Contents

List of Figures

List of Tables

Chapter 13

Ordinary Differential Equations

13.1 Initial Value Problems for ODEs. Theoretical Background

13.1.1 Introduction

To start with, we shall study the following problem: given a function $f(t, y)$, find a function $y(t)$ which for $a \leq t \leq b$ is an approximate solution to the **initial value problem** for the ordinary differential equation or, with an established abbreviation, the ODE,

$$\frac{dy}{dt} = f(t, y), \qquad y(a) = c.$$

In mathematics courses, one learns how to determine exact solutions to this problem for certain special functions f. An important special case is when f is an affine function of y. However, for most differential equations, one must be content with approximate solutions.

Many problems in science and technology lead to differential equations. Often, the variable t means time, and the differential equations expresses the rule or law of nature which governs the change in the system being studied. In order to simplify the language in our discussions, we consider t as the time, and we use terms like time step, velocity etc. However, t can be a spatial coordinate in some applications.

As a rule, one has a **system of first-order differential equations** and initial conditions for several unknown functions, $y_1, y_2, \ldots, y_s$, say, where

$$\frac{dy_i}{dt} = f_i(t, y_1, \ldots, y_s), \qquad y_i(a) = c_i, \qquad i = 1, 2, \ldots, s.$$

It is convenient to write such a system in vector form

$$\frac{dy}{dt} = f(t, y), \quad y(a) = c, \quad (f : \mathbf{R} \times \mathbf{R}^s \to \mathbf{R}^s), \tag{13.1.1}$$

where now

$$y = (y_1, \ldots, y_s)^T, \quad c = (c_1, \ldots, c_s)^T, \quad f = (f_1, \ldots, f_s)^T$$

are column vectors. When the vector form is used, it is just as easy to describe numerical methods for systems as it is for a single equation.

Often it is convenient to assume that the system is given in **autonomous form**

$$\frac{dy}{dt} = f(y), \quad y(a) = c, \quad (f : \mathbf{R}^s \to \mathbf{R}^s) \tag{13.1.2}$$

i.e., f does not depend explicitly on t. A non-autonomous system is easily **augmented** to autonomous form by the addition of the trivial extra equation,

$$\frac{dy_{s+1}}{dt} = 1, \qquad y_{s+1}(a) = a,$$

which has the solution $y_{s+1} = t$. [1]

Unless it is stated otherwise, we shall *only consider numerical methods that produce identical results (in exact arithmetic) for a non-autonomous system and for this augmented autonomous system.* The use of the autonomous form in the analysis and in the description of numerical methods is usually no restriction.

Wherever it is necessary or convenient, we shall return to the non-autonomous formulation. For example, a linear system with variable (t-dependent) coefficients is best discussed in the non-autonomous formulation, because the augmented system is no longer linear.

We shall mostly write $\dot{y},\ \ddot{y}$ instead of dy/dt, d^2y/dt^2, (and analogously for variables denoted by an other character than y). So Eqn. (13.1.2) often reads $\dot{y} = f(y)$. $y',\ y''$, may be used, only if the dot notation would be awkward, for typographical reasons, since we prefer to reserve primes for derivatives with respect to a vector argument, e.g. $f'(y) = \partial f/\partial y$, see the notation used in Ch. 12. If $k > 2$, we usually write, e.g., $d^k y/dt^k$ or $u^{(k)}$.

Also higher-order differential equations can be rewritten as a system of first-order equations:

Example 13.1.1 The initial value problem

$$u^{(3)} = g(t, u, \dot{u}, \ddot{u}), \quad u(0) = c_1,\ \dot{u}(0) = c_2,\ \ddot{u}(0) = c_2,$$

is, by the substitution $y = (y_1, y_2, y_3)^T$, where $y_1 = u,\ y_2 = \dot{u},\ y_3 = \ddot{u}$, transformed into the system

$$\dot{y} = \begin{pmatrix} y_2 \\ y_3 \\ g(t, y_1, y_2, y_3) \end{pmatrix}, \qquad y(0) = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix}.$$

Most programs for initial value problems for ODEs are written for non-autonomous first order systems. So, this way of rewriting a higher order system is of practical importance. The transformation to autonomous form mentioned above is, however, rarely needed in practice, but it gives a conceptual simplification in the description and the discussion of numerical methods.

[1] It is sometimes more convenient to call the extra variable y_0 instead of y_{s+1}.

If t denotes *time*, then the differential equation (13.1.2), determines the velocity vector of a particle as a function of time and the position vector y. Thus, the differential equation determines a **velocity field**, and its solution describes a **motion** in this field along an **orbit** or a **path** in $\mathbf{R}^s$. The point set $\{y(t), t\} \in \mathbf{R}^s \times \mathbf{R}$, is called the **solution curve**. [2]

In the examples of this chapter, we shall see graphical representations in a plane both of orbits and of motions, and of one or several components of a solution curve (y_i versus t).

For a non-autonomous system, the velocity field changes with time. You need time as an extra coordinate for visualizing all the velocity fields, just like the standard device mentioned above for making a non-autonomous system autonomous.

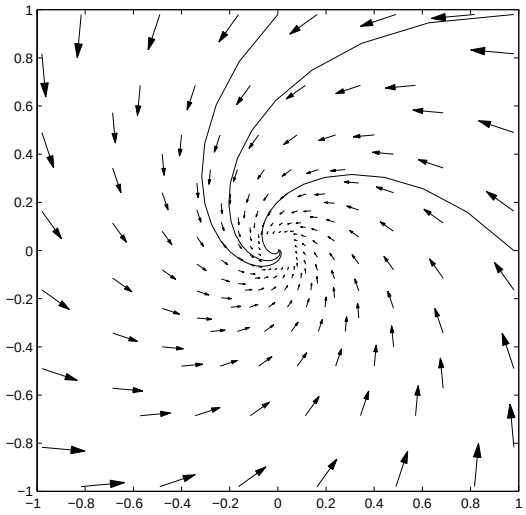

Figure 13.1.1. *Velocity field of the two-dimensional system in Example 13.1.2 .*

Example 13.1.2 The velocity field of the autonomous two-dimensional system

$$\begin{aligned} \dot{y}_1 &= -y_1 - y_2, \\ \dot{y}_2 &= y_1 - y_2, \end{aligned} \tag{13.1.3}$$

is shown in Fig. 13.1.1. The solution of the equation describes the motion of a particle in that field. For various initial conditions, we get a whole *family of solution curves*. Three such curves are shown, for

$$(y_1(0), y_2(0)) = (1, 0), (1, 1), (0, 1).$$

This interpretation is directly generalizable to three dimensions, and these geometric ideas and terms are also suggestive for systems of more than three dimensions.

[2] There are in the literature several different terminologies for these concepts.

It is useful to bear in mind the simple observation that every point of an orbit gives the initial value for the rest of the orbit. "Today is the first day of the rest of your life." Also note that the origin of time is arbitrary for an *autonomous system*: if $y(t)$ is one solution, then $y(t+k)$ is also a solution for any constant k. [3]

It seems plausible that the motion in a given velocity field is uniquely determined by its initial position, provided that the velocity field is sufficiently well-behaved. This statement will be made more precise below (see Theorem 13.1.3). In other words: *under very general conditions, the initial value problem defined by* (13.1.2) *has exactly one solution.* The picture also suggests that if one chooses sufficiently small step size, the rule,

$$\text{Displacement} = \text{Step size} \times \text{Mean velocity over a time step},$$

can be used for a step-by-step construction of an approximate solution. In fact, this is the basic idea of most methods for the numerical integration of ODEs.

More or less sophisticated approximations of the "mean velocity" over a time step yield different methods, sometimes presented under the name of *dynamic simulation*. The simplest one is **Euler's method** that was described already in Chapter 1. We assume that the reader has a clear idea of this method. The methods we shall treat are such that one proceeds step by step to the times $t_1, t_2, \ldots$, and computes approximate values, $y_1, y_2, \ldots$, to $y(t_1), y(t_2), \ldots$. We shall distinguish two classes of methods:

(a) One-step methods, where y_n is the only input data to the step in which y_{n+1} is to be computed. Values of the function $f(y)$ (and perhaps also some of its total time derivatives) are computed at a few points close to the orbit. They are used in the construction of an estimate of y_{n+1}, usually much more accurate than Euler's method would have given with the same step size. An example of a one-step method was given in Sec. 1.4, namely **Runge's 2nd order method**, with two function evaluations per step:[4]

$$k_1 = hf(t_n, y_n); \quad k_2 = hf(t_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_1); \quad y_{n+1} = y_n + k_2. \tag{13.1.4}$$

The step size is chosen so that $k_2 - k_1 \approx 3\,\text{sc tol}$, where TOL is a tolerance chosen by the user. For details, see Sec. 1.4, Theorem 13.2.1, Example 13.2.1 and Sec 13.3. We shall there see a bound for the *global* error that is proportional to TOL; the factor of proportionality depends on certain quantities that describe important features of the function f.

The reader is advised to use the implementation of this that is, together with a few auxiliary programs, available on the web, for the computer exercises of this section. The accuracy required is, as a rule, obtained with much longer steps and less work than Euler's method needs. We shall see, however, that there are exceptions from this rule, due to the possibility of numerical instability, so-called **stiffness**, an important phenomenon that will be discussed in Section 2 and in later sections.

[3] This statement must be modified in an obvious way if $y(t)$ exists on a finite interval only.

[4] It has also been called Heun's method or the improved Euler method, but these names are also used for other methods.

(b) Multistep methods, where longer time steps are made possible, or the accuracy is improved by the use of several values from the past: $y_n, y_{n-1}, \ldots, y_{n-k+1}$. Shampine [31, p.171] calls them methods with memory. In Sec. 3.2, Problem 9 two important families of methods, due to Adams (explicit and implicit), were derived by operator techniques and applied to a simple problem. The explicit 3rd order accurate version of Adams method reads in terms of backward differences:

$$y_{n+1} - y_n = h(y'_{n+1} + \tfrac{1}{2}\nabla y'_{n+1} + \frac{5}{12}\nabla^2 y'_{n+1}). \tag{13.1.5}$$

The next term of the expansion, $\frac{3}{8}\nabla^3 y'_{n+1}$, is used in the step size control. Details are given in Sec.13.4, together with other facts about multistep methods. Numerical stability is a particularly important issue for them. A k-step method requires some special arrangement for starting, because only one of the k initial values that the difference equation needs, is given for the differential equation. One can, e.g., obtain them by computing $k-1$ steps by a one step method. Another possibility is to use the procedure that most implementations of such methods has for the automatic variation of the order k and the step size h. The computations can therefore start with$k = 1$ and a very small step size. Then these quantities are gradually increased.

It is important to distinguish between **global and local error**. Let $y(t; t^*, y^*)$ denote the exact solution of the system $\dot{y} = f(y)$, which satisfies the condition $y(t^*) = y^*$; in other words $y(t; t^*, y^*)$ is the motion that passes through the point (t^*, y^*).

The **global error** at the point (t_{n+1}, y_{n+1}) is $y_{n+1} - y(t_{n+1}; t_0, y_0)$, while the **local error** at the same point is $y_{n+1} - y(t_{n+1}; t_n, y_n)$, for a one-step method. In other words, the local errors are the jumps in the staircase curve in Fig. 13.1.3.

The local error for a multistep method can also be defined by

$$y_{n+1} - y(t_{n+1}; t_n, y_n),$$

but, in order to make it uniquely determined at the point (t_n, y_n) we must, in the definition, use the values $y^*_{n-j} = y(t_{n-j}; t_n, y_n)$, $j = 1 : k-1$, instead of the actually computed values y_{n-j} from the past. (The actual value of y_n is used for $j = 0$.) [5]

The distinction between the global and local error is made already for the computation of integrals (Chapter 5), which is, of course, a special case of the differential equation problem. There the global error is simply the sum of the local errors. We saw, e.g., in Sec. 8.2, that the global error of the trapezoidal rule is $O(h^2)$, while its local error is $O(h^3)$, although this terminology was not used there. The reason for the difference in order of magnitude of the two errors is that the number of local errors which are added is inversely proportional to h.

For differential equations, the error propagation mechanism is, as we shall see, more complicated; but even here it holds that *the difference between the exponents in the asymptotic dependence on h of the local and global errors is equal to 1*, for first order differential systems. For the sake of simplicity, let us consider a computation with constant (though arbitrary) step size h. The local error at a point (t, y), where

[5] There are alternative definitions, see, e.g., Hairer, Nørsett and Wanner,(1993), vol.1, p.368. A related concept is the *local truncation error*, see §13.3.4.

the motion is smooth is, for some p, proportional to h^{p+1}, asymptotically as $h \to 0$. For a particular method, the smallest value of p that can be obtained at such a point is called the **order of consistency** of the method, [6] and from now on *this* is denoted by p. We say that a method is **consistent** iff $p \geq 1$.

For most numerical methods in practical use p also becomes the **order of accuracy** of the method, in the sense that *the global error approaches zero at the rate of* h^p, if we consider an ensemble of numerical computations with decreasing values of h over a finite interval, where the initial value problem in question has a sufficiently smooth solution. This statement supposes certain stability properties of the numerical method; methods that do not satisfy them are no longer in practical use. We shall later prove results of this type for some important families of numerical methods.

There exist one-step methods and multistep methods of any order of accuracy. Although this definition of the order p is based on the application of the method with constant step size, it makes sense to use the same value also for computations with variable step size, under certain conditions.

As mentioned above, the systematic discussion of one-step methods and multistep methods comes in Sec. 13.3 and Sec. 13.4, respectively. Other methods, in particular extrapolation methods, and methods that use higher order derivatives than the the first, in the treatment of 1st order differential systems, are discussed in Sec. 13.5, e.g. Taylor series methods. Special methods for differential systems of special types are discussed in Sec. 13.6, e.g., methods for second order differential systems. Methods for boundary and eigenvalue problems, and for the determination of unknown parameters in differential systems are then presented in Sec. 13.7.

In the other sections, i.e., 1,2,8,9, ideas, concepts and results will be presented, which are relatively independent of the particular numerical methods, although references are made to Euler's method, Runge's 2nd order method and other simple methods for the illumination of the general theory. In Sec. 13.1 we treat existence and uniqueness for initial value problems, error propagation in differential systems, some useful concepts and facts from the qualitative theory of differential equations, and logarithmic norms with applications.

The headlines of Sec. 13.2 are control of step size, local time scale and scale functions, general questions related to implicit methods — with applications to so-called stiff systems and differential algebraic systems, and the construction of stability regions for numerical methods.

Some more advanced topics of general nature are treated in the the last two sections. In Sec. 13.8 more useful ideas from the qualitative theory of differential equations are collected, and there is more theory and applications of logarithmic norms. Finally, Sec. 13.9 is devoted to systems of difference equations, matrix power boundedness, and some other topics like algorithms for stability investigations.

We try to give a logically connected survey of the theory and application of numerical methods for ODEs, but since the space is limited we are happy to be able to refer to two excellent modern monographs for proofs, computational details

[6] There may be exceptional points, or even exceptional differential systems, where the local error is $o(h^{p+1})$.

and also for alternative points of view, and for the treatment of more complicated topics, namely Butcher [4], Hairer, Nørsett and Wanner [20], and Hairer and Wanner [21]. Many other important books are mentioned in the bibliography; a book by Shampine [31] deserves special attention, since it presents a perspective of these topics that results from several decades of research, development and application in scientific laboratories and universities, in a style that makes minimal demands on the mathematical and computational background of the reader.

13.1.2 Existence and Uniqueness for Initial Value Problems

We shall consider initial value problems for the autonomous system

$$\dot{y} = f(y), \qquad y(a) = c, \tag{13.1.6}$$

where $f : \mathbf{R}^s \to \mathbf{R}^s$. In this subsection we shall use a single bar $|\cdot|$ to denote a norm in $\mathbf{R}^s$ or the absolute value of a number, while a double bar $\|\cdot\|$ is used for the max-norm in a Banach space $\mathbf{B}$ of continuous vector valued functions over an interval $I = [a-d, a+d]$, i.e.,

$$\|u\| = \max_{t\in I} |u(t)|.$$

These notations are also used for the corresponding operator norms. Let $D \subseteq \mathbf{R}^s$ be a closed region. We recall, see Sec. 12.2, that f satisfies a Lipschitz condition in D, with the Lipschitz constant L, if

$$|f(y) - f(z)| \le L|y - z|, \quad \forall y, z \in D. \tag{13.1.7}$$

By Lemma 11.2.2, $\max |f'(y)|$, $y \in D$, is a Lipschitz constant, if f is differentiable and D is convex. A point, where a local Lipschitz condition is not satisfied is called a **singular point** of the system (13.1.6).

Theorem 13.1.3.

If f satisfies a Lipschitz condition in the whole of $\mathbf{R}^s$, then the initial value problem (13.1.6) has precisely one solution for each initial vector c. The solution has a continuous first derivative for all t.

If the Lipschitz condition holds in a subset D of $\mathbf{R}^s$ only, then existence and uniqueness hold as long as the orbit stays in D.

Proof. We shall sketch a proof of this fundamental theorem, when $D = \mathbf{R}^s$, based on an iterative construction named after Picard. We define an operator F (usually nonlinear) that maps the Banach space $\mathbf{B}$ into itself:

$$F(y)(t) = c + \int_a^t f(y(x))dx.$$

Note that the equation $y = F(y)$ is equivalent to the initial value problem (13.1.6) on some interval $[a-d, a+d]$, and consider the following iteration in $\mathbf{B}$.

$$y_0 = c, \quad y_{n+1} = F(y_n).$$

For any pair y, z of elements in $\mathbf{B}$, we have,

$$\begin{aligned}\|F(y) - F(z)\| &\le \int_a^{a+d} |f\big(y(t)\big) - f\big(z(t)\big)| \cdot |dt| \\ &\le \int_a^{a+d} L|y(t) - z(t)| \cdot |dt| \le Ld\|y - z\|.\end{aligned}$$

It follows that Ld is a Lipschitz constant of the operator F. If $d < 1/L$, F is a contraction, and it follows from the Contraction Mapping (Theorem 11.2.1) that the equation $y = F(y)$ has a unique solution. For the initial value problem (13.1.6) it follows that there exists precisely one solution, as long as $|t - a| \le d$.

This solution can then be continued to any time by a step by step procedure, for $a + d$ can be chosen as a new starting time and substituted for a in the proof. In this way we extend the solution to $a + 2d$, then to $a + 3d$, etc. and also *backwards* to $a - d,\ a - 2d,\ a - 3d$, etc.. □

Note that this proof is based on two ideas of great importance to numerical analysis: *iteration* and the *step-by-step construction*. (There is an alternative proof that avoids the step-by-step construction, see, e.g., Coddington and Levinson [6, p. 12]). A few points to note are:

A. For the *existence* of a solution, it is *sufficient that f is continuous*, (the existence theorem of Cauchy and Peano, see, e.g., Coddington and Levinson [6, p. 6])). That *continuity is not sufficient for uniqueness* can be seen by the following simple initial value problem,

$$\dot{y} = 2|y|^{1/2}, \qquad y(0) = 0,$$

which has an infinity of solutions for $t > 0$, namely $y(t) = 0$, or, for any non-negative number k,

$$y(t) = \begin{cases} 0, & \text{if } t \le k; \\ (t-k)^2, & \text{otherwise.} \end{cases}$$

The Lipschitz condition is one of the simplest sufficient conditions for uniqueness.

B. The theorem is *extended to non-autonomous systems* by the usual device for making a non-autonomous system autonomous 13.1.2).

C. If the Lipschitz condition holds only in a subset D, then the ideas of the proof can be extended to guarantee existence and uniqueness, *as long as the orbit stays in D*. Let M be an upper bound of $|f(y)|$ in D, and let r be the shortest distance from c to the boundary of D. Since

$$|y(t) - c| = |\int_a^t f\big(y(x)\big)dx| \le M|t - a|,$$

we see that there will be no trouble as long as $|t-a| < r/M$, at least. (This is usually a pessimistic underestimate.) On the other hand, the example

$$\dot{y} = y^2, \qquad y(0) = c > 0,$$

which has the solution $y(t) = c/(1-ct)$, shows that the solution can cease to exist for a finite t (namely for $t = 1/c$), even if $f(y)$ is analytic for all y. Since $f'(y) = 2y$, the Lipschitz condition is guaranteed only as long as $|2y| < L$. In this example, such a condition cannot hold forever, no matter how large L has been chosen.

D. This theory is easily extended to the case of $y \in \mathbf{C}^s$, t is real and $f(t,y)$ is (say) analytic in t, y separately. Nor is the computational practice hard, if complex arithmetic is conveniently available. If not, one can, of course, make a real system of double size for the real and imaginary parts of the variables.

Sometimes one has to integrate a complex analytic differential system, $dw/dz = f(z,w),\ w \in \mathbf{C}^s,\ z \in \mathbf{C}$, along a path in the complex domain. If the path is known in the form $z = \phi(t)$, where t is *real*, we obtain the case just discussed. Each component of the solution becomes an analytic function in some domain around the initial point. Note, however, that the solution $w(z)$ can be a *multivalued* analytic function. If you integrate from $z = a$ to $z = b$ along two different paths, you can obtain obtain two different values of $f(b)$, e.g., if $f(z, w(z))$ has a pole in the region enclosed by the two paths, see Problem P23.

Such a trouble does not occur for a *linear* non-autonomous system, i.e., if $f(z,w) \equiv A(z)w - g(z)$, $w(a) = c$, along a path in a simply connected open region $\mathcal{D}$ where the elements of $A(z)$, $g(z)$ are analytic—poles are allowed outside $\mathcal{D}$ only. Then $w(z)$ is uniquely defined and is analytic in $\mathcal{D}$, see, e.g., Coddington-Levinson, loc.cit. Sec. 3.7.

If one is interested in some end value $w(b),\ b \in \mathbf{R}$ only, it may be possible to find a complex path from a to b, where $f\big(z, w(z)\big)$ is more well-behaved than on the real interval $[a, b]$, so that larger step sizes can be used. We shall not discuss this further (see Problem P23). A similar question for numerical quadrature is mentioned in Volume I, Chapter 5.

E. Isolated *jump discontinuities* in the function f offer no difficulties, if the problem after a discontinuity can be considered as a new initial value problem that satisfies a Lipschitz condition. For example, *in non-autonomous problems of the form*

$$\dot{y} = f(y) + r(t), \quad \text{or} \quad \dot{y} = r_1(t)y + r(t),$$

a Lipschitz condition for f together with integrability conditions for $r(t)$, $r_1(t)$ are sufficient for existence and uniqueness. In this case $\dot{y}(t)$ is discontinuous, only when $r(t)$ or $r_1(t)$ is so, hence $y(t)$ is continuous. [7] There exist also in practical problems, however, more nasty discontinuities, where existence and uniqueness are not obvious; see Problem P19.

[7] The discussion can be extended to the case where $r(t)$ contains an impulse (a Dirac delta function). $y(t)$ then obtains a discontinuity.

F. A point y^* where $f(y^*) = 0$ is called a **critical point** of the autonomous system. (It is usually not a singular point.) If $y(t_1) = y^*$ at some time t_1, the theorem tells that $y(t) = y^*$ is the *unique* solution for all t, forwards as well as backwards. It follows that a solution that does not start at y^* cannot reach y^* exactly in finite time, but it can converge very fast towards y^*.
Note that this does not hold for a non-autonomous system, at a point where $f(t_1, y(t_1)) = 0$, as is shown by the simple example $\dot{y} = t$, $y(0) = 0$, for which $y(t) = \frac{1}{2}t^2 \neq 0$ when $t \neq 0$. *For a non-autonomous system $\dot{y} = f(t, y)$, a critical point is instead defined as a point y^*, such that $f(t, y^*) = 0$, $\forall t \geq a$.* Then it is true that $y(t) = y^*, \forall t \geq a$, if $y(a) = y^*$.

13.1.3 Variational Equations and Error Propagation

We shall discuss the propagation of disturbances (for example numerical errors) in an ODE system. The application to numerical methods comes later. It is a useful model for the error propagation in the application of one step methods, i.e. if y_n is the only input data to the step, where y_{n+1} is computed.

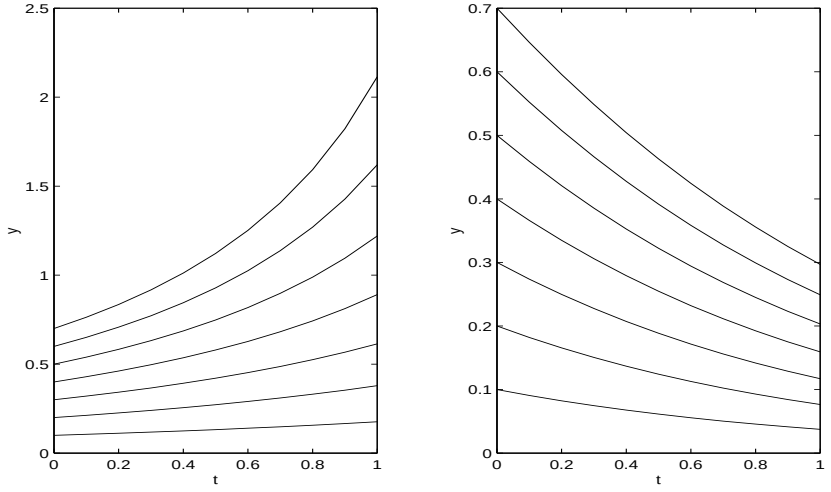

Figure 13.1.2. *Families of solution curves $y(t; c)$ for two different differential equations.*

The solution of the initial-value problem, (13.1.2), can be considered as a function $y(t; c)$, where c is the vector of initial conditions. Here again, one can visualize a *family of solution curves*, this time in the (t, y)-space, one curve for each initial value, $y(a; c) = c$. To begin with, we consider the case of a *single* ODE. The family of solutions can, for example, look like one of the two families of curves in Fig. 13.1.2. The dependence of the solution $y(t; c)$ on c is often of great interest, both for the technical and scientific context it appears in and for the numerical treatment.

A disturbance in the initial condition—e.g., a round-off error in the value of c—means that $y(t)$ is forced to follow "another track" in the family of solutions. Consider, e.g., the numerical treatment by a one-step method. Then there is a small disturbance at each step,—truncation error and/or rounding error—which produces a similar transition to "another track" in the family of solution curves. In Fig. 13.1.3, we give a greatly exaggerated view of what normally happens. The small circles show the computed points (in a fictitious case). One can compare the above process of error propagation to an *interest process*; in each step there is "interest" on previously committed errors. At the same time, a new "error capital" (local error) is put in. In Fig. 13.1.3, the local errors are the jumps in the staircase curve. The "interest rate" can, however, be negative (see Fig. 13.1.2b); an advantage in

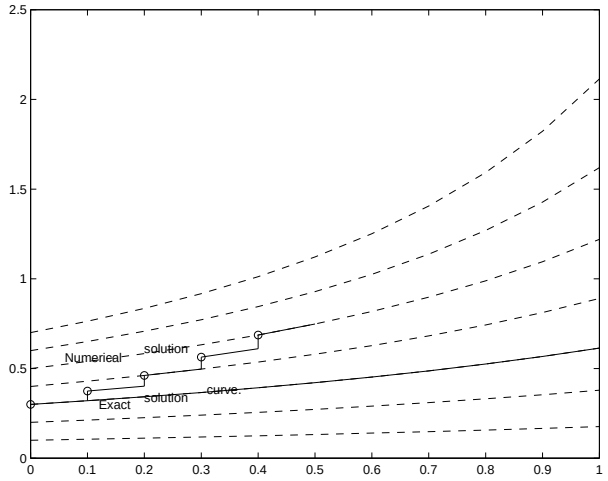

Figure 13.1.3. *An exaggerated picture of the propagation of truncation error. The circles show four steps of the numerical solution.*

this context. If the curves in the family of solutions depart from each other quickly, then the initial value problem is ill-conditioned; otherwise it is well-conditioned. footnote: Some multistep methods can introduce other characteristics in the error propagation mechanism that are *not* inherent in the differential equation itself. *So our discussion in this section are valid, only if the method has adequate stability properties* , for the step size sequence chosen. We shall make this assumption more clear later, e.g., in the beginning of Sec. 13.2. For the two methods mentioned so far, the discussion is relevant (for example) as long as $\|hf'(y)\|^2 \ll 1$.

We can look at the error propagation more quantitatively, *to begin with in the scalar case*. Consider the function $U = \partial y(t;c)/\partial c$. It satisfies a linear differential equation, the **linearized variational equation**

$$\frac{\partial U}{\partial t} = J(t)U, \qquad J(t) = \Big(\frac{\partial f}{\partial y}\Big)_{y=y(t;c)}, \tag{13.1.8}$$

since, under appropriate differentiability conditions, dealt with in any good text on

Calculus,

$$\frac{\partial U}{\partial t} = \frac{\partial}{\partial t}\Big(\frac{\partial y}{\partial c}\Big) = \frac{\partial}{\partial c}\Big(\frac{\partial y}{\partial t}\Big) = \frac{\partial}{\partial c}\big(f(t, y(t;c))\big) = \frac{\partial f}{\partial y}\frac{\partial y}{\partial c}.$$

Note that the variational equation is usually non-autonomous, even if the underlying ODE is autonomous. We can derive many results from the above, since

$$y(t; c+\delta c) - y(t;c) \approx \frac{\partial y}{\partial c}\delta c = U(t;c)\delta c.$$

We rewrite Eqn. (13.1.8) in the form, $\dfrac{\partial \ln|U|}{\partial t} = J(t)$.

Proposition 13.1.4.
Closely lying curves in the family of solutions approach each other, as t increases, if $\partial f/\partial y < 0$ and depart from each other if $\partial f/\partial y > 0$.

$\partial f/\partial y$ corresponds to the "rate of interest" mentioned previously. In the following *we assume that* $\partial f/\partial y < \mu^*$ for all y in some interval D, that contains the range of $y(t;c)$, $(a < t < b)$. Hence $\partial \ln|U|/\partial t \le \mu^*$. The following propositions are obtained by the integration of this inequality.

Proposition 13.1.5.
For $U(t) = \partial y/\partial c$ it holds, even if μ^* is negative, that

$$|U(t)| \le |U(a)|e^{\mu^*(t-a)}, \qquad a \le t \le b.$$

Proposition 13.1.6.
Let $y(t_n)$ be perturbed by a quantity ϵ_n. The effect of this perturbation on $y(t)$, $t > t_n$, will not exceed

$$|\epsilon_n|e^{\mu^*(t-t_n)} \tag{13.1.9}$$

as long as this bound guarantees that the perturbed solution curve remains in D.

Various bounds for the global error can be obtained by adding such local contributions. Assume, e.g., that there is no initial error at $t = t_0$, that the sequence $\{t_n\}$ is increasing, and that the *local error per unit of time* is less than ϵ, i.e., $|\epsilon_n| \le \epsilon(t_{n+1} - t_n)$. Substitute this in (13.1.9) and sum all contributions for $t_{n+1} \le t$, An approximate bound for the global error at the time t is thus obtained:

$$\epsilon \sum_{t_{n+1} \le t} (t_{n+1} - t_n)e^{\mu^*(t-t_n)} \approx \epsilon \int_{t_0}^{t} e^{\mu^*(t-x)}dx,$$

hence

$$|\text{Approximate Global Error}| \le \begin{cases} \epsilon\frac{e^{\mu^*(t-t_0)}-1}{\mu^*}, & \text{if } \mu^* \neq 0; \\ \epsilon(t-t_0), & \text{if } \mu^* = 0. \end{cases} \tag{13.1.10}$$

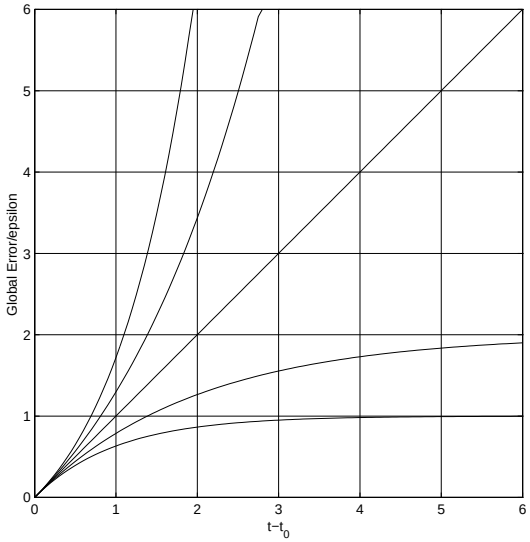

Figure 13.1.4. *The global error bound, divided by ϵ, versus $t - t_0$ for $\mu^* = \pm 1,\ \pm 0.5,\ 0$.*

Note that, if $\mu^* < 0$, the error is bounded by $\epsilon/|\mu^*|$, for all $t > t_0$. The global error bound, divided by ϵ, is shown in Fig. 13.1.4 for $\mu^* = \pm 1,\ \pm 0.5,\ 0$.

We shall see that Fig. 13.1.4 and the inequalities of (13.1.10), with a different interpretation, are typical for the error propagation under much more general and realistic assumptions than those made here. More general versions of the second and the third propositions will be given below.

Now the concept of variational equation will be generalized to **systems** of ODEs. Let $z(t)$ be a function that satisfies the differential system,

$$\dot{z} = f(z) + r(t),$$

where $r(t)$ is a piecewise continuous perturbation. Let $y(t)$ be a solution of the system $\dot{y} = f(y)$. Set

$$u(t) = z(t) - y(t).$$

Then $u = u(t)$ satisfies the differential equation,

$$\dot{u} = f(y(t) + u) - f(y(t)) + r(t), \tag{13.1.11}$$

called the *exact* or the **nonlinear variational equation**. It is sometimes convenient *to allow the perturbation term r to depend on u*, i.e., $r = r(t, u)$.

Lemma 13.1.7.

The nonlinear variational equation (13.1.11) can be written in **pseudo-linear** *form, i.e.*

$$\dot{u} = J(t, u)u + r(t, u), \tag{13.1.12}$$

where $J(t,u)$ is a **neighborhood average** *of the Jacobian matrix,*

$$J(t,u) = \int_0^1 f'\big(y(t)+\theta u\big)\,d\theta; \tag{13.1.13}$$

the dash means differentiation of $f(y)$ with respect to the vector y.

Proof. By the chain rule, $\partial f(y(t)+\theta u)/\partial\theta = f'(y(t)+\theta u)u$, hence

$$f(y(t)+u) - f(y(t)) = \int_0^1 f'(y(t)+\theta u)u\,d\theta = J(t,u)u. \tag{13.1.14}$$

□ For example, $u(t;c) = y(t;c+\delta c) - y(t;c)$ satisfies (13.1.12) exactly, with $r(t,u) = 0$, $u(a;c) = \delta c$.

In §13.1.4 we shall see that strict bounds for the solution of the nonlinear variational equation can be obtained rather conveniently by means of the so-called logarithmic norm technique. In Theorem 13.1.23 we shall obtain a strict generalization of the bounds (13.1.10) to nonlinear systems. In some other respects, however, it can be rather awkward to deal with this pseudo-linear equation where, e.g., the superposition principle does not hold exactly.

Systems of ODEs often contain *parameters*, and it may be of interest to study how the solution $y(t;p)$ depends on a parameter vector p. If we *assume* that the matrix $U(t;p) = \partial y/\partial p$ exists then the following result can be derived formally by the application of the chain rule to (13.1.15). You find a more thorough treatment (with a proof of the validity of this assumption) in the first three chapters of Coddington-Levinson [6]. We may include components of the initial values $y(t_0)$ in the parameter vector.

Theorem 13.1.8.

Let $y = y(t;p)$ be the solution of the initial value problem,

$$dy/dt = f(t,y;p), \qquad y(a) = c(p), \tag{13.1.15}$$

where p is a vector of parameters. Assume that f is a continuous function of t, a differentiable function of y and p everywhere, and assume that $\|\partial f/\partial y\| \le L$ everywhere.

Then, $y(t;p)$ is a differentiable function of t and p. The matrix valued function $U(t;p) = \partial y/\partial p$ is determined by the non-homogeneous **linearized variational equation**,

$$dU/dt - J(t;p)U = \partial f/\partial p, \qquad U(0) = \partial c/\partial p.$$

$J(t;p)$ equals $\partial f/\partial y$ evaluated at $y = y(t;p)$. [8]

If f is an analytic function of $(y;p)$ in some complex neighborhood of $(y_0;p_0)$, then $y(t;p)$ is an analytic function of p that can be expanded into powers of $p-$

[8] The matrix $J(t;p)$ must not be confused with the neighborhood average $J(t,u)$ introduced above.

p_0. *This so-called* **regular perturbation expansion** *converges if* $\|p - p_0\|$ *small enough.*

COMMENTS: The perturbation expansion is called regular, since the same number of initial values $y(a)$ are required for $p = p_0$ as for $p \neq p_0$, see also §3.1.6. This is not the case for *singular* perturbation problems, see Sec. 13.8.

The coefficient vectors in the regular expansion (which are functions of t) can, if p is a scalar, be computed recursively by the solution of *inhomogeneous linear variational equations with the same matrix* $J(t; p_0)$*, but the right hand sides now depend on the coefficients previously computed.* See one of the last problems of this section.

In practice, it may be easier to compute numerical solutions $y(t; p_0 + \epsilon)$ for a few (judiciously selected) values of ϵ. [9] A few coefficients in an approximation of $y(t; p_0 + \epsilon)$ as a polynomial in ϵ can then be computed by a polynomial fitting program. This polynomial is not identical to a truncated perturbation expansion.

We shall now discuss linear systems with variable coefficients in general. We change notation; J is replaced by A. Let $u = u_j(t)$, $j = 1 : s$, be the solution of the differential system

$$\dot{u} = A(t)u, \quad u(a) = e_j. \tag{13.1.16}$$

We can combine these vector differential equations to a *matrix differential equation*

$$\dot{U} = A(t)U, \qquad U(a) = I. \tag{13.1.17}$$

The jth column of the solution $U(t)$ is $u_j(t)$, $j = 1 : s$. The solution of (13.1.16) with a general initial condition at $t = a$ reads

$$u(t; u(a)) = U(t)u(a). \tag{13.1.18}$$

More generally, the solution with a condition at $t = x$ reads

$$u(t) = U(t)(U(x))^{-1}u(x). \tag{13.1.19}$$

$U(t)$ is called a **fundamental matrix** solution for (13.1.16). [10] We summarize and extend this in a theorem.

Theorem 13.1.9.

The solution at time t of a homogeneous linear ODE system with variable coefficients is a linear function of its initial vector. For the system in (13.1.16) this function is represented by the fundamental matrix $U(t)$ defined by (13.1.17).

The solution of the inhomogeneous problem, $\dot{u} = A(t)u + r(t)$, reads,

$$u\big(t; u(a)\big) = U(t)u(a) + \int_a^t U(t)(U(x))^{-1}r(x)dx. \tag{13.1.20}$$

[9] It is important to use *the same step size sequence for all* ϵ, in order to rely on approximate cancellation of truncation errors; see a warning example in Hairer, Nørsett and Wanner [1993, p. 201].

[10] It is nowadays often called *evolution operator*, a terminology that is also applied for differential equations in infinite dimensional spaces and, in nonlinear problems, for the analogous non-linear operator.

For a fixed t, $u(t;u(a))$ is thus an affine function of $u(a)$.

The proof of (13.1.20) is left as an exercise (Problem P16).
If A does not depend on t,

$$U(t) = e^{tA}, \qquad U(t)(U(x))^{-1} = U(t-x). \tag{13.1.21}$$

The matrix exponential is defined in Example 10.2.2. More generally, the fundamental matrix can be expressed in terms of matrix exponentials, if $A(t)$ commutes with its time derivative for all t. *In other cases the fundamental matrix cannot be expressed in terms of matrix exponentials*, and $U(t)(U(x))^{-1} \neq U(t-x)$.

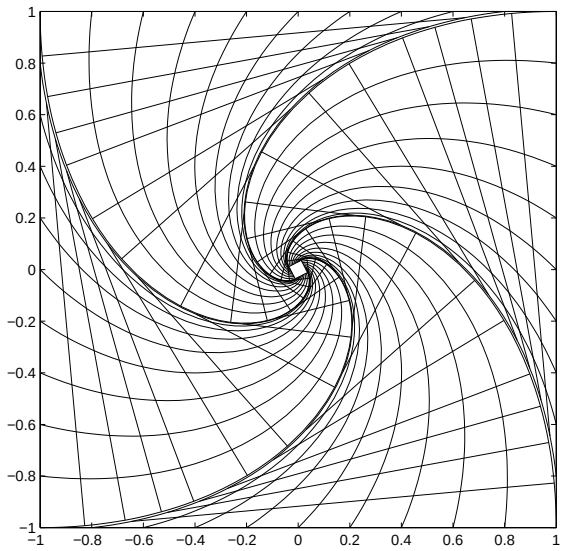

Figure 13.1.5. *Orbits for the linear system (13.1.3) from 40 starting points. See Example 13.1.10 and Problem 7.*

Example 13.1.10 Fig. 13.1.5 shows the orbits from 40 starting points on the boundary of the square with corners at $y_1 = \pm 1$, $y_2 = \pm 1$, for the linear system (13.1.3), i.e. $\dot{y}_1 = -y_1 - y_2$, $\dot{y}_2 = y_1 - y_2$. For some values of t, the points reached at time t are joined. Note that these points are located on a square that is the map at time t of the square boundary that contains the initial points. That all these maps become rotated and diminished squares is due to this special example. Theorem 13.1.9 tells, however, that for any linear system, also with variable coefficients, they would have become parallelograms, at least. See also Problem P7.

Example 13.1.11 Fig. 13.1.6 shows graphical output from simulations of the famous Lotka–Volterra **predator-prey model**, see Braun (1975), by means of a

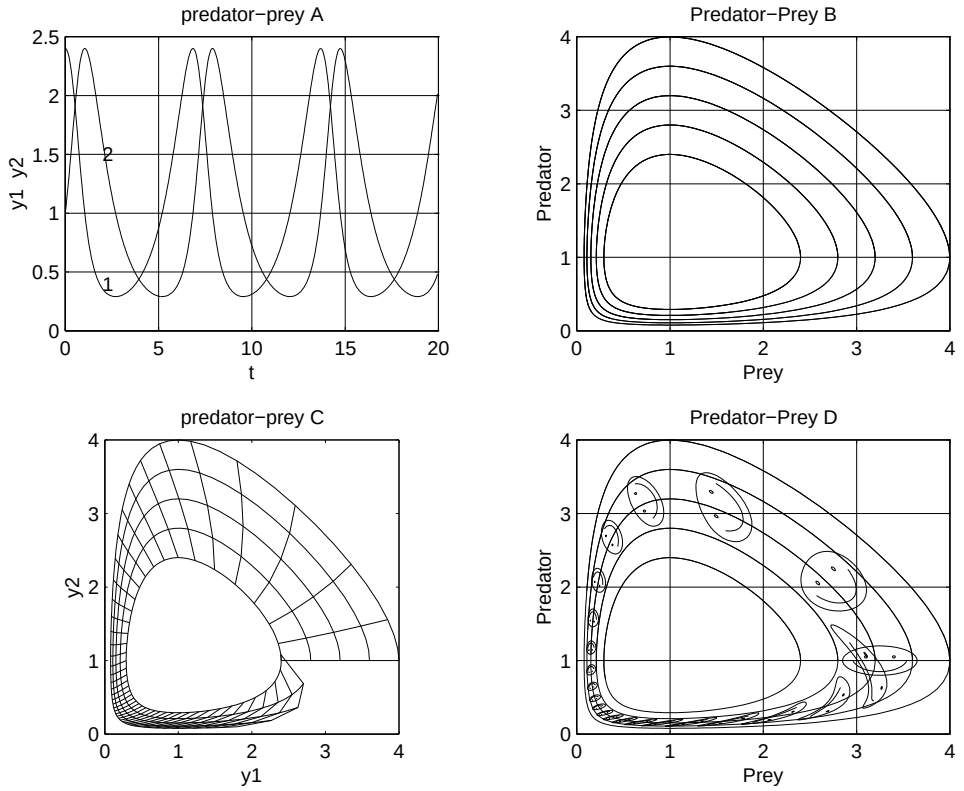

Figure 13.1.6. *Four graphical representations of orbits for the non-linear Predator-Prey Model. Parameter values:* $a = b = c = d = 1$. *See Example 13.1.11 and Exercise C7.*

multistep method with constant step size $h = 0.05$.

$$\dot{y}_1 = ay_1 - by_2y_1, \quad \dot{y}_2 = -cy_2 + dy_1y_2, \quad (a, b, c, d > 0).$$

In this model, the populations of predators and prey are supposed to be large and are approximately described by means of differentiable functions; $y_1(t), y_2(t)$ are approximately the number of prey and predator, respectively. scaled by the division by some large number. The scaled number of prey swallowed during the time interval $[t, t+dt]$ is assumed to be $by_1(t)y_2(t)dt$. The parameter a is the nativity minus the mortality due to other causes than a hungry predator. The parameters c, d have analogous interpretations.

Fig.A shows $y_1(t)$, $y_2(t)$ with the initial condition $y_1(0) = 2.4$, $y_2(0) = 1$. Fig. B shows five orbits, with starting points $y_1 = 2.4 : 0.4 : 4$, $y_2 = 1$. These orbits give experimental (numerical) evidence for the conjecture that the orbits of this problem are closed curves. Each orbit returns to its starting point and, by the uniqueness theorem, it then continues along the same path again and again and

again..., hence $y(t)$ is a periodic function of t, but you see from fig.C or D that the length of the period depends on the starting point. (A hint for a theoretical proof is given in a problem of Section 13.9.)

In fig.C the points which are reached at the same time t are joined, $t = 0 : 0.15 : 6.60$. The mappings $y(0) \mapsto y(0.15n)$, $n = 1 : 44$ are no straight lines here, since the problem is nonlinear. On a "microscopic" scale the mapping is approximately affine: you see that small (approximate) parallelograms are mapped onto small (approximate) parallelograms, in fact by means of the the matrices $U(0.3n, y(0))$, $n = 1 : 22$.

Finally, fig.D illustrates the non-linearity of the mappings on a "macroscopic scale". Smiley initially looks like a honey melon, but after a revolution he is more like a banana; and look what has happened to his smile. For the production of this figure 160 copies of the 2×2 system were run simultaneously, with different initial values — this took on a PC less than 10 seconds, including the numerical and graphical output to file and screen.

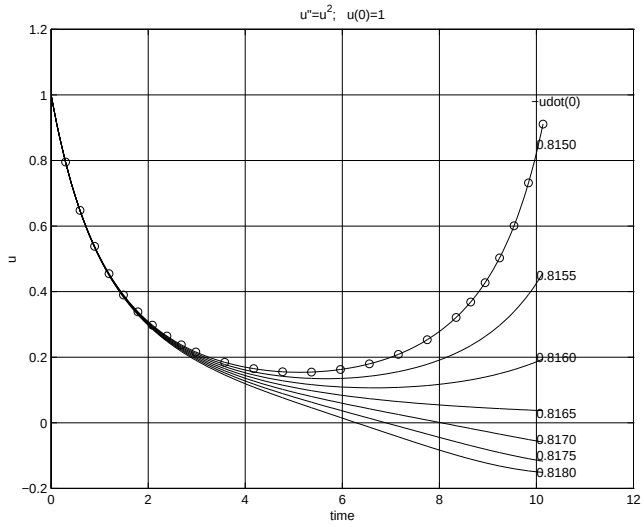

Figure 13.1.7. $\ddot{u} = u^2$, $u(0) = 1$. *Small changes of* $\dot{u}(0)$ *make large changes in the solution.*

Example 13.1.12 Fig. 13.1.7 shows $u(t)$ versus t for the problem $\ddot{u} = u^2$, $u(0) = 1$, for seven close values of $\dot{u}(0)$. This second order equation is written as a first order system, where

$$y \equiv \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} u \\ \dot{u} \end{pmatrix}, \qquad \dot{y} = \begin{pmatrix} y_2 \\ y_1^2 \end{pmatrix} \equiv f(y), \qquad f'(y) = \begin{pmatrix} 0 & 1 \\ 2y_1 & 0 \end{pmatrix}.$$

The eigenvalues of $f'(y)$ are $\pm\sqrt{2y_1}$. Although they do not tell the whole truth when $f'(y(t))$ is not constant, they usually give some indication about the local rate of

growth of a disturbance; note the different behaviour for positive and negative values of $y_1 = u$.

You easily find that $u(t) = 6(t + \sqrt{6})^{-2}$ satisfies the differential equation with initial conditions $u(0) = 1$, $\dot{u}(0) = -2/\sqrt{6} = -0.816497$. Note that $u(t) \to 0$ as $t \to \infty$. It can be shown that $u(t) \to +\infty$ for some finite value of t, for all other solutions of this differential equation; also for those solutions, which are negative in some interval.

These figures were produced by means of a fifth order accurate multistep method. The small circles are delimiters of arcs consisting of five consecutive steps. More questions about this example are asked in exercise C13 b.

13.1.4 Some Elementary Results from the Qualitative Theory of ODEs

The topic of the qualitative theory of differential equations is how to draw conclusions about some essential features of the motions of a system of ODEs, even if the motions cannot be expressed explicitly in analytic form. In a way it seems to be the opposite to the study of ODEs by numerical methods. It is more adequate, however, to consider the qualitative theory as a complement to numerical methods.

The ideas and results from this theory can be very useful in many ways, for example:

- for the planning of numerical experiments,
- for an intelligent interpretation of the results of a simulation,
- for testing a program, in particular for finding out, whether an unexpected result of a simulation is reasonable, or due to a bug in the program, or due to the use of too large time steps, or some other cause.

On the other hand, simulation on a computer is a useful tool also for researchers, whose purpose is to study qualitative features. The reader must find his own switch between computational and analytical techniques, but some ideas from the qualitative theory of ODEs are useful in the bag of tricks.

All ODE systems in this subsection are assumed to satisfy a Lipschitz condition etc., so that there is no trouble about existence and uniqueness. We begin by a simple and useful example.

Example 13.1.13 Consider a *single* autonomous ODE, $\dot{y} = f(y)$, where the graph of $f(y)$ is shown in the left part of Fig. 13.1.8. The equation has three critical points, $y_1 < y_2 < y_3$, i.e. points where $f(y) = 0$. Since $y(t)$ increases if $f(y) > 0$ and decreases if $f(y) < 0$, we see from the arrows of the figure, that $y(t) \to y_1$ if $y(0) < y_2$, and $y(t) \to y_3$ if $y(0) > y_2$, as $t \to \infty$. See the right part of the figure. With an intuitively understandable terminology (that is consistent with the formal definitions given below), we may say that the critical points y_1, y_3 are **stable** (or **attracting**), while the critical point y_2 is **unstable** (or **repelling**).

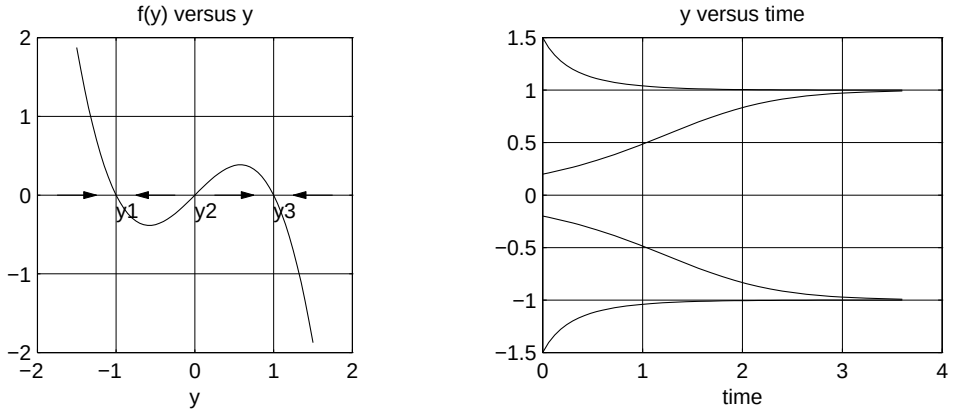

Figure 13.1.8. *Graphs of $f(y)$ and $y(t)$ for Ex. 13.1.13.*

This discussion can be applied to any single autonomous ODE. Notice that a critical point p is stable if $f'(p) < 0$, and unstable if $f'(p) > 0$.

By Taylor's formula, $f(y) \approx f'(p)(y-p)$. If $f'(p) \neq 0$, it is seen that a motion that starts near p is, at the beginning, approximated by $y(t) \approx p + (y_0 - p)\exp f'(p)t$. In the case of repulsion, the neglected terms of this Taylor expansion will play a bigger role, as time goes by.

Now we shall consider a general autonomous system.

Theorem 13.1.14. A BASIC THEOREM IN THE QUALITATIVE THEORY OF ODE'S.

Let $\mathcal{V} \subset \mathbf{R}^s$ be a closed set with a piecewise smooth boundary. A normal pointing into $\mathcal{V}$ is then defined by a vector-valued, piecewise smooth function $n(y)$, $y \in \partial\mathcal{V}$.

Assume that there exists a function $n_1(y)$ that satisfies a Lipschitz condition for $y \in \mathbf{R}^s$, such that

(a) $\|n_1(y)\| \leq K$ *for* $y \in \mathbf{R}^s$,

(b) $n(y)^T n_1(y) \geq c > 0$ *for* $y \in \partial\mathcal{V}$.

Consider an autonomous system $\dot{y} = f(y)$, and assume that

$$n(y)^T f(y) \geq 0, \quad \forall y \in \partial\mathcal{V}, \tag{13.1.22}$$

and that $y(a) \in \mathcal{V}$. Then the motion stays in $\mathcal{V}$ for all $t > a$.

COMMENTS:

- $\mathcal{V}$ is, for example, allowed to be a polyhedron or an unbounded closed set.
- $n_1(y)$ is to be thought of as a smooth function defined in the whole of $\mathbf{R}^s$. On $\partial\mathcal{V}$ it should be a smooth approximant to $n(y)$.

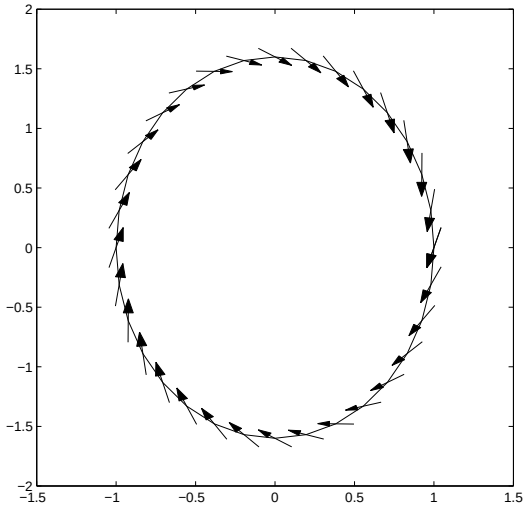

Figure 13.1.9. *Illustration to Theorem 13.1.14. The arrows show the velocity vectors for an autonomous system. A motion that starts in the interior of the oval curve remains inside it all the time.*

Proof. (Sketch.) Consider Fig. 13.1.9. The statement is almost trivial, if the inequality in (13.1.22) is strict. To begin with, we therefore consider a modified problem, $\dot{y} = f(y) + pn_1(y),\ p > 0$, with the solution $y(t;p)$. Then $n(y)^T\dot{y} \geq n(y)^T pn_1(y) \geq pc > 0,\ y \in \partial\mathcal{V}$.

In other words: at every boundary point, the velocity vector for the modified problem points into *the interior of* $\mathcal{V}$. Therefore, an orbit of the modified problem that starts in $\mathcal{V}$ can never escape out of $\mathcal{V}$, i.e., $y(t;p) \in \mathcal{V}$ for $t > a, p > 0$. By Theorem 13.1.8, $y(t;p) \to y(t)$, as $p \to 0$. Since $\mathcal{V}$ is closed, this proves the statement. □

We shall now formulate two useful corollaries of this result.

Theorem 13.1.15 (Comparison Theorem**).**
Let $y(t)$ be the solution of a single non-autonomous equation,

$$\dot{y} = f(t,y), \qquad y(a) = c. \tag{13.1.23}$$

If a function $z(t)$ satisfies the two inequalities, $\dot{z}(t) \leq f(t,z(t))$, $(\forall t \geq a)$, and $z(a) \leq y(a)$, then $z(t) \leq y(t)\ \forall\ t \geq a$.

Proof. You can either convince yourself by a glance at Fig. 13.1.10, or deduce it from the previous theorem, after rewriting (13.1.23) as an autonomous system, and define $\mathcal{V} = \{(t,y) : y \geq z(t)\}$. □

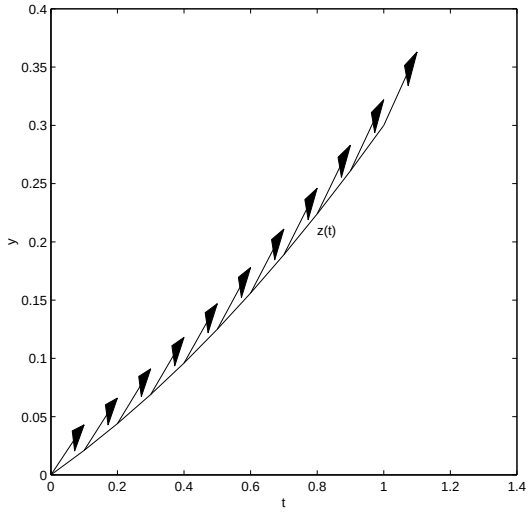

Figure 13.1.10. *The arrows show the slopes of the solution curve of a single ODE, $\dot{y} = f(t, y)$. when it crosses the curve $y = z(t)$. A solution curve that starts above this curve, remains there in the whole interval where the arrows make a positive angle with the tangent of the curve $y = z(t)$.*

There are *variants of this result* with reversed inequalities, which can easily be reduced to the case treated. Another variant: *if strict inequality holds in at least one of the two assumptions concerning $z(t)$, then strict inequality holds in the conclusion, i.e., $z(t) < y(t),\ \forall t > a$.*

Theorem 13.1.16 (POSITIVITY THEOREM**).**

Consider an autonomous system, and assume that for $i = 1, 2, \ldots, s$,
(a) *$y_i(0) \geq 0$;* (b) *$f_i(y) \geq 0$, whenever $y_i = 0$, and $y_j \geq 0$ if $j \neq i$.*
Then $y_i(t) \geq 0$ for all $t > a$.

Another variant: If (a) *is replaced by the condition $y_i(0) > 0$, and* (b) *is unchanged, then $y_i(t) > 0,\ \forall t > a$, (but $y_i(t)$ may tend to zero, as $t \to \infty$).*

Proof. Hint: Choose $\mathcal{V} = \{y : y_i \geq 0, \quad i = 1, 2, \ldots, s\}$. □

In many applications, the components of y correspond to physical quantities known to be non-negative in nature, e.g. mass densities or chemical concentrations. A well designed mathematical model should preserve this natural non-negativeness, but since modeling usually contains idealizations and approximations, it is not self-evident that the objects of a mathematical model possess all the important properties of the natural objects. The positivity theorem can sometimes be used to show that it is the case.

It is important to realize that a numerical method can violate such natural

requirements, for example if the step size control is inadequate, see Example **??**.

Another branch of the qualitative theory of ODEs, is concerned with the **stability of critical points**, not to be confused with the stability of numerical methods. Let p be a critical point of the non-autonomous system, $\dot{y} = f(t, y)$, i.e., $f(t, p) = 0,\ \forall t \geq c$.

Definition 13.1.17.

A critical point p is **stable***, in the sense of Lyapunov,* [11] *if for any given $\epsilon > 0$ there exists a $\delta > 0$, such that, for all $a \geq c$, if $\|y(a) - p\| < \delta$ then $\|y(t) - p\| < \epsilon,\ \forall t > a$. The critical point p is* **asymptotically stable***, if it is stable and $\lim_{t\to\infty} y(t) = p$.*

For the *linear homogeneous system* $\dot{y} = A(t)y$ it follows that the stability of the origin is the same as the boundedness of all solutions, as $t \to \infty$. If A is constant, this means that $\|e^{At}\| \leq C,\ \forall t \geq 0$.

Theorem 13.1.18.

Let A be a constant square matrix. The origin is a stable critical point of the system $\dot{y} = Ay$, if and only if the eigenvalues of A satisfy the following conditions:
(i) *The real parts are less than or equal to zero.*
(ii) *There are no defective eigenvalues on the imaginary axis.*

The stability is asymptotic if and only if all eigenvalues of A have strictly negative real parts.

Proof. Hint: Express e^{At} in terms of the Jordan canonical form of A, see §10.2.4. □

This theorem is not generally valid for linear systems with variable coefficients. You will find a case where the equation $\dot{y} = A(t)y$ has unbounded solutions, though $\Re\lambda(A(t)) \leq -1$ (say) for all t, among the problems of Sec, 13.9.

Another important fact is that *stability and boundedness are not equivalent for nonlinear problems*. We saw in Example 13.1.13 that a solution that started a little above the unstable critical point k_2 became bounded by k_3.

If some, but not all, eigenvalues of A have negative real parts then $y(t) = e^{At}y_0 \to 0$ as $t \to \infty$ iff y_0 belongs to the subspace spanned by the eigenvectors belonging to these eigenvalues. One then talks about *conditional asymptotic stability*. In numerical applications this notion can rarely be applied, and as a rule such conditionally stable cases are to be treated like unstable cases, since truncation and rounding errors will usually kick the solution out from this subspace. There are, however, exceptions. See, e.g., Exercise C13

It is different for another kind of conditional asymptotic stability that is exemplified by the differential equation $\dot{y} = -y^2$. Here $y(t) \to 0$ for any positive

[11] Aleksandr Mikhailovich Lyapunov, Russian mathematician 1857-1918, who gave fundamental contributions to stability theory and probability. Our stability concept is in some more advanced texts called *uniform* stability. In such texts, δ is allowed to depend on a for non-autonomous systems.

value of $y(0)$, while $y(t) \to -\infty$ for any negative value of $y(0)$. A more complicated situation occurs in systems that describe chemical reactions. The right hand sides are often quadratic functions of y, and the positivity theorem can be used to prove that $y_i(t) > 0$, $\forall i$, $\forall t > 0$, if $y_i(0) \geq 0$, $\forall i$. Nevertheless, in a *numerical* solution a component of y can become negative, due to truncation and rounding errors, and if nothing is done about it, the numerical solution may diverge violently. In this case something can be done. Some care is, however, needed in the decision when a variable can be set equal to zero, but it is beyond the scope of this text to go into details.

In the neighborhood of a critical point p a solution $y(t)$ of the system $\dot{y} = f(y)$ is, during a finite time interval, close to a solution $y = y_L(t)$ of the linearized variational equation

$$\dot{y} = f'(p)(y - p).$$

This is a system with constant coefficients. If all eigenvalues of $f'(p)$ have negative real parts, p is asymptotically stable; both $y(t)$ *and* $y_L(t)$ *converge towards* p *at the same exponential rate. The largest real parts of the eigenvalues of* $f'(p)$ *determine the type and the rate of convergence,* e.g., essentially monotonic if the relevant eigenvalues are real, spiraling if the relevant eigenvalues is a conjugate pair of complex numbers.

If the largest real parts are positive, $y(t)$ and $y_L(t)$ move away from p, monotonically or spiraling at the same exponential rate, real or complex. In the case between, i.e. if the largest real part is zero, it can happen that $y(t)$ like $y_L(t)$ remains in the neighborhood of p, e.g., in a 2-dimensional case the orbit can be a closed curve, and both motions are periodic, with approximately the same frequency.

We observe this in the predator-prey example. Here $p = [1; 1]$, the eigenvalues of $f'(p)$ are $\pm i$, and the period is thus approximately 2π; a theoretical proof that all the orbits of this example are indeed closed is found in Sec. 13.8.

But it is not always so; even if the relevant eigenvalues are a conjugate imaginary pair, it can happen that $y(t)$ (unlike $y_L(t)$) can spiral in towards p or spiral away from p, see Theorem 13.1.29 and the problems of Sec. 13.8.

13.1.5 The Logarithmic Norm, Properties and Applications

We shall now develop tools that, among other things, make the generalisation of Propositions 13.1.5 and 13.1.6 to systems of ODEs.

Definition 13.1.19.

Let $\|\cdot\|$ *denote some vector norm and its subordinate matrix norm. Then the subordinate* **logarithmic norm** *of the matrix* A *is given by*

$$\mu(A) = \lim_{\epsilon \to +0} \frac{\|I + \epsilon A\| - 1}{\epsilon} \tag{13.1.24}$$

This limit exists, for any choice of vector norm, since one can show, by the triangle inequality, that, as $\epsilon \to +0$, the right hand side decreases monotonically and is bounded below by $-\|A\|$.

First note that, if $a \in \mathbf{C}$ then,

$$\lim_{\epsilon \to +0} \frac{|1+\epsilon a| - 1}{\epsilon} = \Re a. \tag{13.1.25}$$

Just like the ordinary norm of a matrix is a generalization of the modulus of of a complex number, the logarithmic norm corresponds to the real part. The logarithmic norm is a real number, and $\mu(A) \le \|A\|$. *It can even be negative*, which is very favorable for the estimates and bounds that we are interested in, where a bound for the logarithmic norm, multiplied by t, typically appears in an exponent.

Many of the notations and results below are analogous to familiar things for ordinary vector and matrix norms, e.g. we denote by $\mu_p(A)$ the logarithmic norm subordinate to l_p norm.

Theorem 13.1.20. *The logarithmic norm subordinate to the* max-norm *reads,*

$$\mu_\infty(A) = \max_i \big(\Re(a_{ii}) + \sum_{j, j\neq i} |a_{ij}|\big).$$

More generally, the logarithmic norm subordinate to the weighted max-norm, $\|x\|_w = \max_i |x|_i/w_i$, *reads*

$$\mu_w(A) = \max_i \big(\Re(a_{ii}) + \sum_{j, j\neq i} |a_{ij} w_j/w_i|\big).$$

If all diagonal elements are real and larger than $-1/\epsilon$, then

$$\|I + \epsilon A\|_w = 1 + \epsilon \mu_w(A). \tag{13.1.26}$$

Similarly, the logarithmic norm subordinate to the l_1-norm *reads,*

$$\mu_1(A) = \max_j \big(\Re(a_{jj}) + \sum_{i, i\neq j} |a_{ij}|\big) = \mu_\infty(A^H).$$

Set $B = \frac{1}{2}(A + A^H)$, and let $\lambda_i(B)$ be an eigenvalue of B. Then the logarithmic norm subordinate to the l_2-norm *reads,*

$$\mu_2(A) = \max_i \Re\lambda_i(B) \le \mu(B) \le \tfrac{1}{2}(\mu(A) + \mu(A^H)). \tag{13.1.27}$$

Here $\mu(\cdot)$ denotes the logarithmic norm subordinate to any *norm, e.g., the max-norm or some weighted max-norm. (The last inequalities are of practical importance, since the exact formula for $\mu_2(A)$ may require much computation.)*

Proof. Set $s_i = \sum_{j, j\neq i} |a_{ij}|$. By (6.2.16), $\|I+\epsilon A\|_\infty = \max_i(|1+\epsilon a_{ii}| + \epsilon s_i)$, hence

$$\frac{\|I + \epsilon A\|_\infty - 1}{\epsilon} = \max_i \Big(\frac{|1+\epsilon a_{ii}| - 1}{\epsilon} + s_i\Big) \to \max_i(\Re a_{ii} + s_i).$$

Moreover, if $\epsilon a_{ii} \ge -1 \quad \forall i$, then $\|I + \epsilon A\|_\infty = \max_i(1 + \epsilon a_{ii} + \epsilon s_i) = 1 + \epsilon \mu_\infty(A)$.

The derivations of the formulas concerning the weighted max-norm and the l_1-norm are left for Problem 9b.

Short proofs of the formulas for $\mu_2(A)$ require ideas that are developed later. [12] Here we only note the analog to

$$\begin{aligned}\|I+\epsilon A\|_2^2 &= \max_i |\lambda_i((I+\epsilon A)(I+\epsilon A^H))| = \max_i |1+\epsilon\lambda_i(A+A^H)+O(\epsilon^2)| \\ &= \max_i |1+2\epsilon\lambda_i(B)+O(\epsilon^2)|.\end{aligned}$$

The inequality $\Re(\lambda_i(B) \le \mu(B)$ follows from statement B in Theorem13.1.25. Finally, $\mu(B) \le \frac{1}{2}(\mu(A)+\mu(A^H))$ follows from the important subadditivity property, i.e. property B in the next theorem. □

Theorem 13.1.21.

The logarithmic norm has the following properties:

A. $-\|A\| \le \mu(A) \le \|A\|$.

B. $\mu(\alpha A+\beta B) \le \alpha\mu(A)+\beta\mu(B),\ if\ \alpha \ge 0,\ \beta \ge 0$, **subadditivity**.

C. $\mu(\alpha A+\gamma I) = \alpha\mu(A)+\Re\gamma, \quad if\ \alpha \ge 0,\ \gamma \in \mathbf{C}$.

Proof. Property A follows from the application of the triangle inequality to the definition of $\mu(A)$. We next note that, for $\alpha \ge 0$,

$$\mu(\alpha A) = \lim_{\epsilon\to+0}\frac{\|I+\epsilon\alpha A\|-1}{\epsilon} = \alpha\lim_{\epsilon\to+0}\frac{\|I+(\epsilon\alpha)A\|-1}{(\epsilon\alpha)} = \alpha\mu(A).$$

We can therefore, without loss of generality, put $\alpha = \beta = 1$ in the rest of the proof. By the triangle inequality,

$$\left\|I+\frac{\epsilon}{2}(A+B)\right\|-1 = \left\|\frac{I+\epsilon A}{2}+\frac{I+\epsilon B}{2}\right\|-1 \le \frac{\|I+\epsilon A\|-1}{2}+\frac{\|I+\epsilon B\|-1}{2}.$$

Divide the first and the last expression by $\frac{1}{2}\epsilon$, and let $\epsilon \to +0$; property B follows (for $\alpha = \beta = 1$).

In order to prove property C, we consider the identity,

$$\|(1+\epsilon\gamma)(I+\epsilon A)\|-1 = |1+\epsilon\gamma|(\|I+\epsilon A)\|-1)+(|1+\epsilon\gamma|-1).$$

After division by ϵ and passage to the limit, the right hand side becomes $\mu(A)+\Re\gamma$. The left hand side can be written,

$$\|I+\epsilon\gamma I+\epsilon A+O(\epsilon^2)\|-1 = \|I+\epsilon(\gamma I+A)\|-1+O(\epsilon^2),$$

[12] In fact, Eq.(13.1.27) is a particular case of Theorem 13.8.2, where a general inner-product norm is considered.

where the triangle inequality was used in the last step. After division by ϵ and passage to the limit, this becomes $\mu(\gamma I + A)$. □

REMARK: In general, $\mu(-A) \neq -\mu(A)$. Actually, $\mu(-A) \geq -\mu(A)$, since by the subadditivity, $\mu(A) + \mu(-A) \geq \mu(A - A) = 0$. By induction, the *subadditivity can be extended to any number of terms* and, by a passage to the limit, *also to infinite sums and integrals*. In particular we have, for the neighborhood average $J(t, u)$ defined by (13.1.13),

$$\mu(J(t,u)) = \mu\left(\int_0^1 f'(y+\theta u)d\theta\right) \leq \int_0^1 \mu(f'(y+\theta u)d\theta) \leq \max \mu(f'(z)), \tag{13.1.28}$$

where the domain of z must include the line segment between y and $y + u$.

The most important applications of the logarithmic norm are given in the next two theorems. Recall the concept of a "pseudo-linear" system introduced in Lemma 13.1.7.

Theorem 13.1.22. *The solutions of a "pseudo-linear" system,*

$$\dot{u} = J(t,u)u + r(t,u), \tag{13.1.29}$$

satisfy the inequality, [13]

$$\|u\|' \leq \mu(J(t,u))\|u\| + \|r(t,u)\|. \tag{13.1.30}$$

Let $D_t \subset \mathbf{R}^s$ be the ball $\{w : \|w\| \leq \rho(t)\}$ where $\rho(t)$ varies continuously with t. Assume that $\|u(a)\| < \rho(a)$ and that

$$\mu(J(t,w)) \leq \mu^*(t), \qquad \|r(t,w)\| \leq \epsilon(t), \quad \forall w \in D_t, \tag{13.1.31}$$

where $\mu^*(t)$, $\epsilon(t)$ are piecewise differentiable functions.

Then, $\|u(t)\| \leq \psi(t)$, where $\psi(t)$ is a solution of a single differential equation,

$$\dot{\psi} = \mu^*(t)\psi + \epsilon(t), \qquad \psi(a) = \|u(a)\|, \tag{13.1.32}$$

as long as a bound that can be derived from this differential equation guarantees that $\psi(t) < \rho(t)$.

If μ^*, ϵ are chosen to be independent of t, and if $u(a) = 0$, this leads exactly to the bounds (13.1.10), and the behaviour of $\psi(t)$ is illustrated by Fig. 13.1.4. Concerning bounds for more general situations, see (13.1.20) and problem P10.

Proof. By Taylor's theorem,

$$u(t+h) = u(t) + hJ(t,u)u(t) + hr(t,u) + o(h), \qquad (h > 0),$$

[13] It can happen that $\|u\|'$ is discontinuous, e.g., if the max-norm is used. We shall always refer to the derivative in the positive direction. Usually the inequalities are (a fortiori) valid for the derivative in the negative direction too.

$$\|u(t+h)\| \le \|(I+hJ)u(t)\| + h\|r\| + o(h) \le \|I+hJ\| \cdot \|u(t)\| + h\|r\| + o(h).$$

Subtract $\|u(t)\|$ from the first and the last side, and divide by h.

$$\frac{\|u(t+h)\| - \|u(t)\|}{h} \le \frac{\|I+hJ\| - 1}{h}\|u(t)\| + \|r\| + o(1).$$

As $h \to +0$, the left hand side tends to the right-hand derivative of $\|u(t)\|$, and we obtain the result, $\|u(t)\|' \le \mu(J(t,u))\|u\| + \|r(t,u)\|$, where the last inequality holds as long as $u(t) \in D_t$.

Then, by the Comparison Theorem $\|u(t)\| \le \psi(t)$, where $\psi(t)$ is the solution of (13.1.32), *as long as* the bound derived from this guarantees that $\psi(t) < \rho(t)$ (hence $u(t) \in D_t$).

For example, if μ^* and ϵ do not depend on t,

$$\psi(t) = \begin{cases} \psi(a)e^{\mu^*(t-a)} + \epsilon\frac{e^{\mu^*(t-a)}-1}{\mu^*}, & \text{if } \mu^* \neq 0;. \\ \psi(a) + \epsilon(t-a), & \text{if } \mu^* = 0, \end{cases} \tag{13.1.33}$$

i.e., if $u(a) = 0$ the same bounds as in (13.1.10). Bounds valid when μ^*, ϵ depend on t, can be obtained from (13.1.20), the scalar case. □ Eq. (13.1.29) with the

continuous perturbation term $r(t,u)$ thus yields the same bounds as the discrete perturbation model that led to (13.1.10), previously developed for the scalar case only.

Theorem 13.1.23. *Let $z : \mathbf{R} \to \mathbf{R}^s$ be a known differentiable function that satisfies the differential inequality,*

$$\|\dot{z}(t) - f(z(t))\| \le \epsilon(t), \qquad a \le t \le b, \tag{13.1.34}$$

for some piecewise differentiable function $\epsilon(t)$, and let $y(t)$ be a solution of the differential system,

$$\dot{y} - f(y) = 0, \qquad a \le t \le b.$$

Let $\rho(t)$ be a continuous function, and consider a family of balls in $\mathbf{R}^s$, $D_t = \{w : \|w - z(t)\| < \rho(t)\}$, and assume that $\|y(a) - z(a)\| < \rho(a)$. Also assume that, for every $t \in [a,b]$, there exists a real-valued piecewise differentiable function $\mu^*(t)$, such that

$$\mu(f'(y)) \le \mu^*(t), \qquad \forall y \in D_t.$$

Then $\|z(t) - y(t)\| \le \psi(t)$, where $\psi(t)$ is a solution of the scalar differential equation,

$$\dot{\psi} = \mu^*(t)\psi + \epsilon(t), \qquad \psi(a) = \|z(a) - y(a)\|,$$

as long as $\psi(t) < \rho(t)$, i.e. as long as a bound obtained from this, e.g. (13.1.33), or a bound derived from (13.1.20), guarantees that $y(t) \in D_t$. The behaviour of $\psi(t)$ is illustrated by Fig.13.1.4.

Comment: The union of the sets $(t, D_t) \subseteq \mathbf{R} \times \mathbf{R}^s$ is to be thought of as a hose or "a French horn" enclosing the path in $\mathbf{R} \times \mathbf{R}^s$ defined by the points $(t, z(t))$.

Proof. Set $u(t) = z(t) - y(t)$. Note that we can write $\dot{u}(t) = f(y(t) + u(t)) - f(y(t)) + r(t)$, where $\|r(t)\| \leq \epsilon(t)$. By Lemma 13.1.7,

$$f(y(t) + u(t)) - f(y(t)) = \int_0^1 f'(y(t) + \theta u(t))\, d\theta \; u(t) = J(t, u(t))u(t).$$

By (13.1.28), $\mu(J(t, u(t)) \leq \max_y \mu(f'(y)) \leq \mu^*(t), \quad y \in D_t$. Hence, $u(t)$ satisfies a pseudo-linear system of the form, $\dot{u} = J(t, u)u + r(t)$, where $\|r(t)\| \leq \epsilon(t)$. The result then follows from the previous theorem. □ Theorem 13.1.23 is, for the sake

of simplicity, formulated for an autonomous system. It is, mutatis mutandis, *valid also for a non-autonomous system* $\dot{y} = f(t, y)$. Theorem 13.2.1 is an *important generalization of the last two theorems*, adapted to a continuous model for the error propagation in the numerical treatment of initial value problems. An important corollary about the *convergence towards a critical point* is left for Problem P14. Other generalizations are made in Section 13.8.

Example 13.1.24

Consider the pendulum equation and the linearized pendulum equation, with the same initial values,

$$\ddot{\zeta} = -\sin\zeta; \qquad \ddot{\eta} = -\eta;$$
$$\zeta(0) = \eta(0) = \eta_0 > 0; \qquad \dot{\zeta}(0) = \dot{\eta}(0) = 0.$$

We shall find a bound for $|\zeta(t) - \eta(t)|$. Set

$$y \equiv \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} \eta \\ \dot{\eta} \end{pmatrix}, \qquad z \equiv \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} \zeta \\ \dot{\zeta} \end{pmatrix},$$

and rewrite the differential equations as systems of the first order.

$$\dot{y} = \begin{pmatrix} y_2 \\ -y_1 \end{pmatrix} \equiv f(y), \qquad f'(y) = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

$$\dot{z} = \begin{pmatrix} z_2 \\ -\sin z_1 \end{pmatrix} = \begin{pmatrix} z_2 \\ -z_1 \end{pmatrix} + e(t), \quad e(t) = \begin{pmatrix} 0 \\ z_1(t) - \sin z_1(t) \end{pmatrix}.$$

It can be shown that $\|e(t)\|_2 \leq \eta_0^3/6 \equiv \epsilon(t)$, (problem P15).

Set $B = \frac{1}{2}\big(f'(y) + f'(y)^T\big) = 0, \ \forall y$. By (13.1.27), $\mu_2(f'(y)) \leq \mu(B) = 0$. Then, by Theorem 13.1.23, $\|z(t) - y(t)\|_2 \leq \psi(t)$, where $\psi(t)$ is a solution of the scalar differential equation, $\dot{\psi} = \mu^*(t)\psi + \epsilon(t) \equiv 0 + \eta_0^3/6$, $\psi(0) = 0$. Hence $\|z(t) - y(t)\|_2 \leq t\eta_0^3/6$. This implies

$$|\zeta(t) - \eta(t)| \leq t\eta_0^3/6, \qquad |\dot{\zeta}(t) - \dot{\eta}(t)| \leq t\eta_0^3/6.$$

A sharper bound can be obtained for small values of t (actually for $t < 2$):

$$|\zeta(t) - \eta(t)| \leq \int_0^t |\dot{\zeta}(\tau) - \dot{\eta}(\tau)|\, d\tau \leq \frac{t^2\eta_0^3}{12}, \quad \text{hence} \quad |\zeta(t) - \eta(t)| \leq \min\left(\frac{t}{6}, \frac{t^2}{12}\right)\eta_0^3.$$

The bound is good only if (say) $\eta_0^2 t \ll 12$; one can show that both $|\eta(t)|$ and $|\zeta(t)|$ are bounded by η_0. An experimental study of the sharpness of this bound is left for exercise C5.

Theorem 13.1.25.

A. $\|e^{At}\| \le e^{\mu(A)t},\ if\ t \ge 0.$

B. *The real part of an eigenvalue of A cannot exceed $\mu(A)$.*

C. $\|Au\| \ge |\mu(A)|\ \|u\|,\ if\ \mu(A) < 0.$

D. $\|A^{-1}\| \le |\mu(A)|^{-1},\ if\ \mu(A) < 0.$

Proof. The system $\dot{u} = Au$ has the solution $u(t) = e^{At}u(0)$. Then, by the simplest particular case of Theorem 13.1.22 (with $J = A,\ r = 0, t \ge 0$), $\|e^{At}u(0)\| = \|u(t)\| \le e^{\mu(A)t}\|u(0)\|$. Since this is true for every vector $u(0)$, statement A follows.

In order to prove statement B, note that if $Av = \lambda v$ then $\|e^{A}v\| = \|e^{\lambda}v\| = e^{\Re\lambda}\|v\|$. Then, by statement A, $e^{\mu(A)}\|v\| \ge \|e^{A}v\| = e^{\Re\lambda}\|v\|$. This proves statement B.

By the definition of the logarithmic norm, we have, as $\epsilon \to +0$,

$$\mu(A) \ge \frac{\|u + \epsilon Au\| - \|u\|}{\epsilon\|u\|} + o(1) \ge -\frac{\|Au\|}{\|u\|} + o(1) \quad \forall u.$$

The triangle inequality was used in the last step. Statement C follows.

By the last formula,

$$-\mu(A) \le \inf_u \frac{\|Au\|}{\|u\|} = \inf_v \frac{\|v\|}{\|A^{-1}v\|} = \frac{1}{\|A^{-1}\|}.$$

Since $\mu(A) < 0$, this proves statement D. A completely different proof is indicated by the hints of Problem P11a. See also generalizations in Problem P11b. □

A corollary of statement B is that *A is non-singular if $\mu(A) < 0$.*

For some differential systems *the sharpness of the bound given by Theorems 13.1.22 and 13.1.23 may strongly depend on the choice of vector norm.* We shall therefore here indicate how one can make the logarithmic norm techniques more efficient in practice. Since this is rather abstract and technical, some readers may prefer to proceed directly to Sec 13.2, and to study the end of the present section later, together with §13.8.1 and §13.9.1, where some related matters are developed more thoroughly.

Let $\|u\|$ be a given vector norm, and let T be a given non-singular matrix, with condition number $\kappa(T) = \|T\|\|T^{-1}\|$. Define a new norm by $\|u\|_T = \|T^{-1}u\|$. It is easily proved that the axioms for a vector norm are satisfied. You may call it a **T-norm**. Then

$$\|A\|_T = \|T^{-1}AT\|, \quad \mu_T(J) = \mu(T^{-1}JT). \tag{13.1.35}$$

The proofs are left for Problem P12c. Also note that an arbitrary inner-product norm can be considered as a transformed l_2-norm, where T^{-1} is the right Cholesky factor of the positive definite matrix G that defines the inner-product.

In view of statement B of Theorem 13.1.25 it is natural to ask if we, for a given matrix J can find a transformation T that is *efficient* in the sense that $\mu_T(J) = \max_i \Re\lambda_i(J)$, i.e., the largest real part in the spectrum of J. The analogous question for operator norms was answered by Theorem 10.2.9 and its corollary. The proofs of the following results are omitted, since they are very similar to the proofs given in §10.2.4.

Theorem 13.1.26.
Given a matrix $A \in \mathbf{R}^{n\times n}$, and set $\alpha(A) = \max_i \Re\lambda_i(A)$. Denote by $\|\cdot\|$ any l_p-norm (or weighted l_p-norm), $1 \le p \le \infty$.
The following holds, with the notations of (13.1.35):

(a) *If A has no defective eigenvalues with $\Re\lambda = \alpha(A)$, then there exists a matrix T such that $\mu_T(A) = \alpha(A)$.*

(b) *If A has a defective eigenvalue with $\Re\lambda = \alpha(A)$, then for every $\epsilon > 0$ there exists a matrix $T(\epsilon)$, such that $\mu_{T(\epsilon)}(A) \le \alpha(A) + \epsilon$.*
As $\epsilon \to 0$, the condition number $\kappa(T(\epsilon))$ tends to ∞ like ϵ^{1-m^*}, where m^* is the largest order of a Jordan block belonging to an eigenvalue λ with $\Re\lambda = \alpha(A)$.

(c) *If $\alpha(A) < \alpha^*$, then there exists an inner-product norm, such that the subordinate logarithmic norm is $\mu(A) < \alpha^*$.*

Corollary 13.1.27 (Asymptotic Stability).
If A is a constant matrix, the following conditions are equivalent:

(a) The origin is asymptotically stable for the autonomous system $\dot{y} = Ay$.

(b) $\lim_{t\to\infty} e^{At} = 0$.

(c) $\alpha(A) < 0$.

(d) There exists an inner-product norm such that $\mu(A) < 0$, Equivalently, there exists a positive constant δ such that, in this norm, $\mu(A) < -\delta$.

Corollary 13.1.28 (On Stability).
If A is a constant matrix, the following conditions are equivalent:

(a) *The origin is stable for the autonomous system $\dot{y} = Ay$.*

(b) *e^{At} is bounded for $t \in [0, \infty)$.*

(c) *No eigenvalue of A has a positive real part, and there are no defective eigenvalues on the imaginary axis.*

(d) *There exists an inner-product norm such that $\mu(A) \le 0$,*

Sometimes a weighted max-norm is a simple alternative to the inner-product norm in Condition (d), e.g. for matrices with positive off-diagonal elements, see the beginning of §13.9.1, and Problem 1 of Sec. 13.9.

We shall apply Theorem 13.1.26 to give a short proof of part of a classical theorem due to Lyapunov.

Theorem 13.1.29.

Consider the system $\dot{y} = Ay + g(t, y)$, where $\|g(t, y)\| = o(\|y\|)$, uniformly in t. (The constant matrix A is the Jacobian at the origin, for all t.)

(1) *If the origin is asymptotically stable for the linear approximation $\dot{y} = Ay$, it is so also for the non-linear system.*

(2) *If $\alpha(A) > 0$ then the origin is unstable, or possibly conditionally stable, for the non-linear system.*

(3) *If $\alpha(A) = 0$, then the stability question for the non-linear problem cannot be settled merely by the discussion of the linear approximation $\dot{y} = Ay$.*

Proof. By the above results we can, for some positive δ, find a norm such that $\mu(A) < -\delta$, and then find a constant ρ such that $\|g(t, y)\| < \frac{1}{2}\delta\|y\|$ when $\|y\| < \rho$. Finally we choose $\|y(0)\| < \rho$. Then, as long as $\|y(t)\| < \rho$, we have, by (13.1.30),

$$\|y(t)\|' \le \mu(A)\|y(t)\| + \|g(t, y(t))\| \le -\delta\|y(t)\| + \tfrac{1}{2}\delta\|y(t)\| = -\tfrac{1}{2}\delta\|y(t)\|.$$

We see that $\|y(t)\|$ is decreasing, hence the inequalities $\|y(t)\| < \rho$ and $\|y(t)\|' < -\frac{1}{2}\delta\|y(t)\|$ hold forever, and hence $\|y(t)\| < \rho e^{-\frac{1}{2}\delta t} \to 0$, as $t \to \infty$.

This proves the first part of the theorem. The second part will be proved in §13.8.1. Finally, the third part is demonstrated by the problems $\dot{y} = \pm y^3$, with the solutions $y(t) = y(0)(1 \pm 2y(0)^2 t)^{-1/2}$, which have the same linear approximation, $\dot{y} = 0$, for which the origin is non-asymptotically stable. We see that the origin is unstable for $\dot{y} = y^3$, and asymptotically stable for $\dot{y} = -y^3$. □

We now consider some possibilities and difficulties related to making the bounds given by Theorems 13.1.22 and 13.1.23 sharp. Sometimes it is advisable to use a different norm *internally* for deriving a bound and then transform the result to the *externally* used norm.

By Theorem 13.1.26, the non-negative number $\mu(A) - \max_i \Re\lambda_i(A)$ is a measure of the non-efficiency of the logarithmic norm for a particular choice of vector norm.

This measure does not tell the whole truth, however, since it *can* happen that the norms for which $\mu(A)$ is small are, in Wilkinson's word, *"bizarre"*. This means that the matrix T *can be extremely ill-conditioned in a situation where we wish to express the assumptions and the final results in terms of our external norm* $\|\cdot\|$; the T-norm is then just an aid to obtain μ_T^* instead of μ^* in some exponents.

Consider, for example, the linear system $\dot{u} = J(t)u + r(t)$, and suppose for the sake of simplicity that $\mu^* \neq 0$, $\mu_T^* \neq 0$. By (13.1.32), we have for the external

norm,

$$\|u(t)\| \le \|u(a)\|e^{\mu^* t} + \epsilon\frac{e^{\mu^* t}-1}{\mu^*}.$$

If the T-norm is used, we obtain the bound

$$\|u(t)\|_T \le \|u(a)\|_T e^{\mu_T^* t} + \epsilon_T\frac{e^{\mu_T^* t}-1}{\mu_T^*},$$

where ϵ_T is a bound for $\|r(t)\|_T$. In order to eliminate $\|u(t)\|_T$, $\|u(a)\|_T$, ϵ_T it is, in a general discussion, hard to do anything better than using the inequalities $\|u(a)\|_T \le \|T^{-1}\|\cdot\|u(a)\|$ and $\|\epsilon_T\| \le \|T^{-1}\|\epsilon$ for the input and $\|u(t)\|_T \le \|T\|\cdot\|u(t)\|$ for the output. These two factors make together the condition number $\kappa(T)$. The above bound for $\|u(t)\|_T$ is then to be replaced by the bound

$$\|u(t)\| \le \kappa(T)\left(\|u(a)\|e^{\mu_T^*\cdot t} + \epsilon\frac{e^{\mu_T^* t}-1}{\mu_T^*}\right).$$

The condition number does no harm in a proof of a theorem like 13.1.29; as $t\to\infty$ the effect of the reduced exponents dominates the effect of a big constant factor in front of the expression, in particular if (say) $\mu_T^* < 0$ while $\mu^* > 0$. In a bound that is to be applied in finite time, however, the factor $\kappa(T)$ can be a disaster.

This reminds of the discussion of iterative processes in Ex. 11.1.3, and we shall also see other important situations, where similarity transformations are involved in the discussion of bounds for the growth, e.g. in next section and in Sec. 13.8 — where we shall also discuss modifications when T must depend on time. In all these cases we insist in *emphasizing that the transformation should be well conditioned*. There exist, however, other situations, where the T-norm is no less adequate for external use than the original norm. Then no warning is needed; see, e.g., the beginning of §13.2.1.

In some applications one may need a T-norm, where T depends on t. This can be handled, although it requires a modification of Theorems 13.1.10 and 13.1.11; see Theorems 13.2.1 and 13.9.1.

Among other properties and applications of the logarithmic norm dealt with in later sections, we mention:

• A criterion for the existence and uniqueness of a solution to a non-linear algebraic system; see Theorem 13.2.20.

• The logarithmic norm subordinate to a general inner-product norm; see Sec. 13.8.

• Logarithmic norm techniques for block matrices, with applications to ODEs, see §13.8.3.

Finally, Söderlind [34] has generalized logarithmic norms in a way that for a nonlinear function $f : D \subset \mathbf{R}^s \to \mathbf{R}^s$ requires a Lipschitz condition only. (We have above assumed the integrability of the Jacobian matrix.) Let

$$L[f] = \sup_{u\neq v}\frac{\|f(u)-f(v)\|}{\|u-v\|} \qquad u, v \in D.$$

be the Lipschitz constant over a path-connected set D. Then Söderlind's generalization reads

$$M[f] = \lim_{h \to 0+} \frac{L[I + hf] - 1}{h}.$$

Review Questions

R1. Explain, with an example, how a differential equation of higher order can be written as a system of first-order differential equations.

R2. Define local and global error, and explain with a figure the error propagation for a one-step method for a scalar differential equation. What is meant by order of accuracy?

R3. (a) Formulate the basic existence and uniqueness theorem for initial value problems for the system $\dot{y} = f(y)$. Set up the iteration formula used in the demonstration.

(b) Demonstrate the existence and uniqueness theorem. What is the relation between the Lipschitz constants of $f(y)$ and the iteration operator?

(c) Give a simple example of a differential equation, where the solution escapes to ∞ in finite time. Explain why this does not contradict the existence and uniqueness theorem.

(d) Show that the initial value problem $dy/dt = y^{2/3}$, $y(0) = 0$, has an infinite number of solutions. Explain why this does not contradict the existence and uniqueness theorem.

(e) What is a critical point p of an autonomous system $\dot{y} = f(y)$? Let p be a critical point. Prove the following statement, under appropriate conditions:
If $y(t_1) = p$, at some time t_1, then $y(t) = p$ for all times t.
Can an orbit reach a critical point in finite time, if it is not there from the beginning? (Motivate the answer.)

Hint: Use the uniqueness theorem.

R4. Define and derive the non-linear and the linearized variational equations, and state a few results that can be deduced from them. In Example 13.1.11 (Predator-Prey Problem), it is stated that "on a microscopic scale the mapping is approximately affine". Explain this statement, and give a motivation for it.

R5. (a) Define the logarithmic norm of a matrix.

(b) Give the formulas for the logarithmic norm subordinate to some common norms, and derive it for one of these norms. Compute, for some given numerical 2×2-matrix A, $\mu_\infty(A)$, $\mu_1(A)$ and $\mu_2(A)$, and a simple upper bound for $\mu_2(A)$ that does not require eigenvalues.

(c) Show that if $\dot{u} = J(t)u + r(t)$, $t \geq a$, then $\|u\|'(t) \leq \mu(J(t))\|u(t)\| + \|r(t)\|$. Also show that $\|y(t)\| \leq \psi(t)$, where $\psi(t)$ satisfies an ODE of the form $\psi' = \mu^*(t)\psi + r^*(t)$. Give the conditions that $\psi(a), \mu^*(t), r^*(t)$ satisfy. Solve

this equation, if μ^* and r^* are constant, and compute $\lim_{t\to\infty} \psi(t)$. Sketch the behavior of $\psi(t)$ for various values of μ^*.

(d) Generalize the previous question to an approximate bound for $z(t) - y(t)$, where $y(t), z(t)$, satisfy the ODEs, $\dot{y} = f(y)$, $\dot{z} = f(z)+r(t)$, under appropriate conditions that include the conditions of the previous question as particular cases.

(e) Tell five of the most important general properties of the logarithmic norm, in particular the subadditivity.

(f) Show that $\|e^{At}\| \leq e^{\mu(A)t}$ for $t \geq 0$, and that $(zI - A)^{-1} \leq (\Re z - \mu(A))^{-1}$ under a condition that is to be specified.

R6. Formulate and explain, with your own words and pictures, the Comparison theorem and the Positivity theorem (in several variants). Formulate also the Basic theorem that these are corollaries from. Make applications.

R7. Define stability and asymptotic stability in the sense of Lyapunov. If you like, please draw a figure instead of using ϵ, δ. For a linear autonomous system $\dot{y} = Ay$, give conditions for these types of stability, both in terms of eigenvalues and in terms of the logarithmic norm. Are the conditions necessary, sufficient or both? What do you know of the relations between the stability properties of a critical point p for a nonlinear system $\dot{y} = f(y)$ and for its linear approximation $\dot{y} = f'(p)(y - p)$?

Problems

P1. (a) Rewrite the system

$$\ddot{y} = t^2 - \dot{y} - z^2,$$
$$\ddot{z} = t + \dot{z} + y^3,$$

with initial conditions $y(0) = 0$, $\dot{y}(0) = 1$, $z(0) = 1$, $\dot{z}(0) = 0$, as an initial value problem for a system of first-order equations.

(b) Write the linear differential equation $u^{(k)} = \sum_{i=0}^{k-1} a_i u^{(i)}$ as a system $\dot{y} = Ky$ for the vector $y = (u, \dot{u}, \ldots, u^{(k-1)})^T$. If λ is an eigenvalue of K, what is the corresponding eigenvector? *Hint*: See Problem 8 of Sec 10.1, but note that the conventions are different.

P2. (a) Find the general solution to the differential equation $\dot{y} = y/t$. Where is the Lipschitz condition not satisfied? Study the behavior of orbits in the neighborhood of such singular points.

Also study the system $\dot{y}_1 = y_1, \quad \dot{y}_2 = y_2$. Note that the singular point of the single differential equation $\dot{y} = y/t$ corresponds to a critical point of this system.

(b) Study in a similar way the single differential equation $\dot{y} = at/y$, $a \in \mathbf{R}$, and the system $\dot{y}_1 = ay_2$, $\dot{y}_2 = y_1$. How does the character of the solution manifold depend on the parameter a? (See also Exercise C3 below.) Reduce,

by appropriate scaling of y, t, the discussion to the three cases $a = \pm 1$ and $a = 0$.

P3. (a) Four of the orbits in Fig. 13.1.5 seem to be envelopes of families of straight lines. Explain this observation.

(b) Determine theoretically the speed of the rotation and shrinking of the squares in the figure, and make, for comparison, some measurements in the figure. Find a crude estimate of the end value of t used in the computation of the figure. (Measure the "white hole" .)

(c) More about linear systems with constant coefficients. Incomplete!!!!!

(d) Show by a simple example that a critical point of a non-linear differential equation can be asymptotically stable, although it is only stable for the the linear approximation of the equation.

P4. (a) Consider the differential equation $\dot{y} = 1 - y^2$. Study the behaviour of $y(t)$ as $t \to \infty$ for all initial values, both by the qualitative technique of Example13.1.13 and by means of the exact solution of the differential equation, suitably expressed in the form $t = \phi(y) + C$. Which do you prefer?
If $y(0) = 1$, a crude approximation to the solution reads $y(t) = t$ for $t < 1$, $y(t) = 1$ for $t \geq 1$. What is the largest error of this approximation?
Also show that $y(t) = 1 - \delta$ for $t \approx \frac{1}{2} \ln \frac{2}{\delta}$, if $y(0) = 0,\ \delta \ll 1$. Note that $t \approx 1.5$ for $\delta = 0.1$ and $t \approx 2.6$ for $\delta = 0.01$. We can roughly say that the "scale" of t and y is 1 for this problem.
Similarly, what is the "scale" of t and y for the problem $\dot{y} = a - by^2,\ y(0) = 0,\ a > 0,\ b > 0$, i.e. what is the maximum of $y(t)$ and when is it attained to 90%?
Take $a = 0.01,\ b = 10^6$. (Compare P4 below.)

(b) Discuss in a similar way the **logistic equation**, $\dot{u} = cu(d - u)$. For what value of u does $u(t)$ has an inflexion point? (Answer this without solving the differential equation.)
What is the relation between the solution of $\dot{u} = u(1 - u)$ that starts in the inflexion point and the solution of $\dot{y} = 1 - y^2$ with $y(0) = 0$? If $u(0) < 0$, does $u(t)$ become $-\infty$ for a finite value of t or in the limit as $t \to \infty$? *Hint*: How does the integral $t = \int du/f(u)$ behave?

(c) *Bimolecular reaction.* Consider a chemical reaction

$A + B \xrightarrow{k_1} C$. Denote the *concentration* of the species $A,\ B,\ C$ by $y_1,\ y_2,\ y_3$, respectively; By **the law of mass action**,the concentration y_3 is during an "infinitesimal" time interval of length dt increased by the amount $r_1 dt = k_1 y_1 y_2 dt$, while $y_1,\ y_2$ are decreased by the same amount. k_1 is a characteristic of the reaction called the *rate constant*. It may depend on the temperature, but we assume here that the reaction is kept isothermal.

Suppose that there is a reaction in the opposite direction $C \xrightarrow{k_2} A + B$, where y_3 is decreased by the amount $r_2 dt = k_2 y_3 dt$ during a time interval dt, while y_1, y_2 are increased at the same rate. The quantities $r_i,\ i = 1, 2$, are called

reaction rates. This leads to the system of algebraic and differential equations:

$$\begin{aligned} r_1 &= k_1 y_1 y_2, \qquad r_2 = k_2 y_3 \quad k_1 > 0.\ k_2 > 0, \\ \dot{y}_1 &= -r_1 + r_2, \quad y_1(0) = a > 0, \\ \dot{y}_2 &= -r_1 + r_2, \quad y_2(0) = b > a, \\ \dot{y}_3 &= r_1 - r_2, \quad y_3(0) = 0, \end{aligned}$$

Note that $\dot{y}_1 + \dot{y}_3 = 0$, $\forall t > 0$, and hence $y_1 = a - y_3$, $\forall t > 0$. Show that $y_2 = b - y_3$. The problem is thus reduced to a single differential equation, satisfied by $y(t) = y_3(t)$:

$$\dot{y} = f(y), \quad y(0) = 0, \quad \text{where} \quad f(y) \equiv k_1(a - y)(b - y) - k_2 y.$$

Show, without solving the differential equation, that $y(t) > 0$, $\forall t > 0$, and that the equation has two critical points, one of which, p_1 (say), is asymptotically stable. Why is the other critical point unstable? Also show that $p_1 < a < b$ and that $y_3(t)$ converges monotonically to p_1, ultimately faster than $e^{f'(a)t}$.

Hint: Recall Example **??**. Sketch the graph of $f(y)$, $y \geq 0$, and note that $f(0) > 0$, $f(a) < 0$, $f(y) \to +\infty$ as $y \to +\infty$. Also note that $f'(a) < 0$.

For $k_1 = 1$, $k_2 = 2$, $a = 0.4$, $b = 0.6$, determine p_1, $f'(a)$ and the ultimate rate of exponential convergence $f'(p_1)$.

P5. (a) Show that the *paths* of two solutions of the autonomous system $\dot{y} = f(y)$, $y(t) \in \mathbf{R}^s$ cannot intersect. Show that this is not generally true for two *solution curves* $\big(y(t), t\big) \in \mathbf{R}^s \times \mathbf{R}$ of a non-autonomous system $\dot{y} = f(y, t)$.

(b) By a simple example, show that two solution curves in the yt-plane for a single second order equation $\ddot{y} = f(y)$ can indeed intersect, in fact in more than one point.

Suppose that you compute a one parameter family of initial value problems for the system $\dot{y} = f(y)$, $y \in \mathbf{R}^s$, $s > 2$ and you plot y_1 versus y_2. Can it happen that two curves in this picture intersect in more than one point?

(c) If the path for a solution to $\dot{y} = f(y)$, $y \in \mathbf{R}^s$ returns to a point where it has been before, show that the same orbit will be repeated over and over again. Hence the solution $y(t)$ becomes a periodic function of t. Show, by example, that the length of the period is not necessarily the same for all paths belonging to a system.

Construct a system of 3 autonomous ODEs, where the pictures of y_1 versus y_2 are all circles, but the solutions are not periodic functions of t.

(d) Some of our examples may give the impression that the presence of the term ay^2, $a > 0$ on the right hand side of a single ODE $\dot{y} = f(y)$ may cause a catastrophic growth of most solutions. The following two cases tell that it is not necessarily so.

Consider the equation $\dot{y} = -y + ay^2$. Show that, for any $a > 0$, $y(t)$ tends to zero monotonically and exponentially if $y(0) < 1/a$. What happens if $y(0) > 1/a$? (You do not need to compute then explicit solution.)

Consider the equation $\dot{y} = 1 - y + ay^2$. Suppose that $0 < a < \frac{1}{4}$. Show that there are two critical points, $0 < y_1 < y_2$, and that $y(t)$ converges monotonically and exponentially to y_1 if $y(0) < y_2$. What happens if $a > \frac{1}{4}$ or $a = \frac{1}{4}$?

P6. We use the terminology introduced in P4. Let A, B, C be three chemical species with concentrations y_1, y_2, y_3; initially $y_1 = 1$, $y_2 = y_3 = 0$. The following is a set of chemical reactions called autocatalytic:

$$C + C \xrightarrow{k_1} B + C; \quad C + B \xrightarrow{k_2} A + B; \quad A \xrightarrow{k_3} C.$$

Although B is both a winner and a loser of the second reaction, this must *not* be simplified to $C \longrightarrow A$; its reaction rate is $r_2 = k_2 y_2 y_3$. The other reaction rates are $r_1 = k_1 y_3^2$, $\quad r_3 = k_3 y_1$.

(a) Explain that the law of mass action yields the differential system $\dot{y} = Ar(y)$, where

$$y(0) = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad A = \begin{pmatrix} 0 & 1 & -1 \\ 1 & 0 & 0 \\ -1 & -1 & 1 \end{pmatrix}, \quad r(y) = \begin{pmatrix} k_1 y_3^2 \\ k_2 y_2 y_3 \\ k_3 y_1 \end{pmatrix}.$$

(b) Show that all variables are non-negative for $t > 0$, as they should, because they are concentrations.

(c) Show that $y_1(t) + y_2(t) + y_3(t) = 1$, by adding the three equations etc. Relations like this are common in many applications. They are called **linear invariants**. For a general system $\dot{y} = f(y)$, show that $b^T y(t)$ is a linear invariant, if $b^T f(y) = 0$, $\forall y \in \mathbf{R}^s$. ($b^T$ is a constant row vector.) Also show that every independent linear invariant of a system reduces the rank of $f'(y)$ by one, for all y. Show how invariants for a system of the form $\dot{y} = Ar(y)$, can be found (if there are any), by applying a numerical algorithm to the matrix A.

(d) Determine the critical point p of the system in (a). Show that the Jacobian is of rank 2 if $y_3 \neq 0$, but at the critical point (and at the starting point) it is of rank 1. (It can be shown that this has the effect that the convergence towards the critical point becomes slower than exponential. This is rather common in chemical systems.)
Determine $c \in \mathbf{R}^3$ so that $y(t) = p + ct^{-1}$ satisfies this system approximately for $t \gg 1$,

(e) Let ϵ be a small positive number. Consider the case $k_1 = \epsilon^{-2}$, $k_2 = \epsilon^{-1}$, $k_3 = 1$, set $y_1 = z_1$, $y_2 = z_2$, $y_3 = \epsilon z_3$, and make use of the invariant.
Show that the system reads

$$\begin{aligned} z_1 &= 1 - z_2 - \epsilon z_3, \\ \dot{z}_2 &= z_3^2, \quad z_2(0) = 0, \\ \epsilon \dot{z}_3 &= -z_3^2 - z_2 z_3 + z_1, \quad z_3(0) = 0, \end{aligned}$$

Note that the first equation is an algebraic equation, not a differential equation. This problem is continued in computer exercise C2.

(f) Usually the ODEs derived from the law of mass action, can also be written in the following form, $y_i' = P_i(y) - y_i Q_i(y)$, $i = 1, 2, \ldots, s$, where $P_i(y)$, $Q_i(y)$ are functions of the vector y, usually polynomials, which are non-negative for every vector with non-negative components. Show that $y(t)$ is non-negative for all $t > 0$, if the initial values are so.
Rewrite the equations of Problem (a) in this form.

COMMENT: One may not find *all* linear invariants by the algorithm, that was asked for in (c), but if the components of $r(y)$ are monomials in y — as they are in most systems derived from the law of mass action (and in many other applications) — one has a good chance to succeed.
The matrix A is typically a constant, rectangular and sparse matrix with small integer elements. Although rounding errors are usually important in matrix computations, they will typically cause very little trouble in this context.
In Problem P3(d) the invariants $y_1 + y_3 = a$, $y_2 + y_3 = b$, were used to eliminate y_1, y_2 *symbolically* in the third differential equations, although this terminology was not used there. The differential equations for $\dot{y}_1$ and $\dot{y}_2$ were then replaced by invariants. In general, the order of the differential system can be reduced by the number of linearly independent invariants; some care is needed in the choice of the differential equations to be replaced by algebraic equations.
In more complicated systems it is usually easier to *avoid the symbolical substitutions*: let a variable that has lost its differential equation remain in the formulas for the other differential equations; their *numerical* values are in every time step computed by means of the invariants before the variables that still have differential equations are treated.
Alternatively, linear and non-linear invariants, also called *first integrals*, can often be derived from some physical conservation laws.

P7. A study of a population subject to competition (crowding) and toxins. (Extension of a model treated in Klamkin [24, p. 317].)
Let $u(t)$, $v(t)$ be, respectively, the size of a population and the amount of toxins at the time t. Assume that the difference $r(u)$ between the birth rate and the death rate, due to other causes than the toxins, is of the form $r(u) = k_0 - k_1 u$, like in the logistic model (which is a particular case of this problem).
Furthermore, assume that $k_2 uv$ individuals are killed, per unit of time, by toxins in the environment. These toxins are produced by the population, and assume that the amount produced per unit of time is $k_3 u$. Finally, assume that that the toxins have a spontaneous exponential decay (see the term $-k_4 v$). All parameters are *non-negative*. This leads to equations of the form

$$\dot{u} = (k_0 - k_1 u)u - k_2 uv, \quad \dot{v} = k_3 u - k_4 v.$$

(a) Reduce the number of parameters from 5 to 2 by scaling t, u, v, i.e. set $t = lx, u = my_1, v = ny_2$, so that we obtain a differential system of the more

transparent form

$$\dot{y}_1 = y_1 - y_1^2 - a y_1 y_2, \quad \dot{y}_2 = y_1 - b y_2. \tag{13.1.36}$$

Express $a,\ b,\ l,\ m,\ n$ in terms of the original parameters. See also Exercise C6.

(b) Determine the critical points of (13.1.36) and the rate of convergence to them in the asymptotically stable cases.

(c) Is it true that, with this model, neither y_1 nor y_2 can become negative, if the initial values are positive? Assume that $y_1(0)$ is a small positive number and that $y_2(0) = 0$. Find an upper bound for y_1.

P8. Sometimes the perturbations grow much faster than the exact solution itself. Make theoretical studies of the following example.

Verify that the problem $\dot{y} = y - 2t/y$, $y(0) = 1$, has the solution $y = \sqrt{2t+1}$. Show by the linearized variational equation that a small perturbation at $t = 0$ will be amplified by the factor $e^{2t}/\sqrt{2t+1}$. Compare the growth of the perturbations and the growth of the solution over the interval (0, 12).

Another way to the understanding of this example is to note that $z = y^2$ satisfies the linear problem $\dot{z} = 2z - 4t$. For example, give the starting value a small negative perturbation, $z(0) = 1 - \delta$ (say). Show that $z(t) < 0$, when $2t > \ln(1/\delta) + \sqrt{2\ln(1/\delta)}$; the correspondingly perturbed $y(t)$ is no longer real. (This will be followed up in exercise C10.)

P9. (a) Compute $\exp(\|A\|_\infty)$, and $\exp(\mu_\infty(A))$ for

$$A = \begin{pmatrix} -10 & -1 \\ 3 & -10 \end{pmatrix}.$$

(b) Treat Problem (a) with the l_2-norm instead of the max-norm. Show also that $\mu(A) \geq -10$ for any choice of norm.

(c) Let $y(t)$ be the solution of an equation of the form $\dot{y} = Ay + r(t),\ y(0) = 0$, where A is the same matrix as in (a) and (b), and $\|r(t)\|_\infty \leq 1$. Find a constant c such that $\|y(t)\|_\infty \leq c$ for all positive t.

P10. (a) Derive the expressions for $\mu_1(A),\ \|I + \epsilon A\|_w$, and $\mu_w(A)$ given in Theorem 13.1.20.

(b) Derive, by means of Theorem13.1.9, a bound for $\psi(t)$ from the differential equation (13.1.32). We can obtain a uniform bound for all $t > a$, if we make further assumptions about $\mu^*(t), \epsilon(t)$. For example, assume that

$$\text{either} \quad \mu^*(t) < -\delta \ (\delta > 0), \quad \text{and} \quad \epsilon(t) \leq c_2, \ (\forall t > a),$$

$$\text{or} \quad \mu^*(t) \leq 0, \forall t > a, \quad \text{and} \quad \int_a^\infty \epsilon(t)\, dt \leq c_2.$$

Find uniform bounds for $\psi(t)\ \forall t > a$ under these assumptions.

More generally (also if $\mu^*(t) > 0$), show that $\psi(t) \le e^{c_1}(\psi(a) + c_2), \forall t > a$, if

$$\int_x^t \mu^*(s)ds \le c_1, \ \forall x, t, \ a \le x \le t, \quad \text{and} \quad \int_a^\infty \epsilon(t)dt \le c_2.$$

(c) Consider the system $\dot{y} = A(t)y$, where

$$A(t) = \begin{pmatrix} \dfrac{2}{1+t^2} & \dfrac{2}{2+t^2} \\ \dfrac{1}{2+t^3} & \dfrac{1}{2+t^2} \end{pmatrix}.$$

Show that all solutions are bounded as $t \to \infty$, in spite that both eigenvalues have positive real parts for every $t > 1$ (by Gershgorin's theorem).

(d) Consider the real differential system $\dot{y} = A(t)y + r(t), \quad t > 0$, $y(0)$ given. Assume that the symmetric part of $A(t)$, i.e., $B(t) = \frac{1}{2}(A(t) + A(t)^T)$, is negative semi-definite and that $\int_0^\infty \|r(t)\|dt$ is convergent. Show that $\|y(t)\|$ is bounded as $t \to \infty$.

(e) Show that the same conclusion holds, if you instead assume that the eigenvalues of $B(t)$ are less than a negative constant $-\delta$ and $\|r(t)\| \le C, \ \forall t \ge 0$.

P11. (a) Consider the linear system $\dot{y} = Ay - r$, where A and r are constant, and A is non-singular. Note that $A^{-1}r$ is a critical point, and show that *every motion converges to this point, as* $t \to \infty$, *if* $\mu(A) < 0$. Use this to show that

$$\|A^{-1}\| \le |\mu(A)|^{-1}, \qquad \text{if} \quad \mu(A) < 0.$$

COMMENT: The derivation of this result in Theorem13.1.25 is very different.

(b) Show the following relations:

$$\|(I - hJ)^{-1}\| \le (1 - \mu(hJ))^{-1}, \qquad \text{if } \mu(hJ) < 1,$$

$$\|(A - zI)^{-1}\| \le (\Re z - \mu(A))^{-1}, \qquad \text{if } \Re z > \mu(A).$$

P12. (a) Derive the inequality $|\mu(A) - \mu(B)| \le \|A - B\|$, and show that $\mu(A(t))$ is a continuous function of t if $A(t)$ is so. Also show that $\mu(f'(y))$ is continuous if $f'(y)$ is so.
Find an example where $d\mu(A(t))/dt$ is not continuous, in spite that $A'(t)$ is continuous.

(b) Consider the matrix differential equations,

$$\dot{Y} = A(t)Y, \qquad \dot{Z} = ZB(t),$$

where $Y, Z, A(t), B(t)$, are $n \times n$ matrices. Show that

$$\|Y\|' \le \mu(A(t))\|Y\|, \qquad \|Z\|' \le \mu(B(t))\|Z\|,$$

where the matrix norm and the logarithmic norm are subordinate to the same vector norm. Generalize to inhomogeneous equations,e.g., $\dot{Z} = ZB(t) + C(t)$.

Hint: Proceed like the beginning of the proof of Theorem 13.1.22.

(c) A norm $\|\cdot\|$ is given, and define another norm, $\|u\|_T = \|T^{-1}u\|$, where T is a non-singular matrix. Show that [14] $\|A\|_T = \|T^{-1}AT\|$ and $\mu_T(A) = \mu(T^{-1}AT)$.

P13. There are several alternative definitions of the logarithmic norm, some of which are better suited for a generalization to semibounded operators in infinite-dimensional spaces. Show that $\mu(A)$ is equal to the limits of the following three expressions,

$$\frac{\|(I-\epsilon A)^{-1}\|-1}{\epsilon}, \quad \frac{\ln\|I+\epsilon A\|}{\epsilon}, \quad (\epsilon \downarrow 0); \qquad \|A+kI\|-k, \quad (k\to\infty).$$

Use the third expression in an alternative derivation of Theorem 13.1.22: set $u = e^{-kt}v$, derive a differential inequality for $\|v\|$, then return to $\|u\|$. Show that the inequality becomes sharpest in the limit, $k \to \infty$.

P14. (a) Suppose that $f'(y)$ is continuous, and that $\mu(f'(y)) < 0$ in $D_R = \{y : \|y-p\| \le R\}$, where p is a critical point, $f(p) = 0$.
Show that if $y(t') \in D_R$ for some time t', then $y(t) \to p$ exponentially as $t \to \infty$.

(b) By means of the example $f(y) = -y^3$, show that the convergence is not necessarily exponential if $\mu(f'(y)) < 0$ in $D_R = \{y : \|y-p\| \le R\}$, *with the exception of the point p.*

P15. (a) In the notation of Example 13.1.24, show that $\|e(t)\|_2 \le \eta_0^3/6$.

(b)Let $y(t)$ be the solution of the initial value problem $\dot y = t^2 + y^2, \quad y(0) = 1$. Show that the solution is still finite at $t = 0.833$ by comparing $y(t)$ with a solution of the differential equation, $\dot z = a^2 + z^2$, for some suitable choice of a. Also show theoretically that $y(t)$ becomes infinite before $t = 1$. *Hint:* Set $y = 1/u$. (See also exercise C11.)

P16. Prove Theorem 13.1.9.

P17. (a) Assume that all solutions of the linear system $\dot u = A(t)u$ satisfy the inequality $\|u(t)\| \le c_0\|u(x)\| \ \ \forall t \ge a, \ \ \forall x \in (a,t)$. Let $U(t)$ be the fundamental matrix solution of this linear system, see (13.1.18) and Theorem 13.1.9. Show that $\|U(t)U(x)^{-1}\| \le c_0$.

(b) Assume that $B(t,u)$ is a matrix-valued function such that $\|B(t,u)\| \le c_1, \ \forall t \ge a$ if $\|u\| \le c_2(t)$. Then show that all solutions of the pseudo-linear system $\dot u = \big(A(t) + B(t,u)\big)u$ satisfy the inequality,

$$\|u(t)\| \le c_0\|u(a)\| + c_0\int_a^t c_1\|u(x)\|dx,$$

as long as this inequality implies that $\|u(t)\| \le c_2(t)$. Generalize to the case when c_1 is allowed to depend on t.

(c) **The Gronwall–Bellman Lemma.** Let $g(t)$ be a differentiable function,

[14] If you do not solve this, study the proof of (13.8.2).

and let $k(t)$ be a continuous function, such that $k(t) \geq 0$, $\forall t \geq a$, and set $K(t) = \int_a^t k(x)dx$. Assume that a continuous function $y(t)$ for $t \geq a$ satisfies the inequality

$$y(t) \leq g(t) + \int_a^t k(x)y(x)dx. \tag{13.1.37}$$

Show that if $k(t) = k > 0$, $g(t) = g$ are constant, then

$$y(t) \leq ge^{k(t-a)}.$$

More generally, show that $y(t) \leq g(t) + \int_a^t k(x)g(x)e^{K(t)-K(x)}dx$.
Apply the results to find a bound for $\|u(t)\|$ in Problem (b).

Hint: Let $w(t)$ be the solution of the integral equation obtained when the inequality in (13.1.37) is replaced by an equality. Note that $w(t)$ is the solution of the differential equation $w'(t) = g'(t) + k(t)w(t)$, $w(a) = g(a)$. Solve this differential equation by the formula of Theorem 13.1.9, and integrate the result by parts.

NOTE: This lemma gives an alternative approach to some of the questions treated by the logarithmic norm technique in this book, sometimes with stronger results. On the other hand, the restriction to non-negative functions $k(t)$ is not needed in the logarithmic norm technique. An analogous result for difference equations is given as a problem in Sec. 13.9.

P18. (C. Moler, personal communication.) Consider the initial value problem, $dy/dt = \sqrt{1-y^2}$, $y(0) = 0$. Show that the Lipschitz condition is not satisfied at $y = 1$, but nevertheless the problem has a unique *real* differentiable solution for positive values of t, namely:

$$y(t) = \begin{cases} \sin t, & \text{if } 0 \leq t \leq \frac{1}{2}\pi \ , \\ 1, & \text{if } t > \frac{1}{2}\pi. \end{cases}$$

Hint: Derive contradictions from the suppositions that $y(t) > 1$ or $y(t) < 1$ for some $t > \frac{1}{2}\pi$.

P19. (a) Study the differential equation, $\dot{y} = -1 - 2\ \mathrm{sgn}(y)$, with initial condition $y(0) = 1$. If we insist on the convention that $\mathrm{sgn}(0) = 0$, show that there is no solution for $t > \frac{1}{3}$. Why does not this contradict the existence and uniqueness theorem?
Find out, with paper and pencil or with a computer, what happens if you apply Euler's method with constant step size to this equation.

(b) This problem is related to the study of the motion of particle in a slope with Coulomb friction (dry friction).
Notations: the friction coefficient is γ, the angle between the slope and the horizontal plane is α, $0 < \alpha < \pi/2$, and the acceleration of gravity is g.
Let the velocity at time t be $v(t)$. $v(0) > 0$ is given. The positive direction is uphill. The equation of motion then reads:

$$\frac{dv}{dt} = \begin{cases} 0, & \text{if } v = 0 \text{ and } \gamma \geq \tan(\alpha); \\ -g\sin(\alpha) - \gamma g\cos(\alpha)\mathrm{sgn}(v), & \text{otherwise.} \end{cases}$$

Is it true that this problem has precisely one solution for all $t > 0$?
What value is implicitly given to sgn(0) in this formulation?
For what relation between α and γ can we obtain the initial value problem in (a) after appropriate rescaling of v and t?

Note: For the sake of simplicity, we ignored the distinction between the friction coefficients at rest and at motion. Also note that the particle reaches a critical point in finite time. According to the comments after Theorem 13.1.3 this would have been impossible for a system that satisfies a Lipschitz condition everywhere.

P20. (a) Set $\phi(t; A) = t^{-1} \ln \|e^{At}\|$. Show that $\mu(A) = \lim_{t \downarrow 0} \phi(t; A) = \sup_{t>0} \phi(t; A)$. What happens to $\phi(t; A)$, as $t \to \infty$, and what is $\inf_{t>0} \phi(t; A)$? (Compare Problem 4 of Sec. 10.2.)

(b) Discuss the analogous questions for $t < 0$, $t \uparrow 0$, and $t \to -\infty$.

P21. Find recursion formulas for the coefficients of a regular perturbation expansion, in the form of linear inhomogeneous variational equations, as mentioned in Theorem 13.1.8 and the comments to that theorem.

P22. (a) For every real constant α, the function $y(t; \alpha) \equiv \alpha t - \alpha^2$ evidently satisfies the implicit differential equation $y = \dot{y}t - \dot{y}^2$, named after Clairaut (1734). The graphs of $y(t; \alpha)$ in the (t, y)-plane are straight lines with an envelope with the equation $y = y_E(t)$ (say). Show that $y_E(t)$ also satisfies this differential equation, with the same initial condition as its tangent in the starting point. Why does not this contradict the Existence and Uniqueness Theorem 13.1.1?

(b) Generalize this to an arbitrary family of straight lines with an envelope. Look also at the analogous question for the following family of parabolas, considered by Cauchy (1824), $y = \alpha(t + \alpha)^2$. (Show first that the family satisfies the differential equation $\dot{y}^3 = 8y^2 - 4ty\dot{y}$.)

Computer Exercises

C1. Write or study a program for Runge's 2nd order method, according to Example 1.4.1, or with some other level of ambition. It is good, if it can be applied to most of the non-stiff computer exercises of this section and Sec. 13.2, in order to obtain results matching the resolution of the screen with rather short computing time on a personal computer.
The program should be convenient to apply to e.g. different initial conditions, different values of TOL, and to different values of a small number (e.g. three) of parameters of the differential system. The program should be able to provide output in numerical and in graphical form.
The numerical output should be stored in such a form that it can be processed further outside the program, e.g. for graphical output (other than the default alternative included in the program), for interpolation (in order to obtain a

neat table) or for finding intersections of the orbit with some line or plane, [15] or for the study of the step size sequence.

Read the texts of the exercises below in order to find hints about what should be included in a program to be used not only for the demonstration of the orbits, but also for the experimental study of a numerical method for ODEs. Distinguish between what can be done in *the general part* of the program and what should be done in its *problem dependent parts* of your program, and think of the communication between them.

Think also of convenient program tools for the input of the parameters of the problem and the method, for the post-processing of results that the program has produced, and perhaps also for the interaction during a run. Concerning the input: it is convenient to have a prepared set of default values of the input data, which can be overwritten by an input from the keyboard or by a driver program. The details will depend on your choice of language and the possibilities for interaction.

Note that a rough guess of the size of the first step is needed. [16] The step size control will usually soon find a suitable step size, even if the guess is off by several powers of ten, but there exist exceptional cases, where the behaviour of the program may require some analysis of some hidden difficulty in the problem. The program sketched in Example 1.4.1, usually finds a good step size faster, if the guess is too large than if it is too small.

We suggest that the program should be able to work in the following four modes:

(i) Adaptive step size control, according to Sec. 1.4. In this step size control the tolerance TOL is related to the *global* error, in a way that will be explained by Theorem 13.2.1 and the comments after Theorem 13.2.2. The number of steps will be approximately proportional $\text{TOL}^{-1/2}$. At the time of writing, these features distinguish our program from many other programs, where the relation of the tolerance to the global error is more obscure. The number of steps will, in that case, be approximately proportional to $\text{TOL}^{-1/3}$ for a 2nd order accurate method.

(ii) Constant step size

(iii) According to a prescribed sequence of times $t_0,\ t_1,\ t_2, \ldots$, usually obtained from a previous run. (This is useful when you are interested in differences between solutions with small changes of parameters or initial values. See also comments after Theorem 13.1.8.)

(iv) Every time step defined by the sequence $t_0,\ t_1,\ t_2, \ldots$ is divided into two steps of half size. This is useful for the estimation of the global error, when the exact solution is not known, and for the improvement of the results afterwards by Richardson extrapolation, see §3.3.4. One extrapolation yields 3rd order accuracy.

It is in most cases suitable to begin with $\text{TOL} = 10^{-3}$. You can then judge, for what accuracy you will still get a reasonable computing time and a reasonable

[15] It was stated in Example 1.3.1 that linear interpolation is good enough for this purpose. This is true only if Richardson extrapolation is no used.

[16] More sophisticated programs can choose the first step size without the user's help.

size of the output to be used at the post-processing.
You may also obtain good hints by looking at other codes for ODEs, e.g., those contained in Hairer, Nørsett and Wanner [20] Run also some of the computer exercises of this chapter with some professional package, such as LSODE, EPISODE, DIFEX1 or DEABM. They are described in Hairer, Nørsett and Wanner [20, p. 374 ff.].

C2. Apply the program to the equation $\dot{y} = -my^{1+1/m}$, $y(0) = 1$.

(a) Run the program for $m = \frac{1}{2}$ from $t = 1$ to $t = 10000$ with two different tolerances. Plot $\log y$, $\log h$, $N(t)/100$ and the logarithm of the relative error, divided by TOL, all of them versus $\log t$.

(b) Run also the case $m = 1$ with similar output over a longer time. Automatic interrupt when $y < 10^{-3}$ TOL.

(c) Make some computer experiments related to problems P3 and P5.

(d) Write a program for systems of the form $\dot{y} = Ar(y)$, see problem P4(a), and run it for $k_1 = k_2 = k_3 = 1$. Stop when $y_1 < 0.1$.

(e) Run the system in P4(e) for $\epsilon = 0.1$ and $\epsilon = 0.2$. Compare the solutions with each other and with the approximate expression obtained in P4(d). Then consider the same system for $\epsilon = 0$. The third equation now becomes an algebraic equation. Choose the positive root. The initial condition $z_3(0)$ must be dropped. Compare the solution obtained for the three values of ϵ,
COMMENT: This is an example of a singular perturbation problem. More general problems of this type will be discussed later.

C3. Consider systems of the form $\dot{y} = Ay$, $y(0) = c$, where A is a constant 2×2 matrix. Run a few cases, including the following:

(a) Run "the circle test" with Runge's 2nd order method. This can be written as a single complex equation, $\dot{z} = iz$, $z(0) = 1$. If your program can handle complex variables conveniently, you may run it in this form, but you can also transform it to an equivalent real system of the form $\dot{y} = Ay$, see problem P3. Determine by some combination of experimental and theoretical work, how small the tolerance and the step size have to be, in order that the circle should not become "thick" on your screen, even after 10 revolutions. Run it also with a larger tolerance, so that it becomes "thick" after two or three revolutions. Does it spiral inwards or outwards, when it has become "thick" ?
Is it true that the thickness depends much more on the linear interpolation used in the graphics than on the truncation error of the method? Do your *numerical* results in your file "spiral" in the same way (outwards or inwards) as the curve you see on your screen?
Perform the circle test also for Euler's method.

(b) Reproduce Fig. 13.1.5. This equation can also be written as a scalar complex equation. Notice that you can run all the 40 curves simultaneously by a for-loop in the "function". This gives you a system of 40 complex equations or 80 real equations. Can you handle the output from such a big system, or do you prefer to cut it into smaller pieces?

(c) Consider the differential system,

$$\dot{y} = Ay + c\|y\|_2^2 \cdot y, \quad y(0) = [1,0]^T, \quad A = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}.$$

What is e^{At}? Show that, if $c < 0$, the solution will spiral in towards the origin. What happens if $c > 0$?.

Hint: Set $y_1 + iy_2 = z$. The system then becomes a scalar complex ODE. Then set $z = re^{i\phi}$, and you will get a real system for $\dot{r}$, $\dot{\phi}$ that is easily solved analytically.

Check your conclusions by running either the original system or the complex scalar equation on a computer, for a few positive and negative values of c, with graphical output in the form of a plane curve.

(d) Consider the system $\dot{y}_1 = y_2$; $\dot{y}_2 = y_1$. (This cannot be written as a scalar complex equation.) Show that the paths are (part of) hyperbolas of the form $y_1^2 - y_2^2 = c$. Take as initial points 16 equidistant points on the unit circle.

(e) Run the same system as in (b) a few cases with initial conditions of the form $y_1(0) = -1$, $y_2(0) = 1 - \delta$ and $y_1(0) = -1 + \delta$, $y_2(0) = 1$, where $0 < \delta \ll 1$. Take e.g. $\delta = 0.005$ and $\delta = 0.02$.

Look at the orbits on the screen. One of the asymptotes seems to be "attractive", and the other is "repulsive". Explain why.

Look at the numerical output. Make a conjecture about how the shortest distance from the origin depends on δ, $0 < \delta \ll 1$.

Prove the conjecture.

C4. Recall the shot problem, described in Sec. 1.4. We introduce dimensionless variables by means of scale factors inspired by the case without air resistance, which can be solved exactly. Set $w_0 = \sqrt{u_0^2 + v_0^2}$, $L = w_0^2/g$, $\epsilon = cR^2L/m$. The new variables are $\bar{x} = x/L$, $\bar{y} = y/L$, $\bar{u} = u/\sqrt{gL}$, $\psi = v/u$, $\bar{t} = t\sqrt{g/L}$. We take $\bar{x}$ as independent variable (instead of $\bar{t}$ that is now eliminated). Show that the differential system then becomes

$$\frac{d\bar{y}}{d\bar{x}} = \psi; \quad \frac{d\bar{u}}{d\bar{x}} = -\epsilon\bar{u}\sqrt{1+\psi^2}; \quad \frac{d\psi}{d\bar{x}} = -\bar{u}^{-2},$$

with initial conditions (say) $\bar{y}(0) = 0$, $\bar{u}_0 = \cos(\alpha_0)$, $\psi(0) = \tan(\alpha_0)$. (You may also add an equation for $d\bar{t}/d\bar{x}$.) Run the cases treated in Sec. 1.4.

This particular problem can be run efficiently with constant step size. Run it also with two different step sizes and make Richardson h^2- extrapolation. Use linear inverse interpolation to find the landing point. (Alternatively, do this with variable step size according to mode (iv) described in Exercise 1.)

COMMENT: The interpolation error depends of the location of the point within the step. A consequence is that the error of a value obtained by linear (inverse) interpolation may not show the same proportionality to TOL or h^2 as the errors at the points primarily computed by the integration routine. This can be overcome by the use of higher order (inverse) interpolation.

(b) Make also shots with $\alpha_0 = 0$ from an "infinitely" high mountain. Use

variable step size. Will $\bar{y}(x)$ tend to $-\infty$ for a finite value of x? Make a conjecture about this, and support it by computational and/or theoretical evidence. Would it be better to use another independent variable in this study? Try this too.

C5. Study experimentally (by numerical integrations) the sharpness of the bounds for $|\zeta(t) - \eta(t)|$ obtained in the pendulum example 13.1.24, for some different values of η_0. You may use constant step size in this study. Use several step sizes in order to be able to separate $|\zeta(t) - \eta(t)|$ that is small itself (if η_0^3 is small) from its truncation error.
COMMENT: If you estimate $\zeta(t) - \eta(t)$, from the difference between the results from separate numerical integrations for ζ and η it is advisable to have the same step size sequences in both integrations. (Why?) There are, however, several other ways to perform the study; you can, e.g., set up a differential equation for $\zeta(t) - \eta(t)$. It is your choice.

C6. Treat the system (13.1.36) that describes a population with crowding and toxins, with Runge's 2nd order method. Compare with the results of Problem P8. Start with $y_1(0) \ll \text{TOL}$, $y_2(0) = 0$.
Run the following cases long enough to show the limits as $t \to \infty$:
$a = b = 0.5$; $a = b = 0.1$; $a = 0.5$, $b = 0$; $a = 0.1$, $b = 0$.
Plot y_1 versus t on the same sheet for all the four cases.
Plot y_2 versus y_1 in the first two cases.

C7. (a) Run the Predator-Prey problem as in Example 13.1.11. Choose TOL (or a constant step size) so that the curves (see Fig. 13.1.6) become closed curves to the accuracy of your screen or hard copy output. Determine how the length of the periods depends on the initial value.

(b) Make a second run, and apply Richardson extrapolation to improve the estimates of the periods. (See the comment of exercise C4.)

(c) A modified Lotka–Volterra model reads

$$\dot{y}_1 = (a - \epsilon y_1 - b y_2) y_1, \quad \dot{y}_2 = (-c + d y_1) y_2.$$

Choose (say) $\epsilon = 0.2$, and run it with the same parameters and initial values as before. Note that the qualitative character of the orbits changes a lot.

C8. Treat Problem P18 with Runge's 2nd order method. You are likely to encounter some trouble with values of $y > 1$, due to the inevitable computational errors. How can you rewrite the problem in order to avoid this trouble (without using the known exact solution)?

C9. Dry friction. Test by numerical experiments the result of Problem P19 above, for some representative choices of γ and α.

C10. We saw in problem P8 that the solution of the differential equation $\dot{y} = y - 2t/y$, $y(0) = 1$ is very sensitive to perturbations in the initial value.
Study what happens in the numerical solution (with the correct initial value) up to $t = 12$ (say) with various tolerances that give decent computing times. Make also some experiments with $y(0) = \sqrt{1 - \delta}$, $\delta > 0$, and find out approx-

imately, how the length of the interval of existence of a real solution depends on δ . Design a suitable termination criterion. Plot y versus t with several tolerances on one sheet, in linear scales. Plot on another sheet $\log|\text{error}|$ versus t.

C11. The solution of the differential equation $\dot{y} = t^2 + y^2$, $y(0) = 1$ tends to infinity at $t = a$, where a is to be determined to (say) 3 or 4 decimal places. (See also Problem P10).

(a) Set $y = 1/u$, and solve the differential equation for u numerically. Interrupt when u has become negative, and determine a by inverse interpolation.

(b) For a complicated system, the kind of transformation suggested in (a) may be impractical, and we shall therefore see what can be done with a more direct approach to the original equation and an appropriate termination criterion.
One criterion of a general nature is to stop when $t_{n+1} = t_n$ in the computer. This criterion is worthwhile to have in the program, together with some suitable message, also for many other reasons. Determine how many steps it needs, for various tolerances. How well is a determined? How big is y when it stops?
Another idea is to use Aitken extrapolation of the sequence t_n. Stop when two successive extrapolated values differ by some fraction of TOL. (The fraction is to be tuned.) Determine how many steps it needs for various tolerances, and how well a is determined. Does the cancellation in the denominator of the Aitken formula cause trouble?

(c) Make some experiments, in order to see, if the two strategies described in (b) work for other equations of the form $\dot{y} = t^2 + y^c$, $y(0) = 1$, ($c > 1$, not necessarily an integer).

C12. The numerical solution of the differential equation $dy/dt = f(t)$ may be a practical way to perform numerical quadrature, due to the well developed techniques for the automatic control of stepsize in the packages for solving ODEs. Test your program, and some more professional package, on the computation of the slowly convergent integral

$$\int_0^\infty (t^3 + t^2 + 1)^{-1/2}\, dt,$$

to (say) 4 decimal places, and make a graph or a table that shows the variation with t of the step size and the number of steps. Is it true that the latter grows like $a + b \log t$, for $t \gg 1$?
Decide yourself to what extent one or two terms of an expansion like the one in Example 3.1.9 are useful for large t, e.g., in order to determine when to stop. How do you choose the tolerance and estimate the error? (See computer exercise 1 of this section, suggestion (iv).)

C13. (a) Run the following initial-value problem problem with the 2nd order Runge–Kutta method with a few tolerances,

$$\ddot{y} = 2y, \quad y(0) = 1, \quad \dot{y}(0) = -\sqrt{2}.$$

You know the analytical solution, but you cannot be sure that the numerical solution will in the long run behave in the same way, e.g., due to rounding errors. Design a termination criterion, so that the computation is interrupted with some informative text, if something evidently wrong happens, e.g., if $y(t) \leq 0$, or if $\dot{y}(t) \geq 0$, or if the number of steps exceeds 1000 (say).

(b) The problem $\ddot{u} = u^2$, $u(0) = 1$, $\dot{u}(0)$ arbitrary negative number, was treated in Ex.13.1.12. Run this over a longer time and for a few more values of $\dot{u}(0)$. Determine to 3 decimal digits (say), for a few of these initial values, when $u(t)$ becomes infinite. Make graphical output, both u versus t and in the y_1y_2-plane, with $y_1 = u$, $y_2 = \dot{u}$.

(c) Show that the graphs in th y_1y_2-plane have equations of the form $\frac{1}{2}y_2^2 = \frac{1}{3}y_1^3 + c$, where c, $c \geq -\frac{1}{3}$, depends on the initial values.

Show how to determine at what time $u(t)$ becomes infinite, by means of the values of integrals of the form $\int(\frac{1}{3}u^3 + c)^{-1/2})du$ between appropriate limits. Compute a few values numerically, and compare with the results obtained in (b).

(d) Do the same things for the problem $\dot{u} = u + u^2$, $u(0) = 1$, $\dot{u}(0) < 0$. Note the similarities and differences between this problem and the problem treated in (b).

C14. *Geodesics.* Consider the surface that is generated when a plane curve $[x(\alpha), z(\alpha)]$, $x(\alpha) > 0$, is rotated around the z-axis.

Our terminology stems from the spheroid which approximates the earth, but the extension to an arbitrary surface of revolution is hopefully obvious. For the spheroid

$$x = \cos\alpha, \quad z = (1 - \epsilon)\sin\alpha,$$

where the radius of the equator is the unit of length. The relation of the parameter α to the geographic latitude β reads $\tan\alpha = (1 - \epsilon)\tan\beta$. $\epsilon = 3.367E - 3$ for the earth, but you are advised to use a larger value at some of the numerical experiments in order to distinguish more easily the effects due to ϵ from the effects of truncation errors etc..

The longitude is denoted by t. A dot means differentiation w.r.t. to t, while a prime means differentiation w.r.t the parameter α (see the definition of the plane curve).

It can be shown that, except for the meridians,the geodesics on this surface of revolution satisfy the following differential equations:

$$\dot{s} = \frac{x}{\sin\theta}, \quad \dot{\alpha} = \frac{x\cot\theta}{u'}, \quad \dot{\theta} = -\frac{x'}{u'}, \tag{13.1.38}$$

where s is the arc length of the geodesic;

θ, $0 < \theta < \pi$ is the angle between the geodesic and the local southern direction;

u is the arc length from the north pole along a meridian, i.e., $du^2 = dx^2 + dz^2$.

More generally,

$$ds^2 = du^2 + (xdt)^2, \quad du = -ds\cos\theta, \quad xdt = ds\sin\theta. \tag{13.1.39}$$

(Note that $x > 0$, $\dot{s} > 0$.)
A suitable graphical output may be a curve with the polar coordinates $[t, r(\alpha(t))]$, where $r(\alpha) = x(\alpha)/(1 + z(\alpha)/k)$, where k is the distance from the south pole to the equator plane ($z = 0$); $k = 1 - \epsilon$ for the spheroid. This is the stereographic projection [17] of the geodesic to the equator plane, with the north pole as center. If $\epsilon = 0$ these curves are circles, both on the sphere and on the plane.
Determine from a given point a pencil of a few geodesics. Choose $\theta(0)$, $0 < \theta(0) < \pi$. It can be shown that $\sin\theta(t) = 0$, $\forall t$, and the geodesic stays away from the poles. If, for some t, $\sin\theta(t) = 0$ the geodesic follows a meridian for all t.
Test your program on the unit sphere ($\epsilon = 0$), with adaptive step size control. Then run the spheroid with the same step size sequence (!), and determine the difference with a few digits of *relative* accuracy. Judge the reliability in the usual way, i.e., by repeating these runs with all step sizes divided by 2 etc. Formulate some natural questions and other more quantitative questions related to them, and find, by numerical experiments, preliminary answers to them. For example
• Are the geodesics on a spheroid always closed curves, as they are on a sphere?
• Do all geodesics from one point meet again at some antipodal point, as they do on the sphere?
• Is it true that the *absolute* accuracy of the difference between two solutions, produced as described, is much better than the absolute accuracy of the original runs? How much better?
Save your program, for it will be extended in Sec. 13.7 to a boundary value problem for finding the shortest path on the surface between two given points. If you have time, try some theoretical analysis or some literature search, and try also some surfaces of revolution other than spheroids.
COMMENT: The first two equations of (13.1.38) follow from (13.1.39) and the other definitions. The third one comes from an elegant application of the Euler equations of the calculus of variations to

$$\int ds = \int \sqrt{(\dot{u})^2 + x^2}\, dt,$$

borrowed from de la Vallée–Poussin's classic text [art. 421][10]. Here u is the function to be determined, *x is a function of u*, and the result is simplified by means of the formulas in (13.1.39), and finally the equation is divided by $\sin\theta$. [18]

[17] If $\epsilon = 0$, the stereo-graphic mapping is conformal, and a circle is mapped onto a circle or a straight line.
[18] de la Vallée–Poussin studies a more general class of surfaces using so-called geodetic polar coordinates.

13.2 Control of Step Size and Numerical Stability

In applied mathematics, a theoretical study of a continuous model of a system with discrete structure, is often easier than a direct approach to the original discrete problem; the study of error propagation is one example of this. In Sec. 13.1.3 the error propagation in the numerical treatment of a single differential equation by a one-step method was treated by a direct discrete approach, although eventually a sum was approximated by an integral. The result is the approximate bound (13.1.10), illustrated in Fig. 13.1.4. Later the same type of bound was obtained in Theorem 13.1.22, see (13.1.33), for the change of the solution of a differential system caused by a continuous perturbation function $r(t, u)$, indicating that such a function can be used as a model for the sequence of error pulses. This is, however, not the whole truth. Unless certain restrictions are satisfied. the error propagation mechanism of the numerical computations can namely be rather different from this mechanism of the differential system itself, for which the (exact) variational differential equation tells the story.

For one-step methods we assume, to begin with, that

$$\|hf'(y)\|^2 \ll 1. \tag{13.2.1}$$

In Sec. 13.4 we shall derive the same *continuous model for error propagation* in a different way for the study of multistep methods. For them we have to add the requirement that they be strongly zero-stable (see Def. 13.2.3), a condition that is automatically satisfied by consistent one-step methods.

For some numerical methods condition (13.2.1) can be relaxed considerably and this opens the gate for the efficient application to an important class of differential systems called **stiff**. This encourages a further development in §13.2.1 of the continuous model to include other important features of practical computation. The results in Theorem 13.2.1 and Fig. 13.2.1 have the same simple structure as 13.1.10 and Fig. 13.1.4, but a more realistic interpretation.

An introduction to numerical stability is given in §13.2.2. Implicit and linearly implicit methods with applications to stiff and differential-algebraic systems, will be discussed in §13.2.3.

13.2.1 Scale Functions and Step Size Control

The adaptive control of step size is an important issue in the numerical treatment of ODEs. Unfortunately, in many cases, the assumptions used in the derivation of (13.1.10). lead to a rather inefficient step size sequence. We shall now modify the assumptions in a more useful way. Practical problems often contain the following two complications:

A. Time steps of very different orders of magnitude may be needed in different time intervals. We shall therefore introduce a function $\tau(t)$ that describes a **local time scale** of the motion. This is in the same spirit as many notions in other branches of applied mathematics, such as halving time, half width etc. Roughly speaking, the local time scale of a function should be the length of an interval around t, where the value of the function changes notably.

For example, the solution of the initial value problem

$$\dot{y} = -100(y - \cos t), \quad y(0) = 0,$$

reads $y(t) = (\cos t + 0.01 \sin t - e^{-100t})/1.0001$. At the start, $y(t) \approx 1 - e^{-100t}$; for (say) $t > 0.05$, $y(t) \approx \cos t$. It is natural to set $\tau(0) = 0.01$, and $\tau(t) \approx 1$ for $t > 0.05$. An ideal step size would vary proportionally to $\tau(t)$; we shall see below how the factor of proportionality depends on the tolerated error and on the numerical method.

In the theoretical discussion let, to begin with, $\tau(t)$ be any positive, piecewise continuous function. We then define a function $\xi(t)$, called the **age** of the motion, that measures small increments of time in the local time scale, i.e.,

$$\xi(t) = \int_a^t \frac{dx}{\tau(x)}. \tag{13.2.2}$$

B. The size of different components of $y(t)$ can differ by several orders of magnitude, and an individual component can vary by several orders of magnitude along an orbit. So we need another scale function too, **a diagonal scaling matrix** $S(t)$ for measuring errors etc.. The accuracy requirement may be in terms of relative accuracy for each component. Sometimes it can be rather expensive to obtain high relative accuracy for example near a point, where a component is zero. In some problems high relative accuracy may be necessary also in such cases, but in other problems it is not, and the computer should be able to take advantage of this. In some packages, the user can, for each component y_i, give a non-negative scale factor s_i (or accept a default value given by the program) that defines a breakpoint: relative accuracy is required when $|y_i| > s_i$ and absolute accuracy is required when $|y_i| \leq s_i$. The non-zero elements of the diagonal matrix $S(t)$ are defined thus,

$$S_i(t) = \max(|y_i(t)|, s_i), \qquad i = 1 : s. \tag{13.2.3}$$

The general strategy described here has been applied in several programs by one of the authors. (In the program outlined in §1.4.3, s_i was set to a default value 10^{-3} for all i.) Note that the requirement for either pure relative or pure absolute accuracy comes as particular cases, with appropriate choice of the factors s_i.

The exact formulation of all this may vary between implementations, which is also the case for several other details of this discussion. For example: a smoother alternative is to set $S_i(t) = |y_i(t)| + s_i$. See also the comments after Theorem 13.2.2.
[19]

If $u(t)$ is the absolute global error, then the **scaled global error** $v(t)$ (or the "mixed absolute-relative" error) is defined by the equation,

$$S(t)v(t) = u(t), \tag{13.2.4}$$

[19] Hairer, Nörsett and Wanner compare their error estimate for the component y_i with the quantity $|y_i|(Rtol)_i + (Atol)_i$. Their notion is more flexible than ours, which is obtained for $(Rtol)_i = \text{TOL}, \forall i$, $(Atol)_i = s_i \text{TOL}$.

and similarly for the local error vector. Let $l(t)$ be an estimate of the absolute local error vector in the time step that leads to t. Our **theoretical step control strategy** is to determine the step size $h(t)$ so that

$$\|S(t)^{-1}l(t)\|\tau(t)/h(t) \approx \text{TOL}, \tag{13.2.5}$$

where TOL is a tolerance to be given by the user; TOL *is thus compared with the scaled local error per the intrinsic unit of time.* The following theorem shows, how this leads to a connection of TOL to a bound for the *global* error, in fact a more natural or less obscure connection than most strategies for the step size control have.

Introduce the scaled error vector v and the age ξ into (13.1.29), i.e.,

$$u = Sv, \qquad dt = \tau d\xi, \qquad r(t) = l(t)/h(t),$$

$$\frac{d(Sv)}{d\xi} = \tau(JSv + l/h).$$

Since $d(Sv)/d\xi = Sdv/d\xi + (\tau dS/dt)v$, we obtain,

$$\frac{dv}{d\xi} = \tau\Big(S^{-1}JS - S^{-1}\dot{S}\Big)v + \frac{\tau}{h}S^{-1}l. \tag{13.2.6}$$

This equation has the same structure as equation (13.1.29). By (13.2.5), the norm of the last term is approximately equal to TOL or a little smaller. So, we have the following important consequence of Theorem 13.1.22:

Theorem 13.2.1.

Let $S(t)$ be the scale function defined by (13.2.3), and let the step size control be defined by (13.2.5), i.e. the error per local unit of time should be approximately equal to TOL. *The age ξ of the motion is defined by (13.2.2). Assume that in a neighborhood D of the orbit,*

$$\tau\mu\Big(S^{-1}f'(y)S - S^{-1}\dot{S}\Big) \le \mu^*. \tag{13.2.7}$$

Then, at age ξ, the norm of the scaled global error, $v(\xi)$, does not exceed $\psi(\xi)$, where

$$\frac{d\psi}{d\xi} = \mu^*\psi + \text{TOL}, \qquad \psi(0) = \|v(0)\| = \|S^{-1}u(0)\|. \tag{13.2.8}$$

Hence[20]

$$\|v(\xi)\| \le \|v(0)\|e^{\mu^*\xi} + \begin{cases} \text{TOL}\xi, & \text{if } \mu^* = 0; \\ \text{TOL}\frac{e^{\mu^*\xi}-1}{\mu^*}, & \text{if } \mu^* \neq 0;. \end{cases} \tag{13.2.9}$$

In particular, if $\mu^* < 0$, then $\|v(\xi)\| \le \psi(\xi) \to -\text{TOL}/\mu^*$, as $\xi \to \infty$. See Fig. 13.2.1.

The vector norm is arbitrary, and $\mu(\cdot)$ is the subordinate logarithmic norm.

[20]The bound (13.2.10) holds a fortiori if the step size is smaller than the value determined by (13.2.5). The formula for the number of steps, $N(t)$ is then an underestimate of the actual number of steps.

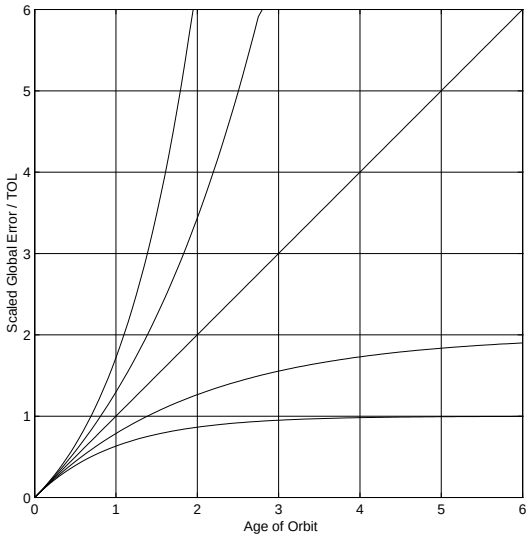

Figure 13.2.1. *Scaled global error divided by* TOL *versus age for* $\mu^* = \pm 1, \pm 0.5, 0$.

This theorem is valid for any positive continuous function $\tau(t)$. In fact, even an old step size strategy based on the local error per unit of (the ordinary) time, i.e., $\tau \equiv \tau(0)$, $\xi \equiv t/\tau(0)$, is included, in spite that $\tau(t)$ barely deserves to be called an *intrinsic* time scale of the motion in that case.

In another classical strategy the tolerance is compared with an estimate of the local error per step. This is not included; see, however, the comment after Theorem 13.2.2.

We shall mainly apply this to two classes of local time scales, $\tau = \tau_q^I$ and $\tau = \tau_q^{II}$, specified below.

If needed, the result (13.2.9) can be applied with piece-wise constant μ^* with the age measured from the most recent breakpoint. So we have the same situation as in Fig. 13.1.4 and (13.1.10), but it allows more general interpretations:

1. The errors are scaled by a diagonal matrix $S(t)$, which allows also "mixed absolute-relative errors".

2. The independent variable is the age ξ instead of the ordinary time t.

3. μ^* is defined by (13.2.7). It is usually more reasonable to assume a uniformly valid bound for τJ than for J itself, provided that the local time scale has been adequately defined.

Although the relation between local and global errors is very complicated, this theorem can at least give you a hint how to choose TOL, in order to meet your requirements concerning the *global* error. You cannot expect it to give you a precise

numerical bound for the global error in practice; mostly no precise estimates of the logarithmic norm, τ and ξ are easily available. The important thing is that for a qualitative picture and for the correct orders of magnitude it is enough to have rough ideas of τ and μ^*. In simple cases, it is indeed possible to carry out a more detailed analysis in these terms, see Example 13.2.1 and a few other examples, and this can reveal unexpected features that may influence your view of the numerical treatment of initial value problems and your language for the discussion of these matters.

For the practical estimation of the global error we suggest instead that you first run the problem with adaptive control, but you may choose TOL 2^p *times larger than you need, or a little less, where* p *is the order of accuracy, which is supposed to be constant. Afterwards you make another run, where each step of the previous run is divided into two equally large steps.* [21]

So, in the second run the step size is still variable though not adaptively controlled. You can then *either* just plot an error curve for the second run, by means of the difference between the results of the two runs, divided by $2^p - 1$; *or* you can make a *global Richardson extrapolation*, see §3.3.4 and Example 13.2.1. In the latter case you obtain better results, but it is sometimes not easy to obtain error estimates that are both reliable and realistic.

This procedure is, for several reasons, probably better than making two adaptive runs with different tolerances, if your program allows this mode of operation. The efficient implementation of this idea may be more complicated for programs where the order is also adaptively controlled.

The local error $l(t)$ is asymptotically proportional to h^{p+1}, $h \to 0$. We write

$$\|S^{-1}l(t)\| \sim |c_p|L(t; f, y)h^{p+1}. \tag{13.2.10}$$

We shall emphasize what happens in a particular motion and write shorter $L(t)$ or L instead of $L(t; f, y)$. The **error constant** c_p is introduced as a kind of calibration. *c_p is defined by the requirement that, in the case of the standard test equation $\dot{y} = \lambda y$, the local error should be asymptotically $c_p(\lambda h)^{p+1}y$.*[22]

Next we define

$$L^*(t) = |c_p|L(t)\tau_q(t); \tag{13.2.11}$$

τ_q here stands for either of two local time scales defined below, denoted $\tau = \tau_q^I$, $\tau = \tau_q^{II}$. With this notation, the theoretical step control strategy (13.2.5) reads,

$$L^*(t)h^p \approx \text{TOL}. \tag{13.2.12}$$

Let h_1 be the step size that has just been used. *If* $L^*h_1^p >$ TOL, the step is to be recomputed with step size

$$\bar{h} = h_1 \cdot (L^*h_1^p/\text{TOL})^{-1/p}/\beta, \tag{13.2.13}$$

[21] Compare the suggestions for the program in exercise C1 of Sec, 13.1.

[22] We have to exclude some methods, for which the order of accuracy is higher than p for this particular differential equation; we believe that this exclusion is harmless.

where β is a safety factor, e.g., $\beta = 1.1$, *else* the step is accepted, and the step size $h = h_1 \min(\bar{h}/h_1, \alpha)$ is suggested for the next step, where α, $\alpha > 1$ is for preventing a temporary small value of $L^*(t)$ from having an undesirable effect on the step size.

The safety factors α, β are to be tuned by the implementer.[23] They account for some approximations used in the error estimate and, above all, for the fact that the value of $L^*(t)$ is based on information from the past, while it is to be used for a step into the near future. In the program outlined in Sec.1.3 we chose $\alpha = 5$. In that program there is also a bound for the suggested reduction of the step size for a recomputation; if necessary the recomputation can be repeated. Many programs require upper and lower bounds for h itself etc.

The results are still valid for any positive function $\tau(t)$. Now we shall specify the notion of local time scale more, and see that it can be quite easy to apply. Assume that we work with a numerical method of order p. We distinguish between two different types.

Type I: Suppose that the local error in a step is approximately,

$$l(t) = c_p h^{p+1} y^{(p+1)}(t), \quad \text{i.e.,} \quad L(t) = \|S^{-1} y^{(p+1)}(t)\|. \tag{13.2.14}$$

This is the case for linear multistep methods and several other classes of methods.

There are then, in this Type, two alternative natural definitions of $\tau(t)$; q stands for *quotient*, and r stands for *root*.

$$\tau_q^I(t) = \|S^{-1} y^{(p)}(t)\| / \|S^{-1} y^{(p+1)}(t)\|, \quad \tau_r^I(t) = \|S^{-1} y^{(p)}(t)\|^{-1/p}. \tag{13.2.15}$$

For the sake of brevity we suppress in the notation the dependence of p. The "dimension" of both measures is "time". For example, if $y(t) = e^{\lambda t}$ then $\tau_q^I(t) = \tau_r^I(t) = |\lambda|^{-1}$; if $y(t) = t^{-m}, \tau_q^I(t)$ and $\tau_r^I(t)$ are both proportional to t, see Example 13.2.3.[24]

By (13.2.11) and (13.2.15),

$$L^*(t) \approx \|c_p S^{-1} y^{(p)}(t)\|. \tag{13.2.16}$$

Note that $y^{(p)}(t)$ appears here, *not* $y^{(p+1)}(t)$. The step size control of Type I is now defined by (13.2.12).

By (13.2.15), we now have $\tau_r^I(t) \approx |L^*(t)/c_p|^{-1/p}$, hence by (13.2.12),

$$\tau_r^I(t) \approx h|\mathrm{TOL}/c_p|^{-1/p}. \tag{13.2.17}$$

Type II: The situation is reversed here; suppose that a *criterion for the step size control is given* in the form $L^*(t)h^p \approx \mathrm{TOL}$. Recall that the norm of the scaled local error in a step is $|c_p| L(t) h^{p+1}$. Here L, L^* usually do *not* satisfy (13.2.14), (13.2.16). (This is the situation, e.g., for embedded Runge–Kutta methods with local extrapolation, see §13.3.2.)

[23] The use of β means de facto that TOL is divided by β^p.

[24] The dimension of L and L^* are (time^{-p-1}) and (time^{-p}), respectively. The dimension of all variants of τ is time; the age, μ^*, c_p, TOL, and the scaled global error are non-dimensional.

The local units of time $\tau_q^{II}(t)$, $\tau_r^{II}(t)$ are not needed for the step size control, but we shall see how they can be defined in this case, in order to make applications of the theorems of this section possible, e.g., for the a priori estimation of global error, the number of steps etc. See also Ex. 13.2.1.

We *define* $\tau(t) = \tau_q^{II}(t)$ by requiring that (13.2.5) should hold, i.e.,

$$|c_p| L h^{p+1} \tau_q^{II}/h = \text{TOL} = L^* h^p, \tag{13.2.18}$$

hence

$$\tau_q^{II} = |L^*/c_p|/L. \tag{13.2.19}$$

We similarly *define* τ_r^{II} by the equation

$$\tau_r^{II} = |L^*/c_p|^{-1/p}. \tag{13.2.20}$$

Recall that the last two equations are (approximately) valid in Type I too. Since $\text{TOL} \approx L^* h^p$ we obtain, just like in Type I,

$$h/\tau_r^{II} \approx |\text{TOL}/c_p|^{1/p}. \tag{13.2.21}$$

The functions τ_q^{II}, τ_r^{II} thus defined are certainly "local" time scales, but sometimes they may not deserve the name "intrinsic" time scales.

We are now in a position to formulate a useful result that is valid in both types.

Theorem 13.2.2.

Set $\tau(t)$ equal to $\tau_q^I(t)$ or $\tau_q^{II}(t)$ in the step size control and in the previous theorem. Put

$$k = |\text{TOL}/c_p|^{1/p}.$$

Set τ_r equal to τ_r^I, τ_r^{II} in the two types. Introduce the age function $\xi_r(t) = \int_a^t dx/\tau_r(x)$, and let $N(t)$ be the number of steps needed for the interval $[a, t]$. Then

$$h(t) \approx \tau_r(t) k, \qquad N(t) \approx \xi_r(t)/k.$$

Proof. The first relation was derived in (13.2.17) for Type I, and in (13.2.21) for Type II. We use the first relation to establish the second relation:

$$N(t) = \sum_{x_i \le t} \frac{h(x_i)}{h(x_i)} \approx \int_a^t \frac{dx}{\tau_r(x) k} = \frac{\xi_r(t)}{k}.$$

□

Comments: (i) If ξ_r were chosen as the independent variable in the ODEs then both Type I and Type II would yield the constant (non-dimensional) step size k. This variable transformation is not to be used analytically in practice, but it

can help conceptually; in fact it can be said that the step size control recursively achieves this transformation.

Typical values of k are between 0.01 and 0.1 times the number of function evaluations in a step, if p is adequately chosen for the accuracy required; a higher value of p when high accuracy is required. (ii) In the 1960's there were animated debates whether to compare the tolerance with the local error per step or with the local error per unit of time. Our strategy gives a kind of Salomonic answer: we use the local error per unit of time in the intrinsic local time scale, $\tau_q(t)$. Formally this sounds like accepting the second opinion, but one can show (Problem P3) that, for both types of step size control,

$$\|S^{-1}l(t)\| \approx k \text{ TOL } \tau_r(t)/\tau_q(t), \tag{13.2.22}$$

where k is defined in Theorem 13.2.2. So, if $\tau_r(t)/\tau_q(t)$ is constant during a motion, *this strategy gives the scaled local error per step a constant norm along an orbit.* That sounds more like accepting the first opinion, but *the tolerance does not mean the same.* Note that TOL is changed, the *global* error and the number of steps will be changed proportionally to TOL and $\text{TOL}^{1/p}$, respectively.

If we ask for a bound for the relative error, i.e. if $|y_i| > s_i$. $\forall i$, then it can be shown (problem P4) that, in Type I, $\tau_r(t)/\tau_q(t)$ is indeed constant, if $y(t)$ is an exponential or a power (but this is usually not exactly the case).

Typically, $\tau_r(t)/\tau_q(t)$ fluctuates between rather moderate bounds, and these measures of local time scale usually do not depend heavily on p either if (say) $2 \le p \le 8$. These notions have no high precision, but they are useful for decision. The notions of age and local timescale describe features of a motion that are related to the amount of work and the step size needed in the numerical simulation, respectively. The role of the tolerance is concentrated in the constant k, and so is, to a large extent, also the dependence of the numerical method.

(This statement is a little exaggerated, for the definition of $\tau_r(t)$ does somewhat depend on the numerical method.)

Example 13.2.3 RUNGE'S 2ND ORDER METHOD STUDIED FROM THE POINT OF VIEW OF TYPE II:

Recall Runge's 2nd order method for the autonomous system $\dot{y} = f(y)$,

$$k_1 = h_n f(y_n); \quad k_2 = h_n f(y_n + \tfrac{1}{2}k_1); \quad y_{n+1} = y_n + k_2.$$

The local error $l(t_n)$ reads

$$y_{n+1} - y(t_n + h_n; t_n, y_n) = \left(\frac{1}{8}f''f^2 - \frac{1}{6}y'''\right)h_n^3 + \dots,$$

see Problem 7 of Appendix. For the test equation $\dot{y} = \lambda y$ this becomes $-\frac{1}{6}(\lambda h)^3 y$. Then, by the definition of the error constant and by Theorem 13.2.2, $p = 2$, $|c_p| = \frac{1}{6}$, $k = \sqrt{6\text{TOL}}$. Hence, by (**??**), $L(t) = \|\frac{6}{8}S^{-1}f''f^2 - S^{-1}y'''\|$.

In the program outlined in Sec. 1.3.4 the step size is controlled by means of the quantity

$$\frac{1}{3}\|k_2 - k_1\| = \frac{1}{3}h\|f(y_n + \tfrac{1}{2}hf) - f(y_n)\| = \|\frac{1}{6}h^2 f'f + \dots\| \approx \frac{1}{6}h^2\|\ddot{y}\|.$$

In the terminology of Type II, we therefore set $L^* = \frac{1}{6}\|\ddot{y}\|$.

By (13.2.19), (13.2.20), (13.2.21), we then obtain,

$$\tau_q^{II} = \frac{\|S^{-1}\ddot{y}\|}{\|S^{-1}(y''' - \frac{3}{4}f''f^2)\|}, \quad \tau_r^{II} = \|S^{-1}\ddot{y}\|^{-1/2}, \quad h = k\tau_r^{II}.$$

In our experience, τ_q^{II} deserves to be called an intrinsic time scale unless the problem is strongly non-linear, e.g., if (say) $\|f''f^2\| > \|y'''\|$.

Now consider the *application to the single equation* $\dot{y} = -\frac{1}{2}y^3$, $y(1) = 1$, with the solution $y(t) = t^{-1/2}$, with relative error control, i.e., set $S = y$. Now

$$\tau_q^{II} = \frac{4}{7}t; \quad \xi_q^{II} = \frac{7}{4}\int_1^t \frac{dx}{x} = \frac{7}{4}\ln t; \quad \tau_r^{II} = \frac{2}{\sqrt{3}}t; \quad \xi_r^{II} = \frac{\sqrt{3}}{2}\ln t;$$

$$h(t) = \sqrt{6\text{TOL}}\tau_r^{II} = \sqrt{8\text{TOL}}t; \quad N(t) = \frac{\ln t}{\sqrt{8\text{TOL}}}; \quad f''f^2/y''' = 0.4.$$

Moreover, $\mu(S^{-1}JS - S^{-1}\dot{S}) = f'(y(t)) - \dot{y}(t)/y(t) = -y^2 = -1/t$ hence, by (13.2.7), $\mu^* = -\tau_q^{II}/t = -4/7 = -0.57$. This example confirms the opinion expressed above that it is usually more reasonable to use a bound for (the non-dimensional quantity) $\tau(t)\mu(S^{-1}JS - S^{-1}\dot{S})$—as we have done in Theorem 13.2.1—than to use a bound for $\mu(S^{-1}JS - S^{-1}\dot{S}) = -1/t$ itself.

By Theorem 13.2.1 or Fig. 13.2.1, $v(\xi)$, i.e., the estimated norm of the *relative* global error, grows like $\xi_q^{II}\text{TOL}$, to begin with, (if there is no initial error), and then it converges towards $\frac{7}{4}\text{TOL}$. Hence the *absolute error bound tends to zero like* $\frac{7}{4}\text{TOL}t^{-1/2}$.

With constant step size $h = h(1)$, the number of steps needed are to reach t is hence $(t-1)/\ln t$ times as large as with our theoretical step control strategy.

This case was run for three different values of TOL, with a program, designed (essentially) according to §1.3.4. The line named "theor.", is calculated according to formulas above.

TOL	rel.err/TOL	$N(10^4)$	$N(10^4)\sqrt{\text{TOL}}$
$4\,10^{-4}$	1.36	192	3.84
10^{-4}	1.40	372	3.72
$\frac{1}{4}\,10^{-4}$	1.37	744	3.72
theor.	1.75		3.26

The steps are about 15% shorter than predicted, and the relative error is also a little smaller than predicted. The main reason is that the safety factor $\beta = 1.2$ used in the program, is not needed in a problem like this, where the trajectory becomes smoother as time goes by.

Global Richardson extrapolation, see §3.3.4, was also applied to this example, i.e. the first run used adaptive step size control. In the second run, each step of the first run was replaced by a pair of steps of half size. Denote the results of these two runs by $y(t;N)$, $y(t;2N)$; the extrapolated values are $y(t;Ex) = y(t;2N) - \frac{1}{3}\big(y(t;2N) - y(t;N)\big)$. In the long run $\text{Error}(y(t;Ex)) \approx 0.5(\text{TOL})^{1/2}\text{Error}(y(t;2N))$. This was tested with the use of $\text{TOL} = 0.001$ and $\text{TOL} = 0.0001$ in the first runs. The relative errors of $y(t;Ex)$ became about $0.25\,10^{-4}$ and $0.75\,10^{-6}$, respectively.

The rule of thumb for the range of tolerances mentioned above becomes for Runge's 2nd order method (without extrapolation) $0.02 < k \approx \sqrt{6\text{TOL}} < 0.2$, i.e., $7\,10^{-5} < \text{TOL} < 7\,10^{-3}$. For problems of moderate size the method can well be used for smaller tolerances, if no higher order method is conveniently available.

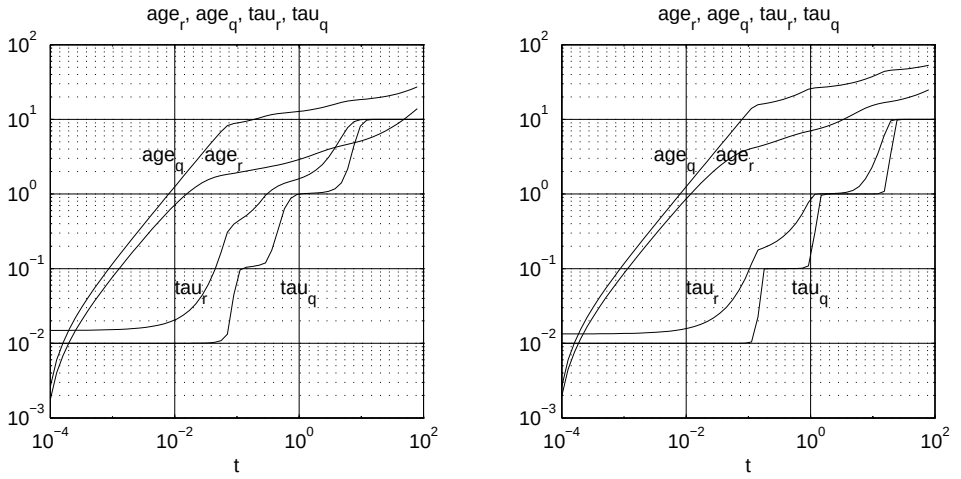

Figure 13.2.2. *The functions $age_q = \xi_q$, $age_r = \xi_r$, τ_r, τ_q for $p = 2$, and for $p = 5$. See Example 13.2.4.*

Example 13.2.4 (*Study of a stiff linear system with the use of the Type I concepts)* Fig. 13.2.2 shows, for $p = 2$ and $p = 5$, the functions ξ_q^I, ξ_r^I, τ_r^I, τ_q^I for a linear system $\dot{y} = Ay$. Here, $A = T\Lambda T^{-1}$, where $\Lambda = \text{diag}(-100, -10, -1, -0.1)$, and T is a random 4×4 matrix, i.e. the elements of T are independent random numbers in the interval $[0, 1]$, and $y(0)$ is a random vector. We choose $S(t))y(t)$.

The figure was obtained by computer with the use of the exact solution of the system. Note the sharp plateaus, when $\tau_q^I(t)$ has reached the reciprocals of the eigenvalues of $-A$, also called the **time constants of the system**. This happens, approximately at $t = 0$, 0.1, 1, 10, a little earlier for $p = 2$ than for $p = 5$.

The curves for $\tau_r(t)$ show a smoother transition between the levels. This is quite natural, since $\tau_r(t)$ is the geometric mean of the first p functions $\tau_q(t)$ when $S(t) = y(t)$.

Another interesting feature of this example is that *the local time scale of the motion is much larger than the smallest time constant of the system, when t is*

large. They become, respectively, 10 and 0.01. When this happens one says that *the motion has become stiff*. A *problem* is said to be stiff, if there is a time interval, where the motion is stiff.

The increase of the local time scale is explained by the following decomposition of the solution $y(t)$. Here t_j is a column of the matrix T, i.e. the eigenvector of A belonging to the eigenvalue λ_j, and γ_j is a component of the vector $T^{-1}y(0)$, $j = 1, 2, \ldots, s$, $(s = 4)$. By (13.1.20) and (13.1.21), $y(t) = \exp(At)y(0) = T\exp(\Lambda t)T^{-1}y(0)$, hence

$$y(t) = \sum_{j=1}^{s} \gamma_j t_j \exp(\lambda_j t) \tag{13.2.23}$$

The term with $\lambda_j = -100$ dies out at about $t = 0.05$ (say). Similarly, the term with $\lambda_j = -10$ dies out at about $t = 0.5$, etc. Roughly speaking, the local time scale is determined by the fastest of those terms which have not yet died out. □

The decomposition (13.2.23) is *valid and interesting for general diagonalizable matrices*, and our discussion can be extended also to ODEs with complex eigenvalues. In that case, either the real part or the modulus of an eigenvalue is relevant, depending on the context.

A linear system with constant coefficients was considered in the example. The concepts introduced are most relevant in discussions of what happens to small disturbances (errors). Therefore the natural generalization of the concept of time constants to a non-linear problem is related to the variational equation (13.1.8). Natural candidates to the name of time constants at a given time are then *the reciprocals of the eigenvalues of the Jacobian*. The smallest time constant can often be estimated by the reciprocal of some norm of the Jacobian, which is often time-dependent. So, we must talk about **local time constants** in the definition of *stiffness in a non-linear problem.*[25]

Stiff problems is an important class of problems, that requires special numerical methods; else the step size control will make the time step much shorter than the one predicted by Theorem 13.2.2. A stiff motion is namely surrounded by other possible motions of the same system, for which the local time scale is more like the *smallest* time constant. The numerical method must have good stability properties, in order that the motion should not be kicked out to one of those tracks, where a much smaller step size is required. We shall return to this question.

Example 13.2.5 T*A study of a family of nonlinear ODEs in Type I*:
differential equation,

$$\dot{y} = -my^{1+1/m}, \qquad y(1) = 1, \qquad (m > 0),$$

has the solution, $y(t) = t^{-m}$. Note that

$$(-1)^p y^{(p)}(t) = m(m+1)(m+2)...(m+p-1)t^{-m-p}, \tag{13.2.24}$$

[25]This is a useful generalization, although the usage of this terminology must be taken with a grain of salt, unless the time scale of the variation of the local time constants is much larger than the time constants themselves.

so that

$$\tau_q^I(t) = t/b_q, \qquad b_q = p + m,$$
$$\tau_r^I(t) = t/b_r, \qquad b_r = \Big(m(m+1)(m+2)...(m+p-1)\Big)^{1/p}.$$

Hence $\xi(t) = b \ln t$, with $b = b_q$ or $b = b_r$. In this example, $\tau_r^I(t)/\tau_q^I(t)$ is constant, so the scaled local error will be constant during the motion. (Actually, $2 < b_q/b_r < 3$, for $0.5 < m < 1$.) By Theorem 13.2.2,

$$h(t) \approx kt/b_r, \qquad N(t) \approx b_r \ln(t)/k, \quad k = |\text{TOL}/c_p|^{1/p}.$$

Notice that the step size grows proportionally to time. The values of p, c_p and k depend on the particular numerical method. We have

$$\mu(J(t)) = f'(y(t)) = -(m+1)y^{1/m} = -(m+1)/t.$$

We require relative accuracy, i.e. $S(t) = y(t) = t^{-m}$. Since $S^{-1}JS = J$, we then obtain,

$$J - S^{-1}\dot{S} = -\frac{m+1}{t} + \frac{m}{t} = -\frac{1}{t}.$$

Note the importance of the term $S^{-1}\dot{S}$. Then, by (13.2.7), we can choose $\mu^* = \max(-\tau_q^I(t)/t) = -1/b_q$. By Theorem 13.2.1, or Fig. 13.2.1, the estimated norm of the *relative* global error, $v(\xi)$, will grow like ξTOL, to begin with, (if there is no initial error), and then it converges towards b_qTOL. Hence the *absolute* error bound tends to zero like $b_q t^{-m}$TOL.

With constant step size $h = h(1)$ the number of steps needed are to reach t is hence $(t-1)/\ln(t)$ times as large as with our theoretical step control strategy. (For example, for $t = 10^4$, we have $(t-1)/\ln(t) \approx 10^3$.) Note that this conclusion is independent of m, p, c_p and TOL. The same conclusion was obtained (for $m = \frac{1}{2}$) in Example 13.2.3 with a Type II strategy.

Note that the age, and hence the number of steps, grows proportionally to $\log t$. This is typical for a large class of problems where the motion approaches the origin like some negative power of t.

13.2.2 Introduction to Numerical Stability

So far, we have discussed a theoretical strategy for the step size control. With a few modifications, this has been used in actual programs, and the actual step size sequences are often close to those predicted by theoretical calculations like the above (when they can be done). Some of the differences between this theory and practice are as follows:

A. Most programs put *restrictions on the time step changes.* (A simple case is seen in the program for Runge's 2nd order method outlined in Sec. 1.3.) There are several reasons. It was mentioned above that a rapid change of the

suggested step size may be due to a weakness of the error estimate. Another reason is that the change of the step size causes some extra computation ("overhead costs"), so it may sometimes, in particular for implicit methods, be better to keep it unchanged for a while than to increase it by only a few per cent.

B. When $\|hf'(y)\|^2$ is not small, the error propagation in the numerical algorithm can be very different from the error propagation in the differential system. We shall now discuss what conditions a numerical method and the step size produce should satisfy, in order that our theory should provide an adequate step size control strategy.

The discussion will, for example, yield a condition for the theoretical step size control to be applicable to Euler's method, but it also shows that this method (like most explicit methods) is unable to handle stiff motions. Moreover, some numerical methods must be discarded altogether, although they look rather promising if one only consider their local errors.

We first need a discrete analog to some of our previous results, e.g. (13.1.10) and Theorems 13.1.23 and **??**.

Lemma 13.2.6. *Consider a sequence of non-negative numbers* $w_0,\ w_1,\ w_2, \ldots$ *that satisfy the inequality,* $w_{n+1} \le a_n w_n + b_n,\ n \ge 0$, *where* $0 \le a_n \le a,\ 0 \le b_n \le b$.

Then, $w_n \le \psi_n$, *where* ψ_n *is a solution of a scalar difference equation,* $\psi_{n+1} = a\psi_n + b, \quad \psi_0 = w_0$, *i.e.*

$$w_n \le \psi_n = \begin{cases} a^n w_0 + \dfrac{b(1-a^n)}{1-a}, & \text{if } a \ne 1; \\ w_0 + bn, & \text{if } a = 1. \end{cases} \tag{13.2.25}$$

The behaviour of the sequence ψ_n *is illustrated by Fig. 13.2.1. If* $a < 1$, $w_n \le \max\{w_0, b/(1-a)\}$, $\forall n \ge 0$, $\lim w_n \le b/(1-a)$.

If $a = 1 + O(k),\ k \to 0,\ \xi$ *fixed, then* $w_n = O(b/k) + O(w_0)$.

Proof. The proof is left for problem P5. □

Here, $k = |\text{TOL}/c_p|^{1/p}$, as defined in Theorem 13.2.2. In typical applications w_n is the norm of a scaled global error, and $a_n = 1 + \mu_n h_n + O(k^2),\ a = 1 + \mu^* k \approx e^{\mu^* k}$. In particular, $a = 1$ corresponds to $\mu^* = 0$. Also note that $nk \approx \xi_n$, by Theorem 13.2.2, so that $a^n \approx e^{\mu^* \xi}$.

The following two examples are typical and lead to interesting conclusions.

Example 13.2.7 *Nonlinear stability analysis for Euler's method.*

Let $y(t)$ be an exact solution of the autonomous differential *system* $\dot{y} = f(y)$. Euler's method, with variable step size, yields the difference equation,

$$y_{n+1} = y_n + h_n f(y_n), \quad y_0 = y(0),$$

while $y(t)$ satisfies the difference equation,

$$y(t_{n+1}) = y(t_n) + h_n f(y(t_n)) + l(t_{n+1}).$$

where the local truncation error $l(t_{n+1})$ is obtained by Taylor's formula with the integral formula for the remainder (12.X.4). For $t = t_n$, $h = h_n$,

$$l(t+h) = y(t+h) - y(t) - h\dot{y}(t) = h^2 \int_0^1 (1-x)\ddot{y}(t+hx)dx \sim \tfrac{1}{2}h^2\ddot{y}(t).$$

We use similar notations as before in this subsection, in particular Theorem 13.2.1 and Lemma 13.2.6. In order to make the treatment more transparent, we here consider absolute error control only, i.e. $S(t) \equiv I$ and use a *weighted max-norm* only.[26] Set

$$u_\nu = y_\nu - y(t_\nu), \quad w_\nu = \|u_\nu\|,$$

and recall that, by (13.1.14), $f(y_n) - f(y(t_n)) = J(t_n, u_n)u_n$, where $J(t,u) = \int_0^1 f'(y(t) + \theta u)d\theta$, hence

$$u_{n+1} = u_n + h_n J(t_n, u_n)u_n - l(t_{n+1}). \tag{13.2.26}$$

Hence,

$$\|u_{n+1}\| = \|I + h_n J(t_n, u_n)\|\|u_n\| + \|l(t_{n+1})\|.$$

Assume that (i) the diagonal elements of $h_n f'(y)$ are greater than -1 for all y in a neighborhood of $y(t_n)$, (so that Theorem 13.1.20 can be applied.)

(ii) one of our theoretical step control strategies (type 1 or 2) is used, (so that, by Theorem 13.2.2, $k = 2\text{TOL}$.

By Theorem 13.1.20, (13.2.7) and Theorem 13.2.2,

$$\|I + h_n J(t_n, u_n)\| \le 1 + \mu h_n J(t_n, u_n) \le 1 + k\mu^* \equiv a, \tag{13.2.27}$$

By (13.2.5) and Theorem 13.2.2,

$$\|l(t_{n+1})\| \approx \text{TOL}h_n/\tau_n \approx k\text{TOL} \equiv b.$$

By Lemma 13.2.6, and the discussion after the lemma, we obtain $nk = \xi$, $a^n \approx e^{\mu^*\xi}$, and the same bound for $\|u_n\|$ as was obtained for $\|v(\xi)\|$ in Theorem 13.2.1.

These assumptions are satisfied over a given finite time interval $[t_0, T]$ if TOL is small enough; it follows from the bounds that *the global error is* $O(\text{TOL})$, hence the solutions provided by means of our theoretical step size control strategies *converge uniformly* to $y(t)$ as $\text{TOL} \to 0$.

In practical computation it is important to realize that for a fixed value of TOL the step size control may eventually lead to step sizes that violate the first assumption, and *afterwards* the continuous model for error propagation cannot be trusted for Euler's method. This trouble characterizes stiff motions, when one uses a method (like Euler's method) that is *not* designed to handle such motions.

Since the assumptions are sufficient conditions only, one might hope that the reality is not that bad. The linear system treated in 13.2.9 tells, however, that the reality is indeed almost that bad; Euler's method can still be used, but the step size

[26] This proof applies to the case of a constant diagonal matrix S. More general results will be derived in Sec. 13.9.

must satisfy a condition that that is close to the first of our assumptions, and that is no good if the motion is stiff.

We return to non-stiff problems, and make the two above assumptions, and set $S(t) \equiv 1$. In the discussion around Theorem 13.2.1 we promised more than a bound for the error; we held out the prospect that the true global error u_n of the numerical method should be close to $u(t_n)$, i.e., the total effect at t_n of all the previously committed local errors (or perturbations) in a continuous model of the error propagation.

We shall therefore estimate $g_n = u_n - u(t_n)$; $u(t)$ is defined by the non-linear variational equation, $\dot{u} = J(t, u(t))u - l(t + h(t))/h(t)$, hence

$$u(t_{n+1}) = u(t_n) + h_n J(t_n, u(t_n))u(t_n) - l(t_{n+1}) + \tfrac{1}{2}h_n^2\ddot{u}(t_n) + \dots.$$

Subtract this from (13.2.26):

$$g_{n+1} = g_n + h_n J(t_n, u(t_n))g_n - \tfrac{1}{2}h_n^2\ddot{u}(t_n) + \ldots + h_n(J(t_n, u_n) - J(t_n, u(t_n)))u_n.$$

We *assume* that $\partial J(t,u)\partial u$ is bounded, uniformly in t. Since $u(t) = O(\text{TOL}) = O(k),\ k \to 0,\ t$ fixed, and the same holds for $\dot{u}(t), \ddot{u}(t), u_n, h_n$ etc., we obtain, by (13.2.27 the inequality, $\|g_{n+1}\| \leq\ (1 + k\mu^*) + O(k^2))\|g_n\| + O(k^3)$, and hence, by Lemma 13.2.6,

$$g_n = O(k^2) = O(\text{TOL}^2).$$

(while $u_n = O(\text{TOL})$. This shows that, for our step size control, the true error agrees well with the result of the continuous model for Euler's method, as promised above.

Example 13.2.8 *Nonlinear stability analysis for the implicit Euler method.*
We still consider the nonlinear *system* $\dot{y} = f(y)$, and our theoretical step size control strategy.

The **implicit Euler method**, or the *backward Euler method*, is defined by the formula $y_{n+1} = y_n + hf(y_{n+1})$. Since y_{n+1} occurs also on the right hand side, a system of equations is generally to be solved in every time step, typically by some iterative method, e.g., by some modified Newton method. In this example we assume that the function f is such that the relevant solution can be accurately obtained with no trouble. We shall return to this question for more general implicit methods in Sec.13.2.4.

We shall see that *the implicit Euler method can handle stiff problems*, because of its impressive stability properties. On the other hand, it is only 1st order accurate (like the usual explicit Euler method), and the implicitness causes more computation in each step, in particular for large systems,

The solution $y(t)$ satisfies the difference equation,

$$y(t_{n+1}) = y(t_n) + hf(y(t_{n+1})) + l(t_{n+1}),$$

where the local truncation error $l(t_{n+1}) \sim -\frac{1}{2}h^2\ddot{y}(t)$ The notations and derivations are almost the same way as for the explicit Euler method, but the conclusions are very different. Another difference is that we can here use any norm, (while the discussion in the previous example was restricted to weighted maximum norms).

The difference equation for $u_\nu = y_\nu - y(t_\nu)$ now reads,

$$u_{n+1} = u_n + h_n J(t_{n+1}, u_{n+1})u_{n+1} - l(t_{n+1}),$$

hence

$$u_{n+1} = \big(I - hJ(t_{n+1}, u_{n+1})\big)^{-1}\big(u_n - l(t_{n+1})\big). \tag{13.2.28}$$

Recall that, by problem $P12$ of Sec.13.1, $\|(I - hJ)^{-1}\| \le (1 - h\mu(J))^{-1}$ if $h\mu(J) < 1$ for *any norm*. Assume that (**??**) holds with $S(t) \equiv I$. Then $\|(I - hJ(\ldots))^{-1}\| \le (1 - k\mu^*)^{-1}$ if $k\mu^* < 1$. The latter restriction is harmless compared to the analogous restriction needed for the explicit method.

By (13.2.5) and Theorem 13.2.2, $l(t) \approx \text{TOL}h(t)/\tau(t) = \frac{1}{2}k^2$. For $w_\nu = \|u_\nu\|$ we then obtain the inequality $w_{n+1} \le aw_n + b$, $\forall n \ge 0$, $w_0 = 0$, with $a = (1 - k\mu^*)^{-1}$, $b = \frac{1}{2}k^2(1 - k\mu^*)^{-1}$.

If $\mu^* < 0$ we obtain, by Lemma 13.2.6, $\|u_n\| = \|w_n\| \le k/(2|\mu^*|) = \text{TOL}/|\mu^*|$, $\forall n$, i.e., the same bound as for the continuous model in Theorem 13.2.1, and this holds *with no restriction on the step size.*

Also for $\mu^* = 0$ and for $\mu^* > 0$ we obtain approximately the same bounds as in Theorem 13.2.1; in the latter case the assumption that $k\mu^* < 1$ means a (mild) restriction on the step size. *All this is valid also for stiff problems.*

Again, in the discussion around Theorem 13.1.2, we promised more than bounds for u_n, and we shall therefore estimate $g_n = u_n - u(t_n)$. Put

$$r_n = \tfrac{1}{2}h_n^2\ddot{u}(t_n) + h_n(J(t_{n+1}, u_{n+1}) - J(t_{n+1}, u(t_{n+1}))u_{n+1} + \ldots.$$

We *assume that (say)* $\|\partial J(t,u)/\partial u\| < C$, *uniformly in* t. Then $\|r_n\| = (O(k^3) + O(k^2\|g_{n+1}\|))$, since $u_n = O(k)$. By straightforward computation, we obtain the difference equation,

$$g_{n+1} = g_n + h_n J(t_{n+1}, u(t_{n+1})g_{n+1}) + r_n,$$

hence

$$g_{n+1} = \big(I - h_n J(t_{n+1}, u(t_{n+1}))\big)^{-1}\big(g_n + r_n\big)$$

By straightforward computation, $\|g_{n+1}\| \le (1 - k\mu^* - O(k^2))^{-1}(\|g_n\| + O(k^3))$.

If $\mu^* < 0$ we then obtain, by Lemma 13.2.6,

$$g_n = O(k^2)/(|\mu^*| - O(k)) = O(\text{TOL}^2)/(|\mu^*| - O(k)),$$

uniformly in tt. This confirms that in this case the continuous model for error propagation is rather accurate, also for stiff problems with smooth solutions, and for large t, with no step size restriction.

Similar conclusions also hold for $0 \le \mu^* k < 1$, except that the bounds may become infinite as $t \to \infty$.

A nonlinear stability analysis can be worked out also for more complicated numerical methods, but it is much harder. Much useful information can, however, be obtained from the study of the **linear model problem**, $\dot{y} = Ay$, $y(0) = c$, where *A is a $s \times s$ matrix that can be diagonalized by well conditioned linear transformation.*

This is called the **linear stability analysis** of a numerical method for ODEs. We shall exemplify (in examples and problems) how the linear stability analysis can be reduced to the analysis of the **scalar test problem**.

$$\dot{y} = \lambda y, \quad \lambda \in \mathbf{C},\ y(0) = 1. \tag{13.2.29}$$

(Note that any complex number can be an eigenvalue of a real matrix.)

Example 13.2.9

We shall study the behaviour of Euler's method for the linear model problem. We obtain the difference equation,

$$y_{n+1} = (I + h_n A)y_n, \qquad y_0 = c. \tag{13.2.30}$$

Assume that A can be diagonalized by a well conditioned transformation T, i.e., $T^{-1}AT = \Lambda = \mathrm{diag}(\lambda_j)$. If we set $y = Tz$, and $d = T^{-1}c$, then the differential and difference equations become

$$\dot{z} = \Lambda z, \qquad z_{n+1} = (I + h_n\Lambda)z_n, \qquad z(0) = z_0,$$

respectively. Each of these vector equations falls apart into s scalar equations, where we simplify the inital conditions, without loss of generality,

$$w' = \lambda w, \qquad w_{n+1} = (1 + h_n\lambda)w_n, \qquad w(0) = w_0 = 1, \tag{13.2.31}$$

where $\lambda \in \mathrm{Spectrum}(A)$, and $w \in \mathbf{C}$, (since a real matrix can have complex eigenvalues). Such a reduction of a linear system to scalar equations can be done for most numerical methods, e.g., for all Runge–Kutta methods and linear multistep methods.

In order to simplify the writing, we now *restrict* the discussion to the case where A *has a real negative spectrum*, e.g., the matrix of Example 13.2.4, where the spectrum is $\{-100, -10, -1, -0.1\}$. Note that if $h_n\lambda < -1$, the sequence $\{w_n\}$ will have sign changes, in spite that the exact solution $w(t)$ has constant sign. [27].

It is even more remarkable that although $w(t)$ is exponentially decreasing, the inequality $|w_{n+1}| \le |w_n|$, holds only if $h_n\lambda \ge -2$. Thus, *the criterion for numerical stability for Euler's method is, in the case of real negative eigenvalues, that*

$$h_n \le h_{\mathrm{stab}} \equiv 2|\lambda_{\max}|^{-1}, \tag{13.2.32}$$

where $|\lambda_{\max}|^{-1}$ *is the smallest time constant of the system.* Then $\|z_{n+1}\| \le \|z_n\|, \forall n \ge 0$, hence $\|z_n\| \le \|z_0\|, \forall n \ge 0$, e.g., for weighted l_p-norms, (though not for every norm). We now return to the original variable y. It follows that $\|y_n\| \le \|T\|\|z_n\| \le \|T\|\|z_0\|$, hence

$$\|z_n\| \le \|z_0\| \Rightarrow \|y_n\| \le \|T\|\|T^{-1}\|\|y(0)\|. \tag{13.2.33}$$

[27] Compare the first assumption of Ex.13.2.7.It here reads $|h_n\lambda| < 1$, which is only twice as restrictive as the condition in (13.2.32)

Theoretically y_n is thus bounded for any non-singular T, but there are important classes of problems, where one requires *uniform boundedness* over an infinite family of ODE systems. Important examples are the huge ODE systems obtained after a discretization in the space variables of a PDE. The order of these systems increases without bounds, as the discretization become finer, and it *can* happen that the condition number of T also increases without bounds. In such cases the analysis of the scalar test problem can be misleading. That is why it is emphasized that T should be well conditioned.

If $h_n > h_{\text{stab}}$ for all $n > n_1$ (say), then $|w_n|$ grows exponentially; after a few steps the sequence $\{y_n\}$ of Example 13.2.4 has very little in common with the exact solution $y(t)$.

The condition in (13.2.32) happens to be valid also for Runge's 2nd order method. For other explicit methods, and for many implicit methods, the coefficient 2 is to be replaced by some other value. We saw, however, in Example **??** that there is no condition on the step size for the implicit Euler method if $\mu(A) \leq 0$. [28] and this is the case for many other *implicit* methods.

With the automatic control of step size the solution is, however, not likely to grow indefinitely, due to numerical instability, even if the method is not designed to handle stiff problems efficiently. The step size control will instead restrict the step size. Exactly what happens, depends on fine details in the implementation of the control, for example how sensitive the approximate error estimate is to perturbations. We shall try to describe it in principle.

In the beginning, the step size is small. A smooth and accurate solution is produced. If the motion becomes smoother as time goes by, the step size increases, and it can happen that the stability criterion is violated during a few steps. The error then grows, although it remains (almost) on the tolerated level. The motion is therefore kicked out a little to one of those tracks, mentioned at the end of Example 13.2.4, where a smaller step size is required. So, the step size is reduced, so that a smooth solution is produced again. Soon, the step size increases again etc.. The whole cycle is repeated again and again, so that *the step size fluctuates around the bound h_{stab} set by the stability criterion.* The results are still reliable. *Note that this is achieved by a well designed step size control, without any estimation of the time constants.* The only disaster that happens, if the automatic step size control is well designed, is that the time steps may become very short, compared to the intrinsic time scale of the motion. [29]

At the time of writing the phenomena just described are still the subject of a lively research, that has been timely reviewed by Hairer and Wanner [21, Section IV.2]. Ideas from Control Theory, e.g., PID Control, have been applied by K. Gustafsson, G. Söderlind and others.

For Euler's method, $p = 1$, $c_p = 2$. By Theorem 13.2.2, $k = 2\text{TOL}$ and the theoretical step size: $h_{\text{theo}} \approx 2\text{TOL}\tau_r$. The smallest time constant of the system of

[28] By Theorem 13.1.13, this is, for an appropriate choice of norm, a milder condition than the conditions on the spectrum of A used here.

[29] This should not make you believe that you have to buy a faster computer. You should instead use another numerical method.

Example 13.2.4 equals 0.01, and the local time scale is $\tau_r \approx t$. Then, by (13.2.32), $h_{\text{stab}} = 0.02$. The stepsize is therefore bounded by the stability property of Euler's method, i.e. $h = h_{\text{stab}}$, when $h_{\text{stab}} < h_{\text{theo}}$, which in this case happens when $t > 0.01/\text{TOL}$. For Runge's 2nd order method, $h_{\text{theo}} \approx \sqrt{6\text{TOL}\tau_r}$ is larger, while h_{stab} is the same, and hence the restriction of h due to the stability bound comes earlier for the more accurate method.

Example 13.2.10

Our program for Runge's 2nd order method (essentially constructed according to exercise C1 of Sec. 13.1) was applied to the problem $\dot{y} = 1 - y$, $y(0) = 0$, $t \in [0, 100]$ with $S = \max(0.001, |y|)$. For $\text{TOL} = 0.0016$ the first step size became $h = 0.01$. The step size then grew rapidly, and at $t \approx 12$ it became 2.7 that exceeds $h_{\text{stab}} = 2$. This shows that the "motion" has become stiff. The step size control then makes the stepsize fluctuate between 1.8 and 2.2. The results are accurate enough. The relative error fluctuates about 20% around $1.2\,\text{TOL}$. The whole computation was done in 86 steps till $t = 100$. Only 4 of them needed recomputation.

Note that the stiffness occurs when $|1 - y(t)| \leq e^{-12} < \text{TOL}/100$. So in this example the computation can be stopped when the motion becomes stiff, but this is not always the case.

For smaller tolerances the behavior is similar, but the violation of the stability condition is smaller. The relative error is close to TOL all the time. For $\text{TOL} < 10^{-6}$ less than 0.5% of the steps are rejected.

A message given by these examples, is that we have to distinguish between intervals, where a motion is not stiff and the step size can be predicted by Theorem 13.2.2, and intervals where it is stiff, and the step size may be restricted by the numerical stability properties of the method.

The reduction of the study of the linear system, $\dot{y} = Ay$, to the scalar equations, $w' = \lambda w$, $\lambda \in \text{spectrum}(A)$, which was done above for Euler's method, can be done in the same way for most numerical methods for ODEs. In general the solution of the difference equation produced by the numerical method depends only on the dimensionless parameter combination $q = \lambda h$.

Definition 13.2.11.

The **stability region** $\mathcal{S}$ *of a numerical method for the initial value problem for ODEs is the set of complex numbers $q = \lambda h$, such that the application of the method with a constant step size h to the scalar test equation $\dot{y} = \lambda y$, produces a sequence $\{y_n(q)\}$ that is bounded as $n \to \infty$, for any set of initial data.*

For a $k-$step method, $k \geq 1$,

$$|y_n(q)| \leq c \max\{|y_0|, |y_1|, \ldots, |y_{k-1}|\}. \tag{13.2.34}$$

More generally, the region $\mathcal{S}_\alpha$, $\alpha > 0$, is defined by the requirement that

$$|y_n(q)| \leq c\alpha^n \max\{|y_0|, |y_1|, \ldots, |y_{k-1}|\}.$$

When all solutions of the system $\dot{y} = Ay$ are bounded for $t > 0$, it is natural to generalize the conclusion of Example 13.2.6 to a requirement that

$$\lambda h \in \mathcal{S}, \ \forall \lambda \in \text{spectrum}(A).$$

Remember the assumption that A can be diagonalized by means of a *well-conditioned* matrix T.

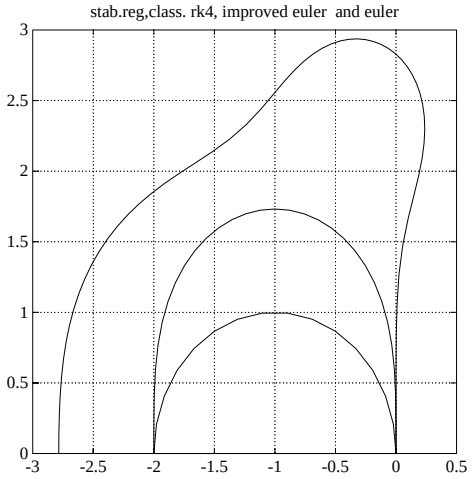

Figure 13.2.3. *Upper half of the stability regions for Euler's method, Runge's 2nd order method and Kutta's Simpson's rule (4th order). See Example 13.2.13.*

Example 13.2.12

By (13.2.31), the stability region of Euler's method is determined by the inequality, $|1 + q| \leq 1$. It is therefore a disc of unit radius in the complex $q-$plane, with center at -1.

Example 13.2.13

When Runge's 2nd order method is applied with constant step size h to the test equation $\dot{y} = \lambda y$, we obtain the difference equation

$$y_{n+1} = \left(1 + q + \tfrac{1}{2}q^2\right) y_n, \quad (q = \lambda h)$$

hence $\mathcal{S}$ is defined by the inequality $|1 + q + \frac{1}{2}q^2| \leq 1$. Its intersections with the real axis are at 0 and -2, just like Euler's method. The verification of these statements is left as an exercise. See Fig. 13.2.3.

Kutta-Simpson's rule is one of the most popular Runge–Kutta methods. It is 4'th order accurate and has four stages, i.e. four function values are computed in each step. Similarly $\mathcal{S}$ is defined by the inequality $|1 + q + \frac{1}{2}q^2 + \frac{1}{6}q^3 + \frac{1}{24}q^4| \leq 1$. More about this method in Sec.13.3.

Example 13.2.14

The implicit Euler method was introduced in Ex.13.2.8. The application to the scalar test equation yields the difference equation $y_{n+1} = y_n + qy_{n+1}$, i.e., $y_{n+1} = (1-q)^{-1}y_n$, hence $\mathcal{S}$ is determined by the inequality $|1-q| \geq 1$. It is thus *the exterior of a disk of radius 1 and center at 1.* Note in particular that the whole negative half-plane belongs to $\mathcal{S}$. The step size is therefore not restricted by stability reasons, if the eigenvalues of the Jacobian have negative real parts. We saw in Ex.13.2.8 how these results are extended to non-linear systems.

On the other hand, $\mathcal{S}$ *contains also a large part of the right half-plane.* Therefore, if the step size is too large in a problem where the (exact) motion is unstable, the computations may produce a smooth solution with no indication of instability. This can in some applications be rather dangerous. For example, if the purpose of a computation is to find out whether a certain physical system is stable or not. The implicit Euler method shares this property with most methods that are designed for stiff ODEs. One must *be aware of this.* For most applications, however, this does no harm.

In order to find $\mathcal{S}$, it is helpful to draw the **boundary locus**. This is the set of $q \in \mathbf{C}$ such that the difference equation obtained when the method is applied to the scalar test problem, $\dot{y} = \lambda y, \quad h\lambda = q$, has (at least) one of its characteristic roots on the unit circle. This condition is satisfied when $q \in \partial\mathcal{S}$, hence $\partial\mathcal{S}$, *if it exists, is part of the boundary locus.* For the simple methods that we have considered so far, $\partial\mathcal{S}$ and the boundary locus are identical, *but that is usually not the case.* It can even happen that $\mathcal{S}$ is void, in spite that the boundary locus is a smooth closed curve. In other cases, the boundary locus may be a curve that intersects itself. There are a number of technical matters concerning $\mathcal{S}$ and the boundary locus that will be discussed in §13.2.3.

Different computational situations require different stability concepts. We discuss only methods that with a sufficient number of in-data to a step, produce a unique result (y_{n+1}), provided that the step size is small enough. (The latter precaution is needed for implicit methods, see §13.2.4)

Definition 13.2.15. [30]

A method is **zero-stable***, if* $0 \in \mathcal{S}$.

A method is **strongly zero-stable** *if, for some* $\delta > 0$, $\mathcal{S}$ *contains the closed disk with radius* δ *and center at* $-\delta$.

A method is **weakly unstable** *if for any* q*, with* $\Re q < 0$*, there exists a solution of the difference equation such that* $y_n \to \infty$ *as* $n \to \infty$*. This notion is relevant for multistep methods only.*

Set $y_n(\infty) = \lim_{q\to\infty} y_n(q)$*, assumed that the limit exists.*

A method is **∞-stable** *if the sequence* $\{y_n(\infty)\}$ *is bounded as* $n \to +\infty$*. In other words:* $\infty \in \mathcal{S}$.

A method is **strongly ∞-stable** *if* $\lim_{n\to+\infty} y_n(\infty) = 0$*. In other words:* $\mathcal{S}$ *is the complement of a bounded region.*

[30] In older literature a zero-stable method was called *stable*, with no prefix. Other names have also been used. Some authors use the term *weakly zero-stable* instead of weakly unstable.

A method is **A-stable** *if $\mathcal{S}$ includes the left half-plane $\{q : \Re q < 0\}$.*
*A method is $A(\alpha)$-***stable***, $0 < \alpha \leq \pi/2$, if $\mathcal{S}$ includes the sector $\{z : |\arg(-z)| < \alpha\}$.*
A k-step method, $k \geq 1$, is **L-stable** *if it is A-stable, and if $y_k(\infty) = 0$ for any bounded initial values, $y_0,\ y_1, \ldots,\ y_{k-1}$.*

For several classes of methods, it can be shown that, as $h \to 0$, *the sequence* $\{y_n\}$ *converges* on a finite interval to the solution of the differential equation, for any differential system that satisfies very general conditions, *if and only if the method is consistent and zero-stable.* The "very general conditions" are roughly the conditions for existence and uniqueness given in Theorem 13.1.1. We saw above special results of this type for the explicit and the implicit Euler methods; more results will be seen in Secs.13.3 and 13.4. For more complete results, we refer to the excellent monographs of Butcher [4] and Hairer et al. [20].

The five stability concepts mentioned last are useful in the search for methods for stiff problems.

A method that is *not zero-stable* is also called **strongly unstable**. For almost all well-conditioned initial value problems, it will, in floating point arithmetic, produce solutions which quickly become useless, no matter how the step size has been chosen; typically the results at any fixed time becomes worse when the step size is small.[31]

Example 13.2.16 *An example of strong instability.*

One can show that the two-step method,

$$y_{n+2} + 4y_{n+1} - 5y_n = h\big(4f(y_{n+1}) + 2f(y_n)\big), \tag{13.2.35}$$

is 3rd order consistent; in fact it has the smallest local truncation error among all explicit two-step methods, see Sec. 13.4. Nevertheless it is useless.

For the scalar test equation the characteristic equation of the difference equation becomes $\zeta^2 + 4\zeta - 5 = q(4\zeta + 2)$. If $q = 0$ the roots are $\zeta_1 = 1,\ \zeta_2 = -5$, hence the method is strongly unstable. It can even be shown that $\mathcal{S}$ is void.

The solution of the difference equation is of the form

$$y_n = a(q)\zeta_1(q)^n + b(q)\zeta_2(q)^n, \quad \text{where} \quad \zeta_1(q) = 1 + O(q),\ \zeta_2(q) = -5 + O(q).$$

We may call the first term *regular* and the second term *irregular*. The latter is also called a *"parasitic oscillation"*. Let $\lambda = -1, h = 0.2$, i.e., $q = -0.2$. With exact initial conditions, i.e., if $y_0 = 1,\ y_1 = e^q$, $b(0.2)$ is only $5\,10^{-6}$, but the parasitic oscillation grows quickly, as shown by the upper line of Fig.13.2.4, where (due to the rapid increase of the oscillations) logarithmic scale is used, and only the absolute value of the error is shown. The lower line of Fig. 13.2.4 shows a computation where $y_1 = \zeta_1(q)$, with full IEEE double precision. The parasitic oscillations are here introduced by the rounding errors only. The figure shows that, with this low

[31] We always assume that the difference equation is used as a recurrence relation in floating point arithmetic. There is a theoretical possibility to use techniques similar to certain procedures that stabilize shooting methods for unstable boundary value problems; see Sec. 13.6, but the additional work would be enormous, and the method is, also in this modification, of no practical interest.

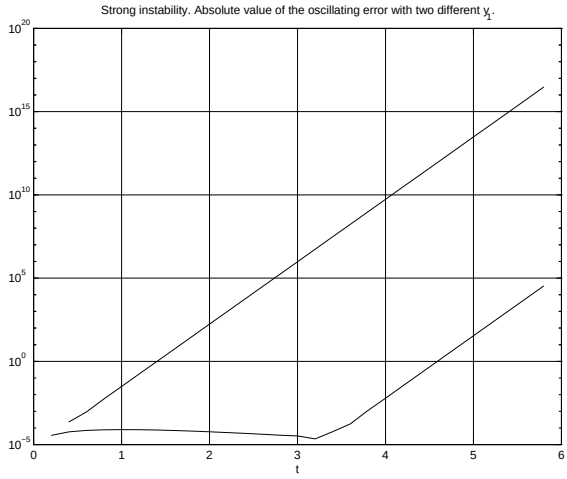

Figure 13.2.4. *The equation $\dot{y} = -y$, is treated by a strongly unstable two-step method, see Example 13.2.16, with step size $h = 0.2$, The error $y_n - \exp(-t_n)$ is oscillating. We here see only the smooth variation of its amplitude (in logarithmic scale), for two different values of y_1, as described in the Example.*

start, it takes some time for them to become visible, but they are amplified by the factor ζ_2 in every step.

Such a value for y_1 that is much better than the value that is exact from the point of view of the differential equation, can be found for this scalar test equation, but it is impractical to try to find it for (say) a nonlinear system. Anyway, it only delays the disaster. Also note that, since the amplitude of the parasitic oscillation grows approximately like $5^n = 5^{t/h}$; a reduction of the step size would make things even worse, due to the *strong* instability. □

Example 13.2.17 (*An example of weak instability.)*

The two-step method,

$$y_{n+2} - y_n = 2hf(y_{n+1}), \tag{13.2.36}$$

is called the **leap-frog method** or the *explicit midpoint method*. It is based on the simplest *central difference formula* for numerical differentiation and is 2nd order consistent.

For the scalar test equation the characteristic equation of the difference equation becomes $\zeta^2 - 1 = 2q\zeta$. If $q = 0$ the roots are $\zeta_1 = 1,\ \zeta_2 = -1$,

The solution of the difference equation is of the form

$$y_n = a(q)\zeta_1(q)^n + b(q)\zeta_2(q)^n,$$

where $\zeta_1(q) = 1+q+\ldots \approx e^q\ \zeta_2(q) = -1/\zeta_1(q) \approx -e^{-q}$. As in the previous example, we call the first term of y_n *regular* and the second term a *"parasitic oscillation"*.

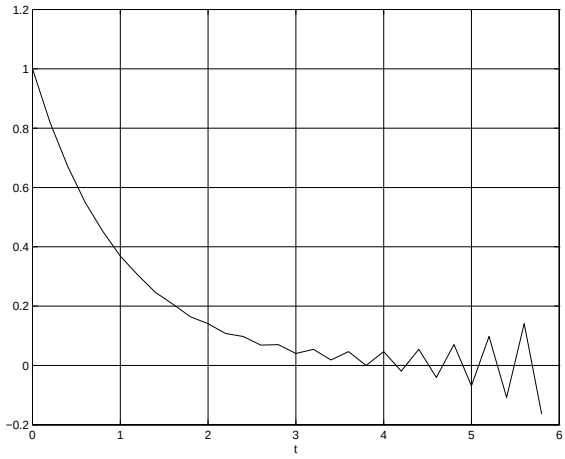

Figure 13.2.5. *The equation $\dot{y} = -y$, is treated by the weakly unstable leap-frog method, see Example 13.2.17, with $h = 0.2$ and $y_1 = \exp(-h)$.*

Let $\lambda = -1$, $h = 0.2$, i.e., $q = -0.2$. With exact initial conditions, i.e., if $y_0 = 1,\ y_1 = e^q$, it can be shown that $b(q) \approx \frac{1}{6}q^3$, but the parasitic oscillation grows like the second term, i.e. by 20% in every step (like the function e^t), thus much slower than in the previous example, see Fig. 13.2.5. If the instability is weak, the size of the oscillations becomes smaller, if the step size is reduced.

A computation was also performed, where $y_1 = \zeta_1(q)$, with full precision, `macheps` $\approx 10^{-16}$. The parasitic oscillations are here introduced by the rounding errors only, and they would not become visible in the figure until at $t \approx 28$. Such a value of y_1 can be found for the scalar test equation, but it is impractical to try to find it if (say) a nonlinear system is to be treated.

Moving averages or other filtering techniques are often used to reduce an oscillating component in a number sequence. In this context, with an unwanted component of the form $(-1)^n k_n$, where k_n varies slowly, a natural recipe is to replace y_n by

$$\bar{y}_n = (1 - \frac{1}{4}\delta^2)y_n = \frac{1}{4}(y_{n-1} + 2y_n + y_{n+1}),$$

as a post processing (or piecewise during the computation). A drawback is that this changes the regular component too by $-\frac{1}{4}\delta^2 y_n$, which may be comparable in size to the global truncation error of the numerical solution of the ODE. One can compensate for this by computing [32]

$$\hat{y}_n = (1 + \frac{1}{4}\delta^2)\bar{y}_n.$$

Note that the oscillations are growing only if $\Re q < 0$. The leap-frog method has successfully been used for linear systems, $\dot{y} = Ay$, with purely imaginary eigenvalues. Such systems appear in mechanical problems, and in hyperbolic partial

[32] Notice the plus sign in the difference correction this time. In fact $\hat{y}_n = (1 - \frac{1}{16}\delta^4)y_n$.

differential equations, if the spatial derivatives are approximated by means of central differences.

Weakly stable methods are, however, avoided in problems where the Jacobian has eigenvalues with negative real part. The technique with moving averages are not used much in this context, but it is good to be aware of this possibility, in problems of some other type.

We obtain the boundary locus by putting $\zeta = e^{i\phi},\ \phi \in [0, 2\pi)$ into the characteristic equation $\zeta^2 - 1 = 2q\zeta$, hence $q = i \sin\phi$. The boundary locus is thus the straight line segment from $-i$ to i, traversed up and down. A closer examination of the characteristic equation reveals that $\mathcal{S}$ is the same, except that the end points $\pm i$ must be excluded, because the root condition, Theorem 3.2.7, is not satisfied there; $\zeta = \pm i$ are double roots. Such a stability region is called *degenerate*. □

It is important to realize that the background to the scalar test problem and the stability region is a *system* of ODEs. Nobody is interested in integrating the test equation itself with $|\lambda h| > 1$ (say). The stability concepts give a helpful guidance in more general situations, e.g. for *nonlinear systems* and *variable stepsize*, although they do not exactly provide "the truth, the whole truth and nothing but the truth" in these cases. Since small perturbations in nonlinear systems are propagated according to the linearized variational equation, it is natural to substitute rough estimates of **the eigenvalues of the Jacobian** along the expected motion for λ in the test equation. For some classes of ODEs, inequalities are known for these eigenvalues, e.g., the Jacobian may be known or expected to be negative definite, hence its eigenvalues are negative.

Finally, it is sometimes believed that numerical methods are useless, e.g., when the eigenvalues are positive, because λh is then outside $\mathcal{S}$. That is a misunderstanding. Methods that are consistent and strongly zero-stable provide results with good *relative* accuracy, when applied to the equation $\dot{y} = y$, see Example 13.1.7. More generally, if a motion is sensitive to perturbations in the ODEs, the results obtained by the most popular numerical methods with a well designed step size control are usually no more sensitive to perturbations than the solutions to the ODEs are themselves. If a method is strongly zero-stable, the relevant region in the complex q-plane is the union of $\mathcal{S}$ and some small circle around the origin,(say) $|q| < 0.1$.

13.2.3 Linear Analysis of Numerical Stability

In this subsection we shall present graphical methods for the investigation of the stability of numerical methods. [33] Some theoretical background is given. Sometimes the graphical methods must be completed by algebraic algorithms (due to Routh, Schur and others) that will be presented in Sec. 13.9.

At the application to the linear test equation,

$$\dot{y} = \lambda y, \quad y(0) = 1, \quad q = \lambda h = \text{const.} \in \mathbf{C}, \tag{13.2.37}$$

[33] The study of this subsection can be postponed until the reader feels motivated for it, e.g., in connection with the last sections of this chapter. Among other things, it contains exercises in Computer Graphics, hopefully informative and amusing.

see also Definition 13.2.1, most numerical methods yield a difference equation, the characteristic equation of which is of the form,

$$\Psi(\zeta, q) = 0, \quad \Psi(\zeta, q) \equiv \psi_k(q)\zeta^k + \psi_{k-1}(q)\zeta^{k-1} + \ldots + \psi_1(q)\zeta + \psi_0(q). \quad (13.2.38)$$

where the $\psi_j(q)$ are real polynomials, the highest degree of which is (say) m. The k characteristic roots are named $\zeta_i(q)$, $i = 1 : k$. $\zeta_i(q)$ is called an *unstable root* if $|\zeta_i(q)| > 1$ or if $\zeta_i(q)$ is a multiple root of unit modulus. [34] It happens that several different methods yield the same characteristic polynomial $\Psi(\zeta, q)$, although they yield different results on problems other than this test problem.

Example 13.2.18

A **general linear multistep** method for the differential system $\dot{y} = f(y)$, $y(0) = y_0$, is defined by the difference equation

$$y_n - \sum_{i=1}^{k} \alpha_i y_{n+i} \; - h \sum_{i=0}^{k} \beta_i f(y_{n+i}) = 0, \quad (13.2.39)$$

where α_i and β_i are real parameters, h the step length. The formula (13.2.39) is also called a *linear k-step method*. We shall study this class of methods more thoroughly in Sec.13.4, together with another class of methods, named **one-leg k-step methods**, defined by a similar difference equation:

$$y_n - \sum_{i=1}^{k} \alpha_i y_{n+i} \; - \; hf\Big(\sum_{i=0}^{k} \beta_i y_{n+i}\Big) = 0. \quad (13.2.40)$$

The **generating polynomials**

$$\rho(\zeta) = \zeta^k - \sum_{i=1}^{k} \alpha_i \zeta^{k-i}, \qquad \sigma(\zeta) = \sum_{i=0}^{k} \beta_i \zeta^{k-i}, \quad (13.2.41)$$

play a fundamental role in the theory of multistep methods. We shall always *assume that the polynomials ρ and σ have no common factors.* For the standard test problem $\dot{y} = \lambda y$, $y(0) = 1$, $\lambda h = q$, the linear multistep method and the one-leg method with the same coefficients yield identical difference equations. The characteristic equation now reads,

$$\Psi(\zeta, q) \equiv \rho(\zeta) - q\sigma(\zeta) = 0, \text{ hence } \quad q = \rho(\zeta)/\sigma(\zeta). \quad (13.2.42)$$

For $q = \infty$ the characteristic equation reads $\sigma(\zeta) = 0$. The boundary locus becomes a single curve that is rather easily plotted after the computation of the rational function $q(\zeta) = \rho(\zeta)/\sigma(\zeta)$ for $\zeta = e^{2\pi ij/N}$, $j = 0, 1, \ldots N$ for some suitable value of N. If $\sigma(\zeta)$ has zeros on the unit circle, they must be determined, and the values of ζ, for which $|q(\zeta)|$ become large, are to be omitted. Sometimes the choice of N requires some trial and error; typically the uniformly distributed points on the unit

[34] Recall the root condition, Theorem 3.2.6.

circle are mapped to a sequence of points which are unexpectedly far from being equidistant.

Typically the boundary locus intersects itself many times, see e.g., Fig. 13.2.6. We shall below discuss how to proceed in order to find the stability region in this chaos. □

The map of the unit disk by the rational function $\rho(\zeta)/\sigma(\zeta)$ is usually much larger than $\mathcal{S}$, because it contains all q such that $\rho(\zeta) - q\sigma(\zeta)$ has *at least one* zero in the unit disk, while the condition $q \in \mathcal{S}$ requires that *all* zeros have to be located there. Find these sets in Fig. 13.2.6. The correct expression for $\mathcal{S}$ in terms of $\rho(\zeta)/\sigma(\zeta)$ is instead as follows:

Theorem 13.2.19. *Let $\mathcal{S}$ be the stability region of the linear multistep or one-leg method, generated by the polynomials ρ, σ. Then the complement of the closed unit disk is mapped onto the interior of the complement of $\mathcal{S}$ by the rational function $q = \rho(\zeta)/\sigma(\zeta)$. A point on $\partial\mathcal{S}$ belongs to $\mathcal{S}$, unless it is a cusp, i.e., the characteristic equation has a multiple root for this value of q.*

By the implicit function theorem for analytic functions, ζ and q are analytic functions (a conformal mapping) of each other in the neighborhood of any point (ζ_0, q_0) that satisfies $\Psi(\zeta_0, q_0) = 0$, unless it is a branch point, i.e. a point, where $\partial\Psi/\partial\zeta = 0$ or $\partial\Psi/\partial q = 0$. The first of these relations expresses that (13.2.38) has a multiple root, when it is considered as an equation for ζ for a given q. The roots $\zeta_i(q)$, $i = 1, 2, \ldots, k$ are branches of an analytic function. Two or more branches can meet, where $\partial\Psi/\partial\zeta = 0$. $\zeta_i(q)$ is continuous also at branch points, though it may not be differentiable there. Take for example $\Psi(\zeta, q) \equiv \zeta^2 - q$. (Consider also the different behavior in the example $\Psi(\zeta, q) \equiv \zeta^2 - q^2$.)

There are exceptional cases, where some of the above statements are not strictly true, for example, the points where $\psi_k(q) = 0$. We can avoid dealing with them as exceptional by considering the complex variables ζ and q as points on a **Riemann sphere** and hence consider ∞ as an ordinary point, (say) the north pole of a stereo-graphic projection from the sphere to the equatorial plane; the origin is the south pole. Neighborhoods, distances, continuity etc. are to be considered on the sphere.

If $\psi_k(q) \to 0$ as $q \to q_1$ then $\zeta_i(q) \to \infty$ for at least one i. We then say that $\zeta_i(q_1) = \infty$. The multiplicity of this root is $k - k'$, if the degree of of $\Psi(\zeta, q)$ drops from k to k', as $q \to q_1$. The use of the Riemann sphere is convenient in many other respects. It allows us, for example, to say that the function $\zeta = 1/q$, defined by the equation $q\zeta - 1 = 0$ is continuous everywhere, also at $q = 0$. Similarly, the continuity of the branches $\zeta_j(q)$ holds without exceptions on the Riemann sphere. (If you feel insecure about the handling of ∞, it may help to introduce $\hat{\zeta} = 1/\zeta$ and/or $\hat{q} = 1/q$ into the characteristic equation, and see what happens as $\hat{\zeta} \to 0$ and/or $\hat{q} \to 0$.)

What we have said about $\zeta(q)$ holds, mutatis mutandis, also for the inverse function. It has m branches denoted $q_j(\zeta)$, $j = 1 : m$. The characteristic polynomial when $q = \infty$ consists of the terms of $\Psi(\zeta, q)$ which contain q^m.

We shall only consider *consistent* methods. Since $y(h) = e^q$, one of the roots, denoted $\zeta_1(q)$ and called the *principal root*, should therefore approximate e^q, when $|q|$ is small. We say that the **order of linear consistency** is $\bar{p}$, if

$$\zeta_1(q) - e^q \sim \bar{c}q^{\bar{p}+1}, \quad \bar{c} \neq 0, \quad q \to 0. \tag{13.2.43}$$

$\bar{c}$ is called the **linear error constant**. For most methods $\bar{p}$ is equal to the usual order of consistency p, defined in Sec. 13.2.1, e.g., for all linear multistep methods, but there exist methods (of little practical interest) for which $p < \bar{p}$.

By (13.2.33) we can also write $\zeta_1(q) = e^{\lambda' h}$ where $\lambda' = (1+\bar{c}q^{\bar{p}})\lambda$. Set $nh = t$. Then

$$\zeta_1(q)^n = e^{\lambda' t}\big(1 + O(q^{1+\bar{p}}\lambda t)\big) = e^{\lambda t}(1 + \bar{c}q^{\bar{p}}\lambda t + \ldots) \tag{13.2.44}$$

The last expression reveals that *the global error is* $O(q^{\bar{p}})$. The detailed proof is left for Problem P10. For a consistent method $p \geq 1$, and hence $\bar{p} \geq 1$. *A necessary and sufficient condition for linear consistency thus reads*

$$\zeta_1(0) = 1, \quad \zeta_1'(0) = 1. \tag{13.2.45}$$

The stability region $\mathcal{S}$ is the set of complex numbers q such that the root condition (Theorem 3.2.5) is satisfied by $\Psi(\zeta, q)$, considered as a polynomial in ζ with q as a parameter. $\mathcal{S}$ is symmetric about the real axis, since we obtain conjugate values for q for conjugate values of ζ.

We saw in Example 13.2.14 that the stability region of the implicit Euler method is the exterior of a bounded region (actually a disk). This can also be expressed by saying that ∞ belongs to the interior of $\mathcal{S}$. An example, where ∞ is a boundary point of $\mathcal{S}$, is the θ-method for $\theta = \frac{1}{2}$, for which $\mathcal{S}$ is the half-plane $\Re q \leq 0$, see Problem P8. The term ∞-stable introduced in Sec.13.2.2 is an expression for this point of view.

For an *explicit* method the degree of $\psi_k(q)$ is less than m. Thus, for $q = \infty$ at least one of the roots of $\Psi(\zeta, q)$ is infinite. It follows that an explicit method of the class considered cannot be ∞-stable. In other words: *$\mathcal{S}$ for an explicit method is a bounded set in* $\mathbf{C}$.

It follows from continuity considerations that *a point q on the boundary $\partial\mathcal{S}$ belongs to $\mathcal{S}$, unless the characteristic equation has, for this value of q, a multiple zero of unit modulus.* One can show that the multiplicity cannot exceed 2 when $q \in \partial S$ (but it can be higher at other parts of the boundary locus), and that the presence of a double root is visible as a **cusp** on the boundary $\partial\mathcal{S}$, i.e., *a point from which at most one ray points into $\mathcal{S}$. Corners* of $\partial\mathcal{S}$ belong, however, to $\mathcal{S}$; you will see later that each method in Fig. 13.2.6 has corner(s) on the boundary of $\partial\mathcal{S}$.

So, *$\mathcal{S}$ is a closed set on the Riemann sphere, if its boundary has no cusps.* This has some nice consequences. For example, in Sec. 13.1.4 a method is defined to be A-stable if $\mathcal{S}$ includes the *open* left half-plane $\{q : \Re q < 0\}$. If there were a boundary cusp on the imaginary axis, then there must also be points in the open left half plane that do not belong to $\mathcal{S}$. The conclusion is that *for an A-stable method, $\mathcal{S}$ includes the imaginary axis too.*

A typical kind of cusp is shown in Fig. 13.2.6b for $\Psi(\zeta, q) = (\zeta - 1)(\zeta + 1)^2 - 4q\zeta^3$. The cusp at $q = 0$ is generated by a double root at $\zeta = -1$. In this example $\mathcal{S}$ is the outer region. o, from the cusp only one direction points *into* $\mathcal{S}$.

Less typical cusps occur for the leap-frog method, (Example 13.2.17). Recall that $\mathcal{S}$ *is the open line segment from* $-i$ *to* i. The endpoints are cusps (of an unusual kind), but the points between are not cusps, because *two* directions from them (up and down) lead into $\mathcal{S}$.

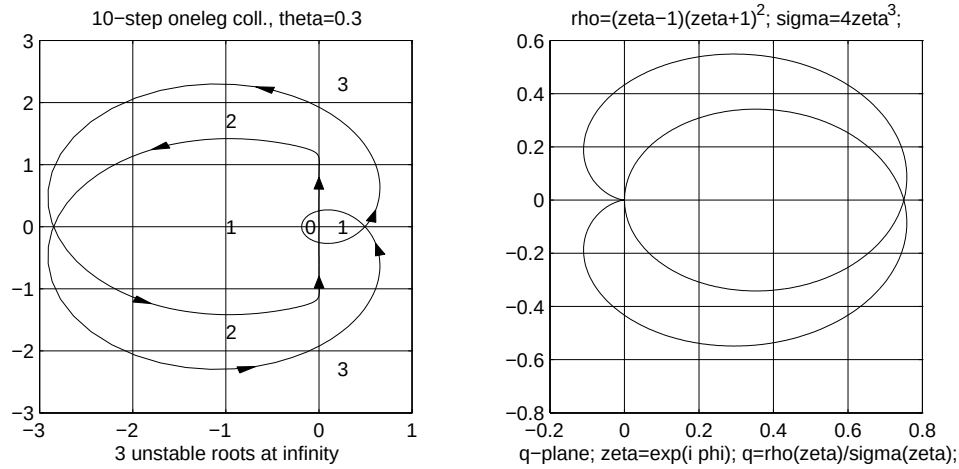

Figure 13.2.6. *(*a*) Boundary locus of a 10-step method. The small region near the origin that contains a zero, is the stability region. (*b*) Boundary locus for* $\Psi(\zeta, q) = (\zeta - 1)(\zeta + 1)^2 - 4\zeta^3 q$. *The stability region is the outer region, except for the origin that is a cusp.*

The *boundary locus*, which was defined in Sec. 13.2.2, can also be characterized as the conformal map to the q-plane of the unit circle of the ζ-plane, $\{\zeta = e^{i\phi} : 0 \leq \phi < 2\pi\}$. consists of one or more (at most m) *curves that divide the* $q - plane$ *into several parts* (e.g., six parts in Fig. 13.2.6a that shows the boundary locus of a certain 10-step method). Note the following **important rule:** *The number of unstable roots, counted with their multiplicity and including infinite roots, is constant within each part.* This follows from the continuity of $\zeta_j(q)$. (These integers are shown in Fig. 13.2.6a.) $\mathcal{S}$ is the small area marked with the digit 0. Notice that the boundary $\partial\mathcal{S}$ is composed by only a a few of the many pieces of the boundary locus.

Another important rule is due to the fact that, on the "microscopic scale", the orientation is conserved at the conformal mapping from the ζ-plane to the q-plane. We therefore look upon the boundary locus as a motion in the q-plane, along every branch of $q(\zeta)$, (to be marked by arrows in a plot) generated by a counter-clockwise motion along the unit circle in the ζ-plane. The neighborhoods of a short arc of the unit circle outside and inside the circle are then mapped into, respectively, the right and the left neighborhood of the corresponding arc of the boundary locus. *If* q *is moved from the right to the left of an arc of the boundary locus, the corresponding move in the* ζ*-plane tells that the number of unstable roots is decreased by one.* It is not necessary to watch the plotting process in order to set the arrows correctly. For a consistent method it follows from (13.2.43) that at least one branch of the

boundary locus will, at the beginning, move from $q = 0$ upwards, closely to the positive imaginary axis. In most cases this is enough for a correct setting of the arrows along the whole boundary locus; check the arrows in Fig. 13.2.6a, and set the arrows in Fig. 13.2.6b.

There is a hidden assumption in the last paragraph: *We assume that a small arc of the boundary locus corresponds to it only one small arc of the unit circle.* We continue on this assumption, because it has an important practical consequence:

If we know the number of unstable roots for **one** *value of q only, then we can, by the application of the rule of the previous paragraph, successively obtain the number of unstable roots for all the parts that the complex plane has been divided into by the boundary locus.* For the 10-step method of Fig. 13.2.6a, we assume that it is known that there are 3 unstable roots for $q = \infty$. Starting from this fact, check the markings in the figure, and then set the markings in Fig. 13.2.6b! In particular: *the stability region $\mathcal{S}$ is easily found*; it is the union of the parts, where the number of unstable roots is equal to zero. (In Fig. 13.2.6a there is only one such part.) It can happen that there is no such part; *$\mathcal{S}$ can be empty.* $\mathcal{S}$ can also, for example, *degenerate* into a line or a curve segment (see the next example), or into a point, Hairer and Wanner [1991, p.263].

There are, however, exceptions from this, e.g. if $F(\zeta, q)$ contains only even powers of ζ. The boundary locus is then circumscribed twice, hence the number of unstable roots decreases by 2 instead of 1, when the locus is passed from the right to the left. A less obvious exceptional case of the same type is found in Problem 20a. Next example illustrates that it can also happen that the boundary locus is traversed back and forth; the number of unstable roots is then the same on both sides. We *conjecture* that the most general exceptional case are combinations of these types, where the whole of the boundary locus (or at least each unicursal closed part of it) is traversed the same number of times (where backwards is negative). If this is true, it is enough to *check the number of unstable roots for one more value of q*, in a different part, in order to determine the number of unstable roots in the whole q-plane. (In the case of Fig. 13.2.6a, it was checked that there are 3 unstable roots for $q = -1$, so if we believe the conjecture, we conclude that the marking is correct.)

We shall now limit the discussion to the simplest cases, where $\Psi(\zeta, q)$ is either an affine function of q or an affine function of ζ. The linear multistep methods belong to the former category; the Runge–Kutta methods belong to the latter.

The **Runge–Kutta methods**, which will be studied more deeply in Sec. 13.3, yield at the application to the test equation $\dot{y} = \lambda y$ a difference equation of the form $y_{n+1} = R(q)y_n$, where $R(q)$ is a *rational function*, $R(q) = F(q)/G(q)$, where the polynomials F, G should have no common divisor. Hence we may write $\Psi(\zeta, q) = F(q) - G(q)\zeta$. Here F is a mth degree polynomial, while the degree of G is at most m. (m is the number of stages of the method.) The classical Runge–Kutta methods are *explicit*. In this case, $G(q) \equiv 1$, i.e., $R(q)$ is a polynomial.

Fig. 13.2.7 shows the boundary locus of two widely used Runge–Kutta methods. The former is **Kutta's Simpson's rule**, also called the classical fourth order

Runge–Kutta method, $m = p = 4$,

$$\zeta = \sum_{j=0}^{4} q^j/j! = e^q - q^5/120 - \dots . \tag{13.2.46}$$

The latter is called **Dopri5** (see Sec. 13.3), $m = 6$, $p = 5$, for which

$$\zeta = \sum_{j=0}^{5} q^j/j! \; + q^6/600 = e^q + q^6/3600 + \dots . \tag{13.2.47}$$

In order to plot the boundary locus in the general case, *all* roots of the mth degree

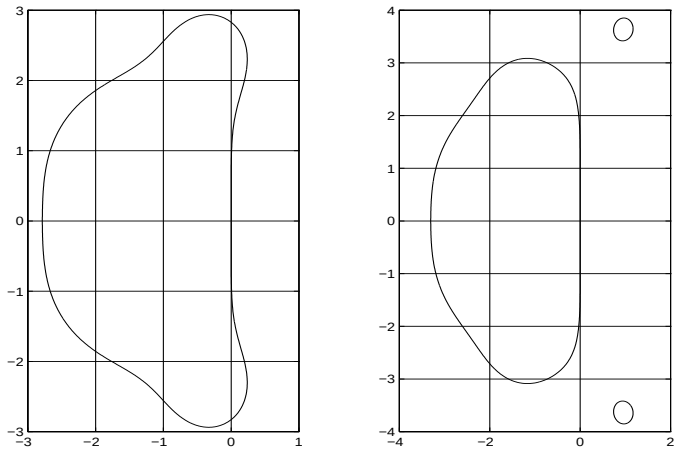

Figure 13.2.7. *Stability regions of two Runge–Kutta methods. (*a*) The classical 4'th order Runge–Kutta method, also called Kutta's Simpson's rule. (*b*) A popular 5'th order method called Dopri5. See also Sec.13.3.*

algebraic equation $F(q) - G(q)\zeta = 0$ are to be computed for $\zeta = e^{2\pi ij/N}$, $j = 0, 1, \dots N$ for some suitable value of N, to be chosen after some trial and error.

If we use a program that returns all roots of an algebraic equation, without the use a first approximation suggested by the user, the ordering of the roots must be inspected for every ζ and perhaps changed, otherwise the plot of the boundary locus can become rather strange.

The following alternative approach has the advantage that it is not necessary to know the coefficients of F and G. We shall see, in Sec. 13.3, that the numerical value of $R(q)$ is easily computed directly from the coefficients which define the method. The algorithm starts a "trip" at $\zeta = 1$, $q = 0$, and follows a "quasi-continuous" variation of a root of the equation $R(q) = \zeta$, i.e., one value of q is determined for $\zeta = e^{2\pi ij/N}$, $j = 0, 1, \dots, mN-1$, using e.g. the secant method with the previous values of q (or something better) as initial guesses. (A special rule is needed at the first point.) Note that the unit circle will be traversed m times during

the "trip". We will obtain m branches, and if we are lucky, they are all different, and we have obtained the boundary locus, Fig. 13.2.7a.

It happens, however, that we do not find m different branches this way. For example, in Fig. 13.2.7b the boundary locus consists of three separate curves. There is no chance to find the small curves with $q = 0$ as a starting point. The algorithm therefore must record the computed roots of the equation $R(q) = 1$. If less than m different roots have been obtained (within a tolerance), all roots of the equation $R(q) = 1$ must be computed, and the algorithm has to make a new "trip" (or more), starting from one of the remaining roots. We refrain from a discussion of the multiple root case.

For the Runge–Kutta methods it is easy to find $\mathcal{S}$ in a correct plot of the boundary locus. Since the characteristic equation is linear in ζ, the number of unstable roots (ζ for a given q) can be 0 or 1 only. Moreover, the boundary locus cannot intersect itself. (It can have a sort of double cusp, at a point where $\partial\Psi/\partial q = 0$. It is easier to find examples of this, if one plots more general level curves, $|\zeta| = r$, for a method. This is sometimes of practical interest.)

In Fig. 13.2.7a, $\mathcal{S}$ is marked by a zero; it is the interior of the closed curve. In Fig. 13.2.7b, $\mathcal{S}$ is the union of the interior of the three closed curves. A Runge–Kutta method is always strongly zero-stable; the origin and the area ust to the left of it belong to $\mathcal{S}$. $\mathcal{S}$ can be unbounded, if $R(q)$ is not a polynomial.

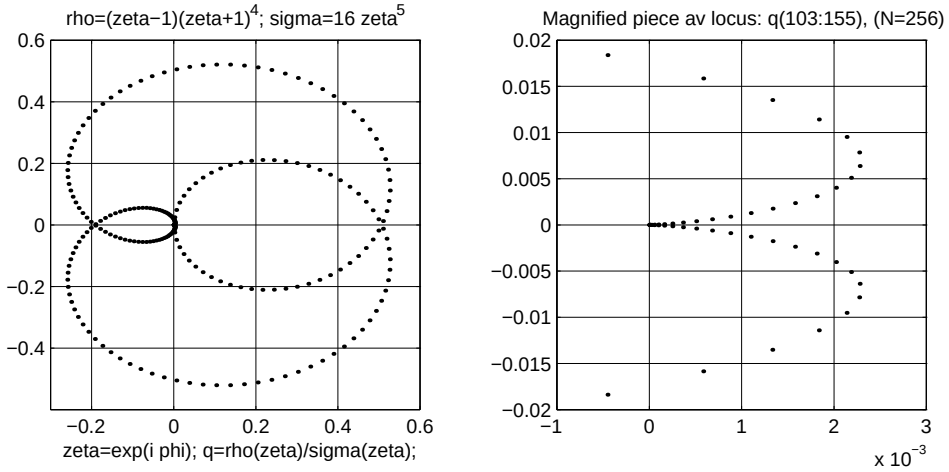

Figure 13.2.8. *The character of the cusp at $q = 0$ is not resolved well on the boundary locus of the multistep method $\rho(\zeta) = (\zeta - 1)(\zeta + 1)^4$; $\sigma(\zeta) = 16\zeta^5$. The map from 256 equidistant points on the unit circle in the ζ-plane is far from an (interpolated) equidistant point set in the q-plane. The picture to the right is a (linearly interpolated) magnified map of the 53 central points of the equidistant point set. This gives a different view of the cusp, which fits better to what one may expect from the analytic form of these characteristic polynomials.*

* Se efter om följande rader fortfarande är av intresse. Line 5600.

Sometimes the plotted boundary locus may not be accurate enough for displaying certain interesting features of a method, and it may be necessary to *magnify* part of the boundary locus, see Fig. 13.2.8. It happens, however, that this is not good enough. Some algebraic criteria, due to Routh-Hurwitz, Schur, Sturm etc., may be helpful as a complement to the graphics. See Sec. 13.9 that also contains a brief introduction to **order stars**, a very elegant and powerful tool for linear stability analysis. Many questions concerning the relations between stability and accuracy have been settled for a wide class of numerical methods, by the aid of this tool.

13.2.4 Implicit and Linearly Implicit Methods.

* Incomplete

Now we shall use the concept of logarithmic norm for proving the existence and uniqueness of a solution of a system of non-linear algebraic equations that may not be easily brought to a form required by the theorems of Sec. 13.1. We first note that such results are needed, because *a nonlinear system can have more than one solution, even if the Jacobian is non-singular everywhere.* A simple example is the system,

$$e^{y_1}\cos y_2 = b_1, \quad e^{y_1}\sin y_2 = b_2, \qquad (b_1, b_2) \neq 0.$$

The Jacobian determinant equals $\exp(2y_1) \neq 0$. The general solution is $(y_1, y_2) = (\ln r, \phi)$, where (r, ϕ) are the polar coordinates for the point with Cartesian coordinates (b_1, b_2). y_2 is determined only modulo 2π.

The result, Theorem 13.2.20, is in principle due to Desoer and Haneda [11]. Such systems may occur at every time step of the treatment of a stiff system of ODEs by an implicit method. The systems can be written in the form $F(y) = 0$, where

$$F(y) = \beta h f(y) - y + \gamma, \qquad (\beta > 0), \tag{13.2.48}$$

where β, γ and the time step h are constant during a time step. In practice a damped and modified Newton method (see Sec. 12.1.6) usually works well. Since global uniqueness is not to be expected for such systems, it can, however, happen that an unwanted solution of the system is computed, unless the time step and the error of the initial guess are small enough. The following theorem provides sufficient conditions for existence and uniqueness.

Theorem 13.2.20.

Let y_0 be a given point in $\mathbf{R}^s$, and set

$$D_r = \{y \in \mathbf{R}^s \quad : \quad \|y - y_0\| \leq r\}.$$

Consider the system $F(y) = 0$, where $F : D_r \to \mathbf{R}^s$ is in C^1. Assume that

1. $\|F(y_0)\| < r\delta$, $(\delta > 0)$.
2. $\mu(F'(y)) \leq -\delta$ *for* $y \in D_r$.

Then, the system $F(y) = 0$ has exactly one solution y^* in D_r, and $y^* = \lim_{s\to\infty} u(s)$, where [35] *$u(s)$ is the solution of the initial value problem $du/ds = F(u)$, $y(0) = y_0$.*

Proof. We first prove *uniqueness*. Suppose that y_1, y_2 are two different roots in D_r to the system $F(y) = 0$. By (13.1.14), we can write, $F(y_1) - F(y_2) = J(y_1 - y_2)$, where the matrix J is a neighborhood average of F'. By (13.1.28) and Assumption 2, $\mu(J) \le -\delta$. Then, by Theorem 13.1.25, statement C,

$$\|J(y_1 - y_2)\| \ge \delta\|y_1 - y_2\|.$$

Since $J(y_1 - y_2) = F(y_1) - F(y_2) = 0$ this contradicts the assumption that $y_1 \ne y_2$. Hence there is *at most one solution* in D_r to the system.

In order to prove the *existence*, let $u(s)$ be the solution of the initial value problem $du/ds = F(u)$, $u(0) = y_0$, which exists for all $s > 0$, unless it leaves D_r at some "time", i.e., unless there exists an $s' > 0$ such that

$$\|u(s') - y_0\| = r, \quad \|u(s) - y_0\| < r, \forall s < s'. \tag{13.2.49}$$

Since $F(u(s))$ satisfies the differential equation $dF(u(s))/ds = F'(u(s))F(u(s))$, we have $d\|F(u(s))\|/ds \le \mu(F'(u(s)))\|F(u(s))\| \le -\delta\|F(u(s))\|$. It follows that

$$\|F(u(s))\| \le \|F(y_0)\|e^{-\delta s} < r\delta e^{-\delta s}, \quad (s < s').$$

Then $du/ds = F(u(s))$ implies that

$$\|u(s') - y_0\| \le \int_0^{s'} \|F(u(s)\|ds < \int_0^{\infty} r\delta e^{-\delta s} ds = r,$$

but this contradicts (13.2.49). The contradiction shows that $u(s) \in D_r, \forall s > 0$. Then $F(u(s))$ converges exponentially to zero as $s \to \infty$; hence there exists *at least one point* y^* where $F(y^*) = 0$.

Summarizing our results, we conclude that the system $F(y) = 0$ has precisely one solution y^* in D_r, and that $y^* = \lim_{s\to\infty} u(s)$. □

COMMENT: In the application to (13.2.48) Assumption 2 becomes

$$\mu(f'(y)) \le (1 - \delta)/(\beta h) \text{ for } y \in D_r. \tag{13.2.50}$$

Note that this is a very liberal assumption for the function $f(y)$ (occurring in the differential system $\dot{y} = f(y)$.

Linearly Implicit metods. *** Incomplete

[35] We use the notation $u(s)$ in order to avoid contamination with the notation $y(t)$ used for the solution of *another* initial value problem, namely $\dot{y} = f(y)$, which is connected with our first application of this theorem.

13.2.5 Stiff and Differential-Algebraic Systems.

Example 13.2.21

Pyrolysis example, Incomplete, to be revised.
The following equations are based on a model for chemical pyrolysis studied by Datta [9]. (We have simplified the coefficients.)

$$\begin{aligned}
\dot{z}_1 &= -10^{-9}z_1 - 10^7 z_1 z_3,\\
\dot{z}_2 &= 10^{-9}z_1 - 10^9 z_2 z_3,\\
\dot{z}_3 &= 10^{-9}z_1 - 10^7 z_1 z_3 + 10^3 z_4 - 10^9 z_2 z_3,\\
\dot{z}_4 &= 10^7 z_1 z_3 - 10^3 z_4,
\end{aligned}$$

$$z_1(0) = 0.002 \quad z_2(0) = 0 \quad z_3(0) = 0 \quad z_4(0) = 0.$$

(a) Find a relation of the form $a_1 z_1(t) + a_2 z_2(t) + a_3 z_3(t) + a_4 z_4(t) = c$, and use this to eliminate z_4.

(b) Find a transformation of the form $z_i = 10^{b_i} y_i$ that brings this system (after the elimination of z_4) to the form, given in Problem 7 of the BDF package, see the version in `amods.m`. (Make a modification, if you believe it is necessary.)

NOTE. This scaling is advantageous for the analysis of what is going on in the problem. It is also easier in this form to guess how to set the scale factor vector `scavec`, the components of which define the breakpoints between the use of absolute and relative error estimates for the components of the solution. Find in the file `arcon...`, how `scavec` is used.

(c) Run Problem 7 with BDF, the default case. Make the step size variation ($\log h$ or $\log h/t$) visible on the screen by an appropriate choice of the plot constants. Note the number of steps.

(d) The following questions are only about rough estimates. Sometimes the order of magnitude is enough. Have the graphs from your computer run available or, if you have not solved (c) , the enclosed graph from a run done with another program. You are allowed to use them to inspire and to support your assumptions, but try to explain as much as possible by a theoretical analysis of the equations.

Note that y_1 is practically constant during a long time. How long ? This makes it rather easy to explain from the equations what is going on, until y_2 and y_3 reach their maxima. y_2 seems to vary linearly for a long time. Why, and what is the rate? Why and when does the linear variation stop?

y_3 seems to be the first variable to become stiff. When? It seems to try to approach a steady state, but then it becomes almost proportional to y_2. Why? What is the factor of proportionality, and why? Use this to explain the (approximate) max-value of y_2.

How small is the left hand side of the 3rd equation compared to the largest terms on the right hand side when $10^{-3} < t < 10^2$? (When a variable becomes stiff, its differential equation becomes approximately an algebraic equation.) What is the smallest time constant of the system? How is the step size variation in this interval?

When y_1 starts to change, the other variables also show a different behaviour. When $10^5 < t < 10^9$ the graphs of all variables are approximately straight lines in the log-log-diagram. The variables are therefore almost proportional to some powers of t. What powers? Try to explain why. First, note that the ratio y_3/y_2 becomes almost constant again. How large is it now? Why?

Some terms in the equations can be neglected in this interval, and the system can then be (approximately) solved by analytic techniques. When $t > 2\,10^9$, the graphs are no longer almost straight lines. What is the reason for the change?

If you find some other interesting features in this example, you are welcome to report them!

13.2.6 Other Special Types of Differential Systems.

* Incomplete

Singular points 0/0
Other discontinuities Filippov
Termination criteria Aitken extrapolation and modifications
Hamiltonian systems snd symplectic methods
Delay-differential systems
Systems with invariants

Review Questions

R1. Describe the theoretical step size strategy of the text, its concepts, assumptions and results, in particular Fig 13.2.1.

R2. What is, in this text, meant by the time constants of a system, the local time scale of a motion, a stiff motion and a stiff problem? Consider also a non-linear problem. Give an example of a stiff problem.

R3. Define the stability region of a numerical method for the initial value problem for ODEs. How is the study of the numerical solution of $\dot{y} = Ay$ (under a certain condition) reduced to the study of the scalar test equation. (It is sufficient to explain it for Runge's 2nd order method).

R4. Define zero-stability, strong zero-stability, strong instability, weak instability, $\infty-$stability, strong $\infty-$stability, A-stability, $A(\alpha)$-stability,

R5. Tell what is likely to happen to the step size variation, when Euler's method, or some other method with a bounded stability region, is applied to a stiff problem with our theoretical step control strategy . You may assume that the Jacobian is negative definite.

R6. Describe Runge's 2nd order method and the implicit Euler method. Give the formulas for their stability regions, and sketch the regions. Are the methods (strongly) zero-stable, (strongly) $\infty-$stable, A-stable ?

R7. Give the main features and results of the nonlinear error analysis of the explicit and the implicit Euler methods.

Problems

P1. (a) Is it true that the basic formulas for Runge's 2nd order method, given for the non-autonomous system $\dot{y} = f(t, y)$, $y(0)$ given, yield the same results as the formula you obtain, when you apply the method to the autonomous system $\dot{y}_1 = 1$, $\dot{y}_2 = f(y_1, y_2)$, $y_1(0) = 0$, $y_2(0) = y(0)$? Motivate your answer.

P2. (a) Show that the application of Runge's 2nd order method to the system $\dot{y} = Ay$ yields the formula $y_{n+1} = (1 + Ah + \frac{1}{2}A^2h^2)y_n$.

(b) Consider the inhomogeneous scalar problem

$$\dot{y} = \lambda y + (\alpha - \lambda)e^{\alpha t}, \quad y(0) \text{ given}, \tag{13.2.51}$$

and the system $\dot{z} = Az, \quad z(0) = (1,\ y(0))^T$, with

$$A = \begin{pmatrix} \alpha & 0 \\ \alpha - \lambda & \lambda \end{pmatrix}.$$

Show that $z_2(t) = y(t)$. If $\alpha \neq \lambda$, show that Runge's 2nd order method yields different results in the two problems, already in the first step, even if $y(0) = 1$).

(c) Equation (13.2.51), $y(0) = 1$, with the solution $y(t) = e^{\alpha t}$ is a useful test problem for the study of numerical methods. Now we shall use it for the study of Runge's 2nd order method with constant step size. Set $p = \alpha h$, $q = \lambda h$, $t_0 = 0$, $t = t_n = nh$, and notice that $pn = \alpha t$. Show that

$$y_{n+1} = (1 + q + q^2/2)y_n + (p - q)(q/2 + e^{p/2})e^{pn},$$

(d) Verify that the only solution of this recurrence relation with the initial condition $y_0 = 1$ reads $y_n = be^{pn} + (1 - b)(1 + q + q^2/2)^n$,, where

$$b = \frac{(e^{p/2} + q/2) \cdot (p - q)}{e^p - (1 + q + q^2/2)}.$$

Show that *the global relative error* equals $(b - 1)(1 - (1 + q + q^2/2)^n e^{-pn})$.

P3. (a) Show that the relations

$$N'(t) \approx \frac{1}{h(t)}, \qquad \frac{h_{n+1} - h_n}{h_n} \approx h'(t_n),$$

are valid for any reasonably smooth step size sequence.

(b) Show that if $h(t) \approx pt + q$ for $t \in [a, b]$, then

$$N(b) - N(a) \approx \frac{\ln(b + q/p) - \ln(a + q/p)}{p}.$$

How is this simplified when $q/p \ll a < b$?

(c) Suppose that $h(t) \approx 0.1t$ for $10^{-5} \leq t \leq 10^5$. How many per cent does the step size increase from a step to the next? Estimate the total number of steps for the whole computation.

(d) Assume that $h(t) \approx k\tau(t)$, where $\tau(t)$ is smooth, and k is independent of t. Show that $(h_{n+1} - h_n)/h_n^2 \approx \tau'(t)/\tau(t)$. Note that the right hand side is independent of TOL, but it may depend strongly on t. A value of k is given in Theorem 13.2.2 for the theoretical control strategy, but the result just derived holds under more general conditions.

P4. Derive the formula given in Sec. 13.2.3, according to which the norm of the scaled local error is proportional to $\tau_r(t)/\tau_q(t)$ during a motion (if the order p is constant). Show also that $\tau_r(t)/\tau_q(t)$ is constant if $y(t)$ is an exponential or a power function.

P5. (a) Derive Lemma 13.2.1 (in the beginning of §13.2.1) and the formulas given immediately after the lemma.

(b) Extend Lemma 13.2.1 to the case of piecewise constant parameters a, b. Do the analogous extension in Theorem 13.2.1.

P6. (a) Derive the expression, given in Example 13.2.13, for the stability region $\mathcal{S}$ of Runge's 2nd order method. Show that $\mathcal{S}$ is symmetric around the line $\Re q = -1$, and determine the exact coordinates of the top of $\mathcal{S}$.

(b) Show how the discussion of the linear problem $\dot{y} = Ay$, where A is a constant diagonalizable matrix, can be reduced to the study of the scalar test equation $\dot{y} = \lambda y$, $y(0) = 1$, $\lambda h = q$.

P7. Let A be the tridiagonal matrix of Example 10.3.4.

(a) Show in two ways that Euler's method produces bounded solutions to the system,

$$\frac{dy}{dt} = -cAy \qquad (c > 0),$$

if $0 < hc \leq \frac{1}{2}$. The first way is to apply Gershgorin's Theorem to find an interval that contains the spectrum of A, and use this to show that the spectrum of $-hcA$ lies in $\mathcal{S}$ of Euler's method for all $hc \in [0, \frac{1}{2}]$. In the second way you first show that $\|I - hcA\|_\infty \leq 1$, for all $hc \in [0, \frac{1}{2}]$. Is this true for variable step size?

(b) Is the same true for Runge's 2nd order method?

(c) Show that all solutions of the ODE system are bounded, as $t \to \infty$, by the use of $\mu_\infty(A)$.

(d) Show that all solutions of ODE system tend to zero, as $t \to \infty$, by the use of the knowledge about the spectrum of A.

P8. The *θ-method* is a one parameter family [36] of methods defined by the equation,

$$y_{n+1} - y_n = h\big(\theta f(y_{n+1}) + (1-\theta)f(y_n)\big), \quad 0 \leq \theta \leq 1.$$

[36] Some authors substitute $1 - \theta$ for θ.

(a) Show that $\mathcal{S}$ is a disk if $\theta < \frac{1}{2}$, a half-plane if $\theta = \frac{1}{2}$, and the exterior of a disk if $\theta > \frac{1}{2}$. Determine the intersections of the disks with the real axis.

(b) Show also that the method is 2nd order accurate for the standard test equation $\dot{y} = \lambda y$, if $\theta = \frac{1}{2}$, and only 1st order accurate for other values of θ. (We shall later see that this holds for any differential system.)

(c) Let a θ method with $\frac{1}{2} \leq \theta < 1$ be applied to the standard test equation with $|q| = |\lambda h| \gg 1$. Show that the sequence $y_1, y_2, y_3, \ldots$ have damped oscillations. For $\theta = \frac{1}{2}$, however, the damping is very weak and disappears as $q \to \infty$.

P9. (a) For the problem of Example 13.2.5, i.e. $\dot{y} = -my^{1+1/m}$, $y(1) = 1$, $(m > 0)$, show that if the relative error is considered all the time, then $|h(t)f'(y(t))| \leq (1 + 1/m)k \ \forall t$, where $k = |\text{TOL}/c_p|^{1/p}$. So, although the step size grows proportionally to time, *this problem does not require a method designed for stiff problems.* (In the past, there was a common misconception that the presence of different time scales during a motion makes the problem stiff, but it is rather when the *local* time scale becomes very much larger than the smallest *local* time constant, the motion becomes stiff.)

(b) For the standard scalar test equation $\dot{y} = \lambda y$, $y(0) = 1$, $(\lambda \in \mathbf{R})$, with the mixed absolute-relative error strategy, show that h reaches the stability limit for Runge's 2nd order method when $y(t) \approx 0.0015\text{TOL}$. (The circumstances are very different in a system with eigenvalues of different order of magnitude).

P10. In the study of numerical methods, one often encounters estimations of the following type. Let

$$\phi(q) = aq^{p+1} + bq^{p+2} + O(q^{p+3}), \quad |q| \ll 1, \quad nq = z = O(1).$$

(a) Show that $(e^q - \phi(q))^n = e^{q'n}$ where

$$q' = q - (1 - q)\phi(q) + O(q^{p+3}), \quad (|q| \ll 1).$$

(b) Show that

$$(e^q - \phi(q))^n - e^z \approx -ze^z\Big(aq^p + (b - a + cza^2/2)q^{p+1} + ...\Big), \qquad (13.2.52)$$

where $c = 0$ for $p > 1$, and $c = 1$ for $p = 1$. For example, when the test equation $\dot{y} = \lambda y$, is used in the study of a p'th order method, one sets $q = \lambda h$, $z = \lambda t$. The result is also valid if λ is a matrix, a, b are scalars, and $\|q\| \ll 1$.

(c) Apply this result to *"the circle test"*, i.e. the test problem $\dot{y} = iy$, $y(0) = 1$, step size h. The orbit is the unit circle of the complex plane. (See also Prob. 1.3.3.) Show that the global error for Runge's 2nd order method is approximately $te^{it}(ih^2/6 + h^3/8 + \ldots)$, and that the step size with the theoretical strategy becomes constant, $h = \sqrt{6\text{TOL}}$. Note that the radial error is positive and an order of magnitude smaller than the tangential error. How is TOL to be chosen in order that the error should be less than 10^{-3} after 10 revolutions? Suppose that the orbit is plotted with a straight line segment for each step.

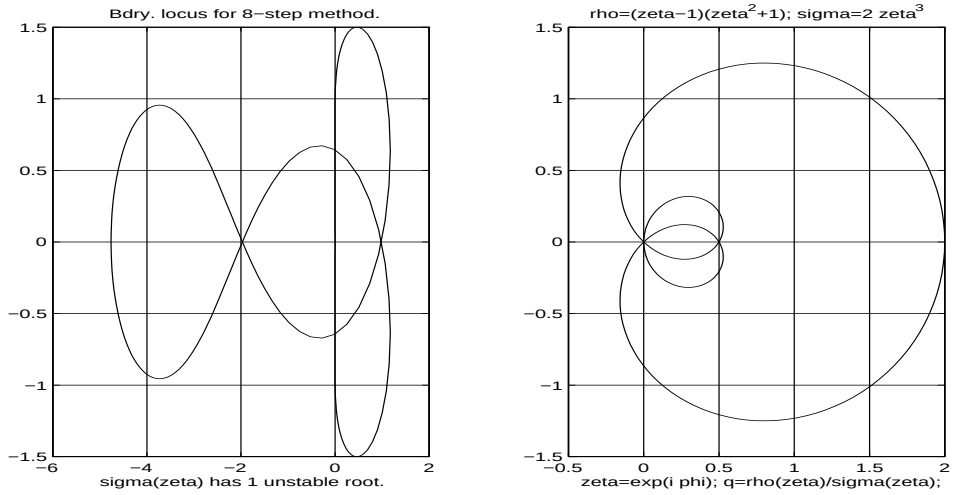

Figure 13.2.9. *Boundary locus for two multistep methods, see P 11.*

At what value of t does the global error become larger than the (local) interpolation error of the plotting?
Also show that he global error for Euler's method is

$$te^{it}(h/2 + (t/8 - i/3)h^2 + \ldots),$$

so that, for this method, the radial error is positive and much larger than the tangential error. Also show that $h = 2\text{TOL}$ with the theoretical strategy.

P11. (a) The left half of Fig. 13.2.9 shows the boundary locus for a consistent linear 8-step method, about which it is known that $\sigma(\zeta)$ has one unstable root. Find $\mathcal{S}$, if it exists. Is the method zero-stable?

(b) The right half of Fig. 13.2.9 shows the boundary locus of the linear 3-step method generated by the polynomials, $\rho(\zeta) = (\zeta - 1)(\zeta^2 + 1)$, $\sigma(\zeta) = 2\zeta^3$.
Find $\mathcal{S}$,if it exists. Is the method consistent? Is it zero-stable? Is it strongly zero-stable? Is it ∞-stable?

P12. Set

$$r(z) = \sum_{i=0}^{k} a_i z^i, \qquad s(z) = \sum_{i=0}^{k} b_i z^i, \tag{13.2.53}$$

A linear k-step method can also be expressed in the following form, where $\nabla = 1 - E^{-1}$ is the backward difference operator:

$$r(\nabla)y_{n+1} = hs(\nabla)f(y_{n+1}).$$

For a one-leg method the corresponding equation reads

$$r(\nabla)y_n = hf\big(s(\nabla)y_n\big).$$

(a) Prove the formula $\rho(\zeta) = \zeta^{k-1} r(1-\zeta^{-1})$, and the analogous formula for the polynomials σ and s.

(b) Show how to plot the boundary locus, when the coefficients of r and s are given, without calculating the coefficients of ρ and σ.

(c) The most widely used methods for stiff problems are probably the BDF methods, based on the truncated expansion of the differentiation operator given in section 3.2: $hD = \ln E = -\ln(1-\nabla)$, i.e.,

$$r(\nabla) = \nabla + \tfrac{1}{2}\nabla^2 + \frac{1}{3}\nabla^3 + \ldots + \frac{1}{k}\nabla^k, \qquad s(\nabla) = 1.$$

Compute $\rho(\zeta), \sigma(\zeta)$, for the BDF method, for a few values of k.(See also Exercise C7 of this section, and Sec. 13.4.)

Show that, for $k = 1$, the BDF method is identical to the trapezoidal method. What is its stability region?

Computer Exercises

C1. Vacant. (Old problem is moved to Sec.13.1, C10).

C2. Run the inhomogeneous test equation,

$$\dot{y} = \lambda y + (\alpha - \lambda)e^{\alpha t}, \quad t \in [0,3], \qquad \text{TOL} = 10^{-3},$$

and compare the actual global error divided by TOL with the bounds obtained in Theorem 13.2.1 and problem P2. Take $\lambda = -2$, $\alpha = -2, -0.5, +0.5$, $y(0) = 1$, and make also one run with $y(0) = 0$.

C3. A generalization of the classical dog curve problem. A dog chases a rabbit. The dog is smarter than usual, for it looks ahead of the dog by an angle equal to α radians. The speed of the rabbit is 1; the speed of the dog is $b > 1$. The motions can be described in the complex plane as follows. Let $z_D(t), z_R(t)$ be the positions at the time t of, respectively, the dog and the rabbit; $z_R(t) = (1 + a/16) + it$, i.e. the rabbit runs along a straight line. For the dog we obtain the differential equation:

$$z_D' = be^{i\alpha}\frac{z_R - z_D}{|z_R - z_D|}, \quad z_D(0) = a/16.$$

If necessary this can be replaced by two real equations. A chasing is terminated, e.g. when $|z_D - z_R| < 2\text{TOL}$ or $t = t_{end}$. Run five cases, $\alpha = 0.8 - 0.2a$, $a = 0,\ 1,\ 2,\ 3,\ 4$, and plot them on the same sheet. Try different values of b.

Note: A variant is to write $z_R = z_D + re^{i\phi}$, and solve the real differential equations for r, ϕ, or a single equation with either r or ϕ as the independent variable.

C4a. (Move to Sec. 13.8 ?????) Consider the initial value problem $\dot{y} = |1 - y|^{1/2}$, $y(0) = 0$, $0 \le t \le 5$.

(a) Find the exact solution $y(t)$, and compute $f'(y)$. For what value of t does the solution become singular. Note that the solution is not unique to the right of the singular point.

(b) Solve the problem numerically by the implicit Euler method, with a few fixed step sizes, e.g. $h = 0.01, 0.02, 0.04, 0.08$. In every step you have to solve a quadratic equation. Solve this by the algebraic formula; which root should you choose? Try Richardson extrapolation to the results, and compare with the exact solution.

(c) Also try explicit Euler with these step sizes, and Runge's 2nd order method with adaptive step size control.

C4. * Bör utvidgas !!!!!!
Study the problem $\dot{y} = 1000(\cos t - y)$, $y(0) = 0$, $0 \le t \le 1$. (see also p. XXX). Run it with the implicit Euler method (h=0.01 and 0.02), with the explicit Euler method (the same fixed stepsizes), and with Runge's second order method (adaptive control). Try Richardson extrapolation of the results, and compare with the exact solution. Explain the different behaviour of the methods.

C5. Heat equation. Unfinished.

C6. a) Reconstruct Fig. 13.2.3 and/or Fig. 13.2.7(b). The latter shows the stability region for the Runge–Kutta method Dopri5, see Eqn.(13.2.47).

b) Plot the level curves $|\zeta| = r$, for Kutta-Simpson's method for a few values of r. See Eqn.(13.2.46). Find to (say) two decimal places a value of r, such that the level curve intersects itself.

C7. Plot the stability regions of the BDF methods for $2 \le k \le 7$. (See Problem P12.) Give a (non-rigorous) answer to the following questions by a look at the plots. For which of these values of k is the method zero-stable? For which values is it ∞-stable? For which values is it A-stable?

C8. (a) The two-step method $y_{n+2} + 4y_{n+1} - 5y_n = h\big(4f(y_{n+1}) + 2f(y_n)\big)$ was discussed in Example 13.2.16. Plot the boundary locus. It looks very normal, but show that the stability region is void. Note that $\sigma(\zeta)$ has one zero at ∞ !

(b) Make a numerical experiment with this method on the initial value problem $\dot{y} = y^2$, $y(0) = 0.1$, $h = 1$. You need one more initial value; choose y_1 equal to the value of $y(0.1)$ correct to 5 decimals. (See also Example 13.2.11 and Fig. 13.2.4.) Would you recommend this method to your best friend?

(c) Make a similar numerical experiment for the leap frog method. It was studied in Example 13.2.17.

(d) A two-step method with degenerate stability region. (Unfinished).

C9. (Arenstorf orbits; a restricted 3-body problem of Astronomy.) The following 2nd order *complex* equation is a simple model for the motion of a satellite of negligible mass in the gravitational field of the earth (mass=$1 - m$) and the

moon (mass=m). The earth E and the moon M rotate in a circle around their mass center. The motion takes place in the plane of this circle. It is studied in a coordinate system that rotates with the earth and the moon. The location at time t of the satellite is described by a complex number, $z(t) = x(t) + iy(t)$. $E = -m,\ M = 1 - m$, (hence the mass center is at the origin.)
The satellite is therefore influenced by a centrifugal force and a Coriolis force in addition to the gravitational forces from the earth and the moon. The equation reads, in dimensionless form, (after a scaling of space and time): [37]

$$\ddot{z} = z - 2i\dot{z} - (1-m)(z-E)|z-E|^{-3} - m(z-M)|z-M|^{-3}.$$

We choose (according to Hairer et al.[20, pp. 128 and 197], $m = 0.012277471$,

$$z(0) = 0.994, \quad \dot{z}(0) = -i(2.0015851063790825 + 0.0301475231782543a).$$

For $a = 0$ and $a = 1$, the orbits should become (very different) closed curves. For $0 < a < 1$ the motion looks rather chaotic. The motion is, however, very sensitive to perturbations. Run it with TOL $= 10^{-3}$, 10^{-4}, until $t = 18$, for $a = 0$, $a = 1$, and some value between. Then take $a = 0$, with tolerances 10^{-3}, 10^{-4}, $10^{-5}, \ldots$. What tolerance and how many steps are needed for $a = 0$ in order that the orbit should look closed on the screen,

(a) at the first return to the neighborhood of the starting point?

(b) also at the second return to the neighborhood of the starting point?

You are likely to give up case (b), due to lack of time, if you try to solve it with Runge's 2nd order method. After this you will appreciate that there are problems for which a method of high order of accuracy is needed, even if the requirements for final accuracy is modest.

An experiment on a VGA screen with the 5th order Runge–Kutta method DOPRI5, see Sec.13.3, with an adequate step size control and an adequate (cubic Hermitean) interpolation for the graphics, showed that the final accuracy in case b) required a small tolerance, TOL $= 10^{-7}$. 681 steps (4086 function evaluations) were needed. How far do you get with 4086 function evaluations with Runge's 2nd order method when TOL $= 10^{-7}$?

13.3 One-Step Methods

We have several times encountered *Runge's 2nd order method*, see Sec. 13.1.1. It reads, for a non-autonomous system, $\dot{y} = f(t, y)$, ,

$$\begin{aligned} k_1 &= h_n f(t_n, y_n), \\ k_2 &= h_n f(t_n + \tfrac{1}{2}h_n, y_n + \tfrac{1}{2}k_1), \\ y_{n+1} &= y_n + k_2. \end{aligned} \tag{13.3.1}$$

[37] If your program cannot handle a complex system, rewrite it as a real system.

This method is an example of a *one-step method*; more specifically, it is an explicit *two-stage* second order accurate Runge–Kutta method.

The idea behind most one-step methods, is that (t_n, y_n) and a suggested value for the step size $h_n = t_{n+1} - t_n$ are the only input data to the step, where y_{n+1} is computed, and the error committed is estimated. If the error is acceptable, a new step size is suggested for the next step, otherwise a new step size is suggested for a recomputation of y_{n+1}. The one-step method is characterized by an **increment function** Ψ and a step size h such that

$$y_{n+1} = y_n + h\Psi(t_n, y_n; h).$$

A one step method can be *implicit*. Then this relation has, to begin with, the form $y_{n+1} = y_n + h\Psi^*(t_n, y_n, y_{n+1}; h)$. If Ψ^* satisfies the regularity conditions specified in Ch.$\tilde{1}$2, and if h is small enough, this vector equation can be solved, and the form $y_{n+1} = y_n + h\Psi(t_n, y_n; h)$ is still valid. See also §13.2.x.

Denote by $y(t; t_n, y_n)$ the trajectory of the system $\dot{y} = f(t, y)$, which passes through the point (t_n, y_n). We say that that the **order of consistency** equals p, iff $y_{n+1} - y(t_n + h; t_n, y_n) = O(h^{p+1})$. If $p \geq 1$ the method is **consistent**. We shall discuss consistent methods only.

If Ψ satisfies a Lipschitz condition, the method is automatically zero-stable, see the end of §13.3.1. This is the case for all explicit one-step methods in practical use. In *non-stiff* intervals, the order of consistency then becomes the **order of accuracy**; the *global* error is $O(h^p)$. In *stiff* intervals, however, it turns out that the order of consistency does not tell the whole truth about the order of accuracy, and several implicit one-step methods which seem attractive, by older criteria, are no longer recommended. A brief introduction to the important *order reduction* phenomenon is therefore given in §13.3.4.

The best known one-step methods for non-stiff problems are the *explicit Runge–Kutta*, or in short explicit **RK-methods**. They will be studied in §13.3.1, in particular the order conditions. The practical step size control is treated in §13.3.2, where examples of so-called *embedded* RK-methods are given.

Linear stability is discussed in §13.3.3 for both explicit and *implicit* RK-methods. Some of the most interesting implicit RK-methods, namely the *collocation* and the DIRK methods, will be discussed in §13.3.4 together with a brief introduction to the order reduction. Miscellaneous topics, e.g., dense output will be presented in §13.3.5 and in the remaining subsections there is an introduction to other types of one-step methods, in particular *Rosenbrock methods* and the *Taylor series method*.

Our standard references are, in the present section, Hairer, Nørsett and Wanner [20], in particular Ch.II, and Butcher [4], where most of the omitted proofs can be found.

13.3.1 Runge–Kutta Methods and Their Classical Order Conditions.

The idea of the *explicit* Runge–Kutta methods is that the calculations within a step proceeds in several *stages*. In each stage a value of $f(t, y)$ is computed in a

strategically chosen point near the trajectory. $y_{n+1} - y_n$ is a linear combination of these values, chosen in order that the expansion of this into powers of h should equal as many terms as possible of the corresponding expansion of $y(t_n + h; t_n, y_n)$.

A general **explicit** Runge–Kutta method with r stages [38] reads

$$\begin{aligned} k_1 &= h_n f(t_n, y_n), \\ k_2 &= h_n f(t_n + c_2 h, y_n + a_{21} k_1), \\ k_3 &= h_n f(t_n + c_3 h, y_n + a_{31} k_1 + a_{32} k_2), \\ &\cdots, \\ k_r &= h_n f_n(t_n + c_r h, y_n + a_{r1} k_1 + \cdots a_{r,r-1} k_{r-1}), \\ y_{n+1} &= y_n + b_1 k_1 + b_2 k_2 + \cdots + b_r k_r. \end{aligned}$$

The method is thus characterized by a tableau,

$$\begin{array}{cc} c & A \\ & b^T \end{array} \tag{13.3.2}$$

Here A is *a strictly lower triangular $r \times r$ matrix*, i.e., all elements above and on the main diagonal are zero. $b^T \in \mathbf{R}^r$ is a row, and $c \in \mathbf{R}^r$ is a column; $c_1 = 0$.

Note that if $f(t, y)$ is independent of y, then

$$y(t_n + h) - y(t_n) = \int_{t_n}^{t_{n+1}} f(t)dt, \quad y_{n+1} - y_n = \sum_{i=1}^{r} b_i f(t_n + c_i h).$$

In other words, the parameters b_i, c_i should be the weights and the nodes in an accurate quadrature formula. The first methods of Runge, Heun and Kutta were results of attempts to generalize well known quadrature formulas to ordinary differential equations.

As stated earlier, we require that a general purpose numerical method should give the same results when applied to a non-autonomous system as for the autonomous system obtained for the vector (t, y) if the system is augmented by the equation $\dot{t} = 1$. It can be shown (Problem 13.3.cA1) that this is the case iff $c_i = a_{i1} + a_{i2} + \ldots a_{i,i-1}$, $i = 1 : s$, and $\sum b_i = 1$, i.e., in vector-matrix notation,

$$c = A\mathbf{1}, \quad b^T\mathbf{1} = 1 \quad \text{where} \quad \mathbf{1} = [1, 1, \ldots, 1]^T. \tag{13.3.3}$$

We assume that this holds and can therefore in the following *restrict most of the discussion to the autonomous system $\dot{y} = f(y)$.*

We now display the tableaux[39] for Runge's 2nd order method and for two famous explicit methods with 4 stages, due to Kutta 1905, The latter are constructed as generalizations of classical quadrature rules namely *Newton's 3/8 rule*

[38] In most texts the number of stages is denoted by s. We cannot use this, since s has been reserved for the number of differential equations in a system.

[39] The zeros on and above the main diagonal of A are usually omitted from such tableaux. Here we print them, in order to emphasize that A is an $r \times r$ matrix.

and *Simpson's rule*, respectively.

$$\begin{array}{ccc} 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ \hline & 0 & 1 \end{array} \qquad \begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \\ \frac{2}{3} & -\frac{1}{3} & 1 & 0 & 0 \\ 1 & 1 & -1 & 1 & 0 \\ \hline & \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{array} \qquad \begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

The last of them has been particularly popular for a long time, and has been called the *classical Runge–Kutta method* or even "the" Runge–Kutta method. Nowadays they are often named *Kutta's 3/8 rule* and *Kutta's Simpson's rule*, respectively. One can show, by formulas and tables given below, that they are both 4'th order accurate. They have the same stability region, and the difference in performance is small.

The values $y_n + a_{j1}k_1 + a_{j2}k_2 + \ldots + a_{jj}k_j,\ j = 2 : r-1$, are approximations to $y(t_n + c_j h)$, but *these are typically much less accurate than the approximation* y_{n+1} *for* $y(t_n + h)$, and they are therefore (typically) of little use for the interpolation of the solution.

During the computer age a lively activity has been devoted to the design of higher order explicit RK-methods. This is a very hard task, since the number of order conditions for a method of order p equals the number of elementary differentials (or rooted trees) of order $\leq p$, and this grows rapidly. We quote a table from Sec. 12.5:

$$\begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 1 & 1 & 2 & 4 & 9 & 20 & 48 & 115 & 286 & 719 \end{array} \tag{13.3.4}$$

By adding five values of this table, we find that there are 17 order conditions for $p = 5$. This rises to 1205 for $p = 10$.

It is beyond the scope of this text to discuss the creative process of designing a RK-method that satisfies given demands for accuracy and stability. We shall, however, present algorithms and tables, which can be used for determining the order of accuracy of any suggested method, and for finding the principal error term when the order is 4 or less.

An interesting and difficult question is how many stages (r) are necessary to construct an explicit RK-method of order p. The question if there might be a fifth order method with five stages was not answered until the mid-sixties, and then the (negative) answer was given independently by Ceschino and Kuntzmann [5] and Butcher. That six stages are needed for $p = 5$ partly explains the popularity of the classical four-stage method; it takes two more stages to gain one order of accuracy. Butcher has then contributed with the results for higher order methods shown in the following table. They are called the Butcher barriers.

At the time of writing, the highest order obtained for an explicitly constructed RK-method is 10. Hairer [1978] achieved this using 17 stages. It was mentioned above that 1205 order conditions have to be satisfied for this. It is remarkable that this can be done by a method with only $17 \cdot 18/2 = 153$ parameters.

Table 13.3.1. *Maximal order of explicit Runge–Kutta methods.*

p	2	3	4	5	6	7	≥ 8
s	2	3	4	6	7	9	$> p+2$

For an *implicit* Runge–Kutta method (IRK) the defining formula **??** is almost the same, and the tableaux 13.3.2 is the same. The difference is that A *can now be a filled matrix*, not necessarily triangular. If A is a lower triangular matrix, but at least one diagonal element $a_{ii} \neq 0$, we have a *diagonally-implicit RK-method* (DIRK).

The numerical computation of the r vectors $k_1, k_2, \ldots, k_r$ is laborious for a general IRK, even if the huge system is can be transformed to a system where, in each step, r smaller systems of the form $y - \beta h f(y) = v$ has to be solved, see §13.2.4. It is simpler for a DIRK; the system is directly of this simpler form. If a_{ii} is the same for all i, it is even simpler, since β is the same for all the smaller systems, although the right hand side v is different. Such methods are called SDIRK (S stands for singly).

The following are the tableaux for *the implicit Euler method, the implicit midpoint method and a two stage SDIRK-method*, which is of order 3 for $\gamma = \frac{3 \pm \sqrt{3}}{6}$.

$$\begin{array}{cc} 1 & 1 \\ \hline & 1 \end{array} \qquad \begin{array}{cc} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array} \qquad \begin{array}{ccc} \gamma & \gamma & 0 \\ 1-\gamma & 1-2\gamma & \gamma \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

We shall now discuss the classical order conditions for non-stiff applications, for both explicit and implicit methods. You are advised to take a look at Section 12.5 first, since we shall use the notations and some results concerning multilinear mappings, Fréchet derivatives, elementary differentials and rooted trees. Recall that Taylor's formula reads, with vector-valued arguments and functions, e.g., $y_0, k, f(y) \in \mathbf{R}^s$,

$$f(y_0 + k) = f(y_0) + f'(y_0)k + \tfrac{1}{2} f''(y_0)(k,k) + \ldots. \tag{13.3.5}$$

Also recall that, by Eqn. (**??**), the formal Taylor expansion of the solution $y(t)$ around $t = t_n \in \mathbf{R}$ reads

$$y(t_n + h) = y(t_n) + \sum_{\mathbf{t}} \frac{h^{\rho(\mathbf{t})}}{\rho(\mathbf{t})!} \alpha(\mathbf{t}) F(\mathbf{t}) y(t_n). \tag{13.3.6}$$

Here the variable $\mathbf{t}$ runs through the set of all rooted trees; $\rho(\mathbf{t})$ is the order of the tree, and $\alpha(\mathbf{t})$ is another combinatorical parameter. $F(\mathbf{t})$ is an elementary differential tabulated below for low order trees, together with all other relevant tree functions. (Note that t and $\mathbf{t}$ denote very different concepts.)

It is conceivable that the vector y_{n+1} produced by a step of length h with an arbitrary RK-method, also admits a formal expansion into powers of h, that can be

rearranged as a sum over all rooted trees; to find its exact form is a non-trivial task, see Hairer et al., Ch.II, Theorems 2.11 and 3.2. The final result is not as messy as you might have expected, see Table 13.3.1.

Each term contains a factor $b^T\Phi(\mathbf{t};A)$, where A, b^T come from the tableau of the RK-method. The integer-valued factor $\gamma(\mathbf{t})$ and $\Phi(\mathbf{t};A) \in \mathbf{R}^r$ will be given in Table 13.3.2. The expansion reads

$$y_{n+1} - y_n = \sum_{\mathbf{t}\ \rho(\mathbf{t})\le p+1} \frac{h^{\rho(\mathbf{t}),}}{\rho(\mathbf{t})!}\alpha(\mathbf{t})\gamma(\mathbf{t})b^T\Phi(\mathbf{t};A)F(\mathbf{t})y_n + O(h^{p+2}). \tag{13.3.7}$$

By matching the coefficients of this expansion with (13.3.6), i.e., the Taylor expansion of the exact solution, we obtain the following theorem.

Theorem 13.3.1. *The classical order of an RK-method is the largest integer p, such that*

$$\gamma(\mathbf{t})b^T\Phi(\mathbf{t};A) = 1$$

for all rooted trees $\mathbf{t}$ of order $\le p$.
The local error $y_{n+1} - y(t_n + h; t_n, y_n)$ reads

$$\frac{h^{p+1}}{(p+1)!}\sum_{\mathbf{t},\ \rho(\mathbf{t})=p+1} \alpha(\mathbf{t})\big(\gamma(\mathbf{t})b^T\Phi(\mathbf{t};A) - 1\big)F(\mathbf{t})(y_n) + O(h^{p+2}).$$

The local error in a step is thus bounded by Ch^{p+1}, where C is expressible in terms of bounds for the partial derivatives of order $p+1$ or less, uniformly for y in some closed region D.

Of course, the expansion (13.3.7) can be used, only as long as all partial derivatives of f needed for the computation of $F(\mathbf{t})(y_n)$ exist. So, if $f \in C^q(D)$ only, where $q < p$, we can only state that the local error is $O(h^{q+1})$.

Note that the knowledge of $\alpha(\mathbf{t})$ is not needed for the determination of the order, but the principal error term requests $\alpha(\mathbf{t})$ for the trees of order $p+1$.

Before we go into the details about trees, tree functions and elementary differentials, we shall take a look at global error bounds and convergence. We have already such bounds for the continuous model for the error propagation in Theorems 13.1.23 and 13.2.1. In order to prove convergence we shall now consider the actual discrete error propagation; recall the staircase curve (Fig. 13.1.3).

Theorem 13.3.2. *Let $D \subset \mathbf{R} \times \mathbf{R}^s$ be a neighborhood of the trajectory $(t, y(t;t_0,y_0))|\, t_0 \le t \le b$. Suppose that $\mu(f'(y)) \le \mu^*$, $\forall y \in D$, and that, for some $p \ge 1$ all elementary differentials of f, of order up to p are continuous in D.*

Let (t_i, y_i), $0 \le i \le N$, $h = \max(t_i - t_{i-1})$, be the point sequence obtained by means of the explicit RK-method (13.3.2); the order of consistency is $p, p \ge 1$.

Then the local error has a bound of the form Ch^{p+1}, as long as h, N are small enough for this sequence to remain in D. Set $C' = C\exp(-\mu^* h)$ if $\mu^* > 0$, otherwise $C' = C$.

Then, the global errors are, for $t = t_i$, bounded by

$$\|y_i - y(t_i; t_0, y_0)\| \leq \begin{cases} C' h^p \frac{e^{\mu^*(t-t_0)}-1}{\mu^*}, & \text{if } \mu^* \neq 0;. \\ C' h^p (t - t_0), & \text{if } \mu^* = 0; \end{cases}$$

The **order of accuracy** *of the method thus equals p. If $f \in C^q(D)$ only, where $1 \leq q < p$, the order of accuracy is typically reduced to q.*

As $h \to 0$ the solutions produced by the method converge uniformly to the exact solution $y(t; t_0, y_0)$.

The proof is omitted, since it is very similar to the proofs of (**??**) and Theorem 13.1.23. See also Fig. 13.1.4.

Example 13.3.3 *The truncation error of Runge's 2nd order method*

See Example 13.2.1. (Some additional comments should perhaps be made here.) For the sake of brevity we write $f, f', \ldots$ for $f(y_n),\ f'(y_n), \ldots$. We obtain, by the definition of the method and (13.3.x+1)

$$\begin{aligned} y_{n+1} &= y_n + f(y_n + \tfrac{1}{2}hf)h = hf + hf'\tfrac{1}{2}h + \tfrac{1}{2}hf''(\tfrac{1}{2}hf, \tfrac{1}{2}hf) \\ &= y_n + \dot{y}h + \tfrac{1}{2}\ddot{y}h^2 + \frac{1}{8}f''(f,f)h^3 + \ldots \\ &= y(t_n + h) + \big(\frac{1}{8}f''(f,f) - \frac{1}{6}y'''\big)h^3 + \ldots \end{aligned}$$

(We regret that we must use dashes also for a time derivative here.) In the simple program outlined in Example 1.3.2 the term $\frac{1}{8}f''(f,f)$ is considered less important than $\frac{1}{6}y'''(t)$ and is therefore neglected. *This is the simplification of the error estimate*, referred to in a comment to the program outlined in Example 1.3.2. It is, of course, not generally valid, but it is all right for linear autonomous systems, and seems acceptable in several analyzed non-linear test examples too, e.g., Example 13.1.6 (??). So far it has worked so well in practice that we do not even think it is necessary to have a safety factor. (You can instead choose a tolerance on the low side of your demands, if you feel insecure.)

The program uses, in the step size control, $\frac{1}{3}\|k_2 - k_1\| \approx \|hf'\frac{1}{2}hf\| = \frac{1}{6}\|\ddot{y}\|h^2$. with a scaled max-norm. Note that this is, no estimate of the local ' error but, in accordance with the theory of Sec. 13.6.1 (???), a (simplified) *estimate of the error per local unit of time*, where the local unit of time means $\tau_q(t) = \|\ddot{y}(t)\|/\|y'''(t)\|$.

13.3.2 On the Computation of Elementary Differentials and Tree Functions.

We shall now describe two algorithms for the computation of $\rho(\mathbf{t}), \gamma(\mathbf{t}), F(\mathbf{t}), \Phi(\mathbf{t}; A)$ for a rooted tree $\mathbf{t}$ that is composed by simpler rooted trees,
for which these functions are known. In fact, given the trivial values of these functions for $\mathbf{t} = \mathbf{t}_{11}$ (one single vertex), the algorithms are sufficient for computing the functions for any rooted tree. Results obtained by these algorithms are collected in Table 13.3.2.

The first algorithm is, in principle, taken from Hairer, Nørsett and Wanner, loc. cit., Ch.II, Eqns. (2.17)–(2.19).

Algorithm I. (m), $m \geq 1$. Suppose[40] that $\mathbf{t}$ is created, when $\mathbf{t}_1, \mathbf{t}_2, \dots \mathbf{t}_m$ are set on a common new root. Then

$$\begin{aligned}
\rho(\mathbf{t}) &= 1 + \sum_{k=1}^{m} \rho(\mathbf{t}_k);\\
\gamma(\mathbf{t}) &= \rho(\mathbf{t}) \cdot \gamma(\mathbf{t}_1) \cdot \gamma(\mathbf{t}_2) \cdots \gamma(\mathbf{t}_m)\\
F(\mathbf{t}) &= f^{(m)}\big(F(\mathbf{t}_1), F(\mathbf{t}_2), \dots, F(\mathbf{t}_m)\big)\\
\Phi(\mathbf{t}) &= A\Phi(\mathbf{t}_1) .* A\Phi(\mathbf{t}_2) .* \dots . * A\Phi(\mathbf{t}_m)
\end{aligned}$$

We here use the Matlab notation $.*$ for the elementwise multiplication of vectors (arrays). In order to reduce the number of parentheses, we make the convention that the multiplication of a matrix by a vector has a *higher priority* than the elementwise multiplication.

The function $\alpha(\mathbf{t})$ is not changed by Algorithm $I(1)$. Otherwise we refer to Butcher, loc. cit., or Hairer, Nørsett and Wanner concerning the computation of $\alpha(\mathbf{t})$. Also note that, by Theorem 13.3.2, $\alpha(\mathbf{t})$ is not needed for the determination of the classical order of an RK-method, but it is needed for finding the error constants.

Next algorithm can be derived from Algorithm I, (Problem 6), but we give it a name, because it is used so frequently in the formation of new rooted trees and is more convenient than the direct application of Algorithm I.

Algorithm II. Given a rooted tree $\mathbf{t}'$, consisting of $m-1$ trees $\mathbf{t}_1, \mathbf{t}_2, \dots \mathbf{t}_{m-1}$ connected to its root. (Algorithm I(m-1) can thus be applied to $\mathbf{t}'$.) A new tree $\mathbf{t}$ is created by connecting one more vertex to the root of $\mathbf{t}'$. Then:

$$\begin{aligned}
\rho(\mathbf{t}) &= 1 + \rho(\mathbf{t}');\\
\gamma(\mathbf{t}) &= \gamma(\mathbf{t}')\rho(\mathbf{t})/\rho(\mathbf{t}');\\
F(t) &= f^{(m)}(F(\mathbf{t}_1), F(\mathbf{t}_2), \dots F\mathbf{t}_{m-1}), f);\\
\Phi(\mathbf{t}) &= \Phi(\mathbf{t}') .* c.
\end{aligned}$$

Example 13.3.4 *Examples of trees, and applications of the algorithms.*

Denote by $\theta(q)$ the number of rooted trees of order q. By (12.5.11???), we have, for $q \geq 1$, $\theta(q) = 1,\ 1,\ 2,\ 4,\ 9,\ 20, \dots$. The trees with $\rho(\mathbf{t}) = \mathbf{q} \geq \mathbf{3}$ are recursively obtained as follows. We use two subscripts for a tree; the first one is the order $\rho(\mathbf{t})$.

Suppose that $q \geq 3$; the cases $q = 1$ and $q = 2$ are trivial. The trees of order q obtained by the application of Algorithm II to the trees $\mathbf{t}_{q-1,\nu}$, $nu = 1 : \theta(q-1)$, are named $\mathbf{t}_{q\nu}$. For example, for $\nu = 1$ we find, by induction in q, that the tree $\mathbf{t}_{q,1}$ is a root directly connected to $q-1$ vertices. Moreover, for this particular tree, $\gamma(\mathbf{t}) = \rho(\mathbf{t}) = q$, $F(\mathbf{t}) = f^{(q)}f^q$, $\Phi(\mathbf{t}; A) = c.^q$.

[40] When $\mathbf{t}$ has only *one* subscript, this has nothing to do with the order of the tree. When there are *two* subcripts, however, the first of them equals the order.

Similarly, the trees of order q obtained by the application of Algorithm I(1) to the trees $\mathbf{t}_{q-1,\nu},\ \nu = 1 : \theta(q-1)$, are named $\mathbf{t}_{q,\nu+\theta(q)-\theta(q-1)}$.
There is no conflict in the choice of names, as long as $\theta(q) - 2\theta(q-1) \geq 0$. For $q = 3 : 10$ this quantity takes the values $0, 0, 1, 2, 8, 19, 56, 147$, according to the table in (12.5.11)??? This time we find, for $\nu = \theta(q-1)$ that So, $\mathbf{t}_{q,\theta(q)}$ is "the tall tree" with all q vertices on the trunk. For this particular tree, $\gamma(\mathbf{t}) = q!,\ F(\mathbf{t}) = (f')^q,\ \Phi(\mathbf{t}; A) = A^{q-2}c$. Further applications of algorithms I(1) and II are seen in the upper part of Figure 13.3.1.
We do not give a general proof that the $2\theta(q-1)$ trees of order q thus obtained by these algorithms are all different, but it is easy to check this, when they have been computed. It is, e.g., easy to see that all trees in Table 13.3.2 are different. For if two trees were equal, they would have the same $(\rho(\mathbf{t}), \gamma(\mathbf{t}), \alpha(\mathbf{t})$, and you see at a glance that this is not the case. If this test turns out to be insufficient for an extension of the table, there are several other simple tests.
You now find that Algorithms II and I(1) are sufficient for computing the trees and the tree functions, up to $q = 4$. For $q = 5$, we have seen how to find $\mathbf{t}_{\mathbf{5},\nu}$ for $\nu = 1 : 4$ by Algorithm II, and for $\nu = 6 : 9$ by Algorithm I(1). Now, only $\mathbf{t}_{\mathbf{55}}$ is missing, but Algorithm I(2) can map the pair $\mathbf{t}_{21}, \mathbf{t}_{21}$ to a new tree of order 5 that we are free to call $\mathbf{t}_{55}$. See the lower part of Fig. 13.3.1.
Now all trees up to $q = 5$ are known; the reader is advised to draw them; in Sec. 12.5 you see examples of different "topologically equivalent" graphs for the same tree. Also check a few of the results in Table 13.2.3 (except $\alpha(\mathbf{t}_{\mathbf{5},\nu})$). We ignore the computation of $\alpha(\mathbf{t})$; as mentioned above, this function is not needed for the determination of the order.
We proceed to $q = 6$. Algorithm II yields $\mathbf{t}_{\mathbf{6},\mathbf{1:9}}$, and since $\theta(6) - \theta(5) = 11$, Algorithm I(1) yields $\mathbf{t}_{\mathbf{6},\mathbf{12:20}}$. Now, only $\mathbf{t}_{\mathbf{6},\mathbf{10}}$ and $\mathbf{t}_{\mathbf{6},\mathbf{11}}$ are missing. Again Algorithm I(2) helps for mapping the pair $\mathbf{t}_{21}, \mathbf{t}_{31}$ to $\mathbf{t}_{6,10}$, and for mapping the pair $\mathbf{t}_{21}, \mathbf{t}_{32}$ to $\mathbf{t}_{6,11}$. See the lower part of Fig. 13.3.1.

$Tree$	$F(\mathbf{t})$	$\Phi(\mathbf{t}; A)$	$\gamma(\mathbf{t}))$
$\mathbf{t}_{6,10}$	$f''(f'f, f''(f, f))$	$Ac. * A(c. * c)$	36
$\mathbf{t}_{6,11}$	$f''(f'f, f'f'f)$	$Ac. * A^2c$	72

□

Table 13.3.2 Rooted trees and elementary differentials etc. up to order 5.

$Tree$	$F(\mathbf{t})$	$\Phi(\mathbf{t};A)$	$\gamma(\mathbf{t})$	$\alpha(\mathbf{t})$
$\mathbf{t}_{11}$	f	$\mathbf{1}$	1	1
$\mathbf{t}_{21}$	$f'f$	c	2	1
$\mathbf{t}_{31}$	$f''(f,f)$	$c.*c$	3	1
$\mathbf{t}_{32}$	$f'f'f$	Ac	6	1
$\mathbf{t}_{41}$	$f'''(f,f,f)$	$c.*c.*c$	4	1
$\mathbf{t}_{42}$	$f''(f'f,f)$	$Ac.*c$	8	3
$\mathbf{t}_{43}$	$f'f''(f,f)$	$A(c.*c)$	12	1
$\mathbf{t}_{44}$	$f'f'f'f$	A^2c	24	1
$\mathbf{t}_{51}$	$f^{iv}(f,f,f,f)$	$c.*c.*c.*c$	5	1
$\mathbf{t}_{52}$	$f'''(f'f,f,f)$	$Ac.*c.*c$	10	6
$\mathbf{t}_{53}$	$f''(f''(f,f),f)$	$A(c.*c).*c$	15	4
$\mathbf{t}_{54}$	$f''(f'f'f,f)$	$A^2c.*c$	30	4
$\mathbf{t}_{55}$	$f''(f'f,f'f)$	$Ac.*Ac$	20	3
$\mathbf{t}_{56}$	$f'f'''(f,f,f)$	$A(c.*c.*c)$	20	1
$\mathbf{t}_{57}$	$f'f''(f'f,f)$	$A(Ac.*c)$	40	3
$\mathbf{t}_{58}$	$f'f'f''(f,f)$	$A^2(c.*c)$	60	1
$\mathbf{t}_{59}$	$f'f'f'f'f$	A^3c	120	1

13.3.3 Error Estimation and Step Size Control

It is hard to apply the expression for the local error as given in Theorem 13.3.1, Two other procedures have been used. The first one is based on the Richardson idea, Sec. 3.3.5; in fact it was used for error estimation without (active) extrapolation by Runge as early as 1895. The second one, which is indirectly based on the error expression, is called the technique with **embedded RK-formulas**.

The oldest and most straightforward device for the control of the local error is **step doubling** and Richardson extrapolation. Suppose we compute starting from (x_n, y_n) *two* steps with step size $h/2$ using a Runge–Kutta method of order p giving a value y_{n+1}. We then compute, again starting from (x_n, y_n), *one* step with step size h to obtain $\tilde{y}_{n+1}$. Then

$$y(x_n) = y_{n+1} + 2c\Big(\frac{h}{2}\Big)^{p+1} + O(h^{p+2})$$
$$y(x_n) = \tilde{y}_{n+1} + ch^{p+1} + O(h^{p+2}),$$

and subtracting we obtain

$$y_{n+1} - \tilde{y}_{n+1} = 2c\Big(\frac{h}{2}\Big)^{p+1}(2^p - 1) + O(h^{p+2}).$$

Hence an approximation $\hat{y}_{n+1}$ of order $p+1$ is

$$\hat{y}_{n+1} = y_{n+1} + e_{n+1}, \qquad e_{n+1} = (y_{n+1} - \tilde{y}_{n+1})/(2^p - 1). \tag{13.3.8}$$

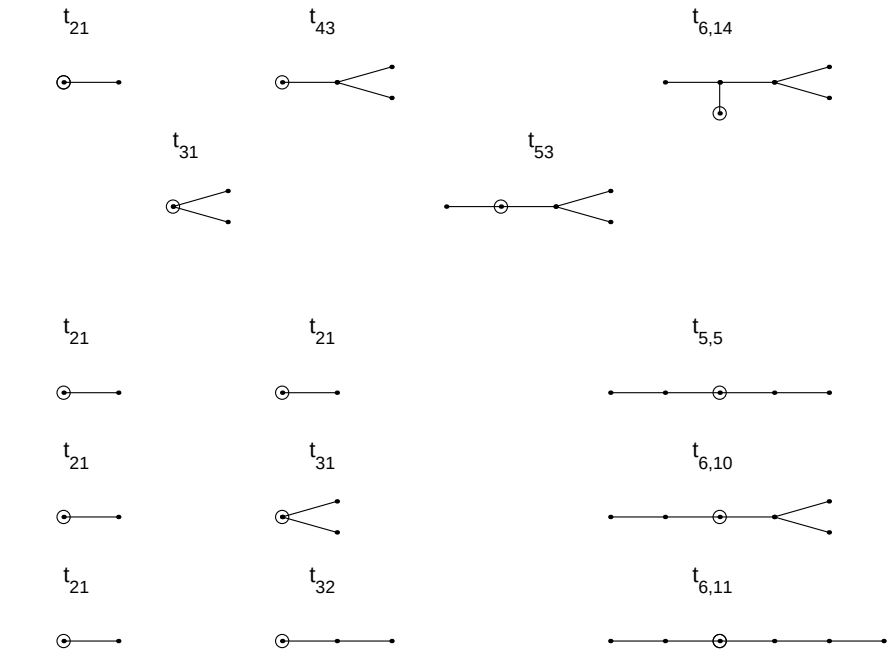

Figure 13.3.1. *Applications mentioned in the text of Example13.3.4. In the upper part, Algorithm II maps the trees of the 1st row to the trees on the 2nd row, half a step to the right, while Algorithm I(1) maps the trees of the 2nd row to the trees of the 1st row, half a step to the right. The lower part shows three applications of Algorithm I(2).*

Here e_{n+1} gives a simple estimate of the error in the *unextrapolated* value y_{n+1}. The solution is then advanced either from y_{n+1} or $\hat{y}_{n+1}$. In the latter case this is called local extrapolation.

The error estimate in (13.3.8) can be used to automatically adjust the step size as follows. Let $l_n = \|e_{n+1}\|$ be a measure of the local error. A common policy is to keep the local error per unit step below a given tolerance ϵ,

$$l_n \leq \epsilon(x_{n+1} - x_n) = \epsilon h. \tag{13.3.9}$$

The new step size h' is chosen to satisfy this condition for the next step. This leads to the choice

$$h' = h\left(\frac{\theta\epsilon h}{l_n}\right)^{1/p}, \tag{13.3.10}$$

where $\theta \leq 1$ (typically $\theta = 0.9$) is a preset safety factor. If the criterion (13.3.9) is not satisfied for the current step, this step is *rejected* and the computations of y_{n+1} repeated with the new step size h'. In several programs the user is asked to set upper and lower bounds for the permissible step size. Further, h is usually not allowed to increase or to decrease too fast.

Using (13.3.8) with the classical fourth order Runge–Kutta method we need $8+3 = 11$ function evaluations to proceed two (half) steps from y_n to $\hat{y}_{n+1}$. (Note that the function evaluation of $f(x_n, y_n)$ can be shared by the two different steps sizes.) This is an overhead of of $3/8 = 37.5\%$.

It is more efficient to use Runge–Kutta formulas which simultaneously give approximations to the local error. The idea is to use *a pair* of Runge–Kutta methods characterized by the tableau

$$\begin{array}{cccccc} 0 & & & & & \\ c_2 & a_{21} & & & & \\ c_3 & a_{31} & a_{32} & & & \\ \vdots & \vdots & \vdots & \vdots & & \\ c_s & a_{r1} & a_{r2} & \cdots & a_{r,r-1} & \\ \hline & b_1 & b_2 & \cdots & b_{r-1} & b_r \\ & \hat{b}_1 & \hat{b}_2 & \cdots & \hat{b}_{r-1} & \hat{b}_r \end{array}$$

The pair of formulas are usually constructed so that the methods

$$y_{n+1} = y_n + \sum_{i=1}^{s} b_i k_i, \qquad \hat{y}_{n+1} = y_n + \sum_{i=1}^{r} \hat{b}_i k_i,$$

have orders p and $q = p+1$. The difference of the two results then provides an error estimate. Such formulas are called **embedded Runge–Kutta methods**.

The idea of using such formulas was first proposed by Merson. The most well-known of his methods is given by the five stage method

$$\begin{array}{ccccccc} 0 & & & & & & \\ \frac{1}{3} & \frac{1}{3} & & & & & \\ \frac{1}{3} & \frac{1}{6} & \frac{1}{6} & & & & \\ \frac{1}{2} & \frac{1}{8} & 0 & \frac{3}{8} & & & \\ 1 & \frac{1}{2} & 0 & -\frac{3}{2} & 2 & & \\ \hline b_i & \frac{1}{10} & 0 & \frac{3}{10} & \frac{2}{5} & \frac{1}{5} \\ \hat{b}_i & \frac{1}{6} & 0 & 0 & \frac{2}{3} & \frac{1}{6} \end{array}$$

It can be shown that $\hat{y}_n$ is a fourth order method. Although y_n is in general only a third order method, for a linear differential equations with constant coefficients $f(x, y) = Ay + bx$ it becomes effectively fifth order. An estimate of the local truncation error is given by the difference

$$l_{n+1} = \frac{1}{30}(-2k_1 + 9k_3 - 8k_4 + k_5), \tag{13.3.11}$$

see Lambert [1973, pp. 131–132]. Two steps here take ten function evaluations against the eleven required by the step doubling process described previously. However, we have to continue with the fourth order estimate whereas with Richardson extrapolation we can use local extrapolation to get a fifth order estimate. Also, when applied to a nonlinear differential equation it frequently grossly overestimates the error which leads to a poor step-size control. In spite of this the method has been used successfully, e.g., in the NAG subroutine library.

Another popular embedded Runge–Kutta method is the method of order 4(5)[41] developed by Fehlberg [15] which requires six function evaluations per step. The first five of these combine to produce a fourth-order method and all six give a fifth order method. A method with the same coefficients has been implemented as a method of order 5(4) in a much used program called RKF45 by Shampine and Watts [1977]. However, it suffers from the disadvantage that the two formulas are based on the same quadrature formula, which leads to poor step-size control for some problems.

Fehlberg devised his methods so that the error terms for the lower order result y_1 were minimized. Practical results indicate that local extrapolation is preferable. If this is used, then the error terms of the higher order result $\hat{y}_1$ should instead be minimized. This is done in a more recent seven stage method of order 5(4) by Dormand and Prince given below, called DOPRI5.

0							
$\frac{1}{5}$	$\frac{1}{5}$						
$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$					
$\frac{4}{5}$	$\frac{44}{55}$	$-\frac{56}{15}$	$\frac{32}{9}$				
$\frac{8}{9}$	$\frac{19372}{6561}$	$-\frac{25360}{2187}$	$\frac{64448}{6561}$	$-\frac{212}{729}$			
1	$\frac{9017}{3168}$	$-\frac{355}{33}$	$\frac{46732}{5247}$	$\frac{49}{176}$	$-\frac{5103}{18656}$		
1	$\frac{35}{384}$	0	$\frac{500}{1113}$	$-\frac{125}{192}$	$-\frac{2187}{6184}$	$\frac{11}{84}$	
b_i	$\frac{35}{384}$	0	$\frac{500}{1113}$	$-\frac{125}{192}$	$-\frac{2187}{6184}$	$\frac{11}{84}$	0
$\hat{b}_i$	$\frac{5179}{57600}$	0	$\frac{7571}{16695}$	$\frac{393}{640}$	$-\frac{92097}{339200}$	$\frac{187}{2100}$	1/40

This method is also constructed so that $a_{si} = b_i$ for all i, and therefore the next to last evaluation of f in a current step can be re-used for the first evaluation in the the following step. This method seems to be the most efficient of the methods of order 5(4), and is suitable for tolerances down to about 10^{-7}.

Another well known Runge–Kutta code with step size control is DVERK of order 6(5) with 8 stages. This method, due to Verner, is based on a pair of formulas of order 5 and 6. It is available in the IMSL Library. Fehlberg has made a method of order 7(8) with 13 stages. It has been much used for high precision computations, e.g., in astronomy. Excellent results have also been obtained by the 8(7) method of

[41] the first integer is the order of the accepted value; the second is the order of the other value used in the error estimate.

Prince and Dormand [14]. Material about several methods of Dormand and Prince is found also in Hairer, Nørsett, and Wanner [20, Sec. II.5]; the 1st edition of that book (1987) contains additional material. which also includes a subroutine DOPRI8 implementing the Prince and Dormand method with step size control. These codes are preferable for tolerances between approximately 10^{-7} and 10^{-13}.

For a code to be efficient over a wide range of accuracy requirements, it is important to be able to vary the order of the formulas being used. Such variable-order codes must estimate the error that would have resulted in using formulas of different orders. Most codes that implement Runge–Kutta methods, however, are of fixed order. Variable-order codes are common for multistep methods, see Sec. 13.3.

13.3.4 Linear Consistency and Stability Analysis for Runge–Kutta Methods

Consider a general RK-method with r stages, implicit or explicit.

$$k_i = hf(y_n + a_{i1}k_1 + a_{i2}k_2 + \cdots + a_{ir}k_r), \quad i = 1:r. \tag{13.3.12}$$

$$y_{n+1} = y_n + \sum_{j=1}^{r} b_j k_j, \quad t_{n+1} = t_n + h. \tag{13.3.13}$$

Set,as above, $A = [a_{ij}]_{i,j=1}^{r}$.

$$\mathbf{1}^T = (1, 1, \ldots, 1), \quad \mathbf{k}^T = (k_1, k_2, \ldots, k_r), \quad \mathbf{b}^T = (b_1, b_2, \ldots, b_r).$$

For the scalar test equation $y' = \lambda y$ with $q = \lambda h$, we obtain
$\mathbf{k} = qy_n \cdot \mathbf{1} + qA\mathbf{k}$; $\quad y_{n+1} = y_n + \mathbf{b}^T\mathbf{k}$, hence

$$\begin{aligned} \mathbf{k} &= qy_n \cdot (I - qA)^{-1} \cdot \mathbf{1} \\ y_{n+1} &= (1 + q\mathbf{b}^T(I - qA)^{-1}\mathbf{1})y_n \equiv R(q)y_n. \end{aligned} \tag{13.3.14}$$

$R(q)$ is called the **stability function**. The *stability region* $\mathcal{S}$ of an RK-method is thus the set $\{q : |R(q)| \le 1\}$. You read in §13.2.3 how to construct $\mathcal{S}$.

It follows, *e.g.*, from the expression of the matrix inverse in terms of determinants, that $R(q)$ is a *rational function* of q. The denominator is $\det(I - qA)$, which is a polynomial of degree $\le r$. The numerator is also a polynomial of degree $\le r$.

The exact solution of the scalar test equation reads $y_n = e^{\lambda t_n} = e^{qn}$, hence $y_{n+1} = e^q y_n$. The consistency of a method is thus related to how well the power series expansion of $R(q)$ approximates e^q, though the notion *linear consistency* $\bar{p}, \bar{p} \ge p$, was introduced in §13.2.3, since it can happen that a method is more consistent for the special test equation than for a general nonlinear system.

The Padé approximants studied in §3.4.3 are, in this sense, the best rational approximations to a power series. In fact, Example 3.4.1 is concerned with the Padé approximants just to the exponential. Theorem 3.4.2 shows that the remainder of the Padé approximant $f_{m,n}$, where m, n are the degrees of the numerator and the denominator, respectively, is $C_{m,n}q^{m+n+1}$. The Padé approximants are ordered in

the so-called *Padé table* where m, n are the row and column indices, respectively. ixPadé table

There are exceptions where $C_{m,n} = 0$, but Hairer and Wanner [21, Eq. (IV.3.30)] show that there are no such exceptions for the exponential.

The conclusion is that $p \leq \bar{p} \leq 2r$ for every RK method.. [42] We shall see below that, for every $r > 0$, there exists a IRK method, based on collocation, such that $p = 2r$, hence $m = n = r$.

One can show that the rational functions on the main diagonal of the Padé table, i.e. if $m = n = r$, are A-acceptable, and the approximants in the first and the second subdiagonals, i.e., $m = n - 1$ and $m = n - 2$, are even L-acceptable. All the other Padé approximants are not even A-acceptable; the approximants above the main diagonal are, of course, not even zero-acceptable. [43]

If a method is *explicit*, then A is subtriangular, hence $A^r = 0$, and we can write

$$(I - qA)^{-1} = \sum_{\nu=0}^{r-1} (qA)^\nu, \qquad R(q) = 1 + \sum_{\nu=1}^{r} b^T A^{\nu-1} \mathbf{1} q^\nu.$$

hence $R(q)$ *is a polynomial of degree* r *for an explicit RK-method.* It follows that $p \leq \bar{p} \leq r$ for explicit RK methods, but it was mentioned above that $p = r$ is achieved for $r \leq 4$ only.

The stability region $\mathcal{S}$ of an explicit RK method is bounded, since $|R(q)| \to \infty$ as $|q| \to \infty$.

13.3.5 Collocation and Order Reduction.

• Preliminary text.

Choose points $t_n + h\theta_i$, $i = 1, 2, \ldots, r$. For $t \in (t_n, t_{n+1})$, define an r'th degree polynomial $\psi(t)$ by the equations,

$$h\psi'(t_n + h\theta_i) = k_i, \quad (i = 1 : r),$$
$$\psi(t_n) = y_n,$$

By Lagrange's interpolation formula we have

$$h\psi'(t_n + h\theta) = \sum_{i=1}^{r} k_i \delta_i(\theta), \tag{13.3.15}$$

where $\delta_i(\theta)$ is the $(r-1)$-degree polynomial uniquely determined by the conditions $\delta_i(\theta_j) = \delta_{ij}$, $j = 1 : r$, well known from the theory of interpolation, although with a different notation. Note that, for any polynomial of degree r or less,

$$\sum_{i=1}^{r} p(c_i)\delta_i(\theta) = p(\theta) \text{ , e.g., } \sum i = 1^r \delta_i(\theta) = 1.$$

[42] Recall that the order of consistency equals the order of the local error minus one.

[43] We hope that this terminology can be understood.

Now integrate (13.3.15) from 0 to θ_i:

$$\psi(t_n + h\theta_i) = y_n + \sum_{j=1}^{r} a_{ij}k_j, \quad \text{where} \quad a_{ij} = \int_0^{\theta_i} \delta_j(\theta)\, d\theta.$$

Now we require that the autonomous differential system $y' = f(y)$ should be satisfied by $\psi(t)$ in the points $t = t_n + h\theta_i$, $i = 1 : r$, which we call *collocation points*.

In fact, this exemplifies a very old and general idea called **collocation** that has numerous applications to partial differential equations and integral equations. We shall meet it again. We shall now see that*here the collocation idea leads to an IRK method*. Hence

$$h\psi'(t_n + h\theta_i) = hf(\psi(t_n + h\theta_i)), \quad i = 1 : r,$$

i.e.

$$k_i = hf(y_n + \sum_{j=1}^{r} a_{ij}k_j). \tag{13.3.16}$$

Now we can construct an implicit RK-method by setting,

$$y_{n+1} = \psi(t_{n+1}), \quad b_i = \int_0^1 \delta_i(\theta)\, d\theta$$

for by the integration of (13.3.15) from 0 to 1, we obtain,

$$y_{n+1} - y_n = \sum_{i=1}^{r} b_i k_i, \tag{13.3.17}$$

which, together with (13.3.16), defines the method, with

$$c_i = \theta_i, \quad i = 1 : r.$$

For the extension of this to non-autonomous systems we need to show that

$$c_i = \sum_{j=1}^{r} a_{ij}.$$

First, we saw above that $\sum_{j=1}^{r} \delta_j(\theta) = 1$, It follows that

$$\sum_{j=1}^{r} a_{ij} = \int_0^{\theta_i} \sum_{j=1}^{r} \delta_j(\theta)\, d\theta = \theta_i = c_i.$$

Under appropriate differentiability assumptions concerning f, we have

$$\psi'(t) = f(t, \psi(t)) + O(h^r), \quad (t_n \le t \le t_{n+1}).$$

One may expect that the accuracy of y_{n+1}, becomes higher, if the θ_i are the abscissae of the *Gauss quadrature* formula over the interval $(0, 1)$. In fact, it can be shown

(see Hairer et al. Theorem II.7.9) that the order of consistency becomes $2r$, which, as mentioned, is the highest attainable order for r-stage RK-methods. This method therefore realizes the diagonal element P_{rr} in the Padé table. Similarly, it can be shown that the order of consistency equals $2r-1$ for methods which are based on the two Radau quadrature formulas with, respectively $\theta_1 = 0$ or $\theta_r = 1$, while the remaining θ are chosen to produce the highest possible order of accuracy. Note that the collocation interpretation is not suitable for all RK-methods.

It is remarkable that the method based on Gauss quadrature is A-stable. This compares very favorably with linear multistep methods, or one-leg methods, where A-stability implies $p \leq 2$. In many other contexts high order of consistency is bought at the price of deteriorated stability properties. The price here is the complexity of the nonlinear system with $r \times s$ unknowns which has to be solved at each step.

• More material is to be added. For the time being the reader is referred to Hairer–Nørsett–Wanner: Sec.II.7 (Collocation Methods), Sec.IV.5 (Gauss...,Radau...,Lobatto...).

The stability function does not define an RK-method uniquely, e.g. the Simpson's rule and the 3/8 rule of Kutta have the same truncated power series as stability function. Butcher, loc. cit,, seems to make a more direct statement about which of the published Gauss etc. rules are derived from collocation.

• Some material will be added about **order reduction**, The demonstration of this phenomenon by means of Prothero–Robinson's inhomogeneous test equation

$$\dot{y} = \lambda(y - \phi(t)) + \dot{\phi}(t), \quad y(t_0) = \phi(t_0), \quad \Re\lambda \leq 0.$$

will be exemplified. A merger of extracts from Tables IV.5.13 and IV.15.1 is to be included.

13.3.6 Miscellaneous about Runge–Kutta Methods

• Very preliminary text about **dense output** for RK methods.

A typical problem with RK-methods is that the steps are rather large and that *the estimates of* $y(t_n + c_j h)$ *at the intermediate stages are usually less accurate than the estimate of* $y(t_{n+1})$. So it is not easy to interpolate the results to the accuracy they deserve. Usually, the *local* errors of the interpolation do not need to be (much) smaller than the *global* errors of the available values.

Dense output is required in many cases, such as graphical output, event location (i.e., finding an intersection with some hyperplane or treating a discontinuity etc.). Another application is to use (say) one or two steps with an accurate RK method for obtaining starting values for a long run with a multistep method. (By the way, multistep methods provide, by their definition, a natural dense output.)

The methods which are derived from collocation are exceptional, though even for them the local accuracy of the collocation polynomials is usually not higher than $O(h^{r+1})$, although some of these methods (Gauss, Radau, etc.) are more accurate at the endpoints of the step.

Since accurate estimates of $y(t)$ as well as $\dot{y}(t)$ are available, it seems natural to apply cubic (or quintic) *Hermite interpolation* (see Ch.$\tilde{4}$) to the computed results in one step (or two adjacent steps].

As an alternative to this, one has, for some RK methods, derived an estimate of the form

$$y(t_n + \theta h) \approx y_n + \sum_{j=1}^{r} b_j(\theta) k_j.$$

For example, for Kutta's Simpson's rule one has found that

$$b_1(\theta) = \theta - \frac{3\theta^2}{2} + \frac{2\theta^3}{3}, \quad b_2(\theta) = b_3(\theta) = \theta^2 - \frac{2\theta^3}{3}, \quad b_4(\theta) = -\frac{\theta^2}{2} + \frac{2\theta^3}{3},$$

provides a dense output solution with local error $O(h)^4$. It is continuous, but the derivative may have $O(h^3)$ discontinuities.

For some methods, additional stages have been added, for providing the requested accuracy, see [20, Sec. II.6].

13.3.7 Introduction to Rosenbrock Methods

• Preliminary text.

The *Rosenbrock methods* can be seen as *linearly implicit* variant of of the DIRK methods, although they are older. The idea is to avoid the non-linear systems that most methods for stiff systems require; they are replaced by linear systems; the traditional Rosenbrock require $J = f'(y_n)$, i.e., the *exact* Jacobian at the beginning of a step.

An r-stage Rosenbrock method is, for a non-autonomous system, given by the formulas

$$k_i = hf\Big(y_n + \sum_{j=1}^{i-1} \alpha_{ij} k_j\Big) + hJ \sum_{j=1}^{i} \gamma_{ij} k_j, \quad i = 1 : s \tag{13.3.18}$$

$$y_{n+1} = y_n + \sum_{j=1}^{r} b_j k_j,$$

where $\alpha_{ij},\ \gamma_{ij},\ b_j$ are the determining coefficients, and $J = f'(y_n)$

In the i'th stage, a linear system is solved with the unknown k_i and with matrix $I - h\gamma_{ii}J$. The analog to the SDIRK methods,i.e., methods with the same γ_{ii} for all i are of particular interest.

The reader is referred to Hairer, Nørsett and Wanner, Sec. IV.7, concerning order conditions, examples and references to test results. You find 6 examples of 4-stage methods of (classical) order 4; 1 of them is L-stable, 3 others are A-stable, and remaining method are very nearly A-stable. A Rosenbrock method is said to be nearly as simple to code as an explicit RK method.

The stability function is the same as for DIRK method where the elements of the tableau matrix A are equal to $\alpha_{i,j} + \gamma_{i,j}$. matrix

Rosenbrock methods that allow an *inexact Jacobian*, also called W-methods, were introduced in 1979 by Steihaug and Wolfbrandt. The number of order conditions grows much faster, and the stability investigation is very complicated, see Hairer et al. loc. cit.

13.3.8 The Taylor-Series Method

• Preliminary text.

In Example 3.1.2 we solved an initial-value problem by substituting a power series $y(x) = \sum_{n=0}^{\infty} c_n x^n$ with undetermined coefficients. ¿From the differential equation a recursion formula for the computation of the coefficients was derived. We now show how this method can be extended to a stepwise process. If the solution is sufficiently smooth we have by Taylor's formula

$$y(x+h) = y(x) + hy'(x) + \frac{h^2}{2}y''(x) + \cdots + \frac{h^p}{p!}y^{(p)}(x) + \frac{h^{p+1}}{(p+1)!}y^{(p+1)}(\xi). \tag{13.3.19}$$

Euler's method can be viewed as an approximation of the first two terms in this expansion. If we can evaluate higher derivatives of y, we can obtain a method of order p by neglecting the remainder term in (13.3.19) and using the formula

$$y_{n+1} = y_n + hy'_n + \frac{h^2}{2}y''_n + \frac{h^3}{3!}y'''_n + \cdots + \frac{h^p}{p!}y_n^{(p)}. \tag{13.3.20}$$

The first neglected term can be used as an estimate of the local discretization error.

Following Euler we express the derivatives of $y(x)$ in terms of the partial derivatives of $f(x, y)$. Starting with the differential equation $y'(x) = f(x, y(x))$ we differentiate both sides with respect to x to obtain

$$y'' = f_x + f_y y' = f_x + f_y f, \tag{13.3.21}$$

where we have used the notation $f_x = \partial f/\partial x$ and $f_y = \partial f/\partial y$. Differentiating again we obtain

$$y''' = f_{xx} + f_{xy}y' + f_y(f_x + f_y f) + (f_{xy} + f_{yy}y')f \tag{13.3.22}$$

$$= (f_{xx} + 2f_{xy}f + f^2 f_{yy}) + (f_x + f_y f)f_y. \tag{13.3.23}$$

For higher derivatives the formulas soon become very complicated. For an autonomous system $f_x = f_{xy} = f_{xx} = \ldots = 0$, and the formula simplifies considerably. On the other hand, for systems y and f are vectors, and so f_x is a vector, f_y a matrix, etc.

If $f(x, y)$ is composed of elementary functions it is often possible to obtain simple recursion formulas for the successive derivatives using an extension of Newton's series approach. If we introduce the **Taylor coefficients** of $y(x)$ and $f(x, y(x))$ at x_n

$$Y_i = \frac{1}{i!}y_n^{(i)}, \qquad F_i = \frac{1}{i!}f(x, y(x))_n^{(i)},$$

we can write (13.3.20)

$$y_{n+1} = \sum_{i=0}^{p} h^i Y_i.$$

Differentiating $y' = f(x, y(x))$ we get the relation

$$(i+1)Y_{i+1} = F_i, \qquad i = 0, 1, 2, \ldots.$$

If $f(x, y)$ is an algebraic composition of elementary functions we can find formulas for recursively generating the Taylor coefficients F_i and Y_i. We have, for example,

$$f = p \pm q \Longrightarrow F_i = P_i + Q_i,$$

where P_i and Q_i are the Taylor coefficients of p and q. Similarly, by the Cauchy formula

$$f = pq \Longrightarrow F_i = \sum_{j=0}^{i} P_j Q_{i-j}.$$

For the Taylor coefficients F_i of $f = p/q$ we write $p = fq$, and use the Cauchy formula $P_i = \sum_{j=0}^{i} F_j Q_{i-j}$. Solving for F_i w get

$$F_i = \frac{1}{Q_0}\Big(P_i - \sum_{j=0}^{i-1} F_j Q_{i-j}\Big).$$

which is a recursion formula for F_i. Recursion formulas can also be derived for the Taylor coefficients of many elementary functions, see Hairer et al. [1987].

Example 13.3.5

Determine the first six Taylor coefficients for the function which is the solution to the initial value problem

$$y' = 1 + xy + y^2.$$

Using the formulas above for the Taylor coefficients of a sum and product leads to the recursion $Y_0 = y(x_n)$, $Y_1 = F_0 = f(x_n, y_n)$,

$$(i+1)Y_{i+1} = F_i = Y_{i-1} + X_0 Y_i + \sum_{j=0}^{i} Y_j Y_{i-j}, \qquad i = 1, 2, 3, \ldots,$$

where $X_0 = x_n$. In particular if we take $x_n = 0$, $y(0) = 0$, we obtain $Y_0 = 0$, $Y_1 = 1$,

$$\begin{aligned}
2Y_2 &= Y_0 + X_0 Y_1 + 2Y_0 Y_1 \quad &\Longrightarrow Y_2 = 0,\\
3Y_3 &= Y_1 + X_0 Y_2 + 2Y_0 Y_2 + (Y_1)^2 \quad &\Longrightarrow Y_3 = 2/3,\\
4Y_4 &= Y_2 + X_0 Y_3 + 2(Y_0 Y_3 + Y_1 Y_2) \quad &\Longrightarrow Y_4 = 1/6 = 0,\\
5Y_5 &= Y_3 + X_0 Y_4 + 2(Y_0 Y_4 + Y_1 Y_3) + (Y_2)^2 \quad &\Longrightarrow Y_5 = 3/2 = 0.
\end{aligned}$$

Thus

$$y(h) = h + \frac{2}{3}h^3 + \frac{1}{6}h^4 + \frac{3}{2}h^5 + \ldots.$$

Notice that the computation of the recursion formulas for the Taylor coefficients need only be done once. The same recursion formulas can be used at each step (the numerical values of the Taylor coefficients are of course different at each step).

The Taylor series method was used a great deal in hand computation, but was less popular during the first years of the computer age. Since then, programming techniques and languages have been improved, and the popularity of the method has risen again. If $f(x, y)$ is the composition of a sequence of algebraic operations and elementary functions it is easy to write subroutines, which recursively compute the Taylor coefficients. There exist programs which automatically generate such subroutines from a Fortran statement for $f(x, y)$.

It is difficult to make a general comparison between the efficiency of Taylor series methods and the methods previously mentioned. An advantage of the Taylor series approach is that the order of the method and the step size can be varied simply in order to get the required precision. It can also be executed in interval analysis and developed to provide reliable error bounds, see Volume I, Sec. 2.5.3 and Moore [28].

• I intend to add more about the techniques used by D. Barton, I. M. Willers, and R. V. M. Zahar, see Rice [2] Also something will probably be added about the use af automatic differentiation and interval analysis.

Other types of series expansions have also been used with success in differential equation problems–for example, Chebyshev series, see Fox and Parker [1968].

13.3.9 Rosenbrock Methods

Review Questions

1.

Very preliminary. Many important types of problems are still missing.

Run a few of the computer exercises of the two previous sections with an established Runge–Kutta code for non-stiff problems. Choose in particular problems, where your experience with the 2nd order RK method was unsatisfactory.

Problems

1. Use the classical RK-fourth order method to compute an approximation to $y(0.2)$, where $y(x)$ is the solution to the differential equation $y' = x + y$ with $y(0) = 1$. Compute with six decimals, for two different step sizes, $h = 0.2$ and $h = 0.1$. Extrapolate. Compare with the exact result.

2. Determine the order of the two RK-methods

$$a) \quad y_{n+1} = y_n + k_2, \qquad \hat{y}_{n+1} = y_n + \frac{1}{6}(k_1 + 4k_2 + k_3),$$

where

$$k_1 = hf(t_n, y_n),$$

$$k_2 = hf(t_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1),$$
$$k_3 = hf(t_n + h, y_n - k_1 + 2k_2).$$

3. We want to compute values of the function

$$y(x) = \int_0^\infty \frac{e^{-t^2}}{t+x} dt$$

for a sequence of values of $x > 1$. We can proceed in the following way: $y(x)$ is computed for $x = 1$ using some method for numerical integration; one finds $y(1) = 0.6051$. Show that y satisfies the differential equation

$$y' + 2xy = -1/x + \sqrt{\pi}.$$

By solving the differential equation numerically with initial value $y(1) = 0.6051$, more values can be computed. Determine $y(1.4)$ by means of Kutta's Simpsons rule with $h = 0.2$ and $h = 0.4$, and perform Richardson extrapolation.

4.

5. *A Lipschitz constant for the increment function of an RK-method.* Consider the application of the RK-method defined the tableau (13.3.2'), where the matrix A is not necessarily triangular, to an autonomous system $\dot{y} = f(y)$ with two different input vectors $y_0,\ y_0 + \delta y_0$. Let $k_i,\ k_i + \delta k_i$ be the corresponding intermediate vectors, and let $y_1,\ y_1 + \delta y_i$ be the output vectors. Denote by L a Lipschitz constant for f.
Set $|A| = [|a_i j|], \quad |b|^T = [|b|_1, |b|_2, \ldots, |b|_r], \delta\mathbf{k} = [\|k_1 - l_1\|,\ \|k_2 - l_2\|, \ldots, \|k_r - l_r\|]^T$.
Show that, with componentwise inequalities,

$$\delta\mathbf{k} \leq hL\|\delta y_0\|\mathbf{1} + hL|A|\delta\mathbf{k}$$

If the spectral radius of $hL|A|$ is less than unity, show that $L_h = L|b|^T(I - hL|A|)^{-1}\mathbf{1}$ is a Lipschitz constant for the increment function of the RK-method.
Apply this to obtain an alternative derivation of Theorem 13.3.2, with the condition $\|f'(y)\| \leq L$ instead of $\mu(f'(y)) \leq \mu^*$, and with C instead of C'.

6. (Newton (1671). Derive the Taylor series approximation up to terms of order h^6 for the initial value problem

$$y' = 1 - 3x + y + x^2 + xy, \qquad y(0) = 0.$$

7. Determine a Taylor series expansion for the solution of the equation $y' = y^2$, $y(0) = 1$, about $x = 0$. Use this approximation to compute y for $x = 0.2$ and $x = 1.2$ to four decimals. Compare with the exact solution,and explain why the second case ($x = 1.2$) was unsuccessful.

Notes and References

13.4 Multistep Methods

One-step methods only use information from the previous point (x_n, y_n) to compute the approximation of y_{n+1}. In contrast in multistep formulas we assume that we know approximations $y_n, y_{n-1}, \ldots, y_{n-k+1}$ to the exact solution at the k points x_{n-j}, $j = 0, 1, \ldots, k-1$.

A **general linear multistep** method for the differential equation $y' = f(x, y)$, $y(0) = y_0$, is defined by the difference equation

$$\sum_{i=0}^{k} (\alpha_i y_{n+i} - h\beta_i f_{n+i}) = 0, \tag{13.4.1}$$

where α_i and β_i are real parameters, h the step length and $f_i = f(x_i, y_i)$. The formula (13.4.1) is also called a linear k-step method. This class includes all the previously considered classical linear multistep methods, in particular the midpoint method, Euler's method and the trapezoidal method. The y_n can be computed recursively from (13.4.1) if in addition to the initial value y_0, $k-1$ more values $y_1, \ldots, y_{k-1}$ are given. If $\beta_k \neq 0$ the method is **implicit**, and then this may be true only for sufficiently small h.

13.4.1 The Adams Methods

An important class of linear multistep methods dates back to work by Adams about 1855. Following Adams we consider the integrated form of the first order differential equation (13.1.1)

$$y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} f(t, y(t))dt.$$

We now replace the function $f(t, y(t))$ in the integral by the polynomial $p_{k-1}(t)$ of degree $k-1$ interpolating the values

$$(x_i, f_i), \qquad i = n-k+1, \ldots, n,$$

where $f_i = f(x_i, y_i)$. Assume that the points are equidistant, $x_i = x_0 + ih$, and use Newton's interpolation formula for equidistant interpolation (see (3.4.9) we can write this polynomial

$$p_{k-1}(t) = p_{k-1}(x_n + sh) = \sum_{j=0}^{k-1} (-1)^j \binom{-s}{j} \nabla^j f_n,$$

where ∇ denotes the backward difference operator, see Sec. 3.2.1. Inserting $p_{k-1}(t)$ and integrating we get the numerical formula

$$y_{n+1} = y_n + h \sum_{j=0}^{k-1} \gamma_j \nabla^j f_n, \qquad \gamma_j = (-1)^j \int_0^1 \binom{-s}{j} ds.$$

Note that the coefficients γ_j do not depend on the order k. Inserting numerical values for we get the family of **explicit Adams methods** of increasing order

$$y_{n+1} - y_n = h\Big(1 + \frac{1}{2}\nabla + \frac{5}{12}\nabla^2 + \frac{3}{8}\nabla^3 + \frac{251}{720}\nabla^4 + \dots\Big) f_n. \tag{13.4.2}$$

The backward differences can be expressed in function values using $\nabla^j f_n = (1 - E^{-1})^j f_n$, and we obtain in particular for $k = 1, 2, 3, 4$ the methods of order up to $p = 4$.

$$\begin{aligned}
y_{n+1} &= y_n + hf_n;\\
y_{n+1} &= y_n + \frac{h}{2}(3f_n - f_{n-1});\\
y_{n+1} &= y_n + \frac{h}{12}(23f_n - 16f_{n-1} + 5f_{n-2});\\
y_{n+1} &= y_n + \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3});
\end{aligned}$$

The first formula here is the explicit Euler method. An attractive feature of these methods is that independent of order k only one evaluation of the function f is needed at each step.

By interpolating also the point (x_{n+1}, f_{n+1}) we obtain a family of implicit formulas. In the equidistant case the corresponding interpolation polynomial equals

$$p_k^*(t) = p_k^*(x_n + sh) = \sum_{j=0}^{k} (-1)^j \binom{-s+1}{j} \nabla^j f_{n+1}.$$

Inserting this into the integral we obtain

$$y_{n+1} = y_n + h\sum_{j=0}^{k} \gamma_j^* \nabla^j f_{n+1}, \qquad \gamma_j^* = (-1)^j \int_0^1 \binom{-s+1}{j} ds,$$

which gives the family of **implicit Adams methods** of increasing order $p = k + 1$

$$y_{n+1} - y_n = h\Big(1 - \frac{1}{2}\nabla - \frac{1}{12}\nabla^2 - \frac{1}{24}\nabla^3 - \frac{19}{720}\nabla^4 - \frac{3}{160}\nabla^5 - \dots\Big) f_{n+1}. \tag{13.4.3}$$

Expressed in function values the methods up to order $p = 5$ are

$$\begin{aligned}
y_{n+1} &= y_n + hf_{n+1};\\
y_{n+1} &= y_n + \frac{h}{2}(f_{n+1} + f_n);\\
y_{n+1} &= y_n + \frac{h}{12}(5f_{n+1} + 8f_n - f_{n-1});\\
y_{n+1} &= y_n + \frac{h}{24}(9f_{n+1} + 19f_n - 5f_{n-1} + f_{n-2});\\
y_{n+1} &= y_n + \frac{h}{720}(251f_{n+1} + 646f_n - 264f_{n-1} + 106f_{n-2} - 19f_{n-3});
\end{aligned}$$

The first two formulas here are the implicit Euler method, and the trapezoidal method, respectively. For an operator derivation of the Adams methods see Problem 9 of Section 3.2.

The local errors in the Adams methods are approximately equal to the first neglected term in the series. Note that the coefficients γ_j^* decrease much faster than γ_j, and hence the implicit Adams methods have smaller local error.

Multistep methods need a special starting procedure to compute the $k-1$ extra starting values $y_{k-1}, \ldots, y_1$. These can be obtained, e.g., by using a Runge–Kutta method of appropriate order. However, the usual practice is to simply start with a multistep method of order one and a very small step size, and then successively increase the order and step size. We discuss this in greater depth in Sec. 13.4.4.

If f is a nonlinear function, then using the implicit Adams methods one has to solve a nonlinear system at each step. This can be done by fixed point iteration. We write

$$y_{n+1} = h\beta_k f_{n+1} + u_n, \qquad u_n = y_n + h\Big(\beta_{k-1} f_n + \ldots + \beta_0 f_{n-k+1}\Big),$$

where u_n is known and iterate

$$y_{n+1}^{(m+1)} = h\beta_k f(x_{n+1}, y_{n+1}^{(m)}) + u_n, \qquad m = 0, 1, 2, \ldots. \tag{13.4.4}$$

If the step size h is small enough $y_{n+1}^{(m+1)}$ converges to the solution y_{n+1} of the implicit formula. A sufficient condition for convergence is

$$h\beta_k \left\| \frac{\partial f}{\partial y} \right\| < 1. \tag{13.4.5}$$

A good initial approximation for the implicit Adams method of order $p+1$ can be obtained by using the explicit Adams method of order p. The explicit formula is called a **predictor**, while the implicit formula is called a **corrector**. The whole procedure is called a **predictor-corrector** method.

The stopping of the iterations may be controlled by comparing the difference $y_{n+1}^{(m+1)} - y_{n+1}^{(m)}$ to some preset tolerance. In this case usually a maximum of three iterations are allowed. Another possibility is to use a predetermined number of iterations. The latter is more common, and it is advisable to choose a step length such that one iteration will suffice. These codes usually recompute the function value $f(x_{n+1}, y_{n+1}^{(1)})$ (note that this value is needed for the next step) and hence use two function evaluation per step.

More generally we could consider the integral equation

$$y(x_{n+1}) = y(x_{n-i}) + \int_{x_{n-i}}^{x_{n+1}} f(t, y(t))dt.$$

Taking $i = 0$ we get the Adams methods. With $i = 1$ we obtain the **Nyström methods**. Inserting the polynomial $p_{k-1}(t)$ and integrating we obtain the explicit methods

$$y_{n+1} = y_{n-1} + h\sum_{j=0}^{k-1} \kappa_j \nabla^j f_n,$$

or

$$y_{n+1} = y_{n-1} + h\Big(2f_{n+1} + \frac{1}{3}\nabla^2 f_n - \frac{1}{3}\nabla^3 f_n - \frac{29}{90}\nabla^4 f_n + \ldots\Big).$$

The special case

$$y_{n+1} = y_{n-1} + 2hf_n,$$

is the **explicit midpoint** method which we recognize from Sec. 13.3.3. Implicit Nyström methods can be similarly derived. The Nyström methods are in general not useful since they suffer from a weak instability as exemplified by the explicit midpoint method.

13.4.2 Local Error and Order Conditions

With the multistep method (13.4.1) we associate the linear difference operator

$$\mathcal{L}_h y(x) = \sum_{i=0}^{k} (\alpha_i y(x+ih) - h\beta_i y'(x+ih)), \tag{13.4.6}$$

where $y(x)$ is an arbitrary function, continuously differentiable on an interval that contains the values $x+ih$ for $i = 0, 1, \ldots, k$. We say that the **order of consistency** is p, if p is the largest integer such that $\mathcal{L}_h P(x)$ vanishes identically for any pth degree polynomial. An equivalent definition is:

Definition 13.4.1.

The method (13.4.1) is said to be of **order** *p, if for all y with continuous derivatives of order $p+1$ it holds that*

$$\mathcal{L}_h y(x) \sim c_{p+1} h^{p+1} y^{(p+1)}(x), \quad h \to 0. \tag{13.4.7}$$

If $p \geq 1$, the multistep method is said to be **consistent**.

Expanding $y(x+ih)$ and its derivative $y'(x+ih)$ in Taylor series about x, inserting these into (13.4.6), and collecting terms gives

$$\mathcal{L}_h y(x) = c_0 y(x) + c_1 h y'(x) + \ldots + c_p h^p y^{(p)}(x) + \ldots,$$

where $c_0, c_1, \ldots, c_p, \ldots$ are constants. The order of the method is given by the first non-vanishing term in this expansion. Hence, the value of p and the constant c_{p+1} can be determined by using as test functions the polynomials $x^q/q!$, $q = 1, \ldots, p+1$, which leads to the following order conditions:

Theorem 13.4.2.

The multistep method (13.4.1) is of order p, if and only if, the following conditions are satisfied:

$$\sum_{i=0}^{k} \alpha_i = 0, \qquad \sum_{i=0}^{k} \alpha_i i^q - q \sum_{i=0}^{k} \beta_i i^{q-1} = 0, \quad q = 1, \ldots, p. \tag{13.4.8}$$

The constant c_{p+1} is given by

$$c_{p+1} = \frac{1}{(p+1)!}\Big(\sum_{i=0}^{k} \alpha_i i^{p+1} - (p+1)\sum_{i=0}^{k} \beta_i i^p\Big) \neq 0. \tag{13.4.9}$$

Definition 13.4.3.
By the local truncation error of a multistep method (13.4.1) at x_{n+k} we mean the error $y(x_{n+k}) - y_{n+k}$, where $y(x)$ is the exact solution of $y' = f(x, y)$, $y(x_n) = y_n$, and y_{n+k} is the numerical solution obtained from (13.4.1) by using the exact starting values $y_i = y(x_i)$ for $i = n, n+1, \ldots, n+k-1$.

For $k = 1$ this definition coincides with the definition of the local error for one-step methods. The local error is essentially equal to $\alpha_k^{-1}\mathcal{L}_h y(x)$, see Hairer et al. [20, Chap. III.2].

The **generating polynomials**

$$\rho(\zeta) = \sum_{i=0}^{k} \alpha_i \zeta^i, \qquad \sigma(\zeta) = \sum_{i=0}^{k} \beta_i \zeta^i, \tag{13.4.10}$$

play a fundamental role in the theory of multistep methods. We have

$$\mathcal{L}_h e^x = (\rho(e^h) - h\sigma(e^h))e^x.$$

Hence, the method is of order p if and only if

$$\rho(e^h) - h\sigma(e^h) \sim ch^{p+1}, \qquad h \to 0. \tag{13.4.11}$$

In particular, consistency is easily shown to be equivalent to the equations

$$\rho(1) = 0, \qquad \rho'(1) = \sigma(1).$$

To compare the local errors of multistep methods of the same order we could use the constant c_{p+1} in (13.4.9). However, that is not a suitable measure of accuracy, since multiplication of (13.4.1) by a constant will change c_{p+1}. It can be shown that a more relevant measure is

$$C = c_{p+1}/\sigma(1), \qquad \sigma(1) = \sum_{i=0}^{k} \beta_i, \tag{13.4.12}$$

which is called the **error constant** of the method (13.4.1).

Example 13.4.4
For the Adams methods we have

$$\rho(\zeta) = \zeta^k - \zeta^{k-1}, \qquad \rho'(\zeta) = k\zeta^{k-1} - (k-1)\zeta^{k-2},$$

and hence consistency gives $\rho'(1) = 1 = \sigma(1)$. It follows that for these methods $C = c_{p+1}$.

13.4.3 Linear Stability Theory

In Sec. 13.2.2 we used the simple differential equation problem

$$y' = \lambda y, \qquad y(0) = 1, \tag{13.4.13}$$

where λ is a complex constant, as a test problem for studying the stability of numerical methods for initial value problems. The stability region $\mathcal{S}$ of a numerical method was defined in Def. 13.1.17 as the set of complex values of $q = \lambda h$ for which all solutions y_n of the test problem (13.4.8) remain bounded as $n \to \infty$. If $\mathcal{S}$ contains the origin, the method is zero stable.

A linear multistep method is zero-stable if and only if all solutions of the difference equation

$$\sum_{i=0}^{k} \alpha_i y_{n+i} = 0 \tag{13.4.14}$$

are bounded for all positive n. The solution y_n can be interpreted as the numerical solution for the differential equation $y' = 0$. We find that $y_n = \zeta_j^n$ satisfies (13.4.14) if and only if ζ is a root of $\rho(\zeta)$ defined in (13.4.10). Further, if ζ_j has multiplicity $m_j > 1$, then a solution is $y_n = p_j(n)\zeta_j^n$, where $p_j(n)$ is a polynomial of degree $m_j - 1$. Thus we have the following result:

Theorem 13.4.5.

Necessary and sufficient for stability of the linear multistep method (13.4.1) are the following **root conditions***:*

i. All roots of $\rho(\zeta)$ should be located inside or on the unit circle $|z| \le 1$;

ii. The roots on the unit circle should be simple.

Example 13.4.6

For the explicit and implicit Adams methods $\rho(\zeta) = \zeta^k - \zeta^{k-1}$, and besides the simple root $\zeta = 1$ there is a root $\zeta = 0$ of multiplicity $k - 1$. Note that by the consistency condition $\rho(1) = 0$, there is always one root equal to 1. For the Adams methods all the other roots are at the origin.

The relevance of the stability concept defined above is shown by the following theorem, which summarizes several theorems proved in Henrici [22, Chap. 5]. A hint to a proof is given by Problems 5 and 6 (c) of this section.

Theorem 13.4.7.

Suppose that $y(x)$ is a $p+1$ times differentiable solution of the initial-value problem, $\dot{y} = f(x, y)$, $y(0) = y_0$, $p \ge 1$, $\|y^{(p+1)}(x)\| \le K_0$, and that $f(x, y)$ is differentiable for all x, y. Suppose further that $\{y_n\}$ is defined by the equations

$$y_n = y(x_n) + \epsilon_n, \qquad n = 0 : k-1,$$

$$\sum_{i=0}^{k}(\alpha_i y_{n+i} - h\beta_i f(x_{n+i}, y_{n+i})) = \epsilon_n, \qquad k \le n+k \le (b-a)/h.$$

If the multistep is stable and satisfies (13.4.7), then there exist constants K_1, K_2, h_0 such that for all $x_n \in [a,b]$, $h \le h_0$,

$$\|y_n - y(x_n)\| \le \Big(c_{p+1}h^p(x_n - a)K_0 + \sum_{i=0}^{n}\|\epsilon_i\|\Big)K_1 e^{K_2(x_n-a)}. \tag{13.4.15}$$

K_1 depends only on the coefficients of the method, while K_2 also contains an upper bound for $\|\partial f/\partial y\|$.

In view of this result, the integer p is called the **order of accuracy** of the method.

It is sufficient to consider the trivial case $f(x,y)$ constant in order to show that stability and consistency are *necessary* for such a result, with $p > 0$. A corollary of the theorem and this remark in a more precise formulation is that

$$\text{Consistency} + \text{Stability} \iff \text{Convergence}$$

Convergence here includes uniform convergence in $[a,b]$, when $h \to 0$, for all f which satisfy the assumptions made in Sec. 13.1.1, as well as a requirement that the effect of perturbations of the initial values should tend to zero when the perturbations do so themselves. The formulation given above occurs in numerous other applications of finite-difference methods to ordinary and partial differential equations where these three concepts are defined appropriately for each problem area. "Consistency" usually means that the difference equation formally converges to the differential equation as $h \to 0$, while "convergence" is related to the behavior of *the solutions* of the difference equations.

If the termination of the iteration in an implicit multistep method is controlled by a tolerance on the residual, then an error bound can be obtained by Theorem 13.4.7. If a predictor-corrector technique with a fixed number of iterations is used, then this theorem does not guarantee that the $\|\epsilon_n\|$ do not grow. The stability of such a scheme can be different from the stability of the corrector method.

Predictor-corrector methods are considered in an extension of the multistep methods, named **multivalue methods**; see Gear [18, Chap. 9]. This extension is important in several other respects, but it is beyond the scope of this presentation.

It is important to distinguish between the stability question of the differential equation (see Figs. 13.1.2a and 13.1.2b) and the stability questions for the numerical methods. The former is well-conditioned—e.g., if $e^{L(b-a)}$ is of moderate size, where L is an upper bound for the logarithmic norm, defined in Sec. 13.1.4. Compare the general distinction between an ill-conditioned problem and an unstable algorithm in Sec. 2.4.5.

For a multistep method to be of order p it has to satisfy the $p+1$ order conditions (13.4.8). For a k-step method we have $2k+1$ free parameters, if we scale the coefficients so that $\alpha_k = 1$. It therefore seems that order $p = 2k$ should be possible to attain. However, because there is a *conflict between consistency and*

stability these methods are not zero-stable are therefore not of practical interest. Stability requirements impose the following restrictions on the attainable order of multistep methods, the so called "first Dahlquist-barrier":

Theorem 13.4.8.
The order p of a zero-stable linear k-step method satisfies:

$$p \leq \begin{cases} k+2, & \text{if } k \text{ is even;} \\ k+1, & \text{if } k \text{ is odd;} \\ k, & \text{if } \beta_k/\alpha_k \leq 0 \text{ (if the method is explicit);} \end{cases}$$

A zero-stable method with $p = k+2$ is called an **optimal** method. An example of an optimal method is the **Milne–Simpson method**

$$y_{n+2} = y_n + h\frac{1}{3}(f_{n+2} + 4f_{n+1} + f_n),$$

for which $p = 4$, $k = 2$. However, the Milne–Simpson method like all optimal methods is only weakly stable and may show an exponential error growth of the type illustrated for explicit midpoint method. This severely limits its use as a general purpose method.

Example 13.4.9
The method

$$y_{n+2} = -4y_{n+1} + 5y_n + h(4f_{n+1} + 2f_n)$$

is the only explicit 2-step method with order $p = 3$. The characteristic equation

$$\rho(\zeta) = \zeta^2 + 4\zeta - 5 = 0$$

has the two roots $\zeta_1 = 1$ and $\zeta_2 = -5$, and hence is not zero-stable.

The midpoint method exemplifies that K_2 may be positive even though L is negative in (13.4.15). Hence if $b-1$ is large, the error bound of Theorem 13.4.7 (as well as the actual error) can be large unless h and the perturbation level are very small. Therefore, *the stability concept just defined is not always sufficient.* This is true in particular for stiff problems, see Sec. 13.3.4.

13.4.4 Variable Step and Order

For efficiency it is necessary to vary the step size and order used with multistep methods during the integration. To change the order in the family of Adams methods is simple. We can increase or decrease the order one step by just adding or deleting a term in the formulas in Eqs. (13.4.3) and (13.4.5). Since the number of function evaluations per step is independent of the order, the order can be chosen such that the new step size is maximal, consistent with the local error criterion.

Changing the step size, on the other hand, is not as simple as with one step methods. In the derivation of the classical multistep methods in Sec. 13.4.1 it was assumed that numerical approximations $y_n, \ldots, y_{n-k+1}$ are available at equidistant points $x_{n-j} = x_n - jh$, $j = 0 : k-1$. One possibility, used already by Adams, is to use interpolation to reconstruct initial values on an equidistant net, whenever the step size is changed. This technique is no longer used, since the resulting formulas are not very stable even in case the change is restricted by

$$\frac{1}{2} = \omega \le h_n/h_{n-1} \le \Omega = 2, \qquad h_n = x_{n+1} - x_n.$$

Instead we now outline how to directly derive Adams methods for variable step sizes. We now use Newton's general interpolation method, which can be written

$$\begin{aligned} p(t) &= \sum_{j=0}^{k-1} \left(\prod_{i=0}^{j-1} (t - x_{n-i}) \right) f[x_n, x_{n-1}, \ldots, x_{n-j}] \\ &= \sum_{j=0}^{k-1} \left(\prod_{i=0}^{j-1} \frac{t - x_{n-i}}{x_{n+1} - x_{n-i}} \right) \Phi_j^*(n), \end{aligned}$$

where we have introduced the scaled divided differences

$$\Phi_j^*(n) = \left(\prod_{i=0}^{j-1} (x_{n+1} - x_{n-i}) \right) f[x_n, x_{n-1}, \ldots, x_{n-j}].$$

The explicit Adams method can then be written

$$y_{n+1} = y_n + \int_{x_n}^{x_{n+1}} p(t)dt = h_n \sum_{j=0}^{k-1} g_j(n) \Phi_j^*(n),$$

where

$$g_j(n) = \frac{1}{h_n} \int_{x_n}^{x_{n+1}} \prod_{i=0}^{j-1} \frac{t - x_{n-i}}{x_{n+1} - x_{n-i}} dt.$$

It is fairly easy to see that recursion formulas can be developed for computing the scaled divided differences $\Phi_j^*(n)$. Recursion formulas can also be derived for $g_j(n)$, although these are more complicated, see Hairer et al. [1987, III.5]. The cost of computing these integration coefficients is the biggest disadvantage to permitting arbitrary variations in step size for the Adams methods.

Formulas for the implicit Adams methods can be similarly developed. Here

$$y_{n+1} = y_n + \int_{x_n}^{x_{n+1}} p^*(t)dt,$$

where

$$p^*(t) = p(t) + \prod_{i=0}^{k-1} (t - x_{n-i}) f[x_{n+1}, x_n, \ldots, x_{n-k+1}].$$

Hence,

$$y_{n+1} = y_{n+1}^{(p)} + h_n g_k(n)\Phi_k(n+1),$$

where $y_{n+1}^{(p)}$ is the value predicted by the explicit Adams method, and

$$\Phi_k(n+1) = \prod_{i=0}^{k-1}(x_{n+1} - x_{n-i})f[x_{n+1}, x_n, \ldots, x_{n-k+1}].$$

13.4.5 Backward Differentiation Methods

The multistep methods derived above were all based on numerical integration. We now derive a formula based instead on numerical differentiation. Let $q(x)$ be a polynomial which interpolates the values y_i, $i = n-k+1 : n+1$. Then we can write

$$q_k(t) = q_k(x_n + sh) = \sum_{j=0}^{k}(-1)^j \binom{-s+1}{k} \nabla^j y_{n+1}.$$

To determine y_{n+1} we require that

$$q'(x_{n+1}) = f(x_{n+1}, y_{n+1}).$$

This leads to the implicit **backward differentiation formulas**

$$\sum_{j=1}^{k} \delta_j^* \nabla^j y_{n+1} = h f_{n+1}, \qquad \delta_j^* = (-1)^j \frac{d}{ds}\binom{-s+1}{j}|_{s=1} = \frac{1}{j}.$$

These methods can also be derived using the formula

$$hD = -\ln(1-\nabla) = \sum_{j=1}^{\infty} \frac{1}{j}\nabla^j,$$

see the table in Sec. 3.2.2.

The BDF family of multistep methods

$$\nabla y_{n+1} + \frac{1}{2}\nabla^2 y_{n+1} + \frac{1}{3}\nabla^3 y_{n+1} + \ldots + \frac{1}{k}\nabla^k y_{n+1} = h f(x_{n+1}, y_{n+1}) \qquad (13.4.16)$$

has been used with success on stiff differential equations. In particular, for $k = 1, 2$, we obtain

$$\begin{aligned} y_{n+1} - y_n &= h f(x_{n+1}, y_{n+1}), \\ \frac{3}{2}y_{n+1} - 2y_n + \frac{1}{2}y_{n-1} &= h f(x_{n+1}, y_{n+1}). \end{aligned}$$

For $k = 1$ we recognize the implicit Euler method. As k increases the local truncation error decreases, but the stability properties become worse. For $k > 6$ these methods are unstable and therefore useless.

The fixed point iteration in (13.4.4), however, is no good on stiff problems, since the convergence condition (13.4.5), means exactly that the fastest components limit the rate of convergence and thus also the step size which can be used. This is unacceptable, and instead one uses some *modification of Newton's method*. For example, for the BDF methods we have to solve a system of equations of the form

$$F(y_{n+1}) = y_{n+1} - h\beta_0 f(x_{n+1}, y_{n+1}) - \phi_n = 0, \tag{13.4.17}$$

where ϕ_n is a constant that groups the terms from the previous points. Newton's method for this system is

$$(I - hJ)(y_{n+1}^{m+1} - y_{n+1}^{m}) = -y_{n+1}^{m} + h\beta_0 f(x_{n+1}, y_{n+1}^{m}) + \phi_n = 0,$$

where

$$J = \Big(\frac{\partial f}{\partial y}\Big)_{n+1}.$$

Here to compute the Jacobian matrix $J \in \mathbf{R}^{s\times s}$ we must evaluate s^2 partial derivatives $\partial f_i/\partial y_j$, $1 \le i, j \le s$, which often can be costly.

The BDF methods generalize more easily to variable step size than the Adams methods. The interpolation polynomial $q(t)$ of degree k that interpolates (x_i, y_i) for $i = n+1, n, \ldots, n-k+1$ can now be written using divided differences

$$q(t) = \sum_{j=0}^{k} \prod_{i=0}^{j-1} (t - x_{n-i+1}) y[x_{n+1}, \ldots, x_{n-j+1}].$$

Differentiating with respect to t and putting $t = x_{n+1}$ we get

$$q'(x_{n+1}) = \sum_{j=1}^{k} \prod_{i=1}^{j-1} (x_{n+1} - x_{n-i+1}) y[x_{n+1}, \ldots, x_{n-j+1}] = f(x_{n+1}, y_{n+1}).$$

13.4.6 Differential-Algebraic Systems

(To be written.)

Problems

1. (a) Show that for coefficients in the Adams methods it holds that

$$\gamma_0^* = \gamma_0, \qquad \sum_{i=0}^{k} \gamma_i^* = \gamma_k\,.$$

(b) Show by induction using the result in (a) that (13.4.4) can be written as

$$y_{n+1} = y_{n+1}^{(p)} + h\gamma_k \nabla^k f_{n+1},$$

where $y_{n+1}^{(p)}$ is the value obtained by the predictor.

2. Determine $\mathcal{S}$ of:

(a) the Euler method $y_{n+1} = y_n + hf(x_n, y_n)$;

(b) the backward Euler method $y_{n+1} = y_n + hf(x_{n+1}, y_{n+1})$;

(c) What is the order of accuracy of the latter method?

(d) Are the methods stable?

3. (a) Design a third order method for the solution of a differential equation $y' = f(x, y)$ based on the explicit Adams formulas (13.4.3). Apply it to $y' = y^2$, $h = 0.1$, and compute y_3 and y_4 when

$$y_0 = 1.0000, \qquad y_1 = 1.1111, \qquad y_2 = 1.2500$$

are given. Use four decimals. Compare with the exact results.

(b) Improve the value of y_3 using an implicit Adams formula of (13.4.3) truncated after the second term. Go on from this value, compute y_4 using the explicit formula, compute f_4, and improve y_4 using the implicit method (predictor-corrector technique). Improve y_4 by another iteration.

(c) What is the approximate convergence rate for the iterations in (b)?

4. Show that β_1 and β_2 can be determined so that $p = 3$ for the multistep method

$$y_{n+1} = -4y_n + 5y_{n-1} + h\big(\beta_1 f(x_n, y_n) + \beta_2 f(x_{n-1}, y_{n-1})\big).$$

Calculate the error constant c_{p+1} in equation (13.4.9).

5. The explicit midpoint method applied to the test equation gives the difference equation

$$y_{n+1} = y_{n-1} + 2hqy_n.$$

(a) Show that the characteristic equation has the two roots

$$u_1 = z + (1 + z^2)^{1/2}, \qquad u_2 = -1/u_1, \qquad (z = hq).$$

(b) Show that for $x = nh$, $nz = nhq = qx$,

$$u_1^n = e^{qx(1 - z^2/6 + O(z^3))},$$

so that if $|z| \ll 1$ then u_1^n is close to the correct solution e^{qx}.

(c) Show that

$$u_2^n = (-1)^n u_1^{-n} = (-1)^n e^{-qx(1 - O(z^2))},$$

and hence if $q < 0$, then u_2^n produces exponentially growing oscillations, even though the solution to the differential equation decreases exponentially! This explains the weak instability seen previously.

6. (a) Show that all solutions of the difference equation

$$y_{n+1} - 2\lambda y_n + y_{n-1} = 0$$

are bounded, as $n \to \infty$, if $-1 < \lambda < 1$, while for any other λ in the complex plane there exists at least one solution which is unbounded.

(b) Let A be a diagonalizable matrix. Give, in terms of the eigenvalues of A, a necessary and sufficient condition for the boundedness as $n \to \infty$ of all solutions of the difference equation

$$y_{n+1} - 2Ay_n + y_{n-1} = 0$$

7. Consider the application of the linear multistep method, defined by the polynomials ρ and σ, to the usual test problem $y' = qy$, $y(0) = 1$.

(a) Show that if ζ_j is a simple root of ρ, then the difference equation has a particular solution which is close to $y_n = \zeta_j^n e^{\lambda_j q x_n}$, $x_n = nh$, where the so called **growth parameter** λ_j is given by

$$\lambda_j = \frac{\sigma(\zeta_j)}{\zeta_j \rho'(\zeta_j)}.$$

(b) Show that if $\zeta_j = 1$ and the method is consistent, then $\lambda_1 = 1$.

(c) Compute the growth parameters for the midpoint method and compare with the results of Problem 6.

(d) Compute the growth parameters for the Milne–Simpson's method

$$y_{n+1} = y_{n-1} + \frac{1}{3}h\big(f_{n+1} + 4f_n + f_{n-1}\big).$$

Is the method weakly stable?

8. The following results were obtained for the problem

$$y' = x^2 - y^2, \qquad y(0) = 1,$$

using the modified midpoint method with different step sizes:

$$h = 0.05 : 0.83602; \qquad h = 0.1 : 0.83672.$$

Compute a better value by extrapolation.

9. Determine $\mathcal{S}$ of backward Euler method

$$y_{n+1} = y_n + f_{n+1}.$$

Is it A-stable? Is it L-stable? What is the order of accuracy of this method?

10. Show that the second-order backward differentiation method

$$\nabla y_n + \frac{1}{2}\nabla^2 y_n = hf_n$$

is $A(0)$-stable. (Actually it is A-stable.)

11. (a) Show that $\nabla^j = (1-E^{-1})^j$, $E^{-j} = (1-\nabla)^j$. Set $\tilde{\nabla}_k = (\nabla^k,\ \nabla^{k-1}, \ldots,\ \nabla, 1)$, $\tilde{E}_k = (E^{-k},\ E^{-k+1}, \ldots,\ E^{-1}, 1)$. Determine a matrix P_k such that $\tilde{\nabla}_k = \tilde{E}_k P_k$, and show that $P_k^{-1} = P_k$. (The latter can be shown with almost no

computation.) Find a recurrence formula for the construction of the matrix $P_k,\ k = 1, 2, 3, \ldots$.
Set $\tilde{a} = (a_k,\ a_{k-1}, \ldots, a_0)^T$, $\tilde{\alpha} = (\alpha_0,\ \alpha_1, \ldots,\ \alpha_k)$, and show that $\tilde{\nabla}_k \cdot \tilde{a} = \tilde{E}_k \cdot \tilde{\alpha}$ and $P_k\tilde{a} = \alpha, \quad P_k\tilde{\alpha} = a$. ($P$ stands for Pascal.)
(b) Compute $\rho(\zeta),\ \sigma(\zeta)$, for the BDF method, for $k \leq 3$.

13.5 Extrapolation Methods

13.5.1 Extrapolated Euler's Method

In Euler's method for the initial-value problem

$$y' = f(x, y), \qquad y(0) = y_0, \tag{13.5.1}$$

one seeks approximate values $y_1, y_2, \ldots$ to the exact solution $y(x_1), y(x_2), \ldots$ by approximating the derivative at the point (x_n, y_n), $x_n = x_0 + nh$, with the difference quotient $(y_{n+1} - y_n)/h$. This gives the recursion formula

$$y_{n+1} = y_n + hf(x_n, y_n), \qquad y_0 = y(0). \tag{13.5.2}$$

The weakness of Euler's method is that the step size must be chosen quite small in order to attain acceptable accuracy.

Example 13.5.1
For the initial value problem

$$y' = y, \qquad y(0) = 1,$$

the solution is computed with Euler's method first with $h = 0.2$ and then with $h = 0.1$, and is compared with the exact solution $y(x) = e^x$:

x_n	$y(x_n)$	y_n	hf_n	error	y_n	hf_n	error
0	1.000	1.000	0.200	0.000	1.000	0.100	0.000
0.1	1.105				1.100	0.110	-0.005
0.2	1.221	1.200	0.240	-0.021	1.210	0.121	-0.011
0.3	1.350				1.331	0.133	-0.019
0.4	1.492	1.440	0.288	-0.052	1.464	0.146	-0.028
0.5	1.649				1.610	0.161	-0.039
0.6	1.822	1.728		-0.094	1.771		-0.051

The error grows as x increases and is approximately proportional to h.

The following theorem gives a theoretical basis for the use of repeated Richardson extrapolation in connection with Euler's method.

Theorem 13.5.2.

Denote by $y(x,h)$ the result of the use of Euler's method with step length h on the differential equation problem (13.5.1). Then for the global error there is an expansion of the form

$$y(x,h) - y(x) = e_1(x)h + e_2(x)h^2 + \cdots + \cdots + e_q(x)h^q + O(h^{q+1}). \quad (13.5.3)$$

Proof. See Hairer et al. [1987, II.8]. The proof is related to the fact that there is a similar expansion for the local error of Euler's method. From Taylor's formula we know that

$$\frac{y(x+h) - (y(x) + hy'(x))}{h} = \frac{h}{2}y''(x) + \frac{h^2}{3!}y'''(x) + \cdots + \cdots + \frac{h^q}{q!}y^{(q)}(x) + O(h^{q+1}).$$

□

In an extrapolation method based on Euler's method a basic step size H, and a sequence of integers $n_1 < n_2 < n_3 < \ldots$ are chosen. This defines a sequence of step sizes

$$h_i = H/n_i, \qquad i = 1, 2, 3, \ldots.$$

Denote by $A_{i,1} = y_{h_i}(x_0 + H)$ the result of the numerical method using the step size h_i. We eliminate as many terms as possible from the error expansion (13.5.3) by computing the interpolating polynomial $p(h)$ such that

$$p(h_i) = A_{i,1}, \qquad i = j, j-1, \ldots, j-k+1,$$

and extrapolate to the limit $h \to 0$. The integration then proceeds from the point $(x_0 + H, A_{jk})$, where $A_{j,k} = p(0)$.

By the above theorem repeated Richardson extrapolation can be performed using the Aitken–Neville algorithm (see Sec. 4.2.3).

$$A_{i,k+1} = A_{i,k} + \frac{A_{i,k} - A_{i-1,k}}{(n_i/n_{i-k}) - 1}, \qquad i = 2, 3, \ldots \quad k = 1, \ldots, i-1. \quad (13.5.4)$$

The values $A_{i,k}$ then represents a numerical method of order k.

Several step sequences can be used. The classical choice used by Romberg are:

$$1, 2, 4, 8, 16, 32, 64, 128, \ldots,$$

For this sequence the denominators in (13.5.4) are

$$(n_i/n_{i-k}) - 1 = 1, 3, 7, 15, 31, \ldots.$$

If the round-off error in $y(x,h)$ has magnitude less than ϵ, then the resultant error in an extrapolated value is less than 8.26ϵ.

Example 13.5.3

In the table below the value $A_{i,0}$ is the result for $x = 1$ of integrating the differential equation $y' = -y$ with initial condition $y(0) = 1$ and step size $h_i = 0.25 \cdot 2^{-i}$. This corresponds to taking $H = 1$ and $n_i = 4, 8, 16, 32$.

$$
\begin{array}{lllll}
A_{00} = 0.316406 & & & & \\
& 27203 & & & \\
A_{10} = 0.343609 & & .370812 & & \\
& 12465 & & -758 & \\
A_{20} = 0.356074 & & .368539 & & .367781 \\
& 5981 & & -168 & \qquad 12 \\
A_{30} = 0.362055 & & .368036 & & .367868 \qquad .367880
\end{array}
$$

We accept $A_{33} = 0.367880$ and estimate the truncation error as $|A_{32} - A_{33}| = 12 \cdot 10^{-6}$. The correct value is $y(1) = e^{-1} = 0.367879$.

The above sequence and the related

$$1, 2, 3, 4, 6, 8, 12, 16, \ldots.$$

have the advantage that for numerical quadrature, i.e., $y' = f(x)$ many function values can be saved and reused for larger n_i. However, for differential equations the most economical sequence is simply the harmonic sequence

$$1, 2, 3, 4, 5, 6, 7, 8, \ldots.$$

Above we used the result of the extrapolation after a basic step H as new starting value for the rest of the integration. This is called **active extrapolation**. Another way is **passive extrapolation**. This means that the results of extrapolation are accepted as output data, but that they are not used in the remaining of the calculation. Thus a passive extrapolation can be performed after the problem has been solved from start to finish with a sequence of step sizes. The result of Example 13.5.3 can be viewed as the result of passive extrapolation performed several times at $x = 1$.

Example 13.5.4

See the table in Example **??**. Denote by $\tilde{y}$ the result of *one* passive Richardson extrapolation:

x	$y(x, 0.1) - y(x, 0.2)$	$\tilde{y}$	$\tilde{y}(x) - y(x)$
0	0.000	1.000	0.000
0.2	0.010	1.220	-0.001
0.4	0.024	1.488	-0.004
0.6	0.043	1.814	-0.008

The accuracy in $\tilde{y}(x)$ is much better than in $y(x, 0.1)$. If one wants an improved result in an intermediate point–e.g., $x = 0.3$—then one gets a suitable correction by interpolating linearly in the second column, i.e.,

$$\tilde{y}(0.3) = y(0.3, 0.1) + \frac{1}{2}(0.010 + 0.024) = 1.348.$$

The error in $\tilde{y}(0.3)$ is -0.002.

Figure 13.5.1. *Passive extrapolation for two different initial step sizes.*

One might think that active extrapolation should always be preferable, but with certain types of systems, passive extrapolation is better because it is numerically more stable. Note that passive extrapolation can only be used under conditions that the *changes in step size* are done in the same way for the different initial step sizes. A situation where passive extrapolation is permissible for two different initial step sizes h_0 and $h_0/2$ is illustrated in Fig. 12.6.1.

Repeated Richardson extrapolation can be used more generally to improve a numerical method. Given the differential equation 13.5.1 and a numerical method of order p, which we write $y_{n+1} = y_n + h\Phi(x_n, y_n, h)$. Denote by $y(x, h)$ the result of the numerical method at x using the step size h. Then extrapolation can be applied if it can be shown that the global error for the method has an asymptotic expansion of the form

$$y(x,h) - y(x) = e_p(x)h^p + e_{p+1}(x)h^{p+1} + \cdots + \cdots + e_q(x)h^q + O(h^{q+1}).$$

13.5.2 The Explicit Midpoint Method

Euler's method together with repeated Richardson extrapolation is simple to use and, in some applications, also economically satisfying. Even more efficient extrapolation methods can be developed based on **symmetric** methods, whose error expansion only involve even powers of h,

$$y_h(x) - y(x) = e_p(x)h^p + e_{p+2}(x)h^{p+2} + \cdots + e_q(x)h^q + O(h^{q+2}), \qquad (13.5.5)$$

(p even). Then each extrapolation step takes the form

$$A_{i,k+1} = A_{i,k} + \frac{A_{i,k} - A_{i-1,k}}{(n_i/n_{i-k})^2 - 1}, \quad i = 2, 3 \ldots \quad k = 1, \ldots, i-1. \qquad (13.5.6)$$

and will increase the order by 2.

The **explicit midpoint method** or **leap-frog method**

$$y_{n+1} = y_{n-1} + 2hf(x_n, y_n), \qquad n = 1, 2, \ldots.$$

is a symmetric method of order two. By symmetry we mean that if we replace

$$h \leftrightarrow -h, \qquad y_{n+1} \leftrightarrow y_{n-1}$$

then $y_{n-1} = y_{n+1} - 2hf(x_n, y_n)$, i.e., we get back the original method. It is a two-step method, since it requires two values from the past, namely y_{n-1} and y_n, in the step where y_{n+1} is computed. It therefore requires a special formula for the calculation of y_1. This should be chosen so that it does not destroy symmetry, because otherwise error terms involving also odd power of h would be introduced. It was shown by W. B. Gragg in 1963 that *if Euler's method*

$$y_1 = y_0 + hf(x_0, y_0)$$

is used as starting formula then for $x = x_0 + nh$ *and* n *even* y_n *has an expansion of the form*

$$y(x,h) = y(x) + c_1(x)h^2 + e_2(x)h^4 + e_6(x)h^6 + \cdots .$$

If the harmonic sequence $2, 4, 6, 8, 10, 12, \ldots$ is chosen the denominators in (13.5.6) become

$$(n_j/n_{j-k})^2 - 1 = 3, 8, 15, 24, 35, \ldots .$$

One extrapolation will give a method of order four using only five evaluations of f. Note that a more accurate starting formula than Euler's method will give worse extrapolated results!

The following example shows that the stability properties of the modified midpoint method are not generally acceptable.

Figure 13.5.2. *Oscillations in the modified midpoint method solution.*

Example 13.5.5

Apply the modified midpoint method to the equation $y' = -y$, $y(0) = 1$, with $h = 0.1$. The exact solution is $y(x) = e^{-x}$, and $y(0.1) = 0.904837$. In Fig. 12.6.2 the numerical solution corresponding to $y_1 = 0.90$ is shown by black circles, while the solution corresponding to $y_1 = 0.85$ is shown with white circles. Note that the perturbation of the initial value gives rise to growing oscillations with a growth of approximately 10% per step. This phenomenon is sometimes called **weak instability**.

In more realistic examples the oscillations become visible much later. For example, if $y_1 = e^{-0.1}$ correct to ten decimals, we have

Table 13.5.1. *Error in solution of $\dot{y} = -y$ by the modified midpoint method.*

x_n	0	0.1	...	5.0	5.1	5.2	5.3
$y(x_n)$	1	0.90484	...	0.00674	0.00610	0.00552	0.00499
y_n	1	0.90484	...	0.01803	−0.00775	0.01958	−0.01166
$y_n - y(x_n)$	0	0.00000	...	0.01129	−0.01385	0.01406	−0.01665

The oscillations can be damped by applying the following symmetric smoothing formula for certain points where n is even

$$\hat{y}_n = \frac{1}{4}(y_{n-1} + 4y_n + y_{n+1}).$$

Because of symmetry this smoothing step will not introduce terms of odd powers and the asymptotic error expansion of $\hat{y}_n$ is again of the form (13.5.6). Another way of writing the smoothing step is

$$\hat{y}_n = (y_{n-1} + y_n + hf(x_n, y_n)).$$

This finally leads to the following **modified midpoint method** as the basis for Richardson extrapolation.

Algorithm 13.1. (W. B. GRAGG).

Let N be even, take $h = H/N$, and compute

$$y_1 = y_0 + hf(x_0, y_0), \tag{13.5.7}$$

$$y_{n+1} = y_{n-1} + 2hf(x_n, y_n), \qquad n = 1, 2, \ldots, N; \tag{13.5.8}$$

$$\hat{y}_N = \frac{1}{2}(y_{N-1} + y_N + hf(x_N, y_N)). \tag{13.5.9}$$

Again it can be shown that $\hat{y}_N$ has an error expansion of the form (13.5.5). A simple proof of this is based on rewriting the method as a one-step algorithm in terms of odd and even indices,

$$u_k = y_{2k}, \qquad v_k = y_{2k+1}.$$

The method can then be written $u_0 = v_0 = y_0$,

$$\begin{pmatrix} u_{k+1} \\ v_{k+1} \end{pmatrix} = \begin{pmatrix} u_k \\ v_k \end{pmatrix} + 2h \begin{pmatrix} f\big(x_{2k} + h, v_k + hf(x_{2k}, u_k)\big) \\ \frac{1}{2}\Big(f\big(x_{2(k+1)}, u_{k+1}\big) + f(x_{2k}, u_k)\Big) \end{pmatrix},$$

$k = 0, 1, \ldots, N/2$. This mapping from (u_k, v_k) to (u_{k+1}, v_{k+1}) can be shown to be *symmetric* since exchanging

$$u_{k+1} \leftrightarrow u_k, \qquad v_{k+1} \leftrightarrow v_k, \qquad h \leftrightarrow -h, \qquad x_k \leftrightarrow x_{k+2}$$

gives back the original formula.

Example 13.5.6

For the initial-value problem $y' = y^2$, $y(0) = 0.25$ we get with $h = 0.5$, $N = 2$,

n	x_n	y_n	$hf(x_n, y_n)$
0	0	0.25	0.03125
1	0.5	0.28125	0.039550781
2	1.0	0.329101562	0.054153919

and

$$\hat{y}_2 = \frac{1}{2}(0.28125 + 0.329101562 + 0.054153919) = 0.332252741.$$

Performing similar computations for $N = 4{,}8$ and forming the extrapolating scheme we get

$$\begin{array}{lllll} A_{00} = 0.3322\ 52741 & & & & \\ & 265,468 & & & \\ A_{10} = 0.3330\ 49146 & & 0.333314614 & & \\ & 70,726 & & 1,162 & \\ A_{20} = 0.3332\ 61325 & & 0.3333\ 32051 & & 0.3333\ 33213 \end{array}$$

Since the exact solution of the problem is $y(x) = 1/(4 - x)$, we have $y(1) = 0.333333333$. The error in the extrapolated error is thus $1/600$ of the error in $y(1, 1/8)$.

Problems

1. Given the initial value problem $y' = 1 + x^2y^2$, $y(0) = 0$. Compute $y(0.5)$ by Euler's method with repeated Richardson extrapolation. Use four decimals.

2. In a computation with Euler's method, the following results were obtained with various step sizes:

$h = 0.05$	$h = 0.1$	$h = 0.2$
1.22726	1.22595	1.22345

Compute a better value by extrapolation.

3. (a) Determine an explicit expression for y_n when Euler's method is applied to the test problem

$$y' = -y, \qquad y(0) = 1.$$

(b) For which values of h is the sequence $\{y_n\}_0^\infty$ bounded?

(c) Compute $\lim_{h\to 0}(y(x, h) - e^{-x})/h$.

13.6 Second Order Equations and Other Special Problems

13.6.1 Second-Order Differential Equations

Equations of the form

$$y'' = f(x, y, y') \tag{13.6.1}$$

with initial conditions

$$y(x_0) = y_0, \qquad y'(x_0) = z_0,$$

are often encountered in, e.g., astronomy and mathematical physics. The equation can be expressed as a set of two simultaneous first order differential equations

$$\begin{pmatrix} y' \\ z' \end{pmatrix} = \begin{pmatrix} z \\ f(x, y, z) \end{pmatrix}, \qquad \begin{pmatrix} y(x_0) \\ z(x_0) \end{pmatrix} = \begin{pmatrix} y_0 \\ z_0 \end{pmatrix}. \tag{13.6.2}$$

This system can be solved by the methods developed previously in this chapter.

Often a simplified version of (13.6.1) occurs, where the right hand side does not depend on y',

$$y'' = f(x, y). \tag{13.6.3}$$

For such systems many special methods have been developed. A simple finite-difference approximation to (13.6.3) is obtained by replacing the derivatives in the differential equation and initial condition by symmetric difference approximations. If we put $f_n = f(x_n, y_n)$, a method defined by the following relations:

$$y_{n+1} - 2y_n + y_{n-1} = h^2 f_n, \qquad y_1 - y_{-1} = 2hz_0 \tag{13.6.4}$$

This method is the simplest member of the **Störmer family** of methods, and we shall call it the **explicit central difference method**. The local error is obtained from the Taylor expansion

$$y(x_n+h)-2y(x_n)+y(x_n-h) = h^2y''(x_n)+\frac{h^4}{12}y^{(4)}(x_n)+\frac{h^6}{360}y^{(6)}(x_0)+\cdots. \tag{13.6.5}$$

The value y_{-1} can be eliminated by means of the first equation with $n = 0$. The starting procedure is, therefore

$$y_1 = y_0 + hz_0 + \frac{1}{2}h^2 f_0,$$

which is just the first terms in the Taylor-expansion of y_1. Then y_{n+1}, $n \geq 1$ can be computed successively by means of $y_{n+1} = 2y_n - y_{n-1} + h^2 f_n$. Note that at each step there is an addition of the form $O(1) + O(h^2)$; this gives unfavorable rounding errors when h is small. If we put $u_{i-1/2} = (y_i - y_{i-1})/h$ and rewrite the method as

$$\frac{1}{h}(y_{n+1} - y_n) = \frac{1}{h}(y_n - y_{n-1}) + hf_n$$

then the method can be defined by the formulas $u_{1/2} = z_0 + \frac{1}{2}hf_0$, and

$$y_n = y_{n-1} + hu_{n-1/2}, \qquad u_{n+1/2} = u_{n-1/2} + hf_n, \qquad n \geq 1. \tag{13.6.6}$$

This **summed form** of the method is mathematically equivalent, but numerically superior to the difference method (13.6.5). An alternative is to store y in double precision, while single precision is used in the computation of f, which is usually the most time-consuming part of the work. If such **partial double precision** is used, then the advantage of the summed form is reduced. See also Example 2.3.4 for a solution when double precision is not available.

Because of symmetry the following expansion holds for the explicit central difference method

$$y(x,h) = y(x) + c_1(x)h^2 + c_2(x)h^4 + c_3(x)h^6 + \cdots$$

(As usual, the rounding errors are ignored in this expansion, and certain conditions about the differentiability of f have to be satisfied). The expansion shows that the global error is $O(h^2)$ and that higher accuracy can be obtained with Richardson extrapolation according to the scheme (13.6.6).

The method (13.6.4) and its summed form (13.6.6) can be extended to equations of the form of (13.6.1) if one puts

$$y'_n \approx \frac{1}{2h}(y_{n+1} - y_{n-1}) = \frac{1}{2}(u_{n+1/2} + u_{n-1/2}),$$

but in this case the method becomes implicit.

Note that another extrapolation method is obtained by applying the modified midpoint method, Algorithm 13.5.2, directly to the first order system (13.6.2). This avoids the problem with rounding errors, which motivated the summed form of Störmer's method.

There are other ways to improve the order of accuracy of the method (13.6.4). This method is a special case of multistep methods of the form

$$\sum_{i=0}^{k}(\alpha_i y_{n+i} - h^2\beta_i f_{n+i}) = 0,$$

cf. (13.4.1). The family of methods for which

$$\sum_{i=0}^{k}\alpha_i y_{n+i} = y_{n+2} - 2y_{n+1} + y_n$$

is frequently referred to as **Störmer–Cowell methods**. The best known such method is **Numerov's method** or the implicit difference correction method:

$$y_{n+1} - 2y_n + y_{n-1} = h^2\Big(f_n + \frac{1}{12}(f_{n+1} - 2f_n + f_{n-1})\Big) \tag{13.6.7}$$

A sufficiently accurate starting procedure is obtained from the formula

$$2hy'_0 = 2hz_0 = (y_1 - y_{-1}) - \frac{1}{6}h^2(f_1 - f_{-1}) + O(h^5),$$

see Problem 1.

By expressing the higher derivatives in (13.6.5) in terms of central differences we obtain the expansion

$$y_{n+1} - 2y_n + y_{n-1} = h^2\Big(f_n + \frac{1}{12}\Delta^2 f_{n-1} - \frac{1}{240}\Delta^4 f_{n-2} + \cdots)\Big).$$

Numerov's method is obtained by taking the first two terms of the right hand side. Note that taking further terms is not practical since these contains unknown expressions f_{n+2} etc.

In Numerov's method one can proceed in the following way. If we put

$$v_i = y_i - \frac{1}{12}h^2 f_i,$$

then the difference equation then takes the form, similar to (13.6.4)

$$v_{n+1} - 2v_n + v_{n-1} = h^2 f_n, \tag{13.6.8}$$

although in order to compute $f_n = f(x_n, y_n)$ one has to solve for y_n from the equation

$$y_n - h^2\frac{1}{12}f(x_n, y_n) = v_n.$$

The summed form is similar to (13.6.6) with y replaced by v and a more accurate starting procedure. The error expansion for Numerov's method has the form

$$y(x, h) = y(x) + c_1(x)h^4 + c_2(x)h^6 + \cdots$$

and Richardson extrapolation can be applied.

If the differential equation is nonlinear, Numerov's method requires some iterative method. For the linear case, see Problem 2. Starting values can then be obtained from the explicit method obtained by using a **backward difference correction**

$$y_{n+1} - 2y_n + y_{n-1} = h^2\Big(f_n + \frac{1}{12}(f_n - 2f_{n-1} + f_{n-2})\Big) \tag{13.6.9}$$

The estimate here is less accurate, and therefore the global error of this method is only $O(h^3)$.

In the **deferred difference correction method** due to L. Fox one first computes a sequence y_n by solving the difference equation (13.6.4). Using this sequence one computes a correction term

$$C_n = h^2\frac{1}{12}(f_{n+1} - 2f_n + f_{n-1}).$$

An improved solution $\hat{y}_n$ is then obtained by solving the difference equation

$$\hat{y}_{n+1} - 2\hat{y}_n + \hat{y}_{n-1} = h^2 f(x_n, \hat{y}_n) + C_n.$$

The procedure can be iterated and more sophisticated formulas for the correction C_n can be used. The global error of the solution produced by this methods also is $O(h^4)$. The deferred correction method is very useful in solving certain boundary value problems, see Sec. **??**.

13.7 Boundary Problems

In this section we shall consider two point **boundary-value problems** for a second order scalar differential equation

$$y'' = f(x, y, y') \tag{13.7.1}$$

with **boundary** conditions

$$y(a) = \alpha \qquad y(b) = \beta, \tag{13.7.2}$$

We assume that f has continuous partial derivatives of arbitrary order in the *closed* interval $[a, b]$. More generally, we also consider boundary-value problems for a system of first order equations

$$y' = f(x, y), \qquad y \in \mathbf{R}^{s\times s}, \tag{13.7.3}$$

with boundary conditions given in two points a and b

$$r(y(a), y(b)) = 0, \tag{13.7.4}$$

where $r(y, z)$ is a vector of s, possibly nonlinear, functions. Often the boundary conditions are linear

$$r \equiv Ay(a) + By(b) - c = 0.$$

The boundary value problem (13.7.1)–(13.7.2) can be reduced to this form, by the standard substitution $y_1 = y$, $y_2 = y_1'$. The boundary conditions correspond to

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \qquad B = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, c = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}.$$

In contrast to the initial-value problem, it can happen that the boundary-value problem has several or even no solution. Sufficient conditions given in the literature for the existence of a unique solution are often astonishingly poor. For example, it is often assumed in the case of linear boundary conditions that the matrix $A + B$ is nonsingular, which is not the case in most common applications. For the practically important special case of a single second order equation, better results exist.

In **free boundary problems** b is unknown, and we have $s + 1$ equations of the form (13.7.4). We introduce a new independent variable $t = (x - a)/(b - a)$, and put

$$z_{s+1} = b - a, \qquad x = a + tz_{s+1}, \qquad 0 \le t \le 1,$$

Then this reduces to a standard boundary problem for $z(t) = y(a + tz_{s+1})$. We have the $s + 1$ differential equations

$$\frac{dz}{dt} = f(a + tz_{s+1}, z(t))z_{s+1}, \qquad \frac{dz_{s+1}}{dt} = 0,$$

with boundary conditions which now can be written $\hat{r}(z(0), z(1)) = 0$. Eigenvalue problems are considered in Sec. 13.7.3.

In the following two different types of methods will be described, **shooting** methods and **finite difference** methods.

13.7.1 The Shooting Method

There is also an initial-value problem for (13.7.1):

$$y'' = f(x, y, y'), \qquad y(a) = \alpha \qquad y'(a) = \gamma, \tag{13.7.5}$$

If γ is fixed, then the value of y at $x = b$ can be considered as a function of γ—say, $g(\gamma)$. The boundary-value problem (13.7.1)–(13.7.2) can then be written

$$g(\gamma) = \beta. \tag{13.7.6}$$

In the shooting method, one solves this equation with, for example, the secant method; see Sec. 5.4. One guesses a value γ_0, and computes (approximately) $g(\gamma_0)$ by solving the initial-value problem of (13.7.5) using one of the methods described earlier in this chapter. One then chooses another value, γ_1, computes $g(\gamma_1)$ in the same way, and then, iterates according to

$$\gamma_{k+1} = \gamma_k - \big(g(\gamma_k) - \beta\big)\frac{\gamma_k - \gamma_{k-1}}{g(\gamma_k) - g(\gamma_{k-1})}, \qquad k = 1, 2, \ldots. \tag{13.7.7}$$

One can show that $g(\gamma)$ is linear in γ when (13.7.1) is a linear differential equation, even if the coefficients depend on x. In this case, γ_2 is the solution to to (13.7.6)—aside from the discretization errors and rounding errors in the computation of $g(\gamma_0)$ and $g(\gamma_1)$.

Note that there are several variations of the shooting method. We can, e.g., consider the initial value problem $y(b) = \beta$, $y'(b) = \gamma$, integrate in reverse time direction and match the boundary condition at $x = a$. This is called **reverse** shooting. Another possibility is to integrate from *both* boundaries, and matching the solutions at an interior point $x = m$ by continuity condition on $y(x)$. The latter approach is similar to the idea of multiple shooting described in detail below.

The shooting method can also be applied to the more general boundary-value problem (13.7.3)–(13.7.4). Let $y(x, p)$ be the solution to the differential equation $y' = f(x, y)$ with *initial conditions*

$$y(a) = p, \qquad p \in \mathbf{R}^{s\times s}.$$

Then the boundary-value problem is equivalent to the system of s equations

$$F(p) \equiv r(p, y(b, p)) = 0. \tag{13.7.8}$$

If the differential equation and boundary conditions are linear, then $y(x, p)$ and $F(p)$ become affine functions of p. Hence (13.7.8) becomes a linear system.

$$Ap + By(b, p) - c = 0. \tag{13.7.9}$$

In this case the problem has a unique solution, unless the corresponding homogeneous problem has a "non-trivial" solution.

The following example shows that the initial-value problem is ill-conditioned, even when the boundary-value problem is well-conditioned.

Example 13.7.1 ((Stoer–Bulirsch))
Consider the system

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix}' = \begin{pmatrix} 0 & 1 \\ 110 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

with boundary-values

$$y_1(0) = 1, \qquad y_1(10) = 1.$$

The general solution to the initial value problem $y_1(0) = 1$, $y_1'(0) = -10 + p$ is

$$y(x,p) = (1 - \frac{p}{21})e^{-10x} \begin{pmatrix} 1 \\ -10 \end{pmatrix} + \frac{p}{21} e^{11x} \begin{pmatrix} 1 \\ 11 \end{pmatrix}.$$

Hence $y_1(10) = 1$ corresponds to

$$\frac{p}{21} = \frac{1 - e^{-100}}{e^{110} - e^{-100}} \approx 3.5 \cdot 10^{-47}.$$

With floating point arithmetic one may well obtain, say $y_1'(0) = -10 + 10^{-9}$ instead; this would give $y_1(5) \approx e^{55}10^{-9}/21 \approx 3.7 \cdot 10^{13}$ instead of $2e^{-55} \approx 2.6 \cdot 10^{-24}$.(!!)

Example 13.7.2 ((Troesch))
The exact solution of the boundary value problem

$$y'' = \lambda \sinh(\lambda y), \quad y(0) = 0, \quad y(1) = 1,$$

becomes infinite for $x \approx 1.03$ if $\lambda = 5$. The solution of the corresponding initial value problem $y(0) = 0$, $y'(0) = p$ becomes infinite for $x < 1$ if $p > 8e^{-\lambda} \approx 0.054$. The correct value is $p = 4.57504614 \cdot 10^{-2}$. Obviously it may be very difficult to find a sufficiently good initial value for p.

In such cases, the method described in Sec. 13.7.2 can be more advantageous. Another possibility is to use the **multiple shooting** method. In multiple shooting the interval $[a, b]$ is divided into m subintervals $[x_{i-1}, x_i]$, $i = 1, \ldots, m$,

$$a = x_0 < x_1 < \cdots x_m = b.$$

(Here m should be much smaller than the number of grid points needed in the numerical method for solving the initial value problem.) Let $y(x; x_k, p_k)$ be the solution of the initial value problem

$$y' = f(x, y), \qquad y(x_k) = p_k, \quad x \in [x_k, x_{k+1}],$$

$k = 0, 1, \ldots, m-1$. These m initial value problems are solved simultaneously. The continuity conditions,

$$F_i(p_i, p_{i+1}) = y(x_{i+1}; x_i, p_i) - p_{i+1} = 0, \quad i = 1, \ldots, m-1.$$

now appear as boundary conditions, in addition to the condition

$$F_m(p_1, p_m) = r(p_1, p_m).$$

Boundary and continuity conditions can now be written $F(p) = 0$. The Jacobian is

$$F'(p) = \begin{pmatrix} G_1 & -I & 0 & \cdots & 0 & 0 \\ 0 & G_2 & -I & \cdots & 0 & 0 \\ 0 & 0 & G_3 & \cdots & 0 & 0 \\ \cdots & & & & & \\ 0 & 0 & 0 & \cdots & G_{m-1} & -I \\ A & 0 & 0 & \cdots & 0 & B \end{pmatrix},$$

where

$$G_k = \frac{\partial F_k}{\partial p_k}, \qquad A = \frac{\partial r}{\partial p_1}, \qquad B = \frac{\partial r}{\partial p_m}.$$

Several decisions are to be made in the application of multiple shooting method:

1. Choice of starting trajectory $\hat{y}(x)$, e.g., some function that satisfies the boundary conditions.

2. Subdivision of the interval; it is often suitable to choose x_{i+1} from the initial approximation $\hat{y}(x)$ such that

$$\|y(x_{i+1}; x_i, \hat{y}(x_i)) - \hat{y}(x_{i+1})\| \approx \|\hat{y}(x_{i+1})\|.$$

3. Choice of iterative method, e.g., some modified Newton method. Pivoting for size can be essential in the solution of the linear systems encountered in each iteration!

Example 13.7.3

For the problem in Example 13.7.2, with $\lambda = 5$ we can choose $\hat{y}(x)$ as linear function. A better initial solution can be determined form the linearized problem $y'' = 25y$.

(This is a particular case of the problem defined by (13.7.10) and (13.7.12) in the next section). When shooting is applied to this system, one obtains a system of $2p$ equations for the vector $y(0)$. The system is nonlinear if the differential equation is nonlinear. The techniques of Sec. 12.2.4 which do not require derivatives can then be applied. It is also possible to use techniques where derivatives are needed, e.g., Newton's method, but then one has to solve also the variational equation of the system $y' = f(y, t)$; see Sec. 13.1.2.

13.7.2 The Finite Difference Method

We first consider the boundary value problem (13.7.1)–(13.7.2) for a single second order equation. Divide the interval $[a, b]$ into N equal parts and put $h = (b - a)/N$.

Let y_i denote the desired estimate of $y(x_i)$, $x_i = a + ih$. Replace derivatives in the differential equation as in Sec. 13.6.1 by symmetric difference approximations

$$vy'_n \approx \frac{y_{n+1} - y_{n-1}}{2h}, \qquad y''_n \approx \frac{y_{n+1} - 2y_n + y_{n-1}}{h^2}.$$

In this way, the differential equation is transformed into a nonlinear system of equations

$$y_{n+1} - 2y_n + y_{n-1} = h^2 f_n, \qquad n = 1, 2 \ldots, N-1,$$

where

$$f_n = f\Big(x_n, y_n, \frac{y_{n+1} - y_{n-1}}{2h}\Big).$$

Together with the boundary conditions $y_0 = \alpha$, $y_N = \beta$ this system can be written in matrix form

$$Ay = h^2 f - r, \tag{13.7.10}$$

where

$$A = \begin{pmatrix} -2 & 1 & 0 & \cdots & 0 & 0 \\ 1 & -2 & 1 & \cdots & 0 & 0 \\ 0 & 1 & -2 & \cdots & 0 & 0 \\ \cdots & & & & & \\ 0 & 0 & 0 & \cdots & -2 & 1 \\ 0 & 0 & 0 & \cdots & 1 & -2 \end{pmatrix}, \qquad r = \begin{pmatrix} \alpha \\ 0 \\ 0 \\ \vdots \\ 0 \\ \beta \end{pmatrix}, \qquad y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_{N-2} \\ y_{N-1} \end{pmatrix}.$$

Thus A is a **band matrix** (and in fact a tridiagonal matrix in this example.) If the differential equation is linear, then the system of equations is linear and tridiagonal—thus it can be solved by very little work. Note that the matrix of the system is not A, since f depends on y and y'.

For the error, even in the nonlinear cases, we have

$$y(x, h) = y(x) + c_1(x)h^2 + c_2(x)h^4 + c_3(x)h^6 + \cdots,$$

and Richardson extrapolation can be used with correction terms $\Delta/3, \Delta/15, \Delta/63, \ldots$.

Example 13.7.4

The boundary-value problem

$$y'' + y = x, \qquad y(0) = 1, \qquad y(\frac{1}{2}\pi) = \frac{1}{2}\pi - 1$$

has the exact solution $y(x) = \cos x - \sin x + x$. The difference equation becomes

$$y_{n+1} - 2y_n + y_{n-1} - h^2 y_n = h^2 x_n, \qquad n = 1, 2 \ldots, N-1.$$

The solution of the system of equations for $N = 5$ and $N = 10$ and the result $\hat{y}$ of

Richardson extrapolation is given by the following table:

$2x/\pi$	$y(x)$	$y(x,0.1\pi)$	$y(x,0.05\pi)$	$\Delta/3$	$\hat{y}(x)$	error
0	1.000000	1.000000	1.000000	0	1.000000	0
0.1	0.988334		0.988402			
0.2	0.956199	0.956572	0.956291	−94	0.956197	−2
0.3	0.908285		0.908337			
0.4	0.849550	0.849741	0.849597	−48	0.849549	−1
0.5	0.785398		0.785398			
0.6	0.721246	0.721056	0.721199	+48	0.721247	1
0.7	0.662541		0.662460			
0.8	0.614598	0.614224	0.6145505	94	0.614599	1
0.9	0.582463		0.582395			
1.0	0.570796	0.570796	0.570796	0	0.570796	0

The methods for improving accuracy described in Sec. 13.6.1, can also be used for boundary value problems. In particular, for equations of the form $y'' = f(x, y)$ Cowell's method gives $O(h^4)$ accuracy with about the same amount of computation as the $O(h^2)$-method just described.

If the equation is nonlinear, then one can use some modified Newton method. As a first approximation one can—in the absence of a better proposal—choose a linear function satisfying the boundary condition, i.e.

$$y_i^{(0)} = \alpha + (\beta - \alpha)/N, \qquad i = 0, 1, \ldots, N.$$

In boundary value problems one often encounters differential expressions of the form

$$\frac{d}{dx}\Big(p(x)\frac{dy}{dx}\Big). \tag{13.7.11}$$

These can be approximated at x_n by

$$\frac{1}{h}\Big(p_{n+1/2}\big(\frac{y_{n+1}-y_n}{h}\big) - \big(p_{n-1/2}\big(\frac{y_n - y_{n-1}}{h}\big)\Big) \tag{13.7.12}$$

with global error of of the form $c_1(x)h^2 + c_2(x)h^4 + \ldots$.

With boundary conditions of the form $b_0y(b) + b_1p(b)y'(b) = b_2$ one can introduce an *extra point*, $x_{N+1} = b + h$, and approximate the condition by

$$b_0y_N + \frac{p_1(y_{N+1} - y_{N-1})}{2h} = b_2,$$

and similarly for the condition at $x = a$. One can also put b *between* two grid points. The form of the boundary conditions only affect the first and last rows in the matrix A in (13.7.10).

For systems of first-order equations $y' = f(x, y)$ the trapezoidal method can be used

$$y_{n+1} - y_n = \frac{1}{2}h\big(f(y_n, x_n) + f(y_{n+1}, x_{n+1})\big), \qquad n = 1, 2, \ldots, N.$$

With linear boundary conditions $Ay(a)+By(b)=c$ one obtains a nonlinear system of simple structure.

If no natural initial approximation is available an **embedding technique**, described in Sec. 12.1.7, is often useful. One can introduce a parameter in the differential equation, or sometimes it may be sufficient to use the step size h of the difference method as a parameter, i.e., one starts with a very crude grid, and refines it successively in using, e.g., the previously obtained solution and interpolation as initial approximation for the solution on the next finer grid.

13.7.3 Eigenvalue Problems

Many important eigenvalue problems in applied mathematics have the form

$$(p(x)y')' - q(x)y + \lambda r(x)y = 0, \tag{13.7.13}$$

subject to homogeneous boundary conditions

$$a_0y(a) - a_1p(a)y'(a) = 0, \qquad b_0y(b) + b_1p(b)y'(b) = 0, \tag{13.7.14}$$

where λ is a scalar parameter to be determined. This is called a **Sturm–Liouville problem**. Note that $y(x)$ has to be normed to be uniquely determined.

A more general form of the eigenvalue problem is

$$y' = f(x, y, \lambda), \qquad Ey(a) + Fy(b) - c = 0. \tag{13.7.15}$$

By introducing

$$y_0(x) = \lambda \text{ (constant)}, \qquad y_{s+1} = \int_a^x y^T y dx$$

and the differential equations

$$y_0' = 0, \qquad y_{s+1}' = y^T y,$$

this can be reduced to the standard form above for $\tilde{y} = (y_0, y_1, \ldots, y_{s+1})^T$.

Example 13.7.5

For which value of λ does the boundary-value problem

$$y'' + \lambda y = 0, \qquad y(0) = y(1) = 0,$$

have solutions other than $y = 0$?

The general solution to the differential equation is

$$y(x) = a\cos\mu x + b\sin\mu x, \qquad \mu = \sqrt{\lambda},$$

¿From $y(0) = 0$ it follows that $a = 0$. Further since $y(1) = 0$ we have $\mu = n\pi$, $n = 0, \pm1, \pm2, \ldots$. Thus the **eigenvalues** are

$$\lambda = n^2\pi^2, \qquad n = 1, 2, 3, \ldots.$$

Note that $n = 0$ gives the trivial solution $y = 0$; $n = -k$ gives the same solution as $n = k$. The solution of the differential equation when λ is an eigenvalue are called **eigenfunctions**. In this example the eigenfunctions $y(x) = b \sin n\pi x$ belong to the eigenvalue $\lambda = n^2\pi^2$.

Eigenvalue problems occur in most areas of classical and modern physics (for eigen-vibrations, etc.) Example 13.7.5 comes up, e.g., in the computation of wave numbers for a vibrating string. Some other important problems in partial differential equations from physics can be reduced, by separation of variables, to eigenvalue problems for ordinary differential equations.

The difference method according to Sec. 13.7.2 gives an approximation to the eigenvalues which satisfies

$$\lambda(h) = \lambda + c_2h^2 + c_3h^3 + c_4h^4 + \cdots,$$

where c_3 is zero in some cases (among others, in Example 13.7.3). Note the regularity assumptions made in Sec. 13.6.1.

Example 13.7.6

For the problem in Example 13.7.5, the difference method with $h = 1/3$ gives the system of equations

$$\begin{aligned} -2y_1 + y_2 + \lambda h^2 y_1 &= 0 \\ y_1 - 2y_2 + \lambda h^2 y_2 &= 0. \end{aligned}$$

This is a homogeneous system of equations, two equations and two unknowns, which has a nontrivial solution if and only if λh^2 is an eigenvalue of the matrix

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}.$$

Thus $\lambda h^2 - 2 = \pm 1$, with solutions

$$\begin{aligned} \lambda_1 &= 9, \quad (\text{Exact value} \quad \pi^2 = 9.8696), \\ \lambda_2 &= 27, \quad (\text{Exact value} \quad 4\pi^2 = 39.48). \end{aligned}$$

The higher eigenvalues cannot even be estimated using such a course grid.

By similar calculations with various values of h we get the following results for the smallest eigenvalue:

h			Richardson	Error
$\frac{1}{2}$	8			
$\frac{1}{3}$	9	$\Delta/\frac{5}{4} = 0.8$	9.8	-0.0696
$\frac{1}{4}$	9.3726	$\Delta/\frac{7}{9} = 0.4791$	9.8517	-0.0179

A second Richardson extrapolation will cancel the $O(h^4)$ error term and gives

$$\lambda = 9.8517 + \frac{1}{3}0.0517 = 9.8689$$

correct to four decimal places. Motivate this extrapolation!

There are computational methods for solving eigenvalue problems for much larger matrices; see Sec. 10.10. By using Richardson extrapolation one can, however, obtain good accuracy with a reasonable number of points. The same general procedure can be used with differential expressions of the form of (13.7.13) with the difference expression of the form of (13.7.12).

The shooting method can also be used on eigenvalue problems, see Problems 3 and 4 below.

Problems

1. (a) Put

$$hy_0' = A\mu\delta(y_0 + Bh^2y'') + R_T,$$

and determine A and B so that the remainder term is of as high order as possible. Give an asymptotically correct expression for R_T.

(b) Let $y(x)$ be the solution to the differential equation problem

$$y'' = f(x)y, \qquad y(0) = \alpha, \qquad y'(0) = \beta.$$

Put $y(kh) = y_k$, $k = -1, 0, 1, \ldots$. With Numerov's formula and the formula in (a), one gets a system of equations for determining y_{-1} and y_1. Give an asymptotically correct expression for the error in the determination of y_1 obtained from this system of equations.

(c) Apply the formula in (b) to compute $y(0.1)$ with $h = 0.1$ in the case

$$y'' = e^x y, \qquad y(0) = 0, \qquad y'(0) = 1.$$

2. For what positive values of the step size h does Numerov's method produce bounded solutions when applied to the differential equation

$$y'' = -y?$$

3. Consider the application of Numerov's method to the linear equation

$$y'' = p(x)y + q(x).$$

Show that with the notation of (13.6.8)

$$f_n = \big(p(x_n)v_n + q(x_n)\big)\,/\,\big(1 - \frac{h^2}{12}p(x_n)\big).$$

4. Write programs for the solution of $y'' = f(x, y)$ with the difference method (13.6.4) and the summed form (13.6.5). Apply them to the equation $y'' = -y$, compare with the exact solution, and print out the errors. Perform a series of numerical experiments in order to get acquainted with the accuracy obtained with Richardson extrapolations and see the effect of rounding errors.

5. Consider the initial-value problem

$$y'' = (1 - x^2)y, \qquad y(0) - 1, \qquad y'(0) = 0.$$

(a) Show that the solution is symmetric about $x = 0$.

(b) Determine $y(0.4)$ using Numerov's method without any special start formula, with step lengths $h = 0.2$ and $h = 0.4$. Perform Richardson extrapolation.

6. The function $y(x)$ is defined by the problem

$$y''' = yx, \qquad y(0) = 0, \qquad y'(0) = 1, \qquad y''(0) = 1.$$

To determine $y(1)$ one puts

$$y'''(x_n) \approx \frac{\mu\delta^3 y_n}{h^3} = \frac{y_{n+2} - 2y_{n+1} + y_{n-1} - y_{n-2}}{2h^3},$$

where $x_n = nh$, and derives the recursion formula $y_0 = 1$,

$$y_{n+2} = 2y_{n+1} + 2h^3 x_n y_n - 2y_{n-1} + y_{n-2}, \quad n \geq 2.$$

The initial values y_1, y_2, y_3 needed in the recursion formula can be obtained, for example, by using the classical Runge–Kutta method with step h.

(a) What is the order of accuracy of the method?

(b) We give below computed values $y(h_k) = y(1, h_k)$ using the above method and step lengths $h_0 = 0.1$, $h_k = h_0/2^k$, $k = 0, 1, \ldots, 7$:

$$1.54568, 1.54591, 1.54593, 1.54592, 1.52803, 1.50045, 1.51262, 1.48828.$$

What is puzzling about these results? Explain what the trouble is.

7. In using the shooting method on the problem

$$y'' = \frac{1}{2}y - \frac{2(y')^2}{y}, \qquad y(0) = 1, \qquad y(1) = 1.5,$$

the following results are obtained using $y'(0) = p$:

$$p = 0: \quad y(1) = 1.543081, \qquad p = -0.05: \quad y(1) = 1.425560.$$

What value of p should be used on the next "shot"?

8. (From Collatz [7].) The displacement u of a loaded beam of length $2L$ satisfies under certain assumptions the differential equation

$$\frac{d^2}{ds^2}\Big(EI(s)\frac{d^2u}{ds^2}\Big) + Ku = q(s), \qquad -L \leq s \leq L,$$

with boundary conditions

$$u''(-L) = u'''(-L) = 0, \qquad u''(L) = u'''(L) = 0.$$

For a certain beam we have:

$$I(s) = I_0(2 - (s/L)^2), \quad q(s) = q_0(2 - (s/L)^2), \quad K = 40EI_0/L^4.$$

One wants to know the displacement at $s = 0$.

(a) Introduce more suitable variables for this problem, and write it as a system of two second-order equations with boundary conditions. Prove, and make use of the symmetry property $u(s) = u(-s)$. (Assume that the system has a unique solution.)

(b) Propose a difference approximation for this problem, where $h = L/N$, N an arbitrary positive integer. Count the number of equations and unknowns. Verify that for $N = 1$ one gets

$$u(0) = 13/280c = 0.046429c, \qquad c = q_0L^4/(EI_0).$$

(c) In a computation one obtained

$$N = 2: \quad u(0) = 0.045752c, \qquad N = 5: \quad u(0) = 0.045332c.$$

Perform Richardson extrapolation, first with $N = 1$ and $N = 2$, and then with $N = 2$ and $N = 5$.

(d) How should one number the variables to get a small band width in the matrix?

9. (a) Compute approximately, the smallest eigenvalue λ for which the problem

$$\frac{d}{dx}\Big((1 + x^2)\frac{dy}{dx}\Big) + \lambda y = 0, \qquad y(-1) = y(1) = 0$$

has a nontrivial solution. Use a difference method with step size $h = 2/3$ and then $h = 2/5$, and perform Richardson extrapolation. (*Hint*:Utilize the symmetry of the eigenfunctions about $x = 0$.)

(b) Use the same difference method with $h = 1/5$ to solve the differential equation with initial-values $y(0) = 1$, $y'(0) = 1$, for $\lambda = 3.50$ and $\lambda = 3.75$. Use inverse interpolation to compute the intermediate value of λ for which $y(1) = 0$, and make a new computation with the value of λ so obtained. Then improve the estimate of the eigenvalue using Richardson extrapolation.

(c) Compute the next smallest eigenvalue in the same way.

10. One seeks the solution of the eigenvalue problem

$$\frac{d}{dx}\Big(\Big(\frac{1}{1 + x}\Big)\frac{dy}{dx}\Big) + \lambda y = 0, \qquad y(0) = y(1) = 0$$

by integrating, for a few values of λ, an equivalent system of two first-order differential equations with initial values $y(0) = 0$, $y'(0) = 1$, with the classical Runge–Kutta method. Computations using three different λ, each with three

different step sizes, gave the following values of $y(1) \cdot 10^4$:

h \ λ	6.76	6.77	6.78
$\frac{1}{10}$	16.126	5.174	−5.752
$\frac{1}{15}$	16.396	4.441	−6.490
$\frac{1}{20}$	15.261	4.304	−6.627

Compute, for each value of λ, a better value for $y(1)$ using Richardson extrapolation. Then use inverse interpolation to determine the value of λ which gives $y(1) = 0$.

11. The eigenvalue for a stretched circular membrane is

$$\frac{1}{r}\frac{d}{dr}\Big(r\frac{du}{dr}\Big) + \lambda u = 0,$$

with $u(1) = 0$, $u(0)$ and $u''(0)$ finite.

(a) Set up a difference equation with grid points

$$r_i = \frac{2i+1}{2N+1} = r_0 + ih, \qquad i = -1, 0, 1, \ldots, N, \qquad h = \frac{2}{2N+1}.$$

(Thus the origin lies between the two grid points r_{-1} and r_0.)

(b) Determine the smallest eigenvalue, first for $N = 1$ and then for $N = 2$. Perform Richardson extrapolation under the assumption that the global error is proportional to h^2.

(c) For large values of N, one would like to use a standard program for computing eigenvalues of symmetric matrices. How does one put the above problem in a suitable form?

Remark: The origin is a singular point for the differential equation, but not for its solution. The singularity causes no trouble in the above treatment.

12. Show that (13.7.6) is a first-degree equation when (13.7.1) is a linear differential equation, even if the coefficients depend on x. Show also that multiple shooting yields a linear system for the determination of $y(0)$ when the differential equation is linear.

13.8 Qualitative Theory and Separably Stiff Equations

13.8.1 On Lyapunov Stability Theory

We shall in this section prove and generalize some results mentioned in Sec. 13.1. The reader is advised to repeat §13.1.4 and the later part of §13.1.5 (from Theorem 13.1.12).

We shall, in *real two-dimensional examples*, occasionally use an established terminology for a critical point, based on the eigenvalues of the Jacobian. For the linear system $\dot{y} = Ay$ we have the following:

- If they are real and of the same sign, the point is a stable or unstable **node**.
- If they are real and of opposite sign, the point is a **saddle point**.
- If they are conjugate complex with a non-zero real part, the point (or rather an orbit in its neighborhood) is a stable or unstable **spiral point**.
- If they are pure imaginary, the point is a neutrally stable **center**.
- If A is singular, one or both eigenvalues are zero, the critical point is not unique. The orbits are rays or, if $A = 0$, just the critical points.

If you are not familiar with this terminology, see Problem 9.

By Theorem **??** the behavior in *the neighborhood of a critical point of a nonlinear system* is approximately the same, except that a neutrally stable center can also become a stable or unstable spiral point in the nonlinear case. There is also a case named **elliptic sector** that has no counterpart in a linear problem; see Problem 5.

We omit a detailed discussion of the case of a singular A, where one has to consider, whether or not the Jacobian has the same rank at the critical point as in its neighborhood.

The above definitions of stability etc. are essentially due to Lyapunov. In some texts our notions are named *uniform* stability etc., since one considers also a more general case, where δ may depend on the initial time a, so that δ is not uniformly bounded away from zero.

Notice that, in the unstable linear autonomous case, the solution is bounded, if the initial value is restricted to the subspace spanned by the eigenvectors belonging to the eigenvalues with negative real parts. Some authors use the term *conditionally stable* for this case, and applies this notion also to the nonlinear case, where in general the set of exceptional initial values will be a nonlinear manifold. We shall not use this terminology.

Lyapunov's name is also associated with a technique for investigating stability. For the sake of simplicity, we restrict the presentation to autonomous systems. A more general treatment is found in Lefschetz [26]. We begin by a generalization of two notions earlier used for quadratic forms only.

Definition 13.8.1.

A real-valued function $V(y)$ is **positive definite** *in an open environment Ω of a critical point p of the system $\dot{y} = f(y)$, if*

(a) *$V(y)$ and $V'(y)$ are continuous in Ω,*

(b) *$V(p) = 0$,*

(c) *$V(y) > 0$ for $y \in \Omega,\ y \neq p$.*

$V(y)$ is **negative definite** *if $-V(y)$ is positive definite.*

Definition 13.8.2.
A positive definite function $V(y)$ is called a **Lyapunov function**, *if $V'(y)f(y) \leq 0$ in Ω.*

Recall the definitions of stability and asymptotic stability (in the sense of Lyapunov), see Definition 13.1.1 in§13.1.4.

The importance of $V'(y)f(y)$ is explained by the following equation:

$$dV(y)/dt = V'(y)dy/dt = V'(y)f(y), \text{ for } y = y(t),$$

i.e. *$V(y(t))$ is non-increasing along any trajectory in Ω.*

Theorem 13.8.3 ((Lyapunov)**).**
If there exists a Lyapunov function $V(y)$ in some neighborhood Ω of a critical point p, then the critical point is stable.
Set $V'(y)f(y) = -W(y)$. If $W(y)$ is positive definite in Ω, then the critical point is asymptotically stable.

Proof. Let $C_\epsilon = \{y : \|y - p\| = \epsilon\}$, and set $v_\epsilon = \inf\{V(y) : y \in C_\epsilon\}$. Set $\delta(\epsilon) = \inf\{\|y\| : V(y) = v_\epsilon\}$. Therefore, if $\|y_0\| < \delta_\epsilon$ then $V(y(0)) < v_\epsilon$. Since $V(y(t))$ does not increase. It follows that $V(y(t)) < v_\epsilon, \forall t > 0$, hence $y(t)$ cannot reach C_ϵ. The stability of the critical point p is proved.

Now we shall prove the *asymptotic* stability, i.e., prove that $\lim_{t\to\infty} V(y(t)) = 0$. It is *sufficient to prove that for any (t_1, v_1), such that $V(y(t_1)) \leq v_1$, there exists a t_2, such that $V(y(t_2)) \leq v_1/2$*, because this condition similarly implies that there exists a t_3 such that $V(y(t_3)) \leq v_1/2^2$. This argument can be repeated. So, for any n, there exists a t_{n+1} such that $V(y(t_{n+1})) \leq v_1/2^n$. This clearly implies asymptotic stability.

The proof is indirect, i.e., *suppose that*, for some (t_1, v_1),

$$V(y(t_1)) \leq v_1, \quad \text{and} \quad V(y(t)) > v_1/2, \ \forall t > t_1.$$

Let $\beta > 0$ be the infimum of $W(z)$ for all z such that $v_1/2 < V(z) < v_1$. Take any $t_2 > t_1$. For any $\tau \in [t_1, t_2]$, $v_1/2 \leq V(y(\tau)) \leq v_1$, due to the monotonicity of $V(y(t))$. By the mean value theorem, there exists a τ, such that

$$V(y(t_1)) - V(y(t_2)) = -\dot{V}(y(\tau)) \cdot (t_2 - t_1) = W(y(\tau)) \cdot (t_2 - t_1) \geq \beta(t_2 - t_1).$$

Now choose t_2 such that $\beta(t_2 - t_1) = v_1/2$. Then $v_1 - V(y(t_2)) \geq v_1/2$, i.e., $V(y(t_2)) \leq v_1/2$, but this contradicts the supposition. The contradiction shows that there exists a t_2, such that $V(y(t_2)) \leq v_1/2$, and we found above that this implies asymptotic stability. □

It is typically not trivial to construct a Lyapunov function for a given problem. The most common type of Lyapunov function is a *norm* (or, e.g., some power of a norm). The technical details are worked out by means of the theory of logarithmic norms that yields estimates, valid not only for sufficiently large t and in a sufficiently small neighborhood of a critical point etc..

Quadratic Lyapunov functions are typically equivalent to inner-product norms. There are, however, important cases, where norms are not successful, in particular for the study of periodic solutions. See, e.g., Example 13.8.6, where a Lyapunov function is found by energy considerations of a type that can be useful in many problems. In problem P6 (the predator-prey problem) a rather special explicit construction of a different type is possible.

We shall prove the formula for $\mu_2(A)$ that was announced but unproved in Theorem 13.1.8. It is generalized to an arbitrary inner- product norm. We first derive a formula for the matrix norms and logarithmic norms, subordinate to l_p-norms, for a diagonal matrix D,

Let $D = \operatorname{diag}(d_i)$, $\max_i |d_i| = |d_{i'}|$. Then

$$\|Du\|_p^p = \sum_i |d_i u_i|^p \le \max_i |d_i|^p \|u\|_p^p.$$

Equality is obtained, e.g., when $u = e_{i'}$, (one of the basis vectors). Hence $\|D\|_p = |d_{i'}|$. Now substitute $I + \epsilon D$ for D. Then $\|I + \epsilon D\|_p = |1 + \epsilon d_{i^*}|$, where i^* is the same for all sufficiently small ϵ (though not necessarily equal to i'). Then, by the definition of logarithmic norm, $\mu_p(D) = \Re d_{i^*}$, which must be equal to $\max \Re d_i$. It follows that the formulas

$$\|D\|_p = \max |d_i|, \qquad \mu_p(D) = \max \Re d_i, \tag{13.8.1}$$

are valid for any l_p-norm. *The same derivation also holds for weighted l_p-norms.*

Theorem 13.8.4.

Let H be a positive definite Hermitean matrix. Then $(u, v) = u^H H v$ is an inner product in $\mathbf{C}^s$, and $\|u\| = \sqrt{(u,u)}$. Then the subordinate logarithmic norm equals

$$\mu(A) = \max_{(u,u)=1} \Re(u, Au) = \max\{\kappa : \det(A^H H + HA - 2\kappa H) = 0\}, \tag{13.8.2}$$

$$\mu(A) < 0 \ \leftrightarrow \ A^H H + HA \text{ negative definite} \tag{13.8.3}$$

For the l_2-norm $H = I$, and

$$\mu_2(A) = \max_i d_i, \tag{13.8.4}$$

where d_i denotes an eigenvalue of the Hermitean part $B = \frac{1}{2}(A + A^H)$ of A, (Note that (13.8.4) *contains* $\max d_i$*, not* $\max |d_i|$.)

Proof. In order to derive (13.8.2), consider the equation

$$\|I + \epsilon A\|^2 - 1 = \max_{(u,u)=1} ((u + \epsilon Au, u + \epsilon Au) - 1).$$

The second member equals the maximum of

$$(u,u) + \epsilon(Au, u) + \epsilon(u, Au) + \epsilon^2(Au, Au) - 1 = \epsilon\big(2\Re(Au, u) + O(\epsilon)\big).$$

The first member can be written $(\|I+\epsilon A\|+1)\cdot(\|I+\epsilon A\|-1)$. Hence

$$(2+O(\epsilon))\frac{\|I+\epsilon A\|-1}{\epsilon} = 2\max_{(u,u)=1}\Re(Au,u)+O(\epsilon),$$

The first part of (13.8.2) follows, as $\epsilon \to +0$. The proofs of (13.8.3) and the second part of (13.8.2) are left for Problem 5.

Finally, the l_2-norm is an inner product norm, where $(u,v)=u^Hv$. Hence

$$\mu_2(A) = \max_{\|u\|_2=1}\Re u^HAu = \max_{\|u\|_2=1} u^HBu,$$

where $B=\frac{1}{2}(A+A^H)$ is the Hermitean part of A. Let T be a unitary matrix that diagonalizes B, i.e., $T^HBT=\operatorname{diag}(d_i)$, and set $u=Tv$. Note that $\|v\|_2=\|u\|_2=1$. Then

$$u^HBu = v^HT^HBTv = v^H\operatorname{diag}(d_i)v = \sum_i d_i|v_i|^2 \le \max_i d_i.$$

The same type of argument as was used in the derivation of (13.8.1) then shows that

$$\mu_2(A) = \max_{\|u\|_2=1} u^HBu = \max_i d_i.$$

(The maximum is obtained for $u=Te_i$, where e_i is one of the basis vectors.) □

The logarithmic norm can also be used to derive inequalities with reversed sign to the inequalities given earlier. The derivations are analogous. In some cases simple substitutions are enough. The details are left for Problem 10. There are similar modifications of other properties of $\mu(A)$.

Theorem 13.8.5.

A. *If* $\dot u = Ju+r$ *then* $\|u\|' \ge -\mu(-J)\|u\|-\|r\|$ *for* $t\ge 0$.
If $-\mu(-J)\ge\alpha$, *and* $\|r\|\le\epsilon$, *then* $\|u(t)\|\ge\psi(t)$, *where* $\psi'=\alpha\psi-\epsilon$, $\psi(0)\le\|u(0)\|$. *Moreover,* $\|e^{Jt}\|\ge e^{-\mu(-J)t}$, $\forall t\ge 0$.

B. *For any choice of norm,* $-\mu(-A)\le\min\Re\lambda(A)$, *and, for any* $\epsilon>0$, *there is a norm such that* $-\mu(-A)\ge\min\Re\lambda(A)-\epsilon$.

C. $\mu(A)+\mu(-A)\ge\max\Re\lambda(A)-\min\Re\lambda(A)$,
$\mu_2(A)+\mu_2(-A)=\max\Re\lambda(B)-\min\Re\lambda(B)$, *where* B *is the Hermitean part of* A.

We are now in a position prove the second part of Theorem **??** that was not proved in Sec. 13.1.

Proof. We consider nonlinear systems of the form $\dot y = Ay+g(t,y)$, where A is a constant matrix, and $\|g(t,y)\|/\|y\|\to 0$, uniformly in t, as $y\to 0$.

In the second part, it is assumed that A has at least one eigenvalue with a positive real part, and we shall prove that the origin is unstable for the nonlinear system. We can then make a coordinate transformation $y = Tu,\ g = Th$, that brings the operator A to block diagonal form, $T^{-1}AT = \text{blockdiag}(J_1, J_2)$, where (say)

$$\Re\lambda(J_1) \geq \alpha_1 > 0, \quad \Re\lambda(J_2) \leq \alpha_2 < \alpha_1.$$

The transformed system reads, in partitioned form,

$$u_1' = J_1u_1 + h_1(t,u), \quad u_2' = J_2u_2 + h_2(t,u).$$

Our proof is indirect. Suppose that the origin is stable, i.e., for any positive ϵ, we can find a $\delta > 0$ such that $\|u_i(0)\| \leq \delta,\ i = 1,\ 2$, implies that $\|u_i(t)\| \leq \epsilon,\ \forall t > 0$. For any $\theta > 0$ we can, by Theorems 12.2.15B and 12.2.13, choose norms so that

$$-\mu(-J_1) \geq \alpha_1 - \theta, \quad \mu(J_2) \leq \alpha_2 + \theta.$$

Next we choose two numbers θ and η, both less than $\frac{1}{8}\min(\alpha_1 - \alpha_2,\ \alpha_1)$. For later use, note that

$$\alpha_1 - \theta - 2\eta \geq \alpha_2 + \theta + 2\eta, \quad \alpha_1 - \theta - 2\eta \geq 5\alpha_1/8.$$

Finally ϵ should be small enough that $\|h_i(t,u)\| \leq \eta(\|u_1\| + \|u_2\|),\ i = 1,\ 2$. This is possible due to the assumption originally made for $\|g(t,y)\|$. Then, by Theorem 13.8.5 (statement A) and formula (13.8.13),

$$\begin{aligned}\|u_1\|' &\geq (\alpha_1 - \theta)\|u_1\| - \eta(\|u_1\| + \|u_2\|),\\ \|u_2\|' &\leq (\alpha_2 + \theta)\|u_2\| + \eta(\|u_1\| + \|u_2\|).\end{aligned}$$

It follows that

$$\begin{aligned}\|u_1\|' - \|u_2\|' &\geq (\alpha_1 - \theta - 2\eta)\|u_1\| - (\alpha_2 + \theta + 2\eta)\|u_2\|\\ &\geq (\alpha_1 - \theta - 2\eta)(\|u_1\| - \|u_2\|).\end{aligned}$$

We choose $\|u_1(0)\| = \delta \ll \epsilon,\ u_2(0) = 0$. Then $\|u_1(t)\| - \|u_2(t)\| \geq 0$ for $t \geq 0$. Hence, $(\|u_1\| - \|u_2\|)' \geq \frac{5}{8}\alpha_1(\|u_1\| - \|u_2\|)$. We obtain,

$$\|u_1(t)\| - \|u_2(t)\| \geq \delta e^{5\alpha_1 t/8} \to \infty \quad t \to \infty.$$

This contradicts our hypothesis that the origin is stable. □

13.8.2 On Periodic Solutions of ODEs and Related Questions.

Example 13.8.6 * Se tillatt beteckningarna blir konsekventa med Exercise C7.** The equation for a damped or undamped **pendulum** reads (after a scaling of the time variable): $\phi'' + a\phi' + \sin\phi = 0$, $(a \geq 0)$. We can write this in the form $\dot{y} = f(y)$, if we set $y = [y_1, y_2]^T = [\phi,\ \phi']^T$, $f(y) = [y_2,\ -ay_2 - \sin y_1]^T$. We shall give the

main features of a study of the stability of the critical point at $y = 0$. The details are left for Problem 10d.

The sum of the potential and the kinetic energy is $E(y) = 1 - \cos y_1 + \frac{1}{2}y_2^2$. We choose $V(y) = E(y) + \eta y_1 y_2$, where η is a small positive quantity. Then $V'(y)f(y) = -\eta y_1 \sin y_1 - (a - \eta)y_2^2 - a\eta y_1 y_2$.

In the *damped case*, $a > 0$, we choose $\eta \ll a$. $V'(y)f(y)$ is a negative definite function in a sufficiently small neighborhood Ω, since we obtain a negative definite quadratic form, if $y_1 \sin y_1$ is replaced by y_1^2. Then, by the corollary above, the origin is *asymptotically stable*. The orbit in the y-plane spirals in towards the origin. Phase plane plots for pendulums, damped and undamped, are shown in Fig. 13.8.3, in connection with exercise C???).

If we had chosen $\eta = 0$, the quadratic form, which approximates $V'(y)f(y)$, would have become semi-definite only. That does not prove asymptotic stability.

In the *undamped case*, $a = 0$, we take $\eta = 0$ and find that $V'(y)f(y) = 0$, hence the origin is *stable*. Note that this means that $dV(y(t))/dt = 0$, hence $V(y)$ *is constant during the motion*. (This is not unexpected, since in this case $V(y)$ equals the total energy.) If the starting point is sufficiently close to the origin, the motion will be periodic along a closed level curve for $V(y)$. See Fig. 13.2.5. If we let the starting point approach the origin, the period of the the solution tends to the period for the linear approximation, $\dot{y} = Ay$, where A is the Jacobian at the critical point. (In this example the period of the linear system is 2π.)

At a simulation the period can be computed by means of the times for the intersections between the orbit and some ray. (For larger systems there will be a plane or a hyperplane instead of a ray.) □

** Se till att det inte blir fr mkt upprepning här. **

In numerical analysis a primary interest is to estimate the difference between a perturbed solution $z(t)$ and an unperturbed solution $y(t)$ of a system of ODEs. In the particular case where we consider the effect of a perturbation at the initial point only, the variable transformation $u(t) = y(t) - z(t)$ makes the origin a critical point for a differential system for the function $u(t)$. (Note however that this system usually becomes non-autonomous even if the original system for y is autonomous.) We usually discuss this in terms of a norm $\|u(t)\|$, and in the next section techniques will be developed, based on the notion of logarithmic norms, to make this type of analysis efficient. *A norm is a particular kind of Lyapunov function*, except that differentiability is not required, e.g., the max-norm and the l_1-norm are not differentiable everywhere.

A positive definite quadratic form is a frequently used type of Lyapunov function. Since such a form is the square of an inner-product norm, its usage as a Lyapunov function is equivalent to the use of an appropriate norm. For the discussion of asymptotic stability and for most questions encountered at the study of numerical methods the restriction to norms does not seem hindering, but norms are sometimes too crude for the delicate case of non-asymptotic stability.

In the problems with periodic solutions that we have encountered so far, there has been a whole family of periodic orbits. There is another type of periodic solutions in nonlinear problems, called **limit cycles**.

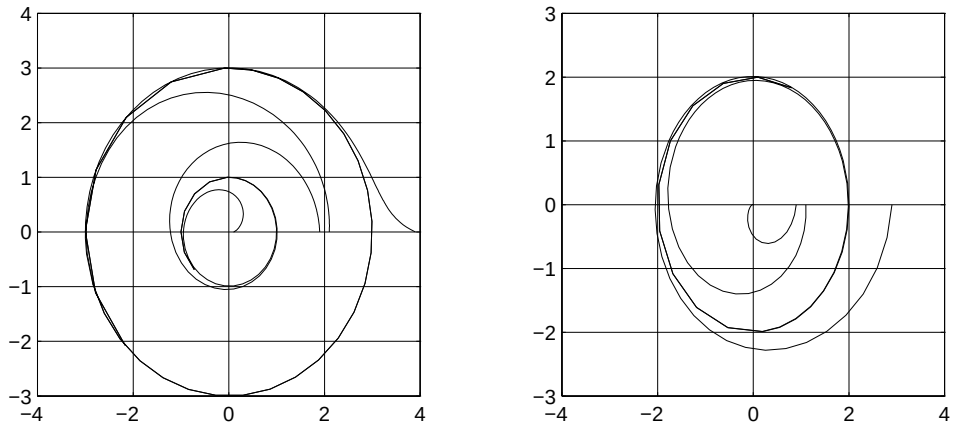

Figure 13.8.1. *(a) Two stable limit cycles for Example 13.8.7. Integration forwards in time; motion is counter-clockwise. (b) An unstable limit cycle is found by integration backwards in time; motion is clockwise*

Example 13.8.7

The complex differential equation

$$\dot{z} = iz - 0.5z(|z| - 1)(|z| - 2)(|z| - 3),$$

which can be written as a system of two real differential equations, provides a simple example of limit cycles. Set $z = re^{i\phi}$, and separate real and imaginary parts: $r' = -0.5r(r-1)(r-2)(r-3)$, $\phi' = 1$. The first of these equations can be discussed like Example **??**. There are unstable critical points at 0 and 2, while 1 and 3 are stable critical points. The second equation shows that a uniform rotation is superimposed on the development of $r(t)$.

So, for $0 < |z(0)| < 1$, r increases towards 1, while ϕ increases monotonically. The orbit is a spiral that approaches the unit circle from the inside. See the left part of Fig. 13.8.1.

Similarly, for $1 < |z(0)| < 2$ the orbits approach the unit circle from the outside. The unit circle is therefore said to be a stable limit cycle. The circle $|z| = 3$ is also a stable limit cycle, for initial values $|z(0)| > 2$. The point $r = 2$ is an unstable critical point for the real differential equation for r. Therefore the circle $|z| = 2$ becomes an unstable limit cycle for the complex differential equation.

If the problem is run backwards in time, the situation is opposite; the circle $|z| = 2$ becomes a stable limit cycle, while the other two become unstable. See the right part of Fig. 13.8.1. □

The previous simple example should not make you believe that limit cycles are always circles. Fig. 13.8.2 shows the limit cycle of a famous problem in chemistry, named the Brusselator problem that is described by the system,

$$y_1' = a + y_1^2 y_2 - (b+1)y_1, \quad y_2' = by_1 - y_1^2 y_2,$$

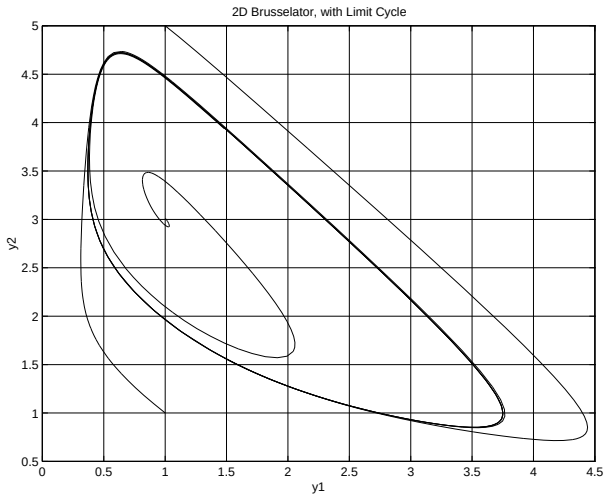

Figure 13.8.2. *The Brusselator problem with a limit cycle, see Exercise C6.*

see also Exercise C6. A derivation is found in [20, p. 112].

Limit cycles play a central role in the theory of autonomous ODEs in $\mathbf{R^2}$, due to the following classical theorem from around 1900. We quote the formulation of [20, p. 113]. A proof is found in [6].

Theorem 13.8.8 (The Poincaré–Bendixson Theorem)**.**
Each bounded solution of an autonomous system $\dot{y} = f(y)$ *in* $\mathbf{R^2}$ *must*

(i) *tend to a critical point for an infinity of points* $t_i \to \infty$*; or*

(ii) *be periodic; or*

(iii) *tend to a limit cycle.*

The complicated formulation of the first alternative is related to the possibility of a critical point on the limit cycle.

There may be limit cycles also in larger systems of ODEs. For example, a three- dimensional version of the Brusselator is discussed in [20, p. 117], where an introduction to **Hopf bifurcations** is also given. A stiff 3D problem with a limit cycle is the Oregonator, see next subsection, and exercise C9.

There is, however, no general result like the Poincaré–Bendixson theorem. For instance, a motion in three dimensions can become **chaotic**, and its limiting set can have a fractal structure, a so-called **strange attractor**. A famous example of this is due to E. N. Lorenz, see Fig. 13.8.3 and Exercise C7. A very illuminating discussion and a numerical study is presented by Hairer et al. [20, p. 120 ff.].

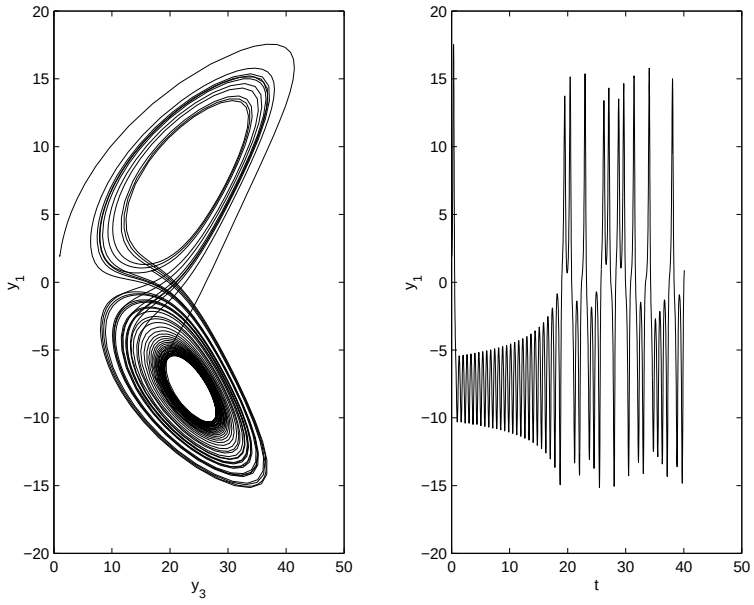

Figure 13.8.3. *The Lorenz problem with a butterfly-like strange attractor. The equations are given in Exercise C7.*

13.8.3 Singular Perturbations and Separably Stiff Equations.

The iteration used in the proof of Theorem 13.1.3 was based on the rewriting of the differential equation $dy/dt = f(y)$ in terms of an integral operator that is bounded and, under appropriate conditions, contractive. Would it have been possible to use a different iterative scheme? For example, let $y_0(t)$ be a solution of the algebraic equation $f(t, y(t)) = 0$, and define

$$f(t, y_i(t)) = \frac{dy_{i-1}(t)}{dt}, \quad i = 1,\ 2,\ 3\ldots \tag{13.8.5}$$

(with appropriate conditions that make $y_i(t)$ unique).

The answer is *"No"*, because the differential operator $\frac{d}{dt}$ is an unbounded operator, at least in any function space that contains exponentials $y(t) = e^{\alpha t}$ for some arbitrarily large values of $|\alpha|$, (since for such functions $\|\frac{d}{dt}y\| = |\alpha|\|y\|$). The differential operator $\frac{d}{dt}$ is a bounded operator in some particular function spaces only, such as the space of polynomials of a fixed degree or a space of so-called band-limited functions, i.e. Fourier transforms of functions with a compact support.

Nevertheless the iterative scheme (13.8.5) is interesting for special classes of differential systems, called **separably stiff systems** or singular perturbation problems. First consider a special case:

$$\dot{y} = \lambda(y - ce^{\alpha t}), \quad \alpha \in \mathbf{C}. \tag{13.8.6}$$

This equation has the particular solution $y(t) = c(1-\alpha/\lambda)^{-1}e^{\alpha t}$ Let $y_0(t) = ce^{\alpha t}$. Then $y_i(t) = c_i e^{\alpha t}$ where $c_0 = c$, $\lambda(c_i - c)\alpha e^{\alpha t} = c_{i-1}\alpha e^{\alpha t}$, $i = 1, 2, 3, \ldots$, i.e., $c_i = c + c_{i-1}\alpha/\lambda$. By induction

$$c_i = c\left(1 + \frac{\alpha}{\lambda} + \frac{\alpha^2}{\lambda^2} + \ldots + \frac{\alpha^i}{\lambda^i}\right) \to (1 - \frac{\alpha}{\lambda})^{-1},$$

if $|\alpha| < |\lambda|$*, i.e.,* y_i *converges towards the particular solution mentioned above.* Note that, if $\Re(\lambda) < 0$ any solution of (13.8.6) will approach this particular solution at the rate of $e^{\lambda t}$ as t increases.

When the iteration formula (13.8.5) is applied to the more general equation

$$\dot{y} = \lambda(y - g(t)), \quad y(a) = c, \quad \Re(\lambda) < 0, \tag{13.8.7}$$

with $y_0(t) = g(t)$, each iteration yields a new term in the expansion,

$$y(t) \sim g(t) + \lambda^{-1}g'(t) + \lambda^{-2}g''(t) + \ldots + \lambda^{-k+1}g^{(k-1)}(t) + \ldots. \tag{13.8.8}$$

We first note that this is (at most) a particular solution of (13.8.7), but it can then approximate any solution when $\Re(\lambda)(t-a) \ll -1$. Moreover, it is common that this expansion is *semiconvergent* only (in the sense of Sec. 3.1.8). The expansion is to be truncated, when the terms do not decrease rapidly any longer, in absolute value. Another derivation of the expansion is indicated in Problem 13.7.8, together with a remainder. It shows that the first neglected term provides a useful error estimate, as long as the second neglected term is much smaller, in absolute value, than the first, i.e., if $|g^{(k)}(t)/g^{(k+1)}(t)| \gg |1/\lambda|$ or, in other words, if the local time scale $\tau_q(t; g, k) \gg |1/\lambda|$. It is worth noting that it often happens, in the applications we have in mind, that this expansion is *not* alternating, and that, for a fixed t, $\tau_q(t; g, k) \to 0$ slowly, as $k \to \infty$. Take, for example, $g(t) = 1/t$. Then $\tau_q(t; g, k) = t/(k+1)$, and the expansion reads

$$y(t) \sim \frac{1}{t} - \frac{1}{\lambda t^2} + \frac{2!}{\lambda^2 t^3} - \ldots \pm \frac{(k-1)!}{\lambda^{k-1}t^k} + \ldots.$$

This reminds of the alternating semiconvergent expansion studied in Example 3.2.14, but now the most interesting case is when $\lambda < 0$, $t > 0$, i.e., when all terms are positive. Now the first neglected term is a useful error estimate only if $k+1 \ll |\lambda t|$.

Note, assume that $\lambda < 0$ that

$$g(t + \lambda^{-1}) \approx g(t) + \lambda^{-1}g'(t) + \frac{1}{2}\lambda^{-2}g''(t) + \ldots.$$

By comparison of this with the expansion (13.8.8), we find that the graph of $y(t)$ can be looked upon as a translation $g(t - |\lambda|^{-1})$ of the graph of $g(t)$, with an error of $O\big(|\lambda\tau|^{-2}\big)$, where τ is the local time scale of $g(t)$, provided that the transient proportional to $e^{\lambda(t-a)}$ has died out, i.e. for $t - a > |\lambda|^{-1}\ln 1/\text{TOL}$, or $t - a > |\lambda|^{-1}\ln|\lambda|$ depending on the criterion used for deciding that a term is negligible.

Equation (13.8.8) is valid also if we substitute a matrix A for λ, such that $\Re\lambda < 0$ and $|\lambda\tau| \gg 1$ for all eigenvalues of A. Now consider a nonlinear system, where ϵ corresponds to $1/|\lambda|$,

$$\epsilon\dot{y} = F(t, y), \quad \epsilon > 0. \tag{13.8.9}$$

Assume that $F(t, y)$ and its partial derivatives of low order are bounded, and that there exists a solution $y_0(t)$ of **the reduced problem** $F(t, y) = 0$, such that the Jacobian $J(t) = \partial F/\partial y$ at $y = y_0(t)$ is non-singular and has a negative logarithmic norm $\mu(J(t)) < \mu^* < 0$. (In fact, the non-singularity follows from the latter assumption, see Theorem **??**.) Let then $y_1(t)$ be defined by the equation

$$\epsilon y_0'(t) = F(t, y_1(t)). \tag{13.8.10}$$

Take the total derivative of the equation $F(t, y_0(t)) = 0$ with respect to t, i.e., $\partial F/\partial t + J(t)y_0' = 0$. This determines $y_0'(t)$. By Taylor's formula, we can now write (13.8.10) as follows:

$$\epsilon y_0'(t) = F(t, y_1(t)) - F(t, y_0(t)) = J(t)(y_1(t) - y_0(t)) + O(\|y_1 - y_0\|^2). \tag{13.8.11}$$

It can then be shown that

$$y_1(t) = y_0(t) + \epsilon J(t)^{-1} y_0'(t) + O(\epsilon^2). \tag{13.8.12}$$

and that $y_2(t) - y_1(t) = O(\epsilon^2)$. It is then conceivable that $y(t) - y_1(t) = O(\epsilon^2)$, when a transient, that decays faster than $e^{\mu^*(t-a)}$, has died out.

If (13.8.9) is a *single nonlinear equation*, $J(t) \leq \mu^* < 0$ then we conclude from (13.8.12) that

$$y(t) = y_1(t) + O(\epsilon^2) = y_0(t - \epsilon/|J(t)|) + O(\epsilon^2) \tag{13.8.13}$$

i.e. *the graph of $y(t)$ is approximately a translation of the graph of $y_0(t)$, after the transient, but the delay $\epsilon/|J(t)|$ depends on time.*

A few things are worth to be noted for the case of a single equation:

(a) The left hand side of (13.8.11) is smaller than the absolute value of each of the terms $J(t)y_1(t)$, $J(t)y_0(t)$, by a factor that can be described as the ratio of the smallest local time constant of the system to the local time scale of the approximate solution $y_0(t)$. So, when the solution of a single differential equation has become stiff, the differential equation describes *a moving approximate balance between positive and negative terms on the right hand side.*

(b) The graphs of $y(t)$ and $y_0(t)$ intersect at points where $\dot{y}(t) = 0$. (Why is that no contradiction to "the translation point of view"?)

(c) The asymptotics is valid only when the delay $\epsilon/|J(t)|$ is much smaller than the local time scale. In particular, it becomes unreliable when t approaches a point, where $J(t)$ is zero.

These points of view, applied to individual equations of a system, where the other variables are considered as driving terms, can often provide a good insight in what is going on in the system. This makes it natural *to extend the use of the term stiffness to individual variables or subsystems of a larger system.*

The function defined by a few iterations according to (13.8.5) is close to a solution of (13.8.10) if $\epsilon \ll \tau(t)$. If $\mu^* < 0$, it is **attracting** to other solutions of the same system, as t increases. If the Jacobian has a positive eigenvalue, the iterative process may still be semi-convergent, but the resulting function is **repelling** to the other solutions of the system. In non-linear problems the expressions for the successive iterates quickly become messy, but Problem 13.7.8 indicates that rather simple bounds can be obtained.

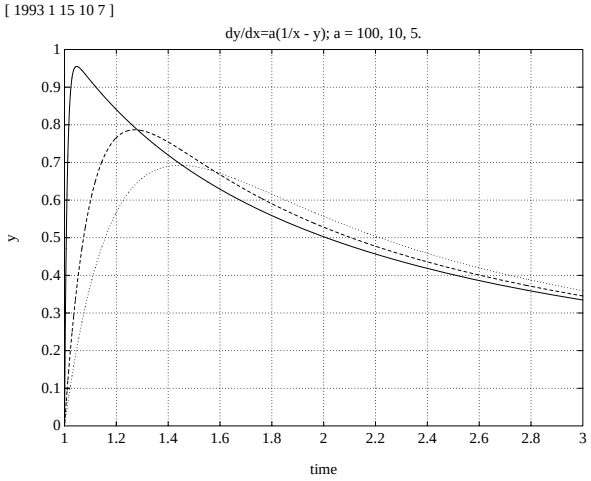

Figure 13.8.4. *The solution of* $\dot{y} = a(1/t - y)$, $y(1) = 0$ *for* $a = 100,\ 10,\ 5$.

Example 13.8.9

Fig. 13.8.3 shows the solution of $\dot{y} = a(1/t - y)$, $y(1) = 0$ for $a = 100,\ 10,\ 5$. The curve $y = 1/t$ is not drawn, but it proceeds from the upper left corner through the top point of the leftmost curve and continues then immediately to the left of the leftmost curve. It is seen that, after the transient, the other two curves are delayed about a^{-1} after the left curve, as predicted above. □

Example 13.8.10

Fig. 13.8.2 shows the solution of "the rectifier equation"

$$\epsilon \frac{dy}{dt} = \big((\sin 2\pi t)^2 - y^2\big), \quad y(0.1) = 0.5, \tag{13.8.14}$$

for $\epsilon = 0.0001,\ 0.01,\ 0.02$. The reduced problem has two solutions: $y_0(t) = \pm \sin 2\pi t$. Note that $f_y(t, y) = -2y/\epsilon$, hence $y_0(t) = |\sin 2\pi t|$ is attracting, while the branch $y_0(t) = -|\sin 2\pi t|$ is repelling. The points where $y_0(t) = 0$ are branch points, where

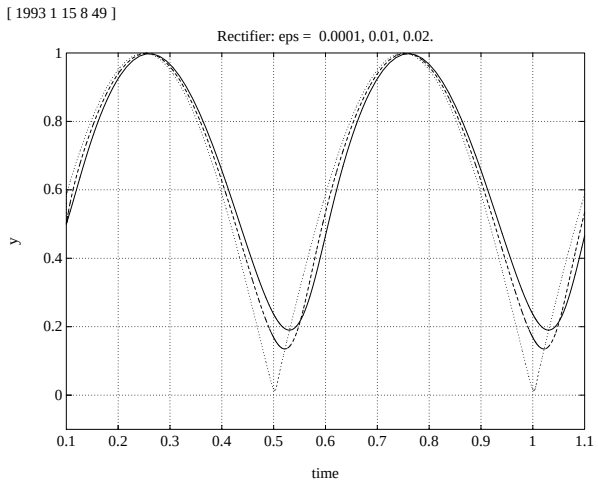

Figure 13.8.5. *Solution of "the rectifier equation".*

$J(t) = 0$. The smallest time constant is $\epsilon/|y|$, the local time scale is approximately $\frac{1}{2\pi}$, and $y(t) \approx |\sin 2\pi(t - \epsilon/(2y))|$, except in the transient and in the neighborhood og the branch-points. It appears that the asymptotics becomes unreliable when $y = O(\sqrt{\epsilon})$, where $y(t)$ has a minimum, with a sharp turn.

It may be conjectured that

$$\min y(t) \sim c\sqrt{\epsilon}, \quad (\epsilon \to 0).$$

It is not quite trivial to determine c by an asymptotic analysis. If one does it numerically instead, by running the problem for different values of ϵ (Computer exercise 13.7.5), one should avoid to choose TOL $> \epsilon$ (say). It has happened also with widely used programs for stiff problems and a careless choice of TOL that the computed solution becomes close to $\sin 2\pi t$ instead of $|\sin 2\pi t|$. This can be very misleading, in particular since the erroneous solution is much smoother and looks more trustworthy than the correct solution! Roughly speaking, the step size has become so big for $t > 0.25$ that the computed points never come close enough to $y = 0$, for the program to recognize the change of local time scale. It may also fail to recognize the repelling nature of the wrong branch for $y < 0$, since $\mathcal{S}$ of many numerical methods for stiff problems contains part of the right half plane where $\Re\lambda h > 0$. □

Many problems in Chemical Kinetics and other branches of Applied Mathematics can be brought to the form

$$\frac{dy_i}{dt} = \epsilon_i^{-1} F_i(t, y_1, y_2, \ldots, y_m), \quad y_i \in \mathbf{R}^{n_i}, \tag{13.8.15}$$

$$0 < \epsilon_1 \ll \epsilon_2 \ll \ldots \epsilon_m, \quad n = n_1 + n_2 + \ldots + n_m,$$

by appropriate scaling transformations. They are called **separably stiff problems**.

The ϵ_i are assumed to be constant during the time interval under consideration, but there are problems, where one had better divide the given interval into two, with different scaling and partitioning. Set $E = \text{blockdiag}[\epsilon_1 I, \epsilon_2 I, \ldots, \epsilon_m I]$, where I denotes unit matrices of various orders, n_1, $n_2, \ldots$. Finally, set $\epsilon = \max(\epsilon_i/\epsilon_{i+1})$.

The Jacobian of the ODE system, becomes $J = E^{-1}F\prime(y)$ (with an obvious notation). We assume that $F\prime(y)$ is a "bounded" matrix, i.e. its elements are small compared to the ratios $\epsilon_{i+1}/\epsilon_i$. Matrices of this type are called *graded matrices*. We further assume that there exists a factorization, $F\prime = LR$, where L, R are, respectively, lower and upper *block triangular* matrices, where the partitioning is as indicated in the ODE system (13.8.15). The diagonal blocks of L should be unit matrices.

A **fundamental assumption** is that *the eigenvalues of R have strictly negative real parts*, except for the eigenvalues of the lowest block. Then, one can show, (Dahlquist [8] that *if ϵ is small enough, the eigenvalues of the Jacobian are close to the eigenvalues of the diagonal block matrices $\epsilon_i^{-1}R_i$.*

It is too technical to discuss in detail *how small is "small"?*. Essentially, the smallest eigenvalue of $\epsilon_i^{-1}R_i$ should be larger than the largest eigenvalue of $\epsilon_{i+1}^{-1}R_{i+1}$, for $i = 1 : m-1$, by a factor that mainly depends on the size of the elements in the first subdiagonal blocks of L.

The system (13.8.15) can also be written as follows:

$$\epsilon_i \frac{dy_i}{dt} = F_i(t, y_1, y_2, \ldots, y_m), \quad y_i \in \mathbf{R}^{n_i}, \tag{13.8.16}$$

This is a generalisation of the classical case of *singular perturbation*. The classical case will now be discussed for an autonomous system,

$$\begin{aligned} \epsilon \dot{y}_1 &= F_1(y_1, y_2), \qquad y_1 \in \mathbf{R}^{n_1} \\ \dot{y}_2 &= F_2(y_1, y_2), \qquad y_2 \in \mathbf{R}^{n_2} \end{aligned} \tag{13.8.17}$$

We refer to this as the *complete problem*, in contrast to the *reduced problem* that is obtained for $\epsilon = 0$.

$$\begin{aligned} 0 &= F_1(\tilde{y}_1, \tilde{y}_2), \qquad \tilde{y}_1 \in \mathbf{R}^{n_1} \\ \frac{d\tilde{y}_2}{dt} &= F_2(\tilde{y}_1, \tilde{y}_2), \qquad \tilde{y}_2 \in \mathbf{R}^{n_2} \end{aligned} \tag{13.8.18}$$

This is a *differential-algebraic* system of equations (DAE). We want to estimate the difference $x = y - \tilde{y}$, and employ therefore results of a generalisation of the theory of logarithmic norms. We shall only sketch the derivations, which are similar to those in the DB manuscript, Sec 12.2.3, that was handed out in Part 1 of this course. Set $\partial F_i/\partial y_j = A_{ij}$, and **assume** that

$$\mu(A_{ii}) \le \alpha_{ii}, \quad \|A_{ij}\| \le \alpha_{ij} \text{ if } i \ne j.$$

We further *assume* that $\alpha_{ii} < 0$, $i = 1 : m-1$. This implies the fundamental assumption introduced above. It is often useful to let the bounds α_{ij} depend on t but, for the sake of simplicity, we here give them constant values that should

be valid when t is large. We now subtract (13.8.18) from (13.8.17), and introduce the notation $\bar{A}_{ij}$ for a neighborhood average of the sub-Jacobian A_{ij}, like in Eqn. (12.2.4) of DB. We obtain, after some simplification,

$$\begin{aligned} \epsilon\dot{x}_1 &= \bar{A}_{11}x_1 + \bar{A}_{12}x_2 - \epsilon\dot{\tilde{y}}_1 \\ \dot{x}_2 &= \bar{A}_{21}x_1 + \bar{A}_{22}x_2. \end{aligned} \tag{13.8.19}$$

The block triangular factorization mentioned above here yields

$$R_1 = A_{11}, \quad R_2 = A_{22} - A_{21}A_{11}^{-1}A_{12}.$$

Good approximations to the eigenvalues of the full Jacobian are given by the eigenvalues of the smaller matrices $\epsilon^{-1}A_{11}$ and R_2. The latter approximation is independent of ϵ.

Now we return to Eq.(13.8.19). Let η_1 be an upper bound of $\|\dot{\tilde{y}}_1\|$. We find, after proceeding similarly to the derivation of Theorem 12.1.2 that $\|x_i\| \leq \xi_i$, where

$$\begin{aligned} eps\dot{\xi}_1 &= \alpha_{11}\xi_1 + \alpha_{12}\xi_2 + \epsilon\eta_1 \\ \dot{\xi}_2 &= \alpha_{21}\xi_1 + \alpha_{22}\xi_2. \end{aligned} \tag{13.8.20}$$

Set

$$\mathrm{A} = \begin{pmatrix} \epsilon^{-1}\alpha_{11} & \epsilon^{-1}\alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{pmatrix}.$$

We now assume that the eigenvalues of A *have negative real parts.* Then $(\xi_1(t), \xi_2(t))^T \to \xi^* = -\mathrm{A}^{-1}(\eta_1, 0)^T = O(\epsilon)$, as $t \to \infty$. A vector inequality, analogous to Eq.(12.1.6), valid for finite t, can also be obtained. It can be shown that if $\tilde{y}_2$ has the correct initial value, i.e. if $\xi_2(t_0) = 0$, then $\xi_2(t) = O(\epsilon)\ \forall t > t_0$. By means of the first row of (7), we can then obtain the following important result.

Theorem 13.8.11.

Assume that $\xi_1(0) = O(1)$. Outside a fast transient (or boundary layer) of length (width) $(\epsilon/|\alpha_{11}|)\ln(1/\epsilon)$, the solution of the reduced problem becomes an $O(\epsilon)$-accurate solution of the complete problem.

The behavior of the solution shows that, after the first transient, there is no need for short time steps, even with a classical explicit numerical method, if the reduced problem is applied, and if an $O(\epsilon)$-accurate solution is satisfactory. Problems 17 and 18 of Sec.12.2 indicate a possibility to obtain corrections to the solution of the reduced problem, i.e. more terms in a singular perturbation expansion of the "outer solution". Such an expansion easily becomes messy, and the use of a numerical method with good stability properties is recommended, so that a step size h can be used that is, in principle, determined by the local time scale of the orbit, i.e. $\|hf'(y)\|$ should be allowed to be large.

It is, however, not necessary to apply such a method to the second row of (13.8.17), because that is not a stiff equation. For this reason, such systems are called **separably stiff problems**. The general ideas of this note can be generalised

to systems of the type in (13.8.15), which may also be called separably stiff, although the frontier between "the stiff part" and the "non-stiff part" changes with t.

Recall the assumption that *the eigenvalues should have strictly negative real parts* except for the eigenvalues related to the last lines of (13.8.15) or (13.8.17). There are, however, problems where, this assumption is not valid in certain subintervals, see e.g., Example D20.2 (Oregonator). An important requirement of a program for stiff equations, is that it is able to detect when this happens, and to reduce the step size appropriately. The program described below seems to be rather reliable in this respect.

It is often of interest to find out which variables in an ODE system will become zero in the limit, without extensive calculations. The following three propositions may be useful to that purpose. The Comparison Theorem, DB Sec.12.2.2, is then useful for finding the speed of convergence, and also for other conclusion concerning the behavior of a variable. The proofs are omitted.

Proposition 13.8.12.

Let $g : \mathbf{R} \to \mathbf{R}$ be a monotonic and bounded function for all $t \geq 0$. Then $g(t)$ converges to a finite limit as $t \to \infty$. If,in addition, $\ddot{g}(t)$ is bounded for all $t \geq 0$, then $\dot{g}(t) \to 0$ as $t \to \infty$.

This proposition can be applied to an individual equation of a differential system. The other variables are then treated like given functions of t. The assumed boundedness of g can often be established in various ways, e.g. by a combination of the Positivity Theorem (see DB Sec.12.2.2) and linear relations between the variables (see the example below). The boundedness of g'' can often be established by means of the following proposition. It covers, for example, autonomous systems, $y' = f(y)$, where the components of $f(y)$ are polynomials of the components of y — an important class of differential systems in Chemical Kinetics, Mathematical Ecology and many other areas.

Proposition 13.8.13.

Let $y(t)$ be a solution of the ODE system $dy/dt = f(t, y)$. Assume that $\|y(t)\| \leq K \ \forall t \geq 0$. Assume that f, $\partial f/\partial y$, $\partial f/\partial t$ are bounded for $\|y\| \leq K$, uniformly in t. Then $\ddot{y}(t)$ is bounded for all $t \geq 0$.

Note that $\ddot{y} = f_y \dot{y} + f_t$. (A stronger version of this proposition may sometimes be needed.)

Proposition 13.8.14.

Let a, δ be positive constants, let P be a real-valued function such that $P(0) = 0$, $P(u) > 0$ for $u > 0$, and let u be a positive-valued function that satisfies an initial value problem of the form $\dot{u} = -f_1(t, u) + f_2(t, u), \quad u(t_0) > 0$, where $f_1(t, u) \geq P(u)t^{-1+a}$, and $f_2(t, u) = o(t^{-1+a})$, uniformly when $u \in [0, \delta]$.-Then $u(t) \to 0$, and $\dot{u}(t) = o(t^{-1+a})$ as $t \to \infty$.

Example 13.8.15

The following set of autocatalytic chemical reactions [44] is based on a study of H.H.Robertson (1966). It is one of the standard test problems for numerical methods for stiff ODEs, sometimes with different coefficients.

$$Y \xrightarrow{k_1} Z; X + Z \xrightarrow{k_2} Y + Z;$$

$$X + X \xrightarrow{k_3} Z;$$

Consider the case with

$$k_1 = 0.04; \quad k_2 = 2\,10^4; \quad k_3 = 1.5\,10^7.$$

A system of ODEs for this system of reactions is derived, according to the law of Mass Action, see Problem P3 of Sec.,13.1. More general accounts of this are found in Hairer, Wanner and Nörsett [], p.115 or in Lin-Segel, [], Sec. 10.1. The former text also mentions a number of simplifications to be done in common special situations.

Denote by z_1, z_2, z_3 the concentrations of,respectively, the species X, Y, Z. Then the differential equations with initial conditions read,

$$\dot{z}_1 = -k_2 z_1 z_3 - 2\,k_3 z_1^2 + k_1 z_2 \tag{13.8.21}$$

$$\dot{z}_2 = k_2 z_1 z_3 - k_1 z_2 \tag{13.8.22}$$

$$\dot{z}_3 = k_3 z_1^2. \tag{13.8.23}$$

$$z_1(0) = 0 \qquad z_2(0) = 1 \qquad z_3(0) = 0.$$

Note that $\dot{z}_1 + \dot{z}_2 + 2\dot{z}_3 = 0$, hence we have *a linear invariant*: $z_1(t) + z_2(t) + 2z_3(t) =$ const. $= 1$.

Put $z_1 = 10^{-4} y_1$, $z_2 = y_2$, $z_3 = y_3$, $10^{-4} = \epsilon$. Then the equations become (Exercise

$$\begin{aligned} \epsilon \dot{y}_1 &= -2y_1 y_3 - 0.3y_1^2 + 0.04y_2 \\ \dot{y}_2 &= 2y_1 y_3 - 0.04y_2 \\ \dot{y}_3 &= 0.15y_1^2 \end{aligned} \tag{13.8.24}$$

with initial conditions

$$y_1(0) = 0 \qquad y_2(0) = 1 \qquad y_3(0) = 0.$$

Now the linear relation reads, $\epsilon y_1(t) + y_2(t) + 2y_3(t) = 1$. By the Positivity Theorem (Theorem 3.1.5), $y_i(t) > 0 \; \forall \; t > 0$, $i = 1, 2, 3$. By the linear relation, we then obtain,

$$0 < y_1(t) \leq \epsilon^{-1}, \quad 0 < y_2(t) \leq 1, \quad 0 < y_3(t) \leq 0.5, \quad \forall t > 0. \tag{9}$$

(A better bound for $y_1(t)$ is obtained below.)

In the terminology of singular perturbations, we first look for the *inner solution*, approximately valid during the transient. Set $t = \epsilon\tau$, $y_i(t) \approx \hat{y}_i(\tau)$. Let

[44] This is a little different from Problem P4 of Sec. 13.1.

$\hat{y}_2(\tau) = 1$, $\hat{y}_3(\tau) = 0$, i.e. they are kept equal to their initial values during the transient. Then

$$d\hat{y}_1/d\tau = 0.04 - 0.3\hat{y}_1^2,$$

hence $\hat{y}_1(\tau) \to \sqrt{0.04/0.3} = 0.365148$. Note that the partial derivative of the right hand side equals $-0.6\hat{y}_1 \to -0.22$, hence the length of the transient is approximately $-\epsilon \ln \epsilon/0.22 = 42\epsilon$. Another estimate of the width is obtained by setting $\hat{y}_1(\tau) \approx 0.04\tau$ until $0.04 - 0.3\hat{y}_1^2 = 0$, which happens at $\tau \approx 9\epsilon$. (Sketch roughly the graph of $\hat{y}_1(\tau)$, and interpret these two estimates.)

Since $d\hat{y}_2/d\tau \geq -\epsilon 0.04\hat{y}_2 \approx -0.04\epsilon$, $\hat{y}_2$ decreases only by 1.7ϵ during the transient. Similarly $\hat{y}_3 \leq \epsilon$ during the transient. The Jacobian of the system is a graded matrix:

$$J = \begin{pmatrix} -(2y_3 + 0.6y_1)\epsilon^{-1} & 0.04\epsilon^{-1} & -2y_1\epsilon^{-1} \\ 2y_3 & -0.04 & 2y_1 \\ 0.3y_1 & 0 & 0 \end{pmatrix}.$$

Note that this matrix is singular — we have the same dependence between the rows in the Jacobian as in the ODE system. In the terminology of the discussion above $A_{11} = -(2y_3 + 0.6y_1) < 0$ and the numerically large eigenvalue is approximately $\epsilon^{-1}A_{11}$. The rank is 2, except that it becomes 1, when $y_1 = 0$, and we shall see that this happens at the critical point that the solution converges to. This turns out to be quite common in chemical systems and has as a consequence that the convergence towards the critical point is not exponential but only like t^{-c} for some positive number c. We shall see that $c = 1$ in this example.

The matrix of the non-linear system to be solved at every step in a numerical solution is $I - h\gamma J$ (see the description of the BDF method below), and this is, fortunately, non-singular for any reasonable step size h!

We shall now consider the *outer solution*. We therefore modify (13.8.24) by putting $\epsilon = 0$ and ignoring the initial condition for y_1. We drop the subscript O that Lin and Segel use for the outer solution. Hence y_1 is the positive root of the quadratic

$$-2y_3y_1 - 0.3y_1^2 + 0.04y_2 = 0, \quad \text{where } 2y_3 = 1 - y_2,$$

i.e.

$$y_1 = \frac{0.08y_2}{(1 - y_2) + \sqrt{(1 - y_2)^2 + 0.048y_2}}. \tag{13.8.25}$$

Note that this bound for the outer solution y_1 is much better than (9). It is derived for the reduced problem but, by the above theorem, it also holds (approximately) for small positive ϵ. Since $y_2 = 1 - 2y_3$, we have $\dot{y}_2 = -2\dot{y}_3$, hence

$$\dot{y}_2 = -0.3y_1^2, \qquad y_2(0) = 1. \tag{13.8.26}$$

If we plug in the above expression for y_1 here, we obtain a single separable ODE that can be solved by quadrature, at least numerically. We omit the details. Instead, we note that the assumptions of Prop.2 are satisfied by (13.8.26). So $\ddot{y}$ is bounded. Then, by Prop.1, $\dot{y}_2 \to 0$, hence $y_1 \to 0$. It then follows from the quadratic equation that $y_2 \to 0$. Finally, the linear relation now shows that $y_3 \to 0.5$.

We also see from the quadratic (or its solution) that $y_1/y_2 \to 0.04$, i.e., $y_2 \approx 25y_1$,[45] as the system approaches the critical point. Then, Eqn.(10) becomes, $\dot{y}_2 \sim -0.48\,10^{-3}y_2^2$. The solutions are, for large values of t,

$$y_2 \sim 2083.3/t, \quad y_1 \sim 83.33/t, \quad (y_1 \approx 0.04y_2). \tag{13.8.27}$$

It can be shown that the smaller non-zero eigenvalue of J is approximately $\lambda = -0.024y_1$ as the critical point is approached. Note that $\lambda \to 0$ from the negative side. If the system like (13.8.24) is integrated numerically, however, it can happen that, for very large t, y_1 may become negative, due to truncation and rounding errors, and then λ becomes positive. Instabilities due to this have been observed in extremely long runs, in particular with a large tolerance.

The conclusions of the previous paragraphs were concerned with the *reduced* problem, and they are therefore at least $O(\epsilon)$-accurate for the complete problem. We can, however, obtain sharper results by discussing the *complete* problem (13.8.24) directly. The equation for $\dot{y}_3$ shows that y_3 is monotonic, and we already know that y_3 and $\ddot{y}$ are bounded. Then, by Proposition 1, $\dot{y}_3 \to 0$, and hence $y_1 \to 0$. By the application of Prop.3 to the equation for $\dot{y}_2$, we then find that $y_2 \to 0$, and finally the linear relation shows that $y_3 \to 0.5$. [46] □

** Use the following material for text and problem on the Oregonator.

Study the Oregonator equations (Problem 1), and calculate its Jacobian. It is interesting to see where and when the numerically large eigenvalue changes sign, and to relate that to the behavior of the solution and the step size. Report your observations.

Run it first with `tend=0.2`, in order to see the details during an "outbreak". Then run it with `tend=20`, to see the periodic nature of the solution of this problem. (The latter run may take about 5 times longer time than the former.) Make another set of runs in order to estimate the accuracy.

Make plots (or hand drawings) that show the step size variation, one with details within the outbreak and the other for the large features. In what subintervals would you consider the system to be stiff? Determine the period of the solution to a few per cents accuracy. NOTE. This model made some sensation in Theoretical Chemistry , and it also caused some trouble for some of the first programs for stiff problems, which were not prepared for the situation, that a system can become non-stiff again, after a stiff interval.

Move this to Ch 13.4:

On the BDF Method and the Program BDF

Program BDF is a solver for stiff systems of ODEs,

$$y' = f(t, y), \tag{13.8.28}$$

[45] check that it should not be 12.5.

[46] In my lecture notes it is remarked that y_1 bounded, Prop.2applic. implies y_2 bounded. I must look at this.

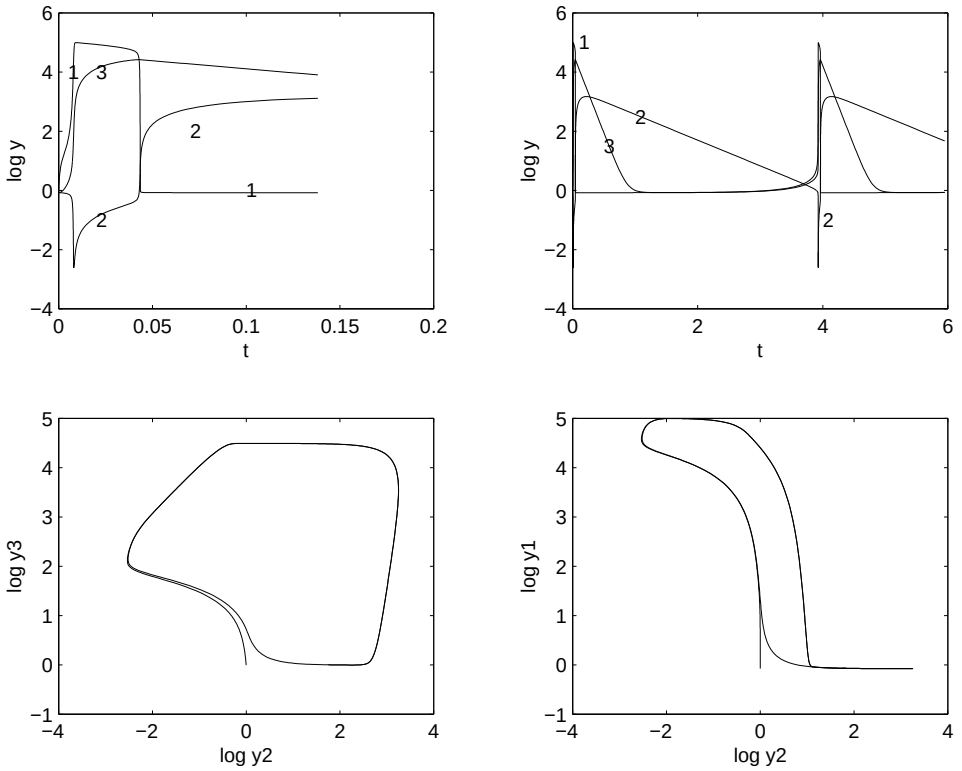

Figure 13.8.6. *The Oregonator problem with a limit cycle, see Example* **??** *and Exercise C6.*

based on the widely used *Backwards Differentiation Formula* (BDF), For constant stepsize this reads,

$$\left(\nabla + \frac{1}{2}\nabla^2 + \ldots + \frac{1}{k}\nabla^k\right) y_n = hf(t_n, y_n). \tag{13.8.29}$$

Here ∇ is the backwards difference operator, defined for functions and sequences by the relations,

$$\nabla y(t) = y(t) - y(t-h), \quad \nabla y_n = y_n - y_{n-1}.$$

The two definitions are identical if $y_0 = y(t_0)$, $y_n = y(t_0 + nh)$, $n = 1, 2, 3, \ldots$.

We shall use some results and some terminology, developed in the following three papers, which were handed out in Part I of this course:
(i) DB, Sec 4.6, *Calculus of Difference Operators.*
(ii) G. Dahlquist, *The Integration Method of the OLEG Package*,
(iii) Th. Ekman, *A Portable Interactive Package for the Numerical Treatment of ODEs.*, i.e. the OLEG report. The BDF method is a *one-leg collocation method.*

Our implementation is a modification of the OLEG package. The step size h is automatically controlled, but (similarly to OLEG) it (where $p = k$ for BDF, while $p = k + 1$ for OLEG, see below). Note that the left hand side of (2) is a truncated operator power series for the differential operator $hD = \ln E = -\ln(1 - \nabla)$. It therefore equals the derivative at t_n of the local k'th degree polynomial $Y(t)$, which is determined by the conditions $Y(t_n - jh) = y_{n-j}$, $j = 0 : k$. In OLEG the collocation parameter is chosen optimally, in the sense that the order of accuracy p becomes as high as possible for an one-leg k-step method, i.e. $p = k+1$. In BDF, the collocation parameter is chosen more with the consideration of the stability properties desirable for stiff problems, see the enclosed plots of stability regions. *The order of accuracy is $p = k$ for the k-step BDF method.* For non-stiff problems, it is therefore less efficient than the OLEG method with the same value of k. On the other hand OLEG is unable to treat stiff problems. The stability region of the k-step BDF method is, for every k, the exterior of some closed bounded region, but *it is zero-stable for $k \leq 6$ only.* Our implementation can be used for $k \leq 5$, but we do not recommend it for $k = 5$, unless very high accuracy is requested. The most important difference from the OLEG package is the treatment of algebraic system of equations encountered at every step. By (13.8.29), it is of the form,

$$y_n - h_n \gamma f(t_n, y_n) = \text{terms known from the past}, \tag{13.8.30}$$

where $\gamma = (1 + 1/2 + 1/3 + \ldots + 1/k)^{-1}$. It cannot be solved by the predictor-corrector scheme used in OLEG, unless $\|h_n \gamma f'(y)\| \ll 1$. This is an unacceptable restriction for stiff problems. Therefore *a Newton-like method* is used. The Jacobian $I - h_n \gamma \partial f/\partial y$ is computed and inverted only once every arc. Prediction is obtained by the extrapolation of the local k'th degree polynomial from the previous step. There is only one iteration at every step plus an extra iteration at the last step of every arc, in order to check that the scheme works satisfactorily. This program has inherited most of the post-processing facilities of OLEG.

The package is started by the MATLAB command `bdf`. *If you use a UNIX system*, note that you cannot use `ctrl-z` to leave the keyboard mode. Instead there is a command `cz` that calls an `mfile` that does it. The interrupt commands `k, b` etc. cannot be used, unless you can produce a `mexfile` by linking the C-program `getch.c` to this package. There may be a few more differences.

look at these for problems and computer exercises:

D20.1(0.2) (a) Look into the files `rmods.m` and `amods.m`, in order to find out what equations are treated in Problems 1,2,5,6 and 7, and the default settings of the parameters. Do you recognize the example treated above? Are there any changes?

You are also advised to look at the file `globals.m` in order to see the notations used. For example, the order of accuracy p is denoted by `pp`. What is meant by `yout, difout, tout, yshort, tshort`? (Don't hand in your answer to the last question.)

(0.8) (b) Run Prob.2, default case, with a diary. What is plotted? As post-processing, plot the relative step size h/t, (use the vector `tshort`) and make a table of the solution at $t = 10^n, n = -5, -4, \ldots$, until `tend`. (There is an m-file for such interpolations.) Then make an extra run, with a smaller tolerance, (or with

`stepctrl=2`) in order to estimate the accuracy.

Explain why the solutions for different values of b approach each others. Look into the outfile and compare the max value of y_1 and the behavior of the solution for large values of t with the theoretically results derived above. Describe quantitatively and approximately the variation of the step size, when t is large, and how the number of steps grows with t.

Review Questions

R1. (a) Determine $y_0(t), y_1(t)$, so that, under conditions to be stated, the function $\tilde{y}(t) = y_0(t) + \epsilon y_1(t)$ is at an $O(\epsilon^2)$-distance from a particular solution to the ODE system $\epsilon\dot{y} = F(y)$, $0 < \epsilon \ll 1$ that is attractive for other solutions that start in its neighborhood.

(b) For a single ODE of the form $\epsilon\dot{y} = F(y)$, $0 < \epsilon \ll 1$, show that, under appropriate conditions, the solution $y(t)$ is approximately an $O(\epsilon)$-delay of a solution of the reduced problem, i.e. the algebraic equation obtained for $\epsilon = 0$. Give an example, where the conditions are no longer valid in the neighborhood of some points, and tell what happens.

R2. Derive the non-homogeneous *linear* variational equation associated with a differential system $dy/dt = f(t, y; p),\ y(a) = c(p)$, where p is a vector of parameters.
What is the *nonlinear* variational equation? Exemplify how it is used.

R3. Consider a differential system $\dot{y} = f(y)$. Formulate a general theorem, that guarantees that a motion that starts in a domain V will remain there forever. Exemplify how the theorem is to be applied on a domain with singular points on the boundary, e.g., a closed circular cone. Formulate and exemplify the use of its corollaries, in the text called the Comparison theorem and the Positivity Theorem.

R4. (a) Consider the linear autonomous system $\dot{y} = Ay$. Formulate in terms of the spectrum of A necessary and sufficient conditions for the stability and asymptotic stability of the origin.

(b) Give the main features of an example that shows that a nonautonomous linear system can have unbounded solutions, even if, for every t, the eigenvalues of $A(t)$ are less than some negative constant.

R5. (a) Define the basic notions of the Lyapunov stability theory: stability, asymptotic stability, Lyapunov function. Formulate, in terms of Lyapunov functions, the basic theorems about the stability and asymptotic stability of a critical point.

(b) Exemplify the distinction between stability and boundedness for a nonlinear ODE.

(c) Show, with a reference to a theorem in the text (i.e. the corollary of Theorem **??**) that a critical point is asymptotically stable for a non-linear system,

if it is asymptotically stable for the linear approximation. Does the statement remain true, if you remove the words "asymptotically"? Give a proof or a counterexample.

R6. Give an example of a problem with a limit cycle. Draw a simple phase plane sketch. What is the difference between a limit cycle and a periodic solution of the type exemplified by the undamped pendulum and the predator-prey problem? Quote the Poincaré–Bendixson theorem.

Problems

P1. The different types of neighborhood to a critical point in $\mathbf{R^2}$ may be illustrated on single complex equations of the form $\dot{z} = \lambda z$ with $\lambda = -1,\ 0,\ 1,\ -1 + i,\ i,\ 1 + i$, *and* real systems of the form $\dot{y} = Ay$, with matrices of the form

$$\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \quad \begin{pmatrix} \lambda & 1 \\ 0 & \lambda, \end{pmatrix}$$

where one should distinguish between λ positive, negative or zero. (There are in all something like 16 types). Draw some sketches with or without a computer, and/or consult a modern text on ODEs, or Strang [36].
For the system $\dot{y} = Ay$ in $\mathbf{R^2}$, find out how the trace and the determinant of A provide almost complete information about the type of a critical point.

P2. (a) Determine the stability type (or instability) of the origin for $\dot{y} = ay^3$ and for the complex equation $\dot{z} = iz + a|z|^2 z,\ a \in \mathbf{R}$. Write the latter system as a real system in both Cartesian and polar coordinates. Use the latter for settling the stability question, and sketch the solution, for a positive and a negative value of a.

(b) Determine the stability or instability of the origin for the system $y_1' = y_2 - y_1^3,\quad y_2' = 0$. Is it true that this shows that the origin can be stable for a nonlinear system, even though it is unstable for the linear approximation?

(c) Consider the equation $\dot{y} = -y^2 + r(t)$, $y(0) = \epsilon$, $(\epsilon > 0)$. Find upper and lower bounds for $y(t)$, valid for all $t > 0$, if $0 \le r(t) < \epsilon$? Can you get a lower bound, if you only know that $|r(t)| < \epsilon$? (This is related to trouble that has been encountered at the numerical solution of stiff problems, when the system has similar properties for small values of $\|y\|$ as this scalar example.)

(d) Work out the details of Example **??** (the pendulum). Study also the (in)stability of the other critical points.

P3. (Gomory, Lefschetz). Consider the system $y_1' = y_1,\ y_2' = -y_2$. Show that the orbits are hyperbolas, which are, in polar coordinates, described by the equation $r^2 \sin 2\phi = \text{const.}]$ By the transformation by reciprocal radii, we obtain a curve family, $r^2 = \text{const.}] \sin 2\phi$. Sketch these curves. Show that they satisfy the system $y_1' = y_1^3 - 3y_1y_2^2,\ y_2' = 3y_1^2y_2 - y_2^3$. This exemplifies the *elliptic sector* type of critical point mentioned in Sec. **??**.

P4. Set $B = (I - A)^{-1}(I + A)$.

(a) For real inner product norms, show that if $\mu(A) < 0$ then $\|B\| < 1$.

Hint: First show that $y = Bx \;\Rightarrow\; y - x = A(y + x)$. Then show that $\|y\|^2 < \|x\|^2$, if $x \neq 0$.

(b) Show by an example that this is not generally true for the max-norm.

(c) Set $m = (1 + \mu(A))/(1 - \mu(A))$. For inner product norms, show that $\|B\| \leq m$, if $0 \leq \mu(A) < 1$, while this is not generally true if $\mu(A) < 0$.

P5 Let $(u, v) = u^H H v$, $|u\|^2 = (u, u)$, and let $\|\cdot\|$, $\mu(\cdot)$ be, respectively, the subordinate matrix norm and the subordinate logarithmic norm.

(a) Show that $\mu(A) < 0 \;\leftrightarrow\; A^H H + HA$ negative definite, and that $\mu(A) = \max\{\kappa : \det(A^H H + HA - 2\kappa H) = 0\}$.

Hint: Apply the results proved in Theorem 13.8.4. For the latter statement, either use the Lagrange multiplier rule, or reduce the problem to the l_2-case for the matrix RAR^{-1}, by the Cholesky decomposition $H = R^H R$ and the substitution $w = Ru$.

(b) Show that $\|A\|^2 = \max\{\kappa : \det(A^H HA - \kappa H) = 0\}$.

P6. Consider the predator-prey problem.

$$y_1' = ay_1 - by_1y_2, \quad y_2' = cy_1y_2 - dy_2, \quad (a, b, c, d \geq 0).$$

(a) Show that an orbit that starts in the first quadrant will remain there.

(b) Show that the origin is a saddle point, and that $p = (d/c, a/b)$ is a stable center for the linear approximation (around p), and determine the period of the solutions of the linear approximation.

(c) If you divide the two differential equations, you obtain a single equation, $dy_2/dy_1 = \ldots$. Show that its general solution is $F(y) = \text{const.}$, where $F(y) = a \ln y_2 - by_2 - cy_1 + d \ln y_1$. Show that $F(p) - F(y)$ is a Lyapunov function. For what values of k does the equation $F(y) = k$ represent closed orbits in the first quadrant? of the problem? How can you be sure that they are closed?

P7. Vacant

P8. Consider the matrix differential equation,

$$\dot{U} = J(t)U, \quad U(0) = I.$$

Note that $U(t + \epsilon) = \big(I + \epsilon J(t)\big)U(t) + o(\epsilon)$, and that [47]

$$\|I + \epsilon J\| = 1 + \epsilon\mu(J) + o(\epsilon), \quad d\|U(t)\|/dt \leq \mu(J(t))\|U(t)\|.$$

Similarly, show that $\det(I + \epsilon J) = 1 + \epsilon \operatorname{trace}(J) + o(\epsilon)$, and that

$$d(\det U(t))/dt = \operatorname{trace} J(t) \det U(t).$$

Note that if $\dot{y} = f(t, y)$, $J(t) = f_y'(t, y(t))$, then $\operatorname{trace} J(t) = \sum \partial f_i/\partial y_i = \operatorname{div} f$ at $y = y(t)$.

[47] $I + \epsilon J$ is called an infinitesimal perturbation of the unit matrix in the direction of the matrix J.

Show that these results hold for the matrix differential equation $\dot{U} = UJ(t)$ too. *Hint:* See Sec. 13.1, problem P10.

P9. There are many variations of Theorem **??**. The following can be useful in the study of certain types of stiff problems, when $g(t)$ is positive and sufficiently smooth, and $\mu^* \times \big($the local time-scale of $g(t)\big) \ll -1$. Assume that $d\|u\|/dt \leq \mu^*(\|u\| - g(t))$, $(\mu^* < 0,\ t \geq a)$.

(a) Show that

$$\|u(t)\| \leq \psi_k(t) + \max_{a \leq x \leq t} g^{(k)}(x)(\mu^*)^{-k} + Ce^{\mu^*(t-a)},$$

where

$$C = \|u(0) - \psi_{k+1}(a)\|, \quad \psi_0(t) = 0, \quad \psi_k(t) = \sum_{p=0}^{k-1} g^{(p)}(t)(\mu^*)^{-p}.$$

(cf. (13.8.8).)

Hint: Show this first for $k = 0$. Then derive a similar differential inequality for $\|u(t)\| - \psi_k(t)$, where $g(t)$ is replaced by something else.

(b) Since $\psi_k(t)$ is independent of a, it seems strange that the maximization mentioned in the result should be over the whole interval $[a, t]$. Try to improve this. Also show that the first neglected term is a good error estimate if $|g^{(k)}(t)/g^{(k+1)}(t)| \ll |1/\lambda|$.

* There are some question marks to P8 in the Ch. 9,12–14 book.

P10. Derive a bound for the difference between a solution $y(t)$ of the differential system,

$$\epsilon\dot{y} = f(t, y), \qquad (0 < \epsilon \ll 1), \tag{13.8.31}$$

and the solution $z(t)$ of the reduced problem, $f(t, z) = 0$, in terms of ϵ and upper bounds for $\|\partial f/\partial t\|$ and $\mu(\partial f/\partial y)$ in some suitably defined domain. The latter is called μ^* and is assumed to be negative. Show that this difference is $O(\epsilon)$ for $t > (\epsilon/|\mu^*|)\ln(1/\epsilon)$, i.e. "after a fast transient" or "outside a thin boundary layer".

Hint: By the chain rule, $\partial f/\partial z\ \dot{z}(t) + \partial f/\partial t = 0$. Rewrite the reduced problem as a differential system:
$\epsilon\dot{z}(t) = f(t, z) + \epsilon\dot{z}(t), \qquad (0 < \epsilon \ll 1)$.

Problems and Computer Exercises

C1. Treat Problem 13.1.8 (A population with crowding and toxins) with Runge's 2nd order method. Start with $y_1(0) \ll$ TOL, $y_2(0) = 0$.

Run the following cases long enough to show the limits as $t \to \infty$:
$a = b = 0.5$; $a = b = 0.1$; $a = 0.5,\ b = 0$; $a = 0.1,\ b = 0$.
Plot y_1 versus t on the same sheet for all the four cases.
Plot y_2 versus y_1 in the first two cases.

C2. Treat Problem 13.1.2 with Runge's 2nd order method. You are likely to encounter some trouble with values of $y > 1$, due to the inevitable computational errors. How can you rewrite the problem in order to avoid this trouble (without using the known exact solution)?

C3. Test experimentally the result of Problem 3 above, for some representative choices of λ and α.

C4. Solve by computer the equation $\dot{y} = t^2 - y^2$, $y(0) = 1$. Choose the tolerance so that you can rely on 4 decimal places. Plot the difference between $y(t)$ and the approximations mentioned above in Problem 4.

C5. Solve by computer the "rectifier problem", (13.8.14). In particular, determine the minimum value of $y(t)$. Choose TOL and various values of ϵ, so that you can test the hypothesis that $\min y(t) \sim C\sqrt{\epsilon}$. If it seems true, estimate C to about 1% relative accuracy. (Extrapolate appropriately to $\epsilon = 0$.)

Hint: Remember that the right hand side is zero at the minimum point.
Also prove theoretically that $y(t) > 0,\ \forall t > 0.1$.

C6. The Brusselator problem is a chemical system with a limit cycle. A simple version, see Hairer, Wanner and Nörsett [20, p. 112]: is described by the system,

$$y_1' = a + y_1^2 y_2 - (b+1)y_1, \quad y_2' = by_1 - y_1^2 y_2.$$

Choose $a = 1,\ b = 3$, and show that the critical point is unstable for this choice. Compute one orbit with the origin as starting point, and another orbit that starts near the critical point. You will find that both orbits approach the same limit cycle. Estimate its period to about 1% relative accuracy.

C7. Lorenz's example of a chaotic motion. See Fig. 13.7.5b. The following equations occurred in Theoretical Meteorology, see Hairer et al. [20, p. 117]:

$$y_1' = -\sigma y_1 + \sigma y_2, \quad y_2' = -y_1 y_3 + r y_1 - y_2, \quad y_3' = y_1 y_2 - b y_3,$$

with initial conditions: $y_1(0) = -8$, $y_2(0) = 8$, $y_3(0) = r - 1$.
Take e.g. $b = 8/3$, $\sigma = 10$, $r = 28$, and run the problem over the interval $0 \le t \le 30$, with TOL $= 0.001$ and 0.002. Plot y_1 versus t. You are likely to find big differences in the results.
Plot also the projections of the orbit into the coordinate planes, and try to obtain an idea of the behavior of the orbit in $\mathbf{R^3}$.
Finally plot the intersection of the orbit with the plane $y_3 = r - 1$, a so-called *Poincaré section*, a useful picture for the theoretical analysis of a non-linear system.

C8. The equation of an undamped pendulum $\ddot{\phi} + \frac{1}{\pi}\sin\pi\phi = 0$ can be rewritten as a system, $\dot{y}_1 = \frac{1}{\pi}y_2$, $\dot{y}_2 = -\sin(\pi y_1)$. The upper part of Fig. 13.8.7 is a phase plane plot with 14 orbits for this system.
Similarly, the equation of a damped pendulum $\ddot{\phi} + \frac{1}{\pi}\dot{\phi} + \frac{1}{\pi}\sin\phi = 0$ can be written in the form, $\dot{y}_1 = \frac{1}{\pi}y_2$, $\dot{y}_2 = -\frac{1}{\pi}y_2 - \sin(\pi y_1)$. The lower part of the same figure is a phase plane plot with initial values, $y_1(0) = 0$, $y_2(0) = 2\sqrt{a}$ for 15 values of a: $a = 1:1:10$ and $12:2:20$.

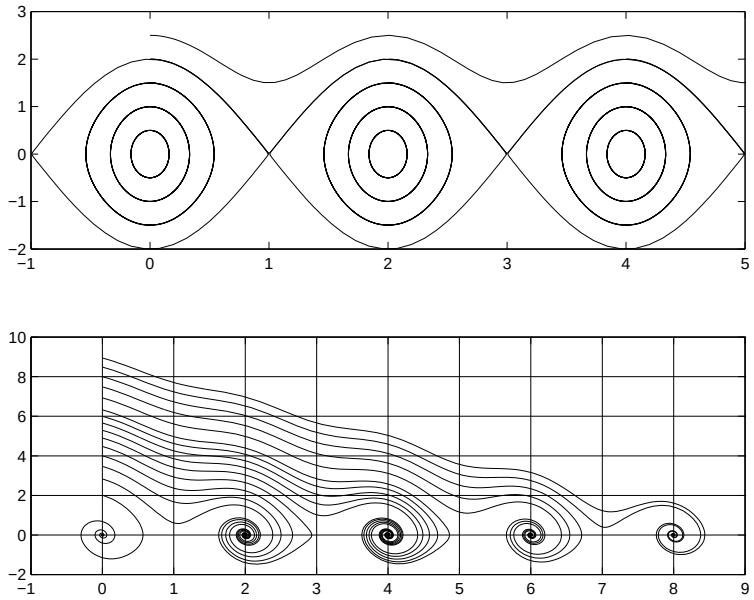

Figure 13.8.7. *Phase plane plots in the $y_1\,y_2$-plane for the undamped pendulum (upper picture) and the damped pendulum (lower picture) of Problem 8.*

(a) Run the undamped case with $y_1(0) = 0$, $y_2(0) = -0.05,\ 0.95,\ 1.95$. Determine the periods to about 1% accuracy, and compare with the results of "the theory of small oscillations". Explain theoretically and in physical terms the change of the character of the orbits that occurs at $y_2(0) = 2$.

(b) For the damped pendulum, determine by numerical experiment (to about 1% relative accuracy) the initial speeds needed at the bottom point in order that the pendulum should be able to do $1, 2, 3, 4, 5$ complete revolutions, before it starts to oscillate back and forth around the bottom point.

Hint: Run the system backwards in time from one (or more) appropriate starting point(s). Repeat the run with other tolerances and slightly different starting points in order to estimate the accuracy.

C9. The Oregonator...

C10. (a) Run the system

$$y_1' = (1 + ay_1)(y_2 - y_1); \quad y_2' = \tfrac{1}{2}y_1 - y_2; \quad y_3' = y_2;$$

with initial values $y_1(0) = 0$; $y_2(0) = 1$; $y_3(0) = 0$, for a few values of the parameter a. Choose the tolerance and the final value of t, so that $\lim_{t\to\infty} y_3(t)$ can be determined to three correct decimals. The result will probably surprise you.

Make a conjecture about $\lim y_3(t)$, and try to prove it. Try also to generalize the result to a more general system of ODEs.

(b) The following example may give a hint to the proof of the conjecture. The differential equation $\dot{y} = -e^y y$ is separable, but the expression for the exact solution is rather difficult to handle. Show instead by a rather simple manipulation of the differential equation that $\int_0^\infty y(t)dt = 1 - e^{-y(0)}$.

13.9 More about Logarithmic Norms, Difference Equations and Stability Criteria.

* An introduction is needed. Ref. to Sec. 3.1.

13.9.1 Difference Equations and Matrix Power Boundedness

Inledning! Gallra bort en del nedan

* The next lines should be changed with ref. to the end of Sec. 13.1.

A simple and useful upper bound of $\mu_2(A)$ is obtained by the combination of (13.8.4) with Theorem 13.1.25, statement B:

$$\mu_2(A) = \max \Re\lambda(B) \leq \mu_\infty(B), \tag{13.9.1}$$

where B is the Hermitean part of A.

For some classes of matrices, an efficient (or almost efficient) norm can be found more easily than by the construction used in the proof of Theorem 10.2.9. This may have other advantages as well, e.g. a better conditioned T. Consider a **weighted max-norm** $\|x\|_w = \max_i |x|_i/w_i = \|T^{-1}x\|_\infty$, where $T = \operatorname{diag}(w_i)$. Then

$$\mu_w(A) = \mu_\infty(T^{-1}AT) = \max_i \Re a_{ii} + \sum_{j,\ j\neq i} |a_{ij}|w_j/w_i. \tag{13.9.2}$$

Note that $\kappa(T) = \max w_i / \min w_i$.

Set $\tilde{A} = [\tilde{a}_{ij}]$, where $\tilde{a}_{ii} = \Re a_{ii}$, $\tilde{a}_{ij} = |a_{ij}|$ for $i \neq j$. Note that $\mu(A) = \mu(\tilde{A})$, when $\mu(\cdot)$ is subordinate to $\|\cdot\|_w$, i.e., a weighted max-norm or a weighted l_1-norm. Also note that the inequality $\mu_w(A) \leq \beta$ is equivalent to the inequalities $\tilde{A}w \leq \beta w,\ w > 0$.

If A is irreducible, a modified form, see Problem 14, of the *Perron–Frobenius Theorem* (Theorem 10.2.12) tells us that there exists a positive eigenvector w, such that $\tilde{A}w = \alpha(\tilde{A})w$, hence *the logarithmic norm $\mu_w(\cdot)$ is efficient for the matrix $\tilde{A}$*. It is in general not efficient for A itself, since $\alpha(A)$ may be less than $\alpha(\tilde{A})$, but still it can be useful also for A.

The latter result can be extended to some reducible matrices. e.g. to *any upper triangular $n \times n$ matrix A, such that $\Re a_{ii} < \Re a_{nn} = \alpha(A)$ for all $i < n$*. Then a positive vector w such that $\tilde{A}w \leq \alpha(A)w$ can be found by solving the inequalities $\Re(a_{nn} - a_{ii})w_i \geq \sum_{j,\ j>i} |a_{ij}|w_j$, for $i = n-1,\ n-2, \ldots, 1$. As in the analogous case discussed in Sec. 10.2.4 one may obtain a smaller value of $\kappa(T)$ by choosing w_i larger than necessary, for some i. (For example: the usual max-norm is efficient, if A is very strongly diagonally dominant and$\Re a_{ii} < \Re a_{nn}$.) [48]

[48] Interesting applications are to matrices where the elements are norms or logarithmic norms of submatrices of some matrix. See, e.g., (13.8.22).

In the application of these results to yield improved bounds for the solution of ODEs there is one more complication: if the Jacobian varies with time then the matrix T is also likely to do so. It is sufficient to study a pseudo-linear system, since a general non-linear system can be reduced to this case, as in Theorem 13.1.23.

Theorem 13.9.1.
Consider the pseudo-linear system

$$\frac{du}{dt} = J(t,u)u + r(t,u).$$

Let $T(t)$ be a smooth non-singular matrix-valued function. In addition to a given norm $\|\cdot\|$, we consider a time-dependent vector norm $\|y\|_T = \|T^{-1}y\|$. Assume that, for every $t \in [a,b]$, there exists a real-valued function $\mu^*(t)$, and a convex domain $D_t \subseteq \mathbf{R}^s$, such that

$$\mu_T(J(t,w)) + \mu(-T^{-1}T'(t)) + \mu(T^{-1}T'(t)) \le \mu^*(t), \quad \kappa(T(t))\|r(t,w)\| \le \epsilon(t),$$

$\forall w \in D_t$. Then $\|u(t)\| \le \psi(t)$, where $\psi(t)$ is a solution of the scalar differential equation,

$$\frac{d\psi}{dt} = \mu^*(t)\psi + \epsilon(t), \quad \psi(a) \ge \|u(a)\|,$$

*as long as the bounds obtained from this [49] guarantee that $u(t) \in D_t$. (See e.g. (**??**), if μ^* and ϵ are constant.)*

Proof. Set $u = Tz$. Then $T\dot{z} + T'z = JTz + r$, i.e.,

$$\dot{z} = (T^{-1}JT - T^{-1}T')z + T^{-1}r.$$

Hence, $\|z(t)\| \le \zeta(t)$, where

$$\zeta' = (\mu_T(J) + \mu(-T^{-1}T'))\zeta + \|T^{-1}r\|, \quad \zeta(a) \ge \|z(a)\|.$$

Now set $\eta = \|T\|\zeta$. Note that $\|u\| = \|Tz\| \le \|T\|\|z\| \le \|T\|\zeta = \eta$. Also note that $\|T\|' \le \mu(T^{-1}T')\|T\|$, since $dT/dt = T(T^{-1}T')$. Then

$$\eta' = \|T\|'\zeta + \|T\|\zeta' \le \mu(T^{-1}T')\|T\|\zeta + (\mu_T(J) + \mu(-T^{-1}T'))\|T\|\zeta + \kappa(T)\|r\|,$$

i.e.,

$$\eta' \le (\mu(T^{-1}T') + \mu_T(J) + \mu(-T^{-1}T'))\eta + \epsilon(t) \le \mu^*(t)\eta + \epsilon(t).$$

Hence $\|u(t)\| \le \eta(t) \le \psi(t)$, where $\psi(t)$ is defined above. The argument is valid as long as $\|u(t)\| \in D_t$. □

[49] See Theorem 3.1.10.

Example 13.9.2

The application of this theorem is particularly simple, when $T(t)$ is a diagonal matrix. Consider the differential system

$$\dot{u} = J(t)u, \quad \text{where} \quad J(t) = \begin{pmatrix} -\frac{1}{2}t^{-1} & 2t^{-3} \\ -\frac{1}{2}t & -\frac{1}{2}t^{-1} \end{pmatrix}.$$

Set $T = \operatorname{diag}(1, q)$, where $q = \frac{1}{2}t^2$. After straightforward computation,

$$T^{-1}JT = t^{-}1 \begin{pmatrix} -\frac{1}{2} & 1 \\ -1 & -\frac{1}{2} \end{pmatrix}, \qquad T^{-1}T' = t^{-1}\operatorname{diag}(0, 2).$$

Set $\|y\|_T = \|T^{-1}y\|_2$. Then $\mu_T(J) = mu_2(T^{-1}JT = \frac{1}{2}t^{-1}$, $\mu_T(T^{-1}T') = t^{-1}\max(0, 2) = 2t^{-1}$, $\mu_T(-T^{-1}T') = t^{-1}\max(0, -2) = 0$, hence

$$\mu^*(t) = t^{-1}(-\tfrac{1}{2} + 2 + 0) = \frac{3}{2}t^{-1}.$$

The differential equation $\dot{\psi} = \frac{3}{2}t^{-1}\psi$ has the solution $\psi(t) = ct^{3/2}$. The theorem thus gives a warning that $\|u(t)\|$ may be unbounded, in spite that the eigenvalues of of $T^{-1}JT$, and hence also of J, are $t^{-1}(-\frac{1}{2} \pm i)$.

This warning is correct, although the growth is overestimated. Actually, it is easily verified that the exact solution reads,

$$u(t) = c_1 \begin{pmatrix} t^{-3/2} \\ -\frac{1}{2}t^{1/2} \end{pmatrix} + c_2 \begin{pmatrix} 2t^{-3/2}\ln t \\ -t^{1/2}(1 - \ln t) \end{pmatrix}.$$

This expression also shows that the solution of a linear differential system with variable coefficients may not exhibit oscillatory behavior, even though it has complex conjugate eigenvalues.

Recall that we, in problem P10 of Section 3.1, saw a system that had the opposite type of behavior; the solutions were bounded, even though the logarithmic norm was positive for every $t > 0$. (The logarithmic norm converged fast enough to zero, as $t \to \infty$.) □

Notice the similarities and the differences of this theorem and Theorem 13.2.1 (where, in a way, $S(t)$ corresponds to $T(t)$). One difference is the presence of the term $\mu(T^{-1}T')$ in the condition for μ^*. This is due to the fact that the norm $\|\cdot\|_T$ is here only an internally used aid for the derivation of a sharp bound that is to be expressed in terms of the original (external) norm $\|\cdot\|$. In Theorem 13.2.1, however, the matrix S performs a transformation to a norm that is used also in the result. The other characteristic feature of Theorem 13.2.1, namely the transformation of the independent variable ("age" instead of "time"), can be used as a preprocessing also to an application of Theorem 13.9.1 or Theorem 13.1.23, whenever it is appropriate.

Recursion formulas of the type

$$y_{n+k} = f(y_n, y_{n+1}, \ldots, y_{n+k-1}, n), \tag{13.9.3}$$

play a large part in the numerical solution of differential equations. Its solution is uniquely determined when k initial values $y_0, \ldots, y_{k-1}$ are given, and these can be chosen arbitrarily—at least if the function f is defined in the whole space. Such recursions can also be written in the form of a kth order difference equation

$$\Delta^k y_n = g(y_n, \Delta y_n, \ldots, \Delta^{k-1} y_n, n). \tag{13.9.4}$$

The properties of such equations were discussed in Sec. 3.2.3.

We next consider nonhomogeneous **linear systems of first-order difference equations**, written in vector-matrix form:

$$y_{n+1} = A_n y_n + x_n,$$

where $y_n, x_n \in \mathbf{R}^s$ and $A_n \in \mathbf{R}^{s\times s}$. If the initial value y_0 is given, then by induction we obtain

$$y_n = P_{n,0} y_0 + \sum_{j=1}^{n} P_{n,j} x_{j-1}, \tag{13.9.5}$$

$$P_{n,j} = A_{n-1} A_{n-2} \ldots A_j, \qquad P_{n,n} = I. \tag{13.9.6}$$

This is a discrete analog of (13.1.20), (which may make the formula for the continuous case more intelligible).

If the matrices A_i are non-singular then we may write $P_{n,j} = P_n P_j^{-1}$. In particular, if all A_i are equal to A, then

$$y_n = A^n y_0 + \sum_{j=1}^{n} A^{n-j} x_{j-1}, \tag{13.9.7}$$

This formula holds, of course, also in the scalar case.

The following analog to Theorem **??** is easily proved by induction.

Theorem 13.9.3.

The solutions of a "pseudo-linear" system of difference equations,

$$u_{n+1} = A(n, u_n) u_n + r(n, u_n),$$

satisfy the inequality, $\|u_{n+1}\| \le \|A(n, u_n)\| \cdot \|u_n\| + \|r(n, u_n)\|$. *Let* D_n *be a convex domain in* $\mathbf{R}^s$, *and assume that*

$$\|A(n, w)\| \le a_n, \quad \|r(n, w)\| \le b_n, \quad \forall w \in D_n.$$

Then, $\|u_n\| \le \psi_n$, *where* ψ_n *is a solution of a scalar difference equation,* $\psi_{n+1} = a_n \psi_n + b_n$, $\psi_0 = \|u_0\|$, *as long as the bound derived from this guarantees that* $u_n \in D_n$.

*** Change to a reference to Lemma 13.2.1.*

If $a_n = a, b_n = b$, *independently of* n, *then*

$$\|u_n\| \le \psi_n = \begin{cases} a^n \|u_0\| + \dfrac{b(1-a^n)}{1-a}, & \text{if } a \ne 1; \\ \|u_0\| + bn, & \text{if } a = 1. \end{cases} \tag{13.9.8}$$

If $a < 1$, $\|u_n\| \le \max\{\|u_0\|, b/(1-a)\}$, $n \ge 0$.

Proof. The proof is left for Problem 3. □

The natural relation is via a step size parameter h, such that $nh = t$, $\quad a = e^{\mu^* h}$. In particular, $a = 1$ corresponds to $\mu^* = 0$.

Example 13.9.4

In the analysis of numerical methods for differential equations, see e.g., Sec. 13.4, one sometimes encounters a vector sequence that satisfies a difference equation of the form,

$$y_{n+1} = (A + hB_n)y_n + r_n, \qquad n = 0, 1, 2, \ldots,$$

where h is a step length. We want to estimate $\|y_n\|$.

Assume that, for some l_p-norm, $\|B_n\| \le K$, $\|r_n\| \le ch^{p+1}$, $\forall n$, and that A is a constant matrix with spectral radius equal to 1, such that the eigenvalues of unit modulus are non-defective. Then, by Theorem 10.2.9, there exists a norm $\|y\|_T = \|T^{-1}y\|$, such that $\|A\|_T = 1$, for the subordinate matrix norm. Then $\|B_n\|_T \le K_1$, where

$$K_1 = \kappa(T)K, \quad \kappa(T) = \|T\| \cdot \|T^{-1}\|,$$

and $\kappa(T)$ is the condition number. It follows that

$$\|y_{n+1}\|_T \le (1 + K_1 h)\|y_n\|_T + ch^{p+1}\|T^{-1}\|.$$

This is the situation treated by Theorem 13.9.3, with $a_n = a = 1 + K_1 h$, $\quad b_n = b = ch^{p+1}\|T^{-1}\|$. By (13.9.8),

$$\|y_n\|_T \le (1 + K_1 h)^n \|y_0\|_T + \frac{(1 + K_1 h)^n - 1}{K_1 h} ch^{p+1}\|T^{-1}\|$$

Since $\|y\| \le \|T\| \cdot \|y\|_T$, $\quad \|y\|_T \le \|T^{-1}\| \cdot \|y\|$, we can return to the original norm:

$$\|y_n\| \le \|T\| \left((1 + K_1 h)^n \|y_0\| \cdot \|T^{-1}\| + \frac{(1 + K_1 h)^n - 1}{K_1} ch^p \|T^{-1}\| \right)$$

Finally, we use the definition of condition number and the following relations,

$$1 + K_1 h \le e^{K_1 h}, \quad nh = t, \quad e^{K_1 t} - 1 \le K_1 t e^{K_1 t}, \quad \text{for } t \ge 0,$$

in order to the obtain the result that was referred to in Section 3.4.

$$\|y_n\| \le \kappa(T) e^{K_1 t}(\|y_0\| + tch^p). \tag{13.9.9}$$

If we had not used to the T-norm internally, the result is likely to have become less sharp, in many cases even useless. Still the positive constant K_1 is for many applications a weak point of this result. A stronger bound may be obtained by the use of a sequence of similarity transformations T_n according to the following general theorem that is a discrete analog of Theorem 13.9.1. For the sake of simplicity it is formulated for a linear system, but the generalization to a pseudo-linear system is straight-forward (see Theorems 13.9.1 and 13.9.3). Recall that $\|A\|_T = \|T^{-1}AT\|$.

Theorem 13.9.5.

Consider the linear system $y_{n+1} = A_n y_n + r_n$. Let T_n be a sequence of non-singular matrices. Then $\|y_n\| \le \psi_n$, where ψ_n is defined by the difference equation

$$\psi_{n+1} = \kappa(T_{n+1}^{-1} T_n)\|A_n\|_{T_n} \psi_n + \kappa(T_{n+1})\|r_n\|, \quad \psi_0 = \kappa(T_0)\|y_0\|.$$

If $\kappa(T_{n+1}^{-1} T_n)\|A_n\|_{T_n} \le a$, and $\kappa(T_{n+1})\|r_n\| \le b$, then the bounds given in (13.9.8) *are valid, and the behavior of ψ_n is illustrated by Fig. 13.1.3.*

Proof. The proof is left for Problem P3. □

The question whether a matrix sequence $\{A^n\}_0^\infty$ is bounded or not is often of interest, e.g. in the application of (13.9.7). We also saw in the previous example that it is interesting to know whether there is a norm such that $\|A\| \le 1$. The following theorem, that is a discrete analog of Theorem **??**, shows, among other things, that these two questions are equivalent.

Theorem 13.9.6 (Power Boundedness of a Single Matrix**).**

Let A be a given square matrix with spectral radius $\rho(A)$. The following statements are equivalent:

(i) *The sequence $\{A^n\}_0^\infty$ is bounded.*

(ii) *All eigenvalues of A are located inside or on the unit circle, and there are no defective eigenvalues on the unit circle.*

(iii) *There exists an operator norm such that $\|A\| \le 1$.*

Proof. We shall establish the equivalence by showing that (i) implies (ii), (ii) implies (iii), and (iii) implies (i). In order to show that (i) implies (ii), we first consider a Jordan box $J_m(\lambda) = \lambda I + N$, where $N^m = 0$.

$$\|J_m(\lambda)^n\|_\infty = \left\| \sum_{p=0}^{\min(n,m-1)} \binom{n}{p} N^p \lambda^{n-p} \right\| = \sum_{p=0}^{\min(n,m-1)} \binom{n}{p} |\lambda|^{n-p} \ge c n^{m-1} |\lambda|^n.$$

It follows that $\|A^n\|$ can be bounded only if, for every eigenvalue λ, $|\lambda| \le 1$, and $m = 1$ if $|\lambda| = 1$. This is, in other words, condition (ii).

Next, (ii) implies (iii), by Theorem 10.2.9. Finally, (iii) implies (i), since $\|A^n\| \le \|A\|^n \le 1$. □

Difference equations of order k can be written as a system of first-order difference equations. After the substitution $z_n := (y_n, y_{n+1}, \ldots, y_{n+k-1})^T$, the difference equation $y_{n+k} + a_1 y_{n+k-1} + \ldots + a_k y_n = 0$ can be written in the vector-matrix form

$$z_{n+1} = A z_n, \qquad A = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_k & -a_{k-1} & -a_{k-2} & \cdots & -a_1 \end{pmatrix}, \tag{13.9.10}$$

where the matrix A is a **companion matrix**. In the literature such matrices appear in different forms, depending, e.g., on the naming of the coordinates of z_n or the coefficients of the difference equation, see, e.g., Problem 10.1.7 With our convention, the difference equation and the matrix A have the same characteristic equation, i.e. the roots of the characteristic equation of the difference equation are the eigenvalues of its companion matrix. The eigenvector belonging to the eigenvalue λ reads $(1,\ \lambda,\ \lambda^2, \ldots,\ \lambda^{k-1})^T$. It is unique, hence *any multiple eigenvalue of a companion matrix A is defective*. The verification of these statements is left for Problem 1.

Theorem 13.9.7.

The following **root conditions** *are necessary and sufficient, for the boundedness of all solutions of the difference equation*

$$y_{n+k} + a_1 y_{n+k-1} + \ldots + a_k y_n = 0,$$

or equivalently the power boundedness of the companion matrix (13.9.10)*, as $n \to \infty$:*

(i) *All roots of the characteristic equation should be located inside or on the unit circle;*

(ii) *The roots on the unit circle should be simple.*

Proof. This follows from Theorem 13.9.6, since any multiple eigenvalue of a companion matrix is defective. □

In many areas, notably in the analysis of finite difference methods for partial differential equations, it is important to know, if there is a constant C, such that $\|A^n\| \leq C$ hold for an *infinite family of matrices*. It turns out that statement (ii) of Theorem 13.9.6 is not sufficient, and statement (iii) must also be modified, as shown by the following example and theorem.

Example 13.9.8

A matrix family is defined by $A(\delta) = \begin{pmatrix} 1-\delta & \delta^{1/2} \\ 0 & 1-\delta \end{pmatrix}$, $0 \leq \delta \leq 1$, where using (10.2.14)

$$A(\delta)^n = \begin{pmatrix} (1-\delta)^n & n(1-\delta)^{n-1}\delta^{1/2} \\ 0 & (1-\delta)^n \end{pmatrix}.$$

For each δ statement (ii) of Theorem 13.9.6 is satisfied, but that tells only that $\|A(\delta)^n\| \leq C(\delta)$. We see,

$$\|e^{tB}\| \leq C, \quad \forall B \in \mathcal{F}, \quad \forall t \geq 0.$$

however, that

$$\max_{\delta} \|A(\delta)^n\|_\infty \geq \|A(1/n)^n\|_\infty \geq (1+\sqrt{n})(1-1/n)^n \sim n^{1/2}e^{-1} \to \infty,\ n \to \infty.$$

(Since all norms in a finite-dimensional space are equivalent the same final conclusion holds in nay norm.) This is *not* due to the fact that $1-\delta$ is a defective eigenvalue of $A(\delta)$, for if we replace one of the diagonal elements of $A(\delta)$ by $(1-\frac{1}{2}\delta)$, then no member of the matrix family has a defective eigenvalue, but $\|A(\delta)^n\|$ would become even larger. (Note that $A(0) = I$.)

The following important theorem gives necessary and sufficient conditions for the *power boundedness for a family of matrices*, cf. Theorem 13.9.6.

Theorem 13.9.9 (The Kreiss Matrix Theorem: Discrete Case)**.**
Consider a matrix family $\mathcal{F} \subset \mathbf{R}^{\mathbf{s}\times\mathbf{s}}$ for a fixed s. The following four statements are equivalent.

(a) *There exists a constant C such that for all $A \in \mathcal{F}$ and all $n = 1,\ 2,\ 3, \ldots,$*

$$\|A^n\| \leq C.$$

(Note that the bound C must be the same for all $A \in \mathcal{F}$.)

(b) *There exists a constant C_1 such that for all $A \in \mathcal{F}$ and all $z \in \mathbf{C}$ with $|z| > 1$, the* **resolvent** *$(A - zI)^{-1}$ exists and*

$$\|(A - zI)^{-1}\| \leq \frac{C_1}{|z| - 1}, \quad \text{resolvent condition.}$$

(c) *There exist constants $C_2,\ C_3$, and to each $A \in \mathcal{F}$, a matrix S such that the condition number $\kappa(S) \leq C_2$ and $B = S^{-1}AS$ is upper triangular with*

$$|b_{ij}| \leq C_3 \min(1 - |b_{ii}|, 1 - |b_{jj}|), \quad \text{for } i \neq j.$$

(d) *There exists a constant C_5 and, for each $A \in \mathcal{F}$ a matrix T, such that*

$$\|A\|_T \leq 1, \quad \kappa(T) \leq C_5.$$

Proof. Sketch. The general plan is to show that (a) ⇒(b) ⇒(c) ⇒(d) ⇒(a). We first note that for each $A \in \mathcal{F}$ the eigenvalues are located in the closed unit disc, hence for $|z| > 1$, $A - zI$ is invertible and

$$\|(A - zI)^{-1}\| = \|\sum_{n=0}^{\infty} A^n z^{-n-1}\| \leq C \sum_{n=0}^{\infty} |z|^{-n-1} = \frac{C}{|z| - 1}.$$

Hence (a) ⇒ (b). The proof that (b) ⇒(c) is the most difficult part, and we refer to Richtmyer and Morton [29]. The proof that (c) ⇒(d) is a more complicated variant of the proof of Theorem 10.2.9, and again we refer to Richtmyer and Morton [29] for details. Note the important assumption about $\kappa(T)$. (This is automatically satisfied in the single matrix case.) Finally, (d) ⇒(a), since

$$\|A^n\| \leq \kappa(T)\|A^n\|_T \leq C_5\|A\|_T^n = C_5.$$

□

In recent applications of this theorem to the study of difference methods to partial differential equations, the resolvent condition (b) is the easiest one to verify. It is therefore interesting that LeVeque and Trefethen has found a direct proof that (b)⇒(a), with $C = 2esC_1$ based on the representation of A^n by a Cauchy integral. (Note that the resolvent appears in this integral.) Their proof is reproduced in Hairer and Wanner [21, p. 349].

There is a continuous version of this theorem that can be heuristically derived from the above discrete version as follows. (Historically, the continuous version was found first.) Set $A = e^{hB}$, $nh = t$. Then $A^n = e^{tB}$, $\|A^{\|} \approx e^{h\,\mu(B)}$, and the outside of the unit disk in the discrete case corresponds to the positive half-plane, $\Re z > 0$ in the continuous case.

Theorem 13.9.10 (*The Matrix Theorem: Continuous Case***).**

Consider a matrix family $\mathcal{F} \subset \mathbf{R}^{\mathbf{s}\times\mathbf{s}}$ for a fixed s. The following four statements are equivalent.

(a) *There exists a constant C such that*

$$\|e^{tB}\| \le C, \quad \forall B \in \mathcal{F}, \quad \forall t \ge 0.$$

(b) *There exists a constant C_1 such that for all $B \in \mathcal{F}$ and all $z \in \mathbf{C}$ with $\Re z > 0$, the resolvent $(B - zI)^{-1}$ exists and*

$$\|(B - zI)^{-1}\| \le \frac{C_1}{\Re z}, \quad \text{resolvent condition.}$$

(c) *There exist constants C_2, C_3, and to each $B \in \mathcal{F}$, a matrix S such that the condition number $\kappa(S) \le C_2$ and $K = S^{-1}BS$ is upper triangular with*

$$|k_{ij}| + C_3 \max(\Re k_{ii},\ \Re k_{jj}) \le 0, \quad \text{for } i \ne j.$$

(d) *There exists a constant C_5 and, for each $B \in \mathcal{F}$ a matrix T such that*

$$\mu_T(B) \le 0, \quad \kappa(T) \le C_5.$$

13.9.2 More Results on Stability Theory

There are other algebraic criteria and algorithms than given in Sec. 13.2.3 for the investigation of the root condition of a polynomial, perhaps most useful at the study the the root condition and related matters, for algebraic equations containing parameters. They can be used either numerically (floating point operations on vectors of polynomial coefficients) or algebraically (symbolically). There is a short discussion of these two approaches to problems with two parameters, in Example 13.9.20.

We first consider an analogous problem connected with the left half-plane This problem has an interest in its own right for differential equations, but it means more to us that the two problems can be transformed into each other by a Möbius transformation,

$$z = \frac{\zeta+1}{\zeta-1}, \quad \zeta = \frac{z+1}{z-1}, \quad |\zeta| < 1 \leftrightarrow \Re z < 0, \quad |\zeta| > 1 \leftrightarrow \Re z > 0, \tag{13.9.11}$$

and we set

$$R(z) = \left(\frac{z-1}{2}\right)^k \rho\left(\frac{z+1}{z-1}\right), \Rightarrow \ \rho(\zeta) = (\zeta-1)^k R\left(\frac{\zeta+1}{\zeta-1}\right), \tag{13.9.12}$$

and similarly for $\sigma(\zeta)$, $S(z)$. The polynomials R and S have no common factors (like ρ and σ). In the theory of multistep methods these formulas are called the **Greco-Roman transformation**. It can be seen as a linear coordinate transformation in the linear space of multistep methods (ρ, σ) that provides many simplifications, see Problem 18 and Sec. 13.4.

We need some notions and notations. $\mathbf{C}^+ = \{q : \Re q > 0\}$. $\bar{\mathbf{C}}^+$ is the closure of $\mathbf{C}^+$. It includes also the imaginary axis and ∞. Similarly, $\mathbf{C}^- = \{q : \Re q < 0\}$. The degree p of a polynomial is denoted $\deg p$.

Definition 13.9.11.
A k-th degree polynomial p, with zeros α_j, $j = 1, 2, \ldots, k$ is called a **Hurwitz polynomial**, *if $\alpha_j \in \mathbf{C}^-$, $\forall j$. The closure of the set of Hurwitz polynomials are the polynomials for which $\alpha_j \in \bar{\mathbf{C}}^-$, $\forall j$.* [50]

The following simple necessary condition is often helpful for the proof that a given polynomial is *not* a Hurwitz polynomial.

Theorem 13.9.12.
A necessary condition for a real polynomial to be a Hurwitz polynomial is that all coefficients are different from zero and have the same sign. This condition is sufficient for linear and quadratic polynomials only. For the closure of Hurwitz polynomials, a coefficient is also allowed to be zero.

Proof. In the factorization $c\prod(z-\alpha_j)$, a root α_j is either negative, or it occurs together with its complex conjugate in a factor of the form $(z-\alpha)(z-\bar{\alpha}) = z^2+az+b$ where $a > 0$, $b > 0$. The expansion of the product will therefore have positive coefficients only. The statement about sufficiency will appear as a consequence of the necessary and sufficient conditions presented in Theorem 13.9.13 or, more explicitly, in Problem 9 (a). The last statement follows from continuity considerations. □

A rational function f is called a **positive function**, if $\Re z > 0 \to \Re f(z) > 0$. In other words: it maps $\mathbf{C}^+$ into itself. A useful property follows directly: The

[50] Note hat the closure of Hurwitz polynomials also includes polynomials with multiple zeros on the imaginary axis. These do not satisfy the analog of the root condition.

function $f \circ g$, *defined by the equation* $f \circ g(z) = f(g(z))$, *is positive if* f *and* g *are so.* Notice that $1/z$ is a positive function, since $\Re\ 1/z = \Re\bar{z}/|z|^2 = \Re z/|z|^2$. Hence *if the function* f *is positive then* $1/f$ *is also positive.* In particular, the rational function S/R is positive if R/S is positive.

When the attribute **real** is added to terms like polynomial, rational function, positive function etc., it means that it is real on the real axis. In other words, it has real coefficients, when expanded about a point in $\mathbf{R}$.

Since $\rho(\zeta)/\sigma(\zeta) = R(z)/S(z)$, it follows from Theorem 13.2.19 that the function $q = R(z)/S(z)$ *maps* $\mathbf{C}^+$ *onto the interior of the complement of* S. In particular, *a linear multistep method is A-stable, iff* $R(z)/S(z)$ *is positive real.* This leads to a connection between Hurwitz polynomials and positive functions.

If R/S *is a positive rational function, then the polynomial* $R - qS$ *is a Hurwitz polynomial for any constant* $q \in \mathbf{C}^-$. An even more important connection is expressed by the following theorem.

Theorem 13.9.13.

Let $p_0(z) = c_0 z^k + c_2 z^{k-2} + c_4 z^{k-4} \cdots, \quad p_1(z) = c_1 z^{k-1} + c_3 z^{k-3} + c_5 z^{k-5} \cdots,$ *be two real polynomials. Set* $p(z) = p_0(z) + p_1(z)$, *and assume that* p_0 *and* p_1 *have no common divisor. Then* p *is a Hurwitz polynomial, if and only if*

$$p_0/p_1 \text{ is a positive function.} \tag{13.9.13}$$

Proof. We first note, that the statement in (13.9.13) is, by the second and fourth statements of (13.9.11), equivalent to the condition $|p_0(z)/p_1(z)+1|/|p_0(z)/p_1(z)-1| > 1$, i.e. $|p_0(z) + p_1(z)|/|p_0(z) - p_1(z)| > 1$,, i.e.

$$|p(z)| > |p(-z)|, \forall z \in \mathbf{C}^+. \tag{13.9.14}$$

This strict inequality immediately shows that $p(z) \neq 0$ for $z \in \mathbf{C}^+$. Moreover, when z is pure imaginary, the real part is equal to either $p_0(z)$ or $p_1(z)$ (depending on the parity of k) and the other is the imaginary part. Anyway, a pure imaginary zero of $p(z)$ would be a common zero of $p_0(z)$ and $p_1(z)$, and this contradicts the assumption. Hence the "if" part is proved.

The converse is more interesting. Assume that $p(z)$ is a Hurwitz polynomial. A root α_j is either negative, or it occurs together with its conjugate, both in $\mathbf{C}^-$. We have $p(z) = c_0 \prod (z - \alpha_j)$ and shall establish relation (13.9.14). Then $|p(z)/p(-z)|$ is a product of factors *either* of the form,

$$|z - \alpha|/|-z - \alpha| = |z - \alpha|/|z + \alpha| > 1, \quad \forall z \in \mathbf{C}^+,$$

or of the form

$$\left|\frac{(z-\alpha)(z-\bar{\alpha})}{(-z-\alpha)(-z-\bar{\alpha})}\right| = \left|\frac{z-\alpha}{z+\bar{\alpha}}\right| \left|\frac{z-\bar{\alpha}}{z+\alpha}\right| > 1, \ \forall z \in \mathbf{C}^+.$$

(Draw a picture with five points: $z \in \mathbf{C}^+$, $\alpha \in \mathbf{C}^-$, $\bar{\alpha}$, $-\alpha$, $-\bar{\alpha}$.) This proves (13.9.14) and hence the 'only if' part. □

The division algorithm, described at the end of Sec. 3.4.1 (Continued fractions), applied to the polynomials $p_0(z)$, $p_1(z)$ defined above, yields a sequence of alternatingly odd or even polynomials,

$$\frac{p_m(z)}{p_{m+1}(z)} = d_m z + \frac{p_{m+2}(z)}{p_{m+1}(z)}, \quad m = 0, 1, 2, \ldots, n-1, \quad \deg p_{m+1} = \deg p_m - 1.$$

Since the algorithm is, in fact, identical to the Euclid algorithm for finding common divisors of the polynomials p_0 and p_1, it will, if the assumption of the theorem is satisfied, terminate when p_{m+1} has become a non-zero constant. If $c_0 > 0$, a repeated use of Theorem 13.9.14 (below) tells that p_0/p_1 *is a positive function, hence* $p = p_0 + p_1$ *is a Hurwitz polynomial, if and only if* $d_m > 0$ *for all* m, the **Routh criterion**. This algorithm, the **Routh algorithm**, is easily programmed, with vectors for the polynomial coefficients, see Problem 17. If $d_m = 0$, within a tolerance, then p_m is the greatest common divisor of $p_0\, p_1$. This should be examined further. It contains the zeros of p on the imaginary axis, (these are common zeros of p_0 and p_1), but it may also contain real zeros, symmetrically located with respect to the imaginary axis.

use of *positive* scaling factors

The Routh criterion was developed by Routh in 1877, with a completely different derivation. An extension to polynomials with complex coefficients was made in 1946 by E. Frank. See Problem 16 for a different approach, due to Hurwitz 1895.

Theorem 13.9.14.

Let $f(z) = az + g(z)$, *where* $g(z)$ *is regular in* $\mathbf{C}^+$ *and bounded at* ∞. *Then* $f(z)$ *is a real positive function, if and only if* $a > 0$, *and* $g(z)$ *is either a positive function or identically zero.*

Proof. The "if" part is obvious; we shall prove the deeper "only if" part. First note that, since $g(z)$ is bounded, $\arg f(z) = \arg a + \arg z + o(1)$, as $z \to \infty$. We then note that $\Re w > 0 \;\leftrightarrow\; |\arg w| < \pi/2$. It follows that $|\arg f(z)| < \pi/2$, for all z, such that $|\arg z| < \pi/2$, if and only if $\arg a = 0$, i.e., iff $a > 0$. It then follows that $\liminf_{z\to iy} \Re g(z) = \liminf_{z\to iy} \Re f(z) - 0 \geq 0$ for $z \in \mathbf{C}^+$ and for all real y, (including $y = \pm\infty$). Then, by the minimum principle for harmonic functions; see Dinghas [13, p. 303], $\Re g(z) > 0$ in $\mathbf{C}^+$, unless $g(z) = 0$ everywhere. □

Theorem 13.9.15.

A rational function $f = u/v$ *can be a positive function only if* $|\deg u - \deg v| \leq 1$. *If* $|\deg u - \deg v| = 1$, *the leading coefficients of* u *and* v *have a positive ratio.*

Proof. Suppose that $f(z) \sim az^\alpha$, $z \to \infty$. Analogously to the previous proof we have $\arg f(z) = \arg a + \alpha \arg z$. If $|\alpha| > 1$ the right hand side covers an open sector of width greater than π and a positive function cannot do this. The rest is proved like the beginning of the previous proof. □

Results of a similar type can be obtained by a substitution like $z := 1/z$ or by applying the ideas of the proof. (See Problems.)

The Routh algorithm provided a clear criterion for the positiveness of an odd rational function. For more general rational functions the following result has for a long time been well known in circuit theory, where positive functions play an important role in the study of passive circuits.

Theorem 13.9.16.

A rational function $R(z)/S(z)$, where R, S have no common divisor, and where the leading terms of R and S are different from zero, is a positive function, iff

(i) $\Re\big(R(iy)/S(iy)\big) \geq 0$, *for all real y, such that $S(iy) \neq 0$;*

(ii) $R + S$ *is a Hurwitz polynomial.*

Proof. Think of a boundary locus construction $q = R(z)/S(z)$ with z traversing the imaginary axis from $-i\infty$ to $i\infty$ (instead of ζ traversing the unit circle). Condition (i) tells that $\mathbf{C}^-$ is to the left of the boundary locus, and hence the number of unstable roots of the equation $R(z) - qS(z) = 0$ is the same for all $q \in \mathbf{C}^-$. Condition (ii) tells that this number is 0, by a test at the point $q = -1$. □

For a real rational function Condition (i) can also written $\Re R(iy)S(-iy) \geq 0$. We can also formulate this

$$P(y^2) \equiv \Re R(iy)S(-iy) \geq 0, \tag{13.9.15}$$

where P is a polynomial of kth degree at most. It often simplifies the matters to use also the necessary condition that *both R and S belong to the closure of Hurwitz polynomial*, (although it should be possible to derive these from the others).

Example 13.9.17

Consider the real function $f(z) = R(z)/S(z)$, where

$$R(z) = az + b,\ S(z) = cz + d,\ |c| + |d| > 0,\ ad - bc \neq 0.$$

R and S are in the closure of Hurwitz polynomials, hence all non-zero coefficients have the same sign, and the other conditions imply that there is at least one non-zero coefficient in both the denominator and the numerator. Condition (13.9.15) becomes $P(y^2) = \Re(aiy+b)(-ciy+d) = acy^2+bd$, and this adds no new constraints, nor does Condition (ii).

A necessary and sufficient condition for the positiveness of a function $(az + b)/(cz+d)$, that is not identically a constant or $0/0$, is that the non-zero coefficients have the same sign.

Next we shall find the positivity condition for the real function,

$$\frac{R(z)}{S(z)} \equiv \frac{z + a_0}{b_2z^2 + b_1z + b_0}, \quad b_2 \neq 0,$$

where r, s have no common divisor. All non-zero coefficients must have the same sign. Since $a_1 = 1$, they must be non-negative and $b_2 > 0$. Condition (i) then tells that either $b_0 > 0$ or $a_0 > 0$ The function R/S is positive at the same time as S/R. We consider S/R in order to apply Theorem 13.9.14. We have

$$S(z)/R(z) = b_2 + (b_1 - b_2a_0)z + b_0)/(z + a_0).$$

By Theorem 13.9.14, and the first part of this example, we obtain the following necessary and sufficient conditions:

$$b_2 > 0, \quad b_0 + a_0 > 0, \quad b_1 - b_2a_0 \geq 0,$$

(in addition to the condition $s(-a_0) \neq 0$ for no common divisor.)

Finally we consider

$$\frac{R(z)}{S(z)} \equiv \frac{z^2 + a_1z + a_0}{b_2z^2 + b_1z + b_0},$$

with no common divisors. We have the non-negativity conditions for all coefficients, and Condition (i) adds the strict inequalities $b_0 + a_0 > 0,\ b_1 + a_1 > 0$. Condition (13.9.15) becomes after a short calculation, if we set $y^2 = t,\ a = a_1b_1 - a_0 - b_0,\ b = a_0b_0$,

$$P(t) = t^2 + at + b \geq 0, \quad \forall t \geq 0. \tag{13.9.16}$$

Note that $\min P(t) = P(0) = b$, if $a \geq 0$, or $b - \frac{1}{4}a^2$ if $a < 0$. Since we have the condition $b = a_0b_0 \geq 0$ already, we finally have, *in addition to the previous conditions, the new condition* $b \geq a^2/4$ *if* $a < 0$.

For rational functions of higher degree, a classical device, named **Sturm chains**, can be applied to check the validity of the inequality (13.9.15), after it has been rephrased as proving that

$$P(0) \geq 0, \quad p(t) \equiv P(t) + \epsilon \neq 0, \quad t \in (0, \infty), \tag{13.9.17}$$

where ϵ is to be chosen a little larger than what is needed to compensate for the errors of the elements of the two sequences, due to rounding.

Theorem 13.9.18. Sturm chains.

Let p be a polynomial with derivative p'. Set $p_0 = p$, $p_1 = p'$, and consider the Euclid algorithm,

$$p_{i-1} = q_ip_i - p_{i+1}, \quad i = 1, 2, \ldots, m.$$

Here q_i is the quotient at the division p_{i-1}/p_i, hence $\deg p_{i+1} < \deg p_i$. The algorithm terminates when $p_{m+1} = 0$; p_m is the greatest common divisor of the polynomials $p,\ p'$. The sequence of polynomials $p_0, p_1, \ldots, p_m$ is called a **Sturm chain**.

If $p(b) \neq 0$, $p(c) \neq 0$, then the number of distinct roots of the equation $p(t) = 0$ in the open real interval (b, c) equals the difference between the number of sign changes in the sequence $p_0(b), p_1(b), \ldots, p_m(b)$ and in the sequence $p_0(c), p_1(c), \ldots, p_m(c)$. A multiple root is counted as one root.

For $c = \infty$ the leading coefficient of the polynomial p_i is to be substituted for $p_i(c)$. Similarly for $b = -\infty$, with appropriate choice of signs.

Comment: A zero in these sequences can be ignored at the counting of sign changes. If, e.g., $p_i(b) = 0$, $i > 0$, the recurrence relation shows that there is exactly one sign change in the subsequence $p_{i-1}(b)$, $p_i(b)$, $p_{i+1}(b)$, both for $p(b) = \epsilon$ and $p(b) = -\epsilon$, unless two consecutive numbers in the sequence are zero, but in that case the whole sequence would be zero, and this is this is inconsistent with the assumption.

Proof. For a complete proof, see, e.g., Gantmacher [17, Vol. 2, p. 175], □

Example 13.9.19

The following is a classical application of Sturm chains for the separation of the real roots of an algebraic equation. Given $p(t) = 5t^5 - 3t^3 - 2t + 1$. The Sturm chain becomes (rounded to 4 decimals):

$$\begin{aligned} p_1(t) &= p'(t) = 25t^4 - 9t^2 - 2 \\ p_2(t) &= 1.2t^3 + 1.6t - 1 \\ p_3(t) &= 42.3333t^2 - 20.8333t + 2 \\ p_4(t) &= -1.8339t + 1.0279 \\ p_5(t) &= -3.6221. \end{aligned}$$

The number of sign changes are

$$\begin{array}{rccccc} & 4 & 3 & 2 & 1 & 1 \\ \text{for } t = & -\infty & 0 & 0.5 & 1 & \infty \end{array}.$$

After counting the number of "lost sign changes" we conclude that $p(t)$ has 1 negative and 2 positive zeros. □

Example 13.9.20

Consider the situation which occurred in Example 13.9.17, i.e., we want to find conditions for $p(t) = t^2 + at + b$ to be positive for $t \geq 0$. A necessary condition is $b \geq 0$. In an algebraic treatment, the Sturm chain becomes $p(t)$, $p_1(t) = 2t + a$, $p_2(t) = a^2/4 - b$. At $t = 0$ the sign sequence is $(+,\ a,\ a^2/4 - b)$, and at $t = \infty$, it is $(+,\ +,\ a^2/4 - b)$.

If $a^2/4 - b < 0$, there is 1 sign change at both places, for all a.

If $a^2/4 - b > 0$, there is no sign change at $t = \infty$, and so is the case at $t = 0$ if $a > 0$, but if $t = 0$, $a < 0$, there are two sign changes.

We reach the same conclusion as in Example 13.9.17: $p(t) > 0$ for $t \geq 0$ iff $b > 0$, and $a > 0$ or $b > a^2/4$□.

In a problem with (say) two parameters, it is not hard to work entirely with a numerical version of the Sturm algorithm, instead of a symbolic version, if the purpose is to obtain a plot of the the set $\{(a,b) : p(t;a,b) > 0 \ \ \forall t > 0.\}$. It is not hard to find suitable stepsize control, interpolation techniques, etc., for a program that you can be happy with yourself in an interactive environment. You can even allow yourself to use one or two big values of t, (though not so big that you risk overflow) instead of having a special branch in your code for handling ∞. It is less trivial to design software that works well under any conditions (good step size control, good criteria for deciding whether a quantity of small absolute value is to be counted as positive or not, signals for ill-conditioned cases, etc.). It seems likely that the complexity grows more slowly with $\deg p$ with this approach, than with symbolic computation with automatic handling of the inequality logic. (We are still discussing a two parameter problem.) This particular example seems easy both ways. This discussion is applicable to the other algorithms in this section.

There is also a criterion due to Schur 1916, that directly tests whether or not all zeros of a polynomial are located inside the unit circle, without the Greco-Roman transformation.

Theorem 13.9.21 (The Schur criterion)**.**
Let $P(\zeta) = c_0\zeta^n + c_1\zeta^{n-1} + \ldots + c_n$, and set $\hat{P}(\zeta) = \bar{c}_n\zeta^n + \bar{c}_1\zeta^{n-1} + \ldots + \bar{c}_n$. We write $P \in$ Sch, iff all zeros of P are strictly inside the unit circle. Let $p_0(\zeta)$ be a kth degree polynomial, and define recursively,

$$p_{i+1}(\zeta) = \frac{\hat{p}_i(0)p_i(\zeta) - p_i(0)\hat{p}_i(\zeta)}{\zeta}.$$

The algorithm terminates when $|p_i(0)| \geq |\hat{p}_i(0)|$.

The following statements are the basis of the algorithm:

(i) $(p_i \in \text{Sch}) \leftrightarrow \big(p_{i+1} \in \text{Sch}) \wedge (|p_i(0)| < |\hat{p}_i(0)|)\big)$,

(ii) $p_0 \in$ Sch*, iff p_i is equal to a constant at the termination.*

Proof. Sketch. If $|\hat{p}_i(0)| > |p_i(0)|$, then $\deg p_{i+1} = \deg p_i - 1$. The statement (i) is proved by the Rouché theorem of complex analysis, that tells that $\zeta p_{i+1}(\zeta)$ and $p_i(\zeta)$ have the same number of zeros inside the unit circle, if $|-p_i(0)\hat{p}_i(\zeta)| < |\hat{p}_i(0)p_i(\zeta)|$ on the unit circle. Since $|\hat{p}_i(\zeta)| = |p_i(\zeta)|$ on the unit circle, the latter condition is equivalent to the inequality $|\hat{p}_i(0)| > |p_i(0)|$.

If $P(\alpha) = 0$ then $\hat{P}(1/\bar{\alpha}) = 0$, and vice versa. If $\deg P > 0$, the modulus of the product of the zeros of P equals $|P(0)/\hat{P}(0)|$. Hence p_i cannot belong to Sch, if the termination criterion $|p_i(0)| \geq |\hat{p}_i(0)|$ is satisfied, before p_i has become a constant. □

13.9.3 Order Stars and Comparison Theorems

To be written.

Problems and Computer Exercises

1. From the classical Perron–Frobenius theorem (Theorem 10.2.12) we have the following: If P is a matrix with *strictly positive* elements, then P has a positive eigenvalue r, that we call the *Perron value*, with the following properties:

(i) r is a simple root of the characteristic equation.

(ii) r has a strictly positive eigenvector w, that we call the *Perron vector*.

(iii) If λ is any other eigenvalue of P, then $|\lambda| < r$.

The first two properties hold also for a matrix A that is *irreducible* and has *non-negative* elements, but the inequality of Property (iii) becomes $|\lambda| \leq r$.

(a) Find a 2×2 irreducible matrix with non-negative elements that does not possess Property (iii), unmodified. Also find a 2×2 matrix with non-negative elements that does not possess Properties (i) and (ii).

(b) Derive from these results a modified form concerned with $\max \Re\lambda$ instead of $\max |\lambda|$, for any irreducible matrix B, such that $b_{ij} \geq 0$, when $i \neq j$.

Hint: Apply the Perron–Frobenius theorems to a matrix $A = I + \epsilon B$ with positive elements, and let $\epsilon \to 0$.

(c) Let A and B satisfy the respective conditions stated above. Take the coordinates of the Perron vector of A as weights in a weighted maximum norm. Prove that the subordinate matrix norm is efficient for the matrix A. Then formulate and prove the analogous result for the subordinate logarithmic norm of the matrix B.

(d) Let $A = [a_{ij}]$ be irreducible, and let $C = [c_{ij}]$ be a complex matrix, such that $|c_{ij}| \leq a_{ij}$, then $\rho(C) \leq \rho(A)$, where $\rho(\cdot)$ denotes the spectral radius. Formulate and show the analogous result for the matrix B, (where the non-negativity is required for the off-diagonal elements only).

Hint: Use the norms discussed in (c).

(e) Find a weighted max-norm that produces an efficient logarithmic norm for the matrix

$$\begin{pmatrix} -3 & 4 & 6 \\ 0 & -2 & 5 \\ 0 & 0 & -1 \end{pmatrix}.$$

Comment: This matrix is *reducible*, but it can be handled according to the remarks after Theorem **??**.

2. (a) Set

$$A_{2m} = \begin{pmatrix} 0.5 & 0 \\ 1.5 & 0.1 \end{pmatrix}, \quad A_{2m+1} = A_{2m}^T, \quad m = 0,\ 1.\ 2, \ldots.$$

Show that the spectral radius of $A_{2m+1}A_{2m}$ is greater than unity, and that the recurrence relation $y_{n+1} = A_n y_n$ has solutions that are unbounded as $n \to \infty$, in spite that the spectral radius $\rho(A_n) = 0.5,\ \forall n$.

(b) Set $B(t) = \ln A_n$, $n \leq t < n+1$, $n = 0,\ 1,\ 2, \ldots$. Show that the eigenvalues

of $B(t)$ are strictly less than -0.6 for all $t \geq 0$, and that nevertheless the system $\dot{y} = B(t)y$ has solutions that are unbounded as $t \to \infty$.

(c) $B(t)$ is in this example discontinuous. Try to modify the example so that $B(t)$ obtains a continuous derivative everywhere.

(d) Let $A(t) = A + B(t)$, where all eigenvalues of A have negative real parts, and $\int_0^\infty \|B(t)\|dt$ is finite. Show that all solutions of the system $\dot{y} = A(t)y$ are bounded. Try also to relax the conditions on A and $B(t)$.

Hint: Use the corollary of Theorem **??**.

3. (a) Verify the statements made above concerning the companion matrix defined by (13.9.10), its characteristic equation, eigenvalues, eigenvectors etc.

Hint: An easy way to find the characteristic equation and the eigenvector is to solve the equation $Az = \lambda z$ from the top to the bottom. The last equation gives the condition that λ must satisfy!

(b) Similarly, find the inverse of the companion matrix A is by solving the equation $Ax = y$ from the top to the bottom. This gives you $x = A^{-1}y$.

(c) Let λ be a double eigenvalue of a companion matrix. Show that a principal vector is $(0,\ 1,\ 2\lambda, \ldots,\ (k-1)\lambda^{k-2})^T$. Formulate a generalization to eigenvalues of higher multiplicity.

(d) Prove Theorem 3.2.4 (the general solution of a single linear difference equation with constant coefficients), by the application of the Jordan Normal Form to the equation $y_n = A^n y_0$.

(e) Rewrite analogously the differential equation $y^{(k)} + a_1 y^{(k-1)} + \ldots + a_k y = 0$.

(f) What bounds do you obtain for the roots of an algebraic equation by the consideration of the maximum norm or the subordinate logarithmic norm of its companion matrix? Consider also the l_1-norm and weighted variants of both norms, e.g. with weights $w_i = c^i$, for some suitable choice of c.

4. Does (13.9.9) remain valid, if the condition on r_n is replaced by the more liberal condition, $\|r_n\| \leq ch^{p+1}(1 + K_1 h)^{n+1}$?

5. Let $T(t)$ be a real orthogonal matrix for every t, with the derivative $T'(t)$. Show that $\mu_2(\pm T^{-1}T') = 0$. (The application of Theorem 13.9.1 is thus as simple and sharp as it should be in this case.)

Hint: Show that $T^{-1}T'$ is a skew-symmetric matrix.

6. (a) Show that $\|I - hT^{-1}T'(t)\| \approx 1 + h\mu(-T^{-1}T'(t)$.

(b) Prove Theorems 13.9.3 and 13.9.5, by imitating the proofs of Theorem 13.1.10 and Theorem 13.9.1, respectively. You may need the inequality $\|T_{n+1}\|\|T_n\|^{-1} \leq \|\|T_{n+1}T_n^{-1}$.

7. The Mathieu equation reads, $\ddot{u} + (a + b\cos 2\pi t)u = 0$; see, e.g., Coddington and Levinson [6, pp. 218–220]. Determine experimentally, if $a = 1$, for what values of b all solutions of this equation are bounded. (Or, more ambitiously, find the "stability region" for $(a, b) \in \mathbf{R}^2$.)

Hint: Rewrite the equation as a system, and start at $t = 0$. Note that its

fundamental matrix $U(t)$ satisfies the conditions,

$$U(t) = U(t-1)U(1) = \ldots = U(t^*)U(1)^n,$$

where $t = n + t^*$, $0 \le t^* < 1$. The spectrum of $U(1)$ is therefore the crucial thing. Note that $\det U(t) = 1$, according to Problem 22. So, the question is reduced to finding out experimentally for what values of b, a certain condition is satisfied by $U(1)$. What condition?

8. Vacant. Check in Sec.4.6. or 3.2.

9. (a) Show that all solutions of the difference equation

$$y_{n+1} - 2\lambda y_n + y_{n-1} = 0$$

are bounded, as $n \to \infty$, if $-1 < \lambda < 1$, while for any other λ in the complex plane there exists at least one solution which is unbounded.

(b) Let A be a diagonalizable matrix. Give, in terms of the eigenvalues of A, a necessary and sufficient condition for the boundedness as $n \to \infty$ of all solutions of the difference equation

$$y_{n+1} - 2Ay_n + y_{n-1} = 0$$

10. (a), (b) (c) See Problems 9 and 10 of Sec. 8.5 in the old Dahlquist-Björck (unfinished).

11. Difference equation for power series coefficients of a rational function, and the partial fraction decomposition. Relate to Padé, epsilon and Shanks.(unfinished)

12. (a) Assume that the matrices A_n are non-singular, and that all solutions of the linear system $u_{n+1} = A_n u_n$ satisfy the inequality

$$\|u_n\| \le k_0 \|u_j\|, \quad \forall j, \quad 0 \le j \le n.$$

Set $P_n = A_{n-1}A_{n-2}\ldots A_1$, see (13.9.5) and Problem 13 of Sec. 13.1. Show that $\|P_n P_j^{-1}\| \le k_0$.

(b) Assume that $B_n(u)$ is a matrix such that

$$\|B_n(u)\| \le c_1, \ \forall n \ge 0 \ \text{ if } \|u_n\| \le c_2(n).$$

Show that all solutions of the pseudo-linear system $u_{n+1} = \big(A_n + B_n(u)\big)u_n$ satisfy the inequality, $\|u_n\| \le c_0\|u_0\| + c_0 \sum_{j=1}^{n} c_1\|u_{j-1}\|$, as long as this inequality implies that $\|u_n\| \le c_2(n)$.

(c) A difference analog of the Gronwall–Bellman Lemma: Let $\{g_n\}$, $\{k_n\}$, be two scalar sequences, such that $k_n \ge 0$, $\forall n \ge 0$, and set

$$K_j = (1 + k_{j-1})(1 + k_{j-2}) \cdots (1 + k_0).$$

Assume that a (scalar) sequence $\{y_n\}$ for $n \ge 0$ satisfies the inequality $y_n \le g_n + \sum_{j=1}^{n} k_{j-1} y_{j-1}$. Show that

$$y_n \le g_n + \sum_{j=1}^{n} (K_n/K_j) k_{j-1} g_{j-1},$$

Show also that if $k_n = k$ and $g_n = g$ for all $n > 0$, then $y_n \leq (1+k)^n g$. Apply the result to find a simple bound for $\|u_n\|$ in Problem (a).

Hint and note: See the hint and note of Problem 13 of Sec. 13.1.

13. Check the following in 13.2 or 13.4. (a) Plot the stability regions of the BDF methods for $1 \leq k \leq 7$...

(b) Check in Ch3, Sec 13.2, 13.4??

14. Some multistep plots with exceptions: 2-step methods with degenerate S, or rotating twice (unfinished).

15. Runge–Kutta plots with complications: several level curves for Kutta-Simpson; the boundary locus not unicursal. (unfinished.)

16. (a) With the notations of the description of the Routh's algorithm in Sec. 13.8.3, show that, *if* $c_0 > 0$, then the first conditions that come out of the Routh algorithm, applied algebraically, are

$$c_1 > 0,\ c_1c_2 - c_0c_3 > 0,\ c_3(c_1c_2 - c_0c_3) + c_1(c_0c_5 - c_1c_4) > 0, \ldots,$$

and show that $c_j > 0$, for $1 \leq j \leq 3$ are consequences, if we set $c_j = 0$ for $j > 3$.

(b) Test by the Routh algorithm the suspicion that the polynomial $z^5 + z^4 + mz^3 + z^2 + nz + 1$ cannot be a Hurwitz polynomial for any values of the parameters $m,\ n$.

Hint: When one works algebraically it often simplifies to divide the denominator or the numerator by a factor that has to be positive, if the rational function is a positive function. Moreover, the algebra can be simplified by appropriate substitutions.

17. (a) Hurwitz showed in 1895 the criterion for a kth degree to be a "Hurwitz polynomial" is that the first k principal minors of the matrix H with elements $h_{ij} = c_{2j-i}$, $i, j = 1, 2, \ldots k$, should be positive. Here the c_ν are the polynomial coefficients; the notation is the same as in Problem 15 and Theorem 13.9.13, with the conventions that $c_0 > 0$, and $c_\nu = 0$ for $\nu < 0$ and $\nu > k$. Note that each row contains the coefficients of either $p_0(z)$ or $p_1(z)$. Write down the matrix for $k = 5$.

Interpret *the Routh algorithm as an elimination process* for bringing the matrix H to upper triangular form.

(b) Is it true that the condition $c_4 > 0$ together with the three inequalities mentioned in Problem 15 (a), are sufficient for the Hurwitz property, if $k = 4$?

(c) Is it true, that it does not matter for the application of the Hurwitz criterion, if the coefficients are ordered as in our description, with c_0 as the leading coefficient, or the other way around, with c_k as the leading coefficient? Is the analogous thing true for the Schur algorithm?

18. (a) Write a program for the Routh algorith m for a polynomial with numerically given coefficients, as described in Sec. 13.8.3, and test it on suitable polynomials, including cases where there are roots on the imaginary axis.

(b) Write a program for the Schur algorithm, as described in Sec. 13.8.3, and test it on suitable polynomials, including cases where there are roots on the unit circle.

(c) Write a program for the Sturm chain, as described in Sec. 13.8.3, and test it on suitable polynomials. Test also the sensitivity to perturbations of the coefficients.

19. (a) Convince yourself about the validity of the relations in (13.9.11) and (13.9.12) which are not definitions.

(b) The following is a short MATLAB program for the *Greco-Roman transformation*. It computes by recurrence relation the $k \times k$ matrix called `gr` which maps the coefficient vector of $R(z)$ to the coefficient vector of $\rho(\zeta)$. The vectors are column vectors with the leading coefficient last.

```
gr = 1; a1 = 1;
forn = 1 : k,
    bg = [0; a1] − [a1; 0]; z = zeros(1, n);
    c = [gr; z] + [z; gr]; a1 = bg; gr = [bg, c];
end;
```

Read the program, and make sure you understand the algorithm. A semicolon (inside brackets) means partitioning in the vertical direction, while a comma means means partitioning in the horizontal direction. Note that there is dynamic memory allocation, e.g., `a1` is a column, the length of which increases from 1 to k. Is it true that the inverse of `gr` equals 2^{-k}`gr` ?
Note that the algorithm actually computes all transformation matrices for the orders $n = 1, 2, \ldots, k$. Test it on some simple cases, either in MATLAB or after translation to another language.

20. Order Star Problem. Unfinished.

which are functions of t,

We shall now collect some formulas that are useful, when one works with other norms than the ones treated in Sec. 13.1. Let T be a non-singular matrix, and let $\|\cdot\|$ be any vector norm. Then it is easily seen that

$$\|u\|_T = \|T^{-1}u\|. \tag{13.9.18}$$

satisfies the three conditions for a vector norm, stated at the beginning of Sec. 6.2.5.

Next, set $u = Tv$. Then

$$\frac{\|Bu\|_T}{\|u\|_T} = \frac{\|T^{-1}BTv\|}{\|v\|}$$

Since T is non-singular, $\max_u$ means the same as $\max_v$. It follows that the subordinate matrix norm is

$$\|B\|_T = \|T^{-1}BT\|.$$

For $B = I + \epsilon A$ we then obtain, $\|I + \epsilon A\|_T = \|T^{-1}(I + \epsilon A)T\| = \|I + \epsilon T^{-1}AT\|$, and it follows from the definition of the subordinate logarithmic norm that

$$\mu_T(A) = \mu(T^{-1}AT). \tag{13.9.19}$$

Appendix A

Calculus in Vector Spaces

We shall introduce some notions and notations from the calculus in vector spaces that will be useful in this and in later chapters. A more general and rigorous treatment can be found, e.g., in Dieudonné [12]. Our presentation is also much influenced by Butcher [4, Chapter 1], whose purpose is rather similar to ours, but his discussion is stricter. In these books the reader may find some proofs that we omit here. There are, in the literature, several different notations for these matters, e.g., **multilinear mapping** notation, **tensor** notation, or, in some cases, **vector-matrix** notation. None of them seems to be perfect or easy to handle correctly in some complex situations. This may be a reason to become familiar with several notations.

A.1 Multilinear Mappings

Consider $k+1$ vector spaces X_1, $X_2, \ldots$, X_k, Y, and let $x_\nu \in X_\nu$. A function $A\colon X_1 \times X_2 \ldots \times X_k \to Y$ is called k-**linear**, if it is linear in each of its arguments x_i separately. For example, the expression $(Px_1)^T Q x_2 + (Rx_3)^T S x_4$ defines a 4-linear function, mapping or operator (provided that the constant matrices P, Q, R, S have appropriate size). If $k = 2$ such a function is usually called **bilinear**, and more generally one uses the term **multilinear**. , , , ,

Let $X_\nu = \mathbf{R}^{n_\nu}$, $\nu = 1, 2, \ldots, k$, $Y = \mathbf{R}^m$, and let e_{j_i} be one of the basis vectors of X_i. We use *superscripts* to denote coordinates in these spaces. Let $a^i_{j_1,j_2,\ldots,j_k}$ denote the ith coordinate of $A(e_{j_1}, e_{j_2}, \ldots, e_{j_k})$. Then, because of the linearity, the ith coordinate of $A(x_1, x_2, \ldots, x_k)$ reads

$$\sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} \cdots \sum_{j_k=1}^{n_k} a^i_{j_1,j_2,\ldots,j_k} x_1^{j_1} x_2^{j_2} \ldots x_k^{j_k}, \quad x_\nu \in X_\nu. \tag{A.1.1}$$

We shall sometimes use the **sum convention** of tensor analysis; if an index occurs both as a subscript and as a superscript, the product should be summed over the range of this index, i.e., the ith coordinate of $A(x_1, x_2, \ldots, x_k)$ reads shorter $a^i_{j_1,j_2,\ldots,j_k} x_1^{j_1} x_2^{j_2} \ldots x_k^{j_k}$. (Remember always that the superscripts are no exponents.)

Suppose that $X_i = X$, $i = 1, 2, \ldots, k$. Then, the set of k-linear mappings from X^k to Y is itself a linear space called $L_k(X, Y)$. For $k = 1$, we have the space of linear functions, denoted more shortly by $L(X, Y)$. Linear functions can, of course, also be described in vector-matrix notation; $L(\mathbf{R}^n, \mathbf{R}^m) = \mathbf{R}^{m\times n}$, the set of matrices defined in Section 6.2. Matrix notation can also be used for each coordinate of a bilinear function. These matrices are in general unsymmetric.

Norms of multilinear operators are defined analogously to subordinate matrix norms. For example,

$$\|A(x_1, x_2, \ldots, x_k)\|_\infty \le \|A\|_\infty \|x_1\|_\infty \|x_2\|_\infty \ldots \|x_k\|_\infty,$$

where

$$\|A\|_\infty = \max_{i=1}^{m} \sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} \cdots \sum_{j_k=1}^{n_k} |a^i_{j_1,j_2,\ldots,j_k}|. \tag{A.1.2}$$

A multilinear function A is called *symmetric*, if $A(x_1, x_2, ..., x_k)$ is symmetric with respect to its arguments. In the cases mentioned above, where matrix notation can be used, the matrix becomes symmetric, if the multilinear function is symmetric.

We next consider a function $f\colon X \to Y$, not necessarily multilinear, where X and Y are normed vector spaces. This function is *continuous*, at the point $x_0 \in X$ if $\|f(x) - f(x_0)\| \to 0$ as $x \to x_0$, (i.e. as $\|x - x_0\| \to 0$). The function f satisfies a **Lipschitz condition** in a domain $D \subset X$, if a constant α, called a *Lipschitz constant*, can be chosen so that $\|f(x') - f(x'')\| \le \alpha\|x' - x''\|$ for all points x', $x'' \in D$.

The function f is *differentiable* at x_0, in the sense of Fréchet, if there exists a *linear* mapping A such that

$$\|f(x) - f(x_0) - A(x - x_0)\| = o(\|x - x_0\|), \quad x \to x_0.$$

This linear mapping is called the **Fréchet derivative** of f at x_0, and we write $A = f'(x_0)$ or $A = f_x(x_0)$. Note that (the value of) $f'(x_0) \in L(X, Y)$. (Considered as a function of x_0, $f'(x_0)$ is, of course, usually non-linear.)

These definitions apply also to infinite dimensional spaces. In the finite dimensional case, the Fréchet derivative is represented by the **Jacobian** matrix, the elements of which are the partial derivatives $\partial f^i/\partial x^j$, also written f^i_j, in an established notation, e.g., in tensor analysis; superscripts for coordinates and subscripts for partial derivation. If vector-matrix notation is used, it is important to note that the derivative g' of a real-valued function g is a *row* vector, since

$$g(x) = g(x_0) + g'(x_0)(x - x_0) + o(\|x - x_0\|).$$

We suggest that the notation gradient, or grad g is used for the transpose of $g'(x)$.

A *differential* reads, in the multilinear mapping notation, $df = f'dx$ or $df = f_x dx$. In tensor notation with the sum convention, it reads $df^i = f^i_j dx^j$.

Many results from elementary calculus carry over to vector space calculus, such as the rules for the differentiation of products. The proofs are in principle the same.

If $z = f(x, y)$ where $x \in \mathbf{R}^k$, $y \in \mathbf{R}^l$, $z \in \mathbf{R}^m$ then we define *partial derivatives* f_x, f_y with respect to the *vectors* x, y by the differential formula

$$df(x, y) = f_x dx + f_y dy, \quad \forall dx \in \mathbf{R}^k, \quad dy \in \mathbf{R}^l. \tag{A.1.3}$$

If x, y are functions of $s \in \mathbf{R}^n$, then a general version of the *chain rule* reads

$$f'(x(s), y(s)) = f_x x'(s) + f_y y'(s). \tag{A.1.4}$$

The extension to longer chains is straightforward. These equations can also be used in infinite dimensional spaces.

Consider a function $f\colon \mathbf{R}^k \to \mathbf{R}^k$, and consider the equation $x = f(y)$. By formal differentiation, $dx = f'(y)dy$, and we obtain $dy = \big(f'(y)\big)^{-1} dx$, provided that the Jacobian $f'(y)$ is non-singular. In Section 13.2.4, we shall see sufficient conditions for the solvability of the equation $x = f(y)$, so that it defines, in some domain, a differentiable *inverse function of* f, such that $y = g(x)$, $g'(x) = \big(f'(y)\big)^{-1}$.

Another important example: if $f(x, y) = 0$ then, by (A.1.4), $f_x dx + f_y dy = 0$. If $f_y(x_0, y_0)$ is a non-singular matrix, then, by the *implicit function theorem* (see Dieudonné [12, Section 10.2]) y becomes, under certain additional conditions, a differentiable function of x in a neighborhood of (x_0, y_0), and we obtain $dy = -(f_y)^{-1} f_x dx$, hence $y'(x) = -(f_y)^{-1} f_x|_{y=y(x)}$.

One can also show that

$$\lim_{\epsilon \to +0} \frac{f(x_0 + \epsilon v) - f(x_0)}{\epsilon} = f'(x_0)v.$$

There are, however, functions f, where such a *directional derivative* exists for any v but, for some x_0, is not a linear function of v. An important example is $f(x) = \|x\|_\infty$, where $x \in \mathbf{R}^n$. (Look at the case $n = 2$.) The name *Gateaux derivative* is sometimes used in such cases, in order to distinguish it from the Fréchet derivative $f'(x_0)$ previously defined.

If $f'(x)$ is a differentiable function of x at the point x_0, its derivative is denoted by $f''(x_0)$. This is a linear function that maps X into the space $L(X, Y)$ that contains $f'(x_0)$, i.e., $f''(x_0) \in L(X, L(X, Y))$. This space may be identified in a natural way with the space $L_2(X, Y)$ of bilinear mappings $X^2 \to Y$; if $A \in L(X, L(X, Y))$ then the corresponding $\bar{A} \in L_2(X, Y)$ is defined by $(Au)v = \bar{A}(u, v)$ for all $u, v \in X$; in the future it is not necessary to distinguish between A and $\bar{A}$. So,

$$f''(x_0)(u, v) \in Y, \qquad f''(x_0)u \in L(X, Y), \qquad f''(x_0) \in L_2(X, Y).$$

It can be shown that $f''(x_0)\colon X^2 \to Y$, *is a symmetric bilinear mapping*, i.e. $f''(x_0)(u, v) = f''(x_0)(v, u)$. The second order partial derivatives are denoted f_{xx}, f_{xy}, f_{yx}, f_{yy}. One can show that

$$f_{xy} = f_{yx}.$$

If $X = \mathbf{R}^n$, $Y = \mathbf{R}^m$, $m > 1$, $f''(x_0)$ reads $f^p_{ij}(x_0) = f^p_{ji}(x_0)$ in tensor notation. It is thus characterized by a three-dimensional array, which one rarely needs to store

or write. Fortunately, most of the numerical work can be done on a lower level, e.g., with directional derivatives. For each fixed value of p we obtain a symmetric $n \times n$ matrix, named the **Hessian** matrix $H(x_0)$; note that $f''(x_0)(u, v) = u^T H(x_0)v$. The Hessian can be looked upon as the derivative of the gradient. An element of this Hessian is, in the multilinear mapping notation, the pth coordinate of the vector $f''(x_0)(e_i, e_j)$.

We suggest that the vector-matrix notation is replaced by the multilinear mapping formalism when handling derivatives of vector-valued functions of order higher than one. The latter formalism has the further advantage that it can be used also in infinite-dimensional spaces (see Dieudonné [12]). In finite dimensional spaces the tensor notation with the summation convention is another alternative.

Similarly, higher derivatives are recursively defined. If $f^{(k-1)}(x)$ is differentiable at x_0, then its derivative at x_0 is denoted $f^{(k)}(x_0)$ and called the kth derivative of f at x_0. One can show that $f^{(k)}(x_0) : X^k \to Y$ is a *symmetric* k-linear mapping. **Taylor's formula** then reads, when a, $u \in X$, $f : X \to Y$,

$$f(a+u) = f(a) + f'(a)u + \tfrac{1}{2}f''(a)u^2 + \ldots + \frac{1}{k!}f^{(k)}(a)u^k + R_{k+1}, \quad \text{(A.1.5)}$$

$$R_{k+1} = \int_0^1 \frac{(1-t)^k}{k!} f^{(k+1)}(a+ut)dtu^{k+1};$$

it follows that

$$\|R_{k+1}\| \leq \max_{0\leq t\leq 1} \left\| f^{(k+1)}(a+ut) \right\| \frac{\|u\|^{k+1}}{(k+1)!}.$$

After some hesitation, we here use u^2, u^k, etc. as abbreviations for the lists of input vectors (u, u), $(u, u, \ldots, u)$ etc.. This exemplifies simplifications that you may allow yourself (and us) to use when you have got a good hand with the notation and its interpretation. Abbreviations that reduce the number of parentheses often increase the clarity; there may otherwise be some risk for ambiguity, since parentheses are used around the arguments for both the usually non-linear function $f^{(k)}$: $X \to L_k(X, Y)$ and the k-linear function $f^{(k)}(x_0)$: $X^k \to Y$. You may also write, e.g., $(f')^3 = f'f'f'$; beware that you do not mix up $(f')^3$ with f'''.

The mean value theorem of differential calculus and Lagrange's form for the remainder of Taylor's formula are not true, but they can in many places be replaced by the above *integral form of the remainder*. All this holds in complex vector spaces too.

In the following subsections we show some relevant applications of these notions to numerical mathematics.

A.2 Taylor Coefficients for the Solution of a System of Ordinary Differential Equations.

Let y be a function of the real variable t, that satisfies the autonomous differential system $\dot{y} = f(y)$, $f : Y \to Y$.[51] We shall derive recursion formulas for the derivatives

[51] A differential system of equations is said to be autonomous if it does not explicitly contain the independent variable.

of the solution $y(t)$ with respect to t. We use dots for differentiation with respect to t of order less than 3, and we set $\ddot{y} = z$.

By repeated application of the chain rule, the time derivatives of $y(t)$ are expressed in terms of the derivatives of f with respect to the vector y. In the tables below the results are given first in the multilinear mapping notation with primes for differentiation with respect to y (as above). In the last line of the tables, the same vectors are expressed in tensor notation.

$\dot{y}$	$z = \ddot{y}$	$\dot{z} = y^{(3)}$
$f(y)$	$f'(y)\dot{y}$	$f''(y)\dot{y}^2 + f'(y)\ddot{y}$
f	$f'f$	$f''f^2 + (f')^2 f$
f^j	$f^j_k f^k$	$f^j_{kl} f^k f^l + f^j_k f^k_l f^l$

$\ddot{z} = y^{(4)}$
$f'''(y)\dot{y}^3 + 3f''(y)(\ddot{y}, \dot{y}) + f'(y)\dot{z}$
$f'''f^3 + 3f''(f'f, f) + f'f''f^2 + (f')^3 f$
$f^j_{klm} f^k f^l f^m + 3f^j_{km} f^k_l f^l f^m + f^j_k f^k_{lm} f^l f^m + f^j_k f^k_l f^l_m f^m$

Note that, at some places, we have here omitted the *obvious* argument y. We often do so when there is no doubt about the argument.

The individual terms on the third and fourth lines of these tables are called **elementary differentials**. The qth order derivative of y is a linear combination of the qth order elementary differentials with *integer coefficients*. They are fundamental in the theory of one-step methods for ordinary differential equations; see Section 13.3.

These matters can easily become rather messy. J. Butcher and others have made the analysis more transparent by employing an one-to-one correspondence between the qth order elementary differentials and a **rooted tree** with q vertices. We denote a rooted tree by $\mathbf{t}$; its order, that is the number of vertices, is denoted $\rho(\mathbf{t})$, and the corresponding elementary differential is denoted $F(\mathbf{t})$. The qth order trees are denoted $\mathbf{t}_{q1}$, $\mathbf{t}_{q2}, \ldots$.

Table 11.5.1 displays up to order 4 the elementary differentials and trees. (analogous to the tree $\mathbf{t_{32}}$). It corresponds to the elementary differential $(f')^3 f$. Study the table, and see Problem 7. Note the monotonic ordering of the labels along the branches, and see how well the tensor notation corresponds to this labeling. $F(\mathbf{t})$ denotes the elementary differential, which corresponds to the tree $\mathbf{t}$, e.g., $F(\mathbf{t_{21}}) = f'f$. A tree $\mathbf{t}$ can be labeled in several ways. A parameter named $\alpha(\mathbf{t})$ equals, in a certain sense, the number of essentially different monotonic labelings of $\mathbf{t}$; $\alpha(\mathbf{t}) = 1$ for all trees in the figure, except for $\alpha(\mathbf{t_{42}}) = 3$. (Pure permutation of the labels of leaves on the same branch is not "essential".) The precise definition of $\alpha(\mathbf{t})$ is rather subtle, and we refer to Hairer, Nørsett and Wanner [1993, Ch.2] or Butcher loc. cit. for more detailed information. We give in §13.3.1 a table with $\alpha(\mathbf{t})$ and some other data for $\rho(\mathbf{t}) \leq 5$.

Table A.2.1. *Elementary differentials and the corresponding trees up to order* $\rho(\mathbf{t}) = 4$.

order	$\mathbf{t}$	graph	$F(\mathbf{t})$	
1	$\mathbf{t}_{11}$		f	f^j
2	$\mathbf{t}_{21}$		$f'f$	$f^j_k f^k$
3	$\mathbf{t}_{31}$		$f''f^2$	$f^j_{kl} f^k f^l$
	$\mathbf{t}_{32}$		$(f')^2 f$	$f^j_k f^k_l f^l$
4	$\mathbf{t}_{41}$		$f'''f^3$	$f^j_{klm} f^k f^l f^m$
	$\mathbf{t}_{42}$		$f''(f'f, f)$	$f^j_{km} f^k_l f^l f^m$
	$\mathbf{t}_{43}$		$f'f''f^2$	$f^j_{km} f^k_l f^l f^m$
	$\mathbf{t}_{44}$		$(f')^3 f$	$f^j_k f^k_{lm} f^l f^m$

With these notations, the formal Taylor expansion of the solution $y(t)$ around $t = t_0$ reads

$$\begin{aligned} y(t_0 + h) &= y(t_0) + y'(t_0)h + \frac{1}{2!}y''(t_0)h^2 + \frac{1}{3!}y'''(t_0)h^3 + \dots \\ &= y(t_0) + hfy(t_0) + \frac{h^2}{2!}h^2 f'fy(t_0) + \frac{h^3}{3!}(f''f^2 + (f')^2 f)y(t_0) + \dots \\ &= y(t_0) + \Big(hF(\mathbf{t_{11}}) + \frac{h^2}{2!}F(\mathbf{t_{21}}) + \frac{h^3}{3!}\big(F(\mathbf{t_{31}}) + F(\mathbf{t_{32}})\big) + \dots\Big)y(t_0). \end{aligned}$$

More generally, the Taylor expansion becomes,

$$y(t_0 + h) = y(t_0) + \sum_{\mathbf{t}} \frac{h^{\rho(\mathbf{t})}}{\rho(\mathbf{t})!}\alpha(\mathbf{t})F(\mathbf{t})y(t_0), \quad \rho(\mathbf{t}) \geq 1. \tag{A.2.1}$$

This expression is useful for the design and analysis of numerical methods. If you want to use a Taylor expansion for computing the numerical solution of a system, however, you had better use the techniques of *automatic differentiation*, see Section 3.1 and Section 13.3.

The number of elementary differentials for $q = 1 : 10$ are as follows:

$$\begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 1 & 1 & 2 & 4 & 9 & 20 & 48 & 115 & 286 & 719 \end{array} \tag{A.2.2}$$

Much more about this can be found in Butcher, loc. cit., and Hairer, Nørsett and Wanner, loc. cit..

The formulas for an autonomous system, $\dot{y} = f(y)$, include also the non-autonomous case, i.e. a system of the form $\dot{y} = f(t, y)$, for if we add the trivial equation $\dot{t} = 1$ to the latter system, then we obtain an autonomous system for the vector (t, y), (written as a column). Nevertheless, since the variable t plays a special role, it is sometimes interesting to see the formulas for the non-autonomous system more explicitly. Recall that $f_{ty} = f_{yt}$.

$$\begin{aligned} &\dot{y} = f(t, y) \\ &z = \ddot{y} = df(t, y(t))/dt = f_t + f_y\dot{y} = f_t + f_y f \\ &\dot{z} = (f_t + f_y f)_t + (f_t + f_y f)_y f = f_{tt} + f_y f_t + 2f_{yt} f + f_{yy} f^2 + f_y f_y f, \end{aligned}$$

Problems

4. Consider the multilinear operator A defined by (A.1.1), and suppose that $X_\nu = \mathbf{R}^n$, $\forall \nu$. What is $\|A\|$ if a weighted max-norm is used in $\mathbf{R}^n$?

5. Write a program for the approximate computation of the Hessian of a real-valued function, by central differences.

6. Consider an autonomous system, $\dot{y} = f(y)$, $f : \mathbf{R}^s \to \mathbf{R}^s$. Such a system has an infinity of solutions $y(t)$, but we shall see in Section 13.1 that, for given $\tau \in \mathbf{R}$, $\eta \in \mathbf{R}^s$, there is, under very general conditions on the function f, only one solution for which $y = \eta$ for $t = \tau$. Denote this solution by $y(t; \tau, \eta)$. Runge's 2nd order method, introduced in Section 1.3, reads $k_1 = hf(y_n)$, $k_2 = hf\big(y_n + \frac{1}{2}k_1\big)$, $y_{n+1} = y_n + k_2$. Show that

$$y_{n+1} - y(t_n + h; t_n, y_n) = h^3\big(\tfrac{1}{8}f''\dot{y}^2 - \tfrac{1}{6}y'''\big) + O(h^4).$$

(This is called the local error.) Also show that $k_2 - k_1 = \frac{1}{2}h^2\ddot{y} + O(h^3)$. (The vector $k_2 - k_1$ is used for the choice of step size in the algorithm of Section 1.3. See also Section 13.2.)

7. (a) Draw the tree $\mathbf{t}_{44}$, and write down the corresponding elementary differential in multilinear mapping notation and in tensor notation.

(b) Given all trees of order $q - 1$, two ways of producing (different) trees of order q are as follows. You can *either* put one more vertex on the first level above the root (and label it with the next character in the alphabet), *or* you can create a new root (labeled j) below the old one and change the other labels. Note that for $q = 3$ and $q = 4$ these operations yield all trees. Find the rules, how the elementary differentials are modified at these tree operations.

(c) For $q = 5$, however, the operations in (b) produce together 8 trees, instead

of 9, according to the table in Example A.2. What does the missing tree look like? Find the corresponding elementary differential.

Comment: There is more material about this in Section 13.3. How many missing trees are there for $q = 6$? Find the trees and the elementary differentials.

8. Consider a function $f : X \to X$, $\dim X > 1$. Do expressions like $f''(x_0)f''(x_0)$ and $f''(x_0)f''(x_0)f''(x_0)$ ever make sense?

Bibliography

[1] Uri M. Ascher and Linda R. Petzold. *Computer Methods for Ordinary Differential Equations and Differntial-Algebraic Systems*. SIAM, Philadelphia, 1998.

[2] D. Barton, I. M. Willers, and R. V. M. Zahar. Taylor series methods for ordinary differential equations. In John Rice, editor, *Mathematical Software*, pages 369–390. Academic Press, New York, 1971.

[3] K. Brenan, S. Campbell, and Linda R. Petzold. *Numerical Solution ofInitial Value Problems in Differntial-Algebraic Equations*. SIAM, Philadelphia, second edition, 1998.

[4] John C. Butcher. *The Numerical Analysis of Ordinary Differential Equations: Runge–Kutta and General Linear Methods*. John Wiley, New York, NY, 1987.

[5] F. Ceschino and J. Kuntzmann. Impossibilité d'un certain type de formule d'intégration approchées à pas liés. *Chiffres*, 2:249–252, 1958.

[6] E. A. Coddington and N. Levinson. *Theory of Ordinary Differential Equations*. McGraw-Hill, New York, NY, 1955.

[7] L. Collatz. *The Numerical Treatment of Differential Equations*. Springer Verlag, Berlin, third edition, 1960.

[8] Germund Dahlquist. An evaluation of the approximate inverse algorithm for numerical integration of stiff differential equations. *Numer. Math.*, 47(3):363–385, 1985.

[9] A. K. Datta. On transformations of graded matrices with applications to stiff ODE's. Technical Report Report MSH/67/84, Inperical Chemical Industries, Cheshire, 1967.

[10] Ch.-J. de la Vallée Poussin. *Cours d'Analyse Infinitésimale*, volume II. Gautier-Villars, Paris, fifth edition, 1925.

[11] C. A. Desoer and H. Haneda. The measure of a matrix as tool to analyze computer algorithms for circuit analysis. *IEEE Trans. CT*, 19:480–486, 1972.

[12] J. Dieudonné. *Foundations of Modern Analysis*. Academic Press, New York, NY, 1961.

[13] A. Dinghas. *Vorlesungen über Funktionentheori*. Springer, Berlin, 1961.

[14] J. R. Dormand and P. J. Prince. A family of embedded Runge–Kutta formulae. *J. Comp. Appl. Math.*, 6:19–26, 1980.

[15] E. Fehlberg. Klassische Runge–Kutta Formeln fünfter und siebenter Ordnung mit Schrittweiten-Kontrolle. *Computing*, 4:93–106, 1969.

[16] E. Fehlberg. Klassische Runge–Kutta Formeln vierter und niedriger Ordnung mit Schrittweiten-Kontrolle und ihre Anvendung auf Wärmeleitungsprobleme. *Computing*, 6:61–71, 1970.

[17] F. R. Gantmacher. *The Theory of Matrices. Vols. I and II*. Chelsea Publishing Co, New York, 1959.

[18] C. W. Gear. *Numerical Initial Value Problems in Ordinary Differential Equations*. Prentice-Hall, Englewood Cliffs, NJ, 1971.

[19] Ernst Hairer, Ch. Lubich, and Gerhard Wanner. *Geometric Numerical Integration Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer-Verlag, Berlin, 2002.

[20] Ernst Hairer, Syvert P. Nørsett, and Gerhard Wanner. *Solving Ordinary Differential Equations I. Nonstiff Problems*. Springer-Verlag, Berlin, 1993.

[21] Ernst Hairer and Gerhard Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential Algebraic Problems*. Springer-Verlag, Berlin, second edition, 1996.

[22] P. Henrici. *Discrete Variable Methods in Ordinary Differential Equations*. Prentice-Hall, Englewood Cliffs, NJ, 1962.

[23] K. Heun. Neue Methode zur approximativen Integration der Differentialgleichungen einer unabhängigen Variablen. *Z. Math. Phys.*, 45:23–38, 1900.

[24] M. S. Klamkin, editor. *Mathematic Modeling: Classroom Notes in Applied Mathematics*. SIAM, Philadelphia, 1987.

[25] W. Kutta. Beitrag zur näherungsweisen Integration totaler Differentialgleichungen. *Z. Math. Phys.*, 46:435–453, 1901.

[26] Solomon Lefschetz. *Differential Equations: Geometric Theory*, volume Volume VI of *Pure and Applied Mathematics*. Wiley Interscience, New York, second edition, 1963. Republished by Dover Publications 2005.

[27] R. H. Merson. An operational method for the study of integration processes. In *Proceedings Symposium on Data Processing*, pages 629–639., Salisbury, S. Australia, 1957. Weapons Research Establishment.

[28] Ramon E. Moore. *Interval Analysis*. Prentice-Hall, Englewood Cliffs, NJ, 1966.

[29] R. D. Richtmyer and K. W. Morton. *Difference Methods for Initial Value Problems*. John Wiley-Interscience, New York, second edition, 1967.

[30] C. Runge. Über die numerische Auflösung von Differentialgleichungen. *Math. Ann.*, 46:167–178, 1895.

[31] L. F. Shampine. *Numerical Solution of Ordinary Differential Equations*. Chapman & Hall, New York, 1994.

[32] L. F. Shampine, I. Gladwell, and S. Thomson. *Solving ODE's with Matlab*. Cambridge University Press, UK, 2003.

[33] E. B. Shanks. Solutioms of differential equations by evaluation of functions. *Math. Comp.*, 20:21–38, 1966.

[34] Gustaf Söderlind. On nonlinear difference and differential equations. *BIT*, 24:667–680, 1984.

[35] Joseph Stoer and Roland Bulirsch. *Introduction to Numerical Analysis*. Springer-Verlag, New York, third edition, 2002.

[36] Gilbert Strang. *Introduction to Applied Mathematics*. Wellesley-Cambridge Press, Wellesley, MA, 1988.

Index