UNIVERSITY OF CAMBRIDGE

Numerical Analysis Reports

Numerical solution of implicit neutral functional differential equations

Yunkang Liu

DAMTP 1997/NA4
March 1997

Department of Applied Mathematics and Theoretical Physics
Silver Street
Cambridge
England CB3 9EW

Numerical solution of implicit neutral functional differential equations

Yunkang Liu

Gonville and Caius College, Cambridge University, Cambridge CB2 1TA, U.K

Abstract. This paper is concerned with the numerical solution of implicit neutral functional differential equations. Based on the continuous Runge–Kutta method (for ordinary differential equations) and the collocation method (for functional equations), two general one-step methods are formulated and their uniform order of approximation are discussed. Numerical stability of a class of Runge–Kutta-Collocation methods is analysed. In the case where a neutral equation can be expressed in an explicit as well as an implicit form the advantage of constructing a numerical scheme directly from the implicit form is explained theoretically and demonstrated by a numerical example.

Key words. Neutral functional differential equation, functional equation, Runge–Kutta method, collocation method, stability analysis

AMS subject classifications. 65L05, 65L06, 65L60, 65L20, 34K20

1 Introduction

Neutral functional differential equations with one state-independent time delay are usually formulated in the form

$$y'(t) = f(t, y(t), y(\phi(t)), y'(\phi(t))), \qquad t \geq t_0, \tag{1.1}$$

where f and ϕ are given functions with $\phi(t) \leq t$ for $t \geq t_0$. Variations of (1.1) include equations with multiple time delays:

$$y'(t) = f(t, y(t), y(\phi_1(t)), \ldots, y(\phi_m(t)), y'(\phi_1(t)), \ldots, y'(\phi_m(t))),$$

equations with state-dependent time delay:

$$y'(t)) = f(t, y(t), y(\phi(t, y(t))), y'(\phi(t, y(t))))$$

and equations with distributional delay:

$$y'(t) = f\left(t, y(t), \int_{[t_0,t]} y \,\mathrm{d}\mu, \int_{[t_0,t]} y' \,\mathrm{d}\nu\right),$$

where μ and ν are Lebesgue measures. In contrast to (1.1) there are neutral functional differential equations of the form

$$[y(t) - g(t, y(\phi(t)))]' = f_0(t, y(t), y(\phi(t))), \qquad t \geq t_0, \tag{1.2}$$

where f_0, g and ϕ are given functions with $\phi(t) \leq t$ for $t \geq t_0$. To distinguish (1.2) from (1.1), we call (1.1) an explicit neutral equation and (1.2) an implicit neutral equation. This is because in (1.1) the derivative y' at any fixed time $t \geq t_0$ depends in an explicit manner on itself at the past time $\phi(t)$, whereas in the case of (1.2) this kind of dependence is implicit and it becomes explicit only upon rewriting (1.2) in the form

$$y'(t) = f_1(t, y(t), y(\phi(t)), y'(\phi(t))), \qquad t \geq t_0, \tag{1.3}$$

where f_1 is defined by

$$f_1(t, x_1, x_2, x_3) = f_0(t, x_1, x_2) + \frac{\partial}{\partial t} g(t, x_2) + \phi'(t) \frac{\partial}{\partial x_2} g(t, x_2) x_3,$$

provided that g and ϕ are differentiable functions and that, most importantly, we are dealing with differentiable solutions of (1.2). Numerical solution for the explicit neutral equation (1.1) has been studied extensively in numerous papers such as [4, 13, 17, 18, 19]. Some variations of (1.2) have been discussed by Ito and Turi [16] and Hartung, Herdman and Turi [12], using special numerical schemes. By and large, the implicit neutral equation (1.2) has received relatively little attention. One possible reason for the lack of interest in (1.2) is that in certain applications the functions g and ϕ are smooth, which means that (1.2) can be formally rewritten as (1.3), a special case of (1.1). However, under careful scrutiny, one may find that equation (1.2) is not necessarily equivalent to (1.3) even if g and ϕ are differentiable, noting that a non-differentiable function can be a solution of (1.2). We refer to the monographs of Hale and Verduyn Lunel [11] and of Kolmanovskii and Myshkis [20] for detailed discussion of this issue.

Even if equations (1.2) and (1.3), in tandem with proper restriction such as an initial condition, are equivalent, there are certain advantages in constructing numerical schemes based on (1.2). Suppose that both equations have the same solution $y(t) \in C^m(\mathbb{R})$ for some $m \in \mathbb{N}$. As functions of t the right hand sides of (1.2) and (1.3) are in the function spaces $C^m(\mathbb{R})$ and $C^{m-1}(\mathbb{R})$, respectively, assuming that the functions f_0 and g are sufficiently smooth. The advantage of using (1.2) instead of (1.3) is clear, since the accuracy of a traditional numerical method depends very much on the smoothness of the functions involved. Bearing in mind that solutions of neutral functional differential equations are not smooth in general, a slight increase of smoothness of the functions involved can make a big difference. Furthermore, when solving a functional differential equation it is usually necessary to find for the exact solution $y(t)$, a piecewise polynomial approximant, $y^h(t)$. In the case of (1.3) there is an order reduction in approximating $y'(\phi(t))$ by $(y^h)'(\phi(t))$. It is our belief that one should construct numerical scheme on (1.2) directly. In other words, if equation (1.1) can be written in the form of (1.2), one should take advantage of it. We remark in passing that most theoretical studies of neutral equations, especially their stability analysis, focus on the implicit ones (see, e.g., the book of Hale and Verduyn Lunel [11]).

It is perhaps relevant to recall that a similar situation occurs in the numerical solution of partial differential equations. According to Morton and Mayers [26, p. 48], when solving the parabolic equation

$$u_t = (p(x,t)u_x)_x, \tag{1.4}$$

it is usually better to construct a numerical scheme (based on a Galerkin method or a difference method, say) directly from (1.4) rather than from its equivalent form (assuming that $p(x,t)$ is differentiable with respect to x)

$$u_t = p(x,t)u_{xx} + p_x(x,t)u_x.$$

This paper is concerned with the initial value problem of (1.2), namely, finding a function y that satisfies the initial condition

$$y(t) = Y(t), \qquad t \le t_0, \tag{1.5}$$

where Y is an appropriate initial function, and that $y(t)-g(t,y(\phi(t)))$ is differentiable and obeys equation (1.2) for $t \ge t_0$. In Section 2, we transform equation (1.2) into a system that consists a functional equation and a retarded functional differential equation. Two general one-step numerical methods that are based on the standard continuous Runge–Kutta scheme (for ordinary differential equations) and the collocation method (for functional equations) are formulated and analysed. A simple numerical example is given to demonstrate the advantage of solving the initial value problem (1.2,1.5) instead of (1.3,1.5). The final section discusses "A-stability" of those Runge–Kutta–Collocation methods formulated in Section 2. The test model is a globally stable system of linear neutral functional differential equations with constant delay. It will be shown that the stability property of the exact solution is inherited by numerical solutions of a large class of Runge–Kutta–Collocation methods on a constraint mesh.

2 Numerical methods

Let $z(t) = y(t) - g(t,y(\phi(t)))$. Then equation (1.2) is equivalent to the hybrid system

$$y(t) = z(t) + g(t,y(\phi(t))), \qquad t \ge t_0, \tag{2.1}$$

$$z'(t) = f_0(t,y(t),y(\phi(t))), \qquad t \ge t_0, \tag{2.2}$$

Substituting (2.1) into (2.2) yields yet another equivalent system

$$y(t) = z(t) + g(t,y(\phi(t))), \qquad t \ge t_0, \tag{2.3}$$

$$z'(t) = f(t,z(t),y(\phi(t))), \qquad t \ge t_0, \tag{2.4}$$

where f is defined by

$$f(t,x_1,x_2) = f_0(t,x_1+g(t,x_2),x_2).$$

We shall formulate two classes of numerical methods, one based on (2.1,2.2), the other on (2.3,2.4).

Let $t_0 < t_1 < t_2 < \dots$ be a mesh in the integration interval $[t_0, T]$ for some $T > 0$. Before formulating our numerical method for solving (1.2,1.5) we recall that the s-stage Runge–Kutta method defined by the Butcher tableau

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array} \tag{2.5}$$

for solving the initial value problem

$$x'(t) = f(t, x(t)), \quad t \geq t_0, \qquad x(0) = x_0 \tag{2.6}$$

can be written as

$$x_{n+1} = x_n + h_n \sum_{i=1}^{s} b_i K_{n,i}, \qquad n \geq 0,$$

where $h_n = t_{n+1} - t_n$, and $K_{n,1}, \dots, K_{n,s}$ are determined by the system of algebraic equations

$$K_{n,i} = f\left(t_n + c_i h_n, x_n + h_n \sum_{j=1}^{s} a_{ij} K_{n,j}\right), \qquad i = 1, \dots, s.$$

A continuous extension of this Runge–Kutta method gives a continuous numerical solution of the form

$$x^h(t_n + \tau h_n) = x_n + h_n \sum_{i=1}^{s} b_i(\tau) M_{n,i}, \qquad \tau \in [0, 1].$$

where $b_i(\tau)$, $i = 1, 2, \dots, s$, are polynomials that obey the conditions

$$b_i(0) = 0, \quad b_i(1) = b_i, \qquad i = 1, 2, \dots, s.$$

A typical example is the collocation Runge–Kutta method. Other continuous Runge–Kutta methods can be found in Zennaro [29], Hairer, Nørsett and Wanner [9] and in Owren and Zennaro [27]. We mention in passing that one can also use the Hermite interpolation [28] and the boot-strapping process [8] to generate continuous numerical solutions. A positive integer m is said to be the *uniform order* of the Runge–Kutta method if

$$x^h(t) - x(t) = \mathcal{O}(h_0^{m+1}), \qquad t \in [t_0, t_1].$$

We now return to the initial value problem (1.2,1.5). Let

$$y^h(t) = Y(t), \qquad t \leq t_0$$

and

$$y_0 = Y(t_0), \qquad z_0 = Y(t_0) - g(t_0, Y(\phi(t_0))).$$

Suppose that we have $y^h(t) \approx y(t)$ and $z^h(t) \approx z(t)$ on the interval $[t_0, t_n]$ for some $n \in \mathbb{Z}^+$. Then a continuous numerical solution y^h of (1.2,1.5) on the interval $[t_n, t_{n+1}]$ can be obtained by letting

$$y^h(t_n + \tau h_n) = \sum_{i=1}^{s^*} B_i(\tau) Y_{n,i}, \tag{2.7}$$

$$z^h(t_n + \tau h_n) = z_n + h_n \sum_{i=1}^{s} b_i(\tau) M_{n,i}, \tag{2.8}$$

for $\tau \in [0, 1]$, where

$$B_i(\tau) = \prod_{j \neq i} \frac{\tau - c_j^*}{c_i^* - c_j^*}, \qquad i = 1, 2, \ldots, s^*,$$

with $c_i^* \in (0, 1)$ for $i = 1 \ldots, s^* - 2$, $c_{s^*-1}^* = 0$, $c_{s^*}^* = 1$, the functions b_i are defined by the continuous Runge–Kutta method, $Y_{n,i}$ is our approximation to the exact solution $y(t)$ at $t = t_n + c_i^* h_n$ for $i = 1, 2, \ldots, s^*$, and $M_{n,i}$ is our approximation to $z'(t)$ at $t_n + c_i h_n$ for $i = 1, 2, \ldots, s$. While $Y_{n,i}$, $i = 1, \ldots, s^*$, are determined by the system of algebraic equations

$$Y_{n,i} = z^h(t_n + c_i^* h_n) + g(t_n + c_i^* h_n, y^h(\phi(t_n + c_i^* h_n))), \qquad i = 1, \ldots, s^*. \tag{2.9}$$

$M_{n,i}$, $i = 1, \ldots, s$, are chosen to satisfy the system

$$M_{n,i} = f_0\left(t_n + c_i h_n, y^h(t_n + c_i h_n), y^h(\phi(t_n + c_i h_n))\right), \qquad i = 1, \ldots, s \tag{2.10}$$

or

$$M_{n,i} = f\left(t_n + c_i h_n, z_n + h_n \sum_{j=1}^{s} a_{ij} M_{n,j}, y^h(\phi(t_n + c_i h_n))\right), \qquad i = 1, \ldots, s, \tag{2.11}$$

depending on whether we use (2.2) or (2.4). We call the methods that use (2.10) and (2.11), respectively, RKC(I) and RKC(II). In general, RKC(I) and RKC(II) are different. However, they are the same if $\{c_i\}_{i=1}^s \subset \{c_i^*\}_{i=1}^{s^*}$ and the corresponding continuous Runge–Kutta method is *natural* in the sense that the functions b_i, $i = 1, 2, \ldots, s$, satisfy

$$b_i(c_j) = a_{ji}, \qquad i, j = 1, 2, \ldots, s.$$

Evidently, (2.7) is based on the collocation method with the collocation points $t_n + c_i^* h_n$, $i = 1, 2, \ldots, s^*$, since (2.1) is satisfied (assuming that $z^h(t) = z(t)$) at these points. It should be noted that we choose $c_{s^*-1}^* = 0$ and $c_{s^*}^* = 1$ in order that y^h is continuous at the mesh points. Theoretically, one can define y^h directly from (2.3) in the following way:

$$y^h(t) = z^h(t) + g(t, y^h(\phi(t))), \qquad t \geq t_0.$$

However, it is expensive to realise it numerically unless the functions Y, g and ϕ are very simple so that y^h can have an explicit analytic representation. An alternative to

(2.7) is

$$y^h(t_n + \tau h_n) = y^h(t_n) + h_n \sum_{i=1}^{s^*} W_{n,i} \int_0^\tau B_i(\xi)\,\mathrm{d}\xi, \qquad \tau \in [0,1],$$

where $W_{n,i}$ is our approximation to $y'(t)$ at the collocation point $t_n + c_i^* h_n$, $i = 1, \ldots, s^*$. The advantage of this approach is that the continuity of y^h at mesh points is then automatically built in. The disadvantage is that the corresponding system of equations that determines $W_{n,i}$, $i = 1, \ldots, s^*$, is not as simple as (2.9), if (2.1) is required to be satisfied at the collocation points.

If there exists $m \in \mathbb{N}$ such that

$$y^h(t) - y(t) = \mathcal{O}(h^m), \qquad t \in [t_0, T],$$

where y^h is the numerical solution generated by RKC(I) (or RKC(II)), then the *uniform order* of RKC(I) (or RKC(II)) is said to be m. In the sequel, we assume for simplicity that the solution y of (1.2) is sufficiently smooth except possibly at the mesh points and that equation (1.2) has no vanishing delay, namely, $t - \phi(t) > 0$ for every $t \in [t_0, T]$. The last assumption implies that one can choose $h_n := t_{n+1} - t_n$ small enough such that

$$\phi(t) \le t_n - \frac{1}{2} \max_{t \in [t_0, T]} [t - \phi(t)], \qquad t \in (t_n, t_{n+1}).$$

Noting that y^h is a Lagrange interpolation, it is not difficult to show by using the method-of-steps that the following result holds for RKC(II).

Theorem 1 *Suppose that the uniform order of the continuous Runge–Kutta method defined by the Butcher tableau (2.5) is p^*. Then the uniform order of RKC(II) is* $\min\{p^*, s^*\}$.

Analogously, the following holds for RKC(I):

Theorem 2 *Suppose that the uniform order of the continuous Runge–Kutta method defined by the Butcher tableau*

$$\begin{array}{c|c} c & (b_i(c_j))_{s \times s} \\ \hline & b^T \end{array} \tag{2.12}$$

is p^*, and the degree of the polynomial b_i is less than s^*, $i = 1, 2, \ldots, s$. Then the uniform order of RKC(I) is $\min\{p^*, s^*\}$.

When applied to the initial value problem (2.6) the continuous Runge-Kutta method defined by the Butcher tableau (2.12) yields

$$x^h(t_n + \tau h_n) = x_n + h_n \sum_{i=1}^{s} b_i(\tau) M_{n,i}, \qquad \tau \in [0,1],$$

with $M_{n,i}$, $i = 1, \ldots, s$, being determined by the system of equations

$$M_{n,i} = f(t_n + c_i h_n, x^h(t_n + c_i h_n)), \qquad i = 1, \ldots, s.$$

To prove the preceding theorem, we observe that $y^h - z^h$ can be regarded as the Lagrange interpolation of

$$\left(t_n + c_i^* h_n, g(t_n + c_i^* h_n, y^h(\phi(t_n + c_i^* h_n)))\right), \qquad i = 1, 2, \ldots, s^*.$$

Replacing $y^h(t_n + c_i h_n)$ in (2.10) with

$$z^h(t_n + c_i h_n) + \left[(y^h(t_n + c_i h_n) - z^h(t_n + c_i h_n)) - g(t_n + c_i h_n, y^h(\phi(t_n + c_i h_n)))\right]$$

and noting that the second term is $\mathcal{O}(h^{s^*})$, one can see that the uniform order of RKC(I) is $\min\{p^*, s^*\}$.

A special case of the RKC(II) method is the linear θ-method

$$\begin{aligned} y_{n+1} = y_n + h_n &\left[(1-\theta) f_0(t_n, y_n, y^h(\phi(t_n))) + \theta f_0(t_{n+1}, y_{n+1}, y^h(\phi(t_{n+1})))\right] \\ &+ g(t_{n+1}, y^h(\phi(t_{n+1}))) - g(t_n, y^h(\phi(t_n))), \qquad n \geq 0 \end{aligned} \tag{2.13}$$

where $\theta \in [0, 1]$ is a parameter and

$$y^h(t) = \begin{cases} Y(t), & t \leq t_0, \\ \frac{t_{n+1} - t}{h_n} y_n + \frac{t - t_n}{h_n} y_{n+1}, & t \in [t_n, t_{n+1}], \ n \in \mathbb{Z}^+. \end{cases} \tag{2.14}$$

The continuous Runge–Kutta component of the linear θ-method corresponds to the Butcher tableau

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1-\theta & \theta \\ \hline & 1-\theta & \theta \end{array}$$

with $b_1(\tau) = (1 - \theta)\tau$, $b_2(\tau) = \theta\tau$. Its collocation part corresponds to the method with $c_1^* = 0$ and $c_2^* = 1$. We call this method the trapezoidal rule when $\theta = 1/2$.

The following one-leg θ-method is a special case of the RKC(I):

$$\begin{aligned} y_{n+1} = y_n + h_n f_0(t_n + \theta h_n, &(1-\theta) y_n + \theta y_{n+1}, y^h(\phi(t_n + \theta h_n))) \\ &+ g(t_{n+1}, y^h(\phi(t_{n+1}))) - g(t_n, y^h(\phi(t_n))), \qquad n \geq 0 \end{aligned} \tag{2.15}$$

where θ and y^h are the same as in the case of linear θ-method. The continuous Runge–Kutta component of the one-leg θ-method corresponds to the Butcher tableau

$$\begin{array}{c|c} \theta & \theta \\ \hline & 1 \end{array}$$

with $b_1(\tau) \equiv \tau$. This method is called the midpoint rule when $\theta = 1/2$.

Since the solution of (1.2) satisfies the identity

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f_0(\tau, y(\tau), y(\phi(\tau)))\,\mathrm{d}\tau + g(t_{n+1}, y(\phi(t_{n+1}))) - g(t_n, y(\phi(t_n))),$$

the connection of the linear θ-method and the one-leg θ-method, respectively, with the quadratures

$$\int_{t_n}^{t_{n+1}} f(\tau)\,\mathrm{d}\tau \approx h_n[(1-\theta)f(t_n) + \theta f(t_{n+1})]$$

and

$$\int_{t_n}^{t_{n+1}} f(\tau)\,\mathrm{d}\tau \approx h_n f(t_n + \theta h_n)$$

are obvious.

Consider as an example the initial value problem

$$\begin{cases} [y(t) - y(qt)]' = -y(t) + f(t), & 1 \le t \le 10 \\ y(t) = \sin(t), & t \le 1, \end{cases} \tag{2.16}$$

where $q = 3/\pi$ and $f(t) = \cos(t) - q\cos(qt) + \sin(t)$. The exact solution of (2.16) is $y(t) = \sin(t)$. The above initial value problem is equivalent to

$$\begin{cases} y'(t) = -y(t) + qy'(qt) + f(t), & 1 \le t \le 10 \\ y(t) = \sin(t), & t \le 1. \end{cases} \tag{2.17}$$

Let us solve (2.16) and (2.17) by both the trapezoidal rule and the midpoint rule on a uniform mesh, i.e., $t_n = 1 + nh$, $n = 0, 1, \ldots$, where h is the stepsize. In the case of (2.16), the trapezoidal rule gives the recurrence relation

$$y_{n+1} = \frac{2-h}{2+h} y_n + \frac{2}{2+h}[y^h(qt_{n+1}) - y^h(qt_n)] + \frac{h}{2+h}[f(t_n) + f(t_{n+1})], \qquad n = 0, 1, \ldots$$

while the midpoint rule yields the recurrence relation

$$y_{n+1} = \frac{2-h}{2+h} y_n + \frac{2}{2+h}[y^h(qt_{n+1}) - y^h(qt_n)] + \frac{2h}{2+h} f((n+1/2)h), \qquad n = 0, 1, \ldots .$$

where

$$y^h(t) = \begin{cases} \sin(t), & t \le 1, \\ \frac{t_{n+1}-t}{h} y_n + \frac{t-t_n}{h} y_{n+1}, & t \in (t_n, t_{n+1}],\ n = 0, 1, \ldots . \end{cases}$$

In the case of (2.17), approximating $y'(qt)$ by $(y^h)'(qt)$, the trapezoidal rule gives the recurrence relation

$$\begin{aligned} y_{n+1} = \frac{2-h}{2+h} y_n + \frac{qh}{2+h}[w^h(qt_n) + w^h(qt_{n+1})] \\ + \frac{h}{2+h}[f(t_n) + f(t_{n+1})], \qquad n = 0, 1, \ldots \end{aligned}$$

and the midpoint rule results in the recurrence relation

$$\begin{aligned} y_{n+1} = \frac{2-h}{2+h} y_n + \frac{2qh}{2+h} w^h(q(n+1/2)h) \\ + \frac{2h}{2+h} f((n+1/2)h), \qquad n = 0, 1, \ldots, \end{aligned}$$

where w^h is defined as

$$w^h(t) = \begin{cases} \cos(t), & t \le 1, \\ \frac{y_{n+1} - y_n}{h}, & t \in [t_n, t_{n+1}),\ n = 0, 1, \ldots. \end{cases}$$

The following two tables list $\max_{t_n \in [1,10]} |y(t) - y_n|$, the maximal errors at mesh points of the trapezoidal rule and the midpoint rule when they are applied to the equivalent initial value problems (2.16) and (2.17).

The trapezoidal rule

h	1/50	1/100	1/200	1/400
Eq. (2.16)	1.291×10^{-4}	3.836×10^{-5}	1.225×10^{-5}	3.651×10^{-6}
Eq. (2.17)	1.8×10^{-3}	8.579×10^{-4}	3.139×10^{-4}	9.184×10^{-5}

The midpoint rule

h	1/50	1/100	1/200	1/400
Eq. (2.16)	1.586×10^{-4}	4.134×10^{-5}	1.377×10^{-5}	3.846×10^{-6}
Eq. (2.17)	2.1×10^{-3}	1.0×10^{-3}	2.899×10^{-4}	1.189×10^{-4}

It is obvious that the numerical solutions obtained by solving the initial value problem (2.16) instead of (2.17) are more accurate. Less obvious are the orders of these methods. When solving (2.16), the average error reduction for the trapezoidal rule and the midpoint rule are about 3.28 and 3.47, respectively. In contrast, when solving (2.17), the average error reduction for the trapezoidal rule and the midpoint rule are about 2.75 and 2.66, respectively.

If (1.2) has vanishing delay it is perhaps necessary to assume that the function $g(t, \cdot)$ is Lipschitz continuous and the corresponding Lipschitz constant is small. We shall not discuss the vanishing delay case in this paper, but refer instead to the papers by Jackiewicz [18], Enright and Hu [7] and Hayashi [13] for their study of equations with general variable delay, and to the papers by Brunner [5] and Liu [24] for treatment of equations with proportional delay.

3 Stability analysis

This section investigates the stability of some special cases of the RKC method introduced in the previous section. The test equation is of the form

$$[y(t) - By(t-r)]' = A_0 y(t) + A_1 y(t-r), \qquad t \geq 0, \tag{3.1}$$

where $r > 0$, A_0, A_1 and B are $d \times d$ matrices such that (3.1) is *stable globally* [10], i.e., its zero solution is asymptotically stable for every $r > 0$. Equation (3.1) is equivalent to

$$y(t) = z(t) + By(t-r), \qquad t \geq 0, \tag{3.2}$$

$$z'(t) = A_0 y(t) + A_1 y(t-r), \qquad t \geq 0, \tag{3.3}$$

as well as to

$$y(t) = z(t) + By(t-r), \qquad t \geq 0, \tag{3.4}$$

$$z'(t) = A_0 z(t) + A_2 y(t-r), \qquad t \geq 0, \tag{3.5}$$

where $A_2 = A_1 + A_0 B$.

Having specified the test equation we now describe the RKC(I) and RKC(II) methods whose stability we intend to analyse under the following conditions:

1. The s-stage Runge–Kutta method defined by the Butcher tableau (2.5) is nonconfluent, i.e., $c_i \neq c_j$ if $i \neq j$; and its continuous extension is *natural* in the sense that the polynomials b_i, $i = 1, 2, \ldots, s$, satisfy

$$b_i(c_j) = a_{ji}, \qquad i, j = 1, 2, \ldots, s; \tag{3.6}$$

2. The set of collocation points of the collocation method for (2.3) in the interval $[t_n, t_{n+1}]$ is $\{t_n + c_i h, i = 1, \ldots, s\} \bigcup \{t_n, t_{n+1}\}$.

These two assumptions imply that RKC(I) and RKC(II) are equivalent for the test model.

Theorem 3 *If the Runge–Kutta method (2.5) is A-stable then in the case of a constrained mesh, namely, $t_n = nh$ for $n \geq 0$ with $h = r/m$ for some positive integer m, the numerical solution y_n of the test model tends to zero as $n \to \infty$.*

Before proving the preceding theorem, we discuss some properties of those matrix coefficients of the test equation (3.1). The characteristic function of (3.1) is $P(z, \mathrm{e}^{-rz})$ with the function P defined by

$$P(z, \xi) = \det[(I - B\xi)z - (A_0 + A_1 \xi)], \qquad z, \xi \in \mathbb{C}.$$

In the sequel, $\sigma[A]$, $\rho(A)$ and $\alpha[A]$ for a matrix A designate the spectrum, spectral radius and maximal real parts of the eigenvalues of A, respectively. Other notation includes

$$\mathbb{C}^+ = \{z \in \mathbb{C} \mid \operatorname{Re} z > 0\}, \;\; \mathbb{C}^0 = \{z \in \mathbb{C} \mid \operatorname{Re} z = 0\}, \;\; \mathbb{C}^- = \{z \in \mathbb{C} \mid \operatorname{Re} z < 0\},$$
$$D = \{z \in \mathbb{C} \mid |z| < 1\}, \quad \Gamma = \{z \in \mathbb{C} \mid |z| = 1\}.$$

Since equation (3.1) is stable globally, a general result of Hale, Infante and Tsen [10] implies that the following three conditions hold:

(H_1) $\rho(B) < 1$,

(H_2) $P(z,\xi) \neq 0$ for all $z \in \mathbb{C}^0 \backslash \{0\}$ and $\xi \in \Gamma$,

(H_3) $\alpha[(I-B)^{-1}(A_0+A_1)] < 0$.

The last condition implies that the zero solution of (3.1) is asymptotically stable when $r = 0$. Moreover, the following are also true:

(H_4) $\alpha[A_0] < 0$,

(H_5) $P(z,\xi) \neq 0$ for all $z \in \mathbb{C}^0$ and $\xi \in D$,

(H_6) $P(z,\xi) \neq 0$ for all $z \in \mathbb{C}^+$ and $\xi \in D \cup \Gamma$.

The first two, i.e, (H_4) and (H_5), are proved in [10]. In the special case $B = 0$ these results have been obtained independently by in 't Hout [15]. We now prove (H_6) by contradiction. Suppose that there exist $z_0 \in \mathbb{C}^+$ and $\xi_0 \in D \cup \Gamma$ such that $P(z_0, \xi_0) = 0$. Let

$$\omega(t) = \max\{\operatorname{Re} z \mid P(z, t\xi_0) = 0\}.$$

We have $\omega(0) = \alpha[A_0] < 0$ and $\omega(1) \geq \operatorname{Re} z_0 > 0$. Hence, there exists $t_0 \in (0,1)$ such that $\omega(t_0) = 0$, which contradicts (H_5). In the special case $B = 0$ the statement (H_6) has been proved by Koto [21]. The proof of the general case presented here is simpler than, though in principle the same as, that of Koto's.

Proof of Theorem 3. The proof is similar to the one by Koto [21] for a system of retarded functional differential equations. We have

$$y_{n+1} = \sum_{i=1}^{s^*} B_i(1) Y_{n,i}, \tag{3.7}$$

$$z_{n+1} = z_n + h \sum_{i=1}^{s} b_i M_{n,i}, \tag{3.8}$$

where $Y_{n,i}$, $i = 1, \dots, s^*$, satisfy

$$Y_{n,i} = \begin{cases} z_n + h \sum_{j=1}^{s} a_{ij} M_{n,j} + B Y_{n-m,i}, & i = 1, 2, \dots, s \\ z_n + B y_{n-m}, & i = s^* - 1 \\ z_{n+1} + B y_{n+1-m}, & i = s^* \end{cases} \tag{3.9}$$

and $M_{n,i}$, $i = 1, \dots, s$, satisfy

$$M_{n,i} = A_0 Y_{n,i} + A_1 Y_{n-m,i}, \qquad i = 1, 2, \dots, s \tag{3.10}$$

if we use RKC(I), or

$$M_{n,i} = A_0 \left(z_n + h \sum_{j=1}^{s} a_{ij} M_{n,j} \right) + A_2 Y_{n-m,i}, \qquad i = 1, 2, \dots, s \tag{3.11}$$

if we use RKC(II). Because of (3.9), (3.10) is equivalent to (3.11). Let

$$X_n = (z_{n+1}^T, M_{n,1}^T, \ldots, M_{n,s}^T, Y_{n,1}^T, \ldots, Y_{n,s}^T)^T.$$

and

$$L_0 = \begin{pmatrix} I_d & -hb^T \otimes I_s & 0 \\ 0 & I_{sd} - hA \otimes A_0 & 0 \\ 0 & -hA \otimes I_d & I_d \end{pmatrix}, \qquad L_1 = \begin{pmatrix} I_d & 0 & 0 \\ e_s \otimes A & 0 & 0 \\ e_s \otimes I_d & 0 & 0 \end{pmatrix},$$

$$L_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & I_d \otimes A_2 \\ 0 & 0 & I_d \otimes B \end{pmatrix},$$

where the symbol $\otimes$ denotes the Kronecker product, I_d is the $d \times d$ identity matrix and $e_d = (1, 1, \ldots, 1)^T \in \mathbb{R}^d$. It follows from (3.8), (3.9) and (3.11) that $\{X_n\}$ satisfies the system of difference equations

$$L_0 X_n = L_1 X_{n-1} + L_2 X_{n-k}, \qquad n = 0, 1, \ldots. \tag{3.12}$$

If L_0 is nonsingular then the solution of (3.12), together with the first equation of (3.7), fully determines the numerical solution of (3.1). We recall that the A-stability of the Runge–Kutta method (2.5) implies the following two conditions:

(H_7) $\sigma[A] \subset \mathbb{C}^+ \cup \mathbb{C}^0$,

(H_8) the stability function

$$R(z) = 1 + zb^T (I_s - zA)^{-1} e_s$$

satisfies

$$|R(z)| < 1, \qquad \forall\, z \in \mathbb{C}^-.$$

Since

$$\det(L_0) = \det(I_{sd} - hA \otimes A_0) = \prod_{\lambda \in \sigma[A]} \prod_{\mu \in \sigma[A_0]} (1 - h\lambda\mu)$$

it follows from (H_4) and (H_7) that L_0 is nonsingular. Hence, the characteristic polynomial of (3.12) can be written as

$$P^*(z) = \det(L_0)^{-1} \det(z^m L_0 - z^{m-1} L_1 - L_2).$$

Let us prove by contradiction that the characteristic polynomial $P^*(z)$ of system (3.12) has no zero outside the open unit disk D. Suppose that there exists a $\lambda \notin D$ such that $P^*(\lambda) = 0$. Then the matrix $I_{sd} - \lambda^{-m} I_d \otimes B$ is nonsingular, since $\rho(B) < 1$ implies that

$$\det(I_{sd} - \lambda^{-m} I_d \otimes B) = \left[\det(I_d - \lambda^{-m} B)\right]^d \neq 0.$$

Let

$$W_2 = (A_0 + \lambda^{-m} A_1)(I_d - \lambda^{-m} B)^{-1}$$

such that

$$\begin{aligned} &I_{sd} - hA \otimes A_0 - h\lambda^{-m}(I_d \otimes A_0)[I_{sd} - \lambda^{-m} I_d \otimes B]^{-1}(A \otimes I_d) \\ &= I_{sd} - hA \otimes W_2 \end{aligned}$$

Since (H_2), (H_5) and (H_6) implies that

$$\det(I_{sd} - hA \otimes W_2) = \det(I - \lambda^{-m})^{-s} \prod_{\mu \in \sigma[A]} \mu h P((\mu h)^{-1}, \lambda^{-m}) \neq 0,$$

the matrix

$$W_1 := \begin{pmatrix} I_{sd} - hA \otimes A_0 & -\lambda^{-m} I_d \otimes A_0 \\ -hA \otimes I_d & I_{sd} - \lambda^{-m} I_d \otimes B \end{pmatrix},$$

is nonsingular. Since

$$P^*(\lambda) = \lambda^{(2m+1)d-1} \det(W_1) \det[(\lambda - 1)I_d - W_3],$$

where

$$\begin{aligned} W_3 &= (hb^T \otimes I_d, 0) W_1^{-1} \begin{pmatrix} e_s \otimes A_0 \\ e_s \otimes I_d \end{pmatrix} \\ &= (hb^T \otimes I_d)(I_{sd} - hA \otimes W_2)^{-1}(e_s \otimes W_2), \end{aligned}$$

we have

$$P^*(\lambda) = \lambda^{(2m+1)d-1} \det(W_1) \prod_{\mu \in \sigma[W_2]} [\lambda - R(h\mu)].$$

Since $\lambda \notin D$ is a zero of $P^*(z)$, there exists at least one $\mu_0 \in \sigma[D]$ such that $\lambda = R(h\mu_0)$. Because of

$$P(\mu_0, \lambda^{-m}) = \det(I - \lambda^{-m})^{-1} \det(\mu_0 I - W_2),$$

(H_2) implies that either $\mu_0 \in \mathbb{C}^-$ or $\mu_0 = 0$. If $\mu_0 \in \mathbb{C}^-$ we have $|\lambda| = |R(h\mu_0)| < 1$ which contradicts our assumption that $|\lambda| \geq 1$. If $\mu_0 = 0$ then $\lambda = R(0) = 1$ and $0 \in \sigma[W_2]$. Noting that $\det(W_2) = 0$, we obtain $\det(A_0 + A_1) = 0$ which contradicts (H_3). Therefore, $P^*(z)$ has no zero outside the open unit disk D.

Hence, the solution of (3.12) obeys $\lim_{n\to\infty} X_n = 0$, which implies that $\lim_{n\to\infty} z_n = 0$. Since $0 \in \{c_i^* \mid i = 1, 2, \ldots, s^*\}$, we have

$$y_n = z_n + By_{n-m}, \qquad n = 0, 1, \ldots.$$

Therefore, $\lim_{n\to\infty} y_n = 0$ due to the fact that $\rho(B) < 1$. □

Similar test equations, mostly scalar equations and/or retarded type equations, have been used in papers such as [4, 15, 21, 22, 23, 32]. In some other papers such as [2, 6, 24, 25], the so called pantograph equation (equation with proportional time delay) is used as the test equation. A recent paper of Zennaro [31] deals with a nonlinear system of equations. In particular, we mention that the test equation

$$u'(t) = A_0 u(t) + A_1 u(t - r) + Bu'(t - r), \qquad t \geq 0 \tag{3.13}$$

is used in Hu and Mitsui [14]. As far as the stability of zero solution is concerned, there is no difference between (3.1) and (3.13), though the stability condition assumed in [14] is perhaps stronger (certainly not weaker) than the one assumed in this paper. The numerical method being tested in [14] is the standard adaptation of the continuous Runge–Kutta method that yields

$$u_{n+1} = u_n + \sum_{i=1}^{s} b_i K_{n,i}, \tag{3.14}$$

where

$$\begin{aligned} K_{n,i} = hA_0 \left(u_n + \sum_{j=1}^{s} a_{ij} K_{n,j} \right) + hA_1 \left(u_{n-m} + \sum_{j=1}^{s} b_j(c_i) K_{n-m,j} \right) \\ + B \sum_{j=1}^{s} b_j'(c_i) K_{n-m,j}, \qquad i = 1, 2, \ldots, s. \end{aligned} \tag{3.15}$$

An analogue of Theorem 3 was proved in [14] under the assumptions that (3.6) and

$$(b_j'(c_i))_{s \times s} = I \tag{3.16}$$

are satisfied. It should be noted that (3.16) is a severe restriction. In fact, a Runge–Kutta method whose continuous extension satisfies both (3.6) and (3.16) belongs to the class of projection methods [30], which is a small subset of all Runge–Kutta methods. The condition (3.16) was also assumed in [4] and it seemed to be essential to the stability analysis in [4] as well as in [14]. It should be noted that (3.16) is not assumed in this paper.

Remark. A referee of this paper found a very interesting connection between the proof of a special case of Theorem 3 of this paper and the proof of Theorem 4.2 of Hu and Mitsui [14]. Suppose that (3.6) and (3.16) hold (as in [14]). Then (3.15) can be written as

$$\begin{aligned} K_{n,i} = hA_0 \left(u_n + \sum_{j=1}^{s} a_{ij} K_{n,j} \right) + hA_1 \left(u_{n-m} + \sum_{j=1}^{s} a_{ij} K_{n-m,j} \right) \\ + BK_{n-m,i}, \qquad i = 1, 2, \ldots, s. \end{aligned} \tag{3.17}$$

Suppose that $1 \in \{c_i \,|\, i = 1, \ldots, s\}$ (a special case of Theorem 3). Assuming without loss of generality that $c_s = 1$, then (3.6) implies that

$$b_j = a_{sj}, \qquad j = 1, 2, \ldots, s.$$

Consequently,

$$Y_{n,s} = y_{n+1}.$$

Since $0 \in \{c_i^* \,|\, i = 1, \ldots, s^*\}$, we have

$$y_n = z_n + By_{n-m}.$$

The preceding three equations, together with (3.9) and (3.10), imply that

$$y_{n+1} = y_n + hA_0 \sum_{j=1}^{s} b_j Y_{n,j} + hA_1 \sum_{j=1}^{s} b_j Y_{n-m,j} + B(y_{n+1-m} - y_{n-m}) \tag{3.18}$$

and

$$Y_{n,i} = y_n + hA_0 \sum_{j=1}^{s} a_{ij} Y_{n,j} + hA_1 \sum_{j=1}^{s} a_{ij} Y_{n-m,j} + B(y_{n+1-m} - y_{n-m}), \qquad i = 1, 2, \ldots, s. \tag{3.19}$$

If

$$\begin{cases} y_{n+1} = u_{n+1} \\ Y_{n,i} = u_n + \sum_{j=1}^{s} a_{ij} K_{n,j}, \qquad i = 1, 2, \ldots, s \end{cases} \tag{3.20}$$

holds for all $n = m, m+1, \ldots, 2m-1$, then it can be shown by induction that (3.20) holds for all $n \geq 2m$. $\square$

It should be noted that (3.20) may not be true for $n = m, m+1, \ldots, 2m-1$, even if the initial functions for (3.1) and (3.13) are the same, i.e., $y(t) = u(t)$ for $t \in [-r, 0]$. For instance, the derivative of the initial function is needed in the case of the standard adaptation of Runge–Kutta methods, but it is not needed for the methods presented in this paper.

We are unable to prove Theorem 3 when the mesh is unconstrained. In the case of θ-methods, some relevant results can be found in M.Z. Liu and Spijker [23], in't Hout [15] and Kuang, Xiang and Tian [22].

Finally, we mention that solving (1.2,1.5) is equivalent to solving the integral-functional equation

$$y(t) - g(t, y(\phi(t))) = Y(t_0) - g(t_0, Y(\phi(t_0))) + \int_{t_0}^{t} f_0(\tau, y(\tau), y(\phi(\tau)))\,\mathrm{d}\tau, \qquad t \geq t_0$$

with the initial condition (1.5). However, equations of integral type are usually more expensive to solve numerically.

Acknowledgements. The author thanks Dr. A. Iserles (Cambridge) for his careful reading of the manuscript and Prof. M.N. Spijker (Leiden) for his comment which led to the discussion of (1.4). He is also grateful to the two referees for their invaluable comments and suggestions.

References

[1] C.T.H. Baker, C.A.H. Paul and D.R. Willé, Issues in the numerical solutions of evolutionary delay differential equations, *Advances in Comput. Math.* **3** (1995), 171–196.

[2] A. Bellen, N. Guglielmi and L. Torelli, Asymptotic stability properties of θ-methods for the pantograph equation, *Applied Numer. Math.* **24** (1997), 279–294.

[3] A. Bellen and R. Vermiglio, Some applications of continuous Runge–Kutta methods, *Applied Numer. Math.* **22** (1996), 63–80.

[4] A. Bellen, Z. Jackiewicz and M. Zennaro, Stability analysis of Runge–Kutta methods for neutral delay-differential equations, *Numer. Math.* **52** (1988), 605–619.

[5] H. Brunner, On the discretization of differential and Volterra integral equations with variable delay, *BIT*, **37** (1997), 1–12.

[6] M.D. Buhmann and A. Iserles, On the dynamics of a discretized neutral equation, *IMA J. Numer. Anal.* **12** (1992), 339–364.

[7] W.H. Enright and M. Hu, Interpolating Runge–Kutta methods for vanishing delay differential equations,*Computing* **55** (1995), 223–236.

[8] W.H. Enright, K.R. Jackson, S.P. Nørsett and P.G. Thomson, Interpolants for Runge–Kutta formulas, *ACM Trans. Math. Softw.* **12** (1986), 193–218.

[9] E. Hairer, S.P. Nørsett and G. Wanner, *Solving Ordinary Differential Equations I:* (2nd ed.), Springer-Verlag (Berlin), 1987.

[10] J.K. Hale, E.F. Infante and F-S. P. Tsen, Stability in linear delay equations, J. Math. Anal. Appl. **105** (1985), 533–555.

[11] J.K. Hale and S.M. Verduyn Lunel, *Introduction to Functional Differential Equations*, Springer-Verlag, New York, 1993.

[12] F. Hartung, T.L. Herdman and J. Turi, On existence, uniqueness and numerical approximation for neutral equations with state-dependent delays, *Applied Numer. Math.* **24** (1997), 393–409.

[13] H. Hayashi, Numerical solution of retarded and neutral delay differential equations using continuous Runge–Kutta methods. Ph.D thesis, University of Toronto, 1996.

[14] G. Hu and T. Mitsui, Stability analysis of numerical methods for systems of neutral delay-differential equations, *BIT* **35** (1995), 504–515.

[15] K.J. in't Hout, The stability of θ-methods for systems of delay differential equations, *Annals Numer. Math.* **1** (1994), 323–334.

[16] K. Ito and J. Turi, Numerical methods for a class of singular integral-differential equations based on semi-group approximation, *SIAM J. Numer. Anal.* **28** (1991), 1698–1722.

[17] Z. Jackiewicz, Adams methods for neutral functional differential equations, *Numer. Math.* **39** (1982), 221–230.

[18] Z. Jackiewicz, One-step methods of any order for neutral functional differential equations, *SIAM J. Numer. Anal.* **21** (1984), 486–511.

[19] Z. Jackiewicz and E. Lo, The numerical solution of neutral functional differential equations by Adams predictor-corrector methods, *Appl. Numer. Math.* **8** (1991), 477–491.

[20] V. Kolmanovskii and A. Myshkis, *Applied Theory of Functional Differential Equations*, Kluwer Academic Publishers, 1992.

[21] T. Koto, A stability property of A-stable natural Runge–Kutta methods for systems of delay differential equations, *BIT*, **34** (1994), 262–267.

[22] J. Kuang, J. Xiang and H. Tian, The asymptotical stability of one-parameter methods for neutral differential equations, *BIT*, **34** (1994), 400-408.

[23] M.Z. Liu and M.N. Spijker, The stability of the θ-methods in the numerical solution of delay differential equations, *IMA J. Numer. Anal.* **10** (1990), 31–48.

[24] Y. Liu, Stability analysis of θ-methods for neutral differential equations, *Numer. Math.* , **70** (1995), 473–485.

[25] Y. Liu, On θ-method for delay differential equations with infinite lag, *J. Comp. Appl. Math.* **71** (1996), 177–190.

[26] K.W. Morton and D.F. Mayers, *Numerical Solution of Partial Differential Equations*, Cambridge University Press, 1994.

[27] B. Owren and M. Zennaro, Derivation of efficient continuous explicit Runge–Kutta methods, *SIAM J. Sci. Stat. Comput.* **13** (1992), 1488–1501.

[28] L.F. Shampine, Interpolation for Runge–Kutta methods, *SIAM J. Numer. Anal.* **22** (1985), 1014–1027.

[29] M. Zennaro, Natural continuous extensions of Runge–Kutta methods, *Math. Comput.* **46** (1986), 119–133.

[30] M. Zennaro, Natural Runge–Kutta and projection methods, *Numer. Math.* **53** (1988), 423–438.

[31] M. Zennaro, Asymptotic stability analysis of Runge–Kutta methods for nonlinear systems of delay differential equations, *Numer. Math.* **77** (1997), 549–563.

[32] W. Zhu and L. Petzold, Asymptotic stability of linear delay differential algebraic equations and numerical methods, *Applied Numer, Math.* **24** (1997), 247–264.