Содержание

Предисловие ... 7

▼ Глава 1

Задача Коши и методы ее решения ... 11

1.1. Обыкновенные дифференциальные уравнения ... 11
1.2. Точность и устойчивость численных методов ... 12
1.3. Жесткие задачи ... 15
1.4. Меры жесткости, колебательности и неустойчивости задачи Коши ... 17
1.5. Колебательные задачи ... 21
1.6. Плохо обусловленные задачи ... 25
1.7. Задачи с разрывами ... 28
1.8. Одношаговые методы Рунге–Кутты ... 29
1.9. Многошаговые методы ... 31
1.10. Явные методы для жестких задач ... 32
1.11. Дифференциально-алгебраические уравнения ... 34

▼ Глава 2

Явные методы Рунге–Кутты для нежестких задач ... 37

2.1. Условия порядка и коэффициенты погрешности ... 37
2.2. Требования к параметрам методов ... 40
2.3. Управление размером шага ... 42
2.4. Методы 1-го и 2-го порядков ... 44
2.5. Методы 3-го порядка ... 47
2.6. Методы 4-го порядка ... 48
2.7. Методы 5-го порядка ... 51
2.8. Тестовое сравнение методов ... 53
2.9. Решение задач с разрывами ... 56

▼ Глава 3

Неявные методы Рунге–Кутты и Розенброка 2-го порядка 59

3.1. Методы и их свойства 59
3.2. Схемы реализации 63
3.3. Метод трапеций 67
3.4. Метод TR-BDF2 68
3.5. Метод Лобатто IIIC 70
3.6. Численные эксперименты 71
3.7. Методы типа Розенброка 75
3.8. Схемы решения дифференциально-алгебраических уравнений 79

▼ Глава 4

Сходимость методов Рунге–Кутты при решении жестких и дифференциально-алгебраических задач 83

4.1. Сводка результатов о сходимости 83
4.2. Феномен снижения порядка 86
4.3. Сходимость явных методов при решении жестких задач 91
4.4. Неявные методы, обратные к явным методам 94
4.5. Модельные уравнения для нежестких задач 97
4.6. Модельные уравнения для ДАУ индекса 1 99
4.7. Жесткие модельные уравнения 101
4.8. Функции погрешности и псевдостадийный порядок 102
4.9. Модельные уравнения для ДАУ индекса 2 105
4.10. Модельные уравнения для ДАУ индекса 3 109

▼ Глава 5

Диагонально-неявные методы Рунге–Кутты 113

5.1. Функция устойчивости 113
5.2. Функции погрешности 116
5.3. Условия порядка 119
5.4. Методы 3-го порядка 121
5.5. Методы 4-го порядка 122
5.6. Методы 5-го порядка 127
5.7. Методы ESDIRK 3-го псевдостадийного порядка 129

5.8. Двухшаговые диагонально-неявные методы 132
5.9. Диагонально расширенные однократно неявные методы 135
5.10. Реализация методов ESDIRK 137
5.11. Реализация методов DESI 142
5.12. Изменение размера шага и обновление матрицы Якоби 143
5.13. Численные эксперименты 144

▼ Глава 6

Неявные методы повышенной точности для жестких задач и ДАУ 147

6.1. Коллокационные методы Рунге–Кутты для жестких задач 147
6.2. Коллокационные методы Рунге–Кутты для ДАУ индексов 2 и 3 150
6.3. Неявные методы Рунге–Кутты с явными внутренними стадиями 155
6.4. Неявный двухшаговый метод пятого порядка для жестких задач и ДАУ 162

▼ Глава 7

Явные методы с расширенными областями устойчивости 169

7.1. Явные стабилизированные методы Рунге–Кутты 169
7.2. Многочлены устойчивости 170
7.3. Построение стабилизированных методов Рунге–Кутты 2-го порядка 174
7.4. Упорядочение внутренних шагов (стадий) 177
7.5. Стабилизированные методы порядков 3 и 4 181
7.6. Двухшаговые стабилизированные методы 1-го порядка 182
7.7. Трехшаговый стабилизированный метод 2-го порядка 186
7.8. Оценивание границы жесткого спектра 189
7.9. Численные эксперименты 191

▼ Глава 8

Явные адаптивные методы для жестких и колебательных задач 194

8.1. Построение явных адаптивных методов Рунге–Кутты 194
8.2. Сходимость адаптивных методов 198
8.3. Адаптивный метод порядка 2 для нежестких и 1 для жестких задач 201

8.4. Адаптивные методы Рунге–Кутты порядков 2 и 3 205
8.5. Методы с покомпонентным оцениванием двух собственных значений 207
8.6. Построение многошаговых адаптивных методов 209
8.7. Двухшаговый адаптивный метод 213
8.8. Многошаговый адаптивный метод переменного порядка и шага 214
8.9. Численные эксперименты 217

Литература 221

Предисловие

На современном этапе развития цивилизации прогресс во многих областях науки и техники определяется степенью внедрения в научно-технические разработки математического и имитационного моделирования. Замена натурных экспериментов компьютерным моделированием существенно удешевляет и ускоряет научные исследования, а также позволяет избежать трагических ошибок, вызванных критическими состояниями человеческого организма, технических объектов, окружающей среды.

Многие процессы в природе и технике описываются обыкновенными дифференциальными уравнениями (ОДУ) и дифференциальными уравнениями в частных производных. Лишь в редких случаях такие уравнения имеют аналитическое решение, поэтому приходится решать их численно. Уравнения в частных производных можно привести к системе ОДУ, применив метод прямых (method of lines), т. е. заменив пространственные производные конечными разностями. Переменные, входящие в систему ОДУ, могут быть связаны некоторыми алгебраическими соотношениями, в этом случае получаем систему дифференциально-алгебраических уравнений (ДАУ).

Таким образом, многие явления и процессы в физике, химии, астрономии, биологии, технике могут быть описаны в виде системы ОДУ или ДАУ, а моделирование этих процессов сводится к численному решению таких систем. Поэтому очевидна важность построения и реализации в виде компьютерных программ эффективных методов численного решения ОДУ и ДАУ. Компьютерные программы – решатели ОДУ и ДАУ – рассматривались в [4, 74, 75, 128]. Такие программы удобно применять, если математическая модель задана непосредственно в виде системы уравнений. Однако в различных предметных областях применяют также и другие способы представления математической модели: структурные схемы систем автоматического управления, электрические схемы в электротехнике и электронике, кинематические схемы в механике и робототехнике и т. д. Для удобства моделирования таких систем программы снабжают специализированным интерфейсом, позволяющим пользователю задавать модель в удобном виде.

Наиболее известным и популярным программным средством моделирования разнородных (т. е. содержащих элементы разной физической природы) динамических систем является пакет Simulink, входящий в систему

математических вычислений MATLAB. Среди аналогичных отечественных разработок особого внимания заслуживает программное обеспечение (ПО) «Среда динамического моделирования технических систем SimInTech». Далее будем использовать название ПО SimInTech или SimInTech (сокращение от Simulation In Technic). ПО SimInTech является результатом модернизации программного комплекса «Моделирование в технических устройствах» (ПК МВТУ) [24–26], который был разработан коллективом ученых и выпускников МГТУ им. Н. Э. Баумана под руководством О. С. Козлова. Разработка, а также дальнейшее развитие, сопровождение и распространение ПО SimInTech выполняются специалистами ООО «3В Сервис» (www.3v-services.com). Ознакомиться с ПО SimInTech можно в [22] и на сайте http://simintech.ru.

Автор этой книги участвовал в разработке ПК МВТУ и ПО SimInTech и имеет большой опыт по реализации самых различных алгоритмов. Возможность включить свои алгоритмы в современный программный продукт стала серьезным стимулом для исследований в области численного решения ОДУ и ДАУ. В настоящее время SimInTech имеет обширный набор методов решения ОДУ и ДАУ, содержащий два классических явных метода (Рунге–Кутты и Мерсона) и их модификации, пять явных адаптивных методов, диагонально-неявные методы Рунге–Кутты 2-го, 3-го и 4-го порядков, неявные методы Гира и Эйлера. Почти все методы (за исключением классических) являются оригинальными либо содержат оригинальные решения. Особый интерес представляют явные адаптивные методы, которые позволяют эффективно решать многие жесткие системы. Наряду с методами, реализованными в SimInTech и показавшими высокую эффективность при решении множества прикладных задач, в книге рассмотрены новые перспективные методы. Уделено внимание эффективной реализации методов и тестовому сравнению с известными решателями, среди которых решатели системы MATLAB+Simulink и RADAU5.

Глава 1 является вводной, в ней даны постановки задачи Коши для систем ОДУ и ДАУ, рассмотрены различные классы задач и методов их решения. К трудным для численного решения отнесены жесткие, колебательные и плохо обусловленные задачи, задачи с разрывами и ДАУ высших индексов. Предложены количественные меры жесткости, колебательности и неустойчивости задачи Коши, приведены значения этих мер для известных тестовых задач.

В главе 2 рассмотрены явные методы Рунге–Кутты для нежестких задач. Приведены условия порядка до 5-го включительно и даны рекомендации по выбору оптимальных коэффициентов. Рассмотрены два способа построения вложенных пар методов с оцениванием ошибки. Приведены коэффициенты известных и новых вложенных пар до 5-го порядка, а также результаты их тестового сравнения. Рассмотрен эффективный способ решения задач с разрывами.

В главе 3 рассмотрены неявные одношаговые методы низкой точности. Для реализации выбраны три метода 2-го порядка: трапеций, TR-BDF2 и Лобатто IIIC. На примере метода трапеций рассмотрены четыре схемы реализации не-

явных методов. Представлены детальные схемы реализации выбранных методов и новые схемы типа Розенброка. Приведены результаты их тестового сравнения с решателями MATLAB.

Глава 4 содержит теоретические и экспериментальные результаты о сходимости методов Рунге–Кутты при решении жестких и дифференциально-алгебраических задач. Предложены простейшие модельные уравнения, объясняющие снижение точности и порядка при решении таких задач. Получены выражения для ошибок решения модельных уравнений и показано, что минимизация этих ошибок позволяет построить методы повышенной точности, свободные от снижения порядка.

В главе 5 рассмотрены диагонально-неявные методы Рунге–Кутты порядков 3, 4 и 5. Получены упрощенные условия порядка, а также функции погрешности, описывающие поведение жестких составляющих ошибки. Построены конкретные методы с минимизированными функциями погрешности. Рассмотрены схемы реализации и приведены результаты решения тестовых задач в сравнении с решателем RADAU5.

В главе 6 рассмотрены неявные методы, обладающие повышенной точностью при решении жестких задач и ДАУ. К ним относятся коллокационные методы, узлы которых выбраны из условия минимизации ошибок решения модельных уравнений, неявные методы с явными внутренними стадиями, а также двухшаговый метод 5-го порядка, который не снижает точности и порядка при решении жестких задач и ДАУ индексов 2 и 3. Приведены результаты тестового сравнения с методами Радо IIA и Лобатто IIIA.

В главе 7 рассмотрены явные одношаговые и многошаговые методы с расширенными областями устойчивости, позволяющие эффективно решать жесткие задачи с распределенным вещественным спектром матрицы Якоби. Предложен простой и эффективный способ расчета «почти оптимальных» многочленов устойчивости произвольной степени. Рассмотрены способы построения методов Рунге–Кутты с заданным многочленом устойчивости и вложенной формулой для оценивания ошибки. Построены методы порядков 2, 3 и 4; приведены результаты решения тестовых задач (в том числе и в сравнении с решателями RKC, DUMKA3, ROCK4).

В главе 8 рассмотрены явные адаптивные методы, использующие полученные на основе предварительных стадий покомпонентные оценки наибольшего по модулю собственного значения матрицы Якоби для настройки формулы интегрирования на решаемую задачу. Приведены расчетные схемы одношаговых методов порядков 1, 2, 3 и многошагового метода переменного порядка. Показано, что такие методы могут быть эффективными для решения жестких и колебательных задач. Приведены результаты численных экспериментов, которые показали, что при решении многих жестких задач явные адаптивные методы не уступают неявным методам, а иногда и превосходят их.

В качестве инструментов для исследования методов численного решения ОДУ и ДАУ автор использовал алгоритмы, реализованные в ПК МВТУ, ПО

SimInTech, а также в системе компьютерных вычислений MathCAD. Такие программные инструменты заметно сокращают объем рутинной работы по построению, реализации и тестированию новых методов. Некоторые из этих программ, а также некоторые дополнительные материалы по численному решению ОДУ и ДАУ размещены на сайте ООО «3В Сервис» (http://3v-services.com/books/978-5-97060-636-0/).

Автор благодарен коллективу ООО «3В Сервис» за помощь и содействие в издании книги.

Задача Коши и методы ее решения

ГЛАВА 1

1.1. Обыкновенные дифференциальные уравнения

Рассмотрим задачу Коши для системы ОДУ

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(0) = \mathbf{y}_0, \quad 0 \leqslant t \leqslant T, \tag{1.1}$$

где t – независимая переменная, $\mathbf{y} = (y_1, \ldots, y_n)^{\mathrm{T}}$ – вектор переменных состояния, $\mathbf{f}(t, \mathbf{y}) = (f_1(t, \mathbf{y}), \ldots, f_n(t, \mathbf{y}))^{\mathrm{T}}$ – нелинейная векторная функция. Если решается задача моделирования во времени, то t – модельное время. Система ОДУ называется *автономной*, если правая часть не зависит от t, т. е. $\mathbf{f}(t, \mathbf{y}) = \mathbf{f}(\mathbf{y})$. Неавтономную систему (1.1) нетрудно привести к автономной, добавив уравнение $t' = 1$. Поэтому все теоретические результаты, полученные для автономных систем, справедливы также и для неавтономных систем.

Численное решение (интегрирование) задачи (1.1) сводится к нахождению последовательности векторов $\mathbf{y}_1, \ldots, \mathbf{y}_N$, аппроксимирующих истинное решение $\mathbf{y}(t)$ в дискретные моменты модельного времени $t_1, \ldots, t_N = T$. Интервал между двумя соседними моментами времени $h_i = t_{i+1} - t_i$ называется шагом интегрирования (размером шага). Размер шага может быть постоянным ($h_i = h = \mathrm{const}$) либо переменным.

В дальнейшем будем предполагать, что задача (1.1) имеет единственное решение, а функция $\mathbf{f}$ – гладкая в любой точке решения на интервале интегрирования $[0, T]$. Тогда на всем интервале определена и непрерывна матрица Якоби системы (1.1)

$$\mathbf{J}(t) = \frac{\partial \mathbf{f}(t, \mathbf{y}(t))}{\partial \mathbf{y}}. \tag{1.2}$$

Требование гладкости правой части не всегда согласуется с реальными моделями, в составе которых могут быть различные релейные и переключательные элементы. При наличии таких элементов будем предполагать, что число переключений конечно, а весь интервал интегрирования можно разбить на несколько интервалов, на каждом из которых функция $\mathbf{f}$ остается гладкой. Тогда решения «склеиваются», т. е. на каждом последующем интервале в качестве начального условия принимается решение, полученное в конце текущего ин-

тервала. Таким образом, и в этом случае можно считать функцию **f** гладкой, а якобиан (1.2) – непрерывным.

Простейшим методом численного интегрирования является метод Эйлера, формула которого при решении задачи (1.1) имеет вид

$$\mathbf{y}_{i+1} = \mathbf{y}_i + h\mathbf{f}(t_i, \mathbf{y}_i). \tag{1.3}$$

Этот метод является *явным*, поскольку вектор переменных в очередной момент модельного времени явно выражается через уже рассчитанный вектор в предыдущий момент. Для решения ОДУ применяют также *неявные* методы, простейший из них – неявный (обратный) метод Эйлера, формула которого

$$\mathbf{y}_{i+1} = \mathbf{y}_i + h\mathbf{f}(t_{i+1}, \mathbf{y}_{i+1}). \tag{1.4}$$

В неявных методах формула шага интегрирования представляет собой систему нелинейных алгебраических уравнений, для решения которой используют итерационные методы (обычно это метод Ньютона или его модификации).

Методы решения ОДУ подразделяются также на *одношаговые* (Эйлера, Рунге–Кутты, Розенброка) и *многошаговые* (Адамса, Гира, прогноза-коррекции и др.). В одношаговых методах для нахождения $\mathbf{y}_{i+1}$ используются только векторы $\mathbf{y}_i$ и $\mathbf{y}'_i = \mathbf{f}(t_i, \mathbf{y}_i)$. В многошаговых методах используется также информация, полученная на предыдущих шагах: в k-шаговом методе это $\mathbf{y}_{i-1}$, …, $\mathbf{y}_{i-k+1}$, $\mathbf{y}'_{i-1}$, …, $\mathbf{y}'_{i-k+1}$. Первый шаг всегда выполняется одношаговым методом. На последующих шагах может быть произведен переход на многошаговый метод с последовательным увеличением числа используемых шагов. Не следует продолжать решение многошаговым методом при резком изменении правой части системы ОДУ, поскольку накопленная информация оказывается устаревшей. В этом случае следует отбросить всю предыдущую информацию и вновь начать решение одношаговым методом.

1.2. Точность и устойчивость численных методов

Основные характеристики методов численного решения ОДУ связаны с их точностью и устойчивостью. Размер шага выбирается исходя из точности численного решения. Ошибка интегрирования

$$\mathbf{e}_i = \mathbf{y}(t_i) - \mathbf{y}_i \tag{1.5}$$

складывается из двух составляющих: методической (или ошибки дискретизации), обусловленной неточностью метода, и ошибки округления, обусловленной ограниченностью разрядной сетки компьютера. При уменьшении размера шага методическая ошибка уменьшается, а ошибка округления возрастает. Ошибка округления обычно пренебрежимо мала и заметно сказывается лишь в некоторых исключительных случаях.

При точном выполнении всех вычислений ошибка состоит только из методической составляющей. При заданном начальном условии ошибка (1.5) называется *глобальной*, поскольку она получена в результате накопления ошибок на

всех предыдущих шагах. Ошибка, полученная на одном шаге при предположении, что все используемые предыдущие значения точные, называется *локальной*. Для одношаговых методов ошибка (1.5) будет локальной, если $\mathbf{y}_{i-1} = \mathbf{y}(t_{i-1})$. При некоторых (достаточно общих) предположениях локальная ошибка при $h \to 0$ пропорциональна h^{p+1}, где целое число p называется *порядком сходимости* метода. Глобальная ошибка получается в результате накопления локальных ошибок на всех шагах, поэтому она пропорциональна числу шагов $N = T/h$ и усредненной локальной ошибке. В результате при $h \to 0$ глобальная ошибка пропорциональна h^p. Для явного и неявного методов Эйлера $p = 1$, поэтому уменьшение размера шага в 2 раза приводит к уменьшению локальной ошибки примерно в $2^{p+1} = 4$ раза. Но при этом в 2 раза увеличивается число шагов, поэтому глобальная ошибка уменьшится только в $2^p = 2$ раза.

Порядок используемого метода следует соотносить с требуемой точностью численного решения. Чтобы убедиться в этом, рассмотрим задачу

$$y_1' = -22y_1 + 20y_2^2,\ y_2' = y_1 - y_2 - y_2^2,\quad y_1(0) = 1,\ y_2(0) = 1,\ 0 \leqslant t \leqslant 1, \tag{1.6}$$

решение которой $y_1(t) = \exp(-2t)$, $y_2(t) = \exp(-t)$. Ошибку решения оценим в виде $\varepsilon = \max(e(t_i),\ 0 \le t_i \le 1)$, где $e(t_i)$ – евклидова норма абсолютной ошибки в точке $t_i = ih$. Вычислительные затраты оценим числом вычислений правой части Nf на всем интервале. Зависимости ошибки от вычислительных затрат для явных одношаговых методов 1-го, 2-го и 4-го порядков приведены на рис. 1.1. Из этого рисунка видно, что выбор порядка метода определяется требованиями к точности. Если допустима достаточно большая ошибка, преимущество имеют методы невысоких порядков, позволяющие получить решение с малыми вычислительными затратами. А при малой допустимой ошибке следует использовать методы более высоких порядков.

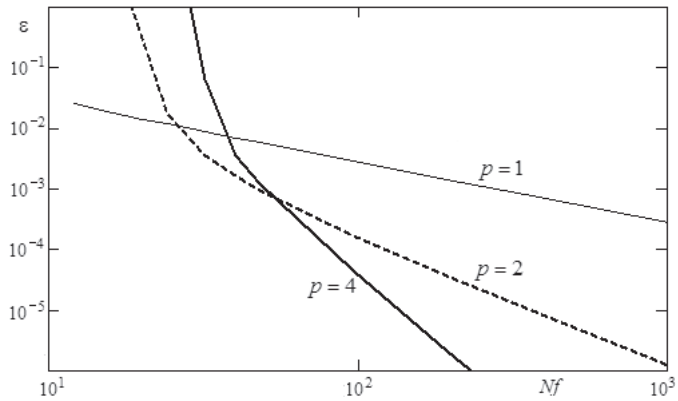

Рис. 1.1. Зависимости ошибки решения задачи (1.6) от вычислительных затрат для методов порядков 1, 2 и 4

При численном решении дифференциальные уравнения заменяются разностными. Решение полученных разностных уравнений может оказаться неустойчивым, хотя исходная система ОДУ была устойчивой. Неустойчивость проявляется как катастрофический рост ошибки численного решения при увеличении размера шага. Покажем это на примере задачи

$$y' = 50(e^{-t} - y),\ y_0 = 1,\ 0 \le t \le 3 \tag{1.7}$$

с решением $y(t) = (50/49)e^{-t} - (1/49)e^{-50t}$, которую будем решать методом Эйлера. При размере шага $h < 0.04$ численное решение сходится и почти не отличается от точного решения. Но уже при $h = 3/73 = 0.0411$ получаем быстро расходящееся решение, показанное на рис. 1.2 тонкой линией (толстой линией показано точное решение).

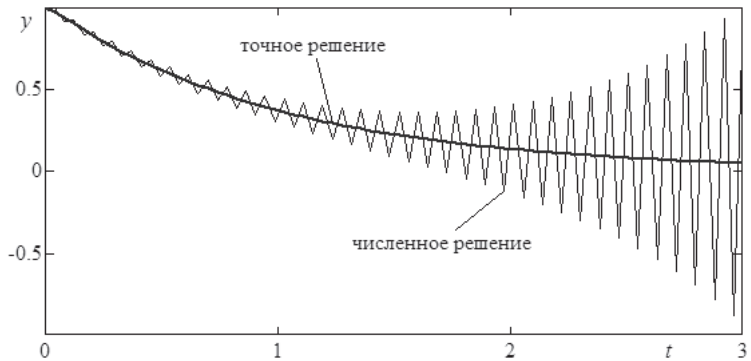

Рис. 1.2. Точное и численное решения задачи (1.7) при $h = 3/73$

Обычно для конкретной задачи и конкретного метода существует некоторое граничное значение шага $h_{\max}$, превышение которого приводит к неустойчивости численного решения. Исследуем устойчивость численных методов решения ОДУ на примере линейной системы

$$\mathbf{y}' = \mathbf{J}\mathbf{y},\ \mathbf{y}(t_0) = \mathbf{y}_0, \tag{1.8}$$

где $\mathbf{J}$ – матрица размером $n \times n$. Система (1.8) устойчива, если все собственные значения матрицы $\mathbf{J}$ имеют отрицательные действительные части. Применим для решения этой системы явный метод Эйлера (1.3). Подставив в (1.3) $\mathbf{y}'_i = \mathbf{J}\mathbf{y}_i$, получим

$$\mathbf{y}_{i+1} = (\mathbf{I} + h\mathbf{J})\mathbf{y}_i, \tag{1.9}$$

где $\mathbf{I}$ – единичная матрица. Мы получили систему линейных разностных уравнений, решение которой аппроксимирует решение исходной системы ОДУ (1.8). Полученная система (1.9) будет устойчивой, если все собственные числа матрицы $\mathbf{I} + h\mathbf{J}$, равные $1 + h\lambda_j$, по абсолютной величине меньше 1 (λ_j – собственные числа матрицы $\mathbf{J}$). Таким образом, условие устойчивости численного решения системы (1.8) методом Эйлера запишется в виде системы неравенств

$$|1 + h\lambda_j| < 1,\ j = 1, \ldots, n. \tag{1.10}$$

Вместо системы (1.8) для исследования устойчивости используют скалярное линейное уравнение (уравнение Далквиста)

$$y' = \lambda y, \tag{1.11}$$

в котором λ может быть комплексным числом. Применение одношагового метода типа Рунге–Кутты для решения этого уравнения приводит к формуле интегрирования $y_{i+1} = R(h\lambda)y_i$, где $R(z)$ называется *функцией устойчивости*. Область, задаваемая неравенством $|R(z)| \leq 1$, называется *областью устойчивости*. Функция устойчивости явного метода Эйлера $R(z) = 1 + z$, а его область устойчивости задается неравенством $|1 + z| \leq 1$ и представляет собой круг единичного радиуса с центром в точке $(-1, 0)$. Функция устойчивости неявного метода Эйлера $R(z) = (1 - z)^{-1}$, а его область устойчивости задается неравенством $|1 - z| \geq 1$. При интегрировании устойчивой линейной системы (1.8) размер шага следует выбирать таким, чтобы все числа $h\lambda_j$ попали в область устойчивости.

Метод называется *A-устойчивым*, если его область устойчивости включает всю левую полуплоскость комплексной плоскости. Метод называется *A*(α)-*устойчивым*, если его область устойчивости включает сектор, задаваемый неравенством $|\arg(-z)| \leq \alpha$. Для методов решения жестких задач часто требуют также выполнения условия $R(\infty) = 0$. *A*- и *A*(α)-устойчивые методы, удовлетворяющие этому условию, называются, соответственно, *L*- и *L*(α)-*устойчивыми*. Приведенные определения распространяются и на многошаговые методы, в этом случае вместо неравенства $|R(z)| \leq 1$ рассматривают аналогичные неравенства для корней характеристического полинома разностной схемы, которые также зависят от $z = h\lambda$.

1.3. Жесткие задачи

Пусть все собственные числа матрицы **J** в системе (1.8) вещественные и отрицательные. Тогда условие (1.10) запишется в виде $h < 2\tau_{\min}$, где минимальная постоянная времени $\tau_{\min} = \min(\tau_i,\ i = 1, \ldots, n)$, $\tau_i = -1/\lambda_i$. Таким образом, для обеспечения устойчивости численного решения явным методом Эйлера размер шага должен быть меньше двух минимальных постоянных времени. Аналогичные условия накладываются на размер шага и при использовании других явных методов. Время переходного процесса в системе (1.8) определяется максимальной постоянной времени $\tau_{\max}$ и составляет примерно $3\tau_{\max}$. При большом разбросе постоянных времени $\tau_{\max}/\tau_{\min}$ число шагов интегрирования оказывается очень большим, что может привести к большим затратам машинного времени. В то же время размер шага неявного метода Эйлера и многих других неявных методов ограничен только требованиями к точности решения и может быть значительно больше $\tau_{\min}$.

Рассмотрим, например, задачу

$$y_1' = -y_1/\tau_1,\quad y_2' = (y_1 - y_2)/\tau_2,\ y_1(0) = y_2(0) = 1, \tag{1.12}$$

имеющую при $\tau_1 \neq \tau_2$ решение

$$y_1 = e^{-t/\tau_1}, \quad y_2 = \frac{\tau_1}{\tau_1 - \tau_2} e^{-t/\tau_1} - \frac{\tau_2}{\tau_1 - \tau_2} e^{-t/\tau_2}.$$

При $\tau_1 >> \tau_2$ компонента решения, соответствующая постоянной τ_2, мала и быстро затухает. Несмотря на это, шаг интегрирования при использовании явных методов следует выбирать малым на всем интервале интегрирования. В некоторых случаях малой постоянной времени можно пренебречь, заменив, например, второе уравнение в (1.12) на равенство $y_2 = y_1$. Но в общем случае подобная замена может привести к появлению алгебраического уравнения вместо дифференциального. К тому же для сложных нелинейных систем выделить в явном виде малую постоянную времени не всегда возможно.

Задачи, подобные рассмотренной выше, получили название *жестких*. Жесткие задачи характеризуются наличием собственных значений матрицы Якоби, имеющих большие отрицательные действительные части. Соответствующие составляющие решения быстро затухают и, за исключением малых участков (пограничных слоев), пренебрежимо малы. Решение жестких задач традиционными явными методами требует больших вычислительных затрат, поэтому для их решения обычно применяют неявные методы, которые обеспечивают устойчивое интегрирование с большим размером шага. Неявные методы не свободны от недостатков, к которым относятся прежде всего сложность реализации и необходимость вычислять матрицу Якоби. В тех случаях, когда правая часть содержит разрывы или логические условия, вычисление якобиана может представлять собой сложную и далеко не тривиальную задачу. Отметим также, что при решении некоторых жестких задач применение неявных методов дает неудовлетворительные, а иногда и качественно неверные результаты (примеры таких задач приведены в разделе 8.9). Поэтому наряду с неявными методами разрабатывают специальные явные методы, пригодные для решения жестких задач.

Жесткие задачи весьма разнообразны, поэтому класс таких задач трудно поддается формальному определению, а среди существующих определений нет общепринятого. Наиболее известно определение Ламберта (см. [72]), согласно которому задача Коши называется жесткой, если на всем интервале интегрирования выполняются условия

$$\mathrm{Re}\,\lambda_i < 0, \quad i = 1, \ldots, n;$$

$$S(t) = \frac{\max(\mathrm{Re}(-\lambda_i), i = 1, \ldots, n)}{\min(\mathrm{Re}(-\lambda_i), i = 1, \ldots, n)} >> 1,$$

где величина $S(t)$ названа локальным коэффициентом жесткости. Часто приводят эквивалентное определение, введя постоянные времени $\tau_i = -1/\mathrm{Re}\,\lambda_i$: задача считается жесткой, если имеет большой разброс постоянных времени. Максимальная постоянная времени τ_{max} определяет длительность переходного процесса. Если решение гладкое и не содержит большого числа колебаний, то размер шага из соображений точности может быть выбран значительно боль-

ше τ_{min}. Но при интегрировании явными методами приходится выбирать шаг из соображений устойчивости, т. е. порядка τ_{min}. Таким образом, разброс постоянных времени характеризует вычислительные затраты при интегрировании задачи явными методами.

Определение Ламберта не охватывает всего класса жестких задач, поскольку оно применимо только к устойчивым системам, при условии что постоянные времени существенно не изменяются на интервале интегрирования. Другой недостаток этого определения виден на примере задачи

$$y' = \lambda(y - \sin t) + \cos t, \quad y(0) = 0,$$

имеющей гладкое решение $y(t) = \sin t$, не зависящее от λ. При больших отрицательных значениях λ для устойчивого решения этой задачи классическими явными методами приходится выбирать очень малый шаг интегрирования. Таким образом, данная задача проявляет свойство жесткости, хотя по определению Ламберта не является таковой, поскольку имеет только одну постоянную времени.

Авторы известной книги по численному решению жестких задач [75] Э. Хайрер и Г. Ваннер полагают, что наиболее практичным определением понятия «жесткий» является самое раннее, данное в 1952 г. Кертиссем и Хиршфельдером [97]: «Жесткие уравнения – это уравнения, для которых определенные неявные методы, в частности ФДН, дают лучший результат, обычно несравненно более хороший, чем явные методы». В настоящее время известны специальные явные методы, эффективные для многих жестких задач, поэтому под явными методами в этом определении следует понимать классические явные методы Рунге–Кутты и Адамса. Отметим, что термины «жесткие уравнения» и «жесткая задача» применимы к конкретной задаче Коши, т. е. при заданных начальных условиях и интервале интегрирования, поскольку задача, жесткая на большом интервале, может оказаться нежесткой на меньшем интервале или при других начальных условиях.

Данное определение дает практический способ оценивания жесткости задачи как отношения затрат явного метода к затратам неявного метода. Количественной мерой затрат на решение может быть машинное время, необходимое для решения задачи. В наше время нетрудно оценить жесткость задачи, поскольку современные системы математических вычислений имеют в своем составе достаточно обширные наборы методов интегрирования, включая неявные методы и классические явные методы. Однако для оценивания жесткости задачи желательно использовать меру вычислительных затрат, не зависящую от используемого метода и его программной реализации.

1.4. Меры жесткости, колебательности и неустойчивости задачи Коши

В качестве меры затрат на решение жесткой задачи явным методом можно использовать приблизительное число вычислений правой части при умеренных требованиях к точности. Для линейной задачи (1.8) это число пропорциональ-

но интервалу интегрирования T и обратно пропорционально минимальной постоянной времени $\tau_{\min}$. Поэтому для линейной задачи вычислительные затраты можно оценить числом

$$M_{\text{ж}} = \mu T,\ \mu = 1/\tau_{\min} = \max_i \operatorname{Re}(-\lambda_i).$$

Если все собственные значения матрицы **J** вещественные и отрицательные, то значение $M_{\text{ж}}$ равно минимальному числу вычислений $\mathbf{f}(t, \mathbf{y})$, необходимому для устойчивого решения задачи (1.8) явным методом Рунге–Кутты 2-го порядка с функцией устойчивости $R(z) = 1 + z + z^2/2$. Такой метод реализован в решателе RK2(1), рассмотренном в разделе (2.4). Для явного метода Эйлера $M_{\text{ж}}$ – минимальное число вычислений правой части при решении задач, удовлетворяющих условиям $\operatorname{Re}\lambda_i < 0$, $|\operatorname{Im}\lambda_i| \le |\operatorname{Re}\lambda_i|$.

В общем случае нелинейной неавтономной задачи (1.1) якобиан (1.2) и его спектр зависят от t, поэтому вычислительные затраты явных методов можно оценить с помощью интеграла

$$M_{\text{ж}} = \int_0^T \max(\max_i \operatorname{Re}(-\lambda_i(t)), 0)dt. \tag{1.13}$$

Величину $M_{\text{ж}}$ назовем *мерой жесткости* задачи Коши. В формуле (1.13) оцениваются вычислительные затраты, вызванные только жесткостью задачи. Реальное число шагов может значительно превышать эти оценки вследствие негладкости правой части, наличия больших собственных значений вблизи мнимой оси или в правой полуплоскости (колебательные и плохо обусловленные задачи), а также по другим причинам.

Трудности, возникающие при решении задачи Коши, в значительной степени определяются спектром матрицы Якоби. В зависимости от расположения наибольших по модулю собственных значений (в левой полуплоскости, вблизи мнимой оси, в правой полуплоскости) можно выделить жесткие, колебательные и плохо обусловленные задачи. Колебательные задачи имеют собственные значения вблизи мнимой оси, а плохо обусловленные – в правой полуплоскости. Для эффективного решения колебательных задач применяют неявные симметричные методы либо специальные методы, в том числе и явные. Отметим, однако, что характер задачи может изменяться в процессе решения, а также может быть разным для разных компонент.

По аналогии с мерой жесткости оценим также колебательность и неустойчивость задачи Коши. Величину

$$M_{\text{к}} = \int_0^T \max_i \operatorname{Im}(\lambda_i(t))dt$$

назовем *мерой колебательности*, а величину

$$M_{\text{ну}} = \int_0^T \max\left(\max_i \operatorname{Re}(\lambda_i(t)), 0\right)dt$$

назовем *мерой неустойчивости* задачи Коши.

На разных участках решения задача может иметь разный характер, поэтому имеет смысл ввести обобщенный показатель, оценивающий трудность решения задачи Коши при использовании классических методов. Такой показатель примем в виде

$$M_\Sigma = \int_0^T \max_i |\lambda_i(t)|\, dt.$$

Конечно, трудность решения задачи зависит также и от многих других причин, но нас сейчас интересуют характеристики задачи, связанные только со спектром матрицы Якоби.

Вычислим определенные выше меры для конкретного примера. Возьмем наиболее распространенный тест – осциллятор Ван-дер-Поля, уравнения которого имеют вид

$$y_1' = y_2,\ y_2' = \mu(1 - y_1^2)y_2 - y_1,$$
$$y_1(0) = 2,\ \ y_2(0) = y_{20},\ \ 0 \le t \le T. \tag{1.14}$$

Здесь T – период предельного цикла, а значение y_{20} выбрано таким, чтобы начальная точка лежала на траектории предельного цикла. Для вычисления этих значений использовался ПК МВТУ [26]. Вычисленные при различных значениях μ характеристики этой задачи приведены в табл. 1.1. Здесь же приведено число вычислений правой части *Nf* при решении задачи с допуском на ошибку *Tol* = 0.01 методом RK2(1). При μ = 0 задача – чисто колебательная. При увеличении μ возрастает жесткость задачи, а также появляется неустойчивая составляющая. При μ > 10 жесткость задачи пропорциональна μ^2, а число вычислений правой части явного метода практически совпадает с мерой жесткости и на порядки больше, чем у неявного метода. При $\mu = 10^3$ решение задачи неявным методом TR-BDF2, рассмотренным в главе 3, потребовало всего 715 вычислений правой части и 10 вычислений матрицы Якоби.

Таблица 1.1. Характеристики уравнения Ван-дер-Поля (1.14) на одном цикле решения

μ	T	$y_2(0)$	$M_ж$	$M_к$	$M_{ну}$	M_Σ	Nf
0	6.28319	0	0	6.28	0	6.28	113
1	6.66329	–0.16898	9.37	4.13	3.28	13.4	182
10	19.0784	–0.0665099	323.5	4.00	12.87	331.7	653
100	162.837	-6.66654×10^{-3}	2.90×10^4	4.02	24.10	2.90×10^4	29498
1000	1614.40	-6.66667×10^{-4}	2.89×10^6	4.02	35.54	2.89×10^6	2887653

В качестве тестовой задачи обычно используется нормированное уравнение Ван-дер-Поля, полученное путем замены переменных

$$y_1(t) = x_1(t/\mu),\ \ \mu y_2(t) = x_2(t/\mu).$$

В результате получаем

$$x_1' = x_2,\ \ x_2' = \mu^2((1 - x_1^2)x_2 - x_1),$$
$$x_1(0) = 2,\ \ x_2(0) = x_{20},\ \ 0 \le t \le T. \tag{1.15}$$

Такую задачу удобнее исследовать потому, что при больших μ период предельного цикла T почти не зависит от μ, а в пределе при $\mu \to \infty$ получаем $T = 3 - 2\ln(2)$.

Интересно посмотреть, как изменяются собственные числа матрицы Якоби на траектории решения. Для уравнения (1.15) при $\mu^2 = 10$ кривые изменения x_1 и собственных чисел приведены на рис. 1.3. При увеличении μ качественная картина сохраняется, но время перехода переменной x_1 от 1 до –2 сокращается, а собственные числа увеличиваются. Такая задача часто используется как тест для методов численного решения жестких ОДУ. Исследуем уравнение (1.15) при $\mu^2 = 10^6$ (такое значение используется в жесткой тестовой задаче). В этом случае получаем $x_{20} = -0.6666665432102$, $T = 1.614401125809$. Разобьем полупериод $T/2$ на 5 интервалов таким образом, чтобы границами интервалов были моменты t_i, в которые переменная x_1 принимает целые значения. В табл. 1.2 приведены значения переменных и собственных чисел λ_1, λ_2 в эти моменты, а также величины интервалов времени.

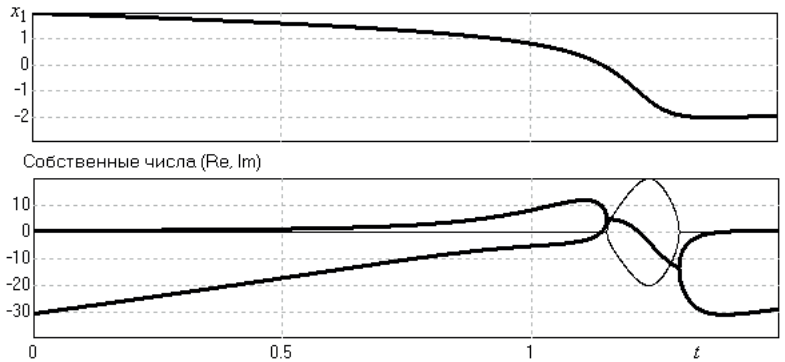

Рис. 1.3. Изменение собственных чисел матрицы Якоби на траектории решения уравнения Ван-дер-Поля (толстые линии – действительные части, тонкие – мнимые)

Таблица 1.2. Собственные числа уравнения Ван-дер-Поля (1.15) при $\mu^2 = 10^6$

i	$x_1(t_i)$	$x_2(t_i)$	λ_1	λ_2	$t_i - t_{i-1}$
0	2	−0.6667	0.5556	-3.000×10^6	– – –
1	1	-1.021×10^2	1.452×10^4	-1.452×10^4	0.80695
2	0	-6.669×10^5	1.000×10^6	1.000	1.323×10^{-4}
3	−1	-1.334×10^6	$j1.633 \times 10^6$	$-j1.633 \times 10^6$	9.619×10^{-7}
4	−2	-2.228×10^2	-2.974×10^2	-3.000×10^6	3.470×10^{-6}
5	−2	0.6667	0.5556	-3.000×10^6	1.117×10^{-4}

Характеристики некоторых известных тестовых задач приведены в табл. 1.3, где первые 6 тестов – нежесткие, остальные 8 – жесткие. При решении обыч-

ными явными методами все жесткие задачи требуют очень больших вычислительных затрат, а решение наиболее жесткой задачи ROBER практически невозможно получить такими методами. Неявные методы успешно и с малыми затратами решают эти задачи (результаты приведены в разделах 3.6 и 5.13). Отметим, что задача BEAM имеет чисто мнимый спектр матрицы Якоби. Поэтому формально ее можно отнести к колебательным задачам, но она проявляет свойство жесткости, поскольку эффективно может быть решена только неявными методами. Для решения задач с вещественным жестким спектром (к ним относятся тесты VDPOL, ROBER, OREGO, HIRES, CUSP и BRUSS) успешно применяют специальные явные методы, рассмотренные в главах 7 и 8.

Таблица 1.3. Характеристики тестовых задач

Задача	Источник	n	T	$M_{ж}$	$M_{к}$	$M_{ну}$	M_{Σ}
JACB	[74]	3	20	5.6	17.1	5.8	17.7
TWOB	[74]	4	20	31.0	21.9	31.0	31.0
VDPL	[74]	2	20	28.1	12.4	9.8	40.2
BRUS	[74]	2	20	205.1	3.7	20.2	227.9
LAGR	[74]	10	10	0	54.7	0	54.7
PLEY	[74, 128]	28	3	40.6	28.3	40.6	40.6
VDPOL	[75, 128]	2	2	3.84×10^{6}	4.0	35.8	3.84×10^{6}
ROBER	[75, 128]	3	10^{11}	10^{15}	0	0	10^{15}
OREGO	[75, 128]	3	360	1.13×10^{7}	1.5	27.1	1.13×10^{7}
HIRES	[75, 128]	8	321.8122	3.44×10^{4}	0.006	0	3.44×10^{4}
PLATE	[75]	80	7	6.96×10^{3}	1.02×10^{4}	0	1.08×10^{4}
BEAM	[75, 128]	80	5	0	3.2×10^{4}	0	3.2×10^{4}
CUSP	[75]	96	1.1	6.87×10^{4}	1.8	21.0	6.87×10^{4}
BRUSS	[75]	1000	10	2×10^{5}	0	0	2×10^{5}

1.5. Колебательные задачи

Колебательные задачи имеют собственные значения матрицы Якоби вблизи мнимой оси, а их решения представляют собой колебательные процессы с медленно изменяющимися амплитудой и частотой. Трудность решения таких задач обусловлена необходимостью обеспечить правильные значения амплитуды и фазы на протяжении многих периодов.

Простейшая колебательная задача имеет вид

$$x' = -\omega y, \quad y' = \omega x, \quad x_0 = 1, \quad y_0 = 0, \quad 0 \le t \le T \tag{1.16}$$

и описывает незатухающие колебания $x(t) = \cos(\omega t)$, $y(t) = \sin(\omega t)$ с амплитудой $A(t) = 1$ и фазой $\varphi(t) = \omega t$. Однако большинство используемых на практике методов дает медленно расходящееся или медленно сходящееся численное решение, фаза которого отстает или опережает фазу точного решения.

Посмотрим, как изменяются амплитуда и фаза при численном решении методом Рунге–Кутты. Представим (1.16) в виде скалярного уравнения

$u' = j\omega u, \quad u = x + jy, \quad u_0 = 1,$

где j – мнимая единица. Обозначим $H = \omega h$, тогда решение на одном шаге методом с функцией устойчивости $R(z)$ получим в виде $u_1 = R(jH)$, а в конце интервала в виде $u_N = R(jH)^N$, где $N = T/h$ – число шагов. Значения амплитуды и фазы численного решения после первого шага:

$$\tilde{A}(h) = \left|R(jH)\right|, \quad \tilde{\varphi}(h) = \arg(R(jH)).$$

Соответствующие локальные ошибки выражаются формулами

$$\delta A(h) = A(h) - \tilde{A}(h) = 1 - \left|R(jH)\right|, \quad \delta\varphi(h) = \varphi(h) - \tilde{\varphi}(h) = H - \arg(R(jH)), \tag{1.17}$$

а глобальные ошибки равны $\Delta A(T) = 1 - |R(jH)|^N$, $\Delta\varphi(T) = N\delta\varphi(T)$.

Разлагая выражения (1.17) в ряд Тейлора, получаем

$$\delta A(h) = C_A H^{q+1} + O(H^{q+3}), \quad \delta\varphi(h) = C_\varphi H^{r+1} + O(H^{r+3}),$$

где C_A и C_φ – коэффициенты ошибки по амплитуде и по фазе. Соответствующие глобальные ошибки пропорциональны H^q и H^r, где q – нечетное число, а r – четное. Значение q называют *порядком диссипативности* (*dissipation order*), а r – *порядком сдвига фазы* (*phase lag order*) [99, 147].

Для методов Рунге–Кутты не ниже 4-го порядка имеем

$$R(z) = 1 + z + z^2/2 + z^3/6 + z^4/24 + a_5 z^5 + \cdots + a_9 z^9 + O(z^{10}),$$

а выражения для локальных ошибок по амплитуде и фазе запишутся в виде

$$\delta A(h) = \left(\frac{1}{144} - a_5 + a_6\right)H^6 + \left(\frac{-1}{1152} + \frac{a_5}{6} - \frac{a_6}{2} + a_7 - a_8\right)H^8 + O(H^{10}),$$

$$\delta\varphi(h) = \left(\frac{1}{120} - a_5\right)H^5 + \left(\frac{-1}{336} + \frac{a_5}{2} - a_6 + a_7\right)H^7 + \left(\frac{1}{5184} - \frac{a_5}{24} + \frac{a_6}{6} - \frac{a_7}{2} + a_8 - a_9\right)H^9 + O(H^{11}).$$

Приведем порядки (классический p, диссипативности q и сдвига фазы r) и коэффициенты C_A и C_φ некоторых известных методов Рунге–Кутты.

Метод Ральстона:

$$R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6}, \quad p = 3, \quad q = 3, \quad r = 4, \quad C_A = \frac{1}{24}, \quad C_\varphi = \frac{-1}{30}.$$

Классический метод Рунге–Кутты:

$$R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24}, \quad p = 4, \quad q = 5, \quad r = 4, \quad C_A = \frac{1}{144}, \quad C_\varphi = \frac{1}{120}.$$

Метод Мерсона:

$$R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24} + \frac{z^5}{144}, \quad p = 4, \quad q = 7, \quad r = 4, \quad C_A = \frac{1}{3456}, \quad C_\varphi = \frac{1}{720}.$$

Метод Дорманда–Принса:

$$R(z)=1+z+\frac{z^2}{2}+\frac{z^3}{6}+\frac{z^4}{24}+\frac{z^5}{120}+\frac{z^6}{600},\quad p=5,\quad q=5,\quad r=6,\quad C_A=\frac{1}{3600},\ C_\varphi=\frac{-1}{2100}.$$

Метод Лобатто IIIA 4-го порядка:

$$R(z)=\frac{1+z/2+z^2/12}{1-z/2+z^2/12},\quad p=4,\quad q=\infty,\quad r=4,\quad C_A=0,\quad C_\varphi=\frac{1}{720}.$$

Метод Радо IIA 5-го порядка:

$$R(z)=\frac{1+(2/5)z+z^2/20}{1-(3/5)z+(3/20)z^2-z^3/6},\quad p=5,\quad q=5,\quad r=6,\quad C_A=\frac{1}{7200},\quad C_\varphi=\frac{1}{42000}.$$

Метод Ральстона был предложен в [136] и вместе с вложенной формулой 2-го порядка образует метод Богацки–Шампайна с автоматическим выбором шага [86], реализованный в решателе ode23 системы MATLAB. Метод Дорманда–Принса также реализован в MATLAB в решателе ode45. Метод Мерсона реализован в одном из явных решателей SimInTech. Методы Лобатто IIIA относятся к симметричным методам [74, 75], для которых $|R(jH)| = 1$, благодаря чему они сохраняют амплитуду колебаний на любом интервале. Такие методы позволяют обеспечить правильный характер огибающей колебательного решения при моделировании высокочастотных колебательных процессов, модулированных по амплитуде. Метод Радо IIA 5-го порядка реализован в одном из наиболее эффективных решателей жестких и дифференциально-алгебраических задач RADAU5 [75]. Специальные явные методы, имеющие повышенную точность при решении колебательных задач и пригодные также и для решения жестких задач, рассмотрены в разделе 8.5.

Чтобы оценить возможности наиболее известных методов при решении колебательных задач, приведем результаты решения трех таких задач. Для их решения используем следующие явные методы: Ralston3 – метод Ральстона, Merson4 – метод Мерсона, DP5 – метод Дорманда–Принса, а также неявные методы Лобатто IIIA (Lobatto4) и Радо IIA (Radau5). Цифра в обозначении метода показывает его порядок. Чтобы вычислительные затраты всех методов были примерно одинаковы, размер шага явного метода выбираем таким, чтобы на одном периоде колебаний длиной T выполнялось 120 вычислений правой части, т. е. принимаем $h = Ts/120$, где s – число стадий, совпадающее с числом вычислений правой части на одном шаге. Наш опыт показывает, что при эффективной реализации трудоемкость выполнения одной неявной стадии в 2...3 раза больше, чем явной стадии, поэтому размер шага неявного метода принимаем в 2.5 раза больше, чем у явного метода с таким же числом стадий. Первая стадия метода Лобатто – явная и не требует вычислений, поэтому ее не учитываем.

Первая задача – простейший линейный тест

$$y_1' = y_2,\quad y_2' = -y_1,\quad y_1(0) = 0,\quad y_2(0) = 1,\quad 0 \le t \le NT, \tag{1.18}$$

где $T = 2\pi$ – период колебаний, N – число периодов на интервале интегрирования. Мера колебательности этой задачи $M_к = 2\pi N$. Ошибку численного решения вычисляем по формуле

$$error = \max\left(\sqrt{e_1^2(t) + e_2^2(t)},\ 0 \le t \le NT\right), \quad e_i(t) = y_i(t) - \tilde{y}_i(t), \quad i = 1,2, \tag{1.19}$$

где $y_1(t) = \sin(t)$, $y_2(t) = \cos(t)$ – точное решение, а $\tilde{y}_i(t)$ – численное решение. Полученные ошибки при трех значениях числа периодов N приведены в табл. 1.4. Видно, что методы более высокого порядка имеют преимущество, а ошибки решения всех методов возрастают пропорционально интервалу интегрирования. Методы Merson4 и Lobatto4 показывают близкие результаты, что вполне объяснимо, поскольку они имеют одинаковые значения C_φ при $p = r = 4$ и доминировании ошибки по фазе.

Таблица 1.4. Ошибки решения колебательной задачи (1.18)

Метод	h	Ошибка		
		$N = 1$	$N = 10$	$N = 100$
Ralston3	$T/40$	1.01×10^{-3}	1.01×10^{-2}	9.65×10^{-2}
Merson4	$T/24$	4.20×10^{-5}	4.20×10^{-4}	4.20×10^{-3}
DP5	$T/20$	5.52×10^{-6}	5.52×10^{-5}	5.52×10^{-4}
Lobatto4	$T/24$	4.08×10^{-5}	4.08×10^{-4}	4.08×10^{-3}
Radau5	$T/16$	8.09×10^{-6}	8.09×10^{-5}	8.09×10^{-4}

Вторая задача – нелинейное уравнение маятника $\alpha'' = -\sin(\alpha)$, где α – угол отклонения маятника от вертикальной оси. Обозначив $y_1 = \alpha$, $y_2 = \alpha'$, получим систему ОДУ

$$y_1' = y_2, \quad y_2' = -\sin(y_1), \quad y_1(0) = \pi/2, \quad y_2(0) = 1, \quad 0 \le t \le NT, \tag{1.20}$$

где $T = 7.416298709205$ – период колебаний. Задача имеет колебательное решение с амплитудой $\pi/2$ (y_1) и $\sqrt{2}$(y_2), которое мало отличается от синусоиды с такими же амплитудой, периодом и фазой (разность не превышает 0.036). Эта задача не имеет аналитического решения, но известно точное решение в отдельных точках:

$\mathbf{y}(kT) = (\pi/2, 0)^{\mathrm{T}}$, $\mathbf{y}((k + 1/4)T) = (0, -\sqrt{2})^{\mathrm{T}}$, $\mathbf{y}((k + 1/2)T) = (-\pi/2, 0)^{\mathrm{T}}$,

$\mathbf{y}((k + 3/4)T) = (0, \sqrt{2})^{\mathrm{T}}$, $k = 0, 1, 2, \ldots$,

в которых мы и вычисляли ошибку, используя формулу (1.19).

Мера колебательности этой задачи $M_к = 4.14N$, что немного меньше, чем у задачи (1.18). Ошибки решения приведены в табл. 1.5. На этот раз ошибки всех методов, за исключением Lobatto4, пропорциональны квадрату интервала интегрирования, а метод Lobatto4 сохраняет линейную зависимость ошибки от интервала интегрирования. Аналогично ведут себя и другие симметричные методы Лобатто и Гаусса, что подтверждает преимущество симметричных методов при решении колебательных задач.

Таблица 1.5. Ошибки решения уравнения маятника (1.20)

Метод	h	Ошибка		
		$N = 1$	$N = 10$	$N = 100$
Ralston3	$T/40$	2.15×10^{-3}	1.82×10^{-1}	3.06
Merson4	$T/24$	2.39×10^{-5}	1.85×10^{-3}	2.31×10^{-1}
DP5	$T/20$	3.89×10^{-5}	2.81×10^{-3}	2.78×10^{-1}
Lobatto4	$T/24$	3.27×10^{-5}	4.22×10^{-4}	4.32×10^{-3}
Radau5	$T/16$	4.59×10^{-5}	4.08×10^{-3}	4.27×10^{-1}

Третий тест – простейшая задача двух тел, одно из которых неподвижно, а второе движется по круговой орбите. Уравнения имеют вид

$$x'' = -x/r^3, \quad y'' = -y/r^3, \quad r = \sqrt{x^2 + y^2}.$$

Примем $x_0 = y_0' = 0$, $y_0 = x_0' = 1$, тогда период обращения $T = 2\pi$ и решение $x(t) = \sin(t)$, $y(t) = \cos(t)$. На интервале $0 \le t \le NT$ задача имеет колебательность $M_к = 6.28N$ (такую же, как и первый тест), но при этом проявляет неустойчивость ($M_{ну} = M_ж = 8.89N$). Ошибки решения приведены в табл. 1.6. Видно, что и на этот раз симметричный метод Lobatto4 имеет преимущество. При решении этой задачи, как и задачи (1.20), ошибка метода Lobatto4 и других симметричных методов пропорциональна интервалу интегрирования, а остальные методы демонстрируют более быстрый (примерно квадратичный) рост ошибки. Резюмируя, можно сказать, что для эффективного решения колебательных задач на больших интервалах следует использовать методы высоких порядков, а если задача нелинейная, то преимущество имеют симметричные методы.

Таблица 1.6. Ошибки решения задачи двух тел

Метод	h	Ошибка		
		$N = 1$	$N = 10$	$N = 100$
Ralston3	$T/40$	2.12×10^{-3}	2.14×10^{-1}	2.04
Merson4	$T/24$	5.11×10^{-4}	2.78×10^{-2}	1.92
DP5	$T/20$	2.50×10^{-4}	1.19×10^{-2}	1.02
Lobatto4	$T/24$	2.69×10^{-4}	2.67×10^{-3}	2.67×10^{-2}
Radau5	$T/16$	1.65×10^{-4}	1.54×10^{-2}	1.40

1.6. Плохо обусловленные задачи

Рассмотренные выше колебательные задачи описываются в общем случае системой 2-го порядка

$$\mathbf{y}'' = \mathbf{F}(t, \mathbf{y}), \tag{1.21}$$

которую можно преобразовать в систему 1-го порядка

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}' \end{bmatrix}' = \begin{bmatrix} \mathbf{y}' \\ \mathbf{F}(t, \mathbf{y}) \end{bmatrix}, \quad \begin{bmatrix} \mathbf{y}(t_0) \\ \mathbf{y}'(t_0) \end{bmatrix} = \begin{bmatrix} \mathbf{y}_0 \\ \mathbf{y}_0' \end{bmatrix}.$$

Матрица Якоби такой системы имеет вид

$$\mathbf{J}(t,\mathbf{y})=\begin{bmatrix}\mathbf{0} & \mathbf{I}\\ \mathbf{F}_{\mathbf{y}} & \mathbf{0}\end{bmatrix},\quad \mathbf{F}_{\mathbf{y}}=\frac{\partial \mathbf{F}(t,\mathbf{y})}{\partial \mathbf{y}},$$

а ее собственные значения равны $\lambda_i=\pm\sqrt{\mu_i}$, где μ_i – собственные значения матрицы $\mathbf{F}_{\mathbf{y}}$. Таким образом, если спектр матрицы Якоби содержит собственное число с отрицательной действительной частью, то он содержит также и число с такой же положительной действительной частью. Тогда $M_{\text{ну}}=M_{\text{ж}}$, а система ОДУ является неустойчивой. Наличие собственных чисел с положительной действительной частью вносит дополнительные трудности при численном решении, поскольку малейшее отклонение от точного решения может привести к быстрому росту ошибки.

Задачи с большим значением $M_{\text{ну}}$ будем называть плохо обусловленными. В частности, многие задачи, описываемые уравнениями 2-го порядка вида (1.21) или более общего вида $\mathbf{y}''=\mathbf{F}(t,\mathbf{y},\mathbf{y}')$, могут быть не только колебательными, но и плохо обусловленными. Рассмотрим, например, задачу

$y''_1=2y_1y_2,\quad y''_2=30(y_2-y_1^2)+6y_1^2y_2,$

$y_1(0)=y_2(0)=1,\quad y'_1(0)=-1,\quad y'_2(0)=-2,\ 0\leqslant t\leqslant T$

с решением $y_1(t)=(1+t)^{-1}$, $y_2(t)=(1+t)^{-2}$. При любом T задача имеет $M_{\text{ну}}=M_{\text{ж}}=(5.48\ldots5.74)T$. При умеренных требованиях к точности все решатели системы MATLAB и ПО SimInTech дают правильное решение этой задачи на интервале $0\leq t\leq 1$. Но при $T=10$ ни один из этих решателей не смог обеспечить правильного решения на всем интервале даже при минимально возможном допуске на ошибку. Например, при допуске на ошибку $Tol=10^{-14}$ и $T=1$ ошибка метода Мерсона не превысила заданного допуска. Но уже при $T=4$ ошибка была 1.6×10^{-4}, а при $T=7$ ошибка в конце интервала достигла 22.3.

К плохо обусловленным можно также отнести многие задачи небесной механики, трудность решения которых связана в первую очередь с наличием собственных чисел матрицы Якоби в правой полуплоскости. Одна из них – задача Аренсторфа, рассмотренная в [74]. Рассматриваются два тела с массами $1-\mu$ и μ, участвующие в совместном движении в некоторой плоскости, и движущееся в той же плоскости третье тело пренебрежимо малой массы. Уравнения имеют вид:

$x''=x+2y'-\mu'(x+\mu)/D_1-\mu(x-\mu')/D_2,$

$y''=y-2x'-\mu' y/D_1-\mu y/D_2,$

$D_1=((x+\mu)^2+y^2)^{3/2},\quad D_2=((x-\mu')^2+y^2)^{3/2},$

$\mu=0.012277471,\quad \mu'=1-\mu,$

$x(0)=0.994,\quad x'(0)=y(0)=0,$

$y'(0)=-2.00158510637908252240537862224,$

$T=17.06521656015796255889172062 49.$

Начальные условия тщательно подобраны, чтобы получить замкнутую орбиту с периодом T. Траектория решения показана на рис. 1.4. На интервале $0 \le t \le T$ задача имеет $M_{\text{ну}} = M_{\text{ж}} = 31.4$ и $M_{\text{к}} = 30.7$. Для получения приемлемого решения на таком интервале методами 4-го порядка пришлось выполнить 16 000 шагов размером $h = T/16000$. При этом ошибка в конечной точке равна 2.41×10^{-3} у метода Merson4 и 1.33×10^{-3} у метода Lobatto4. Но уже на двух периодах при таком же размере шага ошибка была 0.794 у метода Merson4 и 0.401 у метода Lobatto4. Мы видим, что ошибка возрастает значительно быстрее, чем при решении рассмотренных ранее колебательных задач, при этом симметричный метод Lobatto4 не имеет ощутимого преимущества.

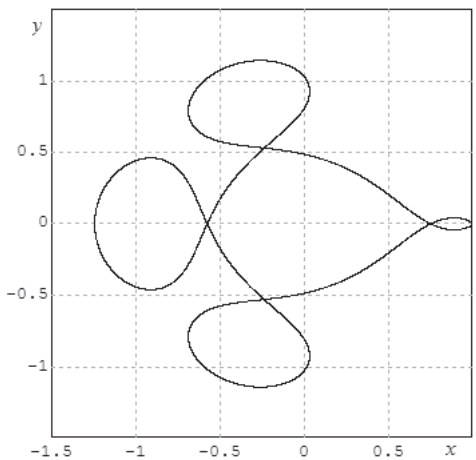

Рис. 1.4. Орбита Аренсторфа

Поскольку движение по полученной траектории крайне неравномерно, значительно более эффективно решение этой задачи с автоматическим выбором размера шага. При задаваемом допуске на ошибку $Tol = 10^{-4}$ потребовалось выполнить 197 шагов и 1005 вычислений правой части метода Мерсона для получения решения на одном периоде с ошибкой 2.10×10^{-3}. А на двух периодах потребовалось 378 шагов и 1910 вычислений правой части, но ошибка в этом случае составила уже 0.336. В [74] было показано, что для эффективного решения задачи Аренсторфа и многих подобных задач следует использовать методы высоких порядков, например метод Дорманда–Принса 8-го порядка или экстраполяционный метод переменного порядка, реализованные в решателях DOPRI8 и ODEX.

При решении плохо обусловленных задач важно убедиться в достоверности полученного результата. Для этого следует повторить расчет на более частой сетке (уменьшив размер шага или значение *Tol*) либо использовать другой метод. Если решение не изменяется, то это повышает вероятность получения действительно правильного решения. Однако существуют задачи, при реше-

нии которых разными методами и с различными значениями *Tol* будет получен один и тот же неправильный результат. Это жесткие локально-неустойчивые задачи. Одна из таких задач имеет вид:

$y_1' = y_2, \quad y_2' = \mu(1 - y_1^2)(y_1 + y_2),$
$y_1(0) = 2, \quad y_2(0) = 0, \quad 0 \le t \le 3.$

Если решать эту задачу при $\mu \ge 10^8$ и умеренных требованиях к точности одним из известных неявных решателей, то наверняка будет получено неправильное монотонно затухающее решение (точное решение – периодическое). Для получения правильного результата придется задать очень малое значение *Tol*. В главе 8 рассмотрены специальные явные методы, позволяющие эффективно решать такие задачи.

1.7. Задачи с разрывами

До сих пор мы рассматривали задачи, в которых функция $\mathbf{f}(t, \mathbf{y})$ является гладкой. Однако в практических задачах часто встречаются зависимости, содержащие разрывы (реле, люфт, гистерезис, цифровой регулятор и т. д.). При решении таких задач следует уменьшать шаг в окрестности разрыва, поэтому эффективность их решения в значительной степени определяется алгоритмом выбора размера шага.

Простейшая задача с нелинейностью релейного типа имеет вид:

$$y_1' = y_2, \quad y_2' = -2\text{sign}(y_1), \quad y_1(0) = 1, \quad y_2(0) = 0, \; 0 \leqslant t \leqslant 8. \tag{1.22}$$

Она имеет периодическое решение с периодом 4, состоящее из отрезков парабол (y_1) и прямых (y_2). При решении задачи методом Мерсона с шагом $h = 0.1$ требуется выполнить 400 вычислений правой части, при этом ошибка в конце интервала равна 0.937. При шаге $h = 0.01$ затраты возрастают в 10 раз, а ошибка равна 0.0946, т. е. уменьшается пропорционально размеру шага. Мы видим, что даже при выборе размера шага, обеспечивающего точное попадание в точки переключения ($t = 1, 3, 5, 7$), результаты оказываются неудовлетворительными. И дело тут не только в ошибках округления, поскольку для точного прохождения точки разрыва следует изменять значение y_2' не в процедуре вычисления правой части, а в отдельной процедуре, вызываемой после выполнения очередного шага непосредственно перед точкой разрыва.

Поскольку в общем случае при решении задач с разрывами невозможно заранее знать, в какие моменты модельного времени случаются разрывы, следует использовать методы с переменным размером шага. При этом можно использовать два способа:

1) применять алгоритм управления размером шага на основе получаемой на каждом шаге оценки локальной ошибки. В этом случае можно использовать обычные решатели, не внося в них никаких изменений;
2) при определении размера шага использовать не только оценку ошибки, но и прогноз ближайшего момента разрыва. Интегрирование до очеред-

ной точки разрыва выполняется обычным образом, далее управление передается программе, изменяющей некоторые переменные. После этого интегрирование возобновляется с новыми значениями переменных.

Решение задачи (1.22) с автоматическим выбором шага оказалось значительно более эффективным. При задаваемой точности $Tol = 10^{-4}$ мы получили ошибку $e = 6.85 \times 10^{-4}$ и число вычислений функции $Nf = 615$ при использовании первого способа и $e = 3.66 \times 10^{-15}$, $Nf = 96$ при использовании второго способа. Для решения задачи вторым способом была введена дискретная переменная s, которая описывает состояние реле и в начальный момент равна 1. Тогда второе уравнение в (1.22) запишется в виде $y_2' = -2s$, где переменная s изменяется (меняет знак) только после выполнения успешного шага непосредственно перед моментом разрыва.

Второй способ не только более эффективен, но и позволяет решать задачи, которые принципиально невозможно решать первым способом. Одна из таких задач – скачущий мяч, который при отскоке от пола меняет направление движения, уменьшая или сохраняя скорость. Уравнения имеют вид:

$$y' = v, \quad v' = -g, \quad \text{if } (y \le 0) \text{ and } (v < 0) \text{ then } v = -kv, \qquad (1.23)$$
$$0 < k \le 1, \quad y(0) = 1, \quad v(0) = 0.$$

В отличие от (1.22), эти уравнения не могут быть описаны в подпрограмме вычисления правой части. Задачи вида (1.22) или (1.23) часто описывают в терминах событийного моделирования [41, 43, 142]. Событием называется выполнение условий, при которых происходит скачкообразное изменение переменных, параметров или даже структуры моделируемой системы. При наличии таких условий, кроме решения дифференциальных уравнений, необходимо решать две задачи: локализация события (какое событие и в какой момент должно произойти) и реализация события, т. е. выполнение вычислений, составляющих суть события. При решении уравнений (1.23) локализация события сводится к определению момента касания мяча пола (т. е. момента, когда $y = 0$ при $v < 0$), а его реализация заключается в вычислении нового значения скорости по формуле $v = -kv$. В общем случае локализация события сводится к решению нелинейного алгебраического уравнения. Мы вернемся к таким задачам в разделе 2.9.

1.8. Одношаговые методы Рунге–Кутты

Один шаг s-стадийного метода Рунге–Кутты для решения задачи Коши (1.1) задается формулами:

$$\mathbf{y}_1 = \mathbf{y}_0 + h\sum_{i=1}^{s} b_i \mathbf{F}_i, \quad \mathbf{F}_i = \mathbf{f}(t_0 + c_i h, \mathbf{Y}_i), \quad \mathbf{Y}_i = \mathbf{y}_0 + h\sum_{j=1}^{s} a_{ij}\mathbf{F}_j, \quad i = 1,\ldots,s \qquad (1.24)$$

(приводим формулы первого шага, поскольку на последующих шагах используются точно такие же формулы). Коэффициенты метода можно представить в виде таблицы Бутчера:

$$\begin{array}{c|ccc} c_1 & a_{11} & \dots & a_{1s} \\ \vdots & \vdots & \dots & \vdots \\ c_s & a_{s1} & \dots & a_{ss} \\ \hline & b_1 & \dots & b_s \end{array} = \begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^{\mathrm{T}} \end{array}.$$

Метод является явным, если $a_{ij} = 0$ при $j \geq i$, в противном случае он неявный. В случае явного метода формулы (1.24) могут быть непосредственно реализованы. Для неявного метода эти формулы задают систему нелинейных алгебраических уравнений относительно стадийных значений $\mathbf{Y}_i$, размер которой в общем случае равен произведению числа стадий s на число уравнений в системе ОДУ n. Среди неявных методов Рунге–Кутты наиболее просто реализуются диагонально-неявные (DIRK – Diagonally Implicit Runge–Kutta), у которых матрица $\mathbf{A}$ имеет нижнюю треугольную форму. В этом случае система алгебраических уравнений размера sn распадается на s последовательно решаемых систем размера n. Обычно также требуют, чтобы все ненулевые диагональные элементы матрицы $\mathbf{A}$ были равны между собой, что позволяет выполнять только одно LU-разложение матрицы $\mathbf{I} - h\gamma\mathbf{J}$ на шаге интегрирования, где γ – диагональный элемент матрицы $\mathbf{A}$, $\mathbf{J}$ – матрица Якоби системы ОДУ. Такие методы называют однократно диагонально-неявными (SDIRK – Singly DIRK).

Простейшими методами Рунге–Кутты являются методы Эйлера: явный (1.3) и неявный (1.4), которые имеют таблицы Бутчера

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array} \quad \text{и} \quad \begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}.$$

Классический метод Рунге–Кутты 4-го порядка имеет таблицу

$$\begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ 1 & 0 & 0 & 1 & \\ \hline & 1/6 & 1/3 & 1/3 & 1/6 \end{array}$$

(для явных и диагонально-неявных методов обычно опускают нулевые элементы матрицы $\mathbf{A}$).

К неявным методам Рунге-Кутты второго порядка относятся методы средней точки и трапеций с таблицами Бутчера

$$\begin{array}{c|c} 1/2 & 1/2 \\ \hline & 1 \end{array} \quad \text{и} \quad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \end{array}.$$

Эти методы имеют одинаковую функцию устойчивости $R(z) = \dfrac{1+z/2}{1-z/2}$ и являются A-устойчивыми (но не L-устойчивыми). Метод трапеций реализован в системе MATLAB (решатель ode23t). Примером L-устойчивого метода DIRK второго порядка является метод

$$\begin{array}{c|ccc} 0 & 0 & & \\ 2\gamma & \gamma & \gamma & \\ 1 & (1-\gamma)/2 & (1-\gamma)/2 & \gamma \\ \hline & (1-\gamma)/2 & (1-\gamma)/2 & \gamma \end{array} \qquad \gamma = 1-\sqrt{2}/2.$$

Его можно интерпретировать как последовательное применение правила трапеций и формулы дифференцирования назад 2-го порядка, поэтому он получил название TR-BDF2. Этот метод реализован в системе MATLAB под названием ode23tb и в ПО SimInTech, в котором реализованы также методы DIRK третьего и четвертого порядков (DIRK3 и DIRK4).

Для уменьшения вычислительных затрат при реализации неявных методов было предложено ограничить решение алгебраических уравнений одной ньютоновской итерацией. Например, применяя одну итерацию при решении алгебраических уравнений в неявном методе Эйлера, получим метод, задаваемый формулой

$$\mathbf{y}_1 = \mathbf{y}_0 + (\mathbf{I} - h\mathbf{J})^{-1} h\mathbf{f}(t_0, y_0).$$

Такие методы получили название линейно-неявных. Применительно к диагонально-неявным методам Рунге-Кутты методы такого типа были предложены Розенброком [139]. L-устойчивый метод Розенброка второго порядка реализован в системе MATLAB под названием ode23s.

1.9. Многошаговые методы

В общем случае линейные k-шаговые методы задаются формулами вида

$$\mathbf{y}_{i+1} = \sum_{j=1}^{k} a_j \mathbf{y}_{i+1-j} + h\sum_{j=0}^{k} b_j \mathbf{f}_{i+1-j}, \quad \mathbf{f}_l = \mathbf{f}(t_l, \mathbf{y}_l). \tag{1.25}$$

Среди них наиболее известны и популярны явные и неявные методы Адамса, а также неявные методы, основанные на формулах численного дифференцирования. Неявные методы имеют $b_0 \neq 0$. Для неявных методов действует второй барьер Далквиста: если метод A-устойчив, то его порядок $p \leq 2$.

Методы Адамса имеют $a_j = 0$ при $j > 1$ и получены из условия максимального порядка при заданном k. Явные методы Адамса имеют порядок k, а неявные – порядок $k + 1$. При $k = 1$ получаем, соответственно, явный метод Эйлера и неявный метод трапеций. При $k = 2$ и постоянном размере шага формула явного ме-

тода $\mathbf{y}_{i+1} = \mathbf{y}_i + \frac{h}{2}(3\mathbf{f}_i - \mathbf{f}_{i-1})$, а неявного метода $\mathbf{y}_{i+1} = \mathbf{y}_i + \frac{h}{12}(5\mathbf{f}_{i+1} + 8\mathbf{f}_i - \mathbf{f}_{i-1})$. При $k \geq 2$ явные методы Адамса имеют очень ограниченные области устойчивости, поэтому самостоятельно они не применяются. Области устойчивости неявных методов Адамса при $k \geq 2$ также ограничены, что делает неэффективным их использование для решения жестких задач. В то же время сочетание явных и неявных формул Адамса позволяет построить весьма эффективные методы прогноза-коррекции. При $k = 2$ и $h = \text{const}$ формулы такого метода имеют вид:

- прогноз: $\hat{\mathbf{y}}_{i+1} = \mathbf{y}_i + \frac{h}{2}(3\mathbf{f}_i - \mathbf{f}_{i-1}), \quad \hat{\mathbf{f}}_{i+1} = \mathbf{f}(t_{i+1}, \hat{\mathbf{y}}_{i+1})$;
- коррекция: $\mathbf{y}_{i+1} = \mathbf{y}_i + \frac{h}{12}(5\hat{\mathbf{f}}_{i+1} + 8\mathbf{f}_i - \mathbf{f}_{i-1}), \quad \mathbf{f}_{i+1} = \mathbf{f}(t_{i+1}, \mathbf{y}_{i+1})$.

Формулы Адамса удобны для реализации явных методов переменного порядка и шага, например в решателе ode113 системы MATLAB реализованы формулы порядка от 2-го до 13-го. Явные многошаговые методы, основанные на формулах Адамса, рассмотрены в разделах 8.6–8.8.

Для решения жестких задач используют методы вида (1.25), основанные на формулах дифференцирования назад (ФДН, или BDF – backward differentiation formulas), в которых производная в точке t_{i+1} аппроксимируется по значениям $\mathbf{y}_{i+1-j}$, $j = 0,\ldots, k$. ФДН порядка k имеет вид

$$\mathbf{y}_{i+1} = \sum_{j=1}^{k} a_j \mathbf{y}_{i+1-j} + hb_0 \mathbf{f}(t_{i+1}, \mathbf{y}_{i+1}).$$

При $k = 1$ получаем $b_0 = 1$, что соответствует неявному методу Эйлера, а при $k = 2$ получаем L-устойчивый метод $\mathbf{y}_{i+1} = \frac{4}{3}\mathbf{y}_i - \frac{1}{3}\mathbf{y}_{i-1} + \frac{2}{3}h\mathbf{f}(t_{i+1}, \mathbf{y}_{i+1})$. ФДН до 6-го порядка включительно являются $L(\alpha)$-устойчивыми. На их основе Ч. В. Гир (C. W. Gear) разработал метод переменного порядка и шага и реализовал его в программе DIFSUB, которая была опубликована в 1971 году [102]. Метод Гира до сих пор считается одним из самых эффективных для жестких задач. Он реализован в SimInTech, а в системе MATLAB в решателе ode15s реализован метод NDF (Numerical Differential Formulas), который практически является модификацией метода Гира.

1.10. Явные методы для жестких задач

Наряду с неявными методами для решения жестких задач успешно применяют специальные явные методы, позволяющие эффективно решать многие задачи с вещественным жестким спектром. Принципы построения таких методов рассмотрим на примере явного двухстадийного метода Рунге–Кутты 1-го порядка с функцией устойчивости $R(z) = 1 + z + dz^2$. На рис. 1.5 приведены области устойчивости такого метода при различных значениях d. При $d > 1/8$ область

устойчивости односвязная, а ее длина равна $1/d$. Значение $d = 1/4$ соответствует двум шагам метода Эйлера. Уменьшая d, можно увеличить длину области. При $d < 1/8$ область устойчивости перестает быть односвязной и состоит из двух разделенных областей, наиболее удаленная из которых позволяет обеспечить стабилизацию метода в жесткой части спектра.

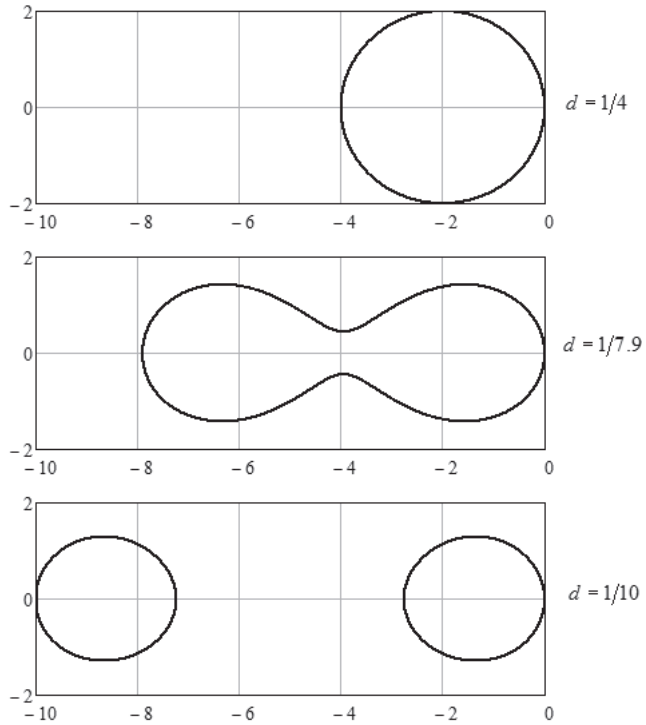

Рис. 1.5. Области устойчивости двухстадийного метода

Таким образом, можно сформулировать два способа построения явных методов для жестких задач. Первый способ основан на максимальном расширении области устойчивости. Построение многочленов устойчивости таких методов выполняется исходя из условия чебышевского альтернанса, т. е. чередования равных по модулю максимальных и минимальных значений многочлена. Методы с расширенными областями устойчивости рассмотрены в главе 7.

Идея второго способа заключается в получении на основе предварительных стадий оценок наибольших по модулю собственных значений матрицы Якоби, которые используются в заключительной формуле для стабилизации расчетной схемы в полученных точках жесткого спектра. Такие методы, названные адаптивными, рассмотрены в главе 8.

1.11. Дифференциально-алгебраические уравнения

Часто уравнения математической модели представлены не в нормальной форме Коши (1.1), а в виде системы ДАУ:

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}, \mathbf{z}),\ \mathbf{y}(t_0) = \mathbf{y}_0, \tag{1.26а}$$

$$\mathbf{0} = \mathbf{g}(\mathbf{y}, \mathbf{z}),\ \mathbf{z}(t_0) = \mathbf{z}_0. \tag{1.26б}$$

Предполагаем, что размерность векторной функции $\mathbf{g}$ совпадает с размерностью вектора $\mathbf{z}$, а начальные условия $\mathbf{y}_0$, $\mathbf{z}_0$ согласованы (для ДАУ индекса 1 это означает, что они удовлетворяют алгебраической подсистеме (1.26б)). Будем называть компоненты вектора $\mathbf{y}$ дифференциальными переменными, а вектора $\mathbf{z}$ – алгебраическими переменными.

Для численного решения уравнений (1.26) можно использовать два способа [75]. Первый из них – метод пространства состояний – основан на приведении уравнений к нормальной форме (1.1) путем численного решения алгебраической подсистемы (1.26б) при заданном векторе $\mathbf{y}$. Подставляя затем полученное значение вектора $\mathbf{z}$ в (1.26а), получаем искомые значения производных. Метод пространства состояний позволяет разделить задачи решения дифференциальных и алгебраических уравнений, поэтому его можно применять в сочетании с любым методом интегрирования. Но его нельзя использовать при решении задач высших индексов, когда алгебраическая подсистема вырождена.

Второй способ – метод ε-вложения – основан на совместном решении дифференциальной и алгебраической подсистем и может быть интерпретирован как решение сингулярно возмущенной задачи

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}, \mathbf{z}),\ \mathbf{y}(t_0) = \mathbf{y}_0,$$

$$\varepsilon\mathbf{z}' = \mathbf{g}(\mathbf{y}, \mathbf{z}),\ \mathbf{z}(t_0) = \mathbf{z}_0$$

при $\varepsilon \to 0$. Применяя метод Рунге–Кутты, получим формулы одного шага интегрирования системы (1.26) в виде:

$$\mathbf{y}_1 = \mathbf{y}_0 + h\sum_{i=1}^{s} b_i \mathbf{Y}_i',\quad \mathbf{z}_1 = \mathbf{z}_0 + h\sum_{i=1}^{s} b_i \mathbf{Z}_i', \tag{1.27а}$$

$$\mathbf{Y}_i = \mathbf{y}_0 + h\sum_{j=1}^{s} a_{ij} \mathbf{Y}_j',\quad \mathbf{Z}_i = \mathbf{z}_0 + h\sum_{j=1}^{s} a_{ij} \mathbf{Z}_j', \tag{1.27б}$$

$$\mathbf{Y}'_i = \mathbf{f}(\mathbf{Y}_i, \mathbf{Z}_i),\ \ \mathbf{0} = \mathbf{g}(\mathbf{Y}_i, \mathbf{Z}_i),\ \ i = 1,\ldots, s. \tag{1.27в}$$

Формулы (1.27б, в) задают систему нелинейных алгебраических уравнений относительно векторов $\mathbf{Y}_i'$, $\mathbf{Z}_i'$, $i = 1,\ldots, s$ (векторы стадийных значений $\mathbf{Y}_i$, $\mathbf{Z}_i$ нетрудно исключить). Решая эти уравнения, находим векторы $\mathbf{Y}_i'$, $\mathbf{Z}_i'$, которые подставляем в (1.27а). Метод ε-вложения позволяет решать задачи высших индексов, но его можно использовать только в сочетании с неявным методом интегрирования, поскольку для явных методов система алгебраических уравнений (1.27б, в) будет вырожденной.

Среди неявных методов Рунге–Кутты для решения ДАУ обычно применяют жесткоточные методы, у которых последняя строка матрицы $\mathbf{A}$ совпадает с $\mathbf{b}^{\mathrm{T}}$. В этом случае нет необходимости находить векторы $\mathbf{Z}'_i$, а формулы (1.27) принимают вид:

$$\mathbf{Y}_i = \mathbf{y}_0 + h\sum_{j=1}^{s} a_{ij}\mathbf{Y}'_j, \quad \mathbf{y}_1 = \mathbf{Y}_s, \quad \mathbf{z}_1 = \mathbf{Z}_s,$$

$$\mathbf{Y}'_i = \mathbf{f}(\mathbf{Y}_i, \mathbf{Z}_i), \quad \mathbf{0} = \mathbf{g}(\mathbf{Y}_i, \mathbf{Z}_i), \quad i = 1, \ldots, s.$$

Преимущество жесткоточных методов заключается в том, что они обеспечивают точное выполнение алгебраического соотношения (1.26б). Для жесткоточных методов при решении задач индекса 1 метод ε-вложения идентичен методу пространства состояний.

Согласно определению Гира и др. (см. [75]), *индекс дифференцирования* системы (1.26) есть наименьшее число аналитических дифференцирований, требующихся для того, чтобы из уравнений (1.26) можно было бы получить систему ОДУ в форме Коши. При этом каждое дифференцирование понижает индекс на 1. Продифференцировав алгебраическую подсистему (1.26б) и обозначив $\mathbf{g}_y = \partial\mathbf{g}/\partial\mathbf{y}$, $\mathbf{g}_z = \partial\mathbf{g}/\partial\mathbf{z}$, получим

$$\mathbf{0} = \mathbf{g}_y\mathbf{y}' + \mathbf{g}_z\mathbf{z}'. \tag{1.28}$$

Если матрица $\mathbf{g}_z$ обратима, то из (1.26), (1.28) можно получить систему в нормальной форме Коши:

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}, \mathbf{z}), \quad \mathbf{y}(t_0) = \mathbf{y}_0,$$

$$\mathbf{z}' = -\mathbf{g}_z^{-1}\mathbf{g}_y\mathbf{f}(\mathbf{y}, \mathbf{z}), \; \mathbf{z}(t_0) = \mathbf{z}_0.$$

Таким образом, если матрица $\mathbf{g}_z$ обратима в любой точке на траектории решения, то система (1.26) имеет индекс 1, а в противном случае (если матрица $\mathbf{g}_z$ вырождена) индекс системы больше 1. Системы высших индексов (2 и выше) возникают при решении многих прикладных задач. Например, уравнения механической системы со связями, сформированные методом Лагранжа, имеют индекс 3. Такие системы наиболее трудны для численного решения и могут быть решены только неявными методами.

В качестве примера рассмотрим систему ДАУ

$$x' = u, \quad y' = v, \quad u' = -xz, \quad v' = -1 - yz, \tag{1.29а}$$

$$0 = x^2 + y^2 - 1, \tag{1.29б}$$

описывающую колебания маятника в декартовой системе координат. Продифференцировав алгебраическое уравнение (1.29б), получим

$$0 = xu + yv, \tag{1.30}$$

а продифференцировав (1.30), получим уравнение

$$0 = -z(x^2 + y^2) - y + u^2 + v^2, \tag{1.31}$$

из которого можно выразить алгебраическую переменную z через дифференциальные переменные x, y, u, v. Таким образом, система (1.29а), (1.31) имеет индекс 1, система (1.29а), (1.30) – индекс 2, а исходная система (1.29) – индекс 3.

Трудность решения систем ДАУ высших индексов обусловлена тем, что порядок сходимости численного решения оказывается ниже классического порядка метода. Например, при решении ДАУ индекса 3 методом Радо IIA 5-го порядка обеспечивается только второй порядок сходимости алгебраических переменных. Если же понизить индекс ДАУ путем аналитических дифференцирований, то возникает другая трудность – полученная система будет плохо обусловленной. Это проявляется в медленном расхождении численного решения, при котором алгебраическое соотношение (1.29б) перестает выполняться (явление сноса [75]). В случае уравнений маятника (1.29) можно преодолеть указанные трудности, перейдя к другой системе координат и сформировав уравнения относительно угла отклонения, т. е. в виде (1.20). Однако в общем случае подобное преобразование может быть практически невыполнимым. Вопросы повышения эффективности методов численного решения ДАУ высших индексов рассмотрены в главах 4, 5, 6.