JOURNAL OF APPLIED CRYSTALLOGRAPHY

ISSN: 1600-5767
journals.iucr.org/j

A new algorithm for the reconstruction of protein molecular envelopes from X-ray solution scattering data

John Badger

J. Appl. Cryst. (2019). **52**, 937–944

JOURNAL OF APPLIED CRYSTALLOGRAPHY
ISSN 1600-5767

A new algorithm for the reconstruction of protein molecular envelopes from X-ray solution scattering data

John Badger*

DeltaG Technologies, 4360 Benhurst Avenue, San Diego, CA 92122, USA. *Correspondence e-mail: info1.dgtech@gmail.com

Received 22 April 2019
Accepted 8 July 2019

Edited by D. I. Svergun, European Molecular Biology Laboratory, Hamburg, Germany

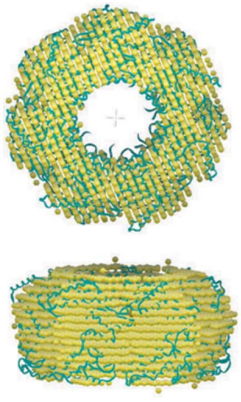
Keywords: X-ray solution scattering; protein shape determination; molecular envelopes.

At sufficiently low resolution, the scattering density within the volume occupied by a well folded protein molecule appears relatively flat. By enforcing this condition, three-dimensional protein molecular envelopes may be reconstructed using information obtained from X-ray solution scattering profiles. A practical approach for solving the low-resolution structures of protein molecules from solution scattering data involves modelling the protein shape using a set of volume-filling points ('beads') and transforming the scattering data to a more convenient target, the pair distance distribution function, $P(r)$. Using algorithms described here, the beads interact via a modified Lennard–Jones potential and their positions are adjusted and confined until they fit the expected protein volume and agreement with $P(r)$ is obtained. This methodology allows the protein volume to be modelled by an arbitrary, user-defined number of beads, enabling the rapid reconstruction of protein structures of widely varying sizes. Tests carried out with a variety of synthetic and experimental data sets show that this approach gives efficient and reliable determinations of protein molecular envelopes.

1. Introduction

Over the last two decades X-ray solution scattering has become a mainstream research method employed by many research groups to obtain structural information on protein molecules. With data modelling and analysis techniques now available it is often possible to obtain far more information from X-ray solution scattering experiments than basic structural parameters such as radii of gyration, mass and volume [for modern approaches to the determination of these parameters, see Rambo & Tainer (2013)]. Despite the one-dimensional nature of the X-ray scattering data, unique three-dimensional molecular envelopes with arbitrary shapes may be recovered using *ab initio* reconstruction techniques (Chacón *et al.*, 1998; Svergun, 1999; Walther *et al.*, 2000; Svergun *et al.*, 2001).

The key observation that underpins computational methodologies for the reconstruction of protein envelopes from solution scattering data is that, at sufficiently low resolution (up to ∼25 Å), the protein density within the molecular envelope is relatively flat. Scattering effects arising from inhomogeneities in the atom number density due to secondary structural elements only start to become significant at higher resolutions [a cogent discussion of these resolution regimes is given by Svergun *et al.* (2001)]. Algorithms for determining protein molecular envelopes typically represent the protein volume with a set of points ('beads'), the positions of which are adjusted to fit either the scattered intensity distribution or the pair distance distribution, $P(r)$, obtained from a transform



of the data (Svergun, 1992). Physically plausible solutions are obtained when the volume-defining beads are required to cluster into contiguous volumes with relatively uniform packing density. Since the number of beads used in these models is typically relatively large, and no convenient analytical optimization technique is available, structure solutions are obtained using Monte Carlo or other methods (genetic algorithms) that employ random searches. A recently developed alternative approach adapts the iterative image reconstruction method to the determination of electron densities representing the protein volume (Grant, 2018).

One tactic for rapid and practical calculations is to deploy the beads on a uniform grid and optimize a scoring function that encourages clustering of occupied grid points (Chacón *et al.*, 1998; Svergun, 1999; Walther *et al.*, 2000). A disadvantage of this approach is that, because of the regularity of the grid and the binary nature of the solutions (grid points are either occupied or not), the representation of deviation from an almost completely uniform scattering density is curtailed and only intensity data in the lowest resolution regime ($q < \sim 0.2$ Å^{-1}) are adequately fitted. A strategy for using grid-based programs is to truncate the intensity data at a resolution limit where the model is unable to express non-uniformities in the protein scattering density. If possible, incorporation of data to somewhat higher resolution, where scattering effects resulting from the overall shape of the protein still dominate but become modulated by density inhomogeneities due to secondary structural elements, should lead to more accurate and better-defined structure solutions. In the absence of a grid representation, restraints between arbitrarily placed beads are needed, which will cause them to cluster together with a relatively uniform packing density. Since the number of beads is large compared with the information content in the scattering data, overly loose restraints will cause an under-determined optimization problem while very rigid restraints will result in inadequate modelling of the higher-resolution portion of the data. An ingenious basis for establishing an appropriate set of inter-bead restraints is to consider the beads to be 'dummy amino acids' and employ interaction terms that cause them to connect in chains and cluster to mimic Cα-to-Cα distributions observed in experimentally determined protein structures (Svergun *et al.*, 2001).

In this paper, an algorithm that uses a modified Lennard–Jones potential as an energy term between the beads that represent the protein volume is described. Essentially, the beads act as 'sticky hard spheres' which, when optimally packed, fill the anticipated protein volume. Tests using synthetic and experimentally derived $P(r)$ functions as input data show that a computer program that incorporates this approach is fast enough for routine use on both small and large proteins and gives consistent and accurate structure solutions.

2. Methods

2.1. Target function

The operation of the new program, *SHAPES*, is based on the idea that the pair distance distribution function $P(r)$, derived from solution scattering data, may be fitted by a histogram $H(r)$ of distances between a set of beads. Upon convergence of the optimization process, the beads should be arranged with relatively uniform density within the expected volume of the target protein. The beads are driven to cluster with uniform density by an inter-bead interaction energy $V(r)$ that depends on the distance r between them, and the complete target function for structure determination is

$$\text{Score} = \left\{\sum [P(r) - H(r)]^2 / N\right\}^{1/2} + kV(r), \qquad (1)$$

where the summation extends over the N data points in the input $P(r)$ that are greater than the ideal model bead–bead separation, r_0. The scale factor k is set so that shifts in the agreement between $P(r)$ and $H(r)$ and the changes in the inter-particle interaction energy $V(r)$ remain almost equal as the bead positions change during the structure determination process. This choice of weight ensures that the bead positions are adjusted to fit $P(r)$ and minimize the inter-bead energy throughout the optimization process. In tests, equal weighting of these two terms gave slightly better results than more asymmetric choices. Other than the scale factor k, no additional weighting scheme was used.

Since the final congealed mass of beads at the conclusion of the optimization is required to occupy the target protein volume, the inter-bead interaction term should become sufficiently repulsive at distances much shorter than r_0 to prevent compression of the volume. Conversely, attractive interactions between beads separated by more than a few r_0 should approach zero to avoid biasing the final structure to spherical shapes. These considerations are taken into account in the algorithm by having the set of beads interact via a Lennard–Jones 6–12 potential where the region around the minimum is truncated to a flat-bottomed basin:

$$V(r) = \text{Max}\left[(r_0/r)^{12} - 2(r_0/r)^6, -0.291\right]. \qquad (2)$$

For Lennard–Jones potentials the repulsive force increases very steeply for beads closer to each other than the optimal contact distance r_0, and the attractive region decays to almost zero for beads separated by distances greater than $\sim 3r_0$. The optimal contact distance is set to the value for a close-packed array of beads filling the partial specific volume of the target protein. Truncating the Lennard–Jones energy minima (-1 energy unit) to a flat-bottomed basin (-0.291 energy units) allows some freedom in the local bead density at no cost in inter-bead energy. For the case where the number of beads is set to be equal to the number of amino acids in the structure, the target bead–bead separation is $r_0 = 5.6$ Å and the basin extends the lowest energy region to larger r by ~ 2 Å and slightly inwards into the steeply repulsive region. Interactions between beads that are separated by distances greater than $3r_0$ are neglected. The numerical parameters used in a program run are automatically set depending on the number of beads chosen to model the protein volume.

2.2. Optimization procedure

The shape reconstruction process is initiated from a collection of beads occupying a space that is about 15% larger than twice the expected partial specific volume of the target protein. Protein partial specific volumes are estimated from the number of amino acids entered into the program, assuming 110 Da per amino acid and using 1.21 as a conversion factor between molecular weight and partial specific volume. A relatively smooth distribution of starting positions is obtained by first randomly placing beads within an approximately spherical volume with maximum bead separation, $D_{\max}$, equal to the maximum separation specified by the input $P(r)$. To prevent the elimination of highly unfavourable short contacts ($r < r_0$) between beads from dominating the start of the optimization process, the bead positions are adjusted by applying small shifts that eliminate overly close contacts. For disk- or rod-shaped protein assemblies that contain high rotational symmetries, the reconstruction process may sometimes become trapped by a misalignment of the symmetry axis with the correct shape. These unwanted solutions may be avoided by initiating the reconstruction process from slightly flattened or extended bead distributions. In practice, the overall shape of symmetric assemblies may be known from structural data *a priori*, detected from reconstruction trials in the absence of the symmetry restraint or inferred from unphysically disconnected regions in the incorrect structure solutions.

The optimization of bead positions is driven by a Monte Carlo process in which a random shift may move a bead to any position within an allowed protein volume. Changes in bead positions are accepted according to standard criteria for these types of optimization problem; if repositioning a bead improves the score for the system or raises the score by an increment that is less than a value drawn from a random Boltzmann distribution then the move is accepted. Parameters are adjusted so that at least 10% of trial moves are accepted throughout the Monte Carlo process. In this way, the beads remain 'on the boil' and premature convergence is prevented. Since most of the allowed volume is occluded by beads and the Lennard–Jones potential is highly unfavourable for bead separations much closer than r_0, the acceptance rate is necessarily quite small.

A key driver for convergence of the optimization process is a progressive restriction in the bead-occupied volume from $\sim$2.30$\times$ the target partial specific volume to $\sim$1.15$\times$ the target partial specific volume. This concept is analogous to the 'shrink wrap' approach (Marchesini *et al.*, 2003) commonly used to reconstruct real-space objects from Fourier transform data by iterative methods. The allowed volume is defined by a grid on which local bead densities are calculated in spheres with radii $\sim 1.8 r_0$, and a density threshold is determined so as to give the appropriate number of grid points for the required volume. Beads may only move to positions that are in proximity to the allowed grid but they are not confined to the grid points. The program is currently set for 140 volume contraction cycles within which each cycle includes ten trial moves per bead. A final set of 20 optimization cycles are run at a fixed final volume.

2.3. Program operation

The required inputs for program operation are (i) the pair distance distribution function $P(r)$, (ii) the expected number of amino acids in the structure and (iii) the number of beads that will be used to represent the protein volume. For most reconstruction problems it is appropriate to set the number of beads to be approximately equal to the number of amino acids in the protein; however, the reconstruction algorithm is not specifically parameterized for this equivalence. More robust convergence might sometimes be obtained for small proteins (<300 amino acids) by using a larger number of beads, and for large proteins (>1000 amino acids), computer time may be reduced by modelling the shape with a smaller number of beads than amino acids. If a point rotational symmetry is anticipated (*i.e.* for oligomeric structures), the symmetry may be applied and calculation times are reduced by the elimination of redundant operations. For convenience, the program may be set to perform multiple reconstruction runs.

The key outputs from the program are pseudo-PDB-format files that contain (i) the final distribution of beads and (ii) a space-filling array represented by positions on a 5 Å rectangular grid that fills the protein partial specific volume. The space-filling array is generated from the final local average density of beads in the same way as the determination of the allowed volume during the optimization process.

2.4. Analysis software and hardware

Programs from the *ATSAS 2.7.2* program suite (Franke *et al.*, 2017) were used for calculations related to the analysis of solution scattering data. Specifically, *CRYSOL 2.8.3* (Svergun *et al.*, 1995) was used to generate scattering data from atomic models and *GNOM 4.6* (Svergun, 1992) was used to calculate pair distance distributions from intensity profiles. In order to compare the reconstructions obtained with *SHAPES* with reconstructions obtained with other widely used software, reconstruction trials were also performed with *GASBOR 2.2* (real-space version; Svergun *et al.*, 2001), *DAMMIN 5.3* (Svergun, 1999) and *DAMMIF* (Franke & Svergun, 2009).

Overlays and numerical comparisons of sets of beads, molecular envelopes and atomic models were made using the *SUPCOMB* program as incorporated into *DAMSEL 5.0* (Kozin & Svergun, 2001). In brief, the *SUPCOMB* algorithm finds the optimal alignment of two sets of points as scored by the normalized spatial discrepancy (NSD). This measure combines the mean of the minimum distance from each point in set 1 to points in set 2 and, the reciprocal measure, the mean of the minimum distance from each point in set 2 to points in set 1. The two scores are normalized by the mean minimum separations of points within the sets and are added together. Using the NSD it is possible to compare structures defined by different numbers of points and at different resolutions (*i.e.* a low-resolution molecular envelope described by relatively few points can be compared with an atomic model). NSD scores less than unity represent objects that are similar to each other and, conversely, NSD scores much greater than unity indicate dissimilar objects.

Atomic models and outputs from reconstruction trials were visualized using the *MIFit* molecular graphics program (http://code.google.com/p/mifit).

The *SHAPES* program was initially developed using the Python programming language (version 2.6.5) and subsequently adapted for compatibility with the Python 3 standard. The program does not utilize any non-standard modules. All calculations reported here were performed on a Dell Inspiron 3542 laptop computer running an i5 1.70 GHz processor on the Windows 7 operating system.

2.5. Test data

The performance of the *SHAPES* program was evaluated using both simulated intensities calculated from atomic models obtained from the Protein Data Bank (PDB; Berman *et al.*, 2000) and publicly available experimental data from the Small Angle Scattering Biological Data Bank (SASBDB; Valentini *et al.*, 2015).

Simulated data sets were calculated using the default settings of the program *CRYSOL* from a variety of protein structures obtained from the PDB, *i.e.* the resulting intensities in these synthetic data sets include a contribution from modelling a surface solvent layer and are not just the result of *in vacuo* scattering from model protein atoms. Pair distance distribution functions were calculated from these intensities using the program *GNOM*. Test cases with simulated data were selected to include a variety of protein sizes, shapes and symmetries. Examples shown here are the inactive form of the c-Abl tyrosine kinase (PDB code 2fo0; Nagar *et al.*, 2006), an oligomeric form of ArnA (PDB entry 4wkg; Fischer *et al.*, 2015), a single disc of the chaperonin GroEL (PDB entry 1ssb; deMel *et al.*, 1994), a complex of KRAS4b with PDE delta (PDB entry 5tar; Dharmaiah *et al.*, 2016), lysozyme (PDB entry 253l; Shoichet *et al.*, 1995) and c-Src in an open conformation (PDB entry 1y57; Cowan-Jacob *et al.*, 2005) (Table 1).

The examples taken from the SASBDB were the first five data sets that appeared in the search interface on 8 November 2018 for which both a solution scattering reconstruction and an associated model were reported. The accompanying $P(r)$ files were taken from the SASBDB and used as input for the reconstruction program. These test cases include Kif2A-bound tandem tubulin heterodimers (SASDCR9; Trofimova *et al.*, 2018), fluorescence recovery protein dimer (SASDD42; Slonimskiy *et al.*, 2018), phox homologue C2 domains of human phosphatidylinositol 4-phosphate 3-kinase C2 domain containing subunit alpha monomer (SASDD76; Chen *et al.*, 2018), VNG0258H/RosR dimer (SASDD89; Kutnowski *et al.*, 2018) and the 2:1 complex of a fluorescence recovery protein dimer with orange carotenoid-binding protein monomer (SASDDG9; Sluchanko *et al.*, 2018). The amino acid counts used by the program were calculated from the quoted molecular weights, assuming 110 Da per amino acid.

Table 1
Synthetic data sets and parameters for protein reconstruction tests.

The total number of amino acids and the rotational symmetry are given for each PDB entry used to generate data. The data over range q were calculated using *CRYSOL* and the pair distance distribution function extending to D_{max} was calculated with *GNOM*.

PDB code	No. of amino acids	Symmetry	q (Å^{-1})	D_{max} (Å)
2fo0	465	1	0.005–0.500	80
4wkg	3837	3	0.010–0.300	180
1ssb	3668	7	0.010–0.500	145
5tar	332	1	0.003–0.500	100
253l	164	1	0.010–0.500	55
1y57	452	1	0.005–0.500	110

3. Results and discussion

3.1. Reconstructions of protein molecular volumes using *SHAPES*

The proof of concept and limitations of these algorithms were first explored using simulated data calculated from protein coordinate sets obtained from the PDB. The examples cover a variety of protein sizes, shapes and symmetries (Table 1). To quantify the reproducibility of the structure solutions obtained with *SHAPES*, ten replicate runs were performed for each example. Pairwise superpositions of the space-filling arrays obtained for each set of replicates gave mean NSD values of ∼0.6, *i.e.* the replicate reconstructed volumes have very similar shapes on the length scale of the

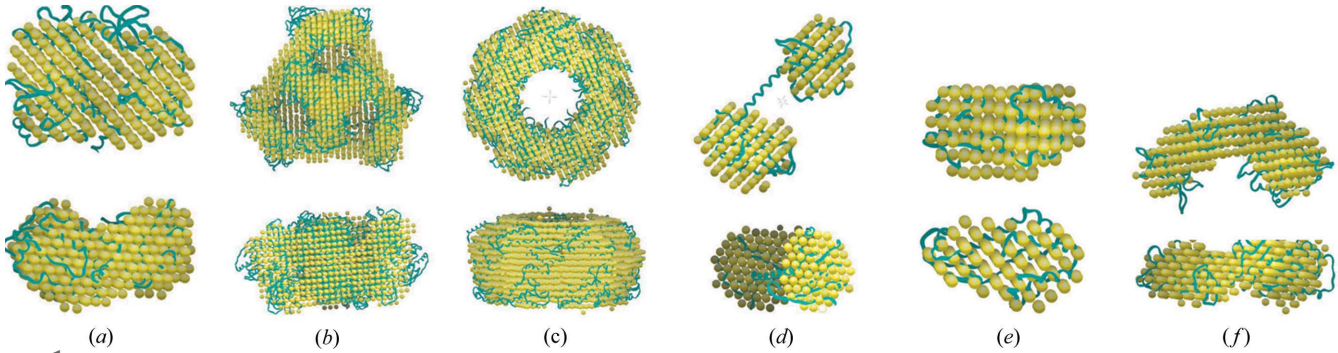


Figure 1
Representative volume-filling arrays (yellow dots) obtained from reconstructions carried out with calculated data superimposed on the polypeptide chain traces (blue tubes) for the target proteins. Two orthogonal views are displayed for each example. PDB IDs: (*a*) 2fo0, (*b*) 4wkg, (*c*) 1ssb, (*d*) 5tar, (*e*) 253l, (*f*) 1y57.

Table 2
Statistics for protein reconstructions obtained with *SHAPES* from synthetic data.

Listed for each set of replicate runs are the number of beads used to represent the protein volume, the minimum and maximum values for $\Delta P(r) = \sum|P(r) - H(r)|/\sum P(r)$, the maximum and minimum Lennard–Jones energies per bead (average energy), the NSD value for pairwise comparison between volume-defining arrays (NSD-vols), the NSD value for pairwise comparison between sets of beads (NSD-beads), and the discrepancy between the most representative set of beads and the Cα positions in the input model (NSD).

PDB entry	No. of beads	$\Delta P(r)$	Average energy	NSD-vols	NSD-beads	NSD
2fo0	465	0.019, 0.027	−1.40, −1.52	0.597	0.716	1.029
4wkg	3960	0.009, 0.013	−1.46, −1.52	0.603	0.714	1.240
1ss8	3780	0.014, 0.016	−1.41, −1.55	0.543	0.681	1.130
5tar	332	0.031, 0.035	−1.32, −1.46	0.567	0.710	1.059
253l	298	0.027, 0.036	−1.38, −1.51	0.504	0.692	0.944
1y57	452	0.022, 0.027	−1.39, −1.46	0.796	0.860	1.064

5 Å grids used to map the protein shapes (Table 2). Fig. 1 compares the most representative space-filling array obtained from each of these examples with the chain-trace of the target protein. In all cases there is strong visual correspondence between the shape of the target protein model and the shape derived from the simulated data. For all the reconstruction trials with synthetic data the number of beads used to represent the protein volume was approximately equal to the number of amino acids in the structure. Comparisons of the most representative constellations of beads (typical nearest neighbour separations ∼5.3 Å) with the Cα positions (separated by 3.8 Å) in the target models gave NSD values of approximately unity, suggesting that the accuracy of the reconstructed shapes may approach the dimensions of an amino acid.

Further tests of the program were performed using experimental data and files obtained from the SASBDB (Table 3). In all of these examples the arrays used to demark the reconstructed protein volumes were found to be relatively similar in replicated reconstruction runs (mean NSD values ∼0.7, Table 4) and, visually, the representative volumes were consistent with the atomic models associated with these data (Fig. 2). Values of the χ^2 statistic obtained from a comparison of discrepancies between observed and calculated intensities with experimental errors indicate that the final bead models typically fit the data to the expected error [the standard deviations associated with intensities for data for D76 may be significantly underestimated since reconstructions with *GASBOR* and *DAMMIN*/*DAMMIF* (*cf.* 5.2) also give very low values]. Additional diagnostic information is provided in the form of plots comparing $P(r)$ with $H(r)$ (Fig. 3) and observed intensities with intensities calculated from the final set of bead positions (Fig. 4). For most of these tests the number of beads used to represent the protein volume was set

Table 3
Experimental data sets and parameters for protein reconstruction tests.

The total number of amino acids in the structure and the rotational symmetry are given for each SASBDB entry from which data were obtained. The number of amino acids was estimated from the protein molecular weight. The data range (q) and the pair distance distribution function extending to D_{max} were obtained from the *GNOM* output files from the SASBDB.

SASBDB	No. of amino acids	Symmetry	q (Å^{-1})	D_{max} (Å)
CR9	2418	1	0.009–0.215	195
D42	209	2	0.013–0.387	105
D76	297	1	0.012–0.273	93
D89	263	2	0.027–0.311	75
DG9	564	1	0.011–0.264	130

Table 4
Statistics for protein shape reconstructions obtained with *SHAPES* from experimental data.

Listed for each set of replicate runs are the minimum and maximum values for $\Delta P(r) = \sum|P(r) - H(r)|/\sum P(r)$, the maximum and minimum Lennard–Jones energies per bead (average energy), the NSD value for pairwise comparison between volume-defining arrays (NSD-vols), the NSD value for pairwise comparison between sets of beads (NSD-beads), and the discrepancy between the most representative set of beads and the Cα positions in the input model (NSD). The χ^2 values compare the intensity data and standard deviations with intensities calculated for the most representative set of beads as point scattering centres.

SASBDB	$\Delta P(r)$	Average energy	NSD-vols	NSD-beads	NSD	χ^2
CR9	0.015, 0.024	−1.44, −1.50	0.839†	0.981†	1.317	1.4
D42	0.063, 0.073	−1.22, −1.33	0.653	0.804	1.227	1.3
D76	0.041, 0.051	−1.32, −1.40	0.598	0.750	1.166	0.2
D89	0.064, 0.075	−1.28, −1.44	0.556	0.776	1.214	2.4
DG9	0.038, 0.045	−1.39, −1.46	0.648	0.756	1.275	1.2

† Includes one extreme outlier reconstruction indicated by NSD-vols = 1.543, NSD-beads = 1.411 versus the other nine runs; after excluding this reconstruction NSD-vols = 0.663 and NSD-beads = 0.906.

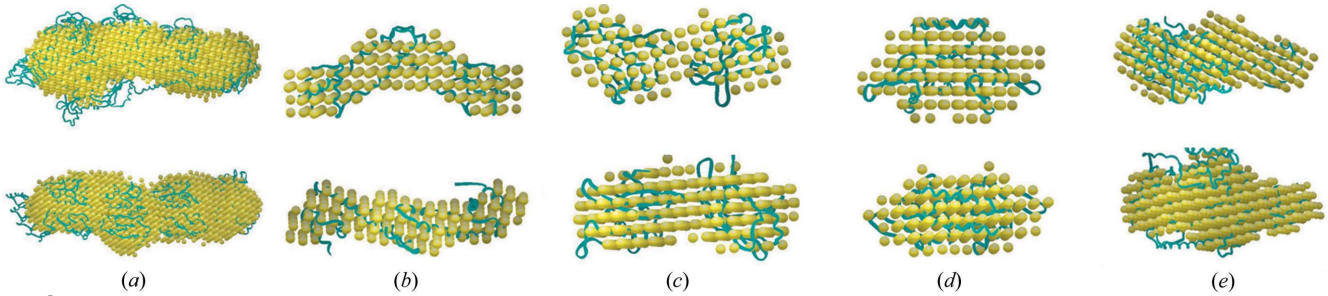


Figure 2
Representative volume-filling arrays (yellow dots) obtained from reconstructions carried out with experimental data superimposed on polypeptide chain traces (blue tubes) for the target proteins. Two orthogonal views are displayed for each example. SASBDB IDs: (*a*) CR9, (*b*) D42, (*c*) D76, (*d*) D89, (*e*) DG9.

to be approximately equal to the number of amino acids in the structure. Comparisons of the positions of beads in the most representative solution with the Cα positions in the target models gave NSD values of ∼1.25, slightly higher than for the tests using synthetic data.

All the reconstruction trials shown here resulted in relatively well defined outcomes and representative volumes that were consistent with the expected shapes of the protein targets (Figs. 1 and 2). Tests with synthetic and experimental data included examples for which q_{max} ranges from 0.215 to 0.500 (Tables 1 and 3), indicating that successful applications of the *SHAPES* program are not unduly predicated on the exact choice of upper resolution limit and that these data may extend somewhat beyond the regime where the scattering is almost completely dominated by the contribution from the overall molecular shape. Although data at the high-resolution limit may contribute relatively little to the fitting (Fig. 4), this component might still prove to be beneficial in terms of the robustness and accuracy of the calculation of $P(r)$ by indirect Fourier transform (Svergun, 1992).

Only a small number of reconstruction trials (perhaps 3–4) would seem to be required to detect and eliminate occasional outlier solutions. It is expected that in the majority of applications the most representative volume-defining array output by *SHAPES* may be used to depict the results of the solution scattering analysis and the common practice of averaging multiple disparate reconstruction runs to obtain a consensus reconstruction will not be necessary.

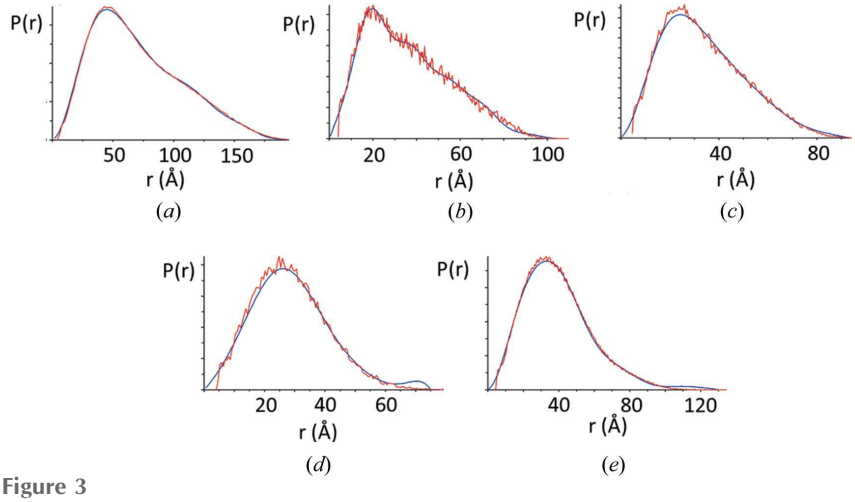


Figure 3
Comparisons of the pair distance distribution function, $P(r)$, computed from experimental intensity data (blue curve) with the set of pair distances, $H(r)$, for the set of beads that gave the most representative protein shape (red curve). The jitter in the model curve is caused by sampling the distribution of pair distances with a relatively small number of beads and the fact that this function is shown in 'pure' form for point distances, without the smoothing that would result from convolution with a form factor representing a finite size for the bead volume elements. SASBDB IDs: (*a*) CR9, (*b*) D42, (*c*) D76, (*d*) D89, (*e*) DG9.

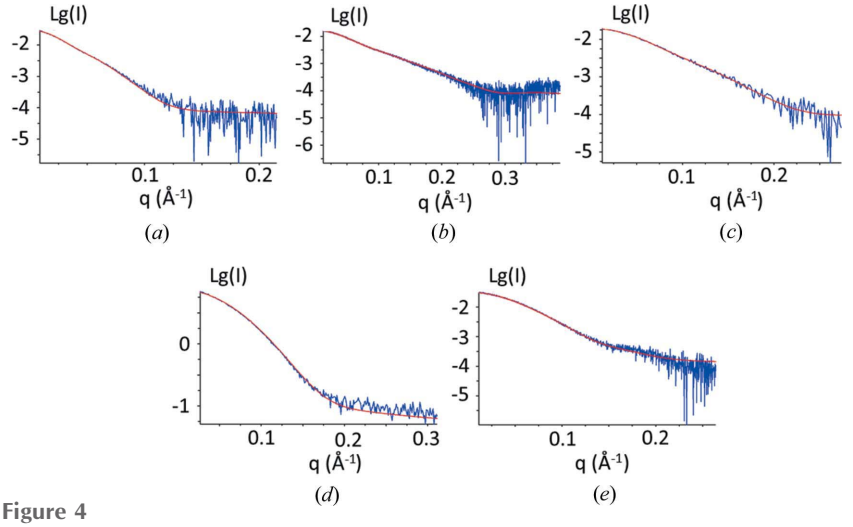


Figure 4
Comparisons of experimentally determined intensities (blue curve) with intensities calculated from the set of beads that gave the most representative protein shape (red curve). R values between the observed and calculated intensities were all ∼0.02. SASBDB IDs: (*a*) CR9, (*b*) D42, (*c*) D76, (*d*) D89, (*e*) DG9.

3.2. Comparison of *SHAPES* with *GASBOR* and *DAMMIN*/*DAMMIF*

To compare the behaviour and performance of *SHAPES* with those of the most widely used software for *ab initio* protein envelope reconstruction, parallel reconstruction trials using the experimental data from the SASBDB (Table 3) were run using *GASBOR* (Svergun *et al.*, 2001) as well as *DAMMIN* (Svergun, 1999) and its CPU-efficient reimplementation *DAMMIF* (Franke & Svergun, 2009). Since *GASBOR* was parameterized by considering the volume-defining beads to be dummy amino acids, the number of beads was set equal to the expected number of amino acids in the target structure. The program version that utilizes the real-space target $P(r)$ was used, known symmetries were applied and all other parameters were set to default values. *DAMMIN*/*DAMMIF* represent the protein volume using beads placed on a fixed hexagonal grid with spacing determined by the program. The intensity-based target was used, known symmetries were applied and all other parameters were set to default values for 'fast' modes. Both *GASBOR* and *DAMMIN* were accessed through the GUIs provided with the *ATSAS* installation, and *DAMMIF* was accessed via the command line. All calculations were run on the same Windows laptop computer used for the calculations with *SHAPES* (*cf.* Section 2.4).

Direct comparisons of the absolute reproducibility and accuracy of the reconstructed protein volumes between different programs are greatly complicated by the fact that their operating principles differ and the optimal numbers of beads used to represent the

Table 5
Statistics for protein shape reconstructions from experimental data obtained with *GASBOR*, *DAMMIN* and *DAMMIF*.

Listed are the NSD values for pairwise comparisons between beads (NSD-beads) and discrepancies between the most representative set of beads and the Cα positions in the associated model (NSD). The χ^2 values are given for the most representative reconstruction as reported by *GASBOR*, *DAMMIN* and *DAMMIF*.

	GASBOR			*DAMMIN*			*DAMMIF*		
SASBDB	NSD-beads	NSD	χ^2	NSD-beads	NSD	χ^2	NSD-beads	NSD	χ^2
CR9	1.417	1.489	1.1	0.537	1.124	1.0	0.582	1.133	1.0
D42	1.281	1.248	1.2	0.693	1.079	1.1	0.633	1.055	1.2
D76	1.173	1.211	0.3	0.589	1.104	0.2	0.588	1.084	0.2
D89	1.148	1.284	1.6	0.541	1.258	1.0	0.555	1.220	1.1
DG9	1.303	1.238	1.3	0.579	1.142	1.0	0.526	1.158	1.0

Table 6
Comparison of output representations and runtimes for reconstruction calculations carried out with *SHAPES*, *GASBOR*, *DAMMIN* and *DAMMIF*.

Listed are the numbers of volume-representing beads in reconstruction runs with *SHAPES*, *GASBOR*, *DAMMIN* and *DAMMIF*. Wall clock times per reconstruction run are given in minutes as the average of replicate runs on an otherwise quiet computer system.

	SHAPES		*GASBOR*		*DAMMIN*		*DAMMIF*	
SASBDB	No. of beads	Time	No. of beads	Time	No. of beads	Time	No. of beads	Time
CR9	1209	76.7	2418	215.8	270	22.1	260	1.8
D42	298	2.8	210	11.0	319	29.1	180	1.2
D76	297	4.4	297	15.2	282	19.7	291	1.7
D89	292	2.3	264	21.0	960	36.5	632	1.3
DG9	564	16.5	564	29.4	207	19.4	227	1.9

protein volumes and the typical minimum distances between neighbouring beads vary. Specifically, *GASBOR* assembles beads into chains of beads separated by the characteristic Cα–Cα distance of 3.8 Å, whereas for *DAMMIN*/*DAMMIF* the number of beads and grid spacings vary considerably but typically result in much coarser-grained models (14.0/14.0, 6.0/7.5, 6.8/6.8, 4.2/5.2 and 9.4/9.4 Å for CR9, D42, D76, D89 and DG9, respectively). With *SHAPES*, the typical minimum bead separation is ∼5.3 Å when the number of beads is set equal to the number of amino acids, the case that is most comparable to usage of *GASBOR*.

As measured by the mean NSD for pairwise comparisons of reconstruction replicates, the final distributions of beads obtained with the *SHAPES* program (Table 4) are more tightly clustered than those obtained with *GASBOR* (Table 5). A high level of reproducibility is also achieved with the reconstructions performed with *DAMMIN*/*DAMMIF*, albeit often on a much coarser grid. As a practical matter, an optimization method that provides more singular solutions is more efficient and convenient (fewer runs are needed to provide 'an answer'), but it should be noted that although these results are more precise they are not necessarily more accurate. A methodology for determining the effective resolution of solution scattering reconstructions based on the reproducibility of the resulting reconstructions has been proposed (Tuukkanen *et al.*, 2016), but this approach requires a program-specific calibration. Comparing bead positions for the most representative reconstruction with the Cα positions in the target model, all programs gave NSD values that were slightly greater than unity, suggesting relatively modest differences from the expected structures. However, this measure is influenced by the typical nearest neighbour separation between beads, and the structures obtained with *DAMMIN*/*DAMMIF* display less detail than reconstructions obtained with *SHAPES* and *GASBOR*.

Reconstructions with *SHAPES* for all the examples shown here (Figs. 1 and 2) proved to be successful and the number of aberrant (outlier) reconstruction runs for the most elongated structures did not usually exceed ∼10%. Protein molecules that are more highly elongated than these cases are more challenging and it is more likely that reconstruction runs will result in shapes that contain improperly disconnected protein volumes; the energetic term that causes beads to clump together becomes less valid for defining these types of structures. With *GASBOR*, this problem is alleviated by an additional bead modelling term that assembles the beads into protein-like chains.

In the tests on the four relatively small proteins (<600 amino acids) where *SHAPES* was run with the number of beads comparable to the number of amino acids, the execution times for *SHAPES* were 2–9 times shorter than for the comparable runs with *GASBOR* (Table 6). With *DAMMIN* the number of beads used to represent the protein volume varied considerably and the performance of *SHAPES* ranged from marginally faster to 16× faster. *DAMMIF* was the fastest program tested, with all runs completing in <2 min, although for structures characterized by fewer than 400 beads the runtimes for *SHAPES* were also <5 min and the time difference is not likely to be of much operational significance. In the absence of symmetry, runtimes with *SHAPES* appear to be approximately quadratic relative to the number of beads, whereas runtimes with *GASBOR* appear to grow less steeply. When using *SHAPES*, modelling large proteins (over ∼1000 amino acids) with a smaller number of amino acids may be expedient. For example, the CR9 structure contains 2418 amino acids and performing reconstruction runs using half the number of beads resulted in a performance that is almost 3× quicker than the parallel runs with *GASBOR*. If the same calculation is performed with *SHAPES* using 2418 beads, the reconstruction results are similar to the results obtained with the coarser-grained model but *GASBOR* is then slightly faster. Reconstructions of CR9 performed with *DAMMIN* used 1/3 of the time needed for *SHAPES*, and with *DAMMIF* the runtimes are accelerated 12-fold further, but the resulting volumes were modelled by an extremely small number of beads (1/10 the number of amino acids) on a 14 Å grid and the very coarse grained nature of these models may not

fully capture all the information available in the scattering data.

In the presence of a low rotational symmetry the relative speedups for *SHAPES* are considerable but this is not the case with *GASBOR*. For example, test runs with *SHAPES* for D42 and D89 with and without the application of twofold symmetry showed that the incorporation of symmetry reduced the *SHAPES* runtimes by a factor of ∼0.6, but increased the *GASBOR* runtimes by a factor of ∼1.4. *DAMMIN*/*DAMMIF* achieve similar speedups to *SHAPES* provided that the bead radius is set so that it remains unchanged between comparative test runs.

Although the relative completion times for these programs may differ when calculations are performed on computers running other operating systems, with faster processors or in alternative user modes, the calculations presented here show that reconstructions with *SHAPES* may be readily performed on ordinary laptop computers when the number of beads is smaller than ∼1000.

3.3. Program distribution

The *SHAPES* program and documentation are available for download at http://saxs2shapes.com. The program is released in the form of Python source code as an open source distribution under a GNU GPLv3 licence. As such, the program is freely available to both academic and commercial users and may be customized or modified for local use as required.

References

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N. & Bourne, P. E. (2000). *Nucleic Acids Res.* **28**, 235–242.

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

Chen, K. E., Tillu, V. A., Chandra, M. & Collins, B. M. (2018). *Structure*, **26**, 1612–1625.

Cowan-Jacob, S. W., Fendrich, G., Manley, P. W., Jahnke, W., Fabbro, D., Liebetanz, J. & Meyer, T. (2005). *Structure*, **13**, 861–871.

deMel, V. S., Doscher, M. S., Glinn, M. A., Martin, P. D., Ram, M. L. & Edwards, B. F. (1994). *Protein Sci.* **3**, 39–50.

Dharmaiah, S., Bindu, L., Tran, T. H., Gillette, W. K., Frank, P. H., Ghirlando, R., Nissley, D. V., Esposito, D., McCormick, F., Stephen, A. G. & Simanshu, D. K. (2016). *Proc. Natl Acad. Sci. USA*, **113**, E6766–E6775.

Fischer, U., Hertlein, S. & Grimm, C. (2015). *Acta Cryst.* D**71**, 687–696.

Franke, D., Petoukhov, M. V., Konarev, P. V., Panjkovich, A., Tuukkanen, A., Mertens, H. D. T., Kikhney, A. G., Hajizadeh, N. R., Franklin, J. M., Jeffries, C. M. & Svergun, D. I. (2017). *J. Appl. Cryst.* **50**, 1212–1225.

Franke, D. & Svergun, D. I. (2009). *J. Appl. Cryst.* **42**, 342–346.

Grant, T. D. (2018). *Nat. Methods*, **15**, 191–193.

Kozin, M. B. & Svergun, D. I. (2001). *J. Appl. Cryst.* **34**, 33–41.

Kutnowski, N., Shmuely, H., Dahan, I., Shmulevich, F., Davidov, G., Shahar, A., Eichler, J., Zarivach, R. & Shaanan, B. (2018). *J. Struct. Biol.* **204**, 191–198.

Marchesini, S., He, H., Chapman, H. N., Hau-Riege, S. P., Noy, A., Howells, M. R., Weierstall, U. & Spence, J. C. H. (2003). *Phys. Rev. B*, **68**, 140101.

Nagar, B., Hantschel, O., Seeliger, M., Davies, J. M., Weis, W. I., Superti-Furga, G. & Kuriyan, J. (2006). *Mol. Cell*, **21**, 787–798.

Rambo, R. P. & Tainer, J. A. (2013). *Nature*, **496**, 477–481.

Shoichet, B. K., Baase, W. A., Kuroki, R. & Matthews, B. W. (1995). *Proc. Natl Acad. Sci. USA*, **92**, 452–456.

Slonimskiy, Y. B., Maksimov, E. G., Lukashev, E. P., Moldenhauer, M., Jeffries, C. M., Svergun, D. I., Friedrich, T. & Sluchanko, N. N. (2018). *Biochim. Biophys. Acta*, **1859**, 382–393.

Sluchanko, N. N., Slonimskiy, Y. B., Shirshin, E. A., Moldenhauer, M., Friedrich, T. & Maksimov, E. G. (2018). *Nat. Commun.* **9**, 3869.

Svergun, D. I. (1992). *J. Appl. Cryst.* **25**, 495–503.

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

Svergun, D., Barberato, C. & Koch, M. H. J. (1995). *J. Appl. Cryst.* **28**, 768–773.

Svergun, D. I., Petoukhov, M. V. & Koch, M. H. J. (2001). *Biophys. J.* **80**, 2946–2953.

Trofimova, D., Paydar, M., Zara, A., Talje, L., Kwok, B. H. & Allingham, J. S. (2018). *Nat. Commun.* **9**, 2628.

Tuukkanen, A. T., Kleywegt, G. J. & Svergun, D. I. (2016). *IUCrJ*, **3**, 440–447.

Valentini, E., Kikhney, A. G., Previtali, G., Jeffries, C. M. & Svergun, D. I. (2015). *Nucleic Acids Res.* **43**, D357–D363.

Walther, D., Cohen, F. E. & Doniach, S. (2000). *J. Appl. Cryst.* **33**, 350–363.