Групповой проект. Этап 4

Результаты проекта. Самооценка деятельности

Александрова Ульяна Вадимовна Волгин Иван Алексеевич Голощапов Ярослав Вячеславович Дворкина Ева Владимировна Серёгина Ирина Андреевна Чемоданова Ангелина Александровна

RUDN University, Moscow, Russian Federation

Информация

Докладчик

- Серегина Ирина Сергеевна
- студентка 3-го курса учебной группы НФИбд-01-22
- Российский университет дружбы народов
- https://github.com/irinaseregina

Докладчик

- Волгин Иван Алексеевич
- студент 3-го курса учебной группы НФИбд-01-22
- Российский университет дружбы народов
- https://github.com/VolginIA

Введение

Актуальность

Появление дендритов играет ключевую роль в металлургии и в литейном производстве, особенно при затвердевании металлов и сплавов.

Объект и предмет исследования:

- Дендриты
- Кристаллические дендриты

Цель:

- Исследовать модель роста дендритов.
- Описать алгоритм решения задачи моделирования роста дендритов.
- Реализовать модель роста дендритов и проанализировать результаты.

Задачи

- Рассмотреть модель роста дендритов.
- Рассмотреть алгоритм построения модели.
- Описать основные этапы алгоритма.
- Написать программу для моделирования.
- Проанализировать результаты.

Рис. 1: Дендритная кристаллизация после плавления внутри герметичных ампул из металлического рубидия и цезия

Теоретическое описание задачи.

Образование дендритов

$$S = c_p \frac{(T_m - T_\infty)}{L} \tag{1}$$

Уравнение теплопроводности

$$\rho c_p \frac{\partial T}{\partial t} = \kappa \nabla^2 T \equiv \kappa \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) \tag{2}$$

Условие Стефана

$$\mathbf{n} \cdot \mathbf{V} = \frac{\kappa}{\rho L}(\mathbf{n} \cdot \nabla T|_s - \mathbf{n} \cdot \nabla T|_l) \tag{3}$$

Условие Гиббса-Томсона

$$T_b = T_m \left(1 - \frac{\gamma T_m}{\rho L^2 R}\right) \tag{4}$$

Безразмерная температура и уравнение теплопроводности

Вводится безразмерная температура $\tilde{T} = c_p(T - T_\infty)/L$, где T_∞ – начальная температура расплава. Уравнение теплопроводности для $\tilde{T}$ имеет вид

$$\frac{\partial \tilde{T}}{\partial t} = \chi \nabla^2 \tilde{T}, \tag{5}$$

где $\chi = \kappa/\rho c_p$ – коэффициент температуропроводности.

Описание модели

Изменение температуры

Новое значение температуры после шага m вычисляется по формуле

$$\hat{T}_{i,j} = T_{i,j} + \frac{\chi \Delta t \nabla^2 T}{m}. \tag{6}$$

Рост дендрита

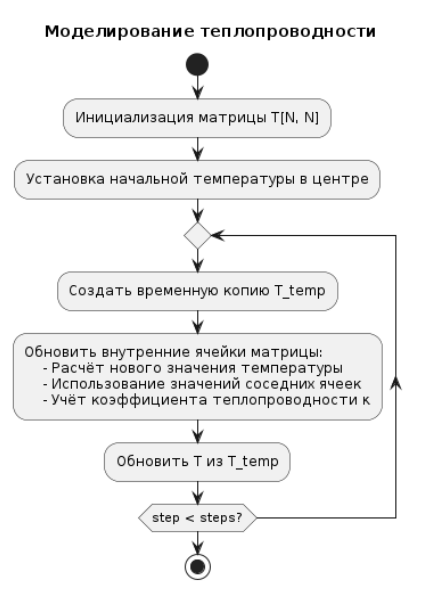
$$T \le \tilde{T}_m (1 + \eta_{i,j}\delta) + \lambda s_{i,j}, \tag{7}$$

Учет кривизны границы

Кривизна границы $1/R$ приближенно вычисляется по соседям узла:

$$1/R \approx s_{i,j} = \sum_1 n_{i,j} + w_n \sum_2 n_{i,j} - \left(\frac{5}{2} + \frac{5}{2} w_n\right), \tag{8}$$

Описание алгоритма

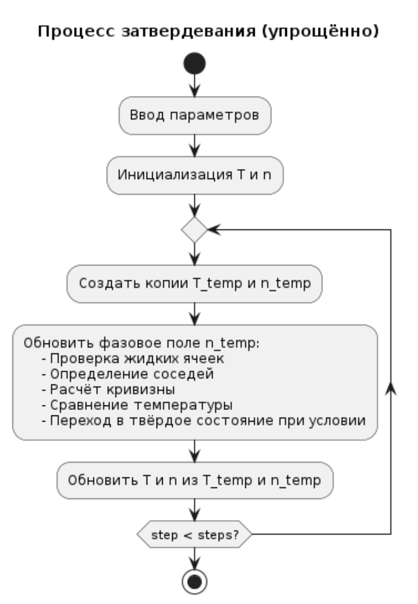


Шаг 1: Задание параметров

Шаг 2: Настройка симуляционной сетки

Шаг 3: Расчет температурного поля

- Применение уравнения теплопроводности,
- Численная реализация уравнения



Шаг 4: Моделирование роста дендритов

Реализуется моделирование роста дендритов, основываясь на рассчитанных температурных полях и соответствующих физических законах.

Определение фрактальной размерности

$$N(r) \sim r^D \tag{9}$$

где $N(r)$ – число точек внутри круга радиуса r.

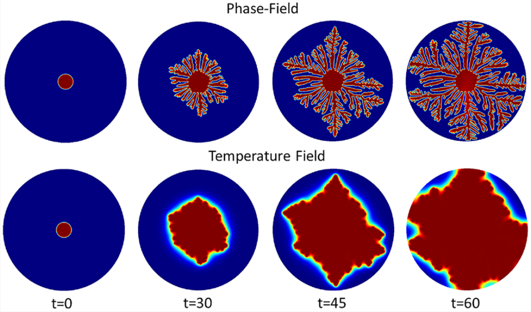


Рис. 4: Фазовое и температурное поле при росте дендрита

Практическая часть

Задание базовых параметров модели

```
using Plots, LinearAlgebra, Statistics

# Параметры модели
N = 150          # Размер сетки (N x N)
T_initial = -1 # Начальная температура в центральной точке
steps = 200     # Количество временных шагов
dt = 1          # Шаг по времени
h = 1           # Расстояние между узлами
kappa = 0.1         # Коэффициент теплопроводности
w = 0.5         # Коэффициент для диагональных соседей
T_m = 0         # Температура плавления
$\lambda$ = 0.01         # Капиллярный радиус
$\delta$ = 0.02        # Величина флуктуаций температуры
```

Инициализация сетки

```
# Инициализация сетки
T = zeros(N, N)              # Матрица температур
n = zeros(Int, N, N)         # Матрица состояний (0 - жидкое, 1 - твердое)
T[N÷2+1, N÷2+1] = T_initial  # Установка начальной температуры в центральной
n[N÷2+1, N÷2+1] = 1
```

Базовые функции

- Метод полиномиальной аппроксимации
- Среднее значение температуры
- Кривизна границы
- Количества затвердевших частиц
- Среднеквадратичный радиус

```
function curvature(n, i, j, w)
    horizontal_vertical_neighbors = [
        n[i-1, j], n[i+1, j], n[i, j-1], n[i, j+1]
    ]
    diagonal_neighbors = [
        n[i-1, j-1], n[i-1, j+1], n[i+1, j-1], n[i+1, j+1]
    ]
    sum_hv = sum(horizontal_vertical_neighbors)
```

Рис. 5: Функция curvate

Модель теплопроводности

Функция `simulate_heat_conduction` на основе уравнения обновления температуры:

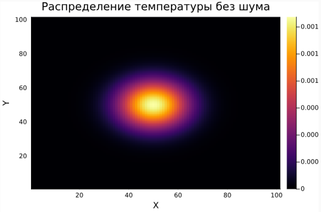


Рис. 6: Распределение температуры без шума

```julia
function simulate_heat_conduction(N, steps, κ)
    T = zeros(N, N)
    center = div(N, 2)
    T[center, center] = 1.0  # начальная температура в центре

    for step in 1:steps
        T_temp = copy(T)
        for i in 2:N-1
            for j in 2:N-1
                T_temp[i, j] = T[i, j] + κ * (T[i+1, j] + T[i-1, j] + T[i, j+1] + T[i, j-1] - 4 * T[i, j])
            end
        end
        T .= T_temp
    end

    heatmap(T, title="Распределение температуры без шума", xlabel="X", ylabel="Y")
end
```

Рис. 7: Функция simulate_heat_conduction

Добавление процесса затвердевания

Реализована функция `simulate_solidification`, которая выполняет следующие шаги:

1. Обновление температур
2. Проверка условия затвердевания
3. Обновление состояний

```
for i in 2:size(n, 1)-1
    for j in 2:size(n, 2)-1
        if n[i, j] == 0  # Только для жидких узлов
            # Проверяем наличие соседей в твердой фазе
            neighbors = [n[i-1, j], n[i+1, j], n[i, j-1], n[i, j+1],
                         n[i-1, j-1], n[i-1, j+1], n[i+1, j-1], n[i+1, j+1]]
            if any(neighbors .== 1)  # Если есть хотя бы один твердый сосед
                # Вычисляем кривизну границы
                s_ij = curvature(n, i, j, w)

                # Вычисляем локальную температуру плавления
                local_T_m = T_m + λ * s_ij

                # Проверяем условие затвердевания
                if T_temp[i, j] <= local_T_m
                    n_temp[i, j] = 1  # Узел затвердевает
                end
            end
        end
    end
end

# Обновляем основные матрицы
T .= T_temp
n .= n_temp
```

Рис. 8: Фрагмент функции simulate_solidification

Результаты моделирования. Исследование влияния капиллярного радиуса

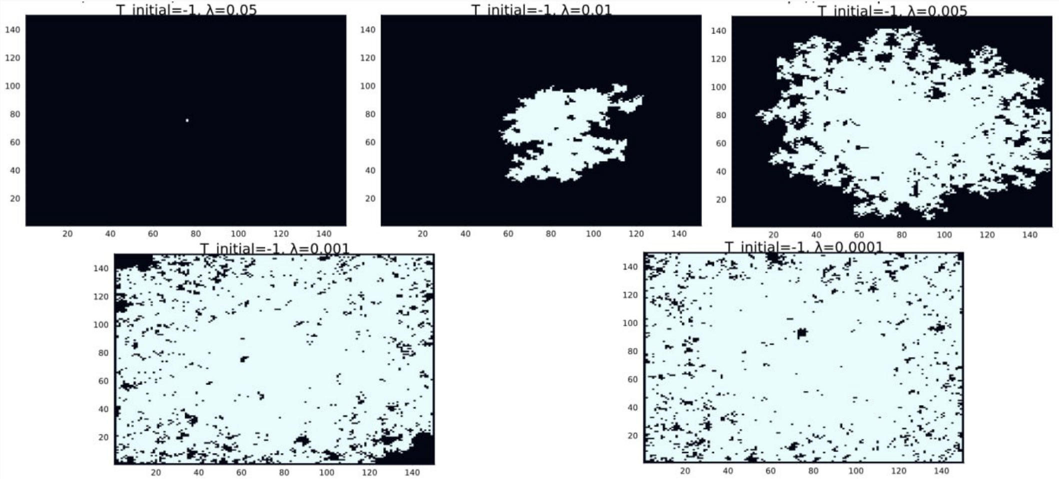


Рис. 9: Исследование влияния начального переохлаждения и величины капиллярного радиуса

Динамика роста агрегата

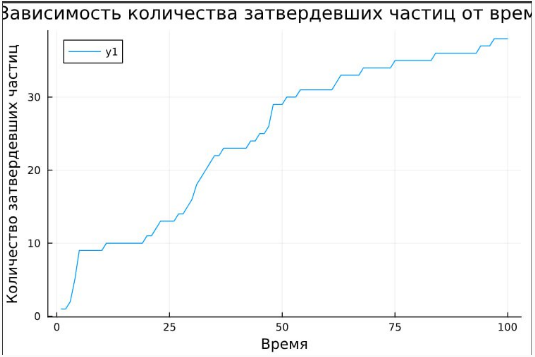


Рис. 10: Зависимость числа затвердевших частиц от времени

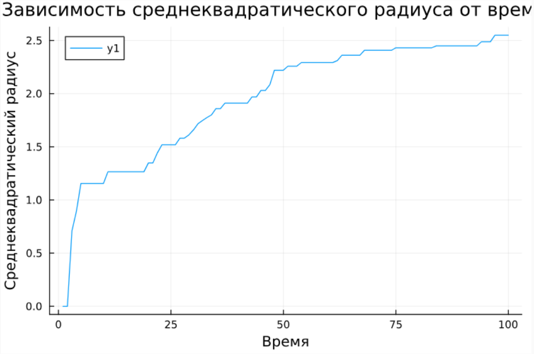


Рис. 11: Зависимость среднеквадратического радиуса от времени

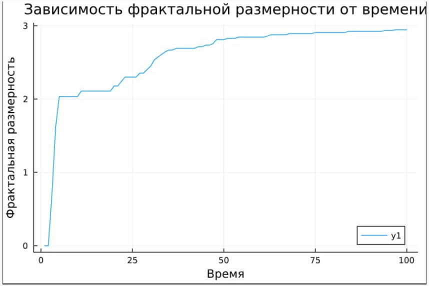


Рис. 12: Зависимость фрактальной размерности от времени

Фрактальную размерность D можно определить через логарифмическую регрессию:

$$D = \frac{\log N(r)}{\log r} \tag{10}$$

где:

- $N(r)$ - количество частиц внутри радиуса r
- D - искомая фрактальная размерность

Реализована функция `fractal_dimension`

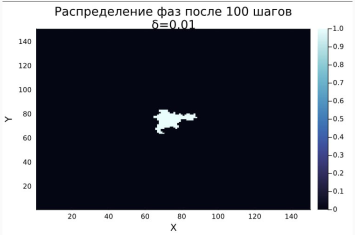


Рис. 13: Значение теплового шума (δ) 0.01

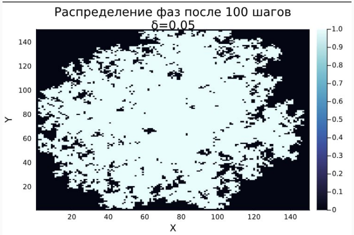


Рис. 14: Значение теплового шума (δ) 0.05

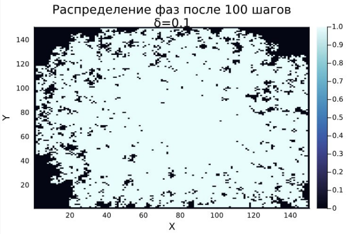


Рис. 15: Значение теплового шума (δ) 0.1

Выводы

Выводы

Во время выполнения группового проекта мы:

- сделали теоретическое описание модели роста дендритов и определили задачи дальнейшего исследования,
- описали процесс создания алгоритма для моделирования роста дендритов, включающий все ключевые этапы
- смоделировали процесс теплопроводности.
- исследовали влияние начального переохлаждения и капиллярного радиуса на форму дендритов,
- проанализировали динамика роста агрегата и его фрактальная размерность,
- изучили влияние теплового шума на морфологию агрегатов.