# Construction of the ef-based Runge-Kutta methods revisited

R. D'Ambrosio[1], L. Gr. Ixaru[2], B. Paternoster[3]

[1,3] *Dipartimento di Matematica e Informatica, Università degli Studi di Salerno, I-84084 Fisciano*
[2] *"Horia Hulubei" National Institute of Physics and Nuclear Engineering, PO Box MG-6, Magurele, Bucharest, Romania*
*Academy of Romanian Scientists, 54 Splaiul Independenţei, 050094, Bucharest, Romania*

**Abstract**

The purpose of this paper is to revisit the exponential fitting (ef) technique when building up Runge-Kutta methods for solving ordinary differential equations. We propose a modification in such a way that the contamination of the final stage by the errors produced in the internal stages becomes visible. The modified technique is illustrated on a simple version, namely the two-stage explicit Runge-Kutta method, for which we obtain new expressions for the coefficients. The version obtained in this way is then compared for accuracy and stability with that obtained by means of the standard ef technique.

*Keywords:* Ordinary differential equations, Runge-Kutta methods, exponential fitting.

## 1. Introduction

In this paper we focus on the solution by Runge-Kutta methods (RK) of the following initial value problem

$$\begin{cases} y'(x) = f(x, y(x)), \quad x \in [x_0, X], \\ y(x_0) = y_0 \in \mathbb{R}, \end{cases} \tag{1.1}$$

where $f : [x_0, X] \times \mathbb{R} \to \mathbb{R}$ is smooth enough in order to ensure the well-posedness of this problem.
The general algorithm of a one-step RK method is (see [1], [7])

$$y_{n+1} = y_n + h \sum_{i=1}^{s} b_i f(x_n + c_i h, Y_i) \tag{1.2}$$

where

$$Y_i = y_n + h \sum_{j=1}^{s} a_{ij} f(x_n + c_j h, Y_j), \qquad i = 1, 2, ..., s. \tag{1.3}$$

It allows computing $y_{n+1}$ in terms of the input $y_n$ by the formula written in the first row (so called final stage), in which the values of $Y_i$ are as resulting from the set of formulae in the second row (internal stages).
We are actually interested in the cases when the solution $y(x)$ exhibits a shape which recommends RK versions with coefficients $a_{ij}$, $b_i$, $c_i$ obtained by means of the exponential fitting (ef) technique as appropriate solvers of the problem. Such shapes include exponential or oscillatory behaviours, and in each case the set of functions to be used for the ef-based computation of the coefficients is selected in terms of the current behaviour. Many versions derived in this frame have been reported in the literature, to mention only [4], [6], [8], [9], [10], [11] and references therein.
The issue which we want to revisit in this paper is the way of deriving the coefficients. Our intention actually consists in accommodating two features. In fact, the structure of the algorithm shows that the error in $y_{n+1}$ *cumulates* the

*Email address:* rdambrosio@unisa.it, ixaru@theory.nipne.ro, beapat@unisa.it (R. D'Ambrosio[1], L. Gr. Ixaru[2], B. Paternoster[3])

error associated directly to the form of the final stage *and* the errors generated when the intermediary values $Y_i$ were generated in the internal stages but in the literature (see, e.g., [4], [6], [8], [9], [10], [11]), when constructing the coefficients, each stage is treated *separately* and then the error contamination process is disregarded. The problem of concern then consists in modifying the way of constructing the coefficients such that the propagation of the error along the stages becomes visible. To make our proposal of modification easier to follow we choose the very simple case of the explicit two-stage RK method, and the functions $1,\ e^{\mu x},\ x\,e^{\mu x}$ as the basis for the ef approach.
The paper is organized as follows: in section 2 we examine the chosen simple version to find out that in this case the modification affects only the external stage. We then derive the corresponding exponential fitting version according to the revisited technique, and obtain expressions for the coefficients $b_1$ and $b_2$ which are different from those derived in the standard ef-based frame. In section 3 we examine the stability properties of the new algorithm and also derive the formula of its local error. The new and the standard ef-based versions are then compared experimentally on two test equations (section 4) while in section 5 we suggest some possible future developments and applications of the idea suggested in the paper.

## 2. Construction of the method

As announced, we focus on the class of two stage explicit Runge-Kutta methods

$$\begin{array}{lcl} y_{n+1} & = & y_n + h[b_1 f(x_n, Y_1) + b_2 f(x_n + c_2 h, Y_2)] \\ Y_1 & = & y_n \\ Y_2 & = & y_n + h a_{21} f(x_n, Y_1) \end{array} \tag{2.1}$$

which can be represented using the Butcher array (see [1], [7])

$$\begin{array}{c|cc} 0 & & \\ c_2 & a_{21} & \\ \hline & b_1 & b_2 \end{array}.$$

It is well known from Dahlquist barrier results that the maximum attainable order of an $s$-stage explicit Runge-Kutta method is at most $s$ which means, in our case, that the computed methods can attain at most order 2, while the stage order (i.e. the order of the internal stages approximations) is equal to 1 (cfr. [1], [7]). As a consequence, in the classical case the external stage exactly integrates all problems whose solution is expressed as linear combination of the set of functions $\{1, x, x^2\}$, while the internal ones are exact when the solution is linear combination of $\{1, x\}$. In an analogous way, since we aim to derive the exponential fitting version of (2.1), we impose that the final stage exactly integrates the set of functions

$$\{1,\ e^{\mu x},\ x\,e^{\mu x}\}, \tag{2.2}$$

while the internal stages exactly integrate the basis functions

$$\{1,\ e^{\mu x}\}. \tag{2.3}$$

It is well known that the coefficients of standard ef methods are not constant as in the classical case, but they are functions of $z = \mu h$. In the remainder we emphasize this dependence denoting the coefficients of the ef version of (2.1) as $a_{21}(z)$, $b_1(z)$ and $b_2(z)$. When $z$ tends to 0 the coefficients of standard ef methods tend to classical ones. In our revisited ef technique, the coefficients of the resulting methods will depend not only on $z$ but also on the partial derivatives of the function $f$ and, therefore, they depend on the equation we aim to solve. Following the formalism in [5], [6], we introduce the local discretization errors associated to the second internal stage in (2.1)

$$\mathcal{L}_2[h, \mathbf{a}]y(x)\Big|_{x=x_n} = y(x_n + c_2 h) - y_n - h a_{21} f(x_n, Y_1).$$

We next assume the localizing condition $y_n = y(x_n)$ and, therefore, we obtain

$$\mathcal{L}_2[h, \mathbf{a}]y(x)\Big|_{x=x_n} = y(x_n + c_2 h) - y(x_n) - h a_{21} y'(x_n). \tag{2.4}$$

In force of the localizing assumption there is no error in $Y_1$. As for $Y_2$, we annihilate (2.4) on the set of functions (2.3), and in this way it is guaranteed that the second internal stage $Y_2$ in (2.1) is error free for all problems whose solutions are linear combinations of the functions $y(x) = 1$ and $y(x) = e^{\mu x}$. We have:

$$\begin{aligned} \mathcal{L}_2[h, \mathbf{a}]1\Big|_{x=x_n} &= 0, \\ \mathcal{L}_2[h, \mathbf{a}]e^{\mu x}\Big|_{x=x_n} &= e^{\mu(x_n+c_2h)} - e^{\mu x_n} - \mu h a_{21} e^{x_n} = 0. \end{aligned}$$

The last equation gives directly $a_{21}(z) = (e^{c_2 z} - 1)/z$ but, since this has a $0/0$ undeterminacy at $z = 0$, in numerical evaluations a series expansion has to be used for $z$ around 0. We then write:

$$a_{21}(z) = \begin{cases} \dfrac{e^{c_2 z} - 1}{z} & \text{if } |z| > z^* \\ c_2 + \frac{c_2^2}{2} z + \frac{c_2^3}{6} z^2 + \frac{c_2^4}{24} z^3 + \frac{c_2^5}{120} z^4 + O(z^5) & \text{otherwise} \end{cases} \tag{2.5}$$

For $z^* = 10^{-2}$, as we use, the written terms in the series are sufficient for an accuracy with 12 exact figures for any $c_2 \in (0, 1]$.

In short, if this $a_{21}(z)$ is used, the second internal stage in (2.1) is exact if the solution of the problem is a linear combination of the functions 1 and $e^{\mu x}$ such that, since these two functions are solutions of the differential equation

$$y'' - \mu y' = 0,$$

the leading term of the error in the second internal stage can be expressed in the following form

$$err = h^2 F(y''(x) - \mu y'(x)), \tag{2.6}$$

where $F$ is the error constant. To determine this constant we follow the procedure introduced in [5]: we evaluate (2.4) and the error (2.6) on $y(x) = x$ and, due to the invariance to translation, we can consider $x = 0$. Therefore, we have

$$\mathcal{L}_2[h, \mathbf{a}]x\Big|_{x=0} = (c_2 - a_{21})h$$

and

$$err|_{y(x)=x} = -h^2 \mu F.$$

Comparing the last two formulas we obtain

$$F = \frac{-1 - c_2 z + e^{c_2 z}}{z^2}. \tag{2.7}$$

We now consider the local error associated to the external stage $y_{n+1}$ in (2.1). We define

$$\hat{\mathcal{L}}[h, \mathbf{b}]y(x)\Big|_{x=x_n} = y(x_n + h) - y(x_n) - h(b_1 y'(x_n) + b_2 f(x_n + c_2 h, Y_2)). \tag{2.8}$$

In standard derivations of exponentially-fitted Runge-Kutta methods (see, for instance, [4], [6], [8], [9], [10], [11]), the coefficients $b_1$ and $b_2$ are determined under the tacit assumption that $Y_2 = y(x_n + c_2 h)$, that is the error (2.6) is neglected. In a derivation of this type eq. (2.8) has the form

$$\hat{\mathcal{L}}^S[h, \mathbf{b}]y(x)\Big|_{x=x_n} = y(x_n + h) - y(x_n) - h(b_1 y'(x_n) + b_2 y'(x_n + c_2 h)). \tag{2.9}$$

By annihilating this for $y(x)$ belonging to (2.2) we obtain

$$b_1^S(z) = \begin{cases} \dfrac{-1 - c_2 z + e^z(1 + (-1 + c_2)z)}{c_2 z^2}, & \text{if } |z| > z^*, \\ \frac{-1+2c_2}{2c_2} + \frac{-2+3c_2}{6c_2} z + \frac{-3+4c_2}{24c_2} z^2 + \frac{-4+5c_2}{120c_2} z^3 & \\ + \frac{-5+6c_2}{720c_2} z^4 + O(z^5), & \text{otherwise} \end{cases}$$

$$b_2^S(z) = \begin{cases} \dfrac{1-e^z+ze^z}{c_2 z^2 e^{c_2 z}}, & \text{if } |z| > z^*, \\ \frac{1}{2c_2} + \frac{2-3c_2}{6c_2}z + \frac{3-8c_2+6c_2^2}{24c_2}z^2 + \frac{4-15c_2+20c_2^2-10c_2^3}{120c_2}z^3 & \\ + \frac{15c_2^4-40c_2^3+45c_2^2-24c_2+5}{720c_2}z^4 + O(z^5), & \text{otherwise} \end{cases}$$

where the superscript $S$ stands for the standard method.
However, $Y_2$ approximates $y(x_n + c_2 h)$ with the error (2.6) which vanishes only when $y(x) = 1, e^{\mu x}$. Correspondingly, $Y_2$ brings no contribution to the error in $y_{n+1}$ only when $y(x)$ is a linear combination of these functions. In all other possible cases the error (2.6) is nonzero, i.e. the stage value $Y_2$ does not represent $y(x_n + c_2 h)$ exactly. The central problem then consists in determining how the error in $Y_2$ contaminates the error in $y_{n+1}$, and finding a way such that the coefficients $b_1$ and $b_2$ minimize this contamination effect. We write

$$y'(x_n + c_2 h) = f(x_n + c_2 h, y(x_n + c_2 h)) = f(x_n + c_2 h, Y_2 + err)$$

and, on applying the Taylor formula, we obtain

$$y'(x_n + c_2 h) = f(x_n + c_2 h, Y_2) + err\ f_y(x_n + c_2 h, Y_2) + O(err^2) \tag{2.10}$$

and, since $err \sim h^2$ we disregard the last term. This approximation is used for revisiting (2.8), which now gets the form

$$\begin{aligned} \hat{\mathcal{L}}^R[h, \mathbf{b}]y(x)\Big|_{x=x_n} &= y(x_n + h) - y(x_n) - hb_1^R y'(x_n) - hb_2^R[y'(x_n + c_2 h) \\ &- h^2 F f_y(y''(x_n) - \mu y'(x_n))], \end{aligned} \tag{2.11}$$

where the superscript $R$ stands for the *revisited* ef Runge-Kutta methods we aim to derive; hereinafter $f_y$ is the shorthand notation for $f_y(x_n + c_2 h,\ Y_2)$. We evaluate (2.11) on $y(x) = 1,\ e^{\mu x},\ xe^{\mu x}$, obtaining

$$\begin{aligned} \hat{\mathcal{L}}^R[h, \mathbf{b}]1\Big|_{x=0} &= 0, \\ \hat{\mathcal{L}}^R[h, \mathbf{b}]e^{\mu x}\Big|_{x=0} &= -1 + e^z - (b_1^R + b_2^R e^{c_2 z})z, \\ \hat{\mathcal{L}}^R[h, \mathbf{b}]xe^{\mu x}\Big|_{x=0} &= -\frac{z}{\mu^2}[(b_1^R - e^z)\mu - b_2^R(f_y(-1 + e^{c_2 z} - c_2 z) - e^{c_2 z}(1 + c_2 z)\mu)]. \end{aligned}$$

The values of $b_1^R(z)$ and $b_2^R(z)$ that annihilate these expressions are

$$b_1^R(z) = \frac{\alpha(z) f_y h + b_1^S(z)}{\gamma(z) f_y h + 1}, \tag{2.12}$$

$$b_2^R(z) = \frac{b_2^S(z)}{\gamma(z) f_y h + 1}, \tag{2.13}$$

where

$$\alpha(z) = \begin{cases} \dfrac{(1-e^z)(-1+e^{c_2 z} - c_2 z)}{c_2 z^3 e^{c_2 z}}, & \text{if } |z| > z^*, \\ -c_2(\frac{1}{2} + \frac{3-4c_2}{12}z + \frac{3c_2^2-4c_2+2}{24}z^2 + \frac{-24c_2^3+45c_2^2-40c_2+15}{720}z^3 & \\ + \frac{5c_2^4-12c_2^3+15c_2^2-10c_2+3}{720}z^4) + O(z^5), & \text{otherwise} \end{cases}$$

$$\gamma(z) = \begin{cases} \dfrac{1-e^{c_2 z} + c_2 z}{c_2 z^2 e^{c_2 z}}, & \text{if } |z| > z^*, \\ -\frac{c_2}{2} + \frac{c_2^2}{3}z - \frac{c_2^3}{8}z^2 + \frac{c_2^4}{30}z^3 - \frac{c_2^5}{144}z^4 + O(z^5), & \text{otherwise} \end{cases}$$

and this completes the derivation of the revised version.

*Systems of equations*

The coefficients $a_{21}$, $b_1$ and $b_2$ derived before for the standard and revised versions cover the case of the scalar problem (1.1). When systems of equations are of concern they remain as stated for the standard version (the same holds true also for $a_{21}$ in the revised version) but $b_1^R$ and $b_2^R$ become matrices of functions.

To be specific, the system of differential equations to be solved is

$$\begin{cases} y'(x) = f(x, y(x)), \quad x \in [x_0, X], \\ y(x_0) = y_0 \in \mathbb{R}^d, \end{cases} \tag{2.14}$$

where $f : [x_0, X] \times \mathbb{R}^d \to \mathbb{R}^d$ and $d > 1$. The form of eq. (2.4) which serves for the determination of $a_{21}$ is unchanged but the vector form of the scalar eq. (2.10) has now a slightly different expression,

$$y'(x_n + c_2 h) = f(x_n + c_2 h, Y_2) + J(x_n + c_2 h, Y_2)\, err + O(err^2)\,,$$

because the $d \times d$ Jacobian matrix $J$ takes the place of $f_y$. The vector extension of the local truncation error expression (2.11) then is

$$\begin{aligned} \hat{\mathcal{L}}^R[h, \mathbf{b}]y(x)\Big|_{x=x_n} &= y(x_n + h) - y(x_n) - h b_1^R y'(x_n) - h b_2^R [y'(x_n + c_2 h) \\ &- h^2 F J (y''(x_n) - \mu y'(x_n))], \end{aligned} \tag{2.15}$$

where $J$ is the short-hand notation for $J(x_n + c_2 h, Y_2)$ and, to determine the matrices $b_1^R$ and $b_2^R$, we apply a slightly adapted version of the procedure described after eq. (2.11).
Specifically, let $v^{(i)}$, $i = 1, 2, 3$ be three $d$-dimensional column vectors with arbitrary constant elements. We introduce the vectors of functions

$$y(1)(x) = v^{(1)}, \quad y^{(2)}(x) = e^{\mu x} v^{(2)}, \quad y^{(3)}(x) = x e^{\mu x} v^{(3)},$$

and evaluate (2.15) on them. We obtain $\hat{\mathcal{L}}^R[h, \mathbf{b}]y^{(1)}\Big|_{x=0} = T^{(i)} v^{(i)}$ where

$$T^{(1)} = 0, \quad T^{(2)} = I(-1 + e^z) - (b_1^R + b_2^R e^{c_2 z}) z\,,$$

$$T^{(3)} = -\frac{z}{\mu^2}[(b_1^R - I e^z)\mu - b_2^R (J(-1 + e^{c_2 z} - c_2 z) - I e^{c_2 z}(1 + c_2 z)\mu)]\,.$$

0 and $I$ are the $d \times d$ zero and unity matrices, respectively. The system $T^{(2)} = T^{(3)} = 0$ is solved for $b_1^R(z)$ and $b_2^R(z)$ to obtain the result (compare with eqs.(2.12-2.13)):

$$\begin{aligned} b_1^R(z) &= (I + \gamma(z) J h)^{-1} (\alpha(z) J h + b_1^S(z) I), \\ b_2^R(z) &= (I + \gamma(z) J h)^{-1} b_2^S(z)\,. \end{aligned} \tag{2.16}$$

## 3. Properties

*Accuracy*

The local truncation error of standard ef method is

$$\begin{aligned} \hat{\mathcal{L}}^S[h, \mathbf{b}]y(x_n) &= -\frac{h^3}{12}[(-2 + 3c_2) y'''(x_n) + (4\mu - 6c_2\mu - 3c_2 f_y) y''(x_n) \\ &+ \mu(-2\mu + 3c_2\mu + 3c_2 f_y) y'(x_n)] + O(h^4). \end{aligned} \tag{3.1}$$

but for the revisited version we have

$$\hat{\mathcal{L}}^R[h, \mathbf{b}]y(x_n) = -\frac{h^3}{12}(-2 + 3c_2)[y'''(x_n) - 2\mu y''(x_n) + \mu^2 y'(x_n)] + O(h^4). \tag{3.2}$$

We see that in general the two versions have the same order 2. However, the order of the revised version can be easily increased. Indeed, if $c_2 = 2/3$ then the order becomes 3.

*Stability*

We consider the *linear* stability analysis, i.e. we study the stability properties of the two versions (standard and revised) of the Runge-Kutta method (2.1) by means of the linear scalar problem $y' = \lambda y$, where $\mathrm{Re}(\lambda) < 0$. We recall that for both versions the coefficients depend on $z$. For the revised version coefficients $b_1$ and $b_2$ depend also on $\omega = h\lambda$ (they are rational functions of $\omega$); we tacitly assume that $c_2$ is kept fixed.
Applying the internal stage equation in (2.1) we obtain $Y_2 = (1 + \omega a_{21})y_n$ and, by substituting this expression in the external stage equation we derive the recurrence relation $y_{n+1} = R(\omega, z)y_n$, where

$$R(\omega, z) = 1 + \omega[b_1^V + b_2^V] + \omega^2 a_{21}(z)b_2^V$$

is the so-called *stability function* of the standard ($V = S$) or of the revised ($V = R$) version.
This function is then used for building up the three-dimensional stability region, thus extending a concept introduced in [2] for second order differential equations. We recall that

**Definition 3.1.** *The region of the three-dimensional* $(\mathrm{Re}(\omega), \mathrm{Im}(\omega), z)$ *space on which the inequality*

$$|R(\omega, z)| < 1 \tag{3.3}$$

*is satisfied is called a region of stability* $\Omega$ *for the method (2.1).*

Since for fixed $z$ the stability function is a polynomial in $\omega$ for version $S$ but a rational function for version $R$ it is clear that the two 3D stability regions will show differently. To illustrate the difference we present sections through the stability regions by planes $z = -1$, $z = -2$, $z = -3$ and $z = -4$. On Fig.1 we take $c_2 = 3/4$ and show these sections for standard/revised version on the left/right column. Fig.2 presents the same graphs for $c_2 = 2/3$. For the standard version a weak variation with $z$ of stability area is seen for both values of $c_2$: there is some enlargement in width but the maximal height remains practically unchanged. For contrast, a massive increase is seen for the revised version. The revised version is thus clearly better than the standard one for stability.

## 4. Numerical experiments

We now compare for accuracy the standard and revised versions with $c_2 = 3/4$ on three problems (two scalar, one linear and another nonlinear, and a nonlinear system) whose solutions show an exponential behavior. For the revised version we report also on the case $c_2 = 2/3$ which is expected to be of order 3. A fixed stepsize $h = 1/2^i$, with different integer values of $i$, was used for all compared methods. In tables 1-4 we present the relative errors at $x_{max}$, $rerr^S$ and $rerr^R$ for each $h$, and also the improvement factor for the revisited with respect to the standard version,

$$if = \frac{rerr^S}{rerr^R}.$$

We first consider the linear problem

$$\begin{cases} y'(x) = \lambda y + kx^{k-1}e^{\lambda x} \\ y(1) = e^{\lambda}, \end{cases} \tag{4.1}$$

with $x \in [x_{min} = 1, x_{max} = 5]$, whose exact solution is $y(x) = x^k e^{\lambda x}$. If $\mu = \lambda$ and $k = 0, 1$ the solution $y(x)$ falls in the set of functions (2.2) such that both versions provide the exact value of the solution, but for other values of $k$ the solution $y(x)$ does not belong to the set (2.2) and, therefore, none of them is exact. Table 1 shows the results obtained for $k = 2$ and several values of $\lambda$, $\mu = \lambda$, and $h$. The superiority of the revised version is obvious. As expected, the order is 2 for $c_2 = 3/4$ but 3 for $c_2 = 2/3$. The improvement factor is substantial, especially for $c_2 = 2/3$. All these features continue to hold true also when $\mu = 0$, that is when the classical forms of these methods are involved, see Table 2.

Table 1: Performance of the two versions for the problem (4.1) for different values of $\lambda$; case $\mu = \lambda$.

| | | $c_2 = 3/4$ | | | $c_2 = 2/3$ | | |
|---|---|---|---|---|---|---|---|
| | $h$ | $rerr^S$ | $rerr^R$ | $if$ | $rerr^S$ | $rerr^R$ | $if$ |
| $\lambda = -1$ | 1/64 | 3.11(-05) | 2.49(-06) | 12.5 | 2.62(-05) | 9.64(-08) | 272.0 |
| | 1/128 | 7.76(-06) | 6.29(-07) | 12.3 | 6.53(-06) | 1.20(-08) | 543.0 |
| | 1/256 | 1.93(-06) | 1.58(-07) | 12.2 | 1.63(-06) | 1.50(-09) | 1085.1 |
| $\lambda = -2$ | 1/128 | 3.77(-05) | 1.18(-06) | 32.0 | 3.28(-05) | 1.16(-07) | 281.9 |
| | 1/256 | 9.39(-06) | 3.06(-07) | 30.7 | 8.17(-06) | 1.45(-08) | 562.9 |
| | 1/512 | 2.34(-06) | 7.80(-08) | 30.0 | 2.03(-06) | 1.81(-09) | 1124.8 |
| $\lambda = -4$ | 1/128 | 1.65(-04) | 1.68(-06) | 98.3 | 1.45(-04) | 1.01(-06) | 143.4 |
| | 1/256 | 4.10(-05) | 5.25(-07) | 78.1 | 3.61(-05) | 1.26(-07) | 285.8 |
| | 1/512 | 1.02(-05) | 1.45(-07) | 70.5 | 8.99(-06) | 1.57(-08) | 570.5 |

Table 2: Performance of the two versions for the problem (4.1) for different values of $\lambda$; case $\mu = 0$.

| | | $c_2 = 3/4$ | | | $c_2 = 2/3$ | | |
|---|---|---|---|---|---|---|---|
| | $h$ | $rerr^S$ | $rerr^R$ | $if$ | $rerr^S$ | $rerr^R$ | $if$ |
| $\lambda = -1$ | 1/64 | 1.86(-05) | 1.97(-06) | 9.4 | 1.76(-05) | 2.76(-08) | 637.0 |
| | 1/128 | 4.62(-06) | 4.95(-07) | 9.3 | 4.37(-06) | 3.44(-09) | 1268.8 |
| | 1/256 | 1.15(-06) | 1.24(-07) | 9.3 | 1.09(-06) | 4.30(-10) | 2532.1 |
| $\lambda = -2$ | 1/128 | 6.69(-05) | 4.57(-06) | 14.6 | 6.36(-05) | 5.67(-08) | 1121.4 |
| | 1/256 | 1.66(-05) | 1.15(-06) | 14.5 | 1.58(-05) | 7.08(-09) | 2234.7 |
| | 1/512 | 4.15(-06) | 2.88(-07) | 14.4 | 3.94(-06) | 8.84(-10) | 4461.0 |
| $\lambda = -4$ | 1/128 | 8.12(-04) | 7.98(-05) | 10.2 | 7.95(-04) | 1.80(-06) | 441.7 |
| | 1/256 | 2.01(-04) | 2.01(-05) | 10.0 | 1.96(-04) | 2.24(-07) | 876.5 |
| | 1/512 | 4.99(-05) | 5.04(-06) | 9.9 | 4.88(-05) | 2.80(-08) | 1746.3 |

Table 3: Performance of the two versions for the problem (4.2) for different values of $\lambda$; case $\mu = \lambda$.

| | | $c_2 = 3/4$ | | | $c_2 = 2/3$ | | |
|---|---|---|---|---|---|---|---|
| | $h$ | $rerr^S$ | $rerr^R$ | $if$ | $rerr^S$ | $rerr^R$ | $if$ |
| $\lambda = -1$ | 1/64 | 2.64(-05) | 1.53(-06) | 17.2 | 2.28(-05) | 9.00(-08) | 253.5 |
| | 1/128 | 6.55(-06) | 3.91(-07) | 16.7 | 5.67(-06) | 1.12(-08) | 504.6 |
| | 1/256 | 1.63(-06) | 9.90(-08) | 16.5 | 1.41(-06) | 1.41(-09) | 1006.6 |
| $\lambda = -2$ | 1/128 | 2.61(-05) | 7.12(-07) | 36.6 | 2.27(-05) | 8.42(-08) | 269.8 |
| | 1/256 | 6.48(-06) | 1.89(-07) | 34.3 | 5.65(-06) | 1.05(-08) | 537.5 |
| | 1/512 | 1.62(-06) | 4.87(-08) | 33.2 | 1.41(-06) | 1.31(-09) | 1073.0 |
| $\lambda = -4$ | 1/128 | 1.01(-04) | 9.29(-07) | 108.9 | 8.88(-05) | 6.27(-07) | 141.8 |
| | 1/256 | 2.50(-05) | 3.15(-07) | 79.4 | 2.20(-05) | 7.80(-08) | 282.0 |
| | 1/512 | 6.22(-06) | 8.94(-08) | 69.6 | 5.47(-06) | 9.72(-09) | 562.6 |

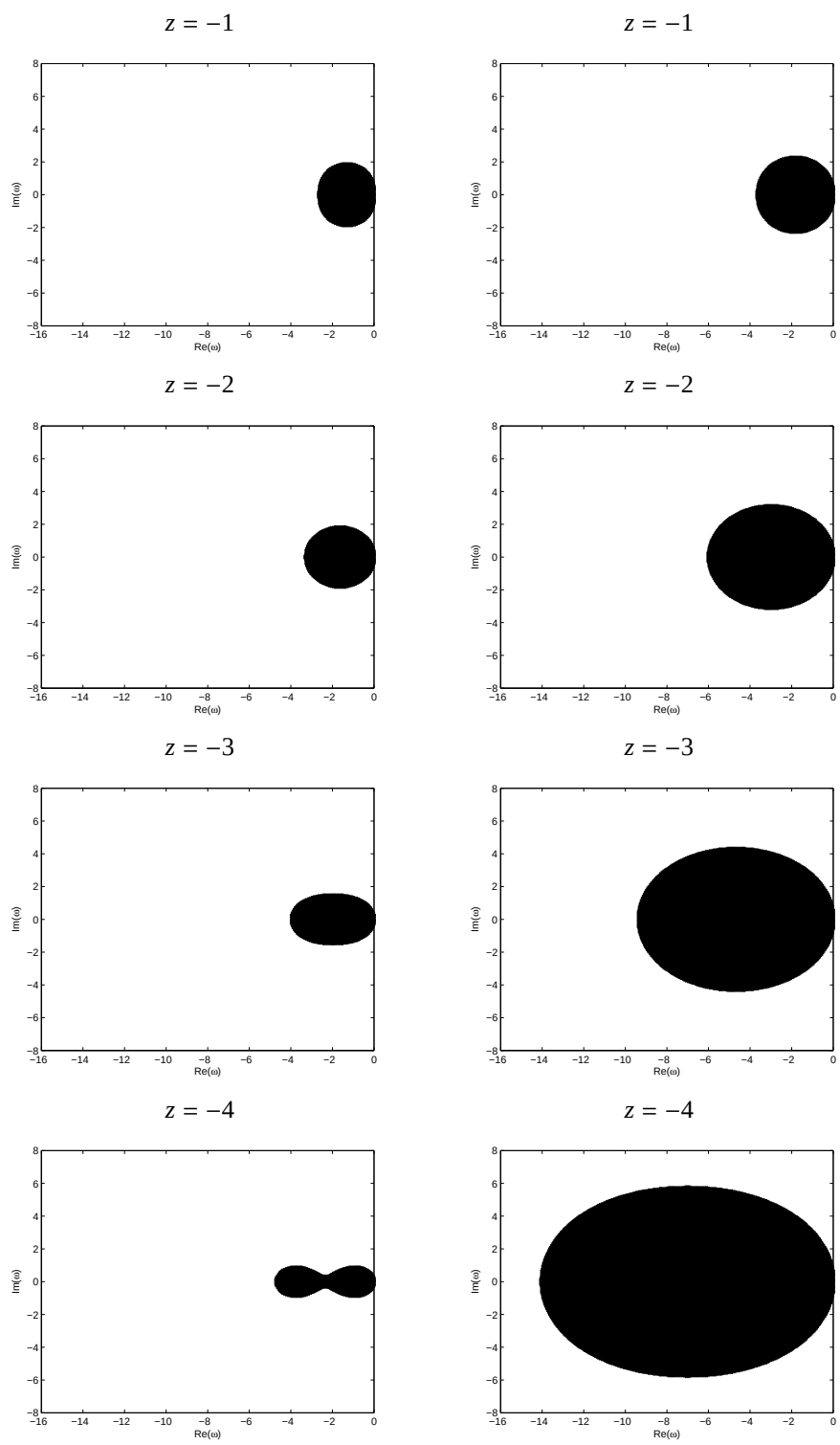


Figure 1: Sections through the stability region by plane $z = const$ for fixed $c_2 = \frac{3}{4}$: standard version (left column), revised version (right column).

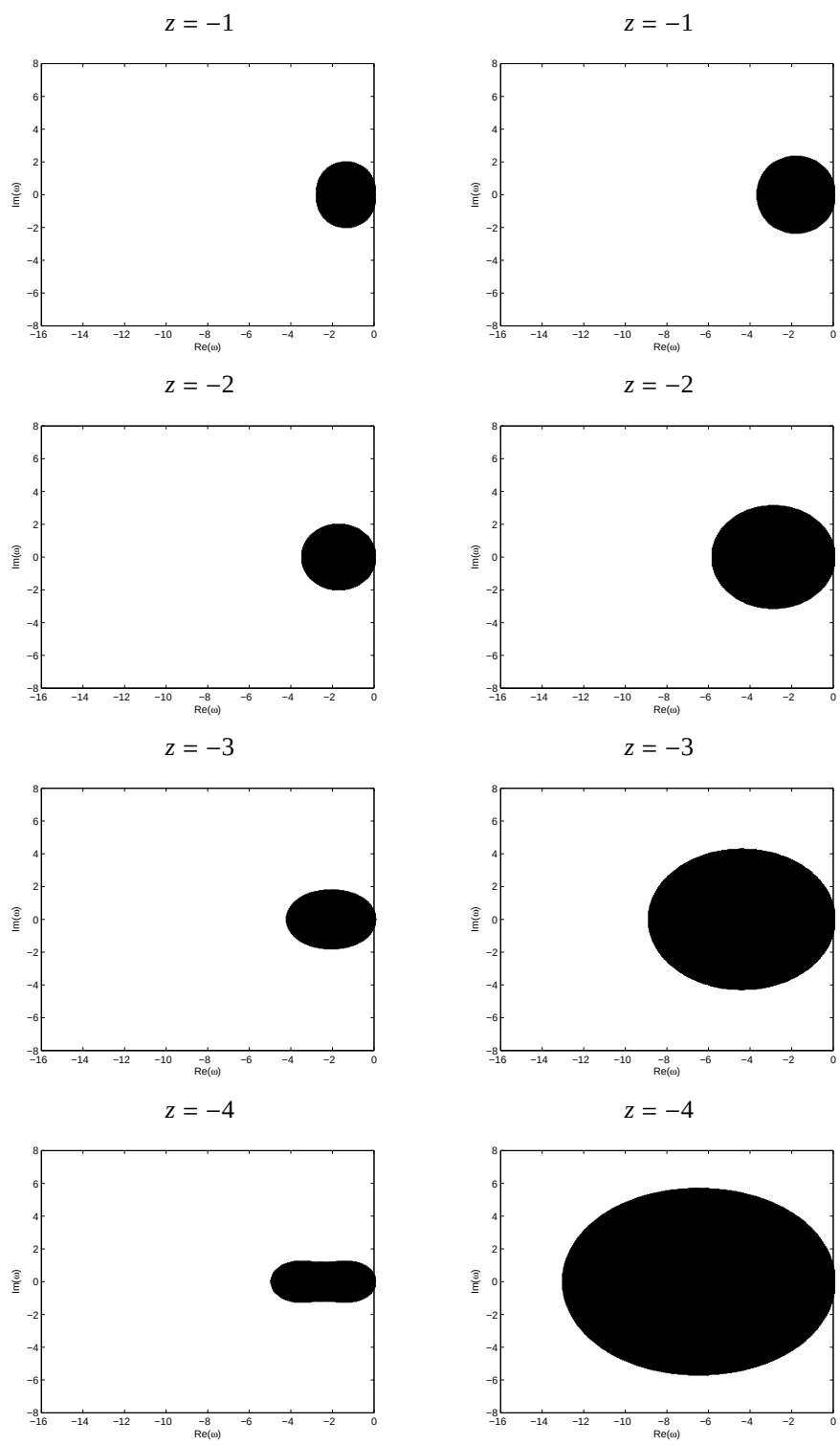


Figure 2: Sections through the stability region by plane $z = const$ for fixed $c_2 = \frac{2}{3}$: standard version (left column), revised version (right column).

Table 4: Performance of the two versions for the problem (4.2) for different values of $\lambda$; case $\mu = 0$.

| | | $c_2 = 3/4$ | | | $c_2 = 2/3$ | | |
|---|---|---|---|---|---|---|---|
| | $h$ | $rerr^S$ | $rerr^R$ | $if$ | $rerr^S$ | $rerr^R$ | $if$ |
| $\lambda = -1$ | 1/64 | 2.81(-05) | 1.21(-06) | 23.1 | 2.52(-05) | 3.11(-08) | 812.0 |
| | 1/128 | 6.95(-06) | 3.06(-07) | 22.7 | 6.25(-06) | 3.86(-09) | 1619.1 |
| | 1/256 | 1.73(-06) | 7.68(-08) | 22.5 | 1.56(-06) | 4.81(-10) | 3233.1 |
| $\lambda = -2$ | 1/128 | 5.76(-05) | 3.53(-06) | 16.3 | 5.44(-05) | 5.36(-08) | 1015.0 |
| | 1/256 | 1.43(-05) | 8.86(-07) | 16.2 | 1.35(-05) | 6.71(-09) | 2015.4 |
| | 1/512 | 3.57(-06) | 2.22(-07) | 16.1 | 3.37(-06) | 8.39(-10) | 4016.2 |
| $\lambda = -4$ | 1/128 | 5.85(-04) | 5.21(-05) | 11.2 | 5.68(-04) | 1.34(-06) | 424.8 |
| | 1/256 | 1.44(-04) | 1.31(-05) | 11.0 | 1.40(-04) | 1.67(-07) | 839.1 |
| | 1/512 | 3.59(-05) | 3.29(-06) | 10.9 | 3.49(-05) | 2.09(-08) | 1668.0 |

The second test case is the nonlinear problem

$$\begin{cases} y'(x) = \dfrac{\lambda y^2(x) + 2x^3 e^{2\lambda x}}{y(x)} \\ y(1) = e^{\lambda}, \end{cases} \tag{4.2}$$

with $x \in [x_{min} = 1, x_{max} = 5]$, whose exact solution is $y(x) = x^2 e^{\lambda x}$. The results are collected in Table 3 and Table 4. Once again, the new version is clearly much more accurate than the standard one.

We conclude this section presenting the numerical results obtained for the nonlinear system

$$\begin{cases} y_1'(x) = 3\,(y_2(x) - x) + \dfrac{\lambda y_1^2(x)}{x^3 e^{\lambda x}}, \\ y_2'(x) = \dfrac{y_2(x)\left(x^2 + 2y_1(x) + \lambda x^2 y_2(x) - \lambda x^3\right)}{x^3(1 + xe^{\lambda x})}, \\ y_1(1) = e^{\lambda},\ y_2(1) = 1 + e^{\lambda}, \end{cases} \tag{4.3}$$

for $x \in [x_{min} = 1, x_{max} = 2]$, whose exact solution is $y(x) = [y_1(x); y_2(x)]^T = [x^3 e^{\lambda x}; x(1 + xe^{\lambda x})]^T$. For this case $rerr$ is the biggest from the relative errors at $x_{max}$ in the two components.
The results for $\lambda = -1, -2, -4$ are collected in Tables 5 and 6. The accuracy gain with the revised version is visible.

## 5. Concluding remarks

We have revised the standard technique for the construction of ef Runge-Kutta methods for the numerical solution of first order initial value problems (1.1). For the particular case of two-stage methods we have focused our attention on the contribution in the final stage of the error coming from the second internal stage, and also experimentally compared the revised ef version with the standard ef one. The results clearly confirm the superiority of the revised version. Based on this it makes sense to investigate how the idea developed in this paper on the two-stage Runge-Kutta method may be applied on Runge-Kutta methods with an increased number of stages and/or on other nonlinear methods.

Moreover, the simple expression of the local truncation error of the revised version can be useful in order to furnish new reliable estimations of the parameter $\mu$ on which depend the coefficients of the method. In fact, up to now, a mathematical theory for the exact determination of this parameter has not yet been developed. The usual technique used in literature consists in estimating $\mu$ in such a way that the local error of the resulting methods is minimized. Therefore, it may be interesting to examine the behaviour of the estimations for the revisited ef methods, in order to understand if the improvement on the local error in the external stage (which now takes also into account the contribution of the internal stages) is also inherited by the parameter estimators.

Table 5: Performance of the two versions for the problem (4.3) for different values of $\lambda$; case $\mu = \lambda$.

| | | $c_2 = 3/4$ | | | $c_2 = 2/3$ | | |
|---|---|---|---|---|---|---|---|
| | $h$ | $rerr^S$ | $rerr^R$ | $if$ | $rerr^S$ | $rerr^R$ | $if$ |
| $\lambda = -1$ | 1/ 128 | 0.285(-05) | 0.216(-06) | 13.2 | 0.679(-05) | 0.342(-07) | 198.8 |
| | 1/ 256 | 0.716(-06) | 0.532(-07) | 13.5 | 0.171(-05) | 0.427(-08) | 399.8 |
| | 1/ 512 | 0.180(-06) | 0.132(-07) | 13.6 | 0.428(-06) | 0.534(-09) | 801.7 |
| | 1/1024 | 0.449(-07) | 0.329(-08) | 13.7 | 0.107(-06) | 0.667(-10) | 1605.6 |
| $\lambda = -2$ | 1/ 128 | 0.715(-04) | 0.114(-04) | 6.3 | 0.546(-05) | 0.422(-06) | 12.9 |
| | 1/ 256 | 0.180(-04) | 0.279(-05) | 6.4 | 0.142(-05) | 0.525(-07) | 27.0 |
| | 1/ 512 | 0.450(-05) | 0.690(-06) | 6.5 | 0.362(-06) | 0.654(-08) | 55.3 |
| | 1/1024 | 0.113(-05) | 0.172(-06) | 6.6 | 0.913(-07) | 0.817(-09) | 111.7 |
| $\lambda = -4$ | 1/ 128 | 0.954(-03) | 0.847(-03) | 1.1 | 0.182(-03) | 0.182(-04) | 10.0 |
| | 1/ 256 | 0.249(-03) | 0.210(-03) | 1.2 | 0.479(-04) | 0.223(-05) | 21.4 |
| | 1/ 512 | 0.638(-04) | 0.523(-04) | 1.2 | 0.123(-04) | 0.276(-06) | 44.4 |
| | 1/1024 | 0.161(-04) | 0.131(-04) | 1.2 | 0.310(-05) | 0.343(-07) | 90.3 |

Table 6: Performance of the two versions for the problem (4.3) for different values of $\lambda$; case $\mu = 0$.

| | | $c_2 = 3/4$ | | | $c_2 = 2/3$ | | |
|---|---|---|---|---|---|---|---|
| | $h$ | $rerr^S$ | $rerr^R$ | $if$ | $rerr^S$ | $rerr^R$ | $if$ |
| $\lambda = -1$ | 1/ 128 | 0.703(-05) | 0.813(-06) | 8.6 | 0.492(-06) | 0.361(-08) | 136.4 |
| | 1/ 256 | 0.176(-05) | 0.203(-06) | 8.7 | 0.125(-06) | 0.452(-09) | 277.1 |
| | 1/ 512 | 0.441(-06) | 0.508(-07) | 8.7 | 0.316(-07) | 0.565(-10) | 558.6 |
| | 1/1024 | 0.110(-06) | 0.127(-07) | 8.7 | 0.792(-08) | 0.706(-11) | 1121.5 |
| $\lambda = -2$ | 1/ 128 | 0.212(-04) | 0.173(-05) | 12.3 | 0.434(-05) | 0.108(-07) | 401.1 |
| | 1/ 256 | 0.532(-05) | 0.433(-06) | 12.3 | 0.109(-05) | 0.135(-08) | 803.2 |
| | 1/ 512 | 0.133(-05) | 0.108(-06) | 12.3 | 0.272(-06) | 0.169(-09) | 1607.3 |
| | 1/1024 | 0.334(-06) | 0.271(-07) | 12.3 | 0.682(-07) | 0.212(-10) | 3215.6 |
| $\lambda = -4$ | 1/ 128 | 0.964(-04) | 0.303(-05) | 31.8 | 0.310(-05) | 0.799(-07) | 38.8 |
| | 1/ 256 | 0.242(-04) | 0.765(-06) | 31.7 | 0.790(-06) | 0.983(-08) | 80.3 |
| | 1/ 512 | 0.607(-05) | 0.192(-06) | 31.6 | 0.199(-06) | 0.122(-08) | 163.4 |
| | 1/1024 | 0.152(-05) | 0.482(-07) | 31.5 | 0.500(-07) | 0.152(-09) | 329.3 |

## References


[1] J. C. Butcher, Numerical methods for Ordinary Differential Equations, Second Edition (Wiley, Chichester, 2008).
[2] J. P. Coleman, L. Gr. Ixaru, IMA J. Numer. Anal. 16/2 (1996) 179.
[3] R. D'Ambrosio, Highly Stable Multistage Numerical Methods for Functional Equations: Theory and Implementation Issues, Ph.D. thesis (University of Salerno, 2010).
[4] J. M. Franco, J. Comput. Appl. Math. 167/1 (2004) 1.
[5] L. Gr. Ixaru, Comput. Phys. Commun. 105/1 (1997) 1.
[6] L. Gr. Ixaru, G. Vanden Berghe, Exponential Fitting (Kluwer Academic Publishers, Dordrecht, 2004).
[7] J. D. Lambert, Numerical methods for ordinary differential systems: The initial value problem (Wiley, Chichester, 1991).
[8] T. E. Simos, Comput. Phys. Comm. 115/1 (1998) 1.
[9] G. Vanden Berghe, H. De Meyer, M. Van Daele, T. Van Hecke, J. Comput. Appl. Math. 125/1-2 (2000) 107.
[10] G. Vanden Berghe, H. De Meyer, M. Van Daele, T. Van Hecke, Comput. Phys. Comm. 123/1-3 (1999) 7.
[11] H. Van De Vyver, J. Comput. Appl. Math. 188/2 (2006) 309.