

**Министерство науки и высшего образования Российской
Федерации**

**Федеральное государственное бюджетное образовательное
учреждение высшего образования**

**« Российский химико-технологический университет имени
Д.И. Менделеева »**

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №10

Вариант 7

Выполнил студент группы КС-46: Золотухин Андрей Александрович

Ссылка на репозиторий: [https://github.com/
CorgiPuppy/
info-processes-systems-theory-labs](https://github.com/CorgiPuppy/info-processes-systems-theory-labs)

Приняла: Зинченко Дарья Ивановна

Дата сдачи: 20.12.25

Москва

2025

Оглавление

Описание задачи	3
Работа 10	3
Выполнение задачи	4
Аналитическое решение	4
Графики	5
Код	9
Выводы	12

Описание задачи

Работа 10

Выполнение задачи

Аналитическое решение

На Рис. 1 представлено аналитическое решение.

Анализировать решение §10. Задание: Андрей Александрович
Класс: Параллель 7

$$\begin{cases} \frac{dx}{dt} = 5(3-x) \\ \frac{dy}{dt} = -x-y-xz \\ \frac{dz}{dt} = -xy-bz \end{cases} \quad \begin{cases} 3=7, b=5 \\ r - \text{эквивалентный параметр} \end{cases}$$

$$\begin{cases} 5(y-x) < 0 \\ x-y-xz=0 \\ x-y-bz \end{cases} \Rightarrow \begin{cases} y \geq x \\ x(y-x-z)=0 \Rightarrow \\ z = x^2/b \end{cases}$$

(\Rightarrow) $\begin{cases} \bar{y}=0 \\ \bar{x}=0 \\ \bar{z}=0 \end{cases}$ $\bar{y}(0,0,0)$

$$\begin{cases} \bar{y} = \sqrt{b(z-1)} \\ \bar{x} = \sqrt{b(z-1)} \\ \bar{z} = z-1 \\ \bar{x} = -\sqrt{b(z-1)} \\ \bar{z} = z-1 \end{cases}$$

$$\lambda_1 = -b = -5$$

$$\lambda_{2,3} = \frac{-1+5}{2} \pm \frac{\sqrt{(5-1)^2 - 4 \cdot 5 \cdot 1}}{2} = \frac{-8 \pm \sqrt{36+28}}{2}$$

$$r > 1; \sqrt{...} > 8 \Rightarrow \text{седло}$$

$$r < 1; \sqrt{...} < 8 \Rightarrow \text{уст. узел}$$

Симметричные:

$$r_2 = \frac{5-b+5}{5-b-1} = \frac{7+5+5}{7-5-1} = 10.5$$

$$1 < r < 10.5: \text{уст. узел}$$

$$r > 10.5: \text{седло}$$

Рис. 1: Аналитическое решение.

Графики

На Рис. 2 представлена бифуркационная диаграмма.

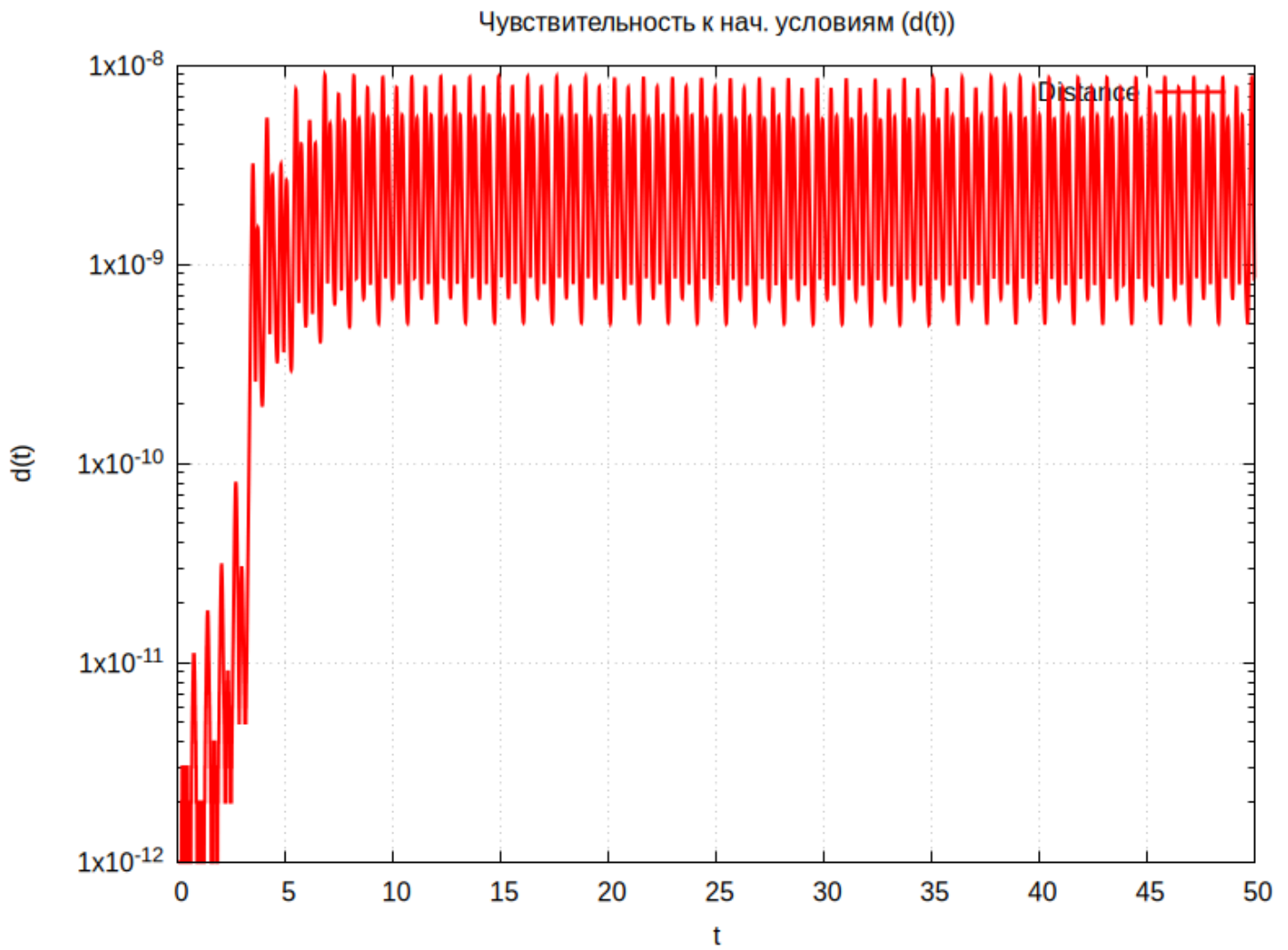


Рис. 2: Бифуркационная диаграмма.

На Рис. 3 представлен фазовая плоскость при $r = 0.5$.

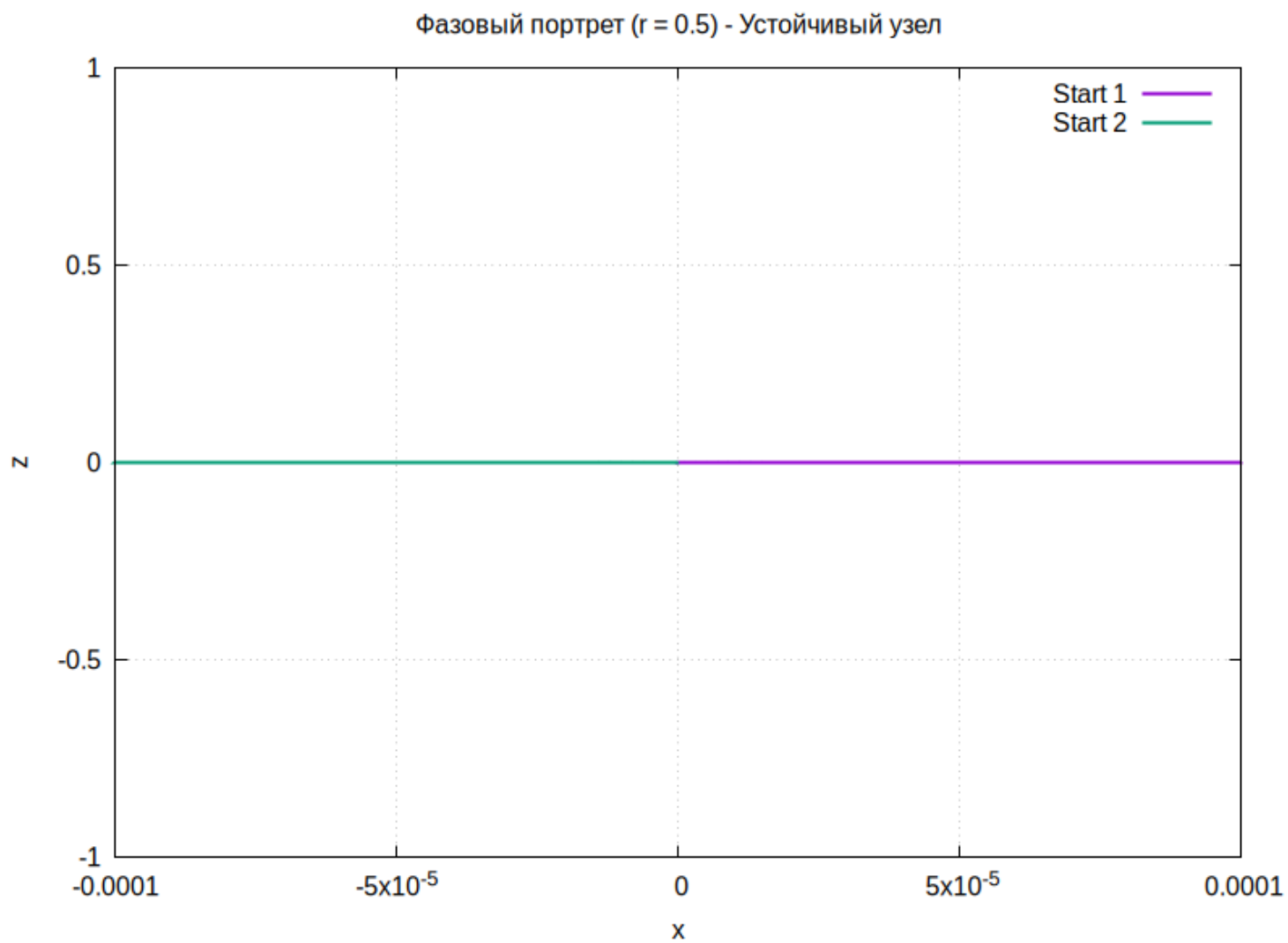


Рис. 3: Фазовая плоскость при $r = 0.5$.

На Рис. 4 представлен фазовая плоскость при $r = 50$.

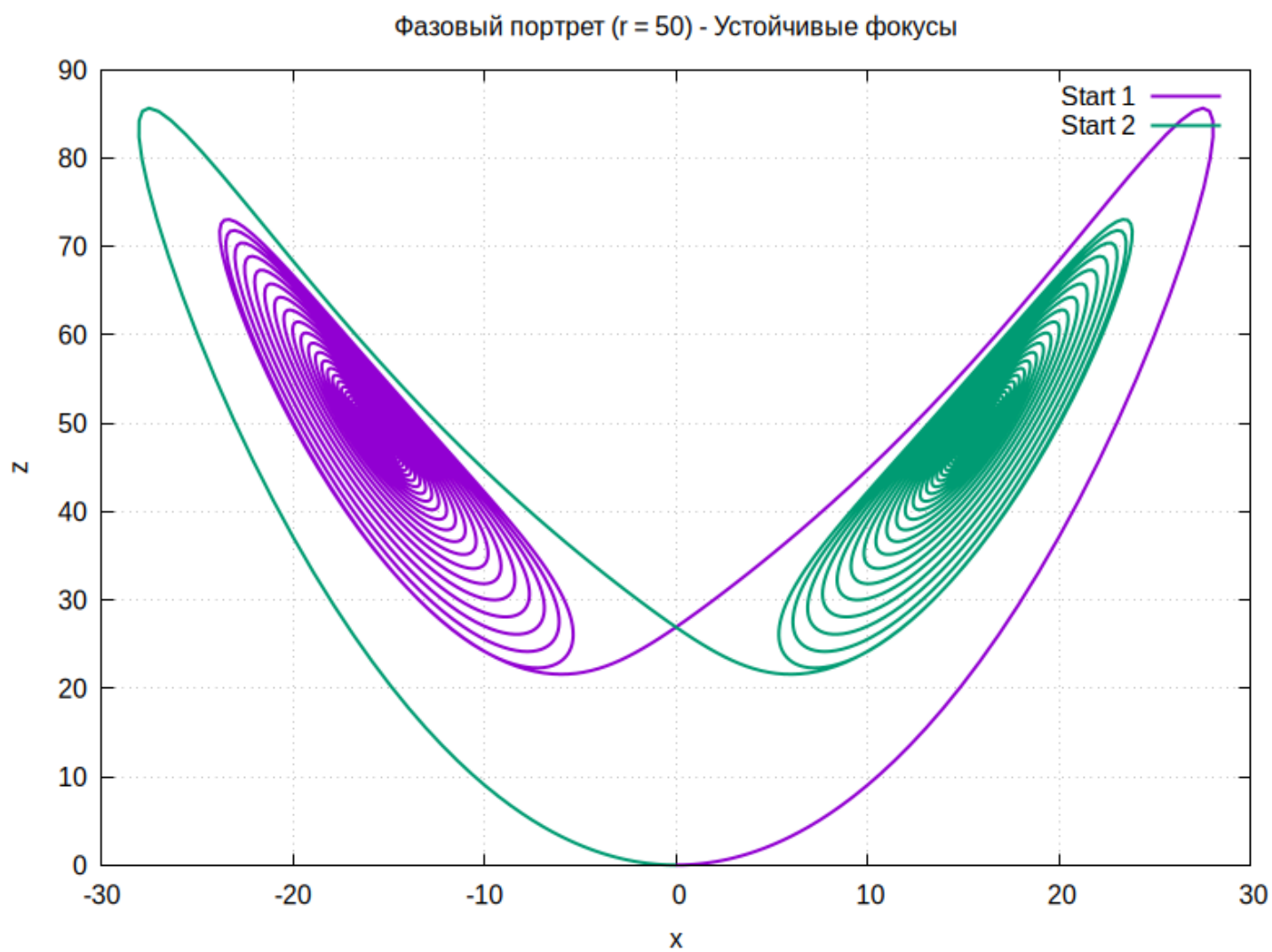


Рис. 4: Фазовая плоскость при $\gamma = 50$.

На Рис. 5 представлен фазовая плоскость при $\gamma = 120$.

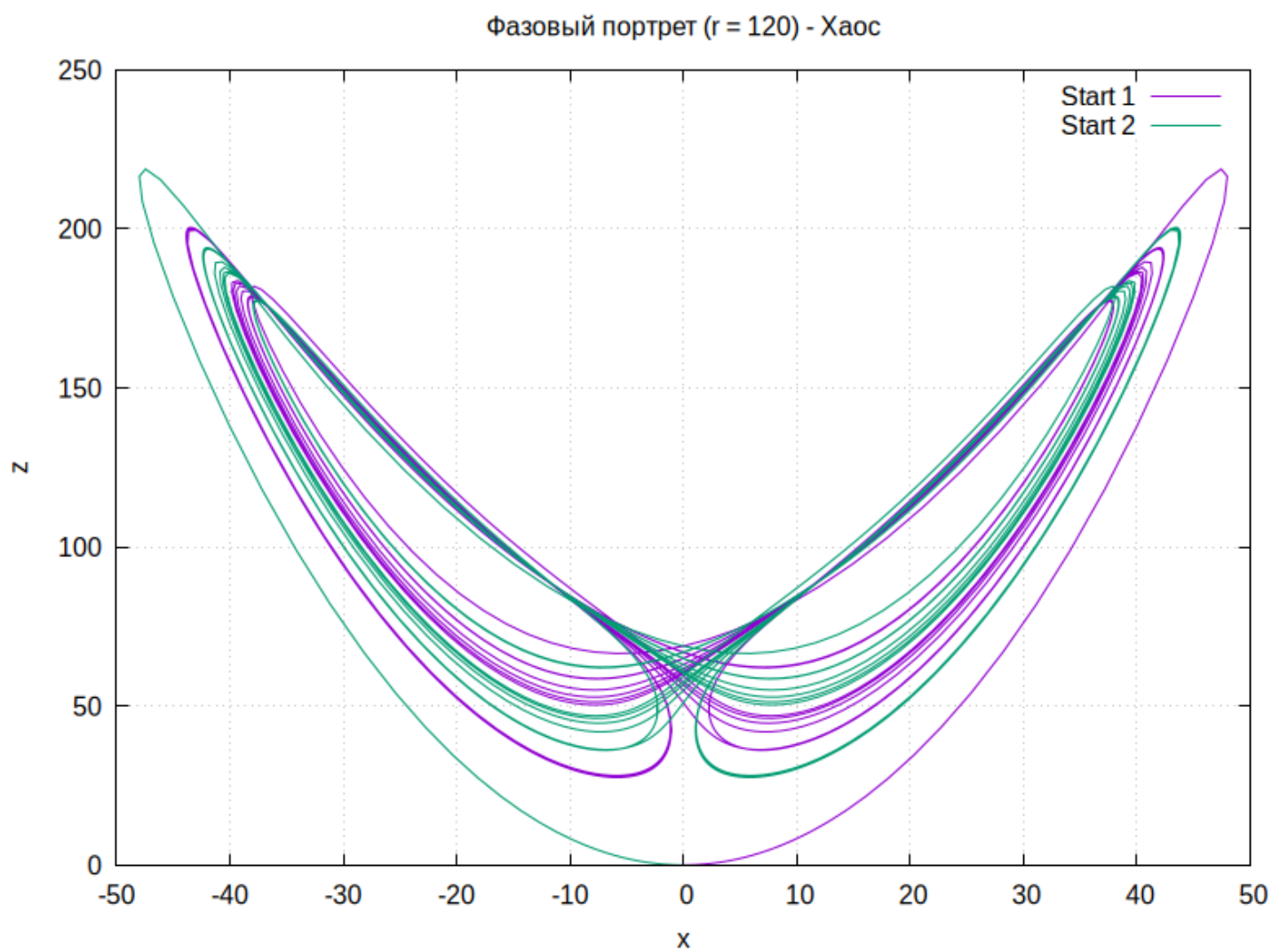


Рис. 5: Фазовая плоскость при $r = 120$.

Код

```
#ifndef CONSTANTS_H
#define CONSTANTS_H

#include <string>
#include <vector>

namespace Constants {
    const std::string DAT_DIR = "plots/dat-files/";

    const double sigma = 7.0;
    const double b = 5.0;

    const double dt = 0.005;
    const double T = 50.0;

    const std::vector<double> r_values = {0.5, 50.0, 120.0};

    const std::vector<std::string> filenames = {
        "r_0.5.dat",
        "r_50.0.dat",
        "r_120.0.dat"
    };

    const double r_chaos = 120.0;
    const double epsilon = 1e-12;
    const std::string div_filename = "divergence.dat";
}

#endif
```

```
#include <iostream>
#include <vector>
#include <cmath>
#include <fstream>
#include <iomanip>
#include <string>

#include "Constants.h"

using namespace std;

struct Point {
    double x;
    double y;
    double z;
};

Point get_derivatives(Point p, double r) {
    double dx = Constants::sigma * (p.y - p.x);
    double dy = r * p.x - p.y - p.x * p.z;
    double dz = p.x * p.y - Constants::b * p.z;
    return {dx, dy, dz};
}

void step_rk4(Point& p, double dt, double r) {
    Point k1 = get_derivatives(p, r);
```

```

Point p2 = {p.x + 0.5*dt*k1.x, p.y + 0.5*dt*k1.y, p.z + 0.5*dt*k1.z};
Point k2 = get_derivatives(p2, r);

Point p3 = {p.x + 0.5*dt*k2.x, p.y + 0.5*dt*k2.y, p.z + 0.5*dt*k2.z};
Point k3 = get_derivatives(p3, r);

Point p4 = {p.x + dt*k3.x, p.y + dt*k3.y, p.z + dt*k3.z};
Point k4 = get_derivatives(p4, r);

p.x += (dt / 6.0) * (k1.x + 2*k2.x + 2*k3.x + k4.x);
p.y += (dt / 6.0) * (k1.y + 2*k2.y + 2*k3.y + k4.y);
p.z += (dt / 6.0) * (k1.z + 2*k2.z + 2*k3.z + k4.z);
}

// Задание 2: Фазовые портреты
void solveTask2() {
    vector<Point> starts = {
        {0.0001, 0.0, 0.0},
        {-0.0001, 0.0, 0.0}
    };

    for(size_t i = 0; i < Constants::r_values.size(); ++i) {
        double r = Constants::r_values[i];
        string path = Constants::DAT_DIR + Constants::filenames[i];

        ofstream file(path);
        if (!file.is_open()) {
            cerr << "Error opening: " << path << endl;
            continue;
        }
        file << fixed << setprecision(6);

        for(auto start : starts) {
            Point curr = start;
            for(double t = 0; t <= Constants::T; t += Constants::dt) {
                file << t << " " << curr.x << " " << curr.y << " " << curr.z << "\n";
                step_rk4(curr, Constants::dt, r);
            }
            file << "\n\n";
        }
        file.close();
    }
}

void solveTask3() {
    double r = Constants::r_chaos;
    string path = Constants::DAT_DIR + Constants::div_filename;

    ofstream file(path);
    if (!file.is_open()) return;
    file << fixed << setprecision(12);

    Point p1 = {1.0, 1.0, 1.0};
    Point p2 = {1.0, 1.0, 1.0 + Constants::epsilon};

    for(double t = 0; t <= Constants::T; t += Constants::dt) {
        double dist = sqrt(pow(p1.x - p2.x, 2) + pow(p1.y - p2.y, 2) + pow(p1.z - p2.z, 2));
    }
}

```

```

        file << t << " " << dist << "\n";

        step_rk4(p1, Constants::dt, r);
        step_rk4(p2, Constants::dt, r);
    }
    file.close();
}

int main() {
    solveTask2();
    solveTask3();
    return 0;
}

```

Выводы