CANADIAN APPLIED
MATHEMATICS QUARTERLY
Volume 17, Number 3, Fall 2009

ON FIFTH AND SIXTH ORDER EXPLICIT RUNGE-KUTTA METHODS: ORDER CONDITIONS AND ORDER BARRIERS

J. C. BUTCHER

ABSTRACT. Although Runge-Kutta methods up to order 4 satisfy exactly the same conditions in the case of a single scalar equation as for a general high-dimensional system, the two order theories start to diverge above this order. For example, for order 5, two of the 17 "elementary differentials" from which the Taylor expansions for both the exact and numerical solutions are constructed, coincide and this means that there are only 16 conditions for order 5 in the case of a scalar problem. A method will be presented which exhibits different order behaviours for scalar and high-dimensional problems. The paper will also give a new proof of the first of the Runge-Kutta order barriers.

1 Introduction Since the publication of the famous papers by Runge [**10**], Heun [**6**] and Kutta [**8**], explicit Runge-Kutta methods have been widely used in the solution of initial value problems. Even in the early days of these methods, the conditions for order seemed to have been thoroughly understood and methods of order 4 were successfully used to solve both single equations and systems of differential equations. Although fifth order methods were introduced by Kutta, there were errors in the presentation of his results, partly corrected by Nyström [**9**]. Deriving explicit fifth order methods is now an easy matter (see, for example, [**5**]).

The first explicit method of order six was found by Professor Anton Huťa of the University of Bratislava [**7**]. For orders as high as this, analysing the Taylor expansions of the exact solution and the numerical approximation, in the traditional way, is a formidable task. Some individual equations in Huťa's paper occupied more than two pages.

It is interesting that, as a starting point, it was assumed that eight stages would be necessary. Huťa made this choice because there are 31 order conditions *in the scalar case* and eight stages would make 36 free parameters available. If Professor Huťa had known that there were really 37 conditions, he undoubtedly would have decided to use nine stages, because that would provide 45 free parameters. Remarkably, the six extra conditions he didn't know about, are actually satisfied by his method [**2**]. Thus the method is sixth order for a system and not just for a single scalar equation. Even more remarkable, sixth order methods can be found with only 7 stages (providing only 28 free parameters to satisfy 37 conditions) (see, for example, [**5**]).

Building on these brief historical notes, it can be said that, above order 4, complications arise that at one time would have seemed surprising. Specifically, it is known that the conditions for order 5 or more, are *not* the same for a single equation and for a system. Furthermore, although an s stage method can have order s when $s \leq 4$, above this number of stages, this is impossible. A new proof of this result will be given.

The structure of the paper is as follows. In Section 2 an outline will be given of the tree structure of the order conditions for Runge-Kutta methods. Following this, in Section 3, attention will be focussed on the scalar case, including an explanation of why this differs from the vector case. Taking this further, we consider in Section 4, the numerical behaviour of a method whose order depends on whether the problem being solved is scalar, or a system of dimension greater than one.

Finally, in Section 5, a new proof will be given of an order barrier for orders greater than 4.

2 The order of Runge-Kutta methods Even though the classical Runge-Kutta methods of Runge, Heun and Kutta give satisfactory results on most problems, they are not adequate for solving problems to great accuracy because they would be too expensive. Although high order methods generally perform more computations in each time step, the error decreases very rapidly as the stepsize is reduced. In this section we will introduce the basic nomenclature and tools required to analyse the order conditions of Runge-Kutta methods.

In an s stage explicit Runge-Kutta method, denote the stage values by Y_1, Y_2, $\ldots$, Y_s, for example in the initial step from x_0 to $x_0 + h$, where Y_i is a first order approximation to $y(x_0 + hc_i)$. That is, $Y_i = y(x_0 + hc_i) + O(h^2)$ The stage derivatives F_1, F_2, $\ldots$, F_s are defined by

$$F_i = f(x_0 + hc_i, Y_i), \qquad i = 1, 2, \ldots, s,$$

so that $F_i = y'(x_0 + hc_i) + O(h^2)$, if $f(x, y)$ is Lipschitz continuous with respect to y.

For an arbitrary (possibly implicit) method, each stage value is calculated from the initial value and the s stage derivatives, using the equation

$$Y_i = y_0 + h\sum_{j=1}^{s} a_{ij}F_j, \qquad i = 1, 2, \ldots, s,$$

but throughout this paper we will consider only *explicit* methods, for which the summation is only for $j \le i-1$.

When we calculate the stage value Y_i from $Y_i = y_0 + h\sum_{j=1}^{i-1} a_{ij}F_j$, it can be shown that $Y_i = y\big(x_0 + h\sum_{j=1}^{i-1} a_{ij}\big) + O(h^2)$. Hence, in the formula $F_i = f(x_0 + hc_i, Y_i)$, we can always assume that $c_i = \sum_{j=1}^{i-1} a_{ij}$ because if one of the components of $y(x)$ is forced to always equal x by requiring its derivative to always equal 1, then we want agreement between these values to carry over to numerical approximations. After we have calculated the stage derivatives, we can use these to obtain an approximation to the solution at the end of the step

$$y_1 = y_0 + h\sum_{i=1}^{s} b_iF_i.$$

To represent any particular Runge–Kutta method, the coefficients defining the method can be written in a tableau:

$$\begin{array}{c|ccccc}
0 & & & & & \\
c_2 & a_{21} & & & & \\
c_3 & a_{31} & a_{32} & & & \\
\vdots & \vdots & \vdots & \ddots & & \\
c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & \\
\hline
 & b_1 & b_2 & \cdots & b_{s-1} & b_s\,.
\end{array}$$

As an example, the classical fourth-order method of Kutta has the following tableau, where the actual formulae for the various quantities used in the computation are also shown.

$$\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & \frac{1}{2} & & & \\
\frac{1}{2} & 0 & \frac{1}{2} & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
\end{array}
\qquad
\begin{aligned}
Y_1 &= y_0, \\
Y_2 &= y_0 + \tfrac{1}{2}hF_1, \\
Y_3 &= y_0 \phantom{+ \tfrac{1}{2}hF_1} + \tfrac{1}{2}hF_2, \\
Y_4 &= y_0 \phantom{+ \tfrac{1}{2}hF_1 + \tfrac{1}{2}hF_2} + hF_3, \\
y_1 &= y_0 + \tfrac{1}{6}hF_1 + \tfrac{1}{3}hF_2 + \tfrac{1}{3}hF_3 + \tfrac{1}{6}hF_4.
\end{aligned}$$

The conditions for a Runge-Kutta method to have an order p can be found by finding two Taylor expansions: (a) The Taylor expansion of the exact solution about the initial point (x_0, y_0) and (b) The Taylor expansion of the result computed by the Runge–Kutta method after a single step. Once these Taylor expansions have been found, they can be compared term by term to yield the order conditions.

A method has order p if these two Taylor series agree up to the h^p terms. As a first step, we will calculate the first few derivatives of the solution y assuming that the problem is autonomous:

$$y'(x) = f(y(x)), \qquad f : \mathbb{R}^N \to \mathbb{R}^N. \tag{1}$$

Since $y(x)$ lies in $\mathbb{R}^N$, we can rewrite (1) in the form

$$y_i'(x) = f_i(y_1(x), y_2(x), \ldots, y_N(x)), \qquad i = 1, 2, \ldots, N,$$

and we can calculate y_i'' using the chain rule

$$y_i'' = \sum_{j=1}^{N} \frac{\partial f_i}{\partial y_j} y_j' = \sum_{j=1}^{N} \frac{\partial f_i}{\partial y_j} f_j.$$

This can be written compactly as

$$y''(x) = \mathbf{f}'\mathbf{f}$$

where $\mathbf{f}$ is the vector $f(y(x))$ and $\mathbf{f}'$ is the linear operator defined by the matrix of partial derivatives.

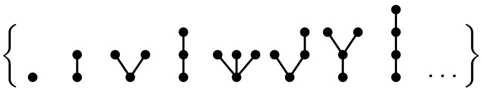
Introduce the bi-linear operator $\mathbf{f}''$ of second partial derivatives and the third derivative of y is found to be

$$y'''(x) = \mathbf{f}''(\mathbf{f}, \mathbf{f}) + \mathbf{f}'\mathbf{f}'\mathbf{f}.$$

Expressions like $\mathbf{f}$, $\mathbf{f}'\mathbf{f}$, $\mathbf{f}''(\mathbf{f}, \mathbf{f})$, $\mathbf{f}'\mathbf{f}'\mathbf{f}$ are known as "elementary differentials." They are related to rooted trees by the very simple rule that $\mathbf{f}^{(m)}$ is represented by a parent of m children. For example,

$\mathbf{f}$	is represented by	,
$\mathbf{f}'\mathbf{f}$	is represented by	,
$\mathbf{f}''(\mathbf{f}, \mathbf{f})$	is represented by	,
$\mathbf{f}'\mathbf{f}'\mathbf{f}$	is represented by	.

Let T denote the set of all (rooted) trees:

$T =$ 

For each $t \in T$, the corresponding elementary differential, built up from $\mathbf{f}, \mathbf{f}', \mathbf{f}'', \dots,$ is written $\mathcal{F}(t)(y(x))$. In addition to this function on T, define $r(t)$ to be the "order" of t (the number of vertices in t), $\sigma(t)$ to be the symmetry of t, and $\gamma(t)$ the density of t. Detailed information on these functions is given in [**5**] and values, up to trees up to order four, are given in Table 1. The elementary weights, to be introduced below, are also included in addition to the elementary differentials.

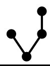
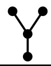
t								
$r(t)$	1	2	3	3	4	4	4	4
$\sigma(t)$	1	1	2	1	6	1	2	1
$\gamma(t)$	1	2	3	6	4	8	12	24
$\mathcal{F}(t)$	$\mathbf{f}$	$\mathbf{f}'\mathbf{f}$	$\mathbf{f}''(\mathbf{f},\mathbf{f})$	$\mathbf{f}'\mathbf{f}'\mathbf{f}$	$\mathbf{f}'''(\mathbf{f},\mathbf{f},\mathbf{f})$	$\mathbf{f}''(\mathbf{f},\mathbf{f}'\mathbf{f})$	$\mathbf{f}'\mathbf{f}''(\mathbf{f},\mathbf{f})$	$\mathbf{f}'\mathbf{f}'\mathbf{f}'\mathbf{f}$
$\Phi(t)$	$\sum b_i$	$\sum b_i c_i$	$\sum b_i c_i^2$	$\sum b_i a_{ij} c_j$	$\sum b_i c_i^3$	$\sum b_i c_i a_{ij} c_j$	$\sum b_i a_{ij} c_j^2$	$\sum b_i a_{ij} a_{jk} c_k$

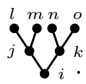
TABLE 1: Various quantities which depend on trees.

Theorem 1. *The formal Taylor series for* $y(x_0 + h)$ *is*

$$y(x_0 + h) = y_0 + \sum_{t \in T} \frac{h^{r(t)}}{\sigma(t)\gamma(t)} \mathcal{F}(t)(y_0).$$

Our aim now is to find a similar expression for the Taylor expansion for the *computed* solution. Given a Runge–Kutta method defined by the trio (A, b^T, c), define for each tree t the "elementary weight" $\Phi(t)$.

We will illustrate this by a single example

$t =$ 

.

For this tree, the formula for the elementary weight is

$$\Phi(t) = \sum_{i,j,k,l,m,n,o=1}^{s} b_i\, a_{ij}\, a_{jl}\, a_{jm}\, a_{ik}\, a_{kn}\, a_{ko}, \tag{2}$$

where we see that the subscript on b_i is attached to the root of t and the subscripts attached to factors such as a_{ij} correspond to labels at the end of an edge. Because $\sum_l a_{jl} = c_j$, and similarly for summations over m, n and o, (2) can be written as

$$\Phi(t) = \sum_{i,j,k=1}^{s} b_i\, a_{ij}\, c_j^2\, a_{ik}\, c_k^2.$$

Theorem 2. *The formal Taylor series for the approximate solution y_1 is*

$$y_1 = y_0 + \sum_{t \in T} \frac{h^{r(t)}}{\sigma(t)}\, \Phi(t)\mathcal{F}(t)(y_0).$$

Theorem 3. *A Runge-Kutta method has order p if and only if*

$$\Phi(t) = \frac{1}{\gamma(t)} \quad \textit{for } r(t) \le p.$$

We see this result by comparing the coefficients of $h^{r(t)}\mathcal{F}(t)(y_0)$ in

$$\frac{h^{r(t)}}{\sigma(t)}\, \Phi(t)\mathcal{F}(t)(y_0) \quad \text{and} \quad \frac{h^{r(t)}}{\sigma(t)\gamma(t)}\, \mathcal{F}(t)(y_0).$$

3 The order of methods for scalar problems Early theories of Runge-Kutta methods analysed order conditions using the non-autonomous scalar problem

$$y' = f(x, y). \tag{3}$$

To link the two versions of the theory, write (3) as a two dimensional autonomous problem $\mathbf{y}' = \mathbf{f}(\mathbf{y})$ as follows:

$$\mathbf{y} = \begin{bmatrix} y \\ x \end{bmatrix}, \qquad \mathbf{f} = \begin{bmatrix} f(x, y) \\ 1 \end{bmatrix}.$$

We now note that

$$\mathbf{f}'\mathbf{f} = \begin{bmatrix} f_y f + f_x \\ 0 \end{bmatrix},$$

and that the second component of $\mathbf{f}^{(m)}$ is the zero m-linear operator, for $m = 1, 2, \ldots$.

Hence, if two elementary differentials can be written in the forms $\mathbf{XYf'f}$ and $\mathbf{YXf'f}$, where $\mathbf{X}$ and $\mathbf{Y}$ are linear operators then, for the scalar problem, their values are equal.

For example, in the case $\mathbf{X} = \mathbf{A}$, $\mathbf{Y} = \mathbf{B}$, where $\mathbf{A}$ and $\mathbf{B}$ are given below, we have

$$\mathbf{XYf'f} = \begin{bmatrix} f_y & f_x \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_{yy}f + f_{xy} & f_{xy}f + f_{xx} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_y f + f_x \\ 0 \end{bmatrix},$$

$$\mathbf{YXf'f} = \begin{bmatrix} f_{yy}f + f_{xy} & f_{xy}f + f_{xx} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_y & f_x \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_y(f_y f + f_x) \\ 0 \end{bmatrix},$$

and these have identical values

$$\begin{bmatrix} f_y(f_y f + f_x)(f_{yy}f + f_{xy}) \\ 0 \end{bmatrix}.$$

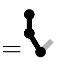
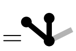
We will identify 5 linear operators relevant to questions on methods of orders 5 and 6, together with 3 vectors which have the property of having second component zero in the scalar case. In the case of the vectors, these can be represented by elementary differentials, or equivalently by trees. In the case of the four linear operators they can be represented also as trees stumps. The five operators are

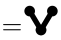
$$\mathbf{A} = \mathbf{f}', \quad \mathbf{B} = \mathbf{f}''(\mathbf{f}, \cdot), \quad \mathbf{C} = \mathbf{f}'\mathbf{f}'(\cdot), \quad \mathbf{D} = \mathbf{f}''(\mathbf{f}'\mathbf{f}, \cdot), \quad \mathbf{E} = \mathbf{f}'''(\mathbf{f}, \mathbf{f}, \cdot),$$

and the vectors are

$$\mathbf{u} = \mathbf{f}'\mathbf{f}, \qquad \mathbf{v} = \mathbf{f}'\mathbf{f}'\mathbf{f}, \qquad \mathbf{w} = \mathbf{f}''(\mathbf{f}, \mathbf{f}).$$

The five linear and bilinear operators and the three special vectors correspond to tree stumps and trees as follows:

$\mathbf{A} =$ , $\mathbf{B} =$ , $\mathbf{C} =$ , $\mathbf{D} =$ , $\mathbf{E} =$ 

$\mathbf{u} =$ , $\mathbf{v} =$ , $\mathbf{w} =$ 

To explain the loss of an order condition in the scalar case of order 5, we see that the identification of the elementary differentials represented by $\mathbf{ABu}$ and $\mathbf{BAu}$ correspond to the order conditions $\Phi(t_1) = 1/\gamma(t_1)$ and $\Phi(t_2) = 1/\gamma(t_2)$ being replaced by the combined condition

$$\Phi(t_1) + \Phi(t_2) = \frac{1}{\gamma(t_1)} + \frac{1}{\gamma(t_2)},$$

where

$t_1 =$ , $t_2 =$ .

In the case of order 6, there is a loss of 5 conditions, when we move from the vector to the scalar case. These correspond to the identification of three pairs of elementary differentials

$$\mathbf{ADu} \equiv \mathbf{DAu}, \qquad \mathbf{AEu} \equiv \mathbf{EAu} \qquad \text{and} \qquad \mathbf{ABw} \equiv \mathbf{BAw},$$

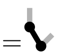
together with a triple of coincident elementary differentials

$$\mathbf{ABAu} \equiv \mathbf{BAAu} \equiv \mathbf{AABu}, \tag{4}$$

where $\equiv$ denotes that two elementary differentials are equal in the scalar case, but not necessarily in the vector case. The result given by (4) can also be seen from

$$\mathbf{ABv} \equiv \mathbf{BAv} = \mathbf{BCu} \equiv \mathbf{CBu},$$

where $\mathbf{BAv} = \mathbf{BCu}$ means that these elementary differentials are identical even for the vector case. Although we will not discuss the seventh order case in detail, we will introduce a bi-linear operator and its tree-stump representation, which are relevant to that order

$\mathbf{Z} = \mathbf{f}''(\mathbf{f}(\cdot), \cdot) =$ .

For the seventh order we have a coincidence of trees based on the non-symmetric nature of $\mathbf{X}$. That is, for the scalar scalar case, we have

$$\mathbf{Z}(\mathbf{u}, \mathbf{w}) \equiv \mathbf{Z}(\mathbf{w}, \mathbf{u}).$$

To summarise, the number of conditions for orders up to 6 are as follows, in the scalar and vector cases, respectively:

p	1	2	3	4	5	6
Scalar	1	2	4	8	16	31
Vector	1	2	4	8	17	37

4 A method of ambiguous order Because fifth order methods are required to satisfy 17 conditions for an arbitrary vector-valued problem but only 16 for a scalar problem, we might well ask

Do Runge-Kutta methods exist which have order 5 for a scalar problem but only order 4 for a vector problem?

The answer is "yes" and here is an example of such a method.

$$
\begin{array}{c|cccccc}
0 \\
\frac{1}{3} & \frac{1}{3} \\
\frac{3}{4} & -\frac{123}{256} & \frac{315}{256} \\
\frac{1}{5} & \frac{193}{750} & -\frac{189}{1250} & \frac{176}{1875} \\
\frac{2}{3} & -\frac{26}{81} & \frac{7}{15} & -\frac{304}{4455} & \frac{175}{297} \\
1 & \frac{151}{150} & -\frac{351}{250} & \frac{304}{4125} & -\frac{5}{77} & \frac{243}{175} \\
\hline
 & \frac{1}{24} & 0 & 0 & \frac{125}{336} & \frac{27}{56} & \frac{5}{48}
\end{array} \quad . \tag{5}
$$

For the scalar case, the following equation needs to be satisfied

$$\left(\sum b_i\, c_i\, a_{ij}\, a_{jk}\, c_k - \frac{1}{30}\right) + \left(\sum b_i\, a_{ij}\, c_j\, a_{jk}\, c_k - \frac{1}{40}\right) = 0,$$

but for the vector case, the two terms must be equal to zero separately.

For method (5), the remaining 15 of the order conditions are satisfied and, in addition,

$$\sum b_i\, c_i\, a_{ij}\, a_{jk}\, c_k = \frac{1}{30} - \frac{1}{1920},$$
$$\sum b_i\, a_{ij}\, c_j\, a_{jk}\, c_k = \frac{1}{40} + \frac{1}{1920}.$$

showing that the method has order 5 for a scalar problem but only order 4 in the general vector problem.

Each of the following problems was solved using this method.

Problem 1

$$y' = -y + \exp(\sin(x)) + \cos(x)\exp(\sin(x)), \qquad y(0) = 1.$$

Problem 2

$$\begin{aligned} y_1' &= -y_1 + y_3 + \log(y_2)y_3, & y_1(0) &= 1,\\ y_2' &= -\log(y_3)y_2, & y_2(0) &= \exp(1),\\ y_3' &= \log(y_2)y_3, & y_3(0) &= 1. \end{aligned}$$

with the results calculated for $x = 0.5$, using $n = 5, 10, 20, 40, 80$ steps. Note that these two problems are closely related in that the solution to Problem 1 is identical with the solution for y_1 in problem 2. The results are shown in Table 2 showing for each n the values of the errors $\exp(\sin(0.5))$ minus the computed approximation. Note that for Problem 2, only y_1 is given in the table since this approximates the same exact result as for Problem 1. Also shown are the error ratios between successive approximations. These confirm our expectation that the errors are asymptotically proportional to h^5 for Problem 1 and to h^4 for Problem 2.

n	Problem 1	Ratio	Problem 2	Ratio
5	4.73022×10^{-9}		6.96863×10^{-8}	
		32.531		19.258
10	1.45405×10^{-10}		3.61847×10^{-9}	
		32.311		18.221
20	4.50018×10^{-12}		1.98585×10^{-10}	
		31.967		17.306
40	1.40776×10^{-13}		1.14748×10^{-11}	
		31.700		16.681
80	4.44089×10^{-15}		6.87894×10^{-13}	

TABLE 2: Test results for a method of ambiguous order.

5 Order barriers The order condition related to the tree $[\tau^{j-1}[_\ell \tau^m]_\ell]$ is

$$b^T C^{j-1} A^l C^{m-1} c = \frac{m!}{(j+\ell+m)(\ell+m)!},$$

where we have written $C = \mathrm{diag}(c)$. Also write $\widehat{C} = \mathrm{diag}(\widehat{c})$, with $\widehat{c}_i = 1 - c_i$, $i = 1, 2, \ldots, s$.

Lemma 4. *If a Runge-Kutta method has order not less than $k+\ell+m$, then*

$$b^T \widehat{C}^{k-1} A^l C^{m-1} c = \frac{(k-1)!m!}{(k+\ell+m)!}.$$

Proof. By writing $\widehat{C}^{k-1}$ in terms of powers of C by the binomial theorem, we have

$$\begin{aligned} b^T \widehat{C}^{k-1} A^l C^{m-1} c &= \frac{m!}{(\ell+m)!} \sum_{j=1}^{k} \binom{k-1}{j-1} \frac{(-1)^{j-1}}{(j+\ell+m)} \\ &= \frac{m!}{(\ell+m)!} \int_0^1 (1-x)^{k-1} x^{\ell+m}\, dx \\ &= \frac{m!}{(\ell+m)!} \frac{(k-1)!(\ell+m)!}{(k+\ell+m)!} \\ &= \frac{(k-1)!m!}{(k+\ell+m)!}. \end{aligned}$$

□

This result will be used in a new proof of an order barrier.

Theorem 5. *There do not exist s stage methods with order s if $s \geq 5$.*

Proof. By using Lemma 4 several times we evaluate and refactorize the following matrix product

$$\begin{bmatrix} b^T A^{s-3} \\ b^T(\widehat{C} - \widehat{c}_s I) A^{s-4} \\ b^T(\widehat{C} - \widehat{c}_s I)(\widehat{C} - \widehat{c}_{s-1} I) A^{s-5} \end{bmatrix} \begin{bmatrix} Ac & (C - c_2 I)c \end{bmatrix} \tag{6}$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 1 & -\widehat{c}_s & 0 \\ 2 & -\widehat{c}_s - \widehat{c}_{s-1} & \widehat{c}_s\widehat{c}_{s-1} \end{bmatrix} \begin{bmatrix} \frac{1}{s!} & \frac{1}{(s-1)!} \\ \frac{1}{(s-1)!} & \frac{1}{(s-2)!} \\ \frac{1}{(s-2)!} & \frac{1}{(s-3)!} \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 0 & -c_2 \end{bmatrix}.$$

If u^T is a typical row of the first factor on the left-hand side of (6), and v is a typical column in the second factor, then $u_i = 0$ unless $i \le 3$. Similarly $v_i = 0$ unless $i \ge 3$. Hence the product $u^T v$ is equal to $u_3 v_3$. This implies that the matrix product has rank 1. Hence, one of the matrices

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & -\widehat{c}_s & 0 \\ 2 & -\widehat{c}_s - \widehat{c}_{s-1} & \widehat{c}_s\widehat{c}_{s-1} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1 & 2 \\ 0 & -c_2 \end{bmatrix}$$

has rank only 1.

It is impossible that $c_2 = 0$ since this would imply that $b^T A^{s-2} c = 0$, rather than the correct value $1/s!$. Hence $\widehat{c}_s = \widehat{c}_{s-1} = 0$, implying that $b^T \widehat{C} A^{s-3} c = 0$, rather than the correct value $1/s!$.

If $s^*(p)$ denotes the minimum number of stages for a method to have order p, then for orders 5 and 6, Theorem 5 tells us that $s^*(p) \ge p+1$ for these cases. From the existence of known methods, it is true that $s^*(p) = p+1$ for these values of p. A more detailed investigation shows that for $s^*(7) = 9$ [**3**], and $s^*(p) \ge p+3$ for $p \ge 8$ [**4**]. □

Acknowledgement The author is grateful to a number of referees whose valuable comments and suggestions have led to a much improved paper.

REFERENCES

1. J. C. Butcher, *Coefficients for the study of Runge-Kutta integration processes,* J. Austral. Math. Soc. **3** (1963), 185–201.
2. J. C. Butcher, *On the integration processes of A. Huťa,* J. Austral. Math. Soc. **3** (1963), 202–206.
3. J. C. Butcher, *On the attainable order of Runge-Kutta methods*, Math. Comp. **19** (1965), 408–417.
4. J. C. Butcher, *The non-existence of ten stage eighth order explicit Runge-Kutta methods*, BIT **25** (1985), 521–540.

5. J. C. Butcher, *Numerical Methods for Ordinary Differential Equations,* 2nd ed., Wileys, 2008.
6. K. Heun, *Neue Methoden zur approximativen Integration der Differentialgleichungen einer unabhängigen Veränderlichen,* Z. Math. Phys. **45** (1900), 23–38.
7. A. Huťa, *Une amélioration de la méthode de Runge-Kutta-Nyström pour la résolution numérique des équations différentielles du premier ordre*, Acta Fac. Nat. Univ. Comenian. Math. **1** (1956), 201–224.
8. W. Kutta, *Beitrag zur näherungsweisen Integration totaler Differentialgleichungen,* Z. Math. Phys. **46** (1901), 435–453.
9. E. J. Nyström, *Über die numerische Integration von Differentialgleichungen,* Acta Soc. Sci. Fennicae **50**(13) (1925), 55pp.
10. C. Runge, *Über die numerische Auflösung von Differentialgleichungen,* Math. Ann. **46** (1895), 167–178.

Department of Mathematics, University of Auckland,
Auckland 1142, New Zealand.
E-mail address: butcher@math.auckland.ac.nz