Journal of
Applied
Crystallography

ISSN 1600-5767

Received 17 May 2013
Accepted 17 October 2013

Modelling of high-symmetry nanoscale particles by small-angle scattering[1]

Cassio Alves,[a] Jan Skov Pedersen[b] and Cristiano Luis Pinto Oliveira[a]*

[a]Institute of Physics, University of São Paulo, Rua do Matão 187, São Paulo, São Paulo, 05314-970, Brazil, and [b]Department of Chemistry, University of Aarhus, Gustav Wieds Vej 14, Aarhus, DK-8000 C, Denmark. Correspondence e-mail: crislpo@if.usp.br

A versatile procedure to build high-symmetry objects and to calculate their corresponding small-angle scattering intensity is presented. Starting from a set of vertex positions, available from a large and extensible database, it is possible to build several types of bodies using spherical subunits. A fast implementation, based on the Debye formula using a histogram of distance, is then used to compute the theoretical scattering intensity. Since the model is built from the definition of a small set of parameters, it is possible to perform an optimization of structural parameters against experimental data. Finally, affine size polydispersities can be easily included by the rescaling of the histogram of the positions used in the calculations. Several examples of the calculations are presented, demonstrating the method and its applicability.

1. Introduction

Particles with nanoscale dimensions have an enormous range of applications in almost all areas. From colloidal suspensions to magnetic materials, nanoparticles play a very important role since they might control the chemical and physical properties of such systems (Jones, 2002; Uchida *et al.*, 2007). Therefore, there is a large group of techniques that are aimed at the characterization of such systems, providing important and, in several cases, complementary information. Some of the key techniques for such studies are the scattering methods. In this class, small-angle scattering (SAS) can give important information about the particle dimensions, shapes, polydispersity, orientation *etc.*, among several other structural parameters. Since it is a simple and non-invasive technique, it can be applied in a broad range of applications, particularly to investigate colloidal systems directly in solution. A reasonably large region of the sample is illuminated by the X-ray or neutron beam and therefore the scattering data are a result of an ensemble average over a very large number of particles. Because of these properties the SAS technique is a unique way to investigate nanoscaled particles.

An interesting example of nanoparticles is the particles obtained by programmed assembly of DNA structures. With recent important developments in bioengineering and bioinformatics, it is possible to design sequences of DNA that can assemble in programmed pre-defined shapes. There are several examples of such constructs in the literature. The first proposition of DNA self-assembled structures in the literature was made by Seeman for an assembly with a cubic shape (Chen & Seeman, 1991), and later for a truncated octahedron (Zhang & Seeman, 1994). However, those propositions were only theoretical models. The approach was then successfully implemented experimentally, providing the self-assembly of DNA in the form of an octahedron (Shih *et al.*, 2004; Andersen *et al.*, 2008; Falconi *et al.*, 2009; He *et al.*, 2010), a tetrahedron (Goodman *et al.*, 2005; He *et al.*, 2008; Kato *et al.*, 2009; Li *et al.*, 2009), trigonal bipyramids (Erben *et al.*, 2007), a dodecahedron (He *et al.*, 2008), an icosahedron (Zhang *et al.*, 2008; Bhatia *et al.*, 2009) and a truncated icosahedron (He *et al.*, 2008). It was also possible to build hybrid structures with synthetic connections at the vertices (Aldaye & Sleiman, 2007; Zimmermann *et al.*, 2008; Yang *et al.*, 2009), and large structures, based on the development of a DNA origami technique (Rothemund, 2006; Dietz *et al.*, 2009; Douglas *et al.*, 2009; Endo *et al.*, 2009; Ke *et al.*, 2009). In other applications, it was shown that it was possible to encapsulate proteins (Erben *et al.*, 2006) and nanoparticles (Erben *et al.*, 2007; Yang *et al.*, 2009) inside of a DNA polyhedron. There are several examples of the assembly of DNA nanostructures with high symmetry (*e.g.* Seeman, 2010), and Han *et al.* (2011) have used the DNA origami approach to build structures with high curvature. In all cases, after the experimental assembly, it is necessary to be able to confirm the formed shape and the assembly efficiency. One common procedure to check the assembly formation is gel chromatography (Douglas *et al.*, 2009), where by the indication of the molecular weight, the assembly can be inferred. Although this procedure gives a good indication of the correct assembly, it is not a measure of it. Electron microscopy techniques permit a direct visualization of such structures and can, in several cases, provide a three-dimensional reconstruction (Kato *et al.*, 2009). Nevertheless, such techniques can have limitations since they might require quite invasive sample treatment for the experiments, which can influence the results. Also the information obtained

[1] This article will form part of a virtual special issue of the journal, presenting some highlights of the 15th International Small-Angle Scattering Conference (SAS2012). This special issue will be available in early 2014.

about the assembly efficiency on the system is limited because such methods usually cover a very small volume of the system (Andersen *et al.*, 2008). Scattering methods, on the other hand, are non-invasive methods, which can be applied directly in solution and provide ensemble-averaged results on a very large number of particles. The main drawback of the scattering techniques is that they provide information in reciprocal (Fourier) space. It has been shown that an ideal situation is the combination of electron microscopy techniques and scattering methods, since it might enable a better investigation of the system (Andersen *et al.*, 2009; Oliveira *et al.*, 2010) with direct visualization of the structure as well as information on the ensemble-average structure of the particles.

Although scattering experiments are relatively simple, the data analysis and interpretation can be quite complicated. The scattering intensity of a sample is given as the absolute square of the Fourier transformation of the scattering length density contrast with additional ensemble-average and orientation-average operations. Because of this, analytical expressions are restricted to simple shapes and geometries (Pedersen, 1997). More complicated shapes cannot be described analytically and, for such cases, alternative methods are required.

In this work a useful procedure is presented, where models with a large number of possible geometries can be easily built and the corresponding scattering intensity calculated. Since the number of parameters describing the models is very low, they can be quickly optimized against experimental data. A special scheme is presented to efficiently model the assembled DNA structures.

2. Theoretical aspects, methods and results

2.1. Small-angle scattering (SAS) and model calculation

In a typical SAS experiment, the incoming beam passes through the sample, probing a large number of randomly oriented particles, which then leads to an ensemble averaging of the total scattering intensity that also includes averaging over the orientation of the particles. A general expression for the measured scattering intensity is given by

$$I(q) = N\Delta\rho^2 V^2 P(q) S_{\rm m}(q). \qquad (1)$$

In this expression, q is the modulus of the scattering vector [defined as $q = (4\pi/\lambda)\sin\theta$, where 2θ is the scattering angle and λ is the wavelength of the monochromatic beam], N is the average number of particles illuminated by the beam, $\Delta\rho$ is the scattering length density contrast between the particles and the medium in which they are immersed, and V is the particle volume. $P(q)$ is the normalized particle form factor [$P(0) = 1$], which is the squared Fourier transformation of the particle scattering length density contrast, averaged over all possible particle orientations:

$$P(q) = \left\langle \frac{A(q)^2}{A(0)^2} \right\rangle_\Omega = \left\langle \left[\frac{\int_V \Delta\rho(\mathbf{r}) \exp(i\mathbf{q}\cdot\mathbf{r}) {\rm d}\mathbf{r}}{\int_V \Delta\rho(\mathbf{r}) {\rm d}\mathbf{r}} \right]^2 \right\rangle_\Omega . \qquad (2)$$

Here $\mathbf{q}$ is the three-dimensional scattering vector, $\Delta\rho(\mathbf{r})$ is the particle scattering length density contrast distribution at position $\mathbf{r}$ and the angle brackets $\langle\rangle$ indicate an average over the solid angle Ω. This equation reflects that the form factor $P(q)$ is directly related to the particle shape. It is interesting to note that, for X-rays, the scattering length density contrast is proportional to the difference in electron density between the particles and the medium. The term $S_{\rm m}(q)$ is the (effective) structure factor, which describes possible particle interactions. For very dilute systems, the structure factor always approaches a constant value of unity and, therefore, the expression for this case reduces to

$$I(q) = N\Delta\rho^2 V^2 P(q) = N I_1(q), \qquad (3)$$

where the last equal sign is valid for identical particles without any internal degrees of freedom like flexibility, so that $I_1(q)$ is the scattering intensity of a single particle. Therefore the measured intensity, which corresponds to the sum of a large number of scatterers, is a direct representation of the scattering from a single particle and can provide information about its structure. There are several methods in the literature for data analysis and modelling of SAS data (Glatter, 1977; Glatter & Kratky, 1982; Feigin & Svergun, 1987; Semenyuk & Svergun, 1991; Pedersen, 1997). However, the application of traditional methods for programmed self-assembly structures has some intrinsic peculiarities. Even with the loss of information related to the ensemble average of the randomly oriented particles, the so called *ab initio* methods can provide reasonable three-dimensional models directly from the one-dimensional experimental data (Svergun & Stuhrmann, 1991; Chacón *et al.*, 1998; Svergun, 1999; Walther *et al.*, 2000). These methods can also be combined with knowledge of the atomic resolution structure of proteins, DNA and RNA in order to obtain the overall organization of the complex that fits the scattering data (Petoukhov & Svergun, 2005). However, the restored models are not unique and there might be several possible models that fit the experimental data (Svergun *et al.*, 1996; Volkov & Svergun, 2003; Petoukhov & Svergun, 2013). In order to increase the reliability of such procedures, the modelling of the scattering data has been combined with external information, for example, the inclusion of model constraints, symmetry conditions and contrast variation data (mostly available for neutron scattering), and also, in very recent applications, the inclusion of simultaneous modelling of scattering and NMR data (Evrard *et al.*, 2011; Wassenaar *et al.*, 2012). Such procedures and methods constitute a very useful tool box for modelling structures for biological complexes in a growing number of applications (Petoukhov & Svergun, 2013). However, there are some cases where such methods can fit the scattering data but the resulting models are not in agreement with the known shape. Such a limitation occurs either from the lack of external information that can be used or from the lack of the proper symmetry constraint available in the programs. One example is the case of the programmed self-assembly of DNA cages, which can adopt highly symmetrical shapes (Douglas *et al.*, 2009) on which it might be difficult to apply *ab initio* methods in the modelling of the corresponding scattering data (Andersen *et al.*, 2008; Oliveira *et al.*, 2010). On the other hand, since the expected shape for

these cases is known, one could *a priori* build the scattering length density profile $\rho(\mathbf{r})$ and calculate the theoretical intensity. However, this approach depends on the calculation of a Fourier integral and the orientation averaging, which can be solved analytically or semi-analytically only for simple shapes, like spheres, cylinders and ellipsoids (Pedersen, 1997). For more complex shapes, equation (2) is not directly applicable and alternative methods have to be used. One very powerful method is the finite element technique, where the model is built using small subunits like spheres or points (Glatter, 1972). In the case of spherical subunits, the normalized scattering intensity can be calculated using of the Debye formula (Debye, 1915):

$$I(q) = (1/M_{\rm Tot}^2)\sum_{i,j}^{P} M(i)M(j)A_i(q)A_j(q)\sin(qr_{ij})/(qr_{ij}), \quad (4)$$

where $A_i(q)$ and $M(i)$ are, respectively, the normalized scattering amplitude and the scattering mass [$M(i) = V(i)\Delta\rho$, the product of the subunit volume and the scattering length density contrast] of the ith homogeneous sphere, P is the total number of subunits in the model, r_{ij} is the distance between the ith and jth sphere, and

$$M_{\rm Tot}(i) = \sum_{i,j}^{P} M(i). \quad (5)$$

The computation requires prior construction of a model composed by the known subunits. It should be noted that the term subunits refers only to the small beads that are used to build the model. Having the model, the calculation can be very time consuming for models composed of a large number of subunits since equation (4) involves a double summation and the computation time grows with the square of the number of subunits [$O(P^2)$].

In order to speed up the calculation, it is possible to replace the double sum by a single sum on the histogram of pair distances $h(r)$. This histogram is calculated using all the subunits on the model and it is weighted by the local electron density of each point (Glatter, 1972; Pantos & Bordas, 1994):

$$I(q) = (1/M_{\rm Tot}^2)\left[\sum_{i=1}^{P} I(q) + 2A^2(q)\sum_{k=1}^{n_{\rm bins}} h(r_k)\sin(qr_k)/(qr_k)\right], \quad (6)$$

where $n_{\rm bins}$ is the total number of channels (bins) in the histogram, $A(q)$ is the normalized form factor amplitude of the subunit and $h(r)$ is the histogram of pairs of distances. In this calculation, the approximation is made by the discretization of the distances over the histogram channels. This implementation permits the fast computation of the theoretical scattering intensity since the sum is now made over the histogram channels and, therefore, the computation cost is of the order of the number of channels $O(n_{\rm bins})$. The construction of the histogram, however, still goes as P^2 but it is performed only once for all q values.

If the number of points that composes the model is sufficiently large (typically more than 5000 subunits), it is also possible to speed up the histogram calculation. This is performed by the division of the points into several blocks. The definition of the block size can be coupled to the maximum value of the scattering vector, $q_{\rm max}$. As a rule of thumb, the subunit radius has to be smaller than $\pi/q_{\rm max}$ to eliminate any influence of the subunit size on the calculation of the scattering intensity (Glatter, 1972). This information can also be used for an automatic determination of the block size. For each block the histogram is calculated individually, and at the end, all the contributions are summed. As also shown by other authors (Hansen, 1990; Pedersen *et al.*, 2012), this procedure speeds up the computation of the histogram since it now runs with $\sim(P^2/n_{\rm blocks})$, where $n_{\rm blocks}$ is the number of blocks the system is split into. Finally, having the histogram, one can easily introduce affine polydispersities by the rescaling of the histogram itself (Pedersen *et al.*, 2012). In this way the 'affine polydispersity' indicates that all dimensions of the model are scaled by the same relative value. Another very effective procedure for fast computation of scattering intensities is the use of spherical harmonics as proposed by Stuhrmann and Svergun (Stuhrmann, 1970; Svergun & Stuhrmann, 1991; Svergun, 1999), but this approach is not used in the present work.

2.2. Short program description

The procedures presented in this work enable the modelling, simulation and optimization of bodies with different geometries and shapes. A database with more than 100 different shapes (see the complete list in Table 1) containing information of vertex position and edge connectivity for the structures is used. There is also an implementation that allows the automatic connectivity of any set of points in a three-dimensional arrangement. Such large feasibility is to our knowledge not available in any of the procedures present in the literature (Zhou *et al.*, 2005). With this information different types of models can be built: solid, shell-like and edge-like structures. The models are represented by the use of spherical subunits, which allows the precise computation of the theoretical intensity. The main advantage of this approach is that the model construction only requires a very low number of parameters for its description, even for very complicated geometrical bodies. This enables an easy optimization of the model against experimental data. The optimization is performed by the use of standard least-squares methods (Press, 1992; Pedersen, 1997). The routines were written in Fortran90. Images of the models were built using the programs *MolMol* (Koradi *et al.*, 1996), the UCSF *Chimera* package (Pettersen *et al.*, 2004) and the *PyMOL* Molecular Graphics System (Schrödinger LLC, Version 1.3, 2010; http://www.pymol.org).

2.3. Procedures to create models composed of subunits

In the present work, two general strategies were used to build the models composed of subunits. In one case, the subunits with a given radius form a compact set with well defined positions to build up the model. This approach, which will be called the finite element (FE) method, is used for the

Table 1
Structures with high symmetry.

0	Tetrahedron	57	Pentagonal dipyramid (J13)	100	Metabiaugmented hexagonal prism (J56)
1	Cube	58	Elongated triangular dipyramid (J14)	101	Triaugmented hexagonal prism (J57)
2	Octahedron	59	Elongated square dipyramid (J15)	102	Augmented dodecahedron (J58)
3	Dodecahedron	60	Elongated pentagonal dipyramid (J16)	103	Parabiaugmented dodecahedron (J59)
4	Icosahedron	61	Gyroelongated square dipyramid (J17)	104	Metabiaugmented dodecahedron (J60)
5	Small stellated dodecahedron	62	Elongated triangular cupola (J18)	105	Triaugmented dodecahedron (J61)
6	Great dodecahedron	63	Elongated square cupola (J19)	106	Metabidiminished icosahedron (J62)
7	Great stellated dodecahedron	64	Elongated pentagonal cupola (J20)	107	Tridiminished icosahedron (J63)
8	Great icosahedron	65	Elongated pentagonal rotunda (J21)	108	Augmented tridiminished icosahedron (J64)
22	Triangular prism	66	Gyroelongated triangular cupola (J22)	109	Augmented truncated tetrahedron (J65)
23	Pentagonal prism	67	Gyroelongated square cupola (J23)	110	Augmented truncated cube (J66)
24	Hexagonal prism	68	Gyroelongated pentagonalcupola (J24)	111	Biaugmented truncated cube (J67)
25	Octagonal prism	69	Gyroelongated pentagonal rotunda (J25)	112	Augmented truncated dodecahedron (J68)
26	Decagonal prism	70	Gyrobifastigium (J26)	113	Parabiaugmented truncated dodecahedron (J69)
27	Square antiprism	71	Triangular orthobicupola (J27)	114	Metabiaugmented truncated dodecahedron (J70)
28	Pentagonal antiprism	72	Square orthobicupola (J28)	115	Triaugmented truncated dodecahedron (J71)
29	Hexagonal antiprism	73	Square gyrobicupola (J29)	116	Gyraterhombicosidodecahedron (J72)
30	Octagonal antiprism	74	Pentagonal orthobicupola (J30)	117	Parabigyraterhombicosidodecahedron (J73)
31	Decagonal antiprism	75	Pentagonal gyrobicupola (J31)	118	Metabigyraterhombicosidodecahedron (J74)
33	Rhombic dodecahedron	76	Pentagonal orthocupolarontunda (J32)	119	Trigyraterhombicosidodecahedron (J75)
34	Triakis octahedron	77	Pentagonal gyrocupolarotunda (J33)	120	Diminished rhombicosidodecahedron (J76)
35	Tetrakis hexahedron	78	Pentagonal orthobirotunda (J34)	121	Paragyrate diminished rhombicosidodecahedron (J77)
36	Trapezoidal icositetrahedron	79	Elongated triangular orthobicupola (J35)	122	Metagyrate diminished rhombicosidodecahedron (J78)
37	Hexakis octahedron	80	Elongated triangular gyrobicupola (J36)	123	Bigyrate diminished rhombicosidodecahedron (J79)
38	Pentagonal icositetrahedron (dextro)	81	Elongated square gyrobicupola (J37)	124	Parabidiminishedrhombicosidodecahedron (J80)
39	Rhombic triacontahedron	82	Elongated pentagonal orthobicupola (J38)	125	Metabidiminishedrhombicosidodecahedron (J81)
40	Triakis icosahedron	83	Elongated pentagonal gyrobicupola (J39)	126	Gyrate bidiminished rhombicosidodecahedron (J82)
41	Pentakis dodecahedron	84	Elongated pentagonal orthocupolarotunda (J40)	127	Tridiminished rhombicosidodecahedron (J83)
42	Trapezoidal hexecontahedron	85	Elongated pentagonal gyrocupolarotunda (J41)	128	Snub disphenoid (J84)
43	Hexakis icosahedron	86	Elongated pentagonal orthobirotunda (J42)	129	Snub square antiprism (J85)
44	Pentagonal hexecontahedron (dextro)	87	Elongated pentagonal gyrobirotunda (J43)	130	Sphenocorona (J86)
45	Square pyramid (J1)	88	Gyroelongated triangular bicupola (J44)	131	Augmented sphenocorona (J87)
46	Pentagonal pyramid (J2)	89	Gyroelongated square bicupola (J45)	132	Sphenomegacorona (J88)
47	Triangular cupola (J3)	90	Gyroelongated pentagonal bicupola (J46)	133	Hebesphenomegacorona (J89)
48	Square cupola (J4)	91	Gyroelongated pentagonal cupolarotunda (J47)	134	Disphenocingulum (J90)
49	Pentagonal cupola (J5)	92	Gyroelongated pentagonalbirotunda (J48)	135	Bilunabirotunda (J91)
50	Pentagonal rotunda (J6)	93	Augmented triangular prism (J49)	136	Triangular hebesphenorotunda (J92)
51	Elongated triangular pyramid (J7)	94	Biaugmented triangularprism (J50)	137	Tetrahemihexahedron
52	Elongated square pyramid (J8)	95	Triaugmented triangular prism (J51)	138	Octahemioctahedron
53	Elongated pentagonal pyramid (J9)	96	Augmented pentagonal prism (J52)	139	Small ditrigonal icosidodecahedron
54	Gyroelongated square pyramid (J10)	97	Biaugmented pentagonal prism (J53)	140	Dodecadodecahedron
55	Gyroelongated pentagonal pyramid (J11)	98	Augmented hexagonal prism (J54)	141	Echidnahedron
56	Triangular dipyramid (J12)	99	Parabiaugmented hexagonal prism (J55)		

construction of models with a relatively small number of subunits. In the other case, the model is built by the use of a large number of points, which are randomly distributed within the structure. This approach, which will be called the Monte Carlo (MC) method, usually uses a larger number of subunits. Depending on the application, one method might be more suitable than the other, as will be demonstrated by the examples below. In general, the MC method can be used to build complex models, either solid or shell-like, since the large number of subunits can be combined to describe the interfaces. On the other hand, the FE method uses a small number of subunits and is suitable for cases where the subunit itself can be seen as an important model element. For example, for the cases of DNA-like structures or protein-like structures, the subunits can describe the backbone of such structures. Also, the two approaches can be combined, permitting a more flexible procedure. One example of such an application is shown in Fig. 1, where a cube is built. The corners constitute the vertices and a model can be generated by placing an array of small subunits along the edges of the cubes. The construction of other polyhedron types is very easy if information

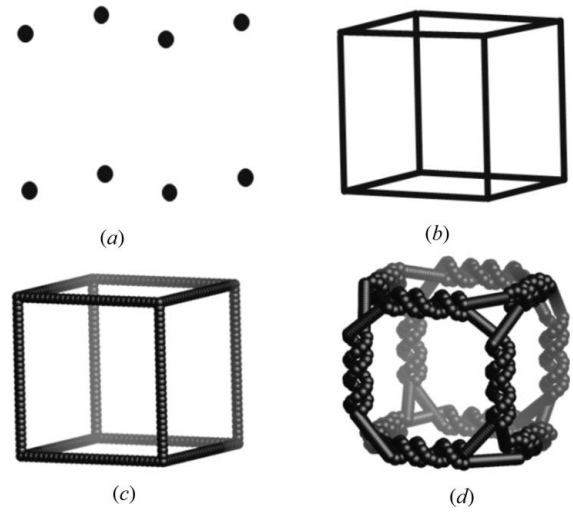

Figure 1
Construction of the model. Starting from the set of vertex positions (*a*), the skeleton of the model can be built by just connecting the vertices (*b*). In order to enable an easy computation of the intensity the vertices can be represented using spherical subunits (*c*). The vertices can be connected using more elaborate edges like ones with DNA-like double helices (*d*).

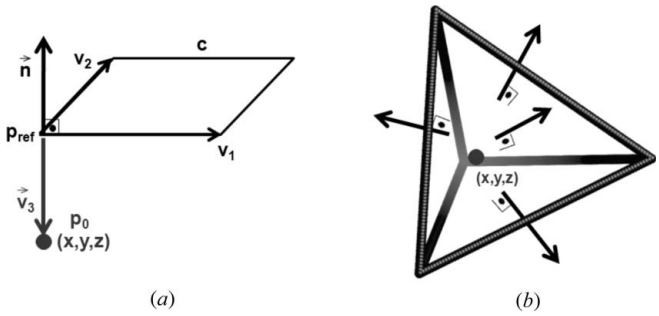

Figure 2
Considerations about positions and planes. The vector **n** is the normal to the plane defined by the vectors $\mathbf{v}_1$ and $\mathbf{v}_2$. Also, the vector $\mathbf{v}_3$ connects the reference point $\mathbf{p}_{ref}$ on the plane to a given point $\mathbf{p}_0$. (*a*) If $\mathbf{v}_3 \cdot \mathbf{n} = 0$, $\mathbf{p}_0$ is in the plane. If $\mathbf{v}_3 \cdot \mathbf{n} < 0$, $\mathbf{p}_0$ is on the negative semi-space of the plane, and if $\mathbf{v}_3 \cdot \mathbf{n} > 0$, $\mathbf{p}_0$ is on the positive semi-space of the plane. (*b*) $\mathbf{p}_0$ is inside of the tetrahedron when $\mathbf{v}_3 \cdot \mathbf{n} < 0$ for all vectors $\mathbf{n} = \mathbf{v}_1 \times \mathbf{v}_2$ built with the vectors $\mathbf{v}_1$, $\mathbf{v}_2$ oriented in a counterclockwise direction on the faces.

about the positions of the vertices and faces is available. Such information on Cartesian coordinates of the vertices as well as the connectivity between them is indeed available in the literature (Coxeter, 1973) and on the web (Andrew Hume's Polyhedron Database; http://www.netlib.org/polyhedra/). An automatic procedure was developed in the present work for reading this information and building several types of models from the vertices. It is possible to construct the skeleton of the polyhedron, placing spheres at the edge positions (Fig. 1*c*). It is also possible to place DNA-like structures at the edges and connect the double strands at a given corner in an appropriate way (Fig. 1*d*). The latter approach uses the FE method and it has already been successfully applied for the study of truncated octahedron systems (Oliveira *et al.*, 2010).

If the information about the connectivity between the edges is not known, a prior procedure for generating this is required. Given a set of coordinates representing the polyhedron vertices, is possible to use an adaptation of the Chand & Kapur (1970) algorithm and the Jarvis (1973) algorithm for generating the skeleton of the polyhedron. The data structure used in this case assumes that the polyhedron is convex, *i.e.* each edge divides no more than two faces and each face divides the semi-space into semi-regions, and that each semi-region starts at the corresponding side of this face. In summary, the program searches for the first three vertices in a certain direction and makes a triangular face. Starting from this first triangular face, the other triangular faces are constructed by the proper selection of vertices, in such a way that the new triangles have a common edge with a previous one. The correct face is chosen in order to have the minimum angle between the current and the new face. The process is repeated until all the most peripheral points are considered. By the use of the Euler relation [equation (7)], the polyhedrons can be built automatically using the three-dimensional Gift Wrap (GW) (Preparata & Shamos, 1985) algorithm. A relation exists between the number of vertices, v, the number of faces, f, and the number of edges, e,

$$v + f = e + 2. \qquad (7)$$

Information about the position of a given point (inside or outside the object) can easily be obtained by simple scalar products. For a given face plane, the vector normal to this face can be found with three points at the plane face, $\mathbf{n} = \mathbf{v}_1 \times \mathbf{v}_2$ (Fig. 2*a*), where $\mathbf{v}_1$ and $\mathbf{v}_2$ connect the points pairwise. With the normal vector **n** it is possible to determine at which side of the plane a given point $\mathbf{p}_0(x, y, z)$ is located. Let $\mathbf{v}_3$ be a vector starting on the plane face and ending at the position $\mathbf{p}_0$. If the scalar product between the normal vector **n** and vector $\mathbf{v}_3$ is $\mathbf{v}_3 \cdot \mathbf{n} < 0$, $\mathbf{p}_0$ is placed at the internal side (negative semi-space) of the plane. If $\mathbf{v}_3 \cdot \mathbf{n} > 0$, $\mathbf{p}_0$ is placed on the external side (positive semi-space). Finally, if $\mathbf{v}_3 \cdot \mathbf{n} = 0$, $\mathbf{p}_0$ lies on the plane of the face. If we extrapolate this procedure to all polyhedral faces, it is possible to determine if $\mathbf{p}_0$ is inside or outside the body (Fig. 2*b*).

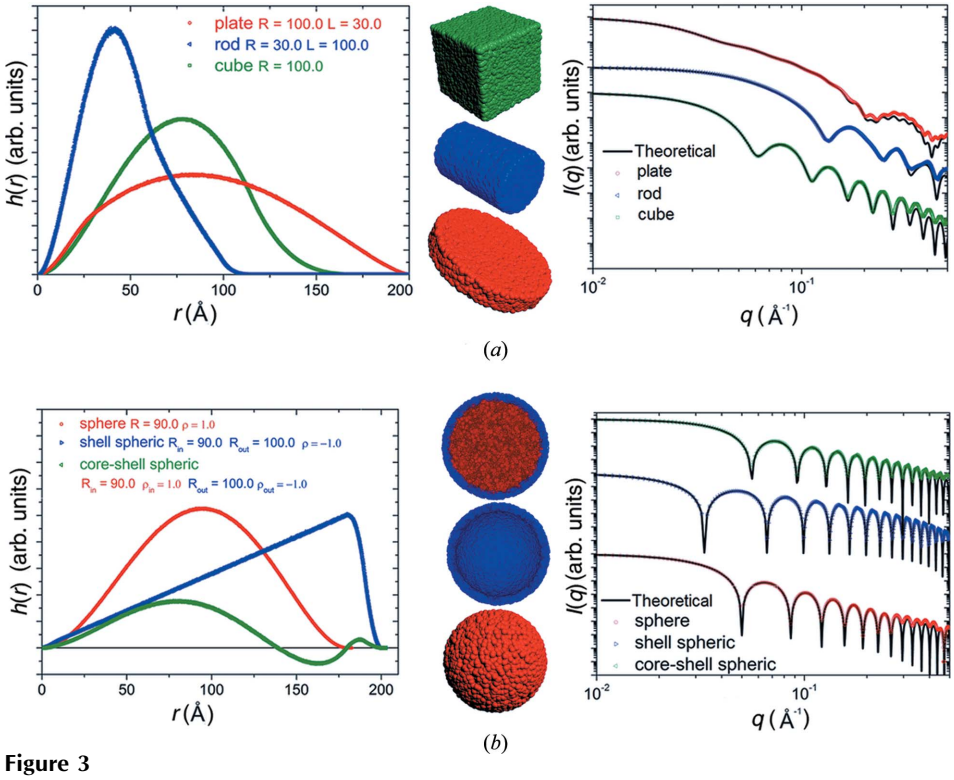

Figure 3
Computation of test examples for simple bodies. The calculation for solid spheres, spherical shells, spherical core–shell structures (*a*) and cubes, short rods and plates (*b*), compared with theoretical intensities. The very good agreement between the curves demonstrates an accurate calculation of the scattering intensity by the use of the Debye formula.

This definition of internal and external regions of a given object can be applied in several model strategies. If one considers only the edges, a skeleton-like model can easily be constructed. By the use of the above method, solid or shell-like models are easily built. This approach is the basis for the construction of more sophisticated models, as shown in the next section.

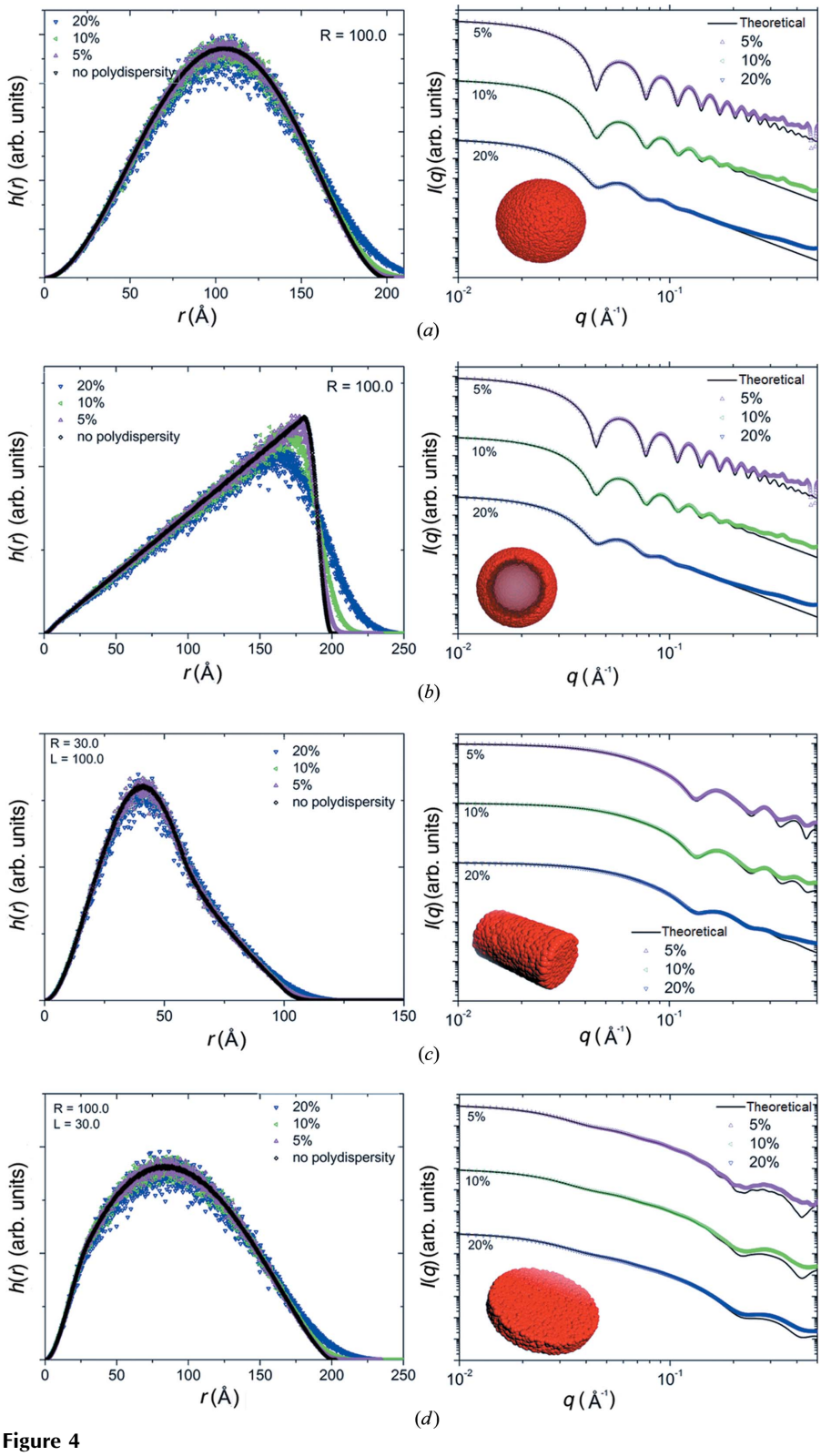

Figure 4
Computation of affine polydispersity by rescaling the histogram of distances. The simulated intensities are compared with the theoretical curves obtained from equation (8).

2.4. Test examples

In order to test the whole procedure, bodies with simple geometries were built to enable a comparison with the known analytic expressions (Pedersen, 1997). These results provide the necessary validation before the computation of the scattering for more sophisticated models. Fig. 3 shows the computation for spheres, spherical shells and core–shell models with negative/positive contrasts, as well as for cubes, rods and plates.

Affine polydispersities can be introduced in the models by the rescaling of the histogram as described by Pedersen *et al.* (2012). Briefly summarized, the bin size of the histogram is kept fixed and the original histogram is expanded or shrunk according to a given distribution. The rescaled histograms are weighted by the distribution function and the square of the scattering mass, and then projected onto the bins. As mentioned above, with this procedure all dimensions of the model are scaled by the same relative value. As a result, the final histogram can be used in the calculation of the intensity. The precision of such an approach can be confirmed by the calculation of simple geometrical bodies for which the theoretical intensity can be computed. A general expression for polydisperse systems is given by

$$I(q) = \frac{\int_0^\infty D_{\rm N}(R) M^2(R) P(q,R)\,{\rm d}R}{\int_0^\infty D_{\rm N}(R) M^2(R)\,{\rm d}R}, \tag{8}$$

where $D_{\rm N}(R)$ is the number distribution, $M(R) = V(R)\Delta\rho$ is the scattering mass given by the product of the particle volume and the scattering length density contrast, and $P(q, R)$ is the normalized form factor. It is possible to have different types of distributions by a combination of the number weight and the volume factor. For the volume weighted distribution $[D_{\rm V}(R) = D_{\rm N}(R)M(R)]$, the scattering mass term is combined with the number distribution and for the intensity weighted distribution $[D_{\rm I}(R) = D_{\rm N}(R)M^2(R)]$ the square

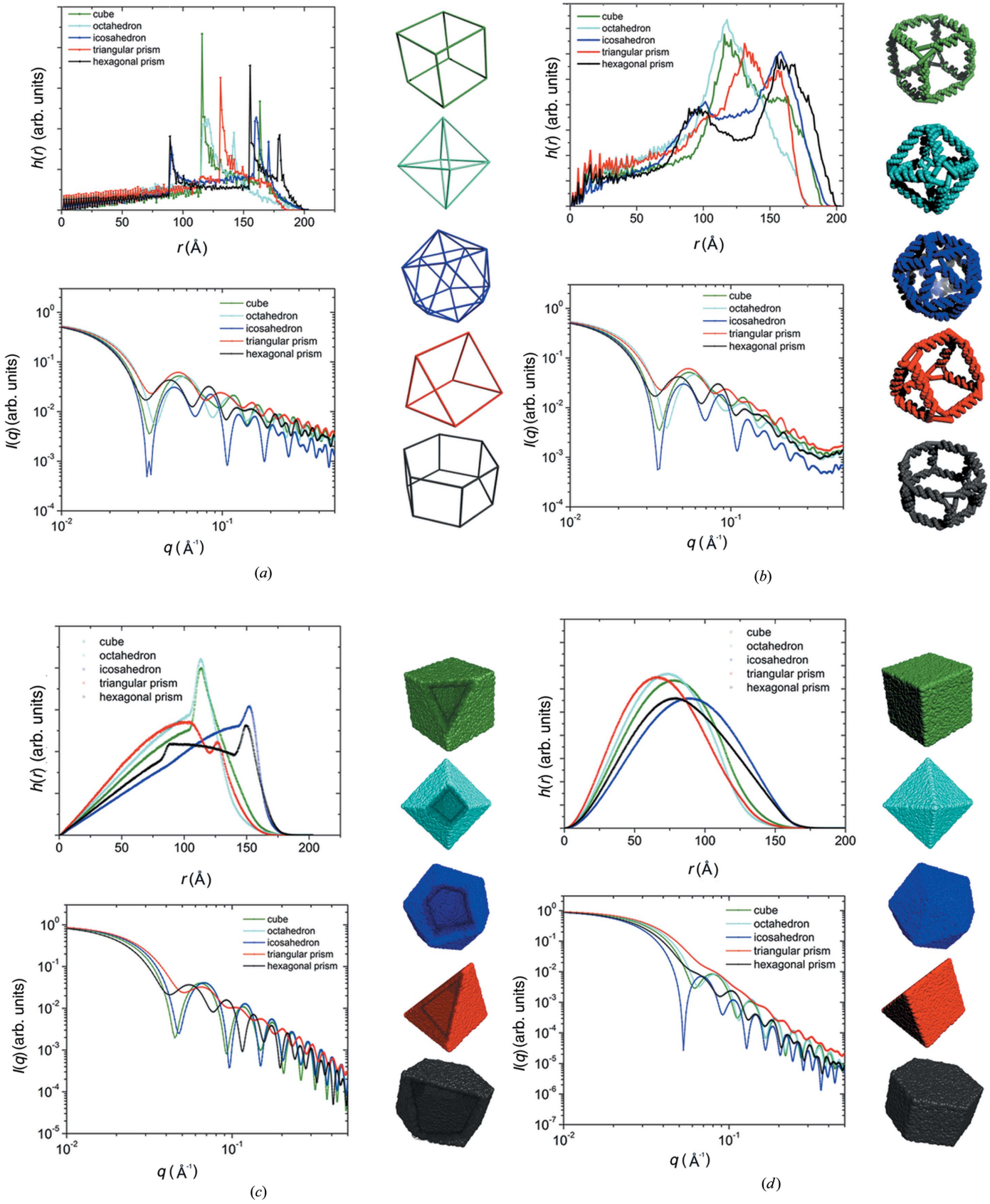

Figure 5
Calculation of theoretical intensities for several geometrical bodies. Edge-like bodies (*a*), DNA-like bodies (*b*), shell-like bodies (*c*) and solid bodies (*d*) were built and their respective histograms and theoretical intensities were computed. In all cases, the bodies were confined within a sphere of radius 100 Å.

of the scattering mass is combined with the number distribution. These variations are very useful when comparing the model with real experimental data.

Fig. 4 shows the intensity with the polydispersity calculated by the use of the histogram rescaling and the theoretical computation using the full analytical calculation [equation (8)]. In all cases, the polydispersity was assumed to have Gaussian form. One can clearly see that the affine polydispersity calculations applied on the histogram are in very good agreement with the theoretical curves. Altogether, these simulated tests give the necessary support for the application of the method to the computation of intensities of any proposed shape.

2.5. SAXS simulation of polyhedron shapes

The applicability of the method is demonstrated on the computation of scattering intensities from geometrical bodies. Starting from a set of vertices, solid, shell-like and skeleton-like bodies were built and the intensities were calculated using the above-mentioned methods.

Several examples are shown in Fig. 5. Using the same geometries (cube, octahedron, icosahedron, triangular prism and hexagonal prism), a set of constructions is shown with the respective histograms and theoretical scattering intensity. It is interesting to note that the obtained histograms and intensities are remarkably different between those bodies, indicating that the small-angle scattering technique is capable of distinguishing between them. There are also large differences between solid and shell-like structures.

As mentioned above, by the use of specially designed oligo sequences, it is possible to assemble DNA structures into specific predefined shapes. In the majority of the examples available in the literature, the structure resembles the skeleton-like structure that is shown in Fig. 5(*a*). However, the edges in this case are built by DNA strands and therefore the use of a simple array of spheres to represent the DNA helix can be a too crude approximation, as already demonstrated in the literature (Oliveira *et al.*, 2010), and which is also clearly evident from the above simulations. For modelling such structures the edges can be built using DNA-like double helices as building blocks. Those parts can be obtained from known atomic resolution structures available for DNA molecules and, in order to speed up the calculations, one can use coarse grain models by only considering the positions of the C* atoms (or any other that is selected by the user) and placing spherical subunits at these positions. In this way, the edges can be built by the double helix and the termini connected by single strands. Starting from the information on the vertices and connections (Polyhedron Database; Preparata & Shamos, 1985) an automatic procedure has been implemented, which builds the whole structure with DNA double helices and the connections with single linkers (Fig. 5*b*).

The affine polydispersities can also be included for the DNA-like structures by the use of the rescaling of the histogram. An example of such an application is shown in Fig. 6 where a polydispersity of 20% in number is included. As a result, intensity with less pronounced oscillations averaged over the distribution of sizes is obtained. This approach is not entirely correct for DNA-like structures since the number of nucleotides is fixed. As mentioned above, the affine polydispersity assumes that the model keeps its original shape but varies its size, according to some distribution function. For DNA-like structures it is known that, owing to some degree of flexibility in its structure, the particle shape can fluctuate and a more correct approach would be to create an ensemble of shapes and take averages over it. Since the correct creation of such an ensemble may demand molecular dynamics simulations, this procedure can be a very complicated and cumbersome task (Falconi *et al.*, 2009; Oteri *et al.*, 2011). Nevertheless, because of the low resolution of the SAS data, affine polydispersity might provide similar results to the ensemble average over different shapes, principally for not too large variations in shape.

There are several routes to design nanomaterials to work as nanocapsules or nanocages. Beyond the design of nanocages using DNA and blending of DNA with organic and inorganic molecules (Sleiman *et al.*, 2011), other materials can be used such as metal–organic polyhedrons (Seidel & Stang, 2002; Hamilton & MacGillivray, 2004; Perry *et al.*, 2009), viral capsids and other protein cages (Uchida *et al.*, 2007) which have also presented the possibility of use as nanocontainers or nanocapsules. In this approach, it was demonstrated that it is possible to transport different kinds of molecules inside or

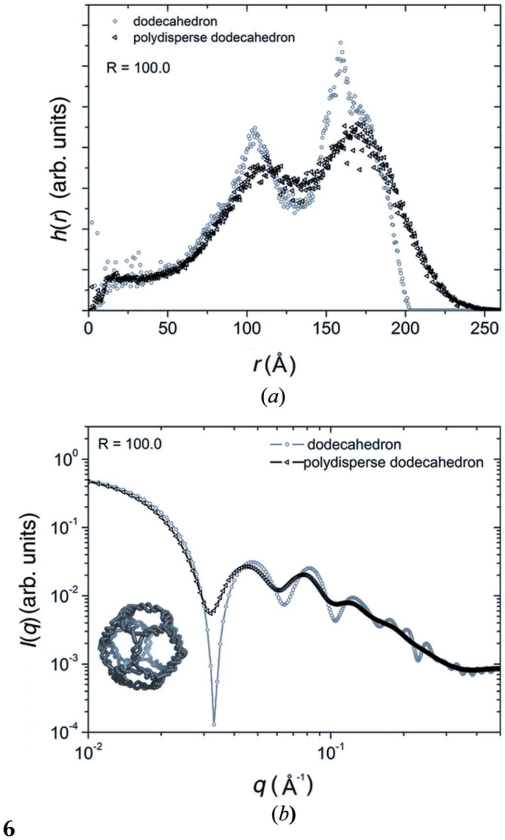

Figure 6
The influence of affine polydispersity for DNA cages. (*a*) Histogram of pair of distances. (*b*) Calculated scattering intensity. The polydispersity smears out the oscillations in the scattering intensity.

attached to the nanocapsules (Uchida *et al.*, 2007; Bhatia *et al.*, 2009; Walsh *et al.*, 2011). Inspired by these experiments, an example of the combination of shapes is presented in Fig. 7, where an icosahedral shell-like structure with a DNA cage in its interior is simulated. In this simulation the initial model of the DNA cage, built by the use of the FE approach, provided the arrangement of spheres in a well defined way. This arrangement is later replaced by random points in an MC box in such a way that the DNA cage has the same density of points as the icosahedral shell. This step is necessary to ensure the correct relative electron density values. The procedure enables an easy construction and calculation of the scattering intensity for the composed model.

As a final example, the construction of a model for a given set of points is shown in Fig. 8. In this case a set of points is selected and the GW algorithm creates the connectivity of such points by the definition of triangular faces. The points may correspond to regular polyhedra or to any random arrangement of points (Chand & Kapur, 1970; Jarvis, 1973; Preparata & Shamos, 1985). Having this information about faces and edges, all of the modelling approaches mentioned before can be applied. In the example shown in Fig. 8, two random sets of vertices are created, one with five vertices and other with nine vertices. The GW algorithm defines the outer faces of the polyhedra as triangles and, as an example of application, builds them with DNA double helices (Fig. 8*b*). The corresponding histogram and scattering intensity are shown for the case of thin edges and DNA-like edges (Figs. 8*a* and 8*c*). This procedure can be used as a general framework for modelling scattering data. The selection of random points can be combined with an heuristic procedure in order to enable the optimization of the shape against experimental data. This procedure has not yet been implemented but it might be a potential *ab initio* modelling method for SAS data.

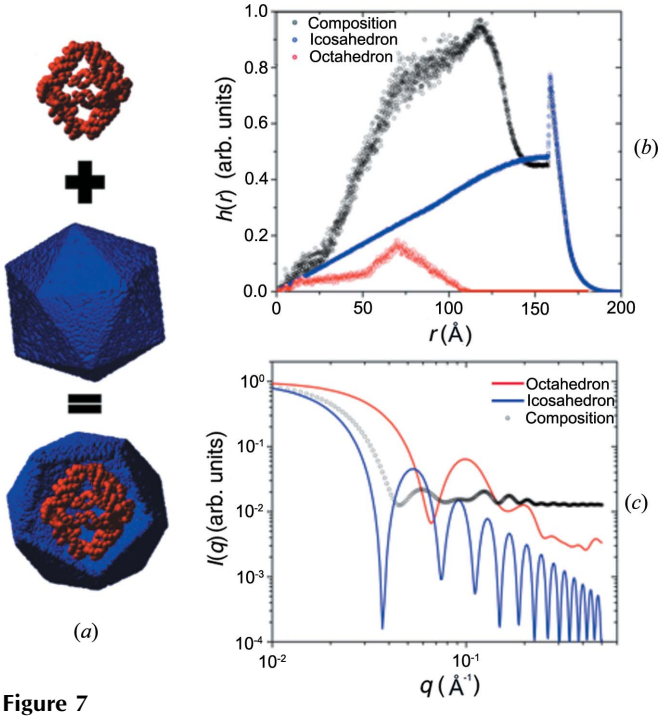

Figure 7
Composition of a shell-like structure with a DNA cage in its interior. (*a*) Models assumed. (*b*) Histograms of pairs of distances for each model. (*c*) Calculated scattering intensities. The assumed sizes and relative electron densities for each object are given on the figure. The procedure enables an easy construction and calculation of the scattering intensity for the composed model.

3. Conclusions

A useful method for the fast construction of high-symmetry objects and the computation of scattering intensity is

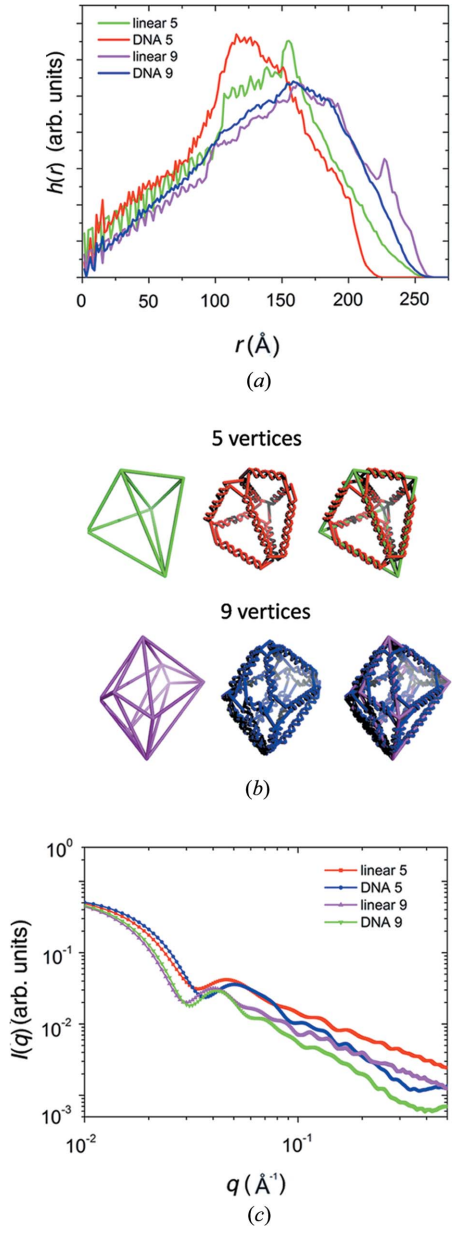

Figure 8
Example of the model construction based on a set of random points. (*a*) Histograms of pairs of distances for each model. (*b*) The obtained model for five and nine random points composed of linear arrays of spheres and DNA-like strands, respectively. The superposition of such models is just for comparison. The points were randomly generated on a surface of a sphere and the GW algorithm was used to build the object, as described in §2.3. (*c*) Calculated scattering intensities.

presented. Starting from a set of vertex positions, different types of models, such as solids, shell-like, edge-like and DNA-like models, can be built. Also, as presented in the results, the methods can be combined, permitting a versatile toolbox for the calculation of the small-angle scattering from complicated models.

By the use of a large and expandable database of shapes, it is possible to have access to a large number of shapes of regular polyhedra. Also, by the use of the GW method, it is possible to build polyhedrons from any collection of points defining a convex solid.

A specific procedure was developed permitting the construction of DNA-like structures for any geometry. This procedure fills an existing gap in the literature where several DNA cage-like structures have already been assembled but the major experimental technique used to investigate the structure has been that of microscopy. As demonstrated in several works (Andersen *et al.*, 2008, 2009; Oliveira *et al.*, 2010), the use of scattering methods permits a direct access to the ensemble-average structure and, therefore, also gives information on the assembly efficiency.

The possibility of combining several types of models, including also models with different electron densities for each component with a low number of parameters describing the system, provides a powerful tool for modelling and analysing scattering data.

The procedure is implemented in a program called *POLYGEN*, which is available upon request to the corresponding author.

CLPO was supported by FAPESP, CNPq and INCT(Fcx). CA was supported by CAPES, CNPq and INCT(Fcx).

References

Aldaye, F. A. & Sleiman, H. F. (2007). *J. Am. Chem. Soc.* **129**, 10070–10071.

Andersen, E. S., Dong. M, Nielsen, M. M., Jahn, K., Subramani, R., Mamdouh, W., Golas, M. M., Sander, B., Stark, H., Oliveira, C. L., Pedersen, J. S., Birkedal, V., Besenbacher. F., Gothelf, K. V. & Kjems, J. (2009). *Nature*, **459**, 73–76.

Andersen, F. F. *et al.* (2008). *Nucleic Acids Res.* **36**, 1113–1119.

Bhatia, D., Mehtab, S., Krishnan, R., Indi, S., Basu, A. & Krishnan, Y. (2009). *Angew. Chem.* **48**, 4134–4137.

Bou Sleiman, H., Ritacco, L. E., Aponte-Tinao, L., Muscolo, D. L., Nolte, L. P. & Reyes, M. (2011). *Ann. Biomed. Eng.* **39**, 1720–1727.

Coxeter, H. S. M. (1973). *Regular Polytopes*, 3rd ed. New York: Dover Publications.

Chacón, P., Morán, F., Díaz, J. F., Pantos, E. & Andreu, J. M. (1998). *Biophys. J.* **74**, 2760–2775.

Chand, D. R. & Kapur, S. S. (1970). *J. Assoc. Comput. Mach.* **17**, 78–86.

Chen, J. H. & Seeman, N. C. (1991). *Nature*, **350**, 631–633.

Debye, P. (1915). *Ann. Phys.* **46**, 809–823.

Dietz, H., Douglas, S. M. & Shih, W. M. (2009). *Science*, **325**, 725–730.

Douglas, S. M., Dietz, H., Liedl, T., Hoegberg, B., Graf, F. & Shih, W. M. (2009). *Nature*, **459**, 414–418.

Endo, M., Hidaka, K., Kato, T., Namba, K. & Sugiyama, H. (2009). *J. Am. Chem. Soc.* **131**, 15570–15571.

Erben, C. M., Goodman, R. P. & Turberfield, A. J. (2006). *Angew. Chem.* **45**, 7414–7417.

Erben, C. M., Goodman, R. P. & Turberfield, A. J. (2007). *J. Am. Chem. Soc.* **129**, 6992–6993.

Evrard, G., Mareuil, F., Bontems, F., Sizun, C. & Perez, J. (2011). *J. Appl. Cryst.* **44**, 1264–1271.

Falconi, M., Oteri, F., Chillemi, G., Andersen, F., Tordrup, D., Oliveira, C., Pedersen, J., Knudsen, B. & Desideri, A. (2009). *ACS Nano*, **3**, 1813–1822.

Feigin, L. & Svergun, D. (1987). *Structure Analysis by Small-Angle X-ray and Neutron Scattering.* New York: Plenum Press.

Glatter, O. (1972). *Acta Phys. Aust.* **36**, 307–315.

Glatter, O. (1977). *J. Appl. Cryst.* **10**, 415–421.

Glatter, O. & Kratky, O. (1982). *Small Angle X-ray Scattering.* London: Academic Press.

Goodman, R. P., Schaap, I. A. T., Tardin, C. F., Erben, C. M., Berry, R. M., Schmidt, C. F. & Turberfield, A. J. (2005). *Science*, **310**, 1661–1665.

Hamilton, T. D. & MacGillivray, L. R. (2004). *Cryst. Growth Des.* **4**, 419–430.

Han, D., Pal, S., Nangreave, J., Deng, Z., Liu, Y. & Yan, H. (2011). *Science*, **332**, 342–346.

Hansen, S. (1990). *J. Appl. Cryst.* **23**, 344–346.

He, Y., Su, M., Fang, P.-A., Zhang, C., Ribbe, A. E., Jiang, W. & Mao, C. (2010). *Angew. Chem. Int. Ed. Engl.* **49**, 748–751.

He, Y., Ye, T., Su, M., Zhang, C., Ribbe, A. E., Jiang, W. & Mao, C. (2008). *Nature*, **452**, 198–201.

Jarvis, R. A. (1973). *Inf. Process. Lett.* **2**, 18–21.

Jones, R. A. L. (2002). *Soft Condensed Matter.* Oxford University Press.

Kato, T., Goodman, R. P., Erben, C. M., Turberfield, A. J. & Namba, K. (2009). *Nano Lett.* **9**, 2747–2750.

Ke, Y., Sharma, J., Liu, M., Jahn, K., Liu, Y. & Yan, H. (2009). *Nano Lett.* **9**, 2445–2447.

Koradi, R., Billeter, M. & Wüthrich, K. (1996). *J. Mol. Graph.* **14**, 51–55.

Li, Z., Wei, B., Nangreave, J., Lin, C., Liu, Y., Mi, Y. & Yan, H. (2009). *J. Am. Chem. Soc.* **131**, 13093–13098.

Oliveira, C., Juul, S., Jorgensen, H., Knudsen, B., Tordrup, D., Oteri, F., Falconi, M., Koch, J., Desideri, A., Pedersen, J., Andersen, F. & Knudsen, B. (2010). *ACS Nano*, **4**, 1367–1376.

Oteri, F., Falconi, M., Chillemi, G., Andersen, F. F., Oliveira, C. L. P., Pedersen, J. S., Knudsen, B. R. & Desideri, A. (2011). *J. Phys. Chem. C*, **115**, 16819–16827.

Pantos, E. & Bordas, J. (1994). *Pure Appl. Chem.* **66**, 77–82.

Pedersen, J. S. (1997). *Adv. Colloid Interface Sci.* **70**, 171–210.

Pedersen, J. S., Oliveira, C. L., Hübschmann, H. B., Arleth, L., Manniche, S., Kirkby, N. & Nielsen, H. M. (2012). *Biophys. J.* **102**, 2372–2380.

Perry, J. J., Perman, J. A. & Zaworotko, M. J. (2009). *Chem. Soc. Rev.* **38**, 1400–1417.

Petoukhov, M. V. & Svergun, D. I. (2005). *Biophys. J.* **89**, 1237–1250.

Petoukhov, M. V. & Svergun, D. I. (2013). *Int. J. Biochem. Cell Biol.* **45**, 429–437.

Pettersen, E. F., Goddard, T. D., Huang, C. C., Couch, G. S., Greenblatt, D. M., Meng, E. C. & Ferrin, T. E. (2004). *J. Comput. Chem.* **25**, 1605–1612.

Preparata, F. P. & Shamos, M. I. (1985). *Computational Geometry: an Introduction.* New York: Springer-Verlag.

Press, W. H. (1992). *Numerical Recipes in FORTRAN: the Art of Scientific Computing.* Cambridge University Press.

Rothemund, P. W. K. (2006). *Nature*, **440**, 297–302.

Seeman, N. C. (2010). *Ann. Rev. Biochem.* **79**, 65–87.

Seidel, S. R. & Stang, P. J. (2002). *Acc. Chem. Res.* **35**, 972–983.

Semenyuk, A. V. & Svergun, D. I. (1991). *J. Appl. Cryst.* **24**, 537–540.

Shih, W. M., Quispe, J. D. & Joyce, G. F. (2004). *Nature*, **427**, 618–621.

Stuhrmann, H. B. (1970). *Acta Cryst.* A**26**, 297–306.

Svergun, D. I. (1999). *Biophys. J.* **76**, 2879–2886.

Svergun, D. I. & Stuhrmann, H. B. (1991). *Acta Cryst.* A**47**, 736–744.

Svergun, D. I., Volkov, V. V., Kozin, M. B. & Stuhrmann, H. B. (1996). *Acta Cryst.* A**52**, 419–426.
Uchida, M., Klem, M., Allen, M., Suci, P., Flenniken, M., Gillitzer, E., Varpness, Z., Liepold, L., Young, M. & Douglas, T. (2007). *Adv. Mater.* **19**, 1025–1042.
Volkov, V. V. & Svergun, D. I. (2003). *J. Appl. Cryst.* **36**, 860–864.
Walsh, A. S., Yin, H., Erben, C. M., Wood, M. J. A. & Turberfield, A. J. (2011). *ACS Nano*, **5**, 5427–5432.
Walther, D., Cohen, F. E. & Doniach, S. (2000). *J. Appl. Cryst.* **33**, 350–363.
Wassenaar, T. A. *et al.* (2012). *J. Grid Comput.* **10**, 743–767.
Yang, H., McLaughlin, C. K., Aldaye, F. A., Hamblin, G. D., Rys, A. Z., Rouiller, I. & Sleiman, H. F. (2009). *Nat. Chem.* **1**, 390–396.
Zhang, C., Su, M., He, Y., Zhao, X., Fang, P. A., Ribbe, A. E., Jiang, W. & Mao, C. (2008). *Proc. Natl Acad. Sci. USA*, **105**, 10665–10669.
Zhang, Y. W. & Seeman, N. C. (1994). *J. Am. Chem. Soc.* **116**, 1661–1669.
Zhou, J., Deyhim, A., Krueger, S. & Gregurick, S. (2005). *Comput. Phys. Commun.* **170**, 186–204.
Zimmermann, J., Cebulla, M. R. J., Moenninghoff, S. & Kiedrowski, G. von (2008). *Angew. Chem. Int. Ed. Engl.* **47**, 3626–3630.