МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
им. М.В. ЛОМОНОСОВА

Механико – математический факультет
Кафедра вычислительной математики

И. О. Арушанян

Практикум на ЭВМ
Численное решение интегральных уравнений методом квадратур

Третье издание,
дополненное

Москва, 2012

И О. Арушанян

ЧИСЛЕННОЕ РЕШЕНИЕ ИНТЕГРАЛЬНЫХ УРАВНЕНИЙ МЕТОДОМ КВАДРАТУР

ОГЛАВЛЕНИЕ

ЧИСЛЕННОЕ РЕШЕНИЕ ИНТЕГРАЛЬНЫХ УРАВНЕНИЙ МЕТОДОМ КВАДРАТУР

Введение

Данное учебное пособие посвящено рассмотрению метода квадратур для решения линейных интегральных уравнений второго рода, предлагаемого для практической реализации на ЭВМ при выполнении заданий студенческого практикума на механико-математическом факультете Московского университета.

В пособии рассматриваются два типа интегральных уравнений. Это интегральные уравнения Фредгольма второго рода

$$u(x) - \int\limits_a^b K(x,s)u(s)\,ds = f(x), \quad x \in [a,\, b], \tag{$*$}$$

и интегральные уравнения Вольтерра второго рода

$$u(x) - \int\limits_a^x K(x,s)u(s)\,ds = f(x), \quad x \in [a,\, b]. \tag{$**$}$$

Здесь $u(x)$ — искомая функция с областью определения $[a,\, b]$. Заданные функции $K(x,s)$ и $f(x)$ называются соответственно ядром и правой частью интегрального уравнения.

Формально уравнение (**) является частным случаем уравнения (*), если ядро уравнения отвечает условию $K(x,s) \equiv 0$ при $s > x$. Однако это свойство функции K имеет настолько глубокие следствия, что делает теории интегральных уравнений (**) и (*) принципиально различными.

В рамках пособия имеет смысл остановиться на том, какое влияние оказывают эти отличительные особенности на методы численного решения интегральных уравнений.

Из всего многообразия методов численного решения интегральных уравнений вида (*) и (**) в пособии рассматривается только метод квадратур и некоторые его модификации. Входящие в уравнения интегралы заменяются квадратурными суммами, а полученные конечные соотношения могут быть как вспомогательными, так и имеющими самостоятельный характер в качестве окончательных расчетных выражений.

В уравнениях Вольтерра (**) значение искомой функции u в точке x определяется ее значениями в предшествующих точках $s < x$, а значения в последующих точках $s > x$ не влияют на $u(x)$. Для приближенного решения уравнения на отрезке $[a, b]$ берут сетку из точек $a \le s_0 < s_1 < \ldots < s_n \le b$ и находят приближенные значения $u(s_k) \approx u_k^n$ только в узлах сетки. Для этого интегральное уравнение аппроксимируют системой линейных алгебраических уравнений относительно величин u_k^n. Эта система строится так, чтобы по известным значениям $u_0^n, u_1^n, \ldots, u_{k-1}^n$ могло быть построено уравнение для нахождения u_k^n.

В случае уравнения Фредгольма (*) ситуация принципиально иная. Интеграл в (*) зависит от значений неизвестной функции $u(x)$ на всем отрезке. Поэтому в каждое уравнение аппроксимирующей системы должны входить значения u_k^n во всех точках сетки и система должна решаться не последовательно, а “в целом”.

В учебной литературе описание численных методов решения интегральных уравнений состоит, как правило, в построении аппроксимирующей системы линейных уравнений, алгоритме ее решения и (далеко не всегда) оценке погрешности в точках сетки.

В данном пособии в качестве задания практикума для решения интегральных уравнений предложено несколько численных методов, связанных общим принципом — построением последовательности *непрерывных функций*, сходящихся в *интегральной норме* к точному решению.

Для реализации подобных методов существенное значение имеет изучение алгоритмов численного интегрирования с автоматическим выбором распределения узлов, описанию которых посвящен первый раздел пособия. Во втором и третьем разделах рассматриваются методы численного решения интегральных уравнений второго рода. В четвертом разделе излагаются этапы выполнения задания и дан перечень вариантов. Заключает пособие набор тестовых задач для отладки программ студенческого практикума.

1. ЧИСЛЕННОЕ ИНТЕГРИРОВАНИЕ

Пусть требуется вычислить значение определенного интеграла

$$I = \int\limits_a^b f(x)\,dx,$$

где функция $f(x)$ непрерывна на отрезке $[a, b]$. Числа a и b называют пределами интегрирования; они могут быть как конечными, так и бесконечными (тогда интеграл называется несобственным). Как известно, через элементарные функции интегралы выражаются далеко не всегда. По этой и другим причинам для их вычисления применяют численные методы, основанные на замене $f(x)$ такой аппроксимирующей функцией, чтобы интеграл от нее вычислялся в элементарных функциях достаточно просто.

1.1. Квадратурные формулы интерполяционного типа

В качестве аппроксимирующих функций возьмем интерполяционные многочлены Лагранжа $L_n(x)$. Тогда искомый интеграл заменяется линейной комбинацией значений подынтегральной функции $f(x)$:

$$I = \int\limits_a^b f(x)\,dx = \sum_{i=1}^{n} c_i f(x_i) + R = \sum_{i=1}^{n} c_i f_i + R. \qquad (1)$$

Выписанное соотношение называется квадратурной формулой (или просто квадратурой), в которой величины x_i называют узлами, c_i — весами, а R — погрешностью (остаточным членом). Таким образом, интеграл приближенно заменяется суммой, похожей на интегральную сумму, причем узлы и веса не зависят от $f(x)$.

Рассмотрим квадратуру (1) более подробно. Пусть узлы x_i образуют равномерную сетку с шагом h: $x_{i+1} = x_i + h$, $i = 1, \ldots, n$. Если пределы интегрирования a и b входят в состав сетки, то $h = (b - a)/(n - 1)$ и

$$x_1 = a, \quad x_i = x_1 + (i - 1)\,h, \quad x_n = b.$$

В этом случае формула (1) называется формулой замкнутого типа и имеет по крайней мере два узла. Исходный отрезок интегрирования разбивается на $n-1$ подотрезков длины h.

Если a и b не входят в состав сетки, то $h=(b-a)/(n+1)$ и $x_i = a+ih$. Тогда формула (1) называется формулой открытого типа и может иметь один узел. Количество подотрезков равно $n+1$. Если либо a, либо b включены в состав сетки, то соответствующие квадратуры носят название полуоткрытых (или полузамкнутых).

Теперь заменим $f(x)$ на многочлен Лагранжа $L_n(x)$, интерполирующий $f(x)$ в точках (x_i, f_i) на отрезке $[a,b]$:

$$f(x) = L_n(x) = \sum_{i=1}^{n} f_i l_i(x) + \frac{\omega_n(x)}{n!}\, f^{(n)}(\xi), \quad \xi \in [a,b],$$

где $\omega_n(x) = (x-x_1)\dots(x-x_n)$ и

$$\begin{aligned} l_i(x) &= \prod_{\substack{j=1\\ j\neq i}} \frac{x-x_j}{x_i-x_j} = \\ &= \frac{(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)}. \end{aligned}$$

Функции $l_i(x)$ называют фундаментальными многочленами Лагранжа. Подставив выписанное приближенное представление $f(x)$ в (1), получим

$$\begin{aligned} I &= \sum_{i=1}^{n} \left(\int_a^b l_i(x)\,dx \right) f_i + \int_a^b \frac{\omega_n(x)}{n!}\, f^{(n)}(\xi)\,dx = \\ &= (b-a)\sum_{i=1}^{n} c_i f_i + R, \end{aligned} \tag{2}$$

где

$$c_i = \frac{1}{b-a}\int_a^b l_i(x)\,dx, \quad R = \frac{1}{n!}\int_a^b \omega_n(x)\, f^{(n)}(\xi)\,dx.$$

Если сетка $\{x_i\}$ не является равномерной, то формулу (2) называют интерполяционной квадратурной формулой (или квадратурной формулой интерполяционного типа). Если же (как в рассматриваемом случае) сетка равномерна, то формулы типа (2) носят название квадратурных формул Ньютона–Котеса, а веса c_i называют весами Котеса. Легко видеть, что веса, соответствующие узлам, симметричным относительно середины отрезка, равны. Кроме того, $\sum\limits_{i=1}^{n} c_i = 1$, поскольку формула (2) точна для $f(x) \equiv 1$.

Говорят, что квадратура (1) имеет алгебраический порядок точности p, если ее остаточный член R равен нулю для всех алгебраических многочленов степени меньшей или равной p. Поскольку многочлен Лагранжа L_n является алгебраическим многочленом степени $n-1$, то по построению формула (2) имеет алгебраический порядок точности не ниже $n-1$. Однако если n нечетно, т.е. когда середина отрезка $[a, b]$ входит в состав сетки, то формула (2) оказывается точна и для многочленов степени n.

Действительно, при нечетном n один из узлов сетки совпадает с серединой отрезка интегрирования $\bar{x} = (a + b)/2$, а остальные узлы лежат симметрично относительно $\bar{x}$. Рассмотрим многочлен $q(x) = (x-\bar{x})^n$. Этот многочлен является нечетным относительно $\bar{x}$; следовательно

$$\int\limits_a^b q(x) = 0 \qquad \sum_{i=1}^{n} c_i q_i = 0.$$

Отсюда $R = 0$, т.е. формула (2) точна для $q(x)$. Покажем, что эта формула точна и для любого многочлена $p_n(x)$ степени n:

$$p_n(x) = a_1 x^n + a_2 x^{n-1} + \cdots + a_n x + a_{n+1}.$$

Для этого представим $p_n(x)$ в виде

$$p_n(x) = a_1(x - \bar{x})^n + p_{n-1}(x) = a_1 q(x) + p_{n-1}(x)$$

и подставим его в (2). Поскольку формула (2) по построению точна для $p_{n-1}(x)$ и точна для $q(x)$, то она точна и для $p_n(x)$.

Главный член погрешности формулы Ньютона–Котеса с n узлами при условии достаточной гладкости $f(x)$ имеет порядок $O\left(h^{2[(n-1)/2]+3}\right)$, где $[\cdot]$ — целая часть числа. Порядок по h главного члена погрешности называется порядком точности (сходимости) квадратурной формулы.

Для формул замкнутого типа коэффициенты Котеса c_i положительны при $1 \le n \le 8$, а при $n = 9$ и $n \ge 11$ среди них есть отрицательные, что приводит к увеличению ошибок, содержащихся в $f(x)$. Действительно, пусть ошибка задания функции $f(x)$ в каждом узле сетки оценивается сверху по модулю некоторой величиной ε. Погрешность, которая может получиться в сумме $\sum c_i f_i$, оценивается величиной $\varepsilon \sum |c_i|$. Поскольку $\sum c_i = 1$, то наличие отрицательных c_i приводит к увеличению $\sum |c_i|$. Коэффициент увеличения ошибок иллюстрируется следующими цифрами:

$$\sum |c_i| \approx 1.45, \quad n = 9; \qquad \sum |c_i| \approx 3.1, \quad n = 11;$$
$$\sum |c_i| \approx 8.3, \quad n = 15; \qquad \sum |c_i| \approx 560, \quad n = 20.$$

Значения c_i при больших n быстро растут по абсолютной величине, а $\sum |c_i|$ будет большой и быстро возрастающей величиной. Коэффициент увеличения ошибок становится неприемлемым, а соответствующие формулы непригодными для расчетов.

Для формул открытого типа коэффициенты Котеса положительны при $n = 1, 2, 4$, а при остальных n среди них есть отрицательные.

1.2. Элементарные формулы трапеций, средних прямоугольников и Симпсона

Рассмотрим первые три формулы Ньютона–Котеса. Сначала приблизим $f(x)$ на $[a, b]$ многочленом Лагранжа $L_2(x)$ с узлами $x_1 = a$ и $x_2 = b$. Это означает, что вместо кривой $f(x)$ мы взяли многочлен первой степени, проходящий через точки $(a, f(a))$ и $(b, f(b))$. Теперь искомый интеграл, равный площади криволинейной фигуры, заменим на площадь трапеции с основаниями $f(a)$ и $f(b)$ и высотой $h = b - a$:

$$I \approx T = h\,\frac{f(a) + f(b)}{2} = \frac{b-a}{2}\bigl(f(a) + f(b)\bigr). \tag{3}$$

Мы получили квадратурную формулу трапеций, которую можно также получить из (2) при $n = 2$ с сеткой для формул замкнутого типа. Заметим, что алгебраический порядок точности этой формулы равен единице, т.к. середина отрезка $[a, b]$ не входит в состав сетки.

Если на $[a, b]$ взять единственный узел квадратурной формулы, то $f(x)$ аппроксимируется многочленом Лагранжа $L_1(x)$, т.е. многочленом нулевой степени. Возьмем в качестве этого узла середину отрезка $\bar{x} = (a + b)/2$. Тогда получим формулу средних прямоугольников

$$I \approx P = h\,f(\bar{x}). \tag{4}$$

Геометрический смысл этой формулы состоит в том, что площадь криволинейной фигуры заменяется на площадь прямоугольника с основанием $h = b - a$ и высотой $f(\bar{x})$. Ту же самую формулу можно получить из (2) при $n = 1$ с сеткой для формул открытого типа. Алгебраический порядок формулы средних прямоугольников равен единице, поскольку середина отрезка интегрирования входит в состав узлов сетки.

Теперь получим выражения для остаточных членов $R^T = I - T$ и $R^P = I - P$. Для этого разложим $f(x)$ в ряд Тейлора относительно точки $\bar{x} = (a + b)/2$ в предположении о достаточной гладкости функции $f(x)$:

$$\begin{aligned} f(x) \quad &= \quad f(\bar{x}) + (x - \bar{x})\,f'(\bar{x}) + \frac{(x-\bar{x})^2}{2}\,f''(\bar{x}) + \\ &\quad + \frac{(x-\bar{x})^3}{6}\,f'''(\bar{x}) + \frac{(x-\bar{x})^4}{24}\,f^{IV}(\bar{x}) + \cdots . \end{aligned} \tag{5}$$

Воспользовавшись этим разложением, получим представление остаточного члена формулы средних прямоугольников:

$$I = hf(\bar{x}) + \frac{h^3}{24}\,f''(\bar{x}) + \frac{h^5}{1920}\,f^{IV}(\bar{x}) + \cdots = P + R^P, \tag{6}$$

поскольку

$$\int\limits_a^b (x-\bar{x})^j\,dx = \begin{cases} h, & \text{j=0;} \\ 0, & \text{j=1;} \\ \dfrac{h^3}{12}, & \text{j=2;} \\ 0, & \text{j=3;} \\ \dfrac{h^5}{80}, & \text{j=4.} \end{cases}$$

Подставим в (5) $x = a$ и $x = b$ и учтем, что $a - \bar{x} = -h/2$ и $b - \bar{x} = h/2$:

$$f(a) = f(\bar{x}) - \frac{h}{2}\,f'(\bar{x}) + \frac{h^2}{8}\,f''(\bar{x}) - \frac{h^3}{48}\,f'''(\bar{x}) + \frac{h^4}{384}\,f^{IV}(\bar{x}) + \cdots\,,$$

$$f(b) = f(\bar{x}) + \frac{h}{2}\,f'(\bar{x}) + \frac{h^2}{8}\,f''(\bar{x}) + \frac{h^3}{48}\,f'''(\bar{x}) + \frac{h^4}{384}\,f^{IV}(\bar{x}) + \cdots\,.$$

Сложим эти два равенства:

$$\frac{f(a)+f(b)}{2} = f(\bar{x}) + \frac{h^2}{8}\,f''(\bar{x}) + \frac{h^4}{384}\,f^{IV}(\bar{x}) + \cdots\,.$$

Отсюда получим

$$hf(\bar{x}) = h\,\frac{f(a)+f(b)}{2} - \frac{h^3}{8}\,f''(\bar{x}) - \frac{h^5}{384}\,f^{IV}(\bar{x}) + \cdots\,.$$

Подставим это выражение вместо $hf(\bar{x})$ в (6):

$$I = h\,\frac{f(a)+f(b)}{2} - \frac{h^3}{12}\,f''(\bar{x}) - \frac{h^5}{480}\,f^{IV}(\bar{x}) + \cdots = T + R^T. \quad (7)$$

Итак, главные члены погрешности в формулах средних прямоугольников и трапеций равны $\dfrac{h^3}{24}\,f''(\bar{x})$ и $-\dfrac{h^3}{12}\,f''(\bar{x})$ и имеют противоположные знаки. Это означает, что точное значение интеграла лежит в вилке между ними.

Объединяя (6) и (7), можно записать (напомним, что здесь $h = b - a$):

$$I = \ + \frac{h^3}{24}\,f''(\bar{x}) + \frac{h^5}{1920}\,f^{IV}(\bar{x}) + \cdots\,,$$

$$I = T - \frac{h^3}{12}\,f''(\bar{x}) - \frac{h^5}{480}\,f^{IV}(\bar{x}) + \cdots\,.$$

Умножим первое равенство на 2/3, а второе на 1/3 и сложим:

$$I = \frac{2}{3} \; + \frac{1}{3}\,T - \frac{h^5}{2880}\,f^{IV}(\bar{x}) + \cdots = S - \frac{h^5}{2880}\,f^{IV}(\bar{x}) + \cdots\ .$$

Мы видим, что новая формула S, называемая формулой Симпсона (или формулой парабол), имеет вид

$$S = \frac{2}{3} \; + \frac{1}{3}\,T = \frac{h}{6}\left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right), \quad h = b - a. \tag{8}$$

Главный член погрешности этой формулы равен $-\dfrac{h^5}{2880}\,f^{IV}(\bar{x})$.

Отметим важное обстоятельство: из самого вывода этой формулы видно, что ее остаточный член R^S на равномерной сетке имеет разложение по нечетным степеням $h = b - a$, начиная с h^5. Следовательно, формула Симпсона точна для многочленов третьей степени.

Коэффициенты формулы Симпсона и оценку остаточного члена можно вывести из (2) при $n = 3$ для сетки замкнутого типа. Из этого построения следует, что полученная формула должна быть точна для многочленов второй степени, однако она обладает повышенной точностью в силу своей симметрии.

Изменим запись формулы Симпсона (8), рассматривая ее как частный случай формулы Ньютона–Котеса для равномерной сетки из трех узлов с шагом $h = (b - a)/2$:

$$S = \frac{h}{3}\left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right).$$

1.3. Составные формулы трапеций, средних прямоугольников и Симпсона

В общем случае длина отрезка $[a,\, b]$ не мала, а потому остаточный член в рассматриваемых формулах может быть велик. Для повышения точности на отрезке интегрирования вводят достаточно густую сетку $a = x_1 < x_2 < \ldots < x_n = b$, что соответствует разбиению исходного отрезка на $n - 1$ подотрезков, которые иногда называют элементарными. Через $h_i = x_{i+1} - x_i$, $i = 1, \ldots, n-1$, обозначим длину каждого подотрезка. Искомый интеграл I разбивают на сумму интегралов I_i по каждому элементарному подотрезку, которые вычисляются по формулам из п. 1.2.

Таким образом, получают составные, или обобщенные, квадратурные формулы. В нашем случае это:

— составная формула трапеций

$$I = \sum_{i=1}^{n-1} T_i + \sum_{i=1}^{n-1} R_i^T = \frac{1}{2} \sum_{i=1}^{n-1} h_i (f_i + f_{i+1}) -$$

$$- \frac{1}{12} \sum_{i=1}^{n-1} h_i^3 f'' \left(\frac{x_i + x_{i+1}}{2} \right) - \frac{1}{480} \sum_{i=1}^{n-1} h_i^5 f^{IV} \left(\frac{x_i + x_{i+1}}{2} \right) + \cdots ;$$

— составная формула средних прямоугольников

$$I = \sum_{i=1}^{n-1} {}_i + \sum_{i=1}^{n-1} R_i = \sum_{i=1}^{n-1} h_i f \left(\frac{x_i + x_{i+1}}{2} \right) +$$

$$+ \frac{1}{24} \sum_{i=1}^{n-1} h_i^3 f'' \left(\frac{x_i + x_{i+1}}{2} \right) + \frac{1}{1920} \sum_{i=1}^{n-1} h_i^5 f^{IV} \left(\frac{x_i + x_{i+1}}{2} \right) + \cdots ;$$

— составная формула Симпсона

$$I = \sum_{i=1}^{n-1} S_i + \sum_{i=1}^{n-1} R_i^S = \frac{1}{6} \sum_{i=1}^{n-1} h_i \left(f_i + 4 f \left(\frac{x_i + x_{i+1}}{2} \right) + f_{i+1} \right) -$$

$$- \frac{1}{2880} \sum_{i=1}^{n-1} h_i^5 f^{IV} \left(\frac{x_i + x_{i+1}}{2} \right) + \cdots .$$

На равномерной сетке остаточные члены этих квадратурных формул могут быть представлены следующим образом (отбрасываем члены, содержащие более высокие степени h):

$$R^T \approx - \frac{h^2}{12} \sum_{i=1}^{n-1} h f'' \left(\frac{x_i + x_{i+1}}{2} \right) \approx - \frac{h^2}{12} \int_a^b f''(x)\, dx;$$

$$\approx \frac{h^2}{24}\sum_{i=1}^{n-1} h f''\left(\frac{x_i+x_{i+1}}{2}\right) \approx \frac{h^2}{24}\int\limits_a^b f''(x)\,dx;$$
$$R$$

$$R^S \approx -\frac{h^4}{2880}\sum_{i=1}^{n-1} h f^{IV}\left(\frac{x_i+x_{i+1}}{2}\right) \approx -\frac{h^4}{2880}\int\limits_a^b f^{IV}(x)\,dx.$$

Приведенные оценки являются асимптотическими, т.е. выполняются при $h \to 0$ с точностью до членов более высокого порядка малости. Но для справедливости этих оценок необходимо существование непрерывных производных подынтегральной функции соответствующих порядков. Если эти производные кусочно-непрерывны, то можно сделать только мажорантные оценки:

$$|R^T| \le \frac{(b-a)}{12}\,h^2 M_2,\ |R^{|\le} \frac{(b-a)}{24}\,h^2 M_2,\ |R^S| \le \frac{(b-a)}{2880}\,h^4 M_4,$$
$$M_2 = \max_{[a,b]} |f''(x)|, \quad M_4 = \max_{[a,b]} |f^{IV}(x)|.$$

1.4. Построение формул Ньютона–Котеса методом неопределенных коэффициентов

Для повышения точности интегрирования можно увеличивать число точек n, по которым подынтегральная функция аппроксимируется интерполяционным многочленом Лагранжа. В этом случае коэффициенты c_i и остаточный член R определяются из (2).

Более наглядным способом мы построили формулы замкнутого типа для $n = 2$ и $n = 3$ и формулу открытого типа для $n = 1$.

Рассмотрим другой способ построения квадратурных формул, называемый методом неопределенных коэффициентов. Будем рассматривать только формулы замкнутого типа.

Пусть $n = 4$. Формула не будет симметричной, т.к. середина отрезка не входит в состав сетки. Построим формулу вида

$$I = \int\limits_a^b f(x)\,dx \approx c_1 f(x_1) + c_2 f(x_2) + c_3 f(x_3) + c_4 f(x_4),$$

где

$$x_1 = a,\ \ x_2 = a + \frac{b-a}{3},\ \ x_3 = a + 2\,\frac{b-a}{3},\ \ x_4 = b.$$

Коэффициенты c_i будем подбирать так, чтобы эта формула была точна для многочленов как можно более высокой степени. Для упрощения выкладок проведем стандартную замену переменных

$$x = \frac{b+a}{2} + \frac{b-a}{2}\,t, \quad -1 \le t \le 1, \quad a \le x \le b,$$

в результате которой получим

$$\int\limits_a^b f(x)\,dx = \frac{b-a}{2}\int\limits_{-1}^{1} f\big(x(t)\big)\,dt = \frac{b-a}{2}\int\limits_{-1}^{1} g(t)\,dt.$$

Теперь, с учетом проведенной замены, исходная задача свелась к поиску таких коэффициентов d_i, чтобы квадратурная формула

$$\int\limits_{-1}^{1} g(t)\,dt \approx d_1 g(-1) + d_2 g(-1/3) + d_3 g(1/3) + d_4 g(1)$$

была точна для многочленов наиболее высокой степени.

Погрешность квадратуры имеет вид

$$R(g) = \int\limits_{-1}^{1} g(t)\,dt - \big(d_1 g(-1) + d_2 g(-1/3) + d_3 g(1/3) + d_4 g(1)\big).$$

Подставим в это выражение многочлен $g(t) = \sum\limits_{j=0}^{m} a_j t^j$ степени m:

$$R(g) = \sum_{j=0}^{n} a_j R\big(t^j\big).$$

Подбором d_i мы попытаемся добиться выполнения равенств

$$R(1) = 0,\ R(t) = 0,\ \ldots\ ,\ R(t^m) = 0$$

при возможно большем значении m. Подставим в эти равенства последовательно $g(t) \equiv 1$, $g(t) \equiv t$, $g(t) \equiv t^2$, $g(t) \equiv t^3$. Получим систему из четырех линейных уравнений, решение которой дает следующие значения коэффициентов квадратурной формулы:

$$d_1 = d_4 = \frac{1}{4}, \quad d_2 = d_3 = \frac{3}{4}.$$

Проверкой убеждаемся, что построенная квадратурная формула не будет точна для $g(t) \equiv t^4$. Учитывая, что $c_i = d_i(b-a)/2$ и $b - a = 3h$, получим искомую квадратурную формулу с введенными выше узлами x_i:

$$\int_a^b f(x)\,dx \approx \frac{3}{8}h\,(f_1 + 3f_2 + 3f_3 + f_4). \tag{9}$$

Эту квадратуру называют правилом (или формулой) трех восьмых. Поскольку она точна для многочленов третьей степени, то ее остаточный член имеет порядок $O(h^5)$ (т.е. тот же, что и остаточный член формулы Симпсона), однако раскладывается в ряд Тейлора по последовательным степеням h. Ту же формулу можно получить из (2) при $n = 4$. Погрешность формулы (9) представляется в виде $-3h^5 f^{IV}(\xi)/80$, $\xi \in [a, b]$.

Составная формула трех восьмых может быть построена разбиением $[a,\, b]$ на элементарные подотрезки с последующим применением (9) на каждом из них. Мы же построим ее в несколько другом виде.

Возьмем число элементарных подотрезков кратным трем, т.е. равным $3m$, $m = 1, 2, \ldots$. Тогда число узлов полученной сетки $n = 3m + 1$, а ее шаг $h = (b-a)/(n-1) = (b-a)/3m$. Возьмем строенный отрезок $[a+kh,\, a+(k+3)h]$, $k = 0, 3, 6, \ldots$, и применим на нем правило (9):

$$\int_{a+kh}^{a+(k+3)h} f(x)\,dx = \frac{3}{8}h\,(f_1 + 3f_2 + 3f_3 + f_4) + R_{k+1},$$

где

$$R_{k+1} = -\frac{3h^5}{80} f^{IV}(\xi_{k+1}), \quad \xi_{k+1} \in [a + kh,\, a + (k+3)h].$$

Если эти равенства расписать для всех остальных строенных подотрезков и сложить их почленно, то получим составную формулу трех восьмых:

$$\int_a^b f(x)\,dx \;=\; \frac{3h}{8}\big[(f_1+f_n)+2(f_4+f_7+\cdots+f_{n-3})+$$
$$+\,3(f_2+f_3+f_5+f_6+\cdots+f_{n-2}+f_{n-1})\big]+R,$$

где погрешность R ведет себя следующим образом:

$$R=-\frac{h^4}{80}\big(3h\big[f^{IV}(\xi_1)+\cdots+f^{IV}(\xi_{m/3})\big]\big)\approx-\frac{h^4}{80}\int_a^b f^{IV}(x)\,dx.$$

Мажорантная оценка имеет вид

$$|R|\le\frac{(b-a)}{80}\,h^4M_4,\quad M_4=\max_{[a,b]}\left|f^{IV}(x)\right|.$$

Хотя формула Симпсона по точности и количеству вычислений значений подынтегральной функции предпочтительнее, правило трех восьмых имеет самостоятельное значение, т.к. оно может быть использовано в случае таблично заданной функции и нечетного числа элементарных отрезков, когда формула Симпсона неприменима.

Возьмем теперь $n=5$, $h=(b-a)/4$. В этом случае квадратурная формула будет симметрична, т.к. середина отрезка входит в состав сетки; следовательно, она будет точна не только для многочленов четвертой степени, но и пятой. Применяя описанный выше метод неопределенных коэффициентов, получим следующую формулу (ее иногда называют формулой Боде):

$$\int_a^b f(x)\,dx\approx\frac{(b-a)}{2}\left(\frac{7}{45}f_1+\frac{32}{45}f_2+\frac{12}{45}f_3+\frac{32}{45}f_4+\frac{7}{45}f_5\right)=$$
$$=h\left(\frac{14}{45}f_1+\frac{64}{45}f_2+\frac{24}{45}f_3+\frac{64}{45}f_4+\frac{14}{45}f_5\right).$$

Главный член ее погрешности равен

$$-\frac{8}{945}\,h^7 f^{VI}(\xi), \quad \xi \in [a,b],$$

а остаточный член разлагается по нечетным степеням h, начиная с h^7.

Может быть построена составная формула Боде с главным членом погрешности $O(h^6)$.

1.5. Квадратурные формулы Гаусса

При построении квадратурных формул Гаусса важную роль играют ортогональные многочлены Лежандра, имеющие вид:

$$P_n(x) = \frac{1}{2^n n!}\,\frac{d^n}{dx^n}\big[(x^2-1)^n\big], \quad n = 0, 1, 2, \ldots\,.$$

Эти многочлены обладают следующими свойствами:

1) $P_n(1) = 1, \quad P_n(-1) = (-1)^n, \quad n = 0, 1, 2, \ldots;$

2) $\int\limits_{-1}^{1} P_n(x)\,Q_k(x)\,dx = 0, \quad k < n, \quad Q_k(x)$ — любой многочлен степени $k < n$;

3) многочлен Лежандра $P_n(x)$ имеет n различных вещественных корней, расположенных симметрично относительно нуля на интервале $(-1,\,1)$.

Выпишем вид многочленов $P_n(x)$ для $n = 0, 1, 2, 3$:

$$\begin{aligned} P_0(x) &= 1, & P_1(x) &= x, \\ P_2(x) &= \frac{1}{2}\big(3x^2 - 1\big), & P_3(x) &= \frac{1}{2}\big(5x^3 - 3x\big). \end{aligned}$$

Теперь рассмотрим интеграл на стандартном отрезке $[-1, 1]$:

$$I = \int\limits_{-1}^{1} f(t)\,dt.$$

Поставим задачу определения узлов $t_1, t_2, \ldots, t_n$ на $[-1,\,1]$ и коэффициентов $c_1, c_2, \ldots, c_n$, чтобы квадратурная формула

$$\int\limits_{-1}^{1} f(t)\,dt \approx \sum_{i=1}^{n} c_i f(t_i)$$

была точной для многочленов наиболее высокой степени.

Формулы, обладающие таким свойством, называются квадратурными формулами (квадратурами) Гаусса. Другое название — квадратурные формулы наивысшей алгебраической степени точности.

Поскольку в квадратуре Гаусса мы имеем $2n$ свободных параметров t_i и c_i, $i = 1, 2, \ldots, n$, а многочлен степени $2n-1$ определяется $2n$ коэффициентами, то можно подобрать эти параметры так, чтобы наивысшая степень многочлена в общем случае равнялась $N = 2n - 1$.

Для того чтобы квадратурная формула была точна для многочленов степени до $2n-1$ включительно, необходимо и достаточно, чтобы она была точна для одночленов

$$f(t) \equiv 1,\, t,\, t^2,\, \ldots\,,\, t^{2n-1}.$$

Будем подбирать параметры t_i и c_i методом неопределенных коэффициентов, который мы использовали ранее для построения формул Ньютона–Котеса. Получим систему $2n$ уравнений с $2n$ неизвестными:

$$\begin{aligned}
\sum_{i=1}^{n} c_i &= 2,\\
\sum_{i=1}^{n} c_i t_i &= 0,\\
\ldots&\\
\sum_{i=1}^{n} c_i t_i^{2n-2} &= \frac{2}{n-1},\\
\sum_{i=1}^{n} c_i t_i^{2n-1} &= 0.
\end{aligned} \tag{10}$$

Эта система нелинейная, поэтому ее решение затруднено уже при небольших значениях n. Кроме того, нет никакой гарантии, что она вообще разрешима, или что ее решения t_i вещественны и принадлежат отрезку $[-1, 1]$.

Докажем разрешимость системы (10). Рассмотрим семейство многочленов вида

$$f_k(t) = t^k P_n(t), \quad k = 0, 1, \ldots, n-1,$$

где $P_n(t)$ — многочлен Лежандра. Так как степени многочленов $f_k(t)$ не превосходят $2n-1$, то, если t_i и c_i удовлетворяют системе (10), должны быть выполнены равенства

$$\int_{-1}^{1} t^k P_n(t)\, dt = \sum_{i=1}^{n} c_i t_i^k P_n(t_i), \quad k = 0, 1, \ldots, n-1.$$

В силу свойства ортогональности многочленов Лежандра справедливы равенства

$$\int_{-1}^{1} t^k P_n(t)\, dt = 0 \qquad k < n.$$

Следовательно,

$$\sum_{i=1}^{n} c_i t_i^k P_n(t_i) = 0, \quad k = 0, 1, \ldots, n-1.$$

Эти равенства заведомо будут выполняться при любых c_i, если положить

$$P_n(t_i) = 0, \quad i = 1, 2, \ldots, n,$$

т.е. если в качестве узлов t_i взять корни многочлена Лежандра $P_n(t)$. Напомним, что эти корни вещественны, различны и симметрично относительно нуля расположены на $(-1,\, 1)$.

Если мы подставим t_i в (10), то из первых n линейных уравнений найдем коэффициенты c_i. Они определяются при этом однозначно, поскольку определителем соответствующей линейной системы является определитель Вандермонда.

Осталось показать, что построенная квадратурная формула точна для всех многочленов степени до $2n-1$ включительно. Действительно, представим произвольный многочлен $Q_{2n-1}(t)$ степени $2n-1$ в виде

$$Q_{2n-1}(t) = G_{n-1}(t)P_n(t) + F_{n-1}(t),$$

где G_{n-1} и F_{n-1} — многочлены степени $n-1$. Предложенная квадратурная формула по своему построению точна для каждого из слагаемых в этом разложении, следовательно, точна и для произвольного многочлена степени не выше $2n-1$.

В случае применения квадратуры Гаусса для вычисления интеграла на произвольном отрезке $[a,\, b]$ делаем замену переменных

$$x = \frac{b+a}{2} + \frac{b-a}{2}\, t,$$

в результате которой получим

$$\int\limits_a^b f(x)\, dx = \frac{b-a}{2} \int\limits_{-1}^{1} f\left(\frac{b+a}{2} + \frac{b-a}{2}\, t\right) dt.$$

Последний интеграл заменим квадратурной формулой Гаусса:

$$\int\limits_a^b f(x)\, dx \approx \frac{b-a}{2} \sum_{i=1}^{n} c_i f(x_i),$$

где $x_i = \dfrac{b+a}{2} + \dfrac{b-a}{2} t_i$, $i = 1, \ldots, n$, а t_i — корни многочлена Лежандра $P_n(t)$ на $[-1,\, 1]$.

Остаточный член формулы Гаусса с n узлами имеет вид

$$R_n = \frac{(b-a)^{2n+1}(n!)^4 f^{(2n)}(\xi)}{[(2n)!]^3(2n+1)} \approx$$

(с учетом формулы Стирлинга)

$$\approx \frac{b-a}{2.5\sqrt{n}} \left(\frac{b-a}{3n}\right)^{2n} f^{(2n)}(\xi), \quad \xi \in [a, b].$$

В частности,

$$R_2 \approx \frac{1}{135}\left(\frac{b-a}{2}\right)^5 f^{(4)}(\xi),$$

$$R_3 \approx \frac{1}{15750}\left(\frac{b-a}{2}\right)^7 f^{(6)}(\xi).$$

Кроме того, имеем $t_i = -t_{n-i+1}$ (в силу свойств корней многочлена Лежандра) и $c_i = c_{n-i+1}$ (в силу равенства коэффициентов при симметричных узлах).

Могут быть построены составные формулы Гаусса, если отрезок интегрирования разбить на подотрезки и на каждом из них применить формулу Гаусса. Однако в отличие от составных формул Ньютона–Котеса замкнутого типа, в которых можно экономить одно обращение к подынтегральной функции на каждой граничной точке подотрезка (кроме концов всего отрезка интегрирования), в составных формулах Гаусса такой возможности нет.

1.6. Правило Рунге практической оценки погрешности квадратурных формул

Из п. 1.1. следует, что чем выше степень многочлена Лагранжа $L_n(x)$, аппроксимирующего заданную подынтегральную функцию, тем выше точность соответствующей квадратуры. Это верно для достаточно малых длин отрезков интегрирования. Кроме того, не следует забывать, что для аппроксимации, как правило, используют $L_n(x)$ для n, не бóльших 4 или 5. Поэтому применяется аппроксимация не сразу на всем отрезке, а кусочно-полиномиальная аппроксимация многочленами невысокой степени, что приводит к построению составных квадратур. При этом возможны два способа:

— либо берут достаточно густую равномерную сетку и на ней строят составную формулу;

— либо разбивают отрезок на подотрезки и затем на каждом из них применяют элементарную квадратуру с одновременным суммированием полученных приближений интеграла на этих подотрезках.

Шаг сетки и длины подотрезков следует подбирать из того требования, чтобы значение интеграла вычислялось с некоторой заранее предписанной точностью. Для оценки достигнутой при вычислении интеграла точности мы имеем пока только асимптотические или мажорантные оценки остаточных членов квадратурных формул, которые либо трудно использовать на практике, либо дают излишне малые значения для шага интегрирования и длины подотрезков.

Поэтому применяется другой практически эффективный и легко реализуемый способ контроля точности интегрирования, основанный на оценке главного члена погрешности квадратуры при помощи правила Рунге. Этот метод называют еще методом двойного пересчета или экстраполяцией по Ричардсону.

Пусть на $[a,\, b]$ взята равномерная сетка $\{x_i\}$ с шагом h. Вычислим на этой сетке приближенное значение I_h интеграла I по какой-либо составной квадратурной формуле, имеющей алгебраический порядок точности $p-1$. Это означает, что имеет место равенство

$$I - I_h = ch^p + O(h^{p+1}), \tag{11}$$

где в правой части стоит разложение остаточного члена по степеням h.

Теперь построим сетку с шагом $h/2$ и вычислим $I_{h/2}$ по той же самой квадратурной формуле. Тогда имеем

$$I - I_{h/2} = \tilde{c}\left(\frac{h}{2}\right)^p + O\left(\left(\frac{h}{2}\right)^{p+1}\right). \tag{12}$$

Будем считать, что $c \approx \tilde{c}$ в силу достаточной малости h. Из этих двух равенств мы можем выразить главный член погрешности ch^p через I_h и $I_{h/2}$ с точностью до $O\left(h^{p+1}\right)$:

$$I_{h/2} - I_h = ch^p - c\frac{h^p}{2^p} + O\left(h^{p+1}\right),$$

откуда

$$ch^p = \frac{I_{h/2} - I_h}{1 - 2^{-p}} + O\left(h^{p+1}\right). \tag{13}$$

Равенство (13) называют первой формулой Рунге.

Таким образом, можно сформулировать простейший алгоритм вычисления интеграла с заданной точностью ε по выбранной квадратурной формуле: если $\delta = |ch^p| \le \varepsilon$, то интеграл вычислен с заданной точностью, если $\delta > \varepsilon$, то шаг h еще раз делится пополам и процедура повторяется.

Подставим теперь (13) в (11). В результате получим новую квадратурную формулу $I_{h,h/2}$ с главным членом погрешности порядка h^{p+1}, а не h^p, т.е. вновь полученная формула будет иметь порядок точности на единицу больше, чем первоначальная формула I_h:

$$I = \underbrace{I_h + \frac{I_{h/2} - I_h}{1 - 2^{-p}}}_{I_{h,h/2}} + c_1 h^{p+1} + O\left(h^{p+2}\right).$$

Выписанное представление интеграла носит название второй формулы Рунге.

Таким образом, мы описали еще один способ построения квадратурных формул, который может быть использован наряду с методом неопределенных коэффициентов и аппарата интерполирования.

Поскольку можно ожидать, что приближение $I_{h/2}$ более точно, чем I_h, то разумно в качестве нового приближения брать

$$I = \underbrace{I_{h/2} + \frac{I_{h/2} - I_h}{2^p - 1}}_{I_{h,h/2}} + c_2 h^{p+1} + O\left(h^{p+2}\right), \tag{14}$$

которое получается, если в (12) подставить оценку

$$\tilde{c}\left(\frac{h}{2}\right)^p \approx \frac{I_{h/2} - I_h}{2^p - 1}$$

Повторив процесс с вдвое меньшим шагом, получим формулу с главным членом погрешности порядка h^{p+2} и т.д.

Таким образом, мы получили не только способ контроля точности при вычислении интеграла, дающий возможность выбора надлежащей величины шага h, но и простой метод построения все более точных квадратурных формул (без явного выписывания их в виде квадратурной суммы с коэффициентами

Котеса). В качестве исходной формулы можно взять простую формулу невысокой точности (например, трапеций или средних прямоугольников).

Покажем, что операция построения формулы $I_{h,h/2}$ (носящей имя Ричардсона) есть операция экстраполирования. Для этого надо показать, что, если $I_h \neq I_{h/2}$, то $I_{h,h/2}$ всегда лежит вне отрезка с границами I_h и $I_{h/2}$.

Действительно, если $I_{h/2} > I_h$, то

$$I_{h,h/2} = I_{h/2} + \frac{I_{h/2} - I_h}{2^p - 1} > I_{h/2},$$

т.к. $p \geq 1$. Если $I_{h/2} < I_h$, то

$$I_{h,h/2} = I_{h/2} - \frac{I_h - I_{h/2}}{2^p - 1} < I_{h/2}.$$

Если же $I_{h/2} = I_h$, то $I_{h,h/2} = I_{h/2} = I_h$ и увеличения точности не будет.

Наиболее эффективна экстраполяция по Ричардсону (или, иными словами, применение второго правила Рунге), если в качестве первоначальной формулы I_h взять симметричную формулу, у которой остаточный член раскладывается по четным степеням h. Тогда каждое применение второго правила Рунге позволяет получать формулу, порядок точности которой увеличивается на два, а не на единицу, как это имеет место в случае несимметричных формул. Наиболее употребительным является случай, когда в качестве первоначальной формулы выбирается формула трапеций T, для которой, как было показано выше, имеет место следующее разложение остаточного члена (если подынтегральная функция имеет $2s+2$ непрерывные производные на $[a, b]$):

$$I = T_h + c_1 h^2 + c_2 h^4 + c_3 h^6 + \cdots + c_s h^{2s} + O\left(h^{2s+2}\right).$$

Для целей, которые будут видны из последующего изложения, переобозначим $T_h \equiv T_{h,0}$ и $T_{h/2} \equiv T_{h/2,0}$. Тогда по второму правилу Рунге исключим первый член $c_1(h/2)^2$ разложения в ряд Тейлора остаточного члена у $T_{h/2,0}$ и получим новую формулу

$$T_{h/2,1} = T_{h/2,0} + \frac{T_{h/2,0} - T_{h,0}}{2^2 - 1},$$

у которой остаточный член раскладывается также по четным степеням, но начиная с h^4:

$$I = T_{h/2,1} + b_2 h^4 + b_3 h^6 + \cdots + b_s h^{2s} + O\left(h^{2s+2}\right).$$

Новая формула $T_{h/2,1}$ имеет порядок точности, превышающий на два порядок точности формулы трапеций, т.е. она точна для многочленов третьей степени.

Покажем, что новая формула $T_{h/2,1}$ есть формула Симпсона, записанная в неявном виде, т.е. без непосредственного выписывания коэффициентов Котеса. Для простоты вывода будем полагать, что $h = b - a$ и формула трапеций строится по двум узлам. Тогда

$$\begin{aligned}
T_{h,0} \quad &= \frac{h}{2}(f_1 + f_2), \quad T_{h/2,0} = \frac{h}{4}(f_1 + 2f_{1/2} + f_2),\\
T_{h/2,1} &= \frac{h}{4}(f_1 + 2f_{1/2} + f_2) +\\
&\quad + \frac{1}{3}\left(\frac{h}{4}(f_1 + 2f_{1/2} + f_2) - \frac{h}{2}(f_1 + f_2)\right) =\\
&= \frac{h}{6}(f_1 + 4f_{1/2} + f_2) = S.
\end{aligned}$$

Получили формулу Симпсона, выписанную по сетке с тремя узлами.

Теперь мы можем применить правило (14) уже к формуле Симпсона $T_{h/2,1}$, использовав уже вычисленное $T_{h/4,1}$:

$$T_{h/4,2} = T_{h/4,1} + \frac{T_{h/4,1} - T_{h/2,1}}{2^4 - 1}.$$

Здесь в знаменателе двойка возводится в четвертую степень, т.к. при вычислении $T_{h/4,2}$ мы исключили член $b_2 h^4$ в разложении по степеням h остаточного члена формулы $T_{h/2,1}$. В результате для новой формулы $T_{h/4,2}$ остаточный член разлагается по четным степеням h, начиная с h^6:

$$I = T_{h/4,2} + a_3 h^6 + \cdots + a_s h^{2s} + O\left(h^{2s+2}\right).$$

Следовательно, алгебраический порядок точности формулы $T_{h/4,2}$ превышает порядок точности формулы Симпсона на два и равен пяти (т.е. она точна для многочленов пятой степени). Можно показать, что она совпадает с ранее выписанной формулой Боде, которая может быть получена методом неопределенных коэффициентов на сетке с пятью узлами.

Если мы применим правило Рунге к формуле Боде $T_{h/4,2}$, то получим формулу по девяти узлам, имеющую седьмой порядок точности; в разложении остаточного члена этой формулы присутствуют четные степени h, начиная с восьмой.

С каждым следующим применением правила Рунге к вновь полученным формулам мы будем получать формулы с порядком точности 9, 11 и т.д.

Отметим, что, начиная с формулы, построенной по (14) на основе формулы Боде, порядок точности будет меньше, чем если бы эти формулы строились в качестве формул Ньютона–Котеса, т.е. как формулы интерполяционного типа. Например, для $n = 9$ формула Ньютона–Котеса имеет девятый порядок точности, а не седьмой, как формула, полученная из формулы Боде по (14). Поэтому начиная с этого момента по второму правилу Рунге нельзя построить формулы Ньютона–Котеса — это будут уже другие формулы, не являющиеся точными для многочленов наиболее высокой степени.

1.7. Процесс Эйткена оценки фактической точности квадратурной формулы

У всех рассмотренных квадратурных формул остаточный член можно разложить в ряд по степеням шага сетки. Следовательно, к ним применим метод Рунге (при этом требуется, чтобы был заранее известен порядок точности исходной формулы). Кроме того, разложение остаточного члена проводится при условии достаточной гладкости подынтегральной функции. Если же функция не обладает требуемой гладкостью, то это разложение уже не имеет смысла, т.е. нам уже не известен фактический (реальный) порядок точности формулы.

Рассмотрим процесс Эйткена, которой на основе повторных просчетов на нескольких сетках дает возможность:

— определить фактический порядок точности p квадратурной формулы для заданной подынтегральной функции;

— уточнить результат, полученный на исходной сетке.

В упрощенном виде процесс Эйткена можно описать следующим образом.

Выберем три сетки с шагами $h_1 = h$, $h_2 = h/2$, $h_3 = h/4$. Вычислим приближения I_h, $I_{h/2}$ и $I_{h/4}$ к интегралу I по выбранной квадратурной формуле.

Тогда, если учитывать только главный член погрешности, имеем три уравнения для определения I, и p, где p — фактический, заранее неизвестный порядок точности формулы для данной подынтегральной функции:

$$I = I_h + ch^p,$$
$$I = I_{h/2} + \frac{1}{2^p}\, ch^p,$$
$$I = I_{h/4} + \frac{1}{4^p}\, ch^p,$$

в которой значения I, c и p неизвестны. Из первого и второго уравнений имеем

$$ch^p \left(1 - \frac{1}{2^p}\right) = I_{h/2} - I_h.$$

Из второго и третьего уравнений получим

$$\frac{1}{2^p}\, ch^p \left(1 - \frac{1}{2^p}\right) = I_{h/4} - I_{h/2}.$$

Из последних двух равенств получаем уравнение для определения p:

$$2^p = \frac{I_{h/2} - I_h}{I_{h/4} - I_{h/2}}\,.$$

Оценка для главного члена погрешности имеет вид

$$ch^p = \frac{\left(I_{h/2} - I_h\right)^2}{2I_{h/2} - I_h - I_{h/4}}\,.$$

Полученную оценку можно использовать для уточнения I_h.

Пример 1. Пусть вычисляется следующий интеграл методом трапеций:

$$\int\limits_0^1 \sqrt{x}\,dx.$$

Тогда для этого интеграла фактический порядок точности p будет равен $1/2$.

Пример 2. Пусть вычисляется следующий интеграл методом трапеций:

$$\int\limits_0^1 \sqrt[n]{x}\,dx.$$

Тогда для этого интеграла фактический порядок точности p будет равен $(n+1)/n$.

1.8. Способы построения практических алгоритмов численного интегрирования. Адаптивные алгоритмы

Ранее мы рассмотрели различные квадратурные формулы и способ Рунге контроля достигнутой при их применении точности, однако оставили в стороне вопрос построения практически пригодных алгоритмов.

Всегда можно подобрать такую подынтегральную функцию $f(x)$, для которой любая заранее выбранная программа численного интегрирования дает совершенно неверные результаты. Сказанное иллюстрирует следующий простейший пример. Возьмем $f(x) \equiv 1$ и вычислим интеграл по данной программе. Запомним выбранные этой программой узлы интегрирования x_i и доопределим функцию $f(x)$ следующим образом: $f(x_i) = 1$, $f(x) = A \neq 1$, $x \neq x_i$. Очевидно, что при повторном запуске для видоизмененной функции программа даст прежний результат, хотя верный результат может значительно отличаться от вычисленного при надлежащем задании A.

Таким образом, при построении практических алгоритмов класс подобных примеров должен быть максимально сужен, насколько это возможно без неоправданного усложнения логики или потери эффективности на распространенных типах задач.

Рассмотрим несколько алгоритмов такого рода, в которых для контроля достижения предписанной точности применяется автоматический выбор шага интегрирования. При этом будем учитывать, что бóльшая часть машинного времени в реальных расчетах тратится на вычисление подынтегральной функции. Поэтому программа считается тем эффективней, чем меньше таких вычислений она производит для достижения заданной точности.

1.8.1. Простейший (неадаптивный) алгоритм

Зададим на отрезке интегрирования n узлов и вычислим по выбранной квадратурной формуле приближенное значение I_n интеграла I по этим узлам. Затем увеличим число узлов вдвое и вычислим I_{2n}.

По первому правилу Рунге вычислим оценку погрешности ошибки, которая, конечно, не учитывает погрешность суммирования R_Σ, а дает лишь оценку главного члена погрешности самой формулы, т.е. метода интегрирования:

$$\delta = \frac{I_{2n} - I_n}{2^p - 1},$$

где p — порядок точности используемой формулы.

Если $|\delta| < \varepsilon$, где ε — предписанная точность, то полагаем $I \approx I_{2n} + \delta$ и заканчиваем вычисления, считая, что требуемая точность достигнута. Заметим, что добавление δ лишь немного подправляет величину I_{2n}, т.к. δ мало по модулю; тем самым нет существенного повышения точности, как это происходит при применении второго правила Рунге.

Если $\delta > \varepsilon$, то мы опять удваиваем текущее значение n и повторяем описанный процесс.

Данный алгоритм не учитывает локальный характер поведения подынтегральной функции $f(x)$. Общее количество узлов равномерной сетки подбирается с учетом наихудшего поведения $f(x)$ на отдельных участках (которые могут иметь небольшую суммарную длину), при этом не учитываются участки, где $f(x)$ меняется медленно. Такие алгоритмы называют неадаптивными.

Рассмотренный алгоритм удобен при применении формул Ньютона–Котеса, для которых нетрудно реализовать составные формулы. Для формул Гаусса этот алгоритм неудобен, т.к. надо хранить в программе большое число узлов и коэффициентов из-за того, что заранее неясно, какое n окажется пригодным.

1.8.2. Модификация простейшего алгоритма

Отрезок $[a, b]$ разбивается на n подотрезков (частей), а затем на каждом подотрезке применяется выбранная квадратурная формула. В процессе вычислений приближенные значения интегралов по подотрезкам суммируются, что дает в результате приближение I_n по всему отрезку. Затем n удваивается, вычисляется I_{2n}, после чего поступают как в алгоритме 1.8.1. Данная модификация обладает теми же недостатками, но более удобна для применения формул Гаусса, т.к. требует хранения узлов и коэффициентов только одной выбранной формулы.

Контроль точности разумно вести по критерию:

$$|\delta| < \max(\varepsilon_A,\, \varepsilon_O \cdot |I_{2n}|),$$

где ε_A и ε_O — предписанные абсолютная и относительная погрешности интегрирования. Такой подход позволяет избегать неоправданного увеличения n там, где значение I мало или велико.

Примечание к алгоритмам 1.8.1 *и* 1.8.2. Для экономии трудозатрат значения $f(x)$, ранее вычисленные в n узлах (или на n подотрезках), в случае применения формул Ньютона–Котеса замкнутого типа могут быть использованы при вычислениях с $2n$ узлами (или на $2n$ подотрезках). При применении формул Гаусса такой возможности нет, поскольку в обоих случаях при повторном счете узлы будут совершенно другими. Поэтому применение правила Рунге для получения оценки погрешности при использовании формул Гаусса является неэффективным.

Вместо правила Рунге для формул Гаусса могут быть использованы различные подходы. Простейший из них следующий. Выполним вычисления по двум формулам Гаусса с n и $n + 1$ узлами. На это требуется $2n + 1$ вычисление $f(x)$. Разность между ними примем за оценку погрешности, допущенную формулой с n узлами. Однако формула с $n+1$ узлами ненамного точнее формулы с n узлами, поэтому вычисление оценки погрешности примерно столь же трудоемко, как и вычисление самого интеграла.

Поэтому более эффективен другой подход. Возьмем формулу Гаусса с n узлами и построим другую формулу (она называется формулой Гаусса–Кронрода) с теми же n узлами и с $n+1$ дополнительными узлами:

$$\int\limits_a^b f(x)\,dx = \sum_{i=1}^{n} c_i f(x_i) + \sum_{j=1}^{n} d_j f(\xi_j) + R_{2n+1}(f).$$

Узлы ξ_j и коэффициенты c_i, d_j подбираются так, чтобы данная формула имела полиномиальный порядок точности $3n+1$. Выполним оба просчета — по формуле Гаусса с n узлами и по новой формуле с $2n+1$ узлами, затратив на это $2n+1$ вычислений значений подынтегральной функции. Разность между этими просчетами принимается за оценку погрешности. Таким образом при просчетах по формулам Гаусса с n и $n+1$ узлами и по формуле Гаусса–Кронрода с n и $2n+1$ узлами требуется одинаковое количество вычислений $f(x)$ в $2n+1$ точках, однако результат, полученный по формуле Гаусса–Кронрода, гораздо более точен.

Пользуясь свойствами формул Гаусса–Кронрода, алгоритм 1.8.1 можно представить следующим образом. Возьмем формулу Гаусса с тремя узлами, добавим четыре узла по Кронроду, а затем выполним вычисления по обеим формулам. Если их разность, являющаяся оценкой погрешности, больше предписанной точности, то добавляем еще 8 узлов (всего получается 15), вычисляем приближенное значение интеграла по новой формуле Гаусса–Кронрода и сравниваем полученный результат с результатом по предыдущей формуле. Если точность не достигнута, то добавляем еще 16 узлов (всего получается 31 узел) и т.д.

Данное видоизменение алгоритма 1.8.1 обеспечивает механизм получения все более точных результатов без потери ранее вычисленных значений подынтегральных функций.

Если заданная точность так и не оказалась достигнутой, то исходный отрезок $[a,\, b]$ можно разбить на два подотрезка и на каждом из них применить описанный алгоритм.

1.8.3. Адаптивный алгоритм последовательного передвижения “слева–направо”

Рассмотрим адаптивный алгоритм (т.е. алгоритм, учитывающий при контроле точности локальное поведение подынтегральной функции), который широко применяется при интегрировании обыкновенных дифференциальных уравнений.

Зададимся некоторым шагом $h \le b - a$ и вычислим приближенное значение I_h интеграла I по выбранной элементарной квадратурной формуле на отрезке $[a,\, a+h]$. Затем вычислим по той же формуле два приближения на отрезках $[a,\, a+h/2]$ и $[a+h/2,\, a+h]$ и сложим их. Полученную сумму обозначим $I_{h/2}$. Вычислим локальную погрешность на шаге h по первой формуле Рунге:

$$\delta_h = \frac{I_{h/2} - I_h}{2^p - 1}.$$

Если $|\delta_h| > \max(\varepsilon_A,\, \varepsilon_O \cdot |I_{h/2}|)$, то положим $h = h/2$ и повторим вычисления.

В противном случае считаем, что локальная погрешность обеспечивает заданную точность и текущий шаг, который обозначим через h_1, признается удачным.

Теперь возникает вопрос: каким должен быть следующий шаг? Здесь возможны два случая:

— либо следующий шаг выбирается равным предыдущему,

— либо, если текущая локальная погрешность очень мала (например, мы находимся в области гладкости подынтегральной функции), то следующий шаг следует выбрать бо́льшим, чем предыдущий.

Часто выбирают следующую стратегию. Если

$$|\delta_h| > \frac{1}{2^p} \max\bigl(\varepsilon_A,\, \varepsilon_O \cdot |I_{h/2}|\bigr),$$

то следующий шаг h_2 полагается равным h_1, в противном случае берем $h_2 = 2h_1$. Такое удвоение обосновывается возможным прогнозом о гладкости подынтегральной функции на следующем шаге. Показатель p выбирается из таких соображений. Для следующего шага h_2 имеет место соотношение

$$I - I_{h_2/2} \approx c\Bigl(\frac{h_2}{2}\Bigr)^p,$$

причем на предыдущем шаге h_1 было выполнено

$$c\Big(\frac{h_1}{2}\Big)^p < \frac{1}{2^p}\max(\varepsilon_A,\, \varepsilon_O \cdot |I_{h_1/2}|).$$

Это означает, что локальная погрешность излишне мала и мы, вероятно, можем увеличить следующий шаг, чтобы уменьшить вычислительную работу на гладких участках. Действительно, предполагая, что постоянная c мало изменится из-за гладкости $f(x)$, возьмем $h_2 = 2h_1$. Тогда возможно соотношение

$$c\Big(\frac{h_2}{2}\Big)^p = c\ (h_1)^p \le \max(\varepsilon_A,\, \varepsilon_O \cdot |I_{h_1/2}|),$$

т.е. предположительно (с некоторой вероятностью) шаг $h_2 = 2h_1$ окажется удачным. Выбрав следующий шаг, проверяем, не выйдет ли он за правый предел отрезка $[a,\, b]$. Если нет, то повторяем процесс, передвигаясь слева направо, если да, то предварительно меняем следующий шаг, полагая его равным разности между b и достигнутой точкой отрезка.

Итак, строится последовательность шагов h_i, значений $I_{h_i/2}$, которые суммируются (по достижении конца отрезка их сумма дает приближенное значение всего интеграла), и локальных оценок δ_i достигнутой точности на шаге h_i. Теперь мы видим, что там, где подынтегральная функция резко меняется, значения h_i уменьшаются, а на гладких участках увеличиваются, и лишние вычисления не выполняются. Таким образом, алгоритм как бы адаптируется к характеру поведения $f(x)$.

Критерием достигнутой точности мы выбираем

$$|\delta_i| < \max\Big(\varepsilon_A,\, \varepsilon_O \sum_{j=1}^{i} |I_{h_j/2}|\Big).$$

Однако локальные ошибки δ_i при передвижении суммируются и, если они оказываются одного знака, то в худшем случае общая достигнутая точность E, равная их сумме на всем отрезке (ее еще называют глобальной ошибкой), может значительно превзойти заданную точность. Покажем, какой критерий контроля точности можно задать, чтобы и в этом худшем случае глобальная ошибка не превосходила заданной точности ε (ведь обычно задают желаемую величину малости не для локальной, а именно для глобальной ошибки). Таким критерием является

$$|\delta_i| \le \frac{h_i}{b-a}\,\varepsilon. \tag{15}$$

Действительно, обозначив через n общее число шагов, а через I_i — точное значение интеграла на i-м отрезке, имеем:

$$E = |I - I| = \Big|\sum_{i=1}^{n} \big(I_{h_i/2} - I_i\big)\Big| \le \sum_{i=1}^{n} |I_{h_i/2} - I_i| \le$$
$$\le \sum_{i=1}^{n} \left|\frac{I_{h_i/2} - I_{h_i}}{2^p - 1}\right| = \sum_{i=1}^{n} |\delta_i| \le \sum_{i=1}^{n} \frac{h_i}{b-a}\,\varepsilon = \varepsilon.$$

Итак, более строгий критерий ориентирован на худший случай. Однако естественно ожидать, что в сумме $E = \sum_{i=1}^{n} \delta_i$ будут встречаться δ_i разных знаков, которые будут взаимно уничтожаться, хотя и не полностью. Поэтому при использовании критерия (15) значение $|E|$ может оказаться излишне меньшим ε, т.е. результат будет получен с завышенной точностью. Вследствие этого часто использую более слабый критерий, задавая для локальной ошибки значение ε примерно равным $\tilde{\varepsilon}/(b-a)$, где $\tilde{\varepsilon}$ — требуемая точность для всего отрезка интегрирования. При этом достигается экономия вычислительной работы.

Для формул Ньютона–Котеса замкнутого типа на равномерной сетке дополнительная экономия достигается также, если запоминать значения подынтегральной функции в общих, уже использованных узлах. Например, если в качестве расчетной формулы была выбрана формула Симпсона, то на шаге h_i она она использует три узла, а на двух шагах $h_i/2$ — пять узлов, причем три из них уже использованы. Следовательно, во втором применении формулы Симпсона требуется лишь два новых значения $f(x)$. Ясно, что при передвижении вправо с шагом h_{i+1} должны использоваться значения $f(x)$, уже вычисленные в правом конце i-го подотрезка.

Кроме того, при каждом делении h_i пополам следует запоминать узлы и значения $f(x)$ из правой половины для последующего использования. Поскольку надо заранее учитывать максимальный объем памяти ЭВМ для хранения этих величин, то задается предел HALFMAX на уровень деления отрезка пополам. Когда этот максимум достигнут, то текущий подотрезок считается в любом случае приемлемым, даже если локальная ошибка на нем слишком велика. Длина такого подотрезка может быть сделана совсем небольшой, если выбрать HALFMAX достаточно большим: например, если HALFMAX=30, то длина подотрезка равна $(b-a)/2^{30}$.

Отметим возникающее при этом важное обстоятельство. Появление такого рода подотрезка заданной минимальной длины означает, что $f(x)$ имеет внутри него некоторую особенность (например, неинтегрируемую особенность). Поэтому необходимо сообщить пользователю границы подотрезка для последующего осмысления результата. Поскольку длина такого подотрезка мала, то можно надеяться, что локальная ошибка на нем все же не очень большая и при дальнейшем интегрировании она может быть компенсирована другими локальными ошибками, имеющими противоположные знаки. Поэтому процесс разумно продолжить, а не прерывать расчеты. При таком подходе мы имеем возможность не только получить в конечном счете значение интеграла с глобальной ошибкой, меньшей предписанного $\tilde{\varepsilon}$, но и выявить все точки особенностей $f(x)$, препятствующие получению локальных погрешностей с приемлемой точностью.

Такой подход к составлению программы численного интегрирования позволяет всегда получить приближенное значение интеграла; при этом, если предписанная точность не достигнута, то выдается наилучший возможный для нее результат, оценка E реально достигнутой точности и, если это имеет место, точки особенностей (неинтегрируемости) $f(x)$.

Если в случае формул Ньютона–Котеса замкнутого типа удается достигнуть максимальной экономии вычислений $f(x)$, то для формул Гаусса этого сделать нельзя, поскольку узлы при их применении на всем подотрезке и на двух его половинах будут совершенно различными. Поэтому применение правила Рунге для оценки локальной погрешности для формул Гаусса будет неэффективно с точки зрения производимых затрат. Для уменьшения трудоемкости применяют формулы Гаусса–Кронрода либо выбирают формулу Гаусса с небольшим числом узлов (например, $n = 5$). В последнем случае поступают так: на текущем подотрезке вычисляют значения интеграла по пяти и шести узлам, а в качестве оценки локальной ошибки берут разность между ними. При этом вследствие высокого порядка точности формул Гаусса можно надеяться, что продвижение по $[a, b]$ слева направо будет идти с относительно большими шагами, что позволит снизить общие трудозатраты, обусловленные двумя независимыми расчетами при $n = 5$ и $n = 6$ на каждом подотрезке для оценки локальной ошибки.

Дополнительное улучшение может быть сделано из практически подтвержденного наблюдения, что для многих подотрезков достигается бо́льшая точность, чем это требуется, тогда как для некоторых подотрезков заданная граница погрешности превышается ненамного. Поэтому может быть использован прием, получивший название "банкирование": мы "вкладываем в банк" разность между предписанной точностью и оценкой локальной погрешности для приемлемого подотрезка с тем, чтобы использовать ее впоследствии для уменьшения точности на других интервалах, на которых предписанная точность не достигается. Это может быть сделано, например, так: если $|\delta_i| > \max\left(\varepsilon_A, \varepsilon_O \sum\limits_{j=1}^{i} |I_{h_j/2}|\right)$, то проверяется менее жесткое условие $|\delta_i| < \max\left(\varepsilon_A, \varepsilon_O \sum\limits_{j=1}^{i} |I_{h_j/2}|\right) + \theta \cdot \mathrm{BANK}$, в котором $0 < \theta < 1$ — параметр, ограничивающий максимальное изъятие из "банка". Можно предусмотреть операцию "кредитования", когда θ берется бо́льшим 1. Тогда "кредит" должен быть "погашен" последующими подотрезками.

Выпишем одну из реализаций алгоритма передвижения "слева–направо" в упрощенной форме, оставляя в стороне вопрос использования ранее вычисленных значений $f(x)$ и возможность удвоения текущего шага интегрирования:

начало $\alpha = a$; $\beta = b$; $I = 0$; $E = 0$; $k = 0$; $kmax = 0$;
пока $\alpha < \beta$
  $h_i = \beta - \alpha$; I_{h_i}; $I_{h_i/2}$; $\delta_i = \left(I_{h_i/2} - I_{h_i}\right)/(2^p - 1)$;
    если $|\delta_i| < \max\left(\varepsilon_A,\, \varepsilon_O\,|I|\right) h_i/(b-a)$;
      $k = 0$; $\alpha = \beta$; $\beta = b$;
      $I = I + I_{h_i/2} + \delta_i$; $E = E + \delta_i$;
    иначе
      $\beta = (\alpha + \beta)/2$;
      $k = k + 1$;
      если $k > kmax$
        $k = 0$;
        печать(“точка особенности” $(\alpha,\, \beta)$);
        $I = I + I_{h_i/2} + \delta_i$; $E = E + \delta_i$;
        $\alpha = \beta$; $\beta = b$;
печать$(I,\, E)$;
конец

Эта модификация выписана для формул Ньютона–Котеса с контролем локальной точности по правилу Рунге. В случае формул Гаусса можно положить $I_{h_i} = G_5$, $I_{h_i/2} = G_6$, где G_5 и G_6 — результаты вычисления интеграла на шаге h_i по формулам Гаусса с 5 и 6 узлами соответственно и $\delta_i = G_6 - G_5$, либо применить формулы Гаусса–Кронрода.

1.8.4. Адаптивный алгоритм с контролем точности по глобальной ошибке

Рассмотренный алгоритм 1.8.3 при движении “слева–направо” реализует локальный способ контроля точности, поскольку вывод о том, удовлетворяет ли полученная локальная оценка интеграла на каждом подотрезке предписанной точности или нет, делается независимо от локальных оценок на других подотрезках. Это означает, что порядок, в котором выбираются подотрезки, не играет никакой роли для получения конечного результата. Таким образом, выбор очередного подотрезка для разбиения следует делать целиком из соображений легкости программной реализации, т.к. он не отражается на эффективности алгоритма. Алгоритм использует принцип “слева–направо”, когда разбивается самый левый подотрезок, для которого локальная ошибка велика. Такой принцип нетруден для программирования и при относительно небольших затратах памяти дает возможность запоминать, а затем использовать ранее вычисленные значения подынтегральной функции.

Однако можно предложить алгоритм, в котором проверка локальных ошибок не проводится вовсе. Вместо этого контроль точности осуществляется по глобальной погрешности. Выпишем схему алгоритма:

начало $\alpha = a;\ \beta = b;\ I = 0;\ E = 0;\ h_i = \beta - \alpha;$
  $I_{h_i};\ I_{h_i/2};\ \delta_i = (I_{h_i/2} - I_{h_i})/(2^p - 1);$
  $E = E + \delta_i; I = I + I_{h_i/2} + \delta_i;$
пока $E > \max(\varepsilon_A,\, \varepsilon_O \cdot |I|)$
  поиск *i-го* подотрезка, на котором достигается $\max|\delta_i|$;
  для *i-го* подотрезка :
    $E = E - \delta_i;\ I = I - I_{h_i/2} - \delta_i;\ h_i = h_i/2;$
    $I^{(1)}_{h_i};\ I^{(1)}_{h_i/2};\ \delta^{(2)}_i$ для левого подотрезка
    $I^{(2)}_{h_i};\ I^{(2)}_{h_i/2};\ \delta^{(2)}_i$ для правого подотрезка
    $E = E + \delta^{(1)}_i + \delta^{(2)}_i;\ I = I + I^{(1)}_{h_i/2} + I^{(2)}_{h_i/2} + \delta^{(1)}_i + \delta^{(2)}_i;$
печать $(I,\, E)$
конец

Здесь мы оставили в стороне вопрос использования уже вычисленных значений $f(x)$ и выделения подотрезка, содержащего особенность (если она существует). Кроме того, и это относится ко всем другим алгоритмам, можно задать максимально допустимое количество вычислений значений подынтегральной функции с тем, чтобы избежать слишком долгой работы программы; при достижении этого максимума выдается полученное приближение к интегралу и оценка глобальной погрешности.

Остановимся только на главных аспектах. В рассматриваемом алгоритме принцип перемещения слева направо вдоль отрезка интегрирования полностью отсутствует. Алгоритм можно назвать глобальным, поскольку в нем лишены смысла проверки приемлемости подотрезков с точки зрения малости локальных ошибок. Вычисления заканчиваются, когда глобальная ошибка становится меньше предписанной точности.

Конечно, данный алгоритм более труден в реализации, чем рассмотренные ранее. Количество подотрезков, предназначенных для разбиения, растет, особенно на начальных этапах вычислений. Чтобы смягчить возникающие при этом трудности с памятью, все-таки приходиться проверять на малость δ_i с тем, чтобы отбросить "хорошие" подотрезки. Для хранения "плохих" подотрезков и связанной с ними информации удобно использовать стековую организацию структуры данных.

2. ИНТЕГРАЛЬНЫЕ УРАВНЕНИЯ ФРЕДГОЛЬМА ВТОРОГО РОДА

Пусть на конечном отрезке $[a, b]$ задана функция $f(x)$ и функция двух переменных $K(x, s)$. Требуется найти функцию $u(x)$, удовлетворяющую уравнению

$$u(x) - \int\limits_a^b K(x, s)u(s)\, ds = f(x), \quad x \in [a, b]. \tag{16}$$

Из теории интегральных уравнений известно, что если функция $K(x, s)$ (ядро интегрального уравнения) отвечает условию

$$\int\limits_a^b \int\limits_a^b K^2(x, s)\, dx\, ds < \infty,$$

а однородное интегральное уравнение

$$u(x) - \int\limits_a^b K(x, s)u(s)\, ds = 0$$

имеет только нулевое решение, то уравнение (16) имеет при любой $f(x) \in L_2[a, b]$ одно и только одно решение.

2.1. Метод квадратур решения интегральных уравнений Фредгольма второго рода

Одним из наиболее распространенных методов, применяемых для решения уравнения (16), является метод квадратур, состоящий в замене входящего в левую часть уравнения интеграла какой-либо формулой численного интегрирования. В дальнейшем будем считать, что при некотором целом $m > 0$ функции

$$\frac{d^m f(x)}{dx^m}, \quad \frac{\partial^m K(x, s)}{\partial x^m}, \quad \frac{\partial^m K(x, s)}{\partial s^m}$$

являются непрерывными. Продифференцировав m раз обе части уравнения (16), получим

$$\frac{d^m u(x)}{dx^m} = \int\limits_a^b \frac{\partial^m K(x,s)}{\partial x^m}\, u(s)\, ds + \frac{d^m f(x)}{dx^m}.$$

Таким образом, функция $u(x)$ (решение интегрального уравнения) обладает той же гладкостью, что правая часть и ядро.

Для функций с непрерывной m-й производной введем в рассмотрение квадратурную формулу

$$\int\limits_a^b g(s)\, ds = \sum_{j=1}^{n} A_j^{(n)} g\big(s_j^{(n)}\big) + O\Big(\frac{1}{n^m}\Big). \tag{17}$$

При $m = 2$ это может быть составная формула трапеций или средних прямоугольников, при $m = 4$ — составная формула Симпсона или Гаусса по двум узлам и т.д.

В дальнейшем будем считать, что решение интегрального уравнения (16) существует и единственно.

В уравнении (16) заменим интеграл по формуле (17):

$$u(x) - \sum_{j=1}^{n} A_j^{(n)} K\big(x, s_j^{(n)}\big) u\big(s_j^{(n)}\big) = f(x) + R_n(x), \quad x \in [a,b],$$

где $\big|R_n(x)\big| \le \mathrm{const} \cdot n^{-m}$. Отбросим малую величину $R_n(x)$ и получим уравнение

$$u_n(x) - \sum_{j=1}^{n} A_j^{(n)} K\big(x, s_j^{(n)}\big) u_n\big(s_j^{(n)}\big) = f(x), \quad x \in [a,b]. \tag{18}$$

Функцию $u_n(x)$, если она существует, примем за приближенное решение уравнения (16).

Процесс решения уравнения (18) может быть сведен к решению системы линейных алгебраических уравнений

$$U_i^{(n)} - \sum_{j=1}^{n} A_j^{(n)} K\big(s_i^{(n)}, s_j^{(n)}\big) U_j^{(n)} = f\big(s_i^{(n)}\big), \quad i = 1, 2, \ldots, n. \tag{19}$$

Действительно, для любого решения u_n уравнения (18) величины $U_i^{(n)} = u_n\bigl(s_i^{(n)}\bigr)$, $i = 1, 2, \ldots, n$, удовлетворяют системе (19), и наоборот, если имеется решение $U_i^{(n)}$, $i = 1, 2, \ldots, n$, системы (19), то функция

$$u_n(x) = \sum_{j=1}^{n} A_j^{(n)} K\bigl(x, s_j^{(n)}\bigr) U_j^{(n)} + f(x) \tag{20}$$

удовлетворяет уравнению (18).

Таким образом, можно выделить следующие этапы алгоритма численного решения уравнения (16):

— выбирается квадратурная формула (17),

— строится и решается система линейных алгебраических уравнений (19),

— по ее решению находится приближенное решение исходного интегрального уравнения в виде функции $u_n(x)$ по формуле (20).

2.1.1. Сходимость метода квадратур

Для доказательства осуществимости предложенного алгоритма необходимо показать, что система (19) (или, что то же самое, уравнение (18)) однозначно разрешима при достаточно больших n и что последовательность функций $\bigl\{u_n(x)\bigr\}$ сходится по норме к решению уравнения (16) при $n \to \infty$.

Запишем уравнение (16) в операторной форме

$$u - Ku = f, \tag{21}$$

где через K обозначен линейный ограниченный оператор на пространстве функций из $C\bigl([a, b]\bigr)$:

$$\bigl(Kv\bigr)(x) = \int_a^b K(x, s) v(s)\, ds.$$

Введем в рассмотрение линейный ограниченный оператор K_n:

$$\bigl(K_n v\bigr)(x) = \sum_{j=1}^{n} A_j^{(n)} K\bigl(x, s_j^{(n)}\bigr) v\bigl(s_j^{(n)}\bigr)$$

и исследуем вопрос о разрешимости уравнения

$$u_n - K_n u_n = f. \tag{22}$$

Теорема 1 (без доказательства). *Существует такое целое $N_1 > 0$, что при всех $n \ge N_1$ выполнено*

$$\left\| \left(I - K_n\right)^{-1} \right\|_C \le c_1, \tag{23}$$

где постоянная c_1 не зависит от выбора n.

Теорема 2. *Существует такое целое $N_2 > 0$, что при всех $n \ge N_2$ решение u_n уравнения*

$$u_n + K_n u_n = f$$

существует и единственно, причем выполнена оценка

$$\|u - u_n\|_C \le \mathrm{const} \cdot n^{-m},$$

где u является решением уравнения (16), *а постоянная не зависит от выбора n.*

Доказательство. Оценка (23) обеспечивает существование и единственность решения уравнения (22). Вычтем уравнение (22) из уравнения (21) и в полученном выражении прибавим и вычтем $K_n u$. Тогда получим равенство

$$(I - K_n)(u - u_n) = (K - K_n)\, u.$$

Отсюда следует оценка

$$\begin{aligned} \|u - u_n\|_C &= \left\|(I - K_n)^{-1}(K - K_n)\, u\right\|_C \le \\ &\le c_1 \left\|(K - K_n)\, u\right\|_C \le c_2 \left\|u^{(n)}\right\|_C n^{-m}. \end{aligned}$$

Теорема доказана.

Пример. Решается уравнение

$$u(x) - \frac{1}{2} \int\limits_0^1 x e^s u(s)\, ds = e^{-x}.$$

Выберем узлы $s_1 = 0, s_2 = 0.5, s_3 = 1$ и используем квадратурную формулу Симпсона. Для определения приближенных значений U_i, $i = 1, 2, 3$, решения $u(x)$ в узлах сетки s_i получаем систему

$$\begin{aligned}
U_1 &= 1, \\
U_2 - \frac{0.5}{6}\left(0,5\,U_1 + 3.2976\,U_2 + 1.3592\,U_3\right) &= 0.6065, \\
U_3 - \frac{0.5}{6}\left(U_1 + 6.5948\,U_2 + 2.7183\,U_3\right) &= 0.3679,
\end{aligned}$$

решением которой являются

$$U_1 = 1, \quad U_2 = 1.1079, \quad U_3 = 1.3706.$$

Построим в соответствии с (20) приближенное решение рассматриваемого уравнения:

$$u_3(x) = e^{-x} + 1.003\,x.$$

Заметим, что точное решение уравнения есть $u(x) = e^{-x} + x$.

2.1.2. Практический алгоритм численной реализации метода квадратур

Для численного решения уравнения (16) зададим семейство квадратурных формул S_{n_k} вида (17). Переход от n_k к n_{k+1} определяется увеличением количества элементарных отрезков разбиения в составных квадратурных формулах (17) (удвоение или увеличение на фиксированное число). Для каждого n_k из полученной последовательности составляем систему линейных алгебраических уравнений вида (19) и строим приближенное решение $u_{n_k}(x)$ интегрального уравнения (16).

Вычисления прекращаются, когда

$$\|u_{n_{k-1}} - u_{n_k}\| \le \varepsilon,$$

где ε — предписанная точность. Интеграл в левой части этой оценки вычисляется посредством одного из описанных выше алгоритмов с автоматическим выбором шага.

Таким образом, алгоритм численной реализации метода квадратур определяется выбором элементарной квадратуры, лежащей в основе составной формулы (17), выбором метода решения аппроксимирующей системы линейных алгебраических уравнений (19) и выбором алгоритма численного интегрирования с автоматическим распределением узлов.

2.2. Метод последовательных приближений

Итерационные методы позволяют получить относительно простые численные алгоритмы решения интегральных уравнений. Кроме того, они становятся неизбежными при решении многих нелинейных задач. Примером может служить процесс решения нелинейных интегральных уравнений методом квадратур, который, несмотря на дискретизацию задачи, не освобождает от необходимости применять итерационные процедуры при решении аппроксимирующих нелинейных конечных уравнений.

Итерационные методы могут иметь различное назначение: для теоретического исследования задач с целью доказательства существования и единственности решений; для приближенного аналитического решения уравнений, когда в качестве решения принимается аналитическое выражение какого-либо приближения, и, наконец, для получения приближенных численных решений. При реализации на ЭВМ основные недостатки итерационных методов состоят в проблеме сходимости — процесс должен быть сходящимся, а скорость сходимости высокой.

Из всего многообразия итерационных методов мы приведем наиболее простой — метод последовательных приближений.

Рассмотрим интегральное уравнение Фредгольма второго рода

$$u(x) - \int_a^b K(x,s)u(s)\,ds = f(x) \tag{24}$$

и предположим, что

$$\max_{x,s}\bigl|K(x,s)\bigr| \le M, \quad \max_{x}\bigl|f(x)\bigr| \le N, \quad x,s \in [a,b].$$

Будем искать решение уравнения (24) в виде ряда

$$u(x) = \sum_{k=0}^{\infty} \varphi_k(x), \tag{25}$$

где

$$\varphi_0(x) = f(x), \quad \varphi_{k+1} = \int\limits_a^b K(x,s)\varphi_k(s)\,ds, \quad k \geq 0. \tag{26}$$

Если выполнено условие

$$M(b-a) < 1, \tag{27}$$

то ряд (25) абсолютно сходится, при этом

$$\|\varphi_k\|_C \leq M^k N(b-a)^k.$$

Если в качестве приближенного решения уравнения (24) взять первые n слагаемых из (25), то можно оценить погрешность:

$$\left\| u - \sum_{k=0}^{n} \varphi_k \right\| \leq \sum_{k=n+1}^{\infty} \|\varphi_k\|_C \leq \frac{N\big(M(b-a)\big)}{1 - M(b-a)}.$$

Данная оценка относится к погрешности приближенного решения уравнения (24) при точном вычислении интегралов, т.е. не учитывает ошибок, возникающих в случае применения квадратурных формул.

На практике разложение (25), (26) реализуют в несколько иной форме, более удобной для вычислений: строится последовательность функций $\big\{u_n(x)\big\}$, являющихся приближениями к искомому решению уравнения (24).

Начальное приближение $u_0(x)$ выбирается произвольно (например, $u_0(x) = f(x)$). Последующие приближения строятся по рекуррентной формуле

$$u_{n+1}(x) = f(x) + \int\limits_a^b K(x,s)u_n(s)\,ds, \quad k = 1, 2, \ldots\ . \tag{28}$$

Пример. Методом последовательных приближений решить уравнение
$$u(x) - \frac{1}{2}\int\limits_0^1 u(s)\,ds = \sin\pi x.$$
В данном случае $K(x,s) = 1/2$ и условие (27) выполнено. Положим $u_0(x) = \sin\pi x$ и последовательно находим
$$u_1(x) = \sin\pi x + \frac{1}{2}\int\limits_0^1 u_0(s)\,ds = \sin\pi x + \frac{1}{\pi},$$
$$u_2(x) = \sin\pi x + \frac{1}{2}\int\limits_0^1 u_1(s)\,ds = \sin\pi x + \frac{1}{\pi} + \frac{1}{2\pi},$$
$$u_3(x) = \sin\pi x + \frac{1}{2}\int\limits_0^1 u_2(s)\,ds = \sin\pi x + \frac{1}{\pi} + \frac{1}{2\pi} + \frac{1}{2^2\pi}.$$
Нетрудно заметить, что
$$u_n(x) = \sin\pi x + \frac{1}{\pi}\sum_{k=0}^{n-1}\frac{1}{2^k},$$
откуда
$$\lim_{n\to\infty} u_n(x) = \sin\pi x + \frac{2}{\pi}.$$
Поэтому решением уравнения является функция
$$u(x) = \sin\pi x + \frac{2}{\pi},$$
в чем можно убедиться непосредственной проверкой.

В дополнение к условию (27) сходимости ряда (25) достаточным условием сходимости для итерационного процесса (28) является выполнение неравенств
$$q_1 = \max_{a\le x\le b}\int\limits_a^b \big|K(x,s)\big|\,ds < 1$$

или

$$q_2 = \left(\int_a^b \int_a^b \left| K(x,s) \right|^2 dx\, ds \right)^{1/2} < 1.$$

При этом априорные оценки погрешностей определяются следующими неравенствами:

$$\| u - u_n \|_C \le q_1^n \| u - u_0 \|_C, \quad \| u - u_n \|_{L_2} \le q_2^n \| u - u_0 \|_{L_2},$$

а апостериорные — неравенствами

$$\| u - u_n \|_C \le \frac{q_1^n \| u_n - u_0 \|_C}{1 - q_1^n}, \quad \| u - u_n \|_{L_2} \le \frac{q_2^n \| u_n - u_0 \|_{L_2}}{1 - q_2^n}.$$

Итерационный процесс (28) заканчивается при том значении n, при котором какая-либо апостериорная оценка погрешности не окажется меньше требуемой точности вычисления решения исходного уравнения.

Важным свойством линейных уравнений является независимость сходимости последовательных приближений от вида правой части $f(x)$ и начального приближения $u_0(x)$, которые влияют на скорость сходимости. В некоторых случаях полезно использовать оценку ошибки k-го приближения, представленную в виде

$$\left| u(x) - u_k(x) \right| \le Q_f Q_k q_2^{-1} \frac{q_2^k}{1 - q_2} + Q_0 Q_k q_2^{k-1}, \tag{29}$$

где

$$Q_f = \left(\int_a^b f^2(x)\, dx \right)^{1/2}, \quad Q_0 = \left(\int_a^b u_0^2(x)\, dx \right)^{1/2},$$

$$Q_k = \left(\max_{a \le x \le b} \int_a^b K^2(x,s)\, ds \right)^{1/2}.$$

Пример. Решается уравнение

$$u(x) - \int_0^1 x s^2 u(s)\, ds = 1.$$

Для определения сходимости оценим

$$q_2 = \sqrt{\int_0^1 \int_0^1 x^2 s^4 \, dx \, ds} = \frac{1}{15}.$$

По приведенному выше достаточному условию итерационный процесс (28) сходится. Положив $u_0 = x$, вычисляем:

$$u_1(x) = 1 + \int_0^1 xs^2 s \, ds = 1 + \frac{x}{4},$$

$$u_2(x) = 1 + \int_0^1 xs^2 \left(1 + \frac{s}{4}\right) ds = 1 + \frac{19x}{48},$$

$$u_3(x) = 1 + \int_0^1 xs^2 \left(1 + \frac{19s}{48}\right) ds = 1 + \frac{83x}{192}.$$

Остановившись на третьем приближении, оцениваем ошибку по формуле (29), для чего вычисляем

$$Q_f = \left(\int_0^1 dx\right)^{1/2} = 1, \quad Q_0 = \left(\int_0^1 x^2 \, dx\right)^{1/2} = \frac{1}{\sqrt{3}},$$

$$Q_k = \left(\max_{0 \le x \le 1} \int_0^1 x^2 s^4 \, ds\right)^{1/2} = \frac{1}{\sqrt{5}}.$$

Тогда

$$\left|u(x) - u_3(x)\right| \le 0.044.$$

2.2.1. Первый алгоритм численной реализации метода последовательных приближений

Зададим составную квадратурную формулу

$$S_N(F)=\sum_{j=1}^{N} A_j^{(N)} F\big(s_j^{(N)}\big), \tag{30}$$

такую, что

$$\int_a^b F(s)\,ds = S_N(F)+O\left(N^{-m}\right), \quad m>0.$$

На начальном этапе положим

$$u_0^{(N)}\big(s_j^{(N)}\big)=f\big(s_j^{(N)}\big), \quad j=1,2,\ldots,N.$$

Пусть при некотором $n\ge 1$ уже вычислены величины

$$u_{n-1}^{(N)}\big(s_j^{(N)}\big), \quad j=1,2,\ldots,N.$$

Тогда значения приближения $u_n^{(N)}(x)$ в узлах квадратурной формулы получаем на n-м шаге следующего итерационного процесса:

$$u_n^{(N)}\big(s_i^{(N)}\big)=f\big(s_i^{(N)}\big)+\sum_{j=1}^{N} A_j^{(N)} K\big(s_i^{(N)},s_j^{(N)}\big)u_{n-1}^{(N)}\big(s_j^{(N)}\big)$$

при $i=1,2,\ldots,N$. Процесс заканчивается при таком значении n, что

$$\frac{1}{n}\sum_{i=1}^{n}\Big|u_n^{(N)}\big(s_i^{(N)}\big)-u_{n-1}^{(N)}\big(s_i^{(N)}\big)\Big|\le\varepsilon,$$

где ε — заданная абсолютная погрешность вычисления решения уравнения (24).

Как только такое n найдено, строим приближенное решение

$$u^{(N)}(x)=f(x)+\sum_{j=1}^{N} A_j^{(N)} K\big(x,s_j^{(N)}\big)u_n^{(N)}\big(s_j^{(N)}\big),$$

определенное во всех точках x отрезка $[a, b]$.

Зададим теперь семейство квадратурных формул $\{S_{N_k}\}$ вида (30), таких, что последовательность $\{N_k\}$ монотонно возрастает по некоторому заданному правилу (например, удвоение количества элементарных отрезков разбиения).

Используя приведенную схему вычислений, строим последовательность приближенных решений $\left\{u^{(N_k)}(x)\right\}$ до тех пор, пока не будет выполнено неравенство

$$\left(\int_a^b \left(u^{(N_k)}(x) - u^{(N_{k-1})}(x)\right)^2 dx\right)^{1/2} \leq \varepsilon$$

В этом неравенстве интеграл может быть вычислен одним из алгоритмов численного интегрирования, рассмотренных в п. 1.8.

2.2.2. Второй алгоритм численной реализации метода последовательных приближений

Как и в первом алгоритме, метод базируется на составной квадратурной формуле S_N вида (30). Зафиксируем число N и построим последовательность функций $u_k^{(N)}(x)$ по следующему правилу. Положим

$$u_0^{(N)}(x) = f(x).$$

Далее вычисляем по рекуррентной формуле

$$u_{k+1}^{(N)}(x) = f(x) + \sum_{j=1}^{N} A_j^{(N)} K\left(x, s_j^{(N)}\right) u_k^{(N)}\left(s_j^{(N)}\right).$$

Вычисления прекращаются, когда

$$\left(\int_a^b \left(u_n^{(N)}(x) - u_{n-1}^{(N)}(x)\right)^2 dx\right)^{1/2} \leq \varepsilon \qquad (31)$$

Далее введем приближенное решение, определяемое квадратурной формулой (30):

$$u^N(x) = u_n^{(N)}(x),$$

где номер n определяется соотношением (31).

Теперь построим семейство S_{N_k}, $k = 1, 2, 3, \ldots$, составных квадратурных формул вида (30), где $S_{N_1} \equiv S_N$, а S_{N_k} получается из $S_{N_{k-1}}$ путем удвоения числа элементарных отрезков.

Каждая из этих квадратурных формул позволяет построить описанным выше способом приближенное решение u^{N_k}. Вычисления прекращаются, когда

$$\left(\int\limits_a^b \left(u^{N_k}(x) - u^{N_{k-1}}(x) \right)^2 \, dx \right)^{1/2} \le \varepsilon$$

3. ИНТЕГРАЛЬНЫЕ УРАВНЕНИЯ ВОЛЬТЕРРА ВТОРОГО РОДА

Из теории интегральных уравнений известно, что если ядро $K(x, s)$ — непрерывная функция в области $\{a \leq s \leq x \leq b\}$, а $f(x)$ — непрерывная функция на отрезке $[a, b]$, то интегральное уравнение Вольтерра второго рода

$$u(x) - \int_a^x K(x, s) u(s)\, ds = f(x) \tag{32}$$

имеет единственное решение $u(x)$.

3.1. Метод квадратур решения интегральных уравнений Вольтерра второго рода

Для приближенного решения уравнения (32) можно применить метод прямой замены интеграла, входящего в уравнение, какой-либо квадратурной формулой.

Допустим, что на отрезке $[a, b]$ взята сетка равноотстоящих точек с шагом h: $s_k = a + kh$, $k = 0, 1, \ldots, n$, $a + nh \leq b < a + (n+1)h$. Положим $x = s_k$ в интегральном уравнении (32) и рассмотрим систему равенств

$$\begin{aligned} &u(s_0) = f(a), \\ &u(s_k) - \int_a^{s_k} K(s_k, s) u(s)\, ds = f(s_k), \quad k = 1, \ldots, n\,. \end{aligned} \tag{33}$$

Пусть для вычисления интегрального члена в последнем равенстве взята квадратурная формула с узлами $s_0, s_1, \ldots, s_k$:

$$\int_a^{s_k} K(s_k, s) u(s)\, ds = h \sum_{j=0}^{k} A_{kj}^{(n)} K(s_k, s_j) u(s_j) + R_k^{(n)}(u), \tag{34}$$

где $R_k^{(n)}(u)$ — погрешность используемой квадратурной формулы и коэффициенты $A_{kj}^{(n)}$ квадратурной формулы не зависят от

h. Подставляя выписанные представления интегралов в равенства (33), получим

$$u(s_0) = f(s_0),$$
$$u(s_k) - h\sum_{j=0}^{k} A_{kj}^{(n)} K(s_k, s_j)u(s_j) = f(s_k) + R_k^{(n)}(u),$$
$$k = 1, \ldots, n\,.$$

Отбрасывая в каждом равенстве малые величины $R_k^{(n)}(u)$ и обозначая приближенные значения искомого решения $u(x)$ в узлах s_k через $U_k^{(n)}$, получим систему линейных алгебраических уравнений относительно $U_k^{(n)}$:

$$U_0^{(n)} = f(s_0),$$
$$U_k^{(n)} - h\sum_{j=0}^{k} A_{kj}^{(n)} K(s_k, s_j)U_j^{(n)} = f(s_k), \tag{35}$$
$$k = 1, \ldots, n\,.$$

Эта система разрешима при выполнении условий

$$1 - h\,A_{kk}^{(n)} K(s_k, s_k) \neq 1, \quad k = 1, \ldots, n\,.$$

Однако при достаточно малых значений h мы потребуем выполнения более сильных ограничений:

$$h\,A_{kk}^{(n)} K(s_k, s_k) \leq q < 1, \quad k = 1, \ldots, n\,. \tag{36}$$

Теперь решение $U_k^{(n)}$ линейной системы (34) может быть представлено в следующем виде для $k = 1, \ldots, n$:

$$U_k^{(n)} = \Big(1 - hA_{kk}^{(n)} K(s_k, s_k)\Big)^{-1}\Big(f(s_k) + h\sum_{j=0}^{k-1} A_{kj}^{(n)} K(s_k, s_j)U_j^{(n)}\Big).$$

Пример. Задано уравнение

$$u(x) - \int\limits_0^x e^{-(x-s)}u(s)\,ds = e^{-x}, \quad x \in [0,\, 0.1].$$

Ищем решение в точках $s_i = 0.00; 0.02; 0.04; 0.06; 0.08; 0.10$. Воспользуемся обобщенной формулой трапеций для замены интеграла квадратурной суммой. Это приводит к последовательному вычислению приближенных значений $U_k^{(6)}, k = 1, 2, \ldots, 6$, по приведенной выше формуле для решения системы (35), где шаг $h = 0.02$. Обозначив $K_{ij} = e^{-(s_i - s_j)}$ и $f_i = e^{-s_i}$, в результате получим следующие равенства:

$$U_1^{(6)} = f_1 = 1.0000;$$
$$U_2^{(6)} = \Big(1 - \frac{h}{2}\,K_{22}\Big)^{-1}\Big(f_2 + \frac{h}{2}\,K_{21}U_1^{(6)}\Big) = 1.00001;$$
$$U_3^{(6)} = \Big(1 - \frac{h}{2}\,K_{33}\Big)^{-1}\Big(f_3 + \frac{h}{2}K_{31}U_1^{(6)} + hK_{32}U_2^{(6)}\Big) = 0.999405;$$
$$U_4^{(6)} = \Big(1 - \frac{h}{2}\,K_{44}\Big)^{-1}\Big(f_4 + \frac{h}{2}\,K_{41}U_1^{(6)} + h\,\Big(K_{42}U_2^{(6)} + K_{43}U_3^{(6)}\Big)\Big) = 1.000002;$$
$$U_5^{(6)} = \Big(1 - \frac{h}{2}\,K_{55}\Big)^{-1}\Big(f_5 + \frac{h}{2}\,K_{51}U_1^{(6)} + h\,\Big(K_{52}U_2^{(6)} + K_{53}U_3^{(6)} + K_{54}U_4^{(6)}\Big)\Big) = 0.999991;$$
$$U_6^{(6)} = \Big(1 - \frac{h}{2}\,K_{66}\Big)^{-1}\Big(f_6 + \frac{h}{2}\,K_{61}U_1^{(6)} + h\,\Big(K_{62}U_2^{(6)} + K_{63}U_3^{(6)} + K_{64}U_4^{(6)} + K_{65}U_5^{(6)}\Big)\Big) = 0.999991.$$

Отметим, что точное решение уравнения $u(x) \equiv x$.

3.2. Сходимость метода квадратур

Исследуем вопрос сходимости $U_k^{(n)}$ к значениям точного решения интегрального уравнения (32) в узлах сетки при $n \to \infty$. Для этого введем числа $A^{(n)}$ и K, определив их равенствами

$$A^{(n)} = \max_{k,j}\left|A_{kj}^{(n)}\right|, \quad K = \max_{x,s}\bigl|K(x,s)\bigr|.$$

Будем также считать, что при всех $k = 1, \ldots, n$ выполнены неравенства
$$\left|R_k^{(n)}(u)\right| \le R(h).$$

Теорема 3. *При выполнении условия* (36) *справедлива равномерная по всем k оценка*

$$\left|U_k^{(n)} - u(s_k)\right| \le \frac{1}{1-q}\, e^{(b-a)AK/(1-q)} R(h). \tag{37}$$

Доказательство. Рассмотрим погрешность приближенных значений $\varepsilon_k^{(n)} = U_k^{(n)} - u(s_k)$. Уравнение погрешности имеет вид

$$\varepsilon_k^{(n)} - h\sum_{j=0}^{k} A_{kj}^{(n)} K(s_k, s_j)\, \varepsilon_j^{(n)} = R_k^{(n)}(u), \quad k = 1, \ldots, n\,.$$

Перепишем это уравнение в виде

$$\Big(1 - hA_{kk}^{(n)} K(s_k, s_k)\Big)\, \varepsilon_k^{(n)} = R_k^{(n)}(u) + h\sum_{j=0}^{k-1} A_{kj}^{(n)} K(s_k, s_j)\, \varepsilon_j^{(n)}.$$

Используя (36) и приведенные выше обозначения, получим рекурсионное неравенство для погрешностей $\varepsilon_k^{(n)}$:

$$\left|\varepsilon_k^{(n)}\right| \le \frac{1}{1-q}\Big(R(h) + hA^{(n)}K\sum_{j=0}^{k-1}\left|\varepsilon_j^{(n)}\right|\Big).$$

Введем величины $E_k^{(n)}$, $k = 0, 1, \ldots, n$, определив их начальным условием $E_0^{(n)} = \left|\varepsilon_0^{(n)}\right|$ и соотношениями

$$E_k^{(n)} = \frac{1}{1-q}\Big(R(h) + hA^{(n)}K\sum_{j=0}^{k-1} E_j^{(n)}\Big). \tag{38}$$

Легко проверяется, что $\left|\varepsilon_k^{(n)}\right| \le E_k^{(n)}$. Для этого воспользуемся индукцией. При $k = 0$ неравенство, очевидно, выполнено. Пусть $\left|\varepsilon_j^{(n)}\right| \le E_j^{(n)}$ для $j = 0, 1, \ldots, k-1$. Тогда

$$\left|\varepsilon_k^{(n)}\right| \le \frac{1}{1-q}\Big(R(h) + hA^{(n)}K\sum_{j=0}^{k-1} E_j^{(n)}\Big) = E_k^{(n)}.$$

Для упрощения выкладок введем обозначения:

$$\frac{1}{1-q}\,R(h)=\alpha,\qquad \frac{1}{1-q}\,hAK=\beta.$$

Соотношение (38) примет вид

$$E_k^{(n)}=\alpha+\beta\sum_{j=0}^{k-1}E_j^{(n)}.$$

Убедимся в том, что набор величин $E_1^{(n)},E_2^{(n)},\dots,E_k^{(n)}$, связанных рекуррентным соотношением

$$E_j^{(n)}=\bigl(\alpha+\beta E_0^{(n)}\bigr)(1+\beta)^{k-1},\quad j=1,2,\dots\ ,$$

является решением этого уравнения. Для этого подставим в правую часть (38) значения $E_j^{(n)}$. Получим

$$\begin{aligned}\alpha+\beta\sum_{j=0}^{k-1}E_j^{(n)} \;&=\; \alpha+\beta E_0^{(n)}+\beta\bigl(\alpha+\beta E_0^{(n)}\bigr)\sum_{j=1}^{k-1}(1+\beta)^{j-1}=\\ &=\;\bigl(\alpha+\beta E_0^{(n)}\bigr)\bigl(1+\beta\beta^{-1}[(1+\beta)^{k-1}-1]\bigr)=\\ &=\;\bigl(\alpha+\beta E_0^{(n)}\bigr)(1+\beta)^{k-1}=E_k^{(n)}.\end{aligned}$$

Тем самым можно утверждать, что для погрешности $\varepsilon_k^{(n)}$ при всех значениях $k=1,2,\dots$ верна оценка

$$\begin{aligned}|\varepsilon_k^{(n)}|\;&\le\bigl(\alpha+\beta|\varepsilon_0^{(n)}|\bigr)(1+\beta)^{k-1}=\\ &=\frac{1}{1-q}\left(R(h)+hAK|\varepsilon_k^{(n)}|\right)\left(1+\frac{1}{1-q}hAK\right)^{k-1}.\end{aligned}$$

Получим теперь равномерную относительно k оценку погрешности. Воспользуемся неравенствами $k\le n\le(b-a)/h$ и тем фактом, что при $t\to0^+$ величина $(1+t)^{1/t}$ возрастает и стремится к e. Получаем цепочку неравенств

$$\left(1+\frac{1}{1-q}\,hAK\right)^{k-1}\le\left(1+\frac{1}{1-q}\,hAK\right)^{(b-a)/h}=$$

$$=\left[\left(1+\frac{1}{1-q}\,hAK\right)^{(1-q)/hAK}\right]^{(b-a)AK/(1-q)}<e^{(b-a)AK/(1-q)},$$

откуда получаем равномерную относительно k оценку погрешности. Теорема доказана.

Полученная оценка погрешности позволяет высказать заключение о сходимости приближенного решения $U_k^{(n)}$ к точному решению $u(x)$ для $x \in [a, b]$. Предположим, что число узлов сетки неограниченно возрастает, а шаг сетки по-прежнему определяется соотношением $h = (b-a)/n$. Если окажется, что для выбранной квадратурной формулы верхняя граница $R(h)$ погрешности стремится к нулю, то и правая часть неравенства (37) также стремится к нулю. Поэтому можно утверждать, что будет иметь место равномерная сходимость приближенного решения к точному, т.е. для любого $\varepsilon > 0$ существует такое значение h_0, что для всех $h < h_0$ выполняются неравенства

$$\left|U_k^{(n)} - u(s_k)\right| < \varepsilon, \quad k = 0, 1, \ldots, n.$$

3.3. Построение приближенного решения в виде непрерывной функции

По аналогии с интегральным уравнением Фредгольма второго рода для каждого n, при котором может быть построена система (35), определим приближенное решение $u^{(n)}(x)$ уравнения (32).

Для точного решения u уравнения (32) выполнено

$$u(x) = f(x) + \int\limits_a^{s_k} K(x, s)u(s)\,ds + \int\limits_{s_k}^{x} K(x, s)u(s)\,ds, \tag{39}$$

где $s_k \le x \le s_{k+1}$. Для приближенного вычисления первого интеграла в (39) воспользуемся квадратурной формулой вида (34), где вместо значений $u(s_j)$ возьмем величины $U_j^{(n)}$, $j = 1, \ldots, k$. Для приближенного вычисления второго интеграла воспользуемся формулой трапеций:

$$\int\limits_{s_k}^{x} K(x, s)u(s)\,ds \approx \frac{x - s_k}{2}\big(K(x, s_k)u(s_k) + K(x, x)u(x)\big).$$

При этом суммарная ошибка при вычислении интегралов будет величиной порядка $O\big(R(h)+h^3\big)$.

Определим приближенное решение $u^{(n)}(x)$ уравнения (32) следующим образом. Если $x = s_k$ при некотором $k = 0, 1, \ldots, n$, то положим $u^{(n)}(s_k) = U_k^{(n)}$, где величины $U_k^{(n)}$ определены в (35). Если $s_k < x < s_{k+1}$ при некотором $k = 0, 1, \ldots, n-1$, то потребуем, чтобы $u^{(n)}(x)$ в каждой точке x удовлетворяла равенству

$$u^{(n)}(x) = f(x) + h\sum_{j=0}^{k} A_{kj}^{(n)} K(x, s_j) U_j^{(n)} + \\ + \frac{x - s_k}{2}\Big(K(x, s_k)U_k^{(n)} + K(x, x)u^{(n)}(x)\Big).$$

Мы всегда можем считать величину h настолько малой, что

$$1 - \frac{x - s_k}{2} K(x, x) \neq 0.$$

Окончательно имеем

$$u^{(n)}(x) = \frac{f(x) + h\sum\limits_{j=0}^{k} A_{kj}^{(n)} K(x, s_j) U_j^{(n)} + \dfrac{x - s_k}{2} K(x, s_k) U_k^{(n)}}{1 - \dfrac{x - s_k}{2} K(x, x)}.$$

3.4. Способы построения квадратурных формул для решения уравнений Вольтерра второго рода

Перейдем к построению семейства составных квадратурных формул (34) и оценке погрешности при построении приближенного решения.

Наиболее простым является выбор составной квадратурной формулы трапеций:

$$A_{k0}^{(n)} = A_{kk}^{(n)} = \frac{1}{2}, \quad A_{kj}^{(n)} = 1, \quad j = 1, \ldots, k-1, \quad h = \frac{b-a}{n-1}.$$

Тогда на каждом отрезке $[s_k, s_{k+1}]$ погрешность замены интеграла квадратурной суммой будет величиной порядка $O\big(h^3\big)$, а глобальная погрешность на отрезке $[a, s_k]$ — величиной порядка $O\big(h^2\big)$.

При выполненном ранее построении приближенного решения $u^{(n)}(x)$ для вычисления интеграла по отрезку $[s_k, x]$ мы использовали формулу трапеций с погрешностью $O\big(h^3\big)$. Поэтому возникает вопрос о возможности построения семейства квадратурных формул вида (34) с глобальной погрешностью $O\big(h^3\big)$.

Рассмотрим задачу построения при каждом натуральном n семейства квадратур вида

$$\int\limits_0^{kh} F(s)\,ds \approx \sum_{j=0}^{k} A_{kj}^{(n)} F(jh), \quad k = 1, \dots, n.$$

Пусть $k = 1$. Тогда мы имеем три возможности для выбора квадратурной формулы:

$$\int\limits_0^{h} F(s)\,ds \approx hF(0), \quad \int\limits_0^{h} F(s)\,ds \approx hF(h),$$

$$\int\limits_0^{h} F(s)\,ds \approx \frac{h}{2}\,\big(F(0) + F(h)\big).$$

Первые две формулы имеют погрешность $O\big(h^2\big)$, а третья формула — $O\big(h^3\big)$. Поэтому первые две формулы мы в дальнейшем рассматривать не будем.

Для случая $k = 2$ и достаточно гладкой подынтегральной функции рассмотрим три варианта.

Формула средних прямоугольников:

$$\int\limits_0^{2h} F(s)\,ds = 2hF(h) + O\big(h^3\big).$$

Составная формула трапеций:

$$\int\limits_0^{2h} F(s)\,ds = \frac{h}{2}F(0) + hF(h) + \frac{h}{2}F(2h) + O\big(h^3\big).$$

Формула Симпсона:

$$\int\limits_0^{2h} F(s)\,ds = \frac{h}{3}F(0) + \frac{4h}{3}F(h) + \frac{h}{3}F(2h) + O\big(h^5\big).$$

Остановимся на последней формуле и положим

$$A_{20}^{(n)} = A_{22}^{(n)} = \frac{1}{3}, \quad A_{21}^{(n)} = \frac{4}{3}.$$

Для $k = 3$ количество вариантов увеличивается. В частности, можно использовать правило “трех восьмых”

$$\int\limits_0^{3h} F(s)\,ds = \frac{3h}{8}\,\big(F(0) + 3F(h) + 3F(2h) + F(3h)\big) + O\big(h^5\big)$$

или два варианта комбинации формулы Симпсона и формулы трапеций:

$$\int\limits_0^{3h} F(s)\,ds = \frac{h}{3}\big(F(0) + 4F(h) + F(2h)\big) + \frac{h}{2}\big(F(2h) + F(3h)\big) + O\big(h^3\big),$$

$$\int\limits_0^{3h} F(s)\,ds = \frac{h}{2}\big(F(0) + F(h)\big) + \frac{h}{3}\big(F(h) + 4F(2h) + F(3h)\big) + O\big(h^3\big).$$

Отметим, что для произвольных значений k использование в (34) формулы Ньютона–Котеса с $(k+1)$ узлами неоправданно. Можно показать, что в этом случае алгоритм становится неустойчивым.

На практике в рассматриваемой задаче наиболее сложная из используемых квадратур — это правило “трех восьмых”. При большем количестве узлов используются составные квадратурные формулы.

Нами уже рассмотрен подход с использованием составной формулы трапеций. Рассмотрим другие возможности.

Комбинация составной формулы Симпсона и формулы трапеций (первый способ). Будем использовать квадратурные формулы

$$\int\limits_0^{2kh} F(s)\,ds \approx \frac{2h}{3}\sum_{j=0}^{k} F(2jh) + \frac{4h}{3}\sum_{j=0}^{k-1} F(2jh+h)$$

и

$$\int\limits_0^{(2k+1)h} F(s)\,ds \approx \frac{h}{2}\big(F(0)+F(h)\big) + \frac{2h}{3}\sum_{j=0}^{k} F(2jh+h) + \\ + \frac{4h}{3}\sum_{j=1}^{k} F(2jh). \tag{40}$$

Погрешности этих формул составляют $O\big(h^4\big)$ и $O\big(h^3\big)$ соответственно. Значения коэффициентов $A_{kj}^{(n)}$ приведены в следующей таблице.

$k\backslash j$	0	1	2	3	4	5	...
1	1/2	1/2					
2	1/3	4/3	1/3				
3	1/2	5/6	4/3	1/3			
4	1/3	4/3	2/3	4/3	1/3		
5	1/2	5/6	4/3	2/3	4/3	1/3	
...	...	...	...	...	...	...	

Комбинация составной формулы Симпсона и формулы трапеций (второй способ). Вместо (40) можно использовать квадратурную формулу

$$\int\limits_0^{(2k+1)h} F(s)\,ds = \frac{2h}{3}\sum_{j=0}^{k} F(2jh) + \frac{4h}{3}\sum_{j=0}^{k-1} F(2jh+h) + \\ + \frac{h}{2}\big(F(2kh)+F(2kh+h)\big) + O\big(h^3\big).$$

Коэффициенты полученной формулы приведены в следующей таблице

$k\backslash j$	0	1	2	3	4	5	...
1	1/2	1/2					
2	1/3	4/3	1/3				
3	1/3	4/3	5/6	1/2			
4	1/3	4/3	2/3	4/3	1/3		
5	1/3	4/3	2/3	4/3	5/6	1/2	
...	...	...	...	...	...	...	

Во всех приведенных выше вычислительных схемах применение формулы трапеций понижало глобальную погрешность до $O(h^2)$. Этого можно избежать, если использовать правило "трех восьмых", которая имеет локальную погрешность $O(h^5)$ (такую же, как формула Симпсона).

Комбинация формулы Симпсона и правила "трех восьмых" (*первый способ*). Коэффициенты представлены в следующей таблице:

$k\backslash j$	0	1	2	3	4	5	6	...
1	1/2	1/2						
2	1/3	4/3	1/3					
3	3/8	9/8	9/8	3/8				
4	1/3	4/3	2/3	4/3	1/3			
5	3/8	9/8	9/8	17/24	4/3	1/3		
6	1/3	4/3	2/3	4/3	2/3	4/3	1/3	
...	...	...	...	...	...	...	...	

Комбинация формулы Симпсона и правила "трех восьмых" (*второй способ*). Здесь правило "трех восьмых" используется не в начале, а в конце каждого интервала с соответствующим количеством точек разбиения:

$k\backslash j$	0	1	2	3	4	5	6	...
1	1/2	1/2						
2	1/3	4/3	1/3					
3	3/8	9/8	9/8	3/8				
4	1/3	4/3	2/3	4/3	1/3			
5	1/3	4/3	17/24	9/8	9/8	3/8		
6	1/3	4/3	2/3	4/3	2/3	4/3	1/3	
...	...	...	...	...	...	...	...	

3.5. Практический алгоритм построения приближенного решения с заданной точностью

Пусть имеется последовательность натуральных чисел n_k, для каждого из которых можно построить составную квадратурную формулу вида (34). Тогда для каждого $n = n_k$ мы можем построить описанным выше способом приближенное решение $u^{(n)}$ уравнения (32). Наиболее естественный способ построения последовательности приближенных решений — это удвоение числа узлов в формулах (34).

Процесс продолжается до тех пор, пока для заданной точности $\varepsilon > 0$ не будет выполнена оценка

$$\left(\int\limits_a^b \left(u^{(n_k)}(x) - u^{(n_{k-1})}(x)\right)^2 dx\right)^{1/2} \leq \varepsilon. \tag{41}$$

В этом неравенстве интеграл может быть вычислен одним из адаптивных алгоритмов, рассмотренных в п. 1.8.

Варианты задания определяются способами построения квадратурной формулы (34):

— составная формула трапеций;

— комбинация формулы трапеций и составной формулы Симпсона;

— комбинация формул трапеций, Симпсона и составной формулы “трех восьмых” и выбор алгоритма для вычисления интеграла в (41).

4. ВАРИАНТЫ ЗАДАНИЙ. СОДЕРЖАНИЕ ОТЧЕТА

Целью практикума является численное решение интегрального уравнения одного из двух основных типов — Вольтерра и Фредгольма.

Независимо от варианта, выполнение задания состоит из следующих этапов.

Этап 1. Составление и решение аппроксимирующей системы линейных алгебраических уравнений на основе составной квадратурной формулы.

Этап 2. Построение последовательности непрерывных функций $u^n(x)$, сходящихся к точному решению исходного интегрального уравнения. Поиск в этой последовательности такого элемента $u^n(x)$, начиная с которого выполняются неравенства

$$\left\| u^n(x) - u^{n-1}(x) \right\| \leq \varepsilon,$$

где ε — заранее заданная точность.

Этап 3. Сравнение приближенного решения с точным решением тестовой задачи в L_2- и C-нормах. Графическое представление точного и приближенного решений.

Всего можно выделить четыре базовых варианта для выполнения задания практикума:

Ф — метод квадратур для решения интегрального уравнения Фредгольма второго рода (п. 2.1.2);

Ф1 — первый алгоритм численной реализации метода последовательных приближений (п. 2.2.1);

Ф2 — второй алгоритм численной реализации метода последовательных приближений (п. 2.2.2);

В — метод квадратур для решения интегрального уравнения Вольтерра второго рода (п. 3.3).

На первом этапе для вариантов Ф, Ф1 и Ф2 в качестве составной квадратурной формулы при построении аппроксимирующей системы линейных уравнений рекомендуется использовать следующие варианты:

— составная формула средних прямоугольников (п. 1.3);

— составная формула трапеций (п. 1.3);

— составная формула Симпсона (п. 1.3);

— составная формула трех восьмых (п. 1.4);

— составная формула Боде (п. 1.4).

Для варианта В на первом этапе при построении аппроксимирующей системы линейных уравнений в качестве составной квадратурной формулы рекомендуется использовать следующие варианты (п. 3.4):

В1 — комбинация составной формулы Симпсона и формулы трапеций (первый способ);

В2 — комбинация составной формулы Симпсона и формулы трапеций (второй способ);

В3 — комбинация формулы Симпсона и правила трех восьмых (первый способ);

В4 — комбинация формулы Симпсона и правила трех восьмых (второй способ).

На втором этапе для вычисления интегралов

$$\int_a^b \left(u^n(x) - u^{n-1}(x)\right)^2 dx$$

в каждом из четырех основных вариантах применяется один из следующих алгоритмов с автоматическим выбором шага:

А1 — простейший неадаптивный алгоритм (п. 1.8.1);

А2 — модификация простейшего алгоритма (п. 1.8.2);

А3 — адаптивный алгоритм последовательного передвижения “слева-направо” (п. 1.8.3);

А4 – адаптивный алгоритм с контролем точности по глобальной ошибке (п. 1.8.4).

В алгоритмах А1 – А4 используются квадратурные формулы Гаусса с числом узлов от трех до пяти (п. 1.5). В программе должна быть реализована возможность использования любой из этих квадратур.

Варианты заданий практикума могут быть построены по следующей таблице:

В задания	Базовый вариант	Первый этап
1	Ф	ф-ла средних прямоугольн.
2	Ф	ф-ла трапеций
3	Ф	ф-ла Симпсона
4	Ф	ф-ла трех восьмых
5	Ф	ф-ла Боде
6	Ф1	ф-ла Симпсона
7	Ф1	ф-ла трех восьмых
8	Ф1	ф-ла Боде
9	Ф2	ф-ла Симпсона
10	Ф2	ф-ла трех восьмых
11	Ф2	ф-ла Боде
12	В	В1
13	В	В2
14	В	В3
15	В	В4

На втором этапе выполнения практикума каждое из заданий 1 – 15 допускает четыре варианта (А1 – А4) применения алгоритма численного интегрирования с автоматическим распределением узлов. Таким образом, общее количество вариантов есть $15 \times 4 = 60$.

По окончании выполнения задания практикума оформляется отчет о проделанной работе по следующей схеме.

1) Постановка задачи.

2) Описание реализованных алгоритмов.

3) Описание отладочных тестов и анализ их результатов.

4) Графическое представление результатов расчетов.

5. ТЕСТОВЫЕ ЗАДАЧИ ДЛЯ ОТЛАДКИ ПРОГРАММ

Приведем набор тестовых задач для отладки заданий практикума. Уравнения даны вместе с решениями, что позволяет оценить реальную погрешность построенного приближенного решения.

Уравнения Вольтерра второго рода:

1. $u(x) - \int\limits_0^x e^{-(x-t)} u(t)\, dt = e^{-x}$, $u(x) \equiv 1$

2. $u(x) - \int\limits_0^x \big(1 - (x-t)\big) e^{2x} u(t)\, dt = (1 - xe^{2x}) \cos 1 - e^{2x} \sin 1$,
$u(x) = e^x \big(\cos(e^x) - e^x \sin(e^x)\big)$

3. $u(x) - \int\limits_0^x e^{x-s} u(s)\, ds = e^x$, $u(x) = e^{2x}$

4. $u(x) - 2\int\limits_0^x e^{x-s} u(s)\, ds = \sin x$, $u(x) = 0.2e^{3x} - 0.2\cos x + 0.4 \sin x$

5. $u(x) + \int\limits_0^x 3^{x-s} u(s)\, ds = 3^x x$, $u(x) = 3^x(1 - e^{-x})$

6. $u(x) - \int\limits_0^x \dfrac{2 + \cos x}{2 + \cos s} u(s)\, ds = e^x \sin x$,
$u(x) = e^x \sin x + (2 + \cos x) e^x \ln \dfrac{3}{2 + \cos x}$

7. $u(x) + \int\limits_0^x e^{x^2 - s^2} u(s)\, ds = 1 - 2x$, $u(x) = e^{x^2 - x} - 2x$

8. $u(x) - 2\int\limits_0^x e^{x^2 - s^2} u(s)\, ds = e^{x^2 + 2x}$, $u(x) = e^{x^2 + 2x}(1 + 2x)$

9. $u(x) - \int\limits_0^x \frac{1+x^2}{1+s^2} u(s)\,ds = 1 + x^2$, $u(x) = e^x\big(1+x^2\big)$

10. $u(x) - \int\limits_0^x e^{s-x} u(s)\,ds = \frac{1}{1+x^2}$,

$u(x) = \frac{1}{1+x^2} + x \operatorname{arctg} x - \frac{1}{2}\ln\big(1+x^2\big)$

Уравнения Фредгольма второго рода:

11. $u(x) - \frac{1}{2}\int\limits_0^1 x e^t u(t)\,dt = e^{-x}$, $u(x) = x + e^{-x}$

12. $u(x) - \int\limits_0^{0.5} \sin xt\, u(t)\,dt = 1 + \frac{1}{x}\left(\cos\frac{x}{2} - 1\right)$, $u(x) \equiv 1$

13. $u(x) + \frac{1}{4\pi}\int\limits_0^{2\pi} \frac{1}{\sin^2\Big(\frac{x+t}{2}\Big) + 0.25\cos^2\Big(\frac{x+t}{2}\Big)}\, u(t)\,dt =$

$= \frac{1}{16\pi}\,(5 + 3\cos 2x)$, $u(x) = \frac{1}{160\pi}\,(25 + 27\cos 2x))$.

14. $u(x) - \lambda\int\limits_0^1 (2x - t)\,u(t)\,dt = \frac{x}{6}$,

$u(x) = \frac{1}{6}\left(x + \frac{(6x-2)\lambda - \lambda^2}{\lambda^2 - 3\lambda + 6}\right)$

15. $u(x) - \int\limits_0^{2\pi} \sin x \cos t\, u(t)\,dt = \cos 2x$, $u(x) = \cos 2x$

16. $u(x) - \lambda\int\limits_0^1 \big(4xt - x^2\big)\,u(t)\,dt = x$, $u(x) = \frac{3x(2\lambda - 3\lambda x + 6)}{\lambda^2 - 18\lambda + 18}$

17. $u(x) - \int\limits_0^1 x t^2\, u(t)\,dt = 1$, $u(x) = 1 + \frac{4}{9}\,x$

18. $u(x) - \frac{1}{2}\int\limits_0^1 xt\,u(t)\,dt = \frac{5}{6}\,x,\; u(x) = x$

19. $u(x) - \int\limits_{-1}^1 x^2 e^{xt}\,u(t)\,dt = 1 - x(e^x - e^{-x}),\; u(x) = 1$

20. $u(x) - \lambda\int\limits_0^\pi \cos^2 x\,u(t)\,dt = 1,\; u(x) = 1 + \frac{2\lambda}{2 - \pi\lambda}\cos^2 x$

Литература

1. *Арушанян И.О., Чижонков Е.В.* Материалы семинарских занятий по курсу "Методы вычислений" / под ред. Арушаняна О.Б. — М.: Изд-во ЦПИ при механико-математическом факультете МГУ, 1999.

2. *Бахвалов Н.С.* Численные методы. М.: Наука, 1975.

3. *Бахвалов Н.С., Жидков Н.П., Кобельков Г.М.* Численные методы. М.: Наука, 1987.

4. *Верлань А.Ф., Сизиков В.С.* Интегральные уравнения: методы, алгоритмы, программы. Справочное пособие. Киев: Наукова думка, 1968.

5. *Калиткин Н.Н.* Численные методы. М.: Наука, 1978.

6. *Каханер Д., Моулер К., Нэш С.* Численные методы и программное обеспечение. М.: Мир, 1998.

7. *Краснов М.Л., Киселев А.И., Макаренко Г.И.* Интегральные уравнения. М.: Наука, 1968.

8. *Крылов В.И., Бобков В.В., Монастырный П.И.* Вычислительные методы. М.: Наука, 1977.