Отчет по групповому проекту. Этап 4

Неравновесная агрегация, фракталы

Шалыгин Г. Э.

Низамова А. А.

Голощапова И. Б.

Серегин Д. А.

Пиняева А. А

11 марта 2023

Содержание

# 1 Цель работы

Цель:

* Реализовать алгоритмы моделирования неравновесной агрегации.

Задачи:

* Рассмотреть возможности языков для программной реализации алгоритмов
* Реализовать алгоритмы, описанные втором этапе

# 2 Теоретическое введение

## 2.1 Постановка задачи

Существуют разнообразные физические процессы, основная черта которых — неравновесная агрегация. Примеры: образование частиц сажи, рост осадков при электрическом осаждении и т. п. Во всех случаях происходит необратимое прилипание частиц к растущему кластеру из-за сильного смещения равновесия в сторону твердой фазы, и вырастают разветвленные агрегаты (рост правильных ограненных кристаллов происходит в условиях, близких к равновесным, когда возможно как прилипание частиц, так и их обратный переход в раствор)

## 2.2 Случайные блуждания

### 2.2.1 Одномерные случайные блуждания

Рассмотрим простую модель — пусть частицы могут двигаться только вдоль прямой, делая шаги случайной длины. Примем также, что величина и направление каждого шага определяются независимо от положения частицы и от предыдущих шагов (модель «пьяного моряка»). Будем наблюдать за частицей через равные промежутки времени. Координата частицы вычисляется по рекуррентной формуле , где — очередной шаг блуждания. При отсутствии силового поля смещение влево и вправо равновероятны. В общем случае вероятность того, что длина шага лежит в промежутке от #$ до , равна . Функция w называется плотностью вероятности для величины шага δ, который называют случайной величиной.

Подробнее в [**qq?**].

### 2.2.2 Многомерные случайные блуждания

Предоставим теперь нашим частицам возможность двигаться также по координате y — рассмотрим двумерные случайные блуждания (случай трех измерений получается аналогично). Можно независимо задавать смещение по вертикали равномерно распределенным таким же образом, как и смещение по горизонтали .

Функция распределения в двумерном случае представляется в виде произведения двух функций распределения по координатам x и y , так как x и y являются независимыми случайными величинами. Подробнее в [**qq?**].

## 2.3 Фрактальная размерность

Фигура на плоскости или тело в пространстве имеют размерность. Определить ее можно разными способами. В случае, когда у фигуры есть выделенная центральная точка, можно построить много сфер различного радиуса с центром в этой точке. Для каждой сферы можно вычислить массу части фигуры, которая оказалась внутри этой сферы. В случае, когда масса пропорциональна радиусу сферы в некоторой степени (), показатель степени D называется размерностью тела. Это так называемый метод сфер или ящиков. Для линий D = 1, у плоских фигур D = 2, у «обычных» тел D = 3. Однако, многие объекты в природе имеют размерность, выражающуюся дробным числом.

Такие тела, следуя Б. Мандельброту [9], называют фракталами (от латинского слова fractus — дробный). Фракталами являются также дендриты, вырастающие при электроосаждении металлов, кластеры, полученные при агрегации коллоидов; фрактальную структуру имеют ветви деревьев, кровеносная система и т. п.

Еще один способ может применяться при наблюдении за процессом роста агрегата от центра. В этом случае число частиц в кластере . В качестве характерного радиуса Rch можно выбирать, например, максимальный радиус кластера ,

Желающим более подробно познакомиться с фракталами рекомендуем книгу [**ww?**].

# 3 Математические модели

## 3.1 Агрегация, ограниченная диффузией

### 3.1.1 Сеточная модель.

Возьмем регулярную сетку на плоскости, например, квадратную. В центр поместим затравочную частицу. Затем с достаточно большого расстояния будем выпускать по одной новые частицы. Выпущенная частица совершает случайные блуждания по сетке, делая шаги в одном из четырех доступных направлений с равной вероятностью. Если частица оказывается по соседству с затравкой, она прилипает и остается в этом узле. Затем выпускаем следующую частицу, которая может прилипнуть к одному из занятых узлов. Шаг решетки в этой модели соответствует длине связи между частицами (расстоянию устойчивого равновесия для взаимодействия двух частиц).

*Некоторые указания.* Для ускорения работы программы разумно выпускать частицы с круга радиусом немного больше Rmax — текущего максимального радиуса агрегата.

Если частица уходит далеко, уничтожаем ее и выпускаем новую.

### 3.1.2 Бессеточная модель.

Структура полученных DLA-кластеров отражает структуру сетки (имеются выделенные направления). Чтобы получить более симметричные кластеры, можно отказаться от сетки. В этом случае рост происходит следующим образом: вначале помещаем в центр поля затравочную частицу, затем с круга некоторого радиуса выпускаем следующую, которая случайно блуждает. Если частицы сближаются на расстояние взаимодействия (например, их удвоенный радиус), они слипаются. После этого выпускаем новую частицу и т. д.

Детальная информация в [**mod?**]

### 3.1.3 Химически-ограниченная агрегация

При диффузионно-ограниченной агрегации частица всегда прилипает к кластеру с вероятностью 1. Можно уменьшить вероятность прилипания. Такой процесс роста называется химически-ограниченной агрегацией. Он моделирует ситуацию, когда вероятность зависит от того, каким концом молекула повернута к другой. Это приведет к появлению более плотных агрегатов (увеличению размерности), потому что у частицы увеличится шанс проникать во внутренние области и заполнять пустоты. Размерность, однако, остается меньше размерности пространства, т. е. кластер остается фракталом.

### 3.1.4 Баллистическая агрегация

До сих пор мы рассматривали рост кластеров с точечной затравки. Однако, довольно часто встречаются ситуации, когда агрегаты растут на поверхности, например, при выпадении осадка на дне или стенках сосуда. Если новые частицы доставляются к растущему кластеру за счет диффузии, имеем просто модель DLA с измененными начальными условиями.

Другой случай — *баллистическая агрегация*, при которой частицы свободно падают по прямолинейным траекториям. Частица прилипает, когда оказывается рядом с занятым узлом. В этом процессе получается более плотный агрегат (но не сплошной), однако его граница сильно изрезана и является фракталом.

# 4 Прогамная реализация

## 4.1 Используемые возможности языка Julia

Округление: - round(x) округление к ближайшему целому числуб typeof(x) - round(T, x) округление к ближайшему целому числу T - floor(x) округление x к -Inf typeof(x) - floor(T, x) округление x к -Inf T - ceil(x) округление x к +Inf typeof(x) - ceil(T, x) округление x к +Inf T - trunc(x) округление x по направлению к нулю typeof(x) - trunc(T, x) округление x по направлению к нулю T

Математические операции: - abs(x) модуль - sqrt(x), квадратный корень x - sin(x), cos(x) - maximum(X), minimum(X) - максимум и минимум массивов

Макрос @time позволяет измерить время работы функции. Подробное описание в [**julia?**]

## 4.2 Листинги используемых функций для модели DLA

### 4.2.1 Генерация псевдослучайных чисел

next = 0  
function rand()  
 global next = (next \* 1664525 + 1013904223) % 2^32  
 return next / 2 ^ 32  
end

### 4.2.2 Генерация координат следующей частицы

function GetNextParticular(x\_center, y\_center, r)  
 r = r  
 angle = 2 \* pi \* rand()  
 x = r \* cos(angle) + x\_center  
 y = r \* sin(angle) + y\_center  
 return round(x), round(y)  
end

### 4.2.3 Дополнительные функции

**Расстояние между точками**

function dist(x1, y1, x2, y2)  
 return sqrt((x2-x1)\*(x2-x1) + (y2-y1)\*(y2-y1))  
end

**Проверка того, что частица столкнулась с кластером**

function check(x, y)  
 for i in 1:n  
 if abs(X[i] - x) + abs(Y[i] - y) == 1  
 return true  
 end  
 end  
 return false

### 4.2.4 Блуждание частицы

function RandomWalk(x, y, i, r, xl, xr, yu, yd)  
 step = 1; dx = [1, -1, 0, 0]; dy = [0, 0, 1, -1]  
 while step < 500 && dist(x, y, (xl + xr) / 2, (yu+ yd) / 2) < 4 \* r  
 if check(x, y)  
 X[i] = x; Y[i] = y  
 return true  
 end  
 j = floor(Int, 100 \* rand()) % 4 + 1  
 x = x + dx[j]  
 y = y + dy[j]  
 step += 1  
 end  
 return false  
end

### 4.2.5 Псевдокод модели DLA

function DLA(t)  
 i = 1  
 while i < t  
 xl = minimum(X)  
 xr = maximum(X)  
 yu = minimum(Y)  
 yd = maximum(Y)  
 r = dist(xl, yd, xr, yu) / 2 + 3  
 x, y = GetNextParticular((xr+xl)/2, (yu+yd)/2, r)  
 ok = RandomWalk(x, y, i, r, xl, xr, yu, yd)  
 if ok  
 i += 1  
 end  
 end  
end

## 4.3 Другие модели

* **Бессеточная**: добавляется выбор случайной длины шага.
* **Химически-ограниченная**: вводится условие прилипания.

### 4.3.1 Бессеточная

function RandomWalk(x, y, i, r, xl, xr, yu, yd)  
 <...>  
 x = x + 2 \* rand() - 1  
 y = y + 2 \* rand() - 1  
 <...>  
end

### 4.3.2 Химически-ограниченная

function RandomWalk(x, y, i, r, xl, xr, yu, yd)  
 <...>  
 if check(x, y) and random() > 0.2  
 X[i] = x  
 Y[i] = y  
 <...>  
end

# 5 Анализ результатов

## 5.1 Сеточная модель

### 5.1.1 Кластеры

Рассмотрим результаты работы кода для разных значений числа частиц .

|  | fig: | fig: | fig: |
| --- | --- | --- | --- |
|  | fig: | fig: | fig: |

|  | fig: | fig: | fig: |
| --- | --- | --- | --- |
|  | fig: | fig: | fig: |

Видно, что с увеличением  получающийся кластер стремится к агрегату наблюдаемому в физических явлениях с неравновесной агрегацией.

## 5.2 Время выполнения, с

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 50 | 0.24 | 0.20 | 0.18 | 0.21 |
| 100 | 0.65 | 0.82 | 0.66 | 0.71 |
| 200 | 4.57 | 4.48 | 3.72 | 4.26 |
| 500 | 44.66 | 52.40 | 41.20 | 46.09 |

# 6 Бессеточная модель

## 6.1 Кластеры

|  | fig: | fig: | fig: |
| --- | --- | --- | --- |
|  | fig: | fig: | fig: |

## 6.2 Кластеры

|  | fig: | fig: | fig: |
| --- | --- | --- | --- |
|  | fig: | fig: | fig: |

Получившиеся кластеры имеют бессеточную структуру, приближённую к реальным агрегатам.

## 6.3 Время выполнения, с

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 50 | 0.22 | 0.21 | 0.23 | 0.22 |
| 100 | 0.82 | 0.81 | 0.64 | 0.76 |
| 200 | 3.24 | 4.07 | 3.36 | 3.56 |
| 500 | 45.22 | 33.48 | 34.98 | 37.99 |

Построим график зависимости времени выполнения от числа частиц.

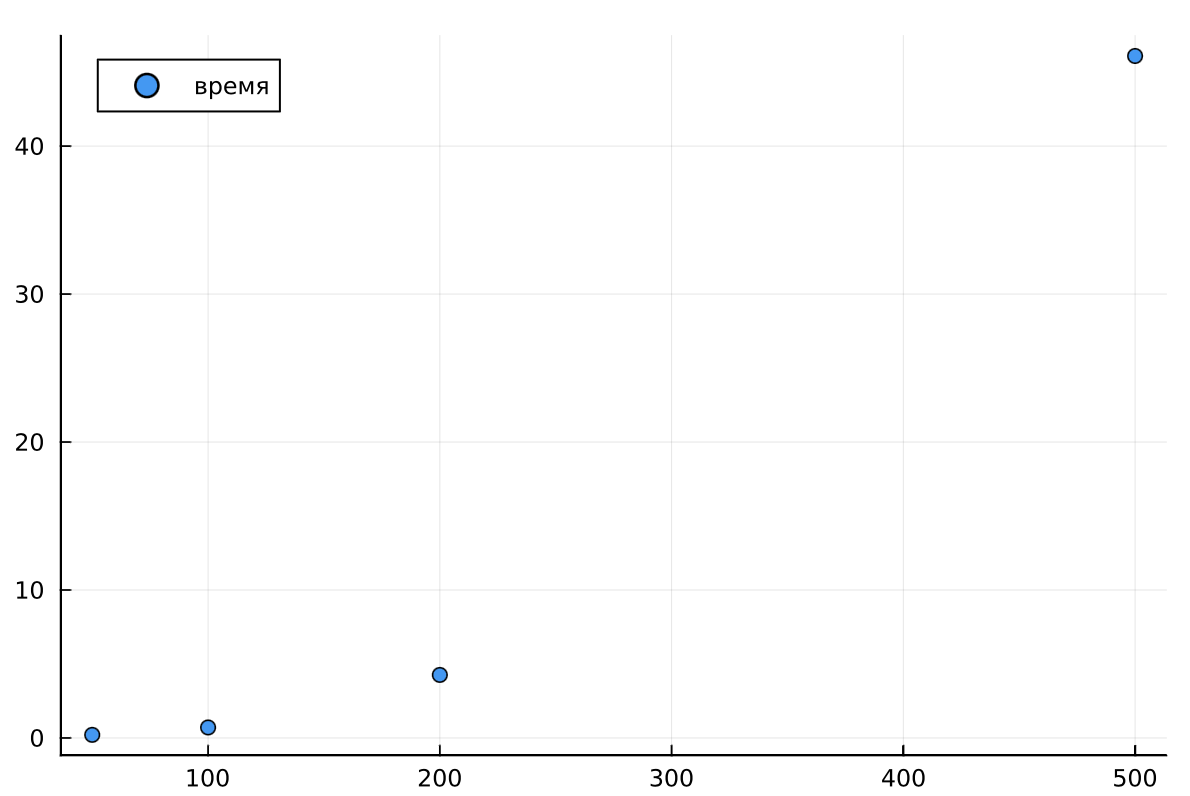


График роста времени

Таким образом время возрастает нелинейно, а полиномиально или даже экспоненциально.

# 7 Фрактальная размерность

|  | Сеточная | Бессеточная |
| --- | --- | --- |
| 50 | 1.26 | 1.40 |
| 100 | 1.41 | 1.49 |
| 200 | 1.58 | 1.59 |
| 500 | 1.69 | 1.70 |
|  | **1.49** | **1.55** |

С ростом числа частиц фрактальная размерност должна стремиться к 1.71, что можно увидеть в результате вычислительных экспериментов. ы

# 8 Выводы

* Создан инструмент проведения вычислительынй экспериментов для изучения моделей неравновесной агрегации.
* Полученные результаты согласуются с эмпирическими данными.

# Список литературы