

# Содержание

<b>Предисловие</b> .....	7
--------------------------	---

## ▼ Глава 1

<b>Задача Коши и методы ее решения</b> .....	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

<b>Явные методы Рунге–Кутты для нежестких задач</b> .....	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

<b>Неявные методы Рунге–Кутты и Розенброка 2-го порядка</b> .....	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 IIIС.....	70
3.6. Численные эксперименты .....	71
3.7. Методы типа Розенброка.....	75
3.8. Схемы решения дифференциально-алгебраических уравнений .....	79

## ▼ Глава 4

<b>Сходимость методов Рунге–Кутты при решении жестких и дифференциально-алгебраических задач</b> .....	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

<b>Диагонально-неявные методы Рунге–Кутты</b> .....	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**

<b>для жестких задач и ДАУ.....</b>	<b>147</b>
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.....**

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.....**

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
<b>Литература .....</b>	<b>221</b>

## 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 натуральных 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](http://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 ПС. 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



## 1.1. Обыкновенные дифференциальные уравнения

Рассмотрим задачу Коши для системы ОДУ

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

где  $t$  – независимая переменная,  $\mathbf{y} = (y_1, \dots, y_n)^T$  – вектор переменных состояния,  $\mathbf{f}(t, \mathbf{y}) = (f_1(t, \mathbf{y}), \dots, f_n(t, \mathbf{y}))^T$  – 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  $\mathbf{y}_1, \dots, \mathbf{y}_N$ , аппроксимирующих истинное решение  $\mathbf{y}(t)$  в дискретные моменты модельного времени  $t_1, \dots, t_N = T$ . Интервал между двумя соседними моментами времени  $h_i = t_{i+1} - t_i$  называется шагом интегрирования (размером шага). Размер шага может быть постоянным ( $h_i = h = \text{const}$ ) 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)

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

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  $\mathbf{f}$  гладкой, а якобиан (1.2) – непрерывным.

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

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

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

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

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  $\mathbf{y}_{i-1}, \dots, \mathbf{y}_{i-k+1}, \mathbf{y}'_{i-1}, \dots, \mathbf{y}'_{i-k+1}$ . 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

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

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  $h \rightarrow 0$  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  $h \rightarrow 0$  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

$$\begin{aligned} y_1' &= -22y_1 + 20y_2^2, & y_2' &= y_1 - y_2 - y_2^2, \\ y_1(0) &= 1, & y_2(0) &= 1, & 0 \leq t \leq 1, \end{aligned} \quad (1.6)$$

решение которой  $y_1(t) = \exp(-2t)$ ,  $y_2(t) = \exp(-t)$ . Ошибку решения оценим в виде  $\varepsilon = \max(e(t_i), 0 \leq t_i \leq 1)$ , 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.

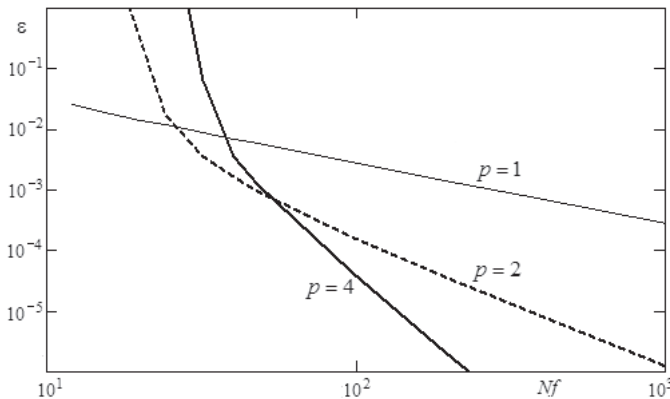


Рис. 1.1. Зависимости ошибки решения задачи (1.6) от вычислительных затрат для методов порядков 1, 2 и 4

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

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

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).

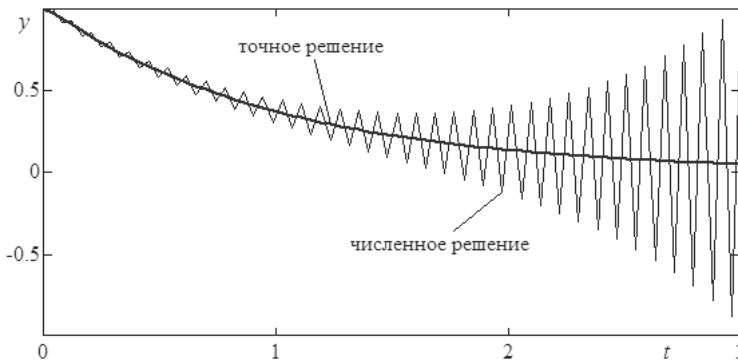


Рис. 1.2. Точное и численное решения задачи (1.7) при  $h = 3/73$

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

$$y' = \mathbf{J}y, \quad y(t_0) = y_0, \quad (1.8)$$

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)  $y'_i = \mathbf{J}y_i$ , получим

$$y_{i+1} = (\mathbf{I} + h\mathbf{J})y_i, \quad (1.9)$$

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 ( $\lambda_j$  – собственные числа матрицы  $\mathbf{J}$ ). Таким образом, условие устойчивости численного решения системы (1.8) методом Эйлера запишется в виде системы неравенств

$$|1 + h\lambda_j| < 1, \quad j = 1, \dots, n. \quad (1.10)$$

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

$$y' = \lambda y, \quad (1.11)$$

в котором  $\lambda$  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( $\alpha$ )-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( $\alpha$ )-устойчивые методы, удовлетворяющие этому условию, называются, соответственно, *L-* и *L( $\alpha$ )-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  $\mathbf{J}$  в системе (1.8) вещественные и отрицательные. Тогда условие (1.10) запишется в виде  $h < 2\tau_{\min}$ , где минимальная постоянная времени  $\tau_{\min} = \min(\tau_i, i = 1, \dots, n)$ ,  $\tau_i = -1/\lambda_i$ . 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

$$y'_1 = -y_1/\tau_1, \quad y'_2 = (y_1 - y_2)/\tau_2, \quad y_1(0) = y_2(0) = 1, \quad (1.12)$$

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  $\tau_1 \gg \tau_2$  компонента решения, соответствующая постоянной  $\tau_2$ , 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

$$\operatorname{Re} \lambda_i < 0, \quad i = 1, \dots, n;$$

$$S(t) = \frac{\max(\operatorname{Re}(-\lambda_i), i = 1, \dots, n)}{\min(\operatorname{Re}(-\lambda_i), i = 1, \dots, n)} \gg 1,$$

где величина  $S(t)$  названа локальным коэффициентом жесткости. Часто приводят эквивалентное определение, введя постоянные времени  $\tau_i = -1/\operatorname{Re} \lambda_i$ : задача считается жесткой, если имеет большой разброс постоянных времени. Максимальная постоянная времени  $\tau_{\max}$  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  $\tau_{\min}$ . Но при интегрировании явными методами приходится выбирать шаг из соображений устойчивости, т. е. порядка  $\tau_{\min}$ . 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  $\lambda$ . При больших отрицательных значениях  $\lambda$  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

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

Если все собственные значения матрицы  $\mathbf{J}$  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  $\operatorname{Re}\lambda_i < 0$ ,  $|\operatorname{Im}\lambda_i| \leq |\operatorname{Re}\lambda_i|$ .

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

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

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

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

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

$$\begin{aligned} y_1' &= y_2, y_2' = \mu(1 - y_1^2)y_2 - y_1, \\ y_1(0) &= 2, y_2(0) = y_{20}, \quad 0 \leq t \leq T. \end{aligned} \quad (1.14)$$

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 МВГУ [26]. Вычисленные при различных значениях  $\mu$  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  $\mu = 0$  задача – чисто колебательная. При увеличении  $\mu$  возрастает жесткость задачи, а также появляется неустойчивая составляющая. При  $\mu > 10$  жесткость задачи пропорциональна  $\mu^2$ , 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

$\mu$	$T$	$y_2(0)$	$M_{ж}$	$M_{к}$	$M_{ну}$	$M_{\Sigma}$	$Nf$
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	$-6.66654 \times 10^{-5}$	$2.90 \times 10^4$	4.02	24.10	$2.90 \times 10^4$	29498
1000	1614.40	$-6.66667 \times 10^{-4}$	$2.89 \times 10^6$	4.02	35.54	$2.89 \times 10^6$	2887653

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

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

В результате получаем

$$\begin{aligned} x_1' &= x_2, \quad x_2' = \mu^2((1 - x_1^2)x_2 - x_1), \\ x_1(0) &= 2, \quad x_2(0) = x_{20}, \quad 0 \leq t \leq T. \end{aligned} \quad (1.15)$$

Такую задачу удобнее исследовать потому, что при больших  $\mu$  период предельного цикла  $T$  почти не зависит от  $\mu$ , а в пределе при  $\mu \rightarrow \infty$  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  $\mu$  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  $\lambda_1, \lambda_2$  в эти моменты, а также величины интервалов времени.

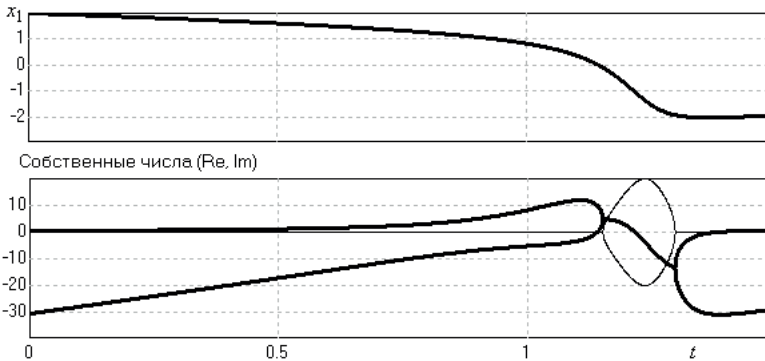


Рис. 1.3. Изменение собственных чисел матрицы Якоби на траектории решения уравнения Ван-дер-Поля (толстые линии – действительные части, тонкие – мнимые)

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

$i$	$x_1(t_i)$	$x_2(t_i)$	$\lambda_1$	$\lambda_2$	$t_i - t_{i-1}$
0	2	-0.6667	0.5556	$-3.000 \times 10^6$	---
1	1	$-1.021 \times 10^2$	$1.452 \times 10^4$	$-1.452 \times 10^4$	0.80695
2	0	$-6.669 \times 10^5$	$1.000 \times 10^6$	1.000	$1.323 \times 10^{-4}$
3	-1	$-1.334 \times 10^6$	$j1.633 \times 10^6$	$-j1.633 \times 10^6$	$9.619 \times 10^{-7}$
4	-2	$-2.228 \times 10^2$	$-2.974 \times 10^2$	$-3.000 \times 10^6$	$3.470 \times 10^{-6}$
5	-2	0.6667	0.5556	$-3.000 \times 10^6$	$1.117 \times 10^{-4}$

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	$n$	$T$	$M_{\text{ж}}$	$M_{\text{к}}$	$M_{\text{ну}}$	$M_{\Sigma}$
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	$3.84 \times 10^6$	4.0	35.8	$3.84 \times 10^6$
ROBER	[75, 128]	3	$10^{11}$	$10^{15}$	0	0	$10^{15}$
OREGO	[75, 128]	3	360	$1.13 \times 10^7$	1.5	27.1	$1.13 \times 10^7$
HIRES	[75, 128]	8	321.8122	$3.44 \times 10^4$	0.006	0	$3.44 \times 10^4$
PLATE	[75]	80	7	$6.96 \times 10^3$	$1.02 \times 10^4$	0	$1.08 \times 10^4$
BEAM	[75, 128]	80	5	0	$3.2 \times 10^4$	0	$3.2 \times 10^4$
CUSP	[75]	96	1.1	$6.87 \times 10^4$	1.8	21.0	$6.87 \times 10^4$
BRUSS	[75]	1000	10	$2 \times 10^5$	0	0	$2 \times 10^5$

## 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

$$x' = -\omega y, \quad y' = \omega x, \quad x_0 = 1, \quad y_0 = 0, \quad 0 \leq t \leq T \quad (1.16)$$

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:

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

Соответствующие локальные ошибки выражаются формулами

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

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  $C_\varphi$  – 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

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

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  $C_\varphi$  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:

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

Конец ознакомительного фрагмента.

Приобрести книгу можно

в интернет-магазине

«Электронный универс»

[e-Univers.ru](http://e-Univers.ru)