

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
им. М.В. ЛОМОНОСОВА  
ФИЗИЧЕСКИЙ ФАКУЛЬТЕТ

Кафедра квантовой статистики и теории поля

**КУРСОВАЯ РАБОТА**  
**Динамика модели решеточного газа**

Пышный Владимир Владимирович  
2 курс, 213 группа

Руководитель курсовой работы  
**К.ф.-м.н. Павловский О.В.**

Москва  
2024 г.

# Содержание

1. Введение.....	3
2. Постановка задачи.....	4
2.1 Основы решеточных моделей.....	4
2.2 Модель двумерного газа с треугольной решеткой.....	5
3. Программная реализация.....	6
3.1 Строение программы.....	7
3.2 Обработка соударений в программе.....	7
4. Результаты численного моделирования.....	9
4.2 Удар – термализация.....	9
4.3 Исследование джетов.....	12
5. Заключение.....	14

## Список литературы

# 1. Введение

Цель текущей работы заключается в изучении динамики модели решеточного газа с использованием компьютерного моделирования. Необходимо понять различные аспекты динамики этой модели и получить новые результаты, которые могут быть полезны для понимания поведения газов в различных условиях.

Внедрение современных средств компьютерной техники в практику исследований позволило резко увеличить эффективность математического моделирования процессов переноса энергии и массы в различных средах. В настоящее время широкое распространение получают дискретные модели таких процессов. Отдельным направлением таких динамических моделей для исследования процессов переноса являются клеточные автоматы

Клеточными автоматами принято называть равномерные сетки из элементов, меняющих свое состояние в дискретные моменты времени по определенному закону в зависимости от состояния элемента и его соседей в сетке в предыдущий момент времени. Данный метод используется в различных областях:

- Моделирование распространения вирусов и микроорганизмов, [1]
- Моделирование распространения пожара и других катаклизмов, [2]
- Анализ явлений переноса в физике, [3]
- и др.

Из описания клеточных автоматов можно сделать вывод о том, что это один из самых простых и наглядных способов моделирования динамики двумерного решеточного газа.

Для выполнения цели курсовой работы на языке Python 3.0 была написана компьютерная программа, которая моделирует поведение двумерного газа на треугольной решетке. С ее помощью были собраны все необходимые статистические данные системы, которые используются текущей курсовой работе.

## 2. Постановка задачи

### 2.1 Основы решеточных моделей

В решеточных моделях газ представляется некоторым набором идеальных частиц, которые перемещаются в пространстве по узлам специальной решетки. Такая модель обладает следующими правилами:

- Частицы могут находиться только в узлах равномерной решетки.
- Скорости всех частиц одинаковы по модулю.
- Скорость частицы может быть направлена лишь вдоль одной из связей, которые соединяют ее узел с соседними.
- В каждом узле может находиться не более одной частицы с данным направлением скорости.
- Взаимодействие частиц друг с другом может происходить только в узлах решетки.

В наше время существует большое количество различных моделей решеточных газов. Одной из самых простых можно считать модель Харди-Пацциса-Помо (НРР). В данной модели рассматривается динамика газа на ортогональной (квадратной) решетке с постоянным шагом  $h$ . (Рис1.) Исследования показали, что данная модель может вполне хорошо описывать такое поведение решеточного газа, однако, в целом результат отличается от реальных экспериментов. [4]

Несмотря на это, в силу простоты своей реализации, модель НРР стала основой для развития более сложных газовых клеточных автоматов.

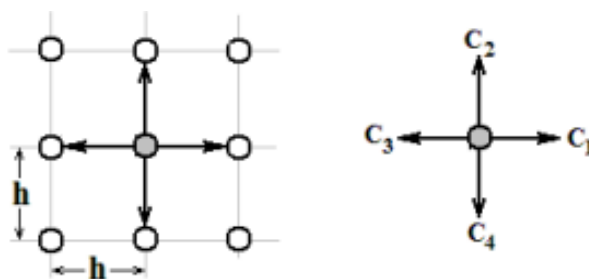


Рис. 1. Схема решетки НРР и векторы скорости

## 2.2 Модель двумерного газа с треугольной решеткой.

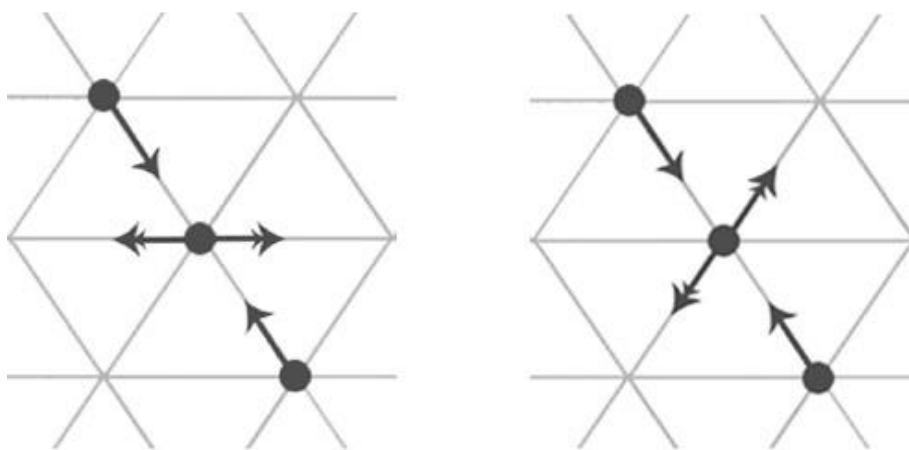
В данной работе будет использована усовершенствованная модель НРР, а именно, модель двумерного газа с треугольной решеткой. [3] Это увеличит число степеней свободы частицы до 6, что даст более точные результаты в симуляции.

Развитие поведения нашей системы можно описать с помощью двух этапов:

- 1) Перемещение частиц в соседние узлы;
- 2) Проверка и обработка условий соударения.

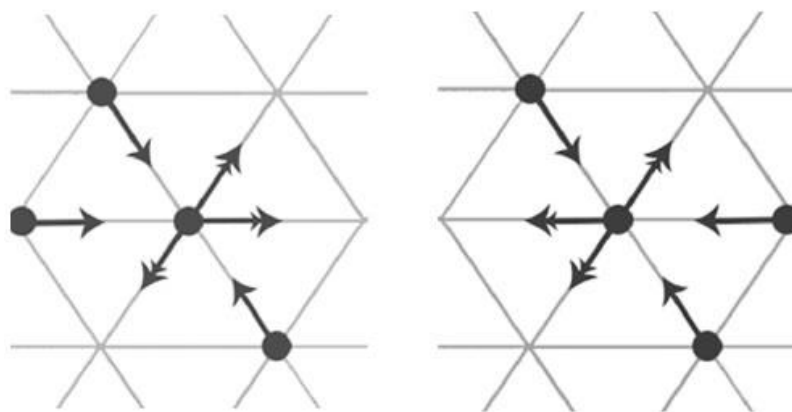
Далее будет описан процесс рассеивания частиц при взаимодействии друг с другом. Поскольку в текущий момент мы не различаем частицы, то случаев, при которых молекулы могут менять направления своих скоростей всего два.

При «лобовом» соударении двух частиц в узле, направление скорости частицы может изменяться на угол  $\varphi = \pm\pi/3$ . Таким образом возникают две равновероятные ситуации рассеивания:



*Рис. 2.* Единичными и двойными стрелками показаны направления векторов скоростей частиц в дискретные моменты времени  $n$  и  $n+1$  соответственно.

В том случае, если один из «каналов» рассеивания частицы закрыт другой, скоростью которой направлена в данный узел. То возможен лишь один случай разлета:



*Рис. 3.* Единичными и двойными стрелками показаны направления векторов скоростей частиц в дискретные моменты времени  $n$  и  $n+1$  соответственно.

В остальных же случаях мы считаем, что никакого рассеивания не происходит, и частицы как бы пролетают мимо друг друга, не меняя направления своих скоростей.

### 3. Программная реализация

Основной целью при написании программы было создание законченного, модульного и масштабируемого проекта. Который мог бы использоваться в дальнейшей научной деятельности. Для этого были подключены такие основные библиотеки Python, как: Numpy, Matplotlib.pyplot, Matplotlib.animation, Random и Itertools. В качестве набора решаемых задач были выбраны основные направления:

- Моделирование динамики решеточного газа.
- Визуализация движения частиц в режиме реального времени.
- Сбор информации о направлении вектора скорости каждой частицы.
- Экспортирование данных в OriginLab для последующей обработки.

## 3.1 Структура программы

Для оптимизации работы приложение было разбито на отдельные функции, каждая из которых выполняла свою часть работы. В силу большого объема программы обзор всего кода не представляется возможным. Поэтому опишу работу только основных частей программы. Изначально для удобства обработки был написан класс `Point()`.

```
class Point():
    def __init__(self, x=0, y=0, vx=0, vy=0, colour=0):
        self.x = x
        self.y = y
        self.vx = vx
        self.vy = vy
        self.colour = colour
```

Он представляет собой простую модель точки в двумерном пространстве, которая обладает своими координатами (x,y), вектором скорости  $V = \{V_x, V_y\}$  и определенным цветом. До инициализации точки все эти параметры равны 0.

## 3.2 Обработка соударений в программе

Главной функцией, ответственной за взаимодействие частиц между собой, является `collision`.

```
def collision(point0, point1, point2, point3):

    if collision_of_4_points(point0, point1, point2, point3):
        return [point0, point1, point2, point3]

    if collision_of_3_points(point0, point1, point2)\
        or collision_of_3_points(point0, point1, point3)\
        or collision_of_3_points(point0, point2, point3)\
        or collision_of_3_points(point1, point2, point3):
        return [point0, point1, point2, point3]

    if collision_of_2_points(point0, point1)\
        or collision_of_2_points(point0, point2)\
        or collision_of_2_points(point0, point3)\
        or collision_of_2_points(point1, point2)\
        or collision_of_2_points(point1, point3)\
        or collision_of_2_points(point2, point3):
        return [point0, point1, point2, point3]

    return
```

За одну итерацию функция принимает на вход 4 точки из общего массива данных и возвращает значения уже с учетом модели соударения. Функции `collision_of_4_points`, `collision_of_3_points` и `collision_of_2_points` проверяют условия соударений 4-х, 3-х и 2-х точек соответственно. Метод проверки по «четверкам» и такой порядок функций позволяют оптимизировать работу кода, так как по правилам модели, при встрече 4-х или более точек в одном узле направления векторов их скоростей не изменяются.

Из-за ограничений вычислительных способностей ЭВМ невозможно проводить симуляцию двумерного газа на бесконечной плоскости. Поэтому было выбрано поле в виде квадрата с периодическими границами.

При подлете к границе частица не соударяется с ней, а переносится противоположно направлению ее вектора скорости до встречи с другой границей. При этом координаты самого вектора не изменяются.

```
def advance(board, n):
    global base
    global s2

    board = sorted(board).copy()

    for i in range(n - 3):
        collision(board[i], board[i + 1], board[i + 2], board[i + 3])

    newstate = []
    for i in range(n):
        grid = 300
        speed = 1

        newstate.append([board[i][0] + board[i][2], board[i][1] +
            board[i][3], board[i][2], board[i][3], board[i][4]])

        if right_out_of_bounds(board[i], grid, speed):
            newstate[i] = right_out_of_bounds(board[i], grid, speed)
        if left_out_of_bounds(board[i], grid, speed):
            newstate[i] = left_out_of_bounds(board[i], grid, speed)
        if up_out_of_bounds(board[i], grid, speed):
            newstate[i] = up_out_of_bounds(board[i], grid, speed)
        if down_out_of_bounds(board[i], grid, speed):
            newstate[i] = down_out_of_bounds(board[i], grid, speed)

    return newstate
```



За эту операцию отвечают функции `right_out_of_bounds`, `left_out_of_bounds`, `up_out_of_bounds`, `down_out_of_bounds`. Они обрабатывают взаимодействие с правой, левой, верхней и нижней границей квадрата соответственно. Функция `advance` вызывается каждый дискретный момент времени, обеспечивая движение частиц между узлами и вызов функций.

Функции, разобранные выше, являются основными блоками программы

## 4. Результаты численного моделирования

### 4.1 Удар - термализация

В ходе моделирования рассматривалась эволюция квадратного поля размерами 600 x 600 узлов решетки. В исходном состоянии система представляла собой две капли, каждая из которых состояла из 357 молекул. В ходе симуляции данные массивы точек испытывали полное лобовое соударение. Далее на рисунке 4 показаны результаты моделирования за 50 машинных секунд. Время проставлено под каждым рисунком соответственно.

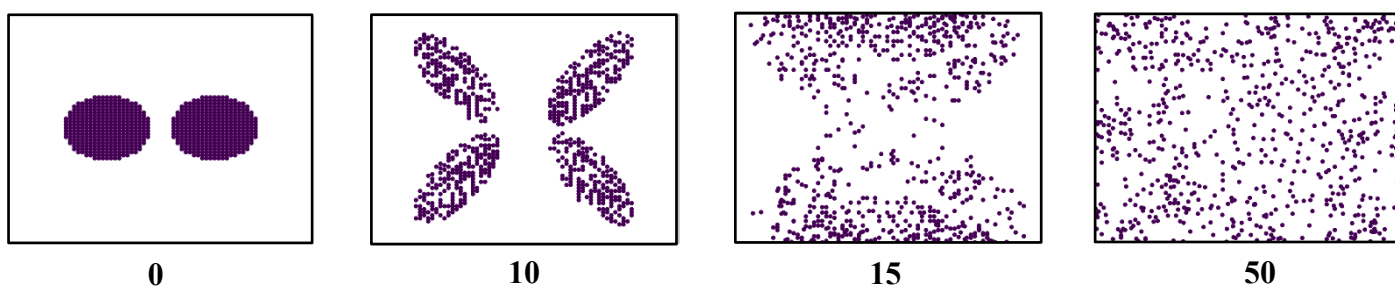
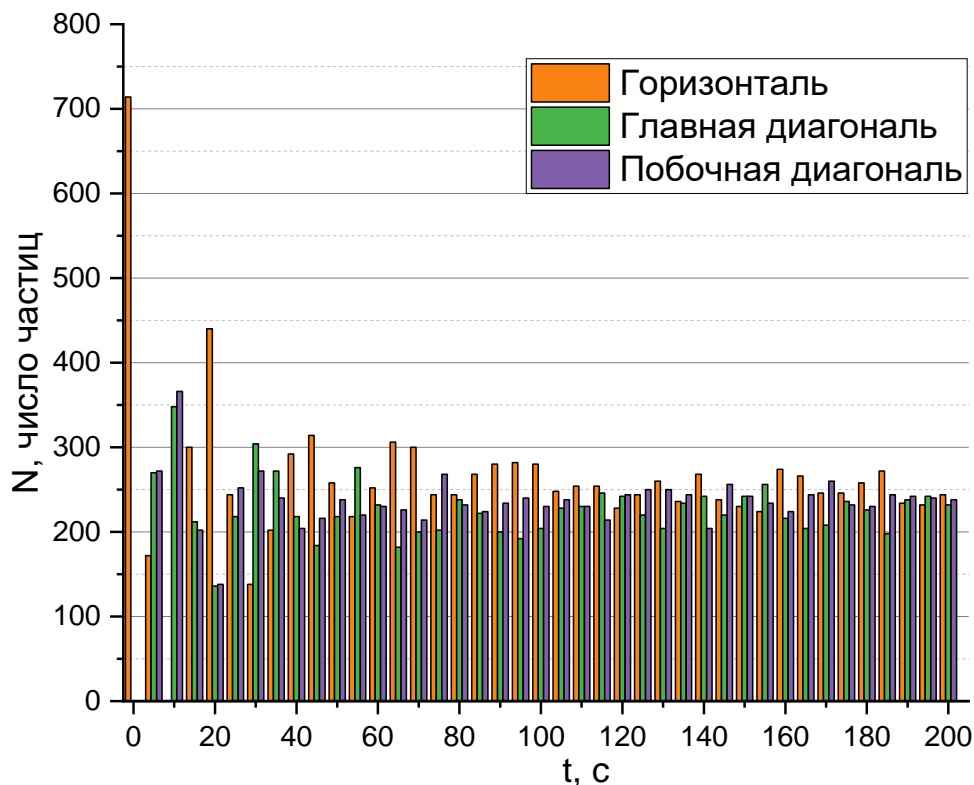


Рис 4. Изменение состояния клеточного массива

Анализ полученных данных показывает, что по мере развития процесса массив точек разделяется на джеты, а затем равномерно заполняет все рабочее поле.

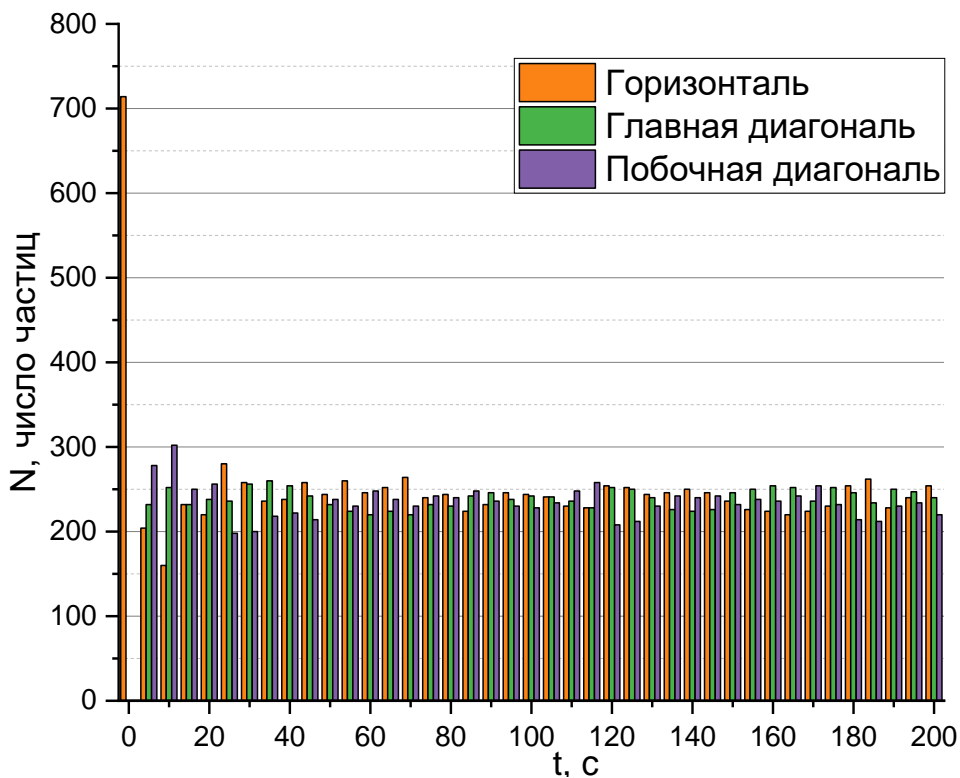
Для более точной оценки представлена гистограмма распределения числа точек по направлениям скоростей. На рисунке 5 изображено распределение для классического случая соударения двух капель.



**Рис 5.** Распределение точек по направлениям скоростей для двух капель. Вероятность взаимодействия равна 1.

Можно заметить, что система переходит в стационарное состояние, около которого совершает флуктуации, лишь после 80 секунд симуляции.

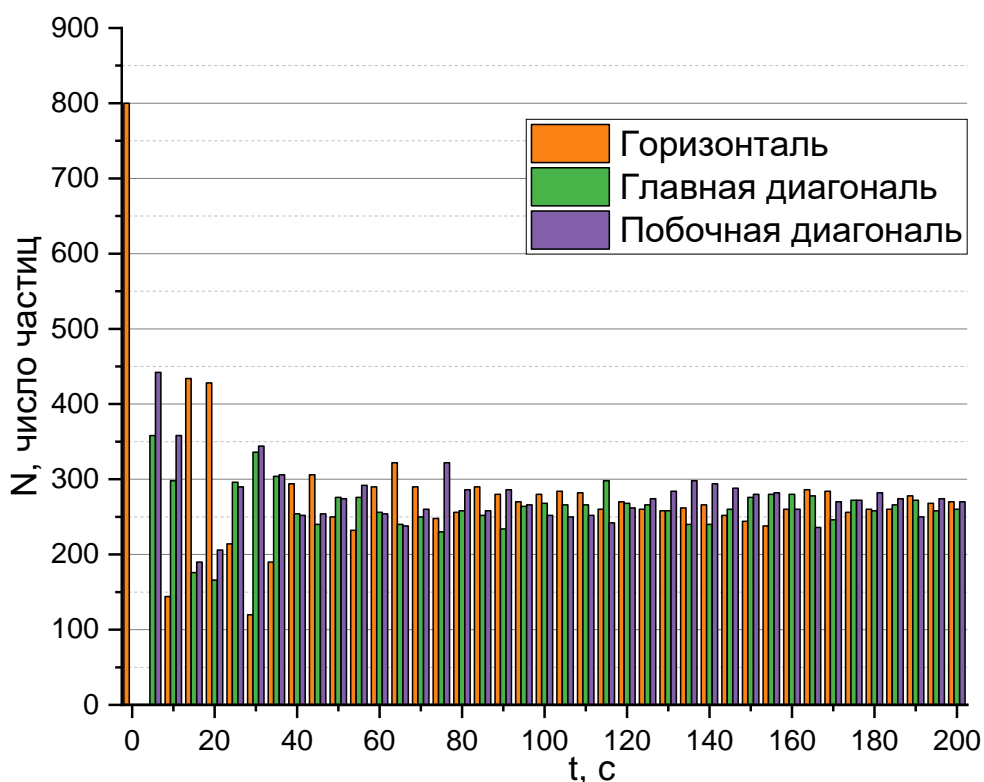
Далее была уменьшена вероятность взаимодействия частиц между собой. В новой симуляции она стала равна 0.2.



**Рис 6.** Распределение точек по направлениям скоростей двух капель. Вероятность взаимодействия равна 0.2.

Сравнивая распределения, изображенные на рисунках 5 и 6, можно наблюдать, что при низкой вероятности взаимодействия частиц между собой система в меньшей степени подвержена колебаниям, а установление стационарного режима происходит быстрее, чем в первом случае. Отсюда можно сделать о том, что уменьшение параметра вероятности взаимодействия положительно сказывается на стабильности всей системы.

В следующем исходном состоянии симуляции система представляла собой два плоских вертикальных фронта, движущихся на встречу друг другу. Весь массив данных содержал в себе 800 точек с максимальной вероятностью взаимодействия между собой. Гистограмма распределения точек по их направлениям изображена на рисунке 7.



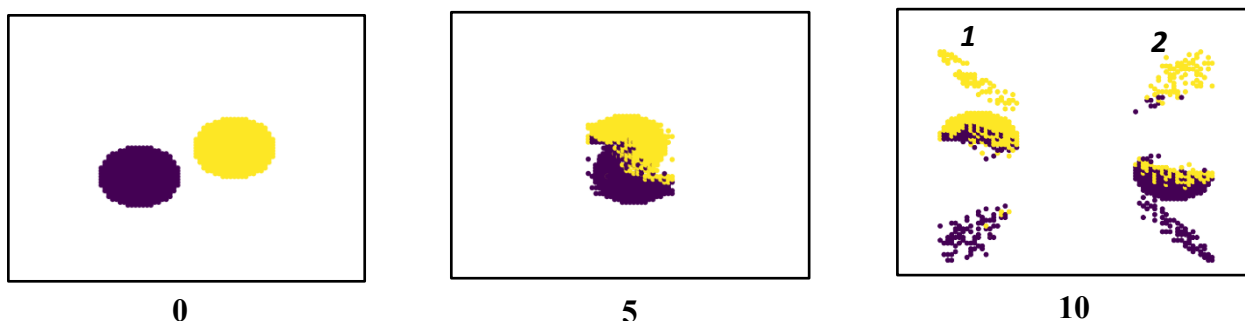
**Рис. 7.** Распределение точек по их скоростям для двух вертикальных фронтов. Вероятность взаимодействия равна 1.

Анализируя рисунки 7 и 5, можно заметить, что при большем числе точек в системе и равной вероятности взаимодействия, стационарное состояние для двух фронтов наступает раньше, чем для двух капель, а флуктуации системы намного меньше.

Таким образом можно утверждать, что параметр термализации системы зависит не только от физических характеристик газа, но и от начальной конфигурации системы.

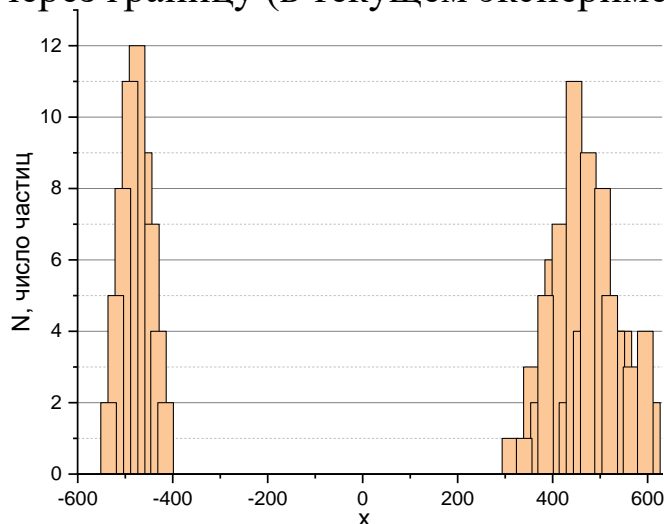
## 4.2 Исследование джетов

В данном эксперименте будет рассматриваться случай горизонтального соударения двух капель различных веществ, сдвинутых относительно друг друга на прицельное расстояние, равное их радиусу. На рисунке 8 показан результат такой симуляции в течение 10 машинных секунд.



**Рис 8.** Изменение состояния клеточного массива для капель разного вещества. Вероятность взаимодействия равна 1.

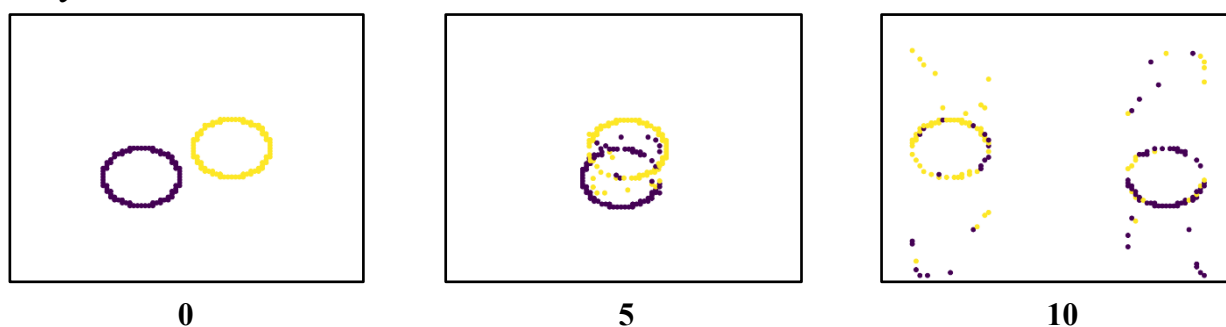
При анализе полученных данных можно заметить, что в отличие от центрального лобового удара разлетающиеся джеты несимметричны. Особенно хорошо это заметно на примере 1 и 2 джетов. Для более точного анализа данной симуляции была построена гистограмма распределения числа точек от их координаты при пролете через границу (в текущем эксперименте она открыта).



**Рис. 9.** Распределение точек по их координатам при пролете через верхнюю границу поля

На рисунке 9 можно наблюдать 2 различных распределения. Левый и правый пики описывают 1 и 2 джеты соответственно. Такое рассеяние можно объяснить тем, что первый джет является «сканирующим», т.е. он прорывается сквозь оставшуюся часть капли, выбивая из нее другие частицы. Таким образом он теряет часть своей мощности и меняет начальную форму.

В следующей симуляции заменим две капли на пустые оболочки с идентичными размерами и вероятностями взаимодействия. На рисунке 10 показана симуляция первых 10 секунд системы.



*Рис. 10.* Изменение состояния клеточного массива для оболочек разного вещества. Вероятность взаимодействия равна 1.

На данном рисунке можно наблюдать, что такого рассеивания частиц, как в предыдущей симуляции, очевидно, нет. Все джеты имеют схожую форму.

Результаты, полученные в ходе экспериментов, подтверждают тот факт, что картина рассеивания на прямую зависит от изначальной конфигурации системы, а с помощью исследования отдельных джетов можно судить о её состоянии.

## 5. Заключение

В данной работе были исследованы основы математического моделирования явлений переноса с использованием дискретных моделей. В рамках исследования была разработана компьютерная программа, имитирующая поведение плоского газа на треугольной решетке. Кроме того, были созданы инструменты для визуализации и сбора статистических данных системы.

Результаты моделирования продемонстрировали, что вероятность взаимодействия частиц и их начальная конфигурация являются ключевыми параметрами, определяющими динамику и термализацию системы. Анализ отдельных кластеров при рассеивании крупных скоплений частиц позволяет установить их начальные геометрические параметры и предсказать дальнейшее поведение газа.

Интерес к результатам данного исследования обусловлен тем, что клеточные автоматы могут быть применены для качественного описания более сложных систем, где обработка взаимодействия частиц значительно усложняется. Например, перспективным направлением является симуляция газа на додекаэдрической решетке в трёхмерном пространстве.

Таким образом, проведённое исследование подтверждает значимость выбранного подхода для изучения сложных систем и открывает возможности для дальнейших научных изысканий в области дискретного моделирования физических явлений.

## Список литературы

- [1] Daley, D. J. & Gani, J. (2005). Epidemic Modeling: An Introduction. NY / Cambridge University Press.
- [2] Виноградов В. И., Моделирование распространение лесного пожара с помощью клеточного автомата: Введение / ред. ВШЭ 2023
- [3] Лоскутов А. Ю., Михайлов А. С. Введение в синергетику: Обработка информации распределенными динамическими системами / Физ.-мат.-лит. – М., 1990
- [4] Бобков С. П., Соколов В. Л. Моделирование поведения газа с использованием решеточных моделей: Инженерно-технические науки / Вестник ИГЭУ 2015