Численное интегрирование функций

К методам приближенного и численного интегрирования функций приходится прибегать в случаях, когда

1. подынтегральная функция f(x) задана таблично на участке [a,b];
2. подынтегральная функция f(x) задана аналитически, но ее первообразная не выражается через элементарные функции;
3. подынтегральная функция f(x) задана аналитически, имеет первообразную, но ее определение слишком сложно.

Методы приближенного интегрирования используют разложение подынтегральных функций в ряды Тейлора (Маклорена) и дальнейшего почленного интегрирования членов ряда.

Определение. Функция $F(x)$ называется **первообразной** для функции $f(x)$ на заданном промежутке, если для всех x из этого промежутка выполняется равенство $F'(x) = f(x)$.

Нахождение первообразной – интегрирование – операция, обратная операции дифференцирования.

К недостаткам методов приближенного интегрирования относится требование дифференцируемости подынтегральных функций до порядка, который требуется при разложении функций в ряд Тейлора. От этого недостатка свободны **методы численного интегрирования**, в которых подынтегральная функция удовлетворяет только условию непрерывности (для существования определённого интеграла).

В численных методах интегрирования не используется нахождение первообразной. Основу алгоритма численных методов интегрирования составляет геометрический смысл определенного интеграла. Интеграл численно равен площади S криволинейной трапеции, расположенной под подынтегральной кривой f(x) на участке [a,b] (рис.10.1).

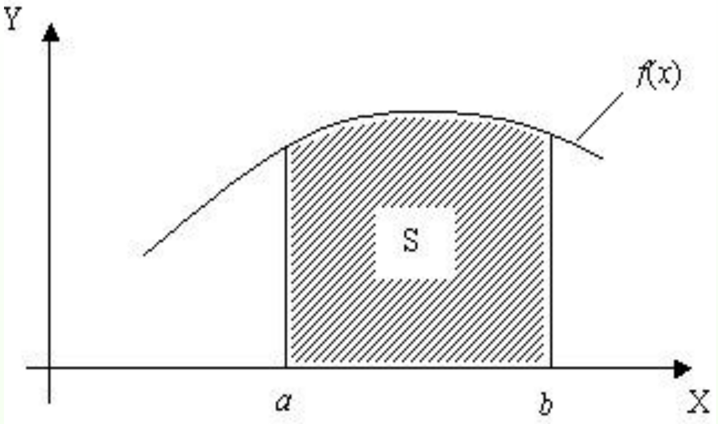


Рис. 10.1. Геометрический смысл определенного интеграла

Суть всех численных методов интегрирования состоит в приближенном вычислении указанной площади. Поэтому все численные методы являются приближенными.

При вычислении интеграла подынтегральная функция f(x) аппроксимируется интерполяционным многочленом

$\phi(x) = \sum_{i=1}^{n} c_i \varphi_i(x)$, то есть $I = \int_a^b f(x)dx = \int_a^b \varphi(x)dx + R$, где $R = \int_a^b r(x)dx$

-**априорная погрешность метода** или **остаточный член** на интервале интегрирования, а *r(x)* – априорная погрешность метода на отдельном шаге интегрирования.

На практике, чтобы не иметь дело с многочленами высоких степеней, весь участок [a,b] делят на части и интерполяционные многочлены строят для каждой части деления.

<u>Обзор методов интегрирования</u>

Методы вычисления однократных интегралов называются **квадратурными**, а формулы для приближенного вычисления интегралов - квадратурными формулами или квадратурными суммами. (для кратных интегралов – **кубатурными**).

1. **Методы Ньютона-Котеса.** Здесь *φ(x)* – полином различных степеней. Сюда относятся метод прямоугольников, трапеций, Симпсона.

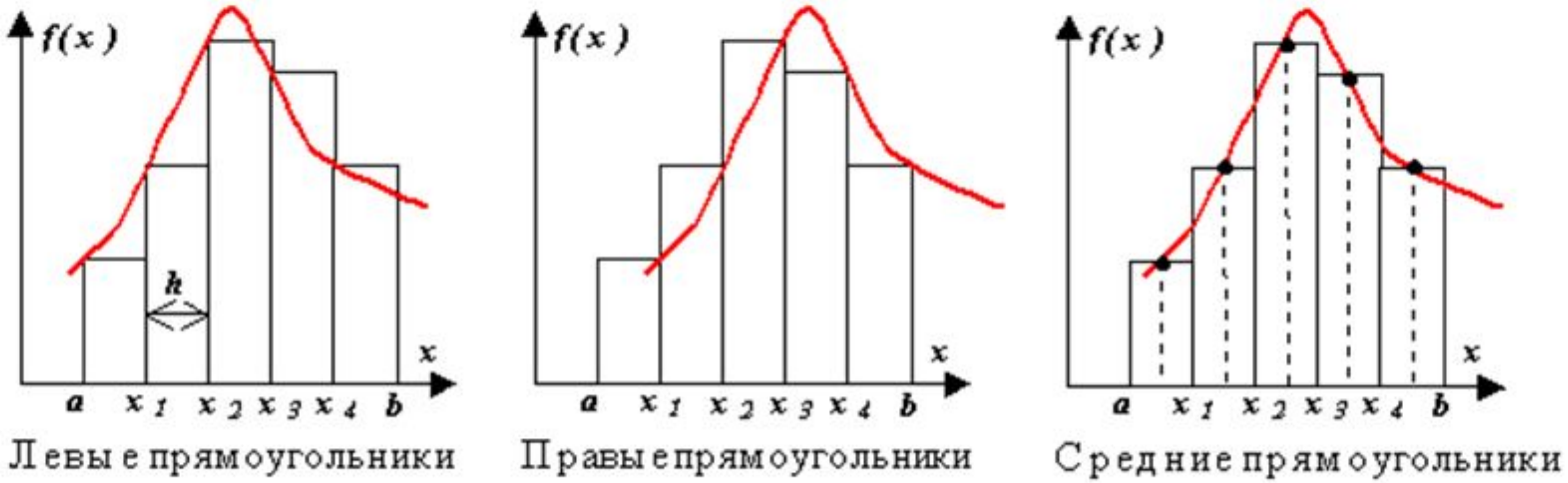
2. **Методы статистических испытаний (методы Монте-Карло).** Здесь узлы сетки для квадратурного или кубатурного интегрирования выбираются с помощью датчика случайных чисел, ответ носит вероятностный характер. В основном применяются для вычисления кратных интегралов.
3. **Сплайновые методы.** Здесь *φ(x)* – кусочный полином с условиями связи между отдельными полиномами посредством системы коэффициентов.
4. **Методы наивысшей алгебраической точности.** Обеспечивают оптимальную расстановку узлов сетки интегрирования и выбор весовых коэффициентов *ρ(x)* в задаче

$$\int_a^b \varphi(x)\rho(x)dx$$

Сюда относятся метод Гаусса-Кристоффеля (вычисление несобственных интегралов) и метод Маркова.

Метод прямоугольников

Различают метод левых, правых и средних прямоугольников. Суть метода ясна из рисунка. На каждом шаге интегрирования функция аппроксимируется полиномом **нулевой** степени – отрезком, параллельным оси абсцисс.



Выведем формулу метода прямоугольников из анализа разложения функции *f(x)* в ряд Тейлора вблизи некоторой точки $x = x_i$.

$$f(x)\big|_{x=x_i} = f(x_i) + (x - x_i)f'(x_i) + \frac{(x - x_i)^2}{2!}f''(x_i) + \ldots$$

Рассмотрим диапазон интегрирования от x_i до x_i+h, где h – шаг интегрирования.

Вычислим $$\int_{x_i}^{x_i+h} f(x)dx = x\cdot f(x_i)\Big|_{x_i}^{x_i+h} + \frac{(x-x_i)^2}{2} f'(x_i)\Big|_{x_i}^{x_i+h} + \frac{(x-x_i)^3}{3\cdot 2!} f''(x_i)\Big|_{x_i}^{x_i+h} + \ldots =$$

$$= f(x_i)h + \frac{h^2}{2} f'(x_i) + O(h^3) = \boxed{f(x_i)h + r_i}$$. Получили формулу *правых (или левых) прямоугольников* и априорную оценку погрешности r на отдельном шаге интегрирования. Основной критерий, по которому судят о точности алгоритма – степень при величине шага в формуле априорной оценки погрешности.

В случае равного шага h на всем диапазоне интегрирования общая формула имеет вид

$$\int_a^b f(x)dx = h\sum_{i=1}^{n} f(x_i) + \sum_{i=1}^{n} r_i$$

Здесь n – число разбиений интервала интегрирования,

$$R = \sum_{i=1}^{n} r_i = \frac{h^2}{2}\sum_{i=1}^{n} f'(x_i) = \frac{h^2}{2}\int_a^b f'(x)dx$$. Для справедливости существования этой оценки необходимо существование непрерывной $f'(x)$.

На всём отрезке $[a,b]$ погрешность r_i необходимо

просуммировать n раз, $n = (b-a/h)$, получим: $R = \frac{nh^2}{2} f'(x_i)$

$$R = \frac{(b-a)h}{2} f'(x_i) \quad , \qquad x_i \epsilon (a,b) \tag{1}$$

Поскольку местоположение точки x_i на интервале $x \in [a, b]$ неизвестно, то на основе погрешности (1) можно выписать верхнюю оценку абсолютной погрешности метода прямоугольников и при заданной точности ε метода выписать неравенства

$$|R_{\text{пр}}| \leqslant \frac{(b-a)h}{2} M_1 \leqslant \varepsilon, \quad M_1 = \max_{x \in [a,b]} \left| f'(x) \right|.$$

Последнее неравенство можно использовать для верхней оценки шага h численного интегрирования по методу прямоугольников:

$$h \leqslant \frac{2\varepsilon}{(b-a)\, M_1}, \quad M_1 = \max_{x \in [a,b]} \left| f'(x) \right|.$$

Для табличной функции

x_i	x_0	x_1	$\ldots$	x_n
y_i	y_0	y_1	$\ldots$	y_n

формула прямоугольников на отрезке *x є [a,b]*

имеет вид

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} y_{i-1}h_i. \qquad (2)$$

Ее погрешность определяется выражением (1).

Таким образом, определяющими формулами метода прямоугольников являются формула (2) численного интегрирования и формула (1) погрешности.

Из (1) видно, что на каждом отрезке $[x_{i-1}, x_i]$ формула прямоугольников имеет погрешность, пропорциональную h^2, а на всем отрезке $x \in [a, b]$ — шагу численного интегрирования h. В соответствии с этим *метод прямоугольников является методом первого порядка точности (главный член погрешности пропорционален шагу в первой степени)*.

Алгоритм метода прямоугольников с автоматическим выбором шага:

1. Весь участок *[a,b]* делим на *n* равных частей с шагом

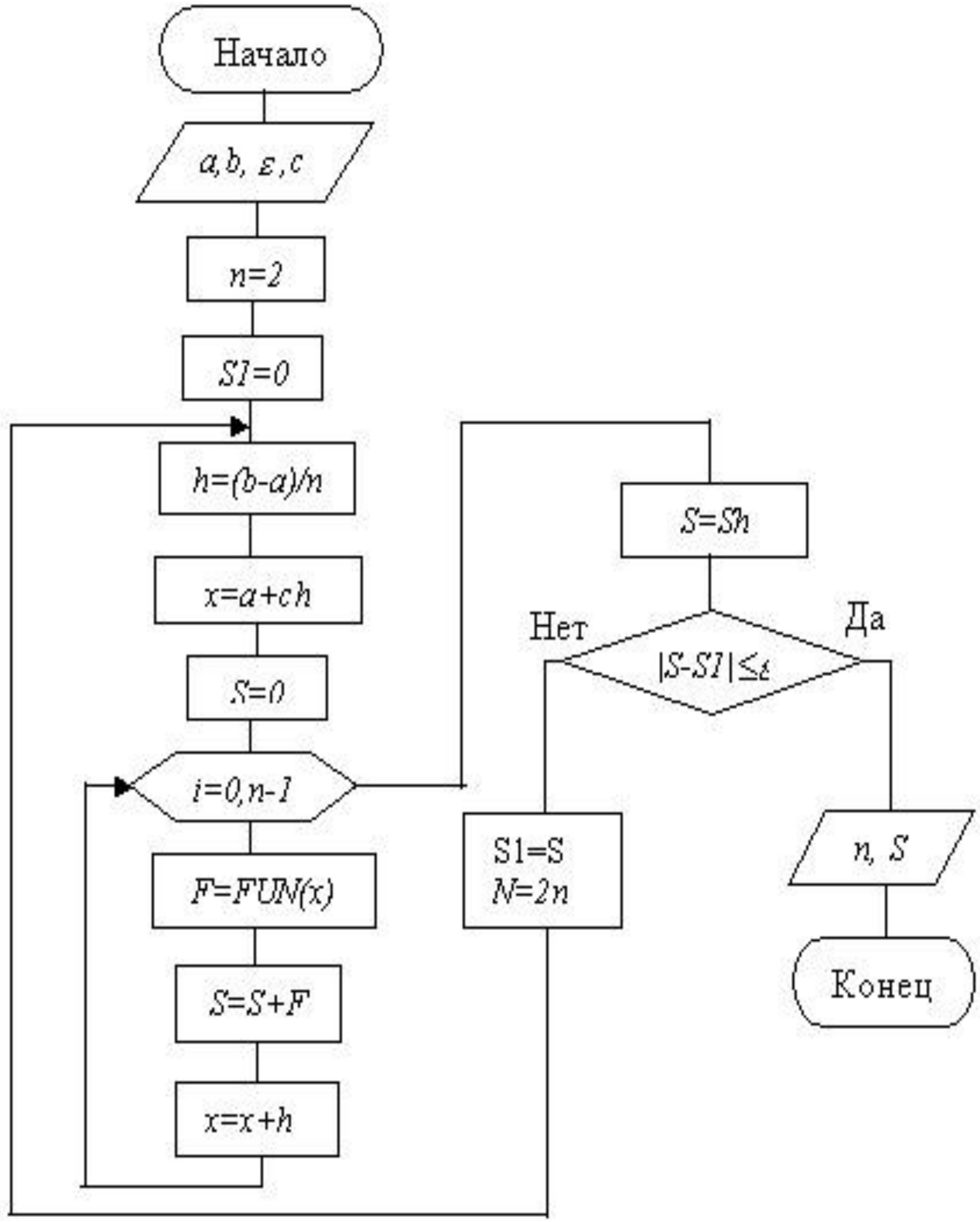
   $h=(b-a)/n$.

1. Определяем значение y_i подынтегральной функции *f(x)* в каждой части деления $y_i = f(x_i), i = \overline{0, n}.$
2. В каждой части деления подынтегральную функцию *f(x)* аппроксимируем интерполяционным многочленом степени *n = 0,* т.е. прямой, параллельной оси *OX*. В результате вся подынтегральная функция на участке *[a,b]* аппроксимируется ломаной линией.
3. Для каждой части деления определяем площадь S_i частичного прямоугольника.
4. Суммируем эти площади. Приближенное значение интеграла *I* равно сумме площадей частичных прямоугольников. $I = \sum_{i=1}^{n-1} S_i.$

Алгоритм вычисления интеграла построим в виде итерационного процесса поиска с автоматическим выбором шага. На каждом шаге будем уменьшать шаг h в два раза, то есть увеличивать число шагов *n* в два раза. Выход из процесса поиска организуем по точности вычисления интеграла. Начальное число шагов *n=2*. Схема алгоритма методов прямоугольников представлена на [рис.](#)

Условные обозначения:

a,b – концы интервала, *ε*- заданная точность, *c=0* - метод левых прямоугольников, *c=1* - метод правых прямоугольников, *S1* - значение интеграла на предыдущем шаге, *S* - значение



Метод средних прямоугольников

Здесь на каждом интервале значение функции считается в точке $\bar{x} = x_i + \frac{h}{2}$, то есть

$$\int_{x_i}^{x_i+h} f(x)dx = hf(\bar{x}) + r_i$$

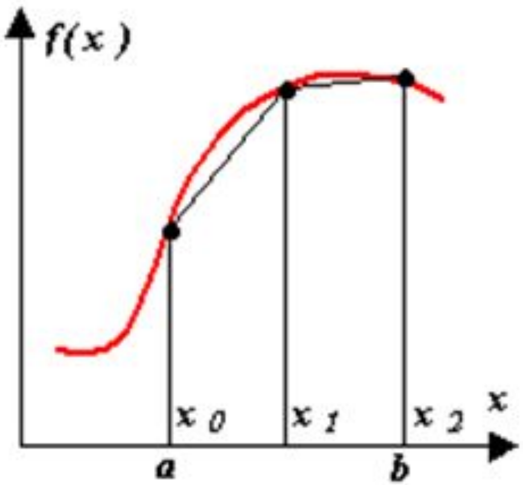
. Разложение функции в ряд Тейлора показывает, что в случае средних прямоугольников точность метода существенно выше:

$$r = \frac{h^3}{24} f''(\bar{x}), R = \frac{h^2}{24}\int_a^b f''(x)dx$$

.

Метод трапеций

Аппроксимация в этом методе осуществляется полиномом **первой** степени. Суть метода ясна из рисунка.



Метод трапеций

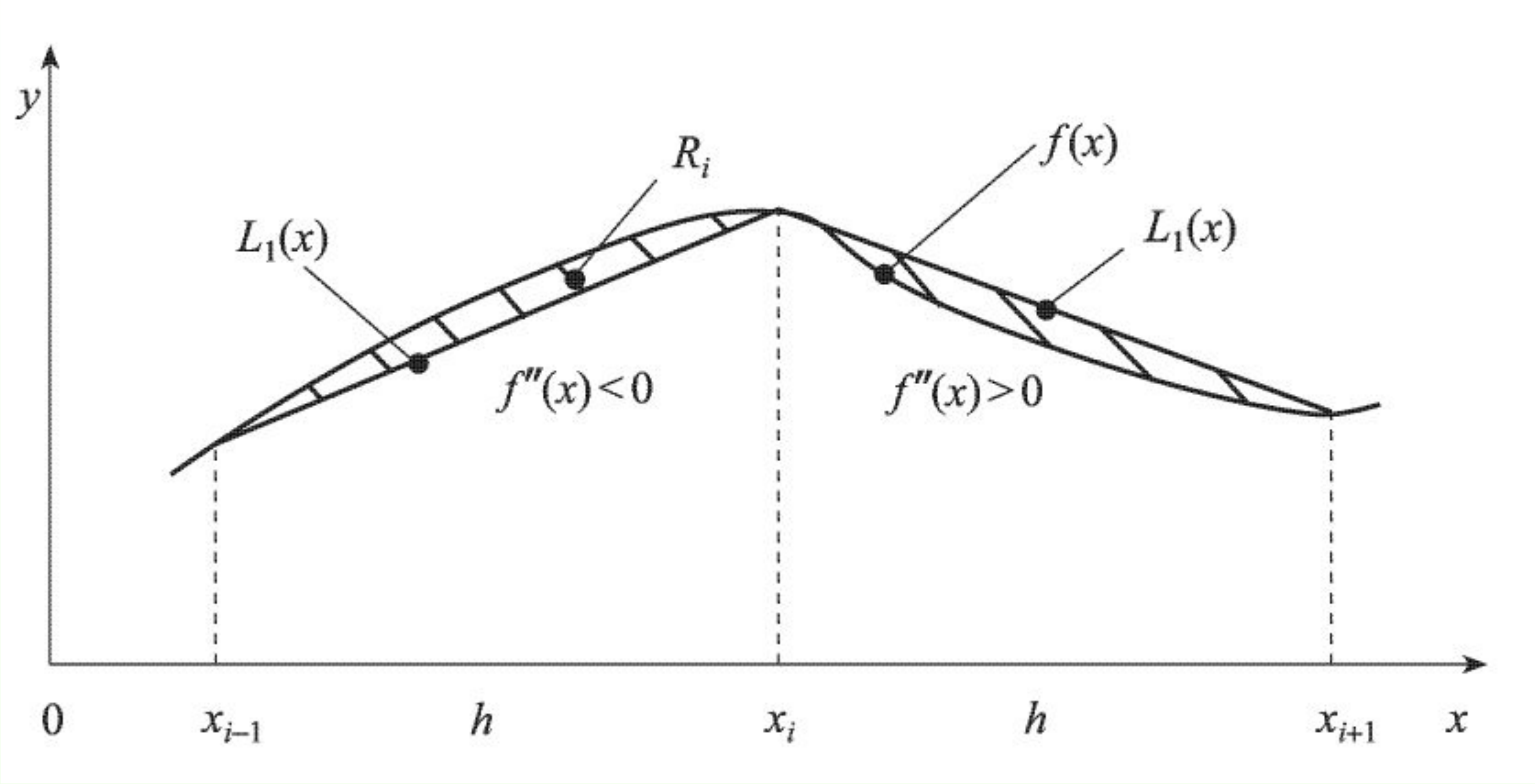
На единичном интервале

$$\int_{x_i}^{x_i+h} f(x)dx = \frac{h}{2}\left(f(x_i) + f(x_i+h)\right) + r_i$$

В случае равномерной сетки ($h = \text{const}$)

$$\int_a^b f(x)dx = h\left(\frac{1}{2}f(x_0) + \sum_{i=1}^{n-1} f(x_i) + \frac{1}{2}f(x_n)\right) + R$$

При этом $r_i = -\frac{h^3}{12}f''(x_i)$, а $R = -\frac{h^2}{12}\int_a^b f''(x)dx$.



Из рис. видно, что если $f''(x) < 0$, то $R_i > 0$

если же $f''(x) > 0$, то $R_i < 0$.

Для всего отрезка [a,b] необходимо сложить выражение для одного интервала n раз:

$$\int_a^b f(x)\,dx \approx \frac{h}{2}\left(y_0 + y_n + 2\sum_{i=1}^{n-1} y_i\right). \qquad (3)$$

Выражение (3) *называют формулой трапеций численного интегрирования для всего отрезка* $[a,\,b]$.

На всем отрезке $[a,\,b]$ погрешность r_i необходимо увеличить в n раз:

$$R_{\text{тр}} = n\,r_i = -\frac{nh^3}{12} f''(x_i) = -\frac{(nh)h^2}{12} f''(x_i) =$$

$$= -\frac{b-a}{12} h^2 f''(x_i)\;, \quad x \in (a,b). \qquad (4)$$

Таким образом, метод трапеций — *метод второго порядка точности относительно шага* h (*главный член погрешности пропорционален шагу в квадрате*).

Поскольку положение точки χ_i на интервале (a, b) неизвестно, то, задавая точность ε численного интегрирования, можно записать следующие неравенства, используемые для определения шага h численного интегрирования:

$$|R_{\text{тр}}| \leqslant \max_{x \in [a,b]} \left| f''(x) \right| \ \frac{b-a}{12}\, h^2 \leqslant \varepsilon,$$

откуда

$$h \leqslant \sqrt{\frac{12 \cdot \varepsilon}{(b-a) M_2}}, \quad M_2 = \max_{x \in [a,b]} \left| f''(x) \right|. \qquad (5)$$

Если точность ε не задана, то выбирая шаг h численного интегрирования, можно по формуле (4) оценить погрешность $R_{\text{тр}}$ формулы трапеций.

Алгоритм метода трапеций с автоматическим выбором шага:

1. Интервал [a,b] делим на n равных частей с шагом

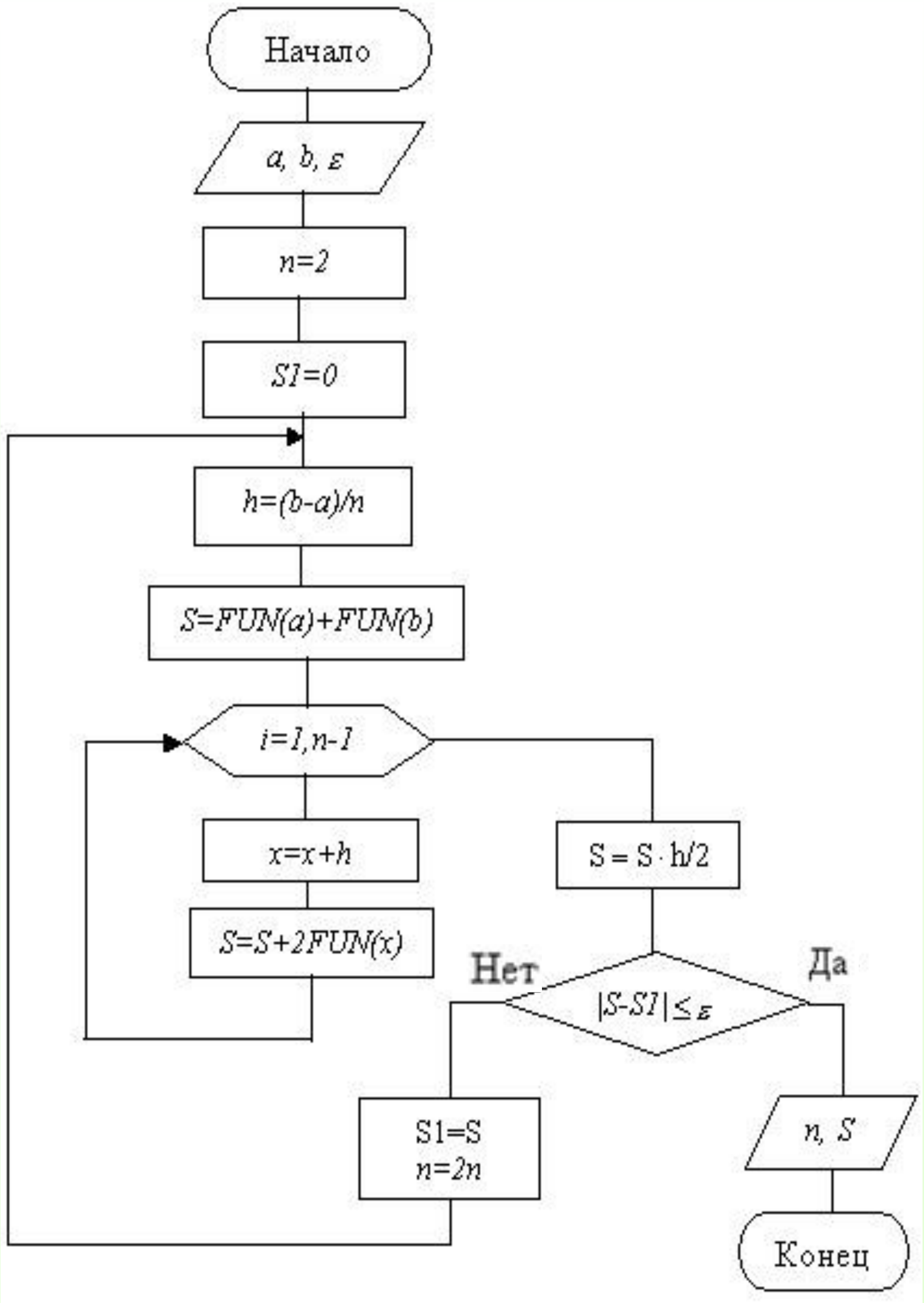
   h=(b-a)/n.

1. Вычисляем значение подынтегральной функции в каждой узловой точке $y_i = f(x_i), i = \overline{0,n}.$
2. На каждом шаге подынтегральную функцию f(x) аппроксимируем прямой, соединяющей две соседние узловые точки. В результате вся подынтегральная функция на участке [a,b] заменяется ломаной линией проходящей через все узловые точки.
3. Вычисляем площадь каждой частичной трапеции.
4. Приближенное значение интеграла равно сумме площадей частичных трапеций.

$$I = \sum_{i=1}^{n-1} S_i.$$

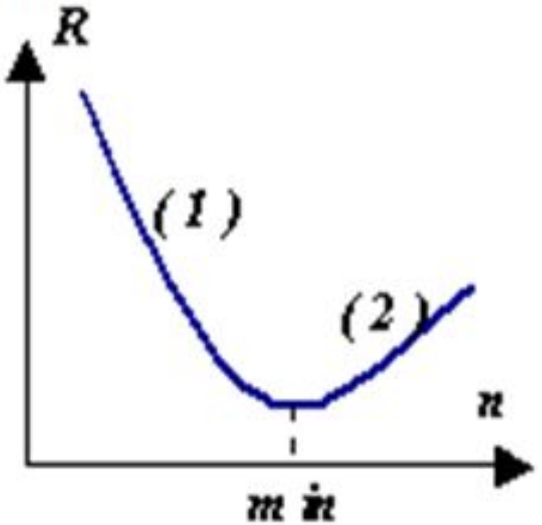
Рис. Схема алгоритма метода трапеций

(с автоматическим выбором шага)



Особенности поведения погрешности

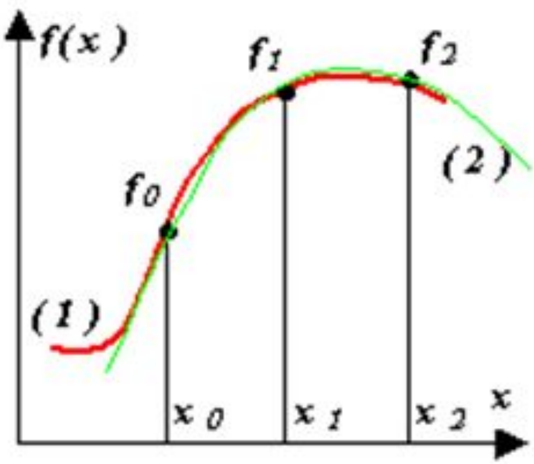
Казалось бы, зачем анализировать разные методы интегрирования, если мы можем достичь высокой точности, просто уменьшая величину шага интегрирования. Однако, рассмотрим график поведения апостериорной погрешности R результатов численного расчета в зависимости от числа n разбиений интервала (то есть при $n \to \infty$ шаг $h \to 0$.



На участке (1) погрешность уменьшается в связи с уменьшением шага h. Но на участке (2) начинает доминировать вычислительная погрешность, накапливающаяся в результате многочисленных арифметических действий. Таким образом, для каждого метода существует своя R_{min}, которая зависит от многих факторов, но прежде всего от априорного значения погрешности метода R.

<u>Метод Симпсона</u>

Подынтегральная функция *f(x)* заменяется интерполяционным полиномом **второй** степени *P(x)* – параболой, проходящей через три узла, например, как показано на рисунке ((1) – функция, (2) – полином).



Метод Симпсона

Рассмотрим два шага интегрирования (h = const = $x_{i+1} - x_i$), то есть три узла x_0, x_1, x_2, через которые проведем параболу, воспользовавшись уравнением Ньютона:

$$P(x) = f_0 + \frac{x - x_0}{h}(f_1 - f_0) + \frac{(x - x_0)(x - x_1)}{2h^2}(f_0 - 2f_1 + f_2)$$

Пусть $z = x - x_0$,
тогда

$$P(z) = f_0 + \frac{z}{h}(f_1 - f_0) + \frac{z(z - h)}{2h^2}(f_0 - 2f_1 + f_2) =$$

$$= f_0 + \frac{z}{2h}(-3f_0 + 4f_1 - f_2) + \frac{z^2}{2h^2}(f_0 - 2f_1 + f_2)$$

Теперь, воспользовавшись полученным соотношением, сосчитаем интеграл по данному интервалу:

$$\int_{x_0}^{x_2} P(x)dx = \int_{0}^{2h} P(z)dz = 2hf_0 + \frac{(2h)^2}{4h}(-3f_0 + 4f_1 - f_2) + \frac{(2h)^3}{6h^2}(f_0 - 2f_1 + f_2) =$$

$$= 2hf_0 + h(-3f_0 + 4f_1 - f_2) + \frac{4h}{3}(f_0 - 2f_1 + f_2) = \frac{h}{3}(6f_0 - 9f_0 + 12f_1 - 3f_2 + 4f_0 - 8f_1 + 4f_2).$$

В итоге

$$\boxed{\int_{x_0}^{x_2} f(x)dx = \frac{h}{3}(f_0 + 4f_1 + f_2) + r} \qquad (6)$$

На всем отрезке $[a, b]$ выражение (6) необходимо сложить m раз, поскольку имеется m пар отрезков длиной h, получим *формулу Симпсона численного интегрирования определенного интеграла*

$$\int_a^b f(x)dx \approx \frac{h}{3}\left(y_0 + y_n + 4\sum_{i=1}^{m} y_{2i-1} + 2\sum_{i=1}^{m-1} y_{2i}\right). \qquad (7)$$

Погрешность формулы Симпсона на одной паре шагов записывается следующим образом:

$$r_i = \int_{x_{i-1}}^{x_{i+1}} f(x)dx - \frac{h}{3}(y_{i-1} + 4y_i + y_{i+1}) =$$

$$= -\frac{h^5}{90} f^{IV}(x_i) \qquad x \in (x_{i-1}, x_{i+1}).$$

Для всего отрезка $[a, b]$ эту погрешность необходимо умножить на m пар отрезков:

$$R_c = m\, r_i = -\frac{mh^5}{90} f^{IV}(x) = -\frac{2mh^5}{180} f^{IV}(x) =$$

$$= -\frac{nh \cdot h^4}{180} f^{IV}(x) = -\frac{(b-a)h^4}{180} f^{IV}(x) \quad x \in (a, b), \qquad (8)$$

т. е. в формуле Симпсона на всем отрезке $[a, b]$ *погрешность пропорциональна четвертой степени шага, и, следовательно, метод Симпсона является методом четвертого порядка точности (т. е. главный член погрешности пропорционален четвертой степени шага* h*).*

Поскольку положение точки на отрезке $[a, b]$ неизвестно, то в соответствии с (8) можно записать верхнюю оценку погрешности и при заданной точности ε получить

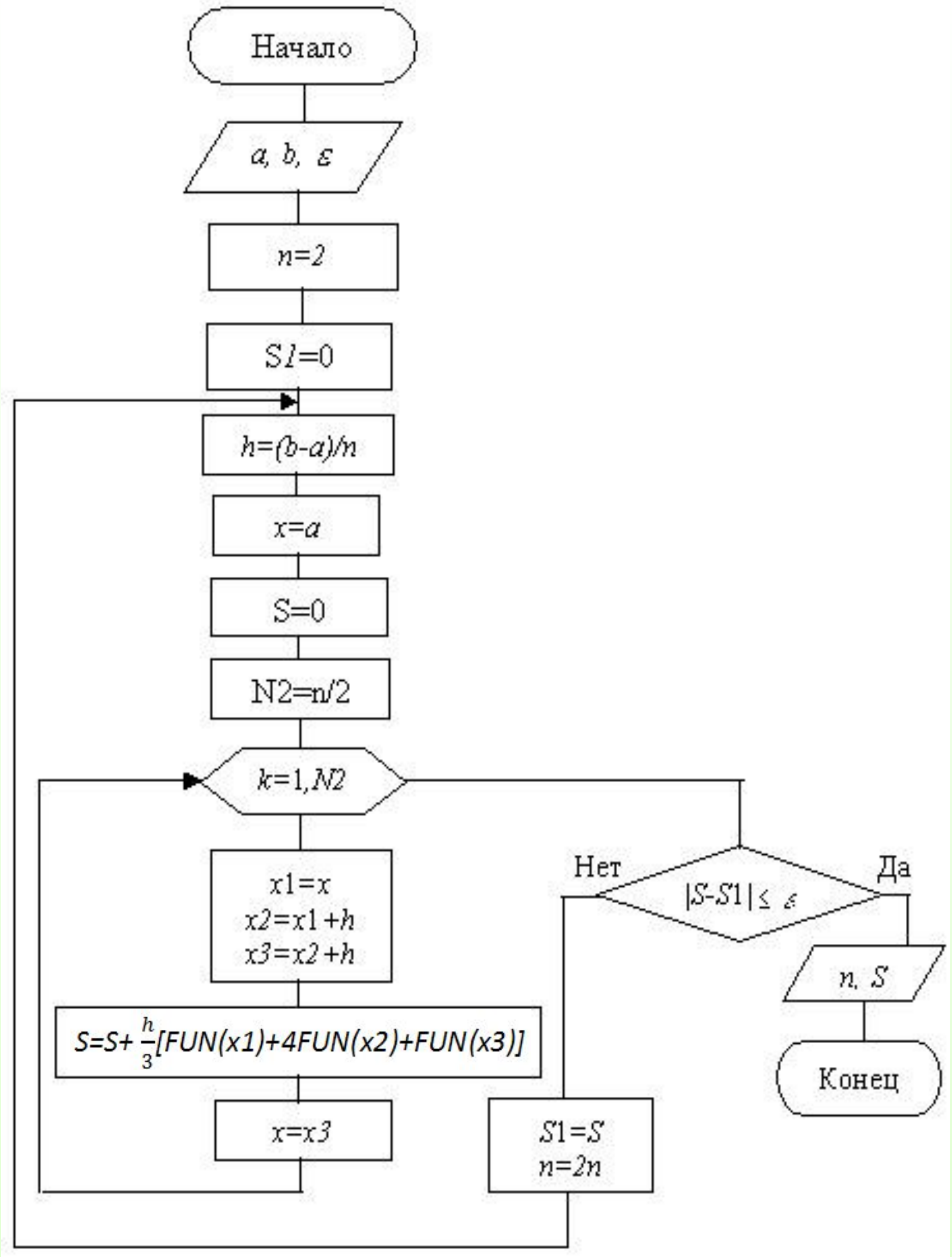
$$|R_c| \leqslant \frac{(b-a)h^4}{180} M_4 \leqslant \varepsilon, \quad M_4 = \max_{x \in [a,b]} \left| f^{IV}(x) \right|,$$

откуда

$$h \leqslant \sqrt[4]{\frac{180\varepsilon}{(b-a)M_4}}, \quad M_4 = \max_{x \in [a,b]} \left| f^{IV}(x) \right|. \qquad (9)$$

Таким образом, определяющими формулами метода Симпсона являются выражения (7) , (8) , (9) , в соответствии с которыми по заданной точности ε из ур.(9) находится шаг h численного интегрирования, с его помощью составляется сеточная функция $y_i = f(x_i)$, $i = \overline{0, n}$, $n = 2m$, а затем приближенно вычисляется интеграл по формуле (7) . Если точность ε неизвестна, то, задаваясь шагом h, можно по формуле (8) вычислить погрешность численного интегрирования.

Рис. Схема алгоритма Симпсона (с автомати-ческим выбором шага), где N2 - количество частей деления.



Процедура Рунге оценки погрешности и уточнения формул численного интегрирования. Процедура Рунге позволяет оценить погрешность и повысить на единицу порядок метода путем многократного (в простейшем случае двукратного) просчета с различными шагами.

Пусть используется какой-либо метод численного интегрирования с шагами h и $h/2$. И пусть порядок выбранного метода равен p, тогда

$$I = I_h + \psi h^p + O(h^{p+1}), \tag{10}$$

$$I = I_{h/2} + \psi \left(\frac{h}{2}\right)^p + O(h^{p+1}), \tag{11}$$

где I — точное значение интеграла; I_h, $I_{h/2}$ — вычисленные значения интеграла с шагом h и $h/2$ соответственно; вторые слагаемые справа — главные члены погрешности метода численного интегрирования порядка p. Для их вычисления вычтем из выражения (11) выражение (10), получим

$$\left(I_{h/2} - I_h\right) + \psi\left(\frac{h}{2}\right)^p [1 - 2^p] + O\left(h^{p+1}\right) = 0,$$

$$\psi\left(\frac{h}{2}\right)^p = \frac{I_{h/2} - I_h}{2^p - 1} \qquad (12)$$

Выражение (12) позволяет провести апостериорную оценку погрешности вычисленного значения определенного интеграла.

Подставим (12) в (11), получим формулу численного интегрирования уже порядка $p+1$:

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

Таким образом, формула (13) — простейшая процедура Рунге уточнения на один порядок формулы численного интегрирования.

Замечание. Если для подынтегральной функции $y = f(x)$ построена сеточная функция $y_i = f(x_i)$ с переменным шагом h_i, то погрешность численного интегрирования определяется как интеграл от погрешности интерполяционного многочлена.

Использование полиномов высоких степеней в квадратурных формулах Ньютона-Котеса сопряжено со значительными вычислительными трудностями. Поэтому на практике поступают так: разбивают промежуток интегрирования на достаточно большое число маленьких отрезков и к каждому из них применяют квадратурную формулу Ньютона-Котеса с небольшим числом ординат. В результате получаются не сложные формулы и расчёты по ним дают достаточно высокую точность. Известны также квадратурные формулы Чебышёва и Гаусса.

Ошибка в выборе величины шага интегрирования либо не обеспечит нужной точности, либо приведёт к необоснованным затратам машинного времени. Можно использовать такую стратегию: сначала задаются довольно большим шагом интегрирования, затем последовательно его дробят до тех пор, пока различие между двумя последующими значениями интегралов, рассчитанными при различном шаге, не станет незначительным.

Пример Методом трапеций с точностью $\varepsilon = 10^{-2}$ и Симпсона с точностью $\varepsilon_1 = 10^{-4}$ вычислить определенный интеграл (вычисляемый точно):

$$\int\limits_0^1 \frac{dx}{1+x} = \ln|1+x| \Big|_0^1 = \ln 2 = 0{,}69315.$$

Решение. 1) *Метод трапеций.* Исходя из заданной точности $\varepsilon = 10^{-2}$ вычислим шаг численного интегрирования, для чего используется формула (5) :

$$h \leqslant \sqrt{\frac{12\varepsilon}{(b-a)\,M_2}}, \quad M_2 = \max_{x\in[0;1]} \left|f''(x)\right| = \max_{x\in[0;1]} \left|\frac{2}{(1+x)^3}\right| = 2;$$

$$h \leqslant \sqrt{\frac{12 \cdot 0{,}01}{(1-0)\cdot 2}} = \sqrt{6} \cdot 0{,}1 = 0{,}2449.$$

Необходимо выбрать такой шаг, который удовлетворяет неравенству $h \leqslant 0{,}2449$, и чтобы на отрезке интегрирования $x \in [0;\,1]$ он укладывался целое число раз. Принимаем шаг $h = 0{,}2$. Он удовлетворяет обоим этим требованиям.

Для подынтегральной функции $f\,(x) = (1 + x)^{-1}$ с независимой переменной x_i, изменяющейся в соответствии с равенством $x_i = x_0 + ih = 0 + i \cdot 0{,}2$, $i = \overline{0,\,5}$, составляем сеточную функцию с точностью до второго знака после запятой:

i	0	1	2	3	4	5
x_i	0	0,2	0,4	0,6	0,8	1,0
y_i	1,0	0,83	0,71	0,63	0,56	0,5

Используется формула трапеций (3) численного интегрирования ($n = 5$):

$$\int\limits_0^1 \frac{dx}{1+x} \approx \frac{h}{2}\left(y_0 + y_n + 2\sum_{i=1}^{n-1} y_i\right) =$$

$$= \frac{0{,}2}{2}\left[1{,}0 + 0{,}5 + 2\,(0{,}83 + 0{,}71 + 0{,}63 + 0{,}56)\right] = 0{,}696.$$

Сравнивая это значение с точным, видим, что абсолютная погрешность не превышает заданной точности ε: $|0{,}69315 - 0{,}696| < 0{,}01$.

Таким образом, за приближенное значение определенного интеграла по методу трапеций с точностью$\varepsilon = 0{,}01$ принимается значение

$$\int\limits_0^1 \frac{dx}{1+x} \approx 0{,}696.$$

2) *Метод Симпсона*. Исходя из заданной точности $\varepsilon_1 = 10^{-4}$ вычисляется шаг численного интегрирования для метода Симпсона по формуле (9) :

$$h \leqslant \sqrt[4]{\frac{180\varepsilon}{(b-a)\,M_4}}, \quad M_4 = \max_{x\in[0;\,1]} \left|f^{IV}(x)\right| = \max_{x\in[0;\,1]} \left|\frac{24}{(1+x)^5}\right| = 24;$$

$$h \leqslant \sqrt[4]{\frac{180\cdot 10^{-4}}{(1-0)\cdot 24}} = 10^{-1}\sqrt[4]{7{,}5} = 0{,}165.$$

Необходимо выбрать такой шаг, чтобы он удовлетворял неравенству $h \leqslant 0{,}165$, и чтобы на отрезке интегрирования $x \in \in [0;\,1]$ он укладывался *четное число* раз. Принимаем $h = 0{,}1$. С этим шагом для подынтегральной функции $f(x) = (1+x)^{-1}$ формируется сеточная функция с независимой переменной x_i, изменяющейся по закону $x_i = x_0 + ih = 0 + i\cdot 0{,}1$, $i = \overline{0,\,10}$, $n = = 10$, $m = 5$, причем значения сеточной функции вычисляются с точностью до четвертого знака после запятой:

i	0	1	2	3	4	5	6	7	8	9	10
x_i	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1,0
y_i	1,0	0,9091	0,8333	0,7692	0,7143	0,6667	0,625	0,5882	0,5556	0,5263	0,5

Используется формула Симпсона (7) численного интегрирования ($n = 10$, $m = 5$)

$$\int_0^1 \frac{dx}{1+x} = \frac{h}{3}\left(y_0 + y_n + 4\sum_{i=1}^{m} y_{2i-1} + 2\sum_{i=1}^{m-1} y_{2i}\right) = \frac{0{,}1}{3}\cdot[1{,}0+$$

$$+0{,}5 + 4\,(y_1 + y_3 + y_5 + y_7 + y_9) + 2\,(y_2 + y_4 + y_6 + y_8)] =$$

$$= \frac{0{,}1}{3}\,[1{,}5 + 4\,(0{,}9091 + 0{,}7692 + 0{,}6667 + 0{,}5882 + 0{,}5263) +$$

$$+2\,(0{,}8333 + 0{,}7143 + 0{,}625 + 0{,}5556)] =$$

$$= \frac{0{,}1}{3}\,(1{,}5 + 4\cdot 3{,}4595 + 2\cdot 2{,}7281) = \frac{0{,}1}{3}20{,}7942 = 0{,}69314.$$

Сравнение этого значения с точным значением интеграла показывает, что абсолютная погрешность не превышает заданной точности ε_1:

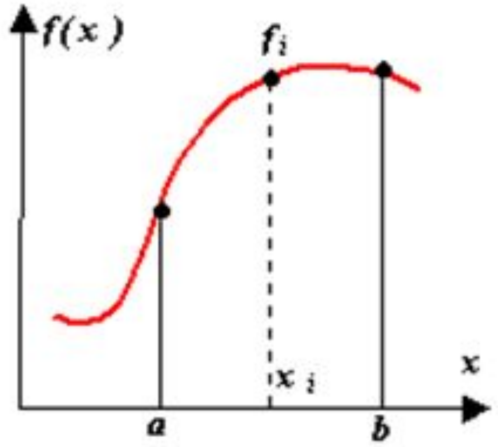
$$|0{,}69315 - 0{,}69314| < 0{,}0001.$$

Таким образом, за приближенное значение определенного интеграла по методу Симпсона с точностью $\varepsilon_1 = 0{,}0001$ принимается значение

$$\int\limits_0^1 \frac{dx}{1+x} \approx 0{,}6931.$$

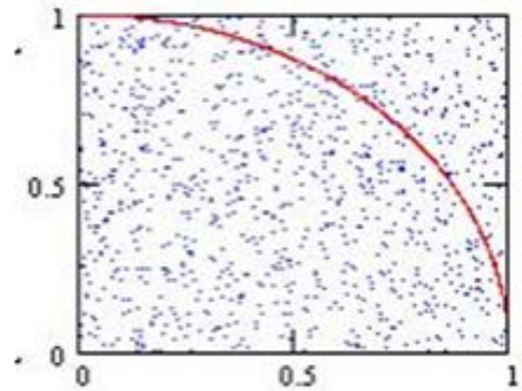
З а м е ч а н и е. Ясно, что для большинства интегралов от непрерывных функций первообразная не вычисляется (однако она существует) и вычисленное приближенное значение сравнивать не с чем, однако шаг численного интегрирования, вычисленный по заданной точности, гарантирует эту точность вычисления.

Методы Монте-Карло



1) одномерная *случайная величина* – статистический вариант метода прямоугольников.
В качестве текущего узла x_i берется случайное число, равномерно распределенное на интервале интегрирования [*a*, *b*]. Проведя *N* вычислений, значение интеграла определим по следующей формуле: $\int_a^b f(x)dx = \frac{b-a}{N}\sum_{i=1}^{N} f(x_i) + R$. Для *R* можно утверждать хотя бы $\sim \frac{1}{\sqrt{N}}$.

2) двумерная случайная величина– оценка площадей.



Рассматриваются две равномерно распределенных случайных величины x_i и y_i, которые можно рассматривать как координаты точки в двумерном пространстве. За приближенное значение интеграла принимается количества точек *S*, попавших под кривую $y = f(x)$, к общему числу испытаний *N*, т.е. $\int_a^b f(x)dx \approx \frac{S}{N}$.

И первый, и второй случаи легко обобщаются на кратные интегралы.

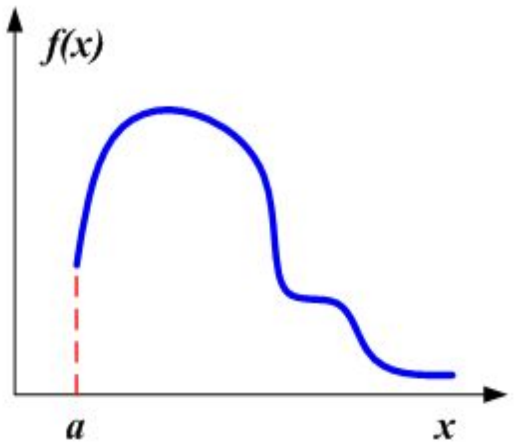
<u>Приближенное вычисление несобственных интегралов</u>

Несобственными называются интегралы, у которых один или оба предела интегрирования бесконечны:

$$\int_a^{+\infty} f(x)dx = \lim_{b\to\infty}\int_a^b f(x)dx \qquad \int_{-\infty}^{+\infty} f(x)dx = \lim_{\substack{a\to-\infty \\ b\to+\infty}}\int_a^b f(x)dx$$

Если пределы в правых частях равенств существуют, то интегралы называются ***сходящимися***, а в противном случае – ***расходящимися***.

Геометрически для неотрицательной подынтегральной функции несобственный интеграл $\int_a^{+\infty} f(x)dx$ представляет собой площадь криволинейной трапеции, ограниченной графиком подынтегральной функции, осью абсцисс и прямой $x=a$.



В основе приближенного вычисления несобственных интегралов лежит их представление в виде:

$$\int_{-\infty}^{+\infty} f(x)dx = \int_{-\infty}^{a} f(x)dx + \int_{a}^{b} f(x)dx + \int_{b}^{+\infty} f(x)dx \qquad (14)$$

При этом в случае сходящихся интегралов решение задачи заключается в следующем:

а) нахождение пределов интегрирования $[a,b]$, при которых значения первого и третьего интегралов несущественны;

б) приближенный расчёт второго интеграла в найденных пределах $[a,b]$ любым из рассмотренных выше методов.

Практически алгоритм определения несобственного интеграла включает итерационную процедуру расчётов. Сначала задаются одни пределы интегрирования, и вычисляется значение среднего интеграла , затем более широкие пределы $[a,b]$ и снова рассчитывается его значение. Так происходит до тех пор, пока два последовательных значения интеграла $\int_{a}^{b} f(x)dx$ при различных пределах $[a,b]$ не будут несущественно отличаться друг от друга.

Численное интегрирование в MATLAB

При вычислениях интегралов численными методами подынтегральную функцию целесообразно представлять в наиболее простом виде. Это может ускорить вычисления. Упрощение подынтегральной функции можно выполнить, воспользовавшись функциями `simplify()` и `factor()`. Имеют место случаи, когда система до упрощения не может вычислить неопределенный интеграл и легко его определяет после упрощения.

Метод трапеций реализован в MATLAB несколькими функциями, приведенными ниже.

Функция

`cumtrapz(y)`

осуществляет вычисление интеграла в случае, когда значения функции `y` заданы в виде вектора или матрицы неограниченных размеров. Откликом этой функции является n интегралов, где n — число элементов вектора или число элементов в каждом столбце матрицы.

Интегралы вычисляются по методу трапеций, когда значения функции y состоят из одного элемента вектора (первого), двух элементов вектора (первого и второго) и т. д. до последнего интеграла, когда число значений y равно числу элементов вектора или элементов столбцов матрицы (количество элементов в каждом столбце одинаково).

Такое вычисление интеграла называется *интегрированием с накоплением*. Программа вычисляет интеграл с шагом $h=1$. Если же ординаты получены с иным шагом, то результат вычислений интеграла нужно умножить на h.

Пример 1

Пусть функция $y(x)$ имеет значения, представленные в виде следующего вектора: `y=[1,2,3,4,5,6,7,8,9,10]`. Необходимо вычислить

$$\int_a^b y(x)dx .$$

При этом $a=1$, $b=1, 2, 3, ..., 10$.

Функция вычисления интеграла методом трапеций будет иметь вид:

```
>> y=[1,2,3,4,5,6,7,8,9,10];
>> cumtrapz (y)
ans =
     0   1.5000   4.0000    7.5000    12.0000    17.5000    24.0000
     31.5000    40.0000   49.5000
```

Проверим правильность вычисления интеграла:

$$\int_{1}^{1} 1\,dx = 0\,.$$

Далее воспользуемся формулой трапеций

$$\int_{a}^{b} f(x)\,dx = h\left(\frac{y_0}{2} + \sum_{k=1}^{n-1} y_k + \frac{y_n}{2}\right),$$

где:

- y_0 — значение подынтегральной функции при $x = a$;
- y_n — значение подынтегральной функции при $x = b$;
- h — шаг интегрирования.

$$\int_1^2 y(x)dx = \frac{y_0}{2} + \frac{y_1}{2} = \frac{1}{2} + \frac{2}{2} = 1.5,$$

$$\int_1^3 y(x)dx = \frac{y_0}{2} + y_1 + \frac{y_2}{2} = \frac{1}{2} + 2 + \frac{3}{2} = 4,$$

$$\int_0^{10} y(x)dx = \frac{y_0}{2} + y_1 + y_2 + ... + y_9 + \frac{y_{10}}{2} =$$

$$= \frac{1}{2} + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + \frac{10}{2} = 49.5.$$

Пример 2

Функция $y(x)$ является матрицей.

```
>> y=[1,2,3;2,3,4;3,4,5;4,5,6];
>> cumtrapz(y)
ans =
     0        0         0
     1.5      2.5       3.5
     4        6         8
     7.5     10.5      13.5
```

В каждом столбце исходной матрицы получены значения интеграла с накоплением.

Функция *cumtrapz(x,y)*

Эта функция выполняет интегрирование с накоплением функции $y(x)$ также методом трапеций. При этом x и y могут быть либо векторами одной и той же размерности, либо x — это вектор-столбец, а y — матрица.

Откликом является значение интеграла с диапазоном значений x.

Пример 3

Функция $y(x)$ задана в виде следующих векторов:

x=[1,3,7,9,10], y=[1,3,5,7,9].

Необходимо вычислить методом трапеций значение интеграла с накоплением.

Решение:

```
>> x=[1,3,7,9,10];
>> y=[1,3,5,7,9];
>> cumtrapz(x,y)
```

```
ans =
      0     4    20    32    40
```

Здесь вычисления значений интеграла с накоплением на i-м шаге S_i осуществляется по формуле:

$$S_i = S_{i-1} + \left(\frac{y_i}{2} + \frac{y_{i+1}}{2}\right)\Delta x,$$

где:

- S_{i-1} — предыдущее значение интеграла;
- y_i, y_{i+1} — значения функции в начале и в конце интервала Δx;
- Δx — шаг интегрирования на участке $[y_i; y_{i+1}]$.

Для нашего примера вычисления имеют вид:

- на участке вектора $[1; 3]$:

$$S_1 = 0 + \left(\frac{1}{2} + \frac{3}{2}\right) \cdot 2 = 4;$$

- на участке вектора $[3; 7]$:

$$S_2 = S_1 + \left(\frac{3}{2} + \frac{5}{2}\right) \cdot 4 = 20;$$

- на участке вектора $[7;9]$:

$$S_3 = 20 + \left(\frac{5}{2} + \frac{7}{2}\right) \cdot 2 = 32;$$

- на участке вектора $[9;10]$:

$$S_4 = 32 + \left(\frac{7}{2} + \frac{9}{2}\right) \cdot 1 = 40.$$

Как видно из примера, функция `cumtrapz(x,y)` вычисляет интеграл с накоплением при переменном шаге h.

Пример 4

Функция $y(x)$ задана в виде матрицы:

`y(x)= [1 3 5; 3 5 7; 4 6 8; 4 7 9; 5 7 10]`

При этом аргумент x представляет собой вектор следующего вида: `x=[1,3,7,9,10]`.

Вычислить значения интеграла.

Программа и результат вычисления интеграла с помощью функции `cumtrapz(x,y)` имеют вид:

```
>> x=[1  3  7  9  10];
>> y=[1 3 5;  3 5 7;  4 6 8; 4 7 9;  5 7 10];
>> cumtrapz(x,y)
```

```
ans =
     0         0      0
     4         8     12
    18        30     42
    26        43     59
    30.5      50     68.5
```

Функция *trapz(y)*

Функция возвращает значение интеграла

$$\int_a^b y(x)dx\,.$$

При этом $y(x)$ может быть вектором или матрицей. Если $y(x)$ — матрица, то функция возвращает вектор значений интеграла каждого столбца матрицы.

Пример 5

Необходимо вычислить методом трапеций значение интеграла, если функция $y(x)$ является следующим вектором:

`y=[1,3,5,7,9].`

```
>> y=[1 3 5 7 9];
>> trapz(y)
ans =
     20
```

Пример 6

Пусть функция $y(x)$ является матрицей, тогда программа решения будет иметь вид:

```
>> y=[1 3 5;  3 5 7;  4 6 8; 4 7 9;  5 7 10];
>> trapz(y)
ans =
     14    23    31.5
```

Функция *trapz(x,y)*

Вычисляет интеграл функции $y(x)$ по x методом трапеций. Аргумент и функция задаются в виде векторов или x — в виде вектора, а y — в виде матрицы.

Пример 7

Пусть аргумент x и функция $y(x)$ заданы в виде следующих векторов: `x=[1 3 7 9 10]`, `y=[1 3 5 7 9]`.

Необходимо вычислить интеграл методом трапеций.

```
>> x=[1 3 7 9 10];
>> y=[1 3 5 7 9];
>> trapz(x,y)
ans =
     40
```

Пример 8

Пусть аргументом функции $y(x)$ является вектор `x=[1 3 7 9 10]`, а функция — матрицей `y= [1 3 5; 3 5 7; 4 6 8; 4 7 9; 5 7 10]`. Вычислим значение интеграла методом трапеций, используя функцию `trapz(x,y)`.

```
>> x=[1 3 7 9 10];
>> y=[1 3 5;  3 5 7;  4 6 8; 4 7 9;  5 7 10];
>> trapz(x,y)
ans =
     30.5    50    68.5
```

Функции trapz(y) и trapz(x,y) допускают аналитическое задание подынтегральной функции $y(x)$. Однако в этом случае аргументу x должны быть присвоены численные значения во всем диапазоне интегрирования.

Пример 9

Пусть подынтегральная функция имеет вид

$$y(x) = xe^x + \ln x + 1.$$

Необходимо вычислить определенный интеграл в диапазоне от 1 до 10 с шагом 0.5.

```
>> x=1: 0.5: 10;
>> y=x.*exp(x)+log(x)+1;
>> trapz(y)
ans=
     4.0657e + 005
```

```
>> trapz(x,y)
ans =
     2.0328e + 005
```

Следует иметь в виду, что при вычислении интеграла с помощью функции trapz(x,y) его значение зависит от шага интегрирования.

Если функция задана таблично, для остаточного члена формулы трапеций можно использовать приближенное выражение

$$R_{трап} = -\frac{(b - a)^2}{12}\overline{\Delta^2 Y},$$

где черта сверху означает среднее значение по отрезку интегрирования.

Для вычисления среднего значения компонентов любого вектора в MATLAB используется функция `mean`. В данном случае ее надо применить к вектору вторых разностей (пример 14.5).

Пример 14.5. Использование приближенного значения остаточного члена

```
>> x=1:0.1:2;
>> y=log(x);
>> d2y=diff(y,2);
>> mean(d2y)
ans =
   -0.0049
```

Подставляя найденное среднее в формулу остаточного члена, получим $R_{трап} \approx 0.0004$. Если прибавить эту поправку к ранее найденному значению `0.3859`, получим `0.3863`, что в пределах отображаемого количества разрядов совпадает с точным значением.

Вычислительный алгоритм метода Симпсона реализован функцией **quad:**

Минимальная форма обращения к функции, реализующей интегрирование по Симпсону, — `q=quad(fun,a,b)`. В качестве первого аргумента задается указатель на подынтегральную функцию, второй и третий аргументы определяют пределы интегрирования. Указатель `fun` может быть задан одним из трех способов:

- именем m-функции, заключенным в одинарные кавычки;
- указателем `@fun`, где `fun` — имя функции;
- строкой, содержащей любую формулу с одной независимой переменной:

```
>> quad('log(x)',1,2)
ans =
     0.3863
```

Заметим, что в пределах отображаемого количества разрядов этот результат совпадает с точным значением.

Функция `quad` допускает задание четвертого входного параметра — абсолютной погрешности `eps`:

```
q=quad(fun,a,b,eps)
```

По умолчанию эта погрешность принимается равной `1.e-6`. Если задать ее более высокой (например, `1.e-16`), интеграл будет вычисляться точнее, зато существенно медленнее. О степени замедления можно судить по количеству `fcnt` обращений к вычислению значения подынтегральной функции

Функция *quad('fun',a,b)*

Функция вычисляет определенный интеграл

$$\int_a^b f(x)dx$$

с погрешностью, не превышающей 10^{-3}.

Подынтегральная функция $f(x)$ представляется в аналитическом виде с соблюдением правил записи функций в системе MATLAB.

Пример 10.

Подынтегральная функция имеет вид:

$$f(x) = e^x + x^2 + 2\sin x - 5 .$$

Необходимо вычислить интеграл

$$\int_1^5 f(x)dx .$$

```
>> y='exp(x)+x.^2+2*sin(x)-5';
>> quad(y,1,5)
```

```
ans =
      167.5415
```

Функция *quad('fun',a,b,tol)*

В функции `quad('fun',a,b,tol)` параметр `tol` является желаемой погрешностью, которая представляется в виде `1e-n`. По умолчанию `tol=1.e-3`.

Пример 11

Пусть подынтегральная функция имеет вид:

$$f(x) = e^x + x^2 + 2\sin x - 5 .$$

Необходимо вычислить интеграл в диапазоне от 1 до 5 с погрешностью не выше 10^{-7}.

```
>> quad('exp(x)+x.^2+2*sin(x)-5',1,5,1e-7)
ans =
    167.5415
```

Ответ тот же, что и в примере 10, когда вычисление интеграла выполнялось с погрешностью не более 10^{-3}. Это объясняется тем, что в ответе содержится только 4 значащих цифры после запятой. Отличие можно будет видеть при большем числе цифр после запятой.

Функция *dblquad('fun',a,b,c,d)*

В функции

```
dblquad ('fun',a,b,c,d)
```

приняты следующие обозначения:

- `'fun'` — это функция с двумя переменными;
- `a, b` — пределы по внутренней переменной;
- `c, d` — пределы по внешней переменной.

Пример 12

Функция двух переменных имеет вид:

$$z = x^2 + y^2 - 2\,.$$

Необходимо вычислить интеграл

$$\int_1^2 \int_0^3 z(x, y)\,dx dy\,.$$

```
>> z='x.^2+y.^2-2';
>>  dblquad (z,1,2,0,3)
```

```
ans =
      10
```

Решение получено с погрешностью, не превышающей 10^{-3}.

Повысим точность вычисления интеграла. Пусть `tol=1e-7`.

Тогда функция будет иметь вид:

```
>>   dblquad (z,1,2,0,3,1·e-7)
```

и после нажатия клавиши <Enter> получим:

```
ans =
     10
```

Ответ прежний потому, что интеграл и полученный ответ является точным решением.

Еще один дополнительный управляющий параметр `trace` в обращении к `quad` позволяет более детально проследить за последовательными итерациями (пример 14.8):

```
q=quad(fun,a,b,tol,trace)
```

Если `trace=1`, в процессе работы выдается последовательность строк вида `[fcnt a b-a q]`, в которых показаны текущие значения соответствующих переменных на каждой итерации:

- количество вычислений интегрируемой функции;
- левый конец промежутка;
- его длина;
- найденное значение интеграла по этому промежутку.

Если `trace=0` или этот параметр опущен, трассировка отключается.

Пример 14.8. Трассировка

```
>> q = quad('log(x)',1,2,[],1)
```

(Задание параметра `tol=[]` приводит к использованию его значения по умолчанию.)

Трассировка включает пять следующих строк:

```
     5 1.0000000000 1.00000000e+000 0.3862878935
     7 1.0000000000 5.00000000e-001 0.1081975273
     9 1.0000000000 2.50000000e-001 0.0289294372
    11 1.2500000000 2.50000000e-001 0.0792682225
    13 1.5000000000 5.00000000e-001 0.2780966816
```

И окончательный результат:

```
q = 0.3863
```

Если интегрируемую функцию невозможно (или трудно) задать одним арифметическим выражением, в качестве параметра можно указать имя файла с расширением m, в котором вычисляется эта функция. Обращение в этом случае выглядит так:

```
q = quad(@fun,...)
```

где fun.m — файл, доступный для MATLAB.

Символьные вычисления *неопределенных интегралов* в MATLAB осуществляется при помощи функции:

int(fun, var),

где fun – символьное выражение, представляющее собой подынтегральную функцию, а var – переменная интегрирования.

Пример 1. Вычисление неопределенного интеграла:

```
syms x %Определение символьной переменной
f=sym('exp(x) -x'); %Определение символьной функции
int(f,x) %Вычисление неопределенного интеграла
```

Результатом будет:

```
ans =
exp(x)-1/2*x^2
```

Для того чтобы вычислить *определенный интеграл*, можно использовать функцию:

int(fun, var, a, b),

где fun –подынтегральная функция, а var – переменная интегрирования, a, b – пределы интегрирования.

Для функции, заданной таблицей значений, построим сплайн-аппроксимацию (пример 14.10).

Пример 14.10. Интегрирование сплайн-функции

```
>> x = -3:3;
>> y =[-1 -1 -1 0 1 1 1];
>> pp = pchip(x,y);
```

Значение полученной сплайн-функции в любой точке `xx` (или для вектора `xx`) можно найти с помощью команды `yy=ppval(pp,xx)`. К сожалению, порядок аргументов функции `ppval` таков, что непосредственно подставить `ppval` в `quad` невозможно — придется использовать функцию-посредника:

```
function yy = newppval(xx,pp)
yy = ppval(pp,xx);
```

Теперь сплайн-функцию можно проинтегрировать, передавая посреднику структуру `pp`:

```
>> q = quad(@newppval,-3,3,[],[],pp)
q =
    0
>> q = quad(@newppval,0,3,[],[],pp)
q =
    2.5833
```

Те же самые результаты можно получить и другим способом: сначала найти первообразную от сплайна с помощью функции `ppint=fnint(pp)`, а потом взять разность значений первообразной в соответствующих точках:

```
>> ppval(ppint,3)- ppval(ppint,-3)

ans =

     0

>> ppval(ppint,3)- ppval(ppint,0)

ans =

     2.5833
```

Пример 2. Вычисление определенного интеграла:

```
I1=int('exp(x)-x','x',-1,0); %Символьное решение
vpa(I1,5)   %Численное решение с 5 значащими цифрами
```

Результатом будет: I1 = -exp(-1)+3/2

```
ans =
1.1321
```

Пример 3.

В М-файле с именем Simpson.m пишем:

```
function y=G(x)
y=exp(x)-x;
end
```

Потом в командном окне вызываем функцию quad:

```
format long %Формат вывода значений
quad('Simpson',-1,0,1.0e-05)
```

Результатом будет:

```
ans =
1.13212056020538
```

Итак, функция **int** вычисляет:

- неопределенный интеграл;
- неопределенный интеграл с символьными переменными;
- определенный интеграл с символьными значениями пределов интегрирования;
- определенный интеграл от алгебраических функций;
- кратные интегралы;
- несобственные интегралы.

Пример 4

Пусть необходимо вычислить двойной неопределенный интеграл вида:

$$\iint \frac{x}{1-x^2}\,dx\,.$$

Решение путем двойного интегрирования имеет вид:

```
>> syms x;
>> y=x/(1-x^2);
>> Z=int(y)
Z =
     -1/2*log(x-1)-1/2*log(x+1)
>> int(Z)
ans =
     -1/2*log(x-1)*(x-1)+x-1/2*log(x+1)*(x+1)
```

Программа будет иметь более простой вид, если функцию `int()` повторить n раз при n-кратном интегрировании.

Для нашего примера решение будет иметь вид:

```
>> syms x;
>> y=x/(1-x^2);
>> int(int(y))
ans =
      -1/2*log(x-1)*(x-1)+x-1/2*log(x+1)*(x+1)
```

Пример 5

Необходимо вычислить следующий определенный интеграл:

$$\int_0^\infty \frac{x}{\sinh(5x)}dx.$$

```
>> syms x;
>> y=x/sinh(5*x);
>> int(y,0,inf)
ans =
      pi^2/100
```

А теперь вычислим тот же интеграл, введя символьную переменную a:

$$\int_0^\infty \frac{x}{\sinh(ax)}dx.$$

Повторим вычисление интеграла:

```
>> syms x a;
>> y=x/sinh(a*x);
>> int(y,0,inf)
ans =
```

Решение в явном виде не получено. Причина в том, что переменная a не определена. Это может быть число положительное или отрицательное, действительное или комплексное. Предположим, что это число действительное и положительное. Тогда программа будет иметь вид:

```
>> syms x a;
>> y=x/sinh(abs(a)*x);
>> int(y,0,inf)
abs =
     pi^2/(4*a^2)
```

Пример 6

Требуется вычислить следующий несобственный интеграл:

$$\int_0^{\infty} \frac{dx}{(x+\sqrt{x^2+a^2})^3}.$$

Решение имеет вид:

```
>> syms x a;
>> y=1/(x+sqrt(x^2+a^2))^3;
>> int(y,0,inf)
ans =
     3/(8*a^2)
```

Не нужно указывать программе на знак числа a и писать `abs(a)`, т. к. число a возводится в квадрат и является положительным.

Задания.

1. Вычислить интеграл $\int_0^{10}\left(xe^{-x}+\ln x+1\right)dx$.

   а) аналитически

   б) методом трапеций с точностью ε=10^{-2}

   в) методом Симпсона с точностью ε=10^{-4}

   Для метода трапеций применить процедуру Рунге уточнения формулы численного интегрирования.

   Решить задачу, используя стандартные функции MATLAB. Сравнить полученные результаты .

2. Вычислить неопределённый интегр $\int a^x e^{-x}\,dx$

3. Вычислить несобственный интеграл

$$\int_0^{\infty}\frac{1+x}{(x+a)^{p+1}}dx$$