МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
им. М. В. ЛОМОНОСОВА

Научно-исследовательский вычислительный центр

ПРАКТИКУМ НА ЭВМ

ПО ВЫЧИСЛИТЕЛЬНЫМ МЕТОДАМ

О. Б. Арушанян, С. Ф. Залёткин

РЕШЕНИЕ ЗАДАЧИ КОШИ ДЛЯ ОБЫКНОВЕННЫХ

ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

ОДНОШАГОВЫМИ РАЗНОСТНЫМИ МЕТОДАМИ

Работа поддержана РФФИ (проект 01–07–90173)

Москва, 2002

З А Д А Н И Е

“Решение задачи Коши для обыкновенных дифференциальных уравнений одношаговыми разностными методами”

1. Цель задания

1) Практика работы на ЭВМ.

2) Практика применения одношаговых методов численного решения задачи Коши для обыкновенных дифференциальных уравнений.

3) Практика в модульном программировании с использованием алгоритмического языка ФОРТРАН.

4) Практика в использовании библиотек программ решения типовых математических задач.

2. Постановка задачи

Дана задача Коши для обыкновенных дифференциальных уравнений первого порядка

$$y' = f(x, y), \quad x_0 \le x \le x_0 + X; \tag{1}$$

$$y(x_0) = y_0, \tag{2}$$

которая имеет на заданном отрезке $[x_0, x_0 + X]$ единственное решение. Требуется найти приближенное решение этой задачи с заданной точностью, используя явные методы типа Рунге–Кутта.

Для решения задачи надо выполнить одно из двух заданий А или Б.

Задание А. Вычислить приближенное решение данной задачи Коши методом Рунге–Кутта с заданной точностью, используя различные способы оценки погрешности приближенного решения. Провести сравнительный анализ полученных результатов.

Задание Б. Найти приближенное решение задачи Коши методом Рунге–Кутта с заданной точностью, используя различные способы автоматического выбора шага интегрирования. Провести сравнительный анализ полученных результатов.

В обоих заданиях требуется также решить данную задачу Коши, применяя библиотеку программ решения типовых задач численного анализа, имеющуюся в штатном программном обеспечении используемой ЭВМ, и сравнить результаты, полученные по составленной программе, с результатами библиотечной программы.

Здесь приведена общая постановка задачи. Она будет уточняться при описании структуры программы и вариантов.

3. Одношаговые методы решения обыкновенных дифференциальных уравнений

В различных предметных областях часто встречаются в качестве математических моделей изучаемых процессов обыкновенные дифференциальные уравнения. Во многих случаях такие уравнения не интегрируются в явном виде. Поэтому необходимо использовать методы, позволяющие получать приближенное решение задачи. С примером таких методов знакомятся уже в общей теории обыкновенных дифференциальных уравнений, когда при доказательстве существования решения дифференциального уравнения (1), удовлетворяющего начальному условию (2), используется метод последовательных приближений Пикара. Решение (1), (2) получается как предел последовательности

$$y_0(x), y_1(x), \ldots, y_n(x), \ldots,$$

где

$$y_n(x) = y_0 + \int\limits_{x_0}^{x} f(\xi, y_{n-1}(\xi))\,d\xi, \quad y_0(x) \equiv y_0. \tag{3}$$

Функцию $y_n(x)$ с достаточно большим номером n можно принять за приближенное решение задачи (1) и (2). Однако, если правая часть $f(x,y)$ уравнения (1) — сложная функция, то применение этого метода наталкивается на большие трудности, поскольку интеграл в (3) не будет браться в квадратурах и решение нельзя получить в аналитическом виде. Мы будем применять *численные* методы, которые позволяют получать приближенное решение задачи в виде *таблицы* чисел.

3.1. Общее представление одношаговых методов

Пусть требуется найти решение задачи (1), (2) на отрезке $[x_0, x_0 + X]$. Возьмем разбиение отрезка точками

$$x_0 < x_1 < x_2 < \ldots < x_n = x_0 + X.$$

Этот набор точек называется *сеткой*, а точки x_n — *узлами* сетки. Рассмотрим *одношаговые* численные методы, т.е. такие, которые последовательно дают приближение y_n к значениям точного решения $y(x_n)$ в каждом узле x_n сетки на основе известного приближения y_{n-1} к решению в предыдущем узле x_{n-1}. В общем виде их можно представить так:

$$y_{n+1} = F(f; x_{n+1}, x_n; y_{n+1}, y_n). \tag{4}$$

Мы займемся только *явными* одношаговыми методами, для которых функция F не зависит от y_{n+1}. Обозначая

$$h = x_{n+1} - x_n,$$

явные одношаговые методы будем записывать также в виде:

$$y_{n+1} = F(f; h, x_n, y_n). \tag{5}$$

3.2. Метод рядов Тейлора

Предположим, что правая часть $f(x,y)$ дифференциального уравнения (1) имеет непрерывные частные производные до порядка s. Тогда искомое решение $y(x)$ будет иметь непрерывные производные до $s+1$ порядка включительно. Точное значение решения в узле x_1 запишем по формуле Тейлора

$$y(x_1) = y_0 + hy_0' + \frac{h^2}{2}y_0'' + \ldots + \frac{h^s}{s!}y_0^{(s)} + \frac{h^{s+1}}{(s+1)!}y^{(s+1)}(\xi), \tag{6}$$

где

$$y_0^{(k)} = y^{(k)}(x_0), \quad h = x_1 - x_0, \quad x_0 < \xi < x_1.$$

Может оказаться, что для получения решения с нужной нам точностью не требуется использовать все члены формулы (6). Производные, входящие в правую часть формулы (6), могут быть фактически найдены:

$$\begin{aligned}
y' &= f(x_0, y_0),\\
y'' &= \left\{f_x' + ff_y'\right\}_0,\\
y_0''' &= \left\{f_{xx}'' + 2ff_{xy}'' + f^2 f_{yy}'' + (f_x' + ff_y')f_y'\right\}_0\\
&\ldots\ldots\ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots
\end{aligned}$$

С увеличением порядка выражения для производных становятся все более громоздкими, и определение всех этих производных требует большего объема вычислений. Это является существенным недостатком данного метода.

3.3. Явные методы типа Рунге–Кутта

Рунге предложил следующую идею, основанную на вычислении приближенного решения y_1 в узле $x_0 + h$ в виде линейной комбинации с постоянными коэффициентами

$$y_1 = y_0 + p_{q1}k_1(h) + p_{q2}k_2(h) + \ldots + p_{qq}k_q(h), \tag{7}$$

где

$$\begin{aligned} k_1(h) &= hf(x_0, y_0), \\ k_2(h) &= hf(x_0 + \alpha_2 h, y_0 + \beta_{21}k_1(h)), \\ \ldots \quad & \ldots\ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \\ k_q(h) &= hf(x_0 + \alpha_q h, y_0 + \beta_{q1}k_1(h) + \ldots + \beta_{q,q-1}k_{q-1}(h)). \end{aligned}$$

Числа α_i, β_{ij}, p_{qi} выбираются так, чтобы разложение выражения (7) по степеням h совпадало с разложением (6) до максимально возможной степени при произвольной правой части $f(x, y)$ и произвольном шаге h. Это эквивалентно следующему. Если ввести вспомогательную функцию

$$\varphi_q(h) = y(x_0 + h) - y_0 - \sum_{i=1}^{q} p_{qi}k_i(h), \tag{8}$$

то ее разложение по степеням h должно начинаться с максимально возможной степени:

$$\varphi_q(h) = \frac{h^{s+1}}{(s+1)!}\, \varphi_q^{(s+1)}(0) + o(h^{s+1}). \tag{9}$$

Если можно определить эти постоянные так, чтобы разложение $\varphi_q(h)$ имело вид (9), то говорят, что формула (7) с выбранными коэффициентами имеет *порядок точности* s. Величина

$$\rho_1 = \varphi_q(h) = y(x_0 + h) - y_1$$

называется *погрешностью метода на шаге* или *локальной погрешностью* метода, а первое слагаемое в (9)

$$\frac{h^{s+1}}{(s+1)!}\, \varphi_q^{s+1}(0) \tag{10}$$

называется *главным членом* локальной погрешности метода. Доказано, что если $q = 1, 2, 3, 4$, то всегда можно выбрать коэффициенты α_i, β_{ij}, p_{qi} так, чтобы получить метод типа Рунге–Кутта порядка точности q. При $q = 5$ невозможно построить метод типа Рунге–Кутта (7) пятого порядка точности, т.е. нужно брать в комбинации (7) более пяти членов.

3.3.1. Одночленная формула

Будем рассматривать такую формулу, когда приближенное решение в точке $x_1 = x_0 + h$ ищется в виде:

$$y_1 = y_0 + p_{11}k_1(h),$$

где

$$k_1(h) = hf(x_0, y_0).$$

Единственный коэффициент p_{11} мы подбираем так, чтобы в разложении по степеням h функции (8), в данном случае равной

$$\varphi_1(h) = y(x_0 + h) - y_0 - p_{11}k_1(h),$$

максимальное количество членов обратилось в нуль. Этому требованию удовлетворяет единственное значение

$$p_{11} = 1,$$

для которого разложение (9) имеет вид:

$$\varphi_1(h) = \frac{h^2}{2}(f'_x + ff'_y)_0 + o(h^2).$$

Таким образом, мы получили формулу первого порядка точности

$$y_1 = y_0 + hf(x_0, y_0).$$

Это — известная формула Эйлера. Итак, метод типа Рунге–Кутта при $q = 1$ есть метод Эйлера.

Геометрический смысл метода заключается в том, что интегральная кривая $y(x)$ приближается ломаной Эйлера (рис. 1).

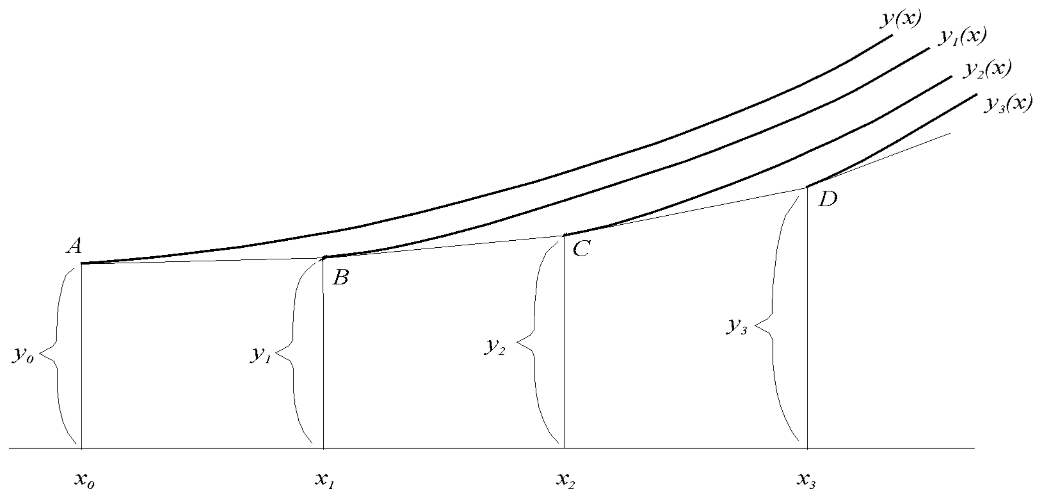


Рис. 1. Геометрическая интерпретация метода Эйлера

На $[x_0, x_1]$ интегральная кривая заменяется отрезком касательной AB, проведенной к интегральной кривой $y(x)$ в точке A. На $[x_1, x_2]$ интегральная кривая заменяется отрезком касательной BC, проведенной в точке B к интегральной кривой $y_1(x)$, проходящей через точку B. В качестве численного решения берутся ординаты вершин построенной ломаной $ABCD\ldots$. Таким образом, после выполнения каждого шага метода Эйлера приближенное решение переходит с одной интегральной кривой на другую.

3.3.2. Двучленные формулы

Строим формулу следующего вида:

$$y_1 = y_0 + p_{21}k_1(h) + p_{22}k_2(h),$$

где

$$k_1(h) = hf(x_0, y_0),$$
$$k_2(h) = hf(x_0 + \alpha_2 h, y_0 + \beta_{21}k_1(h)).$$

В выписанной формуле четыре неизвестных параметра: α_2, β_{21}, p_{21}, p_{22}. Чтобы их найти, опять строим вспомогательную функцию (8), равную

$$\varphi_2(h) = y(x_0 + h) - y_0 - p_{21}k_1(h) - p_{22}k_2(h),$$

и подбираем коэффицины так, чтобы в разложении этой функции по степеням h максимальное количество членов обратилось в нуль. Чтобы обратить в нуль первые две производные

$$\varphi_2'(0) = (1 - p_{21} - p_{22})f(x_0, y_0),$$
$$\varphi_2''(0) = \Big\{(1 - 2\alpha_2 p_{22})f_x' + (1 - 2\beta_{21}p_{22})ff_y'\Big\}_{\substack{x=x_0\\ y=y_0}}$$

необходимо, чтобы искомые коэффициенты удовлетворяли системе уравнений

$$\begin{cases} 1 - p_{21} - p_{22} = 0, \\ 1 - 2\alpha_2 p_{22} \quad = 0, \\ 1 - 2\beta_{21}p_{22} \quad = 0. \end{cases} \tag{11}$$

Тогда разложение (9) будет иметь вид:

$$\varphi_2(h) = \frac{h^3}{6}\varphi_2'''(0) + o(h^3). \tag{12}$$

Как бы мы ни выбирали параметры, третью производную

$$\varphi_2'''(0) = \Big\{(1 - 3\alpha_2^2 p_{22})f_{xx}'' + 2(1 - 3\alpha_2\beta_{21}p_{22})ff_{xy}'' + (1 - 3\beta_{21}^2 p_{22})f^2 f_{yy}'' + (f_x' + ff_y')f_y'\Big\}_{\substack{x=x_0\\ y=y_0}} \tag{13}$$

обратить в нуль для произвольной функции $f(x, y)$ нельзя. Из последних двух уравнений в (11) следует, что $\alpha_2 \neq 0$, $\beta_{21} \neq 0$, $\alpha_2 = \beta_{21}$. Принимая α_2 за свободный параметр, имеем

$$\beta_{21} = \alpha_2, \quad p_{22} = \frac{1}{2\alpha_2}, \quad p_{21} = 1 - \frac{1}{2\alpha_2}.$$

Таким образом, мы получили однопараметрическое семейство формул типа Рунге–Кутта второго порядка точности. Следует выбирать такие коэффициенты, которые дают удобные для вычисления формулы. Например, их можно выбирать так, чтобы главный член погрешности был простейший или наименьший. Рассмотрим несколько примеров.

1). Пусть $\alpha_2 = 1$. Тогда

$$\begin{aligned} y_1 &= y_0 + \frac{1}{2}(k_1 + k_2), \\ k_1 &= hf(x_0, y_0), \\ k_2 &= hf(x_0 + h, y_0 + k_1). \end{aligned} \tag{14}$$

Погрешность формулы (14), как следует из (12) и (13), имеет вид:

$$\rho_1 = \frac{h^3}{6}\Big\{-\frac{1}{2}(f_{xx}'' + 2ff_{xy}'' + f^2 f_{yy}'') + (f_x' + ff_y')f_y'\Big\}_{\substack{x=x_0\\ y=y_0}} + o(h^3). \tag{15}$$

Геометрический смысл формулы (14) заключается в следующем (рис. 2).

Через точку A проводится не касательная к интегральной кривой $y(x)$, а прямая с угловым коэффициентом, равным среднему арифметическому угловых коэффициентов касательных AB и BC, которые строятся в методе Эйлера, и в качестве решения берется ордината точки B' пересечения этой прямой с прямой $x = x_1$. Данный метод называется методом Хойна.

Если правая часть уравнения (1) не зависит от y, то формула (14) переходит в квадратурную формулу трапеций.

2). Пусть $\alpha_2 = \dfrac{1}{2}$. Тогда

$$\begin{aligned} y_1 &= y_0 + k_2, \\ k_1 &= hf(x_0, y_0), \\ k_2 &= hf\Big(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_1\Big). \end{aligned} \tag{16}$$

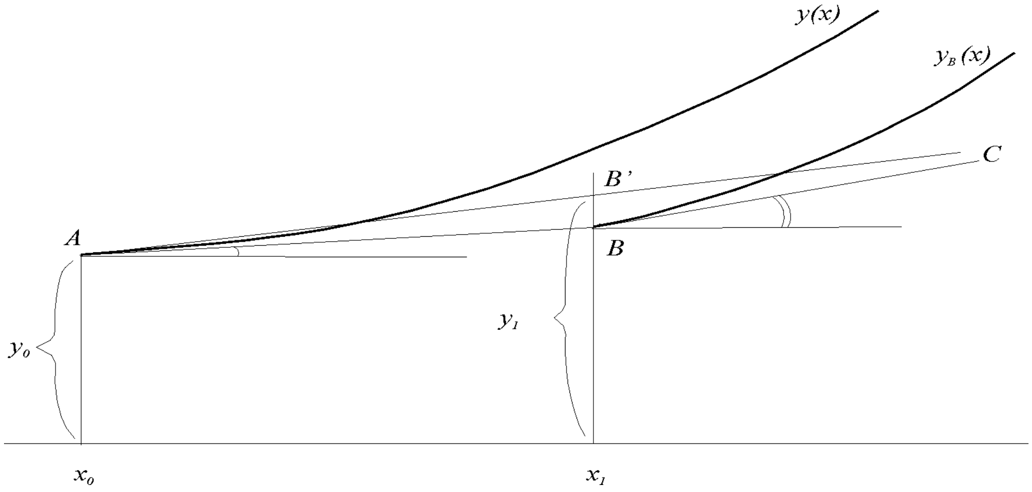


Рис. 2. Геометрическая интерпретация метода Хойна

Погрешность формулы (16), как следует из (12) и (13), имеет вид:

$$\rho_1 = \frac{h^3}{6}\Big\{\frac{1}{4}(f''_{xx} + 2ff''_{xy} + f^2 f''_{yy}) + (f'_x + ff'_y)f'_y\Big\}_{\substack{x=x_0\\ y=y_0}} + o(h^3). \tag{17}$$

Геометрический смысл формулы (16) заключается в том, что через точку A (рис. 3) проводится прямая, угловой коэффициент которой равен угловому коэффициенту касательной BC к интегральной кривой $y_B(x)$, проходящей через промежуточную точку B, построенную по методу Эйлера с шагом $\dfrac{h}{2}$. В качестве ответа берется ордината точки D пересечения этой прямой с прямой $x = x_1$.

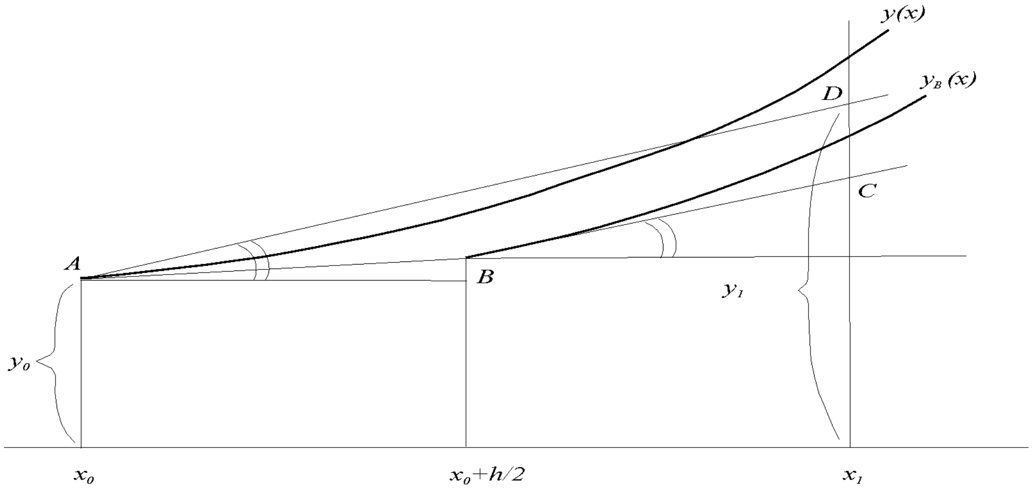


Рис. 3. Геометрическая интерпретация метода второго порядка с $\alpha_2 = 1/2$

Если правая часть дифференциального уравнения (1) не зависит от y, то формула (16) переходит в квадратурную формулу средних прямоугольников.

3). Пусть $\alpha_2 = \dfrac{2}{3}$. Тогда

$$\begin{aligned} &y_1 = y_0 + \frac{1}{4}(k_1 + 3k_2),\\ &k_1 = hf(x_0, y_0),\\ &k_2 = hf\Big(x_0 + \frac{2}{3}h, y_0 + \frac{2}{3}k_1\Big). \end{aligned} \tag{18}$$

Погрешность формулы (18), как следует из (12) и (13), имеет вид:

$$\rho_1 = \frac{h^3}{6}\Big\{(f'_x + ff'_y)f'_y\Big\}_{\substack{x=x_0\\y=y_0}} + o(h^3). \tag{19}$$

В данном случае прямая AD (рис.4) имеет угловой коэффициент, равный не среднему арифметическому угловых коэффициентов касательных AB и BC, как в методе Хойна, а взвешенному среднему их значений, при этом касательная BC проведена к интегральной кривой $y_B(x)$, проходящей через промежуточную точку B, построенную по методу Эйлера с шагом $\dfrac{2}{3}h$.

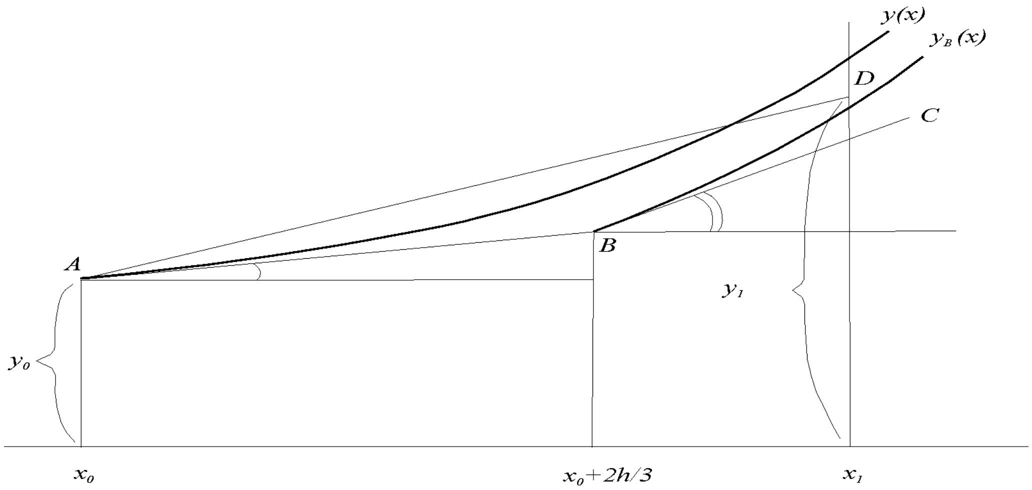


Рис. 4. Геометрическая интерпретация метода второго порядка с $\alpha_2 = 2/3$

Из трех формул (14), (16), (18) нельзя выбрать одну наилучшую (например, с точки зрения малости величины главного члена погрешности) для всех уравнений. Для одних уравнений будет предпочтительней один из методов, для других — другой, для третьих — третий.

Например, в случае интегрирования уравнения

$$y' = y^\lambda, \quad y_0 > 0,$$

при $\lambda > 1$ главный член погрешности формулы (14) оказывается меньше главных членов погрешностей формул (16) и (18), а при $\lambda < 1$ главный член погрешности формулы (16) будет меньше главных членов погрешностей формул (14) и (18) (в этом легко убедиться, сравнивая (15) и (17), а также (19) для $f(x,y) = y^\lambda$). В случае интегрирования уравнения

$$y' = f(x,y) = g(x) \tag{20}$$

первое слагаемое в локальной погрешности метода (18), как видно из (19), вообще равно нулю, поэтому метод (18) имеет для этого класса уравнений более высокий порядок точности по сравнению с методами (14) и (16).

3.3.3. Трехчленные формулы

Строим формулу следующего вида:

$$y_1 = y_0 + p_{31}k_1(h) + p_{32}k_2(h) + p_{33}k_3(h),$$

где

$$\begin{aligned} k_1(h) &= hf(x_0, y_0), \\ k_2(h) &= hf(x_0 + \alpha_2 h,\ y_0 + \beta_{21}k_1(h)), \\ k_3(h) &= hf(x_0 + \alpha_3 h,\ y_0 + \beta_{31}k_1(h) + \beta_{32}k_2(h)). \end{aligned}$$

Восемь неизвестных параметров выбираются так, чтобы разложение (9) функции (8) начиналось с максимально возможной степени. Для отыскания этих восьми параметров получается система из шести уравнений:

$$\begin{cases} \alpha_2 &= \beta_{21}, \\ \alpha_3 &= \beta_{31} + \beta_{32}, \\ p_{31} + p_{32} + p_{33} &= 1, \\ 2(\alpha_2 p_{32} + \alpha_3 p_{33}) &= 1, \\ 3(\alpha_2^2 p_{32} + \alpha_3^2 p_{33}) &= 1, \\ 6\alpha_2 \beta_{32} p_{33} &= 1. \end{cases}$$

Эта система имеет два семейства решений: двухпараметрическое со свободными параметрами α_2, α_3, причем $\alpha_2 \neq \alpha_3$ и $\alpha_2 \neq \dfrac{2}{3}$, и однопараметрическое со свободным параметром β_{32} (при $\alpha_2 = \alpha_3 = 2/3$). Для таким образом найденных параметров разложение (9) имеет вид:

$$\varphi_3(h) = \frac{h^4}{24}\varphi_3^{(4)}(0) + o(h^4).$$

Пример. Пусть $\alpha_2 = \dfrac{1}{2}$, $\alpha_3 = 1$. Тогда

$$\begin{aligned} y_1 &= y_0 + \frac{1}{6}(k_1 + 4k_2 + k_3), \\ k_1 &= hf(x_0,\ y_0), \\ k_2 &= hf\Big(x_0 + \frac{1}{2}h,\ y_0 + \frac{1}{2}k_1\Big), \\ k_3 &= hf(x_0 + h,\ y_0 - k_1 + 2k_2). \end{aligned} \tag{21}$$

Для уравнения (20) формула (21) превращается в квадратурную формулу Симпсона, которая, как известно, имеет порядок точности $O(h^5)$ и, следовательно, в этом частном случае порядок точности формулы (21) повышается.

3.3.4. Четырехчленные формулы

Существует семейство формул типа Рунге–Кутта (7) четвертого порядка точности, для которых $q = s = 4$. В качестве примера таких формул приведем формулу классического метода

Рунге–Кутта

$$
\begin{aligned}
&y_1 = y_0 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4),\\
&k_1 = hf(x_0,\, y_0),\\
&k_2 = hf\Big(x_0 + \frac{1}{2}h,\, y_0 + \frac{1}{2}k_1\Big),\\
&k_3 = hf\Big(x_0 + \frac{1}{2}h,\, y_0 + \frac{1}{2}k_2\Big),\\
&k_4 = hf(x_0 + h,\, y_0 + k_3).
\end{aligned}
\tag{22}
$$

Погрешность формулы (22), как и всех формул Рунге–Кутта четвертого порядка точности, представляется следующим образом:

$$\rho_1 = \frac{h^5}{120}\varphi_4^{(5)}(0) + o(h^5),$$

где

$$\varphi_4(h) = y(x_0 + h) - y_0 - p_{41}k_1(h) - p_{42}k_2(h) - p_{43}k_3(h) - p_{44}k_4(h).$$

Геометрическая интерпретация классического метода Рунге–Кутта (22) иллюстрируется на рис. 5.

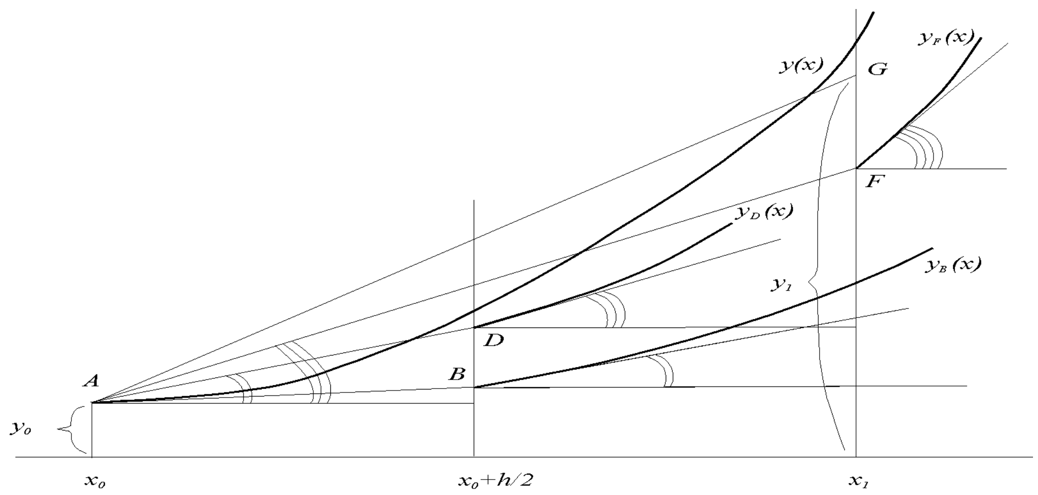


Рис. 5. Геометрическая интерпретация классического метода Рунге–Кутта

Прямая AG, проходящая через точку (x_0, y_0), имеет угловой коэфициент, равный взвешенному среднему угловых коэффициентов касательных в точках A, B, D и F, проведенных к проходящим через эти точки интегральным кривым $y(x)$, $y_B(x)$, $y_D(x)$, $y_F(x)$.

3.3.5. Формулы порядка выше четвертого

Чтобы построить метод типа Рунге–Кутта (7) порядка выше четвертого (для которого погрешность на шаге имеет порядок выше пятого для произвольного дифференциального уравнения (1) с достаточно гладкой правой частью $f(x, y)$) необходимо, чтобы в формуле (7) число слагаемых $k_i(h)$ было больше пяти: $q > 5$. Примеры таких формул приводятся в п. 3.5.2.2.3. Здесь существенно то, что значения правой части $k_i(h)$ дифференциального уравнения входят в формулу (7) в виде линейной комбинации. Может быть построен явный одношаговый метод пятого порядка точности $y_1 = y_0 + \Delta y$, использующий пять вычислений правой части, но не имеющий вида (7), так как значения $k_i(h)$ будут входить в Δy нелинейным образом.

Общим для всех рассмотренных методов численного интегрирования является то обстоятельство, что после выполнения очередного шага точка, изображающая полученное приближенное решение на плоскости (x, y), переходит с одной интегральной кривой на другую.

Методы Рунге–Кутта без труда переносятся на системы обыкновенных дифференциальных уравнений
$$\bar{y}\,' = \bar{f}(x, \bar{y}), \tag{23}$$
где
$$\bar{y} = (y^1, y^2, \ldots, y^M)^T, \quad \bar{f} = (f^1(x, y^1, \ldots, y^M), \ldots, f^M(x, y^1, \ldots, y^M))^T.$$
Формулы Рунге–Кутта записываются в векторном виде:
$$\bar{y}_1 = \bar{y}_0 + p_{q1}\bar{k}_1(h) + p_{q2}\bar{k}_2(h) + \ldots + p_{qq}\bar{k}_q(h), \tag{24}$$
где
$$\begin{aligned}
\bar{k}_1(h) &= h\bar{f}(x_0, \bar{y}_0),\\
\bar{k}_2(h) &= h\bar{f}(x_0 + \alpha_2 h, \bar{y}_0 + \beta_{21}\bar{k}_1(h)),\\
\ldots\ & \ldots\ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots\\
\bar{k}_q(h) &= h\bar{f}(x_0 + \alpha_q h, \bar{y}_0 + \beta_{q,1}\bar{k}_1(h) + \ldots + \beta_{q,q-1}\bar{k}_{q-1}(h)).
\end{aligned}$$
Всюду в дальнейшем черточку над обозначением векторов писать не будем.

3.4. Сходимость явных одношаговых методов
3.4.1. Классификация погрешностей

Прежде чем перейти к классификации погрешностей, вернемся к обсуждению исходной задачи (1),(2).

Предположим, что требуется найти функцию $y(x)$, которая является решением дифференциального уравнения (1)
$$y'(x) = f(x, y(x)), \quad x_0 \le x \le x_0 + X,$$
и принимает в точке x_0 некоторое определенное значение (2)
$$y(x_0) = y_0.$$
Может оказаться, что начальное условие $y(x_0)$ известно не точно, а определяется в результате эксперимента, например, с помощью измерений или в результате решения какой-либо другой задачи. В этом случае вместо точного начального условия $y(x_0)$ приходится использовать его приближенное значение $\tilde{y}_0$, а вместо задачи Коши (1), (2) решать задачу
$$y_0'(x) = f(x, y_0(x)); \tag{25}$$
$$y_0(x_0) = \tilde{y}_0 \tag{26}$$
с измененным начальным условием
$$y_0 - \tilde{y}_0 = R_0 \ne 0. \tag{27}$$
Решение задачи (25), (26) зависит от $\tilde{y}_0$ и не совпадает с искомым решением $y(x)$ задачи (1), (2). Разность
$$\zeta_n = y(x_n) - y_0(x_n) \tag{28}$$

называется *неустранимой погрешностью* решения $y_0(x)$.

Разность между значением решения $y_0(x)$ задачи (25), (26) и его приближенным значением y_n, полученным по формуле (5),

$$\varepsilon_n = y_0(x_n) - y_n \tag{29}$$

называется *погрешностью метода* или *глобальной погрешностью метода.* В действительности же вследствие ошибок округления и приближенного вычисления правой части $f(x,y)$ дифференциального уравнения вычисления значений y_{n+1} по формуле (5) выполняются, как правило, неточно. Фактически найденные значения $\tilde{y}_n$ удовлетворяют не соотношению (5), а следующему соотношению:

$$F(f, h, x_{n-1}, \tilde{y}_{n-1}) - \tilde{y}_n = \delta_n. \tag{30}$$

Невязка δ_n называется *погрешностью округления на n-м шаге.*

Разность между точным решением $y(x_n)$ задачи (1), (2) и приближенным фактически найденным значением $\tilde{y}_n$

$$R_n = y(x_n) - \tilde{y}_n \tag{31}$$

называется *полной погрешностью приближенного решения.* Величина

$$\eta_n = y_n - \tilde{y}_n \tag{32}$$

называется *вычислительной погрешностью.*

Из соотношений (28), (29), (31) и (32) следует, что

$$R_n = \zeta_n + \varepsilon_n + \eta_n, \tag{33}$$

т.е. полная погрешность приближенного решения равна сумме неустранимой погрешности, погрешности метода и вычислительной погрешности.

Рассмотрим поведение полной погрешности. Представим R_n в следующем виде (рис. 6):

$$\begin{aligned} R_n = y(x_n) - \tilde{y}_n &= (y(x_n) - y_{n-1}(x_n)) + (y_{n-1}(x_n) - \tilde{y}_n) = \\ &= (y(x_n) - y_{n-2}(x_n)) + (y_{n-2}(x_n) - y_{n-1}(x_n)) + (y_{n-1}(x_n) - y_n(x_n)) = \\ &= (y(x_n) - y_0(x_n)) + \sum_{j=1}^{n} (y_{j-1}(x_n) - y_j(x_n)). \end{aligned} \tag{34}$$

Здесь $y_j(x)$ — интегральная кривая, проходящая через точку $(x_j, \tilde{y}_j)$.

Разность

$$\omega_j(x) = y_{j-1}(x) - y_j(x)$$

двух решений уравнения (1) удовлетворяет линейному дифференциальному уравнению

$$(y_{j-1}(x) - y_j(x))' = f'_y(x, \hat{y}_j)(y_{j-1}(x) - y_j(x)),$$

где $\hat{y}_j(x)$ заключено между $y_{j-1}(x)$ и $y_j(x)$. Из этого уравнения находим

$$\omega_j(x) = y_{j-1}(x) - y_j(x) = \omega_j(x_j)\, e^{\int\limits_{x_j}^{x} \frac{\partial f}{\partial y}(\xi, \hat{y}_j(\xi))\, d\xi}. \tag{35}$$

Точно также с учетом (27)

$$y(x_n) - y_0(x_n) = R_0\, e^{\int\limits_{x_0}^{x} \frac{\partial f}{\partial y}(\xi, \hat{y}_0(\xi))\, d\xi}. \tag{36}$$

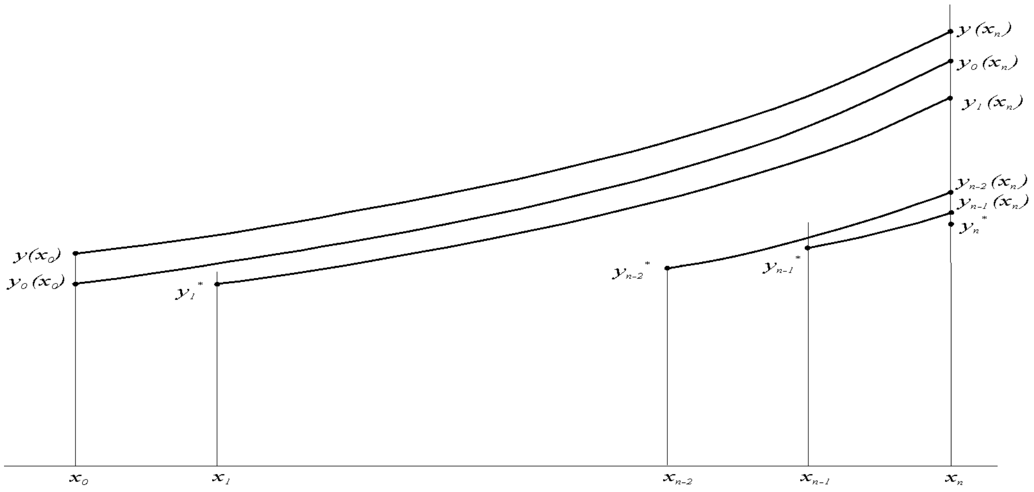


Рис. 6. Полная погрешность приближенного решения $y_n^* = \tilde{y}_n$ в точке x_n

Значение

$$\omega_j(x_j) = y_{j-1}(x_j) - y_j(x_j)$$

представляет собой сумму локальной погрешности ρ_j метода и погрешности округления δ_j на шаге $x_j - x_{j-1}$:

$$\omega_j(x_j) = \rho_j + \delta_j. \tag{37}$$

Подставляя представления (35) и (36) с учетом (37) в выражение (34) для R_n, получаем

$$R_n = \sum_{j=1}^{n} (\rho_j + \delta_j)\, e^{\int\limits_{x_j}^{x_n} \frac{\partial f}{\partial y}(\xi, \hat{y}_j(\xi))\, d\xi} + R_0\, e^{\int\limits_{x_0}^{x_n} \frac{\partial f}{\partial y}(\xi, \hat{y}_0(\xi))\, d\xi}. \tag{38}$$

Из (38) следует,что полная погрешность приближенного решения задачи (1), (2) в точке x_n равна сумме локальных погрешностей на каждом шаге, взятых с коэффициентами

$$e^{\int\limits_{x_j}^{x_n} \frac{\partial f}{\partial y}(\xi, \hat{y}_j(\xi))\, d\xi}.$$

Из соотношения (38) следует, что характер отклонения приближенного решения от точного решения, т.е. эволюция полной погрешности, зависит от поведения интегральных кривых уравнения (1). Если $f'_y = 0$, а это будет в том случае, когда правая часть уравнения не зависит от y (20), полная погрешность равна сумме локальных погрешностей:

$$R_n = R_0 + \sum_{j=1}^{n} \omega_j(x_j) = R_0 + \sum_{j=1}^{n} (\rho_j + \delta_j).$$

Если $f'_y > 0$, т.е. когда интегральные кривые расходятся, влияние локальных погрешностей, полученных на предыдущих шагах, возрастает и полная погрешность больше суммы локальных погрешностей. Если $f'_y < 0$, т.е. интегральные кривые сближаются, влияние локальных погрешностей ослабевает и глобальная ошибка, как правило, меньше суммы локальных ошибок.

Такой характер поведения справедлив как для полной погрешности, так и для отдельных частей, из которых складывается полная погрешность, а именно: неустранимой погрешности,

погрешности метода и вычислительной погрешности. На рис. 7 и 8 иллюстрируется влияние $\operatorname{sign} \dfrac{\partial f}{\partial y}$ на характер эволюции глобальной погрешности метода.

3.4.2. Мажорантная оценка полной погрешности

Опираясь на (38), можно получить мажорантную оценку полной погрешности приближенного решения задачи Коши (1), (2):

$$|R_n| \leqslant e^{LX}\Big(|R_0| + \sum_{j=1}^{n}(|\rho_j| + |\delta_j|)\Big), \tag{39}$$

где

$$L = \sup\left|\frac{\partial f}{\partial y}\right| < \infty.$$

Учитывая оценку для локальной погрешности метода s-го порядка

$$|\rho_j| = O(h^{s+1}) \leqslant C\,|x_j - x_{j-1}|^{s+1},$$

имеем

$$|R_n| \leqslant e^{LX}\Big(|R_0| + CXh^s + n\delta\Big), \tag{40}$$

где

$$\delta = \max_{1\leqslant j\leqslant n} |\delta_j|, \quad h = \max_{1\leqslant j\leqslant n} |x_j - x_{j-1}|.$$

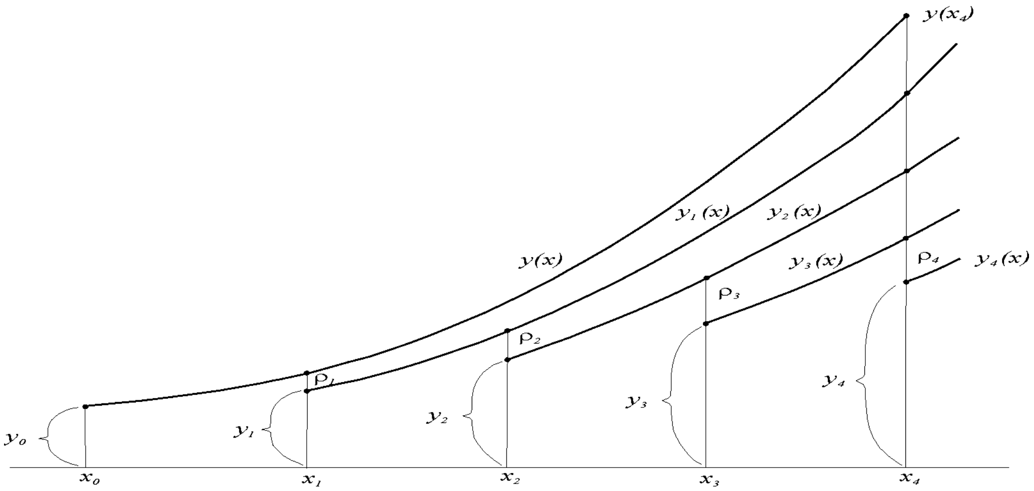


Рис. 7. $\dfrac{\partial f}{\partial y} > 0$. Интегральные кривые расходятся. Глобальная погрешность метода больше суммы локальных погрешностей: $y(x_4) - y_4 > \rho_1 + \rho_2 + \rho_3 + \rho_4$

В случае системы уравнений (23) для метода Рунге–Кутта (24) имеет место мажорантная оценка

$$||R_n|| \leqslant e^{MLX}(||R_0|| + CXh^s + n\delta), \tag{41}$$

где

$$||R_n|| = \max_{1\leqslant i\leqslant M} |R_n^i|,$$

$$\delta = \max_{1\leqslant j\leqslant n} ||\delta_j||.$$

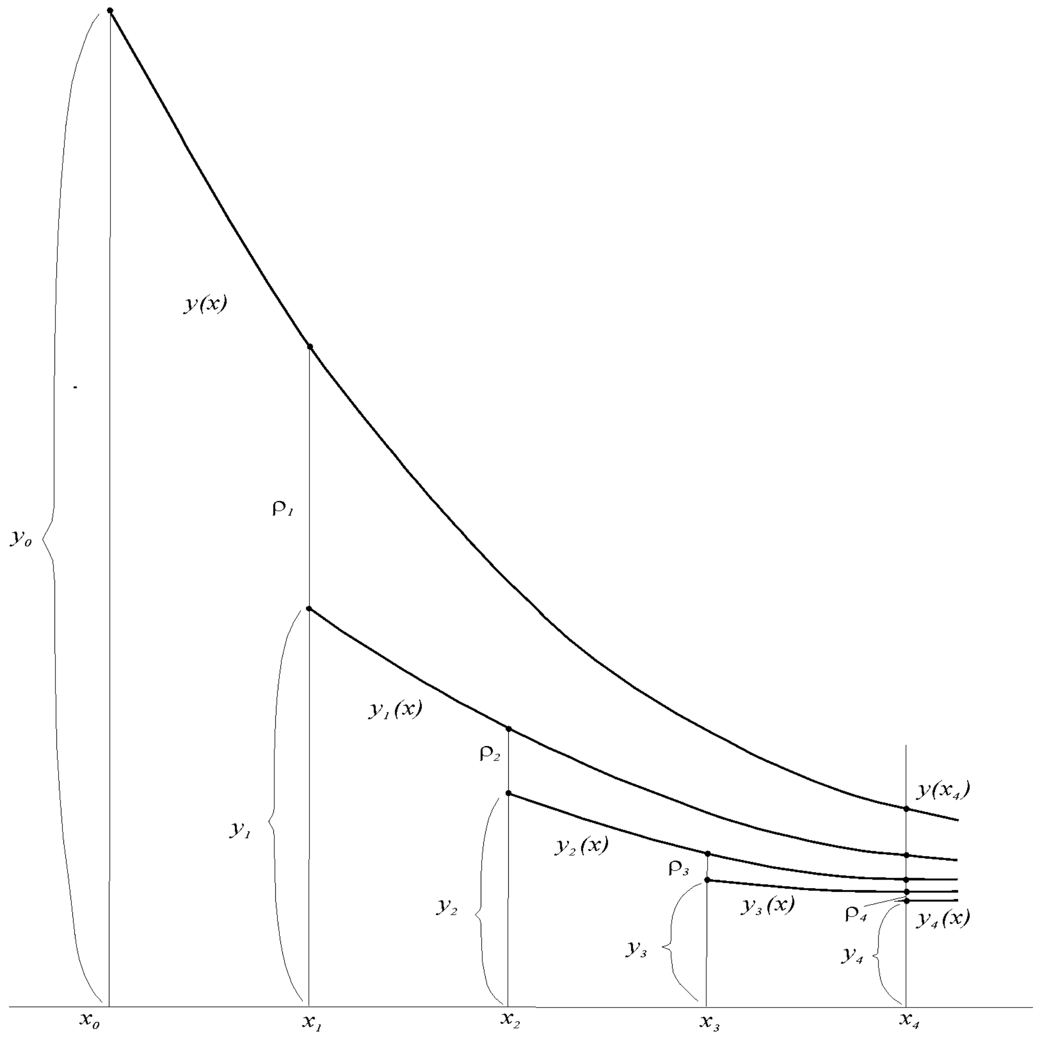


Рис. 8. $\dfrac{\partial f}{\partial y} < 0$. Интегральные кривые сближаются. Глобальная погрешность метода меньше суммы локальных погрешностей: $y(x_4) - y_4 < \rho_1 + \rho_2 + \rho_3 + \rho_4$

Из соотношений (40), (41) следует, что приближенное решение задачи Коши, полученное по методу Рунге–Кутта порядка s, сходится к точному решению задачи при $h \to 0$, если

$$n\delta \to 0, \quad |R_0| \to 0 \quad (||R_0|| \to 0). \tag{42}$$

Рассмотрим смысл каждого из трех слагаемых в (40), (41) и смысл условия (42).

Присутствие в (40), (41) слагаемого $|R_0|$ ($||R_0||$) означает, что погрешность от начального значения будет распространяться на все узлы сетки. Часть полной погрешности приближенного решения, зависящая от ошибки в начальных условиях и называемая *неустранимой погрешностью*, не будет превосходить $|R_0|\, e^{LX}$ $(||R_0||\, e^{MLX})$. Если начальные значения решения задаются точно, то этого члена не будет.

Второй член получается за счет того, что мы находим не точное решение задачи, а приближение к нему по формуле Рунге–Кутта. Это есть погрешность метода и она имеет порядок h^s.

Третий член получается за счет ошибок округления. Часть полной погрешности приближенного решения, источником которой являются ошибки округления, как уже указывалось, называется *вычислительной погрешностью*. Скорость возрастания вычислительной погрешности не

превосходит
$$\frac{e^{LX} n\delta}{nh} = \frac{e^{LX}\delta}{h} \qquad \left(\text{или} \quad \frac{\mathrm{e}^{\mathrm{MLX}}\delta}{\mathrm{h}}\right).$$

Если величина δ ограничена снизу: $0 < \delta_0 \leqslant \delta$ (а в практике вычислений на ЭВМ так обычно и бывает), и при этом длина шага h слишком мала, а значит, число шагов очень велико, то вычислительная погрешность может достигать больших значений.

В действительности же ошибки округления δ_j могут иметь различные знаки и частично компенсировать друг друга. Разные знаки могут иметь и погрешности метода ρ_j. Отдельные составляющие, входящие в полную погрешность R_n (38), могут давать отклонения от точного решения в разные стороны. Поэтому оценка (40) (или 41) по сравнению с (38) является завышенной. Кроме того, ее применение затруднено еще и из-за сложности определения величины C, выражаемой через производные высокого порядка от правой части $f(x, y)$ уравнения (1). Поэтому эта оценка на практике не используется для определения точности окончательного результата.

Из приведенных оценок можно сделать следующий вывод, подтверждаемый практикой численного решения на ЭВМ дифференциальных уравнений. Если заранее обеспечена необходимая малость неустранимой погрешности, то в полной погрешности преобладает либо погрешность метода, либо вычислительная погрешность. Погрешность метода может быть сделана сколь угодно малой за счет уменьшения шага h. Вычислительная погрешность может быть снижена за счет увеличения числа используемых в промежуточных вычислениях значащих цифр (*значащими цифрами* называются все цифры в записи числа, начиная с первой ненулевой), однако наши практические возможности в этом далеко не беспредельны из-за конечности разрядной сетки машины.

Если же погрешость (27) в начальных условиях велика, то неустранимая погрешность может также оказаться значительной и ее нельзя будет уменьшить никаким сокращением длины шага интегрирования, так как она не зависит от численного решения задачи. Неустранимую погрешность можно уменьшить только за счет более точного определения начальных условий. Поэтому остается только надеяться, что неустранимая погрешность будет незначительной по сравнению с другими видами погрешности.

Из вышесказанного следует, что вычислительный процесс должен быть организован таким образом, чтобы поддерживался баланс между всеми видами погрешности, составляющими полную погрешность приближенного решения. Этот баланс может быть достигнут, если надлежащим образом будут согласованы между собой требуемая точность решения задачи, точность задания начальных условий, порядок численного метода, величина шага интегрирования и используемая длина разрядной сетки ЭВМ.

3.4.3. Асимптотическая оценка погрешности метода

Пусть правая часть $f(x, y)$ уравнения (1) имеет непрерывные частные производные до порядка $s + 2$. Предположим, что задача Коши (1), (2) решается с постоянным шагом
$$x_{i+1} - x_i = h_i = h$$
методом Рунге–Кутта (7) порядка s, так что для локальной погрешности (8) метода справедлива оценка (9), которую запишем в следующем виде:
$$y(x_{n+1}) - y_{n+1} = \psi(x_n, y_n) h^{s+1} + O(h^{s+2}), \tag{43}$$
где
$$\psi(x_n, y_n) = \frac{1}{(s+1)!}\, \varphi_q^{(s+1)}(0).$$

Будем считать равными нулю погрешность начального условия и погрешности округлений: $R_0 = 0$ (27), $\delta_n = 0$ (30). Тогда для глобальной погрешности (29) метода имеет место асимптотическое представление

$$\varepsilon_n = z(x_n)h^s + O(h^{s+1}), \tag{44}$$

где

$$z(x_n) = \int\limits_{x_0}^{x_n} \psi(\xi, y(\xi))\, e^{\int\limits_{\xi}^{x_n} \frac{\partial f}{\partial y}(\tau, y(\tau))d\tau} d\xi\,. \tag{45}$$

Заметим, что $\psi(x, y) \not\equiv 0$, $z(x) \not\equiv 0$. Более того, если $\psi(x, y)$ не меняет знака, то $z(x) \neq 0$ ни при каком x. Первое слагаемое в (44)

$$z(x_n)h^s, \tag{46}$$

которое указывает малую величину главного порядка, называется *главным членом погрешности метода*. При достаточно малых значениях h, таких, что в (44) можно пренебречь членом $O(h^{s+1})$, для погрешности метода справедлива формула

$$\varepsilon_n \cong z(x_n)h^s. \tag{47}$$

3.4.4. Реальная область асимптотики

Рассмотрим, когда главный член (46) погрешности метода хорошо отражает полную погрешность (33) приближенного решения. Для этого необходимо, чтобы члены порядка $O(h^{s+1})$, входящие в (44), наряду с вычислительной погрешностью η_n были малы по сравнению с главным членом. При этом неустранимая погрешность ζ_n также должна быть малой по отношению к нему. Условия, обеспечивающие близость главного члена погрешности метода к полной погрешности приближенного решения, математически можно записать в виде:

$$R_0 = O(h^{s+1}), \quad \delta = O(h^{s+2}). \tag{48}$$

Если предположить, что условия (48) выполняются при $h \to 0$, то для полной погрешности приближенного решения имеет место асимптотическое разложение вида (44), а именно:

$$R_n = z(x_n)h^s + O(h^{s+1}). \tag{49}$$

При этом разложения (44) и (49) отличаются друг от друга только членами порядка $O(h^{s+1})$. При достаточно малых значениях h, таких, что в (49) можно пренебречь членом $O(h^{s+1})$, для полной погрешности справедлива формула

$$R_n \approx z(x_n)h^s. \tag{50}$$

Множество таких значений шага h называется *областью асимптотики.*

Однако в реальных условиях вычислительного процесса требования (48), как правило, не выполняются (по крайней мере, второе из них). Погрешность R_0 задания начальных условий вообще не зависит от длины шага интегрирования, а погрешности округления на шаге (30) при решении задачи на ЭВМ с фиксированной разрядной сеткой остаются ограниченными по абсолютной величине снизу

$$0 < \delta_0 \leqslant \delta.$$

Нарушение требований (48) при $h \to 0$ приводит к тому, что неустранимая погрешность будет ограничена снизу, а вычислительная погрешность может даже возрастать при $h \to 0$, в результате формулы (49) и (50) перестают быть справедливыми.

Поэтому в реальном процессе численного решения задачи Коши на ЭВМ множество значений шагов, для которых главный член (46) погрешности метода (29) хорошо представляет полную погрешность (33) приближенного решения (при достаточной малости неустранимой погрешности), т.е. для которых справедлива формула (50), ограничено не только сверху, но и снизу:

$$0 < \underline{h} < h < \overline{h}. \tag{51}$$

При переходе через верхнюю границу $\overline{h}$ в сторону увеличения значений шагов растет вклад членов в правой части формулы (49), содержащих высшие степени h, и формула (50) становится неверной. При переходе через нижнюю границу $\underline{h}$ в сторону уменьшения значений шагов увеличивается вычислительная погрешность η_n (32) и формула (50) снова перестает быть действительной. Таким образом, формула (50) для полной погрешности приближенного решения может оказаться несправедливой как при больших ($h > \overline{h}$), так и при очень малых ($h < \underline{h}$) значениях шага h.

Множество значений шагов (51) называется *реальной областью асимптотики.* Для фиксированной задачи (1), (2) и фиксированного метода решения (7) нижняя граница зависит от разрядной сетки вычислительной машины. Чем шире разрядная сетка, тем, вообще говоря, меньше нижняя граница реальной области асимптотики. Расширить разрядную сетку можно за счет выполнения вычислений с удвоенным числом значащих цифр.

3.5. Практические способы оценки погрешности приближенного решения

3.5.1. Апостериорная оценка глобальной погрешности метода

Предположим, что для погрешности приближенного решения справедливо асимптотическое разложение (49), (50). На этом разложении основывается важный для практики способ Рунге *апостериорной* оценки погрешности. Правило Рунге состоит в том, что решение задачи в некоторой точке x_n интервала интегрирования вычисляется дважды по одной и той же формуле (7) с разными малыми шагами, и полученные значения решения используются для получения апостериорной оценки погрешности. Обычно в качестве шагов выбирают h и $\dfrac{h}{2}$.

Допустим, что в точке x_n по формуле Рунге–Кутта (7) с постоянным шагом h вычислено решение $\bar{y}_n$. На основании (50) погрешность этого решения будет приближенно равна

$$y(x_n) - \bar{y}_n \approx z(x_n)h^s.$$

Затем, используя ту же формулу с шагом $\dfrac{h}{2}$, вычислим в точке x_n другое значение решения $\bar{\bar{y}}_n$, для которого потребуется в два раза больше шагов и погрешность которого будет приближенно выражаться равенством

$$y(x_n) - \bar{\bar{y}}_n \approx z(x_n)\Big(\frac{h}{2}\Big)^s.$$

Исключая из этих соотношений точное значение решения $y(x_n)$, имеем:

$$\bar{\bar{y}}_n - \bar{y}_n \approx z(x_n)h^s\Big(1 - \frac{1}{2^s}\Big).$$

Отсюда

$$z(x_n) \approx \frac{\bar{\bar{y}}_n - \bar{y}_n}{h^s(1 - \frac{1}{2^s})}. \tag{52}$$

Окончательно получаем оценки погрешности для приближенных значений решения:

$$\bar{R}_n = y(x_n) - \bar{y}_n \approx \frac{\bar{\bar{y}}_n - \bar{y}_n}{1 - \frac{1}{2^s}}, \tag{53}$$

$$\bar{\bar{R}}_n = y(x_n) - \bar{\bar{y}}_n \approx \frac{\bar{\bar{y}}_n - \bar{y}_n}{2^s - 1}. \tag{54}$$

Полученное приближенное значение можно уточнить, прибавив к нему величину главного члена погрешности, т.е. положив

$$y(x_n) \approx y_n = \bar{y}_n + \bar{R}_n,$$

или

$$y(x_n) \approx y_n = \bar{\bar{y}}_n + \bar{\bar{R}}_n.$$

При этом

$$y(x_n) - y_n = O(h^{s+1}).$$

В случае системы уравнений (23) правило Рунге оценки погрешности (53), (54) может быть записано в координатной форме следующим образом:

$$\bar{R}_n^i = y^i(x_n) - \bar{y}_n^i \approx \frac{\bar{\bar{y}}_n^i - \bar{y}_n^i}{1 - \dfrac{1}{2^s}}, \quad i = 1, 2, \dots, M; \tag{55}$$

$$\bar{\bar{R}}_n^i = y^i(x_n) - \bar{\bar{y}}_n^i \approx \frac{\bar{\bar{y}}_n^i - \bar{y}_n^i}{2^s - 1}, \quad i = 1, 2, \dots, M; \tag{56}$$

В качестве решения в точке x_n примем значение $\bar{\bar{y}}_n$ как более точное по сравнению с $\bar{y}_n$. Для него имеем оценку погрешности (54). Эта величина может быть как больше, так и меньше некоторого значения ε, являющегося наперед заданной *допустимой погрешностью*.

Если $|\bar{\bar{R}}_n| \leqslant \varepsilon$, то заданная точность приближенного решения достигается, в противном случае — нет[1]. Если точность не достигается, то необходимо решить вопрос, какую нужно взять длину шага, чтобы все-таки достигнуть заданной точности. Если же точность достигается, то естественно поставить такой вопрос: можно ли увеличить длину шага интегрирования для того, чтобы уменьшить объем вычислительной работы и одновременно с этим сохранить заданную точность? В обоих случаях такую величину шага h_ε можно определить, если положить

$$|z(x_n)|\, h_\varepsilon^s = \varepsilon.$$

Отсюда находим

$$h_\varepsilon = \sqrt[s]{\frac{\varepsilon}{|z(x_n)|}}. \tag{57}$$

Подставляя в (57) выражение (52) для $z(x_n)$, получаем

$$h_\varepsilon \cong \sqrt[s]{\frac{\varepsilon h^s (2^s - 1)}{|\bar{\bar{y}}_n - \bar{y}_n| 2^s}} = \frac{h}{2} \sqrt[s]{\frac{(2^s - 1)\varepsilon}{|\bar{\bar{y}}_n - \bar{y}_n|}}. \tag{58}$$

Из (58) следует, что если $|\bar{\bar{R}}_n| > \varepsilon$, то новое значение шага уменьшается:

$$h_\varepsilon < \frac{h}{2};$$

если $|\bar{\bar{R}}_n| < \varepsilon$, то новое значение шага увеличивается:

$$h_\varepsilon > \frac{h}{2}.$$

Таким образом, формула (58) дает более подходящее значение шага интегрирования.

[1]Контроль точности для системы уравнений (23) подробно излагается в п. 3.6.3.

3.5.2. Апостериорные оценки локальной погрешности метода

В предыдущем разделе мы познакомились с практическим способом оценки глобальной погрешности метода, когда решение задачи вычисляется в некоторой фиксированной точке интервала интегрирования. Теперь обсудим, как можно практически оценить локальную погрешность метода. Мы рассмотрим несколько способов, которые начнем с известного нам правила Рунге.

3.5.2.1. Оценка погрешности по правилу Рунге

Правило Рунге заключается в том, что по одной и той же формуле (7) вычисляются два приближения к решению в одной точке, но с разными малыми шагами, которые затем используются для получения апостериорной оценки погрешности.

Пусть в начальном узле x_0 решение y_0 известно. Выполним из точки x_0 один шаг h по формуле Рунге–Кутта (7). Полученное в точке $x_1 = x_0 + h$ решение обозначим через $\bar{y}_1$. Для локальной погрешности метода воспользуемся формулой (43). Если пренебречь членами порядка $O(h^{s+2})$, то для погрешности метода на данном шаге справедлива формула

$$y(x_0 + h) - \bar{y}_1 \approx \psi(x_0, y_0)h^{s+1}. \tag{59}$$

Затем вернемся в исходную точку x_0 и из точки x_0 сделаем подряд два шага, каждый величиной $h/2$. Сделав такой шаг первый раз, получим приближение $\hat{y}$ к решению в точке $x_0 + h/2$. Тогда погрешность на этом шаге согласно (43) будет равна

$$y\Big(x_0 + \frac{h}{2}\Big) - \hat{y} \approx \psi(x_0, y_0)\Big(\frac{h}{2}\Big)^{s+1}. \tag{60}$$

Сделав шаг $h/2$ второй раз, но уже из точки $x_0 + h/2$, получим приближение $\bar{\bar{y}}_1$ к решению в точке $x_1 = x_0 + h$. На этом втором шаге погрешность метода будет равна

$$\hat{y}(x_0 + h) - \bar{\bar{y}}_1 \approx \psi\Big(x_0 + \frac{h}{2}, \hat{y}\Big)\Big(\frac{h}{2}\Big)^{s+1}, \tag{61}$$

где $\hat{y}(x)$ — точное решение уравнения (1), удовлетворяющее условию $\hat{y}(x_0 + h/2) = \hat{y}$. Так как мы используем малую величину шага h, то точка $(x_0 + h/2, \hat{y})$ находится близко от точки (x_0, y_0). Поэтому в силу ограниченности по предположению частных производных ψ'_x и ψ'_y главный член погрешности метода на втором шаге будет таким же, как и на первом:

$$\hat{y}(x_0 + h) - \bar{\bar{y}}_1 \approx \psi(x_0, y_0)\Big(\frac{h}{2}\Big)^{s+1}. \tag{61$'$}$$

Разность

$$y(x) - \hat{y}(x)$$

двух решений уравнений (1) удовлетворяет линейному дифференциальному уравнению

$$(y(x) - \hat{y}(x))' = f'_y(x, \breve{y})(y(x) - \hat{y}(x)),$$

где $\breve{y}(x)$ заключено между $y(x)$ и $\hat{y}(x)$, и может быть представлена в виде

$$y(x) - \hat{y}(x) = \Big(y(x_0 + \frac{h}{2}) - \hat{y}(x_0 + \frac{h}{2})\Big)\, e^{\int\limits_{x_0+h/2}^{x} \frac{\partial f}{\partial y}(\xi, \breve{y}(\xi))d\xi}.$$

Отсюда следует, что

$$y(x_0+h)-\hat{y}(x_0+h)=\Big(y(x_0+\frac{h}{2})-\hat{y}(x_0+\frac{h}{2})\Big)\,e^{\int\limits_{x_0+h/2}^{x_0+h}\frac{\partial f}{\partial y}(\xi,\breve{y}(\xi))d\xi}=$$

$$=y(x_0+\frac{h}{2})-\hat{y}(x_0+\frac{h}{2})+O\Big(h(y(x_0+\frac{h}{2})-\hat{y}(x_0+\frac{h}{2}))\Big)\approx$$

$$\approx y\Big(x_0+\frac{h}{2}\Big)-\hat{y}\Big(x_0+\frac{h}{2}\Big)=y\Big(x_0+\frac{h}{2}\Big)-\hat{y}.$$

Теперь может быть найдена погрешность метода на двух последовательных шагах $h/2$:

$$y(x_0+h)-\bar{\bar{y}}_1=(y(x_0+h)-\hat{y}(x_0+h))+(\hat{y}(x_0+h)-\bar{\bar{y}}_1)\approx$$

$$\approx(y(x_0+\frac{h}{2})-\hat{y})+(\hat{y}(x_0+h)-\bar{\bar{y}}_1).$$

Подставляя в полученное соотношение правые части формул (60) и (61′), окончательно получаем

$$y(x_0+h)-\bar{\bar{y}}_1\approx 2\psi(x_0,y_0)\Big(\frac{h}{2}\Big)^{s+1}. \qquad (62)$$

Из (59) и (62) вытекают представления главных членов погрешности метода на шаге h и на двух последовательных шагах $\frac{h}{2}$:

$$\psi(x_0,y_0)h^{s+1}\approx\frac{\bar{\bar{y}}_1-\bar{y}_1}{1-\dfrac{1}{2^s}} \qquad (63)$$

и

$$2\psi(x_0,y_0)\Big(\frac{h}{2}\Big)^{s+1}\approx\frac{\bar{\bar{y}}_1-\bar{y}_1}{2^s-1}. \qquad (64)$$

Если в качестве приближения к решению в точке x_1 принять $\bar{y}_1$, то локальная погрешность метода будет равна

$$y(x_0+h)-\bar{y}_1\approx\frac{\bar{\bar{y}}_1-\bar{y}_1}{1-\dfrac{1}{2^s}}. \qquad (65)$$

Если в качестве приближения к решению в точке x_1 принять $\bar{\bar{y}}_1$, то погрешность метода на двух последовательных шагах $\frac{h}{2}$ будет равна

$$y(x_0+h)-\bar{\bar{y}}_1\approx\frac{\bar{\bar{y}}_1-\bar{y}_1}{2^s-1}. \qquad (66)$$

В случае системы уравнений (23) правило Рунге оценки локальной погрешности (65), (66) может быть записано в координатной форме следующим образом:

$$y^i(x_0+h)-\bar{y}_1^i\approx\frac{\bar{\bar{y}}{}_1^i-\bar{y}_1^i}{1-\dfrac{1}{2^s}},\qquad i=1,\ldots,M; \qquad (65')$$

$$y^i(x_0+h)-\bar{\bar{y}}{}_1^i\approx\frac{\bar{\bar{y}}{}_1^i-\bar{y}_1^i}{2^s-1},\qquad i=1,\ldots,M. \qquad (66')$$

Полученное приближенное значение $\bar{y}_1$ или $\bar{\bar{y}}_1$ можно уточнить, прибавив к нему величину главного члена погрешности, т.е. положив

$$y(x_1)\approx y_1=\bar{y}_1+\frac{\bar{\bar{y}}_1-\bar{y}_1}{1-\dfrac{1}{2^s}}, \qquad (67)$$

или

$$y(x_1) \approx y_1 = \bar{\bar{y}}_1 + \frac{\bar{\bar{y}}_1 - \bar{y}_1}{2^s - 1}. \tag{67'}$$

Тогда

$$y(x_1) - y_1 = O(h^{s+2}).$$

В данном способе оценки погрешности формула Рунге–Кутта (7) применяется три раза и требует $3q-1$ вычислений правой части $f(x,y)$ дифференциального уравнения (1). Поэтому при сложных и трудоемких для вычисления правых частях этот способ влечет большие вычислительные затраты.

3.5.2.2. Оценка погрешности на основе комбинации формул разных порядков точности

Рассмотрим второй способ оценки локальной погрешности метода. Этот способ также основан на использовании двух приближенных значений решения в одной точке. Однако эти приближения в отличие от правила Рунге вычисляются не по одной, а по двум формулам разных порядков точности p и s с одним и тем же шагом. Начнем обсуждение данного способа с общего случая, когда применяемые формулы Рунге–Кутта не связаны друг с другом, а потом перейдем к рассмотрению специально подобранных формул.

3.5.2.2.1. Комбинация независимых формул

Данный способ основан на комбинации двух формул вида (7) разных порядков точности p и s:

$$y_1^p = y_0 + \sum_{i=1}^{r} p_i k_i, \tag{68}$$

где

$$k_1 = hf(x_0, y_0),$$

$$k_i = hf(x_0 + \alpha_i h,\; y_0 + \sum_{j=1}^{i-1} \beta_{ij} k_j),$$

и

$$y_1^s = y_0 + \sum_{i=1}^{\tilde{r}} \tilde{p}_i \tilde{k}_i, \tag{69}$$

где

$$\tilde{k}_1 = hf(x_0, y_0),$$

$$\tilde{k}_i = hf(x_0 + \tilde{\alpha}_i h,\; y_0 + \sum_{j=1}^{i-1} \tilde{\beta}_{ij} \tilde{k}_j),$$

Пусть $p > s$, r ¿ $\tilde{r}$. Локальные погрешности в этих формулах будут такие:

$$\rho^p = y(x_0 + h) - y_1^p = O(h^{p+1})$$

и

$$\rho^s = y(x_0 + h) - y_1^s = O(h^{s+1}).$$

Из последних равенств следует оценка локальной погрешности формулы (69):

$$\rho^s = y_1^p - y_1^s + O(h^{p+1}). \tag{70}$$

Оставляя в (70) только члены главного порядка, получаем

$$\rho^s \cong y_1^p - y_1^s. \tag{71}$$

Полученная оценка погрешности требует $r + \tilde{r} - 1$ вычислений правой части уравнения (1).

3.5.2.2.2. Комбинация специально подобранных формул

Если коэффициенты в формулах (68) и (69) таковы, что

$$\alpha_i = \tilde{\alpha}_i, \quad \beta_{ij} = \tilde{\beta}_{ij}, \quad i = 1, \ldots, \tilde{r}, \tag{72}$$

то

$$k_i = \tilde{k}_i, \quad i = 1, \ldots, \tilde{r},$$

и для локальной погрешности (71) формулы (69) получается выражение следующего вида:

$$\rho^s \approx y_1^p - y_1^s = \sum_{i=1}^{r} q_i k_i, \tag{73}$$

где

$$\begin{aligned} q_i &= p_i - \tilde{p}_i, \qquad i = 1, \ldots, \tilde{r}, \\ q_i &= p_i, \qquad\qquad i = \tilde{r} + 1, \ldots, r. \end{aligned} \tag{73$'$}$$

Оценка (73) помимо тех значений правой части, которые вычисляются на текущем шаге h и входят в формулу (69), включает дополнительные значения k_i, $i = \tilde{r} + 1, \ldots, r$.

Такой подход к оценке локальной погрешности позволяет уменьшить по сравнению с правилом Рунге (65) и оценкой (71) количество вычислений правой части уравнения (1).

Оценка (73), как и более общая оценка (71), является асимптотической, так как она учитывает только члены главного порядка, и справедлива при достаточно малых размерах шага интегрирования.

В практике вычислений в качестве приближенного значения решения принимается значение y_1^p как имеющее более высокий порядок точности.

Величина

$$E = \sum_{i=1}^{r} q_i k_i, \tag{74}$$

где коэффициенты q_i определяются с помощью (73′), называется *контрольным членом*. В случае системы уравнений (23) контрольный член может быть записан в координатной форме следующим образом:

$$E^j = \sum_{i=1}^{r} q_i k_i^j, \quad j = 1, \ldots, M. \tag{74$'$}$$

3.5.2.2.3. Контрольные члены для методов Рунге–Кутта. Оценка погрешности в методах Мерсона, Инглэнда, Фельберга

Рассмотрим несколько примеров.

1). Методы (21)

$$y_1 = y_0 + \frac{1}{6}\,(k_1 + 4k_2 + k_3)$$

и (16)

$$y_1 = y_0 + k_2$$

удовлетворяют условию (72). Контрольный член (74) записывается в данном случае в виде:

$$E = \frac{1}{6}\,(k_1 - 2k_2 + k_3) \tag{75}$$

и имеет порядок $O(h^3)$. Здесь $p = 3$, $s = 2$, $r = 3$, $\tilde{r} = 2$. Если оценку локальной погрешности метода (16) вести по правилу Рунге (65), то для этого потребуется пять обращений к правой части вместо трех.

2). Классический метод Рунге–Кутта (22)

$$y_1 = y_0 + \frac{1}{6}\,(k_1 + 2k_2 + 2k_3 + k_4)$$

и метод второго порядка

$$y_1 = y_0 + \frac{1}{2}\,(-k_1 + 2k_2 + 2k_3 - k_4) \tag{76}$$

удовлетворяет условию (72). Контрольный член (74) записывается в данном случае в виде:

$$E = \frac{2}{3}\,(k_1 - k_2 - k_3 + k_4) \tag{77}$$

и имеет порядок $O(h^3)$. Здесь $p = 4$, $s = 2$, $r = 4$, $\tilde{r} = 4$. Он известен как контрольный член Егорова.

3). Классический метод Рунге–Кутта (22) и метод (16) также удовлетворяют условию (72). Контрольный член записывается в виде:

$$E = \frac{1}{6}\,(k_1 - 4k_2 + 2k_3 + k_4) \tag{78}$$

и имеет порядок $O(h^3)$. Здесь $p = 4$, $s = 2$, $r = 4$, $\tilde{r} = 2$.

4). Мерсон предложил следующую модификацию классического метода Рунге–Кутта:

$$y_1 = y_0 + \frac{1}{6}\,(k_1 + 4k_4 + k_5), \tag{79}$$

$$\begin{aligned}
k_1 &= hf(x_0, y_0),\\
k_2 &= hf\Big(x_0 + \frac{1}{3}h,\ y_0 + \frac{1}{3}k_1\Big),\\
k_3 &= hf\Big(x_0 + \frac{1}{3}h,\ y_0 + \frac{1}{6}k_1 + \frac{1}{6}k_2\Big),\\
k_4 &= hf\Big(x_0 + \frac{1}{2}h,\ y_0 + \frac{1}{8}k_1 + \frac{3}{8}k_3\Big),\\
k_5 &= hf\Big(x_0 + h,\ y_0 + \frac{1}{2}k_1 - \frac{3}{2}k_3 + 2k_4\Big).
\end{aligned}$$

Формула (79) и формула третьего порядка

$$y_1 = y_0 + \frac{1}{10}\,(k_1 + 3k_3 + 4k_4 + 2k_5) \tag{80}$$

удовлетворяют условию (72). Контрольный член (74) записывается в данном случае в виде

$$E = \frac{1}{30}\,(2k_1 - 9k_3 + 8k_4 - k_5) \tag{81}$$

и имеет порядок $O(h^4)$. Здесь $p = 4$, $s = 3$, $r = 5$, $\tilde{r} = 5$. Если бы контроль точности для метода третьего порядка производился по правилу Рунге (65), то это потребовало бы восьми вычислений правой части вместо пяти в методе Мерсона.

На более узком, чем (1), классе уравнений, а именно, линейных уравнений вида

$$y' = f(x, y) = ax + by + c \tag{82}$$

формула (80) имеет не третий, а пятый порядок точности. Порядок формулы (79) остается по-прежнему равным четырем. Поэтому величина

$$\rho = -E = -\frac{1}{30}\,(2k_1 - 9k_3 + 8k_4 - k_5) \tag{83}$$

служит главным членом локальной погрешности формулы (79) и имеет порядок $O(h^5)$. Таким образом, метод Мерсона (79) обладает той особенностью, что на классе уравнений (82) главный член его локальной погрешности выражается с помощью линейной комбинации только тех значений правой части, которые непосредственно входят в формулу (79). Следовательно, для получения оценки погрешности отпадает необходимость в дополнительных вычислениях.

5). Фельберг разработал множество методов, удовлетворяющих условию (72). Приведем предложенные им формулы четвертого и пятого порядков:

$$y_1^{(5)} = y_0 + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6, \tag{84}$$

$$y_1^{(4)} = y_0 + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5, \tag{85}$$

$$k_1 = hf(x_0, y_0),$$

$$k_2 = hf\Big(x_0 + \frac{1}{4}h,\ y_0 + \frac{1}{4}k_1\Big),$$

$$k_3 = hf\Big(x_0 + \frac{3}{8}h,\ y_0 + \frac{3}{32}k_1 + \frac{9}{32}k_2\Big),$$

$$k_4 = hf\Big(x_0 + \frac{12}{13}h,\ y_0 + \frac{1932}{2197}k_1 - \frac{7200}{2197}k_2 + \frac{7296}{2197}k_3\Big),$$

$$k_5 = hf\Big(x_0 + h,\ y_0 + \frac{439}{216}k_1 - 8k_2 + \frac{3680}{513}k_3 - \frac{845}{4104}k_4\Big),$$

$$k_6 = hf\Big(x_0 + \frac{1}{2}h,\ y_0 - \frac{8}{27}k_1 + 2k_2 - \frac{3544}{2565}k_3 + \frac{1859}{4104}k_4 - \frac{11}{40}k_5\Big).$$

Контрольный член (74) записывается в виде:

$$E = \frac{1}{360}k_1 - \frac{128}{4275}k_3 - \frac{2197}{75240}k_4 + \frac{1}{50}k_5 + \frac{2}{55}k_6 \tag{86}$$

и имеет порядок $O(h^5)$. Здесь $p = 5$, $s = 4$, $r = 6$, $\tilde{r} = 5$. Если оценку погрешности для метода четвертого порядка производить по правилу Рунге, то это потребует одиннадцати вычислений правой части вместо шести в методе Фельберга.

6). Инглэндом построены следующие формулы четвертого и пятого порядков:

$$y_1^{(5)} = y_0 + \frac{1}{336}\,(14k_1 + 35k_4 + 162k_5 + 125k_6), \tag{87}$$

$$y_1^{(4)} = y_0 + \frac{1}{6}\,(k_1 + 4k_3 + k_4), \tag{88}$$

$$
\begin{aligned}
&k_1 = hf(x_0, y_0),\\
&k_2 = hf\Big(x_0 + \frac{1}{2}h,\ y_0 + \frac{1}{2}k_1\Big),\\
&k_3 = hf\Big(x_0 + \frac{1}{2}h,\ y_0 + \frac{1}{4}(k_1 + k_2)\Big),\\
&k_4 = hf\,(x_0 + h,\ y_0 - k_2 + 2k_3),\\
&k_5 = hf\Big(x_0 + \frac{2}{3}h,\ y_0 + \frac{1}{27}\,(7k_1 + 10k_2 + k_4)\Big),\\
&k_6 = hf\Big(x_0 + \frac{1}{5}h,\ y_0 + \frac{1}{625}\,(28k_1 - 125k_2 + 546k_3 + 54k_4 - 378k_5)\Big).
\end{aligned}
$$

Контрольный член

$$
E = \frac{1}{336}\,(-42k_1 - 224k_3 - 21k_4 + 162k_5 + 125k_6) \qquad (89)
$$

имеет порядок $O(h^5)$. Здесь $p = 5$, $s = 4$, $r = 6$, $\tilde{r} = 4$. Выигрыш в количестве вычислений правой части по сравнению с правилом Рунге при оценке локальной погрешности метода такой же, как в методе Фельберга.

3.5.2.3.Оценка погрешности с помощью нелинейного контрольного члена

Для приведенных в предыдущем разделе методов оценка локальной погрешности выражалась с помощью линейной комбинации (74) нескольких значений правой части дифференциального уравнения. Существуют ли другие оценки локальной погрешности, отличные от (74) и содержащие меньшее количество значений k_i правой части? Оказывается, что такие оценки существуют. В качестве примера можно привести формулу типа Рунге–Кутта четвертого порядка точности:

$$
\begin{aligned}
&y_1 = y_0 + \frac{17}{162}k_1 + \frac{81}{170}k_3 + \frac{32}{135}k_4 + \frac{250}{1377}k_5,\\
&k_1 = hf(x_0, y_0),\\
&k_2 = hf\Big(x_0 + \frac{2}{9}h,\ y_0 + \frac{2}{9}k_1\Big),\\
&k_3 = hf\Big(x_0 + \frac{1}{3}h,\ y_0 + \frac{1}{12}k_1 + \frac{1}{4}k_2\Big),\\
&k_4 = hf\Big(x_0 + \frac{3}{4}h,\ y_0 + \frac{3}{128}\,(23k_1 - 81k_2 + 90k_3)\Big),\\
&k_5 = hf\Big(x_0 + \frac{9}{10}h,\ y_0 + \frac{9}{10000}\,(-345k_1 + 2025k_2 - 1224k_3 + 544k_4)\Big)
\end{aligned}
$$

и выражение для погрешности формулы на шаге

$$
\rho_1 = y(x_0 + h) - y_1 = O(h^5)
$$

в виде нелинейного контрольного члена Скрэтона:

$$
\rho_1 = \frac{uv}{w} + O(h^6),
$$

где

$$
\begin{aligned}
u &= -\frac{1}{18}k_1 + \frac{27}{170}k_3 - \frac{4}{15}k_4 + \frac{25}{153}k_5, \\
v &= \frac{19}{24}k_1 - \frac{27}{8}k_2 + \frac{57}{20}k_3 - \frac{4}{15}k_4, \\
w &= k_4 - k_1.
\end{aligned}
$$

Полученное значение решения y_1 можно уточнить, если прибавить к нему оценку локальной погрешности, т.е. в качестве приближенного решения взять сумму

$$y_1^{(5)} = y_1 + \rho_1.$$

В результате получаем новую формулу вида (5) пятого порядка точности

$$y_1^{(5)} = y_0 + \frac{17}{162}k_1 + \frac{81}{170}k_3 + \frac{32}{135}k_4 + \frac{250}{1377}k_5 + \frac{uv}{w},$$

использующую пять вычислений правой части на одном шаге. Но эта формула уже не является формулой типа Рунге–Кутта (7), поскольку величина

$$\Delta y_0 = y_1^{(5)} - y_0$$

не выражается линейно через k_i.

3.6. Интегрирование с переменным шагом. Автоматический выбор шага интегрирования

До сих пор мы рассматривали такой процесс решения задачи (1), (2), когда формула Рунге–Кутта применялась с одной и той же величиной шага интегрирования во всей области вычисления решения вне всякой зависимости от характера поведения решения. Это был либо шаг h, либо $h/2$ в случае повторного счета для оценки погрешности по правилу Рунге (53), либо h_ε (58). Какая бы из этих величин не использовалась, она не изменялась от точки к точке. В этом случае говорят, что решение задачи получено с *постоянным шагом* интегрирования. Применение *переменного шага* интегрирования позволяет учитывать характер поведения решения и уменьшить общее число шагов, сохранив при этом требуемую точность приближенного решения. Тем самым могут быть снижены объем работы и машинное время и замедлен рост вычислительной погрешности.

Имея в распоряжении перечисленные способы (65), (66), (71), (73) оценки локальной погрешности метода, величину шага интегрирования можно выбирать автоматически в процессе счета. При этом можно исходить из того, чтобы на каждый шаг приходилась приблизительно одинаковая погрешность. Наиболее простой и распространенный алгоритм автоматического выбора шага является предметом дальнейшего обсуждения.

3.6.1. Алгоритм выбора с помощью удвоения и деления шага пополам

Пусть ρ_{n+1} — оценка локальной погрешности метода на шаге h, допущенной при вычислении приближенного значения решения y_{n+1}^h в точке $x_n + h$. Если оценка превосходит некоторую наперед заданную границу ε:

$$|\rho_{n+1}| > \varepsilon, \tag{90}$$

то считается, что значение y_{n+1}^h решения не удовлетворяет предписанной точности и шаг h объявляется неприемлемым. Полученная точка $x_n{+}h$ и значение y_{n+1}^h исключаются из рассмотрения. Выбирается новое значение шага

$$h^{(1)} = \frac{h}{2},$$

и вновь по той же формуле Рунге–Кутта с шагом $h^{(1)}$ вычисляется новое значение решения $y_{n+1}^{h^{(1)}}$ в новой точке $x_n + h^{(1)}$. Пусть $\rho_{n+1}^{(1)}$ — оценка локальной погрешности метода на данном шаге $h^{(1)}$. Если оценка опять превосходит заданную границу

$$|\rho_{n+1}^{(1)}| > \varepsilon,$$

то точка $x_n + h^{(1)}$ и значение $y_{n+1}^{h^{(1)}}$ опять исключаются из рассмотрения, шаг снова делится пополам:

$$h^{(2)} = \frac{h^{(1)}}{2}$$

и вычисления повторяются. Так происходит до тех пор, пока при какой-то величине шага (обозначим ее через h_n) оценка локальной погрешности не станет меньше ε:

$$|\rho_{n+1}| \leqslant \varepsilon. \tag{91}$$

После этого считается, что решение дифференциального уравнения продолжено до точки $x_{n+1} = x_n + h_n$. Дальнейшее интегрирование уравнения производится из точки x_{n+1} с шагом h_{n+1}, который выбирается описанным ниже способом.

Если оценка локальной погрешности на шаге $h_n = x_{n+1} - x_n$ удовлетворяет неравенству

$$|\rho_{n+1}| < \frac{\varepsilon}{K}, \tag{92}$$

где K — некоторая константа, то считается, что достигнута точность, превышающая заданную, и шаг интегрирования удваивается:

$$h_{n+1} = 2h_n.$$

Если выполняется неравенство

$$\frac{\varepsilon}{K} \leqslant |\rho_{n+1}| \leqslant \varepsilon, \tag{93}$$

то считается, что полученное в точке x_{n+1} решение удовлетворяет заданной точности, и шаг интегрирования остается без изменения:

$$h_{n+1} = h_n.$$

Т а б л и ц а 1.

Формула Рунге–Кутта и ее порядок s	Формула для уточнения решения и ее порядок p	Формула для оценки локальной погрешности метода и порядок оценки	Константа удвоения шага K	Способ оценки локальной погрешности метода
(16) $s = 2$	(21) $p = 3$	(75) 3	8	Контрольный член
(76) $s = 2$	(22) $p = 4$	(77) 3	8	Контрольный член Егорова
(16) $s = 2$	(22) $p = 4$	(78) 3	8	Контрольный член
(80) $s = 3$	(79) $p = 4$	(81) 4	16	Контрольный член в методе Мерсона
(79) $s = 4$	(80) $p = 5$	(83) 5	32	Контрольный член в методе Мерсона для $y' = ax + by + c$
(85) $s = 4$	(84) $p = 5$	(86) 5	32	Контрольный член в методе Фельберга
(88) $s = 4$	(87) $p = 5$	(89) 5	32	Контрольный член в методе Инглэнда
(7) s	$\bar{\bar{y}}$ s	(65), (66) $s+1$	2^{s+1}	Правило Рунге

Таким образом, на тех участках изменения независимой переменной, где достигается высокая точность приближенного решения, шаг интегрирования возрастает, а там, где точность не достигается, шаг интегрирования сокращается до необходимых для ее достижения значений. Тем самым обеспечивается выбор величины шага в зависимости от характера поведения решения дифференциального уравнения. Константа K обычно полагается равной 2^ν, где ν — порядок используемой оценки локальной погрешности метода. Константы, определяющие переход к удвоению шага для различных формул Рунге–Кутта, приведены в таб. 1.

Иногда для того, чтобы сократить число неприемлемых шагов, в изложенный здесь алгоритм выбора шага вносится изменение, которое заключается в следующем. Если при продолжении решения из точки x_n в точку $x_{n+1} = x_n + h_n$ шаг интегрирования сокращался хотя бы один раз, то при выборе следующего значения шага h_{n+1} удвоение предыдущего шага h_n не происходит, даже если и выполняется соотношение (92).

3.6.2. Выбор максимальной для заданной точности длины шага

Рассмотрим еще один алгоритм выбора шага, применив ту же идею, которая была использована при выводе формулы (58).

Так как оценка ρ_{n+1} локальной погрешности метода равна с точностью до членов более высокого порядка малости главному члену локальной погрешности метода, то в силу (43)

$$\rho_{n+1} \cong \psi(x_n, y_n) h^{s+1}. \tag{94}$$

Соотношение (94) справедливо для всех оценок, выведенных в п.3.5.2. (см. также таб. 1.). Если оценка ρ_{n+1} погрешности превосходит заданную границу ε:

$$|\rho_{n+1}| > \varepsilon,$$

то считается, что на данном шаге h метод не достигает требуемой точности и вычисленное значение y_{n+1} решения вместе с точкой $x_n + h$ исключается из рассмотрения. В этом случае выбирается новый размер шага, но не последовательным делением пополам, как в выше описанном способе, а с помощью соотношения

$$h_\varepsilon = \alpha h, \tag{95}$$

где α находится из условия выполнения равенства

$$|\psi(x_n, y_n) h_\varepsilon^{s+1}| = \varepsilon. \tag{96}$$

Из (94), (96) получаем, что

$$\alpha^{s+1} = \frac{\varepsilon}{|\rho_{n+1}|}$$

и

$$\alpha = \sqrt[s+1]{\frac{\varepsilon}{|\rho_{n+1}|}}. \tag{97}$$

Здесь $\alpha < 1$ и новое значение шага

$$h_\varepsilon = \sqrt[s+1]{\frac{\varepsilon}{|\rho_{n+1}|}}\, h \tag{98}$$

меньше предыдущего. Далее по формуле Рунге–Кутта из точки x_n выполняется один шаг h_ε и вычисляется приближение $y_{n+1}^{h_\varepsilon}$ к решению дифференциального уравнения в точке $x_n + h_\varepsilon$.

Если первоначальная оценка ρ_{n+1} локальной погрешности метода не превосходит заданную границу ε:

$$|\rho_{n+1}| \leqslant \varepsilon,$$

то считается, что полученное приближение y_{n+1} к решению удовлетворяет требуемой точности, и значение $x_n + h$ независимой переменной принимается в качестве следующего узла x_{n+1} интервала интегрирования. Дальнейшее интегрирование уравнения осуществляется из точки x_{n+1} с шагом h_ε, который определяется с помощью соотношений (95), (97). Теперь $\alpha > 1$ и $h_\varepsilon > h$.

Преимущество данного алгоритма выбора шага заключается в большей гибкости по сравнению с описанным в предыдущем разделе способом. Напомним, что в нем при достижении требуемой точности абсолютная величина шага интегрирования либо увеличивается в два раза, либо не изменяется в зависимости от того, выполняется или нет неравенство (92). Если это неравенство не выполняется из-за незначительного превышения оценки $|\rho_{n+1}|$ над величиной ε/K, то шаг интегрирования не будет увеличиваться и останется прежним. В алгоритме, основанном на использовании формул (95), (97), имеется возможность увеличения шага в любое число α раз даже тогда, когда это число меньше двух. Это приводит к более сглаженному изменению шага интегрирования и, как следствие, к сокращению общего количества шагов и снижению вычислительных затрат.

В действительности берется несколько меньшее, чем определяемое с помощью (97), значение для α, например,

$$\alpha^* = 0,9\alpha = 0,9 \sqrt[s+1]{\frac{\varepsilon}{|\rho_{n+1}|}}, \tag{97$'$}$$

и, соответственно, меньшее по сравнению с (95), (98) значение шага интегрирования

$$h^*_\varepsilon = \alpha^* h. \tag{95$'$}$$

Это делается для того, чтобы избежать тех шагов, для которых не достигается требуемая точность.

3.6.3. Использование различных характеристик точности

Автоматический выбор шага интегрирования имеет важный аспект, о котором надо помнить при практической реализации алгоритма на ЭВМ. Этот аспект заключается в следующем.

Дело в том, что для достижения заданной точности может (в зависимости от алгоритма выбора шага) или происходить большое число делений пополам шага интегрирования, или величина α, вычисляемая по формуле (97), оказаться настолько малой, что вновь определяемая длина шага h не будет вызывать изменения независимой переменной, т.е. в условиях машинной арифметики будет выполняться равенство

$$x \oplus h = x, \tag{99}$$

где $\oplus$ означает машинную операцию арифметического сложения, а точнее, сложение чисел с плавающей точкой, т.е. чисел, представленных в следующей форме

$$x = \pm\beta^p(\alpha_1\beta^{-1} + \alpha_2\beta^{-2} + \cdots + \alpha_t\beta^{-t}), \tag{100}$$

здесь β — основание системы счисления, p — порядок числа, удовлетворяющий неравенству

$$\underline{p} \leqslant p \leqslant \bar{p},$$

где $\underline{p}$, $\bar{p}$ — минимальный и максимальный порядки в используемой плавающей системе представления чисел на ЭВМ. Равенство (99) будет выполняться тогда, когда текущее значение h станет по абсолютной величине меньше или минимального положительного числа, представимого на данной ЭВМ и равного $\sigma = \beta^{\underline{p}-1}$, или некоторого числа $r > 0$, равного расстоянию от x до соседнего справа (при $h > 0$) или слева (при $h < 0$) вещественного числа, которое представимо на ЭВМ[2]. Можно показать, что

$$r = r(x,t) = \beta^{1-t}\beta^{p-1} = \mathrm{macheps}\cdot\beta^{p-1} \leqslant \mathrm{macheps}\cdot|x|. \tag{101}$$

Величина $\mathrm{macheps} = \beta^{1-t}$, называемая *машинным эпсилон*, равна расстоянию от 1 до соседнего справа вещественного числа, представимого на ЭВМ. Таким образом, если шаг интегрирования h удовлетворяет условию

$$|h| > \max\Big\{\sigma, r(x,t)\Big\}, \tag{102}$$

то при этом h равенство (99) выполняться не будет. Без проверки условия (99) во время работы программы с автоматическим выбором шага может произойти зацикливание. Этого можно избежать, если вместо верхней границы локальной погрешности метода, которая может оказаться слишком малой в сравнении с порядком искомого решения, задавать число верных цифр в приближенном значении решения. Это позволяет более осторожно подходить к определению точности приближенного решения с учетом длины разрядной сетки машины.

Чтобы прояснить ситуацию, напомним определение верных цифр числа. Цифра a_k в приближенном числе

$$y^* = a_1\beta^{p-1} + a_2\beta^{p-2} + \cdots + a_m\beta^{p-m} + \cdots$$

[2]Интегрирование уравнения может выполняться не только слева направо, но и справа налево; в последнем случае шаг интегрирования отрицательный.

считается *верной*, если *абсолютная погрешность* A_{y^*} числа y^* удовлетворяет неравенству

$$A_{y^*} \leqslant \omega\beta^{p-k}, \tag{103}$$

где ω — некоторое число, удовлетворяющее условию $1/2 \leqslant \omega \leqslant 1$. Если $\omega = 1$, то абсолютная погрешность числа y^* не превосходит единицы разряда, соответствующего цифре a_k, т.е. β^{p-k}.

Ясно, что, задавая ограничение сверху для ошибки в виде допустимой абсолютной погрешности, мы тем самым фиксируем разряд, соответствующий самой младшей верной цифре числа, а не число верных цифр в нем. В результате количество требуемых верных цифр в приближенном решении становится зависимым от порядка p искомого решения и может превзойти длину t разрядной сетки вычислительной машины и быть не достижимым на данной ЭВМ.

Поясним сказанное на примере. Предположим, что допустимая абсолютная погрешность метода составляет $\dfrac{1}{2}\cdot 10^{-5}$, а точное решение задачи в некоторой точке интервала интегрирования достигает значения с десятичным порядком $p = 5$, например, $10\,000,33333\ldots$. Допустим, что вычисления ведутся на ЭВМ с семью десятичными знаками. Заданная верхняя граница абсолютной погрешности соответствует требованию, чтобы приближенное значение решения имело, по крайней мере, пять верхних знаков после запятой. Но так как решение имеет порядок $p = 5$, то общее количество верных десятичных цифр должно быть не менее десяти, что превышает возможности условий проведения счета. Вообще, если допустимая абсолютная погрешность равна $\omega\cdot 10^{-k}$, порядок решения равен p, а количество используемых при вычислении десятичных знаков равно t, то для того, чтобы заданная точность была достижима, необходимо выполнение соотношения:

$$t - p > k. \tag{104}$$

Как видно, для рассмотренного примера это соотношение не выполняется.

В таких случаях более целесообразно использовать в алгоритме выбора шага не абсолютную, а *относительную погрешность*

$$\frac{\rho}{|y|}, \tag{105}$$

так как требуя, чтобы решение имело k верных цифр (десятичных), мы тем самым требуем, чтобы относительная погрешность этого решения не превосходила $\omega\cdot 10^{1-k}$. Однако здесь надо следить за тем, чтобы решение не обращалось в нуль.

Более гибким является использование меры погрешности. *Мерой погрешности* приближенного значения y называется величина μ, определяемая соотношением

$$\mu = \begin{cases} \dfrac{\rho}{|y|}, & |y| > P; \\ \rho, & |y| \leqslant P, \end{cases} \tag{106}$$

где P — некоторое фиксированное положительное число. Контроль точности по мере погрешности состоит в том, что на тех участках интервала интегрирования, где абсолютная величина решения не превосходит некоторого значения P, контроль точности ведется по абсолютной погрешности, а там, где абсолютная величина решения превосходит это значение, контроль точности ведется по относительной погрешности.

Другим эффективным средством может служить ограничение на число последовательных делений шага интегрирования, совершаемых в одной точке, или ограничение снизу на величину шага интегрирования. Например, ограничивая количество делений двадцатью, мы допускаем максимальное уменьшение шага в 10^6 раз. В большинстве случаев этого достаточно, если при этом, конечно, не нарушается требование (102), обеспечивающее невыполнение условия (99). Если и после двадцати делений точность по-прежнему не достигается, то, как правило, это свидетельствует о том, что метод Рунге–Кутта не подходит для решения данной задачи (конечно, если в программе нет ошибок!). Такая ситуация обычно встречается при интегрировании дифференциальных уравнений, у которых

$$\max\left|\frac{\partial f}{\partial y}\right| \gg 1.$$

К таким уравнениям относятся уравнения и системы уравнений, правая часть которых имеет большую константу Липшица

$$\left|f^i(x, y^1, \ldots, y^M) - f^i(x, z^1, \ldots, z^M)\right| \leqslant L\sum_{i=1}^{M}|y^i - z^i|, \tag{107}$$

$$L \gg 1, \tag{107$'$}$$

в частности, жесткие системы (см. п. 3.7).

Указанный прием позволяет получить в процессе счета полезную информацию о характере решаемой задачи, обнаружить сильное измельчение шага интегрирования, избежать большого увеличения общего числа шагов и, как следствие, чрезмерного возрастания машинного времени.

Для системы уравнений (23) с проверкой на точность могут вычисляться либо все компоненты решения, либо некоторые из них, в частности, одна компонента. При этом контроль точности может вестись покомпонентно или по норме. В первом случае каждая компонента проверяется на точность отдельно от остальных, причем для разных компонент могут использоваться как различные характеристики точности (абсолютная, относительная погрешность, мера погрешности), так и разные допустимые значения погрешности. В последнем варианте ε — вектор $(\varepsilon^1, \varepsilon^2, \ldots, \varepsilon^{M'})$, длина которого M' равна количеству проверяемых на точность компонент решения, а неравенства (90), (92), (93) будут своими для каждой компоненты решения, а именно:

$$|\rho_{n+1}^{i_j}| > \varepsilon^j, \tag{108}$$

$$|\rho_{n+1}^{i_j}| < \frac{\varepsilon^j}{K}, \tag{109}$$

$$\frac{\varepsilon^j}{K} \leqslant |\rho_{n+1}^{i_j}| \leqslant \varepsilon^j. \tag{110}$$

Здесь $j = 1, \ldots, M'$, $M' \leqslant M$, i_j — номера проверяемых на точность компонент, $1 \leqslant i_j \leqslant M$. Решение о деление шага пополам принимается, если условие (108) выполняется хотя бы для одной компоненты, а решение об удвоении шага принимается тогда, когда условие (109) выполняется для всех этих компонент.

Те компоненты решения, которые проверяются на точность по мере погрешности (106), могут иметь каждая свое значение P для перехода от абсолютной погрешности к относительной и обратно. В этом случае P — вектор $(P^1, P^2, \ldots, P^{M''})$, длина которого M'' равна количеству проверяемых на точность по мере погрешности компонент решения, а мера погрешности определяется отдельно для каждой компоненты:

$$\mu^{l_j} = \begin{cases} \dfrac{\rho^{l_j}}{|y^{l_j}|}, & |y^{l_j}| > P^j; \\ \rho^{l_j}, & |y^{l_j}| \leqslant P^j. \end{cases} \tag{111}$$

Здесь $j = 1, \ldots, M''$, $M'' \leqslant M' \leqslant M$, l_j — номера компонент, проверяемых на точность по мере погрешности, $i_1 \leqslant l_j \leqslant i_{M'}$.

Контроль точности по норме означает, что контролируется некоторая норма оценки погрешности $||\rho||$. Часто используются нормы:

$$||\rho||_\infty = \max_{1 \leqslant i \leqslant M} |\rho^i|,$$

$$||\rho||_1 = \sum_{i=1}^{M} |\rho^i|,$$

$$||\rho||_2 = \sqrt{\sum_{i=1}^{M} |\rho^i|^2}.$$

В этом случае ε — скаляр, а неравенства (90), (92), (93) и соотношение (97′) записываются соответственно в виде:

$$||\rho_{n+1}|| > \varepsilon,$$

$$||\rho_{n+1}|| < \frac{\varepsilon}{K},$$

$$\frac{\varepsilon}{K} \leqslant ||\rho_{n+1}|| \leqslant \varepsilon,$$

$$\alpha^* = 0,9\alpha = 0,9 \sqrt[s+1]{\frac{\varepsilon}{||\rho_{n+1}||}}.$$

Из вышеизложенного следует, что формулы Рунге–Кутта очень хорошо приспособлены для интегрирования с переменным шагом, так как они позволяют легко менять шаг интегрирования и при этом не требуют никаких дополнительных вычислений и преобразований. Этим они выгодно отличаются от других методов, таких как конечно-разностные методы (см. п.4), в которых при изменении шага интегрирования требуются дополнительные вычислительные затраты.

3.7. Чего не могут явные методы Рунге–Кутта?

Рассмотрим случай линейных уравнений

$$y' = \lambda y, \tag{112}$$

$$y(0) = y_0 \tag{113}$$

с $\lambda < 0$. Для метода Эйлера имеем

$$y_{n+1} = y_n + h\lambda y_n = (1 + \lambda h) y_n. \tag{114}$$

Точное решение задачи (112), (113)

$$y(x) = e^{\lambda x} y_0$$

монотонно убывает, $y(x) \to 0$, сохраняя знак, когда x растет. Естественно требовать, чтобы и решение разностной задачи (114), (113) тоже обладало таким свойством. Очевидно, что условием такого поведения решения является выполнение неравенства

$$0 < 1 + \lambda h < 1,$$

или

$$|\lambda|\, h < 1. \tag{115}$$

Для метода второго порядка точности (14) имеем:

$$y_{n+1} = y_n + \frac{1}{2}(\lambda h y_n + \lambda h(y_n + \lambda h y_n)) = y_n + \lambda h y_n + \frac{\lambda^2 h^2}{2} y_n = \left(1 + \lambda h + \frac{\lambda^2 h^2}{2}\right) y_n. \qquad (116)$$

Чтобы решение y_n задачи (116), (113) монотонно убывало, необходимо, чтобы

$$0 < 1 + \lambda h + \frac{\lambda^2 h^2}{2} < 1.$$

Отсюда следует неравенство, которому должна удовлетворять длина шага h:

$$-2 < \lambda h < 0,$$

или

$$|\lambda|\, h < 2. \qquad (117)$$

Такое же условие (117) на величину шага интегрирования получается и для метода (16). Ограничения, аналогичные (115) и (117), должны выполняться, хотя и с несколько большей константой, и для методов Рунге–Кутта более высокого порядка (7), которые, подобно (114) и (116), могут быть записаны в виде

$$y_{n+1} = F_q(\lambda h) y_n, \qquad (118)$$

где $F_q(\lambda h)$ — многочлен степени q от λh.

Таким образом, применение методов типа Рунге–Кутта (7) для решения задачи (112), (113) возможно только при выполнении условия

$$|\lambda|\, h \leqslant \text{const}. \qquad (119)$$

Рассмотрим систему линейных уравнений с постоянными коэффициентами

$$y' = Ay, \quad A = (a_{ij}), \quad i, j = 1, \ldots, M, \qquad (120)$$

$$y(0) = y_0, \quad y_0 = (y_0^i), \quad i = 1, \ldots, M. \qquad (121)$$

Можно показать, что применение явных методов типа Рунге–Кутта (24) к (120), (121) приводит к соотношению

$$y_{n+1} = F_q(Ah) y_n, \qquad (122)$$

где $F_q(Ah)$ — многочлен степени q от матрицы Ah.

Пусть матрица A является матрицей простой структуры, т.е. имеет M линейно независимых собственных векторов. Пусть λ_i, $\mathbf{e}_i$ — собственные значения и соответствующие им собственные векторы матрицы A. Тогда общее решение задачи (120), (121) может быть представлено в виде

$$y(x) = \sum_{i=1}^{M} C_i e^{\lambda_i x} \mathbf{e}_i. \qquad (123)$$

При интегрировании линейной системы (120), (121) каждая составляющая

$$C_i e^{\lambda_i x} \mathbf{e}_i$$

решения (123), пропорциональная одному из собственных векторов, интегрируется независимо от остальных компонент. В силу (122) имеем

$$y_{n+1} = F_q(Ah) y_n = \sum_{i=1}^{M} F_q(Ah) C_n^i \mathbf{e}_i = \sum_{i=1}^{M} F_q(\lambda_i h) C_n^i \mathbf{e}_i = \sum_{i=1}^{M} C_{n+1}^i \mathbf{e}_i. \qquad (124)$$

Формулы преобразования коэффициентов C_n^i

$$C_{n+1}^i = F_q(\lambda_i h) C_n^i \tag{125}$$

совпадают с формулами (118) численного интегрирования уравнения (112) с $\lambda = \lambda_i$. Поэтому условие (119) с $\lambda = \lambda_i$, $i = 1, \ldots, M$, должно выполняться также и для системы уравнений (120). Однако при интегрировании системы с большими по модулю отрицательными собственными значениями $|\lambda_j| \gg 1$ требование выполнения условия (119) является слишком обременительным, так как оно влечет необходимость применения очень малого шага интегрирования

$$h \leqslant \frac{\text{const}}{|\lambda_j|}.$$

Существенно, что малый шаг интегрирования не может быть увеличен даже тогда, когда быстро изменяющиеся составляющие

$$C_j e^{\lambda_j x} \mathbf{e}_j$$

станут ничтожно малыми и решение (123) задачи (120), (121) будет характеризоваться медленно изменяющимися компонентами с малыми по модулю производными. Поэтому использование явных методов Рунге–Кутта требует огромного объема вычислений, что делает их для таких задач практически непригодными.

Для жестких дифференциальных уравнений характерно поле направлений, изображенное на рис. 9. На нем иллюстрируется поведение явных методов типа Рунге–Кутта на примере метода Эйлера, когда они применяются к жестким уравнениям. Эти методы дают беспорядочные колебания с быстро возрастающей амплитудой, ничего общего не имеющие с точным решением задачи. Для таких задач применяются специально сконструированные *неявные* численные методы, напрмер, неявный метод Эйлера

$$y_{n+1} = y_n + hf(x_{n+1}, y_{n+1}). \tag{126}$$

Подставляя вместо $f(x_{n+1}, y_{n+1})$ правую часть уравнения (112), получаем

$$y_{n+1} = y_n + h\lambda y_{n+1},$$

отсюда

$$y_{n+1} = F(\lambda h) y_n,$$

где

$$F(\lambda h) = \frac{1}{1 - \lambda h}.$$

Так как $\lambda < 0$, то $0 < F(\lambda h) < 1$. Таким образом, применение формулы (126) обеспечивает монотонное убывание y_n. Тем самым обеспечивается монотонное убывание всех составляющих

$$C_n^i \mathbf{e}_i$$

приближенного решения (124) задачи (120), (121), для которых $\lambda_i < 0$.

Рис. 10 иллюстрирует использование неявного метода Эйлера (126) для интегрирования жесткого уравнения.

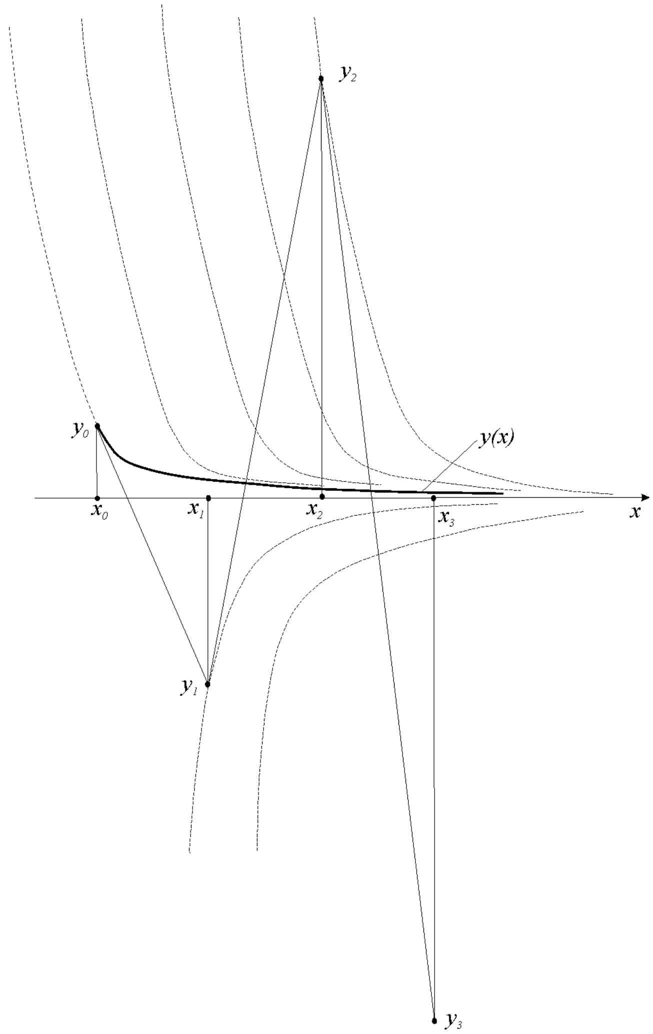


Рис. 9. Поведение ломаной Эйлера для жесткого уравнения, когда не соблюдается условие $|\lambda|h < 1$

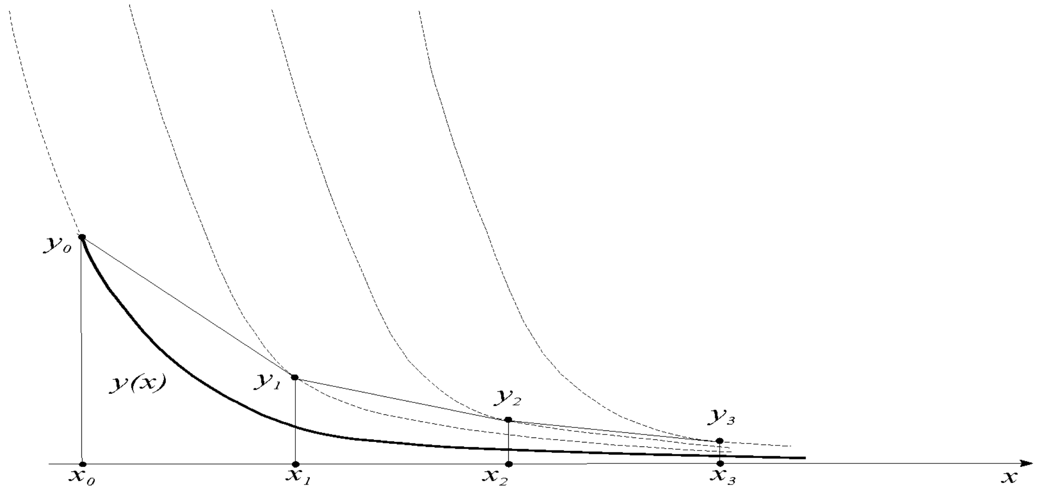


Рис. 10. Поведение неявного метода Эйлера

4. Другие методы численного решения дифференциальных уравнений

Методы Рунге–Кутта составляют важный, но не единственный класс численных методов решения обыкновенных дифференциальных уравнений. Широкое практическое применение находят так называемые *многошаговые*, или *конечно-разностные*, методы, в которых при вычислении решения в текущем узле используются значения решения и его производной в нескольких предыдущих узлах. Многошаговые методы позволяют уменьшить число вычислений правых частей на одном шаге, что является существенным достоинством этих методов, особенно, если правые части громоздкие и требуют большого числа арифметических операций. Однако многошаговые методы требуют применения специальной процедуры в начале интегрирования (так называемой процедуры разгона) и более трудоемки при изменении шага интегрирования. К много- шаговым методам относятся методы Адамса, методы Милна, методы Хемминга.

Другим важным классом методов являются методы экстраполяции. В этих методах в каждом узле интегрирования по явной формуле невысокого порядка вычисляются несколько приближенных значений решения с различными шагами. На основе полученных значений с помощью полиномиальной или дробно-рациональной экстраполяции строится более точное значение решения. Методы экстраполяции очень эффективны, если требуется высокая точность приближенного решения. Примером метода экстраполяции служит метод Грэгга-Булирша-Штера.

Имеются и другие методы.

Однако, все эти методы не предназначены для решения жестких задач. Для решения таких задач также существуют различные методы. К ним относятся наиболее популярный метод Гира, неявные методы Рунге–Кутта, методы Розенброка и другие. Эти методы более трудоемки и опираются на использование матрицы Якоби правой части исходной системы дифференциальных уравнений. Практике работы с некоторыми из перечисленных методов будут посвящены специальные задания.

5. Этапы выполнения задания

Опираясь на рассмотренные в предыдущих разделах теоретические и практические аспекты численного интегрирования дифференциальных уравнений одношаговыми методами, займемся изложением того, в чем же заключается содержание предлагаемого студентам для выполнения на ЭВМ задания.

Каждому студенту предлагается решить конкретную задачу Коши, определяемую одним из вариантов, перечень которых приводится в п. 8.2. Для решения поставленной задачи требуется выполнить одно из двух заданий А или Б. Начнем с изложения задания А.

Задание А состоит в вычислении приближенного решения данной задачи Коши методом Рунге–Кутта с заданной точностью, используя автоматический выбор шага интегрирования и различные способы оценки точности приближенного решения. Оно распадается на следующие этапы.

1. Вычисление решения задачи Коши по заданной формуле Рунге–Кутта с автоматическим выбором шага интегрирования и оценкой локальной погрешности метода ρ_n по правилу Рунге (66). Решение вычисляется в узлах некоторой, вообще говоря, неравномерной сетки узлов, которая заранее не задается, а строится автоматически в процессе численного интегрирования. Узлами такой сетки служат те точки x_n отрезка интегрирования, в которых выполняется практический критерий (91) достижения заданной точности. В каждом узле, кроме того, вычисляется точное решение дифференциального уравнения, печатаются полученные результаты и проверяется, не превосходит ли полная погрешность приближенного решения заданной величины (допустимой погрешности). После вычисления решения в конце интервала интегрирования необходимо обработать и распечатать накопленную в процессе интегрирования статистическую информацию об используемом алгоритме, а именно: количество вычислений правой части дифференциального уравнения, количество выполненных шагов, средний шаг, число шагов, на которых полная погрешность превосходит заданную (хотя и выполняется практический критерий

достижения точности) и т.п. (см. подпрограмму обработки результатов и процедуру интегрирования на отрезке с автоматическим выбором шага в разделе 6).

2. Второй этап полностью совпадает с первым этапом с той лишь разницей, что оценка локальной погрешности метода определяется не по правилу Рунге, а с помощью комбинации двух независимых формул (68) и (69) разного порядка точности.

3. Третий этап полностью совпадает с первыми двумя с той лишь разницей, что оценка локальной погрешности и само решение вычисляются с помощью формулы с контрольным членом (73), (73′).

4. На четвертом этапе задача Коши решается с постоянным шагом интегрирования, который равен среднему арифметическому значению шагов, полученному после первого этапа. Таким образом, на данном этапе решение вычисляется на равномерной сетке узлов, шаг которой равен шагу численного интегрирования. После каждого шага определяются точное решение уравнения, полная (истинная) погрешность, оценка глобальной погрешности по правилу Рунге (53) и проверяется, достигается ли в каждом узле требуемая точность; ведется также подсчет числа точек, в которых заданная точность не достигается. После окончания интегрирования накопленная статистическая информация распечатывается в том же виде, как и при автоматическом выборе шага (см. подпрограмму обработки результатов и процедуру интегрирования на отрезке с постоянным шагом в разделе 6).

Варианты методов для задания А представлены в п. 8.3.

5. На пятом этапе задача Коши решается по какой-нибудь одной из библиотечных подпрограмм, имеющихся в штатном программном обеспечении используемой ЭВМ.

6. Строятся графики точного решения, полной погрешности приближенного решения и составляется отчет по заданию. О том, каким должно быть содержание отчета, говорится в разделе 7.

Перейдем к изложению задания Б.

Задание Б состоит в вычислении решения задачи Коши методом Рунге–Кутта с заданной точностью, используя различные способы автоматического выбора шага интегрирования и один из способов оценки локальной погрешности метода. Задание Б распадается на следующие этапы.

1. Вычисление решения задачи Коши по заданной формуле Рунге–Кутта с автоматическим выбором шага интегрирования с помощью удвоения и деления шага пополам в зависимости от величины локальной погрешности метода. Локальная погрешность метода оценивается при этом либо по правилу Рунге, либо с помощью контрольного члена. В остальном первый этап полностью аналогичен первому этапу для задания А.

2. Второй этап полностью совпадает с первым этапом с той лишь разницей, что автоматический выбор шага интегрирования ведется не с помощью удвоения и деления шага пополам, а с помощью выбора оптимального его значения по формулам (95′) и (97′), т.е. максимального для заданной точности размера шага.

3. На третьем этапе задача Коши решается с постоянным шагом, который равен максимальному среднему арифметическому шагов, сделанных на первом и втором этапах. В остальном третий этап полностью аналогичен соответствующему этапу для задания А.

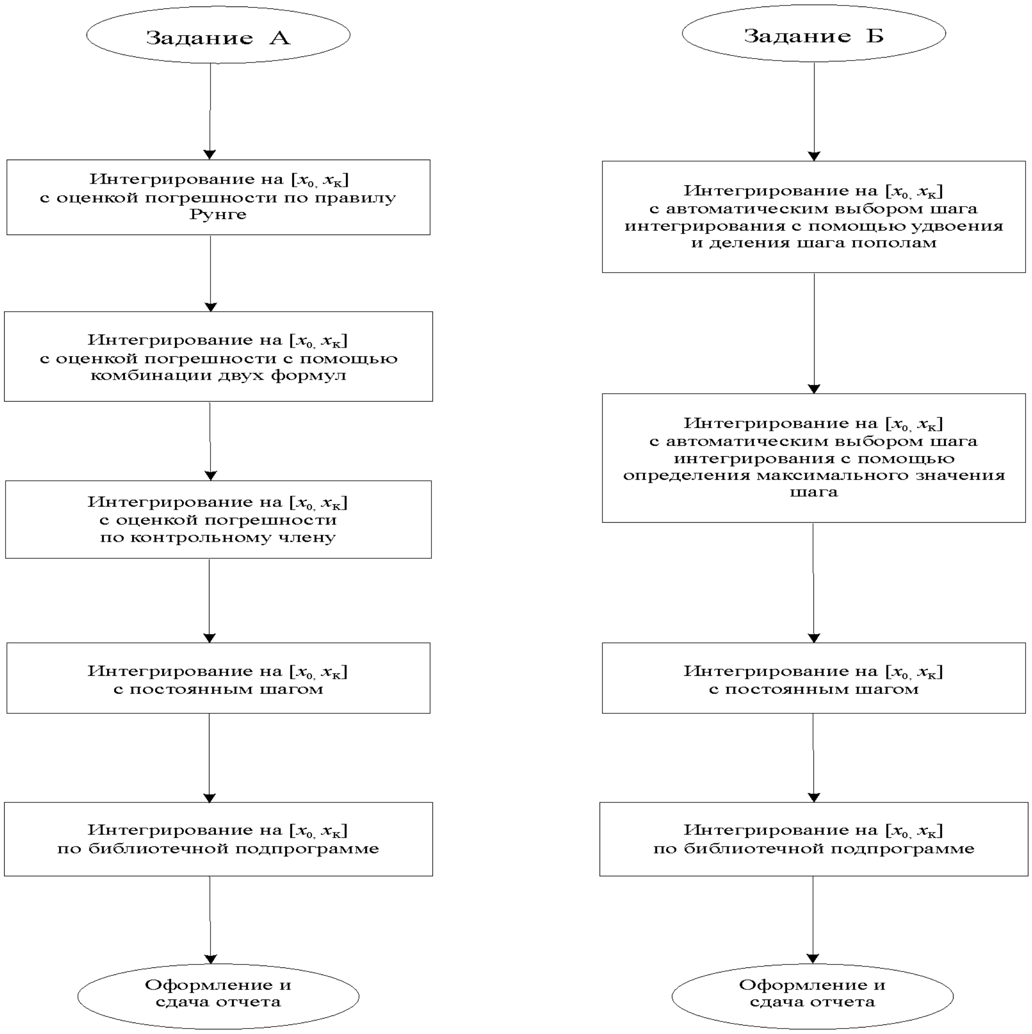
Дальнейшие этапы выполнения задания такие же, как этапы 5 и 6 для задания А.

Варианты методов для задания Б представлены в п. 8.4.

6. Структура программы и численные результаты

Для выполнения задания требуется составить программу и провести по ней необходимые расчеты. При составлении программы следует придерживаться предлагаемой модульной структуры и использовать ниже описываемые процедуры.

1). Процедура вычисления правой части дифференциального уравнения. Процедура вычисляет правую часть $z = f(x, y)$ дифференциального уравнения (1) при произвольных допустимых значениях x и y и ведет подсчет числа вычислений правой части. Процедура может



быть оформлена как подпрограмма с первым оператором

$$\text{SUBROUTINE} \quad \text{F(X, Y, Z)}$$

или подпрограмма-функция с первым оператором

$$\text{FUNCTION} \quad \text{F(X, Y)}$$

Подсчет числа вычислений правой части может быть организован прибавлением единицы к переменной NDER из общего блока

$$\text{COMMON} \quad /\ldots/\text{NDER}$$

которое выполняется при каждом обращении к процедуре F. Перед началом интегрирования дифференциального уравнения переменной NDER в головной программе должно быть присвоено нулевое значение.

Замечание. Приводимые здесь и далее имена подпрограмм и подпрограмм-функций носят условный характер и могут быть изменены.

2). Процедура выполнения одного шага численного интегрирования. Процедура выполняет из произвольной точки (x, y) в области определения правой части $f(x, y)$ один шаг h по формуле Рунге–Кутта (7) и вычисляет решение в точке $x + h$. Содержит обращение к процедуре вычисления правой части. В задании А используются три формулы Рунге–Кутта;

две из этих формул — без контрольного члена, а третья — с контрольным членом. Поэтому здесь необходимо составить две такие процедуры, выполняющие один шаг по первой и второй формулам. В задании Б составляется одна такая процедура, если используется формула без контрольного члена.

Процедура может быть оформлена как подпрограмма с первым оператором

$$\mathrm{SUBROUTINE} \quad \mathrm{RKS(F, X, Y, H)}$$

На выходе из подпрограммы значение переменной Y полагается равным значению решения, вычисленному в точке $x+h$. Переменные X и H остаются без изменения. Эта процедура может быть оформлена и как подпрограмма-функция с первым оператором

$$\mathrm{FUNCTION} \quad \mathrm{RKS(F, X, Y, H)}$$

причем на выходе из подпрограммы-функции значения всех переменных остаются без изменения.

3). Процедура выполнения одного шага с оценкой погрешности. Процедура выполняет то же, что и процедура RKS, и при этом вычисляет оценку локальной погрешности ρ метода на шаге одним из трех способов:

- по правилу Рунге (65) или (66);
- с помощью соотношения (71), получаемого применением двух формул (68), (69);
- по контрольному члену (73), ($73'$).

В задании А используются три способа оценки локальной погрешности метода. В этом случае необходимо составить три процедуры RKSE1, RKSE2, RKSE3, выполняющие один шаг и вычисляющие оценку погрешности по каждому из указанных способов. При этом процедура RKSE1 содержит вызов процедуры RKS1, а процедура RKSE2 — вызовы процедур RKS1 и RKS2.

Процедура оформляется как подпрограмма с первым оператором

$$\mathrm{SUBROUTINE} \quad \mathrm{RKSE(F, X, Y, H, E)}$$

На выходе из подпрограммы значение переменной Y полагается равным значению решения, вычисленному по формуле Рунге–Кутта в точке $x+h$, а переменной E присваевается оценка локальной погрешности метода на данном шаге h. Параметры X и H остаются без изменения.

4). Подпрограмма обработки результатов. В этой подпрограмме нужно предусмотреть вычисление точного решения $y(x)$ дифференциального уравнения (1), (2) при произвольном допустимом значении аргумента x, вычисление полной (истинной) погрешности $R_n = y(x_n) - y_n$ приближенного решения y_n, полученного по формуле Рунге–Кутта, подсчет общего количества N выполненных шагов, числа NF "неудачных" шагов h_n^*, т.е. тех, для которых полная (истинная) погрешность превосходит заданную ε, и их суммарной длины $\mathrm{XF} = \sum\limits_n h_n^*$, печать текущей строки таблицы результатов в следующем виде:

при интегрировании с автоматическим выбором шага —

$$x_n \qquad y_n \qquad y(x_n) \qquad \mathrm{R}_n = y(x_n) - y_n \qquad h = x_n - x_{n-1}$$

при интегрировании с постоянным шагом —

$$x_n \qquad y_n \qquad y(x_n) \qquad \mathrm{R}_n = y(x_n) - y_n \qquad \bar{\mathrm{R}}_n$$

Здесь $\bar{\mathrm{R}}_n$ – оценка глобальной погрешности по правилу Рунге. Для распечатки таблиц подпрограмма должна располагать значениями текущего узла интервала интегрирования $x = x_n$, приближенного решения y_n в нем, предыдущего узла $xp = x_{n-1}$, оценкой $\bar{\mathrm{R}}_n$ глобальной погрешности решения y_n, полученной по правилу Рунге (53) для постоянного шага, а также значениями переменных N, NF и XF для хранения, соответственно, общего количества шагов, числа "неудачных" шагов и их суммарной длины. Передача этих значений в подпрограмму может быть организована через общий блок, который описывается в данной подпрограмме и вызывающих ее подпрограммах:

$$\mathrm{COMMON} \quad /\ldots/\mathrm{X, Y, XP, N, NF, XF, EPS, R}$$

Если полная погрешность вычисленного значения y_n превосходит заданную величину ε, т.е.

$$|\mathrm{R_n}| = |\mathrm{y(x_n)} - \mathrm{y_n}| > \varepsilon,$$

то производится изменение значений переменных NF и XF:

$$\mathrm{NF = NF + 1, \quad XF = XF + (X - XP).}$$

Если $|y(x_n) - y_n| \leq \varepsilon$, то переменные NF и XF не изменяются. Переменная N увеличивается при каждом входе в подпрограмму обработки результатов. Перед началом интегрирования в головной программе переменным N, NF и XF присваивается нулевое значение.

5). Процедура интегрирования на отрезке с автоматическим выбором шага. Процедура организует интегрирование дифференциального уравнения (1) методом Рунге-Кутта на отрезке $[x_0, x_0 + X]$ и выход в конец интервала интегрирования, т.е. вычисление решения как во внутренних узлах, так и в конце $x_K = x_0 + X$ интервала интегрирования. Содержит обращение к процедуре RKSE выполнения одного шага из точки (x_n, y_n) для вычисления нового значения решения в точке $x_{n+1} = x_n + h$ и оценки локальной погрешности на шаге h. Для задания А необходимо трехкратное решение задачи (1), (2) с разными способами оценки локальной погрешности. Поэтому в этом случае целесообразно иметь в процедуре три обращения к подпрограммам RKSE1, RKSE2, RKSE3 и предусмотреть переключение на вызов нужной из них при каждом интегрировании уравнения. Процедура реализует один из алгоритмов автоматического выбора шага интегрирования: (I) удвоение и деление шага пополам или (II) выбор максимальной для заданной точности длины шага с помощью соотношений (95'), (97').

В задании Б задача (1), (2) решается дважды с различными способами автоматического выбора шага, при этом оценка погрешности вычисляется только одним каким-то способом. Поэтому для задания Б в данной процедуре целесообразно иметь два алгоритма автоматического выбора шага и предусмотреть выбор нужного из них при каждом интегрировании дифференциального уравнения.

После выполнения критерия (91) достижения заданной точности производится обращение к подпрограмме обработки результатов.

Процедура оформляется в виде подпрограммы с первым оператором

$$\mathrm{SUBROUTINE} \quad \mathrm{RKAI(F, X0, Y0, XK, H0, EPS, \ldots)}$$

В результате работы подпрограммы RKAI для задания А будут распечатаны (с помощью подпрограммы обработки результатов) три таблицы для приближенных решений задачи (1), (2), полученных с автоматическим выбором шага интегрирования с использованием перечисленных выше трех способов оценки локальной погрешности. Будут также подсчитаны число NDER вычислений правой части дифференциального уравнения (с помощью подпрограммы F), общее количество выполненных шагов N, число NF шагов h_n^*, на которых не достигается заданная точность (хотя и выполняется практический критерий достижения точности), суммарная длина таких шагов $\mathrm{XF} = \sum\limits_n h_n^*$.

После работы подпрограммы RKAI должны быть определены следующие характеристики вычислительного алгоритма:

- средний шаг $\bar{h} = \dfrac{X}{\mathrm{N}}$,

- относительное число неудачных шагов $\dfrac{\mathrm{NF}}{\mathrm{N}}$,

- относительная суммарная длина неудачных шагов $\dfrac{\mathrm{XF}}{\mathrm{X}} = \dfrac{\sum\limits_n h_n^*}{\mathrm{X}}$.

Эти характеристики, полученные для трех способов оценки локальной погрешности, следует распечатать в следующем виде:

$$\begin{array}{cccccc}
\mathrm{NDER}_1 & \mathrm{N}_1 & \bar{h}_1 & \mathrm{NF}_1 & \dfrac{\mathrm{NF}_1}{\mathrm{N}_1} & \dfrac{\mathrm{XF}_1}{X} \\
\mathrm{NDER}_2 & \mathrm{N}_2 & \bar{h}_2 & \mathrm{NF}_2 & \dfrac{\mathrm{NF}_2}{\mathrm{N}_2} & \dfrac{\mathrm{XF}_2}{X} \\
\mathrm{NDER}_3 & \mathrm{N}_3 & \bar{h}_3 & \mathrm{NF}_3 & \dfrac{\mathrm{NF}_3}{\mathrm{N}_3} & \dfrac{\mathrm{XF}_3}{X}
\end{array}$$

Для задания Б в результате работы подпрограммы RKAI будут распечатаны (с помощью подпрограммы обработки результатов) две таблицы для приближенных решений задачи (1), (2), полученных с использованием указанных выше двух способов автоматического выбора шага интегрирования и одного из трех способов оценки локальной погрешности. Остальные показатели, характеризующие точность и быстродействие вычислительных алгоритмов, как и для задания А, должны быть распечатаны в следующем виде:

$$\begin{array}{cccccc}
\mathrm{NDER}_1 & \mathrm{N}_1 & \bar{h}_1 & \mathrm{NF}_1 & \dfrac{\mathrm{NF}_1}{\mathrm{N}_1} & \dfrac{\mathrm{XF}_1}{X} \\
\mathrm{NDER}_2 & \mathrm{N}_2 & \bar{h}_2 & \mathrm{NF}_2 & \dfrac{\mathrm{NF}_2}{\mathrm{N}_2} & \dfrac{\mathrm{XF}_2}{X}
\end{array}$$

Следует обратить внимание на определение среднего значения шага $\bar{h}$, когда оценка локальной погрешности метода ведется по правилу Рунге (66) и в качестве приближенного решения принимается значение $\bar{\bar{y}}$, полученное исходя из решения в предыдущем узле интегрирования не за один шаг h, а за два половинных шага $\dfrac{h}{2}$: сначала от x_{n-1} до точки $x_{n-1} + \dfrac{h}{2}$, затем от $x_{n-1} + \dfrac{h}{2}$ до точки $x_{n-1} + h$ (см. п. 3.5.2.1). При этом обращение к подпрограмме обработки результатов происходит только после вычисления решения в узле $x_{n-1} + h$ и счетчик числа шагов увеличивается на единицу. Поэтому при определении среднего значения шага интегрирования надо пользоваться формулой $\bar{h} = \dfrac{X}{2\mathrm{N}}$.

6). Процедура интегрирования на отрезке с постоянным шагом. Процедура организует интегрирование дифференциального уравнения (1) методом Рунге–Кутта на отрезке $[x_0, x_0{+}X]$ и выход в конец интервала интегрирования, т.е. вычисление решения как во внутренних узлах, так и в конце $x_K = x_0{+}X$ интервала интегрирования. Интегрирование производится с постоянным шагом, который определяется по результатам просчетов с автоматическим выбором шага. Для задания А постоянный шаг интегрирования полагается равным $h = \bar{h}_1$, для задания Б — $h = \max(\bar{h}_1, \bar{h}_2)$. Содержит обращение к процедуре RKS выполнения одного шага из точки (x_n, y_n) для вычисления нового значения решения в точке $x_{n+1} = x_n + h$ и производит оценку глобальной погрешности метода по правилу Рунге (53). После вычисления приближенного решения в точке x_{n+1} и оценки глобальной погрешности происходит обращение к подпрограмме обработки результатов.

Процедура оформляется в виде подпрограммы с первым оператором

```
SUBROUTINE    RKCI(F, X0, Y0, XK, H)
```

В результате работы подпрограммы RKCI будет распечатана (с помощью подпрограммы обработки результатов) таблица для приближенного решения задачи, полученного с постоянным шагом интегрирования. Как и для автоматического выбора шага будут подсчитаны общее количество N шагов, число NF шагов, на которых не достигается заданная точность, и суммарная длина этих шагов $\mathrm{XF} = \mathrm{NF} \times h$. После окончания работы подпрограммы эти величины должны быть распечатаны в таком же виде, как и при автоматическом выборе шага.

7). Головная программа. Головная программа задает начальные значения задачи Коши, конец интервала интегрирования, начальный шаг интегрирования, требуемую точность приближенного решения и другую необходимую информацию, содержит обращения к подпрограммам интегрирования на отрезке и библиотечной программе.

Такая модульная структура облегчает программирование и исследование различных численных алгоритмов решения дифференциальных уравнений.

7. Содержание отчета

Отчет должен содержать следующую информацию.

1. Наименование задания: задание А или задание Б.

2. Конкретный вариант решаемой задачи Коши (вид дифференциального уравнения и т.д.). Точное аналитическое решение задачи.

3. Для задания А: конкретный вид используемых формул Рунге–Кутта, способ автоматического выбора шага.

Для задания Б: конкретный вид используемой формулы Рунге–Кутта, способ оценки локальной погрешности метода.

4. Для задания А: три таблицы для приближенных решений задачи, полученных с автоматическим выбором шага интегрирования с использованием перечисленных выше трех способов оценки локальной погрешности метода.

Для задания Б: две таблицы для приближенных решений задачи, полученных с использованием указанных выше двух способов автоматического выбора шага интегрирования и одного из трех способов оценки локальной погрешности метода.

5. График точного решения задачи и графики полной (истинной) погрешности каждого приближенного решения. Графики решения и погрешности должны быть построены в разных масштабах, обеспечивавших наглядность характеров поведения кривых.

6. Таблицу для приближенного решения задачи, полученного с постоянным шагом интегрирования.

7. Графики главного члена глобальной погрешности метода, вычисленного по правилу Рунге, и полной (истинной) погрешности для приближенного решения, полученного с постоянным шагом, выполненные в масштабе, обеспечивающем их наглядность (например, приняв за единицу длины величину, равную $\frac{\varepsilon}{2}; \frac{\varepsilon}{4}; \frac{\varepsilon}{10}; \ldots$).

8. Сводную таблицу точностных и скоростных характеристик используемых алгоритмов.

9. Значение решения задачи Коши в конце интервала интегрирования, вычисленное по одной из библиотечных программ, и его абсолютную погрешность.

10. Блок-схему программы.

11. Листинг с программой на алгоритмическом языке.

8. Варианты

8.1. Общая сводка формул Рунге–Кутта

Всюду в дальнейшем формулы Рунге–Кутта (7) будем записывать в виде

$$y_{n+1} = y_n + \Delta y_n, \quad \Delta y_n = \sum_{i=1}^{q} p_{qi} k_i.$$

Принята составная нумерация формул. Первая цифра означает порядок формулы; вторая цифра означает порядковый номер конкретной формулы в группе формул данного порядка. Эти две цифры отделяются друг от друга точкой. Буква K означает, что формула для метода Рунге–Кутта используется вместе с выражением для локальной погрешности специально подобранного метода более низкого порядка.

8.1.1. Формулы второго порядка

2.1 $$\Delta y_n = \frac{1}{2}(k_1 + k_2),$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf(x_n + h,\, y_n + k_1).$$

2.2 $$\Delta y_n = k_2,$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{1}{2}h,\, y_n + \frac{1}{2}k_1\Big).$$

2.3 $$\Delta y_n = \frac{1}{4}(k_1 + 3k_2),$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{2}{3}h,\, y_n + \frac{2}{3}k_1\Big).$$

8.1.2. Формулы третьего порядка

3.1 $$\Delta y_n = \frac{1}{6}(k_1 + 4k_2 + k_3),$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{1}{2}h,\, y_n + \frac{1}{2}k_1\Big), \quad k_3 = hf(x_n + h,\, y_n - k_1 + 2k_2).$$

3.2 $$\Delta y_n = \frac{1}{4}(k_1 + 3k_3),$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{1}{3}h,\, y_n + \frac{1}{3}k_1\Big), \quad k_3 = hf\Big(x_n + \frac{2}{3}h,\, y_n + \frac{2}{3}k_2\Big).$$

3.3 $$\Delta y_n = \frac{1}{9}(2k_1 + 3k_2 + 4k_3),$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{1}{2}h,\, y_n + \frac{1}{2}k_1\Big), \quad k_3 = hf\Big(x_n + \frac{3}{4}h,\, y_n + \frac{3}{4}k_2\Big).$$

8.1.3. Формулы четвертого порядка

4.1 $$\Delta y_n = \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{1}{2}h,\, y_n + \frac{1}{2}k_1\Big), \quad k_3 = hf(x_n + \frac{1}{2}h,\, y_n + \frac{1}{2}k_2),$$

$$k_4 = hf(x_n + h,\, y_n + k_3).$$

4.2 $$\Delta y_n = \frac{1}{6}(k_1 + 4k_3 + k_4),$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{1}{4}h,\, y_n + \frac{1}{4}k_1\Big), \quad k_3 = hf\Big(x_n + \frac{1}{2}h,\, y_n + \frac{1}{2}k_2\Big),$$

$$k_4 = hf\Big(x_n + h,\, y_n + k_1 - 2k_2 + 2k_3\Big).$$

4.3 $$\Delta y_n = \frac{1}{8}\,(k_1 + 3k_2 + 3k_3 + k_4),$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{1}{3}\,h,\, y_n + \frac{1}{3}\,k_1\Big), \quad k_3 = hf\Big(x_n + \frac{2}{3}\,h,\, y_n - \frac{1}{3}\,k_1 + k_2\Big),$$

$$k_4 = hf\Big(x_n + h,\, y_n + k_1 - k_2 + k_3\Big).$$

8.1.4. Формулы пятого порядка

5.1 $$\Delta y_n = \frac{1}{24}\,k_1 + \frac{5}{48}\,k_4 + \frac{27}{56}\,k_5 + \frac{125}{336}\,k_6,$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{1}{2}\,h,\, y_n + \frac{1}{2}\,k_1\Big), \quad k_3 = hf\Big(x_n + \frac{1}{2}\,h,\, y_n + \frac{1}{4}\,(k_1 + k_2)\Big),$$

$$k_4 = hf(x_n + h,\, y_n - k_2 + 2k_3), \quad k_5 = hf\Big(x_n + \frac{2}{3}\,h,\, y_n + \frac{1}{27}\,(7k_1 + 10k_2 + k_4)\Big),$$

$$k_6 = hf\Big(x_n + \frac{1}{5}\,h,\, y_n + \frac{1}{625}\,(28k_1 - 125k_2 + 546k_3 + 54k_4 - 378k_5)\Big).$$

5.2 $$\Delta y_n = \frac{16}{135}\,k_1 + \frac{6656}{12825}\,k_3 + \frac{28561}{56430}\,k_4 - \frac{9}{50}\,k_5 + \frac{2}{55}\,k_6,$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{1}{4}\,h,\, y_n + \frac{1}{4}\,k_1\Big), \quad k_3 = hf\Big(x_n + \frac{3}{8}\,h,\, y_n + \frac{3}{32}\,k_1 + \frac{9}{32}\,k_2\Big),$$

$$k_4 = hf\Big(x_n + \frac{12}{13}\,h,\, y_n + \frac{1932}{2197}\,k_1 - \frac{7200}{2197}\,k_2 + \frac{7296}{2197}\,k_3\Big),$$

$$k_5 = hf\Big(x_n + h,\, y_n + \frac{439}{216}\,k_1 - 8k_2 + \frac{3680}{513}\,k_3 - \frac{845}{4104}\,k_4\Big),$$

$$k_6 = hf\Big(x_n + \frac{1}{2}\,h,\, y_n - \frac{8}{27}\,k_1 + 2k_2 - \frac{3544}{2565}\,k_3 + \frac{1859}{4104}\,k_4 - \frac{11}{40}\,k_5\Big).$$

8.1.5. Формула третьего порядка с контрольным членом

3.1 К. Метод определяется формулой 3.1.

Контрольный член

$$E = \frac{1}{6}\,(k_1 - 2k_2 + k_3)$$

имеет порядок $O(h^3)$ и является асимптотической оценкой погрешности формулы второго порядка 2.2.

8.1.6. Формулы четвертого порядка с контрольным членом

4.1 К. Метод определяется формулой 4.1.

Контрольный член

$$E = \frac{2}{3}\,(k_1 - k_2 - k_3 + k_4)$$

имеет порядок $O(h^3)$ и является асимптотической оценкой погрешности формулы второго порядка (76).

4.2 К. Метод определяется формулой 4.1.
Контрольный член

$$E = \frac{1}{6}\,(k_1 - 4k_2 + 2k_3 + k_4)$$

имеет порядок $O(h^3)$ и является асимптотической оценкой погрешности формулы второго порядка 2.2.

4.3 К. Метод определяется формулой

$$\Delta y_n = \frac{1}{6}\,(k_1 + 4k_4 + k_5),$$

$$k_1 = hf(x_n,\, y_n), \quad k_2 = hf\Big(x_n + \frac{1}{3}\,h,\, y_n + \frac{1}{3}\,k_1\Big), \quad k_3 = hf\Big(x_n + \frac{1}{3}\,h,\, y_n + \frac{1}{6}\,(k_1 + k_2)\Big),$$

$$k_4 = hf\Big(x_n + \frac{1}{2}\,h,\, y_n + \frac{1}{8}\,k_1 + \frac{3}{8}\,k_3\Big), \quad k_5 = hf\Big(x_n + h,\, y_n + \frac{1}{2}\,k_1 - \frac{3}{2}\,k_3 + 2k_4\Big).$$

Контрольный член

$$E = \frac{1}{30}\,(2k_1 - 9k_3 + 8k_4 - k_5)$$

имеет порядок $O(h^4)$ и является асимптотической оценкой погрешности формулы третьего порядка (80). Если интегрируется уравнение (82), то формула (80) имеет пятый порядок и величина $-E$ является оценкой погрешности формулы 4.3 К.

8.1.7. Формулы пятого порядка с контрольным членом

5.1 К. Метод определяется формулой 5.1.
Контрольный член

$$E = \frac{1}{336}\,(-42k_1 - 224k_3 - 21k_4 + 162k_5 + 125k_6)$$

имеет порядок $O(h^5)$ и является асимптотической оценкой погрешности формулы четвертого порядка (88).

5.2 К. Метод определяется формулой 5.2.
Контрольный член

$$E = \frac{1}{360}\,k_1 - \frac{128}{4275}\,k_3 - \frac{2197}{75240}\,k_4 + \frac{1}{50}\,k_5 + \frac{2}{55}\,k_6$$

имеет порядок $O(h^5)$ и является асимптотической оценкой погрешности формулы четвертого порядка (85).

8.2. Варианты для задачи Коши

Рассматривается задача Коши (1), (2). Начальные условия x_0, y_0, правая часть $f(x,y)$ уравнения, конец интервала интегрирования $x_0 + X$, а также начальный шаг интегрирования определяется конкретным вариантом. Необходимо решить задачу (1), (2) в соответствии с требованиями задания А или Б; при этом помимо вычисления решения во внутренних узлах интервала интегрирования определить его значение также в конце интервала $x_K = x_0 + X$.

Перечень вариантов задачи Коши для случая, когда правая часть уравнения имеет вид

$$f(x,y) = g(x)y(x) + \varphi(x)\psi(x), \tag{127}$$

приведен в табл. 2.

Т а б л и ц а 2.

x_0	y_0	x_0+X	h_0	$g(x)$	$\varphi(x)$	$\psi(x)$
1	10	6	0.5	$2(2-x)$	1) $e^{-(x+4)x}$ 2) e^{-x^2} 3) e^{-x^2+2x} 4) $x\,e^{-x^2}$ 5) $(x+2)\,e^{-x^2}$ 6) $e^{-x^2}\sin x$ 7) $e^{-x^2}\cos(0.8x)$ 8) $\cos^2\left(\dfrac{x}{2}\right)\,e^{-x^2+3x}$ 9) $\sin^2(0.6x)\,e^{-x^2+1.5x}$	1) $e^{-4(x+1)}$ 2) 0.01 3) e^x 4) e^{-2x} 5) e^{3x} 6) $\dfrac{1}{\sqrt{2\pi}}$ 7) e^{2x-3} 8) e^{3x-2} 9) $e^{x-1.5}$
-1	8	$2\pi-1$	0.4	$\sin(x+1)$	10) $\sin(x+1)$ 11) $e^{-\cos(x+1)}$ 12) $(x+2)\,e^{-cos(x+1)}$	10) $\dfrac{1}{6}\cos(x+1)$ 11) $\dfrac{1}{8}$ 12) $4\cos(x+1)$

Конкретный вариант для задачи Коши определяется двумя номерами: номером функции $\varphi(x)$ и номером функции $\psi(x)$. Номеру функции $\varphi(x)$ из диапазона с 1 по 9 соответствует любой номер функции $\psi(x)$ в диапазоне с 1 по 9. Номеру функции $\varphi(x)$ из диапазона с 10 по 12 соответствует любой номер функции $\psi(x)$ в диапазоне с 10 по 12. Таким образом, общее количество вариантов для задачи Коши, содержащихся в данной таблице, равно 90.

8.3. Варианты методов для задания А

Задание А заключается, как уже упоминалось, в четырехкратном решении задачи (1), (2) на отрезке $[x_0, x_0+X]$. Первые три раза задача решается с автоматическим выбором шага интегрирования с использованием трех способов оценки локальной погрешности метода. Перечень вариантов методов для данного задания приведен в табл.3.

Указанные в таблице методы используются в следующем сочетании со способами оценки погрешности.

1. Правило Рунге (66). При данном способе оценки погрешности интегрирование надо выполнять по формуле, номер которой указан во втором столбце таблицы.

Т а б л и ц а 3.

Номер варианта	Методы				
	Основная формула	Дополнительная формула для оценки локальной погрешности	Формула с контрольным членом	Способ автоматического выбора шага	Точность интегрирования
1	2	3	4	5	6
1	2.1	3.1	4.1 K и 4.2 K	I или II	
2	2.2	3.2	4.1 K и 4.2 K	I или II	10^{-2}
3	2.3	3.3	4.1 K и 4.2 K	I или II	
4	3.1	4.1	4.3 K	I или II	
5	3.2	4.3	4.3 K	I или II	10^{-3}
6	3.3	4.1	4.3 K	I или II	
7	4.1	5.1	5.1 K	I или II	
8	4.2	5.2	5.2 K	I или II	10^{-4}
9	4.3	5.1	5.1 K	I или II	

2. Комбинация двух формул (68) и (69), одна из которых (формула (69)) указана во втором столбце (т.е. та же, что и в оценке погрешности по правилу Рунге), а номер другой, соответствующей (68), приведен в третьем столбце таблицы.

3. Формула с контрольным членом (73), (73′). Номер этой формулы дан в четвертом столбце. Для методов второго порядка точности задача на третьем этапе решается дважды двумя методами с разными контрольными членами.

Допустимая абсолютная погрешность интегрирования (точность приближенного решения) приведена в шестом столбце таблицы.

Способ автоматического выбора шага выбирается одним из двух: либо (I) — удвоение и деление шага пополам, либо (II) — выбор максимальной для заданной точности длины шага с помощью соотношений (95′), (97′).

Четвертый раз (а для методов второго порядка — пятый раз) задача решается с постоянным шагом, равным среднему арифметическому шагов, выполненных при одном из предыдущих расчетов, а именно, тогда, когда оценка погрешности производилась по правилу Рунге. При интегрировании с постоянным шагом применяется та же формула, что и на первом этапе (т.е. из второго столбца), при этом в каждом узле равномерной сетки производится оценка глобальной погрешности метода по правилу Рунге (63).

Количество вариантов методов для задания А равно 18. Общее число вариантов с учетом дифференциальных уравнений — 1620.

8.4. Варианты методов для задания Б

Задание Б заключается в трехкратном решении задачи Коши (1), (2) на отрезке $[x_0, x_0+X]$. Первые два раза задача решается с контролем точности с применением двух способов автоматического выбора шага интегрирования: (I) — удвоение и деление шага пополам и (II) — выбор максимальной для заданной точности длины шага с помощью соотношений (95′), (97′). Перечень вариантов методов для задания Б приведен в табл. 4.

Т а б л и ц а 4.

Номер варианта	Алгоритм			
	Метод	Способ оценки локальной погрешности	Способ автоматического выбора шага	Точность интегрирования
1	2	3	4	5
1 2 3	2.1 2.2 2.3	Правило Рунге (66)	I и II	10^{-2}
4 5 6	3.1 3.2 3.3	Правило Рунге (66)	I и II	10^{-3}
7 8 9	4.1 4.2 4.3	Правило Рунге (66)	I и II	10^{-4}
10 11 12 13 14	3.1 К 4.1 К и 4.2 К 4.3 К 5.1 К 5.2 К	Контрольный член (73), (73′)	I и II	10^{-3} 10^{-4} 10^{-4} 10^{-5} 10^{-5}

Третий раз (а для варианта 11 – четвертый раз) задача решается с постоянным шагом, равным максимальному среднему арифметическому шагов, сделанных при первом и втором просчетах: $h = \max(\bar{h}_1, \bar{h}_2)$. При этом в каждом узле равномерной сетки производится оценка глобальной погрешности метода по правилу Рунге (53).

Количество вариантов методов для задания Б равно 14.

СПИСОК ЛИТЕРАТУРЫ

1. *Арушанян О.Б., Залеткин С.Ф.* Пакет прикладных программ решения типовых задач для обыкновенных дифференциальных уравнений // Вопросы конструирования библиотек программ. М.: Изд-во Моск. ун-та, 1985.
2. *Арушанян О.Б., Залеткин С.Ф.* Об использовании Библиотеки численного анализа НИВЦ МГУ для решения обыкновенных дифференциальных уравнекий // Вопросы конструирования библиотек программ. М.: Изд-во Моск. ун-та, 1984.
3. *Бабушка И., Витасек Э., Прагер М.* Численные процессы решения дифференциальных уравнений. М.: Мир, 1979.
4. *Бахвалов Н.С., Жидков Н.П., Кобельков Г.М.* Численные методы. М.: Наука, 1987.
5. *Залеткин С.Ф.* О численном решении задачи Коши для обыкновенных линейных однородных дифференциальных уравнений на больших отрезках интегрирования // Вычислительные методы и программирование. **XXVI.** М.: Изд-во Моск. ун-та, 1977.

6. *Залеткин С.Ф.* Коллекция дифференциальных уравнений для тестирования вычислительных алгоритмов и программ // Вопросы конструирования библиотек программ. М.: Изд-во Моск. ун-та, 1983.
7. *Калиткин Н.Н.* Численные методы. М.: Наука, 1978.
8. *Арушанян О.Б., Залеткин С.Ф., Захаров А.Ю., Калиткин Н.Н.* О тестировании программ решения обыкновенных дифференциальных уравнений // Препринт ИПМ АН СССР, № 139. М., 1983.