

## К работе допущены

**Работа выполнена**

**Отчет принят**

Смоделировать взлёт ракеты с Земли в двумерном пространстве. Произвести расчёты координат ракеты, скорости ракеты по  $x$  и по  $y$ , модуля скорости, массы топлива – посекундно, а также написать программу для вывода этих величин посекундно с помощью **ВХОДНЫХ ДАННЫХ**.

# Моделирование. Задача 1. Расчет.

Колыбеб, Маммалов,  
М3115

$\omega$  - скорость вращения газа

$V$  - скорость ракеты

$V_x$  -  $V$  по оси  $Ox$

$V_y$  -  $V$  по оси  $Oy$

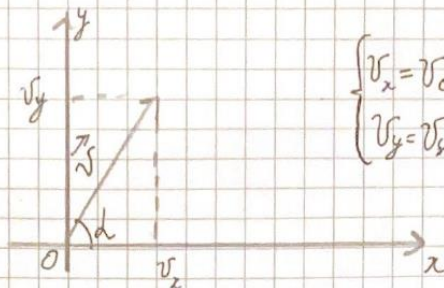
$\alpha$  - угол наклона ракеты

$M$  - масса ракеты

$m$  - масса топлива

$c$  - скорость сгорания

$t$  - время полета



$$\begin{cases} V_x = V \cos \alpha \\ V_y = V \sin \alpha \end{cases}$$

1. По БУ:

$$OX: (M - ct) \frac{dV_x}{dt} = cW \cos \alpha$$

$$OY: (M - ct) \frac{dV_y}{dt} = cW \sin \alpha - \frac{(M - ct) g R_z^2}{(R_z + h)^2}$$

2. Приращение  $V_x$  и  $V_y$  с учетом изменения  $\alpha$ :

$$\left. \begin{aligned} \frac{dV_x}{dt} &= \frac{dV}{dt} \cos \alpha - V \sin \alpha \frac{d\alpha}{dt} \\ \frac{dV_y}{dt} &= \frac{dV}{dt} \sin \alpha + V \cos \alpha \frac{d\alpha}{dt} \end{aligned} \right\} \rightarrow$$

$$\Rightarrow \begin{cases} (M - ct) \left( \frac{dV}{dt} \cos \alpha - V \sin \alpha \frac{d\alpha}{dt} \right) = cW \cos \alpha \\ (M - ct) \left( \frac{dV}{dt} \sin \alpha + V \cos \alpha \frac{d\alpha}{dt} \right) = cW \sin \alpha - \frac{(M - ct) g R_z^2}{(R_z + h)^2} \end{cases}$$

$$3) \begin{cases} \frac{dV}{dt} - V \tan \alpha \frac{d\alpha}{dt} = \frac{cW \cos \alpha}{(M - ct) \cos \alpha} = \frac{cW}{(M - ct)} \\ \frac{dV}{dt} + V \cot \alpha \frac{d\alpha}{dt} = \frac{cW \sin \alpha}{(M - ct) \sin \alpha} - \frac{(M - ct) g R_z^2}{(M - ct) (R_z + h)^2 \sin \alpha} = \frac{cW}{(M - ct)} - \frac{g R_z^2}{(R_z + h)^2 \sin \alpha} \end{cases}$$

$$\frac{d\alpha}{dt} V \cot \alpha + V \tan \alpha \frac{d\alpha}{dt} = \frac{g R_z^2}{(R_z + h)^2 \sin \alpha} \cdot \frac{d\alpha}{dt} \cdot \frac{V}{\cos \alpha \sin \alpha} = \frac{g R_z^2}{(R_z + h)^2 \sin \alpha} = \vec{g}$$

$$\frac{d\alpha}{dt} = -\frac{g \cos \alpha}{V}$$

$$4) \frac{dV}{dt} = V \tan \alpha \frac{d\alpha}{dt} + \frac{cW}{(M - ct)} = \frac{cW}{(M - ct)} - \vec{g} \sin \alpha$$

$$\frac{dV_x}{dt} = \frac{dV}{dt} \cos \alpha - V \sin \alpha \frac{d\alpha}{dt} = \frac{cW \cos \alpha}{M - ct} - g \sin \alpha \cos \alpha + g \sin \alpha \cos \alpha = \frac{cW \cos \alpha}{M - ct}$$

$$\frac{dV_y}{dt} = \frac{dV}{dt} \sin \alpha + V \cos \alpha \frac{d\alpha}{dt} = \frac{cW \sin \alpha}{M - ct} - \vec{g}$$

$$X = X_0 + V_x \cdot \frac{t^2}{2} = X_0 + \frac{t^2}{2} \cdot \frac{cW \cos \alpha}{(M - ct)}$$

$$y = h = y_0 + V_y \cdot \frac{t^2}{2} - y_0 + \frac{t^2}{2} \left( \frac{cW \sin \alpha}{(M - ct)} - \vec{g} \right)$$

Код:

```
import math
import matplotlib.pyplot as plt

class Rocket:
    def __init__(self, _x0=0, _y0=0, _angle=0, _rocket_mass=0, _fuel_mass=0, _gas_outflow=0, _fuel_burn=0):
        self.earth_mass, self.g, self.x0, self.y0, self.angle, \
            self.rocket mass, self.fuel mass, self.gas outflow, self.fuel burn = \
            5.97e+20, 9.81, _x0, _y0, _angle, _rocket_mass, _fuel_mass, _gas_outflow, _fuel_burn

    def x(self, time):
        return self.x0 + time * time * (self.fuel_burn * self.gas_outflow / (
            self.rocket mass + self.fuel mass - self.fuel burn * time)) * math.cos(self.angle * math.pi
/ 180) / 2

    def y(self, time):
        return self.y0 + time * time * ((self.fuel burn * self.gas outflow / (
            self.rocket mass + self.fuel mass - self.fuel burn * time)) * math.sin(
            self.angle * math.pi / 180) - self.g) / 2

    def speedX(self, time):
        return self.fuel burn * self.gas outflow / (
            self.rocket mass + self.fuel mass - self.fuel burn * time) * math.cos(
            self.angle * math.pi / 180) * time

    def speedY(self, time):
        return ((self.fuel burn * self.gas outflow / (
            self.rocket mass + self.fuel mass - self.fuel burn * time)) * math.sin(
            self.angle * math.pi / 180) - self.g) * time

    def speed(self, time):
        return math.fabs(((self.fuel burn * self.gas outflow / (
            self.rocket mass + self.fuel mass - self.fuel mass * time)) - self.g) * time)

    def fuelMass(self, time):
        return self.fuel mass - self.fuel burn * time

rocket = Rocket(0, 0, 70, 100000, 70000, 3000, 2000)
speed = []
mass = []
traekt = []
for t in range(35):
    '''print("Секунд со старта: ", t, '\n', "Координата по оси Ох: ", rocket.x(t), '\n', "Координата по оси
Оу: ",
        rocket.y(t), '\n', "Скорость по оси Ох: ", rocket.speedX(t), '\n', "Скорость по оси Оу: ",
        rocket.speedY(t), '\n', "Модуль скорости: ", rocket.speed(t), '\n', "Масса топлива: ",
rocket.fuelMass(t),
        '\n')'''
    traekt.append([rocket.x(t), rocket.y(t)])
    mass.append([t, rocket.fuelMass(t)])
    speed.append([rocket.speedX(t), rocket.speedY(t)])

pt.title('Траектория полета ракеты')
pt.plot([i[0] for i in traekt], [i[1] for i in traekt])
pt.grid()
pt.savefig('traekt', )
pt.show()

pt.title('Скорость ракеты по осям')
pt.xlabel('Скорость по оси Ох')
pt.ylabel('Скорость по оси Оу')
pt.grid()
pt.plot([i[0] for i in speed], [i[1] for i in speed])
pt.savefig('speed', )
pt.show()

pt.title('Изменение массы ракеты')
pt.xlabel('Масса, кг')
pt.ylabel('Время, с')
pt.grid()
pt.plot([i[0] for i in mass], [i[1] for i in mass])
pt.savefig('mass', )
pt.show()
```

Графики:

Входные данные:

$x_0 = 0$

$y_0 = 0$

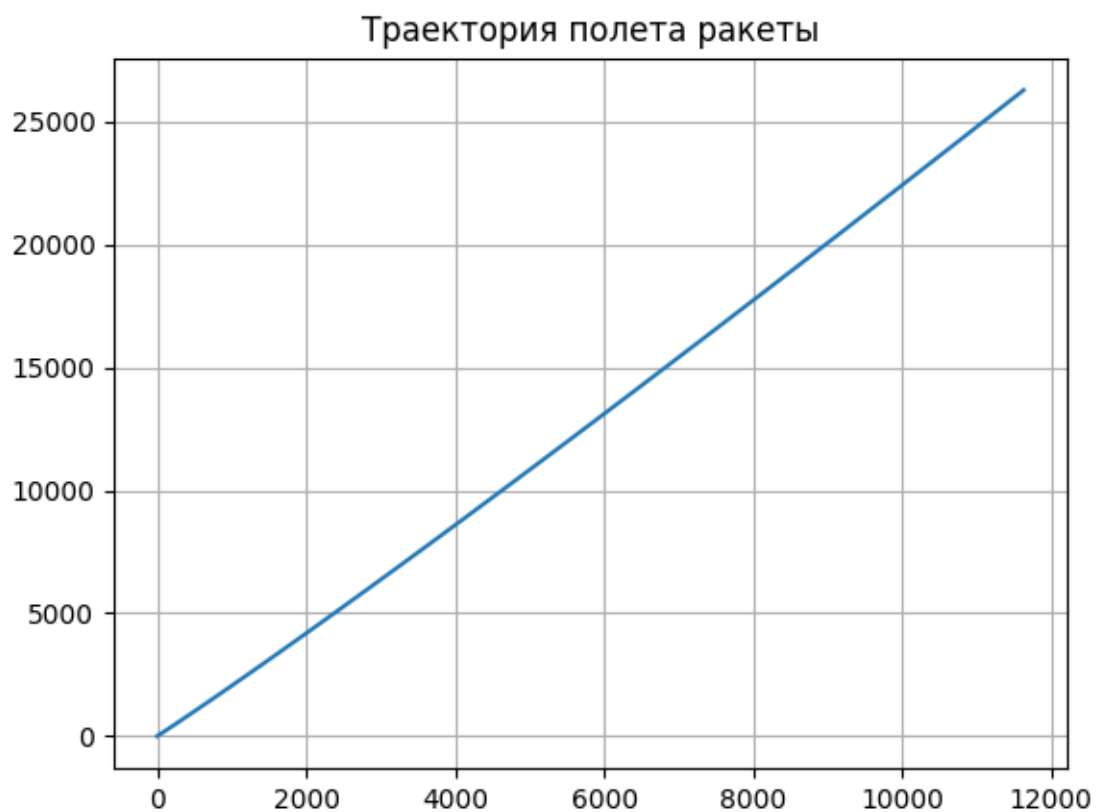
Угол старта ракеты – 70 градусов

Масса ракеты – 100 000 кг

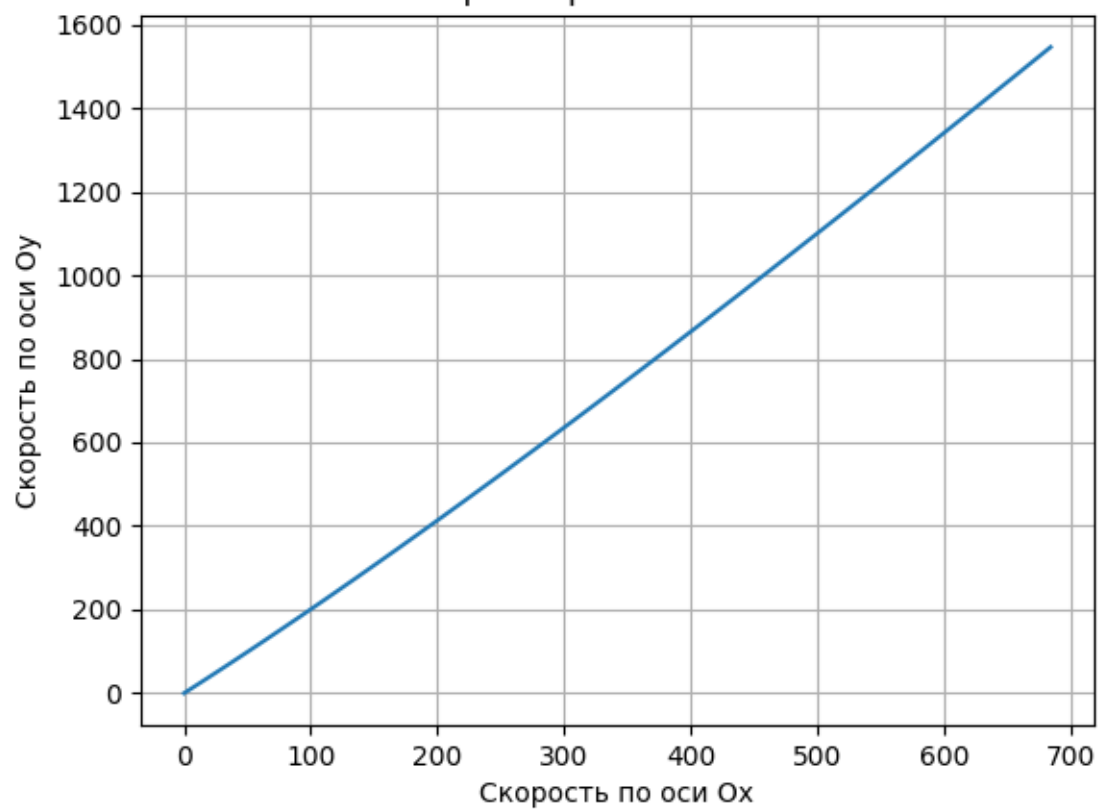
Масса топлива – 70 000 кг

Скорость истечения газа из сопла ракеты – 3000 м/с

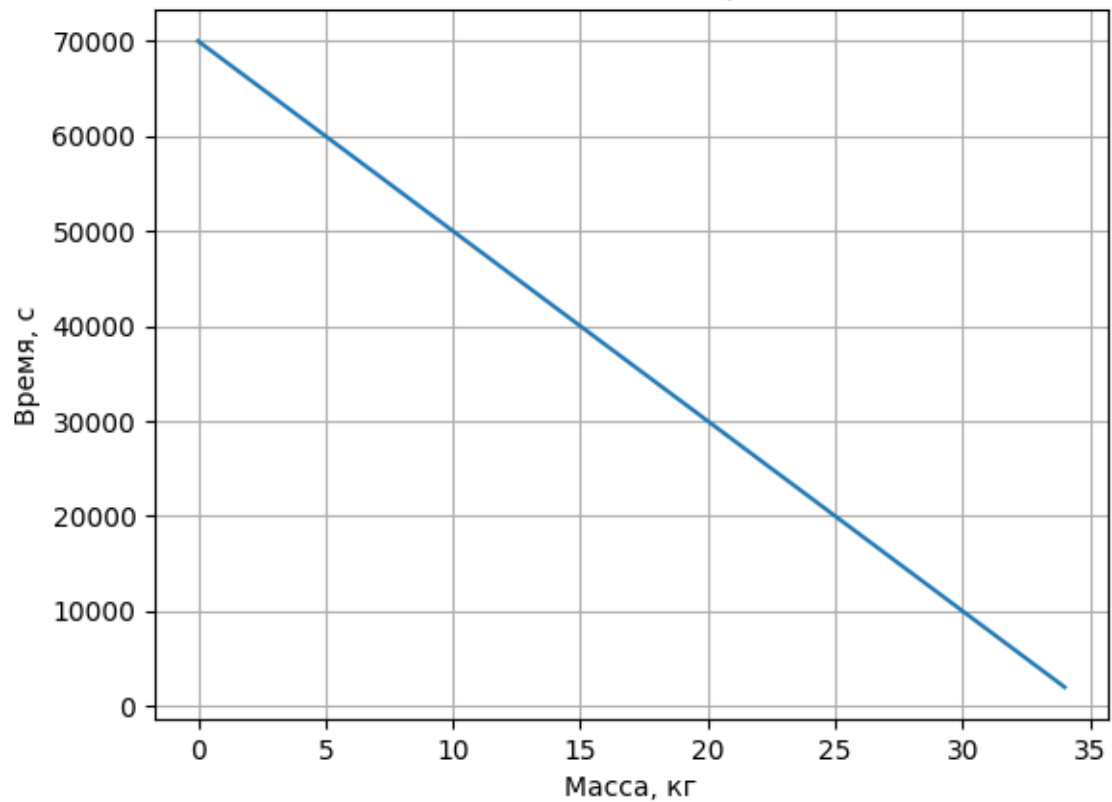
Скорость сжигания топлива – 2000 кг/с



Скорость ракеты по осям



Изменение массы ракеты



## **Заключение:**

Мы вывели формулы для нахождения требуемых выходных данных. Затем на основе этих формул была написана программа на питоне, принимающая на вход данные о ракете и отправляющая на выход посекундно координаты, проекции скорости полёта ракеты на оси  $Ox$  и  $Oy$ , модуль скорости и остаток топлива. Для проверки работы программы мы придумали тривиальную ракету. По выходным координатам ракеты посекундно был построен график траектории движения ракеты