

BornAgain

Form Factor Catalog

Software version 1.17

This document last updated May 11, 2020

Céline Durniak, Marina Ganeva, Gennady Pospelov, Walter Van Herck,
Joachim Wuttke

Scientific Computing Group
Jülich Centre for Neutron Science
at Heinz Maier-Leibnitz Zentrum Garching
Forschungszentrum Jülich GmbH

Homepage: <http://www.bornagainproject.org>

Copyright: Forschungszentrum Jülich GmbH 2013–2020

Licenses: Software: GNU General Public License version 3 or higher
Documentation: Creative Commons CC-BY-SA

Authors: Céline Durniak, Marina Ganeva, Gennady Pospelov, Walter Van Herck, Joachim Wuttke
Scientific Computing Group
at Heinz Maier-Leibnitz Zentrum (MLZ) Garching

Disclaimer: Software and documentation are work in progress.
We cannot guarantee correctness and accuracy.
If in doubt, contact us for assistance or scientific collaboration.

Funding: This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 654000.

Contents

Preface	4
About BornAgain	4
Citation	4
1 Introduction	5
2 Hard particles	8
2.1 AnisoPyramid (rectangle-based)	10
2.2 Box (cuboid)	12
2.3 Cone (circular)	14
2.4 Cone6 (hexagonal)	16
2.5 Cuboctahedron	18
2.6 Cylinder	20
2.7 Dodecahedron	22
2.8 EllipsoidalCylinder	24
2.9 FullSphere	26
2.10 FullSpheroid	29
2.11 HemiEllipsoid	31
2.12 Icosahedron	33
2.13 Prism3 (triangular)	35
2.14 Prism6 (hexagonal)	37
2.15 Pyramid (square-based)	39
2.16 Tetrahedron	41
2.17 TruncatedCube	43
2.18 TruncatedSphere	45
2.19 TruncatedSpheroid	47
3 Ripples	49
3.1 Ripple1 (sinusoidal)	50
3.2 Ripple2 (saw-tooth)	52
References	54
Index	55

Preface

About BornAgain

BornAgain is a software package to simulate and fit reflectometry, off-specular scattering, and grazing-incidence small-angle scattering (GISAS) of X-rays and neutrons. It provides a generic framework for modeling multilayer samples with smooth or rough interfaces and with various types of embedded nanoparticles. The name, BornAgain, alludes to the central role of the distorted-wave Born approximation (DWBA) in the physical description of the scattering process.

BornAgain is maintained by the Scientific Computing Group of the Jülich Centre for Neutron Science (JCNS) at Heinz Maier-Leibnitz Zentrum (MLZ) Garching, Germany. It free and open source software. The source code is released under the GNU General Public License (GPL, version 3 or higher), the documentation under the Creative Commons license CC-BY-SA.

Citation

The canonical reference for BornAgain is the journal article

Gennady Pospelov, Walter Van Herck, Jan Burle, Juan M. Carmona Loaiza, Céline Durniak, Jonathan M. Fisher, Marina Ganeva, Dmitry Yurov and Joachim Wuttke:

BornAgain: software for simulating and fitting grazing-incidence small-angle scattering

[J. Appl. Cryst. 53, 262–276 \(2020\)](https://doi.org/10.1107/S002188981903007X)

Use of the software should additionally be documented by citing a specific version thereof

BornAgain — Software for simulating and fitting X-ray and neutron small-angle scattering at grazing incidence, version `<version>` (`<release date>`),
<http://www.bornagainproject.org>

Citation of the present document is only necessary when referring to specific information about form factors.

1 Introduction

BornAgain comes with a comprehensive collection of hard-coded shape transforms for standard particle geometries like spheres, cylinders, prisms, pyramids or ripples. This collection is documented in the following. For each shape, the real-space geometry is shown in orthogonal projections, the parameters of the BornAgain method are defined, an analytical expression for the form factor is given, and exemplary results for $|F(\mathbf{q})|^2$ versus α_f, ϕ_f are shown for small-angle scattering conditions ($\alpha_i = \phi_i = 0$).

The computation of $F(\mathbf{q})$ is based on shapes $S(\mathbf{r})$ given in Cartesian coordinates, as defined in the orthogonal projections. Typically, the vertical (z) direction is chosen along a symmetry axis of the particle. The origin is always at the center of the bottom side of the particle. Different parametrization or a different choice of the origin cause our analytic form factors to trivially deviate from expressions given in the `IsGISAXS` manual [1, Sec. 2.3] or in the literature [2, Appendix].

We made sure that all expressions also hold for complex scattering vectors \mathbf{q} , used to describe in order to take any material absorption into account. In standard reflectometry geometry, with reference surface normal to $\hat{\mathbf{z}}$, only the vertical components of \mathbf{k}_i and \mathbf{k}_f can have imaginary parts. However, for tilted particles $F(\tilde{\mathbf{q}})$ needs to be computed with a rotated scattering vector $\tilde{\mathbf{q}}$ that may be complex in all three components. Therefore BornAgain allows all three components of \mathbf{q} to be complex.

In the following, information about the implemented geometries is given in standardized form. Analytical expressions are given for the form factor $F(\mathbf{q})$, for the volume $V = F(0)$, and for the maximum horizontal section S (the area of the particle as seen from above). Mathematical notation in the form factor expressions includes the cardinal sine functions $\text{sinc}(z) := \sin(z)/z$ and the Bessel function of first kind and first order $J_1(z)$ [3, Ch. 9]. If results contain an integral, then no analytical form was found, and the integral is evaluated by numeric quadrature. For polyhedral figures, except a few simple ones like the rectangular box, we use a generic form factor computation, parametrized by the vertices of the figure, that is described in full detail in a mathematical paper [4].

Almost all analytical expressions for $F(\mathbf{q})$ contain removable singularities for certain values of \mathbf{q} . Our implementation uses proper analytic continuations at these singularities, though this is not explicitly denoted in the following formula collection. Furthermore, series expansions are used to ensure numeric accuracies in the neighborhood of the singularities. For polyhedra, see Ref. [4] for a meticulous discussion.

Geometrical objects can be parametrized in different ways. Concerns about user experience and about code readability sometimes lead to different choices. For the

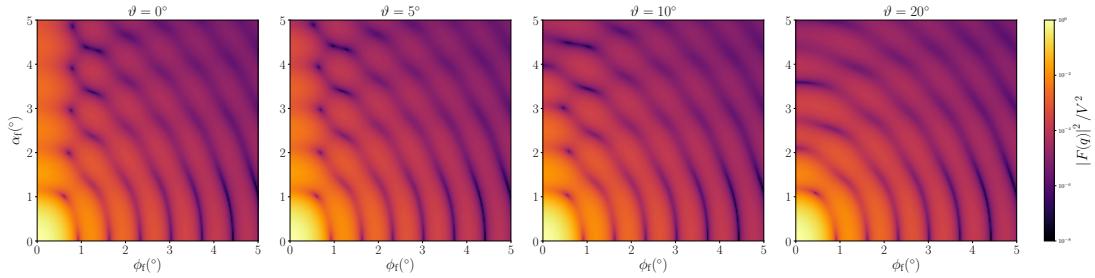


Figure 1.1: Normalized intensity $I(\alpha_f, \phi_f)$ for small-angle scattering by a truncated sphere with $R = 4.2$ nm and $H = 6.1$ nm, for four different tilt angles ϑ (rotation around the y axis). Since I possess the standard symmetry (1.2), data are only shown for first quadrant $0^\circ \leq \phi_f, \alpha_f \leq 5^\circ$.

BornAgain user interfaces (GUI and API) we have chosen the most standard parameters, as used in elementary geometry, like length, height, radius, even if this is at variance from the `IsGISAXS` precedent. Where our parametrization made analytic expressions too tedious, we use alternate internal parameters to alleviate the formulæ.

Exemplary form factors are numerically computed in Born approximation. All simulations scripts can be found in the BornAgain sources in directory [Doc/FFCatalog/fig/ff2](#). The particles are assigned a refractive index of $n = 10^{-5}$. Parameters are chosen such that the particle volume V is about 250 nm 3 (within $\pm 5\%$); except ripples, which are chosen with a vertical section V/L of 40 nm 2 and a length of 25 nm. The incident wavelength is 1 Å. The incident beam is always in x direction, hence $\alpha_i = \phi_i = 0$. Simulated detector images are normalized to the maximum scattering intensity at $F(0) = V$,

$$I(\alpha_f, \phi_f) := |F(q(\alpha_f, \phi_f))|^2 / V^2. \quad (1.1)$$

All plots have the same logarithmic color scale, extending over eight decades from 10^{-8} to 1 . Plot ranges in α_f and ϕ_f are also standardized as far as reasonably possible. For some geometries, the simulated detector image has some symmetry, namely horizontal or/and vertical mirror planes:

$$I(\alpha_f, \phi_f) = I(\alpha_f, -\phi_f) = I(-\alpha_f, -\phi_f) = I(\alpha_f, -\phi_f). \quad (1.2)$$

In these cases, we tend to restrict plots of I to the quadrant $\alpha_f \geq 0, \phi_f \geq 0$. However, it requires some experience to fully appreciate the information content of these plots. For a demonstration of this, try to grasp the main features of Fig. 1.1. Then compare with Fig. 1.2.

Finally, one warning: For large particles (typically of order 1000 nm), the form factor oscillates rapidly within one detector bin so that analytical calculations (performed for the bin center) may give a completely wrong intensity pattern. Several ways to work around this problem are proposed in Sect. 5.3 of our reference paper [5].

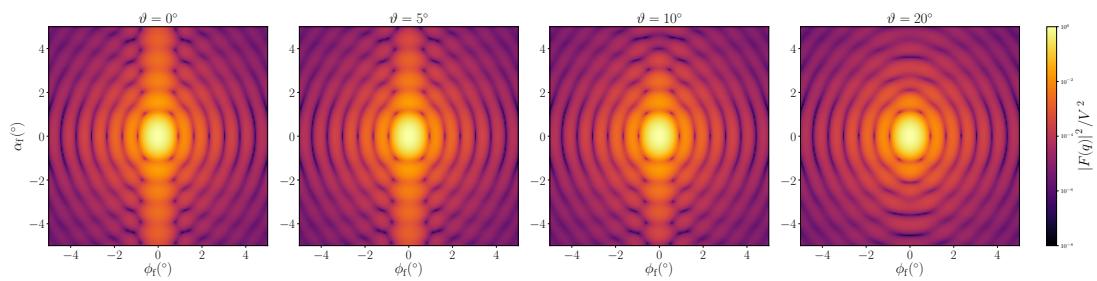


Figure 1.2: Same data as in Fig. 1.1, but now shown for all four quadrants ($-5^\circ \leq \phi_f, \alpha_f \leq 5^\circ$). The vertical interference pattern, which gradually disappears with increasing tilt angle, is much more salient in this plot than in the preceding one-quadrant representation.

2 Hard particles

The following tables summarize the implemented particle geometries, roughly ordered by decreasing symmetry. Afterwards, the detailed documentation is in alphabetical order.

Shape	Name	Symmetry	Parameters	Reference
	<code>FullSphere</code>	R_3	R	Page 26
	<code>FullSpheroid</code>	$D_{\infty h}$	R, H	Page 29
	<code>Cylinder</code>	$D_{\infty h}$	R, H	Page 20
	<code>TruncatedSphere</code>	$C_{\infty v}$	R, H	Page 45
	<code>TruncatedSpheroid</code>	$C_{\infty v}$	R, H, f_p	Page 47
	<code>Cone</code>	$C_{\infty v}$	R, H, α	Page 14
	<code>Icosahedron</code>	I_h	L	Page 33
	<code>Dodecahedron</code>	I_h	L	Page 22
	<code>TruncatedCube</code>	O_h	L, t	Page 43
	<code>Prism6</code>	D_{6h}	R, H	Page 37

	Cone6	C_{6v}	R, H, α	Page 16
	Pyramid	C_{4v}	L, H, α	Page 39
	Cuboctahedron	C_{4v}	L, H, r_H, α	Page 18
	Prism3	D_{3h}	L, H	Page 35
	Tetrahedron	C_{3v}	L, H, α	Page 41
	EllipsoidalCylinder	D_{2h}	R_a, R_b, H	Page 24
	Box	D_{2h}	L, W, H	Page 12
	HemiEllipsoid	C_{2v}	R_a, R_b, H	Page 31
	AnisoPyramid	C_{2v}	L, W, H, α	Page 10

2.1 AnisoPyramid (rectangle-based)

Real-space geometry

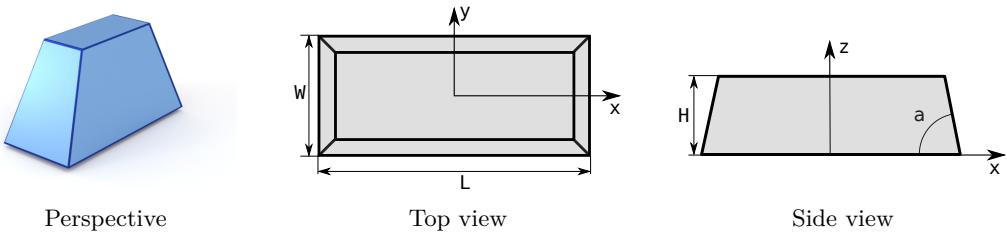


Figure 2.1: A truncated pyramid with a rectangular base.

Syntax and parameters

```
FormFactorAnisoPyramid(double length, double width, double height,
double alpha)
```

with the parameters

- `length` of the base, L ,
- `width` of the base, W ,
- `height`, H
- `alpha`, angle between the base and a side face, α .

They must fulfill

$$H \leq \frac{\tan \alpha}{2} \min(L, W).$$

Form factor, volume, horizontal section

F : computed using the generic polyhedron form factor [4],

$$V = H \left[LW - \frac{(L + W)H}{\tan \alpha} + \frac{4}{3} \frac{H^2}{\tan^2 \alpha} \right].$$

$$S = LW.$$

Examples

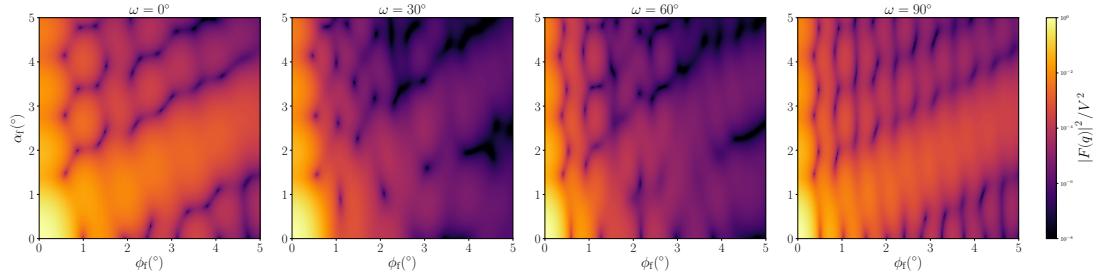


Figure 2.2: Normalized intensity $|F|^2/V^2$, computed with $L = 13 \text{ nm}$, $W = 8 \text{ nm}$, $H = 4.2 \text{ nm}$, and $\alpha = 60^\circ$, for four different angles ω of rotation around the z axis.

History

Agrees with the *In-plane anisotropic pyramid* form factor of `IsGISAXS` [1, Eq. 2.40] [2, Eq. 217], except for different parametrization. This is *not* the *anisotropic pyramid* of `FitGISAXS`, which is a true pyramid with an off-center apex [6].

Formfactors $F(\mathbf{q})$ have been checked against the different computation of `IsGISAXS`, and were found to fully agree.

2.2 Box (cuboid)

Real-space geometry

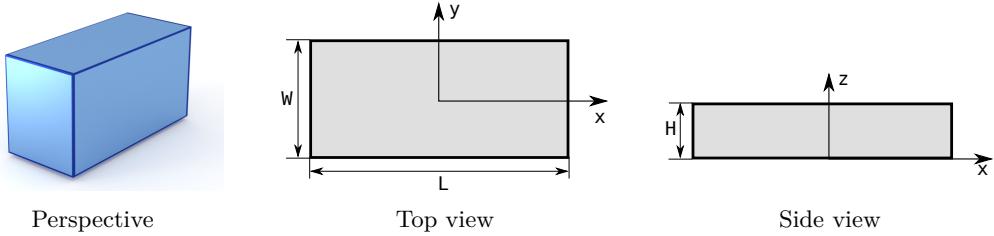


Figure 2.3: A rectangular cuboid.

Syntax and parameters

```
FormFactorBox(double length, double width, double height)
```

with the parameters

- **length** of the base, L ,
- **width** of the base, W ,
- **height**, H .

Form factor, volume, horizontal section

$$F = LWH \exp\left(iq_z \frac{H}{2}\right) \operatorname{sinc}\left(q_x \frac{L}{2}\right) \operatorname{sinc}\left(q_y \frac{W}{2}\right) \operatorname{sinc}\left(q_z \frac{H}{2}\right),$$

$$V = LWH,$$

$$S = LW.$$

Examples

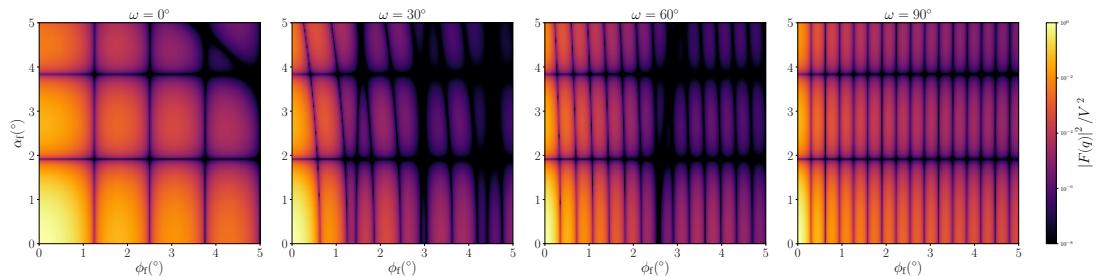


Figure 2.4: Normalized intensity $|F(q)|^2/V^2$, computed with $L = 18$ nm, $W = 4.6$ nm, and $H = 3$ nm, for four different angles ω of rotation around the z axis.

History

Agrees with Box form factor of `IsGISAXS` [1, Eq. 2.38] [2, Eq. 214], except for factors 1/2 in the definitions of parameters L , W , H .

2.3 Cone (circular)

Real-space geometry

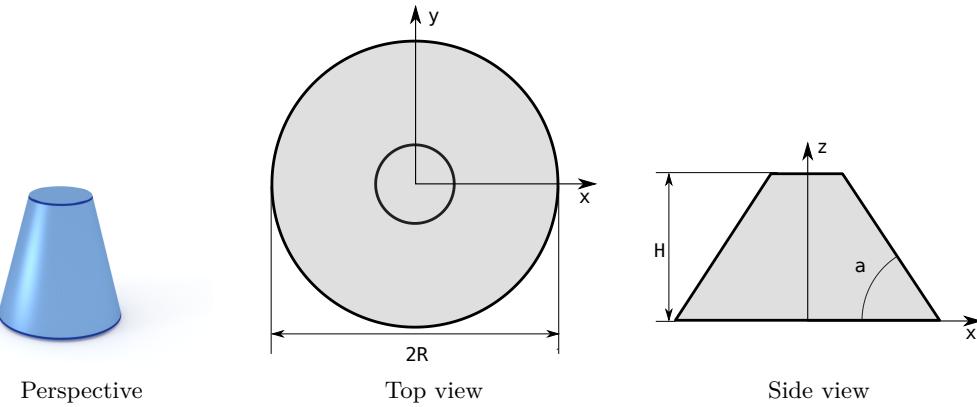


Figure 2.5: A truncated cone with circular base.

Syntax and parameters

```
FormFactorCone(double radius, double height, double alpha)
```

with the parameters

- **radius, R ,**
- **height, H ,**
- **alpha,** angle between the side and the base, α .

They must fulfill

$$H \leq R \tan \alpha.$$

Form factor, volume, horizontal section

Notation:

$$R_H := R - \frac{H}{\tan \alpha}, \quad q_{\parallel} := \sqrt{q_x^2 + q_y^2}, \quad \tilde{q}_z := q_z \tan \alpha.$$

Results:

$$F = 2\pi \tan \alpha e^{i\tilde{q}_z R} \int_{R_H}^R d\rho \rho^2 \frac{J_1(q_{\parallel}\rho)}{q_{\parallel}\rho} e^{-i\tilde{q}_z \rho},$$

$$V = \frac{\pi}{3} \tan \alpha (R^3 - R_H^3),$$

$$S = \pi R^2.$$

Examples

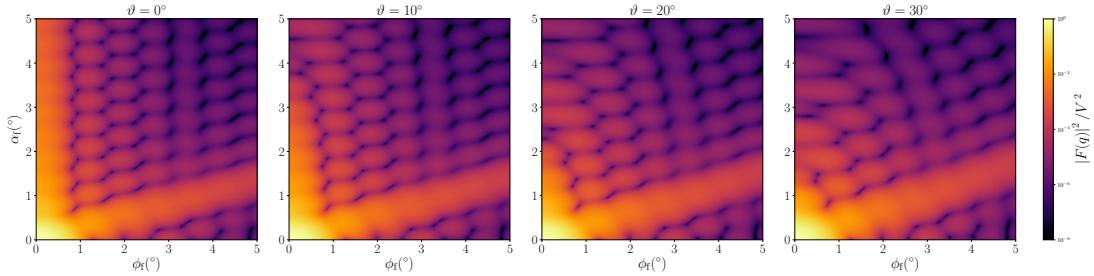


Figure 2.6: Normalized intensity $|F|^2/V^2$, computed with $R = 4 \text{ nm}$, $H = 11 \text{ nm}$, and $\alpha = 75^\circ$, for four different tilt angles ϑ (rotation around the y axis).

History and Derivation

Agrees with Cone form factor of IsGISAXS [1, Eq. 2.28] [2, Eq. 225], except for a substitution $z \rightarrow \rho$ in our expression for F . Justification for complex q in the same way as for the Cylinder form factor in Sec. 2.6.

2.4 Cone6 (hexagonal)

Real-space geometry

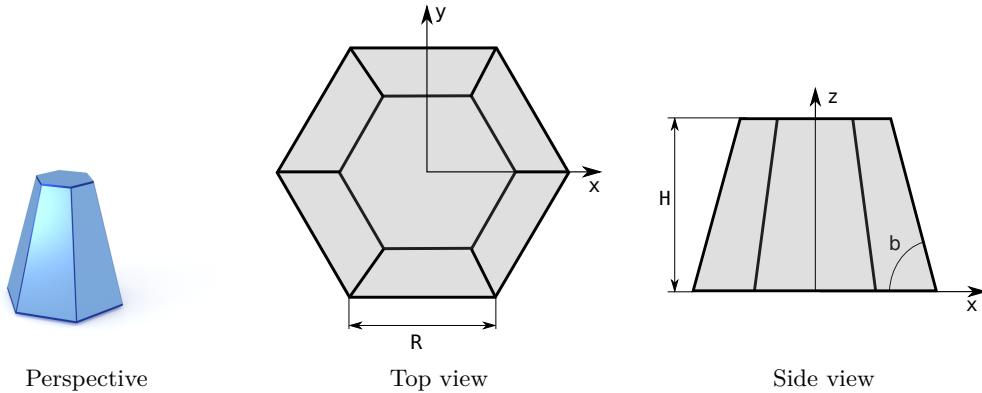


Figure 2.7: A truncated pyramid, based on a regular hexagon

Syntax and parameters

```
FormFactorCone6(double base_edge, double height, double alpha)
```

with the parameters

- `base_edge`, edge of the regular hexagonal base, R ,
- `height`, H ,
- `alpha`, dihedral angle between the base and a side face, α .

Note that the orthographic projection does not show α , but the angle β between the base and a side edge. They are related through $\sqrt{3} \tan \alpha = 2 \tan \beta$. The following is written more conveniently in terms of β . The parameters must fulfill

$$H \leq (\tan \beta)R.$$

Form factor, volume, horizontal section

F : computed using the generic polyhedron form factor [4],

$$V = \tan \beta \left(R^3 - \left(R - \frac{H}{\tan \beta} \right)^3 \right),$$

$$S = \frac{3\sqrt{3}R^2}{2}.$$

Examples

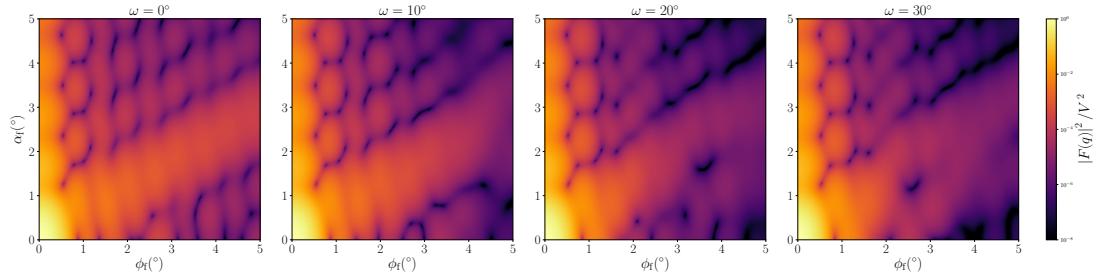


Figure 2.8: Normalized intensity $|F|^2/V^2$, computed with $R = 6$ nm, $H = 5$ nm, and $\alpha = 60^\circ$, for four different angles ω of rotation around the z axis.

History

Our parametrization deviates from the form factor *Cone6* of **IsGISAXS**[1, Eq. 2.32] [2, Eq. 222].

Up to BornAgain-1.5 computed by numeric integration, as in **IsGISAXS**. Since BornAgain-1.6 higher speed and better accuracy are achieved by using the generic polyhedron form factor [4], with series expansions near singularities.

2.5 Cuboctahedron

Real-space geometry



Figure 2.9: A compound of two truncated pyramids with a common square base and opposite orientations.

Syntax and parameters

```
FormFactorCuboctahedron(double length, double height, double
height_ratio, double alpha)
```

with the parameters

- **length** of the shared square base, L ,
- **height** of the bottom pyramid, H ,
- **height_ratio** between the top and the bottom pyramid, r_H ,
- **alpha**, angle between the base and a side face, α .

They must fulfill

$$H \leq \frac{\tan \alpha}{2} L \quad \text{and} \quad r_H H \leq \frac{\tan \alpha}{2} L.$$

Form factor, volume, horizontal section

F : computed using the generic polyhedron form factor [4],

$$V = \frac{1}{6} \tan(\alpha) L^3 \left[2 - \left(1 - \frac{2H}{L \tan(\alpha)} \right)^3 - \left(1 - \frac{2r_H H}{L \tan(\alpha)} \right)^3 \right],$$

$$S = L^2.$$

Examples

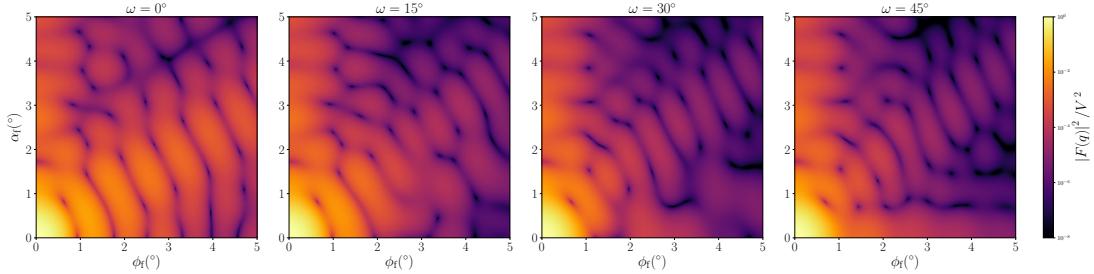


Figure 2.10: Normalized intensity $|F|^2/V^2$, computed with $L = 8$ nm, $H = 5$ nm, $r_H = 0.5$, and $\alpha = 60^\circ$, for four different angles ω of rotation around the z axis.

History

Agrees with Cuboctahedron form factor of **IsGISAXS** [1, Eq. 2.34] [2, Eq. 218], except for different parametrization $L = 2R_{\text{IsGISAXS}}$. Since BornAgain-1.6 implemented using the generic polyhedron form factor [4].

2.6 Cylinder

Real-space geometry

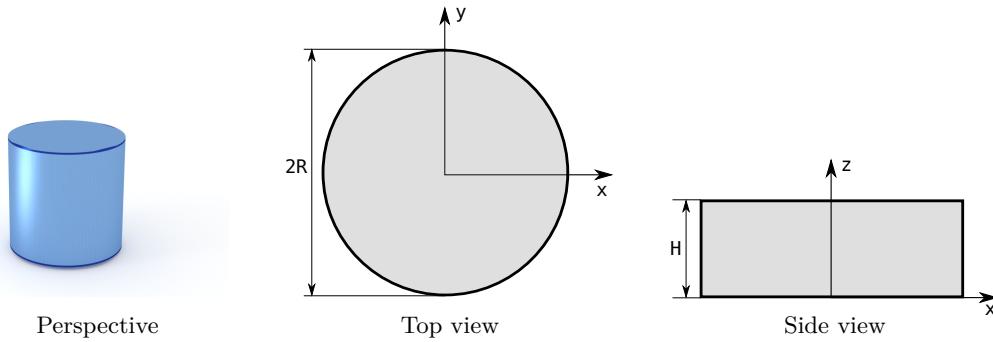


Figure 2.11: An upright circular cylinder.

Syntax and parameters

```
FormFactorCylinder(double radius, double height)
```

with the parameters

- **radius** of the circular base, R ,
- **height**, H .

Form factor, volume, horizontal section

Notation:

$$q_{\parallel} := \sqrt{q_x^2 + q_y^2}$$

Note that this does *not* involve the sesquilinear product $|q_x|^2 = q_x^* q_x$ but the plain product $q_x q_x$ of complex numbers (and analogous for q_y).

Results:

$$F = 2\pi R^2 H \operatorname{sinc}\left(q_z \frac{H}{2}\right) \exp\left(i q_z \frac{H}{2}\right) \frac{J_1(q_{\parallel} R)}{q_{\parallel} R},$$

$$V = \pi R^2 H,$$

$$S = \pi R^2.$$

Examples

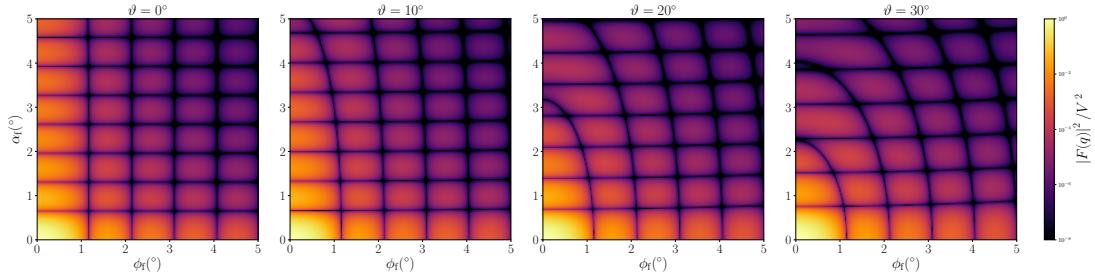


Figure 2.12: Normalized intensity $|F|^2/V^2$, computed with $R = 3$ nm and $H = 8.8$ nm, for four different tilt angles ϑ (rotation around the y axis).

History and Derivation

For real wavevectors, this form factor is well known; it goes back to Lord Rayleigh. In **IsGISAXS**, it has been implemented as form factor *Cylinder* [1, Eq. 2.27] [2, Eq. 223], allowing for complex wavevectors.

Since it is not obvious that the standard formula also holds for complex \mathbf{q} , let us provide a derivation. We only consider the integral over the polar angle,

$$I(\mathbf{q}) := \int_0^{2\pi} d\varphi \exp(iq_x r \sin \varphi + iq_y r \cos \varphi). \quad (2.1)$$

With the abbreviations $a := r(q_x + iq_y)/2$ and $b := r(q_x - iq_y)/2$,

$$I(\mathbf{q}) = \int_0^{2\pi} d\varphi \exp(a e^{i\varphi} - b e^{-i\varphi}). \quad (2.2)$$

Expansion of the exponential, combined with a binomial expansion of its argument, yields

$$I(\mathbf{q}) = \int_0^{2\pi} d\varphi \sum_{n=0}^{\infty} \sum_{k=0}^n (-)^k \frac{a^{n-k} b^k}{(n-k)! k!} e^{i(n-2k)\varphi}. \quad (2.3)$$

The integral over φ vanishes except for $n = 2k$. Hence

$$I(\mathbf{q}) = 2\pi \sum_{k=0}^{\infty} (-)^k \frac{\sqrt{ab}^{2k}}{k! k!} = 2\pi J_0(rq_{||}). \quad (2.4)$$

To compute the ensuing radial integral $\int dr r J_0(rq_{||})$, use $t J_0(t) = d[t J_1(t)]/dt$ [3, Formula 9.1.30a].

2.7 Dodecahedron

Real-space geometry



Figure 2.13: A regular dodecahedron.

Syntax and parameters

```
FormFactorDodecahedron(double edge)
```

with the parameter

- `edge`, length of one edge, a .

Form factor, volume, horizontal section

F : computed using the generic form factor of a polyhedron with inversion symmetry [4],

$$V = \frac{1}{4}(15 + 7\sqrt{5})a^3 \approx 7.663 a^3,$$

Examples

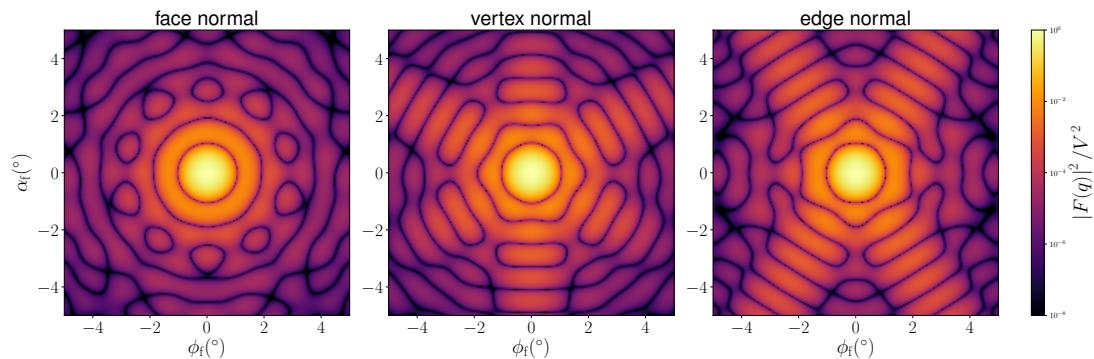


Figure 2.14: Normalized intensity $|F|^2/V^2$, computed with $a = 3.2$ nm, for three orientations of high symmetry: x axis perpendicular to a polygonal face; vertex on the x axis; edge in the xy plane and perpendicular to the x axis.

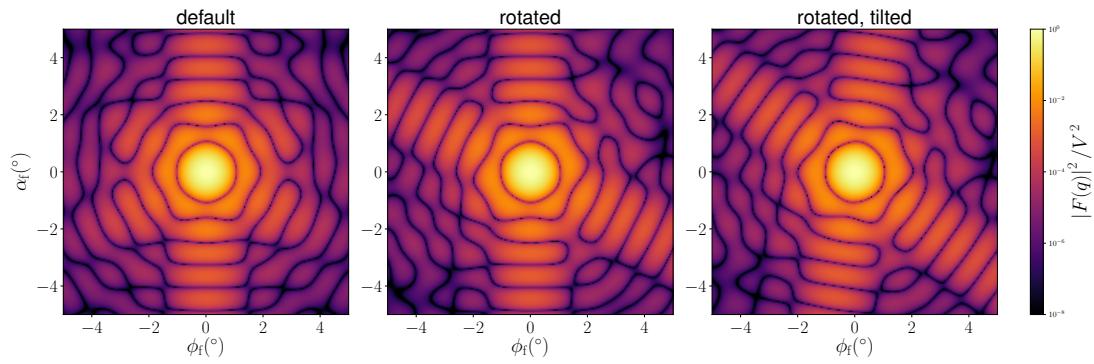


Figure 2.15: Normalized intensity $|F|^2/V^2$, computed with $a = 3.2$ nm, for three orientations of decreasing symmetry: base pentagon in xy plane and pointing in x direction; rotated by 13° around the z axis; ditto, and tilted by 9° around the x axis.

History

New in BornAgain-1.6, based on the generic form factor of the polyhedron [4].

2.8 EllipsoidalCylinder

Real-space geometry

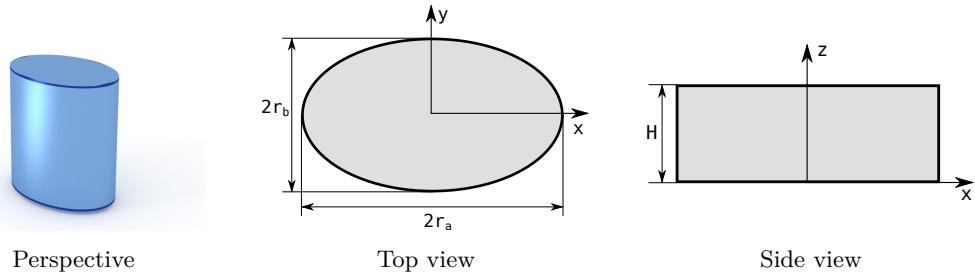


Figure 2.16: A upright cylinder whose cross section is an ellipse.

Syntax and parameters

```
FormFactorEllipsoidalCylinder(double radius_a, double radius_b,
    double height)
```

with the parameters

- `radius_a`, in x direction, R_a ,
- `radius_b`, in y direction, R_b ,
- `height`, H .

Form factor, volume, horizontal section

Notation:

$$\gamma := \sqrt{(q_x R_a)^2 + (q_y R_b)^2}$$

Results:

$$F = 2\pi R_a R_b H \exp\left(i \frac{q_z H}{2}\right) \operatorname{sinc}\left(\frac{q_z H}{2}\right) \frac{J_1(\gamma)}{\gamma},$$

$$V = \pi R_a R_b H,$$

$$S = R_a R_b.$$

Examples

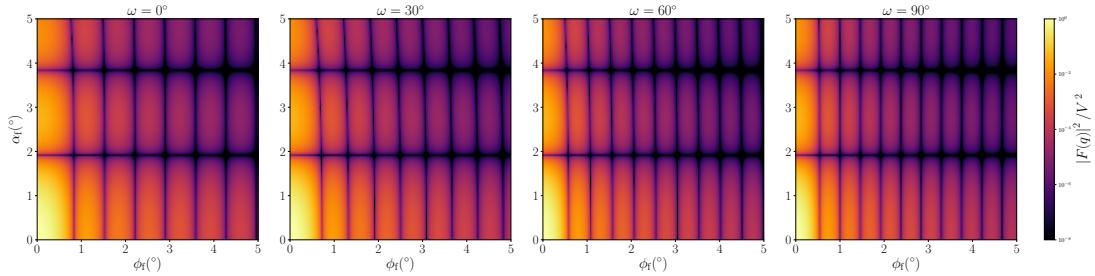


Figure 2.17: Normalized intensity $|F|^2/V^2$, computed with $R_a = 6.3$ nm, $R_b = 4.2$ nm and $H = 3$ nm, for four different angles ω of rotation around the z axis.

History

Agrees with the **IsGISAXS** form factor *Ellipsoid* [1, Eq. 2.41, wrongly labeled in Fig. 2.4] or *Ellipsoidal Cylinder* [2, Eq. 224].

2.9 FullSphere

Real-space geometry

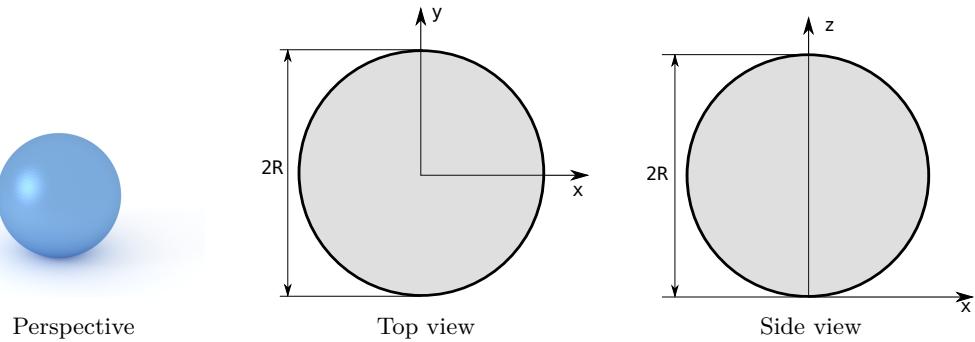


Figure 2.18: A full sphere.

Syntax and parameters

```
FormFactorFullSphere(double radius)
```

with the parameter

- **radius, R .**

Form factor, volume, horizontal section

Notation:

$$q := \sqrt{q_x^2 + q_y^2 + q_z^2}.$$

Note that this does *not* involve the sesquilinear product $|q_x|^2 = q_x^* q_x$ but the plain product $q_x q_x$ of complex numbers (and analogous for q_y, q_z).

$$F = \frac{4\pi}{q^3} \exp(iq_z R) [\sin(qR) - qR \cos(qR)],$$

$$V = \frac{4\pi}{3} R^3,$$

$$S = \pi R^2.$$

Example

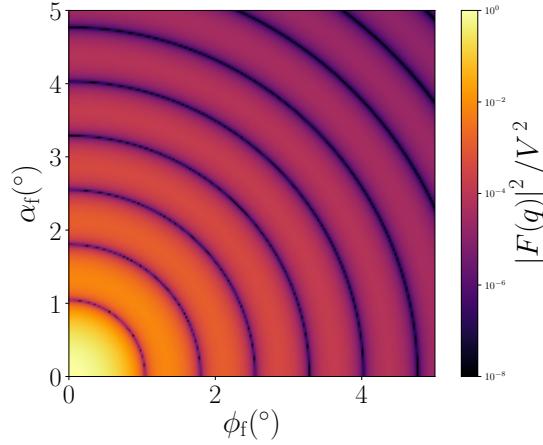


Figure 2.19: Normalized intensity $|F|^2/V^2$, computed with $R = 3.9$ nm.

History and Derivation

For real wavevectors, this form factor is well known; it goes back at least to Lord Rayleigh. In `IsGISAXS`, it has been implemented as form factor *Full sphere* [1, Eq. 2.36] [2, Eq. 226], allowing for complex wavevectors. Since it is not obvious that Rayleigh's formula also holds for complex \mathbf{q} , let us outline a derivation (if you know a more elegant one, we would like to hear).

If the origin is at the center of the sphere, then the form factor is

$$I(\mathbf{q}, R) = \int_0^R dr r^2 \int_0^\pi d\theta \sin \theta \int_0^{2\pi} d\varphi e^{i\mathbf{qr}} \quad (2.5)$$

with $\mathbf{qr} = q_x r \sin \theta \cos \varphi + q_y r \sin \theta \sin \varphi + q_z r \cos \theta$. For the integration over φ , see Sec. 2.6 on the form factor of a cylinder:

$$I(\mathbf{q}, R) = 2\pi \int_0^R dr r^2 \int_0^\pi d\theta \sin \theta \exp(iq_z \cos \theta) J_0(q_{\parallel} r \sin \theta) \quad (2.6)$$

with $q_{\parallel} = \sqrt{q_x^2 + q_y^2}$. By symmetry, the imaginary part is zero, so that the exponential reduces to a cosine:

$$I(\mathbf{q}, R) = 2\pi \int_0^R dr r^2 \int_0^\pi d\theta \sin \theta \cos(q_z \cos \theta) J_0(q_{\parallel} r \sin \theta). \quad (2.7)$$

Expand the outer cosine and the Bessel function:

$$I(\mathbf{q}, R) = 2\pi \int_0^R dr r^2 \int_0^\pi d\theta \sin \theta \sum_{j=0}^{\infty} (-)^j \frac{(q_z r \cos \theta)^{2j}}{(2j)!} \sum_{k=0}^{\infty} (-)^k \frac{(q_{\parallel} r \sin \theta)^{2k}}{4^k k!^2}. \quad (2.8)$$

Sort by powers of r , and integrate:

$$I(\mathbf{q}, R) = 2\pi \sum_{n=0}^{\infty} (-)^n \frac{R^{2n+3}}{2n+3} \sum_{k=0}^n \frac{q_z^{2n-2k}}{(2n-2k)!} \frac{q_{\parallel}^{2k}}{4^k k!^2} \zeta(k, n) \quad (2.9)$$

with

$$\zeta(k, n) := \int_0^\pi d\theta \sin \theta (\cos \theta)^{2n-2k} (\sin \theta)^{2k}. \quad (2.10)$$

This integral [7, no. 2.512.4] yields

$$\zeta(k, n) = \frac{2^{2k+1} (2n-2k)! n! k!}{(2n+1)! (n-k)!}. \quad (2.11)$$

Hence

$$I(\mathbf{q}, R) = 4\pi \sum_{n=0}^{\infty} (-)^n \frac{R^{2n+3}}{(2n+3)(2n+1)!} \sum_{k=0}^n \frac{n!}{(n-k)! k!} q_z^{2n-2k} q_{\parallel}^{2k}. \quad (2.12)$$

The inner sum happens to be the binomial expansion of $q^{2n} = (q_z^2 + q_{\parallel}^2)^n$. Therefore (2.12) coincides with the series expansion of

$$I(\mathbf{q}, R) = 4\pi q^{-3} (\sin(qR) - qR \cos(qR)), \quad (2.13)$$

which is what we wanted to prove.

2.10 FullSpheroid

Real-space geometry

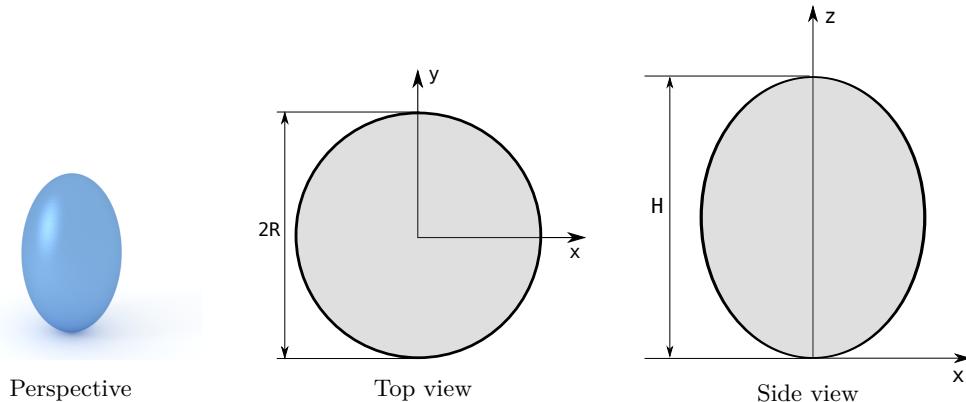


Figure 2.20: A full spheroid, generated by rotating an ellipse around the vertical axis.

Syntax and parameters

```
FormFactorFullSpheroid(double radius, double height)
```

with the parameters

- **radius**, R ,
- **height**, H .

Form factor, volume, horizontal section

Notation:

$$h := H/2, \quad s := \sqrt{(Rq_x)^2 + (Rq_y)^2 + (hq_z)^2}.$$

Results:

$$F = 4\pi \exp(iq_z h) R^2 h \frac{\sin(s) - s \cos(s)}{s^3},$$

$$V = \frac{4\pi}{3} R^2 h,$$

$$S = \pi R^2.$$

Example

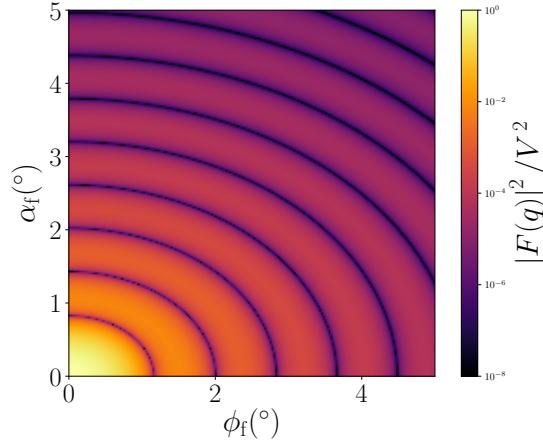


Figure 2.21: Normalized intensity $|F|^2/V^2$, computed with $R = 3.5$ nm and $H = 9.8$ nm.

History and Derivation

Replicates the *Full spheroid* of `IsGISAXS` [1, Eq. 2.37] [2, Eq. 227], except for wrong factors of 2 in their volume formula and form factor implementation. Up to BornAgain 1.16, our form factor computation followed `IsGISAXS` in using numeric integration in the z coordinate.

Thanks to Matt Thompson (Australian National University) who pointed out that the form factor of any spheroid can be reduced to that of the regular sphere (Sec. 2.9) by rescaling $\mathbf{r}\mathbf{q} = (M\mathbf{r})(M^{-1}\mathbf{q})$. In the present case (with revolution axis along z), the transformation matrix is just $M = \text{diag}(1/R, 1/R, 1/h)$. The resulting simple expression for the form factor goes back at least to Guinier [8, p. 193].

2.11 HemiEllipsoid

Real-space geometry

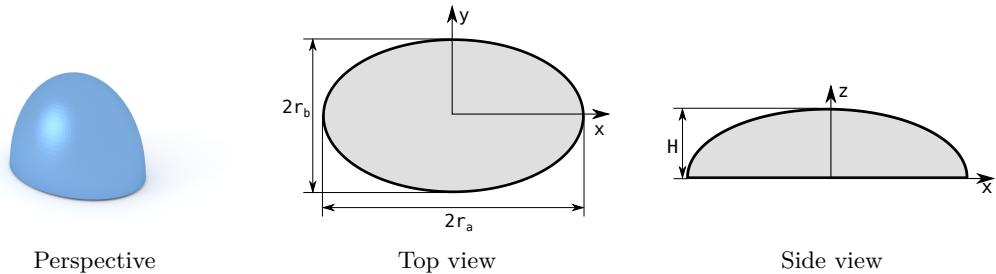


Figure 2.22: An horizontally oriented ellipsoid, truncated at the central plane.

Syntax and parameters

```
FormFactorHemiEllipsoid(double radius_a, double radius_b, double
height)
```

with the parameters

- `radius_a`, in x direction, R_a ,
- `radius_b`, in y direction, R_b ,
- `height`, equal to radius in z direction, H

Form factor, volume, horizontal section

Notation:

$$r_{a,z} := R_a \sqrt{1 - \left(\frac{z}{H}\right)^2}, \quad r_{b,z} := R_b \sqrt{1 - \left(\frac{z}{H}\right)^2}, \quad \gamma_z = \sqrt{(q_x r_{a,z})^2 + (q_y r_{b,z})^2}.$$

Results:

$$F = 2\pi \int_0^H dz r_{a,z} r_{b,z} \frac{J_1(\gamma_z)}{\gamma_z} \exp(i q_z z),$$

$$V = \frac{2}{3} \pi R_a R_b H,$$

$$S = \pi R_a R_b.$$

Examples

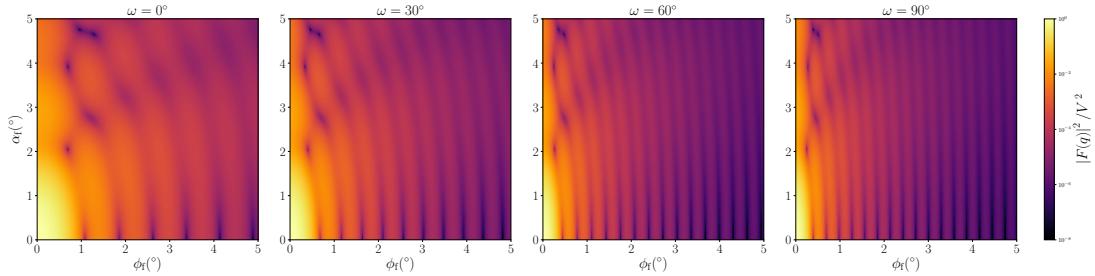


Figure 2.23: Normalized intensity $|F|^2/V^2$, computed with $R_a = 10$ nm, $R_b = 3.8$ nm and $H = 3.2$ nm, for four different angles ω of rotation around the z axis.

History

Agrees with the `IsGISAXS` form factor *Anisotropic hemi-ellipsoid* [1, Eq. 2.42, with wrong sign in the z -dependent phase factor] or *Hemi-spheroid* [2, Eq. 229].

2.12 Icosahedron

Real-space geometry



Figure 2.24: A regular icosahedron.

Syntax and parameters

```
FormFactorIcosahedron(double edge)
```

with the parameter

- `edge`, length of one edge, a .

Form factor, volume, horizontal section

F : computed using the generic form factor of a polyhedron with inversion symmetry [4],

$$V = \frac{5}{12}(3 + \sqrt{5})a^3 \approx 2.182 a^3$$

Examples

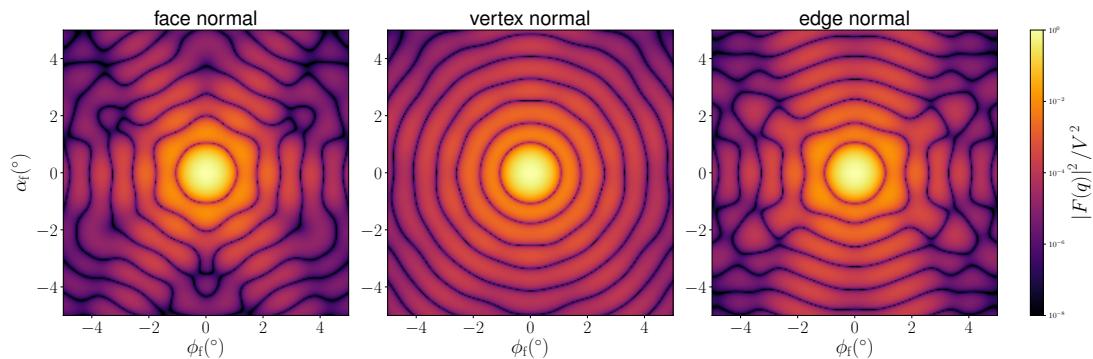


Figure 2.25: Normalized intensity $|F|^2/V^2$, computed with $a = 4.8$ nm, for three orientations of high symmetry: x axis perpendicular to a polygonal face; vertex on the x axis; edge in the xy plane and perpendicular to the x axis.

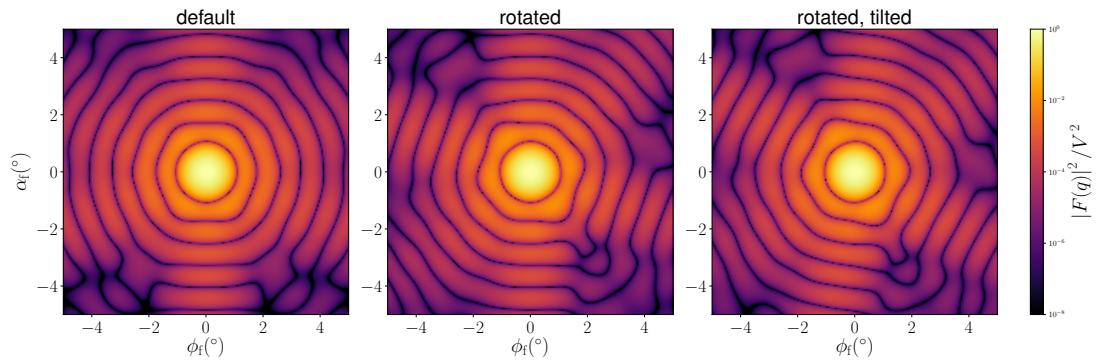


Figure 2.26: Normalized intensity $|F|^2/V^2$, computed with $a = 4.8$ nm, for three orientations of decreasing symmetry: base pentagon in xy plane and pointing in x direction; rotated by 13° around the z axis; ditto, and tilted by 9° around the x axis.

History

New in BornAgain-1.6, based on the generic form factor of the polyhedron [4].

2.13 Prism3 (triangular)

Real-space geometry

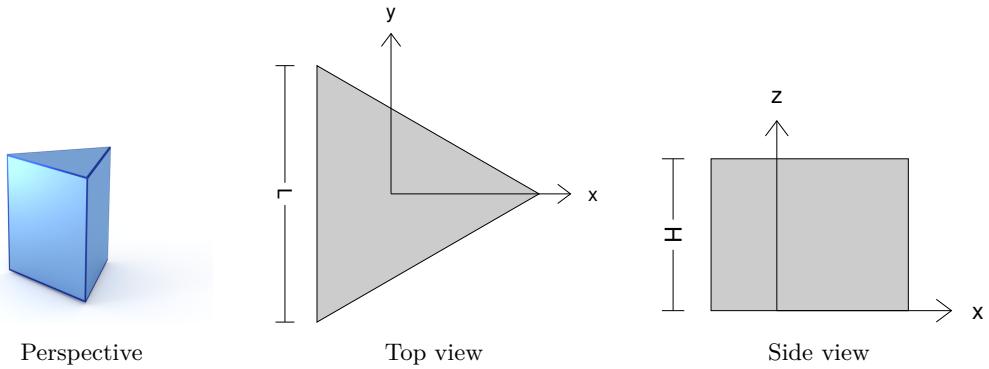


Figure 2.27: A prism based on an equilateral triangle.

Syntax and parameters

```
FormFactorPrism3(double length, double height)
```

with the parameters

- `length` of one base edge, L ,
- `height`, H .

Form factor, volume, horizontal section

$$F = H \operatorname{sinc} \left(q_z \frac{H}{2} \right) \exp \left(-iq_z \frac{H}{2} \right) F_{\parallel}(\mathbf{q}_{\parallel})$$

with the form factor F_{\parallel} of the base triangle computed using the generic form factor of a planar polygon [4],

$$V = \frac{\sqrt{3}}{4} H L^2,$$

$$S = \frac{\sqrt{3}}{4} L^2.$$

Examples

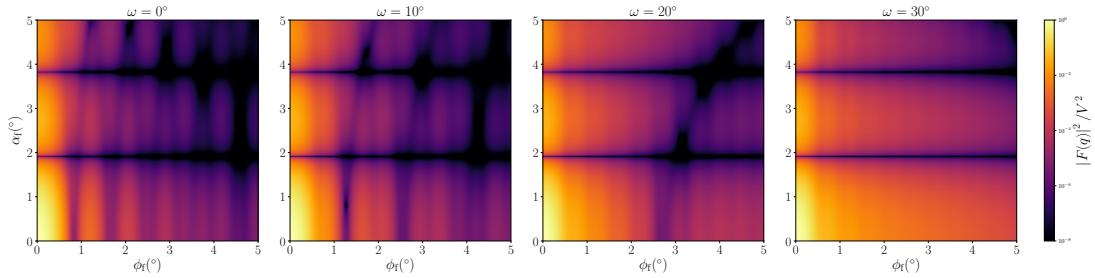


Figure 2.28: Normalized intensity $|F|^2/V^2$, computed with $L = 13.8$ nm and $H = 3$ nm, for four different angles ω of rotation around the z axis.

History

Has been validated against the *Prism3* form factor of `IsGISAXS` [1, Eq. 2.29] [2, Eq. 219]. Note the different parameterization $L = 2R_{\text{IsGISAXS}}$. In `FitGISAXS` just called *Prism* [6]. In `BornAgain-1.6`, redefined to let the x axis point along a symmetry axis (rotated by 30° with respect to the previous version).

Reimplemented in `BornAgain-1.6` using the generic form factor of a polygonal prism [4], to achieve numerical stability near the removable singularity at $q \rightarrow 0$.

2.14 Prism6 (hexagonal)

Real-space geometry

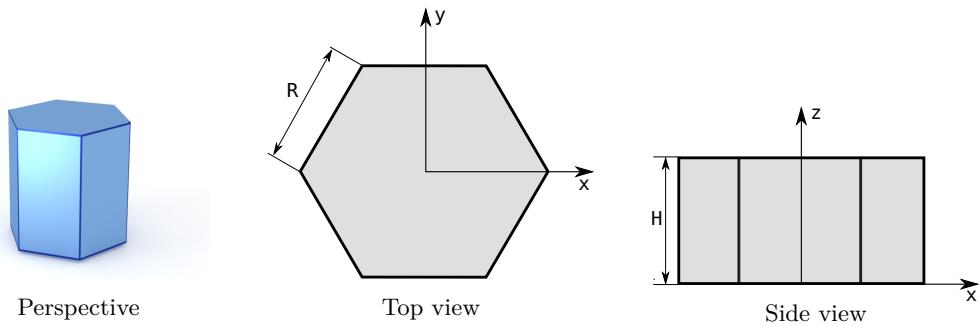


Figure 2.29: A prism based on a regular hexagon.

Syntax and parameters

```
FormFactorPrism6(double radius, double height)
```

with the parameters

- **radius** of the hexagonal base, R ,
- **height**, H .

Form factor, volume, horizontal section

$$F = H \operatorname{sinc} \left(q_z \frac{H}{2} \right) \exp \left(-iq_z \frac{H}{2} \right) F_{\parallel}(\mathbf{q}_{\parallel})$$

with the form factor F_{\parallel} of the base hexagon computed using the generic form factor of a planar polygon with two-fold symmetry (S_2) [4],

$$V = \frac{3\sqrt{3}}{2} H R^2,$$

$$S = \frac{3\sqrt{3}R^2}{2}.$$

Examples

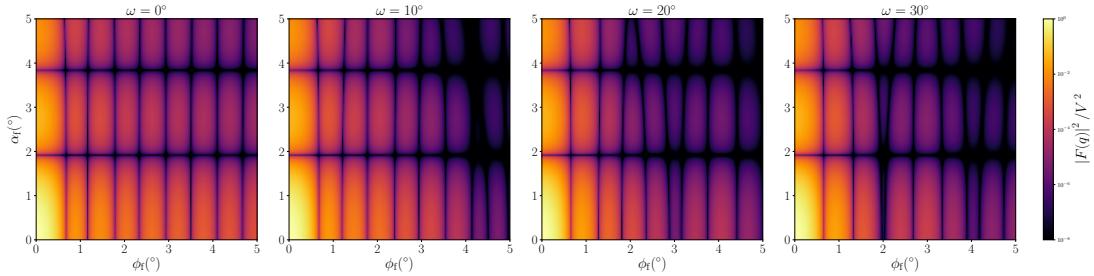


Figure 2.30: Normalized intensity $|F|^2/V^2$, computed with $R = 5.7$ nm and $H = 3$ nm, for four different angles ω of rotation around the z axis.

History

Has been validated against the *Prism6* form factor of *IsGISAXS* [1, Eq. 2.31] [2, Eq. 221], which has different parametrization and lacks a factor H in $F(\mathbf{q})$.

Reimplemented in BornAgain-1.5 using the generic form factor of a polygonal prism with symmetry S_2 [4], to achieve numerical stability near the removable singularity at $q \rightarrow 0$.

2.15 Pyramid (square-based)

Real-space geometry

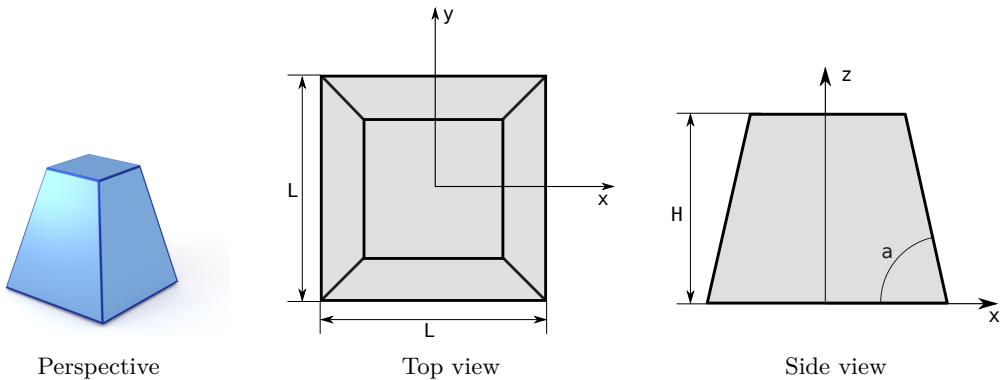


Figure 2.31: A truncated pyramid with a square base.

Syntax and parameters

```
FormFactorPyramid(double length, double height, double alpha)
```

with the parameters

- **length** of one edge of the square base, L ,
- **height**, H ,
- **alpha**, angle between the base and a side face, α ,

They must fulfill

$$H \leq \frac{\tan \alpha}{2} L.$$

Form factor, volume, horizontal section

F : computed using the generic polyhedron form factor [4],

$$V = \frac{1}{6} L^3 \tan \alpha \left[1 - \left(1 - \frac{2H}{L \tan \alpha} \right)^3 \right],$$

$$S = L^2.$$

Examples

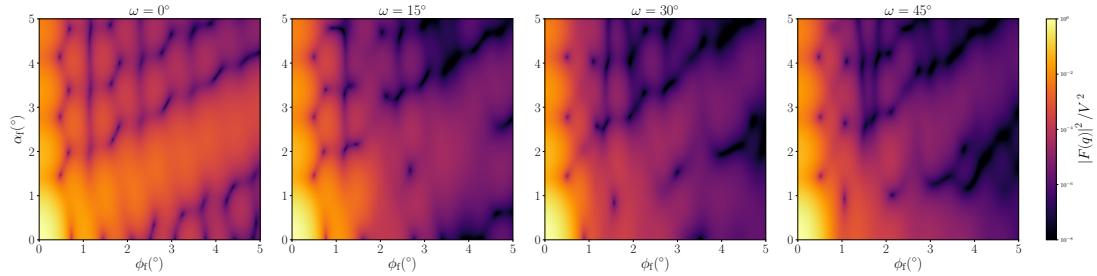


Figure 2.32: Normalized intensity $|F|^2/V^2$, computed with $L = 10$ nm, $H = 4.2$ nm and $\alpha = 60^\circ$, for four different angles ω of rotation around the z axis.

History

Corresponds to *Pyramid* form factor of **IsGISAXS** [1, Eq. 2.31] [2, Eq. 221], except for different parametrization $L = 2R_{\text{IsGISAXS}}$ and a corrected sign.

Reimplemented in BornAgain-1.6 using the generic form factor of a polygonal prism [4], to achieve numerical stability near the removable singularity at $q \rightarrow 0$.

2.16 Tetrahedron

Real-space geometry

Incorrectly named so, since it actually has five, not four surfaces.

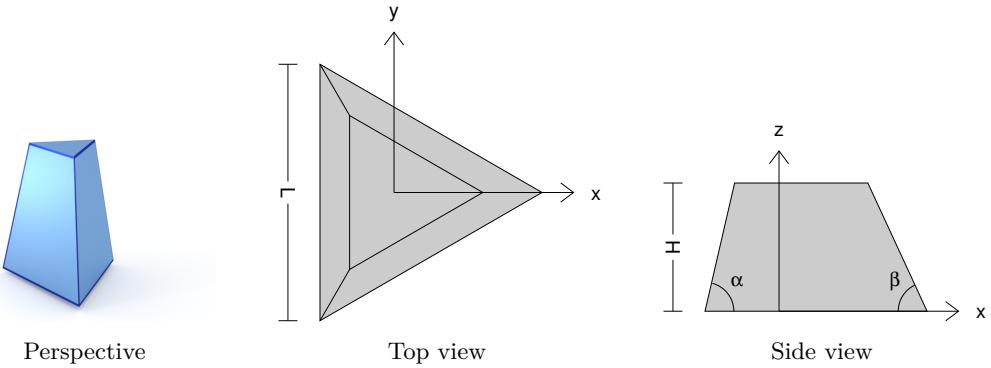


Figure 2.33: A truncated pyramid, based on an equilateral triangle.

Syntax and parameters

```
FormFactorTetrahedron(double length, double height, double alpha)
```

with the parameters

- `length` of one edge of the equilateral triangular base, L ,
- `height`, H ,
- `alpha`, dihedral angle between the base and a side face, α .

They must fulfill

$$H \leq \frac{\tan \alpha}{2\sqrt{3}} L.$$

The orthographic projection also shows the angle β between the base and a side edge. It is related to the dihedral angle through $\tan \alpha = 2 \tan \beta$.

Form factor, volume, horizontal section

F : computed using the generic polyhedron form factor [4],

$$V = \frac{\tan(\alpha)L^3}{24} \left[1 - \left(1 - \frac{2\sqrt{3}H}{L \tan(\alpha)} \right)^3 \right],$$

$$S = \frac{\sqrt{3}}{4} L^2.$$

Examples

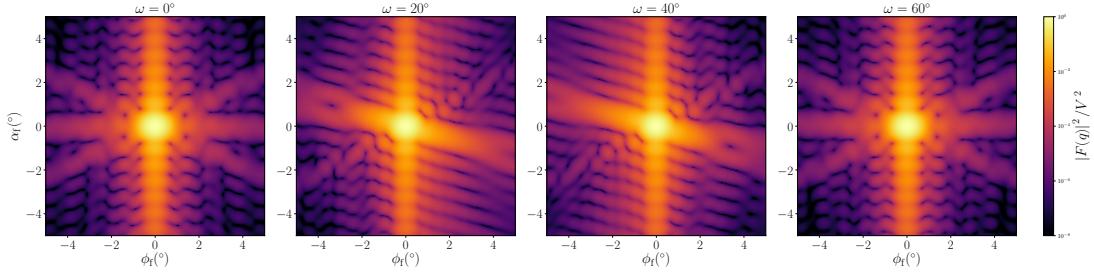


Figure 2.34: Normalized intensity $|F|^2/V^2$, computed with $L = 12$ nm, $H = 8$ nm, and $\alpha = 75^\circ$, for four different angles ω of rotation around the z axis. The low symmetry requires other angular ranges than used in most other figures.

History

Previous implementations as *Tetrahedron* in **IsGISAXS** [1, Eq. 2.30] [2, Eq. 220], and as *Truncated tetrahedron* in **FitGISAXS** [6]. In BornAgain-1.6, redefined to let the x axis lie in a mirror plane (rotated by 30° with respect to the previous version).

Up to BornAgain-1.5, we computed the form factor by numeric integration, as in **IsGISAXS**. Since BornAgain-1.6 higher speed and accuracy are achieved by using the generic polyhedron form factor [4], with series expansions near singularities.

2.17 TruncatedCube

Real-space geometry

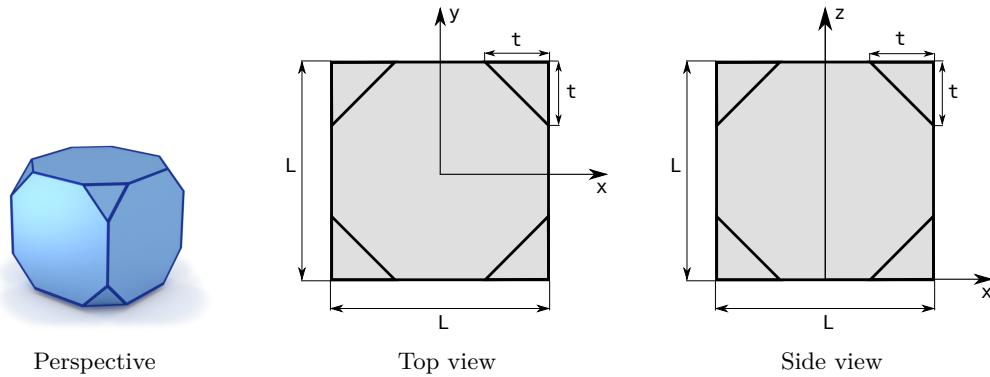


Figure 2.35: A cube whose eight vertices have been removed. The truncated part of each vertex is a trirectangular tetrahedron.

Syntax and parameters

```
FormFactorTruncatedCube(double length, double removed_length)
```

with the parameters

- `length` of the full cube, L ,
- `removed_length`, side length of the trirectangular tetrahedron removed from the cube's vertices, t .

They must fulfill

$$t \leq L/2.$$

Form factor, volume, horizontal section

F : computed using the generic form factor of a polyhedron with inversion symmetry [4],

$$V = L^3 - \frac{4}{3}t^3,$$

$$S = L^2.$$

Examples

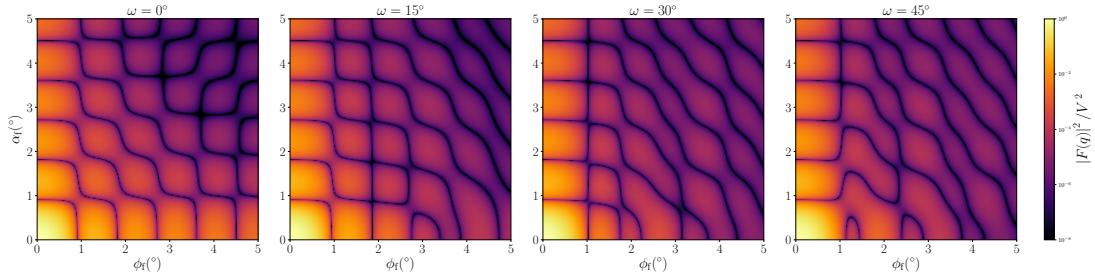


Figure 2.36: Normalized intensity $|F|^2/V^2$, computed with $L = 25 \text{ nm}$, $W = 10 \text{ nm}$, $H = 8 \text{ nm}$, and $d = 5 \text{ nm}$, for four different angles ω of rotation around the z axis.

History

Motivated by [9]. Reimplemented in BornAgain-1.6 using the generic form factor of a polygonal prism [4].

2.18 TruncatedSphere

A *spherical segment*, obtained from a spherical ball by two parallel cuts.

Real-space geometry

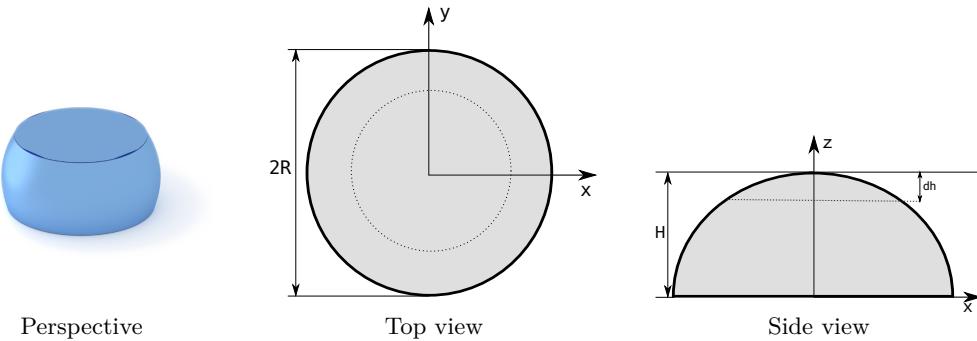


Figure 2.37: A truncated sphere.

Syntax and parameters

```
FormFactorTruncatedSphere(double radius, double height, double dh)
```

with the parameters

- `radius`, R ,
- `height`, H ,
- `top removal`, dh .

They must fulfill

$$0 < H \leq 2R,$$

$$dh < H.$$

Special cases

A *spherical cap* is obtained from a spherical ball by a single cut. This is covered by the following special parameterization of `TruncatedSphere`:

- Single cut at the bottom: $dh = 0$.
- Single cut at the top: $H = 2R$.

Form factor, volume, horizontal section

Notation:

$$q_{\parallel} := \sqrt{q_x^2 + q_y^2}, \quad R_z := \sqrt{R^2 - z^2}.$$

Results:

$$F = 2\pi \exp[iq_z(H - R)] \int_{R-H}^{R-dh} dz R_z^2 \frac{J_1(q_{\parallel} R_z)}{q_{\parallel} R_z} \exp(iq_z z) dz,$$

$$V = \frac{\pi}{3} [3R(H^2 - dh^2) + dh^3 - H^3],$$

$$S = \begin{cases} \pi(2RH - H^2), & H < R \\ \pi R^2, & H \geq R, dh < R \\ \pi(2Rdh - dh^2), & H \geq R, dh \geq R \end{cases}.$$

Example

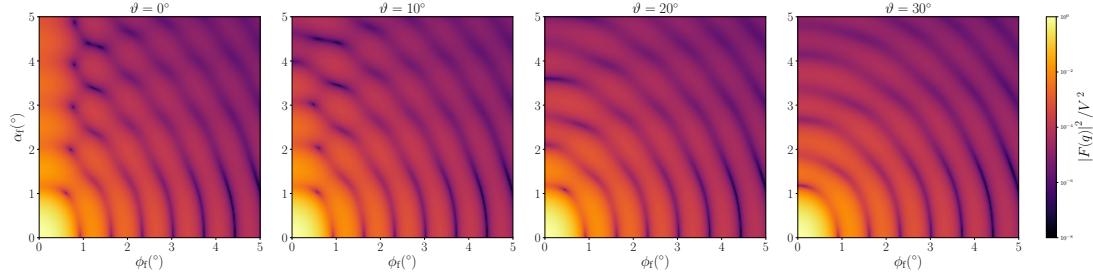


Figure 2.38: Normalized intensity $|F|^2/V^2$, computed with $R = 4.2$ nm and $H = 6.1$ nm, for four different tilt angles ϑ (rotation around the y axis).

History and Derivation

Agrees with the *IsGISAXS* form factor *Sphere* [1, Eq. 2.33] or *Truncated sphere* [2, Eq. 228]. Justification for complex q in the same way as for the *Cylinder* form factor in Sec. 2.6.

2.19 TruncatedSpheroid

Real-space geometry

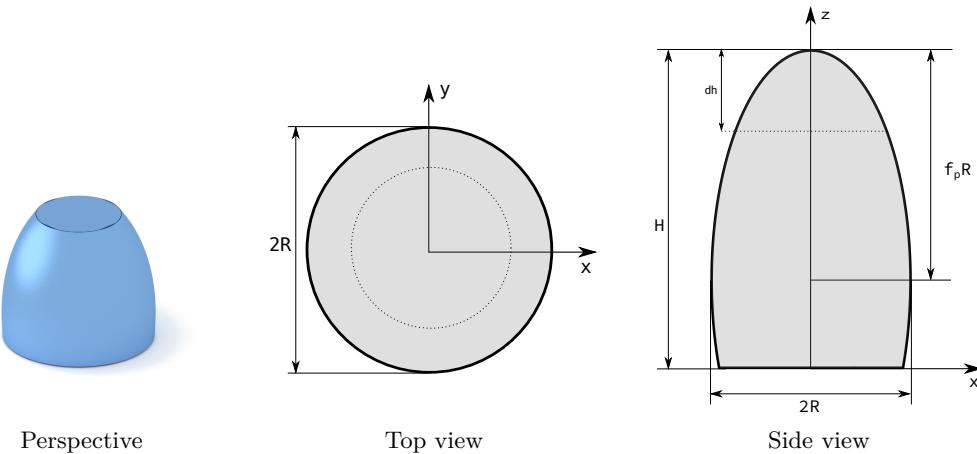


Figure 2.39: A vertically oriented, horizontally truncated spheroid.

Syntax and parameters

```
FormFactorTruncatedSpheroid(double radius, double height, double
    height_flattening, double dh)
```

with the parameters

- `radius, R,`
- `height, H,`
- `height_flattening, fp,`
- `top removal, dh.`

They must fulfill

$$0 < \frac{H}{R} \leq 2f_p.$$

$$dh < H.$$

Form factor, volume, horizontal section

Notation:

$$q_{\parallel} := \sqrt{q_x^2 + q_y^2}, \quad R_z := \sqrt{R^2 - z^2/f_p^2}.$$

Results:

$$F = 2\pi \exp[iq_z(H - f_p R)] \int_{f_p R - H}^{f_p R - dh} dz R_z^2 \frac{J_1(q_{\parallel} R_z)}{q_{\parallel} R_z} \exp(iq_z z)$$

$$V = \frac{\pi}{3f_p^2} [3Rf_p(H^2 - dh^2) + dh^3 - H^3],$$

$$S = \begin{cases} \pi(2Rf_pH - H^2)/f_p^2, & H < Rf_p \\ \pi R^2, & H \geq Rf_p, dh < Rf_p \\ \pi(2Rf_pdh - dh^2)/f_p^2, & H \geq Rf_p, dh \geq Rf_p \end{cases}.$$

Example

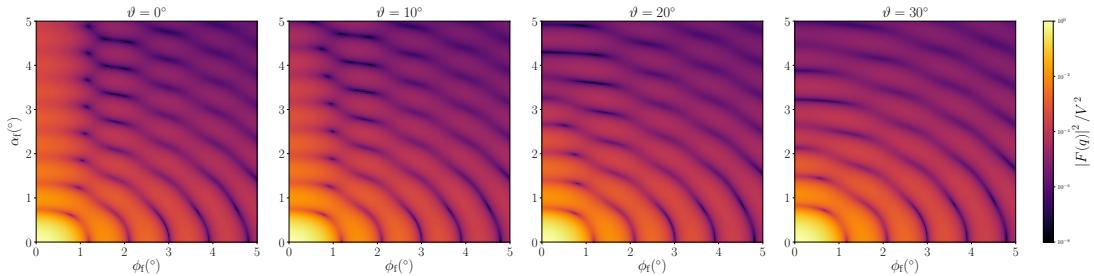


Figure 2.40: Normalized intensity $|F|^2/V^2$, computed with $R = 3.3$ nm, $H = 9.8$ nm, and $f_p = 1.8$, for four different tilt angles ϑ (rotation around the y axis).

History and Derivation

Agrees with the `IsGISAXS` form factor *Sphere* [1, Eq. 2.33] or *TruncatedSpheroid* [2, Eq. 228]. Justification for complex q in the same way as for the *Cylinder* form factor in Sec. 2.6.

3 Ripples

3.1 Ripple1 (sinusoidal)

Real-space geometry

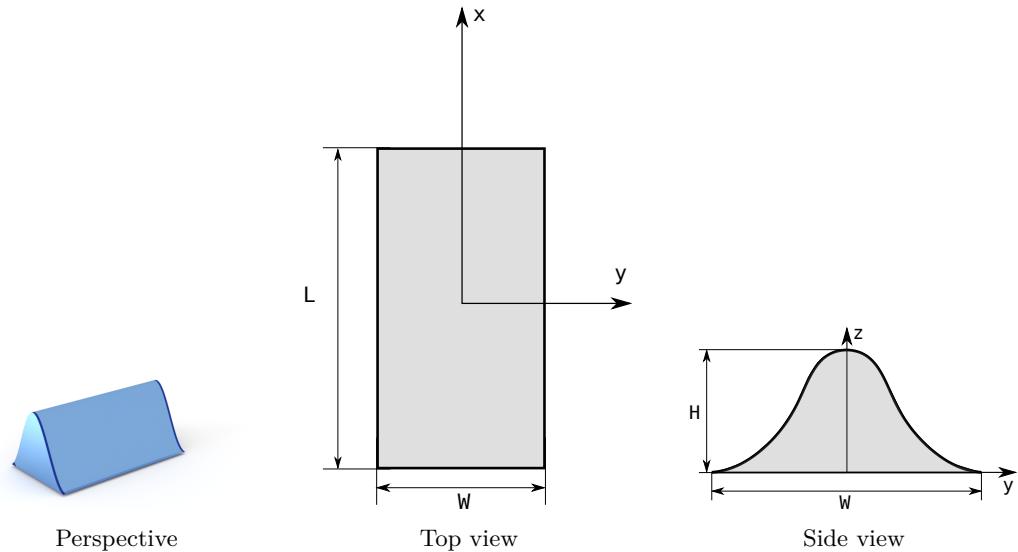


Figure 3.1: A ripple with a sinusoidal profile.

Syntax and parameters

```
FormFactorRipple1(double length, double width, double height)
```

with the parameters

- `length`, L ,
- `width`, W ,
- `height`, H .

The ripple is modelled as a surface

$$Z(y) = \frac{H}{2} \left[1 + \cos \frac{2\pi y}{W} \right].$$

Form factor

Using the inverse profile

$$Y(z) = \frac{W}{2\pi} \arccos \left(\frac{2z}{H} - 1 \right),$$

the form factor is computed by numeric integration:

$$F = L \operatorname{sinc} \left(\frac{q_x L}{2} \right) \int_0^H dz e^{iq_z z} 2Y(z) \operatorname{sinc} (q_y Y(z)).$$

The integration is substantially accelerated by the substitution $u = \arccos(2z/H - 1)$.

Volume, horizontal section

$$V = \frac{LWH}{2},$$

$$S = LW.$$

Examples

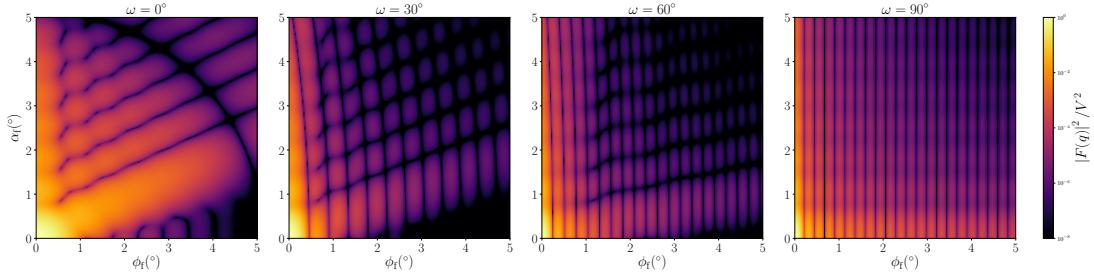


Figure 3.2: Normalized intensity $|F|^2/V^2$, computed with $L = 25$ nm, $W = 10$ nm and $H = 8$ nm, for four different angles ω of rotation around the z axis.

History

Agrees with the *Ripple1* form factor of [FitGISAXS](#) [6].

3.2 Ripple2 (saw-tooth)

Real-space geometry

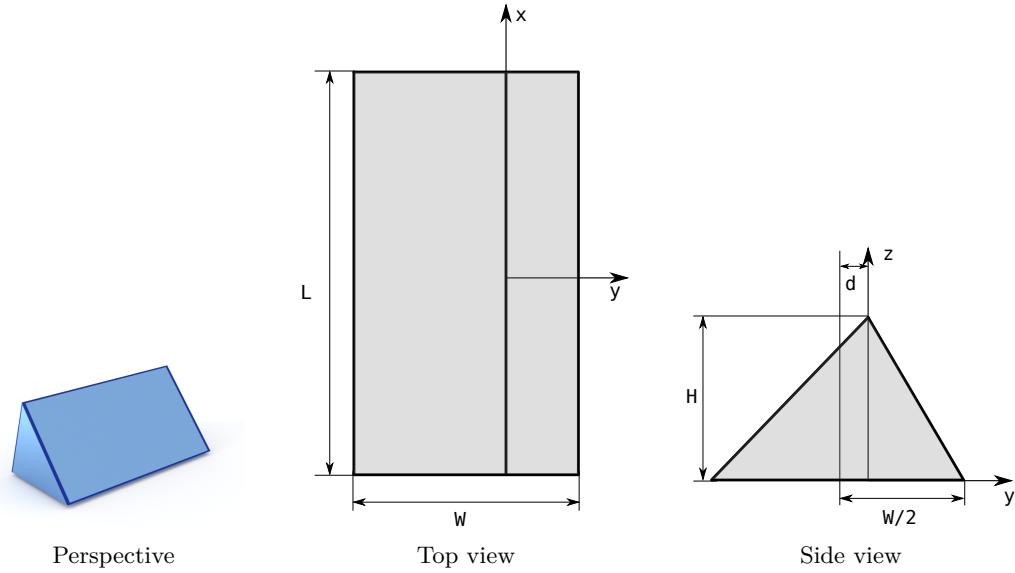


Figure 3.3: A ripple with an asymmetric saw-tooth profile.

Syntax and parameters

```
FormFactorRipple2(double length, double width, double height,
                   asymmetry)
```

with the parameters

- `length`, L ,
- `width`, W ,
- `height`, H .
- `asymmetry`, d .

They must fulfill

$$|d| \leq W/2.$$

Form factor, volume, horizontal section

$$F = \frac{LH}{q_y} \operatorname{sinc}\left(\frac{q_x L}{2}\right) i e^{-i q_y d} \left[e^{i \alpha_- / 2} \operatorname{sinc}\left(\frac{\alpha_+}{2}\right) - e^{i \alpha_+ / 2} \operatorname{sinc}\left(\frac{\alpha_-}{2}\right) \right],$$

$$\alpha_+ = H q_z + \frac{q_y W}{2} + q_y d, \quad \alpha_- = H q_z - \frac{q_y W}{2} + q_y d,$$

$$V = \frac{LWH}{2},$$

$$S = LW.$$

Examples

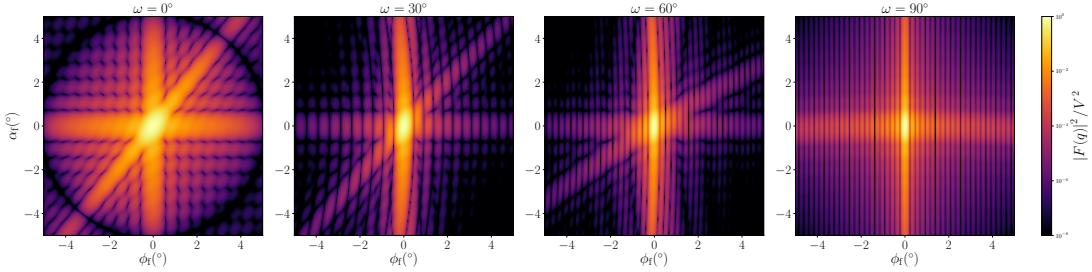


Figure 3.4: Normalized intensity $|F|^2/V^2$, computed with $L = 25$ nm, $W = 10$ nm, $H = 8$ nm, and $d = 5$ nm, for four different angles ω of rotation around the z axis. The low symmetry requires other angular ranges than used in most other figures.

History

Agrees with the *Ripple2* form factor of *FitGISAXS* [6].

Bibliography

- [1] R. Lazzari, *IsGISAXS*. Version 2.6. <http://www.insp.jussieu.fr/oxydes/IsGISAXS/isgisaxs.htm> (2006). 5, 11, 13, 15, 17, 19, 21, 25, 27, 30, 32, 36, 38, 40, 42, 46, 48
- [2] G. Renaud, R. Lazzari and F. Leroy, Surf. Sci. Rep. **64**, 255 (2009). 5, 11, 13, 15, 17, 19, 21, 25, 27, 30, 32, 36, 38, 40, 42, 46, 48
- [3] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions*, National Bureau of Standards (1964). 5, 21
- [4] J. Wuttke, *Form factor (Fourier shape transform) of polygon and polyhedron*. <https://arxiv.org/abs/1703.00255> (2017). 5, 10, 16, 17, 18, 19, 22, 23, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44
- [5] G. Pospelov, W. Van Herck, J. Burle, J. M. Carmona Loaiza, C. Durniak, J. M. Fisher, M. Ganeva, D. Yurov and J. Wuttke, J. Appl. Cryst. **53**, 262 (2020). 6
- [6] D. Babonneau, *FitGISAXS*. Version 130531. http://www.pprime.fr/sites/default/files/pictures/d1/FINANO/FitGISAXS_130531.zip (2013). 11, 36, 42, 51, 53
- [7] I. S. Gradshteyn, I. M. Ryzhik, A. Jeffrey and D. Zwillinger, *Table of Integrals, Series, and Products*, Academic Press: Amsterdam (72007). 28
- [8] A. Guinier, Ann. Phys. (Paris) **11**, 161 (1939). 30
- [9] R. W. Hendricks, J. Schelten and W. Schmatz, Philos. Mag. **30**, 819 (1974). 44

Index

- Absorption, 5
- Anisotropic pyramid, 10
- Ball, *see* Sphere
- Bessel function, 5
- Box, 12
- Cap, 45
- Citation, 4
- Cone, *see* Frustum
- Coordinates, 5
- CreateRotateX, 9
- Cube, 12
 - facetted, 43
- Cuboctahedron, 18
- Cuboid, 12
- Cylinder, 20
 - ellipsoidal, 24
- Distorted-wave Born approximation, 4
- Dodecahedron, 22
- Ellipsoid
 - truncated, 31
- Ellipsoidal cylinder, 24
- Facetting
 - cube, 43
- FormFactorAnisoPyramid, 10
- FormFactorBox, 12
- FormFactorCone, 14
- FormFactorCone6, 16
- FormFactorCuboctahedron, 18
- FormFactorCylinder, 20
- FormFactorDodecahedron, 22
- FormFactorEllipsoidalCylinder, 24
- FormFactorFullSphere, 26
- FormFactorFullSpheroid, 29
- FormFactorHemiEllipsoid, 31
- FormFactorIcosahedron, 33
- FormFactorPrism3, 35
- FormFactorPrism6, 37
- FormFactorPyramid, 39
- FormFactorRipple1, 50
- FormFactorRipple2, 52
- FormFactorTetrahedron, 41
- FormFactorTruncatedCube, 43
- FormFactorTruncatedSphere, 45
- FormFactorTruncatedSpheroid, 47
- Frustum
 - circular base, 14
 - hexagonal base, 16
 - reactangular base, 10
 - square base, 39
 - triangular base, 41
- Full sphere, 26
- Full spheroid, 29
- Hemi ellipsoid, 31
- Icosahedron, 33
- J_1 (Bessel function), 5
- Large particles
 - numeric difficulty, 6
- Mirror planes, 6
- Monte-Carlo integration
 - for large particle form factor, 6
- Numeric difficulty
 - form factor oscillation, 6
- Orientation of particles, 9
- Oscillation
 - from large particle form factor, 6
- Particle
 - hard, 8–48
 - rapid form factor oscillation, 6
- Platonic solid
 - cube, 12

dodecahedron, 22
icosahedron, 33
octahedron, 18
tetrahedron, 41
Plotting, 6
Polyhedron
 generic algorithm, 5
Prism
 hexagonal, 37
 reactangular, 12
 triangular, 35
Pyramid
 hexagonal, 16
 rectangular, 10, 41
 square, 39
 q (scattering vector), 5
Quadrature, 5
Refractive index, 6
Ripple, 49–53
 saw-tooth, 52
 sinusoidal, 50
Rotation of particles, 9
Saw-tooth ripple, 52
Scattering vector, 5
Segment
 spherical, 45
Shape transform, 5
sinc (sinus cardinalis), 5
Singularity
 in form factor computation, 5
Sinusoidal ripple, 50
Sphere, 26
 cap, 45
 segment, 45
Spheroid, 29
 truncated, 47
Surface, 5
Symmetry, 6
Tetrahedron, 41
Tilt, *see* Rotation
Transform3D, 9
Truncation, *see also* Facetting
 cone, *see* Frustum
 ellipsoid, 31
 pyramid, *see* Frustum
 sphere, 45
 spheroid, 47
Volume, 5, 6
 z (surface normal coordinate), 5