

ЛАБОРАТОРНА РОБОТА №5

ФРАКТАЛЬНІ КЛАСТЕРИ

Цель: построение фрактальных кластеров с помощью модели ограниченной диффузией агрегации.

Под массовыми фракталами понимают моделируемые и реальные объекты, состоящие из дискретного набора частиц-мономеров, агрегированных в большие кластеры с фрактальной геометрией. Часто при исследовании таких объектов прибегают к анализу пространственного распределения плотности вещества, или, другими словами, распределению массы объекта. Отсюда и название – массовые фракталы.

Модель ограниченной диффузией агрегации (ОДА) является наиболее популярным подходом для описания образования массовых фракталов. Эта модель может быть легко реализована в виде компьютерного алгоритма и прекрасно подходит для численного исследования фрактальной геометрии таких объектов.

Существует множество вариантов модели ОДА, различающихся геометрией задачи, особенностями взаимодействия частиц и кластеров и их перемещения в пространстве. В рамках данного пособия мы ограничимся рассмотрением самого простого классического варианта ОДА – модели частица-кластер. В этой модели растущий кластер всегда неподвижен в пространстве, а присоединяющиеся к нему частицы-мономеры передвигаются в пространстве по случайным траекториям независимо друг от друга до момента соприкосновения с кластером, после чего они становятся частью кластера.

Рассмотрим эту задачу более подробно. В качестве зародыша будущего кластера выступает частица-мономер, к которой в последующем будут присоединяться другие частицы. Частицы запускаются по одной с произвольной случайно выбранной точки на окружности (в 2D задаче) или сфере (в 3D задаче) заданного радиуса R_0 , построенной вокруг центрального зародыша. Далее пошагово моделируется случайное броуновское движение частицы в пространстве – на каждом шагу частица случайным образом перемещается в один из ближайших соседних узлов сетки, на которой происходит моделирование. В случае, если происходит касание частицей кластера, она становится частью кластера и осуществляется запуск новой частицы. Если в процессе своего движения частица удаляется от центра кластера на расстояние, превышающее $3R_0$, то эта частица исключается из рассмотрения и запускается следующая. Обычно задача останавливается, когда кластер достигает либо определенной заданной массы (общего количества частиц, составляющих кластер), либо когда максимально удаленная точка кластера выходит за некоторый заданный радиус $R < R_0$.

Результат такого моделирования на 2D квадратной решетке представлен на Рис. 18(а). Образовавшийся кластер имеет довольно сложную дендритную структуру и характеризуется фрактальной размерностью $D \approx 1,7$.

Необходимо отметить, что фрактальная размерность для данной задачи зависит от типа выбранной решетки. Чтобы избежать этой проблемы, моделирование предпочтительнее проводить "вне решетки", позволяя частицам перемещаться в произвольных направлениях и присоединяться к кластеру в произвольных местах. Такое моделирование, конечно же, является более сложной задачей, но получающиеся кластеры оказываются более реалистичными. Геометрию задачи можно изменить, если рост кластеров производить не из центральной точки, а с некоторой линии (в 2D задаче) или плоскости (в 3D задаче), которые являются дном некоторого сосуда. Частицы при этом запускаются из верхней части сосуда и в процессе своего хаотичного движения отражаются от боковых стенок и крышки сосуда, но оседают на его дне и растущих кластерах. В результате происходит рост "деревьев" на дне сосуда, как показано на Рис. 18(б).

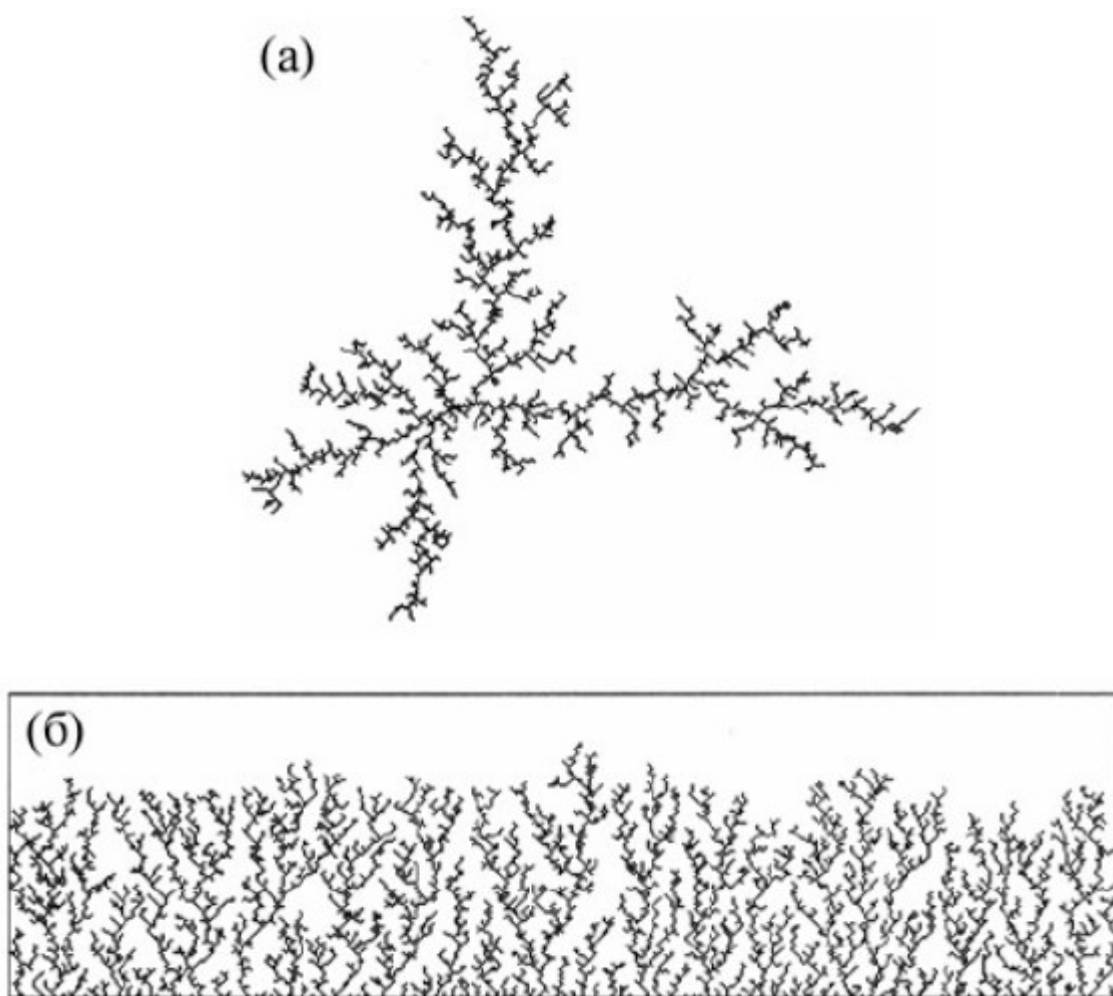


Рис. 18. Моделирование роста фрактальных кластеров с помощью алгоритма ОДА на 2D квадратной решетке: (а) рост из центральной точки, (б) рост от линии

Задачу можно усложнить и сделать более интересной, если ввести вероятность p прилипания частицы к кластеру. При этом каждый раз при соприкосновении свободной частицы с кластером производится проверка условия прилипания - величина p сравнивается со случайным числом $r \in [0,1]$. Если $p \leq r$, то частица присоединяется к кластеру и запускается новая, а иначе частица продолжает свое блуждание. Понятно, что чем меньше вероятность прилипания, тем больше у частицы шансов проникнуть вглубь кластера между его ветвями, тем более компактным оказывается его структура и тем больше его фрактальная размерность. В пределе, размерность кластера оказывается близка к евклидовой размерности: $D \rightarrow 2$ для 2D задачи и $D \rightarrow 3$ для 3D задачи.

Листинг dla.m

```
function z = DLA003(filename, dim , N)
% Diffusion Limited Aggregation (DLA) version 0.03
% by Jonathan Leto <jonathan@leto.net>
% Tested in Matlab 6.0 and 7.1
% Sat Jan 06 00:27:21 EST 2007
% Arguments:
% filename - a string for where to save the file Ex:
'pic10.jpg'
% dim - create a (dim x dim)
space
% N - number of particles
% New in version 0.03:
% Reseed speedups
% New in version 0.02:
% Faster, implemented:
% Introduce points at a radius a little bigger than
structure ( moderate speedup )
% Reseed points if they wander out of radius ( huge
speedup! )
% TODO: Implement stickiness
started=datestr(now);
step = 3;
randtheta = rand*2*pi;
middle = round(dim/2);
pic = 127*ones(dim,dim);
pic(middle,middle) = 0;
c = 0;
dist = 0;
min_radius = 5;
reseed=0;
num_reseeds=0;
stat_total_sum = 0;
stat_total_avg = 0;
% Let's simulate a drunken sailor
for n=1:N,
% TODO: make this an adjustable parameter
% this makes the initial radius %1 of dim, and linearly
increases it until r=dim/2 at n=N
% TODO: if dim is >=1000, than the initial radius should
be less than %1
radius = round(dim*(1 + 49*(n-1)/N)/100);
if( radius < min_radius )
radius = min_radius;
end
randtheta = rand*2*pi;
randx = round( middle + (radius-2) * cos(randtheta) );
```

```

randy = round( middle + (radius-2) * sin(randtheta) );
dist = sqrt((randx-middle)^2 + (randy-middle)^2);
disp( sprintf('%d: init=(%d,%d) dist=%f radius=%d',
n,randx,randy,dist,radius) );
% repeat until we hit something
% if outside radius after 20 steps, RESEED
% TODO: make # of steps adjustable, or find a better
default
while ( 1 > 0 ),
c=c+1;
newx = randx + (-1)^round(4*rand) *
mod( round( rand*(step*2) ), step ) ;
newy = randy + (-1)^round(4*rand) *
mod( round( rand*(step*2) ), step ) ;
if( mod(c,20) == 0 )
if( abs( newx - middle ) > radius | abs( newy -
middle) > radius )
disp('RESEED');
reseed=1;
num_reseeds=num_reseeds+1;
n=n-1;
break;
end
end
if( newx > 1 & newx < (dim-1) )
randx = newx;
end
if( newy > 1 & newy < (dim-1) )
randy = newy;
end
% Did we walk into something?
found = 0;
for j=-1:1,
for k=-1:1,
if( randx+j < 1 | randy+k < 1)
break;
end
p = pic(randx+j,randy+k);
% Hit something!
if ( p == 0 )
disp(sprintf('%d: hit at (%d,%d)', n, randx+j,
randy+k) );
found = 1;
break;
end
end
if ( found )
break;

```

```
end
end
if( found )
break;
end
end
% we have hit something if reseed=0
if( reseed == 1 )
% reset seed state
reseed=0;
else
% count total number of iterations
stat_total_sum = stat_total_sum + c;
disp(sprintf('%d: took %d attempts to hit',n,c));
c=0;
% aggregate
pic(randx,randy) = 0;
end
end
finished=datestr(now);
imwrite(pic,filename);
image(pic);
colormap gray;
disp(sprintf('Started %s', started) );
disp(sprintf('Finished %s', finished) );
disp(sprintf('Avg # attempts: %d',
round(stat_total_sum/N )) );
disp(sprintf('# of Reseeds : %d', num_reseeds ) );
```

ЗАДАНИЕ

1. Изучить основные алгоритмы реализации модели ОДА.
2. Модифицировать программу моделирования фрактальных кластеров в системе SciLab (или другой выбранной студентом), чтобы получить кластер, изображённый на рис. 18(б).
3. Реализовать программу в системах MAXIMA и MathCAD.