ISSN 2070-0482, Mathematical Models and Computer Simulations, 2017, Vol. 9, No. 4, pp. 498–510. © Pleiades Publishing, Ltd., 2017.
Original Russian Text © L.M. Skvortsov, 2017, published in Matematicheskoe Modelirovanie, 2017, Vol. 29, No. 1, pp. 3–19.

On Implicit Runge–Kutta Methods Obtained as a Result of the Inversion of Explicit Methods

L. M. Skvortsov

Bauman State Technical University, Moscow, 105005 Russia

e-mail: lm_skvo@rambler.ru

Received November 26, 2015

Abstract—We consider methods that are the inverse of the explicit Runge–Kutta methods. Such methods have some advantages, while their disadvantage is the low (first) stage order. This reduces the accuracy and the real order in solving stiff and differential-algebraic equations. New methods possessing properties of methods of a higher stage order are proposed. The results of the numerical experiments show that the proposed methods allow us to avoid reducing the order.

Keywords: inverse Runge–Kutta methods, stiff equations, differential-algebraic equations, order reduction phenomenon

DOI: 10.1134/S2070048217040123

INTRODUCTION

Consider the Cauchy problem for the system of ordinary differential equations (ODEs)

$$\mathbf{y}' = \mathbf{f}(t,\mathbf{y}), \quad \mathbf{y}(t_0) = \mathbf{y}_0, \quad t_0 \le t \le t_0 + T, \tag{1}$$

where $\mathbf{y}$ is a vector of variables, $\mathbf{f}$ is a vector function, and t is an independent variable. One step of the numerical solution of system (1) by the s-stage Runge–Kutta method is performed in accordance with the formulas

$$\mathbf{y}_1 = \mathbf{y}_0 + h\sum_{i=1}^{s} b_i\mathbf{F}_i, \quad \mathbf{F}_i = \mathbf{f}(t_0 + c_ih, \mathbf{Y}_i), \quad \mathbf{Y}_i = \mathbf{y}_0 + h\sum_{j=1}^{s} a_{ij}\mathbf{F}_j, \quad i = 1,\ldots,s. \tag{2}$$

This particular method is conveniently represented in the form of the Butcher table

$$\begin{array}{c|ccc} c_1 & a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & \vdots & \vdots \\ c_s & a_{s1} & \cdots & a_{ss} \\ \hline & b_1 & \cdots & b_s \end{array} = \begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^{\mathrm{T}} \end{array}.$$

If $a_{ij} = 0$ for $j \ge i$, then the method is explicit and can be implemented directly by formulas (2). Otherwise, the method is implicit and formulas (2) assign the system of algebraic equations during which it is solved by an iterative method (usually, by the Newton method or its modifications). An implicit method is called ***stiffly accurate*** if $\mathbf{b}^{\mathrm{T}}$ coincides with one of the rows of the matrix $\mathbf{A}$ (it is usually the last row, but by changing the order of the stages we can always ensure that the desired row becomes the last row). The ***stage order*** of the method is defined as the largest integer q for which the following equalities are fulfilled:

$$\mathbf{c}^k = k\mathbf{A}\mathbf{c}^{k-1}, \quad \mathbf{b}^{\mathrm{T}}\mathbf{c}^{k-1} = 1/k, \quad k = 1,\ldots,q \tag{3}$$

(we assume the componentwise operations of multiplication and exponentiation for the vectors). Conditions (3) imply the implementation of the simplifying assumptions $C(q)$ and $B(q)$ [1, 2].

The ***inverse method*** for method (2) we obtain by interchanging $\mathbf{y}_0$ and $\mathbf{y}_1$, along with replacing h with $-h$ and t_0 with $t_1 = t_0 + h$. The inverse method possesses the following property: if we perform one step of the direct method according to (2) and then perform a step by the inverse method in the opposite direction

(by replacing h by $-h$), we obtain the initial vector $\mathbf{y}_0$. Note that the term *inverse* (*backward*), used to specify such methods in [3] and some other works, is not universally accepted. In parallel with it, the terms *adjoint* [4] and *reflected* [2] are used. For the method assigned by the triple $(\mathbf{c}, \mathbf{A}, \mathbf{b})$, we determine the coefficients of the inverse method $(\mathbf{c}^*, \mathbf{A}^*, \mathbf{b}^*)$ as follows:

$$\mathbf{c}^* = \mathbf{e} - \mathbf{c}, \quad \mathbf{A}^* = \mathbf{e}\mathbf{b}^{\mathrm{T}} - \mathbf{A}, \quad \mathbf{b}^* = \mathbf{b}, \quad \mathbf{e} = [1,\ldots,1]^{\mathrm{T}}.$$

The inverse method has the same order as the original method [4, Theorem II.8.4] and the same stage order as the original method [2, Theorem 343B]. It is easy to show that as a result of the double inversion we obtain the original method.

Sometimes, instead of (2), it is convenient to use the alternative representation of the implicit Runge–Kutta methods [5, 6] in the form

$$\mathbf{y}_1 = \mathbf{y}_0 + h\sum_{i=1}^{s} b_i\mathbf{F}_i, \quad \mathbf{F}_i = \mathbf{f}\left(t_0 + c_i h, \mathbf{Y}_i\right), \quad \mathbf{Y}_i = \left(1 - v_i\right)\mathbf{y}_0 + v_i\mathbf{y}_1 + h\sum_{j=1}^{s} x_{ij}\mathbf{F}_j, \quad i = 1,\ldots,s. \tag{4}$$

The corresponding modified table is presented as follows:

$$\begin{array}{c|c|ccc} c_1 & v_1 & x_{11} & \cdots & x_{1s} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_s & v_s & x_{s1} & \cdots & x_{ss} \\ \hline & & b_1 & \cdots & b_s \end{array} = \begin{array}{c|c|c} \mathbf{c} & \mathbf{v} & \mathbf{X} \\ \hline & & \mathbf{b}^{\mathrm{T}} \end{array}. \tag{5}$$

Of particular interest are methods that have $x_{ij} = 0$ for $j \geq i$. Here, for a known $\mathbf{y}_1$, we can find all the stage values of $\mathbf{Y}_i$ directly by formulas (4). Due to this, the system of algebraic Eqs. (4) in the vectors $\mathbf{Y}_1,\ldots,\mathbf{Y}_s$ can be reduced to the equation only in the vector $\mathbf{y}_1$, and this ensures an efficient implementation of the method. In [7], such methods are called ***mono-implicit***; they are considered in many works, including [3, 5–9].

The modified table (5) is also convenient for representing inverse methods whose coefficients we determine by the formulas $\mathbf{c}^* = \mathbf{e} - \mathbf{c}$, $\mathbf{v}^* = \mathbf{e} - \mathbf{v}$, and $\mathbf{X}^* = -\mathbf{X}$, $\mathbf{b}^* = \mathbf{b}$. The modified tables of the explicit method and the implicit method that is the inverse of it have the form

Direct method Inverse method

$$\begin{array}{c|c|ccccc} 0 & 0 & & & & & \\ c_2 & 0 & a_{21} & & & & \\ c_3 & 0 & a_{31} & a_{32} & & & \\ \vdots & 0 & \vdots & \vdots & \ddots & & \\ c_s & 0 & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & \\ \hline & & b_1 & b_2 & \cdots & b_{s-1} & b_s \end{array} \qquad \begin{array}{c|c|ccccc} 1 & 1 & & & & & \\ 1-c_2 & 1 & -a_{21} & & & & \\ 1-c_3 & 1 & -a_{31} & -a_{32} & & & \\ \vdots & 1 & \vdots & \vdots & \ddots & & \\ 1-c_s & 1 & -a_{s1} & -a_{s2} & \cdots & -a_{s,s-1} & \\ \hline & & b_1 & b_2 & \cdots & b_{s-1} & b_s \end{array} \tag{6}$$

(we omit the zero elements of matrix $\mathbf{A}$). Tables (6) show that the inverse of the explicit method is a mono-implicit method. It is proposed in [3] to use the Runge–Kutta methods that are the inverse of explicit methods for solving stiff problems. These inverse methods have several useful properties. They are stiffly accurate (the first stage coincides with the final formula of integration), have a high order of the L-damping, and are convenient to implement (since they are mono-implicit). At the same time, they have a significant disadvantage, namely, the first-stage order; this reduces the accuracy and the real order in solving stiff and differential–algebraic equations.

We can decrease the order-reduction effect or completely avoid it by increasing the stage order of the method. However, for some classes of the Runge–Kutta methods, this is not possible; here, this refers to methods that are most efficiently implemented. For example, singly diagonally implicit methods and methods that are the inverse of explicit methods can have only the first-stage order. This article describes an alternative approach, which allows one to increase the accuracy and avoid reducing the order without increasing the stage order. The new methods are developed: they are the inverse of the explicit methods and free of order reduction.

2. ORDER REDUCTION PHENOMENON

The main indicator that characterizes the accuracy of the methods for solving the Cauchy problem is the order of convergence p. When $h \to 0$, the global error is proportional to h^p. The step size for nonstiff problems is taken from the accuracy of the numerical solution; this usually ensures convergence with a given order. If the problem is stiff, then for its solution implicit methods ensuring the numerical stability with a sufficiently large step size should be used. However, in this case, the real order of convergence can be smaller than the classical order, and this causes an increase of the computational costs with the aim of attaining a given accuracy. The larger the difference between the classical and the stage order the more noticeable the order reduction. Hence, for computations with improved accuracy, it is desirable to use methods that have a sufficiently high stage order. The effect of the order reduction can be decreased also by the use of stiffly accurate methods.

For implicit methods used in practice, the higher the classical order of a method the larger the difference between the classical order and the stage order. For example, the Radau IIA methods have the order $p = 2s - 1$ for the stage order $q = s$. Therefore, the order reduction emerges to a greater extent in high-order methods. Methods whose stage order is equal to the classical order have no order reduction. However, such methods usually require large computational costs for solving algebraic system (2).

We demonstrate the effect of the order reduction by the example of the Kaps problem

$$\begin{aligned} y_1' &= -(\mu + 2)y_1 + \mu y_2^2, \quad y_1(0) = 1, \\ y_2' &= y_1 - y_2 - y_2^2, \quad y_2(0) = 1, \quad 0 \le t \le 1, \end{aligned} \tag{7}$$

whose solution is $y_1(t) = \exp(-2t)$ and $y_2(t) = \exp(-t)$. For large μ, the largest eigenvalue by magnitude of the Jacobian matrix of this system is approximately $-\mu$. Hence, μ can serve as a measure of the problem's stiffness. To solve the problem, we use the methods considered in [3], which are the inverse of the following explicit Runge–Kutta methods:

$$\text{RK11:}\ \begin{array}{c|c} 0 & \\ \hline 1 & \end{array} \qquad \text{RK21:}\ \begin{array}{c|cc} 0 & & \\ 2/3 & 2/3 & \\ \hline & 1/4 & 3/4 \end{array} \qquad \text{RK31:}\ \begin{array}{c|ccc} 0 & & & \\ 1/2 & 1/2 & & \\ 3/4 & 0 & 3/4 & \\ \hline & 2/9 & 1/3 & 4/9 \end{array}$$

$$\text{RK41:}\ \begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ 1 & 0 & 0 & 1 & \\ \hline & 1/6 & 1/3 & 1/3 & 1/6 \end{array}.$$

In the designation of a method, the first and second digits are the order and the stage order, respectively. These methods are optimal in the sense that they have the minimum error coefficients and the nonnegative coefficients of a method. The corresponding inverse methods we designate by IRK11, IRK21, IRK31, and IRK41.

We determine the numerical-solution error ε and the order estimate of a method $\tilde{p}$ for the given step size $h = 1/30$ by the formulas

$$\varepsilon(h) = \max\big(e(h,t_i), 0 \le t_i \le 1\big), \quad \tilde{p} = \frac{\log\big(\varepsilon(h_1)/\varepsilon(h_2)\big)}{\log(h_1/h_2)}, \quad h_1 = \frac{1}{25}, \quad h_2 = \frac{1}{36}, \tag{8}$$

where $e(h,t_i)$ is the Euclidean norm of the absolute error at the point $t_i = ih$. The results of the computations are presented in Fig. 1. The most noticeable effect of the order reduction is found in the IRK31 and IRK41 methods. The errors of these methods have the maximum values for a moderate stiffness, and for large μ the error is even smaller than for its small values. Such behavior of an error is typical for many stiff problems and derives from the fact that the first equation in (7) for $\mu \to \infty$ degenerates into the algebraic relation $0 = -y_1 + y_2^2$, whose accurate fulfillment is ensured by stiffly accurate methods. If a method is not stiffly accurate, the error increases with increasing μ and remains large for large μ. Note that the real order of the IRK31 and IRK41 methods in a wide range of the change of μ reduces to the stage order $q = 1$.

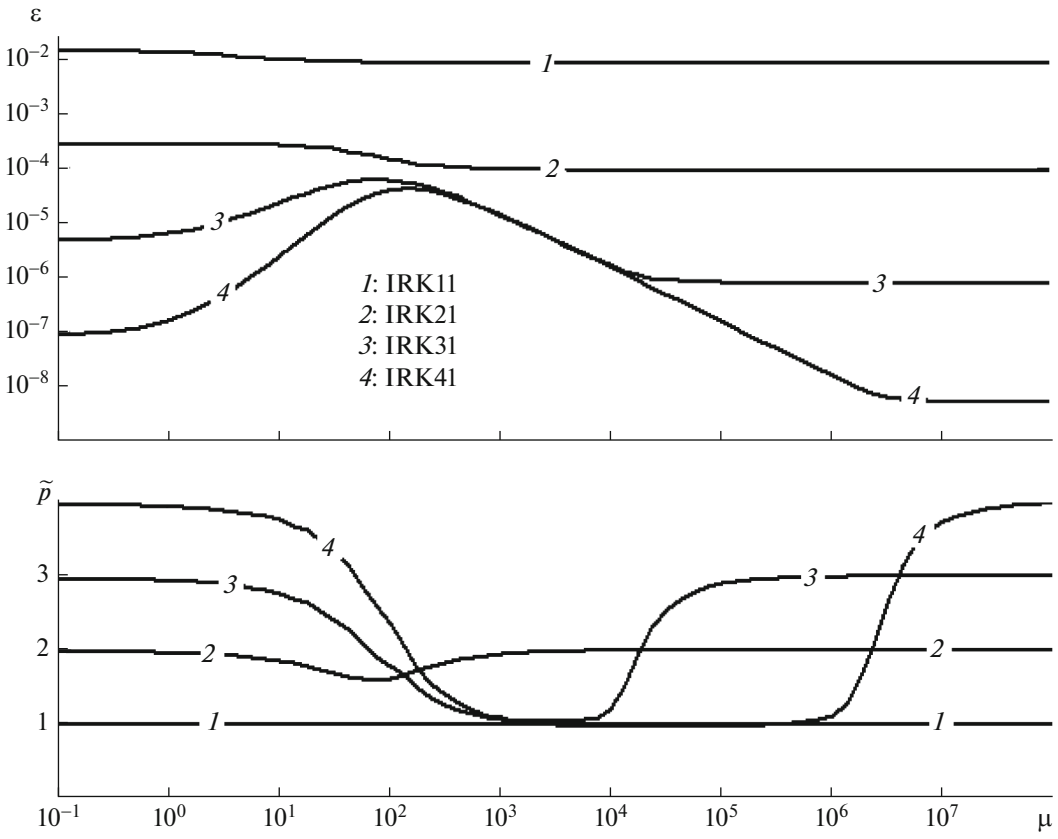

Fig. 1.

We can assume from the above-described results that stiffly accurate methods having a large difference between the classic order and the stage order can efficiently solve problems with a smooth solution if the spectrum of the Jacobian $\{\lambda_i\}$ is clearly divided into nonstiff (with small values of $|\lambda_i|$) and stiff (with very large values of $|\lambda_i|$) parts. At the same time, if the problem is moderately stiff or has the distributed spectrum of the Jacobian (this is typical for the problems obtained by the discretization of partial differential equations), then only stiff accuracy is not enough for an efficient solution and it is necessary to use methods with the sufficiently high stage order.

Many stiff problems are characterized by the availability of boundary layers, namely, segments dominated by the components of the solution corresponding to λ_i that are large in magnitude. Within such a segment, a problem is not stiff, because in selecting a step from the accuracy condition, all values of $h\lambda_i$ are quite small, and this enables efficiently using not only implicit but also explicit methods in this segment. However, in the transition between slow and fast segments with a varying step, intermediate values of $h\lambda_i$ can cause an order reduction of stiffly accurate methods even if the spectrum of the Jacobi matrix is clearly divided into nonstiff and stiff parts. Hence, for the efficient solution of problems with boundary layers, it is also necessary to use methods that are free from order reduction.

The order-reduction phenomenon is studied in many works, including [1, 2, 10–18]. It was first explained in [10] by using the Prothero–Robinson equation

$$y' = \lambda\big(y - \varphi(t)\big) + \varphi'(t), \quad y(t_0) = \varphi(t_0), \tag{9}$$

which has the solution $y(t) = \varphi(t)$. The local error of the solution of this equation by the Runge–Kutta method is determined as follows:

$$\delta_1 = \varphi(t_0 + h) - y_1 = \sum_{i=q+1}^{\infty} e_i(h\lambda)\frac{d^i\varphi(t_0)}{dt^i}\frac{h^i}{i!}, \tag{10}$$

where $e_i(z) = z\mathbf{b}^{\mathrm{T}}(\mathbf{I} - z\mathbf{A})^{-1}\left(\mathbf{c}^i - i\mathbf{A}\mathbf{c}^{i-1}\right) + \left(1 - i\mathbf{b}^{\mathrm{T}}\mathbf{c}^{i-1}\right)$ are error functions proposed in [12, 13] and $\mathbf{I} = \mathrm{diag}(1, \ldots, 1)$. The global error is expressed by the formula

$$\Delta_{n+1} = \varphi(t_{n+1}) - y_{n+1} = R(h\lambda)\Delta_n + \delta_{n+1}, \tag{11}$$

where $R(z) = 1 + z\mathbf{b}^{\mathrm{T}}(\mathbf{I} - z\mathbf{A})^{-1}\mathbf{e}$ is the stability function of the method and δ_{n+1} is the local error at the $(n + 1)$th step.

It is seen from (10) that the principle term of an error in solving Eq. (9) is determined by the function $e_{q+1}(z)$. Suppose $R(z)$ and $e_{q+1}(z)$ are the stability and error functions of a certain Runge–Kutta method, whereas $R^*(z)$ and $e^*_{q+1}(z)$ are the corresponding functions of the inverse method. We employ the fact that the step of the inverse method and the subsequent step of the original method by replacing h by $-h$ bring about the initial condition $\mathbf{y}_0$. We apply this property to the Dahlquist equation $y' = \lambda y$ and Eq. (9) for $\varphi(t) = t^{q+1}$ and obtain $R^*(z)R(-z) = 1$ and $R(-z)e^*_{q+1}(z)h^{q+1} + e_{q+1}(-z)(-h)^{q+1} = 0$ $(z = h\lambda)$; from this it follows that

$$R^*(z) = R^{-1}(-z), \quad e^*_{q+1}(z) = (-1)^q R^{-1}(-z)e_{q+1}(-z). \tag{12}$$

For $p = s$, the stability function of explicit methods is presented as $R(z) = 1 + z + \ldots + z^p/p!$, while the function $e_{q+1}(z) = e_2(z)$ has the form $e_2(z) = c_2 z^{p-1}/p!$ [15]. We substitute these expressions into (12) and obtain the corresponding functions of the inverse methods: for IRK31 in the form

$$R^*(z) = \left(1 - z + z^2/2 - z^3/6\right)^{-1}, \quad e^*_2(z) = -c_2 z^2/\left(6 - 6z + 3z^2 - z^3\right), \tag{13}$$

while for IRK41 as

$$R^*(z) = \left(1 - z + z^2/2 - z^3/6 + z/24\right)^{-1}, \quad e^*_2(z) = c_2 z^3/\left(24 - 24z + 12z^2 - 4z^3 + z^4\right). \tag{14}$$

With a fixed step size, the local solution-error of Eq. (9) for $\varphi(t) = t^{q+1}$ at all stages is the same and is expressed by the formula $\delta = e_{q+1}(h\lambda)\,h^{q+1}$. In accordance with (11), the global error for $|R(h\lambda)| < 1$, after a large number of steps, converges to

$$\Delta(h) = E_{q+1}(h\lambda)h^{q+1}, \quad E_{q+1}(z) = \frac{e_{q+1}(z)}{1 - R(z)}. \tag{15}$$

For the order of the method in solving the considered problem we get the expression

$$\tilde{p}(z) = \frac{h|\Delta(h)|'_h}{|\Delta(h)|} = q + 1 + \frac{z|E_{q+1}(z)|'_z}{|E_{q+1}(z)|}. \tag{16}$$

After substituting (13) and (14) into (15) and (16), we obtain for the IRK31 method

$$E_2(z) = \frac{c_2 z}{6 - 3z + z^2}, \quad \tilde{p}(z) = 2 + \frac{6 - z^2}{6 - 3z + z^2}$$

and for the IRK41 method

$$E_2(z) = \frac{-c_2 z^2}{24 - 12z + 4z^2 - z^3}, \quad \tilde{p}(z) = 2 + \frac{48 - 12z + z^3}{24 - 12z + 4z^2 - z^3}.$$

Figure 2 presents the dependences $|E_2(z)|$ and $\tilde{p}(z)$, which explain the behavior of errors and orders of the IRK31 and IRK41 methods shown in Fig. 1. We see that with small and large $h\mu$, the nonstiff error's component corresponding to horizontal sections of the curves $\varepsilon(\mu)$ and $\tilde{p}(\mu)$ is dominant. With moderate $h\mu$, the stiff component is dominant; this is seen from a comparison of the curves $\varepsilon(\mu)$ and the corresponding curves $|E_2(z)|$ in Fig. 2. A more detailed analysis shows that the dependence $\varepsilon(\mu)$ at moderate values of $h\mu$ almost coincides (accurate up to the constant multiplier) with $|E_2(-h\mu)|$.

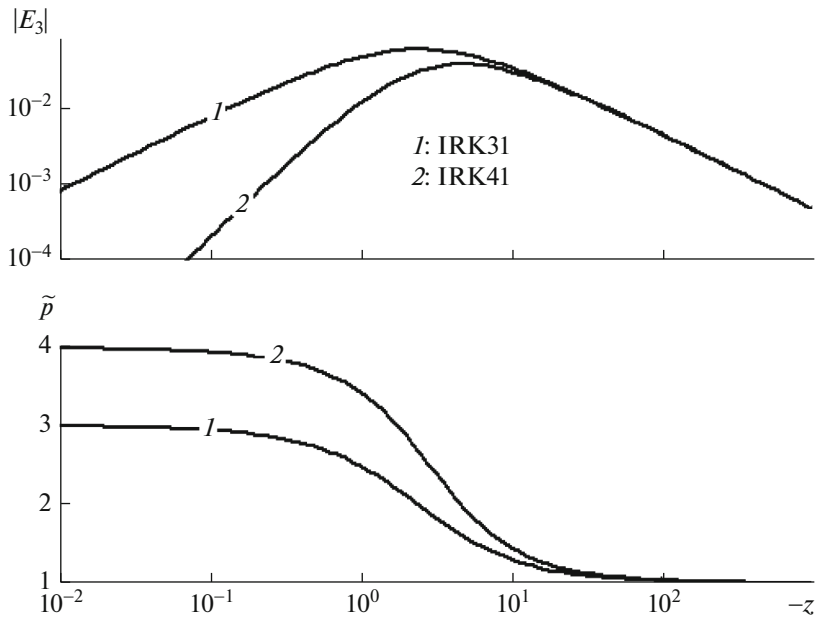

Fig. 2.

The behavior of a solution error of nonlinear stiff problems is not always explained by using Eq. (9). Therefore, in [14] also the other equations and the error functions $e_{ij}(z)$ corresponding to them are proposed, which allowed one to explain the behavior of an error in these cases. The functions $e_{ij}(z)$ for $i \le 4$ are presented in [14, 17], where it is taken that $e_{i1}(z) = e_i(z)$. For $i = 2$, one such a function is available; for $i = 3$ and $i = 4$ there are two and five such functions, respectively. For the method of the stage order q we have $e_{ij}(z) \equiv 0$ when $i \le q$, while all functions $e_{q+1,j}(z)$ are equal. The minimization of $\|e_{q+1}(z)\|$ allows one to decrease the effect of the order reduction and develop methods with an improved accuracy in the solution of stiff problems [12–17]. It is possible to develop methods having $e_{q+1}(z) \equiv 0$, and this gives an effect analogous to the increase of the stage order by one. Here, if we additionally require that $e_{q+2,j}(z) \equiv 0$, we get the effect analogous to an increase of the stage order by two. Such methods are considered in [15–17].

The foregoing makes it possible to extend the notion of the stage order. We shall say that a method has a ***pseudostage order*** $\overline{q}$ if $\overline{q}$ is the largest integer for which $\overline{q} \le p, e_{ij}(z) \equiv 0, i = 1,\ldots,\overline{q}$. In the general case, $\overline{q} \ge q$; however, for ordinary methods $\overline{q} = q$. For the methods we shall use the designations Method pq if $\overline{q} = q$ and Method $pq\overline{q}$ if $\overline{q} > q$.

This paper considers two ways of improving the accuracy and the real order of inverse methods. The first way rests on using small values of the abscissas of the original method. It is seen from (13) and (14) that if c_2 decreases, then $\left\|e_2^*(z)\right\|$ also decreases. In this case, if we assign small values of c_2 and c_3, then $\left\|e_3^*(z)\right\|$ and $\left\|e_{32}^*(z)\right\|$ are also small. The second way rests on using methods that have $\overline{q} > q = 1$. The methods having $\overline{q} = 2$ and $\overline{q} = 3$ are also proposed.

3. SECOND-ORDER METHODS

The Butcher table of an explicit two-stage second-order method has the form

$$\begin{array}{c|cc} 0 & & \\ c_2 & c_2 & \\ \hline & 1-1/(2c_2) & 1/(2c_2) \end{array}$$

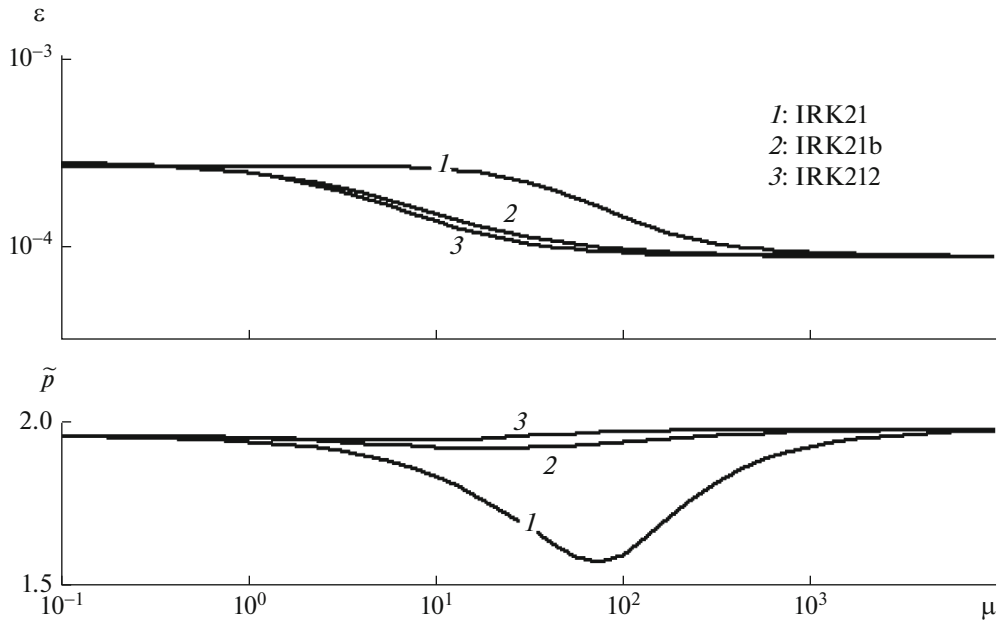

Fig. 3.

and the error function $e_2(z) = c_2 z/2$ of this method is proportional to c_2. Suppose $c_2 = 0.1$. The corresponding inverse method we present in the form of a modified table

$$\begin{array}{c|c|cc} 1 & 1 & & \\ 0.9 & 1 & -0.1 & \\ \hline & & -4 & 5 \end{array}$$

and denote by IRK21b. Recall that the IRK21 method is obtained from the explicit method having $c_2 = 2/3$ (then $c_2^* = 1/3$).

Consider now an explicit three-stage method. From the condition $q = 1$, we obtain $a_{21} = c_2$, $a_{31} = c_3 - a_{32}$, and $b_1 = 1 - b_2 - b_3$. For such a method we have

$$e_2(z) = 1 - 2(b_2c_2 + b_3c_3) + (b_2c_2^2 + b_3c_3^2)z + b_3a_{32}c_2^2z^2.$$

We call for the fulfillment of the identity $e_2(z) \equiv 0$ and obtain the two-parametric family of second-order methods with the coefficients

$$a_{32} = 0, \quad b_2 = \frac{c_3}{2c_2(c_3 - c_2)}, \quad b_3 = \frac{c_2}{2c_3(c_2 - c_3)}$$

and free parameters c_2 and c_3. Suppose $c_2 = 1/2$ and $c_3 = 1$; then the modified table of the corresponding inverse IRK212 method is presented as follows:

$$\begin{array}{c|c|ccc} 1 & 1 & & & \\ 1/2 & 1 & -1/2 & & \\ 0 & 1 & -1 & 0 & \\ \hline & & -1/2 & 2 & -1/2 \end{array}.$$

The results of solving the Kaps problem by the second-order methods for $h = 1/30$ are shown in Fig.3. Compared to IRK21, the IRK21b and IRK212 methods are more accurate and demonstrate no significant order reduction for moderate values of μ.

4. THIRD-ORDER METHODS

For $p = s = 3$, the modified table of an inverse method is presented as follows:

$$\begin{array}{c|c|cccc} 1 & 1 & & & \\ 1-c_2 & 1 & -c_2 & & \\ 1-c_3 & 1 & a_{32}-c_3 & -a_{32} & \\ \hline & & 1-b_2-b_3 & b_2 & b_3 \end{array}, \quad a_{32} = \frac{1}{6b_3c_2}, \quad b_2 = \frac{2-3c_3}{6c_2(c_2-c_3)}, \quad b_3 = \frac{2-3c_2}{6c_3(c_3-c_2)},$$

where the free parameters c_2 and c_3 are the abscissas of the original explicit method. Error functions of the inverse method have the form

$$e_2(z) = -c_2z^2\big/D(z), \quad e_3(z) = [(c_2 - 2c_3 + 3c_2c_3)z + (c_2^2 - 3c_2)z^2]\big/D(z),$$

$$e_{32}(z) = [(3c_2 - 2c_3)z - 2c_2^2]\big/D(z), \quad D(z) = 6 - 6z + 3z^2 - z^3.$$

For these functions to be small, it is necessary to assign small values of c_2 and c_3. However, they should not be assigned as very small numbers, because in this case the coefficients of the method increase and this increases the rounding errors. Assume that $c_2 = 0.01$ and $c_3 = 0.1$ and denote the obtained implicit method by IRK31b.

Explicit four-stage three-order methods that have $e_2(z) \equiv 0$ form the four-parametric family with the free coefficients c_2, c_3, c_4, and b_4 [15]. The methods of this family have the functions

$$e_3(z) = \frac{1}{6}c_2c_3z^2 + \left[\frac{1}{3}(c_2 + c_3) - \frac{1}{2}c_2c_3 - b_4c_4(c_4 - c_2)(c_4 - c_3)\right]z, \quad e_{32}(z) = \frac{1}{3}c_4z.$$

We assign $c_4 = 0$ and obtain $e_{32}(z) \equiv 0$. Hence, we obtain the three-parametric family of explicit methods having $s = 4$, $p = 3$, $e_2(z) \equiv 0$, and $e_{32}(z) \equiv 0$, together with the free coefficients c_2, c_3, and b_4. The other coefficients are determined by the formulas

$$c_4 = a_{32} = 0, \quad a_{42} = \frac{c_3}{6b_4c_2(c_3-c_2)}, \quad a_{43} = \frac{c_2}{6b_4c_3(c_2-c_3)}, \quad b_2 = \frac{2-3c_3}{6c_2(c_2-c_3)},$$

$$b_3 = \frac{2-3c_2}{6c_3(c_3-c_2)}, \quad a_{21} = c_2, \quad a_{31} = c_3, \quad a_{41} = c_4 - a_{42} - a_{43}, \quad b_1 = 1 - b_2 - b_3 - b_4.$$

Inverse methods for this family also have $e_2(z) \equiv 0$ and $e_{32}(z) \equiv 0$. We assign $c_2 = 1/2$, $c_3 = 1$, and $b_4 = 1/3$ and obtain the inverse method with the modified table

$$\begin{array}{c|c|cccc} 1 & 1 & & & & \\ 1/2 & 1 & -1/2 & & & \\ 0 & 1 & -1 & 0 & & \\ 1 & 1 & 3/2 & -2 & 1/2 & \\ \hline & & -1/6 & 2/3 & 1/6 & 1/3 \end{array};$$

we denote this method by IRK312.

The results of solving the Kaps problem by third-order methods for $h = 1/30$ are shown in Fig. 4. We investigate the dependence of an error on a step size for various μ. For small μ, all three methods give very close results, during which the real order coincides with the classical order. With increasing μ, the order-reduction effect begins to reveal itself. The dependences $\varepsilon(h)$ for two values of μ are presented in Fig. 5, where the order of a method is specified by the slope of a curve. As in Fig. 4, the order reduction is most noticeable for moderate values of $h\mu$. As μ increases, the range of values of h with a reduced order expands; this increases the advantage of the IRK312 method for small values of h.

5. FOURTH-ORDER METHODS

The construction of explicit four-stage methods is considered in [4], where the formulas for determining the coefficients of such methods are presented. These methods have $c_4 = 1$ and for $c_2 \neq c_3$ form a two-parametric family. Suppose $c_2 = 0.01$ and $c_3 = 0.1$ and denote the obtained inverse method by IRK41b.

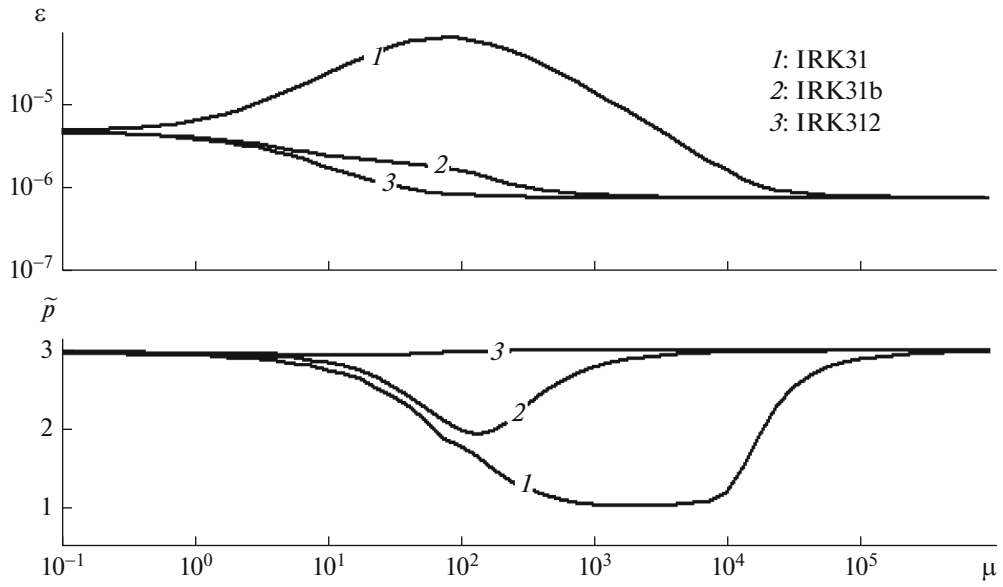

Fig. 4.

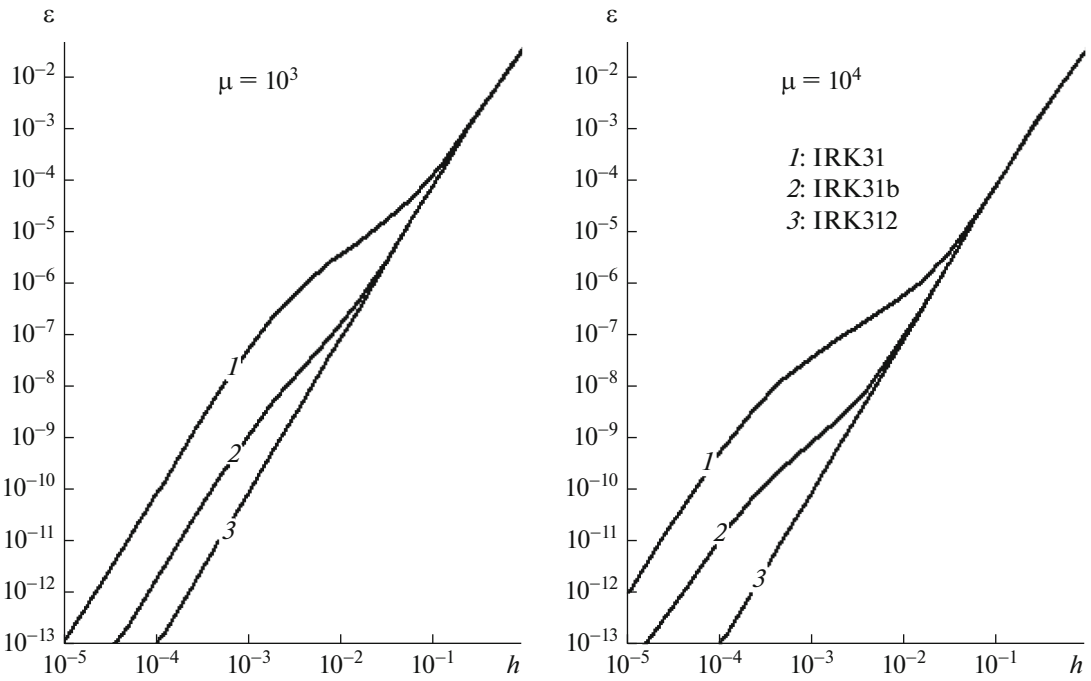

Fig. 5.

We now construct explicit fourth-order methods that have $\overline{q} = 3$, i.e., $e_2(z) \equiv 0$, $e_3(z) \equiv 0$, and $e_{32}(z) \equiv 0$. One of the methods of this family is presented in [15]. The number of stages of such methods must not be less than six. For $s = 6$, they form a six-parametric family assigned by the coefficients c_2, c_3, c_4, c_6, b_5, and b_6. All these methods have $a_{32} = a_{42} = a_{43} = c_5 = 0$. We find the weights b_2, b_3, and b_4 by solving the equations $b_2c_2 + b_3c_3 + b_4c_4 = 1/2 - b_6c_6$, $b_2c_2^2 + b_3c_3^2 + b_4c_4^2 = 1/3 - b_6c_6^2$, and $b_2c_2^3 + b_3c_3^3 + b_4c_4^3 = 1/4 - b_6c_6^3$. We determine the remaining coefficients by the formulas

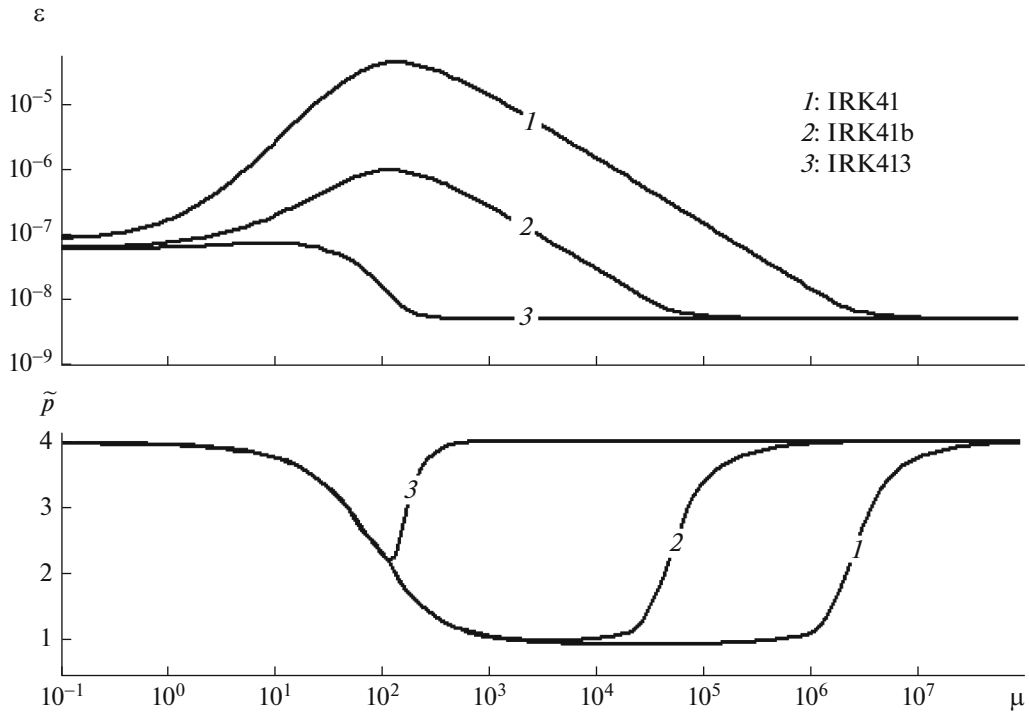

Fig. 6.

$$a_{54} = \frac{c_2c_3(4c_6-3)}{24b_5c_6c_4(c_4-c_2)(c_4-c_3)},\quad a_{52} = \frac{a_{54}c_4^2(c_4-c_3)}{c_2^2(c_3-c_2)},\quad a_{53} = \frac{a_{54}c_4^2(c_4-c_2)}{c_3^2(c_2-c_3)},$$

$$a_{64} = \frac{3c_2c_3 - 2c_6(c_2+c_3)}{24b_6c_6c_4(c_4-c_2)(c_4-c_3)},\quad a_{63} = \frac{12b_6a_{64}c_4^2(c_4-c_2)+c_2}{12b_6c_3^2(c_2-c_3)},\quad a_{62} = -\frac{a_{63}c_3^2 + a_{64}c_4^2}{c_2^3},$$

$$a_{65} = \frac{b_5c_6}{b_6(4c_6-3)},\quad a_{i1} = c_i - \sum_{j=2}^{i-1} a_{ij},\quad i = 2,\ldots,6,\quad b_1 = 1 - \sum_{j=1}^{6} b_j.$$

The inverse methods also have $\overline{q} = 3$. Setting $c_2 = 1/3$, $c_3 = 2/3$, $c_4 = c_6 = 1$, $b_5 = -1/4$, and $b_6 = 1/4$, we obtain the inverse IRK413 method with the modified table

1	1						
2/3	1	−1/3					
1/3	1	−2/3	0				
0	1	−1	0	0			
1	1	−11/12	3/2	−3/4	1/6		
0	1	−9/4	3	−15/4	1	1	
		3/8	3/8	3/8	−1/8	−1/4	1/4

The results of solving the Kaps problem by the fourth-order methods for $h = 1/30$ are shown in Fig. 6. The results presented in Figs. 4 and 6 demonstrate the convincing advantage of the methods that have $\overline{q} > q = 1$.

6. SOLUTION OF DIFFERENTIAL–ALGEBRAIC EQUATIONS

Consider the system of differential–algebraic equations (DAEs) given in the semiexplicit form

$$\begin{gathered}\mathbf{y}' = \mathbf{f}(\mathbf{y},\mathbf{z}),\quad \mathbf{y}(t_0) = \mathbf{y}_0,\\ 0 = \mathbf{g}(\mathbf{y},\mathbf{z}),\quad \mathbf{z}(t_0) = \mathbf{z}_0,\quad t_0 \le t \le t_0 + T.\end{gathered} \tag{17}$$

The formulas of one step of solving this system by the inverse of the explicit two-stage method with the coefficients $a_{21} = c_2$, b_1, and $b_2 = 1 - b_1$ have the form

$$\begin{gathered}\mathbf{Y}_1 = \mathbf{y}_0 + h\left[b_1\mathbf{f}(\mathbf{Y}_1,\mathbf{Z}_1) + b_2\mathbf{f}(\mathbf{Y}_2,\mathbf{Z}_2)\right],\quad 0 = \mathbf{g}(\mathbf{Y}_1,\mathbf{Z}_1),\\ \mathbf{Y}_2 = \mathbf{Y}_1 - ha_{21}\mathbf{f}(\mathbf{Y}_1,\mathbf{Z}_1),\quad 0 = \mathbf{g}(\mathbf{Y}_2,\mathbf{Z}_2),\quad \mathbf{y}_1 = \mathbf{Y}_1,\quad \mathbf{z}_1 = \mathbf{Z}_1.\end{gathered} \tag{18}$$

From these equations one can exclude $\mathbf{Y}_2$; then in the case of a system of ODEs, we obtain the formula of one step $\mathbf{y}_1 = \mathbf{y}_0 + h\left[b_1\mathbf{f}(\mathbf{y}_1) + b_2\mathbf{f}(\mathbf{y}_1 - ha_{21}\mathbf{f}(\mathbf{y}_1))\right]$. Analogously, when solving the system of ODEs, a step of any mono-implicit method can be represented in the form of an algebraic equation with respect only to the vector $\mathbf{y}_1$. However, if the system contains algebraic equations, then in the general case, it is not possible to eliminate internal stage values. For example, one cannot exclude $\mathbf{Z}_2$ from (18). In the general case, it is also impossible to exclude from the equations of the method the values of the internal stages if the system of DAEs is given in the implicit form $\mathbf{f}(\mathbf{y},\mathbf{y}') = 0$. Hence, in solving DAEs the advantage of mono-implicit methods cannot be fully used.

Consider the use of inverse methods for solving DAEs. When implementing the IRK212, IRK312, and IRK413 methods, it is necessary to eliminate the dependent stage values by setting $\mathbf{Y}_2 = (\mathbf{Y}_1 + \mathbf{Y}_3)/2$ in the IRK212 and IRK312 methods, along with $\mathbf{Y}_2 = (2\mathbf{Y}_1 + \mathbf{Y}_4)/3$ and $\mathbf{Y}_3 = (\mathbf{Y}_1 + 2\mathbf{Y}_4)/3$ in the IRK413 method. Due to this, the complexity of the obtained system of algebraic equations is identical with that of the IRK21, IRK31, and IRK41 methods having the smaller number of stages and the same order. We derive computational relations using the ε-embedding method. For this purpose, we write the formulas of integration of the system of ODEs $\mathbf{y}' = \mathbf{f}(\mathbf{y},\mathbf{z})$ and $\varepsilon\mathbf{z}' = \mathbf{g}(\mathbf{y},\mathbf{z})$; then we assume that $\varepsilon = 0$. As an example, for the IRK212 method, the obtained equations with respect to stage values in solving DAEs (17) are presented as

$$\begin{gathered}\mathbf{Y}_1 = \mathbf{y}_0 + h\left[-\frac{1}{2}\mathbf{f}(\mathbf{Y}_1,\mathbf{Z}_1) + 2\mathbf{f}\left(\frac{\mathbf{Y}_1 + \mathbf{Y}_3}{2},\frac{\mathbf{Z}_1 + \mathbf{Z}_3}{2}\right) - \frac{1}{2}\mathbf{f}(\mathbf{Y}_3,\mathbf{Z}_3)\right],\\ 0 = -\frac{1}{2}\mathbf{g}(\mathbf{Y}_1,\mathbf{Z}_1) + 2\mathbf{g}\left(\frac{\mathbf{Y}_1 + \mathbf{Y}_3}{2},\frac{\mathbf{Z}_1 + \mathbf{Z}_3}{2}\right) - \frac{1}{2}\mathbf{g}(\mathbf{Y}_3,\mathbf{Z}_3),\\ \mathbf{Y}_3 = \mathbf{Y}_1 - h\mathbf{f}(\mathbf{Y}_1,\mathbf{Z}_1),\quad 0 = \mathbf{g}(\mathbf{Y}_1,\mathbf{Z}_1),\quad \mathbf{y}_1 = \mathbf{Y}_1,\quad \mathbf{z}_1 = \mathbf{Z}_1.\end{gathered}$$

The equations of the more general form for solving ODEs and DAEs are implemented as a structure diagram in the MVTU software package (such a diagram for solving ODEs is presented in [19]). The results of solving various systems of ODEs and DAEs are obtained by using this diagram, including the results presented in the present paper.

Systems of DAEs of the highest indices, i.e., indices that are larger than one (the definition of the ***differentiation index*** of DAEs is presented in [1]), are the most difficult to solve numerically. At the same time, such equations often arise when solving problems in mechanics, control theory, and electrical engineering [1]. For example, equations of a mechanical system (with links) that are formed on the basis of the Lagrange principle have index three.

Consider the system of DAEs of index three

$$y_1' = -(y_1y_2z_1z_2)^{1/6},\quad y_2' = y_1(y_2 - 3z_2)/z_1,$$

$$z_1' = -z_1z_2u/(y_1y_2),\quad z_2' = -(y_1y_2 + z_1z_2)/u,$$

$$0 = y_1^2 - y_2,\quad y_1(0) = y_2(0) = z_1(0) = z_2(0) = u(0) = 1,\quad 0 \le t \le 1,$$

whose solution is $y_1(t) = z_1(t) = u(t) = e^{-t}$ and $y_2(t) = z_2(t) = e^{-2t}$. For the numerical solution we use the third- and fourth-order inverse methods, along with the fifth-order Radau IIA method, which we denote by Radau5. At $h = 1/30$, we determine the errors and estimates of the order by formulas (8) for each of the following components of the solution: $\mathbf{y} = (y_1, y_2)$, which is the y component (the variables of index one); $\mathbf{z} = (z_1, z_2)$, which is the z component (the variables of index two); and u, i.e., the u component (the variable of index three). The results are presented in the Table 1.

Table 1

Method	ε_y	ε_z	ε_u	$\tilde{p}_y$	$\tilde{p}_z$	$\tilde{p}_u$
IRK31	1.03×10^{-3}	5.52×10^{-3}	2.06×10^{-1}	−0.10	0.10	−0.09
IRK31b	3.17×10^{-6}	9.18×10^{-5}	2.93×10^{-3}	0.76	0.94	−0.33
IRK312	1.11×10^{-6}	5.42×10^{-7}	2.64×10^{-5}	2.86	3.04	2.08
IRK41	2.39×10^{-3}	1.13×10^{-2}	2.98×10^{-1}	0.02	0.07	−0.06
IRK41b	1.08×10^{-6}	6.07×10^{-5}	4.45×10^{-3}	−0.01	0.94	−0.24
IRK413	1.72×10^{-8}	1.40×10^{-8}	2.42×10^{-5}	3.95	4.04	2.00
Radau5	3.83×10^{-10}	1.83×10^{-6}	2.29×10^{-4}	4.07	2.97	1.99

Methods IRK31, IRK31b, IRK41, and IRK41b do not ensure convergence, because with decreasing a step, the error does not decrease as illustrated by the near-zero and negative estimates of the order. Therefore, these methods are unsuitable for solving DAEs of index three (but for solving equations of index two with poor accuracy we can use them). The other methods from the Table 1 successfully solve systems of index three; here, IRK413 is highly competitive with Radau5, which is considered to be one of the most efficient methods (in various cases, including the solution of DAEs of index three).

CONCLUSIONS

The results of the experiments show that the IRK212, IRK312, and IRK413 methods having $\overline{q} > q$ offer a convincing advantage among inverse methods. Unlike other inverse methods, they barely reduce the order when solving stiff ODEs and ensure convergence when solving DAEs of index three. Note that the absence of order reduction is very important in calculations on thickening grids [20] and in all cases where the extrapolation by Richardson is used [4] for estimating an error or improving a solution.

REFERENCES

1. E. Hairer and G. Wanner, *Solving Ordinary Differential Equations. II: Stiff and Differential-Algebraic Problems* (Springer, Berlin, 1996).
2. J. C. Butcher, *Numerical Methods for Ordinary Differential Equations,* 2th ed. (Wiley, Chichester, 2008).
3. N. N. Kalitkin and I. P. Poshivailo, "Computations with inverse Runge-Kutta methods," Math. Models Comput. Simul. **6**, 272–285 (2014).
4. E. Hairer, S. P. Norsett, and G. Wanner, *Solving Ordinary Differential Equations. I: Nonstiff Problems* (Berlin, Springer, 1987).
5. P. Muir and B. Owren, "Order barriers and characterizations for continuous mono-implicit Runge-Kutta schemes," Math. Comput. **61** (204), 675–699 (1993).
6. K. Burrage, F. H. Chipman, and P. H. Muir, "Order results for mono-implicit Runge-Kutta methods," SIAM J. Numer. Anal. **31**, 876–891 (1994).
7. J. R. Cash and A. Singhal, "Mono-implicit Runge-Kutta formulae for the numerical integration of stiff differential systems," IMA J. Numer. Anal. **2**, 211–227 (1982).
8. G. Yu. Kulikov and S. K. Shindin, "Adaptive nested implicit Runge-Kutta formulas of Gauss type," Appl. Numer. Math. **59**, 707–722 (2009).
9. G. Yu. Kulikov, "Embedded symmetric nested implicit Runge-Kutta methods of Gauss and Lobatto types for solving stiff ordinary differential equations and Hamiltonian systems," Comput. Math. Math. Phys. **55**, 983–1003 (2015).
10. A. Prothero and A. Robinson, "On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations," Math. Comput. **28**, 145–162 (1974).
11. K. Dekker and J. G. Verwer, *Stability of Runge-Kutta Methods for Stiff Nonlinear Differential Equations* (Elsevier, Amsterdam, 1984).
12. L. M. Skvortsov, "Increasing the accuracy of explicit Runge-Kutta methods in solving moderately stiff problems," Dokl. Math. **63**, 387–389 (2001).

13. L. M. Skvortsov, "Diagonally implicit Runge–Kutta FSAL methods for stiff and differential-algebraic systems," Mat. Model. **14** (2), 3–17 (2002).
14. L. M. Skvortsov, "Accuracy of Runge-Kutta methods applied to stiff problems," Comput. Math. Math. Phys. **43**, 1320–1330 (2003).
15. L. M. Skvortsov, "Explicit Runge-Kutta methods for moderately stiff problems," Comput. Math. Math. Phys. **45**, 1939–1951 (2005).
16. L. M. Skvortsov, "Diagonally implicit Runge-Kutta methods for stiff problems," Comput. Math. Math. Phys. **46**, 2110–2123 (2006).
17. L. M. Skvortsov, "Model equations for accuracy investigation of Runge-Kutta methods," Math. Models Comput. Simul. **2**, 800–811 (2010).
18. J. Rang, "An analysis of the Prothero-Robinson example for constructing new adaptive ESDIRK methods of order 3 and 4," Appl. Numer. Math. **94**, 75–87 (2015).
19. O. S. Kozlov and L. M. Skvortsov, "MVTU software package in the scientific research and applied developments," Mat. Model. **27** (11), 32–46 (2015).
20. N. N. Kalitkin, A. B. Alshin, E. A. Alshina, and B. V. Rogov, *Computations on Quasi-Uniform Grids* (Fizmatlit, Moscow, 2005) [in Russian].

Translated by L. Kartvelishvili