Computational Complexity of One-Step Methods for Systems of Dif--Etc(u)

A. G. Werschulz

Sep 76

N00014-76-C-0370

UNCLASSIFIED
Computational Complexity of One-Step Methods for Systems of Differential Equations

Arthur G. Werschulz

September, 1976

Department of Mathematics
Carnegie-Mellon University
Pittsburgh, PA 15213

Abstract: The problem is to calculate an approximate solution of an initial value problem for an autonomous system of N ordinary differential equations. Using fast power series techniques, we exhibit an algorithm for the p\textsuperscript{th}-order Taylor series method requiring only O(pN ln p) arithmetic operations per step as p \to +\infty. (Moreover, we show that any such algorithm requires at least O(pN) operations per step.) We compute the order which minimizes the complexity bounds for Taylor series and linear Runge-Kutta methods, and show that in all cases, this optimal order increases as the error criterion \epsilon decreases, tending to infinity as \epsilon tends to zero. Finally, we show that if certain derivatives are easy to evaluate, then Taylor series methods are asymptotically better than linear Runge-Kutta methods for problems of small dimension N.

This research was supported in part by the National Science Foundation under Grant MCS75-222-55 and the Office of Naval Research under Contract N00014-76-C-0370, NR 044-422.
1. Introduction

Let \( \mathcal{D} \) be a set of points in the real \( N \)-dimensional linear space \( \mathbb{R}^N \), and let \( \mathcal{O} \) be a set of operators on \( \mathbb{R}^N \), such that the initial value problem of finding a function \( x : [0, 1] \to \mathbb{R}^N \) satisfying

\[
\begin{align*}
\dot{x}(t) &= v(x(t)) & \text{if } 0 < t < 1 \\
x(0) &= x_0
\end{align*}
\]

has a unique solution for every \((x_0, v) \in \mathcal{D} \times \mathcal{O}\); we assume that \( x \) is analytic on \([0, 1]\). The autonomous form of this system is no restriction, since any non-autonomous system may be made autonomous by increasing the dimension of the system by one.

In Werschulz [76], we looked at the computational complexity of using one-step methods to generate an approximate solution to (1.1) on an equidistant grid in the sense of Stetter [73], that is, the methods considered computed approximations \( x_i \) to \( x(ih) \) by the recursion

\[
(1.2) \quad x_{i+1} = x_i + h \varphi(x_i, h) \quad (0 \leq i \leq n-1, \ n = h^{-1}),
\]

where \( h = n^{-1} \) is the step-size of a grid with \( n \) points, and \( \varphi \) is the increment function (Henrici [62]) for the method. (To be brief, we will refer to "the method \( \varphi \).") In that paper, we discussed the problem of optimal order and minimal complexity for rather general classes of one-step methods.

In this paper, we will use the techniques and results of Werschulz [76] to analyze the complexity of using Taylor series methods and linear Runge-Kutta methods to generate approximate solutions whose error does not exceed \( \varepsilon \). The model of computation, error measure, and complexity measure to be used are described in Section 2, as well as the relevant results from Werschulz [76].
We discuss the complexity of Taylor series methods in Section 3. Using the fast power series techniques of Brent and Kung [76], we show that \( O(p^N \ln p) \) arithmetic operations suffice to compute the \( p \)th-order Taylor series approximation; moreover, we show that \( O(p^N) \) operations are necessary. In Section 4, we discuss the complexity of linear Runge-Kutta methods. In both Sections, we compute lower and upper bounds on the complexity using a fixed method of given order; these results are then used to compute optimal orders which minimize these complexity bounds. We show that in all cases, the optimal order increases as \( s \) decreases, tending to infinity as \( s \) tends to zero.

Finally, we compare these two classes of methods in Section 5, where we show that if the partial derivatives of \( v \) are easy to evaluate, then Taylor series methods are asymptotically better (as \( s \) tends to zero) than linear Runge-Kutta methods for problems of small dimension \( N \).
2. Preliminary Results

Before proceeding any further, we will establish some notational conventions. Let \( X \) be an ordered ring; then \( X^+ \) and \( X^{++} \) respectively denote the nonnegative and positive elements of \( X \). (This is used in the cases \( X = \mathbb{R} \), the real numbers, and \( X = \mathbb{Z} \), the integers.) The symbol "\( := \)" means "is defined to be," while "\( \equiv \)" means "is identically equal to." We use "\( I \)" to denote the unit interval \([0, 1]\). The symbol "\( \nabla \)" is used to denote the gradient of a mapping. The notations "\( x \downarrow a \)" and "\( x \uparrow a \)" are used to indicate one-sided limits, as in Buck [65]. Finally, we write "\( (a,b)_c \)" to indicate the \( c^{th} \) part of equation \((a,b)\), as in Gurtin [75].

We next describe the model of computation to be used. We assume only that all arithmetic operations are performed exactly in \( \mathbb{R} \) (i.e., infinite-precision arithmetic) and that for any algorithm to be considered for the solution of \((1.1)\), a set of procedures is given for the computation of any information about \( v \) required by that algorithm. (For instance, with Runge-Kutta methods, we must be able to compute \( v \) at any point in its domain.)

In addition, we must pick an error measure, so that we may measure the discrepancy between the approximate solution produced by \( \varphi \) (via \((1.2)\)) and the true solution. For the sake of definiteness, we use the global error

\[
\epsilon_{0G}(\varphi;h) := \max_0 \text{Sise}_{\mathbb{R}} \| x(ih) - x \| ,
\]

where \( \| \cdot \| \) is a norm on \( \mathbb{R}^N \). Other error measures may be used, such as the local error per step \( \epsilon_L \) and the local error per unit step \( \epsilon_{LU} \) (see Henrici [62] and Stetter [73] for definitions); this would involve only a slight modification of the results contained in the sequel.
We finally describe the complexity measure to be used. Let \( \Phi = \{ \varphi_p : p \in \mathbb{Z}^{++} \} \) be a basic sequence in the sense of Werschulz [76], that is, there exist functions \( \kappa : \mathbb{R}^+ \times \mathbb{R}^+ \rightarrow \mathbb{R}^+ \) and \( \kappa_l, \kappa_U : \mathbb{R}^+ \rightarrow \mathbb{R}^+ \) such that

\[
\varepsilon_G(\varphi_p, h) = \kappa(p, h) h^p \quad \text{for } h \in \mathbb{R}^+ \text{ and } p \in \mathbb{Z}^{++},
\]

where

\[
0 < \kappa_l(p) \leq \kappa(p, h) \leq \kappa_U(p) < +\infty \quad \text{for } h \in \mathbb{R}^+.
\]

We say that \( \varphi_p \) has order \( p \). This is a slight extension of the definition of order given in Cooper and Verner [72]; the function \( \kappa_l \) introduced here is necessary and sufficient for the "order" of a method to be unique. (For the sake of exposition, we assume that \( \kappa_l \) and \( \kappa_U \) are analytic on \( \mathbb{R}^+ \), and that \( \lim_{p \to 0} \kappa_l(p)^{1/p} \) and \( \lim_{p \to 0} \kappa_U(p)^{1/p} \) exist and are positive real numbers; this will always be the case in the examples we consider.)

Then we will be interested in the total number of arithmetic operations \( C(p, \alpha) \) required to guarantee that

\[
\varepsilon_G(\varphi_p, h) \leq \varepsilon := e^{-\alpha},
\]

for a given \( p \) and a given \( \alpha \). (Here \( e \) is the base of the natural logarithms.) We suppose that \( 0 < \varepsilon < 1 \), so that \( \alpha \) is positive. Clearly, \( \alpha \) increases as \( \varepsilon \) decreases, and \( \alpha \) tends to infinity as \( \varepsilon \) tends to zero.

In the methods we consider, we may write

\[
C(p, \alpha) = n c(p),
\]

where \( n \) is the minimal number of steps required and the cost per step \( c(p) \) is the number of arithmetic operations required for the method of order \( p \). As in Traub and Woźniakowski [76], we shall express the cost per step associated with \( \varphi_p \) in the form

\[
c(p) = \alpha(p) + d(p).
\]

Here \( \gamma_p(v) \) is the information about \( v \) required to perform one step of \( \varphi_p \), and we
write $e(\mathcal{G}_p(v))$ for the informational cost of $\varphi_p$; we call $d(p)$ the combinatorial cost of $\varphi_p$.

Note that we explicitly indicate the dependence of $\mathcal{G}_p$ on $v$, so that we may compare the cost of (say) an evaluation of $v$ with a scalar arithmetic operation. Basically, $e(\mathcal{G}_p(v))$ measures the cost of getting new data about $v$ required by $\varphi_p$, while $d(p)$ measures the cost of combining this new data to get an approximate value of the solution at a new point. For example, Euler's method in $\mathbb{R}^N$

$$x_{i+1} = x_i + h v(x_i)$$

has informational cost $\sum_{i=1}^{N} e(v_i)$, where $v_1, \ldots, v_N$ are the components of $v$ and for any function $\omega: \mathbb{R}^N \to \mathbb{R}$, we define

$$e(\omega) := \text{cost of evaluating } \omega \text{ at one point}.$$  

The combinatorial cost is $2^N$ arithmetic operations, i.e., one scalar multiplication and one scalar addition for each of the $N$ components.

We must now face a problem that occurs in almost all areas of complexity theory. The number of operations $c(p)$ required for one step of a $p^{th}$-order method is usually unknown per se; we only have bounds of the form

$$c_L(p) \leq c(p) \leq c_U(p).$$

That is, $c_L(p)$ is a lower bound on the number of operations required per step, usually derived via theoretical considerations, and $c_U(p)$ is an upper bound on the number of operations required per step, which is derived by exhibiting an algorithm for computing the $p^{th}$-order method. (In what follows, we shall assume that the functions $c_L, c_U: \mathbb{R}^+ \to \mathbb{R}^+$ are analytic, although this requirement may be greatly weakened. However, this assumption holds for all examples that we consider.)

From the discussion in Section 3 of Werschulz [76], we find that the step-size $h$ must satisfy
\begin{align*}
\text{(2.9)} \quad h_U(p,a) & \leq h \leq h_L(p,a), \\
\text{where} \quad h_L(p,a) := e_p^{(p)} - h_s(a)/p \quad \text{and} \quad h_U(p,a) := e_p^{(p)} - h_s(a)/p.
\end{align*}

Using (2.5), (2.8), (2.9), and (2.10), we may find bounds on the complexity \( C(p,a) \).

**Theorem 2.1**: Define
\[ C_L(p,a) := f_L(p) e^{a/p}, \quad \text{where} \quad f_L(p) := e_p^{(p)} c_L(p), \]
and
\[ C_U(p,a) := f_U(p) e^{a/p}, \quad \text{where} \quad f_U(p) := e_p^{(p)} c_U(p). \]

Then
\begin{equation}
\text{(2.11)} \quad C_L(p,a) \leq C(p,a) \leq C_U(p,a).
\end{equation}

**Proof**: See Theorem 3.1 of Werschulz [76].

Thus we have bounds on the complexity of using \( \varphi_p \) to compute an approximate solution satisfying (2.4). We now wish to consider the problem of optimality. Define
\begin{equation}
\text{(2.12)} \quad C^*(a) := \inf \{ C(p,a) : \varphi_p \in \Phi \}.
\end{equation}
We are interested in bounds for \( C^*(a) \) under reasonable assumptions about \( f_L \) and \( f_U \).

We first suppose that
\begin{equation}
\text{(2.13)} \quad f_L(p) > 0 \quad \text{and} \quad f_U(p) > 0 \quad \text{if} \quad p > 0
\end{equation}
and
\begin{equation}
\text{(2.14)} \quad \lim_{p \to \infty} f_L(p) = \lim_{p \to \infty} f_U(p) = +\infty.
\end{equation}
Assumption (2.13) is that there is no method whose cost per step is zero, while (2.14) essentially means that the "better" a method is (i.e., the higher its order is), the more we should expect to pay for its use.

Using the techniques of elementary calculus, we find that a necessary condition for \( p \) to minimize \( C_L(\cdot,a) \) is that
\begin{equation}
\text{(2.15)} \quad a = G_L(p) := p^2 f_L''(p) / f_L(p),
\end{equation}
similarly, \( C_L(\cdot, \alpha) \) takes its minimum at \( p \) only if
\[
(2.16) \quad \alpha = G_L(p) := p^2 f_L'(p) / f_L(p) .
\]
Sufficient conditions for the existence and uniqueness of solutions to (2.15) and (2.16)
(i.e., for well-defined functional inverses of \( G_L \) and \( G_U \)) which actually minimize \( C_L(\cdot, \alpha) \)
and \( C_U(\cdot, \alpha) \) are given in

**Lemma 2.1:** Let \( f_L \) and \( f_U \) be as above, and suppose that
\[
(2.17) \quad G_L'(p) > 0 \text{ if } G_L(p) > 0 \text{ and } G_U'(p) > 0 \text{ if } G_U(p) > 0 .
\]
Then \( G_L \) and \( G_U \) have respective functional inverses \( p_L^*, p_U^* : \mathbb{R}^{++} \rightarrow \mathbb{R}^{++} \) such that
for all \( p \in \mathbb{R}^{++} \)
\[
(2.18) \quad C_L^*(\alpha) := C_L(p_L^*(\alpha), \alpha) \leq C_L(p, \alpha)
\]
and
\[
(2.19) \quad C_U^*(\alpha) := C_U(p_U^*(\alpha), \alpha) \leq C_U(p, \alpha)
\]
with equality in (2.18) or (2.19) if and only if \( p = p_L^*(\alpha) \) or \( p = p_U^*(\alpha) \), respectively.

**Proof:** See Theorem 2.1 and Lemma 3.1 of Werschulz [76].

We call \( p_L^*(\alpha) \) (respectively, \( p_U^*(\alpha) \)) the lower (upper) optimal order, \( C_L^*(\alpha) \)
(respectively, \( C_U^*(\alpha) \)) the lower (upper) optimal complexity, and
\[
(2.20) \quad h_L^*(\alpha) := h_L(p_L^*(\alpha), \alpha) \quad \text{(respectively, } h_U^*(\alpha) := h_U(p_U^*(\alpha), \alpha) \text{)}
\]
the lower (upper) optimal step-size. Combining (2.11), (2.12), and Lemma 2.1, we have

**Theorem 2.2:**
\[
C_L^*(\alpha) \leq C^*(\alpha) \leq C_U^*(\alpha).
\]

We next describe the behavior of these quantities as \( \alpha \) increases and tends to
infinity.

**Theorem 2.3:** Let \( f_L \) and \( f_U \) be as in Lemma 2.1. Then \( p_L^*(\alpha), p_U^*(\alpha), C_L^*(\alpha), \) and
\( C_U^*(\alpha) \) all increase monotonically and tend to infinity with \( \alpha \).

**Proof:** See Theorems 2.2 and 3.3 of Werschulz [76].
Finally, we need a restriction of the problem class $D^x\mathcal{Q}$ to "sufficiently difficult" problems; this will allow us to determine $\varepsilon_L$ and thus establish lower bounds. We will assume that

$$v_G(\varphi_p, h) \geq (M_L h)^p \quad \text{if } h \leq 1 \text{ and } p \in \mathbb{Z}^+$$

for some $M_L > 0$ independent of $h$ and $p$. In the methods we study, (2.23) holds provided all sharp upper bounds are attained.
3. Taylor Series Methods

The class $\Theta_T$ of Taylor series methods is defined by expanding $x$ in a truncated Taylor series. Thus the increment function $\varphi_p$ is given by

$$\varphi_p(x;h) := \sum_{k=0}^{p-1} v^{(k)}(x_i) \frac{h^k}{(k+1)!},$$

where

$$v^{(k)}(x_i) := \frac{d}{dt}^k [v(x(t))] \bigg|_{x(t) = x_i}.$$

The usual method of computing (3.2), as described in "classical" numerical analysis texts such as Henrici [62], invokes the chain rule. This quickly leads to expressions of horrifying complexity; for this reason, most texts quickly abandon the discussion of high-order Taylor series methods.

We are interested in faster algorithms for computing $\varphi_p$. First, we address the problem of a lower bound for the combinatorial cost $d(p)$.

**Proposition 3.1:** There exists a constant $a_1 > 0$ such that any sequence of algorithms for computing $\Theta_T$ must satisfy

$$d(p) \geq a_1 p^N.$$

**Proof:** Any algorithm for computing $\varphi_p$ requires the information

$$\mathcal{N}_p(v) := \{D^\alpha v: 0 \leq |\alpha| \leq p - 1\}.$$

(We use the standard multi-index notation found in Friedman [69].) It is then easy to see that the above set has $O(p^N)$ (as $p \uparrow \infty$) distinct elements, which are (generally) independent; this is an immediate consequence of Problem 11 in Chapter I of Pólya and Szegö [25]. Thus (3.3) gives a linear lower bound. \[\square\]

Note that the constant $a_1$ in (3.3) depends on $N$. Since we are treating the case where $N$ is fixed and $p$ is allowed to vary, we will not indicate this dependence explicitly. We now see how close we can get to an optimum value for $d(p)$. 
Theorem 3.1: There exists a constant \( a_U > 0 \) such that the combinatory cost \( d(p) \) of computing \( \varphi_p \in \Phi_T \) satisfies the bound

\[
d(p) \leq a_U p^N \ln (p+e) .
\]

Proof: We first consider the case \( N = 1 \). Note that \( x(h) \) is the zero of

\[
F(z) := \int_{x_0}^{x} d\xi / v(\xi) - h .
\]

As in Brent and Kung [76], we consider the formal power series

\[
P(s) := F(x_0+s) - F(x_0),
\]

where \( s \) is an indeterminate. Let \( V \) be the power series reversion of \( P \). Adopting the notation of Brent and Kung [76], we see that

\[
x(s) = x_0 + V(s) = x_0 + V_p(s) + O(s^{p+1}) .
\]

By the uniqueness of the Taylor coefficients of an analytic function, we see that

\[
\varphi_p(x_0,h) = h^{-1}V_p(h) .
\]

Since the number \( V_p(h) \) can be computed in \( O(p \ln p) \) operations from the Taylor coefficients of \( v \) (by Theorem 6.2 of Brent and Kung [76]), the result for \( N = 1 \) follows.

For \( N \geq 2 \), we use Newton's method (Rall [69]) applied to the formal power series operator \( P \) given by

\[
(Py)(s) := y(s) - x_0 - \int_{0}^{y} v(y(r)) dr ;
\]

clearly, the formal power series \( x(s) \) is the zero of \( P \). The algorithm itself is defined recursively. Let a formal power series \( x(p)(s) \) satisfying

\[
x(p)(s) = x(s) + O(s^{p+1})
\]

be given. Precompute

\[
w(s) := \int_{0}^{s} v(x(p)(r)) dr - x_0 - x(p)(s) + O(s^{2p+2}) ,
\]

\[
Q(s) := \nabla v(x(p)(s)) + O(s^{2p+2}) ,
\]

and let \( u(0)(s) := 0 \). Then set

\[
x(2p+1)(s) := x(p)(s) + u(p+1)(s) ,
\]
where
\[(3.8)\quad u_{(k+1)}(s) := \int_{0}^{s} Q(r) u_{(k)}(r) \, dr + w(s) + O(s^{2p+2}), \quad 0 \leq k \leq p.\]

Following the proof given in Rall [69], we find that
\[x(2p+1)(s) = x(s) + O(s^{2p+2}).\]

We need only consider the cost \(T(p,N)\) of computing the series \(x(p)(s)\) in determining \(d(p)\), since \(x(h)\) may be recovered from the formal power series in \(O(p)\) operations. Clearly, we have the recursion
\[(3.9)\quad T(2p+1,N) \leq T(p,N) + T_6 + T_7 + T_8,\]

where \(T_m\) is the cost of step \((3.m)\) for \(m = 6, 7, 8\). Let \(COMP(p,N)\) be the time required to find the first \(p\) terms of the formal power series \(f(y_1(s), \ldots, y_N(s))\), where \(f, y_1, \ldots, y_N\) are formal power series, and \(y_1, \ldots, y_N\) have zero constant term. Theorem 7.1 of Brent and Kung [76] states that
\[COMP(p,2) = O(p^2 \ln p),\]

and it is easy to show that for any \(N \in \mathbb{Z}^{++}\),
\[COMP(p,N+1) = O(p \, COMP(p,N)).\]

Thus for \(N \geq 2\), we have
\[(3.10)\quad COMP(p,N) = O(p^N \ln p),\]

and so we see that
\[T_6 + T_7 = O((2p+1)^N \ln p).\]

Finally, let \(MULT(p)\) be as in Brent and Kung [76]; we see that
\[T_8 = (p+1) \left[ N^2 \, MULT(2p+1) + O(p) \right] = O((2p+1)^2 \ln p)\]

if Fast Fourier Transform multiplication (Borodin and Munro [75]) is used. Since \(N \geq 2\), we have
\[(3.11)\quad T_6 + T_7 + T_8 = O((2p+1)^N \ln p),\]

and so (3.9) and (3.11) imply that
\[ T(p, N) = O(p^N \ln p), \]

which completes the proof. ■

(Note that the second algorithm is inferior to the first algorithm when applied to the scalar case \( N = 1 \), where we find that the second algorithm requires \( O(p^2 \ln p) \) arithmetic operations.)

We now determine bounds on \( C(p, a) \). First, consider lower bounds. Clearly, there exists \( e_L(v) \geq 0 \) such that

\[ e(\mathcal{D}^i v) \geq e_L(v) \quad (1 \leq i \leq n, |\alpha| \in \mathbb{Z}^+) . \]

Since \( \mathcal{M}_p(v) \) has \( O(p^N) \) elements, there exists a constant \( b_L > 0 \) such that

\[ e(\mathcal{M}_p(v)) \geq b_L e_L(v) p^N . \]

From (3.3) and (3.13), we have a lower-bound cost per step of

\[ c_L(p) = [a_L + b_L e_L(v)] p^N . \]

This leads to

**Theorem 3.2:** \( C_L(p, a) = M_L [a_L + b_L e_L(v)] p^N e^a/p \).

**Proof:** This is an immediate consequence of (2.23) and (3.14). ■

Note that \( f_L(p) := M_L C_L(p) \) satisfies the conditions of Lemma 2.1. Thus, the optimality theory of Section 2 holds. In particular, we have

**Theorem 3.3:** \( C_L^*(a) = M_L [a_L + b_L e_L(v)] (e/N) e^a \).

**Proof:** From (2.18) and (3.14), we find that \( G_L(p) = Np \), so that

\[ p_L^*(a) = e/N \quad \text{and} \quad h_L^*(a) = (M_L e^N)^{-1} . \]

The result follows by letting \( p = p_L^*(a) \) in the definition of \( C_L(p, a) \). ■

However, recall that we assumed that the non-identical mixed partial derivatives of \( v \) are independent. There are a number of systems for which this is not true (for instance, constant coefficient linear systems); for such systems, it is clear that we may
be able to use the extra information of non-independence to find algorithms that are faster than the lower bounds given above. However, we will ignore this case and only consider the problem for a "general" function \( v \).

Next, we turn to upper bounds on the complexity. Theorem 3.1 tells us how to combine the necessary information to get the solution at a new grid-point; we need only measure the cost of getting the information. So, let

\[
e^{(k)}(v) = \max \{ \epsilon(\partial^i v_i); \ 1 \leq i \leq N, |\partial^i| = k \}.
\]

Using the result in Pólya and Szegő [25], we see that

\[
\epsilon(\mathcal{M}_p(v)) \leq N \sum_{k=0}^{p-1} \epsilon^{(k)}(v) (N+k-1)! / [k!(N-1)!].
\]

Unfortunately, the right-hand side of (3.15) does not fit our general model, so we must assume that we know how \( \epsilon^{(k)}(v) \) changes as \( k \) increases. We will consider the case where the cost of derivative evaluation is bounded; that is, we will assume that

\[
\epsilon^{(k)}(v) \leq \epsilon_U(v)
\]

for some \( \epsilon_U(v) \) independent of \( k \). Other cases (e.g., \( \epsilon^{(k)}(v) = O(k^m) \) for some \( m > 0 \)) may be analyzed in a similar manner; of course, they will give different results. By (3.15) and (3.16), there is a \( b_U > 0 \) such that

\[
\epsilon(\mathcal{M}_p(v)) \leq b_U \epsilon_U(v)p^N.
\]

From (3.4) and (3.17), we have an upper-bound cost per step of

\[
C_U(p) = \epsilon_U p^N \ln (p+\epsilon) + b_U \epsilon_U(v)p^N.
\]

This leads to

**Theorem 3.4:** There exists an \( M_U > 0 \) such that

\[
C_U(p,\alpha) = M_U [\epsilon_U p^N \ln (p+\epsilon) + b_U \epsilon_U(v)p^N] \epsilon^\alpha / \alpha.
\]

**Proof:** By Cauchy's Integral Theorem (Ahlfors [66], pg. 122), there exists a \( B > 0 \) such that

\[
\|x^{(k+1)}\| / (k+1)! \leq B^k.
\]
where we define

\[ \|y\| := \max_{t \in I} \|y(t)\| \]

for any \( y : I \to \mathbb{R}^N \). Thus by Section 3.3-3 of Henrici [62], we see that a Lipschitz constant for \( \varphi_p \) in \( \Phi_T \) is given by

\[ \sum_{k=0}^{p-1} \|y(k+1)\| h^k / (k+1)! \leq \sum_{k=0}^{p-1} (Bh)^k \leq L := (1 - Bh_0)^{-1}, \]

provided that \( h \leq h_0 < B^{-1} \). By Section 3.3-2 and 3.3-4 of Henrici [62], there exists a constant for \( \epsilon \) given by

\[ \epsilon = \left( M_U h \right)^p. \]

The result now follows from Theorem 4.1 and (3.18).

We are now ready to consider the optimal \( p \) for \( C_U(p, \varepsilon) \).

**Theorem 3.5:**

1. For all \( \alpha > 0 \), there exists \( p_0^*(\alpha) \) such that (2.19) holds.
2. \( p_0^*(\alpha) \) increases monotonically with \( \alpha \), and
   \[ p_0^*(\alpha) \sim \alpha / N \text{ as } \alpha \uparrow \infty. \]
3. \( C_0^*(\alpha) \) increases monotonically with \( \alpha \), and
   \[ C_0^*(\alpha) \sim M_U a_U (\alpha / N)^N a^N \ln a \text{ as } \alpha \uparrow \infty. \]
4. \( h_0^*(\alpha) \sim (M_U eN)^{-1} \) as \( \alpha \uparrow \infty. \)

**Proof:** Clearly \( c_U \) satisfies (2.13) and (2.14). Now write

\[ G_U(p) = G_1(p) + G_2(p), \]

where

\[ G_1(p) = N p \quad \text{and} \quad G_2(p) = \varepsilon p^2 / D_2(p), \]

where we set

\[ D_2(p) := (p+\varepsilon) \ln (p+\varepsilon) + 1 \quad \text{and} \quad \varepsilon := a_U / \left[ b_U \sigma_U(v) \right]. \]

We see immediately that \( G_1 \) satisfies (2.17); a straightforward calculation shows that

\[ G_2^\prime(p) = \varepsilon \left[ D(p) \right]^{-2} \left[ \varepsilon \ln (p+\varepsilon) + 1 \right] + 2\varepsilon \left[ \varepsilon \ln (p+\varepsilon) + 1 \right], \]
so that $G_2'(p) > 0$ for $p > 0$. Thus $G_2$ satisfies (2.17), which shows that $G_U$ satisfies (2.17). Hence $p_U^*$ and $C_U^*$ behave as described in Theorem 2.2.

Since $p_U^*(a)$ goes to infinity with $a$, we see that

$$a = G_U(p_U^*(a)) \sim N p_U^*(a) + p_U^*(a) / \ln p_U^*(a) \sim N p_U^*(a),$$

which gives the asymptotic estimate in (2.). The rest of the Theorem follows from this estimate. $\blacksquare$

Unfortunately, the estimates given above are only asymptotic as $a \uparrow \infty$; this will be typical, since many of the equations to be solved involve products of logarithmic and polynomial terms, and thus cannot be solved exactly. On the other hand, these asymptotic expressions are sufficient for our purposes, since they describe how quickly $p_U^*(a)$ and $C_U^*(a)$ increase with $a$.

Note that as $a$ tends to infinity, $C_U^*(a)$ becomes independent of $e_U(v)$, which measures how hard it is to evaluate the derivatives of $v$; this is because the combinatorial cost eventually overwhelms the informational cost. This kind of behavior will be typical of the complexity analyses in this paper. Finally, note that the bound

$$(3.20) \quad C_L^*(a) = O(a^N) \leq C^*(a) \leq O(a^N \ln a) = C_U^*(a) \text{ as } a \uparrow \infty$$

implies that

$$C_U^*(a) / C_L^*(a) = O(\ln a) \text{ as } a \uparrow \infty;$$

this indicates the gap in our knowledge of the complexity of solving (1.1) via Taylor series methods.
4. Linear Runge-Kutta Methods

For many functions \( v \), calculation of the derivatives required by Taylor series methods is prohibitively expensive. For this reason, we are interested in methods which use information that is somewhat more readily available. In particular, we will consider methods that use only evaluations of \( v \), combined in a highly structured manner. We say that \( \Phi_{LRK} \) is a class of linear Runge-Kutta methods (abbreviated, "LRK methods") if each increment function \( \varphi_p \) may be written in the form

\[
\varphi_p(x_i, h) := \sum_{i=0}^{s-1} \lambda_i k_i
\]

where

\[
k_i := v(x_i + h \sum_{j=0}^{s-1} \lambda_j k_j) \quad \text{for} \quad 0 \leq i \leq s - 1,
\]

the integer \( s = s(p) \) is said to be the number of stages of \( \varphi_p \); the number of stages is equal to the number of times the vector function \( v \) must be evaluated. (In order to simplify notation, we will not explicitly indicate the dependence of \( \lambda_j \) and \( k_j \) on \( p \).) The method \( \varphi_p \) defined by (4.1) and (4.2) is explicit in that \( k_i \) depends only on \( k_0, \ldots, k_{i-1} \); see Butcher [64] for a discussion of semi-explicit and implicit methods. (We use the adjective "linear" to distinguish these methods from "nonlinear Runge-Kutta methods," which were first proposed in Brent [74].)

Since the function \( \varphi_p \) is (in practice) always evaluated by using the obvious algorithm suggested by its definition, we shall identify an algorithm for evaluating \( \varphi_p \) with \( \varphi_p \) itself. Thus the problem of finding the best algorithm for evaluating \( \varphi_p \) in \( \Phi_{LRK} \) is equivalent to the problem of finding the best basic sequence of LRK methods possible. This is related to the problem of finding the smallest value of \( s(p) \) such that \( \varphi_p \) has order \( p \). This minimal value is given by
For methods of order greater than seven, a gap develops. For instance, eighth-order methods with eleven stages exist, and it is known that any eighth-order method requires at least ten stages. For arbitrary $p \geq 8$, the best bounds known for the optimum value of $s(p)$ are

$$s(p) = \begin{cases} p & p = 1, 2, 3, 4 \\ p + 1 & p = 5, 6 \\ p + 2 & p = 7 \\ \text{unknown} & p \geq 8 \end{cases}$$

(4.3)

where $\theta(p) = c \ln p$ for all sufficiently large $p$ (for some $c > 0$). The lower bound is given in Butcher [75]; the proof is quite involved, and the result is not much better than the "trivial" lower bound $s(p) \geq p$ (Hindmarsh [74], page 84). A class $\Phi_{CVRK}$ of methods such that $e_p$ requires only $(p^2 - 7p + 14)/2$ stages is given in Cooper and Verner [72].

We first consider lower bounds on the complexity $C(p,a)$ using LRK methods. The "trivial" lower bound $s(p) \geq p$ will be used, since the term $\theta(p)$ will be small when $p$ is small and will not affect the asymptotic behavior of optimal order and complexity for $p$ large. It is known (Butcher [64]) that at least $O(p^2)$ of the subdiagonal elements of the matrix $A$ (whose elements are the $a_{ij}$ in (4.2)) must be non-zero in order for $A$ to define a $p^{th}$-order method. Thus there exists $a_L > 0$ such that

$$d(p) \geq a_L p^2$$

(4.5)

since $s(p) \geq p$, we see that

$$e(\tilde{e}_p(v)) \geq Na_L(v)p$$

(4.6)

where we now write

$$e_L(v) := \min_{1 \leq i \leq N} e(v_i)$$
Thus (4.5) and (4.6) show that a lower bound on the cost per step for \( \varphi_p \) is given by

\[
(4.7) \quad c_L(p) = a_L p^2 + N e_L(v) p.
\]

**Theorem 4.1:**

\[
C_L(p,a) = M_L [a_L p^2 + N e_L(v) p] e^a / p.
\]

**Proof:** This follows immediately from (2.23) and (4.7).

It is clear that \( f_1(p) := M_L [a_L p^2 + N e_L(v) p] e^a / p \) satisfies (2.13) and (2.14).

We claim that \( f_L \) yields a \( G_L \) satisfying (2.17). Indeed, write

\[
f_L(p) = f_1(p) f_2(p),
\]

where

\[
f_1(p) := M_L a_L p
\]

and

\[
f_2(p) := p + \gamma, \quad \text{where} \quad \gamma := N e_L(v) / a_L.
\]

Clearly \( f_1 \) yields a \( G_1 \) satisfying (2.17). Since \( f_2 \) is a linear polynomial with a negative zero, it may be shown that \( f_2 \) yields a \( G_2 \) satisfying (2.17). Thus \( f_L \) yields a \( G_L \) satisfying (2.17). In fact, we have

\[
(4.8) \quad G_L(p) = G_1(p) + G_2(p) = p [1 + (1 + \gamma p^{-1})^{-1}].
\]

This leads us to

**Theorem 4.2:**

\[
C_L^*(a) \sim [M_L a_L e^2 / 4] e^a \quad \text{as} \quad a \uparrow \infty.
\]

**Proof:** From (4.8), we see that \( G_L(p) \sim 2 p \) as \( p \uparrow \infty \). Since (2.13), (2.14), and (2.17) hold, \( p_L^*(a) \) tends to infinity with \( a \). Thus

\[
a = G_L(p_L^*(a)) \sim 2 p_L^*(a) \quad \text{as} \quad a \uparrow \infty,
\]

i.e., \( p_L^*(a) \sim a/2 \) as \( a \uparrow \infty \). The result now follows from Theorem 4.1.

We now turn to upper bounds on complexity. The class \( \Phi_{CVRK} \) derived in
Cooper and Verner [72] has two deficiencies, the first of which is that no uniform upper bound on $\varphi_{LU}(\varphi_p,h)$ is known for $\Phi_{CVRK}$; in addition, the combinatory cost for this class of methods is $O(p^4)$ as $p \to \infty$. Instead, we turn to the basic sequence $\Phi_{CRK}$ discussed in the Appendix. There, we prove that there is an $M_U > 0$ such that

\[(4.9) \quad \varepsilon_G(\varphi_p,h) \leq (M_U \ln (p + e) h)^p,\]

provided $h \leq h_p$, where $h_p = O((\ln p)^{-1})$ as $p \to \infty$. Furthermore, there are a large number of extra zeros in the matrix $A$ for $\varphi_p \in \Phi_{CRK}$. Using the notation of the Appendix, we see that the number of non-zero entries in $A$ is

\[
\sum_{i=0}^{s} \xi_i = \sum_{i=1}^{p-1} i^2 + p \leq p^3/3 - p^2/2 + 7p/6 \leq p^3/3 + 2p^2/3
\]

for $p \in \mathbb{Z}^+$. Finally, note that the number of stages $s(p)$ required for $\varphi_p \in \Phi_{CRK}$ is

\[(4.10) \quad s(p) = \left[\frac{(p^2 - 2p + 4)/2}{p^2/2 + p}\right].
\]

for $p \in \mathbb{Z}^+$, which shows that the number of stages required for a $p^{th}$-order method in $\Phi_{CRK}$ asymptotically equals the number requires for a $p^{th}$-order method in $\Phi_{CVRK}$. Thus (considering the combinatory costs), the class $\Phi_{CVRK}$ actually costs more per step than does $\Phi_{CRK}$; ignoring the combinatory costs would have caused us to reach the opposite conclusion.

First, we look at the cost per step. By (4.10), we see that

\[(4.11) \quad e(\varphi_p(v)) \leq \frac{1}{2} (p^2 + p) N \varepsilon_U(v),
\]

where

\[
\varepsilon_U(v) := \max_{1 \leq i \leq N} e(v_i).
\]

Since we are using $\Phi_{CRK}$, it is easy to see that there is a $b_U \geq 2/3$ such that

\[(4.12) \quad d(p) \leq (p^3/3 + b_U p^2) \cdot 2N.
\]
Combining (4.11) and (4.12), we see that the total combinatorial cost per step is bounded by

\begin{equation}
\mathcal{C}(p) = N \left[ 2p^3/3 + \beta_1 p^2 + \beta_2 p \right],
\end{equation}

where

\[ \beta_1 := \frac{e_V}{2+2b_U} \quad \text{and} \quad \beta_2 := \frac{e_U}{2}. \]

Using (4.9) and (4.13) gives

\begin{align*}
\mathcal{C}(p,a) &= M_U N \left[ 2p^3/3 + \beta_1 p^2 + \beta_2 p \right] \ln(p + a) e^{-a/p}. 
\end{align*}

Now we look at the optimality theory for the upper bound.

\textbf{Theorem 4.3:}

\begin{enumerate}
\item For all \( a > 0 \), there exists \( p_U^*(a) \) such that (2.19) holds.
\item \( p_U^*(a) \) increases monotonically with \( a \), and \( p_U^*(a) \sim a/3 \) as \( a \to \infty \).
\item \( C_{U_U}(a) \) increases monotonically with \( a \), and \( C_n^U \sim \frac{2}{a} \ln a - \frac{2}{a} \ln a \) as \( a \to \infty \).
\item \( h_{U_0}(a) \sim \left( \frac{M_U e^2}{a} \right) \) as \( a \to \infty \).
\end{enumerate}

\textbf{Proof:} We write

\[ f_U(p) := M_U \ln(p + a) \mathcal{C}(p) \]

in the form

\[ f_U(p) = f_1(p) f_2(p), \]

where

\[ f_1(p) = M_U N p \ln(p + a) \quad \text{and} \quad f_2(p) = 2p^2/3 + \beta_1 p + \beta_2. \]

It is clear that \( f_1 \) satisfies the hypotheses of Lemma 2.1. Now we consider \( f_2 \). Clearly \( f_2 \) has no positive zeros; it may be seen that the condition \( b_U \geq 2/3 \) implies that \( f_2 \) has
a positive discriminant and hence has no complex roots. Thus \( f_2 \) has only negative roots; one may then show that this guarantees that \( f_2 \) satisfies the hypotheses of Lemma 2.1. Thus, the same may be said for \( f = f_1 f_2 \).

Thus \( p_U^* \) and \( C_U^* \) behave as described in (1.) of Theorem 2.3. We also see that \( G_U(p) \sim 3 p \) as \( p \to \infty \). Thus the estimate in (2.) holds, from which we get the estimates in (3.) and (4.).

So in the class of linear Runge-Kutta methods, we find that

\[
C_U^*(a) = O(a^2) \leq C^*(a) \leq C_U^*(a) = O(a^3 \ln a)
\]

as \( a \) tends to infinity; hence, the ratio

\[
\frac{C_U^*(a)}{C_L^*(a)} = O(a \ln a)
\]

indicates the gap in our knowledge of the complexity of linear Runge-Kutta methods.
5. Comparison of the Methods

We now wish to compare the classes of Taylor series methods and LRK methods. Write $C_{U,T}^*$, $C_{L,T}^*$, and $C_T^*$ (respectively, $C_{U,LRK}^*$, $C_{L,LRK}^*$, and $C_{LRK}^*$) for $C_{U}^*$, $C_{L}^*$, and $C^*$ in the class $\Phi_T$ (respectively, the class $\Phi_{LRK}$). Since we have only asymptotic expressions for these quantities, we are forced to use an asymptotic comparison. If $f, g : \mathbb{R}^+ \to \mathbb{R}^+$ satisfy $\lim_{a \to \infty} f(a) = \lim_{a \to \infty} g(a) = +\infty$, we will write

\begin{equation}
(5.1) \quad f \prec g \iff f(a) = o(g(a)) \text{ as } a \to \infty;
\end{equation}

we say that $f$ is asymptotically less than $g$. If $f \prec g$, there is an $a_0 > 0$ such that $f(a) < g(a)$ for $a > a_0$, so there is a non-asymptotic interpretation of the order relation $\prec$. Thus if $f$ and $g$ are cost functions, the statement "$f \prec g$" implies that the method whose cost is given by $f$ is "better" (i.e., cheaper) than the method whose cost is given by $g$, for $s$ sufficiently small. Using the results of (3.20) and (4.14), we then have the following.

**Theorem 5.1:** Suppose that (3.16) holds.

1. If $N = 1$, then $C_{U,T}^* < C_{L,LRK}^*$.
2. If $N = 2$, then $C_{L,T}^* < C_{U,LRK}^*$.
3. If $N = 3$, then

   \[ C_{U,T}^*(a) = O(C_{U,LRK}^*(a)) \]

   \[ C_{U,LRK}^*(a) = O(C_{U,T}^*(a)) \]

   as $a \to \infty$.
4. If $N \geq 4$, then $C_{U,LRK}^* < C_{L,T}^*$.
If (3.16) does not hold, then (1.), (2.), and (3.) may be false, but (4.) will certainly be true. As an immediate corollary to the above theorem, we have

**Theorem 5.2:**

1. If \( N = 1 \) and (3.16) holds, then \( C_T^* < C_{LRK}^* \).

2. If \( N \geq 4 \), then \( C_{LRK}^* < C_T^* \).

So if the derivatives of \( v \) are cheap to evaluate, we see that the best Taylor series method known is better than the best linear Runge-Kutta method possible for the scalar case \( N = 1 \); but if \( N \geq 4 \), the best linear Runge-Kutta method known is better than the best Taylor series method possible.
Appendix: Error Bounds for a Sequence of LRK Methods

In this Appendix, we describe a subclass of a class of linear Runge-Kutta ("LRK") methods due to Cooper [69]. We shall first prove the following.

Theorem A.1: There is a basic sequence \( \Phi_{\text{CRK}} \) of LRK methods such that

1. Each \( \varphi_p \in \Phi_{\text{CRK}} \) requires
   \[
   s(p) = \frac{(p^2 - p + 2)}{2}
   \]
evaluations of \( v \) per step.

2. There exists an \( M > 0 \) such that
   \[
   \varepsilon_G(\varphi_p, h) \leq (M \ln (p+e)) h^p
   \]
for \( h \leq h_p = O((\ln p)^{-1}) \).

We use the notation of Cooper and Verner [72]. Let \( p \in \mathbb{Z}^+ \) be given; define \( \rho: \mathbb{Z}^+ \to \mathbb{Z}^+ \) by

\[
\rho(j) := \begin{cases} 
\sum_{k=0}^{j} k = j(j+1)/2 & \text{if } j < p \\
\frac{s}{1} & \text{if } j = p,
\end{cases}
\]

where we write "s" for "s(p)" as defined above. Next, a set \( \{\xi_0, \ldots, \xi_s\} \) of integers is defined by picking \( \xi_0 := p \), and setting \( \xi_i \) (\( i \neq 0 \)) to be the unique integer in \( [1, p] \) satisfying

\[
\rho(\xi_i - 1) < i \leq \rho(\xi_i).
\]

We now pick \( u_0, \ldots, u_s \in I \) satisfying

\[
u_0 = 0, \quad u_s = 1, \quad u_i \neq 0 \quad \text{if } i \neq 0
\]

and

\[
(i \neq j \text{ and } i \neq j) \quad \text{implies} \quad u_i \neq u_j.
\]

Finally, we pick a matrix of coefficients \( A := \{a_{ij}: 0 \leq j \leq i-1, 1 \leq i \leq s\} \) such that

\[
a_{ij} = 0 \quad \text{if } \xi_i < \xi_j - 1 \quad (1 \leq i, j \leq s)
\]
Cooper and Verner [72] point out that these conditions may always be fulfilled; the resulting $A$ defines a $p^\text{th}$-order LRK method with $s$ stages.

We are interested in a choice of $u_0, \ldots, u_s$ which will give a small error coefficient. To this end, we will choose

\[(A.8) \{u_j: \xi_j = n \} = \{(1 + x_{kn}) / 2: 1 \leq k \leq n\} \quad (1 \leq n \leq s - 1),\]

where $x_{1n}, \ldots, x_{n^2}$ are the zeros of the Jacobi polynomial $P_n := P_n^{(1,1)}$ (see Szegö [59]). Since these zeros are distinct and lie in $[-1, 1]$, conditions (A.4) and (A.5) may be satisfied.

Now we are able to exhibit a solution to the $i^\text{th}$ system in (A.7). First, note that the equation for $r = 0$ may be separated from the others, since $u_0 = 0$. Setting

\[n := \xi_i - 1,\]

we see that

\[(A.9) \lambda_{i0} = u_i - \sum_{j=1}^{i-1} \lambda_{ij} = u_i - \sum \{ \lambda_{ij}: j < i \text{ and } \xi_j \geq n \},\]

the last by (A.6). We wish to determine the nonzero $\lambda_{ij}$, i.e., those $\lambda_{ij}$ for which $\xi_j \geq n$ and $j < i$. So setting

\[\lambda_{ij} = 0 \quad \text{unless} \quad j \in \{i_1, \ldots, i_n\},\]

we see that the remaining $\lambda_{ij}$ are the solution of the system

\[\sum_{k=1}^{n} u_{jk} \lambda_{ij} = (\tau + 1)^{-1} u_{ij}^{\tau+1} \quad (1 \leq \tau \leq n).\]

Thus the $\lambda_{ijk}$ are the weights for an interpolatory quadrature formula on $[0, u_i]$ with abscissae $u_{j1}, \ldots, u_{jn}$. From the usual expression for such weights and (A.6), we see that

\[\lambda_{ijk} = \pi_{ikn} := [2P_n'(\cos \phi_{kn})]^{-1} \int_{\phi_{i,n+1}}^{\pi} [P_n(\cos \phi) / (\cos \phi - \cos \phi_{kn})] \sin \phi \, d\phi,
\]

where $x_{kn} = \cos \phi_{kn} (1 \leq k \leq n)$. 
**Lemma A.1:** \( m_{ikn} = O(n^{-1} \ln n) \) as \( n \to \infty \).

**Proof:** Since the zeros of \( P_n \) are symmetric about the origin, we may assume that \( 0 < \phi_{kn} \leq \pi/2 \). Using (8.9.2) of Szegö [59], we then find

\[
m_{ikn} = O(k^{5/2}n^{-3}) \int_{\phi_{i,n+1}}^{\pi} \left[ P_n(\cos \theta) / (\cos \theta - \cos \phi_{kn}) \right] \sin \theta \, d\theta.
\]

**Case 1:** \( \phi_{i,n+1} \leq \phi_{i,n+1} \leq \theta_{k,n+1}/2 \). We consider the integral over \([\phi_{i,n/2}, \phi_{i,n+1}]\), since Theorem 15.4 of Szegö [59] proves that

\[
O(k^{5/2}n^{-3}) \left| \int_{0}^{\pi} \left| \int_{0}^{\phi_{i,n/2}} \right| \right| = O(n^{-1}).
\]

(Here the integrand is the same as in the preceding integral.) But the proof of (15.4.12) in Szegö [59] extends almost immediately to a proof that the remaining integral is \( O(k^{-2}n) \), since (15.4.12) is proved by order-of-magnitude estimates. Thus \( m_{ikn} = O(n^{-1}) = O(n^{-1} \ln n) \) for Case 1.

**Case 2:** \( \theta_{k,n+1}/2 \leq \phi_{i,n+1} \leq 3\theta_{k,n+1}/2 \). We consider the integral over \([\phi_{kn}/2, \phi_{i,n+1}]\), since Szegö [59] shows that

\[
O(k^{5/2}n^{-3}) \left| \int_{\phi_{kn}/2}^{\phi_{i,n+1}} \right| = O(n^{-1}).
\]

As in (15.4.13) of Szegö [59], we have

\[
\int_{\phi_{kn}/2}^{\phi_{i,n+1}} = O(nk^{-3/2}) I_1 + I_2.
\]

Here

\[
I_1 := \int_{\phi_{kn}/2}^{\phi_{i,n+1}} D(\phi) \sin \phi \, d\phi,
\]

with

\[
D(\phi) := \frac{[\cos (N\phi + \gamma) - \cos (N\phi_{kn} + \gamma)]}{[\cos \phi - \cos \phi_{kn}]},
\]

where \( N := n + 3/2 \) and \( \gamma := -3\pi/4 \), and

\[
I_2 := \int_{\phi_{kn}/2}^{\phi_{i,n+1}} R_n(\phi, \phi_{kn}) \sin \phi \, d\phi = O(nk^{-3/2}),
\]

with \( R_n \) the remainder term in (8.8.2) of Szegö [59]. Unfortunately, the proof that (15.4.14) of Szegö [59] is bounded does not extend to a proof that \( I_1 \) is bounded,
since the proof of the former requires that the interval of integration be symmetric about \( \phi_{kn} \). However, it is straightforward to verify that

\[
I_1 = O(1) \int_0^{\pi/4} |\sin N\phi / \phi| d\phi = O(\ln n).
\]

Thus \( \mu_{ikn} = O(n^{-2k \ln n}) = O(n^{-1} \ln n) \) for Case 2.

**Case 2:** \( 3\phi_{kn+1} \leq \phi_{i,n+1} \leq 3\pi/4 \). We consider the integral over \([3\phi_{kn}/2, \phi_{i,n+1}]\), since Szegő [59] proves that

\[
O(k^{5/2} n^{-3}) \int_{3\phi_{kn}/2}^{\pi} \, | = O(n^{-1}).
\]

But the proof of (15.4.19) in Szegő [59] extends to prove that the remaining integral is \( O(k^{-5/2}/n) \) (as in Case 1). Thus \( \mu_{ikn} = O(n^{-1}) = O(n^{-1} \ln n) \) for Case 3.

**Case 4:** \( 3\pi/4 \leq \phi_{i,n+1} \leq 3\pi/4 \). We consider the integral over \([3\pi/4, \phi_{i,n+1}]\), since Szegő [59] shows that

\[
O(k^{5/2} n^{-3}) \int_{3\pi/4}^{\pi} \, | = O(n^{-1}).
\]

As in Cases 1 and 3, the proof of the above may be extended to prove a similar bound on the integral of interest. Thus \( \mu_{ikn} = O(n^{-1}) = O(n^{-1} \ln n) \) in Case 4, completing the proof of the Lemma.

Thus (A.9) and Lemma A.1 show the existence of a \( \lambda > 0 \) such that

\[
(A.10) \quad \sum_{j=0}^{s-1} |\kappa_{ij}| \leq \lambda \ln (\xi + \varepsilon);
\]

here \( \lambda \) is independent of \( p \). Moreover, the result for the case \( i = s \) may be sharpened.

We see that \( \lambda_{sj} \geq 0 \), since the \( u_j \) for the \( s \)th system in (A.7) are the abscissae for Lobatto quadrature. Thus

\[
(A.11) \quad \sum_{j=0}^{s-1} |\lambda_{sj}| = \sum_{j=0}^{s-1} \lambda_{sj} = 1,
\]

the consistency condition in the last equality being a consequence of (A.7) with \( \varepsilon = 0 \).

**Proof of Theorem A.1:** As in Cooper and Verner [72], we define

\[
s_i := \hat{x}(u_i) - k_i
\]

and
\[ \delta_i := \int_0^{u_i} \hat{s}(u h) \, du - \sum_{j=0}^{i-1} \lambda_{ij} \hat{s}(u_j h) \]

for \( 0 \leq i \leq s_i \); note that \( \delta_0 = \delta_1 = 0 \). Let \( z(h) \) be the computed approximation to \( x(h) \); then

\[
\begin{align*}
    h^{-1} \| x(h) - z(h) \| &= \| h^{-1} [x(h) - x(0)] - \sum_{i=0}^{s_i-1} \lambda_{bi} \delta_i \| \\
    &\leq \| \beta_i \| + \| \sum_{i=0}^{s_i-1} \lambda_{bi} \delta_i \| \\
    &\leq \| \beta_i \| + \max_{i} \| \delta_i \| + \| \beta_i \| = p^{-1} \| \beta_i \|,
\end{align*}
\]

the last by (A.6) and (A.11). By the analyticity of \( x \), there is an \( A_1 > 0 \) such that

\[ \beta_i := h^{-1} \| x(u_i h) - \sum_{i=0}^{s_i-1} (u_i h)^P x^{(P)}(0) / P! \| \leq (A_1 h)^{\delta_i} \]

and

\[ \gamma_{ij} := \| \hat{s}(u_j h) - \sum_{i=0}^{s_i-1} (u_j h)^P \hat{s}^{(P)}(0) / P! \| \leq (A_1 h)^{\delta_i} \]

so that the definition of \( \delta_i \) gives

\[
\begin{align*}
    \| \delta_i \| &\leq \beta_i + \sum_{j=0}^{i-1} |\lambda_{ij}| \gamma_{ij} \\
    &\leq (A_1 h)^{\delta_i} + \sum_{j=0}^{i-1} |\lambda_{ij}| (A_1 h)^{\delta_i} \\
    &\leq (A_2 h)^{\delta_i}
\end{align*}
\]

for a suitable \( A_2 > 0 \). Thus (A.12) becomes

\[
(A.14) \quad h^{-1} \| x(h) - z(h) \| \leq (A_2 h)^P + \max_{i} \| \delta_i \| = p^{-1} \| \delta_i \|.
\]

We now use Lemma 1.1 of Cooper and Verner (72) and (A.6) to find that if \( L \) is a Lipschitz constant for \( v \), then there exists \( A_3 > 0 \) such that

\[
\begin{align*}
    \| e_i \| &\leq h L \| \beta_i \| + h L \sum_{j=0}^{i-1} |\lambda_{ij}| \max_j \| e_j \| \\
    &\leq (A_3 h)^{\delta_i+1} + (A_3 h) \ln (\delta_i + e) \max_j \| e_j \|,
\end{align*}
\]

the last by (A.10) and (A.13); here, the maximum is taken over all \( j < i \) such that \( \delta_j \geq \delta_i - 1 \). A straightforward induction shows that if \( (1 + \ln 2) A_3 h < 1 \), then

\[ \| e_i \| \leq (A_4 \ln (\delta_i + e) h)^{\delta_i+1} \]

for a suitable \( A_4 > 0 \). Combining this with (A.14), we find

\[
(A.15) \quad h^{-1} \| x(h) - z(h) \| \leq (A_5 \ln (p+e) h)^P,
\]
the desired bound for the local error for a single unit step.

To extend (A.15) to a global error result, we must look at the Lipschitz constants for the increment functions. Let \( L \) be a bound on \( \| \nabla \psi \| \), and write "\( \nabla \psi \)" to indicate gradient with respect to the vector variable \( y \). Now

\[
\| \nabla \psi_p(y,h) \| \leq \sum_{i=0}^{s-1} |k_i| \max_0 \| k_i(y,h) \|,
\]

where we write "\( k_i(y,h) \)" to indicate the dependence of \( k_i \) upon \( y \) and \( h \). By the definition of \( k_i(y,h) \), we find

\[
\nabla k_i(y,h) = \nabla \nabla^T - \sum_{j=0}^{s-1} \lambda_j \nabla k_j(y,h),
\]

where \( u := y + h \sum_{j=0}^{s-1} \lambda_j k_j(y,h) \) and \( I_{N\times N} \) is an \( N\times N \) identity matrix. Taking norms in the above gives the result

\[
f_j \leq L \lambda + h L \lambda \left\{ \ln \left( \frac{\lambda + e}{\lambda + e - k} \right) \max \{ f_j : j < i \text{ and } f_j \geq f_i - 1 \} \right\},
\]

where \( f_j := \| \nabla k_j(y,h) \| \). Writing \( \lambda_p \) for the Lipschitz constant for \( \psi_p \), it is easy to see that (A.16) and the above inequality imply

\[
\lambda_p \leq \sum_{j=0}^{p-1} (h \lambda)^j \prod_{k=1}^{j-1} \ln \left( \frac{\lambda + e - k}{\lambda + e - k - 1} \right),
\]

which is bounded for all \( p \), provided that \( h \leq h_p < \frac{\lambda \ln(p+e)}{p+e-1} \). Thus (A.1) follows from this result, (A.15), and Theorem 3.3 of Henrici [62].

The value for \( s(p) \) indicated in Theorem A.1 may be improved somewhat by noting that since we are using a Lobatto quadrature, higher order may be expected with fewer steps. Indeed, if we use the strategy outlined in the comments following Theorem 4 of Cooper and Venter [72], we have

**Theorem A.2**: There exists a basic sequence \( \Phi_{CRK} \) of LRK methods such that (A.1) holds and \( \psi_p \) requires

\[
s(p) := \left( \frac{p^2 - 2p + 4}{2} \right)
\]

evaluations of \( \psi \) per step.
Acknowledgements

I would like to thank Professor R. P. Brent of the Australian National University, Professors H. T. Kung and J. F. Traub of Carnegie-Mellon University, and Professor H. Woźniakowski of the University of Warsaw for their comments and suggestions on the results reported in this paper. In addition, a number of the results in Sections 3 and 4 were obtained by using the MACSYMA system developed by the Mathlab group at Massachusetts Institute of Technology, which is supported by the Defense Advanced Research Projects Agency work order 2095, under Office of Naval Research Contract N00014-75-C-0661.

References

Ahlfors [66]:

Borodin and Munro [75]:

Brent [74]:

Brent and Kung [76]:

Buck [65]:
Butcher [64]:

Butcher [75]:

Cooper [69]:

Cooper and Verner [72]:

Friedman [69]:

Gurtin [75]:

Henrici [62]:

Hindmarsh [74]:

Pólya and Szegő [25]:

Rall [69]:

Stetter [73]:

Szegő [59]:
Traub and Woźniakowski [76]:

Werschulz [76]:
The problem is to calculate an approximate solution of an initial value problem for an autonomous system of N ordinary differential equations. Using fast power series techniques, we exhibit an algorithm for the pth-order Taylor series method requiring only \(O(p^N \ln p)\) arithmetic operations per step as \(p \to +\infty\). (Moreover, we show that any such algorithm requires at least \(O(p^N)\) operations per step.) We compute the order which minimizes the complexity bounds for Taylor series and linear Runge-Kutta methods, and show that in all cases, this optimal order increases as the error criterion \(\varepsilon\) decreases, tending to infinity as \(\varepsilon\) tends to zero. Finally, we show...
that if certain derivatives are easy to evaluate, then Taylor series methods are asymptotically better than linear Runge-Kutta methods for problems of small dimension $N$. 