

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Дальневосточный федеральный университет»**

**ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ**

**Департамент математического и компьютерного моделирования**

**ОТЧЕТ**

К лабораторной работе №4 по дисциплине

«Математическое и компьютерное моделирование»

Направление подготовки   
01.03.02 «Прикладная математика и информатика»

|  |  |  |
| --- | --- | --- |
|  |  | Выполнил студент гр.  Б9122-01.03.02сп  Носков Я. В. \_\_\_\_\_\_\_\_\_\_\_  (Ф.И.О.) (подпись)  Проверил профессор д.ф.-м.н.  Пермяков М. С. \_\_\_\_\_\_\_\_ |
|  |  | (Ф.И.О.) (подпись)  « » 2025г. |
|  |  |  |

**г. Владивосток**

**2025**

**Оглавление**

[Введение 3](#_Toc201156953)

[1. Построение математической модели 4](#_Toc201156954)

[1.1 Уравнение движения 4](#_Toc201156955)

[1.2 Вывод модели движения материальной точки 4](#_Toc201156956)

[2. Анализ модели 6](#_Toc201156957)

[2.1 Существование и единственность 6](#_Toc201156958)

[2.2 Сохраняемая величина 6](#_Toc201156959)

[2.3 Гармоническая природа скорости 7](#_Toc201156960)

[2.4 Траектория движения 7](#_Toc201156961)

[3. Вычислительные эксперименты 9](#_Toc201156962)

[3.1 Влияние на траекторию 9](#_Toc201156963)

[3.2 Влияние начальной скорости на траекторию 9](#_Toc201156964)

[3.3 Изменение энергии 10](#_Toc201156965)

[Заключение 12](#_Toc201156966)

[Приложения 14](#_Toc201156967)

# Введение

Изучение движения материальной точки в системах отсчёта, связанных с вращающимися телами, представляет собой важную задачу как в теоретической, так и в прикладной механике. Одним из ключевых эффектов, возникающих в таких системах, является сила Кориолиса — инерционная сила, проявляющаяся при движении тела в неинерциальной (вращающейся) системе отсчёта. Несмотря на то, что в реальных условиях на тело действуют также центробежные и внешние силы, задача, в которой рассматривается только влияние силы Кориолиса, представляет значительный интерес, так как позволяет выделить и проанализировать чистый эффект этой силы на поведение системы.

Ярким примером действия силы Кориолиса в природе является формирование циклонов и антициклонов в атмосфере Земли. Из-за вращения планеты воздушные массы, перемещающиеся в северном и южном полушариях, отклоняются вправо или влево соответственно. Это приводит к закручиванию атмосферных вихрей и существенным изменениям в глобальной и локальной погодной динамике. Аналогичные эффекты наблюдаются и в океанических течениях, а также в динамике снарядов и спутников.

Рассмотрение изолированного влияния силы Кориолиса позволяет упростить математическую модель и, в то же время, даёт важные качественные представления о поведении тел в системах, подобных атмосфере Земли или быстро вращающимся машинам. Особенно важно учитывать такие эффекты в задачах навигации, баллистики и моделирования динамики в ряде физических и технических приложений, включая гироскопические системы и планетарную атмосферу.

В данной работе будет построена и исследована математическая модель движения материальной точки в горизонтальной плоскости, перпендикулярной оси вращения. В модели учитывается исключительно сила Кориолиса, возникающая из-за вращения системы отсчёта. Центробежная и другие внешние силы предполагаются пренебрежимо малыми и исключаются из рассмотрения. Такой подход позволяет сосредоточиться на изучении геометрических и динамических свойств траекторий, возникающих под действием одной инерционной силы.

Целью настоящей работы является построение и исследование указанной модели, включая анализ устойчивости и характера движения точки, а также проведение численного эксперимента с использованием стандартных методов интегрирования обыкновенных дифференциальных уравнений. Результаты эксперимента позволят визуализировать поведение системы и подтвердить теоретические выводы.

# Построение математической модели

Рассмотрим ситуацию, в которой материальная точка перемещается по поверхности диска, вращающегося с постоянной угловой скоростью вокруг вертикальной оси. Поверхность диска считается абсолютно гладкой, то есть силы трения отсутствуют. Вследствие этого на точку не действуют внешние силы со стороны поверхности – остаются только силы инерции, порождаемые движением в неинерциальной системе отсчёта, связанной с диском.

## Уравнение движения

Для описания движения удобно перейти в вращающуюся систему координат, в которой диск остаётся неподвижным. Однако такая система не является инерциальной, поэтому в неё необходимо ввести дополнительные силы: кориолисову и центробежную. Поскольку задача касается чисто кинематической модели и силы трения отсутствуют, центробежная сила будет компенсироваться геометрически, а основное влияние будет оказывать сила Кориолиса.

Тогда уравнение движения в неинерциальной системе отсчёта выглядит следующим образом:

|  |  |
| --- | --- |
|  | *(1)* |

где:

* – вектор угловой скорости вращения диска;
* – вектор скорости точки в системе, вращающейся вместе с диском;
* – масса тела;
* – сила Кориолиса.

## Вывод модели движения материальной точки

Для дальнейшего анализа рассмотрим двумерное движение в плоскости диска. Пусть ось направлена вверх (вдоль оси вращения), тогда , а вектор скорости точки в плоскости имеет вид , где и – проекции скорости на оси и соответственно.

Вычислим векторное произведение:

|  |  |
| --- | --- |
|  | *(2)* |

Подставляя в уравнение движения и сокращая массу, получаем систему:

|  |  |
| --- | --- |
|  | *(3)* |

Это система дифференциальных уравнений, описывающая, как меняются проекции скорости частицы под действием силы Кориолиса.

Чтобы описать траекторию, введём уравнения для координат:

|  |  |
| --- | --- |
|  | *(4)* |

Таким образом, итоговая система уравнений, определяющая поведение тела на вращающемся диске без трения, имеет вид:

|  |  |
| --- | --- |
|  | *(5)* |

Добавим начальные условия при :

Полученная система уравнений совместно с начальными условиями является задачей Коши системы обыкновенных дифференциальных уравнений (ОДУ). Её решение позволяет определить как траекторию движения частицы (функции ), так и эволюцию её скорости во времени.

Эта система уравнений описывает колебательное поведение: производные компонент скорости пропорциональны друг другу с коэффициентами, зависящими от , что аналогично гармоническому движению.

Такая модель применяется в задачах, связанных с динамикой тел на вращающихся платформах, гироскопических устройствах, а также при анализе движения на вращающихся небесных телах (например, Земле), где эффект Кориолиса играет ключевую роль.

# Анализ модели

Полученная система из четырёх дифференциальных уравнений описывает поведение точки (частицы), движущейся по гладкой горизонтальной поверхности, связанной с вращающейся системой отсчёта. Влияние внешних сил исключено, а вся динамика определяется исключительно действием инерционной силы Кориолиса.

Рассмотрим подробнее свойства системы (5):

## Существование и единственность

Скоростную подсистему можно переписать как

|  |  |
| --- | --- |
|  | *(6)* |

Матрица имеет собственные значения .

Комплексные спряжённые корни чисто мнимы, следовательно в линейной системе это **центр**: все траектории скорости замкнуты.

**Теорема Коши-Липшица**

Если правая часть системы непрерывна и удовлетворяет условию Липшица по переменным на некотором множестве, то для любых начальных данных ​ существует единственное локальное (а при линейности — и глобальное) решение задачи Коши.

* В нашей системе .
* Все четыре функции линейны по с постоянными коэффициентами , значит непрерывны.
* Их частные производные по любому из аргументов — константы , следовательно, ограничены и непрерывны.

**Вывод:** для любых начальных условий существует единственное решение, определённое на всём интервале времени.

## Сохраняемая величина

Ищем комбинацию , постоянную вдоль решения. Рассмотрим производную:

|  |  |
| --- | --- |
|  | *(7)* |

Отсюда следует, что:

|  |  |
| --- | --- |
|  | *(8)* |

Это означает, что **скорость частицы в вращающейся системе остаётся постоянной по модулю**. Физически это отражает тот факт, что сила Кориолиса, будучи перпендикулярной к скорости, не выполняет работы и **не изменяет кинетическую энергию**.

## Гармоническая природа скорости

Из первых двух уравнений системы выводятся уравнения второго порядка:

|  |  |
| --- | --- |
|  | *(9)* |

Общее решение данных уравнений гармонических колебаний:

|  |  |
| --- | --- |
|  | *(10)* |

Здесь частота колебаний отражает то, что скорость «вращается» в плоскости с удвоенной угловой скоростью диска.

## Траектория движения

Подставляя найденные в и интегрируя от 0 до , получаем:

|  |  |
| --- | --- |
|  | *(11)* |

Введём центр Тогда:

|  |  |
| --- | --- |
|  | *(9)* |

**Вывод:** траектория частицы—это окружность радиуса ​​​.

# Вычислительные эксперименты

Для численного решения будем использовать функцию метод Рунге-Кутты четвертого порядка точности, а также solve\_ivp из модуля scipy, которая реализует метод Рунге-Кутты 4(5) порядка. Используемый язык программирования – python.

## Влияние на траекторию

Зафиксируем начальные условия: . Рассмотрим траектории движения при разных .

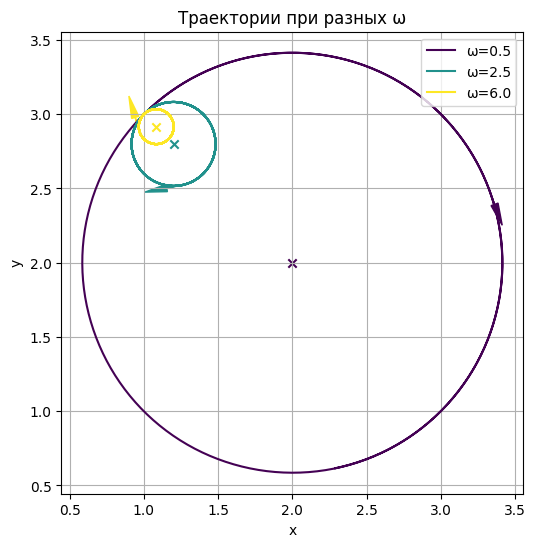


Рисунок 1 – Траектория при разных

На Рисунок 1 показаны траектории при фиксированных начальных условиях и разных . Все траектории – окружности, а радиус окружности обратно пропорционален угловой скорости

## **Влияние начальной скорости на траекторию**

Рассмотрим, как начальная скорость влияет на траекторию тела. Для этого зафиксируем: .

Построим графики траектории с разной начальной скоростью:

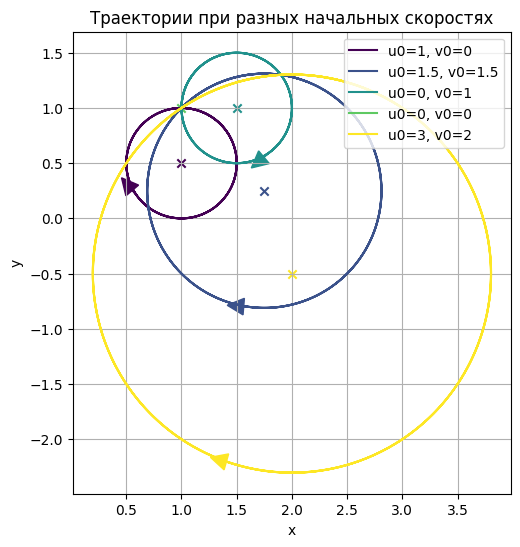


Рисунок 2 – траектория при разной начальной скорости

На Рисунке 2 видно, что начальная скорость влияет как на радиус окружности, так и на положении центра окружности. При этом чем больше начальная скорость, тем больше радиус, что подтверждает формулу .

## Изменение энергии

Построим график относительной ошибки энергии и график относительной энергии при разном разбиении по времени. Первый график – относительная ошибка энергии с разбиением по времени , второй график – относительная энергия с разбиением по времени, а третий – относительная ошибка энергии с разбиением .

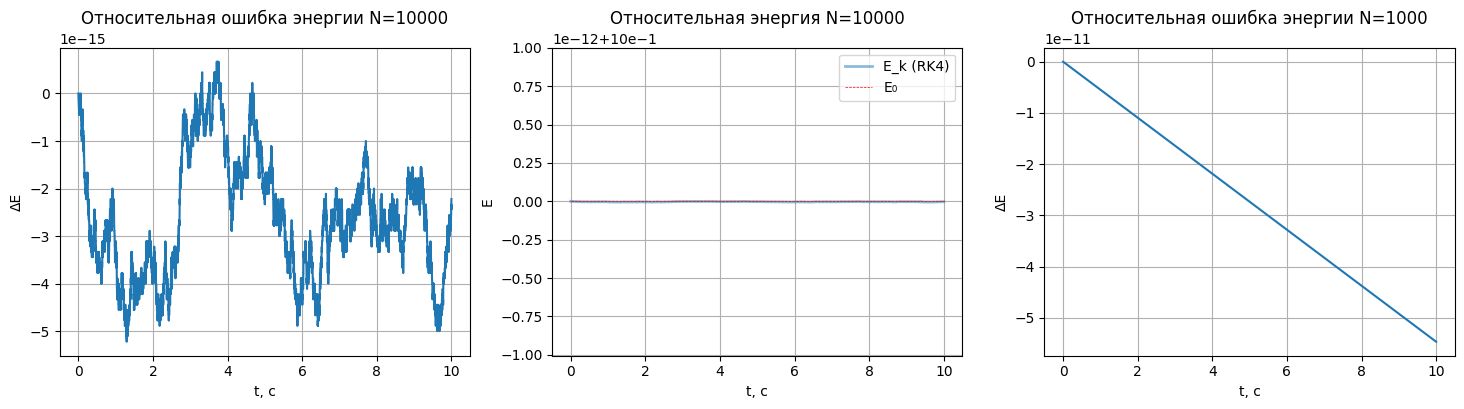


Рисунок 3 – ошибка энергии – график 1, относительная энергия – график 2, ошибка энергии (меньшее разбиение) – график 3.

Как видно на Рисунке 3, при разбиении по времени на большее количетсво шагов , относительная ошибка энергии колеблится около 0 , в то время как при относительно небольшом разбении по времени () закон сохранения энергии не выполняется,а график ошибки энергии линейно убывает. Это значит, что решение не обладает досточной точностью для сохранения энергии и нужно увеличивать разбение по времени.

# Заключение

В данной работе была исследована задача движения материальной точки в неинерциальной системе отсчёта, вращающейся с постоянной угловой скоростью. Целью являлось построение математической модели, её аналитический анализ и численное моделирование поведения системы.

На этапе **математического моделирования** были выведены дифференциальные уравнения, описывающие движение точки под действием силы Кориолиса в плоскости. При отсутствии других сил система сводится к однородной системе ОДУ первого порядка, где переменные скорости и положения связаны через угловую скорость .

Анализ модели показал, что движение происходит по окружности. Радиус окружности и координаты её центра зависят от начальных условий и угловой скорости. Также было показано, что в системе отсутствуют силы, изменяющие модуль скорости, поэтому величина сохраняется во времени — это интеграл движения.

Для **численного решения** использовался метод Рунге–Кутты 4порядка и функция solve\_ivp из библиотеки scipy. Была также реализована собственная версия классического метода РК4 для верификации результатов и оценки численной ошибки энергии.

В ходе **численных экспериментов** было проведено моделирование поведения системы при различных параметрах:

* **Эксперимент 1**: влияние величины угловой скорости. С увеличением радиус окружности уменьшается, а при уменьшении – увеличивается, что согласуется с теоретическим соотношением
* **Эксперимент 2**: влияние начальной скорости. Изменение начальных компонентов скорости влияет на радиус окружности и положение её центра, но форма траектории сохраняется.
* **Эксперимент 3**: сохранение энергии. Расчёт относительного изменения величины показал, что данная величина при небольшом количестве шагов по времени отклоняется от , однако если количество шагов по времени увеличить, то величина почти не отклоняется от , что подтверждает точность численного метода.

Таким образом, численные результаты полностью подтверждают аналитические выводы: система описывает консервативное вращательное движение, где скорость сохраняется, а траектория — замкнутая окружность.

Численные методы показали высокую точность, устойчивость и пригодность для моделирования движения в неинерциальных системах.

# Приложения

Приложения  
import numpy as np

import matplotlib.pyplot as plt

from scipy.integrate import solve\_ivp

def coriolis(t, z, omega):

    u, v, x, y = z

    return [2\*omega\*v, -2\*omega\*u, u, v]

def center(u0, v0, x0, y0, omega):

    return x0 + v0/(2\*omega), y0 - u0/(2\*omega)

t\_eval = np.linspace(0, 10, 500)

u0, v0, x0, y0 = 1.0, 1.0, 1.0, 3.0

omega\_vals = [0.5, 2.5, 6.0]

colors = plt.cm.viridis(np.linspace(0, 1, len(omega\_vals)))

plt.figure(figsize=(6,6))

for i, omega in enumerate(omega\_vals):

    sol = solve\_ivp(coriolis, [0, 10], [u0, v0, x0, y0],

                    args=(omega,), t\_eval=t\_eval)

    u\_vals, v\_vals, x\_vals, y\_vals = sol.y

    plt.plot(x\_vals, y\_vals, label=f"ω={omega}", color=colors[i])

    idx = 100

    vel = np.array([u\_vals[idx], v\_vals[idx]])

    arrow = vel / np.linalg.norm(vel) \* 0.1

    plt.arrow(x\_vals[idx], y\_vals[idx],

              arrow[0], arrow[1],

              head\_width=0.05, head\_length=0.15,

              fc=colors[i], ec=colors[i], color=colors[i])

    xc, yc = center(u0, v0, x0, y0, omega)

    plt.scatter(xc, yc, marker='x', color=colors[i])

plt.gca().set\_aspect('equal')

plt.title("Траектории при разных ω")

plt.xlabel("x"); plt.ylabel("y")

plt.grid(True)

plt.legend(loc='upper right')

plt.show()

mport numpy as np

import matplotlib.pyplot as plt

from scipy.integrate import solve\_ivp

# Используем те же функции coriolis и center, и временную сетку t\_eval

omega = 1.0

x0, y0 = 1.0, 1.0

init\_speeds = [(1,0), (1.5,1.5), (0,1), (0, 0), (3, 2)]

colors = plt.cm.viridis(np.linspace(0, 1, len(init\_speeds)))

plt.figure(figsize=(6,6))

for i, (u0, v0) in enumerate(init\_speeds):

    sol = solve\_ivp(coriolis, [0, 10], [u0, v0, x0, y0],

                    args=(omega,), t\_eval=t\_eval)

    u\_vals, v\_vals, x\_vals, y\_vals = sol.y

    xc, yc = center(u0, v0, x0, y0, omega)

    plt.plot(x\_vals, y\_vals, label=f"u0={u0}, v0={v0}", color=colors[i])

    idx = 100

    vel = np.array([u\_vals[idx], v\_vals[idx]])

    arrow = vel / np.linalg.norm(vel) \* 0.1

    plt.arrow(x\_vals[idx], y\_vals[idx],

              arrow[0], arrow[1],

              head\_width=0.15, head\_length=0.15,

              fc=colors[i], ec=colors[i], color=colors[i])

    plt.scatter(xc, yc, marker='x', color=colors[i])

plt.gca().set\_aspect('equal')

plt.title("Траектории при разных начальных скоростях")

plt.xlabel("x"); plt.ylabel("y")

plt.grid(True)

plt.legend(loc='upper right')

plt.show()

import numpy as np

import matplotlib.pyplot as plt

# Параметры задачи

omega = 2 \* np.pi / 10   # угловая скорость ≈0.628

u0, v0, x0, y0 = 1.0, 0.0, 0.0, 0.0

t0, t1, N = 0.0, 10.0, 10000   # интервал времени и число шагов

h = (t1 - t0) / N

# Определяем правые части

def f(z):

    u, v, x, y = z

    return np.array([2\*omega\*v, -2\*omega\*u, u, v])

# Сетка времени и массивы для хранения

ts = np.linspace(t0, t1, N+1)

zs = np.zeros((4, N+1))

energies = np.zeros(N+1)

rel\_error = np.zeros(N+1)

# Начальные условия

zs[:,0] = [u0, v0, x0, y0]

rel\_error[0] = 0.0

energies[0] = (u0\*\*2 + v0\*\*2)/(u0\*\*2 + v0\*\*2)

# Собственный RK4

for i in range(N):

    z = zs[:,i]

    k1 = f(z)

    k2 = f(z + 0.5\*h\*k1)

    k3 = f(z + 0.5\*h\*k2)

    k4 = f(z +   h\*k3)

    zs[:,i+1] = z + (h/6)\*(k1 + 2\*k2 + 2\*k3 + k4)

    u, v = zs[0,i+1], zs[1,i+1]

    rel\_error[i+1] = (u\*\*2 + v\*\*2 - (u0\*\*2 + v0\*\*2)) / (u0\*\*2 + v0\*\*2)

    energies[i+1] = (u\*\*2 + v\*\*2)/(u0\*\*2 + v0\*\*2)

# График относительной ошибки энергии

plt.figure(figsize=(18,4))

plt.subplot(1,3,1)

plt.plot(ts, rel\_error)

plt.title(f"Относительная ошибка энергии N={N}")

plt.xlabel("t, с")

plt.ylabel("ΔE")

plt.grid(True)

plt.subplot(1,3,2)

plt.plot(ts, energies, label='E\_k (RK4)',linewidth=2, alpha=0.5)

plt.hlines(energies[0], t0, t1, linestyles='dashed', label='E₀', linewidth=.5, color='r')

plt.title(f"Относительная энергия N={N}")

plt.xlabel("t, с")

plt.ylabel("E")

plt.grid(True)

plt.legend()

omega = 2 \* np.pi / 10   # угловая скорость ≈0.628

u0, v0, x0, y0 = 1.0, .0, .0, .0

t0, t1, N = 0.0, 10.0, 1000   # интервал времени и число шагов

h = (t1 - t0) / N

# Определяем правые части

# Сетка времени и массивы для хранения

ts = np.linspace(t0, t1, N+1)

zs = np.zeros((4, N+1))

rel\_error = np.zeros(N+1)

# Начальные условия

zs[:,0] = [u0, v0, x0, y0]

rel\_error[0] = 0.0

# Собственный RK4

for i in range(N):

    z = zs[:,i]

    k1 = f(z)

    k2 = f(z + 0.5\*h\*k1)

    k3 = f(z + 0.5\*h\*k2)

    k4 = f(z +   h\*k3)

    zs[:,i+1] = z + (h/6)\*(k1 + 2\*k2 + 2\*k3 + k4)

    u, v = zs[0,i+1], zs[1,i+1]

    rel\_error[i+1] = (u\*\*2 + v\*\*2 - (u0\*\*2 + v0\*\*2)) / (u0\*\*2 + v0\*\*2)

plt.subplot(1,3,3)

plt.plot(ts, rel\_error)

plt.title(f"Относительная ошибка энергии N={N}")

plt.xlabel("t, с")

plt.ylabel("ΔE")

plt.grid(True)

plt.show()