

# Research on Error Estimation and Continuous Runge-Kutta Algorithm for Real-time Simulation

Huang Zhen-quan<sup>a</sup>, Tong Er-ke<sup>b</sup>, Huang Dong-lin, Yang Zhi-qiang

Air Force Paratroops Academy, Guilin, China, 541003

<sup>a</sup>hzq922@163.com, <sup>b</sup>oakey@sina.com,

**Keywords:** real-time simulation, step-change, error estimation, continuous RK algorithm.

**Abstract.** In order to evaluate the error of the Runge-Kutta (RK) algorithm used in real-time simulation and apply it to real-time simulation problems with dense output, based on the ODE45 with maximum stability region and least truncation error, this paper adopts a variable step-size strategies and derives a relative ODE35 RK algorithm(a combination of a third- and fifth-order method) to solve error estimation problem. Further, it is extended as the continuous RK algorithm. Thus it can not only be used for real-time simulation, but also be applied to non-real-time variable step-size simulation.

## 1 Introduction

The single-step explicit Runge-Kutta(RK) algorithm is one of the extensively adopted algorithms in various engineering problems. The ODE45 RK algorithm suggested in <sup>[1]</sup>, with the minimum truncation error coefficients when stability region is maximum, is the best numerical integration algorithm both in speed and accuracy. Usually, to estimate the error, a RK algorithm with lower order should be found and the step-size be varied, so it's necessary to determine the corresponding ODE35 RK algorithm.

In addition, some real-time simulations would observe dense output, namely the distance between the output time points is very small. To obtain the solution on these time points, the usual way is shrinking the step-size. However, it disturbs the control of size-step, increases calculation time and introduces larger rounding error. In order to overcome these shortcomings, it's necessary to deduce the continuous RK algorithm which can give solutions of all interval time points .

## 2 ODE35 RK algorithm

Consider a differential equation

$$\dot{y}(t) = f(t, y(t)) \quad (1)$$

The ODE45 RK algorithm with the minimum truncation error coefficients when stability region is maximum given in <sup>[2]</sup> is:

$$\begin{cases} y_{n+1} = y_n + h \sum_{i=1}^5 c_i k_i \\ k_i = f[t_n + a_i h, y_n + h \sum_{j=1}^{i-1} b_{ij} k_j, u(t_n + a_i h)] \\ a_i = \frac{i-1}{5}, \quad i = 1, 2, 3, 4, 5 \end{cases} \quad (2)$$

Where  $c_1=-0.389584$  ;  $c_2=2.016669$  ;  $c_3=-2.295837$  ;  $c_4=1.6$  ;  $c_5=0.068749$  ;  $b_{21}=0.2$  ;  $b_{31}=0.116609$  ;  $b_{32}=0.283391$  ;  $b_{41}=-0.106439$  ;  $b_{42}=0.469396$  ;  $b_{43}=0.2370424$  ;  $b_{51}=-0.118888$  ;  $b_{52}=7.076287$  ;  $b_{53}=-11.023254$  ;  $b_{54}=4.865854$  .

The accuracy of (2) can reach fourth-order. To get the local truncation error estimation of  $y_{n+1}$  to realize variable step-size calculation, and to make sure real-time simulation effective, it's necessary to determine the corresponding ODE35 RK algorithm.

The coefficients in ODE35 should satisfy the following condition [2]:

$$\begin{cases} c_1 + c_2 + c_3 + c_4 + c_5 = 1 \\ c_2 + 2c_3 + 3c_4 + 4c_5 = \frac{5}{2} \\ c_2 + 4c_3 + 9c_4 + 16c_5 = \frac{25}{3} \\ b_{32}c_3 + (b_{42} + 2b_{43})c_4 + (b_{52} + 2b_{53} + 3b_{54})c_5 = \frac{5}{6} \end{cases} \quad (3)$$

In literature[3], Lambert gives the stability function of s-level p-order algorithm:

$$\varphi_{(z)} = 1 + z + \frac{1}{2!}z^2 + \dots + \frac{1}{p!}z^p + \sum_{i=p+1}^s \omega_i z^i \quad (4)$$

Then the stability equation of ODE35 is:

$$\hat{\Gamma}(\lambda h) = 1 + \lambda h + \frac{1}{2!}(\lambda h)^2 + \frac{1}{3!}(\lambda h)^3 + [c_4 a_2 b_{32} b_{43} + c_5 (a_2 b_{32} b_{53} + a_2 b_{42} b_{54} + a_2 b_{43} b_{54})](\lambda h)^4 + c_5 b_{21} b_{32} b_{43} b_{54} (\lambda h)^5 \quad (5)$$

On the other hand, the local truncation error of ODE35 is [4]

$$E_n = h^4 \left( \frac{\zeta_1 f_y D^2 f}{2} + \zeta_2 Df Df_y \right) \quad (6)$$

where

$$\begin{cases} \zeta_1 = c_3 a_2^2 b_{32} + c_4 (a_2^2 b_{42} + a_3^2 b_{43}) + c_5 (a_2^2 b_{52} + a_3^2 b_{53} + a_4^2 b_{54}) - \frac{1}{12} \\ \zeta_2 = c_3 a_3 a_2 b_{32} + c_4 (a_2 b_{42} + a_3 b_{43}) a_4 + c_5 (a_2 b_{52} + a_3 b_{53} + a_4 b_{54}) a_5 - \frac{1}{8} \end{cases} \quad (7)$$

$$A = |\zeta_1| + |\zeta_2| \quad (8)$$

The optimization calculation of the stability region and truncation error then can be carried out by (5) – (8). Considering the coefficients of RK, the stability region is 13.548, and the truncation error is 0.03468779,  $c_5=0.05329989$ .

Substituting the above  $c_5$  into (3) we receive the ODE35 RK algorithm (RTRK3 for short):

$$\hat{y}_{n+1} = y_n + h(0.863367k_2 - 1.173433k_3 + 1.256767k_4 + 0.053299k_5) \quad (9)$$

In literature [5],  $y_{n+1}$  has a smaller truncation error coefficient, and the fourth-order variable step-size RKL4 (3) algorithm is successfully applied to simulation of self-renovation controlling system on fighters. However, the fourth-order variable size-step RK algorithm formed by (2) and (9) has larger stability region and smaller truncation error, making it more applicable. To make it even more extensively used, it's necessary to expand it to continuous RK algorithm couplet.

### 3 Deduction of continuous RK algorithm

As the coefficients  $a_i$  and  $b_{ij}$  in continuous RK algorithm have no relations with  $\theta$ , thus in  $(t_k, t_{k+1})k_i$  need only be calculated once, as to an arbitrary point  $t_{k+\theta}$  in the region, we can get the corresponding solution by calculating the suitable coefficients, and the calculation added is very small. Introducing new step-size  $\bar{h} = \theta h$ , according to (2) we have

$$\begin{cases} y_{n+\theta} = y_n + \bar{h} \sum_{i=1}^s \bar{c}_i \bar{k}_i \\ \bar{k}_i = f(t_n + \bar{a}_i \bar{h}, y_n + \bar{h} \sum_{j=1}^{i-1} \bar{b}_{ij} \bar{k}_j) \\ i = 1, 2, \dots, s \end{cases} \quad (10)$$

Obviously, to an arbitrary point in integration region with step-size h region, inequality  $0 < \theta < 1$  holds. To fully utilize (2) and not to increase calculation times off, we hope

$$\bar{k}_i = k_i, \quad i = 1, 2, \dots, s \quad (11)$$

Comparing the corresponding coefficients in (2) and (10), the sufficient and necessary condition for (11) is

$$\bar{a}_i = a_i/\theta, \quad \bar{b}_{ij} = b_{ij}/\theta, \quad i = 1, 2, \dots, s; \quad j = 1, 2, \dots, i-1 \quad (12)$$

Thus, once  $\theta$  is fixed,  $\bar{a}_i$  and  $\bar{b}_{ij}$  are determined. We can only choose the combination coefficients  $\bar{c}_i (i = 1, 2, \dots, s)$  to make (10) have the maximum accuracy. For convenience let  $c_i$  be  $c_i(\theta)$ .

Consider the fifth level continuous RK algorithm

$$y_{n+\theta} = y_n + \theta h \sum_{i=1}^5 c_i(\theta) k_i \quad (13)$$

To match the accuracy of calculation nodes and that of non calculation nodes, it's required the accuracy of (13) be third-order, and  $y_{n+\theta} \rightarrow y_{n+1}$  when  $\theta \rightarrow 1$  (fourth-level accuracy). So the equations satisfied by  $y_{n+\theta}$  are<sup>[6]</sup>

$$\begin{cases} c_1(\theta) + c_2(\theta) + c_3(\theta) + c_4(\theta) + c_5(\theta) = 1 \\ a_2 c_2(\theta) + a_3 c_3(\theta) + a_4 c_4(\theta) + a_5 c_5(\theta) = \frac{\theta}{2} \\ a_2^2 c_2(\theta) + a_3^2 c_3(\theta) + a_4^2 c_4(\theta) + a_5^2 c_5(\theta) = \frac{\theta^2}{3} \\ a_2 b_{32} c_3(\theta) + (a_2 b_{42} + a_3 b_{43}) c_4(\theta) + (a_2 b_{52} + a_3 b_{53} + a_4 b_{54}) c_5(\theta) = \frac{\theta^2}{6} \end{cases} \quad (14)$$

Substituting the corresponding coefficients of  $k_i$  we have

$$\begin{cases} c_1(\theta) + c_2(\theta) + c_3(\theta) + c_4(\theta) + c_5(\theta) = 1 \\ c_2(\theta) + 2c_3(\theta) + 3c_4(\theta) + 4c_5(\theta) = \frac{5}{2}\theta \\ c_2(\theta) + 4c_3(\theta) + 9c_4(\theta) + 16c_5(\theta) = \frac{25}{3}\theta^2 \\ 0.0566782223c_3(\theta) + 0.1886963256c_4(\theta) - 0.07453133c_5(\theta) = \frac{\theta^2}{6} \end{cases} \quad (15)$$

There are five unknowns and four equations in (15). Choosing  $c_5 = -0.9312506677\theta + \theta^2$  as free parameter we then get

$$\begin{cases} c_1 = 1 + 15.9366431\theta - 17.3262271025\theta^2 \\ c_2 = -53.12867863682\theta + 55.1453479743\theta^2 \\ c_3 = 55.0161773\theta - 57.31201464\theta^2 \\ c_4 = -16.8928910983\theta + 18.4928937692\theta^2 \\ c_5 = -0.9312506677\theta + \theta^2 \end{cases} \quad (16)$$

Obviously, when  $\theta = 1$

$$c_i(1) = c_i, \quad i = 1, 2, \dots, 5 \quad (17)$$

By now we get the fourth-order continuous RK algorithm couplet consisting of (2), (13) and (16), with minimum truncation error when the stability region is maximum.

#### 4 Analysis on calculation examples

As to continuous algorithm, we mainly concern with whether the accuracy of calculation nodes matches with that of non calculation nodes. Consider the dynamic simulation of a marine propulsion model

$$\dot{y}(t) = -10y^2(t) + 1 + \sin(2\pi t), \quad y(0) = 0$$

Here we compare the mean value of absolute error (A.M.S.) and the maximum absolute error (M.ABS.) from the view of time domain behavior.

$$A.M.S. = \left( \sum_{i=1}^N |y_{i+\theta} - y(t_i + \theta h)| \right) / N \quad M.ABS. = \max_{1 \leq i \leq N} |y_{i+\theta} - y(t_i + \theta h)|$$

Where the accurate solution  $y(t)$  is got by fifth-order RKF algorithm<sup>[11]</sup> with small step-size and calculated by double precision. The simulation result is calculated by single precision,  $N$  is the total number of steps.

Table 1 contains the result of dynamic simulation with  $h=0.05s, 0.1s, 5s$ . RK4 refers to the classical RK4 algorithm.

Table 1. Results of  $h=0.05s, 0.1s$

Value of $\theta$	Absolute error	$h=0.05s$	$h=0.1s$
$\theta=0.2$	A.M.S.	4.2492281e-06	7.4290578e-05
	M.ABS.	2.2784933e-05	3.4501728e-04
$\theta=0.4$	A.M.S.	1.5611003e-05	2.7106502e-04
	M.ABS.	6.5212591e-05	9.8638636e-04
$\theta=0.6$	A.M.S.	2.4390665e-05	4.2596347e-04
	M.ABS.	9.3083254e-05	1.4065046e-03
$\theta=0.8$	A.M.S.	2.0996711e-05	3.6732462e-04
	M.ABS.	7.8369209e-05	1.1773955e-03
$\theta=1$ (node)	A.M.S.	3.5106874e-06	7.2971982e-05
	M.ABS.	1.0943352e-05	2.3392433e-04
RK4 ( $\theta=1$ )	A.M.S.	8.2710513e-06	1.5568267e-04
	M.ABS.	2.5993525e-05	5.0563210e-04

From Table 1 we can see that there are no obvious differences between the accuracy of solutions on nodes and that on non nodes when using (13). Their accuracies are higher than that obtained by RK4 algorithm. It indicates that the simulation result of the non nodes is valuable and the algorithm is applicable in continuous system simulation.

## 5 Conclusion

By analysis on Runge-Kutta method we find out the ODE35 RK algorithm which belongs to fifth-order variable step-size RK algorithm and can be applied to error estimation problems. The output of simulation results on non calculation nodes in variable step-size simulation is also solved. In this paper we give a fourth-order continuous RK algorithm couplet with minimum truncation error when the stability region is maximum. The simulation experiments show that this algorithm is better than other fourth-order algorithms.

## Acknowledgments

Guangxi Natural Science Foundation(No.2013GXNSFAA019021)

## References

- [1] M. Jiang and Z. Q. Huang, "Real-time RK4 Formula with Maximum Stability Region". Journal of System Simulation, vol. 18, no. 2, pp. 306-308, 2006.
- [2] D. S. Bernstein, "The treatment of inputs in real-time digital simulation" . *Simulation*, vol. 33, no. 2, pp. 65-68, 1979.
- [3] J. D. Lambert, *Computational methods in ordinary differential equations*. Wiley, New York , 1973.
- [4] M. Tanaka, "Runge-Kutta formulas with the ability of error estimation". Rep. Stat. Appl. Res. JUSE, vol. 13, no. 3, pp. 42-61, 1966.
- [5] G. Cai and S. S. Hu. "Research of Fighters self-repairing control system simulation". Acta Aeronautica ET Astronautica Sinica, vol. 20, no. 2, pp. 122-126, 1999.
- [6] Z. Q. Huang and M. Jiang, "A Pair of Fourth Order Continuous RK Formulae with a Big Stability Region". Computer Simulation, vol. 23, no. 2, pp. 70-71, 2006.