Lecture Notes

Numerical Methods for Ordinary Differential Equations

Simone Göttlich

UNIVERSITÄT MANNHEIM

These notes are for internal use only.

September 2022

Course Description:

This course introduces the basic numerical concepts for solving differential equations and gives furthermore an insight into the wide field of their application.

Most of the problems in science, engineering and technology can be modeled by a set of differential equations. These equations are in general too complex to be solved analytically. Thus, this course provides the necessary methods to treat ordinary differential equations numerically by help of the computer.

These lecture notes are just some kind of complementary material for the lecture, which is a compilation from many different sources. It is not planned that they replace the study of good text books.

The script does not claim to be free of errors or complete in its content and style of presentation. In case of doubt, the reader should check the above cited references or similar works.

Many parts of these lecture notes are based on the books:

- *Moler, C. B.*, Numerical Computing with MATLAB, SIAM, 2004.
- *Deuflhard, P., Bornemann, F.*, Scientific Computing with Ordinary Differential Equations, Springer, 2002.
- *Deuflhard, P., Bornemann, F.*, Numerische Mathematik II, de Gruyter, 2008.
- *Quarteroni, A., Saleri, F.*, Scientific Computing with MATLAB, Springer, 2005.
- *Quarteroni, A., Sacco, R., Saleri, F.*, Numerical Mathematics, Springer, 2000.
- *Dahmen, W., Reusken, A.*, Numerik für Ingenieure und Naturwissenschaftler, in german, Springer, 2008.

We would be happy to receive feedback, in particular suggestions for improvement and notifications of typos and other errors.

Simone Göttlich Mannheim, September 2022

Contents

Ordinary Differential Equations

Let us start with some motivating examples that show the wide application range of ordinary differential equations (ODEs). After briefly recalling analytical methods for determining the exact solution of certain kinds of ODEs, we will then focus on numerical methods that enable the approximation of highly complex, analytically unsolvable, (non-)stiff ODEs. Thereby, we will also answer the crucial questions of stability, consistence and convergence.

1 Motivation and Preliminaries

1.1 Introductory Example

There are basically three classes of examples presented: a simple linear equation, a system of linear equations and a system of non-linear equations.

1.1.1 Radioactive Decay

The computation of the radioactive decay of certain products is based on the assumption that the rate of decay is proportional to the recent amount of the material for small time differences Δt. Let $y(t)$ denote the mass of the decaying material at time $t > 0$ and let $\lambda > 0$ be the proportionality constant. Then,

$$-\frac{\Delta y(t)}{y(t)\Delta t} = -\frac{y(t+\Delta t) - y(t)}{y(t)\Delta t} = \lambda.$$

In the limit $\Delta t \to 0$, we obtain the differential equation

$$y'(t) = -\lambda y(t),$$

whose solution reads $y(t) = y(0)e^{-\lambda t}$ for given initial value $y(0)$.

This differential equation is very simple. But we will use it quite often in the following to test the numerical methods, since it possesses some very special properties ($\lambda \gg 1$ $\Rightarrow$ stiff problem).

1.1.2 Harmonic oscillator

The harmonic oscillator model is very important in physics, because any mass subject to a force in stable equilibrium acts as a harmonic oscillator for small vibrations. A simple example for a harmonic (damped) oscillator is a pendulum. We define $y = 0$

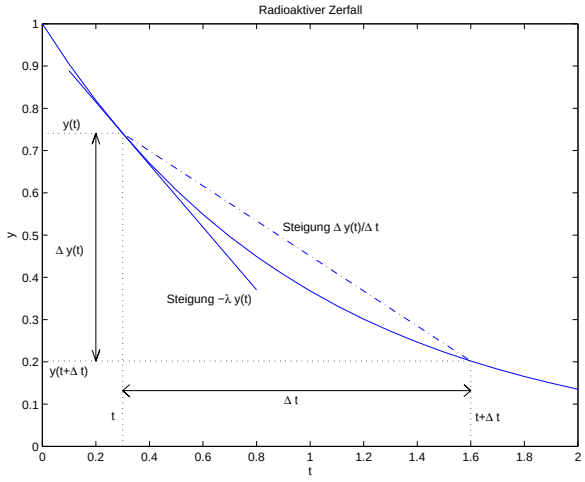


Figure 1.1: Solution of radioactive decay

to be the equilibrium position of the pendulum. The quantity $y(t)$ will be a measure of the displacement from this equilibrium at a given time. Then, the dynamics of the harmonic (damped) oscillator can be described by

$$y'' + \alpha y' + y = 0$$

with $\alpha \geq 0$. We transform this into a first order differential equations and obtain

$$\mathbf{x}' = \mathbf{A}\mathbf{x} \quad \begin{cases} x_1' &= y' = x_2 \\ x_2' &= y'' = -x_1 - \alpha x_2 \end{cases}$$

with $x_1(0)$ and $x_2(0)$ given. The system matrix $\mathbf{A}$ reads

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ -1 & -\alpha \end{pmatrix}.$$

1.1.3 Predator-Prey Model

A further application that is described by differential equations is the population dynamics in biology. We focus on the dynamics of biological systems in which two species interact, one a predator and one its prey. This leads in general to a system of first order, non-linear differential equations which were proposed independently by Alfred J. Lotka in 1925 and Vito Volterra in 1926.

Let us introduce some notations: We denote by $x(t)$ the number of some prey (for example fish) at time t and by $y(t)$ the number of some predator (for example sharks) at time t.

If there were no predators and the prey species had an unlimited food supply, the exponential growth is represented by

$$x'(t) = \alpha x(t), \quad \alpha > 0.$$

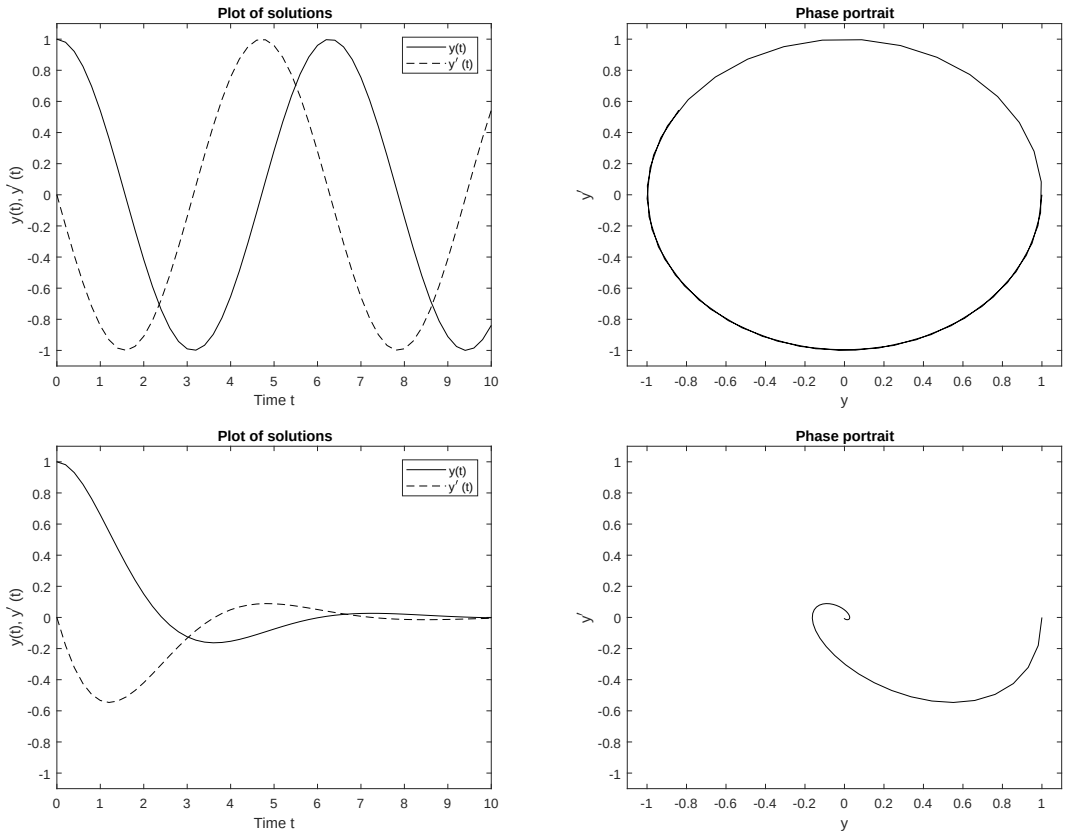


Figure 1.2: The left pictures show the solutions of the model for the undamped case, i.e. $\alpha = 0$ (top) and the damped case with $\alpha > 0$ (bottom). Around the equilibrium a closed trajectory of the undamped harmonic oscillator model describes the relation between the measure of the displacement and its velocity (top right). For the damped harmonic oscillator the trajectory asymptotically tends to the equilibrium.

The factor α can be interpreted as the difference between birth and death rate. But there are predators, which must account for a negative component in the prey growth rate. The rate at which predators encounter prey is jointly proportional to the sizes of the two populations. Encounters which lead to the death of the prey are denoted by the factor β.

$$x'(t) = \alpha x(t) - \beta x(t)y(t), \quad \alpha, \beta > 0. \tag{1.1}$$

Now we consider the predator population. If there were no food supply, the population would die out at a rate proportional to its size, i.e. we would find

$$y'(t) = -\gamma y(t), \quad \gamma > 0.$$

In the absence of food, there is no energy supply to support the birth rate - in contrast to the positive rate α. But there is a food supply: the prey. And what is bad for fish

is good for sharks. That is, the energy to support growth of the predator population is proportional to deaths of prey, so

$$y'(t) = -\gamma y(t) + \delta x(t)y(t), \quad \gamma, \delta > 0. \tag{1.2}$$

Altogether, the equations for the predator-prey model are given by (1.1) and (1.2).

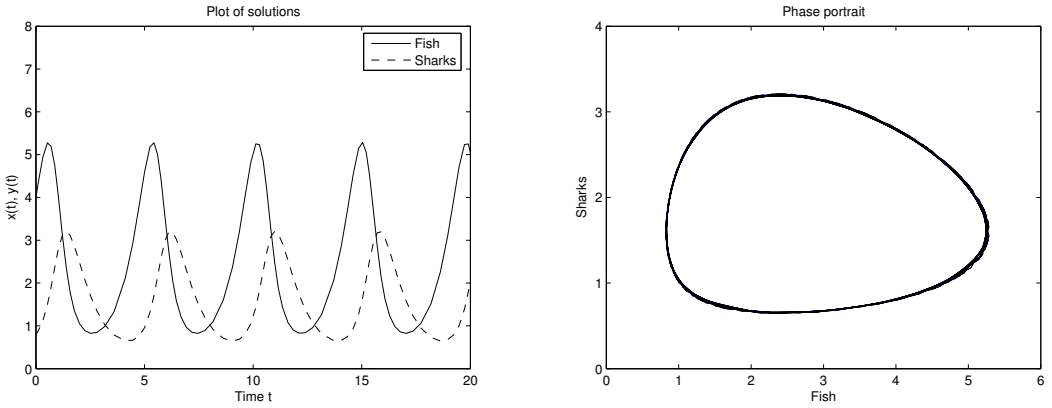


Figure 1.3: The left picture shows the periodical solution of the model, i.e. $\exists T > 0$ with $x(t+T) = x(t), y(t+T) = y(t)$ for all t. Around the equilibrium a closed trajectory of the predator-prey model describes the relation between the population of fish and sharks (right picture).

1.2 Basic Notations

Let $I \subset \mathbb{R}$ be an open interval and $V \subset \mathbb{R}^m$, $m \in \mathbb{N}$. Then we denote the set of functions from I into V that have continuous derivatives up to the order k by $\mathcal{C}^k(I;V)$, $k \in \mathbb{N}_0$. We abbreviate the set of continuous functions by $\mathcal{C}^0(I;V) =: \mathcal{C}(I;V)$ and $\mathcal{C}^k(I,V) =: \mathcal{C}^k(I)$.

Definition 1.1. *Let $t \in I \subset \mathbb{R}$. An* ordinary differential equation (ODE) of order n *is an equation of the form*

$$F(t,\, u,\, u',\, u'',\, \ldots,\, u^{(n-1)},\, u^{(n)}) = 0, \tag{1.3}$$

with $F \in \mathcal{C}^0(U;\mathbb{R})$ and $U \subset \mathbb{R}^{n+2}$ open set.

A function u is called solution *of Eq (1.3) on the interval I, if $u \in \mathcal{C}^n(I;\mathbb{R})$ and Eq (1.3) for all $t \in I$ are valid.*

Unfortunately, the analysis of differential equations that are given in such an implicit form, cf. Eq (1.3), is very difficult. Thus, we assume that the equation can be rewritten in terms of its highest derivative.

Definition 1.2. *The ODE* (1.3) *is called* explicit*, if it is prescribed by*

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

otherwise it is called implicit*.*

Extending these definitions on vector-valued functions $\mathbf{u} : \mathbb{R} \to \mathbb{R}^m$ we state

Definition 1.3. *Let* $f_1, \ldots, f_m \in \mathcal{C}^0(U;\mathbb{R})$ *for the open set* $U \subset \mathbb{R}^{n+1}$*. Then*

$$\begin{aligned} u_1^{(n)} &= f_1(t, \mathbf{u}, \mathbf{u}', \ldots, \mathbf{u}^{(n-1)}) \\ &\vdots \\ u_m^{(n)} &= f_m(t, \mathbf{u}, \mathbf{u}', \ldots, \mathbf{u}^{(n-1)}) \end{aligned} \tag{1.4}$$

is called system of ordinary differential equations of order n*.*

Definition 1.4. *A system of ODEs is said to be* linear*, if the functions* f_i *are linear in the variables* $\mathbf{u},\ \mathbf{u}',\ \ldots,\ \mathbf{u}^{(n)}$*, i.e. Eq* (1.4) *is given by the linear combination*

$$u_i^{(n)} = g_i(t) + \sum_{l=1}^{m} \sum_{j=0}^{n-1} f_{i,l,j}(t)\ x_l^{(j)}$$

where $f_{i,l,j}(t) = f_i(t, u_l^{(j)})$ *and* $x_l^{(j)}$ *are constants. It is called* homogeneous*, if* $g_i \equiv 0$*, otherwise* inhomogeneous*.*

Remark 1.5. *A system of ODEs can be always transformed into a system of first order by introducing new dependent variables* $\mathbf{y} = (\mathbf{u}, \mathbf{u}', \ldots, \mathbf{u}^{(n)})$*. This yields*

$$\begin{aligned} \mathbf{y_1}' &= \mathbf{y_2} \\ &\vdots \\ \mathbf{y_{n-1}}' &= \mathbf{y_n} \\ \mathbf{y_n}' &= \mathbf{f}(t, \mathbf{y}). \end{aligned}$$

Systems whose right-hand side does not explicitly depend on t are of special interest.

Definition 1.6. *A system of ODEs of the form* $\mathbf{F}(\mathbf{u},\ \mathbf{u}', \ldots,\ \mathbf{u}^{(n)}) = \mathbf{0}$ *is called* autonomous*.*

Remark 1.7. *A non-autonomous system is always transferable into an autonomous one by treating* t *as an additional dependent variable.*

Definition 1.8. *Let* $U \subset \mathbb{R}^2$*,* $f : U \to \mathbb{R}$ *and* $(\xi, \eta) \in U$*. Find a differentiable function* $u : I \to \mathbb{R}$ *,* $\xi \in I$*, such that*

$$\begin{aligned} u'(t) &= f\,(t, u(t)) \qquad \forall t \in I \\ u(\xi) &= \eta \end{aligned} \tag{1.5}$$

is called initial value problem (IVP)*. The second equation is known as* initial condition (IC)*.*

1.3 Explicit Solutions for ODEs of First Order

Some differential equations can be solved explicitely if one finds the corresponding primitive (antiderivative). In general, the chance of finding explicit solutions for ODEs of first order is quite good, but for ODEs of higher order is almost hopeless.

1.3.1 ODEs with Separated Variables

Consider the IVP

$$u'(t) = f(t)\, g(u), \qquad u(\xi) = \eta. \tag{1.6}$$

It is an ODE with separated variables.

The following theorem answers the questions of solvability and determination of solutions for Eq (1.6).

Theorem 1.9. *Let $I_t \subset \mathbb{R}$ and $I_u \subset \mathbb{R}$ be two intervals and $f : I_t \to \mathbb{R}$ and $g : I_u \to \mathbb{R}$ be continuous. Let $\xi \in I_t$ and $\eta \in \operatorname{int} I_u$ (interior of I_u). Assume*

$$g(\eta) \neq 0,$$

then there exists a neighborhood $U_\xi \subset I_t$ of ξ (only on one side, if $\xi \in \partial I_t$ is boundary element) and the IVP (1.6) has an unique solution $u : U_\xi \to \mathbb{R}$. This is implicitly given by

$$\int_\eta^{u(t)} \frac{\mathrm{d}s}{g(s)} = \int_\xi^t f(y)\, \mathrm{d}y.$$

Proof: Set

$$G(u) = \int_\eta^u \frac{\mathrm{d}s}{g(s)} \quad \text{and} \quad F(t) = \int_\xi^t f(y)\, \mathrm{d}y.$$

Since $g(\eta) \neq 0$, the function G exists in a neighborhood of η. Here, G is differentiable with $G' = 1/g$ and thus strictly monotonous. According to the Theorem of Inverse Functions, there exists H with $u = H(G(u))$. Therefore, $u = H(F(t))$ holds. Differentiation with respect to t yields $G'(u) \cdot u' = F'$, thus $u' = f(t) \cdot g(u)$. Moreover, $G(\eta) = 0$, $F(\xi) = 0$ and $H(0) = \eta$, thus $u(\xi) = H(F(\xi)) = \eta$.

Let v be an other solution, i.e. $v'(t)/g(v(t)) = f(t)$. Integration with respect to t results in

$$\int_\xi^t f(s)\, \mathrm{d}s = \int_\xi^t \frac{v'(s)\, \mathrm{d}s}{g(v(s))} = \int_{v(\xi)}^{v(t)} \frac{\mathrm{d}s}{g(s)}.$$

Thus, $F(t) = G(v(t))$ and consequently $v(t) = H(F(t)) = u(t)$. ■

1.3.2 Variation of Constants

A descriptive example for an ODE with separated variables is the following linear ODE of first order

$$u'(t) = a(t)u + b(t)\ . \tag{1.7}$$

For the construction of a solution of Eq (1.7) we deal firstly with the homogeneous problem

$$u'(t) = a(t)u\ . \tag{1.8}$$

Applying Th 1.9 the *general solution* reads

$$\begin{aligned}
&\int \frac{1}{u}\,\mathrm{d}u = \int a(t)\,\mathrm{d}t \\
\Longrightarrow\quad &\ln u = A(t) + C_1 \quad \text{with} \quad A(t) = \int a(s)\,\mathrm{d}s, \quad C_1 \text{ integration constant} \\
\Longrightarrow\quad &u(t) = e^{A(t)+C_1} = Ce^{A(t)}.
\end{aligned} \tag{1.9}$$

Consequently, the solution of the IVP (1.8) with IC $u(\xi) = \eta$ is

$$u(t) = \eta e^{A(t)} \qquad \text{with} \quad A(t) = \int_\xi^t a(s)\,\mathrm{d}s\ .$$

To obtain the solution of the inhomogeneous problem, we use the method *"variation of constants"* that is based on the general solution of Eq (1.9). Therefore, the constant C is replaced by a function $C(t)$. This ansatz helps by finding a solution of the original problem in Eq (1.7), a so-called *particular solution*

$$\begin{aligned}
u(t) &= C(t)e^{A(t)}, \qquad & u'(t) &= C'(t)e^{A(t)} + C(t)A'(t)e^{A(t)}, \\
b(t) &= u'(t) - a(t)\,u = \left(C'(t) + C(t)A'(t) - a(t)C(t)\right)e^{A(t)}.
\end{aligned}$$

Since $A(t) = \int a(s)\,\mathrm{d}s$, we have $A'(t) = a(t)$ and thus

$$\begin{aligned}
C'(t) &= b(t)e^{-A(t)} \\
C(t) &= C_0 + \int b(s)e^{-A(s)}\,\mathrm{d}s\ , \qquad \text{where} \quad C_0 = C.
\end{aligned}$$

The solution of the IVP (1.7)

$$u'(t) = a(t)u + b(t), \qquad u(\xi) = \eta$$

is thus given by

$$\begin{aligned}
u(t) &= \left(\eta + \int_\xi^t b(s)e^{-A(s)}\,\mathrm{d}s\right)e^{A(t)} \\
&= \eta e^{A(t)} + \int_\xi^t b(s)e^{A(t)-A(s)}\,\mathrm{d}s
\end{aligned}$$

with $A(t) = \int_\xi^t a(\tau)\,\mathrm{d}\tau$ and in particular $A(t) - A(s) = \int_s^t a(\tau)\,\mathrm{d}\tau$.

Remark 1.10. *The solution u of an inhomogeneous linear ODE of first order can be always written as sum of the solution u_h of the homogeneous problem and the particular solution u_p of the inhomogeneous one*

$$u = u_h + u_p.$$

1.3.3 Linear Systems

To complete the section of analytically solving linear differential equations, we briefly recall the solution technique for linear systems of the form

$$u' = Au, \quad u(0) = u_0$$

with a diagonalizable matrix A. The following procedure describes how the ODE sytem can be solved within a few steps:

1. Diagonalize A, i.e. compute its independent eigenvalues λ_i and eigenvectors v_i. Then,
$$A = S\Lambda S^{-1},$$
where $\Lambda = \mathrm{diag}(\lambda_i)$ and the ith column of S is v_i.

2. Compute the matrix exponential of At, using the formula
$$\exp(At) = S\exp(\Lambda t)S^{-1},$$
where $\exp(\Lambda t) = \mathrm{diag}(e^{\lambda_i t})$.

3. Finally, the solution to the ODE is given by
$$u(t) = \exp(At)u_0.$$

If the matrix A only has a Jordan normal form, then you can follow the steps above. The only difference is that instead of the diagonal matrix we have the Jordan normal form. For the proof of this method we refer to standard ODE literature.

1.4 Existence, Uniqueness and Condition

Consider the following family of IVPs

$$\begin{aligned} \mathbf{x}'(t) &= \mathbf{f}(t, \mathbf{x}(t)), \quad a \le t \le b \\ \mathbf{x}(t_0) &= \mathbf{x_0}, \qquad\quad a \le t_0 \le b \end{aligned} \tag{1.10}$$

where the right-hand side is given by the vector-valued function

$$\mathbf{f}: \ [a,b] \times \mathbb{R}^d \ \to \ \mathbb{R}^d, \quad \mathbf{f} = \begin{pmatrix} f_1 \\ \vdots \\ f_d \end{pmatrix}.$$

The solution of Eq (1.10) has the form

$$\mathbf{x}: \ [a,b] \ \to \ \mathbb{R}^d, \quad \mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix}.$$

Definition 1.11. *The function* $\mathbf{f}: \ [a,b] \times \mathbb{R}^d \ \to \ \mathbb{R}^d$ *is called* globally Lipschitz–continuous (L–continuous) *with respect to* $\mathbf{x}$, *if*

$$\|\mathbf{f}(t,\mathbf{x_1}) - \mathbf{f}(t,\mathbf{x_2})\| \le L\|\mathbf{x_1} - \mathbf{x_2}\|, \quad \forall t \in [a,b], \quad \forall \mathbf{x_1}, \mathbf{x_2} \in \mathbb{R}^d \tag{1.11}$$

with the Lipschitz constant $L > 0$.

For globally L–continuous right-hand sides, the existence and uniqueness result by Picard–Lindelöf is valid:

Theorem 1.12 (Picard–Lindelöf's Existence and Uniqueness Theorem)**.** *Let* $\mathbf{f}$ *be globally* L*–continuous with respect to* $\mathbf{x}$. *Then there exists a unique solution* $\mathbf{x} \in \mathcal{C}^1([a,b];\mathbb{R}^d)$ *of the IVP (1.10) for each given* $\mathbf{x_0} \in \mathbb{R}^d$ *and* $a \le t_0 \le b$.

Proof: By help of Banach's Fixpoint Theorem. ■

Example 1.13. *For linear functions* $\mathbf{f}(t,\mathbf{x}) = \lambda\mathbf{x}$, *the Lipschitz constant is* $L = |\lambda|$.

Remark 1.14. *The function* $\mathbf{f}$ *often satisfies the Lipschitz condition (1.11) only locally, i.e. for* $\|\mathbf{x_1}\| \le const$, $\|\mathbf{x_2}\| \le const$. *Then it might happen that the solution does not exist on the whole interval* $[a,b]$, *but only on a subset* $[\alpha,\beta] \subset [a,b]$, $\alpha \le t_0 \le \beta$.

Example 1.15. *Let* $\mathbf{f}: [a,b] \times \mathbb{R} \to \mathbb{R}$

$$\mathbf{f}(t,\mathbf{x}) = \mathbf{x}^2, \qquad t_0 = 0,$$

then

$$|\mathbf{f}(t,\mathbf{x_1}) - \mathbf{f}(t,\mathbf{x_2})| \ = \ |(\mathbf{x_1^2} - \mathbf{x_2^2})| \ = \ \underbrace{|\mathbf{x_1} + \mathbf{x_2}|}_{=L} |\mathbf{x_1} - \mathbf{x_2}|.$$

Therefore, the Lipschitz constant satisfies

$$L = L(\mathbf{x_1},\mathbf{x_2}) \le const, \quad \textit{if} \ \ |\mathbf{x_1}|, |\mathbf{x_2}| \le const.$$

The local existence of the solution can be concluded from the following consideration: Let $\mathbf{x_0} > \mathbf{0}$*, then* $\mathbf{x}(t) > 0$ *holds for* $t > 0$ *because* $\mathbf{x}'(t) \geq 0$*. Integration yields*

$$\int_0^t \frac{x'(s)}{x(s)^2}\,\mathrm{d}s \;=\; \int_0^t 1\,\mathrm{d}s$$

and thus

$$t = \int_{x_0}^{x(t)} \frac{\mathrm{d}z}{z^2} = \frac{1}{x_0} - \frac{1}{x(t)} \quad \Longrightarrow \quad \mathbf{x}(t) = \frac{1}{\frac{1}{\mathbf{x_0}} - t}.$$

Consequently, the solution depends on the initial value and exists only for $t < 1/\mathbf{x_0}$*.*

For the further analysis we introduce the following useful notation. Let $\mathbf{x}$ be the solution of the IVP $\mathbf{x}' = \mathbf{f}(t, \mathbf{x})$, $\mathbf{x}(t_0) = \mathbf{x_0}$ on $[a, b]$. Then,

$$\mathbf{\Phi}^{t,t_0}\mathbf{x_0} = \mathbf{x}(t; t_0, \mathbf{x_0})$$

denotes the *evolution of the differential equation*.

For the evolution the following statements are valid:

- $\mathbf{\Phi}^{t_0,t_0}\,\mathbf{x_0} = \mathbf{x_0}$
- $\dfrac{\mathrm{d}}{\mathrm{d}\tau}\mathbf{\Phi}^{t+\tau,t}\mathbf{x}\Big|_{\tau=0} \;=\; \mathbf{f}(t, \mathbf{x})$
- $\mathbf{\Phi}^{t,\sigma}\mathbf{\Phi}^{\sigma,s}\mathbf{x} \;=\; \mathbf{\Phi}^{t,s}\mathbf{x}$

Remark 1.16. *Consider the autonomous system*

$$\mathbf{x}' = \mathbf{f}(\mathbf{x}), \quad \mathbf{x}(t_0) = \mathbf{x_0},$$

then the translation with $\tau \in \mathbb{R}$ *results in the solution* $\hat{\mathbf{x}}(t) = \mathbf{x}(t - \tau)$ *of the IVP*

$$\hat{\mathbf{x}}' = \mathbf{f}(\hat{\mathbf{x}}), \quad \hat{\mathbf{x}}(t_0 + \tau) = \mathbf{x_0}.$$

The starting (time) point plays no role for autonomous problems. Thus, we always choose $t_0 = 0$ *and* $a \leq 0 \leq b$*.*

Remark 1.17. *Let* $\mathbf{x}$ *be the solution of the autonomous IVP* $\mathbf{x}' = \mathbf{f}(\mathbf{x})$, $\mathbf{x}(0) = \mathbf{x_0}$ *on* $[a, b]$*. Then,*

$$\mathbf{\Phi}^t\mathbf{x_0} = \mathbf{x}(t; 0, \mathbf{x_0})$$

denotes the phase flux *of the differential equation and*

$$\boldsymbol{\gamma}(\mathbf{x_0}) = \{\mathbf{\Phi}^t\mathbf{x_0},\ t \in [a, b]\}$$

the trajectory *or* orbit *.*

We analyze now the behavior of the solution for small perturbations of the initial data. A problem is called *well-posed* if small perturbations on the initial data cause only small deviations in the solutions of the problem. Otherwise, it is said to be *ill-posed*.

Let $a = t_0$.

Lemma 1.18 (Gronwall)**.** *Let* $\chi : [a, b] \to \mathbb{R}$ *be continuous,* $\alpha, \beta > 0$ *and*

$$\chi(t) \leq \alpha + \beta \int_{t_0}^{t} \chi(s)\ \mathrm{d}s, \quad t_0 \leq t \leq b.$$

Then,

$$\chi(t) \leq \alpha e^{\beta(t-t_0)} \quad \forall t \in [t_0, b].$$

The Gronwall Lemma implies the following theorem about the dependency of the solution on the initial data (stability).

Theorem 1.19. *Let the assumptions of the Existence Theorem 1.12 (Picard–Lindelöf) be valid. Then, the solutions* $\mathbf{x}(t), \mathbf{y}(t)$ *of the IVP (1.10) with respect to the initial values* $\mathbf{x}(t_0) = \mathbf{x_0}$ *and* $\mathbf{y}(t_0) = \mathbf{y_0}$ *fulfill the following estimate*

$$\|\mathbf{x}(t) - \mathbf{y}(t)\| \leq e^{L(t-t_0)} \|\mathbf{x_0} - \mathbf{y_0}\|, \quad \forall t \in [t_0, b] \tag{1.12}$$

Proof: For all $t \in [t_0, b]$ we have:

$$\begin{aligned}
\|\mathbf{x}(t) - \mathbf{y}(t)\| &= \|\mathbf{x_0} - \mathbf{y_0} + \int_{t_0}^{t} \mathbf{f}(s, \mathbf{x}(s)) - \mathbf{f}(s, \mathbf{y}(s))\ \mathrm{d}s\| \\
&\leq \|\mathbf{x_0} - \mathbf{y_0}\| + \int_{t_0}^{t} \|\mathbf{f}(s, \mathbf{x}(s)) - \mathbf{f}(s, \mathbf{y}(s))\|\ \mathrm{d}s \quad \text{with triangle inequality} \\
&\leq \|\mathbf{x_0} - \mathbf{y_0}\| + L \int_{t_0}^{t} \|\mathbf{x}(s) - \mathbf{y}(s)\|\ \mathrm{d}s \qquad \text{with } L\text{–continuity.}
\end{aligned}$$

The estimate (1.12) follows now directly from the Gronwall Lemma. ■

Remark 1.20. *The estimate (1.12) cannot be improved, since the ODE* $\mathbf{x}' = \lambda \mathbf{x},\ \lambda > 0$ *gives*

$$\mathbf{x}(t) = \mathbf{x_0}\, e^{\lambda(t-t_0)} \quad \textit{and} \quad \|\mathbf{x}(t) - \mathbf{y}(t)\| = \|\mathbf{x_0} - \mathbf{y_0}\|\, e^{\lambda(t-t_0)}.$$

But in most cases it is too pessimistic, see for example $\mathbf{x}' = -\lambda \mathbf{x}\,,\ \lambda > 0$

$$\mathbf{x}(t) = \mathbf{x_0}\, e^{-\lambda(t-t_0)} \quad \textit{and} \quad \|\mathbf{x}(t) - \mathbf{y}(t)\| = \|\mathbf{x_0} - \mathbf{y_0}\|\, e^{-\lambda(t-t_0)} < \|\mathbf{x_0} - \mathbf{y_0}\|\, e^{\lambda(t-t_0)}.$$

Remark 1.21. *The number*

$$\max_{t \in [t_0, b]} e^{L(t-t_0)} = e^{L(b-t_0)}$$

is called condition *of the IVP. It describes the possible effect of a small perturbation for bounded time intervals* $t \in [t_0, b]$.

2 Explicit One-Step Methods

It is well-known that only a limited number of ODEs can be solved in a closed form. Consider for example the ODE $x' = \frac{x-t}{x+t}$ whose solution is only implicitly defined by $\frac{1}{2}\log(\left(\frac{x}{t}\right)^2+1)+\log(t)+\arctan(\frac{x}{t}) = c$, where c is a constant depending on the initial value. Therefore, we are now interested in methods which can be applied to any kind of ODE under the condition that the particular method admits a unique solution.

In this chapter we discuss the numerical solution of the IVP

$$\mathbf{x}'(t) = \mathbf{f}(t, \mathbf{x}(t)), \qquad \mathbf{x}(t_0) = \mathbf{x_0}.$$

A famous approximation of the derivative is the finite difference

$$\mathbf{f}(t, \mathbf{x}(t)) \approx \frac{\mathbf{x}(t+\tau) - \mathbf{x}(t)}{\tau}$$

converging towards the derivative for $\tau \to 0$. In particular, we use a polygon to approximate the IVP numerically that can be obtained in the following way.

Choose a grid $\Delta = \{t_0, \cdots, t_n;\, t_0 < t_1 < \cdots < t_n = T\}$ of the underlying interval $[t_0, T]$ and construct an approximation $\mathbf{x_\Delta}$ by means of the recursion:

$$\begin{aligned}
&\mathbf{x_\Delta}(t_0) = \mathbf{x_0}, \\
&\mathbf{x_\Delta}(t) = \mathbf{x_0} \quad + (t-t_0)\,\mathbf{f}(t_0, \mathbf{x_0}) && \text{for } \ t \in [t_0, t_1], \\
&\mathbf{x_\Delta}(t) = \mathbf{x_\Delta}(t_j) + (t-t_j)\,\mathbf{f}(t_j, \mathbf{x_\Delta}(t_j)) && \text{for } t \in [t_j, t_{j+1}],\ j = 0, \ldots, n-1.
\end{aligned}$$

Note that $\mathbf{f}(t, \mathbf{x_\Delta}(t_j)) = \mathbf{x_\Delta}'(t_j)$ so that we have here a Taylor expansion up to the linear term. Summing up, the function value $\mathbf{x_j} := \mathbf{x_\Delta}(t_j)$ is described by

$$\boxed{\mathbf{x_{j+1}} = \mathbf{x_j} + (t_{j+1} - t_j)\mathbf{f}(t_j, \mathbf{x_j})}$$

with $\mathbf{x_0}$ given. This iteration can be easily implemented using e.g. MATLAB. The method is called *explicit Euler method*.

Remark 2.1. *Explicit versus implicit methods, what is the difference?*
Explicit*:* $\mathbf{x_{j+1}}$ *can be directly computed in terms of the previous values* $\mathbf{x_j}$
Implicit*:* $\mathbf{x_{j+1}}$ *depends implicitly on itself through the right-hand side* $f(t_j, \mathbf{x_{j+1}})$

Implicit methods will be the topic of the next chapter.

Definition 2.2. *Let* $0 < \tau \ll 1$, $k \in \mathbb{R}^+$. *A scalar-valued function* g *is* $\mathcal{O}(\tau^k)$, "large O of τ^k", *if*

$$|g(\tau)| \le C\left|\tau^k\right| \qquad \text{with constant } C > 0.$$

A scalar-valued function g *is* $o(\tau^k)$, "little o of τ^k", *if*

$$\lim_{\tau \to 0} \left| \frac{g(\tau)}{\tau^k} \right| \to 0.$$

For vector-valued functions $\mathbf{g}$ *the definitions above hold componentwise.*

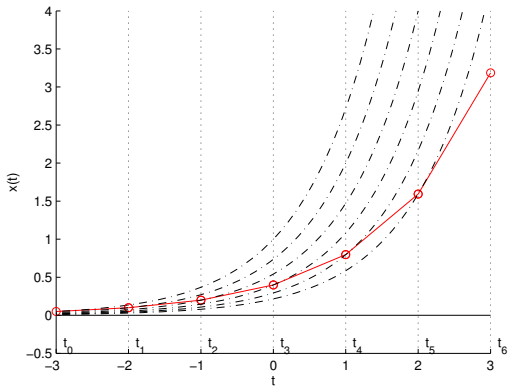


Figure 2.1: The Euler polygon method.

If $\mathbf{x}_{\boldsymbol{\Delta}}(t_j) = \mathbf{x}(t_j)$ is valid, then the error in the $(j+1)$ th iteration step can be computed on basis of the Taylor expansion. With $\tau_j := t_{j+1} - t_j$, we obtain

$$\mathbf{x}(t_{j+1}) = \mathbf{x}(t_j) + (t_{j+1} - t_j)\mathbf{x}'(t_j) + \mathcal{O}(\tau_j^2).$$

Hence,

$$\mathbf{f}(t_j, \mathbf{x}(t_j)) = \mathbf{x}'(t_j) = \frac{\mathbf{x}(t_{j+1}) - \mathbf{x}(t_j)}{t_{j+1} - t_j} + \mathcal{O}(\tau_j) \tag{2.1}$$

and substitution and multiplication with τ_j results in an error of $\mathcal{O}(\tau_j^2)$

$$\mathbf{x}_{\boldsymbol{\Delta}}(t_{j+1}) = \mathbf{x}_{\boldsymbol{\Delta}}(t_j) + (t_{j+1} - t_j)\mathbf{f}(t_j, \mathbf{x}_{\boldsymbol{\Delta}}(t_j)) = \mathbf{x}(t_{j+1}) + \mathcal{O}(\tau_j^2).$$

In the following we use the general notation for the discrete evolution

$$\mathbf{x}_{\boldsymbol{\Delta}}(t_{j+1}) = \boldsymbol{\Psi}^{t_j+\tau_j, t_j}\mathbf{x}_{\boldsymbol{\Delta}}(t_j).$$

2.1 Consistency and Convergence

The quality of the numerical method depends obviously on the error between exact and the approximate solution. In this section we estimate the arising error. Therefore, we define an arbitrary grid $\Delta = \{t_0, \cdots, t_n;\, t_0 < t_1 < \cdots < t_n = T\}$ for the underlying interval $[t_0, T]$. Let the step size be given by $\tau_j = t_{j+1} - t_j,\ j = 0, \ldots, n-1$, and the maximal one by

$$\tau_\Delta = \max_{0 \le j < n} \tau_j.$$

The task we have to deal with is now:

Design $\mathbf{x}_{\boldsymbol{\Delta}} : \Delta \to \mathbb{R}^d$ such that it is a good approximation of the solution $\mathbf{x}(t)$ of the IVP,

$$\mathbf{x}_{\boldsymbol{\Delta}}(t) \approx \mathbf{x}(t), \quad t \in \Delta.$$

Thereby, the function $\mathbf{x}_{\boldsymbol{\Delta}}$ should be recursively computed, i.e.

$$\mathbf{x}_{\boldsymbol{\Delta}}(t_0) = \mathbf{x_0} \to \mathbf{x}_{\boldsymbol{\Delta}}(t_1) = \mathbf{x_1} \to \cdots \to \mathbf{x}_{\boldsymbol{\Delta}}(t_n) = \mathbf{x}_{\boldsymbol{\Delta}}(T) = \mathbf{x_n}.$$

Definition 2.3. *A numerical method is called* one-step method *if* $\mathbf{x}_{\boldsymbol{\Delta}}(t_{j+1})$ *depends only on* $\mathbf{x}_{\boldsymbol{\Delta}}(t_j)$. *Otherwise the scheme is called* multi-step method. *The computational procedure for all grids* Δ *is described by a two-term relation:*

1. $\mathbf{x}_{\boldsymbol{\Delta}}(t_0) = \mathbf{x_0}$
2. $\mathbf{x}_{\boldsymbol{\Delta}}(t_{j+1}) = \underbrace{\boldsymbol{\Psi}^{t_{j+1},t_j}}_{=\boldsymbol{\Psi}^{t_j+\tau_j,t_j}} \mathbf{x}_{\boldsymbol{\Delta}}(t_j), \quad j = 0,\ldots,n-1$ *with the discrete evolution* $\boldsymbol{\Psi}^{t+\tau,t}$ *being independent of* Δ.

Recall the following notations since they become important in the derivation of the error estimates:

- continuous case: $\mathbf{x}(t+\tau) = \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x}(t)$, $\boldsymbol{\Phi}^{t+\tau,t}$ (continuous) evolution
- discrete case: $\mathbf{x}_{\boldsymbol{\Delta}}(t+\tau) = \boldsymbol{\Psi}^{t+\tau,t}\mathbf{x}_{\boldsymbol{\Delta}}(t)$, $\boldsymbol{\Psi}^{t+\tau,t}$ discrete evolution

Definition 2.4. *The difference*

$$\boldsymbol{\epsilon}(t,\mathbf{x},\tau) = \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \boldsymbol{\Psi}^{t+\tau,t}\mathbf{x}$$

is called consistency error.

Lemma 2.5. *Let the discrete evolution* $\boldsymbol{\Psi}^{t+\tau,t}\mathbf{x}$ *be continuously differentiable with respect to* τ. *Then the following three statements are equivalent:*

1. $\boldsymbol{\Psi}^{t,t}\mathbf{x} = \mathbf{x}$ *and*
$$\frac{\mathrm{d}}{\mathrm{d}\tau}\boldsymbol{\Psi}^{t+\tau,t}\mathbf{x}\Big|_{\tau=0} = \mathbf{f}(t,\mathbf{x})$$
hold, cf. with the continuous evolution $\dfrac{\mathrm{d}}{\mathrm{d}\tau}\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x}\Big|_{\tau=0} = \mathbf{f}(t,\mathbf{x})$.
2. *The discrete evolution has the representation*
$$\boldsymbol{\Psi}^{t+\tau,t}\mathbf{x} = \mathbf{x} + \tau\boldsymbol{\Psi}(t,\mathbf{x},\tau)$$
with the increment function $\boldsymbol{\Psi}$ *satisfying*
$$\boldsymbol{\Psi}(t,\mathbf{x},0) = \lim_{\tau\to 0}\boldsymbol{\Psi}(t,\mathbf{x},\tau) = \mathbf{f}(t,\mathbf{x}).$$

3. The consistency error reads

$$\boldsymbol{\epsilon}(t, \mathbf{x}, \tau) = \mathcal{O}(\tau), \quad \tau \to 0.$$

If the discrete evolution (resp. the increment function) satisfies this property, it is called consistent*.*

Proof:

1. (1) $\to$ (2): It holds that

$$\boldsymbol{\Psi}^{t+\tau,t}\mathbf{x} = \mathbf{x} + \tau \underbrace{\int_0^1 \frac{d}{\mathrm{d}\sigma}\boldsymbol{\Psi}^{t+\sigma,t}\mathbf{x}\Big|_{\sigma=s\tau} \mathrm{d}s}_{:=\boldsymbol{\Psi}(t,\mathbf{x},\tau)}$$

   Further, using (1), we get

$$\boldsymbol{\Psi}(t, \mathbf{x}, \tau) = \frac{\boldsymbol{\Psi}^{t+\tau,t}\mathbf{x} - \mathbf{x}}{\tau} \xrightarrow{\tau\downarrow 0} \frac{d}{\mathrm{d}\tau}\boldsymbol{\Psi}^{t+\tau,t}\mathbf{x}\Big|_{\tau=0} \overset{(1)}{=} \mathbf{f}(t, \mathbf{x})$$

2. (2) $\to$ (3): Further, it follows from (2) that

$$\begin{aligned}\frac{\boldsymbol{\epsilon}(t, \mathbf{x}, \tau)}{\tau} &= \frac{1}{\tau}\Big(\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \boldsymbol{\Psi}^{t+\tau,t}\mathbf{x}\Big) \\ &= \frac{1}{\tau}\Big(\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \mathbf{x} - \tau\boldsymbol{\Psi}(t, \mathbf{x}, \tau)\Big) \\ &= \frac{1}{\tau}\Big(\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \mathbf{x}\Big) - \boldsymbol{\Psi}(t, \mathbf{x}, \tau)\end{aligned}$$

$$\xrightarrow[\boldsymbol{\Psi} \text{ continuous}]{\tau\downarrow 0} \mathbf{f}(t, \mathbf{x}) - \boldsymbol{\Psi}(t, \mathbf{x}, 0) = 0$$

3. (3) $\to$ (1):

$$\begin{aligned}0 = \lim_{\tau\downarrow 0} \frac{\boldsymbol{\epsilon}(t, \mathbf{x}, \tau)}{\tau} &= \lim_{\tau\downarrow 0} \frac{\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \mathbf{x} - (\boldsymbol{\Psi}^{t+\tau,t}\mathbf{x} - \mathbf{x})}{\tau} \\ &= \mathbf{f}(t, \mathbf{x}) - \frac{d}{\mathrm{d}\tau}\boldsymbol{\Psi}^{t+\tau,t}\mathbf{x}\Big|_{\tau=0}\end{aligned}$$

■

Example 2.6. *Choose the increment function* $\boldsymbol{\Psi}(t, \mathbf{x}, \tau) = \mathbf{f}(t, \mathbf{x})$

$$\begin{aligned}&\Rightarrow \quad \boldsymbol{\Psi}^{t+\tau,t}\mathbf{x} = \mathbf{x} + \tau\,\mathbf{f}(t, \mathbf{x}) \\ &\Rightarrow \quad \mathbf{x}_{\boldsymbol{\Delta}}(t_{j+1}) = \boldsymbol{\Psi}^{t_j+\tau_j,t_j}\,\mathbf{x}_{\boldsymbol{\Delta}}(t_j) = \mathbf{x}_{\boldsymbol{\Delta}}(t_j) + \tau_j\,\mathbf{f}(t_j, \mathbf{x}_{\boldsymbol{\Delta}}(t_j)),\end{aligned}$$

This yields the explicit Euler method.

Now, we analyze the convergence of the discrete approximation $\mathbf{x}_{\boldsymbol{\Delta}}(t)$ towards the continuous solution $\mathbf{x}(t)$. For this purpose, we define the *error (grid error)*

$$\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t) := \mathbf{x}(t) - \mathbf{x}_{\boldsymbol{\Delta}}(t), \qquad t \in \Delta$$

and the *discretization* resp. *truncation error*

$$\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}\|_\infty := \max_{t\in\Delta} \|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t)\|.$$

Definition 2.7. *Let $\mathbf{x}_{\boldsymbol{\Delta}}$ be given for any arbitrary grid Δ. Then, $\mathbf{x}_{\boldsymbol{\Delta}}$ is said to* converge *towards $\mathbf{x}$, if*

$$\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}\|_\infty \to 0 \quad \textit{for } \tau_\Delta \to 0.$$

The convergence is of order *$p > 0$, if*

$$\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}\|_\infty = \mathcal{O}(\tau_\Delta^p) \quad \textit{for } \tau_\Delta \to 0.$$

Remark 2.8. *The consistency error $\boldsymbol{\epsilon}$ in Def 2.4 describes only the error that arises due to a single iteration step. In contrast, $\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}$ describes the error of the whole approximation at any time t.*

Definition 2.9. *The discrete evolution has* consistency order *p, if*

$$\boldsymbol{\epsilon}(t, \mathbf{x}, \tau) = \mathcal{O}(\tau^{p+1}), \quad \textit{for } \tau \to 0.$$

Example 2.10. *The Euler method fulfills*

$$\boldsymbol{\epsilon}(t, \mathbf{x}, \tau) = \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} \underbrace{-\mathbf{x} - \tau\mathbf{f}(t, \mathbf{x})}_{-\boldsymbol{\Psi}^{t+\tau,t}\mathbf{x}}.$$

To estimate the error we use

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} = \mathbf{f}(t+\tau, \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x})$$

and

$$\frac{\mathrm{d}^2}{\mathrm{d}\tau^2}\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} = \partial_1\mathbf{f}(t+\tau, \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x}) + \partial_2\mathbf{f}(t+\tau, \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x}) \cdot \underbrace{\frac{\mathrm{d}}{\mathrm{d}\tau}\boldsymbol{\Phi}^{t+\tau}\mathbf{x}}_{=\mathbf{f}(t+\tau,\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x})}.$$

Then, we get

$$\begin{aligned}
\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \mathbf{x} &= \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \boldsymbol{\Phi}^{t,t}\mathbf{x} \\
&= \tau\frac{\mathrm{d}}{\mathrm{d}\tau}\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x}\Big|_{\tau=0} + \mathcal{O}(\tau^2) \qquad \textit{with Taylor expansion} \\
&= \tau\mathbf{f}(t, \mathbf{x}) + \mathcal{O}(\tau^2)
\end{aligned}$$

$$\begin{aligned}
&\Longrightarrow \quad \boldsymbol{\epsilon}(t, \mathbf{x}, \tau) = \mathcal{O}(\tau^2). \\
&\Longrightarrow \quad \textit{consistency order } p = 1
\end{aligned}$$

Theorem 2.11 (Convergence Theorem)**.** *Let the assumptions of Th 1.12 (Picard–Lindelöf) be fulfilled and* $\mathbf{x}(t)$ *be the solution of the IVP* $\mathbf{x}' = \mathbf{f}(t, \mathbf{x}),\ \mathbf{x}(t_0) = \mathbf{x_0}$ *. Let the discrete evolution* $\mathbf{\Psi}^{t+\tau,t}$ *be consistent of order* $p > 0$ *. Moreover, let the increment function* $\mathbf{\Psi}$ *be globally Lipschitz–continuous in* $\mathbf{x}$ *(Stability). Then, the grid function* $\mathbf{x_\Delta}$ *converges towards* $\mathbf{x}$ *with order* p *for* $\tau_\Delta \to 0$ *.*

Proof: The error in the grid point t_{j+1} can be split according to

$$\begin{aligned}
\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_{j+1}) &= \mathbf{x}(t_{j+1}) - \mathbf{x}_{\boldsymbol{\Delta}}(t_{j+1}) \\
&= \mathbf{\Phi}^{t_{j+1},t_j}\mathbf{x}(t_j) - \mathbf{\Psi}^{t_{j+1},t_j}\mathbf{x}_{\boldsymbol{\Delta}}(t_j) \\
&= \mathbf{\Phi}^{t_{j+1},t_j}\mathbf{x}(t_j) - \mathbf{\Psi}^{t_{j+1},t_j}\mathbf{x}(t_j) + \mathbf{\Psi}^{t_{j+1},t_j}\mathbf{x}(t_j) - \mathbf{\Psi}^{t_{j+1},t_j}\mathbf{x}_{\boldsymbol{\Delta}}(t_j) \\
&= \boldsymbol{\epsilon}(t_j, \mathbf{x}(t_j), \tau_j) + \boldsymbol{\epsilon_j}
\end{aligned}$$

with

$$\boldsymbol{\epsilon_j} = \mathbf{\Psi}^{t_{j+1},t_j}\mathbf{x}(t_j) - \mathbf{\Psi}^{t_{j+1},t_j}\mathbf{x}_{\boldsymbol{\Delta}}(t_j)$$

describing the propagation of the error due to the discrete evolution. Since

$$\boldsymbol{\epsilon_j} = \mathbf{x}(t_j) - \mathbf{x}_{\boldsymbol{\Delta}}(t_j) + \tau_j\Big(\mathbf{\Psi}(t_j, \mathbf{x}(t_j), \tau_j) - \mathbf{\Psi}(t_j, \mathbf{x}_{\boldsymbol{\Delta}}(t_j), \tau_j)\Big),$$

$$\|\boldsymbol{\epsilon_j}\| \le \|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_j)\| + \tau_j L\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_j)\| = (1 + \tau_j L)\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_j)\|, \tag{2.2}$$

holds because of the L–continuity of $\mathbf{\Psi}$ with respect to $\mathbf{x}$.

The error estimate

$$\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_j)\| \le \frac{C}{L}\tau_\Delta^p(e^{L(t_j - t_0)} - 1) \qquad \forall t_j \in \Delta$$

with Lipschitz $L > 0$ and consistency constant $C \ge 0$ is now proved by induction with respect to j:

- $j = 0$: $\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_0)\| = 0$.
- For j the estimate is assumed to be valid.
- $j \to j + 1$:

$$\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_{j+1})\| = \boldsymbol{\epsilon}(t_j, \mathbf{x}(t_j), \tau_j) + \boldsymbol{\epsilon_j} \le \underbrace{C\tau_j^{p+1}}_{\text{consistency}} + (1 + \tau_j L)\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_j)\| \quad j = 0, \ldots, n-1$$

according to Eq (2.2). Applying the induction hypothesis results in

$$\begin{aligned}
\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_{j+1})\| &\le C\tau_j^{p+1} + (1 + \tau_j L)\tau_\Delta^p \frac{C}{L}(e^{L(t_j - t_0)} - 1) \\
&\le \tau_\Delta^p \frac{C}{L}\left(\tau_j L + (1 + \tau_j L)\ (e^{L(t_j - t_0)} - 1)\right) \\
&= \tau_\Delta^p \frac{C}{L}((1 + \tau_j L)e^{L(t_j - t_0)} - 1)
\end{aligned}$$

From $(1+\tau_j L) \leq e^{\tau_j L}$, it follows $(1+\tau_j L)e^{L(t_j-t_0)} \leq e^{L(t_{j+1}-t_j)}e^{L(t_j-t_0)} = e^{L(t_{j+1}-t_0)}$ and thus the estimate

$$\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_{j+1})\| \leq \tau_\Delta^p \frac{C}{L}(e^{L(t_{j+1}-t_0)} - 1).$$

Then, the discretization error satisfies for $t_0 \leq t \leq T$

$$\|\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}\|_\infty \leq \tau_\Delta^p \frac{C}{L}(e^{L(T-t_0)} - 1) = \mathcal{O}(\tau_\Delta^p)$$

which concludes the proof. ■

Remark 2.12.

$$\boxed{\textit{Consistency order } p \implies \textit{convergence order } p}$$

Example 2.13. *Consider the explicit Euler method with* $\mathbf{f}(t,\mathbf{x})$ *L–continuous in* $\mathbf{x}$.

$\Rightarrow$ $\boldsymbol{\Psi}(t,\mathbf{x},\tau) = \mathbf{f}(t,\mathbf{x})$ *fulfills the assumptions of Th 2.11, and the method is consistent of order* $p=1$.

$\Rightarrow$ *The Euler method converges towards the continuous solution with* $\mathcal{O}(\tau_\Delta)$.

Remark 2.14. *Two basic problems might arise by the application of the Euler method:*

1. *From the convergence order* $\mathcal{O}(\tau_\Delta)$ *it can be concluded that halving the error requires halving the step size and thus doubling the number of grid points. This leads to the doubling of the computational costs which might cause an enormous effort. Thus, the goal of the following section is the construction of higher consistency order methods.*

2. *A further problem occurs by so-called* stiff *differential equations, e.g.*

$$x'(t) = -\lambda x(t), \quad \lambda \gg 1, \qquad x(0) = 1.$$

that has the solution

$$x(t) = e^{-\lambda t} \to 0, \quad t \to \infty.$$

The Euler method yields

$$\begin{aligned} \boldsymbol{\Psi}^{t+\tau,t} &= \mathbf{x} + \tau \mathbf{f}(t,\mathbf{x}) \\ \implies \qquad \mathbf{x_{j+1}} &= \mathbf{x_j} + \tau_j \mathbf{f}(t_j,\mathbf{x_j}), \quad \mathbf{x_j} = \mathbf{x_\Delta}(t_j) \\ &= (1-\tau\lambda)\mathbf{x_j}, \quad \textit{if } \tau_j = \tau, \ \forall j \end{aligned}$$

Recursively,

$$\mathbf{x_j} = (1-\tau\lambda)^j \mathbf{x_0} = (1-\tau\lambda)^j$$

If $|1-\tau\lambda| > 1$, *then* $|\mathbf{x_j}| \overset{j\to\infty}{\longrightarrow} \infty$ *which does obviously not approximate the exact solution, since* $x(t) \overset{t\to\infty}{\longrightarrow} 0$. *In contrast, the assumption* $|1-\tau\lambda| < 1$ *or* $1-\tau\lambda > -1$ *results in a correct approximation. However, it implies a restriction on the choice of the step size:*

$$\tau\lambda < 2 \quad or \quad \tau < 2/\lambda. \tag{2.3}$$

In particular for $\lambda \gg 1$, *Eq* (2.3) *formulates a very strong restriction on* τ. *The problem is stiff for* $\lambda \gg 1$. *Its reasonable handling will be discussed in Ch 3.*

2.2 Taylor Methods

The explicit Euler method exemplifies a powerful tool of the Taylor expansion for consistency and convergence results. Based upon Taylor expansions, we construct now further one-step methods with higher consistency order $p > 1$.

In general, the consistency error reads

$$\begin{aligned}\boldsymbol{\epsilon}(t,\mathbf{x},\tau) &= \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \boldsymbol{\Psi}^{t+\tau,t}\mathbf{x} \\ &= \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \mathbf{x} - \tau\boldsymbol{\Psi}(t,\mathbf{x},\tau).\end{aligned}$$

Using

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} = \mathbf{f}(t+\tau, \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x})$$

and

$$\frac{\mathrm{d}^2}{\mathrm{d}\tau^2}\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} = \partial_1\mathbf{f}(t+\tau, \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x}) + \partial_2\mathbf{f}(t+\tau, \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x}) \cdot \mathbf{f}(t+\tau, \boldsymbol{\Phi}^{t+\tau,t}\mathbf{x})$$

Taylor expansion with respect to τ gives the following first order approximation

$$\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \mathbf{x} = \tau\,\mathbf{f}(t,\mathbf{x}) + \mathcal{O}(\tau^2)$$

where $\mathbf{f}(t,\mathbf{x}) = \mathbf{x}'(t)$. The choice of the increment function $\boldsymbol{\Psi} = \mathbf{f}$ eliminates the term of order τ and leads consequently to the Euler method with $p = 1$, i.e.

$$\boldsymbol{\epsilon}(t,\mathbf{x},\tau) = \mathcal{O}(\tau^2).$$

This procedure might be extended up to higher orders:

$$\boldsymbol{\Phi}^{t+\tau,t}\mathbf{x} - \mathbf{x} = \tau\mathbf{f}(t,\mathbf{x}) + \frac{\tau^2}{2}\left(\partial_1\mathbf{f}(t,\mathbf{x}) + \partial_2\mathbf{f}(t,\mathbf{x}) \cdot \mathbf{f}(t,\mathbf{x})\right) + \mathcal{O}(\tau^3).$$

The choice of the increment function

$$\boldsymbol{\Psi}(t,\mathbf{x},\tau) = \mathbf{f}(t,\mathbf{x}) + \frac{\tau}{2}(\partial_1\mathbf{f}(t,\mathbf{x}) + \partial_2\mathbf{f}(t,\mathbf{x}) \cdot \mathbf{f}(t,\mathbf{x})) =: \boldsymbol{\Psi}^*(t,\mathbf{x},\tau) \tag{2.4}$$

gives then a method with consistency order $p = 2$. Analogously, methods of higher orders p might be constructed by further expansions. However, this construction rule is quite complicated, since higher derivatives need to be computed. A simpler alternative is described in the following section.

2.3 Runge–Kutta Methods

Constructing one-step methods of higher order, the following procedure turns out to be more efficient than the Taylor methods.

Choose the increment function $\mathbf{\Psi}$ as

$$\mathbf{\Psi}(t, \mathbf{x}, \tau) = \mathbf{\Psi}^*(t, \mathbf{x}, \tau) + \mathcal{O}(\tau^p).$$

Then the consistency order p of the method certainly remains.

Let us firstly consider the case $p = 2$ and make the ansatz

$$\mathbf{\Psi}(t, \mathbf{x}, \tau) = \mathbf{f}(t + c_2\tau, \mathbf{x} + \tau a_{21}\mathbf{f}(t, \mathbf{x})), \quad c_2, a_{21} \in \mathbb{R}.$$

Taylor expansion with respect to τ leads to

$$\mathbf{\Psi}(t, \mathbf{x}, \tau) = \mathbf{f}(t, \mathbf{x}) + c_2\tau\partial_1\mathbf{f}(t, \mathbf{x}) + \tau a_{21}\partial_2\mathbf{f}(t, \mathbf{x}) \cdot \mathbf{f}(t, \mathbf{x}) + \mathcal{O}(\tau^2).$$

Comparing $\mathbf{\Psi}$ with $\mathbf{\Psi}^*$ of Eq (2.4) to get $\mathbf{\Psi} = \mathbf{\Psi}^* + \mathcal{O}(\tau^2)$ yields the coefficients

$$c_2 = \frac{1}{2}, \quad a_{21} = \frac{1}{2}$$

and thus

$$\mathbf{\Psi}(t, \mathbf{x}, \tau) = \mathbf{f}\left(t + \frac{\tau}{2}, \mathbf{x} + \frac{\tau}{2}\mathbf{f}(t, \mathbf{x})\right).$$

This is the *Runge method*. It has consistency order $p = 2$. Recursively, the method can be written as

$$\begin{aligned} \mathbf{k_1} &= \mathbf{f}(t, \mathbf{x}), \\ \mathbf{k_2} &= \mathbf{f}(t + \tfrac{\tau}{2}, \mathbf{x} + \tfrac{\tau}{2}\mathbf{k_1}) = \mathbf{f}(t + c_2\tau, \mathbf{x} + \tau a_{21}\mathbf{k_1}) \\ \mathbf{\Psi}^{t+\tau,t}\mathbf{x} &= \mathbf{x} + \tau\mathbf{k_2} \end{aligned}$$

The generalization $p > 2$ leads to the *explicit Runge–Kutta methods (RK-methods) of stage s*. Therefore, the following ansatz is used:

1. $\mathbf{k_1} = \mathbf{f}(t + c_1\tau, \mathbf{x})$
2. $\mathbf{k_i} = \mathbf{f}(t + c_i\tau, \mathbf{x} + \tau \sum_{j=1}^{i-1} a_{ij}\mathbf{k_j}), \quad i = 2, \ldots, s$
3. $\mathbf{\Psi}^{t+\tau,t}\mathbf{x} = \mathbf{x} + \tau \sum_{i=1}^{s} b_i\mathbf{k_i}$

   with coefficients $c_i, a_{ij}, b_i \in \mathbb{R}$.

The $\mathbf{k_i}$, $i = 1, \dots, s$ are called *stages* of the method. For a better memorizing and efficient data transfer the coefficients are collected in the *Butcher array*:

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

$$\mathbf{b} = (b_1, \dots, b_s)^T \in \mathbb{R}^s, \qquad \mathbf{c} = (c_1, \dots, c_s)^T \in \mathbb{R}^s,$$

$$\mathbf{A} = \begin{bmatrix} 0 & & & & \\ a_{2,1} & 0 & & 0 & \\ \vdots & a_{3,2} & \ddots & & \\ \vdots & \vdots & \ddots & \ddots & \\ a_{s,1} & \cdots & \cdots & a_{s,s-1} & 0 \end{bmatrix} \in \mathbb{R}^{s \times s}$$

Example 2.15.

- *Explicit Euler method: stages* $s = 1$, *consistency order* $p = 1$

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

- *Runge method: stages* $s = 2$, *consistency order* $p = 2$

$$\begin{array}{c|cc} 0 & 0 & \\ \frac{1}{2} & \frac{1}{2} & 0 \\ \hline & 0 & 1 \end{array} \qquad \begin{array}{c|cc} c_1 & 0 & \\ c_2 & a_{21} & 0 \\ \hline & b_1 & b_2 \end{array}$$

Are we able to choose the coefficients in such a way that we obtain a method of consistency order $p > 2$?

Before answering the question let us firstly analyze the consistency of the RK-methods.

Lemma 2.16. *An explicit RK-method* $(\mathbf{b}, \mathbf{c}, \mathbf{A})$ *is consistent if and only if*

$$\sum_{i=1}^{s} b_i = 1.$$

Proof: Due to the definition of explicit RK-methods we consider

$$\mathbf{\Psi}^{t+\tau,t}\mathbf{x} = \mathbf{x} + \tau \mathbf{\Psi}(t, \mathbf{x}, \tau), \quad \mathbf{\Psi}(t, \mathbf{x}, \tau) = \sum_{i=1}^{s} b_i \mathbf{k_i}, \quad \mathbf{k_i} = \mathbf{k_i}(t, \mathbf{x}, \tau).$$

Lemma 2.5 (ii) ensures consistency if

$$\boldsymbol{\Psi}(t, \mathbf{x}, 0) = \mathbf{f}(t, \mathbf{x}).$$

Hence, it holds

$$\begin{aligned}\boldsymbol{\Psi}(t, \mathbf{x}, 0) = \sum_{i=1}^{s} b_i \mathbf{k_i}(t, \mathbf{x}, 0) &= \Big(\sum_{i=1}^{s} b_i \Big) \mathbf{f}(t, \mathbf{x}) \\ &= \mathbf{f}(t, \mathbf{x}).\end{aligned}$$

■

Methods of higher consistency order can be only obtained by increasing the number of stages.

Lemma 2.17. *Let an explicit RK-method with s stages be consistent of order p for a smooth right-hand side $f \in \mathcal{C}^\infty$. Then: $p \leq s$.*

Proof: Consider the IVP $x' = x,\ x(0) = 1$. Then we know that

$$\boldsymbol{\Phi}^{\tau,0} 1 = e^\tau = 1 + \tau + \frac{\tau^2}{2} + \cdots + \frac{\tau^p}{p!} + \mathcal{O}(\tau^{p+1}).$$

Moreover, $\mathbf{f}(t, \mathbf{x}) = \mathbf{x}$ implies that

$$\mathbf{k_i}(0, 1, \tau) = 1 + \tau \sum_{j=1}^{i-1} a_{ij} \mathbf{k_j}(0, 1, \tau),$$

$\mathbf{k_1}(0, 1, \tau) = 1$ and hence

$$\mathbf{k_i}(\overset{t}{0}, \overset{x}{1}, \overset{\tau}{\cdot}) \in P_{i-1}, \qquad i = 1, \ldots, s,$$

i.e. $\boldsymbol{\Psi}^{\tau,0} 1$ is a polynomial in τ of degree $\leq s$, since

$$\boldsymbol{\Psi}^{\tau,0} 1 = 1 + \tau \sum_{i=1}^{s} b_i \mathbf{k_i}(0, 1, \tau).$$

For the consistency error to satisfy

$$\boldsymbol{\epsilon}(0, 1, \tau) = \boldsymbol{\Phi}^{\tau,0} 1 - \boldsymbol{\Psi}^{\tau,0} 1 = \mathcal{O}(\tau^{p+1})$$

we must have $s \geq p$. ■

Let us simplify the family of RK-methods by focusing on their behavior for autonomous systems. Note that this is no restriction, since

$$\mathbf{x}' = \mathbf{f}(t, \mathbf{x}), \qquad \mathbf{x}(t_0) = \mathbf{x}_0.$$

is equivalent to

$$\begin{pmatrix} \mathbf{x} \\ t \end{pmatrix}' = \begin{pmatrix} \mathbf{f}(t, \mathbf{x}) \\ 1 \end{pmatrix}, \qquad \begin{pmatrix} \mathbf{x}(0) \\ t(0) \end{pmatrix} = \begin{pmatrix} \mathbf{x}_0 \\ t_0 \end{pmatrix}.$$

Let $\hat{\mathbf{\Phi}}$ be the extended (continuous) evolution, then

$$\hat{\mathbf{\Phi}}^{t+\tau,t} \begin{pmatrix} \mathbf{x} \\ t \end{pmatrix} = \begin{pmatrix} \mathbf{\Phi}^{t+\tau,t}\mathbf{x} \\ t+\tau \end{pmatrix},$$

with the original evolution $\mathbf{\Phi}$. We are now only interested in RK-methods that yield the same results for the original and the autonomous system. Define therefore the extended discrete evolution $\hat{\mathbf{\Psi}}$ to the original one $\mathbf{\Psi}$. A method is said to be *invariant with respect to autonomization* if

$$\begin{pmatrix} \mathbf{\Psi}^{t+\tau,t}\mathbf{x} \\ t+\tau \end{pmatrix} \stackrel{!}{=} \hat{\mathbf{\Psi}}^{t+\tau,t} \begin{pmatrix} \mathbf{x} \\ t \end{pmatrix}.$$

This information can be also taken from the Butcher array.

Lemma 2.18. *An explicit RK-method* $(\mathbf{b}, \mathbf{c}, \mathbf{A})$ *is invariant with respect to autonomization if and only if it is consistent with* $c_1 = 0$ *and*

$$c_i = \sum_{j=1}^{i-1} a_{ij}, \qquad i = 2, \ldots, s.$$

Proof: Let $\begin{pmatrix} \hat{\mathbf{k_i}} \\ \theta_i \end{pmatrix}$ denote the stages of the RK-method applied to the autonomous problem:

$$\hat{\mathbf{\Psi}}^{t+\tau,t} \begin{pmatrix} \mathbf{x} \\ t \end{pmatrix} = \begin{pmatrix} \mathbf{x} + \tau \sum_{i=1}^{s} b_i \hat{\mathbf{k_i}} \\ t + \tau \sum_{i=1}^{s} b_i \theta_i \end{pmatrix},$$

where

$$\begin{pmatrix} \hat{\mathbf{k_i}} \\ \theta_i \end{pmatrix} = \begin{pmatrix} \mathbf{f}(t + \tau \sum_{j=1}^{i-1} a_{ij}\theta_j \ , \ \mathbf{x} + \tau \sum_{j=1}^{i-1} a_{ij} \hat{\mathbf{k_j}}) \\ 1 \end{pmatrix} \quad i = 1, \ldots, s.$$

This leads exactly ($\theta_i = 1$) to $\mathbf{k_i} = \hat{\mathbf{k_i}}$ and $t + \tau \sum_{i=1}^{s} b_i \theta_i = t + \tau$ if $c_1 = 0$

$$\sum_{j=1}^{i-1} a_{ij} = c_i \quad \text{and} \quad \sum_{i=1}^{s} b_i = 1.$$

■

In the following we restrict on autonomous problems and methods that are invariant with respect to autonomization. Then methods are prescribed by $(\mathbf{b}, \mathbf{A})$ where $c_i = \sum_{j=1}^{i-1} a_{ij}$, $c_1 = 0$ is exclusively used as notation. Consequently, we consider IVPs of the form

$$\mathbf{x}' = \mathbf{f}(\mathbf{x}), \quad \mathbf{x}(0) = \mathbf{x_0}$$

and methods of the form

1. $\mathbf{k_1} = \mathbf{f}(\mathbf{x})$
2. $\mathbf{k_i} = \mathbf{f}(\mathbf{x} + \tau \sum_{j=1}^{i-1} a_{ij}\mathbf{k_j}), \quad i = 2, \ldots, s$
3. $\mathbf{\Psi}^\tau \mathbf{x} = \mathbf{x} + \tau \sum_{i=1}^{s} b_i \mathbf{k_i}$ with coefficients $a_{ij}, b_i \in \mathbb{R}$.

The method is denoted by

$$\mathbf{\Psi}^{t+\tau,t}\mathbf{x} = \mathbf{\Psi}^\tau \mathbf{x} = \mathbf{x} + \tau \mathbf{\Psi}(\mathbf{x}, \tau),$$

and the consistency error is

$$\boldsymbol{\epsilon}(\mathbf{x}, \tau) = \mathbf{\Phi}^\tau \mathbf{x} - \mathbf{\Psi}^\tau \mathbf{x}.$$

Constructing RK-methods of order p requires the determination of the

$$s + (1 + \cdots + (s-1)) = \sum_{i=1}^{s} i = \frac{s(s+1)}{2}$$

coefficients in $(\mathbf{b}, \mathbf{A})$. Therefore, we consider $\boldsymbol{\epsilon}(\mathbf{x}, \tau) = \mathbf{\Phi}^\tau \mathbf{x} - \mathbf{\Psi}^\tau \mathbf{x}$ and estimate the error up to order $\mathcal{O}(\tau^{p+1})$ by expanding both summands in a Taylor series. For $\mathbf{\Phi}^\tau$ we have:

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\tau}\mathbf{\Phi}^\tau \mathbf{x}\Big|_{\tau=0} &= \mathbf{f} \\
\frac{\mathrm{d}^2}{\mathrm{d}\tau^2}\mathbf{\Phi}^\tau \mathbf{x}\Big|_{\tau=0} &= \mathbf{f}'\,\mathbf{f} \\
\frac{\mathrm{d}^3}{\mathrm{d}\tau^3}\mathbf{\Phi}^\tau \mathbf{x}\Big|_{\tau=0} &= \cdots \\
\Longrightarrow \qquad \mathbf{\Phi}^\tau \mathbf{x} &= \mathbf{x} + \tau\mathbf{f} + \frac{\tau^2}{2}\mathbf{f}'\,\mathbf{f} + \mathcal{O}(\tau^3)
\end{aligned}$$

Analogously we can argue for $\mathbf{\Psi}^\tau$. In particular, we use here the fact that in the expression $\mathbf{k_i} = \mathbf{f}(\mathbf{x} + \tau \sum_{j=1}^{i-1} a_{ij}\mathbf{k_j})$ the parameters $\mathbf{k_j}$ are multiplied with τ in the argument of $\mathbf{f}$. Since $k_1 = \mathbf{f}(x)$ we have

$$k_2 = \mathbf{f}(x + \tau a_{21} k_1) = \mathbf{f}(x + \tau a_{21}\mathbf{f}) = \mathbf{f} + \tau c_2 \mathbf{f}'\mathbf{f} + \mathcal{O}(\tau^2).$$

Note that also $k_2 = \mathbf{f}(\mathbf{x}) + \mathcal{O}(\tau)$ holds. Repeating the arguments and successively plugging them in, gives us

$$\mathbf{k_i} = \mathbf{f}(\mathbf{x} + \tau \sum_{j=1}^{i-1} a_{ij}\mathbf{f} + \mathcal{O}(\tau^2)) = \mathbf{f} + \tau c_i \mathbf{f}'\,\mathbf{f} + \mathcal{O}(\tau^2).$$

Then, we get

$$\begin{aligned}\mathbf{\Psi}^\tau \mathbf{x} &= \mathbf{x} + \tau \sum_{i=1}^{s} b_i \mathbf{k_i} \\ &= \mathbf{x} + \tau \sum_{i=1}^{s} b_i \mathbf{f} + \frac{\tau^2}{2}(2 \sum_{i=1}^{s} b_i\, c_i\, \mathbf{f}'\,\mathbf{f}) + \mathcal{O}(\tau^3).\end{aligned}$$

Comparing both results, the consistency error reads $\mathbf{\Phi}^\tau \mathbf{x} - \mathbf{\Psi}^\tau \mathbf{x} = \mathcal{O}(\tau^3)$ if we choose the coefficients appropriately.

Theorem 2.19. *Let the right-hand side of the IVP satisfy $f \in \mathcal{C}^p$. An explicit RK-method of stage s has consistency order $p = 1$, if*

$$\sum_{i=1}^{s} b_i = 1. \tag{2.5}$$

It has $p = 2$, if additionally

$$\sum_{i=1}^{s} b_i c_i = \frac{1}{2}. \tag{2.6}$$

Example 2.20. *For $s = 1$, the first condition (2.5) yields $b_1 = 1 \;\Rightarrow$ explicit Euler method.*

For $s = 2$, the conditions (2.5), (2.6) yield a method of second order if

$$\begin{aligned} b_1 + b_2 = 1, \qquad & c_1 = 0 \\ b_1 c_1 + b_2 c_2 = b_2 c_2 = b_2 a_{21} = & \frac{1}{2} \end{aligned}$$

- $b_1 = 0,\ b_2 = 1,\ a_{21} = \frac{1}{2},\ c_2 = \frac{1}{2} \;\Rightarrow$ Runge method

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

- $b_1 = \frac{1}{2},\ b_2 = \frac{1}{2},\ c_2 = 1,\ a_{21} = 1 \;\Rightarrow$ explicit Trapezoidal rule

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

or

$$\begin{aligned}\mathbf{k_1} &= \mathbf{f}(\mathbf{x})\\ \mathbf{k_2} &= \mathbf{f}(\mathbf{x}+\tau\mathbf{k_1})\\ \mathbf{\Psi}^\tau\mathbf{x} &= \mathbf{x}+\frac{\tau}{2}(\mathbf{k_1}+\mathbf{k_2}) = \mathbf{x}+\frac{\tau}{2}[\mathbf{f}(\mathbf{x})+\mathbf{f}(\mathbf{x}+\tau\mathbf{f}(\mathbf{x}))]\end{aligned}$$

Example 2.21. *Be aware that higher order methods require more conditions on the coefficients. A method of stage $s = 4$ and consistency order $p = 4$ is given by the* classical (standard) Runge–Kutta method

0	0			
$\frac{1}{2}$	$\frac{1}{2}$	0		
$\frac{1}{2}$	0	$\frac{1}{2}$	0	
1	0	0	1	0
	$\frac{1}{6}$	$\frac{2}{6}$	$\frac{2}{6}$	$\frac{1}{6}$

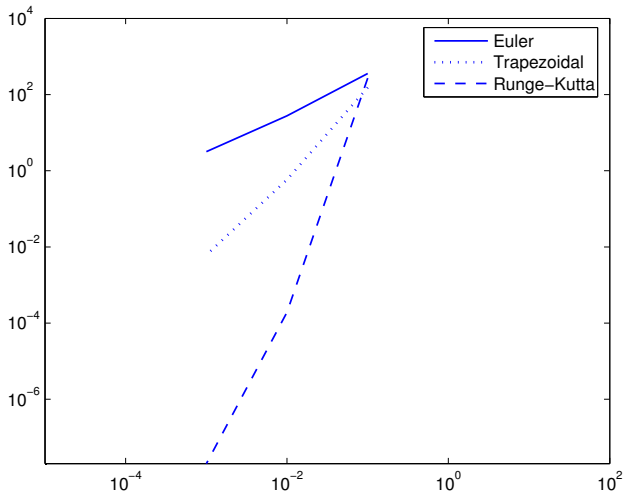


Figure 2.2: Plot of the discretization error as a function of step size in a log-log diagram: Explicit Euler method ($p = 1$), Trapezoidal rule ($p = 2$) and standard Runge-Kutta scheme ($p = 4$).

Remark 2.22. *The number of conditions N_p for the consistency order p is*

p	1	2	3	4	5	6	7	8	9	10	20
N_p	1	2	4	8	17	37	85	200	486	1205	$20 \cdot 10^6$

It certainly remains the question, whether the required equations on the coefficients are solvable. The answer depends on the number of stages s ($\Rightarrow$ number of coefficients).

In general, Lem 2.17 states: $s \geq p$. *More precisely, the following relations for the minimal number of stages* s_p *for a RK-method of order* p *can be shown:*

p	1	2	3	4	5	6	7	8	≥ 9
s_p	1 ↑ *Euler*	2 ↑ *Trapezoidal*	3	4 ↑ *classical RK*	6	7	9	11	$\geq p+3$

Exemplary, a method of order $p = 10$ *has* $s_p \leq 17$ *stages.*

To conclude finally the convergence order from the consistency order, the L–continuity of the increment function has to be proved.

Lemma 2.23. *Let* $\mathbf{f}(t, \mathbf{x}) \in \mathcal{C}^1$ *be* L*–continuous with respect to* $\mathbf{x}$ *on the interval* $-\infty < a \leq t \leq b < \infty$. *Then, the discrete evolution of the RK-method* $(\mathbf{b}, \mathbf{A})$ *has a* L*–continuous increment function* $\mathbf{\Psi}$.

Consequently, the RK-method with consistency order p *converges with convergence order* p.

2.4 Step Size Control

In general, a fixed step size in combination with methods of higher consistency order yield good results for smooth solutions. However, in practice we often find extremely varying, complicated solutions. Here, the step size must be adapted during the computation.

Keep the following in mind:

step size
- ↗ large ⟶ small computational effort (fast computations)
- ↘ small ⟶ good approximation quality (little truncation error)

The step size must be chosen as compromise between accuracy and efficiency. Let us assume that an efficient estimator of the error

$$\boldsymbol{\epsilon}_{\boldsymbol{\Delta}}(t_{j+1}) = \mathbf{x}(t_{j+1}) - \mathbf{x}_{\boldsymbol{\Delta}}(t_{j+1}) = \mathbf{\Phi}^{t_{j+1},t_j}\mathbf{x}(t_j) - \mathbf{\Psi}^{t_{j+1},t_j}\mathbf{x}_{\boldsymbol{\Delta}}(t_j)$$

is available. This *a posteriori* error estimator may be used in many applications and can be constructed in basically two ways:

- The same RK-method is used, but with two different step sizes (typically τ and $\frac{\tau}{2}$).
- RK-methods of different order are used where the number of stages s should match.

In the first case, if a RK-method of order p is being applied, one pretends that, starting from $\mathbf{x}(t_j) = \mathbf{x}_{\mathbf{\Delta}}(t_j)$ the error at t_{j+1} is less than a fixed tolerance. The following relation holds

$$\mathbf{x}(t_{j+1}) - \mathbf{x}_{\mathbf{\Delta}}(t_{j+1}) = c(t_j) \cdot \tau_j^{p+1} + \mathcal{O}(\tau_j^{p+2}). \tag{2.7}$$

Carrying out the same computation twice with a step size $\frac{\tau}{2}$, starting from t_j, going first to $t_{j+\frac{1}{2}}$ and then to t_{j+1}, and denoting by $\tilde{\mathbf{x}}_{\mathbf{\Delta}}(t_{j+1})$ the new solution, yields

$$\mathbf{x}(t_{j+1}) - \tilde{\mathbf{x}}_{\mathbf{\Delta}}(t_{j+1}) = 2c(t_j) \cdot \left(\frac{\tau_j}{2}\right)^{p+1} + \mathcal{O}(\tau_j^{p+2}). \tag{2.8}$$

Substracting (2.7) from (2.8), we get

$$(\frac{1}{2^p} - 1)\tau^{p+1} \cdot c(t_j) = \mathbf{x}_{\mathbf{\Delta}}(t_{j+1}) - \tilde{\mathbf{x}}_{\mathbf{\Delta}}(t_{j+1}) + \mathcal{O}(\tau_j^{p+2}),$$

and hence

$$\mathcal{E} = \frac{\tilde{\mathbf{x}}_{\mathbf{\Delta}}(t_{j+1}) - \mathbf{x}_{\mathbf{\Delta}}(t_{j+1})}{(2^p - 1)}.$$

If $\mathcal{E}$ is less than a fixed tolerance TOL, the scheme moves to the next iteration step, otherwise the computation for the estimate is repeated with a halved step size. This approach involves an additional computational effort, due to the extra functional evaluations needed for $\tilde{\mathbf{x}}_{\mathbf{\Delta}}(t_{j+1})$.

An alternative that does not require additional functional evaluations consists of simultaneously using two different RK-methods with a total of s stages, of order p and $p+1$, respectively. That means, those methods share the same values $\mathbf{k_i}$ and are therefore called *embedded RK-methods*. These methods are represented by the same (modified) Butcher array

$\mathbf{c}$	$\mathbf{A}$
	$\mathbf{b}^T$
	$\hat{\mathbf{b}}^T$

where the method of order p is identified by $(\mathbf{b}, \mathbf{c}, \mathbf{A})$ and that of order $p+1$ by $(\hat{\mathbf{b}}, \mathbf{c}, \mathbf{A})$. The difference of the two approximate solutions at t_{j+1} provides an estimate of the error for the lower order scheme. We consider the two RK-methods of different orders $\mathbf{x}_{\mathbf{\Delta}}^{\mathbf{p}}(t_{j+1})$ and $\mathbf{x}_{\mathbf{\Delta}}^{\mathbf{p+1}}(t_{j+1})$ for the same step size τ_j. By

$$E(t_{j+1}) := |\mathbf{x}_{\mathbf{\Delta}}^{\mathbf{p}}(t_{j+1}) - \mathbf{x}_{\mathbf{\Delta}}^{\mathbf{p+1}}(t_{j+1})|$$

we define the difference of both approximations. Again, let TOL be the desired precision. Then, the step size control works as it is shown in Algorithm 1.

Algorithm 1 Basic adaptive algorithm

Require: safety factor $\rho < 1$, maximal step size τ_{max}, max. multiplication factor $q > 1$

$j := 0$
$\tau_0 := \tau_{max}$
$\Delta := \{t_0\}$
$x_\Delta(t_0) := x_0$
while $t_j < T$ **do**
  $t := t_j + \tau_j$
  $x := x^p_\Delta(t)$
  compute the error estimate $E(t)$
  $\tilde{\tau} := \sqrt[p+1]{\frac{\rho \cdot TOL}{E(t)}}\tau_j$
  $\tau := \min(q\tau_j, \tau_{max}, \tilde{\tau})$
  if $E(t) \leq TOL$ **then**
    $t_{j+1} := t$
    $\Delta := \Delta \cup \{t_{j+1}\}$
    $x_\Delta(t_{j+1}) := x$
    $\tau_{j+1} := \min(\tau, T - t_{j+1})$
    $j := j + 1$
  else
    $\tau_j := \tau$
  end if
end while

Remark 2.24. *Since the stages* $\mathbf{k_i}$ *of an embedded RK-method coincide, this approach does not require extra functional evaluations.*

Example 2.25. *Explicit Euler method (*$p = 1$*) and explicit Trapezoidal rule (*$p = 2$*)*

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

A more general Butcher array can be written as

$$\begin{array}{c|ccc} c_1 & 0 & & \\ c_2 & a_{21} & 0 & \\ c_3 & a_{31} & a_{32} & 0 \\ \hline & b_1 & b_2 & \\ & \hat{b}_1 & \hat{b}_2 & \hat{b}_3 \end{array}$$

3 Implicit One-Step Methods

3.1 Stability of ODEs

Since we want to study the stability behavior of the solutions for a **long** time period, we consider the following theoretical concepts.

3.1.1 General Concept

Let $(t_0, \mathbf{x_0})$ and the evolution $\mathbf{\Phi}^{t,t_0}\mathbf{x_0}$ of the ODE $\mathbf{x}' = \mathbf{f}(t, \mathbf{x})$ be given for all $t \geq t_0$. The solution $\mathbf{\Phi}^{t,t_0}\mathbf{x_0}$ is called

- *stable (Ljapunov–stable)*, if for all $\varepsilon > 0$ there exists a $\delta > 0$ such that

  $$\mathbf{\Phi}^{t,t_0}\mathbf{x} \in B_\varepsilon(\mathbf{\Phi}^{t,t_0}\mathbf{x_0}) \qquad \forall t \geq t_0$$

  for all perturbed initial values $\mathbf{x} \in B_\delta(\mathbf{x_0})$.

- *asymptotically stable*, if there exists additionally a $\delta_0 > 0$ such that

  $$\lim_{t\to\infty} \|\mathbf{\Phi}^{t,t_0}\mathbf{x_0} - \mathbf{\Phi}^{t,t_0}\mathbf{x}\| = 0 \qquad \forall \mathbf{x} \in B_{\delta_0}(\mathbf{x_0}).$$

- *unstable* in all other cases.

Here, $B_\delta(\mathbf{x})$ denotes a "ball" of radius δ around the center point $\mathbf{x}$. For illustrating the stability concept the trajectories of the problem are very helpful: If they stay together for neighboring initial values, the solution is stable. If they even run together into a common point (converge), the solution is asymptotically stable. If they diverge, the solution is unstable.

Example 3.1. *Consider the linear ODE* $x'(t) = \lambda x(t)$, $x(0) = 1$, $t \in \mathbb{R}$ *with its solution* $x(t) = e^{\lambda t}$. *If its initial value is slightly perturbed, i.e.* $z(0) = 1 + \epsilon$, $\epsilon > 0$, *the difference of the solutions reads*

$$d(t) := (z - x)(t) = (1 + \epsilon)e^{\lambda t} - e^{\lambda t} = \epsilon e^{\lambda t}$$

and the limit for $t \to \infty$ *depends on the choice of* λ*:*

- $\lambda > 0$: *the difference grows exponentially for every small* ϵ, $\lim_{t\to\infty} d(t) \to \infty$ $\Longrightarrow$ *unstable*
- $\lambda = 0$: *the difference is constant,* $\lim_{t\to\infty} d(t) = \epsilon$ $\Longrightarrow$ *stable*
- $\lambda < 0$: *the difference decays exponentially,* $\lim_{t\to\infty} d(t) = 0$ $\Longrightarrow$ *asymptotically stable*

A stability statement for the linear system of ODEs $\mathbf{x}'(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{b}(t)$ can be derived from the homogeneous problem. Thus, let us restrict on the homogeneous, linear and autonomous system

$$\mathbf{x}'(t) = \mathbf{A}\,\mathbf{x}(\mathbf{t}), \qquad \text{with constant } \mathbf{A} \in \mathbb{R}^{d\times d} \text{ and } \mathbf{x} : [a,b] \to \mathbb{R}^d$$

Then, $\mathbf{\Phi}^t : \mathbb{R}^d \to \mathbb{R}^d$ is a linear mapping.

Remark 3.2. *The solution of* $\mathbf{x}' = \mathbf{A}\,\mathbf{x}$ *can be represented by means of the exponential function for the matrix*

$$\mathbf{\Phi}^t = e^{t\mathbf{A}} = \sum_{k=0}^{\infty} \frac{(t\mathbf{A})^k}{k!}.$$

We can easily determine the stability behavior of the linear autonomous system by help of the following theoretical results.

Lemma 3.3. *The solution corresponding to* $\mathbf{x}(0) = \mathbf{x_0}$ *is stable, if and only if*

$$\sup_{t\geq 0} \|\mathbf{\Phi}^t\| < \infty.$$

It is asymptotically stable, if and only if

$$\lim_{t\to\infty} \|\mathbf{\Phi}^t\| = 0.$$

In particular, all solutions are (asymptotically) stable, if there is only one solution.

But generally, the solution is not obvious. That is why we seek an approach which exploits matrix properties to analyze whether a system is (asymptotic) stable or not.

Definition 3.4.

- Spectrum *of a matrix* $\mathbf{A} \in \mathbb{C}^{d\times d}$ *is the set of eigenvalues:*

$$\sigma(\mathbf{A}) = \{\lambda \in \mathbb{C} \,|\, \det(\lambda\mathbf{I} - \mathbf{A}) = 0\}$$

- Spectral abcissca *of* $\mathbf{A}$ *is the maximal real part of the eigenvalues:*

$$\nu(\mathbf{A}) = \max_{\lambda\in\sigma(\mathbf{A})} \mathcal{R}e(\lambda)$$

- Index $i(\lambda)$ *of an eigenvalue* $\lambda \in \sigma(\mathbf{A})$ *is the maximal dimension of the Jordan blocks of* $\mathbf{A}$ *with respect to* λ*.*

Theorem 3.5 (Stability)**.** *The linear IVP* $\mathbf{x}' = \mathbf{A}\mathbf{x}$, $\mathbf{A} \in \mathbb{C}^{d\times d}$ *is stable if and only if* $\nu(\mathbf{A}) \leq 0$ *and all eigenvalues* $\lambda \in \sigma(\mathbf{A})$ *with* $\mathcal{R}e(\lambda) = 0$ *have the index* $i(\lambda) = 1$*. It is asymptotically stable, if* $\nu(\mathbf{A}) < 0$*.*

Let us now focus on the stability of special solutions, i.e. of the fixpoints of the autonomous ODE $\mathbf{x}' = \mathbf{f}(\mathbf{x})$. Let $\mathbf{x}_*$ be chosen such that

$$\mathbf{\Phi}^t \mathbf{x}_* = \mathbf{x}_* \quad \Longleftrightarrow \quad \mathbf{f}(\mathbf{x}_*) = \mathbf{0}$$

holds. Then, $\mathbf{x}_*$ is called *fixpoint* of $\mathbf{\Phi}^t$. It is also known as *singular point* or *equilibrium point*.

Example 3.6. *The system*

$$\mathbf{x}' = -\lambda \mathbf{x}, \quad \lambda = a + ib \in \mathbb{C}, \quad \mathcal{R}e(\lambda) > 0$$

(resp. its solution) is asymptotically stable, since $\nu(\mathbf{A}) = \nu(-\lambda) = -a < 0$ *or*

$$\begin{aligned} \lim_{t\to\infty} ||\mathbf{\Phi}^t \mathbf{x_0} - \mathbf{\Phi}^t \mathbf{x}|| &= \lim_{t\to\infty} \|\mathbf{\Phi}^t(\mathbf{x_0} - \mathbf{x})\| \\ &= \lim_{t\to\infty} |e^{-(a+ib)t}| \ \|\mathbf{x_0} - \mathbf{x}\| \\ &= \lim_{t\to\infty} |e^{-at}| \ \|\mathbf{x_0} - \mathbf{x}\| = 0. \end{aligned}$$

Example 3.7. *The harmonic (damped) oscillator has the system matrix*

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ -1 & -\alpha \end{pmatrix}, \quad \textit{with eigenvalues } \lambda_{1,2} = -\frac{\alpha}{2} \pm \sqrt{\frac{\alpha^2}{4} - 1}.$$

The stability of the fixpoint $\mathbf{x}_* = (0,0)^T$, *here* $\mathbf{f}(\mathbf{x}_*) = \mathbf{A}\mathbf{x}_* = (0,0)^T$, *depends on* α*:*

$$\begin{array}{lllll} \alpha > 0 \ \Rightarrow & \mathcal{R}e(\lambda_i) < 0, \quad i = 1,2 & \Longrightarrow & \textit{asymptotically stable} \\ \alpha = 0 \ \Rightarrow & \lambda_{1,2} = \pm i, \quad i(\lambda_{1,2}) = 1 & \Longrightarrow & \textit{stable} \end{array}$$

In the non-linear case, the spectral abcissca of the Jacobian matrix $\mathbf{Df}(\mathbf{x}_*)$ sometimes decides about the stability behavior of the solution.

Theorem 3.8. *Let* $\mathbf{x}_*$ *be fixpoint of* $\mathbf{x}' = \mathbf{f}(\mathbf{x})$, $\mathbf{x}(0) = \mathbf{x_0}$, $\mathbf{f} \in \mathcal{C}^1(\mathbb{R}^d)$. *If* $\nu(\mathbf{Df}(\mathbf{x}_*)) < 0$, *i.e. the linearized ODE* $\mathbf{x}' = \mathbf{Df}(\mathbf{x}_*)\mathbf{x}$ *is asymptotically stable, then* $\mathbf{x}_*$ *is an asymptotically stable fixpoint of the non-linear system* $\mathbf{x}' = \mathbf{f}(\mathbf{x})$. *This means that* $\mathbf{x}$ *converges towards* $\mathbf{x}_*$, *if* $\mathbf{x_0} \in B_\delta(\mathbf{x}_*)$.

Linear $\mathbf{x}' = \mathbf{Df}(\mathbf{x}_*)\mathbf{x}$ asymptotically stable
$\Downarrow$
Non-linear $\mathbf{x}' = \mathbf{f}(\mathbf{x})$ asymptotically stable

Remark 3.9. *Could we also conclude the stability of the non-linear system from the stability of the linear one? The answer is* **NO***!*

Take for example

$$x'(t) = \beta x^3(t) = f(x(t)), \quad x(0) = x_0, \quad \beta \in \mathbb{R}$$

with fixpoint $x_* = 0$ *. Then, the linearization around the fixpoint yields*

$$\mathrm{D}f(x_*) = 3\beta x_*^2 = 0.$$

Consequently, the corresponding linear system reads $x'(t) = 0$ *, and its constant solution* $x(t) = x_0$ *is stable.*

But x_* *is not a stable fixpoint of the non-linear system* $x' = f(x)$ *! Using separation of variables we obtain the exact solution of the non-linear ODE*

$$x(t) = \frac{x_0}{\sqrt{1 - 2x_0^2\beta t}}$$

whose stability behavior depends obviously on the choice of β *:*

- $\beta > 0$ *:* $\lim_{t \to (2x_0^2\beta)^{-1}} |x(t)| \to \infty \quad \Longrightarrow$ *unstable*
- $\beta = 0$ *:* $x(t) = x_0 \quad \Longrightarrow$ *stable*
- $\beta < 0$ *:* $\lim_{t \to \infty} |x(t)| = 0 \quad \Longrightarrow$ *asymptotically stable*

3.2 Inheritance of Stability Concepts

The convergence of a method is an asymptotical result of the step size that has to be chosen sufficiently small, as we have seen in Ch 2. However, for a given step size τ, the derived convergence result says nothing about the quality of the solution. And sufficiently small means sometimes so small that it is practically impossible to compute acceptable solutions. In case of stiff problems, the discussed explicit methods will even lead to **absolutely wrong** results for relatively small step sizes.

Example 3.10. *The explicit Euler method applied on* $\mathbf{x}' = -\lambda\mathbf{x}, \ \lambda \gg 1$ *yields*

$$\mathbf{x_{j+1}} = (1 - \tau\lambda)\mathbf{x_j}.$$

According to Rem 2.14 the step size τ *must satisfy the restriction* $\tau < 2/\lambda$ *to result in the correct solution* $\mathbf{x_j} \overset{j\to\infty}{\longrightarrow} \mathbf{0}$ *and not in the nonsense* $|\mathbf{x_j}| \overset{j\to\infty}{\longrightarrow} \boldsymbol{\infty}$ *.*

To be more precise, what is the mathematical meaning of the expression 'absolutely wrong'? In Sec 3.1 we have introduced the definitions for the stability of an ODE $\mathbf{x}' = \mathbf{f}(\mathbf{x})$ and of a recursive mapping $\mathbf{x_{j+1}} = \mathbf{\Psi}(\mathbf{x_j})$. Since each one-step method defines a recursive mapping, the method can be stable or not. And even more, it can inherit the stability behavior of the corresponding underlying ODE or not.

For our Ex 3.10:

- The linear ODE is asymptotically stable $(\mathbf{x}(t) \to \mathbf{0},\, t \to \infty)$.
- The linear recursion $\mathbf{x_{j+1}} = (1 - \tau\lambda)\mathbf{x_j} = B\mathbf{x_j}$ is only asymptotically stable, if $\mathbf{x_j} \overset{j\to\infty}{\longrightarrow} \mathbf{0}$, hence if the step size satisfies $\tau < 2/\lambda$.

Handling stiff ODEs, we require appropriate methods that yield correct (asymptotically stable) numerical results independent of the choice of the step size. This implies that the discrete evolution inherits the asymptotical stability of the continuous ODE for **all step sizes**. Otherwise, it might lead to absolutely wrong (unstable) results as seen in our example.

Definition 3.11. *The value*

$$S = \frac{\max_{\lambda\in\sigma(A)} |\operatorname{Re}\,(\lambda)|}{\min_{\lambda\in\sigma(A)} |\operatorname{Re}\,(\lambda)|}$$

is called the quotient of stiffness for a linear system of ODEs $y' = Ay$. *The linear system is called stiff for a numerical scheme if*

- *the real part of all eigenvalues of* A *are negative and*
- *if there is a strong restriction on the step size* τ *to obtain a stable iteration, i.e.*

$$||x_{n+1}|| \leq ||x_n||.$$

So-called *implicit methods* satisfy our demands. Here, a linear system of equations has to be solved for determining the next function value $\mathbf{x_{j+1}}$.

Example 3.12. *The* implicit Euler method *for the ODE* $\mathbf{x}' = -\lambda\mathbf{x}$ *is obtained if – instead of* $\mathbf{x_{j+1}} = \mathbf{x_j} - \tau\lambda\mathbf{x_j}$ *– the following recursion is used*

$$\begin{aligned} \mathbf{x_{j+1}} = \mathbf{x_j} - \tau\lambda\mathbf{x_{j+1}} \quad &\Rightarrow \quad \mathbf{x_{j+1}} = (1+\tau\lambda)^{-1}\mathbf{x_j} \\ &\Rightarrow \quad\ \mathbf{x_j} = \frac{1}{(1+\tau\lambda)^j}\mathbf{x_0}, \end{aligned}$$

i.e. obviously $\mathbf{x_j} \to \mathbf{0},\ j \to \infty$ *for all* $\tau > 0$ *!*

3.3 Consistent Rational Approximation

Consider the linear autonomous system of ordinary differential equations

$$\mathbf{x}'(t) = \mathbf{A}\mathbf{x}(t), \quad \mathbf{A} \in \mathbb{R}^{d\times d}, \quad \mathbf{x}(0) = \mathbf{x_0} \tag{3.1}$$

whose solution is given by $\mathbf{x}(t) = \mathbf{\Phi}^t\mathbf{x_0} = e^{t\mathbf{A}}\mathbf{x_0}$. Moreover, consider the discrete method

$$\mathbf{x_{n+1}} = \mathbf{\Psi}^\tau\mathbf{x_n}, \qquad \mathbf{x_0} \text{ given}$$

for the numerical solution of Eq (3.1), where the discrete evolution $\mathbf{\Psi}^\tau$ is an approximation of the continuous one

$$\mathbf{\Phi}^\tau = e^{\tau \mathbf{A}}.$$

In the following we deal with the approximation qualities of $\mathbf{\Psi}^\tau$. For this purpose we focus on the class of discrete evolutions being described by rational functions. Let

$$R(z) = \frac{P(z)}{Q(z)} \qquad \text{where} \qquad R : \mathbb{C} \to \mathbb{C}$$

be a rational function with coprime polynomials P, Q, i.e. P, Q have no common divisor. The matrix function $R(\mathbf{A}) = Q^{-1}(\mathbf{A})\, P(\mathbf{A})$ is well-defined, if $Q(\mathbf{A})$ is invertible.

Remark 3.13.

1. *The matrix function $R(\mathbf{A})$ is defined if and only if*

$$R(\lambda) \neq \infty \quad \forall \lambda \in \sigma(\mathbf{A}),$$

   i.e. $Q(\mathbf{A})$ is invertible, if $Q(\lambda) \neq 0$ for all eigenvalues λ of the matrix $\mathbf{A}$, $\lambda \in \sigma(\mathbf{A})$.
2. *The Dunford-Taylor-Calculus (Spectral-Calculus) is valid:*

$$\sigma(R(\mathbf{A})) = R(\sigma(\mathbf{A})).$$

Let the discrete evolution be of the form

$$\mathbf{\Psi}^\tau = R(\tau \mathbf{A}).$$

Then, $\mathbf{\Psi}^\tau$ is the solution of the linear system

$$Q(\tau \mathbf{A})\, (\mathbf{\Psi}^\tau \mathbf{x}) = P(\tau \mathbf{A}) \mathbf{x}.$$

The corresponding numerical method is explicit if $Q = 1$, i.e. $R = P$ holds, otherwise it is implicit.

Example 3.14.

- *Explicit Euler:* $Q(z) = 1,\ P(z) = 1 + z$, *(* $R(z) \neq \infty\ \forall z \in \mathbb{C}$ *)*

$$\Rightarrow \quad \mathbf{x_{n+1}} = \mathbf{\Psi}^\tau \mathbf{x_n} = (\mathbf{I} + \tau \mathbf{A})\, \mathbf{x_n}.$$

- *Implicit Euler:* $Q(z) = 1 - z,\ P(z) = 1$, *(* $R(z) = \infty$ *for* $z = +1$ *)*

$$\Rightarrow \quad (\mathbf{I} - \tau \mathbf{A})\, \mathbf{x_{n+1}} = (\mathbf{I} - \tau \mathbf{A}) \mathbf{\Psi}^\tau \mathbf{x_n} = \mathbf{x_n}.$$

Remark 3.15. *If $\mathbf{x}' = \mathbf{A}\mathbf{x}$ is asymptotically stable, then $\mathcal{R}e(\lambda) < 0$ holds for all $\lambda \in \sigma(\mathbf{A})$. This implies that $\lambda \neq 1$ and thus $R(\lambda) \neq \infty$, $\lambda \in \sigma(\mathbf{A})$. The rational approximation in Ex 3.14 is well-defined.*

Example 3.16. *For the explicit RK-methods,* $Q = 1$ *and the discrete evolution reads*

$$\mathbf{\Psi}^\tau = P(\tau\mathbf{A}), \quad \text{with } P \in \mathbb{P}_s \text{ polynomial of degree } s\,,$$

where s *denotes the number of stages of the method.*

- *Runge method:* $s = 2\,,\ P(z) = 1 + z + z^2/2\,.$
- *Classical Runge–Kutta method:* $s = 4\,,\ P(z) = 1 + z + z^2/2 + z^3/6 + z^4/24\,.$

In general, each rational approximation of the exponential function $R(z) \approx e^z$ leads to an approximation of the evolution

$$\mathbf{\Psi}^\tau = R(\tau\mathbf{A}) \approx e^{\tau\mathbf{A}} = \mathbf{\Phi}^\tau.$$

Definition 3.17. *The* consistency order *of the approximation of the exponential function that is prescribed by the rational function* R *is the greatest number* $p \in \mathbb{N}$ *with*

$$R(z) = e^z + \mathcal{O}(z^{p+1}) \quad \text{for } z \to 0.$$

If R *is a rational approximation of order* p*, then the respective numerical method has consistency order* p

$$\mathbf{\Psi}^\tau\mathbf{x} = \mathbf{\Phi}^\tau\mathbf{x} + \mathcal{O}(\tau^{p+1}),$$
$$R(\tau\mathbf{A})\mathbf{x} = e^{\tau\mathbf{A}}\mathbf{x} + \mathcal{O}(\tau^{p+1}).$$

Lemma 3.18. *Let* $R = P/Q$ *be a rational approximation of the exponential function of order* p*. Then,*

$$p \le \deg P + \deg Q.$$

Proof: Suppose there exist polynomials P and Q with $\deg P \le k$, $\deg Q \le j$ such that $k + j < p$.

$$\Rightarrow \frac{P(z)}{Q(z)} - e^z = \mathcal{O}(z^{k+j+2}) \quad \text{for } z \to 0$$
$$\Rightarrow P(z) - Q(z)e^z = \mathcal{O}(z^{k+j+2}) \qquad (*)$$

We claim: Equation $(*)$ implies $P = Q = 0$ which is the desired contradiction. Induction with respect to k yields:

- $k = 0 \ \Rightarrow P = \text{const.}$

  $$\overset{(*)}{\Rightarrow} Pe^{-z} - Q(z) = \mathcal{O}(z^{j+2}) \qquad (**)$$

  By differentiating this $(j+1)$-times with respect to z we find ($\deg Q \le j$)

  $$(-1)^{j+1}P = \mathcal{O}(z),\ z \to 0 \Rightarrow P = 0, \text{ since } P = \text{const.}$$
  $$\overset{(**)}{\Rightarrow} Q = 0 \quad [Q \sim z^j,\ Q = \mathcal{O}(z^{j+1})]$$

  Now we assume that the statement holds for $k - 1 \ge 0$.

- $k-1 \to k$ Differentiation of $(*)$ with respect to z gives

$$(*) \Rightarrow P'(z) - (Q'(z) + Q(z))e^z = \mathcal{O}(z^{k+j+1}),\ z \to 0$$

Since deg $P' \leq k-1$ and deg $(Q+Q') \leq j$ we are in the setting of the induction hypothesis and can conclude that

$\Rightarrow$ $P' = 0$ (and $Q' + Q = 0$)
$\Rightarrow$ $P = \text{const.}$
$\Rightarrow$ $Q = 0$ as for $k = 0$. ■

Remark 3.19. *Compare Lem 3.18 with Lem 2.17 that states the relation between the consistency order p and the stages s of the explicit RK-methods, $p \leq s$. Here, $Q = 1$, thus its degree is* $\deg Q = 0$.

3.4 A–Stability

Let the linear autonomous system of Eq (3.1), $\mathbf{x}' = \mathbf{A}\mathbf{x}$, $\mathbf{A} \in \mathbb{R}^{d\times d}$, be asymptotically stable. Hence, $\mathcal{R}e(\lambda) < 0$ holds for all $\lambda \in \sigma(\mathbf{A})$. Let R be a consistent rational approximation of the exponential function. For the numerical treatment of Eq (3.1), the understanding of the stability behavior of the discrete method is of great importance. In particular, we are interested in answering the following question: For which step sizes τ is the asymptotical stability of the ODE inheritable on $\mathbf{\Psi}^\tau = R(\tau\mathbf{A})$?

The recursion is asymptotically stable, if

$$\varrho(R(\tau\mathbf{A})) = \max_{\lambda\in\sigma(R(\tau\mathbf{A}))} |\lambda| < 1$$

or

$$\max_{\lambda\in\sigma(\mathbf{A})} |R(\tau\lambda)| < 1, \quad \text{since } \sigma(R(\tau\mathbf{A})) = R(\tau\sigma(\mathbf{A}))$$

Let us define the stability domain of the recursion.

Definition 3.20.

$$S := \{z \in \mathbb{C} \mid |R(z)| < 1\}$$

is called stability domain.

Remark 3.21. *If $\tau\lambda \in S$, then $|R(\tau\lambda)| < 1$ holds, i.e. to get an asymptotically stable method we need $\tau\lambda \in S$ for all $\lambda \in \sigma(\mathbf{A})$.*

Definition 3.22.

$$\tau_c := \sup\{\bar\tau > 0 \mid \mathbf{x_{n+1}} = \mathbf{\Psi}^\tau\mathbf{x_n} \text{ is asymptotically stable for } 0 < \tau < \bar\tau\}$$

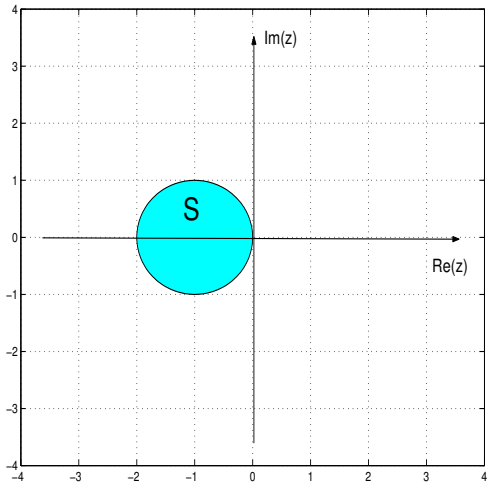
is called characteristic step size.

Remark 3.23. *If only the inheritance of stability (not asymptotical stability) is of interest, the boundary* $\{z \in \mathbb{C} \mid |R(z)| \leq 1\}$ *in combination with additional conditions on the eigenvalues of the system, cf. Th 3.5, has to be studied in detail.*

Example 3.24.

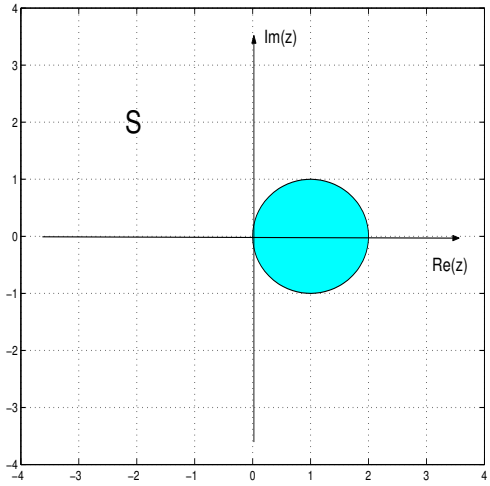
- *Explicit Euler:*

$$R(z) = 1 + z$$
$$S = \{z \in \mathbb{C} : \ |1 + z| < 1\}$$

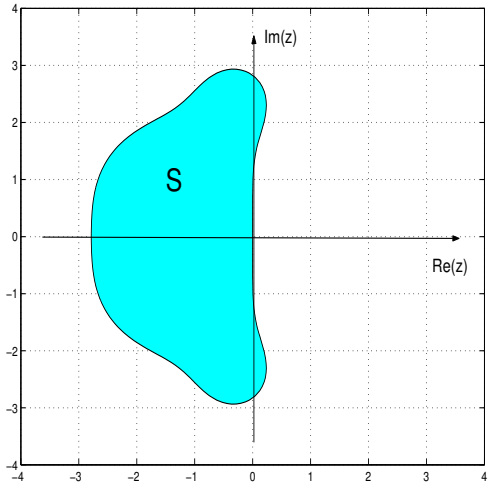


- *Implicit Euler:*

$$R(z) = (1 - z)^{-1} = 1 + z + \mathcal{O}(z^2)$$
$$S = \{z \in \mathbb{C} : \ |1 - z| > 1\}$$



- *Explicit classical Runge-Kutta method ($p = 4$):*



Remark 3.25.

- *For the considered explicit examples we observe that $\tau\lambda \in S$ requires a characteristic step size $\tau_c < \infty$, since the stability domain S is bounded.*

  $\Longrightarrow$ *The larger λ, the smaller the step size τ.*

- *In general, the stability domain of methods that are prescribed by $R = P$ with polynomial P is bounded, because $|P(z)| \to \infty$ for $|z| \to \infty$. Thus, methods given by polynomials, i.e. explicit methods, require a certain choice of τ_c.*

- *If particularly $|\lambda| \gg 1$, $\mathcal{R}e(\lambda) < 0$, very small step sizes are required for stability ($\Rightarrow$ stiff problems).*

Remark 3.26. *The observations of Rem 3.25 are not valid for the implicit Euler method. In this case,*

$$\mathbb{C}^- := \{z \in \mathbb{C},\ \mathcal{R}e(z) < 0\} \subset S.$$

Hence, for $\mathcal{R}e(\lambda) < 0$ all step sizes $\tau > 0$ satisfy $\tau\lambda \in S$. The method is asymptotically stable for all $\tau > 0$.

This defines a family of methods.

Definition 3.27. *A method is called* A–stable, *if*

$$\mathbb{C}^- \subset S.$$

An A–stable method is asymptotically stable for all step sizes $\tau > 0$, if the underlying ODE $\mathbf{x}' = \mathbf{A}\mathbf{x}$ is asymptotically stable.

Remark 3.28. *These A–stable methods, i.e. implicit methods, are appropriate for handling stiff problems. Explicit methods are never A–stable.*

Remark 3.29. *A restriction of A–stability is the so-called* A(α) –stability. *Let*

$$S_\alpha := \{z \in \mathbb{C} \mid |\arg(-z)| < \alpha\},\ \alpha \in \left[0, \frac{\pi}{2}\right].$$

The rational approximations R whose stability domain fulfills $S_\alpha \subset S$ prescribe A(α) *–stable methods. For $\alpha < \pi/2$, this condition is weaker, but it is enough to ensure asymptotical stability if $\lambda \in \sigma(\mathbf{A})$ is contained in S_α.*

Example 3.30. A(α) –stability *is used to analyze the family of BDF (backward differentiation formula) implicit multi-step methods.*

We study now the inheritance of asymptotical stability in case of non-linear problems. According to Th 3.8, we can conclude the asymptotical stability of the fixpoint $\mathbf{x}_*$ in the non-linear problem from the asymptotical stability of the linearization. Consider the non-linear autonomous system

$$\mathbf{x}' = \mathbf{f}(\mathbf{x}), \qquad \mathbf{f}(\mathbf{x}_*) = \mathbf{0},$$
$$\max_{\lambda \in \sigma(\mathbf{Df}(\mathbf{x}_*))} \mathcal{R}e(\lambda) = \nu(\mathbf{Df}(\mathbf{x}_*)) < 0$$

and its linearization around the fixpoint $\mathbf{x}_*$

$$\mathbf{x}' = \mathbf{f}(\mathbf{x}_*) + \mathbf{A}(\mathbf{x} - \mathbf{x}_*) = \mathbf{A}(\mathbf{x} - \mathbf{x}_*), \qquad \mathbf{A} = \mathbf{Df}(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_*}.$$

Furthermore, consider the one-step method given by $\mathbf{\Psi}^\tau$ for the non-linear equation. We linearize $\mathbf{\Psi}^\tau$ around $\mathbf{x}_*$: $\mathbf{\Psi}^\tau\mathbf{x} = \mathbf{x}_* + \mathbf{D_x}\mathbf{\Psi}^\tau\mathbf{x}|_{\mathbf{x}=\mathbf{x}_*}(\mathbf{x} - \mathbf{x}_*) + \cdots$ and define

$$\mathbf{\Psi}^\tau_*\mathbf{x} = \mathbf{x}_* + \mathbf{D_x}\mathbf{\Psi}^\tau\mathbf{x}|_{\mathbf{x}=\mathbf{x}_*}(\mathbf{x} - \mathbf{x}_*).$$

A consistent one-step method is called *invariant with respect to linearization*, if the discrete evolution $\mathbf{\Psi}^\tau$ of the non-linear equation and its linearization $\mathbf{\Psi}^\tau_*$ satisfy

- $\mathbf{\Psi}^\tau\mathbf{x}_* = \mathbf{x}_*$.
- $\mathbf{D_x}\mathbf{\Psi}^\tau\mathbf{x}|_{\mathbf{x}=\mathbf{x}_*} = R(\tau\mathbf{A})$ with rational function R
  $\Longrightarrow\ \mathbf{\Psi}^\tau_\star\mathbf{x} = \mathbf{x}_* + R(\tau\mathbf{A})(\mathbf{x} - \mathbf{x}_*)$.
- The discrete evolution $\mathbf{\Psi}^\tau_\star$ defines a consistent one-step method for the linearized system $\mathbf{x}' = \mathbf{A}(\mathbf{x} - \mathbf{x}_*)$.

Remark 3.31. *All considered methods so far are invariant with respect to linearization.*

Theorem 3.32. *Consider the situation of Th 3.8. Let $\mathbf{\Psi}^\tau$ be an invariant one-step method with respect to linearization around the fixpoint $\mathbf{x}_*$. Let $\tau_c > 0$ be the characteristic step size of $\mathbf{\Psi}^\tau_*$ to ensure the inheritance of asymptotical stability on*

the discretization in the linear case. Then, $\mathbf{x}_*$ *is an asymptotically stable fixpoint of the non-linear recursion*

$$\mathbf{x_{n+1}} = \mathbf{\Psi}^\tau \mathbf{x_n}, \qquad n = 0, 1, 2, \ldots$$

for $\tau < \tau_c$.

If the one-step method is A–stable, then $\tau_c = \infty$.

Example 3.33. *Consider the IVP* $x' = \lambda(1 - x^2),\ \lambda > 0\,,\ x(0) = x_0$. *Then* $x_{*1} = 1$ *is asymptotically stable fixpoint (in contrast,* $x_{*2} = -1$ *is unstable). For* $x_0 > -1\,,$ $x(t) \to x_{*1},\ t \to \infty$ *holds. The linearization around the stable fixpoint yields*

$$x' = A(x - x_{*1}) = -2\lambda(x - 1),$$

since

$$A = \mathrm{D}f(x)|_{x=x_{*1}} = -2\lambda x|_{x=1} = -2\lambda.$$

As already known, the explicit Euler method is stable for the linearized equation, if $\tau < 1/\lambda$. *Then the linear iteration converges towards* x_{*1} . *According to Th 3.32,* x_{*1} *is also fixpoint of the non-linear iteration for these step sizes. But note that the choice of the starting values* x_0 *is restricted.*

Using Banach's Fixpoint Theorem direct computation shows that the non-linear iteration converges, if $\tau < 1/\lambda$ *and* $x_0 \in [0, 5/4]$.

3.5 Isometry

Until now, we have only discussed the inheritance of the asymptotical stability. But there are also other properties that have to be kept in the numerical method, take for example the conservation of the norm which implies the conservation of energy.

Consider the linear autonomous IVP $\mathbf{x}' = \mathbf{A}\mathbf{x}\,,\ \mathbf{A} \in \mathbb{R}^{d\times d}\,,\ \mathbf{x}(0) = \mathbf{x_0}$. In case of energy conservation, the continuous evolution fulfills

$$\|\mathbf{\Phi}^t \mathbf{x_0}\|_2 = \|\mathbf{x_0}\|_2, \quad \text{for } t \in \mathbb{R}$$

in the Euclidian norm $\|\cdot\|_2$ or respectively,

$$\|e^{t\mathbf{A}} \mathbf{x_0}\|_2 = \|\mathbf{x_0}\|_2, \quad \text{for } t \in \mathbb{R}.$$

We call here $e^{t\mathbf{A}}$ an *isometry* (*orthogonal*, i.e. $(e^{t\mathbf{A}})^T\, e^{t\mathbf{A}} = \mathbf{I}$).

Example 3.34. *Let*

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix},$$

then

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}' = \begin{pmatrix} x_2 \\ -x_1 \end{pmatrix} \ , \quad \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

The solution of this system reads

$$\begin{aligned} x_1(t) &= \cos t, \\ x_2(t) &= -\sin t. \end{aligned}$$

Obviously, the system is norm / energy conserving, $\|(x_1, x_2)^T(t)\|_2 = 1$ *for all* $t \in \mathbb{R}$.

Theorem 3.35. *Let* $\mathbf{A} \in \mathbb{R}^{d\times d}$, $t \in \mathbb{R}$. *Then,* $e^{t\mathbf{A}}$ *is an isometry (orthogonal), if* $\mathbf{A}$ *is skew-symmetric, i.e.* $\mathbf{A}^T = -\mathbf{A}$. *In this case, all eigenvalues of* $\mathbf{A}$ *are imaginary,* $\mathcal{R}e(\lambda) = 0$ *for all* $\lambda \in \sigma(\mathbf{A})$.

Proof: If $\mathbf{A}$ is skew-symmetric, then

$$I = e^{-t\mathbf{A}}e^{t\mathbf{A}} = e^{t\mathbf{A}^T}e^{t\mathbf{A}} = \left(e^{t\mathbf{A}}\right)^T e^{t\mathbf{A}}.$$

This proves that $e^{t\mathbf{A}}$ is an isometry. The spectrum of a skew-symmetric matrix is purely imaginary since

$$(i\mathbf{A})^H = -i\mathbf{A}^T = i\mathbf{A}$$

which means that the matrix $i\mathbf{A}$ is Hermitian in the complexification of $\mathbb{R}^d$. ■

Our goal is now to find discretizations for the ODE $\mathbf{x}' = \mathbf{A}\mathbf{x}$, $\mathbf{A}^T = -\mathbf{A}$ whose discrete evolutions $\mathbf{\Psi}^\tau$ are also isometries (orthogonal). For this purpose, we study again the properties of the rational approximation R of the exponential function.

Theorem 3.36. *Let* R *be a consistent, rational approximation of the exponential function with real coefficients and let* $R(z) \neq \infty$ *for all* z *with* $\mathcal{R}e(z) = 0$. *Moreover, let* R *satisfy*

$$R(z)R(-z) = 1 \quad \underline{\text{(reversibility)}}.$$

If $\mathbf{A}$ *is skew-symmetric, then* $R(\tau\mathbf{A})$ *is an isometry for all* $\tau \in \mathbb{R}$.

Proof: Reversibility $1 = R(z)R(-z)$ and commutativity with the transposition $R(\tau A^T) = (R(\tau A))^T$ are used as in the previous proof. Note that the rational function R can be applied to skew-symmetric matrices, since they have a purely imaginary spectrum and hence R has no poles on the imaginary axis. ■

Example 3.37. *The explicit Euler (*$R(z) = 1 + z$*) and the implicit Euler method (*$R(z) = (1 - z)^{-1}$*) do not satisfy the condition of reversibility. Thus,* $R(\tau\mathbf{A})$ *is no isometry for* $\mathbf{A}^T = -\mathbf{A}$ *! Note that particularly the explicit Euler increases the energy in the system, whereas the implicit one decreases it.*

Example 3.38. *Consider the rational approximation*

$$\begin{aligned} R(z) &= \frac{1+\frac{z}{2}}{1-\frac{z}{2}} = 1 + z + \frac{z^2}{2} + \mathcal{O}(z^3) \\ &= e^z + \mathcal{O}(z^3). \end{aligned}$$

R has consistency order $p = 2$ and fulfills $R(z)R(-z) = 1$. The corresponding method

$$\mathbf{x_{n+1}} = \mathbf{x_n} + \tau \mathbf{A}\left(\frac{\mathbf{x_{n+1}}}{2} + \frac{\mathbf{x_n}}{2}\right)$$

is either called linear implicit Trapezoidal *or* Mid-Point rule *which is an isometry.*

Note that in general implicit methods differ in the non-linear case $\mathbf{x}' = \mathbf{f}(\mathbf{x})$:

- *Trapezoidal rule:* $\mathbf{x_{n+1}} = \mathbf{x_n} + \tau\,[\mathbf{f}(\mathbf{x_{n+1}}) + \mathbf{f}(\mathbf{x_n})]/2$ *no isometry*
- *Mid-point rule:* $\mathbf{x_{n+1}} = \mathbf{x_n} + \tau\,\mathbf{f}([\mathbf{x_{n+1}} + \mathbf{x_n}]/2)$ *isometry*

3.6 Implicit Runge–Kutta Methods

In this section we present numerical methods of higher consistency order $p > 1$ keeping asymptotical stability and in general also isometry. They are called *implicit Runge–Kutta methods*.

3.6.1 Scheme and its Properties

Formally, we allow the Runge–Kutta coefficient matrix $\mathcal{A}$ to be a full matrix.

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

This yields the following form of a method with s levels:

$$\begin{aligned} \mathbf{k_i} &= \mathbf{f}\left(t + c_i\tau, \mathbf{x} + \tau \textstyle\sum_{j=1}^{s} a_{ij}\mathbf{k_j}\right), \quad i = 1, \dots s \quad \in \mathbb{R}^d \\ \mathbf{\Psi}^{t+\tau,t}\mathbf{x} &= \mathbf{x} + \tau \textstyle\sum_{j=1}^{s} b_j \mathbf{k_j} \qquad\qquad\qquad\qquad\quad \in \mathbb{R}^d \end{aligned}$$

Note that the summation runs until s and not until $i-1$.

Remark 3.39. *The RK-method is explicit, if $\mathcal{A}$ is a strictly lower triangle matrix, i.e. $a_{ij} = 0$ for all $j \geq i$. Otherwise, it is implicit.*

In general, implicit methods require a non-linear system of equations to be solved for the determination of the stages $\mathbf{k_i}$. Therefore we need to adress the question of unique solvability of the $\mathbf{k_i}$ for sufficiently small step sizes τ. We rewrite the method using

$$\mathbf{g_i} = \mathbf{x} + \tau \sum_{j=1}^{s} a_{ij}\mathbf{k_j} \qquad \in \mathbb{R}^d,$$

then

$$\begin{aligned} \mathbf{g_i} &= \mathbf{x} + \tau \textstyle\sum_{j=1}^{s} a_{ij}\, \mathbf{f}(t + c_j\tau, \mathbf{g_j}), \quad i = 1, \ldots, s \\ \mathbf{\Psi}^{t+\tau,t}\mathbf{x} &= \mathbf{x} + \tau \textstyle\sum_{j=1}^{s} b_j\, \mathbf{f}(t + c_j\tau, \mathbf{g_j}). \end{aligned}$$

Example 3.40. *Implicit Euler method*

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

$$\begin{aligned} \mathbf{g_1} &= \mathbf{x} + \tau\, \mathbf{f}(t + \tau, \mathbf{g_1}) \\ \mathbf{\Psi}^{t+\tau,t}\mathbf{x} &= \mathbf{x} + \tau\, \mathbf{f}(t + \tau, \mathbf{g_1}) = \mathbf{g_1} \\ \Rightarrow \mathbf{x_{n+1}} &= \mathbf{x_n} + \tau\, \mathbf{f}(t + \tau, \mathbf{x_{n+1}}) \end{aligned}$$

Theorem 3.41. *Let* $\mathbf{f}$ *be continuous,* $\mathbf{f}(t, \mathbf{x})$ *globally* L*–continuous with respect to* $\mathbf{x}$*. For an implicit RK-method there exists* $\tau^* > 0$ *as well as unique continuous functions* $\mathbf{g_i}$*,* $\mathbf{g_i} = \mathbf{g_i}(\tau)$ *on* $(-\tau^*, \tau^*)$ *for every* $(t, \mathbf{x})$*, such that*

- $\mathbf{g_i}(0) = \mathbf{x}, \qquad i = 1, \ldots, s$
- *For* $|\tau| < \tau^*$*, the functions* $\mathbf{g_i}(\tau),\ i = 1, \ldots, s$ *satisfy the implicit equations of the RK-method.*

These $\mathbf{g_i}$ *define a discrete evolution* $\mathbf{\Psi}$ *that is consistent if and only if* $\displaystyle\sum_{i=1}^{s} b_i = 1$*.*

Proof: Since the whole proof is rather technical, we sketch the main ideas. We consider the parameter-dependent fixed-point equation for this system of Runge-Kutta equations. Let $(\mathbf{x}, t)$ be fixed, then

$$\begin{gathered} \mathbf{g} = F(\tau, \mathbf{g}) \\ \text{where} \quad \mathbf{g} = (\mathbf{g_1}, \ldots, \mathbf{g_s}), \quad F(\tau, \mathbf{g}) = (F_1(\tau, \mathbf{g})), \ldots, F_s(\tau, \mathbf{g}))^T \\ \text{and} \quad F_i(\tau, \mathbf{g}) = \mathbf{x} + \tau \sum_{j=1}^{s} a_{ij}\, f(t + c_j\tau, \mathbf{g_j}), \quad i = 1, \ldots, s. \end{gathered}$$

We choose the norm

$$\|\mathbf{g}\| = \max_{1 \le i \le s} \|\mathbf{g_i}\|$$

on $\mathbb{R}^{s \cdot d}$ and define the open set

$$U =]-\tau_*, \tau_*[.$$

Then, by the choice of τ_* the function

$$F :\ U \times \mathbb{R}^{sd} \to \mathbb{R}^{sd} \tag{1}$$

is well-defined and continuous. Moreover, the contractive Lipschitz condition

$$\|F(\tau, \mathbf{g}) - F(\tau, \bar{\mathbf{g}})\| \le \theta \|\mathbf{g} - \bar{\mathbf{g}}\|, \quad \mathbf{g}, \bar{\mathbf{g}} \in \mathbb{R}^{sd},\ \tau \in U, \tag{2}$$

is satisfied which follows directly from the global Lipschitz condition for $\mathbf{f}$:

$$\begin{aligned}\|F(\tau,\mathbf{g})-F(\tau,\bar{\mathbf{g}})\| &\le \tau_*\|\mathcal{A}\|_\infty \max_{1\le j\le s}\|f(t+c_j\tau,\mathbf{g_j})-f(t+c_j\tau,\bar{\mathbf{g}_j})\|\\ &\le \tau_*\|\mathcal{A}\|_\infty L\|\mathbf{g}-\bar{\mathbf{g}}\|\\ &\le \theta\|\mathbf{g}-\bar{\mathbf{g}}\|.\end{aligned}$$

The L–constant $\theta=\tau_*\|\mathcal{A}\|_\infty L$ is less than 1, if τ_* is sufficiently small.

It follows from (1) and (2) and by the parameter-dependent Banach fixed point theorem that for $\tau\in U$ exists a unique $\mathbf{g}(\tau)\in\mathbb{R}^{sd}$ that satisfies

$$\mathbf{g}(\tau)=F(\tau,\mathbf{g}(\tau)).$$

One can additionally prove that these unique solutions define a continuous mapping $\mathbf{g}:\ U\to\mathbb{R}^{sd}$ with $\mathbf{g}(0)=(x,\dots,x)$.

The discrete evolution for $\mathbf{g_i}$ reads:

$$\mathbf{\Psi}^{t+\tau,t}\mathbf{x}=\mathbf{x}+\tau\sum_{j=1}^{s}b_j\mathbf{f}(t+c_j\tau,\mathbf{g_j}(\tau)).$$

It holds that

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\tau}\mathbf{\Psi}^{t+\tau,t}\mathbf{x}|_{\tau=0}&=\lim_{\tau\to0}\frac{\mathbf{\Psi}^{t+\tau,t}\mathbf{x}-\mathbf{\Psi}^{t,t}\mathbf{x}}{\tau}\\ &=\Big(\sum_{j=1}^{s}b_j\Big)f(t,\mathbf{x}).\end{aligned}$$

For $\mathbf{f}\neq0$ we can choose $(t,\mathbf{x})$ such that $\mathbf{f}(t,\mathbf{x})\neq0$. Hence, the discrete evolution $\mathbf{\Psi}$ is consistent if $\sum_{j=1}^{s}b_j=1$ is fulfilled. ■

Remark 3.42. *The assumption $\theta<1$ requires*

$$\tau_*<\frac{1}{L\|\mathcal{A}\|_\infty},$$

i.e. small step sizes $\tau<\tau_*$. But this constraint represents exactly the restriction that we would like to remove in the case of stiff problems. Basically, this results from using a fixed-point iteration for the solution of the non-linear system. An alternative solution procedure (Newton's method) will be presented in the next section.

Example 3.43. *We return to the ODE with a not globally L–continuous right-hand side (no unique $\mathbf{g_i}$!). Consider $x'=\lambda(1-x^2)$, $\lambda>0$ and solve it applying the implicit Euler method,*

$$\Psi^\tau x=g_1,\quad g_1=x+\tau\lambda(1-g_1^2)$$

f is locally, but not globally L–continuous, since

$$\|\lambda(1-x^2)-\lambda(1-y^2)\| = \lambda\|x^2-y^2\| = \lambda\|x+y\|\,\|x-y\| \le L\|x-y\|$$

with $L(x,y) = \lambda\|x+y\|$ bounded for $x, y \in D$, D bounded. The evaluation of $\Psi^\tau x = g_1$ requires the solution of the quadratic equation $g_1 = x + \tau\lambda(1-g_1^2)$. We obtain two solutions

$$g_1^{\pm}(\tau) = \frac{1}{2\lambda\tau}(-1 \pm \sqrt{1+4\tau\lambda(x+\tau\lambda)}).$$

The discrete evolution is prescribed by

$$\Psi^\tau x = g_1^+(\tau).$$

Note that for g_1^+, the term in brackets is of order $\mathcal{O}(\tau\lambda)$ as $\tau \to 0$. Together with the prefactor, $g_1^+(\tau)$ is well-defined in the limit. While for the other solution g_1^- the term is brackets is of order $\mathcal{O}(1)$ as $\tau \to 0$. Hence, together with the prefactor we see that $g_1^-(\tau) \to -\infty$ explodes for $\tau \to 0$!

The discrete evolution is given for all $\tau > 0$, if $x \ge -1$ holds. In this case the solution tends to $x = 1$ asymptotically.

Remark 3.44. *The properties of the explicit RK-methods can be transferred to the implicit ones defined in Th 3.41:*

- *Invariance with respect to autonomization if*

$$c_i = \sum_{j=1}^{s} a_{ij}, \quad i = 1, \dots, s$$

- *Invariance with respect to linearization around a fixpoint*
- *Coefficients of the Butcher array are obtained by solving the required equations that are imposed on the method to ensure a certain consistency order.*

Remark 3.45. *Another approach for the determination of the coefficients and thus the construction of implicit Runge-Kutta-methods is presented in Sec 3.8.*

We focus now on the stability region of the implicit RK-methods. Consider the linear, autonomous system of ODEs

$$\mathbf{x}' = \mathbf{A}\mathbf{x},$$

and the discrete evolution

$$\mathbf{\Psi}^\tau = R(\tau\mathbf{A}).$$

Lemma 3.46. *The stability function R corresponding to an (implicit) RK-method $(\mathbf{b}, \mathcal{A})$ with s stages is given by*

$$R(z) = 1 + z\mathbf{b}^T(\mathbf{I} - z\mathcal{A})^{-1}\mathbf{e}, \qquad \mathbf{e} = (1, \cdots, 1)^T \in \mathbb{R}^s.$$

R can be uniquely decomposed into

$$R(z) = \frac{P(z)}{Q(z)}$$

with $P, Q \in \mathbb{P}_s$ coprime polynomials (i.e. without common divisor) of degree s with $P(0) = Q(0) = 1$.

Proof: Apply the implicit Runge-Kutta-method to the model problem

$$\mathbf{x}' = \lambda\mathbf{x}, \quad \mathbf{x}(0) = 1, \quad \lambda \in \mathbb{C}.$$

Then, we get

$$\mathbf{\Psi}^\tau 1 = R(\tau\lambda) = 1 + \tau \sum_{j=1}^{s} b_j \lambda \mathbf{g_j} \tag{1}$$

$$\mathbf{g_i} = 1 + \tau \sum_{j=1}^{s} a_{ij} \lambda \mathbf{g_j}, \quad i = 1, \ldots, s. \tag{2}$$

With $z = \tau\lambda$ and $\mathbf{g} = (\mathbf{g_1}, \ldots, \mathbf{g_s}) \in \mathbb{R}^s$

$$\overset{(1),(2)}{\Rightarrow} R(z) = 1 + z\mathbf{b}^T\mathbf{g}, \quad \mathbf{g} = \mathbf{e} + z\mathcal{A}\mathbf{g}.$$

Solving with respect to $\mathbf{g}$ yields $\mathbf{g} = (\mathbf{I} - z\mathcal{A})^{-1}\mathbf{e}$ which is the claimed equation for R. If this system is solved by Cramer's rule, we find that

$$\mathbf{g_i} = \frac{P_i}{\det(\mathbf{I} - z\mathcal{A}))}, \quad i = 1, \ldots, s$$

with polynomials $P_i \in \mathbb{P}_{s-1}$. Since $\hat{Q}(z) = \det(\mathbf{I} - z\mathcal{A}) \in \mathbb{P}_s$ with $\hat{Q}(0) = 1$ it follows that

$$R(z) = \frac{\hat{Q}(z) + z\sum_{j=1}^{s} b_j P_j(z)}{\hat{Q}(z)}.$$

The desired result is a quotient of two polynomials once all common divisors have been removed. ■

Example 3.47.

- *Implicit Trapezoidal rule*

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

$$\begin{aligned} \mathbf{\Psi}^{t+\tau,t}\mathbf{x} &= \mathbf{x} + \frac{\tau}{2}\left(\mathbf{f}(t,\mathbf{g_1}) + \mathbf{f}(t+\tau,\mathbf{g_2})\right) = \mathbf{g_2} \\ \mathbf{g_1} &= \mathbf{x} \\ \mathbf{g_2} &= \mathbf{x} + \frac{\tau}{2}\left(\mathbf{f}(t,\mathbf{g_1}) + \mathbf{f}(t+\tau,\mathbf{g_2})\right) \\ \Longrightarrow \mathbf{x_{n+1}} &= \mathbf{x_n} + \frac{\tau}{2}\left(\mathbf{f}(t,\mathbf{x_n}) + \mathbf{f}(t+\tau,\mathbf{x_{n+1}})\right) \end{aligned}$$

- *Implicit Mid-point rule*

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

$$\begin{aligned} \mathbf{\Psi}^{t+\tau,t}\mathbf{x} &= \mathbf{x} + \tau\mathbf{f}(t+\frac{\tau}{2},\mathbf{g_1}) \\ \mathbf{g_1} &= \mathbf{x} + \frac{\tau}{2}\mathbf{f}(t+\frac{\tau}{2},\mathbf{g_1}) \\ \Longrightarrow \mathbf{g_1} &= \frac{\mathbf{x} + \mathbf{\Psi}^{t+\tau,t}\mathbf{x}}{2} \\ \Longrightarrow \mathbf{x_{n+1}} &= \mathbf{x_n} + \tau\,\mathbf{f}\left(t+\frac{\tau}{2}, \frac{\mathbf{x_n}+\mathbf{x_{n+1}}}{2}\right) \end{aligned}$$

In both cases the stability function $R(z)$ *is given by*

$$R(z) = \frac{1+\frac{z}{2}}{1-\frac{z}{2}} = 1 + z\mathbf{b}^T(\mathbf{I} - z\mathcal{A})^{-1}\mathbf{e}$$

with $\mathbf{b}$ *and* $\mathcal{A}$ *taken from the respective Butcher schemes. Both methods have consistency order* $p = 2$ *and stability domain* $S = \mathbb{C}^-$*, since*

$$\left|1+\frac{z}{2}\right| < \left|1-\frac{z}{2}\right| \implies \mathcal{R}e(z) < 0.$$

Similarly to the explicit case, the maximal consistency order of an implicit RK-method can be estimated.

Lemma 3.48. *If an implicit RK-method with* s *stages has the consistency order* $p \in \mathbb{N}$ *for the right-hand side* $\mathbf{f} \in \mathcal{C}^\infty$*. Then*

$$p \leq 2s.$$

Proof: We apply the Runge-Kutta-method to the initial value problem

$$\mathbf{x}' = \mathbf{x}, \quad \mathbf{x}(0) = 1.$$

This gives

$$\mathbf{\Phi}^\tau 1 - \mathbf{\Psi}^\tau 1 = e^\tau - R(\tau) = \mathcal{O}(\tau^{p+1}),$$

i.e. R is a rational approximation of order p. By lemma 3.46 the rational function is the quotient of two polynomials with

$$\deg P \leq s \quad \text{and} \quad \deg Q \leq s.$$

Hence, lemma 3.18 implies that $p \leq 2s$, since

$$p \leq \deg P + \deg Q.$$

■

Remark 3.49. *The bound $p \leq 2s$ cannot be improved. In fact, there exists a family of implicit Runge-Kutta-methods with $p = 2s$ called Gauss-methods (Section 3.8).*

3.6.2 Newton Method for Solving Non-linear Systems

Applying implicit methods for the discretization of ordinary differential equations, we come in general up with a non-linear system of equations. For solving them, the Newton method is an appropriate tool.

Consider the following non-linear system of equations

$$\mathbf{g_i} = \mathbf{x} + \tau \sum_{j=1}^{s} a_{ij}\, \mathbf{f}(t + c_j\tau, \mathbf{g_j}), \quad i = 1, \ldots, s$$

$$\mathbf{\Psi}^{t+\tau,t}\mathbf{x} = \mathbf{x} + \tau \sum_{j=1}^{s} b_j\, \mathbf{f}(t + c_j\tau, \mathbf{g_j}),$$

with $\mathbf{g_i} \in \mathbb{R}^d$, $i = 1, \ldots, s$. This is a system in the s vectors $\mathbf{g_i} \in \mathbb{R}^d$ and hence a system of $s \cdot d$ equations in $s \cdot d$ unknowns. By introducing the difference $\mathbf{z_i} := \mathbf{g_i} - \mathbf{x}$, it can be rewritten as

$$\mathbf{z_i} = \tau \sum_{j=1}^{s} a_{ij}\, \mathbf{f}(t + c_j\tau, \mathbf{x} + \mathbf{z_j}), \quad i = 1, \ldots, s$$

$$\mathbf{\Psi}^{t+\tau,t}\mathbf{x} = \mathbf{x} + \tau \sum_{j=1}^{s} b_j\, \mathbf{f}(t + c_j\tau, \mathbf{x} + \mathbf{z_j}).$$

This is useful since the differences $\mathbf{g_i} - \mathbf{x} = \mathcal{O}(\tau)$ are expected to be small (danger of cancellations!). We now solve these non-linear equations by Newton's method. Therefore, we apply the following compact notation:

$$\mathbf{z} = \begin{pmatrix} \mathbf{z_1} \\ \vdots \\ \mathbf{z_s} \end{pmatrix} \in \mathbb{R}^{sd}$$

$$\mathbf{F}(\mathbf{z}) = \mathbf{z} - \tau \begin{pmatrix} \sum_{j=1}^{s} a_{1j}\mathbf{f}(t + c_j\tau, \mathbf{x} + \mathbf{z_j}) \\ \vdots \\ \sum_{j=1}^{s} a_{sj}\mathbf{f}(t + c_j\tau, \mathbf{x} + \mathbf{z_j}) \end{pmatrix} = \mathbf{0}$$

Since the components, i.e. the differences $\mathbf{z_i}$, are small, it is reasonable to choose here the starting value $\mathbf{z}^{(0)} = \mathbf{0}$ for the Newton iteration. The iteration reads

$$\begin{aligned} \mathbf{z}^{(0)} &= \mathbf{0} \\ \mathbf{DF}(\mathbf{z}^{(k)})\ \Delta\mathbf{z}^{(k)} &= -\mathbf{F}(\mathbf{z}^{(k)}) \\ \mathbf{z}^{(k+1)} &= \mathbf{z}^{(k)} + \Delta\mathbf{z}^{(k)}, \quad k = 0, 1, \ldots. \end{aligned}$$

Each iteration step requires the solving of a linear system of equations in $\mathbb{R}^{sd}$ with the Jacobian matrix

$$\mathbf{DF}(\mathbf{z}) = \begin{pmatrix} \mathbf{I} - \tau a_{11}\partial_2\mathbf{f}(t + c_1\tau, \mathbf{x} + \mathbf{z_1}) & \cdots & -\tau a_{1s}\partial_2\mathbf{f}(t + c_s\tau, \mathbf{x} + \mathbf{z_s}) \\ \vdots & & \vdots \\ -\tau a_{s1}\partial_2\mathbf{f}(t + c_1\tau, \mathbf{x} + \mathbf{z_1}) & \cdots & \mathbf{I} - \tau a_{ss}\partial_2\mathbf{f}(t + c_s\tau, \mathbf{x} + \mathbf{z_s}) \end{pmatrix} \in \mathbb{R}^{sd\times sd}$$

where $\mathbf{I} \in \mathbb{R}^{d\times d}$ denotes the identity matrix and $\partial_2\mathbf{f}(t, \mathbf{x})$ the partial derivatives of $\mathbf{f}$ with respect to the second argument.

Remark 3.50. *Newton's method converges qudratically (locally) for sufficiently small step sizes. But this fast convergence has a high price since in each iteration step the jacobian matrix has to be evaluated for s different arguments. This is computationally extremely costly and can be avoided by using the* simplified Newton's method *which is only of linear convergence. We might replace $\mathbf{DF}(\mathbf{z})$ by*

$$\mathbf{DF}(\mathbf{z}^{(0)}) = \begin{pmatrix} \mathbf{I} - \tau a_{11}\mathbf{J} & \cdots & -\tau a_{1s}\mathbf{J} \\ \vdots & & \vdots \\ -\tau a_{s1}\mathbf{J} & \cdots & I - \tau a_{ss}\mathbf{J} \end{pmatrix}$$

with $\mathbf{J} = \partial_2\mathbf{f}(t, \mathbf{x})$.

The simplified Newton iteration reads then

$$\begin{aligned} \mathbf{z}^{(0)} &= \mathbf{0} \\ \mathbf{DF}(\mathbf{z}^{(0)})\ \Delta\mathbf{z}^{(k)} &= -\mathbf{F}(\mathbf{z}^{(k)}) \\ \mathbf{z}^{(k+1)} &= \mathbf{z}^{(k)} + \Delta\mathbf{z}^{(k)}, \quad k = 0, 1, \ldots. \end{aligned}$$

In spite of this simplification, a linear system in $\mathbb{R}^{sd}$ has to be solved for each iteration step. But now the respective matrix is the same in all iterations. This reduces the computational effort drastically and could be realized by a LR-decomposition.
$\Longrightarrow$ Effort $\mathcal{O}((s \cdot d)^3)$

Remark 3.51. *Note that the Newton iteration needs an appropriate stopping criterion. Assume there exists some contraction with $\theta < 1$:*

$$\|\Delta\mathbf{z}^{(k+1)}\| \leq \theta\|\Delta\mathbf{z}^{(k)}\|.$$

We know from Banach's fixed point theorem that the error of iterate $\mathbf{z}^{(k+1)}$ *satisfies*

$$\|\mathbf{z} - \mathbf{z}^{(k+1)}\| \leq \frac{\theta}{1-\theta}\|\Delta\mathbf{z}^{(k)}\|.$$

In a next step, the unknown contraction factor θ *is replaced by*

$$\theta_k = \frac{\|\Delta\mathbf{z}^{(k)}\|}{\|\Delta\mathbf{z}^{(k-1)}\|}, \quad k = 1, 2, \ldots.$$

Then, the stopping criterion becomes

$$\frac{\theta_k}{1-\theta_k}\|\Delta\mathbf{z}^{(k)}\| \leq \sigma \cdot TOL, \quad \sigma \ll 1,$$

where TOL *is the desired tolerance. Note that in the first step for* $k = 0$ *we have to choose* θ_0 *heuristically such that the iteration stops immediately when applied to a linear problem.*

3.7 Linearly Implicit Runge-Kutta Methods

By using a simple idea, we can avoid the costly solving of nonlinear systems of equations for implicit RK-methods. One can use the so-called linearly implicit RK-methods. We consider the autonomous differential equation

$$x' = f(x), \qquad f \in \mathcal{C}^1$$

which is equivalent to

$$\begin{aligned} x'(t) &= Jx(t) + (f(x(t)) - Jx(t)), \\ J &= Df(\mathbf{x}) \end{aligned} \tag{3.2}$$

to compute one step $\mathbf{\Psi}^{t+\tau,t}\mathbf{x}$ of a discrete flow. The right hand side of (3.2) consists of a linear $Jx(t)$ and a nonlinear part $(f(x(t)) - Jx(t))$. The base idea is to solve the linear part implicitly and the nonlinear one explicitly resulting in a scheme like this

$$\begin{aligned} \mathbf{\Psi}^{\tau}\mathbf{x} &= \mathbf{x} + \tau \sum_{j=1}^{s} b_j \mathbf{k_j}, \\ \mathbf{k_i} &= J \cdot \left(\mathbf{x} + \tau \sum_{j=1}^{i} \beta_{ij} \mathbf{k_j} \right) + \\ &\qquad \left(f\left(\mathbf{x} + \tau \sum_{j=1}^{i-1} \alpha_{ij} \mathbf{k_j} \right) - J \cdot \left(\mathbf{x} + \tau \sum_{j=1}^{i-1} \alpha_{ij} \mathbf{k_j} \right) \right), \ i = 1, \ldots, s. \end{aligned}$$

The reduction to an upper limit i instead of s in the sum yields the ability to solve the resulting linear systems of equations successively. We further simplify the scheme by specifically choosing $\beta_{ii} = \beta$ for $i = 1, \dots, s$. Then, solving for k_i yields

$$\begin{aligned} J &= Df(\mathbf{x}), \\ (I - \tau\beta J)\mathbf{k_1} &= f(\mathbf{x}), \\ (I - \tau\beta J)\mathbf{k_i} &= \tau \sum_{j=1}^{i-1} (\beta_{ij} - \alpha_{ij}) J\mathbf{k_j} + f(\mathbf{x} + \tau \sum_{j=1}^{i-1} \alpha_{ij}\mathbf{k_j}), \quad i = 2, \dots, s, \\ \mathbf{\Psi}^\tau \mathbf{x} &= \mathbf{x} + \tau \sum_{j=1}^{s} b_j \mathbf{k_j}. \end{aligned}$$

Thus in each step we have to solve s linear systems of equations in $\mathbb{R}^d$. Because of $\beta_{ii} = \beta$ we once need to compute a LR-decomposition of $(I - \tau\beta J)$. This computation has only complexity $\mathcal{O}(d^3)$ and is thus not costly compared to implicit RK-methods. We show now the solvabilty of the systems of equations.

Lemma 3.52. *Let $\beta \geq 0$ and $J \in \mathbb{R}^{d\times d}$. The matrix $I - \tau\beta J$ is invertible for $0 < \tau < \tau_*$ given by*

$$\tau_* = \begin{cases} \infty & \text{if} \quad \nu(J) \leq 0, \\ \frac{1}{\beta\nu(J)} & \text{if} \quad \nu(J) > 0, \text{where} \quad \nu(J) = \max_{\lambda\in\sigma(J)} \text{Re } (\lambda). \end{cases}$$

Proof: Let $\lambda \in \sigma(J)$ and $0 < \tau < \tau_*$. We need to show that $1 - \tau\beta\lambda \neq 0$. For $\text{Re } (\lambda) \leq 0$ we have

$$\text{Re } (1 - \tau\beta\lambda) = 1 - \tau\beta \,\text{Re } (\lambda) \geq 1, \quad \forall\tau > 0.$$

For

$$0 < \text{ Re } (\lambda) \leq \nu(J), \qquad 0 < \tau < \tau_* = \frac{1}{\beta\nu(J)}$$

we have

$$\begin{aligned} 1 - \tau\beta\nu(J) &> 1 - \tau_*\beta\nu(J) = 0 \\ \Rightarrow \text{ Re } (1 - \tau\beta\lambda) &\geq 1 - \tau\beta\nu(J) > 0. \end{aligned}$$

■

Remark 3.53. *We do not have any restriction to the step size for eigenvalues with nonpositive real part. In this case one may choose $\tau_* = \infty$.*

Remark 3.54. *The case of non-autonomous ODEs follows analogously.*

For autonomous linear ODEs $f(x) = Jx$, the linearly implicit methods $(\mathbf{b}, \mathcal{A}, \mathcal{B})$ coincide with an RK-method given by $(\mathbf{b}, \mathcal{B})$. Hence, the stability function is given by

$$R(z) = 1 + z\mathbf{b}^T(I - z\mathcal{B})^{-1}\mathbf{1}.$$

Example 3.55. *The linearly implicit Euler method can be written in the form*

$$\begin{aligned}(I - \tau J)\mathbf{k_1} &= f(\mathbf{x}),\\ \mathbf{\Psi}^\tau \mathbf{x} &= \mathbf{x} + \tau \mathbf{k_1}.\end{aligned}$$

Thus it is a one-stage linearly implicit RK-method with

$$s = 1, \quad \mathbf{b}_1 = 1, \quad \beta = 1.$$

One gets

$$\mathbf{\Psi}^\tau \mathbf{x} = \mathbf{x} + \tau \left(I - \tau D f(\mathbf{x})\right)^{-1} f(\mathbf{x}).$$

The scheme is of consistency order one and the stability function equals the stability function of the implicit Euler scheme: applying the method to the test equation $f(x) = \lambda x$ yields the discrete evolution

$$\mathbf{\Psi}^\tau \mathbf{x} = \mathbf{x} + \tau\lambda(1 - \tau\lambda)^{-1}\mathbf{x} = \left(1 + \frac{\tau\lambda}{1 - \tau\lambda}\right)\mathbf{x} = R(\tau\lambda)\mathbf{x}$$

with

$$R(z) = 1 + \frac{z}{1 - z} = \frac{1}{1 - z}.$$

Remark 3.56. *The in the literature also common names for this linearly implicit RK-methods are Rosenbrock methods, Rosenbrock-Wanner methods with the abbreviation ROW-methods.*

3.8 Collocation Methods

In this section, we construct implicit RK-methods by using the idea of collocation. Consider the system of ordinary differential equations $\mathbf{x}' = \mathbf{f}(t, \mathbf{x})$. For $(t, \mathbf{x})$ and τ we determine a discrete evolution $\mathbf{\Psi}^{t+\tau,t}\mathbf{x}$. For this purpose, let $\mathbf{u} = \mathbf{u}(t) \in \mathbb{P}_s$ be a polynomial of degree s that satisfies ('collocates') the system of ODEs at s different given points:

- $\mathbf{u}(t) = \mathbf{x}$
- $\mathbf{u}'(t + c_i\tau) = \mathbf{f}(t + c_i\tau, \mathbf{u}(t + c_i\tau)), \quad i = 1, \ldots, s$
- $\mathbf{\Psi}^{t+\tau,t}\mathbf{x} = \mathbf{u}(t + \tau)$

Hence, the method is prescribed by $\mathbf{c} = (c_1, \ldots, c_s)$. We require $0 \le c_1 < \cdots < c_s \le 1$ and thus $t + c_i\tau \in [t, t+\tau]$. This means we impose $s+1$ conditions on the polynomial of degree s.

Instead of analyzing the system of conditions in detail, we interpret the collocation approach as a $s-$stage implicit Runge-Kutta-method. Assume that there exists a solution $\mathbf{u} \in \mathbb{P}_s$. Let now $\{L_1, \dots, L_s\}$ be the Lagrange basis of $\mathbb{P}_{s-1}$ with respect to $c_1, \dots, c_s$, i.e. it holds

$$L_i(c_j) = \delta_{ij}, \qquad i, j = 1, \dots, s.$$

For brevity, let

$$\begin{aligned} \mathbf{k_i} &= \mathbf{u}'(t + c_i\tau), \qquad i = 1, \dots, s \\ \mathbf{u}'(t + \theta\tau) &= \sum_{j=1}^{s} \mathbf{k_j} L_j(\theta), \quad \text{since } \mathbf{u}' \in \mathbb{P}_{s-1}. \end{aligned}$$

Applying $\mathbf{u}(t) = \mathbf{x}$, integration yields

$$\mathbf{u}(t + c_i\tau) = \mathbf{x} + \tau \int_0^{c_i} \mathbf{u}'(t + \theta\tau)\ \mathrm{d}\theta = \mathbf{x} + \tau \sum_{j=1}^{s} a_{ij} \mathbf{k_j}$$

with

$$a_{ij} = \int_0^{c_i} L_j(\theta)\ \mathrm{d}\theta, \quad i, j = 1, \dots, s. \tag{3.3}$$

The collocation conditions lead to

$$\mathbf{k_i} = \mathbf{f}\Big(t + c_i\tau, \underbrace{\mathbf{x} + \tau \sum_{j=1}^{s} a_{ij} \mathbf{k_j}}_{\mathbf{u}(t+c_i\tau)}\Big), \quad i = 1, \dots, s. \tag{3.4}$$

Analogously, it follows

$$\mathbf{\Psi}^{t+\tau,t}\mathbf{x} = \mathbf{u}(t + \tau) = \mathbf{x} + \tau \int_0^1 \mathbf{u}'(t + \theta\tau)\, \mathrm{d}\theta \ = \ \mathbf{x} + \tau \sum_{j=1}^{s} b_j \mathbf{k_j} \tag{3.5}$$

with

$$b_j = \int_0^1 L_j(\theta)\ \mathrm{d}\theta. \tag{3.6}$$

The coefficients $\mathcal{A} = (a_{ij})$, $\mathbf{b} = (b_1, \dots, b_s)$ of Eqs (3.3), (3.6) depend only on $c_1, \dots, c_s$.

Remark 3.57. *From Eqs* (3.4), (3.5), *the form of an implicit RK-method* $(\mathbf{b}, \mathbf{c}, \mathcal{A})$ *can be recognized.*

Conversely, if the RK-method that is prescribed by $(\mathbf{b}, \mathbf{c}, \mathcal{A})$ according to the procedure above has a solution with $\mathbf{k_1}, \ldots, \mathbf{k_s}$, then we can reverse all computations and obtain a collocation polynomial

$$\mathbf{u}(t + \theta\tau) = \mathbf{x} + \tau \sum_{j=1}^{s} \mathbf{k_j} \int_0^\theta L_j(\eta)\ \mathrm{d}\eta.$$

Theorem 3.58. *The collocation polynomial corresponding to the vector* $\mathbf{c}$ *is equivalent to the implicit RK-method* $(\mathbf{b}, \mathbf{c}, \mathcal{A})$ *defined by Eqs* (3.3), (3.6).

Remark 3.59. *The degree of freedom is thus just* s, *i.e* $(c_1, \ldots, c_s)$ *instead of* $2s+s^2$ *values for the Butcher array* $(\mathbf{b}, \mathbf{c}, \mathcal{A})$.

By this procedure, we implicitly get a number of conditions that a RK-method has to satisfy to be consistent with order s.

Lemma 3.60. *The coefficients of an implicit RK-method* $(\mathbf{b}, \mathbf{c}, \mathcal{A})$ *defined by collocation fulfill*

- $\displaystyle\sum_{j=1}^{s} b_j c_j^{k-1} = \frac{1}{k}, \quad k = 1, \ldots, s$
- $\displaystyle\sum_{j=1}^{s} a_{ij} c_j^{k-1} = \frac{c_i^k}{k}, \quad i, k = 1, \ldots, s$

In particular, the method is consistent and invariant under autonomization if

$$c_i = \sum_{j=1}^{s} a_{ij} \quad \textit{and} \quad \sum_{j=1}^{s} b_j = 1.$$

Proof: Since

$$b_j = \int_0^1 L_j(\theta)\ \mathrm{d}\theta,$$

we have

$$\sum_{j=1}^{s} b_j c_j^{k-1} = \sum_{j=1}^{s} \int_0^1 c_j^{k-1} L_j(\theta)\ \mathrm{d}\theta = \int_0^1 \theta^{k-1}\,\mathrm{d}\theta = \frac{1}{k}\ .$$

where

$$\theta^{k-1} = \sum_{j=1}^{s} c_j^{k-1} L_j(\theta)$$

is used for $k = 1, \ldots, s$. The last equation is valid because the Lagrange polynomials form a basis of $\mathbb{P}_{s-1}$. From $k = 1$ we conclude the consistency $\sum_{j=1}^{s} b_j = 1$.

The definition of a_{ij} yields

$$\sum_{j=1}^{s} a_{ij} c_j^{k-1} = \sum_{j=1}^{s} \int_0^{c_i} c_j^{k-1} L_j(\theta) \, \mathrm{d}\theta = \int_0^{c_i} \theta^{k-1} \, \mathrm{d}\theta = \frac{c_i^k}{k}.$$

For $k = 1$ we receive the invariance under autonomization $\sum_{j=1}^{s} a_{ij} = c_i$. ■

Remark 3.61. *The consequence of condition*

$$\sum_{j=1}^{s} b_j c_j^{k-1} = \frac{1}{k}, \qquad k = 1, \ldots, s$$

is that the quadrature rule

$$\sum_{j=1}^{s} b_j \varphi(c_j) \approx \int_0^1 \varphi(t) \, \mathrm{d}t \tag{3.7}$$

is exact for polynomials $\varphi \in \mathbb{P}_{s-1}$. Thus, the quadrature rule is exact at least up to order s:

$$\left| \int_t^{t+\tau} \varphi(s) \, \mathrm{d}s - \tau \sum_{j=1}^{s} b_j \varphi(t + \tau c_j) \right| = \mathcal{O}(\tau^{s+1}).$$

We focus now on the consistency order of a RK-method defined by collocation. It is given by the order of the quadrature rule.

Theorem 3.62. *An implicit RK-method $(\mathbf{b}, \mathbf{c}, \mathcal{A})$ generated by collocation has the consistency order p for the right-hand side $\mathbf{f} \in \mathcal{C}^p$, if and only if the nodes $\mathbf{c}$ and the corresponding weights $\mathbf{b}$ prescribe a quadrature rule that possesses the order p for p-times differentiable functions, i.e.*

$$\left| \int_t^{t+\tau} \varphi(s) \, \mathrm{d}s - \tau \sum_{j=1}^{s} b_j \varphi(t + \tau c_j) \right| = \mathcal{O}(\tau^{p+1}), \quad p \geq 1.$$

Remark 3.63. *For the possible consistency order p of an implicit RK-method defined by collocation, the following estimate holds:*

From Numerical Analysis we know that a quadrature rule with s nodes is exact for polynomials being at most P_{2s-1}. (This is the Gauss–Legendre quadrature rule). This means the maximal consistency order is $p = 2s$.

According to Lem 3.60 / Rem 3.61 in contrast, a quadrature rule that is given by collocation is exact at least for polynomials of degree $s - 1$. This means the minimal consistency order is $p = s$.

Summing up,

$$\boxed{s \leq p \leq 2s.}$$

Remark 3.64. *To get a method of consistency order $p = 2s$ for s nodes $c_1, \cdots c_s$, the Gauss–Legendre quadrature rule has to be applied (e.g. $s = 1$: Mid-point rule).*

In general: If a quadrature rule with s nodes c_i, $i = 1, \ldots, s$ is exact for polynomials of degree $2s - 1$, then the nodes are uniquely determined by the ones of the Gauss–Legendre quadrature, i.e. by the zeros of the Legendre polynomials of degree s on $[0, 1]$. If these nodes are used, then the method is called Gauss method.

Theorem 3.65. *For $\mathbf{f} \in \mathcal{C}^{2s}$, the Gauss method with s levels has consistency order $p = 2s$.*

Example 3.66. *Consider*

$s = 1, \quad p = 2$

$c_1 = \frac{1}{2}$

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

$\mathbf{x_{n+1}} = \mathbf{x_n} + \tau \mathbf{f}\left(\frac{\mathbf{x_n} + \mathbf{x_{n+1}}}{2}\right)$

implicit Mid-point rule

$s = 2, \quad p = 4$

$c_1 = \frac{1}{2} - \frac{\sqrt{3}}{6} \quad c_2 = \frac{1}{2} + \frac{\sqrt{3}}{6}$

$$\begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

In the following, we show that Gauss methods are

- A–stable
- isometry preserving.

To prove A–stability we provide some fundamental definitions and theorems. In particular, we introduce the definitions dissipative and B–stable and restrict on autonomous systems

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

Definition 3.67. *The function $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^d$ is called* dissipative *with respect to the scalar product $\langle \cdot, \cdot \rangle$, if*

$$\langle \mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \leq 0 \quad \forall \mathbf{x}, \mathbf{y}, \ \mathbf{x} \neq \mathbf{y}.$$

Example 3.68. *Consider $\mathbf{f}(\mathbf{x}) = -\lambda \mathbf{x}, \ \lambda > 0$.*

Theorem 3.69. *Let $\mathbf{x}' = \mathbf{f}(\mathbf{x})$ with L–continuous $\mathbf{f}$ and $\| \cdot \| := \sqrt{\langle \cdot, \cdot \rangle}$. The continuous evolution $\mathbf{\Phi}$ satisfies*

$$\|\mathbf{\Phi}^t \mathbf{x} - \mathbf{\Phi}^t \mathbf{y}\| \leq \|\mathbf{x} - \mathbf{y}\|, \qquad t > 0, \ \forall \mathbf{x}, \mathbf{y}, \ \mathbf{x} \neq \mathbf{y},$$

if $\mathbf{f}$ is dissipative .

Proof: Let

$$q(t) := \|\mathbf{\Phi}^t\mathbf{x} - \mathbf{\Phi}^t\mathbf{y}\|^2 = \langle \mathbf{\Phi}^t\mathbf{x} - \mathbf{\Phi}^t\mathbf{y}, \mathbf{\Phi}^t\mathbf{x} - \mathbf{\Phi}^t\mathbf{y}\rangle.$$

Proof follows immediately by differentiation

$$q'(t) = 2\langle \mathbf{f}(\mathbf{\Phi}^t\mathbf{x}) - \mathbf{f}(\mathbf{\Phi}^t\mathbf{y}), \mathbf{\Phi}^t\mathbf{x} - \mathbf{\Phi}^t\mathbf{y}\rangle.$$

For dissipative $\mathbf{f}$ integration (fundamental theorem of calculus) yields

$$q(t) = q(0) + 2\int_0^t \langle \mathbf{f}(\mathbf{\Phi}^s\mathbf{x}) - \mathbf{f}(\mathbf{\Phi}^s\mathbf{y}), \mathbf{\Phi}^s\mathbf{x} - \mathbf{\Phi}^s\mathbf{y}\rangle \ \mathrm{d}s \leq q(0).$$

■

Numerical methods that inherit this property are called B–stable.

Definition 3.70. *A method prescribed by its discrete evolution* $\mathbf{\Psi}$ *for sufficiently smooth, dissipative functions* $\mathbf{f}$ *is called* B–stable*, if*

$$\|\mathbf{\Psi}^\tau\mathbf{x} - \mathbf{\Psi}^\tau\mathbf{y}\| \leq \|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y}, \ \mathbf{x} \neq \mathbf{y}, \ \tau > 0.$$

Theorem 3.71. *Gauss methods are B–stable.*

Proof: Let $\mathbf{f}$ be dissipative and sufficiently smooth. Let $\mathbf{u}, \mathbf{v} \in \mathbb{P}_s$ be the collocation polynomials corresponding to

$$\begin{aligned} \mathbf{u}(0) &= \mathbf{x}, & \mathbf{u}(\tau) &= \mathbf{\Psi}^\tau\mathbf{x} \\ \mathbf{v}(0) &= \mathbf{y}, & \mathbf{v}(\tau) &= \mathbf{\Psi}^\tau\mathbf{y} \end{aligned}$$

with $\mathbf{x} \neq \mathbf{y}$. Stating

$$q(\theta) = \|\mathbf{u}(\theta\tau) - \mathbf{v}(\theta\tau)\|^2 \in \mathbb{P}_{2s},$$

we obtain

$$\|\mathbf{\Psi}^\tau\mathbf{x} - \mathbf{\Psi}^\tau\mathbf{y}\|^2 = q(1) = q(0) + \int_0^1 q'(\theta) \ \mathrm{d}\theta = \|\mathbf{x} - \mathbf{y}\|^2 + \int_0^1 q'(\theta) \ \mathrm{d}\theta.$$

The discrete evolution $\mathbf{\Psi}$ is B–stable, if $\int_0^1 q'(\theta) \ \mathrm{d}\theta \leq 0$ holds. This we will prove in the following. Applying the Gauss quadrature yields ($q' \in \mathbb{P}_{2s-1}$)

$$\int_0^1 q'(\theta) \ \mathrm{d}\theta = \sum_{j=1}^s b_j q'(c_j).$$

The collocation conditions

$$\mathbf{u}'(c_j\tau) = \mathbf{f}(\mathbf{u}(c_j\tau)), \qquad \mathbf{v}'(c_j\tau) = \mathbf{f}(\mathbf{v}(c_j\tau)),$$

lead to

$$\begin{aligned} q'(c_j) &= 2\tau\langle \mathbf{u}'(c_j\tau) - \mathbf{v}'(c_j\tau), \mathbf{u}(c_j\tau) - \mathbf{v}(c_j\tau)\rangle \\ &= 2\tau\langle \mathbf{f}(\mathbf{u}(c_j\tau)) - \mathbf{f}(\mathbf{v}(c_j\tau)), \mathbf{u}(c_j\tau) - \mathbf{v}(c_j\tau)\rangle \\ &\leq 0, \quad \text{for } j = 1, \ldots, s, \end{aligned}$$

since $\mathbf{f}$ is dissipative. As the weights b_j of the Gauss quadrature are positive, we obtain the desired result $\int_0^1 q'(\theta)\ \mathrm{d}\theta \leq 0$. ■

Example 3.72. *The Mid-point rule is B–stable, while the implicit Trapezoidal rule is not B–stable.*

Now, we can show the A–stability of Gauss methods:

Lemma 3.73. *B–stable RK-methods are A–stable.*

Proof: Consider the complex IVP

$$x' = \lambda x, \quad x(0) = 1, \quad \mathcal{R}e(\lambda) < 0.$$

The right-hand side of the problem is dissipative. This follows from the consideration of the the equivalent real system

$$x = u + iv, \qquad \lambda = \alpha + i\beta,$$

$$\begin{pmatrix} u \\ v \end{pmatrix}' = \begin{pmatrix} \alpha & -\beta \\ \beta & \alpha \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = A \begin{pmatrix} u \\ v \end{pmatrix}.$$

Since $\alpha = \mathcal{R}e(\lambda) < 0$, we obtain the dissipative property of the linear mapping $x \mapsto Ax$:

$$\langle Ax, x\rangle = \alpha(u^2 + v^2) < 0 \quad \text{for } x \neq 0.$$

B–stable methods have discrete evolutions $\Psi^\tau = R(\tau\lambda)$ that satisfy

$$|R(\tau\lambda)| < 1, \quad \forall \tau > 0$$

for the linear differential equation above. Using particularly $\tau = 1$, we get

$$|R(\lambda)| < 1 \ \Rightarrow\ \lambda \in S.$$

Since $\lambda \in \mathbb{C}^-$ is arbitrary, we obtain $\mathbb{C}^- \subset S$ and thus the A–stability of the method. ■

At last we show the inheritance of the isometry property:

Theorem 3.74. *Let* $\mathbf{f}$ *be* L*–continuous and let* $\mathbf{x}' = \mathbf{f}(\mathbf{x})$ *satisfy*

$$\|\mathbf{\Phi}^t\mathbf{x}\|_2 = \|\mathbf{x}\|_2.$$

Each Gauss method generates a discrete evolution $\mathbf{\Psi}$ *with*

$$\|\mathbf{\Psi}^t\mathbf{x}\|_2 = \|\mathbf{x}\|_2.$$

Proof: Consider the collocation polynomial $\mathbf{u} \in \mathbb{P}_s$ with

$$\mathbf{u}(0) = \mathbf{x}, \quad \mathbf{u}(\tau) = \mathbf{\Psi}^\tau\mathbf{x}.$$

Let

$$q(\theta) = \|\mathbf{u}(\theta\tau)\|_2^2 \in \mathbb{P}_{2s},$$

then

$$\|\mathbf{\Psi}^\tau\mathbf{x}\|_2^2 = q(1) = q(0) + \int_0^1 q'(\theta)\ \mathrm{d}\theta = \|\mathbf{x}\|_2^2 + \int_0^1 q'(\theta)\ \mathrm{d}\theta,$$

holds. For the inheritance of isometry we need the disappearance of the integral term

$$\int_0^1 q'(\theta)\ \mathrm{d}\theta = 0,$$

which can be proved as follows. The Gauss quadrature yields ($q' \in \mathbb{P}_{2s-1}$)

$$\int_0^1 q'(\theta)\ \mathrm{d}\theta = \sum_{j=1}^{s} b_j q'(c_j).$$

The collocation conditions

$$\mathbf{u}'(c_j\tau) = \mathbf{f}(\mathbf{u}(c_j\tau))$$

in combination with

$$0 = \frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{\Phi}^t\mathbf{x}\|^2 = 2\langle\mathbf{\Phi}^t\mathbf{x}, \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{\Phi}^t\mathbf{x}\rangle = 2\langle\mathbf{\Phi}^t\mathbf{x}, \mathbf{f}(\mathbf{\Phi}^t\mathbf{x})\rangle$$

or $0 = 2\langle\mathbf{x}, \mathbf{f}(\mathbf{x})\rangle$ if $t = 0$ is chosen, lead to

$$q'(c_j) = 2\tau\,\langle\mathbf{u}(c_j\tau), \mathbf{u}'(c_j\tau)\rangle = 2\tau\,\langle\mathbf{u}(c_j\tau), \mathbf{f}(u(c_j\tau)\rangle = 0.$$

Consequently, $\int_0^1 q'(\theta)\ \mathrm{d}\theta = 0$ is valid. ■

4 Multistep methods

Throughout the previous chapters we have only considered one-step methods, meaning that the computation of $\mathbf{x}_\Delta(t_{j+1})$ solely depends on $\mathbf{x}_\Delta(t_j)$. Now we want to give a brief overview on multi-step methods considering further time evaluations. The motivation of multi-step methods is to get a higher order of consistency with relatively few function evaluations.

First, we want to start with some first attempts to derive such methods. Afterwards, we present a broader class of numerical multi-step methods and discuss the consistency and convergence similar to one-step methods.

4.1 Derivation through numerical differentiation

Recall that we initially derived the first schemes by using numerical approximation of the derivatives. For example the first order backward approximation of the derivative, i.e.,

$$\mathbf{f}(t+\tau, \mathbf{x}(t)) \approx \frac{\mathbf{x}(t+\tau) - \mathbf{x}(t)}{\tau},$$

gives us the implicit Euler scheme:

$$\mathbf{x}_{j+1} = \mathbf{x}_j + \tau \mathbf{f}(t_{j+1}, \mathbf{x}_{j+1}).$$

Now, we want to derive a k-step method. Therefore, we consider the points $(t_{j-k+\ell}, \mathbf{x}_{j-k+\ell})$ for $\ell = 1, \dots, k$. With these points we can construct the interpolation polynomial $p \in \mathbb{P}_k$ with

$$p(t_{j-k+\ell}) = \mathbf{x}_{j-k+\ell} \text{ for } \ell = 1, \dots, k+1.$$

Since the value $\mathbf{x}_{j+1}$ is included in the interpolation polynomial, the scheme is implicit. Furthermore, the derivative of $\mathbf{x}'(t)$ can be approximated by p'. As we want to obtain the new value $\mathbf{x}_{j+1}$, the differential equation should be satisfied at t_{j+1} and the scheme is then determined by

$$p'(t_{j+1}) = \mathbf{f}(t_{j+1}, \mathbf{x}_{j+1}). \tag{4.1}$$

These schemes are called backward differentiation formulas (BDF) methods. Now, the interpolation polynomial can be determined by the Lagrange basis of $\mathbb{P}_k$. This gives us

$$p(t) = \sum_{\ell=0}^{k} \mathbf{x}_{j+1-\ell} L_\ell(t).$$

Calculating the derivative and using (4.1) we end up with

$$\sum_{\ell=0}^{k} \mathbf{x}_{j+1-\ell} L'_\ell(t_{j+1}) = \mathbf{f}(t_{j+1}, \mathbf{x}_{j+1}).$$

	α_0	α_1	α_2	α_3	α_4
$k=1$	1	-1			
$k=2$	$3/2$	-2	$1/2$		
$k=3$	$11/6$	-3	$3/2$	$-1/3$	
$k=4$	$25/12$	-4	3	$-4/3$	$1/4$

Table 1: Parameters for the BDF methods up to $k=4$.

In order to simplify the situation, we restrict to an equidistant time grid with $t_j = t_0 + j\tau$ as in the previous chapters. Hence, the Lagrange polynomials can be transformed to

$$\tilde{L}_\ell(i) = \prod_{h=0, h\neq\ell}^{k} \frac{i+h-1}{h-\ell},$$

where we have $t = t_j + i\tau$. These polynomials do not depend on j. Finally, the k-step method is given by

$$\alpha_0 \mathbf{x}_{j+1} + \alpha_1 \mathbf{x}_j + \ldots + \alpha_k \mathbf{x}_{j-k+1} = \tau \mathbf{f}(t_{j+1}, \mathbf{x}_{j+1})$$

with

$$\alpha_\ell = \tilde{L}'_\ell(1) \text{ for } \ell = 0, \ldots, k,$$

since we have $L'_\ell(t_{j+1}) = \tilde{L}'_\ell(1)/\tau$.

The parameters for the first four multi-step BDF methods can be seen in table 1.

Remark 4.1. *Alternatively to numerical differentiation, schemes can also be derived through numerical integration using $(t_{j-k+\ell}, \mathbf{x}_{j-k+\ell})$ for $\ell = 1, \ldots, k$. By choosing an integer $J \in \mathbb{N}$ we can solve the initial value problem by*

$$\mathbf{x}(t_{j+1}) = \mathbf{x}_{j-J+1} + \int_{t_{j-J+1}}^{t_{j+1}} \mathbf{f}(s, \mathbf{x}(s)) ds.$$

Similar to before we can use an interpolation polynomial with Lagrange polynomials but based on the points

$$(t_{j-k+\ell}, \mathbf{f}(t_{j-k+\ell}, \mathbf{x}_{j-k+\ell})) \quad \textit{for} \quad \ell = 1, \ldots, k.$$

This leads to schemes of the form

$$\mathbf{x}_{j+1} = \mathbf{x}_{j-J+1} + \tau \sum_{\ell=1}^{k} \beta_\ell \mathbf{f}(t_{j-\ell+1}, \mathbf{x}_{j-\ell+1}).$$

These schemes are explicit and one very famous class of these methods are called Adams-Bashforth methods where we set $J = 1$. The parameters can be found in table 2. It can be seen that for $k = 1$ we obtain the explicit Euler scheme.

	β_1	β_2	β_3	β_4
$k=1$	1			
$k=2$	3/2	$-1/2$		
$k=3$	23/12	$-16/15$	5/12	
$k=4$	55/24	$-59/24$	37/24	$-9/24$

Table 2: Parameters for the Adams-Bashforth methods up to $k=4$.

	$\tilde{\beta}_0$	$\tilde{\beta}_1$	$\tilde{\beta}_2$	$\tilde{\beta}_3$	$\tilde{\beta}_4$
$k=1$	1/2	1/2			
$k=2$	5/12	8/12	$-1/12$		
$k=3$	9/24	19/24	$-5/24$	1/24	
$k=4$	251/720	646/720	$-264/720$	106/720	$-19/720$

Table 3: Parameters for the Adams-Moulton methods up to $k=4$.

We can obtain implicit schemes with this approach by using the point $(t_{j+1}, \mathbf{f}(t_{j+1}, \mathbf{x}_{j+1}))$ *for the interpolation polynomial. This leads to*

$$\mathbf{x}_{j+1} = \mathbf{x}_{j-J+1} + \tau \sum_{\ell=0}^{k} \tilde{\beta}_\ell \mathbf{f}(t_{j-\ell+1}, \mathbf{x}_{j-\ell+1}).$$

Choosing $J=1$ *represents the Adams-Moulton methods, see also table 3. For* $k=1$ *we have the implicit trapezoidal rule.*

4.2 Linear Multi-Step Methods

In general, multi-step methods are of the form

$$\mathbf{x}_{j+k} = \mathbf{\Psi}(t_j, \mathbf{x}_j, \ldots, \mathbf{x}_{j+k}, \tau).$$

Nevertheless, only linear multi-step methods are commonly used in practice. These are defined as following:

Definition 4.2. *For an equidistant grid with grid size* τ *we call*

$$\sum_{\ell=0}^{k} \alpha_\ell \mathbf{x}_{j+\ell} = \tau \sum_{\ell=0}^{k} \beta_\ell \mathbf{f}(t_{j+\ell}, \mathbf{x}_{j+\ell}) \tag{4.2}$$

a linear k*-step method with coefficient* $\alpha_0, \ldots, \alpha_k, \beta_0, \ldots, \beta_k \in \mathbb{R}$. *Furthermore,* $\alpha_k \neq 0$ *and* $\alpha_0 \beta_0 \neq 0$ *hold.*

In the case $\beta_k = 0$ the method is explicit, otherwise it is implicit.

Remark 4.3. *Note that with the above definition the aim is to compute* $\mathbf{x}_{j+k}$*, so in comparison to the previous section the indexes are shifted. In general, we have*

- *BDF:* $\beta_0 = \ldots,\ \beta_{k-1} = 0,\ \beta_k = 1$ *and in particular, e.g., for* $k = 1$*:* $\alpha_0 = -1,\ \alpha_1 = 1$ *(compare table 1),*
- *Adams-Bashforth:* $\alpha_0 = \ldots = \alpha_{k-2} = 0,\ \alpha_{k-1} = -1,\ \alpha_k = 1,\ \beta_k = 0$*, and in particular, e.g., for* $k = 1$*:* $\beta_0 = 1,\ \beta_1 = 0$ *(compare table 2),*
- *Adams-Moulton:* $\alpha_0 = \ldots = \alpha_{k-2} = 0,\ \alpha_{k-1} = -1,\ \alpha_k = 1,\ \beta_k \neq 0$*.*

Remark 4.4. *In contrast to one-step methods we need more information about the initial value problem for* k*-step methods. In particular, we need* $k-1$ *additional starting values. As in practice, the analytic starting values are most of the time not available, one can use one-step methods (e.g. Runge-Kutta methods) in order to approximate them.*

4.2.1 Consistency and Convergence

Now we want to analyze the consistency and stability of a k-step method. For simplicity, we assume that the exact starting values $\mathbf{x}_0, \ldots, \mathbf{x}_{k-1}$ are known.

Remark 4.5. *Alternatively, one can investigate starting values which fulfill*

$$\lim_{\tau \to 0} \mathbf{x}_\Delta(t_0 + j\tau) = x_0 \quad \textit{for} \quad j = 0, \ldots, k-1.$$

The consistency error is defined as following:

Definition 4.6. *The consistency error of a linear mulit-step method* (4.2) *is defined by*

$$\epsilon(t, \mathbf{x}, \tau) = \sum_{\ell=0}^{k} \alpha_\ell \mathbf{x}(t_{j+\ell}) - \tau \sum_{\ell=0}^{k} \beta_\ell \mathbf{f}(t_{j+\ell}, \mathbf{x}(t_{j+\ell})),$$

where $\mathbf{x}$ *is the exact solution of the IVP.*
A linear multi-step method is of consistency order p *if*

$$\epsilon(t, \mathbf{x}, \tau) = \mathcal{O}(\tau^{p+1}).$$

Example 4.7. *Let us consider the explicit Euler scheme. We have already seen that this can also be viewed as an Adams-Bashforth method with* $k = 1$*. The consistency error is then given by*

$$\begin{aligned} \epsilon(t, \mathbf{x}, \tau) =& \mathbf{x}(t+\tau) - \mathbf{x}(t) - \tau f(t, \mathbf{x}(t)) \\ =& \mathbf{x}(t+\tau) - \mathbf{x}(t) - \tau \mathbf{x}'(t). \end{aligned}$$

Using the discrete evolution of the explicit Euler scheme, i.e., $\mathbf{\Psi}^{t+\tau,t}\mathbf{x}_\Delta(t) = \mathbf{x}(t) + \tau f(t, \mathbf{x}(t))$*, we can see that we have obtained the classical definition of the consistency error from the previous chapters.*

Using Taylor expansions we can now derive the conditions for a consistency order p.

Lemma 4.8. *A linear k-step method (4.2) is of consistency order p if and only if the following conditions on the coefficients are fulfilled*

$$\sum_{\ell=0}^{k} \alpha_\ell = 0, \quad \sum_{\ell=0}^{k} \alpha_\ell \ell^j - j\beta_\ell \ell^{j-1} = 0 \quad \text{for } j = 1, \dots, p. \tag{4.3}$$

Remark 4.9. *Using the conditions (4.3) it is possible to construct multi-step methods up to order $p = 2k$.*

Example 4.10. *Recall the Adams-Moulton method for $k = 1$, i.e., the trapezoidal rule:*

$$-\mathbf{x}_j + \mathbf{x}_{j+1} = \frac{\tau}{2}\left(f_j + f_{j+1}\right),$$

where we use as an abbreviation $f_j := \mathbf{f}(t_j, \mathbf{x}_j)$. Hence, the coefficients are $\alpha_0 = -1$, $\alpha_1 = 1$, $\beta_0 = \beta_1 = 1/2$. We see immediately that the first condition

$$\alpha_0 + \alpha_1 = 0$$

holds. Then for $j = 1$ the second condition gives us

$$\sum_{\ell=0}^{1} \alpha_\ell \ell - \beta_\ell = 0.$$

This is fulfilled as

$$\sum_{\ell=0}^{1} \alpha_\ell \ell - \beta_\ell = -\beta_0 + \alpha_1 - \beta_1 = -1/2 + 1 - 1/2 = 0.$$

Hence, the consistency order is at least 1. Now, we check the second condition for $j = 2$ and obtain

$$\sum_{\ell=0}^{1} \alpha_\ell^2 \ell - 2\beta_\ell \ell = \sum_{\ell=1}^{1} \alpha_\ell^2 \ell - 2\beta_\ell \ell = 1 \cdot 1^2 - 2 \cdot 1/2 = 0.$$

So we observe the consistency order $p \geq 2$. Finally checking that for $j = 3$

$$\sum_{\ell=1}^{1} \alpha_\ell^3 \ell - 3\beta_\ell \ell^2 = 1 \cdot 1 - 3 \cdot 1/2 \neq 0,$$

we obtain the convergence order of $p = 2$.

Remark 4.11. *As a small exercise you can check the consistency order of the Adams-Moulton method for $k = 2$ given in table 3, i.e.,*

$$-\mathbf{x}_{j+1} + \mathbf{x}_{j+2} = \tau\left(\frac{5}{12} f_{j+2} + \frac{8}{12} f_{j+1} - \frac{1}{12} f_j\right).$$

In general, one can show that Adams-Moulton methods are of consistency order $p = k + 1$.

In contrast to one-step methods the consistency of a k-step method with $k > 1$ does not imply convergence anymore. In particular, small perturbations, e.g., through rounding, can lead to a significant large global error. In order to avoid this, we have to additionally ensure the stability of multi-step methods. Here, it is sufficient to discuss the stability for the case $f \equiv 0$. For the stability we need the following definition:

Definition 4.12. *The characteristic polynomial of a linear multi-step method is given by*

$$p(\lambda) = \sum_{\ell=0}^{k} \alpha_\ell \lambda^\ell.$$

Now, we are able to define the stability.

Definition 4.13. *A linear multi-step method is stable, if the zeros $\lambda_1, \ldots, \lambda_k \in \mathbb{C}$ of the corresponding characteristic polynomial satisfy the following conditions:*

i) $|\lambda_j| \leq 1$ *if* λ_j *is a zero,*

ii) $|\lambda_j| < 1$ *if* λ_j *is a multiple zero.*

The latter condition is also known as the Dahlquist root condition.

Remark 4.14. *The characteristic polynomial of a one-step method is given by $p(\lambda) = 1 - \lambda$. Therefore, the zero is 1 and the Dahlquist root condition is satisfied. Hence, one-step methods are stable.*

The condition of stability comes with a restriction of the consistency order as the following theorem by Dahlquist shows:

Theorem 4.15. *Each linear and stable k-step method with consistency order p satisfies:*

$$\begin{aligned} p &\leq k + 2 \quad \text{if } k \text{ even}, \\ p &\leq k + 1 \quad \text{if } k \text{ odd}. \end{aligned}$$

These upper bounds are sharp.

The latter theorem shows that we are not able to construct a stable and consistent method above the order $k+2$ or $k+1$, respectively. Nevertheless, both, stability and consistency, are necessary to ensure the convergence of a linear multi-step method.

Theorem 4.16. *A linear multi-step method is convergent of order p if and only if it is consistent of order p and stable.*

Now, we want to discuss the stability and convergence of the numerical methods introduced at the beginning of this chapter.

We start with the Adams-methods (meaning the explicit Adams-Bashforth and the implicit Adams-Moulton methods). The coefficients are given by $\alpha_0 = \ldots = \alpha_{k-2} = 0$ and $\alpha_{k-1} = -1$, $\alpha_k = 1$. Therefore, the characteristic polynomial of an Adams-method is

$$p(\lambda) = (\lambda - 1)\lambda^{k-1}.$$

Hence, we have the multiple zero 0 ($k-1$ times) and the zero 1. Both fulfill the Dahlquist root condition and therefore Adams-methods are always stable. In general, we can get the following convergence result:

Theorem 4.17. *The k-step Adams-Bashforth methods are stable and consistent (hence convergent) with order k for every $k \in \mathbb{N}$, the Adams-Moulton methods are stable and consistent with order $k+1$.*

Therefore, Adams-Moulton methods have (at least for k being odd) the optimal convergence order.

In contrast to the Adams-methods, BDF methods are less stable.

Theorem 4.18. *The k-step BDF method is consistent with order k. These methods are only stable (hence convergent), if $k \leq 6$.*

We finish this section with some important remarks.

Remark 4.19. *As already mentioned the starting values $\mathbf{x}_0, \ldots, \mathbf{x}_{k-1}$ can be obtained by one-step Runge-Kutta method. Here, it is important to use at least the same order of convergence as the underlying multi-step method in order to avoid a loss in the convergence rates.*

Remark 4.20. *Similar to one-step methods, explicit multi-step methods should not be used for stiff problems.*