Проект: Электрический пробой.

Этап 3. Программная реализация

Кадров В. М.

Tуем Г.

Адабор К.

Содержание

# 1 Постановка задачи

Необходимо написать программную реализацию решения задачи моделирования электрического пробоя на языке Julia.

# 2 Задание начальных условий

Задаем начальные условия для расчета электрического поля: сетка 50x50, два электрода:

* К правому приложено напряжение
* На левом потенциал равен нуля, как на других границах
* Начальные значения в узлах сетки также равны нулю

# Параметры и начальные условия  
rows, cols = 50, 50  
field = fill(0.0, rows, cols)  
field[:, 1] .= 0.0 # левый электрод  
field[:, end] .= 100.0 # правый электрод

# 3 Расчет электрического поля (решение уравнения Лапласа)

Итерации в цикле останавливаются, если мы достигли сходимости, либо же если количество итераций превысило максимум (max\_iter).

# Решение уравнения Лапласа  
function solve\_field(field; max\_iter=10000, tol=1e-6)  
 rows, cols = size(field)  
 new\_field = copy(field)  
 # Основной цикл итераций  
 for \_ in 1:max\_iter  
 # Проходимся по всем узлам сетки  
 for i in 2:rows-1, j in 2:cols-1  
 new\_field[i, j] = 0.25 \* (field[i-1, j] + field[i+1, j] + field[i, j-1] + field[i, j+1])  
 end  
 if maximum(abs.(new\_field - field)) < tol  
 break  
 end  
 field, new\_field = new\_field, field  
 end  
  
 return field  
end

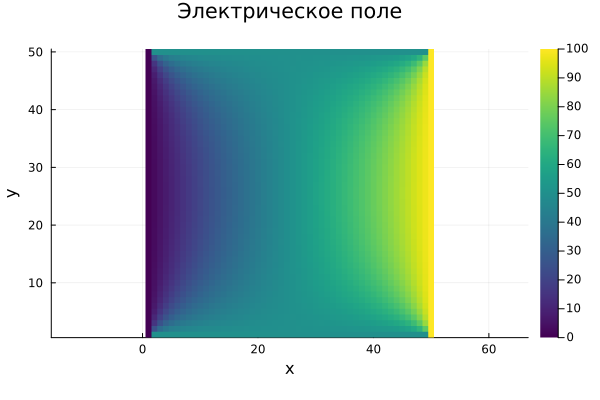


Рис. 1: Электрическое поле для заданной системы

# 4 Моделирование роста стримера

Расчитываем структуру стримера по модели НПВ (Нимейера, Пьетронеро и Висмана)[1].

function grow\_streamer(field, growth\_factor=2.0, max\_steps=10\_000)  
 rows, cols = size(field)  
 path = [(rand(1:rows), 1)]  
 # задаем максимальное количество шагов  
 for \_ in 1:max\_steps  
 curr = path[end]  
 # Находим возможные направления роста  
 neighbors = [(curr[1]+di, curr[2]+dj) for di in -1:1, dj in -1:1  
 if (di != 0 || dj != 0) &&  
 1 ≤ curr[1]+di ≤ rows &&  
 1 ≤ curr[2]+dj ≤ cols &&  
 !((curr[1]+di, curr[2]+dj) in path)]  
  
 if isempty(neighbors)  
 break  
 end  
 # Считаем сумму вероятностей и умножаем на случайное число   
 probs = [abs(field[i, j])^growth\_factor for (i, j) in neighbors]  
 total\_prob = sum(probs)  
 r = rand() \* total\_prob  
  
 acc = 0.0  
 # Находим узел, в котором накопленная сумма превышает порог r  
 for ((i, j), p) in zip(neighbors, probs)  
 acc += p  
 if acc > r  
 push!(path, (i, j))  
 break  
 end  
 end  
 # Проверяем, не достигли ли мы противоположного электрода  
 if path[end][2] == cols  
 break  
 end  
 end

# 5 Моделирование

Необходимо дополнительно задать фактор роста ()и добавить визуализацию.

field = solve\_field(field)  
path = grow\_streamer(field, 3.0)  
  
# Визуализация  
heatmap(field, aspect\_ratio=1, title="Электрическое поле и стример", xlabel="x", ylabel="y", color=:viridis)  
x, y = [i for (i, \_) in path], [j for (\_, j) in path]  
plot!(y, x, color=:red, lw=2, label="Стример", legend=:topright)

Попробуем запустить моделирования с разным фактором роста

При малом параметре роста структура получается достаточно запутанной.

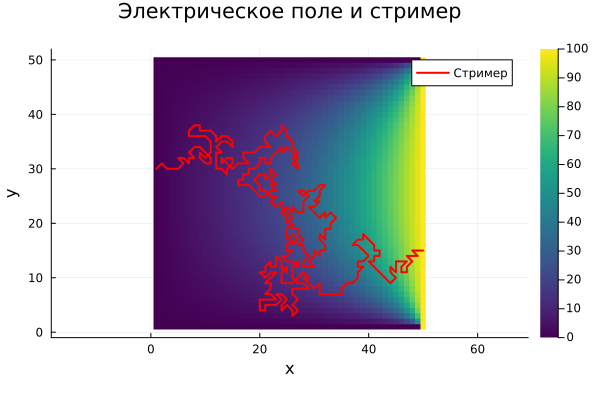


Рис. 2: Стример, показатель роста равен 1.2

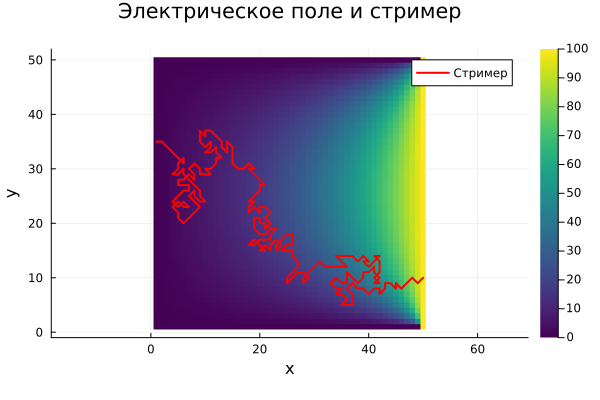


Рис. 3: Стример, показатель роста равен 2

Хорошо видно, что чем выше показатель роста, тем более прямолинейная получается структура стримера

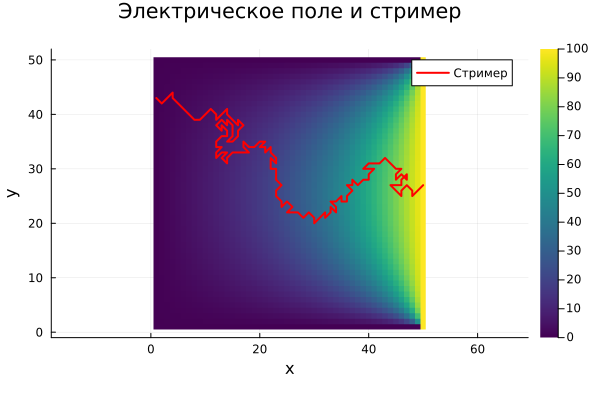


Рис. 4: Стример, показатель роста равен 3

# 6 Заключение

Была написана программная реализация решения задачи моделирования электрического пробоя на языке Julia.

# Список литературы

1. Медведев Д. и др. Моделирование физических процессов и явлений на ПК.