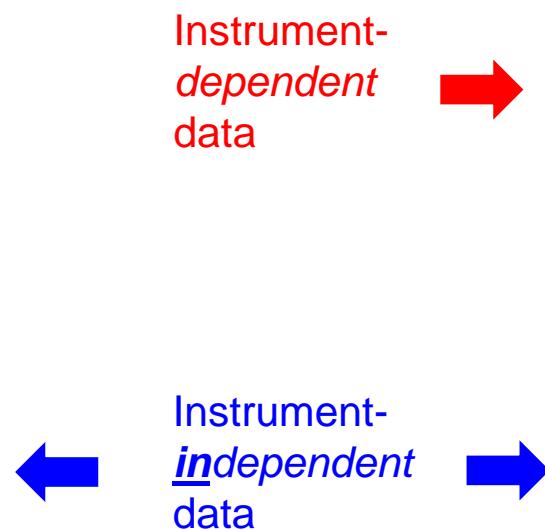


<http://www.cansas.org>



Science and  
Technology  
Facilities Council

ISIS Neutron and  
Muon Source



Data Acquisition System

'raw' data file  
(eg, NeXuS)

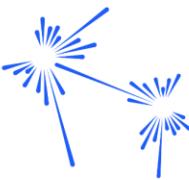
Data Reduction Programs  
(eg, *Mantid*, *Grasp*, *Dawn*,...)

reduced data file  
(1D: HDF / XML / TXT; 2D: HDF / TXT)

Data Analysis Programs

Paper / Thesis

# Reduced Data



## Real data:

- the scattering data; 1D (*intensity vs  $q$* ) or 2D (*intensity vs  $q_{parallel,x}$  vs  $q_{perpendicular,y}$* )
- the associated uncertainties
- maybe *transmission vs  $\lambda$*  from ToF-SANS

## Meta data:

- identifying date, time, user, instrument, sample, reduction software & version, ...

## Non-neutron data:

- some instrument settings (but for informational use only)
- parameters ‘scanned’ during the course of a measurement (eg, T, P,  $\dot{\gamma}$ ,  $\theta$ , etc)
- simultaneous ancillary measurements (eg, DLS correlation functions)
- simulation data

# Reduced Data Formats

## 'Minimalist' (e.g. 'plain text', TXT, DAT, CSV\*):

- typically just contains some numbers & a header (if you are lucky)
- but need to be psychic to know what the numbers are!

SANS2D Fri 26-APR-2024 18:21 Workspace: EVA_145_15 at -30.0C Step 0_SANS_SANS						
100	0	0	0	1	100	0
0	0	0	0	0	0	0
3 (F12.5,2E16.6)						
0.00154	2.279545e+03	4.132414e+02	2.892517e-04			
0.00164	1.964024e+03	1.973429e+02	2.952573e-04			
0.00174	1.749913e+03	1.418314e+02	2.996766e-04			
0.00184	1.547591e+03	9.235745e+01	3.059102e-04			

## Self-describing (e.g. XML, CanSAS1D):

- uses 'tags' to order and describe the data/metadata to an agreed structure
- but relies on an external 'schema' file

The screenshot shows an XML editor interface with an open schema file named 'cansasid.xsd'. The code defines a schema for a 'SASroot' element with version 1.1, using the namespace 'urn:cansasid:1.1'. It includes declarations for 'xs:xs' and 'xs: schemaLocation'. The schema defines a 'SAStitle' element under 'SASEntry' and a 'SASdata' element under 'SASdata'. The 'SASdata' element contains several 'I' and 'Q' elements with specific unit definitions.

## Hierarchical (e.g. HDF5, NXcanSAS):

- also self-describing, but fully portable (no outside information needed)
- can contain a mix of data types ('a file-system-in-a-file')
- but not human-readable



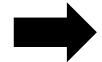
SasView inputs all these!

\* NB: But beware commas (,) for thousands separators and semi-colons (;) !

# Options for Data Analysis

Are many & varied...

<http://smallangle.org/content/Software>



- not an exhaustive list
- >70 software packages!
- 11 model-fitting packages!  
*GENFIT, pySAXS, SASfit, SasView, scÅtter, WillItFit, X+, ...*

But:

- some packages are ‘ancient’!
- many of variable quality
- many with little/no user support
- many are closed-source
- some are very niche!

...Choose wisely!

The screenshot shows the 'Software' page of the SAS Portal. The page header includes the SAS logo and the text 'THE home for Small Angle Scattering'. A 'Select Language' dropdown is located in the top right corner. The main content area has a heading 'Software' with a sub-section 'SOFTWARE'. Below this, a paragraph explains that the list is not comprehensive but can help guide users. It encourages suggestions to [info@smallangle.org](mailto:info@smallangle.org). The 'CONTENTS' section lists various software categories with their respective links.

- [Software for Data Reduction and Visualisation](#)
- [Software for Model-Fitting](#)
- [Software for the Analysis of Biomolecular & Fibre Systems](#)
- [Software for Peak-Fitting and Correlation Function Analysis](#)
- [Software for Parameter Distribution Analysis](#)
- [Software Utilities](#)
- [Commercial SAS Software](#)
- [Software for the Analysis of Grazing Incidence SAS](#)
- [Other SAS Software](#)
- [Source Code Repositories](#)
- [Software for the Analysis of Reflectometry Data](#)



Science and  
Technology  
Facilities Council

ISIS Neutron and  
Muon Source

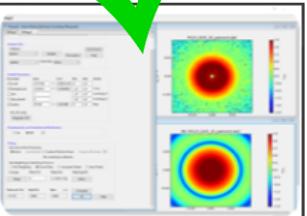
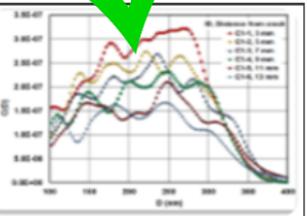
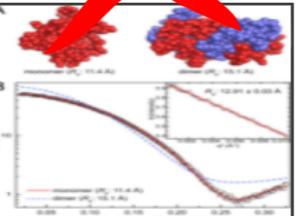
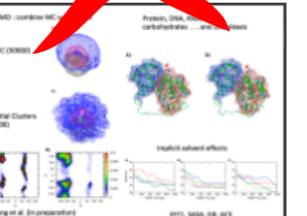
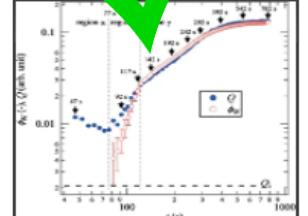


### Note!

- not every option is needed every time!
- often one option will suffice

### How to choose?

- your field/speciality
- what do you want to know?
- the data quality
- etc...

Model-Fitting Methods	Real-Space Methods	Ab-Initio Methods	MC/MD Methods	Other Methods
 <p>Example of 2D model-fitting using the SasView application 10.1179/1743284714Y.0000000577</p>	 <p>Cavity size distributions in a steel weldment as derived from SANS 10.1179/1743284714Y.0000000577</p>	 <p>Ab-initio modelling of polcalcin constrained by SAXS 10.1002/pro.3376</p>	 <p>MC &amp; TAMD modelling of proteins constrained by SANS 10.1016/j.jmgm.2017.02.010</p>	 <p>Time evolution of the invariant during crystallisation of P4MP1 10.1038/pj.2012.204</p>
<p>This approach uses iterative optimisation to match the calculated scattering from a model function describing the scattering objects to the measured scattering data. Each iteration one or more physical parameters describing the model (e.g. concentration, size, scattering length density) are adjusted.</p>	<p>This approach uses mathematical transformations (e.g. Fourier Transforms) to convert the measured scattering data in reciprocal-space (i.e. in Q-space) into a function in real-space. Typical outputs are density correlation functions, volume fraction distributions, and size distributions.</p>	<p>This approach uses iterative optimisation to match the calculated scattering from assemblies of spheres or from a 3D 'shape envelope function' to the measured scattering data. Each iteration the number and/or position of the spheres, or the curvature of the envelope function, is adjusted.</p>	<p>This approach uses iterative optimisation in combination with Monte Carlo (MC) and/or Molecular Dynamics (MD) techniques or RRT searches to match a calculated 'atomistic level' structure for the scattering objects to the measured scattering data.</p>	<p>Other approaches to data analysis may involve identifying, for example: any Q-dependencies in the measured data, particular patterns in the Q-values of any peaks present, asymptotic extrapolations, calculation of the integral under the measured data (the 'invariant'), or the intensity at Q=0.</p>

<https://www.isis.stfc.ac.uk/Pages/SANSdataanalysisOverview.aspx>

# (Classic) Model-Fitting Methods

**Recap:** ' $I(q)$ ' =  $(\partial\Sigma/\partial\Omega)(q) = I_{q=0} P(q) S(q) + B(q)$

$$= \Lambda N V^2 (\rho - \rho_{matrix})^2 P(q) S(q) + B(q)$$
$$= \Lambda \phi V (\rho - \rho_{matrix})^2 P(q) S(q) + B(q)$$

Form Factor

$$P(q, R)_{sphere} = 9 \left[ \frac{\sin(qR) - (qR)\cos(qR)}{(qR)^3} \right]^2, \quad P(q, R_g)_{debyecoil} = \frac{2 \left[ \exp(-(qR_g)^2) + (qR_g)^2 - 1 \right]}{\left[ (qR_g)^2 \right]^2}, \text{ etc...}$$

SasView offers:

~90  $P(q)$

Structure Factor

$$S(q) = 1 + 4\pi N \int_0^\infty \left[ (g(r) - 1)r^2 \frac{\sin(qr)}{(qr)} \right] dr; \quad \text{where } g(r) \propto U(r)$$

4  $S(q)$

But you can add more!

## Method:

- choose a  $P(q)$  &  $S(q)$  if required
- choose a set of starting parameters
- compute  $I(q)$
- compare the model calculation (theory) with the experimental data
- adjust some parameters
- iterate until an acceptable solution is found (hopefully...)



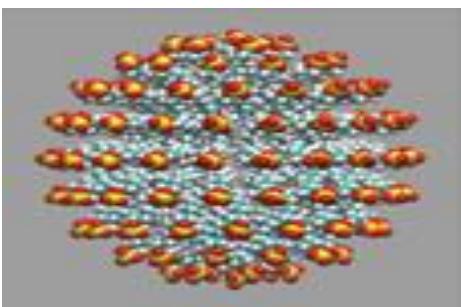
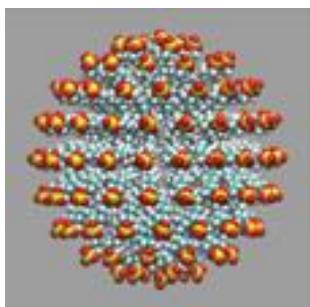
# (Classic) Model-Fitting Methods

## Objective:

- to find a model calculation (theory) that:
  - describes the form of the experimental data
  - offers minimal uncertainty
  - has converged nicely
- and above all is...

**physically realistic !!!**

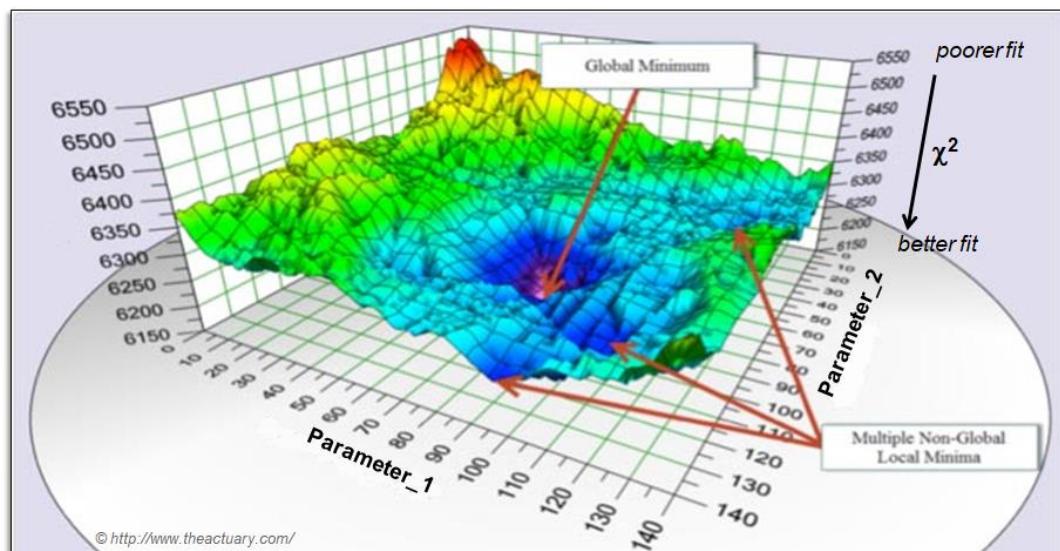
***"The fact that A model fits your data is NOT proof that it is THE appropriate model!"***



# (Classic) Model-Fitting Methods

- Where possible, select the model/parameters using **a priori knowledge**:
  - your instinct!
  - ancillary data (EM, DLS, etc)
  - published literature
- A model with more parameters is not always better!
- Fitting too many parameters at the same time can be a bad idea!

Need the program to find  
the global minimum on  
the  $\chi^2$  landscape, not a  
local minimum!

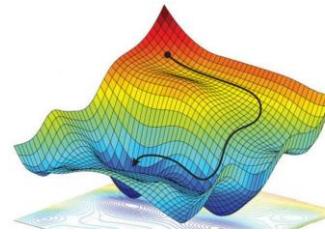


Science and  
Technology  
Facilities Council

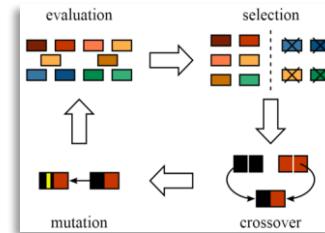
ISIS Neutron and  
Muon Source

# (Classic) Model-Fitting Methods

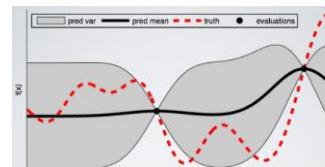
- Finding the global minimum → **optimisation**
  - you can have speed or robustness, but not (generally-speaking) both!
  - **For speed:** → gradient-descent algorithms
  - **For robustness:** → population algorithms
    - also called ‘evolutionary’ or ‘genetic’ algorithms
  - Bayesian (probability) algorithms
    - predict the future based on the past
  - AI methods...



© Alexander Amini



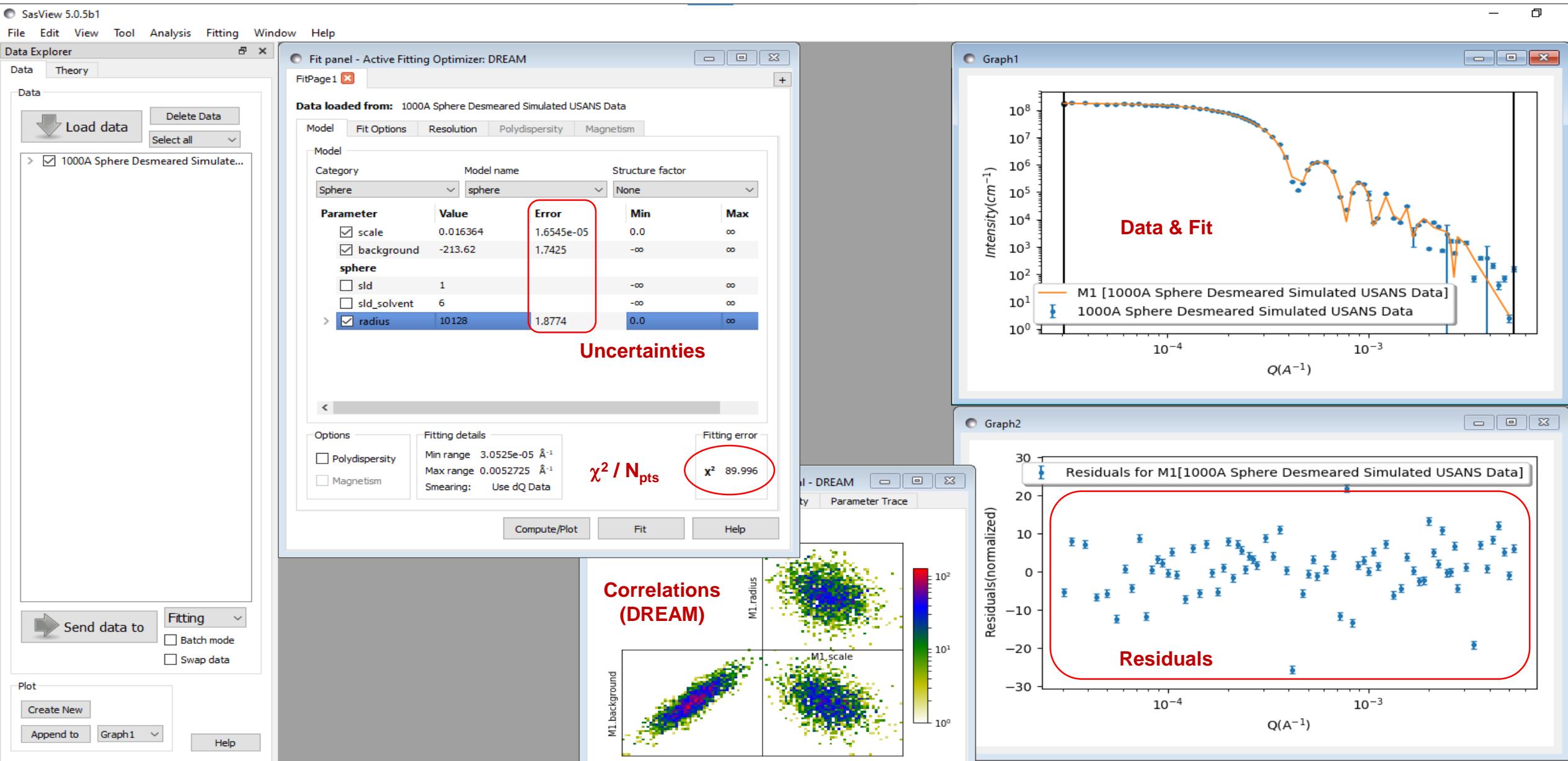
© Cheng Yu



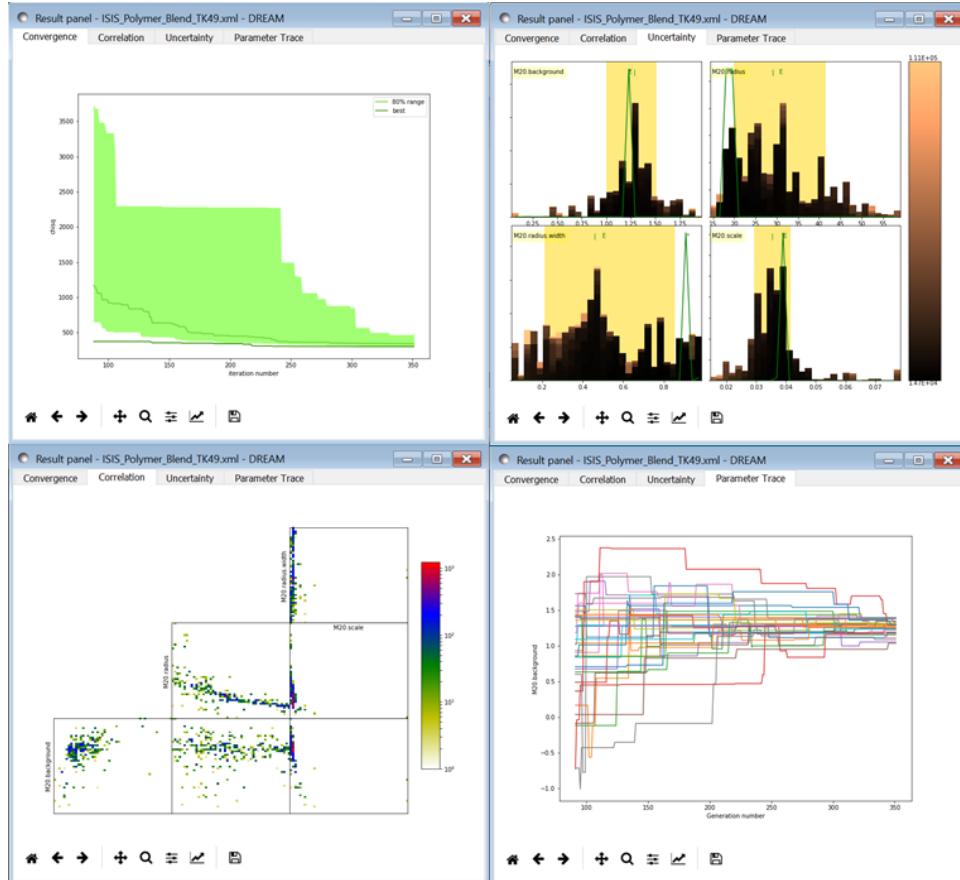
© Will Koehrsen

## SasView offers a choice of 5 optimisers

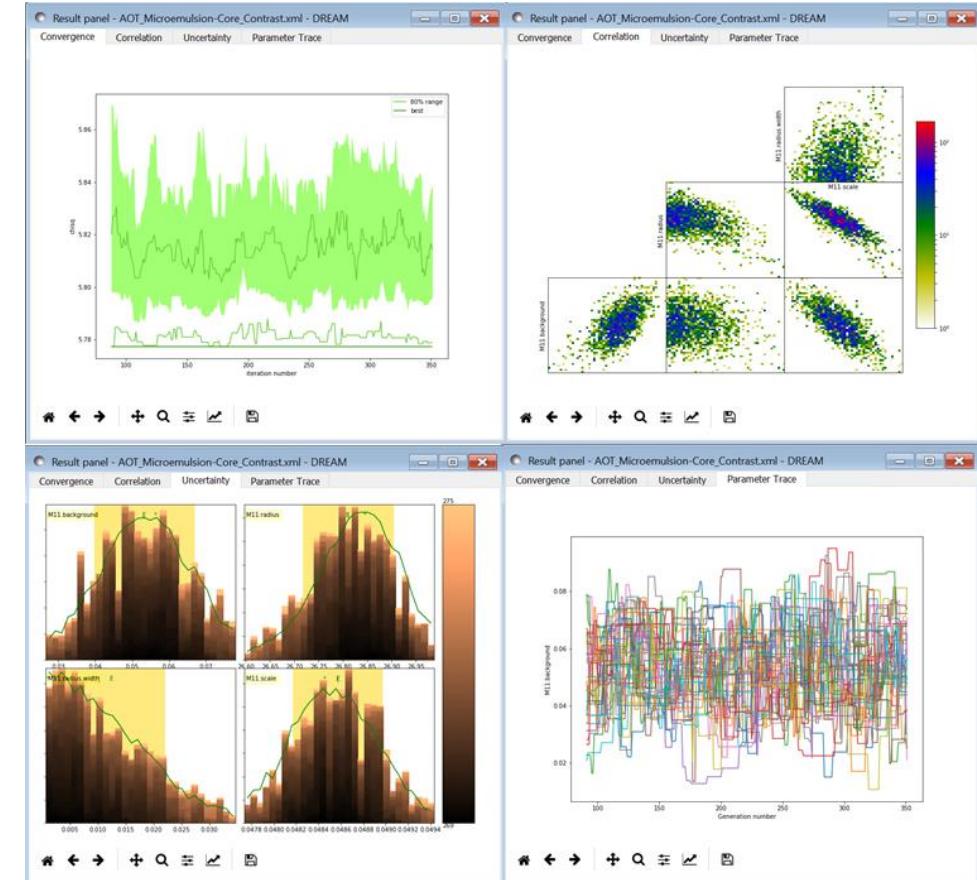
# Assessing Fit Quality



# Assessing Fit Quality



Poor fit !



Good fit !

Only available with the DREAM optimiser

<https://www.sasview.org/docs/user/qtgui/Perspectives/Fitting/optimizer.html#fit-dream>

# (Classic) Model-Fitting Methods

But there is an issue...

- what happens when the sample exhibits **dispersity in size?**

- consider a cylinder of radius  $R$  & length  $L$ , then:

$$'I(q)' = \Lambda \int_{R_{min}}^{R_{max}} \int_{L_{min}}^{L_{max}} N(R, L) V(R, L)^2 (\rho - \rho_{matrix})^2 P(q, R, L)_{cylinder} S(q, r) + B(q)$$

*NB: if the cylinders were aligned by shear / magnetic / electric field, would also need to integrate over a range of orientation angles! Every integral slows the calculation!*

- either:

1. pre-select the dispersity distributions as part of the model calculation, or
2. let the model-fitting 'discover' the dispersity distributions

'classic'

'advanced'

**SasView offers a choice of 6 dispersity distributions  
+ 2 user-supplied options**

# (Advanced) Model-Fitting Methods

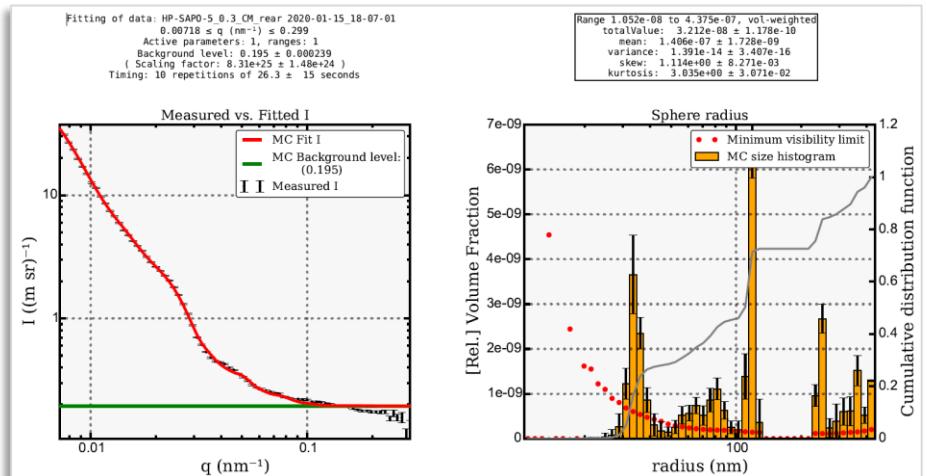
- search distribution-space *in addition* to parameter-space and return ‘form-free’ dispersity distributions

This functionality is not currently available in SasView

## McSAS

Bressler et al, (2015)

<https://github.com/BAMresearch/McSAS>

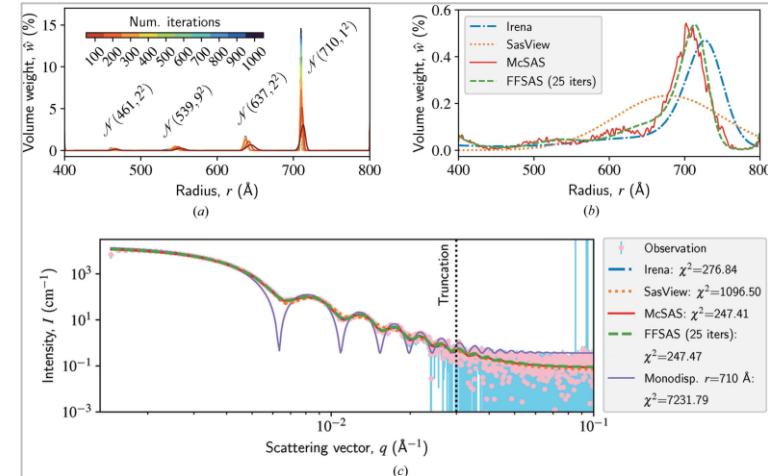


Uses iterative Monte-Carlo  
Very slow!

## FFSAS

Leng et al, (2022)

<https://github.com/stfc-sciml/ffsas>



Draws on ML concepts  
Very fast!

NB: also finds orientation distributions



Science and  
Technology  
Facilities Council

ISIS Neutron and  
Muon Source

# Linearised Model-Fitting Methods

- some rely on the limiting behaviour of  $P(q)$ :

when  $qR$  is ‘small’:

$$P(q, R)_{sphere} \approx \frac{\sin(qR)}{qR} \approx \left[1 - \frac{(qR)^2}{10}\right]^2 \approx \exp\left(-\frac{(qR)^2}{5}\right)$$

- so plot of  $\ln[I(q)]$  vs  $q^2$  is then linear with gradient:  $-R^2/5 \equiv -R_g^2/3$

$$\text{intercept: } \frac{M_z \phi (\Delta\rho)^2}{N_A \delta}$$

NB: non-linearity indicates polydispersity, aggregation or repulsive interactions!

alternatively:

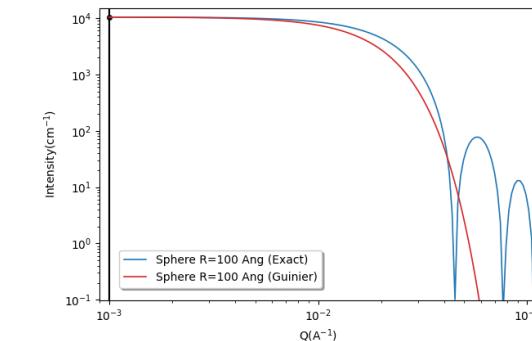
$$\frac{1}{P(q, R)_{sphere}} \approx 1 + \frac{(qR)^2}{5}$$

- so plot of  $[I(q)]^{-1}$  vs  $q^2$  is then linear with gradient:  $R^2/5 \equiv R_g^2/3$

$$\text{intercept: } \frac{N_A \delta}{M_z \phi (\Delta\rho)^2}$$

- implemented on a concentration series, can be used to extract the 2<sup>nd</sup> & 3<sup>rd</sup> virial coefficients

Guinier approximation, (1937)



Zimm approximation, (1948)



Difficulty is knowing when  $qR$  is ‘small’; many textbooks say  $qR < 0.8 – 1.3$ .  
Data must also cover maximum size present;  $q_{min} < (\pi / D_{max})$

# Linearised Model-Fitting Methods

when  $q$  is 'large':

$$I(q)_{q \rightarrow \infty} = \Lambda \frac{C_P}{q^4} + B(q)$$

Porod's 'law', (1951)

- so plot of  $[q^4 I(q)]$  vs  $q$  is then linear with gradient:  $C_P$  (the Porod Constant)
- assumes at sufficiently high- $q$  there is no shape information & the scattering is just reflections off sharp interfaces
- in a 2-phase system:  $C_P = 2\pi(\rho - \rho_{matrix})^2 S_V$  where  $S_V$  is the surface area per unit volume

NB:  $I(q)$  must be on an absolute scale to estimate  $S_V$ !

Porod's Law is actually a specific instance of **fractal scattering**:

$$I(q)_{q \rightarrow \infty} = \Lambda \frac{C_F}{q^n} + B(q)$$

where  $n = (2D_m - D_s) = (6 - \varepsilon)$  Hausdorff, (1918)

NB: all real systems have upper and lower '**cut-off lengths**' beyond which they are no longer fractal; be careful not to over-interpret the data!

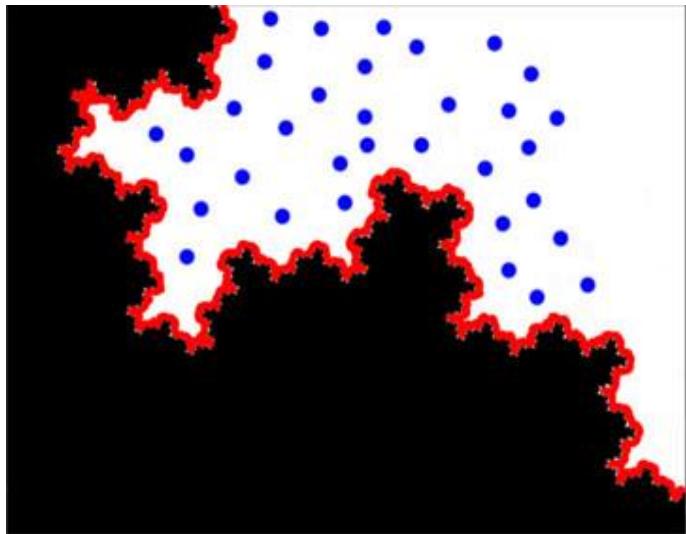
$D_m$  is the mass fractal dimension

$D_s$  is the surface fractal dimension

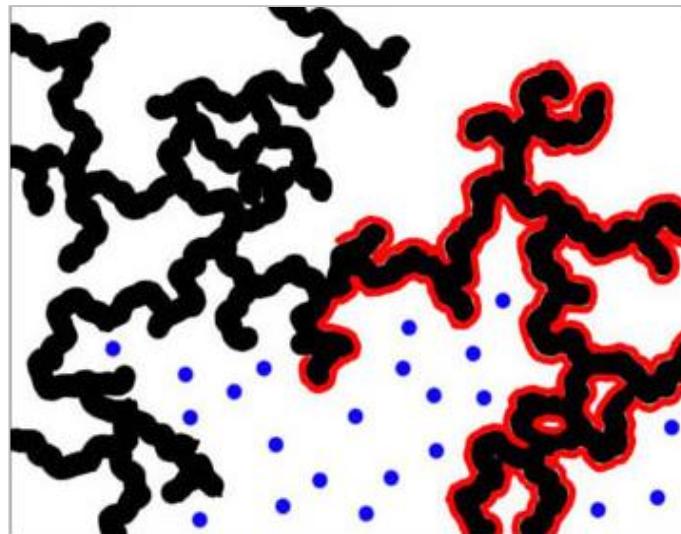
$\varepsilon$  is the Euclidean dimensionality (2,3)

# Fractals

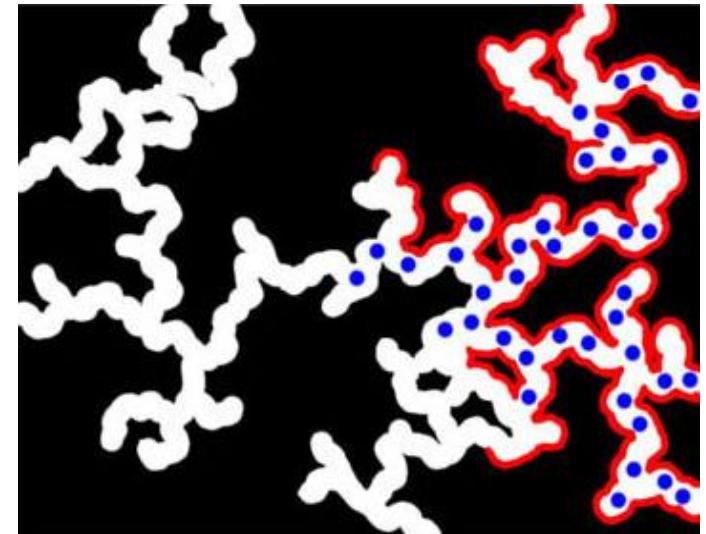
- objects that are ‘self-similar’ across a wide range of length scales
  - usually possess a dimensionality (the fractal dimension) outside Euclidean norms
  - in scattering terms the power law dependence should (ideally) extend over multiple decades in  $q$



*object with a fractal surface*



*mass fractal with a fractal surface*



*pore fractal with a fractal surface*

© [www.fractal.org/Life-Science-Technology/Publications/Pore-fractal.htm](http://www.fractal.org/Life-Science-Technology/Publications/Pore-fractal.htm)

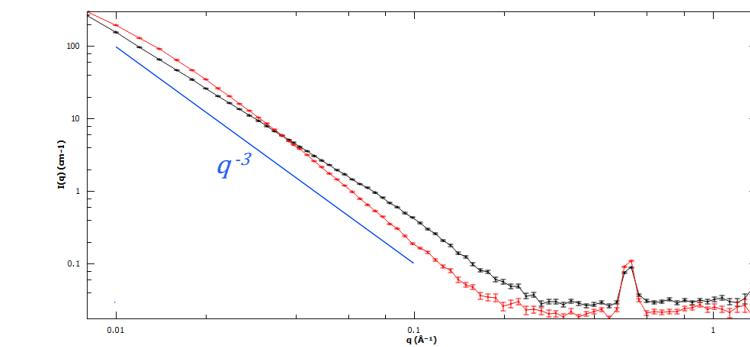
# Linearised Model-Fitting Methods

- some infer behaviour from the  $q$ -dependence:
  - so plot of  $\text{Log}[I(q)]$  vs  $\text{Log } q$  is linear with gradient:  $-n$

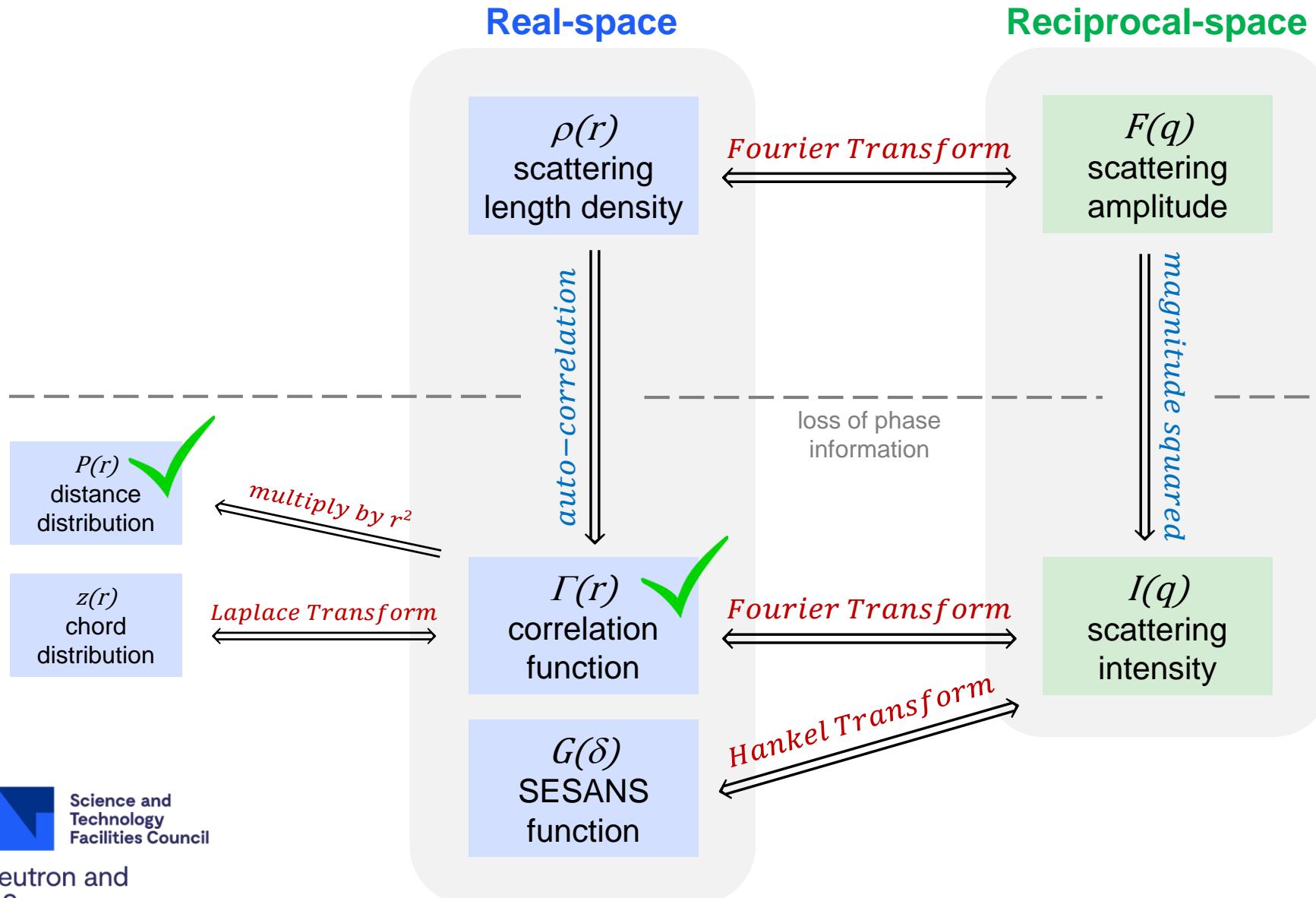
System	$-n$
Thin rods, filaments	1.0
Polymer chains with excluded volume (good solvent)	1.66
Semiflexible polymer chains with excluded volume	1.7
DLCA cluster	1.75 – 1.8
2D network	1.9
Polymer chains in theta solvent, lamellae, discs	2.0
RLCA cluster	2.1 – 2.25
Branched polymer chain in theta solvent	2.28
3D network (weakly segregated), percolation clusters	2.5
Volume fractal, pore fractal	3.0
3D network (strongly segregated), sharp interface	4.0

$n < 3$ : mass fractal

$n > 3$ : surface fractal



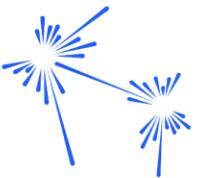
# The ‘Magic Square’



After Stribeck, (2007) & others



# Real-Space Methods



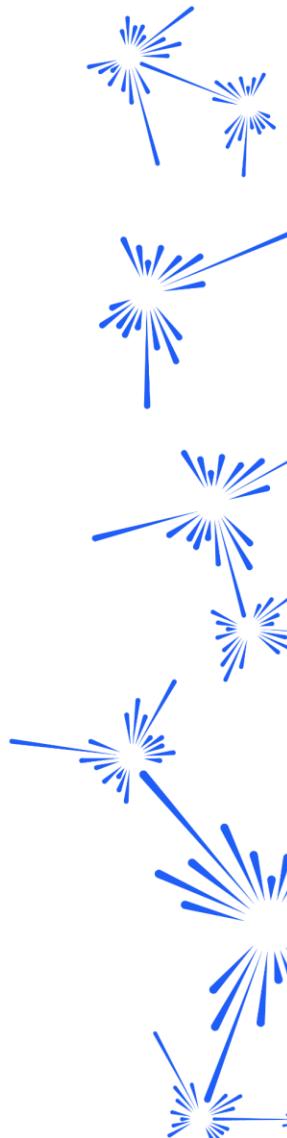
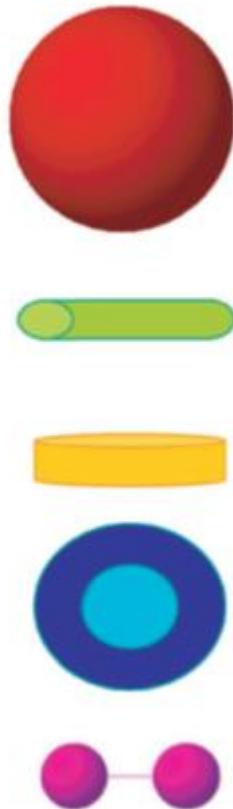
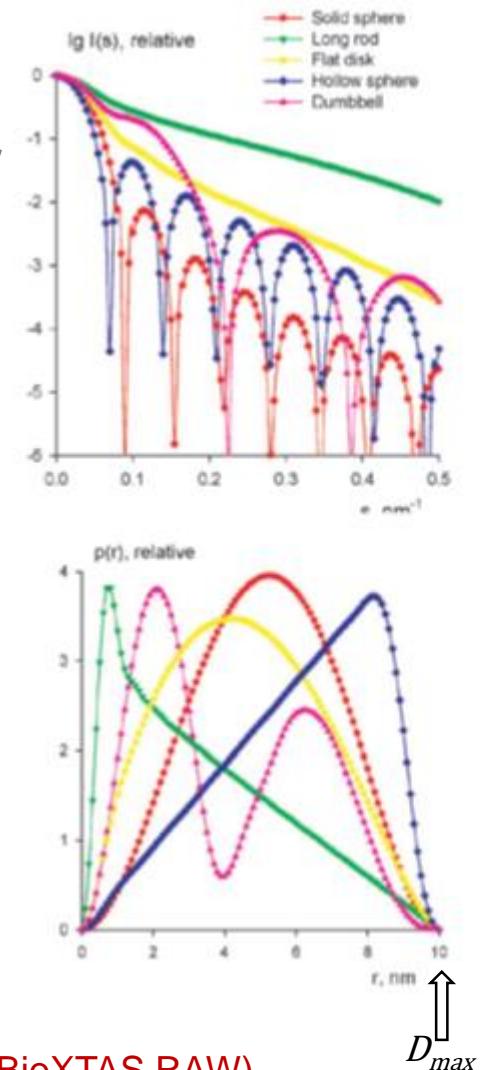
## (Pair) Distance Distribution Function, $P(r)$

- in essence, a **histogram of pair-wise distances** between atoms in the scatterer

$$P(r) = \frac{r^2}{2\pi^2} \int_0^\infty q^2 I(q) \frac{\sin(qr)}{qr} dq$$

- solve set of basis functions by *Indirect Fourier Transformation* (IFT)
- Glatter, (1977); Moore, (1980); Hansen & Pedersen, (1991)
- SasView implements the Moore method**

$$I_{q=0} = 4\pi \int_0^{D_{max}} P(r) dr \propto M_z \quad R_g^2 = \frac{\int_0^{D_{max}} r^2 P(r) dr}{2 \int_0^{D_{max}} P(r) dr}$$



**Other implementations:**

*GIFT* (from Glatter); *GNOM* (in ATSAS Suite); *BIFFT* (in BioXTAS RAW)

Svergun & Koch, (2003)



# Real-Space Methods

## PolyDisperse Spherical Pore (PDSP) Method

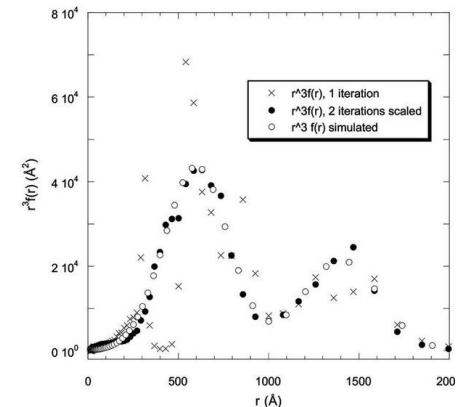
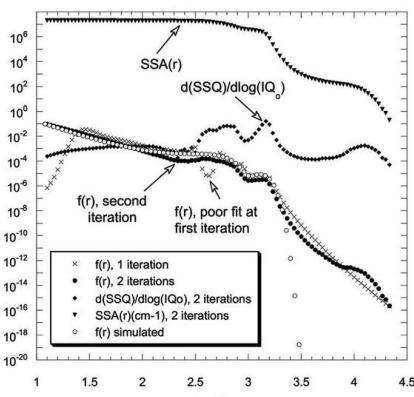
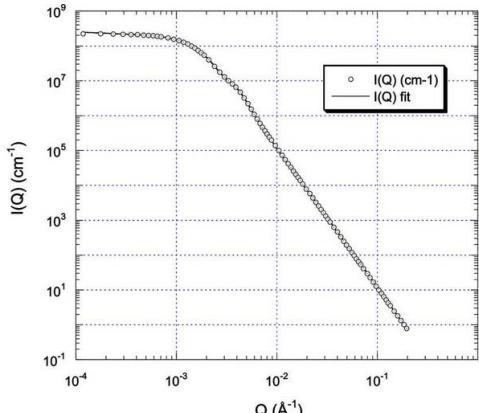
- **models the scattering from a fractal surface as that from a polydisperse ensemble of spheres**

$$I(\mathbf{q}) = \Lambda (\rho - \rho_{matrix})^2 \frac{\phi}{V} \int_{R_{min}}^{R_{max}} V(R)^2 D(R) P(Q, R) dR + B(q)$$

- iteratively extracts the size distribution  $D(R)$
- method typified by the *PRINSAS* software – developed for the geological community

*NB: unfortunately PRINSAS is now difficult to obtain & deploy*

**This functionality is not currently available in SasView**



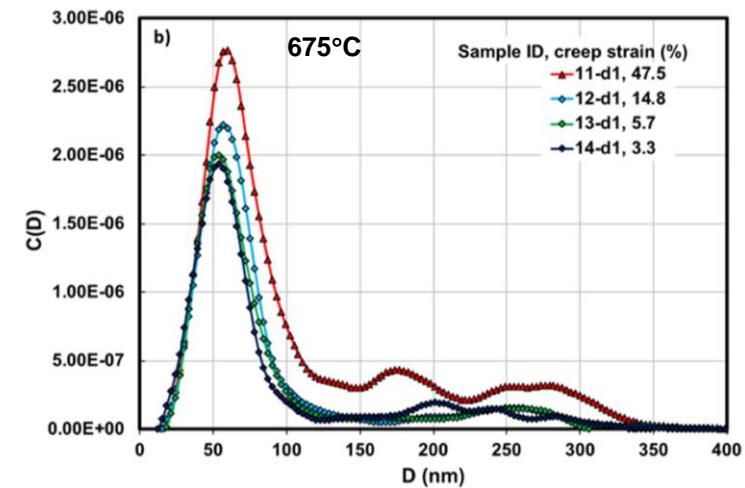
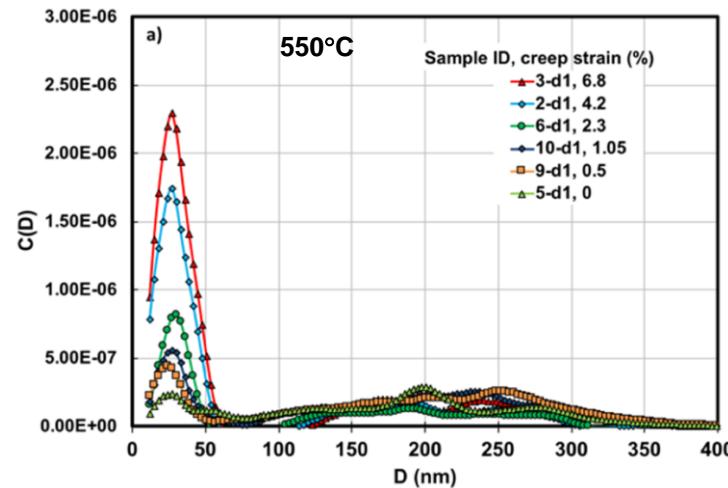
Hind, (2004)

# Real-Space Methods

## Maximum Entropy Method

- also **models the scattering from a fractal surface as that from a polydisperse ensemble of spheres**
  - also extracts the size distribution  $D(R)$ , but using Jaynes' '*Principle of Maximum Entropy*'
  - method typified by the *MAXE* software – developed for the metallurgy community
  - Potton, Daniell & Rainford, (1988)

**This functionality is not currently available in SasView**



Jazaeri et al, (2019)

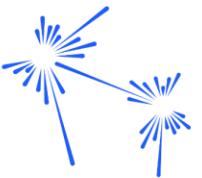


Science and  
Technology  
Facilities Council

ISIS Neutron and  
Muon Source



# Real-Space Methods

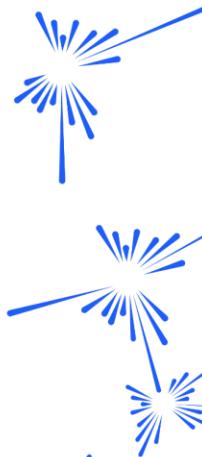
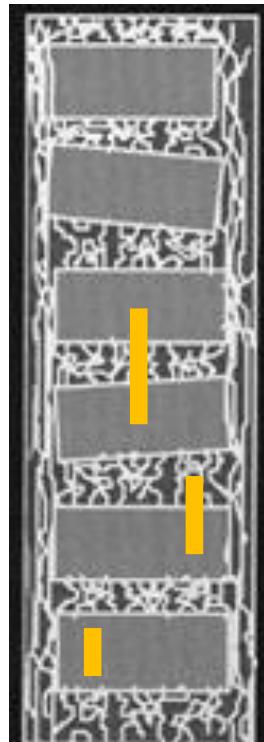
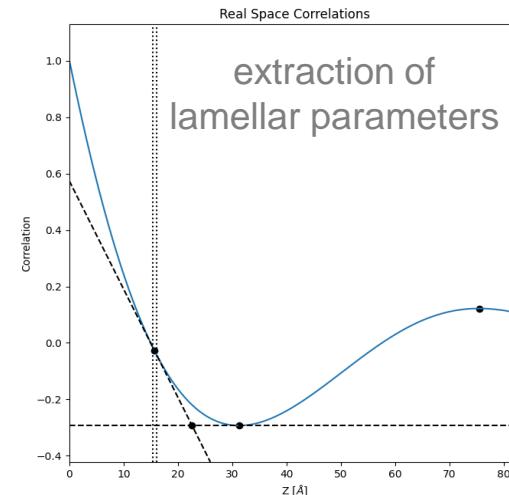
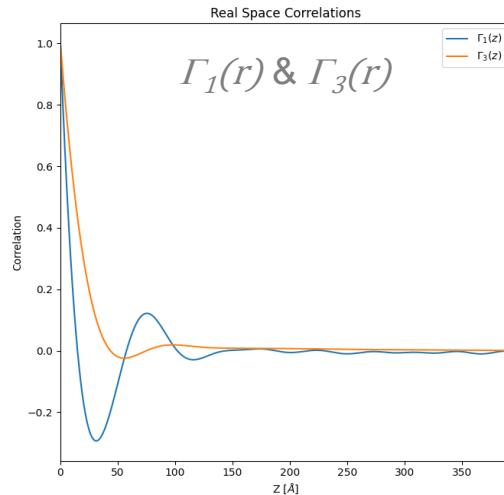
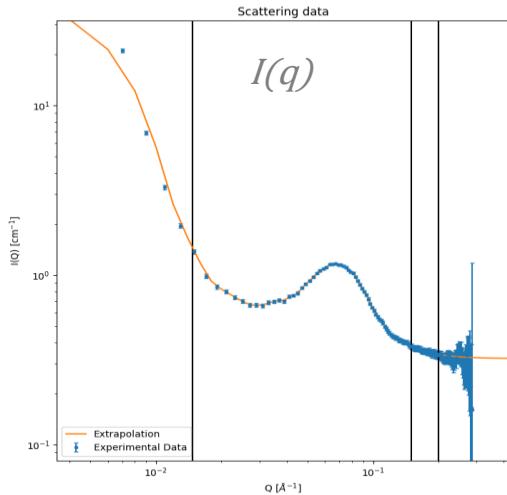


## Correlation Function, $\Gamma(r)$

- **describes spatial inhomogeneity in SLD in the sample**
  - in essence, the probability that rods of different lengths moving through the sample have equal SLD at either end

$$1D: \Gamma_1(r) = \frac{2}{Q^*} \int_0^\infty I(q) \cos(2\pi qr) dq$$

$$3D: \Gamma_3(r) = \int_0^{r_{max}} \frac{\Gamma_1(r)}{r} dr$$



Science and  
Technology  
Facilities Council

ISIS Neutron and  
Muon Source

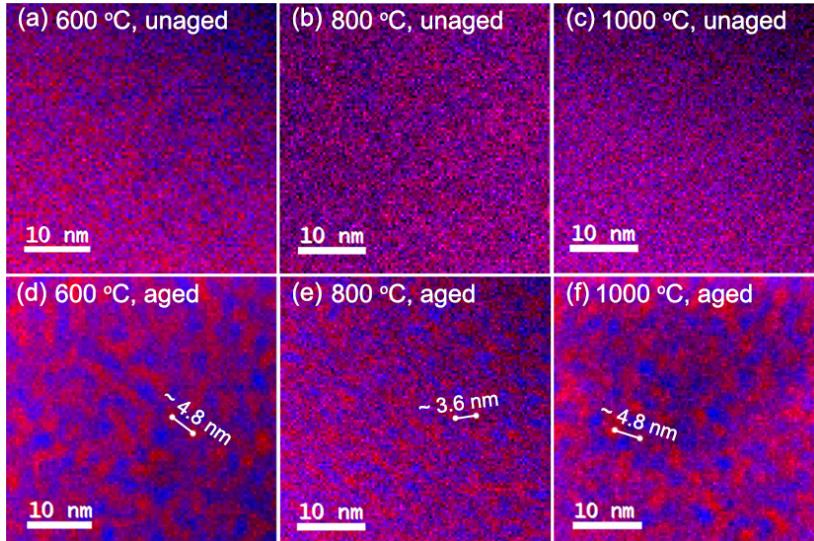
SasView computes  $\Gamma_1(r)$  &  $\Gamma_3(r)$  + IDF

# Real-Space Methods

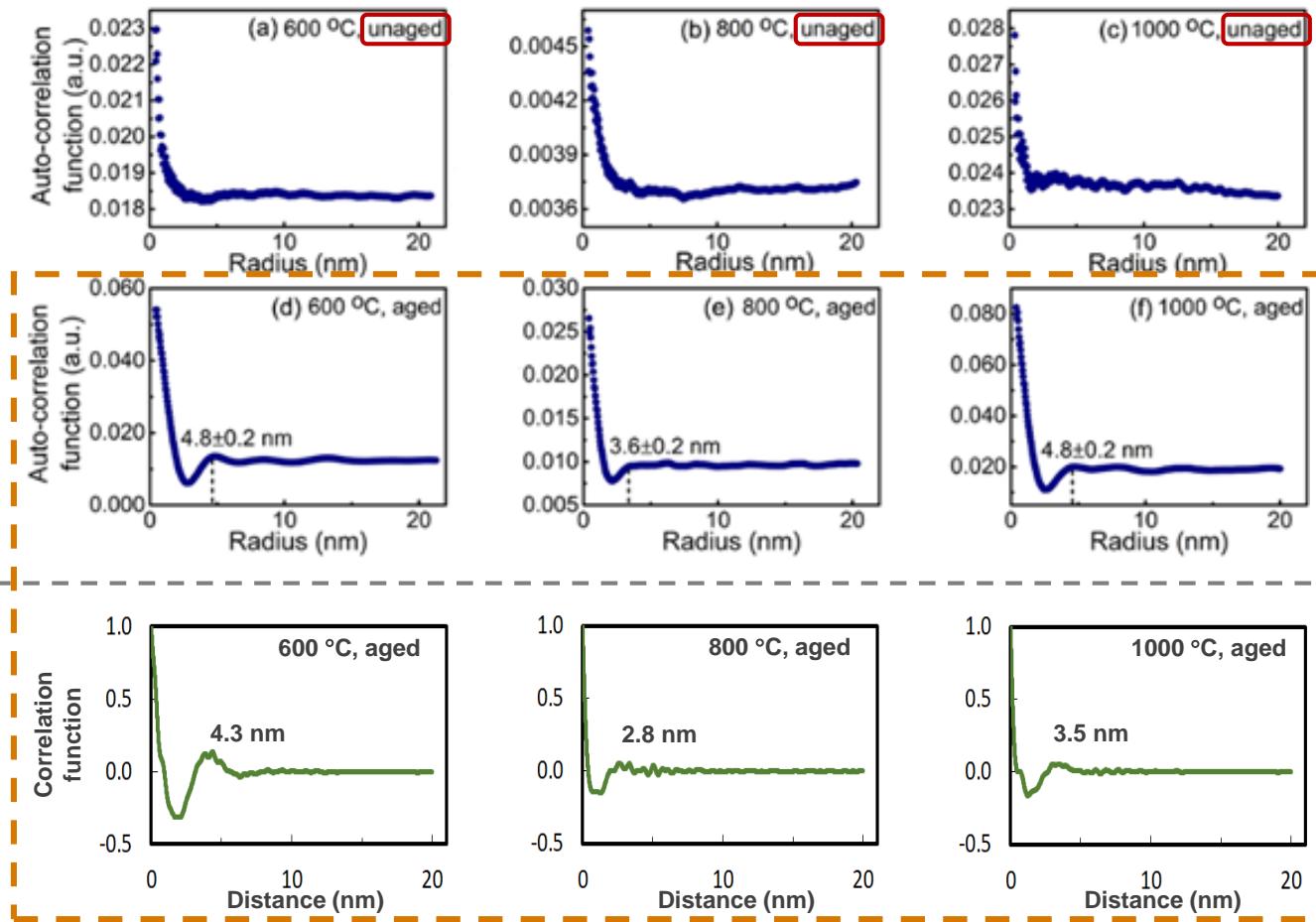
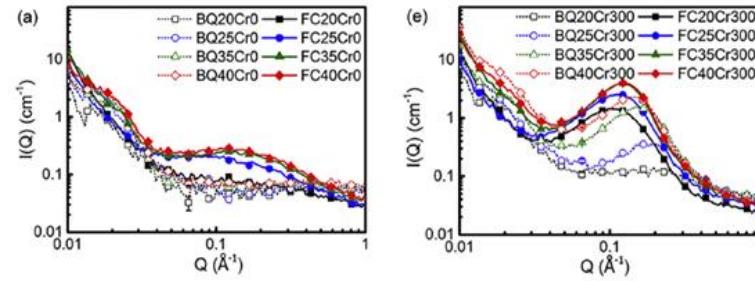
- application not limited to (quasi-)lamellar systems!
  - here is spinodal decomposition in a thermally-aged Fe-Cr alloy; Xu et al, (2017)

EELS

L<sub>3</sub>-edge:  
Fe (blue)  
Cr (red)



SANS



# Real-Space Methods

- few analytical expressions for  $\Gamma(r)$

- homogeneous sphere:  
(radius,  $R$ )

$$\Gamma(r) = 1 + \frac{3}{4} \frac{r}{R} + \frac{1}{16} \left( \frac{r}{R} \right)^3$$

where  $0 \leq (r/R) \leq 2$   
see Glatter & Kratky, (1982)

- **randomly-distributed 2-phase system:**  $\Gamma(r) = \exp\left(-\frac{r}{\xi}\right)$

Debye & Bueche, (1949)  
Debye, Anderson & Brumberger (1957)

$$I(q) = 8\pi\phi(1-\phi)(\Delta\rho)^2 \frac{\xi^3}{[1 + (q\xi)^2]^2}$$

- randomly-distributed 2-phase system:  
dual correlation lengths,  $\xi$  &  $\Xi$ )

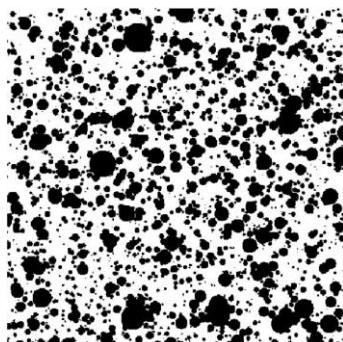
$$\Gamma(r) = f \exp\left(-\frac{r}{\xi}\right) + (1-f) \exp\left(-\frac{r^2}{\Xi^2}\right)$$

Debye, Anderson &  
Brumberger (1957)

$$I(q) = f K_1 \frac{\xi^3}{[1 + (q\xi)^2]^2} + (1-f) K_2 \Xi^3 \exp\left[-\frac{(q\Xi)^2}{4}\right]$$

$$K_1 = 8\pi\phi(1-\phi)(\Delta\rho)^2$$
$$K_2 = \pi^{3/2}\phi(1-\phi)(\Delta\rho)^2$$

Moritani et al, (1970)



Science and  
Technology  
Facilities Council

ISIS Neutron and  
Muon Source

# Other Methods

## Scattering Invariant, $Q^*$

- a quantity ***independent*** of the size, shape & spatial arrangement of the scatterers and only depending on the sum of their scattering
  - also called ‘Porod invariant’ & ‘scattering power’
  - also denoted in the literature as  $Q$ ,  $Q_s$ , and  $Z$ ! Be careful!
  - for ‘pinhole’ (ie, not slit-smeared) data from a 2-phase sample:

$$Q^* = \frac{1}{2\pi^2} \int_0^\infty q^2 I(q) dq = 2\pi^2 (\rho - \rho_{matrix})^2 \phi \phi_{matrix} = 2\pi^2 (\rho - \rho_{matrix})^2 \phi (1 - \phi)$$

Porod, (1951)

NB:  $I(q)$  must be on an absolute scale to estimate  $Q^*$ !

- calculation can be extended to 3 or more phases... with some complexity!

**Difficulty is the integration always extends beyond the range of the measured data!**

Requires extrapolations!

# Other Methods

- uses:

- estimating  $\phi$  if the contrast ( $\Delta\rho$ ) is known (**SasView does this!**), or
- estimating ( $\Delta\rho$ ) if  $\phi$  is known

*NB:  $\phi$  here represents the minority phase!*

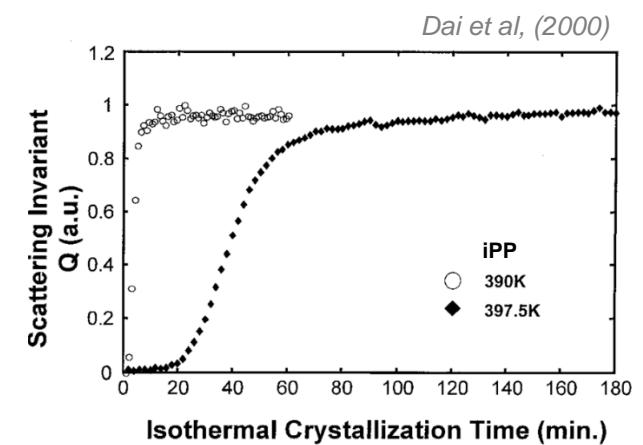
But... ‘*Babinet’s Principle*’ still applies!

- cross-calibrating SAXS & SANS data since:
- estimating surface area (recall Porod’s Law):
- discriminating between morphological models
- estimating degree of mixing
- monitoring kinetic processes (eg, crystallization)
- etc...



$$\frac{Q_{SAXS}^*}{(\Delta\rho)_{SAXS}^2} = \frac{Q_{SANS}^*}{(\Delta\rho)_{SANS}^2} = \text{Constant}$$

$$\frac{I(q)}{Q^*} \propto \frac{1}{q^4} \frac{S_V}{\phi(1-\phi)}$$

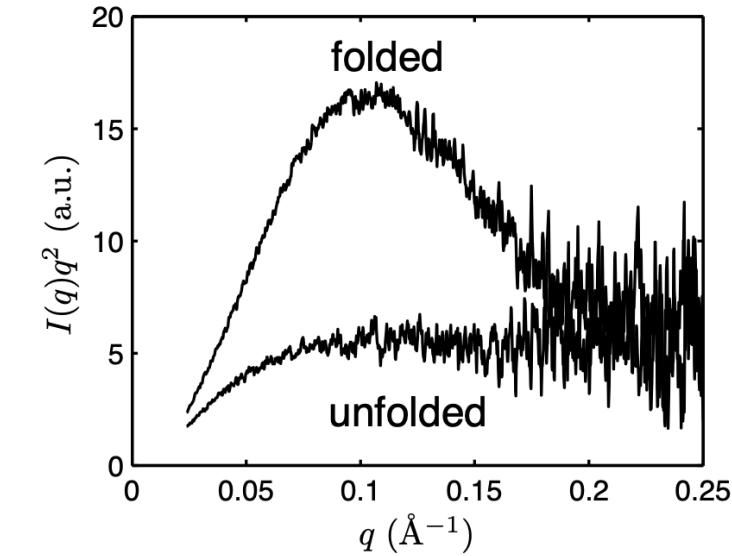
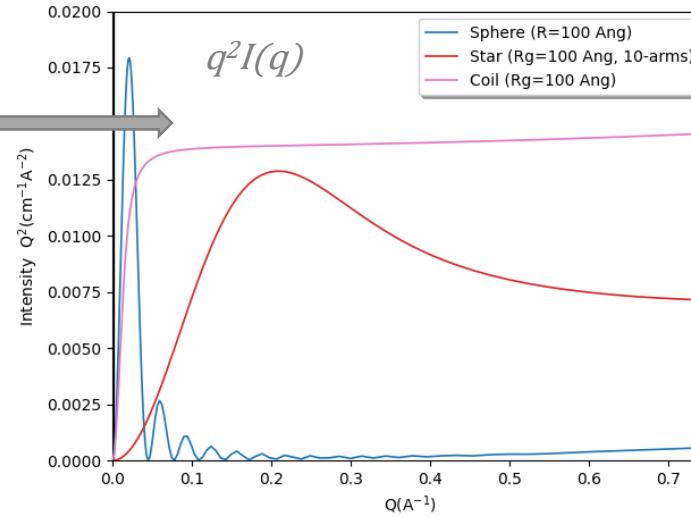
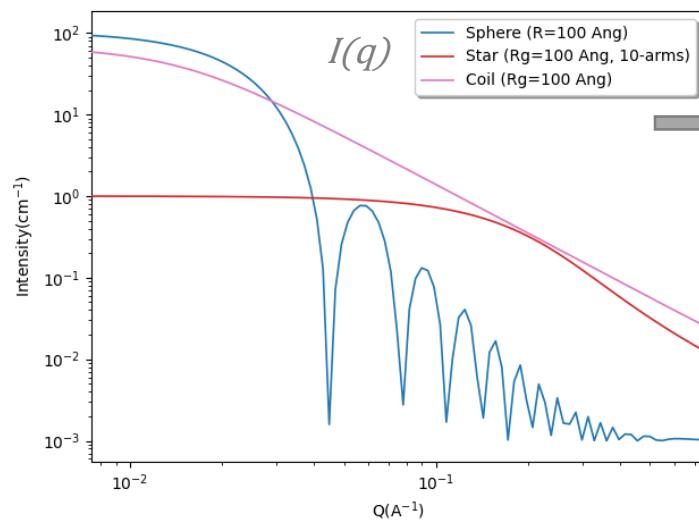


# Other Methods

## Kratky Plot

- Sensitive means of estimating the ‘compactness’ of the scatterer
  - normally plot:  $[q^2 \cdot I(q)]$  vs  $q$  but also alternative ‘dimensionless’ form:  $[ (qR)^2 \cdot I(q) / I(q=0) ]$  vs  $qR$

**NB: this method is very sensitive to the background subtraction!**



Ando et al, (2008)



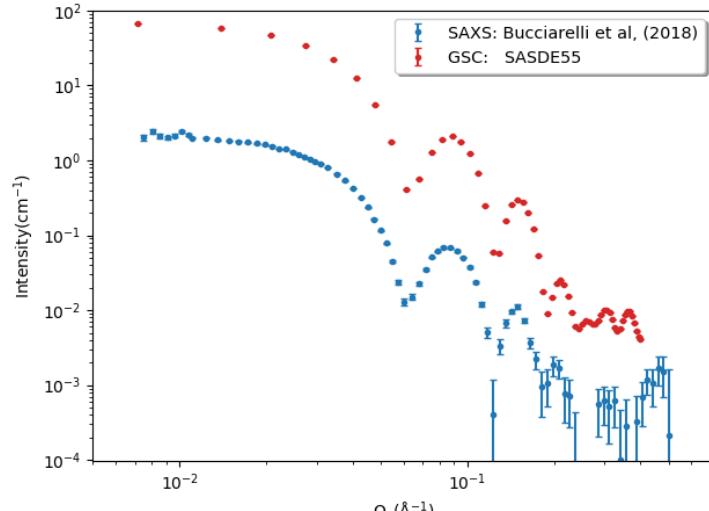
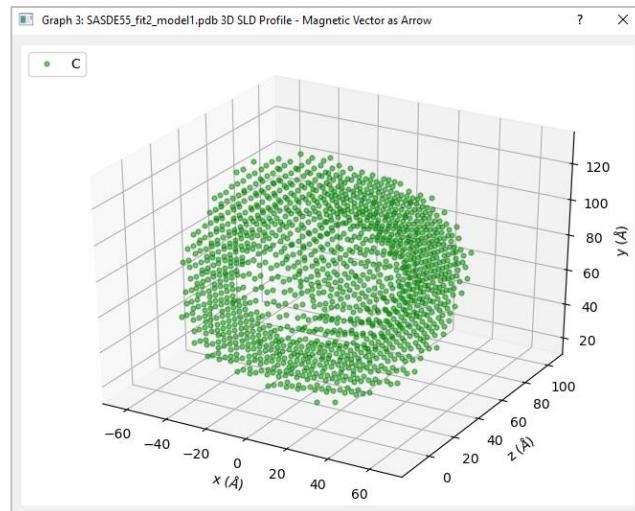
# Other Methods

## SasView Generic Scattering Calculator (GSC)

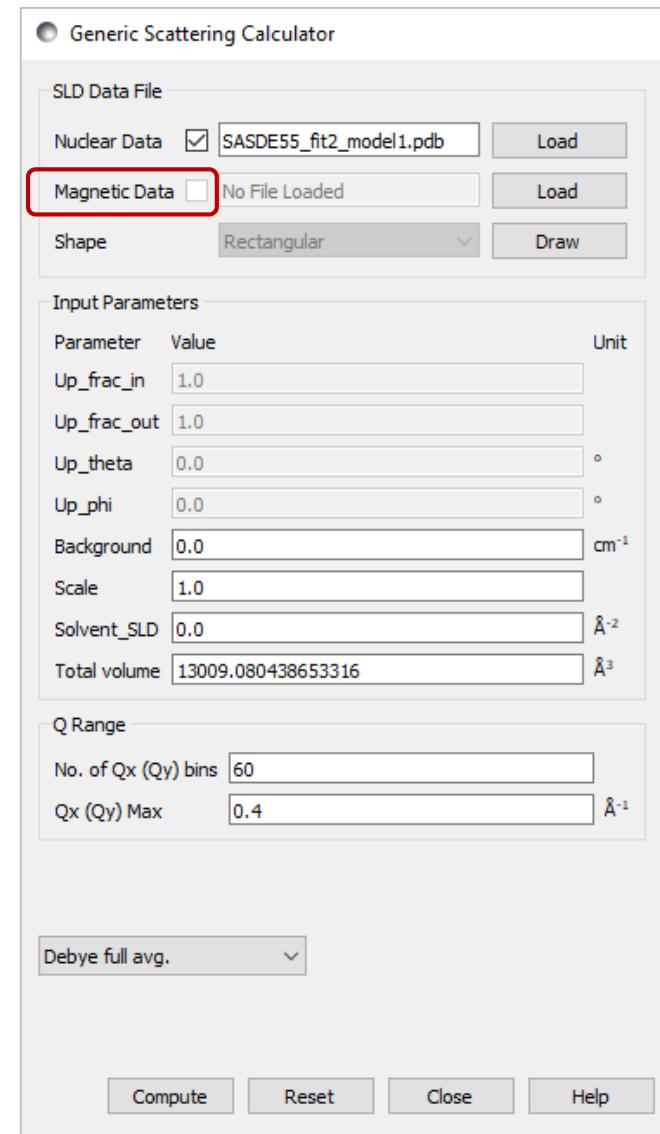
- Calculates  $I(q)$  from coordinate data
  - Supports .pdb, .omf (v1.0), .sld, .vtk ( $\leq$  v3.0) data formats

$$I(|\mathbf{q}|) = \frac{1}{V} \sum_j^N v_j \rho_j \sum_k^N v_k \rho_k \frac{\sin(|\mathbf{q}| |\mathbf{r}_j - \mathbf{r}_k|)}{|\mathbf{q}| |\mathbf{r}_j - \mathbf{r}_k|}$$

Debye, (1915)



NB: difference between these data is just the contrast and concentration!





# Grazie!



[www.isis.stfc.ac.uk](http://www.isis.stfc.ac.uk)



[@isisneutronmuon](https://twitter.com/isisneutronmuon)



[uk.linkedin.com/showcase/isis-neutron-and-muon-source](https://uk.linkedin.com/showcase/isis-neutron-and-muon-source)



[smallangle.org](http://smallangle.org)



[www.cansas.org](http://www.cansas.org)



[stephen.king@stfc.ac.uk](mailto:stephen.king@stfc.ac.uk)



# Sasview and SAS Data Analysis

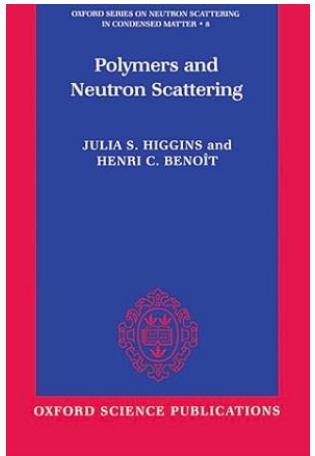
Dr Stephen KING

TNT 2024, San Giovanni, Italy  
17 Jun 2024

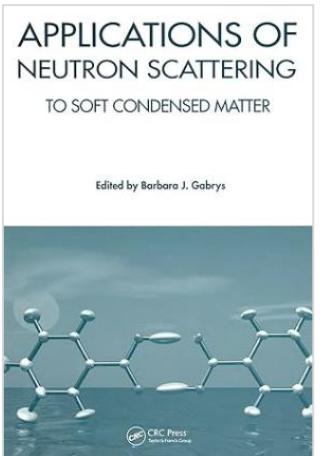


## Useful References

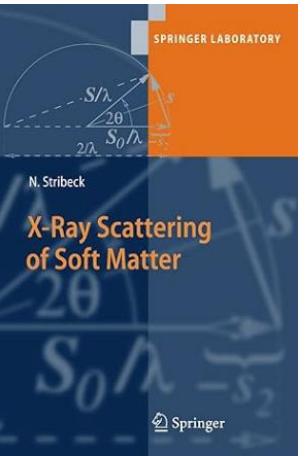




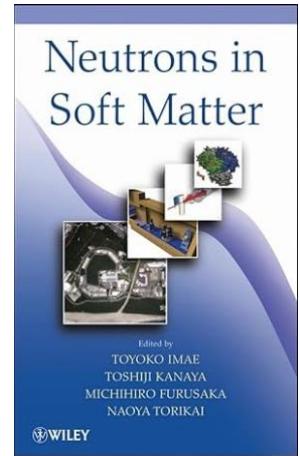
Higgins & Benoit  
1997



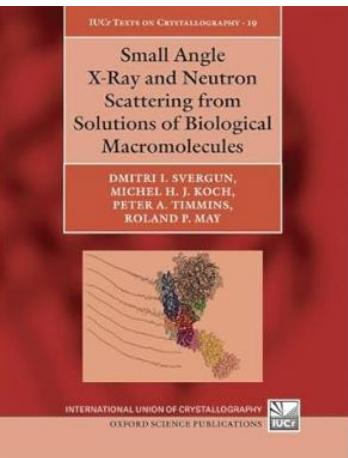
Gabrys  
2000



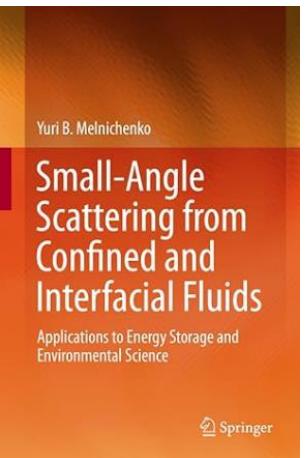
Stribeck  
2007



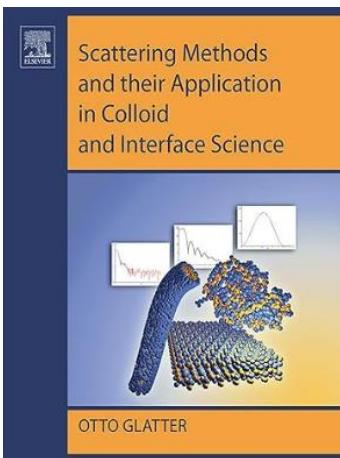
Imae, Kanaya,  
Furusaka, Torikai  
2011



Svergun, Koch,  
Timmins & May  
2013



Melnichenko  
2016



Glatter  
2018



# Sasview and SAS Data Analysis

Dr Stephen KING

TNT 2024, San Giovanni, Italy  
17 Jun 2024



## Supplementary Content



# The CanSAS Standards

## CanSAS1D

- v1.0 released May 2009
- v1.1 released Mar 2013
- for 1D *reduced* data only
- XML-based (.xml)
- still human-readable

<http://www.cansas.org/formats/canSAS1d/1.1/doc/>

## NXcanSAS

- v1.0 released Jan 2017
- for nD *reduced* data
- HDF5-based (.h5)
- NeXus-structured
- not human-readable

<http://www.cansas.org/wgwiki/index.php/NXcanSAS>

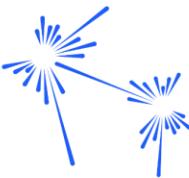
<https://www.nexusformat.org/>

<https://www.hdfgroup.org/downloads/hdfview/> (free!)

**HDF® VIEW**



# SasView Optimisers



- For speed: → gradient-descent algorithms
  - **Levenberg-Marquardt**
    - the tried-and-trusted workhorse found in every model-fitting package
    - use when you have a reasonable fit in a minimum & want the best values
    - but beware it finding local minima
  - **Quasi-Newton BFGS**
- In-between:
  - **Nelder-Mead Simplex**
    - a good first choice (especially if your function is well-behaved)
- For robustness: → population & Bayesian algorithms
  - **Differential Evolution**
  - **DREAM**
    - use when you need a really robust fit &/or want to explore correlations

<https://pypi.org/project/bumps/>

Paul Kienzle, NIST

'bumps'



# Assessing Fit Quality

## Residuals

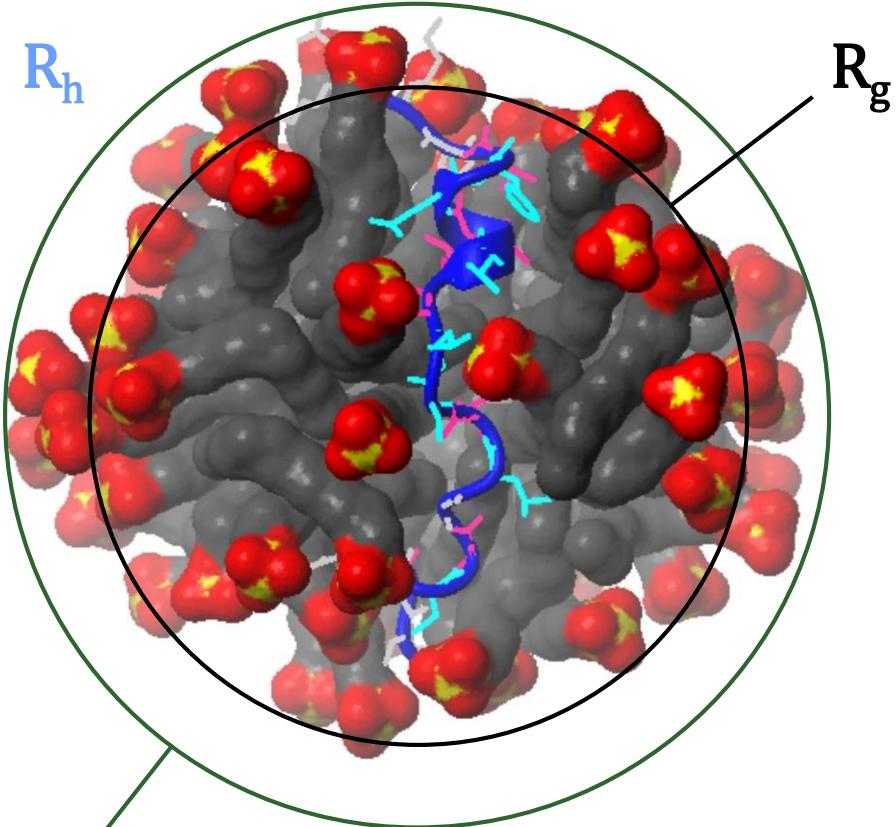
- $Residual_{i,expt} = I(Q)_{i,expt} - I(Q)_{i,model}$
- Good fit as  $Residual \rightarrow 0$
- SasView actually calculates ‘normalised residuals’:

$$Norm. Residual_{i,expt} = \frac{I(Q)_{i,expt} - I(Q)_{i,model}}{Error\_I(Q)_{i,expt}}$$

## $\chi^2 / N_{pts}$ (‘normalised $\chi^2$ ’)

- Also called the ‘goodness-of-fit’ parameter
- $$\frac{\chi^2}{N_{pts}} = \frac{\left\{ \sum_{i=1, N_{pts}} \left[ \frac{(I(Q)_{i,expt} - I(Q)_{i,model})^2}{(I(Q)_{-Error_i})^2} \right] \right\}}{N_{pts}}$$
- Good fit as  $\chi^2 / N_{pts} \rightarrow 0$
- NB:  $normalised \chi^2 \approx reduced \chi^2$  when  $N_{pts} \gg$  number of parameters

# Radius-of-Gyration, $R_g$



SDS micelle with solubilized peptide  
Perdih et al, (2012)

Hard-Sphere Radius,  $R_{HS} \sim 20 \text{ \AA}$   
Radius-of-Gyration,  $R_g \sim 16 \text{ \AA}$   
Hydrodynamic Radius,  $R_h$

$R_g$  is the root-mean-square distance from the centre-of-mass (~centre-of-SLD)

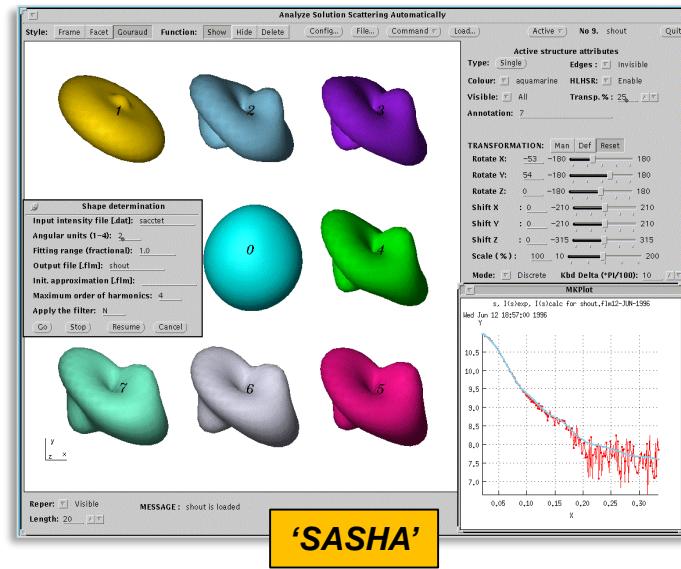
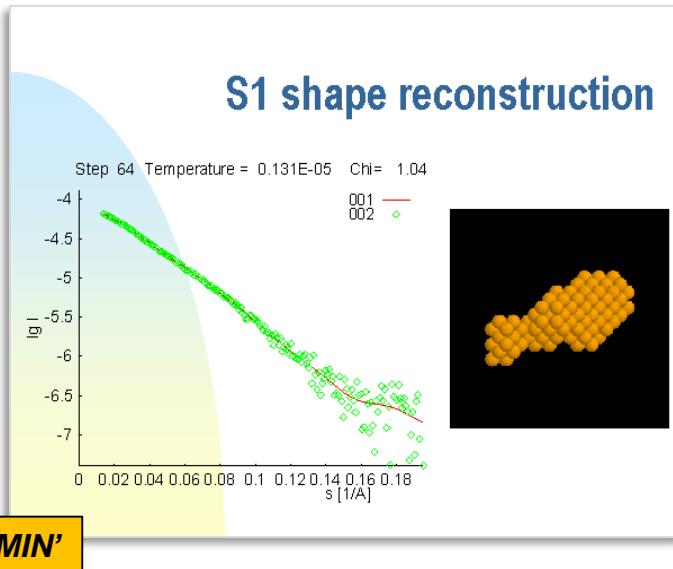
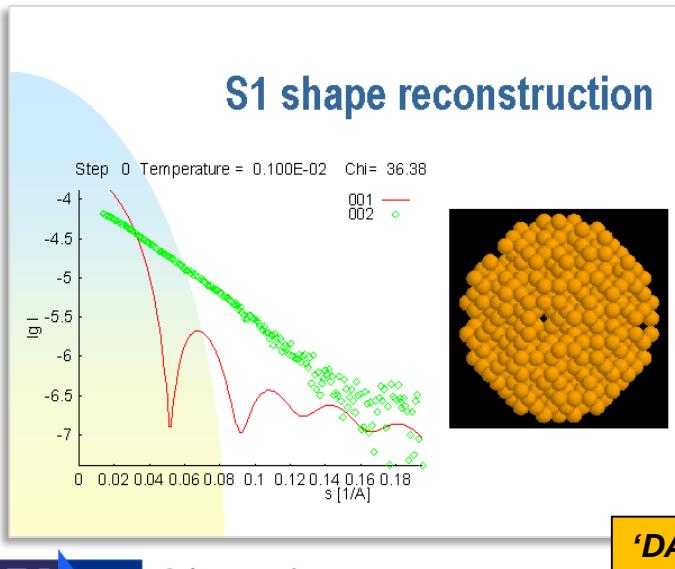
$$R_g^2 = \frac{1}{N} \sum_i \|\mathbf{r}_i - \mathbf{r}_{com}\|^2 = \frac{\int_0^{D_{max}} r^2 P(r) dr}{2 \int_0^{D_{max}} P(r) dr}$$

Object	Formula
Sphere (radius, $R$ )	$R_g^2 = \frac{3}{5} R^2$
Cylinder (radius, $R$ ; length, $L$ )	$R_g^2 = \frac{R^2}{2} + \frac{L^2}{12}$
Ellipsoid (semi-axes, $a$ & $b$ )	$R_g^2 = \frac{a^2 + b^2}{4}$
Gaussian polymer ( $N$ segments of length $l$ )	$R_g^2 = \frac{Nl^2}{6}$
Star polymer ( $f$ arms)	$R_g^2 = \left(3 - \frac{2}{f}\right) \frac{Nl^2}{6}$
Helix	Muroga, (1992); Eqn 49

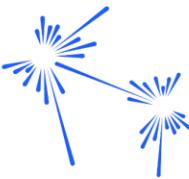
# Ab-initio Methods

- describe the macromolecule / micelle / etc as a **3D assembly of spheres** or as a **3D isovolume**
  - then iterate to find a ‘shape envelope’ consistent with the measured  $I(q)$
  - method typified by the ATSAS Suite: <https://www.embl-hamburg.de/biosaxs/software.html>

*NB: no chemical bonding or thermodynamics involved!*

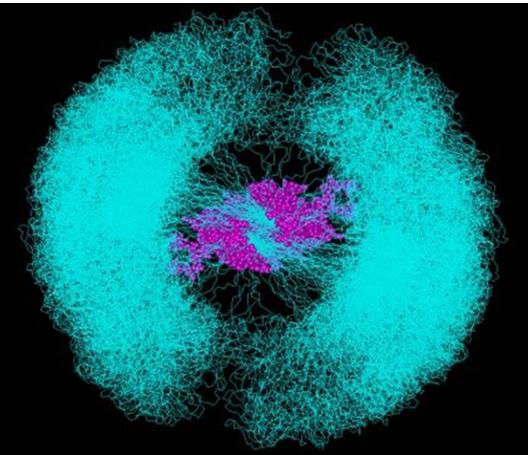
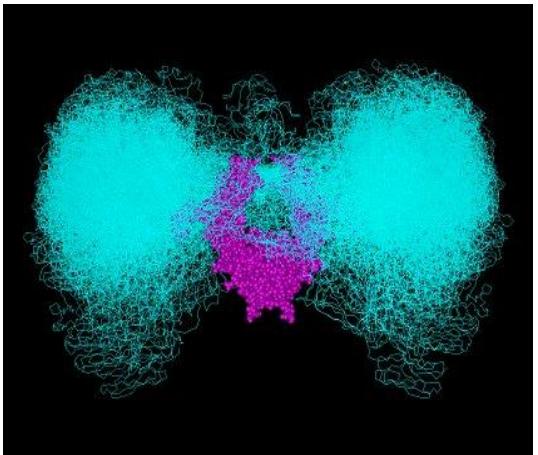


# MD/MC Methods (Biological)

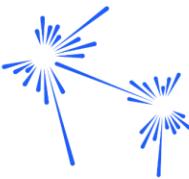


- Use MD methods to build **atomistic/'coarse-grained'** structures constrained by  $I(q)$  and MC methods to equilibrate them
  - typically applied in biological solution scattering
  - take starting structures from PDB and SASPDB databases
  - method typified by SASSIE-Web: <https://www.ccpsas.org/>

*NB: need starting structures! not fast!*



# MD/MC Methods (Soft Matter)



- Use MD methods to build **atomistic/'coarse-grained' structures** and MC methods to equilibrate them
  - build starting structures using SMILES-strings, Gromacs, etc, and output as .pdb file
  - method typified by Shapespyer: <https://www.scd.stfc.ac.uk/Pages/Shapespyer.aspx>
  - intended for mainstream soft matter; eg surfactants, lipids, etc
  - use SasView GSC to compute the ensemble scattering!

*NB: in development! collaborators wanted!*

