4.14. NUMERICAL INTEGRATION

Suppose on the interval $[a,b]$, there is a discrete set of mismatched points x_i, $i = 0,...,n$, $x_0 = a$, $x_n = b$ (interpolation nodes), at which the values of a certain function $y(x)$, generating the table $y_i = y(x_i)$, $i = 0,...,n$, are known. It is required to determine the value $\int_a^b y(x)\,dx$ of the definite integral of the function on the interval $[a,b]$. The values a and b are called the lower and upper limits, and the function $y(x)$ is called the subintegral function. The problem of numerical integration arises either in those cases when the function $y(x)$ is a table one, or when it is not possible to analytically find the primitive $\int y(x)\,dx$ or calculate a definite integral $\int_a^b y(x)\,dx$.

Definition. Let's recall that the *definite integral* $\int_a^b y(x)\,dx$ of the function $y(x)$ of one variable x on the interval $[a,b]$ according to Riemann is the limit of the sequence of the following integral sums when the partition tends to zero independent of the choice of the partition h and points ξ_i:

$$\int_a^b y(x)\,dx = \lim_{h \to 0} \sum_{i=1}^{n} y(\xi_i) h_i ,$$

where $h = (h_1, h_2, \ldots, h_n)^T$ is the partition vector, $h_i = x_i - x_{i-1}$, $i = 1,...,n$ and $\xi_i \in [x_{i-1}, x_i]$.

Then, for the table $y_i = y(x_i)$, $i = 0,...,n$, a definite integral of the function $y(x)$ on the interval $[a,b]$ in accordance with the Riemann definition can be numerically calculated as follows:

$$\int_a^b y(x)\,dx = \sum_{i=1}^{n} y(\xi_i) h_i ,\ h_i = x_i - x_{i-1},\ i = 1,...,n .$$

Depending on the rule for choosing the point ξ_i on the interval $[x_{i-1}, x_i]$, this formula can take several different forms:

when $\xi_i = x_{i-1}$, the integral is defined as $\int_a^b y(x)\,dx = \sum_{i=1}^{n} y(x_{i-1}) h_i$, when $\xi_i = x_i$, the integral is defined as $\int\limits_a^b y(x)\,dx = \sum\limits_{i=1}^n y(x_i)h_i$,

when $\xi_i = \dfrac{x_{i-1}+x_i}{2}$, the integral is defined as $\int\limits_a^b y(x)\,dx = \sum\limits_{i=1}^n y(\dfrac{x_{i-1}+x_i}{2})h_i$.

The first form is called the left-point rectangle rule, the second one – the right-point rectangle rule, and the third one – the midpoint quadrature rule. Obviously, in the third case, when $\xi_i = \dfrac{x_{i-1}+x_i}{2} = x_{i-1/2}$, the corresponding value of the function $y(\dfrac{x_{i-1}+x_i}{2}) = y_{i-1/2}$ can be obtained either if the subintegral function $y(x)$ is initially known analytically, or (if the subintegral function $y(x)$ is given on a table on the uniform partition h) as a result of renumbering the interpolation nodes x_i from $i = 0,\ldots,n = 2l$ to $i = 0,\ldots,l$.

Geometrically definite integral of a positive function is numerically equal to the area of the curvilinear trapezoid limited by the abscissa axis, vertical lines $x = a$ and $x = b$, and the function graph. If the subintegral function can take negative values, the area of the corresponding part of the figure under the abscissa axis is considered negative.

In this regard, it is obvious that the rectangle rules, in accordance with their name, geometrically represent an approximation of a definite integral through the sum of the areas of rectangles limited by the abscissa axis, vertical lines $x = x_{i-1}$ and $x = x_i$, and the horizontal line $y = y(\xi_i)$ (fig. 9).

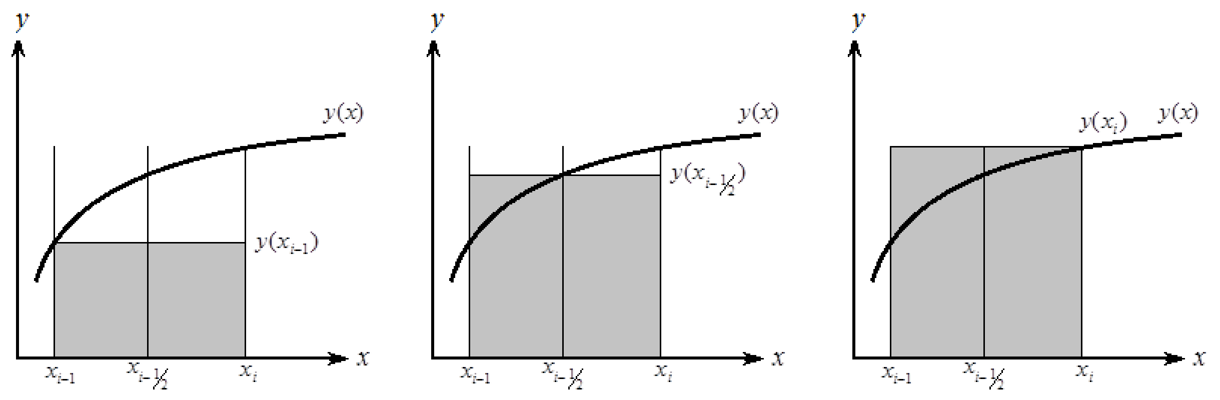

Fig. 9. Geometric interpretation for the left-point, midpoint and right-point rectangle rules

However, the main approach to deriving the necessary formulas for numerical integration (of any order of accuracy) is to use polynomial interpolation.

The original subintegral function $y(x)$ for the table $y_i = y(x_i)$, $i = 0,...,n$ on each interval $[x_{i-1}, x_i]$, $h_i = x_i - x_{i-1}$, $i = 1,...,n$ is replaced by a certain approximating, easily integrable function $\varphi(x,a)$, $y(x) = \varphi(x,a) + R(x)$, where $R(x)$ is the remainder of the approximation, a is the vector of parameters, and it is assumed that $\int\limits_{x_{i-1}}^{x_i} y(x)\,dx \approx \int\limits_{x_{i-1}}^{x_i} \varphi(x,a)\,dx$.

The interpolation polynomial of degree m $\varphi(x,a) = P_m(x) = \sum\limits_{j=0}^{m} a_j x^j$ is most often taken as the approximating function $\varphi(x,a)$, which is constructed using local interpolation formulas, after which this polynomial is integrated. Herewith, the coefficients a_j of this polynomial, generally speaking, are different on each interval $[x_{i-1}, x_i]$ and are determined from the $m+1$ conditions of equality of the values of this polynomial and the values of the function $y(x)$ at points x_{i-1} and x_i and at $m-1$ intermediate points of the interval $[x_{i-1}, x_i]$. As discussed earlier, when $m > 1$, the values of the subintegral function $y(x)$ at these intermediate points are obtained either if the function $y(x)$ is initially known analytically, or (if the subintegral function $y(x)$ is given on a table on a uniform partition h) as a result of renumbering the interpolation nodes x_i from $i = 0,...,n = m \cdot l$ to $i = 0,...,l$.

Herewith, it is usually convenient to use an interpolation polynomial in the Lagrangian form to derive formulas for numerical integration.

To demonstrate the approach to numerical integration using polynomial interpolation, let's present the derivation of various formulas for numerical integration of different orders of accuracy.

If the Lagrange interpolation polynomial of zero degree $P_0(x) = a_0$ is used to approximate the subintegral function $y(x)$ on the interval $[x_{i-1}, x_i]$, $h_i = x_i - x_{i-1}$,

$i=1,\dots,n$, the function $y(x)$ can be approximated by any horizontal line segments $y=y(\xi)$, $\xi\in[x_{i-1},x_i]$, for example, $y=y(x_{i-1/2})$, then $y(x)\approx\varphi(x)=y_{i-1/2}$.

After polynomial integration, we get:

$$\int_{x_{i-1}}^{x_i} y(x)\,dx \approx \int_{x_{i-1}}^{x_i} \varphi(x)\,dx = \int_{x_{i-1}}^{x_i} y_{i-1/2}\,dx = y_{i-1/2}\,x\Big|_{x_{i-1}}^{x_i} = y_{i-1/2}(x_i - x_{i-1}) = y_{i-1/2}h_i .$$

As a result, we get the midpoint quadrature rule as follows:

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

In case of a constant integration step $h_i = x_i - x_{i-1} = h = \dfrac{b-a}{n}$, $i=1,\dots,n$ and the existence of $f''(x),\ x\in[a,b]$, the following estimate of the remainder of the midpoint quadrature rule takes place:

$$R \le \frac{b-a}{24}h^2 M_2, \text{ where } M_2 = \max_{x\in[a,b]}\left| f''(x)\right| ,$$

i.e. the midpoint quadrature rule has the second order of accuracy with respect to the integration step *h*, which is associated with the symmetric choice of the integration node $x_{i-1/2}$.

If the function $y(x)$ is approximated by the horizontal line segments $y=y(x_{i-1})$ or $y=y(x_i)$, then the left-point and right-point rectangle rules are derived similarly:

$$\int_a^b y(x)\,dx \approx \sum_{i=1}^{n} y_{i-1}h_i ,\quad \int_a^b y(x)\,dx \approx \sum_{i=1}^{n} y_i h_i .$$

However, the following estimate of the remainder takes place for them:

$$R \le \frac{b-a}{2}hM_1, \text{ where } M_1 = \max_{x\in[a,b]}\left| f'(x)\right| ,$$

i.e. all other rectangle rules have the first order of accuracy with respect to the integration step *h*.

If the Lagrange interpolation polynomial of the first degree $P_1(x)=a_0+a_1x$ is used to approximate the subintegral function $y(x)$ on the interval $[x_{i-1},x_i]$,

$h_i = x_i - x_{i-1}$, $i = 1,...,n$, the function $y(x)$ can be approximated by a straight line segment that passes through points (x_{i-1}, y_{i-1}) and (x_i, y_i):

$$y(x) \approx \varphi(x) = y_{i-1}\frac{(x-x_i)}{(x_{i-1}-x_i)} + y_i\frac{(x-x_{i-1})}{(x_i-x_{i-1})} = \frac{1}{h_i}\big(y_i(x-x_{i-1}) - y_{i-1}(x-x_i)\big).$$

After polynomial integration, we get:

$$\int_{x_{i-1}}^{x_i} y(x)\,dx \approx \int_{x_{i-1}}^{x_i} \varphi(x)\,dx = \int_{x_{i-1}}^{x_i} \frac{1}{h_i}\big(y_i(x-x_{i-1}) - y_{i-1}(x-x_i)\big)dx =$$

$$= \frac{1}{2h_i}\big(y_i(x-x_{i-1})^2 - y_{i-1}(x-x_i)^2\big)\Big|_{x_{i-1}}^{x_i} = \frac{1}{2h_i}\big(y_i h_i^2\big) - \frac{1}{2h_i}\big(-y_{i-1}h_i^2\big) = \frac{h_i}{2}\big(y_i + y_{i-1}\big).$$

As a result, we get the trapezoid rule as follows:

$$\int_a^b y(x)\,dx \approx \frac{1}{2}\sum_{i=1}^{n}(y_{i-1} + y_i)h_i .$$

It is easy to see that the trapezoid rule is a half-sum of the left-point and right-point rectangle rules.

In case of a constant integration step $h_i = x_i - x_{i-1} = h = \frac{b-a}{n}$, $i = 1,...,n$ and the existence of $f''(x), x \in [a,b]$, the value of the remainder of the trapezoid rule is estimated as follows:

$$R \le \frac{b-a}{12}h^2 M_2, \text{ where } M_2 = \max_{x\in[a,b]}|\,f''(x)|\,,$$

i.e. the trapezoid rule has the second order of accuracy, herewith its error value turns out to be twice as much as that of the midpoint quadrature rule, which is associated with the absence of the additional information provided as a result of inclusion of the integration node $x_{i-1/2}$.

The trapezoid rule, in accordance with its name, geometrically represents an approximation of a definite integral through the sum of the areas of rectangular trapezoids limited by the abscissa axis, vertical lines $x = x_{i-1}$ and $x = x_i$, and the straight line $\varphi(x) = \frac{1}{h_i}\big(y_i(x-x_{i-1}) - y_{i-1}(x-x_i)\big)$ (fig. 10).

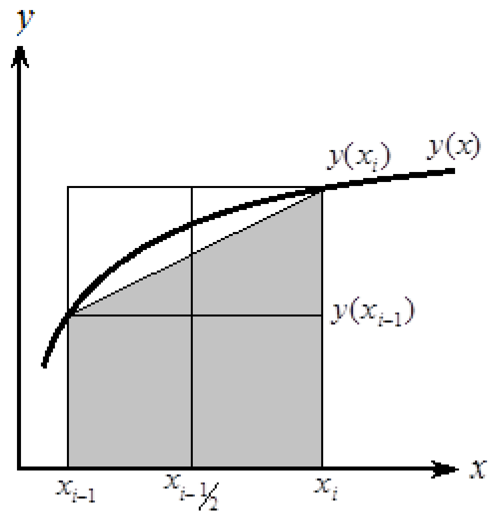

Fig. 10. Geometric interpretation for the trapezoid rule

If the Lagrange interpolation polynomial of the second degree $P_2(x) = a_0 + a_1 x + a_2 x^2$ is used to approximate the subintegral function $y(x)$ on the interval $[x_{i-1}, x_i]$, $h_i = x_i - x_{i-1}$, $i = 1,\ldots,n$, the function $y(x)$ can be approximated by a parabola segment that passes through the ends and the middle of the integration interval – points (x_{i-1}, y_{i-1}), (x_i, y_i) and $(x_{i-1/2}, y_{i-1/2})$:

$$y(x) \approx \varphi(x) = y_{i-1}\frac{(x - x_{i-1/2})}{(x_{i-1} - x_{i-1/2})}\frac{(x - x_i)}{(x_{i-1} - x_i)} +$$

$$+ y_{i-1/2}\frac{(x - x_{i-1})}{(x_{i-1/2} - x_{i-1})}\frac{(x - x_i)}{(x_{i-1/2} - x_i)} + y_i\frac{(x - x_{i-1})}{(x_i - x_{i-1})}\frac{(x - x_{i-1/2})}{(x_i - x_{i-1/2})} =$$

$$= \frac{2}{h_i^2}\Big(y_{i-1}(x - x_{i-1/2})(x - x_i) - 2y_{i-1/2}(x - x_{i-1})(x - x_i) + y_i(x - x_{i-1})(x - x_{i-1/2})\Big).$$

After polynomial integration, we get:

$$\int_{x_{i-1}}^{x_i} y(x)\,dx \approx \int_{x_{i-1}}^{x_i} \varphi(x)\,dx =$$

$$= \int_{x_{i-1}}^{x_i} \frac{2}{h_i^2}\Big(y_{i-1}(x - x_{i-1/2})(x - x_i) - 2y_{i-1/2}(x - x_{i-1})(x - x_i) + y_i(x - x_{i-1})(x - x_{i-1/2})\Big)dx =$$

$$= \frac{1}{3h_i^2}\Big(3y_{i-1}(x - x_{i-1/2})(x - x_i)^2 - y_{i-1}(x - x_i)^3\Big)\Big|_{x_{i-1}}^{x_i} +$$

$$+ \frac{2}{3h_i^2}\Big(-3y_{i-1/2}(x - x_{i-1})(x - x_i)^2 + y_{i-1/2}(x - x_i)^3\Big)\Big|_{x_{i-1}}^{x_i} +$$

$$+\frac{1}{3h_i^2}\left(3y_i(x-x_{i-1/2})(x-x_{i-1})^2-y_i(x-x_{i-1})^3\right)\Bigg|_{x_{i-1}}^{x_i}=$$

$$=-\frac{1}{3h_i^2}\left(-\frac{3}{2}y_{i-1}h_i^3+y_{i-1}h_i^3\right)-\frac{2}{3h_i^2}\left(-y_{i-1/2}h_i^3\right)+\frac{1}{3h_i^2}\left(\frac{3}{2}y_ih_i^3-y_ih_i^3\right)=$$

$$=\frac{h_i}{3}\left(\left(\frac{3}{2}y_{i-1}-y_{i-1}\right)+\left(2y_{i-1/2}\right)+\left(\frac{3}{2}y_i-y_i\right)\right)=$$

$$=\frac{h_i}{3}\left(\frac{1}{2}y_{i-1}+2y_{i-1/2}+\frac{1}{2}y_i\right)=\frac{h_i}{6}\left(y_{i-1}+4y_{i-1/2}+y_i\right).$$

As a result, we get the Simpson's rule (the parabolic rule) as follows:

$$\int_a^b y(x)\,dx\approx\frac{1}{6}\sum_{i=1}^{n}(y_{i-1}+4y_{i-1/2}+y_i)h_i\ .$$

In case of a constant integration step $h_i=x_i-x_{i-1}=h=\frac{b-a}{n}$, $i=1,...,n$ and the existence of $f^{(4)}(x),\ x\in[a,b]$, the value of the remainder of the Simpson's rule is estimated as follows:

$$R\le\frac{b-a}{180}h^4M_4\text{, where } M_4=\max_{x\in[a,b]}\left|f^{(4)}(x)\right|,$$

i.e. the Simpson's rule immediately has the fourth order of accuracy, which is again associated with the symmetric choice of the integration node $x_{i-1/2}$.

The Simpson's rule geometrically represents an approximation of a definite integral through the sum of the areas of curvilinear trapezoids limited by the abscissa axis, vertical lines $x=x_{i-1}$ and $x=x_i$ and the parabola $\varphi(x)=\frac{2}{h_i^2}\left(y_{i-1}(x-x_{i-1/2})(x-x_i)-y_{i-1/2}(x-x_{i-1})(x-x_i)+y_i(x-x_{i-1})(x-x_{i-1/2})\right)$ (fig. 11), which is associated with its second name – the parabolic rule.

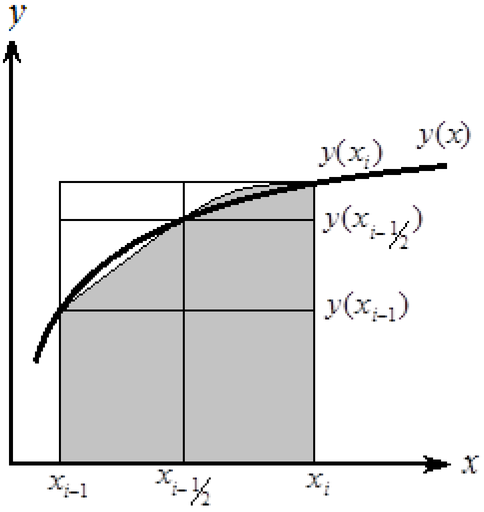

Fig. 11. Geometric interpretation for the Simpson's rule

If the Lagrange interpolation polynomials of a higher degree are used to approximate the subintegral function $y(x)$ on the interval $[x_{i-1}, x_i]$, $h_i = x_i - x_{i-1}$, $i = 1,...,n$, other numerical integration rules of even higher orders of accuracy can be obtained in a similar way that use the values of the function $y(x)$ at several intermediate points of the integration interval $[x_{i-1}, x_i]$.

Example. Using a table of values of the function $y(x)$ (values y_i calculated at points x_i, $i = 0,...,n$), calculate numerically the definite integral $\int_{x_0}^{x_n} y(x)\,dx$ with the left-point and right-point rectangular rules, midpoint quadrature rule, trapezoid rule and Simpson's rule. Calculate the value of the interpolation error for each method.

$y = x\ln(x)$, $n = 4$, $i = 0,...,4$, $x_i =$ 0.1, 0.5, 0.9, 1.3, 1.7.

Let's note that all nodes x_i, $i = 0,...,n$ are equally spaced and let's introduce the notation $h = x_i - x_{i-1} = 0.4$, $i = 1,...,n$.

To begin with, let's calculate for subsequent comparison the exact value of the definite integral of the function $y(x)$ on the interval:

$$\int y(x)\,dx = \int x\ln(x)\,dx = \int \ln(x)\,d\,\frac{x^2}{2} = \ln(x)\frac{x^2}{2} - \int \frac{x^2}{2}\,d\ln(x) =$$

$$= \ln(x)\frac{x^2}{2} - \int \frac{x}{2}\,dx = \ln(x)\frac{x^2}{2} - \frac{x^2}{4} = \frac{x^2}{4}(2\ln(x) - 1).$$

$$\int_{x_0}^{x_n} y(x)\,dx = \int_{x_0}^{x_n} x\ln(x)\,dx = \frac{x^2}{4}(2\ln(x) - 1)\Bigg|_{x_0}^{x_n} = \frac{x_n^2}{4}(2\ln(x_n) - 1) - \frac{x_0^2}{4}(2\ln(x_0) - 1).$$

$$I=\int_{0.1}^{1.7} x\ln(x)\,dx=\frac{1.7^2}{4}(2\ln(1.7)-1)-\frac{0.1^2}{4}(2\ln(0.1)-1)=0.0583\,.$$

Then, let's calculate numerically the definite integral using the left-point and right-point rectangular rule and midpoint quadrature rule:

$$I_l=\sum_{i=1}^{n} y_{i-1}h_i=h\sum_{i=1}^{4} y_{i-1}=0.4\cdot(y_0+y_1+y_2+y_3)=-0.1322\,.$$

$$I_r=\sum_{i=1}^{n} y_i h_i=h\sum_{i=1}^{4} y_i=0.4\cdot(y_1+y_2+y_3+y_4)=0.3207\,.$$

$$I_m=\sum_{i=1}^{n} y_{i-1/2}h_i=h\sum_{i=1}^{4} y_{i-1/2}=0.4\cdot(y_{0.5}+y_{1.5}+y_{2.5}+y_{3.5})=0.0409\,.$$

The absolute interpolation error is:

$$\Delta_l=|I_l-I|=|-0.1322-0.0583|=0.1905\,.$$

$$\Delta_r=|I_r-I|=|0.3207-0.0583|=0.2624\,.$$

$$\Delta_m=|I_m-I|=|0.0409-0.0583|=0.0174\,.$$

Now, let's calculate numerically the definite integral using the trapezoid rule:

$$I_t=\frac{1}{2}\sum_{i=1}^{n}(y_{i-1}+y_i)h_i=\frac{h}{2}\sum_{i=1}^{4}(y_{i-1}+y_i)=\frac{0.4}{2}\cdot(y_0+2y_1+2y_2+2y_3+y_4)=0.0942\,.$$

It is easy to see that $I_t=\dfrac{I_l+I_r}{2}$.

The absolute interpolation error is:

$$\Delta_t=|I_t-I|=|0.0942-0.0583|=0.0360\,.$$

Finally, let's calculate numerically the definite integral using the Simpson's rule:

$$I_s=\frac{1}{6}\sum_{i=1}^{n}(y_{i-1}+4y_{i-1/2}+y_i)h_i=\frac{h}{6}\sum_{i=1}^{4}(y_{i-1}+4y_{i-1/2}+y_i)=$$

$$=\frac{0.4}{6}\cdot(y_0+4y_{0.5}+2y_1+4y_{1.5}+2y_2+4y_{2.5}+2y_3+4y_{3.5}+y_4)=0.0587\,.$$

The absolute interpolation error is:

$$\Delta_s=|I_s-I|=|0.0587-0.0583|=0.0004\,.$$

It is easy to see that the use of the formulas with a higher order of accuracy (second for the midpoint quadrature and trapezoid rules, fourth for the Simpson's rule) allows getting an answer with a lower error.

Let's note that to calculate the values of the required definite integrals, the values of the function y_i were required, calculated both at main and at intermediate fractional points x_i.

Calculation results are given in table 13.

Table 13

i	**0**	**1**	**2**	**3**	**4**
x_i	0.1	0.5	0.9	1.3	1.7
y_i	-0.2303	-0.3466	-0.0948	0.3411	0.9021
i	**0.5**	**1.5**	**2.5**	**3.5**	
x_i	0.3	0.7	1.1	1.5	
y_i	-0.3612	-0.2497	0.1048	0.6082	

4.15. PROGRAM #17

Below is a proposed variant of the program algorithm for the numerical calculation of the value of a definite integral of a function on an interval with the left-point and right-point rectangle rule, midpoint quadrature rule, trapezoid rule, Simpson's rule.

```
ALGORITHM “Numerical integration”
INPUT     f(), n, x[2*n-1]
OUTPUT    i, y[2*n-1], s, t, u, v, w
BEGIN
    CYCLE “Values” FOR i FROM 1 TO 2*n-1 BY 1
        y[i]:=f(x[i])
    s:=0, t:=0, u:=0, v:=0, w:=0
    CYCLE “Summands” FOR i FROM 1 TO n-1 BY 1
```

```
            s:=s+y[2*i-1]*(x[2*i+1]-x[2*i-1])
            t:=t+y[2*i+1]*(x[2*i+1]-x[2*i-1])
            u:=u+y[2*i]*(x[2*i+1]-x[2*i-1])
            v:=v+(y[2*i+1]+y[2*i-1])/2*(x[2*i+1]-x[2*i-1])
            w:=w+(y[2*i+1]+4*y[2*i]+y[2*i-1])/6*(x[2*i+1]-x[2*i-1])
        PRINT s, t, u, v, w
    END
```

4.16. RUNGE-ROMBERG-RICHARDSON RULE

Suppose on the interval $[a,b]$, there is a discrete set of mismatched points x_i, $i=0,...,n$, $x_0=a$, $x_n=b$ (interpolation nodes) with a constant step $h_i=x_i-x_{i-1}=h=\dfrac{b-a}{n}$, at which the values of a certain function $y(x)$, generating the table $y_i=y(x_i)$, $i=0,...,n$, are known.

When numerically calculating either some derivative of a function at a certain point of an interval or a definite integral on the interval, it is obvious that a more accurate result is obtained on a partition with a larger number of nodes *n*, but also the corresponding calculations require significantly more arithmetic operations. The Runge-Romberg-Richardson rule allows to obtain a higher order of accuracy for the desired approximation of the derivative or integral *I* using one and the same formulas and with the same number of nodes *n* but without the significant increase of the number of operations.

Suppose there is a known numerical approximation I_h for *I*, which was calculated using a formula with a known order of accuracy *p* on this partition $i=0,...,n=m\cdot l$, $m=2,3,...$ with the step *h*. Suppose also the same numerical approximation I_{mh} is known, but calculated on a certain smaller, proportional to the initial one, partition $i=0,...,l$ with the step $m\cdot h$.

Then, the Runge-Romberg-Richardson rule allows, having these two approximations I_h and I_{mh} on two proportional partitions, to give an a-posteriori estimate of the error of the numerical approximation I_h. This estimate can then be used to refine the given result of numerical differentiation or numerical integration, allowing a higher order of accuracy of the calculation $p+1$ to be obtained.

Indeed, on a partition with the step *h*, the following is true:

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

where $\psi \cdot h^p$ is the main term of error of the approximate calculation for *I*.

Similarly, on a partition with the step $m \cdot h$ the following is true:

$$I = I_{mh} + O((mh)^p) = I_{mh} + \psi \cdot (mh)^p + O(m^{p+1}h^{p+1}).$$

After subtraction, the unknown factor ψ can be expressed as follows:

$$0 = I_h - I_{mh} + \psi \cdot h^p \cdot (1 - m^p) + O(h^{p+1}), \ \psi = \frac{I_h - I_{mh}}{h^p \cdot (m^p - 1)} + O(h).$$

Then the error of the numerical approximation for I_h is estimated as follows:

$$|I - I_h| \approx |\psi \cdot h^p| = \frac{|I_h - I_{mh}|}{m^p - 1}.$$

Accordingly, the refinement of the numerical approximation I_h looks as follows:

$$I = I_h + \psi \cdot h^p + O(h^{p+1}) = I_h + \frac{I_h - I_{mh}}{m^p - 1} + O(h^{p+1}) = \tilde{I}_h + O(h^{p+1}), \ \tilde{I}_h = I_h + \frac{I_h - I_{mh}}{m^p - 1}.$$

So, the Runge-Romberg-Richardson rule can be used to estimate the error and obtain the result with a higher degree of accuracy when solving a variety of different problems in which a certain value is calculated on partitions with different proportional steps, for example, during numerical integration, numerical differentiation, solving differential equations.

To increase the order of accuracy of the numerical approximation by more than one order, the idea of the Runge-Romberg-Richardson rule can be used on a chain of proportional partitions.

Having the Runge-Romberg-Richardson mechanism of increasing the order of accuracy of a numerical approximation by one and *k* calculated approximations $I_h, I_{mh}, I_{m^2h}, \ldots, I_{m^{k-1}h}$ on proportional partitions with a known order of accuracy *p* each, we can compose a pyramidal algorithm that allows to obtain approximations I_h

with the order of accuracy up to $q = p + k - 1$. For this purpose, a total of $l = \frac{k(k-1)}{2}$ refinements are carried out, on each of which, on the basis of the existing set of obtained approximations, new ones are constructed with the order of accuracy increased by one.

An example of such a pyramid of refinements for $k = 4$ and $l = \frac{k(k-1)}{2} = 6$ refinements is shown in fig. 12.

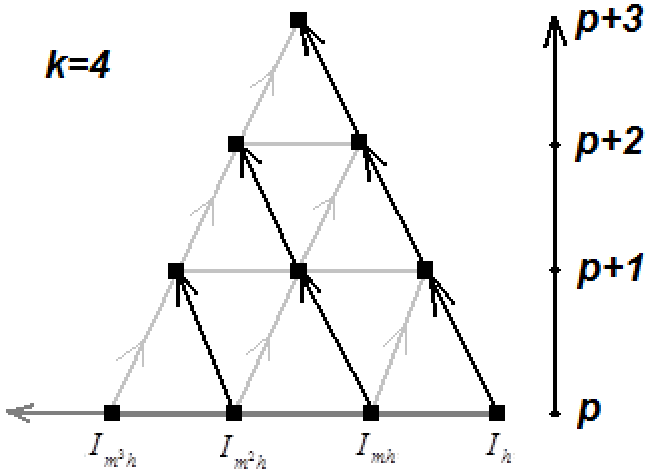

Fig. 12. An example of a pyramid of refinements by the Runge-Romberg-Richardson rule

When constructing a pyramid of refinements, it is convenient to use a table of refinements, which is filled in by columns sequentially from left to right – the first column is filled in according to the chosen solution method, and the following columns are filled in according to the Runge-Romberg-Richardson formula.

It is easy to see that on the initial partition with the number of nodes $n+1$, the maximum increase in the order of accuracy of the initial approximation is $q - p = k - 1 = [\log_m n]$; herewith, the calculations require approximately double the number of arithmetic operations (for calculating all auxiliary approximations), as the refinements take around $\log^2 n$.

Example. Using a table of values of the function $y(x)$ (values y_i calculated at points x_i, $i=0,...,n$), calculate numerically the definite integral $\int_{x_0}^{x_n} y(x)\,dx$ with the trapezoid rule. Refine the obtained solution with the Runge-Romberg-Richardson rule by two orders.

$y = x\ln(x)$, $n=4$, $i=0,...,4$, $x_i =$ 0.1, 0.5, 0.9, 1.3, 1.7.

Let's note that all nodes x_i, $i=0,...,n$ are equally spaced and let's introduce the notation $h = x_i - x_{i-1} = 0.4$, $i=1,...,n$.

The exact value of the definite integral of the function $y(x)$ on the interval is:

$$I = \int_{0.1}^{1.7} x\ln(x)\,dx = \frac{x^2}{4}(2\ln(x)-1)\Big|_{0.1}^{1.7} = \frac{1.7^2}{4}(2\ln(1.7)-1) - \frac{0.1^2}{4}(2\ln(0.1)-1) = 0.0583.$$

The order of accuracy of the trapezoid rule is known and is equal to $p=2$. Let's use the partition ratio coefficient $m=2$. The maximum increase in the order of accuracy on the given table of values is indeed $q-p = [\log_m n] = [\log_2 4] = 2$. So, to increase the accuracy of the desired solution by two orders up to $q=4$, first of all, it is necessary to calculate $k=(q-p)+1=2+1=3$ numerical values of the definite integral on proportional partitions.

First, let's calculate numerically the definite integral using the trapezoid rule with the step h for the table of values:

$$I_t^h = \frac{h}{2}\sum_{i=1}^{4}(y_{i-1}+y_i) = \frac{0.4}{2}\cdot(y_0 + 2y_1 + 2y_2 + 2y_3 + y_4) = 0.0942.$$

The absolute interpolation error is:

$$\Delta_t^h = \left|I_t^h - I\right| = |0.0942 - 0.0583| = 0.0360.$$

Second, let's calculate numerically the definite integral using the trapezoid rule with the step $2h$ for the table of values:

$$I_t^{2h} = \frac{2h}{2}\sum_{i=1}^{2}(y_{2(i-1)}+y_{2i}) = \frac{0.8}{2}\cdot(y_0 + 2y_2 + y_4) = 0.1929.$$

The absolute interpolation error is:

$$\Delta_t^{2h} = \left|I_t^{2h} - I\right| = |0.1929 - 0.0583| = 0.1346.$$

Third, let's calculate numerically the definite integral using the trapezoid rule with the step 4*h* for the table of values:

$$I_t^{4h}=\frac{4h}{2}\sum_{i=1}^{1}(y_{4(i-1)}+y_{4i})=\frac{1.6}{2}\cdot(y_0+y_4)=0.5374\,.$$

The absolute interpolation error is:

$$\Delta_t^{4h}=\left|I_t^{4h}-I\right|=\left|0.5374-0.0583\right|=0.4792\,.$$

Obviously, the accuracy of the solution significantly decreases as far as the step increases.

Now let's run a series of $l=\frac{k(k-1)}{2}=3$ refinements.

An increase in the accuracy by an order of the numerical approximation I_t^h, according to the Runge-Romberg-Richardson rule, taking into account the known order of accuracy of the trapezoid rule $p=2$ and partitions ratio $m=2$, looks as follows:

$$\tilde{I}_t^h=I_t^h+\frac{I_t^h-I_t^{2h}}{m^p-1}=I_t^h+\frac{I_t^h-I_t^{2h}}{2^2-1}=0.0942+\frac{0.0942-0.1929}{3}=0.0614\,.$$

The absolute interpolation error is:

$$\tilde{\Delta}_t^h=\left|\tilde{I}_t^h-I\right|=\left|0.0614-0.0583\right|=0.0031\,.$$

An increase in the accuracy by an order of the numerical approximation I_t^{2h}, according to the Runge-Romberg-Richardson rule, looks as follows:

$$\tilde{I}_t^{2h}=I_t^{2h}+\frac{I_t^{2h}-I_t^{4h}}{m^p-1}=I_t^{2h}+\frac{I_t^{2h}-I_t^{4h}}{2^2-1}=0.1929+\frac{0.1929-0.5374}{3}=0.0780\,.$$

The absolute interpolation error is:

$$\tilde{\Delta}_t^{2h}=\left|\tilde{I}_t^{2h}-I\right|=\left|0.0780-0.0583\right|=0.0197\,.$$

Finally, an increase in the accuracy by an order for the numerical approximation $\tilde{I}_t^h$ also represents an increase in the accuracy by two orders for the numerical approximation I_t^h, and according to the Runge-Romberg-Richardson rule with respect to new order of accuracy $p=3$, looks as follows:

$$\tilde{\tilde{I}}_t^h=\tilde{I}_t^h+\frac{\tilde{I}_t^h-\tilde{I}_t^{2h}}{m^p-1}=\tilde{I}_t^h+\frac{\tilde{I}_t^h-\tilde{I}_t^{2h}}{2^3-1}=0.0614+\frac{0.0614-0.0780}{7}=0.0590\,.$$

The absolute interpolation error is:

$$\tilde{\tilde{\Delta}}_t^h = \left|\tilde{\tilde{I}}_t^h - I\right| = |0.0590 - 0.0583| = 0.0007\ .$$

It is easy to see that the use of such a chain of refinements according to the Runge-Romberg-Richardson rule allowed to reduce significantly the error of calculating the definite integral with the trapezoid rule by increasing the accuracy to the fourth order without resorting to the inclusion of additional interpolation nodes.

Calculation results are given in tables 13, 14.

Table 14

p	2	3	4
I_t^h	**0.0942**	**0.0614**	**0.0590**
I_t^{2h}	0.1929	0.0780	
I_t^{4h}	0.5374		

4.17. PROGRAM #18

Below is a proposed variant of the program algorithm for the numerical calculation of the value of a definite integral of a function on an interval with the trapezoid rule refined with the Runge-Romberg-Richardson rule.

```
ALGORITHM "Pyramid of refinements"
INPUT     f(), n, x[2*n-1], k
OUTPUT    i, j, m, p, q, y[2*n-1], d[k][k]
BEGIN
    CYCLE "Values" FOR i FROM 1 TO 2*n-1 BY 1
        y[i]:=f(x[i])
    m:=2, p:=2
    q:=p+LOGARITHM(n-1,m)
    IF (k>q-p+1) END
    CYCLE "Rows" FOR j FROM 1 TO k BY 1
```

```
                d[j][1]:=0
                q:=POWER(m,j-1)
                CYCLE "Summands" FOR i FROM 1 TO DIV(n-1,q) BY 1
                        d[j,1]:=d[j,1]+(y[2*q*i+1]+y[2*q*(i-1)+1])/2*
(x[2*q*i+1]-x[2*q*(i-1)+1])
        CYCLE "Columns" FOR j FROM 2 TO k BY 1
                q:=POWER(m,p+j-2)
                CYCLE "Rows" FOR i FROM 1 TO k-j+1 BY 1
                        d[i][j]:=d[i][j-1]+(d[i][j-1]-d[i+1][j-1])/(q-1)
        PRINT d
    END
```