

Global error estimation of linear multistep methods through the Runge-Kutta methods

J. Farzi*

Abstract

In this paper, we study the global truncation error of the linear multistep methods (LMM) in terms of local truncation error of the corresponding Runge-Kutta schemes. The key idea is the representation of LMM with a corresponding Runge-Kutta method. For this, we need to consider the multiple step of a linear multistep method as a single step in the corresponding Runge-Kutta method. Therefore, the global error estimation of a LMM through the Runge-Kutta method will be provided. In this estimation, we do not take into account the effects of roundoff errors. The numerical illustrations show the accuracy and efficiency of the given estimation.

Keywords: Linear multistep methods; Runge-Kutta methods; Local truncation error; Global error; Error estimation.

1 Introduction

The error estimation is one of the major issues in designing numerical algorithms. In the study of the linear multistep methods (LMM)

$$\sum_{j=0}^k \alpha_j y_{n+j} = h \sum_{j=0}^k \beta_j f_{n+j}, \quad (1)$$

for solving an ordinary differential system

$$\begin{cases} y' = f(x, y), \\ y(x_0) = y_0, \end{cases} \quad (2)$$

where, $f : \mathbb{R} \times \mathbb{R}^m \rightarrow \mathbb{R}^m$, there is a challenging issue of estimation of global truncation error (GTE) or simply global error. In spite the local truncation error (LTE), the estimation of GTE is much more complicated. The

*Corresponding author

Received 25 December 2015; revised 9 March 2016; accepted 11 2016

J. Farzi

Department of Mathematics, Sahand University of Technology, P.O. Box 51335-1996, Tabriz, Iran. e-mail: farzi@sut.ac.ir

LTE estimations have been studied in some special cases, e.g., predictor-corrector (PC) and embedded Runge-Kutta methods. In the case of PC methods that predictor is an Adams-Bashforth scheme of order p and corrector is an Adams-Moulton scheme of the same order, the Milne estimation provides an estimation for LTE of the resulted PC method [9, 10]. Recently, Cao and Petzold have developed an estimation for global error with adjoint method [3]. However, many efforts have previously been reported for providing GTE bounds [6, 7]. These bounds are of no practical value and the LTE form estimation of GTE, given in this paper, is more simpler than the usual theoretical bounds. For more extensive discussion on linear multistep methods and their important subclasses, like backward difference formula (BDF) schemes see [1, 2, 8–10, 12]. The more important application of the LMMs is in the time discretization of time dependent partial differential equations. The LMMs with strong stability preserving property (SSP) have a major role in this context [4, 5].

In this paper, we represent a multiple step of a LMM as a single step of a new Runge-Kutta method. Then, we accomplish the GTE estimation of the LMM by estimation of LTE of the corresponding new Runge-Kutta method.

This paper has been organized as follow: In Section 2 a very short review of the Runge-Kutta schemes is presented. Then, in Section 3 the main idea of the paper is given for one-step methods with a rather detailed study of the LTE and stability region of the new method. In Section 4 the idea of previous section is generalized to formulate a general LMM in the form of a new Runge-Kutta method. The new RK method has a popular structure in view of Butcher array. We take into account a starting procedure, subsequently, the method order and its LTE determined, which is the GTE of the original scheme. Finally, in Section 5 we present some numerical tests including a fourth order total variation bounded (TVB) scheme to illustrate efficiency of the given theory.

2 A review of Runge-Kutta methods

In this section, we shortly review the main concepts of Runge-Kutta methods that are required in the rest of the paper. For more details we refer to [2, 9]. The s -stage Runge-Kutta method for the problem (2) is defined as follow

$$y_{n+1} = y_n + h \sum_{i=1}^s b_i k_i, \quad (3)$$

where

$$k_i = f(x_n + c_i h, y_n + h \sum_{j=1}^s a_{ij} k_j), \quad i = 1, 2, \dots, s. \quad (4)$$

Traditionally the Runge-Kutta methods is represented by the following Butcher array

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array} \tag{5}$$

where,

$$c = [c_1, c_2, \dots, c_s]^T, \quad b = [b_1, b_2, \dots, b_s]^T, \quad A = [a_{ij}]_{i,j=1}^s. \tag{6}$$

The derivation of a Runge-Kutta method of an arbitrary order is a crucial work without using the advanced concepts of elementary differentials and most related rooted trees. In fact, there is a 1-1 corresponding between elementary differential (that are defined by Fréchet derivative) and the rooted trees. Therefore, using the rooted trees one can easily define the order condition and LTE of a Runge-Kutta method. The local truncation error of the p^{th} order Runge-Kutta method (5) is given by

$$\text{LTE} = \frac{h^{p+1}}{(p+1)!} \sum_{r(t)=p+1} \alpha(t)[1 - \gamma(t)\psi(t)]F(t) + O(h^{p+2}), \tag{7}$$

where,

$$\alpha(t) = \frac{r(t)!}{\sigma(t)\gamma(t)}$$

and $r(t), \sigma(t)$ and $\gamma(t)$ are order, symmetry and density of a tree t . The function $F(t)$ is defined on the set T of all trees which corresponds between the rooted trees and elementary differentials. The elementary differentials are evaluated at the value $y(x_n)$. The $\psi(t)$ function is also defined on the set of all trees T . For example we have,

$$F(\bullet) = f,$$

$$F(\begin{array}{c} \bullet \\ | \\ \bullet \end{array}) = \{f\} = f^{(1)}(f),$$

$$F(\begin{array}{c} \bullet & \bullet \\ \diagdown & / \\ \bullet \end{array}) = \{f^2\} = f^{(2)}(f, f),$$

$$F(\begin{array}{c} \bullet \\ | \\ \bullet \\ | \\ \bullet \end{array}) = \{{}_2f\}_2 = f^{(1)}(f^{(1)}(f)),$$

where, $f^{(M)}(K_1, K_2, \dots, K_M)$, $K_t \in \mathbb{R}^m, t = 1, 2, \dots, M$ is the M th order Fréchet derivative of f . For more detailed definition of these functions see [2, 9].

The following theorem gives the coefficients of the linear combination of $y^{(q)}$ for a general q in terms of elementary differentials [9]:

Theorem 1. *Let y be the solution of the autonomous problem (1). Then*

$$y^{(q)} = \sum_{r(t)=q} \alpha(t)F(t). \quad (8)$$

The stability function of a Runge-Kutta method is given by

$$R(\hat{h}) = \frac{\det [I - \hat{h}A + \hat{h}eb^T]}{\det [I - \hat{h}A]}, \quad (9)$$

where, $\hat{h} = \lambda h$, here λ is typically an eigenvalue of the jacobian matrix of f that is equivalently the eigenvalues of the linearized equation.

3 One step methods

In this section, we firstly study the situation for the one-step methods. The idea will be extended to a general linear multistep methods in the next sections. Consider the following linear one step method

$$y_{n+1} = y_n + h(\beta_0 f_n + \beta_1 f_{n+1}) \quad (10)$$

for a consistent method we have $\beta_0 + \beta_1 = 1$. Applying the above rule on N successive steps to advance the solution from x_0 to x_N we obtain

$$\begin{aligned} y_1 &= y_0 + h(\beta_0 f_0 + \beta_1 f_1) \\ y_2 &= y_1 + h(\beta_0 f_1 + \beta_1 f_2) \\ &\vdots \\ y_N &= y_{N-1} + h(\beta_0 f_{N-1} + \beta_1 f_N). \end{aligned}$$

Introducing the following slopes

$$k_1 = f(x_0, y_0), k_2 = f(x_1, y_1), \dots, k_{N+1} = f(x_N, y_N),$$

we can write (10) as a $(N+1)$ -stage Runge-Kutta method with the steplength $H = Nh$,

$$y_{m+1} = y_m + \frac{H}{N}(\beta_0 k_1 + k_2 + \dots + k_N + \beta_1 k_{N+1})$$

where, $t_m = x_0, t_{m+1} = t_m + H = x_N$ and,

$$\begin{aligned}
k_1 &= f(t_m, y_m) \\
k_2 &= f\left(t_m + \frac{H}{N}, y_m + \frac{H}{N}(\beta_0 k_1 + \beta_1 k_2)\right) \\
&\vdots \\
k_{N+1} &= f\left(t_m + H, y_m + \frac{H}{N}(\beta_0 k_1 + k_2 + \cdots + k_N + \beta_1 k_{N+1})\right).
\end{aligned}$$

Thus, the corresponding Butcher array takes the following form

$$\begin{array}{c|cccc}
0 & 0 & & & \\
\frac{1}{N} & \frac{\beta_0}{N} & \frac{\beta_1}{N} & & \\
\frac{2}{N} & \frac{\beta_0}{N} & \frac{1}{N} & \frac{\beta_1}{N} & \\
\vdots & \vdots & \vdots & \ddots & \\
\frac{N}{N} & \frac{\beta_0}{N} & \frac{1}{N} & \cdots & \frac{1}{N} & \frac{\beta_1}{N} \\
\hline
& \frac{\beta_0}{N} & \frac{1}{N} & \cdots & \frac{1}{N} & \frac{\beta_1}{N}
\end{array} \quad (11)$$

3.1 Local truncation error

In this section we obtain the LTE and the order of the reduced method (11). To determine the order of the deduced RK scheme we verify the order conditions for the Butcher array (11):

$$\psi(\bullet) = \sum_{i=1}^{N+1} b_i = 1, \quad (12)$$

$$\psi(\downarrow) = \sum_{i=1}^{N+1} b_i c_i = \frac{1}{2}. \quad (13)$$

Since we have $\beta_0 + \beta_1 = 1$, substituting the data from (11) we observe that the first condition always holds

$$\sum_{i=1}^{N+1} b_i = 1 \Rightarrow \frac{\beta_0}{N} + \frac{1}{N} + \cdots + \frac{1}{N} + \frac{\beta_1}{N} = 1.$$

However, for the second condition to be valid we have,

$$\begin{aligned}
\frac{\beta_0}{N} \cdot 0 + \frac{1}{N} \left(\frac{1}{N} + \frac{2}{N} + \cdots + \frac{N-1}{N} \right) + \frac{\beta_1}{N} \cdot \frac{N}{N} \\
= \frac{N^2 - N + 2\beta_1 N}{2N^2} = \frac{1}{2}, \text{ only if } \beta_1 = \frac{1}{2}.
\end{aligned}$$

where, we have used $\beta_0 + \beta_1 = 1$. Therefore, the only possible second order one step method is the trapezoidal rule.

Now, we can find the general form of the local truncation error for both first (forward and backward euler) and second order (Trapezoidal) methods.

For the second order method we have, $\beta_1 = \frac{1}{2}$,

$$\begin{aligned} \psi(t_1 = \text{---}) &= \sum_{i=1}^{N+1} b_i c_i^2 = \frac{2N^2 + 1}{6N^2}, \\ \psi(t_2 = \text{---}) &= \sum_{i,j=1}^{N+1} b_i a_{ij} c_j = \frac{1}{N} \left[\frac{1}{2}, 1, \dots, 1, \frac{1}{2} \right] A c = \frac{2N^2 + 1}{12N^2}. \end{aligned}$$

It is evident that in the limit the above order conditions, as $N \rightarrow \infty$, tend to the following infinite dimensional exact order conditions:

$$\begin{aligned} \sum_{i=1}^{\infty} b_i c_i^2 &= \frac{1}{3}, \\ \sum_{i,j=1}^{\infty} b_i a_{ij} c_j &= \frac{1}{6}. \end{aligned}$$

The corresponding elementary differentials with the rooted trees t_1 and t_2 are,

$$F(t_1) = f^{(2)}(f, f), \quad F(t_2) = f^{(1)}(f^{(1)}(f)).$$

To specify the LTE of second order scheme we need a representation of $y^{(3)}$ in terms of elementary differentials. According to Theorem 1, we have

$$y^{(3)} = F(t_1) + F(t_2), \quad (14)$$

therefore, the principle term in local truncation error (PLTE) for $\beta_1 = \frac{1}{2}$, where $p = 2$ reads

$$\begin{aligned} \text{PLTE} &= \frac{H^3}{3!} \sum_{r(t)=3} \alpha(t) [1 - \gamma(t)\psi(t)] F(t) \quad (15) \\ &= \frac{H^3}{3!} \left(-\frac{1}{2N^2}\right) (F(t_1) + F(t_2)) \\ &= -\frac{1}{12} N h^3 y^{(3)}(x_0) \\ &= -\frac{1}{12} h^2 (x_N - x_0) y^{(3)}(x_0), \end{aligned}$$

and then,

$$\text{LTE} = -\frac{1}{12}h^2(x_N - x_0)y^{(3)}(x_0) + O(h^3). \quad (16)$$

We can also regard (16) as the global error in N steps of the trapezoidal rule. Similarly, for $\beta_1 \neq \frac{1}{2}$ we obtain

$$\text{LTE} = \frac{1}{2}N(N-1+2\beta_1)h^2y^{(2)}(x_0) + O(h^3). \quad (17)$$

3.2 Stability regions

To construct the stability function of (11) we simply note that

$$\hat{H}A = \frac{\hat{H}}{N} \begin{pmatrix} \beta_0 & \beta_1 & & & \\ & & \ddots & & \\ & & & \ddots & \\ \beta_0 & 1 & \dots & 1 & \beta_1 \end{pmatrix}, \quad \hat{H}eb^T = \frac{\hat{H}}{N} \begin{pmatrix} \beta_0 & 1 & \dots & 1 & \beta_1 \\ \beta_0 & 1 & & & \beta_1 \\ \vdots & \vdots & & & \vdots \\ \beta_0 & 1 & \dots & 1 & \beta_1 \end{pmatrix},$$

thus we have

$$I - \hat{H}A + \hat{H}eb^T = \begin{pmatrix} 1 + \hat{H}\frac{\beta_0}{N} & \frac{\hat{H}}{N} & \dots & \frac{\hat{H}}{N} & \frac{\hat{H}}{N}\beta_1 \\ & 1 + \hat{H}\frac{\beta_0}{N} & & \frac{\hat{H}}{N} & \frac{\hat{H}}{N}\beta_1 \\ & & \ddots & & \vdots \\ & & & 1 + \hat{H}\frac{\beta_0}{N} & \frac{\hat{H}}{N}\beta_1 \\ & & & & 1 \end{pmatrix},$$

and thereby,

$$\det[I - \hat{H}A + \hat{H}eb^T] = (1 + \hat{H}\frac{\beta_0}{N})^N.$$

Similarly, we can show that the following relation is also valid

$$\det[I - \hat{H}A] = (1 - \hat{H}\frac{\beta_1}{N})^N.$$

Inserting the above results into (9) we obtain

$$R(\hat{H}) = \frac{(1 + \hat{H}\frac{\beta_0}{N})^N}{(1 - \hat{H}\frac{\beta_1}{N})^N}. \quad (18)$$

which is the stability function of (11).

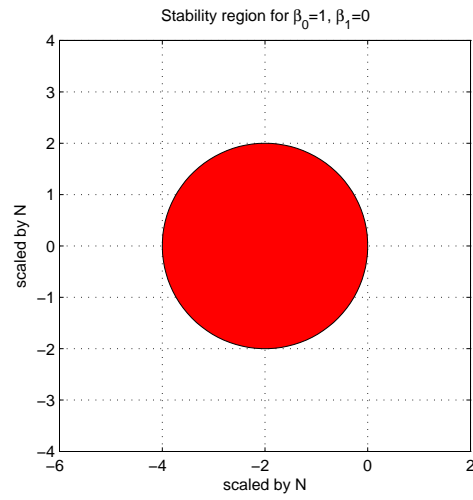


Figure 1: The stability region of the new RK method with $\beta_0 = 1, \beta_1 = 0$

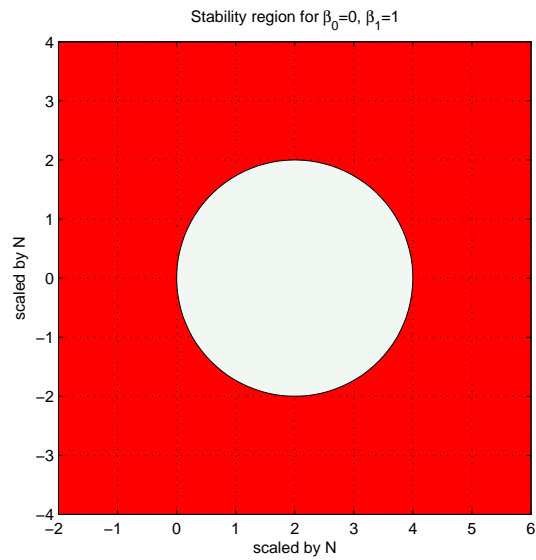


Figure 2: The stability region of the new RK method with $\beta_0 = 0, \beta_1 = 1$

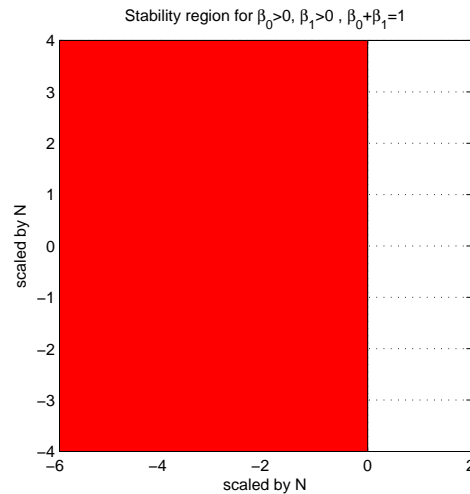


Figure 3: The stability region of the new RK method with $\beta_0 > 0, \beta_1 > 0, \beta_0 + \beta_1 = 1$

In Figures 1, 2, 3, the absolute stability regions of the given RK method (11) have been demonstrated for various values of β_0 and β_1 . We observe that in the case $\beta_0 > 0, \beta_1 > 0, \beta_0 + \beta_1 = 1$ the new RK method is A-stable and in the cases $\beta_0 = 0, \beta_1 = 1$ and $\beta_0 = 1, \beta_1 = 0$ the absolute stability regions, in terms of \hat{h} , tend to the same regions of the forward and backward Euler methods, respectively.

4 Linear multistep methods

In this section, we present the general theory for an arbitrary linear multistep method. We demonstrated the main idea by using the one-step and multi-step starting procedures. In both cases, we obtain the corresponding Runge-Kutta scheme and the LTE of this method provide an estimation of the global truncation error of the given LMM.

4.1 A single step method as starting procedure

Now, we represent a general linear multistep method (LMM) in the form of a Runge-Kutta scheme. For simplicity, we use the trapezoidal rule as starting procedure of the LMM. We show the starting values $\{y_{n+j}\}_{j=1}^{k-1}$ as a linear combination of the y_n and $\{f_{n+j}\}_{j=0}^{k-1}$. Therefore, starting with

$$y_{n+1} = y_n + \frac{h}{2}(f_n + f_{n+1}),$$

we find

$$y_{n+j} = y_n + \frac{h}{2}(f_n + 2f_{n+1} + \cdots + 2f_{n+j-1} + f_{n+j}), \quad j = 1, 2, \dots \quad (19)$$

therefore, we have

$$\begin{aligned} \sum_{j=0}^k \alpha_j y_{n+j} &= y_{n+k} + \left(\sum_{j=0}^{k-1} \alpha_j \right) y_n + \frac{h}{2} \left\{ \left(\sum_{j=1}^{k-1} \alpha_j \right) f_n + \left(\alpha_1 + 2 \sum_{j=2}^{k-1} \alpha_j \right) f_{n+1} \right. \\ &\quad \left. + \cdots + \left(\alpha_{k-2} + 2\alpha_{k-1} \right) f_{n+k-2} + \alpha_{k-1} f_{n+k-1} \right\}. \quad (20) \end{aligned}$$

Substituting (20) into the LMM (1), we obtain

$$\begin{aligned} y_{n+k} &= - \sum_{j=0}^{k-1} \alpha_j y_{n+j} + h \sum_{j=0}^{k-1} \beta_j f_{n+j} \\ &= - \left(\sum_{j=0}^{k-1} \alpha_j \right) y_n + h \left\{ \left(\beta_0 - \frac{1}{2} \sum_{j=1}^{k-1} \alpha_j \right) f_n + \left(\beta_1 - \frac{1}{2} \alpha_1 - \sum_{j=2}^{k-1} \alpha_j \right) f_{n+1} \right. \\ &\quad \left. + \cdots + \left(\beta_{k-2} - \frac{1}{2} \alpha_{k-2} - \alpha_{k-1} \right) f_{n+k-2} \right. \\ &\quad \left. + \left(\beta_{k-1} - \frac{1}{2} \alpha_{k-1} \right) f_{n+k-1} + \beta_k f_{n+k} \right\}, \end{aligned}$$

or,

$$y_{n+k} = y_n + h \sum_{i=1}^{k+1} b_i f_{n+i-1}, \quad (21)$$

where,

$$\begin{aligned} b_1 &= \beta_0 - \frac{1}{2} \sum_{j=1}^{k-1} \alpha_j, \\ b_i &= \beta_{i-1} - \frac{1}{2} \alpha_{i-1} - \sum_{j=i}^{k-1} \alpha_j, \quad 2 \leq i \leq k-1 \\ b_k &= \beta_{k-1} - \frac{1}{2} \alpha_{k-1} \\ b_{k+1} &= \beta_k. \end{aligned}$$

It is straightforward to show that

$$\sum_{i=1}^{k+1} b_i = k.$$

On defining

$$K_{i+1} = f(x_{n+i}, y_{n+i}) = f(x_n + ih, y_n + \frac{h}{2}(f_n + 2f_{n+1} + \dots + 2f_{n+i-1} + f_{n+i})),$$

the scheme (21) can be represented in the form of the following RK method

$$y_{m+1} = y_m + H \sum_{i=1}^{k+1} b'_i K_i,$$

where,

$$K_1 = f(t_m, y_m),$$

$$K_i = f(t_m + c_i H, y_m + H \sum_{j=1}^i a_{ij} K_j), \quad i = 2, 3, \dots, k+1.$$

We will change the subscript n to m in order to show that when LMM runs from y_n to y_{n+k} the RK scheme runs just a single step from $y_m = y_n$ to $y_{m+1} = y_{n+k}$. Therefore, we set

$$\begin{aligned} t_m &= x_n, t_{m+1} = x_n + H, \\ a_{ij} &= \begin{cases} \frac{1}{2k}, & j = 1, i, \\ \frac{1}{k}, & 2 \leq j \leq i-1, \\ 0, & j > i \text{ or } i = 1. \end{cases} \\ c_i &= \frac{i-1}{k}, \\ b'_i &= \frac{b_i}{k}, \\ H &= kh. \end{aligned}$$

In this case, the first order condition holds

$$\sum_{i=1}^{k+1} b'_i = 1,$$

but, the second order condition no longer holds

$$\sum_{i=1}^{k+1} b'_i c_i = \frac{k+1}{2k}.$$

In $k \rightarrow \infty$ this condition turns out to be the exact order condition. Again, the principle local truncation error (PLTE), where $p = 1$ reads

$$\text{PLTE} = \frac{H^2}{2!} \sum_{r(t)=2} \alpha(t)[1 - \gamma(t)\psi(t)]F(t) = -\frac{1}{2}h(x_k - x_0)y^{(2)}(x_0),$$

and therefore, we have

$$\text{LTE} = -\frac{1}{2}h(x_k - x_0)y^{(2)}(x_0) + O(h^2),$$

which is the global error in k times application of a k -step LMM with Trapezoidal rule as starting procedure.

4.2 A Runge-Kutta method as starting procedure

Now, we have considered the general explicit Runge-Kutta method (5)-(6) as the starting procedure for the linear multistep method and we find the Runge-Kutta representation of the given LMM (1). Based on this starting procedure, we have obtained approximate values for y_{n+j} , $j = 1, \dots, k-1$ as follows

$$y_{n+j} = y_n + jh \sum_{i=1}^s b_i k_i^{(j)},$$

$$k_i^{(j)} = f(x_n + jc_i h, y_n + jh \sum_{l=1}^{s+1} a_{il} k_l^{(j)}), \quad i = 1, \dots, s+1,$$

where $c_{s+1} = 1$, $a_{i,s+1} = 0$, $i = 1, 2, \dots, s+1$, $a_{s+1,j} = b_j$, $j = 1, 2, \dots, s$. For an implicit method corresponding to the y_{n+k} , we define

$$k_{s+1}^{(k)} = f(x_n + kc_{s+1}h, y_n + kh \sum_{l=1}^{s+1} a_{s+1,l} k_l^{(j)}).$$

By inserting these approximations into the LMM (1) we obtain

$$y_{n+k} = -\sum_{j=0}^{k-1} \alpha_j y_{n+j} + h \sum_{j=0}^k \beta_j f_{n+j}$$

$$= -\sum_{j=0}^{k-1} \alpha_j y_n - h \sum_{j=1}^{k-1} \sum_{i=1}^s \alpha_j b_i k_i^{(j)} + h \sum_{j=0}^k \beta_j k_{s+1}^{(j)},$$

there is $s(k-1) + 2$ different $k_i^{(j)}$ in the above representation, however we do not distinguish them and consider $(s+1)(k-1) + 1$ moments. The advantage of ignoring the similarity in the moments is that the resulted Butcher array is simpler to work and it is convenient to prove the theorems. Let,

$$t_m = x_n, H = kh, t_{m+1} = t_m + H = x_{n+k}$$

and

$$\begin{aligned}\bar{c}_{i+(j-1)(s+1)} &= \frac{j}{k} c_i, \quad i = 1, \dots, s+1, j = 1, \dots, k-1, \\ \bar{c}_{(s+1)(k-1)+1} &= \frac{1}{k},\end{aligned}$$

and

$$\begin{aligned}b_1^{(1)} &= -\frac{1}{k}(\alpha_1 b_1 - \beta_0), \\ b_i^{(j)} &= -\frac{1}{k} j \alpha_j b_i, \quad i = 1, \dots, s, j = 1, \dots, k-1, \\ b_{s+1}^{(j)} &= \frac{1}{k} \beta_j, j = 1, \dots, k,\end{aligned}$$

the vector \bar{b} consist of these values:

$$\bar{b} = [b^{(1)}, b^{(2)}, \dots, b^{(k-1)}, b_{s+1}^{(k)}].$$

Therefore, we find the new Runge-Kutta scheme

$$\begin{aligned}y_{m+1} &= y_m + H \sum_{i=1}^{\bar{s}} \bar{b}_i \bar{k}_i, \\ k_i^{(j)} &= f\left(t_m + \frac{j}{k} c_i H, y_m + \frac{j}{k} H \sum_{l=1}^{s+1} a_{il} k_l^{(j)}\right), \quad i = 1, \dots, s+1,\end{aligned}$$

where, $\bar{s} = (s+1)(k-1) + 1$ and

$$\bar{k} = [k^{(1)}, k^{(2)}, \dots, k^{(k-1)}, k_{s+1}^{(k)}].$$

The corresponding Butcher array is given in Table 1. where $\mathbf{0}$ is a $(s+1) \times 1$ zero vector. Introducing D as a $(k-1) \times (k-1)$ diagonal matrix

$$D = \frac{1}{k} \begin{bmatrix} 1 & & & & \\ & 2 & & & \\ & & 3 & & \\ & & & \ddots & \\ & & & & k-1 \end{bmatrix}.$$

The more compact form of the above scheme is resulted.

Table 1: Butcher array of the Runge-Kutta representation of LMM (1) scheme

$\frac{1}{k} \mathbf{c}$	$\frac{1}{k} \begin{bmatrix} A \\ b \\ \mathbf{0} \end{bmatrix}$		
$\frac{2}{k} \mathbf{c}$		$\frac{2}{k} \begin{bmatrix} A \\ b \\ \mathbf{0} \end{bmatrix}$	
\vdots		\ddots	\ddots
$\frac{k-1}{k} \mathbf{c}$			$\frac{k-1}{k} \begin{bmatrix} A \\ b \\ \mathbf{0} \end{bmatrix}$
$\frac{k}{k}$		\bar{b}^T	
		\bar{b}^T	

Table 2: Butcher array of the Runge-Kutta representation of LMM (1) scheme

$\bar{\mathbf{c}}$	$D \otimes \begin{bmatrix} A \\ b \\ \mathbf{0} \end{bmatrix}$
	\bar{b}^T
	\bar{b}^T

Now, to verify the order of the new Runge-Kutta scheme suppose that the order of LMM (1) and Runge-Kutta scheme (5)-(4) are p and \bar{p} , respectively. Then, we can prove the following theorem.

Theorem 2. *If the Runge-Kutta scheme as starting procedure has order \bar{p} and the order of linear multistep method (1) is p , then the corresponding Runge-Kutta method with Butcher array in Table 2 is of order $\min\{p, \bar{p}\}$.*

Proof. The order condition corresponding to the tree



with m leaves is

$$\sum_{i=1}^s b_i c_i^m = \frac{1}{m+1}, \quad m = 0, 1, \dots, \bar{p}$$

for any $m \leq \min\{p, \bar{p}\}$ we prove that

$$\sum_{i=1}^s \bar{b}_i \bar{c}_i^m = \frac{1}{m+1}, \quad m = 0, 1, \dots, \min\{p, \bar{p}\}.$$

According to the definition of \bar{c} and \bar{b} , we have

$$\begin{aligned}
\sum_{i=1}^{\bar{s}} \bar{b}_i \bar{c}_i^m &= - \sum_{j=1}^{k-1} \frac{1}{k} j \alpha_j \sum_{i=1}^s b_i \left(\frac{j}{k} c_i\right)^m + \frac{1}{k} \sum_{j=1}^k \beta_j \left(\frac{j}{k}\right)^m \\
&= - \frac{1}{m+1} \frac{1}{k^{m+1}} \sum_{j=1}^{k-1} j^{m+1} \alpha_j + \frac{1}{k^{m+1}} \sum_{j=1}^k j^m \beta_j \\
&= \frac{1}{(m+1)k^{m+1}} \left(k^{m+1} - \sum_{j=0}^k j^{m+1} \alpha_j \right) + \frac{1}{k^3} \sum_{j=0}^k j^m \beta_j \\
&= \frac{1}{m+1}.
\end{aligned}$$

note that in the above relations we have used the order conditions for starting Runge-Kutta method as well as the order conditions for linear multistep methods:

$$\frac{1}{m+1} \sum_{j=0}^k j^{m+1} \alpha_j = \sum_{j=0}^k j^m \beta_j, \quad m = 2, \dots, p.$$

The rest of the proof is closely related to the block structure of the Butcher array in Table 1. The extracted elements of the Butcher array is demonstrated in the Table 3.

Table 3: Elements of Butcher array of Table 1

$\frac{l}{k} c$	$\frac{l}{k} \begin{bmatrix} A \\ b \end{bmatrix} \mathbf{0}$	$, \quad l = 1, 2, \dots, k-1$
	$b^{(l)}$	
1	\bar{b}^T	
	$b_{s+1}^{(k)} = \frac{1}{k} \beta_k$	

These partial elements will help us to exactly find the effect of them in multiple sums in order conditions.

The last condition to complete the proof of the third order conditions is $\sum_{i,j=1}^{\bar{s}} \bar{b}_i \bar{a}_{ij} \bar{c}_j$. The role of each element in summation, separately, is

$$\begin{aligned}
&\sum_{i,j=1}^s \left(-\frac{l}{k} \alpha_l b_i\right) \left(\frac{l}{k} a_{ij}\right) \left(\frac{l}{k} c_j\right) + \left(\frac{1}{k} \beta_l\right) \sum_{i=1}^s \left(\frac{l}{k} b_i\right) \left(\frac{l}{k} c_i\right), \quad l = 1, 2, \dots, k-1 \\
&\frac{1}{k} \beta_k \left(\sum_{i=1}^{\bar{s}} \bar{b}_i \bar{c}_i\right),
\end{aligned}$$

summing up these terms we obtain

$$\begin{aligned} \sum_{i,j=1}^{\bar{s}} \bar{b}_i \bar{a}_{ij} \bar{c}_j &= \sum_{l=1}^{k-1} \left(\sum_{i,j=1}^s \left(-\frac{l}{k} \alpha_l b_i \right) \left(\frac{l}{k} a_{ij} \right) \left(\frac{l}{k} c_j \right) + \left(\frac{1}{k} \beta_l \right) \sum_{i=1}^s \left(\frac{l}{k} b_i \right) \left(\frac{l}{k} c_i \right) \right) \\ &\quad + \frac{1}{k} \beta_k \left(\sum_{i=1}^{\bar{s}} \bar{b}_i \bar{c}_i \right) \\ &= \frac{1}{6k^3} \left(k^3 - \sum_{l=1}^k l^3 \alpha_l \right) + \frac{1}{2k^3} \sum_{l=0}^k l^2 \beta_l \\ &= \frac{1}{6}. \end{aligned}$$

similarly we can prove the higher order conditions. \square

4.3 Local truncation error of the new Runge-Kutta method

We have shown that the order of the new Runge-Kutta method with Butcher array in Table 2 is $p^* = \min\{p, \bar{p}\}$. Ignoring the effect of the roundoff errors, we can consider LTE of this method as the global error of the given linear multistep method in evaluation of y_{n+k} . The LTE of this scheme now reads

$$\text{LTE} = \frac{h^{p^*+1}}{(p^*+1)!} \sum_{r(t)=p^*+1} \alpha(t) [1 - \gamma(t)\psi(t)] F(t) + O(h^{p^*+2}). \quad (22)$$

5 Numerical illustrations

Example 1. As an example we consider the Heun's third order 3-stage formula

$$\begin{array}{c|cc} 0 & & \\ \frac{1}{3} & \frac{1}{3} & \\ \frac{2}{3} & 0 & \frac{2}{3} \\ \hline \frac{1}{4} & 0 & \frac{3}{4} \end{array}$$

as starting procedure for the third order convergent linear multistep method

$$y_{n+3} + \frac{1}{4}y_{n+2} - \frac{1}{2}y_{n+1} - \frac{3}{4}y_n = \frac{h}{8}[19f_{n+2} + 5f_n], \quad (23)$$

the corresponding Runge-Kutta method is

$$\begin{array}{c|cccccc}
 0 & & & & & & \\
 \frac{1}{9} & \frac{1}{9} & & & & & \\
 \frac{2}{9} & 0 & \frac{2}{9} & & & & \\
 \frac{1}{3} & \frac{1}{12} & 0 & \frac{3}{12} & & & \\
 0 & 0 & 0 & 0 & 0 & & \\
 \frac{2}{9} & 0 & 0 & 0 & 0 & \frac{2}{9} & \\
 \frac{4}{9} & 0 & 0 & 0 & 0 & 0 & \frac{4}{9} \\
 \frac{2}{3} & 0 & 0 & 0 & 0 & \frac{2}{12} & 0 & \frac{6}{12} \\
 1 & \frac{6}{24} & 0 & \frac{3}{24} & 0 & -\frac{1}{24} & 0 & -\frac{3}{24} & \frac{19}{24} & 0 \\
 \hline
 & \frac{6}{24} & 0 & \frac{3}{24} & 0 & -\frac{1}{24} & 0 & -\frac{3}{24} & \frac{19}{24} & 0
 \end{array} \tag{24}$$

This method is of order $p^* = \min\{3, 3\} = 3$. Substituting the above data in LTE (7) we find that

$$\begin{aligned}
 \text{LTE} &= \frac{h^4}{4!} \sum_{r(t)=4} \alpha(t)[1 - \gamma(t)\psi(t)]F(t) + O(h^5) \\
 &= \frac{h^4}{4!} \left(\frac{73}{729}y^{(4)} - \frac{7}{729}\{2f^2\}_2 - \frac{28}{729}\{3f\}_3 \right) + O(h^5),
 \end{aligned}$$

where all functions are evaluated at $x = x_n$ and $y = y_n$. Note that this formulation maintains the actual order of both schemes, while the error term is only exact for $\{1, x\}$.

To numerical illustrations, we consider (2) with the following data [9]

$$f(x, y) = [v, v(v - 1)/u]^T, \quad x \in [0, 1], \tag{25}$$

where,

$$y = [u, v]^T, \quad y(0) = [1/2, -3]^T.$$

In this test we take $N = 51$ with $h = 1/50$ for 3-step method (23) and $H = Nh = 3/50$ for the corresponding Runge-Kutta scheme (24).

Figure 4 illustrates the global error (accumulation error) of (23) for test problem (25). The estimation of GTE of (23) is shown in Figure 5. As we have proven the LTE of (24) is the GTE of (23). However, to find the true LTE we make localizing assumptions in implementation of (24), i.e., in evaluation of y_{n+1} we assume that $y_n = y(x_n)$. The comparison of the third portions of Figure 4 and Figure 5 justify the efficiency of the given estimation. The negligible difference in the error is due to roundoff errors.

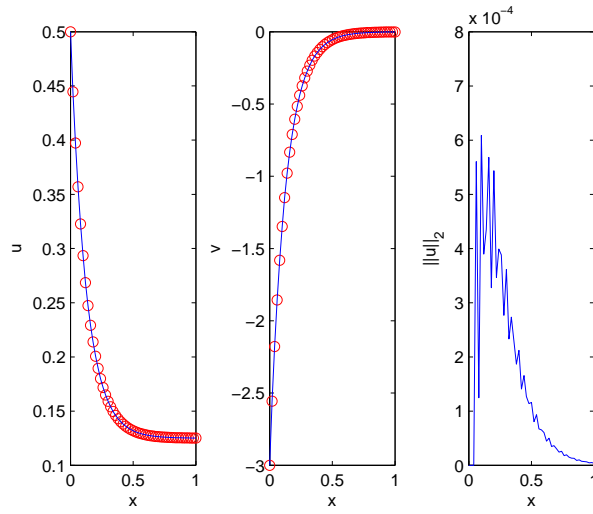


Figure 4: The numerical (red circles) and exact solutions (solid line) of (25) with (23), and the GTE of the method

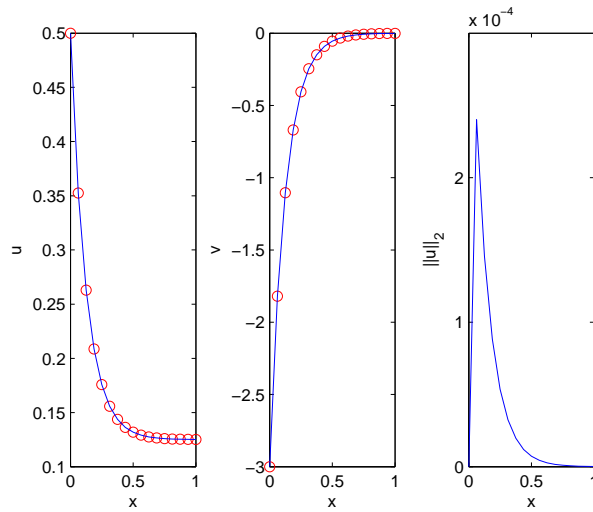


Figure 5: The numerical (red circles) and exact solutions (solid line) of (25) with (24), and the LTE of the RK method

Example 2. In this example, we consider a four-step, fourth-order total variation bounded (TVB(4,4)) linear multistep scheme (1) with the data given in Table 4 [11].

Table 4: The coefficients of the four-step, fourth-order linear multistep scheme (TVB(4,4))

	α_i	β_i
$i = 0$	-0.345464734400857	-0.620278703629274
$i = 1$	1.494730011212510	2.229909318681302
$i = 2$	-2.777506277494861	-3.052866947601049
$i = 3$	2.628241000683208	1.618795874276609
$i = 4$	1	0

We utilize the fourth order classic Runge-Kutta method as a starting procedure.

$$\begin{array}{c|ccc}
 0 & 0 & & \\
 \frac{1}{2} & \frac{1}{2} & & \\
 \frac{1}{2} & 0 & \frac{1}{2} & \\
 1 & 0 & 0 & 1 \\
 \hline
 & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
 \end{array} \quad (26)$$

It turns out that the corresponding Runge-Kutta scheme takes the form given in Butcher array (27), where the elements of b is given in the Table 5.

Table 5: The elements of vector b in Runge-Kutta scheme (27)

	b_i	b_{i+8}
$i = 1$	-0.092789258773464	-0.231458856457905
$i = 2$	0.124560834267709	-0.763216736900262
$i = 3$	0.124560834267709	0.328530125085401
$i = 4$	0.062280417133855	0.657060250170802
$i = 5$	0.557477329670325	0.657060250170802
$i = 6$	-0.231458856457905	0.328530125085401
$i = 7$	-0.462917712915810	0.404698968569152
$i = 8$	-0.462917712915810	0

$$\begin{array}{c|cccccccc}
 0 & 0 & & & & & & & \\
 \frac{1}{8} & \frac{1}{8} & & & & & & & \\
 \frac{1}{8} & 0 & \frac{1}{8} & & & & & & \\
 \frac{1}{4} & 0 & 0 & \frac{1}{4} & & & & & \\
 \frac{1}{4} & \frac{1}{24} & \frac{1}{12} & \frac{1}{12} & \frac{1}{24} & 0 & & & \\
 0 & & & & & & 0 & & \\
 \frac{1}{4} & & & & & & \frac{1}{4} & & \\
 \frac{1}{4} & & & & & & 0 & \frac{1}{4} & \\
 \frac{1}{2} & & & & & & 0 & 0 & \frac{1}{2} \\
 \frac{1}{2} & & & & & & \frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{12} & 0 \\
 0 & & & & & & & & & & 0 \\
 \frac{3}{8} & & & & & & & & & & \frac{3}{8} \\
 \frac{3}{8} & & & & & & & & & & 0 & \frac{3}{8} \\
 \frac{3}{4} & & & & & & & & & & 0 & 0 & \frac{3}{4} \\
 \frac{3}{4} & & & & & & & & & & \frac{1}{8} & \frac{1}{4} & \frac{1}{4} & \frac{1}{8} & 0 \\
 1 & & & & & & & & & & & & & & b \\
 \hline
 & & & & & & & & & & & & & & b
 \end{array} \tag{27}$$

The test problem is the same as the previous example. For numerical illustrations we take $N = 120$, thus we have $h = 1/120$ and accordingly $H = 4/120 = 1/30$. The numerical results including the GTE of TVB(4,4) scheme and LTE of (27) are presented in Figure 6 and Figure 7, respectively. By comparison, we find the excellent estimation of global truncation error of TVB(4,4) based on new Runge-Kutta method. The maximum error in this estimation is $2.67E - 05$.

6 conclusion

In this paper, we have developed an estimation for the global truncation error of a linear multistep method. The global error analysis is more complicated in comparison with the local error analysis. The key idea is the representation of several steps of the LMM as a single step of a corresponding Runge-Kutta method. Therefore, the analysis of global error of a LMM accomplished by estimating the local truncation error the corresponding new Runge-Kutta method. We have demonstrated the theoretical aspects for some important class of linear multistep methods with total variation bounded (TVB) property, which is a crucial property in selecting an appropriate time marching method for solving nonlinear conservation laws [11].

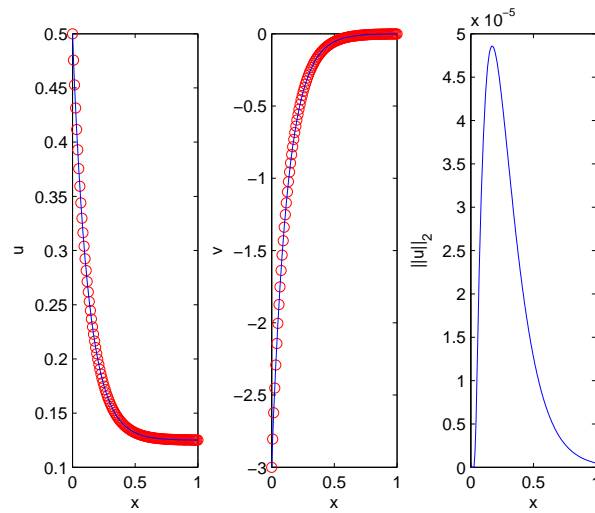


Figure 6: The numerical (circles) and exact solutions (solid line) of (25) with TVB(4,4), and the GTE of the method

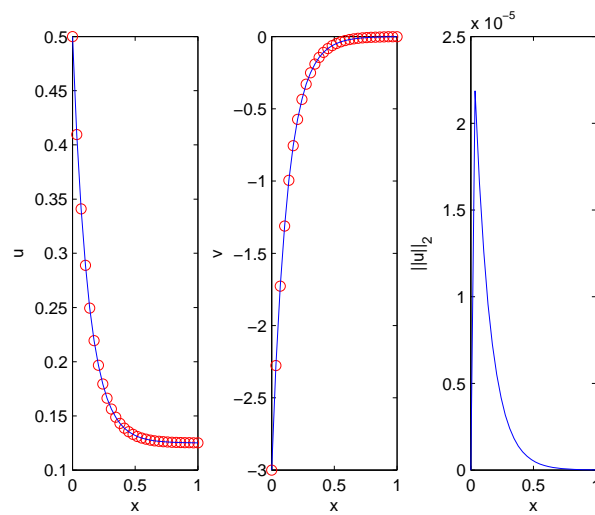


Figure 7: The numerical (circles) and exact solutions (solid line) of (25) with (27), and the LTE of the RK method

References

1. Ascher, U. M. and Petzold, L. R. *Computer methods for ordinary differential equations and differential-algebraic equations*, SIAM, Philadelphia,

- 2008.
2. Butcher, J. C. *Numerical methods for ordinary differential equations*, 2nd Edition, Wiley, 2008.
 3. Cao, Y. and Petzold, L. *A posteriori error estimation and global error control for ordinary differential equations by the adjoint method*, SIAM J. Sci. Comput. 26 (2004), 359-374.
 4. Gottlieb, S., Ketcheson, D. I. and Shu, C. W. *High order strong stability preserving time discretizations*, J. Sci. Comput. 38 (2009) 251-289.
 5. Hadjimichael, Y., Ketcheson, D., Lóczi, L. and Németh, A. *Strong stability preserving explicit linear multistep methods with variable step size*, Submitted.
 6. Henrici, P. *Discrete variable methods in ordinary differential equations*, Wiley, New York, 1962.
 7. Henrici, P. *Error propagation for difference methods*, Wiley, New York, 1963.
 8. Iserles, A. *A First course in the numerical analysis of differential equations*, Cambridge University Press, 1996.
 9. Lambert, J. D., *Numerical methods for ordinary differential systems: The initial value problem*, Wiley, 1993.
 10. Press, W.H., Teukolsky, S.A. and Vetterling, W.T., Flannery, BP. *Numerical recipes: The art of scientific computing*, 3rd ed., New York: Cambridge University Press, 2007.
 11. Ruuth, S. J. and Hundsdorfer, W. *High-order linear multistep methods with general monotonicity and boundedness properties*, Journal of Computational Physics, 209 (2005) 226–248.
 12. Süli, E., and Mayers, D. *An Introduction to numerical analysis*, Cambridge University Press, 2003.

تخمین خطای سراسری روشهای چندگامه خطی توسط روشهای رونگه-کوتا

جواد فرضی

تبریز، دانشگاه صنعتی سهند، دانشکده علوم پایه، گروه ریاضی

دریافت مقاله ۵ دی ۱۳۹۴، دریافت مقاله اصلاح شده ۱۹ اسفند ۱۳۹۴، پذیرش مقاله ۲۳ فروردین ۱۳۹۵

چکیده : در این مقاله خطای برشی سراسری روشهای چندگامه خطی را با کمک خطای برشی محلی روشهای رونگه-کوتا مطالعه می کنیم. ایده اصلی، نمایش یک روش چندگامه خطی با یک روش رونگه-کوتای متناظر است. برای این کار باید چندگام روش چندگامه خطی را به عنوان یک گام ساده روش رونگه-کوتای متناظر در نظر بگیریم. بنابراین، خطای برشی سراسری روش چندگامه خطی از طریق روش رونگه-کوتا فراهم می شود. در این تخمین تاثیرات خطاهای گردکردن را در نظر نمی گیریم. نتایج عددی دقت و کارآمدی تخمین ارائه شده را نمایش می دهد.

کلمات کلیدی : روشهای چندگامه خطی؛ روشهای رونگه-کوتا؛ خطای برشی محلی؛ خطای سراسری؛ تخمین خطا.