Piecewise interpolation solution of ordinary differential equations with application to numerical modeling problems

Ya E Romm and G A Dzhanunts
Taganrog Branch of the Rostov State University of Economics, 48 Initsiativnaya Street, 347926 Taganrog, Russia

E-mail: romm@list.ru

Abstract. Piecewise interpolation approximation of functions of one real variable, derivatives and integrals is constructed with the help of Lagrange and Newton interpolation polynomials. The polynomials are transformed to the form of algebraic polynomials with numerical coefficients by means of restoring the polynomial coefficients by its roots. Formulas different from Vieta's formulas are applied. In the resulting form, the polynomials interpolate the right-hand sides of ordinary differential equations, the expression of the antiderivative ones is used to approximate the solution. Iterative refinement is performed. Error estimates and results of numerical experiments for stiff and non-stiff problems in physical, chemical models and other processes are presented.

1. Introduction and formulation of the question
An approximate solution of the Cauchy problem for ordinary differential equations (ODEs) is constructed on the basis of Lagrange and Newton interpolation polynomials. Both polynomials are equivalently transformed to the form of algebraic polynomials with numerical coefficients. In this case, the analytical expression of the derivative and the antiderivative is obtained. With the help of transformed polynomials, the right-hand sides of each equation of the ODE system are approximated. The antiderivative of the polynomial with the corresponding value of the constant – the coordinate of the approximate solution, is substituted into the right-hand side of the function. The process is iteratively repeated, and as a result, the solution is approximated with relatively high accuracy. The task of the message is to present a mathematical description of the method, its algorithmization and software implementation. The method is combined with piecewise interpolation. Numerical stability, error bounds, and time complexity for stiff and non-stiff problems are analyzed. A comparison is made with the methods of Runge-Kutta, Butcher and Dormand-Prince [1, 2]. The aim of the work is to show the distinctive quality of the proposed method, which consists in a limited accumulation of error while maintaining an acceptable labor intensity, which is in demand in problems of physical and chemical processes' modeling [3] and in problems of planetary astronomy [4]. A numerical experiment is described, and the possibility of reducing the error without automatic selection of parameters described in [5] is investigated. Together with the simplification of the method, it is necessary to reduce its labor intensity.
2. Invariant restoration of the polynomial coefficients by its roots

Consider a polynomial \( P_n(x) = \sum_{i=0}^{n} d_i x^i \) with the coefficient \( d_0 = 1 \) and roots \( x_0, x_1, \ldots, x_{n-1} \) so that

\[
\sum_{i=0}^{n} d_i x^i = \prod_{i=0}^{n-1} (x-x_i)
\]  

(1)

If \( P_1(x) = x - x_0 \) then it is considered that \( P_1(x) = d_1 x + d_0 \), where \( d_1 = 1, \ d_0 = -x_0 \). If the polynomial coefficients have already been calculated \( P_k(x) = \prod_{i=0}^{k-1} (x-x_i) \), \( k \geq 2 \) then \( P_k(x) = \prod_{i=0}^{k-1} (x-x_i) \) and \( P_1(x) = P_{k-1}(x) \cdot (x-x_{k-1}) \). Hence

\[
d_{k1} = d_{k-1}(x-x_{k-1}), \quad d_{k1} = d_{k-1}(x-x_{k-2}), \quad \ldots, \quad d_{k1} = d_{k-1}(x-x_{k-1}) \]

(2)

Here \( k = 1, 2, \ldots, k-1 \). When \( k=n \) the left-hand side parts of (2) will coincide with the desired values of the polynomial coefficients (1). The algorithm is stored for complex roots and coefficients, and is programmed in the form of a double cycle [6]. Relations (2) are used to transform interpolation polynomials. For the interpolation of the function \( y = f(x), \ x \in [a, b] \), the Lagrange polynomial has the form:

\[
P_n(x) = \sum_{j=0}^{n} f(x_j) \prod_{r=0}^{n} (x-x_r) / \prod_{r \neq j}^{n} (x_j-x_r),
\]  

(3)

where \( x_j \) are interpolation nodes, based on numerical experiment, the computer implementation gives a lower error in the case of equidistant nodes.

3. Transformation of the Lagrange interpolation polynomial

Let on the \( x \in [a, b] \) segment equidistant nodes for the interpolation polynomial (3) are taken:

\( x_i = a, x_n = b \). Hence

\[
\prod_{r=0}^{n} (x_j-x_r) = \prod_{r=0}^{n} (j-r)h, \quad P_n(x) = \sum_{j=0}^{n} f(x_j) \prod_{r=0}^{n} (x-x_r) / (j-r)h^{-1}.
\]  

A variable \( t = (x-x_0)h^{-1} \) is introduced. Then \( (x-x_r)h^{-1} = t - 1, \ldots, (x-x_r)h^{-1} = t - r, \ r \in [0, n] \), and

\[
P_n(x) = \sum_{j=0}^{n} f(x_j) \prod_{r=0}^{n} (t-r)/(j-r), \quad t = (x-x_0)h^{-1}.
\]  

(4)

A similar transformation is given in [7]. The difference will be in the conversion of the numerators from (4) to the form of polynomials with integer coefficients. As a result, analytical expression of the antiderivatives, derivatives and organization of the iterations in the right-hand sides of the ODE will become possible. Let the numerator of the fraction \( \prod_{r=0}^{n} (t-r)/(j-r) \) be written as

\[
P_n(t) = \prod_{i=0}^{n-1} (t-t_i),
\]  

(5)
where the role of the polynomial roots is played by the consecutive integers, among which \( j \):
\[
\tau = \begin{cases} 
  r, & r < j; \\
  r + 1, & r \geq j.
\end{cases}
\]
is omitted. According to the pattern (2) we have:
\[
P_n(t) = d_{o_j} + d_{i_j}t + d_{j_j}t^2 + ... + d_{n_j}t^n.
\]  

(6)

The coefficients of the polynomial (6) will be integers. They can be calculated a priori and stored in the computer memory as constants that do not depend on the interpolated function \( f(x) \), on the range and location of its argument, and this can be done for any \( j \) and all degrees of the \( n \) polynomial in a priori given boundaries. The denominator in (4) differs from the numerator in that it has \( t = j \) in it. As a result
\[
P_n(x) = \sum_{j=0}^{n} f(x_j) \left( d_{o_j} + d_{i_j} t + d_{j_j} t^2 + ... + d_{n_j} t^n \right) / (d_{o_j} + d_{i_j} + d_{j_j} j^2 + ... + d_{n_j} j^n),
\]

(7)

where \( t = (x - x_0)/h \) the numerator and denominator in (7) are conveniently calculated according to the Horner's method (scheme)
\[
P_n(t) = d_{o_j} + d_{i_j} t + d_{j_j} t^2 + ... + d_{n_j} t^n = (-d_{o_j} x + d_{i_j} j + d_{j_j} j^2 + ... + d_{n_j} j^n),
\]

In this notation
\[
P_n(t) = \sum_{j=0}^{n} f(x_j) P_n(t) / P_n(j), \quad t = (x - x_0)/h.
\]

(8)

If we collect the coefficients with equal degrees of terms then
\[
P_n(t) = \sum D_i t^i,
\]

(9)

where \( D_i \) is the result of this type. Further, the approximation \( f(x) \approx P_n(t) \) is performed with the help of (8), (9). Hence two varieties of the derivative approximation:
\[
f'(x) \approx P'_n(x) = h^{-1} \sum_{j=0}^{n} f(x_j) (d_{i_j} + 2d_{j_j} t + ... + (n-1) d_{n_j} t^{n-1}) / (d_{o_j} + d_{i_j} + d_{j_j} j^2 + ... + d_{n_j} j^n),
\]

(10)

\[
f'(x) \approx P'_n(x) = h^{-1} P'_n(t) = h^{-1} \sum_{i=1}^{n} \ell D_i t^{i-1}.
\]

(11)

In practice, (10) is somewhat more accurate than (11).

4. Piecewise interpolation

If (8) – (11) are applied on small subintervals of equal length with common partition boundaries
\[
[a, b] = \bigcup_{i=0}^{n-1} (a_i, b_i),
\]

(12)

then the accuracy of the approximation will increase; the function and integral will be approximated with accuracy to \( 10^{-19} - 10^{-20} \), the derivative – with accuracy \( 10^{-18} - 10^{-15} \) (extended format, Delphi). It is assumed below that \( b_i - a_i = (b-a)p^{-i} \), unless otherwise specified, then \( p = 2^k \), \( k \) is an integer.

Transformations (8), (9) formally coincide with each other and with the polynomial (3), provided that on one and the same segment the nodes are equal, the degrees of the polynomials are the same, and the
same function is interpolated. Under the same conditions, all three polynomials coincide with the Newton's interpolation polynomial [7]. The relation [8] is valid for \( k \geq 0 \):

\[
[a, b] = \bigcup_{i=0}^{n-1} [a_i, b_i], \quad |f(x) - \Psi_{i,n}(t)| \leq c \ 2^{-k(n+1)} h^{n+1} \quad \forall i = 0, 2^k - 1, \ \forall x \in [a_i, b_i],
\] (13)

where it is assumed to be that \( f(x) \) is defined and continuously differentiable at \([a, b]\) \( n+1 \) times. Here \( \Psi_{i,n}(t) \) is the Newton interpolation polynomial for forward interpolation on the subinterval \([a_i, b_i]\), \( t = (x-a_i)h^{-1} \), \( h \) is the interpolation step of the polynomial \( \Psi_{o,n}(t) \) on \([a, b]\) (if \( k = 0 \)), \( c = \max_{[a, b]} |f^{(n+1)}(x)| \), \( n_o = \text{const.}, \ c = \text{const.} \). From (13) the sequence of polynomials \( \Psi_{i,n}(t) \) converges uniformly to \( f(x) \) on \([a, b]\) if \( k \rightarrow \infty \). Both these statements and (13) as well are preserved for the considered transformations of the Lagrange interpolation polynomial.

5. Approximate calculation of integrals

Directly from (8), (9), (12) the formulas of approximate calculation of integrals follow. With partitioning in (12)

\[
\int_a^b f(x) \, dx = \sum_{i=0}^{n-1} \int_{a_i}^{b_i} f(x) \, dx, \quad \int_{a_i}^{b_i} f(x) \, dx \approx \int_{a_i}^{b_i} P_{i,n}(x) \, dx,
\] (14)

where \( P_{i,n}(x) \) is polynomial (7) constructed on the subinterval \([a_i, b_i]\):

\[
P_{i,n}(x) = \sum_{j=0}^{n} f(x_{i,j}) (d_{o,j} + d_{j,1} t + d_{j,2} t^2 + \ldots + d_{j,n} t^n) / (d_{o,j} + d_{j,1} j + d_{j,2} j^2 + \ldots + d_{j,n} j^n),
\] (15)

Hence

\[
\int_{a_i}^{b_i} P_{i,n}(x) \, dx = \int_0^h \left( d_{o,j} + d_{j,1} t + d_{j,2} t^2 + \ldots + d_{j,n} t^n \right) dt / (d_{o,j} + d_{j,1} j + d_{j,2} j^2 + \ldots + d_{j,n} j^n),
\]

or

\[
\int_{a_i}^{b_i} P_{i,n}(x) \, dx = (b_i - a_i) \times

\sum_{j=0}^{n} f(x_{i,j}) (d_{o,j} + d_{j,1} 2^{-1} n + d_{j,2} 3^{-1} n^2 + \ldots + d_{j,n} (n+1)^{-1} n^n) / (d_{o,j} + d_{j,1} j + d_{j,2} j^2 + \ldots + d_{j,n} j^n),
\] (16)

Adding (16) over all subintervals implies

\[
\int_a^b f(x) \, dx \approx (b-a) p^{-1} \times

\sum_{j=0}^{n} \sum_{i=0}^{n} f(x_{i,j}) (d_{o,j} + d_{j,1} 2^{-1} n + d_{j,2} 3^{-1} n^2 + \ldots + d_{j,n} (n+1)^{-1} n^n) / (d_{o,j} + d_{j,1} j + d_{j,2} j^2 + \ldots + d_{j,n} j^n).
\] (17)

If on each segment in the right-hand side (14) \( f(x) \) approximates with the estimate (13), also valid for the considered transformation of the Lagrange polynomial, where \( h = (b-a)n^{-1} \), \( b_i - a_i = (b-a)2^{-k} \), then

\[
\int_{a_i}^{b_i} f(x) \, dx - \int_{a_i}^{b_i} P_{i,n}(x) \, dx \leq c \ 2^{-k(n+1)} ((b-a)n^{-1})^{n+1} (b-a)2^{-k}. \]

Hence
\[
\int_a^b f(x) \, dx - \sum_{i=0}^{n-1} P_{n_i}(x) \, dx \leq \sum_{i=0}^{n-1} \left( f(x) \, dx - P_{n_i}(x) \, dx \right) \leq 2^{-(n+1)} ((b-a)^{-1})^{n+1} (b-a)^{-2}.
\]

Finally,
\[
\int_a^b f(x) \, dx - \sum_{i=0}^{n-1} P_{n_i}(x) \, dx \leq 2^{-(n+1)} ((b-a)^{-1})^{n+1} (b-a).
\]

The estimation (18) takes place under the conditions described above, under which (13) is fulfilled. Under these conditions \( 2^{-(n+1)} ((b-a)^{-1})^{n+1} (b-a) \to 0, \) if \( k \to \infty \). Formula (17) can be interpreted as an analytical version of the Newton-Cotes formulas [7].

6. Piecewise interpolation solution of the Cauchy problem for ODEs with iterative refinement

Let the Cauchy problem be considered

\[
y' = f(x, y), \quad y(x_0) = y_0,
\]

in the domain \( R = \{ a \leq x \leq b; |y - y_0| \leq B; B = \text{const} \} \), where the function \( f(x, y) \) is defined, it is continuously differentiable (at points \( a \) – in the right-hand side, \( b \) – in the left-hand side) and satisfies the Lipschitz condition:

\[
|f(x, y) - f(x, \bar{y})| \leq L |y - \bar{y}|, \quad L = \text{const}, \quad \forall (x, y), (x, \bar{y}) \in R.
\]

It is assumed that in \( R \) the solution of problem (19) exists and is unique. For simplicity of notation, \( a, b \) are the same as in (12). To interpolate the right-hand side of (19), an approximate value \( y \) is substituted in \( f(x, y) \). First, \( y \) is substituted for \( y_0 \). The function \( f(x, y_0) \) is approximated by polynomials of the form in (8), (9) according to the above scheme, as if \( f(x) \) coincides with \( f(x, y_0) \).

For fixed \( n \) and \( k \) on the segment \([a_i, b_i]\), \( i = 0 \), then, similarly, for \( i = 1, 2, \ldots \), iterative refinement is performed. Let, for definiteness, on \([a_i, b_i]\) a polynomial of the form (9), denoted by \( P_{n_i}(t) = \sum_{l=0}^{n_i} D_{l} t^l \), \( f(x, y_0) \equiv P_{n_i}(t) \), \( t = (x-a_i) h_i^{-1} \), where \( h_i \) is the interpolation step on \([a_i, b_i]\), be constructed. The antiderivative \( P_{0(n_i+1)}(x) = y_{0(i-1)} + \int_0^x f(t) \, dt \), equal to

\[
y_{0(i-1)} + h_i \sum_{l=0}^{n_i} D_{l} (\ell + 1) t^{\ell+1},
\]

is taken as an approximation of the solution: \( y(x) \approx P_{0(n_i+1)}(x) \). Further, it is assumed that \( f(x, y) \approx f(x, P_{0(n_i+1)}(x)) \), and for the same \( n \), on the same segment, an interpolation polynomial of the form (9) is again constructed: \( P_{1n}(t) \equiv f(x, P_{0(n_i+1)}(x)) \), \( t = (x-a_i) h_i^{-1} \). Again, the antiderivative with the same value of the constant

\[
P_{1n}(x) = y_{0(i-1)} + h_i \int_0^x P_{1n}(t) \, dt
\]

is taken, substituted in the right-hand side, \( f(x, y) \approx f(x, P_{1n}(x)) \), which is then similarly interpolated: \( P_{2n}(t) \equiv f(x, P_{1n}(x)) \). In fact, iterations \( P_{1n}(t) \approx f(x, P_{1n}^{(\ell-1)}(x)) \), \( t = (x-a_i) h_i^{-1} \), \( P_{1n}^{(\ell)}(x) = y_{0(i-1)} + h_i \int_0^x P_{1n}^{(\ell)}(t) \, dt \), \( \ell = 1, 2, \ldots \), continue up to a given boundary \( \ell \leq q = \text{const} \), abstractly their number is not limited. The formula \( P_{q(n_i+1)}(h_{i-1}) \) was taken for the above value \( y_{0(i-1)} \). At the end of the iterations on \([a_i, b_i]\), the
transition to \([a_i, b_i]\) is performed, where the value \(P_{(i+1)}^{(i)}(b_i)\) is taken for \(y_{b_i}\). Error estimates will be performed below under additional assumptions. Given that for equidistant nodes on the same segment, when interpolating the same function, the Lagrange and Newton interpolation polynomials coincide to the same degree in all the considered forms, to simplify the notation, it is assumed that the interpolation is performed by a polynomial of the form \((3)\). More precisely, the interpolation is assumed without transition to a variable \(t\), while the polynomial has fully reduced numerical coefficients: 
\[
P_n(x) = \sum_{i=0}^{n} d_i x^i, \quad x \in [a_i, b_i].
\]

The interpolation error on \([a_i, b_i]\) is estimated from \((13)\), the right-hand side of this estimate is denoted by 
\[
\int_a^{b_i} f(x,y) \, dx, \quad y(a_i) = y_{b_i},
\]
along with 
\[
P^{(i)}_{(i+1)}(x) = y_{b_i} \int_{a_i}^{x} P_n(x) \, dx,
\]
where \((20)\) is taken into account, so that no replacement of the variable is required. First, it is assumed that on a sufficiently large segment of the sequence of numbers 
\[
0 < c_{i\bar{k}} \leq \max_{[a_i, b_i]} |y(x) - P^{(i)}_{(i+1)}(x)|, \quad \ell = 0, 1, \ldots,
\]
where 
\[
P^{(i)}_{(i+1)}(x) = P^{(i)}_{(i+1)}(x)\]
taking into account the same initial values of the exact solution and its approximation, the absolute error of the \(\ell\) iteration will take the form
\[
\left| y(x) - P^{(i)}_{(i+1)}(x) \right| = \left| \int_{a_i}^{x} f(x,y) - P^{(i)}_{(i+1)}(x) \, dx \right|,
\]
hence
\[
\left| y(x) - P^{(i)}_{(i+1)}(x) \right| \leq \int_{a_i}^{x} \left| f(x,y) - P^{(i)}_{(i+1)}(x) \right| \, dx, \quad \ell = 0, 1, 2, \ldots
\]
(22)

By construction 
\[
P^{(i)}_{(i+1)}(x) \equiv f(x, P^{(i)}_{(i+1)}(x)),
\]
the interpolation error is denoted by \(\bar{c}_{i\bar{k}}\), in this notation
\[
f(x, P^{(i)}_{(i+1)}(x)) = P^{(i)}_{(i+1)}(x) + \bar{c}_{i\bar{k}}.
\]
Substitution in \((22)\) implies
\[
\left| y(x) - P^{(i)}_{(i+1)}(x) \right| \leq \int_{a_i}^{x} \left| f(x,y) - f(x, P^{(i-1)}_{(i+1)}(x)) - \bar{c}_{i\bar{k}} \right| \, dx \quad \forall x \in [a_i, b_i].
\]
(23)

According to \((15)\)
\[
\left| \bar{c}_{i\bar{k}} \right| \leq c_{i\bar{k}},
\]
therefore
\[
\left| y(x) - P^{(i)}_{(i+1)}(x) \right| \leq \int_{a_i}^{x} \left| f(x,y) - f(x, P^{(i-1)}_{(i+1)}(x)) \right| + c_{i\bar{k}} \, dx.
\]
Hence, provided that \((21)\) is true for the index \(\ell - 1\), \(\forall x \in [a_i, b_i]\) holds the inequality
\[
\left| y(x) - P_{(i)\ell}^{(f)}(x) \right| \leq \int_{a_i}^{b_i} \max_{[a_j, b_j]} f(x, y) - f(x, P_{(i)\ell}^{(f-1)}(x)) \, dx + \max_{[a_j, b_j]} \left| y(x) - P_{(i)\ell}^{(f-1)}(x) \right| \, dx.
\]

Applying the Lipschitz condition,

\[
\left| y(x) - P_{(i)\ell}^{(f)}(x) \right| \leq \int_{a_i}^{b_i} L \max_{[a_j, b_j]} y(x) - P_{(i)\ell}^{(f-1)}(x) + \max_{[a_j, b_j]} y(x) - P_{(i)\ell}^{(f-1)}(x) \, dx \forall x \in [a_i, b_i],
\]

or,

\[
\left| y(x) - P_{(i)\ell}^{(f)}(x) \right| \leq \int_{a_i}^{b_i} \max_{[a_j, b_j]} y(x) - P_{(i)\ell}^{(f-1)}(x) \, dx \forall x \in [a_i, b_i], \quad N = L + 1.
\]

Due to the \( x \in [a_i, b_i] \) arbitrariness in both parts of the inequality, you can shift to the maximum:

\[
\max_{[a_j, b_j]} \left| y(x) - P_{(i)\ell}^{(f)}(x) \right| \leq \max_{[a_j, b_j]} \left( y(x) - P_{(i)\ell}^{(f-1)}(x) \right) |(b_i - a_i) \, \ell = 0, 1, 2, \ldots.
\]

Let the left-hand side of inequality (25) be denoted by \( \epsilon_{\ell} \), then

\[
\epsilon_{\ell} \leq N \epsilon_{\ell-1} (b_i - a_i).
\]

In (26) \( \epsilon_{\ell} = \max_{[a_j, b_j]} \left| y(x) - P_{(i)\ell}^{(f)}(x) \right| \), and \( \epsilon_{\ell} \leq N \max_{[a_j, b_j]} y(x) - P_{(i)\ell}^{(f-1)}(x) \) \((b_i - a_i)\). By construction \( f(x, y_{0(i-1)}) = P_{\alpha}(x) + c_{\alpha} \), therefore, taking into account (22) if \( \ell = 0 \) and the above transformations,

\[
\left| y(x) - P_{(i)\ell}^{(f)}(x) \right| \leq \int_{a_i}^{b_i} f(x, y_{0(i-1)}) - f_{\alpha} \, dx \forall x \in [a_i, b_i].
\]

Hence \( \max_{[a_j, b_j]} \left| y(x) - P_{(i)\ell}^{(f)}(x) \right| \leq 2 c_{\alpha} \int_{a_i}^{b_i} dx \), where \( c_{\alpha} = \alpha_{a_i} = 2 - \frac{1}{2^\ell} \). As a result,

\[
\epsilon_{\ell} \leq (b_i - a_i) \times \alpha_{a_i} = 2 c_{\alpha}.
\]

From (26), (27)

\[
\epsilon_{\ell} \leq N \times (b_i - a_i)^2 \times \alpha_{a_i}.
\]

From (26), (28) \( \epsilon_{\ell} \leq N^2 (b_i - a_i)^2 \times \alpha_{a_i} \). By induction

\[
\epsilon_{\ell} \leq \alpha_{a_i} N^{-1} ((b_i - a_i) N)^{\ell+1}, \quad \ell = 0, 1, 2, \ldots.
\]

The inequality (29) is a consequence of (25) and is true for those consecutive ones \( \ell = 0, 1, 2, \ldots \) for which (21) is not violated. Without detracting from generality, we can assume that \( (b_i - a_i) N = 2^{-\ell} (b_i - a_i) N < 1 \), then in (29) \( \epsilon_{\ell} \rightarrow 0, \ell \rightarrow \infty \). Therefore, inequality (21) will prove to be violated. Violation of (21) means that

\[
0 < \max_{[a_j, b_j]} \left| y(x) - P_{(i)\ell}^{(f)}(x) \right| \leq c_{\ell}, \quad \ell = \ell_0 + 1.
\]
Let $\ell = \ell_0 + 1$ be the first element of the sequence in which inequality (21) is violated and (30) is satisfied. At the same time, (25) and (29) are still saved for the number $\ell = \ell_0 + 1$. In these conditions, from (23) with the application of the Lipschitz condition, it follows that

$$
\left| y(x) - P^{(k+2)}_{(i+1)(i+1)}(x) \right| \leq \int_{a_i} \left( L \max_{[a_i, b_i]} \left| y(x) - P^{(k)}_{(i+1)(i+1)}(x) \right| + c_{ik} \right) \, dx \quad \forall x \in [a_i, b_i]. \quad (31)
$$

Hence, taking into account the implementation of (29) for $\ell = \ell_0 + 1$ and the fact that

$$
\max_{[a_i, b_i]} \left| y(x) - P^{(k+1)}_{(i+1)(i+1)}(x) \right| < c_{ik} < \max_{[a_i, b_i]} \left| y(x) - P^{(k)}_{(i+1)(i+1)}(x) \right|,
$$

the integrand in (31) can be replaced by the corresponding expression in (24):

$$
\left| y(x) - P^{(k+2)}_{(i+1)(i+1)}(x) \right| \leq \int_{a_i} \left( L \max_{[a_i, b_i]} \left| y(x) - P^{(k+1)}_{(i+1)(i+1)}(x) \right| + \max_{[a_i, b_i]} \left| y(x) - P^{(k)}_{(i+1)(i+1)}(x) \right| \right) \, dx \quad \forall x \in [a_i, b_i]. \quad (32)
$$

But for the right-hand side (32), after shifting to the maximum in both parts of the inequality, the estimate (29) is preserved, at which the left-hand side of the inequality (29) already violates (31) and does not exceed $c_{ik}$. Therefore, the left-hand side of (32) will not exceed $c_{ik}$:

$$
\max_{[a_i, b_i]} \left| y(x) - P^{(k+2)}_{(i+1)(i+1)}(x) \right| < c_{ik} < \max_{[a_i, b_i]} \left| y(x) - P^{(k+1)}_{(i+1)(i+1)}(x) \right| < \max_{[a_i, b_i]} \left| y(x) - P^{(k)}_{(i+1)(i+1)}(x) \right|. \quad (33)
$$

By analogy,

$$
\left| y(x) - P^{(k+3)}_{(i+1)(i+1)}(x) \right| \leq \int_{a_i} \left( L \max_{[a_i, b_i]} \left| y(x) - P^{(k+2)}_{(i+1)(i+1)}(x) \right| + c_{ik} \right) \, dx \quad \forall x \in [a_i, b_i], \quad (34)
$$

and

$$
\max_{[a_i, b_i]} \left| y(x) - P^{(k+3)}_{(i+1)(i+1)}(x) \right| < c_{ik} < \max_{[a_i, b_i]} \left| y(x) - P^{(k+2)}_{(i+1)(i+1)}(x) \right| < \max_{[a_i, b_i]} \left| y(x) - P^{(k+1)}_{(i+1)(i+1)}(x) \right|, \quad (35)
$$

Therefore, the integrand on the right-hand side of (34) can be replaced by the corresponding integrand from (24), which is the same as the integrand from (32):

$$
\left| y(x) - P^{(k+3)}_{(i+1)(i+1)}(x) \right| \leq \int_{a_i} \left( L \max_{[a_i, b_i]} \left| y(x) - P^{(k+2)}_{(i+1)(i+1)}(x) \right| + \max_{[a_i, b_i]} \left| y(x) - P^{(k+1)}_{(i+1)(i+1)}(x) \right| \right) \, dx \quad \forall x \in [a_i, b_i].
$$

Repeating the previous reasoning entails

$$
\max_{[a_i, b_i]} \left| y(x) - P^{(k+3)}_{(i+1)(i+1)}(x) \right| < c_{ik}. \quad (35)
$$

By obvious induction, inequalities (33) and (35) will become inequality

$$
\max_{[a_i, b_i]} \left| y(x) - P^{(k+r)}_{(i+1)(i+1)}(x) \right| < c_{ik},
$$

where $r$ is an arbitrary natural number.

From the stated above we derive Lemma 1.
Lemma 1. Suppose that in the considered conditions including a performance of condition (13) applied to the right-hand side of (19) and condition (21) and the assumption $2^{-k} (b-a) N < 1$, a piecewise interpolation is performed with iterative refinement of the (19) problem solution. Then, for an arbitrary segment $[a_i, b_i]$ from (12), there is a number $r_0$ such that the iterative refinement will satisfy the inequality

$$\max_{[a_i, b_i]} \left| y(x) - P_{(i) (n+1)}^{(r)}(x) \right| < c_{ik} \ \forall \ r \geq r_0, \quad (36)$$

For numbers from (36), (23) is preserved, whence by applying the Lipschitz condition

$$\left| y(x) - P_{(i) (n+1)}^{(r)}(x) \right| \leq \int_{a_i}^{x} \left( L \max_{[a_i, b_i]} \left| y(x) - P_{(i) (n+1)}^{(r)}(x) \right| + c_{ik} \right) dx \ \forall \ x \in [a_i, b_i].$$

Due to (36)

$$\left| y(x) - P_{(i) (n+1)}^{(r+1)}(x) \right| \leq N c_{ik} (x - a_i) \ \forall \ x \in [a_i, b_i].$$

Thus,

$$\max_{[a_i, b_i]} \left| y(x) - P_{(i) (n+1)}^{(r+1)}(x) \right| \leq N c_{ik} (b_i - a_i) \ \forall \ r \geq r_0, \ N = L + 1,$$

or

$$\max_{[a_i, b_i]} \left| y(x) - P_{(i) (n+1)}^{(r+1)}(x) \right| \leq N c_{ik} (b - a) 2^{-k} \ \forall \ r \geq r_0, \ N = L + 1. \quad (37)$$

Theorem 1. Under the conditions of Lemma 1, the solution of problem (19) on an arbitrary segment $[a_i, b_i]$ from (12), (37) with an estimate of the absolute error is approximated. This means reducing of the piecewise interpolation error on a separate subinterval proportionally where $(b-a) 2^{-k}$ due to iterative refinement.

Corollary 1. Under the same conditions, the rate of convergence to the estimate (37) is determined by the geometric progression from (29).

Corollary 2. Under the conditions of Theorem 1, the absolute error of the solution of problem (19) for the entire segment of solution (12) is estimated from the ratio

$$\sum_{j=0}^{k+1} \max_{[a_i, b_i]} \left| y(x) - P_{(i) (n+1)}^{(r+1)}(x) \right| \leq \sum_{j=0}^{k+1} N c_{ik} (b - a) 2^{-j},$$

or

$$\sum_{j=0}^{k+1} \max_{[a_i, b_i]} \left| y(x) - P_{(i) (n+1)}^{(r+1)}(x) \right| \leq N(b - a)c_{ik}. \quad (38)$$

The estimate (38) means that the application of iterative refinement allows us to approximate the solution on the entire segment (12) with an absolute error of piecewise interpolation on one, separately taken, subinterval from (12), up to a constant multiplier $N(b-a)$.

Substituting in (38) $c_{ik}$ from (13) implies

$$\sum_{j=0}^{k+1} \max_{[a_i, b_i]} \left| y(x) - P_{(i) (n+1)}^{(r+1)}(x) \right| \leq N(b - a)((b - a)/n)^{k+1} c 2^{-k(n+1)}. \quad (39)$$
Let \( \forall \varepsilon > 0 \) be given. In (39), you can specify such \( k_0 = k_0(\varepsilon) \) that the right-hand side will not exceed \( \varepsilon \) when \( \forall k > k_0 \). That is \( k_0 = (n+1)^{-1} \log_2(N(b-a)(b-a)/n)^{n+1}c^{-1} \).

**Theorem 2.** Under the conditions of Lemma 1, the absolute error of the (19) problem solution over the entire segment of the solution from (12) is estimated from (39). In this case if \( \forall \varepsilon > 0 \) the following relation

\[
\sum_{i=0}^{k-1} \max_{[x_i,x_{i+1}]} y(x) - P_{i+1}^{(r+1)}(x) \leq \varepsilon \forall k > (n+1)^{-1} \log_2(N(b-a)(b-a)/n)^{n+1}c^{-1}, \forall x \in [a, b]
\]

is correct, which means uniform convergence of the method if \( k \to \infty \).

The above arguments and estimates are transferred to the case of the Newton interpolation polynomial, represented in the form of an algebraic polynomial with numerical coefficients.

The theorem and corollaries give a formal estimate of the error under abstract conditions involving the existence of \( n+1 \) derivative of the (19) function on the right-hand side. However, interpolation itself is possible under the broadest conditions, so the method can always be applied in conditions of unique existence. In practice, much is determined not only by the smallness of subinterval in (39), but by a view of the right-hand side of (19), the stability of the solution in the sense of Lyapunov, stiffness or non-stiffness class of problems. Nevertheless, in all such experimentally considered cases, the proposed method has a lower error than the well-known methods, differing in a limited accumulation of error with an increase in the interval of the approximate solution.

### 7. Numerical Experiment

Part of the experiment is given on the examples of problems with analytical solutions. The results of chemical processes modeling and problems of celestial mechanics are also discussed. The experiment was conducted using a PC based on an Intel(R) Core(TM) i5-2500.

**Example 1.** The problem \( y' = \cos(x+y), \ y(0)=0 \), has a solution \( y=-x+2\arctg(x) \). The absolute error of approximation in 100 uniformly distributed points of the segment \([0, 512]\) by differential and piecewise interpolation (PP) methods is given in table 1 along with the number of calls \((fc)\) to the right-hand side.

| Method          | Steps | \( h \) | \( fc \) | Error          |
|-----------------|-------|---------|---------|----------------|
| Runge-Kutta_4   |       | \( h = 10^{-3} \) | \( 2 \times 10^6 \) | 1.3 \times 10^{-15} |
|                 |       |         |         | 3.4 \times 10^{-16} |
|                 |       |         |         | ... |
|                 |       |         |         | 3.1 \times 10^{-16} |
|                 |       |         |         | 8.1 \times 10^{-16} |
| Butcher_6       |       | \( h = 10^{-2} \) | \( 3.5 \times 10^5 \) | 2.7 \times 10^{-16} |
|                 |       |         |         | 6.8 \times 10^{-17} |
|                 |       |         |         | ... |
|                 |       |         |         | 1.8 \times 10^{-15} |
|                 |       |         |         | 2.1 \times 10^{-15} |
| Dormand-Prince_8|       | \( h = 10^{-2} \) | \( 6.5 \times 10^5 \) | 1.5 \times 10^{-18} |
|                 |       |         |         | 7.4 \times 10^{-18} |
|                 |       |         |         | ... |
|                 |       |         |         | 4.5 \times 10^{-15} |
|                 |       |         |         | 3.8 \times 10^{-15} |
| PP              |       | \( h = 2.3 \times 10^{-2} \) | \( 1.8 \times 10^5 \) | 1.1 \times 10^{-18} |
|                 |       |         |         | 0.0 |
|                 |       |         |         | ... |
|                 |       |         |         | 2.8 \times 10^{-17} |
|                 |       |         |         | 5.6 \times 10^{-17} |

The error boundary \( 10^{-15} \) corresponds to the methods of the 4th (Runge-Kutta_4), 6th (Butcher_6) and 8th (Dormand-Prince_8) orders. The values of the steps for each difference method are chosen for the purpose of the smallest error. The Butcher method corresponds to the number of calls to the right-hand side of the function \( fc = 3.5 \times 10^5 \). The piecewise interpolation solution with parameters \( b_i-a_i=0.345 \), \( n=15 \), \( \ell=13 \) is characterized by the order of error \( 10^{-17} \) if \( fc=183344 \).

**Example 2.** The system \( y'_1 = x + 2y_1 / x - \sqrt{y_1}, \ y'_2 = 2\sqrt{y_1} \), if \( y_1(1)=2, \ y_2(1)=4 \) has a solution \( y_1 = x(1+x), \ y_2 = (1+x)^2 \). The canonical norms of the vector of absolute errors of the components on the segment \([1, 513]\) are presented in table 2 along with \( fc \).
Table 2. Error and number of calls to the right-hand side when solving the problem of Example 2

| Method               | $h = 10^{-3}$ | $fc = 2 \times 10^4$ | $2.1 \times 10^{-17}$ | $1.1 \times 10^{-16}$ | $\ldots$ | $1.6 \times 10^{-13}$ | $1.4 \times 10^{-13}$ |
|----------------------|---------------|----------------------|-----------------------|-----------------------|----------|-----------------------|-----------------------|
| Runge-Kutta_4        | $h = 10^{-4}$ | $fc = 3.6 \times 10^3$| $3.5 \times 10^{-16}$ | $1.4 \times 10^{-16}$ | $\ldots$ | $1.7 \times 10^{-13}$ | $4.8 \times 10^{-13}$ |
| Butcher_6            | $h = 10^{-3}$ | $fc = 6.7 \times 10^6$| $8.7 \times 10^{-17}$ | $3.6 \times 10^{-16}$ | $\ldots$ | $2.4 \times 10^{-13}$ | $4.3 \times 10^{-13}$ |
| Dorman-Prince_8      | $h = 8 \times 10^{-2}$ | $fc = 5.6 \times 10^4$| $0.0$                 | $0.0$                 | $\ldots$ | $0.0$                 | $0.0$                 |

With the same order of error among the difference methods, the Dorman-Prince method is characterized by the value $fc = 6656000$. Piecewise interpolation approximation with the parameters: $b_i - a_i = 0.25$, $n = 3$, $\ell = 20$, in most test points gives the values of "zero" errors in extended precision, with an increase in the number of test points, the values $10^{-18} - 10^{-16}$ occur. At the same time $fc = 56028$.

**Example 3.** In [1] on a series of test problems, a comparison of the computational qualities of the most effective methods for high-precision solution of non-stiff Cauchy problems is presented. Typical results are given by a two-body problem:

$$y'_1 = y_3, \quad y'_2 = y_4, \quad y'_3 = -y_1(y_2^2 + y_4^2)^{3/2}, \quad y'_4 = -y_2(y_2^2 + y_4^2)^{3/2}, \quad y_1(0) = 0.5, \quad y_2(0) = 0, \quad y_3(0) = 0, \quad y_4(0) = \sqrt{3}. \quad (40)$$

For $[0, 6\pi]$ the Dorman-Prince method of the 8th order gives the solution of the problem (40) with an error of the order $10^{-14}$ if $fc \approx 3.9 \times 10^4$, the extrapolation program ODEX on the same segment reaches an error of the order $10^{-13}$ if $fc \approx 3.6 \times 10^4$ [1]. The Gauss-Everhart integrator of the 19th order – GAUSS_32 [9, 10], implemented in the Delphi environment, reaches the lowest error limit, of the order $10^{-16}$, on the same segment, among the studied methods of numerical approximation. The method is adapted for solving problems of celestial mechanics, in particular, the mechanism for choosing the size of the integration step is implemented taking into account the specific nature of the planar two-body problem [10]. The piecewise interpolation solution of the problem (40) is characterized by the error bound of an order $10^{-17}$ if $fc = 275924$ (method parameters: $b_i - a_i = 2\pi/1024$, $n = 10$, $\ell = 20$). The error of the order $10^{-15}$ is achieved if $fc = 35334$ (method parameters: $b_i - a_i = 2\pi/64$, $n = 12$, $\ell = 20$). Similar results are obtained for solving other non-stiff problems.

**Example 4.** The periodic Belousov-Zhabotinsky reaction is modelled by a stiff system [3]:

$$y'_1 = 77.27(y_2 + y_1(1 - 8.375 \times 10^{-6} y_1 y_2)), \quad y'_2 = 77.27^{-1}(y_1 - y_2(1 - y_1)), \quad y'_3 = 0.161(y_1 - y_3).$$

The results of piecewise interpolation approximation of the solution for the initial data $y_1(0) = 4$, $y_2(0) = 1.1$, $y_3(0) = 4$ on the segment $[0, 512]$ with variations in the degree of the interpolation polynomial and the number of subintervals of the partition are presented in [5]. Fixing the method parameters and excluding additional clarifying procedures of the program allowed to reduce the time for solving the problem with the error bound $10^{-14}$ from $t \approx 11\text{min}$ to $t = 4\text{min} 14\text{s} 677\text{ms}$ ($b_i - a_i = 0.01/512$, $n = 4$, $\ell = 5$).

**8. Conclusion**

An approximate solution of the Cauchy problem for ODE is constructed by applying piecewise interpolation based on Lagrange and Newton interpolation polynomials, equivalently transformed to the form of algebraic polynomials with numerical coefficients. With the help of transformed polynomials, the right-hand sides of the equations of the system are approximated. The antiderivatives of polynomials are the coordinates of the approximate solutions. The process is iteratively refined, and
as a result, the solution is approximated with relatively high accuracy. The numerical experiment shows the computational stability of the solution of stiff and non-stiff problems within acceptable limits of labor intensity.

References
[1] Hairer E, Nørsett S P and Wanner G 1993 Solving Ordinary Differential Equations I: Nonstiff Problems (Berlin, Heidelberg: Springer-Verlag) p 528.
[2] Butcher J C 2016 Numerical Methods for Ordinary Differential Equations (Chichester: John Wiley & Sons, Ltd) p 519.
[3] Hairer E and Wanner G Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems (Berlin, Heidelberg: Springer-Verlag) p 614.
[4] Bordovitsyna T V 1984 Modern numerical methods in problems of celestial mechanics (Moscow: Nauka) p 136.
[5] Dzhanunts G A and Romm Ya E 2017 The varying piecewise interpolation solution of the Cauchy problem for ordinary differential equations with iterative refinement Computational Mathematics and Mathematical Physics Journal 57 (10) 1616–34.
[6] Romm Ya E 2007 Sorting-based localization and stable computation of zeros of a polynomial. II Cybern. Syst. Analysis 43 (2) 291–302.
[7] Berezin I S and Zhidkov N P 1966 Calculation methods vol1 (Moscow: Nauka) p 632.
[8] Romm Ya E and Dzhanunts G A 2013 The computer method of variable piecewise polynomial approximation of functions and solutions of ordinary differential equations Cybern. Syst. Analysis 49 (3) 169–89.
[9] Everhart E 1974 Implicit single sequence methods for integrating orbits Celest. Mech. 10 35–55.
[10] Avdyushev V A 2010 Gauss-Everhart Integrator Computational technologies 15 (4) 31–46.