A new algorithm for computer simulation of the off-lattice ballistic aggregates

1988 J. Phys. A: Math. Gen. 21 739

(http://iopscience.iop.org/0305-4470/21/3/029)

J. Phys. A: Math. Gen. 21 (1988) 739-746. Printed in the UK

A new algorithm for computer simulation of the off-lattice ballistic aggregates

P S Joag

Department of Physics, University of Poona, Pune 411007, India

Received 16 July 1987, in final form 29 September 1987

Abstract. We present here a new algorithm to simulate 2D ballistic aggregates with one-sided rain on a seed. This algorithm makes use of the non-local shadowing effect by which a particle in the aggregate gets shadowed by a chain of particles originated elsewhere in the cluster. This algorithm has far less memory requirement as compared to the conventional algorithm. Physical implications of the triangle rule proposed as a part of the algorithm are discussed. Possible generalisations of the algorithm are indicated.

1. Introduction

In recent years considerable attention has been focused on the irreversible aggregation processes in which particles approaching from outside are irreversibly added to growing aggregates [1, 2]. One of the first such models was diffusion limited aggregation (DLA) in which the diffusion of particles to the aggregate is the rate limiting step [3]. A simpler model in which the particles travel towards the aggregate in straight lines has also been studied quite extensively. This is known as the ballistic driven aggregation (BDA) model [4-12]. This model corresponds to the growth of amorphous thin films [8, 13] and also corresponds to the 'drift only' limit of DLA [1, 5].

Two versions of the BDA model have been studied. In the on-lattice version [9], point particles move on the lattice and stick to the cluster if one of the nearest neighbours belongs to the cluster. In the off-lattice case [10-12] particles of finite size move in a continuum and stick to the first contacted particle in the aggregate. In this paper we are exclusively concerned with the off-lattice BDA.

The rules for BDA are fairly simple. Particles move in straight lines, in a single direction, and fall on the previously grown aggregate of size N ($N \geq 1$), A falling particle adds to the aggregate at the first contact with a previously added particle. There are two variants: growth on a single seed particle [7] and growth on a line [8, 13]. In the first case one gets a fan, while in the second case a columnar structure appears for large angles of incidence. As we shall see in § 3, the ingredients of columnar growth are present even in the fan structure. The algorithm presented here applies to growth on a seed particle, but can be easily modified to incorporate growth on a line.

The BDA model has the following characteristics. There is no interaction between the falling particles. Also the grown aggregate does not influence the falling particles in any way except at the contact. Within these limitations we can assume that the events of addition of particles to the aggregate are statistically independent. In the computer simulations, therefore, we can add one particle at a time to the aggregate.

0305-4470/88/030739+08$02.50

Probably the most important geometrical fact about BDA is the non-local shadowing of one part of the structure by another. To be precise, a particle in the aggregate exposed to the falling particles at a certain stage of growth may get shadowed (partly or fully) at a later stage of growth, by a chain of particles originating somewhere else in the aggregate. A particle can, of course, get shadowed locally, by another particle directly falling over it. This shadowing effect plays a central role in most of the geometric properties of BDA.

2. The algorithm

At a given stage of growth, a particle in the aggregate is defined to be a surface particle if a falling particle can make direct contact with it. A particle which is not a surface particle is a core particle. Due to the shadowing effect, a surface particle at a certain stage of growth becomes a core particle some time later as the aggregate grows. This particular fact lies at the heart of the present algorithm.

Consider the aggregate at a time t after the growth starts. Let x_L and x_R denote the x coordinates of the two extreme particles of the aggregate†. Further, let r denote the radius of the particle. Note that the extreme particles are necessarily surface particles. We now associate with every surface particle an interval along the x axis with the following defining property. A falling particle having the x coordinate of its centre within the interval associated with a surface particle necessarily hits that surface particle. Such an interval can be associated with every surface particle because the event that a falling particle simultaneously hits more than one surface particle has zero probability. The values of the x coordinates for which such an event can take place forms a discrete set of points on the x axis which has zero probability. Let us call the associated interval the growth interval of the surface particle. Two successive growth intervals are separated by a vertical demarcation line. A growth interval has a left demarcation line (LDL) and right demarcation line (RDL). A particle falling with its centre exactly along a demarcation line hits more than one particle simultaneously. As stated above, such an event has zero probability. It is easy to see that all the growth intervals are intrinsically ordered. If x_1 and x_2 denote the x coordinates of the surface particles 1 and 2, respectively, and if $x_1 < x_2$, then the growth interval of particle 2 lies on the right of that of particle 1. Also, the set of all growth intervals partitions the range $x_L - 2r$ to $x_R + 2r$. A part of the surface particles and their growth intervals is shown in figure 1.

We now assume that we are given an aggregate comprising N particles. This means that positions of all the N particles as well as the list of surface particles with a corresponding list of growth intervals is known. We shall now present a procedure to

(i) add the $(N+1)$th particle to the aggregate; and

(ii) restructure the list of growth intervals and surface particles as a result of addition of the $(N+1)$th particle.

As a result of the second step, the lists of growth intervals and surface particles get reconstructed. In this process some of the surface particles may become core particles, as a result of total shadowing due to the newly added particle. When the second step is completed, the $(N+2)$th particle can be added to the aggregate. Steps

† There will be exactly two extreme particles in the cluster because more than two extreme particles corresponds to the event that two particles fall exactly on top of each other. This event has zero probability.

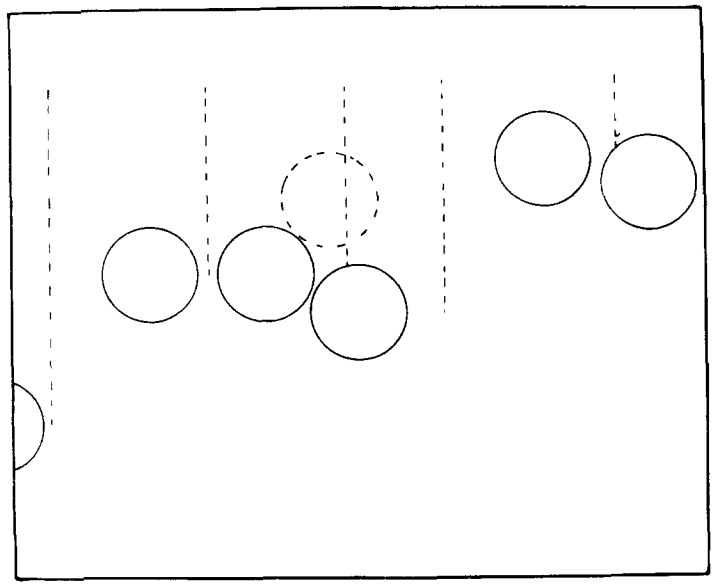

Figure 1. A section of the surface particles and their growth intervals. Note that all the particles are not in direct contact, which shows that shadowing is non-local. By assuming various particle positions of the $(N+1)$th particle (the broken circle), one can verify step (ii) in the algorithm (see the text).

(i) and (ii) form the core of our algorithm which is iterated every time a new particle is added.

The first step is fairly easy to accomplish. We choose at random a value of x (say x_f) lying between $x_L - 2r$ and $x_R + 2r$ as the x coordinate of the falling particle. This x_f value must lie in one and only one growth interval which fixes the surface particle to which this falling particle sticks. It is trivial to calculate the y coordinate of the newly added $(N+1)$th particle in the cluster. This completes step (i).

In order to accomplish step (ii), note that the newly added particle will affect only the fraction of surface particles whose centres fall in the range $x_f \pm 2r$. Therefore, we concentrate only on such surface particles and their growth intervals. We call these particles 'concerned' particles. Let the number of concerned particles be n_c.

There is a surprisingly simple rule to determine the reconstructed position of the demarcation line separating successive intervals and hence the reconstructed intervals themselves. As shown in figure 2(a), let A and B represent the centres of two surface particles at a distance less than $4r$. This figure also shows two arcs of circles both having radius $2r$, drawn around A and B respectively. These arcs form the locus of the centre of a particle stuck to either A or B. C is the point of intersection of these two arcs and a particle with its center at C will simultaneously touch both A and B. Obviously a particle falling with its center on the left of the vertical line drawn through C will stick to A and that falling on the right will stick to B. Thus this line is the demarcation line for the growth intervals of A and B. As shown in the figure, point C lies at the apex of the isosceles triangle with sides $AC = BC = 2r$ and $AB < 4r$. Thus the position of the demarcation line is the position of vertex C of this isosceles triangle.

Figures 2(b) and 2(c) depict situations where this triangle rule does not apply. Let y_A denote the y coordinate of particle A and y_s denote the y coordinate of a particle added to the cluster after A and stuck to A. By the rules of the BDA model it is clear

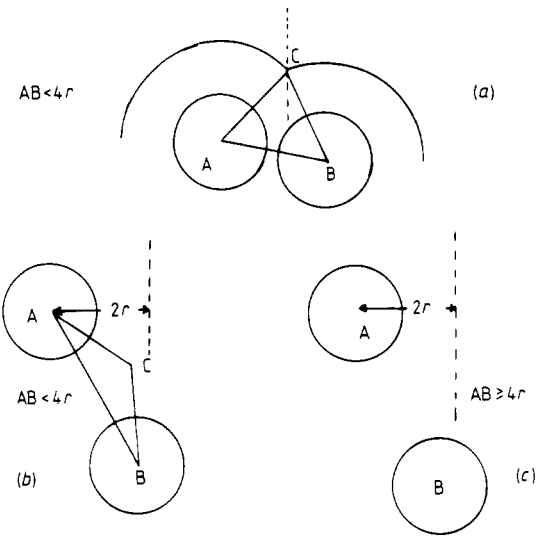

Figure 2. Determination of growth intervals for two surface particles (a) when the triangle rule applies and (b), (c) when the triangle rule does not apply. For details see the text.

that $y_s \geq y_A$. Thus when the vertex of the triangle lies below the horizontal line drawn through the centre of the upper particle, the position of the demarcation line is at a distance $2r$ from the centre of the upper particle. As in figure 2(c) whenever the distance $AB \geq 4r$ the triangle rule does not apply and the demarcation line is at a distance $2r$ from the centre of the upper particle.

Note that the heights of the concerned particles relative to that of the newly added particle can be either positive or negative. The same is true for their relative positions along the x axis (see figure 1).

Counting from the left, consider the ith concerned particle and the ith growth interval corresponding to it. Assuming that this particle lies on the left of the newly added particle, we can find the demarcation line between these two particles by the triangle rule or otherwise. If this demarcation line lies on the left of the RDL of the ith interval then a subsequent particle falling with its centre between this demarcation line and the RDL of the ith interval cannot stick to the ith particle. Hence the new demarcation line becomes the RDL of the ith growth interval. Note that if the new demarcation line lies on the left of the LDL of the ith interval, the ith particle gets completely shadowed and becomes a core particle. If the new demarcation line lies on the right of the ith interval, the ith interval remains unaltered. In any case, the new demarcation line forms the current position of the LDL of the new particle's growth interval, because a particle falling with its centre on the left of it cannot hit the new particle.

For the case where ith particle is on the right of the new particle, the above argument goes through, word for word, provided we interchange the words LDL and RDL and also the phrases 'on the left of' and 'on the right of wherever they occur.

To accomplish step (ii) the procedure described in the above two paragraphs is repeated for $i = 1, 2, \ldots, n_c$, starting from the leftmost concerned particle.

It is easy to see that, when a particle on the left (right) of x_f gets completely shadowed, the demarcation line formed by pairing it with the new particle may lie on the left (right) of the current LDL (RDL) for the new particle. In this case the current

LDL (RDL) of the new particle is not changed. Thus the LDL (RDL) of the new particle is always pushed towards right (left).

Note that we have implicitly made use of the intrinsic ordering of the concerned particles and their growth intervals: for $x_j > x_i$ the demarcation line of the particle at x_j, when paired off with the new particle, lies on the right of the corresponding line for x_i, except when the particle at x_j gets completely shadowed. This completes step (ii).

Steps (i) and (ii) give us a means to construct an aggregate of size $N+1$, given an aggregate of size N.

Using this algorithm we have simulated several BDA clusters typically of the size 2.2×10^5 particles. The largest aggregate generated comprised 4×10^5 particles. We emphasise that these clusters were grown using a computer whose maximum usable core memory was about 80 000 words. Even then the cluster size was limited by the computer time rather than memory. The analysis of the grown clusters is reported in a separate paper [12].

The major advantage of this algorithm is the reduction in the computer memory required. Only the coordinates of the surface particles and the growth intervals need to be stored. This memory requirement is proportional to the radius of the aggregate, say R. In the program written according to the above algorithm, as soon as a new particle is added, its relevant information is written onto a disc. Whenever a surface particle becomes a core particle its information is washed out of the computer memory.

We now analyse the overheads involved in implementing our algorithm. While implementing step (ii) we have to extract the sublists of the concerned particles and their growth intervals from the lists of total surface particles and their growth intervals. Here we need to locate the growth interval corresponding to the positions $x_f + 2r$ and $x_f - 2r$. In our implementation this is done by the binary search method. Hence, on the average $2 \log_2 M$ searches are required to locate both the positions $x_f \pm 2r$, where M is the number of surface particles. Afterwards the list of the growth intervals is reconstructed and the shadowed particles are eliminated. The reconstructed lists are inserted back into the parent lists of growth intervals and surface particles. Due to the reconstruction, either an extra particle is to be inserted in the lists or a few core paticles are to be removed. This involves on average $\frac{1}{2}M$ movements. However, if one divides the lists of surface particles and their growth intervals into a number of 'boxes', each of fixed width $2r$, then obtaining the box corresponding to the value x_f involves a simple formula:

$$n = \lfloor x_f / 2r \rfloor + 1$$

where n is the number of the box relevant to x_f (x_f is now measured from the left edge of the leftmost box) and $\lfloor x \rfloor$ means the greatest integer $\leq x$. Now we need to implement our algorithm on the surface particles and growth intervals corresponding to $(n-1)$th, nth and $(n+1)$th box. This modification in the implementation of our algorithm would drastically reduce the overheads described above and make them essentially independent of M and hence of the size of the aggregate. We feel that with this modification in its implementation our algorithm can become both fast and have minimal storage requirements.

In the conventional algorithm [4, 14] to generate BDA on a seed with single-sided rain one finds all the particles in the cluster whose centres fall in the range $x_f \pm 2r$. Out of these particles the new particle sticks to that particle for which the y coordinate of the new particle (when stuck) is the highest. Obviously this requires the information on a large number of particles (compared to the number of surface particles) to be

stored in the computer memory. Additionally this algorithm requires sorting on x and y coordinates of the cluster particles.

Various generalisations of this algorithm are possible. Generalisation to growth on a line (with rain in one direction) is evident. Particles of different radii can be considered. A multisided rain can also be tackled provided the number of rain directions is discrete and fixed beforehand. It is also possible to grow extended range ballistic aggregates (ERBA) using our algorithm [15].

3. Significance of the triangle rule

The triangle rule used to define growth intervals in this algorithm seems to have a much deeper significance. It embodies the competition for growth between the surface particles. When the triangle rule applies, the growth interval of the upper particle shrinks by a certain amount, which is gained by that of the lower particle. Figures 3(a) and (b) show the triangle and non-triangle situations on the background of the average shape of the top surface [8]. One can easily see that, in the non-triangle case, the local direction of growth deviates by a larger amount compared with the triangle case, from the average growth direction. Thus at the microscopic level the triangle rule seems to be a realisation of the columnar growth.

Analysis of our computer grown aggregates, on the same lines as [9], shows that the limiting semivertical cone angle is about 15.5° [12]. Meakin *et al* [11] also report a very close value (16°) on the basis of their simulations of BDA on a line. However, all of the previous theoretical estimates of this angle lie between 18 and 20° [8, 10, 12]. We can understand this discrepancy on the basis of the triangle rule. Suppose a particle A has stuck to a particle B. If we disregard the triangle rule, then a simple calculation

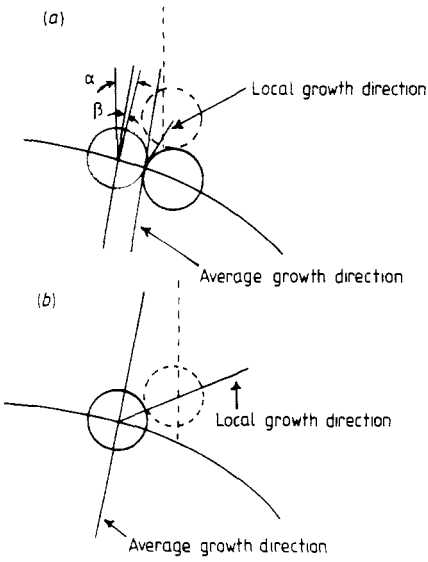

Figure 3. The deviation of the local growth direction from the average growth direction is displayed (a) when the triangle rule applies and (b) when it does not apply. Particles are drawn on the average top curve of the cluster. The broken particle is the falling particle and the broken line is the relevant demarcation line. Here α is the angle between the direction of incidence and the normal to the top curve and β is the column angle.

shows that the average angle made by the particle A with the vertical line drawn through B is 32.5°. This corresponds to an average increase in height per particle of $\frac{1}{2}\pi r$. Whenever the triangle rule applies, it eliminates the possibility of A sticking to B making an angle larger than a certain value: A sticks to the other competing surface particle instead, making a smaller angle with the vertical line. Thus if the triangle rule applies for the major fraction of events, the probability that a particle sticks to a previous particle making large angles with the vertical line is reduced as compared to the situation when the triangle rule is disregarded. This implies a smaller average angle made by A with the vertical line through B and a larger average height increase per particle. This in turn will mean a smaller semivertical fan angle. In order to check this argument, we estimated from our grown aggregates the angle with the vertical line made by the chains of particles which grow outwards and terminate at the edges of the fan. This average angle turned out to be 28° [12]. This is less than 32.5°, showing that the average increase in height is greater than $\frac{1}{2}\pi r$. This clearly indicates the important role played by the triangle rule in the BDA process.

We now give an argument to show that the triangle rule applies for a large fraction of surface particles. The triangle rule applies whenever the line joining the competing surface particles makes an angle with the horizontal which is less than the base angle of the triangle. This angle is determined by the distance between the two surface particles, say D and is given by $\cos^{-1}(D/4r)$ where $2r \leqslant D < 4r$. For $D = 2r$, this critical angle is $\frac{1}{3}\pi$. Imagine now that the average top curve (see figure 3) is covered by adjacent surface particles. Then the angle made by the line joining two adjacent particles with the x axis can be taken to be the angle (with the x axis) made by the tangent to the curve at the point of contact of the two particles. It is easy to see that this angle is the same as α as defined in figure 3. The triangle rule applies to all the cases corresponding to $\alpha < \frac{1}{3}\pi$. The angle α is related to the fan angle θ by [8]

$$\theta = \tan^{-1}[\tan\alpha/(2+\tan^2\alpha)].$$

For $\alpha = \frac{1}{3}\pi$, $\theta = 29°$ which is greater than the observed limiting fan angle [11, 12]. In reality, fluctuations in the positions of surface particles around the average top curve tend to reduce this value of θ. At any rate, the triangle rule should apply to a large fraction of surface particles. The above arguments indicate that the triangle rule embodies a microscopic process which results in macroscopic effects, like the columnar structures observed in BDA.

Acknowledgment

The author would like to thank Professor A P B Sinha for many useful discussions and encouragement.

Refrences

[1] Herrmann H J 1986 *Phys. Rep.* **136** 153
[2] Pietronero L and Tosatti E (ed) 1986 *Fractals in Physics* (Amsterdam: North-Holland)
[3] Witten T A and Sander L M 1981 *Phys. Rev. Lett.* **47** 1400; 1983 *Phys. Rev.* B **27** 5686
[4] Vold M J 1963 *J. Colloid. Sci.* **18** 684
Hutchinson H P and Sutherland D N 1985 *Nature* **206** 1036
Sutherland D N 1966 *J. Colloid. Interface Sci.* **22** 300; 1967 *J. Colloid. Interface Sci.* **25** 373; 1970 *Nature* **226** 1241
Sutherland D N and Goodarz-Nia I 1971 *Chem. Eng. Sci.* **26** 2071

[5] Meakin P 1983 *Phys. Rev.* B **28** 5221
[6] Ball R C and Witten T A 1984 *Phys. Rev.* A **29** 2966
[7] Bensimon D, Shraiman B and Liang S 1984 *Phys. Lett.* **102A** 238
[8] Ramanlal P and Sander L M 1985 *Phys. Rev. Lett.* **54** 1828
[9] Liang S and Kadanoff L P 1985 *Phys. Rev.* A **31** 2628
[10] Limaye A V and Amritkar R E 1986 *Phys. Rev.* A **34** 5085
[11] Meakin P, Ramanlal P, Sander L M and Ball R C 1986 *Phys. Rev.* A **34** 5091
[12] Joag P S, Limaye A V and Amritkar R E 1987 *Phys. Rev.* A to be published
[13] Leamy H J, Gilmer G H and Dirks A G 1980 *Current Topics in Materials Science* vol 6, ed E Kaldis (Amsterdam: North-Holland)
[14] Meakin P 1985 *J. Colloid. Interface Sci.* **105** 240
[15] Gelband P and Strenski P N 1985 *J. Phys. A: Math. Gen.* **18** 611