Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт вычислительной математики и информационных технологий
Кафедра вычислительной математики

КУРСОВАЯ РАБОТА

Задача 20 - Задача о полете снаряда 1

Студент 4 курса

группы 09-822 Шабаков Ильвар Жомортханович

Руководитель Трифонова Галина Олеговна

Казань – 2021

Оглавление

Постановка задачи

Рассмотрим полет снаряда, выпущенной с начальной скоростью V под углом θ_0 к горизонту, при следующих предположениях:

- сила сопротивления воздуха пропорциональна квадрату скорости снаряда;
- дальность полета снаряда не превышает 10 км;
- боковой ветер отсутствует.

При сделанных допущениях можно считать, что земля плоская и вся траектория снаряда лежит в одной плоскости xOy. Уравнения движения центра масс снаряда в проекциях на оси координат запишутся в виде

$$x'' = -C\rho SV^2 \cos(\theta)/2m \qquad (1)$$
$$y'' = -C\rho SV^2 sin(\theta)/2m - g$$

Здесь m –масса снаряда, $V = (x'^2 + y'^2)^{1/2}$ – скорость движения, $\theta = arctg(\frac{y'}{x'})$ – угол между касательной к траектории и осью Ox, g – ускорение силы тяжести, S – площадь поперечного сечения снаряда, ρ – плотность воздуха, C – коэффициент лобового сопротивления снаряда
Система (1) дополняется начальными условиями

$$x(0) = 0, y(0) = 0, \theta(0) = \theta_0\ V(0) = V_0 \qquad (2)$$

Для численного решения удобно преобразовать два уравнения второго порядка (1) к системе четырех уравнений первого порядка. Дифференцируя соотношения

$$x' = v\,cos(\theta), y' = v\,sin(\theta) \qquad (3)$$

получим

$$x'' = v' cos(\theta) - v\,sin(\theta)\theta',$$
$$y'' = v' sin(\theta) + v\,cos(\theta)\theta'.$$

Подставляя теперь эти производные в уравнения (1) и разрешая полученные соотношения относительно v′ и θ′ получим

$$v' = -C\rho Sv2/(2m) - g\,sin(\theta), \qquad (4)$$
$$\theta' = -g\,cos(\theta)/v.$$

Уравнения (3), (4), вместе с начальными условиями (2), полностью описывают траекторию полета снаряда.

Задания

1. Проверьте правильность вывода исходной системы уравнений. Приведите соответствующий рисунок с указанием действующих на снаряд сил.
2. Напишите программу интегрирования задачи Коши для системы из n уравнений первого порядка вида
$y' = f(t, y),\ y(0) = y0,\ y(t)\ \in Rn,$
на произвольном отрезке [a, b], используя метод Рунге – Кутты 4-го порядка точности с постоянным шагом h:

$$k1 = f(tn, yn),$$
$$k2 = f(tn + h/3, yn + h/3k1),$$
$$k3 = f(tn + 2/3h, yn - h/3k1 + hk2),$$
$$k4 = f(tn + h, yn + hk1 - hk2 + hk3),$$
$$yn + 1 = yn + h(k1 + 3k2 + 3k3 + k4)/8.$$

3. Протестируйте программу на примере системы уравнений

$$y'1 = y2,$$
$$y'2 = -2 * y1\hat{}2(1 - 4t * 2y1),$$

на отрезке [0, 5] с точным решением (проверьте!)

$$y1 = 1/(1 + t2),$$
$$y2 = -2t/(1 + t2)2.$$

4. Для тестовой задачи постройте графики зависимости максимальной погрешности решения e и e/h4 от выбранного шага h. Какие выводы можно сделать из полученных графиков?
5. Решите систему уравнений (3), (4) при помощи разработанной программы. Рассчитайте траектории полета снаряда при следующих исходных данных: m = 45 кг., C = 0.25, ρ = 1.29 кг./м3, S = 0.35 м2, g = 9.81, v0 = 60 м/сек. и значениях начального угла θ0 ∈ [20, 70]. При каком значении θ0 дальность полета L = L(θ0) будет максимальной. Приведите характерные графики траекторий полета в координатах (x, y), соответствующие им графики v(t), θ(t), а также L(θ0).

Задание 1

Проверим правильность вывода системы уравнений:
На снаряд действуют две силы, а именно сила тяжести и сила лобового сопротивления

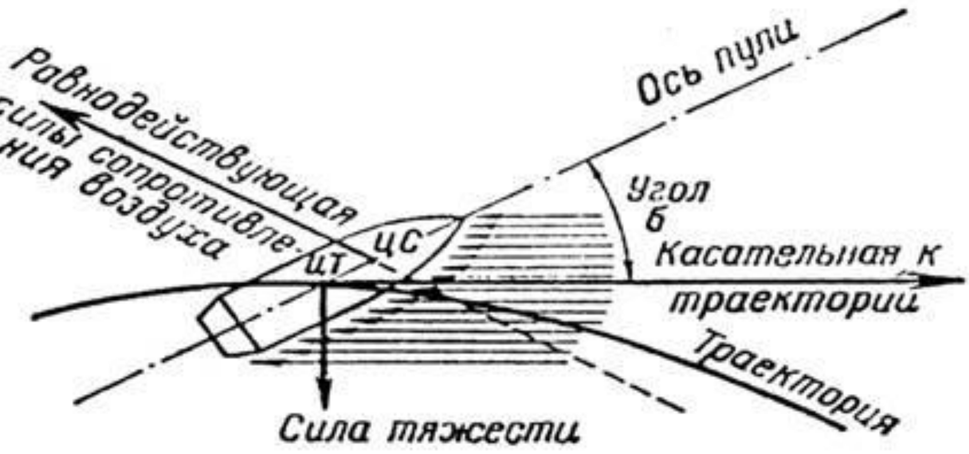

Рис. 1.1 Действующие силы на снаряд

В следствие чего и получаем нашу систему равнений:

$$x'' = -C\rho SV^2 \cos(\theta)/2m$$
$$y'' = -C\rho SV^2 sin(\theta)/2m - g$$

С начальными условиями:

$$x(0) = 0, y(0) = 0, \theta(0) = \theta_0\ V(0) = V_0$$

Задание 2

Для решения данной задачи нам потребуется метод Рунге-Кутта 4-ого порядка. Программа имеет векторный вид, поэтому нет проблем с решением системы уравнений. Часть кода, отвечающий за решение системы:

```
function y = RK4(odefun, tspan, y0)
    t = tspan(:);
    h = t(2)-t(1);
    N = length(t);

    y = zeros(N, numel(y0));
    y(1, :) = y0(:)';

    k = zeros(4, numel(y0));

    for i = 1:(N-1)
        k(1, :) = odefun(t(i), y(i,:));
        k(2, :) = odefun(t(i) + (h/3), y(i,:) + (h/3)*k(1,:));
        k(3, :) = odefun(t(i) + (2*h/3), y(i,:) - (h/3)*k(1,:) + h*k(2,:));
        k(4, :) = odefun(t(i) + h, y(i,:) + h*k(1,:) - h*k(2,:) + h*k(3,:));

        y(i+1, :) = y(i, :) + (h/8)*(k(1,:)+3*k(2,:)+3*k(3,:)+k(4,:));

    end
end
```

На вход подается система уравнений в виде функции(odefun), рассматриваемый временной промежуток(tspan) и начальные условия(y0)

Задание 3

Протестируем метод Рунге-Кутта, описанный выше. Для этого решим тестовую систему уравнение, для которой известно точное решение:

$$y'1 \ = \ y2,$$
$$y'2 \ = \ -2 * y1\text{^}2(1 \ - \ 4t * 2y1),$$

Точное решение:

$$y1 \ = \ 1/(1 \ + \ t2),$$
$$y2 \ = \ -2t/(1 \ + \ t2)2.$$

Временной отрезок t =[0;5]. Начальные условия берем из точного решения. Шаг h= 0.01

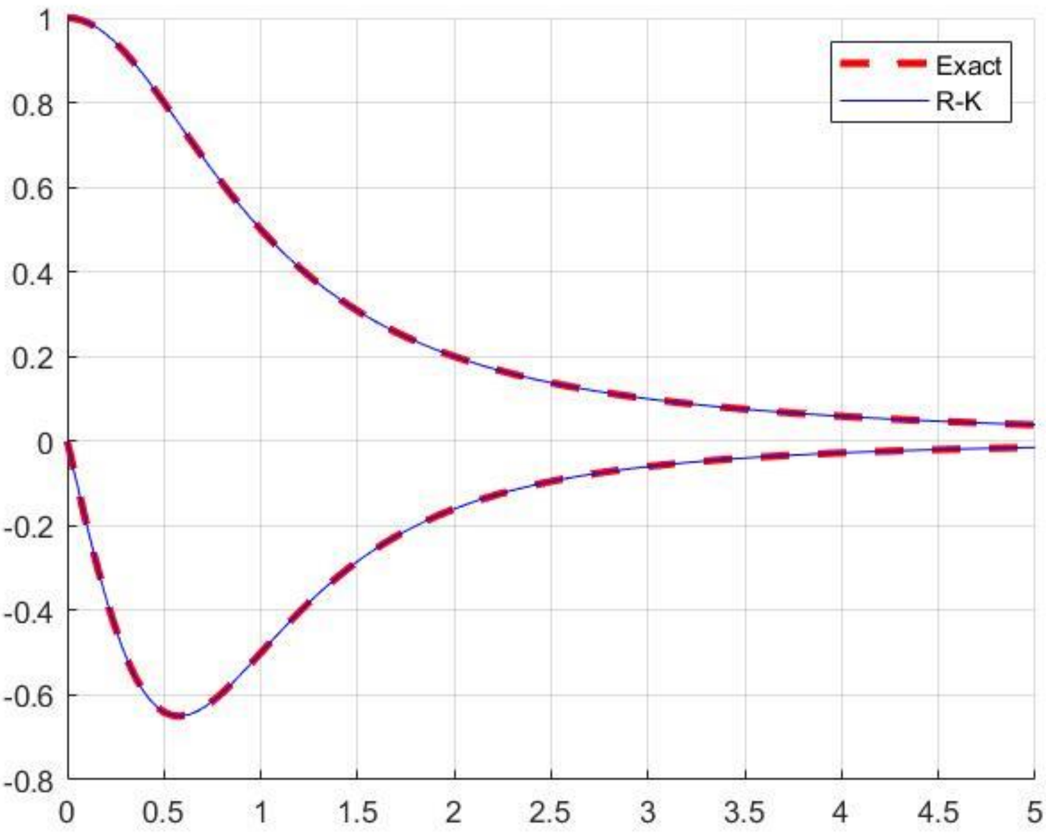

Рис. 3.1 График решения системы диф. Уравнений методом Р-К

Из графика видим, что вычисление методом Рунге-Кутта 4-ого порядка близко к точному решению.

Задание 4

Построим для тестовой задачи графики зависимости погрешности e от шага:

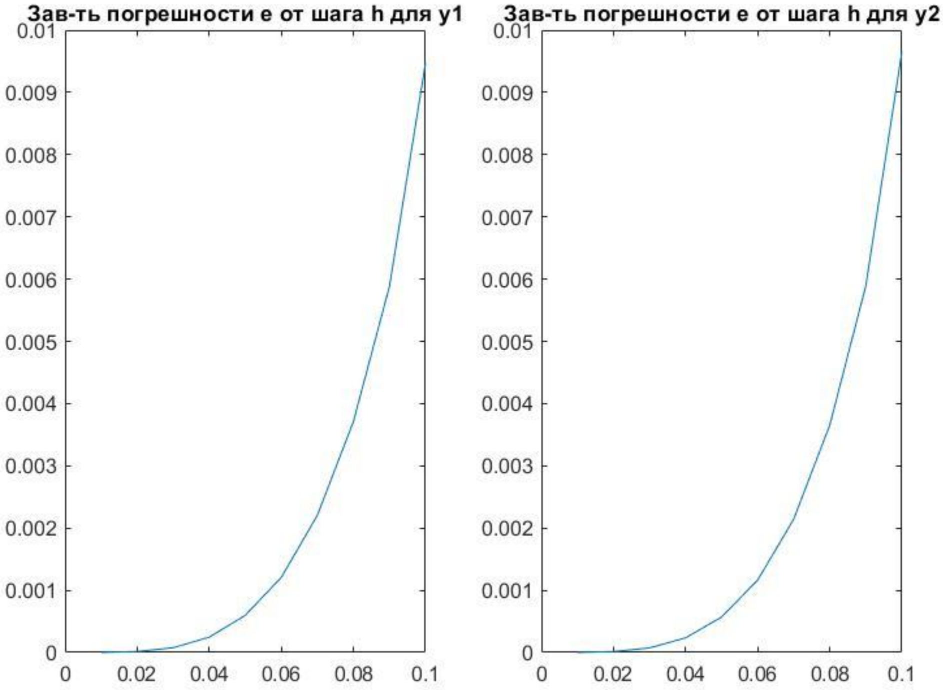

Рис. 4.1 График зависимости погрешности e от шага h для системы уравнений

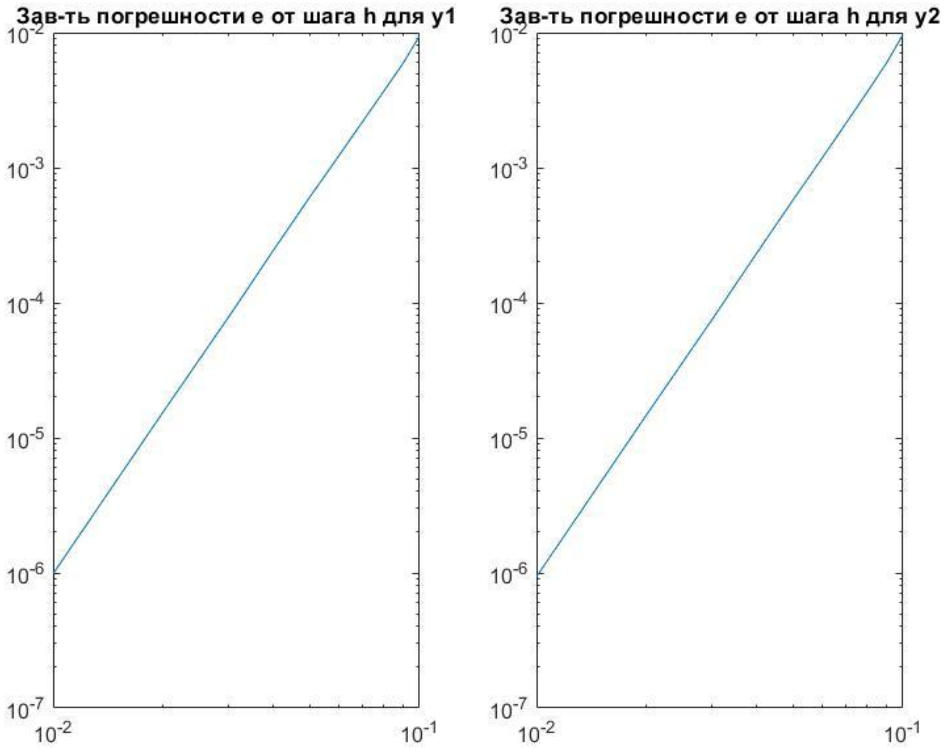

Рис. 4.2 График зависимости погрешности e от шага h для системы уравнений в двойных логарифмических координатах

Из первого графика видим, что зависимость погрешности от шага является степенной, второй график подтверждает это.

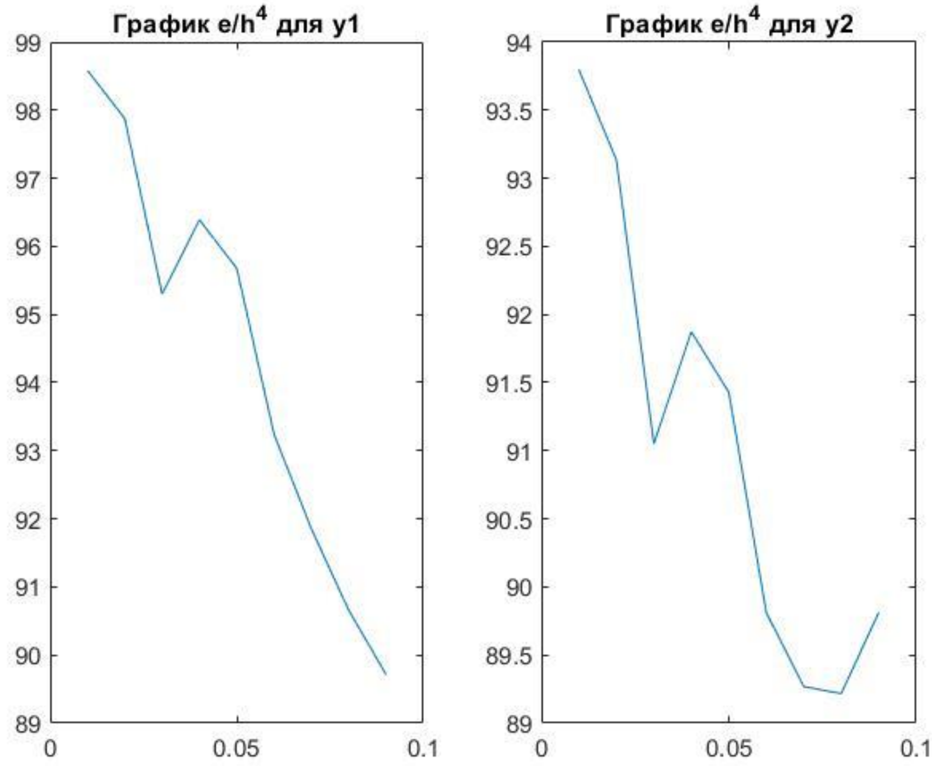

Рис. 4.3 График зависимости e/h 4 от h

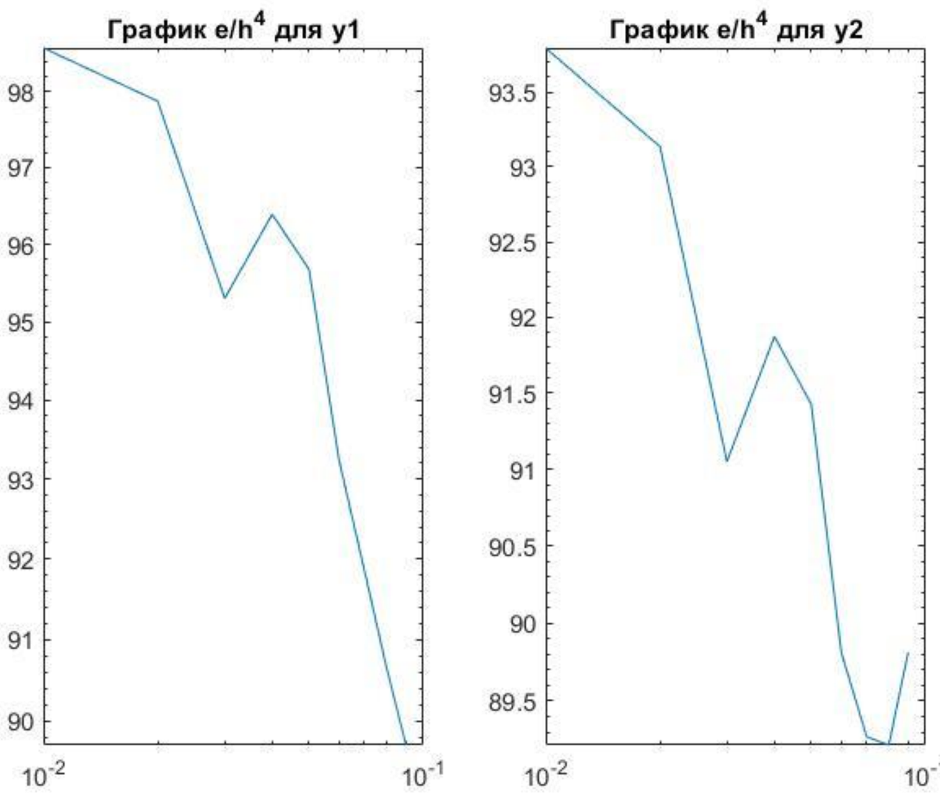

Рис. 4.4 График зависимости e/h 4 от h в двойных логарифмических

координатах

Так же можем удостовериться, что порядок точности равен 4. Для этого вычислим показатель α.

$$\alpha i = log10(ei + 1/ei)/log10(hi + 1/hi),$$
$$i = 1,\ldots,K - 1.$$

В нашем случае среднее α примерно равняется 4. Что соответствует порядку нашего метода Рунге-Кутта

Задание 5

Теперь решим систему уравнений (3) и (4). Так как функция является векторной – для нас не составит проблем решить данную систему. Достаточно передать функцию, определяющую четыре уравнения + начальные условия. Только нам понадобится немного модифицировать функцию, отвечающую за решение системы. Так как нам неизвестно время полета снаряда- прекратим вычисление после того, как снаряд приземлится (нужно задать условие). Функция Р-К для данной задачи:

```
function [y,t] = RK4(odefun, y0, h_t)
    t(1)=0;
    h = h_t;
    y(1, :) = y0(:)';
    i=1;
    while true
        k(1, :) = odefun(t(i), y(i,:));
        k(2, :) = odefun(t(i) + (h/3), y(i,:) + (h/3)*k(1,:));
        k(3, :) = odefun(t(i) + (2*h/3), y(i,:) - (h/3)*k(1,:) + h*k(2,:));
        k(4, :) = odefun(t(i) + h, y(i,:) + h*k(1,:) - h*k(2,:) + h*k(3,:));

        y(i+1, :) = y(i, :) + (h/8)*(k(1,:)+3*k(2,:)+3*k(3,:)+k(4,:));

        if  y(end,2)<0 || y(end,1)<0
            y(end,:)=[];
            break
        else
            i=i+1;
            t(i)=t(i-1) + h;
        end
    end
end
```

В качестве параметров она принимает так же функцию, определяющую наше уравнение, начальные условия и шаг по времени вместо отрезка времени. Так же метод возвращается сетку времени полета снаряда(пригодится для графиков).

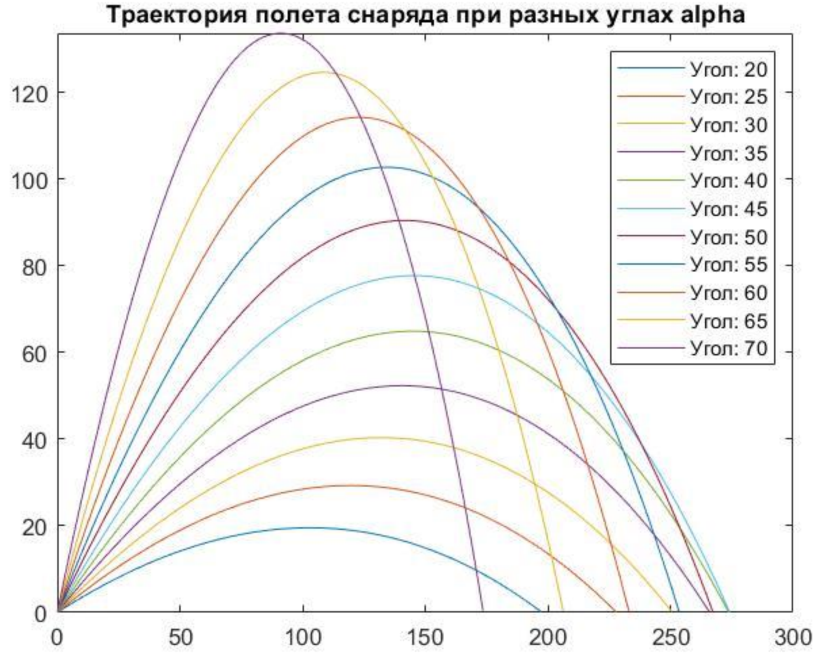

Рис. 5.1 График траектории снаряда при разных углах
Из Рис. 5.1 видим, что искомый угол лежит в промежутке 40-45 градусах. Продолжая вычисления приходим к выводу, что искомый угол равняется 42.9 градусам, то есть при данном угле максимальная дальность полета снаряда. При этом максимальная дальность полета равняется 274.75 метрам.

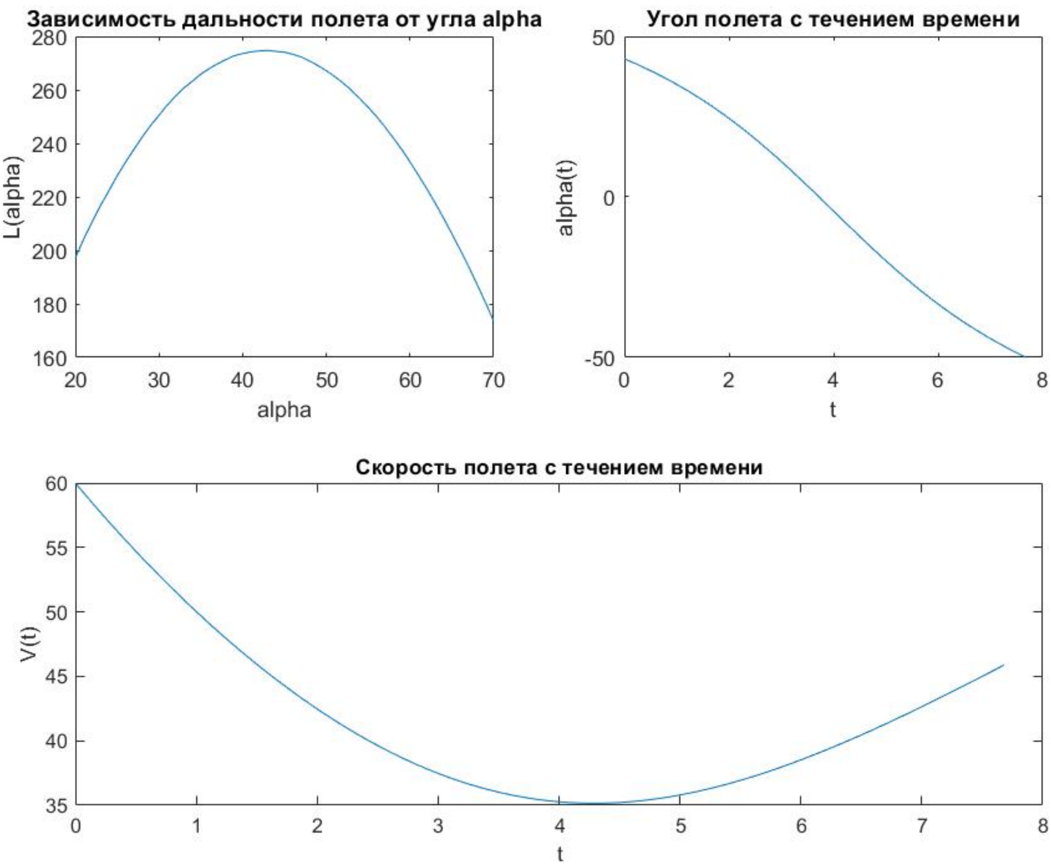

Рис. 5.2 Графики траектории полета от угла/угла полета с течением времени(для угла = 42.9)/скорости полета с течением времени(для угла = 42.9)

Вывод

Мною была выполнена курсовая работа по решению системы дифференциальных уравнений методом Рунге-Кутта. Для этого были рассмотрены две задачи: тестовая и модельная, представляющая собой полет снаряда, на который действуют силы. При решении тестовой системы стало понятно, что есть зависимость, причем степенная, между погрешностью e и шагом h.

Были проведены испытания полета снаряда при разных углах и получен оптимальный угол, при котором дальность полета максимальна. Решение модальной задачи сопровождается графиками, отражающими поведение снаряда во время полета.

Литература

1. Дж.Ортега, У.Пул. Введение в численные методы решения дифференциальных уравнений. —М.: Наука, 1986.
2. Самарский А.А., Гулин А.В. Численные методы. — М.: Наука, 1989.
3. Бахвалов Н.С., Жидков Н.П., Кобельков Г.М. Численные методы.— М.: Наука, 1987.

Листинг

Основная задача:

```
clear all; close all; clc
m=45; C=0.25; p=1.29; S=0.35; g=9.81; v0=60;

angles=[20:1:70]; rad_angles=deg2rad(angles);
N=length(angles)

Lmax=0;
Angle_max=0;
legend_arr=cell(N,1);

for i=1:N
    h_t=0.01;
    V0=[0; 0; v0; rad_angles(i)];
    fun=@(t,y) [y(3)*cos(y(4)); y(3)*sin(y(4));-C*p*S*y(3).^2./(2*m)-g*sin(y(4)); -
g*cos(y(4))./y(3)];

    [y,t] = RK4(fun, V0, h_t);
    plot(y(:,1)',y(:,2)'); ylim([0 inf])
    hold on
```

```
    legend_arr{i}=append('Угол: ', num2str(angles(i)));

    L(i)=y(end,1);
    buf=y(end,1);
    if buf>Lmax
        Lmax=buf;
        Angle_max = rad2deg(rad_angles(i));
        Varr_max=y(:,3);
        angle_max_arr=rad2deg(y(:,4));
        t_best=t;
    end
end
    title('Траектория полета снаряда при разных углах alpha')
    xlabel('x(t)')
    ylabel('y(t)')
    legend(legend_arr)

['Максимальная дальность полета = ' num2str(Lmax)]
['Искомый угол = ' num2str(Angle_max)]

    figure
    subplot(2,2,1)
    plot(angles, L)
    title('Зависимость дальности полета от угла alpha')
    xlabel('alpha')
    ylabel('L(alpha)')

    subplot(2,2,2)
    plot(t_best,angle_max_arr)
    title('Угол полета с течением времени')
    xlabel('t')
    ylabel('alpha(t)')

    subplot(2,2,3:4)
    plot(t_best,Varr_max)
    title('Скорость полета с течением времени')
    xlabel('t')
    ylabel('V(t)')

%     y(end,1)
%     y(end,2)
```

Основной Р-К:

```
function [y,t] = RK4(odefun, y0, h_t)
    t(1)=0;
    h = h_t;
    y(1, :) = y0(:)';
    i=1;
    while true
        k(1, :) = odefun(t(i), y(i,:));
        k(2, :) = odefun(t(i) + (h/3), y(i,:) + (h/3)*k(1,:));
        k(3, :) = odefun(t(i) + (2*h/3), y(i,:) - (h/3)*k(1,:) + h*k(2,:));
        k(4, :) = odefun(t(i) + h, y(i,:) + h*k(1,:) - h*k(2,:) + h*k(3,:));

        y(i+1, :) = y(i, :) + (h/8)*(k(1,:)+3*k(2,:)+3*k(3,:)+k(4,:));

        if  y(end,2)<0 || y(end,1)<0
            y(end,:)=[];
            break
        else
            i=i+1;
            t(i)=t(i-1) + h;
```

```
        end
    end
end
```

Тестовая задача:

```
clc
close all
clear all

odefun = @(t, y) [y(2); -2*y(1)^2*(1-4*t^2*y(1))];
y0 = [1;0];

% %reshenie funkciei ode45()
% [T, Y] = ode45(odefun, [0,5], y0);

%Решение методом Рунге-Кутта 4-го порядка
t = 0:0.01:5;
y = RK4_test(odefun, t, y0);

%Точное решение
 y_exact(1,:)=1./(1+t.^2);
 y_exact(2,:)=-2*t./((1+t.^2).^2);


hold on;
grid on
% plot(T, Y,'y');
plot(t, y, 'r--', 'linewidth', 2.5)
plot(t, y_exact,'b-', 'linewidth', 0.5);
legend('Exact','', 'R-K','')


h=0.01:0.01:0.09;
for i=1:length(h)
    t=0:h(i):5;
    y = RK4_test(odefun, t, y0);
    y_exact1 =1./(1+t.^2);
    y_exact2 =-2*t./((1+t.^2).^2);
    y_exact = [y_exact1;y_exact2]
    err(i,:)=max(abs(y-y_exact'));
    err_divide(i,:)=max(abs(y-y_exact'))/(h(i)^4);
end
    figure
    subplot(1,2,1)
    plot(h,err(:,1))
    title('Зав-ть погрешности e от шага h для y1')
    subplot(1,2,2)
    plot(h,err(:,2))
    title('Зав-ть погрешности e от шага h для y2')

    figure
    subplot(1,2,1)
    loglog(h,err(:,1))
    title('Зав-ть погрешности e от шага h для y1')
    subplot(1,2,2)
    loglog(h,err(:,2))
    title('Зав-ть погрешности e от шага h для y2')

    figure
    subplot(1,2,1)
    plot(h,err_divide(:,1))
    title('График e/h^4 для y1')
```

```
    subplot(1,2,2)
    plot(h,err_divide(:,2))
    title('График e/h^4 для y2')

    figure
    subplot(1,2,1)
    loglog(h,err_divide(:,1))
    title('График e/h^4 для y1')
    subplot(1,2,2)
    loglog(h,err_divide(:,2))
    title('График e/h^4 для y2')
    for i=2:length(h)
        alpha(i-1) = log10(err(i, 1)./err(i-1,1))./log10(h(i)./h(i-1));
    end
mean(alpha)
```

Тестовый Р-К:

```
function y = RK4(odefun, tspan, y0)

    t = tspan(:);
    h = t(2)-t(1);
    N = length(t);

    y = zeros(N, numel(y0));
    y(1, :) = y0(:)';

    k = zeros(4, numel(y0));

    for i = 1:(N-1)
        k(1, :) = odefun(t(i), y(i,:));
        k(2, :) = odefun(t(i) + (h/3), y(i,:) + (h/3)*k(1,:));
        k(3, :) = odefun(t(i) + (2*h/3), y(i,:) - (h/3)*k(1,:) + h*k(2,:));
        k(4, :) = odefun(t(i) + h, y(i,:) + h*k(1,:) - h*k(2,:) + h*k(3,:));

        y(i+1, :) = y(i, :) + (h/8)*(k(1,:)+3*k(2,:)+3*k(3,:)+k(4,:));

    end
end
```