Simulation of small-angle scattering from large assemblies of multi-type scatterer particles

Emmanuel Pantos\textsuperscript{a,}\textsuperscript{*}, Harold F. van Garderen\textsuperscript{b}, Peter A.J. Hilbers\textsuperscript{b}, Theo P.M. Beelen\textsuperscript{b}, Rutger A. van Santen\textsuperscript{b}

\textsuperscript{a}CLRC, Daresbury Laboratory, Keckwick Lane, Warrington, Cheshire WA4 4AD, UK
\textsuperscript{b}Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands

Received 27 September 1995; accepted 1 November 1995

Abstract

We describe a central processing unit (CPU)-efficient expansion of the Debye scattering formula for the calculation of small-angle scattering patterns of model systems composed of different types of scatterers. The algorithm permits the use of atomic scattering factors or form factors of hard spheres of variable radius and scattering density. We apply the algorithm to the computation of partial small-angle scattering profiles in biological multi-type systems and examine the relative importance of particles with different connectivities in determining the fractal dimension of large particle networks.

Keywords: Small angle scattering; Simulation; Polymer networks

1. Introduction

Solution small-angle scattering (SAS) with X-rays (SAXS) or neutrons (SANS) is an established experimental technique for the study of structural parameters of individual particles such as large biomolecules [1] and of the morphology of composite structures such as gels and zeolites. Synchrotron radiation sources have enhanced the power of the technique in many different respects, more notably in permitting the study of the time evolution of complex structures [2].

The basic theory of modeling SAS profiles was developed nearly 50 years ago and is described in the classic textbooks of Guinier and Fournet [3], and Glatter and Kratky [4]. For models built of individual scatterers, typically non-penetrating spheres of given scattering density, the approximate formula of Debye [5] can be employed

\[ I(q) = \sum_{i=1}^{N} f_i(q) + 2 \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} F_i(q)F_j(q) \frac{\sin qr_{ij}}{qr_{ij}} \]

(1)

where \( F_i(q) \) is the form factor of a sphere of radius \( R_i \) and scattering density \( \rho_i \), and \( r_{ij} \) is the distance between spheres \( i \) and \( j \).

Although Eq. (1) is simple enough to implement into code it is rather expensive in computing time.
for models composed of more than a few hundred spheres. This problem has been addressed [4,6,7] by introducing two additional approximations. (a) All spheres have the same radius and scattering density. (b) Computation of the \((\sin q_{rij})/q_{rij}\) values is handled via a pair-distance histogram \(g(r)\) with a bin-width commensurate with the experimental resolution. Assumption (a) is considered adequate for protein systems at resolution up to 2–3 nm while approximation (b) is perfectly adequate for bin-widths at least 50 times smaller than the sphere radius and a high resolution limit of the order of the sphere radius. Using these approximations we can write the scattered intensity as

\[
I(q) = N I_s(q) + 2F_s^2(q) \sum_{i=1}^{N_{\text{bins}}} g(r_i) \frac{\sin q_{ri}}{q_{ri}}
\]

where the function \(g(r_i)\) is the pair–distance histogram. Because \(I_s(q) = F_s^2(q)\), Eq. (2) can be simplified as the product of two functions, \(S(q)\) the structure factor, and \(P(q)\) the form factor square

\[
I(q) = F_s^2(q) \left[N + 2 \sum_{i=1}^{N_{\text{bins}}} g(r_i) \frac{\sin q_{ri}}{q_{ri}}\right] = P(q) \cdot S(q)
\]

Eq. (3) brings significant computational advantages over Eq. (1). The computationally expensive \((\sin q_{ri})/q_{ri}\) terms need be calculated only for \(N_{\text{bins}}\) terms, typically a few thousand, rather than \(N(N-1)\) as in Eq. (1). Even very large systems composed of tens of hundreds of thousands of spheres such as used to simulate gel formation and ageing [8,9] can be handled in realistic timescales. The implementation is easily adapted for execution on parallel supercomputers or networked workstations using a parallel virtual machine (PVM) harness [10], the scaling in time being inversely proportional to the number of processors employed.

The most important drawback of Eq. (3) is that it can only be used for systems of identical scatterers. This is a serious loss of generality, and for systems containing more than one type of scatterer one has to return to the original central processing unit (CPU)-demanding Debye formula Eq. (1). We thus face the problem of adapting the algorithm to efficiently handle systems where each individual (atomic even) scatterer type is characterized by its own scattering factor.

2. Different scattering types

The optimization of Eq. (3) is achieved by sorting out pair distances by type so that the distance histogram decomposes in blocks. This is illustrated in Fig. 1(a) that depicts a simple two-dimensional

---

Fig. 1. (a) A simple two-dimensional system of 13 scatterers. (b) The distance matrix. (c) The type-sorted distance–distance matrix. Note that the matrices (b) and (c) are diagonally symmetric. This causes the factor two in Eqs. (1)–(3) and (5).
example system of 13 scatterers of three different types, seven of type A, two of type B and four of type C, and Fig. 1(b) shows the corresponding distance matrix. In Fig. 1(c) the matrix elements have been sorted according to type. Each block corresponds to a partial \( g(r) \) for all the type–type pairs. The number of blocks for an \( N \)-type system is

\[
N_{\text{blocks}} = \frac{(N_{\text{types}} + 1)}{2} = \frac{N_{\text{types}}(N_{\text{types}} + 1)}{2} \tag{4}
\]

Of course the type sorting approach can be combined with the histogram approach leading to \( N_{\text{blocks}} \) partial histograms that each represent the distances between a pair of scatterer types. The sum over \( N_{\text{bins}} \) in Eq. (3) can then be written as an expanded sum, each using the appropriate \( g(r) \). The partial intensities \( I_{ij}(q) \) can be simply generated from the partial structure factors by multiplying by the appropriate form factor product \( F_{ij}(q) = F_i(q)F_j(q) \). The total \( I(q) \) of the scatterers in a multi-type system can be easily obtained now by a double summation over all the partial intensities

\[
I(q) = \sum_{i=1}^{N_{\text{types}}} N_i F_i^2(q) + 2 \sum_{i=1}^{N_{\text{types}}} \sum_{j=1}^{N_{\text{types}}} I_{ij}(q)
\]

\[
= \sum_{i=1}^{N_{\text{types}}} N_i P_{ii}(q) + 2 \sum_{j=1}^{N_{\text{types}}} P_{ij}(q)S_{ij}(q) \tag{5}
\]

At first sight the expansion of Eq. (3) into Eq. (5) for multi-type systems introduces considerable computational complexity with increased demands on CPU and memory requirements. In practice, however, most systems contain only a few different types of scatterers. Biological molecules such as proteins consist mainly of H, C, N, O and S atoms and occasionally contain P (in phosphates), cations (Ca\(^{2+}\), Mg\(^{2+}\)) or other metal atoms. Other chemical systems, such as aluminosilicate-based zeolites, are also composed of a small number of scatterer types. When this number of types is low, typically below ten, it becomes practical to implement the multi-type expression of the Debye formulation. The sorting of the pair distances according to type is an essential step in simplifying the implementation. The reduced computational efficiency resulting from folding-out the computation of Eq. (3) turns out to be an acceptable penalty to pay.

### 3. Examples of application of multi-type scattering calculations

#### 3.1. Use of atomic scattering factors

The use of hard spheres in the Debye formula is only for analytical (and computational) convenience. Gaussian sphere form factors may be used instead to suppress the sharp minima of the sphere form factor at high \( q \) values which limit the useful resolution range of the computed pattern. Atomic form factors used in protein structure refinement programs (e.g. XPLOR [11]) use a parameterized Gaussian expression for the form factors (valid up to \( \sin \theta/\lambda = 2 \)) of the form

\[
F(q) = \sum_{i=1}^{4} a_i \exp(-b_i q^2/4) + c \tag{6}
\]

The values of parameters \( a_i, b_i \) and \( c \) are tabulated in the International Tables for Crystallography [12]. The algorithm makes use of this table and assigns form factor values according to the type of atom in the input coordinate set. The impact on computational speed is insignificant because form factor values are calculated at the start and tabulated in arrays for the \( q \) range interval selected.

The difference in scattering mass (number of electrons for SAXS) between individual atomic scatterers is taken into account in the value of the parameters \( a_i, b_i \) and \( c \) in Eq. (6). In protein systems where the bulk of the scattering mass is due to C, N S and O atoms of comparable scattering strength and similar form factor shape, it is to be expected that the computed pattern will differ little from that where all atoms are represented by a single scattering type. Comparison with model SAS patterns employing hard spheres of a radius equal to the van der Waals’ radius of individual atoms (Fig. 2) shows that the difference is significant only
3.2. Application to GDP–tubulin double rings

Diaz et al. [13] have interpreted the scattering pattern of GDP–tubulin polymer formations as arising out of double rings each composed of 24 and 32 tubulin monomers for the inner and outer

...
ring, respectively, with the monomer subunits at a particular orientation. SAXS patterns of rings are expected to exhibit a series of Bessel function maxima, the positions of which can be used to calculate ring parameters. A puzzling feature in the experimental data was the “washing out” of the Bessel peak series in the $s$ range $0.07-0.14$ nm$^{-1}$. Labelling of the tubulin monomers according to type (inner or outer ring) permits the computation of the partial profiles and the interference term (Fig. 3) from which it can be clearly seen how the three components combine to cancel out Bessel function maxima. Simulations with a number of alternative schemes and comparison with the experimental data have led to the conclusion that the most plausible arrangement of the monomers is that shown in Fig. 3.

A similar approach may be used in the case of multi-domain protein molecules. Interest in SAXS measurements of proteins with known atomic structure from high resolution protein crystallographic measurements arises mainly from the need to establish whether the molecular conformation in the crystal differs from that in solution, the natural environment for proteins to perform their function. The liganded/unliganded form of a protein is also often followed by large domain movements. An example is the case of the transferrin proteins [14]. Just as for the tubulin double rings, it is possible to easily distinguish the contributions of each individual domain and the interference term by labelling the atoms according to the domain in which they are situated.

### 3.3. Application to fractal DLCA systems

The CPU efficiency of the code permits the calculation of SAS profiles of large assemblies of particles. This is of particular interest for colloidal systems because one could actually attempt to simulate the time evolution of the aggregation process itself, given realistic modelling parameters for volume fractions, sticking coefficients, relative orientation after contact, etc., using a particle aggregation program such as GRASP [8]. Fig. 4 gives an example of application in analysing the fractal backbone and fractal dimension in particle network systems. Wet gels and aerogels contain fractal aggregates, i.e. systems which show morphological similarity at a range of length scales and which can be characterized by a measure called the fractal dimension $D_f$. At length scales greater

![Fig. 4. (a) 160000-particle connected gel and (b) SAS profiles for the partial structure factors and the whole system. CPU times for calculation of the partial histograms and structure factors using Eq. (5) are 3588.5 and 1866.3 s on one R8000 processor of a Silicon Graphics PowerChallenge computer. Using the single type method of Eq. (3) these timings are 3708.6 s and 207.9 s, respectively. These timings indicate that the only penalty incurred for the decomposition into partial structure factors is paid during the computation of the partial structure factors themselves.](image-url)
than about 5–10 particle radii, the log $I(q)$–log $q$ SAS graph exhibits a linear dependence with a slope of $-D_f$ [8]. The low-$q$ limit of this linear region corresponds to the particle correlation length, $\xi$.

The particle network can be imaged as a collection of connected chains that we can classify using NMR terminology: single-, doubly-, etc., connected monomers are called $Q_1, Q_2, \ldots, Q_n$ particles. In the distance histogram we can easily identify blocks of distances belonging to a specific type of connectivity. All the distances between, for example, the $Q_1$ monomers are in the upper left block of Fig. 1(c) and that block is therefore termed the $Q_{11}$ block. Likewise, the distance block representing distance between $Q_1$ and $Q_2$ monomers is called the $Q_{12}$ block, and so on.

It would be interesting now to determine, which $Q$ components of the particle network determine the fractal dimension by computing the partial SAS patterns of each of the connectivity type pairs. Fig. 4(a) shows the real-space morphology of a two-dimensional fractal system of particles prepared using diffusion-limited cluster–cluster aggregation (DLCA) conditions and (b) the SAS profile for the whole system and partial $S(q)$ patterns.

It is clear that from the slope of the graphs that the $Q_{12}$, $Q_{22}$ and $Q_{23}$ skeletons are all fairly representative of the whole system for the aggregation conditions used in this example, i.e. the chain end ($Q_{12}$), the chain ($Q_{22}$) and the chain-branch ($Q_{23}$) contributions have the same fractal properties as the complete particle network, whereas the other contributions are either non-fractal or add little to the overall intensity.

4. Conclusions

We have outlined the modifications to the basic SAS simulation algorithm for handling multi-scatterer model systems with hard-sphere or atomic form factors. The implementation is based on a decomposition of the distance histogram in a type–type pair block and permits the analysis of large systems using the Debye formula.

We have presented three examples in which the multi-type Debye method is used for both biological and chemical systems. For lactoferrin it has been shown that the use of atomic scattering factors results in a significant difference in the SAS pattern only outside the resolution range covered at present by experiment. In the second example, analysis of the partial profiles of the double tubulin ring model has shown that the experimentally observed damping of the Bessel oscillations in the $s$ range between 0.07 and 0.14 nm$^{-1}$ can be accounted for by the interference term between the inner and the outer ring. The third example, where a model silica system is decomposed in connectivity classes, has shown that the chain ends, the chains and the branch points have the same fractal properties as the whole system. The other contributions are either non-fractal or add little to the overall pattern.

References