Continuous Explicit Runge-Kutta Methods with Applications to Ordinary and Delay Differential Equations

Brynjulf Owren
Abstract

When solving non-stiff ordinary or delay differential equations one might sometimes wish to use continuous explicit Runge-Kutta methods (CERK methods) which differ from discrete methods in having polynomial instead of scalar weights. Traditionally these methods are constructed by supplying a discrete method with a continuous extension, often called an interpolant. One may also construct the continuous method directly and obtain an underlying discrete method as a by-product. To this end one defines the order of the method in a uniform sense, giving rise to a new meaning of the term order barrier which is the minimal number of stages necessary to obtain a method of a given order. Theorems related to such order barriers are derived, and the barriers are determined for order 3,4 and 5. This leads to the discovery of 5th order CERK methods with 8 stages. In particular, methods with a reusable stage are considered and their close relation to $C^1$-interpolants is revealed. There exists no discrete embedded error estimation method of higher order than the CERK method.

An idea by Jeltsch and Nevanlinna on the stability of explicit Runge-Kutta methods is further developed. The theory of positive functions is used to provide bounds for the largest possible disk to be inscribed in the stability region of an explicit Runge-Kutta method. In particular it is shown that the closed disk $|\zeta + r| \leq r$ can be contained in the stability region of an explicit $s$-stage Runge-Kutta method of order two if and only if $r \leq s - 1$.

The implementation of CERK methods can be done with the classical strategies for error estimation and step size selection. Some data structures intended for storing the requisite data are suggested, an absolutely necessary feature for the code to handle delay differential equations. Some numerical results are shown using a 5th order CERK method with 8 stages. These experiments confirm the expected theoretical properties of the CERK method, but says nothing about the how well these methods perform when applied to real world problems.
Acknowledgements

For most students, a doctoral program is a time of devotion to intellectual and professional goals which would be unattainable without the assistance of many other people. My debt in this regard is larger than most, not only because I frequently sought help, but also because those I went to gave so freely of their talents.

I have learned a lot from my thesis advisor Syvert P. Nørsett over these years, not only mathematics. He made me realize the importance of keeping the balance between being selfcritical and to believe in oneself, which I think is the basic key to progress for any scientist. Thanks to him I was introduced to many fine workers in the field, and twice he sent me to Trieste, Italy to work with Marino Zennaro and Alfredo Bellen. And several times Zennaro came to Trondheim as well. It was by one such occasion that the main ideas behind this thesis were born. Zennaro made the initial studies on order barriers for continuous explicit Runge-Kutta methods, and without his contributions most of the contents of Chapter 2 would never have been written. Equally important was the collaboration with my good friend from the early school days, Kristian Seip. Some loose talk on the way home from a football match ended with the stability results for Runge-Kutta methods of Chapter 3.

Over the past few years I have also benefited from discussions with Hans Munthe-Kaas, Harald Simonsen, Arieh Iserles, Finn Faye Knudsen, Anne Kværnø, Lawrence Shampine, Juan Montijano and the referees of our paper for the London conference on Computational ODEs 1989.
Contents

1 Introduction .................................................. 1
   1.1 The equations we solve. ............................. 4
   1.2 Numerical solution of delay differential equations. .... 6

2 Continuous explicit Runge-Kutta methods ..................... 9
   2.1 A motivating example - DOPRI(5,4) ................... 10
   2.2 Order barriers ......................................... 14
       2.2.1 Some general order results ..................... 14
       2.2.2 Finding CEN(p) for $p \leq 5$ ................. 24
   2.3 Practical questions ..................................... 39
       2.3.1 Some general results .......................... 42
       2.3.2 Stage reuse for optimal CERK methods of order $p=3-5$
            and monotonicity of the error function ....... 43

3 Stability of explicit Runge-Kutta methods .................... 55
   3.1 The largest disk of stability ....................... 55
       3.1.1 Introduction ................................ 55
       3.1.2 Preliminary results .......................... 57
       3.1.3 Some stability theorems ..................... 58
3.1.4 Numerical results .................................. 61
3.2 P-stability properties of CERK methods .......... 64

4 Implementation of CERK Methods ................. 69
  4.1 Local error estimate, step size selection .......... 69
  4.2 Datastructures implemented in C ................. 71
  4.3 Some numerical results ............................ 74

5 Concluding remarks .................................. 81

A A remark on an error polynomial of some fifth order continuous extensions of Dormand-Prince(5,4) 85

B The real zeros of polynomials related to the generalized Bessel polynomials 89

Bibliography ........................................... 95
Chapter 1

Introduction

Initial value problems (IVPs) for ordinary (ODEs) and delay differential equations (DDEs) occur frequently in applications. The huge number of ODEs used in modelling of dynamic systems are well-known. But also the latter kind of mathematical model can be found in various fields of engineering and physics. There are many good examples, for instance in control theory where there typically may be a delay in the output measurements of the process (see e.g. [4]). Another example is from population dynamics [42] where the growth rates of various competing species may depend on the population of the preceding generation. Then the size of the delay may be several years. Finally we consider in some detail an application from geophysics that is studied by several authors, e.g. Owren et al. [61]. In marine seismic experiments one sometimes uses arrays of airguns to generate oscillating air bubbles that emit pressure waves throughout the water. The motion and emitted pressure wave from such a bubble can be well approximated by a system of ODEs [61]. However, it turns out that the bubbles interact with each other such that the pressure wave emitted from a single airgun is not the same as it would be in the presence of other airguns. And clearly, if two airguns fire simultaneously at a distance, say $x$ meters apart, they will not “notice” one another before the pressure wave from one airgun reaches the other. Hence there is a delay in the interaction, $\tau = x/c$ where $c$ is the velocity of sound in water. Now putting $n$ airguns together one will in general obtain a system with $s = n(n - 1)/2$ delays, corresponding to the
edges in a graph with \( n \) vertices. The hydrodynamics of the water close to
the bubble is discussed by Kirkwood and Bethe [53]. They show that the
emitted pressure wave depends on the time derivative of the enthalpy at the
bubble wall. Thus, the interaction between the airguns also depends on the
derivative of the solution. Such equations are called neutral DDEs. A code,
MOD3DS, solves these equations numerically for an arbitrary number of air-
guns by using the code DEABM along with linear interpolation to compute
the interactions. This method for DDEs turns out to be of order 1 due to
the inaccurate approximation of the interactions. As we shall see DDEs are
often solved numerically by modifying a method for ODEs. The most
important ingredient of this adaptation is to extend the discrete solution in a
way such that a sufficiently accurate approximation can be obtained at any
point prior to the current point of integration at reasonable computational
cost.

Most of the traditional methods for ODEs have been designed in order to
furnish the solution at a discrete set of points. But there are applications,
also for ODEs, where a continuous approximation to the solution is use-
ful. These include problems where dense output is required, e.g. data for
graphics, and problems where discontinuities are present. Enright [33, 36]
considers some new error control strategies by using a continuous approxi-
mation to the solution.

Many methods have such a facility naturally available, like some linear mul-
tistep methods that yield an interpolant with no additional cost. But their
poor ability to handle equations with discontinuities, and the fact that they
cannot combine high order of accuracy with good stability properties, are
serious problems that have to be taken into account. Collocation methods
represent another example. One may then use the collocation polynomial as
a continuous approximation to obtain accurate and stable continuous meth-
ods. Unfortunately, they are usually expensive. These methods are known
to be a subclass of the celebrated Runge-Kutta methods. In general, such
continuous one-step methods have recently been investigated by a number
of authors like Bellen and Zennaro[15], Enright et al. [34], Horn[46], Nørsett
and Wanner [58], Shampine[69], Calvo et al.[21] and Zennaro [74, 76, 77].

The major part of this thesis is devoted to the study of explicit Runge-
Kutta methods with continuous extensions. It appears that one can either construct a continuous one-step method directly, giving a discrete method as a by-product, or one can extend an already existing discrete method, possibly by doing some additional function evaluations. The two strategies both have their pros and cons. The latter approach allows one to choose a well-known underlying discrete method that is known to possess the desired qualities. And if the continuous extension is required only occasionally, as might be the case when it is used for handling discontinuities, the computational cost will practically be that of the discrete method. We illustrate this approach in Section 2.1 by supplying the Dormand-Prince (5,4) pair with a family of fifth order continuous extensions at the cost of two extra function evaluations. In the remaining parts of Chapter 2 we consider in a systematic way the general concept of continuous explicit Runge-Kutta methods (CERK-methods), keeping in mind that the method we want to obtain shall yield a continuous approximation. The traditional discrete methods can then be viewed as an underlying part of our methods.

The stability properties of these methods are considered in Chapter 3. When one constructs a method with a given order of consistency, one is usually allowed to choose some of the coefficients freely. Using these coefficients, one might wish to optimize the stability properties of the method somehow. We discuss the stability polynomials with corresponding stability regions that contains the largest possible disk. This case is relatively simple to analyze, and for a general purpose ODE-solver it represents a trade-off between stretching the stability region in the real and imaginary direction. We also touch the stability problems of our methods when applied to DDEs.

In Chapter 4 we discuss the implementation of these continuous explicit Runge-Kutta methods, taking advantage of the advanced data types available in C. The numerical results of this section are included merely to indicate that the qualitative behaviour of our new CERK methods is in accordance with what one might expect from their theoretical properties. Although it would indeed be interesting, it is not the intention of this work to show that some particular method behaves well for real world problems. It will be the subject of future research to find good methods by optimizing their error constants stability regions and other parameters related to their implementation.
1.1 The equations we solve.

Since we want to apply CERK methods both to ordinary and delay differential equations, we briefly discuss properties like the existence, uniqueness, stability and smoothness of their solutions.

By an ordinary differential equation (ODE) we shall mean a system

\[ \frac{dy(x)}{dx} = f(x, y(x)), \quad y(x_0) = y_0 \quad (1.1) \]

where \( f : R \times R^m \to R^m \) defines \( m \) generally non-linear equations, and \( y_0 \) is an \( m \)-vector called the initial point. The existence of a unique solution is ensured if \( f \) is continuous and satisfies a Lipschitz condition in its second argument. For further details see for instance [23]. It is also well-known that the smoothness of the solution of (1.1) is roughly inherited from that of \( f \). Since we shall only consider linear stability, we recall that the test equation \( y' = \lambda y, \quad \lambda \in C \) is stable whenever \( \text{Re}(\lambda) \leq 0 \). We shall assume that the problems we consider are non-stiff. Lambert [55, p. 231] relates stiffness of linear systems of ODEs to the ratio between the largest and smallest real part of the eigenvalues of the system. However, his definition is controversial. More recent definitions relates the concept of stiffness to the numerical method which is used.

Now consider the class of delay differential equations (DDE) given by

\[ \frac{dy(x)}{dx} = f(x, y(x), y(x - \tau)), \quad y(x) = \phi(x), \quad x_0 - \tau \leq x \leq x_0 \quad (1.2) \]

where we have introduced the constant delay parameter \( \tau > 0 \) and \( \phi(x) \) is a given continuous function called the initial function. We will assume that \( \phi(x_0) = y(x_0 + 0) \). One may easily generalize (1.2) for instance by letting \( \tau \) depend on \( x \) or even on the solution itself, i.e. \( \tau := \tau(x, y(x)) \). One may also introduce several delays denoted \( \tau_1, \ldots, \tau_s \) such that \( f \) becomes a function of \( s + 2 \) arguments. In any case, we shall always require that a unique solution exists. The sufficient conditions for this can be found in [32, p. 23]. We also mention the problem class \textit{neutral} DDEs in which the argument \( y'(x - \tau) \) is present in \( f \).
1.1. The equations we solve.

It turns out that the question of smoothness of the solution of DDEs is not as simple as it is in the ODE case. Considering (1.2) the solution obviously may have a discontinuous derivative at \( x = x_0 \) even if the function \( f \) is analytic in all its arguments. This irregularity is inherited by the delay argument at \( x = x_0 + \tau \) causing a jump in the second derivative of the solution at \( x = x_0 + \tau \). By induction it becomes evident that the solution generally has a jump in the \((k+1)\)th derivative at \( x = x_0 + k\tau, \ k \geq 0 \). These points are called breaking points and they will also occur for the more general class of DDEs. For neutral DDEs the situation is even worse since one can then expect a jump in \( y' \) at all breaking points.

Also in the case of delay equations we consider linear stability. We shall be interested in the equation

\[
y'(x) = ay(x) + by(x - \tau), \ a, b \in C. \tag{1.3}
\]

Zennaro [75] has proved that the set \( S_{DDE} \) of pairs \((a, b) \in C \times C\) such that (1.3) is asymptotically stable for all \( \tau \) and for all initial functions \( g \) is given by

\[
S_{DDE} = S'_{DDE} \cup B_{DDE},
\]

where \( S'_{DDE} = \{(a, b) \in C \times C : \Re(a) < 0 \text{ and } |b| < -\Re(a)\} \) and \( B_{DDE} = \{(a, b) \in C \times C : a \in R, |b| = -a, a + b \neq 0\} \). We confine ourselves to non-stiff DDEs. Roth [67] defines stiff DDEs as the cases where the roots of the characteristic quasipolynomial (see [32]) related to (1.3) has all its roots far into the left half-plane and he proves that the only way a DDE can become stiff is by introducing a stiff ODE-component after Lambert’s definition.

For a discussion of the linear test equation for neutral DDEs and the corresponding stability criterion, see [14].
1.2 Numerical solution of delay differential equations.

The interest in the numerical solution of DDEs has risen considerably since Cryer presented his survey [25] in 1972. These recent developments are divided into four types of methods by Bellen [10]:

1. Transformation of DDEs into ODEs (Krasowski [54] developed by Banks and Burns [5, 6]).

2. Iterative Methods considered in [8, 12, 73].

3. Collatz-type methods, see [24] and


The last type of methods is the one where most work has been done since Cryer [25] 1972. Surveying the literature we find that almost any ODE-method has been employed with suitable adaptations to solve DDEs. Multistep methods were first considered by Zverkina [78, 79, 80] and later by Bock and Schlöder [16, 17], Cryer [26], VanderHouwen and Sommeijer [47], McKee [56] and Tavernini [71]. As for one-step methods some authors have considered extensions of the most popular explicit Runge-Kutta methods like Runge-Kutta-Fehlberg adapted for DDEs by Oberle and Pesch [59] and Oppelstrup [60] and Runge-Kutta-Merson studied by Neves [57]. One-step collocation and implicit Runge-Kutta methods are considered by Bellen and Zennaro [9, 11, 13] and extrapolation methods by DeGee [28].

The idea of all these adaptations is to supply a discrete method with a suitable continuous extension, say $u(x)$, and replace the DDE (1.2) with

$$y'(x) = f(x, y(x), v(x - \tau)), \quad v(x) = \begin{cases} \phi(x), & x_0 - \tau \leq x \leq x_0 \\ u(x), & x_0 < x \end{cases} \quad (1.4)$$

One could then say that the choice of a strategy for solving (1.2) consists of the following stages

- Select a suitable discrete method for ODEs.
- Find a continuous extension or endow the method to furnish the solution at the required extra-nodal points.
- Analyze the consequences of replacing (1.2) with (1.4).

As can be seen from the quotations above, the discrete method has usually been one that has proved successful for the solution of ODEs. Various continuous approximations have been used. From the early days the most popular strategy was interpolation over several nodal points as is natural for multistep methods and that has been applied with some success by Oberle and Pesch [59] and Oppelstrup [60] to Runge-Kutta methods using Hermite interpolation.

In the literature quoted above one will find that the last issue is the one that has been emphasized. Three questions immediately arise: Under what conditions (on \( u(x) \)) will the order of the ODE-method applied to (1.4) be retained when (1.2) is replaced by (1.4)? How does the delay approximation affect stability? How shall the method cope with breaking points? Zennaro [76] considers Runge-Kutta methods for DDEs and finds that when certain restrictions are imposed on the mesh, it is sufficient to approximate the delay term by means of a natural continuous extension that may have an order of accuracy which is less than the method itself. However, this constrained mesh is not feasible for the more general DDEs as e.g. the case of several delays. Oppelstrup finds that the interpolation error should generally be of order \( q \geq p + 1 \) for a method of order \( p \). This is in accordance with the continuous methods discussed in Chapter 2.

The question of stability of the method is more complicated than for ODEs. Surveying the literature we find that even the meaning of linear stability is vague. And it is probably impossible to find a test equation of which the solution resembles the behaviour of any member from the large variety of DDE-problems. We shall only briefly discuss the concept of P-stability in Section 3.2 using the framework and theory developed by Barwell [7] and Zennaro [75]. When the delay \( \tau \) of (1.2) becomes smaller than the steplength one may have problems with the adaptation strategy (1.4). This may happen if the delay \( \tau = \tau(x) = 0 \) for some \( x \) (this is sometimes denoted 'the singular
case'). The problem can be solved by extrapolation, but as pointed out by Oppelstrup[60] this affects stability.

To avoid loss of accuracy at the breaking points any algorithm for DDEs should include a device for handling discontinuities. Many of the presented codes require the user to give the breaking points before or during integration, or worse, they do not pay any attention to them at all (for instance the DOPRI54-method implemented by SUBROUTINE RETARD [42, p.450]). In some cases, like when the delays are state dependent i.e. \( \tau = \tau(y(x)) \) it is not even possible to specify the breaking points in advance. Various authors have considered automatic detection and localization of discontinuities, we mention Gear and Østerby [40] and Enright et al. [35].
Chapter 2

Continuous explicit
Runge-Kutta methods

Consider continuous explicit Runge-Kutta methods of the form

\[ K_i = f(x_0 + c_i h, y_0 + h \sum_{j=1}^{i-1} a_{ij} K_j), \quad i = 1, \ldots, s \]  \hspace{1cm} (2.1a)

\[ u(x_0 + \theta h) = y_0 + h \sum_{i=1}^{s} b_i(\theta) K_i, \quad \theta \in [0, 1] \]  \hspace{1cm} (2.1b)

\( u(x_0 + \theta h) \) is a continuous approximation to \( y(x) \) in the interval \( [x_0, x_0 + h] \) and \( b_i(\theta) \) \( i = 1, \ldots, s \) are polynomials of degree \( \leq d \) where \( d \) is a positive integer. We shall also require \( c_i = \sum_{j=1}^{i-1} a_{ij} \), and \( b_i(0) = 0 \) for \( i = 1, \ldots, s \). Remark that \( c_1 = 0 \) which implies that the first stage reduces to \( K_1 = f(x_0, y_0) \). Moreover, the coefficients \( a_{ij} \) define a strictly lower triangular \( s \times s \)-matrix \( A \). Observe that a conventional discrete Runge-Kutta method is obtained from (2.1b) simply by setting \( y_1 := u(x_0 + h) \). Although we shall primarily be interested in the local approximation on an interval \( [x_0, x_0 + h] \) it should be pointed out that if the integration is proceeded by means of this underlying discrete method one obtains a globally continuous approximation. Although this is natural, for instance Horn [46] considers continuous extensions of the Runge-Kutta-Fehlberg (4,5) pair that do not possess this property.
We define the *uniform order* (which we shall simply refer to as the *order*) as the greatest integer $p$ for which

$$\max_{0 \leq \theta \leq 1} |y(x_0 + \theta h) - u(x_0 + \theta h)| = O(h^{p+1}) \quad (2.2)$$

Here $|\cdot|$ stands for any norm on $\mathbb{R}^n$. Notice that we require the same local order of accuracy in the interior of the step as at the end point ($\theta = 1$). Again, if the value at the end point is used to proceed integration for a problem of the kind (1.1) the global order of accuracy will roughly be one less than the local order. So one could argue that it would be sufficient that the uniform order be one less than the order at the end point. However, for the case of delay equations one must generally approximate the retarded argument to the same order as the underlying discrete method for the global order to be maintained at the nodes.

It is well-known that efficiency, intended as the ratio between the computational effort and the accuracy of the computed approximations is an important parameter to be considered when designing new numerical methods. Thus, the main goal of Section 2.2 is to determine the order barriers for CERK methods. Most of this material is taken from Owren and Zennaro [63, 64]. In Section 2.3 we discuss some questions related to the implementation of CERK methods.

### 2.1 A motivating example - DOPRI(5,4)

We consider the embedded pairs of explicit Runge-Kutta methods suggested by Dormand and Prince [29]. It is given in the Butcher tableau of Table 2.1. Many authors have developed continuous extensions for this pair, among them are Shampine [69, 70] and Calvo et al. [21]. Recently Higham [43] has developed some extensions with globally continuous second derivatives.

According to [76, Theorem 5] we can expect existence of an interpolant of order $d$ between 3 and 5 for the Dormand Prince method. In [42] a fourth
2.1. A motivating example - DOPRI(5,4)

\[
\begin{array}{c|cccc}
0 & 1/5 & 1/5 & 3/40 & 9/40 \\
1/5 & 44/45 & -3/5 & 32/9 & 8/9 \\
3/10 & 19372/6561 & -25360/6561 & 6448/2197 & 32/2197 \\
4/5 & 5005/512 & 56/9 & 363/35 & -6656/6561 \\
8/9 & 125/384 & 0 & 500/1113 & 125/2197 \\
1 & 384/256 & 0 & 500/1113 & 125/2197 \\
\hline
y_1 & 35/384 & 0 & 500/1113 & 125/2197 \\
\hat{y}_1 & 5179/57600 & 0 & 757/16656 & 390/6446 \\
\end{array}
\]

Table 2.1: Butcher tableau for the Dormand-Prince(5,4) method. The method \( y_1 \) is the one with order 5.

order continuous method based on the first six stages of Table 2.1 is given.

\[
u_0(x_n + \theta h) = y_n + \sum_{i=1}^{6} b_i(\theta) K_i
\]  \hspace{1cm} (2.3)

with

\[
\begin{align*}
b_1(\theta) &= \theta - \frac{1337}{480} \theta^2 + \frac{1039}{360} \theta^3 - \frac{1103}{1152} \theta^4 \\
b_2(\theta) &= 0 \\
b_3(\theta) &= \frac{4216}{1113} \theta^2 - \frac{18728}{3339} \theta^3 + \frac{7580}{3339} \theta^4 \\
b_4(\theta) &= -\frac{27}{16} \theta^2 + \frac{9}{2} \theta^3 - \frac{415}{192} \theta^4 \\
b_5(\theta) &= -\frac{2187}{8480} \theta^2 + \frac{2673}{2120} \theta^3 - \frac{6991}{6784} \theta^4 \\
b_6(\theta) &= \frac{32}{35} \theta^2 - \frac{319}{105} \theta^3 + \frac{187}{84} \theta^4 
\end{align*}
\]

This is in fact the only fourth order interpolant for the method of Table 2.1 that uses only the first six stages and it is easy to see that if the last stage
is included one may obtain a 3-parameter family of such interpolants. It is also easy to prove (see e.g. [62]) that there does not exist any fifth order interpolant without making any additional function evaluations. Shampine [70] and Calvo et al. [21] construct fifth order continuous extensions by adding two stages. Their idea is the following: Add one stage in order to obtain a fifth order approximation $y_{n+a}$ at some point $x_n + \alpha h$. This can be done by examining the order conditions where the step size $h$ is replaced by $\alpha h$. Then construct a fifth order Hermite interpolant through $x_n, x_n + \alpha h$ and $x_n + h$. This requires the function evaluation, $f(x_n + \alpha h, y_{n+a})$ which represents another additional stage. We shall derive a class of fifth order interpolants by using an idea of Enright et al. [34]. Rather than examining the order conditions we

- Choose the 4th order interpolant above denoted by $u_0(x_0 + \theta h)$.
- Pick two points $c_8, c_9 \in [0,1]$.
- Construct a 5th degree polynomial $u_1(x_0 + \theta h)$ satisfying the following conditions

$$
\begin{align*}
  u_1(x_0) &= y_0, & u_1'(x_0) &= k_1 \\
  u_1(x_0 + h) &= y_1, & u_1'(x_0 + h) &= k_7 \\
  u_1'(x_0 + c_8 h) &= f(x_0 + c_8 h, u_0(x_0 + c_8 h)) = k_8 \\
  u_1'(x_0 + c_9 h) &= f(x_0 + c_9 h, u_0(x_0 + c_9 h)) = k_9
\end{align*}
$$

(2.5)

Notice that these interpolants also require two additional stages. In order to compute the coefficients of the 5th order polynomial $u_1$, we define a third degree polynomial $p_3(x_0 + \theta h)$ satisfying the first 4 conditions of (2.5)

$$
p_3(x_0 + \theta h) = (\theta - 1)^2(1 + 2\theta)y_0 + \theta^2(3 - 2\theta)y_1 \\
+ \theta(\theta - 1)^2hk_1 + \theta^2(\theta - 1)hk_7.
$$

(2.6)

For $u_1$ to obey the same conditions and be of degree 5, we must have

$$
  u_1(x_0 + \theta h) - p_3(x_0 + \theta h) = \theta^2(\theta - 1)^2(\alpha\theta + \beta)
$$

(2.7)
where $\alpha$ and $\beta$ are determined by the two remaining conditions. Note that $u_1$ is a Hermite Birkhoff interpolant, and will not exist for all $c_8$ and $c_9$. We have

**Proposition 2.1** $u_1(x_0 + \theta h)$ defines a fifth order CERK method for all $c_8, c_9$ satisfying

$$c_8, c_9 \neq 0, \quad c_8, c_9 \neq 1, \quad c_8 \neq c_9, \quad c_8 \neq \frac{c_8 - \frac{1}{2}}{c_9 - 1} \quad (2.8)$$

**Proof:** The existence of the polynomial $u_1$ follows by differentiating (2.6) and (2.7) to obtain a $2 \times 2$ linear system for $\alpha$ and $\beta$ from the last two conditions of (2.5). The determinant of this system is non-zero if (2.8) is satisfied. The remaining part is a simple consequence of the discussion in Enright et al. [34]. □

Instead of giving the expressions for $\alpha$ and $\beta$ we present a canonical form of $u_1$ where we have replaced $c_8$ and $c_9$ by $u$ and $v$ for the convenience of the reader.

$$u_1(\theta) = \phi_0(\theta) y_0 + \phi_1(\theta) y_1 + \psi_0(\theta) h k_1 + \psi_4(\theta) h k_9 + \psi_6(\theta) h k_9 + \psi_1(\theta) h k_7$$

with

$$\begin{align*}
\phi_0(\theta) & = (\theta - 1)^2 (1 + 2\theta) + \frac{2}{5}\xi(\theta - 1)^2 \theta^2 (4\theta - 5(u + \hat{v})) \\
\phi_1(\theta) & = \theta^2 (3 - 2\theta) - \frac{2}{5}\xi(\theta - 1)^2 \theta^2 (4\theta - 5(u + \hat{v})) \\
\psi_0(\theta) & = \theta(\theta - 1)^2 [1 + \frac{2\theta}{u(\theta - 1)}((3v - 1)u + \frac{1}{2} - v))]\theta \\
& \quad + \frac{1}{2}(u(\theta + \hat{v})(1 - 3v) + \hat{v}v)] \\
\psi_1(\theta) & = \theta^2 (\theta - 1)[1 + \frac{2\theta}{u(\theta - 1)}((3v - 2)u + \frac{3}{2} - 2v))]\theta \\
& \quad + \frac{1}{2}(u(2 - 3v)(u + \hat{v} - 1) + 2\hat{v}(v - 1)) \\
\psi_4(\theta) & = \theta^2 (\theta - 1)^2 \frac{\xi(\theta - 1)}{u(\theta - 1)(\theta - 1)}\frac{2}{5}(v - \frac{1}{2})\theta - \frac{1}{2}\hat{v}v) \\
\psi_6(\theta) & = -\theta^2 (\theta - 1)^2 \frac{\xi(\theta - 1)}{u(\theta - 1)(\theta - 1)}(\frac{2}{5}(u - \frac{1}{2})\theta - \frac{1}{2}\hat{u}u) \\
\end{align*}$$
where
\[ \xi = [(2v - 1)u - \hat{v}]^{-1} \]

and
\[ \hat{v} = v - \frac{2}{5} \]
\[ \hat{u} = u - \frac{3}{5} \]

The obvious question is now, how shall we choose \( c_8 \) and \( c_9 \) (\( u \) and \( v \))? Since the extra stages do not influence the way integration proceeds (at least as long as one sticks to ODEs) we do not have to consider stability. It is not obvious in what sense the error should be minimized and we postpone this discussion until Section 2.3. We discuss the error constant of the interpolant \( u_1 \) in Appendix A and we prove that when using a max-norm the size of the error function becomes practically independent of \( c_8 \) and \( c_9 \). Some test runs indicate that \( c_8 = 0.2 \) and \( c_9 = 0.5 \) can be recommended. For further details see [62].

2.2 Order barriers

2.2.1 Some general order results

In this section we shall use the theory developed by Butcher [18, 19] extensively without giving specific references as we shall rely on the reader’s acquaintance with trees, order conditions and related topics. We recommend the books by Butcher [20] and Hairer et al. [42] as background material, and we will use the notation of the latter.

The first part is devoted to providing theorems which can be used to determine lower bounds for the numbers
\[ CEN(p) := \min_{m(s) \in M_p} s \]
2.2. Order barriers

where \( m(s) \) is a CERK method with \( s \) stages, and \( M_p \) is the set of all CERK methods with order \( p \). The numbers \( CEN(p) \) are similar to the famous Butcher barriers \( EN(p) \) for the discrete case.

It is well-known that for implicit Runge-Kutta methods the minimal number of stages, say \( N(p) \) and \( CN(p) \), necessary to get order \( p \) for the discrete and continuous case respectively, are easy to find in general, and are attained by collocation methods. They are

\[
N(p) = \lfloor (p + 1)/2 \rfloor \quad \text{and} \quad CN(p) = p.
\]

For CERK methods things are, like for the discrete case, a lot more complicated. One has the obvious result

\[
CEN(p) \geq EN(p),
\]

and from the literature referred above, one can extract the following bounds

\[
\begin{align*}
CEN(1) &= 1, \quad CEN(2) = 2, \quad 3 \leq CEN(3) \leq 4, \\
5 \leq CEN(4) \leq 6, \quad 6 \leq CEN(5) \leq 9,
\end{align*}
\]

where the upper bounds are determined by known CERK methods. Although we solve the problem completely up to \( p = 5 \), we will not be able to derive a general formula for \( CEN(p) \), and we suspect that, like for \( EN(p) \), this is a very hard task.

It is easy to see that in order to fulfill (2.2), the degree \( d \) of the polynomials \( b_i(\theta) \) must satisfy \( d \geq p \). On the other hand, allowing \( d > p \) can lead to approximate solutions \( u(x) \) which derivatives are unbounded as \( h \to 0 \) (see [58]). Therefore we always choose \( d = p \) such that, according to [76, theorem 5], the polynomials \( b_i(\theta) \) span the space \( \Pi_{p-1} \) of polynomials of degree \( p - 1 \). With reference to (2.1a-b) it is necessary that the number \( s^* \) of distinct \( c_i \)’s satisfies

\[
s^* \geq p.
\]
Consider the continuous version of the order conditions which becomes

\[ \sum_{j=1}^{s} b_j(\theta) \Phi_j(t) = \frac{\theta^{\rho(t)}}{\gamma(t)} \quad \text{for all trees } t \text{ such that } \rho(t) \leq p, \]  

(2.11)

where \( \Phi_j(t) \) is the \( j \)th elementary weight of the tree \( t \), \( \rho(t) \) is the order of \( t \), and \( \gamma(t) \) is a coefficient depending on the tree \( t \). Now, putting

\[ z_j(\theta) := b_j(\theta), \quad j = 1, \ldots, s, \]  

(2.12)

(2.11) becomes

\[ \sum_{j=1}^{s} z_j(\theta) \Phi_j(t) = \frac{\rho(t)\theta^{\rho(t)-1}}{\gamma(t)} \quad \text{for all trees } t \text{ such that } \rho(t) \leq p. \]  

(2.13)

For each \( r \geq 1 \), let \( n_r \) be the number of trees such that \( \rho(t) = r \). Thus, a CERK method of order \( p \) must satisfy \( N_p \) conditions (2.13) where \( N_p = \sum_{r=1}^{p} n_r \). It is well-known that \( n_1 = 1, n_2 = 1, n_3 = 2, n_4 = 4, n_5 = 9 \), such that \( N_1 = 1, N_2 = 2, N_3 = 4, N_4 = 8, N_5 = 17 \). In general, we can number the \( N_p \) trees \( t \) increasingly in terms of \( \rho(t) \), such that \( \rho(t_i) > \rho(t_j) \) only if \( i > j \). We then rewrite the conditions (2.13) as

\[ \sum_{j=1}^{s} \phi_{ij} z_j(\theta) = \frac{\rho(t_i)\theta^{\rho(t_i)-1}}{\gamma(t_i)}, \quad i = 1, \ldots, N_p, \]  

(2.14)

where \( \phi_{ij} = \Phi_j(t_i) \). Moreover, by writing

\[ z_j(\theta) = \sum_{k=0}^{p-1} z_{jk} \theta^k, \]  

(2.15a)

\[ \frac{\rho(t_i)\theta^{\rho(t_i)-1}}{\gamma(t_i)} = \sum_{l=0}^{p-1} q_{il} \theta^l, \]  

(2.15b)
and by defining the $N_p \times s$ matrix $\Phi := ((\phi_{ij}))$, the $s \times p$ matrix $Z := (z_{jk})$ and the $N_p \times p$ matrix $Q := ((q_{ii}))$, (2.14) becomes

$$\Phi Z = Q. \quad (2.16)$$

The $N_p \times s$ matrix $\Phi$ depends on the $s \times s$ matrix $A$ of the coefficients of the RK-method, whereas the $N_p \times p$ matrix $Q$ is independent of $A$. By the way, observe that (2.15b) implies

$$q_{i \rho(t_i)-1} = \frac{\rho(t_i)}{\gamma(t_i)} \quad \text{and} \quad q_{ii} = 0 \quad \text{for} \quad i \neq \rho(t_i) - 1. \quad (2.17)$$

For convenience we introduce the applications

$$F_p : \bigcup_{s \geq 1} \mathcal{L}(R^s, R^e) \longrightarrow \bigcup_{s \geq 1} \mathcal{L}(R^s, R^{N_p}) \text{ such that } F_p(A) := \Phi,$$

and

$$G_p : \bigcup_{s \geq 1} \mathcal{L}(R^s, R^e) \longrightarrow \bigcup_{s \geq 1} \mathcal{L}(R^{s+p}, R^{N_p}) \text{ such that } G_p(A) := \Phi|Q,$$

where $\Phi|Q$ is the $N_p \times (s + p)$ matrix obtained by attaching the rows of $Q$ to the rows of $\Phi$.

**Proposition 2.2** A strictly lower triangular $s \times s$ matrix $A$ defines an $s$ stage CERK method of order $p$ if and only if $\text{rank}(F_p(A)) = \text{rank}(G_p(A))$.

**Proof:** From (2.16) it is obvious that any CERK method of order $p$ satisfies $\text{rank}(F_p(A)) = \text{rank}(G_p(A))$. Vice versa, if an $s \times s$ matrix $A$ is such that $\text{rank}(F_p(A)) = \text{rank}(G_p(A))$, then the system $F_p(A)Z = Q$ has at least one solution $Z$. By (2.12) and (2.15a), this matrix $Z$ defines uniquely $s$ polynomials $b_i(\theta)$, $i = 1, \ldots, s$ of degree $\leq p$ such that $b_i(0) = 0$, and hence, the matrix $A$ defines the stages of some CERK method of order $p$. □
In view of the result above, we are only interested in matrices belonging to the set
\[
\mathcal{M}^p := \{ A \in \bigcup_{s \geq 1} \mathcal{L}(R^s, R^s) | \quad A \text{ is strictly lower triangular, } \quad \text{rank}(F_p(A)) = \text{rank}(G_p(A)) \}.
\]
From this point, we shall say that \( s \) is the dimension of \( A \) if \( A \) is an \( s \times s \) matrix, and we write \( \text{dim}(A) = s \). It is clear that in general we have \( \text{dim}(A) \geq \text{rank}(F_p(A)) \).

**Definition 2.3** \( A \in \mathcal{M}^p \) is **\( p \)-minimal** if \( \text{dim}(A) = \text{rank}(F_p(A)) \). Moreover, we define
\[
\mathcal{M}^p_* := \{ A \in \mathcal{M}^p | A \text{ is } p\text{-minimal} \}.
\]

**Proposition 2.4** If \( A \in \mathcal{M}^p_* \), then it cannot have two identical rows, in particular, we must have \( c_2 \neq 0 \). Moreover, \( \text{dim}(A) \leq N_p \).

*Proof:* The first part follows easily from the fact that two identical rows in \( A \) originate two identical columns in \( F_p(A) \). To see that \( \text{dim}(A) \leq N_p \), it is sufficient to observe that \( \text{rank}(F_p(A)) \leq N_p \). \( \Box \)

The following theorem represents a basic result for our theory, since it allows us to restrict ourselves to consider only \( p \)-minimal matrices.

**Theorem 2.5** Let \( A \in \mathcal{M}^p \) be such that \( \rho := \text{rank}(F_p(A)) < \text{dim}(A) \). Then there exists a matrix \( A^* \in \mathcal{M}^p_* \) such that \( \text{dim}(A^*) = \rho \).

*Proof:* It is sufficient to prove that there exists a matrix \( A' \in \mathcal{M}^p \) such that \( \text{dim}(A') = \text{dim}(A) - 1 \) and \( \text{rank}(F_p(A)) = \rho \). In fact, this procedure can be applied \( \text{dim}(A) - \rho \) times in order to get the desired result. Let \( s := \text{dim}(A) \) and \( \Phi := F_p(A) \). By hypothesis, we can find a column, say the \( k \)th column \( (\phi_{1k}, \ldots, \phi_{N_p k})^T \), which is a linear combination of the preceding \( k - 1 \) columns, that is
\[
(\phi_{1k}, \ldots, \phi_{N_p k})^T = \sum_{j=1}^{k-1} \lambda_j (\phi_{1j}, \ldots, \phi_{N_p j})^T
\]
(2.18)
2.2. Order barriers

for some $\lambda_j \in R$. Now define the $s \times s$ matrix $A''$ as follows

\begin{align}
  a''_{k,j} &:= 0 \quad \forall j = 1, \ldots, s, \quad (2.19a) \\
  a''_{i,k} &:= 0 \quad \forall i = 1, \ldots, s, \quad (2.19b) \\
  a''_{i,j} &:= a_{i,j} + \lambda_j a_{i,k} \quad \forall i = 1, \ldots, s, \ i \neq k, \ j = 1, \ldots, k - 1, \quad (2.19c) \\
  a''_{i,j} &:= a_{i,j} \quad \forall i = 1, \ldots, s, \ i \neq k, \ j = k + 1, \ldots, s. \quad (2.19d)
\end{align}

In order to prove that $A'' \in \mathcal{M}$, first observe that the strictly lower triangular form of $A$ is inherited by $A''$. Now, define $\Phi'': = F_p(A'')$. In view of (2.19a), we can easily conclude that the $k$th column $(\phi_{1,k}, \ldots, \phi_{s,k})^T$ of $\Phi''$ is equal to $(1, 0, \ldots, 0)^T$. Moreover, since $A$ and $A''$ are strictly lower triangular, the first column of both $\Phi$ and $\Phi''$ are equal to $(1, 0, \ldots, 0)^T$. As for the remaining columns of $\Phi''$ we prove by induction on the row index $i$ that they are all equal to the corresponding columns of $\Phi$. This is clearly true for $i = 1$, since the first row of $F_p(A)$ is equal to $(1, \ldots, 1)^T$ for any matrix $A$, and corresponds to the only condition (2.14) of order $r = 1$. We assume that the property is true for all $i \leq n - 1$ and prove it for $i = n$. Select the $n$th condition of (2.14) which corresponds to the tree $t_n$ where $\rho(t_n) \geq 2$. This tree can either have the form $[t_{n}]$ for some tree $t_{n'}$ of order $\rho(t_n) - 1$ or the form $[t_{v_1}, \ldots, t_{v_u}]$ for $u(\geq 2)$ trees $t_{v_i}$ where $1 \leq \rho(t_{v_i}) \leq \rho(t_n) - 2$ and $\rho(t_n) = 1 + \sum_{i=1}^u \rho(t_{v_i})$. In the former case, since $a_{ji} = a''_{ji} = 0$ for $l \geq j$, we have

\begin{align}
  \phi_{n,j} &= \sum_{l=1}^{j-1} a_{jl} \phi_{n,l}, \quad j = 2, \ldots, s, j \neq k, \quad (2.20a) \\

\end{align}

and

\begin{align}
  \phi''_{n,j} &= \sum_{l=1}^{j-1} a''_{jl} \phi''_{n,l}, \quad j = 2, \ldots, s, j \neq k, \quad (2.20b)
\end{align}
whereas in the latter case, with \( t_{ni} \equiv [t_{ni}], \ i = 1, \ldots, u \), (with \( \rho(t_{ni}) = 1 + \rho(t_{ni}) \leq \rho(t_n) - 1 \)), we have

\[
\phi_{n_j} = \prod_{i=1}^{u} \phi_{ni,j} \quad \text{and} \quad \phi_{n_j}'' = \prod_{i=1}^{u} \phi_{ni,j}'' \quad j = 2, \ldots, s, j \neq k. \tag{2.21}
\]

Since we have numbered the conditions increasingly in terms of their order, we get in any case \( n', n_1, \ldots, n_u \leq n - 1 \) and hence by the inductive hypothesis,

\[
\phi_{n_j}'' = \phi_{n_j}'', \quad \forall j = 2, \ldots, s, j \neq k, \tag{2.22}
\]

and

\[
\phi_{n_i,j}'' = \phi_{n_i,j}, \quad \forall i = 1, \ldots, u, \text{ and } j = 2, \ldots, s, j \neq k. \tag{2.23}
\]

Therefore, in the latter case, by (2.21) and (2.23) we immediately get \( d_{n_j}'' = \phi_{n_j}, \ j = 2, \ldots, s, j \neq k \). As for the former case, by (2.20b),(2.22) and (2.19c-d), and since \( d_{n_k}'' = 0 \), we have

\[
\phi_{n,j}'' = \sum_{l=1}^{j-1} a_{jl}^{''} \phi_{n_l}'' = \sum_{l=1}^{j-1} a_{jl}^{''} \phi_{n_l} + a_{jk} \sum_{l=1}^{j-1} \lambda_l \phi_{n_l}''
\]

and hence by (2.18) and since \( a_{jk} = 0 \) for \( j \leq k \),

\[
\phi_{n,j}'' = \sum_{l=1}^{j-1} a_{jl} \phi_{n_l},
\]

which by (2.20a) yields \( d_{n,j}'' = \phi_{n,j}, \ j = 2, \ldots, s, j \neq k \). So the induction works. By (2.18), and since \( (\phi_{n_1}'', \ldots, \phi_{n_k}'')^T = (\phi_{11}'', \ldots, \phi_{N_k}'')^T = (1, 0, \ldots, 0)^T \), we can conclude that the range of \( \Phi \) is equal to the range of \( \Phi'' \) and that \( A'' \in \mathcal{M}^p \) with \( \text{rank}(\Phi'') = \rho \). Moreover, it is clear that
the \( s \times p \) matrix \( Z \) satisfying \( \Phi''Z = Q \) (see (2.16)) can be chosen with the 
kth column equal to the zero-vector, which means that \( b_k(\theta) \equiv 0 \) in (2.1b). 
Furthermore, by (2.19b), the kth stage in (2.1a) is completely useless for the 
CERK method defined by \( A'' \), as it is not involved in the computations of the 
following stages. Consequently, the \((s - 1) \times (s - 1)\) matrix \( A' \) obtained by 
suppressing the kth row and the kth column of \( A'' \) defines the same CERK 
method (without the useless kth stage) and the matrix \( F_p(A') \) is obtained by 
suppressing the kth column of \( \Phi'' \). So \( A' \) is the desired matrix, satisfying 
\( \dim(A') = \dim(A) - 1 \) and \( \text{rank}(F_p(A)) = \rho. \Box \)

By virtue of the theorem above, the following result is now obvious.

**Corollary 2.6** The set \( \mathcal{M}_p^r \) is non-empty for all \( p \geq 1 \), and the minimum 
number of stages \( \text{CEN}(p) \) required for a CERK method of order \( p \) is

\[
\text{CEN}(p) = \min_{A \in \mathcal{M}_p^r} \dim(A).
\]

Moreover, if \( A \in \mathcal{M}_p^r \) and \( A \notin \mathcal{M}_p^r \), then \( \dim(A) > \text{CEN}(p) \).

The problem of finding \( \text{CEN}(p) \) can be slightly simplified by isolating the 
following \( p \) conditions of (2.14) which we shall call the **primary conditions**:

\[
\sum_{j=1}^{s} c_j^{r-1} z_j(\theta) = \theta^{r-1}, \quad r = 1, \ldots, p.
\]

These conditions correspond to the trees defined recursively by \( \tau^r := [\tau, \tau^{r-2}] \), 
where \( \tau := [\tau] \) and \( \tau^2 := [\tau] \). Since the matrices \( A \) are strictly lower triangular, 
the remaining \( N_p - p \) conditions of (2.14), which we shall call **secondary conditions**, 
do not explicitly involve the polynomials \( z_1(\theta) \) and \( z_2(\theta) \), as they satisfy \( \phi_{11} = \phi_{12} = 0 \). Roughly speaking, the dimension of the problem is, 
in some sense, reduced by two units.

**Remark 2.7** Since all the secondary conditions correspond to trees of order \( r \geq 3 \), they always yield \( q_{11} = 0 \) in (2.15b).
Now, for a $s \times s$ matrix $A \in \mathcal{M}_p^*$ with $p \geq 3$, we introduce the following equivalence relation on the set of indices $\{1, \ldots, s\}$

$$i \equiv j \text{ if and only if } c_i = c_j.$$ 

There are $s^*$ equivalence classes $S_1, \ldots, S_{s^*}$, and we assume without restrictions that $1 \in S_1$ (i.e. $c_i = 0 \iff i \in S_1$) and that $2 \in S_2$ (recall that $c_2 \neq 0$ by Proposition 2.4).

**Definition 2.8** For an $s \times s$ matrix $A \in \mathcal{M}_p^*$ with $p \geq 3$ we shall call a **good index set** either the empty set $\emptyset$ or any non-empty subset of $\{3, \ldots, s\}$ which elements do not belong to more than $p - 3$ equivalence classes among $S_3, \ldots, S_{s^*}$.

**Remark 2.9** If $p = 3$, then $S \subset S_1 \cup S_2$ for any good index set $S$.

**Lemma 2.10** Let $A \in \mathcal{M}_p^*$ with $p \geq 3$ and let $S$ be a good index set for $A$. Then, with reference to (2.14) and (2.15a), in the set of polynomials $\{z_j(\theta) \mid j \geq 3, j \notin S\}$ (which is non-empty by (2.10)) there exists at least one, say $z_{J_1}(\theta)$, such that $z_{J_1} \neq 0$.

**Proof:** Choose an index $j_k \in S_k$ for any $k = 3, \ldots, s^*$, and assume, without restrictions, that $S \subset S_1 \cup S_2 \cup \cdots \cup S_r$ for some $r \leq p - 1 \leq s^* - 1$. Since the polynomials $z_j(\theta)$, $j = 1, \ldots, s$ satisfy the primary conditions (2.24) for any polynomial $\pi(\theta) \in \Pi_{p-1}$, we easily get

$$\pi(\theta) = \sum_{j=1}^s \pi(c_j) z_j(\theta) = \sum_{k=1}^{s^*} \pi(c_{j_k}) \sum_{j \in S_k} z_j(\theta). \tag{2.25}$$

So, if we define $\pi(\theta) := \theta(\theta - c_2)(\theta - c_{j_1})\cdots(\theta - c_{j_r})$, we get $\pi(c_j) = 0$ for all $j \in S_1 \cup S_2 \cup \cdots \cup S_r$ such that (2.25) becomes

$$\pi(\theta) = \sum_{k=r+1}^{s^*} \pi(c_{j_k}) \sum_{j \in S_k} z_j(\theta).$$
2.2. Order barriers

Since the coefficient of $\theta$ in $\pi(\theta)$ is $(-1)^{r-1}c_2c_3\cdots c_r \neq 0$, and since $\pi(c_{jk}) \neq 0$ for all $k = r + 1, \ldots, s^*$, the proof is complete. □

We shall say that $N(\geq 1)$ conditions (2.14) are linearly independent if and only if the corresponding $N$ rows of the matrix $G_p(A)$ are linearly independent.

Lemma 2.11 Let $A \in M^r$ and let $N$ conditions (2.14) be linearly independent. Then they explicitly involve $N$ polynomials $z_j(\theta)$.

Proof: It is sufficient to observe that, since $\text{rank}(F_p(A)) = \text{rank}(G_p(A))$, if $N$ rows of the matrix $G_p(A)$ are linearly independent, then the same $N$ rows of the matrix $F_p(A)$ must be linearly independent too. □

Now we are in a position to state the main result of this section, which is a tool to find lower bounds for $CEN(p)$.

Theorem 2.12 Let $A \in M^r$ with $p \geq 3$, and let $N$ secondary conditions from (2.14) be linearly independent. Let $S$ be the set formed by the indices $j \geq 3$ of the polynomials $z_j(\theta)$ which are not explicitly involved by these $N$ conditions (possibly $S = \emptyset$). Then

$$\dim(A) \geq N + \mu + 2,$$

where $\mu$ is the cardinality of $S$. Moreover, if $S$ is a good index set for $A$, then

$$\dim(A) \geq N + \mu + 3.$$

Proof: Since the $N$ conditions we consider are secondary, they involve neither $z_1(\theta)$ nor $z_2(\theta)$. Thus, by Lemma 2.11 we have in any case that $\dim(A) \geq N + \mu + 2$. To prove the stronger inequality, assume that $S$ is a good index set for $A$ and that $\dim(A) = N + \mu + 2$. Then, still by Lemma 2.11, there are exactly $N$ polynomials $z_j(\theta)$, the only existing ones with $j \geq 3$ and $j \notin S$, which are involved by these $N$ conditions. Moreover, the $N \times N$
matrix $\Phi^*$, obtained by suppressing the $\mu + 2$ vanishing elements relevant to the the missing polynomials $z_j(\theta)$ in each of the corresponding $N$ rows of $F_p(A)$, is non-singular. Thus, solving the subsystem (2.16) defined by these $N$ conditions yields $z_j = 0$ for all $j \geq 3, j \notin S$, which contradicts Lemma 2.10. □

2.2.2 Finding $\text{CEN}(p)$ for $p \leq 5$

Now we shall apply the results from Section 2 in order to find the minimum number of stages $\text{CEN}(p)$ for $p = 3, 4, 5$. Our strategy will always be the following:

i) In view of Corollary 2.6 we consider an $s \times s$ matrix $A \in \mathcal{M}_s^*$. We assume the maximum number of linearly independent secondary conditions to be 1, 2, \ldots, $N_p - p$ in sequence, so that we either obtain an absurdity or, by Theorem 2.12, a lower bound for $s = \dim(A)$.

ii) We will compare these lower bounds to the upper bound given by some existing method, already known for $p = 3, 4$ (see (2.9)) and new for $p = 5$.

We shall denote by $\phi^{(i)}$ the $i$th row $(\phi_{i1}, \ldots, \phi_{is})$ of the matrix $F_p(A)$ and $\psi^{(i)}$ the $i$th row $(\phi_{i1}, \ldots, \phi_{is}, q_{0i}, \ldots, q_{i-1})$ of the matrix $G_p(A)$. Moreover, for each $r \geq 1$, let $R_r$ be the set of rows of the matrix $F_p(A)$ which correspond to conditions of order $r$. In the proof of Theorem 2.5 we saw that, if $\phi^{(i)} \in R_r$, then either $\phi^{(i)} = \phi^{(i')} A^T$ where $\phi^{(i')} \in R_{r-1}$ (and we shall say that $\phi^{(i)}$ is an $A$-transformation) or $\phi^{(i)} = \prod_{s=1}^{i} \phi_{isj}$, $j = 1, \ldots, s$, where $\phi^{(i_s)} \in R_{r_n}$ with $r_n \leq r - 1$. In particular, in the latter case we may have $\phi^{(i)} = \phi^{(i')} C$ where $\phi^{(i')} \in R_{r-1}$ and $C := \text{diag}(0, e_2, \ldots, e_s)$ (and we shall say that $\phi^{(i)}$ is a $C$-transformation). Note that the primary condition of order $r$ is the $C$-transformation of the primary condition of order $r - 1$. In view of this, and since we have decided to number the conditions increasingly in terms of their order, each set of $n_r$ conditions of order $r$ will be numbered as follows: First the $C$-transformations, then the $A$-transformations, and finally
the remaining conditions if there are any. It turns out that the primary condition always will be the first in each set of $n_p$ conditions.

Order $p=3$

There is only one secondary condition for the case $p = 3$. This condition corresponds to $\psi^{(1)} = (\phi^{(4)}, 0, 0, 1/2)$, where $\phi^{(4)} = \phi^{(2)} A^T$. Since $\psi^{(4)} \neq 0$, Theorem 2.12 yields in any case $s \geq 4$. Thus, the existence of the 4-stage CERK method of order 3 associated with the RKN(3,4) embedded pair (see Enright et al [34]) implies

$$CEN(3) = 4$$

Alternatively, this upper bound also follows from Proposition 2.4 since $N_3 = 4$. Moreover, it is clear that every $4 \times 4$ matrix $A$ that results in a nonsingular $F_3(A)$ (which is a $4 \times 4$ matrix as well) determines a 4-stage CERK method of order 3.

Order $p=4$

There are 4 secondary conditions: one of order 3, corresponding to

$$\psi^{(4)} = (\phi^{(4)}, 0, 0, 1/2, 0),$$

and 3 ones of order 4, corresponding to

$$\psi^{(6)} = (\phi^{(6)}, 0, 0, 1/2) \quad \psi^{(7)} = (\phi^{(7)}, 0, 0, 0, 1/3)$$

$$\psi^{(8)} = (\phi^{(8)}, 0, 0, 0, 1/6)$$

Moreover, $\phi^{(6)} = \phi^{(4)} C$, $\phi^{(7)} = \phi^{(3)} A^T$ and $\phi^{(8)} = \phi^{(4)} A^T$. To begin with, we observe that the maximum number $N$ of linearly independent conditions clearly obeys $N \geq 2$. Then assume $N = 2$. In this case $\psi^{(6)}, \psi^{(7)}$ and $\psi^{(8)}$ are proportional and therefore, since $\phi_{83} = 0$, we get $\phi_{63} = 0$ and $\phi_{73} = a_{32} e_2^3 = 0$.
as well. So $a_{32} = 0$ because $c_2 \neq 0$, and hence $\phi_{43} = a_{32}c_2 = 0$, which means that $z_3(\theta)$ is not involved by the secondary conditions. Consequently, with
reference to Theorem 2.12, we have $S \supset \{3\}$, which is a good index set for $A$, and hence, in any case we obtain $s \geq 6$. $N \geq 3$ implies in any case $s \geq 6$ by Theorem 2.12. The 6 stage CERK method of order 4 associated with
the Dormand Prince(5,4) embedded pair (see Section 2.1) provides an upper bound for $CEN(4)$ so we have

$$CEN(4) = 6$$

Now, consider the conditions to be imposed on a $6 \times 6$ matrix $A$, necessarily 4-minimal (see Corollary 2.6), in order that it determines a CERK method of order 4. First, observe that, in view of the case $p = 3$ above, the rows $\psi^{(1)}, \psi^{(2)}, \psi^{(3)}, \psi^{(4)}$ and $\psi^{(5)}$ must be linearly independent. Note that $\psi^{(1)} = (1, 1, 1, 1, 1, 0, 0)$, $\psi^{(2)} = (0, c_2, c_3, c_4, c_5, 0, 1, 0)$, $\psi^{(3)} = (0, c_2^2, c_3^2, c_4^2, c_5^2, 0, 0, 1, 0)$ and $\psi^{(5)} = (0, c_2^3, c_3^3, c_4^3, c_5^3, 0, 0, 0, 0, 1)$ correspond to the primary conditions. We can conclude that to have $\text{rank}(G_4(A)) = 6$, it is necessary and sufficient that at least one of the following conditions is satisfied, where $S := \text{span}\{\psi^{(1)}, \psi^{(2)}, \psi^{(3)}, \psi^{(4)}, \psi^{(5)}\}$:

j) $\psi^{(6)} \notin S$ and $\psi^{(7)}, \psi^{(8)} \in \text{span}\{S, \psi^{(6)}\}$

jj) $\psi^{(7)} \notin S$ and $\psi^{(6)}, \psi^{(8)} \in \text{span}\{S, \psi^{(7)}\}$

jjj) $\psi^{(8)} \notin S$ and $\psi^{(6)}, \psi^{(7)} \in \text{span}\{S, \psi^{(8)}\}$

Of course we must also require $\text{rank}(F_4(A)) = 6$. Now consider the following three groups of conditions for the coefficients of $A$.

\begin{align*}
c_2(c_4^2 - 2\phi_{44}) &= c_4^3 + \lambda c_4 \phi_{44} + \mu(a_{42}c_2^2 + a_{43}c_3^2) \\
c_2(c_5^2 - 2\phi_{45}) &= c_5^3 + \lambda c_5 \phi_{45} + \mu(a_{52}c_2^2 + a_{53}c_3^2 + a_{54}c_4^2) \\
c_2(c_6^2 - 2\phi_{46}) &= c_6^3 + \lambda c_6 \phi_{46} + \mu(a_{62}c_2^2 + a_{63}c_3^2 + a_{64}c_4^2 + a_{65}c_5^2)
\end{align*} \hspace{1cm} (2.26a/2.26b/2.26c)
where

$$\lambda = 2 \frac{a_{32} c_2^3 + c_3 (c_2 - c_3)}{a_{32} c_2 (2c_3 - 3c_2)} \quad \text{and} \quad \mu = \frac{3(c_2 - c_3)(2a_{32} c_2 - c_3^2)}{a_{32} c_2 (2c_3 - 3c_2)}. \quad (2.26d)$$

$$c_2(c_4^2 - 2\phi_{44}) = c_4^3 + \lambda c_4 \phi_{44} + \mu a_{43} \phi_{43} \quad (2.27a)$$

$$c_2(c_5^2 - 2\phi_{45}) = c_5^3 + \lambda c_5 \phi_{45} + \mu(a_{53} \phi_{43} + a_{54} \phi_{44}) \quad (2.27b)$$

$$c_2(c_6^2 - 2\phi_{46}) = c_6^3 + \lambda c_6 \phi_{46} + \mu(a_{63} \phi_{43} + a_{64} \phi_{44} + a_{65} \phi_{45}) \quad (2.27c)$$

where

$$\lambda = \frac{c_2^3 (c_2 - c_3) - 2a_{32} c_2^2}{a_{32} c_2 c_3} \quad \text{and} \quad \mu = \frac{3(c_2 - c_3)(2a_{32} c_2 - c_3^2)}{a_{32} c_2 c_3}. \quad (2.27d)$$

$$c_2(c_4^2 - 2\phi_{44}) = c_4^3 + \lambda (a_{42} c_2^2 + a_{43} c_2^2) + \mu a_{43} \phi_{43} \quad (2.28a)$$

$$c_2(c_5^2 - 2\phi_{45}) = \left( c_5^3 + \lambda (a_{52} c_2^2 + a_{53} c_3^2 + a_{54} c_4^2) \right)$$

$$\quad + \mu(a_{53} \phi_{43} + a_{54} \phi_{44}) \quad (2.28b)$$

$$c_2(c_6^2 - 2\phi_{46}) = \left( c_6^3 + \lambda (a_{62} c_2^2 + a_{63} c_3^2 + a_{64} c_4^2 + a_{65} c_5^2) \right)$$

$$\quad + \mu(a_{63} \phi_{43} + a_{64} \phi_{44} + a_{65} \phi_{45}) \quad (2.28c)$$

where

$$\lambda = \frac{c_2^3 (c_2 - c_3) - 2 a_{32} c_2^2}{a_{32} c_2 c_3} \quad \text{and} \quad \mu = -\frac{2 a_{32} c_2^2 + 2 (c_2 - c_3)}{a_{32} c_2 c_3}. \quad (2.28d)$$

Recall that $\phi_{43} = a_{32} c_2$, $\phi_{44} = a_{42} c_2 + a_{43} c_3$, $\phi_{45} = a_{52} c_2 + a_{53} c_3 + a_{54} c_4$ and $\phi_{46} = a_{62} c_2 + a_{63} c_3 + a_{64} c_4 + a_{65} c_5$. 

2.2. Order barriers
Simple, but tedious calculations lead to the fact that, for all cases (j), (jj), and (jjj), we must have $a_{32} \neq 0$ (otherwise the matrix $A$ would not be 4-minimal) and that:

(j) is equivalent to (2.26a-d) and (2.27a-d) where $c_2 \neq c_3$, $c_3 \neq 0$, $3c_2 - 2c_3 \neq 0$ and $2a_{32}c_2 - c_3^2 \neq 0$.

(jj) is equivalent to (2.26a-d) and (2.28a-d) where $3c_2 - 2c_3 \neq 0$ and $a_{32}c_2^2 + c_3^2(c_2 - c_3) \neq 0$.

(jjj) is equivalent to (2.27a-d) and (2.28a-d) where where $c_3 \neq 0$ and $c_3^2(c_2 - c_3) - 2a_{32}c_2^2 \neq 0$.

Moreover, it is also easy to see that, for all cases (j), (jj) and (jjj), the following conditions, expressing that $\psi^{(6)}, \psi^{(7)}$ and $\psi^{(8)}$ are linearly dependent, can equivalently replace either of the two corresponding groups of conditions among (2.26a-d), (2.27a-d) and (2.28a-d)

\[
a_{42}c_2^2(c_3 - c_4) + a_{43}c_3(c_3^2 - c_2c_4) = (2c_3 - 3c_2)a_{43}\phi_{43} \quad (2.29a)
\]

\[
a_{52}c_2^2(c_3 - c_5) + a_{53}c_3(c_3^2 - c_2c_5) + a_{54}c_4(c_3c_4 - c_2c_5) = (2c_3 - 3c_2)(a_{53}\phi_{43} + a_{54}\phi_{44}) \quad (2.29b)
\]

\[
a_{62}c_2^2(c_3 - c_6) + a_{63}c_3(c_3^2 - c_2c_6) + a_{64}c_4(c_3c_4 - c_2c_6) + a_{65}c_5(c_3c_5 - c_2c_6) = (2c_3 - 3c_2)(a_{63}\phi_{43} + a_{64}\phi_{44} + a_{65}\phi_{45}). \quad (2.29c)
\]

The method associated with the Dormand-Prince (5,4) pair obeys $2c_3 - 3c_2 = 0$ (i.e. $\psi^{(6)}$ and $\psi^{(7)}$ are proportional and (jjj) holds, but (j) and (jj) do not). This method is included in the class of 6-stage CERK methods of order 4 which is obtained by imposing $c_3 \neq 0$ and the following two groups of conditions

\[
2\phi_{43} = c_3^2 \quad (2.30a)
\]

\[
2\phi_{44} = c_4^2 \quad (2.30b)
\]
2.2. Order barriers

\[ 2\phi_{45} = c_5^2 \]  
\[ 2\phi_{46} = c_6^2, \]  
\[ 2a_42c_2c_3 + 3a_43c_3^2 = c_4^3 \]  
\[ 2a_52c_2c_3 + 3(a_53c_3^2 + a_54c_4^2) = c_5^3 \]  
\[ 2a_62c_2c_3 + 3(a_63c_3^2 + a_64c_4^2 + a_65c_5^2) = c_6^3. \]

Indeed, \( c_3 \neq 0 \) and (2.30a-d) imply (2.27a-c) with \( \lambda = -2 \) and \( \mu = 0 \). Moreover, since (2.27a-c) are satisfied, conditions (2.30a-c) are equivalent to conditions (2.30a-d) with \( \lambda = -2c_3/c_2 \) and \( \mu = 2c_3^2/c_2 \). Therefore, since \( c_3(c_2 - c_3) - 2a_32c_2^2 = -c_3^3 \neq 0 \) it follows that (jjj) holds. Summarizing, we can choose arbitrary \( c_2, c_3 \) and \( c_4 \), subject to the only restrictions \( c_2 \neq 0, \ c_3 \neq 0 \), \( c_3 \neq c_4 \) and we get

\[ a_{32} = \frac{c_3^2}{2c_2} \]  
\[ a_{42} = \frac{(3c_3 - 2c_4)c_4^2}{2c_2c_3} \quad a_{43} = \frac{(c_4 - c_3)c_4^2}{c_3^2} \]  

Furthermore, we can choose arbitrary \( c_5, a_{54}, c_6, a_{64}, a_{65} \) (subject to \( A \) being 4-minimal) leading to

\[ a_{52} = \frac{(3c_3 - 2c_5)c_5^2 + 6a_{54}c_5(c_3 - c_5)}{2c_3c_5} \]  
\[ a_{53} = \frac{(c_5 - c_3)c_5^2 - a_{54}c_5(3c_4 - 2c_3)}{c_3^2}, \]  
\[ a_{62} = \frac{(3c_3 - 2c_6)c_6^2 + 6a_{64}c_6(c_3 - c_6) + 6a_{65}c_6(c_5 - c_6)}{2c_3c_5} \]  
\[ a_{63} = \frac{(c_6 - c_3)c_6^2 - a_{64}c_6(3c_4 - 2c_3) - a_{65}c_6(3c_5 - 2c_3)}{c_3^2}. \]
Order $p=5$

There are 12 secondary conditions, corresponding to

$$\psi^{(4)} = (\phi^{(4)}, 0, 0, 1/2, 0, 0) \quad \text{of order 3},$$

$$\psi^{(6)} = (\phi^{(6)}, 0, 0, 0, 1/2, 0), \quad \psi^{(7)} = (\phi^{(7)}, 0, 0, 0, 1/3, 0),$$

$$\psi^{(8)} = (\phi^{(8)}, 0, 0, 0, 1/6, 0) \quad \text{of order 4},$$

and

$$\psi^{(10)} = (\phi^{(10)}, 0, 0, 0, 0, 1/2), \quad \psi^{(11)} = (\phi^{(11)}, 0, 0, 0, 0, 1/3),$$

$$\psi^{(12)} = (\phi^{(12)}, 0, 0, 0, 0, 1/6), \quad \psi^{(13)} = (\phi^{(13)}, 0, 0, 0, 0, 1/4),$$

$$\psi^{(14)} = (\phi^{(14)}, 0, 0, 0, 0, 1/8), \quad \psi^{(15)} = (\phi^{(15)}, 0, 0, 0, 0, 1/12),$$

$$\psi^{(16)} = (\phi^{(16)}, 0, 0, 0, 0, 1/24), \quad \psi^{(17)} = (\phi^{(17)}, 0, 0, 0, 0, 1/4)$$

of order 5.

Moreover, $\phi^{(10)} = \phi^{(6)} C$, $\phi^{(11)} = \phi^{(7)} C$, $\phi^{(12)} = \phi^{(8)} C$, $\phi^{(13)} = \phi^{(5)} A^T$, $\phi^{(14)} = \phi^{(6)} A^T$, $\phi^{(15)} = \phi^{(7)} A^T$, $\phi^{(16)} = \phi^{(8)} A^T$ and $\phi^{17}_j = (\phi^{4}_j)^2$, $j = 1, \ldots, s$.

To begin with, we observe that the maximum number $N$ of linearly independent secondary conditions clearly obeys $N \geq 3$. If we assume $N = 3$, then $\psi^{(16)}$ and $\psi^{(17)}$ are proportional and hence, since $\phi^{(16)} = \phi^{(4)}(A^T)^2$ and $\phi^{17}_j = (\phi^{4}_j)^2$, $j = 1, \ldots, s$, we easily obtain the absurdity $\phi^{(16)} = \phi^{(17)} = 0$. Therefore, $\psi^{(16)}$ and $\psi^{(17)}$ must be linearly independent, and we can conclude that $N \geq 4$. So we assume $N = 4$. From above, we know that the dimension of span$\{\psi^{(10)}, \psi^{(11)}, \psi^{(12)}, \psi^{(13)}, \psi^{(14)}, \psi^{(15)}, \psi^{(16)}, \psi^{(17)}\}$ equals 2. First assume that $\psi^{(17)}$ is a linear combination of $\psi^{(12)}, \psi^{(14)}, \psi^{(15)}$ and $\psi^{(16)}$. In this case, since $\phi^{12}_3 = \phi^{14}_3 = \phi^{15}_3 = \phi^{16}_3 = 0$ we also have $\phi^{17}_3 = (a_{32} c_2)^2 = 0$. Again, because $c_2 \neq 0$, we must have $a_{32} = 0$ and hence, $\phi_{43} = a_{32} c_2 = 0$, $\phi_{53} = a_{32} c_2 c_3 = 0$, $\phi_{73} = a_{32} c_2^2 = 0$, $\phi_{10} = a_{32} c_2 c_3^2 = 0$, $\phi_{11} = a_{32} c_2^2 c_3 = 0$ and $\phi_{13} = a_{32} c_2^3 = 0$. Since also $\phi_{83} = 0$, 

\[ \text{...} \]
the polynomial $z_3(\theta)$ is not involved by the secondary conditions, and therefore, with reference to Theorem 2.12 we get $S \supset \{3\}$, which is a good index set for $A$, and $s \geq 8$. Then assume $a_{32} \neq 0$ and that $\psi^{[17]}$ is linearly independent of $\psi^{[12]}, \psi^{[14]}, \psi^{[15]}$ and $\psi^{[16]}$. In this case, the rows $\psi^{[12]}, \psi^{[14]}, \psi^{[15]}$ and $\psi^{[16]}$ must be proportional, and consequently, since $\phi_{16} 4 = 0$, we must also have $\phi_{12} 4 = \phi_{14} 4 = 0$ and $\phi_{15} 4 = a_{43} a_{32} c_2^2 = 0$. $a_{32} c_2^2 \neq 0$ implies $a_{43} = 0$ and $a_{84} = a_{43} a_{32} c_2 c_4 = 0$. Moreover, since $\psi^{[6]}, \psi^{[7]}$ and $\psi^{[8]}$ cannot be linearly independent (this would imply $N \geq 5$), (2.29a) holds, and reduces to $a_{42} c_2^2(c_3 - c_4) = 0$. So only two cases are possible, either $a_{42} \neq 0$ and $c_3 = c_4$ or $a_{42} = 0$. Indeed, the former case cannot be true, since it contradicts the fact that $A$ is 5-minimal. In fact, since $\psi^{[13]}$ must depend linearly on $\psi^{[16]}$ and $\psi^{[17]}$, and since $\phi_{16} 3 = \phi_{16} 4 = 0$, there should exist $\lambda \neq 0$ such that $\phi_{13} 3 = \lambda \phi_{17} 3$ and $\phi_{13} 4 = \lambda \phi_{17} 4$, leading to $a_{32} c_2^3 = \lambda(a_{32} c_2^2)^2$ and $a_{42} c_2^2 = \lambda(a_{42} c_2)^2$ such that $a_{32} = a_{42}$. Thus, the matrix $A$ has two equal rows, contradicting Proposition 2.4. So we are left with $a_{42} = 0$ which implies $\phi_{44} = a_{42} c_2 + a_{43} c_3 = 0$, $\phi_{04} = c_4(4 a_{42} c_2 + a_{43} c_3) = 0$, $\phi_{74} = a_{42} c_2^2 + a_{43} c_3^2 = 0$, $\phi_{10} 4 = c_4^2(a_{42} c_2 + a_{43} c_3) = 0$, $\phi_{11} 4 = c_4(a_{42} c_2^2 + a_{43} c_3^2) = 0$, $\phi_{12} 4 = a_{42} c_2^3 + a_{43} c_3^2 = 0$ and $\phi_{17} 4 = (a_{42} c_2 + a_{43} c_3)^3 = 0$. In conclusion, $z_4(\theta)$ is not involved by the secondary conditions, so with reference to Theorem 2.12 we have $S \supset \{4\}$ which is a good index set for $A$, and hence, $s \geq 8$. If $N \geq 5$, then Theorem 2.12 yields again, in any case, $s \geq 8$.

As far as we know, the cheapest known CERK methods of order 5 require 9 stages. As an example, we quote the 9-stage CERK method of order 5 associated with the RKV(5,6) embedded pair (see Enright et al. [34]). However, since we shall find examples of 8-stage CERK methods of order 5, we can conclude that

$$CEN(5) = 8$$

Now we want to find conditions to be imposed on an $8 \times 8$ matrix $A$, necessarily 5-minimal, such that it determines a CERK method of order 5. In view of the previous case, $p = 4$, we observe that at least 6 rows among $\psi^{[i]}$, $i = 1, \ldots, 8$, must be linearly independent. So we can for example assume that $\psi^{[1]}, \psi^{[2]}, \psi^{[3]}, \psi^{[4]}, \psi^{[5]}$ and $\psi^{[8]}$ are linearly independent, and then impose condition (jjjj). Then we must have $a_{32} \neq 0$, $c_3 \neq 0$ and
\(c_3^2(c_2 - c_3) - 2a_{32}c_2^2 \neq 0\). Moreover conditions (2.27a-d) and (2.28a-d) (or, equivalently (2.27a-d) and (2.29a-d)) must be satisfied. However, since we now have \(\text{dim}(A) = 8\), we must supply both (2.27a-d) and (2.28a-d) with two conditions corresponding to the last two stages

\[
c_2(c_2^2 - 2\phi_{47}) = c_7^3 + \lambda c_7 \phi_{47} + \mu (a_{73} \phi_{43} + a_{74} \phi_{44} + a_{75} \phi_{45} + a_{76} \phi_{46})
\]

\[
c_2(c_8^2 - 2\phi_{48}) = c_8^3 + \lambda c_8 \phi_{48} + \mu (a_{83} \phi_{43} + a_{84} \phi_{44} + a_{85} \phi_{45} + a_{86} \phi_{46} + a_{87} \phi_{47})
\]

and

\[
c_2(c_7^2 - 2\phi_{47}) = c_7^3 + \lambda (a_{72} c_2^2 + a_{73} c_3^2 + a_{74} c_4^2 + a_{75} c_5^2 + a_{76} c_6^2) + \mu (a_{73} \phi_{43} + a_{74} \phi_{44} + a_{75} \phi_{45} + a_{76} \phi_{46})
\]

\[
c_2(c_8^2 - 2\phi_{48}) = c_8^3 + \lambda (a_{82} c_2^2 + a_{83} c_3^2 + a_{84} c_4^2 + a_{85} c_5^2 + a_{86} c_6^2) + \mu (a_{83} \phi_{43} + a_{84} \phi_{44} + a_{85} \phi_{45} + a_{86} \phi_{46} + a_{87} \phi_{47})
\]

where \(\phi_{47} = a_{72} c_2 + a_{73} c_3 + a_{74} c_4 + a_{75} c_5 + a_{76} c_6\) and \(\phi_{48} = a_{82} c_2 + a_{83} c_3 + a_{84} c_4 + a_{85} c_5 + a_{86} c_6 + a_{87} c_7\).

Now, observe that, if \(\phi^{(i)} \in R_r\) is a C-transformation of \(\phi^{(i)} \in R_{r-1}\), then by (2.17) we get \(q_{i-1} = q_{i-2}\), whereas if \(\phi^{(i)} \in R_r\) is an A-transformation of \(\phi^{(i)} \in R_{r-1}\), then \(q_{i-1} = q_{i-2}/(r-1)\). So we can conclude that condition (v) automatically implies

1. \(\psi^{(10)}, \psi^{(11)} \in \text{span}\{\psi^{(2)}, \psi^{(3)}, \psi^{(5)}, \psi^{(6)}, \psi^{(9)}, \psi^{(12)}\}\) and
2. \(\psi^{(14)}, \psi^{(15)} \in \text{span}\{\psi^{(2)}, \psi^{(4)}, \psi^{(7)}, \psi^{(8)}, \psi^{(13)}, \psi^{(16)}\}\). Therefore, we assume that \(\psi^{(9)}, \psi^{(10)} \notin \text{span}\{S, \psi^{(8)}\}\) and impose that
\[ v_j, \psi^{(12)}, \psi^{(13)}, \psi^{(17)} \in \text{span}\{ S, \psi^{(8)}, \psi^{(9)}, \psi^{(16)} \} \]

in order to get \( \text{rank}(G_5(A)) = 8 \).

Once again, simple but tedious calculations lead to \( v_j \) being equivalent to the following three groups of conditions on the coefficients of \( A \).

\[
\lambda(c_5^2 - 2\phi_{45}) + \mu[c_5^2 - 6(a_{53}\phi_{43} + a_{54}\phi_{44})] = c_5^4 + \\
\nu(a_{53}\phi_{43} + a_{54}\phi_{44})c_5 + \rho a_{54}a_{43}\phi_{43},
\]

(2.33a)

\[
\lambda(c_6^2 - 2\phi_{46}) + \mu[c_6^2 - 6(a_{63}\phi_{43} + a_{64}\phi_{44} + a_{65}\phi_{45})] = c_6^4 + \\
\nu(a_{63}\phi_{43} + a_{64}\phi_{44} + a_{65}\phi_{45})c_6 + \rho(a_{64}a_{43}\phi_{43} + a_{65}(a_{53}\phi_{43} + a_{54}\phi_{44})),
\]

(2.33b)

\[
\lambda(c_7^2 - 2\phi_{47}) + \mu[c_7^2 - 6(a_{73}\phi_{43} + a_{74}\phi_{44} + a_{75}\phi_{45} + a_{76}\phi_{46})] = c_7^4 + \\
\nu(a_{73}\phi_{43} + a_{74}\phi_{44} + a_{75}\phi_{45} + a_{76}\phi_{46})c_7 + \rho(a_{74}a_{43}\phi_{43}) + a_{75}(a_{53}\phi_{43} + a_{54}\phi_{44} + a_{65}\phi_{45})),
\]

(2.33c)

\[
\lambda(c_8^2 - 2\phi_{48}) + \mu[c_8^2 - 6(a_{83}\phi_{43} + a_{84}\phi_{44} + a_{85}\phi_{45} + a_{86}\phi_{46} + a_{87}\phi_{47})] = c_8^4 + \\
\nu(a_{83}\phi_{43} + a_{84}\phi_{44} + a_{85}\phi_{45} + a_{86}\phi_{46} + a_{87}\phi_{47})c_8 + \rho(a_{84}a_{43}\phi_{43} + a_{85}(a_{53}\phi_{43} + a_{54}\phi_{44}) + a_{86}(a_{63}\phi_{43} + a_{64}\phi_{44} + a_{65}\phi_{45}) + a_{87}(a_{73}\phi_{43} + a_{74}\phi_{44} + a_{75}\phi_{45} + a_{76}\phi_{46})),
\]

(2.33d)

where \( \rho = -4\nu - 24 \), and \((\lambda, \mu, \nu)\) is the solution of the system

\[
\lambda + \mu c_2 = c_2^2,
\]

(2.33e)
\[
\lambda(c_3^2 - 2\phi_{43}) + \mu c_3^3 = c_3^4, \quad (2.33f)
\]

\[
\lambda(c_4^2 - 2\phi_{44}) + \mu (c_4^3 - 6a_{43}\phi_{43}) = c_4^4 + \nu a_{43}\phi_{43}c_4. \quad (2.33g)
\]

\[
\lambda(c_5^2 - 2\phi_{45}) + \mu [c_5^3 - 6(a_{53}\phi_{43} + a_{54}\phi_{44})] = c_5^4 + \nu (a_{52}c_3^3 + a_{53}c_4^3 + a_{54}c_4^3) + \rho a_{54}\phi_{43},
\]

\[
\lambda(c_6^2 - 2\phi_{46}) + \mu [c_6^3 - 6(a_{63}\phi_{43} + a_{64}\phi_{44} + a_{65}\phi_{45})] = c_6^4 + \nu (a_{62}c_3^3 + a_{63}c_4^3 + a_{64}c_4^3 + a_{65}c_5^3) + \rho [a_{64}a_{43}\phi_{43} + a_{65}(a_{53}\phi_{43} + a_{54}\phi_{44})],
\]

\[
\lambda(c_7^2 - 2\phi_{47}) + \mu [c_7^3 - 6(a_{73}\phi_{43} + a_{74}\phi_{44} + a_{75}\phi_{45} + a_{76}\phi_{46})] = c_7^4 + \nu (a_{72}c_3^3 + a_{73}c_4^3 + a_{74}c_4^3 + a_{75}c_5^3 + a_{76}c_6^3) + \rho [a_{74}a_{43}\phi_{43} + a_{75}(a_{53}\phi_{43} + a_{54}\phi_{44} + a_{55}\phi_{45})],
\]

where \(\rho = -6\nu - 24\), and \((\lambda, \mu, \nu)\) is the solution of the system

\[
\lambda + \mu c_2 = c_2^2, \quad (2.34e)
\]

\[
\lambda(c_3^2 - 2\phi_{43}) + \mu c_3^3 = c_3^4 + \nu a_{32}c_2^3, \quad (2.34f)
\]
\begin{align*}
\lambda (c_4^2 - 2\phi_{44}) + \mu (c_4^3 - 6a_{43}\phi_{43}) &= c_4^4 + \nu (a_{42}c_2^3 + a_{43}c_3^3). \tag{2.34g} \\
\lambda (c_5^2 - 2\phi_{45}) + \mu [c_5^3 - 6(a_{53}\phi_{43} + a_{54}\phi_{44})] &= c_5^4 + \nu (\phi_{45})^2 + \rho a_{54}a_{43}\phi_{43}. \tag{2.35a} \\
\lambda (c_6^2 - 2\phi_{46}) + \mu [c_6^3 - 6(a_{63}\phi_{43} + a_{64}\phi_{44} + a_{65}\phi_{45})] &= c_6^4 + \\
&+ \nu (\phi_{46})^2 + \rho [a_{64}a_{43}\phi_{43} + a_{65}(a_{53}\phi_{43} + a_{54}\phi_{44})], \tag{2.35b} \\
\lambda (c_7^2 - 2\phi_{47}) + \mu [c_7^3 - 6(a_{73}\phi_{43} + a_{74}\phi_{44} + a_{75}\phi_{45} + a_{76}\phi_{46})] &= c_7^4 + \nu (\phi_{47})^2 + \rho [a_{74}a_{43}\phi_{43} + a_{75}(a_{63}\phi_{43} + a_{64}\phi_{44})] \\
&+ a_{76}(a_{63}\phi_{43} + a_{64}\phi_{44} + a_{65}\phi_{45}), \tag{2.35c} \\
\lambda (c_8^2 - 2\phi_{48}) + \mu [c_8^3 - 6(a_{83}\phi_{43} + a_{84}\phi_{44} + a_{85}\phi_{45} + a_{86}\phi_{46} + \\
a_{87}\phi_{47})] &= c_8^4 + \nu (\phi_{48})^2 + \rho [a_{84}a_{43}\phi_{43} + a_{85}(a_{73}\phi_{43} + a_{74}\phi_{44})] \\
&+ a_{86}(a_{73}\phi_{43} + a_{74}\phi_{44} + a_{75}\phi_{45} + a_{76}\phi_{46})], \tag{2.35d} \\
\end{align*}

where \( \rho = -6\nu - 24 \), and \( (\lambda, \mu, \nu) \) is the solution of the system

\begin{align*}
\lambda + \mu c_2 &= c_2^2, \tag{2.35e} \\
\lambda (c_3^2 - 2\phi_{43}) + \mu c_3^3 &= c_3^4 + \nu (\phi_{43})^2, \tag{2.35f} \\
\lambda (c_4^2 - 2\phi_{44}) + \mu (c_4^3 - 6a_{43}\phi_{43}) &= c_4^4 + \nu (\phi_{44})^2. \tag{2.35g} 
\end{align*}

Of course the elements of \( A \) must be such that all the three systems (2.33e-g), (2.34e-g) and (2.35e-g), are uniquely solvable with \( \nu \neq 0 \). Finally recall that we must have rank\( (F_5(A)) = 8 \).
Chapter 2. Continuous explicit Runge-Kutta methods

A particular class of these methods is obtained by imposing $c_3 \neq 0$ and the conditions (2.30a-d) and (2.31a-c) together with their counterparts for the last two stages

$$2\phi_{47} = c_7^2,$$  \hspace{1cm} (2.30c)

$$2\phi_{48} = c_8^2,$$  \hspace{1cm} (2.30f)

$$2\tau_2c_2c_3 + 3(a_7c_3^2 + a_4c_4^2 + a_7c_5^2 + a_7c_6^2) = c_7^3,$$  \hspace{1cm} (2.31d)

$$2a_2c_2c_3 + 3(a_8c_3^2 + a_4c_4^2 + a_8c_5^2 + a_5c_6^2 + a_7c_7^2) = c_8^3.$$  \hspace{1cm} (2.31e)

With this choice the system (2.35e-g) has the unique solution $\lambda = c_2^2$, $\mu = 0$, $\nu = -4$. It follows that $\rho = 0$ and that the conditions (2.35a-d) are automatically satisfied. The system (2.33e-g) has the unique solution $\lambda = c_2(c_2 - c_3)$, $\mu = c_3$, $\nu = -2c_2^c + 3c_3^c$, implying $\rho = 8c_3^c - 2c_4^c$. Finally, the system (2.34e-g) has the unique solution $\lambda = c_2^c(2c_4^c + 3c_6^c - 2c_3^c - 2c_7^c)$, $\mu = \frac{2c_3^c - 2c_4^c + 3c_6^c}{2c_3^c}$, $\nu = -\frac{2c_3^c + 3c_6^c}{c_3^c}$ implying $\rho = 6c_3^c$. It follows that we must impose $c_4 \neq 0$ and $c_4 + 3c_3 \neq 0$. Long and tedious calculations show that conditions (2.30c), (2.31b), (2.33a) and (2.34a) can be satisfied if and only if we choose $c_5 = c_4 = 2c_3$. This leads to $\nu = -5$ and $\rho = -4$ in (2.33a-g) and to $\lambda = \frac{c_2^c(5c_3^c + 2c_4^c - 2c_3^c)}{2c_3^c}$, $\mu = \frac{2c_3^c - 5c_3^c}{2c_3^c}$, $\nu = -5$ and $\rho = 6$ in (2.33a-g). On the other hand, this causes the conditions for $a_{52}$, $a_{53}$ and $a_{54}$ imposed by (2.33a) and (2.34a) to depend linearly on (2.30c) and (2.31b). So, like for $p = 4$, $a_{54}$ is a free parameter. However for the matrix $A$ to be 5-minimal, we must require $a_{54} \neq 0$ and $3c_2 - 2c_3 \neq 0$. In conclusion, we can choose arbitrary $c_2, c_3, a_{54}$ subject to the restrictions $c_2 \neq 0$, $c_3 \neq 0$, $3c_2 - 2c_3 \neq 0$ and $a_{54} \neq 0$. By using $c_5 = c_4 = 2c_3$ in (2.32a-c) we get

$$a_{32} = \frac{c_3^2}{2c_2},$$  \hspace{1cm} (2.36a)
2.2. Order barriers

\[ a_{42} = \frac{2c_3^2}{c_2}, \quad a_{43} = 4c_3, \]  
\[ a_{52} = \frac{2c_3(3a_{54} - c_3)}{c_2}, \quad a_{53} = 4c_3 - 8a_{54}. \]  
\[ (2.36b) \]

Furthermore, we can choose arbitrary \( c_6 \), subject to the only restrictions \( c_6 \neq c_3 \) and \( c_6 \neq 2c_3 \), and we get the following solution of the \( 4 \times 4 \)-system of linear equations derived from (2.30d), (2.31c), (2.33b) and (2.34b)

\[ a_{62} = \frac{c_2^2(2c_3 - a_6)}{2c_2c_3} \quad a_{63} = \frac{c_2^2(a_6 - c_2)}{3c_2^2} \]
\[ a_{64} = \frac{c_2^2(a_6 - c_2)(2c_3 + a_{54} - a_6)}{12a_{54}c_2^2} \quad a_{65} = \frac{c_2^2(a_6 - c_2)(c_6 - 2c_3)}{12a_{54}c_2^2} \]  
\[ (2.36d) \]

Finally, we can choose arbitrary \( c_7, a_{76}, a_8, a_{86}, a_{87} \) (apart from combinations leading to \( A \) not being 5-minimal), and solve the two linear \( 4 \times 4 \) systems derived from (2.30e-f), (2.31d-e), (2.33c-d) and (2.34c-d) for the remaining coefficients \( a_{72}, \ldots, a_{75} \) and \( a_{82}, \ldots, a_{85} \). Their general expressions are quite complicated, so we prefer to present the Butcher tableau \( c|A \) as an example of such methods together with the continuous weights \( b_3(\theta) \). The choices for the free parameters are not motivated by stability considerations nor error constant minimization, we have just aimed for obtaining simple coefficients.
Example 2.13 CERK method of order 5 with 8 stages.

\[
\begin{array}{cccccc}
0 & \frac{1}{4} & \frac{1}{4} \\
\frac{1}{4} & \frac{1}{8} & \frac{1}{8} \\
\frac{1}{2} & 0 & -\frac{1}{2} & 1 \\
\frac{1}{2} & \frac{1}{12} & 0 & \frac{1}{3} & \frac{1}{12} \\
\frac{1}{4} & 0 & -\frac{9}{5} & \frac{3}{2} & -\frac{1}{4} & \frac{9}{8} \\
1 & 0 & \frac{4}{5} & 0 & -\frac{3}{5} & 0 & \frac{4}{5} \\
1 & \frac{1}{6} & 0 & 0 & \frac{4}{15} & \frac{2}{5} & 0 & \frac{1}{6}
\end{array}
\]

\[
b_1(\theta) = \frac{32}{15}\theta^5 - \frac{20}{3}\theta^4 + \frac{70}{9}\theta^3 - \frac{25}{6}\theta^2 + \theta
\]

\[
b_2(\theta) = 0
\]

\[
b_3(\theta) = -\frac{128}{15}\theta^5 + 240\theta^4 - \frac{208}{9}\theta^3 + 8\theta^2
\]

\[
b_4(\theta) = \frac{32}{5}\theta^5 - 120\theta^4 + \frac{15}{3}\theta^3
\]

\[
b_5(\theta) = \frac{32}{5}\theta^5 - 200\theta^4 + 200\theta^3 - 6\theta^2
\]

\[
b_6(\theta) = -\frac{128}{15}\theta^5 + \frac{25}{3}\theta^4 - \frac{112}{9}\theta^3 + \frac{8}{3}\theta^2
\]

\[
b_7(\theta) = -\frac{20}{3}\theta^5 + 15\theta^4 - \frac{25}{9}\theta^3 + \frac{5}{2}\theta^2
\]

\[
b_8(\theta) = \frac{44}{5}\theta^5 - 19\theta^4 + 13\theta^3 - 3\theta^2
\]

We close this section by considering the minimal number of stages that must be added to a discrete Runge-Kutta method to extend it to a CERK method of the same order. In view of the theory of this section the following result is easy to prove, but could be useful anyway.
2.3. Practical questions

**Theorem 2.14** Let the \( s \times s \)-matrix \( A \) define a discrete explicit Runge-Kutta method of order \( p \). Consider a continuous extension of this method with \( \hat{s} \) stages and uniform order \( p \). Then

\[
\hat{s} - s \geq \delta
\]

where \( \delta = \text{rank}(G_p(A)) - \text{rank}(F_p(A)) \).

**Proof:** Let the \( \hat{s} \times \hat{s} \)-matrix \( \hat{A} \) define the extended method. Since the first \( s \) columns of \( F_p(A) \) and \( F_p(\hat{A}) \) are identical it follows that \( \text{rank}(F_p(\hat{A})) - \text{rank}(F_p(A)) \leq \hat{s} - s \). On the other hand, the \( \hat{s} - s \) additional stages cannot decrease the rank of \( G_p(A) \) such that we must have \( \text{rank}(G_p(\hat{A})) \geq \text{rank}(G_p(A)) \). So using Proposition 2.2 we have

\[
\delta = \text{rank}(G_p(A)) - \text{rank}(F_p(A)) \leq \text{rank}(G_p(\hat{A})) - \text{rank}(F_p(\hat{A})) = \text{rank}(F_p(\hat{A})) - \text{rank}(F_p(A)) \leq \hat{s} - s.
\]

and the proof is complete. \( \square \)

In Section 2.1 we supplied the discrete Dormand-Prince method with two extra stages in order to obtain a 5th order continuous method. It was also pointed out that at least one extra stage was required. In view of the theorem above we now have.

**Corollary 2.15** There exist no 5th order continuous extensions of the Dormand-Prince \((5,4)\) method with only one extra stage, i.e. with 8 stages.

2.3 Practical questions

In this section we shall discuss some of the requirements one might wish to impose on a CERK method. As is usual for discrete methods we wish to construct embedded pairs of formulae in order to perform stepsize control. This feature is supposed to work in exactly the same way as for discrete methods, i.e. we want to compare the result of our CERK method at \( \theta = 1 \)
with an auxiliary discrete method of one order less or higher. We shall always assume that we use the CERK method to proceed integration with since we at least shall require global continuity of our uniform approximation. We denote by CERK(p, q) a pair of methods of order p and q respectively, where we proceed integration with the main method of order p, and the auxiliary method of order q is used for stepsize control. It should be pointed out that our CERK methods can be used with the new error control strategies suggested by Enright [33, 36] and further investigated by Higham [44, 45]. These strategies require that a continuous auxiliary method is available. For the 5th order methods of the previous section such an additional method can be easily obtained since, by assumption, the first 6 stages define a 4th order CERK method.

In the literature (e.g. [31]) we frequently find the term FSAL used for some specific Runge-Kutta pairs. This refers to the function evaluation \( f(x_n, y_n) \) which will always be the stage \( K_1 \) in the step from \( x_n \) to \( x_{n+1} \). It has been pointed out that this stage may also be used in the step from \( x_{n-1} \) to \( x_n \). Since the method is explicit this stage can not be a part of the main approximation, but it is sometimes used in the auxiliary method like in the Dormand-Prince (5,4) case of Section 2.1. Such pairs are called FSAL. We will consider this property from a slightly different point of view. Instead of regarding this function evaluation as an extra available stage, we shall include it in the construction of our CERK methods. To this end we impose the conditions

\[
c_s = 1 \quad \text{and} \quad a_{sj} = b_j(1), \ j = 1,\ldots, s,
\]

and name the property \textit{stage reuse}. In the traditional setting one would say that we use the FSAL evaluation for construction of the interpolant. We shall see that these \textit{stage reuse conditions} relates closely to global \( C^1 \)-continuity of the continuous approximation. It is proved by Dormand and Prince [30] that this is achieved if one puts \( y_1 = u(x_0 + h) \) and

\[
b_i(0) = \delta_{i1}, \quad b_i(1) + b'_i(1) = \delta_{is}
\]

where \( \delta_{ij} = 1 \) if \( i = j \) and \( \delta_{ij} = 0 \) otherwise.
2.3. Practical questions

Also the principal error function will be given some attention. For a CERK method of order \( p \), the local truncation error is given by

\[
\|y(x_0 + \theta h) - u(x_0 + \theta h)\| = \frac{h^{p+1}}{(p+1)!} \sum_{i=N_p+1}^{N_{p+1}} e_i(\theta) \alpha(t_i) F(t_i) + O(h^{p+2})
\]

where \( F(t) \) is the elementary differential corresponding to the tree \( t \), \( \alpha(t) \) is a weight corresponding to the tree \( t \) and \( e_i(\theta) \), \( i = N_p + 1, \ldots, N_{p+1} \) are the error polynomials given by

\[
e_i(\theta) = \theta^{\nu(t_i)} - \gamma(t_i) \sum_{j=1}^{s} \phi_{ij} b_j(\theta), \quad i = N_p + 1, \ldots, N_{p+1} \quad (2.37)
\]

Obviously we are interested in making these error polynomials small in some sense. The literature is not unified with regard to what norm that should be used on this \( n_{p+1} \)-vector of polynomials. Calvo et al. [21] minimize the quantity

\[
g^* = \int_0^1 g(\theta) d\theta
\]

over the free parameters, where

\[
g(\theta) = \sqrt{\frac{\sum_{i=N_p+1}^{N_{p+1}} [e_i(\theta) \alpha(t_i)]^2}{\sum_{i=N_p+1}^{N_{p+1}} [e_i(1) \alpha(t_i)]^2}}. \quad (2.38)
\]

The denominator of (2.38) is nothing but the error measure used by Dormand and Prince [29] when they constructed their famous discrete \((5,4)\) pair. Enright et al. [34] use the max-norm and they do not scale the error polynomial \( e_i(\theta) \) with the weight \( \alpha(t_i) \) i.e. they consider

\[
\max_{N_p+1 \leq i \leq N_{p+1}} \left\{ \max_{\theta \in [0,1]} |e_i(\theta)| \right\}.
\]

In fact, for the purpose of handling discontinuities they also let \( \theta \) range over \([0,2]\) in the inner maximum. Apart from Appendix A where we apply this
latter error measure, we do not attempt to minimize the error in our CERK methods. Perhaps, in the continuous case, it is even more important that the error polynomials are monotonic for $0 < \theta < 1$. This prevents the error from blowing up between the nodal points so that, in some sense, the error estimate is valid also in the interior of the step.

### 2.3.1 Some general results

**Lemma 2.16** Let $A \in \mathcal{M}$ define a CERK method with stage reuse. Then $\Phi := F_{p+1}(A) := ((\phi_{i,j}))$ is such that

$$\phi_{i,s} = \frac{\rho(t_i)}{\gamma(t_i)}, \quad i = 1, \ldots, N_{p+1}.$$  

**Proof:** The lemma is proved by induction on the row index. Obviously the result holds for $i = 1$. Then assume that the lemma is true for all $i$ such that $i \leq n - 1$. Now like in Theorem 2.5 the $n$th condition corresponds to the tree $t_n$ which either has the form $[t_{n'}]$ for some tree $t_{n'}$ of order $\rho(t_n) - 1$ or the form $[t_{v_1}, \ldots, t_{v_u}]$ for $u(\geq 2)$ trees $t_{v_i}$ where $1 \leq \rho(t_{v_i}) \leq \rho(t_n) - 2$ and $\rho(t_n) = 1 + \sum_{i=1}^{u} \rho(t_{v_i})$. In the latter case with $t_{n_i} = [t_{v_i}]$ we have

$$\gamma(t_n) = \rho(t_n) \prod_{i=1}^{u} \frac{\gamma(t_{n_i})}{\rho(t_{n_i})}$$

such that

$$\phi_{n,s} = \prod_{i=1}^{u} \phi_{n_i,s} = \prod_{i=1}^{u} \frac{\rho(t_{n_i})}{\gamma(t_{n_i})} = \frac{\rho(t_n)}{\gamma(t_n)}.$$  

In the former case we get

$$\phi_{n,s} = \sum_{j=1}^{s-1} a_{sj} \phi_{n',j} = \sum_{j=1}^{s-1} b_{j}(1) \phi_{n',j} = 1/\gamma(t_n'),$$

where we have applied the stage reuse conditions along with the order conditions at $\theta = 1$. Since the last equation is an $A$-transformation we have
1/γ(t_n) = ρ(t_n)/γ(t_n) and we complete the proof by remarking that the induction works for all n such that ρ(t_n) ≤ p + 1. □

**Theorem 2.17** Let A ∈ M_p^n be a CERK method with stage reuse. Then the global continuous approximation is continuously differentiable.

**Proof:** It is sufficient to prove that u'(x_0) = K_1 and that u'(x_0 + h) = K_a. Since b_2'(0) = · · · = b_a'(0) = 0 and b_1'(0) = 1 by the first order condition we must have u'(x_0) = K_1. By Lemma 2.16 the last column of F_p(A) is equal to the right hand side of (2.14) evaluated at θ = 1. Since A is p-minimal it follows that z_a(1) = 1 and z_1(1) = · · · = z_{a-1}(1) = 0 such that u'(x_0 + h) = K_a. □

**Proposition 2.18** Let A ∈ M_p^n be a CERK method with stage reuse. Then all the error polynomials e_i(θ), i = N_p + 1, . . . , N_p+1 have a stationary point at θ = 1.

**Proof:** The derivatives of the error polynomials are given by

\[ e'_i(θ) = ρ(t_i)θ^{p(t_i)-1} - γ(t_i) \sum_{j=1}^{s} φ_{ij}z_j(θ), \quad i = N_p + 1, . . . , N_p+1 \]

As above we must have z_a(1) = 1 and z_1(1) = · · · = z_{a-1}(1) = 0 and by Lemma 2.16 it follows that e'_i(1) = 0. □

**Proposition 2.19** There exist no CERK(p,p+1) pairs with s = CEN(p) stages.

**Proof:** Assume that s = CEN(p) for a method given by the s × s-matrix A. Then since rank(F_p(A)) = s it is impossible to find two distinct sets of weights that satisfy the first N_p discrete order conditions. □

### 2.3.2 Stage reuse for optimal CERK methods of order p=3-5 and monotonicity of the error function

We consider CERK methods with CEN(p) stages and see when stage reuse can be imposed.
Order p=3

It is easy to see that, given a 3-stage Runge-Kutta method of order 3

\[
\begin{array}{c|ccc}
0 & c_2 & c_2 \\
0 \neq c_2 & c_3 - a_{32} & a_{32} \\
& b_1 & b_2 & b_3 \neq 0 \\
\end{array}
\]

(2.39)

we can always extend it to a 4-stage CERK method of order 3 with stage reuse by following [42, p. 177] Example 5.2. The reusable stage is

\[
K_4 = f(t + h, y_0 + h(b_1K_1 + b_2K_2 + b_3K_3))
\]

and the continuous weights are

\[
\begin{align*}
b_1(\theta) &= (1 - 2b_1)\theta^2 + (3b_1 - 2)\theta + \theta \\
b_2(\theta) &= -b_2(2\theta^3 - 3\theta^2) \\
b_3(\theta) &= -b_3(2\theta^3 - 3\theta^2) \\
b_4(\theta) &= \theta^3 - \theta^2
\end{align*}
\]

(2.40)

Observe that this continuous extensions of the 3-stage discrete method above is nothing but the Hermite interpolant at the endpoints \( t \) and \( t+h \). Remark also that the matrix \( F_3(A) \) is always non-singular.

There are 4 error polynomials of the kind (2.37) corresponding to the trees \( t_5, \ldots, t_8 \). Using the two conditions

\[
\begin{align*}
b_2c_2 + b_3c_3 &= \frac{1}{7} \\
b_2c_2^2 + b_3c_3^2 &= \frac{1}{3}
\end{align*}
\]
for the underlying discrete method implies

\[ a_{42} = b_2 = \frac{3c_2 - 2}{6c_2(c_2 - c_3)} \]
\[ a_{43} = b_3 = -\frac{3c_2 - 2}{6c_3(c_3 - c_2)} \]

while the two remaining conditions gives

\[ a_{41} = b_1 = \frac{6c_2 - 3(c_2 + c_3) + 2}{6c_2(c_2 - c_3)} \]
\[ a_{32} = b_2 = -\frac{3c_2 - 2}{6c_3(c_3 - c_2)} \]

Thus, by (2.40) and (2.37) we get after some tedious algebra

\[ e_5(\theta) = \theta^4 - \frac{4}{5}[3c_2c_3 - 2(c_2 + c_3) + 3]\theta^3 + 2[3c_2c_3 - 2(c_2 + c_3) + 2]\theta^2 \]
\[ e_6(\theta) = \theta^4 - 4(1 - \frac{2}{3}c_3)\theta^3 + 4(1 - c_3)\theta^2 \]
\[ e_7(\theta) = \theta^4 - 4(1 - 2c_2)\theta^3 + 2(2 - 3c_2)\theta^2 \]
\[ e_8(\theta) = \theta^2(\theta - 2)^2 \]

The derivative of all these polynomials must clearly vanish at \( \theta = 0 \) and by Proposition 2.18 at \( \theta = 1 \). In addition, \( e'_5(\theta) \) is zero at \( \theta = 3c_2c_3 - 2(c_2 + c_3) + 2 \), while \( e'_6(\theta) \) and \( e'_7(\theta) \) vanish at \( \theta = 2 - 2c_3 \) and \( 2 - 3c_2 \) respectively. So for \( e_6(\theta) \) and \( e_7(\theta) \) to be increasing in \((0, 1)\) we must have

\[ c_2 \leq \frac{1}{3} \quad \text{and} \quad c_3 \leq \frac{1}{2} \]

With \( c_2 \leq \frac{1}{3} \), \( e_6(\theta) \) will be increasing in \((0, 1)\) if

\[ c_3 \leq \frac{2c_2 - 1}{3c_2 - 2} \]

Notice that the function \( f(x) = (2x - 1)/(3x - 2) \) is monotone and decreasing from \( \frac{1}{2} \) to \( \frac{1}{3} \) over the interval \([0, 1/3]\). Obviously \( e_8(\theta) \) is unconditionally increasing in \((0, 1)\).
Chapter 2. Continuous explicit Runge-Kutta methods

Order p=4

Now, turning to order 4 we consider 6-stage CERK methods with stage reuse belonging to the class satisfying (2.30a-d) and (2.31a-c). Let \( b_i := b_i(1) \) such that \( b_0 = 0 \). Since the simplifying assumptions yield \( b_2(\theta) \equiv 0 \) we also have \( b_2 = 0 \). Because \( b_0 = 0 \) the first 5 stages define a discrete Runge-Kutta method of order 4. Hence, the primary conditions

\[

c_3 b_3 + c_4 b_4 + c_5 b_5 = \frac{1}{2} \\
c_3^2 b_3 + c_4^2 b_4 + c_5^2 b_5 = \frac{1}{3} \\
c_3^3 b_3 + c_4^3 b_4 + c_5^3 b_5 = \frac{1}{4}
\]

and the secondary condition

\[
b_4 a_{43} \phi_{43} + b_5 (a_{53} \phi_{43} + a_{54} \phi_{44}) = \frac{1}{24}
\]

must be satisfied. Now the simplifying assumptions yield

\[
b_4 a_{43} c_3^2 + b_5 (a_{53} c_3^2 + a_{54} c_4^2) = \frac{1}{12}
\]

so by (2.32b-c) we get

\[
b_4 (c_4 - c_3) c_4^2 + b_5 [(c_5 - c_3) c_5^2 - a_{54} c_4 (3c_4 - 2c_3) + a_{54} c_4^2] = \frac{1}{12}
\]

which in conjunction with (2.41) gives

\[
12 b_5 a_{54} c_4 (c_4 - c_3) = 1 - 2c_3
\]

Vice versa, we now find a procedure for constructing 6-stage CERK methods of order 4 with stage reuse. The main result is the following
Theorem 2.20 Let \( c_2 \neq 0, c_3 \neq 0, c_4 \neq 0, c_5 \neq 0 \) with \( c_3, c_4, c_5 \) distinct be given and consider 6-stage CERK methods of order 4 with stage reuse satisfying (2.30a-d) and (2.31a-c). Then if \( 6c_3c_4 - 4(c_3 + c_4) + 3 \neq 0 \) a method is uniquely determined. If \( c_3 = \frac{1}{2} \), \( c_4 = 1 \) then there exists a one-parameter family of methods (with \( a_{54} \) arbitrary).

\textbf{Proof:} By hypothesis the system

\[
\begin{align*}
    c_3 X_3 + X_4 b_4 + X_5 b_5 &= \frac{1}{2} \\
    c_3^2 X_3 + c_4^2 X_4 + c_5^2 X_5 &= \frac{1}{3} \\
    c_3^3 b_3 + c_4^3 X_4 + c_5^3 X_5 &= \frac{1}{4}
\end{align*}
\]

has a unique solution \((X_3, X_4, X_5)\). The coefficient \( a_{54} \) is determined by

\[ 12X_5a_{54}c_4(c_4 - c_3) = 1 - 2c_3 \tag{2.42} \]

In any case this condition can be satisfied if \( X_5 \neq 0 \) which is equivalent to

\[ 6c_3c_4 - 4(c_3 + c_4) + 3 \neq 0 \]

If \( X_5 = 0 \) (2.42) can only be satisfied if \( c_3 = \frac{1}{2} \), implying \( c_4 = 1 \). But in this case \( a_{54} \) is arbitrary. Now let \( a_{32}, a_{42}, a_{43}, a_{52}, a_{53} \) by means of conditions (2.32a-c) and we set \( a_{61} := 1 - X_3 - X_4 - X_5, a_{62} := 0, a_{63} := X_3, a_{64} := X_4, a_{65} := X_5 \). One may easily check that the conditions (2.32d) are satisfied such that the \( 6 \times 6 \)-matrix \( A \) determines a CERK method of order 4. Now since \( A \) is 4-minimal it follows that \( b_i(1) = b_i = a_{6i}, \) \( i = 1, \ldots, 5, b_6 = 0 \) so indeed we have stage reuse. \( \square \)

The question of monotonicity of the error polynomials for 6 stage CERK methods of order 4 with stage reuse is of course more complicated than the corresponding case of order 3. There are originally 9 such polynomials, corresponding to the 9 trees of order 5, but some of them turns out to be identical due to the conditions already imposed on the coefficients of the method. We have not proved whether it is possible to construct methods
where all these polynomials are monotonic, but if so, it seems that the restrictions one must impose on the remaining parameters are rather strong. We illustrate this by considering the error polynomial, \( e_9(\theta) \), related to the primary condition of order 5. Just by using the primary conditions for \( p = 1, \ldots, 4 \) along with \( c_5 = 1 \) we find that

\[
e'_9(\theta) = 5(P(\theta) - P(1)b'_9(\theta))
\]

where

\[
P(\theta) = \theta(\theta - c_3)(\theta - c_4)(\theta - c_5).
\]

Long but straightforward computations lead to

\[
b'_9(\theta) = \theta[3\theta - 2 + 2a(\theta - 1)(2\theta - 1)], \quad a = \frac{-3c_3c_4 - c_3 + 2c_4}{6c_3c_4 - 6c_3c_4 + 2c_4}
\]

Thus, for \( e_9(\theta) \) to be monotonically increasing for \( \theta \in (0, 1) \) it is necessary that \( e_9'(0) > 0 \) and \( e_9''(1) < 0 \) since \( e_9'(0) = e_9'(1) = 0 \). To achieve this, with \( c_3, c_4, c_5 \in (0, 1) \) one may check that the inequalities

\[
c_4 < \frac{c_3}{6c_3^2 - 6c_3 + 2}, \quad \frac{1}{1 - c_3} + \frac{1}{1 - c_4} + \frac{1}{1 - c_5} < 5
\]

must hold, restricting the size of \( c_3, c_4, c_5 \).

We complete this discussion by giving an example of a 6-stage CERK method of order 4 with stage reuse where \( c_3 = \frac{1}{2} \) and \( c_4 = 1 \).
Example 2.21 A 6-stage CERK method of order 4 with stage reuse.

\[
\begin{array}{c|cccc}
0 & & b_1(\theta) &=& \theta\left(-\frac{2}{3}\theta^3 + 2\theta^2 - \frac{12}{6}\theta + 1\right) \\
1/2 & 1/2 & b_2(\theta) &=& 0 \\
1/2 & 1/4 & 1/4 & b_3(\theta) &=& \theta^2\left(4\theta^2 - \frac{28}{9}\theta + 6\right) \\
1 & 0 & -1 & 2 & b_4(\theta) &=& \theta^2\left(\theta^2 - \frac{7}{2}\theta + \frac{5}{2}\right) \\
3/4 & 3/16 & 0 & 9/16 & 0 & b_5(\theta) &=& \theta^2\left(-\frac{16}{9}\theta^2 + \frac{32}{9}\theta - \frac{16}{9}\right) \\
1 & 1/6 & 0 & 2/3 & 1/6 & 0 & b_6(\theta) &=& \theta^3\left(\theta - 1\right)
\end{array}
\]

Order \(p=5\)

For the case \(p = 5\) we consider the class of 8-stage CERK methods derived in the previous section. In addition to the conditions we impose there, we require

\[
\begin{align*}
as_{ij} &= b_j(1), & j &= 1, \ldots, 7, \\
b_8(1) &= 0, \\
c_8 &= 1.
\end{align*}
\]

(2.43)

for the last stage. Now, first observe that the simplifying assumptions imply \(b_2(\theta) \equiv 0\) such that \(a_{82} = 0\). Next, consider the conditions to be satisfied by \(b_j(1), \ j = 1, \ldots, 8.\)

\[
\sum_{j=1}^{8} \phi_j(t_{i_k})b_j(1) = \frac{1}{\gamma(t_{i_k})}, \ k = 1, \ldots, 8
\]

(2.44)

where \(i_1, \ldots, i_8\) are the indices of the rows of \(F_5(A)\) that are chosen to form basis for the row space of \(F_5(A)\). It is quite easy to see that it is sufficient
to impose the following 6 conditions for the last stage:

\[
\begin{align*}
    c_3 a_{83} + 2 c_3 a_{84} + 2 c_3 a_{85} + c_6 a_{86} + c_7 a_{87} &= \frac{1}{2} \\
    c_3^2 a_{83} + 4 c_3^2 a_{84} + 4 c_3^2 a_{85} + c_6^2 a_{86} + c_7^2 a_{87} &= \frac{1}{3} \\
    c_3^3 a_{83} + 8 c_3^3 a_{84} + 8 c_3^3 a_{85} + c_6^3 a_{86} + c_7^3 a_{87} &= \frac{1}{4} \\
    c_3^4 a_{83} + 16 c_3^4 a_{84} + 16 c_3^4 a_{85} + c_6^4 a_{86} + c_7^4 a_{87} &= \frac{1}{5} \\
    2 c_3^3 a_{84} + 2(c_3 - a_{54}) c_3^2 a_{85} + \frac{1}{3} c_6^2(c_6 - c_3) a_{86} + \phi_8 \tau a_{87} &= \frac{1}{24} \\
    2 a_{54} c_3^3 a_{85} + \phi_{16} a_{54} a_{86} + \phi_{16} \tau a_{87} &= \frac{1}{120}
\end{align*}
\]

(2.45)

where

\[
\begin{align*}
    \phi_8 \tau &= (\frac{1}{2} a_{73} + 2 a_{74} + 2 a_{75}) c_3^2 + \frac{1}{6} a_{76} c_6^2 \\
    \phi_{16} a &= -\frac{1}{6} c_6^2 (c_6 - c_3)(c_6 - 2c_3) \\
    \phi_{16} \tau &= 2(a_{14} a_3 + a_{75}(c_3 - a_{54})) c_3^2 + \frac{1}{3} a_{76} c_6^2 (c_6 - c_3)
\end{align*}
\]

Just by considering the first 4 equations of the linear system (2.45), it follows that

\[
\begin{align*}
    a_{86} &= -\frac{2c_3^2 - \frac{3}{4} c_3 + \frac{1}{6} - c_7(c_3 - \frac{1}{2})^2}{c_6(c_6 - c_3)(c_6 - 2c_3)(c_7 - c_6)} \\
    a_{87} &= \frac{2c_3^2 - \frac{3}{4} c_3 + \frac{1}{6} - c_6(c_3 - \frac{1}{2})^2}{c_7(c_7 - c_3)(c_7 - 2c_3)(c_7 - c_6)}
\end{align*}
\]

(2.46a, 2.46b)

where we must require \( c_7 \neq c_3, c_7 \neq 2c_3, c_6 \neq c_3, c_6 \neq 2c_3 \) and \( c_7 \neq c_6 \).

Now since (2.45) is an overdetermined linear system, we must impose conditions on the coefficients to ensure the existence of a solution. By eliminating \( a_{84} \) from the 5th equation, using the preceding equations and then eliminating \( a_{85} \) using the 6th equation, we can substitute the above expressions for
2.3. Practical questions

\( a_{86} \) and \( a_{87} \) in order to obtain a necessary and sufficient condition for the existence of a solution of (2.45). It turns out that a solution exists if and only if

\[
a_{76} = \frac{N_1 \times N_2}{D_1 \times D_2}
\]

(2.47a)

with

\[
N_1 = c_7 \left( \frac{1}{2} c_3^3 - \frac{2}{9} c_3^2 - \frac{1}{12} c_3 + \frac{1}{24} \right) - \frac{1}{6} c_6 c_7 (c_3 - \frac{1}{2})^2
\]

\[
- c_6 \left( \frac{1}{6} c_3^2 - \frac{1}{12} c_3 + \frac{1}{120} \right) - c_7 \left( \frac{1}{3} c_3^2 - \frac{5}{8} c_3 + \frac{1}{10} \right),
\]

(2.47b)

\[
N_2 = c_7 (c_7 - c_3) (c_7 - 2c_3)
\]

(2.47c)

\[
D_1 = \frac{5}{9} c_3^3 - \frac{5}{8} c_3 + \frac{1}{6} - \frac{5}{6} c_6 (c_3 - \frac{1}{2})^2
\]

(2.47d)

\[
D_2 = c_6 (c_6 - c_3) (c_6 - 2c_3)
\]

(2.47e)

Note that we must have \( D_1 \neq 0 \).

Summarizing, 8-stage 5th order CERK methods with stage reuse can be constructed in the following way:

- Choose \( c_2, c_3, a_{54} \) and \( c_6 \) due to the given restrictions.
- Put \( c_5 = c_4 = 2c_3 \) and compute \( a_{32}, a_{42}, a_{43}, a_{52}, a_{53}, a_{62}, a_{63}, a_{64} \) and \( a_{65} \) using the formulas (2.36a-d).
- Choose arbitrary \( c_7 \) and put \( c_8 = 1 \).
- Compute \( a_{76} \) from (2.47a-e) and \( a_{86} \) and \( a_{87} \) from (2.46a-b).
- Compute \( a_{72}, \ldots, a_{75}, a_{82}, \ldots, a_{85} \), from the two linear 4 × 4 systems derived from (2.30e-f), (2.31d-e), (2.33c-d) and (2.34c-d).
We modify the last two stages of the method given by Example 2.13 to obtain stage reuse.

**Example 2.22** CERK method of order 5 with 8 stages and stage reuse. The weights in the bottom line of the Butcher tableau ($\hat{y}_{n+1}$) define a fourth order method intended for error estimation.

\[
\begin{array}{c|cccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
\frac{1}{4} & 0 & \frac{1}{4} & & & & \\
\frac{1}{4} & 0 & \frac{1}{8} & \frac{1}{8} & & & \\
\frac{1}{2} & 0 & -\frac{1}{2} & 1 & & & \\
\frac{3}{4} & 0 & -\frac{9}{8} & \frac{3}{2} & -\frac{3}{4} & \frac{9}{8} & \\
1 & 0 & \frac{20}{7} & -\frac{12}{7} & \frac{9}{7} & -\frac{18}{7} & \frac{8}{7} \\
1 & \frac{7}{50} & 0 & \frac{16}{45} & -\frac{4}{15} & \frac{2}{5} & \frac{16}{45} & \frac{7}{50} \\
\end{array}
\]

\[
\hat{y}_{n+1} = 0, 0, \frac{2}{3}, -\frac{4}{3}, 1, \frac{2}{3}, 0, 0
\]

where:

- \( b_1(\theta) = \frac{32}{15} \theta^5 - \frac{20}{3} \theta^4 + \frac{70}{9} \theta^3 - \frac{25}{6} \theta^2 + \theta \)
- \( b_2(\theta) = 0 \)
- \( b_3(\theta) = -\frac{120}{15} \theta^5 + 24 \theta^4 - \frac{208}{9} \theta^3 + 8 \theta^2 \)
- \( b_4(\theta) = \frac{32}{5} \theta^5 - 120 \theta^4 + \frac{15}{3} \theta^3 \)
- \( b_5(\theta) = \frac{32}{5} \theta^5 - 200 \theta^4 + 200 \theta^3 - 60 \theta^2 \)
- \( b_6(\theta) = -\frac{128}{15} \theta^5 + \frac{20}{3} \theta^4 - \frac{112}{9} \theta^3 + \frac{8}{3} \theta^2 \)
- \( b_7(\theta) = -\frac{25}{15} \theta^5 + \frac{25}{5} \theta^4 - \frac{133}{45} \theta^3 + \frac{7}{10} \theta^2 \)
- \( b_8(\theta) = 4 \theta^5 - \frac{41}{5} \theta^4 + \frac{27}{5} \theta^3 - \frac{6}{5} \theta^2 \)
Notice that the auxiliary method only uses the first six stages of the fifth order method. In fact, this method is nothing but the 4th order CERK method defined by the first 6 stages evaluated at $\theta = 1$. 
Chapter 3

Stability of explicit Runge-Kutta methods

In this chapter we shall examine stability properties of CERK methods. As pointed out, one can apply these methods both to ordinary and delay differential equations. The former case is studied in Section 3.1 where we consider the largest disk that can be contained in the stability region of an explicit Runge-Kutta method. This material is taken from Owren and Seip [65]. Section 3.2 gives an indication of how some results by Zennaro [75] on P-stability of Runge-Kutta methods can be applied to our CERK methods.

3.1 The largest disk of stability.

3.1.1 Introduction

It is well-known that an explicit s-stage Runge-Kutta method applied to the linear test equation, \( y' = \lambda y, \quad \lambda \in C \), results in a recursion formula of the form \( y_{n+1} = P(\lambda h)y_n \) where \( P(\zeta) \) is a polynomial of degree \( s \) and \( h \) is the step size. The region of absolute stability \( S_P \) for such a method is given by

\[
S_P = \{ \zeta \in C : |P(\zeta)| \leq 1 \}.
\]
When designing a Runge-Kutta method one will usually have some freedom in choosing the coefficients of the method. One might wish to utilize this freedom to optimize $S_P$ in some sense. It is impossible to find an optimization criterion that would be appropriate in general, since the desired shape of the stability region will depend on the problem at hand. In this section we consider the largest disk centered at $(-r, 0)$ with radius $r$ that can be contained in the stability region of the method. In doing this, we compromise between stretching the stability region in the real and in the imaginary direction, and in addition the problem is relatively simple to analyze.

This idea was introduced by Jeltsch and Nevanlinna [50]. They proved that the closed disk $|\zeta + r| \leq r$ can be contained in the stability region of a consistent $s$-stage explicit Runge-Kutta method if and only if $r \leq s$. This largest disk is obtained only if the stability polynomial is given by

$$P(\zeta) = (1 + \frac{\zeta}{s})^s. \quad (3.1)$$

As pointed out in [51], this can be viewed as a simple consequence of Bernstein’s inequality [22, p.91] in conjunction with the following result of [51]: $S_P \not\subset S_Q$ whenever $P \neq Q$ and the two corresponding methods have the same number of stages.

The above result is clearly of great theoretical value. But it does not provide much information for the most commonly used Runge-Kutta methods, since (3.1) can only be the stability polynomial of an $s$-stage first order method. Thus it would be of interest to know the corresponding result under the additional requirement that the method be of order $p > 1$.

We shall see how the proof of [50] can be modified to cover the case $p = 2$. We also intend to illustrate to what extent the technique of [50] is applicable in the general case. This approach leads to a study of polynomials closely related to the generalized Bessel polynomials yielding upper bounds for the optimal radii. For some special cases we provide numerical values for the optimal radii and for the corresponding stability polynomial.
We close this introduction by stating the problem in a precise manner. Let $\mathcal{P}_{s,p}$ be the class of polynomials

$$P(\zeta) = \sum_{n=0}^{s} \alpha_n \frac{\zeta^n}{n!}$$

(3.2)

where $\alpha_0 = \alpha_1 = \cdots = \alpha_p = 1$, and $1 \leq p < s$. Introducing the disk $D_r = \{ \zeta \in C : |\zeta + r| \leq r \}$, we may define

$$\rho = \rho(s,p) = \sup_{P \in \mathcal{P}_{s,p}} \{ r : D_r \subset S_p \}$$

which is our main object of interest.

### 3.1.2 Preliminary results

As in [50] we shall make use of the theory of positive real functions. For details on this subject in general, we recommend the survey by Dahlquist [27]. We define $C^-$ and $C^+$ as the open left and right half-planes, respectively. Next we recall that $f(z)$ is a positive function if it is analytic in $C^+$ and maps $C^+$ into $C^+$. If such a function is real-valued for real $z$, we say that it is a positive real function. The following well-known facts [27] about positive real functions will be needed.

**Lemma 3.1** If $f(z) = q(z)/r(z)$ is a positive real rational function, all coefficients of $q$ and $r$ have the same sign.

**Lemma 3.2** If $g(z) = f(z) - az$ is regular at $\infty$, then $f(z)$ is positive or an imaginary constant if and only if $a \geq 0$ and $g(z)$ is positive or an imaginary constant.

**Lemma 3.3** A rational function $f(z) = q(z)/r(z)$ where $q$ and $r$ are relatively prime and at least one of them is not a constant is positive if and

---

1 Obviously, the stability polynomials of $s$-stage explicit Runge-Kutta methods of order $p$ constitute a subclass (possibly empty due to the order barriers) of $\mathcal{P}_{s,p}$. 

only if \( q + r \) has all its roots in \( C^- \) and \( \text{Re}(f(iy)) \geq 0 \) for all \( y \) such that \( r(iy) \neq 0 \).

Following [50] we introduce the transformation \( \zeta = -2r/(1 + z) \) mapping \( C^+ \) one-to-one onto \( D_r \). If \( D_r \subset S_P \) then

\[
f(z) = \left[ 1 + P \left( \frac{-2r}{1 + z} \right) \right] / \left[ 1 - P \left( \frac{-2r}{1 + z} \right) \right]
\]

is a positive real function. Defining

\[
q(z) = \sum_{n=0}^{s-1} \beta_n z^n = -\sum_{i=1}^{s} \frac{\alpha_i}{i!}(-2r)^i(1 + z)^{s-i}
\]

we may write

\[
f(z) = \frac{2(1 + z)^s - q(z)}{q(z)}
\]

### 3.1.3 Some stability theorems

We start by deducing a general upper bound for \( \rho \). To this end we define the polynomial

\[
H_{s,p}(r) = \binom{s}{s - p} - (1)^p \sum_{n=0}^{p} \frac{(-2r)^n}{n!} \binom{s - n}{s - p}.
\]

Then Lemma 3.1 immediately yields

**Proposition 3.4** \( \rho \leq r_0(s, p) \) where \( r_0(s, p) \) is the unique positive root of \( H_{s,p} \).

**Proof.** These polynomials are discussed in some detail in Appendix B. Proposition B.1 ensures the existence and the uniqueness of positive roots.
3.1. The largest disk of stability.

The inequality follows from Lemma 3.1 applied to (3.4) by observing that for odd \( p \) \( H_{s,p}(r) \) equals \( 2 \left( \frac{s}{s-p} \right) - \beta_{s-p} \) and that for even \( p \) it equals \( \beta_{s-p} \). □

From the discussion of the Appendix B (Theorem B.2) we next give a general localization result.

**Theorem 3.5** We have

\[
s - p + 1 \leq r_0(s, p) \leq s - \frac{1}{2} - (-1)^p \frac{1}{2}.
\]

As \( s \to \infty \) we have

\[
s - p + 1 - r_0(s, p) = O(s^{-1}).
\]

As already mentioned it is known from [50] that the bound of Proposition 3.4 is sharp for \( p = 1 \). We shall now prove that this is also so for \( p = 2 \) and that the corresponding optimal polynomial is unique. One may notice that in this case the question of uniqueness cannot be settled by using the previously quoted result from [51].

**Theorem 3.6** Let \( P \in \mathcal{P}_{s,2} \). Then \( D_{s-1} \subset S_P \) if and only if \( P(\zeta) = \frac{s-1}{s}(1 + \frac{\zeta}{s-1})^s + \frac{1}{s} \).

In the proof of this theorem we shall need the following lemma

**Lemma 3.7** Let

\[
f(z) = \frac{2(1 + z)^s - q(z)}{q(z)}
\]

where

\[
q(z) = \sum_{n=0}^{s-1} \beta_n z^n
\]
and
\[ 0 < \beta_{s-1} \leq 2s, \quad \beta_{s-2} = 0. \]

Then \( f(z) \) is a positive real function if and only if
\[ q(z) = \frac{\beta_{s-1}}{2s}((1 + z)^s - (1 - z)^s) \]

**Proof.** Assume that \( f(z) \) is a positive real function. Then, as in the proof of the lemma in [50], we deduce that \( q(z) \) is an even (odd) polynomial with nonnegative coefficients if \( s - 1 \) is even (odd). We next make use of Lemma 3.3. Since \( q(z) \) is either even or odd with nonnegative coefficients, -1 is not a root of \( q(z) \), thus \((1 + z)^s - q(z)\) and \( q(z) \) are relatively prime. The roots of \((1 + z)^s\) are definitely in \( C^- \), hence it remains to be checked when \( \text{Re}(f(iy)) \geq 0 \). If \( s - 1 \) is even, then
\[ \text{Re}(f(iy)) = \frac{(1 + iy)^s + (1 - iy)^s - q(iy)}{q(iy)}. \]

We observe that the roots of \((1 + iy)^s + (1 - iy)^s\) are real and distinct since \( \frac{1+iy}{1-iy} \) is a one-to-one transformation of \( R \cup \{\infty\} \) onto the unit circle. Thus \( q(iy) \) must vanish at each of these roots and we conclude that
\[ q(z) = \frac{\beta_{s-1}}{2s}((1 + z)^s + (1 - z)^s). \]

The argument is completely analogous when \( s - 1 \) is odd.

The \( f(z) \) thus constructed is clearly a positive real function, and the proof is finished. \( \square \)

**Proof of Theorem 3.6.** We observe that \( D_{s-1} \subset S_P \) if \( P(\zeta) = \frac{s-1}{s}(1 + \frac{\zeta}{s-1})^s + \frac{1}{s} \).

We next assume that \( D_{s-1} \subset S_P \) and proceed as above, that is we find that
\[ f(z) = \frac{2(1 + z)^s - q(z)}{q(z)} \]
must be a positive real function with \( q(z) \) given by (3.3). From this we see that \( \beta_{s-1} = 2(s - 1) \) and \( \beta_{s-2} = 0 \). By Lemma 3.7 we thus have
\[
q(z) = \frac{2}{\beta_1}(1 + z)^s - (z - 1)^s.
\]
By using (3.3) and the transformation \( \zeta = -2(s - 1)/(1 + z) \) we arrive at the desired result. \( \square \)

For \( p > 2 \) the bounds of Proposition 3.4 will not be sharp. This is illustrated by considering the simplest case \( p = 3 \) and \( s = 4 \). Then if \( \beta_1 = 8 \) (corresponding to the bound of Propostion 3.4),
\[
f(z) = \frac{2}{\beta_3} z + g(z)
\]
where
\[
g(z) = \frac{(8 - \beta_3 - \frac{2}{\beta_2} \beta_2)z^3 + (12 - \beta_2 - \frac{2}{\beta_1} \beta_1)z^2 - \frac{2}{\beta_1} \beta_0 z + 2 - \beta_0}{\beta_3 z^3 + \beta_2 z^2 + \beta_1 z + \beta_0}.
\]

By Lemma 3.1 we must have \( \beta_0 = 0 \) for \( f \) to be positive real. But then it turns out that the numerator of \( g \) has zeros in \( C^+ \), which is impossible if \( f \) is to be positive.

We are thus left with the problem of how to estimate \( \rho \) for \( p > 2 \). A naturally related question is that of the sharpness of the bound of Proposition 3.4. Do we have \( \rho - (s - p + 1) \to 0 \) as \( s \to \infty \)? The numerical experiments of the next section may be in agreement with an affirmative answer to this question, but they are of course too few to indicate much. The complexity of such computations increases rapidly with \( s - p \), making it difficult to use them as a guideline for further investigations. From these experiments it may however be suggested that the bound of Proposition 3.4 is "reasonably" good.

### 3.1.4 Numerical results

For some special \((s, p)\)-pairs we have computed the bounds \( r_0(s, p) \). In addition we have used some routines from the NAG-library to seek for the optimal radii and the corresponding polynomials. These tests were run on a CRAY XMP-28. Some of the results are presented in the table below, and
Table 3.1: Numerical results for some selected values of $s$ and $p$.

\[
\begin{array}{|c|c|c|c|c|}
\hline
p & s & r_0(s, p) & \rho(s, p) & \alpha_{p+1} & \alpha_{p+2} \\
\hline
3 & 4 & 2.19 & 2.07 & 0.5698 & \\
3 & 5 & 3.15 & 2.94 & 0.7374 & 0.2771 \\
4 & 5 & 2.34 & 2.22 & 0.5806 & \\
4 & 6 & 3.30 & 3.06 & 0.7707 & 0.3053 \\
5 & 6 & 2.53 & 2.37 & 0.6083 & \\
5 & 7 & 3.47 & 3.17 & 0.7956 & 0.3305 \\
6 & 7 & 2.71 & 2.55 & 0.6185 & \\
\hline
\end{array}
\]

Figure 3.1: The stability region and the largest inscribed disk for $p = 3, s = 4$. 
3.1. The largest disk of stability.

Figure 3.2: The stability region and the largest inscribed disk for $p = 2, s = 4$.

we have plotted the stability region with the corresponding largest disk for two examples.

The stability region $S_P$ may consist of several components i.e. $S_P = \cup_{k=1}^{n} \Omega_k$, $n \geq 1$ where $\Omega_k \cap \Omega_j = \emptyset$, $k \neq j$. Exactly one of these, say $\Omega_1$, will have $\zeta = 0$ on its boundary. The algorithm we use for determining the stability regions traces the boundary of $\Omega_1$. In all the examples we consider, $\Omega_1 = S_P$. To verify this one may just check that all the zeros of $P(\zeta)$ are located inside $\Omega_1$ since any component must contain at least one zero. Our algorithm consists in parametrizing the curve defined by $|P(\zeta)| = 1$ by putting

$$P(\zeta(\theta)) = e^{i\theta}$$

and differentiate with respect to $\theta$ to obtain the complex ordinary differential equation

$$\zeta'(\theta) = i \frac{P(\zeta)}{P'(\zeta)}, \quad \zeta(0) = 1.$$
This initial value problem is solved by a Runge-Kutta method. A similar technique is used in the optimization procedure for determining $\rho(s,p)$ in Table 3.1.

### 3.2 P-stability properties of CERK methods

It is clear that when a CERK method of the previous chapter is applied to a delay differential equation of the kind given by (1.2) by using the approach suggested by (1.4) the stability of the numerical solution is affected. There are many concepts of stability for DDEs which are based on various test equations (e.g. Al-Mutib [1], Barwell [7], Cryer [26] and Jackiewicz-Bakke [49]). We use the equation (1.3) and we shall adopt the following definition of the region of P-stability suggested by Barwell [7]

**Definition 3.8** Given a numerical method for DDEs, the P-stability region of the method is the set $S_P$ of pairs of complex numbers $(\alpha, \beta)$, $\alpha = ah$, $\beta = bh$ such that the numerical solution of (1.3) converges to zero as $x \to \infty$ for steplengths $h$ satisfying

\[ mh = \tau, \quad m \text{ positive integer} \quad (3.5) \]

If we drop the restriction (3.5) we get the definition of the GP-stability region.

Watanabe and Roth [72] have used the argument principle to analyze the P-stability properties of both Runge-Kutta formulas and linear multistep methods. We shall adopt the techniques suggested by Zennaro [75] and we resume here his the most important results along with an indication of the result of applying this theory to continuous explicit Runge-Kutta methods adapted to DDEs. Clearly, when solving (1.3) we need to approximate the terms

\[ y(x_n + c_i h - \tau), \quad i = 1, \ldots, s \]
by means of our continuous method. Because of (3.5) these values are all in the interval \( x_s < x < x_s + h \) where \( s = n - m + 1 \). More precisely, we get

\[
K_i^{n+1} = a(y^n + h \sum_{j=1}^{s-1} a_{ij} K_j^{n+1}) + b(y^{s-1} + h \sum_{j=1}^{s} b_j e_i K_j^n)
\]

\[
y^{n+1} = y^n + h \sum_{i=1}^{s} b_i(1) K_i^{n+1}
\]

(3.6)

where we use superscript for the step index. Now let us introduce the matrix

\[ B = ((b_{ij})) := b_j(e_i) \]

and define the vector \( Y^n := (y^n, h K_1^n, \ldots, h K_s^n)^T \) having \( s + 1 \) components and for convenience we define \( b = (b_1(1), \ldots, b_s(1))^T \) and \( \eta = b^T(I - \alpha A)^{-1}u \) with \( u = (1, \ldots, 1)^T \). Then (3.6) can be written

\[
Y^{n+1} = LY^n + MY^{s-1} + NY^{s-1}, \quad s \geq 1
\]

(3.7)

where \( L, M, N \) are all \((s + 1) \times (s + 1)\)-matrices

\[
L = \begin{pmatrix}
1 + \alpha \eta \\
\alpha(I - \alpha A)^{-1}u & 0
\end{pmatrix},
\]

\[
M = \begin{pmatrix}
0 \\
\beta b^T(I - \alpha A)^{-1}B \\
\vdots \\
0
\end{pmatrix},
\]

\[
N = \begin{pmatrix}
\beta \eta \\
\beta(I - \alpha A)^{-1}u & 0
\end{pmatrix}.
\]

So we consider the characteristic equation of (3.7)

\[
\det(\lambda^{m+1}I - \lambda^m L - \lambda M - N) = 0
\]

(3.8)
It is known that P-stability is equivalent to (3.8) having all its roots inside the unit circle. Zennaro proves that these roots are also the roots of

$$\lambda = r_\alpha(\beta/\lambda^m) \quad (3.9)$$

where $r_\alpha(z)$ is the rational function

$$r_\alpha(z) = 1 + (\alpha + z)b^T(I - \alpha A - zB)^{-1}u. \quad (3.10)$$

Notice that $\hat{r}(\alpha) = r_\alpha(0)$ is nothing but the stability function for ODEs and we denote by $S_A$ the region of absolute stability. Consider the algebraic curve $\Gamma_r$ defined by $|r_\alpha(z)| = 1$. We are interested in the shortest distance from this curve to the origin

$$\sigma_\alpha = \inf_{z \in \Gamma_r} |z|$$

This quantity $\sigma_\alpha$ plays an important role in the main result of [75] which we give below.

**Theorem 3.9 (Zennaro).** The P-stability region of the Runge-Kutta method for DDEs (3.6) is

$$S_P = S'_P \cup B_P,$$

where $S'_P = \{ (\alpha, \beta) \in C \times C | \alpha \in S_A \text{ and } |\beta| < \sigma_\alpha \}$ and $B_P = \{ (\alpha, \beta) \in C \times C | \alpha \in S_A, |\beta| = \sigma_\alpha, \beta \neq \sigma_\alpha \}^m \forall m \geq 1 \text{ and } \forall z : |r_\alpha(z)| = 1 \text{ and } |z| = \sigma_\alpha \}$.

Remark that the set of points excluded from the boundary $S_P \setminus S_P$ will be infinite if the number $C = \arg(r_\alpha(z))/(2\pi)$ is irrational for any $z$ such that $|r_\alpha(z)| = 1$ and $|z| = \sigma_\alpha$.

Although Zennaro also provides a tool for computing $\sigma_\alpha$ for a given method this task seems still not to be trivial. We are especially interested in applying Theorem 3.9 to the CERK methods of the previous chapter. We shall only give an indication of this by looking at first and second order CERK
methods. P-stability properties for first order methods involves no continu-
ous approximation since values for the retarded argument is only needed at
the nodes. We trivially have

\[ r_\alpha(z) = 1 + \alpha + z \]

and \( S_P = \{ (\alpha, \beta) : |\beta| \leq 1 \text{ and } |1 + \alpha| \leq 1 - |\beta| \} \). For the second order case
we get

\[ r_\alpha(z) = 1 + \alpha + z + \frac{\frac{1}{2} (\alpha + z)^2}{1 - \frac{1}{2} c_2 z} \] (3.11)

If we choose \( c_2 = 1 \), which again implies that the retarded argument is only
needed at the nodes, (3.11) becomes

\[ r_\alpha(z) = \frac{\frac{1}{2} (\alpha + 1) z + 1 + \alpha + \frac{1}{2} \alpha^2}{-\frac{1}{2} z + 1} \] (3.12)

So \( \Gamma_r \) is a straight line if \( |1 + \alpha| = 1 \) otherwise it is the circle centered at \( C_0 \)
with radius \( \rho \)

\[ C_0 = 2 + \frac{(1 + \bar{\alpha})(2 + \alpha)^2}{1 - |1 + \alpha|^2}, \quad \rho = \frac{|\alpha + 2|^2}{|1 - |1 + \alpha|^2|} \]

Thus we obtain the closure of the P-stability region by taking \( \beta \leq ||C_0| - \rho| \)
with \( \alpha \in S_A \). For \( \alpha \) real we get \( |\beta| \leq -\alpha < 2 \). Now, still letting \(-2 < \alpha < 0\)
one may check that if \( 0 < c_2 < 1 \)

\[ \sigma_\alpha = \min\{-\alpha, \frac{\alpha + 2}{1 - c_2}\} \]

suggesting that \( c_2 = 1 \) is not such a bad choice when P-stability properties
are considered.
Chapter 4

Implementation of CERK Methods

4.1 Local error estimate, step size selection.

We assume that a CERK(p,q) pair of methods is available. Let us define $b_i = b_i(1), \ i = 1, \ldots, s$ and for a moment we consider only the underlying discrete method of order $p$ along with the auxiliary method of order $q$ having weights $\hat{b}_1, \ldots, \hat{b}_s$. We define $y_{n+1} = u(x_n + h)$ and $\hat{y}_{n+1} = y_n + h_n \sum_{i=1}^{s} \hat{b}_i K_i$.

Now we can either have $p < q$ as in the methods by Fehlberg [38, 39] or $p > q$ like in the example of Section 2.1. We prefer the latter approach here such that we may have the optimal number of stages CEN(p) for the main method (see Proposition 2.19) and we shall, in fact, assume that $q = p - 1$. We define the local error estimate in the usual way as

$$\delta_{n+1} = \hat{y}_{n+1} - y_{n+1}.$$

It is important that the error coefficients

$$\hat{e}_i = 1 - \gamma(t_i) \sum_{j=1}^{q} \hat{b}_j \phi_j(t_i) \neq 0, \ i = N_q + 1, \ldots, N_{q+1}$$

69
for the error estimate to be reliable. For the CERK pair of Example 2.22 the
error coefficients become

\[
\begin{array}{cccccccc}
7 & 7 & 13 & 3 & 1 & 1 & 3 & 11 & 7 \\
192 & 192 & 128 & 32 & 6 & 6 & 32 & 16 & 192
\end{array}
\]

where the numbering is as explained on page 24. We shall apply the usual
step size control formula (see e.g. [48]) computing \( f_s \)

\[
f_s = \min\{2, \max\left\{ \frac{1}{2}, 0.9(\text{TOL} / \text{EST})^{1/(p+1)} \right\}\}
\]

where \( \text{EST} = \| \delta_{n+1} \|_\infty \) and \( \text{TOL} \) is the tolerance for the local error. The
factor \( f_s \) is both used to reduce the step size whenever a failure occurs and
to compute the next step size as \( h_{n+1} = f_s h_n \).

This classical strategy for local error control and step size selection is re-
sumed in the following algorithmic description

**Algorithm 4.1** The main function.

repeat
  repeat
    attempt step
    compute error estimate \( \delta_{n+1} \)
    adjust step size
  until step accepted (or discontinuity detected)
  (step to discontinuity)
save step
until endpoint reached (or out of memory)

Sometimes one would prefer to use error per unit step and \( \text{TOL} \) can be
interpreted as an absolute, relative or mixed error tolerance. We refer to
Shampine [68] for a discussion of the various possibilities.
4.2 Datastructures implemented in C.

Many production codes for IVPs do not provide data structures for global storage of the numerical solution. The user is usually allowed to specify output points in increasing (or decreasing) order along the axis of the independent variable. The responsibility for keeping track of the solution outside the current step is left to the user. To run such a code efficiently, one must either decide in advance (before integration starts) where the output points should be, or one must examine the code very carefully in order to identify the data necessary to reconstruct the numerical solution at arbitrary points. The former alternative is not even possible in some cases, e.g. when solving certain IVPs with deviating arguments like (1.2). The latter alternative requires that the user builds his own data structures for storing the data. We believe that this deficiency in some of the existing software is mainly due to the following historical circumstances:

- Memory has traditionally been an expensive resource, leaving room only for local storage.
- The programming language Fortran has (up to now) not supported the appropriate data structures for such facilities.

Since memory has become considerably cheaper over the past few years, and since the programming tools available today (e.g. Fortran 8X and C) do have powerful tools for handling advanced data structures, we believe that there is time to reconsider the design and implementation of ODE-codes.

We restrict ourselves to the implementation of CERK methods, using the language C, and we begin by considering the data to be handled by such codes. See Figure 4.1. We suggest that the rectangular single-line boxes are implemented as structures in C, containing the following information:

**ODE-SOLVER:** The characterization of the method.

- The coefficients of the method, i.e. the Butcher-matrix $A$.
- The continuous weights $b_1(\theta), \ldots, b_s(\theta)$
Figure 4.1: The data modularity of a CERK code.

- The order of the method $p$.

**ODE-PROBLEM:** The user specifications.

- Dimension of the system, $m$.
- Tolerances, Relative and absolute.
- Start and endpoint of integration.
- Initial value
- Maximal step length
- A pointer to a function that defines the differential equations.

**COMPUTED DATA:** Information about the process of integration.

- An array of pointers to structures (called IVDATA) containing necessary information about the solution in the consecutive integration steps.
- Pointers to the (IVDATA) structures containing the start and (current) end point.
- Statistical information (Number of steps, number of function calls etc.).
- A pointer to a function (called csol) computing the solution at an arbitrary point between the start and (current) end point above.
4.2. Datastructures implemented in C.

The user will be responsible for initializing the ODE-PROBLEM structure and of course, for writing the function defining the differential equations.

The IVDATA structure is composed of the following attributes

- The start and endpoint of the step.
- The numerical solution at the start points.
- The stepsize.
- The error estimate.
- Either $K_i$, $i = 1, \ldots, s$ or more economically $\sum_{j=1}^{s} b_{ij} K_j$, $i = 1, \ldots, p$ where $b_{ij}(\theta) = \sum_{j=1}^{p} b_{ij} \theta^j$, $i = 1, \ldots, s$.

The array of pointers to the IVDATA structures are organized in a ring buffer, the size of which is tuned in accordance with the available amount of memory. In this way the space will be reused if the program runs out of memory at some point during the integration. When solving delay differential equations this might be useful when the delays are not too large compared to the typical step size. In any case, The IVDATA space allocation should be dynamic.

The usage of this system would be slightly different from most existing ODE-codes. Instead of doing integration and producing output concurrently, one keeps on with the integration until either the endpoint is reached or the available amount of memory is used. Then, with all the information available, the solution at the desired output points are computed simply by calls to the csol function. The unavoidable difficulties with this approach is of course the sheer volume of the data and the fact that one does not know in advance how much data will be formed. In general one would probably have to require that the code resorts some kind of secondary storage. This will not be discussed here.

Notice how simple it is to replace the CERK method with another method. One just has to create a new ODE-SOLVER structure and set the appropriate parameters. The remaining part of the code is not affected.
4.3 Some numerical results.

Figure 4.2: The distinct error polynomials of the new method.

We have written a program built on the ideas of Section 4.1 and 4.2 using the language C on a PC-AT. Some tests have been performed with a few methods. But since we have not yet optimized or tuned the free parameters of our new methods with respect to error constants or stability regions we shall not challenge the most commonly used codes by making comparisons on standard test problems. It will be the subject of future research to find the best candidate among the families of methods of Chapter 2. Nevertheless, we reproduce some of the results obtained with the 9 stage extensions of Dormand Prince (5,4) and with the 8-stage method from Example 2.22 sometimes called the "new method" in the remaining part of this section. To
begin with, let us consider the error polynomials of this new method which are plotted in Figure 4.2. It turns out that only 10 of the 20 error polynomials of order six are distinct. For all polynomials, \( \theta = 1 \) is a stationary point as

expected, but for some of them, these are local minima. The dominating polynomial is the one corresponding to the tree, \([57]_5\), and in fact, this is also so for the Dormand Prince extensions (see Appendix A and [62]). Even if \( e(t) \) is replaced by \( \alpha(t)e(t) \) this same polynomial would dominate.

Figure 4.3: Global error versus \( x \). TOL=1.e-4. The solid line refers to Example 2.22, and the dashed line to the methods on page 13 with \( c_8 = 1/5 \) and \( c_9 = 1/2 \).

Consider the problem

\[
y'(x) = e^y, \quad y(0) = 0
\]

(4.1)

which has the exact solution \( y(x) = -\ln(1 - x) \) for \( 0 \leq x < 1 \). Notice
that at $x = 0$ all the elementary differentials have the value 1. To give a qualitative view of the 9-stage Dormand-Prince interpolants of Section 2.1 and our new 8-stage methods represented by Example 2.22 we present the result of applying these methods to the problem (4.1) in Figure 4.3. We have used the stepsizes selected by the former method in both cases. It turns out that for most of the examples we have tried this new method on, we obtain a global accuracy that is better than the given tolerance. It is likely that the auxiliary method has a large error constant such that our local error estimate usually determines a step size that is smaller than necessary. This method is the only one of order 4 that uses only the first 6 stages, but by using also the 7th stage one would expect an improvement. The 8th stage should be avoided if possible, since the absence of this stage in the error estimation formula causes rejected steps to cost only 6 function evaluations.

**Van der Pol’s equation**

We consider the problem

\[
\begin{align*}
y_1' &= y_2, \\
y_1(0) &= 2, \\
y_2' &= \epsilon(1 - y_1^2)y_2 - y_1, \\
y_2(0) &= 0.
\end{align*}
\]

(4.2)

solved on the interval $0 \leq x \leq 20$. By putting $\epsilon = 1$ and $\epsilon = 5$ we obtain the problem E2 from the non-stiff and stiff DETEST package [37] respectively. It is known that the solution of (4.2) tends to a limit cycle as $x \to \infty$. Since the initial point is located very close to this limit cycle for $\epsilon = 1$ the solution is almost periodic from the start. For more details about this equation, see [52]. We shall discuss the global behaviour of the methods above or roughly speaking, how the underlying discrete methods behave when applied to the non-stiff Van der Pol’s equation. Considering Table 4.1, we observe from the number of steps taken by the Dormand-Prince method (DP) and the new method (NM) respectively that the latter selects a smaller stepsizes than the former. Hence one should also look at the ERR column, which displays the ratio between the maximum global
4.3. Some numerical results.

<table>
<thead>
<tr>
<th>TOL</th>
<th>STEPS</th>
<th>REJEC</th>
<th>EVAL</th>
<th>ERR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DP</td>
<td>NM</td>
<td>DP</td>
<td>NM</td>
</tr>
<tr>
<td>1.0E-04</td>
<td>72</td>
<td>106</td>
<td>26</td>
<td>25</td>
</tr>
<tr>
<td>1.0E-05</td>
<td>108</td>
<td>159</td>
<td>30</td>
<td>18</td>
</tr>
<tr>
<td>1.0E-06</td>
<td>166</td>
<td>257</td>
<td>31</td>
<td>19</td>
</tr>
<tr>
<td>1.0E-07</td>
<td>256</td>
<td>388</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>1.0E-08</td>
<td>402</td>
<td>640</td>
<td>13</td>
<td>12</td>
</tr>
<tr>
<td>1.0E-09</td>
<td>635</td>
<td>963</td>
<td>11</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 4.1: The continuous Dormand-Prince method versus the method of Example 2.22
Figure 4.4: Global error for the new method (solid line) and the Dormand-Prince method (dashed line) applied to the Van der Pol equations.
4.3. Some numerical results.

error and the tolerance, before drawing any conclusions. We see that the accuracy achieved by the new method is considerably higher than that of Dormand-Prince. The columns RJEC and EVAL show the number of rejected steps and function evaluations respectively. The global error for the two methods is presented in Figure 4.4 for two different tolerances 1.0E-04 and 1.0E-06. All estimates of the global error were computed by subtracting the results obtained with TOL=1.0E-09.

Stability

We show the stability regions of the new method and the Dormand Prince method respectively in Figure 4.5. Notice that the stability region of our

![Figure 4.5: Stability region of the new method (solid line) and the Dormand Prince method (dashed line)](image)

new method, which is not optimized, has a significant intersection with the positive half plane, a property that it does not share with the carefully constructed Dormand-Prince method.
Chapter 5

Concluding remarks

In Chapter 2 we introduced the general concept of continuous explicit Runge-Kutta methods. The definition itself does not reveal anything new as these methods can all be viewed as conventional Runge-Kutta methods with interpolants which are extensively studied by many authors. However, our definition of uniform order gave birth to a new meaning of the term order barriers discussed in Section 2.2. In some sense one may say that the problem of determining order barriers is simpler for the continuous than the discrete case. This is due to the "decoupling" of the conditions into $p$ groups each consisting of $n_p$ conditions. To some extent, we were able to treat these groups independently of each other, an approach that is impossible for the discrete case. In general we found theorems that can be used to determine lower bounds for the minimal number of stages necessary to obtain a CERK method of some order $p$. Specifically, we investigated the cases $p = 3, 4, 5$, and by combining the lower bounds we found with some known methods, we were able to determine the order barriers for $p = 3$ and $4$. The most important result was probably the discovery of 5th order CERK methods with 8 stages. As far as we know, no such methods have been presented in the literature. However, the strategy we used to construct these methods gave rather strong restrictions on the coefficients of the method. Our choice of basis for the row space of the order condition matrix along with a simplifying assumption led to the requirement $2c_3 = c_4 = c_5$. It would be interesting to know whether there exist 8 stage 5th order methods that
do not have to satisfy this condition. At least we know that the method by Dormand and Prince discussed in Section 2.1 cannot be extended to a 5th order CERK method with 8 stages. And another consequence of Theorem 2.14 is that the method by Fehlberg, RKF45, having six stages, cannot be extended to a 5th order continuous method with less than three additional stages. We saw in Section 2.3 that stage reuse or FSAL evaluation can be included in the methods of order 3, 4 and 5. Whereas some of the traditional methods use this stage for error estimation, we found it more useful as a part of the continuous method and thereby saving one function evaluation in each rejected step. However, the most important consequences of imposing this stage reuse property is the one-stage reduction in effective cost per step and the resulting $C^1$-continuity of the global approximation.

We discussed superficially the monotonicity of the error polynomials, but the conclusion is that it seems too ambitious to hope for all error polynomials to be monotonic as the order increases. In Chapter 4 we discussed briefly the implementation of CERK methods in general, and even though we chose to use the traditional error estimation strategy, our methods should be well suited for the defect control strategies mentioned earlier. We presented some results with a prototype 5th order CERK method with 8 stages along with the continuous extension of the Dormand-Prince method of Section 2.1. The only conclusion one can draw from these tests is that the method behaves as one might expect from its theoretical properties. Unfortunately, they do not say anything about the potential of our new methods when applied to real world problems. We did not find it worthwhile to perform thorough tests with the prototype. We prefer to spend some time optimizing error constants and stability regions before throwing it into the relentless DETEST package [37]. This is the major remaining task. We will then also look for methods particularly suited for differential equations with deviating arguments.

A natural continuation of the ideas of Section 2.2 and 2.3 would be to consider higher order CERK methods, but as in the discrete case, to perform a rigorous analysis of order barriers is difficult due to the immense growth in the number of order conditions as the order increases. Still, some interesting things can be done easily. Lately we have proved that at least 3 additional stages are necessary in order to supply the Dormand-Prince(8,7) [66] method with a 7th order continuous extension.
The stability questions considered in Chapter 3 is somewhat loosely connected to the rest. Section 3.1 was devoted to the stability of explicit Runge-Kutta methods. Clearly, stability of CERK methods applied to ODEs is that of the underlying discrete methods and hence, exhaustively studied in the literature. However, some interesting problems still remain, at least from a theoretical point of view. We investigated the largest disk of stability following an idea by Jeltsch and Nevanlinna. As we pointed out, a major reason for choosing this measure for the size of the stability region was its simplicity. We solved the problem completely for order two methods and we gave some upper bounds for the optimal radii in the general case. We posed the question if these upper bounds become sharp as \( s - p \to \infty \) without being able to give an answer. Even though this result is mostly of theoretical interest, it certainly is a challenge for future research. The original purpose of this section was to investigate how one should choose the free parameters of the method in order to somehow maximize the region of stability. The numerical values we gave for some specific cases should be used as a guideline for the remaining search for good CERK methods. Section 3.2 about P-stability of CERK methods brought little interesting news and was meant to be an indication on how this problem can be attacked. For order 1 and 2, the rational function to be studied was simple and the information we got about P-stability agreed well with what one might expect from the stability of the related ODE-method and the stability region of the test equation. One may observe that, except from a general paper by Zennaro, the literature does not bring many stability results for higher order Runge-Kutta methods applied to delay differential equations. Further investigations should therefore be conducted.
A remark on an error polynomial of some fifth order continuous extensions of Dormand-Prince(5,4)

We start by summarizing some of the results from [62]. Let $A$ be the Butcher matrix of the Dormand Prince (4,5) method, and let $b_1, \ldots, b_6$ be the weights of the 5th order method. Let $b_1(\theta), \ldots, b_6(\theta)$ be continuous weights in a 4th order continuous method, $u_0$ such that

$$u_0(x_0 + \theta h) = y_0 + h \sum_{i=1}^{6} b_i(\theta) K_i.$$ 

Now, define an extended method with two more stages such that $a_{ij} = b_j(c_i)$, $i = 8, 9, j = 1, \ldots, 6$ and $a_{8j} = a_{9j} = 0$ otherwise, i.e.

$$K_i = f(x_0 + c_i h, u_0(x_0 + c_i h)) \quad i = 8, 9.$$  

85
A. A remark on an error polynomial of some fifth order continuous extensions of Dormand-Prince\((5,4)\)

We can now, with certain restrictions on \(c_8\) and \(c_9\) construct a 5th order continuous method, \(u_1\) defined by

\[
\begin{align*}
u_1(x_0 + \theta h) &= \eta_0(\theta)y_0 + \eta_1(\theta)y_1 + \xi_1(\theta)hK_1 + \sum_{i=7}^{9} \xi_i(\theta)hK_i \\
&= y_0 + h\sum_{i=1}^{9} w_i(\theta)K_i
\end{align*}
\]

The functions \(\eta_i(\theta), \xi_i(\theta)\) and \(w_k(\theta)\) above are 5th degree polynomials in \(\theta\), and their coefficients depend on \(c_8\) and \(c_9\). Clearly, the following conditions are satisfied

\[
\begin{align*}
1 &= \eta_0(\theta) + \eta_1(\theta) \\
w_1(\theta) &= \xi_1(\theta) + b_1\eta_1(\theta) \\
w_i(\theta) &= b_i\eta_1(\theta) \quad i = 2, \ldots, 6 \\
w_i(\theta) &= \xi_i(\theta) \quad i = 7, \ldots, 9
\end{align*}
\]

(A.1)

A strategy for choosing the parameters \(c_8\) and \(c_9\) is to minimize the functional \(G(c_8, c_9)\) defined by:

\[
G(c_8, c_9) = \max_{N_i + 1 \leq i \leq N_e} \left\{ \max_{\theta \in [0,1]} |e_i(c_8, c_9; \theta)| \right\}
\]

Numerical experiments show that if \((c_8, c_9)\) is not too close to the singular cases (again, see [62]), we have

\[
G(c_8, c_9) = \max_{\theta \in [0,1]} |e'_i(c_8, c_9; \theta)|
\]

where \(i'\) corresponds to the tree \(t_{i'} = [57]_5\) using the notation of Butcher [20]. Consequently, the following result is important.

**Theorem A.1** The error polynomial \(e'_i(c_8, c_9; \theta)\) is independent of \(c_8\) and \(c_9\), i.e.

\[
e'_i(c_8, c_9; \theta) = e'_i(\theta) = \theta^6 + \frac{186}{25}\theta^5 - \frac{432}{25}\theta^4 + \frac{216}{25}\theta^3
\]
Proof: Ignoring the first order condition, we can write the remaining order conditions in the following way.

$$\gamma(t_i) \sum_{j=2}^{9} \phi_j(t_i) w_j(\theta) = \theta^{\rho(t_i)} \quad 2 \leq \rho(t_i) \leq 5$$  \hspace{1cm} (A.2)

Using (A.1) together with the order conditions for the discrete 5th order Dormand Prince method we can simplify (A.2) into

$$\eta_1(\theta) + \gamma(t_i) \sum_{j=7}^{9} \phi_j(t_i) w_j(\theta) = \theta^{\rho(t_i)} \quad 2 \leq \rho(t_i) \leq 5 \hspace{1cm} (A.3)$$

We now want to compute $\phi_j(t_i)$, $j = 7 \ldots 9$ for some specific trees $t_i$. We consider the trees $t_{ik}$, $k = 0, \ldots, 4$ such that $t_{i0} = \tau$ and $t_{ik} = [t_{ik-1}]$, $k = 1, \ldots, 4$. Utilizing the fact that the polynomials $b_i(\theta)$ obeys the order conditions up to and including order 4, we get for $j = 7, \ldots, 9$

$$\phi_j(t_{ik}) = \sum_{l=2}^{7} a_{jl} \phi_l(t_{ik-1}) = \sum_{l=2}^{6} b_l(c_j) \phi_l(t_{ik-1}) = \frac{1}{(k-1)!} c_j^{k-1}.$$

In view of (A.3) we end up with the following system of equations

$$\begin{bmatrix} 1 & 3 & 3c_8^2 & 3c_9^2 \\ 1 & 4 & 4c_8^3 & 4c_9^3 \\ 1 & 5 & 5c_8^4 & 5c_9^4 \end{bmatrix} \begin{bmatrix} \eta_1(\theta) \\ w_7(\theta) \\ w_8(\theta) \\ w_9(\theta) \end{bmatrix} = \begin{bmatrix} \theta^3 \\ \theta^4 \\ \theta^5 \end{bmatrix}$$

or for short

$$M(c_8, c_9) z(\theta) = d(\theta)$$

Let us consider the tree $t_{i'}$ which we in consistency with the notation above, call $t_{is}$. We define $e_1(\theta)$ by

$$e_1(\theta) = \gamma(t_{is}) \sum_{l=2}^{9} \phi_l(t_{is}) w_l(\theta)$$
A remark on an error polynomial of some fifth order continuous extensions of Dormand-Prince(5,4)

such that the error polynomial \(e_\varphi(v_8, v_9; \theta) = \theta^6 - e_1(\theta)\). Now, computing 
\(\phi(t_i) = A^i \phi(t_i)\), and using \(w_6(\theta) = b_6 \eta_1(\theta)\) we get

\[e_1(\theta) = p^T z(\theta)\]

with

\[p = \begin{bmatrix} 6 & 5 \end{bmatrix} \begin{bmatrix} v(v_8) \end{bmatrix}^T\]

and

\[v(\theta) = \frac{6}{25} ( -108 \theta^2 + 288 \theta^3 - 155 \theta^4 )\]

Trivial computations now show that \(p\) is contained in the row space of \(M\), in fact we have:

\[p^T = \begin{bmatrix} -216 \over 25 \ 432 \over 25 \ -186 \over 25 \end{bmatrix} M\]

and the error polynomial \(v_\varphi(v_8, v_9; \theta)\) is given by:

\[v_\varphi(v_8, v_9; \theta) = \theta^6 + \frac{186}{25} \theta^5 - \frac{432}{25} \theta^4 + \frac{216}{25} \theta^3\]

\[\square\]
B

The real zeros of polynomials related to the generalized Bessel polynomials

It turns out that the polynomials of Proposition 3.4 are closely related to the generalized Bessel polynomials. We will use this link to localize the real roots of the polynomials of interest to us.

For any nonnegative integer $p$ and any $x > 0$ we define

$$G_p(x, y) = \sum_{k=0}^{p} \binom{p}{k} (x)_k \cdot y^{p-k}$$

and

$$R_p(x, y) = (x)_p - (-1)^p G_p(x, y)$$

where we have used the Pochhammer notation for ascending factorials,

$$(x)_k = \begin{cases} 
1 & , \quad k = 0 \\
 x \cdot (x + 1) \cdots (x + k - 1) & , \quad k > 0.
\end{cases}$$
B. The real zeros of polynomials related to the generalized Bessel polynomials

In the notation of [41] we have

\[ G_p(x, y) = 2^p \Theta_p(\frac{y}{2}, x - p + 1) \]  \hspace{1cm} (B.1)

where \( \Theta_p(z, a) \) is the generalized reverse Bessel polynomial. From Theorem 1’ on p. 80 in [41] and the obvious relation

\[ \frac{\partial G_p}{\partial y} = p \ G_{p-1} \]  \hspace{1cm} (B.2)

we have immediately

**Proposition B.1** For even \( p \) and each fixed \( x \), \( G_p(x, y) \geq 0 \) and \( R_p(x, y) \) has precisely one real root different from zero, which is simple and negative. For odd \( p \) and each fixed \( x \), both \( G_p(x, y) \) and \( R_p(x, y) \) have precisely one real zero, both being simple and negative.

We are interested in locating \( y_0(x, p) \), the unique negative root of \( R_p(x, \cdot) \), and in particular in seeing how it behaves for each fixed \( p \) as \( x \) grows. We have arrived at the following result.

**Theorem B.2** For any \( p \) and \( x > 0 \)

\[ -2(x + p - \frac{3}{2} - (-1)^{p+1} \frac{1}{2}) \leq y_0(x, p) \leq -2x. \]  \hspace{1cm} (B.3)

As \( x \to \infty \) we have

\[ y_0(x, p) + 2x = O(x^{-1}). \]  \hspace{1cm} (B.4)

We shall find that this result is a simple consequence of some elementary properties of the functions \( G_p(x, \cdot) \).

The generalized reverse Bessel polynomials \( \Theta_p(z, a) \) satisfy the differential equation [41, p. 13]

\[ z \frac{\partial^2 \Theta_p}{\partial z^2} - (2p - 2 + a) \frac{\partial \Theta_p}{\partial z} + 2p \Theta_p = 0 \]
which in conjunction with (B.1) and (B.2) gives the recurrence relation

\[(p - 1)yG_{p-2} - (x + y + p - 1)G_{p-1} + G_p = 0.\]  \hspace{1cm} (B.5)

Denoting, for odd \(p\), the negative root of \(G_p(x, \cdot)\) by \(\eta_0(x, p)\) this leads to the following lemma.

**Lemma B.3** For any odd \(p\) we have

\[-(x + p - 1) \leq \eta_0(x, p) \leq -x\]  \hspace{1cm} (B.6)

and

\[\eta_0(x, p + 2) < \eta_0(x, p).\]  \hspace{1cm} (B.7)

**Proof.** We prove the lemma by induction on \(p\). It is obviously true when \(p = 1\). Assuming it to hold for \(p - 2\), we have by (B.5)

\[G_p(x, \eta_0(p - 2, x)) = (x + p - 1 + \eta_0(p - 2, x))G_{p-1}(x, \eta_0(p - 2, x)) > 0\]

proving (B.7) and thus the right inequality of (B.6). The left inequality of (B.7) is proved by putting \(y = -(x + p - 1)\) into (B.5) yielding

\[G_p(x, -(x + p - 1)) = (p - 1)(x + p - 1)G_{p-2}(x, -(x + p - 1)) < 0. \square\]

We shall need the following consequence of this lemma.

**Lemma B.4** Let \(p\) be odd. Then for any \(\eta > 0\) we have the following inequalities

\[G_p(x, -x + \eta) \geq |G_p(x, -x - \eta)|\]  \hspace{1cm} (B.8)

\[G_{p+1}(x, -x + \eta) \geq G_{p+1}(x, -x - \eta)\]  \hspace{1cm} (B.9)

\[G_p(x, \eta_0(x, p) + \eta) \leq |G_p(x, \eta_0(x, p) - \eta)|\]  \hspace{1cm} (B.10)

\[G_{p+1}(x, \eta_0(x, p) + \eta) \leq G_{p+1}(x, \eta_0(x, p) - \eta)\]  \hspace{1cm} (B.11)
proof. Again the proof is by induction on p.

We start with (B.8) and (B.9), which of course are trivially true when p = 1. First, by (B.2) we have

\[ G_p(x, -x + \eta) = G_p(x, -x) + \int_0^x G_{p-1}(x, -x + t) \, dt. \]

Next by Lemma B.3 and the induction hypothesis

\[ |G_p(x, -x - \eta)| = |G_p(x, -x) - \int_0^x G_{p-1}(x, -x - t) \, dt| \leq G_p(x, -x) + \int_0^x G_{p-1}(x, -x + t) \, dt. \]

That (B.8) implies (B.9) is an obvious consequence of Lemma B.3 and of (B.2).

We next prove (B.10) and (B.11), which are also trivially true when p = 1. Again by Lemma B.3 and the induction hypothesis we have

\[ G_{p-1}(x, \eta_0(x, p) - \eta) \geq G_{p-1}(x, \eta_0(x, p - 2) - \eta_0(x, p) + \eta_0(x, p - 2) + \eta) \]

\[ = G_{p-1}(x, \eta_0(x, p) - 2(\eta_0(x, p) - \eta_0(x, p - 2)) + \eta) \]

and thus clearly

\[ G_{p-1}(x, \eta_0(x, p) - \eta) \geq G_{p-1}(x, \eta_0(x, p) + \eta) \]

which establishes (B.10). Finally, (B.11) is an obvious consequence of (B.10).

\( \square \)

Proof of Theorem B.2. (B.3) is now an immediate consequence of Lemma B.3 and of Lemma B.4.

For any fixed \( \epsilon > 0 \) we put \( y = -2x - \epsilon \) into \( R_p(x, y) \) and consider the resulting polynomial in \( x \),

\[ F_{p, \epsilon}(x) = \sum_{k=0}^{p} \alpha_k x^k. \]
We first compute

\[ \alpha_p = 1 - (-1)^p \sum_{k=0}^{p} \binom{p}{k} (-2)^{p-k} = 1 - (-1)^p (1 - 2)^p = 0. \]

Next we find

\[ \alpha_{p-1} = \frac{p(p-1)}{2} - (-1)^p \sum_{k=0}^{p} \binom{p}{k} \left( \frac{k(k-1)}{2} (-2)^{p-k} - (p-k) (-2)^{p-k-1} \right) \]
\[ = -pe. \]

which in conjunction with (B.3) proves (B.4). ∎
B. The real zeros of polynomials related to the generalized Bessel polynomials
Bibliography


