Федеральное государственное бюджетное учреждение науки
Институт прикладной математики им. М.В. Келдыша
Российской академии наук

На правах рукописи

Пошивайло Илья Павлович

ЖЕСТКИЕ И ПЛОХО ОБУСЛОВЛЕННЫЕ НЕЛИНЕЙНЫЕ МОДЕЛИ И МЕТОДЫ ИХ РАСЧЕТА.

Специальность 05.13.18 — Математическое моделирование,
численные методы и комплексы программ

Диссертация
на соискание ученой степени
кандидата физико-математических наук

Научный руководитель:
член-корреспондент РАН,
доктор физико-математических наук, профессор
Калиткин Николай Николаевич

Москва — 2014

Оглавление

Введение. **5**

1 Оптимальные обратные схемы Рунге-Кутты. **13**
1.1 Схемы Рунге-Кутты. 13
1.1.1 Явные схемы. 14
1.1.2 О схемах высших порядков. 14
1.1.3 Обратные схемы. 15
1.1.4 Моно-неявные схемы Рунге-Кутты. 16
1.1.5 Неоптимальные обратные схемы. 16
1.1.6 Устойчивость. 17
1.1.7 Интерполяционность. 18
1.2 Алгоритм. 19
1.2.1 Простые итерации. 19
1.2.2 Метод Ньютона. 20
1.2.3 Усечение. 21
1.2.4 Первая итерация. 21
1.3 Дифференциально-алгебраические системы. 21
1.3.1 Сходимость. 22
1.3.2 Уменьшение трудоемкости алгоритма. 23

2 Оценка погрешности решения в задачах с пограничными слоями. **24**
2.1 Метод Ричардсона для жестких задач. 24
2.1.1 Стандартная процедура. 24
2.1.2 Особенности жестких задач. 25
2.1.3 Разрешение пограничного слоя. 27
2.2 Метод перехода к длине дуги. 27
2.2.1 Уравнения. 28
2.2.2 Сохранение балансов в задачах химической кинетики. 29

3 Модификации метода Ньютона. **32**
3.1 Область сходимости. 32
3.1.1 Классический метод Ньютона. 32
3.1.2 Непрерывный аналог метода Ньютона. 33
3.1.3 Дифференциальный аналог метода Ньютона. 34
3.1.4 Сходимость конечношаговых итерационных методов. 35

3.1.5 Выводы. 36
3.2 Уравнения с кратными корнями. 37
3.2.1 Определение кратности корня. 37
3.2.2 Ускорение сходимости ньютоновских итераций. 38

4 Расчеты моделей прикладных задач. 40
4.1 Бегущая тепловая волна. 40
4.1.1 Аппроксимация коэффициента теплопроводности. 41
4.1.2 Разностные схемы. 41
4.1.3 Сходимость к точному решению. 42
4.1.4 Реализация итерационного процесса. 45
4.1.5 Усеченные ньютоновские итерации. 45
4.1.6 Разрывные начальные данные. 46
4.2 Дифференциально-алгебраические системы. 48
4.2.1 Тестовая задача. 48
4.2.2 Транзисторный усилитель. 49
4.3 Уравнение ван дер Пола. 52
4.3.1 Разностные схемы. 53
4.3.2 Сравнение схем. 54
4.3.3 Влияние жесткости. 55
4.4 Сверхжесткость. 57
4.5 Химическая кинетика. 59
4.5.1 Постановка задачи. 59
4.5.2 Требования к разностным схемам. 61
4.5.3 Результаты расчетов. 62
4.6 Выводы. 63

5 Комплекс программ GEABORK. 65
5.1 Численные методы решения систем ОДУ. 65
5.1.1 Явные методы Рунге-Кутты. 65
5.1.2 Методы Розенброка. 67
5.1.3 Обратные и полностью неявные методы Рунге-Кутты. 68
5.2 Подпрограммы для интегрирования на серии сгущающихся сеток. 74
5.2.1 Интегрирование по аргументу “время”. 74
5.2.2 Интегрирование по аргументу “длина дуги”. 75
5.3 Вспомогательные подпрограммы. 77
5.3.1 Разностное вычисление матрицы Якоби. 77
5.3.2 Метод Ньютона для решения нелинейных систем. 77
5.3.3 Определение погрешности. 79

Заключение 81

Список иллюстраций 83

Список таблиц **84**

Литература **85**

Введение.

Модельные задачи. Настоящая диссертационная работа посвящена численному решению моделей, возникающих в задачах физики плазмы, химической кинетики и ряде других прикладных областей. Такие модели часто описываются системами обыкновенных дифференциальных уравнений, численное решение которых оказывается трудной задачей. Рассмотрим несколько примеров.

Известной трудной задачей для численного интегрирования является уравнение теплопроводности. В простейшем случае уравнение линейно и задано в ограниченной области при постоянном коэффициенте теплопроводности $\varkappa$: $u_t = \varkappa u_{xx} + f(t,x)$, $0 < x < a$, $0 < t \leq T$. В случае однородной задачи точное решение разлагается в ряд по пространственным гармоникам, которые затухают как $e^{-\lambda_m t}$, $\lambda_m \sim m^2$, где m – номер гармоники. То есть затухание высоких гармоник становится неограниченно быстрым. Еще одно интересное свойство линейного уравнения теплопроводности на бесконечной прямой $-\infty < x < +\infty$ заключается в том, что формально для него скорость распространения тепла от точечного источника является бесконечной. Таким образом, если в начальный момент времени тепло имелось только в точке x_0, то в любой отличный от нуля момент времени температура будет отличная от нуля во всех точках бесконечной прямой [1].

При высоких температурах коэффициент теплопроводности $\varkappa$ нередко является функцией от температуры, $\varkappa = \varkappa(u)$. В этом случае уравнение теплопроводности называют квазилинейным:

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left[\varkappa(x,t,u)\frac{\partial u}{\partial x}\right] + f(x,t,u), \quad \varkappa(x,t,u) \geq 0.$$

Такие задачи часто встречаются в физике плазмы: коэффициент теплопроводности имеет степенную зависимость от температуры, $\varkappa(u) \sim u^m$. Например, для электронной теплопроводности плазмы $m = 5/2$.

При определенных граничных и начальном условиях эта задача на полупрямой $0 \leq x < +\infty$ будет иметь автомодельное решение $u(x,t) \equiv u(ct-x)$, предложенное Самарским и Соболем [2]. Это решение имеет вид тепловой волны, бегущей с постоянной скоростью c по нулевому фону. Само решение является непрерывным, однако сложность заключается в том, что оно имеет резко выраженный фронт, в котором производная u_x обращается в бесконечность. Это может приводить к возникновению пилообразного профиля в численном решении, если в расчете не используются специальные надежные схемы.

Другой пример возьмем из моделирования электрической цепи с нелинейным затуханием. Схема построена таким образом, что слабые колебания тока в ней усиливаются, а сильные зату-

хают [3,4]. Задача описывается обыкновенным дифференциальным уравнением второго порядка:

$$z'' + \sigma(z^2 - 1)z' + z = 0.$$

Это уравнение называется осциллятором ван дер Пола. Оно имеет устойчивое периодическое решение. При значении параметра $\sigma = 0$ задача вырождается в гармонический осциллятор. Его фазовый портрет в координатах z, z' представляет собой окружность, а точное решение выражается в виде синусоид $z(t) = sin(t)$, $z'(t) = cos(t)$. С ростом значения σ синусоиды переходят в пределе в прямоугольные "ступеньки", т.е. в решении присутствуют узкие внутренние пограничные слои. Это делает задачу трудной для численного интегрирования.

В качестве третьего примера рассмотрим модель, описывающую горение метана в воздухе. Достаточно полная система для такой задачи содержит десятки компонент и сотни реакций. Упрощенная модель, сохраняющая основные особенности решения, приведена в [4]. При сжигании обычного бытового газа выделяется тепловая энергия, а также образуются в небольшом количестве вредные окислы азота и совсем небольшая примесь канцерогенных циклических углеводородов. Задача описывается системой из 20 обыкновенных дифференциальных уравнений с заданными начальными концентрациями компонент. Сложность для численного решения заключается в том, что в системе присутствуют как медленно, так и очень быстро протекающие реакции (их скорости отличаются на ~ 15 порядков). Это приводит к тому, что концентрации промежуточных радикалов могут быть очень малыми. Кроме того, для расчета установившегося режима нужно вести расчет до большого значения времени T. Такая разномасштабность делает задачу весьма трудной для численного расчета.

Жесткие системы ОДУ. Все перечисленные модели математически сводятся к решению задачи Коши для системы обыкновенных дифференциальных уравнений:

$$\frac{d\mathbf{u}}{dt} = \mathbf{f}(\mathbf{u}, t), \quad \mathbf{u}, \mathbf{f} \in \mathbb{R}^M, \quad \mathbf{u}(0) = \mathbf{u}_0, \quad 0 \le t \le T. \tag{1}$$

Еще в конце 1940-х годов выяснилось, что для решения систем (1), полученных в результате моделирования описанных выше задач, явные схемы Рунге-Кутты, Адамса и другие непригодны: они требуют нереалистично малого шага, а зачастую и нереалистично большой разрядности чисел для проведения расчетов. В мировой литературе принято называть такие задачи жесткими. Исторически наиболее раннее определение жесткости, предложенное Кертиссом и Хиршфельдером в 1952 году [5], звучит следующим образом: *"Жесткие уравнения – это уравнения, для которых определенные неявные методы, в частности ФДН, дают лучший результат, обычно несравненно более хороший, чем явные методы"*.

Обычно трудности в решении систем (1) принято разделять на несколько классов в зависимости от состава спектра матрицы Якоби правой части $\mathbf{f}_u$. Если все собственные значения λ матрицы Якоби $\mathbf{f}_u$ в комплексной плоскости попадают во внутреннюю часть круга некоторого радиуса l, такого, что $lT \sim 1$, то задачу будем называть мягкой. В этом случае система легко решается классическими явными методами. Будем говорить, что задача является жесткой, если в спектре ее матрицы Якоби присутствуют собственные числа с большими отрицательными действительными частями, $\mathrm{Re}\,(\lambda T) \ll -1$. Тогда решение быстро затухает в узком пограничном слое и переходит в некоторое предельное (интегральные кривые быстро сходятся). Если в

спектре матрицы Якоби задачи присутствуют собственные числа с большими положительными действительными частями, $\mathrm{Re}\,(\lambda T) \gg 1$, то такие задачи являются плохо обусловленными. В этом случае интегральные кривые быстро расходятся, решение сильно меняется при небольшом изменении начальных данных. Наконец, если в спектре присутствуют собственные числа с большими мнимыми частями, задачи называют быстро осциллирующими.

Отметим, что в случае систем многих уравнений возможны комбинации различных типов сложности: часть компонент может быть жесткой, часть – плохо обусловленной. Кроме того, тип трудности системы может меняться в разные моменты времени. Одна и та же компонента может иметь участки жесткости и плохой обусловленности.

Требования к численным методам. Для решения жестких задач предъявляются специфичные требования к численным методам. Во-первых, очевидно, численный метод должен иметь хорошую аппроксимацию, т.е. погрешность численного решения Δ должна стремиться к нулю при уменьшении шага сетки $h \to 0$. Весьма желательным является убывание ошибки по степенному закону, $\Delta = O(h^p)$, с не слишком низким порядком p. Второй важной характеристикой численного метода является устойчивость. В случае жесткой системы в спектре матрицы Якоби присутствуют элементы с $Re\lambda \ll 0$, при этом точное решение быстро затухает. Численное решение в этом случае также должно обеспечивать достаточно быстрое затухание. Простейшее исследование устойчивости разностных схем традиционно проводится на линейном тесте Далквиста:

$$\frac{du}{dt} = \lambda u, \quad u(0) = 1. \tag{2}$$

Решением этой задачи является $u(t) = e^{\lambda t}$, т.е. при $\mathrm{Re}\,\lambda < 0$ и $t \to +\infty$ имеем экспоненциальное затухание. Для любой линейной разностной схемы переход на новый слой по времени с шагом τ при решении задачи (2) будет иметь вид $\hat{u} = R(z)u$, $z = \lambda\tau$. Множитель $R(z)$ называют функцией устойчивости. Введем следующие определения.

Определение 1 (Далквист, 1963)**.** *Схема называется A-устойчивой, если $|R(z)| \le 1$ при $\mathrm{Re}\, z \le 0$.*

Ненарастание численной ошибки на последующих шагах является минимальным требованием, которому должна удовлетворять разностная схема, пригодная для решения жестких задач. Однако A-устойчивость не гарантирует достаточно быстрого затухания жестких компонент в задачах большой жесткости, когда $\mathrm{Re}\, z \ll -1$.

Определение 2 (Ил, 1969)**.** *Схема называется L-устойчивой, если она A-устойчива, и $R(z) \to 0$ при $z \to \infty$.*

Для уточнения скорости затухания жестких компонент вводится понятие Lp-устойчивости:

Определение 3 (Калиткин, 1981)**.** *Схема называется Lp-устойчивой, если она A-устойчива, и $R(z) = O(z^{-p})$ при $z \to \infty$.*

В ряде случаев свойство A-устойчивости оказывается слишком сильным: действительно, если метод является устойчивым во всей левой комплексной полуплоскости, то он в частности должен быть устойчивым и при наличии высокочастотных колебаний $|\,\mathrm{Im}\, z| \gg 1$. Если о задаче известно, что она является жесткой, но колебания в ней не присутствуют, то требование A-устойчивости можно ослабить и потребовать $A(\alpha)$-устойчивости:

Определение 4 (Видлунд, 1967)**.** *Схема называется$A(\alpha)$-устойчивой, если $|R(z)| \leq 1$ внутри сектора $S_\alpha = \{z;\ |arg(-z)| \leq \alpha, z \neq 0\}$.*

Аналогично определяются $L(\alpha)$-устойчивость и $Lp(\alpha)$-устойчивость. Этим требованиям удовлетворяют многие неплохие методы, пригодные для решения жестких задач.

Часто бывает априори известно, что точное решение задачи является монотонным. В этом случае желательно, чтобы поведение численного решения было качественно похоже на поведение точного решения. Сформулируем это свойство следующим образом.

Определение 5 (Калиткин, 1981)**.** *Схема называется t-монотонной, если она A-устойчива, и $R(z) > 0$ при вещественном отрицательном z.*

Наличие свойства t-монотонности схемы позволяет избежать получения пилообразного решения на жестких участках.

Точность расчетов. Оценка погрешности выдаваемого результата является неотъемлемой частью производимых расчетов. Более того, с ростом производительности компьютеров трудоемкость расчетов уходит на второй план, и именно надежность приобретает все большее значение. В настоящее время самым надежным методом оценки погрешности является метод Ричардсона, требующий проведения расчетов на последовательности сгущающихся сеток [6]. Как известно, метод применим в том случае, когда при расчетах на сгущающихся сетках значения погрешностей ложатся на прямую, наклон которой соответствует порядку точности метода.

Классическая формулировка метода Ричардсона рассчитана на равномерные сетки. В монографии [7] рассмотрено применение рекуррентного уточнения по Ричардсону к расчетам на квазиравномерных сетках, адаптированных к конкретным задачам. Метод сгущения сеток полностью распространяется на квазиравномерные сетки. Однако априорные знания об особенностях задачи не всегда удается получить.

Программы с автоматическим выбором шага в жестких задачах не всегда работают удовлетворительно. Методические расчеты показывают, что фактическая погрешность может отличаться от запрашиваемой пользователем на 2-3 порядка [8].

Таким образом, если точное решение задачи является достаточно гладким, и есть возможнось провести расчет на последовательности сгущающихся сеток, использование метода Ричардсона является предпочтительным способом получения оценки погрешности. При этом целесообразна визуализация расчетов: вместе с полученным на самой подробной сетке результатом пользователю выдается зависимость погрешности в выбранной норме от числа узлов сетки. Эту зависимость нужно вывести на график в двойном логарифмическом масштабе. Если на подробных сетках происходит выход на асимптотику, и зависимость приближается к прямой линии с углом наклона к оси асцисс $\mathrm{tg}\,\alpha = -p$, где p – теоретический порядок точности метода, то полученной оценке погрешности можно доверять.

Актуальность темы исследования. Моделированию и расчету жестких систем посвящено множество работ. Основные методы и подходы были предложены в работах Далквиста, Гиршфельдера, Кертисса, Ваннера, Розенброка и их последователей. Важнейшие зарубежные работы собраны в монографии [5]. Однако в этой книге мало представлены работы российских авторов.

В разные годы этой задачей занимались Е.А. Альшина, О.Б. Арушанян, В.В. Бобков, А.Н. Заворин, Н.Н. Калиткин, Е.Б. Кузнецов, Г.Ю. Куликов, Е.А. Новиков, Л.М. Скворцов, С.С. Филиппов, В.И. Шалашилин, П.Д. Ширков.

Упомянем также ряд работ в данной области за последние несколько лет. Статьи [9–11] и другие работы этих авторов посвящены исследованию семейства явно-неявных схем Розенброка с комплексными коэффициентами. Эти схемы обладают высокой устойчивостью и являются экономичными, так как не требуют итераций. В частности, построены схемы, имеющие 4 порядок точности на чисто дифференциальных и 3 на дифференциально-алгебраических задачах индекса 1.

В статьях [12, 13] выводится и исследуется новый класс явно-неявных схем, названных авторами ABC-схемами. Эти схемы одностадийные и одношаговые; они обладают A- и L-устойчивостью и также не содержат итераций: для перехода на новый временной слой достаточно решить одну линейную систему.

В работах [14, 15] строятся явно-неявные схемы типа Розенброка (не требующие итераций), обладающие функциями устойчивости, эквивалентными диагонально-неявным ($DIRK$) и жестко точным [5] полностью неявным ($FIRK$) схемам Рунге-Кутты на линейных автономных и неавтономных задачах.

Работы [16–18] посвящены построению и исследованию методов типа Розенброка, названных авторами (m, k)-методами (k – количество вычислений правой части, m — число стадий схемы). Независимо от числа стадий, на каждом шаге выполняется лишь одно вычисление матрицы Якоби. При этом, в частности, был построен L-устойчивый метод 4-го порядка точности.

Статьи [19–21] посвящены построению явных схем Рунге-Кутты, адаптированных для решения жестких систем обыкновенных дифференциальных уравнений определенного типа. Показано, что сконструированные подобным образом схемы не только позволяют решать жесткие задачи, на которых классические оптимальные явные схемы [22] разваливаются, но и позволяют при значительно меньших вычислительных затратах достичь точности, сопоставимой, например, с диагонально-неявными схемами Рунге-Кутты.

Множество работ посвящено исследованию диагонально-неявных схем Рунге-Кутты, впервые предложенных в работе [23]. Их конструирование и эффективная реализация (включая упрощенные ньютоновские итерации и автоматический выбор шага интегрирования) обсуждается, например, в монографии [5], а также работах [24–26].

Отдельное внимание в литературе уделяется аспектам эффективной реализации неявных схем и контролю погрешности получаемого решения. В работах [27, 28] исследуются методы решения нелинейных алгебраических систем, возникающих при использовании неявных схем Рунге-Кутты: метод простых итераций, классический и модифицированный метод Ньютона. Рассмотрены случаи тривиального и нетривиального предиктора (начального значения для итераций на новом временном слое). В работах [29] и [30] предложен алгоритм автоматического выбора шага интегрирования для явных и неявных методов и построенный на его основе алгоритм контроля локальной и глобальной ошибки. В частности в этих работах демонстрируется, что контроль только локальной погрешности не позволяет надежно обеспечить заданную пользователем желаемую точность численного решения. Для одношаговых методов типа Розенброка построение алгоритма оценки глобальной погрешности исследуется в работе [31].

Целью данной работы является разработка пакета программ для моделирования бегущей тепловой волны и процесса горения метана в воздухе. Вместе с решением программа должна предоставлять пользователю оценку его погрешности, а также давать возможность визуального контроля достоверности полученных результатов. Для достижения поставленной цели необходимо было решить следующие задачи:

1. Разработать полностью неявные Lp-устойчивые схемы Рунге-Кутты порядка точности p от 1 до 4, допускающие экономичную реализацию по сравнению с классическими полностью неявными схемами Рунге-Кутты.
2. Провести исследование возможности автономизации исходной задачи методом перехода к длине дуги. Для преобразованной новой задачи найти оценку точности получаемого решения по методу Ричардсона.
3. Реализовать пакет программ, позволяющий проводить расчет модельных задач с использованием существующих, а также разработанных в рамках данной работы алгоритмов. Программа должна предоставлять пользователю возможность визуальной оценки достоверности полученного результата.
4. Применить разработанный пакет программ для моделирования бегущей тепловой волны, расчета транзисторного усилителя и процесса горения метана в воздухе.

Научная новизна. В рамках данной работы выведен новый подкласс разностных схем из семейства полностью неявных схем Рунге-Кутты порядка точности p от 1 до 4. Эти схемы обладают Lp–устойчивостью, а по трудоемкости вычислений сравнимы с диагонально-неявными схемами Рунге-Кутты, обладающими лишь $L1$-устойчивостью. Исследован известный метод автономизации системы ОДУ методом перехода к длине дуги. Показано, что при новом аргументе возможно применение сгущения сеток по методу Розенброка для получения апостериорной оценки погрешности. Выявлены области применимости метода. Для решения систем нелинейных уравнений, возникающих при интегрировании жестких систем ОДУ неявными схемами, разработано простое обобщение, основанное на методе Ньютона, но обладающее гораздо лучшей областью сходимости. На основе построенных методов создан пакет программ для решения жестких и плохо обусловленых задач.

Практическая ценность работы. Разработанный пакет программ можно применять для численного решения практических задач, а также использовать в процессе подготовки специалистов по математическому моделированию и вычислительной математике. Построенные методы использовались для моделирования задач из физики плазмы, химической кинетики и теории электрических цепей. Однако они могут быть применены и для более широкого круга задач, сводящихся к решению задачи Коши для жесткой или плохо обусловленной системы уравнений.

Апробация работы. Результаты диссертации докладывались на конференции “Современные проблемы вычислительной математики и математической физики” памяти академика А.А. Самарского к 95-летию со дня рождения (Москва, 16-17 июня 2014). Также по материалам диссертации были сделаны доклады на семинарах кафедры вычислительной математики ВМК МГУ (декабрь 2013), Института прикладной математики им. М.В. Келдыша РАН и кафедры матема-

тики физического факультета МГУ (апрель 2014).

На защиту выносятся следующие результаты:

1. Предложены и обоснованы оптимальные обратные схемы Рунге-Кутты с 1 по 4 порядок точности, позволяющие решать широкий круг существенно нелинейных жестких и дифференциально-алгебраических задач индекса 1. Разработан эффективный алгоритм их реализации, включая усеченный многомерный метод Ньютона для решения систем нелинейных уравнений.
2. Создан пакет программ для расчета жестких систем ОДУ и дифференциально-алгебраических систем индекса 1, выдающий апостериорную асимпторически точную оценку погрешности.
3. Проведены расчеты моделей, описывающих прикладные задачи из плазменной теплопроводности, химической кинетики и нелинейной радиотехники.

Личный вклад автора. Автор под руководством Н.Н. Калиткина построил оптимальные обратные схемы Рунге-Кутты, разработал численный алгоритм их реализации, получил коэффициенты в форме матрицы Бутчера, выполнил адаптацию к дифференциально-алгебраическим задачам и трансформировал эти схемы для выбора длины дуги в качестве аргумента.

Автор самостоятельно предложил оригинальное усечение ньютоновских итераций, повысившее надежность алгоритма решения систем нелинейных алгебраических уравнений; на основе всех разработанных алгоритмов написал пакет программ для расчета жестких задач с одновременным вычислением апостериорной асимптотически точной оценки погрешности, и с помощью разработанного пакета выполнил моделирование практических задач.

Публикации. По теме диссертации всего опубликовано 7 работ в журналах, входящих в список ВАК. Среди них следующие рецензируемые работы:

1. *Н.Н. Калиткин, И.П. Пошивайло*, **Обратные Ls-устойчивые схемы Рунге-Кутты** // ДАН, 2012 г., т.442, №2, с.175-180.
2. *Н.Н. Калиткин, И.П. Пошивайло*, **Гарантированная точность при решении задачи Коши методом длины дуги** // ДАН, 2013 г., т.452, №5, с.499-502.
3. *И.П. Пошивайло*, **Усеченный многомерный метод Ньютона** // Математическое моделирование, 2012 г., т.24, №1, с.103-108.
4. *Н.Н. Калиткин, И.П. Пошивайло*, **Вычисления с использованием обратных схем Рунге-Кутты** // Математическое моделирование, 2013 г., т.25, №10, с.79-96.
5. *Н.Н. Калиткин, И.П. Пошивайло*, **Решение задачи Коши для жестких систем с гарантированной точностью методом длины дуги** // Математическое моделирование, 2014 г., т.26, №7, с.3-18.
6. *Н.Н. Калиткин, И.П. Пошивайло*, **Определение кратности корня нелинейного алгебраического уравнения** // ЖВМиМФ, 2008 г., т.48, №7, с.1181-1186.
7. *Н.Н. Калиткин, И.П. Пошивайло*, **О вычислении простых и кратных корней нелинейного уравнения** // Математическое моделирование, 2008 г., т.20, №7, с.57-64.

Структура и объем работы. Диссертация состоит из введения, пяти глав, заключения и списка литературы. Общий объем диссертации 89 стр., рисунков 24, таблиц 7. Список литературы включает 62 наименования.

Глава 1

Оптимальные обратные схемы Рунге-Кутты.

В данной главе строится новый подкласс разностных схем – оптимальные обратные схемы Рунге-Кутты. Чтобы привести необходимые для дальнейшего изложения сведения и ввести требуемые обозначения, напомним общий вид формул Рунге-Кутты в форме Бутчера.

1.1. Схемы Рунге-Кутты.

Обозначим решение на исходном шаге через $\mathbf{u}$, а на новом через $\hat{\mathbf{u}}$ ($\mathbf{u}, \hat{\mathbf{u}} \in \mathbb{R}^M$). Схемы Рунге-Кутты являются одношаговыми (для перехода от момента времени t_n к t_{n+1} нужно знать лишь значение в исходном узле сетки $\mathbf{u}(t_n)$) и s-стадийными. В общем виде их формулы записываются следующим образом:

$$\hat{\mathbf{u}} = \mathbf{u} + \tau \sum_{k=1}^{s} b_k \mathbf{w}_k, \quad \mathbf{w}_k = \mathbf{f}(\mathbf{u} + \tau \sum_{l=1}^{L} a_{kl} \mathbf{w}_l, t + \tau c_k). \tag{1.1}$$

Матрица коэффициентов $\{a_{kl}\}$ называется матрицей Бутчера. Векторы коэффициентов $\{c_k\}$ и $\{b_k\}$ можно рассматривать как столбцы, дополняющие квадратную матрицу Бутчера до прямоугольной.

Обычно выделяют следующие классы схем. Если $L = k - 1$, т.е. лишь находящиеся ниже главной диагонали элементы матрицы Бутчера отличны от нуля, схемы называют явными (ERK – explicit Runge-Kutta). В них переход на новый слой происходит по явным формулам. Явные схемы просты в реализации, а трудоемкость вычислений мала.

Если $L = k$, то схемы называют диагонально-неявными (DIRK – diagonal implicit Runge-Kutta). В таких схемах на каждой стадии для нахождения очередного $\mathbf{w}_k$ необходимо решить систему алгебраических уравнений порядка M. Это делается итерационными методами. Поэтому диагонально-неявные схемы существенно более трудоемкие, чем явные.

Если же $L = s$, схемы называют полностью неявными схемами РК (FIRK – fully implicit Runge-Kutta). В них вычисление всех стадий происходит не последовательно, а одновременно. Для нахождения всех $\mathbf{w}_k$ требуется решить систему алгебраических уравнений порядка sM. Трудоемкость полностью неявных схем много больше, чем диагонально-неявных.

1.1.1. Явные схемы.

Схемы ERK с $s \leq 4$ стадиями хорошо изучены. При этом нельзя построить схему порядка точности $p > s$, но можно построить схему с $p = s$. Именно последние схемы и будем далее рассматривать. Для автономных задач при $s = 1$ имеется единственная схема, при $s = 2$ – однопараметрическое семейство схем, при $s = 3$ и 4 – двупараметрические семейства схем [22]. Наложим дополнительные требования: интерполяционности схемы и минимального числа слагаемых в невязке схемы. Тогда в каждом семействе выделяется единственная оптимальная схема. В оптимальных схемах в матрице коэффициентов Бутчера ненулевыми оказываются только элементы нижней кодиагонали $a_{k,k-1}$, где $2 \leq k \leq s$. Из условий аппроксимации следует, что $a_{k,k-1} = c_k$, где c_k – дополнительный столбец матрицы Бутчера (коэффициенты при τ). Тогда, обозначив $a_{k,k-1} = c_k = a_k$, оптимальные явные схемы РК можно записать следующим образом:

$$\hat{\mathbf{u}} = \mathbf{u} + \tau \sum_{k=1}^{s} b_k \mathbf{w}_k, \ s \leq 4; \quad \mathbf{w}_k = \mathbf{f}(\mathbf{u} + \tau a_k \mathbf{w}_{k-1}, t + \tau a_k). \tag{1.2}$$

Значения коэффициентов a_k, b_k приведены в табл. 1.1.

Таблица 1.1: Коэффициенты оптимальных явных схем РК (1.2).

k	s							
	1		2		3		4	
	b_k	a_k	b_k	a_k	b_k	a_k	b_k	a_k
1	1	0	1/4	0	2/9	0	1/6	0
2			3/4	2/3	3/9	1/2	2/6	1/2
3					4/9	3/4	2/6	1/2
4							1/6	1

1.1.2. О схемах высших порядков.

Оптимальные явные схемы Рунге-Кутты до $p = 4$ порядка точности включительно построены в работе [22]. Для $s > 4$ оптимальные схемы построить не удается: при $p = 5$ или 6 необходимое число стадий схемы $s = p + 1$. При $s \geq 7$ требуется число стадий $s \geq p + 2$. Эти ограничения на максимальный порядок точности называют порогами Бутчера. В [3] реализована программа DOPRI5, созданная на основе 7-стадийной схемы Дорманда-Принса 5-го порядка точности. В работах [32] и [33] построены явные 7-стадийные схемы Рунге-Кутты 6-го порядка точности.

При построении неявных схем не удается обеспечить одновременно высокий порядок аппроксимации и хорошие свойства устойчивости. Поэтому в неявных решателях для жестких систем обычно используют схемы не более чем пятого порядка точности: решатель RADAU5 из [5] реализует неявный метод Рунге-Кутты 5 порядка, в системе MATLAB решатель ode23s – схему Розенброка 2-3 порядка точности, решатель ode15s – 1-5 порядка точности.

Кроме того, построение схем более высокого порядка представляется нецелесообразным ввиду следующих соображений. Любой расчет имеет смысл в том случае, когда вместе с результатом выдается надежная оценка его погрешности. В настоящее время самым надежным методом оценки погрешности является метод Ричардсона, требующий проведения расчетов на последовательности сгущающихся сеток. Как известно, метод применим в том случае, когда при расчетах на сгущающихся сетках значения погрешностей ложатся на прямую, наклон которой соответствует порядку точности метода. Таким образом, в диапазоне погрешностей $10^{-2}...10^{-14}$ для схемы 4-го порядка и серии сгущающихся вдвое сеток на графике зависимости погрешности решения от количества узлов сетки получится 10 точек. Этого достаточно для апостериорной оценки точности решения, однако данные расчеты справедливы лишь для очень простых задач и редко достигаются на практике. Для схемы 6-го порядка точности на график ляжет в лучшем случае 6 точек, чего уже может оказаться недостаточно для достоверного выявления прямого участка.

1.1.3. Обратные схемы.

Обратные схемы РК построим следующим образом. Рассмотрим движение в обратном направлении. Напишем явную схему для перехода $\hat{t} \to t$. Для этого в (1.2) нужно поменять местами $t \leftrightarrow \hat{t}$, $\mathbf{u} \leftrightarrow \hat{\mathbf{u}}$ и изменить знак τ. Получим следующие оптимальные обратные схемы (“backward optimal Runge-Kutta”, BORK):

$$\hat{\mathbf{u}} = \mathbf{u} + \tau \sum_{k=1}^{s} b_k \hat{\mathbf{w}}_k, \quad \hat{\mathbf{w}}_k = \mathbf{f}(\hat{\mathbf{u}} - \tau a_k \hat{\mathbf{w}}_{k-1}, \hat{t} - \tau a_k). \tag{1.3}$$

Коэффициенты этих схем те же, что и для схем (1.2). Очевидно, схемы (1.3) также имеют точность $O(\tau^s)$, $s \le 4$. Заметим, что (1.3) для $s = 1$ является известной обратной схемой Эйлера.

Рекурсивная запись. Получим рекурсивную запись обратных схем (1.3). Для простоты записи ограничимся автономными системами: $\mathbf{f}(\mathbf{u}, t) \equiv \mathbf{f}(\mathbf{u})$. Тогда сдвиг $\hat{\mathbf{w}}_1$ равен $\mathbf{f}(\hat{\mathbf{u}})$. $\hat{\mathbf{w}}_2$ выражается уже как рекурсивная функция: $\hat{\mathbf{w}}_2 = \mathbf{f}(\hat{\mathbf{u}} - \tau a_2 \mathbf{f}(\hat{\mathbf{u}}))$. То же самое справедливо для $\hat{\mathbf{w}}_3$ и $\hat{\mathbf{w}}_4$. Тогда обратная схема 4-го порядка примет следующий рекурсивный вид:

$$\begin{gathered} \hat{\mathbf{u}} = \mathbf{u} + \tau \mathbf{F}(\hat{\mathbf{u}}), \\ \mathbf{F}(\hat{\mathbf{u}}) \equiv b_1 \mathbf{f}(\hat{\mathbf{u}}) + b_2 \mathbf{f}(\hat{\mathbf{u}} - \tau a_2 \mathbf{f}(\hat{\mathbf{u}})) + \\ b_3 \mathbf{f}(\hat{\mathbf{u}} - \tau a_3 \mathbf{f}(\hat{\mathbf{u}} - \tau a_2 \mathbf{f}(\hat{\mathbf{u}}))) + b_4 \mathbf{f}(\hat{\mathbf{u}} - \tau a_4 \mathbf{f}(\hat{\mathbf{u}} - \tau a_3 \mathbf{f}(\hat{\mathbf{u}} - \tau a_2 \mathbf{f}(\hat{\mathbf{u}})))). \end{gathered} \tag{1.4}$$

Здесь коэффициенты $\{a_k\}$, $\{b_k\}$ берутся из четвертой пары столбцов табл. 1.1. Чтобы получить схему 3-го порядка, нужно в (1.4) отбросить слагаемое с b_4, а коэффициенты a_k, b_k взять из третьей пары столбцов табл. 1.1. Аналогично получаются схемы 2-го и 1-го порядка.

Схема (1.4) представляет собой нелинейную алгебраическую систему порядка M для определения $\hat{\mathbf{u}}$, записанную через рекурсивные функции. Трудоемкость ее решения соответствует одной стадии диагонально-неявных схем. Тем самым построенная схема действительно имеет невысокую по сравнению с другими неявными схемами трудоемкость.

Переход к форме Бутчера. Покажем, что указанные обратные схемы (1.3) являются подклассом схем FIRK. Действительно, в (1.3) в формуле для $\hat{\mathbf{w}}_k$ выразим $\hat{\mathbf{u}}$ через $\mathbf{u}$, тогда

$$\hat{\mathbf{w}}_k = \mathbf{f}(\mathbf{u} + \tau \sum_{l=1}^{s} b_l \hat{\mathbf{w}}_l - \tau a_k \hat{\mathbf{w}}_{k-1}, \hat{t} - \tau a_k). \tag{1.5}$$

Теперь, переобозначив $\hat{\mathbf{w}}_k \to \mathbf{w}_k$ и заменив $\hat{t}$ на $t + \tau$, получим:

$$\hat{\mathbf{u}} = \mathbf{u} + \tau \sum_{k=1}^{s} b_k \mathbf{w}_k, \quad \mathbf{w}_k = \mathbf{f}\left(\mathbf{u} + \tau \left[\sum_{l=1}^{s} b_l \mathbf{w}_l - a_k \mathbf{w}_{k-1}\right], t + \tau(1 - a_k)\right), \tag{1.6}$$

что совпадает с общим видом формул для схем РК (1.1). Кроме того, здесь $L = s$. Следовательно схемы (1.3) действительно являются подклассом FIRK. Коэффициенты матрицы Бутчера для них легко выражаются через коэффициенты оптимальных явных схем. Для схем 2-4 порядка точности они приведены в табл. 1.2.

Таблица 1.2: Коэффициенты оптимальных обратных схем (1.3) 2–4 порядка точности в форме Бутчера.

$s=2$				$s=3$					$s=4$					
a_{mk}		b_m	c_m	a_{mk}			b_m	c_m	a_{mk}				b_m	c_m
1/4	3/4	1/4	1	2/9	3/9	4/9	2/9	1	1/6	2/6	2/6	1/6	1/6	1
-5/12	3/4	3/4	1/3	-5/18	3/9	4/9	3/9	1/2	-2/6	2/6	2/6	1/6	2/6	1/2
				2/9	-15/36	4/9	4/9	1/4	1/6	-1/6	2/6	1/6	2/6	1/2
									1/6	2/6	-4/6	1/6	1/6	0

1.1.4. Моно-неявные схемы Рунге-Кутты.

Впервые подкласс неявных методов Рунге-Кутты, допускающих рекурсивную форму записи и являющихся неявными только относительно решения на новом временном слое $\hat{\mathbf{u}}$, был предложен в работах [34, 35]. Эти схемы были названы моно-неявными методами Рунге-Кутты (Mono-Implicit Runge-Kutta, MIRK). Ввиду возможности эффективной реализации, методы из семейства MIRK представляются перспективным направлением. В настоящее время в [36] и последующих работах этих авторов исследован класс схем, названный неявными гнездовыми методами (Nested Implicit Runge-Kutta, NIRK), который также является подклассом схем MIRK. В работе [30] предложен алгоритм контроля локальной и глобальной ошибок для методов NIRK 4-го и 6-го порядка точности. Из преимуществ схем BORK следует отметить их высокий порядок L-затухания (см. раздел 1.1.6), экономичность – ввиду того, что на каждой вложенной стадии выполняется лишь одно вычисление правой части, и минимальное число членов в невязке методов (как это было показано в [33] для оптимальных явных схем РК).

1.1.5. Неоптимальные обратные схемы.

Запись полностью неявных схем РК в виде одной стадии (1.4), в которой правая часть записана через рекурсивные функции, возможна для произвольных явных схем Рунге-Кутты. Однако

общий вид этих схем не представляет практических выгод по сравнению с оптимальными схемами. Можно рекомендовать только одну схему второго порядка точности ($s = p = 2$), которая является обратной к модифицированному явному методу Эйлера [37] (в англоязычной литературе его обычно называют “explicit midpoint method”). Этот метод также входит в семейство явных двухстадийных схем Рунге-Кутты (1.1). Формула для явного метода выглядит следующим образом:

$$\hat{\mathbf{u}} = \mathbf{u} + \tau \mathbf{f}(\mathbf{u} + \frac{1}{2}\tau \mathbf{f}(\mathbf{u}, t), t + \frac{1}{2}\tau)). \tag{1.7}$$

Соответствующая обратная схема (Backward Midpoint method, BMP) получается аналогично оптимальным обратным схемам:

$$\hat{\mathbf{u}} = \mathbf{u} + \tau \mathbf{f}(\hat{\mathbf{u}} - \frac{1}{2}\tau \mathbf{f}(\hat{\mathbf{u}}, t + \tau), t + \frac{1}{2}\tau)). \tag{1.8}$$

Эта схема имеет очень простой и красивый вид, а в ее правой части содержится только одна рекурсивная функция (а не две, как во всех остальных обратных схемах второго порядка). Схема (1.8) интерполяционна, а по величине остаточного члена лишь немного уступает оптимальной схеме (1.3).

Схема BMP (1.8) также сводится к классу полностью неявных схем Рунге-Кутты (FIRK). Действительно, положим $\hat{\mathbf{u}} = \mathbf{u} + \tau \mathbf{w}_2$. Тогда:

$$\mathbf{w}_2 = \mathbf{f}\left(\hat{\mathbf{u}} - \frac{1}{2}\tau \mathbf{f}(\hat{\mathbf{u}}, t + \tau), t + \frac{1}{2}\tau\right) = \mathbf{f}\left(\mathbf{u} + \tau \mathbf{w}_2 - \frac{1}{2}\tau \mathbf{f}(\mathbf{u} + \tau \mathbf{w}_2, t + \tau), t + \frac{1}{2}\tau\right) = \mathbf{f}\left(\mathbf{u} + \tau \mathbf{w}_2 - \frac{1}{2}\tau \mathbf{w}_1, t + \frac{1}{2}\tau\right),$$

где $\mathbf{w}_1 = \mathbf{f}(\mathbf{u} + \tau \mathbf{w}_2, t + \tau)$. Получаем FIRK со следующей матрицей Бутчера:

Таблица 1.3: Коэффициенты схемы (1.8) в форме Бутчера.

a_{mk}		b_m	c_m
0	1	0	1
-1/2	1	1	1/2

1.1.6. Устойчивость.

Для жестких систем устойчивость традиционно исследуется на тесте Далквиста – одном уравнении с $f(u) = \lambda u$. Функцией устойчивости называется множитель перехода на новый шаг $R(z) = \hat{u}/u$, $z = \tau\lambda$. Для явных схем (1.2) функция устойчивости есть полиномиальная аппроксимация $\exp(z)$:

$$R_s(z) = \sum_{k=0}^{s} z^k/k!. \tag{1.9}$$

Для обратных схем (1.3) функция устойчивости есть Паде-аппроксимация $\exp(-z)$:

$$R_s(z) = \left[\sum_{k=0}^{s} (-z)^k/k!\right]^{-1}. \tag{1.10}$$

Такой вид обеспечивает хорошее затухание жестких компонент, поскольку для них $|z| = |\lambda\tau|$ велик.

Заметим, что для полностью неявных s-стадийных схем РК $R(z)$ есть отношение двух многочленов. При этом в знаменателе стоит многочлен степени s, а в числителе – многочлен степени не более s. Поэтому затухание жестких компонент будет не лучше (а зачастую хуже), чем для обратных схем РК.

Для решения жестких систем обычно требуют A-устойчивости схем: $|R(z)| \leq 1$ в левой полуплоскости комплексного z. Из (1.9) видно, что для явных схем это условие катастрофически нарушается, и они непригодны для жестких задач. Для обратных схем с $s = 1$ и 2 легко доказывается строгая A-устойчивость. Поэтому согласно [38], они строго Ls-устойчивы. Это означает хорошую надежность данных схем на жестких задачах.

Отмечено [10], что для функции устойчивости (1.10) при $s = 3$ и 4 нет строгой A-устойчивости. Слева вблизи мнимой оси z имеются участки с $|R(z)| > 1$. При этом реализуется лишь $L(\alpha)s$-устойчивость, где α – угол мнимой оси с лучом, отсекающим эти участки. Для $s = 3$ этот угол очень мал ($\alpha \approx 0.3°$), а $\max|R(z)| \approx 1$. Поэтому надежность обратной схемы 3-го порядка можно считать неплохой.

Для $s = 4$ угол $\alpha \approx 6°$, а на мнимой оси достигается $\max|R(z)| = 2$, что отнюдь не мало. Поэтому обратная схема 4-го порядка может оказаться не вполне надежной; то же относится к двухстадийной комплексной схеме [9].

1.1.7. Интерполяционность.

Одностадийная явная схема Эйлера и ее обратная схема одновременно являются интерполяционными. В самом деле, для явной схемы $a_{11} = 0$ и $b_1 = 1$, а для обратной схемы b_1 сохраняет то же значение, а единственный коэффициент матрицы Бутчера $a_{11} = 1$.

Однако уже для двух стадий обратная схема не может быть интерполяционной. В самом деле, для явной двухстадийной схемы точности $O(\tau^2)$ общее решение является однопараметрическим семейством. При этом матрица Бутчера (a_{mk}) и столбец (b_m) имеют следующий вид:

$$(a_{mk}) = \begin{pmatrix} 0 & 0 \\ a & 0 \end{pmatrix}, (b_m) = \begin{pmatrix} 1 - \dfrac{1}{2a} \\ \dfrac{1}{2a} \end{pmatrix}.$$

Здесь для интерполяционности достаточно взять $\dfrac{1}{2} \leq a \leq 1$. Для обратной схемы столбец (b_m) будет таким же, а матрица Бутчера (α_{mk}) будет содержать следующие элементы:

$$(\alpha_{mk}) = \begin{pmatrix} b_1 & b_2 \\ 1 - a - \dfrac{1}{2a} & b_2 \end{pmatrix}.$$

Видно, что три элемента равны b_1 или b_2; тем самым, они интерполяционны при том же условии, что и явная схема. Однако α_{21} будет отрицательным при любом $a > 0$, что является нарушением условия интерполяционности. Наименьшее нарушение будет при $a = \dfrac{1}{\sqrt{2}}$, когда $\alpha_{21} = 1 - \sqrt{2} \approx -0.41$.

Таким образом, при $s = 2$ (очевидно, и при $s > 2$) прямая и обратная ей схемы не могут быть интерполяционными одновременно. Однако это вряд ли приводит к ухудшению схем. Вполне достаточно требовать, чтобы схема была интерполяционна в смысле либо прямого, либо обратного хода. Например, явные оптимальные схемы интерполяционны в смысле прямого хода; тем самым, они не интерполяционны в смысле обратного хода, причем последнее не сказывается на качестве расчетов.

1.2. Алгоритм.

У полностью неявных s-стадийных схем РК матрица коэффициентов Бутчера полностью заполнена, а сама схема является нелинейной алгебраической системой порядка sM относительно неизвестного $\hat{\mathbf{u}}$. Трудоемкость ее решения быстро увеличивается с увеличением s. Обратные же схемы, записанные в виде (1.4), при любых s являются нелинейной алгебраической системой порядка M. Тем самым, обратные схемы сопоставимы по трудоемкости с диагонально-неявными схемами РК (DIRK), но имеют более высокий порядок L-затухания (схемы DIRK могут иметь лишь первый порядок L-затухания [39]).

Рассмотрим более детально реализацию схем (1.4) для автономной задачи. Переход на новый слой сводится к решению уравнения

$$\hat{\mathbf{u}} - \tau \mathbf{F}(\hat{\mathbf{u}}) - \mathbf{u} = 0, \tag{1.11}$$

где $\mathbf{F}(\hat{\mathbf{u}})$ записано в (1.4). Для решения нелинейных алгебраических систем, возникающих при решении нелинейных дифференциальных уравнений общего вида, традиционно применялся метод простой итерации как наиболее простой в реализации и экономичный. В настоящее время обычно используется та или иная модификация метода Ньютона. Обсудим возможности обоих подходов.

1.2.1. Простые итерации.

Метод простой итерации применяют для решения систем нелинейных уравнений, записанных в виде $x = \phi(x)$: $x_{n+1} = \phi(x_n)$. Этот метод легко реализовать на ЭВМ, и накладные расходы на каждую итерацию обычно наиболее дешевы. Условие сходимости простых итераций имеет вид $||\phi'(x)|| < 1$ в некоторой окрестности корня.

В данном случае в качестве правой части нужно выбрать $\mathbf{u} + \tau \mathbf{F}(\hat{\mathbf{u}})$. Поэтому условие сходимости простых итераций будет иметь вид $\tau ||\mathbf{F}_{\hat{\mathbf{u}}}|| < 1$. Для систем большой жесткости норма матрицы Якоби может быть огромной, так что сходимость будет лишь при неприемлемо малом шаге τ. Поэтому метод простых итераций можно применять лишь для систем малой жесткости, где конкурентоспособными являются даже явные схемы. Как общий метод его использовать невозможно.

1.2.2. Метод Ньютона.

Более надежным и быстрым методом является ньютоновский итерационный процесс. Построим ньютоновский процесс для схемы (1.11):

$$[E - \tau \mathbf{F}_{\hat{\mathbf{u}}}(\hat{\mathbf{u}}^q)]\,(\hat{\mathbf{u}}^{q+1} - \hat{\mathbf{u}}^q) = -[\hat{\mathbf{u}}^q - \tau \mathbf{F}(\hat{\mathbf{u}}^q) - \mathbf{u}], \quad \hat{\mathbf{u}}^0 = \mathbf{u}. \tag{1.12}$$

Каждая итерация сводится к линейной системе относительно $\hat{\mathbf{u}}^{q+1}$. Большая норма матрицы Якоби в этом случае не является препятствием для сходимости метода. Однако возникает вопрос о способе вычисления матрицы Якоби на каждой итерации. Вопрос нетривиальный, так как формула (1.4) для $\mathbf{F}(\hat{\mathbf{u}})$ может иметь весьма сложный вид.

Разностные производные. Матрица Якоби $\mathbf{F}_{\hat{\mathbf{u}}}$ состоит из первых производных по всем компонентам вектора $\hat{\mathbf{u}}$. Для их нахождения выгодно использовать симметричные двухточечные разности с некоторым шагом h; это определяет производные с точностью $O(h^2)$. Поскольку вычисления производятся с плавающей точкой, выгодно выбирать смещенные аргументы для m-й производной как $\hat{u}_m \pm h$.

Смещенные аргументы следует подставлять одновременно всюду, где величина $\hat{\mathbf{u}}$ стоит под знаком функции во всех рекурсивных выражениях (1.11). Такой способ наиболее прост, единообразен и экономичен. Он требует лишь несложных вычислений по явным формулам.

Величину шага h нужно выбирать в соответствие с разрядностью компьютера. При 64-разрядных вычислениях относительная ошибка округления $\varepsilon = 10^{-16}$. Потеря точности при вычислении разностей с $+h$ и $-h$ есть $O(\varepsilon/h)$. Ошибка аппроксимации при замене производной на симметричную разность есть $O(h^2)$. Для корректности вычислений требуется, чтобы ошибка округления была меньше ошибки аппроксимации. Это приводит к условию $\varepsilon < \mathrm{const} \cdot h^3$. Величина константы зависит от вида функции и заранее неизвестна. Поэтому целесообразно полагать $h > \varepsilon^{1/3}$ с некоторым запасом. В работе брался десятикратный запас и $h = 5 \cdot 10^{-5}$; это значение обычно давало хорошие результаты.

Полуаналитические производные. При $s = 1$ рекурсивности нет. При $s \geq 2$ функцию $\mathbf{F}(\hat{\mathbf{u}})$ можно рассматривать как сложную функцию и аналитически сводить вычисление ее якобиана к вычислению якобианов $\mathbf{f}_{\hat{\mathbf{u}}}$. Проанализируем этот способ на простейшем примере – схеме (1.8) для автономной задачи.

Нетрудно проверить, что в этом случае матрица Якоби принимает следующий вид:

$$\mathbf{F}_{\hat{\mathbf{u}}} = \mathbf{f}_{\hat{\mathbf{u}}}(\hat{\mathbf{u}} - \frac{\tau}{2}\mathbf{f}(\hat{\mathbf{u}})) \cdot [E - \frac{\tau}{2}\mathbf{f}_{\hat{\mathbf{u}}}(\hat{\mathbf{u}})]. \tag{1.13}$$

Этот способ более трудоемок, так как во-первых требует дополнительного вычисления матриц Якоби $\mathbf{f}_{\hat{\mathbf{u}}}$ в других точках, и во-вторых включает перемножение матриц. Ошибок округления он не уменьшает, так как в конечном итоге все матрицы $\mathbf{f}_{\hat{\mathbf{u}}}$ приходится вычислять разностным методом. Кроме того, перемножение матриц (которые для жестких задач могут быть плохо обусловленными) ухудшает обусловленность окончательной линейной системы в методе Ньютона. Все эти недостатки усиливаются при увеличении s. Данный метод был опробован и признан менее практичным, чем прямое разностное вычисление $\mathbf{F}_{\hat{\mathbf{u}}}$.

1.2.3. Усечение.

Как известно [40], метод Ньютона очень чувствителен к выбору начального приближения: вблизи корня сходимость метода квадратичная, однако если начальное приближение выбрано неудачно, сходимости может не быть вовсе. Это обстоятельство оказывается особенно существенным при решении жестких задач: первая же итерация может бросить решение далеко от точного значения корня.

Преставляется целесообразным применить следующий простой подход. Рассмотрим обобщение классического ньютоновского итерационнного процесса:

$$\begin{aligned} &D(\mathbf{u}_k)\Delta_k = -\mathbf{F}(\mathbf{u}_k), \\ &\mathbf{u}_{k+1} = \mathbf{u}_k + \theta_k\Delta_k, \\ &0 < \theta_k \leq 1. \end{aligned} \tag{1.14}$$

При $\theta_k = 1$ оно переходит в классический метод Ньютона. Выбирать значение параметра θ_k на каждой итерации нужно таким образом, чтобы

$$|\mathbf{F}(\mathbf{u}_{k+1})| < |\mathbf{F}(\mathbf{u}_k)|\,, \tag{1.15}$$

т.е. модуль правой части $|\mathbf{F}(\mathbf{u}_k)|$ монотонно уменьшался от итерации к итерации. При этом нецелесообразно применять сложные способы выбора θ_k. Вполне достаточно сначала положить $\theta_k = 1$. Если соотношение (1.15) не выполняется, то нужно уменьшить значение θ_k вдвое, снова произвести проверку соотношения и т.д.

1.2.4. Первая итерация.

В неявной схеме Эйлера точности $O(\tau)$ начальным приближением является решение на исходном слое, и даже это приближение оказывается недостаточно хорошим при большом τ. Положение ухудшается для схем более высокого порядка точности. Например, для схемы точности $O(\tau^2)$ выбор $u_0 = u(t)$ фактически эквивалентен тому, что правая часть и производные берутся не в момент t, а в момент $t - \tau/2$, что существенно хуже. Поэтому для схем порядка точности $p > 1$ может оказаться полезным следующий прием: вычисление первой итерации по схеме обратного Эйлера, где нулевое приближение не столь плохое. Лишь со второй итерации целесообразно переходить на основную схему.

1.3. Дифференциально-алгебраические системы.

Такие системы возникают во многих важных приложениях: кинематика механизмов с различными сочленениями деталей, разветвленные электрические цепи и др. Условия сочленения или законы сохранения в точках разветвления являются алгебраическими уравнениями. Алгебраические уравнения можно рассматривать как предельный случай дифференциальных, когда коэффициенты при производных стремятся к нулю. По трудности это эквивалентно сверхжестким дифференциальным системам.

В общем случае дифференциальные и алгебраические уравнения не всегда удается разделить. Поэтому системы обычно записывают в следующем виде:

$$G\frac{d\mathbf{u}}{dt} = \mathbf{f}(\mathbf{u}, t), \quad G = const. \tag{1.16}$$

Здесь G есть постоянная матрица, которая в общем случае является особенной: $0 \leq \operatorname{rank} G \leq M$. Величина $\operatorname{rank} G$ есть число дифференциальных компонент. Если $\operatorname{rank} G = M$, система является чисто дифференциальной, а при $\operatorname{rank} G = 0$ – чисто алгебраической.

Известно, что при решении системы (1.16) неавтономными безытерационными схемами, такими как схемы Розенброка, трудно получить точность выше $O(\tau)$ [11, 41]. В этом случае для получения более высокого порядка точности требуется сначала провести автономизацию системы (1.16), т.е. ввести в систему дополнительную функцию $u_0(t) \equiv t$ и дополнительное дифференциальное уравнение $du_0/dt = 1$. Тогда величина t в правых частях (1.16) заменится на функцию $u_0(t)$, и система станет автономной.

При решении системы (1.16) неявными методами Рунге-Кутты приводить систему к автономному виду не требуется [5]. Применим к задаче (1.16) обратные схемы РК (1.3) с $s \leq 4$. Выведем формулы следующим образом. Пусть матрица G не является особенной, тогда (1.16) можно переписать в следующем виде:

$$\frac{d\mathbf{u}}{dt} = \mathbf{F}(\mathbf{u}, t) \equiv G^{-1}\mathbf{f}(\mathbf{u}, t). \tag{1.17}$$

Запишем для (1.17) обратную схему (1.3) с подстановкой $\mathbf{F}(\mathbf{u}, t)$ в правые части. Умножая эти формулы слева на G, получим окончательный вид оптимальных обратных схем для дифференциально-алгебраических задач индекса 1:

$$\hat{\mathbf{u}} = \mathbf{u} + \tau \sum_{k=1}^{s} b_k \hat{\mathbf{w}}_k, \quad G\hat{\mathbf{w}}_k = \mathbf{f}(\hat{\mathbf{u}} - \tau a_k \hat{\mathbf{w}}_{k-1}, \hat{t} - \tau a_k). \tag{1.18}$$

В формулах (1.18) матрица G уже может быть сингулярной*.

Алгоритм решения системы (1.18) по трудоемкости эквивалентен полностью неявным схемам РК, т.е. он гораздо дороже алгоритма для чисто дифференциальных систем. В самом деле, для сингулярной матрицы G не удается выразить величины $\hat{\mathbf{w}}_k$ в (1.18) через разности $(\hat{\mathbf{u}} - \mathbf{u})$. Поэтому исключить $\hat{\mathbf{w}}_k$ из (1.18) невозможно. Поэтому, чтобы отличать эти схемы от экономичных схем (1.3), будем называть схемы (1.18) “optimal implicit Runge-Kutta”, OIRK.

1.3.1. Сходимость.

Для чисто дифференциальных систем $\operatorname{rank} G = M$, и матрица G неособенная. Тогда ее действительно можно обратить, и записать систему в виде (1.17). Это соответствует записи (1). Следовательно, в этом предельном случае обратные схемы РК обеспечивают точность $O(\tau^s)$.

Рассмотрим второй предельный случай - чисто алгебраические системы. Для них $\operatorname{rank} G = 0$, т.е. $G \equiv 0$. Тогда в схеме (1.18) уравнение для $\hat{\mathbf{w}}_1$ приобретает вид $\mathbf{f}(\hat{\mathbf{u}}) = 0$. Тем самым, для чисто алгебраических систем схема (1.18) дает точное решение.

Для произвольного $\operatorname{rank} G$ справедлива следующая теорема.

*Строгий вывод этих формул для произвольных полностью неявных методов Рунге-Кутты можно найти в [5].

Теорема 1.1. *Если явная схема Рунге-Кутты имеет порядок точности p, то и ее обратная схема имеет такой же порядок точности p на дифференциально-алгебраических системах индекса 1.*

Доказательство. Для любой явной схемы Рунге-Кутты $\mathbf{w}_1 = \mathbf{f}(\mathbf{u})$. Соответственно для ее обратной схемы $\hat{\mathbf{w}}_1 = \mathbf{f}(\hat{\mathbf{u}}) \equiv \mathbf{f}(\mathbf{u} + \tau \sum b_k \hat{\mathbf{w}}_k)$. Это означает, что для обратной (полностью неявной) схемы первую строку матрицы Бутчера составляют коэффициенты b_k, то есть эта первая строка совпадает со столбцом коэффициентов b_k. В [5] на стр.423 доказана теорема, что при таком совпадении схема является жестко точной и обеспечивает порядок точности p на дифференциально-алгебраических системах индекса 1. ■

1.3.2. Уменьшение трудоемкости алгоритма.

Трудоемкость расчета по схемам (1.18) удается уменьшить в том случае, когда дифференциальные и алгебраические компоненты системы разделяются. Тогда система принимает следующий вид:

$$\frac{d\mathbf{u}}{dt} = \mathbf{f}(\mathbf{u}, \mathbf{v}), \quad 0 = \mathbf{g}(\mathbf{u}, \mathbf{v}). \tag{1.19}$$

Введем малый параметр ε и приближенно заменим систему (1.19) следующей системой:

$$\frac{d\mathbf{u}}{dt} = \mathbf{f}(\mathbf{u}, \mathbf{v}), \quad \varepsilon \frac{d\mathbf{v}}{dt} = \mathbf{g}(\mathbf{u}, \mathbf{v}), \quad |\varepsilon| \ll 1. \tag{1.20}$$

Система (1.20) является чисто дифференциальной, и алгоритм ее решения имеет меньшую трудоемкость.

Однако этот случай имеет свои трудности. Для достаточной близости систем (1.19) и (1.20) нужно брать ε весьма малым. Но для очень малого ε возникают значительные ошибки округления при вычислении разностных производных в методе Ньютона. Поэтому метод ε-вложений можно применять только на компьютерах с большой разрядностью.

Глава 2

Оценка погрешности решения в задачах с пограничными слоями.

В прикладных задачах часто встречаются как быстро затухающие компоненты решения, так и быстро нарастающие. Примером задач такого типа являются системы, описывающие химические реакции в молекулярных смесях. Участки резкого изменения решения называют пограничными слоями. Для того, чтобы получить на них хорошую точность при сеточных расчетах, приходится брать очень малый шаг интегрирования, в то время как в промежутках между скачками можно было бы ограничиться и более крупным шагом. Если же выбранная разностная сетка слишком груба, численное решение на ней может сильно отличаться от точного. Далее мы обсудим известную методику получения оценки точности численного решения, особенности ее применения для задач с пограничными слоями, а также метод перехода к длине дуги интегральной кривой, помогающий лучше разрешать пограничные слои.

2.1. Метод Ричардсона для жестких задач.

Важнейшим способом получить даже не просто гарантированную (мажорантную), а асимптотически точную оценку погрешности является метод Ричардсона с глобальным сгущением сеток. Опишем его применение, следуя [7].

2.1.1. Стандартная процедура.

Пусть исходная задача (1) решается на отрезке $0 < t \leq T$ разностной схемой порядка аппроксимации p. Введем на $[0; T]$ равномерную или квазиравномерную сетку Ω_N с числом узлов N. Выполним на ней расчет и найдем сеточное решение $u(t; N)$, где $t \in \Omega_N$.

Затем сгустим сетку вдвое и на ней также найдем численное решение $u(t; 2N)$. Для равномерных и квазиравномерных сеток при удвоении числа узлов выполняется следующее правило: все узлы редкой сетки являются четными узлами более подробной. Поэтому можно сопоставлять значения $u(t; N)$ и $u(t; 2N)$ при одинаковых значениях t. По ним можно находить асимптотически точную оценку погрешности на сетке Ω_{2N}

$$\Delta(t; 2N) = [u(t; 2N) - u(t; N)]/(2^p - 1), \tag{2.1}$$

а также проводить экстраполяционное уточнение решения на сетке Ω_{2N}:

$$\tilde{u}(t;2N) = u(t;2N) + \Delta(t;2N). \tag{2.2}$$

Разумеется, погрешность и уточнение находятся только в четных узлах сетки Ω_{2N}.

Процедуру удвоения сетки повторяют многократно. Каждый раз по двум соседним сеткам производят вычисление оценки погрешности и экстраполяционное уточнение; оценка и экстраполяция приписываются более подробной сетке из пары.

Простейшим способом контроля сходимости является вывод всех сеточных решений $u(t;N)$, $u(t;2N)$, $u(t;4N)$, ... на один график. Если мы видим визуальное схождение сеточных кривых к некоторой предельной кривой, то на основании фундаментальных теорем из [42] естественно считать эту предельную кривую решением искомой задачи. Обычно визуальная сходимость сеточных решений к предельной функции наблюдается уже на сравнительно грубых сетках.

Для более тщательной проверки следует вычислить нормы погрешностей на каждой сетке. Например, можно принять

$$\begin{aligned} &||\Delta_{2N}||_c = \max_n(|\Delta(x_n;2N)|), \\ &||\Delta_{2N}||_{l_2} = \left[\frac{1}{N}\sum_n \Delta^2(x_n;2N)\right]^{1/2}; \end{aligned} \tag{2.3}$$

взятие максимума и суммирование производятся только в четных узлах сетки Ω_{2N}, так как в нечетных узлах погрешность не определена. Видно, что вторая норма по существу является среднеквадратичной погрешностью, приходящейся на один узел сетки. Аналогично вычисляются нормы погрешностей на более подробных сетках.

Затем строится график зависимости $\lg||\Delta_N||$ от $\lg N$. Если зависимость погрешности от шага сетки (числа узлов) носит степенной характер, то этот график является прямой линией. Поэтому если наблюдаемый график с хорошей точностью можно считать прямой линией с наклоном $\operatorname{tg}\alpha = p$, то фактический расчет сходится с порядком точности p, и применение метода Ричардсона правомерно.

Обычно на грубых сетках этот график искривлен, но при достаточном сгущении сеток выходит на прямую линию с требуемым наклоном.

2.1.2. Особенности жестких задач.

Стандартная процедура хорошо применима, если на последних сетках расчета каждая характерная деталь точного решения содержит разумное число узлов сетки. Для “мягких” задач этому условию нетрудно удовлетворить. Рассмотрим, что может произойти для жестких задач.

Характерное поведение точного решения жесткой задачи изображено на рис.2.1. В качестве примера здесь взята компонента $u(t)$ решения задачи ван дер Пола (4.17) при $\sigma = 100$ (решение этой задачи подробно разбирается в главе (4.3)). Оно содержит следующие участки: регулярное решение (участки медленного убывания и возрастания решения), пограничные слои (резкие скачки решения) и переходные зоны между ними. В пограничном слое решение меняется очень быстро, а ширина пограничного слоя (промежуток времени) может быть очень малой. В

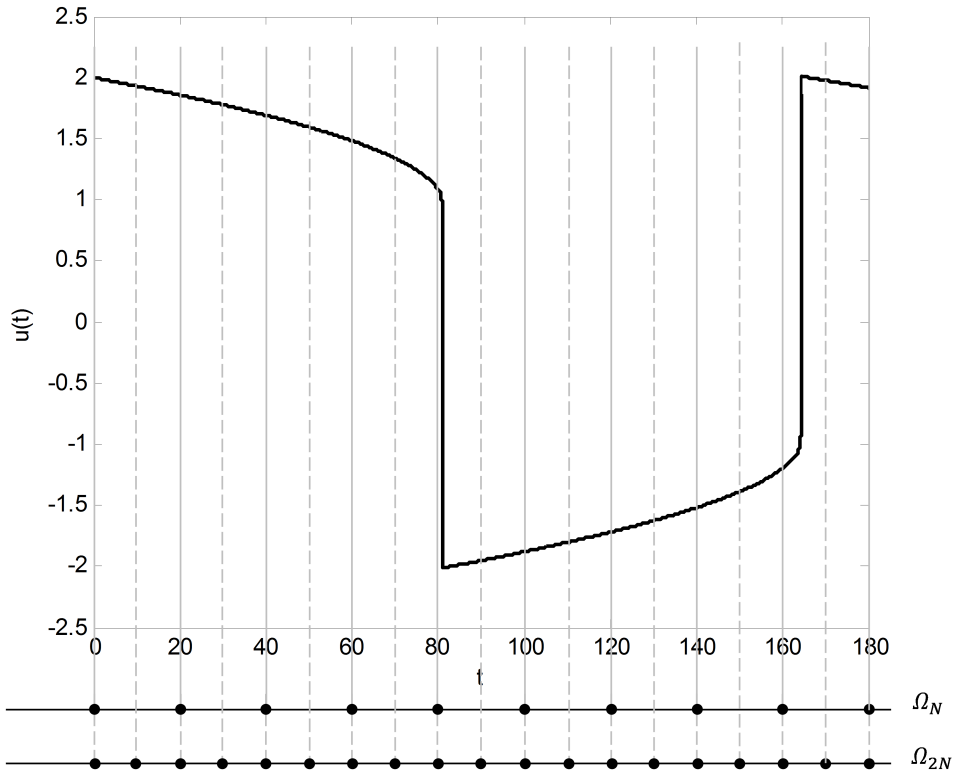

Рис. 2.1: Глобальное сгущение сеток для жесткой задачи (4.17).

регулярной области решение меняется довольно плавно. Между регулярным решением и пограничным слоем лежит переходная зона, ширина которой может в несколько раз превышать толщину пограничного слоя; но она также много меньше регулярной области. Пограничных слоев (и, соответственно, переходных зон) может быть несколько.

На рис.2.1 ниже оси абсцисс показаны последовательно сгущающиеся сетки. Очевидно, ни один узел начальной грубой сетки не попадает в пограничный слой и переходную зону. Если задача очень жесткая, то туда не попадет ни один узел даже весьма подробной сетки. Все узлы сгущающихся сеток будут лежать в регулярной области. При этом визуально будет наблюдаться схождение сеточных решений к предельной функции, а график логарифмической погрешности хорошо выйдет на прямую линию с правильным наклоном (начальная часть графика на рис.2.2). Такие результаты правильно описывают не все точное решение, а только его регулярную часть.

Для того, чтобы описать пограничный слой, продолжим сгущать сетки, воспользовавшись при необходимости повышенной разрядностью вычислений. С некоторого сгущения узлы сетки начнут попадать в переходную область. Визуальная картина сходимости сеточного решения останется той же: с графической точностью все линии будут сливаться. Однако на графике $\log ||\Delta||(N)$ погрешность может начать возрастать. Это возрастание связано с повышенной локальной погрешностью в переходной зоне и пограничном слое (если отдельно выделить погрешность регулярной части, то она будет продолжать убывать).

Продолжим сгущение сеток. Когда в пограничном слое окажется достаточно много точек, норма погрешности снова начнет убывать, и ее график снова выйдет на прямую линию с правильным наклоном. На такой сетке разностное решение хорошо опишет все участки точного

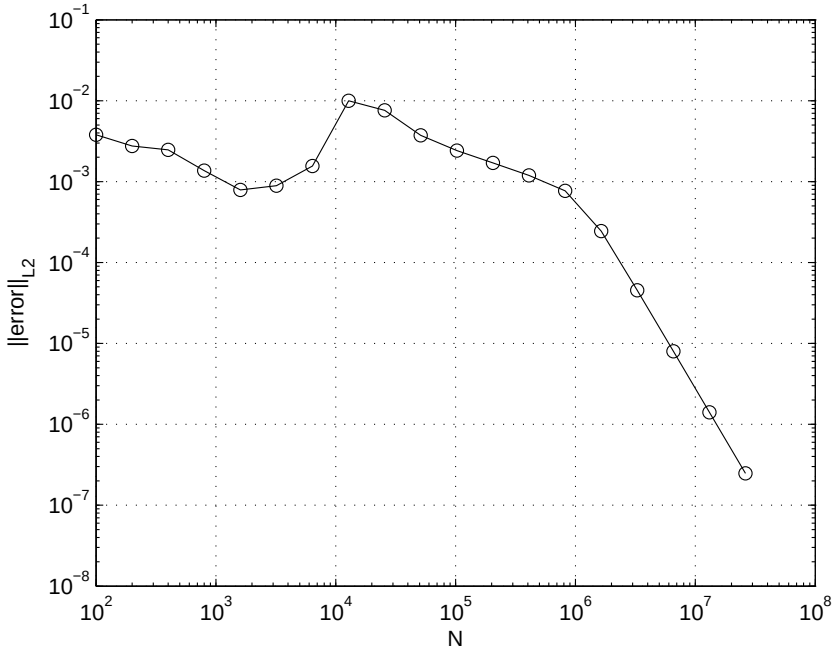

Рис. 2.2: Погрешность при решении задачи (4.17), вычисленная по методу Ричардсона на серии сгущающихся сеток.

решения. Однако на задачах огромной жесткости для этого нужны очень подробные сетки.

2.1.3. Разрешение пограничного слоя.

Иногда удается заранее провести аналитическое исследование задачи в пограничном слое. Тогда можно построить такую квазиравномерную сетку, которая содержит очень малые шаги в пограничном слое и большие шаги в регулярной области при умеренном общем числе узлов. На таких сетках можно хорошо расчитывать жесткие задачи при скромном объеме вычислений. Однако это требует непростых аналитических исследований для каждой отдельной задачи. Кроме того, такие исследования удается провести лишь для начального пограничного слоя, но не для внутренних пограничных слоев.

Далее мы рассмотрим, как можно хорошо разрешить пограничный слой, переходя к новому аргументу – длине дуги интегральной кривой.

2.2. Метод перехода к длине дуги.

Метод введения нового аргумента интегрирования – длины дуги интегральной кривой в $M+1$-мерном пространстве $\{u_0 \equiv t, u_1, ..., u_M\}$ – впервые был предложен в работе [43]. Его с 1993 года детально развили В.И. Шалашилин и Е.Б. Кузнецов в цикле работ, включая монографию [44]. В частности, была доказана теорема о том, что введение длины дуги обеспечивает наилучшую обусловленность задачи Коши. Зарубежных публикаций на эту тему почти нет (см. [45]).

Однако во всех этих работах не указано, как находить гарантированную оценку погрешности полученного решения. Современные программы с автоматическим выбором шага не позволяют этого сделать. Далее будет показано, как применяется сгущение сеток по методу Ричардсона при переходе к длине дуги. Расчеты с гарантированными оценками погрешности позволили установить, что переход к длине дуги дает тем больший выигрыш в точности, чем труднее (хуже обусловленнее или жестче) исходная задача (1). Выигрыш в точности может достигать многих порядков.

2.2.1. Уравнения.

Длина дуги определяется соотношением

$$(dl)^2 = \sum_{m=0}^{M} (du_m)^2, \quad du_0 \equiv dt. \tag{2.4}$$

Принимая l за новый аргумент, получим вместо (1) следующую систему:

$$\frac{du_m}{dl} = F_m(\mathbf{u}) \equiv \frac{f_m}{\sqrt{\sum_{m=0}^{M} f_m^2}}, \quad 0 \le m \le M, \quad f_0 \equiv 1. \tag{2.5}$$

Правые части (2.5) $F_m(\mathbf{u})$ не зависят от нового аргумента l, поэтому система (2.5) является автономной. Выполняется соотношение $\sum_{m=0}^{M} F_m^2 = 1$. Поэтому правые части F_m невелики, и система (2.5) является мягкой даже в том случае, если система (1) была плохо обусловленной или жесткой.

Систему (1) надо интегрировать до момента T. Но до какого значения $l = L$ нужно интегрировать систему (2.5) – не известно. Поскольку $F_0 > 0$, то $u_0 \equiv t$ есть монотонно возрастающая функция l. Следовательно, надо интегрировать (2.5) до тех пор, пока не выполнится $u_0(l) \ge T$.

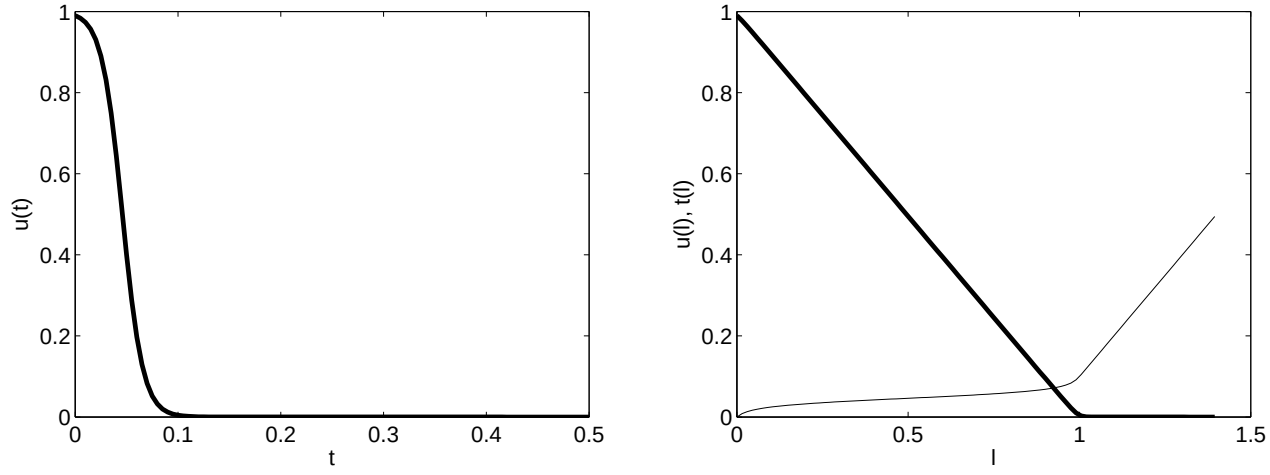

Рис. 2.3: а) - График $u(t)$; б) - Жирная линия - график $u(l)$, тонкая линия - график $t(l)$.

Проиллюстрируем идею перехода к длине дуги. Рассмотрим уравнение следующего вида:

$$du/dt = -\lambda u(1-u), \quad \lambda \gg 1, \quad 0 < u_0 < 1. \tag{2.6}$$

Для определенности возьмем $\lambda = 100$, $u_0 = 0.99$. График решения уравнения (2.6) приведен на рис.2.3.а: функция $u(t)$ имеет настолько резкое убывание, что мало отличается от разрывной. После перехода к длине дуги в соответствие с формулами (2.5)(см. рис.2.3.б) функция $u(l)$ вместо скачка имеет наклон 45°. Почти разрывная кривая $u(t)$ превращается в почти ломаную $u(l)$. Видно, что последнюю кривую гораздо легче численно интегрировать.

2.2.2. Сохранение балансов в задачах химической кинетики.

Модели химической кинетики образуют важный подкласс задач, сводящихся к жестким системам обыкновенных дифференциальных уравнений. Часто стандартных требований к численным методам решения жестких ОДУ, перечисленным во введении к данной работе, оказывается недостаточно для получения качественно верного результата. Численные методы, пригодные для решения задач химической кинетики, должны по возможности удовлетворять некоторым специфичным требованиям. Например, концентрации веществ, участвующих в химических реакциях, не могут принимать отрицательные значения. Исходя из этого требования, был построен ряд специализированных схем (например, [46]).

Другое требование, специфичное для задач химической кинетики, связано с законом сохранения частиц. В химических реакциях образуются и разлагаются молекулы, но суммарное число атомов каждого химического элемента остается при этом неизменным. Для хорошего качественного поведения численного решения необходимо, чтобы в численном расчете также сохранялось число атомов каждого сорта. Такие соотношения балансов являются первыми интегралами системы химических уравнений.

Каждое уравнение баланса можно записать следующим образом. Пусть каждая молекула u_j содержит α_j атомов определенного элемента. Тогда полное число атомов в системе $\sum\limits_j \alpha_j u_j = \mathrm{const}$ не зависит от времени. Легко видеть, что при этом $\sum\limits_j \alpha_j f_j = 0$: сколько атомов выходит из одних молекул, столько же попадает в другие. Эти соотношения удобно записать в матричном виде. Пусть из компонент α_j составлен вектор-столбец $\boldsymbol{\alpha}$; его можно рассматривать как прямоугольную матрицу. Тогда точные решения удовлетворяют следующим балансным соотношениям:

$$\boldsymbol{\alpha}^T \mathbf{f} = 0, \quad \boldsymbol{\alpha}^T \mathbf{u} = \mathrm{const}, \tag{2.7}$$

где умножение строки $\boldsymbol{\alpha}^T$ на столбец $\mathbf{f}$ или $\mathbf{u}$ выполняется по правилам умножения матриц. Различных столбцов $\boldsymbol{\alpha}$ существует столько, сколько различных химических элементов входит в реакции; исходная система должна удовлетворять всем этим балансам. Покажем, что схемы, использующиеся при расчетах в данной работе, сохраняют химические балансы системы.

Теорема 2.1. *Схемы Рунге-Кутты сохраняют химический баланс системы.*

Доказательство. Как известно, в общем виде формулы для s-стадийных методов Рунге-Кутты записываются следующим образом:

$$\hat{\mathbf{u}} - \mathbf{u} = \tau \sum_{k=1}^{s} b_k \mathbf{w}_k, \quad \mathbf{w}_k = \mathbf{f}(\mathbf{u} + \tau \sum_{l=1}^{L} a_{kl} \mathbf{w}_l, t + \tau a_k). \tag{2.8}$$

Умножим первое выражение слева на $\boldsymbol{\alpha}^T$:

$$\boldsymbol{\alpha}^T\hat{\mathbf{u}} - \boldsymbol{\alpha}^T\mathbf{u} = \tau\sum_{k=1}^{s} b_k\boldsymbol{\alpha}^T\mathbf{w}_k. \tag{2.9}$$

Заметим, что выражение для $\mathbf{w}_k$ представляет собой $\mathbf{f}$ от сдвинутого аргумента (см. второе выражение в (2.8)), причем этот сдвиг на каждой стадии фиксированный. Выражение (2.7) должно выполняться для любого аргумента $\mathbf{f}$; следовательно $\boldsymbol{\alpha}^T\mathbf{w}_k = 0$. Значит вся сумма в правой части выражения (2.9) обращается в нуль. Следовательно $\boldsymbol{\alpha}^T\hat{\mathbf{u}} = \boldsymbol{\alpha}^T\mathbf{u}$, т.е. химический баланс системы сохраняется. ■

Замечание. Доказательство справедливо и для явных, и для диагонально-неявных, и для полностью неявных схем Рунге-Кутты (в частности обратных [47]).

Теорема 2.2. *Одностадийные схемы Розенброка сохраняют химический баланс системы.*

Доказательство. В общем виде формулы для семейства одностадийных схем Розенброка имеют следующий вид:

$$\begin{aligned} &(E - a\tau\mathbf{f}_u)\,\mathbf{w} = \mathbf{f}(\mathbf{u}),\\ &\hat{\mathbf{u}} = \mathbf{u} + \tau b\mathbf{w}. \end{aligned} \tag{2.10}$$

Здесь E - единичная матрица, a и b - скалярные параметры схемы. Домножим первое уравнение из (2.10) слева на строку $\boldsymbol{\alpha}^T$:

$$\boldsymbol{\alpha}^T\left(E - a\tau\mathbf{f}_u\right)\mathbf{w} = \boldsymbol{\alpha}^T\mathbf{f} = 0. \tag{2.11}$$

Раскрывая скобки в левой части, получаем:

$$\boldsymbol{\alpha}^T\mathbf{w} - a\tau\boldsymbol{\alpha}^T\mathbf{f}_u\mathbf{w} = 0. \tag{2.12}$$

Из условия $\boldsymbol{\alpha}^T\mathbf{f} = 0$ следует, что $\boldsymbol{\alpha}^T\mathbf{f}_u = 0$. Значит

$$\boldsymbol{\alpha}^T\mathbf{w} = 0. \tag{2.13}$$

Теперь домножим второе уравнение из (2.10) слева на строку $\boldsymbol{\alpha}^T$:

$$\boldsymbol{\alpha}^T\hat{\mathbf{u}} = \boldsymbol{\alpha}^T\mathbf{u} + \tau b\boldsymbol{\alpha}^T\mathbf{w}. \tag{2.14}$$

С учетом (2.13), получаем $\boldsymbol{\alpha}^T\hat{\mathbf{u}} = \boldsymbol{\alpha}^T\mathbf{u}$, т.е. химический баланс системы сохраняется. ■

Замечание. Доказательство справедливо при любом значении параметра a, в том числе для комплексного.

Обобщение. Теорема 2.2 обобщается на многостадийные схемы Розенброка, в том числе схемы с комплексными коэффициентами.

Таким образом, схемы Рунге-Кутты и Розенброка удовлетворяют соотношению балансов. Следовательно, они являются ***консервативными*** в том смысле, который придавал этому термину А.А. Самарский.

Помимо разностных схем, сохраняющих химический баланс системы, таким же свойством обладает исследуемое в данной работе преобразование перехода к длине дуги (2.5). Сформулируем это утверждение в виде теоремы.

Теорема 2.3. *Переход к длине дуги сохраняет химический баланс системы.*

Доказательство. Для доказательства теоремы нужно показать, что если для исходной задачи $\boldsymbol{\alpha}^T\mathbf{f} = 0$, то после перехода к длине дуги будет выполняться $\boldsymbol{\alpha}^T\mathbf{F} = 0$. Тогда схемы, сохраняющие баланс для исходной задачи Коши, будут сохранять химический баланс и после перехода к длине дуги. Действительно,

$$\boldsymbol{\alpha}^T\mathbf{F} = \boldsymbol{\alpha}^T\frac{\mathbf{f}}{\sqrt{\sum\limits_{m=0}^{M} f_m^2}}. \tag{2.15}$$

Поскольку корень в знаменателе дроби один и тот же для всех компонент, а $\boldsymbol{\alpha}^T\mathbf{f} = 0$, то и все выражение $\boldsymbol{\alpha}^T\mathbf{F} = 0$. ■

Глава 3

Модификации метода Ньютона.

Нахождение корней нелинейных уравнений или их систем является важным составляющим шагом множества задач. В частности, система нелинейных уравнений возникает в результате решения систем нелинейных ОДУ неявными схемами. Обычно системы уравнений большой размерности решают методом простых итераций или методом Ньютона. Последний представляется более предпочтительным, поскольку обладает квадратичной сходимостью в малой окрестности простого корня в отличие от линейной сходимости простых итераций. Однако и с применением метода Ньютона связаны известные сложности: область сходимости метода может быть очень небольшой, и в случае неудачного выбора начального приближения первая же итерация может бросить решение далеко от точного значения корня, что сильно замедлит сходимость итераций.

3.1. Область сходимости.

Обсуждим возможность построения надежного алгоритма нахождения корней нелинейного уравнения, обладающего широкой областью сходимости. Перед этим рассмотрим несколько известных подходов к решению данной задачи.

3.1.1. Классический метод Ньютона.

В случае единственного уравнения с одним неизвестным классический метод Ньютона имеет следующий вид:

$$x_{s+1} = x_s - f(x_s)/f'(x_s). \tag{3.1}$$

Область сходимости метода Ньютона к какому-либо корню на комплексной плоскости, так называемая *зона притяжения*, образует в общем случае фрактальную структуру, то есть множество совершенно не связанных между собой точек и областей (например, см. [48]). На рис. 3.1 показана зона притяжения корня $z = 1$ уравнения $z^3 - 1 = 0$. Два другие корня этого уравнения имеют аналогичные зоны притяжения. Вышесказанное означает, что выбирать начальное приближение для метода Ньютона даже в простейшем случае нужно с осторожностью.

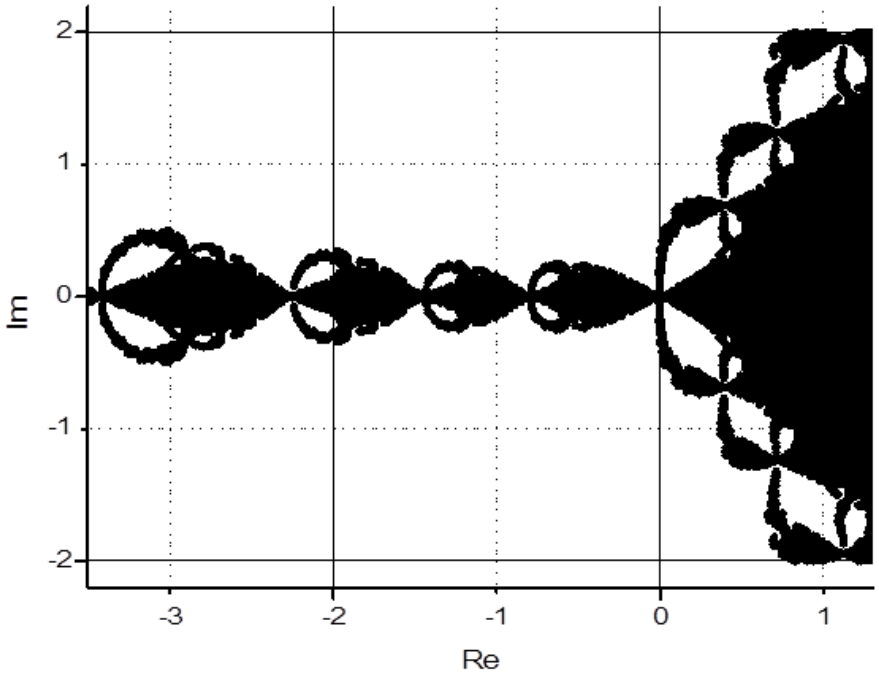

Рис. 3.1: Область притяжения корня $z = 1$ уравнения $z^3 - 1 = 0$ для метода Ньютона на комплексной плоскости.

3.1.2. Непрерывный аналог метода Ньютона.

Множество работ посвящено попыткам расширения области сходимости метода Ньютона. В работах [40, 49, 50] рассматриваются обощения метода Ньютона следующего вида:

$$x_{s+1} = x_s - \tau_s \varphi(x_s), \quad \varphi(x) \equiv f(x)/f'(x), \quad 0 < \tau_s \leq 1. \tag{3.2}$$

Значения τ_s выбирают так, чтобы вдали от корня они были небольшими, а при попадании в малую окрестность корня стремились бы к 1: в таком случае сохранится квадратичная сходимость метода вблизи корня. Оптимальный шаг вводится следующим образом:

$$\tau_s = \frac{f^2(x_s) + \theta f^2(x_s - \varphi(x_s))}{f^2(x_s) + f^2(x_s - \varphi(x_s))}, \quad 0 < \theta \leq 1. \tag{3.3}$$

Здесь θ – управляющий параметр метода. С ростом θ величина τ_s монотонно возрастает от значения

$$\tau_s = \frac{f^2(x_s)}{f^2(x_s) + f^2(x_s - \varphi(x_s))} \tag{3.4}$$

до $\tau_s = 1$. Очевидно, при $\theta = 1$ ($\tau_s = 1$) формулы (3.2)– (3.3) переходят в классический метод Ньютона (3.1).

Расчеты показывают, что предложенный подход действительно несколько расширяет область сходимости ньютоновских итераций, однако структура области сходимости остается сходной с той, что изображена на рис. 3.1 для классических итераций. В случае выбора не слишком удачного начального приближения количество итераций алгоритма до вхождения в малую

окрестность корня может составить несколько десятков. Это связано с тем, что вдали от корня используемый критерий слишком сильно уменьшает ньютоновский шаг, замедляя тем самым сходимость алгоритма. В то же время это приводит к существенному удорожанию используемой модификации метода Ньютона, так как на каждой итерации нужно заново вычислять матрицу Якоби.

3.1.3. Дифференциальный аналог метода Ньютона.

В работе [51] был предложен любопытный метод, получивший название *дифференциальный аналог метода Ньютона*. Суть этого метода состоит в следующем. Пусть $f(x)$ — функция одного переменного. Запишем формулу для обобщенного метода Ньютона в виде:

$$x_{s+1} = x_s - \tau(df/dx)^{-1} f. \tag{3.5}$$

При $\tau = 1$ имеем классический метод Ньютона. Если же $\tau < 1$, то с геометрической точки зрения мы производим не полный спуск по касательной, а лишь его часть, что позволяет уменьшить вероятность попадания в область, в которой классический метод Ньютона заведомо сойтись не может.

Заметим, что при малых значениях параметра τ эту формулу можно рассматривать как численную реализацию некоторого дифференциального уравнения простейшей схемой Эйлера. Само же дифференциальное уравнение (при переходе от дискретных значений τ к бесконечно малым величинам) запишется в виде:

$$\frac{df}{dx}\frac{dx}{dt} = -f(x(t)).$$

Это дифференциальное уравнение, вообще говоря, справедливое и в многомерном случае, имеет точное решение:

$$f(x(t)) = f(x(0))e^{-t}. \tag{3.6}$$

Получается, что при $t \to \infty$ предел решения $x(t)$ дает корень нелинейной системы вне зависимости от начального приближения!

Исследуем поведение описанного метода на двух примерах.

1. Пусть целевое уравнение - парабола, имеющая пару действительных корней:

   $$f(x) \equiv x^2 - a^2 = 0.$$

   Подставляя $f(x)$ в уравнение (3.6), получим:

   $$\begin{gathered}(x^2 - a^2) = (x_0^2 - a^2)e^{-t}, \\ x(t) = ((x_0^2 - a^2)e^{-t} + a^2)^{1/2} \xrightarrow[t\to\infty]{} \pm a. \end{gathered}$$

   Стремление к корню монотонное, переброса на другую сторону корня не происходит ни при каком начальном приближении. Метод сходится к правильному ответу.

2. Пусть теперь целевое уравнение представляет собой параболу, имеющую положительный минимум и пару комплексных корней:

   $$f(x) = x^2 + a^2 = 0.$$

Вновь подставляя $f(x)$ в уравнение (3.6), получим:

$$(x^2 + a^2) = (x_0^2 + a^2)e^{-t},$$
$$x(t) = \left((x_0^2 + a^2)e^{-t} - a^2\right)^{1/2}.$$

Дифференциальное решение из вещественного начального приближения x_0 спускается до минимума в точке $x = 0$ за конечное время $t = \ln(1 + x_0^2/a^2)$. Минимум в этом процессе является точкой бифуркации (см. рис. 3.2): в ней траектория спуска расщепляется на две траектории в комплексной плоскости, которые при $t \to \infty$ приводят к мнимым корням $\pm ia$. Но итерационный процесс при вещественном начальном приближении не может уйти в комплексную плоскость. Он будет требовать все более мелкий шаг τ и никогда не дойдет до минимума. Если же ограничить τ снизу, счет разболтается.

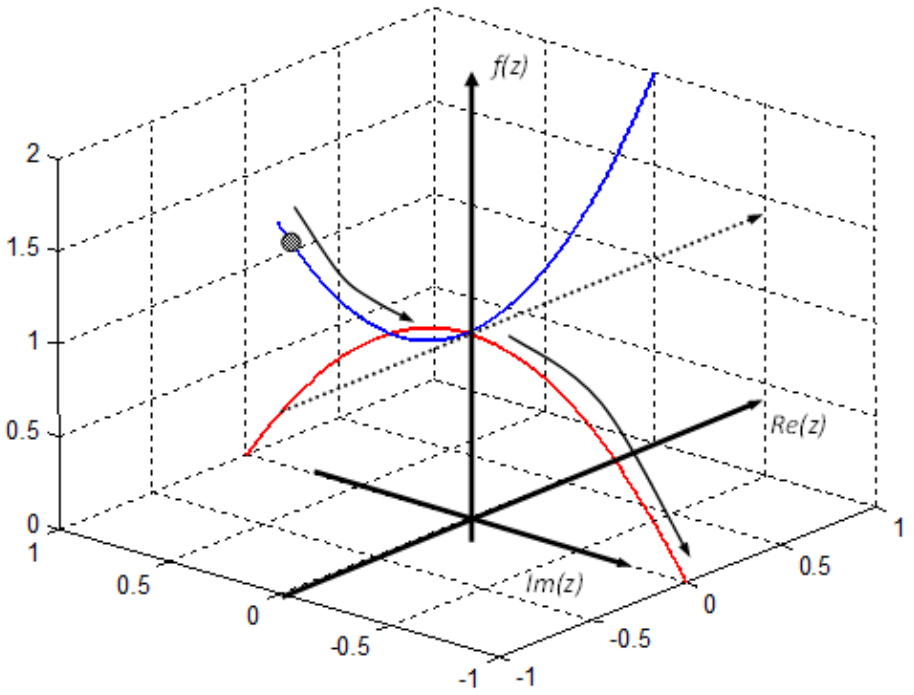

Рис. 3.2: Траектория спуска к комплексному корню дифференциального аналога метода Ньютона для уравнения $f(x) = x^2 + a^2 = 0$.

Таким образом, данный метод может не давать сходимости к корню из произвольного начального приближения, а вместо этого “застрянет” в локальном минимуме.

3.1.4. Сходимость конечношаговых итерационных методов.

В работах [52–54] предлагается также использовать гибридные подходы: при помощи более грубого метода (например, метод деления пополам или золотого сечения) находить малую окрестность корня, и затем с помощью более быстро сходящихся итераций (например, ньютоновских) находить значение корня с заданной точностью. Несмотря на то, что гибридные и обобщенные алгоритмы помогают расширить область сходимости классического метода Ньютона, глобальной сходимости к корню уравнения из произвольного начального приближения достичь, по-видимому, не удастся. Сформулируем это предположение в виде следующей теоремы.

Теорема 3.1. *Пусть заданы:*

1. *Одношаговый итерационный процесс* $x_{s+1} = \Phi(x_s, f(x_s), f'(x_s))$;
2. *Конечное число итераций* S;
3. *Нулевое приближение* x_0.

Тогда можно найти такой многочлен $f(x)$, *что за* S *итераций указанный процесс не сойдется к корню из выбранного нулевого приближения* x_0.

Доказательство. Возьмем два набора ненулевых вещественных чисел $a_s \neq 0$, $a'_s \neq 0$, $0 \leq s \leq S$. Начав с выбранного нулевого приближения x_0, вычислим последующие приближения по заданной итерационной процедуре и выбранным последовательностям: $x_{s+1} = \Phi(x_s, a_s, a'_s)$, $s = 0, 1, ...S-1$.

Построим такую функцию $f(x)$, которая вместе со своей производной на найденной последовательности x_s принимает следующие значения: $f(x_s) = a_s$, $f'(x_s) = a'_s$, $0 \leq s \leq S$. Например, в качестве такой функции можно взять многочлен $f(x) = c_{2S+1}x^{2S+1} + c_{2S}x^{2S} + ... + c_1x + c_0$, содержащий $2S+2$ свободных коэффициентов. Очевидно, для этой функции выбранная итерационная процедура Φ при начале из точки x_0 дает найденную выше последовательность x_s. Во всех точках этой последовательности $f(x_s) = a_s \neq 0$, включая последнюю точку x_S. Тем самым, для данной функции $f(x)$ итерационный процесс не сходится за заданное число S итераций. ■

Теорема естественно обобщается на многошаговые алгоритмы. Однако построенные аналогичным образом примеры многочленов $f(x)$ имеют многоэкстремальный вид и выглядят не очень естественно. Тем не менее, в наиболее общей формулировке теорема звучит следующим образом:

Теорема 3.2. *Пусть заданы:*

1. *Конечношаговый итерационный процесс, использующий конечное число* N *производных* $x_{s+1} = \Phi(x_s, f(x_s), f'(x_s), ..., f^{(N)}(x_s))$;
2. *Конечное число итераций* S;
3. *Нулевое приближение* x_0.

Тогда можно найти такой многочлен $f(x)$, *что за* S *итераций указанный процесс не сойдется к корню из выбранного нулевого приближения* x_0.

3.1.5. Выводы.

Поскольку невозможно построить итерационный процесс для решения нелинейных уравнений и систем, который бы сходился из любого начального приближения, необходимо по возможности уделять внимание выбору хорошего начального приближения. Например, при решении уравнений в частных производных сеточными методами удобно брать в качестве начального приближения на новом слое решение с предыдущего слоя. Впрочем, и такое начальное приближение может оказаться недостаточно хорошим (см. пример в п.4.1.5). В таком случае можно попытаться улучшить сходимость с помощью усеченных итераций (1.15) (аналогичный прием рекомендован в статье [54]).

3.2. Уравнения с кратными корнями.

Задача определения кратности корня и скорости сходимости итераций к нему является специфичной для одиночного уравнения: для систем уравнений понятие кратного корня не определено. Так же как и решение нелинейных систем, решение нелинейных уравнений является весьма широко распространенной проблемой в математике, представляющей интерес и как самостоятельная задача, и как вспомогательный инструмент при решении задач оптимизации, нахождения спектра матриц и других.

Возможность определения кратности найденного корня является важной характеристикой метода решения нелинейного уравнения, особенно если требуется найти все или несколько корней. При решении этой задачи мы сталкиваемся со следующими проблемами. Во-первых, корень высокой кратности часто не удается найти с хорошей точностью из-за маленького диапазона изменения значений $f(x)$ вблизи корня. Во-вторых, при нахождении каждого корня происходит потеря точности, связанная с ошибками округления: при численных расчетах мы находим не точное значение корня x_*, а приближенное $\tilde{x}_*$. Если пытаться исключать корни делением $f(x)$ на $x - x_*$, то последующие разы мы находим тот же самый кратный корень со всё меньшей точностью. Для корней высокой кратности итоговая погрешность может получиться значительной.

В настоящее время продолжают публиковаться работы, посвященные теоретическим исследованиям и новым методам решения нелинейных уравнений с кратными корнями [52], [53], а также методы нахождения всех корней полиномов [55], [56], [57].

3.2.1. Определение кратности корня.

Пусть у функции одного переменного $f(x)$ существует непрерывная $f^{(p+1)}(x)$, а x_* есть p-кратный корень. Тогда в малой окрестности корня

$$f(x) \approx a\Delta^p + b\Delta^{p+1}, \quad \Delta = x - x_*. \tag{3.7}$$

Подставляя (3.7) в формулу для метода Ньютона (3.1) и вычитая из обеих частей равенства x_*, получаем:

$$\Delta_{s+1} = \frac{p-1}{p}\Delta_s + O(\Delta_s^2). \tag{3.8}$$

Отсюда видно, что метод Ньютона для простого корня ($p = 1$) сходится квадратично, а для кратного корня ($p \geq 2$) – линейно.

По скорости сходимости можно определить кратность корня. Из (3.8) следует, что

$$\frac{p-1}{p} = \frac{\Delta_{s+1}}{\Delta_s} + O(\Delta_s). \tag{3.9}$$

Величины Δ_{s+1}, Δ_s нам неизвестны, поскольку в них входит неизвестное x_*. Однако нетрудно показать, что таким же, с точностью до малых величин, является отношение $(x_{s+1} - x_s)/(x_s - x_{s-1})$. Удобно ввести знаменатель линейной сходимости

$$q_s \equiv \frac{x_{s+1} - x_s}{x_s - x_{s-1}} \approx \frac{p-1}{p} \tag{3.10}$$

и величину

$$p_s = \frac{1}{1 - q_s}. \tag{3.11}$$

При сделанных предположениях $p_s \to p$ при $s \to \infty$. Поэтому нужно ввести в ньютоновские итерации расчет p_s по (3.10) и (3.11). Если итерации сходятся, а p_s стремится к целому числу p, то мы находим значени корня x_* и его кратность. Заметим, что данный метод применим даже к функциям, имеющим вблизи корня вид $f(x) \sim (x - x_*)^p$, где p - не целое. Сходимость значений p_s к нецелому числу служит указанием на такой корень дробной кратности.

3.2.2. Ускорение сходимости ньютоновских итераций.

Как было сказано выше, сходимость метода Ньютона для кратных корней оказывается линейной, то есть медленной. Для корней высокой кратности знаменатель сходимости настолько близок к 1, что для получения высокой точности могут потребоваться сотни итераций. В таких ситуациях стандартные программы обычно обрывают расчет, давая лишь 1-2 верных знака.

Известно [58], что модифицированный ньютоновский процесс вида

$$x_{s+1} = x_s - mf(x_s)/f'(x_s), \tag{3.12}$$

где m — кратность корня x_*, обладает квадратичной сходимостью вблизи искомого корня x_*. Однако этим приемом можно пользоваться для ускорения сходимости итераций только если кратность m корня известна заранее. В [59] быстрая сходимость процесса (3.12) используется для определения кратности корня: рекомендуется проводить расчеты с разными $m = 2, 3, 4 \ldots$, выбирая одно и то же нулевое приближение. То значение m, при котором число итераций до сходимости окажется наименьшим, должно быть кратностью найденного корня. Такой подход не очень надежен. Кроме того, он не позволяет определить кратность корня, если она является дробной.

В общем случае же для ускорения сходимости ньютоновских итераций, когда кратность корня заранее неизвестна, целесообразно применить следующий прием. Если значение знаменателя линейной сходимости q_s (3.10) установилось, то по трем значениям x_{s-1}, x_s, x_{s+1} можно просуммировать дальнейшую геометрическую прогрессию и найти экстраполяцию:

$$x_* \approx x_{s+1} + \frac{q_s}{1 - q_s}(x_{s+1} - x_s). \tag{3.13}$$

Это усовершенствование не дает выигрыша на простых корнях, но для кратных корней приводит не только к существенному ускорению сходимости, но и позволяет достичь более высокой точности. Похожие рекомендации по ускорению сходимости ньютоновских итераций для кратных корней даны в работе [57].

На рис.3.3 показаны кривые, характеризующие сходимость классических ньютоновских итераций (3.1) и итераций с применением экстраполяции (3.13) для уравнения

$$e^x - \sum_{k=0}^{p} \frac{x^k}{k!} = 0 \tag{3.14}$$

(при значении $p = 2$ имеем корень $x_* = 0$ кратности = 3) из начального приближения $x_0 = 3$ в зависимости от номера итерации. Кривая, соответствующая экстраполированному значению,

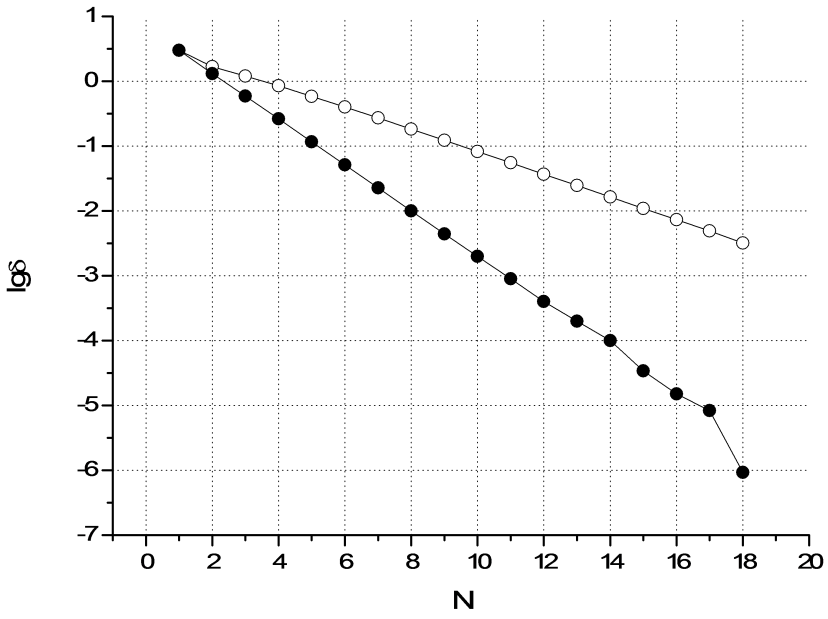

Рис. 3.3: Сходимость итераций классического метода Ньютона (кружки) и с применением экстраполяции (точки) для уравнения (3.14).

подходит к корню значительно быстрее. Кроме того, за 18 итераций экстраполированное значение превосходит по точности неэкстраполированное в ≈ 3000 раз. При экстраполяции возможен небольшой переброс на другую сторону корня, но для кратного корня это безопасно: сходимость обычного метода Ньютона к кратному корню монотонна с любой стороны.

Глава 4

Расчеты моделей прикладных задач.

В рамках данной работы с помощью разработанных численных методов и программ был выполнен расчет ряда моделей, отобраных из различных прикладных областей. Среди этих задач – расчет бегущей тепловой волны из [2], задача ван дер Пола и расчет транзисторного усилителя, которые включены в набор тестов для решателей жестких систем в [5], а также модель горения метана в воздухе, впервые предложенная в [60] и включенная в набор тестов [4]. Тестирование на достаточно широком наборе модельных задач позволяет выявить сильные и слабые стороны новых методов, а также сравнить их с уже существующими.

4.1. Бегущая тепловая волна.

Рассматривается квазилинейное уравнение теплопроводности на полупрямой $0 \le x < +\infty$ с заданными граничными и начальным условиями:

$$\begin{aligned} &\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(\varkappa(u)\frac{\partial u}{\partial x}\right), \quad \varkappa(u) = \varkappa_0 u^m, \\ &u(0,t) = (mc^2 t/\varkappa_0)^{1/m}, \quad u(+\infty,t) = 0, \quad u(x,0) = 0. \end{aligned} \tag{4.1}$$

Оно имеет частное непрерывное решение вида волны, бегущей со скоростью c:

$$u(x,t) = \begin{cases} \left[\dfrac{cm}{\varkappa_0}(ct-x)\right]^{1/m}, & x \le ct, \\ 0, & x > ct. \end{cases} \tag{4.2}$$

Вводя равномерную сетку $\{x_n\}$ с шагом h и обозначая $u_n = u(x_n)$, получим для (4.1) схему метода прямых точности $O(h^2)$:

$$\frac{du_n}{dt} = f_n(\mathbf{u}) \equiv \frac{1}{h^2}[\varkappa_{n+1/2}(u_{n+1}-u_n) - \varkappa_{n-1/2}(u_n - u_{n-1})]. \tag{4.3}$$

Это жесткая система нелинейных ОДУ. Чем больше m, тем большая надежность требуется от численного метода. Здесь в тестовых расчетах использовались $m = 5$, $c = 1$, $\varkappa_0 = 4$.

4.1.1. Аппроксимация коэффициента теплопроводности.

В системе (4.3) фигурируют значения коэффициента теплопроводности в серединах интервалов $\varkappa_{n\pm1/2}$. Для его вычисления пригодны две простейшие аппроксимации:

$$\varkappa_{n+1/2} = \frac{1}{2}\varkappa(u_{n+1}) + \frac{1}{2}\varkappa(u_n), \tag{4.4}$$

и

$$\varkappa_{n+1/2} = \varkappa\left(\frac{u_{n+1}+u_n}{2}\right). \tag{4.5}$$

В работе [61] обоснована лучшая точность аппроксимации (4.4), поэтому в дальнейшем мы будем использовать именно ее. С учетом (4.4), система (4.3) принимает следующий вид:

$$\frac{du_n}{dt} = \frac{1}{2h^2}[(\varkappa(u_{n+1}) + \varkappa(u_n))(u_{n+1}-u_n) - (\varkappa(u_n) + \varkappa(u_{n-1}))(u_n - u_{n-1})]. \tag{4.6}$$

4.1.2. Разностные схемы.

Для расчета задачи (4.6) будем использовать следующий набор схем.

1. Обратная схема Эйлера (OIRK1):
$$\frac{\hat{\mathbf{u}}-\mathbf{u}}{\tau} = \mathbf{f}(\hat{\mathbf{u}}). \tag{4.7}$$
Она входит в семейство оптимальных обратных схем Рунге-Кутты (1.3) при $s=1$. Схема $L1$-устойчива, имеет точность лишь $O(\tau)$, но обладает высокой надежностью.
2. Широкоиспользуемая для решения задач теплопроводности схема Crank-Nicolson (CN), также известная как схема “с полусуммой”:
$$\frac{\hat{\mathbf{u}}-\mathbf{u}}{\tau} = \frac{[\mathbf{f}(\hat{\mathbf{u}}) + \mathbf{f}(\mathbf{u})]}{2}. \tag{4.8}$$
Ее точность есть $O(\tau^2)$, но она лишь A-устойчива и немонотонна; L-устойчивости у нее нет [38].
3. Комплексная схема Розенброка (CROS) [62]:
$$\begin{gathered}\left(E - \frac{1+i}{2}\tau\mathbf{f}_u\right)\mathbf{w} = \mathbf{f}(\mathbf{u}),\\ \hat{\mathbf{u}} = \mathbf{u} + \tau\,\mathrm{Re}\,\mathbf{w}.\end{gathered} \tag{4.9}$$
Ее точность $O(\tau^2)$, и она $L2$-устойчива. Для перехода на новый слой схема CROS требует одного вычисления матрицы Якоби $\mathbf{f}_u$, одного вычисления правой части $\mathbf{f}$ и одного LU-разложения. Однако эта схема безытерационна, и поэтому не позволяет бегущей волне проходить более одного пространственного интервала за один шаг по времени [61].
4. Обратная схема (1.8) (ВМР). Как было показано выше, эта схема $L2$-устойчива и имеет точность $O(\tau^2)$.

Схемы порядка точности $p > 2$ не тестировались, поскольку пространственный оператор (4.6) имеет лишь 2-й порядок аппроксимации.

4.1.3. Сходимость к точному решению.

На рис.4.1 исследована сходимость неявных схем OIRK1, CN и BMP при одновременном сгущении сеток по x и t вдвое. Погрешность бралась в норме $L2$, причем только на гладком участке решения, не захватывающем фронт (итерации считались до сходимости). По наклону линий видно, что каждая из схем выходит на тот порядок точности, который определяется аппроксимацией по времени: первый для OIRK1, и второй для BMP и CN. Поэтому точность схемы OIRK1 оказывается много хуже, что согласуется с рис.4.1. Схема CN немного выигрывает у BMP, благодаря меньшему коэффициенту в остаточном члене, но это лишь на гладком участке решения.

Расчеты показали, что безытерационные явно-неявные разностные схемы, такие, как схема CROS, или неявные схемы, в которых итерации выполняются не до сходимости (число итераций ограничено), не дают сходимости к решению задачи (4.6): счет становится немонотонным и "разваливается". На рис.4.2 видно, что неявные схемы OIRK1 и CN, в которых выполняется только одна итерация, продвигаются с той же скоростью, что и безытерационная схема CROS, и все сильнее отстают от точного решения.

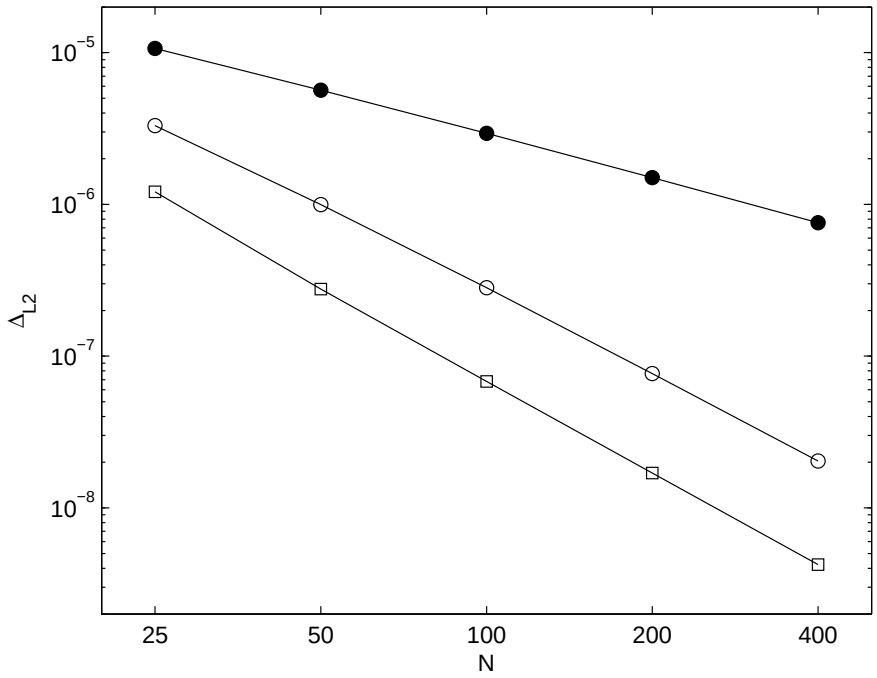

Рис. 4.1: Погрешности на гладком участке бегущей волны в норме $L2$; масштабы по осям логарифмические; $\tau/h = 1/2$; ○ – BMP, ● – OIRK1, □ – CN.

Надежный результат дает только расчет с итерациями, выполненными до сходимости. На рис.4.3 представлены результаты расчетов задачи (4.6) при соотношении шагов $\tau/h = 2$ без ограничения количества итераций для схем BMP, OIRK1 и CN.

Более подробно результаты расчетов с ограниченным числом итераций для схемы CN представлены на рис.4.4. Как было упомянуто выше, счет с одной итерацией сильно отстает от точного решения. При ограничении в 2-5 итераций счет разваливается. Ограничение в 6

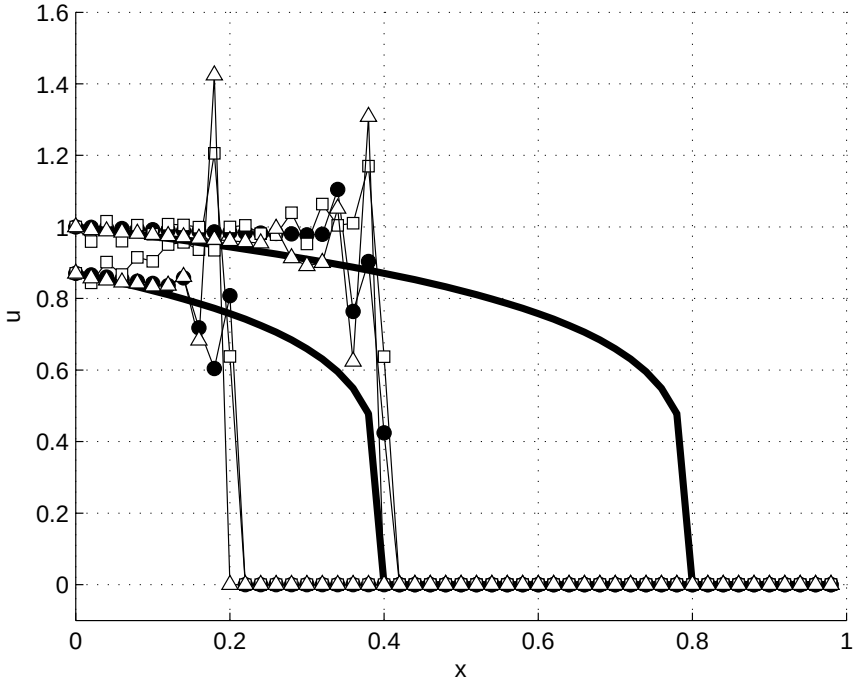

Рис. 4.2: Профили бегущей волны для двух моментов времени $t_1 = 0.4$ и $t_2 = 0.8$. Жирные линии - точное решение. Маркерами показаны расчеты для $\tau/h = 2$: $\triangle$ – CROS, $\bullet$ – OIRK1 с 1 итерацией, $\square$ – CN с 1 итерацией.

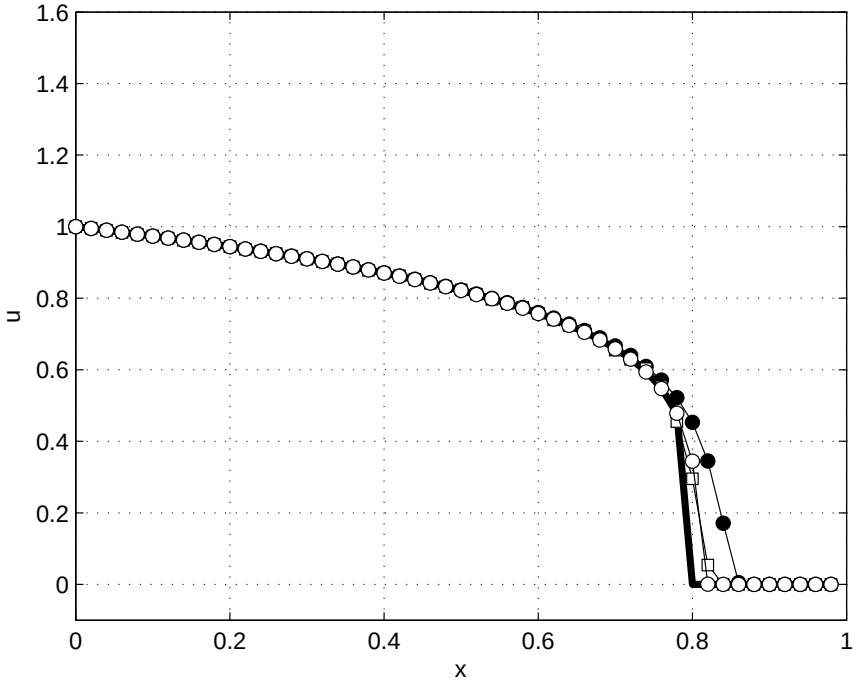

Рис. 4.3: Профиль бегущей волны для момента времени $t = 0.8$. Жирная линия – точное решение. Маркерами показаны расчеты для $\tau/h = 2$ (итерации до сходимости): $\circ$ – BMP, $\bullet$ – OIRK1, $\square$ – CN.

итераций дает ответ близкий к искомому решению, однако он является немонотонным. Заметим, что для отношения $\tau/h = 4$ шести итераций недостаточно для получения гладкого решения, хотя численное решение и не отстает так сильно от точного.

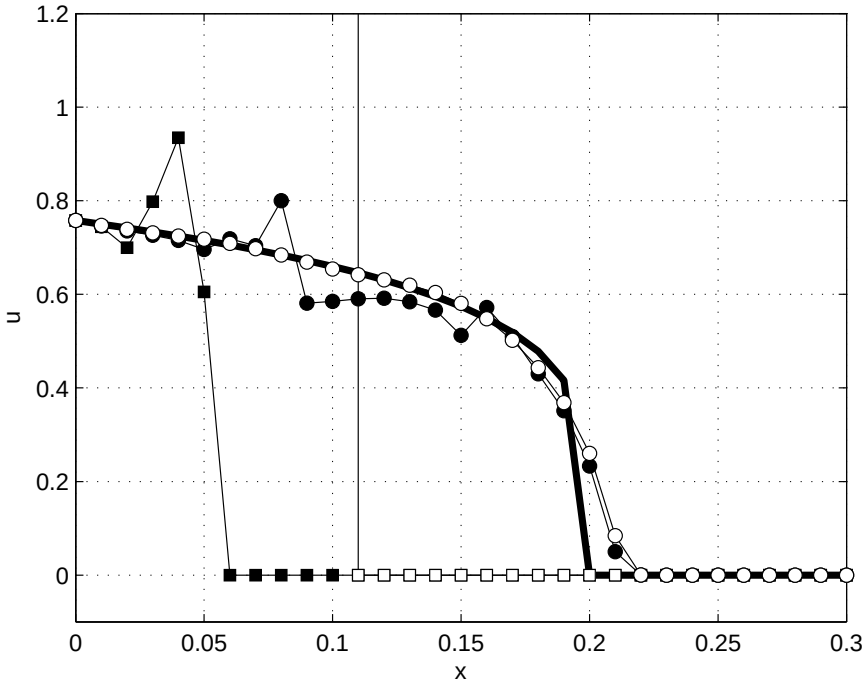

Рис. 4.4: Профиль бегущей волны для момента времени $t = 0.2$. Жирная линия - точное решение. Маркерами показаны расчеты для $\tau/h = 4$: ■ – CN с 1 итерацией, □ – CN с 3 итерациями, • – CN с 6 итерациями, ∘ – CN с итерациями до сходимости.

Приведенные результаты иллюстрируют справедливость следующей теоремы.

Теорема 4.1. *Пусть для уравнения (4.6) составлена любая разностная схема и фиксированы число итераций и отношение τ/h. Всегда найдется такое точное решение, сходимости к которому не будет при $\tau, h \to 0$.*

Доказательство. Поясним, как доказывается эта теорема. Достаточно доказать отсутствие сходимости хотя бы на одном частном случае. В качестве такого случая выберем решение вида тепловой волны, бегущей по нулевому фону (4.2).

Пусть на некотором слое справа перед фронтом волны значение температуры $u_n = u_{n+1} = u_{n+2} = ... = 0$. Тогда из-за степенной зависимости коэффициента теплопроводности от температуры $\varkappa(u) = \varkappa_0(u^m)$ (4.1) значение коэффициента теплопроводности в полуцелых узлах $\varkappa_{n+1/2} = \varkappa_{n+3/2} = \varkappa_{n+5/2} = ... = 0$. Значит при решении системы (4.6) с помощью схемы CROS в правой части первого уравнения в (4.9) $f_{n+1} = f_{n+2} = ... = 0$, а матрица в левой части будет иметь трехдиагональный вид для индексов $1...n$ и диагональный, начиная с индекса $n+1$. Значит на новом слое только значение u_n может получиться отличным от нуля. Значит за один шаг τ фронт волны не может пройти более одного пространственного интервала.

Аналогичную линейную систему необходимо решать на каждой итерации полностью неявных схем. Значит за одну итерацию фронт волны не может пройти более одного пространственного интервала. Если взято K итераций, то за один шаг τ фронт продвигается на не более чем

K интервалов. Значит скорость движения численного фронта не превышает величины Kh/τ. Поскольку по условиям теоремы отношение h/τ фиксировано, эта предельная скорость также будет фиксирована. Выберем такую скорость тепловой волны c в граничном условии в (4.1), чтобы выполнялось $c > Kh/\tau$. Тогда численный фронт будет двигаться в определенное число раз медленнее точного фронта, так что на данном тесте разностное решение не может сходиться к точному при $\tau, h \to 0$.

Очевидно, для безытерационной (явно-неявной) схемы $K = 1$, и схема попадает под данную теорему. В неявных схемах без ограничения числа итераций следует полагать $K = \infty$, так что на них теорема не распространяется. Поэтому для обеспечения сходимости разностного решения к точному необходимо брать неявную схему с итерациями на новом слое, и выполнять итерации до тех пор, пока разностное решение неявной алгебраической системы не будет найдено с высокой точностью (желательно на уровне ошибок округления, хотя в практических расчетах допустима и меньшая точность).

Таким образом, для решения задачи (4.6) подходят только неявные схемы с итерациями до сходимости. ■

Замечание. В доказательстве теоремы 4.1 для неявных методов подразумевалось использование тривиального прогноза начального значения на новом временном слое: $\mathbf{u}^0(t+\tau) = \mathbf{u}(t)$. Однако использование нетривиального прогноза, т.е. полиномиальной экстраполяции $\tilde{\mathbf{u}}^0(t+\tau)$ по значениям $\mathbf{u}$ на предыдущих временных слоях, не позволяет достичь сходимости: справа от фронта волны значения температуры $\mathbf{u}$ были нулевыми на всех предыдущих временных слоях, и нетривиальный прогноз даст справа от фронта то же начальное значение, что и тривиальный.

4.1.4. Реализация итерационного процесса.

Как было отмечено выше, при реализации обратных схем выгодно использовать ньютоновские итерации. Метод требует вычисления матрицы Якоби. Чисто ньютоновские итерации (1.14) в реализации неявных схем для решения задачи (4.6) обладают квадратичной сходимостью. Однако даже небольшая ошибка при вычислении матрицы Якоби или правой части замедляет скорость сходимости итераций до линейной, из-за чего общее количество итераций многократно возрастает. В частности, неэффективными оказываются вычисления с фиксированной матрицей: экономия на вычислениях матрицы Якоби будет незначительной по сравнению с очень сильно возросшим числом итераций до сходимости.

4.1.5. Усеченные ньютоновские итерации.

При увеличении соотношения шагов τ/h счет с использованием ньютоновских итераций становится все менее устойчивым и требует все больше итераций до вхождения в малую окрестность решения и сходимости. Это связано с использованием профиля бегущей волны с предыдущего момента времени в качестве начального приближения для нового слоя. Чем больше соотношение τ/h, тем худшим оказывается такое начальное приближение. Расчеты показали, что в таких условиях усечение (1.15) помогает значительно сократить количество итераций до сходимости во всех тестируемых схемах с итерациями. Этот прием прост в реализации и дешев

при расчетах. Что наболее важно, его применение дает уверенность в том, что каждая сделанная итерация приближает нас к искомому решению.

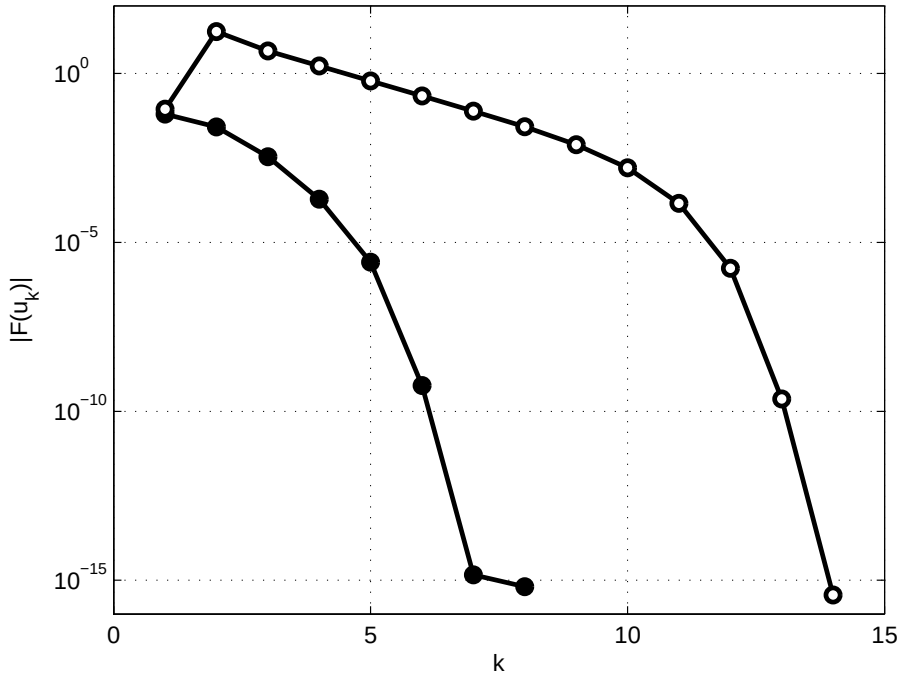

Рис. 4.5: Зависимость модуля правой части (1.14) от номера итерации. ∘ – классические ньютоновские итерации; • – усеченные итерации.

Результат применения усечения для схемы ВМР (1.8) при соотношении шагов $\tau/h = 1$ на одном из временных слоев приведен на рис.4.5. Монотонное убывание правой части $F(u_k)$ не дает первым итерациям уйти в сторону от искомого решения.

Для значений $\tau/h > 2$ итерации без усечения перестают сходиться, в то время как усеченные итерации дают решение за приемлемое число шагов. Выгода от применения усечения для всех исследованных неявных схем с итерациями возрастает с выбором менее близкого начального приближения (то есть большего соотношения τ/h). Кроме того, наибольший выигрыш в числе итераций был получен для схемы ВМР (1.8). Табл. 4.1 иллюстрирует зависимость среднего по всем временным слоям числа итераций от соотношения τ/h для каждой из рассмотренных схем. Счет по схеме (1.8) разваливается при $\tau/h > 1$. Применение усечения позволяет получить для данной схемы сходимость при этих значениях τ/h, причем число итераций при меньших значениях становится небольшим и сопоставимым с более простыми схемами OIRK1 и CN.

4.1.6. Разрывные начальные данные.

Проиллюстрируем надежность расчетов по схеме ВМР (1.8). В качестве примера возьмем квазилинейное уравнение теплопроводности вида (4.1) с разрывным начальным профилем:

$$u(x,0) = \begin{cases} 0, & x \in [0; 0.25) \cup (0.75; +\infty), \\ 1, & x \in [0.25; 0.75]. \end{cases} \tag{4.10}$$

Таблица 4.1: Среднее число классических и усеченных ньютоновских итераций до сходимости в зависимости от соотношения τ/h для неявных схем.

$\frac{\tau}{h}$	OIRK1		CN		BMP	
	Классич.	Усеч.	Классич.	Усеч.	Классич.	Усеч.
0.5	5.4	5.4	5.9	5.9	7.2	7.1
1	7.6	5.9	6.8	6.8	15.1	8.6
2	9.0	8.3	7.6	6.8	–	10.4
4	10.0	9.5	10.6	9.3	–	17.3
8	15.0	13.5	13.0	12.5	–	41.1

На рис.4.6 показаны профили численных решений на первом шаге по времени (итерации выполнялись до сходимости, $\tau/h = 4$). Высоконадежная схема OIRK1 дала гладкий, но чрезмерно размытый счетный профиль. Зато в схеме CN видна очень сильная немонотонность численного решения. Эта немонотонность на последующих шагах уменьшается, но еще долго не исчезает. Это показывает ненадежность схемы CN. Схема BMP и на этом тесте сохраняет точность и надежность.

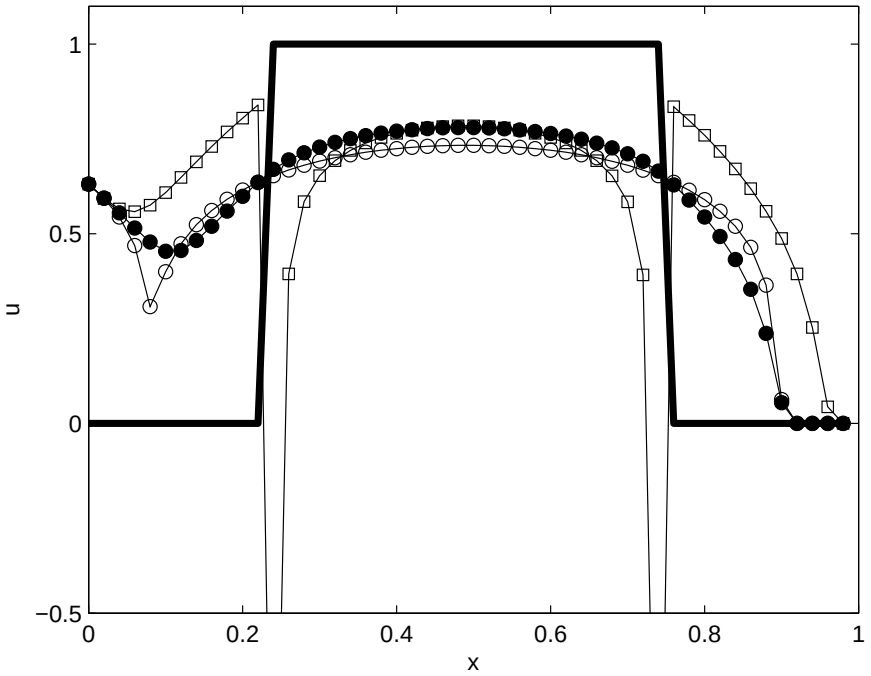

Рис. 4.6: Разрывные начальные данные (жирная линия) и профили на первом шаге при $\tau/h = 4$: ∘ – BMP, • – OIRK1, □ – CN.

4.2. Дифференциально-алгебраические системы.

В п.1.3.1 было отмечено, что оптимальные обратные схемы (1.3) являются жестко точными, а значит сохраняют порядок точности, равный числу стадий схемы при $s \leq 4$ на дифференциально-алгебраических задачах индекса 1. Проведем расчет простой тестовой задачи для проверки этого результата, а затем применим обратные схемы для решения задачи о транзисторном усилителе.

4.2.1. Тестовая задача.

В качестве простой тестовой задачи рассмотрим следующую систему с одной дифференциальной и одной алгебраической компонентами:

$$\begin{aligned} &\frac{dy}{dt} = -z, \\ &0 = y^2 + z^2 - 1, \\ &y(0) = 0, \quad z(0) = -1, \quad 0 \leq t \leq 1. \end{aligned} \tag{4.11}$$

При указанных начальных условиях для этой задачи легко найти точное решение:

$$y(t) = \sin(t), \quad z(t) = -\cos(t) \tag{4.12}$$

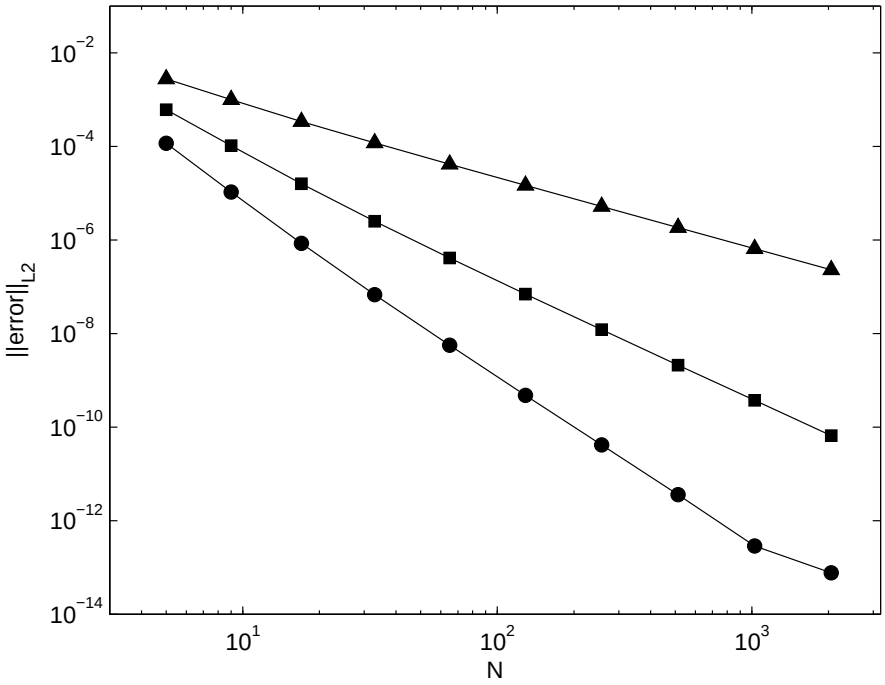

Рис. 4.7: Погрешность численного решения задачи (4.11). Маркерами указаны значения погрешностей для оптимальных обратных схем ▲ – 2го, ■ – 3го, ● – 4го порядка точности.

На рис. 4.7 изображены результаты расчетов с помощью оптимальных обратных схем (1.18) 2-го, 3-го и 4-го порядка точности на серии сгущающихся сеток. Как можно видеть, теоретический порядок точности достигается и на практике. На схеме 4-го порядка удалось выйти на

ошибки округления. Это позволило достичь очень высокой точности $\sim 10^{-15}$ (на схемах меньшего порядка выход на ошибки округления также происходит, но на гораздо более подробных сетках).

Таблица 4.2: Среднее число усеченных ньютоновских итераций при решении задачи (4.11) с помощью оптимальных обратных схем порядка p на серии сгущающихся сеток с числом узлов N.

p \ N	2	5	10	20	40	80	160	320	640	1280	2560
2	5.0	5.0	4.7	4.0	4.0	4.0	4.0	3.0	3.0	3.0	3.0
3	5.0	5.0	4.7	4.0	4.0	4.0	4.0	3.0	3.0	3.0	3.0
4	5.0	5.0	4.7	4.0	4.0	4.0	4.0	3.0	3.0	3.0	3.0

В табл. 4.2 приведены значения среднего числа усеченных ньютоновских итераций для каждой из рассмотренных сеток. Число итераций получилось небольшим (что неудивительно для такой простой задачи), причем с увеличением числа интервалов среднее число итераций уменьшается.

4.2.2. Транзисторный усилитель.

Расчет транзисторного усилителя, схема которого приведена на рис. 4.8, является одной из традиционных задач для тестирования разностных схем решения дифференциально-алгебраических систем [4, 5].

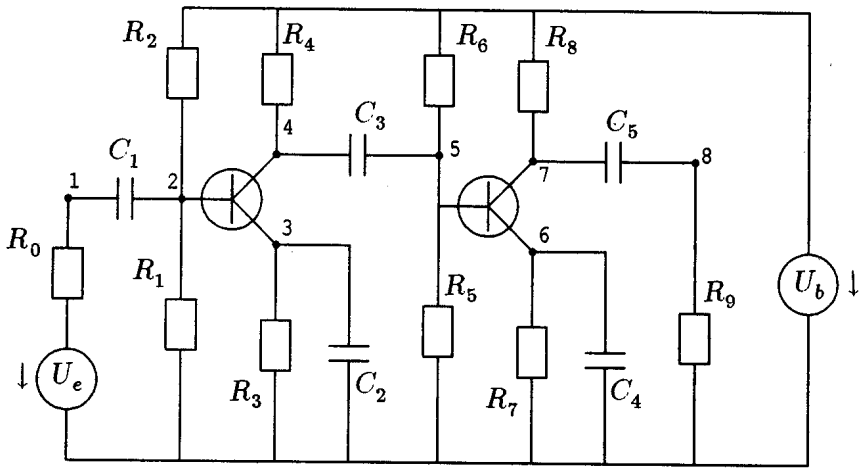

Рис. 4.8: Принципиальная схема транзисторного усилителя.

Опишем обозначения. U_e – входной периодический сигнал: $U_e(t) = 0.1\sin(200\pi t)$, U_8 – напряжение на выходе. Цепь включает в себя два транзистора. Предполагаем, что токи через транзисторы описываются следующими уравнениями:

$$I_G = (1-\alpha)g(U_G - U_S),$$
$$I_D = \alpha g(U_G - U_S),$$
$$I_S = g(U_G - U_S).$$

Здесь U_G и U_S – напряжения на затворе и источнике, значение параметра $\alpha = 0.99$. В качестве функции g возьмем

$$g(U_i - U_j) = \beta \left(e^{\frac{U_i - U_j}{U_F}} - 1 \right),$$

где $\beta = 10^{-6}$, $U_F = 0.026$. Применяя законы Кирхгофа к узлам 1-8, получим следующую дифференциально-алгебраическую систему относительно напряжений U_1,..., U_8, записанную в неявной форме:

$$\begin{array}{l}
\frac{d}{dt}(C_1(U_2 - U_1)) + \frac{U_e(t)}{R_0} - \frac{U_1}{R_0} = 0, \\
\frac{d}{dt}(C_1(U_1 - U_2)) + \frac{U_b}{R_2} - U_2(\frac{1}{R_1} + \frac{1}{R_2}) + (\alpha - 1)g(U_2 - U_3) = 0, \\
-\frac{d}{dt}(C_2 U_3) + g(U_2 - U_3) - \frac{U_3}{R_3} = 0, \\
-\frac{d}{dt}(C_3(U_4 - U_5)) + \frac{U_b}{R_4} - \frac{U_4}{R_4} - \alpha g(U_2 - U_3) = 0, \\
\frac{d}{dt}(C_3(U_4 - U_5)) + \frac{U_6}{R_6} - U_5(\frac{1}{R_5} + \frac{1}{R_6}) + (\alpha - 1)g(U_5 - U_6) = 0, \\
-\frac{d}{dt}(C_4 U_6) + g(U_5 - U_6) - \frac{U_6}{R_7} = 0, \\
-\frac{d}{dt}(C_5(U_7 - U_8)) + \frac{U_b}{R_8} - \frac{U_7}{R_8} - \alpha g(U_5 - U_6) = 0, \\
-\frac{d}{dt}(C_5(U_7 - U_8)) + \frac{U_8}{R_9} = 0.
\end{array} \tag{4.13}$$

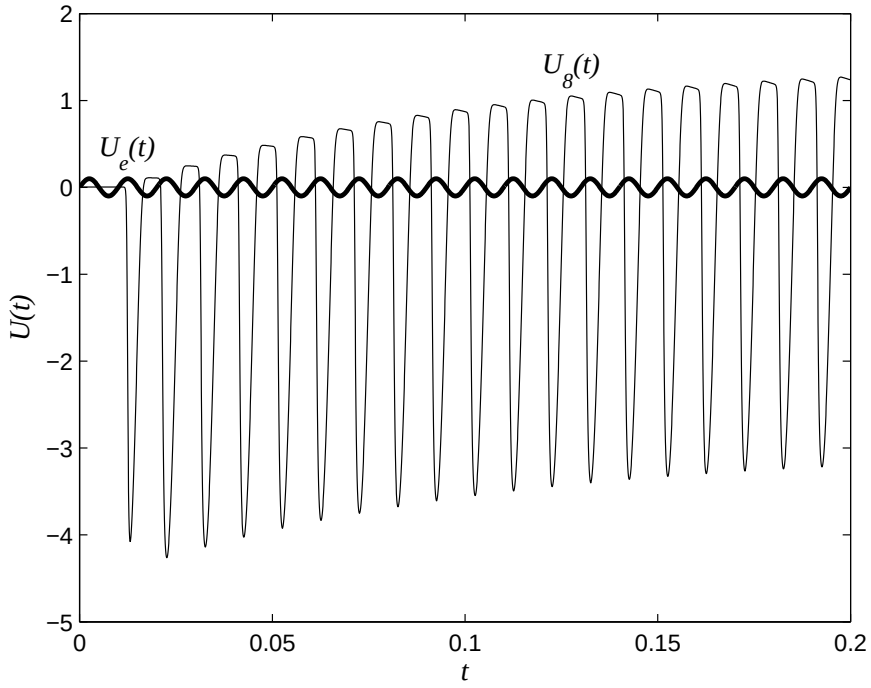

Рис. 4.9: Решение задачи (4.13). $U_e(t)$ – входное напряжение (жирная линия), $U_8(t)$ – напряжение на выходе усилителя.

На рис. 4.9 показаны графики для напряжений на входе и выходе усилителя. График похож на решение, приведенное в [4], однако его точность оценить не представляется возможным. Для получения оценки погрешности проведем расчет на последовательности сгущающихся сеток с помощью оптимальных обратных схем (1.18), одностадийной схемы Розенброка (4.9), имеющей второй порядок точности на дифференциально-алгебраических задачах [41], и следующей

двухстадийной схемы Розенброка с комплексными коэффициентами (CROS4):

$$\begin{aligned}
&(G - \tau\alpha_1 \mathbf{f}_u(\mathbf{u}))\,\mathbf{k}_1 = \mathbf{f}(\mathbf{u}),\\
&(G - \tau\alpha_2 \mathbf{f}_u\,(\mathbf{u} + \tau\,\mathrm{Re}\,a_{21}\mathbf{k}_1))\,\mathbf{k}_2 = \mathbf{f}\,(\mathbf{u} + \tau\,\mathrm{Re}\,c_{21}k_1)),\\
&\hat{\mathbf{u}} = \mathbf{u} + \tau\,\mathrm{Re}\,(b_1\mathbf{k}_1 + b_2\mathbf{k}_2),
\end{aligned} \tag{4.14}$$

где коэффициенты схемы α_1, α_2, a_{21}, c_{21}, b_1, b_2 имеют следующие значения:

$$\begin{aligned}
\alpha_1 &= 0.1 + i\sqrt{11}/30,\\
\alpha_2 &= 0.2 + i0.1,\\
c_{21} &= 0.2554708972958462 - i0.2026195833570109,\\
a_{21} &= 0.5617645150714754 - i1.148223341045841,\\
b_1 &= 0.1941430241155180 - i0.2246898944678803,\\
b_2 &= 0.8058569758844820 - i0.8870089521907592.
\end{aligned} \tag{4.15}$$

Эта схема L2-устойчива и имеет 4-й порядок точности для чисто дифференциальных задач (то есть при $G \equiv E$, где E – единичная матрица) и 3-й для дифференциально-алгебраических [11]. Для перехода на новый слой схема CROS4 требует двух вычислений матрицы Якоби, двух вычислений правой части и двух LU-разложений.

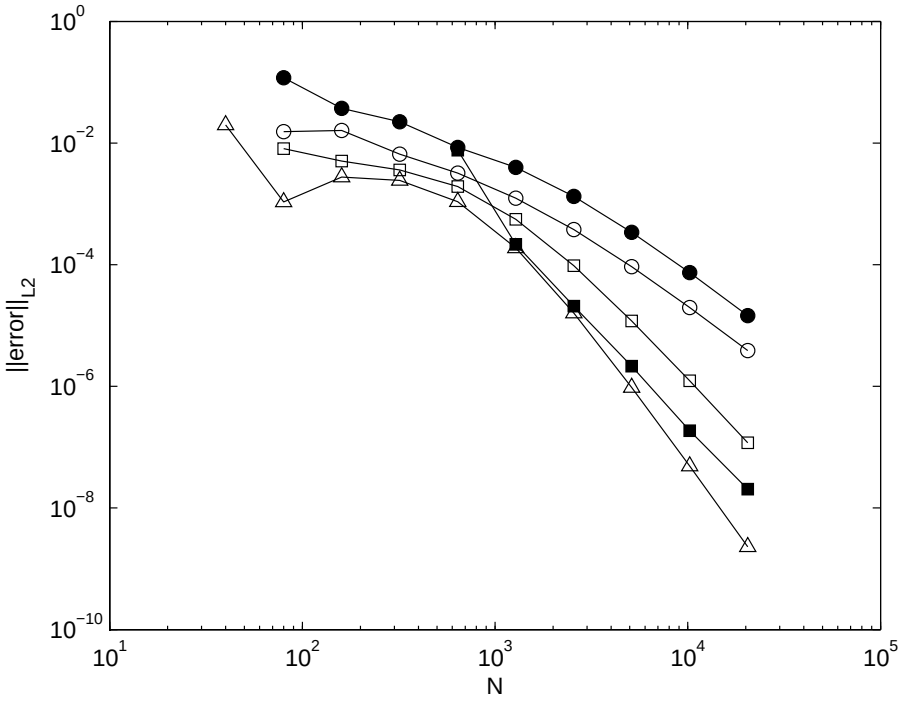

Рис. 4.10: Погрешность численного решения задачи (4.13). Схемы: ● – CROS, ○ – OIRK2, ■ – CROS4, □ – OIRK3, △ – OIRK4.

На рис. 4.10 приведены графики абсолютной погрешности решения в норме $L2$ в зависимости от числа узлов сетки для всех упомянутых схем. Видно, что для схем CROS и OIRK2 достигается второй порядок точности, и схема OIRK2 дает небольшой выигрыш в числе узлов. Схема Розенброка CROS4 дает разумное численное решение, близкое к точному, лишь на более мелкой сетке по сравнению с OIRK3 и другими схемами. Однако при дальнейшем сгущении она позволяет получить более точное решение.

Единственная схема 4-го порядка точности OIRK4 выходит на теоретический порядок точности лишь на последних четырех сетках. При этом, как было отмечено выше, время расчета по обратным схемам быстро увеличивается с ростом стадийности схемы и размерности задачи. При решении задачи о транзисторном усилителе время расчета по явно-неявным схемам Розенброка было в 10–50 раз меньше времени расчета по оптимальным обратным схемам Рунге-Кутты аналогичного порядка точности.

4.3. Уравнение ван дер Пола.

Рассмотрим еще одну задачу, которая получается при моделировании электрической цепи с нелинейным элементом. Электрическая схема, впервые составленная Бальтазаром ван дер Полом, приведена на рис. 4.11 [4]. Она представляет собой RLC-контур, в котором пассивный резистор заменен на активный элемент, усиливающий ток в цепи, если его амплитуда падает ниже определенного уровня. Цепь описывается следующим обыкновенным дифференциальным уравнением второго порядка:

$$z'' + \sigma(z^2 - 1)z' + z = 0, \tag{4.16}$$

где $z \sim I(t)$, а значение σ зависит от параметров элементов цепи. Обозначив $z \equiv u$, $z' \equiv v$, получим систему обыкновенных дифференциальных уравнений первого порядка:

$$\begin{aligned} du/dt &= v, \\ dv/dt &= -u - \sigma(u^2 - 1)v. \end{aligned} \tag{4.17}$$

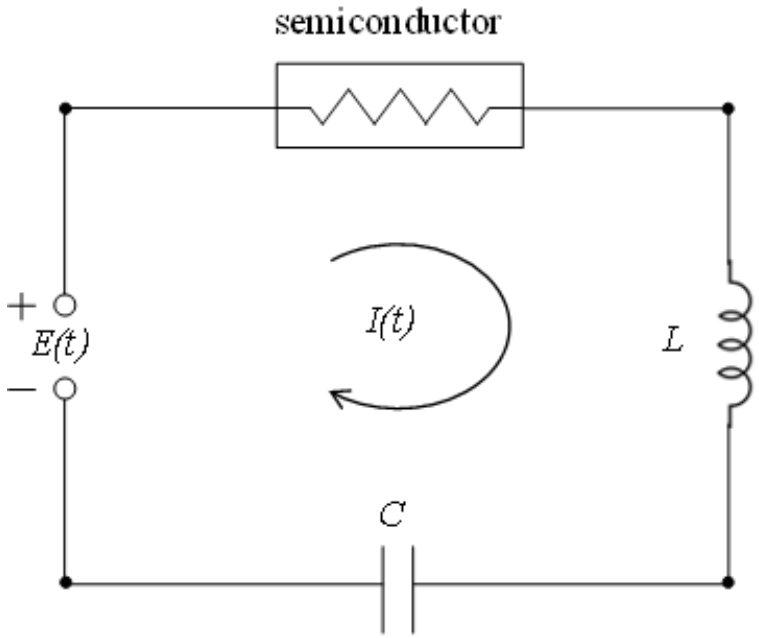

Рис. 4.11: Электрическая схема RLC-контура с активным элементом.

При $\sigma = 0$ система (4.17) вырождается в простой гармонический осциллятор. При $\sigma > 0$ решение в фазовой плоскости также является замкнутой кривой, но ее вид существенно отличен от эллипса. При $\sigma \gg 1$ задача (4.17) является трудной, причем характер трудности неодинаков

на разных участках цикла. На одних участках задача жесткая, на других – плохо обусловленная, на третьих в спектре якобиана возникают большие мнимые части. Профиль решения уравнения (4.17) при $\sigma = 100$ в фазовых переменных $v(u)$ приведен на рис.4.12.

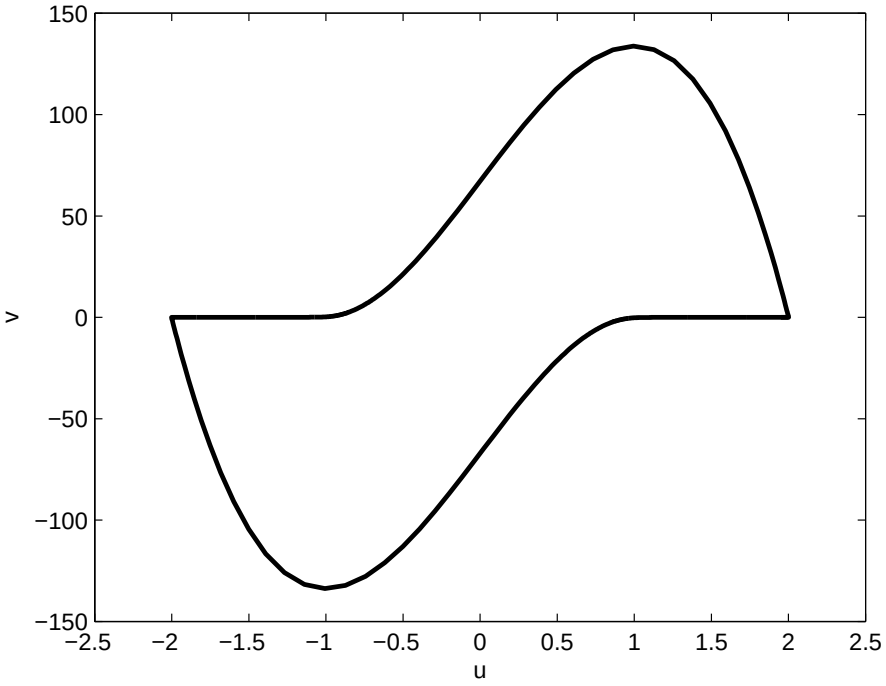

Рис. 4.12: Фазовый портрет системы (4.17) при $\sigma = 100$.

4.3.1. Разностные схемы.

Для численного интегрирования системы (4.17) использовались следующие разностные схемы. Из семейства явных схем были взяты схема Рунге-Кутты 2-го порядка точности (ERK2):

$$\begin{aligned} &\mathbf{p}_1 = \mathbf{f}(\mathbf{u}), \\ &\mathbf{p}_2 = \mathbf{f}(\mathbf{u} + 2\tau\mathbf{p}_1/3), \\ &\hat{\mathbf{u}} = \mathbf{u} + \tau(\mathbf{p}_1 + 3\mathbf{p}_2)/4. \end{aligned} \tag{4.18}$$

и классическая явная схема Кутты 4-го порядка точности (ERK4)

$$\begin{aligned} &\mathbf{p}_1 = \mathbf{f}(\mathbf{u}), \\ &\mathbf{p}_2 = \mathbf{f}(\mathbf{u} + \tau\mathbf{p}_1/2), \\ &\mathbf{p}_3 = \mathbf{f}(\mathbf{u} + \tau\mathbf{p}_2/2), \\ &\mathbf{p}_4 = \mathbf{f}(\mathbf{u} + \tau\mathbf{p}_3), \\ &\hat{\mathbf{u}} = \mathbf{u} + \tau(\mathbf{p}_1 + 2\mathbf{p}_2 + 2\mathbf{p}_3 + \mathbf{p}_4)/6. \end{aligned} \tag{4.19}$$

Обе эти схемы не могут иметь A-устойчивости, и считаются непригодными для жестких задач.

Из семейства явно-неявных схем Розенброка были взяты одностадийная схема с комплексным коэффициентом 2-го порядка точности CROS (4.9) и двухстадийная схема 4-го порядка точности CROS4 (4.14).

Из семейства полностью неявных схем Рунге-Кутты были взяты классическая неявная схема Эйлера 1-го порядка точности OIRK1 (4.7) и рекурсивная обратная схема 2-го порядка точности BMP (1.8). Эти схемы являются наиболее надежными для расчета жестких задач благодаря ньютоновским итерациям до сходимости.

Расчеты проводились как при аргументе “время” t, так и при аргументе “длина дуги” l на последовательности равномерных сеток, рекуррентно сгущающихся вдвое. Точное решение задачи (4.17) не выражается в элементарных функциях. Поэтому контроль точности возможен только по сравнению численных решений на парах соседних сеток.

4.3.2. Сравнение схем.

Сравнение разностных схем проводилось при $\sigma = 100$, что считается достаточно трудной задачей. Расчеты с крупным шагом оказывались невозможными: ни в одной схеме не получалось замыкание цикла. Поэтому приходилось визуально контролировать наличие замыкания. Результаты расчета погрешности представлены на рис. 4.13. На нем по оси абсцисс как для аргумента t, так и для l нанесено количество узлов для одного цикла. Обсудим эти результаты.

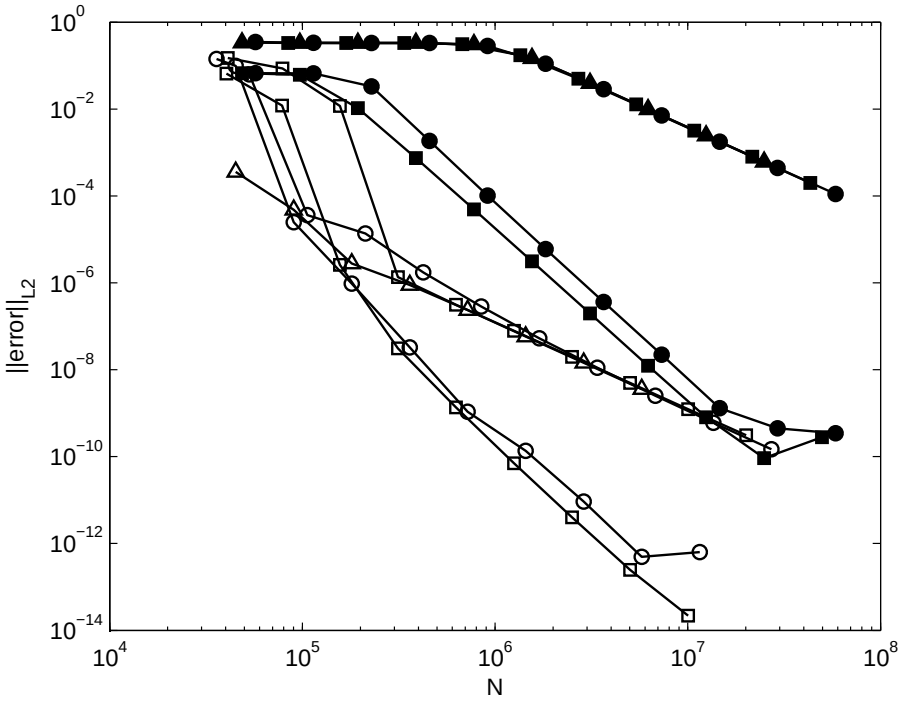

Рис. 4.13: Задача (4.17) для $\sigma = 100$. Маркеры черные для аргумента t и светлые для аргумента l; квадратики для ERK2 и ERK4, кружки для CROS и CROS4, треугольники для схемы BMP.

Каждая кривая имеет хороший прямолинейный (регулярный) участок, наклон которого соответствует теоретическому порядку точности схемы. Это показывает, что метод Ричардсона применим, так что полученные оценки погрешности являются асимптотически точными. Поэтому можно достоверно сопоставить различные схемы.

Верхняя линия есть слияние трех кривых для схем второго порядка при аргументе t. На первый взгляд это кажется странным: явная и неявные схемы дали одинаковую точность, хотя

явные схемы принято считать плохими для жестких задач. Причина по всей видимости в том, что задача (4.17) не является чисто жесткой. В ней участки плохой обусловленности и комплексности спектра матрицы Якоби оказывают не меньшее влияние на расчет, чем участки собственно жесткости. Неявные же схемы эффективны на жестких участках решения.

Третья кривая соответствует тем же схемам второго порядка точности, но при аргументе l. Опять эти схемы дают практически одинаковые результаты. Но теперь погрешность в $\sim 10^6$ раз меньше, чем при аргументе t. Если же требовать в расчетах одинаковой точности, то по длине дуги нужно взять в $\sim 10^3$ меньше узлов (поскольку это схемы второго порядка точности). Это наглядно показывает, какой количественный выигрыш может давать интегрирование по длине дуги.

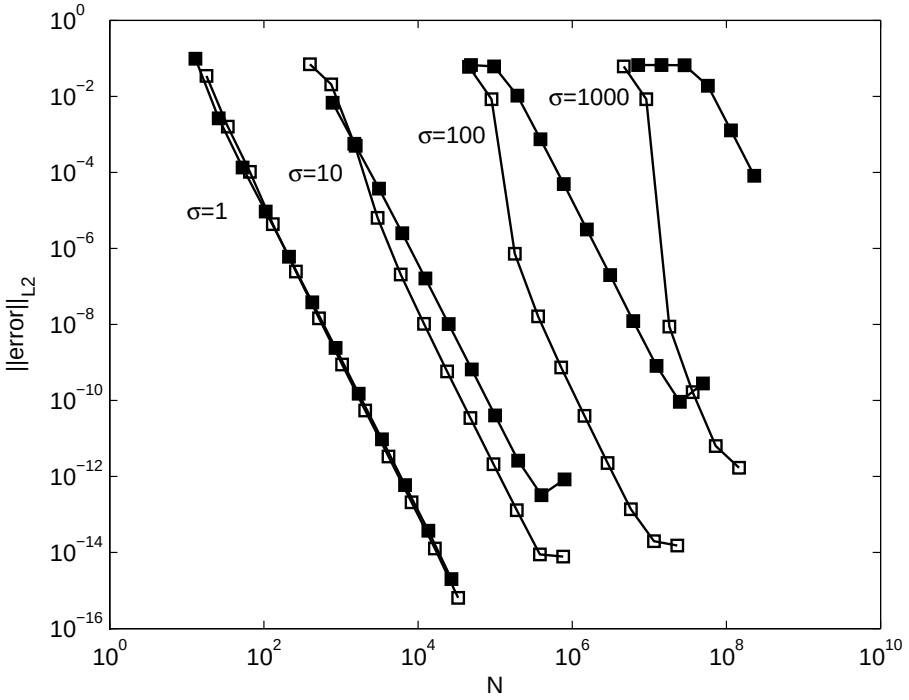

Рис. 4.14: Задача (4.17), схема $ERK4$; ■ – расчеты по t, □ – расчеты по l. Около линий указаны значения σ.

Для схем 4-го порядка выводы качественно аналогичны. В них кривые для явной и неявной схем лишь немного различаются. Явная схема Кутты оказывается чуть точнее схемы $CROS4$ благодаря своему маленькому коэффициенту в остаточном члене. Разница в точности при одинаковых числах узлов по аргументам t или l составляет $\sim 10^5$; это несколько меньше, чем для схем 2-го порядка. Разница в трудоемкости при одинаковой точности составляет ~ 18 раз. Однако не следует делать вывод о меньшей выгодности длины дуги для схем 4-го порядка. Видно, что схемы 4-го порядка по l позволяют выйти на предельно высокую точность – на ошибки округления (излом графика с переходом в горизонтальную линию).

4.3.3. Влияние жесткости.

Влияние жесткости иллюстрируется только на одной явной схеме Кутты 4-го порядка (на остальных схемах получаются сходные результаты). В расчетах брались значения параметра от

$\sigma = 1$ до $\sigma = 1000$. Первое соответствует мягкой задаче, последнее – очень жесткой. Результаты представлены на рис. 4.14.

О трудности задачи свидетельствует минимальное число узлов N, при котором удается получить замыкание цикла. При $\sigma = 1$ это $N \approx 100$, при $\sigma = 10$ это $N \approx 1000$, при $\sigma = 100$ это $N \approx 10^5$, а при $\sigma = 1000$ это $N \approx 10^7$. Отсюда видно, насколько трудной становится задача при $\sigma = 1000$.

Все кривые, даже при наибольшем $\sigma = 1000$, имеют прямолинейные регулярные участки. Их наклоны соответствуют теоретическому порядку точности $p = 4$. При $\sigma = 1$ переход от t к l практически не дает преимуществ. При $\sigma = 10$ длина дуги дает выигрыш в ~ 10 раз. При $\sigma = 100$ выигрыш составляет $\sim 10^5$ раз, а при $\sigma = 1000$ это $\sim 10^9$ раз. Чем выше жесткость, тем большим оказывается выигрыш в точности.

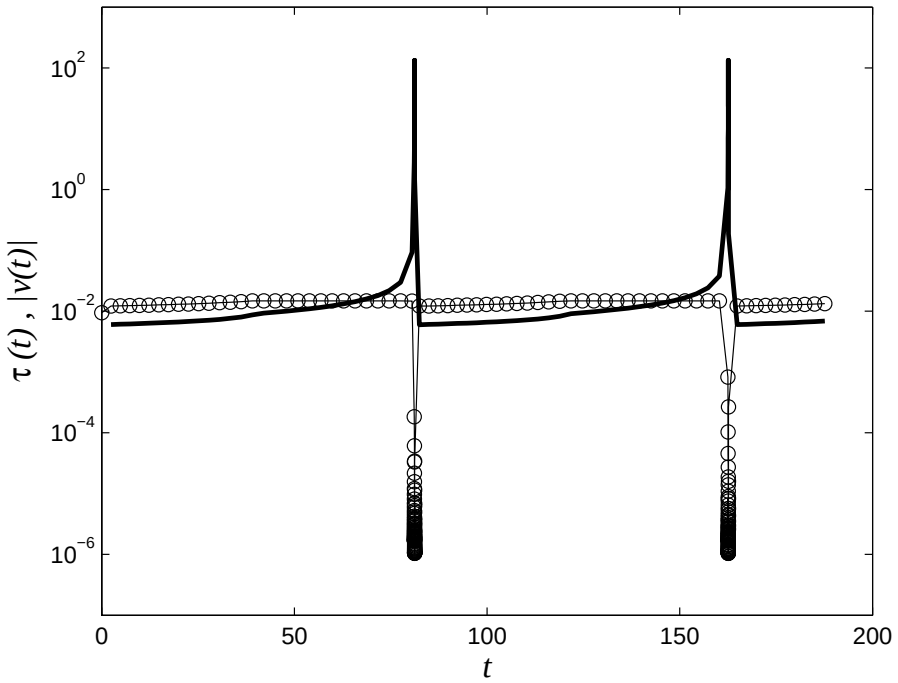

Рис. 4.15: Изменение шага $\tau(t)$ при интегрировании по аргументу l для задачи (4.17), схема ERK4; $\circ$ – зависимость $\tau(t)$, сплошная линия – компонента решения $|v(t)|$.

Причины получаемого выигрыша можно проиллюстрировать следующим образом. При расчете на равномерной сетке по аргументу l сетка по t становится неравномерной. Зависимость размера шага по времени τ от момента времени t и график компоненты $|v(t)|$ решения задачи (4.17) при $\sigma = 100$ приведены на рисунке (4.15). Расчет проводился с помощью схемы ERK4 на равномерной сетке по аргументу l с числом узлов $N = 50000$. Видно, что на регулярных участках шаг по времени довольно крупный, $\tau \sim 10^{-2}$. Однако в пограничном слое размер шага резко снижается до $\tau \sim 10^{-6}$. При интегрировании по аргументу t на сетке с таким же числом узлов величина постоянного шага $\tau = 3.4 \cdot 10^{-3}$. Это даст несколько лучшую точность в регулярной области, но пограничный слой будет покрыт такой сеткой существенно хуже.

Таким образом, даже для задач очень высокой жесткости метод длины дуги позволяет уверенно пользоваться сгущением сетки и оценкой точности по Ричардсону. При этом получается

огромный выигрыш в точности по сравнению с интегрированием по времени.

4.4. Сверхжесткость.

Для уравнения ван дер Пола характерной константой жесткости является величина σ. Задача с $\sigma = 100...1000$ считается достаточно трудной для тестирования методов решения жестких систем (задача ван дер Пола с параметром жесткости $\sigma = 10^4$ входит в набор стандартных процедур системы MATLAB). Однако имеются важные классы задач, в которых параметр жесткости может превышать 10^{10}. Например, такими являются задачи химической кинетики, поскольку скорости различных реакций в них могут отличаться на много порядков. Такие задачи можно назвать сверхжесткими (хотя правильнее назвать их сверхтрудными, поскольку в них есть как процессы сверхбыстрого затухания, так и процессы очень быстрого нарастания). Они предъявляют гораздо более высокие требования к надежности численных методов.

Рассмотрим возникающие трудности на классическом тесте Далквиста с очень большой константой:

$$du/dt = -\lambda u, \quad 0 \le t \le 1, \quad u(0) = 1, \quad \lambda = 10^9 \gg 1. \tag{4.20}$$

Точное решение этой задачи есть $u(t) = \exp(-\lambda t)$; это положительная функция, убывающая очень быстро, почти скачкообразно. Если перейти к длине дуги, то (4.20) заменится системой:

$$\begin{cases} du/dl = -\lambda u/\sqrt{1+\lambda^2 u^2}, \\ dt/dl = 1/\sqrt{1+\lambda^2 u^2}, \end{cases}, \quad u(0) = 1,\ t(0) = 0. \tag{4.21}$$

Решение этой задачи при $\lambda \gg 1$ с высокой точностью описывается ломаными

$$u(l) \approx \begin{cases} 1-l \\ 0 \end{cases}, \quad t(l) \approx \begin{cases} 0, & \text{при } 0 \le l \le 1, \\ l-1, & \text{при } 1 < l \le 2. \end{cases} \tag{4.22}$$

Это решение очень похоже на рис.2.3. Потребуем, чтобы метод давал хорошее решение задачи (4.20) при шаге $\tau \gg 1/\lambda$.

С решением задачи Далквиста для независимой переменной t хорошо справляются известные методы Розенброка, Розенброка-Ваннера и др [5]. Однако при переходе к длине дуги ряд качественых свойств этих методов (монотонность и положительность решения) не сохраняется. Это легко иллюстрируется на численных примерах. Возьмем шаг по длине дуги $h = 0.1$, так что полное число шагов $N = 20$. Опишем результаты расчетов, произведенных по различным неявным схемам.

1. Наиболее надежной принято считать неявную схему Эйлера OIRK1 (4.7), в которой решение неявной алгебраической системы находится ньютоновскими итерациями. Эти расчеты выполнялись как с разностным вычислением якобиана, так и с аналитическими формулами. В обоих случаях расчеты хорошо шли до $l \approx 1$, а далее ньютоновские итерации переставали сходиться. Отсутствие сходимости ньютоновских итераций на сверхжестких задачах часто встречается у любых неявных схем. Поэтому такие схемы целесообразно применять лишь на задачах умеренной жесткости, в которых трудность задачи связана не столько с жесткостью, сколько с нелинейностью задачи.

2. Среди явно-неявных, то есть безытерационных, схем наиболее надежной считается одностадийная неявная схема Розенброка. Она получается из схемы CROS (4.9), если перед матрицей Якоби вместо параметра $(1+i)/2$ поставить 1. Один расчет по такой схеме был проведен с разностным вычислением матрицы Якоби и 64-битовыми числами. Этот расчет отлично шел до значения $l \approx 1$; при этом u_n монотонно убывали и оставались положительными. Но при дальнейшем увеличении l величины u_n стали попеременно принимать значения разных знаков на уровне $\sim 10^{-5}$. При этом значения t_n возрастали очень слабо, оставаясь близкими к 0 вместо стремления к 1 (см. рис.4.16). Возникло предположение, что такое поведение связано с ошибками округления.

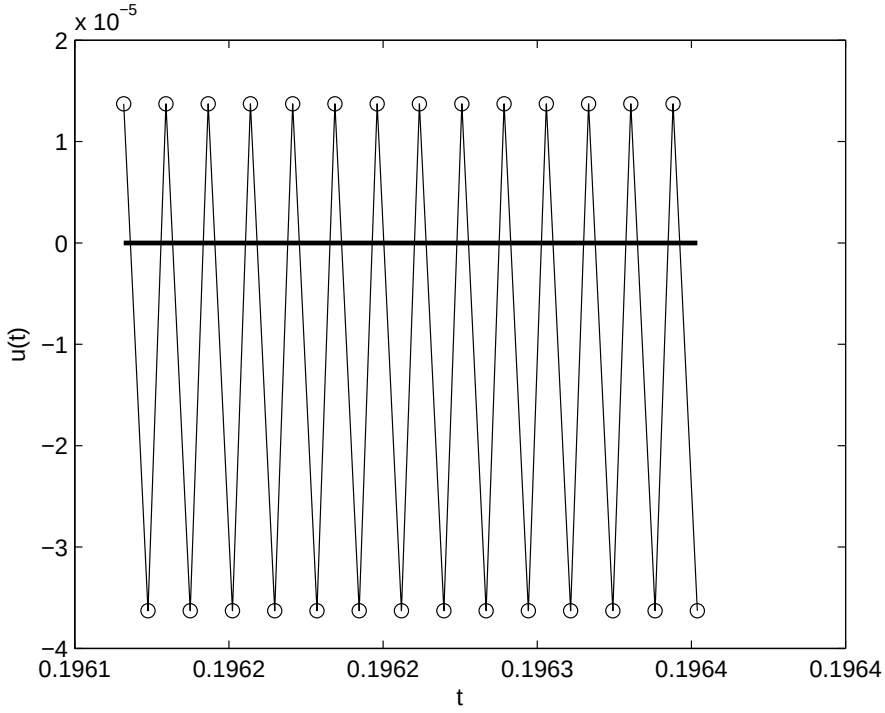

Рис. 4.16: Участок решения задачи (4.21), вычисленной с помощью неявной схемы Розенброка и 64-битовыми числами при $l > 1$. $\circ$ – численное решение, жирная линия – точное решение.

Для проверки этого предположения был проведен аналогичный расчет со 128-битовыми числами. Результаты при $l > 1$ кардинально улучшились. Значения u_n оставались по-прежнему знакопеременными, но на гораздо меньшем уровне $\sim \pm 10^{-20}$ и меньше. Значения t_n при этом возрастали в соответствие с точным решением (4.22). Все это подтверждает гипотезу о влиянии ошибок округления.

Еще лучшие результаты получаются, если матрицу Якоби вычислять не разностно, а по точным формулам. Матрица Якоби для задачи (4.21) имеет следующий вид:

$$D = \begin{pmatrix} -\dfrac{\lambda}{(1+\lambda^2 u^2)^{3/2}} & 0 \\ -\dfrac{\lambda^2 u}{(1+\lambda^2 u^2)^{3/2}} & 0 \end{pmatrix}. \tag{4.23}$$

В этом случае даже при 64-битных вычислениях получаются превосходные результаты. Значения u_n монотонно убывают на всем отрезке. При $l \leq 1$ убывание линейное, в соот-

ветствие с точным решением (4.22). При $l > 1$ u_n очень малы и убывают в геометрической прогрессии со знаменателем $(h\lambda)^{-1} = 10^{-8}$. Такое поведение численного решения можно считать эталонным.

3. Для схемы CROS (4.9) при разностном вычислении матрицы Якоби результаты практически неотличимы от только что описанных как при 64-битовых вычислениях, так и при 128-битовых. Однако при аналитическом вычислении матрицы Якоби и 64-битовых вычислениях они несколько хуже эталонных: для $l_n = 1.1$ значение численного решения становится отрицательным, $u_n \approx -10^{-30}$, и далее остается отрицательным, быстро убывая по модулю. Это указывает на несколько меньшую надежность схемы CROS даже на простейшем тесте Далквиста. На более сложных задачах этот эффект может оказаться сильнее.

4.5. Химическая кинетика.

Задачи химической кинетики, описывающие изменения концентраций веществ в ходе химических реакций, также можно отнести к классу сверхжестких. В таких задачах обычно одновременно присутствуют как медленно, так и очень быстро протекающие химические реакции. Кроме того, практические задачи приводят к системам уравнений большой размерности. Приведем содержательный пример.

4.5.1. Постановка задачи.

Рассмотрим тест из сборника [4], взятый из [60]. В исходной работе рассмотрен процесс горения природного газа в воздухе, причем природный газ является метаном с сернистыми загрязнениями. В тестовом сборнике приведена не полная система, а лишь ее часть, отвечающая за выброс вредных компонент в атмосферу. Она содержит 25 химических уравнений и 20 ком-

понент. Система имеет канонический вид (1). В ней правые части имеют следующий вид:

$$\mathbf{f} = \begin{pmatrix} -\sum\limits_{j\in\{1,10,14,23,24\}} r_j + \sum\limits_{j\in\{2,3,9,11,12,22,25\}} r_j \\ -r_2 - r_3 - r_9 - r_{12} + r_1 + r_{21} \\ -r_{15} + r_1 + r_{17} + r_{19} + r_{22} \\ -r_2 - r_{16} - r_{17} - r_{23} + r_{15} \\ -r_3 + 2r_4 + r_6 + r_7 + r_{13} + r_{20} \\ -r_6 - r_8 - r_{14} - r_{20} + r_3 + 2r_{18} \\ -r_4 - r_5 - r_6 + r_{13} \\ r_4 + r_5 + r_6 + r_7 \\ -r_7 - r_8 \\ -r_{12} + r_7 + r_9 \\ -r_9 - r_{10} + r_8 + r_{11} \\ r_9 \\ -r_{11} + r_{10} \\ -r_{13} + r_{12} \\ r_{14} \\ -r_{18} - r_{19} + r_{16} \\ -r_{20} \\ r_{20} \\ -r_{21} - r_{22} - r_{24} + r_{23} + r_{25} \\ -r_{25} + r_{24} \end{pmatrix}. \tag{4.24}$$

Выражения для вспомогательных переменных r_j и константы реакций k_j приведены в табл. 4.3 и 4.4.

Таблица 4.3: Вспомогательные переменные задачи (4.24).

$r_1 = k_1 \cdot u_1$	$r_{10} = k_{10} \cdot u_{11} \cdot u_1$	$r_{19} = k_{19} \cdot u_{16}$
$r_2 = k_2 \cdot u_2 \cdot u_4$	$r_{11} = k_{11} \cdot u_{13}$	$r_{20} = k_{20} \cdot u_{17} \cdot u_6$
$r_3 = k_3 \cdot u_5 \cdot u_2$	$r_{12} = k_{12} \cdot u_{10} \cdot u_2$	$r_{21} = k_{21} \cdot u_{19}$
$r_4 = k_4 \cdot u_7$	$r_{13} = k_{13} \cdot u_{14}$	$r_{22} = k_{22} \cdot u_{19}$
$r_5 = k_5 \cdot u_7$	$r_{14} = k_{14} \cdot u_1 \cdot u_6$	$r_{23} = k_{23} \cdot u_1 \cdot u_4$
$r_6 = k_6 \cdot u_7 \cdot u_6$	$r_{15} = k_{15} \cdot u_3$	$r_{24} = k_{24} \cdot u_{19} \cdot u_1$
$r_7 = k_7 \cdot u_9$	$r_{16} = k_{16} \cdot u_4$	$r_{25} = k_{25} \cdot u_{20}$
$r_8 = k_8 \cdot u_9 \cdot u_6$	$r_{17} = k_{17} \cdot u_4$	
$r_9 = k_9 \cdot u_{11} \cdot u_2$	$r_{18} = k_{18} \cdot u_{16}$	

Начальные данные $\mathbf{u}_0$ для каждой компоненты берутся по окончании основного горения и соответствуют догоранию:

$$\mathbf{u}_0 = (0; 0.2; 0; 0.04; 0; 0; 0.1; 0.3; 0.001; 0; 0; 0; 0; 0; 0; 0; 0.007; 0; 0; 0)\,. \tag{4.25}$$

Интервал интегрирования по времени возьмем $0 \le t \le 1.2$.

Таблица 4.4: Константы реакций задачи (4.24).

$k_1 = 0.350$	$k_{10} = 0.900 \cdot 10^4$	$k_{19} = 0.444 \cdot 10^{12}$
$k_2 = 0.266 \cdot 10^2$	$k_{11} = 0.220 \cdot 10^{-1}$	$k_{20} = 0.124 \cdot 10^4$
$k_3 = 0.123 \cdot 10^5$	$k_{12} = 0.120 \cdot 10^5$	$k_{21} = 0.210 \cdot 10$
$k_4 = 0.860 \cdot 10^{-3}$	$k_{13} = 0.188 \cdot 10$	$k_{22} = 0.578 \cdot 10$
$k_5 = 0.820 \cdot 10^{-3}$	$k_{14} = 0.163 \cdot 10^5$	$k_{23} = 0.474 \cdot 10^{-1}$
$k_6 = 0.150 \cdot 10^5$	$k_{15} = 0.480 \cdot 10^7$	$k_{24} = 0.178 \cdot 10^4$
$k_7 = 0.130 \cdot 10^{-3}$	$k_{16} = 0.350 \cdot 10^{-3}$	$k_{25} = 0.312 \cdot 10$
$k_8 = 0.240 \cdot 10^5$	$k_{17} = 0.175 \cdot 10^{-1}$	
$k_9 = 0.165 \cdot 10^5$	$k_{18} = 0.100 \cdot 10^9$	

На рис. 4.17 приведены графики изменений некоторых концентраций. Видно, что эти концентрации имеют сильно различающиеся масштабы.

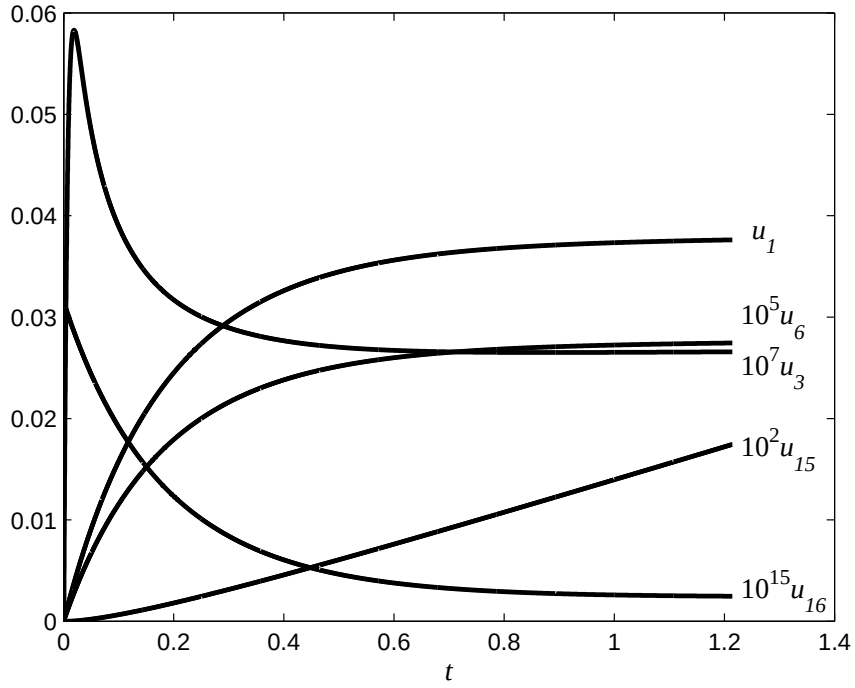

Рис. 4.17: Графики некоторых концентраций задачи (4.24).

4.5.2. Требования к разностным схемам.

Явные схемы Рунге-Кутты оказались непригодными для решения таких задач. При интегрировании по времени счет разваливается на первых же шагах даже при очень маленьких значениях шага $h_t \sim 10^{-8}$: значения концентраций по модулю возрастают и превосходят представимые на компьютере числа.

Переход к длине дуги меняет характер трудностей. Первый шаг после начального приближения дает правдоподобное решение. Однако на дальнейших шагах время t практически не

возрастает. По-видимому, источник проблем – немонотонность явных схем Рунге-Кутты. Расчетные графики концентраций становятся пилообразными аналогично разделу 4.4 (см. рис. 4.16). При этом амплитуда пилы может быть на порядки больше, чем точные значения концентраций. Из-за этого концентрации могут принимать даже отрицательные значения, что химически бессмысленно. Большие концентрации приводят к большим значениям правых частей $\mathbf{f}(\mathbf{u})$. Они входят в знаменатель правой части $\mathbf{F}$ для функции $t(l)$. В результате эта правая часть оказывается исчезающе малой, и увеличения расчетного t от шага к шагу практически не происходит.

Этот анализ показывает, что для расчета подобных задач необходимо пользоваться монотонными схемами. Монотонными [38] являются схемы CROS (4.9), неявная схема Эйлера (4.7) и вообще оптимальные обратные схемы Рунге-Кутты (1.3). При интегрировании как по t, так и по l они дают хорошее качественное поведение численного решения, не содержащее пилы.

Эти соображения справедливы при вычислениях с бесконечной разрядностью. При конечной разрядности вычислений возникает одна тонкость. Расчеты с сильной разномасштабностью констант реакций невозможно выполнять при 32-разрядных числах: ошибки округления становятся сопоставимыми с самим решением. При 64-разрядных числах можно производить хорошие расчеты, если матрица Якоби системы записывается аналитически. Но если матрица Якоби вычисляется разностно, то для нее необходимо использовать 128-разрядные числа (решать же получающуюся линеаризованную систему можно и с 64-разрядными числами). Последняя тонкость особенно существенна, если аргументом является длина дуги.

4.5.3. Результаты расчетов.

Расчеты тестовой задачи (4.24) проводились как для аргумента t, так и для аргумента l. Хорошие результаты в обоих случаях дала схема CROS, где матрица Якоби вычислялась разностно с использованием 128-разрядных чисел. Остальные вычисления проводились с 64-разрядными числами, что при решении вспомогательной линейной системы давало существенную экономию времени.

Если известно аналитическое выражение для вычисления матрицы Якоби правой части при аргументе t, то нетрудно получить аналитическое выражение и для матрицы Якоби при аргументе l. Действительно, из выражения (2.4) для $F_m(\mathbf{u})$ получим:

$$\frac{\partial F_i}{\partial u_j} = \frac{\partial f_i}{\partial u_j} \frac{1}{\sqrt{1+\sum\limits_k f_k^2}} - \left(\sum_k \frac{\partial f_k}{\partial u_j} \right) \frac{f_i}{\left(1+\sum\limits_k f_k^2\right)^{3/2}}. \tag{4.26}$$

Контроль точности выполнялся проведением серии расчетов на последовательности сеток, рекуррентно сгущающихся в два раза. При этом производилась апостериорная асимптотически точная оценка погрешности по методу Ричардсона. На рис. 4.18 приведена в двойном логарифмическом масштабе полученная зависимость погрешности от числа узлов сетки. Наличие хорошего прямолинейного участка с наклоном 2, соответствующим теоретическому порядку точности схемы, является подтверждением законности применения метода Ричардсона.

Видно, что маркеры для аргументов t и l легли практически на общую прямую. Тем самым в данной задаче переход к длине дуги не дает выигрыша в точности по сравнению с аргументом t. Это показывает, что в ней равную трудность представляют как участки затухания компонент,

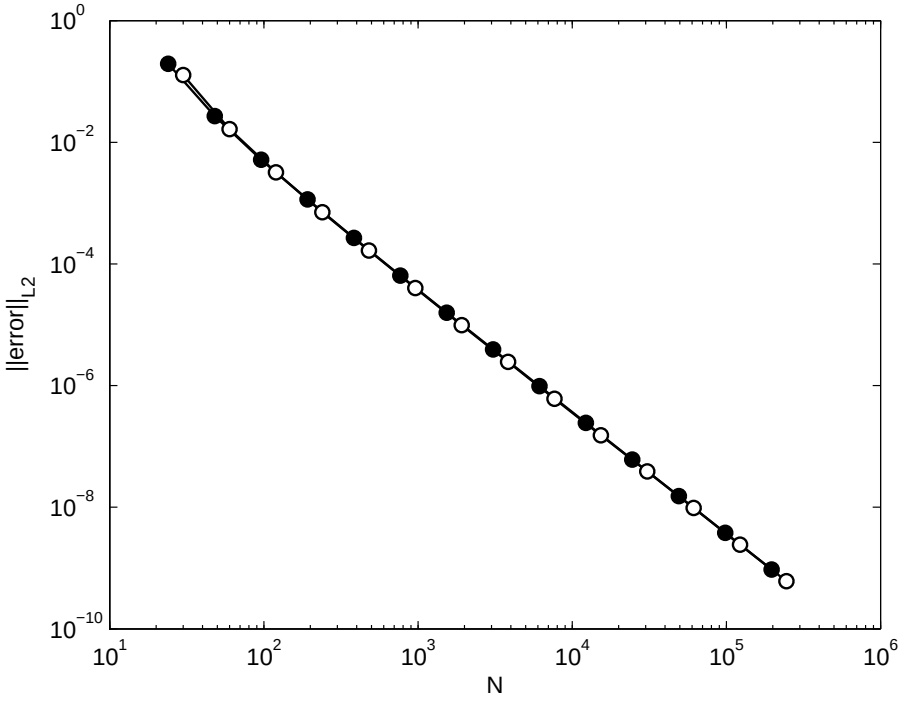

Рис. 4.18: Погрешность расчета задачи (4.24) по схеме CROS на серии сгущающихся сеток. • - интегрирование по времени; ○ - интегрирование по длине дуги.

так и участки быстрого роста (плохой обусловленности). По-видимому, выигрыш от перехода к длине дуги следует ожидать при моделировании реакций, имеющих взрывообразный характер.

Рекомендации. Реальные задачи химической кинетики целесообразно решать, беря в качестве аргумента время t и используя одностадийную схему CROS (возможно также двустадийную схему CROS4). При этом расчеты нужно проводить с числами не менее 64 разрядов, а матрицу Якоби вычислять разностным методом со 128-разрядными числами.

4.6. Выводы.

Подробное исследование моделей разного класса задач позволяет сформулировать следующие выводы.

Несмотря на то, что неявные методы Рунге-Кутты являются более сложными в реализации, и расчет с их использованием занимает больше времени из-за необходимости выполнения итераций до сходимости, их целесообразно применять для решения задач с разрывными решениями или начальными данными. Для таких задач схемы BORK и BMP позволяют при разумных затратах вычислительных ресурсов получать качественно верное решение.

Дифференциально-алгебраические задачи, описывающие электрические схемы, достаточно хорошо решаются с помощью схем CROS и CROS4, 2-го и 3-го порядка точности соответственно. Несмотря на то, что с помощью схемы OIRK4 можно достичь 4-го порядка точности, время расчета с помощью этой схемы становится слишком большим из-за быстро возрастающего порядка нелинейной системы, которую нужно решить для перехода на новый слой.

Переход к длине дуги при интегрировании системы ОДУ часто позволяет получить существенный выигрыш в точности. Однако он приводит к системе ОДУ с существенно более громоздкими правыми частями, чем у исходной системы. Чем выше порядок системы, тем сильнее это усложнение. Оно существенно даже при использовании явных схем. Если же используются неявные схемы, то требуется вычислять матрицу Якоби правых частей; это еще многократно усложняет решение. С учетом этих соображений, обсудим классы задач, для которых метод длины дуги эффективен или неэффективен.

Очевидно, метод высокоэффективен для плохообусловленных систем ОДУ не слишком высокого порядка, поскольку для таких задач достаточны явные схемы. В частности, сюда относятся задачи с сингулярностью.

Системы ОДУ, в которых жесткость сильно преобладает над другими видами сложности, требуют неявных схем. То же относится к задачам, где жесткость и плохая обусловленность примерно равноценны (в эту категорию попадают задачи химической кинетики). Для таких систем выполнение шага по длине дуги гораздо более трудоемко, чем при интегрировании по времени. В этом случае метод длины дуги может оказаться невыгодным, и лучшие результаты дадут схемы Розенброка с комплексными коэффициентами при аргументе t.

Применение длины дуги для одномерных уравнений в частных производных, решаемых методом прямых (см., например [45] и [44]), представляется мало перспективным. В самом деле, современные расчеты должны включать апостериорную оценку погрешности, которая делается сгущением сеток по пространственным координатам x и времени t. Но при сгущении сетки по x соответственно возрастает порядок системы ОДУ и одновременно увеличивается объем расчетов каждой правой части. Полный расчет может стать слишком трудоемким даже при явных схемах, и неприемлемым при неявных. Авторы указанных выше работ тестировали переход к длине дуги на задаче с известным точным решением. Интегрирование по длине дуги при этом давало меньшую погрешность, чем интегрирование по времени.

Таким образом, выбор наилучшего метода решения зависит от особенностей конкретной задачи. Универсальный подход для решения сводящихся к жестким системам ОДУ моделей подобрать не представляется возможным.

Глава 5

Комплекс программ GEABORK.

Комплекс программ GEABORK (“Guaranteed Error for Arc Backward Optimal Runge-Kutta methods”) был создан в процессе разработки и отладки численных методов, представленных в данной работе. Он написан на языке программирования MATLAB, полностью совместимом со свободно распространяемой средой для математических вычислений Octave. Комплекс включает в себя широкий выбор численных методов для интегрирования жестких и плохо обусловленных систем ОДУ и дифференциально-алгебраических систем, а также необходимые вспомогательные подпрограммы (обобщенный метод Ньютона для решения нелинейных алгебраических систем, определение погрешности численного решения по методу Ричардсона, разностное вычисление матрицы Якоби, методы автономизации задачи и т.д.).

Среда Octave позволяет максимально быстро приступать к работе с программным пакетом, так как он полностью предоставляется в исходных кодах и не требует дополнительных шагов по установке. Богатая стандартная библиотека среды позволяет делать исходный код программ достаточно компактным и легко визуализировать получаемые результаты. Недостатком выбранной среды является не всегда достаточное быстродействие. Например, применение подпрограмм автономизации задачи при помощи анонимных функций связано с существенными накладными вычислительными затратами. Поэтому для проведения масштабных расчетов может оказаться необходимым реализовать тот или иной численный метод на языке C/C++.

Все компоненты комплекса GEABORK сопровождаются автоматизированными тестами для быстрой проверки правильности реализации.

5.1. Численные методы решения систем ОДУ.

5.1.1. Явные методы Рунге-Кутты.

Явные схемы Рунге-Кутты, реализованные в данном пакете, описаны в работе [33]. Это оптимальные явные схемы с 1-го по 4-й порядок точности, а также семистадийная схема Хаммуда 6-го порядка точности. В работе [33] были исправлены некоторые опечатки в ее коэффициентах.

```
1 function [u] = erk1(f, t0, u0, tau)
2 %ERK1 ERK1 method for non-stiff ODE problems.
```

```
%   Explicit Runge-Kutta 1st order method. It is not A-stable.
u = u0 + tau * f(t0, u0);
end
```

```
function [u] = erk2(f, t0, u0, h)
%ERK2 ERK2 method for non-stiff ODE problems.
%   Explicit Runge-Kutta 2nd order method. It is not A-stable.
p1 = f(t0, u0);
p2 = f(t0+2/3*h, u0+2*h*p1/3);

u = u0 + h/4*(p1 + 3*p2);
end
```

```
function [u] = erk4(f, t0, u0, h)
%ERK4 ERK4 method for non-stiff ODE problems.
%   Explicit Runge-Kutta 4th order method. It is not A-stable.
%   Method coefficients in the form of Butcher matrix are the following:
%   A = [0, 0, 0, 0;
%        1/2, 0, 0, 0;
%        0, 1/2, 0, 0;
%        0, 0, 1, 0];
%   B = [1, 2, 2, 1] / 6;
%   C = [0; 1/2; 1/2; 1];

k1 = f(t0, u0);
k2 = f(t0 + h/2, u0+h*k1/2);
k3 = f(t0 + h/2, u0+h*k2/2);
k4 = f(t0 + h, u0+h*k3);

u = u0 + h/6*(k1 + 2*k2 + 2*k3 + k4);

end
```

```
function [u] = erk6(f, t0, u0, h)
%ERK6 ERK6 method for non-stiff ODE problems.
%   Explicit Runge-Kutta 7 stages 6th order method also known as Hammud s
%   method (some coefficients are specified by Alshina etc).It is not
%   A-stable.
%   Method coefficients in the form of Butcher matrix are provided below.

A = [0, 0, 0, 0, 0, 0, 0; ...
   4/7, 0, 0, 0, 0, 0, 0; ...
   115/112, -5/16, 0, 0, 0, 0, 0; ...
   589/630, 5/18, -16/45, 0, 0, 0, 0; ...
   229/1200-29/6000*sqrt(5), 119/240-187/1200*sqrt(5), ...
     -14/75+34/375*sqrt(5), -3/100*sqrt(5), 0, 0, 0; ...
```

```
    71/2400-587/12000*sqrt(5), 187/480-391/2400*sqrt(5), ...
    -38/75+26/375*sqrt(5), 27/80-3/400*sqrt(5), (1+sqrt(5))/4, 0, 0; ...
  -49/480+43/160*sqrt(5), -425/96+51/32*sqrt(5), ...
    52/15-4/5*sqrt(5), -27/16+3/16*sqrt(5), 5/4-3/4*sqrt(5), ...
    5/2-1/2*sqrt(5), 0];

B = [1, 0, 0, 0, 5, 5, 1] / 12;
C = [0; 4/7; 5/7; 6/7; (5-sqrt(5))/10; (5+sqrt(5))/10; 1];

DIM = length(u0);
k = zeros(DIM,7);

for i = 1:7
  du = zeros(DIM,1);
  for j = 1:i-1
    du = du + k(:,j)*A(i,j);
  end
  k(:,i) = f(t0 + h*C(i), u0 + h*du);
end

u = u0 + h*(B*k ) ;

end
```

5.1.2. Методы Розенброка.

```
function u = cros(f, t0, u0, tau, G, dfdu)
%CROS CROS method for stiff ODE and DA problems.
%   CROS is semi-implicit one stage Rosenbrock method with complex
%   coefficients. It is L2-stable, 2nd order of accuracy on differential
%   or differential-algebraic autonomous problems with constant mass
%   matrix G.

if nargin < 6
  dfdu = @jacobian;

  if nargin < 5
    G = eye(length(u0));
  end
end

g = @(x) (f(t0, x));

a = (1+1i)/2;

D = (G - a*tau*dfdu(g, u0));
F = f(t0+tau/2,u0);
k = D\F;
```

```
23
24 u = u0 + tau*real(k);
25
26 end
```

```
1  function u = cros4a(f, t0, u0, tau, G, dfdu)
2  %CROS4A CROS4A method for autonomous stiff ODE and DA problems.
3  %   CROS4A is semi-implicit two stage Rosenbrock method with complex
4  %   coefficients. It is L2-stable, 4th order of accuracy for autonomous
5  %   differential problems, 3rd order of accuracy for differential-algebraic
6  %   problems with const mass matrix G.
7
8  if nargin < 6
9    dfdu = @jacobian;
10
11   if nargin < 5
12     G = eye(length(u0));
13   end
14 end
15
16 % f(t,x) is an autonomous problem, so let s use an alias  g(x)  for it.
17 g = @(x) (f(t0,x));
18
19 alpha1 = 0.1 + (sqrt(11)/30)*1i;
20 alpha2 = 0.2 + 0.1i;
21
22 b1 = 0.1941430241155180 - 0.2246898944678803i;
23 b2 = 0.8058569758844820 - 0.8870089521907592i;
24 c21 = 0.2554708972958462 - 0.2026195833570109i;
25 a21 = 0.5617645150714754 - 1.148223341045841i;
26
27 k1 = (G - alpha1 * tau * dfdu(g, u0)) \ (tau*g(u0));
28 k2 = (G - alpha2 * tau * dfdu(g, u0 + real(a21*k1))) \ ...
29     (tau*g(u0 + real(c21*k1)));
30
31 u = u0 + real(b1*k1 + b2*k2);
32 end
```

5.1.3. Обратные и полностью неявные методы Рунге-Кутты.

Все использованные в работе неявные схемы можно свести к семейству полностью неявных схем Рунге-Кутты. Один шаг с помощью такой схемы сводится к нахождению решения нелинейной системы, вид которой определяется коэффициентами схемы. Наиболее просто реализуются схемы Crank-Nikolson (CN), Backward Midpoint Method (BMP) и оптимальные обратные схемы в рекурсивной форме (BORK).

```
function [u, deltas] = cn(f, t0, u0, tau)
%CN CN method for moderately stiff ODE problems.
%   Crank-Nicolson method is implicit, A-stable, 2-nd order of accuracy.
if nargin < 3
  error( Not enough input arguments. );
end

solver_equation = @(x) parametrized_solver_equation(t0, u0, x, tau, f);
[u, deltas] = newton(solver_equation, u0);
end


function F = parametrized_solver_equation(t0, u0, u, tau, f)
% Returns the non-linear equation which represents a numerical
% approximation of Cauchy problem according to a particular solver.
F = u - u0 - tau*(f(t0+tau,u) + f(t0,u0))/2;
end
```

```
function [u, deltas] = bmp(f, t0, u0, tau)
% BMP BMP method for stiff ODE problems.
%   Backward midpoint method is implicit, L2-stable, 2-nd order of
%   accuracy.

solver_equation = @(x) parametrized_solver_equation(t0, u0, x, tau, f);
[u, deltas] = newton(solver_equation, u0);
end


function F = parametrized_solver_equation(t0, u0, u, tau, f)
% Returns the non-linear equation which represents a numerical approximation
% of cauchy problem according to a particular solver.
u_tild = u - tau/2 * f(t0 + tau, u);
F = u - u0 - tau*f(t0 + tau/2, u_tild);
end
```

```
function [u, deltas] = bork2(f, t0, u0, tau)
%BORK2 BORK2 method for stiff ODE problems.
%   Backward optimal Runge-Kutta method. It is implicit, L2-stable,
%   2-nd order of accuracy.

solver_equation = @(x) parametrized_solver_equation(f, t0, u0, tau, x);
[u, deltas] = newton(solver_equation, u0);
end


function F = parametrized_solver_equation(f, t0, u0, tau, u)
% Returns the non-linear equation which represents a numerical approximation
% of cauchy problem according to a particular solver.
```

```
A = [0; 2/3];
B = [1/4; 3/4];

f1 = f(t0+tau,u);
f2 = f(t0+tau*(1-A(2)),u-tau*A(2)*f1);
F = u - u0 - tau*(B(1)*f1 + B(2)*f2);
end
```

```
function [u, deltas] = bork3(f, t0, u0, tau)
%BORK3 BORK3 method for stiff ODE problems.
%   Backward optimal Runge-Kutta method. It is implicit, L3-stable,
%   3-nd order of accuracy.

solver_equation = @(x) parametrized_solver_equation(f, t0, u0, tau, x);
[u, deltas] = newton(solver_equation, u0);
end


function F = parametrized_solver_equation(f, t0, u0, tau, u)
% Returns the non-linear equation which represents a numerical approximation
% of cauchy problem according to a particular solver.
A = [0; 1/2; 3/4];
B = [2/9; 3/9; 4/9];

f1 = f(t0+tau,u);
f2 = f(t0+tau*(1-A(2)),u-tau*A(2)*f1);
f3 = f(t0+tau*(1-A(3)),u-tau*A(3)*f2);
F = u - u0 - tau*(B(1)*f1 + B(2)*f2 + B(3)*f3);
end
```

```
function [u, deltas] = bork4(f, t0, u0, tau)
%BORK4 BORK4 method for stiff ODE problems.
%   Backward optimal Runge-Kutta method. It is implicit, L4-stable,
%   4-th order of accuracy.

solver_equation = @(x) parametrized_solver_equation(f, t0, u0, tau, x);
[u, deltas] = newton(solver_equation, u0);
end


function F = parametrized_solver_equation(f, t0, u0, tau, u)
% Returns the non-linear equation which represents a numerical approximation
% of cauchy problem according to a particular solver.
A = [0; 1/2; 1/2; 1];
B = [1/6; 2/6; 2/6; 1/6];

f1 = f(t0+tau,u);
f2 = f(t0+tau*(1-A(2)),u-tau*A(2)*f1);
```

```
f3 = f(t0+tau*(1-A(3)),u-tau*A(3)*f2);
f4 = f(t0+tau*(1-A(4)),u-tau*A(4)*f3);

F = u - u0 - tau*(B(1)*f1 + B(2)*f2 + B(3)*f3 + B(4)*f4);
end
```

Сходную конструкцию имеют моно-неявные методы Рунге-Кутты [35].

```
function [u] = mirk2a(f, t0, u0, tau)
%MIRK2A MIRK2A method for stiff autonomous ODE problems.
%   Mono-implicit Runge-Kutta (MIRK) implicit method. It is L1-stable,
%   2nd order of accuracy on differential problems.

solver_equation = @(x) parametrized_solver_equation(t0, u0, x, tau, f);
[u] = newton(solver_equation, u0);
end


function F = parametrized_solver_equation(t0, u0, u, tau, f)
v = 1-1/sqrt(2);
w1 = 1-2*v;
a32 = (1/3-w1)/(w1-1/2);
w3 = (w1-1/2)^2/(1/3-w1);
w2 = 1-w1-w3;

k1 = f(t0, u0);
k2 = f(t0, u);
k3 = f(t0, u+tau*a32*k2);

F = u - u0 - tau*(w1*k1 + w2*k2 + w3*k3);
end
```

Более сложный вид имеет программа для полностью неявных многостадийных схем Рунге-Кутты. Методы 2-го, 3-го и 4-го порядка точности имеют общий основной код FIRK и отличаются только матрицей коэффициентов, определяющих метод.

```
function [u, deltas] = oirk1(f, t0, u0, tau, G)
%OIRK1 OIRK1 method for stiff ODE and DA problems.
%   Implicit Euler s method also known as optimal inverse Runge-Kutta
%   (OIRK) implicit method. It is L1-stable, 1st order of accuracy on
%   differential or differential-algebraic problems with constant mass
%   matrix G.

if nargin < 5
  G = eye(length(u0));
end

solver_equation = @(x) parametrized_solver_equation(t0, u0, x, tau, G, f);
%solver_equation = @(u) (M*(u-u0) - tau*f(t0+tau, u));
```

```
14 [u, deltas] = newton(solver_equation, u0);
15 end
16
17
18 function F = parametrized_solver_equation(t0, u0, u, tau, G, f)
19 % Returns the non-linear equation which represents a numerical approximation
20 % of cauchy problem according to a particular solver.
21 F = G*(u-u0) - tau*f(t0+tau, u);
22 end
```

```
1  function [u, deltas] = oirk2(f, t0, u0, tau, G)
2  %OIRK2 OIRK2 method for stiff ODE and DA problems.
3  %   Optimal inverse Runge-Kutta (OIRK) implicit method. It is L2-stable,
4  %   2nd order of accuracy on differential or differential-algebraic
5  %   autonomous problems with constant mass matrix G.
6
7  if nargin < 5
8     G = eye(length(u0));
9  end
10
11 A = [1/4, 3/4; -5/12, 3/4];
12 b = [1/4; 3/4];
13 c = [1; 1/3];
14
15 [u, deltas] = firk(A, b, c, G, f, t0, u0, tau);
```

```
1  function [u, deltas] = oirk3(f, t0, u0, tau, M)
2  %OIRK3 OIRK3 method for stiff ODE and DA problems.
3  %   Optimal inverse Runge-Kutta (OIRK) implicit method. It is L3-stable,
4  %   3rd order of accuracy on differential or differential-algebraic
5  %   autonomous problems with constant mass matrix G.
6
7  if nargin < 5
8     M = eye(length(u0));
9  end
10
11 A = [ 2/9,    3/9,    4/9;
12       -5/18,  3/9,    4/9;
13        2/9, -15/36,   4/9];
14 b = [2; 3; 4] / 9;
15 c = [1; 1/2; 1/4];
16
17 [u, deltas] = firk(A, b, c, M, f, t0, u0, tau);
```

```
1  function [u, deltas] = oirk4(f, t0, u0, tau, M)
2  %OIRK4 OIRK4 method for stiff ODE and DA problems.
```

```
%   Optimal inverse Runge-Kutta (OIRK) implicit method. It is L4-stable,
%   4th order of accuracy on differential or differential-algebraic
%   autonomous problems with constant mass matrix G.

if nargin < 5
   M = eye(length(u0));
end

A = [ 1,  2,  2,  1;
     -2,  2,  2,  1;
      1, -1,  2,  1;
      1,  2, -4,  1] / 6;
b = [1; 2; 2; 1] / 6;
c = [1, 1/2, 1/2, 0];

[u, deltas] = firk(A, b, c, M, f, t0, u0, tau);
```

```
function [u, deltas] = firk(A, b, c, G, f, t0, u0, tau)
%FIRK Generic implicit Runge-Kutta method for stiff ODE and DA problems.
%   Fully implicit Runge-Kutta (FIRK) method with arbitrary coefficients
%   in Butcher s form: A, b and c.

s = length(b); % Number of stages.
M = length(u0); % Dimension of the problem.

% Merge s copies of vector u0 to a single vector U0.
U0 = repmat(u0, [s,1]);

solver_equation = @(x) parametrized_solver_equation(A, c, U0, x, t0, ...
    tau, G, f);

% Solve S*M-dimensional nonlinear problem to advance to a next step.
[Y, deltas] = newton(solver_equation, U0);

UU = reshape(Y, M, s);
v1 = 1 - sum(b/A);
v2 = (b/A*UU) ;

u = v1*u0 + v2;
end


function F = parametrized_solver_equation(A, c, U0, U, t0, tau, G, f)
[s,~] = size(A); % Number of stages.
M = length(U)/s; % Dimension of the origin problem.

FY = zeros(M*s, 1);
for i = 1:s
   FY(M*(i-1)+1:M*i) = f(t0 + tau*c(i), U(M*(i-1)+1:M*i));
```

```
33 end
34
35 AFY = zeros(s*M, 1);
36 GG = zeros(s*M);
37 for i = 1:s
38     i1 = (i-1)*M+1:i*M;
39     for j = 1:s
40         i2 = (j-1)*M+1:j*M;
41         AFY(i1) = AFY(i1) + A(i,j) * FY(i2);
42     end
43
44     GG(i1,i1) = G;
45 end
46
47 F = GG*(U - U0) - tau*AFY;
48
49 end
```

5.2. Подпрограммы для интегрирования на серии сгущающихся сеток.

5.2.1. Интегрирование по аргументу “время”.

```
1  function [tout, uout, conv] = solveodet(f, tspan, u0, G, atol, n_nodes, ...
2                                          solver_name, max_grids)
3  % Autonomous ODE or DA problem (G*du/dt = f(u)) solver.
4  % Problem is integrated on a set of nested uniform grids in order to
5  % calculate aposteriori error estimation in L2 norm according to
6  % Richardson s method.
7
8  if nargin < 8
9    max_grids = 10;
10   if nargin < 7
11     solver_name =  cros4 ;
12     if nargin < 6
13       n_nodes = 1e2;
14       if nargin < 5
15         atol = 1e-5;
16         if nargin < 4
17           G = eye(numel(u0));
18         end
19       end
20     end
21   end
22 end
23
```

```
conv = zeros(2, max_grids);

[solver, p] = get_solver_by_name(solver_name, G);

for i = 1:max_grids
  if (i == 1)
    u1 = integrate_ode(tspan, n_nodes, f, u0, solver, G);
  else
    u1 = u2;
  end

  n_nodes = n_nodes*2;

  u2 = integrate_ode(tspan, n_nodes, f, u0, solver, G);

  [aerr, ~] = get_error_richardson(u1, u2, p);

  conv(:,i) = [n_nodes; aerr];

  if (aerr < atol)
    break;
  end
end

tau = diff(tspan) / n_nodes;
tout = tspan(1):tau:tspan(2);
uout = u2;
conv = conv(:,1:i);

end
```

5.2.2. Интегрирование по аргументу “длина дуги”.

```
function [tout, uout, conv] = solveodel(f, tspan, u0, atol, h0, solver_name,...
                                        max_grids)
% Arbitrary ODE problem (du/dt = f(t,u)) solver.
% Problem is integrated on a set of nested uniform grids using solution arc
% length as integration parameter. Aposteriori error estimation in L2 norm
% is calculated with solution according to Richardson s method.
%
% Integration span and number of steps is not known in advance. Initial
% step size h0 shall be chosen carefully.

if nargin < 7
  max_grids = 10;
  if nargin < 6
    solver_name =  cros4 ;
    if nargin < 5
```

```
16        h0 = 1e-1;
17        if nargin < 4
18          atol = 1e-6;
19        end
20      end
21    end
22  end
23
24  MAX_STEPS = 1e6;
25
26  f_auto = autonomize_problem_arc(f);
27  u0_auto = [u0; tspan(1)];
28  conv = zeros(2, max_grids);
29  [solver, p] = get_solver_by_name(solver_name);
30
31  hl = h0;
32
33  for i = 1:max_grids
34    if (i == 1)
35      u1 = integrate(f_auto, tspan(2), u0_auto, hl, solver, MAX_STEPS);
36    else
37      u1 = u2;
38    end
39
40    hl = hl/2;
41
42    u2 = integrate(f_auto, tspan(2), u0_auto, hl, solver, MAX_STEPS);
43
44
45    [aerr, ~] = get_error_richardson(u1, u2, p);
46    [~,n_nodes] = size(u2);
47    conv(:,i) = [n_nodes; aerr];
48
49    if (aerr < atol)
50      break;
51    end
52  end
53
54  conv = conv(:,1:i);
55  tout = u2(end,:);
56  uout = u2(1:end-1,:);
57  end
58
59
60  function [u, T_reached] = integrate(f_auto, T, u0, hl, solver, max_steps)
61  % Integrate autonomized ODE problem dudt=f(u) with a provided solver until
62  % value of time is less than T.
63
64  % Approximate preallocation.
65  u = zeros(numel(u0), floor(T/hl));
```

```
u(:,1) = u0;

i = 1;
while ((u(end,i) < T) && (i < max_steps))
  u(:,i+1) = solver(f_auto, 0, u(:,i), hl);
  i = i+1;
end

% Truncate a tail of preallocated solution.
u = u(:,1:i);
T_reached = (u(end,i) >= T);
end
```

5.3. Вспомогательные подпрограммы.

5.3.1. Разностное вычисление матрицы Якоби.

```
function D = jacobian_sd(f, u)
% Numerical calculation of Jacoby matrix. Symmetric difference rule is used
% for each derivative caltulation.

hu = 5e-5;
dim = length(u);
D = zeros(dim);

for i = 1 : dim
    ut1 = u;
    ut2 = u;
    ut1(i) = ut1(i) + hu;
    ut2(i) = ut2(i) - hu;
    D(:,i) = (f(ut1) - f(ut2))/2/hu;
end

end
```

5.3.2. Метод Ньютона для решения нелинейных систем.

```
function [u, deltas, num_iterations] = newton(f, u0, dfdu, ...
                                              max_iterations, ...
                                              enable_truncations)
%NEWTON Modified Newton s method for solving non-linear systems.

if nargin < 5
  enable_truncations = true;
```

```
8
9     if nargin < 4
10      max_iterations = 0;
11
12      if nargin < 3
13        dfdu = @(x) jacobian(f,x);
14
15        if nargin < 2
16          error( Not enough arguments. );
17        end
18      end
19    end
20  end
21
22  num_iterations = 1;
23  deltas = zeros(1,max_iterations);
24  max_truncations = 10;
25
26  F = f(u0);
27
28  while (true)
29     D = dfdu(u0);
30     du = D\(-F);
31
32     if (enable_truncations)
33        [u, du, F] = truncate_newton_step(f, u0, du, F, max_truncations);
34     else
35        u = u0 + du;
36        F = f(u);
37     end
38
39     d = norm(du);
40     deltas(num_iterations) = d;
41
42     if (d <= 1e-10 || (max_iterations>0 && num_iterations == max_iterations))
43        break;
44     end
45
46     u0 = u;
47     num_iterations = num_iterations + 1;
48  end
49
50  deltas = deltas(1:num_iterations);
51  end
52
53
54  function [u, du, feval_f_u] = truncate_newton_step(f, u0, du0, feval_f_u0, ...
55                                                     max_truncations)
56  % Truncated Newton iterations: |rhs| shall decrease.
57
```

```
norm1 = norm(feval_f_u0);

n_truncations = 0;
tau = 1;
while (n_truncations < max_truncations)
   du = tau*du0;
   u = u0 + du;
   feval_f_u = f(u);

   norm2 = norm(feval_f_u);
   if (norm(du) <= 1e-10 || norm2 < norm1)
      return;
   end;

   n_truncations = n_truncations + 1;
   tau = tau / 2;
end

warning( Too many truncations. );
end
```

5.3.3. Определение погрешности.

```
function [aerr, rerr] = get_error_richardson(u1, u2, p, r)
%GET_ERROR_RICHARDSON Richardson method for solution error estimation.
%   Richardson method for solution error estimatio of the numerical result
%   calculated on two nested grids. "u2" must be thinner than "u1".
%   "p" is the approximation order of the used solver, "r" is number of
%   times grid "u2" is expected to be thinner than "u1":
%   "u1" is a matrix (dim,N), "u2" is (dim,r*N).

if nargin < 4
  r = 2;
end

% Number of nodes in two grids may be slightly different due to rounding
% errors. But as soon as u1 and u2 are expected to be two consequetive
% grids, i.e. h(u1) = r*h(u2), then we can truncate tail from one of these
% grids.
[~, n10] = size(u1);
[~, n20] = size(u2);

if r*n10-1 < n20
  % Too many nodes in 2nd grid.
  n1 = n10;
  n2 = r*n10-1;
elseif r*n10-1 > n20
  % Too many nodes in 1st grid.
```

```
26    n1 = floor((n20+1)/r);
27    n2 = r*n1-1;
28  else
29    n1 = n10;
30    n2 = n20;
31  end
32
33  assert(n1 <= n10);
34  assert(n2 <= n20);
35
36  dif = u1(:, 1:n1) - u2(:, 1:r:n2);
37
38  % Norm of average squared error summed for all nodes of grids.
39  aerr = norm(sqrt(sum(dif.^2, 2) / n1)) / (r^p-1);
40
41  % Relative error is also average squared error.
42  norm_divider = u2(:, 1:r:n2);
43  % Take absolute error as relative if divider is very small.
44  norm_divider(abs(norm_divider) < 1e-14) = 1;
45
46  rerr = norm(sqrt(sum(dif.^2 ./ norm_divider.^2, 2)) / n1 / (r^p-1));
```

Заключение

Приведём основные результаты, полученные в рамках диссертационной работы:

1. Построен новый класс неявных схем – оптимальные обратные схемы Рунге-Кутты с числом стадий $s \leq 4$. Доказано, что эти схемы Ls-устойчивы и сходятся с s-тым порядком точности как на обыкновенных дифференциальных, так и на дифференциально-алгебраических системах индекса 1. По сочетанию устойчивости, точности и экономичности они являются наилучшими среди всех схем типа Рунге-Кутты. Эти схемы эффективны для решения жестких и сверхжестких задач.
2. Показано, что для жестких задач возможно выполнение расчетов на сгущающихся сетках с получением апостериорной асимптотически точной оценки погрешности не только на регулярной части решения, но и в пограничных слоях. Показано, что автономизация жестких задач с помощью длины дуги улучшает расчет пограничных слоев и также позволяет вычислять апостериорную оценку погрешности.
3. Создан комплекс программ geabork, позволяющий проводить расчет модельных задач с использованием существующих, а также разработанных в рамках данной работы алгоритмов. Вместе с ответом программа предоставляет пользователю апостериорную асимптотически точную оценку полученного решения, а также возможность визуальной оценки достоверности полученного результата.
4. С помощью пакета geabork проведено моделирование бегущей тепловой волны, транзисторного усилителя и процесса горения метана в воздухе.

Автор выражает искреннюю благодарность своему научному руководителю Николаю Николаевичу Калиткину за руководство данной работой.

Список иллюстраций

2.1 Глобальное сгущение сеток для жесткой задачи (4.17). 26
2.2 Погрешность при решении задачи (4.17), вычисленная по методу Ричардсона на серии сгущающихся сеток. 27
2.3 а) - График $u(t)$; б) - Жирная линия - график $u(l)$, тонкая линия - график $t(l)$. . . . 28

3.1 Область притяжения корня $z = 1$ уравнения $z^3 - 1 = 0$ для метода Ньютона на комплексной плоскости. 33
3.2 Траектория спуска к комплексному корню дифференциального аналога метода Ньютона для уравнения $f(x) = x^2 + a^2 = 0$. 35
3.3 Сходимость итераций классического метода Ньютона (кружки) и с применением экстраполяции (точки) для уравнения (3.14). 39

4.1 Погрешности на гладком участке бегущей волны в норме $L2$; масштабы по осям логарифмические; $\tau/h = 1/2$; ○ – BMP, ● – OIRK1, □ – CN. 42
4.2 Профили бегущей волны для двух моментов времени $t_1 = 0.4$ и $t_2 = 0.8$. Жирные линии - точное решение. Маркерами показаны расчеты для $\tau/h = 2$: △ – CROS, ● – OIRK1 с 1 итерацией, □ – CN с 1 итерацией. 43
4.3 Профиль бегущей волны для момента времени $t = 0.8$. Жирная линия – точное решение. Маркерами показаны расчеты для $\tau/h = 2$ (итерации до сходимости): ○ – BMP, ● – OIRK1, □ – CN. 43
4.4 Профиль бегущей волны для момента времени $t = 0.2$. Жирная линия - точное решение. Маркерами показаны расчеты для $\tau/h = 4$: ■ – CN с 1 итерацией, □ – CN с 3 итерациями, ● – CN с 6 итерациями, ○ – CN с итерациями до сходимости. 44
4.5 Зависимость модуля правой части (1.14) от номера итерации. ○ – классические ньютоновские итерации; ● – усеченные итерации. 46
4.6 Разрывные начальные данные (жирная линия) и профили на первом шаге при $\tau/h = 4$: ○ – BMP, ● – OIRK1, □ – CN. 47
4.7 Погрешность численного решения задачи (4.11). Маркерами указаны значения погрешностей для оптимальных обратных схем ▲ – 2го, ■ – 3го, ● – 4го порядка точности. 48
4.8 Принципиальная схема транзисторного усилителя. 49
4.9 Решение задачи (4.13). $U_e(t)$ – входное напряжение (жирная линия), $U_8(t)$ – напряжение на выходе усилителя. 50
4.10 Погрешность численного решения задачи (4.13). Схемы: ● – CROS, ○ – OIRK2, ■ – CROS4, □ – OIRK3, △ – OIRK4. 51
4.11 Электрическая схема RLC-контура с активным элементом. 52

4.12 Фазовый портрет системы (4.17) при $\sigma = 100$. 53
4.13 Задача (4.17) для $\sigma = 100$. Маркеры черные для аргумента t и светлые для аргумента l; квадратики для ERK2 и ERK4, кружки для CROS и CROS4, треугольники для схемы BMP. 54
4.14 Задача (4.17), схема $ERK4$; ■ – расчеты по t, □ – расчеты по l. Около линий указаны значения σ. 55
4.15 Изменение шага $\tau(t)$ при интегрировании по аргументу l для задачи (4.17), схема ERK4; ∘ – зависимость $\tau(t)$, сплошная линия – компонента решения $|v(t)|$. 56
4.16 Участок решения задачи (4.21), вычисленной с помощью неявной схемы Розенброка и 64-битовыми числами при $l > 1$. ∘ – численное решение, жирная линия – точное решение. 58
4.17 Графики некоторых концентраций задачи (4.24). 61
4.18 Погрешность расчета задачи (4.24) по схеме CROS на серии сгущающихся сеток. • - интегрирование по времени; ∘ - интегрирование по длине дуги. 63

Список таблиц

1.1 Коэффициенты оптимальных явных схем РК (1.2). 14
1.2 Коэффициенты оптимальных обратных схем (1.3) 2–4 порядка точности в форме Бутчера. 16
1.3 Коэффициенты схемы (1.8) в форме Бутчера. 17

4.1 Среднее число классических и усеченных ньютоновских итераций до сходимости в зависимости от соотношения τ/h для неявных схем. 47
4.2 Среднее число усеченных ньютоновских итераций при решении задачи (4.11) с помощью оптимальных обратных схем порядка p на серии сгущающихся сеток с числом узлов N. 49
4.3 Вспомогательные переменные задачи (4.24). 60
4.4 Константы реакций задачи (4.24). 61

Литература

1. Самарский А.А. Тихонов А.Н. Уравнения математической физики. Изд-во МГУ, М., 1999.
2. Самарский А.А. Соболь И.М. Примеры численного расчета температурных волн. // ЖВМиМФ. 1963. Т. 3, № 4. С. 702–719.
3. Хайрер Э. Нерсетт С. Ваннер Г. Решение обыкновенных дифференциальных уравнений. Нежесткие задачи. Мир, 1990. С. 512. 512 с.
4. F. Mazzia C. Magherini. Test Set for Initial Value Problem Solvers, release 2.4. // Department of Mathematics, University of Bari and INdAM, Research Unit of Bari. 2008.
5. Хайрер Э. Ваннер Г. Решение обыкновенных дифференциальных уравнений. Жесткие и дифференциально-алгебраические задачи. Мир, 1999. С. 685.
6. Марчук Г.И. Шайдуров В.В. Повышение точности решений разностных схем. Наука, 1979.
7. Калиткин Н.Н. Альшин А.Б. Альшина Е.А. Рогов Б.В. Вычисления на квазиравномерных сетках. Физматлит, 2005. 224 с.
8. Лимонов А. Г. Разработка двухстадийных схем Розенброка с комплексными коэффициентами и их применение в задачах моделирования образования периодических наноструктур // Диссертация кандидата физико-математических наук. Москва, 2009.
9. П.Д. Ширков. Оптимально затухающие схемы с комплексными коэффициентами для жестких систем ОДУ. // Матем. моделирование. 1992. Т. 4, № 8. С. 47–57.
10. Альшин А.Б. Альшина Е.А. Лимонов А.Г. Двухстадийные комплексные схемы Розенброка для жестких систем. // ЖВМиМФ. 2009. Т. 49, № 2. С. 270–287.
11. А. Б. Альшин Е. А. Альшина. Об одной новой двухстадийной схеме Розенброка для дифференциально-алгебраических задач // Матем. моделирование. 2011. Т. 23, № 3. С. 139–160.
12. Филиппов С. С. АБС-схемы для жестких систем обыкновенных дифференциальных уравнений // Докл. РАН. 2004. Т. 399, № 2. С. 170–172.
13. М. В. Булатов А. В. Тыглиян С. С. Филиппов. Об одном классе одношаговых одностадийных методов для жестких систем обыкновенных дифференциальных уравнений // Ж. вычисл. матем. и матем. физ. 2011. Т. 51, № 7. С. 1251–1265.

14. Ширков П. Д. Устойчивость ROW методов для неавтономных систем обыкновенных дифференциальных уравнений // Матем. моделирование. 2012. Т. 24, № 5. С. 97–111.

15. А. М. Зубанов П. Д. Ширков. Численное исследование одношаговых явно-неявных методов, L-эквивалентных жестко точным двухстадийным схемам Рунге–Кутты // Матем. моделирование. 2012. Т. 24, № 12. С. 129–136.

16. Е. А. Новиков Ю. А. Шитов Ю. И. Шокин. Одношаговые безытерационные методы решения жестких систем // Докл. АН СССР. 1988. Т. 301, № 6. С. 1310–1314.

17. А. Л. Двинский Е. А. Новиков. Аппроксимация матрицы Якоби в $(m, 3)$-методах решения жестких систем // Сиб. журн. вычисл. матем. 2008. Т. 11, № 3. С. 283–295.

18. Новиков Е. А. L-устойчивый (4,2)-метод четвертого порядка для решения жестких задач // Вестн. СамГУ. Естественнонаучн. сер. 2011. № 8(89). С. 59–68.

19. Скворцов Л. М. Явный многошаговый метод численного решения жестких дифференциальных уравнений // Ж. вычисл. матем. и матем. физ. 2007. Т. 47, № 6. С. 959–967.

20. Скворцов Л. М. Простые явные методы численного решения жестких обыкновенных дифференциальных уравнений // Вычислительные методы и программирование. 2008. Т. 9, № 6. С. 154–162.

21. Скворцов Л. М. Явные адаптивные методы Рунге–Кутты // Матем. моделирование. 2011. Т. 23, № 7. С. 73–87.

22. Альшина Е. А. Закс Е. М. Калиткин Н. Н. Оптимальные параметры явных схем Рунге–Кутты невысоких порядков. // Матем. моделирование. 2006. Т. 18, № 2. С. 61–71.

23. Alexander R. Diagonally implicit Runge-Kutta methods for stiff ODEs // SIAM J. Numer. Anal. 1977. Т. 14, № 6. С. 1006–1021.

24. Скворцов Л. М. Диагонально-неявные методы Рунге-Кутты для жестких задач // Ж. вычисл. матем. и матем. физ. 2006. Т. 46, № 12. С. 2209–2222.

25. Скворцов Л. М. Экономичная схема реализации неявных методов Рунге-Кутты // Ж. вычисл. матем. и матем. физ. 2008. Т. 48, № 11. С. 2008–2018.

26. Скворцов Л. М. Эффективная реализация неявных методов Рунге–Кутты второго порядка // Матем. моделирование. 2013. Т. 25, № 5. С. 15–28.

27. Куликов Г. Ю. Теоремы сходимости для итеративных методов Рунге-Кутты с постоянным шагом интегрирования // Ж. вычисл. матем. и матем. физ. 1996. Т. 36, № 8. С. 73–89.

28. Куликов Г. Ю. Численное решение задачи Коши для системы дифференциально-алгебраических уравнений с помощью неявных методов Рунге-Кутты с нетривиальным прогнозом // Ж. вычисл. матем. и матем. физ. 1998. Т. 38, № 1. С. 68–84.

29. Г. Ю. Куликов Е. Ю. Хрусталева. Об автоматическом управлении размером шага и порядком в неявных одношаговых экстраполяционных методах // Ж. вычисл. матем. и матем. физ. 2008. Т. 48, № 9. С. 1580–1606.

30. Г. Ю. Куликов Е. Б. Кузнецов Е. Ю. Хрусталева. О контроле глобальной ошибки в неявных гнездовых методах Рунге-Кутты гауссовского типа // Сиб. журн. вычисл. математики. 2011. Т. 14, № 3. С. 245–259.

31. Новиков Е. А. Оценка глобальной ошибки одношаговых методов решения жестких задач // Изв. вузов. Матем. 2011. № 6. С. 80–89.

32. Г.М. Хаммуд. Трехмерное семейство 7-шаговых методов Рунге-Кутта порядка 6. // Вычисл. методы и программирование. 2001. Т. 2, № 2. С. 71–78.

33. Альшина Е. А. Закс Е. М. Калиткин Н. Н. Оптимальные схемы Рунге–Кутты с первого по шестой порядок точности. // ЖВМиМФ. 2008. Т. 48, № 3. С. 418–429.

34. Cash J. R. A Class of Implicit Runge-Kutta Methods for the Numerical Integration of Stiff Ordinary Differential Equations // Journal of the ACM. 1975. Т. 22, № 4. С. 504–511.

35. J. R. Cash A. Singhal. Mono-implicit Runge-Kutta Formulae for the Numerical Integration of Stiff Differential Systems // IMA J. Numer. Anal. 1982. Т. 2, № 2. С. 211–227.

36. G. Yu. Kulikov S. K. Shindin. On a family of cheap symmetric one-step methods of order four // Computational Science – ICCS 2006. 6th International Conference, Reading, UK, May 28–31. 2006. Т. 3991. С. 781–785.

37. В.И. Косарев. 12 лекций по вычислительной математике (вводный курс). М.: Изд-во МФТИ, 2000. 224 с.

38. Калиткин Н.Н. Кузьмина Л.В. Интегрирование жестких систем дифференциальных уравнений. // Препринт ИПМ им. М.В. Келдыша. 1991. Т. 1, № 80. С. 61–71. 23 стр.

39. П.Д. Ширков. *L*-устойчивость диагонально-неявных схем Рунге-Кутты и методов Розенброка. // ЖВМиМФ. 1992. Т. 32, № 9. С. 1422–1432.

40. Н. Н. Калиткин И. П. Пошивайло. О вычислении простых и кратных корней нелинейного уравнения. // Матем. моделирование. 2008. Т. 20, № 7. С. 57–64.

41. Альшин А.Б. Альшина Е.А. Калиткин Н.Н. Корягина А.Б. Численное решение сверхжестких дифференциально-алгебраических систем // Докл. РАН. 2006. Т. 408, № 4. С. 1–5.

42. Рябенький В.С. Филиппов А.Ф. Об устойчивости разностных уравнений. М.: Государственное изд-во технико-теоретической литературы, 1956. 172 с.

43. E. Riks. The application of Newton’s method to the problem of elastic stability. // Journal of Applied Mechanics. 1972. Т. 39, № 4. С. 1060–1065.

44. Шалашилин В.И. Кузнецов Е.Б. Метод продолжения решения по параметру и наилучшая параметризация. Эдиториал УРСС, 1999. 224 с.

45. Wu Jike W. H. Hui Ding Hongli. Arc-length method for differential equations. // Applied Mathematics and Mechanics. 1999. Т. 20, № 8. С. 936–942.

46. В. Я. Гольдин Н. Н. Калиткин. Нахождение знакопостоянных решений обыкновенных дифференциальных уравнений // Ж. вычисл. матем. и матем. физ. 1966. Т. 6, № 1. С. 162–163.

47. Калиткин Н.Н. Пошивайло И.П. Обратные Ls-устойчивые схемы Рунге-Кутты. // Доклады Академии Наук. 2012. Т. 442, № 2. С. 175–180.

48. Gilbert W. J. Newton's method for multiple roots // Comput. and Graphics. 1994. Т. 18, № 2. С. 227–229.

49. Е. П. Жидков И. В. Пузынин. Об одном методе введения параметра при решении краевых задач для нелинейных обыкновенных дифференциальных уравнениий второго порядка // Ж. вычисл. матем. и матем. физ. 1967. Т. 7, № 5. С. 1086–1095.

50. В. В. Ермаков Н. Н. Калиткин. Оптимальный шаг и регуляризация метода Ньютона // Ж. вычисл. матем. и матем. физ. 1981. Т. 21, № 2. С. 491–497.

51. Гавурин М. К. Нелинейные функциональные уравнения и непрерывные аналоги итеративных методов // Изв. вузов. Матем. 1958. № 5. С. 18–31.

52. Н. Н. Калиткин Л. В. Кузьмина. Вычисление корней уравнения и определение их кратности // Матем. моделирование. 2010. Т. 22, № 7. С. 33–52.

53. Н. Н. Калиткин Л. В. Кузьмина. Прецизионное вычисление кратных корней методом секущих с экстраполяцией // Матем. моделирование. 2011. Т. 23, № 6. С. 33–58.

54. Murray W. Newton-Type Methods // Wiley Encyclopedia of Operations Research and Management Science. 2011.

55. Zeng Z. Computing multiple roots of inexact polynomials // Math. Comput. 2005. Т. 74, № 250. С. 869–903.

56. McNamee J. M. Numerical Methods for Roots of Polynomials, Part I. Studies in Computational Mathematics, Vol.14, 2007. 354 p.

57. A. Galantai C. J. Hegedus. A study of accelerated Newton methods for multiple polynomial roots // Numer. Algor. 2010. Т. 54, № 2. С. 219–243.

58. Островский А. М. Решение уравнений и систем уравнений. Издательство иностранной литературы, 1963. 219 с.

59. http://www.exponenta.ru/educat/class/courses/vvm/theme_3/theory.asp. Применение метода Ньютона для нахождения кратного корня.

60. Verwer J. G. Gauss-Seidel iteration for sti ODEs from chemical kinetics. // SIAM J. Sci.Comput. 1994. Т. 15, № 5. С. 1243–1259.

61. Калиткин Н.Н Ритус И.В. Комплексная схема решения параболических систем. // Препринт ИПМ им. М.В. Келдыша. 1981. Т. 1, № 80. 23 стр.

62. H.H. Rosenbrock. Some general implicit processes for the numerical solution of differential equations // Comput. J. 1963. Т. 5, № 4. С. 329–330.