Fakulti Teknologi Maklumat dan Sains Kuantitatif ISSN:1823-0822 Jilid 7, Bil. 1, 2005

UNIVERSITI TEKNOLOGI MARA

JURNAL TEKNOLOGI MAKLUMAT DAN SAINS KUANTITATIF

Kandungan **Muka Surat**

Universiti Teknologi MARA

LEMBAGA PENYUNTING

Dasar Penerbitan:
Jurnal Teknologi Maklumat dan Sains Kuantitatif diterbitkan oleh Fakulti Teknologi Maklumat dan Sains Kuantitatif, Universiti Teknologi MARA. Sumbangan penulisan adalah berkaitan dengan teori, amali, metodologi serta falsafah aspek-aspek kefenomenan dan epistemologi dalam sains matematik dan pengkomputeran. Tujuan utama jurnal ini adalah untuk mengenengahkan bahan/karya yang menunjuk dan mempersembahkan keselarasan serta keharmonian dalam teknologi maklumat dan sains kuantitatif. kandungan makalah yang dimuatkan dalam jurnal ini tidak semestinya mencerminkan pandangan dan pendirian rasmi jurnal ini.

Alamatkan semua sumbangan kepada:
Urusetia Panel Penyunting,
Jurnal Teknologi Maklumat dan Sains Kuantitatif
Fakulti Teknologi Maklumat dan Sains Kuantitatif
UiTM Shah Alam

Tel **03-5543 5329**
Faks **03-5543 5501**
e-mail mshahar@tmsk.uitm.edu.my

DARI MEJA KETUA PENYUNTING

Alhamdulillah, dapat kita terbitkan Jurnal Teknologi Maklumat dan Sains Kuantitatif Jilid 7, Bil.1, 2005. Saya rasa pencinta ilmu menanti-nanti terbitan kali ini.

Seperti biasa jurnal terbitan sesuatu tahun itu, hanya dapat dihantar untuk percetakan dua atau tiga bulan berikutnya. Kadangkala, penulis yang telah menghantar balik artikel yang telah diwasitkan itu tertunggu-tunggu juga adakah artikelnya diterbitkan kali ini. Sememangnya pihak penyunting mengamalkan prinsip giliran FIFO (first in first out), tetapi kadangkala ianya tidak boleh dilakukan. Ini kerana sesuatu bidang pengkhususan itu mempunyai dua atau tiga artikel sekaligus. Jadi pihak penyunting berkemungkinan akan melewatkan salah satu daripada artikel sebidang itu kemudian. Justeru itu, giliran FIFO masih dilakukan dalam bidang yang sama.

Dalam keluaran yang lepas, saya ada mengatakan bahawa minat penulis akan terhakis apabila maklumbalas tentang penerimaan sesuatu artikel untuk diterbitkan itu lambat. Saya hanya boleh memberi nasihat kepada penulis supaya bersabar, sebab ini begantung kepada pewasit yang menilai itu sibuk atau tidak, sanggup atau tidak dan sebagainya. Percayalah, kesabaran itu akan menjadi kita penulis yang berdisiplin.

Akhir kata, saya harap semua penulis-penulis semasa dan yang akan datang tetap gigih untuk menulis supaya karya kita dapat dimanfaatkan oleh para ilmuwan yang lain dalam bidang kita iaitu Teknologi Maklumat dan Sains Kuantitatif

Terima kasih.

Ketua Penyunting.
Prof. Dr. Mohd Sahar Sawiran

Jurnal Tek. Maklumat & Sains Kuantitatif ISSN 1823-0822 Jilid 7,Bil 1, 2005

Different Types Of Interpolations For Solving Delay Differential Equations Using Explicit Runge-Kutta Method

Fudziah Ismail, Aung San Lwin and Mohamed Suleiman
Department Of Mathematics Universiti Putra Malaysia
43400 Serdang, Selangor
fudziah@fsas.upm.edu.my

Abstract

Delay differential equations (DDEs) with single and multiple delays are solved using embedded explicit Runge-Kutta method. The delay terms are approximated by using divided difference interpolation, Hermite interpolation and continuous extensions formula of the Runge-Kutta method itself. Numerical results based on the different types of interpolation are tabulated and compared.

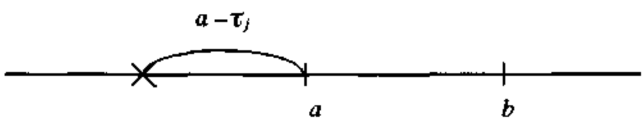
Keywords: Delay Differential Equations; Interpolations ; Runge-Kutta Method.

1.0 Introduction

Recently a lot of research has been focused on delay differential equations. This is because DDEs provide a realistic model of many phenomena arising in applied mathematics such as the spread of infectious diseases and reaction to X-ray treatment. More detailed study of the applications of DDE can be found in Driver (1977), Hairer et al ((1993), Kuang (1991) and Macdonald (1989). A general form of a first order delay differential equations is

$$y'(t)=f(t,y(t),y(t-\tau(t,y(t)))) \quad \text{for} \quad t\in[a,b] \tag{1.1a}$$

where $\tau(t,y(t))\leq t$. $t-\tau(t,y(t))$ is "the lag" and $\tau(t,y(t))$ is "the Delay". Delay differential equations contain derivatives, which depend on the solution at previous time. For example, at t = a we must have the solution at $a-\tau_j$ as shown in the diagram below. If T is the longest delay, the equations generally require us to provide the solution y(t) for $T< t < a$. For DDE we must provide not just the value of the solution at the initial point, but also the history of the solution before the initial point.



The initial conditions generally involve prescribed values of y(t) on an interval. The precise nature of these initial conditions depends on the nature of the delay and the range of values of the argument t . If the solution of the problem is defined by an initial value at a single point, the problem is called an initial value delay differential equation (IVDDE), otherwise it is call an initial function delay differential equation (IFDDE). In general, if we seek the solution of (1.1a) for $a \le t \le b$ an initial function of the following form is required

$$y(t) = \phi(t) \qquad \text{for} \qquad t \in [a^*, a] \tag{1.1b}$$

where $a^* = \min\{\tau(t, y(t));\ \text{for}\ \ t \in [a^*, a]\}$, the delay term.

A general DDE with multiple delays can also be written as follows:

$$y'(t) = f(t, y(t), y(t - \tau_1(t, y(t)), \ldots\ldots, y(t - \tau_q(t, y(t)))), \tag{1.2}$$

where $\tau_i(t, y(t)) \ge 0$ for $i = 1, 2, \ldots\ldots, q$.

A system of n DDE with multiple delays has the form

$$y_i'(t) = f_i(t, \ldots, y_j(t - \tau_{jk}(t, y_1(t)), \ldots\ldots, y_n(t))), \ldots) \tag{1.3}$$

where $i, j = 1, \ldots, n$; $k = 0, 1, \ldots, q$, and $\tau_{jk}(t, y_1(t)), \ldots\ldots, y_n(t)) \ge 0$.
In this paper the method used is the Runge-Kutta DOPRI (4,5) pair developed by Dormand and Prince (1980), where the lower order method is of order four and has six stages and the higher order method is of order five and has seven stages. The method is represented in Table (1.1).

Table (1.1): Coefficients of Explicit Runge-Kutta DOPRI (4,5)

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

2 Numerical method for solving DDEs

Most numerical methods for solving ordinary differential equation (ODE)

$$y'(t)=f(t,y), \qquad t \in [a, b] \tag{2.1}$$
$$y(a) = y_0,$$

can be adapted to solve DDE (1.1). When an s-stage explicit Runge-Kutta method is used to solve (2.1) at the point t_{n+1}, the following equations will be obtained.

$$k_i=f(t_n+c_ih, y_n+h\sum_{j=1}^{i-1} a_{ij}k_j), \qquad i=1,2,\ldots,s$$
$$y_{n+1} = y_n + h \sum_{i=1}^{s} b_i k_i \tag{2.2}$$

When the method is adapted to DDE (1.1) we have

$$k_i=f(t_n+c_ih, y_n+h\sum_{j=1}^{i-1} a_{ij}k_j, y(t_n+c_ih-\tau)), \qquad i=1,2,\ldots,s$$
$$y_{n+1} = y_n + h \sum_{i=1}^{s} b_i k_i \tag{2.3}$$

Where $y(t_n+c_ih-\tau)$ is approximated using previously computed values of y(t). there are a number of techniques for obtaining the approximations of $y(t-\tau)$. For example In't Hout (1992) resorted to the techniques of multistage continuous extensions to approximate the delay term and Al-Mutib (1977) used Hermite Interpolation for the purpose. In this paper three types of interpolation are used for the approximation of the delay terms. They are divided difference interpolation, Hermite interpolation and continuous extensions Runge-Kutta method itself. The interpolation order and hence the number of points used have to be adapted to the order of the method. Since DOPRI method is of order five hence there are six values of y's (six points) used for the divided difference interpolation and the points are chosen such that $t_n+c_ih-\tau$ falls in the middle of the six points. For Hermite interpolation we used both the values of y and y', hence for the interpolation to be of the same order as the Runge-Kutta method itself, values of y and y' at three points are used for the interpolation. Interpolation using continuous extension Runge-Kutta method will be described in the next section.

3 Continuous Extensions Runge-Kutta (CERK) Formula

As we know, Runge-Kutta method produces approximation only at discrete points, CERK formula produces continuous approximation to the solution of an ODE, the formula which was developed by Dormand and Prince (1980) is used here for the purpose of approximating the delay term.

The general form of a continuous extensions Runge-Kutta formula is

$$y_n(t_n+\_h_n)=y_{n-1}(t_n)+h_n\sum_{i=1}^{s} b_i^*(\theta)k_i \tag{3.1}$$

where $0\le\theta\le 1$

Define a CERK formula for DDE by

$$y_n(t_n+\_h_n)=y_{n-1}(t_n)+h_n\sum_{i=1}^{s} b_i^*(\theta)k_i$$

$$k_i = f(t_n + c_i h_n, Y_i, y(t_n+c_ih-\tau\ (t_n+c_ih, y_n))) \tag{3.3a}$$

$$y_n=y_{n-1}+h_n\sum_{j=1}^{i-1} a_{ij}k_i \tag{3.3b}$$

Fourth order continuous extension for Dormand and Prince method is given by the following :

$b_1^*(\theta) = \theta\,(1+\theta\,(-13371/480+\theta\,(1039/360+\theta\,(-1163/1152))))$

$b_2^*(\theta) = 0$

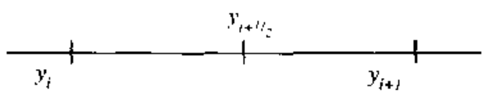
$b_3^*(\theta) = 100\,\theta^2\,(1054/9275+\theta\,(-4682/27825+\theta\,(379/5565)))\,/\,3$

$b_4^*(\theta) = 5\,\theta^2\,(27/40+\theta\,(-9/5+\theta\,(83/96)))\,/\,2$

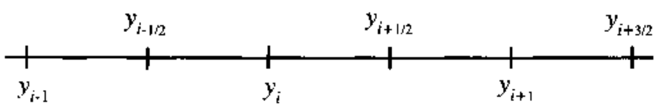
$b_5^*(\theta) = 18225\,\theta^2\,(-3/250+\theta\,(-37/600)))\,/\,848$

$b_6^*(\theta) = 22\,\theta^2\,(-3/10+\theta\,(29/30+\theta\,(-17/24)))\,/\,7$

The method together with the coefficients of Table (1.1) constitutes the continuous extensions Runge-Kutta formula. This fourth order CERK method is used to solve DDE with single and multiple delays. The continuous extensions Runge-Kutta formula with θ=0.5 is used to get the value of y in the middle of the mesh points.



(That is the value of $y_{i+1/2}$ which lies in between y_i and y_{i+1}) .Supposedly the delay term lies in between t_i and t_{i+1} ,so we use the values of $y_{i-1}, y_{i-1/2}, y_i, y_{i+1/2}, y_{i+1}$ and $y_{i+3/2}$ (as shown below).



for the interpolation. CERK formula is used only to find the value of y at mid-point of two mesh points, later this value of y together with the values at mesh points were used for the divided difference interpolation so that the approximation of the delay term can be obtained. Here the DOPRI method is of order five and the CERK method is of order four.

4 Test Problems

The following problems (1) and (2) are given in Al-Mutib (1977) and problem (3), which is a system of delay differential equation having two delays, is from C.A.H. Paul (1997). They are solved using explicit Runge-Kutta DOPRI method in Table (1.1) and the delay terms are approximated using divided difference interpolation, Hermite interpolation and continuous extensions Runge-Kutta formula.

Problem 1

$$y_{-}(t) = 1 - y(\exp(1 - 1/t)) \qquad t \geq 3$$

$$y(t) = \ln(t) \qquad 0 < t \leq 3.$$

Analytical solution: $y(t) = \ln(t)$

Problem 2

$$y_{-1}(t) = y_2(t) \qquad 3 \leq t \leq 10$$

$$y_{-2}(t) = -y_2(\exp(1 - y_2(t))(y_2(t))^2 \exp(t - y_2(t)) \qquad 3 \leq t \leq 10$$

$$y_1(t) = \ln(t) \text{ and } y_2(t) = 1/t \qquad 0 \leq t \leq 3$$

Analytical solution:

$$\left.\begin{array}{l} y_1(t) = \ln(t) \\ y_2(t) = \dfrac{1}{t} \end{array}\right\} \quad 3 \leq t \leq 10$$

Problem 3

$$\begin{array}{ll}
y'_1(t) = y_5(t-1) + y_3(t-1) & t \geq 0, \\
y'_2(t) = y_1(t-1) + y_2(t-1/2) & t \geq 0, \\
y'_3(t) = y_3(t-1) + y_1(t-1/2) & t \geq 0, \\
y'_4(t) = y_5(t-1)\, y_4(t-1) & t \geq 0, \\
y'_5(t) = y_1(t-1) & t \geq 0, \\
y_1(t) = \exp(t+1) & t \leq 0, \\
y_2(t) = \exp(t+1/2) & t \leq 0, \\
y_3(t) = \sin(t+1) & t \leq 0, \\
y_4(t) = \exp(t+1) & t \leq 0, \\
y_5(t) = \exp(t+1) & t \leq 0.
\end{array}$$

Analytical solution:

$$y_1(t) = e^t - \cos(t) + e \qquad 0 \leq t \leq 1,$$

$$y_2(t) = 2e^t + \exp(1/2) - 2 \qquad 0 \leq t \leq 1/2,$$

$= e^t + 2\exp(t-1/2) + t\exp(1/2) - 2t + (3/2)\exp(1/2) - 3 \qquad 1/2 \le t \le 1,$

$y_3(t) = \exp(t+1/2) - \cos(t) + 1 - \exp(1/2) + \sin(1) \qquad 0 \le t \le 1/2,$

$= -\cos(t) + \exp(t-1/2) - \sin(t-1/2) + (t+1/2)e - \exp(1/2) + \sin(1) \qquad 1/2 \le t \le 1,$

$y_4(t) = (1/2)\exp(2t) - 1/2 + e \qquad 0 \le t \le 1,$

$y_5(t) = e^t + e - 1 \qquad 0 \le t \le 1,$

5 Numerical Results

For comparison, the related methods and abbreviations in tables (5.1-5.3) are

DDI : divided difference interpolation;
HI : Hermite interpolation;
CERK : continuous extensions Runge-Kutta formula;
Tol : requested error tolerance;
NF : total number of function evaluations;
NS : total number of steps;
FS : total number of fail steps;
ERR : maximum relative error over mesh points: $\max_i \left| y(t_i) - y_i \right|$

Table 5.1: Numerical results for Problem 1

Tol		NF	NS	FS	ERR
10^{-2}	DDI	28	4	0	3.20462775-E4
	HI	28	4	0	3.20462775-E4
	CERK	56	8	0	4.16753331-E5
10^{-4}	DDI	82	10	2	6.84217994-E4
	HI	61	7	2	6.84217994-E4
	CERK	91	13	0	1.44585205-E5
10^{-6}	DDI	118	16	1	1.02217893-E4
	HI	117	15	2	1.02217893-E4
	CERK	147	21	0	4.39779309-E7
10^{-8}	DDI	216	30	1	3.52151722-E7
	HI	202	28	1	3.52151722-E7
	CERK	259	37	0	3.24283761-E9
10^{-10}	DDI	447	63	1	2.30590421-E10
	HI	426	60	1	2.30590421-E10
	CERK	497	71	0	4.09876087-E11

Table 5.2: Numerical results for Problem 2

Tol		NF	NS	FS	ERR
10^{-2}	DDI	77	11	0	3.36141013-E4
	HI	77	11	0	4.44800007-E4
	CERK	77	11	0	4.44800007-E4
10^{-4}	DDI	147	21	0	1.52490459-E5
	HI	147	21	0	1.39427174-E5
	CERK	175	25	0	1.57390050-E5
10^{-6}	DDI	273	39	0	3.28879844-E7
	HI	252	36	0	2.85984712-E7
	CERK	273	39	0	3.17894034-E7
10^{-8}	DDI	476	68	0	4.85464643-E11
	HI	462	66	0	4.54779151-E11
	CERK	476	68	0	3.90141531-E9
10^{-10}	DDI	924	132	0	4.85464643-E11
	HI	924	132	0	4.54779151-E11
	CERK	931	133	0	6.02358925-E11

Table 5.3: Numerical results for Problem 3

Tol		NF	NS	FS	ERR
10^{-2}	DDI	28	4	0	9.18793396-E5
	HI	28	4	0	1.83758679-E4
	CERK	28	4	0	1.57051313-E5
10^{-4}	DDI	48	6	1	1.59526209-E4
	HI	48	6	1	3.19052418-E4
	CERK	48	6	1	1.59526209-E4
10^{-6}	DDI	83	11	1	1.71687664-E6
	HI	83	11	1	3.07351484-E6
	CERK	82	10	2	2.27373130-E5
10^{-8}	DDI	137	17	3	2.61314527-E8
	HI	151	19	3	1.50962562-E6
	CERK	137	17	3	1.54266568-E5
10^{-10}	DDI	243	33	2	2.14592913-E7
	HI	264	36	2	3.96070704-E8
	CERK	243	33	2	2.19277121-E7

6 Conclusion

Looking at the numerical results Hermite interpolation produced slightly better results based on the function evaluation, number of steps and fail steps except for problem (3) where divided difference interpolation and CERK method performed slightly better than Hermite interpolation. This is expected because the interpolation points in Hermite interpolation are nearer to the point to be interpolated compared to the other two interpolations. Furthermore CERK formula is of order four; so, for the interpolation, we have the intermediate value, which is of order four and the mesh-point value, which is of order five. This is the reason why the results are not as good as the results using divided difference and Hermite interpolations. It is also observed that the errors for problem (3) are slightly larger for most of the tolerances compared to problems (1) and (2). This is expected since the problem has more delay terms. For example the third equation in problem (3) is $y'_3(t) = y_3(t-1) + y_1(t-1/2)$ there are two delay terms in the equation which require interpolations for their approximations resulting in the increase of the total error. Generally, CERK formula is often used for the purpose of obtaining the value of y at off mesh points, but in this paper it is used for approximating the delay term. The results demonstrated that CERK formula together with divided difference interpolation can also be used as an alternative method to approximate the delay term when solving DDEs.

References

Al-Mutib A. N. 1977. *Numerical Methods for Solving Delay Differential Equations*. Ph.D. dissertation, University of Manchester.

Dormand J.R and Prince. P. J.(1980) *A family of embedded Runge-Kutta formulae*. J.Comp. Appl. Math.Vol.6, p.19-26.

Driver R.D.(1977), *Ordinary and Delay Differential Equations*. Berlin:Springer-Verlag.

E. Hairer.E , Norsett. S.P & Wanner. G.(1993) *Solving Ordinary Differential Equations* I, *Nonstiff Problems*. Berlin: Springer-Verlag.

In't Hout. K.J. (1992). *A new interpolation procedure for adapting Runge-Kutta method to Delay Differential Equations*. BIT, 32: 634-649, 1992.

Kuang. Y. (1991). *Delay Differential Equations with Applications in Population Dynamics*. Boston: Academic Press.

Macdonald. N. (1989). *Biological Delay Systems:Linear Stability Theory.* Cambridge:Cambridge University Press.

Paul. C. A. H. (1997). *An Explicit Runge-Kutta Code for Solving Delay Differential Equations and Parameter Estimation Problems*. (Numerical Analysis Report No.283) Manchester M9PL: Manchester Centre for Computational Mathematics, Department of Mathematics.