

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РФ
КЕМЕРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
КАФЕДРА ЮНЕСКО ПО НОВЫМ ИНФОРМАЦИОННЫМ ТЕХНОЛОГИЯМ

К.Е. Афанасьев, А.М. Гудов

**ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
В ЧИСЛЕННЫХ РАСЧЕТАХ**

Учебное пособие

Кемерово 2001

ББК В253.31я73+В192.1я73+3811я73

A94

УДК 532.5.031+519.6+681.3

*Печатается по решению редакционно-издательского
Совета Кемеровского университета*

Рецензенты: доктор физ.-мат. наук, профессор В.П. Житников,

кафедра вычислительной техники и информационных технологий Кузбасского государственного технического университета

A94 Афанасьев К.Е., Гудов А.М. Информационные технологии в численных расчетах: Учебное пособие. – Кемерово: КемГУ, 2001.- 204с.

ISBN 5-8353-0063-8

В пособии рассмотрены основные вопросы гидродинамики идеальной жидкости со свободными границами. Дано описание численных методов граничных элементов для решения таких задач.

Особое внимание уделено использованию пакета прикладных программ "*AKORD*" и разрабатываемой на его основе информационной системе поддержки вычислительного эксперимента. В качестве примеров, иллюстрирующих обоснованность использования описанных методов и алгоритмов, приведены результаты численного решения некоторых тестовых задач гидродинамики идеальной жидкости со свободными границами, полученные с использованием пакета "*AKORD*".

Пособие рассчитано на студентов старших курсов и аспирантов математического факультета, специализирующихся по вычислительной и прикладной математике.

А **1602120000**
ЛР№020464 Без объявл

ISBN 5 -8353-0063-8

ББК

© К.Е. Афанасьев
А.М. Гудов, 2001

© Кемеровский государственный
университет, 2001

ОГЛАВЛЕНИЕ

<u>ВВЕДЕНИЕ</u>	5
<u>Гла в а 1. НЕКОТОРЫЕ СВЕДЕНИЯ ИЗ ТЕОРИИ ИДЕАЛЬНОЙ НЕСЖИМАЕМОЙ ЖИДКОСТИ</u>	7
1.1. НАЧАЛЬНЫЕ СВЕДЕНИЯ О КРАЕВЫХ ЗАДАЧАХ	7
1.2. ОСНОВНЫЕ УРАВНЕНИЯ ТЕОРИИ ИДЕАЛЬНОЙ ЖИДКОСТИ	9
УРАВНЕНИЕ НЕРАЗРЫВНОСТИ	9
УРАВНЕНИЕ ДВИЖЕНИЯ	11
ИНТЕГРАЛЫ ДВИЖЕНИЯ	12
<u>Гла в а 2. ПОСТАНОВКА ЗАДАЧ СО СВОБОДНЫМИ ГРАНИЦАМИ, РЕШАЕМЫХ ПРИ ПОМОЩИ ПАКЕТА “AKORD”</u>	13
2.1. ВЕЩЕСТВЕННЫЕ ПЕРЕМЕННЫЕ	13
ПЛОСКИЙ СЛУЧАЙ	14
ОСЕСИММЕТРИЧНЫЙ СЛУЧАЙ	15
2.2. КОМПЛЕКСНЫЕ ПЕРЕМЕННЫЕ. ПЛОСКИЙ СЛУЧАЙ.....	18
2.3. ПРИВЕДЕНИЕ КРАЕВЫХ ЗАДАЧ К БЕЗРАЗМЕРНОМУ ВИДУ	21
<u>Гла в а 3. МАТЕМАТИЧЕСКИЕ МЕТОДЫ И АЛГОРИТМЫ, ИСПОЛЬЗУЕМЫЕ В ПП «AKORD»</u>	24
3.1. КМГЭ НА ОСНОВЕ ИНТЕГРАЛЬНОЙ ФОРМУЛЫ Коши	24
3.2. МОДИФИЦИРОВАННЫЙ КМГЭ	28
3.3. МГЭ НА ОСНОВЕ ТРЕТЬЕЙ ФОРМУЛЫ ГРИНА	30
ПЛОСКИЙ СЛУЧАЙ	30
ОСЕСИММЕТРИЧНЫЙ СЛУЧАЙ	38
3.4. МЕТОДЫ РЕШЕНИЯ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ	40
3.5. АЛГОРИТМЫ ПОСТРОЕНИЯ СВОБОДНЫХ ГРАНИЦ.....	43
СТАЦИОНАРНАЯ ЗАДАЧА	43
НЕСТАЦИОНАРНАЯ ЗАДАЧА	47
3.6. РАБОТА С ГРАФИЧЕСКИМИ ОБЪЕКТАМИ	48
БАЗОВЫЕ ГРАФИЧЕСКИЕ ОБЪЕКТЫ НА ПЛОСКОСТИ	49
БАЗОВЫЕ ПРОСТРАНСТВЕННЫЕ ГРАФИЧЕСКИЕ ОБЪЕКТЫ	51
3.7. ПОСТРОЕНИЕ СЕТОК В МЕТОДАХ ГРАНИЧНЫХ ЭЛЕМЕНТОВ.....	62
ПОСТРОЕНИЕ ДВУМЕРНОЙ ЧЕТЫРЕХУГОЛЬНОЙ СЕТКИ	63
ПОСТРОЕНИЕ ПРОСТРАНСТВЕННОЙ ТРЕУГУЛЬНОЙ СЕТКИ	67
<u>Гла в а 4. ПАКЕТ ПРИКЛАДНЫХ ПРОГРАММ “AKORD”</u>	73
4.1. ПРЕПРОЦЕССОР.....	74
4.2. РЕШАТЕЛИ.....	77
4.3. ПОСТПРОЦЕССОР	79
<u>Гла в а 5. ОТ ОТДЕЛЬНЫХ ПРОГРАММ К ИНТЕГРИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЕ ПОДДЕРЖКИ ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА</u>	85
5.1. СТРУКТУРА ИНФОРМАЦИОННОЙ СИСТЕМЫ.....	87

5.2. ЛОГИЧЕСКАЯ И ФИЗИЧЕСКАЯ МОДЕЛИ ДАННЫХ	92
5.3. ОБОЛОЧКА.....	96
5.4. ИНТЕРФЕЙСЫ ОБМЕНА ДАННЫМИ МЕЖДУ ПРИЛОЖЕНИЯМИ	101
ПРЕПРОЦЕССОР ДЛЯ ПЛОСКИХ И ОСЕСИММЕТРИЧНЫХ ЗАДАЧ	102
ПОСТПРОЦЕССОР	105
<i>Г л а в а 6. ПРИМЕРЫ ЗАДАЧ, РЕШАЕМЫХ С ПОМОЩЬЮ ППП "AKORD"</i>	108
6.1. ТЕСТОВЫЕ РАСЧЕТЫ	108
ТЕСТ №1. РЕШЕНИЕ УРАВНЕНИЯ ЛАПЛАСА В ЗАДАННОЙ ОБЛАСТИ	108
ТЕСТ №2. ТЕСТИРОВАНИЕ ИТЕРАЦИОННОГО АЛГОРИТМА	110
ТЕСТ №3. Движение уединенной волны по ровному дну	111
ТЕСТ №4. Движение солитона над прямоугольным уступом	113
ТЕСТ №5. Задача обтекания профиля Жуковского	114
ТЕСТ №6. Схлопывание сферической газовой полости (задача Рэлея)	117
ТЕСТ №7. Схлопывание газового пузыря возле твердой стенки	121
6.2. ПРИМЕРЫ РЕШЕННЫХ ЗАДАЧ	123
ЗАДАЧА №1. ОПРЕДЕЛЕНИЕ ФОРМЫ СВОБОДНОЙ ГРАНИЦЫ ПРИ ОБТЕКАНИИ ПОЛУКРУГОВОГО ЦИЛИНДРА	123
ЗАДАЧА №2. ИНТЕГРАЛЬНЫЕ ХАРАКТЕРИСТИКИ УЕДИНЕНИХ ВОЛН	126
ЗАДАЧА №3. НАКАТ УЕДИНЕНОЙ ВОЛНЫ НА ТВЕРДУЮ СТЕНКУ	132
ЗАДАЧА №4. ВЗАИМОДЕЙСТВИЕ СОЛИТОНОВ	135
ЗАДАЧА №5. Движение солитона над дном с полукруговым выступом	137
ЗАДАЧА №6. Горизонтальное движение полукругового цилиндра из состояния покоя по ровному дну	140
ЗАДАЧА №7. Совместное влияние твердых стенок и свободных границ на эволюцию газового пузыря	151
ЗАДАЧА №8. Эволюция пузыря в "сильно" ограниченном объеме	157
ЗАДАЧА №9. Исследование явлений на поверхности воды при схлопывании газовой полости	160
ЛИТЕРАТУРА	166
ПРИЛОЖЕНИЯ	174
ПРИЛОЖЕНИЕ 1. РАБОТА С ПРЕПРОЦЕССОРОМ ДЛЯ ПЛОСКИХ И ОСЕСИММЕТРИЧНЫХ ЗАДАЧ	174
ПРИЛОЖЕНИЕ 2. РАБОТА С ПОСТПРОЦЕССОРОМ	186
ПРИЛОЖЕНИЕ 3. ЕР-ДИАГРАММА МОДЕЛИ ДАННЫХ, ИСПОЛЬЗОВАННОЙ В ИС	195
ПРИЛОЖЕНИЕ 4. СТРУКТУРА ТАБЛИЦ БАЗЫ ДАННЫХ.....	196
ПРИЛОЖЕНИЕ 5. СТРУКТУРА ФАЙЛОВ ОБМЕНА ДАННЫМИ МЕЖДУ ПРИЛОЖЕНИЯМИ ИС	200

ВВЕДЕНИЕ

Проблемам проведения численных расчетов при решении сложных задач математической физики с использованием ресурсов ЭВМ посвящено немало книг и монографий. Однако немалый опыт работы авторов со студентами дает основание утверждать, что информационных источников, где бы использовался комплексный подход при описании методов, технологий и средств численного моделирования задач на ЭВМ, по-прежнему недостаточно. Сказываются также понятные затруднения при использовании соответствующей литературы в учебном процессе.

Настоящее учебное пособие написано на основе материалов, использованных авторами в различные периоды при чтении специальных курсов по численным методам, гидродинамике и использованию специализированных пакетов прикладных программ на математическом факультете Кемеровского государственного университета.

Основное вниманиеделено двум аспектам – описанию численных методов граничных элементов и технологии их практического использования при решении задач с помощью пакета прикладных программ "*AKORD*", разрабатываемого в течение ряда лет в Центре новых информационных технологий КемГУ для проведения численных расчетов в задачах гидродинамики со свободными границами.

Методы граничных элементов стали применяться сравнительно недавно. Однако широко проникли в различные области прикладных дисциплин. В данной работе достаточно полно представлены основные понятия и особенности использования этих методов на примере решения задач гидродинамики со свободными границами, описываемые эллиптическим уравнением.

В *первой главе* для удобства понимания основного материала приведены некоторые сведения из теории идеальной несжимаемой жидкости.

Постановка краевых задач со свободными границами приведена во *второй главе*, здесь же приводятся способы их обезразмеривания.

В *третьей главе* приводится описание методов граничных элементов, реализованных на основе интегральной формулы Коши (плоские задачи) и на

основе третьей формулы Грина (плоские, осесимметричные и трехмерные задачи). Здесь же кратко рассматриваются вопросы решения систем линейных алгебраических уравнений, построение плоских и пространственных сеток, работы с пространственными графическими объектами и приведены алгоритмы построения свободных границ.

Четвертая глава посвящена описанию структуры пакета и компонентов, использованных при его проектировании.

В *пятой главе* рассматриваются вопросы развития пакета прикладных программ до уровня интегрированной информационной системы поддержки вычислительного эксперимента. Описываются структуры информационной системы, логическая модель данных, рассматриваются интерфейсы обмена данными между приложениями.

Шестая глава посвящена решению конкретных (стационарных и нестационарных) задач гидродинамики идеальной жидкости со свободными границами, решенных с использованием пакета «**AKORD**». Приводится решение различных тестовых задач, которые могут быть полезны тем, кто занимается расчетами и тестированием численных алгоритмов.

Завершает учебное пособие раздел "*Приложения*". В нем приводятся подробные инструкции по работе с препроцессором для плоских/осесимметричных задач и постпроцессором. Здесь же приведены структуры таблиц баз данных, ER – диаграмма модели данных, используемой в информационной системе, структура файлов обмена данными между компонентами пакета.

В работе над пакетом прикладных программ активное участие принимали аспиранты Коротков Г.Г. – препроцессор и постпроцессор для двумерных задач, Долаев Р.Р. – постпроцессор для двухмерных задач и оболочки, Стуколов С.В. – решатель для плоских задач, выполненный на основе интегральной формулы Коши.

Пособие адресовано, прежде всего, студентам и аспирантам – математикам и физикам, специализирующимся в области математического моделирования и построения информационных систем. Хочется надеяться, что оно будет полезно и для специалистов, уже занимающихся вопросами численного моделирования физических процессов.

Предлагаемое учебное пособие является частью плановой работы, которая была определена департаментом информационных технологий Организации Объединенных Наций по Вопросам Образования, Науки и Культуры (UNESCO), при создании в Кемеровском государственном университете кафедры ЮНЕСКО по новым информационным технологиям в образовании и науке.

ГЛАВА 1

НЕКОТОРЫЕ СВЕДЕНИЯ ИЗ ТЕОРИИ ИДЕАЛЬНОЙ НЕСЖИМАЕМОЙ ЖИДКОСТИ

Модель идеальной несжимаемой жидкости находит широкое применение для моделирования различных гидродинамических задач, к которым относятся и стационарные (нестационарные) задачи гидродинамики со свободными границами. Подробное описание задач гидродинамики приведено во многих монографиях и учебниках [25, 36, 45, 47, 50, 73, 75].

1.1. Начальные сведения о краевых задачах

Многие важные для приложений задачи физики и техники (колебания, теплопроводность, диффузия, течение идеальной жидкости и др.) приводят к уравнениям эллиптического типа и решаются методами теории потенциала. Такие задачи содержат дифференциальные уравнения, отражающие тот или иной физический закон, и граничные условия, позволяющие выделить из множества решений одно, отвечающее именно этой задаче.

Постановки этих задач различны, искомые функции имеют разный физический смысл. Наиболее распространенными уравнениями этого типа являются уравнение Лапласа

$$\Delta \phi(R) = 0, \quad (1.1)$$

или уравнение Пуассона

$$\Delta \phi(R) = -f(R), \quad (1.2)$$

где $\phi(R)$ и $f(R)$ - функции радиус-вектора точки $R = R(x, y, z)$ в Евклидовом пространстве.

Функцию $\phi(R)$ - зачастую можно интерпретировать как потенциал некоторого безвихревого векторного поля, описываемого уравнением

$$\vec{V}(R) = -\text{grad} \phi(R). \quad (1.3)$$

Уравнения (1.1) и (1.2)) описывают стационарное распределение потенциала $\phi(R)$ внутри некоторого объема D , ограниченного поверхностью S .

Математическая задача о нахождении потенциала $\phi(R)$ формулируется следующим образом [81].

Найти функцию $\phi(R)$, удовлетворяющую внутри D уравнению (1.2) и граничным условиям, которые могут быть заданы в одном из следующих видов:

- I. $\phi = f_1$, на S (первая краевая задача).
II. $\frac{\partial \phi}{\partial n} = f_2$, на S (вторая краевая задача).
III. $\frac{\partial \phi}{\partial n} + \alpha(\phi - f_3) = 0$, на S (третья краевая задача);

где f_1, f_2, f_3, α - это заданные функции, $\frac{\partial \phi}{\partial n}$ - производная по внешней нормали к поверхности S .

Первую краевую задачу для уравнения Лапласа называют «задачей Дирихле», а вторую – «задачей Неймана». Если на одной части S_1 поверхности S задано условие Дирихле, а на другой $S_2 = S \setminus S_1$ – условие Неймана, то краевая задача называется смешанной краевой задачей.

Если решение ищется в области D_0 , внутренней (или внешней) по отношению к поверхности S , то соответствующая задача называется внутренней (или внешней) краевой задачей.

Физическое содержание каждого конкретного потенциала (при решении различных физических задач) можно найти во многих учебниках и монографиях (например, [45, 47, 50, 73]).

1.2. Основные уравнения теории идеальной жидкости

Уравнение неразрывности

Пусть D - фиксированная область с границей S , занятая идеальной несжимаемой и однородной жидкостью, то есть жидкостью с постоянной во всей области течения плотностью $\rho = \text{const}$.

Скорость изменения массы жидкости в области D , обусловленная движением жидкости через границу S со скоростью \vec{v} , равна

$$\int_S \vec{n} \times \rho \vec{v} ds.$$

Здесь \vec{n} - вектор единичной внешней нормали к элементу поверхности ds . Символ “ \times ” обозначает векторное произведение, а символ “ \cdot ” далее по тексту означает скалярное произведение векторов.

С другой стороны, скорость изменения массы жидкости в области D (элемент объема которой обозначается dV) дается формулой

$$\int_D \frac{\partial c}{\partial t} dV.$$

Поскольку материя не исчезает никуда и не возникает ниоткуда, то закон сохранения массы можно выразить уравнением

$$\int_D \frac{\partial \rho}{\partial t} dV = - \int_S \vec{n} \cdot \rho \vec{v} ds. \quad (1.4)$$

Знак минус перед интегралом в правой части взят потому, что этот интеграл положителен, если через поверхность S за единицу времени вытекает больше жидкости, чем втекает, что способствует

уменьшению плотности во времени. Иными словами, интегралы, входящие в выражение (1.4), всегда имеют разные знаки.

Применяя к преобразованию интеграла по границе формулу Гаусса, приведем уравнение (1.4) к виду

$$\int_D \left[\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \vec{v} \right] dV = 0.$$

Поскольку объем D произвольный, то подинтегральная функция равна нулю, т.е.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \vec{v} = 0. \quad (1.5)$$

Вводя в рассмотрение материальную производную

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \vec{v} \cdot \nabla, \quad (1.6)$$

уравнение неразрывности (1.5) можно преобразовать к форме

$$\frac{d\rho}{dt} + \rho \nabla \cdot \vec{v} = 0,$$

которая для несжимаемой однородной жидкости с постоянной плотностью приводится к соотношению

$$\nabla \cdot \vec{v} = 0. \quad (1.7)$$

Если в выражении (1.3) скалярную функцию $\phi(R)$ считать потенциалом поля скоростей, то функция, стоящая в левой части этого выражения будет представлять вектор скорости частиц жидкости. Подставляя (1.3) в (1.7), получим уравнение Лапласа (1.1), описывающее безвихревое течение идеальной несжимаемой жидкости.

Возможность безвихревого течения идеальной жидкости доказывает теорема Лагранжа [45].

Уравнение движения

Основным динамическим уравнением движения материальной точки является второй закон Ньютона [73]

$$\vec{F} = m \frac{d\vec{v}}{dt} = \frac{dm\vec{v}}{dt}, \quad (1.8)$$

где m – масса материальной точки, а \vec{v} – ее скорость.

Произведение массы на скорость $m\vec{v}$ называется количеством движения точки.

С помощью уравнения количества движения (1.8) можно решать две типичные задачи – по известным силам найти закон движения точки или по известному закону движения точки найти действующую на нее силу.

Обобщение уравнения количества движения на случай конечного объема D сплошной среды, ограниченного поверхностью S , в случае предположения об идеальности жидкости и несжимаемости среды позволяет написать основное уравнение количества движения.

$$\int_D \rho \frac{d\vec{v}}{dt} dV = - \int_S p \cdot \vec{n} ds + \int_D \rho \vec{B} dV, \quad (1.9)$$

где p – давление, \vec{B} – внешние массовые силы.

Если массовые силы \vec{B} имеют потенциал G (т.е. $\vec{B} = \text{grad}G$), то преобразуя в уравнении (1.9) интеграл по поверхности с помощью теоремы Гаусса в объемный интеграл, получим

$$\int_D \rho \frac{d\vec{v}}{dt} dV = - \int_D (\rho \nabla G + \nabla p) dV,$$

откуда следует, что

$$\rho \frac{d\vec{v}}{dt} = - \text{grad}(G + \frac{p}{\rho}). \quad (1.10)$$

Интегралы движения

Уравнение (1.10) записано в предположении, что жидкость является идеальной и несжимаемой. Если дополнительно предположить, что течение безвихревое, массовые силы определяются силой тяжести с вектором \vec{g} , направленным противоположно оси z, то из (1.10) с учетом (1.6) следует интеграл Коши-Лагранжа:

$$\frac{\partial \phi}{\partial t} + \frac{1}{2}(\nabla \phi)^2 + gz + \frac{p}{\rho} = c(t). \quad (1.11)$$

Здесь $c(t)$ - функция от времени, равная значению левой части в некоторой точке пространства. Если жидкость на бесконечности покоятся и давление на уровне $z=0$ равно нулю, то $c(t) = 0$.

Для численных расчетов нестационарных течений удобно пользоваться интегралом движения, записанным для фиксированной частицы жидкости. Из выражения (1.6) следует, что $\frac{d\phi}{dt} = \frac{\partial \phi}{\partial t} + (\nabla \phi)^2$ и интеграл (1.11) преобразуется к виду:

$$\frac{d\phi}{dt} - \frac{1}{2}(\nabla \phi)^2 + gz + \frac{p}{\rho} = c(t). \quad (1.12)$$

В случае стационарного течения, когда в фиксированной точке пространства все функции с течением времени сохраняют постоянное значение, $\partial \phi / \partial t = 0$ и из (1.11) следует интеграл Бернулли

$$\frac{1}{2}(\nabla \phi)^2 + gz + \frac{p}{\rho} = c. \quad (1.13)$$

Г л а в а 2

ПОСТАНОВКА ЗАДАЧ СО СВОБОДНЫМИ ГРАНИЦАМИ, РЕШАЕМЫХ ПРИ ПОМОЩИ ПАКЕТА “AKORD”

Основой решения всех краевых задач и задач со свободными границами, в частности, является корректность задания начальных и граничных условий или, другими словами, – корректности постановки краевой задачи. Немаловажным является вопрос выбора характерных размерных параметров для обезразмеривания поставленной краевой задачи.

2.1. Вещественные переменные

Рассматриваемые ниже краевые задачи являются весьма сложными для решения как аналитическими, так и численными методами в силу их нелинейности, которая определяется заранее неизвестной формой свободной поверхности, а также нелинейностью динамического условия, записанного в виде интегралов Бернули или Коши-Лагранжа.

Стационарная задача

Рассмотрим некоторую гипотетическую область решения D (рис. 1), ограниченную свободной поверхностью C_1 , твердой стенкой C_3 и двумя границами C_2 и C_4 , через которые протекает жидкость. Глубина жидкости на бесконечности принимается равной H , скорость течения V_∞ , давление на свободной поверхности предполагается равным нулю ($p = 0$), ускорение силы тяжести \vec{g} направ-

лено против направления оси y . Границы C_2 и C_4 должны быть выбраны на достаточно большом удалении от выступа с тем, чтобы поток через эти участки границы был близок к параллельному.

Плоский случай

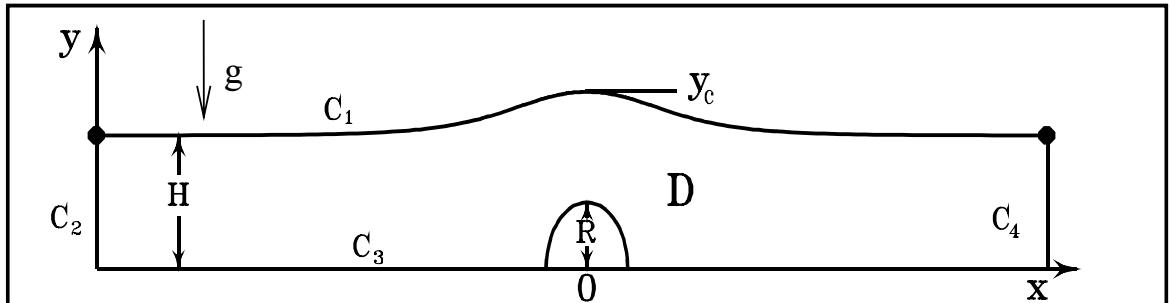


Рис. 1. Схема области течения и основные обозначения

Потенциал поля скоростей $\phi(\vec{x})$ внутри этой области должен удовлетворять уравнению Лапласа (1.1). На твердой границе выполняется условие «непротекания» (условие Неймана) $\frac{\partial \phi}{\partial n} = 0$, $\vec{x} \in C_3$. На боковых участках границы должны также выполняться краевые условия второго рода (условия втекания/вытекания жидкости):

$$\frac{\partial \phi}{\partial n} = \mp V_\infty, \quad \vec{x} \in C_2, C_4. \quad (2.1)$$

Свободная граница C_1 является линией тока и на ней должны выполняться кинематическое условие $\frac{\partial \phi}{\partial n} = 0$, $x \in C_1(t)$ и динамическое условие при $p=0$. Из уравнения Бернулли имеем:

$$\frac{1}{2} |\nabla \phi|^2 + gy = \frac{V_\infty^2}{2} + gH, \quad \vec{x} \in C_1, \quad (2.2)$$

здесь \vec{x} - радиус-вектор точки области решения, \vec{n} - вектор внешней нормали.

Нестационарная задача

Нестационарные задачи, описываемые в данном пособии, решаются в ограниченной области течения и участки области течения

справа и слева являются твердыми стенками, в то время как в стационарных задачах жидкость протекает через область D .

Схематически область течения представлена на рис. 2.

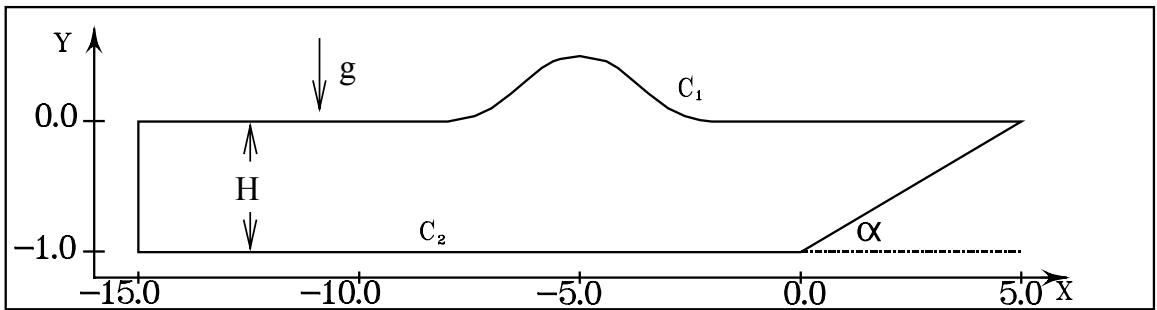


Рис. 2. Схема области течения и основные обозначения

Пусть в расчетной области течения $D(t)$, ограниченной свободной поверхностью C_1 и твердыми стенками C_2 (рис. 2), решается нестационарная краевая задача. Если потенциал поля скоростей зависит от времени $\phi(\vec{x}, t)$, то задача удовлетворяет уравнению Лапласа. На твердой стенке C_2 выполняется условие Неймана ($\partial\phi/\partial n = 0$), а на свободной поверхности C_1 должны выполняться кинематическое и динамическое условия:

$$\frac{d\vec{x}}{dt} = \nabla \phi, \quad \vec{x} \in C_1. \quad (2.3)$$

$$\frac{d\phi}{dt} - \frac{1}{2} |\nabla \phi|^2 + gy = gH, \quad x \in C_1. \quad (2.4)$$

В качестве начальных условий необходимо задать начальное положение свободной границы C_1^0 и распределение потенциала ϕ^0 на ней.

Таким постановкам удовлетворяют задачи о течении идеальной жидкости в каналах различной глубины с неровным дном, о движении одиночных волн, о взаимодействии волн с препятствиями, о движении тел в жидкости и т.д.

Осьсимметричный случай

Нестационарная задача

Само название «осесимметричная задача» говорит о том, что у задачи существует осевая симметрия. Часто за ось симметрии при-

нимается координата z , а система декартовых координат преобразуется к цилиндрической системе по следующему закону $(x, y, z) \rightarrow (r \cos \theta, r \sin \theta, z)$ и предполагается, что процесс не зависит от угла поворота θ .

В новой системе координат уравнение Лапласа запишется в виде:

$$\frac{\partial^2 \phi}{\partial z^2} + \frac{\partial^2 \phi}{\partial r^2} + \frac{1}{r} \frac{\partial \phi}{\partial r} = 0. \quad (2.5)$$

На оси симметрии $r = 0$ ставится условие непротекания:

$$\left. \frac{\partial \phi}{\partial n} \right|_{r=0} = 0. \quad (2.6)$$

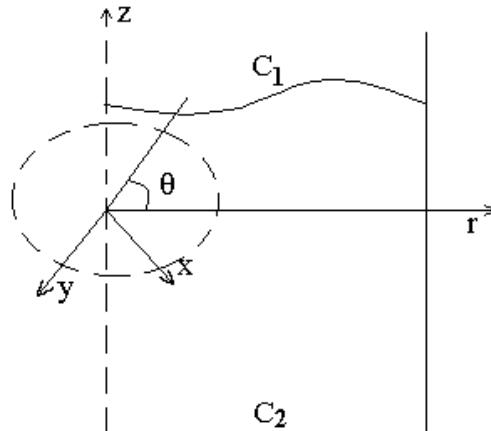


Рис. 3. Переход к цилиндрической системе координат

Остальные граничные условия и условия на свободной границе остаются без изменения. Например, в случае, изображенном на (рис.3), граничные условия будут иметь вид:

$$\begin{aligned} \frac{d\phi}{dt} - \frac{1}{2} |\nabla \phi|^2 + gz &= 0, \quad (z, r) \in C_1(t), \\ \frac{dr}{dt} = \frac{\partial \phi}{\partial r}, \quad \frac{dz}{dt} = \frac{\partial \phi}{\partial z}, &(z, r) \in C_1(t), \\ \frac{\partial \phi}{\partial n} &= 0, \quad (z, r) \in C_2. \end{aligned}$$

Приведенная постановка охватывает задачи колебания жидкости в сосудах, динамики кавитационной полости в ограниченном или безграничном пространстве и т.д.

Поскольку при помощи пакета “AKORD” решаются задачи динамики газового пузыря в идеальной жидкости, то целесообразно привести постановку данной задачи в более общем виде [63].

Пусть в пространстве R^3 точек $\vec{x}=(x,y,z)$ рассматривается область $D(t)$, зависящая от времени $t \geq 0$, ограниченная гладкими поверхностями $S(t)$ и Γ . Поверхность Γ является твердой стенкой, а поверхность $S(t)$ является границей односвязной ограниченной области $Q(t)$ (пузырь), расположенной вблизи Γ (рис. 4).

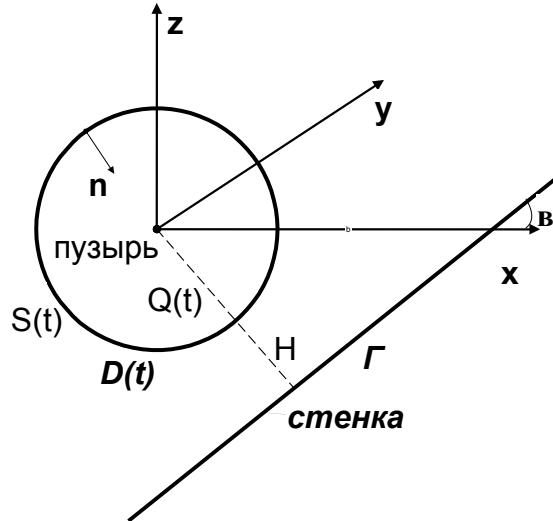


Рис. 4. Начальная геометрия пузыря возле наклонной стенки

Сделаем некоторые предположения относительно движения идеальной несжимаемой жидкости. На бесконечности жидкость поконится. На жидкость действует поле сил тяжести. Давление на бесконечности постоянно и равно p_a . Давление $p_s(t)$ в пузыре $Q(t)$ меняется по адиабатическому закону $p_s = C|Q(t)|^{-\gamma}$, где $|Q(t)|$ - объем области $Q(t)$, $C=const$, $\gamma = const > 1$. Массой газа заключенного в пузыре $Q(t)$ и силами поверхностного натяжения можно пренебречь. Начальное состояние при $t=0$ задается условиями: $S(0)$ – сфера радиуса R с центром в точке $(0,0,0)$, расстояние до поверхности Γ равно H , причем $H > R$, давление в $Q(0)$ задано и равно p_0 .

Краевая задача для уравнения Лапласа ставится обычным образом, а потенциал скорости $\phi(\vec{x},t)$ и гидродинамическое давление $p(\vec{x},t)$ должны удовлетворять интегралу Коши – Лагранжа

$$\frac{d\phi}{dt} - \frac{1}{2} |\nabla \phi|^2 + gz + \frac{1}{\rho} p_s = \frac{1}{\rho} p_a, \quad \vec{x} \in S(t). \quad (2.7)$$

В силу сделанных предположений $p_s = p_0 \left[|Q(t)| / |Q(0)| \right]^{-\gamma}$,
 $|Q(0)| = \frac{4}{3} \pi R^3$.

2.2. Комплексные переменные. Плоский случай

Для решения плоских краевых задач весьма эффективным является аппарат теории функций комплексного переменного, хорошо зарекомендовавший себя в различных областях знаний [36, 45, 50].

Стационарная задача

Пусть в расчетной области течения $D(t)$, ограниченной свободной поверхностью $C_1(t)$ и твердыми стенками C_2 (рис. 1), решается уравнение Лапласа:

$$\Delta w(z) = 0, \quad z = z(x, y) \in D(t), \quad (2.8)$$

для функции комплексного потенциала $w(z) = \phi(x, y) + i\psi(x, y)$, где $\phi(x, y)$ - функция потенциала скоростей, $\psi(x, y)$ - функция тока, удовлетворяющие условиям Коши-Римана [45]:

$$\frac{\partial \phi}{\partial x} = \frac{\partial \psi}{\partial y}, \quad \frac{\partial \phi}{\partial y} = -\frac{\partial \psi}{\partial x}.$$

На твердых границах выполняется условие непротекания:

$$\psi = 0, \quad z \in C_2. \quad (2.9)$$

На боковых участках границы области и на дне выполняются краевые условия вида

$$\begin{aligned} \psi(x, y) &= y, & z \in C_2, C_4, \\ \psi(x, y) &= 0, & z \in C_3. \end{aligned} \quad (2.10)$$

Свободная граница является линией тока ($\psi(x, y) = H$), на которой справедливо уравнение Бернулли (1.13), в данном случае имеющее вид:

$$\frac{1}{2} \left| \frac{dw}{dz} \right|^2 + gy = \frac{V_\infty^2}{2} + gH, \quad z \in C_1(t). \quad (2.11)$$

Стационарная задача циркуляционного обтекания тел

Задачам циркуляционного обтекания профилей посвящено множество работ, но, как правило, в них область течения безгранична, либо ограничена твердыми стенками. Наличие свободной границы привносит дополнительную трудность в численное моделирование данной задачи.

Пусть одиночный профиль с границей C_5 обтекается потоком весомой жидкости, ограниченной свободной поверхностью $C_1(t)$, прямолинейным дном C_3 , участками втекания C_2 и вытекания C_4 (рис.5). Введем обозначения: V_∞ - скорость втекающего потока, H - глубина жидкости на бесконечности, g - ускорение силы тяжести. Границы C_2 и C_4 должны быть выбраны на достаточно большом удалении от обтекаемого объекта с тем, чтобы линии тока жидкости, втекающей через эти участки границы, были параллельны.

Для решения задачи циркуляционного обтекания профилей применять метод комплексных граничных элементов с использованием интегральной формулы Коши для функции комплексного потенциала $w(z)$ затруднительно, так как потенциал скорости $\phi(x, y)$ неоднозначно определен и при циркуляции отличной от нуля терпит разрыв первого рода в острой кромке профиля.

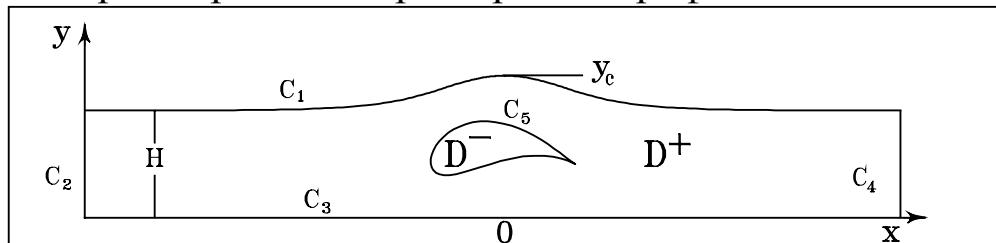


Рис. 5. Схема области течения и основные обозначения

Поскольку поле скоростей остается непрерывным, то данную задачу проще решать в терминах комплексно-сопряженной скорости

$$W(z) = \frac{\partial \phi}{\partial x} + i \frac{\partial \psi}{\partial x} = \frac{\partial \phi}{\partial x} - i \frac{\partial \phi}{\partial y} = V_x - iV_y. \quad (2.12)$$

Пусть ξ - точка, принадлежащая контуру, тогда выражение (2.12) может быть записано в виде

$$W(\xi) = V_x(\xi) - iV_y(\xi) = (V_n(\xi) - iV_s(\xi))e^{-i\alpha(\xi)},$$

где V_s и V_n касательная и нормальная компоненты вектора скорости в точке ξ , α - угол между вектором V_n и осью OX .

Задача об обтекании профиля может быть сведена к решению уравнения Лапласа

$$\Delta W(z) = 0, \quad z = x + iy \in D(t),$$

для аналитической в области течения D функции $W(z)$. На профиле и на дне выполняется условие непротекания $V_n = 0$, $z \in C_5, C_3$. На боковых участках границы области ставятся условия втекания и вытекания жидкости

$$V_n = \mp V_\infty, \quad z \in C_2, C_4. \quad (2.13)$$

Если обозначить касательные составляющие векторов скорости на верхней и нижней сторонах профиля через V_s^+ и V_s^- , то условие Жуковского-Чаплыгина в острой кромке можно записать следующим образом:

$$V_s^+ + V_s^- = 0. \quad (2.14)$$

В силу того, что нормальные составляющие V_n^+ и V_n^- вектора скорости на профиле равны нулю, а касательные составляющие V_s^+ и V_s^- равны по величине и противоположны по знаку, то если ввести обозначение $V_s^0 = V_s^+ + V_s^-$, условие Жуковского-Чаплыгина в острой кромке запишется следующим образом:

$$V_s^0 = 0. \quad (2.15)$$

Свободная граница является линией тока, на которой справедливо уравнение Бернулли

$$\frac{1}{2}|W(z)|^2 + gy = \frac{V_\infty^2}{2} + gH, \quad z \in C_1. \quad (2.16)$$

Нестационарная задача

Пусть в расчетной области $D(t)$ (рис. 5) решается нестационарная задача для уравнения Лапласа (2.8). Краевые условия на твердой границе имеют вид (2.9), а на свободной поверхности выполняются кинематическое и динамическое условия:

$$\frac{dz}{dt} = \frac{\partial \phi}{\partial x} + i \frac{\partial \phi}{\partial y}, \quad z \in C_1(t). \quad (2.17)$$

$$\frac{d\phi}{dt} - \frac{1}{2} \left| \frac{dw}{dz} \right|^2 + g(y - H) = 0, \quad z \in C_1(t). \quad (2.18)$$

Кроме того, необходимо задать начальные условия - положение свободной границы C_1^0 и распределение потенциала ϕ^0 на ней.

2.3. Приведение краевых задач к безразмерному виду

При проведении численных расчетов целесообразно перейти в исходных уравнениях к безразмерным переменным. Для этого необходимо выбрать некоторые характеристические для решаемой задачи независимые размерные переменные. Подробную информацию о методах обезразмеривания можно найти во многих учебниках по гидродинамике, например [45, 73].

Стационарные задачи

Для стационарных задач наиболее часто выбираются в качестве независимых размерных величин скорость втекающего потока V_∞ и глубина канала H .

Введем безразмерные переменные $\bar{x} = x/H$, $\bar{y} = y/H$, $\bar{\phi} = \phi/HV_\infty$, $\bar{V} = V/V_\infty$ и сделаем замену переменных в уравнениях, описывающих стационарную краевую задачу.

Все уравнения и граничные условия останутся без изменения за исключением условий (2.1), (2.2), (2.11) и (2.13), которые преобразуются к виду

<u>В вещественных переменных</u>	<u>В комплексных переменных</u>
$V_n = \mp 1,$	$V_n = \mp 1,$
$ \nabla \phi ^2 = 1 + 2 \frac{1-y}{Fr^2},$	$\left \frac{dw}{dz} \right ^2 = 1 + 2 \frac{1-y}{Fr^2},$

где $Fr = \frac{V_\infty}{\sqrt{gH}}$ - число Фруда, (черта над безразмерными переменными опущена).

Нестационарные задачи

В нестационарном случае при решении задач о взаимодействии уединенных волн с различными видами препятствий удобно в качестве характерных размерных величин выбирать глубину невозмущенного слоя жидкости H и ускорение силы тяжести g .

Безразмерные переменные в этом случае могут быть введены следующим образом:

$$\begin{aligned}\bar{x} &= x/H, \quad \bar{y} = y/H, \quad \bar{\phi} = \phi/H\sqrt{gH}, \quad \bar{V} = V/\sqrt{gH}, \\ \bar{t} &= t/\sqrt{H/g}, \quad \bar{p} = p/\rho g H.\end{aligned}$$

Краевая задача и граничные условия останутся без изменений, за исключением уравнений (2.4) и (2.18).

<u>В вещественных переменных</u>	<u>В комплексных переменных</u>
$\frac{d\phi}{dt} - \frac{1}{2} \nabla \phi ^2 + (y - H) = 0,$	$\frac{d\phi}{dt} - \frac{1}{2} \left \frac{dw}{dz} \right ^2 + (y - H) = 0,$

здесь, как и раньше черта над безразмерными переменными опущена.

Когда в нестационарных задачах речь идет об эволюции газового пузыря, то характерными размерными величинами, как правило, являются максимальный радиус пузыря R и ускорение силы тяжести g . Безразмерные переменные записываются аналогично предыдущему случаю.

Как и раньше все исходные уравнения остаются без изменений, и лишь интеграл Коши-Лагранжа (2.7) преобразуется к виду:

$$\frac{d\phi}{dt} - \frac{1}{2} |\nabla \phi|^2 + z = \beta + \alpha q(t),$$

где

$$\alpha = \frac{p_s}{\rho R g}, \quad \beta = -\frac{p_s - p_a}{\rho R g}, \quad q(t) = 1 - \left[|Q(t)| / |Q(0)| \right]^{-\gamma}.$$

Г л а в а 3

МАТЕМАТИЧЕСКИЕ МЕТОДЫ И АЛГОРИТМЫ, ИСПОЛЬЗУЕМЫЕ В ППП «AKORD»

Методы граничных элементов представляют собой разновидность методов граничных интегральных уравнений, являющихся одним из мощных методов решения задач гидродинамики идеальной несжимаемой жидкости. Суть метода состоит в преобразовании краевой задачи для уравнения Лапласа в исследуемой области к решению интегрального уравнения, записанного по границе этой области. В численных расчетах это приводит к необходимости дискретизации только границы рассматриваемой области. Вместе с тем, если возникает необходимость отыскания решения в любой внутренней точке, это можно сделать, используя уже известные значения функций на границе.

Свести задачу для дифференциального уравнения эллиптического типа к граничному интегральному уравнению можно различными способами. Ниже описаны методы, реализованные в настоящей версии пакета прикладных программ «*AKORD*» - комплексный метод граничных элементов (КМГЭ), построенный на использовании интегральной формулы Коши [32], и метод граничных элементов (МГЭ), построенный на использовании третьей формулы Грина [21, 24, 78].

3.1. КМГЭ на основе интегральной формулы Коши

Для области D , ограниченной кусочно-гладкой замкнутой границей $C = C_1 \cup C_2$, справедлива интегральная формула Коши, кото-

рую можно записать с помощью предельных формул Коши для векторной функции вида:

$$w(z_0) = \frac{1}{\varepsilon(z_0)i} \int_C \frac{w(z)}{z - z_0} dz, \quad (3.1)$$

где $\varepsilon(z_0) = 2\pi$ для внутренней точки, $\varepsilon(z_0) = \pi$ для точки на гладкой границе C , $\varepsilon(z_0) = \beta$ для угловой точки границы C (β - угол при вершине). Положительное направление обхода контура C берется таким образом, чтобы область D оставалась слева.

Учитывая, что на свободной поверхности известна действительная часть $\phi(x, y)$ функции $w(z)$, а мнимая часть $\psi(x, y)$ известна на твердых стенках, мы имеем смешанную краевую задачу для функции $w(z)$.

Численное решение задачи можно получить, разбив контур C на N линейных элементов Γ_j с узлами $z_j (j=1, N)$. Тогда $w(z) = \lim_{\max|\Gamma_j| \rightarrow 0} G(z)$, где $G(z)$ - линейная глобальная пробная функция для $z \in \sum_{j=1}^N \Gamma_j$ и $G(z) = \sum_{j=1}^N w_j \Lambda_j(z)$, w_j - значение $w(z)$ в точке z_j , $\Lambda_j(z)$ - линейная базисная функция:

$$\Lambda_j(z) = \begin{cases} (z - z_j)/(z_j - z_{j-1}), & z \in \Gamma_{j-1}, \\ (z_{j+1} - z)/(z_{j+1} - z_j), & z \in \Gamma_j, \\ 0, & z \notin \Gamma_{j-1} \cup \Gamma_j. \end{cases}$$

Вычисление интегралов

После указанного разбиения и линейной аппроксимации функции $w(z)$ на границе интеграл Коши можно вычислить аналитически в смысле главного значения при $z \rightarrow z_j$.

$$2\pi i w(z_j) = \lim_{z \rightarrow z_j^-} \int_C \frac{G(\zeta)}{\zeta - z} d\zeta.$$

Обозначение $z \rightarrow z_j^-$ означает, что точка z стремится к точке z_j , оставаясь все время внутри области D .

$$\begin{aligned} \lim_{z \rightarrow z_j^-} \int_C \frac{G(\zeta)}{\zeta - z} d\zeta &= \lim_{z \rightarrow z_j^-} \int_{\Gamma_j} \frac{G(\zeta)}{\zeta - z} d\zeta + \\ &+ \lim_{z \rightarrow z_j^-} \int_{\Gamma_{j+1}} \frac{G(\zeta)}{\zeta - z} d\zeta + \sum_{\substack{m=1 \\ m \neq j \\ m \neq j+1}}^N \int_{\Gamma_m} \frac{G(\zeta)}{\zeta - z_j} d\zeta. \end{aligned} \quad (3.2)$$

Введем обозначение: $I_m = \int_{\Gamma_m} \frac{G(\zeta)}{\zeta - z_j} d\zeta$. В выражении (3.2) интегралы под знаком суммы не имеют особенности и вычисляются следующим образом:

$$\begin{aligned} I_m &= \int_{\Gamma_m} \frac{G(\zeta)}{\zeta - z_j} d\zeta = \int_{z_m}^{z_{m+1}} \frac{w_m \Lambda_m + w_{m+1} \Lambda_{m+1}}{\zeta - z_j} d\zeta = \\ &= w_{m+1} - w_m + \left[\frac{(z_j - z_m)w_{m+1}}{z_{m+1} - z_m} - \frac{(z_j - z_{m+1})w_m}{z_{m+1} - z_m} \right] \ln \left(\frac{z_{m+1} - z_j}{z_m - z_j} \right). \end{aligned} \quad (3.3)$$

Два оставшихся интеграла имеют особенности при $z \rightarrow z_j$ и могут быть представлены в виде:

$$\begin{aligned} I_j + I_{j+1} &= \lim_{z \rightarrow z_j} \int_{z_{j-1}}^{z_j} \frac{w_{j-1} \Lambda_{j-1} + w_j \Lambda_j}{\zeta - z_j} d\zeta + \lim_{z \rightarrow z_j} \int_{z_j}^{z_{j+1}} \frac{w_j \Lambda_j + w_{j+1} \Lambda_{j+1}}{\zeta - z_j} d\zeta = \\ &= \lim_{z \rightarrow z_j} \left[\frac{w_j - w_{j-1}}{z_j - z_{j-1}} \int_{z_{j-1}}^{z_j} \frac{\zeta - z}{\zeta - z_j} d\zeta + \left(\frac{z - z_{j-1}}{z_j - z_{j-1}} w_j - \frac{z - z_j}{z_j - z_{j-1}} w_{j-1} \right) \int_{z_{j-1}}^{z_j} \frac{d\zeta}{\zeta - z} \right] + \end{aligned}$$

$$+ \lim_{z \rightarrow z_j} \left[\frac{w_{j+1} - w_j}{z_{j+1} - z_j} \int_{z_j}^{z_{j+1}} \frac{\zeta - z}{\zeta - z_j} d\zeta + \left(\frac{z - z_j}{z_{j+1} - z_j} w_{j+1} - \frac{z - z_{j+1}}{z_{j+1} - z_i} w_j \right) \int_{z_j}^{z_{j+1}} \frac{d\zeta}{\zeta - z} \right].$$

Комбинируя четные и нечетные слагаемые по индексу j , получим:

$$I_j + I_{j+1} = w_{j+1} - w_{j-1} + w_j \ln \left(\frac{z_{j+1} - z_j}{z_{j-1} - z_j} \right).$$

Комплексный логарифм в последнем слагаемом расписывается на действительную и мнимую части следующим образом:

$$\begin{aligned} \ln \left(\frac{z_{j+1} - z_j}{z_{j-1} - z_j} \right) &= \ln \left| \frac{z_{j+1} - z_j}{z_{j-1} - z_j} \right| + i \left[\arg(z_{j+1} - z_j) - \arg(z_{j-1} - z_j) \right] = \\ &= \ln \left| \frac{z_{j+1} - z_j}{z_{j-1} - z_j} \right| + i(2\pi - \varepsilon(z_j)), \end{aligned}$$

где $\varepsilon(z_j)$ - угол при вершине, образованный Γ_j и Γ_{j+1} элементами.

Таким образом, выражение (3.2) окончательно может быть записано:

$$\varepsilon i w_j = w_{j+1} - w_{j-1} + w_j \ln \left| \frac{z_{j+1} - z_j}{z_{j-1} - z_j} \right| + \sum_{\substack{m=1 \\ m \neq j, j+1}}^N I_m, \quad (3.4)$$

где интегралы I_m вычислены по формуле (3.3).

Подставив в равенство (3.4) известные действительные или мнимые части функции $w(z)$ при $j=1, 2, \dots, N$, получим систему $N \times N$ линейных алгебраических уравнений

$$BX=F, \quad (3.5)$$

здесь B - полнозаполненная матрица, X - вектор неизвестных для определения $\operatorname{Re} w$ на C_2 и $\operatorname{Im} w$ на C_1 , F - вектор правой части.

Для решения системы (3.5) используется метод Гаусса с выбором ведущего элемента [69, 84].

3.2. Модифицированный КМГЭ

В ходе решения задачи о циркуляционном обтекании тел на границах области известны касательные и нормальные компоненты вектора скорости V_s и V_n . Поэтому интегральную формулу Коши с помощью предельных формул Сохоцкого для аналитической в области D функции комплексно-сопряженной скорости $W(z)$ можно записать в следующем виде:

$$\begin{aligned} W(z_0) &= (V_n(z_0) - iV_s(z_0)) e^{-i\alpha(z_0)} = \\ &= \frac{1}{\varepsilon(z_0)i} \int_C \frac{(V_n(z) - iV_s(z)) e^{-i\alpha(z)}}{z - z_0} dz, \end{aligned} \quad (3.6)$$

где $\varepsilon(z_0) = 2\pi$ для внутренней точки, $\varepsilon(z_0) = \pi$ для точки на гладкой границе C , $\varepsilon(z_0) = \theta$ для угловой точки границы C (θ - угол при вершине). Положительное направление обхода контура C берется таким образом, чтобы область D оставалась слева, (внешняя граница обходится против часовой стрелки, внутренняя - по часовой). Зная V_s и V_n , можно найти V_x и V_y по формулам:

$$V_x = V_n \cos \alpha - V_s \sin \alpha, \quad V_y = V_s \cos \alpha + V_n \sin \alpha. \quad (3.7)$$

Поскольку в ходе итерационного процесса на свободной границе известна касательная составляющая скорости, а на дне, боковых стенках и профиле - нормальная составляющая, то для функции $W(z)$ имеем смешанную краевую задачу.

Разобьем границу области на N линейных элементов Γ_j с узлами z_j . Воспользовавшись уравнением (3.4), записанным для функции комплексно-сопряженной скорости $W(z)$, расписанной по

касательным V_s и нормальным V_n составляющим вектора скорости, получим:

$$\varepsilon i \left(V_{n_j} - iV_{s_j} \right) e^{-i\alpha_j} = \left(V_{n_{j+1}} - iV_{s_{j+1}} \right) e^{-i\alpha_{j+1}} - \left(V_{n_{j-1}} - iV_{s_{j-1}} \right) e^{-i\alpha_{j-1}} + \\ + \left(V_{n_j} - iV_{s_j} \right) e^{-i\alpha_j} \ln \left| \frac{z_{j+1} - z_j}{z_{j-1} - z_j} \right| + \sum_{\substack{m=1 \\ m \neq j, j+1}}^N I_m, \quad (3.8)$$

где

$$I_m = \left(V_{n_{m+1}} - iV_{s_{m+1}} \right) e^{-i\alpha_{m+1}} - \left(V_{n_m} - iV_{s_m} \right) e^{-i\alpha_m} + \\ + \left[\frac{(z_j - z_m)(V_{n_{m+1}} - iV_{s_{m+1}}) e^{-i\alpha_{m+1}}}{z_{m+1} - z_m} - \frac{(z_j - z_{m+1})(V_{n_m} - iV_{s_m}) e^{-i\alpha_m}}{z_{m+1} - z_m} \right] \ln \left(\frac{z_{m+1} - z_j}{z_m - z_j} \right), \\ e^{-i\alpha_j} = -i \left(\frac{z_{j+1} - z_j}{|z_{j+1} - z_j|^2} + \frac{z_j - z_{j-1}}{|z_j - z_{j-1}|^2} \right) \sqrt{\left| \frac{z_{j+1} - z_j}{|z_{j+1} - z_j|^2} + \frac{z_j - z_{j-1}}{|z_j - z_{j-1}|^2} \right|}.$$

Записывая уравнение (3.8) для каждого узла границы и разделяя мнимые и действительные части, получим $AX+iBX=0$, где A и B - полнозаполненные матрицы $N \times 2N$ (N - строк, $2N$ - столбцов), вектор $X=X(V_{s_1}, V_{n_1}, V_{s_2}, V_{n_2}, \dots, V_{s_N}, V_{n_N})$. Далее, следуя [32], получаем следующую систему уравнений

$$GY=F, \quad (3.9)$$

где матрица G и вектор правой части получаются следующим образом: если в узле z_j задана касательная составляющая скорости V_{s_j} , то берется j строка матрицы B и после выборки элементов строки, соответствующих неизвестным значениям V_s или V_n во всех остальных узлах, получается j строка матрицы C , j элемент вектора Y будет V_{n_j} , j элемент вектора F - сумма известных значений V_s

или V_n , умноженных на соответствующие элементы матрицы B ; если в узле z_j задана нормальная составляющая скорости V_{n_j} , то для построения j строки системы (3.9) используется матрица A .

Случай безграничной области

Интегральная формула Коши справедлива и для безграничной области D , если функция $W(z)$ обращается в нуль на бесконечности. При обтекании профиля безграничным потоком жидкости комплексный потенциал на бесконечности имеет следующее разложение:

$$w(z) = \phi(x, y) + i\psi(x, y) = K + V_\infty e^{-i\beta} z + \frac{\Gamma}{2\pi i} \ln z,$$

где K - произвольная комплексная постоянная, Γ - циркуляция скорости вдоль контура профиля, β - угол атаки профиля, V_∞ - модуль вектора скорости потока на бесконечности.

Комплексно-сопряженная скорость потока на бесконечности

$$W(z) = V_x(x, y) - iV_y(x, y) = V_\infty e^{-i\beta}. \quad (3.10)$$

Уравнение (3.6) с учетом (3.10) примет вид:

$$W(z_0) = V_\infty e^{-i\beta} + \frac{1}{\varepsilon(z_0)i} \int_C \frac{(V_n(z) - iV_s(z))e^{-i\alpha(z)}}{z - z_0} dz.$$

В данном случае контур обходится по часовой стрелке. При построении системы уравнений (3.9) следует добавить слагаемое в известный правый вектор F (мнимую или действительную часть выражения $V_\infty e^{-i\beta}$ к каждому элементу вектора в зависимости от заданного граничного условия в j узле).

3.3. МГЭ на основе третьей формулы Грина

Плоский случай

Поскольку потенциал течения идеальной несжимаемой жидкости является гармонической функцией, можно воспользоваться из-

вестной из теории гармонических функций для уравнения Лапласа функцией Грина, которая для двумерных задач имеет вид:

$$G(\zeta, x) = -\frac{1}{2\pi} \ln(r),$$

где $\zeta = (x_i, y_i)$ - точка наблюдателя, $x = (x, y)$ – текущая точка границы, r – расстояние между точками ζ и x :
 $r(\zeta, x) = [(x_i - x)^2 + (y_i - y)^2]^{1/2}$.

Используя третью формулу Грина, можно записать следующее интегральное уравнение:

$$\alpha(\zeta)\phi(\zeta) + \int_{\Gamma} \phi(x) \frac{\partial G(\zeta, x)}{\partial n(x)} d\Gamma(x) = \int_{\Gamma} \frac{\partial \phi(x)}{\partial n(x)} G(\zeta, x) d\Gamma(x), \quad (3.11)$$

где $\Gamma(x) = \Gamma_1 \cup \Gamma_2$ - граница области D , $\phi(x)$ - гармоническая функция, $\vec{n}(x)$ - внешняя по отношению к области D единичная нормаль к поверхности $\Gamma(x)$. Параметр $\varepsilon(\zeta)$ определяется следующим образом: $\varepsilon(\zeta) = 2\pi$ для внутренней точки, $\varepsilon(\zeta) = \pi$ для точки на гладкой границе, $\varepsilon(\zeta) = \beta$ для угловой точки границы (β - угол при вершине).

Как видно из (3.11) гармоническая функция ϕ определяется в произвольной точке области по значениям самой функции и ее нормальной производной ϕ_n , известным на границе.

Обычно для краевых задач на границе Γ задается либо функция ϕ (задача Дирихле), либо функция $\partial \phi / \partial n$ (задача Неймана), либо на одной части границы (Γ_1) - функция ϕ , а на другой (Γ_2) - функция $\partial \phi / \partial n$ (смешанная краевая задача). Во всех случаях на границе области имеет место граничное интегральное уравнение (3.11), решение которого в аналитическом виде может быть получено только для очень узкого круга задач. Поэтому для нахождения решения применяются численные методы.

Численное решение интегрального соотношения (3.11) можно получить, разбив границу Γ на N линейных элементов Γ_j узлами

$(x_j, y_j) (j = \overline{1, N})$ так, чтобы j -му элементу принадлежали узлы (x_j, y_j) и (x_{j+1}, y_{j+1}) .

Введем глобальную базисную функцию

$$\Lambda_j(x, y) = \begin{cases} N_2^{j-1}(x, y), & (x, y) \in \Gamma_{j-1}, \\ N_1^j(x, y), & (x, y) \in \Gamma_j, \\ 1, & (x, y) = (x_j, y_j), \\ 0, & (x, y) \notin \Gamma_{j-1} \cup \Gamma_j, \end{cases}$$

где $N_j^k, (k = 1, 2)$ - локальные интерполирующие функции на элементе Γ_j от однородной координаты ξ такой, что $\xi = (2s/L_j - 1) \in [-1, 1]$, (L_j - длина j -го элемента, $s \in (0, L_j)$ - текущая координата). Координата s может быть представлена в виде:

$$s = \sqrt{(x - x_{j+1})^2 + (y - y_{j+1})^2}, \quad (x, y) \in \Gamma_j.$$

На j -м элементе локальные интерполирующие функции представляются как:

$$N_1^j = \frac{1 - \xi}{2}; \quad N_2^j = \frac{1 + \xi}{2}. \quad (3.12)$$

Введем, как и в п.3.1, пробную функцию

$$R(x, y) = \sum_{j=1}^N f_j \Lambda_j(x, y), \quad (3.13)$$

где $f_j = f(x_j, y_j)$ - значение непрерывной функции $f(x, y)$ в j -ом узле.

Очевидно, что $f(x, y) = \lim_{\max|\Gamma_j| \rightarrow 0} R(x, y).$

Интегральное уравнение (3.11) после подстановки в него функций (3.13) может быть приведено к следующему виду:

$$\varepsilon\phi_i + \sum_{\substack{j=1 \\ i \neq j}}^N H_{ij}\phi_j + H_{ii}\phi_i = \sum_{\substack{j=1 \\ i \neq j}}^N G_{ij}q_j + G_{ii}q_i, \quad (3.14)$$

где несингулярные интегралы имеют вид:

$$H_{ij} = \int_{P_j} \Lambda_j q_i^*(s) ds; \quad G_{ij} = \int_{P_j} \Lambda_j \phi_i^*(s) ds,$$

здесь $P_j = \Gamma_{j-1} \cup \Gamma_j$ - носитель глобальной базисной функции Λ_j .
Сингулярные интегралы

$$H_{ii} \equiv 0; \quad G_{ii} = \frac{1}{4\pi} \left[L_{i-1} \left(\frac{3}{2} - \ln L_{i-1} \right) + L_i \left(\frac{3}{2} - \ln L_i \right) \right],$$

где L_j - длина j -го элемента.

Приведение интегралов

Входящие в (3.14) интегралы могут быть приведены к виду, удобному для дальнейших вычислений. Имеем:

$$\begin{aligned} H_{ij} &= \int_{P_j} \Lambda_j q_i^* ds = \int_{\Gamma_{j-1}} N_2^{j-1} q_{ij-1}^*(s) ds + \int_{\Gamma_j} N_1^j q_{ij}^*(s) ds = \\ &= -\frac{D_{j-1}}{16\pi} \int_{-1}^1 \frac{(1+\xi)}{r_{ij-1}^2} d\xi - \frac{D_j}{16\pi} \int_{-1}^1 \frac{(1-\xi)}{r_{ij}^2} d\xi = I_1 + I_2, \end{aligned}$$

где

$$I_1 = -\frac{D_{j-1}}{16\pi} \int_{-1}^1 \frac{d\xi}{r_{ij-1}^2} - \frac{D_j}{16\pi} \int_{-1}^1 \frac{d\xi}{r_{ij}^2}; \quad I_2 = -\frac{D_{j-1}}{16\pi} \int_{-1}^1 \frac{\xi d\xi}{r_{ij-1}^2} + \frac{D_j}{16\pi} \int_{-1}^1 \frac{\xi d\xi}{r_{ij}^2},$$

$$D_j = (x_j + x_{j+1} - 2x_i)(y_{j+1} - y_j) - (y_j + y_{j+1} - 2y_i)(x_{j+1} - x_j),$$

$$r_{ij}^2 = \frac{1}{4} \left[\left(x_j + x_{j+1} - 2x_i + (x_{j+1} - x_j)\xi \right)^2 + \left(y_j + y_{j+1} - 2y_i + (y_{j+1} - y_j)\xi \right)^2 \right]$$

Аналогично преобразуются интегралы G_{ij} .

$$\begin{aligned} G_{ij} &= \int_{P_j} \Lambda_j \phi_{ij}^*(s) ds = \int_{\Gamma_{j-1}} N_2^{j-1} \phi_{ij-1}^*(s) ds + \int_{\Gamma_j} N_1^j \phi_{ij}^*(s) ds = \\ &= -\frac{L_{j-1}}{16\pi} \int_{-1}^1 (1+\xi) \ln r_{ij-1}^2 d\xi - \frac{L_j}{16\pi} \int_{-1}^1 (1-\xi) \ln r_{ij}^2 d\xi = I_3 + I_4, \end{aligned}$$

где

$$I_3 = -\frac{L_{j-1}}{16\pi} \int_{-1}^1 \ln r_{ij-1}^2 d\xi - \frac{L_j}{16\pi} \int_{-1}^1 \ln r_{ij}^2 d\xi;$$

$$I_4 = -\frac{L_{j-1}}{16\pi} \int_{-1}^1 \xi \ln r_{ij-1}^2 d\xi + \frac{L_j}{16\pi} \int_{-1}^1 \xi \ln r_{ij}^2 d\xi.$$

Аналитическое вычисление интегралов

Интегралы I_k , ($k = \overline{1, 4}$) состоят из двух слагаемых $I_k = I_k^{j-1} + I_k^j$ и отличаются друг от друга только индексами. Поэтому проведем дальнейшие вычисления интегралов только для j -го индекса:

$$I_1^j = \int_{-1}^1 \frac{d\xi}{r^2}; \quad I_2^j = \int_{-1}^1 \frac{\xi d\xi}{r^2}; \quad I_3^j = \int_{-1}^1 \ln r^2 d\xi; \quad I_4^j = \int_{-1}^1 \xi \ln r^2 d\xi.$$

Введем дополнительные обозначения:

$$L = L_j; \quad r^2 = \frac{1}{4}(A\xi^2 + 2B\xi + C); \quad A = L^2;$$

$$B = (x_{j+1} - x_j)(x_{j+1} + x_j - 2x_i) + (y_{j+1} - y_j)(y_{j+1} + y_j - 2y_i);$$

$$C = (x_{j+1} + x_j - 2x_i)^2 + (y_{j+1} + y_j - 2y_i)^2; D = D_j = B^2 - AC .$$

Геометрическая интерпретация величин A, B, C, D

Рассмотрим элемент Γ_j границы области C и расположим точку наблюдения (x_i, y_i) как показано на рис.6.

Рис. 6. Пояснительная схема к аналитическому вычислению интегралов

Введем вектора

$$\vec{P} = \left\{ \left(\frac{x_{j+1} + x_j}{2} - x_i \right), \left(\frac{y_{j+1} + y_j}{2} - y_i \right) \right\}, \vec{F} = \left\{ (x_{j+1} - x_j), (y_{j+1} - y_j) \right\} .$$

Тогда коэффициенты A, B, C можно записать через скалярные произведения

$$A = (\vec{F} \cdot \vec{F}); \quad B = 2(\vec{P} \cdot \vec{F}); \quad C = (\vec{P} \cdot \vec{P}),$$

отсюда

$$B^2 - AC = 4|\vec{P}|^2 |\vec{F}|^2 (\cos^2(\vec{P}, \vec{F}) - 1) \leq 0 . \quad (3.15)$$

Выражение (3.15) обращается в нуль, если вектора \vec{P} и \vec{F} коллинеарны. Это может быть только в случае, когда точка наблюдателя либо лежит на рассматриваемом элементе и совпадает с одним из узлов (в этом случае $D \equiv 0$, $A \equiv C = \pm B$, знак “+” или “-” зависит от того, с каким узлом совпадает точка наблюдателя), либо точка наблюдателя лежит на прямой, проходящей через элемент Γ_j .

Вычисление интегралов для случая $B^2 - AC < 0$

Проводя несложные выкладки, можем получить аналитические выражения для интегралов $I_k, (k = \overline{1,4})$

$$I_1 = \frac{4}{\sqrt{CA - B^2}} \left(\operatorname{arctg} \frac{B + A}{\sqrt{CA - B^2}} - \operatorname{arctg} \frac{B - A}{\sqrt{CA - B^2}} \right),$$

$$I_2 = \frac{2}{A} \ln \frac{M_1}{M_2} - \frac{B}{A} I_1,$$

$$I_3 = -4(\ln 2 + 1) + \ln(M_1 \cdot M_2) + \frac{B}{2} I_2 + \frac{C}{2} I_1,$$

$$I_4 = \frac{1}{2} \left(1 - \frac{C}{A} \right) \ln \left(\frac{M_1}{M_2} \right) + \frac{2B}{A} + \left(\frac{AC - B^2}{2A} \right) I_2,$$

где $M_1 = A + 2B + C; M_2 = A - 2B + C.$

Вычисление интегралов для случая $B^2 - AC = 0$, но $A \neq C$

В этом случае интегралы примут вид:

$$I_1 = \frac{2A}{B^2 - A^2}, \quad I_2 = -\frac{2B}{B^2 - A^2} + \frac{2}{A^2} \operatorname{arth} \left(\frac{A}{B} \right),$$

$$I_3 = -4 + \ln(M_1 M_2) + \frac{4B}{A} \operatorname{arth} \left(\frac{A}{B} \right), \quad I_4 = \frac{1}{2} \ln \left(\frac{M_1}{M_2} \right) + \frac{2B}{A} - \frac{2C}{A} \operatorname{arth} \left(\frac{A}{B} \right).$$

Численное интегрирование

Интегралы $I_k, (k = \overline{1,4})$ могут быть найдены численно с помощью стандартных квадратурных формул Гаусса [46].

$$\int_{-1}^1 \frac{d\xi}{r^2} = \sum_{i=1}^n \tilde{A}_i / r^2(\xi_i), \quad \int_{-1}^1 \frac{\xi d\xi}{r^2} = \sum_{i=1}^n \tilde{A}_i \xi_i / r^2(\xi_i),$$

$$\int_{-1}^1 \ln r^2 d\xi = \sum_{i=1}^n \tilde{A}_i \ln r^2(\xi_i), \quad \int_{-1}^1 \xi \ln r^2 d\xi = \sum_{i=1}^n \tilde{A}_i \xi_i \ln r^2(\xi_i),$$

где \tilde{A}_i, ξ_i - веса и абсциссы квадратурной формулы Гаусса.

Квадратурные формулы Гаусса

Формулы Гаусса являются наиболее распространенными формулами численного интегрирования, как формулы, имеющие наивысший порядок точности при заданном числе узлов в области интегрирования.

Таблица 1
Веса и абсциссы квадратурной формулы Гаусса

N	Веса A_i	Абсциссы ξ_i
2	1.0000000000000000	0.577350269189626
	1.0000000000000000	-0.577350269189626
3	0.888888888888889	0.000000000000000
	0.555555555555556	0.774596669241483
	0.555555555555556	-0.774596669241483
4	0.652145154862546	0.339981043584856
	0.652145154862546	-0.339981043584856
	0.347854845137454	0.861136311594053
	0.347854845137454	-0.861136311594053
5	0.568888888888889	0.000000000000000
	0.478628670499366	0.538469310105683
	0.478628670499366	-0.538469310105683
	0.236926885056189	0.906179845938664
	0.236926885056189	-0.906179845938664
7	0.417959183673469	0.000000000000000
	0.381830050505119	0.405845151377397
	0.381830050505119	-0.405845151377397
	0.279705391489277	0.741531185599394
	0.279705391489277	-0.741531185599394
	0.129484966168870	0.949107912342759
	0.129484966168870	-0.949107912342759

В одномерном случае погрешность ε квадратуры Гаусса для вычисления интеграла

$$J = \int_{-1}^1 f(\xi) d\xi \approx \sum_{i=1}^n A_i f(\xi_i) + \varepsilon,$$

имеет оценку

$$|\varepsilon| \leq \frac{2^{2n+1}}{2n+1} \frac{(n!)^4}{((2n)!)^3} \max_{\xi \in [-1,1]} \left| \frac{d^{2n} f}{d\xi^{2n}} \right|.$$

Квадратурные формулы Гаусса точны для многочленов степени $2n-1$. Значения весов A_i и абсцисс ξ_i приведены в таблице 1.

Осесимметричный случай

При осесимметричном потенциальном течении в цилиндрических координатах (r, z, θ) функции ϕ и $\frac{\partial \phi}{\partial n}$ не зависят от угловой координаты θ . Поэтому интегральное уравнение (3.11) может быть преобразовано к виду:

$$\begin{aligned} \alpha(\zeta)\phi(\zeta) + \int_{\Gamma} \phi(x) \frac{\partial}{\partial n} G'(\zeta, x) r(x) ds(x) = \\ = \int_{\Gamma} \frac{\partial \phi(x)}{\partial n} G'(\zeta, x) r(x) ds(x). \end{aligned} \quad (3.16)$$

Фундаментальное решение в цилиндрических координатах имеет вид:

$$G'(\zeta, x) = \frac{1}{4\pi} \int_0^{2\pi} \frac{d\theta}{\sqrt{a - b \cos \theta}},$$

где $a = r^2(\zeta) + r^2(x) + [z(\zeta) - z(x)]^2$; $b = 2r(\zeta)r(x)$; $r(x), z(x)$ и $r(\zeta), z(\zeta)$ - координаты фиксированной и переменной точек в меридиональной плоскости; α - контур пересечения тела вращения с меридиональной плоскостью, \vec{n} - внешняя нормаль по отношению к области интегрирования.

Интеграл для функции $G'(\zeta, x)$ сводится к полному эллиптическому интегралу первого рода. Для этого следует интервал интегрирования разбить на два подинтервала $(0, \pi)$ и $(\pi, 2\pi)$ и перейти к новым переменным $\theta = \pi - 2\beta$ и $\theta = \pi + 2\beta$. В результате получаем:

$$G'(\zeta, x) = K(m)/\pi(a+b)^{1/2}, \quad (3.17)$$

$$\frac{\partial G'(\zeta, x)}{\partial n} = \frac{\partial G'}{\partial r} n_r + \frac{\partial G'}{\partial z} n_z = \frac{n_r [E(m) - K(m)]}{2\pi r(x)\sqrt{a+b}} -$$

$$-\frac{n_r [r(\zeta) - r(x)] + n_z [z(\zeta) - z(x)]}{\pi(a-b)\sqrt{a+b}} E(m),$$

$$m^2 = \frac{4r(\zeta)r(x)}{[r(\zeta) + r(x)]^2 + [z(\zeta) - z(x)]^2}, \quad 0 \leq m \leq 1.$$

Здесь n_r , n_z - компоненты нормального вектора в направлении осей r и z , соответственно; $K(m)$ и $E(m)$ - полные эллиптические интегралы, для вычисления которых можно использовать их полиномиальные представления из работы [1]:

$$E(m) = 1 + \sum_{i=1}^n [c_i m_1^i - d_i m_1^i \ln(m_1)] + \varepsilon(m);$$

$$K(m) = \sum_{i=0}^n [a_i m_1^i - b_i m_1^i \ln(m_1)] + \varepsilon(m); \quad (3.18)$$

$$m_1 = 1 - m^2 = \frac{[r(x) - r(\zeta)]^2 + [z(x) - z(\zeta)]^2}{[r(x) + r(\zeta)]^2 + [z(x) - z(\zeta)]^2}.$$

При $n = 4$ остаточный член удовлетворяет оценке $\varepsilon(m) \leq 2 \cdot 10^{-8}$, а a_i , b_i , c_i и d_i - константы, представленные в таблице 2.

Необходимо заметить, что в осесимметричном случае функция Грина зависит не только от расстояния между двумя точками ζ и r , но и от расстояния между рассматриваемой точкой и осью симметрии.

Таблица 2
Коэффициенты в разложении эллиптических интегралов

$a_0 = 1.38629436112$	$b_0 = 0.5$	$c_0 = 0.0$	$d_0 = 0.0$
$a_1 = 0.09666344259$	$b_1 = 0.12498593597$	$c_1 = 0.44325141463$	$d_1 = 0.24998368310$
$a_2 = 0.03590092383$	$b_2 = 0.06880248576$	$c_2 = 0.06260601220$	$d_2 = 0.09200180037$
$a_3 = 0.03742563713$	$b_3 = 0.03328355346$	$c_3 = 0.04757383546$	$d_3 = 0.04069697526$

$a_4=0.01451196212$	$b_4=0.00441787012$	$c_4=0.01736506451$	$d_4=0.00526449639$
---------------------	---------------------	---------------------	---------------------

Таблица 3
Веса и абсциссы пятиточечной квадратуры Гаусса с логарифмическим весом

Веса A_i	Абсциссы ξ_i
0.29789 34717 82894	0.02913 44721 51972
0.34977 62265 13242	0.17397 72133 20898
0.23448 82900 44052	0.41170 25202 84902
0.09893 04595 16633	0.67731 41745 82820
0.01891 15521 43196	0.89477 13610 31008

Вычисление сингулярных интегралов

Из выражений (3.18) видно, что все диагональные элементы матриц h_{ii} и G_{ii} будут иметь логарифмическую особенность. Для того, чтобы применить формулы численного интегрирования к построению диагональных элементов, необходимо сингулярный интеграл разбить на две части - сингулярную и несингулярную. Для сингулярной части можно применить специальную квадратурную формулу Гаусса с логарифмическими весами:

$$J = \int_0^1 f(\xi) \ln(1/\xi) d\xi \approx \sum_{i=1}^n A_i f(\xi_i) + K \cdot \frac{f^{(2n)}(\xi)}{(2n)!}. \quad (3.19)$$

Формула (3.19) точна, если функция $f(\xi)$ - многочлен степени не выше $(2n-1)$. Веса и абсциссы пятиточечной квадратуры (3.19) приведены в таблице 3. Для этого случая $K=5 \cdot 10^{-6}$.

3.4. Методы решения систем линейных алгебраических уравнений

Решение краевых задач численными методами сопряжено с решением больших систем линейных алгебраических уравнений. Все методы решения линейных систем можно разделить на две группы: прямые (Гаусса, Краута, Холецкого и т.д.) и итерационные (Гаусса-Зейделя, минимальных невязок, нелинейные регуляризующие и т.д.). Выбор метода для решения конкретной системы зависит от ее свойств. В методе граничных элементов матрицы систем уравнений получаются полностью заполненными. В случае задачи Неймана исходный эллиптический дифференциальный оператор

сводится к интегральному уравнению Фредгольма II рода. Полученная для него в результате применения метода граничных элементов матрица системы линейных алгебраических уравнений обладает “хорошими” свойствами за счет диагонального преобладания и может быть решена прямым или итерационным методом. Если требуется решить задачу Дирихле, то в результате получается граничное интегральное уравнение Фредгольма I рода, которое является некорректным по Адамару, что, вообще говоря, приводит к “плохо” обусловленной системе линейных алгебраических уравнений [80]. Поэтому для численного решения некорректных задач требуется применение методов, основанных на различных способах регуляризации [80]. В пакете “*AKORD*” используется нелинейный регуляризирующий алгоритм, предложенный в работе [82].

Эффект саморегуляризации

С другой стороны, если линейное интегральное уравнение Фредгольма I рода имеет логарифмическую особенность в ядре, то есть возможность решить его напрямую сведением к системе линейных алгебраических уравнений, не применяя какого-либо регуляризующего алгоритма. Квадратурные формулы Гаусса и кубатурные формулы Радо (пространственный случай) для решения уравнений Фредгольма I рода с ядрами типа функций Грина порождают саморегуляризирующие алгоритмы, в которых параметром регуляризации является шаг квадратурной формулы [27, 28].

Число обусловленности

Важным критерием близости матрицы A к вырожденной является число обусловленности $\text{cond}(A) = |A| \|A^{-1}\|$, которое играет роль множителя в увеличении относительной ошибки. Для определения числа обусловленности применяется алгоритм, изложенный в монографии [84]. Из этой же работы в ППП «*AKORD*» включены программы DECOMP и SOLVE, осуществляющие декомпозицию матрицы A и ее решение прямым методом Гаусса с выбором ведущего элемента [69].

Метод Гаусса – Зейделя

Среди итерационных методов, применяемых в МГЭ для решения систем линейных алгебраических уравнений, популярным является метод Гаусса – Зейделя с релаксацией.

Пусть задана система линейных алгебраических уравнений

$$\sum_{k=1}^n a_{ik} x_k = b_i, \quad i = 1, 2, \dots, n.$$

Рекуррентное соотношение вида

$$x_i^* = -\frac{1}{a_{ii}} \left(\sum_{k=1}^i a_{ik} x_k^* + \sum_{k=i}^n a_{ik} x_k^j - b_i \right),$$

$$x_i^{j+1} = x_i^j + \alpha(x_i^* - x_i^j)$$

определяет метод Гаусса – Зейделя с релаксацией, где α - параметр релаксации ($0 < \alpha \leq 1$). Метод релаксации является обобщением итерационного метода Гаусса-Зейделя и при $\alpha = 1$ точно с ним совпадает.

Вычислительные затраты

При реализации численных методов возникает естественный вопрос о точности метода и о вычислительных затратах. Решая плоские и осесимметричные задачи с помощью КМГЭ и МГЭ, получают полнозаполненные матрицы систем линейных алгебраических уравнений. Число этих уравнений, как правило, менее 1000, а максимальное число обусловленности в самых худших случаях было порядка 10^3 . Решение таких систем осуществлялось методом Гаусса с выбором ведущего элемента. На решение системы в данном случае требуется N^3 операций.

Таблица 4

Параметры релаксации α , на пространственных сетках, близкие к оптимальным

Способ разбиения	Количество узлов на сфере	Параметр релаксации в безграничной жидкости /число итераций	Параметр релаксации при наличии твердой стенки /число итераций
На 4 зоны	402	0.57 / 28	0.50 / 36
	578	0.55 / 30	0.48 / 38
На 6 зон	386	0.59 / 28	0.52 / 37
	866	0.54 / 33	0.47 / 43
Икосаэдр	162	0.35 / 50	0.27 / 68
	642	0.21 / 90	0.16 / 127

В случае решения пространственных задач количество уравнений системы значительно увеличивается. На одну итерацию в итерационных методах требуется n^2 операций и, когда итерационный

процесс сходится меньше чем за n шагов, привлекательность итерационных методов перед прямыми налицо. При решении СЛАУ методом Гаусса – Зейделя с релаксацией необходимо задать оптимальное значение параметра α_{opt} , при котором удается получить наилучшую скорость сходимости. К сожалению, аналитическая оценка этого параметра имеется для очень узкого класса задач. Экспериментально найденный параметр при малом числе уравнений зачастую дает серьезное ускорение и при решении больших систем. В таблице 4 приведены значения параметра $\alpha \approx \alpha_{opt}$, близкие к оптимальным, для трехмерных треугольных сеток, построенных на сфере двумя способами, описанными ниже в разделе 3.6. . Это методы, основанные на зонном разбиении поверхностей и на «дроблении» икосаэдра. Значения параметра релаксации приведены для задачи обтекания сферы безграничным потоком жидкости и для задачи обтекания сферы вблизи твердой стенки.

3.5. Алгоритмы построения свободных границ

Стационарная задача

Итерационная схема

Пусть на k -ом итерационном шаге известна форма свободной границы C_1 (рис. 1). Для того, чтобы продолжить итерационный процесс, необходимо определить значение потенциала ϕ на границе C_1 . Поскольку скорость на границе C_1 направлена по касательной к контуру, то $\partial\phi/\partial s = q$ или $(dW/dz = q)$.

Так как потенциал скорости определяется с точностью до аддитивной константы, полагаем $\phi_1 = 0$. Далее для любой точки свободной границы имеем

$$\phi_{i+1} = \phi_i + \frac{q_i + q_{i+1}}{2} \Delta s_i, \quad (3.20)$$

где $i = \overline{1, N_g - 1}$ - номера узлов точек свободной границы, $q_i = q(y_i)$ определяется формулой (2.19), где $\Delta s_i = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$ -

длина i -го элемента свободной границы, N_g - количество узлов на свободной границе C_1 .

Определение формы свободной границы

Алгоритм нахождения свободной границы осуществляется по следующей схеме:

- 1) задается некоторое положение свободной границы C_1^k ;
- 2) по формуле (3.20) определяются значения потенциала ϕ_i в узлах C_1^k ;
- 3) решается смешанная краевая задача с заданными значениями потенциала ϕ_i^k на свободной границе C_1^k ;
- 4) в точках свободной границы C_1^k определяются значения компонент вектора скорости u_i и v_i ;
- 5) из условия коллинеарности вектора скорости и касательной к границе ($dy/dx = v/u$) вычисляется новое положение свободной границы C_1^{k+1}

$$y_{i+1}^{k+1} = y_i^{k+1} + \Delta y_i^k.$$

Цикл повторяется до выполнения требуемой точности

$$\max_i |y_i^{k+1} - y_i^k| < \varepsilon. \quad (3.21)$$

Нахождение приращения Δy

Существенным моментом построения свободной границы является то, что абсциссы x точек свободной границы C_1 не меняют своих значений в течение всего итерационного процесса, изменяется только ордината y_i . Будем аппроксимировать свободную границу C_1 (на k -ой итерации) кусочным многочленом второй степени

$$y = Ax^2 + Bx + C.$$

Неизвестные коэффициенты A и B на элементе могут быть найдены по значению производной в двух узловых точках свободной границы

$$\frac{dy}{dx} = 2Ax + B. \quad (3.22)$$

Записывая правую часть этого уравнения с учетом кинематического условия в узлах границы C_1 и разрешая полученную систему относительно A и B , приходим к дифференциальному уравнению:

$$\frac{dy}{dx} = \left(\frac{v_i - v_{i+1}}{u_i - u_{i+1}} \right) \frac{x}{x_i - x_{i+1}} - \left(\frac{x_i v_{i+1} - x_{i+1} v_i}{u_{i+1} - u_i} \right) \frac{1}{x_i - x_{i+1}}. \quad (3.23)$$

Интегрируя последнее равенство в интервале $[x_i, x_{i+1}]$, получаем новое значение ординаты на $(k+1)$ -й итерации:

$$y_{i+1}^{(k+1)} = y_i^{(k+1)} + \Delta y_i^{(k)}, \quad \text{где} \quad \Delta y_i^{(k)} = \left(\frac{v_{i+1} - v_i}{u_{i+1} - u_i} \right) \frac{x_{i+1} - x_i}{2}.$$

Другим способом вычисления приращения Δy_i^k , дающим более точные результаты, служит способ разложения в ряд Тейлора:

$$\Delta y_i^k = \frac{v_i}{u_i} (x_{i+1} - x_i) + \frac{1}{2!} \frac{d}{dx} \left(\frac{v_i}{u_i} \right) (x_{i+1} - x_i)^2 + \dots + \frac{1}{4!} \frac{d^3}{dx^3} \left(\frac{v_i}{u_i} \right) (x_{i+1} - x_i)^4 \\ \cdot$$

Данный итерационный алгоритм был развит в работах [18, 78].

Если речь идет о построении решений на основе функции тока или комплексного потенциала, то итерационная схема поиска свободной границы выглядит несколько иначе:

$$y_i^{k+1} = y_i^k + \Delta y_i^k,$$

где $\Delta y_i^k = 1 - \psi_i^k$, а ψ_i^k - это текущее значение функции тока в i -точке свободной границы.

Приведенная к безразмерному виду краевая задача зависит от безразмерного параметра – числа Фруда Fr . Впервые о не единст-

венности возможных форм течений тяжелой жидкости при числах Фруда, близких к единице, было доказано в работе [61]. В работе [66] с помощью вариационного принципа доказано, что для бесконечного множества значений чисел Фруда задача имеет, по крайней мере, два различных решения. Следовательно, для получения однозначного решения использовать число Фруда в качестве параметра задачи нецелесообразно. В работе [35] отмечается, что данная задача имеет единственное решение, если в качестве определяющего течение параметра вместо числа Фруда задавать безразмерный параметр $V = V_0 / V_\infty$, характеризующий отношение скорости V_0 в вершине волны к скорости набегающего потока V_∞ . При этом число Фруда есть функция от V : $Fr=Fr(V)$. Данная зависимость легко получается из уравнений (2.19):

$$Fr^2 = 2 \frac{y_0 - 1}{1 - V^2}. \quad (3.24)$$

Следовательно, выражение для модуля скорости на свободной границе преобразуется к виду:

$$q = \sqrt{1 - (1 - V^2) \frac{y - 1}{y_0 - 1}}. \quad (3.25)$$

Подстановка в итерационные алгоритмы формулы (3.24) вместо уравнения (3.20) дает возможность получать два решения, нахождение которых осуществлялось различными способами: методом граничных элементов с использованием как функции тока, так и потенциала скоростей; методом комплексных граничных элементов, когда на свободной границе задается потенциал или функция тока.

После того, как итерационный процесс устанавливается, число Фруда вычисляется по формуле (3.24).

Очевидно, что параметр V изменяется в пределах $[0;1]$: 1 - соответствует бесконечному числу Фруда; 0 - числу Фруда, близкому к единице, при котором амплитуда построенной волны максимальна.

Алгоритм определения свободной границы в методе, использующем комплексно-сопряженную скорость аналогичен алгоритму, изложенному выше с той лишь разницей, что в точках свободной границы ищется не потенциал ϕ_i , а касательная составляющая вектора скорости v_s .

Нестационарная задача

Алгоритм движения по времени

Нестационарные краевые задачи, описанные выше, являются нелинейными в силу нелинейности динамических граничных условий и заранее неизвестного положения свободной границы.

В работе М.А. Лаврентьева и Б.В. Шабата [47] предложен метод решения нестационарных задач со свободными границами. Его идея состоит в редукции общей нелинейной задачи, определенной на непрерывном интервале времени, к последовательности линейных краевых задач теории потенциала на дискретном множестве временных шагов из заданного временного интервала. Эта методика хорошо зарекомендовала себя и применялась в работах [4, 8, 17, 43, 91, 92].

Описанные нестационарные краевые задачи отличаются от традиционных задач математической физики тем, что время явно входит только в граничные условия (например: (2.17) и (2.20)), представляющие собой обыкновенные дифференциальные уравнения первого порядка, для интегрирования которых используется явный метод Эйлера.

Пусть в некоторый момент времени t_0 задано положение свободной границы C_1^0 и распределение потенциала ϕ^0 на ней. Далее необходимо решить уравнение Лапласа с условием Дирихле на C_1^0 и условием Неймана на C_2 . Новое положение свободной границы и распределение потенциала на ней для момента времени $t_0 + \tau$ можно вычислить, используя условия (например, (2.17) и (2.20)), дискретный аналог которых расписывается по схеме Эйлера следующим образом:

$$\vec{x}^{k+1} = \vec{x}^k + (\nabla \phi)^k \tau,$$

$$\phi^{k+1} = \phi^k + \left(0,5 \left| (\nabla \phi)^k \right|^2 - (y^k - H) \right) \tau, \quad k = 0, 1, 2, \dots, \quad (3.26)$$

где $\vec{x}^k = (x^k, y^k)$, ϕ^k - значения функций на k -м шаге по времени.

Таким образом, для момента времени $t_0 + \tau$ опять получаем смешанную краевую задачу для уравнения Лапласа с граничными условиями Дирихле 3.26 и условиями Неймана, соответствующими решаемой задаче. Повторное ее решение и использование граничных условий позволяет определить положение свободной границы и распределение потенциала на ней для момента времени $t_0 + 2\tau$ и т.д.

Описанный алгоритм представляет собой явную схему Эйлера первого порядка точности, которая, как известно из теории численных методов, является условно устойчивой и требует наличия некоторых дополнительных условий, связывающих шаг по времени с шагами по пространству (аналог условия Куранта в газовой динамике [29]).

Выбор шага по времени подбирается автоматически из следующих соображений:

- 1) любая частица жидкости за временной шаг не может переместиться на расстояние больше заданного;
- 2) узлы любого элемента не могут изменять ориентацию относительно друг друга (исключается самопересечение границы области).

Более подробно алгоритмы движения и выбора шага по времени изложены в работах [14, 43].

3.6. Работа с графическими объектами

При проведении численного эксперимента немаловажное место отводится графическому представлению информации на всех стадиях исследования задачи. При подготовке результатов для расчета графический интерфейс дает неоспоримые преимущества при задании области решения задачи, определении параметров расчета и задании краевых условий. При непосредственном численном решении задачи графическое представление промежуточных результатов позволяет сразу же качественно оценить, например, процесс сходимости к известному решению. При последующем

анализе графически представленные данные дают в руки пользователю все необходимое для комплексной оценки результатов решения.

Следует отметить, что создание графического интерфейса пользователя или графическое представление результатов предъявляет определенные требования к аппаратной части компьютера и к алгоритмам построения графических объектов, используемых разработчиком при проектировании приложения. В различных компонентах пакета “*AKORD*” для оптимизации операций с отображаемыми на экране графическими объектами использованы специальные алгоритмы машинной графики, а также способы хранения и обработки графических примитивов в оперативной памяти компьютера.

Базовые графические объекты на плоскости

При разработке приложений, работающих с машинной графикой, важным является продуманная организация структур классов, представляющих как сами графические объекты, так и объекты, отвечающие за их хранение и отображение. Это позволяет существенным образом увеличить скорость геометрических построений и определяет четкую логическую иерархию структур, используемых приложением.

Для использования в приложениях, составляющих основу пакета прикладных программ “*AKORD*”, был разработан оригинальный набор компонентов, позволяющих легко оперировать графическими объектами (рис. 7). Основными элементами этих компонентов являются:

1. Графическое окно, которое выполняет следующие функции:
 - преобразование логических (экраных) координат в физические и обратно;
 - произвольное масштабирование изображения и увеличение выделенных областей окна (функция “зуммирования”);
 - автоматическая прорисовка осей координат с широкими функциями настройки;
 - реализация быстрой прорисовки графики через страницы видеопамяти.

2. Набор базовых графических объектов, на основе которых может быть реализован графический объект практически любой сложности. Сюда входят:

- *графический примитив* - графический объект нижнего уровня, который не может быть разбит на более простые объекты. В этом объекте определяются функции конструирования и прорисовки примитива, а также функция определения принадлежности заданной точки примитиву;
- *вершина* – объект, определяющий опорную точку для задания графического примитива. Количество опорных точек для задания объекта может быть жестко определено или варьироваться в зависимости от типа графического примитива. Например, примитивы типа Line и Circle имеют две вершины, а примитив типа Arc - три, у примитива типа PolyLine количество вершин заранее не определено;
- *границная точка* – объект, предназначенный для дискретизации графического примитива по заданному закону. По умолчанию граничные точки совпадают с вершинами;
- *сложный графический объект* - определяет составной объект, включающий в себя несколько примитивов. В нем реализованы функции создания, удаления, изменения, выделения примитива;
- *сервис графических объектов* - предназначен для общего управления графическими объектами. В нем помимо списка графических объектов содержится список всех вершин и список всех граничных точек, что позволяет упростить поиск и обработку необходимых объектов.

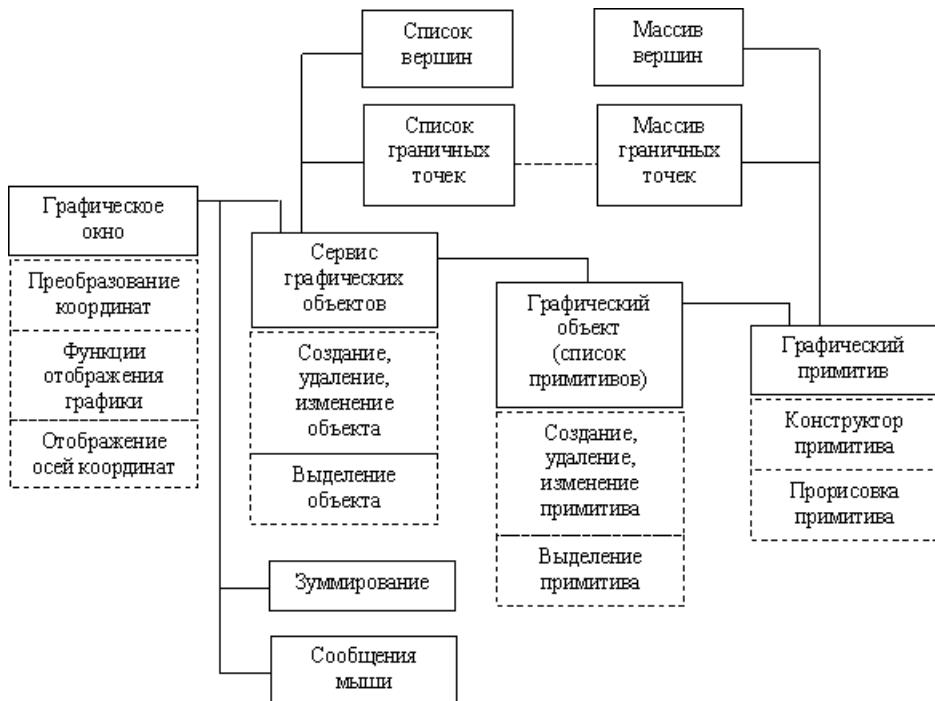


Рис. 7. Структура базовых графических объектов

Базовые пространственные графические объекты

Способы описания и работы с базовыми объектами

Для однозначного задания пространственного объекта введем некоторые понятия. Пространственный объект задается, в первую очередь, последовательностью координат своих вершин. Кроме того, к последовательности вершин добавляются последовательности ребер пространственного объекта и его элементов.

Под элементом понимается плоский или почти плоский (неизбежно выпуклый) треугольник или четырехугольник. Любой из четырехугольников может оказаться неплоским из-за погрешности вычислений, то есть, если вычислить уравнение плоскости по трем вершинам элемента, то четвертая при подстановке в это уравнение может дать величину отличную от нуля, но эта величина должна быть достаточно близка к нулю. Поскольку каждая вершина упоминается несколько раз, то целесообразно задавать ее не координатами, а порядковым номером в последовательности номеров вершин тела. Номера вершин в элементе перечислены в определенном порядке обхода (по часовой стрелке или против), который одинаков для всех элементов тела.

Ребро представляется парой точек из исходной последовательности точек, причем пары (m,l) и (l,m) (l и m — вершины из последовательности вершин) считаются одним и тем же ребром. При опи-

сании элементов их ребра, так же как и вершины, задаются порядковыми номерами в последовательности номеров ребер, при этом каждое ребро упоминается не менее двух раз. В элементе ребра перечисляются в произвольном порядке.

Таким образом, пространственный объект определяется последовательностью номеров его вершин и ребер. Все дальнейшие преобразования выполняются над этими последовательностями.

Преобразования в пространстве

Наилучшее восприятие формы пространственного объекта достигается при частом изменении его изображения в пространстве. Во многих случаях наглядное представление об объекте можно получить лишь путем выполнения операций вращения и переноса, а также путем построения его проекций.

Для осуществления таких преобразований удобно пользоваться однородными координатами [70]. Точки в трехмерном пространстве (x,y,z) представляются четырехмерным вектором $(x,y,z,1)$ или (X,Y,Z,H) . Представление n -мерного вектора $(n+1)$ -мерным называется однородным координатным воспроизведением [70]. При однородном координатном воспроизведении конечные результаты в n -мерном пространстве получаются с помощью обратного преобразования. То есть, трехмерный вектор (x,y,z) представляется четырехмерным (hx, hy, hz, h) . Разделив компоненты приведенного вектора на h , получим $x=hx/h$, $y=hy/h$, $z=hz/h$. Однородное координатное воспроизведение позволяет осуществлять полный набор преобразований в пространстве и, кроме того, дает возможность представлять бесконечно удаленные точки векторами, состоящими из конечных компонент, например $(1, 1, 1, 0)$. Необходимо отметить, что однородное координатное воспроизведение точек в трехмерном пространстве неединственно. Например, однородные координаты $(12, 16, 8, 4)$, $(6, 8, 4, 2)$ и $(3, 4, 2, 1)$ представляют одну и туже точку $(3, 4, 2)$.

Преобразование из однородных координат описывается соотношением

$$(X, Y, Z, H) = (x, y, z, I)T,$$

где T -матрица преобразования, которая может быть представлена в виде четырех отдельных частей

$$\begin{pmatrix} 3 \times 3 & 3 \times 1 \\ 1 \times 3 & 1 \times 1 \end{pmatrix}.$$

Подматрица 3×3 осуществляет линейное преобразование в виде изменения масштаба, отображения, сдвига и вращения. Под линейным преобразованием понимается такое преобразование, которое трансформирует исходную линейную комбинацию в некоторую такую же линейную комбинацию. За изменение масштаба отвечают диагональные элементы матрицы преобразования. Матрица изменения масштаба имеет вид:

$$\begin{pmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Помимо этого, диагональные элементы верхней левой подматрицы 3×3 общей матрицы преобразования 4×4 позволяют осуществлять зеркальное отображение. Для отображения без изменения масштаба необходимо, чтобы определитель преобразования был равен -1. Так, например, при отображении относительно координатной плоскости XY изменяется только знак координаты z. Следовательно, матрица преобразования для отображения относительно плоскости XY имеет вид:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Аналогично осуществляется отображение относительно других координатных плоскостей.

Недиагональные элементы подматрицы осуществляют сдвиг в трех измерениях, т.е.

$$(x \ y \ z \ 1) \begin{pmatrix} 1 & b & c & 0 \\ d & 1 & f & 0 \\ g & h & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = (x + dy + gz \ bx + y + hz \ cx + fy + z \ 1)$$

Общая матрица вращения имеет вид

$$\begin{pmatrix} n_1^2 + (1 - n_1^2) \cos \alpha & n_1 n_2 (1 - \cos \alpha) + n_3 \sin \alpha & n_1 n_3 (1 - \cos \alpha) - n_2 \sin \alpha \\ n_1 n_2 (1 - \cos \alpha) - n_3 \sin \alpha & n_2^2 + (1 - n_2^2) \cos \alpha & n_2 n_3 (1 - \cos \alpha) + n_1 \sin \alpha \\ n_1 n_3 (1 - \cos \alpha) + n_2 \sin \alpha & n_2 n_3 (1 - \cos \alpha) - n_1 \sin \alpha & n_3^2 + (1 - n_3^2) \cos \alpha \\ 0 & 0 & 0 \end{pmatrix}$$

,
где α - угол вращения, а направление оси вращения представлено единичным вектором $n = (n_1 \ n_2 \ n_3)$, (где n_1, n_2, n_3 - косинусы углов, образованных осью вращения n с осями координат x, y и z , соответственно).

Матрицы вращения около осей координат получаются как частные случаи приведенной матрицы вращения около произвольной оси. Нижняя строка 1x3 общей матрицы преобразований производит перенос. В общем случае матрица переноса имеет вид

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ l & m & n & 1 \end{pmatrix}.$$

Правый столбец 3x1 осуществляет преобразования в перспективе. Последний скалярный элемент 1x1 выполняет общее изменение масштаба. Полное преобразование, полученное путем воздействия на вектор матрицей положения 4x4 и нормализацией преобразованного вектора, называется билинейным преобразованием [70] и обеспечивает выполнение комплекса операций сдвига, час-

тичного изменения масштаба, вращения, отображения, переноса, а также изменения масштаба изображения в целом.

Получение проекций

Для получения проекции каждой точки $P(x, y, z)$, принадлежащей визуализируемому объекту, предстоит вычислить точки изображения $P'(x', y', z')$ на экране. Для этого необходимо преобразовать координаты точки P из так называемых мировых координат (x, y, z) в экранные координаты (x', y') ее центральной проекции. Для получения экранных координат необходимо осуществить видовое преобразование [2], при котором точка P остается на своем месте, но система мировых координат переходит в систему видовых координат.

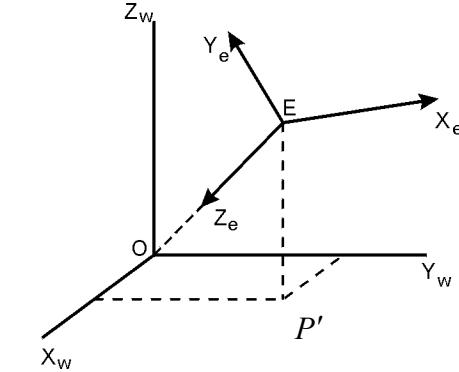


Рис. 8. Система видовых координат

Для выполнения видовых преобразований должны быть заданы точки наблюдения и объекта (рис. 8). Выберем правую систему координат. Кроме того, будем полагать, что экран находится между объектом и глазом E . Для точки P объекта прямая линия PE пересекает экран в точке P' . Удобно, чтобы начало координат находилось где-то вблизи центра объекта, так как объект наблюдается в направлении от E к O .

Точку наблюдения удобно задать в сферических координатах $E(\rho, \theta, \varphi)$, то есть ее мировые координаты могут быть вычислены по формулам:

$$x_e = \rho \sin \varphi \cos \theta,$$

$$y_e = \rho \sin \varphi \sin \theta,$$

$$z_e = \rho \cos \varphi.$$

Вектор OE определяет направление наблюдения. Из точки наблюдения E можно видеть точки объекта только внутри некоторого

конуса, ось которого совпадает с линией OE , а вершина - с точкой E . При направлении взгляда из O в E положительная полуось x_e направляется вправо, а положительная полуось y_e - вверх. Такое направление осей позволяет определить экранные координаты в тех же направлениях. Направление оси z_e выбирается так, чтобы значение координат увеличивалось по мере удаления от точки наблюдения. При таком определении осей, которое представляется вполне удобным и логичным, система видовых координат будет левой. Тем не менее, такое несоответствие мировой и видовой систем координат не представляет никаких затруднений.

Видовое преобразование может быть записано в форме:

$$(x_e \quad y_e \quad z_e \quad 1) = (x_w \quad y_w \quad z_w \quad 1) \cdot V,$$

где V – матрица видового преобразования.

Видовое преобразование включает в себя следующую последовательность действий.

1. Перенос системы координат, при котором точка E становится новым началом координат.
2. Поворот системы координат около оси Z на угол $\pi/2 - \theta$ в отрицательном направлении так, что ось Y совпадет по направлению с горизонтальной составляющей отрезка OE , а ось X будет располагаться перпендикулярно отрезку OE .
3. Другой поворот около оси X на угол $\pi - \varphi$ в положительном направлении для того, чтобы ось z совпала с отрезком OE .
4. Необходимо изменить ориентацию оси X , выполнив отображение относительно плоскости XZ .

Матрица V представляет собой произведение матриц перечисленных линейных преобразований. После некоторого упрощения она будет иметь вид:

$$V = \begin{pmatrix} -\sin\theta & -\cos\varphi\cos\theta & -\sin\varphi\cos\theta & 0 \\ \cos\theta & -\cos\varphi\sin\theta & -\sin\varphi\sin\theta & 0 \\ 0 & \sin\varphi & -\cos\varphi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

После чего можно использовать видовые координаты x_e и y_e , отбросив координату z_e . Ее вычисляют в том случае, когда необходимо получить перспективные изображения и при последующей реализации алгоритма удаления невидимых линий. В этих случаях необходимо иметь полную картину видового преобразования.

Удаление невидимых линий и поверхностей

Для стирания невидимых линий был выбран алгоритм Робертса [70], который представляет собой первое известное решение задачи об удалении невидимых линий. Алгоритм является векторным. Это математически элегантный метод, работающий в объектном пространстве, то есть, используя нашу терминологию, в пространстве видовых координат. Алгоритм работает с выпуклыми телами. Если тело невыпуклое, то оно перед применением этого алгоритма должно быть разбито на совокупность выпуклых тел.

Разрезание невыпуклых тел

Ниже приводится алгоритм разбиения невыпуклых тел на выпуклые части.

Процедура разрезания для каждой полигональной грани тела такова:

1. Перенести тело так, чтобы одна из вершин выбранной грани совпала с началом координат.
2. Повернуть тело вокруг начала координат так, чтобы одно из инцидентных ему ребер совпало с одной из осей координат, например, с осью x .
3. Повернуть тело вокруг выбранной оси так, чтобы выбранная грань совпала с одной из координатных плоскостей, например, с плоскостью $z = 0$.
4. Проверить знаки координаты, которая перпендикулярна выбранной грани (т.е. координаты z), для всех вершин тела, не лежащих на этой грани.
5. Если все знаки совпадают или равны нулю, то тело является выпуклым относительно этой грани. В противном случае оно невыпукло; разрезать тело плоскостью, несущей выбранную грань.
6. Повторить всю процедуру с каждым из вновь образовавшихся тел. Продолжать работу до тех пор, пока все тела не станут выпуклыми.

Метод Робертса для одного выпуклого тела

Уравнение произвольной плоскости в трехмерном пространстве имеет вид

$$ax + by + cz + d = 0.$$

В матричной форме:

$$(x \ y \ z \ 1) \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} = 0,$$

или

$$(x \ y \ z \ 1)(P)^T = 0,$$

где $(P) = (a \ b \ c \ d)$ представляет собой плоскость. Поэтому любое выпуклое твердое тело можно выразить через матрицу, состоящую из коэффициентов уравнений плоскостей, т.е.

$$(V) = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ b_1 & b_2 & \dots & b_n \\ c_1 & c_2 & \dots & c_n \\ d_1 & d_2 & \dots & d_n \end{pmatrix},$$

где каждый столбец содержит коэффициенты одной плоскости.

Напомним, что любая точка пространства представима в однородных координатах вектором $(S) = (x \ y \ z \ 1)$. Более того, если точка (S) лежит на плоскости, то $(S)(P)^T = 0$. Если же (S) не лежит на плоскости, то знак этого скалярного произведения показывает, по какую сторону расположена точка. В алгоритме Робертса предполагается, что точки, лежащие внутри тела дают положительное скалярное произведение.

Хотя уравнение плоскости содержит четыре неизвестных коэффициента, его можно нормировать так, чтобы $d = 1$. Следовательно, трех точек достаточно для определения этих коэффициен-

тов. Подстановка координат этих точек, (x_1, y_1, z_1) , (x_2, y_2, z_2) , (x_3, y_3, z_3) в нормированное уравнение дает:

$$\begin{pmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} -1 \\ -1 \\ -1 \end{pmatrix}.$$

Или в матричной форме:

$$(X)(C) = (D). \quad (3.27)$$

Из решения этого уравнения определяются значения коэффициентов уравнения плоскости:

$$(C) = (X)^{-1}(D).$$

Другой способ используется, если известен вектор нормали к плоскости $\bar{n} = ai + bj + ck$, где i, j, k - единичные векторы осей x, y, z , соответственно. Тогда уравнение плоскости примет вид

$$ax + by + cz + d = 0.$$

Величина d вычисляется с помощью произвольной точки на плоскости. В частности, если компоненты этой точки на плоскости (x_1, y_1, z_1) , то:

$$d = -(ax_1 + by_1 + cz_1).$$

Поскольку объем вычислений в алгоритмах удаления невидимых линий или поверхностей растет с увеличением числа многоугольников, для описания поверхностей выгодно использовать многоугольники с более чем тремя сторонами. Эти многоугольники могут быть как невыпуклыми, так и неплоскими. Следующий метод позволяет найти или точное решение для уравнений плоскостей, содержащих плоские многоугольники, или «наилучшее» при-

ближение для неплоских многоугольников. Этот метод эквивалентен определению нормали в каждой вершине многоугольника посредством векторного произведения прилежащих ребер и усреднения результатов. Если a, b, c, d - коэффициенты уравнения плоскости, то

$$a = \sum_{i=1}^n (y_i - y_j)(z_i + z_j), \quad b = \sum_{i=1}^n (z_i - z_j)(x_i + x_j),$$

$$c = \sum_{i=1}^n (x_i - x_j)(y_i + y_j),$$

где

$$j = \begin{cases} 1, & i = n; \\ j+1, & i \neq n, \end{cases}$$

здесь n - число вершин многоугольника, а d вычисляется с помощью любой точки на плоскости.

Перед началом работы алгоритма удаления невидимых линий или поверхностей для получения желаемого вида сцены часто применяется трехмерное видовое преобразование. Матрицы тел для объектов преобразованной сцены можно получить или преобразованием исходных тел, или вычислением новой матрицы тел, используя преобразованные вершины или точки.

Если (B) - матрица однородных координат, представляющая исходные вершины тела, а (T) - матрица размером 4x4 видового преобразования, то преобразованные вершины таковы:

$$(BT) = (B)(T), \quad (3.28)$$

где (BT) - преобразованная матрица вершин. Использование уравнения (3.27) позволяет получить уравнения исходных плоскостей, ограничивающих тело:

$$(B)(V) = (D), \quad (3.29)$$

где (V) - матрица тела, а (D) - матрица, состоящая из нулей. Аналогично уравнения преобразованных плоскостей задаются следующим образом:

$$(BT)(VT) = (D), \quad (3.30)$$

где (VT) - преобразованная матрица тела. Приравнивая левые части уравнения (3.29) и (3.30), получаем

$$(BT)(VT) = (B)(V).$$

Подставляя уравнение (3.28), сокращая на (B) и умножая слева на $(T)^{-1}$ имеем

$$(VT) = (T)^{-1}(V).$$

Итак, преобразованная матрица получается умножением исходной матрицы тела слева на обратную матрицу видового преобразования.

Тот факт, что плоскости имеют бесконечную протяженность и что скалярное произведение координат точки на матрицу тела отрицательно, если точка лежит вне этого тела, позволяет построить метод, в котором матрица тела используется для определения граней, экранирующих самим телом.

Если зритель находится в бесконечности на положительной полуоси z и смотрит на начало координат, то его взгляд направлен в сторону отрицательной полуоси z . В однородных координатах вектор такого направления равен $(E) = (0 \ 0 \ -1 \ 0)$. Он служит, кроме того, образом точки, лежащей в бесконечности на отрицательной полуоси z . Фактически (E) представляет точку, лежащую на плоскости $z = -\infty$, т.е. любую точку типа $(x, \ y, \ -\infty)$. Поэтому, если скалярное произведение (E) на столбец, соответствующий какой-нибудь плоскости в матрице тела, отрицательно, то (E) лежит по отрицательную сторону этой плоскости. Следовательно,

эти плоскости невидимы из любой точки наблюдения, лежащей в плоскости $z = \infty$, а пробная точка на $z = -\infty$ экранируется самим телом. Такие плоскости называются нелицевыми, а соответствующие им грани - задними. Следовательно, отношение $(E)(V) < 0$ является условием того, что плоскости нелицевые, а их грани задние. Заметим, что для аксонометрических проекций (точка наблюдения в бесконечности) это эквивалентно поиску положительных значений в третьей строке матрицы тела.

Этот метод является наиболее известным алгоритмом удаления невидимых поверхностей для тел, представляющих собой одиночные выпуклые многогранники, его также часто называют отбрасыванием задних плоскостей. Для выпуклых многогранников число граней при этом сокращается примерно наполовину. Метод эквивалентен вычислению нормали для каждого многоугольника. Отрицательность нормали к поверхности говорит о том, что нормаль направлена в сторону от наблюдателя, а следовательно данный многоугольник не виден. Этот метод используется также и для простой закраски, в этом случае интенсивность тона элемента делается пропорциональной проекции нормали к поверхности на направление взгляда наблюдателя.

После удаления нелицевых отрезков необходимо выяснить существуют ли отрезки, которые экранируются другими телами в картине или сцене. При этом очень полезно использовать приоритетные сортировки по глубине и простые (минимаксный и габаритный) тесты, позволяющие удалить целые группы отрезков и тел. Использование этих приемов значительно сокращает число тел, с которыми нужно сравнивать каждый отрезок или ребро. Алгоритм сравнения каждого из оставшихся отрезков с оставшимися неэкранированными телами представляет собой достаточно сложную самостоятельную задачу и здесь не рассматривается. Кроме того, необходимо отметить, что алгоритм для одного выпуклого тела позволяет работать с достаточно простыми невыпуклыми телами, форма которых близка к выпуклой.

3.7. Построение сеток в методах граничных элементов

Важным этапом применения различных численных методов в научных расчетах является этап построения сеток. В методах гра-

ничных элементов приходится иметь дело с двумя типами сеток – поверхностной и внутренней. Поверхностная сетка необходима для дискретизации границ решаемой области. Внутренняя сетка служит для вычисления вектора правой части, если решается уравнение Пуассона, или для нахождения внутренней картины течения после проведения расчета задачи [24]. Здесь рассматриваются способы построения сеток внутри области для плоских и осесимметричных задач. Для пространственного случая приводятся способы построения поверхностных сеток.

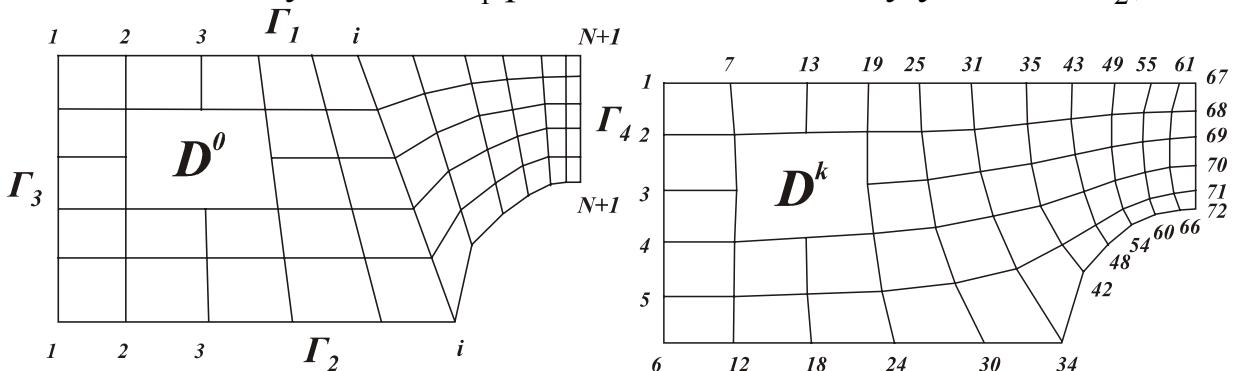
Построение двумерной четырехугольной сетки

Проблеме построения сеток на плоскости посвящено большое количество работ и различных исследований. Отдельные подходы к решению этой задачи рассматриваются, например, в [29]. Многие методы эффективны только в областях с фиксированной геометрией и, кроме того, требуют дополнительной перенумерации узлов для уменьшения вычислительных затрат. Результатом работы любого алгоритма генерации сетки являются массив координат узлов и массив связности, определяющий каждый четырехугольный элемент сетки. Два этих массива однозначным образом определяют геометрию построенной сетки.

Рассмотрим наиболее простые методы построения двумерных четырехугольных сеток по известному распределению узлов на границе односвязной области.

Первый способ изложен в работе [78]:

а) *входная информация*. Верхняя Γ_1 и нижня Γ_2 границы области D (рис.9, а) разбиваются на N частей в соответствии с требованиями задачи. Координаты узлов на Γ_1 и Γ_2 являются входными параметрами. Каждому узлу на Γ_1 и Γ_2 присваивается порядковый номер от 1 до $N+1$. Единственное ограничение состоит в том, чтобы количество узлов на Γ_1 равнялось количеству узлов на Γ_2 ;



а)

б)

Рис. 9. а) Исходная, б) оптимизированная сетка из четырехугольных элементов

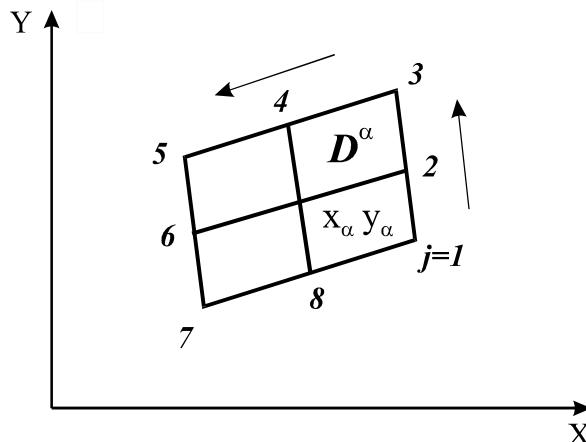


Рис. 10. Схема для построения итерационного процесса

б) *алгоритм.* Узлы с одинаковыми порядковыми номерами на Γ_1 и Γ_2 соединяются прямыми линиями, которые, в свою очередь, делятся на одинаковое (заданное) число отрезков K . В результате получается некоторая совокупность узлов, координаты которых являются выходными параметрами. Очевидно, что общее число точек разбиения будет $(N+1) \times (K+1)$. Для получения сетки четырехугольных элементов необходимо точки области соединить системой отрезков так, чтобы получилась сетка с числом $N \times K$ четырехугольных элементов. Нумерация узлов и элементов осуществляется в произвольном порядке, например, сверху вниз и слева направо, начиная с первого узла границы Γ_1 (рис. 9,б).

Полученная в результате указанного разбиения сетка (рис. 9, а) требует дополнительной доработки (оптимизации). Для этих целей имеется большой набор возможностей, связанный с организацией итерационного процесса, например:

$$\vec{r}_{\alpha}^{n+1} = \sum_{j=1}^l \beta_i \vec{r}_j^n, \vec{r}_j = \{x_j, y_j\} \in D^\alpha. \quad (3.31)$$

Здесь \vec{r}_j - радиус-векторы точек, окружающих точку $\vec{r}_\alpha (\alpha \neq j)$ по границе ∂D^α области D^α (рис.10). Весовые параметры β_i выбираются из различных соображений, например, из требования минимальности разности длин отрезков, пересекающихся в точке \vec{r}_α .

Хороший результат дает способ, в котором коэффициенты $\beta_j = 1/8$, а $l = 8$. При таком выборе весовых коэффициентов координаты каждого внутреннего узла равны среднему арифметическому окружавших его узлов. Итерационный процесс (3.31) быстро сходится и дает достаточно равномерную и почти ортогональную сетку (рис. 9, б).

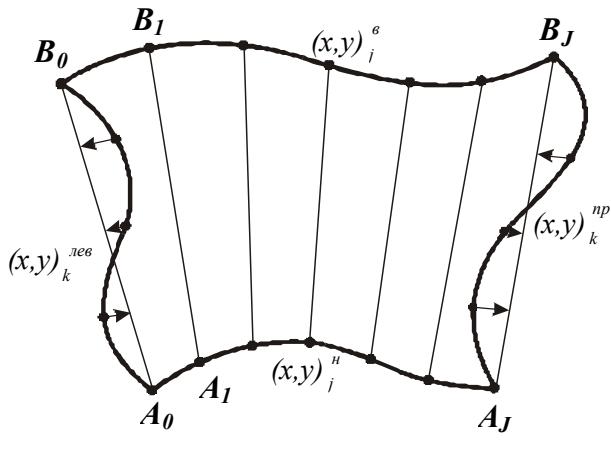


Рис. 11. Первоначальная сетка

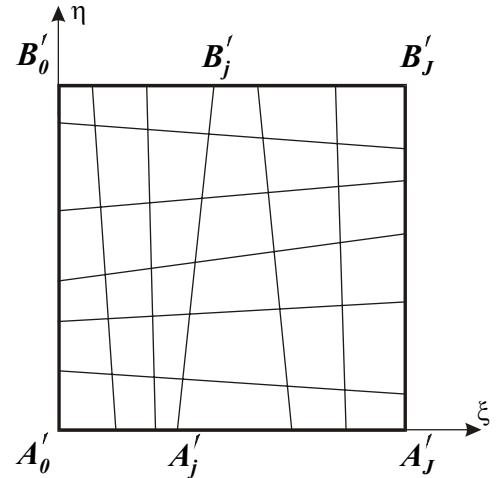


Рис. 12. Оптимизированная сетка

Второй способ, изложенный в работе [29], во многом аналогичен предыдущему, но все же имеет некоторые отличия.

а) *входная информация*. Представим счетную область D на плоскости xu в виде четырехугольника с криволинейными границами. Число узлов на противоположных границах одинаково (рис. 11). Распределение узлов на каждой из четырех границ задано и имеет следующий вид:

$$\begin{aligned} & \left\{ (x, y)_j^n; \quad j = 0, 1, \dots, J \right\}, && \text{для нижней границы } (k = 0), \\ & \left\{ (x, y)_j^e; \quad j = 0, 1, \dots, J \right\}, && \text{для верхней границы } (k = K), \\ & \left\{ (x, y)_j^{лев}; \quad k = 0, 1, \dots, K \right\}, && \text{для левой границы } (j = 0), \\ & \left\{ (x, y)_j^{np}; \quad k = 0, 1, \dots, K \right\}, && \text{для правой границы } (j = J). \end{aligned}$$

Крайние точки этих последовательностей, задающие положение четырех углов области, совпадают:

$$\begin{aligned} (x, y)_0^h &= (x, y)_0^{\text{лев}}, & (x, y)_0^h &= (x, y)_0^{\text{np}}, \\ (x, y)_0^{\text{np}} &= (x, y)_0^{\text{в}}, & (x, y)_0^{\text{лев}} &= (x, y)_0^{\text{в}}. \end{aligned}$$

б) алгоритм. Расстановка точек на контуре счетной области может быть выбрана весьма неравномерной (нерегулярной) и осуществляется исходя из конкретных особенностей задачи. Поэтому необходимо ввести закон расстановки узлов сетки по границе области. Этот закон определяет вспомогательная последовательность, $\{s_n\}$, которая рассчитывается по формуле:

$$\bar{s}_0 = 0, \quad \bar{s}_n = \bar{s}_{n1} + \sqrt{(x_n - x_{n1})^2 + (y_n - y_{n1})^2}, \quad n = 1, \dots, N.$$

Для дальнейшего рассмотрения удобно нормировать полученные значения: $s_n = \bar{s}_n / \bar{s}_N$. Такой расчет выполняется для каждой из четырех границ. В результате получаем:

$$\begin{aligned} \left\{ s_j^h; \quad j = 0, 1, \dots, J \right\}, \quad & \left\{ s_j^{\text{лев}}; \quad k = 0, 1, \dots, K \right\}, \\ \left\{ s_j^{\text{в}}; \quad j = 0, 1, \dots, J \right\}, \quad & \left\{ s_j^{\text{np}}; \quad k = 0, 1, \dots, K \right\}. \end{aligned}$$

По заданному граничному разбиению необходимо найти координаты $\{(x, y)_{j,k}; \quad j = 1, \dots, J-1; k = 1, \dots, K-1\}$ внутренних узлов сетки. Для этого строится отображение криволинейной области на параметрический квадрат $0 \leq \xi \leq 1, 0 \leq \eta \leq 1$ (рис. 12). Соответствующие точки на противоположных сторонах (расставленные по законам $\{s_j^h\}, \{s_j^{\text{в}}\}, \{s_k^{\text{лев}}\}, \{s_k^{\text{np}}\}$) соединяются отрезками прямых. Координаты $(\xi, \eta)_{j,k}$ точек пересечения построенных отрезков вычисляются по формулам:

$$\xi_{j,k} = \frac{s_j^h + (s_j^{\text{в}} - s_j^h)s_k^{\text{лев}}}{1 - (s_j^{\text{в}} - s_j^h)(s_k^{\text{np}} - s_k^{\text{лев}})}, \quad \eta_{j,k} = \frac{s_k^{\text{лев}} + (s_k^{\text{np}} - s_k^{\text{лев}})s_j^h}{1 - (s_j^{\text{в}} - s_j^h)(s_k^{\text{np}} - s_k^{\text{лев}})}. \quad (3.32)$$

Интерполяция координат x, y между нижней и верхней границами осуществляется по формулам:

$$x_{j,k} = x_j^h(1 - \eta_{j,k}) + x_j^e\eta_{j,k}, \quad y_{j,k} = y_j^h(1 - \eta_{j,k}) + y_j^e\eta_{j,k}.$$

Если левая и правая границы области являются отрезками прямых, то на этом расчет сетки заканчивается. В общем же случае формулы (3.32) необходимо “подправить”. В качестве поправок можно взять результат интерполяции по индексу j поправочных векторов для левой и правой границ, соединяющих истинные точки этих границ с вычисленными по формулам (3.32) (рис. 11). Это приводит к следующим формулам для расчета $(x, y)_{j,k}$:

$$\begin{aligned} x_{j,k} = & x_j^h(1 - \eta_{j,k}) + x_j^e\eta_{j,k} + \xi_{j,k}[x_k^{np} - x_J^h(1 - s_k^{np}) - x_J^e s_k^{np}] + \\ & +(1 - s_{j,k})[x_k^{лев} + x_0^h(1 - s_k^{лев}) - x_0^e s_k^{лев}], \end{aligned}$$

$$\begin{aligned} y_{j,k} = & y_j^h(1 - \eta_{j,k}) + y_j^e\eta_{j,k} + \xi_{j,k}[y_k^{np} - y_J^h(1 - s_k^{np}) - y_J^e s_k^{np}] + \\ & +(1 - s_{j,k})[y_k^{лев} + y_0^h(1 - s_k^{лев}) - y_0^e s_k^{лев}]. \end{aligned}$$

На основе описанных методов в пакете “*AKORD*” реализована возможность построения сеток, как для простых, так и для сложных многосвязных областей с сильно искривленными границами. Это реализуется за счет разбиения сложных областей на четырехугольные зоны и применения алгоритма построения сетки для каждой из зон в отдельности. После чего производится склейка построенных сеток с соответствующей перенумерацией узлов.

Построение пространственной треугольной сетки

В трехмерном случае проблема построения поверхностной сетки уже не является задачей очевидной и требует более внимательного к ней отношения. Ниже будут рассмотрены два способа построения треугольных сеток на поверхности пространственных объектов. Первый из них позволяет покрывать сеткой поверхности практически любых сколь угодно сложных объектов. Второй разработан специально для построения сетки на поверхности сферы (эллипсоида). Его преимущество состоит в том, что построенная

сетка является более равномерной, чем при использовании первого способа. Результатом работы обоих алгоритмов являются три массива, полностью определяющие геометрию области: массив координат узлов поверхностной сетки; массив согласования узлов для каждого граничного элемента, которые перечисляются в соответствии с выбранным направлением обхода; массив связности каждого узла и окружающих его элементов (сам узел, в свою очередь, также является частью этих элементов). Такое описание сетки является избыточным, так как по одному из массивов связности можно построить другой. Однако эта избыточность позволяет существенно ускорить работу с сеткой, что в условиях использования большого количества ресурсов компьютера при решении трехмерной задачи представляется очень важным.

Первый способ. Этот метод построения сетки основан на разбиении исходной поверхности на отдельные опорные четырехугольные или треугольные зоны (рис.13).

а) входная информация. Описание опорных зон области, включающее в себя координаты вершин опорных зон сетки, вид поверхности (зона может являться плоским или сферическим (эллиптическим) сегментом), количество точек разбиения на сторонах зоны.

б) алгоритм. Каждая опорная зона, отображаясь на каноническую область (топологически подобную данной), делится на заданное число элементов. Обратное преобразование позволяет получить требуемую сетку на поверхности объекта. После использования специального алгоритма слияния зоны собираются в единую поверхность. Таким образом, "вырезая" на поверхности плоские или криволинейные сегменты и разбивая отдельно каждый из них, можно покрывать сеткой практически любые объекты. Простой сегмент может представлять собой четырехугольную или треугольную зону, которая непосредственно разбивается на элементы. Сложные сегменты поверхности можно разделить, в свою очередь, на более простые опорные зоны.

Данный способ был реализован в работе [97].

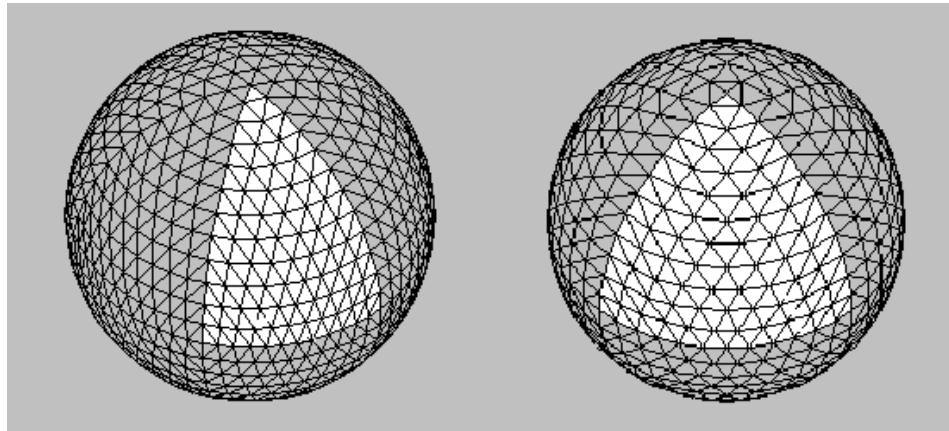


Рис.13. Треугольные опорные зоны на поверхности сферы

Введение L - координат

Рассмотрим плоский треугольный элемент D с координатами вершин P_1, P_2, P_3 и площадью S . Выберем внутри этого треугольника произвольную точку P и соединим ее с вершинами элемента. Пронумеруем подобласти, как показано на рис.14, и введем безразмерные переменные: $L_i = S_i / S$, где S_i - площадь соответствующей подобласти. Переменные L_1, L_2, L_3 называются локальными или L – координатами.

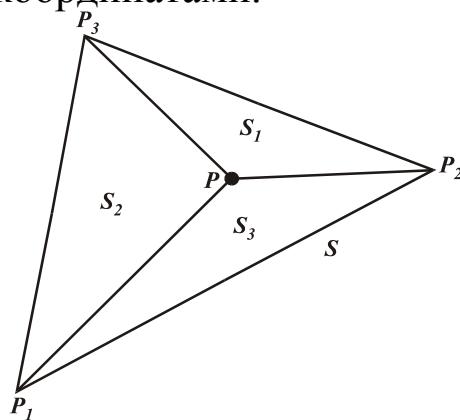


Рис. 14. Нумерация подобластей при введении L -координат

Легко убедиться, что для любой точки P треугольника выполняются соотношения:

$$x = L_1 x_1 + L_2 x_2 + L_3 x_3,$$

$$y = L_1 y_1 + L_2 y_2 + L_3 y_3,$$

$$1 = L_1 + L_2 + L_3.$$

Отсюда нетрудно выразить переменные L_i через координаты узловых точек

$$L_1 = (a_1 + b_1x + c_1y)/2S, \quad L_2 = (a_2 + b_2x + c_2y)/2S, \quad L_3 = (a_3 + b_3x + c_3y)/2S$$

,

где

$$S = \frac{1}{2} \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix},$$

$$a_i = x_j y_k - x_k y_j, \quad b_i = y_j - y_k, \quad c_i = x_k - x_j, \\ i = 1, 2, 3; \quad j = 2, 3, 1; \quad k = 3, 2, 1.$$

Особенность L -координат состоит в том, что они принимают постоянные значения на линиях, параллельных сторонам зоны (см. рис.15). По известной линейной зависимости между декартовыми и L -координатами, а также заданному количеству делений каждой стороны треугольной зоны легко построить плоскую сетку.

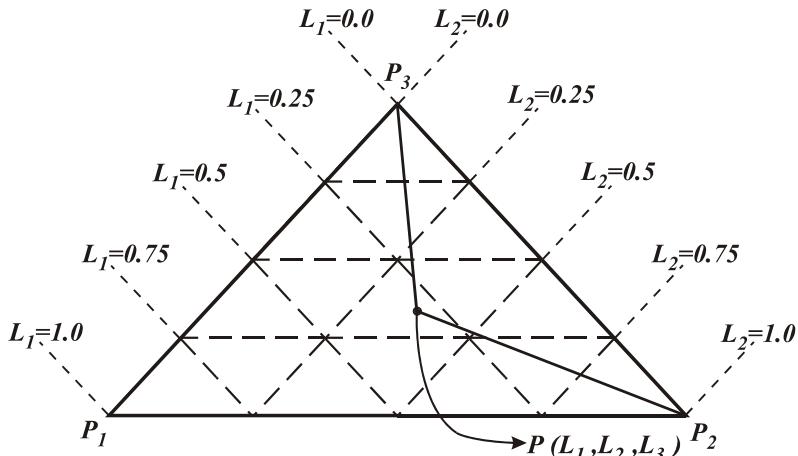


Рис. 15. Разбиение треугольной опорной зоны на треугольные элементы с использованием L -координат

Координаты внутренних узлов сетки получаются из следующих соотношений

$$\left\{ X = \sum_{i=1}^3 L_i X_i, \quad Y = \sum_{i=1}^3 L_i Y_i, \quad Z = \sum_{i=1}^3 L_i Z_i. \right.$$

В случае, когда зона имеет криволинейные границы, используется механизм определения специальных функций формы на тре-

угольном элементе, которые в случае плоского треугольного элемента совпадают с L -координатами. В работе [97] представлен аналогичный метод для разбиения четырехугольных областей, где вместо L -координат используются базисные функции для четырехугольных элементов.

Недостатком описанного подхода является необходимость "вырезания" большого количества опорных зон для сильно искривленных поверхностей. Однако для канонических областей, таких как сфера, эллипсоид и т.п., данный подход легко модифицировать, используя криволинейные координаты, хорошо аппроксимирующие такого рода поверхности. Для этого вводятся полярные "долгота" и "широта", которые и будут играть роль L -координат. При разбиении более сложных объектов можно использовать комбинацию криволинейных координат и L -координат в зависимости от вида вырезаемой зоны. В этом случае каждая зона разбивается в соответствии с одним из подходов, а для связи зон используется алгоритм слияния [30]. Необходимо отметить также, что координаты узлов, расположенных на стыках опорных зон, должны совпадать.

После того, как сетка на каждой опорной зоне построена, требуется «склеить» опорные зоны с тем, чтобы ввести единую нумерацию узлов и элементов. В процессе разбиения допускается произвольная нумерация узлов зоны. Алгоритм склейки зон обеспечивает нахождение совпадающих узлов на границах прилегающих зон и их перенумерацию в определенном для всей области порядке.

Второй способ

а) входная информация. Номер уровня дискретизации.

б) алгоритм. Исходным приближением поверхности сферы является икосаэдр, состоящий из 12 узлов и 20 ребер. На каждой итерации каждый поверхностный треугольник разбивается на четыре следующим образом: каждая его сторона делится пополам и сдвигается в радиальном направлении на поверхность сферы, после чего все новые вершины соединяются ребрами. Как уже отмечалось, на каждом уровне дискретизации количество элементов сетки увеличивается в четыре раза, поэтому количество элементов растет очень быстро. Таким образом, реально удается получить две-три приемлемые расчетные сетки. Количество узлов и элементов для

каждого уровня дискретизации указано в таблице 5, первые уровни дискретизации показаны на рис. 16.

Таблица 5

Количество узлов и элементов для разных уровней дискретизации поверхностной сетки на сфере

Уровень дискретизации	Кол-во узлов	Кол-во элементов
0	12	20
1	42	80
2	162	320
3	642	1280
4	2562	5120

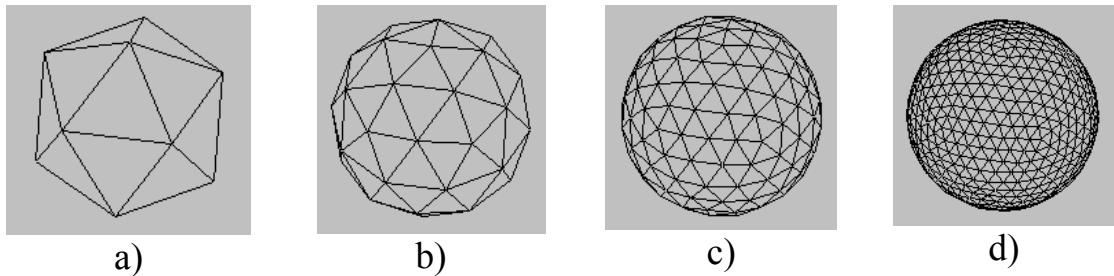


Рис. 16. Различные уровни дискретизации сетки на поверхности сферы (а-нулевой уровень (икосаэдр), б-первый, в-второй, г-третий)

Г л а в а 4

ПАКЕТ ПРИКЛАДНЫХ ПРОГРАММ “*AKORD*”

В практике проведения вычислительных экспериментов использование пакетов прикладных программ является общепринятым (см. например, [89]). Для решения задач гидродинамики идеальной несжимаемой жидкости со свободными границами был разработан пакет “*AKORD*”, описанный в работах [12, 13].

ППП “*AKORD*” состоит из нескольких основных приложений, реализующих все основные этапы численного решения задач, начиная с описания начальных данных задачи и заканчивая графическим анализом полученных результатов:

- *препроцессор* – программный компонент, с помощью которого готовятся входные данные для численного эксперимента;
- *решатель* – позволяет численно решать задачи, опираясь на данные, подготовленные препроцессором;
- *постпроцессор* – программный модуль, направленный на графическое отображение полученных после работы решателя результатов.

Каждый компонент пакета “*AKORD*” может функционировать как отдельная программная единица, которая напрямую получает необходимые данные, обрабатывает их и выдает результаты в соответствии с единым интерфейсом обмена, разработанным для совместимости приложений в составе пакета.

4.1. Препроцессор

Проблема подготовки исходных данных, ввиду сложности начальной геометрии расчетной области, решается при помощи специализированных программ называемых “препроцессорами”. В составе пакета разработан препроцессор для плоских и осесимметричных задач, обеспечивающий подготовку данных для решения задач гидродинамики идеальной несжимаемой жидкости со свободными границами. Простой и понятный пользовательский интерфейс препроцессора позволяет легко генерировать входные данные для расчета.

Основными функциями препроцессора являются:

- 1) аппроксимация расчетной области (областей) решаемой задачи;
- 2) задание начальных и граничных условий в узлах построенной сетки;
- 3) определение дополнительных параметров задачи (физических, временных и т.д.);
- 4) подготовка специфических данных для расчета в виде файла в соответствии с принятым интерфейсом.

Поскольку препроцессор ориентирован на использование численного метода граничных элементов, то это налагает определенные особенности на характер начальных данных. Одна из таких особенностей – возможность использования технологии «двойных узлов» в точках смены типа граничных условий. В этих точках задаются два граничных условия, а длина граничного элемента считается нулевой.

Главное окно препроцессора (рис. 17) представлено несколькими областями:

- a). Область меню. Эта область находится в верхней части окна. Работа с ней подробно описана в приложении 1.
- b). Область пиктограмм. Эта область полностью дублирует некоторые пункты меню. Используется для более быстрого доступа к меню, что упрощает работу пользователя.
- c). Область отображения информации. В этой области располагаются одно или несколько дочерних окон, которые служат для отображения создаваемых рисунков, в дальнейшем - графических окон. Таких окон на экране может быть несколько. Активным окном является то, у которого выделен заголовок. Переключения между окнами

нами осуществляются очень просто, для этого достаточно подвести курсор к любой видимой части нужного окна и щелкнуть мышкой, либо выбрать пункт меню <Windows> с соответствующим именем окна. Окна можно перемещать по экрану, изменять их размер, раскрывать до максимального размера или сжимать в пиктограмму.

d). Область состояния. Эта область находится в нижней части главного окна программы. В ней расположены окна отображения текущего положения курсора в физических координатах, а также выдается сообщение о текущем состоянии или подсказка о необходимых действиях пользователя. Кнопка <OK> вызывает перерисовку содержимого области активного окна.

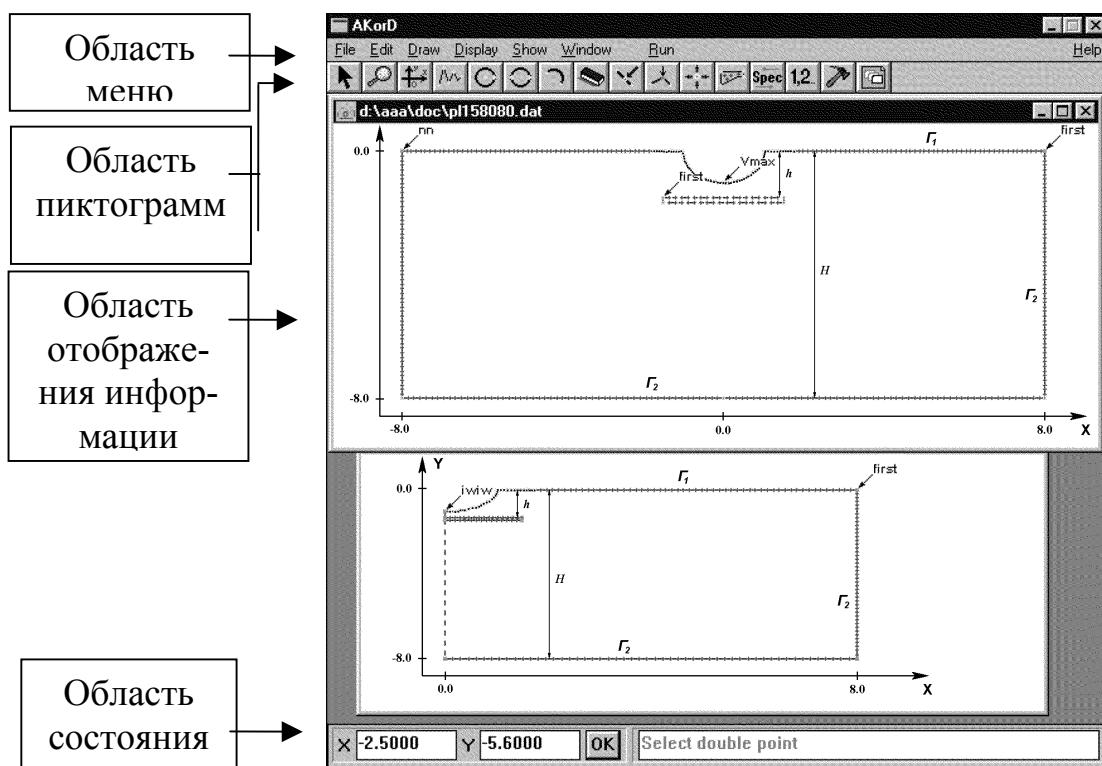


Рис. 17. Главное окно препроцессора

В состав препроцессора входит база данных графических примитивов, которые пользователь использует для построения геометрии расчетной области. Поэтому одной из главных функций препроцессора является предоставление удобного интерфейса для работы с графическими объектами. База данных графических примитивов построена на основе технологии описания графических объектов [12]. Сходные по описанию графические объекты собраны в классы.

На рис. 18 приведена структура графических примитивов. Все примитивы являются потомками класса «DrawPrimitive», который, в свою очередь, порожден от базового графического примитива (смотри п. 3.6.). Примитив «PolyLine» является базовым для сложных графических примитивов задаваемых, например, аналитическими функциями (синус, косинус и т.д.). Примитив «Spline» строится при помощи параметрического кубического сплайна на основе опорных точек, задаваемых пользователем или считываемых из файла. Производным от него является примитив «Soliton», используемый при моделировании задач о распространении уединенной волны. Начальная форма солитона и распределение потенциала на нем задается при помощи приближенных формул (Буссинеска-Лейтона, Овсянникова), или из базы данных, построенной на основе решения стационарной задачи об уединенной волне [15]. Примитив «Wing» применяется при моделировании стационарных задач о циркуляционном обтекании профилей потоком весомой жидкости со свободной границей.

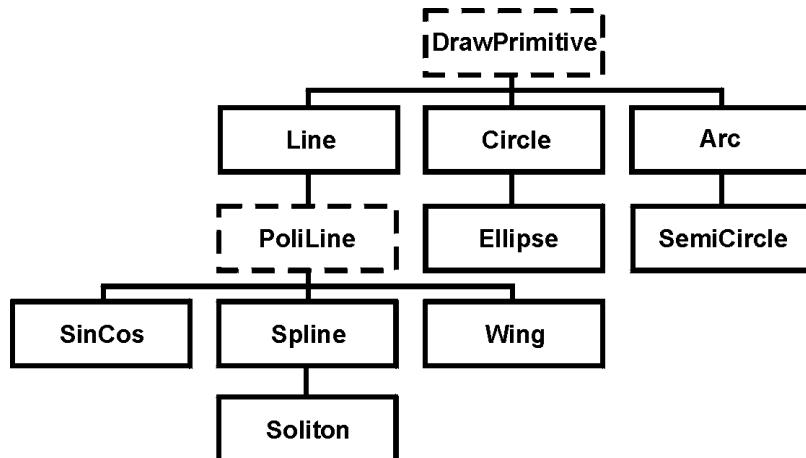


Рис. 18. Структура графических примитивов

Над примитивами можно выполнять следующие действия:

- создание и удаление примитивов;
- перемещение примитивов (либо целиком всего примитива, либо отдельной его вершины);
- разбиение примитивов на граничные элементы с различными параметрами сгущения;
- задание в узлах разбиения необходимых граничных условий.

В соответствии с базовой структурой классов примитивы являются наименьшими неделимыми единицами графического конструирования. В этом случае графическим объектом будет являться

каждая подобласть сконструированной области. Подобласть может состоять из одного (например, «Circle») или из нескольких примитивов. Для каждой подобласти задается первая точка и направление обхода.

Препроцессор обладает развитым пользовательским программным интерфейсом, который позволяет существенно облегчить процесс ввода начальных данных задачи. Имеется возможность визуального ввода и редактирования начальной геометрии области, а также набор диалоговых окон для задания параметров задачи, решаемой методом граничных элементов.

Оболочка препроцессора построена по принципу многодокументного интерфейса (MDI), что дает возможность одновременного редактирования нескольких файлов данных. Набор кнопок на панели инструментов соответствует различным функциям препроцессора и обеспечивает быстрое выполнение необходимых действий.

Препроцессор выполнен в качестве приложения для операционной среды Windows95/98/NT, что делает его легкими в освоении и позволяет использовать широкий спектр возможностей, предоставляемый этими операционными системами.

Для более подробного знакомства с описанием работы препроцессора можно обратиться к приложению 1.

4.2. Решатели

Решатели представляют собой программную реализацию метода граничных элементов, выполненную на основе стандарта языка Fortran-77/90. По сформированному с помощью препроцессора файлу начальных данных решатель выполняет расчет конкретной задачи.

Реализованы решатели для численного исследования следующих задач:

- стационарное обтекание препятствий потоком тяжелой жидкости;
- нестационарное движение тяжелой жидкости в бассейне;
- нестационарное движение тел в жидкости со свободной границей;
- циркуляционное обтекание профиля потоком весомой жидкости;

- эволюция газового пузыря в жидкости.

В зависимости от значения параметров в файле данных происходит выбор соответствующего решателя. Общий алгоритм типичного расчета выглядит следующим образом:

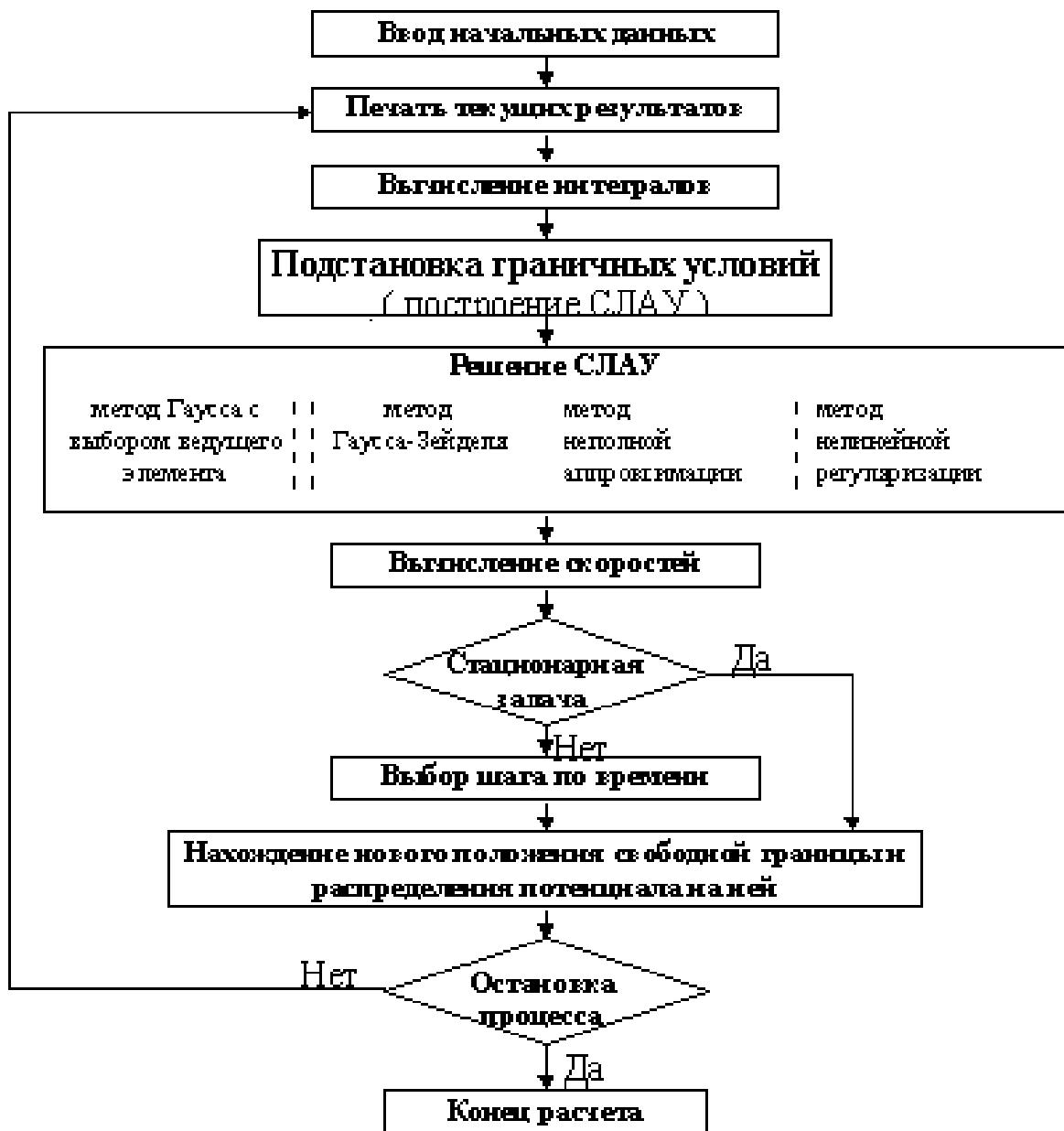


Рис. 19. Типичная блок-схема реализации решателя

Решатели разработаны таким образом, что могут выполняться как на локальной машине, так и на удаленном вычислительном сервере. Результатом работы решателя является текстовый файл, данные в котором сформированы в соответствии с интерфейсом обмена информацией между компонентами ППП “AKORD” так, чтобы их смог обрабатывать постпроцессор.

4.3. Постпроцессор

Постпроцессор предназначен для анализа результатов численного эксперимента и позволяет представить как геометрические, так и физические свойства изучаемого процесса в графическом виде. Постпроцессор может обрабатывать статические данные (например, неизменяющую геометрию области решения задачи), различные физические параметры и данные, зависящие от времени (например, эволюцию свободной границы жидкости), а также просматривать результаты расчетов в динамике.

Постпроцессор состоит из нескольких блоков и оболочки, связанных программным интерфейсом в единое приложение. Он может быть использован как самостоятельно, так и в составе специализированного программного комплекса.

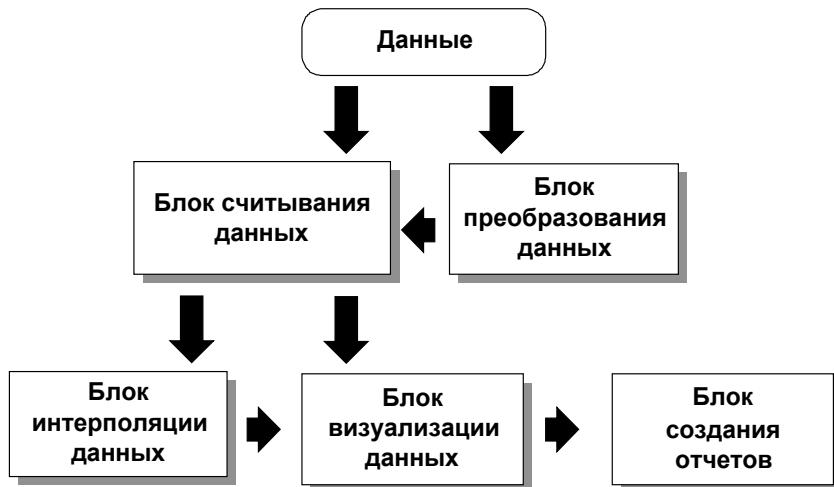


Рис. 20. Блок-схема работы постпроцессора

Схема обработки результатов численного решения задачи может быть представлена следующим образом (рис. 20). Сначала данные из результирующего файла попадают в блок считывания. Если блок считывания не может напрямую работать с файлом данного формата, то задействуется блок преобразования для того, чтобы конвертировать данные в нужный формат с последующей их передачей на вход блока считывания. После распознавания данные передаются, при необходимости, в блок интерполяции. Постпроцессор принимает решение о восполнении данных, если введенной информации не хватает для получения приемлемого графического образа. Если интерполяция не требуется, то данные поступают в блок визуализации или на вход блока создания отчета.

Перечисленные компоненты являются основными в структуре

постпроцессора. Кроме того, для дополнительной обработки информации могут быть задействованы вспомогательные блоки - задания параметров времени и задания зависимостей данных.

Функциональное назначение и основные возможности компонентов постпроцессора представлены ниже.

Блок считывания данных предназначен для ввода данных из файла результатов численного решения задачи и предоставления другим блокам доступа к ним через внутренний интерфейс обмена данными. Опираясь на него остальные блоки могут получить все необходимые для дальнейшей обработки параметры (например, количество данных, границы допустимых значений параметров задачи, время, для которого они получены и т.д.). Если формат вводимых данных не распознается этим блоком, то дополнительно может быть загружен блок преобразования данных.

Блок интерполяции данных позволяет через специальный интерфейс получить данные в заданный момент времени или в заданной точке. Если данных достаточно для дальнейшей обработки, то блок предоставляет их; если данных недостаточно - производится интерполяция на основе ближайших данных. Блок является "простройкой" между блоком считывания данных и блоком визуализации, являясь для последнего источником данных. Такая схема позволяет отображать данные вне зависимости от того, как они физически хранятся.

Блок преобразования данных служит для конвертирования данных из одного формата хранения в другой. Поскольку работа с данными, которые хранятся в двоичном виде проходит значительно быстрее, то целесообразнее хранить бинарные данные. Кроме того, обычно двоичные файлы занимают в 5-10 раз меньше места во внешней памяти, чем эквивалентные им текстовые файлы. По желанию пользователя любой двоичный файл, содержащий данные, с которыми работает постпроцессор, может быть преобразован в текстовый вид.

Блок задания параметров времени позволяет задать текущее время и временной шаг для динамического отображения данных в специальных фреймах, а также включить или остановить режим "прокрутки" информации. Блок определяет временные границы для текущего открытого файла данных, если последний содержит динамически отображаемые данные (например, итерации).

Блок визуализации данных позволяет получить графическое представление зависимостей данных. Для динамических данных есть возможность посмотреть их изменение с течением времени. Блок визуализации имеет три режима работы:

- **режим выбора объектов** (нормальный режим) предоставляет пользователю возможность выбрать графические объекты, которые описываются зависимостями данных, изменить их свойства (цвет, толщину линий, стиль линий и т.д.). Здесь же можно настроить вид отображаемых параметров для зависимостей данных. Например, для зависимости данных типа свободная граница параметрами могут быть скорость, ускорение в каждой точке и т.д. Возможные графические свойства, которые используются при их выводе на экран - это цвет, масштаб вектора; для параметров, которые выводятся в виде текстовых значений - шрифт. Данный режим включен по умолчанию;
- **режим зуммирования** дает возможность увеличить нужные участки зависимости данных или целиком всю указанную область, которая задается либо через диалог, либо визуально с помощью указателя мыши;
- **режим выделения фреймов** служит для определения частей изображения, за которыми необходимо следить с течением времени. Здесь указывается область для отображения в окне и скорость ее "перемещения". В постпроцессоре используются два типа фреймов. Первый тип используется для наблюдения за некоторой областью решения. Для него задаются границы области и вертикальная и горизонтальная компоненты скорости, с которой данный фрейм должен двигаться. Фреймы второго типа, так называемые «следящие» фреймы, используются для наблюдения за перемещением некоторой точки границы. Для этих фреймов задается специальная точка привязки, после чего фрейм «двигается» по области вместе с ней. Пользователь может выделить сразу несколько фреймов обоих типов, они будут располагаться в отдельном окне в виде блокнота. Пользователь сможет легко переходить от одного фрейма к другому. Данный режим имеет смысл использовать только при работе с динамическими данными.

Блок визуализации позволяет пользователю сохранять графические изображения в виде стандартных графических файлов форматов BMP, PCX, WMF.

матов BMP, PCX, WMF.

Пользователь имеет возможность наложить однородные данные друг на друга (например, в одном окне можно просматривать изменение кинетической, потенциальной или общей энергии) или в динамике посмотреть накат нескольких волн на наклонную стенку для получения качественной картины различий в их поведении при варьировании параметров расчета. В блоке реализована дополнительная возможность построения сетки внутри области решения задачи, а затем, на ее основе, определения линий уровня.

Блок задания зависимостей данных служит для задания зависимостей данных и последующего их представления в блоке визуализации. Блок может работать с различными зависимостями: зависимость границы области от времени, зависимость глобального параметра от времени, зависимость суммы глобальных параметров от времени. Заданная зависимость передается блоку визуализации, который отображает ее в определенном пользователем виде.

Постпроцессор в текущей реализации работает со следующими зависимостями данных:

- **свободная граница** - последовательность узлов с координатами (X, Y) в которых могут быть заданы два векторных параметра: скорость и ускорение;
- **зависимость глобального параметра от времени** - последовательность узлов с координатами ($t, Value$), где $Value$ - значение параметра в момент времени t . В узлах дополнительно могут быть заданы два векторных параметра.

Блок создания отчета предназначен для генерации различного рода графических отчетов на основе данных из блоков считывания, интерполяции и визуализации. Графический отчет представляет собой совокупность графических объектов и сопроводительных надписей для них, а также легенд. Графические объекты предоставляются блоком визуализации данных, после чего появляется возможность их редактирования: выбор различного вида осей координат, добавление необходимых сопутствующих надписей и атрибутов (цвет, стиль линий, толщина и т.д.). Характерной особенностью модуля подготовки графических отчетов является то, что все объекты задаются в векторной форме. Это позволяет формировать графические изображения с любым необходимым разрешением. Графические изображения сохраняются в форматах BMP, PCX,

WMF или во внутреннем формате постпроцессора для последующего его редактирования и создания на его основе новых отчетов.

Постпроцессор имеет модульную структуру (рис. 21), которая основана на логической схеме, описанной выше (рис. 20). Все модули вызываются из оболочки постпроцессора, которая, в свою очередь, является обобщающим модулем, связывающим отдельные компоненты в единое приложение.

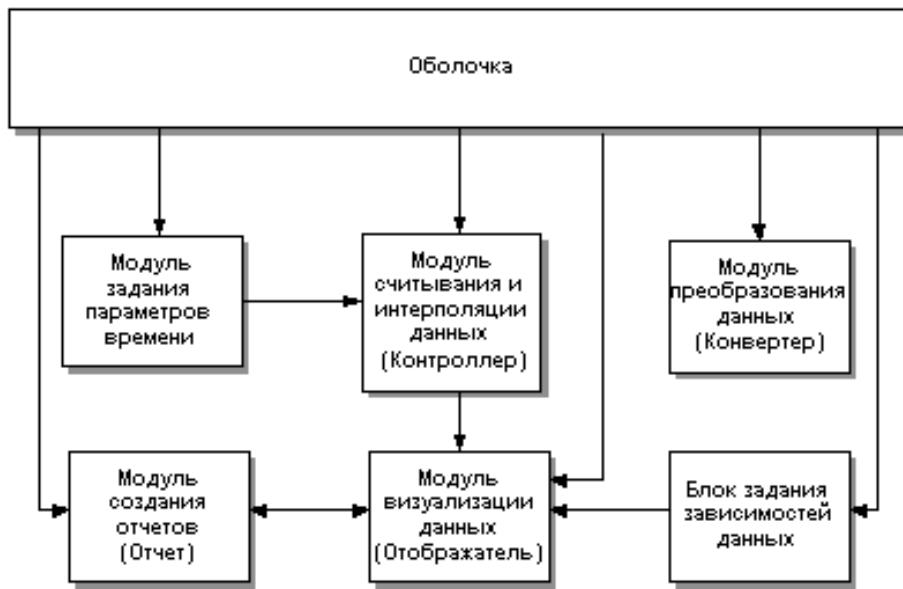


Рис. 21. Структурная схема постпроцессора

Оболочка препроцессора представлена в виде основного окна, имеющего линейки инструментов и меню. В главном окне открываются дополнительные окна, при необходимости создаваемые модулями, обеспечивающими визуализацию данных и создание отчетов.

Логически все блоки системы сделаны либо в виде одного модуля, либо представляют собой набор модулей, каждый из которых реализует какую-то определенную функцию.

Блок считывания данных реализован в виде нескольких модулей, так называемых «контроллеров данных», которые также выполняют функцию блока интерполяции данных.

В данной версии постпроцессора реализован «контроллер итераций» для работы с данными типа итерации (двоичный файл).

Блок визуализации представлен набором модулей, так называемых «отображателей данных». Каждый из отображателей данных содержит окно, в котором отображаются заданные зависимости данных и связанные с ними параметры.

В данной версии постпроцессора реализованы два отображателя данных:

- **отображатель свободной границы** позволяет просматривать в динамике несколько зависимостей данных типа свободная граница. Поддерживает все три режима работы отображателя данных: режим просмотра/задания параметров данных, режим масштабирования и режим выделения фреймов. Здесь же можно задавать параметры отображения зависимостей данных: масштабы и графические атрибуты векторных параметров;
- **отображатель глобальных параметров** позволяет просмотреть зависимость глобального параметра или совокупности глобальных параметров от времени. Может работать только в двух режимах: просмотра/задания свойств объектов и масштабирования. Не поддерживается режим выделения фреймов.

Блочная структура постпроцессора предусматривает возможность разработки дополнительных программных компонентов (модулей), позволяющих использовать существующие алгоритмы визуализации при появлении новых типов результирующих данных.

Постпроцессор реализован в виде приложения для операционной среды Windows95/98/NT, что делает его легким в освоении и позволяет использовать широкий спектр возможностей, предоставляемый этими операционными системами.

Для более подробного знакомства с описанием работы постпроцессора можно обратиться к дополнительной информации (Приложение 2).

Г л а в а 5

ОТ ОТДЕЛЬНЫХ ПРОГРАММ К ИНТЕГРИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЕ ПОДДЕРЖКИ ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА

Пакет прикладных программ «AKORD» в основном поддерживает все этапы проведения численного эксперимента. Однако исследователю требуется инструмент, предоставляющий более широкие возможности. Чтобы это понять, рассмотрим общие проблемы, которые встают перед исследователем, использующим вычислительный эксперимент в качестве основного инструмента для своего познания.

Вычислительный эксперимент определяется как построение математической модели исследуемого объекта посредством использования ЭВМ [72]. Общую схему проведения вычислительного эксперимента (рис. 22) можно представить следующим образом:



Рис. 22. Этапы вычислительного эксперимента

С развитием современных средств вычислительной техники исследователь получает в свои руки все более мощные инструменты, как для проведения непосредственных вычислений, так и для графической обработки и последующего анализа полученных результатов. Как следствие, перед ним встает проблема наиболее эффективного использования мощных технических возможностей для своей работы.

Очень важным является вопрос о систематизации и хранении уже полученных результатов в ходе проведения вычислений. Это могут быть различные варианты решения одной и той же задачи, тестовые расчеты или решение различных задач.

Еще одна проблема – использование вычислительных ресурсов или внешней памяти более мощных вычислительных систем, которые в настоящее время становятся все доступнее для широкого круга пользователей.

Таким образом, в современных условиях научно-технического прогресса и культуры исследований становится совершенно обоснованной необходимость построения интегрированной информационной системы (ИС) для обеспечения вычислительного эксперимента.

Естественно, что эта мысль не нова. Уже существует множество мощных автоматизированных систем, позволяющих решать различные задачи физики и математики (например, ALGOR (<http://www.algor.com>), BEASY (<http://www.beasy.com>), Минкур [89] и др.). Однако эти системы, как правило, направлены на решение только определенного круга однотипных задач. С другой стороны, из схемы вычислительного эксперимента можно выделить общие для решения всех задач блоки – подготовка данных к решению, численное решение задачи каким-нибудь методом и обработка результатов. Эти блоки могут закрываться достаточно широкопрофильными программными компонентами – препроцессором, решателем и постпроцессором, соответственно.

Исходя из вышеизложенных соображений, был построен прототип ИС “**AKORD**”, который, может быть, еще далек от состояния законченной коммерческой системы, но уже обладает основными необходимыми функциональными возможностями. Далее этот раздел посвящен описанию первого действующего прототипа такой системы.

5.1. Структура информационной системы

Отличительной чертой ИС “*AKORD*” является ее модульная структура построения. Каждый из модулей может быть заменен другим, направленным на решение конкретной задачи, сохраняя при этом все связи между модулями. Это позволяет рассматривать разрабатываемую ИС как основу информационной поддержки почти всех стадий вычислительного эксперимента.

Прототип ИС, как и его предшественник, состоит из нескольких «старых» приложений (препроцессор, решатели и постпроцессор), а также дополнен программами, позволяющими готовить данные и проводить расчеты пространственных задач методами граничных элементов. Кроме того, ИС “*AKORD*” оснащена дополнительными структурами для хранения данных и новым программным интерфейсом, которые реализуют все основные этапы численного решения задач, начиная с описания задачи и заканчивая графическим анализом полученных результатов (рис. 23) [31].

Таким образом, прототип ИС “*AKORD*” состоит из следующих элементов:

- **препроцессор** – программный компонент, с помощью которого готовятся входные данные для численного эксперимента (для плоских, осесимметричных и пространственных задач);
- **решатель** – позволяет численно решать задачи, опираясь на данные, подготовленные препроцессором;
- **постпроцессор** – программный модуль, направленный на графическое отображение полученных после работы решателя результатов (пока только для плоских и осесимметричных задач);
- **оболочка** – составной программный интерфейс, объединяющий различные программные компоненты и необходимые данные в единый комплекс;
- **интерфейс обмена данными** – набор правил, определяющих порядок и формат обмена данными между компонентами системы.

Каждый компонент ИС “*AKORD*” может функционировать как отдельная программная единица, так и в составе единого программно-информационного комплекса.

Каждый программный компонент, в свою очередь, представляет собой совокупность программных модулей и необходимых наборов данных. Если следовать известной формуле – *программа = данные + алгоритмы* – то все программные единицы состоят из

пассивного компонента (данные), над которым производятся некоторые действия, и активного компонента (алгоритмы), который, собственно, и производит необходимые операции по обработке данных.

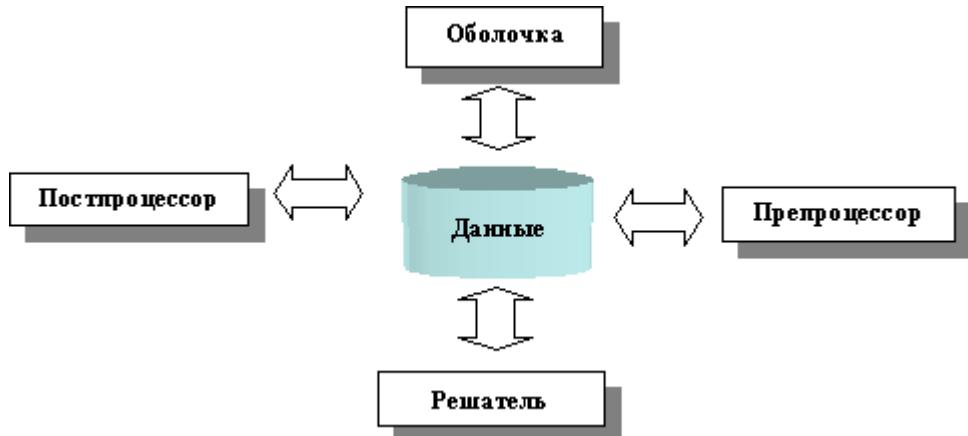


Рис. 23. Общая структурная схема ИС “*AKORD*”

Однако необходимая мощность и гибкость для всестороннего исследования поставленной задачи дается в руки пользователя только в том случае, когда все программные компоненты и соответствующие им наборы данных собраны в единое приложение.

Данные для каждого компонента в составе прототипа ИС представляют собой:

- файлы в текстовом формате, служащие основой обмена данными между программными компонентами;
- двоичные файлы, представляющие собой эквиваленты текстовых файлов (нужны для быстроты обработки);
- файлы в стандартных графических форматах (BMP, PCX, GIF, JPG), являющиеся либо графическими изображениями результатов после обработки постпроцессором, либо результатами каких-нибудь тестовых задач, полученных из других источников;
- файлы в форматах документов (RTF, PDF), содержащие дополнительные элементы описания решаемых задач – постановка задачи, описание теста, блок-схема алгоритма решения и т.д.

Кроме того, в качестве данных в рассматриваемой ИС “*AKORD*” выступают такие же алгоритмы, реализованные в своем формальном представлении посредством языков программирования высокого уровня C++, Pascal, Fortran и языка манипулирования данными SQL. В некотором смысле алгоритмы, а также графиче-

ские файлы результатов, описание задачи и тестов, вспомогательные данные, являются знаниями. Поэтому основой ИС является База Знаний, реализованная при помощи реляционной базы данных (БД), которая размещается на специализированном сервере (в нашем случае это Oracle 7 Server), и программных комплексов, входящих в ее состав.

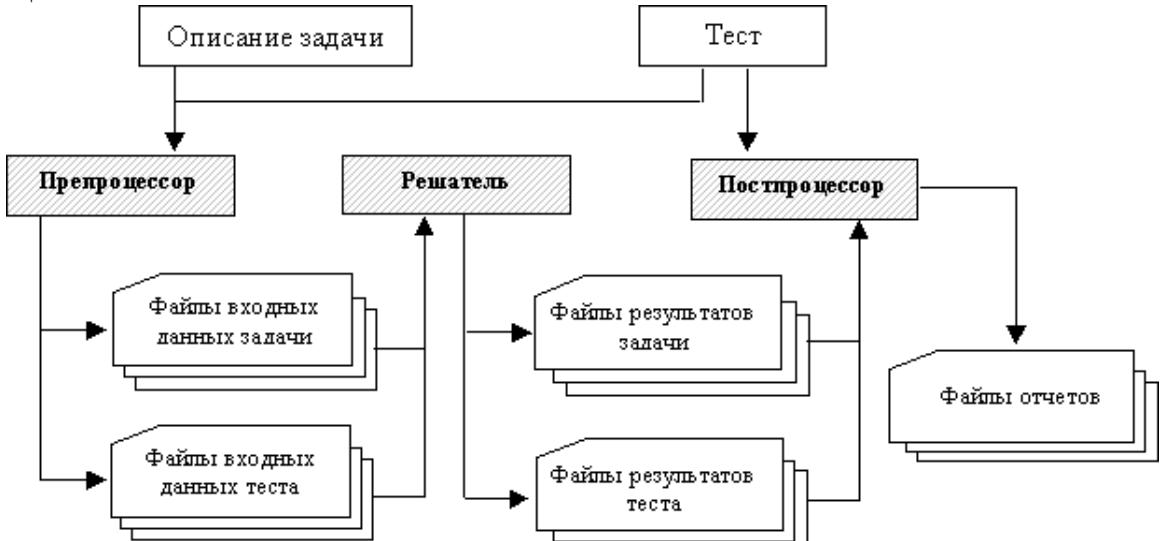


Рис. 24. Схема расчета, принятая в ИС "AKORD"

Под *набором данных* в ИС понимается совокупность всех необходимых данных, обеспечивающих в полном объеме исследование какой-либо конкретной задачи. Поскольку численное исследование задачи, как правило, происходит в соответствии с некоторой заранее определенной схемой, то этот набор данных должен обеспечить как пассивную, так и активную компоненту инструментальных средств на протяжении всего цикла исследования.

Все данные в ИС подчиняются четко определенному интерфейсу, который является связующим звеном между компонентами системы.

В ИС за базовую конструкцию, адекватно описывающую объект исследования (решаемую задачу) и четко следующую логике метода исследования (численному эксперименту) принята специальная логическая конструкция - *схема расчета* (рис. 24), позволяющая пользователю:

- сформировать постановку задачи (тип задачи, название задачи и ее описание);
- сгенерировать плоскую или поверхностную сетку для аппроксимации границ рассматриваемой области решаемой задачи;

- задать необходимые физические, временные и вспомогательные параметры, используемые в дальнейшем решателем;
- получить приближенное решение исследуемой задачи при наличии выбранных данных и алгоритмов;
- сформировать результаты в готовой для последующей обработки форме;
- провести тестовый расчет для сравнения результатов;
- получить графическое представление и провести качественный анализ результатов вычислений;
- сохранить во внешней памяти или вывести в виде твердой копии графические и текстовые отчеты на основе полученных результатов.

В соответствии с ней отправным элементом проведения вычислительного эксперимента с помощью ИС является определение логической структуры расчета. Он должен включать в себя описание задачи, описание тестов (если это требуется), различных данных, обеспечивающих целостность прохождения всего цикла численного исследования, и набор компонент системы, осуществляющих автоматизацию различных стадий решения задачи.

Препроцессор на основании описания задачи в интерактивном режиме готовит входные данные для расчета, сохраняет их в специальных текстовых файлах и передает их в дальнейшем для обработки решателю.

Решатель, реализующий тот или иной численный метод, формирует файлы результатов. Результатом может служить либо приближенное решение исходной задачи, либо решение тестовой задачи (либо для нее также готовятся входные данные с помощью препроцессора, либо ее результаты уже известны).

Постпроцессор принимает на вход файлы результатов и на их основе формирует те или иные отчеты, которые затем анализируются исследователем. На вход препроцессора можно передать готовые файлы, например, таблицы или графики, отражающие решение другой задачи или теста.

Вся информация с описанием задачи, описанием теста, входные данные для проведения непосредственно расчета, результаты и графические отчеты представляют собой элемент записи о данном расчете в специализированной базе данных расчетов. Эта база данных может находиться либо на рабочей станции пользователя (для

(для обеспечения непрерывного доступа к данным), либо в хранилище данных (ХД), расположенном на выделенном сервере под управлением СУБД.

Таким образом, полный цикл решения задачи обеспечивается следующими данными:

- файлы с описанием постановки задачи, входных данных решателя, результатов численного решения и отчетов;
- алгоритмы препроцессора, решателя, постпроцессора;
- программные модули, обеспечивающие занесение, редактирование и изменение данных в БД и ХД расчетов.

Совокупность перечисленных данных и знаний и составляет набор данных, связанный с этой задачей. Этот набор данных поддерживается на рабочей станции пользователя, а при необходимости может быть помещен в БД расчетов для дальнейшего хранения.

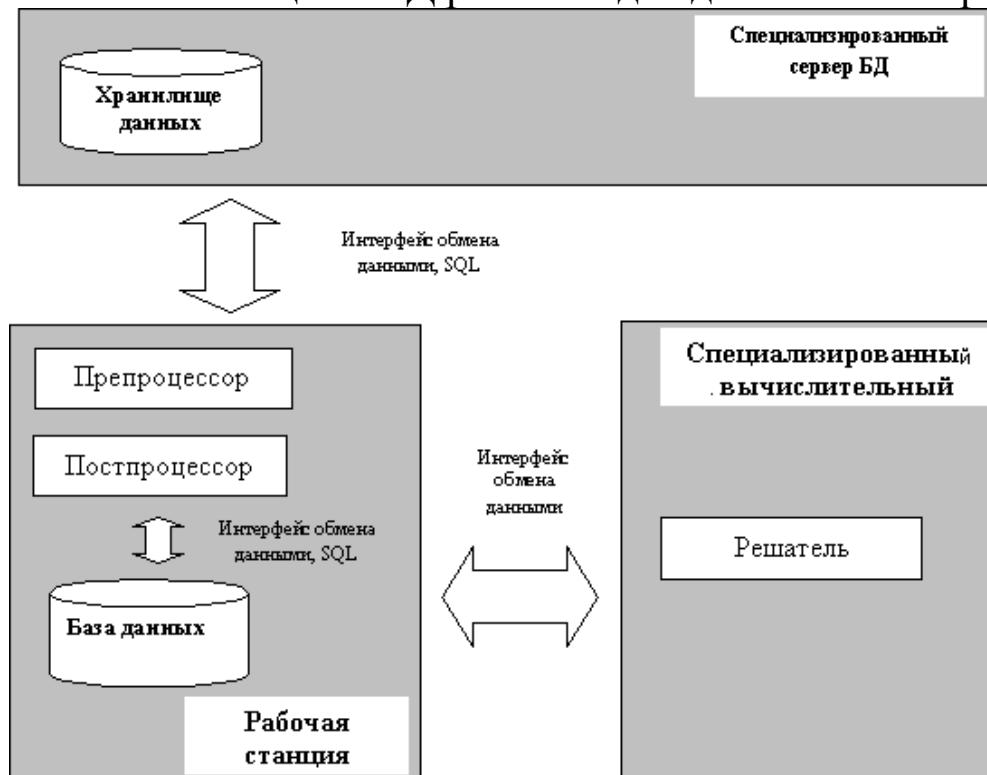


Рис. 25. Архитектура ИС

Из выше сказанного становится понятно, что ИС должна дать в руки пользователя гибкую инструментальную систему, поддерживающую целостность данных и информационных связей между всеми приложениями на протяжении всех этапов численного исследования решаемой задачи.

В дополнение к этому, система поддерживает архитектуру клиент – сервер (рис. 25), что позволяет располагать определенные его части на различных компьютерах в сети, использовать различные

вычислительные и накопительные мощности (т.е. пакет может быть инсталлирован в качестве распределенной системы, что, несомненно, является его дополнительным преимуществом).

5.2. Логическая и физическая модели данных

При проведении численных расчетов для исследуемой задачи на жестком диске организуются большие объемы результатов, которые могут по мере накопления занимать значительный объем дискового пространства. Сбой в системе или некорректное вмешательство может привести к несанкционированному удалению результатов или их изменению. Этого можно избежать, если системой предусмотрено резервное архивирование данных. Но даже в этом случае дисковое пространство не увеличивается, а уязвимость данных уменьшается незначительно. В настоящее время продвинутые системы, позволяющие проводить полное исследование поставленной задачи и на выходе получать множество результатов, используют базы данных предназначенные формировать, редактировать, длительное время хранить и использовать необходимые для проведения исследований численные расчеты.

Если всю «справочную» информацию поместить на удаленный сервер, то система, находящаяся на компьютере пользователя, будет разгружена от больших объемов данных. При этом использование языка SQL [23] (структурированный язык запросов), фактически являющегося стандартным почти для всех современных СУБД, обеспечивает мощное и удобное средство для доступа к удаленным и локальным данным.

С другой стороны, разделение информации по независимым системам позволяет использовать вычислительные ресурсы специализированных систем удаленных вычислений и внешнюю память специализированных хранилищ данных.

База данных призвана хранить всю представляющую интерес информацию в одном или нескольких местах, причем таким способом, который заведомо исключает ненужную избыточность, а вероятность сохранения противоречивых данных минимизирует.

При создании прототипа ИС “*AKORD*” особое внимание уделялось проектированию семантической модели данных, описывающей предметную область и используемой системой для проведения вычислительного эксперимента. Основой этой модели, есте-

ственno, служит схема расчета. Кроме того, при разработке базы данных ИС учитывались следующие принципиальные моменты.

- Способность БД адекватно соответствовать описываемому объекту из предметной области. Перед проектированием базы проводились обширные исследования требований потенциальных пользователей к функционированию пакета и необходимых информационных зависимостей.
- Гарантия непротиворечивости и целостности данных. При проектировании отношений определялись их атрибуты и некоторые правила, ограничивающие возможность ввода пользователем неверных значений. Для верификации данных перед непосредственной записью их в таблицу СУБД осуществляет вызов правил модели данных и, тем самым, гарантируется сохранение целостности информации.
- Естественное, легкое для восприятия структурирование информации. Качественное построение базы позволяет делать запросы к данным более “прозрачными” и легкими для понимания. Следовательно, снижается вероятность внесения некорректных данных и улучшается качество сопровождения базы.

На основании этих требований для реализации проекта была выбрана СУБД фирмы Oracle. Однако проект может быть без особыго труда перенесен на платформу любой СУБД, поддерживающую в качестве стандартного интерфейса манипулирования данными язык SQL.

При проектировании системы были определены следующие информационные потребности:

- возможность системы работать с базой данных - введение базы данных не меняет структуру всей ИС, а только дополняет ее новыми возможностями;
- тип данных, используемых модулями ИС - текстовые файлы, файлы документов с описанием постановки задачи и тестов, исполняемые коды приложений, графические образы, таблицы, файлы входных параметров и результатов;
- управление базой данных через интерфейс оболочки - редактирование, ввод новой информации, обновление данных, извлечение необходимых наборов данных с возможностью переноса их на рабочее место пользователя;
- описание предметной области одной базой данных.

Исходя из этой концепции и используя описанную выше схему расчета, была разработана следующая логическая структура базы данных (Приложение 3). Основной логической единицей данных здесь является расчет, который привязан к решаемой задаче. Совокупность расчетов составляет набор результирующих данных, полученных после численного решения задачи. Таким образом, решаемая задача определяется своей постановкой, последовательностью расчетов и набором тестов. Каждый расчет, в свою очередь, полностью определяется набором входных параметров и результатами их обработки решателем.

Физическая модель БД представлена совокупностью отношений, содержащих всю необходимую информацию и объединенных различными связями (функциональными зависимостями). Точной входа в модель является отношение «Задачи», описывающее собственно решаемую задачу и определяемое своим названием и постановкой. Все задачи группируются по типам (отношение «Тип задачи»), например, плоские стационарные или осесимметричные нестационарные задачи. Каждая задача может быть дополнена каким-нибудь тестом со своим описанием, естественно.

Основным связующим звеном физической модели данных является отношение «Расчеты по задаче», которое не содержит в себе собственно данных по каждому расчету (кроме названия), а ее атрибуты ссылаются на ключевые атрибуты других отношений: «Входные данные решателя», «Входные данные теста», «Результаты решателя», «Результаты теста» и других. Таким образом, на ее основе можно получить любую информацию, касающуюся текущего расчета (например, каким решателем были получены результаты, каким препроцессором подготовлены входные данные, использовался ли тест и т.д.).

Часто исследователь проводит большое количество расчетов с целью выявить влияние на получаемое решение какого-либо одного параметра. В таких ситуациях входные данные к решателю меняются незначительно. Однако именно такие исследования при решении тех или иных задач являются наиболее «тонкими» и позволяют определить группы эффективных и несущественных параметров, влияющих на конечное решение. Использование такой методики при проведении численного эксперимента, как правило,

приводит исследователя к необходимости анализировать большие массивы информации.

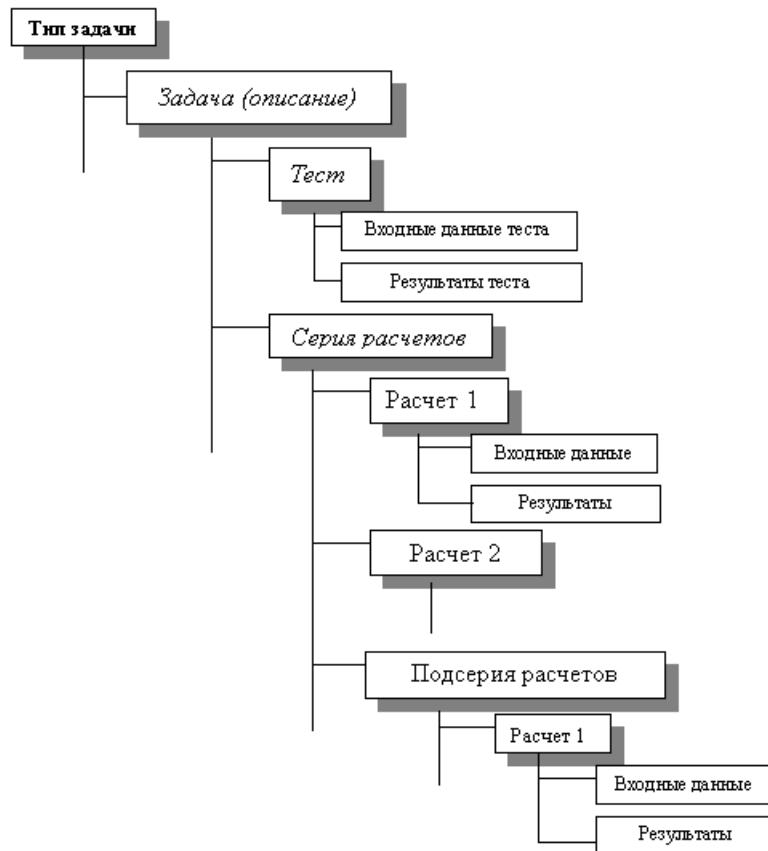


Рис. 26. Структура серии расчетов

Для учета этой специфики численного эксперимента в модель данных добавлено дополнительное отношение («Серии расчетов»), которое позволяет объединить результаты расчетов в некоторую последовательность по признаку, выбранному пользователем (рис. 26). Поскольку серии расчетов представляют в своей основе иерархические данные, то для моделирования таких зависимостей используется рекурсивный тип связи (соединение отношения «само с собой»).

С каждой задачей связан определенный набор инструментов для исследования: препроцессор, решатель и постпроцессор.

Набор таблиц, организованных в соответствии с описанной моделью, приведен в конце книги (Приложение 4).

Все отношения в логической модели данных для прототипа ИС «*AKORD*» удовлетворяют, по крайней мере, требованиям четвертой нормальной формы (4NF) [68]. Результирующие таблицы физической модели данных сгенерированы так, чтобы учитывалась осо-

бенность современных СУБД по автоматическому поддержанию целостности данных.

5.3. Оболочка

Оболочка прототипа автоматизированной информационной системы “*AKORD*” [12] предоставляет пользователю удобный интерактивный интерфейс при реализации всех основных этапов численного решения задач, начиная с ее описания, и заканчивая графическим анализом полученных результатов. Она обеспечивает согласованность наборов данных для исследования конкретной задачи и корректный вызов всех необходимых компонент ИС при проведении расчетов. Кроме этого, оболочка синхронизирует данные как на стороне сервера БД, выполняющего функции хранилища результатов расчетов, так и на стороне рабочей станции пользователя, непосредственно выполняя действия по подготовке процесса численного решения задачи и последующей обработки результатов [1]. Хотя любые приложения, входящие в состав ИС, можно использовать независимо, но функции по согласованности и непротиворечивости данных, необходимых для проведения расчетов, выполняет именно оболочка. Другими словами, исследователь, отказавшись от использования оболочки, лишается возможности использовать хранилище данных для систематизации и архивирования готовых результатов и отчетов, а локальная база данных, содержащая сведения о текущем расчете, не поддерживается в целостном состоянии.

Оболочка состоит из блоков связанных программным интерфейсом в единое приложение, которые обеспечивают процесс численного решения задачи (рис. 27). Оболочка полностью поддерживает логическую структуру схемы расчета (рис. 24).

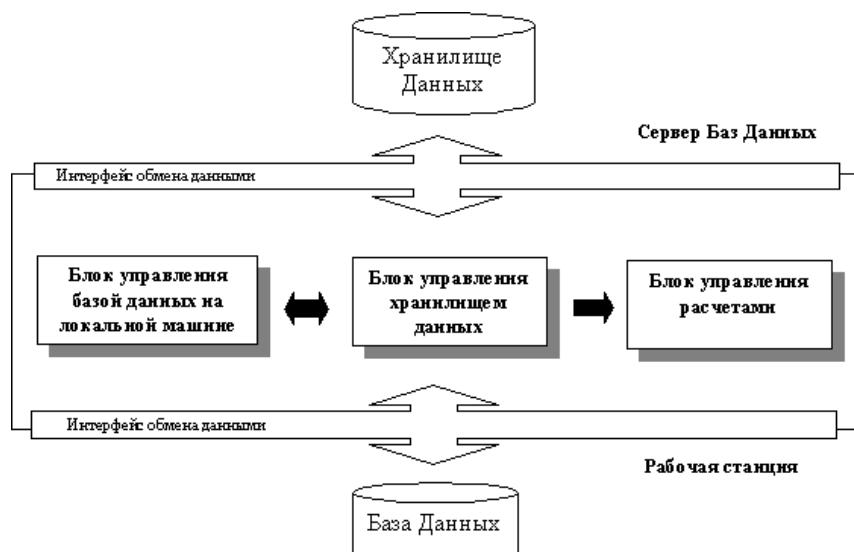


Рис. 27. Принципиальная схема построения оболочки

Работа над новым расчетом начинается с формирования в блоке управления БД на локальной машине описания новой **задачи** (задается тип задачи, название задачи, ее постановка в виде файла формата .rtf). Каждой новой задаче назначаются необходимые компоненты ИС, такие, как препроцессор, постпроцессор и решатель. Затем подготовленные данные передаются в блок управления расчетами, где при помощи компонентов пакета создается плоская или поверхность сетка для аппроксимации границ рассматриваемой области решения, задаются физические, временные и вспомогательные параметры, вычисляется искомое решение в исследуемой области, полученные результаты обрабатываются и представляются в графическом виде.

Исходя из практики проведения численного эксперимента, можно выделить три основных типа проводимых расчетов.

Стандартный тип расчета определяется требованием – получить численное решение задачи с помощью решателя и, если это требуется, обработать полученные результаты. В драйвер расчета передаются требуемые для всего процесса исследования параметры, он же, в свою очередь, снабдит ими связанные с задачей приложения.



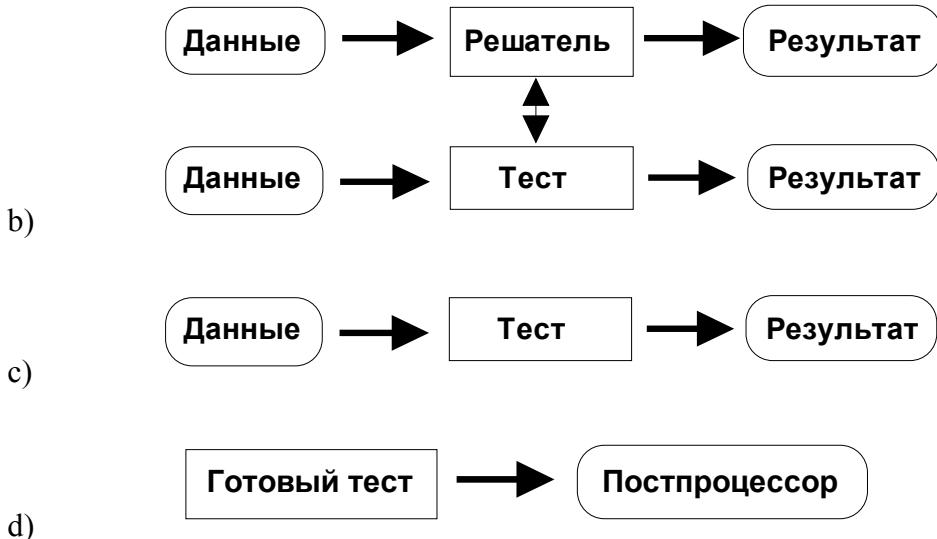


Рис. 28. Возможные типы расчетов

Схема расчета для этого типа представлена на рис. 28,а:

- подготавливаются входные данные для решателя (используя препроцессор или «вручную»);
- проводится расчет;
- на выходе получаются результаты, которые анализируются постпроцессором.

Стандартный + Тест. Этот тип расчета позволяет параллельно со стандартным расчетом получить решение тестовой задачи для последующего сравнения результатов. При этом сразу готовятся входные данные для задачи и теста, затем проводится расчет соответствующим решателем. Полученные результаты анализируются с помощью постпроцессора. Тест так же может быть запущен отдельно со своим препроцессором, решателем и постпроцессором.

Схема расчета для этого типа следующая (рис. 28,б):

- подготавливаются входные данные препроцессором для решателя и теста;
- проводится расчет обеих задач либо одним и тем же решателем, либо по отдельности;
- на выходе получаются результаты, которые совместно анализируются постпроцессором.

Тест - тип расчета проводится только для решения тестовой задачи. При этом существует два вида использования тестов:

- подготавливаются входные данные для теста, которые решаются выбранным решателем;
- существует готовый результат в виде текстовых или графических файлов, таблиц и тому подобное.

В случае (1) пользователь выбирает тип задачи, подготавливает файлы входных данных для теста с помощью соответствующего препроцессора, решателем проводит расчет и полученные результаты анализирует постпроцессором. Этот вариант расчета полностью повторяет действия при стандартном типе расчетов (рис. 28,с).

В случае (2) пользователю нет необходимости выполнять выше перечисленные действия. Достаточно выбрать необходимый тест и посмотреть результаты (рис. 28,д).

Текущий расчет (или серия расчетов) затем может быть помещен в хранилище. Перемещение объектов из хранилища в базу данных и обратно осуществляется при помощи блока управления хранилищем данных.

Перечисленные блоки являются основными в структуре оболочки. Кроме них для дополнительной обработки информации могут быть задействованы вспомогательные блоки.

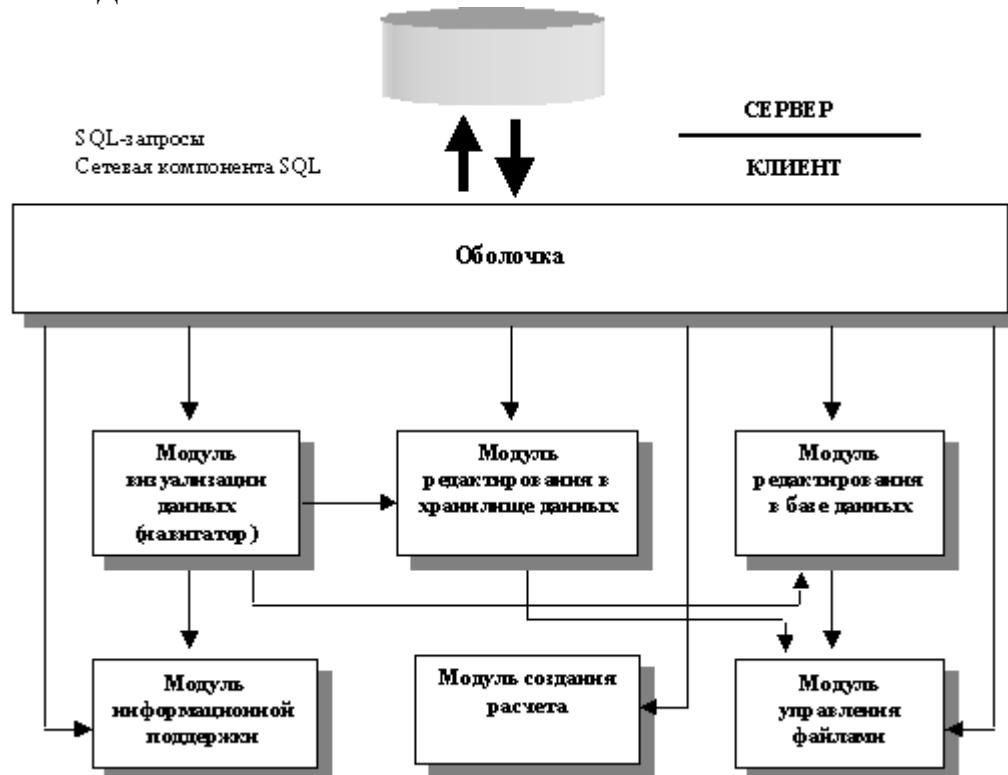


Рис. 29. Модульная структура оболочки

Оболочка имеет модульную структуру (рис. 29), которая основана на описанной выше принципиальной схеме (рис. 27). Все мо-

дули вызываются из оболочки, которая, в свою очередь, является обобщающим модулем, связывающим отдельные компоненты в единое приложение.

Функциональное назначение и основные возможности всех модулей оболочки представлены ниже.

Модуль визуализации данных (навигатор) отображает информацию о состоянии данных на сервере и клиентском месте. Предоставляет полный функциональный интерфейс для работы как с объектами в таблицах базы данных, так и с объектами на локальной машине.

Модуль редактирования в хранилище данных работает с записями на сервере БД, выполняя стандартные функции:

- извлечение – извлекает объект из хранилища данных;
- добавление – позволяет добавить (обновить) записи в хранилище;
- удаление – удаляет объект из хранилища.

Модуль редактирования в локальной базе данных позволяет пользователю напрямую работать с объектами, которые располагаются в базе данных на локальной машине. Выполняет следующие функции:

- создание объекта на локальной машине - позволяет добавить в рабочую базу данных новый объект (описание задачи, приложения, тесты и т.д.);
- удаление - используется для удаления объекта из локальной базы данных.

Модуль информационной поддержки позволяет через специальный интерфейс получить текущие данные о выбранном объекте.

Модуль управления файлами предоставляет интерфейс для работы с информационными файлами, необходимых для работы отдельных приложений. Выполняет операцию копирования файлов на сервер БД или машину пользователя.

Модуль создания расчета используется при подготовке данных для расчета и непосредственного запуска процесса численного решения задачи. Пользователю предлагается интерфейс для ввода необходимых параметров задачи (тип расчета, тип задачи, испрашиваемый препроцессор и т.д.). Также включает в себя функции управления процессом расчета:

- ввод параметров - готовит данные в зависимости от выбранного типа расчета (приведены ниже);
- активизация расчета;
- остановка расчета - прекращает процесс решения;
- просмотр - осуществляет запуск постпроцессора для графического анализа результатов.

Логически все блоки системы сделаны либо в виде одного модуля, либо представляют собой набор модулей, каждый из которых реализует какую-то определенную функцию.

Интерфейс оболочки представлен главным окном, содержащим основное меню, палитрой кнопок быстрого доступа к основным функциям пакета и специальным инструментом манипулирования данными - навигатором.

Программная реализация оболочки для прототипа ИС “*AKORD*” выполнена в среде Windows 95/NT, где доступ к данным (файлам, приложениям) на жестком диске осуществляется через специальный объект – локальную базу данных, а к хранилищу на сервере Oracle - с помощью интерфейса SQL-запросов [12].

5.4. Интерфейсы обмена данными между приложениями

Все составные части прототипа ИС “*AKORD*” обмениваются данными между собой согласно специально разработанному интерфейсу.

Интерфейс обмена данными состоит из нескольких компонент:

- *текстовые и двоичные файлы*, при помощи которых происходит передача данных между приложениями;
- *протоколы обмена* между приложениями и базой данных, позволяющие делать однотипные запросы к хранилищу данных на выделенном сервере или рабочей базе данных на локальной машине пользователя, используя язык SQL [74].

Вся исходная информация для приложений пакета (препроцессор, решатель, постпроцессор) в основном содержится в текстовых файлах. Препроцессор для пространственных задач использует дополнительный двоичный файл с описанием сетки граничных элементов и массивов связности.

Для текстовых файлов разработана специальная структура хранения данных. Эта структура состоит из следующих компонентов:

определение секции данных, значения данных, описание данных.

Все данные в файлах сгруппированы в секции. Каждая секция определяется конструкцией вида:

[ИМЯ_СЕКЦИИ].

Имя секции пишется прописными буквами. Существует предопределенный набор секций, который обрабатывается приложениями системы определенным образом, и специальная секция **[OTHER]**, параметры которой могут быть произвольными и определяются пользователем самостоятельно.

Значения одиночных данных в каждой секции располагаются по одному в каждой строке файла. Массивы данных располагаются построчно.

Вслед за значением может идти описание параметра, отделяемое от него знаком «;» (точка с запятой). Это описание не обрабатывается приложениями ИС.

Ниже приводятся краткие пояснения по содержанию файлов для каждого из существующих в системе приложений. Примеры файлов можно найти в приложении 5.

Препроцессор для плоских и осесимметричных задач

Результатом работы препроцессора является текстовый файл, в котором данные по логическим признакам сгруппированы в следующие секции.

[MAIN] - секция общих параметров. Формат данных:

- название задачи (строка до 80 символов);
- тип задачи (1 – плоская, 2 - осесимметричная);
- общее количество узлов (целое число);
- количество подобластей (целое число);
- количество внутренних узлов (целое число).

[TIME] - секция временных параметров:

- максимальный шаг по времени;
- минимальный шаг по времени;
- время инициализации задачи;
- максимально разрешенное время расчета.

[DOMAIN] - секция параметров, характеризующих область решения:

- ограничение области (0 – безгранична, 1 - ограниченная);

- замкнутость границы (0 - граница не замкнута, 1 – граница замкнута);
- положение нормали (0 - вовнутрь области, 1 - наружи);
- направление нормали (1 – положительное - область при обходе границы остается слева, -1 - отрицательное);
- наличие силы тяжести (0 – нет, 1 – есть).

[PHYSICAL] - секция физических параметров:

- максимальная погрешность расчета;
- давление на поверхности жидкости или в удаленной точке;
- плотность жидкости;
- коэффициент поверхностного натяжения;
- показатель адиабаты (по умолчанию 1,44).

[NODES] - секция параметров, описывающих свойства граничных узлов. Формат данных в строке следующий:

“*n* *x* *y* *code* *value*”,

где *n* - номер узла; *x* - координата *X*; *y* - координата *Y*; *code* - тип граничного условия (0 – условие Дирихле, 1 – Неймана, 2 – специальный тип условия); *value* - значение граничного условия в узле.

[INTERNAL] - секция параметров, описывающих внутренние узлы области (если есть). Формат строки данных:

“*n* *x* *y*”,

где *n* - номер узла, *x* - координата *X*, *y* - координата *Y*.

[REGION] - секция параметров, описывающих границы подобластей. Формат строки данных:

“*N* *n* *nodes*”,

где *N* - номер подобласти; *n* - номер первого узла подобласти; *nodes* - количество узлов в подобласти.

[SPECIAL] – секция, в которой для отдельного узла специально описаны граничные условия. Формат записи:

“*n value*”,

где *n* - номер узла; *value* - значение граничного условия.

[OUTPUT] – секция, описывающая параметры вывода результатов расчета решателем. Каждый решатель может обрабатывать эти параметры, если это необходимо. Формат данных:

- шаг вывода результатов (целое число, 0 – вывод результатов не производится);
- шаг записи промежуточных результатов в файл (значения те же);
- шаг для расчета значений во внутренних точках области;
- шаг для вывода картинки проекции области на плоскость для осесимметричных задач (опционально);
- шаг для вывода картинки расчетной сетки области (опционально);
- шаг для вывода на экран графиков функций во время самого расчета (опционально);
- шаг для выполнения процедуры сглаживания расчетной области (опционально);
- шаг для записи в файл графических окон с картинками, которые были получены во время расчета (опционально);
- очищать окно вывода проекции области перед каждой отрисовкой (“.Т.” – очищать, “.Ф.” – не очищать);
- очищать окно вывода сетки граничных элементов области перед каждой отрисовкой (“.Т.” – очищать, “.Ф.” – не очищать);
- продолжать непрерывный счет при возникновении какой-либо ситуации (разрушение границы области, достижения максимального шага по времени и т.д.). Значение этого параметра регулируется в каждом конкретном решателе;
- Обнулять все временные файлы перед началом расчета (“.Т.” – обнулять, “.Ф.” – не обнулять). Если этого не делать, то расчет начинается с того места, которое было сохранено во временном файле (параметр записи во временные файлы).

[OTHER] – секция, содержащая дополнительные параметры, обрабатываемые решателем. Эта секция параметров не обязательна и присутствует лишь для повышения гибкости системы по обмену данными. Дописывая свои параметры в эту секцию, пользователь может настроить свой решатель на использование специфичных для него данных.

В приложении 5 приводится файл входных данных для решения задачи Рэлея [50] о схлопывании газовой полости единичного радиуса в безграничной идеальной несжимаемой жидкости.

Постпроцессор

В данной версии постпроцессор может работать только с данными типа итерации. Итерация - это набор данных, состоящий из координат точек, к которым привязан набор параметров либо индивидуальных для каждой точки, либо общих для всего набора данных. Итерация всегда привязана к некоторому моменту времени.

Текстовый файл для набора данных типа итераций состоит из заголовка, набора секций, в которых описываются все параметры задачи и собственно самих данных.

Заголовок файла данных содержит две строки. Первая должна содержать одно ключевое слово:

[TEXT_DATA_FILE].

По этому признаку постпроцессор определяет, что файл является файлом постпроцессора. Следующая строка должна содержать конструкцию:

[DATA_FORMAT] TEXT_ITERATION_DATA_COL
или
[DATA_FORMAT] TEXT_ITERATION_DATA_ROW.

Здесь **[DATA_FORMAT]** ключевое слово, а

TEXT_ITERATION_DATA_COL

или

TEXT_ITERATION_DATA_ROW

указывает на то, как данные хранятся в текстовом файле (по столбцам или по строкам).

Далее следуют секции, описывающие параметры задачи. Кажд-

дая секция начинается с идентификатора (слова, не содержащего пробелы и заключенного в квадратные скобки), затем идут строки, содержащие данные в этой секции. Закрывается секция строкой из единственного ключевого слова **[END]**.

Каждая секция не должна содержать пустые или состоящие из одних пробелов строки. Все секции должны располагаться в определенном порядке.

Первая секция **[GLOBAL_DATA_DEF]** описывает глобальные параметры, общие для каждой точки в итерации. Каждая строка секции задает один параметр по следующему формату:

NAME="Название параметра",

где NAME это ключевое слово. Если название параметра не содержит пробелов, то двойные кавычки могут быть опущены. Следует отметить, что время как глобальный параметр явно не описывается, но подразумевается и его значение стоит перед значениями всех остальных глобальных параметров.

Следующая секция **[LOCAL_DATA_DEF]** описывает параметры каждой точки итерации такие, как координаты, компоненты вектора скорости и т.д. По структуре она аналогична предыдущей секции.

Секция **[DATA_PARAM]** описывает параметры данных. Секция состоит из двух строк, каждая из которых содержит одно число: первое - количество итераций, а второе - количество точек в каждой итерации.

Секция определения многосвязных областей **[REGION_DEF]**. Количество областей равно числу строк в данной секции. В каждой строке указывается номер точки, с которой начинается область. Заканчивается область точкой с номером, указанной в следующей строке в качестве начальной для следующей области. Если строка последняя, то область заканчивается последней точкой итерации. Точки нумеруются с нуля. Таким образом, если в расчете используется односвязная область, то данная секция содержит всего одну строку:

[REGION_DEF]

0

[END]

Если в расчете использовалась двухсвязная область (например, определяемая восемью точками, где первые четыре точки описы-

вают первую область, остальные четыре - вторую область), то эта секция должна содержать две строки:

[REGION_DEF]

0

4

[END]

Все записи вне секций считаются комментариями и игнорируются постпроцессором.

Данные типа итерации оформлены в виде блоков. Первая строка блока начинается с ключевого слова [ITERATION], затем идут значение времени и значения глобальных параметров (через пробел), если они присутствуют. Со следующей строки начинаются собственно данные для точек итерации. Их формат зависит от слова, указанного после ключевого слова [DATA_FORMAT] в заголовке. Если указано TEXT_ITERATION_DATA_COL, то данные будут располагаться по столбцам (одна точка итерации будет представлена одной строкой), например:

0.001	1	0	1	1
0.001	1	0	1	1
0.001	1	0	1	1

...

Если же указано слово TEXT_ITERATION_DATA_ROW, то данные будут располагаться построчно (записей будет столько, сколько параметров связано с точкой). При этом каждый столбец будет описывать одну точку итерации, например:

0.001	0.001	0.001	...
1	1	1	...
0	0	0	...
1	1	1	...
1	1	1	...

Пример текстового файла итераций приведен в приложении 5.

Г л а в а 6

ПРИМЕРЫ ЗАДАЧ, РЕШАЕМЫХ С ПОМОЩЬЮ ППП “AKORD”

В данной главе приводятся расчеты, выполненные на основе описанных выше методов и алгоритмов решения задач со свободными границами. Глава делится на два раздела.

В первом разделе приведено решение тестовых задач, которые показывают работоспособность предложенных алгоритмов и могут быть полезными при тестировании различных численных алгоритмов.

Во втором разделе приведены решения конкретных задач, ранее описанных в работах [6, 7, 16, 17, 18, 33, 34].

6.1. Тестовые расчеты

Тест №1. Решение уравнения Лапласа в заданной области

A) Формула Коши. Найти решение уравнения Лапласа в области $D = \{0 \leq x \leq 2\pi; -1 \leq y \leq 0,5 \sin(x)\}$. На дне и вертикальных стенках ставится условие непротекания $\psi(x, y) = 0$, а на свободной поверхности условие $\phi(x, y) = -\cos(x) \operatorname{ch}(y + 1)$ (правая часть здесь является гармонической функцией). Численные значения функции тока $\psi(x, y)$, найденные комплексным методом граничных элементов, сравниваются с точным решением: $\psi^T(x, y) = \sin(x) \operatorname{sh}(y + 1)$. В таблице 6 приведена относительная погрешность

$\varepsilon_1 = \max |\psi^T - \psi^Q| / \max |\psi^T|$ точного и численного значений функции ψ от числа узлов N по границе с указанием числа узлов на свободной поверхности жидкости N_g . Вектор скорости (V_x, V_y) в каждом узле на свободной поверхности вычисляется изложенным в работе [14] методом и сравнивается с точным значением

$$V_x^T = \frac{\partial \phi}{\partial x} = \sin(x) \operatorname{ch}(y+1), \quad V_y^T = \frac{\partial \phi}{\partial y} = -\cos(x) \operatorname{sh}(y+1).$$

В третьей и четвертой колонках приведены значения относительных погрешностей $\varepsilon_2 = \max |V_x^T - V_x^Q| / \max |V_x^T|$, $\varepsilon_3 = \max |V_y^T - V_y^Q| / \max |V_y^T|$.

Таблица 6
Относительные погрешности вычисленных значений от числа узлов

$N(N_g)$	ε_1	ε_2	ε_3
72(30)	5,5E-3	2,0E-3	1,6E-2
145(60)	1,3E-3	6,5E-4	9,3E-3
290(120)	3,1E-4	6,2E-4	8,7E-3
580(240)	7,6E-5	6,0E-4	5,7E-3

B) Формула Грина. Аналогично проверяется точность метода на основе формулы Грина. Отличие состоит в том, что задача решается в терминах потенциала скоростей ϕ и его нормальной производной ϕ_n . На боковых границах и на дне ставится условие непротекания: $\phi_n = 0$. Численные значения нормальной производной ϕ_n^Q , найденные методом граничных элементов, сравниваются с точным решением:

$$\phi_n^T(x, y) = \frac{\cos(x)}{\sqrt{1 + \cos(x)^2}} (\sin(x) \operatorname{ch}(y+1) - \operatorname{sh}(y+1)).$$

В таблице 7 приведена относительная погрешность $\varepsilon_4 = \max |\phi_n^T - \phi_n^Q| / \max |\phi_n^T|$ точного и вычисленного значений

функции ϕ_n . В третьей и четвертой колонках приводятся величины, аналогичные приведенным в таблице 6.

Таблица 7

Относительные погрешности вычисленных значений от числа узлов

$N(N_g)$	ε_4	ε_2	ε_3
72(30)	8,4E-03	4,1E-03	6,6E-02
145(60)	2,2E-03	8,4E-04	9,9E-03
290(120)	5,4E-04	7,2E-04	9,3E-03
580(240)	2,2E-04	7,0E-04	6,4E-03

В таблице 8 приведены значения чисел обусловленности $cond(A)$ матриц КМГЭ МГЭ, относительное время $T_1 = T(A)_{\text{КМГЭ}}/T(A)_{\text{МГЭ}}$, затраченное на построение матриц, и время $T_2 = T(X)_{\text{КМГЭ}}/T(X)_{\text{МГЭ}}$, требуемое для решения системы уравнений $X = A^{-1}F$.

Сравнительно небольшое число обусловленности при решении задачи с помощью КМГЭ объясняется тем, что в основе этого численного метода лежит уравнение Фредгольма второго рода.

Таблица 8

$N(N_g)$	число обусловленности $cond(A)$ формулы Коши / Грина	относительное время T_1 построения матрицы A	относительное время T_2 решения системы $AX = F$
72(30)	3,4 / 96	1,23	1,80
145(60)	3,6 / 201	1,25	1,08
290(120)	4,3 / 393	1,35	1,00
580(240)	5,7 / 803	1,38	1,01

Из приведенных таблиц видно, что оба метода обладают достаточной точностью, но в целом, метод граничных элементов, реализованный на основе интегральной формулы Коши, является более точным и имеет матрицу с лучшими свойствами. Время, затраченное на построение матрицы A и на решение системы $AX=F$, практически одинаково.

Этот пример является хорошим тестом, так как его можно интерпретировать как расчет на одном временном шаге задачи о движении жидкости в прямоугольном бассейне.

Тест №2. Тестирование итерационного алгоритма

В качестве тестового примера рассмотрим плоскую задачу о течении тяжелой жидкости со свободной границей вдоль твердого

дна с расположенным на нем, цилиндрическим выступом радиуса R (рис. 39).

С целью проверки работоспособности предлагаемых в главе 3 алгоритмов был использован тест, предложенный в работе [35]. Согласно этой методике требуется найти линию тока $\psi=1$ с уравнением

$$1 - y(1 - R^2 / \rho^2) = 0,$$

по распределению скоростей на ней из аналитического решения об обтекании цилиндра безграничным потоком идеальной жидкости

$$u = 1 + \frac{R^2}{\rho^2} - \frac{2R^4}{\rho^4}; \quad v = -\frac{2yR^3}{\rho^4}; \quad \rho^2 = R^2 + y^2.$$

Отклонение свободной границы от точного решения в зависимости от числа точек на границе в результате пяти итераций приведены в табл. 9

Таблица 9
Погрешность вычисления свободной границы от числа узлов

$N(N_g)$	$\max y^T - y^\psi $	
	<i>МГЭ</i>	<i>КМГЭ</i>
72(30)	2,5E-3	1,3E-3
145(60)	9,3E-4	7,2E-4
290(120)	7,5E-4	5,4E-4
580(240)	5,3E-4	3,2E-4

Тест №3. Движение уединенной волны по ровному дну

Тест проводится на решении нестационарной задачи о движении уединенной волны амплитуды $A=0,5$ по бассейну постоянной глубины $H=1$. В этом тесте важным является то, что уединенные волны в процессе движения не изменяют амплитуду и скорость, сохраняют форму и полную энергию. Для расчета была выбрана область $D = \{-15 \leq x \leq 15; -1 \leq y \leq y_0\}$, где y_0 описывает уединенную стационарную волну. На границе области взято 350 элементов, из них 200 на свободной поверхности. Вершина волны при $t=0$ находилась в точке $x=-5$, $y=0,5$. Расчет проводился до момента безразмерного времени $t = 8,28$, когда вершина волны перешла в точ-

ку с абсциссой $x=5$. К этому моменту волна прошла путь равный, 1,3 длины волны, определяемой длиной отрезка по оси x , на котором выполняется условие [67]: $\operatorname{Im} z(t) \geq 0,01A(t)$, $z \in C_1$.

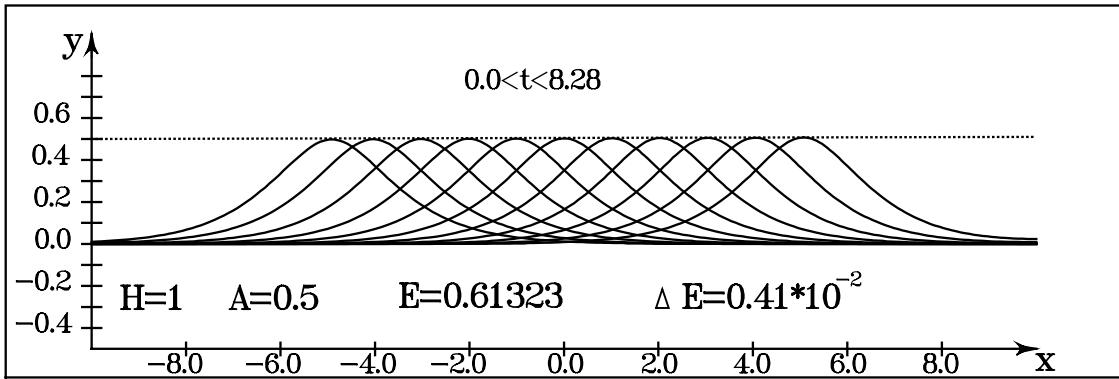


Рис. 30. Движение солитона амплитуды $A=0,5$ по ровному дну

На рис. 30 показаны профили свободной границы для нескольких моментов времени и процент отклонения полной энергии. Отсутствие диспергирующего хвоста из волн малой амплитуды позади основной волны объясняется достаточно точным заданием начальной поверхности солитона и распределением потенциала на ней, полученным на основе численного решения стационарной задачи об уединенной волне [18]. Использование в качестве начальных условий известных приближений уединенных волн [88] дает заметный диспергирующий след. Это обстоятельство изучено в работе [7]. Кроме того, выполнение закона сохранения полной энергии и отсутствие диспергирующего следа позволяет судить о применимости метода Эйлера с выбором шага по времени для решения нелинейных нестационарных задач со свободной поверхностью.

Данный тест дополняется расчетом взаимодействия уединенной волны амплитуды $A=0,4$ с вертикальной стенкой, имеющей абсциссу $x=5$. Остальные параметры задачи были взяты из предшествующего теста. На рис. 31 приведены профили свободной поверхности для нескольких моментов безразмерного времени ($t = 0$, $y_{\max} = 0,4$ - первоначальная форма солитона; $t = 8,91$; $y_{\max} = 0,95$ - форма свободной поверхности в момент максимального заплеска; $t = 19,12$; $y_{\max} = 0,392$ - форма восстановленного солитона, вершина которого в этот момент времени находилась в точке с абсциссой $x=-7,5$).

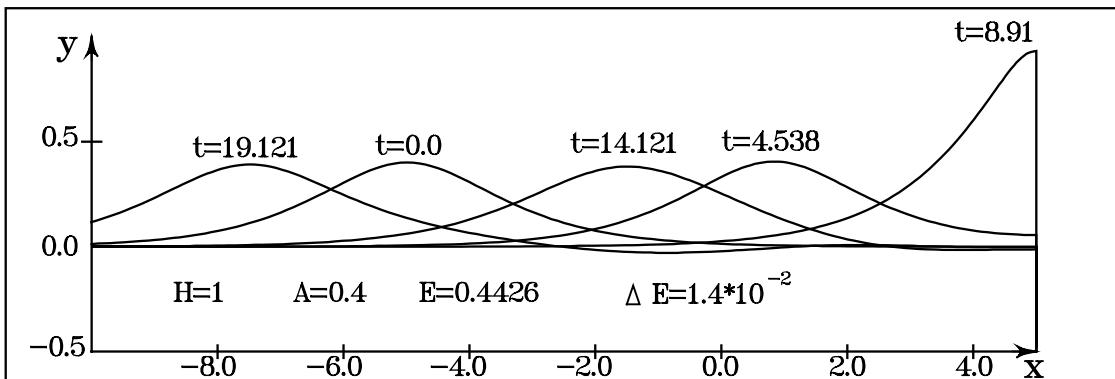


Рис. 31. Тестовый расчет по накату солитона амплитуды $A=0,4$ на вертикальную стенку

Тест №4. Движение солитона над прямоугольным уступом

Для исследования был выбран солитон амплитуды $A = 0,18$. В начальный момент времени $t = 0$ гребень волны имел абсциссу $x_G = 12,5$, уступ располагался в точке $x_y = 25$, бассейн имел длину $L = 90$. Глубина канала $H = 1$, высота уступа $H_y = 0,5$.

Геометрия области течения представлена на рис. 32. Расчеты проводились с различным числом узлов. Приведенные на рис. 33 расчеты выполнялись на "сетке", имеющей 396 узлов, из них 300 на свободной поверхности.

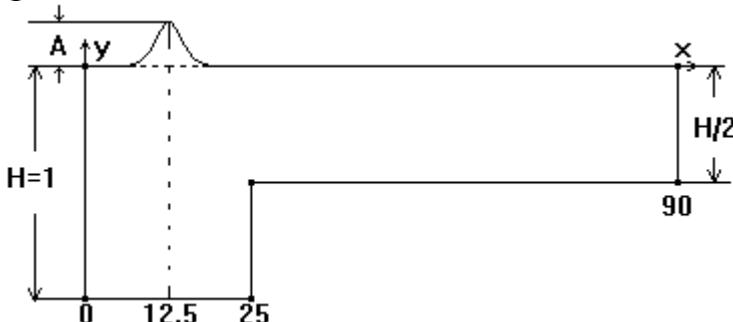


Рис. 32. Геометрия в задаче наката солитона на уступ

В данной задаче все угловые узлы сетки рассматривались как двойные (на рис. 32 они выделены точками). Расчеты, представленные на рис. 33,а, показывают начальную фазу наката волны на уступ и формирование двойного горба. На данном рисунке приведена кривая 3, на которой передний фронт падающей волны имеет легкую выпуклость. Эта выпуклость зарождается в момент подхода переднего фронта к границе уступа и движется против движения солитона, т.е. как бы перекатывается по нему, в результате чего формируется отходящая от солитона первая волна, так называемый дисперсионный след.

При дальнейшем движении волны по каналу (рис. 33,в) ее форма трансформируется, солитон увеличивается по амплитуде, и от него отходит четко сформировавшийся второй солитон (кривая 1,6), бегущий вслед за первым и отстающий от него в силу меньшей амплитуды, а следовательно, и скорости.

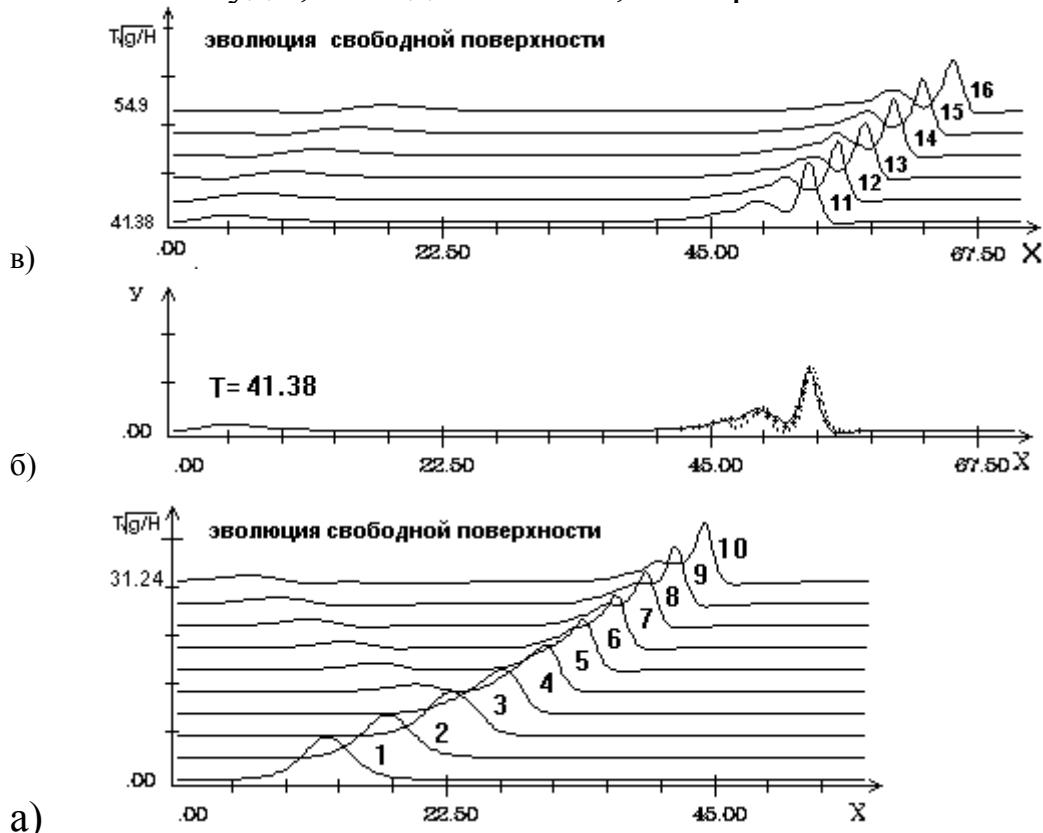


Рис. 33. Движение солитона над уступом. Сравнение с результатами работ [71, 104]

На рис. 33,б приведено сравнение с результатами расчетов из работы [71] и результатами эксперимента из работы [104]. Данное сравнение показывает качественное и количественное совпадение результатов и высокую точность применяемого метода.

Тест №5. Задача обтекания профиля Жуковского

Данная задача имеет аналитическое решение и является хорошей тестовой проверкой численного метода.

Профиль Жуковского может быть задан параметрически в виде [79]:

$$x(t) = \frac{c(c^2 + b^2 + 1)}{2(c^2 + b^2)} - 1, \quad y(t) = \frac{c(c^2 + b^2 - 1)}{2(c^2 + b^2)}, \quad 0 \leq t \leq 2\pi,$$

где

$$c = R \cos(t - \gamma) - d \cos \gamma; b = R \sin(t - \gamma) + d \sin \gamma + h; R = \sqrt{1 + h^2} + d; \\ \gamma = \operatorname{arctg} h.$$

Параметры d и h характеризуют толщину и искривленность профиля. Скорость потока на профиле определяется аналитически в параметрическом виде:

$$V_s(t) = \frac{V_\infty R \sin(\beta + \gamma - t) + \Gamma / 2\pi}{\sqrt{(x'(t))^2 + (y'(t))^2}},$$

где β - угол атаки; $\Gamma = -2\pi RV_\infty \sin(\beta + \gamma)$ - циркуляция скорости вдоль контура профиля. Численное значение циркуляции определяется по формуле [25]

$$\Gamma = \int_C V_s ds.$$

Коэффициенты подъемной силы F_y и силы сопротивления F_x вычисляются следующим образом:

$$F_y = - \int_C V_s^2 \sin \alpha(s) ds, \quad F_x = \int_C V_s^2 \cos \alpha(s) ds.$$

Дополнительным критерием правильности численных расчетов может служить условие $F_x = 0$ - парадокс Даламбера в случае идеальной жидкости.

Коэффициент момента подъемной силы M относительно осткой кромки с координатами (x_0, y_0) может быть вычислен по формуле

$$M(x_0, y_0) = - \int_C V_s^2 ((x - x_0) \sin \alpha(s) + (y - y_0) \cos \alpha(s)) ds.$$

На рис. 34 представлены форма профиля и графики распределения квадрата касательной скорости V_s^2 по профилю. Расчеты проводились при следующих значениях параметров:

- а) $-\beta = 0^0; d = 0,2; h = 0,2;$ б) $-\beta = 0^0; d = 0,2; h = 0,5;$ в) $-\beta = 15^0; d = 0,2; h = 0;$
 г) $-\beta = 15^0; d = 0,2; h = 0,2;$ д) $-\beta = 15^0; d = 0,2; h = 0,5.$

Численные расчеты квадрата скорости по профилю соответствуют аналитическим расчетам с графической точностью. На всех графиках видно, что квадрат касательной скорости при обходе крыла, принимает большие значения на нижней стороне. Циркуляция скорости по профилю была отрицательной, а подъемная сила положительной.

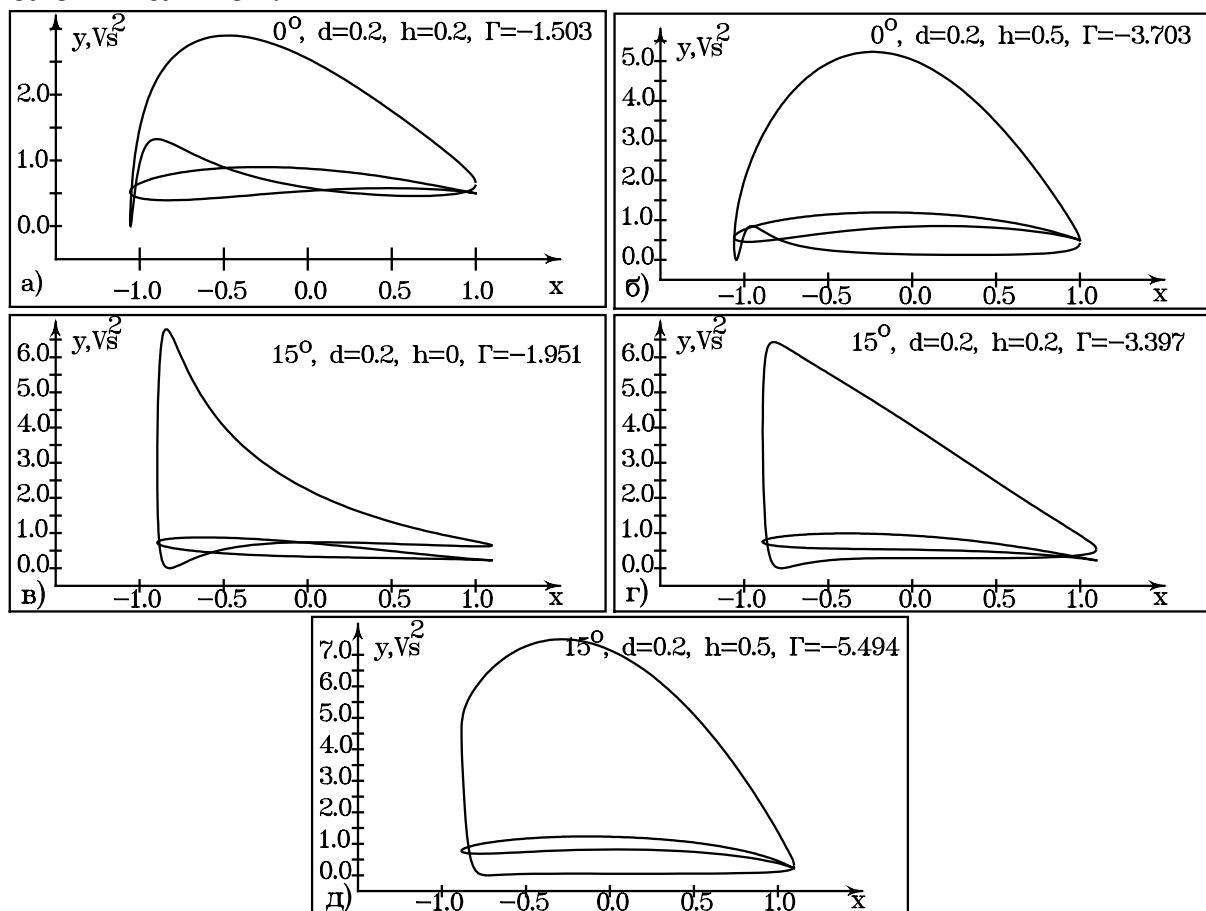


Рис. 34. Распределение квадрата касательной скорости по профилю Жуковского

В табл. 10 приведены значения гидродинамических характеристик в зависимости от угла атаки, толщины и искривленности профиля (Γ -точное значение циркуляции; Γ_{200} - численное значение циркуляции, 200 – количество узлов по границе профиля; $E_2 = \max |V_s^T - V_s^U| / \max |V_s^T|$ - относительная погрешность, V_s^T - точное значение, V_s^U - численное значение функции, полученное КМГЭ; F_x , F_y - коэффициенты силы сопротивления и подъемной

силы, M - момент подъемной силы относительно острой кромки профиля).

Приведенные значения F_x близки к нулю, а относительная погрешность найденных значений V_s незначительна, что свидетельствует о высокой точности разработанного метода.

Таблица 10
Значение гидродинамических характеристик профиля

β , град	d	h	Γ	Γ_{200}	E_2	F_x	F_y	M
0	0,2	0,2	-1,503	-1,489	0,004	0,0002	2,978	-3,318
0	0,2	0,5	-3,703	-3,671	0,007	0,001	7,34	-8,111
15	0,2	0,0	-1,951	-1,947	0,002	0,001	3,893	-5,912
15	0,2	0,2	-3,397	-3,371	0,007	0,002	6,739	-8,886
15	0,2	0,5	-5,494	-5,446	0,009	0,003	10,887	-12,744

Тест №6. Схлопывание сферической газовой полости (задача Рэлея)

Приведем решение, найденное Рэлеем [103] в 1917 г., при следующих предположениях: жидкость несжимаемая, форма пузыря остается сферической во все моменты времени, окружающее давление в жидкости p_∞ существенно больше давления газа и пара внутри пузыря, поверхностным натяжением и вязкостью пренебрегаем, пузырь рассматривается как пустая каверна. В этом случае уравнение движения стенки пузыря записывается следующим образом:

$$R\ddot{R} + \frac{3}{2}\dot{R}^2 = -\frac{p_\infty}{\rho},$$

где R - радиус сферической каверны. Полагая, что давление p_∞ действует мгновенно в начале схлопывания (когда $R = R_0$ - начальный радиус каверны) и остается постоянным в процессе схлопывания, получим решение для скорости стенки пузыря при схлопывании

$$\dot{R}^2 = \frac{2p_\infty}{3\rho} \left(\frac{R_0^3}{R^3} - 1 \right).$$

Согласно этому решению, скорость схлопывания стремится к бесконечности когда $R \rightarrow 0$, но при этом уравнение движения уже непригодно, однако это решение дает очень важную качественную информацию о характере поведения пузыря почти в течение всего процесса схлопывания.

Если давление p_∞ постоянно, а в момент времени $t = 0$ справедливы равенства $R = R_0$ и $\dot{R}_0 = 0$, то время схлопывания пузыря до некоторого радиуса $R = R(t)$ будет равно [48]

$$t = R_0 \left(\frac{3\rho}{2p_\infty} \right)^{1/2} \int_{\zeta}^1 \frac{\zeta^{3/2} d\zeta}{(1 - \zeta^3)^{1/2}},$$

где $\zeta = R/R_0$. Рэлей нашел, что время полного схлопывания ($\zeta = 0$) таково: $t^* = 0,91468 R_0 (\rho/p_\infty)^{1/2}$.

В таблице 11 приводится сравнение аналитического значения времени коллапса пузыря со временем, полученным численно из решения задачи МГЭ с постоянным шагом по времени и шагом, выбираемыми автоматически в процессе расчета.

Таблица 11
Выбор шага по времени и время коллапса пузыря

Шаг по времени	0,02	0,01	0,005	0,001	выбираемый автоматически	точное значение
Время коллапса	0,94	0,92	0,915	0,915	0,91486	0,91468

На рис. 35,1-35,5 приведены зависимости радиуса пузыря и нормальной (радиальной) скорости от времени при различных значениях параметра τ (на рисунках параметр τ обозначен через dt): 1)- $\tau = 0,02(R_m/g)^{1/2}$; 2)- $\tau = 0,01(R_m/g)^{1/2}$; 3)- $\tau = 0,005(R_m/g)^{1/2}$; 4)- $\tau = 0,001(R_m/g)^{1/2}$; 5) значение τ выбиралось автоматически. Время коллапса T_c фиксировалось в момент нарушения сферической формы границы или в момент ее полного разрушения. Цифры на графиках обозначают: 1 - аналитическое решение; 2 - численный расчет.

Рис. 35. Зависимость радиуса пузыря и нормальной производной от времени при различном выборе шага по времени

Форма свободной границы пузыря (рис. 35, б) показана в различные моменты времени при $\tau = 0,005(R_m/g)^{1/2}$. Форма поверхности пузыря наносилась на график через одинаковые промежутки времени, начиная с начального момента. Нелинейный характер процесса схлопывания хорошо заметен в последние моменты времени, когда радиальная скорость движения свободной границы пузыря резко возрастает.

Отметим, что предложенный алгоритм автоматического выбора шага по времени позволяет достигать значений радиуса, близких к нулю, с очень хорошим приближением к точному решению. Кроме того, такой выбор временного шага позволяет проходить "спокойные" участки при решении задачи с достаточно большими значениями параметра τ , что, в свою очередь, экономит процессорное время при численных расчетах на ЭВМ.

Проверка эффекта саморегуляризации

При решении задачи об эволюции газового пузыря в безграничной жидкости приходится иметь дело с интегральным уравнением Фредгольма I рода, которое, как известно, является некорректным по Адамару и для его решения необходимо применять специальные регуляризирующие алгоритмы [80]. Однако после применения процедуры МГЭ к решению интегрального уравнения Фредгольма I рода, наблюдается эффект саморегуляризации, теоретически обоснованный (если ядро имеет логарифмическую особенность) в работах [3, 27, 28]. Для проверки этого утверждения результатирующая система алгебраических уравнений решалась двумя методами: Гаусса с выбором ведущего элемента [84] и с использованием итерационного регуляризующего алгоритма, предложенного в работе [82].

Для проверки эффекта саморегуляризации при решении уравнения Фредгольма I рода методом граничных элементов, решалась краевая задача ($\Delta\phi = 0$; $\phi|_{\Gamma} = \phi_0$; $\Gamma \in (x^2 + y^2 + z^2 = 1)$; $\phi|_{\infty} = 0$), имеющая аналитическое решение. На рис. 36 представлена зависимость абсолютной погрешности численного решения краевой задачи от количества узловых точек на границе области. Хорошо заметно, что абсолютная погрешность решения с увеличением числа узлов убывает. Данный пример доказывает наличие эффекта саморегуляризации при решении краевой задачи методом граничных элементов и подтверждает возможность решения результирующей системы линейных алгебраических уравнений методом Гаусса с выбором ведущего элемента.

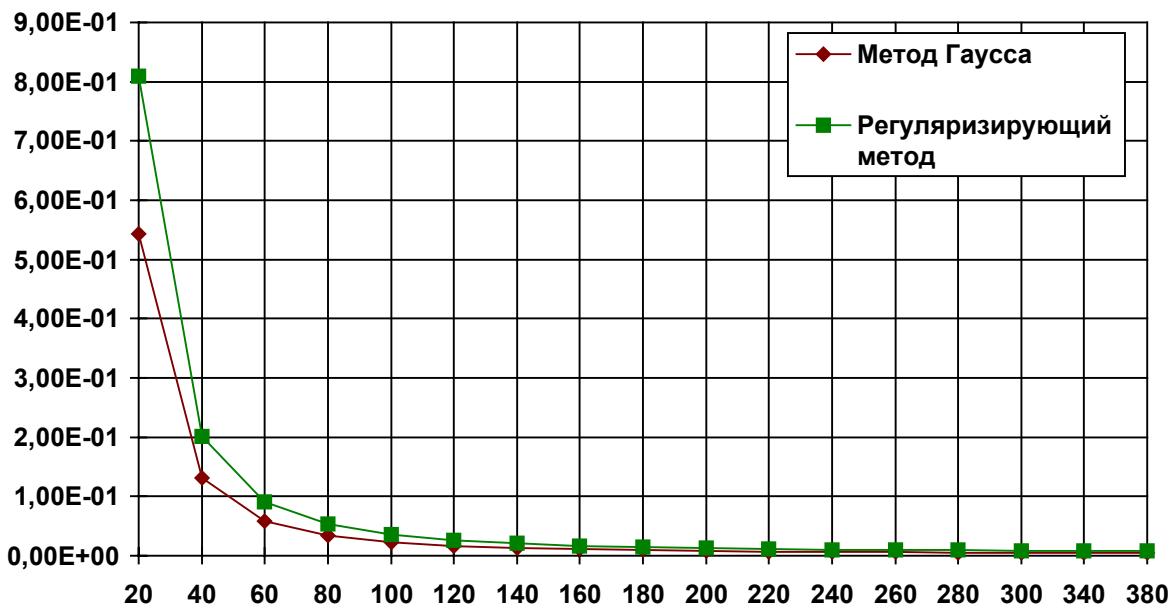


Рис. 36. Зависимость абсолютной погрешности решения от количества узлов на границе

Тест №7. Схлопывание газового пузыря возле твердой стенки

Задача о схлопывании сферической полости возле твердой стенки также является хорошим тестовым примером для метода граничных элементов. Впервые данная задача была решена в работе [102], авторы которой выполнили исследование полной нелинейной задачи о схлопывании сферической каверны, находящейся в начальный момент в покое вблизи твердой стенки. Жидкость принималась невесомой, несжимаемой и невязкой; окружающее давление и давление насыщенного пара считались постоянными на

протяжении всего процесса схлопывания; поверхностное натяжение не учитывалось. Рассчитывалось схлопывание пузыря для двух случаев: $h/R_0 = 1$ и $h/R_0 = 1,5$, где h - расстояние от стенки до центра пузыря, а R_0 - начальный радиус.

На рис. 37,а,б приведены расчеты для обоих случаев, соответственно. Результаты, показанные на рис. 37, разделены сплошной вертикальной линией на две половины: 1 - результаты расчетов; 2 - результаты из работы [102]. Видно, что оба результата имеют приемлемое количественное и качественное совпадение.

Рис. 37. Динамика одиночного пузыря возле твердой стенки: а) на расстоянии $h / R_0 = 1,0$ от стенки; б) на расстоянии $h / R_0 = 1,5$ от стенки

Рис. 38. Динамика двух одинаковых пузырей, расположенных симметрично на расстоянии $2R_0$ друг от друга: а)-форма свободной поверхности, б)-нормальная производная в точках 1 и 2 на оси симметрии

Данный тест дополняется задачей о расширении и схлопывании двух симметрично расположенных относительно горизонтальной оси пузырей, центры которых в начальный момент располагались на расстоянии $2R_0$ друг от друга (R_0 - это максимальный радиус пузыря, до которого он может расширяться в безграничной жидкости). В данной задаче область течения является бесконечной, и фундаментальное решение остается в обычном, для осесимметричного случая, виде. На рис. 38 показаны результаты решения задачи. Цифрами обозначены точки, лежащие на противоположных сторонах обоих пузырей и принадлежащие оси симметрии. На рис. 38,а представлены контуры границы пузырей в некоторые моменты времени, зафиксированные через одинаковые промежутки. На рис. 38,б приведена зависимость нормальной составляющей скорости от времени в отмеченных точках. Временная шкала на рис. 38,б условно разделена на три части: I - соответствует процессу расширения пузырей с начального радиуса $R = 0,1(R_0/g)^{1/2}$; II и III - соответствуют процессу схлопывания. Видно, что на этапе расширения нормальные скорости в точках практически совпадают и пузыри расширяются до момента времени $\tau \approx 0,9(R_0/g)^{1/2}$, не изменяя своей сферической формы. На этапе схлопывания нормальная скорость в точках, обозначенных цифрой 2, начинает увеличиваться, а в точках, обозначенных цифрой 1, остается почти постоянной и близкой к нулю. Скорость в точках 2 продолжает увеличиваться до того момента, пока полностью не сформируются (направленные друг к другу) кумулятивные струйки в обоих пузырях. После этого нормальная скорость не изменяется - зона III. Здесь наблюдается феномен, при котором скорость точки, расположенной на конце кумулятивной струи, остается постоянной вплоть до момента коллапса пузыря. В этот промежуток времени кумулятивная струйка движется только по инерции, обладая кинетической энергией, накопленной ранее. Массовые силы не оказывают никакого влияния на процесс развития струйки. Это указывает на тот факт, что скорость движения кумулятивной струи в момент коллапса пузыря конечна.

6.2. Примеры решенных задач

Задача №1. Определение формы свободной границы при обтекании полукругового цилиндра

Рассмотрим задачу о течении идеальной невязкой несжимаемой жидкости со свободной границей C_1 вдоль дна C_3 , состоящего из прямолинейных участков и цилиндрического выступа. Область течения D ограничена, кроме того, участками втекания C_3 и вытекания C_4 (рис. 39). Постановка задачи и алгоритм построения свободной границы приведены в главе 2.

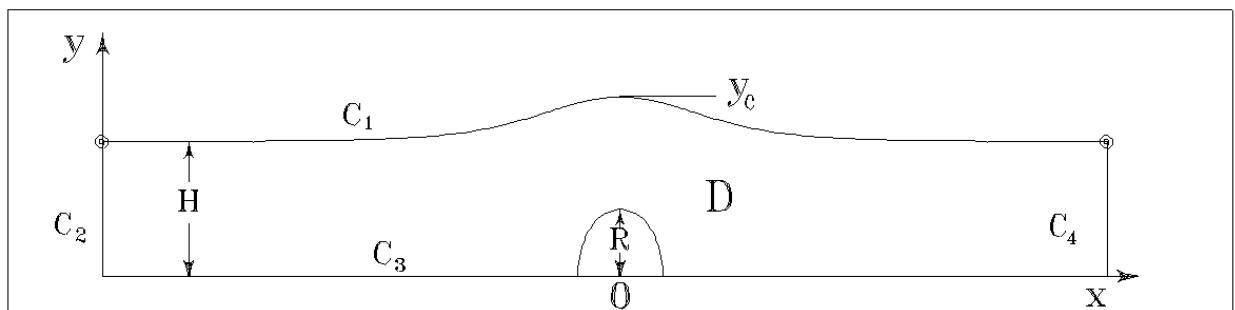


Рис. 39. Картина течения при обтекании цилиндра

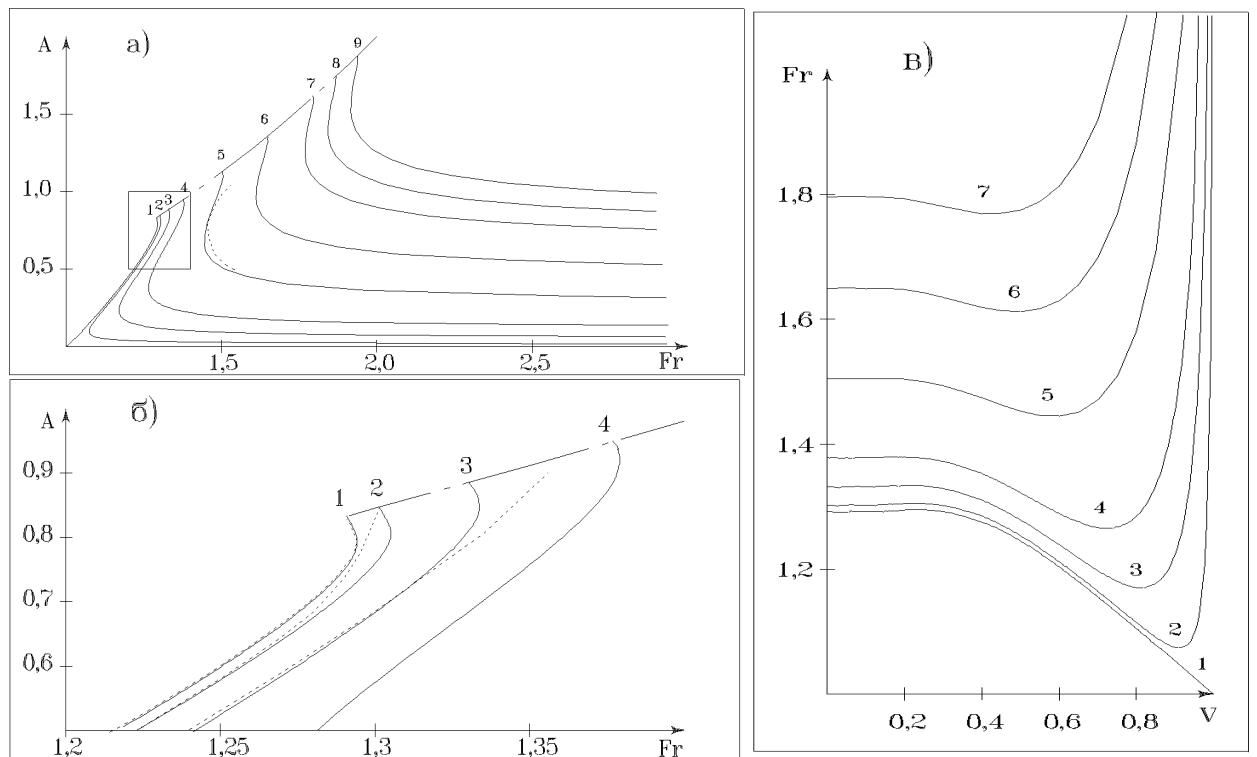


Рис. 40. К вопросу о неединственности решений при обтекании полукругового цилиндра

Если для построения решений задачи обтекания препятствий потоком жидкости использовать интеграл Бернулли в виде (2.19), то удается построить лишь тривиальное решение, которое описывает равномерный поток при исчезновении препятствия. В этом случае решение справедливо для некоторых значений числа Фруда ($Fr \geq 1$), зависящих от отношения R/H (радиуса цилиндра R и глубины потока H), ниже которых стационарного решения не существует. Данная задача допускает и второе решение, построение которого приведено в [35]. Поиску двух решений посвящена и работа [105], где проведены расчеты зависимости амплитуды волны от числа Фруда при $R/H=0,2$ и $0,5$. В этой работе задача не досчитана до конца в области амплитуд, близких к предельному значению. В работе [35] приведен подробный расчет только для $R/H=0,1$. Автором не был отмечен факт неоднозначной зависимости амплитуды $A=A(Fr)$ при значениях амплитуды, близких к предельному. Расчеты, результаты которых приведены на рис. 40, показали, что нелинейная задача обтекания препятствия потоком идеальной несжимаемой тяжелой жидкости имеет еще одно, *третье*, решение в области предельных значений амплитуды волны. Факт наличия неоднозначности решения для уединенной волны отмечен в [105].

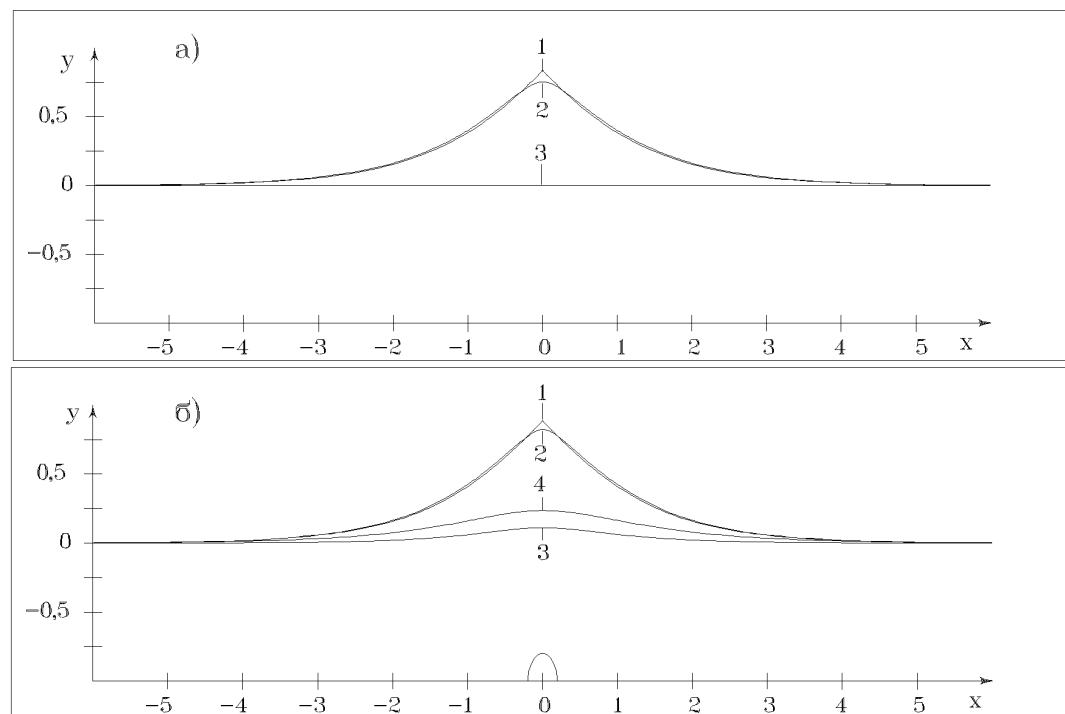


Рис. 41. Формы свободной поверхности при обтекании цилиндра

На рис. 40,а представлены результаты расчетов зависимости амплитуды A от числа Фруда для различных значений R/H . Кривая 1 отвечает $R/H=0$ и описывает уединенную волну, кривые 2 - 9 соответствуют отношению $R/H=0,1; 0,2; 0,3; 0,5; 0,7; 0,9; 1,0; 1,1$. Штрих пунктирная линия соединяет точки кривых, отвечающие максимальному значению $A_{\max} = Fr^2 / 2$. Штриховая линия в окрестности кривой 5 соответствует расчетам работы [105]. Прямоугольником выделена область, где имеется неоднозначное поведение решения в зоне предельной волны. Эта область в увеличенном виде (масштабе) представлена на рис. 40,б, где расчеты для $R/H=0$ (уединенной волны) полностью соответствуют расчетам, проведенным Д.В. Маклаковым на основе разработанной им теории [54] (расчеты для $R/H=0,1$ взяты из [35], для $R/H=0,2$ - из [105]). Зависимость числа Фруда от параметра V при тех же значениях R/H приведена на рис. 40,в, который иллюстрирует, хоть и менее четко выражено, неоднозначную зависимость $Fr=Fr(V)$ в зоне предельных волн.

На рис. 41,а показаны формы свободной поверхности для $R/H=0$ (уединенная волна), соответствующие трем решениям при одном и том же числе Фруда $Fr=1,2910$ для $A=0,8332; 0,7531; 0$ (кривые 1-3); на рис. 41,б - для $R/H=0,2$, $Fr=1,3322$ при $A=0,8875; 0,8220; 0,1106$ (кривые 1-3). Для последнего случая приведена также форма волны для числа Фруда, являющегося максимальным для первого решения и началом для второго решения ($Fr=1,1704$, $A=0,2355$ - кривая 4).

Задача №2. Интегральные характеристики уединенных волн

Прежде чем перейти к вопросам определения интегральных характеристик нелинейных уединенных волн, рассмотрим некоторые вспомогательные вопросы.

Пусть уединенная волна амплитуды A распространяется со скоростью C_0 по поверхности жидкости конечной глубины H , как показано на рис. 42.

Математическая постановка задачи о распространении уединенных волн относится к классу потенциальных течений идеальной жидкости со свободной границей и сводится к определению потенциала скоростей $\phi(x, y, t)$ и функции $y = \eta(x, t)$, описывающей сво-

бодную границу. При этом в области, занятой жидкостью, потенциал должен удовлетворять уравнению Лапласа, а на границе Γ_1 должны выполняться нелинейные кинематическое и динамическое условия, на твердой границе Γ_2 - условие непротекания.

Для решения нестационарной задачи необходимо в начальный момент времени задавать начальную форму свободной поверхности $y(x,0) = \eta(x)$ и начальное распределение потенциала $\phi(0) = \phi(x, \eta, 0)$ на ней. При этом предполагается, что в бесконечности жидкость поконится, а $\eta(-\infty, t) = \eta(+\infty, t) = 0$. Расчетная область ограничивается вертикальными стенками, на которых выставляется условие непротекания.

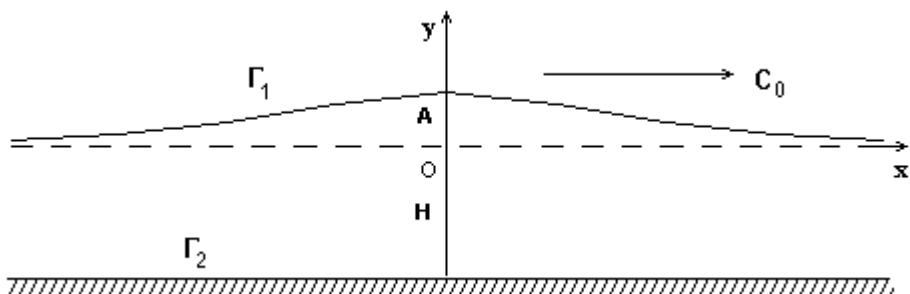


Рис. 42. Обозначения и система координат в задаче об уединенной волне

Длина волны является величиной условной и в линейной теории уединенных волн определяется по формуле Рэлея

$$\lambda = 7,27 \cdot \sqrt{(A+1)/3A}.$$

При численных расчетах длина волны $\lambda(t)$ часто определяется длиной отрезка по x , на котором $\eta(x,t) \geq 0,01 \cdot A(t)$.

Данное определение длины волны использовалось в работе [67]. Число Фруда в данной постановке определяется по формуле

$$Fr = \frac{C_0}{\sqrt{gH}},$$

где C_0 - скорость распространения волны.

В рамках используемой нелинейной модели отсутствует точное явное выражение для уравнения свободной поверхности и распределения потенциала (или скоростей движения частиц) на ней. По-

скольку для начала расчетов нестационарной задачи, наряду с формой волны, достаточно знать распределение потенциала, то исходя из того, что мы имеем дело с волнами, форма которых получена из решения стационарной задачи, нетрудно восстановить потенциал из уравнения Бернулли.

Поскольку свободная граница является линией тока, а вектор скорости на ней направлен по касательной к границе, имеем:

$$q = \frac{\partial \phi}{\partial s} = \sqrt{1 - 2y/Fr^2}, \text{ откуда следует, что } \phi_{i+1} = \phi_i + \frac{q_i + q_{i+1}}{2} \Delta s_i,$$

где $i = \overline{1, Ng}$ - номера узлов точек свободной границы, $\Delta s_i = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$ - длина i -го элемента свободной границы.

Далее необходимо получить потенциал волны, распространяющейся по "спокойной" воде. Для этого необходимо ввести функцию $\Phi = -Fr(\phi - x)$. При этом нормальная производная записывается в виде:

$$\frac{\partial \Phi}{\partial n} = -Fr \frac{dy}{ds}.$$

При изучении динамики волновых движений очень важным, наряду с изучением формы волны и ее амплитуды, является знание интегральных характеристик, таких как энергия, масса, момент, циркуляция, а также соотношения, диктуемые априорным ограничением на высоту волны, исходя из интеграла Бернулли.

В работе [98] доказано, что в теории нелинейных уединенных волн справедливы безразмерные соотношения:

$$I = Fr \cdot M,$$

$$K = Fr \cdot (I - C)/2, \quad (\text{I})$$

$$P = (Fr^2 - 1) \cdot M / 3, \quad (\text{II})$$

где Fr - число Фруда,

$$M = \rho \int_{-\infty}^{\infty} \eta dx \text{ - эффективная масса волны;}$$

$$P = \frac{\rho g}{2} \int_{-\infty}^{\infty} \eta^2 dx \text{ - потенциальная энергия;}$$

$$I = \rho \int_{-\infty}^{\infty} \int_{-1}^{\eta} u dy dx \text{ - импульс;}$$

$$K = \frac{\rho}{2} \int_{-\infty}^{\infty} \int_{-1}^{\eta} (u^2 + v^2) dy dx \text{ - кинетическая энергия;}$$

$$C = \int_{-\infty}^{\infty} u \cdot ds = [\Phi]_{-\infty}^{\infty} \text{ - циркуляция волны.}$$

Соотношения, приведенные выше, могут быть преобразованы к другому виду. Для этого воспользуемся правилом дифференцирования интеграла с переменным верхним пределом, получим

$$\frac{\partial}{\partial x} \int_{-1}^{\eta} \Phi(x, y) dy = \int_{-1}^{\eta} \frac{\partial \Phi}{\partial x} dy + \Phi(\eta) \cdot \frac{\partial \eta}{\partial x}.$$

Интегрирование данного выражения по x , приводит к следующему:

$$C = I + \int_{-\infty}^{\infty} \Phi(\eta) d\eta = I + \Phi \eta|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} \eta d\Phi,$$

или, учитывая уравнение (I), получим $K = \frac{1}{2} Fr \int_{-\infty}^{\infty} \eta d\Phi$.

Запишем соотношение, связывающее потенциал ϕ и функцию η , описывающую свободную поверхность. В виду вышеизложенного, можем записать:

$$d\Phi = -Fr \cdot d\phi + Fr \cdot dx = -Fr \cdot qds + Fr \cdot dx; \quad ds = \sqrt{1 + \eta'^2} dx.$$

В силу этого и из интеграла Бернулли имеем:

$$d\Phi = Fr(1 - \sqrt{1 - 2\eta/Fr^2} \sqrt{1 + \eta'^2}) dx.$$

Из этих соотношений следует:

$$\begin{aligned} K &= \frac{1}{2} Fr^2 \int_{-\infty}^{\infty} [\eta - 1 + (1 - 2\eta/Fr^2)^{1/2} (1 + \eta'^2)^{1/2}] dx \equiv \\ &\equiv \frac{1}{2} Fr^2 \int_{-\infty}^{\infty} \eta [1 - (1 - 2\eta/Fr^2)^{1/2} (1 + \eta'^2)^{1/2}] dx, \\ C &= Fr \cdot \int_{-\infty}^{\infty} [1 - (1 - 2\eta/Fr^2)^{1/2} \cdot (1 + \eta'^2)^{1/2}] dx; \\ I &= Fr \cdot \int_{-\infty}^{\infty} \eta dx. \end{aligned}$$

В нестационарных расчетах естественным является определение интегральных характеристик путем прямого вычисления интегралов по границе области решения задачи.

Одним из основных законов, выполняющихся в модели идеальной жидкости, является закон сохранения величины потока жидкости через границу. Этот закон описывается интегральным соотношением:

$$F = \int_{\Gamma} \frac{\partial \Phi}{\partial n} ds.$$

В зарубежной литературе этот интеграл называется интегралом потока. Согласно теории, суммарный поток жидкости через границу должен равняться нулю, и, следовательно, величина F также должна быть равной нулю.

Для нахождения интеграла потока использовалась формула выда:

$$F = \frac{1}{2} \sum_{i=1}^N (q_i + q_{i+1}) L_i; \quad q_{N+1} \equiv q_1 \quad (q = \frac{\partial \Phi}{\partial n}).$$

Другой, часто используемый критерий консервативности, - это закон сохранения полной энергии $E=P+T$, где потенциальная и кинетическая энергии в области, ограниченной боковыми стенками, определяются численно:

$$P = \frac{1}{2} \int_a^b y^2 dx = \frac{1}{6} \sum_{i=1}^{Ng} (y_i^2 + y_i y_{i+1} + y_{i+1}^2)(x_i - x_{i+1}), \quad (*)$$

$$K = -\frac{1}{2} \oint_{\Gamma} \Phi \frac{\partial \Phi}{\partial n} ds = -\frac{1}{12} \sum_{i=1}^{Ng} (2\Phi_i q_i + \Phi_i q_{i+1} + \Phi_{i+1} q_i + 2\Phi_{i+1} q_{i+1}) L_i. \quad (**)$$

Очень хорошим критерием контроля точности метода, для задач со свободными границами, является закон сохранения массы:

$$\begin{aligned} M &= \frac{1}{2} \left(\oint_{\Gamma} y dx - \oint_{\Gamma} x dy \right) - S_0 = \\ &= \frac{1}{4} \left(\sum_{i=1}^N (y_i + y_{i+1})(x_i - x_{i+1}) + (x_i + x_{i+1})(y_i - y_{i+1}) \right) - S_0, \end{aligned}$$

где S_0 - площадь, занятая жидкостью ниже уровня невозмущенной свободной поверхности.

Циркуляция вычисляется по формуле:

$$C = \int_a^b \frac{\partial \Phi}{\partial s} ds = \Phi_b - \Phi_a.$$

Здесь a и b - абсциссы точек пересечения свободной поверхности с боковыми границами, L_i - длина i -го элемента, Ng - количество точек на свободной границе, N - общее число точек по границе области.

Абсолютная погрешность вычислений потенциальной и кинетической энергий по формулам (I-II) и (*-**) равна 10^{-3} для малых и предельных волн и 10^{-5} для волн в середине исследуемого диапазона. Данные оценки получены при общем числе элементов на свободной границе, равном 290.

На рис. 43 приведены рассчитанные характеристики уединенных волн: масса, число Фруда, полная кинетическая и потенциальная энергии. Эти характеристики представлены в виде графиков (кривые 1-5 соответственно).

Важно отметить, что все максимальные значения характеристик уединенных волн достигаются до наступления максимальной амплитуды, а максимум массы не совпадает с максимумами полной энергии и числа Фруда. Этот факт также отмечается в работе [86].

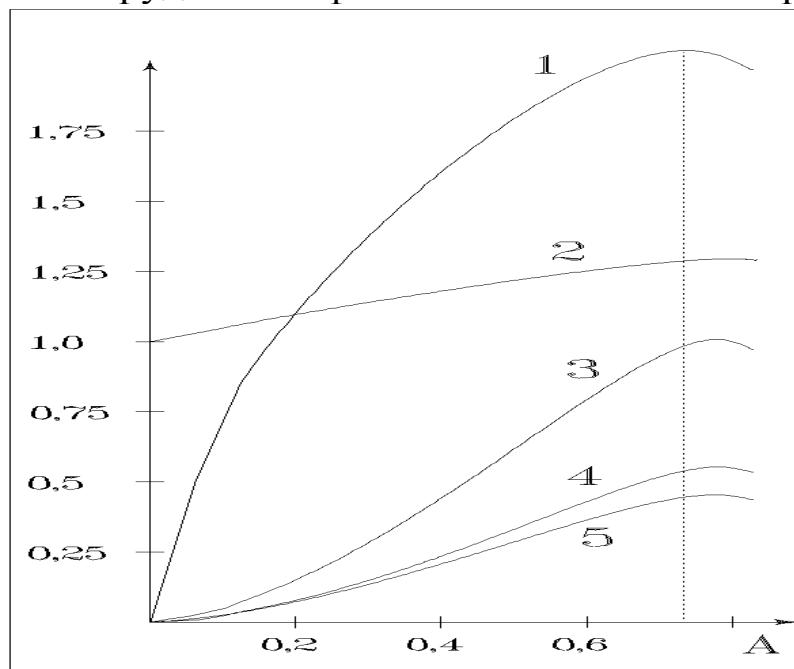


Рис. 43. Интегральные характеристики уединенных волн

Задача №3. Накат уединенной волны на твердую стенку

Ниже приводятся результаты расчетов процесса наката уединенной волны на твердую вертикальную стенку. На рис. 44 и 45 представлены графики максимума гребня волны при движении по

каналу и графики изменения кинетической, потенциальной и полной энергий. График амплитуды рассчитанной уединенной волны (сплошная линия -1) лежит в середине между графиками, построенными по приближениям Буссинеска (штриховая линия -2) и Овсянникова (штрихпунктирная линия -3) [88].

Изменение полной, кинетической и потенциальной энергий

Точкой на оси Т обозначен момент максимума потенциальной энергии. Нетрудно заметить, что моменты наступления максимума потенциальной энергии и максимального заплеска на стенку смешены по времени, друг относительно друга. Особенно это заметно на рис. 45, где показан накат относительно высокой волны: $A=0,71$. Эта временная несогласованность может служить объяснением фазового сдвига набегающей и отраженной волн.

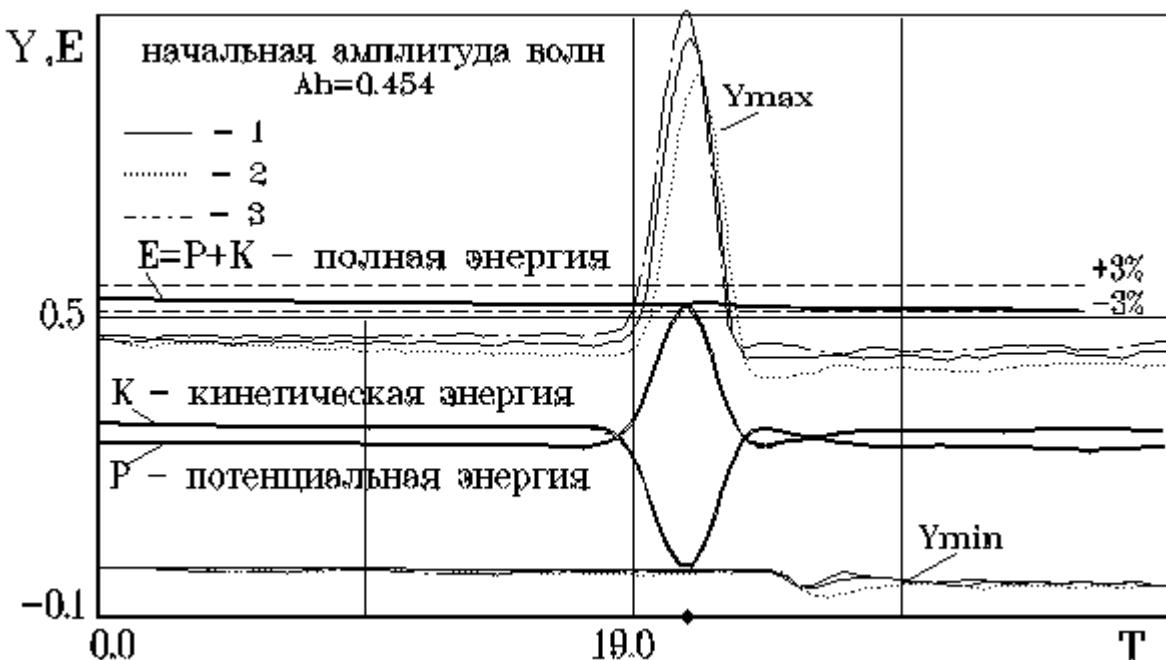


Рис. 44. Сравнение расчетов по амплитуде при накате волны высоты $A=0,454$ на стенку, для разных приближений: 1 - расчет; 2 - Овсянникова; 3 – Буссинеска

После того, как энергия начинает возвращаться к своему первоначальному состоянию, волна еще некоторое время остается “прилипшей” к стенке, а затем резко падает, принимая свое минимальное значение, опускаясь ниже первоначального уровня. В этот момент потенциальная энергия также минимальна, а кинетическая - имеет величину большую, чем в начале движения. Это может служить объяснением, почему скорость волны, имеющей меньшую амплитуду, больше чем у начального солитона [67].

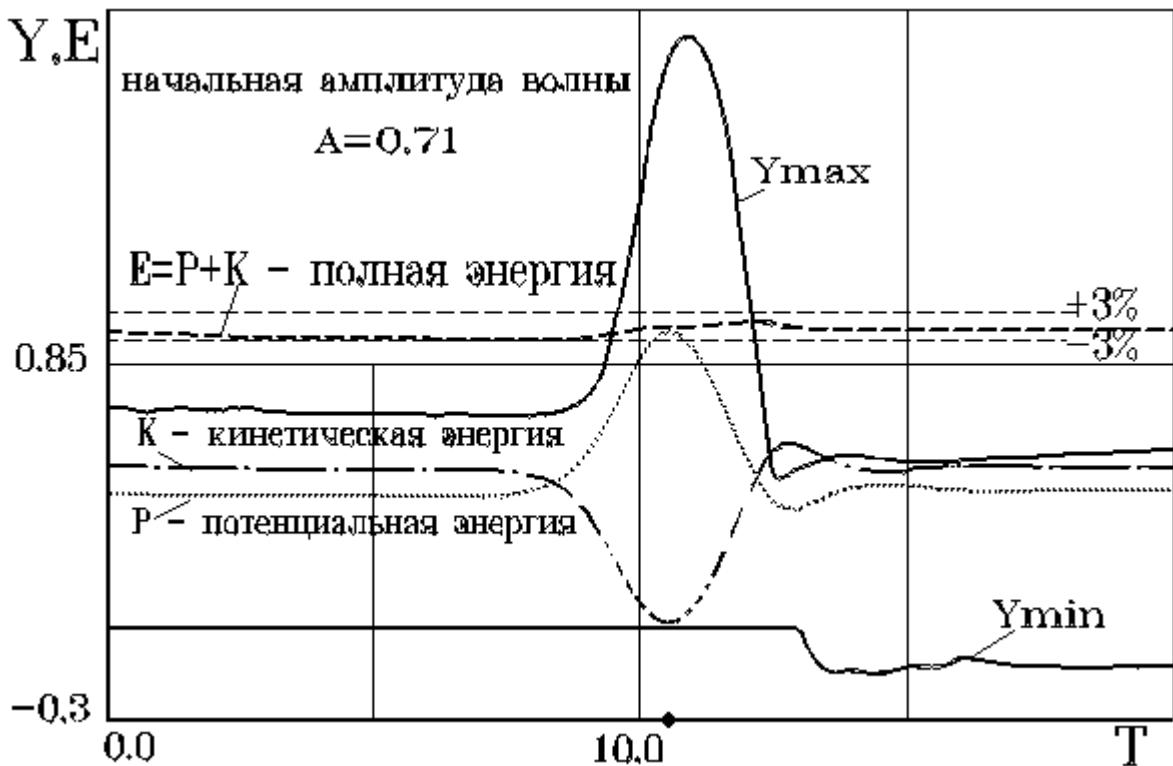


Рис. 45. Изменение амплитуды при накате волны высоты $A=0,71$ на стенку и изменение полной, кинетической и потенциальной энергий

Кроме кинематических особенностей возникающих течений, интерес представляют динамические воздействия волн на вертикальную стенку.

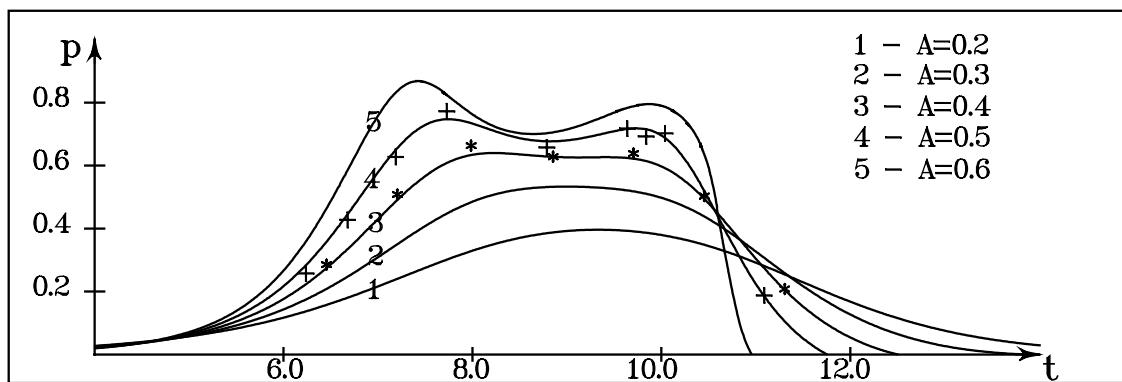


Рис. 46. Хронограммы волнового давления на стенку в точке уреза жидкости

На рис. 46 для волн различных амплитуд показаны хронограммы волнового давления в точке стенки, совпадающей с начальным урезом жидкости. Символами "*" и "+" на график нанесены результаты расчетов для волн, амплитуд $A=0,4$ и $A=0,5$ соответственно, взятые из работы [88]. При амплитудах $A>0,3$ расчетные хронограммы имеют два локальных максимума. Кроме того, первый максимум имеет большее значение, чем второй, при этом момент мак-

симального заплеска волны на стенку не совпадает с моментами локальных максимумов давления. Эти особенности при накате солитонов на вертикальную стенку подтверждаются экспериментами [55].

Задача №4. Взаимодействие солитонов

Солитоны - это удивительный вид волновых явлений. Из линейной теории известно, что солитоны сохраняют в процессе движения свою скорость и амплитуду, а при взаимном столкновении они взаимодействуют между собой, сохраняя свою первоначальную форму [83]. Это утверждение справедливо для солитонов малой амплитуды. Солитоны, чья суммарная амплитуда превышает величину $A \approx 0,4$, после взаимодействия претерпевают изменение формы, связанное с нелинейными эффектами.

Ниже, на рис. 47, приводятся результаты расчетов по движению и взаимодействию солитонов. Солитоны разной амплитуды двигаются навстречу друг другу (амплитуда левого солитона $A_l = 0,376$, амплитуда правого - $A_n = 0,197$).

Рис. 47 разбит на три части. Нижняя - рис. 47,а, характеризует процесс взаимодействия солитонов: слияние в один солитон, обмен энергией и продолжение движения. При этом, как и в линейном случае, при взаимодействии друг с другом, вершины солитонов получают смещение относительно линии их вершин [83]. Такое поведение солитонов дает основание утверждать, что и в нелинейном случае солитоны при взаимодействии не проходят друг через друга, а отталкиваются, меняясь при этом энергией и амплитудой.

На рис. 47,б показана ситуация, когда два солитона разной высоты двигаются в разные стороны, накатываясь на твердые стенки и отражаясь от них.

Третья часть (рис. 47,в) повторяет первую, но с той лишь разницей, что в процессе взаимодействия участвуют солитоны, отличающиеся от первоначальных на 2% по амплитуде. Полная энергия системы сохраняется с точностью до 1,5 % от начальной.

В данной задаче область течения разбивалась на 450 элементов, из них на свободной границе бралось 250. Начальное разбиение свободной границы выбиралось равномерным по длине граничных элементов. Шаг по времени подбирался автоматически согласно алгоритму, описанному ранее.

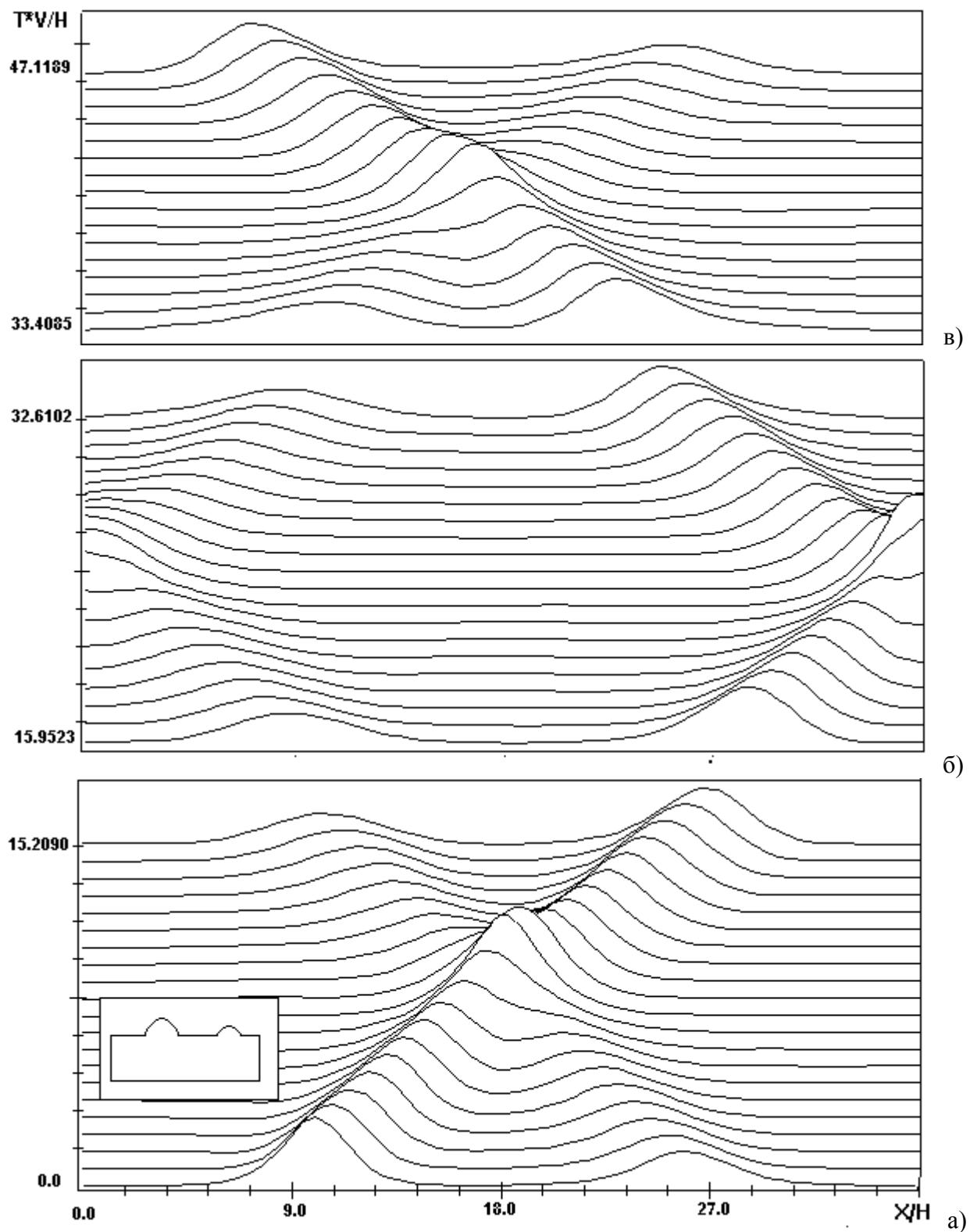


Рис. 47. Взаимодействие двух солитонов: амплитуда левого - $A_l=0,376$, правого - $A_n=0,197$

Для повышения точности расчетов во всех задачах, связанных с накатом волн на стенку, на боковых твердых участках границы области разбиение на элементы проводилось со сгущением узлов к свободной границе и с переразбиением этих участков на каждом

шаге по времени. Сгущение узлов проводилось таким образом, чтобы длины элементов на свободной границе и твердой стенке в районе точки контакта имели одинаковые размеры.

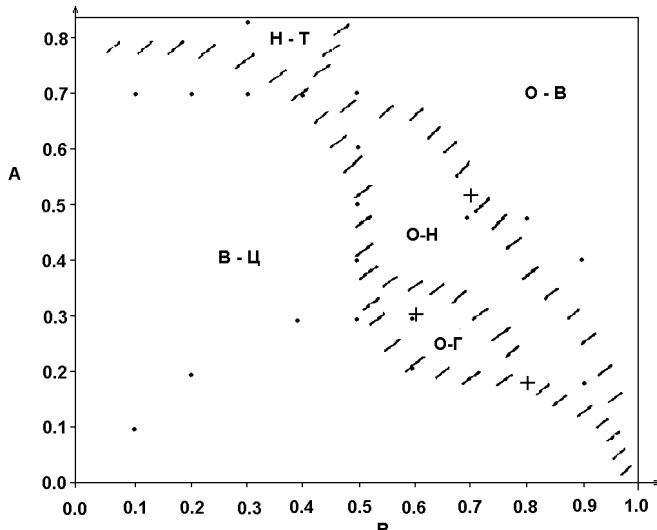


Рис. 48. Диаграмма волновых режимов при взаимодействии уединенной волны с полукруговым цилиндрическим препятствием. По осям отложено: А -амплитуда солитона, R - радиус цилиндра.

Задача №5. Движение солитона над дном с полукруговым выступом

Задача о накате уединенной волны на полукруговой цилиндрический выступ подробно описана в работе [94]. В ней показано, что взаимодействие волны и погруженного цилиндра порождает различные волновые картины. Авторы провели классификацию волновых движений и определили 5 зон, соответствующих различным волновым режимам: В-Ц - волновая цепь; О-В - опрокидывание вперед; О-Н - опрокидывание назад; О-Г - обмен гребнями; Н-Т - неустойчивость Танаки. По данной задаче нами также проводились расчеты, независимо от вышеуказанной работы. После знакомства с приведенной в работе диаграммой было проведено детальное со-поставление результатов. Оказалось, что наши расчеты полностью вкладываются в предлагаемую волновую классификацию и хорошо ложатся на диаграмму (рис. 48), взятую из [94] (точками отмечены выполненные численные расчеты). На рис 49 приведены результаты расчетов, показывающие характерные для соответствующих зон волновые процессы. Геометрия области течения схематически показана на рис. 39.

Глубина бассейна $H=1$, центр цилиндра всегда находился в точке $x = 0, y = -1$, вершина волны располагалась в начальный мо-

мент времени в точке $x = -8, y = A$ (A - амплитуда волны). На границе области бралось 370 элементов, из них 197 - на свободной поверхности.

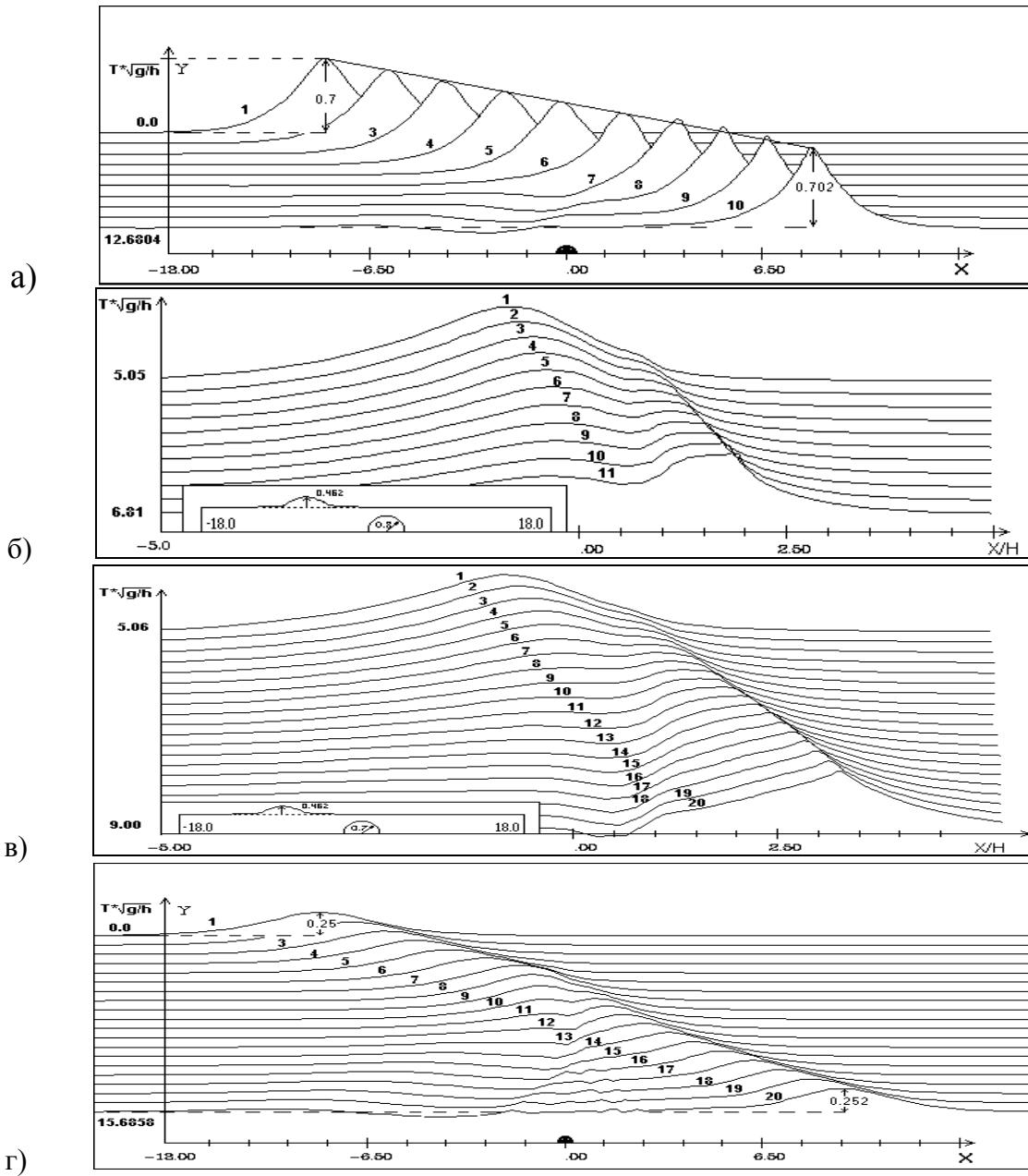


Рис. 49. Волновые картины течений при взаимодействии солитона с полукруговым цилиндрическим выступом: а) В-Ц - волновая цепь; б) О-В - опрокидывание вперед; в) О-Н - опрокидывание назад; г) Н-Т - неустойчивость Танаки

Волновая цепь - это режим, при котором солитон, проходящий над препятствием, практически не изменяет своей формы, а сзади него формируется последовательность затухающих волн малой амплитуды. Расчет по данному волновому режиму представлен на рис. 49,а для волны амплитуды $A=0,7$ и радиуса $R=0,4$. Режим "оп-

"рокидывание вперед" - это режим, при котором волна, проходя над препятствием, трансформируется так, что на ее переднем фронте зарождается "вторая" волна. При этом гребень первой волны уменьшается, а гребень второй волны увеличивается по амплитуде, становясь выше первого, и затем опрокидывается.

На рис. 49,б представлен расчет по накату уединенной волны амплитуды $A=0,462$ на полукруговой цилиндр радиуса $R=0,8$. Для данного случая режим опрокидывания напоминает скользящий бурун. Этот конкретный случай просчитан в работе [94], и на рис. 50,а проводится сравнение формы свободной поверхности в моменты, близкие к обрушению. Другой расчет, также приведенный в работе [94] для амплитуды $A=0,462$ и радиуса $R=0,7$, но уже относящийся к режиму "опрокидывание назад", представлен на рис. 49,в. Данный режим характеризуется тем, что солитон в процессе прохождения над препятствием принимает форму волны, имеющей амплитуду, меньшую первоначальной, но на заднем фронте прошедшей волны образуется впадина, в которой формируется всплеск, опрокидывающийся "против движения" основной волны. Момент, близкий к опрокидыванию, а также сравнение с расчетом уже цитируемых авторов, представлен на рис. 50,б.

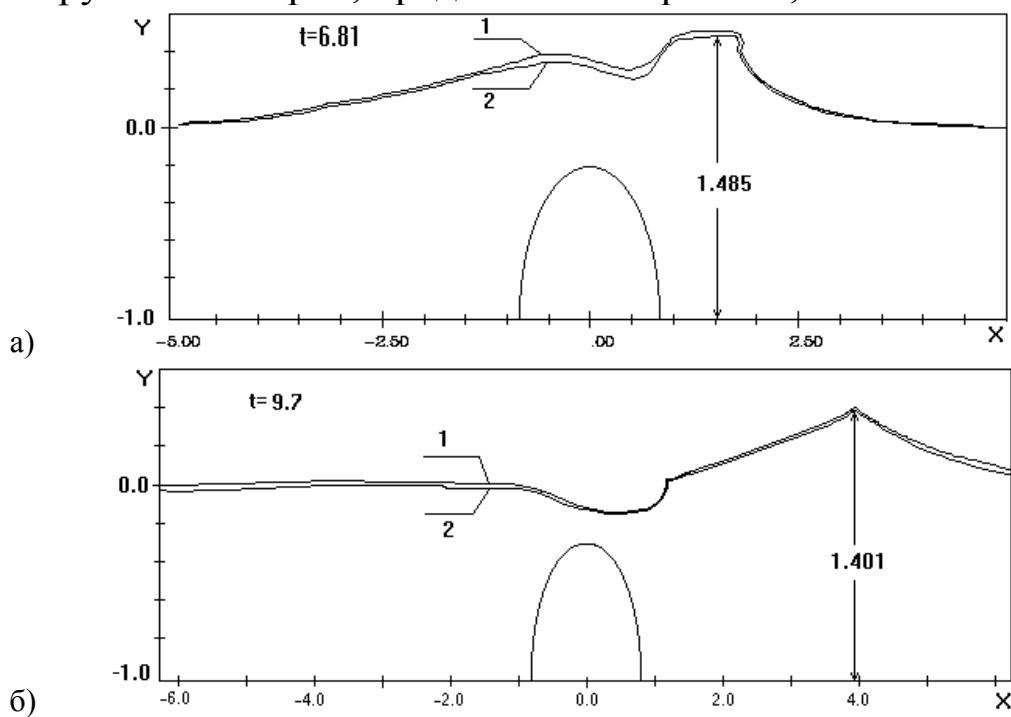


Рис. 50. Сравнение с расчетом из работы [94]- кривая 1, расчет авторов - кривая 2

Режим "обмен гребнями" соответствует режиму, при котором происходит плавный "переток" гребня уединенной волны через препятствие. При подходе волны к препятствию ее амплитуда па-

дает, а за телом волна вновь зарождается в виде другого солитона. В режимах О-В и О-Н также наблюдается обмен гребнями, но здесь более важными являются процессы, связанные с обрушением волн при трансформации солитонов. Характерный для этой зоны режим распространения солитона показан на рис.49,г, где приведена форма свободных границ в различные моменты времени для солитона амплитуды $A=0,25$ при радиусе цилиндра $R=0,65$.

Режим "Неустойчивость Танаки" характеризуется тем, что волны большой амплитуды $A>0,78$ при взаимодействии с препятствиями радиуса, меньшего 0,5, могут вести себя двояко. С одной стороны происходит опрокидывание гребня вперед по движению, а с другой стороны, может наблюдаться трансформация волны в волну меньшей амплитуды, но имеющую ту же полную энергию, что и первоначальная волна.

Относительная ошибка изменения полной энергии для всех расчетов находилась в пределах 2%.

Задача №6. Горизонтальное движение полукругового цилиндра из состояния покоя по ровному дну¹

Задача о движении твердых тел под свободной поверхностью относится к важной и сложной проблеме в гидродинамике: проблеме волнового движения. Эта теория получила серьезное развитие в работах советских и зарубежных ученых в основном в линейной постановке [75]. Нелинейная теория развита значительно хуже, хотя как раз здесь, при режимах движения наиболее интересных для исследований, наблюдаются эффекты, связанные с весомостью, - опрокидывание волн [64], [106], "султаны" [47] и т. д..

В декартовой системе координат x,y рассматривается нестационарное движение идеальной несжимаемой жидкости со свободной границей. Область $D(t)$, занятая жидкостью, ограничена гладкими поверхностями $S(t)$, $Q(t)$ и W . Граница $S(t)$ является свободной поверхностью жидкости, $Q(t)$ - граница движущегося тела, граница W представляет собой твердую стенку. Предполагается, что в области $D(t)$ происходит безвихревое потенциальное движение однородной жидкости, имеющей постоянную плотность ρ . Движение жидкости порождается наличием внутри ее движущегося горизон-

¹ Все векторные поля и изолинии, приведенные в данном параграфе, построены с помощью графического пакета VIZA2. Разработчик И.И. Кузьмин.

тально по прямолинейному дну со скоростью $V(t)$ полукругового цилиндра радиусом R и наличием массовых сил, обладающих потенциалом gz ($g=\text{const}$). Давление P_a на свободной границе постоянно и равно нулю. В начальный момент времени $t=0$ $S(t)=S(0)$. Область течения представляет собой канал глубиной H . Схема течения показана на рис. 51.

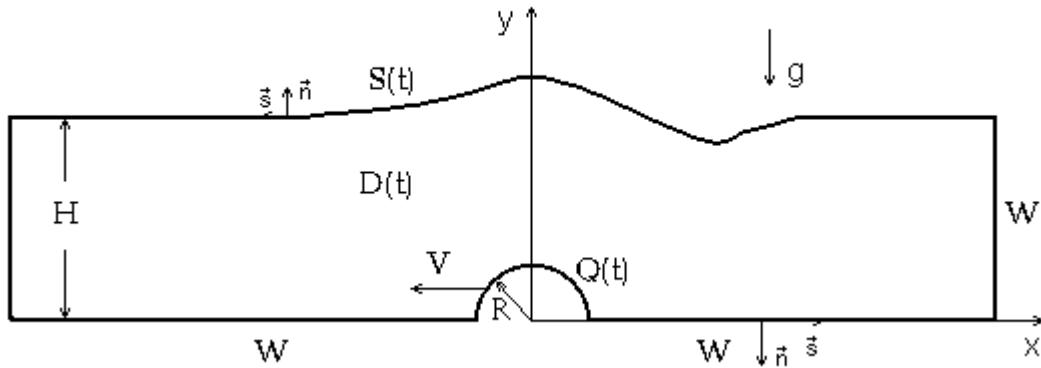


Рис. 51. Схема и обозначения в задаче о неустановившемся движении цилиндра

Математическая задача для потенциала скоростей в безразмерных переменных записывается в следующем виде:

$$\Delta\phi = 0 \quad ; \vec{x}(x, y) \in D(t),$$

$$\frac{d\vec{x}}{dt} = \nabla\phi \quad ; \vec{x}(x, y) \in S(t) \cup Q(t),$$

$$\frac{d\phi}{dt} - \frac{1}{2} |\nabla\phi|^2 + (y - 1)/Fr^2 = 0 \quad ; \vec{x}(x, y) \in S(t),$$

$$(\nabla\phi, \vec{n}) = 0 \quad ; \vec{x}(x, y) \in W;$$

$$(\nabla\phi, \vec{n}) = -V(t)\cos(\vec{n}, x) \quad ; \vec{x}(x, y) \in Q(t),$$

где \vec{n} - вектор внешней нормали к границе области течения.

В данной постановке в качестве характерных линейных величин выбирались глубина канала H и скорость цилиндра V . Число Фруда $Fr^2 = \frac{V^2}{gH}$.

В настоящем параграфе изучаются кинематические характеристики возникающего течения, однако основное внимание уделено исследованию характеристик волны, возникающей за телом, и движению жидкости в том диапазоне чисел Фруда, при которых не существует стационарных течений [76]. Первая группа задач условно называется "существенно нелинейными задачами", вторая - "существенно неустановившимися задачами".

Существенно нелинейные течения

Причиной отсутствия стационарных течений могут быть режимы, при которых за телом на свободной границе образуется всплеск, впоследствии разрушающийся как гребень волны. Последовательные этапы разрушающейся волны показаны на рис. 52-53. Положение центра цилиндра соответствует значению безразмерного времени, отложенному по оси X , со знаком минус. Цифрами 1-6 отмечены кривые в моменты времени, показанные на рисунках.

В этой задаче, наряду с геометрическими характеристиками, интерес представляют и физические особенности возникающей волны. Опрокидывающаяся волна вследствие быстроты происходящего процесса, значительных ускорений и больших искривлений свободной поверхности представляет собой наиболее трудный объект для исследования волновых явлений. В данной проблеме характерным является то, что перед обрушением элемент водной поверхности становится вертикальным, а затем гребень разрушается, образуя струю, направленную в сторону движения волны. Последнее хорошо видно из расчетов, приведенных на рис. 52-53. Эти расчеты проводились в области $[-12H, 6H]$ по оси OX , при глубине $H=1$. Для задачи, приведенной на рис. 52 - $R=0,5H$, $Fr=0,2$, на рис. 53, $R=0,25H$, $Fr=0,5$. Количество узлов на свободной границе было 108, а всего в области - 290 узлов. При этом точки на свободной границе, в начальный момент времени, специальным образом сгущались в окрестности вершины цилиндра.

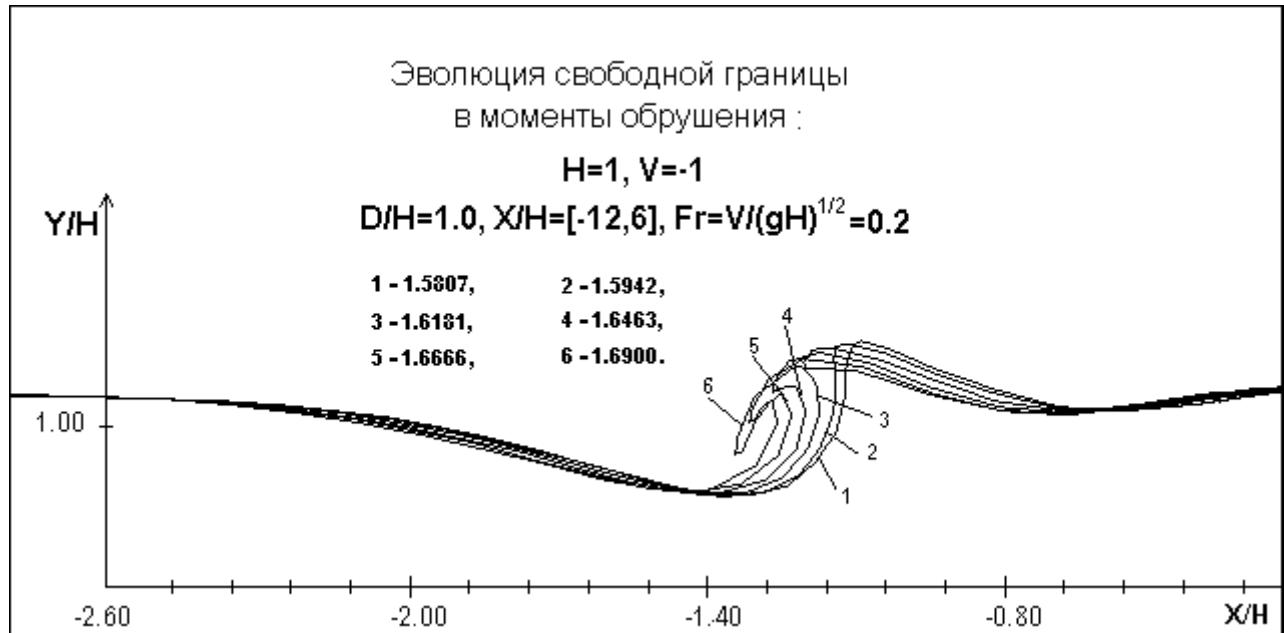


Рис. 52. Движение цилиндра при числе Фруда $Fr=0.2$

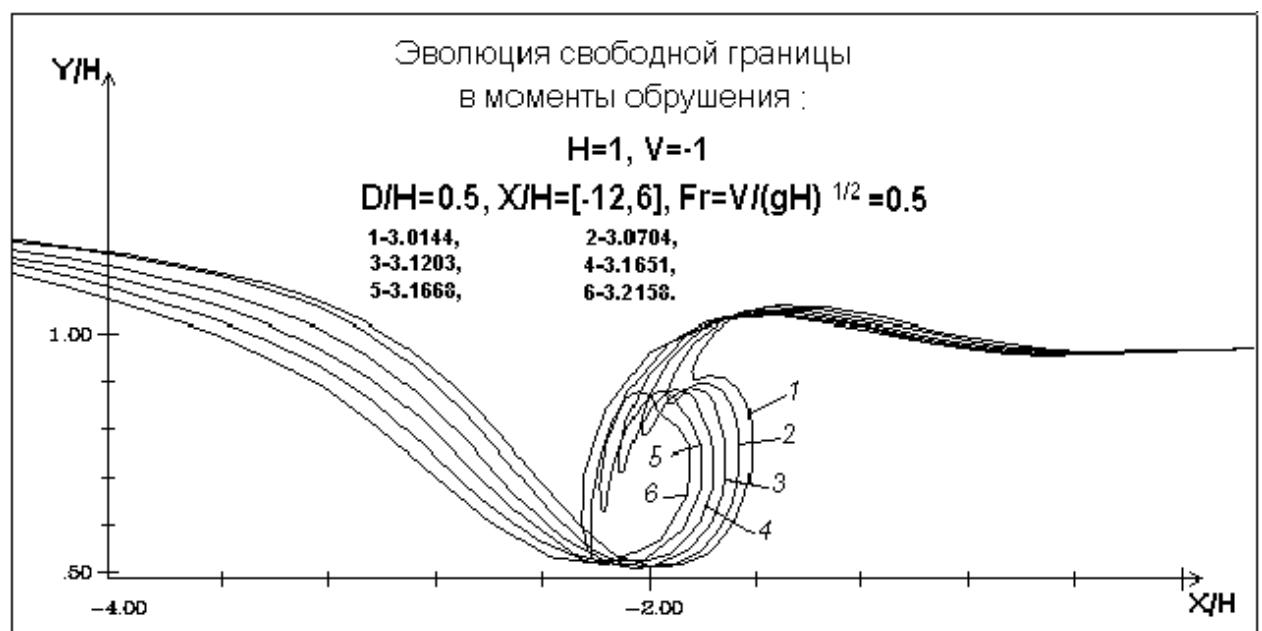


Рис. 53. Движение цилиндра при числе Фруда $Fr=0.5$

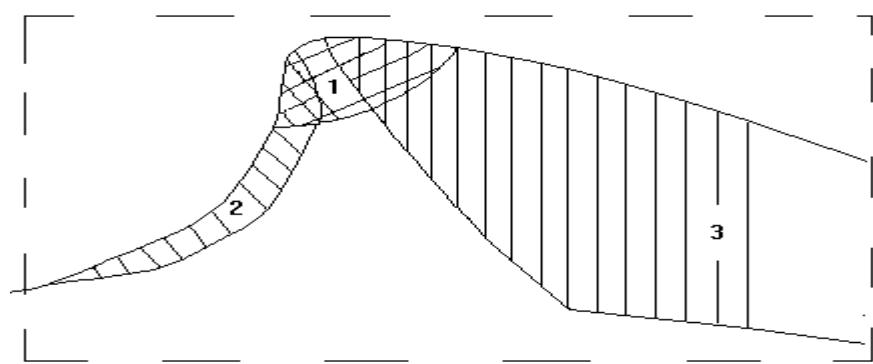


Рис. 54. Структура гребня волны в моменты перед обрушением

В обзоре [64], где рассматриваются уединенные волны, говорится о том, что в опрокидывающейся волне можно выделить три зоны, которые появляются еще до того, как волна станет вертикальной (рис. 54): 1 - скорость отдельных частиц жидкости превышает фазовую скорость волны; 2 - в передней зоне волны имеется тонкий слой, в котором ускорение частиц значительно выше ускорений в остальной волне; 3 - на задней поверхности волны появляется довольно обширная зона, в которой ускорение частиц очень мало. Проведенные в работе [106] исследования задачи об опрокидывании уединенной волны подтверждают наличие существования этих зон.

Рассматриваемые в данном параграфе волны хотя и отличаются от уединенных, но также имеют перечисленные выше особенности, свойственные опрокидывающимся уединенным волнам. Это иллюстрирует рис. 55, где показаны рассчитанные поля скоростей и ускорений. Видно, что зоны, описанные в работе [64] имеют место и в этих задачах (на рисунке они отмечены соответствующими цифрами).

Обращает на себя внимание различие в поведении опрокидывающейся волны (рис. 52, 53). В одном случае гребень формируется за счет жидкости, накатывающейся на цилиндр сзади (рис. 52). По классификации опрокидывающихся волн этот случай может быть отнесен к категории скользящих бурунов. Во втором случае (рис. 53) энергия волны подпитывается за счет "убывания" волны перед цилиндром, и по характеру - это ныряющий бурун.

Структура волн за телом внутри жидкости

Кинематическая картина течения внутри жидкости для чисел Фруда $Fr = 0,1$ и $Fr = 1,0$ показана на рис. 56. Положение цилиндра показано в неподвижной системе координат.

Для того, чтобы получить внутреннюю картину течения, строилась сетка по алгоритму, изложенному в разделе 3.7.

Поля скоростей и ускорений строились по алгоритму, приведенному в работе [14]. Поскольку сетка в расчетах внутренних полей для МГЭ не играет существенной роли, то никакой регуляризации сетки не проводилось. Просто в процессе счета внутренние точки на каждом шаге по времени передвигались по схеме Эйлера.

На рис. 56 показаны векторные поля в слое жидкости, верхняя граница которого есть свободная поверхность, а нижняя граница -

это некоторая условная линия, которая строится на некотором расстоянии от свободной границы и в начальный момент параллельна ей.

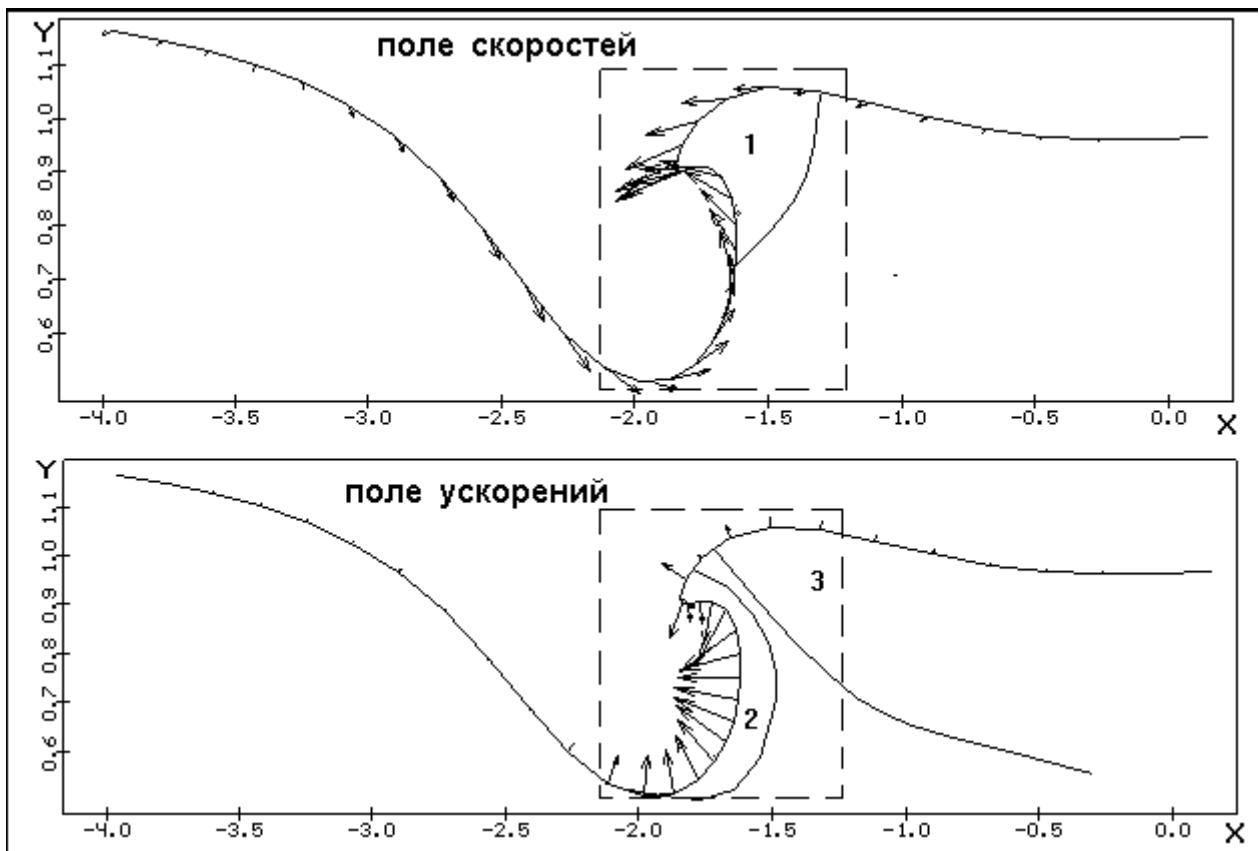
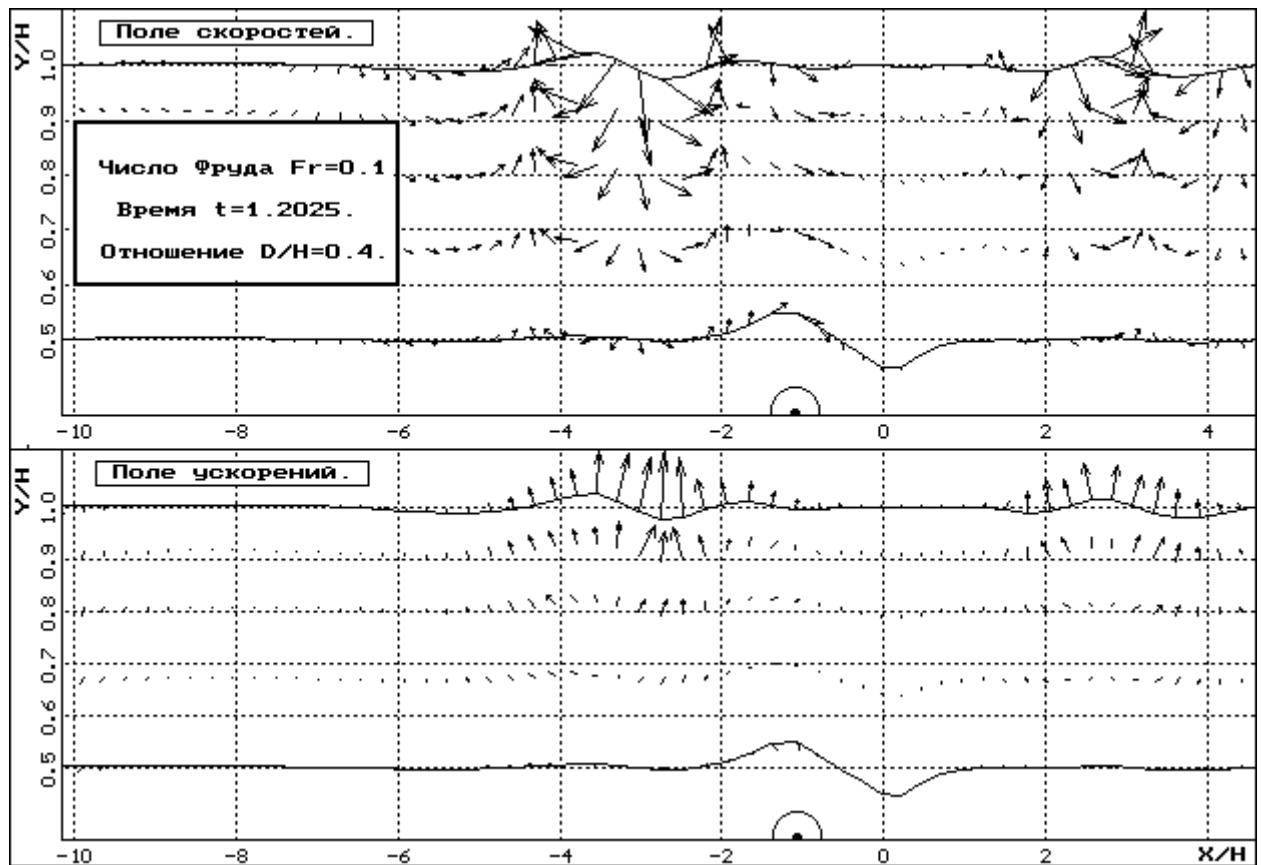


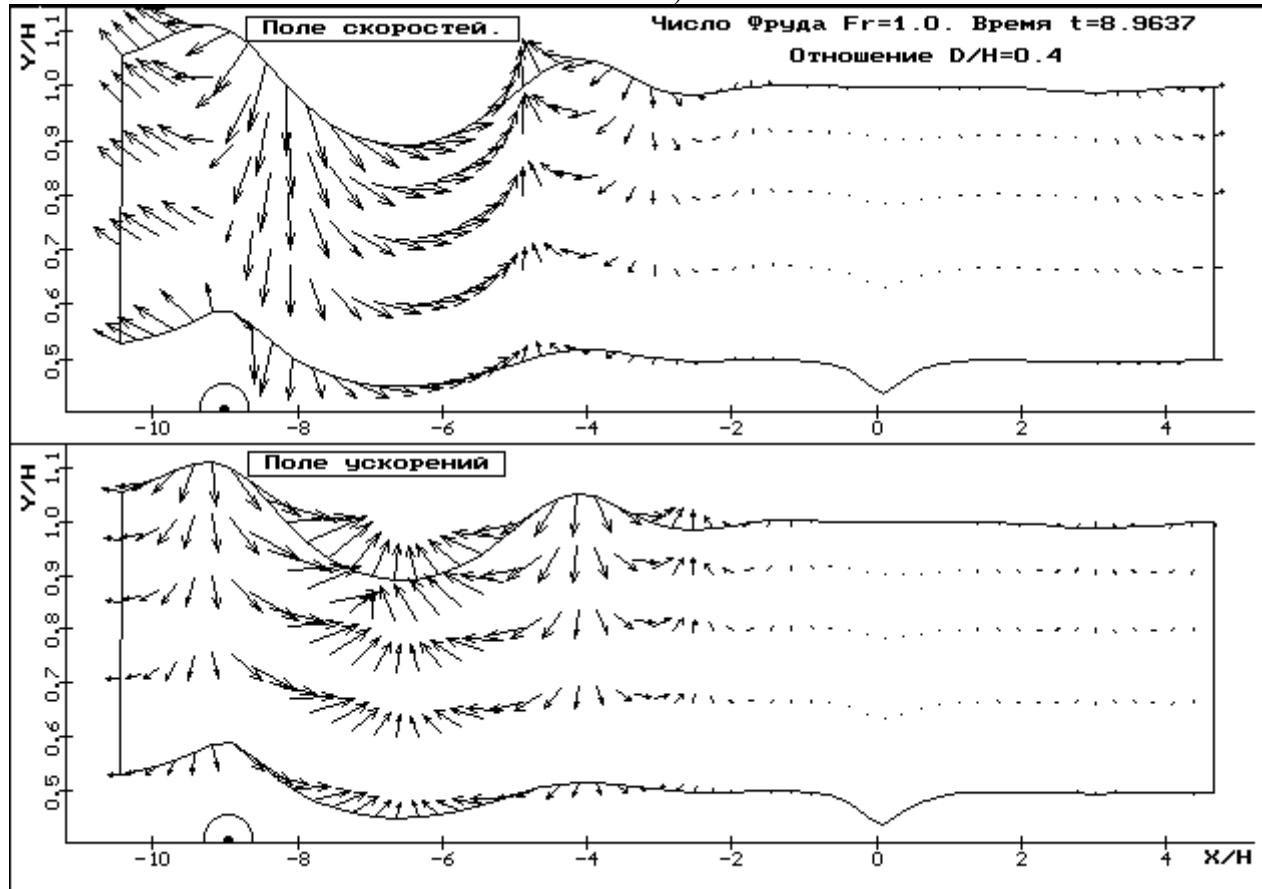
Рис. 55. Рассчитанные векторные поля скоростей и ускорений на гребне волны

Анализируя полученные векторные поля можно заметить, что для числа Фруда $Fr=0,1$ возмущения от тела по жидкости распространяются в обе стороны, причем скорость распространения возмущений выше скорости движения тела. Возмущения из глубины возрастают по мере приближения к свободной поверхности.

С возрастанием числа Фруда возмущения, распространяющиеся в противоположную от тела сторону, быстро затухают, зато перед телом и в непосредственной близости от него поле возмущений сильное, и по-прежнему самые большие возмущения наблюдаются у свободной границы.



a)



б)

Рис. 56. Кинематическая картина течения внутри жидкости

Существенно неустановившиеся течения

При обтекании препятствий ограниченным потоком жидкости существуют режимы, зависящие от числа Фруда, отношения радиуса R тела к глубине H жидкости, при которых не существует стационарных режимов [76]. Данный факт показан в работе [96]. Это же для сверхкритических течений отмечено в работе [18]. В работе [96] показано, что при докритическом течении стационарный волновой цуг может существовать лишь при значениях чисел Фруда Fr , меньших некоторого критического числа $Fr^*(R/H)$. При сверхкритическом режиме обтекания также существуют числа $Fr^{**} > 1$, выше которых возможен выход на стационарный режим течения. В этой работе доказывается, что при $R/H=0,2$ стационарных решений в диапазоне $0,6 < Fr < 1,2$ не существует.

В работе [107] показано и экспериментально доказано, что при околокритическом движении препятствия из состояния покоя возникает течение, не выходящее на стационарный режим. Чтобы проверить эти утверждения и понять природу получаемых течений, был выполнен цикл расчетов (для различных чисел Фруда) по движению с единичной скоростью из состояния покоя полукругового цилиндра, при отношении $R/H=0,2$; при этом расчеты проводились в областях, границы которых показаны на рис. 57. Количество узлов по свободной границе варьировалось от 160 до 270, при общем числе узлов от 280 до 390.

Расчеты проводились в диапазоне чисел Фруда от 0,1 до 100 и были условно разбиты на три группы: I - сверхкритические течения, при которых существует стационарный режим: $1,2 \leq Fr \leq 100$; II - докритические течения, при которых также существуют стационарные решения: $0,1 \leq Fr \leq 0,6$; III – околокритические, течения при которых не существует стационарных решений: $0,6 < Fr < 1,2$. Наиболее изученными являются течения из группы I (рис. 57, а). Здесь известно, что при установившемся течении над телом существует горб, а вверх и вниз по потоку никаких возмущений не распространяется. Теоретически и экспериментально это показано в работах [9,37,78]. При этом существует общая для этой группы закономерность выхода на стационарный режим: в начальный момент движения за телом появляется возмущение, которое, слегка деформируясь, сносится вниз по потоку. Над телом формируется

горб, который по истечении некоторого времени не изменяет своей формы и движется вместе с телом.

Данный режим течения характеризуется еще и тем, что, начиная с некоторого момента времени, сила сопротивления цилиндра стремится к нулю, что является необходимым условием стационарного течения. Коэффициент F силы сопротивления X , действующей на движущееся тело, вычисляется по формуле

$$F = \frac{2X}{\rho V^2 D} = 2 \int_Q P \cos(\vec{n}, x) ds = \sum_{i=1}^{Nt} (P_i + P_{i+1})(y_i - y_{i+1}),$$

где Q - контур тела, \vec{n} - вектор нормали к поверхности тела, Nt – количество точек на теле. Давление P определяется из интеграла Коши-Лагранжа:

$$\frac{P}{\rho} + \frac{\partial \phi}{\partial t} + \frac{1}{2} |\nabla \phi|^2 + \frac{y}{Fr^2} = 0.$$

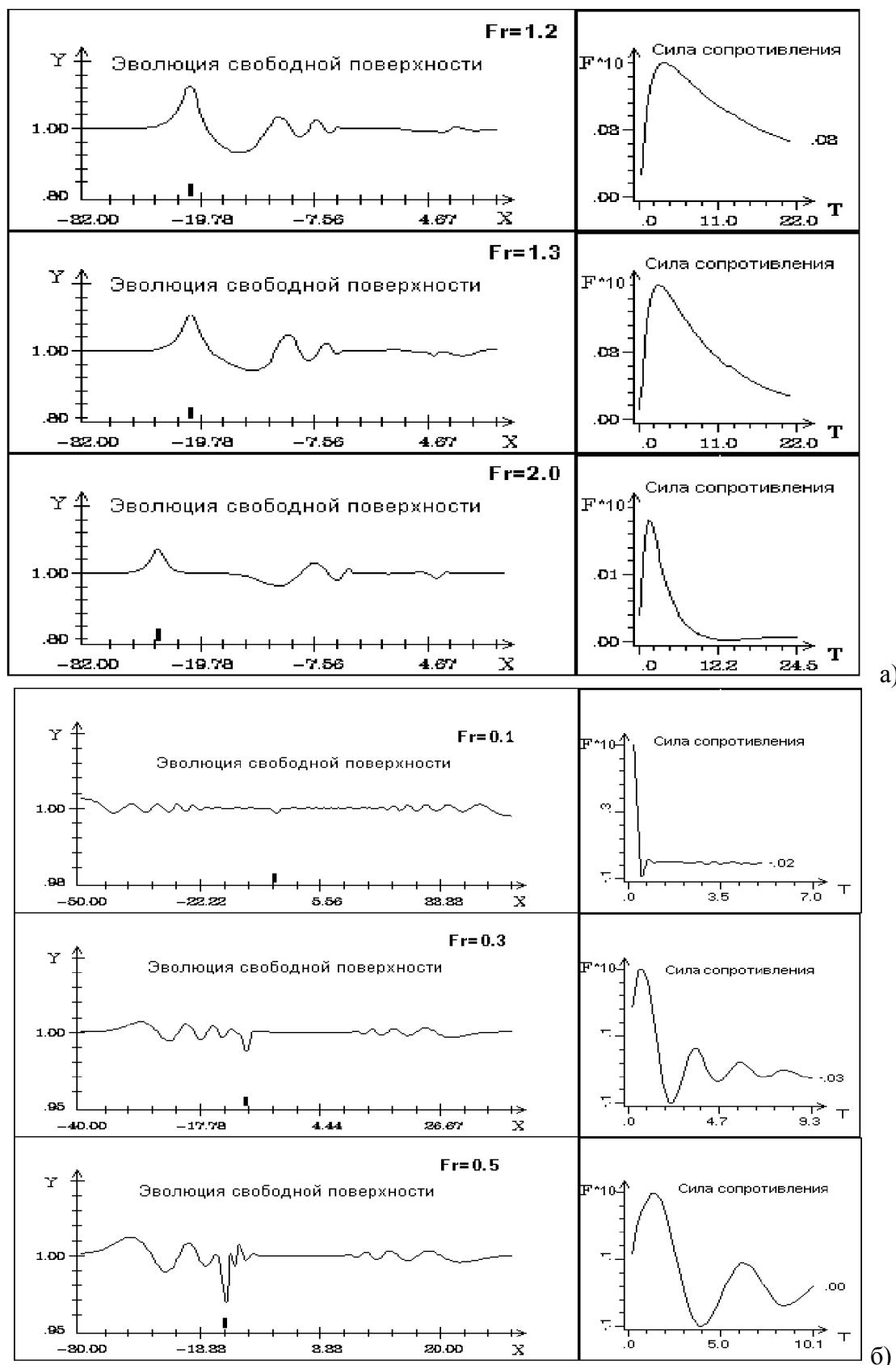
Технология вычисления частных (временных и пространственных) производных описана в работе [14].

Хорошим критерием выхода течения на стационарный режим для данной группы является сравнение формы волны, сформировавшейся над телом с формой волны для чисто стационарной задачи.

На рис.58 для числа Фруда $Fr=2,0$ представлены для сравнения две волны: сформировавшаяся над цилиндром по истечении некоторого периода времени (пунктир) и полученная в результате расчета стационарной задачи обтекания полукругового цилиндрического выступа потоком жидкости конечной глубины [9,35,78] (сплошная линия). Видно хорошее качественное и количественное совпадение результатов. Максимальная относительная погрешность составила 0,2%.

Течения, отнесенные к группе II (рис. 57,б), характеризуются тем, что при движении тела из состояния покоя перед ним формируются гармонические волны, убегающие от тела вверх по потоку, а за телом развиваются гармонические волны меньшей амплитуды. Для этого режима течения характерно то, что график силы сопро-

тивления имеет затухающий волнобразный характер, но по-прежнему сила сопротивления стремится к нулю.



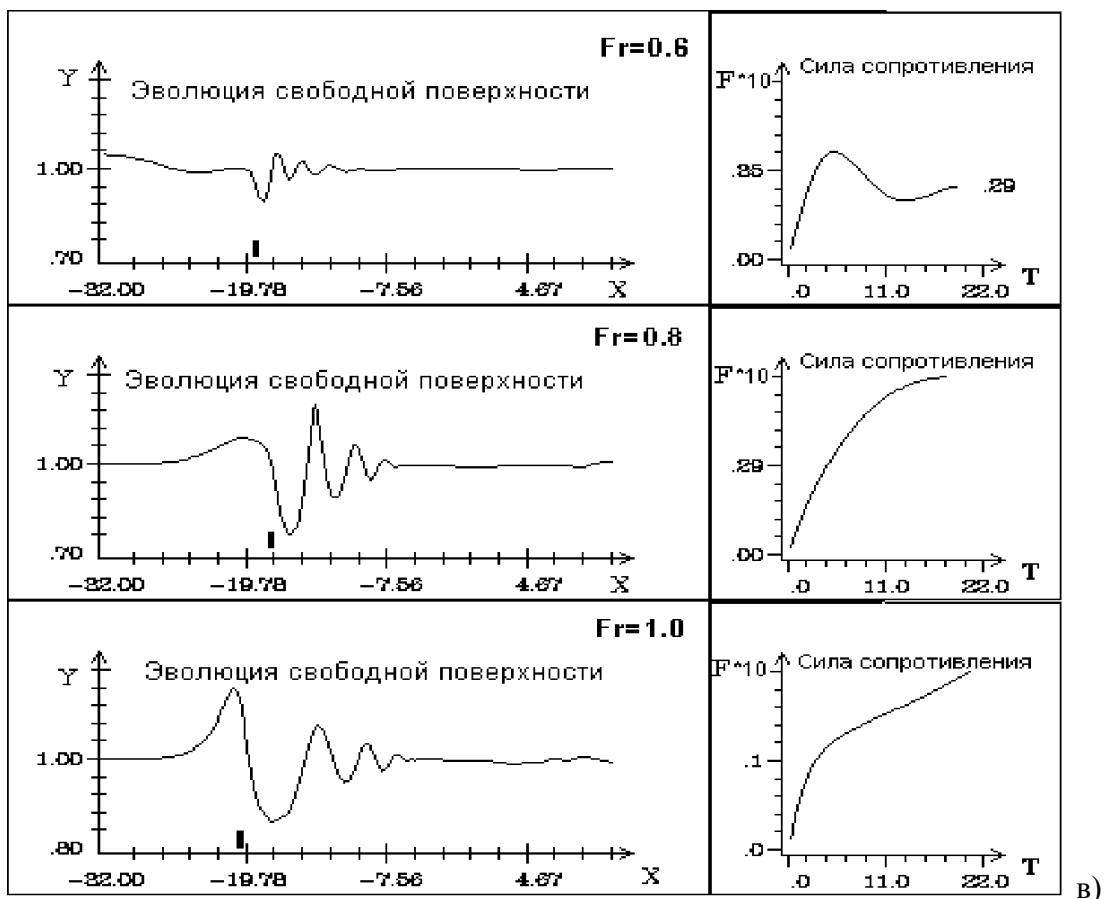


Рис. 57. а – сверхкритические течения (группа I), б – докритические течения (группа II), в – неустановившиеся течения (группа III)

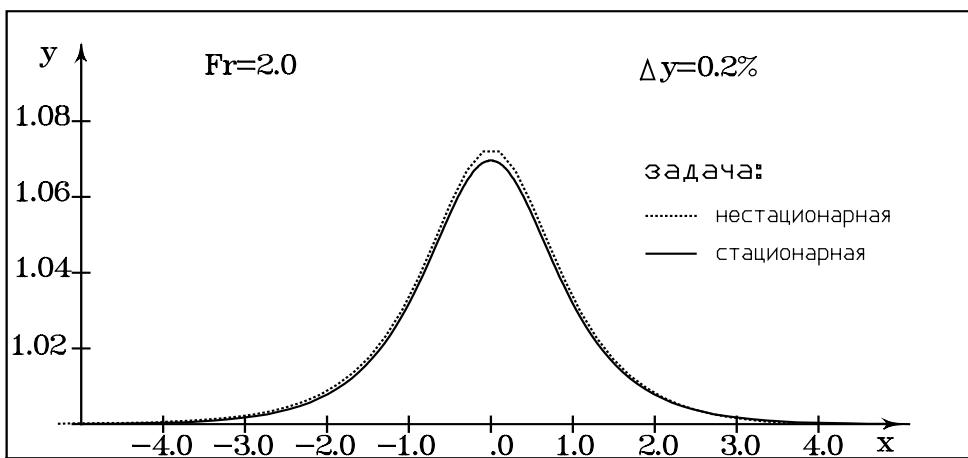


Рис. 58. Сравнение формы волны над телом, полученной из решения нестационарной и стационарной задач

Течения, отнесенные к группе III (рис. 57,в), не должны выходить на стационарный режим. Однако при проведении расчетов каких-либо сложностей, связанных с этим обстоятельством, не возникло. Изучение кинематической картины течения также ничего не показало. Очевидным в этом случае является лишь тот факт, что здесь не выполняется необходимое условие принадлежности реше-

ния к стационарному, то есть сила сопротивления на данных режимах не стремится к нулю и это хорошо видно из графиков для силы сопротивления.

Задача №7. Совместное влияние твердых стенок и свободных границ на эволюцию газового пузыря

Решалась задача о динамике первоначально сферического пузыря, который находился в цилиндрическом сосуде, заполненном жидкостью. В начальный момент времени центр пузыря располагался на расстоянии h от свободной поверхности жидкости $S_2(t)$ и на расстоянии H от твердого дна $R_1(t)$. Боковая стенка сосуда $R_2(t)$ находилась на расстоянии $20R_m$ от центра пузыря для того, чтобы свести к минимуму влияние боковой стенки на процесс схлопывания. Таким образом, моделируется задача об эволюции пузыря между двумя плоскими "бесконечными" границами: твердой стенкой и свободной поверхностью жидкости. Начало системы координат (r, z, θ) совпадает с центром пузыря (см. рис. 59). R_0 - начальный радиус, а R_m - некоторый характерный (максимальный) радиус пузыря.

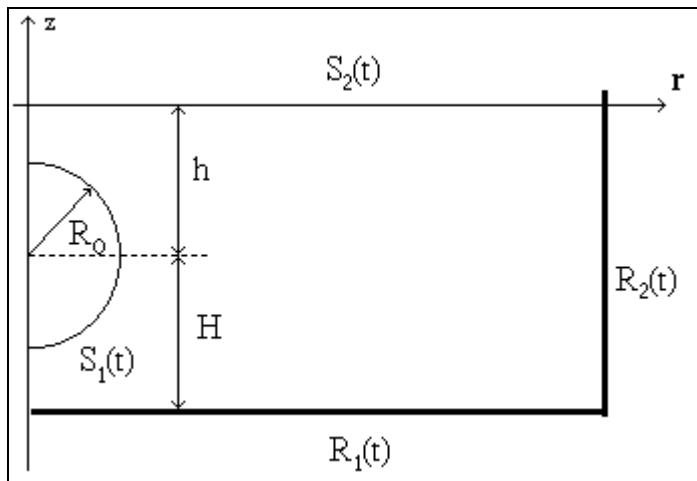


Рис. 59. К задаче о совместном влиянии дна и свободной поверхности

В зависимости от условий задачи начальное распределение потенциала на свободной границе пузыря задавалось двумя способами:

1. Пузырь начинает схлопываться из состояния своего максимально возможного радиуса $R_0 = R_m = 1$. В этом случае на границе

пузыря и свободной поверхности жидкости задается одинаковое распределение потенциала, вытекающее из линейной теории:

$$\phi|_{S_1(0)} = \phi|_{S_2(0)} = 0.$$

2. Пузырь начинает расширяться с начального радиуса R_0 до своего максимального радиуса, а затем схлопывается. На свободной поверхности жидкости распределение потенциала оставалось прежним: $\phi|_{S_2(0)} = 0$; а на границе пузыря задавалось в виде [92]:

$$\phi|_{S_1(0)} = \phi_0^R \left(1 - \frac{R_0}{\left[(z - 2h)^2 + r^2 \right]^{1/2}} \right),$$

где

$$\phi_0^R(R_0, t_\varepsilon) = -R_0 \left[\frac{2}{3} \left(\frac{\Delta P}{\rho} \right) \left\{ \frac{1}{a} - 1 \right\} \right]^{\frac{1}{2}}.$$

Функция $\phi_0^R(R_0, t_\varepsilon)$ представляет собой потенциал, полученный из решения задачи Рэлея о расширении сферической газовой полости от радиуса $R = 0$ до $R = R_0$, аргумент t_ε характеризует время, необходимое пузырю для этого, он может быть выражен через неполные бетта-функции [1]:

$$t_\varepsilon = 3R_0 \left(\frac{3\rho}{2\Delta P} \right)^{\frac{1}{2}} B_a \left(\frac{5}{6}, \frac{2}{3} \right), \quad a = \left(\frac{R_0}{R_m} \right)^3.$$

Если положить, что $R_0 = 0,1$, то ему будут соответствовать значения [90]:

$$\phi_0 = -2,5806976, \quad t_\varepsilon = 0,0015527.$$

На твердых границах (если они существуют) задаются условия непротекания:

$$\left. \frac{\partial \phi}{\partial n} \right|_{R_1(0)} = \left. \frac{\partial \phi}{\partial n} \right|_{R_2(0)} = 0.$$

Начальное распределение давления на свободных границах определялось следующим образом:

- на свободной поверхности жидкости известно начальное (атмосферное) давление $P_a = Const(R_m \rho g)$;
- на границе пузыря давление принимается равным $P_q = P_0 + \rho gh$,

где P_0 - начальное давление газа или паров жидкости внутри пузыря; ρgh - гидростатическое давление жидкости на уровне центра пузыря.

Если считать, что пузырь образовался в жидкости из кавитационного зародыша в результате действия импульса давления, то содержанием газа и паров жидкости, которые выделяются в образовавшуюся полость, можно пренебречь. Если предположить также, что пузырь на начальной стадии имеет сферическую форму, что хорошо согласуется с экспериментальными данными ([44]), а газообразные продукты распределяются равномерно по всему объему, то начальное давление внутри пузыря можно считать постоянным и равным нулю. Кроме того, разность давлений $\Delta P = P_a - P_q$ также считалась постоянной.

Динамическое условие на свободной поверхности жидкости и на границе пузыря имеет вид:

$$\frac{d\phi}{dt} - \frac{1}{2} |\nabla \phi|^2 + Z = 0,$$

$$\frac{d\phi}{dt} - \frac{1}{2} |\nabla \phi|^2 + \alpha + (Z - \gamma) = 0,$$

где

$$\alpha = \frac{\Delta P}{\rho g R_m}; \quad Z = \frac{z}{R_m}; \quad \gamma = \frac{h}{R_m}.$$

Преобразуем граничные условия, введя в них дополнительные параметры, которые бы одновременно определяли положение пузыря относительно свободной поверхности жидкости и твердой стенки. Вместе с тем, для бесконечно удаленных границ - твердой стенки или свободной поверхности - соответствующие параметры должны вносить нулевой вклад в граничные условия.

Представим параметр γ , который определяет относительное положение центра пузыря возле свободной границы, следующим образом:

$$\gamma = \gamma_1 + \gamma_2 = (1 - \omega_1) \frac{h}{R_m} + (1 - \omega_2) \frac{H}{R_m},$$

где

$$\omega_1 = \frac{h - R_m}{h + H} \quad \text{и} \quad \omega_2 = \frac{H - R_m}{h + H},$$

которые также будут определять относительное расположение центра пузыря возле соответствующей границы (ω_1 относительно свободной границы, ω_2 относительно твердой стенки).

Рис. 60. Расширение и схлопывание пузыря с начального радиуса $R_0 = 0,1$ при коэффициенте силы плавучести $\alpha = 1/\delta^2 = 10,014$: а) с учетом влияния только твердой стенки [91]; б) с учетом влияния, как твердой стенки, так и свободной поверхности

Рис. 61. Расширение и схлопывание пузыря с начального радиуса $R_0 = 0,2$ при коэффициенте силы плавучести $\alpha = 1/\delta^2 = 5,005$: а) с учетом влияния только твердой стенки [92]; б) с учетом влияния, как твердой стенки, так и свободной поверхности

На рис. 60 и 61 представлены расчеты, повторяющие результаты, которые получили авторы работы [91], исследуя поведение пузыря возле твердой стенки. Коэффициенты в расчетах подбирались так, чтобы они соответствовали аналогичным коэффициентам из этой работы:

$$\alpha = \frac{\Delta P}{\rho g R_m}; \quad \delta = \left(\frac{\rho g R_m}{\Delta P} \right)^{\frac{1}{2}}; \quad \alpha = \frac{1}{\delta^2}; \quad \gamma_1 + \gamma_2 = \gamma.$$

Кроме формы пузыря, на рисунках показаны зависимости от времени максимального и минимального радиуса пузыря (на графиках обозначены цифрами 1 и 2, соответственно), вертикальной координаты центра пузыря² и нормальных скоростей в двух крайних точках, расположенных на оси симметрии (на графике обозначены цифрами 1 и 2, соответственно).

Результаты на рис. 60,а соответствуют случаю, когда пузырь расширяется с начального радиуса $R_0 = 0,1R_m$ и в первый момент времени находится на расстоянии $H = 1,0R_m$ от твердой стенки в безграничной жидкости. Безразмерные параметры для этого случая принимают значения: $\alpha \approx 10,014$ ($\delta = 0,316$).

² Центральная точка вычисляется как среднее значение от суммы вертикальных координат всех точек, расположенных на границе пузыря.

На рис. 60,b показаны результаты расчета задачи, которая соответствует рис. 60,a, но с той разницей, что теперь в расчетах учитывается влияние свободной поверхности через коэффициент γ_1 (свободная граница жидкости располагалась при этом на расстоянии $h = 5,0R_m$ от центра пузыря). Сравнивая оба рисунка, нетрудно заметить хорошее качественное совпадение результатов, особенно на графиках зависимости от времени максимального и минимального радиусов, а также нормальной скорости в двух крайних точках свободной границы пузыря, расположенных на оси симметрии.

Расчеты, представленные на рис. 61,a,b соответствуют задаче, в которой безразмерные параметры образуют следующую комбинацию значений: $\alpha = 5,005$ (что соответствует параметру из работы [92] $\delta = 0,447$), $H = 1,0R_m$, $\gamma = 1,0$, $h = 5,0R_m$ (для задачи соответствующей рис. 61,b). И в этом случае наблюдается хорошее качественное соответствие обоих результатов.

Для сравнения на рис. 62-63 показаны оригинальные результаты, скопированные из работы [91], и численные расчеты. Жирной линией на обоих рисунках показаны формы пузыря в последние моменты времени перед коллапсом, полученные в работе [91] (рис. 62,a, рис. 63,a), тонкой линией – проведенные расчеты (рис. 62,b и 63,b).

a)

b)

Рис. 62 Формы пузыря в моменты времени, близкие к коллапсу (значения параметров $\sigma = 0,0316$; $\gamma = 1,0$): а) - результаты из работы[91]; б) – результаты численных расчетов.

Как видно из рисунка, численные результаты хорошо согласуются с экспериментальными данными и теорией импульса Кельвина, определяющей направление миграции пузыря в безграничной жидкости.

a)

b)

Рис. 63 Формы пузыря в моменты времени, близкие к коллапсу (значения параметров $\sigma = 0,447$; $\gamma = 1,0$): а) - результаты из работы [92]; б) – результаты численных расчетов

Задача №8. Эволюция пузыря в "сильно" ограниченном объеме

Решается задача о динамике первоначально сферического пузыря, который находится в воронкообразном сосуде, заполненном жидкостью. В начальный момент центр пузыря расположен на расстоянии h от свободной поверхности жидкости $S(t)$ и на расстоянии H от устья сосуда. Боковая стенка сосуда представляет собой совокупность вертикальной стенки $R_2(t)$ и наклонной стенки $R_1(t)$ с устьем $R_3(t)$ в нижней части. Начало системы координат (r, z, θ) совпадает с центральной точкой пузыря (см. рис.64). Пузырь начинает схлопываться из состояния максимально возможного радиуса $R_0 = R_m = 1$.

Выбор такой геометрии для расчета сделан для того, чтобы смоделировать задачу о поведении пузыря в "сильно ограничен-

ном” объеме, где на пузырь оказывает влияние не только свободная поверхность, но и близко расположенные боковые стенки.

Границные условия на свободных границах задаются так же, как и в предыдущей задаче. На стенках сосуда выполняется условие непротекания. Кроме того, в устье воронки может задаваться расход жидкости через границу S (рис. 64, II). В этом случае граничное условие в устье задается в виде

$$\left. \frac{\partial \Phi}{\partial n} \right|_{R_3(t)} = \frac{Q(t)}{S},$$

где $Q(t)$ - заданный расход жидкости, S - площадь поверхности, через которую вытекает жидкость из сосуда.

Начальное распределение давления определяется следующим образом: на свободной поверхности жидкости известно начальное (атмосферное) давление $P_a = Const(R_m \rho g)$; на границе пузыря давление принимается равным $P_q = P_0 + \rho gh$, где P_0 - начальное давление газа или паров жидкости внутри пузыря; ρgh - гидростатическое давление жидкости на уровне центра пузыря. Давление в пузыре изменяется по адиабатическому закону $P_q = P_0 \left(V_q(0) / V_q(t) \right)^{\gamma}$. Пузырь начального радиуса $R_0 = R_m = 1,0$ начинает схлопываться под действием перепада давлений ΔP . Первоначально пузырь располагается в центре сосуда.

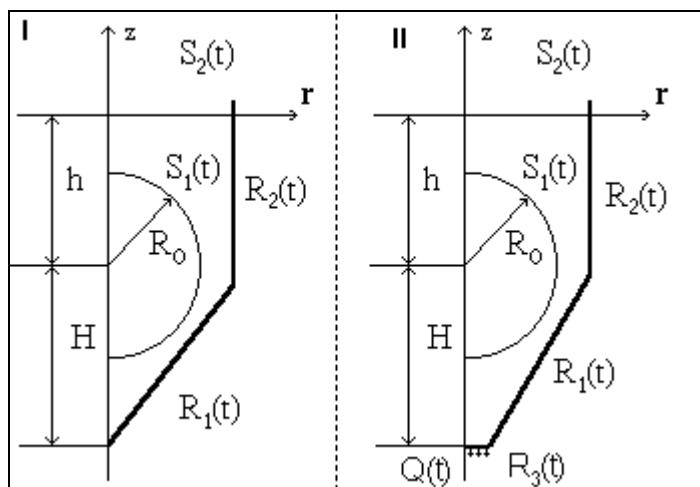


Рис. 64. Геометрия расчетной области

На всех рисунках, приведенных в этой задаче, кроме изменения формы свободной границы пузыря представлены графики зависимости от времени вертикальной координаты центра пузыря, объема, нормальной составляющей скорости в двух крайних точках свободной границы пузыря (они на графике обозначены цифрами 1 и 2) и внутреннего давления в газовой полости.

На рис. 65 представлены результаты расчета задачи о коллапсе пузыря из начального радиуса $R=1,0$. Поскольку с уменьшением объема пузыря увеличивается внутреннее давление, то на последних стадиях развития пузыря наблюдается замедление формирования кумулятивной струйки, что заметно из графика зависимости от времени нормальной составляющей скорости в крайних точках свободной границы пузыря. Здесь нормальная скорость в точке 1 достигает своего абсолютного значения $|\partial \phi / \partial n| = 9,8$ в то время, как нормальная скорость в точке 2 изменяется совершенно незаметно по сравнению с результатами предыдущей задачи. Влияние боковых стенок сосуда на процесс жизни пузыря в таком "ограниченном" объеме превышает влияние силы плавучести и влияние свободной поверхности жидкости вместе взятые. Однако следует отметить тот факт, что пузырь не "прилипает" к твердой стенке, как это происходит в расчетах задачи о поведении пузыря в безграничной жидкости возле твердых стенок.

Рис. 65. Коллапс пузыря начального радиуса 1,0 в поле силы тяжести; внутреннее давление изменяется по адиабатическому закону; пузырь находится в центре сосуда

На рисунке 66 представлены результаты расчетов задачи о схлопывании пузыря в том же воронкообразном сосуде, но с заданным расходом жидкости через границу $R_3(t)$ радиуса $0,1R_m$ (см. схему на рис. 64). Величина расхода жидкости предполагалась постоянной и равной $Q(t) = 0,8$.

Видно, что пузырь хотя и мигрирует по направлению к устью сосуда, однако он не "прилипает" к его стенкам. Смыкание стенок сосуда и пузыря не происходит даже несмотря на то, что истечение жидкости создает дополнительные "трудности" для процесса схлопывания: пузырь засасывается в устье воронки, в некоторых случаях даже не успевая схлопнуться.

Рис. 66. Коллапс пузыря начального радиуса 1,0 в тяжелой жидкости; внутреннее давление изменяется по адиабатическому закону; пузырь находится в центре сосуда; в устье задан расход $Q=0,8$

Задача №9. Исследование явлений на поверхности воды при схлопывании газовой полости

На рис. 67 приведены результаты численных расчетов задачи о схлопывании пузыря начального радиуса $R_0 = 1,0R_m$ (здесь R_m представляет собой характерный линейный размер задачи - максимальный радиус "рэлеевского пузыря", который первоначально располагается на расстоянии $h = 1,5R_m$ от свободной поверхности жидкости. Перепад давлений на свободной поверхности и в пузыре равен $\Delta P = 1,0(\rho g R_m)$, начальное давление в пузыре полагается равным $P_0 = 0,001(\rho g R_m)$.

Начальное состояние задачи изображено на рис. 67,I,a, а состояние системы в моменты времени, близкие к коллапсу пузыря - на рис. 67,I,b. Видно, что пузырь сохраняет свою форму выпуклой на протяжении почти 90% своей жизни. При этом схлопывание происходит при образовании незначительной кумулятивной струй-

ки, которая даже не успевает достаточно сформироваться. В момент коллапса пузыря на свободной поверхности жидкости образуется впадина (рис. 67,II) глубиной $a \approx 0,32R_m$. Известно [47], что дальнейшая эволюция свободной поверхности при наличии впадины приводит к образованию кумулятивной струйки типа "султан". Максимальная высота султана для этого случая составляет $s \approx 0,94R_m$ (рис. 67,III), что сравнимо с начальным радиусом пузыря. Далее сформировавшийся султан под действием силы тяжести начинает распадаться, генерируя расходящиеся волны (рис. 67,IV). В последующие моменты времени султан колеблется с постоянным затуханием. В результате наложения генерируемых колебанием султана и отраженных от берега водоема (расположен на расстоянии $10R_m$) волн на поверхности жидкости образуется рябь. Дальнейшая эволюция приводит к тому, что рябь на свободной поверхности исчезает и образуется одиночная волна с наибольшей максимальной амплитудой $A \approx 0,15R_m$, (что составляет 15% от максимального радиуса пузыря), которая затем начинает распадаться под действием силы тяжести (рис. 67,V).

На рис. 67,VI,a показан график зависимости от времени вертикальной координаты $Y = Z / R_m$, а на рис. 67,VI,b - времени нормальной составляющей скорости в точке свободной поверхности, расположенной на оси симметрии данной задачи. Из графиков заметно, что колебания кумулятивной струйки монотонно затухают и в дальнейшем переходят в почти равномерное колебание жидкости малой амплитуды. При этом, начиная с момента времени $t \approx 30\sqrt{R_m/g}$, процесс эволюции свободной поверхности можно считать установившимся.

На рис. 68 представлены результаты расчетов задачи о расширении пузыря с начального радиуса $R_0 = 0,1R_m$ при тех же начальных условиях. Начальное состояние системы показано на рис. 68,I,a, состояние задачи в момент, когда пузырь достигает своего максимального радиуса $R_{\max} \approx 0,4R_m$ - на рис. 68,I,b.

Из графика зависимости от времени вертикальной координаты точки свободной поверхности, расположенной на оси симметрии (рис. 68,VI,a), видно, что процесс колебания султана затухает гораздо быстрее, чем в предыдущем случае. Здесь процесс эволюции

свободной поверхности можно считать установившимся, уже начиная с момента времени $t \approx 20,0\sqrt{R_m/g}$.

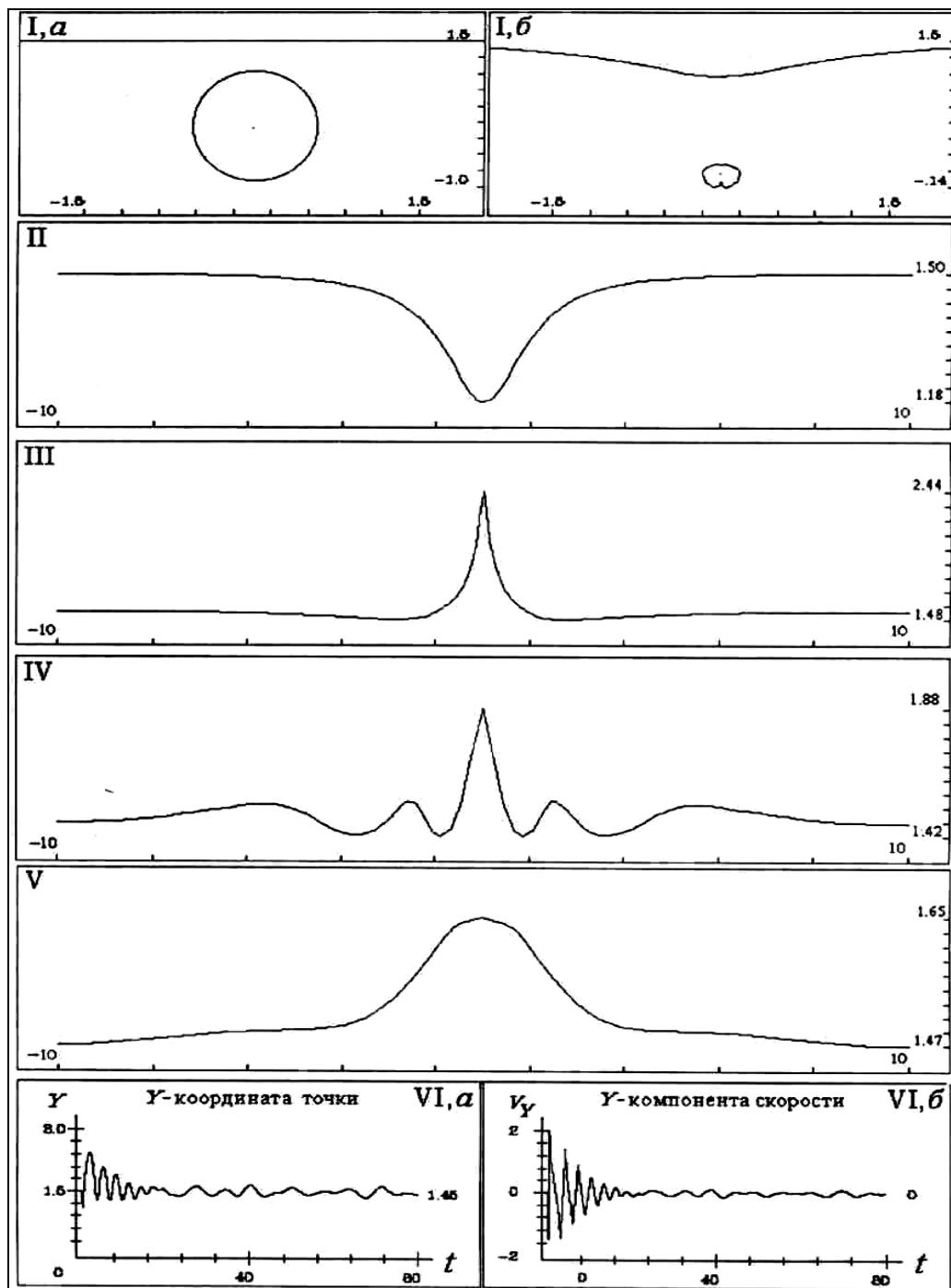


Рис. 67. Возмущение поверхности жидкости при схлопывании пузыря
 $(R_0 = 1,0R_m, P_0 = 0,001(\rho g R_m), h = 1,5R_m)$

В момент коллапса пузыря на свободной поверхности жидкости образуется впадина глубиной $a \approx 0,01R_m$, что составляет 2,5% от максимального радиуса пузыря (рис. 68,II). Для этого случая характерным является образование небольшого султана с более

гладкой границей в вершине (рис. 68,III). Максимальная амплитуда генерируемых при колебании султана волн (рис. 68,IV,V) в этом случае составляет величину $A \approx 0,012R_m$. Дальнейшая картина течения повторяет результаты предыдущей задачи с той разницей, что амплитуды сгенерированных волн составляют величину порядка 2-3% от максимального радиуса пузыря.

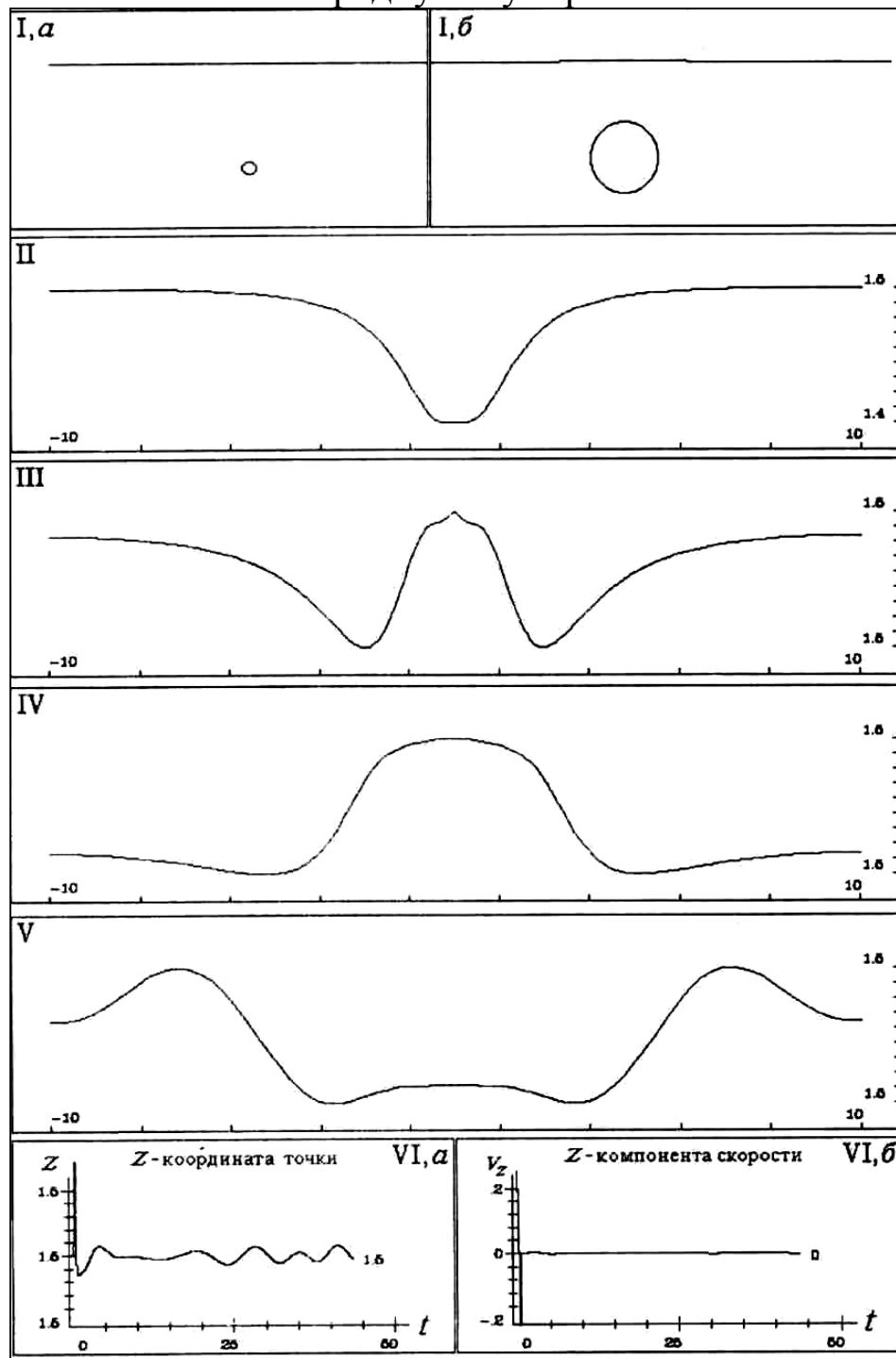


Рис. 68. Возмущение поверхности жидкости при схлопывании пузыря

$$(R_0 = 0,1R_m, P_0 = 0,001(\rho g R_m), h = 1,5R_m)$$

Результаты расчетов следующей задачи о схлопывании пузыря начального радиуса $R_0 = 0,1R_m$ при начальном внутреннем давлении $P_0 = 10,0(\rho g R_m)$ представлены на рис. 69. Пузырь в первые моменты времени располагается на расстоянии $h = 1,0R_m$ от свободной поверхности жидкости. На рис. 69,I показано начальное состояние системы в тот момент, когда пузырь начинает расширяться под действием своего внутреннего давления, на рис. 69,II - состояние задачи незадолго до момента времени, когда верхняя граница пузыря соприкасается со свободной поверхностью жидкости.

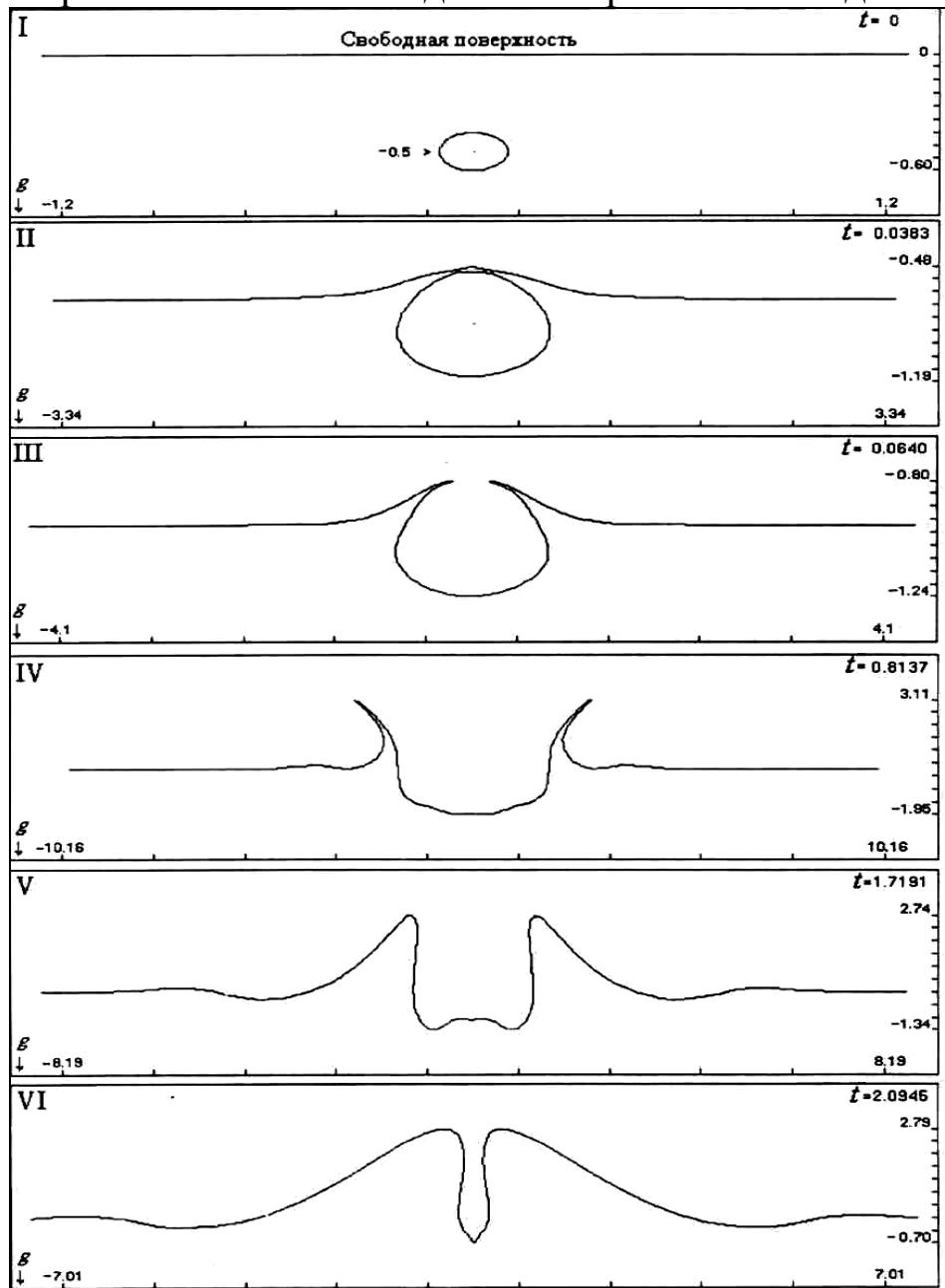


Рис. 69. Возмущение поверхности жидкости при схлопывании пузыря ($R_0 = 0,1R_m$,
 $P_0 = 1,0(\rho g R_m)$, $h = 1,0R_m$)

При расчетах предполагалось, что последнее происходит в том случае, если минимальное расстояние от точек свободной поверхности до точек, расположенных на верхней границе пузыря, составляло величину $d \leq 0,002R_0$, т.е. не превышающую 0,2% от начального радиуса пузыря. После этого группа точек на свободной поверхности и границе пузыря, удовлетворяющая вышенназванному условию, удалялась из рассмотрения.

Положение свободной поверхности, изображенное на рис. 69,III, соответствует моменту времени, когда газ из пузыря вышел в атмосферу. Теперь поверхность жидкости представляет собой выемку, расположенную на месте бывшего пузыря, с наличием боковых брызговых струй и ту часть свободной поверхности, которая деформировалась при приближении к ней пузыря. Видно, что в результате дальнейшей эволюции брызговые струйки выпрямляются (рис. 69,IV) и далее разворачиваются на внешнюю сторону выемки. При этом ширина выемки неуклонно сокращается. На рис. 69,V заметно образование кумулятивной струи в центре "подошвы" выемки.

Однако в дальнейшем боковые стенки выемки становятся почти вертикальными, брызговые струи разворачиваются и опять смыкаются, образуя полость с внутренним давлением, равным атмосферному (рис. 69,VI). Кумулятивная струя в подошве выемки так и не успевает достаточно сформироваться. После нового замыкания и образования каверны численные расчеты прекращались.

ЛИТЕРАТУРА

1. Абрамович М., Стиган И. Справочник по специальным функциям. М.: Наука, 1979.
2. Аммерал Л. Принципы программирования в машинной графике. М.: СолСистем, 1992.
3. Апарчин А.С. Применение метода квадратурных сумм к решению некоторых интегральных уравнений Фредгольма I рода // Дифференц. и интегр. ур-ния. Иркутск: Иркутский гос. ун-т, 1975. Вып. 3. С. 106-119.
4. Афанасьев К.Е. Моделирование свободных границ в гидродинамике идеальной жидкости // Гидродинамика ограниченных потоков / Чуваш. гос. ун-т им. И.Н. Ульянова. Чебоксары, 1988. С. 9-18.
5. Афанасьев К.Е. Приближение нелинейной уединенной волны // Труды VI научной школы "Гидродинамика больших скоростей" / Чуваш. гос. ун-т им. И.Н. Ульянова. Чебоксары, 1996. С. 3 - 10.
6. Афанасьев К.Е. Моделирование сильно нелинейных волновых течений // Вычислительные технологии. Новосибирск, 1998. Т. 3, № 1. С. 3-12.
7. Афанасьев К.Е. Решение нелинейных задач гидродинамики идеальной жидкости со свободными границами методами конечных и граничных элементов // Автореф. дис. ..докт. физ.-мат. наук. Кемерово, 1997. 40 с.
8. Афанасьев К.Е., Афанасьева М.М., Терентьев А.Г. Деформация газовых пузырей в жидкости // Актуальные задачи гидродинамики / Чуваш. гос. ун-т им. И.Н. Ульянова. Чебоксары, 1989. С. 4-10.
9. Афанасьев К.Е., Афанасьева М.М., Терентьев А.Г. Исследование эволюции свободных границ методами конечных и граничных элементов при нестационарном движении тел в идеальной несжимаемой жидкости // Изв. АН СССР, МЖГ. 1986. № 5. С. 8-13.
10. Афанасьев К.Е., Гудов А.М. Численное моделирование динамики пространственного пузыря методом граничных элементов // Моделирование в механике. Новосибирск, 1994. Т. 7(24). № 1. С. 11-19.

- 11.Афанасьев К.Е., Гудов А.М. Эволюция цепочки из трех пузырей в безграничной жидкости // Динамика сплошных сред со свободными границами / Чуваш. гос. ун-т им. И.Н. Ульянова. Чебоксары, 1996. С. 31-41.
- 12.Афанасьев К.Е., Гудов А.М., Коротков Г.Г., Долаев Р.Р., Березин Е.Н. Распределенный пакет прикладных программ «*AKORD*» для проведения вычислительных экспериментов // Моделирование, вычисления, проектирование в условиях неопределенности – 2000: Труды научной международной конференции. Уфа: Гос. авиационный техн. ун-т, 2000. С. 47-57.
- 13.Афанасьев К.Е., Коротков Г.Г., Долаев Р.Р. Разработка пакета прикладных программ «*AKORD*» для решения задач со свободными границами // Вычислительные технологии. Новосибирск, 2000. Т. 5, № 1. С. 5-18.
- 14.Афанасьев К.Е., Самойлова Т.И. Техника использования метода граничных элементов в задачах со свободными границами // Вычислительные технологии. Новосибирск, 1995. Вып. 7, № 11. С. 19-37.
- 15.Афанасьев К.Е., Стуколов С.В. Моделирование опрокидывающихся волн методом комплексных граничных элементов // Труды VI научной школы "Гидродинамика больших скоростей"/ Чуваш. гос. ун-т им. И.Н. Ульянова. Чебоксары, 1996. С. 11 - 17.
- 16.Афанасьев К.Е., Стуколов С.В. Накат уединенной волны на наклонный берег // Вестник Омского ун-та. Омск, 1998. № 3. С. 9-12.
- 17.Афанасьев К.Е., Стуколов С.В. Численное моделирование взаимодействий уединенных волн с препятствиями // Вычислительные технологии. Новосибирск, 2000. Т. 4, № 6. С. 3-15.
- 18.Афанасьев К.Е., Стуколов С.В. О наличии трех решений при обтекании препятствий сверхкритическим установившимся потоком тяжелой жидкости // ЖВММФ. 1999. Т. 40, № 1. С. 27-35.
- 19.Бахвалов Н.С. Численные методы. М.: Наука, 1975.
- 20.Белых В.Н. Численные алгоритмы без насыщения в нестационарных задачах гидродинамики идеальной жидкости со свободными границами // Тр. Ин-та мат. СО АН СССР. 1988. Т. 11. С. 3-67.
- 21.Бенерджи П., Баттерфилд Р. Методы граничных элементов в прикладных науках. М.: Мир, 1984.

- 22.Березин И.С., Жидков Н.П. Методы вычислений. М.: Физматгиз, 1966. Т.1.
- 23.Бобровски С. Oracle 7 и вычисления клиент/сервер. М.: ЛОРИ, 1996.
- 24.Бреббия К., Теллес Ж., Вроубел Л. Методы граничных элементов. М.: Мир, 1987.
- 25.Валландер С.В. Лекции по гидроаэромеханике. Ленинград: из-во Ленинградского ун-та, 1978.
- 26.Вирт Н. Алгоритмы и структуры данных. М.: Мир, 1989.
- 27.Воронин В.В., Цецохо В.А. Численное решение интегрального уравнения I рода с логарифмической особенностью методом интерполяции и коллокации // ЖВММФ. 1981. Т. 21, № 1. С. 40-50.
- 28.Воронин В.В., Цецохо В.А. О прямом методе решения интегральных уравнений I рода с особенностями в ядрах // Методы решения некорректных задач и их приложения. Новосибирск: ВЦ СО АН СССР, 1982.
- 29.Годунов С.К., Забродин А.В., Иванов М.Я. и др. Численное решение многомерных задач газовой динамики. М.: Наука, 1976.
- 30.Григорьева И.В., Гудов А.М. Препроцессор для расчета пространственных задач со свободной поверхностью / Вычислительные технологии, ИВТ СО РАН. Новосибирск, 1999. Т. 4, № 6. С. 68-76.
- 31.Григорьева И.В., Гудов А.М., Коротков Г.Г. Принципиальная схема пакета прикладных программ в курсе «методы математической физики» // Новые информационные технологии в университете образовании: Сборник трудов. Новосибирск: ИДМИ, 1999. С. 34-35.
- 32.Громадка Т., Лей Ч. Комплексный метод граничных элементов. М.: Мир, 1990.
- 33.Гудов А.М. Численное моделирование возмущений свободной поверхности, вызванных коллапсом газового пузыря // Вычислительные технологии. Новосибирск, 1993. Т. 4, № 11. С. 92-103.
- 34.Гудов А.М. Численное исследование явлений на поверхности воды при схлопывании газовой полости // Вычислительные технологии. Новосибирск, 1997. Т. 2, № 4. С. 49-59.
- 35.Гузевский Л.Г. Обтекание препятствий потоком тяжелой жидкости конечной глубины // Динамика сплошных сред с границами раздела / Чуваш. госун-т им. И.Н. Ульянова, 1982. С. 61-69.

- 36.Гуревич М.И. Теория струй идеальной жидкости. М.: Наука, 1979.
- 37.Долина И.С., Ермаков С.А., Пелиновский Е.Н. Смещение свободной поверхности жидкости при обтекании цилиндра // ПМТФ, 1988. № 4. С. 48-51.
- 38.Епанешников А.М., Епанешников В.А.. Программирование в среде Turbo Pascal 7.0. М.: Диалог-Мифи, 1996.
- 39.Житников В.П. Обобщение метода Леви-Чивиты для исследования плоских и осесимметричных течений с нелинейными условиями на неизвестных границах: Автореф. дис. ... докт. физ.-мат. Наук. Казань, 1993. 32 с.
- 40.Иванов А.Н. Гидродинамика развитых кавитационных течений. Л.: Судостроение.- 1980.
- 41.Канторович Л.В., Крылов В.И. Приближенные методы высшего анализа. М.-Л.: Гифмл, 1962.
- 42.Кедринский В.К. Модели М.А. Лаврентьева в задачах неустановившихся течений со свободными границами // Проблемы математики и механики. Новосибирск: Наука, 1983. С. 97-116.
- 43.Коробицин В.А., Пегов В.И. Численное исследование эволюции границы раздела двух жидкостей // МЖГ. 1993. № 5. С. 128-133.
- 44.Коул Р. Подводные взрывы. М.: ИЛ, 1950.
- 45.Кочин Н.Е., Кибель И.А., Розе Н.В. Теоретическая гидромеханика. М.: ГИФМЛ, 1963. Т. 1.
- 46.Крылов В.И. Приближенное вычисление интегралов. М: ФМЛ, 1967.
- 47.Лаврентьев М.А., Шабат Б.В. Проблемы гидродинамики и их математические модели. М.: Наука, 1977.
- 48.Левковский Ю.Л. Структура кавитационных течений. Л.: Судостроение, 1978.
- 49.Логвинович Г.В. Гидродинамика течений со свободными границами. Киев: Наукова Думка, 1969.
- 50.Ламб Г. Гидродинамика.: М.-Л, 1947.
- 51.Мадер Ч.Л. Взрывы вблизи поверхности воды // Подводные и подземные взрывы. М.: Мир, 1974. С. 74-120.
- 52.Мазья В.Г. Границные интегральные уравнения // Итоги науки и техники. ВИНИТИ. Соврем. пробл. мат.: Фундам. направления. 1988. Т. 27. С. 131-228.

- 53.Маклаков Д.В. Плоские нелинейные задачи безотрывного, кавитационного и волнового обтекания препятствий однородными и двухслойными потоками: Автореф. дис. ... докт. физ.-мат. наук.- Казань, 1995. 34 с.
- 54.Маклаков Д.В. Нелинейные задачи задачи потенциальных течений с неизвестными границами. М.: Янус-К, 1997.
- 55.Манойлин С.В. Некоторые экспериментально-теоритические методы определения воздействия волн цунами на гидротехнические сооружения и акватории морских портов. Красноярск. ВЦ СО АН СССР. 1989. Препринт № 5.
- 56.Мартин Дж. Организация баз данных в вычислительных системах. М.: Радио и связь, 1980.
- 57.Марчук Ан.Г., Чубаров Л.Б., Шокин Ю.И. Численное моделирование волн цунами. Новосибирск: Наука, 1983.
- 58.Марчук Г.И., Агошков В.И. Введение в проекционно-сеточные методы. М.: Наука, 1981.
- 59.Матросов В.М. Пакет прикладных программ. М.: Наука, 1989.
- 60.Михлин С.Г. Интегральные уравнения. М.;Л.: Гостех-издат, 1947.
- 61.Моисеев Н.Н. О неединственности возможных форм установившихся течений тяжелой жидкости при числах Фруда, близких к единице // Прикладная математика и механика. 1957. Т. 21, № 6. С. 860 - 864.
- 62.Нох В.Д. СЭЛ - совместный эйлеро-лангранжев метод для расчета нестационарных двумерных задач // Вычислительные методы в гидродинамике. М.: Мир, 1967. С. 128-184.
- 63.Овсянников Л.В. О всплытии пузыря // Некоторые проблемы математики и механики. Л.:Наука, 1970. С. 209-222.
- 64.Перегрин Д. Разрушение волн на отлогих берегах // Нелинейные волновые процессы. Сер. Механика: Новое в зарубежной науке. М.: Мир, 1987. Т. 42. С. 37-71.
- 65.Петров А.Г., Смолянин В.Г. Расчет нестационарных волн на поверхности тяжелой жидкости конечной глубины // ПММ. 1993. Т. 57, в. 4. С. 137-143.
- 66.Плотников П.И. Неединственность решения задачи об уединенных волнах и бифуркации критических точек гладких функционалов // Изв. АН СССР. Сер. мат. 1991. Т. 55, № 2. С. 339 - 366.

- 67.Протопопов Б.Е. Численный анализ трансформации уединенной волны при отражении от вертикальной преграды // Изв. АН. Механика жидкости и газа. 1990. № 5. С. 115-123.
- 68.Пэйдж В., Хьюз Н., Остин Д. и др. Использование Oracle 8. К.& М.; Спб.: Издат. Дом «Вильямс», 1998.
- 69.Райс Дж. Матричные вычисления и математическое обеспечение. М.: Мир, 1984.
- 70.Роджерс Д. Алгоритмические основы машинной графики. М.: Мир, 1989.
- 71.Рузиев Р.А., Хакимзянов Г.С. Численное моделирование трансформации уединенной волны над подводным уступом // Вычислительные технологии. ИВТ СОРАН. 1992. Т.1, № 1. С. 5-21.
- 72.Самарский А.А., Гулин А.В. Численные методы: Учебн. пособие для вузов. М.: Наука. Гл. ред. физ.-мат. лит, 1989.
- 73.Седов Л.И. Механика сплошной среды. Том I,II. М.: Наука, 1973.
- 74.Сингх Л. ORACLE 7.3 Пособие разработчика. К.: Издательство «ДиаСофт», 1997.
- 75.Сретенский Л.Н. Теория волновых движений жидкости. М.: Наука, 1972.
- 76.Стурова И.В. Численные расчеты в задачах генерации плоских поверхностных волн. Красноярск: ВЦ СО РАН, 1990. Препринт № 5. 48 с.
- 77.Сурков К., Сурков Д., Вальвачев А.. Программирование в среде Delphi 2.0. Минск: Попурри, 1997.
- 78.Терентьев А.Г., Афанасьев К.Е. Численные методы в гидродинамике: Учеб. пособие / Чуваш. ун-т. им. И.Н. Ульянова. Чебоксары: ЧГУ, 1987.
- 79.Терентьев А.Г., Картузова Т.В. Численные исследования системы крыловых профилей методом граничных элементов // Актуальные задачи математики и механики / Чуваш. ун-т. Чебоксары, 1995. С. 108-116.
- 80.Тихонов А.Н., Арсенин В.Я. Методы решения некорректных задач. М.: Наука, 1979.
- 81.Тихонов А.Н., Самарский А.А. Уравнения математической физики. М.: Наука, 1977.

82. Трушников В.Н. Один нелинейный регуляризирующий алгоритм и некоторые его приложения // ЖВМ и МФ. 1979. Т. 19, № 4. С. 822-829
83. Филиппов А.Т. Многоликий солитон. М.: Наука: библиотечка "Квант", 1986. В. 48.
84. Форсайт Дж., Малькольм М., Моулер К. Машины методы математических вычислений. М.: Мир, 1980.
85. Франк А.М. Дискретные модели несжимаемой жидкости : Автoref. дис. ... докт. физ.-мат. наук. Новосибирск, 1994. 30 с.
86. Шварц Л., Фентон Дж. Сильно нелинейные волны // Нелинейные волновые процессы. Сер. Механика: Новое в зарубежной науке. М.: Мир, 1987. Т. 42. С. 10-37.
87. Шерыхалина Н.М. Математическое моделирование течений вязкой жидкости со свободными поверхностями, индуцированных погруженным источником: Автoref. дис. ... канд. физ.-мат. наук. Уфа, 1996. 18 с.
88. Шокин Ю.И., Рузиев Р.А., Хакимзянов Г.С. Численное моделирование плоских потенциальных течений жидкости с поверхностными волнами. Препринт. Красноярск.: ВЦ СО РАН, 1990. № 12. 37 с.
89. Чубаров Л.Б. Численное моделирование волн цунами: Автoref. дис. ... докт. физ.-мат. наук.- Новосибирск, 2000. 33 с.
90. Эйнарсон Бо, Шокин Ю.И. ФОРТРАН-90. Книга для программирующих на языке ФОРТРАН-77: Пер. с англ. Новосибирск: издат. СО РАН, «ИНФОЛИО».- 1995.
91. Blake J.R., Taib B.B., Doherty G. Transient cavities near boundaries. Part 1. Rigid boundary // J. Fluid Mech. 1986. № 170. P. 479-498.
92. Blake J.R., Taib B.B., Doherty G. Transient cavities near boundaries. Part 2. Free surface // J. Fluid Mech. 1987. № 181. P. 197-212.
93. Blake J.R., Robinson P.B., Shima A. and Tomita Y. Interaction of two cavitation bubbles with a rigid boundary // J. Fluid Mech. 1993. V. 255/ P. 707-721.
94. Cooker M.J., Peregrine D.H., Vidal C., Dold J.W. The interaction between a solitary wave and a submerged semicircular cylinder // J. Fluid Mech. 1990. V. 215. P. 1-22.
95. Domermuth G.D., Yue D.K.P. Numerical simulations of nonlinear axisymmetric flows with a free surface // J. Fluid Mech. 1987. V. 178. P. 195-219.

96. Forbes L.K., Schwartz L.W. Free surface flow over a simicircular obstruction // J. Fluid Mech. 1982. V. 114. P. 229-314.
97. Ghassemi F. Automatic mesh generation scheme/ Computing and Structures, 1982. Vol.15, №.6. P. 613-626.
98. Longuet-Higgins M.S. On the mass, momentum, energy and circulation of a solitary wave // Proc. R. Soc. Lond. 1974. A. 337. P. 1-13.
99. Longuet-Higgins M.S., Cokelet E.D. The deformation steep surface waves on water. 1. A numerical method of computation // Proc. Roy. Soc., A. 1976. V. 350. P. 1-26.
100. Nakayama T. A computational method for simulating transient motions of an incompressible inviscid fluid with a free surface // Int. J. Numer. Meth. Fluids. 1990. V. 10. P. 683-695.
101. Plesset M. S., Prosperetti A. Bubble dynamics and cavitation // Annu. Rev. Fluid Mech, Calif. 1977. V. 9. P. 145-185.
102. Plesset M.S., Chapman R.B. Collapse of an initially shperical vapour cavity in the neighbourhood of a solid boundary // J. Fluid Mech. 1972. V. 47, Part II. P. 283-290.
103. Rayleigh Lord. On the pressure developed in a liquid during the collapse of a spherical cavity // Phil. Mag. 1917. V. 34. P. 94-98.
104. Seabra-Santos F.J., Renouard D.P., Temperville A.M. Numerical and experimental study of the transformation of a solitary wave over a shelf or isolated obstacle // J. Fluid Mech. 1987. V. 176. P. 117-134.
105. Vanden-Broeck J.M. Free-surface flow over an obstruction in a channel // J. Phys. Fluids. 1987. V. 30, № 8. P. 2315-2317.
106. Vinje T., Brevig P. Numerical simulation of breaking waves // Adv. Water Resour. 1981. V. 4. P. 77-82.
107. Wu T.Y.-T. Generation of upstream advansing solitonsby moving disturbances // J. Fluid Mech. 1987. V. 184. P. 75-99.

ПРИЛОЖЕНИЯ

Приложение 1. Работа с препроцессором для плоских и осесимметричных задач

Для удобства работы с препроцессором опишем пошагово работу с ним.

ОПИСАНИЕ ПУНКТОВ МЕНЮ

File

Пункт <File> главного меню содержит следующее пункты выбора, которые, в свою очередь, могут представлять собой меню более низкого уровня - подменю:



"**New...**". Этот пункт меню предназначен для создания нового файла данных. При его вызове на экране появляется диалоговое окно, изображенное на рис. 70. Переключатель <2D, 2,5D> задает тип задачи: плоская или осесимметричная. Ниже находится окно редактирования, в котором вводится имя создаваемого файла данных (по умолчанию noname.dat). Для подтверждения введенных данных необходимо нажать кнопку <OK>, для отмены - кнопку <Cancel>. При нажатии на кнопку <OK> на экране возникнет новое графическое окно, заголовок которого будет имем нового файла.

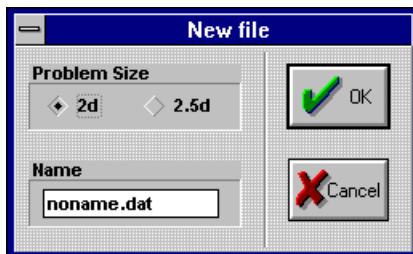


Рис. 70. Графическое окно "New File"

"Open...". Пункт меню предназначен для загрузки с диска файла, содержащего сделанный ранее рисунок. В дальнейшем вы можете работать с этим рисунком обычным образом. Файл будет загружен в окно отображения данных препроцессора после заполнения полей специальной графической формы (рис. 71).

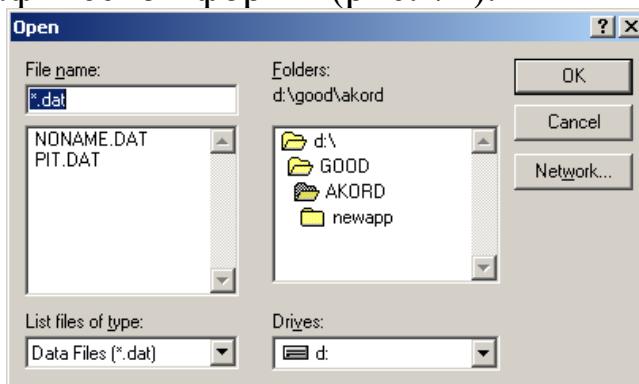


Рис. 71. Графическое окно для загрузки файла

"Save...". При выборе данного пункта меню рисунок, находящийся в активном окне, записывается в файл под именем которое находится в заголовке окна.

"Save as...". Пункт меню предназначен для записи рисунка в файл с именем отличным от того, которое находится в заголовке окна. При этом появляется диалоговое окно, представленное на рис. 72. Файлы в списке выделены серым цветом. Здесь имеется возможность выбирать имена файлов из списка, редактировать их или вводить полностью новые имена.

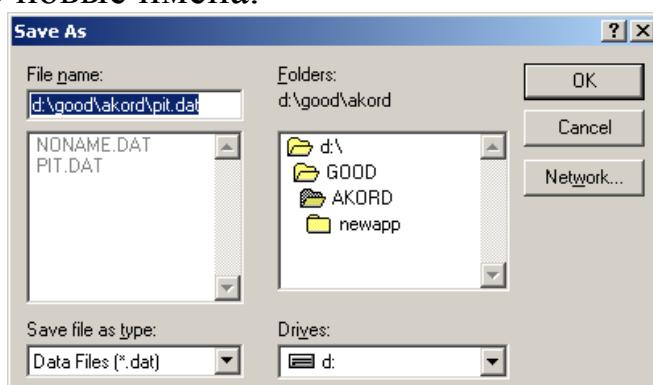


Рис. 72. Окно для выбора имени файла

Info Point Parameters ...
Time Parameters ...
Physical Parameters ...
Domain Parameters ...
Draw , Output Parameters ...
Other Parameters ...

"Exit". Выход из программы. Если рисунок не был сохранен, то перед выходом запрашивается подтверждение на сохранение результатов.

Edit

Меню “Edit” предназначено для ввода различных дополнительных данных необходимых для нормальной работы решателя. Все вводимые в диалоговых формах значения переменных никак не влияют на работу самого препроцессора, а выводятся в результирующий файл в исходном виде.

“Info Point Parameters...”. В этом окне (рис. 73) выводится информация об общем количестве граничных точек в области, о количестве подобластей и о числе внутренних точек.

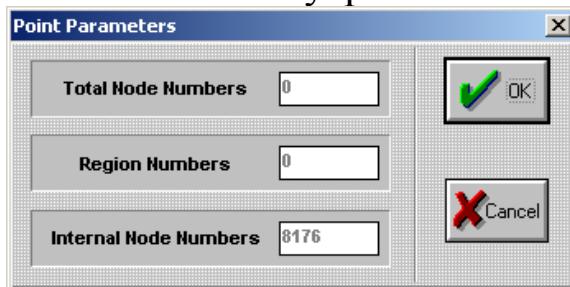


Рис. 73. Окно для вывода информации о количестве точек в рассматриваемой области решения

“Time Parameters...”. В данном окне (рис. 74) вводятся временные параметры такие, как минимальный и максимальный шаг по времени, начальное время и ограничение на максимальное значение времени решения задачи.

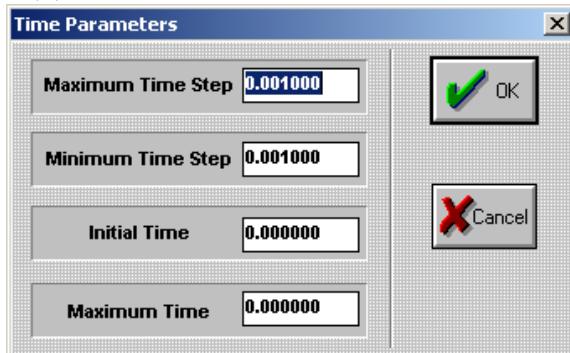


Рис. 74. Форма для ввода параметров времени

“Physical Parameters...”. В этом диалоговом окне (рис. 75) задаются различные физические параметры, необходимые для численного решения задачи:

- “Esstimate” - максимально возможная погрешность;
- “Surface Pressure” - поверхностное давление;
- “Liquid Density” - плотность жидкости;
- “Surface Tention Coeff.” - коэффициент поверхностного натяжения;
- “Adiabate Indicator” - показатель адиабаты.

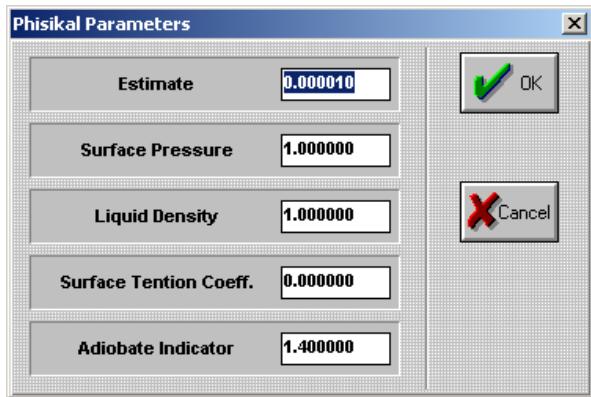


Рис. 75. Форма для задания физических параметров задачи

"**Domain Parameters...**". В диалоговом окне, показанном на рис. 76, задаются параметры области решения задачи:

- "Domain boundary" - признак замкнутости области ("Detached" - замкнутая, "Undetached" – незамкнутая);
- "Domain type" - признак ограниченности области ("Bound" - всюду ограниченная, "Unbound" - безгранична);
- "Normal position" - задает направление нормали к границе области ("Inside" – нормаль направлена вовнутрь области, "Outside" – вовне) ;
- "Normal direction" - направление обхода;
- "Gravitation" – признак, указывающий на то, чтобы решатель учитывал (или нет) силу тяжести при расчетах.

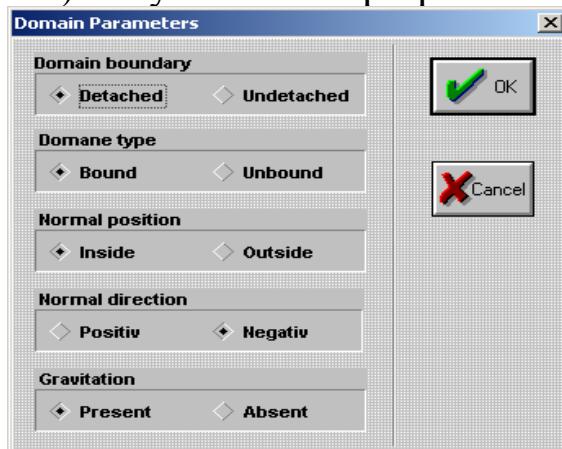


Рис. 76. Форма определения параметров области решения

"**Other Parameters...**". Так как заранее учесть все необходимые параметры достаточно сложно, была предусмотрена возможность ввода дополнительных данных, характерных для конкретного типа решателя. Диалоговое окно (рис. 77) предназначено для создания и редактирования пользовательского блока данных (смотри раздел «Интерфейс обмена данными между приложениями»).

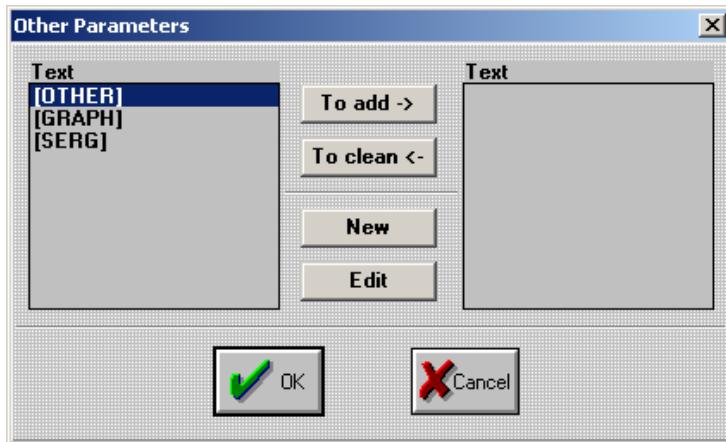


Рис. 77. Форма для задания специфичных параметров

Данное окно имеет следующую структуру. В левом окне находятся все существующие пользовательские блоки, в правом - блоки выводимые в файл данных. Кнопка “To add ->” добавляет блок в правое окно, кнопка “To clean <-” удаляет блок из правого окна. С помощью кнопки “New” создается новый пользовательский блок данных. Кнопка “Edit” позволяет редактировать пользовательский блок данных посредством специального диалогового окна (рис. 78).

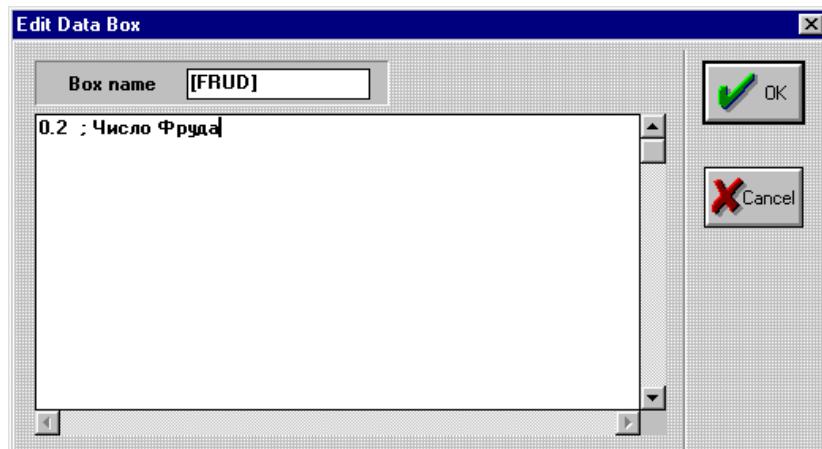
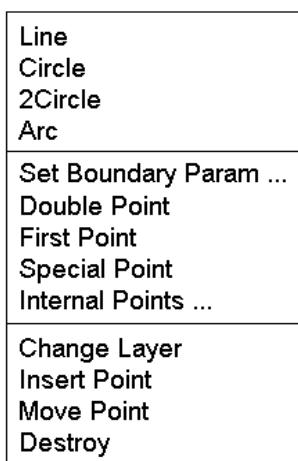


Рис. 78. Окно для редактирования блока данных пользователя

Draw



Данное меню предназначено для отображения в окне различных графических объектов, а также для работы с этими объектами.

“Line” (или пиктограмма). Этот пункт предназначен для отрисовки ломаной линии. Сначала нужно установить курсор в начальную позицию и щелкнуть левой кнопкой

мыши (или нажать клавишу <пробел>). Далее, при перемещении курсора на экране будет отображаться прямая линия. Следующий щелчок зафиксирует первый кусочек ломаной линии и начальную позицию следующей ее части. Сброс отрисовки осуществляется двойным щелчком левой кнопки мыши или клавишей <Enter>. Щелчок правой кнопки мыши или клавиши <Ctrl> переводит курсор в ближайшую к нему начальную точку ломаной линии. Координаты точек, фиксирующих прямую, можно вводить с помощью клавиатуры в поля окна состояния, где отображаются текущие координаты курсора. Для подтверждения введенных координат нужно нажать кнопку <OK>.

"**Circle**" (или пиктограмма ). Этот пункт меню предназначен для изображения на экране окружности, определенной по двум точкам. Первая точка является центром окружности, а вторая принадлежит самой окружности.

"**2Circle**" (или пиктограмма ). Данный пункт меню предназначен для изображения на экране окружности по двум точкам, лежащим на ней.

"**Arc**" (или пиктограмма ). Этот пункт меню предназначен для изображения на экране полукруга по двум точкам принадлежащим его концам.

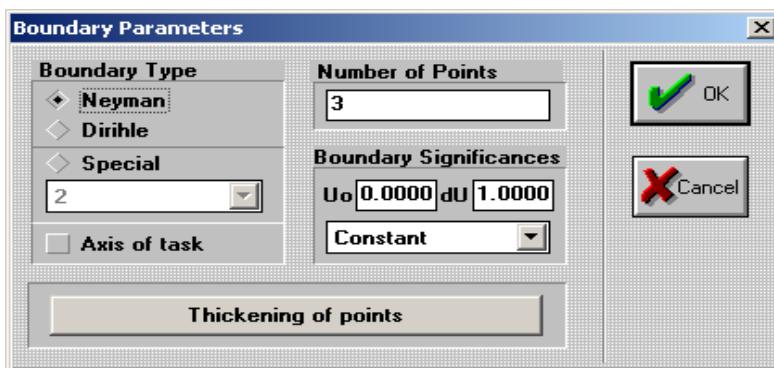


Рис. 79. Форма для задания типа границы и значения граничных условий

"**Set Boundary Param ...**" (или пиктограмма ). Этот пункт меню предназначен для разбиения границы расчетной области на граничные элементы, а также для определения типа границы и значений граничных условий. При его выборе курсор становится квадратным. Левой кнопкой мыши выбирается нужный элемент (отрезок, дуга, окружность) и в диалоговом окне (рис. 79) определяются необходимые параметры:

“Boundary Type” - тип граничных условий (Дирихле, Неймана или специальный). Значение специального вида выбирается из списка ниже.

“Axis of task” - обозначение оси симметрии в осесимметричных задачах;

“Number of points” - количество точек, на которое разбивается графический элемент;

“Boundary Significances” – распределение значений граничного условия, начиная от значения **U₀** и шагом **dU**.

Граничные значения можно задать в отдельно взятой точке. Для этого нужно щелкнуть левой клавишей мыши в выбранной точке области и в диалоговом окне (рис. 80) задать необходимые параметры.

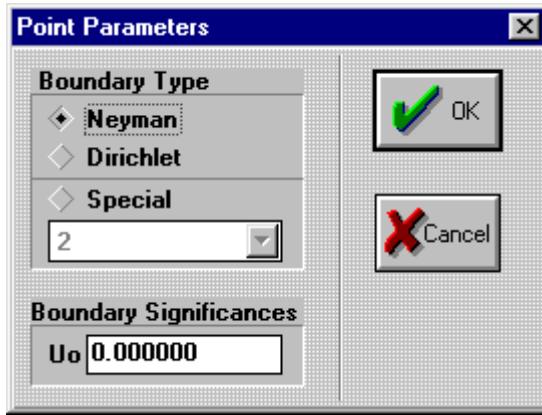


Рис. 80. Дополнительное окно для задания параметров в отдельной точке
Данное окно по своему назначению аналогично предыдущему.

“Double Point” (или пиктограмма). Этот пункт меню предназначен для задания, так называемых, «двойных точек». В этих точках могут быть заданы два или более типа граничных условий. При его выборе курсор становится квадратным. Необходимо подвести его к нужной точке и нажать левую кнопку мыши. При этом точка станет «двойной» и выделится зеленым цветом. При повторном выборе той же точки она опять станет одинарной.

“First Point” (или пиктограмма). Данный пункт меню служит для определения первой точки каждой подобласти и направления ее обхода. При выборе этого пункта меню появляется квадратный курсор, с помощью которого необходимо сначала выбрать точку, а затем элемент, прилегающий к этой точке. В зависимости от расположения этого элемента определится направление обхода границы области.

“Special Point“ (или пиктограмма ). Этот пункт меню предназначен для определения номера какой-либо отдельной точки (или набора точек между двумя одинаковыми специальными точками) и присвоения ей уникального имени. В дальнейшем можно следить за изменениями параметров в этой точке. При его выборе курсор станет квадратным. Левой кнопкой мыши выбирается нужная точка и в появившемся диалоговом окне (рис. 81) задается имя специальной точки.

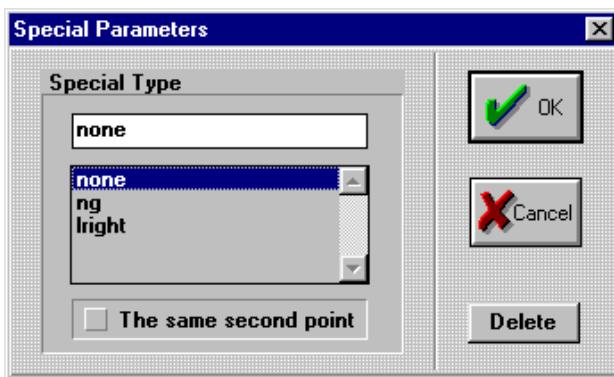


Рис. 81. Задание специальной точки

“Internal Points...“ (или пиктограмма ). При выборе этого пункта меню включается режим ввода координат внутренних точек. Для этого необходимо поставить курсор мыши в нужное место и щелкнуть левой кнопкой.

“Change Layer“ (или пиктограмма ). Данный пункт меню производит переключение между различными подобластями (слоями) рисунка. Активная подобласть рисуется сплошными линиями, а не активные подобласти отображаются пунктирными линиями.

“Insert Point“ (или пиктограмма ). Этот пункт меню выбирается если вам необходимо разбить прямую на два элемента. При его выборе курсор станет квадратным, после чего нужно указать на прямой место, где будет вставлена точка. Если щелкнуть мышкой на пересечении двух прямых, то они будут разбиты на четыре элемента. С полученными прямыми можно производить все стандартные операции (разбиение, удаление, перемещение и т.д.).

“Move Point“ (или пиктограмма ). Данный пункт меню предназначен для изменения конфигурации области. После выбора этого пункта нужно щелкнуть мышкой на точке, координаты которой не-

обходимо изменить. При перемещении мыши происходит изменение конфигурации прилегающих к выбранной точке элементов. Фиксация новых координат точки производится повторным нажатием левой кнопки мыши. После этого можно осуществлять перемещение другой точки без повторного выбора данного. Сброс режима перемещения производится двойным щелчком мыши.



“Destroy” (или пиктограмма) . Этот пункт меню служит для удаления графических элементов. При его выборе курсор становится квадратным. Для удаления нужно щелчком мыши указать нужный графический элемент. Сброс режима удаления производится двойным щелчком.

Display

Step...
Size...
Set Origin
Zoom
Move

Данное меню предназначено для изменения параметров графического окна, его размеров, положения рисунка в окне.

“Step...”. Этот пункт меню предназначен для изменения шага приращения координат. Значения приращений задаются в соответствующих полях диалоговой формы (рис. 82).

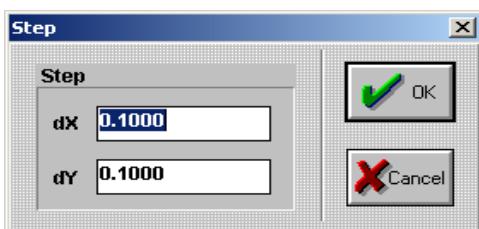


Рис. 82. Окно для задания приращений координат

“Size...”. Данный пункт меню предназначен для изменения логической ширины и высоты окна. Новые размеры окна задаются в диалоговой форме, показанной на рис. 83.

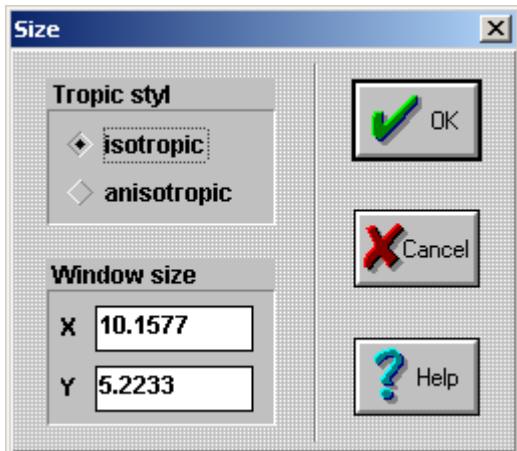


Рис. 83. Задание размеров окна приложения

“Tropic Style” - режимы изменения масштабного коэффициента. В режиме **“Isotropic”** физические размеры логической единицы измерения по осям X и Y совпадают. При этом размер графического окна по оси X вводится пользователем, а по оси Y вычисляется автоматически. При данном режиме отображения достигается наибольшее правдоподобие, так как физические размеры объектов с одинаковыми логическими размерами совпадают. Например, прямоугольник с одинаковыми логическими размерами сторон будет отображаться как квадрат. В режиме **“Anisotropic”** физические размеры логической единицы измерения по осям X и Y не обязательно совпадают. При этом размеры графического окна по осям X и Y вводятся пользователем. Данный режим удобно использовать, если логические размеры окна по осям сильно отличаются друг от друга (область сильно вытянута по оси X или Y).

“Set Origin” (или пиктограмма) . Этот пункт меню используется для переноса начала системы координат (точки с координатами (0,0)) в любую точку области. При этом координаты всех точек области соответствующим образом пересчитываются. После выбора данного пункта меню нужно щелкнуть левой кнопкой мыши в точке, где будет располагаться новый центр осей координат. Эта точка обозначается в окне пересечением нарисованных осей координат. По умолчанию, начало системы координат находится в левом верхнем углу графического окна.

“Zoom” (или пиктограмма) . Данный пункт меню предназначен для увеличения отдельных участков рисунка области. После выбора этого пункта необходимо поместить курсор мыши в левый верхний угол увеличиваемой области и щелкнуть левой клавишей мы-

ши. Затем переместить курсор в нижний правый угол выбранной области и снова щелкнуть мышкой. При перемещении мыши увеличивающийся участок области будет выделяться растягивающимся контуром прямоугольника.



“Move” (или пиктограмма) . Пункт меню служит для перемещения отрисованной области относительно окна. После его выбора курсор будет отображаться в виде черной стрелки. Для перемещения рисунка нужно нажать левую кнопку мыши и, не отпуская ее, переместить рисунок в нужное место графического окна. Рисунок при этом будет перемещаться вместе с курсором. Данная операция введена для удобства просмотра рисунка при выходе его за границы графического окна.

Show

Данное меню предназначено для изменения режимов отображения в окне различных графических объектов. Пункты меню представляют собой переключатели. Во включенном состоянии они помечаются слева галочкой.

<input checked="" type="checkbox"/> Points
<input checked="" type="checkbox"/> Special Points
<input checked="" type="checkbox"/> Internal Points
Numeric Points
Boundary Type Points

“Points”. Переключатель отвечает за отображение на экране всех точек области, на которые разбивается ее граница. По умолчанию находится во включенном состоянии.

“Special Points”. Данный переключатель отвечает за отображение специальных точек. Специальная точка обозначается стрелкой, над которой пишется характерное имя точки. По умолчанию находится во включенном состоянии.

“Internal Points”. Этот переключатель отвечает за отображение на экране внутренних точек. По умолчанию находится во включенном состоянии.

“Numeric Points”. Переключатель отвечает за отображение номеров граничных узлов. Нумерация возможна только тогда, когда задана первая точка (меню <Draw>|First Point>). По умолчанию находится в выключенном состоянии.

“Boundary Type Points”. Данный переключатель отвечает за отображение рядом с каждой граничной точкой типа границы, которой принадлежит данная точка. По умолчанию находится в выключенном состоянии.

Window

Меню “**Window**” является стандартным для многооконных интерфейсов. Здесь находятся пункты меню, управляющие графическими окнами, располагающие в виде каскада или мозаики, а также список всех открытых графических окон. С помощью пункта “**Close all**” можно закрыть все открытые на данный момент графические окна.

РЕКОМЕНДУЕМЫЙ ПОРЯДОК РАБОТЫ

1. Запустить программу “**AKORD**” (файл “*akord.exe*”).
2. Выбрать в меню “**File**” подпункт “**New**”. После чего в появившемся окне задать необходимый тип задачи и имя создаваемого файла. В результате на экране появится чистое графическое окно. При необходимости редактирования уже существующего файла используется подпункт “**Open**”.
3. С помощью пункта меню “**Display/Size**” задать максимальные размеры окна в физических координатах. Это необходимо чтобы рисунок полностью помещался в окне и соответствовал его размерам.
4. При помощи пункта меню “**Display/Step**” задать удобные шаги приращения координат по осям X и Y.
5. Переместить точку начала координат (пункт меню “**Display/Set Origin**”) в нужную точку.
6. Используя пункты “**Draw/Line**”, “**Draw/Arc**”, “**Draw/Circle**” нарисовать границы области. Необходимо помнить, что каждая подобласть рисуется на отдельном слое. Переключение между слоями осуществляется при помощи пункта меню “**Draw/Change Layer**”.
7. В каждой подобласти задать первую точку (пункт меню “**Draw/First Point**”) и направление обхода границы.
8. С помощью пункта меню “**Draw/Set Boundary Param...**” разбить каждый элемент области на необходимое число граничных элементов, задать тип границы и граничные значения на всем элементе или в отдельно взятой точке.
9. При необходимости, определить двойные, специальные или внутренние точки при помощи пунктов меню “**Draw/Double Points**”, “**Draw/Special Points**” или “**Draw/Internal Points**”.

10. Ввести все остальные необходимые параметры (временные, физические, графические и т.д.). Пункт меню "Edit".
11. Сохранить полученное начальное состояние задачи в файле (пункт меню "Edit/Save").

Приложение 2. Работа с постпроцессором

Рассмотрим последовательность действий, которые необходимо предпринять в постпроцессоре для анализа расчетов некоторой задачи.

Открытие файла результатов

Для анализа полученных данных в постпроцессоре необходимо сначала загрузить файл результатов. Для этого необходимо запустить постпроцессор и выбрать команду меню **File/Open Data Files**. Поскольку для дальнейшей работы постпроцессора с файлом данных необходимо его преобразование в двоичный формат, то сразу же запустится конвертор данных, который предложит пользователю выбрать имя нового файла данных. По умолчанию оно совпадает со старым, а расширение файла изменяется на "bin". Для начала конвертирования необходимо нажать на кнопку «Сохранить». Появится диалог (рис. 84), содержащий индикатор, в котором будет отображаться процесс конвертирования. Пользователь имеет возможность прервать процесс, нажав на кнопку «Stop». После конвертирования файл автоматически загружен и для работы с ним будет задействован контроллер итераций.

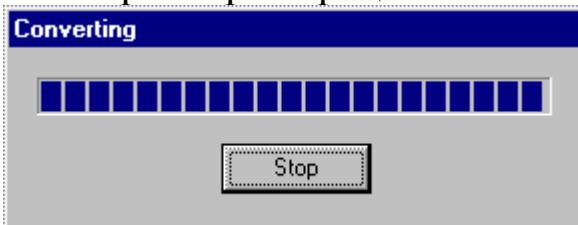


Рис. 84. Индикатор конвертирования файла

Затем постпроцессор выведет диалог (рис. 85) выбора отображателя данных. Выберем "*Отображатель свободной границы*".

Отображатель свободной границы

Для задания свободной границы следует воспользоваться меню **Drawer/Add Free Bound Data Dependency**. В появившемся диалоговом окне (рис. 86) выбрать "X" и "Y", нажать кнопку "Ok". В ок-

не отображателя данных появится графическое представление свободной границы (рис. 87).

При использовании отображателя свободной границы пользователь может задавать различные режимы просмотра данных через меню группы "*View*" или контекстное меню в окне отображателя данных, которое вызывается при нажатии правой кнопки «мыши».

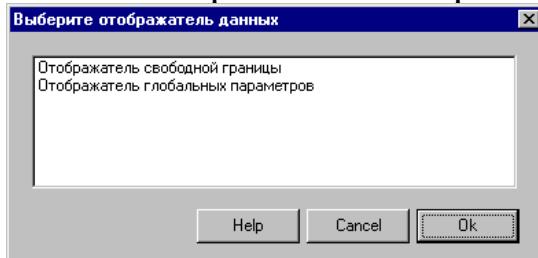


Рис. 85. Диалог выбора отображателя данных



Рис. 86. Создание зависимости свободная граница

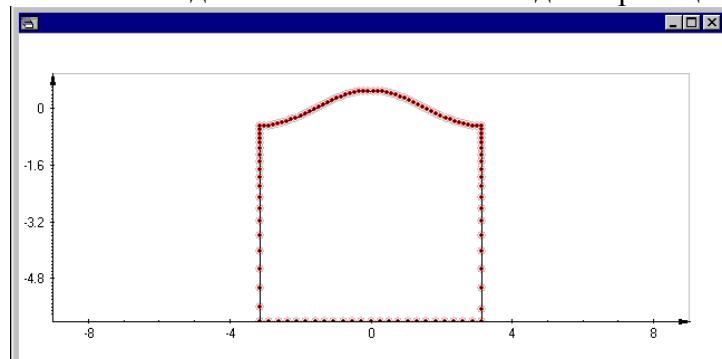


Рис. 87. Окно отображателя данных

Нормальный Режим (режим просмотра) вызывается пунктом меню *View/Set Normal Mode*. Данный режим включен по умолчанию. В нем пользователь может выделять графические объекты и задавать их свойства. Для выделения объекта нужно позиционировать на него указатель мыши и нажать левую кнопку. Изменить свойства активного объекта можно через пункт меню *View/Graphics Object Properties*. В появившемся диалоге можно установить свойства выбранного объекта. Для свободной границы это будет выглядеть следующим образом (рис. 88):

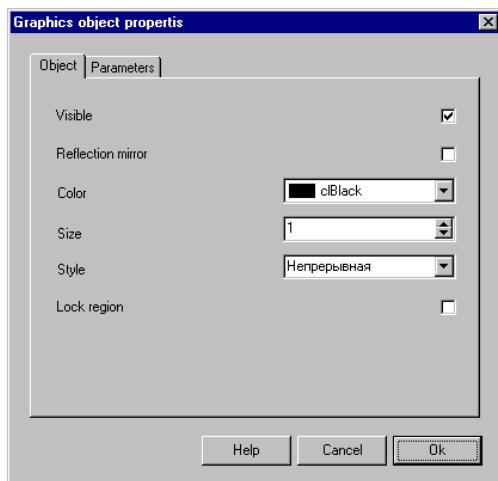


Рис. 88. Свойства графического объекта

Здесь имеются две вкладки: «*Object*» и «*Parameters*». При помощи первой из них можно задать цвет, стиль, толщину линии свободной границы, зеркальное отображение объекта относительно оси ординат. Во второй - задать, какие параметры нужно выводить, а какие нет, поставив галочку напротив нужного параметра.

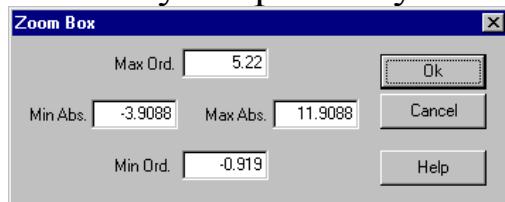


Рис. 89. Диалог "Увеличить область"

Режим зуммирования включается через пункт меню *View/Set Zoom Mode*. Здесь пользователь имеет возможность масштабировать область просмотра. Щелчок левой кнопкой мыши на окне увеличивает масштаб, щелчок левой кнопкой с нажатой клавишей "Ctrl" - уменьшает. При выделении прямоугольной области мышью эта область будет увеличена на весь размер окна. Можно также выбрать пункт *View/Zoom Box* и в диалоге (рис. 89) указать размеры требуемой области.

Режим выделения фреймов включается посредством пункта меню *View/Set Frame Selection Mode*. Этим режимом имеет смысл пользоваться только при анализе динамических данных. В главном окне отображателя свободной границы пользователь может мышкой выделить несколько прямоугольных областей – фреймов. При этом появится дополнительное окно, в котором в виде блокнота будет содержаться список всех фреймов (рис.90).

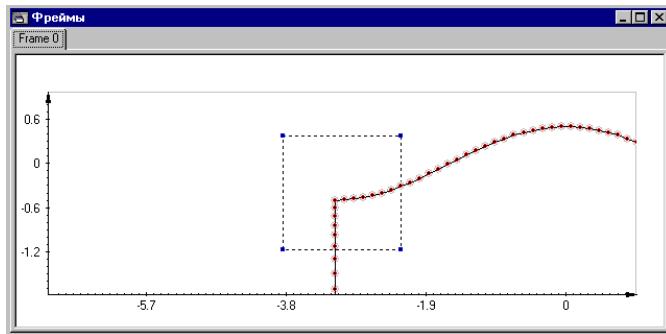


Рис. 90. Окно фреймов

Если просто выделить область мышкой получится фрейм первого типа. Для того, чтобы создать следящий фрейм, нужно выделить область (удерживая нажатой клавишу «*Ctrl*») так, чтобы в ее центре находилась точка за перемещением которой нужно проследить. Если в выделенной области находится несколько точек, то будет выбрана та, которая находится ближе к центру области. После выделения фрейма появляется диалог, в котором можно задать или подкорректировать его параметры. Для каждого фрейма можно указать его геометрические размеры. Для простого фрейма также нужно задать скорость, с которой он будет "перемещаться" при изменении времени (рис. 91). По умолчанию скорость равна 0.

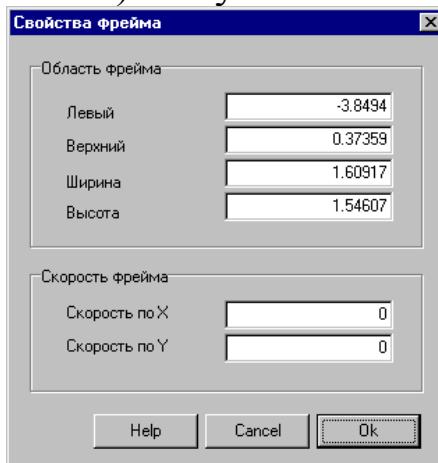


Рис. 91. Диалог "Свойства фрейма"

Поскольку файл содержит динамические данные, можно проследить за изменением свободной границы с течением времени. Для этого пользователю необходимо воспользоваться пунктами меню группы ***Time*** или кнопками в верхней горизонтальной панели инструментов (рис. 92). Выбирая пункты меню ***Time/Resume Time*** и ***Time/Suspend Time*** пользователь может включать и выключать режим «прокрутки» фрейма. Пункты ***Time/Set Prev Time Step*** и ***Time/Set Next Time Step*** позволяют перейти на предыдущий или следующий временной шаг, соответственно. Пункт ***Time/Set Time Params*** выводит диалог настройки параметров времени (рис. 93).

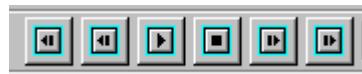


Рис. 92. Кнопки управления временем

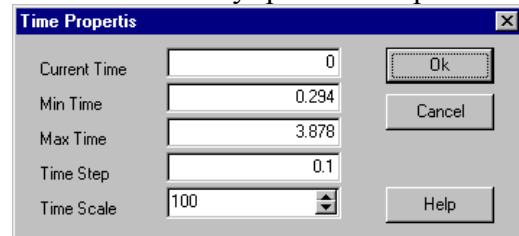


Рис. 93. Диалог "Параметры времени"

Нажатие на кнопку также запускает «прокрутку» фрейма. При этом можно видеть соответствующее изменение свободной границы (рис. 94).

Видно, как на свободной границе отображается поле векторного параметра с именем "Скорость". Пользователь имеет возможность изменить свойства отображения данного параметра (масштаб единичного вектора, цвет). Для этого нужно воспользоваться пунктом меню ***View/Params Properties***. Затем в открывшемся диалоге выбрать параметр и нажать кнопку "*Properties*".

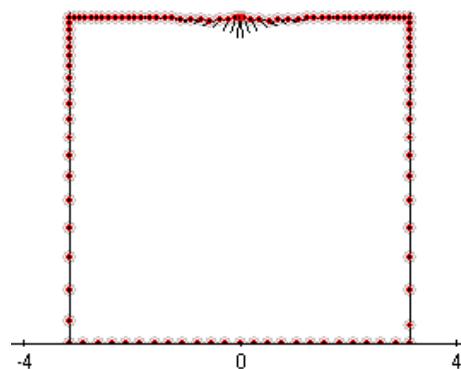


Рис. 94. Изменение свободной границы

Есть альтернативный путь. Необходимо вывести легенду параметров (рис. 95) посредством меню ***View>Show Parameters Legend***.

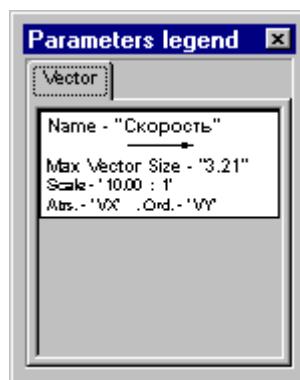


Рис. 95. Легенда параметров

Во вкладке "*Vector*" указываются все векторные параметры текущего отображателя данных. Для каждого параметра указывается его имя - "*Name*", масштаб максимального вектора - "*Scale*", его размер - "*Max Vector Size*", горизонтальная и вертикальная составляющие - "*Abs*" и "*Ord*". В центре отображается максимальный вектор в указанном масштабе и цвете. В данном случае будет всего один векторный параметр "Скорость". Этот параметр был задан не пользователем, а постпроцессором. При открытии файла данных постпроцессор проверяет наличие так называемых "стандартных" имен параметров. Если таковые есть, то для них автоматически задаются параметры по умолчанию. Например, если файл содержит параметры с именами "VX" и "VY", то автоматически будет задан векторный параметр "Скорость" с указанными параметрами в качестве компонент вектора. Число параметров не ограничено, и пользователь может легко задать новые, щелкнув правой кнопкой мыши на вкладке "*Vector*" и воспользовавшись пунктом контекстного меню *Add Param*. Появится диалог (рис. 96)

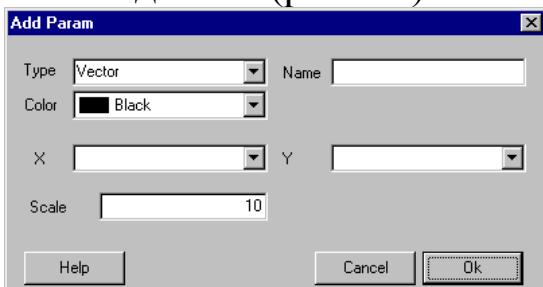


Рис. 96. Создание нового параметра

Здесь можно задать имя нового параметра "*Name*", горизонтальную и вертикальную составляющие "X" и "Y", соответственно, а также цвет "Color" и масштаб единичного вектора "Scale". Параметры, из которых выбираются составляющие вектора, берутся из файла данных итераций. В качестве примера зададим имя нового параметра как "Test". В качестве составляющих возьмем "FI" и "DFI" и укажем цвет "синий". После нажатия кнопки "Ok" в легенде параметров появится новый векторный параметр "Test" (рис. 97).

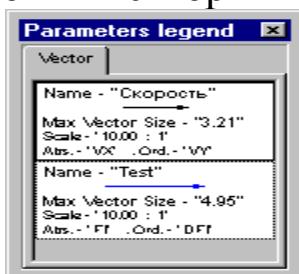


Рис. 97. Легенда после добавления нового параметра

Чтобы увидеть значение нового параметра на свободной границе, сначала удалим старую зависимость данных. Для этого нужно перейти в режим просмотра, выделить зависимость данных, нажав на нее левой кнопкой мыши, и воспользоваться пунктом контекстного меню **Delete Object** (или нажав клавишу "Del"). Создадим новую зависимость, воспользовавшись пунктом меню **Drawer/Add Bound**. В качестве второго векторного параметра укажем "Test". В итоге получим зависимость типа свободная граница, в которой отображаются уже два векторных параметра "Скорость" и "Test". Чтобы увидеть параметр "Test", необходимо изменить его свойства (щелкнув два раза мышью на нем в легенде параметров), как это показано на рис. 98:

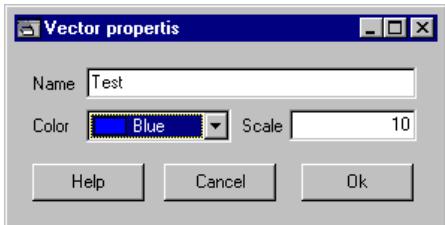


Рис. 98. Задание свойств векторного параметра

Изменим масштаб максимального вектора с 10 на 50. Изображение свободной границы изменится соответствующим образом (рис. 99)

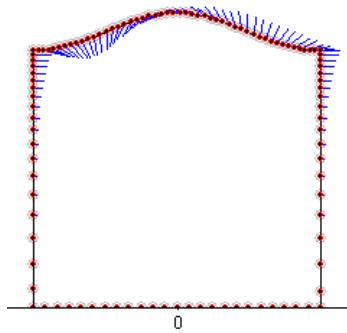


Рис. 99. Вид свободной границы после изменения масштабов векторного параметра

Отображатель глобальных параметров

Если в файле данных есть глобальные параметры, то можно просмотреть их изменение в зависимости от времени. Для этого надо воспользоваться отображателем глобальных параметров. Чтобы создать новый отображатель данных, нужно воспользоваться пунктом меню **Drawer/Create Drawer**. После создания окна отображателя пользователю будет предложено задать зависимость глобального параметра от времени (рис. 100).



Рис. 100. Задание зависимости глобального параметра от времени

По нажатию кнопки "Ok" в окне отображателя появится указанная зависимость (рис. 101).

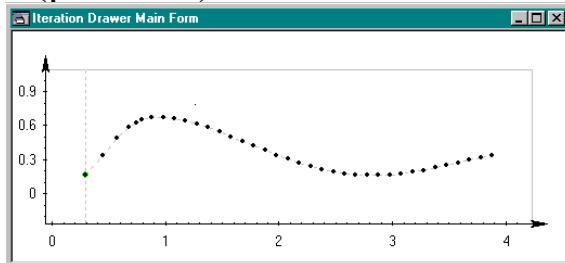


Рис. 101. Отображение зависимости глобального параметра "Кинетическая энергия" от времени

Зависимость параметра "Кинетическая энергия" показана в виде траектории, по которой движется яркая зеленая точка - текущее значение параметра. Также на график наложена пунктирная вертикальная линия, которая указывает текущее значение параметра.

В одном окне можно просматривать зависимости сразу нескольких параметров. Воспользовавшись пунктом меню **Drawer/Add Global Parameter Data Dependency** можно добавить зависимость второго глобального параметра - "Потенциальная энергия". Дополнительно можно указать зависимость суммы двух параметров от времени. Для этого надо воспользоваться пунктом меню **Drawer/Add Summa Global Parameters Data Dependency** и в появившемся диалоге (рис. 102) указать имена нужных параметров.

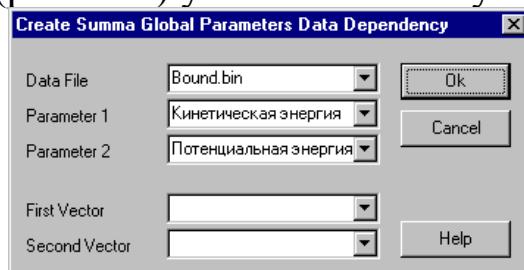


Рис. 102. Создание зависимости суммы глобальных параметров от времени

После этого окно отображателя примет следующий вид (рис. 103).

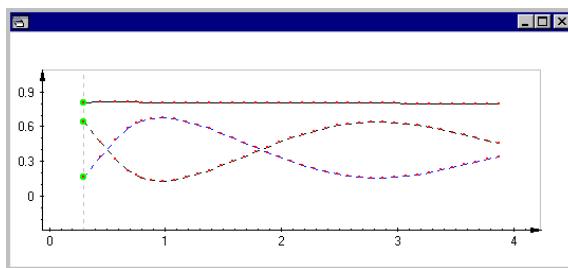


Рис. 103. Окно отображателя глобальных параметров

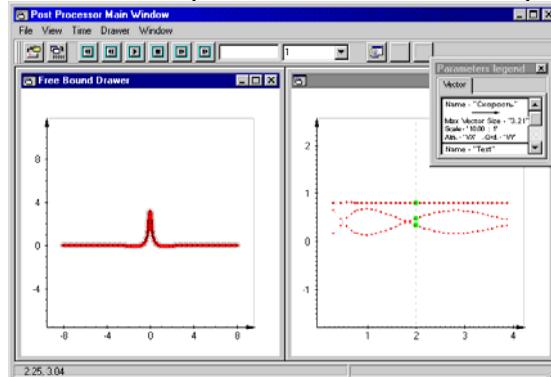


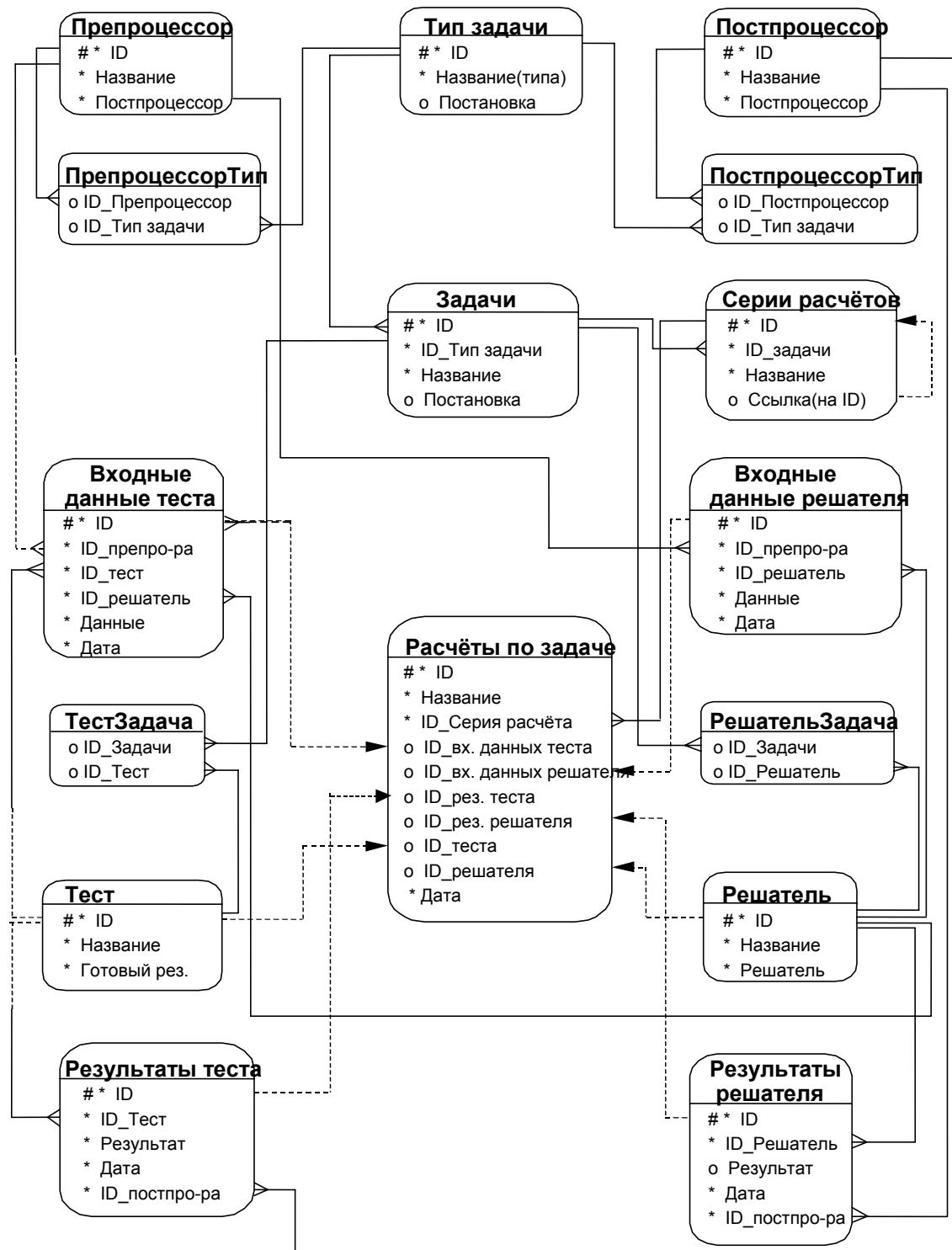
Рис. 104. Отображение нескольких зависимостей

Поскольку оба открытых отображателя связаны с одним и тем же контроллером данных, то задавая изменение времени можно наблюдать за изменением, как свободной границы, так и глобальных параметров одновременно (рис. 104).

Сохранение графических изображений

Пользователь имеет возможность сохранить графическое изображение в виде файла в любом из предлагаемых графических форматов: «bmp», «psx», «wmf». Для этого необходимо воспользоваться пунктом меню **Edit/Save as graphics file** или аналогичным пунктом контекстного меню.

Приложение 3. ER-диаграмма модели данных, использованной в ИС



- первичный ключ, * - обязательное поле, о - необязательное поле

Приложение 4. Структура таблиц базы данных

Тип задачи (TypeZadacha)

Имя столбца	ID	Name_Type	Post_Type
Тип ключа	PK		
Nulls/Unique	NN,U	NN	NN
Таблица FK			
Столбец FK			
Тип данных	Number	Char	Long Raw
Длина	7	50	

Задачи (Zadacha)

Имя столбца	ID	ID_Type	Name_Zadacha	Post
Тип ключа	PK	FK		
Nulls/Unique	NN,U	NN	NN	NN
Таблица FK		TypeZadacha		
Столбец FK		ID		
Тип данных	Number	Number	Char	Long Raw
Длина	7	7	50	

Серии расчетов (CeriaRasch)

Имя столбцов	ID	ID_Zadachi	Name_Ceria	Go_On_Id
Тип ключа	PK	FK		FK
Nulls/Unique	NN,U	NN	NN	
Таблица FK		Zadacha		
Столбец FK		ID		
Тип данных	Number	Number	Char	Number
Длина	7	7	50	7

Расчеты по задаче (RaschetZadachi)

Имя столбцов	ID	Name_Rasch	ID_Ceria	ID_Idt
Тип ключа	PK		FK	FK
Nulls/Unique	NN,U	NN	NN	
Таблица FK			CeriaRasch	IDT
Столбец FK			ID	ID
Тип данных	Number	Char	Number	Number
Длина	7	50	7	7

продолжение таблицы (RaschetZadachi)

Имя столбцов	ID_Idr	ID_ResTest	ID_ResReshatel	ID_Test	ID_Reshatel	Data
Тип ключа	FK	FK	FK	FK	FK	
Nulls/Unique						
Таблица FK	IDR	ResTest	ResReshatel	Test	Reshatel	
Столбец FK	ID	ID	ID	ID	ID	
Тип данных	Number	Number	Number	Number	Number	Date
Длина	7	7	7	7	7	

Входные данные теста (InputDataTest "IDT")

Имя столбцов	ID	ID_Prepr	ID_Test	ID_Reshatel	Input_File_Test	Data
Тип ключа	PK	FK	FK	FK		
Nulls/Unique	NN,U	NN	NN	NN		
Таблица FK		Preproc- essor	Test	Reshatel		
Столбец FK		ID	ID	ID		
Тип данных	Num- ber	Number	Number	Number	Long Raw	Date
Длина	7	7	7	7		

Входные данные решателя (InputDataReshatel "IDR")

Имя столб- цов	ID	ID_Prepr	ID_Reshatel	Input_File_Reshatel	Data
Тип ключа	PK	FK	FK		
Nulls/Unique	NN,U	NN	NN		
Таблица FK		Preprocessor	Reshatel		
Столбец FK		ID	ID		
Тип данных	Number	Number	Number	Long Raw	Date
Длина	7	7	7		

Результаты теста (ResultTest)

Имя столб- цов	ID	ID_Test	ID_Post	Result_File_Test	Data
Тип ключа	PK	FK	FK		
Nulls/Unique	NN,U	NN			
Таблица FK		Test	Postprocessor		
Столбец FK		ID	ID		
Тип данных	Number	Number	Number	Long Raw	Date
Длина	7	7	7		

Результаты решателя (ResultReshatel)

Имя столб- цов	ID	ID_Reshatel	ID_Post	Result_File_Resha	Data
Тип ключа	PK	FK	FK		
Nulls/Unique	NN,U	NN			
Таблица FK		Reshatel	Postprocessor		
Столбец FK		ID	ID		
Тип данных	Number	Number	Number	Long Raw	Date
Длина	7	7	7		

Препроцессор (Preprocessor)

Имя столбцов	ID	Name_Preprocessor	File_Prep	Data
Тип ключа	PK			
Nulls/Unique	NN,U	NN		
Таблица FK				
Столбец FK				
Тип данных	Number	Char	Long Raw	Date
Длина	7	50		

Постпроцессор (Postprocessor)

Имя столбцов	ID	Name_Postprocessor	File_Post	Data
Тип ключа	PK			
Nulls/Unique	NN,U	NN		
Таблица FK				
Столбец FK				
Тип данных	Number	Char	Long Raw	Date
Длина	7	50		

Тест (Test)

Имя столбцов	ID	Name_Test	Input_Data	Data
Тип ключа	PK			
Nulls/Unique	NN,U	NN	NN	
Таблица FK				
Столбец FK				
Тип данных	Number	Char	Number	Date
Длина	7	50	1	

Решатель (Reshatel)

Имя столбцов	ID	Name_Reshatel	File_Reshatel	Data
Тип ключа	PK			
Nulls/Unique	NN,U	NN		
Таблица FK				
Столбец FK				
Тип данных	Number	Char	Long Raw	Date
Длина	7	50		

Препроцессор Тип (PreprType)

Имя столбцов	ID_Prepr	ID_Type
Тип ключа	FK	FK
Nulls/Unique		
Таблица FK	Preprocessor	TypeZadacha
Столбец FK	ID	ID
Тип данных	Number	Number
Длина	7	7

Постпроцессор Тип (PostType)

Имя столбцов	ID_Post	ID_Type
Тип ключа	FK	FK
Nulls/Unique		
Таблица FK	Postprocessor	TypeZadacha
Столбец FK	ID	ID
Тип данных	Number	Number
Длина	7	7

ТестЗадача (TestZadacha)

Имя столбцов	ID_Test	ID_Zadacha
Тип ключа	FK	FK
Nulls/Unique		
Таблица FK	Test	Zadacha
Столбец FK	ID	ID
Тип данных	Number	Number
Длина	7	7

РешательЗадача (ReshatelZadacha)

Имя столбцов	ID_Reshatel	ID_Zadacha
Тип ключа	FK	FK
Nulls/Unique		
Таблица FK	Reshatel	Zadacha
Столбец FK	ID	ID
Тип данных	Number	Number
Длина	7	7

Приложение 5. Структура файлов обмена данными между приложениями ИС

Файл данных, получаемый при использовании препроцессора для плоских и осесимметричных задач.

```
[MAIN]
тест по новому файлу структуры данных
2 ; Размерность задачи
27 ; Общее количество узлов
1 ; Число областей
0 ; Число внутренних точек

[TIME] PARAMETERS
0.002000 ; Максимальный шаг по времени
0.000002 ; Минимальный шаг по времени
0.000000 ; Начальное время счета
1.000000 ; Максимально возможное время счета

[PHISICAL] PARAMETERS
1.000000 ; Допустимая погрешность
1.000000 ; Давление на поверхности жидкости
1.000000 ; Плотность жидкости
0.000000 ; Коэффициент поверхностного натяжения
1.400000 ; Показатель адиабаты

[DOMAIN] PARAMETERS
1 ; Ограничность области решения
1 ; Тип области
1 ; Положение нормального вектора
-1 ; Направление нормального вектора
0 ; Наличие силы тяжести

[OUTPUT] PARAMETERS
100 ; Шаг для вывода результатов
0 ; Шаг для записи состояния задачи во временный файл
1 ; Шаг для расчета внутренней задачи
0 ; Шаг для отрисовки проекции области
1 ; Шаг для отрисовки расчетной сетки
0 ; Шаг для вывода графической информации
0 ; Шаг для сглаживания границ области
0 ; Шаг для вывода граф. информации в файл
.T. ; Очищать окно для вывода проекции
.T. ; Очищать окно для вывода расчетной сетки
.F. ; Не прекращать расчет
.T. ; Очищать все временные файлы

[NODES] POINTS
1 +0.0000000000e+00 +1.0000000000e+00 0 +0.000000
2 +1.3052619222e-01 +9.9144486137e-01 0 +0.000000
3 +2.5881904510e-01 +9.6592582629e-01 0 +0.000000
```

```

4 +3.8268343237e-01 +9.2387953251e-01 0 +0.000000
5 +5.0000000000e-01 +8.6602540378e-01 0 +0.000000
6 +6.0876142901e-01 +7.9335334029e-01 0 +0.000000
7 +7.0710678119e-01 +7.0710678119e-01 0 +0.000000
8 +7.9335334029e-01 +6.0876142901e-01 0 +0.000000
9 +8.6602540378e-01 +5.0000000000e-01 0 +0.000000
10 +9.2387953251e-01 +3.8268343237e-01 0 +0.000000
11 +9.6592582629e-01 +2.5881904510e-01 0 +0.000000
12 +9.9144486137e-01 +1.3052619222e-01 0 +0.000000
13 +1.0000000000e+00 +6.1257422745e-17 0 +0.000000
14 +9.9144486137e-01 -1.3052619222e-01 0 +0.000000
15 +9.6592582629e-01 -2.5881904510e-01 0 +0.000000
16 +9.2387953251e-01 -3.8268343237e-01 0 +0.000000
17 +8.6602540378e-01 -5.0000000000e-01 0 +0.000000
18 +7.9335334029e-01 -6.0876142901e-01 0 +0.000000
19 +7.0710678119e-01 -7.0710678119e-01 0 +0.000000
20 +6.0876142901e-01 -7.9335334029e-01 0 +0.000000
21 +5.0000000000e-01 -8.6602540378e-01 0 +0.000000
22 +3.8268343237e-01 -9.2387953251e-01 0 +0.000000
23 +2.5881904510e-01 -9.6592582629e-01 0 +0.000000
24 +1.3052619222e-01 -9.9144486137e-01 0 +0.000000
25 +0.0000000000e+00 -1.0000000000e+00 0 +0.000000
26 +0.0000000000e+00 -1.0000000000e+00 1 +0.000000
27 +0.0000000000e+00 +1.0000000000e+00 1 +0.000000

```

[INTERNAL] POINTS

[REGION] PARAMETERS (Num Region , Num FIRST , Num Points)

```

1 1 25
0 26 2

```

[SPECIAL] POINTS

[OTHER]

```

0.0 ; Рассстояние до стенки ( HWall )
0.0 ; Рассстояние до свободной поверхности ( HSurf )
1.0 ; Начальный радиус пузыря ( R0(3) )
1.0 ; Максимальный радиус пузыря ( Rm(3) )
0.0 ; Начальное давление в пузыре ( P0(3) )
0 ; Модель для давления внутри пузыря ( ITIP = 0 - constant, 1 - change )
2,0,0,0 ; Parameters for Split ( iSPlace, iDraw, iXScale, iYScale )
1 ; Parameters for Draw ( iDPlace )
1 ; Object of smooth ( iRTyp=1-surface,2-boundary,3-all)
2,0 ; Graphic file save step( iGPlace, iGMode )
.F. ; Mode Split Window ( lSValue )
.T. ; AutoGeneration Mesh ( lMesh_Mode )
3 ; Number Mesh Segments ( Iraz )
2 ; Interior Solving Mode ( IISM : 1=FEM(2d), 2=FEM(2,5d), 3=BEM)

```

[GRAPH]

[WINSTRUCT] PARAMETERS

```
10.000000 ;
4.443393 ;
4.400000 ;
1.800000 ;
0.100000 ;
0.100000 ;
[LINES] PARAMETERS
0.000000 -1.000000 0.000000 1.000000 0.000000 1 2014 1 1 0 25 0.000000 1.000000 0
2 0 -1 0.000000
4 0 -1 0.000000
0.000000 1.000000 0.000000 -1.000000 0.000000 0 2011 0 1 1 2 0.000000 1.000000 0
4 0 -1 0.000000
2 0 -1 0.000000
-3.300000 2.500000 2007.700000 1.500000 1.000000 0 2 0 1 0 0 1.000000 2011.000000 0
1 0 -1 0.000000
1 0 -1 0.000000
```

Файл входных данных для постпроцессора, формируемый решателем.

```
[TEXT_DATA_FILE]
```

```
[DATA_FORMAT] TEXT_ITERATION_DATA_COL
```

```
[global_data_def]
```

```
NAME=D1
```

```
NAME=D2
```

```
[end]
```

```
[local_data_def]
```

```
NAME=X
```

```
NAME=Y
```

```
NAME=fi
```

```
NAME=dfi
```

```
NAME=vx
```

```
NAME=vy
```

```
[end]
```

```
[DATA_PARAM]
```

```
10
```

```
111
```

```
[end]
```

```
[region_def]
```

```
0
```

```
[end]
```

```
[Iteration]
```

```
.294170E+00 .166088E+00 .641507E+00
```

Приложения

.800000E+01	-.133000E-03	.103441E-04	-.998250E-03	.485447E-06	-.998250E-03
.791000E+01	-.133000E-03	.103368E-04	-.998035E-03	-.165905E-06	-.998035E-03
.782000E+01	-.133000E-03	.103746E-04	-.100147E-02	.546225E-06	-.100147E-02
.773000E+01	-.134000E-03	.104339E-04	-.100718E-02	.779577E-06	-.100718E-02
.764000E+01	-.135000E-03	.105186E-04	-.101526E-02	.105789E-05	-.101526E-02
...					

**Афанасьев Константин Евгеньевич
Гудов Александр Михайлович**

**ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
В ЧИСЛЕННЫХ РАСЧЕТАХ**

Учебное пособие

Редактор Григорьева О.С.

Лицензия на издательскую деятельность ЛР № 020464 от 9.06.1997 г.

Подписано к печати 3.07.01 Печать офсетная. Формат 60x84 1/16.
Бумага типографская. Печ.л. 12,75, уч. -изд. л. 8,5. Тираж 300 экз.,
Заказ № 213

Кемеровский государственный университет, 650043, Кемерово, ул. Красная, 6
Отпечатано в типографии ООО “Фирма Полиграф”. 650099, ул. Островского, 16