Numerical Analysis of Ordinary Differential Equations

Guido Kanschat & Robert Scheichl

July 24, 2019

Preface

These notes are a short presentation of the material presented in my lecture. They follow the notes "**Numerik 1:** Numerik gewöhnlicher Differentialgleichungen" by Rannacher (in German) [Ran17b], as well as the books by Hairer, Nørsett, and Wanner [HNW09] and Hairer and Wanner [HW10]. Furthermore, the book by Deuflhard and Bornemann [DB08] was used. Historical remarks are in part taken from the article by Butcher [But96].

We are always thankful for hints and errata.

Thanks go to Dörte Jando, Markus Schubert, Lukas Schubotz, and David Stronczek for their help with writing and editing these notes.

Index for shortcuts

IVP	Initial value problem, s. definition 1.2.7 on page 6
BDF	Backward differencing formula
ODE	Ordinary differential equation
DIRK	Diagonal implicit Runge-Kutta method
ERK	Explicit Runge-Kutta method
IRK	Implicit Runge-Kutta method
LMM	Linear multistep method
VIE	Volterra integral equation, s. Remark 1.2.12 on page 7

Index for symbols

$\mathbb{C}$	The set of complex numbers
e_i	The unit vector of $\mathbb{C}^d$ in direction d
Re	Real part of a complex number
$\mathbb{R}$	The set of real numbers
$\mathbb{R}^d$	The d-dimensional vectorspace of the real d-tuple
u	The exact solution of an ODE or IVP
u_k	The exact solution at time step t_k
y_k	The discrete solution at time step t_k
$\langle x, y \rangle$	The Euclidean scalar product in the space $\mathbb{R}^d$ or $\mathbb{C}^d$
$\lvert x \rvert$	The absolute value of a real number, the modulus of a complex number, or the Euclidean norm in $\mathbb{R}^d$ or $\mathbb{C}^d$, depending on its argument
$\lVert u \rVert$	A norm in a vector space (with exception of the special cases covered by $\lvert x \rvert$)

Contents

Chapter 1

Initial Value Problems and their Properties

1.1 Modeling with ordinary differential equations

Example 1.1.1 (Exponential growth)**.** Bacteria are living on a substrate with ample nutrients. Each bacteria splits into two after a certain time Δt. The time span for splitting is fixed and independent of the individuum. Then, given the amount u_0 of bacteria at time t_0, the amount at $t_1 = t_0 + \Delta t$ is $u_1 = 2u_0$. Generalizing, we obtain

$$u_n = u(t_n) = 2^n u_0, \qquad t_n = t_0 + n\Delta t.$$

After a short time, the number of bacteria will be huge, such that counting is not a good idea anymore. Also, the cell division does not run on a very sharp clock, such that after some time, divisions will not only take place at the discrete times $t_0 + n\Delta t$, but at any time between these as well. Therefore, we apply the continuum hypothesis, that is, u is not a discrete quantity anymore, but a continuous one that can take any real value. In order to accommodate for the continuum in time, we make a change of variables:

$$u(t) = 2^{\frac{t-t_0}{\Delta t}} u_0.$$

Here, we have already written down the solution of the problem, which is hard to generalize. The original description of the problem involved the change of u from one point in time to the next. In the continuum description, this becomes the derivative, which we can now compute from our last formula:

$$\tfrac{d}{dt} u(t) = \frac{\ln 2}{\Delta t} 2^{\frac{t-t_0}{\Delta t}} u_0 = \frac{\ln 2}{\Delta t} u(t).$$

We see that the derivative of u at a certain time depends on u itself at the same time and a constant factor, which we call the growth rate α. Thus, we have arrived at our first differential equation

$$u'(t) = \alpha u(t). \tag{1.1}$$

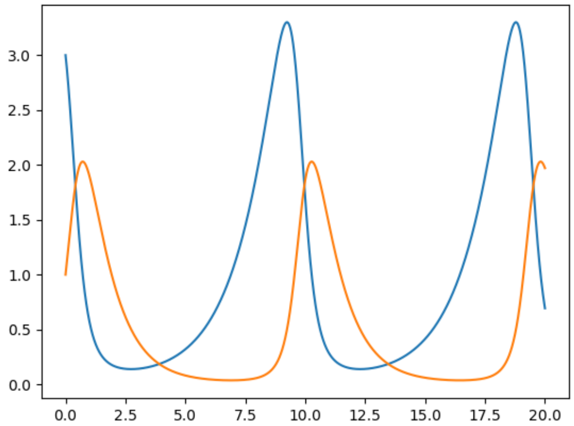


Figure 1.1: Plot of a solution to the predator-prey system with parameters $\alpha = \frac{2}{3}$, $\beta = \frac{4}{3}$, $\delta = \gamma = 1$ and initial values $u(0) = 3$, $v(0) = 1$. Solved with a Runge-Kutta method of order five and step size $h = 10^{-5}$.

What we have seen as well is, that we had to start with some bacteria to get the process going. Indeed, any function of the form

$$u(t) = ce^{\alpha t}$$

is a solution to equation (1.1). It is the initial value u_0, which anchors the solution and makes it unique.

Example 1.1.2 (Predator-prey systems)**.** We add a second species to our bacteria example. Let's say, we replace the bacteria by sardines living in a nutrient rich sea, and we add tuna-eating sardines. The amount of sardines eaten depends on the likelyhood that a sardine and a tuna are in the same place, and on the hunting efficiency β of the tuna. Thus, equation (1.1) is augmented by a negative change in population depending on the product of sardines u and tuna v:

$$u' = \alpha u - \beta uv.$$

In addition, we need an equation for the amount of tuna. In this simple model, we will make two assumptions: first, tuna die of natural causes at a death rate of γ. Second, tuna procreate if there is enough food (sardines), and the procreation rate is proportional to the amount of food. Thus, we obtain

$$v' = \delta uv - \gamma v.$$

Again, we will need initial populations at some point in time to evolve them to later times from that point.

Remark 1.1.3. The predator-prey system (a.k.a. Lotka-Volterra-equations) have periodic solutions. Even though none of these exist in closed form, solutions can be computed numerically (simulated): Lotka and Volterra became interested in this system as they had found that the amount of predatory fish caught had increased during World War I. During the war years there was a strong decrease of fishing effort. In conclusion, they thought, there had to be more prey fish.

A (far too rarely) applied consequence is that in order to diminish the amount of, e.g., foxes one should hunt rabbits, since foxes feed on rabbits.

Example 1.1.4 (Graviational two-body systems)**.** According to Newton's law of universal gravitation, two bodies of masses m_1 and m_2 attract each other with a force

$$\mathbf{F}_1 = G\frac{m_1 m_2}{r^3}\mathbf{r}_1,$$

where $\mathbf{F}_1$ is the force vector acting on m_1 and $\mathbf{r}_1$ is the vector pointing from m_1 to m_2 and $r = |\mathbf{r}_1| = |\mathbf{r}_2|$.

Newton's second law of motion, on the other hand, relates forces and acceleration:

$$\mathbf{F} = m\mathbf{x}'',$$

where $\mathbf{x}$ is the position of a body in space.

Combining these, we obtain equations for the positions of the two bodies:

$$\mathbf{x}_i'' = G\frac{m_{3-i}}{r^3}(\mathbf{x}_i - \mathbf{x}_{3-i}), \qquad i = 1, 2.$$

This is a system of 6 independent variables. However, it can be reduced to three, noting that the distance vector $\mathbf{r}$ is the only variable to be computed for:

$$\mathbf{r}'' = -G\frac{m_1 + m_2}{r^3}\mathbf{r}.$$

Intuitively, it is clear that we need an initial position and an initial velocity for the two bodies. Later on, we will see that this can actually be justified mathematically.

Example 1.1.5 (Celestial mechanics)**.** Now we extend the two-body system to a many-body system. Again, we subtract the center of mass, such that we obtain n sets of 3 equations for an $n + 1$-body system. Since forces simply add up, this system becomes

$$\mathbf{x}_i = -G\sum_{j\neq i}\frac{m_j}{r_{ij}^3}\mathbf{r}_{ij}. \tag{1.2}$$

Here, $\mathbf{r}_{ij} = \mathbf{r}_j - \mathbf{r}_i$ and $r_{ij} = |\mathbf{r}_{ij}|$.

Initial data for the solar system can be obtained from

`https://ssd.jpl.nasa.gov/?horizons`

1.2 Introduction to initial value problems

1.2.1 Definition (Ordinary differential equations): Let $u(t)$ be a function defined on an interval $I \subset \mathbb{R}$ with values in the real or complex numbers or in the space $\mathbb{R}^d$ ($\mathbb{C}^d$). An **ordinary differential equation** (ODE) is an equation for $u(t)$ of the form

$$F\big(t, u(t), u'(t), u''(t), \dots, u^{(n)}(t)\big) = 0. \tag{1.3}$$

Here $F(\dots)$ denotes an arbitrary function of its arguments.

The **order** n of a differential equation is the highest derivative which occurs. If $d > 1$, we talk about **systems of differential equations**.

Remark 1.2.2. A differential equation (DE), which is not ordinary, is called **partial**. These are equations or systems of equations, which involve partial **derivatives with respect to several independent variables**. While the functions in an ordinary differential equation may be dependent on additional parameters, derivatives are only taken with respect to one variable. Often, but not exclusively, this variable is time. This manuscript only deals with ordinary differential equations, and so the adjective will be omitted in the following.

1.2.3 Definition: An **explicit differential equation** of first order is a equation of the form

$$\begin{aligned} u'(t) &= f(t, u(t)) \\ \text{or shorter:} \qquad u' &= f(t, u). \end{aligned} \tag{1.4}$$

A differential equation of order n is called explicit, if it is of the form

$$u^{(n)}(t) = f\left(t, u(t), u'(t), \dots, u^{(n-1)}(t)\right)$$

1.2.4 Lemma: Every differential equation (of arbitrary order) can be written as a system of first-order differential equations. If the equation is explicit, then the system is explicit.

Proof. We introduce the additional variables $u_0(t) = u(t)$, $u_1(t) = u'(t)$ to $u_{n-1}(t) = u^{(n-1)}(t)$. Then, the differential equation in (1.3) can be reformulated as the system

$$\begin{pmatrix} u_0'(t) - u_1(t) \\ u_1'(t) - u_2(t) \\ \vdots \\ u_{n-2}'(t) - u_{n-1}(t) \\ F\big(t, u_0(t), u_1(t), \dots, u_{n-1}(t), u_{n-1}'(t)\big) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}. \tag{1.5}$$

In the case of an explicit equation, the system has the form

$$\begin{pmatrix} u_0'(t) \\ u_1'(t) \\ \vdots \\ u_{n-2}'(t) \\ u_{n-1}'(t) \end{pmatrix} = \begin{pmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_{n-1}(t) \\ f\big(t, u_0(t), u_1(t), \dots, u_{n-1}(t)\big) \end{pmatrix}. \tag{1.6}$$

□

Example 1.2.5. The differential equation

$$u'' + \omega^2 u = f(t) \tag{1.7}$$

can be transformed into the system

$$\begin{aligned} u_1' - u_2 &= 0, \\ u_2' + \omega^2 u_1 &= f(t). \end{aligned} \tag{1.8}$$

The transformation is not uniquely determined. In this example, a more symmetric system can be obtained:

$$\begin{aligned} u_1' - \omega u_2 &= 0, \\ u_2' + \omega u_1 &= f(t). \end{aligned} \tag{1.9}$$

From a numerical perspective, system (1.9) should be chosen over (1.8) to avoid loss of significance or overflow, i.e. if $|\omega| \ll 1$ or $|\omega| \gg 1$.

1.2.6 Definition: A differential equation of the form (1.4) is called **autonomous**, if the right hand side f is not explicitly dependent on t, i.e.

$$u' = f(u). \tag{1.10}$$

Each differential equation can be transformed into an autonomous differential equation. This is called **autonomization**.

$$U = \begin{pmatrix} u \\ t \end{pmatrix}, \qquad F(U) = \begin{pmatrix} f(t,u) \\ 1 \end{pmatrix}, \qquad U' = F(U)$$

A method which provides the same solution for the autonomous differential equation as for the original IVP, is called **invariant under autonomization**.

Differential equations usually provide sets of solutions from which we have to choose a solution. An important selection criterion is setting an initial value which leads to a well-posed problem (see below).

1.2.7 Definition: Given a point $(t_0, u_0) \in \mathbb{R} \times \mathbb{R}^d$ and a function $f(t,u)$ with values in $\mathbb{R}^d$, defined in a neighborhood $I \times U \subset \mathbb{R} \times \mathbb{R}^d$ of (t_0, u_0). Then, an **initial value problem** (IVP) is defined as follows: find a function $u(t)$, such that

$$u'(t) = f\big(t, u(t)\big) \tag{1.11a}$$

$$u(t_0) = u_0 \tag{1.11b}$$

1.2.8 Definition: We call a continuously differentiable function $u(t)$ with $u(t_0) = 0$ a **local solution** of the IVP (1.11), if there exists a neighborhood $J \subset \mathbb{R}$ of t_0, such that $u(t)$ and $f(t, u(t))$ are defined and the equation (1.11a) holds for all $t \in J$.

Remark 1.2.9. We introduced the IVP deliberately in a "local" form because the local solution term is the most useful one for our purposes. Due to the fact that the neighborhood J in the definition above can be arbitrarily small, we will have to deal with the extension to larger intervals below.

Remark 1.2.10. Through the substitution of $t \mapsto \tau$ with $\tau = t - t_0$ it is possible to transform every IVP at the point t_0 to an IVP in 0. We will make use of this fact and soon always assume $t_0 = 0$.

1.2.11 Lemma: Let f be continuous in both arguments. Then, the function $u(t)$ is a solution of the initial value problem (1.11) if and only if it is a solution of the **Volterra integral equation** (VIE)

$$u(t) = u_0 + \int_{t_0}^{t} f\big(s, u(s)\big)\,\mathrm{d}s. \tag{1.12}$$

Remark 1.2.12. The formulation as an integral equation allows on the other hand a more general solution term, because the problem is already well-posed for functions $f(t, u)$, which are just integrable with respect to t. (In that case, the solution u would be just absolutely continuous and not continuously differentiable.) Both the theoretical analysis of the IVP and the numerical methods in this lecture notes (with exception of the BDF methods) are in fact considering the associated integral equation (1.12) and not the IVP (1.11).

1.2.13 Theorem (Peano's existence theorem): Let $\alpha, \beta > 0$ and let the function $f(t, u)$ be continuous on the closed set

$$\overline{D} = \big\{(t, u) \in \mathbb{R} \times \mathbb{R}^d \;\big|\; |t - t_0| \le \alpha,\ |u - u_0| \le \beta\big\}.$$

There exists a solution $u(t) \in C^1(I)$ on the interval $I = [t_0 - T, t_0 + T]$ with

$$T = \min\left(\alpha, \frac{\beta}{M}\right), \quad M = \max_{(t,u)\in\overline{D}} |f(t, u)|.$$

The proof of this theorem is of little consequence for the remainder of these notes. For its verification, we refer to textbooks on the theory of ordinary differential equations or to [Ran17b, Satz 1.1].

Remark 1.2.14. The Peano existence theorem does not make any statements about the uniqueness of a solution and guarantees only local existence. The second limitation is addressed by the following theorem. Uniqueness will be discussed in section 1.4.

1.2.15 Theorem (Peano's continuation theorem): Let the assumptions of Theorem 1.2.13 hold. Then, the solution can be extended to an interval $I_m = [t_-, t_+]$ such that the points $\bigl(t_-, u(t_-)\bigr)$ and $\bigl(t_+, u(t_+)\bigr)$ are on the boundary of $\overline{D}$. Neither the values of t, nor the values of $u(t)$ need to be bounded as long as f remains bounded.

Example 1.2.16. The IVP

$$u' = 2\sqrt{|u|}, \qquad u(0) = 0,$$

has solutions $u(t) = t^2$ and $u(t) = 0$ that both exist for all $t \in \mathbb{R}$ (global existence, but non-uniqueness).

Example 1.2.17. The IVP

$$u' = -u^2, \qquad u(0) = 1.$$

has the unique solution $1/(1+t)$. This solution has a singularity for $t \to -1$ (not global existence, but uniqeuness). However, it exists for all $t > -1$ and thus in particular for all $t > 0 = t_0$, which is all that matters for an IVP.

1.3 Linear ODEs and Grönwall's inequality

1.3.1. The study of linear differential equation turns out to be particularly simple and results obtained here will provide us with important statements for general non-linear IVP. Therefore, we pay particular attention to the linear case.

1.3.2 Definition: An IVP according to definition 1.2.7 is called **linear** if the right hand side f is an affine function of u and the IVP can be written in the form

$$u'(t) = A(t)u(t) + b(t) \qquad \forall t \in \mathbb{R} \tag{1.13a}$$

$$u(t_0) = u_0 \tag{1.13b}$$

with a continuous matrix function $A : \mathbb{R} \to \mathbb{C}^{d\times d}$.

If in addition $b(t) \equiv 0$, we call the IVP **homogeneous**.

1.3.3 Definition: Let the matrix function $A : I \to \mathbb{C}^{d\times d}$ be continuous. Then the function defined by

$$M(t) = \exp\left(-\int_{t_0}^{t} A(s)\,\mathrm{d}s\right) \tag{1.14}$$

is called **integrating factor** of the equation (1.13a).

Corollary 1.3.4. *The integrating factor $M(t)$ has the properties*

$$M(t_0) = \mathbb{I} \tag{1.15}$$

$$M'(t) = -M(t)A(t). \tag{1.16}$$

1.3.5 Lemma: Let $M(t)$ be the integrating factor of the equation (1.13a) defined in (1.14). Then, the function

$$u(t) = M(t)^{-1}\left(u_0 + \int_{t_0}^{t} M(s)b(s)\,\mathrm{d}s\right) \tag{1.17}$$

is a solution of the IVP (1.13) that exists for all $t \in \mathbb{R}$.

Proof. We consider the auxiliary function $w(t) = M(t)u(t)$ with the integrating factor $M(t)$ defined as in eqn. (1.14). It follows by using the product rule that

$$w'(t) = M(t)u'(t) + M'(t)u(t) = M(t)(u'(t) - A(t)u(t)). \tag{1.18}$$

Using the differential equation (1.13a), we obtain

$$w'(t) = M(t)b(t).$$

This can be integrated directly to obtain

$$w(t) = u_0 + \int_{t_0}^{t} M(s)b(s)\,\mathrm{d}s,$$

where we have used (1.15) such that $w(t_0) = M(t_0)u(t_0) = u_0$.

According to lemma A.2.3 about the matrix exponential, $M(t)$ is invertible for all t. Thus we can apply $M(t)^{-1}$ to $w(t)$ to obtain the solution $u(t)$ of (1.13) as given in equation (1.17).

The global solvability follows since the solution is defined for arbitrary $t \in \mathbb{R}$. □

Example 1.3.6. The equation in example 1.2.5 is linear and can be written in the form of (1.13) with

$$A(t) = A = \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix} \quad \text{and} \quad b(t) = \begin{pmatrix} 0 \\ f(t) \end{pmatrix}.$$

Let now $f(t) \equiv 0$, $t_0 = 0$ and $u(0) = u_0$. It is easy to see A has eigenvalues $i\omega$ and $-i\omega$, so that we can write

$$A = C^{-1}\begin{pmatrix} \omega i & 0 \\ 0 & -\omega i \end{pmatrix} C$$

with a suitable transformation matrix C $\boxed{\text{DIY}}$. Using the properties of the matrix exponential, the integrating factor is

$$M(t) = e^{-At} = C^{-1}\begin{pmatrix} e^{-i\omega t} & 0 \\ 0 & e^{i\omega t} \end{pmatrix} C = \begin{pmatrix} \cos\omega t & \sin\omega t \\ -\sin\omega t & \cos\omega t \end{pmatrix}.$$

Thus, the solution is

$$u(t) = \begin{pmatrix} \cos\omega t & -\sin\omega t \\ \sin\omega t & \cos\omega t \end{pmatrix} u_0\,.$$

The missing details in this argument and the case for an inhomogeneity $f(t) = \cos\alpha t$ are left as an exercise $\boxed{\text{DIY}}$.

Remark 1.3.7. If the function $b(t)$ in (1.13a) is only integrable, the function $u(t)$ defined in (1.17) is absolutely continuous and thus differentiable almost everywhere. The chain rule (1.18) is applicable in all points of differentiability and $w(t)$ solves the Volterra integral equation corresponding to (1.13). Thus, the representation formula (1.17) holds generally for solutions of linear Volterra integral equations.

1.3.8 Lemma (Grönwall): Let $w(t)$, $a(t)$ and $b(t)$ be nonnegative, integrable functions, such that $a(t)w(t)$ is integrable. Furthermore, let $b(t)$ be monotonically non-decreasing and let $w(t)$ satisfy the integral inequality

$$w(t) \le b(t) + \int_{t_0}^{t} a(s)w(s)\,\mathrm{d}s, \qquad t \ge t_0. \tag{1.19}$$

Then, for almost all $t \ge t_0$ there holds:

$$w(t) \le b(t) \exp\left(\int_{t_0}^{t} a(s)\,\mathrm{d}s\right). \tag{1.20}$$

Proof. Using the integrating factor

$$m(t) = \exp\left(-\int_{t_0}^{t} a(s)\,\mathrm{d}s\right), \quad \frac{1}{m(t)} = \exp\left(\int_{t_0}^{t} a(s)\,\mathrm{d}s\right),$$

we introduce the auxiliary function

$$v(t) = m(t)\int_{t_0}^{t} a(s)w(s)\,\mathrm{d}s,$$

This function is absolutely continuous, and since $m'(t) = -a(t)m(t)$, we have almost everywhere

$$v'(t) = m(t)a(t)\left[w(t) - \int_{t_0}^{t} a(s)w(s)\,\mathrm{d}s\right].$$

Using assumption (1.19), the bracket on the right can be bounded by $b(t)$. Thus,

$$v'(t) \le m(t)a(t)b(t)$$

and since by definition $v(t_0) = 0$, it follows that

$$v(t) \le \int_{t_0}^{t} m(s)a(s)b(s)\,\mathrm{d}s,$$

which implies, using the definition of $v(t)$, that

$$\int_{t_0}^{t} a(s)w(s)\,\mathrm{d}s = \frac{1}{m(t)}v(t) \le \frac{1}{m(t)}\int_{t_0}^{t} m(s)a(s)b(s)\,\mathrm{d}s.$$

Finally, since $b(t)$ is nondecreasing we obtain almost everywhere

$$\begin{aligned}\int_{t_0}^t a(s)w(s)\,\mathrm{d}s &\le \frac{b(t)}{m(t)}\int_{t_0}^t a(s)\exp\left(-\int_{t_0}^s a(r)\,\mathrm{d}r\right)\,\mathrm{d}s\\ &= \frac{b(t)}{m(t)}\Bigg[-\underbrace{\exp\left(-\int_{t_0}^s a(r)\,\mathrm{d}r\right)}_{m(s)}\Bigg]_{t_0}^t\\ &= \frac{b(t)}{m(t)}\big(m(t_0)-m(t)\big) = \frac{b(t)}{m(t)} - b(t).\end{aligned}$$

Combining this bound with the integral inequality (1.19), we obtain

$$w(t) \le b(t) + \int_{t_0}^t a(s)w(s)\,\mathrm{d}s = \frac{b(t)}{m(t)},$$

which proves the lemma. □

Remark 1.3.9. As we can see from the form of assumption (1.19) and estimate (1.20), the purpose of Grönwall's inequality is to construct a majorant for $w(t)$ that satisfies a linear IVP. The bound is particularly simple when $a, b \ge 0$ are constant.

1.3.10 Corollary: If two solutions $u(t)$ and $v(t)$ of the linear differential equation (1.13a) coincide in a point t_0, then they are identical.

Proof. The difference $w(t) = v(t) - u(t)$ solves the integral equation

$$w(t) = \int_{t_0}^t A(s)w(s)\,\mathrm{d}s.$$

Hence, for an arbitrary vector norm $\|\cdot\|$ (and induced matrix norm also denoted by $\|\cdot\|$), we can obtain the following integral inequality

$$\|w(t)\| \le \int_{t_0}^t \|A(s)w(s)\|\,\mathrm{d}s \le \int_{t_0}^t \|A(s)\|\|w(s)\|\,\mathrm{d}s$$

Now, applying Grönwall's inequality (1.20) with $a(t) = \|A(t)\|$ and $b(t) = 0$, we can conclude that $\|w(t)\| = 0$ and therefore $u(t) = v(t)$, for all t. □

Corollary 1.3.11. *The representation formula* (1.17) *in Lemma 1.3.5 defines the unique solution to the IVP* (1.13)*. In particular, solutions of linear IVPs are always defined for all* $t \in \mathbb{R}$.

Example 1.3.12. Let $A \in \mathbb{C}^{d\times d}$ be diagonalizable with (possibly repeated) eigenvalues $\lambda_1, \dots, \lambda_d$ and corresponding eigenvectors $\psi^{(1)}, \dots, \psi^{(d)}$. The linear IVP

$$\begin{aligned}u' &= Au,\\ u(0) &= u_0.\end{aligned}$$

has unique solution $u(t) = e^{At}u_0$. Using the properties of the matrix exponential (see Appendix A.2.1), with $\Psi \in \mathbb{C}^{d\times d}$ denoting the matrix with ith column $\psi^{(i)}$, we get

$$u(t) = e^{\Psi^{-1}\operatorname{diag}(\lambda_1,\dots,\lambda_d)\Psi t}u_0 = \Psi^{-1}\exp\begin{pmatrix}\lambda_1 t & & \\ & \ddots & \\ & & \lambda_d t\end{pmatrix}\Psi u_0.$$

1.3.13 Lemma: The solutions of the homogeneous, linear differential equation

$$u'(t) = A(t)u(t) \tag{1.21}$$

with $u : \mathbb{R} \to \mathbb{R}^d$, define a vector space of dimension d. Let $\{\psi^{(i)}\}_{i=1,\dots,d}$ be a basis of $\mathbb{R}^d$. Then the solutions $\varphi^{(i)}(t)$ of the equation (1.21) with initial values $\varphi^{(i)}(0) = \psi^{(i)}$ form a basis of the solution space. The vectors $\{\varphi^{(i)}(t)\}_{i=1,\dots,d}$ are linear independent for all $t \in \mathbb{R}$.

Proof. At first we observe that, due to linearity of the derivative and the right hand side, for two solutions $u(t)$ and $v(t)$ of the equation (1.21) and for $\alpha \in \mathbb{R}$, $\alpha u(t) + v(t)$ is also a solution of (1.21) with initial condition $\alpha u(0) + v(0) \in \mathbb{R}^d$. Therefore, the vector space structure follows from the vector space structure of $\mathbb{R}^d$.

Let now $\{\varphi^{(i)}(t)\}$ be solutions of the IVP with linearly independent initial values $\{\psi^{(i)}\}$. As a consequence the functions are linearly independent as well.

Assume that $w(t)$ is a solution of equation (1.21) that cannot be written as a linear combination of the functions $\{\varphi^{(i)}(t)\}$. Then, $w(0)$ is not a linear combination of the vectors $\psi^{(i)}$. Because otherwise, if there exists $\{\alpha_i\}_{i=1,\dots,d}$ with $w(0) = \sum \alpha_i \psi^{(i)}$, then $w(t) = \sum \alpha_i \varphi^{(i)}(t)$ due to the uniqueness of any solution of equation (1.21) proven in corollary 1.3.10, which would lead to a contradiction. However, since $\{\psi^{(i)}\}$ was assumed to form a basis of $\mathbb{R}^d$, that implies $w(0) = 0$ and thus $w \equiv 0$. It follows that the solution space has dimension d and that $\varphi^{(i)}(t)$ forms a basis.

Since in the above argument $t \in \mathbb{R}$ was arbitrary, the $\varphi^{(i)}(t)$ are linearly independent for all $t \in \mathbb{R}$. □

1.3.14 Definition: A basis $\{\varphi^{(1)}, \dots, \varphi^{(d)}\}$ of the solution space of the linear differential equation (1.21), in particular the basis with initial values $\varphi^{(i)}(0) = e_i$, is called **fundamental system** of solutions. The matrix function

$$Y(t) = \big(\varphi^{(1)}(t) \dots \varphi^{(d)}(t)\big) \tag{1.22}$$

with column vectors $\varphi^{(i)}(t)$ is called **fundamental matrix**.

1.3.15 Corollary: The fundamental matrix $Y(t)$ is regular for all $t \in \mathbb{R}$ and it solves the IVP

$$\begin{aligned} Y'(t) &= A(t)Y(t) \\ Y(0) &= \mathbb{I}. \end{aligned}$$

Proof. The initial value is part of the definition. On the other hand, splitting the matrix-valued IVP into its column vectors, we obtain the original family of IVPs defining the solution space. Regularity follows from the linear independence of the solutions for any t. □

1.4 Well-posedness of the IVP

1.4.1 Definition: A mathematical problem is called **well-posed** if the following **Hadamard conditions** are satisfied:

1. A solution exists.
2. The solution is unique.
3. The solution depends continuously on the data.

The third condition in this form is purely qualitative. Typically, in order to characterize problems with good approximation properties, we will require Lipschitz continuity, which has a more quantitative character.

Example 1.4.2. The IVP

$$u' = \sqrt[3]{u}, \qquad u(0) = 0,$$

has infinitely many solutions of the form

$$u(t) = \begin{cases} 0 & \text{for } t \in [0, c], \\ \left(\frac{2}{3}(t-c)\right)^{3/2} & \text{for } t > c, \end{cases}$$

with $c \in \mathbb{R}$. Thus, the solution is not unique and therefore, the IVP is not well-posed.

Let now the initial value be nonzero, but slightly positive. Then, the solution is unique, i.e., $u(t) \approx \left(\frac{2}{3}t\right)^{3/2}$. In contrast, when the initial value is slightly negative, there exists no real-valued solution. Hence, a small perturbation of the initial condition has a dramatic effect on the solution; this is what the third condition for a well-posed problem in definition 1.4.1 excludes.

1.4.3 Definition: The function $f(t, y)$ satisfies a uniform **Lipschitz condition** on the domain $D = I \times \Omega \subset \mathbb{R} \times \mathbb{R}^d$, if it is Lipschitz continuous with respect to y, i.e., there exists a constant $L > 0$, such that

$$\forall t \in I;\ x, y \in \Omega \ :\ |f(t,x) - f(t,y)| \le L|x - y| \tag{1.23}$$

It satisfies a local Lipschitz condition if (1.23) holds for all compact subsets of D.

Example 1.4.4. Let $f(t, y) \in C^1(\mathbb{R} \times \mathbb{R}^d)$ and let all partial derivatives with respect to the components of u be bounded such that

$$\max_{\substack{t\in\mathbb{R} \\ y\in\mathbb{R}^d \\ 1\le i,j\le d}} \left| \frac{\partial}{\partial y_i} f_j(t, y) \right| \le K.$$

Then, f satisfies the Lipschitz condition (1.23) with $L = Kd$. Indeed, by using the Fundamental Theorem of Calculus, we see that

$$\begin{aligned} f_j(t,y) - f_j(t,x) &= \int_0^1 \frac{d}{ds} f_j\bigl(t, x + s(y-x)\bigr)\,\mathrm{d}s \\ &= \int_0^1 \sum_{i=1}^d \frac{\partial}{\partial y_i} f_j\bigl(t, x+s(y-x)\bigr)(y_i - x_i)\,\mathrm{d}s. \end{aligned}$$

Now, exploiting the fact that $|Ax| \le \|A\|_F |x|$, where $\|A\|_F := \sqrt{\sum_{i,j=1}^d a_{ij}^2}$ is the Frobenius norm of the matrix A, we get

$$\begin{aligned} |f(t,y) - f(t,x)| &\le \int_0^1 \left| \sum_{i=1}^d \frac{\partial f}{\partial y_i}\bigl(t, x+s(y-x)\bigr)(y_i - x_i) \right| \mathrm{d}s \\ &\le \int_0^1 \left[\sum_{i,j=1}^d \left| \frac{\partial f}{\partial y_i}\bigl(t, x+s(y-x)\bigr) \right|^2 \right]^{1/2} |y-x|\,\mathrm{d}s \le Kd|y-x|. \end{aligned}$$

1.4.5 Theorem (Stability): Let $f(t,y)$ and $g(t,y)$ be two continuous functions on a cylinder $D = I \times \Omega$ where the interval I contains t_0 and Ω is a convex set in $\mathbb{R}^d$. Furthermore, let f admit a Lipschitz condition with constant L on D. Let u and v be solutions to the IVPs

$$u' = f(t,u) \quad \forall t \in I, \qquad u(t_0) = u_0, \tag{1.24}$$

$$v' = g(t,v) \quad \forall t \in I, \qquad v(t_0) = v_0. \tag{1.25}$$

Then

$$|u(t) - v(t)| \le e^{L|t-t_0|} \left[|u_0 - v_0| + \int_{t_0}^t \max_{x\in\Omega} |f(s,x) - g(s,x)|\,\mathrm{d}s \right]. \tag{1.26}$$

Proof. Both $u(t)$ and $v(t)$ solve their respective Volterra integral equations. Taking the difference, we obtain

$$\begin{aligned} u(t) - v(t) &= u_0 - v_0 + \int_{t_0}^t \bigl[f(s,u(s)) - g(s,v(s))\bigr]\,\mathrm{d}s \\ &= u_0 - v_0 + \int_{t_0}^t \bigl[f(s,u(s)) - f(s,v(s))\bigr]\,\mathrm{d}s + \int_{t_0}^t \bigl[f(s,v(s)) - g(s,v(s))\bigr]\,\mathrm{d}s. \end{aligned}$$

Thus, its norm admits the integral inequality

$$\begin{aligned} |u(t) - v(t)| &\le |u_0 - v_0| + \int_{t_0}^t |f(s,v(s)) - g(s,v(s))|\,\mathrm{d}s + \int_{t_0}^t |f(s,u(s)) - f(s,v(s))|\,\mathrm{d}s \\ &\le \underbrace{|u_0 - v_0| + \int_{t_0}^t \max_{x\in\Omega} |f(s,x) - g(s,x)|\,\mathrm{d}s}_{=:\, b(t)} + \int_{t_0}^t L|u(s) - v(s)|\,\mathrm{d}s. \end{aligned}$$

This inequality is in the form of the assumption in Grönwall's lemma with $a \equiv L$ and $w(t) := |u(t) - v(t)|$, and its application yields the stability result (1.26). □

1.4.6 Theorem (Picard-Lindelöf): Let $f(t,y)$ be continuous on a cylinder

$$D = \{(t,y) \in \mathbb{R} \times \mathbb{R}^d |\ |t-t_0| \le a, |y-u_0| \le b\}.$$

Let f be bounded such that there is a constant $M := \max_{(t,y)\in D}|f(t,y)|$ and satisfy the Lipschitz condition (1.23) with constant L on D. Then the IVP

$$\begin{aligned} u' &= f(t,u) \\ u(t_0) &= u_0 \end{aligned}$$

is uniquely solvable on the interval $I = [t_0 - T, t_0 + T]$ where $T = \min\{a, \frac{b}{M}\}$.

Proof. W.l.o.g., we assume $t_0 = 0$ and let

$$I := [-T, T] \;\text{ and }\; \Omega = \big\{y \in \mathbb{R}^d \big| |y-u_0| \le b\big\}.$$

The Volterra integral equation (1.12) allows us to define the operator

$$F(u)(t) := u_0 + \int_0^t f(s, u(s))\,\mathrm{d}s. \tag{1.27}$$

Obviously, u is a solution of the Volterra integral equation (1.12) if and only if u is a fixed point of F, i.e., $u = Fu$. We can obtain such a fixed-point by the iteration $u^{(k+1)} = F(u^{(k)})$ with some initial guess $u^{(0)} : I \to \Omega$.

From the boundedness of f, we obtain for all $t \le T$ that

$$\left|u^{(k+1)}(t) - u_0\right| = \left|\int_0^t f(s, u^{(k)}(s))\,\mathrm{d}s\right| \le \int_0^t |f(s, u^{(k)}(s))|\,\mathrm{d}s \le TM \le b.$$

Thus, it follows by an inductive argument that $u^{(k)} : I \to \Omega$ is well-defined for all $k \in \mathbb{N}$.

We now show that under the given assumptions, F is a contraction and then apply the Banach Fixed-Point Theorem. We follow the technique in [Heu86, §117] and choose on the space $\mathcal{C}(I)$ of continuous functions defined on I, the weighted maximum-norm

$$\|u\|_e := \max_{t\in I} e^{-2Lt}|u(t)|.$$

Then, for all $u,\ v \in \mathcal{C}(I)$,

$$\begin{aligned} |F(u)(t) - F(v)(t)| &= \left|u_0 - u_0 + \int_0^t (f(s,u(s)) - f(s,v(s)))\,\mathrm{d}s\right| \\ &\le \int_0^t \big|f\big(s,u(s)\big) - f\big((s,v(s)\big)\big|\ \mathrm{d}s \\ &\le \int_0^t L|u(s)-v(s)|\underbrace{e^{-2Ls}e^{2Ls}}_{=1}\,\mathrm{d}s \\ &\le L\|u-v\|_e \int_0^t e^{2Ls}\,\mathrm{d}s \;=\; L\|u-v\|_e \frac{e^{2Lt}-1}{2L} \;\le\; \frac{1}{2}e^{2Lt}\|u-v\|_e \end{aligned}$$

and by multiplying both sides with e^{-2Lt} it follows that

$$\|F(u) - F(v)\|_e \leq \frac{1}{2}\|u - v\|_e.$$

Thus, we have shown that F is a contraction on $(\mathcal{C}(I), \|\cdot\|_e)$. Therefore, we can apply theorem A.3.1, the Banach Fixed-Point Theorem, and conclude that F has exactly one fixed-point, which completes the proof. □

Remark 1.4.7. The norm $\|\cdot\|_e$ had been chosen with regard to Grönwall's inequality, which was not used explicitly in the proof. It is equivalent to the norm $\|\cdot\|_\infty$ because e^{-2Lt} is strictly positiv and bounded. One could have performed the proof (with some extra calculations) also with respect to the ordinary maximum norm $\|\cdot\|_\infty$.

Remark 1.4.8. Currently our solution is restricted to $I = [t_0 - T, t_0 + T]$. Since T is chosen in such a way in theorem 1.4.6 that the graph of u does not leave the domain, this extension always ends at a point (t_1, u_1) in the interior of D with $t_1 := t_0 + T$. One can now extend the solution by solving the next IVP $u' = f(t, u)$ with initial condition $u(t_1) = u_1$ on an interval I_1. This way one obtains a solution on $I \cup I_1 \cup I_2 \cup ....$

If f satisfies a Lipschitz condition everywhere then this leads to the following corollary.

Corollary 1.4.9. *Let the function $f(t, u)$ admit the Lipschitz condition on $\mathbb{R} \times \mathbb{C}^d$. Then, the IVP has a unique solution on the whole real axis.*

Proof. The boundedness was used in order to guarantee that $u(t) \in \Omega$ for any t. This is not necessary anymore, if $\Omega = \mathbb{C}^d$. The fixed point argument does not depend on boundedness of the set. (See *Exercise Sheet 4* for a more detailed proof.) □

Chapter 2

Explicit One-Step Methods and Convergence

2.1 Introduction

Example 2.1.1 (Euler's method)**.** We begin this section with the method that serves as the prototype for a whole class of schemes for solving IVPs numerically.

(As always for problems with infinite dimensional solution spaces, numerical solution refers to finding an approximation by means of a discretization method and the study of the associated discretisation error.)

Consider the following problem:

Given an IVP of the form (1.11) with $t_0 = 0$, calculate the value $u(T)$ at time $T > 0$.

Note first that at the initial point 0, not only the value $u(0) = u_0$ of u, but also the derivative $u'(0) = f(0, u_0)$ are known. Thus, near 0 we are able to approximate the solution $u(t)$ (in blue in Figure 2.1) by a straight line $y(t)$ (in red in Figure 2.1, left) using a first-order Taylor series expansion, i.e.

$$u(t) \approx y(t) = u(0) + tu'(0) = u_0 + tf(0, u_0) \; .$$

The figure suggests that in general the accuracy of this method may not be very good for t far from 0. The first improvement is that we do not draw the line through the whole interval from 0 to T. Instead, we insert intermediate points and apply the method to each subinterval, using the result of the previous interval as the initial point for the next. As a result we obtain a continuous chain of straight lines (in red in Figure 2.1, right) and the so-called **Euler method** (details below).

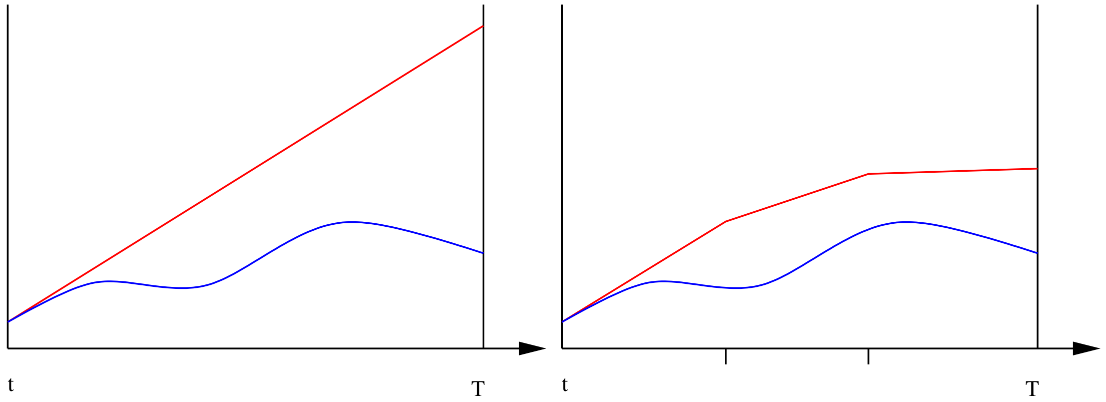
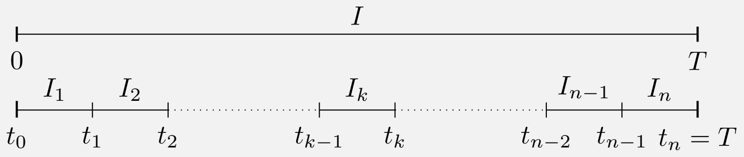


Figure 2.1: Derivation of the Euler method. Left: approximation of the solution of the IVP by a line that agrees in slope and value with the solution at $t = 0$. Right: Euler method with three subintervals.

2.1.2 Definition: On a time interval $I = [0, T]$, we define a partitioning in n subintervals, also known as **time steps**. Here we choose the following notation:



The time steps $I_k = [t_{k-1}, t_k]$ have **step size** $h_k = t_k - t_{k-1}$. A partitioning in n time steps implies $t_n = T$. The term "k-th time step" is used both for the interval I_k and for the point in time t_k (which one is meant will be clear from the context). Very often, we will consider **uniform time steps** and in that case the step size is denoted by h and $h_k = h$, for all k.

Definition 2.1.3 (Notation)**.** In the following chapters we will regularly compare the solution of an IVP with the results of discretization methods. Therefore, we introduce the following convention for notation and symbols.

The solution of the IVP is called the **exact** or **continuous solution**. to emphasize that it is the solution of the non-discretized problem. We denote it in general by u and in addition we use the abbreviation

$$u_k = u(t_k).$$

If u is vector-valued we also use the alternative superscript $u^{(k)}$ and write $u_i^{(k)}$ for the ith component of the vector $u(t_k)$.

The **discrete solution** is in general denoted by y and we write y_k or $y^{(k)}$ for the value of the discrete solution at time t_k. In contrast to the continuous solution, y is only defined at discrete time steps (except for special methods discussed later).

2.1.4 Definition (Explicit one-step methods): An **explicit one-step method** is a method which, given u_0 at $t_0 = 0$ computes a sequence of approximations $y_1 \dots, y_n$ to the solution of an IVP at the time steps $t_1, \dots, t_n$ using an update formula of the form

$$y_k = y_{k-1} + h_k F_{h_k}(t_{k-1}, y_{k-1}). \tag{2.1}$$

The function $F_{h_k}()$ is called **increment function.**[a] We will often omit the index h_k on $F_{h_k}()$ because it is clear that the method is always applied to time intervals. The method is called **one-step method** because the value y_k explicitly depends only on the values y_{k-1} and $f(t_{k-1}, y_{k-1})$, not on previous values.

[a] The adjective 'explicit' is here in contrast to 'implicit' one-step methods, where the increment function depends also on y_k and equation (2.1) typically leads to a nonlinear equation for y_k.

Remark 2.1.5. For one-step methods every step is per definition identical. Therefore, it is sufficient to define and analyze methods by stating the dependence of y_1 on y_0, which then can be transferred to the general step from y_{n-1} to y_n. The general one-step method above then reduces to

$$y_1 = y_0 + h_0 F_{h_0}(t_0, y_0).$$

This implies that the values y_k with $k \geq 2$ are computed through formula (2.1) with the respective h_k and the same increment function (but evaluated at t_{k-1}, y_{k-1}).

2.1.6 Example: The simplest choice for the increment function is $F_{h_k}(t, u) := f(t, u)$, leading to the **Euler method**

$$y_1 = y_0 + hf(t_0, y_0). \tag{2.2}$$

Consider, for example, the (scalar, homogeneous, linear) IVP

$$u' = u, \qquad\qquad u(0) = 1,$$

which has exact solution $u(t) = e^t$. In that case, the Euler method (with uniform time steps) reads

$$y_1 = (1 + h)y_0.$$

The results for $h = 1$ and $h = 1/2$ are:

exact		$h = 1$			$h = 1/2$		
t	$u(t)$	k	y_k	$\lvert u_k - y_k\rvert$	k	y_k	$\lvert u_k - y_k\rvert$
0	1	0	1		0	1	
1	2.71828	1	2	0.718	2	2.25	0.468
2	7.38906	2	4	3.389	4	5.0625	2.236
k	2.71828^k	k	2^k		$2k$	2.25^k	

The error is growing in time. The approximation of the solution is improved by reducing h from 1 to $1/2$. The goal of the following error analysis will be to establish those dependencies.

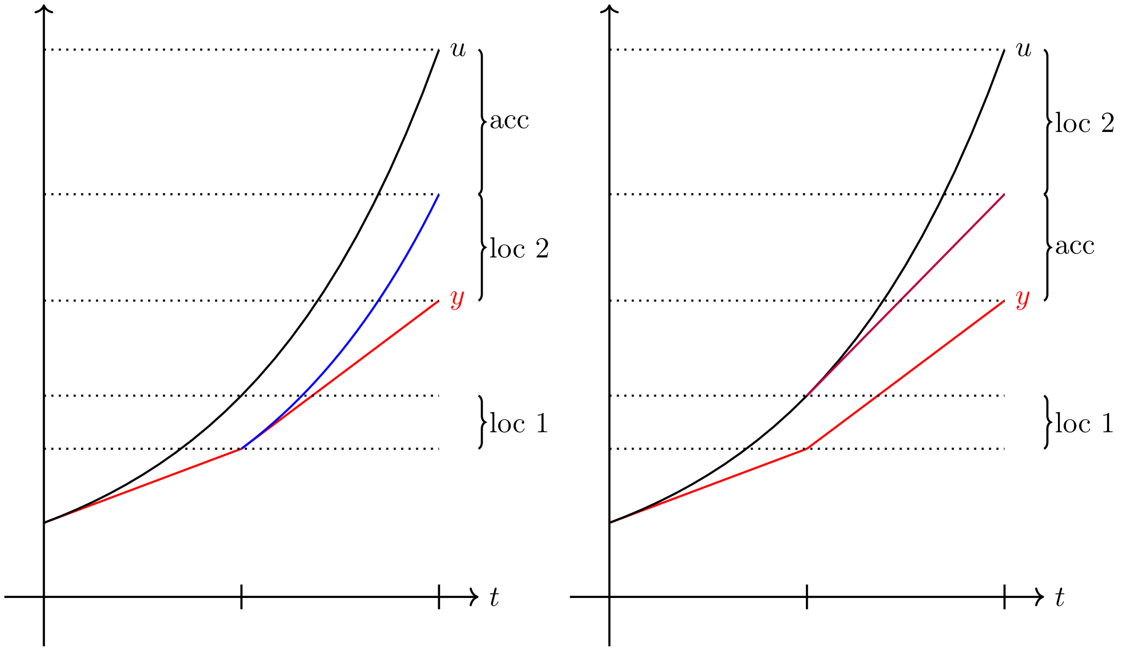


Figure 2.2: Local and accumulated errors. Exact solution in black, the Euler method in red. On the left, in blue the exact solution of an IVP on the second interval with initial value y_1. On the right, in purple the second step of the Euler method, but with exact initial value u_1.

2.2 Error analysis

Remark 2.2.1. In Figure 2.1, we observe that, at a given time t_{k+1}, the error consists of two parts: (i) due to replacing the differential equation by the discrete method on the interval I_k and (ii) due the initial value y_k already being inexact. This is illustrated more clearly in Figure 2.2. Therefore, in our analysis we split the error into the local error and an accumulated error. The local error compares continuous and discrete solutions on a single interval with the same initial value. In the analysis, we will have the options of using the exact (right figure) or the approximated initial value (left figure).

2.2.2 Definition: Let u be a solution of the differential equation $u' = f(t,u)$ on the interval $I = [t_0, t_n] = [0, T]$. Then, the **global error** of a discrete method F_{h_n} is

$$|u(t_n) - y(t_n)|, \tag{2.3}$$

i.e., the difference between the solution u_n of the differential equation at t_n and the result of the one-step method at t_n.

2.2.3 Definition: Let u be a solution of the differential equation $u' = f(t,u)$ on the interval $I_k = [t_{k-1}, t_k]$. Then, the **local error** of a discrete method F_{h_k} is

$$\eta_k = \eta_k(u) = u_k - \bigl[u_{k-1} + h_k F_{h_k}(t_{k-1}, u_{k-1})\bigr], \tag{2.4}$$

i.e., the difference between $u_k = u(t_k)$ and the result of one time step (2.1) with this method with exact initial value $u_{k-1} = u(t_{k-1})$.

The **truncation error** is the quotient of the local error and h_k:

$$\tau_k = \tau_k(u) = \frac{\eta_k}{h_k} = \frac{u_k - u_{k-1}}{h_k} - F_{h_k}(t_{k-1}, u_{k-1}). \tag{2.5}$$

The one-step method $F_{h_k}(t,y)$ is said to have **consistency of order** p, if for all sufficiently regular functions f there exists a constant c independent of $h := \max_{k=1}^n h_k$ such that for $h \to 0$:

$$\max_{k=1}^{n} |\tau_k| \;\le\; ch^p \tag{2.6}$$

Example 2.2.4 (Euler method)**.** To find the order of consistency of the Euler method, where $F_{h_k}(t,y) = f(t,y)$, consider Taylor expansion of u at t_{k-1}:

$$u(t_k) = u(t_{k-1}) + h_k u'(t_{k-1}) + \frac{1}{2}h_k^2 u''(\zeta), \quad \text{for some } \; \zeta \in I_k.$$

As a result the truncation error reduces to:

$$\begin{aligned}\tau_k &= \frac{u_k - u_{k-1}}{h_k} - F_{h_k}(t_{k-1}, u(t_{k-1}))\\ &= \frac{h_k f(t_{k-1}, u_{k-1}) + \frac{1}{2}h_k^2 u''(\zeta)}{h_k} - f(t_{k-1}, u_{k-1}) \;=\; \frac{1}{2}u''(\zeta)\, h_k\,.\end{aligned}$$

If $f \in C^1(D)$ on a compact set D around the graph of u, we can bound the right hand side:

$$\begin{aligned}|\tau_k| \le \frac{1}{2}\max_{t\in I_k}|u''(t)|\, h_k &= \frac{1}{2}\max_{t\in I_k}\Bigl|\frac{\partial f}{\partial t}(t,u(t)) + \nabla_y f(t,u(t))u'(t)\Bigr|\, h_k\\ &\le \underbrace{\frac{1}{2}\max_{(t,y)\in D}\Bigl|\frac{\partial f}{\partial t}(t,y) + \nabla_y f(t,y) f(t,y)\Bigr|}_{=:\,c}\; h_k\,.\end{aligned}$$

Here, we use the assumption that f is sufficiently smooth to conclude that the Euler method is consistent of order 1 (slightly more than Lipschitz continuous).

Next we consider stability of explicit one-step methods. To prove this, we first need a discrete version of Grönwall's inequality.

2.2.5 Lemma (Discrete Grönwall inequality): Let (w_k), (a_k), (b_k) be non-negative sequences of real numbers, such that (b_k) is monotonically nondecreasing. Then, it follows from

$$w_0 \le b_0 \quad \text{and} \quad w_n \le \sum_{k=0}^{n-1} a_k w_k + b_n, \quad \text{for all} \;\; n \ge 1, \tag{2.7}$$

that

$$w_n \le \exp\left(\sum_{k=0}^{n-1} a_k\right) b_n. \tag{2.8}$$

Proof. Let $k \in \mathbb{N}$ and define the functions $w(t)$, $a(t)$, and $b(t)$ such that for all $t \in [k-1,k)$

$$w(t) = w_{k-1}, \quad a(t) = a_{k-1}, \quad b(t) = b_{k-1}.$$

These functions are bounded and piecewise continuous on any finite interval. Thus, they are integrable on $[0,n]$. Therefore, the continuous Grönwall inequality of Lemma 1.3.8 applies and proves the result. □

2.2.6 Theorem (Discrete stability): If $F_{h_k}(t,y)$ is Lipschitz continuous in y for any $t = t_k$, $k < n$, with constant L, then the corresponding one-step method is **discretely stable**, i. e. for arbitrary sequences (y_k) and (z_k), there holds:

$$|y_n - z_n| \le e^{LT} \left(|y_0 - z_0| + \sum_{k=1}^{n} |\eta_k(y) - \eta_k(z)| \right)$$

Proof. Subtracting the equations

$$\begin{aligned} \eta_k(y) &= y_k - y_{k-1} - h_k F_{h_k}(t_{k-1}, y_{k-1}), \\ \eta_k(z) &= z_k - z_{k-1} - h_k F_{h_k}(t_{k-1}, z_{k-1}), \end{aligned}$$

we obtain

$$y_k - z_k = y_{k-1} - z_{k-1} + \eta_k(y) - \eta_k(z) \\ + h_k\big(F_{h_k}(t_{k-1}, y_{k-1}) - F_{h_k}(t_{k-1}, z_{k-1})\big).$$

Recursive application yields

$$|y_n - z_n| \le |y_0 - z_0| + \sum_{k=1}^{n} |\eta_k(y) - \eta_k(z)| + \sum_{k=1}^{n} L h_k |y_{k-1} - z_{k-1}|.$$

The estimate now follows from the discrete Grönwall inequality in Lemma 2.2.5. □

2.2.7 Corollary (One-step methods with finite precision): Let the one-step method F_{h_k} be run on a computer, yielding a sequence (z_k), such that each time step is executed in finite precision arithmetic. Let (y_k) be the mathematically correct solution of the one-step method. Then, the difference equation (2.1) is fulfilled only up to machine accuracy eps, i.e., there exists a $c > 0$:

$$|y_0 - z_0| \le c|z_0|\,\mathrm{eps}\,,$$
$$|\eta_k(y) - \eta_k(z)| = |\eta_k(z)| \le c|z_k|\,\mathrm{eps}\,.$$

Then, the error between the true solution of the one-step method y_n and the computed solution is bounded by

$$|y_n - z_n| \le c\,e^{LT} n \max_{k=0}^{n} |z_k|\,\mathrm{eps}.$$

2.2.8 Theorem (Convergence of one-step methods): Let the one-step method $F_{h_k}(.,.)$ be consistent of order p and Lipschitz continuous in its second argument, for all $t = t_k$, $k < n$. Furthermore, let $y_0 = u_0$ and let $h = \max_{k=1}^{n} h_k$. Then, the global error of the one-step method converges with order p as $h \to 0$ and

$$|u_n - y_n| \le ce^{LT} h^p, \tag{2.9}$$

where the constant c is independent of h.

Proof. Since F_{h_k} is consistent of order p, we have, for all $k = 1, \dots, n$,

$$|\eta_k(u) - \underbrace{\eta_k(y)}_{=0}| = h_k|\tau_k(u)| \le h_k\, 2ch^p, \tag{2.10}$$

where c is the constant in (2.6), which is independent of h.

Now, since F_{h_k} is Lipschitz continuous in its second argument, we can apply the Discrete Stability theorem 2.2.6 and use the bound in (2.10) to obtain

$$|u_n - y_n| \le e^{LT} \sum_{k=1}^{n} |\eta_k(u) - \eta_k(y)| \le e^{LT} \sum_{k=1}^{n} h_k\, 2ch^p = 2cT\, e^{LT} h^p.$$

□

Corollary 2.2.9. *If f is in C^1 in a compact set D around the graph of u over $[0, T]$, then the convergence order of the global error in the Euler method is one.*

Important ! General approach:

$$\boxed{\text{CONSISTENCY} + \text{STABILITY} = \text{CONVERGENCE}}$$

2.3 Runge-Kutta methods

2.3.1. Since the IVP is equivalent to the Volterra integral equation (1.12), we can consider its numerical solution as a quadrature problem. However, the difficulty is that the integrand is not known. It will need to be approximated on each interval from values at earlier times, leading to a class of methods for IVP, called Runge-Kutta methods.

We present the formula again only for the calculation of y_1 from y_0 on the interval from t_0 to $t_1 = t_0 + h$. The formula for a later time step k is obtained by replacing y_0, t_0 and h by y_{k-1}, t_{k-1} and h_k, respectively to obtain y_k. (The coefficients a_{ij}, b_j, c_j remain fixed.)

2.3.2 Definition: An **explicit Runge-Kutta method (ERK)** is a one-step method that uses s evaluations of f with the representation

$$g_i = y_0 + h\sum_{j=1}^{i-1} a_{ij}k_j\,, \qquad i = 1,\dots,s, \tag{2.11a}$$

$$k_i = f\left(t_0 + c_ih, g_i\right)\,, \qquad i = 1,\dots,s, \tag{2.11b}$$

$$y_1 = y_0 + h\sum_{i=1}^{s} b_ik_i\,, \tag{2.11c}$$

i.e., with increment function $F_h(t, y_0) := \sum_{i=1}^{s} b_i f\left(t_0 + c_ih, g_i\right)$. The values $t_0 + c_ih$ are the quadrature points on the interval $[t_0, t_1]$. The values g_i are approximations to the solution $u(t_0+c_ih)$ at the quadrature points, obtained via recursive extrapolation using the evaluations of the function f at previous quadrature points. Since the method uses s intermediate approximations of u on $[t_0, t_1]$, it is called an s-stage method.

Remark 2.3.3. In typical implementations, the intermediate values g_i are not stored separately. However, they are useful for highlighting the structure of the method.

2.3.4 Definition (Butcher tableau): It is customary to write Runge-Kutta methods in the form of a **Butcher tableau**, containing only the coefficients of equation (2.11) in the following matrix form:

$$\begin{array}{c|ccccc}
0 & & & & & \\
c_2 & a_{21} & & & & \\
c_3 & a_{31} & a_{32} & & & \\
\vdots & \vdots & \vdots & \ddots & & \\
c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & \\
\hline
 & b_1 & b_2 & \cdots & b_{s-1} & b_s
\end{array} \tag{2.12}$$

Remark 2.3.5. The first row of the tableau should be read such that $c_1 = 0$, $g_1 = y_0$ and that k_1 is computed directly by $f(t_0, y_0)$. The method is *explicit* since the computation of k_i only involves coefficients with index less than i. The last row below the line is the short form of formula (2.11c) and lists the quadrature weights in the increment function $F_h(t, y_0)$.

Considering the coefficients a_{ij} as the entries of a square $s \times s$ matrix A, we see that A is strictly lower triangular. This is the defining feature of an explicit RK method. We will later see RK methods that also have entries on the diagonal or even in the upper part. Those methods will be called *"implicit"*, because the computation of a stage value also involves the values at the current or future stages. We will also write $b = (b_1, \dots, b_s)^T$ and $c = (0, c_2, \dots, c_s)^T$, such that the Butcher tableau in (2.12) simplifies to

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array}$$

Example 2.3.6. The Euler method Euler method has the Butcher tableau:

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array}$$

That leads to the already known formula:

$$y_1 = y_0 + hf(t_0, y_0)$$

The values $b_1 = 1$ and $c_1 = 0$ indicate that this is a quadrature rule with a single point at the left end of the interval. As a quadrature rule, such a rule is exact for constant polynomials and thus of order 1.

2.3.7 Example (Two-stage methods): The **modified Euler method** is a variant of Euler's method of the following form:

$$\begin{aligned} k_1 &= f(t_0, y_0) \\ k_2 &= f\left(t_0 + \frac{1}{2}h_1, y_0 + h_1\frac{1}{2}k_1\right) \\ y_1 &= y_0 + h_1 k_2 \end{aligned} \qquad \begin{array}{c|cc} 0 & & \\ \frac{1}{2} & \frac{1}{2} & \\ \hline & 0 & 1 \end{array}$$

The so-called **Heun method of order 2** is characterized through the equation

$$\begin{aligned} k_1 &= f(t_0, y_0) \\ k_2 &= f(t_0 + h_1, y_0 + h_1 k_1) \\ y_1 &= y_0 + h\left(\frac{1}{2}k_1 + \frac{1}{2}k_2\right) \end{aligned} \qquad \begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

Remark 2.3.8. The modified Euler method uses one quadrature node at $t_0 + \frac{h}{2} = \frac{t_0+t_1}{2}$ and an approximation to $f(t_0 + \frac{h}{2}, u(t_0 + \frac{h}{2}))$ in F_h, corresponding to the midpoint quadrature rule. The Heun method of order 2 is constructed based on the trapezoidal rule. Both quadrature rules are of second order, and so are these one-step methods. Both methods were discussed by Runge in his article of 1895 [Run95].

2.3.9 Lemma: For f is sufficiently smooth, the Heun method of order 2 and the modified Euler method have consistentcy order two.[a]

[a] Here and in the following proofs of consistency order, we will always assume that all necessary derivatives of f exist and are bounded and simply write"f is sufficiently smooth".

Proof. The proof uses Taylor expansion of the continuous solution u_1 and the discrete solution y_1 around t_0 with $y_0 = u_0$. W.l.o.g. we choose $t_0 = 0$. Considering first only the case $d = 1$ and the abbreviations

$$f_t = \partial_t f(t_0, u_0) \quad \text{and} \quad f_y = \partial_y f(t_0, u_0)$$

and replacing $u'(t_0) = f(t_0, u_0) = f$, we obtain

$$\begin{aligned} u_1 = u(h) = u_0 + hf(t_0, u_0) + \frac{h^2}{2}\big(f_t + f_y f\big) \\ + \frac{h^3}{6}\big(f_{tt} + 2f_{ty}f + f_{yy}f^2 + f_y f_t + f_y^2 f\big) + \dots \end{aligned} \tag{2.13}$$

For the discrete solution of the modified Euler step on the other hand, Taylor expanding f around (t_0, u_0) leads to

$$\begin{aligned} y_1 &= u_0 + hf\left(t_0 + \frac{h}{2}, u_0 + \frac{h}{2}f(t_0, u_0)\right) \\ &= u_0 + hf + \frac{h^2}{2}\big(f_t + f_y f\big) + \frac{h^3}{8}\big(f_{tt} + 2f_{ty}f + f_{yy}f^2\big) + \dots . \end{aligned}$$

Thus, $|\tau_1| = h^{-1}|u_1 - y_1| = \mathcal{O}(h^2)$. Since the truncation error at the kth step can be estimated identically, the method has consistency order two. The proof for the Heun method is left as an exercise.

For $d > 1$, the derivatives with respect to y are no longer scalars, but tensors of increasing rank, or equivalently multilinear operators. Thus, to be precise in d dimensions, $\partial_y f(t_0, u_0)$ is a $d \times d$ matrix that is applied to the vector $f(t_0, u_0)$ and we should write more carefully

$$f_y(f) = \partial_y f(t_0, u_0) f(t_0, u_0).$$

Similarly, $\partial_{yy} f(t_0, u_0)$ is a $d \times d \times d$ rank-3 tensor, or more simply a bilinear operator and to stress this we write $f_{yy}(f, f)$ instead of $f_{yy}f^2$. (However, we will not dwell on this issue in this course.) □

2.3.10 Example: The three stage Runge-Kutta method is

$$\begin{aligned} k_1 &= f(t_0, y_0) \\ k_2 &= f\left(t_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_1\right) \\ k_3 &= f(t_0 + h, y_0 - hk_1 + 2hk_2) \\ y_{n+1} &= y_0 + h\left(\frac{1}{6}k_1 + \frac{4}{6}k_2 + \frac{1}{6}k_3\right) \end{aligned} \qquad \begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & -1 & 2 & \\ \hline & \frac{1}{6} & \frac{4}{6} & \frac{1}{6} \end{array}$$

This method is obviously based on the Simpson rule.

Remark 2.3.11. These Taylor series expansions become tedious very fast. For autonomous ODEs $u' = f(u)$ the analysis simplifies significantly. The Runge-Kutta method (2.11) reduces to

$$
\begin{aligned}
g_i &= y_0 + h \sum_{j=1}^{i-1} a_{ij} f(g_j), \quad i = 1, \dots, s \\
y_1 &= y_0 + h \sum_{j=1}^{s} b_j f(g_j).
\end{aligned}
\tag{2.14}
$$

Each (non-autonomous) ODE can be autonomized (see Def. 1.2.6) using the transformation

$$
U' := \begin{pmatrix} u' \\ t' \end{pmatrix} = \begin{pmatrix} f(t,u) \\ 1 \end{pmatrix} =: F(U). \tag{2.15}
$$

2.3.12 Lemma: An ERK is invariant under autonomization, i.e. its coefficients remain unchanged, if and only if

$$
\sum_{j=1}^{i-1} a_{ij} = c_i, \quad i = 1, \dots, s, \quad \text{and} \quad \sum_{j=1}^{s} b_j = 1\,. \tag{2.16}
$$

Proof. Considering the last components of the vector g_i in (2.14) when applied to the autonomized ODE (2.15) with right hand side $F(\cdot)$, we obtain

$$
t_0 + h \sum_{j=1}^{i-1} a_{ij}\,.
$$

For the ERK to be invariant under autonomization, we require that f is evaluated at $t_0 + hc_i$ in the ith stage leading to the first condition in (2.16). Similarly, the second condition in (2.16) follows from the last component of y_1, when applying (2.14) to (2.15). □

2.3.13 Lemma: An ERK that is invariant under autonomization with s stages is consistent of *first order,* if and only if

$$
b_1 + \dots + b_s = 1; \tag{2.17a}
$$

it is consistent of *second order*, if and only if in addition we have

$$
b_1 c_1 + \dots + b_s c_s = 1/2 \tag{2.17b}
$$

it is consistent of *third order*, if and only if in addition we have

$$
b_1 c_1^2 + \dots + b_s c_s^2 = 1/3, \tag{2.17c}
$$

$$
\textstyle\sum_{i,j=1}^{s} b_i a_{ij} c_j = 1/6; \tag{2.17d}
$$

it is consistent of *fourth order*, if and only if in addition we have

$$
b_1 c_1^3 + \dots + b_s c_s^3 = 1/4, \tag{2.17e}
$$

$$
\textstyle\sum_{i,j=1}^{s} b_i a_{ij} c_j^2 = 1/12, \tag{2.17f}
$$

$$
\textstyle\sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} c_k = 1/24, \tag{2.17g}
$$

$$
\textstyle\sum_{i,j=1}^{s} b_i c_i a_{ij} c_j = 1/8. \tag{2.17h}
$$

Proof. We consider the autonomous ODE $u' = f(u)$ and $u(t_0) = u_0$, where we assume w.l.o.g. again $t_0 = 0$. As in the proof of lemma 2.3.9, we first expand u_1 around t_0, using $u'(t_0) = f(u_0) = f$. Also, since f now only depends on one argument, we abbreviate

$$\frac{\mathrm{d}}{\mathrm{d}t} f(u(t_0)) = \nabla f(u_0) f(u_0) =: f'(f), \quad \text{as well as} \quad \frac{\mathrm{d}^2}{\mathrm{d}t^2} f(u(t_0)) =: f''(f,f) + f'(f'(f)), \quad \ldots$$

Thus, we obtain

$$\begin{aligned} u_1 = u_0 + hf + \frac{h^2}{2} f'(f) + \frac{h^3}{6}\Big(f''(f,f) + f'(f'(f))\Big) \\ + \frac{h^4}{24}\Big(f'''(f,f,f) + 3f''(f'(f),f) + f'(f''(f,f)) + f'(f'(f'(f)))\Big) + \mathcal{O}(h^5). \end{aligned} \tag{2.18}$$

To expand y_1 around $t_0 = 0$ we consider it as a function $y_1(h)$ of the stepsize h. The stage values g_i are also considered as functions $g_i(h)$ of h. First note that

$$y_1(0) = u_0 \quad \text{and} \quad g_i(0) = u_0, \quad \text{for all} \ \ i = 1, \ldots, s. \tag{2.19}$$

To compute the derivatives of y_1 and g_i at $h = 0$, let $q \geq 1$ and note that for an arbitrary function $\varphi = \varphi(h)$, applying Leibniz's rule (the product rule for higher derivatives), gives

$$\begin{aligned} \frac{d^q}{dh^q}\Big(h\varphi(h)\Big)\Big|_{h=0} &= \left[h\varphi^{(q)}(h) + \binom{q}{1}\underbrace{h'}_{=1}\varphi^{(q-1)}(h) + \binom{q}{2}\underbrace{h''}_{=0}\varphi^{(q-2)}(h) + 0\right]_{h=0} \\ &= q\varphi^{(q-1)}(0). \end{aligned} \tag{2.20}$$

Using (2.20) and the definition of an ERK for an autonomous ODE in (2.14) we get

$$y_1^{(q)}(0) = 0 + \sum_{i=1}^{s} b_i \frac{d^q}{dh^q}\Big(hf(g_i(h))\Big)\Big|_{h=0} = q\sum_{i=1}^{s} b_i \frac{d^{q-1}}{dh^{q-1}} f(g_i(h))\Big|_{h=0}, \tag{2.21}$$

$$g_i^{(q)}(0) = 0 + \sum_{j=1}^{s} a_{ij} \frac{d^q}{dh^q}\Big(hf(g_j(h))\Big)\Big|_{h=0} = q\sum_{j=1}^{s} a_{ij} \frac{d^{q-1}}{dh^{q-1}} f(g_j(h))\Big|_{h=0}. \tag{2.22}$$

(where we have assumed again for simplicity that $a_{ij} = 0$, for $j \geq i$).

Finally, we need to apply the chain rule to compute the derivatives of $f(g_i(h))$ needed in (2.21) and (2.22). First for $q = 1$, using again the shorthand notation for the higher derivatives of f as above, it follows from (2.16), (2.19) and (2.22) that

$$\frac{d}{dh} f(g_i(h))\Big|_{h=0} = f'(g_i'(h))\Big|_{h=0} = f'\left(\sum_{j=1}^{s} a_{ij} f(g_j(0))\right) = \sum_{j=1}^{s} a_{ij} f'(f) = c_i f'(f)$$

Similarly, for $q = 2$, we get

$$\begin{aligned} \frac{d^2}{dh^2} f(g_i(h))\Big|_{h=0} &= \left[f''(g_i'(h), g_i'(h)) + f'(g_i''(h))\right]\Big|_{h=0} \\ &= f''\left(\sum_{j=1}^{s} a_{ij} f(g_j(0)), \sum_{j=1}^{s} a_{ij} f(g_j(0))\right) + f'\left(2\sum_{j=1}^{s} a_{ij} \frac{d}{dh} f(g_j(h))\Big|_{h=0}\right) \\ &= c_i^2 f''(f,f) + 2\sum_{j=1}^{s} a_{ij} c_j f'(f'(f)) \end{aligned}$$

The case $q = 3$ can be derived in a similar way and is left as an exercise $\boxed{\text{DIY}}$.

Substituting these formulae into (2.21), we finally obtain

$$
\begin{aligned}
y_1'(0) &= \left(\sum_{i=1}^{s} b_i\right) f, \\
y_1''(0) &= 2\left(\sum_{i=1}^{s} b_i c_i\right) f'(f), \\
y_1'''(0) &= 3\left(\sum_{i=1}^{s} b_i c_i^2\right) f''(f,f) + 6\left(\sum_{i,j=1}^{s} b_i a_{ij} c_j\right) f'(f'(f)),
\end{aligned}
$$

as well as a similar formula for $y_1^{(4)}(0)$, which is again left as an exercise $\boxed{\text{DIY}}$.

Considering now the Taylor series expansion of $y_1(h)$ around $h = 0$, i.e.,

$$
y_1(h) = y_1(0) + hy_1'(0) + \frac{h^2}{2}y_1''(0) + \frac{h^3}{6}y_1'''(0) + \frac{h^3}{24}y_1^{(4)}(0) + \mathcal{O}(h^5)
$$

and comparing coefficients with the coefficients in the expansion of u_1 in (2.18), we obtain the order conditions in (2.17). □

Remark 2.3.14. Butcher introduced a graph theoretical method for order conditions based on trees. While this simplifies the process of deriving these conditions for higher order methods considerably, it is beyond the scope of this course.

2.3.15 Example (The classical Runge-Kutta method of 4th order):

$$
\begin{aligned}
k_1 &= f(t_0, y_0) \\
k_2 &= f\left(t_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_1\right) \\
k_3 &= f\left(t_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_2\right) \\
k_4 &= f(t_0 + h, y_0 + hk_3) \\
y_1 &= y_0 + h(\frac{1}{6}k_1 + \frac{2}{6}k_2 + \frac{2}{6}k_3 + \frac{1}{6}k_4)
\end{aligned}
$$

$$
\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & \frac{1}{2} & & & \\
\frac{1}{2} & 0 & \frac{1}{2} & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6}
\end{array}
$$

Like the 3-stage method in example 2.3.10, this formula is based on Simpson's quadrature rule, but it uses two approximations for the value in the center point and is of fourth order.

Remark 2.3.16 (Order conditions and quadrature)**.** The order conditions derived by recursive Taylor expansion have a very natural interpretation via the analysis of quadrature formulae for the Volterra integral equation, where $c_i h$, $i = 1, \dots, s$, are the quadrature points on $[0, h]$ and the other coefficients are quadrature weights. First, we observe that

$$
h\sum_{i=1}^{s} b_i f(c_i h, g_i) \quad \text{approximates} \quad \int_0^h f(s, u(s))\,\mathrm{d}s.
$$

In this view, conditions (2.17a)–(2.17c) and (2.17e) state that the quadrature formula $\sum_i b_i p(c_i h)$ is exact for polynomials p of degree up to 3. This implies (see Numerik 0) that the convergence of the quadrature rule is of 4th order.

Equally, we deduce from formula (2.11a) for g_i that

$$h\sum_{j=1}^{i-1} a_{ij} f(c_j h, g_j) \quad \text{approximates} \quad \int_0^{c_i h} f(s, u(s))\,\mathrm{d}s.$$

The condition (2.16), which guarantees that the method is autonomizable, simply translates to the quadrature rule being exact for constant functions.

For higher order, the accuracy of the value of g_i only implicitly enters the accuracy of the Runge-Kutta method as an approximation of the integrand in another quadrature rule. Thus, we actually look at approximations of nested integrals of the form

$$\int_0^h \varphi(s) \int_0^s \psi(r)\,\mathrm{d}r\,\mathrm{d}s.$$

Condition (2.17d) for 3rd order states, that this condition must be true for linear polynomials $\psi(r)$ and constant $\varphi(s)$; thus, after the inner integration again any polynomial of second order in the outer rule. Equally, conditions (2.17h) and (2.17f) for fourth order state this for linear polynomials $\psi(r)$ with linear $\varphi(s)$ and for quadratic polynomials $\psi(r)$ with constant $\varphi(s)$, respectively. Finally, condition (2.17g) states that the quadrature has to be exact for any linear polynomial $\varphi(\tau)$ in

$$\int_0^h \int_0^s \int_0^r \varphi(\tau)\,\mathrm{d}\tau\,\mathrm{d}r\,\mathrm{d}s.$$

Remark 2.3.17 (Butcher barriers)**.** The maximal order of an explicit Runge-Kutta method is limited through the number of stages, or vice versa, a minimum number of stages is required for a certain order. The **Butcher barriers** state that in order to achieve consistency of order p one requires s stages, where p and s are related as follows:

p	1	2	3	4	5	6	7	8
# cond.	1	2	4	8	17	37	85	200
s	p	p	p	p	p+1	p+1	p+2	p+3

The Butcher barriers for $p \geq 9$ are not known yet.

2.3.18 Lemma: Let $f(t, y)$ admit a uniform Lipschitz condition on $[0, T] \times \Omega$ with $\{u(t) : t \in [0, T]\} \subset \Omega$. Then every ERK that is invariant under autonomization admits a uniform Lipschitz condition on $[0, T] \times \Omega$.

Proof. The increment function of an ERK is

$$F_h(t, y) = \sum_{j=1}^{s} b_j f\Big(t + c_i h, g_i(y; h)\Big), \tag{2.23}$$

with g_i defined recursively by

$$g_i(y;h) = y + h\sum_{j=1}^{i-1} a_{ij} f\Big(t + c_j h, g_j(y;h)\Big).$$

Let L_f be the Lipschitz constant of f and let $q := hL_f$. Then, for any $x, y \in \Omega$, using (2.16), we get

$$\begin{aligned}
|g_1(y;h) - g_1(x;h)| &= 1\ |y-x| =: L_1|y-x| \\
|g_2(y;h) - g_2(x;h)| &= \Big|y - x + ha_{21}\Big(f\big(t + c_2h, g_1(y;h)\big) - f\big(t+c_2h, g_1(x;h)\big)\Big)\Big| \\
&\le (1 + ha_{21}L_f)|y-x| = (1+qc_2)|y-x| =: L_2|y-x| \\
|g_3(y;h) - g_3(x;h)| &\le \Big(1 + hL_f\big(a_{31} + a_{32}(1 + ha_{21}L_f)\big)\Big)|y-x| \\
&\le \big(1 + qc_3(1+qc_2)\big)|y-x| =: L_3|y-x| \\
&\vdots \\
|g_s(y;h) - g_s(x;h)| &\le \Big(1 + qc_s\big(1 + \ldots(1+qc_2)\big)\ldots\Big)|y-x| =: L_s|y-x|.
\end{aligned}$$

Since $c_i \le 1$, for all $i = 2, \ldots, s$, we can bound

$$L_i \le L_s \le \Big(1 + q\big(1 + \ldots(1+q)\big)\ldots\Big) = 1 + q + q^2 + \ldots + q^{s-1} = \frac{1-q^s}{1-q}$$

Moreover, if $q = hL_f \le 1$ we have $L_s \le s$ and if $q \le 1/2$ we have $L_s \le 2$.

Using the Lipschitz conditions for the g_i together with (2.23) and (2.16) we finally get

$$|F_h(t,y) - F_h(t,x)| \le \sum_{j=1}^{s} b_j L_f L_j |x-y| \le L_f L_s |x-y|.$$

Thus, the increment function F_h admits a Lipschitz condition with constant

$$L := L_f \frac{1-(hL_f)^s}{1-hL_f}$$

for general step size h and with constant $L = 2L_f$ for $h \le (2L_f)^{-1}$. □

Corollary 2.3.19. *Let $f(t,y)$ admit a uniform Lipschitz condition on $[0,T] \times \Omega$ with $\{u(t) : t \in [0,T]\} \subset \Omega$ and let $F_{h_k}(.,.)$ be an ERK that is invariant under autonomization. Then consistency of order p implies convergence with order p and*

$$|u_n - y_n| \le ce^{LT}h^p, \tag{2.24}$$

where L is the Lipschitz constant of F_h from lemma 2.3.18 and the constant c is independent of h.

Proof. Follows directly from lemma 2.3.18 and theorem 2.2.8.

(f Lipschitz ⇒ F_h Lipschitz ⇒ F_h locally stable. Consistency & stability ⇒ convergence)

□

2.4 Estimates of the local error and time step control

2.4.1. The analysis in the last section used a crude a priori bound of the local error based on high-order derivatives of the right hand side $f(t, u)$. In the case of a complex nonlinear system, such an estimate is bound to be inefficient, since it involves global bounds on the derivatives. Obviously, the local error cannot be computed exactly either, because that would require or imply the knowledge of the exact solution.

In this section, we discuss two methods that allow an estimate of the truncation error from computed solutions. These estimates are local in nature and therefore usually much sharper. Thus, they can be used to control the step size, which in turn gives good control over the balance of accuracy and effort.

Nevertheless, in these estimates there is the implicit assumption that the true solution u is sufficiently regular and the step size is sufficiently small, such that the local error already follows the theoretically predicted order.

The main idea is to use two numerical estimates of $u(t_k)$ that converge with different order to estimate the leading order term of the local error for the lower-order method. Given this estimate for the local error, we can then devise an algorithm for step size control that guarantees that the local error of a one-step method remains below a threshold ε in every time step.

Algorithm 2.4.2 (Adaptive step size control)**.** Let y_k and $\widehat{y}_k$ be two approximations of u_k of consistency order p and $\widehat{p} \geq p+1$, respectively, and let $\varepsilon > 0$ be given.

1. Given y_{k-1}, compute y_k and $\widehat{y}_k$ with time step size h_k (both starting from the value y_{k-1} at t_{k-1}).
2. Compute
$$h_{\text{opt}} = h_k \left(\frac{\varepsilon}{|y_k - \widehat{y}_k|} \right)^{\frac{1}{p+1}}. \tag{2.25}$$
3. If $h_{\text{opt}} < h_k$ the time step is rejected; choose $h_k = h_{\text{opt}}$ and recompute y_k and $\widehat{y}_k$.
4. If the time step was accepted, let $h_{k+1} = \min(h_{\text{opt}}, t_n - t_k)$.
5. Increase k by one and return to Step 1.

Remark 2.4.3. When t_k is close to t_n, then the choice $h_{k+1} = h_{\text{opt}}$ in Step 4, leads to $t_n - t_{k+1} \approx$ eps (machine epsilon). To avoid round-off errors in the next time step, it is advisable to choose $h_{k+1} = t_n - t_k$ already for $t_n - t_k \leq c h_{k+1}$, where c is a moderate constant of size around 1.1. This way we avoid that $h_{k+2} \approx$ eps.

Remark 2.4.4. This algorithm controls and equilibrates the local error. It does not control the accumulated global error. The global error estimate still retains the exponential term. Global error estimation techniques involve considerably more effort and are beyond the scope of this course.

The algorithm does not provide an estimate for the leading-order term in the local error of $\widehat{y}_k$. However, since it is a higher order approximation than y_k, we should use $\widehat{y}_k$ as the approximation of u at t_k and as the initial value for the next time step.

Let us now discuss two techniques to compute higher-order approximations $\widehat{y}_k$ of u_k.

2.4.1 Extrapolation methods

2.4.5. Here, we estimate the local error by a method called Richardson extrapolation (cf. Numerik 0). It is based on computing two approximations with the same method, but different step size. In particular, we will use an approximation y_{k+1} with two steps of size h_k and an approximation Y_{k+1} with one step of size $2h_k$, both starting with the same initial value at t_{k-1}.

2.4.6 Theorem: Let y_2 be the approximation of u_2 obtained after two steps of an ERK with step size h and let Y_2 be the approximation after one step of the same method with step size $2h$, both starting from u_0 at t_0. If f is sufficiently smooth and the ERK is consistent of order p, then we can define

$$\widehat{y}_2 = \frac{2^p y_2 - Y_2}{2^p - 1}, \tag{2.26}$$

and have

$$|u_2 - \widehat{y}_2| = O(h^{p+2}). \tag{2.27}$$

Proof. The exact form of the leading-order term in the local error of an ERK of order p can be obtained by explicitly calculating the leading order term in the Taylor expansion in Lemma 2.3.13, i.e. there exists a constant vector $\zeta_k = \zeta_k\left(f_{k-1}, f'_{k-1}, \dots, f^{(p)}_{k-1}\right) \in \mathbb{R}^d$ independent of h such that

$$\eta_k(u) = \zeta_k h^{p+1} + \mathcal{O}(h^{p+2}),$$

where $f^{(j)}_{k-1}$ denotes the jth derivative of f evaluated at (t_{k-1}, y_{k-1}), for $k = 1, 2$. Moreover, since $t_1 = t_0 + h$ and $y_1 = y_0 + \mathcal{O}(h)$, we can also deduce via Taylor expansion that $f^{(j)}_1 = f^{(j)}_0 + \mathcal{O}(h)$ so that $\zeta_2 = \zeta_1 + \mathcal{O}(h)$ and thus

$$\eta_k(u) = \zeta_1 h^{p+1} + \mathcal{O}(h^{p+2}), \quad k = 1, 2. \tag{2.28}$$

Furthermore, we can also use Taylor series expansion of F_h around (t_1, u_1) to obtain

$$\begin{aligned} F_h(t_1, y_1) &= F_h(t_1, u_1) + \nabla_y F_h(t_1, u_1)\eta_k(u) + \mathcal{O}\left(|\eta_k(u)|^2\right) \\ &= F_h(t_1, u_1) + h^{p+1}\nabla_y F_h(t_1, u_1)\zeta_1 + \mathcal{O}(h^{p+2}). \end{aligned} \tag{2.29}$$

Thus, following the same proof technique as in theorem 2.2.6, we obtain for the error after two steps of size h,

$$\begin{aligned} u_2 - y_2 &= \underbrace{u_0 - y_0}_{=0} + \sum_{k=1}^{2}\Big[\eta_k(u) - \underbrace{\eta_k(y)}_{=0} + hF_h(t_{k-1}, u_{k-1}) - hF_h(t_{k-1}, y_{k-1})\Big] \\ &= \sum_{k=1}^{2}\eta_k(u) + h\left[F_h(t_1, u_1) - F_h(t_1, y_1)\right] \\ &= 2\zeta_1 h^{p+1} + \mathcal{O}(h^{p+2}) + h\left[h^{p+1}\nabla_y F_h(t_1, u_1)\zeta_1 + \mathcal{O}(h^{p+2})\right] \\ &= 2\zeta_1 h^{p+1} + \mathcal{O}(h^{p+2}) \end{aligned} \tag{2.30}$$

On the other hand,

$$u_2 - Y_2 = \zeta_1(2h)^{p+1} + \mathcal{O}(h^{p+2}) = 2^{p+1}\zeta_1 h^{p+1} + \mathcal{O}(h^{p+2}). \tag{2.31}$$

Taking 2^p times equation (2.30) and subtracting equation (2.31), we can eliminate the leading order term and obtain

$$\mathcal{O}(h^{p+2}) = 2^p(u_2 - y_2) - (u_2 - Y_2) = (2^p - 1)u_2 - (2^P y_2 - Y_2) = (2^p - 1)(u_2 - \widehat{y}_2)$$

which completes the proof. □

Remark 2.4.7. Thus, $\widehat{y}_2$ provides an approximation of u_2 with consistency order $p+1 > p$ and can thus be used in Algorithm 2.4.2 above to control the step size in each step. In particular, $\widehat{y}_{k+1}$ can be computed cheaply from y_{k+1} and Y_{k+1} via formula (2.26) (with index $k+1$ instead of 2). As mentioned in remark 2.4.4, in practice we expect a better global accuracy, if we use $\widehat{y}_{k-1}$ instead of y_{k-1} as the initial value at t_{k-1} for computing y_{k+1} and Y_{k+1}.

However, in general the computation of Y_2 requires $s-1$ additional evaluations of f, since the stage values will differ from those of y_1 and y_2, leading to a total of $3s-1$ function evaluations for two time steps for this $p+1$-order method. An alternative, that uses the optimal number of stage values s for a $p+1$-order method and reuses all stage values of the lower-order method will be discussed in the next section.

2.4.2 Embedded Runge-Kutta methods

Instead of estimating the local error by doubling the step size, embedded Runge-Kutta methods use two methods of different order to achieve the same effect. The key to efficiency is here, that the computed stages g_i are the same for both methods, and only the quadrature weights b_i differ.

Definition 2.4.8 (Embedded Runge-Kutta methods)**.** An embedded s-stage Runge-Kutta method with orders of consistence p and $\widehat{p}$ computes two approximations y_k and $\widehat{y}_k$ of u_k with the same function evaluations. For this purpose, the stage values g_i and k_i at $t_0 + c_i h_k$ are identical for all $i = 1, \dots, s$, i.e. both methods have the same coefficients a_{ij} and c_i. To compute the final approximations at time step t_k, we use two different quadrature rules, i.e.

$$\begin{aligned} y_k &= y_{k-1} + h_k \sum b_i k_i\,, \\ \widehat{y}_k &= y_{k-1} + h_k \sum \widehat{b}_i k_i\,, \end{aligned} \tag{2.32}$$

such that y_k and $\widehat{y}_k$ are consistent of order p and $\widehat{p} > p$, respectively. Typically, $\widehat{p} = p+1$.

2.4.9 Definition: The Butcher tableau for the embedded method has the form:

$$\begin{array}{c|ccccc} 0 & & & & & \\ c_2 & a_{21} & & & & \\ c_3 & a_{31} & a_{32} & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & \\ \hline & \widehat{b}_1 & \widehat{b}_2 & \cdots & \widehat{b}_{s-1} & \widehat{b}_s \\ & b_1 & b_2 & \cdots & b_{s-1} & b_s \end{array}$$

Remark 2.4.10. For higher order methods or functions $f(t,u)$ with complicated evaluation, most of the work lies in the computation of the stages. Thus, the additional quadrature for the computation of y_k is almost for free. Nevertheless, due to the different orders of approximation, $\widehat{y}_k$ is much more accurate and we obtain

$$u_k - y_k = \widehat{y}_k - y_k + \mathcal{O}(h^p). \tag{2.33}$$

Thus, $\widehat{y}_k - y_k$ is a good estimate for the local error in y_k. This is the error which is used in step size control below. However, as in the Richardson extrapolation above, we use the more accurate value $\widehat{y}_k$ as the final approximation at t_k and as the initial value for the next time step, even if we do not have a computable estimate for its local error.

2.4.11 Definition (Dormand-Prince 45): The embedded Runge-Kutta method of orders 5 for $\widehat{y}_k$ and 4 for y_k due to Dormand and Prince has the Butcher tableau

$$\begin{array}{c|ccccccc}
0 & & & & & & & \\
1/5 & 1/5 & & & & & & \\
3/10 & 3/40 & 9/40 & & & & & \\
4/5 & 44/45 & -56/15 & 32/9 & & & & \\
8/9 & \frac{19372}{6561} & \frac{-25360}{2187} & \frac{64448}{6561} & \frac{-212}{729} & & & \\
1 & \frac{9017}{3168} & \frac{-355}{33} & \frac{46732}{5247} & \frac{49}{176} & \frac{-5103}{18656} & & \\
1 & \frac{35}{384} & 0 & \frac{500}{1113} & \frac{125}{192} & \frac{-2187}{6784} & \frac{11}{84} & \\
\hline
\widehat{y}_k & \frac{35}{384} & 0 & \frac{500}{1113} & \frac{125}{192} & \frac{-2187}{6784} & \frac{11}{84} & 0 \\
y_k & \frac{5179}{57600} & 0 & \frac{7571}{16695} & \frac{393}{640} & \frac{-92097}{339200} & \frac{187}{2100} & \frac{1}{40}
\end{array}$$

It has become a standard tool for the integration of IVP and it is the backbone of `ode45` in Matlab.

Chapter 3

Implicit One-Step Methods and Long-Term Stability

In the first chapter, we studied methods for the solution of IVP and the analysis of their convergence with shrinking step size h. We could gain a priori error estimates from consistency and stability for sufficient small h.

All of these error estimates are based on Grönwall's inequality. Therefore, they contain a term of the form e^{Lt} which increases fast with increasing length of the time interval $[t_0, T]$. Thus, the analysis is unsuitable for the study of long-term integration, since the exponential term will eventually outweigh any term of the form h^p.

On the other hand, for instance our solar system has been moving on stable orbits for several billion years and we do not observe an exponential increase of velocities. Thus, there are in fact applications for which the simulation of long time periods is worthwhile and where exponential growth of the discrete solution would be extremely disturbing.

This chapter first studies conditions on differential equations with bounded long term solutions, and then discusses numerical methods mimicking this behavior.

3.1 Monotonic initial value problem

Example 3.1.1. We consider for $\lambda \in \mathbb{C}$ the (scalar) linear initial value problem

$$\begin{aligned} u' &= \lambda u \\ u(0) &= 1. \end{aligned} \tag{3.1}$$

Splitting $\lambda = \mathrm{Re}(\lambda) + i\,\mathrm{Im}(\lambda)$ into its real and imaginary part, the (complex valued) solution to this problem is

$$u(t) = e^{\lambda t} = e^{\mathrm{Re}(\lambda)t}\big(\cos(\mathrm{Im}(\lambda)t) + i\,\sin(\mathrm{Im}(\lambda)t)\big).$$

The behavior of $u(t)$ for $t \to \infty$ is determined by the real part of λ:

$$\begin{aligned} \mathrm{Re}(\lambda) < 0: &\qquad u(t) \to 0 \\ \mathrm{Re}(\lambda) = 0: &\qquad |u(t)| = 1 \\ \mathrm{Re}(\lambda) > 0: &\qquad u(t) \to \infty \end{aligned} \tag{3.2}$$

Moreover, the solution is bounded for λ with non-positive real part for all points in time t.

Remark 3.1.2. Since we deal in the following again and again with eigenvalues of real-valued matrices and these eigenvalues can be complex, we will always consider complex valued IVP hereafter.

Remark 3.1.3. Due to Grönwall's inequality and the stability Theorem 1.4.5, the solution to the IVP above admits the estimate $|u(t)| \le e^{|\lambda|t}|u(0)|$. This is seen easily by applying the comparison function $v(t) \equiv 0$. As soon as $\lambda \neq 0$ has a non-positive real part, this estimate is still correct but very pessimistic and therefore useless for large t. Since problems with bounded long-term behavior are quite important in applications, we will have to introduce an improved notation of stability.

3.1.4 Definition: The function $f(t,y)$ satisfies on its domain $D \subset \mathbb{R} \times \mathbb{C}^d$ a **one-sided Lipschitz condition** if the inequality

$$\operatorname{Re}\langle f(t,y) - f(t,x), y - x\rangle \le \nu|y-x|^2 \tag{3.3}$$

holds with a constant ν for all $(t,x),(t,y) \in D$. Moreover such a function is called **monotonic** if $\nu = 0$, thus

$$\operatorname{Re}\langle f(t,y) - f(t,x), y - x\rangle \le 0. \tag{3.4}$$

An ODE $u' = f(u)$ is called monotonic if its right hand side f is monotonic.

Remark 3.1.5. The term monotonic from the previous definition is consistent with the term *monotonically decreasing*, which we know from scalar, real-valued functions. We can see this by observing that, for $y > x$,

$$\big(f(t,y) - f(t,x)\big)(y-x) \le 0 \quad \Leftrightarrow \quad f(t,y) - f(t,x) \le 0.$$

3.1.6 Theorem: Let $u(t)$ and $v(t)$ be two solutions of the equation

$$u' = f(t,u), \qquad v' = f(t,v),$$

with initial values $u(t_0) = u_0$ and $v(t_0) = v_0$, resp. Let the function f be continuous and let the one-sided Lipschitz condition (3.3) hold. Then we have for $t > t_0$:

$$|v(t) - u(t)| \le e^{\nu(t-t_0)}|v(t_0) - u(t_0)|. \tag{3.5}$$

Proof. We consider the auxiliary function $m(t) = |v(t) - u(t)|^2$ and its derivative

$$\begin{aligned} m'(t) &= 2\operatorname{Re}\big\langle v'(t) - u'(t), v(t) - u(t)\big\rangle \\ &= 2\operatorname{Re}\big\langle f\big(t, v(t)\big) - f\big(t, u(t)\big), v(t) - u(t)\big\rangle \\ &\le 2\nu|v(t) - u(t)|^2 \\ &= 2\nu m(t). \end{aligned}$$

According to Grönwall's inequality (lemma 1.3.8 on page 10) we obtain for $t > t_0$:

$$m(t) \le m(t_0)e^{2\nu(t-t_0)}.$$

Taking the square root yields the stability estimate (3.5). □

Remark 3.1.7. As in example 3.1.1, we obtain from the stability estimate, that for the difference of two solutions $u(t)$ and $v(t)$ of the differential equation $u' = f(t,u)$ (with different initial conditions) we obtain in the limit $t \to \infty$:

$$\begin{aligned} \nu < 0: &\qquad |v(t) - u(t)| \to 0 \\ \nu = 0: &\qquad |v(t) - u(t)| \le |v(t_0) - u(t_0)| \end{aligned} \tag{3.6}$$

3.1.8 Lemma: Let $A(t) \in \mathbb{C}^{d\times d}$ be a diagonalizable matrix function with eigenvalues $\lambda_j(t)$, $j = 1,\dots,d$. Then the linear function $f(t,y) := A(t)y$ admits the one-sided Lipschitz condition (3.3) on all of $\mathbb{R} \times \mathbb{C}^d$ with the constant

$$\nu = \max_{\substack{j=1,\dots,d\\ t\in\mathbb{R}}} \operatorname{Re}(\lambda_j(t)).$$

Furthermore, the linear differential equation $u' = Au$ with $u(t) \in \mathbb{C}^d$ is monotonic if and only if

$$\operatorname{Re}(\lambda_j(t)) \le 0, \quad \text{for all } t \in \mathbb{R}. \tag{3.7}$$

(This is the vector-valued form of example 3.1.1.)

Proof. For the right hand side of the equation, we have

$$\operatorname{Re}\langle A(t)y - A(t)x, y-x\rangle = \operatorname{Re}\frac{\langle A(t)y - A(t)x, y-x\rangle}{|y-x|^2}|y-x|^2 \le \max_{j=1,\dots,d} \operatorname{Re}(\lambda_j(t))|y-x|^2.$$

This shows that $\nu \le \max_{j=1,\dots,d;\, t\in\mathbb{R}} \operatorname{Re}(\lambda_j(t))$. If we now insert for $y - x$ an eigenvector of eigenvalue $\lambda_j(t)$ for which the maximum is attained, then we obtain the equality and therefore $\nu = \max_{j=1,\dots,d;\, t\in\mathbb{R}} \operatorname{Re}(\lambda_j)$. □

3.1.1 Stiff initial value problems

Example 3.1.9. We consider the IVP

$$u' = Au \quad \text{with} \quad A := \begin{pmatrix} -21 & 19 & -20 \\ 19 & -21 & 20 \\ 40 & -40 & -40 \end{pmatrix} \quad \text{and} \quad u(0) = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}. \tag{3.8}$$

The eigenvalues of A are $\lambda_1 = -2$ and $\lambda_{2,3} = -40 \pm 40i$. The exact solution is

$$u(t) = \begin{pmatrix} \frac{1}{2}e^{-2t} + \frac{1}{2}e^{-40t}[\cos 40t + \sin 40t] \\ \frac{1}{2}e^{-2t} - \frac{1}{2}e^{-40t}[\cos 40t + \sin 40t] \\ -e^{-40t}[\cos 40t - \sin 40t] \end{pmatrix} \to 0 \quad \text{as} \quad t \to \infty.$$

For small time $0 \le t \le 0.2$ all three components are changing rapidly due to the trigonometric terms, since the factor e^{-40t} in front of them is still fairly big. Thus, it is necessary to choose small time step sizes $h \ll 1$.

For $t > 0.2$, we have $u_3 \approx 0$ and $u_1 \approx u_2$, and both those components change fairly slowly, so we could choose a larger time step size $h \geq 0.1$.

However, if we consider the explicit Euler method applied to (3.8) we get

$$y^{(n)} = y^{(n-1)} + hAy^{(n-1)}$$

and thus

$$y^{(n)} = (I + hA)^n \, u_0 \, .$$

Now, if we choose a time step size of $h = 0.01$ the matrix $I + hA$ has eigenvalues $\mu_j = 0.98, 0.6 + 0.4i, 0.6 - 0.4i$, so that

$$|y^{(n)}| = |(I + hA)^n \, u_0| \leq \|I + hA\|^n |u_0| = 0.98^n \sqrt{2} \;\to\; 0 \quad \text{as} \quad t \to \infty,$$

which is, at least qualitatively, the correct behaviour.

For $h = 0.1$, $I + hA$ has eigenvalues $\mu_j = 0.8, -3 + 4i, -3 - 4i$. It is easy to see that the first eigenvector is $v_1 = \frac{1}{\sqrt{2}}(1, 1, 0)^T$; the other two eigenvectors are orthogonal to v_1. Thus, if we apply $(I + hA)^n$ to the second or third eigenvector, v_2 or v_3, we get

$$|(I + hA)^n v_j| = |-3 \pm 4i|^n |v_j| = 5^n \;\to\; \infty \quad \text{as} \quad n \to \infty, \quad \text{for} \;\; j = 2, 3.$$

Since u_0 contains components in the direction of v_2 and v_3, this means that $|y^{(n)}| \to \infty$ as $n \to \infty$, very much in contrast to the behaviour of the exact solution $u(t) \to 0$ for $t \to \infty$.

So, even when $u_3 \approx 0$ and $u_2 - u_1 \approx 0$ and the perturbations are very small, the instability of the explicit Euler method with time step size $h = 0.1$ will lead to an exponential increase in these perturbations.

Remark 3.1.10. The important message here is that from a point of view of approximation error (or consistency), it would be possible to increase the time step significantly at later times, but due to stability problems with the explicit Euler method we cannot increase h beyond a certain stability threshold.

This phenomenon only arises for monotonic ODEs, or for ODEs that satisfy a one-sided Lipschitz condition with constant $0 < \nu \ll 1$ and that are monotonic for all $t \geq t^*$, for some $t^* \geq t_0$. The consistency error is closely linked to the Lipschitz constant L of f, while the stability is linked to the ratio of L and the constant ν in the one-sided Lipschitz condition. In the following definition, we will only focus on monotonic IVPs.

3.1.11 Definition: Let f be Lipschitz continuous with constant $L > 0$ and one-sided Lipschitz continuous with constant $\nu \in \mathbb{R}$. An initial value problem is called **stiff**, if it has the following characteristic properties:

1. The right hand side of the ODE is monotonic.
2. The time scales on which different solution components are evolving differ a lot, i.e.,
$$L \gg |\nu| \, .$$
3. The time scales which are of interest for the application are much longer than the fastest time scales of the equation, i.e.,
$$e^{\nu T} \gg e^{-LT} \approx 0 \, . \tag{3.9}$$

Remark 3.1.12. Note that for the linear IVP in lemma 3.1.8 and when f is monotonic, we have

$$L := \max_{\substack{j=1,\dots,d\\ t\in\mathbb{R}}} |\lambda_j(t)| \geq \max_{\substack{j=1,\dots,d\\ t\in\mathbb{R}}} |\mathrm{Re}(\lambda_j(t))| \quad \text{and} \quad |\nu| := \min_{\substack{j=1,\dots,d\\ t\in\mathbb{R}}} |\mathrm{Re}(\lambda_j(t))| \,.$$

Remark 3.1.13. Even though we used the term definition, the notion of "stiffness of an IVP" has something vague or even inaccurate about it. In fact that is due to the very nature of the problems and cannot be fixed. Instead we are forced to sharpen our understanding by means of a few examples.

Example 3.1.14. First of all we will have a look at equation (3.8) in example 3.1.9. Studying the eigenvalues of the matrix A, we clearly see that $\nu = -2$ and thus the problem is monotonic. We can also find that the Lipschitz constant is $L = \|A\| \approx 72.5$ so that the second condition holds as well.

According to the discussion of example 3.1.9, the third condition depends on the purpose of the computation. If we want to compute the solution at time $T = 0.01$, we would not denote the problem as stiff. On the other hand, if one is interested on the solution at time $T = 1$, on which the terms containing e^{-40t} are already below typical machine accuracy, the problem is stiff indeed. Here, we have seen that Euler's method requires disproportionately small time steps.

Remark 3.1.15. The definition of stiffness and the discussion of the examples reveal that numerical methods are needed, which are not just convergent for time steps $h \to 0$ but also for fixed step size h, even in the presence of time scales clearly below h. In this case, methods still have to produce solutions with correct limit behavior for $t \to \infty$.

Example 3.1.16. The **implicit Euler method** is defined by the one-step formula

$$y_1 = y_0 + hf(t_1, y_1) \quad \Leftrightarrow \quad y_1 - hf(t_1, y_1) = y_0 \,, \tag{3.10}$$

which in general involves solving a nonlinear system of equations. Applied to our linear example (3.8), we get

$$y^{(n)} = (I - hA)^{-1} y^{(n-1)} \quad \Rightarrow \quad y^{(n)} = (I - hA)^{-n} u_0$$

For all $h > 0$, the real part of the eigenvalues of the matrix $I - hA$ is

$$\mathrm{Re}(\mu_j) = \frac{1}{1+2h}, \frac{1}{1+40h}, \frac{1}{1+40h} \,,$$

which are all strictly less than 1, such that we get

$$|y^{(n)}| \to 0 \quad \text{as} \quad n \to \infty,$$

independently of h. Thus, although the implicit Euler method requires in general the solution of a nonlinear system in each step, it allows for much larger time steps than the explicit Euler method, when applied to a stiff problem.

For a visualization see the programming exercise on the last problem sheet and the appendix.

3.2 A-, B- and L-stability

3.2.1. In this section, we will investigate desirable properties of one-step methods for stiff IVP (3.11). We will first study linear problems of the form

$$u' = Au \qquad u(t_0) = u_0. \tag{3.11}$$

and the related notion of A-stability in detail. From the conditions for stiffness we derive the following problem characteristics:

1. All eigenvalues of the matrix A lie in the left half-plane of the complex plane. With (3.2) all solutions are bounded for $t \to \infty$.
2. There are eigenvalues close to zero and eigenvalues with a large negative real part.
3. We are interested in time spans which make it necessary, that the product $h\lambda$ is allowed to be large, for an arbitrary eigenvalue and an arbitrary time step size.

For this case we now want to derive criteria for the boundedness of the discrete solution for $t \to \infty$. The important part is not to derive an estimate holding for $h \to 0$, but one that holds for any value of $h\lambda$ in the left half-plane of the complex numbers.

3.2.2 Definition: Consider the (general) one-step method

$$y_1 = y_0 + hF_h(t_0, y_0, y_1),$$

applied to the scalar, linear test problem $u'(t) = \lambda u(t)$. Then

$$y_1 = R(h\lambda)u_0, \tag{3.12}$$

and

$$y^{(n)} = R(h\lambda)^n u_0, \tag{3.13}$$

for some function $R : \mathbb{C} \to \mathbb{C}$, which is denoted the **stability function** of the one-step method F_h. The **stability region** of the one-step method is the set

$$S = \big\{z \in \mathbb{C} \big| |R(z)| \le 1\big\}. \tag{3.14}$$

Example 3.2.3 (explicit Euler).

$$\begin{aligned} &y_1 = y_0 + h\lambda y_0 = (1 + h\lambda)y_0 \\ \Rightarrow \quad &R(z) = 1 + z \end{aligned} \tag{3.15}$$

The stability region for the explicit Euler is a circle with radius 1 and centre (-1,0) in the complex plane (see Figure 3.1 left).

Example 3.2.4 (Implicit Euler).

$$\begin{aligned} &y_1 = y_0 + h\lambda y_1 \quad \Leftrightarrow (1 - h\lambda)y_1 = y_0 \\ \Rightarrow \quad &R(z) = \frac{1}{1 - z} \end{aligned} \tag{3.16}$$

The stability region for the implicit Euler is the complement of a circle with radius 1 and centre (1,0) in the complex plane (see Figure 3.1 right).

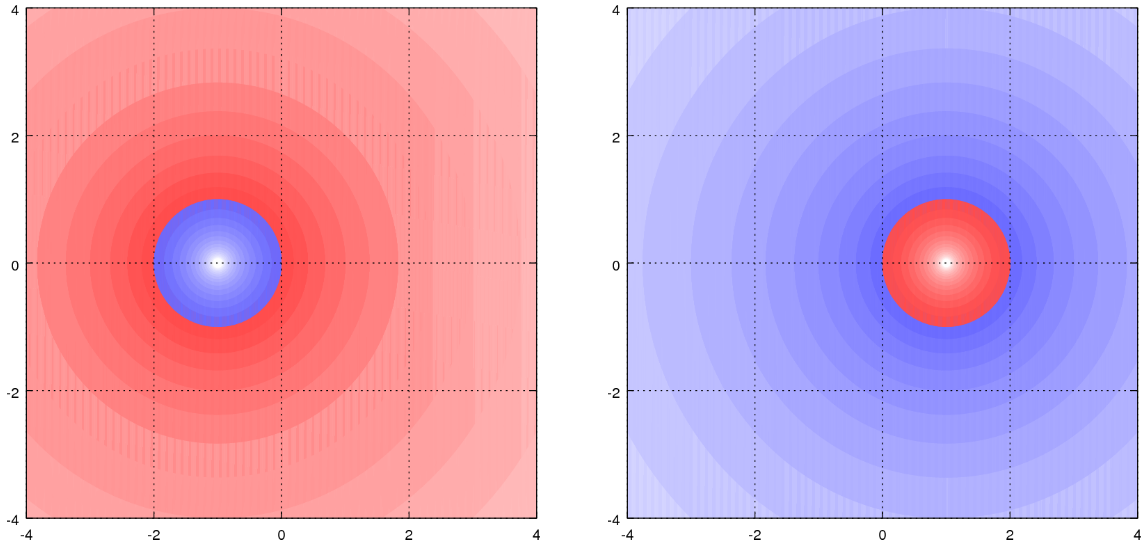


Figure 3.1: Stability regions for explicit and implicit Euler (blue stable, red unstable)

3.2.5 Definition (A-stability): A method is called **A-stable**, if its stability region contains the left half-plane of $\mathbb{C}$. Hence,

$$\{z \in \mathbb{C} \mid \operatorname{Re}(z) \leq 0\} \subset S \tag{3.17}$$

3.2.6 Theorem: Consider the linear, autonomous IVP

$$u' = Au, \qquad u(t_0) = u_0$$

with a diagonalizable matrix A and initial value $y^{(0)} = u_0$. The stability of a one-step method with stability region S applied to this vector-valued problem is inherited from the scalar equation.
In particular, let $\left(y^{(k)}\right)_{k=0}^{\infty}$ be the sequence of approximations, generated by an A-stable one-step method with step size h for this IVP. If all eigenvalues of A have a non-positive real part, then the sequence is uniformly bounded for all h.

Remark 3.2.7. The term "A-stability" was deliberately chosen neutrally by Dahlquist. In particaluar, note that A-stability does **not** stand for asymptotic stability.

Proof (only for ERKs). Since A is diagonalizable, there exists an invertible matrix $V \in \mathbb{C}^{d\times d}$ and a diagonal matrix $\Lambda \in \mathbb{C}^{d\times d}$ such that $A = V^{-1}\Lambda V$. Let $w := Vu$. Then

$$w' = (Vu)' = Vu' = VAu = \Lambda V u = \Lambda w \tag{3.18}$$

and $w(t_0) = Vu(t_0)$, and so the system of ODEs decouples into d independent ODEs

$$w'_\ell = \lambda_\ell w_\ell, \quad w_\ell(t_0) = (w_0)_\ell\,, \quad \ell = 1, \dots, d.$$

Similarly, the stage values of an ERK decouple into d independent, decoupled components:

$$
\begin{aligned}
g_i &= y_0 + h\sum_{j=1}^{s} a_{ij}V^{-1}\Lambda V g_j \quad \Rightarrow \\
\gamma_i &:= V g_i = V y_0 + h\sum_{j=1}^{s} a_{ij}\Lambda V g_j = w_0 + h\sum_{j=1}^{s} a_{ij}\Lambda \gamma_j
\end{aligned}
$$

or equivalently $$(\gamma_i)_\ell = (w_0)_\ell + h\sum_{j=1}^{s} a_{ij}\lambda_\ell(\gamma_j)_\ell\,, \quad \ell = 1,\dots,d.$$

Finally, if we denote by $\eta_j := V y_j$ the transformed numerical solution at the jth time step, we get for the next iterate

$$\eta_1 = V y_1 = V y_0 + h\sum_{i=1}^{s} b_i V g_i = \eta_0 + h\sum_{i=1}^{s} b_i \gamma_i$$

Thus, the ERK applied to a vector valued problem decouples into d decoupled scalar problems solved by the same ERK. But for each of the scalar problems, the definition of A-stability implies boundedness of the solution, if $\mathrm{Re}(\lambda_\ell) \le 0$ for all $\ell = 1,\dots,d$, and thus

$$|y^{(k)}| = |V\eta^{(k)}| \le \|V\||\eta^{(k)}| < \infty.$$

□

3.2.8 Theorem: No explicit Runge-Kutta method is A-stable.

Proof. We show that for such methods $R(z)$ is a polynomial. It is known for polynomials that the absolute value of their value goes to infinity, if the absolute value of the argument goes to infinity. Thus, there exists $z \in \{z \in \mathbb{C} \mid \mathrm{Re}(z) \le 0\}$ such that $|R(z)| > 1$ and thus $z \notin S$ which implies the result of the theorem.

Consider an arbitrary ERK applied to the scalar problem $u' = \lambda u$, $u(t_0) = u_0$. From equation (2.11b) it follows that $k_i = \lambda g_i$, for all $i = 1,\dots,s$. If we insert that into the equation (2.11a), we obtain

$$g_i = y_0 + h\lambda \sum_{j=1}^{i-1} a_{ij} g_j.$$

With $g_1 = y_0$ and $z = h\lambda$ one has

$$
\begin{aligned}
g_2 &= y_0 + a_{21}zy_0 = (1 + a_{21}z)y_0 \\
g_3 &= y_0 + a_{32}zg_2 = y_0 + a_{32}z(1 + a_{21}z)y_0 = (1 + a_{32}z(1 + a_{21}z))y_0.
\end{aligned}
$$

Therefore, one shows easily per induction that g_j is a polynomial of order $j-1$ in z. Substituting into formula (2.11c) it follows that $R(z)$ is a polynomial of order $s-1$. □

Remark 3.2.9. The notion of A-stability is only applicable to linear problems with diagonalizable matrices. Now we are considering its extension to nonlinear problems with monotonic right hand sides.

3.2.10 Definition: A one-step method applied to a monotonic initial value problem $u' = f(t,u)$ with arbitrary initial values y_0 and $\tilde{y}_0$ is called **B-stable** if

$$|y_1 - \tilde{y}_1| \le |y_0 - \tilde{y}_0| \tag{3.19}$$

independent of the time step size h.

3.2.11 Theorem: Let f be monotonic and such that $f(t,0) = 0$, for all $t \in \mathbb{R}$, and consider the IVP

$$u' = f(t,u) \quad \text{with} \quad u(t_0) = u_0\,.$$

Let $\left(y^{(k)}\right)_{k=0}^{\infty}$ be the sequence generated by a B-stable one-step method F_h with initial value $y^{(0)} = u_0$ that satisfies $F_h(t,0) = 0$, for all $t \in \mathbb{R}$. Then the sequence is uniformly bounded for $k \to \infty$ independent of the time step size h.

Proof. The theorem follows immediately by setting $\tilde{y}_0 = 0$ and iterating over the definition of B-stability, since the assumptions of the theorem guarantee that $\tilde{y}_k = 0$, for all k. (Note that $f(t,0) = 0$ implies $F_h(t,0) = 0$ for all Runge-Kutta methods.) □

3.2.12 Corollary: Any B-stable method is A-stable.

Proof. Apply the method to the scalar, linear problem $u' = \lambda u$, which is monotonic for $\mathrm{Re}(\lambda) \le 0$. Now, the definition of B-stability implies $|R(z)| \le 1$, and thus, the method is A-stable. □

An undesirable feature of complex differentiable functions in the context of stability of Runge-Kutta methods is the fact, that $\lim_{z\to\infty} R(z)$ is well-defined on the Riemann sphere, independent of the path chosen to approach this limit in the complex plane. Thus, for any real number x, we have

$$\lim_{x\to\infty} R(x) = \lim_{x\to\infty} R(ix). \tag{3.20}$$

Thus, a method, which has exactly the left half-plane of $\mathbb{C}$ as its stability domain, seemingly a desirable property, has the undesirable property that components in eigenspaces corresponding to very large negative eigenvalues, and thus decaying very fast in the continuous problem, are decaying very slowly if such a method is applied.

This gave rise to the following notion of L-stability. However, note that L-stable methods are not always better than A-stable ones. Similarly, it is also not always necessary to require A-stability. Judgment must be applied according to the problem being solved.

3.2.13 Definition: An A-stable one-step method is called **L-stable**, if

$$\lim_{\mathrm{Re}(z)\to-\infty} |R(z)| = 0. \tag{3.21}$$

Some authors refer to L-stable methods as **strongly A-stable**.

3.3 General Runge-Kutta methods

3.3.1. According to theorem 3.2.8, an explicit Runge-Kutta method cannot be A- or B-stable. Thus, they are not suitable for long term integration of stiff IVPs. The goal of this chapter is the study of methods not suffering from this limitation. The cure will be implicit methods, where stages may not only depend on known values from the past, but also on the value to be computed.

We point out immediately that the main drawback of these methods is the fact that they typically require the solution of nonlinear systems of equations and thus involve much higher computational effort. Therefore, careful judgment should be applied to determine whether a problem is really stiff or whether it is better to use an explicit method.

3.3.2 Definition: A (general) **Runge-Kutta method** is a one-step method of the form

$$g_i = y_0 + h\sum_{j=1}^{s} a_{ij}k_j \qquad i = 1,\dots,s \tag{3.22a}$$

$$k_i = f(t_0 + hc_i, g_i) \qquad i = 1,\dots,s \tag{3.22b}$$

$$y_1 = y_0 + h\sum_{i=1}^{s} b_i k_i \tag{3.22c}$$

where $a_{ij} \neq 0$ for all i, j in general. The method is called

Explicit (ERK) if $a_{ij} = 0$, for all $j \geq i$,

Diagonal Implicit (DIRK) if $a_{ij} = 0$, for all $j > i$,

Singly Diagonal Implicit (SDIRK) if DIRK and $a_{11} = a_{22} = \ldots = a_{ss}$,

Implicit (IRK) in all other cases.

Remark 3.3.3. The corresponding Butcher tableaus are

$$\begin{array}{c|cccc}
0 & a_{11} & a_{12} & \dots & a_{1s} \\
c_2 & a_{21} & a_{22} & \dots & a_{2s} \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
c_s & a_{s1} & \cdots & a_{s,s-1} & a_{s,s} \\
\hline
 & b_1 & \cdots & b_{s-1} & b_s
\end{array}
\qquad
\begin{array}{c|cccc}
0 & a_{11} & & & \\
c_2 & a_{21} & a_{22} & & \\
\vdots & \vdots & \ddots & \ddots & \\
c_s & a_{s1} & \cdots & a_{s,s-1} & a_{s,s} \\
\hline
 & b_1 & \cdots & b_{s-1} & b_s
\end{array}
\qquad
\begin{array}{c|cccc}
0 & a_{11} & & & \\
c_2 & a_{21} & a_{11} & & \\
\vdots & \vdots & \ddots & \ddots & \\
c_s & a_{s1} & \cdots & a_{s,s-1} & a_{11} \\
\hline
 & b_1 & \cdots & b_{s-1} & b_s
\end{array}$$

IRK **DIRK** **SDIRK**

Example 3.3.4 (Two-stage SDIRK)**.** The following two SDIRK methodsare of order three:

$$\begin{array}{c|cc}
\frac12 - \frac{\sqrt3}{6} & \frac12 - \frac{\sqrt3}{6} & 0 \\
\frac12 + \frac{\sqrt3}{6} & \frac{\sqrt3}{3} & \frac12 - \frac{\sqrt3}{6} \\
\hline
 & \frac12 & \frac12
\end{array}
\qquad
\begin{array}{c|cc}
\frac12 + \frac{\sqrt3}{6} & \frac12 + \frac{\sqrt3}{6} & 0 \\
\frac12 - \frac{\sqrt3}{6} & -\frac{\sqrt3}{3} & \frac12 + \frac{\sqrt3}{6} \\
\hline
 & \frac12 & \frac12
\end{array} \tag{3.23}$$

3.3.5 Lemma: Let $\mathbb{I}$ be the $s \times s$ identity matrix and let $e := (1, \dots, 1)^T \in \mathbb{R}^s$. The stability function of a (general) s-stage Runge-Kutta method with coefficients

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1s} \\ \vdots & & \vdots \\ a_{s1} & \cdots & a_{ss} \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_s \end{pmatrix}$$

is given by the two expressions

$$R(z) \;=\; 1 + zb^T\big(\mathbb{I} - zA\big)^{-1}e \;=\; \frac{\det\big(\mathbb{I} - zA + zbe^T\big)}{\det\big(\mathbb{I} - zA\big)} \tag{3.24}$$

Proof. Applying the method to the scalar test problem with $f(u) = \lambda u$, the definition of the stages g_i leads to the system of linear equations

$$g_i = y_0 + h\sum_{j=1}^{s} a_{ij}\lambda g_j, \quad i = 1, \dots, s.$$

In matrix notation, with $z = h\lambda$, we obtain $(\mathbb{I} - zA)g = (y_0, \dots, y_0)^T$, where g is the vector $(g_1, \dots, g_s)^T$. Equally, we obtain

$$\begin{aligned} R(z)y_0 = y_1 &= y_0 + h\sum_{i=1}^{s} b_i\lambda g_i \\ &= y_0 + zb^T g \\ &= y_0 + zb^T(\mathbb{I} - zA)^{-1}\begin{pmatrix} y_0 \\ \vdots \\ y_0 \end{pmatrix} = \big(1 + zb^T(\mathbb{I} - zA)^{-1}e\big)\, y_0. \end{aligned}$$

In order to prove the second representation, we write the whole Runge-Kutta method as a single system of equations of dimension $s + 1$:

$$\begin{pmatrix} \mathbb{I} - zA & 0 \\ -zb^T & 1 \end{pmatrix} \begin{pmatrix} g \\ y_1 \end{pmatrix} = y_0 \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}.$$

Applying Cramer's rule yields the result. $\boxed{\text{DIY}}$ □

3.3.6 Example: Stability functions of the modified Euler method, of the classical Runge-Kutta method of order 4 and of the Dormand-Prince method of order 5 are

$$\begin{aligned} R_2(z) &= 1 + z + \tfrac{z^2}{2} \\ R_4(z) &= 1 + z + \tfrac{z^2}{2} + \tfrac{z^3}{6} + \tfrac{z^4}{24} \\ R_5(z) &= 1 + z + \tfrac{z^2}{2} + \tfrac{z^3}{6} + \tfrac{z^4}{24} + \tfrac{z^5}{120} + \tfrac{z^6}{600} \end{aligned}$$

respectively. $\boxed{\text{DIY}}$ Their stability regions are shown in Figure 3.2.

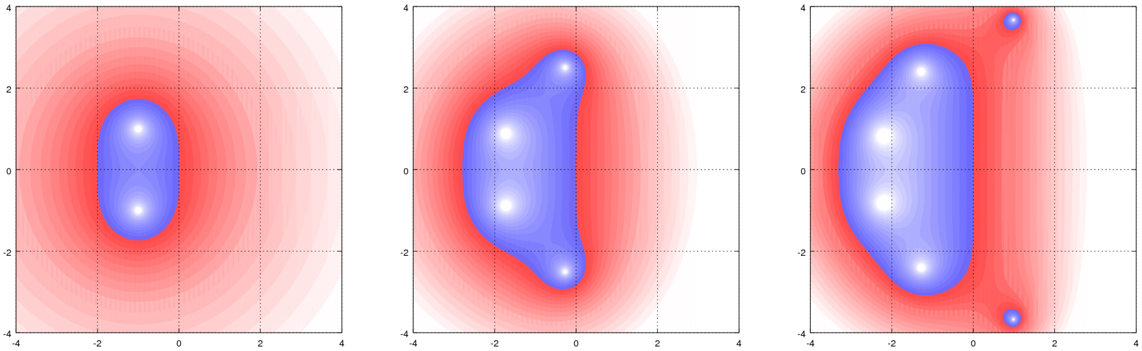


Figure 3.2: Stability regions of the modified Euler method, the classical Runge-Kutta method of order 4 and the Dormand/Prince method of order 5 (blue stable, red unstable)

3.3.7 Definition: The ϑ-scheme is the one-step method, defined for $\vartheta \in [0,1]$ by

$$y_1 = y_0 + h\big((1-\vartheta)f(y_0) + \vartheta f(y_1)\big). \tag{3.25}$$

It is an RKM with the Butcher Tableau

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1-\vartheta & \vartheta \\ \hline & 1-\vartheta & \vartheta \end{array}. \tag{3.26}$$

Three special cases are distinguished:

$$\begin{array}{r|l} \vartheta = 0 & \text{explicit Euler method} \\ \vartheta = 1 & \text{implicit Euler method} \\ \vartheta = 1/2 & \text{Crank-Nicolson method} \end{array}$$

3.3.8 Theorem: The ϑ-scheme is A-stable for $\vartheta \geq 1/2$.

Proof. $\boxed{\text{DIY}}$ (The stability regions for different ϑ are shown in figure 3.3.) □

3.3.1 Existence and uniqueness of discrete solutions

While it was clear that the steps of an explicit Runge-Kutta method can always be executed, implicit methods require the solution of a possibly nonlinear system of equations. The solvability of such a system is not always clear. We will investigate several cases here: First, Lemma 3.3.9 based on a Lipschitz condition on the right hand side. Since this result suffers from a severe step size constraint, we add Lemma 3.3.10 for DIRK methods based on right hand sides with a one-sided Lipschitz condition. Finally, we present Theorem 3.3.11 for general Runge-Kutta methods with one-sided Lipschitz condition.

Recall the definition of the usual maximum row-sum norm of a matrix A:

$$\|A\|_\infty := \max_{i=1,\dots,s} \sum_{j=1}^{s} |a_{ij}| .$$

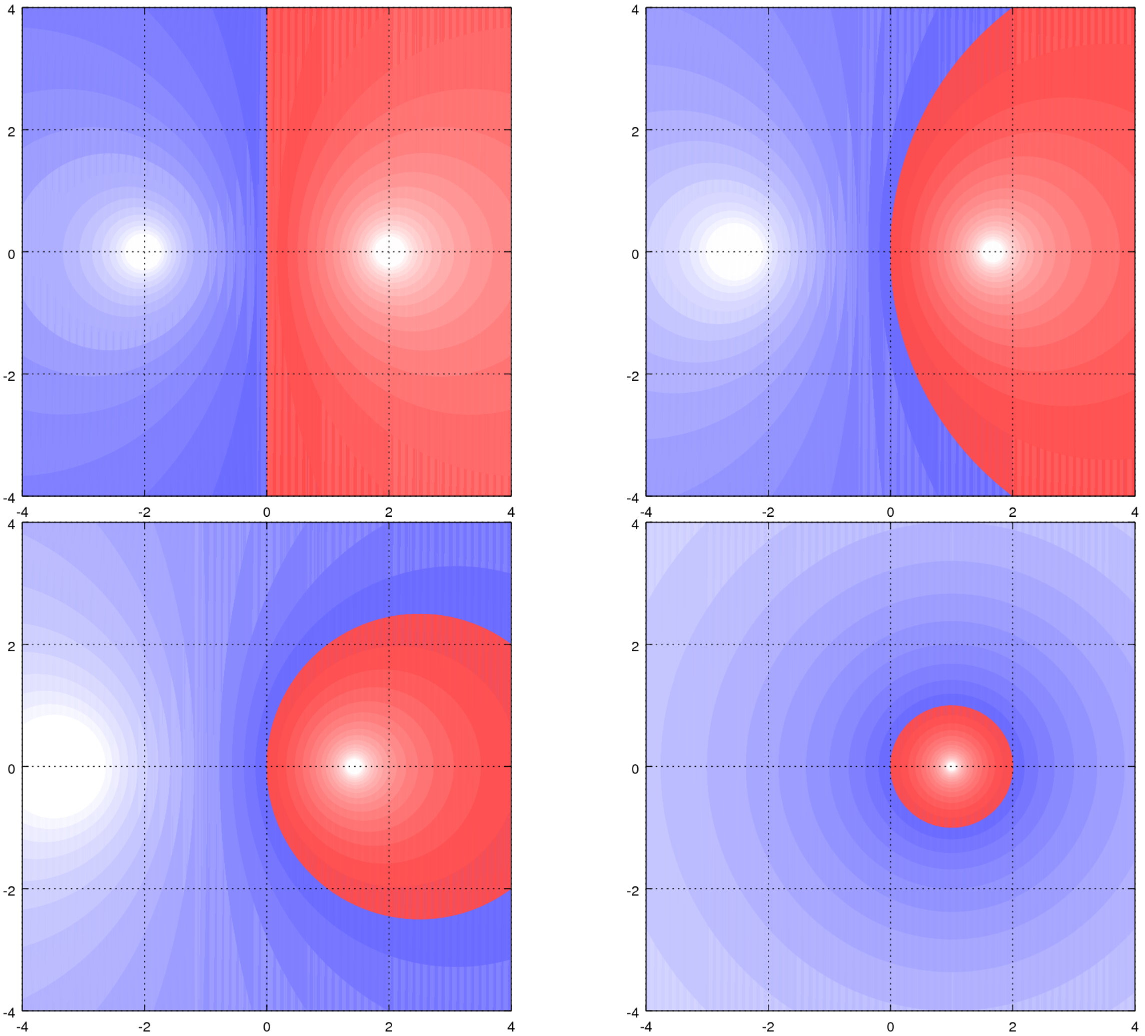


Figure 3.3: Stability regions of the ϑ-scheme with $\vartheta = 0.5$ (Crank-Nicolson), $\vartheta = 0.6$, $\vartheta = 0.7$, and $\vartheta = 1$ (implicit Euler).

3.3.9 Lemma: Let $f : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$ be continuous and satisfy the Lipschitz condition with constant L. If

$$hL\|A\|_\infty < 1 \tag{3.27}$$

then, for any $y_0 \in \mathbb{R}^d$, the Runge-Kutta method (3.22) has a unique solution $y_1 \in \mathbb{R}^d$.

Proof. We prove existence and uniqueness by a fixed-point argument. To this end, given $y_0 \in \mathbb{R}^d$, we define the matrix of stage values $K = [k_1, \dots, k_s] \in \mathbb{R}^{d\times s}$ in (3.22).

Given some initial $K^{(0)} \in \mathbb{R}^{d\times s}$, we consider the fixed-point iteration $K^{(m)} = \Psi(K^{(m-1)})$, $m = 1, 2, \dots$, defined columnwise by

$$k_i^{(m)} = \Psi_i(k^{(m-1)}) = f\Bigg(t_0 + c_i h, y_0 + h\sum_{j=1}^{s} a_{ij}k_j^{(m-1)}\Bigg), \quad i = 1, \dots, s,$$

which clearly has the matrix of stage values K as a fixed point. Using on $\mathbb{R}^{d\times s}$ the norm $\|K\| = \max_{i=1,\dots,s}|k_i|$, where $|.|$ is the regular Euclidean norm on $\mathbb{R}^d$, it follows from the Lipschitz continuity of f in its second argument that

$$\|\Psi(K) - \Psi(K')\| \le \Bigg(hL\max_{i=1,\dots,s}\sum_{j=1}^{s}|a_{ij}|\Bigg)\|K - K'\|.$$

Under assumption (3.27), the term in parentheses is strictly less than one and thus, the mapping Ψ is a contraction. Then the Banach fixed-point theorem (cf. theorem A.3.1) yields the unique existence of y_1 . □

3.3.10 Lemma: Let $f : \mathbb{R} \times \mathbb{C}^d \to \mathbb{C}^d$ be continuous, differentiable in its second argument and satisfy the one-sided Lipschitz condition (3.3) with constant ν. Consider an arbitrary DIRK method with $a_{ii} > 0$. If for all $i = 1, \dots, s$

$$h\nu a_{ii} < 1, \tag{3.28}$$

then, for any $y_0 \in \mathbb{C}^d$, each of the (decoupled) nonlinear equations in (3.22a) has a solution $g_i \in \mathbb{C}^d$.

Proof. The proof simplifies compared to the general case of an IRK, since each stage depends explicitly on the previous stages and implicitly only on itself. Thus, we can write

$$g_i = y_0 + v_i + ha_{ii}f(g_i) \quad \text{with} \quad v_i = h\sum_{j=1}^{i-1} a_{ij}f(g_j). \tag{3.29}$$

For linear IVPs with $f(t, y) := My$ with diagonalizable system matrix M, we have

$$(I - ha_{ii}M)\, g_i = y_0 + v_i\,.$$

Since $\nu = \max_{j=1,\dots,d} \operatorname{Re}(\lambda_j(M))$ (cf. lemma 3.1.8), assumption (3.28) implies that all eigenvalues of $(I - ha_{ii}M)$ have positive real part. Thus, the inverse exists and we obtain a unique solution.

In the nonlinear case, we use a homotopy argument. To this end, we introduce the parameter $\tau \in [0,1]$ and set up the family of equations

$$g(\tau) = y_0 + \tau v_i + ha_{ii} f(g(\tau)) + (\tau - 1) ha_{ii} f(y_0).$$

For $\tau = 0$ this equation has the solution $g(0) = y_0$, and for $\tau = 1$ the solution $g(1) = g_i$. Now, provided g' is bounded on $[0,1]$, we can conclude that a solution exists, since

$$g(1) = g(0) + \int_0^1 g'(s)\,\mathrm{d}s\,. \tag{3.30}$$

To show that g' is bounded, note first that since f was assumed to be differentiable in the second argument

$$\langle f_y(t,y)h + o(|h|), h\rangle = \langle f(t, y+h) - f(x), h\rangle \le \nu |h|^2$$

Dividing by $|h|^2$ and taking the limit as $|h| \to 0$, we obtain with $\widehat{h} = h/|h|$ that

$$\left\langle f_y(t,y)\widehat{h}, \widehat{h}\right\rangle \le \nu \quad \Leftrightarrow \quad \langle f_y(t,y)h, h\rangle \le \nu |h|^2, \quad \text{for all} \quad h \in \mathbb{C}^d.$$

Hence, with

$$g'(\tau) = v_i + ha_{ii} f_y\big(t, g(\tau)\big) g'(\tau) + ha_{ii} f(y_0)$$

we obtain

$$\begin{aligned} |g'(\tau)|^2 &= \big\langle v_i + ha_{ii} f(y_0), g'(\tau)\big\rangle + ha_{ii}\big\langle f_y(t, g(\tau)) g'(\tau), g'(\tau)\big\rangle \\ &\le |v_i + ha_{ii} f(y_0)| |g'(\tau)| + ha_{ii}\nu |g'(\tau)|^2. \end{aligned}$$

Now subtracting the second term on the right hand side and dividing by $1 - ha_{ii}\nu$, which by assumption is positive, it follows that

$$|g'(\tau)|^2 \le \frac{|v_i + ha_{ii} f(y_0)|}{1 - ha_{ii}\nu} |g'(\tau)|,$$

which implies that $g'(\tau)$ is either zero or bounded for all $\tau \in [0,1]$.

Thus, we have proved existence of the stage values g_i. □

If the DIRK method in lemma 3.3.10 is A- or B-stable, then the g_i are unique.

3.3.11 Theorem: Let f be continuously differentiable and let it satisfy the one-sided Lipschitz condition (3.3) with constant ν. If the Runge-Kutta matrix A is invertible and if there exists a diagonal matrix $D = \operatorname{diag}(d_1, \dots, d_s)$ with positive entries, such that

$$h\nu < \frac{\langle x, A^{-1}x\rangle_D}{\langle x, x\rangle_D}, \quad \forall x \in \mathbb{R}^s, \tag{3.31}$$

then the nonlinear system (3.22a) has a solution $(g_1, \dots, g_s)$, where $\langle x, y\rangle_D = \langle Dx, y\rangle$.

Proof. We omit the proof here and refer to [HW10, Theorem IV.14.2] □

3.3.2 Considerations on the implementation of Runge-Kutta methods

3.3.12. As we have seen in the proof of lemma 3.3.9, implicit Runge-Kutta methods require the solution of a nonlinear system of size $s \cdot d$, where s is the number of stages and d the dimension of the system of ODEs. DIRK methods are simpler and only require the solution of systems of dimension d. Thus, we should prefer this class of methods, weren't it for the following theorem.

3.3.13 Theorem: A B-stable DIRK method has at most order 4

Proof. See [HW10, Theorem IV.13.13]. □

Remark 3.3.14. In each step of an IRK, we have to solve a (non-)linear system for the quantities g_i. In order to reduce round-off errors, it is advantageous to solve for $z_i = g_i - y_0$. Especially for small time steps, z_i is expected to be much smaller than g_i. Thus, we have to solve the system

$$z_i = h \sum_{j=1}^{s} a_{ij} f(t_0 + c_j h, y_0 + z_j), \quad i = 1, \dots, s. \tag{3.32}$$

Using the Runge-Kutta matrix A, we rewrite this as

$$\begin{pmatrix} z_1 \\ \vdots \\ z_s \end{pmatrix} = A \begin{pmatrix} hf(t_0 + c_1 h, y_0 + z_1) \\ \vdots \\ hf(t_0 + c_s h, y_0 + z_s) \end{pmatrix}. \tag{3.33}$$

We can avoid further function evaluations by then computing

$$y_1 = y_0 + b^T A^{-1} z, \tag{3.34}$$

which again is numerically much more stable than evaluating f (with a possibly large Lipschitz constant).

3.4 Construction of Runge-Kutta methods via quadrature

We finish our discussion of Runge-Kutta methods by describing a systematic way to construct stable, high-order implicit Runge-Kutta methods.

3.4.1 Definition (Simplifying order conditions):

$$B(p): \quad \sum_{i=1}^{s} b_i c_i^{q-1} = \frac{1}{q} \qquad q = 1, \dots, p \tag{3.35a}$$

$$C(p): \quad \sum_{j=1}^{s} a_{ij} c_j^{q-1} = \frac{c_i^q}{q} \qquad \begin{matrix} q = 1, \dots, p \\ i = 1, \dots, s \end{matrix} \tag{3.35b}$$

$$D(p): \quad \sum_{i=1}^{s} b_i a_{ij} c_i^{q-1} = \frac{b_j}{q}(1 - c_j^q) \qquad \begin{matrix} q = 1, \dots, p \\ j = 1, \dots, s \end{matrix} \tag{3.35c}$$

3.4.2 Theorem: Consider a (general) Runge-Kutta method that satisfies condition $B(p)$ in (3.35a), condition $C(\xi)$ in (3.35b), and condition $D(\eta)$ in (3.35c) with $\xi \geq p/2-1$ and $\eta \geq p-\xi-1$. Then the method has consistency order p.

Proof. For the proof, we refer to [HNW09, Ch. II, Theorem 7.4]. Here, we only observe, that

$$\int_0^1 t^{q-1}\,\mathrm{d}t = \frac{1}{q}, \qquad \int_0^{c_i} t^{q-1}\,\mathrm{d}t = \frac{c_i^q}{q}.$$

If we now insert the function x at the places c_i into the quadrature formula with the quadrature weights b_i, then we obtain (3.35a). Similarly we get (3.35b), if we insert the value t^q/q at the places c_i from the quadrature formula with weights a_{ij} for $j=1,\dots,s$. In both cases we carry this out for all monomials until the desired degree is reached. Due to linearity of the formulas the exactness holds for all polynomials up to that degree. □

3.4.1 Gauss-, Radau-, and Lobatto-quadrature

3.4.3. In this subsection, we review some of the basic facts of quadrature formulas based on orthogonal polynomials (cf. Numerik 0 for details).

3.4.4 Definition: Let $L_n(t)$ be the (shifted) Legendre polynomial of degree n on $[0,1]$, up to scaling. These can be compactly defined by

$$L_n(t) = \frac{d^n}{dt^n} t^n (t-1)^n.$$

A quadrature formula for $\int_0^1 f \mathrm{d}x$ that uses the n roots of L_n as its quadrature points and the integrals of the Lagrange interpolating polynomials at those points as its weights is called a **Gauss rule**.

3.4.5 Lemma: The Gauss quadrature formula $Q_{n-1}^{[a,b]}(f)$ with n points for approximating the integral $\int_a^b f\,\mathrm{d}x$ is exact for polynomials of degree $2n-1$. If $f \in C^{2n}[a,b]$ and $h := b-a$ then

$$\left| Q_{n-1}^{[a,b]}(f) - \int_a^b f\,\mathrm{d}x \right| = \mathcal{O}(h^{2n+1}).$$

Proof. See Numerik 0. (Please note that in Numerik 0 we numbered the quadrature nodes $x_0,\dots,x_{n-1}$ and thus n here is $n-1$ in the notes to Numerik 0.) □

Remark 3.4.6. An important alternative set of quadrature formulae are the Radau and Lobatto formulas.

The **Radau quadrature** formulae are similar to the Gauss rules, but they use one end point of the interval $[0,1]$ and the roots of orthogonal polynomials of degree $n-1$ as their abscissas. We distinguish left and right Radau quadrature formulae, depending on which end is included. **Lobatto quadrature** formulae use both end points and the roots of a polynomial of degree $n-2$. The polynomials are

$$\begin{aligned}
&\text{Radau left} && p_n(t) = \frac{d^{n-1}}{dt^{n-1}}\bigl(t^n(t-1)^{n-1}\bigr), && (3.36)\\
&\text{Radau right} && p_n(t) = \frac{d^{n-1}}{dt^{n-1}}\bigl(t^{n-1}(t-1)^{n}\bigr), && (3.37)\\
&\text{Lobatto} && p_n(t) = \frac{d^{n-2}}{dt^{n-2}}\bigl(t^{n-1}(t-1)^{n-1}\bigr). && (3.38)
\end{aligned}$$

A Radau quadrature formula with n points is exact for polynomials of degree $2n-2$. A Lobatto quadrature formula with n points is exact for polynomials of degree $2n-3$. The quadrature weights of these formulae are positive.

3.4.2 Collocation methods

3.4.7. An alternative to solving IVP in individual points in time, is to develop methods, which first approximate the solution function through a polynomial.

However, as we have seen in Numerik 0, polynomials are not suited though for high-order interpolation over large intervals. Therefore, we apply them again only subintervals in the form of Runge-Kutta methods. The subintervals correspond to the time steps and the quadrature points as the stages.

3.4.8 Definition: The **collocation polynomial** $y(t) \in \mathbb{P}_s$ of an s-stage **collocation method** with pairwise different support points $c_1, \dots, c_s$ is defined uniquely through the $s+1$ conditions:

$$y(t_0) = y_0 \tag{3.39a}$$

$$y'(t_0 + c_i h) = f\bigl(t_0 + c_i h, y(t_0 + c_i h)\bigr) \quad i = 1, \dots, s. \tag{3.39b}$$

The value at the next time step is then defined as

$$y_1 = y(t_0 + h). \tag{3.39c}$$

3.4.9 Lemma: An s-stage collocation method with the points c_1 to c_s defines a Runge-Kutta method, as defined in definition 3.3.2, with the coefficients c_i and

$$a_{ij} = \int_0^{c_i} L_j(t)\,\mathrm{d}t, \qquad b_i = \int_0^1 L_i(t)\,\mathrm{d}t, \tag{3.40}$$

where $L_j(t)$, $j = 1, \dots, s$, are the Lagrange interpolation polynomials associated to the point set $\{c_1, \dots, c_s\}$, i.e.

$$L_j(t) = \prod_{\substack{k=1\\k\neq j}}^{s} \frac{t - c_k}{c_j - c_k}.$$

Proof. The polynomial $y'(t)$ is of degree $s-1$ and thus uniquely defined by the s interpolation conditions in equation (3.39b). Setting $y'(x_0 + c_i h) = f\bigl(t_0 + c_i h, y(t_0 + c_i h)\bigr) = k_i$ we obtain

$$y'(x_0 + th) = \sum_{j=1}^{s} k_j \cdot L_j(t), \tag{3.41}$$

where $L_j(t)$, $j = 1, \dots, s$, are the Lagrange interpolation polynomials. By integration we obtain:

$$g_i = y(x_0 + c_i h) = y_0 + h \int\limits_0^{c_i} y'(x_0 + th)\,\mathrm{d}t = y_0 + h \sum_{j=1}^{s} k_j \int_0^{c_i} L_j(t)\,\mathrm{d}t, \tag{3.42}$$

which, by comparison with (3.22a), defines the coefficients a_{ij}. Integrating from 0 to 1 instead, we obtain the coefficients b_j by comparison with (3.22c). □

3.4.10 Lemma: An implicit s-stage Runge-Kutta method, with pairwise different support points c_i, is a collocation method if and only if simplifying conditions $B(s)$ (3.35a) and $C(s)$ in (3.35b) are satisfied. Thus, an s-stage collocation method is of order (at least) s.

Proof. Consider an s-stage RK method. Condition $B(s)$ leads to a system of s conditions for the s coefficients $b_1, \dots, b_s$. The system matrix is the transpose of the Vandermonde matrix V with entries $V_{i,q} := c_i^{q-1}$ which (for pairwise different c_i) is invertible. Therefore these coefficients are defined uniquely. Similarly, for each $i = 1, \dots, s$, condition $C(s)$ leads to a uniquely solvable system of s conditions for the s coefficients $a_{i,j}$, $j = 1, \dots, s$, with the same system matrix. Thus, all the coefficients are defined uniquely.

On the other hand, (3.35b) yields for $q < s$:

$$\sum_{j=1}^{s} a_{ij} c_j^q = \frac{c_i^{q+1}}{q+1} = \int_0^{c_i} t^q\,\mathrm{d}t.$$

As a consequence of linearity we have

$$\sum_{j=1}^{s} a_{ij} p(c_j) = \int_0^{c_i} p(t)\,\mathrm{d}t, \qquad \forall p \in \mathcal{P}_{s-1}.$$

Applying this to the Lagrange interpolation polynomials $L_j(t)$, we obtain the coefficients of equation (3.40), which were in turn computed from the collocation polynomial, proving the equivalence.

It follows from theorem 3.4.2 that a Runge-Kutta method that satisfies $B(s)$ and $C(s)$ has consistency order (at least) s. □

3.4.11 Theorem: Consider a collocation method with s pairwise different support points c_i and define

$$\pi(t) = \prod_{i=1}^{s}(t - c_i). \tag{3.43}$$

If $\pi(t)$ is orthogonal on $[0,1]$ to all polynomials of degree $r-1$ for $r \leq s$, then the collocation method (3.39) is of consistency order $p = s + r$.

Proof. We have already shown in the proof of Lemma 3.4.10, that for any collocation method with s stages, $B(s)$ and $C(s)$ hold.

The condition on π implies that on the interval $[0,1]$ the quadrature rule is in fact exact for polynomials of degree $s + r - 1$ (cf. Numerik 0 for the case $r = s$), so that we have $B(s+r)$. Therefore, to prove conistency order $p = s + r$ it remains to show $D(r)$.

First, we observe that due to $C(s)$ and $B(s+r)$, for all $p < s$ and $q \leq r$, we have

$$\sum_{j=1}^{s}\left(\sum_{i=1}^{s} b_i a_{ij} c_i^{q-1}\right) c_j^{p-1} = \sum_{i=1}^{s} b_i c_i^{q-1}\frac{c_i^p}{p} = \frac{1}{p}\sum_{i=1}^{s} b_i c_i^{p+q-1} = \frac{1}{p(p+q)}.$$

Furthermore, since $B(s+r)$ we have for the same p and q:

$$\sum_{j=1}^{s} b_j\bigl(1 - c_j^q\bigr)c_j^{p-1} = \sum_{j=1}^{s}\bigl(b_j c_j^{p-1} - b_j c_j^{p+q-1}\bigr) = \frac{1}{p} - \frac{1}{p+q} = \frac{q}{p(p+q)}.$$

Subtracting $\frac{1}{q}$ times the second result from the first we get

$$0 = \frac{1}{p(p+q)} - \frac{1}{p(p+q)} = \sum_{j=1}^{s} c_j^{p-1} \underbrace{\left(\sum_i b_i c_i^{q-1} a_{ij} - \frac{1}{q} b_j\bigl(1 - c_j^q\bigr)\right)}_{:=\xi_j}.$$

This holds for $p = 1, \dots, s$ and thus amounts to a homogeneous, linear system in the variables ξ_j with system matrix V^T. Thus, $\xi_j = 0$ and the theorem holds. □

Corollary 3.4.12. *The consistency order p of an s-stage collocation method satisfies*

$$s \leq p \leq 2s.$$

Proof. The polynomial $\pi(t)$ in (3.43) is of degree s. If $\pi = L_s$, the Legendre polynomial of degree s on $[0,1]$, then π is orthogonal to all polynomials of degree $s-1$ by construction (cf. Numerik 0 for details). Thus, it follows from theorem 3.4.11 that there exists an s-stage collocation method of order $p = 2s$.

On the other hand, we know from Numerik 0 that there exists no quadrature rule such that $B(2s+1)$ is satisfied, otherwise the degree s polynomial $\pi(t)$ would have to be orthogonal to itself. In particular, if we consider the scalar model equation $u' = \lambda u$ with exact solution

$u(t) = e^{\lambda t} = \sum_{j=0}^{\infty} (\lambda t)^j/j!$, the best we can hope for is that the collocation polynomial $y(t)$ matches the first $2s-1$ terms in this infinite sum, such that

$$|u_1 - y_1| = \mathcal{O}(h^{2s}).$$

Hence, it is clear the conistency order of an s-stage collocation method satisfies $p \le 2s$. The lower bound has already been proved in lemma 3.4.10. □

3.4.13 Definition: An s-stage **Gauß-Collocation method** is a collocation method, where the collocation points are the set of s Gauß points in the interval $[0,1]$, namely the roots of the Legendre polynomial of degree s.

3.4.14 Example: (2- and 3-stage Gauss collocation methods)

$$\begin{array}{c|cc}
\frac{3-\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\
\frac{3+\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\
\hline
 & \frac{1}{2} & \frac{1}{2}
\end{array}
\qquad
\begin{array}{c|ccc}
\frac{5-\sqrt{15}}{10} & \frac{5}{36} & \frac{2}{9} - \frac{\sqrt{15}}{15} & \frac{5}{36} - \frac{\sqrt{15}}{30} \\
\frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\
\frac{5+\sqrt{15}}{10} & \frac{5}{36} + \frac{\sqrt{15}}{30} & \frac{2}{9} + \frac{\sqrt{15}}{15} & \frac{5}{36} \\
\hline
 & \frac{5}{18} & \frac{4}{9} & \frac{5}{18}
\end{array}$$

3.4.15 Theorem: The s-stage Gauß-collocation method is consistent of order $2s$ and thus of optimal order.

Proof. Follows immediately from the proof of corollary 3.4.12. □

3.4.16 Theorem: Gauß-collocation methods are B-stable. The stability region of Gauß-collocation is exactly the left half-plane of $\mathbb{C}$.

Proof. Let f be monotonic and let $y(t)$ and $z(t)$ be the collocation polynomials according to (3.39) with respect to initial values y_0 or z_0. Analogous to the proof of theorem 3.1.6 we introduce the auxiliary function $m(t) = |z(t) - y(t)|^2$. In the collocation points $\xi_i = t_0 + c_i h$ we have

$$\begin{aligned}
m'(\xi_i) &= 2\mathrm{Re}\left\langle z'(\xi_i) - y'(\xi_i), z(\xi_i) - y(\xi_i)\right\rangle \\
&= 2\mathrm{Re}\left\langle f(\xi_i, z(\xi_i)) - f(\xi_i, y(\xi_i)), z(\xi_i) - y(\xi_i)\right\rangle \le 0.
\end{aligned} \tag{3.44}$$

Since Gauß quadrature is exact for polynomials of degree $2s-1$ and m' is a polynomial of degree $2s-1$, we have:

$$\begin{aligned}
|z_1 - y_1|^2 = m(t_0 + h) &= m(t_0) + \int_{t_0}^{t_0+h} m'(t)\,\mathrm{d}t \\
&= m_0 + h \sum_{i=1}^{s} b_i m'(\xi_i) \;\le\; m(t_0) = |z_0 - y_0|^2,
\end{aligned}$$

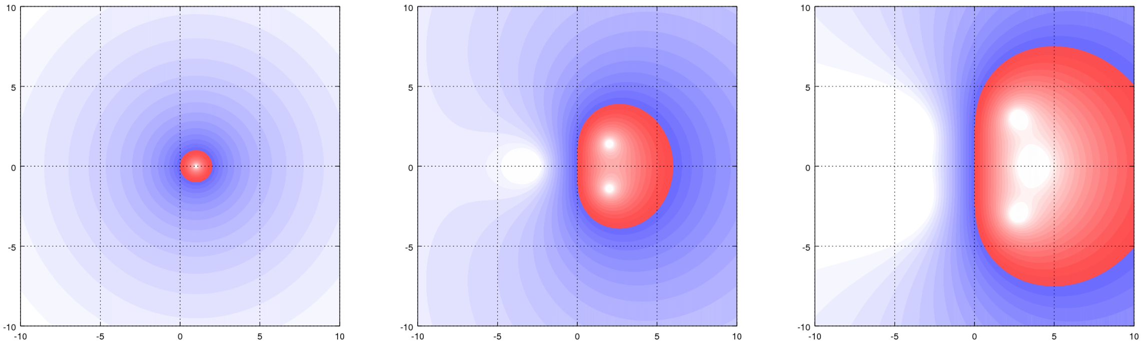


Figure 3.4: Stability domains of right Radau-collocation methods with one (implicit Euler), two, and three collocation points (left to right). Note the different scaling of coordinate axes in comparison with previous figures.

which establishes B-stability.

To show that the stability region is exactly the left half-plane of $\mathbb{C}$ we refer to the problem sheet. $\square$

Remark 3.4.17. Similarly, we can construct collocation rules based on Radau- and Lobatto-quadrature. As in the proof of theorem 3.4.15, it can be shown that the s-stage Radau- and Lobatto-collocation methods are of orders $2s-1$ and $2s-2$, respectively.

Also as in the case of Gauß-quadrature it can be shown that collocation methods based on Radau- and Lobatto quadrature are B-stable (cf. [HW10]). In fact, Radau-collocation methods with right end point of the interval $[0,1]$ included in the quadrature set are L-stable.

The first right Radau collocation method with $s=1$ is simply the implicit Euler method. The definitions of the next two are given in example 3.4.18. The stability regions of the first three are shown in Figure 3.4.

Observe that the stability domains are shrinking with order of the method. Also, observe that the computation of y_1 coincides with that of g_s, such that we can save a few operations.

3.4.18 Example (2- and 3-stage right Radau collocation methods):

$$\begin{array}{c|cc} \frac{1}{3} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{3}{4} & \frac{1}{4} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array} \qquad \begin{array}{c|ccc} \frac{4-\sqrt{6}}{10} & \frac{88-7\sqrt{6}}{360} & \frac{296-169\sqrt{6}}{1800} & \frac{-2+3\sqrt{6}}{225} \\ \frac{4+\sqrt{6}}{10} & \frac{296+169\sqrt{6}}{1800} & \frac{88+7\sqrt{6}}{360} & \frac{-2-3\sqrt{6}}{225} \\ 1 & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \\ \hline & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \end{array}$$

Chapter 4

Newton and quasi-Newton methods

4.1 Basics of nonlinear iterations

4.1.1. The efficient solution of nonlinear problems is an important ingredient to implicit timestepping schemes. Without attempting completeness, we present some important facts about iterative methods for this problem. We introduce the two generic schemes, Newton and gradient methods, discuss their respective pros and cons and combine their features in order to obtain better methods.

Consider the problem of finding $x \in \mathbb{R}^d$ such that

$$f(x) = 0, \qquad \text{for } \; f : \mathbb{R}^d \to \mathbb{R}^d. \tag{4.1}$$

4.1.2 Definition: An iteration

$$x^{(k+1)} = G\left(x^{(k)}\right)$$

to find a fixpoint $x^* = G(x^*)$ is said to be **convergent of order** $p \geq 1$ if

$$\|x^{(k+1)} - x^*\| \leq q \, \|x^{(k)} - x^*\|^p \, .$$

For $p = 1$, in addition we require that $q < 1$. In that case, q is called the **convergence rate**.

We have already seen in the proof of lemma 3.3.9 that the fixpoint iteration, e.g. for the implicit Euler method:

$$y^{(m)} = \Psi(y^{(m-1)}) := y_0 + hf(t_1, y^{(m-1)}), \quad \text{with } \; y^{(0)} = y_0 \, ,$$

converges to y_1 provided $hL < 1$, but the convergence is only of order $p = 1$ (linear) and the convergence rate is $q := Lh$, which may be close to 1. Moreover, it may fail if $hL \geq 1$.

In Numerik 0, we have already seen a faster converging algorithm, the Newton method, and proved there that it converges with order $p = 2$, for sufficiently good initial guess.

4.1.3 Definition: The **Newton method** for finding the root of the nonlinear equation $f(x) = 0$ with $f : \mathbb{R}^d \to \mathbb{R}^d$ reads: given an initial value $x^{(0)} \in \mathbb{R}^d$, compute iterates $x^{(k)} \in \mathbb{R}^d$, $k = 1, 2, \dots$ as follows:

$$\begin{aligned} J &= \nabla f\left(x^{(k)}\right), \\ d^{(k)} &= -J^{-1} f(x^{(k)}), \\ x^{(k+1)} &= x^{(k)} + d^{(k)}. \end{aligned} \tag{4.2}$$

We denote by the term **quasi-Newton method** any modification of this scheme employing an approximation $\widetilde{J}$ of the Jacobian J.

4.1.4 Theorem: Let $U \subset \mathbb{R}^d$ and let $f : U \to \mathbb{R}^d$ be differentiable with

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|, \quad \text{for all} \;\; x, y \in U. \tag{4.3}$$

If there exists a $x^* \in U$ such that $f(x^*) = 0$ and

$$\|(\nabla f(x^*))^{-1}\| \le M, \tag{4.4}$$

then there exists a $0 < R \le \frac{1}{2LM}$ such that for all $x^{(0)} \in \{x \in U : \|x^* - x^{(0)}\| \le R\}$, we have $x^{(k)} \to x^*$ with order $p = 2$.

Remark 4.1.5. The proof of this theorem can be found in the lecture notes for Numerik 0. There are also versions that do not require the existence of the root a priori, such as the Newton-Kantorovich Theorem [Ran17a, Satz 5.5], but we will only discuss some of the main assumptions and features.

The Lipschitz condition on ∇f can be seen as the deviation of f from being linear. Indeed, if f were linear, then $L = 0$ and provided $M \neq 0$ the method converges in a single step for any initial value.

The larger the constant M, the smaller one of the eigenvalues of the Jacobian J. Therefore, the function becomes flat in that direction and the root finding problem becomes unstable.

Most importantly, for an arbitrary initial guess, the method may fail to converge entirely, but close enough to the solution the convergence is very fast, much faster than the fixpoint iteration above.

4.2 Descent methods

Nonlinear root finding of a vector-valued functions $f : \mathbb{R}^d \to \mathbb{R}^d$ – as required in implicit timestepping schemes – is closely related to optimisation of scalar functions $F : \mathbb{R}^d \to \mathbb{R}$, and the following problem is equivalent to (4.5) whenever $f = \nabla F$:

$$x = \operatorname*{arg\,min}_{y \in \mathbb{R}^d} F(y), \qquad \text{for} \;\; F : \mathbb{R}^d \to \mathbb{R}. \tag{4.5}$$

While we assume for most of this discussion that F is known, we will see at the end that the Newton method with line search does not require it.

Obvisously, by choosing $f = \nabla F$, Newton's method also solves the optimsation problem (4.5). An alternative family of methods for (4.5) are the following:

4.2.1 Definition: A **descent method** is an iterative method for finding minimizers of the functional $F : \mathbb{R}^d \to \mathbb{R}$ that, starting from an initial guess $x^{(0)} \in \mathbb{R}^d$, computes iterates $x^{(k)}$, $k = 1, 2 \dots$, by the following steps:

1. If $\nabla F(x^{(k)}) \neq 0$, choose a descent direction

$$s^{(k)} \in \mathbb{R}^d \quad \text{such that} \quad |s^{(k)}| = 1 \text{ and } \left(\nabla F(x^{(k)}), s^{(k)}\right) < 0 \tag{4.6}$$

   and a positive parameter $\alpha^{(k)} > 0$; otherwise terminate.

2. Update: $\quad x^{(k+1)} = x^{(k)} + \alpha^{(k)} s^{(k)}$.

4.2.2 Lemma: Let $F : \mathbb{R}^d \to \mathbb{R}$ be continuously differentiable. For a given point x, assume $\nabla F(x) \neq 0$. Then, there is a constant $\vartheta > 0$ such that for any descent direction d satisfying (4.6) and for any stepsize $0 \leq \alpha \leq \vartheta$ there holds

$$F(x + \alpha s) \leq F(x) - \frac{\vartheta \alpha}{2} |\nabla F(x)|. \tag{4.7}$$

In particular, a positive scaling factor α for the descent method, and thus a strict decrease in the function value, can always be found.

Proof. Skipped. □

The most prominent member of this family of methods is the following.

4.2.3 Definition: The **gradient method** for finding minimizers of $F(x)$ reads: given an initial value $x^{(0)} \in \mathbb{R}^d$, compute iterates $x^{(k)}$, $k = 1, 2, \dots$ by the rule

$$\begin{aligned} d^{(k)} &= -\nabla F(x^{(k)}), \\ \alpha^{(k)} &= \operatorname*{argmin}_{\gamma > 0} F\left(x^{(k)} + \gamma d^{(k)}\right) \\ x^{(k+1)} &= x^{(k)} + \alpha^{(k)} d^{(k)}. \end{aligned} \tag{4.8}$$

It is also called the method of **steepest descent**. The minimization process used to compute α_k, also called **line search**, is one-dimensional and therefore simple. It is sufficient only to find an approximate minimum $\tilde{\alpha}^{(k)}$.

4.2.4 Theorem: Let $F(x) : \mathbb{R}^d \to \mathbb{R}$ be continuously differentiable and let $x^{(0)} \in \mathbb{R}^d$ be chosen such that the set

$$K = \left\{x \in \mathbb{R}^d \middle| F(x) \leq F(x^{(0)})\right\}$$

is compact. Then, each sequence defined by the gradient method has at least one accumulation point and each accumulation point is a stationary point of $F(x)$.

Proof. First, we observe that in any point $x^{(k)}$ with $\nabla F(x^{(k)}) \neq 0$, it follows from lemma 4.2.2 that there exists $\gamma > 0$ such that

$$F(x^{(k)}) > F(x^{(k)} + \gamma d^{(k)}).$$

We conclude, that for such $x^{(k)}$, the line search obtains a positive value of $\alpha^{(k)}$. Thus, the sequence of the gradient iteration is monotonically decreasing and stays within the set K. Since K was assumed to be compact the sequence $x^{(k)}$ has at least one accumulation point x^*. However, the preceding discussion implies that $\nabla F(x^*) = 0$. □

Remark 4.2.5. However, we can also choose $d^{(k)} = -B^{(k)}\nabla F(x^{(k)})$ in (4.8), for any positive definite matrix $B^{(k)}$, leading to **generalised steepest descent methods** that minimise the descent direction in (4.6) in the weighted inner product $(x, y)_{B^{(k)}} = x^T B^{(k)} y$ instead of the Euclidean inner product $(x, y) = x^T y$.

In particular, if the Hessian $D^2F(x^{(k)})$ is positive definite we can choose $B^{(k)} = D^2F(x^{(k)})$ and $\alpha^{(k)} = 1$, which reduces to the Newton method. This link is derived ina different way in the next section.

4.3 Globalization of Newton

4.3.1. The convergence of the Newton method is only local, and it is the faster, the closer to the solution we start. Thus, finding good initial guesses is an important task.

A reasonable initial guess for finding y_1 in a one-step method seems to be y_0, but on closer inspection, this is true only if the time step is small. The convergence requirements of Newton's method would insert a new time step restriction, which we want to avoid. Therefore, we present methods which guarantee global convergence while still converging locally quadratically.

As a rule, Newton's method should never be used without some globalization strategy!

4.3.2 Lemma: Under the assumptions of theorem 4.1.4, Newton's Method applied to the root finding problem $f(x) = 0$ for $f : \mathbb{R}^d \to \mathbb{R}^d$ is a descent method applied to the functional $F(x) = |f(x)|^2$.

Proof. The (multivariate) product rule gives

$$\nabla F(x) = 2f(x)^T \nabla f(x).$$

The search direction of the Newton method applied to $f(x)$ is

$$d^{(k)} = -\bigl(\nabla f(x^{(k)})\bigr)^{-1} f(x^{(k)})$$

Now assume that $f(x^{(k)}) \neq 0$. Then choosing $s^{(k)} := d^{(k)}/|d^{(k)}|$ and (and omitting the arguments $x^{(k)}$), we have

$$\left(\nabla F, s^{(k)}\right) = -\frac{2f^T \nabla f \bigl(\nabla f\bigr)^{-1} f}{|\bigl(\nabla f\bigr)^{-1} f|} \le -\frac{2|f|^2}{\|\bigl(\nabla f\bigr)^{-1}\| \, \|f\|} = -\frac{2|f|}{\|\bigl(\nabla f\bigr)^{-1}\|} < 0,$$

and thus $s^{(k)}$ is a descent direction that satisfies (4.6). Here, we used the fact that for $x^{(k)}$ sufficiently close to x^* we have $\|\bigl(\nabla f(x^{(k)})\bigr)^{-1}\| < \infty$ (Perturbation Theorem, Numerik 0). Finally, choosing $\alpha^{(k)} := |d^{(k)}|$ we have established the equivalence. □

4.3.3 Definition: The **Newton method with line search** for finding the root of the nonlinear equation $f(x) = 0$ reads: given an initial value $x^{(0)} \in \mathbb{R}^d$, compute iterates $x^{(k)} \in \mathbb{R}^d$, $k = 1, 2, \dots$ by the rule

$$\begin{aligned} J &= \nabla f\left(x^{(k)}\right), \\ d^{(k)} &= -J^{-1} f(x^{(k)}), \\ \alpha^{(k)} &= \operatorname*{argmin}_{\gamma>0} \left| f(x^{(k)} + \gamma d^{(k)}) \right|^2 \\ x^{(k+1)} &= x^{(k)} + \alpha^{(k)} d^{(k)}. \end{aligned} \tag{4.9}$$

4.3.4 Definition: A practically most often used variant is the **Newton method with step size control (backtracking line search)**: given an initial value $x^{(0)} \in \mathbb{R}^d$, compute iterates $x^{(k)} \in \mathbb{R}^d$, $k = 1, 2, \dots$ by the rule

$$\begin{aligned} J &= \nabla f\left(x^{(k)}\right), \\ d^{(k)} &= -J^{-1} f(x^{(k)}), \\ x^{(k+1)} &= x^{(k)} + 2^{-j} d^{(k)}. \end{aligned} \tag{4.10}$$

Here, j is the smallest integer number, such that

$$|f(x^{(k)} + 2^{-j} d^{(k)})| < |f(x^{(k)})|. \tag{4.11}$$

Remark 4.3.5. The step size control algorithm can be implemented with very low overhead. In fact, in each Newton step we only have to monitor the norm of the residual $|f(x^{(k)} + d^{(k)})|$, which is typically needed for the stopping criterion anyway. If the residual grows, i.e. $|f(x^{(k)} + d^{(k)})| \ge |f(x^{(k)}|$, we halve the stepsize, recompute the residual norm and check again. A modification to the plain Newton method and additional work are only needed, when the original method was likely to fail anyway.

Under certain assumptions on f it can be shown [NW06] that this backtracking line search algorithm terminates after a finite (typically very small) number of steps. Also, the step size control typically only triggers within the first few steps, then the quadratic convergence of the Newton method starts.

4.4 Practical considerations – quasi-Newton methods

4.4.1. Quadratic convergence is an asymptotic statement, which for any practical purpose can be replaced by "fast" convergence. Most of the effort spent in a single Newton step consists of setting up the Jacobian J and solving the linear system in the second line of (4.2). Therefore, we will consider techniques here, which avoid some of this work. We will have to consider two cases

1. Small systems with $d \lesssim 1000$. For such systems, a direct method like LU- or QR-decomposition is advisable in order to solve the linear system. To this end, we compute the whole Jacobian and compute its decomposition, an effort of order d^3 operations. Comparing to d^2 operations for applying the inverse and order d for all other tasks, this must be avoided as much as possible.

2. Large systems, where the Jacobian is typically sparse (most of its entries are zero). For such a system, factorising the matrix at a cost of order d^3 is typically not affordable. Therefore, the linear problem is solved by an iterative method and we avoid the computation of the Jacobian when possible.

Remark 4.4.2. In order to save numerical effort constructing and inverting Jacobians, the following strategies have been successful.

- Fix a threshold $0 < \eta < 1$ which will be used as a bound for error reduction. In each Newton step, first compute the update vector $\widehat{d}$ using the Jacobian $\widehat{J}$ of the previous step. This yields the modified method

$$\begin{aligned} & J_k = J_{k-1} \\ & \widehat{x} = x^{(k)} - J_k^{-1} f(x^{(k)}) \\ \text{If } |f(\widehat{x})| \leq \eta |f(x^{(k)})| \quad & x^{(k+1)} = \widehat{x} \\ \text{Else } J_k = \big(\nabla f(x^{(k)}\big)^{-1} \quad & x^{(k+1)} = x^{(k)} - J_k^{-1} f(x^{(k)}). \end{aligned} \tag{4.12}$$

  Thus, an old Jacobian and its inverse are used until convergence rates deteriorate. This method is a quasi-Newton method which will not converge quadratically. However, we can obtain linear convergence at any rate η.

- If Newton's method is used within a time stepping scheme, the Jacobian of the last Newton step in the previous time step is often a good approximation for the Jacobian of the first Newton step in the new time step. This holds in particular for small time steps and constant extrapolation. Therefore, the previous method should also be extended over the bounds of time steps.

- An improvement of the method above can be achieved by so called low rank updates, e.g. for the rank-1 update: Let $J_0 = \nabla f(x^{(0)})$ or $J^{(0)} = I$. Then, at the kth step, given $x^{(k)}$ and $x^{(k-1)}$, compute

$$\begin{aligned} p &= x^{(k)} - x^{(k-1)} \\ q &= f(x^{(k)}) - f(x^{(k-1)}) \\ J_k &= J_{k-1} + \frac{1}{|p|^2} \left(q - J_{k-1} p\right) p^T \end{aligned} \tag{4.13}$$

  The fact that the rank of $J_k - J_{k-1}$ is at most one can be used to avoid computing and storing matrices at all. The inverse of such a matrix can be computed via the Sherman-Morrison formula. The practically most efficient and used methods use rank-2 updates, such as the Broyden methods [NW06].

Remark 4.4.3. For problems leading to large, sparse Jacobians, typically space discretizations of partial differential equations, computing inverses of LU-decompositions is infeasible. These matrices typically only feature a few nonzero elements per row, while the inverse and the LU-decomposition is fully populated, thus increasing the amount of memory from d to d^2.

Linear systems like this are often solved by iterative methods, leading for instance to so called Newton-Krylov methods. Iterative methods approximate the solution of a linear system

$$Jd = f$$

only using multiplications of a vector with the matrix J. On the other hand, for any vector $v \in \mathbb{R}^d$, the term Jv denotes the directional derivative of f in direction J. Thus, it can be approximated easily by

$$Jv \approx \frac{f\left(x^{(k)} + \varepsilon v\right) - f\left(x^{(k)}\right)}{\varepsilon}.$$

The term $f\left(x^{(k)}\right)$ must be calculated anyway as it is the current Newton residual. Thus, each step of the iterative linear solver requires one evaluation of the nonlinear function, and no derivatives are computed.

The efficiency of such a method depends on the number of linear iteration steps which is determined by two factors: the gain in accuracy and the contraction speed. It turns out that typically gaining two digits in accuracy is sufficient to ensure fast convergence of the Newton iteration. The contraction number is a more difficult issue and typically requires preconditioning, which is problem-dependent and as such must be discussed when needed.

Chapter 5

Linear Multistep Methods

Instead of using only the *one* initial value at the beginning of the current time interval to the next time step, possibly with the help of intermediate steps, we can also use the values from several previous time steps. Intuitively, this could be more efficient, since function values at these points have been computed already.

Such methods that use values of several time steps in the past in order to achieve a higher order are called **multistep methods**. We will begin this chapter by introducing some of the common formulae, before studying their stability and convergence properties.

5.1 Examples of LMMs

Basically, there are two construction principles for the multistep methods: Quadrature and numerical differentiation.

Example 5.1.1 (Adams-Moulton formulae)**.** Here, the integral from point t_{k-1} to point t_k is approximated by an interpolatory quadrature rule based on the points t_{k-s} to t_k, i.e.,

$$y_k = y_{k-1} + \sum_{r=0}^{s} f_{k-r} \int_{t_{k-1}}^{t_k} L_r(t)\,\mathrm{d}t, \tag{5.1}$$

where f_j denotes the function value $f(t_j, y_j)$ and $L_r(t)$, $r = 0, \dots, s$, the Lagrange interpolation polynomials associated with the points t_{k-r}, $r = 0, \dots, s$.

Since the integral involves the function evaluated at the time step that is being computed, these methods are implicit. Here are the first four in this family:

$$\begin{aligned}
y_k &= y_{k-1} + h f_k && \text{(implicit Euler)}\\
y_k &= y_{k-1} + \frac{1}{2}h\big(f_k + f_{k-1}\big) && \text{(trapezoidal rule)}\\
y_k &= y_{k-1} + \frac{1}{12}h\big(5f_k + 8f_{k-1} - f_{k-2}\big)\\
y_k &= y_{k-1} + \frac{1}{24}h\big(9f_k + 19f_{k-1} - 5f_{k-2} + f_{k-3}\big)
\end{aligned}$$

Example 5.1.2 (Adams-Bashforth formulae)**.** With the same principle we obtain explicit methods by omitting the point in time t_k in the definition of the interpolation polynomial. This yields quadrature formulae of the form

$$y_k = y_{k-1} + \sum_{r=1}^{s} f_{k-r} \int_{t_{k-1}}^{t_k} L_r(t)\,\mathrm{d}t. \tag{5.2}$$

Again, we list the first few:

$$\begin{aligned}
y_k &= y_{k-1} + hf_{k-1} && \text{(explicit Euler)}\\
y_k &= y_{k-1} + \frac{1}{2}h\big(3f_{k-1} - 1f_{k-2}\big)\\
y_k &= y_{k-1} + \frac{1}{12}h\big(23f_{k-1} - 16f_{k-2} + 5f_{k-3}\big)\\
y_k &= y_{k-1} + \frac{1}{24}h\big(55f_{k-1} - 59f_{k-2} + 37f_{k-3} - 9f_{k-4}\big)
\end{aligned}$$

Example 5.1.3 (BDF methods)**.** Backward differencing formulas (BDF) are also based on Lagrange interpolation at the points t_{k-s} to t_k. However, in contrast to Adams formulae they do not use quadrature for the right hand side, but rather the derivative of the interpolation polynomial in the point t_k for the left hand side.

Using the Lagrange interpolation polynomials $L_{k-r}(t)$, we let

$$y(t) = \sum_{r=0}^{s} y_{k-r} L_{k-r}(t),$$

where y_k is yet to be determined. Now we assume that $y(t)$ satisfies the ODE at t_k. Thus,

$$\sum_{r=0}^{s} y_{k-r} L'_{k-r}(t_k) = y'(t_k) = f(t_k, y_k)$$

leading to the following schemes:

$$\begin{aligned}
y_k - y_{k-1} &= hf_k && \text{(implicit Euler)}\\
y_k - \tfrac{4}{3}y_{k-1} + \tfrac{1}{3}y_{k-2} &= \tfrac{2}{3}hf_k\\
y_k - \tfrac{18}{11}y_{k-1} + \tfrac{9}{11}y_{k-2} - \tfrac{2}{11}y_{k-3} &= \tfrac{6}{11}hf_k\\
y_k - \tfrac{48}{25}y_{k-1} + \tfrac{36}{25}y_{k-2} - \tfrac{16}{25}y_{k-3} + \tfrac{3}{25}y_{k-4} &= \tfrac{12}{25}hf_k
\end{aligned}$$

For an example on how to derive these schemes see the appendix.

Remark 5.1.4. Recall from Numerik 0 (or any other introductory course in numerical analysis) that numerical differentiation and extrapolation of interpolation polynomials (i.e. the evaluation outside the interval which is spanned through the interpolation points) are both unstable numerically. Therefore, we expect stability problems for all these methods.

Secondly, recall that Lagrange interpolation with equidistant support points is unstable for higher degree polynomials. Therefore, we also expect all of the above methods to perform well only at moderate order.

5.2 General definition and consistency of LMMs

5.2.1 Definition: A **linear multistep method** (LMM) with s steps is a method of the form

$$\sum_{r=0}^{s} \alpha_{s-r} y_{k-r} = h \sum_{r=0}^{s} \beta_{s-r} f_{k-r}, \tag{5.3}$$

where $f_k = f(t_k, y_k)$ and $t_k = t_0 + hk$, and where we assume $|\alpha_0| + |\beta_0| \neq 0$ and $\alpha_s = 1$. There are explicit ($\beta_s = 0$) and implicit ($\beta_s \neq 0$) methods.
It is convenient to define the **generating polynomials**

$$\varrho(x) = \sum_{r=0}^{s} \alpha_{s-r} x^{s-r} = \sum_{j=0}^{s} \alpha_j x^j \qquad \sigma(x) = \sum_{r=0}^{s} \beta_{s-r} x^{s-r} = \sum_{j=0}^{s} \beta_j x^j. \tag{5.4}$$

for each of these methods.

Remark 5.2.2. The LMM was defined for constant step size h. In principle it is possible to implement the method with a variable step size but we restrict ourselves to the constant case. Notes to the step size control can be found later on in this chapter.

5.2.3 Definition: As for one-step methods, we use the abbreviation $u_k := u(t_k)$, where $u(t)$ denotes the exact solution of $u' = f(t,u),\ u(t_0) = u_0$.
The **local error** of a linear multistep method (LMM) at the kth timestep is again defined by

$$u_k - y_k\,,$$

where y_k is the numerical solution obtained from (5.3) using the exact initial values $y_{k-r} = u_{k-r}$ for $r = 1, ..., s$.
The **truncation error** of an LMM, on the other hand, is defined as

$$\tau_k(u) := h^{-1}(L_h u)(t_k), \tag{5.5}$$

using the linear **difference operator**

$$(L_h u)(t_k) := \sum_{r=0}^{s} \Big(\alpha_{s-r} u_{k-r} - h\beta_{s-r} f\big(t_{k-r}, u_{k-r}\big) \Big). \tag{5.6}$$

Lemma 5.2.4. *For h sufficiently small, the two local errors satisfy the following relation*

$$u_k - y_k = \big(\mathbb{I} - h\beta_s \overline{Df}_k\big)^{-1} (L_h u)(t_k), \tag{5.7}$$

where

$$\overline{Df}_k := \int_0^1 Df\big(t_k, u_k + \vartheta(y_k - u_k)\big)\, d\vartheta.$$

and $Df(t,y)$ is the Jacobian of f with respect to the second argument.

Proof. Sinc we assumed $y_{k-r} = u_{k-r}$, for $r = 1, ..., s$, in the definition of the local error and $\alpha_s = 1$, (5.3) is equivalent to

$$0 = y_k - hf(t_k, y_k) + \sum_{r=1}^{s} \alpha_{s-r} u_{k-r} - h \sum_{r=1}^{s} \beta_{s-r} f(t_{k-r}, u_{k-r}).$$

Subtracting this from (5.6), we obtain

$$(L_h u)(t_k) = (u_k - y_k) - h\beta_s \Big(f(t_k, u_k) - f(t_k, y_k) \Big).$$

Finally, the result follows by applying the Integral Mean Value Theorem (see, e.g., [Numerik 0, Hilfssatz 5.8]) and the fact that for h sufficiently small $\mathbb{I} - h\beta_s \overline{\mathrm{D}f}_k$ is invertible. □

Remark 5.2.5. Note that it follows from lemma 5.2.4 that

$$u_k - y_k = \Big(h + \mathcal{O}(h^2) \Big) \tau_k(u)$$

and that the higher-order term is exactly zero for explicit LMMs.

5.2.6 Definition: An LMM is consistent of order p, if for all sufficiently regular functions f and for all relevant k there holds

$$\tau_k(u) = \mathcal{O}(h^p), \tag{5.8}$$

or equivalently, that the local error is $\mathcal{O}(h^{p+1})$.

5.2.7 Theorem: A LMM with constant step size h is consistent of order p if and only if

$$\sum_{r=0}^{s} \alpha_{s-r} = 0 \quad \text{and} \quad \sum_{r=0}^{s} \left(\alpha_{s-r} r^q + q \beta_{s-r} r^{q-1} \right) = 0, \qquad q = 1, \dots, p \tag{5.9}$$

Proof. We start with the Taylor expansion of the ODE solution u around t_k:

$$u(t) = \sum_{q=0}^{p} \frac{u^{(q)}(t_k)}{q!} (t - t_k)^q + \underbrace{\frac{u^{(p+1)}(\xi)}{(p+1)!} (t - t_k)^{p+1}}_{=:\, R_u(t)},$$

where ξ is a point between t and t_k that depends on t. It follows from $f(t, u) = u'$ that the corresponding right hand side can be expanded as

$$f\big(t, u(t)\big) = \sum_{q=1}^{p} \frac{u^{(q)}(t_k)}{(q-1)!} (t - t_k)^{q-1} + \underbrace{\frac{u^{(p+1)}(\eta)}{p!} (t - t_k)^p}_{=:\, R_f(t)}.$$

with η again a point between t and t_k that depends on t.

Substituting the two expansions into (5.6) we get:

$$\begin{aligned}
L_h u(t_k) =\ & \sum_{r=0}^{s} \alpha_{s-r} \left(\sum_{q=0}^{p} \frac{u^{(q)}(t_k)}{q!} (-rh)^q + R_u(t_{k-r}) \right) - \\
& - \beta_{s-r} h \left(\sum_{q=1}^{p} \frac{u^{(q)}(t_k)}{(q-1)!} (-rh)^{q-1} + R_f(t_{k-r}) \right) \\
=\ & u(t_k) \left(\sum_{r=0}^{s} \alpha_{s-r} \right) + \sum_{q=1}^{p} \frac{u^{(q)}(t_k)}{q!} (-1)^q \left(\sum_{r=0}^{s} \alpha_{s-r} r^q + q\beta_{s-r} r^{q-1} \right) h^q \ +\ Ch^{p+1},
\end{aligned}$$

where

$$C := \sum_{r=0}^{s} \frac{(-1)^{p+1} r^p}{(p+1)!} \Big(\alpha_{s-r} r\, u^{(p+1)}(\xi_r) + (p+1)\beta_{s-r}\, u^{(p+1)}(\eta_r) \Big)$$

and $\xi_r, \eta_r \in [t_{k-r}, t_k]$, for $r = 0, \dots, s$, which in general may all be different.

Since the right hand side f was arbitrary, $L_h u(t_k) = \mathcal{O}(h^{p+1})$ if and only if the conditions in (5.9) hold. In that case we have

$$|L_h u(t_k)| \le \left(\frac{\|u^{(p+1)}\|_\infty}{(p+1)!} \left(\sum_{r=0}^{s} \alpha_{s-r} r^{p+1} + (p+1)\beta_{s-r} r^p \right) \right) h^{p+1}.$$

□

Remark 5.2.8. A consistent LMM is not necessarily convergent. To understand this and to develop criteria for convergence we diverge into the theory of difference equations.

5.3 Properties of difference equations

5.3.1. The stability of LMM can be understood by employing the fairly old theory of difference equations. In order to keep the presentation simple in this section, we use a different notation for numbering indices in the equations. Nevertheless, the coefficients of the characteristic polynomial are the same as for LMM.

5.3.2 Definition: An equation of the form

$$\sum_{r=0}^{s} \alpha_r y_{n+r} = 0 \tag{5.10}$$

is called a homogeneous **difference equation**. Assume that $\alpha_s \alpha_0 \neq 0$ (such that (5.10) does not reduce to a lower order difference equation). A sequence $(y_n)_{n=0,\dots,\infty}$ is solution of the difference equation, if the equation holds true for all $n \ge s$. The values y_n may be from any of the spaces $\mathbb{R}$, $\mathbb{C}$, $\mathbb{R}^d$ or $\mathbb{C}^d$.
The **generating polynomial** of this difference equation is

$$\chi(x) = \sum_{r=0}^{s} \alpha_r x^r. \tag{5.11}$$

5.3.3 Lemma: The solutions of equation (5.10) form a vector space of dimension s.

Proof. Since the equation (5.10) is linear and homogeneous, it is obvious that if two sequences of solutions $(y^{(1)})$ and $(y^{(2)})$ satisfy the equation, then $(\alpha y^{(1)} + y^{(2)})$ also satisfies (5.10), for any $\alpha \in \mathbb{R}$ (or $\mathbb{C}$).

As soon as the initial values y_0 to y_{s-1} are chosen, all other sequence members are uniquely defined. Moreover it holds

$$y_0 = y_1 = \dots = y_{s-1} = 0 \quad \Longrightarrow \quad y_n = 0,\ n \ge 0.$$

Therefore it is sufficient to consider the first s elements. Since those can be chosen arbitrarily, they span an s-simensional vector space. □

5.3.4 Lemma: For each root ξ of the generating polynomial $\chi(x)$ the sequence $y_n = \xi^n$ is a solution of the difference equation (5.10).

Proof. Inserting the solution $y_n = \xi^n$ into the difference equation results in

$$\sum_{r=0}^{s} \alpha_r \xi^{n+r} = \xi^n \sum_{r=0}^{s} \alpha_r \xi^r = \xi^n \chi(\xi) = 0.$$

□

5.3.5 Theorem: Let $\{\xi_j\}_{j=1,\dots,J}$ be the roots of the generating polynomial χ with multiplicity μ_j. Then, the sequences of the form

$$y_n^{(j,k)} = n^{k-1}\xi_j^n \quad j = 1,\dots,J; \quad k = 1,\dots,\mu_j \tag{5.12}$$

form a basis of the solution space of the difference equation (5.10).

Proof. First we observe that the sum of the multiplicities of the roots has to result in the degree of the polynomial:

$$s = \sum_{j=1}^{J} \mu_j\,.$$

Moreover, we know from Lemma 5.3.3, that s is the dimension of the solution space. However, the sequences $(y_n^{(j,k)})$ are also linearly independent. This is clear for sequences of different index j. It is also clear for different roots, because for $n \to \infty$ the exponential function nullifies the influence of the polynomials.

It remains to show that the sequences $(y_n^{(j,k)})$ are in fact solutions of the difference equations. For $k = 0$ we have proven this already in lemma 5.3.4. We proof the fact here for $k = 2$ and for a double zero ξ_j; the principle for higher order roots should be clear then. Equation (5.10) applied to the sequence $(n\xi_j^n)$ results in

$$\begin{aligned}\sum_{r=0}^{s} \alpha_r (n+r)\xi_i^{n+r} &= n\xi_i^n \sum_{r=0}^{s} \alpha_r \xi_i^r + \xi_i^{n+1} \sum_{r=1}^{s} \alpha_r r \xi_i^{r-1} \\ &= n\xi_i^n \chi(\xi_i) + \xi_i^{n+1}\chi'(\xi_i) = 0.\end{aligned}$$

Here the term with α_0 vanishes, because it is multiplied with $r = 0$. $\varrho(\xi_i) = \varrho'(\xi_i) = 0$ because ξ_i is a multiple root. □

5.3.6 Corollary (Root test): All solutions $\{y_n\}$ of the difference equation (5.10) are bounded for $n \to \infty$ if and only if:

- all roots of the generating polynomial $\chi(x)$ lie in $\left\{z \in \mathbb{C} \;\middle|\; |z| \le 1\right\}$ (closed unit circle) and
- all roots on the boundary of the unit circle are simple.

Proof. According to theorem 5.3.5 we can write all solutions as linear combinations of the sequences $(y^{(j,k)})$ in equation (5.12). Therefore, for $n \to \infty$,

1. all solutions with $|\xi_i| < 1$ converge to zero
2. all solutions with $|\xi_i| > 1$ diverge to infinity
3. all solutions with $|\xi_i| = 1$ stay bounded if and only if ξ_i is simple.

This proves the statement of the theorem. □

5.4 Stability and convergence

In contrast to one-step methods, the Lipschitz condition (1.23) for the RHS f of the differential equation is not sufficient to ensure that consistency of a multistep method implies convergence. As for A-stability, stability conditions will again be deduced by means of a simple model problem.

Remark 5.4.1. In the following we investigate the solution to a fixed point in time t with a shrinking step size h. Therefore we choose n steps of step size $h = t/n$ and let n go towards infinity.

5.4.2 Definition: An LMM is **zero-stable** (or simply **stable**) if, applied to the trivial ODE

$$u' = 0 \tag{5.13}$$

with arbitrary initial values $y_0 = u_0$ to $y_{s-1} = u_{s-1}$, it generates solutions y_k which stay bounded at each point in time $t > 0$, if the step size h converges to zero.

5.4.3 Theorem: A LMM is zero-stable if and only if all roots of the first generating polynomial $\varrho(x)$ of equation (5.4) satisfy the root test in corollary 5.3.6.

Proof. The application of the LMM to the ODE (5.13) results in the difference equation

$$\sum_{r=0}^{s} \alpha_{s-r} y_{k-r} = \sum_{r'=0}^{s} \alpha_{r'} y_{n+r'} = 0 \tag{5.14}$$

with $n = k - s$. Thus, the generating polynomial $\varrho(x)$ is equivalent to the generating polynomial $\chi(x)$ of the difference equation in (5.10) which is independent of h.

If $\alpha_s\alpha_0 \neq 0$, the result than follows directly from corollary 5.3.6. Otherwise, (5.14) reduces to a finite difference equation with generating polynomial $\varrho_m(x)$ of order $s-m$, for some $1 \leq m \leq s-1$, and $\varrho(x) = x^m\varrho_m(x)$. Thus, ϱ satisfying the root test is equivalent to ϱ_m satisfying the root test and the result follows again from corollary 5.3.6. □

5.4.4 Corollary: Adams-Bashforth and Adams-Moulton methods are zero-stable.

Proof. For all of these methods the first generating polynomial is $\varrho(x) = x^s - x^{s-1}$. It has the simple root $\xi_1 = 1$ and the $s-1$-fold root 0. □

5.4.5 Theorem: The BDF methods are zero-stable for $s \leq 6$ and not zero-stable for $s \geq 7$.

Proof. See [HNW09, Theorem 3.4]. □

5.4.6 Definition: An LMM is convergent of order p, if for any IVP with sufficiently smooth right hand side f there exists a constant $h_0 > 0$ such that, for all $h \leq h_0$,

$$|u(t_n) - y(t_n)| \leq ch^p, \tag{5.15}$$

whenever the initial values satisfy

$$|u(t_i) - y(t_i)| \leq c_0h^p. \tag{5.16}$$

Here, u is the continuous solution of the IVP and y is the LMM approximation.

To prove convergence, we will for simplicity only consider the scalar case $d = 1$. The case $d > 1$ can be proved similarly.

5.4.7 Lemma: Let $d = 1$. Every LMM can be recast as a one-step method

$$Y_k = AY_{k-1} + hF_h(t_{k-1}, Y_{k-1}) \tag{5.17}$$

where

$$Y_k = \begin{pmatrix} y_k \\ \vdots \\ y_{k-s+1} \end{pmatrix} \in \mathbb{R}^s, \quad A = \begin{pmatrix} -\alpha_{s-1} & -\alpha_{s-2} & \cdots & -\alpha_0 \\ 1 & 0 & \cdots & 0 \\ & \ddots & \cdots & 0 \\ & & 1 & 0 \end{pmatrix} \in \mathbb{R}^{s\times s}, \tag{5.18}$$

and $F_h(t_k, Y_k) = (\psi_k, 0, \dots, 0)^T \in \mathbb{R}^s$ with ψ_k implicitly defined as the solution of

$$\psi_k = \sum_{r=1}^{s} \beta_{s-r} f(t_{k-r}, y_{k-r}) + \beta_s f\left(t_k, h\psi_k - \sum_{r=1}^{s} \alpha_{s-r} y_{k-r}\right). \tag{5.19}$$

Proof. From the general form of LMM we obtain

$$\sum_{r=0}^{s} \alpha_{s-r} y_{k-r} = h \sum_{r=0}^{s-1} \beta_{s-r} f(t_{k-r}, y_{k-r}) + h\beta_s f(t_k, y_k).$$

We rewrite this to

$$y_k = -\sum_{r=1}^{s} \alpha_{s-r} y_{k-r} + h\psi_k,$$

where this formula is also entered implicitly as the value for y_k in the computation of $f(t_k, y_k)$. This is the first equation in (5.17). The remaining equations are simply shifting the entries in the vector Y_{k-1}, i.e. $(Y_k)_{i+1} = (Y_{k-1})_i = y_{k-i}$, for $i = 1, \dots, s-1$. □

5.4.8 Lemma: Let $d = 1$ and let $u(t)$ be the exact solution of the IVP. Suppose $\widehat{Y}_k$ is the solution of a single step

$$\widehat{Y}_k = AU_{k-1} + hF_h(t_{k-1}U_{k-1}),$$

with correct initial values $U_{k-1} = (u_{k-1}, u_{k-2}, \dots, u_{k-s})^T$.
If the multistep method is consistent of order p and f is sufficiently smooth, then there exist constants $h_0 > 0$ and M such that for $h \le h_0$ there holds

$$|U_k - \widehat{Y}_k| \le Mh^{p+1}. \tag{5.20}$$

Proof. The first component of $U_k - \widehat{Y}_k$ is the local error $u_k - y_k$ of step k, as defined in definition 5.2.3, which is of order h^{p+1} by the assumption. The other components vanish by the definition of the method. □

5.4.9 Lemma: Assume that an LMM is zero-stable. Then, there exists a vector norm $\|\cdot\|$ on $\mathbb{C}^s$ such that the induced operator norm of the matrix A satisfies

$$\|A\| \le 1. \tag{5.21}$$

Proof. We notice that $\varrho(x) = \sum \alpha_{s-r} x^r$ is the characteristic polynomial of the matrix A.

By the root test we know that simple roots, which correspond to irreducible blocks of dimension one have maximal modulus one. Furthermore, every Jordan block of dimension greater than one corresponds to a multiple root, which by assumption has modulus strictly less than one. Let ξ_i be such a multiple root with multiplicity μ_i. Such a block admits a modified canonical form

$$J_i = \begin{pmatrix} \lambda_i & 1-|\lambda_i| & & \\ & \lambda_i & \ddots & \\ & & \ddots & 1-|\lambda_i| \\ & & & \lambda_i \end{pmatrix} \in \mathbb{C}^{\mu_i \times \mu_i}.$$

Thus, the canonical form $J = T^{-1}AT$ has norm $\|J\|_\infty \leq 1$. If we define the vector norm

$$\|x\| = \|T^{-1}x\|_\infty,$$

it follows that

$$\|Ax\| = \|T^{-1}Ax\|_\infty = \|JT^{-1}x\|_\infty \leq \|T^{-1}x\|_\infty = \|x\|.$$

□

5.4.10 Theorem: Let f be sufficiently smooth. If a linear multi-step method is zero-stable and consistent of order p, then it is convergent of order p.

Proof. As already stated, we only prove the case $d = 1$ explicitly. See the original notes by Guido Kanschat for the general proof. Since f was assumed to be sufficiently smooth, F_h satsifies a uniform Lipschitz condition with Lipschitz constant L_h.

We reduce the proof to the convergence of the one-step method

$$Y_k = AY_{k-1} + hF_h(t_{k-1}, Y_{k-1}) =: G(Y_{k-1}). \tag{5.22}$$

Let Y_{k-1} and Z_{k-1} be two initial values for the interval I_k. By the previous lemma, we have in the norm defined there, for sufficiently small h, that

$$\|G(Y_{k-1}) - G(Z_{k-1})\| \leq (1 + hL_h)\|Y_{k-1} - Z_{k-1}\|. \tag{5.23}$$

By lemma 5.4.8, the local error $\eta_k = \|U_k - \widehat{Y}_k\|$ at step k is bounded by Mh^{p+1} (where M also contains the equivalence constant γ between the Euclidean norm and the norm defined in the previous lemma). Thus:

$$\begin{aligned}
\|U_1 - Y_1\| &\leq (1 + hL_h)\|U_0 - Y_0\| + Mh^{p+1} \\
\|U_2 - Y_2\| &\leq (1 + hL_h)^2\|U_0 - Y_0\| + Mh^{p+1}\big(1 + (1 + hL_h)\big) \\
\|U_3 - Y_3\| &\leq (1 + hL_h)^3\|U_0 - Y_0\| + Mh^{p+1}\Big(\big(1 + (1 + hL_h) + (1 + hL_h)^2\big)\Big) \\
&\ \ \vdots \\
\|U_n - Y_n\| &\leq (1 + hL_h)^n\|U_0 - Y_0\| + Mh^{p+1}\Big(\big(1 + (1 + hL_h) + \ldots + (1 + hL_h)^n\big)\Big) \\
&\leq e^{nhL_h}\|U_0 - Y_0\| + \frac{Mh^p}{L_h}\big(e^{nhL_h} - 1\big) \leq Ch^p
\end{aligned}$$

where we recall that $U_n = u(t_n)$ and $t_n = t_0 + T$ where $T = nh$ and where

$$C := c_0\gamma e^{TL_h} + \frac{M}{L_h}(e^{TL_h} - 1)$$

with c_0 as defined in (5.16). □

5.5 Starting procedures

5.5.1. In contrast to one-step methods, where the numerical solution is obtained solely from the differential equation and the initial value, multistep methods require more than one start value. An LMM with s steps requires s known start values $y_{k-s}, \dots, y_{k-1}$. Mostly, they are not provided by the IVP itself. Thus, general LMM decompose into two parts:

- a *starting phase* where the start values are computed in a suitable way and
- a *run phase* where the LMM is executed.

It is crucial that the starting procedure provides a suitable order corresponding to the LMM of the run phase, recall condition (5.16) in definition 5.4.6. Possible choices for the starting phase include multistep methods with variable order and one-step methods.

Example 5.5.2 (Self starter)**.** A 2-step BDF method requires y_0 and y_1 to be known. y_0 is given by the initial value while y_1 is unknown so far. To guarantee that the method has order 2, y_1 needs to be at least locally of order 2, i.e.,

$$|u(t_1) - y_1| \le c_0 h^2. \tag{5.24}$$

This is ensured, for example, by one step of the 1-step BDF method (implicit Euler).

However, starting an LMM with $s > 2$ steps by a first-order method and then successively increasing the order until s is reached does not provide the desired global order. That is due to the fact that a one-step method cannot have more than order 2, limiting the overall convergence order to 2. Nevertheless, self starters are often used in practice.

Example 5.5.3 (Runge-Kutta starter)**.** One can use Runge-Kutta methods to start LMMs. Since only a fixed number of starting steps are performed, the local order of the Runge-Kutta approximation is crucial. For an implicit LMM with convergence order p and stepsize h one could use an RK method with consistency order $p-1$ with the same step size h.

Consider a 3-step BDF method. Thus, apart from y_0, we need start values y_1, y_2 with errors less than $c_0 h^3$. They can be computed by RK methods of consistency order 2, for example by two steps of the 1-stage Gauß collocation method with step size h since it has consistency order $2s = 2$, see theorem 3.4.15.

Remark 5.5.4. In practice not the order of a procedure is crucial but rather the fact that the errors of all approximations (the start values and all approximations of the run phase) are bounded by the user-given tolerance, compare Section 2.4. Generally, LMMs are applied with variable step sizes and orders in practice (see e.g. Exercise 7.2).

Thus, the step sizes of all steps are in practice controlled usually controlled using local error estimates. Hence, self starting procedures usually start with very small step sizes and increase them successively. Due to their higher orders RK starters usually are allowed to use moderate step sizes in the beginning.

5.6 LMM and stiff problems

To study A-stability of LMMs we consider again the model equation $u' = \lambda u$. Applying a general LMM (5.3) to this model equation leads to the linear model difference equation

$$\sum_{r=0}^{s} (\alpha_{s-r} - z\beta_{s-r}) y_{k-r} = 0. \tag{5.25}$$

with $z = \lambda h$.

5.6.1 Definition (A-stability of LMM): The **stability region** of an LMM is the set of points $z \in \mathbb{C}$, for which all sequences $(y_n)_{n=0}^{\infty}$ of solutions of the equation (5.25) stay bounded for $n \to \infty$. An LMM is called **A-stable**, if the stability region contains the left half-plane of $\mathbb{C}$.

Note that this definition is equivalent to the definition of A-stability for one-step methods in definition 3.2.5.

5.6.2 Definition: The so-called **stability polynomial** of an LMM is obtained by replacing λh in (5.25) by a general element $z \in \mathbb{C}$ and by inserting $y_n = x^n$ to obtain

$$R_z(x) = \sum_{r=0}^{s} (\alpha_{s-r} - z\beta_{s-r}) x^{s-r}. \tag{5.26}$$

Remark 5.6.3. Instead of the simple amplification function $R(z)$ of the one-step methods, we get here a function of two variables: the point z for which we want to show stability and the artificial variable x from the analysis of the method.

5.6.4 Lemma: Let $\{\xi_1(z), \dots, \xi_s(z)\}$ be the set of roots of the stability polynomial $R_z(x)$ as functions of z. A point $z \in \mathbb{C}$ is in the stability region of a LMM, if these roots satisfy the root test in corollary 5.3.6.

Proof. The proof is analog to that of theorem 5.4.3. □

5.6.5 Theorem (2nd Dahlquist barrier): There is no A-stable LMM of order $p > 2$. Among the A-stable LMM of order 2, the trapezoidal rule (Crank-Nicolson) has the smallest error constant.

Proof. See [HW10, Theorem V.1.4]. □

5.6.1 A(α)-stability

5.6.6. Motivated by the fact that there are no higher order A-stable LMMs people have introduced relaxed concepts of A-stability.

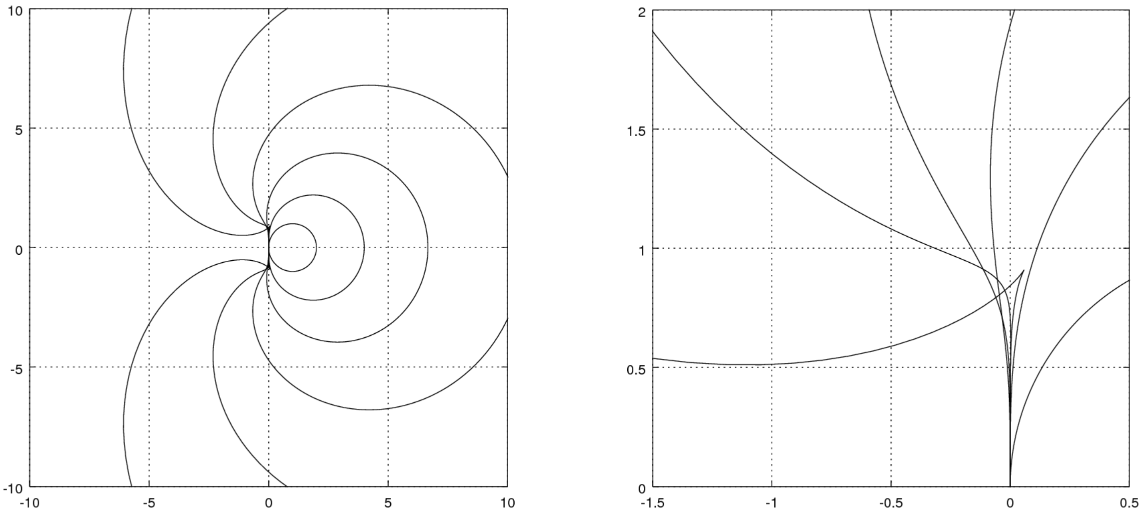


Figure 5.1: Boundaries of the stability regions for BDF(1) to BDF(6); the unstable region is right of the origin. The right figure shows a zoom near the origin.

k	1	2	3	4	5	6
α	90°	90°	86.03°	73.35°	51.84°	17.84°
D	0	0	0.083	0.667	2.327	6.075

Table 5.1: Values for A(α)- and stiff stability for BDF methods of order k.

5.6.7 Definition: A LMM is called **A(α)-stable**, for $\alpha \in [0°, 90°]$, if its stability region contains the sector

$$\left\{ z \in \mathbb{C} \;\middle|\; \mathrm{Re}(z) < 0 \;\text{ and }\; \left|\frac{\mathrm{Im}(z)}{\mathrm{Re}(z)}\right| \le \tan\alpha \right\}.$$

It is called **A(0)-stable**, if the stability region contains the negative real axis.

It is called **stiffly stable**, if it contains the set $\{\mathrm{Re}(z) < -D\}$.

Remark 5.6.8. The introduction of A(0)-stability is motivated by linear systems of the form $u' = -Au$ with symmetric positive definite matrix A. Only stability on the real axis is required in that case, since all eigenvalues are real. Any positive angle α is sufficient.

Similarly, A(α)-stable LMM are suitable for linear problems in which high-frequency vibrations (large Imλ) decay fast (large $-$Reλ).

LMMs behave similarly for nonlinear problems if the Jacobian matrix $\mathrm{D}_y f$ satisfies corresponding properties.

Example 5.6.9. The stability regions of the stable BDF methods are in Figure 5.1. The corresponding values for A(α)-stability and stiff stability are in Table 5.1. (Recall from theorem 5.4.5 that BDF(7) is not even zero-stable.)

Chapter 6

Boundary Value Problems

This chapter deals with problems of a fundamentally different type than the problems we examined in chapter 1, namely boundary value problems. Here, we have prescribed values at the beginning and at the end of an interval of interest. They will require the design of different numerical methods. We will only consider the most classical one.

6.1 General boundary value problems

Due to lemma 1.2.4 we know that every ODE can be written as a system of first-order ODEs. Thus, we make the following definition (restricting our attention to explicit ODEs).

6.1.1 Definition: A **boundary value problem** (BVP) is a differential equation problem of the form: Find $u : [a, b] \subset \mathbb{R} \to \mathbb{R}^d$, such that

$$u'(t) = f\bigl(t, u(t)\bigr) \qquad t \in (a, b) \tag{6.1a}$$

$$r\bigl(u(a), u(b)\bigr) = 0. \tag{6.1b}$$

6.1.2 Definition: A BVP (6.1) is called linear, if the right hand side f as well as the boundary conditions are linear in u, i.e.: Find $u : [a, b] \to \mathbb{R}^d$ such that

$$u'(t) = A(t)u(t) + c(t) \qquad \forall t \in (a, b) \tag{6.2a}$$

$$B_a u(a) + B_b u(b) = g. \tag{6.2b}$$

with $A : [a, b] \to \mathbb{R}^{d\times d},\ c : [a, b] \to \mathbb{R}^d,\ B_a, B_b \in \mathbb{R}^{d\times d}$ and $g \in \mathbb{R}^d$.

Remark 6.1.3. Since boundary values are imposed at two different points in time, the concept of local solutions from Definition 1.2.8 is not applicable. Thus, tricks, such as going forward from interval to interval, as is done in the proof of Péano's theorem using Euler's method, are here not applicable. For this reason, nothing can be concluded from the local properties of the solution and that right hand side f. In fact, it is hard in general even to establish that a solution exists.

Example 6.1.4. Consider the linear BVP

$$\begin{bmatrix} u_1'(t) \\ u_2'(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \text{with}$$

(i) $u_1(0) = 0,\ u_2(1) = 0,$

(ii) $u_1(0) = 0,\ u_1(1) = 0,$

(iii) $u_2(0) = 0,\ u_2(1) = 0.$

By substitution, we can easily see that this first-order system of ODEs is in fact equivalent to the second-order ODE $u_1'' = 1$, which can be explicitly solved by integration to give

$$u_2(t) = u_1'(t) = t + c_1 \quad \text{and} \quad u_1(t) = \frac{t^2}{2} + c_1 t + c_2.$$

But the BVP is **not** solvable for all three choices of boundary conditions (BCs). Using the BCs we get

(i) $c_2 = 0$ and $c_1 = -1$, i.e., $u(t) = \big(t(\frac{t}{2} - 1), t - 1\big)^T$,

(ii) $c_2 = 0$ and $c_1 = -1/2$, i.e., $u(t) = \big(\frac{t}{2}(t - 1), t - \frac{1}{2}\big)^T$,

(iii) but here the two BCs lead to $c_1 = 0$ and $c_1 = -1$, respectively, which cannot be satisfied simultaneously.

Remark 6.1.5. Note that due to lemma 1.3.13 we know that the solution space of the linear ODE in (6.2a) is d-dimensional. Hence, we need d additional pieces of information to determine the solution uniquely. However, whether the d boundary conditions in (6.2b) are sufficient is more subtle than in the case of an IVP, as we have just seen.

We will not discuss this further and instead consider an important subclass of linear BVPs, as well as the most classical numerical method for them. For more details on the general solution theory, see chapter 6 in the original notes by G. Kanschat or [Ran17b, Chap. 8].

6.2 Second-order, scalar two-point boundary value problems

Let us consider following linear, second-order BVP of finding $u : [a, b] \to \mathbb{R}$ such that

$$-u''(x) + \beta(x)u'(x) + \gamma(x)u(x) = f(x), \qquad u(a) = u_a, \quad u(b) = u_b. \tag{6.3}$$

for some functions $\beta, \gamma, f : [a, b] \to \mathbb{R}$ and two boundary values $u_a, u_b \in \mathbb{R}$. (As is common practice, we use x instead of t to denote the independent variable here.)

We introduce the set

$$\mathcal{B} = \Big\{ u \in C^2(a, b) \cap C[a, b] \ \Big|\ u(a) = u_a \text{ and } u(b) = u_b \Big\}.$$

Then, we can see the LHS of (6.3) as a differential operator applied to u, mapping $\mathcal{B}$ to the set of continuous functions. Namely, we define

$$\begin{aligned} L : \mathcal{B} &\to C[a, b] \\ u &\mapsto -u'' + \beta u' + \gamma u. \end{aligned} \tag{6.4}$$

To simplify our life we can (without loss of generality) get rid of the inhomogeneous boundary values u_a and u_b. To this end, let

$$\psi(x) = \frac{b-x}{b-a}\,u_a + \frac{x-a}{b-a}\,u_b\,,$$

and introduce the new function $u_0 := u - \psi$. Then, u_0 solves the BVP

$$\begin{aligned} -u_0''(x) + \beta(x)u_0'(x) + \gamma(x)u_0(x) &= \underbrace{f(x) - \beta(x)\frac{u_b-u_a}{b-a} - \gamma(x)\psi(x)}_{=:\, f_0(x)}, \\ u_0(a) = u_0(b) &= 0. \end{aligned}$$

Thus, it is sufficient to consider the boundary value problem:

6.2.1 Definition: Given an interval $I = [a,b]$, find a function

$$u \in V = \Big\{ u \in C^2(a,b) \cap C[a,b] \;\Big|\; u(a) = u(b) = 0 \Big\}, \tag{6.5}$$

such that for a differential operator of second order as defined in (6.4) above and a right hand side $f \in C[a,b]$ there holds

$$Lu = f. \tag{6.6}$$

6.3 Finite difference methods

We subdivide the interval $I = [a,b]$ again into subintervals and, as in Definition 2.1.2, and consider the solution only at the partitioning points $a = x_0 \le x_1 \ldots \le x_n = b$. As with one-step and multistep timestepping methods, we denote the approximate solution values at those partitioning points by y_k, $k = 0,\ldots,n$.

While one-step methods directly discretize the Volterra integral equation in order to compute the solution at every new step, **finite difference methods** discretize the differential equation on the whole interval at once and then solve the resulting discrete (finite-dimensional) system of equations. We have accomplished the first step and decided that instead of function values $u(x)$ in every point t of the interval I, we only approximate $u(x_k)$ in the points of the partition by y_k, $k = 0,\ldots,n$. What is left is the definition of the discrete operator representing the equation.

6.3.1 Definition (Finite differences): To approximate **first** derivatives of a function u, we introduce the operators

$$\begin{aligned} &\text{Forward difference} && D_h^+u(x) = \frac{u(x+h)-u(x)}{h}, && (6.7)\\ &\text{Backward difference} && D_h^-u(x) = \frac{u(x)-u(x-h)}{h}, && (6.8)\\ &\text{Central difference} && D_h^cu(x) = \frac{u(x+h)-u(x-h)}{2h}. && (6.9) \end{aligned}$$

For **second** derivatives we introduce the

$$\text{3-point stencil} \qquad D_h^2u(x) = \frac{u(x+h)-2u(x)+u(x-h)}{h^2}. \tag{6.10}$$

Remark 6.3.2. Note that the 3-point stencil is the product of the forward and backward difference operators:

$$D_h^2 u(x) = D_h^+\Big(D_h^- u(x)\Big) = D_h^-\Big(D_h^+ u(x)\Big).$$

For simplicity, we only present finite differences of uniform subdivisions. Nevertheless, the definition of the operators can be extended easily to h changing between intervals.

6.3.3 Definition: A finite difference operator D_h^α is consistent of order p with the αth derivative, if there exists a constant $c > 0$ independent of h and a subset $\tilde{I} \subset [a,b]$, such that for any $u \in C^{\alpha+p}(a,b)$ and for any $x \in \tilde{I}$:

$$|D_h^\alpha u(x) - u^{(\alpha)}(x)| \le ch^p \tag{6.11}$$

6.3.4 Lemma: The forward and backward difference operators D_h^+ and D_h^- in definition 6.3.1 are consistent of order 1 with the first derivative, i.e. for any $x \in [a, b-h]$ (resp. $x \in [a+h, b]$),

$$|D_h^+ u(x) - u'(x)| \le ch \quad \text{and} \quad |D_h^- u(x) - u'(x)| \le ch. \tag{6.12}$$

The central difference operator D_h^c and the 3-point stencil D_h^2 are consistent of order 2 with the first and second derivative, respectively, i.e. for any $x \in [a+h, b-h]$,

$$|D_h^c u(x) - u'(x)| \le ch^2 \quad \text{and} \quad |D_h^2 u(x) - u''(x)| \le ch^2. \tag{6.13}$$

Proof. Taylor expansion: Let $x \in [a, b-h]$. Then there exists $\xi \in (x, x+h)$ such that

$$\begin{aligned} D_h^+ u(x) - u'(x) &= \frac{u(x+h) - u(x)}{h} - u'(x) \\ &= \frac{u(x) + hu'(x) + \frac{h^2}{2}u''(\xi) - u(x)}{h} - u'(x) \;=\; \tfrac{h}{2}u''(\xi). \end{aligned}$$

Thus, the result for D_h^+ holds with $c := \frac{1}{2}\max_{x\in(a,b)}|u''(x)|$. The same computation can be applied to $D_h^- u(x)$.

For D_h^c, let $x \in [a+h, b-h]$. Then, there exist $\xi^-, \xi^+ \in (x-h, x+h)$ such that

$$\begin{aligned} &D_h^c u(x) - u'(x) = \\ &\quad \frac{u(x) + hu'(x) + \frac{h^2}{2}u''(x) + \frac{h^3}{6}u'''(\xi^+) - \big(u(x) - hu'(x) + \frac{h^2}{2}u''(x) - \frac{h^3}{6}u'''(\xi^-)\big)}{2h} - u'(x) \\ &\qquad = \tfrac{h^2}{12}\left(u'''(\xi^-) + u'''(\xi^+)\right). \end{aligned}$$

The final result for the 3-point stencil D_h^2 follows in a similar way $\boxed{\text{DIY}}$. □

Remark 6.3.5. When applied to the equation $u' = f(t,u)$ the solutions obtained by forward and backward differences correspond to the explicit and implicit Euler methods, respectively.

6.3.6 Definition: The **finite difference method** (with uniform subdivisions) for the discretization of the BVP $Lu = f$ on the interval $I = [a, b]$ with homogeneous boundary conditions, i.e., for $u \in V$ as in definition 6.2.1, is defined by

1. choosing a partition $a = x_0 < x_1 < \cdots < x_n = b$ with $n \in \mathbb{N}$,

$$h := (b - a)/n \quad \text{and} \quad x_k = kh, \quad k = 0, \dots, n,$$

2. replacing all differential operators in L by finite differences, evaluated at x_k,
3. considering and computing the approximations y_k of the solution $u(x_k)$ at the discrete points $x_0, \dots, x_n$.

Example 6.3.7. Using the 3-point stencil for u'' and central differences for u', the BVP

$$-u''(x) + \beta(x)u'(x) + \gamma(x)u(x) = f(x), \qquad u(a) = u(b) = 0,$$

and the abbreviations $\beta_k = \beta(x_k)$, $\gamma_k = \gamma(x_k)$ $f_k = f(x_k)$, we obtain the discrete system of equations

$$\frac{-y_{k+1} + 2y_k - y_{k-1}}{h^2} + \beta_k \frac{y_{k+1} - y_{k-1}}{2h} + \gamma_k y_k \;=\; f_k\,, \quad \text{for } \; k = 1, \dots, n-1, \tag{6.14}$$

with $y_0 = y_n = 0$, or in matrix notation

$$L_h\, y = \begin{pmatrix} \lambda_1 & \nu_1 & & \\ \mu_2 & \lambda_2 & \ddots & \\ & \ddots & \ddots & \nu_{n-2} \\ & & \mu_{n-1} & \lambda_{n-1} \end{pmatrix} \begin{pmatrix} y_1 \\ \vdots \\ y_{n-1} \end{pmatrix} = \begin{pmatrix} f_1 \\ \vdots \\ f_{n-1} \end{pmatrix} = f_h. \tag{6.15}$$

where

$$\lambda_k = \frac{2}{h^2} + \gamma_k\,, \quad \mu_k = -\frac{1}{h^2} - \frac{\beta_k}{2h}\,, \quad \text{and} \quad \nu_k = -\frac{1}{h^2} + \frac{\beta_k}{2h}\,. \tag{6.16}$$

Remark 6.3.8. Like our view to the continuous BVP has changed w.r.t. IVPs, the discrete problem is now a fully coupled linear system which has to be solved by methods of linear algebra, rather than time stepping. In fact, we have $n - 1$ unknown variables $y_1, \dots, y_{n-1}$ and $n - 1$ equations, such that existence and uniqueness of solutions are equivalent.

6.4 Existence, stability and convergence

6.4.1. Since the solution of the discretized boundary value problem is a problem in linear algebra, we have to study properties of the matrix L_h. The shortest and most elegant way to prove stability is through the properties of M-matrices, which we present here very shortly. We are not dwelling on this approach too long, since it is sufficient for stability, but by far not necessary and particular to low order methods.

The fact that L_h is an M-matrix requires some knowledge of irreducible weakly diagonal dominant matrices, which we have already come across in the last chapter in Numerik 0, in the context of stationary iterative methods.

6.4.2 Definition: A quadratic $n \times n$-matrix A is called an **M-matrix** if it satisfies the following properties:

$$a_{ii} > 0, \quad a_{ij} \le 0, \qquad i,j = 1,\dots,n, \quad j \neq i. \tag{6.17}$$

The entries of $A^{-1} = (c_{ij})_{i,j=1}^n$ satisfy

$$c_{ij} \ge 0, \qquad i,j = 1,\dots,n. \tag{6.18}$$

6.4.3 Lemma: The matrix L_h defined in (6.15) above is an M-matrix provided that

$$\gamma_k \ge 0 \quad \text{and} \quad |\beta_k| < \frac{2}{h}. \tag{6.19}$$

Proof. It is easy to verify that the two conditions in (6.19) are sufficient for the first M-matrix property.

The proof of positivity of the inverse is based on irreducible diagonal dominance, which is too long and too specialized at this point and thus we will omit it. See, e.g., [Ran17b, Hilfssatz 10.2]. □

Remark 6.4.4. The finite element method, discussed next semester in “Numerik 2 – Finite Elements”, provides a much more powerful theory to deduce solvability and stability of the discrete problem.

6.4.5 Lemma: Let A be an M-matrix. If there is a vector w such that for the vector $v = Aw$ there holds

$$v_i \ge 1, \quad i = 1,\dots,n,$$

then

$$\|A^{-1}\|_\infty \le \|w\|_\infty. \tag{6.20}$$

Proof. Let $x \in \mathbb{R}^n$ and $y = A^{-1}x$. Then,

$$|y_i| = |\sum c_{ij} x_j| \le \sum c_{ij} |x_j| \le \|x\|_\infty \sum c_{ij} v_j.$$

Thus,

$$|y_i| \le \|x\|_\infty \bigl(A^{-1}v\bigr)_i = \|x\|_\infty \bigl(A^{-1}Aw\bigr)_i \le \|x\|_\infty |w_i|.$$

Taking the maximum over all i and dividing by $\|x\|_\infty$, we obtain

$$\|A^{-1}\|_\infty = \sup_{x\in\mathbb{R}^n} \frac{\|A^{-1}x\|_\infty}{\|x\|_\infty} \le \|w\|_\infty.$$

□

6.4.6 Theorem: Assume that (6.19) holds and that there exists a constant $\delta < 2$ such that

$$|\beta_k| \le \frac{\delta}{b-a}. \tag{6.21}$$

Then, the matrix L_h defined in (6.15) is invertible and

$$\|L_h^{-1}\|_\infty \le \frac{(b-a)^2}{8-4\delta}. \tag{6.22}$$

Proof. Consider the function

$$p(x) = (x-a)(b-x) = -x^2 + (a+b)x - ab,$$

with derivatives $p'(x) = a+b-2x$ and $p''(x) = -2$, and a maximum of $(b-a)^2/4$ at $x = (a+b)/2$. Choose the values $p_k = p(x_k)$. Due to the consistency results in lemma 6.3.4, we know that $D_h^2 p \equiv p''$ and $D_h^c p \equiv p'$ are exact, such that, for all $k = 1, \dots, n-1$,

$$(L_h p)_k = 2 + \beta_k p'(x_k) + \gamma_k p(x_k) \ge 2 - |\beta_k|(b-a) \ge 2 - \delta.$$

Since L_h is a M-matrix, the vector $w = \frac{1}{2-\delta}p$ can then be used to bound the inverse of L_h using Lemma 6.4.5. □

Remark 6.4.7. The assumptions of the previous theorem involve two sets of conditions on the parameters β_k and γ_k. Since

$$\frac{\delta}{b-a} < \frac{2}{b-a} \le \frac{2n}{b-a} = \frac{2}{h},$$

condition (6.21) actually implies the second condition in (6.19). It is in fact not necessary in this form, but a better estimate requires more advanced analysis.

The condition on γ_k in (6.19) is indeed necessary, as will be seen when we study partial differential equations. The second condition in (6.19), on the other hand, relates the coefficients β_k to the mesh size and can be avoided, as seen in the next example.

Example 6.4.8. By changing the discretization of the first order term to an **upwind** finite difference method, we obtain an M-matrix independent of the relation of β_k and h. To this end define

$$\beta(x)D_h^{\uparrow}u(x) = \begin{cases} \beta(x)D_h^- u(x) & \text{if } \beta(x) > 0 \\ \beta(x)D_h^+ u(x) & \text{if } \beta(x) < 0 \end{cases}. \tag{6.23}$$

This changes the matrix L_h to a matrix $L_h^{\uparrow}$ with entries

$$\lambda_k = \frac{2}{h^2} + \frac{|\beta_k|}{h} + \gamma_k\,, \quad \mu_k = -\frac{1}{h^2} - \frac{\max\{0,\beta_k\}}{h}\,, \quad \nu_k = -\frac{1}{h^2} + \frac{\min\{0,\beta_k\}}{h}\,. \tag{6.24}$$

As a consequence, the off-diagonal elements always remain non-positive and the diagonal elements remain positive provided only that $\gamma_k \ge 0$, for all k. Thus, $L_h^{\uparrow}$ is an M-matrix with a bounded inverse, independent of the values of β_k. However, crucially, the consistency order is reduced from two to one.

6.4.9 Theorem: Consider the boundary value problem defined in definition 6.2.1 with $\beta, \gamma, f \in C^4(a,b)$ and $\gamma(x) \geq 0$ for all $x \in [a,b]$. Let $y \in \mathbb{R}^{n-1}$ be the finite difference approximation for this problem in Example 6.3.7. If there exists a $\delta < 2$ such that $\max_{x\in[a,b]}|\beta(x)| \leq \delta/(b-a)$, then there exists a constant c independent of h such that

$$\max_{0\leq k\leq n} |u_k - y_k| \leq ch^2. \tag{6.25}$$

For the solution $y^{\uparrow} \in \mathbb{R}^{n-1}$ of the upwind finite difference approximation in Example 6.4.8 there exists a constant c independent of h such that

$$\max_{0\leq k\leq n} |u_k - y_k^{\uparrow}| \leq ch. \tag{6.26}$$

without any additional assumptions on the function β.

Proof. Let $n \in \mathbb{N}$ (and thus $h > 0$) be arbitrary but fixed and let $U = (u_k)_{k=1}^{n-1}$ be the vector containing the values of the exact solution at $x_1, \ldots, x_{n-1}$. Considering first the discretisation in Example 6.3.7 and denoting by

$$\tau_k := (L_h U)_k - (Lu)(x_k), \quad k = 1, \ldots, n-1,$$

the consistency errors at the interior grid points. Then

$$\big(L_h(U-y)\big)_k = (L_h U)_k - (Lu)(x_k) + (Lu)(x_k) - (L_h y)_k = \tau_k + f_k - f_k = \tau_k$$

and it follows from (6.13) that

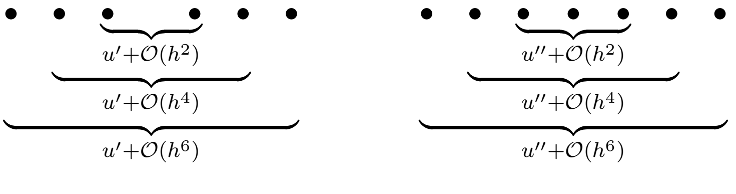
$$\|L_h(U-y)\|_\infty = \|\tau\|_\infty \leq ch^2$$

with c independent of h. Since β, γ satisfy the assumptions of theorem 6.4.6 (for arbitrary $h > 0$), we can conclude that there exists a $c' > 0$ independent of h such that

$$\|U - y\|_\infty = \|L_h^{-1} L_h (U-y)\|_\infty \leq \|L_h^{-1}\|_\infty \|\tau\|_\infty \leq c' h^2.$$

The proof for the upwind discretisation in Example 6.4.8 is identical, but as discussed does not require any boundedness of β to guarantee stability and due to the use of the forward/backward difference quotients, the consistency error is only of $\mathcal{O}(h)$. □

Remark 6.4.10. Finite differences can be generalized to higher order by extending the stencils by more than one point to the left and right of the current point. Whenever we add two points to the symmetric difference formulas, we can gain two orders of consistency.



Similarly, we can define one-sided difference formulas, which get us close to multistep methods. The matrices generated by these formulas are not M-matrices anymore, although you can show for the 4th order formula for the second derivative that it yields a product of two M-matrices. While this rescues the theory in a particular instance, M-matrices do not provide a theoretical framework for general high order finite differences anymore.

Very much like the starting procedures for high order multistep methods, high order finite differences can lead to difficulties at the boundaries. Here, the formulas must be truncated and for instance be replaced by one-sided formulas of equal order.

All these issues motivate the study of different discretization methods in the next course.

Chapter 7

Outlook towards partial differential equations

Finite difference methods for two-point boundary value problems have a natural extension to higher dimensions. There, we deal with partial derivatives $\frac{\partial}{\partial x_1}$, $\frac{\partial}{\partial x_2}$, $\frac{\partial}{\partial x_3}$ and $\frac{\partial}{\partial t}$.

As an outlook towards topics in the numerical analysis of partial differential equations, we close these notes by a short introduction by means of a some examples.

7.1 The Laplacian and harmonic functions

7.1.1 Definition: the **Laplacian** in two (three) space dimensions is the sum of the second partial derivatives

$$\Delta u = \frac{\partial^2}{\partial x_1^2} u + \frac{\partial^2}{\partial x_2^2} u \left(+ \frac{\partial^2}{\partial x_3^2} u \right) \tag{7.1}$$

The **Laplace equation** is the partial differential equation

$$-\Delta u = 0. \tag{7.2}$$

The **Poisson equation** is the partial differential equation

$$-\Delta u = f. \tag{7.3}$$

Solutions to the Laplace equations are called **harmonic functions**.

7.1.2 Theorem (Mean-value formula for harmonic functions): Let $u \in C^2(\Omega)$ be a solution to the Laplace equation. Then, u has the mean value property

$$u(\mathbf{x}) = \frac{1}{r^{d-1}\omega(d)} \int_{\partial B_r(\mathbf{x})} u(\mathbf{y})\, ds, \tag{7.4}$$

where $\partial B_r(\mathbf{x}) \subset \Omega$ is the sphere of radius r around $\mathbf{x}$ and $\omega(d)$ is the volume of the unit sphere in $\mathbb{R}^d$.

Proof. First, we rescale the problem to

$$\Phi(r) = \frac{1}{r^{d-1}\omega(d)} \int_{\partial B_r(\mathbf{x})} u(\mathbf{y})\, ds = \frac{1}{\omega(d)} \int_{\partial B_1(0)} u(\mathbf{x}+r\mathbf{z})\, ds.$$

Then, it follows by the Gauß theorem for the vector valued function ∇u that

$$\begin{aligned}
\Phi'(r) &= \frac{1}{\omega(d)} \int_{\partial B_1(0)} \nabla u(\mathbf{x}+r\mathbf{z}) \cdot \mathbf{z}\, ds_z \\
&= \frac{1}{r^{d-1}\omega(d)} \int_{\partial B_r(\mathbf{x})} \nabla u(\mathbf{y}) \cdot \frac{\mathbf{y}-\mathbf{x}}{r}\, ds_y \\
&= \frac{1}{r^{d-1}\omega(d)} \int_{\partial B_r(\mathbf{x})} \frac{\partial}{\partial \mathbf{n}} u(\mathbf{y})\, ds_y \\
&= \frac{1}{r^{d-1}\omega(d)} \int_{B_r(\mathbf{x})} \Delta u(\mathbf{y})\, d\mathbf{y} = 0.
\end{aligned}$$

Therefore, $\Phi(r)$ is constant. Because of continuity, we have

$$\lim_{r\to 0} \Phi(r) = \lim_{r\to 0} \frac{1}{r^{d-1}\omega(d)} \int_{\partial B_r(\mathbf{x})} u(\mathbf{y})\, ds = u(\mathbf{x}),$$

which proves our theorem. □

7.1.3 Theorem (Maximum principle): Let a function $u \in C^2(\Omega)$ be a solution to the Laplace equation on an open, bounded, connected domain Ω. Then, if there is an interior point $\mathbf{x}_0$ of Ω, such that for a neighborhood $U \subset \Omega$ of $\mathbf{x}_0$ there holds

$$u(\mathbf{x}_0) \ge u(\mathbf{x}) \qquad \forall \mathbf{x} \in U,$$

then the function is constant in Ω.

Proof. Let $\mathbf{x}_0$ be such a local maximum and let $r > 0$ be such that $B_r(\mathbf{x}_0) \subset \Omega$. Assume that there is a point $\mathbf{x}$ on $\partial B_r(\mathbf{x}_0)$, such that $u(\mathbf{x}) < u(\mathbf{x}_0)$. Then, this holds for points $\mathbf{y}$ in a neighborhood of $\mathbf{x}$. Thus, in order that the mean value property holds, there must be a subset of $\partial B_r(\mathbf{x}_0)$ where $u(\mathbf{y}) > u(\mathbf{x}_0)$, contradicting that $\mathbf{x}_0$ is a maximum. Thus, $u(\mathbf{x}) = u(\mathbf{x}_0)$ for all $\mathbf{x} \in B_r(\mathbf{x}_0)$ for all r such that $B_r(\mathbf{x}_0) \subset \Omega$.

Let now $\mathbf{x} \in \Omega$ be arbitrary. Then, there is a (compact) path from $\mathbf{x}_0$ to $\mathbf{x}$ in Ω. Thus, the path can be covered by a finite set of overlapping balls inside Ω, and the argument above can be used iteratively to conclude $u(\mathbf{x}) = u(\mathbf{x}_0)$. □

Corollary 7.1.4. *Let $u \in C^2(\Omega)$ be a solution to the Laplace equation. Then, its maximum and its minimum lie on the boundary, that is, there are points $\underline{\mathbf{x}}, \overline{\mathbf{x}} \in \partial\Omega$, such that*

$$u(\underline{\mathbf{x}}) \le u(\mathbf{x}) \le u(\overline{\mathbf{x}}) \quad \forall \mathbf{x} \in \Omega.$$

Proof. If the maximum of u is attained in an interior point, the maximum principle yields a constant solution and the theorem holds trivially. On the other hand, theorem 7.1.3 does not make any prediction on points at the boundary, which therefore can be maxima. The same holds for the minimum, since $-u$ is also a solution to the Laplace equation. □

Corollary 7.1.5. *The Poisson equation with homogeneous boundary condition, $u \equiv 0$ on $\partial\Omega$, has a unique solution.*

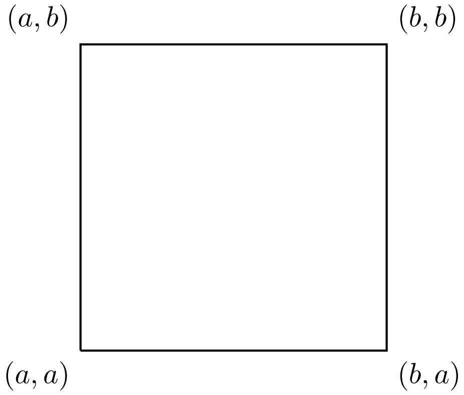
Proof. Assume there are two functions $u, v \in C^2(\Omega)$ with $u = v = 0$ on $\partial\Omega$ such that

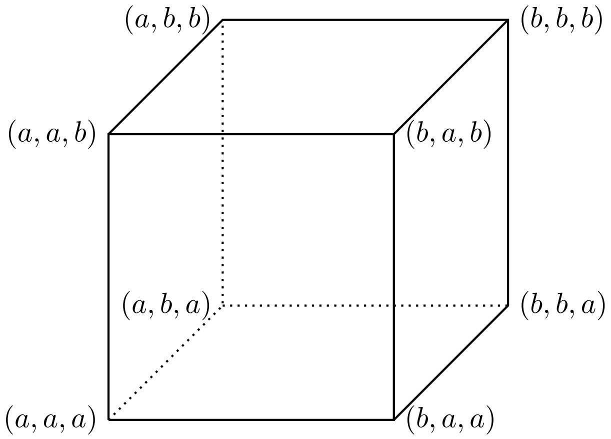
$$-\Delta u = -\Delta v = f.$$

Then, $w = u - v$ solves the Laplace equation with $w = 0$ on $\partial\Omega$. Due to the maximum principle, $w \equiv 0$ and $u = v$. □

7.2 Finite difference methods in higher dimensions

Example 7.2.1. The notion of an interval I can be extended to higher dimensions by a square $\Omega = I^2$ or a cube $\Omega = I^3$.



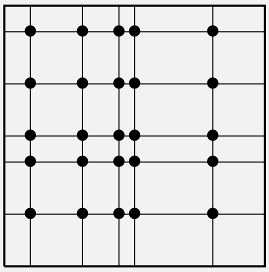


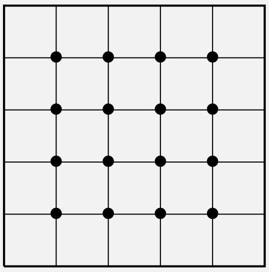
Example 7.2.2. We consider Dirichlet boundary conditions

$$u(\mathbf{x}) = u_B(\mathbf{x}), \qquad \text{for } \mathbf{x} \in \partial\Omega. \tag{7.5}$$

As for two-point boundary value problems, we can reduce our considerations to homogeneous boundary conditions $u_B \equiv 0$ by changing the right hand side in the Poisson equation.

7.2.3 Definition: A **Cartesian grid** on a square (cube) domain Ω consists of the intersection points of lines (planes) parallel to the coordinate axes (planes).

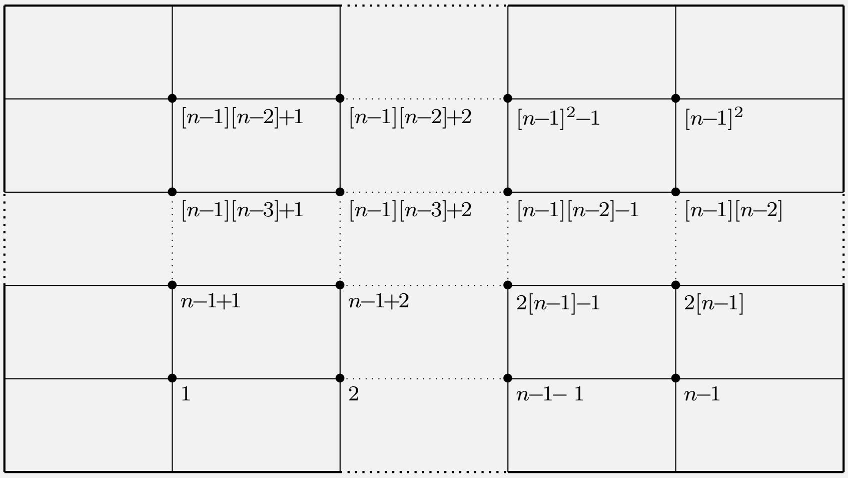




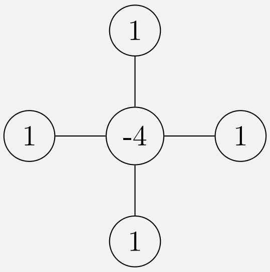
The grid is called **uniform**, if all lines (planes) are at equal distances.

For the remainder of this discussion let us restirct to the two-dimensional case, $d = 2$, and to uniform Cartesian grids.

7.2.4 Definition: The vector y of discrete values is defined in grid points which run in x_1- and x_2-direction. In order to obtain a single index for every entry of this vector in linear algebra, we use **lexicographic numbering**.



7.2.5 Definition: The **5-point stencil** consists of the sum of a 3-point stencil in x_1- and a 3-point stencil in x_2-direction. Its graphical representation is



For a generic row of the linear system, where the associated point is not neighboring the boundary, this leads to

$$D_h^2 u(x_k) = \frac{-u(x_{k-(n-1)}) - u(x_{k-1}) + 4u(x_k) - u(x_{k+1}) - u(x_{k+(n-1)})}{h^2} \tag{7.6}$$

If the point x_k is next to the boundary, the entry corresponding to the neighboring boundary point can be omitted, since the value is assumed to be zero there.

Example 7.2.6. The matrix L_h obtained for the Laplacian on $\Omega = [0, 1]^2$ using the 5-point stencil on a uniform Cartesian mesh of mesh spacing $h = 1/n$ with lexicographic numbering is in $\mathbb{R}^{N \times N}$ with $N = (n-1)^2$ and has the structure

$$L_h = n^2 \begin{bmatrix} D & -I & & & \\ -I & D & -I & & \\ & \ddots & \ddots & \ddots & \\ & & -I & D & -I \\ & & & -I & D \end{bmatrix}, \quad D = \begin{pmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 4 & -1 \\ & & & -1 & 4 \end{pmatrix} \in \mathbb{R}^{(n-1)\times(n-1)}.$$

7.2.7 Theorem: The matrix L_h obtained by discretising the Laplace operator via the 5-point stencil formula is an M-matrix and the solution of the discrete problem

$$L_h y = f$$

is stable in the sense that there is a constant c independent of h such that

$$\|L_h^{-1}\|_\infty \le c.$$

Proof. The proof is identical to the proof for 2-point boundary value problems. To show boundedness of $\|L_h^{-1}\|_\infty$ we can use in a similar way the function

$$p(x,y) = (x-a)(b-x)(y-a)(b-y).$$

□

7.2.8 Theorem: The finite difference approximation in Example 7.2.6 for the Poisson equation in (7.3) on the unit square $\Omega = [0,1]^2$ with homogeneous Dirichlet conditions is convergent of second order, i.e.

$$\max_{k=1,\dots,(n-1)^2} |u(x_k) - y_k| \le ch^2.$$

Proof. We apply the consistency bound in (6.13) in the x_1- and x_2-direction separately, obtaining

$$\left| \frac{\partial^2}{\partial x^2} u(x,y) - \frac{u(x+h,y) - 2u(x,y) + u(x-h,y)}{h^2} \right| \le ch^2$$

$$\left| \frac{\partial^2}{\partial y^2} u(x,y) - \frac{u(x,y+h) - 2u(x,y) + u(x,y-h)}{h^2} \right| \le ch^2,$$

and deduce the second-order consistency of the 5-point stencil by the triangle inequality. The remainder of the proof is identical to the proof of theorem 6.4.9. □

7.2.9 Theorem: Let y be the solution to the finite difference method for the Laplace equation with the 5-point stencil. Then, the maximum principle holds for y, namely, if there is $k \in \{1,\dots,(n-1)^2\}$ such that $y_k \ge y_j$ for all $j \ne k$ and $y_k \le y_B$ for any boundary value, then y is constant.

Proof. From equation (7.6), it is clear that a discrete mean value property holds, that is, y_k is the mean value of its four neighbors. Therefore, if $y_k \ge y_j$, for all neighboring indices j of k, we have $y_j = y_k$. We conclude by following a path through the grid points. □

7.3 Evolution equations

After an excursion to second order differential equations depending on more than one spatial variables, we are now returning to problems depending on time. But this time, on time *and* space. As for the nomenclature, we have encountered ordinary differential equations as equations or systems depending on a time variable only, then partial differential equations (PDE) with several, typically spatial, independent variables. While the problems considered here are covered by the definition of PDE, time and space are fundamentally different. Therefore, we introduce the concept of evolution equations.

While the problems in definition 7.1.1 are PDEs of **elliptic** type. The following problems can be either **parabolic** or **hyperbolic**.

7.3.1 Definition: An equation of the form

$$\frac{\partial u}{\partial t}(t,x) = Lu(t,x), \tag{7.7}$$

where $u(t,.)$ is in a function space V on a domain Ω, for all time $t \in \mathbb{R}$, and $L : V \to C(\Omega)$ is a differential operator with respect to the spatial variables x only, is called a linear **evolution equation** of first order (in time).
An **initial boundary value problem** (**IBVP**) for this evolution equation completes the differential equation by conditions

$$u(0,x) = u_0 \qquad x \in \Omega \tag{7.8}$$

$$u(t,x) = g \qquad x \in \partial\Omega,\ t > 0. \tag{7.9}$$

Example 7.3.2. Consider the case of one spatial variable, i.e. $\Omega = [a,b] \subset \mathbb{R}$, and the differential operator L as defined in (6.4), i.e. a general, linear second order differential operator with respect to the spatial variable x, for simplicity with $\beta = \beta(x)$ and $\gamma = \gamma(x)$ independent of t. Furthermore, let $u(t,a) = u(t,b) = 0$.

This PDE is parabolic and for $\beta = \gamma = 0$ it is called the **heat equation**.

We can now discretise the right hand side of (7.7), for every fixed $t \geq 0$ on a spatial grid $x_0, \dots, x_n$, as in Example 6.3.7, to obtain a system of ODEs

$$y'(t) = L_h y(t)$$

for the unknown (semi-discrete) vector $y(t) \in \mathbb{R}^{n-1}$ of approximations to the solution $u(t,\cdot)$ of (7.7) at time t. By choosing as the initial condition

$$y_k(0) = u_0(x_k), \quad k = 1, \dots, n-1$$

we obtain an autonomous linear IVP for $y : [0,T] \to \mathbb{R}^{n-1}$ that we can now solve with our favourite time stepping method.

For $\gamma_k \geq 0$ and $|\beta_k|$ sufficiently small, the eigenvalues of L_h have negative real part and vary strongly in size, e.g. for $\beta = \gamma = 0$ we have $\lambda_1 = -4n^2 \sin(\pi/2n) \approx -\pi^2$ and $\lambda_{n-1} = -4n^2 \sin(\pi(n-1)/2n) \approx -4n^2$. Thus, the problem is stiff, especially for n large, and we should use a stable time stepping method.

From theorem 6.4.9 we know that the spatial discretisation is of second order. Thus, a common time stepping method to use is the Crank-Nicolson method, which is the second order A-stable LMM with the smallest error constant. To distinguish between spatial grid points and time steps, choose $m \in \mathbb{N}$ and let $\eta = T/m$ be the time step size. We denote the approximation of $y(t_j)$ at the jth time step t_j, $j = 1, \dots, m$, by $Y^{(j)} \in \mathbb{R}^{n-1}$. Applying the Crank-Nicolson method we finally obtain the fully discrete system

$$Y^{(j)} = Y^{(j-1)} + \frac{\eta}{2}\left(L_h Y^{(j)} + L_h Y^{(j-1)}\right) \quad \Leftrightarrow \quad \left(I - \frac{\eta}{2}L_h\right)Y^{(j)} = \left(I + \frac{\eta}{2}L_h\right)Y^{(j-1)}$$

for the jth time step. Since the real part of the spectrum of L_h is negative, the matrix on the left hand side is invertible, so that we can uniquely solve this system for any $\eta > 0$.

We finish by stating the convergence result for this example.

7.3.3 Theorem: Consider the problem in definition 7.3.1 in one space dimension, i.e. $\Omega = [a, b] \subset \mathbb{R}$, and the differential operator L as defined in (6.4) with $\beta = \beta(x)$ and $\gamma = \gamma(x)$ independent of t. Furthermore, let $u(t, a) = u(t, b) = 0$. Then, with central finite difference discretisation of L with mesh width h and applying the Crank-Nicolson method to discretise in time, as described in Example 7.3.2 with step size $\eta \leq h$, there exists a constant $c > 0$ independent of h such that

$$\max_{j=0,\dots,m} \max_{k=0,\dots,n} \left| Y_k^{(j)} - u(t_j, x_k) \right| \leq ch^2.$$

Appendix A

Appendix

A.1 Comments on uniqueness of an IVP

For a first order differential equation, Lipschitz continuity of f is only a sufficient and not, as one might think, a necessary condition for uniqueness of a first order differential equation. The following theorem and proof show that it is indeed possible to have uniqueness without assuming Lipschitz continuity.

A.1.1 Theorem (Non-necessity of L-continuity): Let f be a continous function satisfying $f(x) > 0$ for all $x \in \mathbb{R}$. Then, the solution to the (autonomous) IVP

$$u'(t) = f\big(u(t)\big) \tag{A.1a}$$

$$u(t_0) = u_0 \tag{A.1b}$$

is globally unique for all $(t_0, u_0) \in \mathbb{R}^2$.

Proof. Assume two solutions φ, $\psi \colon I \to \mathbb{R}$ on an open intervall I with $t_0 \in I$. Then,

$$1 = \frac{\varphi(t)'}{f(\varphi(t))} = \frac{\psi(t)'}{f(\psi(t))} \qquad \text{for all } t \in I. \tag{A.2}$$

Define the function $F \colon \mathbb{R} \to \mathbb{R}$ through

$$F(x) = \int_{u_0}^{x} \frac{\mathrm{d}s}{f(s)}.$$

F is continously differentiable since

$$\partial_x F(x) = \partial_x \left(\int_{u_0}^{x} \frac{\mathrm{d}s}{f(s)} \right) = \frac{1}{f(x)}.$$

Obviously, F is also stricly increasing, hence injective on $\mathbb{R}$: Take x, $y \in \mathbb{R}$ and assume without loss of generality that $x < y$. Then we have $F(x) < F(y)$ and thus $F(x) \neq F(y)$. Thus, F is an injection.

Also, for all $t \in I$, it follows from (A.2) that

$$F(\varphi(t)) = \int_{u_0}^{t} \frac{\varphi'(s)}{f(\varphi(s))} \,\mathrm{d}s = \int_{u_0}^{t} \frac{\psi'(s)}{f(\psi(s))} \,\mathrm{d}s = F(\psi(t)).$$

Thus, since F is injective, we have $\varphi(t) = \psi(t)$ for all $t \in I$. In conclusion, the IVP (A.1) has a unique solution. □

A.2 Properties of matrices

A.2.1 The matrix exponential

Definition A.2.1. The matrix exponential e^A of a matrix $A \in \mathbb{R}^{d\times d}$ is defined by its power series

$$e^A = \sum_{k=0}^{\infty} \frac{A^k}{k!}. \tag{A.3}$$

Lemma A.2.2. *The power series* (A.3) *converges for each matrix A. It is therefore valid to write*

$$e^A = \lim_{m\to\infty} \sum_{k=0}^{m} \frac{A^k}{k!} = \sum_{k=0}^{\infty} \frac{A^k}{k!}. \tag{A.4}$$

Proof. Let $\|\cdot\|$ be a submultiplicative matrix norm on $\mathbb{R}^d$. We want to show that the sequence of partial sums $(S_n)_{n\in\mathbb{N}_0}$ with S_n given as $\lim_{m\to\infty} \sum_{k=n}^{m} \frac{A^k}{k!}$ converges to $S := e^A = \lim_{m\to\infty} \sum_{k=0}^{m} \frac{A^k}{k!}$. Consider therefore

$$\|S - S_n\| = \left\| \lim_{m\to\infty} \sum_{k=n+1}^{m} \frac{A^k}{k!} \right\| = \lim_{m\to\infty} \left\| \sum_{k=n+1}^{m} \frac{A^k}{k!} \right\|. \tag{A.5}$$

Using the triangle-inequality and the fact that $\|\cdot\|$ is submultiplicative yields

$$\lim_{m\to\infty} \sum_{k=n+1}^{m} \left\| \frac{A^k}{k!} \right\| \leq \lim_{m\to\infty} \sum_{k=n+1}^{m} \frac{1}{k!} \|A\|^k. \tag{A.6}$$

Considering the limit $n \to \infty$ concludes the proof. □

Lemma A.2.3 (Properties of the matrix exponential)**.** *The following relations hold true:*

$$\begin{aligned}
e^0 &= \mathbb{I} && && \text{(A.7)}\\
e^{\alpha A} e^{\beta A} &= e^{(\alpha+\beta)A}, && \forall A \in \mathbb{R}^{d\times d}\ \forall \alpha, \beta \in \mathbb{R}, && \text{(A.8)}\\
e^A e^{-A} &= \mathbb{I} && \forall A \in \mathbb{R}^{d\times d}, && \text{(A.9)}\\
e^{T^{-1}AT} &= T^{-1}\, e^A\, T && \forall A, T \in \mathbb{R}^{d\times d}\ \textit{invertible}, && \text{(A.10)}\\
e^{\mathrm{diag}(\lambda_1,\ldots,\lambda_d)} &= \mathrm{diag}(e^{\lambda_1}, \ldots, e^{\lambda_d}) && \forall \lambda_i \in \mathbb{R},\ i = 1, \ldots, d. && \text{(A.11)}
\end{aligned}$$

Moreover, e^A is invertible for arbitrary quadratic matrices A with $(e^A)^{-1} = e^{-A}$.

Proof. The equality (A.7) follows directly from the definition.

For (A.8) consider the function $\varphi(\alpha)$ given by

$$\varphi(\alpha) = e^{\alpha A}e^{\beta A} - e^{(\alpha+\beta)A}.$$

Then

$$\varphi'(\alpha) = A\big(e^{\alpha A}e^{\beta A} - e^{(\alpha+\beta)A}\big) = A\varphi(\alpha) \quad \text{and} \quad \varphi(0) = \mathbb{I}e^{\beta A} - e^{\beta A} = 0,$$

giving us an IVP for $\varphi(\alpha)$ with unique solution $\varphi(\alpha) = e^{\alpha A}\varphi(0) = 0$, and the identity in (A.8) follows.

Equation (A.9) is a special case of (A.8) with parameters $\alpha = 1$ and $\beta = -1$, which in combination with (A.7) leads to the result.

For (A.10) note that $\mathbb{R}^{d\times d}$ forms a ring and is thus associative. Then, for $k \in \mathbb{N}_0$, we have

$$\begin{aligned}(T^{-1}AT)^k &= (T^{-1}AT)(T^{-1}AT)\cdots(T^{-1}AT)(T^{-1}AT)\\ &= T^{-1}A(TT^{-1})A(T\cdots T^{-1})A(TT^{-1})AT = T^{-1}A^kT\end{aligned}$$

and thus

$$e^{T^{-1}AT} = \sum_{k=0}^{\infty}\frac{1}{k!}(T^{-1}AT)^k = \sum_{k=0}^{\infty}\frac{1}{k!}T^{-1}A^kT = T^{-1}\cdot\left(\sum_{k=0}^{\infty}\frac{1}{k!}A^k\right)\cdot T = T^{-1}e^AT.$$

To prove (A.11), let $D = \operatorname{diag}(\lambda_1,\ldots,\lambda_d) \in \mathbb{R}^{d\times d}$ where $\lambda_i \in \mathbb{R}$, $i = 1,\ldots,d$. Then, $D^k = \operatorname{diag}(\lambda_1^k,\ldots,\lambda_n^k)$, for any $k \in \mathbb{N}_0$, and we have

$$\begin{aligned}e^D &= \lim_{m\to\infty}\sum_{k=0}^{m}\frac{1}{k!}\operatorname{diag}(\lambda_1^k,\ldots,\lambda_n^k) && \text{(A.12)}\\ &= \lim_{m\to\infty}\sum_{k=0}^{m}\operatorname{diag}\left(\frac{1}{k!}\lambda_1^k,\ldots,\frac{1}{k!}\lambda_n^k\right) && \text{(A.13)}\\ &= \lim_{m\to\infty}\operatorname{diag}\left(\sum_{k=0}^{m}\frac{1}{k!}\lambda_1^k,\ldots,\sum_{k=0}^{m}\frac{1}{k!}\lambda_n^k\right) && \text{(A.14)}\\ &= \operatorname{diag}\left(\lim_{m\to\infty}\sum_{k=0}^{m}\frac{1}{k!}\lambda_1^k,\ldots,\lim_{m\to\infty}\sum_{k=0}^{m}\frac{1}{k!}\lambda_n^k\right) && \text{(A.15)}\\ &= \operatorname{diag}(e^{\lambda_1^k},\ldots,e^{\lambda_n^k}) && \text{(A.16)}\end{aligned}$$

Here, we have used the absolute convergence of the series and that these matrices are elements of the ring $R^{d\times d}$.

The final property follows immediately from (A.9). □

Example A.2.4. We will perform an exemplary calculation of a matrix exponential. Consider

$$A = \begin{pmatrix} 0 & 1 \\ k^2 & 0 \end{pmatrix}.$$

As the matrix exponential of a diagonal matrix is simply a diagonal matrix with the exponential of the entries, we diagonalize A.

To diagonalize A, note that the eigenvalues λ_1, λ_2 of A are $\lambda_1 = k$ and $\lambda_2 = -k$. Let $D = \operatorname{diag}(\lambda_1, \lambda_2) = \operatorname{diag}(k, -k)$. The corresponding eigenvectors are $\psi_1 = \big(1, k\big)^T$ and $\psi_2 = \big(1, -k\big)$. The matrix $\Psi = (\psi_1|\psi_2) \in \mathbb{R}^{2\times 2}$ satisfies

$$A = \Psi^{-1} D \Psi.$$

The inverse of Ψ is given as

$$\Psi^{-1} = \frac{1}{2}\begin{pmatrix} 1 & 1/k \\ 1 & -1/k \end{pmatrix}$$

and with the above lemma we can now calculate

$$e^A = \Psi e^D \Psi^{-1} = \frac{1}{2}\begin{pmatrix} e^k + e^{-k} & 1/k(e^k - e^{-k}) \\ k(e^k - e^{-k}) & e^k + e^{-k} \end{pmatrix} = \begin{pmatrix} \cosh(k) & 1/k \sinh(k) \\ k\sinh(k) & \cosh(k) \end{pmatrix}.$$

A.3 The Banach fixed-point theorem

A.3.1 Theorem (Banach fixed-point theorem): Let $\Omega \subset \mathbb{R}$ be a closed set and $f\colon \Omega \to \Omega$ a contraction, i.e. there exists $\gamma \in (0,1)$ such that $|f(x) - f(y)| \leq \gamma|x - y|$. Then, there exists a unique $x^* \in \Omega$ such that $f(x^*) = x^*$.

Proof. Let $x_0 \in \Omega$ and define $x_{k+1} = f(x_k)$. First, we prove existence using the Cauchy-criterion. Let $k, n \in \mathbb{N}_0$ and consider

$$|x_k - x_{k+m}| = |f(x_{k-1}) - f(x_{k+m-1})| \leq \gamma |x_{k-1} - x_{k+m-1})|.$$

Iteratively, we get

$$|x_k - x_{k+m}| \leq \gamma^k |x_0 - x_m|.$$

We now write $x_0 - x_m = x_0 - x_1 + x_1 - x_2 + \cdots + x_{m-1} - x_m$. The triangle-inequality then yields the estimate

$$\begin{aligned} |x_k - x_{k+m}| &\leq \gamma^k \left(|x_0 - x_1| + |x_1 - x_2| + \cdots + |x_{m-1} - x_m|\right) \\ &\leq \gamma^k |x_0 - x_1| \left(1 + \gamma + \gamma^2 + \cdots + \gamma^{m-1}\right) \leq \frac{\gamma^k}{1-\gamma}|x_0 - x_1|. \end{aligned}$$

As k gets larger this estimate goes to zero.

Concerning uniqueness, let x^* and y^* be fixpoints. Then,

$$|x^* - y^*| = |f(x*) - f(y^*)| \leq \gamma |x^* - y^*|$$

Since $\gamma \in (0,1)$ we immediately obtain $|x^* - y^*| = 0$. Using that $|a| = 0$ if and only if $a = 0$ yields $y^* = x^*$. This concludes the proof. □

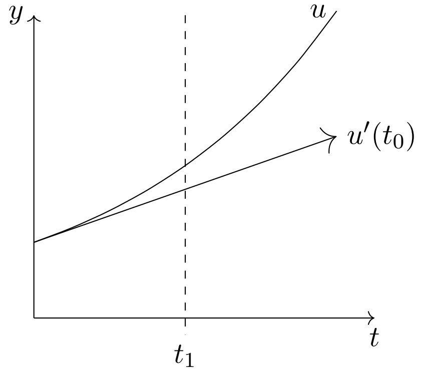
A.4 The implicit and explicit Euler-method

The explicit resp. implicit Euler is given by the one-step method

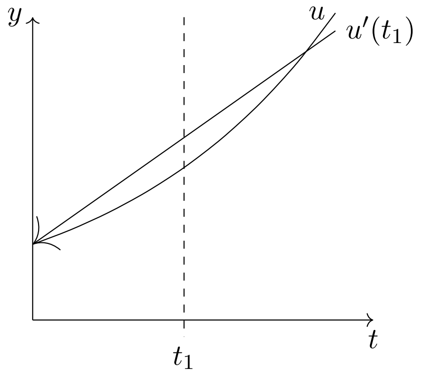
$$y_1 = y_0 + hf(y_0) \qquad \text{resp.} \qquad y_1 = y_0 + hf(y_1)$$

Clearly, the explicit Euler is a rather easy calculation since all one needs are f, h and y_0. The implicit Euler is more difficult to compute since for calculating y_1 we need the value of f at y_1. The goal of this section is to visualize and give an intuition for the two algorithms.

Consider the following visualizations.



For the explicit Euler we take u_0 and u_0'. y_1, our approximated solution for u_1, is chosen as the intersection point of t_1 and $g(t) = y_0 + t \cdot u'(t_0)$.

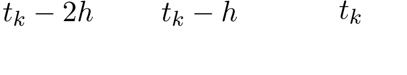


For implicit Euler we go backwards. On the t_1-axis we are looking for an the affine function g that fulfills $g(0) = u_0$ and $g'(t_1) = f(t_1)$. Then we set $y_1 = g(t_1)$.

A.5 Derivation of a BDF-scheme

The BDF formulae use the approximations of the solution at the previous time steps $t_k - sh, \dots, t_k - h$ and the unkown value y_k at t_k that we would like to determine. With the Lagrange polynomial given by $L_i(t) = \prod_{j=0, j\neq i}^{s} \frac{t-t_i}{t_j-t_i}$ we let $y(t) = \sum_{j=0}^{s} y_{k-j} L_{s-j}(t)$. Then, we will assume that y solves the IVP in the point t_k and obtain a linear system from which we derive the desired value y_k.

We now aim to derive the scheme for BDF(2): Let the points $t_k - 2h,\ t_k - h$ and t_k be given.



For the corresponding Lagrange polynomials we have, resp.,

$$L_0(t) = \frac{(t-t_k+h)(t-t_k)}{2h^2}\ ,\ L_1(t) = \frac{(t-t_k)(t-t_kj-2h)}{h^2} \ \text{ and }\ L_2(t) = \frac{(t-t_k+2h)(t-t_k+h)}{2h^2}.$$

By assumption the interpolation polynomial fulfilles the IVP in the point t_k, i.e. there holds $f_k := f(t_k, y(t_k)) = y'(t_k) = \sum_{j=1}^{s} y_{k-j} L'_{k-j}(t)$. Since

$$L_0'(t) = \tfrac{2t-2t_k+h}{2h^2}\,,\ L_1'(t) = -\tfrac{2t-2t_k+2h}{h^2} \quad \text{and} \quad L_2'(t) = \tfrac{2t-2t_k+3h}{2h^2}\,,$$

evaluation at $t = t_k$ yields

$$f_k = \tfrac{1}{2h} y_{k-2} - \tfrac{2}{h} y_{k-1} + \tfrac{3}{2h} y_k\,.$$

The final BDF(2)-scheme is obtained by multiplication with $\frac{2h}{3}$:

$$y_k - \tfrac{4}{3} y_{k-1} + \tfrac{1}{3} y_{k-2} = \tfrac{2}{3h} f_k.$$

Bibliography

[But96] J. C. Butcher. A history of Runge-Kutta methods. *Appl. Numer. Math.*, 20(3):247–260, 1996.

[DB08] P. Deuflhard and F. Bornemann. *Numerische Mathematik 2. Gewöhnliche Differentialgleichungen.* de Gruyter, 3. auflage edition, 2008.

[Heu86] H. Heuser. *Lehrbuch der Analysis. Teil 2.* Teubner, 3. auflage edition, 1986.

[HNW09] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving ordinary differential equations I. Nonstiff problems*, volume 8 of *Springer Series in Computational Mathematics.* Springer, Berlin, second edition edition, 2009.

[HW10] E. Hairer and G. Wanner. *Solving ordinary differential equations II. Stiff and differential-algebraic problems*, volume 14 of *Springer Series in Computational Mathematics.* Springer-Verlag, Berlin, second edition edition, 2010.

[NW06] Jorge Nocedal and Stephen J. Wright. *Numerical optimization.* Springer Series in Operations Research and Financial Engineering. Springer, New York, second edition, 2006.

[Ran17a] R. Rannacher. *Numerik 0: Einführung in die Numerische Mathematik.* Heidelberg University Publishing, 2017. DOI: 10.17885/heiup.206.281.

[Ran17b] R. Rannacher. *Numerik 1: Numerik gewöhnlicher Differentialgleichungen.* Heidelberg University Publishing, 2017. DOI: 10.17885/heiup.258.342.

[Run95] C. Runge. Über die numerische Auflösung von Differentialgleichungen. *Math. Ann.*, 46:167–178, 1895.