Simulation of small-angle scattering patterns using a CPU-efficient algorithm

To cite this article: E M Anitas 2017 J. Phys.: Conf. Ser. 936 012030

View the article online for updates and enhancements.
Simulation of small-angle scattering patterns using a CPU-efficient algorithm

E M Anitas$^{1,2}$

$^1$Joint Institute for Nuclear Research, Dubna 141980, Moscow region, Russia
$^2$Horia Hulubei National Institute of Physics and Nuclear Engineering, RO-077125 Bucharest, Romania

E-mail: anitas<at>jinr.ru

Abstract. Small-angle scattering (of neutrons, x-ray or light; SAS) is a well-established experimental technique for structural analysis of disordered systems at nano and micro scales. For complex systems, such as super-molecular assemblies or protein molecules, analytic solutions of SAS intensity are generally not available. Thus, a frequent approach to simulate the corresponding patterns is to use a CPU-efficient version of the Debye formula. For this purpose, in this paper we implement the well-known DALAI algorithm in Mathematica software. We present calculations for a series of 2D Sierpinski gaskets and respectively of pentaflakes, obtained from chaos game representation.

1. Introduction

Small-angle neutron/x-ray/light scattering (SANS/SAXS/SALS) is widely used to study the structure of disordered systems at nano and micro scales, with applications to metallic alloys, fibers, viruses, proteins, colloidal suspensions, etc. [1–10]. The technique yields the scattering intensity (i.e. elastic cross section per unit solid angle, normalized per unit volume) $I(q)$ vs. $q$, where $q = (4\pi \sin \theta)/\lambda$ is the magnitude of the scattering vector, $\theta$ is half the scattering angle, and $\lambda$ is the wavelength of the radiation. Provided the scattering intensity is measured in relative units and depending on the structure being analyzed, various geometrical and/or fractal characteristics about scattering systems can be obtained, such as: radius of gyration, correlation length, specific surface, the largest dimension, fractal dimension, scaling factor or the number of scattering units [11–18]. If the scattering curve is measured in absolute units, then weight parameters can also be determined.

For simple geometrical structures, analytic expressions for the form factor are well-known [19] and for several types of exact self-similar (deterministic) fractals they have been recently developed [18, 20–27]. However, for more complex systems which don’t have the exact self-similarity property, such as super-molecular assemblies or protein molecules, analytic solutions are generally not available. Therefore, in such cases one has to resort to Debye formula [2, 28].

Although Debye formula is quite straightforward to implement into a computer program, it may become very time consuming when the number of scattering units exceeds few thousands. An elegant solution to this problem has been suggested in Refs. [29, 30] by discretizing the pair distances in a histogram. In Ref. [29] a pseudo code, which is the precursor of the DALAI program, has been provided also.

In this paper, Mathematica symbolic language is used to implement the DALAI’s pseudo code and to efficiently simulate small-angle scattering (SAS) patterns of large structures, containing thousands of scattering centers. The obtained program is used to calculate the scattering curves for different structures.
generated using chaos game representation such as 2D Sierpinski gasket and pentaflakes. The computing
time is greatly reduced as compared to the classical Debye model, and thus the main regions of interest
(Guinier, intermediate/fractal and Porod) are obtained within minutes.

2. Theoretical background
In this work we consider a two-phase approximation where the objects have scattering length \( b_j \)
and scattering length density (SLD) \( \rho_j(r) = \sum_{i=1}^{N} b_j \delta(r - h_j) \). Here, \( h_j \) are the position vectors
of the scattering objects. Then, the differential elastic cross section is \( d\sigma/d\Omega = |A_i(q)|^2 \), where
\( A_i(q) = \int_{V_{tot}} \rho_j(r) \exp(iq \cdot r) dr \) is the total scattering amplitude and \( V_{tot} \) is the total volume irradiated
by the radiation used. Considering that the scattering objects are immersed in a solid matrix of SLD \( \rho_0 \),
the total scattering intensity is

\[
I(q) \equiv \frac{1}{V_{tot}} \frac{d\sigma}{d\Omega} = c |\Delta \rho|^2 V^2 \left\langle |F(q)|^2 \right\rangle ,
\]

where \( |\Delta \rho| = \rho - \rho_0 \) is the scattering contrast, \( c \) is the concentration of fractal, \( V \) is the
volume of each fractal, and \( F(q) = \int_{V} \exp(-iq \cdot r) dr \) is the normalized form factor, with \( F(0) = 1 \).
Averaging over all orientations in 3D space is performed according to \( \langle f(q) \rangle = \frac{1}{4\pi} \int_0^\pi d\theta \sin \theta \int_0^{2\pi} d\phi f(q, \theta, \phi) \).

It is known that the total scattering intensity can be written as [22]

\[
I(q) = I(0)S(q)|F_0(qR)|^2 / N ,
\]

where \( S(q) \) is the structure factor, \( F_0(qR) \) is the form factor of the basic scattering 'unit' composing the
object, and \( N \) is the number of the units. The classical Debye formula [28] gives

\[
I^D(q) = NI_s(q) + 2F_s^2(q) \sum_{i=1}^{N_{\text{bins}}} \sum_{j=i+1}^{k} \frac{\sin qr_{ij}}{qr_{ij}} ,
\]

where \( I_s(q) \) is the intensity scattered by each scattering unit, and \( r_{ij} \) is the distance between units \( i \)
and \( j \). When the number of units exceeds few thousands, the computation of the term \( \sin(qr_{ij})/(qr_{ij}) \)
is very time consuming, and thus it is handled via a pair-distance histogram \( g(r) \), with a bin-width
commensurate with the experimental resolution [29, 30]. Therefore Eq. (3) can be rewritten as

\[
I^D(q) = NI_s(q) + 2F_s^2(q) \sum_{i=1}^{N_{\text{bins}}} q(g(r_i) \sin qr_i/qr_i) ,
\]

where \( g(r_i) \) is the pair-distance histogram at pair distance \( r_i \). The latter quantity is calculated from the
positions of scattering units. Since we are interested in Guinier, Porod and asymptotic regions, we can
neglect the form factor, and consider \( I_s(q) = F_s^2(q) = F_0(qR) = 1 \). Thus, the intensity given by Eq. (4)
becomes

\[
I^D(q) \equiv S^D(q) = N + 2 \sum_{i=1}^{N_{\text{bins}}} g(r_i) \sin qr_i/qr_i ,
\]

and gives the structure factor. Taking into account the normalization used in Eq. (2), subsequently, the
final expression of the scattering structure factor in Eq. (5), is represented as \( S^D(q)/N^2 \).
3. Mathematica algorithm for calculating SAS intensity

Using the direct calculation form of the structure factor equation (3), one can develop an efficient implementation of the Debye formula, similar to that used for DALAI’s program [29, 30]. In both versions, the distances $r_{ij}$ between all pairs of scattering 'units' have to be known. With the Debye formula, one must calculate the terms $\sin(qr_{ij})/(qr_{ij})$ which is very time consuming, when the number $N$ of scattering units exceeds few thousands. In the efficient version of the algorithm this process is accelerated by discretizing the pair distances in a histogram of bin size commensurate with the required spatial resolution in the pattern [29,30]. In what follows, a simple explanation of the method is provided, along with key Mathematica code implementation and steps.

Given a data set $X : \{(x_i, y_i) \in \mathbb{R}^2, i = 1, 2, \cdots , n\}$ with $n$ data points representing the positions of the scattering units, and assuming they are stored in an ASCI file, then we import data from the file, and store them in the list pos by using the command

$$\text{pos} = \text{Import["file"];}$$

The following code calculates the distances between all the pairs $(x_i, y_i), i = 1, 2, \cdots , n$ and store them in the list dist:

$$d = \{\};$$

$$\text{For}[s = 1, s \leq \text{Length[pos]} - 1, ++s,$$

$$\text{AppendTo}[d, \text{EuclideanDistance[pos[[s]], pos[[1 + #]]]} \&/\!\!/\text{Drop[Range[Length[pos] - 1], s - 1]}];$$

$$\text{dist} = \text{Flatten[d];}$$

The next command tallies the elements in dist, storing all distinct distances together with their multiplicity in tallyDistances:

$$\text{tallyDistances} = \text{Tally[dist];}$$

Then, the values of the distances are sorted from the lowest to the highest value by the following command:

$$\text{val} = \text{Sort[tallyDistances]][[\text{All}, 1]];$$

The corresponding multiplicities of the distances are obtained in a similar way:

$$\text{freq} = \text{Sort[tallyDistances]][[\text{All}, 2]];$$

The list of all distances, with each individual distance appearing as many times as given by its corresponding multiplicity, is constructed as:

$$\text{hist} = \text{Flatten[\text{Table[val[[#]], \{freq[[#]]\}] &/\!\!/\text{Length[val]}];}$$

The list of bins and the histogram heights with bins specified by w and commensurate with the required spatial resolution in the pattern (see next section) can be obtained from:

$$\text{hl} = \text{HistogramList[hist, \{w\};}$$

Then, the pair-distance histogram at pair distance $r_i$ (see Eq. (4)) is obtained by:

$$\text{pd} = \text{Drop[Table[\{hl[[1, i]], hl[[2, i]]\}, \{i, 1, \text{Length[hl[[2]]]}\}], 1];}$$

By defining the sinc function involving the the pair-distances pd, such as:
\( \text{sincpd}[q_] := \frac{\text{Sin}[q \text{pd}[[\text{All}, 1]]]}{q \text{pd}[[\text{All}, 1]]} \);

then, finally, the expression of scattering intensity can be written as:

\[
S[q] := \frac{1}{\text{Length}[\text{pos}]} \left( \text{Length}[\text{pos}] + 2 \sum_{i=1}^{\text{Length}[\text{pd}]} \text{pd}[[i, 2]] \text{sincpd}[q[[i]]] \right);
\]

4. Applications

We make use of the algorithm above to calculate the scattering intensity for 2D Sierpinski gaskets and pentaflakes.

4.1. Sierpinski gaskets

We consider a Sierpinski triangle of side length \( a = 1 \), and centered in the origin. SG is generated by playing chaos game on three vertices. The points of the SG are obtained starting with an initial point chosen at random, and calculating each subsequent point as one half of the distance between the previous point and one of the vertices (randomly selected). The results for few sets of points are shown in Fig. 1.

\[ \text{Figure 1.} \quad \text{The structure of two-dimensional SG from chaos game representation, for } N = 100, 250, 500 \text{ and respectively } N = 1000 \text{ points.} \]
In Fig. 2 is shown the structure factor of SG generated from the CGR, for different values of the number of units composing the fractal. All the scattering curves are characterized by the presence of four main regions. At $q \lesssim 1/a$ a plateau is observed (Guinier region, where $I(q) \propto q^0$), and from the end of this plateau, information about the overall size of the fractal can be extracted. At $1/a \lesssim q \lesssim 1/d$ (where $d$ is of the order of the distances between the scattering points) we have the fractal regime (where $I(q) \propto q^{-1.585}$), from which the fractal dimension ($D = 1.585$) can be extracted. Note that by increasing the number of points, the distances between them decreases and thus, the length of the fractal regime increase also. For $1/d \lesssim q$ a transition region appears between fractal and asymptotic regions. For $N = 1000$ this region is at $10^2 \leq qa \leq 8 \cdot 10^2$ (Fig. 2). Finally, when $1/d \ll q$, the asymptotic region is attained.

Figure 2. (Color online) The structure factor of SG using CGR. The horizontal lines indicate the value of the asymptotes in the limit of high $q$ values.

4.2. Pentaflakes
We generate the fractal pentaflake starting from the center of a five-sided polygon of side length $a$. A point is drawn at a fraction $\beta_s = 0.35$ of the distance between a randomly chosen vertex and the polygon center. Fig. 3 show the fractal pentaflake when the process is repeated for $N = 4000$ points. Thus, the corresponding fractal dimension is given by

$$D \approx -\log 5/\log 0.35 \simeq 1.533.$$  

(6)

The structure factor corresponding to the pentaflake is shown in Fig. 4. The numerical value of the fractal dimension obtained from the slope of the scattering curve in the fractal region coincides with the one given by theoretically by Eq. (6). The numerical value $\beta_s = 0.35$ of the scaling factor can be recovered from the periodicity of minima in the fractal region, and this is a specific feature of scattering from mass fractals with a single scale.
Figure 3. The structure factor of two-dimensional pentaflake using CGR for $N = 4000$ points and scaling factor 0.35.

Figure 4. (Color online) The structure factor of pentaflake using CGR. The horizontal line indicates the value of the asymptote in the limit of high $q$ values.
5. Conclusion
We implement in Mathematica software a CPU-efficient version of the Debye formula to calculate the small-angle scattering intensity from complex structures. We use the algorithm to reveal the main structural properties (fractal dimension, the overall fractal size, scaling factor and the number of particles composing the fractal) of fractal structures generated from chaos game representation.

The computational time is few orders of magnitude lower that in the traditional Debye approach, and this can be used to analyze various structures containing thousands of scattering centers (particles) when analytic solutions are not available. The efficiency of the algorithm can be further increased by using the Compile function available in Mathematica.

References
[1] Feigin L A and Svergun D I 1987 Structure Analysis by Small-Angle X-ray and Neutron Scattering (NY: Plenum Press)
[2] Glatter O and Kratky O 1982 Small-angle X-ray Scattering (London: Academic Press)
[3] Stevenson W D, Ahmed Z, Zeng X B, Welch C, Ungur G and Mehli G H 2017 Phys. Chem. Chem. Phys. 19(21) 13449–13454
[4] Eicher B, Heberle F A, Marquardt D, Rechberger G N, Katsaras J and Pabst G 2017 J. Appl. Cryst. 50 419
[5] Demchenko V, Rjabov S and Shitompe’ V 2017 Nanoscale Research Letters 12 235
[6] Lychak O V, Prulutsksyy Y I, Rjabalchenko V K, Kyzyma O A, Soloviov D, Kostjukov V V, Evstigneev M P, Ritter U and Scharff P 2017 Nanoscale Research Letters 12 8
[7] Wisotzki E L, Tempesti P, Fratini E and Mayr S G 2017 Phys. Chem. Chem. Phys. 19(19) 12064–12074
[8] Pandey P K, Rawat K, Aswal V K, Kohlbrecher J and Bohidar H B 2017 Phys. Chem. Chem. Phys. 19(1) 804
[9] Trinh V and Xu J 2017 Nanoscale Research Letters 12 216
[10] Singh A, Vihinen J, Frankenberg E, Hyyrinen L, Honkanen M and Levanen E 2016 Nanoscale Research Letters 11 447
[11] Bahadur J, Medina C R, He L, Melnichenko Y B, Rupp J A, Blach T P and Mildner D FR 2016 J. Appl. Cryst. 49 2021–2030
[12] von Gundlach A R, Garamus V M, Willey T M, Ilavsky J, Hilpert K and Rosenhahn A 2016 J. Appl. Cryst. 49 2210–2216
[13] Benavent V, Frizon F and Poulesquen A 2016 J. Appl. Cryst. 49 2116–2128
[14] Hiroe K, Fujii K, Ueki T, Kitazawa Y, Littrell K C, Watanabe M and Shibayama M 2016 Phys. Chem. Chem. Phys. 18(27) 17881–17889
[15] Sanchez-Fernandez A, Edler K J, Arnold T, Alba Venero D and Jackson A J 2017 Phys. Chem. Chem. Phys. 19(13) 8667–8670
[16] Rossi B, Paciaroni A, Venuti V, Fadda G C, Melone L, Punta C, Crupi V, Majolino D and Mele A 2017 Phys. Chem. Chem. Phys. 19(8) 6022–6029
[17] White C E, Olds D P, Hartl M, Hjelm R P and Page K 2017 J. Appl. Cryst. 50 61–75
[18] Cherny A Y, Anitas E M, Osipov V A and Kulkin A I 2017 Phys. Chem. Chem. Phys. 19(3) 2261–2268
[19] SASFit https://kur.web.psi.ch/sans1/SANSSoft/sasfit.pdf accessed: 9 June 2017
[20] Cherny A Y, Anitas E M, Osipov V A and Kulkin A I 2017 J. Appl. Cryst. 50 919
[21] Anitas E M, Sylanov A, Todoran R and Szakacs Z 2017 Nanoscale Research Letters 12 389
[22] Cherny A Y, Anitas E M, Kulkin A I and Osipov V A 2011 Phys. Rev. E 84 036203–1–036203–11
[23] Cherny A Y, Anitas E M, Osipov V A and Kulkin A I 2015 Rom. Journ. Phys 60 658–663
[24] Cherny A Y, Anitas E M, Osipov V A and Kulkin A I. 2014 J. Appl. Cryst. 47 198–206
[25] Anitas E M 2014 Eur. Phys. J. B 87 139
[26] Anitas E M 2015 Advances in Condensed Matter Physics 2015 Article ID 501281
[27] Anitas E M 2015 Rom. Journ. Phys. 60 647–652
[28] Debye P 1915 Ann. Phys. 46 2927
[29] Pantos E and Bords I 1994 Pure & Appl. Chem. 66 77
[30] van Garderen H F, Pantos E, Dokter W H, Beelen T P M and van Santen R A 1994 Modeling Simul. Mater. Sci. Eng. 2 295