

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ  
Кафедра вычислительной математики

САЗОНОВ Игорь Евгеньевич

**ВЫЧИСЛИТЕЛЬНЫЙ АЛГОРИТМ РАСЧЁТА ФОРМЫ СВОБОДНОЙ  
ПОВЕРХНОСТИ ЖИДЕОСТИ В ЗАЗОРЕ МЕЖДУ  
КОАКСИАЛЬНЫМИ ЦИЛИНДРАМИ**

Дипломная работа

Научный руководитель:  
канд. физ.-мат. наук, доцент  
А.М. Будник

Допущен к защите

«\_\_\_» \_\_\_\_\_ 2025 г.  
Заведующий кафедрой  
вычислительной математики

---

канд. физ.-мат. наук, доцент В.И. Репников

Минск, 2025

## **РЕФЕРАТ**

Дипломная работа, 27 с., 6 источников, 3 рис., 1 прил.

**Ключевые слова:** ЖИДКОСТЬ, ГИДРОСТАТИКА, КОАКСИАЛЬНЫЕ ЦИЛИНДРЫ, МЕТОД ПРОГОНКИ.

**Объект исследования:** задача равновесия со свободной поверхностью.

**Цель работы:** исследовать формы равновесной поверхности жидкости в зазоре между коаксиальными цилиндрами в зависимости от начальных параметров, определяющих решение задачи.

**Задачи:** 1. Сформулировать математическую постановку задачи о форме свободной поверхности жидкости в зазоре между коаксиальными цилиндрами. 2. Разработать и реализовать на ЭВМ вычислительный алгоритм решения рассматриваемой задачи. 3. Провести вычислительный эксперимент по изучению влияния параметров, определяющих решение задачи, на конфигурацию свободной поверхности жидкости.

**Методы исследования:** методы вычислительной математики, вычислительный эксперимент и методы моделирования.

**Результаты работы:** 1. Сформулирована математическая постановка задачи о форме свободной поверхности жидкости, находящейся в зазоре между коаксиальными цилиндрами. 2. Разработан вычислительный алгоритм и соответственное программное обеспечение для определения равновесной формы. 3. Проведён вычислительный эксперимент по расчёту равновесных форм при различных значениях определённых параметров.

## РЭФЕРАТ

Дыпломная работа, 27 с., 6 крыніц, 3 мал., 1 дад.

**Ключавыя слова:** ВАДКАСЦЬ, ГІДРАСТАТЫКА, КААКСІЯЛЬНЫЯ ЦЫЛІНДРЫ, МЕТАД ПРАГОНКІ.

**Аб'ект даследавання:** задача раўнавагі са свабоднай паверхнай.

**Мэта работы:** даследаваць формы раўнаважнай паверхні вадкасці ў прамежку паміж кааксіяльнымі цыліндрамі ў залежнасці ад пачатковых параметраў, што вызначаюць рашэнне задачы.

**Задачы:** 1. Сфармуляваць матэматаічную пастанову задачы аб форме свабоднай паверхні вадкасці ў прамежку паміж кааксіяльнымі цыліндрамі. 2. Распрацаваць і рэалізаваць на ЭВМ вылічальны алгарытм рашэння разгляданай задачы. 3. Правесці вылічальны эксперымент па вывучэнні ўплыву параметраў, што вызначаюць рашэнне задачы, на канфігурацыю свабоднай паверхні вадкасці.

**Метады даследавання:** метады вылічальнай матэматаікі, вылічальны эксперымент і метады мадэлявання.

**Атрыманые вынікі:** 1. Сфармульвана матэматаічная пастанову задачы аб форме свабоднай паверхні вадкасці, што знаходзіцца ў прамежку паміж кааксіяльнымі цыліндрамі. 2. Распрацаваны вылічальны алгарытм і адпаведнае праграмнае забеспячэнне для вызначэння раўнаважнай формы. 3. Праведзены вылічальны эксперымент па разліку раўнаважных форм пры розных значэннях пэўных параметраў.

## SUMMARY

Degree paper, 27 p., 6 sources, 3 fig., 1 application.

**Keywords:** FLUID, HYDROSTATICS, COAXIAL CYLINDER, THOMAS ALGORITHM.

**The object of the research:** the equilibrium problem with a free surface.

**The purpose of the research:** to study the equilibrium surface shapes of a liquid in the gap between coaxial cylinders, depending on the initial parameters that define the problem's solution.

**Tasks:** 1. Formulate the mathematical problem statement regarding the shape of the free surface of a liquid in the gap between coaxial cylinders. 2. Develop and implement a computational algorithm on a computer to solve this problem. 3. Conduct a computational experiment to study the influence of parameters defining the problem's solution on the configuration of the free liquid surface.

**Methods of research:** computational mathematics methods, computational experiments, and modeling techniques.

**The results of the work:** 1. The mathematical formulation of the problem concerning the shape of the free liquid surface in the gap between coaxial cylinders has been established. 2. A computational algorithm and corresponding software for determining the equilibrium shape have been developed. 3. A computational experiment was conducted to calculate equilibrium shapes at various values of key parameters.

# ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	6
ГЛАВА 1 Условия равновесия .....	7
1.1 Основные предположения и обозначения.....	7
1.2 Условия гидростатики. Равновесие свободной поверхности....	8
ГЛАВА 2 Формулировка задачи о равновесной поверхности .....	9
2.1 Параметрическое описание формы свободной поверхности ....	9
2.2 Построение математической модели .....	9
2.3 Обезразмеривание математической модели .....	11
ГЛАВА 3 Построение разностной схемы.....	13
3.1 Основные обозначения.....	13
3.2 Аппроксимация задачи.....	14
3.3 Организация итерационного процесса .....	16
ГЛАВА 4 Решение задачи.....	19
4.1 Метод прогонки.....	19
4.2 Вычислительный эксперимент .....	22
ЗАКЛЮЧЕНИЕ .....	24
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	25
ПРИЛОЖЕНИЕ .....	26

## **ВВЕДЕНИЕ**

Коаксиальные оболочки или цилиндры, содержащие жидкость, широко используются в качестве конструкционных компонентов в различных областях применения, начиная от нефтегазовой промышленности и заканчивая космическими исследованиями. Задача играет немаловажную роль при моделировании поведения жидкостей в трубопроводах, центрифугах и других устройствах, где жидкость со свободной поверхностью может находиться между двумя цилиндрами.

Проблема моделирования таких форм требует глубокого анализа и точного математического описания. Задача представляет собой систему нелинейных дифференциальных уравнений, которые практически нельзя решить аналитически. Для ее решения требуется специальные численные методы, использующие итерационные процессы.

В данной работе будет представлен подход к решению данной задачи, основанный на итерационно-разностном методе.

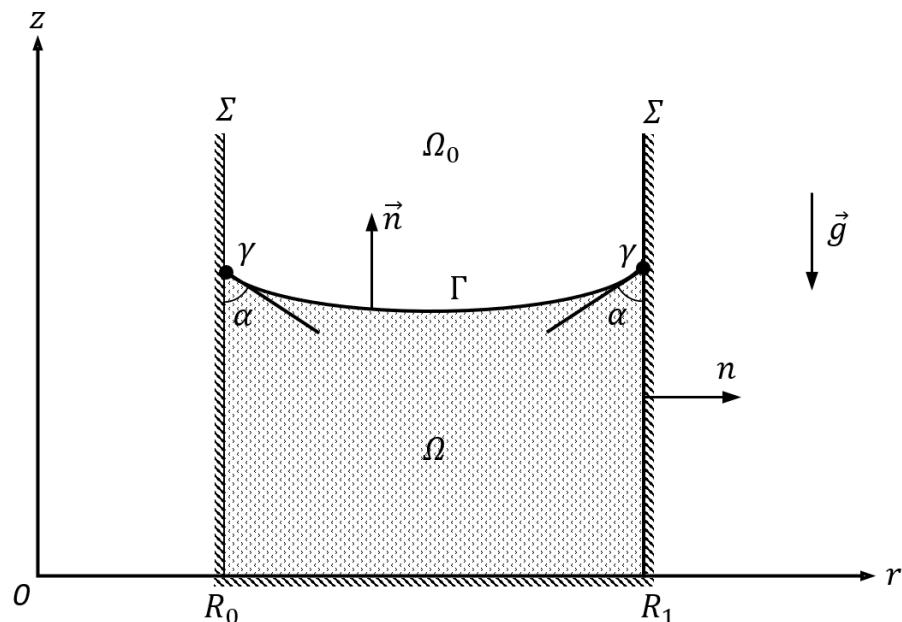
# ГЛАВА 1

## Условия равновесия

### 1.1 Основные предположения и обозначения

Введём цилиндрическую систему координат  $Or\theta z$ , которая будет использоваться для представления поперечного сечения цилиндров. В качестве параметра, характеризующего форму свободной поверхности будем использовать длину дуги  $s$ .

Пусть однородная, несжимаемая жидкость объёма  $V$  занимает пространство между двумя коаксиальными цилиндрами радиусов  $R_0$  и  $R_1$  и горизонтальной поверхностью. При этом поверхности цилиндров недеформируемые и гладкие. Жидкость, плотность которой будем считать равной  $\rho$ , и газ, плотностью которого пренебрегаем, занимают области  $\Omega$  и  $\Omega_0$  соответственно. Пусть  $\Gamma$  – есть поверхность контакта жидкости и газа, а  $\gamma$  – линии контакта жидкости, газа и поверхности цилиндров  $\Sigma$ . Обозначим через  $\alpha$  двугранный угол смачивания, образуемый жидкостью со стенкой сосуда в точках линии  $\gamma$ . А также введём единичные векторы  $\vec{n}$  и  $\vec{n}_\Sigma$  нормали к  $\Gamma$  и  $\Sigma$ . Принимая во внимание осевую симметрию задачи, введём в рассмотрение цилиндрическую систему координат и будем рассматривать задачу на плоскости  $(r, z)$ ,  $0 \leq r \leq \infty$ ,  $-\infty < z < \infty$ . Ось  $z$  совместим с центром внутреннего цилиндра, при этом ось  $r$  будем рассматривать как твердую поверхность (см. рис. 1).



**Рисунок 1 – Геометрия задачи. Основные обозначения параметров жидкости и системы координат  $Or\theta z$**

## 1.2 Условия гидростатики. Равновесие свободной поверхности

Равновесную форму свободной поверхности жидкости определим исходя из условия Юнга-Лапласа, которое описывает разность давлений в жидкости и газе на границе раздела, возникающую из-за капиллярных эффектов. Это условие учитывает влияние кривизны поверхности на давление в системе:

$$\sigma(k_1 + k_2) = p_0 - p, \quad (1.1)$$

где  $\sigma = \text{const}$  – коэффициент поверхностного натяжения,  $k_1$  и  $k_2$  – кривизны главных нормальных сечений поверхности  $\Gamma$ ,  $p_0 = \text{const}$  – давление в газе,  $p$  – давление в жидкости.

Согласно принципу Эйлера, давление жидкости при условии равновесия зависит от высоты её столба и плотности, но не зависит от формы сосуда, поэтому при условии гравитации получим:

$$\nabla p = \rho F, \quad (1.2)$$

где  $\nabla p$  – градиент давления,  $\rho F$  – объёмная плотность массовых сил. Потенциал поля массовых сил  $\vec{F} = -\nabla \Pi$  обеспечивает возможность существования равновесного состояния в среде.

Проинтегрировав уравнение (1.2) по всем направлениям в пространстве, получим:

$$p = C. \quad (1.3)$$

Принимая во внимание уравнение (1.3), условие (1.1) перепишем в виде:

$$\sigma(k_1 + k_2) = -\rho \Pi + c, \quad (1.4)$$

где  $c$  – произвольная константа,  $\Pi$  – потенциал поля массовых сил.

Немаловажным также является условие Дюпре-Юнга. Оно играет роль краевого условия, которое вычисляется на линии  $\gamma$ :

$$\cos \alpha = -\vec{n} \cdot \vec{n}_\Sigma. \quad (1.5)$$

Для достижения равновесия в жидкости, необходимо удовлетворение условия (1.4) на её свободной поверхности  $\Gamma$  и условия (1.5) в точке контакта  $\gamma$ . Условие (1.4) определяет дифференциальное уравнение для поверхности, в то время как (1.5) служит соответствующим граничным условием. Чтобы определить константу  $c$  в уравнении (1.4), задача обычно включает дополнительное условие, основанное на известном объеме  $V$ , который может быть вычислен по формуле:

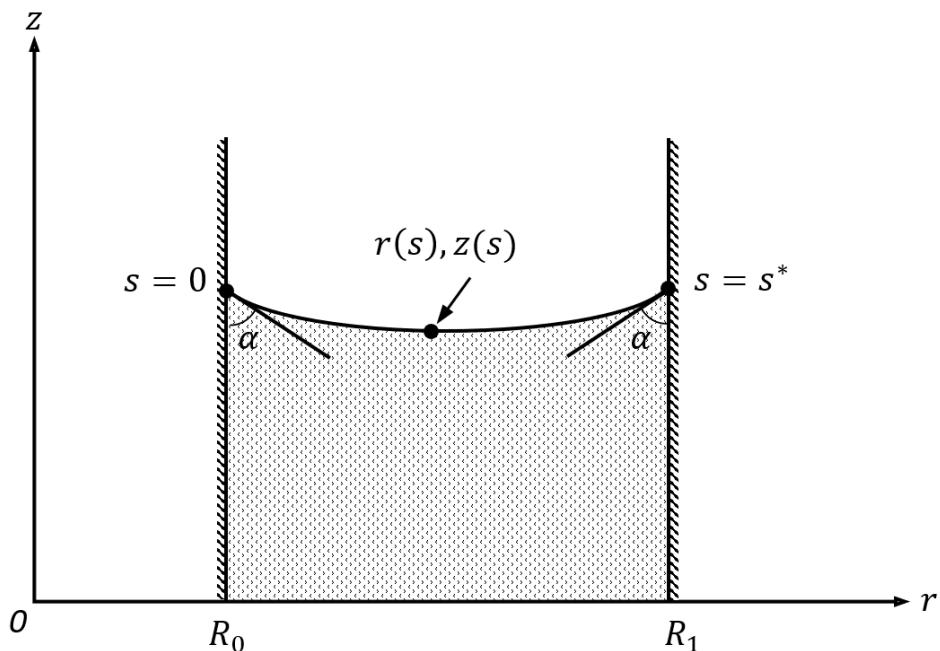
$$V = \int_\Omega d\Omega. \quad (1.6)$$

## ГЛАВА 2

### Формулировка задачи о равновесной поверхности

#### 2.1 Параметрическое описание формы свободной поверхности

С учётом вышеизложенного, будем рассматривать задачу только в первой четверти координатной системы. Форму свободной поверхности опишем в параметрическом виде  $z(s)$ ,  $r(s)$ , где  $s$  – длина дуги равновесной линии, отсчитываемая от точки контакта с внутренним цилиндром  $s = 0$  до точки касания внешнего цилиндра  $s = s^*$  (см. рис. 2).



**Рисунок 2 – Представление формы свободной поверхности в первой четверти координатной системы**

#### 2.2 Построение математической модели

Средняя кривизна в цилиндрической системе координат осесимметричной поверхности определяется выражением:

$$k_1 + k_2 = \frac{1}{rr'}(rz')', \quad (2.1)$$

где  $' = \frac{d}{ds}$ .

Так как потенциал поля массовых сил при наличии силы тяжести, направленной противоположно оси  $z$ , вычисляется, как  $\Pi(r, z) = gz + const$ , то из уравнений (1.4) и (2.1) можно получить:

$$\frac{1}{rr'}(rz')' = \frac{\rho g}{\sigma}z + c. \quad (2.2)$$

Откуда выражая  $z''$  получим первое дифференциальное уравнение:

$$z'' = r' \left( \frac{\rho g}{\sigma} z + c - \frac{z'}{r} \right). \quad (2.3)$$

Так как в параметрической постановке искомым решением задачи являются две функции, то для математической формулировки необходимо кроме условия (2.3) задать ещё одно уравнение. В качестве такого уравнения будем рассматривать условие нормировки параметрически заданной кривой:

$$(z')^2 + (r')^2 = 1. \quad (2.4)$$

Продифференцировав уравнение (2.4) по  $s$  и воспользовавшись уравнением (2.3), получим второе дифференциальное уравнение относительно переменной  $r''$ :

$$r'' = -z' \left( \frac{\rho g}{\sigma} z + c - \frac{z'}{r} \right). \quad (2.5)$$

Таким образом уравнения (2.3), (2.5) задают двухпараметрическое семейство интегральных линий в плоскости  $r, z$ . Помимо этого, они дополняются граничными условиями, которые могут быть получены из условий (1.5) и (2.4), а именно:

$$\begin{aligned} r(0) &= R_0, r(s^*) = R_1, \\ z'(0) &= -\cos \alpha, r'(0) = \sin \alpha, \\ z'(s^*) &= \cos \alpha, r'(s^*) = \sin \alpha. \end{aligned} \quad (2.6)$$

Формулу нахождения объёма можно получить исходя из метода расчёта объёма посредством элементарных колец, когда объём тела вращения равен значению интеграла вида:

$$V = 2\pi \int_0^{s^*} z(r) r dr = 2\pi \int_0^{s^*} z r r' ds. \quad (2.7)$$

Кроме того, для вычисления объёма жидкости можно использовать и такую формулу:

$$V = 2\pi \left( R_1^2 \cdot z(s^*) - R_0^2 \cdot z(0) - \int_0^{s^*} z' r^2 ds \right). \quad (2.8)$$

Таким образом, уравнения (2.3), (2.5) и условия (2.6), (2.8) образуют размерную математическую модель задачи равновесной формы свободной поверхности жидкости.

## 2.3 Обезразмеривание математической модели

Выражения (2.3) и (2.5) содержат величину  $\frac{\rho g}{\sigma}$ , которая обладает размерностью  $(\text{длина})^{-2}$ . При определении характерного для данной системы размера  $l$ , в этих уравнениях появляется безразмерный параметр  $Bo = \frac{\rho g}{\sigma} l^2$ , называемый числом Бонда. Этот параметр определяет соотношение между гравитационными и капиллярными силами. Изменение числа Бонда можно интерпретировать как изменение одной из физических величин, входящих в это число, при фиксированных остальных величинах. Таким образом, появляется возможность исследовать влияние этих величин на форму свободной поверхности жидкости.

При геометрически подобном преобразовании равновесной системы результатом будет также равновесная система. Следовательно, решение, найденное для какого-либо значения числа Бонда, можно пропорционально изобразить и для системы с размерностью.

Проведем обезразмеривание уравнений (2.3), (2.5), (2.6), (2.8), взяв в качестве характерного размера величину расстояния  $R_0$ . Обезразмеривание помогает понять, какие параметры влияют на решение рассматриваемой задачи. Вместо того чтобы работать с конкретными физическими величинами, мы переходим к безразмерным параметрам, которые показывают соотношение различных факторов. Это упрощает анализ и позволяет применять результаты к разным масштабам. Пусть в безразмерных переменных:

$$\delta = \frac{R_1}{R_0}, \quad \bar{r} = \frac{r}{R_0}, \quad \bar{z} = \frac{z}{R_0}, \quad \bar{s} = \frac{s}{R_0}, \quad s_1 = \frac{s^*}{R_0}, \quad Bo = \frac{\rho g}{\sigma} R_0^2, \quad \bar{q} = c \cdot R_0,$$

тогда уравнения (2.3), (2.5) перепишутся в виде:

$$\bar{z}'' = \bar{r}' \left( Bo \cdot \bar{z} + \bar{q} - \frac{\bar{z}'}{\bar{r}} \right), \quad \bar{r}'' = -\bar{z}' \left( Bo \cdot \bar{z} + \bar{q} - \frac{\bar{z}'}{\bar{r}} \right), \quad \text{где } ' = \frac{d}{ds}. \quad (2.9)$$

Перепишем также граничные условия (2.6) в безразмерном виде, для которых  $0 \leq s \leq s_1$ :

$$\begin{aligned} \bar{r}(0) &= 1, \quad \bar{r}(s_1) = \delta, \\ \bar{z}'(0) &= -\cos \alpha, \quad \bar{r}'(0) = \sin \alpha, \\ \bar{z}'(s_1) &= \cos \alpha, \quad \bar{r}'(s_1) = \sin \alpha. \end{aligned} \quad (2.10)$$

И, наконец, условие на объём (2.8):

$$U = \pi \left( \bar{z}(s_1) \delta^2 - \bar{z}(0) - \int_0^{s_1} \bar{z}' \bar{r}^2 ds \right), \quad U = \frac{V}{R_0^3}. \quad (2.11)$$

Далее везде для простоты записи будем опускать чёрточки. Выразив из условия на объём (2.11) величину  $\bar{z}(s_1)$  получим:

$$z(s_1) = \frac{1}{\delta^2} \left( \frac{U}{\pi} + z(0) + \int_0^{s_1} z' r^2 ds \right). \quad (2.12)$$

Равенство (2.12) будет использовано в дальнейшем для численного решения задачи. Значение  $s_1$ , с учётом выражения (2.4), найдём как длину дуги:

$$s_1 = \int_0^{s_1} [(z')^2 + (r')^2] ds. \quad (2.13)$$

Теперь определим константу  $q$ . Для этого воспользуемся условием (2.2). Затем проинтегрируем по  $s$  на интервале от 1 до  $\delta$ , используя условие объёма (2.7) и краевые условия (2.10) :

$$\begin{aligned} \frac{1}{rr'} (rz')' &= Bo \cdot z + q \\ (rz')' &= rr'(Bo \cdot z + q) \\ \int_1^\delta (rz')' ds &= \int_1^\delta rr'(Bo \cdot z + q) ds \\ (rz')|_1^\delta &= \int_1^\delta q \cdot \frac{1}{2} (r^2)' ds + \int_1^\delta Bo \cdot z r r' ds \\ (1 + \delta) \cos \alpha &= \frac{q}{2} (\delta^2 - 1) + Bo \int_1^\delta z r r' ds \\ q &= \frac{2}{(\delta^2 - 1)} \left( (1 + \delta) \cos \alpha - Bo \frac{U}{2\pi} \right). \end{aligned}$$

В итоге получим формулу для определения константы:

$$q = \frac{2}{\delta^2 - 1} \left( (1 + \delta) \cos \alpha - Bo \frac{U}{2\pi} \right). \quad (2.14)$$

Таким образом, уравнения (2.9), (2.10), (2.12) – (2.14) образуют безразмерную математическую модель задачи равновесной формы свободной поверхности жидкости. При этом решение задачи  $(z(s), r(s))$  будет зависеть от следующих параметров:  $Bo, U, \alpha, \delta$ .

## ГЛАВА 3

### Построение разностной схемы

#### 3.1 Основные обозначения

Для получения приближенного решения дифференциального уравнения с использованием метода конечных разностей, необходимо:

1. Перейти от непрерывной области определения аргумента к дискретной области определения.
2. Заменить дифференциальную задачу на её разностный аналог и построить разностную схему.
3. Разработать вычислительный алгоритм, реализующий полученную разностную схему.

После выполнения указанных действий первоначальная задача сводится к решению системы алгебраических уравнений. Рассмотрим этот процесс подробнее. При численном решении математических задач невозможно получить разностное решение для всех значений аргумента в заданной области евклидова пространства. Поэтому необходимо выбрать конечное множество точек внутри этой области и искать приближенное решение только в этих точках. Такое множество называется вычислительной сеткой, а отдельные его точки — узлами. Функция, заданная в узлах сетки, называется сеточной функцией.

Сетка представляет собой разбиение непрерывной области на конечное число узлов, что позволяет перейти от анализа функций в непрерывном пространстве к работе с дискретными значениями. Сетка определяется шагом  $h$ , который равен расстоянию между соседними узлами:  $h = \frac{b-a}{N}$ , где  $N$  — количество узлов на интервале  $[a, b]$  в котором исследуется функция. Дискретные точки на сетке обозначаются переменной  $x_i$ , где  $i$  — индекс узла. Например, если в задаче рассматривается одно измерение, то узлы могут быть равномерно распределены по интервалу, и тогда  $x_i$  могут принимать значения  $x_0, x_1, x_2, \dots, x_N$ . Для равномерной сетки  $h = x_{i+1} - x_i$ .

Искомая функция  $u$ , определяемая в узлах сетки, представляет собой решение задачи, связанной с нелинейными дифференциальными уравнениями в частных производных. Эта функция может описывать различные физические или математические процессы, такие как распределение температуры, концентрацию вещества или механическое напряжение.

Для замены дифференциальных производных могут использоваться следующие разностные операторы:

Прямой разностный оператор

$$u'(x_i) = \frac{u(x_{i+1}) - u(x_i)}{h}. \quad (3.1)$$

Обратный разностный оператор

$$u'(x_i) = \frac{u(x_i) - u(x_{i-1})}{h}. \quad (3.2)$$

Центральный разностный оператор для первой производной

$$u'(x_i) = \frac{u(x_{i+1}) - u(x_{i-1})}{2h}. \quad (3.3)$$

Центральный разностный оператор для второй производной

$$u''(x_i) = \frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1})}{h^2}. \quad (3.4)$$

При этом погрешность аппроксимации выражений (3.1), (3.2) имеет первый порядок  $O(h)$ , тогда как выражения (3.3) и (3.4) обладают аппроксимацией второго порядка  $O(h^2)$ .

### 3.2 Аппроксимация задачи

Для аппроксимации системы (2.9) – (2.14) со вторым порядком в области определения функций  $z(s)$ ,  $r(s)$ , введём следующую равномерную сетку:  $\omega_h = \{s_k = kh, k = \overline{0, N}, h = \frac{s_N - s_0}{N}\}$ , где  $N$  — количество разбиений отрезка  $[s_0, s_N]$ . Таким образом, точки сетки  $s_k$  будут равномерно распределены по интервалу. Далее запишем разностные уравнения для сеточных функций  $z(s_k) = z_k$ ,  $r(s_k) = r_k$ , где  $k = \overline{0, N}$ .

Тогда, применяя формулы (3.1) – (3.4) для уравнений (2.9) получим:

$$\frac{1}{h^2} (z_{k-1} - 2z_k + z_{k+1}) = \frac{r_{k+1} - r_{k-1}}{2h} \left( Bo \cdot z_k + Q - \frac{z_{k+1} - z_{k-1}}{2hr_k} \right), \quad (3.5)$$

$$\frac{1}{h^2} (r_{k-1} - 2r_k + r_{k+1}) = \frac{z_{k+1} - z_{k-1}}{2h} \left( Bo \cdot z_k + Q - \frac{z_{k+1} - z_{k-1}}{2hr_k} \right), \quad (3.6)$$

Начальные условия  $r(0) = \delta$ ,  $r(s) = 1$  аппроксимируем как

$$r_0 = \delta, r_N = 1. \quad (3.7)$$

Чтобы аппроксимировать оставшиеся выражения из (2.8), применим разложение граничных условий в ряд Тейлора:

$$\begin{aligned} z'(0) &= \frac{z_1 - z_0}{h} - \frac{h}{2} z''(0) + O(h^2) = \frac{z_1}{h} - \frac{z_0}{h} - \frac{h}{2} r'(0) \left( Bo \cdot z_0 + Q - \frac{z'(0)}{r(0)} \right) \\ -\cos \alpha &= \frac{z_1}{h} - \frac{z_0}{h} - \frac{h}{2} \sin \alpha \left( Bo \cdot z_0 + Q - \frac{-\cos \alpha}{1} \right) \\ z_1 &= z_0 - h \cos \alpha + \frac{h^2}{2} \sin \alpha (Bo \cdot z_0 + Q + \cos \alpha). \end{aligned} \quad (3.8)$$

$$\begin{aligned} z'(s) &= \frac{z_N - z_{N-1}}{h} - \frac{h}{2} z''(s) + \dots = \frac{z_N}{h} - \frac{z_{N-1}}{h} - \frac{h}{2} r'(s) \left( Bo \cdot z_N + Q - \frac{z'(s)}{r(s)} \right) \\ \cos \alpha &= \frac{z_N}{h} - \frac{z_{N-1}}{h} - \frac{h}{2} \sin \alpha \left( Bo \cdot z_N + Q - \frac{\cos \alpha}{\sigma} \right) \\ z_{N-1} &= z_N - h \cos \alpha - \frac{h^2}{2} \sin \alpha \left( Bo \cdot z_N + Q - \frac{\cos \alpha}{\sigma} \right). \end{aligned} \quad (3.9)$$

$$\begin{aligned} r'(0) &= \frac{r_1 - r_0}{h} - \frac{h}{2} r''(0) + O(h^2) = \frac{r_1}{h} - \frac{r_0}{h} - \frac{h}{2} z'(0) \left( Bo \cdot z_0 + Q - \frac{z'(0)}{r(0)} \right) \\ \sin \alpha &= \frac{r_1}{h} - \frac{r_0}{h} + \frac{h}{2} (-\cos \alpha) \left( Bo \cdot z_0 + Q - \frac{-\cos \alpha}{1} \right) \\ r_1 &= r_0 + h \sin \alpha - \frac{h^2}{2} \cos \alpha (Bo \cdot z_0 + Q + \cos \alpha). \end{aligned} \quad (3.10)$$

$$\begin{aligned} r'(s) &= \frac{r_N - r_{N-1}}{h} - \frac{h}{2} r''(s) + \dots = \frac{r_N}{h} - \frac{r_{N-1}}{h} - \frac{h}{2} z'(s) \left( Bo \cdot z_N + Q - \frac{z'(s)}{r(s)} \right) \\ \sin \alpha &= \frac{r_N}{h} - \frac{r_{N-1}}{h} - \frac{h}{2} \cos \alpha \left( Bo \cdot z_N + Q - \frac{\cos \alpha}{\sigma} \right) \\ r_{N-1} &= r_N - h \sin \alpha - \frac{h^2}{2} \cos \alpha \left( Bo \cdot z_N + Q - \frac{\cos \alpha}{\sigma} \right). \end{aligned} \quad (3.11)$$

Аналогично проведём аппроксимацию условия на объём жидкости, где вычисление интеграла найдём приближённо по составной квадратурной формуле трапеций для достижения необходимой точности, т. е.

$$\begin{aligned} U &= 2\pi \left( z(s_1) \delta^2 - z(0) - \int_0^{s_1} z' r^2 ds \right) \\ U &= 2\pi \left( z_N \delta^2 - z_0 - h \left( \frac{z'(0)r^2(0) + z'(s)r^2(s)}{2} + \sum_{k=1}^{N-1} z' r^2 \right) \right) \end{aligned}$$

$$U = 2\pi \left( z_N \delta^2 - z_0 - h \left( \frac{-\cos \alpha \cdot 1 + \cos \alpha \cdot \delta^2}{2} + \sum_{k=1}^{N-1} \frac{z_{k+1} - z_{k-1}}{2h} r_k^2 \right) \right). \quad (3.12)$$

Также аппроксимируем граничное условие (2.12):

$$\begin{aligned} z(s_1) &= \frac{1}{\delta^2} \left( \frac{U}{\pi} + z(0) + \int_0^{s_1} z' r^2 ds \right) \\ z_N &= \frac{1}{\delta^2} \left( \frac{U}{\pi} + z_0 + \sum_{k=1}^{N-1} \frac{z_{k+1} - z_{k-1}}{2h} r_k^2 \right). \end{aligned} \quad (3.13)$$

И условие геометрической параметризации (2.13):

$$\begin{aligned} s_1 &= \int_0^{s_1} [(z')^2 + (r')^2] ds \\ s_1 &= h \left( \frac{(z'(0))^2 + (r'(0))^2}{2} + \frac{(z'(s_1))^2 + (r'(s_1))^2}{2} + \right. \\ &\quad \left. + \sum_{k=1}^{N-1} \left( \frac{z_{k+1} - z_{k-1}}{2h} \right)^2 + \left( \frac{r_{k+1} - r_{k-1}}{2h} \right)^2 \right) \\ s_1 &= h \left( 1 + \sum_{k=1}^{N-1} \left( \frac{z_{k+1} - z_{k-1}}{2h} \right)^2 + \left( \frac{r_{k+1} - r_{k-1}}{2h} \right)^2 \right). \end{aligned} \quad (3.14)$$

Учитывая также константу:

$$Q = \frac{2}{\delta^2 - 1} \left( (1 + \delta) \cos \alpha - Bo \frac{U}{2\pi} \right). \quad (3.15)$$

Таким образом выражения (3.5) – (3.15) задают систему разностных уравнений, которая позволяет численно решать исходные уравнения (2.9) – (2.14) в заданной области.

### 3.3 Организация итерационного процесса

Поскольку получившаяся задача (3.5) – (3.15) является нелинейной, её приближенное решение требует применения итерационного подхода для нахождения приближенного решения. Итерационно-разностный метод представляет собой обширный класс численных методов, широко используемых для решения задач, связанных с дифференциальными уравнениями в частных производных. Такой метод основывается на использовании разностных аппроксимаций, которые заменяют дифференциальные производные в уравнении конечными разностями. Это даёт возможность преобразовать сколь угодно малые изменения в конечные

измеримые величины, что делает возможным использование компьютеров для расчетов.

Поиск решения для данной задачи – это многошаговый процесс, требующий нескольких итераций, так как она обладает нелинейным характером и не может быть выражена в виде точной аналитической формулы. На первом этапе определяется начальное приближение, опирающееся на аналитическое решение или начальные условия. Это соответствует нулевой итерации ( $m = 0$ ), когда расчёты выполняются без внесения поправок. Далее начинается итерационный процесс ( $m = 1, 2, \dots$ ), в ходе которого значения корректируются с учётом результатов, полученных на предыдущих этапах расчётов. С каждой новой итерацией  $m$  происходит уточнение параметров модели, что обеспечивает постепенное приближение к искомому точному решению.

Поскольку шаг  $h$  определяется как  $h = \frac{s_1}{N}$ , а значение  $s_1$  может изменяться на каждой итерации  $m$ , необходимо обновлять значение  $h^{(m)}$  в системе уравнений, чтобы учесть изменяющуюся геометрию поверхности  $h^{(m)} = \frac{s_1^{(m)}}{N}$ .

Теперь можем выполнить вычисления на  $(m+1)$ -й итерации для всех  $z_k^{(m+1)}$ , где в данном случае для перехода со слоя на слой, т. е. с одной итерации на другую, приходится решать систему уравнений (3.5), (3.8), (3.9), (3.13):

$$\left\{ \begin{array}{l} \frac{1}{h^{(m)2}} \left( z_{k-1}^{(m+1)} - 2z_k^{(m+1)} + z_{k+1}^{(m+1)} \right) = \frac{r_{k+1}^{(m)} - r_{k-1}^{(m)}}{2h^{(m)}} \left( Bo \cdot z_k^{(m)} + Q^{(m)} - \frac{z_{k+1}^{(m)} - z_{k-1}^{(m)}}{2h^{(m)}r_k^{(m)}} \right), \\ z_1^{(m+1)} = z_0^{(m+1)} - h^{(m)} \cos \alpha + \frac{h^{(m)2}}{2} \sin \alpha \left( Bo \cdot z_0^{(m)} + Q^{(m)} + \cos \alpha \right), \\ z_{N-1}^{(m+1)} = z_N^{(m+1)} - h^{(m)} \cos \alpha - \frac{h^{(m)2}}{2} \sin \alpha \left( Bo \cdot z_N^{(m)} + Q^{(m)} - \frac{\cos \alpha}{\delta} \right). \end{array} \right. \quad (3.16)$$

В результате чего найдём значения  $z_k^{(m+1)}$  для всех целых  $k \geq 0$ . И далее, для обеспечения устойчивости итераций, а также ускорения сходимости решения, воспользуемся формулой релаксации, по которой полученные значения  $z_k^{(m+1)}$  пересчитаем как:

$$z_k^{(m+1)} = \tau z_k^{(m+1)} + (1 - \tau) z_k^{(m)}. \quad (3.17)$$

Параметр релаксации  $\tau$  (где  $0 < \tau \leq 1$ ) необходим для контроля вклада текущего и предыдущего значений в обновление итераций.

Аналогичные действия проведём относительно параметра  $r_k^{(m+1)}$  для системы уравнений (3.6), (3.10), (3.11), (3.7):

$$\begin{cases} \frac{1}{h^{(m)2}} \left( r_{k-1}^{(m+1)} - 2r_k^{(m+1)} + r_{k+1}^{(m+1)} \right) = \frac{z_{k+1}^{(m)} - z_{k-1}^{(m)}}{2h^{(m)}} \left( Bo \cdot z_k^{(m)} + Q^{(m)} - \frac{z_{k+1}^{(m)} - z_{k-1}^{(m)}}{2h^{(m)}r_k^{(m)}} \right), \\ r_1^{(m+1)} = r_0^{(m+1)} + h^{(m)} \sin \alpha - \frac{h^{(m)2}}{2} \cos \alpha \left( Bo \cdot z_0^{(m)} + Q^{(m)} + \cos \alpha \right), \\ r_{N-1}^{(m+1)} = r_N^{(m+1)} - h^{(m)} \sin \alpha - \frac{h^{(m)2}}{2} \cos \alpha \left( Bo \cdot z_N^{(m)} + Q^{(m)} - \frac{\cos \alpha}{\delta} \right), \\ r_0^{(m+1)} = \sigma, r_N^{(m+1)} = 1. \end{cases} \quad (3.18)$$

для которой найдём значения  $r_k^{(m+1)}$  для всех  $k \geq 0$  с их дальнейшим пересчётом

$$r_k^{(m+1)} = \tau r_k^{(m+1)} + (1 - \tau) r_k^{(m)}. \quad (3.19)$$

На каждой итерации необходимо уточнять значения  $Q$  и  $U$ , для которых

$$\begin{aligned} U^{(m+1)} = \pi \left( z_N^{(m+1)} \delta^2 - z_0^{(m+1)} - \frac{h^{(m+1)} \cos \alpha}{2} (\delta^2 - 1) + \right. \\ \left. + \sum_{k=1}^{N-1} \frac{z_{k+1}^{(m+1)} - z_{k+1}^{(m+1)}}{2} r_k^{2(m+1)} \right) \end{aligned} \quad (3.20)$$

$$Q^{(m+1)} = \frac{2}{\delta^2 - 1} \left( (1 + \delta) \cos \alpha - Bo \cdot \frac{U^{(m+1)}}{2\pi} \right). \quad (3.21)$$

Итерации продолжаются до тех пор, пока не будет достигнут критерий остановки с точностью  $\varepsilon$ :

$$\left| z_k^{(m+1)} - z_k^{(m)} \right| \leq \tau \varepsilon \text{ и } \left| r_k^{(m+1)} - r_k^{(m)} \right| \leq \tau \varepsilon \text{ для любого } 0 \leq k \leq N.$$

Этот критерий подразумевает, что процесс итерации продолжается, пока разница между итерациями для каждой точки сетки не станет достаточно малой, что, собственно, и свидетельствует о достижении необходимой точности  $\varepsilon$ .

## ГЛАВА 4

### Решение задачи

#### 4.1 Метод прогонки

Ключевым аспектом итерационно-разностного метода является выбор схемы итераций. Для решения нелинейных систем уравнений применяются различные методы, включая метод последовательных приближений, метод Гаусса-Зейделя, метод релаксации и другие. Эффективность каждого из них определяется спецификой решаемой задачи и ее свойствами. Одним из наиболее употребительных способов решения разностных уравнений, возникающих при аппроксимации краевых задач для уравнений математической физики, является метод прогонки.

Метод прогонки представляет собой алгоритм для решения систем линейных уравнений определённого вида. А именно, его применяют для трёхдиагональных систем, где ненулевые элементы находятся только на главной диагонали и двух соседних диагоналях. Метод прогонки требует  $O(n)$  операций, что делает его очень быстрым для трёхдиагональных систем. Кроме того, данный алгоритм отличается относительной простотой в реализации.

Метод прогонки применим для некоторого трёхточечного разностного уравнения:

$$A_i y_{i-1} - C_i y_i + B_i y_{i+1} = -F_i, \quad i = \overline{1, N-1} \quad (4.1)$$

с краевыми условиями

$$y_0 = \kappa_1 y_1 + v_1, \quad y_N = \kappa_2 y_{N-1} + v_2, \quad (4.2)$$

где коэффициенты  $A_i, B_i, C_i, \kappa_1, \kappa_2, v_1, v_2$  — заранее заданные числа.

Реализация метода состоит из двух этапов: прямой ход (определение прогоночных коэффициентов) и обратный ход (вычисление неизвестных значений переменных).

Прямой ход метода прогонки начинается с определения начальных коэффициентов  $\gamma_0$  и  $\beta_0$ . Причём начальное значение  $\gamma_0$  устанавливается равным  $\kappa_1$ , что соответствует первому граничному условию, а начальное значение  $\beta_0$  устанавливается равным  $v_1$ , что также соответствует первому граничному условию. Таким образом,  $\gamma_0 = \kappa_1, \beta_0 = v_1$ .

Следующий шаг — это итерация, в которой мы вычисляем прогоночные коэффициенты  $\gamma_{i+1}, \beta_{i+1}$  для всех узлов  $i = \overline{1, N-1}$ :

$$\gamma_{i+1} = \frac{B_i}{C_i - \gamma_i A_i}, \beta_{i+1} = \frac{A_i \beta_i + F_i}{C_i - \gamma_i A_i}. \quad (4.3)$$

Используя найденные ранее прогоночные коэффициенты, обратный ход алгоритма прогонки позволяет определить значения искомых переменных. Этот шаг является завершающим и приводит к нахождению окончательных решений для заданной системы линейных уравнений. Перед началом обратного хода необходимо установить значения для последних переменных, начиная с последнего уравнения системы:

$$y_N = \frac{\nu_2 + \kappa_2 \beta_N}{1 - \kappa_2 \gamma_N}. \quad (4.4)$$

После того как значение  $y_N$  вычислено, рассчитываются оставшиеся ненайденные значения  $y_i$  по формуле:

$$y_i = \gamma_{i+1} y_{i+1} + \beta_{i+1}, \quad i = \overline{0, N-1}. \quad (4.5)$$

Метод будет являться устойчивым, если выполнены условия устойчивости:

$$A_i > 0, B_i > 0, C_i \geq A_i + B_i, \quad |\kappa_1| + |\kappa_2| \leq 2, \quad 0 \leq |\kappa_l| \leq 1, \quad l = 1, 2$$

Таким образом, формулы (4.1) – (4.5) формирующие метод прогонки позволяют эффективно и точно решать системы линейных уравнений с трехдиагональной структурой.

Применяя алгоритм прогонки к решению итерационной задачи, сначала определим значения коэффициентов и решим системы уравнений для переменных относительно  $Z$ :

$$\begin{aligned} A_i &= 1, C_i = 2, B_i = 1, F_i = \frac{r_{i-1} - r_{i+1}}{2} h \left( Bo \cdot z_i + Q - \frac{z_{i+1} - z_{i-1}}{2h r_i} \right), \\ i &= \overline{1, N-1}, \\ \kappa_1 &= 1, \nu_1 = h \cos \alpha - \frac{h^2}{2} \sin \alpha (Bo \cdot z_0 + Q + \cos \alpha), \\ \kappa_2 &= 1, \nu_2 = h \cos \alpha + \frac{h^2}{2} \sin \alpha \left( Bo \cdot z_N + Q - \frac{\cos \alpha}{\delta} \right), \\ \gamma_{i+1} &= \frac{B_i}{C_i - \gamma_i A_i}, \quad \gamma_0 = \kappa_1, \\ \beta_{i+1} &= \frac{A_i \beta_i + F_i}{C_i - \gamma_i A_i}, \quad \beta_0 = \nu_1, \end{aligned}$$

$$z_N = \frac{v_2 + \kappa_2 \beta_N}{1 - \kappa_2 \gamma_N},$$

$$z_i = \gamma_{i+1} z_{i+1} + \beta_{i+1}, \quad i = \overline{0, N-1}. \quad (4.6)$$

А затем для переменных относительно  $r$ :

$$\begin{aligned} A_i &= 1, C_i = 2, B_i = 1, F_i = \frac{z_{i+1} - z_{i-1}}{2} h \left( Bo \cdot z_i + Q - \frac{z_{i+1} - z_{i-1}}{2 h r_i} \right), \\ i &= \overline{1, N-1}, \\ \kappa_1 &= 1, v_1 = -h \sin \alpha + \frac{h^2}{2} \cos \alpha (Bo \cdot z_0 + Q + \cos \alpha), \\ \kappa_2 &= 1, v_2 = h \sin \alpha + \frac{h^2}{2} \cos \alpha \left( Bo \cdot z_N + Q - \frac{\cos \alpha}{\delta} \right), \\ \gamma_{i+1} &= \frac{B_i}{C_i - \gamma_i A_i}, \quad \gamma_0 = \kappa_1, \\ \beta_{i+1} &= \frac{A_i \beta_i + F_i}{C_i - \gamma_i A_i}, \quad \beta_0 = v_1, \\ r_N &= \frac{v_2 + \kappa_2 \beta_N}{1 - \kappa_2 \gamma_N}, \\ r_i &= \gamma_{i+1} r_{i+1} + \beta_{i+1}, \quad i = \overline{0, N-1}. \end{aligned} \quad (4.7)$$

Таким образом, получается набор формул (4.6), (4.7) для вычисления значений переменных  $z$  и  $r$  с использованием метода прогонки. Эти уравнения включают в себя определение прогоночных коэффициентов, а также граничные условия, которые необходимы для корректного решения системы.

Весь процесс будет включать следующие шаги:

1. Определение значений коэффициентов  $A_i, B_i, C_i, F_i$  на основе входных данных, таких как шаг  $h$ , константа  $Q$ , объём  $U$ , угол примыкания  $\alpha$ , отношение радиусов сосуда  $\delta$ , значение Бонда  $Bo$ .
2. Создание массивов для хранения прогоночных коэффициентов  $\gamma$  и  $\beta$ , а также для значений переменных  $z$  и  $r$ .
3. Реализация итеративного процесса для вычисления значений  $\gamma$  и  $\beta$  с использованием формул (4.6), (4.7), полученных ранее.
4. Вычисление окончательных значений переменных  $z$  и  $r$  с учетом граничных условий.
5. Отображение полученных значений и их визуализация.

## 4.2 Вычислительный эксперимент

Задача численного моделирования состоит в изучении конфигурации равновесной поверхности жидкости с учётом оказываемой силы на ней. Чтобы достичь поставленной цели, необходимо решить исходную задачу при разных значениях числа Бонда  $Bo$ , фиксируя угол смачивания  $\alpha$ . При этом увеличение параметра  $Bo$  можно трактовать как усиление влияния гравитационных сил.

Для решения данной задачи применяется метод продолжения по параметру. Его суть заключается в том, что форма жидкости при каждом последующем значении числа Бонда  $Bo$ , начиная с  $Bo = 0$ , определяется на основе решения, полученного на предыдущем шаге для другого значения  $Bo$ . Такой подход позволяет эффективно отслеживать эволюцию формы капли при изменении внешних условий.

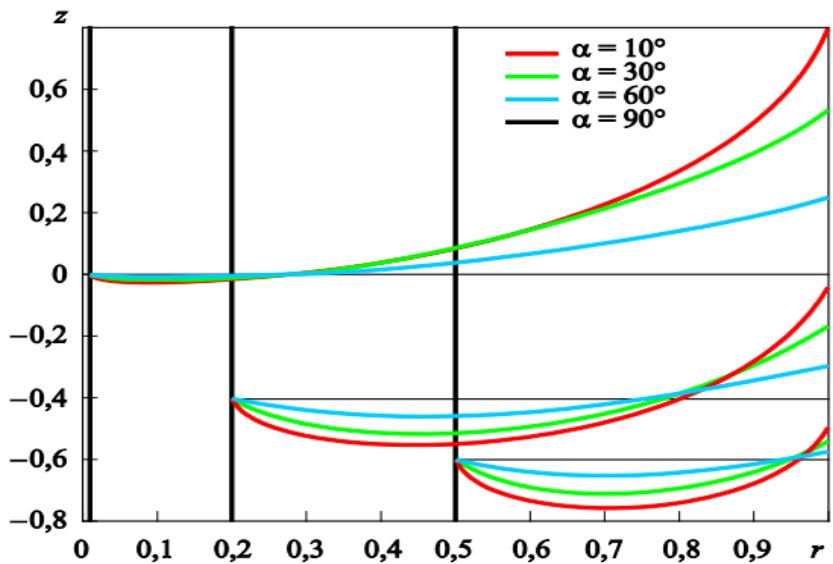
Точным решением поставленной задачи при  $Bo = 0$  и  $\alpha = 90^\circ$  является часть прямой, ограниченной на интервале от 1 до  $\delta$ . Тогда начальным приближением  $z_k^{(0)}, r_k^{(0)}, k = \overline{0, N}$ , для расчета формы капли при следующем значении  $Bo \neq 0$  можно взять следующие функции:

$$\begin{cases} z(s) = \delta - 1, \\ r(s) = 1 + s, 0 \leq s \leq \delta - 1. \end{cases}$$

Тогда начальный объём жидкости  $U = \pi(\delta^2 - 1)(\delta - 1)$ .

После того, как будут определены формы свободной поверхности капли при всех значениях числа Бонда из заданного диапазона ( $0 \leq Bo \leq Bo_{max}$ ), несколько характерных конфигураций можно изобразить графически.

На Рисунке 3 продемонстрированы формы равновесной поверхности жидкости в зазоре между коаксиальными цилиндрами:



**Рисунок 3 – Характерные формы свободной поверхности жидкости при значениях  $\delta = 0.01$ ,  $\delta = 0.2$ ,  $\delta = 0.5$**

Реализация данного метода была произведена на языке программирования Python. Текст кода приведен в приложении.

## ЗАКЛЮЧЕНИЕ

Таким образом, в рассматриваемой дипломной работе были решены поставленная задачи, начиная с формулировки математической постановки задачи и заканчивая тестированием разработанной программы. Теоретические аспекты работы обеспечили основу для практической реализации, включая разработку вычислительной модели и её численное решение. Было проведено исследование влияния параметров, включая угол смачивания  $\alpha$ , отношение радиусов цилиндров  $\delta$ , объём жидкости  $U$  и число Бонда  $Bo$ , определяющее баланс между силами тяжести и поверхностного натяжения, на форму равновесной капиллярной жидкости под воздействием массовых сил.

Исследование равновесных форм жидкости в зазоре между коаксиальными цилиндрами раскрывает множество аспектов взаимодействия жидкостей и твердых тел. Результаты данного анализа могут быть использованы для оптимизации конструкций, требующих работы с жидкостями в условиях ограниченных пространств, а также для прогнозирования поведения таких систем в экспериментальной и практической деятельности. Дальнейшие исследования могут включать влияние температуры, вязкости жидкости и других факторов, что позволит более полно описать сложные взаимодействия в таких системах.

## **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

- [1] Гидромеханика невесомости / В.Г. Бабский [и др.]; под ред. А.Д. Мышкиса. - М.: Наука, 1976. - 504с.
- [2] Будник А.М., Полевиков В.К. Моделирование свободной поверхности кольцевых магнитожидкостных уплотнений с помощью метода граничных элементов // Дифференциальные уравнения. - 1992. - Т. 28, № 7. - С. 1177-1181.
- [3] Будник А.М., Полевиков В.К. Осесимметричные формы равновесия жидкости в тороидальном сосуде при невесомости. Изв. АН СССР. Механика жидкости и газа, 1986, № 6, с. 154–156.
- [4] Самарский А.А., Гулин А.В. Численные методы. - М.: Наука, 1989. - 432 с.
- [5] Юй Джакай, Темнов А.Н. Исследование равновесной свободной поверхности капиллярной жидкости в тороидальном сосуде. Инженерный журнал: наука и инновации, 2021, вып. 3
- [6] Юй Чжаокай, Темнов А.Н. Равновесие и колебания свободной поверхности жидкого топлива в коаксиально-цилиндрических сосудах в условиях микрогравитации. Инженерный журнал: наука и инновации, 2021, вып. 8

# ПРИЛОЖЕНИЕ

## Код программы на языке Python

```
import numpy as np
import matplotlib.pyplot as plt

# Параметры
N = 100
U = 1
h = 1 / N
alpha = np.deg2rad(3.10)
sigma = 0.5

# Начальные условия
r_initial = np.linspace(sigma, 1, N + 1)
z_initial = np.zeros(N + 1)
r = np.copy(r_initial)
z = np.zeros(N + 1)

# Установка z[0] = 0
z[0] = 0

# Определение Q
Q = 2 / (1 - sigma**2) * ((1 + sigma) * np.cos(alpha))

# Коэффициенты
A = np.ones(N - 1) / h**2
B = np.ones(N - 1) / h**2
C = -2 * np.ones(N - 1) / h**2
F_r = np.zeros(N - 1)
F_z = np.zeros(N - 1)

# Границные условия для z
k1_z = 1
v1_z = h * np.cos(alpha) - h**2 / 2 * np.sin(alpha) * (Q + np.cos(alpha)) / sigma
k2_z = 1
v2_z = h * np.cos(alpha) + h**2 / 2 * np.sin(alpha) * (Q - np.cos(alpha))

# Прогонка для z
gamma_z = np.zeros(N - 1)
beta_z = np.zeros(N - 1)

gamma_z[0] = k1_z
beta_z[0] = v1_z

# Прямой ход для z
for i in range(1, N - 1):
    F_z[i] = (r[i + 1] - r[i - 1]) / (2 * h) * (Q - (z[i + 1] - z[i - 1]) / (2 * h * r[i]))
    gamma_z[i] = B[i] / (C[i] - gamma_z[i - 1] * A[i])
    beta_z[i] = (A[i] * beta_z[i - 1] + F_z[i]) / (C[i] - gamma_z[i - 1] * A[i])

# Вычисление z на правой границе
z[N - 1] = (v2_z + k2_z * beta_z[N - 2]) / (1 - k2_z * gamma_z[N - 2])

# Обратный ход для z
for i in range(N - 2, -1, -1):
    z[i] = gamma_z[i] * z[i + 1] + beta_z[i]
```

```

# Границные условия для r
k1_r = 1
v1_r = -h * np.sin(alpha) + h**2 / 2 * np.cos(alpha) * (Q + np.cos(alpha)) /
sigma)
k2_r = 1
v2_r = h * np.sin(alpha) + h**2 / 2 * np.cos(alpha) * (Q - np.cos(alpha))

# Прогонка для r
gamma_r = np.zeros(N - 1)
beta_r = np.zeros(N - 1)

gamma_r[0] = k1_r
beta_r[0] = v1_r

# Прямой ход для r
for i in range(1, N - 1):
    F_r[i] = (z[i + 1] - z[i - 1]) / (2 * h) * (Q - (z[i + 1] - z[i - 1])) /
(2 * h * r[i]))
    gamma_r[i] = B[i] / (C[i] - gamma_r[i - 1] * A[i])
    beta_r[i] = (A[i] * beta_r[i - 1] + F_r[i]) / (C[i] - gamma_r[i - 1] *
A[i])

# Вычисление r на правой границе
r[N - 1] = (v2_r + k2_r * beta_r[N - 2]) / (1 - k2_r * gamma_r[N - 2])

# Обратный ход для r
for i in range(N - 2, -1, -1):
    r[i] = gamma_r[i] * r[i + 1] + beta_r[i]

# Проверка, чтобы r оставался в пределах [sigma, 1]
r = np.clip(r, sigma, 1)

# Построение графика
plt.figure(figsize=(10, 6))
plt.plot(r_initial, z_initial, label='Начальное приближение', linestyle=':')
plt.plot(r, z, label='Найденная линия')
plt.xlabel('r')
plt.ylabel('z')
plt.legend()
plt.title('Форма линии')
plt.grid(True)
plt.show()

print("Значения z:", z)
print("Значения r:", r)

```