**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**МЕХАНИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ**

**Кафедра теоретической и прикладной механики**

**ПРИБЛИЖЕННАЯ МОДЕЛЬ ДВИЖЕНИЯ ФРОНТА ЖИДКОГО СЛОЯ НА ВРАЩАЮЩЕМСЯ ДИСКЕ**

Курсовая работа

Черепанова Кирилла Витальевича

студента 2 курса, специальность «механика и математическое моделирование»

Научный руководитель:

кандидат физико-математических наук,

доцент П.Н. Конон

Минск, 2022

# **Оглавление**

[Введение 3](#_Toc103184507)

[§1. Постановка задачи 3](#_Toc103184508)

[§2. Вывод уравнений движения 4](#_Toc103184509)

[§3. Разработка численного решения методом Рунге-Кутты 5](#_Toc103184510)

[§4. Графическая иллюстрация и анализ результатов 7](#_Toc103184512)

[Заключение 11](#_Toc103184513)

[Список литературы 12](#_Toc103184514)

# **Введение**

Целью данной курсовой работы является исследование движения жидкой капли на горизонтальном вращающемся диске.

В модели данного одним из основных моментов является определение силы вязкого трения. Предлагается приближенная модель его определения в случае медленного относительного движения, в модели исследуемого течения капля рассматривается как материальная точка.

Подобного рода течение встречается при срыве капли с горизонтального вращающегося распылителя в ряде технологических процессов, например, в сельском хозяйстве при распылении и обработки сельскохозяйственных культур пестицидами. Движение сплошного жидкого слоя на вращающейся горизонтальной поверхности используется в процессах нанесения слоев и смазок на поверхности дисков различных механизмов. Еще движение слоя на вращающемся диске используется в процессе образования минеральных волокон центробежно-дисковым способом.

# **§1.** **Постановка задачи и вывод уравнений движения**

# Построим приближенную модель движения капли. Будем исследовать движения капли с вязким трением как материальной частицы постоянной массы m по поверхности вращающегося с угловой скоростью ω горизонтального диска.

# В плоскости её движения введём абсолютную полярную систему координат *(r, φ)*. Пусть движение капли в абсолютной системе координат описывается законом:

*,* (1)

где *t* ­– время.

Кроме силы тяжести, направленной вертикально вниз, действует только одна сила – сила вязкого трения о поверхность вращающегося горизонтального основания. Введём следующие предположения относительно этой силы:

1. сила трения *F(t)* в каждый момент времени направлена противоположно вектору относительной скорости капли в этот момент времени, где

, (2)

где − орты системы *(r, φ);*

1. сила трения F в каждый момент времени зависит только от скорости *v* в данный момент времени, ;
2. при небольших значениях величины *v* относительной скорости зависимость модуля *F* силы трения от *v* близка к прямой пропорциональности, то есть:

; (3)

1. при больших значениях величины *v* относительной скорости зависимость модуля *F* силы трения от *v2* близка к прямой пропорциональности, то есть:

. (4)

# **§2.** **Вывод уравнений движения**

# Точка вверху, как обычно, обозначает производную по времени. Вектор абсолютного ускорения определяется по формуле [1]:

. (5)

# Таким образом, уравнения движения в неподвижной системе координат принимают вид:

(6)

В системе (6) величина скорости определяется формулой:

. (7)

Начальные условия для системы (6) имеют вид:

. (8)

# С помощью замены

. (9)

можно понизить порядок дифференциальных уравнений (6):

(10)

Начальные условия для системы (9), (10) имеют вид:

. (11)

Модуль относительной скорости находится по формуле

. (12)

Будем рассматривать случай медленного умеренного движения, при котором выполняется соотношение (3). Величину k1, которая необходима для практических расчётов, можно определить экспериментально. Также возможно использование альтернативного, теоретического способа определения данного коэффициента. Выделим из фронта движущейся вытянутой капли элемент приблизительно в виде прямоугольного параллелепипеда высоты *h*, длины *l* и ширины *d*, центр масс которого движется со скоростью *v*.

Скорость частиц жидкости рассматриваемого элемента на твердой поверхности в подвижной системе координат равна нулю, а на верхнем слое высотой *h* равна *v­max*. По гидромеханическому закону Ньютона, чтобы обеспечить такое движение, к верхнему слою необходимо приложить касательное напряжение *τ*, равное [11]:

, (13)

где *υ* − коэффициент кинематической вязкости жидкости, *ρ* − ее плотность.

Тогда величина сила трения единицы массы равна:

. (14)

C учетом (14) система уравнений движения с начальными условиями (10) примет вид:

(15)

# **§3. Разработка численного решения методом Рунге-Кутты**

Система дифференциальных уравнений (15) имеет нелинейный вид и вместе с начальными условиями (11) определяет задачу Коши. Для нахождения уравнений движения, воспользуемся методом Рунге-Кутта, и из полученных значений *(a, b)* выразим *(r, φ*). Расчеты проведены для движения я фронта слоев глицерина и воды, имеющих существенно различные вязкости. При температуре 20° С кинематическая вязкость глицерина принята равной , вязкость воды . Высота слоя *h* принималась равной 0.002 м. В результате решения определялись законы движения фронта слоя в абсолютной и относительной системах координат, а также траектории движения в этих двух системах координат.

Для решения системы напишем программу на языке Python. Язык прост в использовании и обладает большим количеством научных библиотек, из-за чего пользуется популярностью при решении подобного рода задач.

Для начала инициализируем переменные и зададим им значения:

h = 0.002 # Высота капли  
nyu = 1.006\*(10\*\*(-6)) # Вязкость воды  
w = math.pi # Угловая скорость вращения диска  
t0 = 0 # Начальный момент отсчёта времени  
tEnd = 10 # Конечный момент отсчёта времени  
step = 0.01 # Шаг

А также зададим начальные параметры :

r0 = 0.01; dr0 = 0  
phi0 = 0; dphi0 = 0   
initValues = [dr0/r0, dphi0] # Значения da/dt, db/dt при t = 0

Уравнение системы () запишем в виде функции, которая принимает значения *(a, b)*, и возвращает :

def f(t, values): # уравнения системы, где f[0] = da/dt, f[1] = db/dt  
 f = zeros([2]) # values = (a, b)  
 f[0] = values[1]\*\*2 - values[0]\*\*2 - values[0]\*2\*nyu/(h\*\*2)  
 f[1] = (w - values[1])\*2\*nyu/(h\*\*2) - 2\*values[1]\*values[0]  
 return f

Также заранее запишем функцию обратной подстановки, для нахождения *(r, φ*), зная *(a, b)*:

def abToRphi(a, b, t):  
 r = r0\*math.e\*\*(a \* t)  
 phi = b\*t + phi0  
 return [r, phi]

Затем напишем сам метод численного решения Рунге-Кутты 4-го порядка. Он будет принимать вид функции, принимающую систему уравнений в виде задачи Коши, ее начальные параметры, а также условия итерации, то есть конечный момент отсчета времени и его шаг, и будет возвращать массив решений системы уравнений:

def rungeKutta(f, t0, value, tEnd, step): # метод Рунге—Кутты  
 def increment(f, t, values):  
 k0 = f(t, values)  
 k1 = f(t + step / 2, values + k0 / 2)  
 k2 = f(t + step / 2, values + k1 / 2)  
 k3 = f(t + step, values + k2)  
 return (k0 + 2 \* k1 + 2 \* k2 + k3) / 6  
  
 t = [t0] # подготовка списка t  
 values = [value] # подготовка списка values  
 while t0 < tEnd: # внесение результатов расчёта в массивы t, values  
 step = min(step, tEnd - t0) # определение минимального шага  
  
 # расчёт в точке t0 значений initValues  
 value += step \* increment(f, t0, value)  
 t0 += step # приращение времени  
  
 t.append(t0) # заполнение массива t  
 values.append(deepcopy(value)) # заполнение массива values  
 return array(t), array(values)

Определив все необходимые переменные и функции, передаем в функцию метода Рунге-Кутты ее параметры и, получив значения *(a, b)*, с помощью ранее заданной функции **abToRphi(a, b, t)** найдем значения *(r, φ)*:

t, values = rungeKutta(f, t0, initValues, tEnd, step) # values - массив (a, b)  
finalValues = array([ abToRphi(\*values[i], t[i]) for i in range(int(values.size / 2)) ]) # finalValues - массив (r, φ)

Теперь можно изобразить решения дифференциальной системы уравнений на графике и проанализировать его.

# **§4. Графическая иллюстрация и анализ результатов**

Получив в §3 решение системы дифференциальных уравнений, мы можем построить график движения тела. Для этого Python предлагает библиотеку **matplotlib**, а именно его интерфейс **pyplot**. Здесь **matplotlib.pyplot** эскортирован в переменную **plt**.

Для начала разобьем массив значений *(r, φ)* на два массива со значениями *r* и *φ*:

r = reshape(array([i[0] for i in finalValues]), -1)  
phi = reshape(array([i[1] for i in finalValues]), -1)

Теперь построим график зависимости радиуса *r* от угла поворота *φ* в полярной системе координат и график зависимости радиуса *r* от времени *t* в прямоугольной:

# r от φ в полярной системе координат  
fig, ax = plt.subplots(subplot\_kw={'projection': 'polar'})  
ax.plot(phi, r)  
ax.set\_rmax(finalValues[-1][0])  
ax.grid(True)   
  
# r от t в прямоугольной системе координат  
plt.figure()  
plt.plot(t,r)  
plt.xlabel('Время, t')  
plt.ylabel('Радиус, r')  
plt.legend('π', loc='best')  
plt.grid(True)  
  
plt.show()

В результате получим график:



График 1. Траектория капли воды в относительной системе координат

Изменив в коде значение вязкости воды на вязкость глицерина:

nyu = 680\*(10\*\*(-6)) # Вязкость глицерина

получаем график движения капли глицерина:



График 2. Траектория капли глицерина в относительной системе координат

Движение капли глицерина на графике 2 объясняется довольно малым влиянием силы Кориолиса по сравнению с силой инерции из-за большого значения вязкости жидкости.

Далее изменяем значения *ω* угловых скоростях диска и заносим полученные значения на график, тем самым сравнивая их.

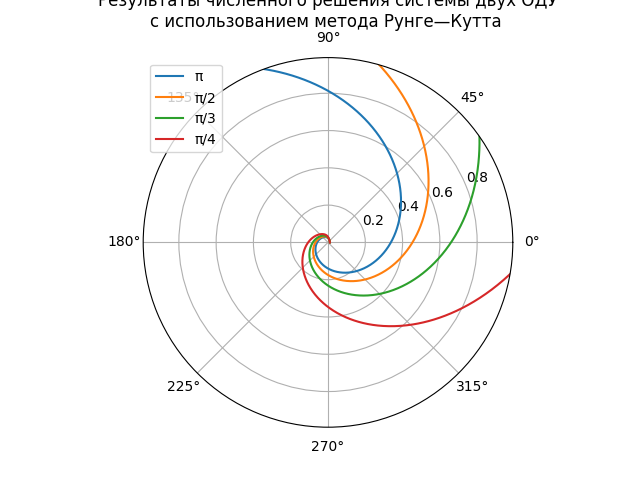


График 3. Траектории капли воды в относительной системе координат при разных *ω* угловых скоростях диска

На графике 3 наглядно видно каким образом изменяется движение капли при различных *ω* угловых скоростях диска. При увеличении скорости, увеличивается и относительная угловая скорость капли.

Также изобразим траекторию движения капли воды в абсолютной системе координат. Для этого необходимо провести такую замену:

. (16)



График 4. Траектория капли воды в абсолютной системе координат

Также необходимо показать зависимость радиуса от времени. Для этого потребуется изменить код вывода графиков:

plt.figure()  
plt.plot(t\_, r\_)  
plt.plot(t1, r1)  
plt.plot(t2, r2)  
plt.plot(t3, r3)  
plt.legend(['π/4', 'π/3', 'π/2', 'π'], loc='best')  
plt.xlabel('Время, t')  
plt.ylabel('Радиус, r')  
plt.grid(True)  
plt.show()

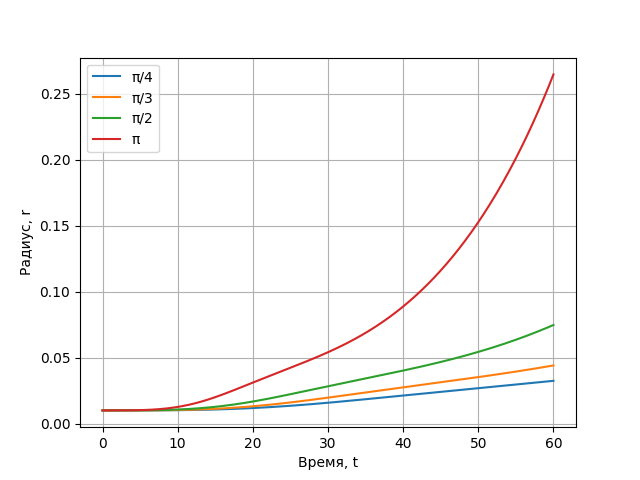


График 5. Закон движения r(t) фронта капель глицерина при различных угловых скоростях вращения диска

**Заключение**

В результате исследований построена приближенная модель движения фронт жидкого слоя на вращающемся диске, выведены уравнения движения капли как материальной точки с вязким трением. Предложен метод, позволяющий связать закон сопротивления капли при движении с коэффициентом кинематической вязкости жидкости. Проведено численное решение уравнений движения как задачи Коши, найдены абсолютные и относительные законы движения и траектории фронтов жидких слоев для различных жидкостей при разных угловых скоростях. Показано, что в случае ривулетного движения сильновязких жидкостей, как глицерин, при умеренных угловых скоростях относительные траектории близки к прямым линиям, а для жидкостей с небольшой вязкостью, как вода, они имеют вид хорошо развитых спиралей. Качественное сравнение результатов с экспериментами подтверждает возможность проведенного приближенного моделирования.

# **Список литературы**

1. Бухгольц Н.Н. Основной курс теоретической механики. Часть 1. М.: Наука, 1965. – 468 с Мещерский И.В. Сборник задач по теоретической механике, 1975
2. Шкадов В.Я., Запрянов З.Д. Течения вязкой жидкости. Издательство: М.: Изд-во МГУ, 200 с.