Journal of Computational Mathematics, Vol.23, No.6, 2005, 647–656.

A CLASS OF TWO-STEP CONTINUITY RUNGE-KUTTA METHODS FOR SOLVING SINGULAR DELAY DIFFERENTIAL EQUATIONS AND ITS STABILITY ANALYSIS *

Xin Leng De-gui Liu
(*Beijing Institute of Computer Application and Simulation Technology, 100854, China; Beijing Institute of Applied Physics and Computational Mathematics, 100088, China*)

Xiao-qiu Song Li-rong Chen
(*Beijing Institute of Computer Application and Simulation Technology, 100854, China*)

Abstract

In this paper, a class of two-step continuity Runge-Kutta(TSCRK) methods for solving singular delay differential equations(DDEs) is presented. Analysis of numerical stability of this methods is given. We consider the two distinct cases: $(i)\tau \geq h$, $(ii)\tau < h$, where the delay τ and step size h of the two-step continuity Runge-Kutta methods are both constant. The absolute stability regions of some methods are plotted and numerical examples show the efficiency of the method.

Mathematics subject classification: 34k20
Key words: Analysis of numerical stability, Singular delay differential equations, Two-step continuity Runge-Kutta methods

1. Introduction

Consider the following delay differential equations (DDEs):

$$\begin{cases} y'(x) = f(x, y(x), y(x-\tau(x))) & a \leq x \leq b \\ y(x) = \varphi(x) & x_{min} \leq x \leq a \end{cases} \tag{1.1}$$

where y, f, φ are n-vector functions, $\varphi(x)$ is initial value function, $\tau(x) \geq 0$ is delay function.

Definition 1.1. *DDEs (1.1) is singular at the point x_α if the delay function satisfies $\tau(x_\alpha) = 0$. If there is no such point $x_\alpha \in [a\ b]$, then the DDEs (1.1) is non-singular.*

In the numerical solution of DDEs (1.1) by a continuous explicit Runge-Kutta method, we suppose that we have an approximation y_n to $y(x)$ at x_n and wish to compute an approximation at $x_{n+1} = x_n + h$. For $i = 1, 2, \cdots, s$, the stages $f_{ni} = f(x_{ni}, y_{ni}, \tilde{y}(x_{ni} - \tau(x_{ni})))$ are defined in terms of $x_{ni} = x_n + c_i h$ and $0 \leq c_i \leq 1$. Continuous explicit Runge-Kutta method is

$$\begin{cases} y_{ni} = y_n + h \sum\limits_{j=1}^{i-1} a_{ij} f_{ni} \\ y_{n+\sigma} = \tilde{y}(x_n + \sigma h) = y_n + h \sum\limits_{i=1}^{s} b_i(\sigma) f_{ni} \end{cases} \tag{1.2}$$

When DDE is singular or has a vanishing delay, a delay may fall in the span of the current step where there is no available approximation for the solution value at the delayed argument. This situation can also arise particularly at relaxed tolerance when the delay does not vanishing but actual optimal step size is larger than the size of the delay. In such case, $x_{ni} - \tau(x_{ni}) > x_n$ for some x_{ni}. Since no approximation for $\tilde{y}(x_{ni} - \tau(x_{ni}))$ is available, the explicit Runge-Kutta

*Received October 22, 2004

formulae (1.2) become implicit. For solving singular delay differential equations several authors have adopt the iteration scheme [1], [3]. However iteration is a computationally expensive approach. In this paper we relax the effect of delay when computing the Runge-Kutta stages in the span of the current step and construct a class of two-step continuity Runge-Kutta (TSCRK) methods. This class of methods keeps the explicit process of computing the Runge-Kutta stages and avoids iteration, then reduces the computing workload. Numerical stability analysis of the methods is given, and the regions of absolute stability for this class of methods are plotted. The numerical results show the efficiency of the TSCRK methods. The vanishing delay can be handled automatically by the methods so the users do not need to know where the delay vanishes.

2. TSCRK Methods

For the numerical solution of DDEs (1.1) we construct the TSCRK methods. These methods have the form

$$\begin{cases} y_{ni} = \alpha_i y_{n-1} + (1-\alpha_i)y_n + h\sum_{j=1}^{s}(a_{ij}f_{n-1j} + b_{ij}f_{nj}) \\ y_{n+\sigma} = Q(x_n+\sigma h) = \tilde{\theta}(\sigma)y_{n-1} + (1-\tilde{\theta}(\sigma))y_n \\ \quad + h\sum_{i=1}^{s}(v_i(\sigma)f_{n-1i} + w_i(\sigma)f_{ni}) \end{cases} \tag{2.1}$$

where $f_{ni} = f(x_{ni}, y_{ni}, Q(x_{ni} - \tau(x_{ni})))$, $Q(x_n + \sigma h)$ is an approximation to $y(x_n + \sigma h)$, $0 \le \sigma \le 1$, $Q(x_n + h) = y_{n+1}$, $Q(x_n) = y_n$, $x_{ni} = x_n + c_i h$, $b_{ij} = 0$, for $j \ge i$. Assume that $c_1 \equiv 0$. $\tilde{\theta}(\sigma)$, $v_i(\sigma)$, $w_i(\sigma)$ are polynomials in σ of degree p^*, $p^* \ge p$, p is the order of the methods. $\tilde{\theta}(0) = v_i(0) = w_i(0) = 0$, $\tilde{\theta}(1) = \theta$, $v_i(1) = v_i$, $w_i(1) = w_i$, $i = 1, \cdots, s$. We define $Q(x) = \varphi(x)$ when $x \le a$. Methods (2.1) are not self-starting and we use the continuous RK method of the same order as the TSCRK methods (2.1) to compute the required approximations, y_1, y_{01}, y_{02}, $\cdots$, y_{0s} and y_σ, $0 \le \sigma \le 1$. When delay falls in the first interval $[x_0,\ x_1]$, we use the iteration scheme constructed in [1] to compute the approximations. When DDEs is singular or has a vanishing delay, a delay may fall in the current step. We relax the effect of delay when computing the Runge-Kutta stages in the span of the current step and assume that $w_2(\sigma) \equiv w_3(\sigma) \equiv \cdots \equiv w_s(\sigma) \equiv 0$, for all $0 \le \sigma \le 1$, we have

$$\begin{cases} y_{ni} = \alpha_i y_{n-1} + (1-\alpha_i)y_n + h\sum_{j=1}^{s}(a_{ij}f_{n-1j} + b_{ij}f_{nj}) \\ y_{n+\sigma} = Q(x_n+\sigma h) = \tilde{\theta}(\sigma)y_{n-1} + (1-\tilde{\theta}(\sigma))y_n \\ \quad + h\sum_{i=1}^{s}v_i(\sigma)f_{n-1i} + hw_1(\sigma)f_{n1} \end{cases} \tag{2.2}$$

Methods (2.2) keep the explicit process when solving singular delay differential equations and vanishing delay differential equations. Introducing the vectors

$z_{n,\sigma} = z(x_n, \sigma h) = (y(x_n + \sigma h), y(x_n), B(\Phi_1, y(x_n)), \cdots, B(\Phi_s, y(x_n)))^T$

$z_n = z(x_n, h) = (y(x_n + h), y(x_n), B(\Phi_1, y(x_n)), \cdots, B(\Phi_s, y(x_n)))^T$

$u_{n,\sigma} = (y_{n+\sigma}, y_n, y_{n,1}, \cdots, y_{n,s})^T$

$u_n = (y_{n+1}, y_n, y_{n,1}, \cdots, y_{n,s})^T$

Then the methods (2.1) can be written in the form

$$\begin{cases} u_{0,\sigma} = \Psi(\sigma h) \\ u_{n+1,\sigma} = Ru_n + h\bar{\Phi}(x_{n+1}, u_n, \sigma; Q(x)) \end{cases} \tag{2.3}$$

Here $\Psi(\sigma h)$ specifies the "starting procedure", the matrix R is given by

$R = \begin{pmatrix} 1-\tilde{\theta}(\sigma) & \tilde{\theta}(\sigma) & \mathbf{0} \\ 1 & 0 & \mathbf{0} \\ \mathbf{1}-\alpha & \alpha & \mathbf{0} \end{pmatrix}$, $\mathbf{1} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$, $\alpha = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_s \end{pmatrix}$, and the increment function