

core_multishell_cylinder

core multishell cylinder or disc

Parameter	Description	Units	Default value
scale	Scale factor or Volume fraction	None	1
background	Source background	cm ⁻¹	0.001
sld_core	Core scattering length density	10 ⁻⁶ Å ⁻²	5
radius_core	Radius of the core	Å	35
length_core	Radius of the core	Å	80
sld_solvent	Solvent scattering length density	10 ⁻⁶ Å ⁻²	6.4
sigma	Interfacial roughness	Å	0
n_shells	number of shells (must be integer)	None	1
sld_shell[n_shells]	scattering length density of shell k	10 ⁻⁶ Å ⁻²	1
thickness[n_shells]	Thickness of cylinder shell k	Å	20
face[n_shells]	Thickness of disc (or cylinder end) shell k	Å	10
theta	cylinder axis to beam angle	degree	60
phi	rotation about beam	degree	60

The returned value is scaled to units of cm⁻¹ sr⁻¹, absolute scale.

Definition

Core multi-shell cylinders or discs.

There must be a minimum of ONE shell (May set its sld to match solvent, thickness1 and face1 to zero or very small so the shell does not contribute to the normalisation volume.)

There may be numerical integration issues at extremes of Q and/or with extreme aspect ratios - particles which are very large in some dimensions compared to others.

2d scattering is so far only minimally tested.

Use of S(Q) with this I(Q) currently not giving correct I(Q) in sasview, (though passes unit tests), due to a more general sasview v5 bug.

The sld profile plot show profiles along both radius and half length simultaneously! (A simple edit of the py code will change which is displayed.)

Scattering is normalised to $V_{total} = \pi \cdot Router^2 Louter$ the outer volume of the particle.

$$Louter = length_core + 2 * (face1 + face2 + face3 + \dots)$$

$$Router = radius_core + thickness1 + thickness2 + thickness3 + \dots$$

Scattered intensity is calculated by a numerical integration over angle α between the axis of the cylinder and \vec{q} ,

$$I(q, \alpha) = \frac{scale}{V_{total}} F^2(q, \alpha) \cdot \sin(\alpha) + background$$

where

$$F(q, \alpha) = \sum_{k=0}^n \left[(\rho_k - \rho_{k+1}) V_k \frac{\sin(q \cdot h_k \cos \alpha)}{q \cdot h_k \cos \alpha} \frac{2J_1(q \cdot r_k \sin \alpha)}{q \cdot r_k \sin \alpha} \right] \exp \left\{ -\frac{1}{2} q^2 \sigma^2 \right\}$$

and

$$\rho_0 = sld_core$$

$$\rho_k = sld_shell_k \quad \text{for } k = 1 \text{ to } n, \text{ the number of shells}$$

$$\rho_{n+1} = sld_solvent$$

$$r_0 = radius_core$$

$$r_1 = r_0 + thickness_1$$

$$r_k = r_{k-1} + thickness_k \quad \text{for } k = 1 \text{ to } n$$

$$h_0 = \frac{1}{2} length_core$$

$$h_1 = h_0 + face_1$$

$$h_k = h_{k-1} + face_k \quad \text{for } k = 1 \text{ to } n$$

$$V_k = 2\pi \cdot h_k \cdot R_k^2$$

J_1 is the first order Bessel function.

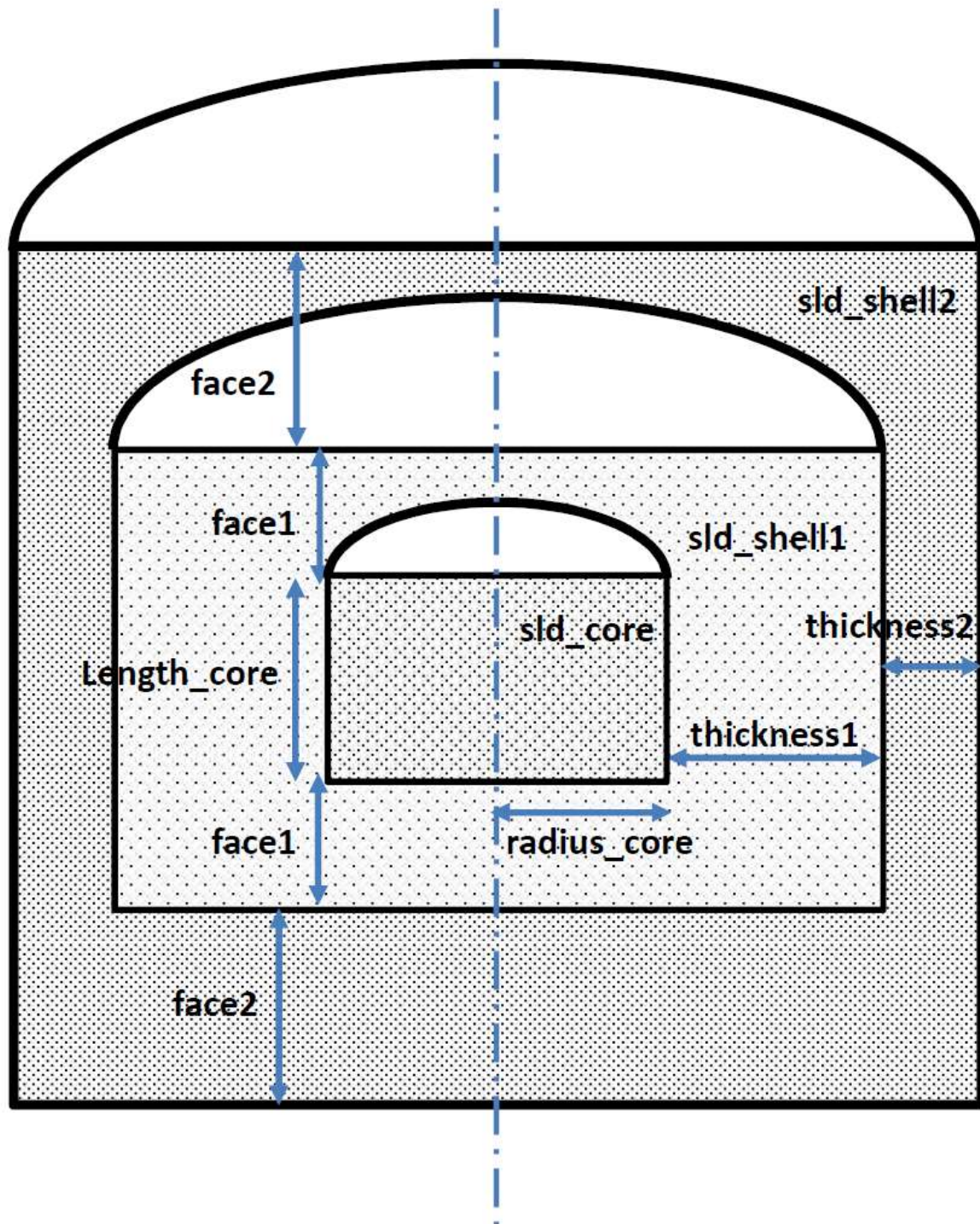


Fig. 7 Core multishell cylinder or disc - schematic cross section with two shells.

An approximation for the effects of “Gaussian interfacial roughness” σ is included, by multiplying $I(Q)$ by $\exp\left\{-\frac{1}{2}Q^2\sigma^2\right\}$. This applies, in some way, to all interfaces in the model not just the external ones. (Note that for a one dimensional system convolution of the scattering length density profile with a Gaussian of standard deviation σ does exactly this multiplication.) Leave σ set to zero for the usual sharp interfaces. There is some debate as to whether the factor of 1/2 is needed or not.

Since the number of parameters may become too large to fit well, fix or constrain as many as possible. e.g. to constrain face1 to thickness1, click on the first, ctrl/click the second, right click and select “mutual constraint”. Note that such constraints are NOT applied inside integrations

over polydispersity, they are only applied to starting values. To do this the constraints would need to be included inside a customized model.

To provide easy access to the orientation of the core-shell cylinder, we define the axis of the cylinder using two angles θ and ϕ . (see [cylinder model](#))

The θ and ϕ parameters are not used for the 1D output.

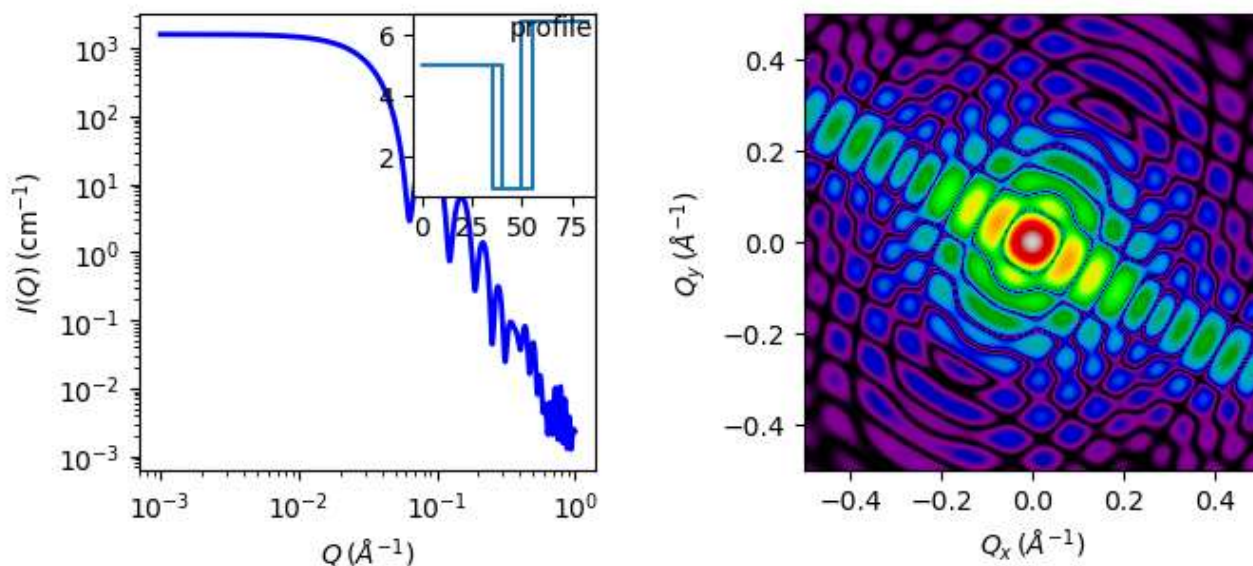


Fig. 8 1D and 2D plots corresponding to the default parameters of the model.

Source

`core_multishell_cylinder.py` ★ `core_multishell_cylinder.c` ★ `gauss76.c` ★ `sas_J1.c` ★ `polevl.c`

Reference

See also Livsey [1] and Onsager [3].

- [1] I Livsey, *J. Chem. Soc., Faraday Trans. 2*, 83 (1987) 1445-1452
- [2] S R Kline, *J Appl. Cryst.*, 39 (2006) 895
- [3] L. Onsager, *Ann. New York Acad. Sci.*, 51 (1949) 627-659

Authorship and Verification

- **Author:** Richard Heenan **Date:** April 2021
- **Last Modified by:** Richard Heenan **Date:** April 2021
- **Last Reviewed by:** Richard Heenan **Date:** April 2021