



## Original Article

# Some Runge-Kutta Algorithms with Error Control and Variable Stepsizes for Solving a Class of Differential-Algebraic Equations

Phan Quang Tuyen<sup>1,\*</sup>*Department of Basic Science, College of Artillery Officer's Training, Son Tay, Hanoi, Vietnam*

Received 09 September 2020

Revised 29 September 2020; Accepted 05 October 2020

**Abstract:** In this paper, we propose and discuss numerical algorithms for solving a class of nonlinear differential-algebraic equations (DAEs). These algorithms are based on half-explicit Runge-Kutta methods (HERK) that have been studied recently for solving strangeness-free DAEs. The main idea of this work is to use the half-explicit variants of some well-known embedded Runge-Kutta methods such as Runge-Kutta-Fehlberg and Dormand-Prince pairs. Thus, we can estimate local errors and choose suitable stepsizes accordingly to a given tolerance. The cases of unstructured and structured DAEs are investigated and compared. Finally, some numerical experiences are given for illustrating the efficiency of the algorithms.

**Keywords:** Differential-algebraic equation, strangeness-free form, half-explicit Runge-Kutta method, Fehlberg and Dormand-Prince embedded pairs.

## 1. Introduction

We consider the initial value problem (IVP) for differential-algebraic equations (DAEs) of the form

$$\begin{aligned} f(t, x(t), x'(t)) &= 0, \\ g(t, x(t)) &= 0, \end{aligned} \tag{1}$$

on an interval  $I = [t_0, t_f]$ , together with an initial condition  $x(t_0) = x_0$ . We assume that  $f : I \times \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}^{m_1}$ ,  $g : I \times \mathbb{R}^m \rightarrow \mathbb{R}^{m_2}$ , where  $m = m_1 + m_2$ , are sufficiently smooth functions

\*Corresponding author.

Email address: [quangtuyen.ktqs@gmail.com](mailto:quangtuyen.ktqs@gmail.com)

<https://doi.org/10.25073/2588-1124/vnumap.4603>

with bounded partial derivatives. We also assume that the IVP has a unique solution  $x(t)$ , which is sufficiently smooth. Futhermore, we assume that (1) is strangeness-free, which means that the combined Jacobian

$$\begin{bmatrix} f_x(t, x, x') \\ g_x(t, x) \end{bmatrix} \text{ is nonsingular along the solution } x(t). \tag{2}$$

In this work, we focus on the structured strangeness-free differential-algebraic equations (DAEs) of the form

$$\begin{aligned} f(t, x, E(t)x') &= 0, \\ g(t, x) &= 0, \end{aligned} \tag{3}$$

on an interval  $I = [t_0, t_f]$ , where  $x \in C^1(I, \mathbb{R}^m)$ ,  $E \in C^1(I, \mathbb{R}^{m_1, m_2})$ , and an initial condition  $x(t_0) = x_0$  is given. We assume that  $f = f(t, u, v) : I \times \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \rightarrow \mathbb{R}^{m_1}$  and

$g = g(t, u) : I \times \mathbb{R}^m \rightarrow \mathbb{R}^{m_2}$ ,  $m = m_1 + m_2$  are sufficiently smooth functions with bounded partial derivatives. Futhermore, we assume that the unique solution of (3) exists and

$$\begin{bmatrix} f_v E \\ g_u \end{bmatrix} \text{ is nonsingular along the exact solution } x(t). \tag{4}$$

Here,  $f_v$  and  $g_u$  denote the Jacobian of  $f$  with respect to  $v$  and that of  $g$  with respect to  $u$ , respectively.

Half-explicit Runge-Kutta methods for solving (1), (3) are proposed in [3], [4] with uniform stepsize  $h$ . It is shown that the half-explicit Runge-Kutta methods applied directly to (1) suffer an order reduction if the order of original Runge-Kutta methods are greater than 2. When we use the half-explicit Runge-Kutta methods applied to (3), all the convergence and stability results are preserved as the underlying Runge-Kutta methods for ODEs, see [4].

Let us recall the ordinary differential equations (ODEs) of the form

$$y' = f(t, y), \quad y(t_0) = y_0 \in \mathbb{R}^m, \quad t \in [t_0, t_f]. \tag{5}$$

The essential idea of the embedded methods is to calculate two approximations  $y_n$  and  $\tilde{y}_n$  at  $t_n$ , such that  $y_n - \tilde{y}_n$  gives an estimate of the local error of the less accurate of the two approximate solutions  $y_n$ . A pair of Runge-Kutta methods of orders  $p$  and  $p+1$ , respectively, is used for this purpose. The key idea of embedded methods is that the pair shares the same stage computations. The general  $s$ -stage embedded Runge-Kutta pair of orders  $p, p+1$  can be written by a combined Butcher tableau

$$\begin{array}{c|c} c & A \\ \hline & \tilde{b}^T \\ \hline & \tilde{b}^T \end{array} \tag{6}$$

where  $A \in \mathbb{R}^{s \times s}$ , is strict lower triangular,  $b^T, \hat{b}^T, c \in \mathbb{R}^s$ . The vector  $b, \hat{b}$  define the coefficients of the  $p$ -th and  $(p+1)$ -th order approximations respectively. For each solution  $y_n$ , an error tolerance ( $TOL$ ) is specified as

$$TOL = ATOL + |y_n| RTOL.$$

$ATOL$  and  $RTOL$  are denoted the absolute tolerance and the relative tolerance, respectively. We have the absolute tolerance if the relative tolerance is set zero. On the other hand, we have the relative error if the absolute tolerance is set zero. At every step we should check  $|y_n - y_n| \leq TOL$ . If this inequality is not satisfied, then the stepsize  $h$  is rejected and another stepsize  $h$  is selected instead. We will choose  $h$  such that

$$\left(\frac{h}{h}\right)^{p+1} |y_n - y_n| \approx frac.TOL,$$

where  $frac$  is safety factor (0.8 or 0.9) and repeat the process until an acceptable stepsize is found.

If the above inequality is satisfied then the same formula can be used to predict a larger stepsize  $h_{n+1} = h$  for the next time step.

The most famous embedded methods are the Fehlberg 4(5) pair and the Dormand-Prince4(5) pair, see [1].

Table 1. Fehlberg 4(5) pair

0						
$\frac{1}{4}$	$\frac{1}{4}$					
$\frac{3}{8}$	$\frac{3}{32}$	$\frac{9}{32}$				
$\frac{12}{13}$	$\frac{1932}{2197}$	$-\frac{7200}{2197}$	$\frac{7296}{2197}$			
1	$\frac{439}{216}$	-8	$\frac{3680}{513}$	$-\frac{845}{4104}$		
$\frac{1}{2}$	$-\frac{8}{27}$	2	$-\frac{3544}{2565}$	$\frac{1859}{4104}$	$-\frac{11}{40}$	
	$\frac{25}{216}$	0	$\frac{1408}{2565}$	$\frac{2197}{4104}$	$-\frac{1}{5}$	0
	$\frac{16}{135}$	0	$\frac{6656}{12825}$	$\frac{28561}{56430}$	$-\frac{9}{50}$	$\frac{2}{55}$

The Fehlberg4(5) pair has 6 stages and it is designed to minimize the local error in  $y_n$ . The Dormand-Prince4(5) pair has 7 stages and it is designed to minimize the local error in  $y_n$ . However,

the last stage is the same as the first stage for the next step ( $y_n = Y_7$ ) and the next step  $Y_7 \rightarrow Y_1$ . So, this method has the cost of a 6-stage method.

Table 2. Demand matrix  $A(s)$  with

0							
$\frac{1}{5}$	$\frac{1}{5}$						
$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$					
$\frac{4}{5}$	$\frac{44}{45}$	$-\frac{56}{15}$	$\frac{32}{9}$				
$\frac{8}{9}$	$\frac{19372}{6561}$	$-\frac{25360}{2187}$	$\frac{64448}{6551}$	$-\frac{212}{729}$			
1	$\frac{9017}{3168}$	$-\frac{355}{33}$	$\frac{46732}{5247}$	$\frac{49}{176}$	$-\frac{5103}{18656}$		
1	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	
	$\frac{5179}{57600}$	0	$\frac{7571}{16695}$	$\frac{393}{640}$	$-\frac{92097}{339200}$	$\frac{187}{2100}$	$\frac{1}{40}$
	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	0

In this paper, we use the half-explicit Runge-Kutta methods based on embedded methods to solve the DAEs of the form (1) and (3). These methods are suitable for estimating the errors. In particular, we demonstrate that these methods are more efficient than the half-explicit Runge-Kutta methods with uniform stepsize.

The paper is organized as follows. In Section 2, we present the half-explicit Runge-Kutta methods with variable stepsize and discuss their convergence. We also investigate these algorithms for two cases: strangeness-free DAEs and structured strangeness-free DAEs. Numerical results given in Section 3 illustrate the theoretical results in Section 2 and we compare with numerical results by the half-explicit Runge-Kutta methods with uniform stepsize.

## 2. Half-explicit Runge-Kutta methods with variable stepsize for solving DAEs

By using the half-explicit Runge-Kutta methods with uniform stepsize, we can estimate the actual errors and the convergence of these methods, see [4]. However, we could not know the exact solutions in general DAEs problems. So, we use the embedded Runge-Kutta methods combined with the half-explicit approach to estimate the errors and choose the suitable stepsize  $h$  accordingly to a given error tolerance.

Let us take a pair of Runge-Kutta methods (6) with order  $p$  and  $p + 1$ , where the coefficients are  $A = [a_{ij}]_{s \times s}$ ,  $b = (b_1 \dots b_s)^T$  and  $\hat{b} = (\hat{b}_1 \dots \hat{b}_s)^T$ .

### 2.1. The strangeness-free DAEs

We use an  $s$ -stage explicit Runge-Kutta method of order  $p$  with  $c_1 = 0$  and a strictly lower triangular matrix  $A$ . We assume that  $a_{i+1,i} \neq 0$  for  $i = 1, 2, \dots, s-1$  and  $b_s \neq 0$ . Consider an interval  $[t_{n-1}, t_n]$  and an approximation  $x_{n-1}$  to  $x(t_{n-1})$  is given. Let  $U_i \approx x(t_{n-1} + c_i h)$  be the stage approximation of solution  $x$  at  $t_{n-1} + c_i h$  and let  $K_i \approx U_i'$  be the approximations to the derivatives of  $U_i$ ,  $i = 1, \dots, s$ . The explicit Runge-Kutta scheme reads

$$\begin{aligned} U_1 &= x_{n-1}, \\ U_i &= x_{n-1} + h \sum_{j=1}^{i-1} a_{i,j} K_j, \quad i = 2, 3, \dots, s. \\ x_n &= x_{n-1} + h \sum_{i=1}^s b_i K_i. \end{aligned} \quad (7)$$

The approximations  $U_{i+1}$ ,  $K_i$ ,  $i = 1, 2, \dots, s-1$  are determined by the systems

$$\begin{aligned} (a) \quad & f \left( t_{n-1} + c_i h, U_i, \frac{1}{a_{i+1,i}} \left[ \frac{U_{i+1} - x_{n-1}}{h} - \sum_{j=1}^{i-1} a_{i+1,j} K_j \right] \right) = 0, \\ (b) \quad & g(t_{n-1} + c_{i+1} h, U_{i+1}) = 0, \quad i = 1, 2, \dots, s-1, \end{aligned} \quad (8)$$

Finally, the numerical solution  $x_n$  with order  $p$  is determined by the system

$$\begin{aligned} (a) \quad & f \left( t_{n-1} + c_s h, U_s, \frac{1}{b_s} \left[ \frac{x_n - x_{n-1}}{h} - \sum_{i=1}^{s-1} b_i K_i \right] \right) = 0, \\ (b) \quad & g(t_n, x_n) = 0. \end{aligned} \quad (9)$$

Here

$$K_1 = \frac{U_2 - x_{n-1}}{h a_{21}}, \quad K_i = \frac{1}{a_{i+1,i}} \left[ \frac{U_{i+1} - x_{n-1}}{h} - \sum_{j=1}^{i-1} a_{i+1,j} K_j \right], \quad i = 2, 3, \dots, s-1,$$

and

$$K_s = \frac{1}{b_s} \left[ \frac{x_n - x_{n-1}}{h} - \sum_{i=1}^{s-1} b_i K_i \right].$$

These formulas are taken from [3]. For  $p \leq 2$ , the order conditions are the same as in the ODE case. We have the convergence result for the 2-stage half-explicit Runge-Kutta method.

**Theorem 1** (Theorem 10, [3]). *Assume that the Runge-Kutta method with  $s = 2$  satisfied*

$$c_2 = a_{21}, \quad b_1 + b_2 = 1, \quad c_2 b_2 = \frac{1}{2}. \quad (10)$$

If the initial condition  $x_0$  is consistent, then the half-explicit Runge-Kutta (HERK) applied to (1) is convergent of order  $p = 2$ .

**Remark 1.** The numerical results presented in [4] have shown that, if the order of the half-explicit Runge-Kutta  $p > 2$ , the order reduction occurs.

Now, we will use a pair of the half-explicit Runge-Kutta methods of orders  $p$  and  $p + 1$ . We suppose that the order  $p$  is reduced to  $p$ . We have the following algorithm.

**Algorithm 1. Input:** Given DAE (1) on an interval  $[t_0, t_f]$  with the consistent initial condition  $x(t_0) = x_0$ , an error tolerance  $TOL$ , and an initial guess of stepsize  $h_0$ .

**Output:** The numerical solution  $\{x_n\}$  on a non-uniform mesh  $\{t_n\}$ . For  $n = 1, 2, \dots$

1. If  $t_{n-1} + h_{n-1} > t_f$  then  $h_{n-1} = t_f - t_{n-1}$ ;
2. Compute the stage approximations  $U_i, i = 1, 2, \dots, s$ , the numerical solutions  $x_n$  and  $y_n$  by (8), (9). Calculate

$$h_{new} = h_{n-1} \left( \frac{frac.TOL}{|y_n - x_n|} \right)^{\frac{1}{p+1}}$$

3. If  $|y_n - x_n| \leq TOL$  (the step is accepted) then

$$t_n = t_{n-1} + h_{n-1};$$

$$x_n = y_n;$$

If  $t_n = t_f$  then stop;

$$h_n = h_{new}; \text{ (the next stepsize is predicted)}$$

else

$$h_{n-1} = h_{new} \text{ (the stepsize will be adjusted); and go back to 2.}$$

### 2.2. The structured strangeness-free DAEs

Consider the DAEs in (3) is transformed as the form

$$\begin{aligned} f(t, x(t), (Ex)'(t) - E'(t)x(t)) &= 0, \\ g(t, x(t)) &= 0, \end{aligned} \tag{11}$$

on an interval  $I = [t_0, t_f]$ .

We take an arbitrary explicit Runge-Kutta method of order  $p$ , i.e., the coefficient matrix  $A$  is a strictly lower triangular matrix. Consider a sub-interval  $[t_{n-1}, t_n]$  and suppose that an approximation  $x_{n-1}$  to  $x(t_{n-1})$  is given. Let  $T_i = t_{n-1} + c_i h$  be the time at stage  $i$  and the stage approximations  $U_i \approx x(T_i), K_i \approx (Ex)'(T_i), i = 1, 2, \dots, s$ .

Futhermore, we assume that the values  $E_i = E(T_i), E'_i = E'(T_i)$  are available. The explicit Runge-Kutta scheme for (11) reads

$$\begin{aligned}
U_1 &= x_{n-1}, \\
E_i U_i &= E(t_{n-1})U_1 + h \sum_{j=1}^{i-1} a_{ij} K_j, \quad i = 2, 3, \dots, s, \\
E(t_n)x_n &= E(t_{n-1})x_{n-1} + h \sum_{i=1}^s b_i K_i.
\end{aligned} \tag{12}$$

The approximations  $U_{i+1}$ ,  $K_i$ ,  $i = 1, 2, \dots, s-1$  are determined by the systems

$$\begin{aligned}
0 &= hf \left( T_i, U_i, \frac{1}{a_{i+1,i}} \left[ \frac{E_{i+1}U_{i+1} - E(t_{n-1})U_1}{h} - \sum_{j=1}^{i-1} a_{i+1,j} K_j \right] - E'_i U_i \right), \\
0 &= g(T_{i+1}, U_{i+1}), \quad i = 1, 2, \dots, s-1.
\end{aligned} \tag{13}$$

Finally, the numerical solution  $x_n$  with order  $p$  is determined by the system

$$\begin{aligned}
0 &= hf \left( T_s, U_s, \frac{1}{b_s} \left[ \frac{E(t_n)x_n - E(t_{n-1})x_{n-1}}{h} - \sum_{i=1}^{s-1} b_i K_i \right] - E'_s U_s \right), \\
0 &= g(t_n, x_n),
\end{aligned} \tag{14}$$

where

$$K_i = \frac{1}{a_{i+1,i}} \left[ \frac{E_{i+1}U_{i+1} - E(t_{n-1})U_1}{h} - \sum_{j=1}^{i-1} a_{i+1,j} K_j \right], \quad i = 1, 2, \dots, s-1.$$

And

$$K_s = \frac{1}{b_s} \left[ \frac{E(t_n)x_n - E(t_{n-1})U_1}{h} - \sum_{i=1}^{s-1} b_i K_i \right].$$

**Remark 2.** The nonlinear systems (13), (14) can be solved approximately by Newton's method, see [4].

**Remark 3.** We note that the first equations of (13), (14) are scaled by  $h$ . The scaling by  $h$  is natural since it helps to balance the factor  $\frac{1}{h}$  in the first equations of (13), (14) as it is done for ODEs.

We have the convergence of HERK methods applied to (11), see [4].

**Theorem 2.** Consider the initial value problem for the DAE (3) with consistent initial value, i.e.,  $g(t_0, x_0) = 0$ . Suppose that (4) holds in a neighborhood of the exact solution  $x(t)$ . The HERK scheme (12-14) applied to (11) is convergent of order  $p$ , i.e.,

$$\|x_n - x(t_n)\| = \Theta(h^p) \quad \text{as } h \rightarrow 0$$

( $t_n \in [t_0, t_f]$  is fixed with  $t_n - t_0 = nh$ ).

We use the HERK methods based on the embedded pair of Runge-Kutta methods of orders  $p$  and  $p + 1$  to solve the DAEs (3). We have the following algorithm.

**Algorithm 2. Input:** Given DAE (11) on an interval  $[t_0, t_f]$  with the consistent initial condition  $x(t_0) = x_0$ , an error tolerance  $TOL$ , and an initial guess of stepsize  $h_0$ .

**Output:** The numerical solution  $\{x_n\}$  on a non-uniform mesh  $\{t_n\}$ . For  $n = 1, 2, \dots$

1. If  $t_{n-1} + h_{n-1} > t_f$  then  $h_{n-1} = t_f - t_{n-1}$ ;
2. Compute the stage approximations  $U_i, i = 1, 2, \dots, s$ , the numerical solutions  $x_n$  and  $y_n$  by (13), (14). Calculate

$$h_{new} = h_{n-1} \left( \frac{frac.TOL}{|y_n - x_n|} \right)^{\frac{1}{p+1}}$$

3. If  $|y_n - x_n| \leq TOL$  (the step is accepted) then

$$t_n = t_{n-1} + h_{n-1};$$

$$x_n = y_n;$$

If  $t_n = t_f$  then stop;

$$h_n = h_{new}; \text{ (the next stepsize is predicted)}$$

else

$$h_{n-1} = h_{new} \text{ (the stepsize will be adjusted); and go back to 2.}$$

Because the HERK methods applied to (3) have the same convergent order as ODE case. So, we get the following result.

**Proposition 1.** The HERK methods based on the embedded pair of Runge-Kutta methods of orders  $p$  and  $p + 1$  for solving the structured strangeness-free DAEs (11) are convergent of order  $p$ .

**Proof.** The proposition is directly obtained from Theorem 2.1, Lemma 3.5 and Theorem 3.7 in [4].

### 3. Numerical Experiments

For illustrating the efficiency of the methods in Section 2, we present some numerical experiments to demonstrate the errors, the stepsize  $h$  and to make a comparison with the corresponding results of the HERK methods with uniform stepsize for a given error tolerance. All numerical experiments are implemented in Matlab software by using Fehlberg 4(5) pair and the Dormand-Prince 4(5) pair.

**Example 1.** We consider the test DAE with some specified value of parameters  $\lambda$  and  $\omega$  on the interval  $[0,5]$

$$\begin{bmatrix} 1 & -\omega t \\ 0 & 0 \end{bmatrix} x' = \begin{bmatrix} \lambda & \omega(1 - \lambda t) \\ -1 & 1 + \omega t \end{bmatrix} x. \tag{15}$$

The initial condition  $x_1(0) = 1, x_2(0) = 1$  yields the exact solution



$$x = \begin{bmatrix} e^{\lambda t} (1 + \omega t) \\ e^{\lambda t} \end{bmatrix}.$$

The DAEs in (15) can be written in the structured form

$$\begin{aligned} ([1 \quad -\omega t]x)' &= [\lambda \quad -\lambda \omega t]x, \\ 0 &= [-1 \quad 1 + \omega t]x. \end{aligned} \tag{16}$$

In the following experiment, we use  $\lambda = -1$  and  $\omega = 100$ .

1. Using the HERK methods based on the embedded Runge-Kutta pairs for the strangeness-free DAEs (15).

a) The embedded pair of Fehlberg 4(5). If we use the HERK methods with the original Runge-Kutta of orders 4 and 5, it may occur order reduction. By testing the numerical experiments of the HERK methods of orders 4 and 5 for the test DAEs with the uniform stepsize  $h$ , we obtain that the embedded methods with a pair of Fehlberg 4(5) is reduced to a pair of orders 1 and 2. So, we do not suggest the embedded method with a pair of Fehlberg 4(5) in this case.

b) The embedded pair of Dormand-Prince 4(5). By testing the numerical experiments of the HERK methods of orders 4 and 5 for the test DAEs with the uniform stepsize  $h$ . We obtain that the embedded methods with a pair of Dormand-Prince 4(5) is reduced to a pair of orders 2 and 3. Let sHERKDP4(5) denote the embedded methods with a pair of Dormand-Prince 4(5) for the strangeness-free DAEs. We denote that the estimate errors in  $x_i, i = 1, 2$  are the differences between two numerical solutions of sHERKDP4(5) method and the actual errors in  $x_i, i = 1, 2$  are the differences between exact solutions and numerical solutions.

Table 3. sHERKDP 4(5) method with  $TOL = 10^{-3}$  and initial stepsize  $h = 0.1$

The number of step (N)	103
Estimated error in $x_1$	5.5582e-04
Estimated error in $x_2$	1.1094e-06
Actual error in $x_1$	6.8593e-02
Actual error in $x_2$	4.6943e-04

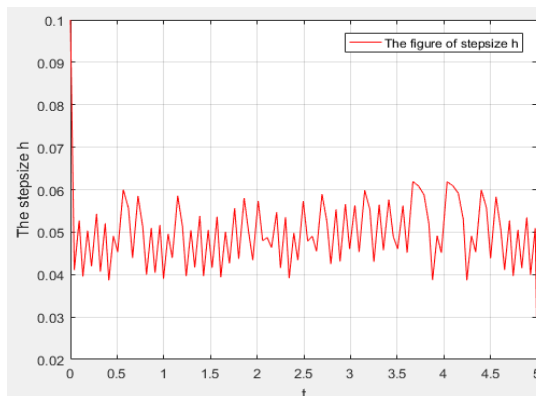


Figure 1. Step size  $h$  versus time  $t$  of sHERKDP 4(5) method.

For comparison, we carry out numerical experiments of the classical 4-stage Runge-Kutta method (HERK) on uniform stepsize  $h$  with Butcher tableau

$$\begin{array}{c|cccc}
 0 & 0 & 0 & 0 & 0 \\
 \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\
 \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\
 1 & 0 & 0 & 1 & 0 \\
 \hline
 & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6}
 \end{array}$$

Table 4. Numerical experiment of sHERK4 method with  $\lambda = -1, \omega = 100$

h=0.1	N	Actual error in $x_1$	Error order in $x_1$	Actual error in $x_2$	Error order in $x_2$
h	50	7.7337e-01	-	5.1928e-03	-
h/2	100	2.5623e-01	1.5937	1.7285e-03	1.5870
h/4	200	7.3599e-02	1.7997	4.9733e-04	1.7973
h/8	400	1.7193e-02	2.0979	1.1624e-04	2.0972
h/16	800	3.2530e-03	2.4019	2.1996e-05	2.4018
h/32	1600	5.2069e-04	2.6433	3.5208e-06	2.6432
h/64	3200	7.4628e-05	2.8026	5.0462e-07	2.8026
h/128	6400	1.0027e-05	2.8958	6.7804e-08	2.8958

The numerical results in Table 3 and Table 4 clearly illustrate that the convergent order of sHERK4 method is reduced from 4 to 3. For a given error tolerance  $TOL = 10^{-3}$ , the number of step of sHERKDP 4(5) method is much smaller than the number of step of HERK4 method.

The embedded Runge-Kutta methods combined with the HERK methods for structured strangeness-free DAEs (16) with initial stepsize  $h = 0.1$  and relative error  $RTOL = 10^{-7}$ . Let us denote the HERK method with the Fehlberg 4(5) pair by HERKF4(5) and the HERK method with the Dormand-Prince 4(5) pair by HERKDP4(5).

Table 5. Compare HERKF 4(5) method and HERKDP 4(5) method

h=0.1	HERKF 4(5)	HERKDP 4(5)
The number of step (N)	37	34
Estimated error in $x_1$	7.4083e-09	1.9075e-08
Estimated error in $x_2$	1.4787e-11	3.8073e-11
Actual error in $x_1$	3.0713e-06	1.6846e-06
Actual error in $x_2$	2.0024e-08	1.0969e-08

In Table 5, the difference between the numbers of step of HERKF 4(5) method and HERKDP 4(5) method is small. It shows that both of the methods are appropriate for solving (16).

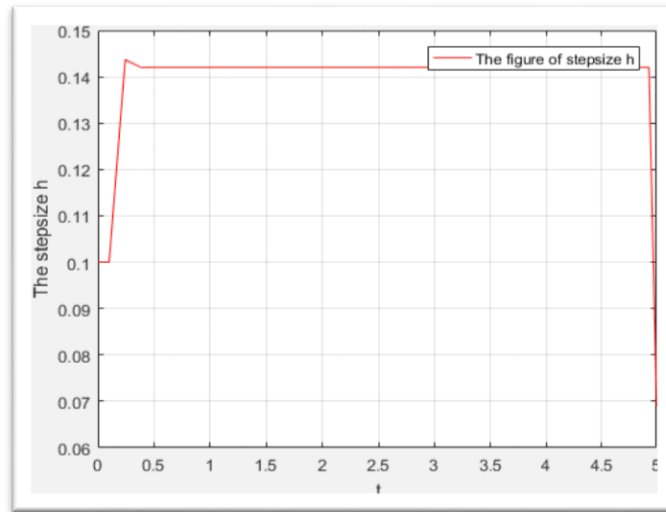


Figure 2. Step size  $h$  versus time  $t$  of HERKF 4(5) method with  $TOL = 10^{-7}$ .

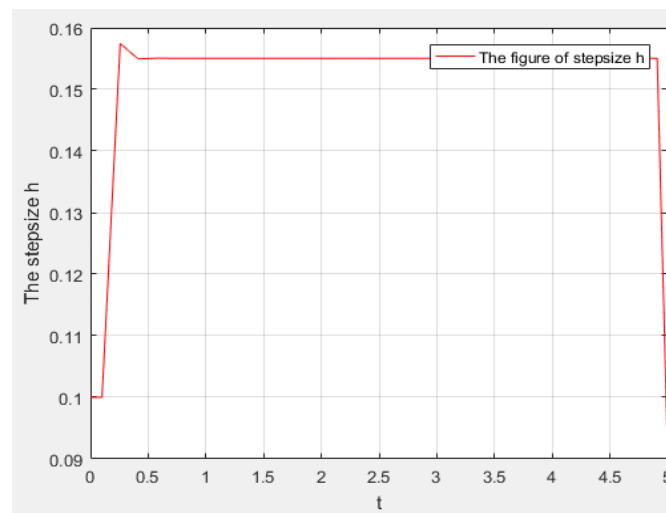


Figure 3. Step size  $h$  versus time  $t$  of HERKDP 4(5) method with  $TOL = 10^{-7}$ .

By using the absolute error  $ATOL$  to solve the DAEs (16), we obtain

Table 6. Compare HERKF 4(5) and HERKDP 4(5) with the absolute error  $ATOL = 10^{-7}$

h=0.1	HERKF 4(5)	HERKDP 4(5)
The number of step (N)	62	57
Estimated error in $x_1$	$4.2558e-08$	$1.1428e-08$
Estimated error in $x_2$	$8.4945e-11$	$2.2811e-11$
Actual error in $x_1$	$1.1870e-07$	$6.1959e-08$
Actual error in $x_2$	$1.0108e-09$	$5.3394e-10$

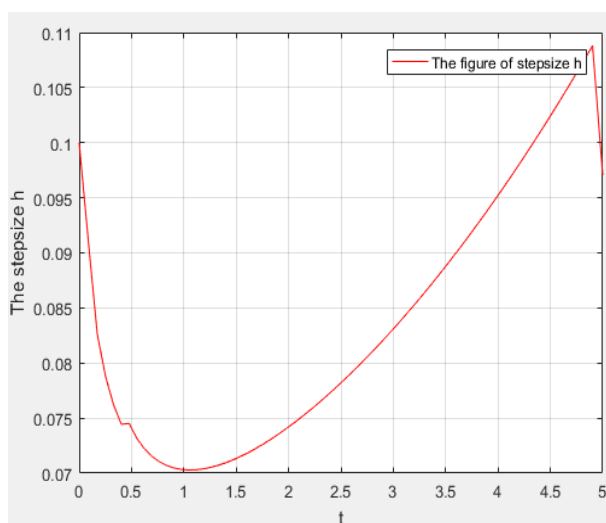


Figure 4. Stepsize  $h$  versus time  $t$  of HERKF 4(5) method with  $ATOL = 10^{-7}$ .

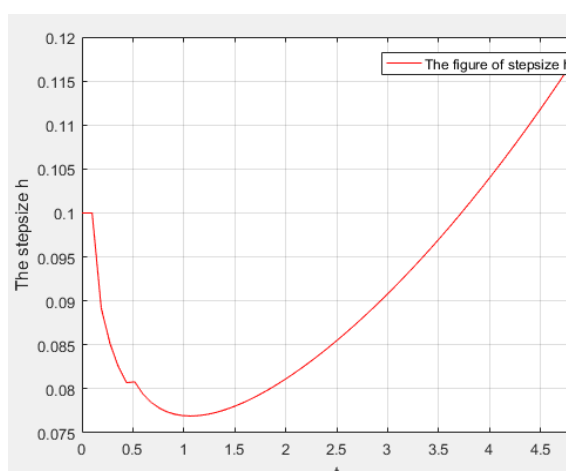


Figure 5: Stepsize  $h$  versus time  $t$  of HERKDP 4(5) method with  $ATOL = 10^{-7}$

We carry out numerical experiments of the classical 4-stage Runge-Kutta method (HERK4) on uniform stepsize  $h$ .

Table 7. Numerical experiment of HERK4 method with  $\lambda = -1, \omega = 100$

h=0.1	N	Actual error in $x_1$	Error order in $x_1$	Actual error in $x_2$	Error order in $x_2$
h	50	$4.9282e-05$	-	$3.3324e-07$	-
h/2	100	$2.9542e-06$	4.0602	$1.9976e-08$	4.0602
h/4	200	$1.8083e-07$	4.0301	$4.2228e-09$	4.0301
h/8	400	$1.1188e-08$	4.0146	$7.5633e-11$	4.0150
h/16	800	$6.9424e-10$	4.0104	$4.6971e-12$	4.0092
h/32	1600	$4.2586e-11$	4.0270	$3.0831e-13$	3.9293

In Table 7, the convergence order of the HERK4 method remains 4, the same as that for ODEs. For a given error tolerance  $TOL = 10^{-7}$ , both of HERKF 4(5) method and HERKDP 4(5) method are better than the HERK4 method because they require less steps.

**Example 2.** We consider the nonlinear DAE

$$\begin{aligned}x_1(x_1' + tx_2') &= x_1x_2e^t + e^{2t} + t \cos(t)e^t - e^{2t} \sin(t), \\0 &= e^{-t}x_1 - x_2 + \sin(t) - 1,\end{aligned}\tag{17}$$

for  $t \in [0, 5]$  with the initial condition  $x(0) = [1 \ 0]^T$ .

It is easy to check that the DAEs (17) is strangeness-free and the exact solution is  $x_1 = e^t, x_2 = \sin(t)$ .

The DAEs (17) can be written in the structured form

$$\begin{aligned}([1 \ t]x)' &= x_2 + x_2e^t + \frac{e^{2t} + t \cos(t)e^t - e^{2t} \sin(t)}{x_1}, \\0 &= e^{-t}x_1 - x_2 + \sin(t) - 1,\end{aligned}\tag{18}$$

Using the HERK methods based on the embedded Runge-Kutta pairs for the strangeness-free DAEs (17).

The embedded pair of Fehlberg4(5). If we use the HERK methods with the original Runge-Kutta of orders 4 and 5, it may occur order reduction. By testing the numerical experiments of the HERK methods of orders 4 and 5 for the DAEs (17) with the uniform stepsize  $h$ , we obtain that the embedded methods with a pair of Fehlberg 4(5) is reduced to a pair of orders 1 and 2, respectively. So, we do not suggest the embedded method with a pair of Fehlberg 4(5) in this case.

The embedded pair of Dormand-Prince 4(5). By testing the numerical experiments of the HERK methods of orders 4 and 5 for the DAEs (17) with the uniform stepsize  $h$ . We obtain that the embedded methods with a pair of Dormand-Prince 4(5) is reduced to a pair of orders 2 and 3, respectively. Let sHERKDP4(5) denote the embedded methods with a pair of Dormand-Prince 4(5) for the strangeness-free DAEs. We denote that the estimate errors in  $x_i, i = 1, 2$  are the differences between two numerical solutions of sHERKDP4(5) method and the actual errors in  $x_i, i = 1, 2$  are the differences between exact solutions and numerical solutions.

Table 8: sHERKDP4(5) method with  $TOL = 10^{-7}$  and initial stepsize  $h = 0.1$

The number of step (N)	58
Estimated error in $x_1$	$2.3272e-09$
Estimated error in $x_2$	$1.5680e-11$
Actual error in $x_1$	$1.4076e-05$
Actual error in $x_2$	$1.0817e-07$

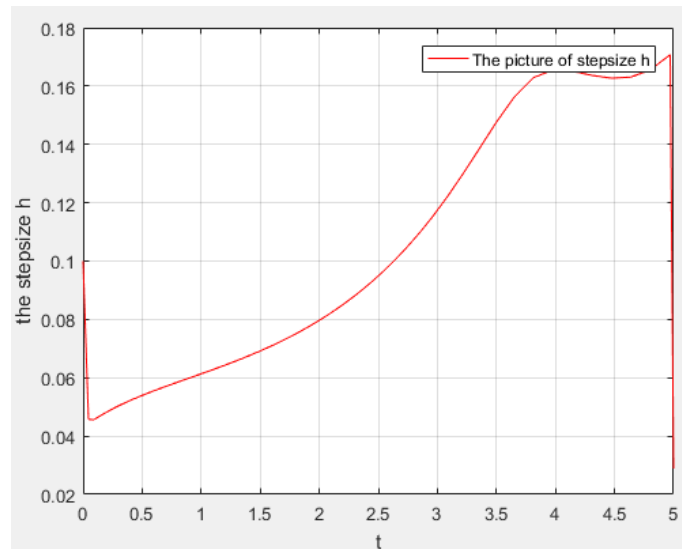


Figure 6. Stepsize  $h$  versus time  $t$  of sHERKDP 4(5) method.

Table 9. Numerical experiment of sHERK4 method

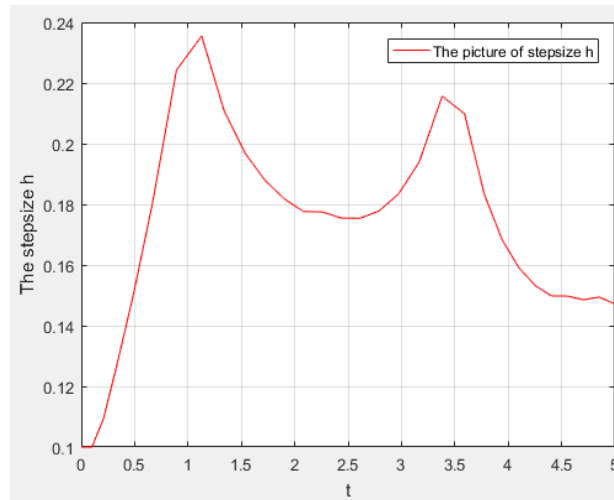
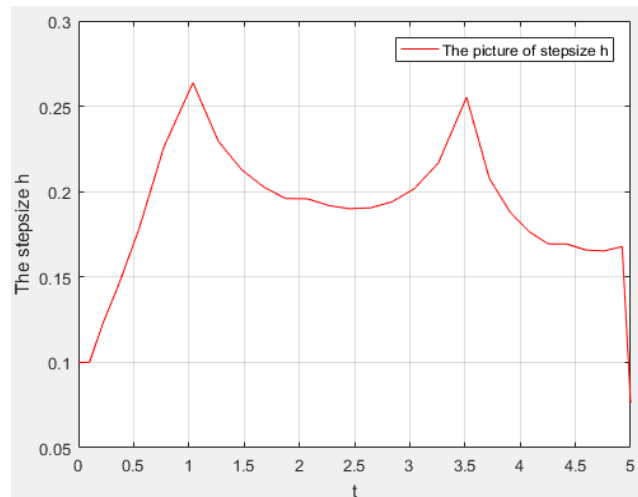
h=0.1	N	Actual error in $x_1$	Error order in $x_1$	Actual error in $x_2$	Error order in $x_2$
h	50	1.9264e-04	-	1.0937e-05	-
h/2	100	2.4929e-05	2.9500	1.3900e-06	2.9760
h/4	200	3.1727e-06	2.9741	1.7518e-07	2.9882
h/8	400	4.0036e-07	2.9864	2.1984e-08	2.9943
h/16	800	5.0289e-08	2.9930	2.7534e-09	2.9972
h/32	1600	6.3017e-09	2.9964	3.4451e-10	2.9986

The numerical results in Table 8 and Table 9 clearly illustrate that the convergent order of sHERK4 method is reduced from 4 to 3. For a given error tolerance  $TOL = 10^{-7}$ , the number of step of sHERKDP 4(5) method is smaller than the number of step of sHERK4 method.

Using the HERK methods based on the embedded Runge-Kutta pairs for structured strangeness-free DAEs (18) with initial stepsize  $h = 0.1$  and relative error  $TOL = 10^{-7}$ .

Table 10. Compare HERKF4(5) method and HERKDP4(5) method

h=0.1	HERKF4(5)	HERKDP4(5)
The number of step (N)	30	28
Estimated error in $x_1$	8.9672e-06	1.9691e-07
Estimated error in $x_2$	6.0420e-08	1.3268e-09
Actual error in $x_1$	9.9287e-06	1.4043e-05
Actual error in $x_2$	1.2034e-07	1.2430e-07

Figure 7. Stepsize  $h$  versus time  $t$  of HERKF 4(5) method.Figure 8: Stepsize  $h$  versus time  $t$  of HERKDP 4(5) method

We solve the DAEs (17) for HERK4 method on uniform stepsize  $h$ .

Table 11. Numerical experiment of HERK4 method

$h=0.1$	N	Actual error in $x_1$	Error order in $x_1$	Actual error in $x_2$	Error order in $x_2$
$h$	50	$2.4888e-04$	-	$1.6881e-06$	-
$h/2$	100	$1.5560e-05$	3.9995	$1.0549e-07$	4.0001
$h/4$	200	$9.7288e-07$	3.9995	$6.5939e-09$	3.9999
$h/8$	400	$6.0819e-08$	3.9997	$4.1216e-10$	3.9999
$h/16$	800	$3.8016e-09$	3.9998	$2.5761e-11$	3.9999
$h/32$	1600	$2.3780e-10$	3.9998	$1.6122e-12$	3.9981

In Table 10 and 11, the convergence order of the HERK4 method remains 4, the same as that for ODEs. For a given error tolerance  $TOL = 10^{-7}$ , both of HERKF 4(5) method and HERKDP 4(5) method are better than the HERK4 method.

#### 4. Conclusion

In this paper, we have constructed the half-explicit Runge-Kutta methods based on embedded pairs such that Fehlberg and Dormand-Prince for solving strangeness-free DAEs and structured strangeness-free DAEs. Thus, the advantage of the embedded Runge-Kutta pairs can be exploited. These methods work with error control and automatic stepsize selection. It is shown that the embedded Dormand-Prince method solves efficiently the strangeness-free DAEs (1). However, the embedded Fehlberg method is less efficient for solving the strangeness-free DAEs (1). Both of the embedded Dormand-Prince method and the embedded Fehlberg method are more efficient than the HERK4 method for solving the structured strangeness-free DAEs (3). Since the underlying methods are explicit, the proposed algorithms are recommendable for solving non-stiff DAEs.

#### Acknowledgments

This work was supported by The college of Artillery Officer's Training and Vietnam university of Science.

#### References

- [1] U. Ascher, L. Petzold, Computer methods for Ordinary differential equations and differential-algebraic equations, SIAM, Philadelphia, 1998.
- [2] P. Kunkel, V. Mehrmann, Differential-Algebraic Equations Analysis and Numerical Solution, European Mathematical Society, Zurich, 2006.
- [3] V.H. Linh, V. Mehrmann, Efficient integration of strangeness-free non-stiff differential-algebraic equations by half-explicit methods, J. Comput. Appl. Math. 262 (2014) 346-360.
- [4] V.H. Linh, N.D. Truong, Runge-Kutta methods revisited for a class of structured strangeness-free differential-algebraic equations, Electr. Trans. Num. Anal. 48 (2018) 131-135.