

Групповой проект. Этап 2

Рост дендритов

Доборщук Владимир, НФИбд-01-18

Голова Варвара, НФИбд-03-18

Дяченко Злата, НФИбд-03-18

Карташова Алиса, НФИбд-03-18

Родина Дарья, НФИбд-03-18

1. изучить теоретическую информацию о дендритах, в том числе и о модели роста дендритов;
2. разработать алгоритмы, позволяющие:
 - моделировать теплопроводность;
 - исследовать влияние начального переохлаждения S и величины капиллярного радиуса λ на форму образующихся дендритов;
 - исследовать зависимость от времени числа частиц в агрегате и его среднеквадратичного радиуса в разных режимах;
 - определить фрактальную размерность полученных образцов;
 - исследовать влияние величины теплового шума δ на вид образующихся агрегатов;
3. написать программу, взяв в основу разработанные ранее алгоритмы;
4. проанализировать полученные результаты.

Алгоритм роста дендритов

Input: $\text{SizeN}[N][M]$ - область роста дендритов
 $\text{SizeN}[N/2][M/2] = 1$ - заправка
 $\text{Temperature}[N][M]$ - температура для каждой ячейки
 T_m - температура плавления
 T_{inf} - начальная температура
 $\text{Temperature}[N/2][M/2] = c_p(T_m - T_{inf})/L$

 $\omega = 1/2$
 k - коэффициент теплопроводности
 ρ - плотность материала
 c_p - теплоемкость при постоянном давлении
 $h = 1$
 $\Delta T = 1$

 Δ - величина флуктуаций температуры
 T_{plav} - температура плавления
 L - удельная теплота плавления на единицу массы
 γ - коэффициент поверхностного натяжения

Output: $\text{SizeN}[N][M]$, в которой значения принимаются равными 1 в том случае, если дендрит рос в том узле

Алгоритм роста дендритов

```
# посчитаем температуры во всех узлах - Temperature[N][M]

GradTemperature[N][M] - градиент температур
hi = k / (p * cp) -

# посчитаем градиенты температур
for всех узлов из Temperature[N][M] where в SizeN[N][M] нет 1
    for всех узлов границы (i = 0, i = N-1, j = 0, j = M-1)
        if соседей нет
            then вместо Temperature[][] недостающих do
                Tij = ((sum(Temperature[][] соседей)) +
                    (Temperature[N/2][M/2] * количество соседей,
                    которых нет)) / (4 + 4 * omega)

        for всех остальных узлов
            Tij = (Temperature[i+1][j] + Temperature[i-1][j] +
                Temperature[i][j-1] + omega * Temperature[i+1][j+1] +
                Temperature[i-1][j+1] + Temperature[i-1][j-1]) / (4 + 4 *
                omega)
            GradTemperature[i][j] = (Tij - Temperature[i][j]) / (4 + 4 * omega)
            * (1 + 2 * omega) / (h * h)

for всех узлов из Temperature[N][M]
    Temperature[i][j] = Temperature[i][j] + hi * deltaT *
    GradTemperature[i][j] / m
```

Алгоритм роста дендритов

```
# построим дендрит - заполним SizeN[N][M]

nu[N][M] = произвольные число в интервале [-1;1], имеющие равномерное
           распределение
noSizePlavT = cp * (plavT - infinitiveT) / L - безразмерная температура
           плавления
lamda = gamma * plavT * cp / (p * L * L) - капиллярный радиус

for всех узлов из SizeN[N][M]
    for всех узлов из SizeN[N][M]
        sumNotDiagonal = SizeN[i-1][j] + SizeN[i+1][j] + SizeN[i][j-1]
        + SizeN[i][j+1]
        sumDiagonal = SizeN[i-1][j-1] + SizeN[i-1][j+1] + SizeN[i+1]
        [j-1] + SizeN[i+1][j+1]
        n = sumNotDiagonal + sumDiagonal - сумма непустых узлов
        if (n >= 1)
            GranS[i][j] = sumNotDiagonal + omega * sumDiagonal -
            (5/2 + 5/2 * omega) - кривизна границы
            T = coldPlavT(1 + nu[i][j] * delta) + lamda * GranS[i][j]
            if (Temperature[i][j] <= T)
                SizeN[i][j] = 1
```