2 Numerical Methods for Initial Value Problems

Contents

2.1 Some Simple Methods
2.2 One-Step Methods – Definition and Properties
2.3 Runge-Kutta-Methods
2.4 Linear Multi-Step Methods
2.5 Convergence of One-Step Methods
2.6 Zero-Stability of Linear Multistep Methods
2.7 Stepsize Control
2.1 Some Simple Methods

Consider an IVP

\[ y' = f(t, y), \quad y(t_0) = y_0. \]  

(IVP)

Under the assumptions of Theorem 1.1, (IVP) possesses a unique solution \( y = y(t) \), say, on the interval \( I \).

We will approximate \( y(t) \) for \( t \in [t_0, t_{\text{end}}] \subset I \) using the Euler method\(^a\) (sometimes called the explicit Euler method), which can be regarded as a prototype for all numerical schemes for solving IVPs.

As all methods we shall encounter in this course, Euler’s method is based on discretization, i.e., approximating the solution \( y \) on only a discrete subset \( \{t_n : n = 0, 1, \ldots, N\} \) of the interval \( [t_0, t_{\text{end}}] \).

\(^a\)Leonhard Euler (1707–1783)
We fix $N \in \mathbb{N}$, set $h := (t_{\text{end}} - t_0)/N$ and define $t_n := t_0 + nh, n = 0, 1, \ldots, N$, i.e. $t_0 < t_1 < \cdots < t_{N-1} < t_N = t_{\text{end}}$. The number $h > 0$ is called the stepsize, which we choose to be constant for convenience only.

One step of Euler’s method requires an approximation $y_n$ to $y(t_n), n = 0, 1, \ldots, N - 1$. Noting that, under the assumption $y \in C^2[t_0, t_{\text{end}}],$

$$y(t_{n+1}) = y(t_n + h) = y(t_n) + hy'(t_n) + \frac{1}{2}h^2y''(\xi) \approx y(t_n) + hy'(t_n) = y(t_n) + hf(t_n, y(t_n)) \approx y_n + hf(t_n, y_n),$$

we obtain the

(Explicit) Euler scheme:

\begin{align*}
y_0 & \text{ given,} \\
y_{n+1} & := y_n + hf(t_n, y_n) \quad n = 0, 1, \ldots, N - 1.
\end{align*}
A common way to visualize a first order ODE \( y' = f(t, y) \) is provided by its associated direction field, obtained by drawing an arrow with slope \( y' = f(t, y) \) at every point in the \((t, y)\)-plane.

The graph of the solution to the IVP

\[
y' = f(t, y), \quad y(t_0) = y_0,
\]

must pass through the point \((t_0, y_0)\) and its tangents at every point \((t, y(t))\) must be parallel to the direction field.

The following figure illustrates the Euler method approximating the solution of the logistic equation \( y' = y(1 - y) \) with IC \( y(0) = 1/10 \) using the step size \( h = 1 \). Rather than following its exact trajectory (which is, of course, impossible), the Euler scheme may be viewed at producing a piecewise linear approximation. At the starting point \( t_0 \) the Euler approximation uses the correct slope \( f(t_0, y_0) = 9/100 \). However, from the next point \( t_1 = h = 1 \) onwards the slope is “wrong”, and the approximation may deviate further and further from the exact solution.
Euler–Verfahren

$y' = y(1-y), \ y(0) = 1/10$
Richtungsfeld von $y' = x(y-2)$

und Lösungen mit Anfangswerten $y(0) = -1:0.5:1$
Does Euler’s method converge, i.e., do the approximations converge to the exact solution $y(t)$ as $h \to 0$?

Note: each value of the step size $h$ is associated with a (finite) collection of approximations

$$y_n = y_n(h), \quad n = 0, 1, \ldots, N(h) := \lfloor (t_{\text{end}} - t_0)/h \rfloor.$$

The method is said to converge (in $[t_0, t_{\text{end}}]$) if

$$\lim_{h \to 0^+} \max_{0 \leq n \leq N(h)} \| y_n(h) - y(t_n) \| = 0. \quad \text{(Conv)}$$

**Theorem 2.1 (Convergence of Euler’s method)** Under the assumptions of Theorem 1.1 the Euler method is convergent. More precisely,

$$\max_{0 \leq n \leq N(h)} \| y_n(h) - y(t_n) \| = O(h) \quad \text{as } h \to 0.$$
### Modified Euler method:

\[ y_0 \text{ given,} \]
\[ y_{n+1} := y_n + hf \left( t_n + \frac{1}{2}h, y_n + \frac{1}{2}hf(t_n, y_n) \right) \quad n = 0, 1, \ldots, N - 1. \]

### Improved Euler method:

\[ y_0 \text{ gegeben,} \]
\[ y_{n+1} := y_n + \frac{1}{2}h[f(t_n, y_n) + f(t_n + h, y_n + hf(t_n, y_n))] \quad n = 0, 1, \ldots, N - 1. \]

The construction of these variants of Euler’s method is also easily interpreted using the direction field. Both are convergent schemes – in fact (by a trivial modification of the proof of Theorem 2.1) one can show that for both methods

\[
\max_{0 \leq n \leq N} \|y_n(h) - y(t_n)\| = O(h^2), \quad \text{as } h \to 0.
\]
2.1 Some Simple Methods
Implicit Euler method:

\[ y_0 \text{ given}, \]
\[ y_{n+1} := y_n + h f(t_n + h, y_{n+1}) \quad n = 0, 1, \ldots, N - 1. \]

In contrast with the methods introduced so far, this is an implicit method, which means that determining \( y_{n+1} \) from \( y_n \) requires the solution of a system of algebraic equations, which, unless \( f \) is linear in \( y \), is generally nonlinear.

As in Theorem 2.1 one can show that for the implicit Euler method

\[
\max_{0 \leq n \leq N(h)} \| y_n(h) - y(t_n) \| = O(h) \quad \text{as } h \to 0.
\]
The schemes we have considered up to now involved two approximations of the solution at consecutive times. Such schemes are called one-step methods.

**Multistep methods** are based on difference formulas involving more than two solution values. Examples are

\[
\frac{y(t + h) - y(t - h)}{2h} = y'(t) + \frac{h^2}{6}y'''(t) + O(h^4),
\]

\[
\frac{3y(t) - 4y(t - h) + y(t - 2h)}{2h} = y'(t) - \frac{h^2}{3}y'''(t) + O(h^3).
\]

When these formulas are evaluated at \( t = t_{n+1} \), we obtain . . .
...the explicit

**Midpoint-rule:** (leapfrog method)

\[ y_0, y_1 \text{ given,} \]
\[ y_{n+1} = y_{n-1} + 2h f(t_n, y_n), \quad n = 1, 2, \ldots, N - 1. \]  

(2.1)

as well as an implicit method belonging to the so-called **BDF-family** (Backward Differentiation Methods)

**A BDF method:**

\[ y_0, y_1 \text{ given,} \]
\[ 3y_{n+1} - 4y_n + y_{n-1} = 2hf(t_n + h, y_{n+1}), \quad n = 1, 2, \ldots, N - 1. \]

We observe that these methods can only proceed once, in addition to the initial value \( y_0 \), an approximation \( y_1 \) is available. All multistep methods require such a **startup calculation**.
2.2 One-Step Methods – Definition and Properties

A one-step method (OSM) has the general form

\[ y_{n+1} = y_n + h \Phi_f(y_{n+1}, y_n, t_n; h). \]  

(OSM)

We will consider exclusively methods whose increment function \( \Phi_f \) possesses the following properties:

\[ \Phi_f \equiv 0(y_{n+1}, y_n, t_n; h) \equiv 0 \]  

\((V_1)\)

and

\[ \| \Phi_f(y_{n+1}, y_n, t_n; h) - \Phi_f(y^*_n, y^*_n, t_n; h) \| \leq M \sum_{j=0}^{1} \| y_{n+j} - y^*_{n+j} \|. \]  

\((V_2)\)

For “reasonable” OSM property \((V_2)\) follows from the Lipschitz continuity of \( f \) (cf. Theorem 1.1), which we always assume to hold.
Examples of more complicated OSMs:

\[ y_{n+1} - y_n = \frac{1}{4} h(k_1 + 3k_3), \text{ where} \]

\[ k_1 = f(t_n, y_n), \]
\[ k_2 = f(t_n + \frac{1}{3} h, y_n + \frac{1}{3} h k_1), \]
\[ k_3 = f(t_n + \frac{2}{3} h, y_n + \frac{2}{3} h k_2), \]

(Example 1)

an explicit OSM belonging to the class of Runge-Kutta methods \(^a\) \(^b\) (cf. §2.3) and the implicit Runge-Kutta method

\[ y_{n+1} - y_n = \frac{1}{2} h(k_1 + k_2), \text{ where} \]

\[ k_1 = f(t_n, y_n), \]
\[ k_2 = f(t_n + h, y_n + \frac{1}{2} h k_1 + \frac{1}{2} h k_2). \]

(Example 2)

\(^a\) CARLE DAVID TOLMÉ RUNGE (1856–1927)

\(^b\) MARTIN WILHELM KUTTA (1867–1944)
We say the method (OSM) is convergent if

\[ \lim_{h \to 0} y_n = \lim_{h \to 0} y_n(h) = y(t) \]

holds — and does so

- for all IVPs which satisfy the assumptions of Theorem 1.1 where \( y(t) \) denotes the solution of such an IVP,
- uniformly for all \( t \in [t_0, t_{\text{end}}] \),
- for all solutions \( \{y_n(h)\} = \{y_n\} \) of (OSM) with initial values \( y_0(h) \) satisfying \( \lim_{h \to 0} y_0(h) = y_0 \).

**Equivalent:** The global discretisation error

\[ e_n = e_n(h) := y(t_n) - y_n(h) \]

converges to 0 uniformly (as \( h \to 0 \)):

\[ \lim_{h \to 0} \max_{0 \leq n \leq N} \|e_n(h)\| = \lim_{h \to 0} \max_{0 \leq n \leq N} \|y(t_n) - y_n(h)\| = 0. \]
We define the **local discretion error** (local truncation error) $T_n = T_n(h)$ of a method at step $n$ to be the difference of the left and right-hand sides of the expression defining the method (scaled in such a way that the differential equation is approximated as $h \to 0$) when the exact solution is substituted for the approximation.

A method is said to be **consistent of order** $p$ if

$$T_n(h) = O(h^p) \quad \text{as } h \to 0.$$  

**Example:** Midpoint rule

$$T_n = \frac{y(t_{n+1}) - y(t_{n-1})}{2h} - f(t_n, y(t_n))$$

$$= \left[ y'(t_n) + \frac{h^2}{6} y'''(t_n) + O(h^4) \right] - y'(t_n) = \frac{h^2}{6} y'''(t_n) + O(h^4),$$

i.e., we have consistency of order $p = 2.$
**Warning:** In the literature one often finds a slightly different definition of the local discretization error based on the method’s defining equation in the form (OSM). This increases the order of consistency by one power of \( h \).

**Motivation:** Description of the error incurred in a single step independent of that accumulated from previous steps using the localization assumption \( y_n = y(t_n) \). In this case one step of the scheme (OSM) compared with the exact solution compares as

\[
\hat{y}_{n+1}(h) = y(t_n) + h\Phi_f(\hat{y}_{n+1}, y(t_n), t_n; h),
\]
\[
y(t_{n+1}) = y(t_n) + h\Phi_f(y(t_{n+1}), y(t_n), t_n; h) + hT_{n+1}(h).
\]

Denoting the difference \( S_{n+1}(h) := y(t_{n+1}) - \hat{y}_{n+1} \) as step error, we obtain using \((V_2)\),

\[
\| S_{n+1} \| \leq h\| \Phi_f(y(t_{n+1}), y(t_n), t_n; h) - \Phi_f(\hat{y}_{n+1}, y(t_n), t_n; h) \| + h\| T_{n+1} \|
\]
\[
\leq hM\| S_{n+1} \| + h\| T_{n+1} \|,
\]

implying \((1 - hM)\| S_{n+1} \| \leq hT_{n+1} \) so that \( S_n \) behaves like \( hT_n \) as \( h \to 0 \).
Relevance of $S_n$: The integration of (IVP) from $t_0$ to $t_{\text{end}}$ requires $(t_{\text{end}} - t_0)/h$ steps. If step error $S_n$ is committed at step $n$, this results, under the simplifying assumption that individual error have no effect on each other, in an accumulated (global) error of

$$\frac{t_{\text{end}} - t_0}{h} S(h) \sim T(h).$$

As we shall see shortly, the crucial property for the validity of this simplifying assumption is the stability of the method.
2.3 Runge-Kutta Methods

2.3.1 Derivation via Quadrature

Fundamental theorem of calculus:

\[ y(t + h) = y(t) + [y(t + h) - y(t)] = y(t) + \int_t^{t+h} y'(s) \, ds \]

Change of variables: \( s = t + \tau h, \ 0 \leq \tau \leq 1 \)

\[ = y(t) + h \int_0^1 y'(t + \tau h) \, d\tau. \]

Approximate integral with quadrature formula:

\[ \int_0^1 g(\tau) \, d\tau \approx \sum_{j=1}^{m} \beta_j g(\gamma_j). \]  

To integrate at least \( g \equiv 1 \) exactly, we require \( \sum_{j=1}^{m} \beta_j = 1. \)
We obtain

\[ y(t + h) \approx y(t) + h \sum_{j=1}^{m} \beta_j y'(t + \gamma_j h) \]

(RK-1)

\[ = y(t) + h \sum_{j=1}^{m} \beta_j f(t + \gamma_j h, y(t + \gamma_j h)) \]

Problem now: don’t know \( y(t + \gamma_j h) = y(t) + h \int_{0}^{\gamma_j} y'(t + \tau h) d\tau. \)

Again approximate by quadrature, but using same nodes \( \{\gamma_j\}_{j=1}^{m} \) as in (*) to avoid introducing new unknowns \( y(t + \text{node} \cdot h) \):

\[ \int_{0}^{\gamma_j} g(\tau) d\tau \approx \sum_{\ell=1}^{m} \alpha_{j,\ell} g(\gamma_\ell), \quad j = 1, \ldots, m. \] (**)

To integrate at least \( g \equiv 1 \) exactly, require \( \sum_{\ell=1}^{m} \alpha_{j,\ell} = \gamma_j, \quad j = 1, \ldots, m. \)
This yields

\[ y(t + \gamma_j h) \approx y(t) + h \sum_{\ell=1}^{m} \alpha_{j,\ell} y'(t + \gamma_{\ell} h) \]

\[ = y(t) + h \sum_{\ell=1}^{m} \alpha_{j,\ell} f(t + \gamma_{\ell} h, y(t + \gamma_{\ell} h)) \]  \hspace{1cm} \text{(RK-2)}

Introduce \( \tilde{k}_j := f(t + \gamma_j h, y(t + \gamma_j h)), \) \hspace{1cm} j = 1, \ldots, m.

\text{(RK-2):} \quad \tilde{k}_j \approx f \left( t + \gamma_j h, y(t) + h \sum_{\ell=1}^{m} \alpha_{j,\ell} \tilde{k}_\ell \right), \quad j = 1, \ldots, m.

\text{(RK-1):} \quad y(t + h) \approx y(t) + h \sum_{j=1}^{m} \beta_j \tilde{k}_j.
**m-stage Runge-Kutta method** (RKM):

\[ y_{n+1} = y_n + h \sum_{j=1}^{m} \beta_j k_j \]

where

\[ k_j = f \left( t_n + \gamma_j h, y_n + h \sum_{\ell=1}^{m} \alpha_{j,\ell} k_\ell \right), \quad j = 1, \ldots, m. \]

Shorthand notation for RKMs:

**Butcher Tableau**

\[
\begin{array}{ccccc}
\gamma_1 & \alpha_{1,1} & \cdots & \alpha_{1,m} \\
\vdots & \vdots & \ddots & \vdots \\
\gamma_m & \alpha_{m,1} & \cdots & \alpha_{m,m} \\
\hline
& \beta_1 & \cdots & \beta_m
\end{array}
\]

\[ ^a \text{JOHN CHARLES BUTCHER} (*1933) \]
**Examples.** The Butcher Tableau

\[
\begin{array}{ccc}
0 & 0 & 0 \\
1 & 1 & 0 \\
\hline
1/2 & 1/2 & \\
\end{array}
\]

represents a two-stage explicit RKM, the **improved Euler method**:

\[
y_{n+1} = y_n + \frac{h}{2} \left[ f(t_n, y_n) + f(t_n + h, y_n + hf(t_n, y_n)) \right] \\
= y_n + \frac{h}{2} (k_1 + k_2), \\
k_1 = f(t_n, y_n), \\
k_2 = f(t_n + h, y_n + h k_1)
\]

**Note:** a RKM is **explicit** when

\[
\alpha_{j,\ell} = 0 \quad \forall j \leq \ell,
\]

i.e., if the coefficients \( \alpha_{j,\ell} \) form a strictly lower triangular matrix.
represents a two-stage implicit RKM:

\[
\begin{align*}
k_1 &= f \left( t_n, y_n + h \left( \frac{1}{4} k_1 - \frac{1}{4} k_2 \right) \right), \\
k_2 &= f \left( t_n + \frac{2}{3} h, y_n + h \left( \frac{1}{4} k_1 + \frac{5}{12} k_2 \right) \right),
\end{align*}
\]

(“two” — generally nonlinear — equations for \(k_1\) and \(k_2\))

\[
y_{n+1} = y_n + \frac{h}{4} (k_1 + 3k_2).
\]

(Example 2 in §2.2 is a further example of an implicit two-stage RKM.)
<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/2</td>
<td>1/2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>−1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>1/6</td>
<td>4/6</td>
<td>1/6</td>
<td></td>
</tr>
</tbody>
</table>

represents an explicit three-stage RKM

### Kutta’s third-order method:

\[
\begin{align*}
k_1 &= f(t_n, y_n), \\
k_2 &= f \left( t_n + \frac{1}{2} h, y_n + \frac{1}{2} h k_1 \right), \\
k_3 &= f \left( t_n + h, y_n + h(-k_1 + 2k_2) \right), \\
y_{n+1} &= y_n + \frac{1}{6} h(k_1 + 4k_2 + k_3).
\end{align*}
\]
Heun’s third-order method:

\[
\begin{align*}
k_1 &= f(t_n, y_n), \\
k_2 &= f \left(t_n + \frac{1}{3} h, y_n + \frac{1}{3} h k_1\right), \\
k_3 &= f \left(t_n + \frac{2}{3} h, y_n + \frac{2}{3} h k_2\right), \\
y_{n+1} &= y_n + \frac{h}{4} (k_1 + 3k_3)
\end{align*}
\]

(cf. Example 1 in §2.2).
The table represents an explicit four-stage RKM (Runge-Kutta Method):

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/2</td>
<td>1/2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1/2</td>
<td>0</td>
<td>1/2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

The values in the table are:

- 1/6
- 2/6
- 2/6
- 1/6

**Classical Runge-Kutta method:**

\[
\begin{align*}
k_1 &= f(t_n, y_n), \\
k_2 &= f(t_n + \frac{1}{2} h, y_n + \frac{1}{2} h k_1), \\
k_3 &= f(t_n + \frac{1}{2} h, y_n + \frac{1}{2} h k_2), \\
k_4 &= f(t_n + h, y_n + h k_3), \\
y_{n+1} &= y_n + \frac{1}{6} h (k_1 + 2k_2 + 2k_3 + k_4).
\end{align*}
\]
An (equivalent) alternative way of representing (RKM) is

\[ y_{n+1} = y_n + h \sum_{j=1}^{m} \beta_j f(t_n + \gamma_j h, \tilde{y}_j) \]

where \( \tilde{y}_j = y_n + h \sum_{\ell=1}^{m} \alpha_{j,\ell} f(t_n + \gamma_{\ell} h, \tilde{y}_{\ell}), \quad j = 1, \ldots, m \)

(simply set \( k_j = f(t_n + \gamma_j h, \tilde{y}_j) \)).

**Interpretation:**

\( \tilde{y}_j \): approximate solution \( y \) at time \( t_n + \gamma_j h \)

\( k_j \): approximate slopes \( y' \) at time \( t_n + \gamma_j h \)
2.3.2 Consistency of Runge-Kutta Methods

All RKM can be written

\[ y_{n+1} = y_n + h \Phi_f(y_{n+1}, y_n, t_n; h) \]

where

\[ \Phi_f(y_{n+1}, y_n, t_n; h) = \sum_{j=1}^{m} \beta_j k_j. \]

A RKM is consistent (and therefore convergent) if, and only if,

\[ \sum_{j=1}^{m} \beta_j = 1. \]

Determining the order of consistency of RKM (or constructing \(m\)-stage RKM with maximal order of consistency) often leads to rather involved calculations.
As a tractable example, we shall analyze all explicit three-stage RKMs. Any such method has a Butcher Tableau of the form

\[
\begin{array}{cccc}
0 & 0 & 0 & 0 \\
\gamma_2 & \gamma_2 & 0 & 0 \\
\gamma_3 & \gamma_3 - \alpha_{3,2} & \alpha_{3,2} & 0 \\
\hline
\beta_1 & \beta_2 & \beta_3
\end{array}
\]

We expand the local truncation error

\[
T_{n+1}(h) = \frac{y(t_{n+1}) - y(t_n)}{h} - \Phi_f (y(t_n), t_n; h) = \frac{y(t_{n+1}) - y(t_n)}{h} - \sum_{j=1}^{3} \beta_j k_j
\]

in powers of $h$ (assuming $y$ and $f$ sufficiently differentiable).
We consider scalar IVPs and, introducing the abbreviations

\[ F := f_t + f_y f \quad \text{and} \quad G := f_{tt} + 2f_{ty}f + f_{yy}f^2, \]

—all derivatives of \( f \) evaluated at \((t_n, y(t_n))\)—employ the chain rule to obtain

\[ \frac{y(t_{n+1}) - y(t_n)}{h} = f + \frac{1}{2}Fh + \frac{1}{6}(G + f_y F)h^2 + O(h^3). \]

On the other hand, we also have

\[ k_1 = f(t_n, y(t_n)) = f, \]

\[ k_2 = f(t_n + h\gamma_2, y(t_n) + h\gamma_2 k_1) = f + h\gamma_2 F + \frac{1}{2}h^2\gamma_2^2 G + O(h^3), \]

\[ k_3 = f(t_n + h\gamma_3, y(t_n) + h(\gamma_3 - \alpha_3,2)k_1 + h\alpha_3,2 k_2) \]

\[ = f + h\gamma_3 F + h^2(\gamma_2 \alpha_3,2 F f_y + \frac{1}{2}\gamma_3^2 G) + O(h^3). \]
This means:

\[ T_{n+1}(h) = \left[ 1 - \sum_{j=1}^{3} \beta_j \right] f + \left[ \frac{1}{2} - \beta_2 \gamma_2 - \beta_3 \gamma_3 \right] Fh \]
\[ + \left[ (\frac{1}{3} - \beta_2 \gamma_2^2 - \beta_3 \gamma_3^2) \frac{1}{2} G + (\frac{1}{6} - \beta_3 \gamma_2 \alpha_{3,2}) F_{f_y} \right] h^2 + O(h^3) \]

**Consequences:**

(1) The Euler method is the only explicit one-stage RKM of order one \((\beta_1 = 1)\).

(2) All explicit two-stage RKM of order 2 are characterized by

\[ \beta_1 + \beta_2 = 1 \quad \text{and} \quad \beta_2 \gamma_2 = \frac{1}{2}. \]

Examples are the modified \((\beta_1 = 0, \beta_2 = 1, \gamma_2 = \frac{1}{2})\) and the improved \((\beta_1 = \beta_2 = \frac{1}{2}, \gamma_2 = 1)\) Euler methods.

There are no explicit two-stage RKMs of order three or higher.
(3) Explicit three-stage RKM of order 3 are characterized by the four equations

\[ \beta_1 + \beta_2 + \beta_3 = 1, \quad \beta_2 \gamma_2^2 + \beta_3 \gamma_3^2 = \frac{1}{3}, \]
\[ \beta_2 \gamma_2 + \beta_3 \gamma_3 = \frac{1}{2}, \quad \beta_3 \gamma_2 \alpha_{3,2} = \frac{1}{6}. \]

(One can show that none of these methods have order 4.) Examples include Heun’s method \((\beta_1 = \frac{1}{4}, \beta_2 = 0, \beta_3 = \frac{3}{4}, \gamma_2 = \frac{1}{3}, \gamma_3 = \alpha_{3,2} = \frac{2}{3})\) and Kutta’s method \((\beta_1 = \frac{1}{6}, \beta_2 = \frac{2}{3}, \beta_3 = \frac{1}{6}, \gamma_2 = \frac{1}{2}, \gamma_3 = 1, \alpha_{3,2} = 2)\).

(4) Similar (complicated) calculations reveal the existence of a two-parameter family of explicit four-stage RKMs, none of which have order 5. An example is the classical RKM; further examples are . . .
### 2.3 Runge-Kutta Methods

#### (3/8-rule)

\[
\begin{array}{cccccc}
0 & 0 & 0 & 0 & 0 & 0 \\
1/3 & 1/3 & 0 & 0 & 0 & 0 \\
2/3 & -1/3 & 1 & 0 & 0 & 0 \\
1 & 1 & -1 & 1 & 0 & 0 \\
\hline
\end{array}
\]

#### (Kuntzmann’s method).

\[
\begin{array}{cccccc}
0 & 0 & 0 & 0 & 0 & 0 \\
2/5 & 2/5 & 0 & 0 & 0 & 0 \\
3/5 & -3/20 & 3/4 & 0 & 0 & 0 \\
1 & 19/44 & -15/44 & 40/44 & 0 & 0 \\
\hline
55/360 & 125/360 & 125/360 & 55/360 & & \\
\end{array}
\]
This way of analyzing consistency of explicit RKMs becomes increasingly tedious for higher orders:

- order 3: 4 nonlinear equations for coefficients (see above)
- order 8: 200 nonlinear equations for coefficients.

Butcher Theory\textsuperscript{a} uses graph theory (trees) for systematic book-keeping of partial derivatives of $f$ for computing the order of a given RKM. It does not, however, provide a technique for constructing RKMs of a desired order.

In the following, we derive a sequence of necessary order conditions, obtained from the specific family of IVPs

\[ y' = y + t^{\ell-1}, \quad y(0) = 0, \quad (\ell \in \mathbb{N}). \]

Theorem 2.2 (Necessary order conditions for RKM) In order for the RKM defined by the Butcher tableau

\[
\begin{array}{c|c}
  c & A \\
  \hline \\
  b^\top & \\
\end{array}
\]

to possess consistency order \( p \) there must hold

\[
b^\top A^k C^{\ell-1} e = \frac{(\ell - 1)!}{(\ell + k)!} = \frac{1}{\ell(\ell + 1) \ldots (\ell + k)}
\]

for \( \ell = 1, 2, \ldots, p \) and \( k = 0, 1, \ldots, p - \ell \).

Notation:

\[
b := [\beta_1, \beta_2, \ldots, \beta_m]^\top, \quad A := [\alpha_{j,\nu}]_{1 \leq j, \nu \leq m}, \quad C := \text{diag}(\gamma_1, \gamma_2, \ldots, \gamma_m) \quad \text{and} \quad e := [1, 1, \ldots, 1]^\top \in \mathbb{R}^m.
\]
Special cases of the necessary conditions of Theorem 2.2 are (for $k = 0$)

$$b^\top C^{\ell-1} e = \sum_{j=1}^{m} \beta_j \gamma_j^{\ell-1} = \frac{1}{\ell} \quad \text{for } \ell = 1, 2, \ldots, p$$

as well as (for $\ell = 1$ with $k \leftarrow k + 1$)

$$b^\top A^{k-1} e = \frac{1}{k!} \quad \text{for } k = 1, 2, \ldots, p.$$

**Remark.** An explicit $m$-stage RKM can have at most order $m$, since in this case $A^m = O$ ($A$ is a strictly lower triangular matrix). In fact, for $m \geq 5$ the maximal order $p(m)$ attainable by an explicit $m$-stage RKM is bounded by $p(m) \leq m - 1$.

The exact maximal orders for $1 \leq m \leq 12$ are

<table>
<thead>
<tr>
<th>$m$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p(m)$</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>
2.4 Linear Multistep Methods

Multistep methods are characterized by using solution approximations earlier than $y_n$ in the update formula for advancing the numerical approximation from $y_n$ to $y_{n+1}$. This is in contrast with one-step methods, which use only $y_n$, besides possibly additional intermediate (e.g. stage) quantities.

Of these, the most important class of linear multistep methods (LMM) possesses the general form

$$\sum_{j=0}^{r} \alpha_j y_{n+j} = h \sum_{j=0}^{r} \beta_j f(t_{n+j}, y_{n+j}), \quad n = 0, 1, 2, \ldots \quad \text{(LMM)}$$

defined by the parameters $\{\alpha_j\}_{j=0}^{r}$ and $\{\beta_j\}_{j=0}^{r}$. Since these are unique only up to a (nonzero) scaling factor, they are typically normalized by the condition $\alpha_r = 1$.

The method (LMM) is explicit if $\beta_r = 0$, otherwise implicit.

We first study various subfamilies of LMM.
2.4.1 Adams-Methods

Adams-Methods have the specific form

\[ y_{n+r} - y_{n+r-1} = h \sum_{j=0}^{r} \beta_j f(t_{n+j}, y_{n+j}), \]  

(2.2)

i.e., they are LMMs characterized by

\[ \alpha_r = 1, \quad \alpha_{r-1} = -1, \quad \text{in addition to} \quad \alpha_0 = \alpha_1 = \cdots = \alpha_{r-2} = 0. \]

The coefficients \( \beta_j \) result from maximizing the order of consistency for a given value of \( r \), which is \( r \) in the explicit case and \( r + 1 \) in the implicit case.

The former class of methods are known as Adams-Bashforth methods, the latter Adams-Moulton methods.
Alternative derivation of Adams methods:

From the differential equation, we have

\[ y(t_{n+r}) = y(t_{n+r-1}) + \int_{t_{n+r-1}}^{t_{n+r}} y'(<\tau>) \, d\tau = y(t_{n+r-1}) + \int_{t_{n+r-1}}^{t_{n+r}} f(\tau, y(\tau)) \, d\tau. \]

Choosing \( \{\beta_j\}_{j=0}^r \) in such a way that

\[ \int_{t_{n+r-1}}^{t_{n+r}} g(\tau) \, d\tau \approx h \sum_{j=0}^{r} \beta_j g(t_{n+j}) \]

is an interpolatory quadrature formula\(^a\) yields the coefficients of the corresponding (explicit or implicit) Adams method.

\(^a\)i.e., replace \( g \) by the unique interpolation polynomial passing through the points \( \{(t_n, g(t_n)), \ldots, (t_{n+r-1}, g(t_{n+r-1}))\} \) of degree \( r - 1 \) (explicit case) or \( \{(t_n, g(t_n)), \ldots, (t_{n+r}, g(t_{n+r}))\} \) of degree \( r \) (implicit case)
For \( r = 1 \) to 4 the explicit Adams-Bashforth methods are as follows:

\[
\begin{align*}
  r = 1 : & \quad y_{n+1} = y_n + hf(t_n, y_n), \\
  r = 2 : & \quad y_{n+2} = y_{n+1} + \frac{h}{2} (-f(t_n, y_n) + 3f(t_{n+1}, y_{n+1})), \\
  r = 3 : & \quad y_{n+3} = y_{n+2} + \frac{h}{12} \left( 5f(t_n, y_n) - 16f(t_{n+1}, y_{n+1}) + 23f(t_{n+2}, y_{n+2}) \right), \\
  r = 4 : & \quad y_{n+4} = y_{n+3} + \frac{h}{24} \left( -9f(t_n, y_n) + 37f(t_{n+1}, y_{n+1}) -59f(t_{n+2}, y_{n+2}) + 55f(t_{n+3}, y_{n+3}) \right). 
\end{align*}
\]

For \( r = 1 \) we recover the explicit Euler method.
The implicit **Adams-Moulton methods** for \( r = 1 \) to \( 4 \) read:

\[
\begin{align*}
    r = 1 : & \quad y_{n+1} = y_n + \frac{h}{2} \left( f(t_n, y_n) + f(t_{n+1}, y_{n+1}) \right), \\
    r = 2 : & \quad y_{n+2} = y_{n+1} + \frac{h}{12} \left( -f(t_n, y_n) + 8f(t_{n+1}, y_{n+1}) \\
                  & \quad + 5f(t_{n+2}, y_{n+2}) \right), \\
    r = 3 : & \quad y_{n+3} = y_{n+2} + \frac{h}{24} \left( f(t_n, y_n) - 5f(t_{n+1}, y_{n+1}) \\
                  & \quad + 19f(t_{n+2}, y_{n+2} + 9f(t_{n+3}, y_{n+3})) \right), \\
    r = 4 : & \quad y_{n+4} = y_{n+3} + \frac{h}{720} \left( -19f(t_n, y_n) + 106f(t_{n+1}, y_{n+1}) \\
                  & \quad - 264f(t_{n+2}, y_{n+2} + 646f(t_{n+3}, y_{n+3}) \\
                  & \quad + 251f(t_{n+4}, y_{n+4}) \right).
\end{align*}
\]

For \( r = 1 \) we recover the trapezoidal rule.
2.4.2 Nyström Methods

Nyström methods have the form

\[ y_{n+r} - y_{n+r-2} = h \sum_{j=0}^{r} \beta_j f(t_{n+j}, y_{n+j}). \]  \hspace{1cm} (2.3)

The explicit Nyström method of order \( r = 2 \) is the midpoint rule (2.1).

The implicit Nyström method of order \( r = 2 \) is Simpson’s rule

\[ y_{n+2} - y_n = \frac{h}{3} \left( f(t_n y_n) + 4f(t_{n+1}, y_{n+1}) + f(t_{n+2}, y_{n+2}) \right). \]
2.4.3 BDF Methods

In contrast with Runge-Kutta and Adams methods, which were derived using quadrature, the BDF methods (backward differentiation formulas) are based on numerical differentiation.

Construction of linear \( r \)-step method:

1. Assume approximations \( y_n, y_{n-1}, \ldots, y_{n-r+1} \) available.

2. To compute new approximation \( y_{n+1} \), denote by \( p \in \mathcal{P}_r \) the unique interpolating polynomial through the \( r + 1 \) points \( \{(t_j, y_j)\}_{j=n-r+1}^{n+1} \).

3. The missing interpolation condition \( p(t_{n+1}) = y_{n+1} \) (we have not yet computed \( y_{n+1} \)) is replaced by the requirement

\[
p'(t_{n+1}) = f(t_{n+1}, y_{n+1}), \tag{2.4}
\]

i.e., that \( p \) satisfy the differential equation at \( t_{n+1} \).
The Newton form of the resulting interpolation polynomial is given by

\[ p(t) = y_{n+1} + y_{n+1,n}(t - t_{n+1}) + y_{n+1,n,n-1}(t - t_{n+1})(t - t_n) + \cdots + y_{n+1,n,...,n-r+2,n-r+1}(t - t_{n+1}) \cdots (t - t_{n-r+2}) \]

in terms of the \( r + 1 \) divided differences \( y_{n+1}, \cdots, y_{n+1,n,...,n+1-r} \).

Under the simplifying assumption of constant stepsize \( h \), its derivative at \( t = t_{n+1} \) is

\[ p'(t_{n+1}) = \sum_{j=1}^{r} y_{n+1,n,...,n+1-j} h^{j-1} (j - 1)! . \tag{2.5} \]

Introducing the backward differences

\[
\begin{align*}
\nabla^0 y_{n+1} & := y_{n+1} \\
\nabla y_{n+1} & := y_{n+1} - y_n, \\
\nabla^j y_{n+1} & := \nabla(\nabla^{j-1} y_{n+1}) = \nabla^{j-1} y_{n+1} - \nabla^{j-1} y_n, \quad j = 2, 3, \ldots,
\end{align*}
\]
$p'(t_{n+1})$ may be expressed as

$$p'(t_{n+1}) = \frac{1}{h} \sum_{j=1}^{r} \frac{\nabla^j y_{n+1}}{j},$$

which, combined with condition (2.4), yields the BDF method of order $r$ as

$$\sum_{j=1}^{r} \frac{1}{j} \nabla^j y_{n+1} = hf(t_{n+1}, y_{n+1}), \quad r \in \mathbb{N},$$

(2.6)

denoted BDF$_r$ for short.

The index translation $n + 1 \mapsto n + r$ and dividing both sides by the coefficient of $y_{n+1}$ transforms the BDF formulas (2.6) to our standard notation (LMM) for linear multistep methods.
The BDF methods for $r = 1$ to 6 are:

$r = 1 : \quad y_{n+1} - y_n = hf(t_{n+1}, y_{n+1}),$

$r = 2 : \quad \frac{3}{2} y_{n+2} - 2 y_{n+1} + \frac{1}{2} y_n = hf(t_{n+2}, y_{n+2})$

$r = 3 : \quad \frac{11}{6} y_{n+3} - 3 y_{n+2} + \frac{3}{2} y_{n+1} - \frac{1}{3} y_n = hf(t_{n+3}, y_{n+3}),$

$r = 4 : \quad \frac{25}{12} y_{n+4} - 4 y_{n+3} + 3 y_{n+2} - \frac{4}{3} y_{n+1} + \frac{1}{4} y_n = hf(t_{n+4}, y_{n+4}),$

$r = 5 : \quad \frac{137}{60} y_{n+5} - 5 y_{n+4} + 5 y_{n+3} - \frac{10}{3} y_{n+2} + \frac{5}{4} y_{n+1} - \frac{1}{5} u_n
\quad = hf(t_{n+5}, y_{n+5}),$

$r = 6 : \quad \frac{147}{60} y_{n+6} - 6 y_{n+5} - \frac{15}{2} y_{n+4} - \frac{20}{3} y_{n+3} + \frac{15}{4} y_{n+2} - \frac{6}{5} y_{n+1} + \frac{1}{6} u_n
\quad = hf(t_{n+6}, y_{n+6}).$

For $r = 1$ we recover the implicit Euler method.
2.4.4 The Local Discretization Error of LMM

Inserting the exact solution into (LMM), using the fact that \( y(t) \) satisfies \( y'(t_n) = f(t_n, y(t_n)) \), we obtain after dividing by \( h \)

\[
T_{n+r}(h) = \frac{1}{h} \left( \sum_{j=0}^{r} \alpha_j y(t_{n+j}) - h \sum_{j=0}^{r} \beta_j y'(t_{n+j}) \right).
\]

Expanding all evaluations of \( y \) and \( y' \) at \( t = t_n \)

\[
y(t_{n+j}) = y(t_n) + jh y'(t_n) + \frac{(jh)^2}{2} y''(t_n) + \frac{(jh)^3}{6} y'''(t_n) + \ldots
\]

\[
y'(t_{n+j}) = y'(t_n) + jh y''(t_n) + \frac{(jh)^2}{2} y'''(t_n) + \frac{(jh)^3}{6} y''''(t_n) + \ldots,
\]

and collecting all terms with the same power of \( h \), we obtain . . .
\[ T_{n+r}(h) = \frac{1}{h} \left( \sum_{j=0}^{r} \alpha_j \right) y(t_n) + \left( \sum_{j=0}^{r} (j \alpha_j - \beta_j) \right) y'(t_n) + h \left( \sum_{j=0}^{r} \left( \frac{j^2}{2} \alpha_j - j \beta_j \right) \right) y''(t_n) + \cdots + h^q \left( \sum_{j=0}^{r} \left( \frac{j^{q+1}}{(q+1)!} \alpha_j - \frac{j^q}{q!} \beta_j \right) \right) y^{(q+1)}(t_n) + O(h^{q+1}). \]

The LMM is consistent, i.e., \( T_n(h) \to 0 \) as \( h \to 0 \), if and only if

\[ \sum_{j=0}^{r} \alpha_j = 0 \quad \text{as well as} \quad \sum_{j=0}^{r} j \alpha_j = \sum_{j=0}^{r} \beta_j. \quad (2.7) \]

Its order of consistency is \( p \in \mathbb{N} \) whenever the first \( p + 1 \) terms in brackets vanish.
2.4.5 Characteristic Polynomials

The polynomials formed with the coefficients \( \{\alpha_j\}_{j=0}^{r} \) and \( \{\beta_j\}_{j=0}^{r} \) of a LMM, known as its characteristic polynomials

\[
\rho(\zeta) := \sum_{j=0}^{r} \alpha_j \zeta^j \quad \text{and} \quad \sigma(\zeta) := \sum_{j=0}^{r} \beta_j \zeta^j,
\]

play an important role in its analysis.

For an \( r \)-step method we have \( \rho \in \mathcal{P}_r \) and, for an implicit \( r \)-step method, also \( \sigma \in \mathcal{P}_r \). When an \( r \) step method is explicit the degree of \( \sigma \) ist strictly less than \( r \).

The two consistency conditions (2.7) can be formulated in terms of the two characteristic polynomials as

\[
\rho(1) = 0 \quad \text{und} \quad \rho'(1) = \sigma(1).
\]
Example: For the Adams-Moulton method with $r = 2$

$$y_{n+2} = y_{n+1} + \frac{h}{12} \left( -f(t_n, y_n) + 8f(t_{n+1}, y_{n+1}) + 5f(t_{n+2}, y_{n+2}) \right),$$

we have

$$\alpha_0 = 0, \alpha_1 = -1, \alpha_2 = 1, \quad \beta_0 = \frac{-1}{12}, \quad \beta_1 = \frac{8}{12}, \quad \beta_2 = \frac{5}{12}$$

and the characteristic polynomials are

$$\rho(\zeta) = \zeta^2 - \zeta, \quad \sigma(\zeta) = \frac{1}{12} \left( 5\zeta^2 + 8\zeta - 1 \right).$$
Because of

\[ \alpha_0 + \alpha_1 + \alpha_2 = 0, \]
\[ 0 \cdot \alpha_0 + 1 \cdot \alpha_1 + 2 \cdot \alpha_2 - (\beta_0 + \beta_1 + \beta_2) = 0, \]
\[ 0 \cdot \alpha_0 + \frac{1}{2} \cdot \alpha_1 + \frac{4}{2} \cdot \alpha_2 - (0 \cdot \beta_0 + 1 \cdot \beta_1 + 2 \cdot \beta_2) = 0, \]
\[ \frac{1}{6} (0 \cdot \alpha_0 + 1 \cdot \alpha_1 + 8 \alpha_2) - \frac{1}{2} (0 \cdot \beta_0 + 1 \cdot \beta_1 + 4 \cdot \beta_2) = 0, \]
\[ \frac{1}{24} (0 \cdot \alpha_0 + 1 \cdot \alpha_1 + 16 \alpha_2) - \frac{1}{6} (0 \cdot \beta_0 + 1 \cdot \beta_1 + 8 \cdot \beta_2) = \frac{5}{8} - \frac{2}{3} \neq 0 \]

this method is consistent of order \( p = 3 \).
2.4.6 Starting Values

When an $r$-step method ($r > 1$) is started, it is necessary to calculate, besides $y_0$, the additional solution approximations $y_1, y_2, \ldots, y_{r-1}$.

To maintain the method’s order $p$, it is sufficient to calculate these using any other method of order at least $p - 1$.

**Explanation:** Since only a fixed number of steps ($r - 1$) is performed with this startup method, independently of $h$, no power of $h$ is lost in the accumulation of step errors as $h \to 0$; for this reason the order of accuracy depends only on that of the method used from step $r$ onwards.

**Example:** To generate $y_1$ to start the midpoint rule (2.1), which is consistent of order $p = 2$, one can use the explicit Euler method without loss of order.
2.4.7 Predictor-Corrector Methods

A popular technique for avoiding the expensive solution of algebraic equation systems required by implicit LMM consists of

(a) first computing an initial approximation \( \hat{y}_{n+r} \) of \( y(t_{n+r}) \) using an explicit method with the same step size (predictor) and

(b) then inserting \( \hat{y}_{n+r} \) in place of \( y_{n+r} \) in an implicit method (corrector).

Example: Combination of \( r = 1 \) Adams-Bashforth method (explicit Euler) with Adams-Moulton method with same step number (trapezoidal rule):

\[
\hat{y}_{n+1} = y_n + hf(t_n, y_n), \quad \text{(predictor step)},
\]

\[
y_{n+1} = y_n + \frac{h}{2} \left( f(t_n, y_n) + f(t_{n+1}, \hat{y}_{n+1}) \right), \quad \text{(corrector step)}.
\]

It can be shown: this predictor/corrector combination has consistency order \( p = 2 \).
2.4.8 One-Step vs. Multistep Methods

Advantages of one-step methods:

- One-step methods are self-starting.
- Changing the stepsize $h$ possible without extra effort.
- Integration across discontinuities of the solution possible without loss of order if these are grid points.

Advantage of multistep methods: one evaluation of right hand side per time step (important if $f$ expensive to evaluate).
2.5 Convergence of One-Step Methods

Theorem 2.3 (Relation between local and global discretization error)
Under the given assumptions (cf. Theorem 1.1 as well as (V_2)) there holds

\[ \|e_n\| \leq \left( (\|e_0\| + (t_n - t_0) \max_{1 \leq j \leq n} \|T_j\|) \exp(M(t_n - t_0)) \right). \]

In particular: a one-step method is convergent if, and only if, it is consistent.
Theorem 2.4 (Total error of explicit OSM) Let the explicit one-step method (OSM) be realized in floating point arithmetic with machine precision $\varepsilon$ as

$$\tilde{y}_{n+1} = \tilde{y}_n + h\Phi_f(\tilde{y}_n, t_n; h) + \varepsilon_{n+1}, \quad \tilde{y}_0 = y_0 + \varepsilon_0.$$ 

If $\|\varepsilon_n\| \leq \varepsilon$ and $\|T_n\| \leq T$ for all $n = 0, 1, \ldots$, then

$$\|y(t_n) - \tilde{y}_n\| \leq (\|\varepsilon_0\| + (t_n - t_0)(T + \frac{\varepsilon}{h})) \exp(M(t_n - t_0)).$$

Dilemma: For $h$ large, in general (local error) $T$ large; for $h$ small $T$ becomes small, however $\frac{\varepsilon}{h}$ grows, i.e., total error dominated by rounding error.

This fact favors high order methods, which achieve small $T$ for moderate values of the step size $h$. 
2.6 Zero-Stability of Linear Multistep Methods

Convergence of one-step methods: Contribution of $n$-th step error $S_n$ bounded by $e^M(t_{\text{end}}-t_0)\|S_n\|$, which still goes to zero as $h \to 0$.

This property of the error is called **zero-stability**, and it always holds for OSM. This is not the case for LMM.

**Example:** For the LMM $y_{n+2} - 3y_{n+1} + 2y_n = -hf(t_n, y_n)$ the local discretization error is

$$T_{n+2}(h) = \frac{h}{2} y''(t_n) + O(h^2).$$

We apply it to IVP

$$y'(t) = 0, \quad y(0) = 0 \quad \text{with exact solution } y(t) \equiv 0. \quad (2.10)$$

For starting values $y_0 = y_1 = 0$ it indeed yields the exact solution.
However, setting $y_1 = h$ to simulate an error of order $h$ when approximating $y(t_1)$, one obtains for the approximation of the solution at $t = 1$ the following values ($h = 1/N$):

<table>
<thead>
<tr>
<th>$N$</th>
<th>$y_N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>6.2</td>
</tr>
<tr>
<td>10</td>
<td>102.3</td>
</tr>
<tr>
<td>20</td>
<td>$5.2 \times 10^4$</td>
</tr>
</tbody>
</table>

Since $y_1 \to 0$ as $h \to 0$ these initial data are admissible. Nonetheless, $y_N$ fails to converge to $y(t_N) = y(1) = 0$.

The approximations satisfy the difference equation

$$y_{n+2} - 3y_{n+1} + 2y_n = 0, \quad n = 0, 1, 2, \ldots, \quad (2.11)$$

with solution

$$y_n = (2^n - 1)y_1 - (2^n - 2)y_0 = 2y_0 - y_1 + 2^n(y_1 - y_0), \quad n \geq 2.$$
2.6.1 Solution of Linear Difference Equations

Solutions of a homogeneous linear difference equation of order $r$

$$\alpha_r y_{n+r} + \alpha_{r-1} y_{n+r-1} + \cdots + \alpha_0 y_n = \sum_{j=0}^{r} \alpha_j y_{n+j} = 0 \quad (2.12)$$

can be obtained by hypothesizing solutions of the form $y_n = \zeta^n$ („Ansatz“). Inserting in (2.12) and dividing by $\zeta^n$ leads to the condition

$$\sum_{j=0}^{r} \alpha_j \zeta^j = 0,$$

i.e., $y_n = \zeta^n$ solves (2.12) if $\zeta$ is a root of the first characteristic polynomial $\rho(\zeta)$ (cf. (2.8)).
If the characteristic polynomial $\rho$ has the factored form
\[
\rho(\zeta) = \alpha_r (\zeta - \zeta_1)(\zeta - \zeta_2) \cdots (\zeta - \zeta_r)
\]
with $r$ distinct roots $\{\zeta_j\}_{j=1}^r$, then $\{\zeta_j^n\}_{j=1}^r$ constitute $r$ linearly independent solutions of (2.12). Due to the linearity and homogeneity of (2.12), any linear combination
\[
y_n = c_1 \zeta_1^n + c_2 \zeta_2^n + \cdots + c_r \zeta_r^n, \quad c_1, \ldots, c_r \text{ arbitrary,}
\]
(2.13) of these is also a solution.

Coefficients $\{c_j\}_{j=1}^r$ determined by fixing first $r$ values of $y_n$:
\[
c_1 + c_2 + \cdots + c_r = y_0, \\
c_1 \zeta_1 + c_2 \zeta_2 + \cdots + c_r \zeta_r = y_1, \\
\vdots \\
c_1 \zeta_1^{r-1} + c_2 \zeta_2^{r-1} + \cdots + c_r \zeta_r^{r-1} = y_{r-1}.
\]
**Example:** The characteristic equation of the difference equation (2.11) is

\[ \rho(\zeta) = \zeta^2 - 3\zeta + 2 = (\zeta - 1)(\zeta - 2). \]

**Conclusion:** If \( \rho(\zeta) \) has roots of modulus \( > 1 \), the associated LMM cannot be convergent.

If two (or more) roots coincide, i.e., \( \zeta_i = \zeta_j \), then \( \zeta^n_i \) and \( \zeta^n_j \) are linearly dependent. A further, linearly independent solution is \( y_n = n\zeta^{n-1}_i \).

For a root \( \zeta_i \) of multiplicity three, \( y_n = n^2\zeta^{n-2}_i \) is linearly independent etc.

**Example:** The (consistent) LMM

\[ y_{n+2} - 2y_{n+1} + y_n = \frac{h}{2} \left[ f(t_{n+2}, y_{n+2}) - f(t_n, y_n) \right]. \]

applied to \( y'(t) = 0 \).

**Conclusion:** If \( \rho(\zeta) \) has multiple roots of modulus 1, the associated LMM is also not convergent.
**Example:** The consistent LMM

\[ y_{n+3} - 2y_{n+2} + \frac{5}{4}y_{n+1} - \frac{1}{4}y_n = \frac{h}{4} f(t_n, y_n) \]

applied to (2.10).

**Conclusion:** For multiple roots of modulus \( > 1 \), the associated LMM is convergent (linear growth is outweighed by exponential decay).

A LMM is called zero-stable, if the roots \( \{\zeta_j\}_{j=1}^r \) of its first characteristic polynomial \( \rho(\zeta) \) satisfy

(a) \( |\zeta_j| \leq 1 \) for all roots.

(b) \( |\zeta_j| < 1 \) for all multiple roots.

**Examples:**

- All Adams-methods are zero-stable.
- The BDF-methods are zero-stable for \( 1 \leq r \leq 6 \).
These examples indicate that zero-stability is a necessary condition for convergence of LMMs. In fact, this condition is also sufficient.

**Theorem 2.5 (Dahlquist, 1956)** For LMM applied to (IVP),

\[ \text{consistency} + \text{zero-stability} \iff \text{convergence}. \]

**Remarks 2.6**

(a) The first characteristic polynomial of a consistent LMM always possesses the root \( \zeta = 1 \) (cf. (2.9)).

(b) We will see that, besides zero-stability, there are further important stability properties for numerical methods for IVPs.

(c) All single-step methods are zero-stable.
2.7 Stepsize Control

In practice, numerical methods for solving IVPs for ODEs almost never employ a constant stepsize $h$. Rather, it is much more efficient to adapt $h$ to the local behavior of the solution $y$ in the sense that

- rapid change in $y$ necessitates small $h$
- slow variation of $y$ allows large $h$.

Here we will introduce one (of many) stepsize control schemes aimed at maintaining the (easily estimated) local discretization error $T_n$ near a prescribed tolerance $tol$

$$\|T_n\| \sim tol, \quad n = 1, 2, \ldots .$$

For systems of ODEs (particularly when solution components vary strongly in magnitude) absolute error tolerances for each solution component as well as a global error tolerance are typically supplied.
Theorem 2.3 states that bounding the local discretization error also results in bounding the global discretization error (the actual quantity of interest).

The local discretization error may be estimated by using two methods of different consistency orders, say, $p$ and $q$ where $p < q$, for computing $y_n$ from $y_{n-1}$:

$$y_n = y_{n-1} + h \Phi_f(y_{n-1}, t_{n-1}; h) \quad \text{and} \quad \hat{y}_n = y_{n-1} + h \hat{\Phi}_f(y_{n-1}, t_{n-1}; h)$$

with associated local discretization errors

$$T_n = \frac{y(t_n) - y(t_{n-1})}{h} - \Phi_f(y(t_{n-1}), t_{n-1}; h) = O(h^p),$$

$$\hat{T}_n = \frac{y(t_n) - y(t_{n-1})}{h} - \hat{\Phi}_f(y(t_{n-1}), t_{n-1}; h) = O(h^q).$$

This implies

$$T_n - \hat{T}_n = \hat{\Phi}_f(y_{n-1}, t_{n-1}; h) - \Phi_f(y_{n-1}, t_{n-1}; h) = \frac{1}{h} (\hat{y}_n - y_n).$$
Because $T_n - \hat{T}_n = T_n(1 + O(h^{q-p})) \sim T_n$, the quantity

$$\frac{1}{h} \| y_n - \hat{y}_n \| \sim \| T_n \|$$

provides a (rough) estimate for $\| T_n \|$.

Whenever $\frac{1}{h} \| y_n - \hat{y}_n \| > \text{tol}$, we discard the stepsize $h$, replacing it with $\tilde{h}$ determined by

$$\left( \frac{\tilde{h}}{h} \right)^p = \alpha \frac{h \text{tol}}{\| y_n - \hat{y}_n \|}. \quad (\star)$$

This choice of $\tilde{h}$ is motivated as follows:

- **discarded stepsize** $h$: $\frac{1}{h} \| y_n - \hat{y}_n \| \sim \| T_n \| = ch^p + O(h^{p+1}) \sim ch^p$,

- **desired stepsize** $\tilde{h}$: $\text{tol} = \| T_n \| = c\tilde{h}^p + O(\tilde{h}^{p+1}) \sim c\tilde{h}^p$.

($\alpha$ is added as a safety factor, e.g. $\alpha = 0.9$.)
Starting again from \( y_{n-1} \), approximations \( y_n \) and \( \hat{y}_n \) — now of \( y(t_{n-1} + \tilde{h}) \) — are then generated and this process repeated until \( \frac{1}{h} ||y_n - \hat{y}_n|| \leq \text{tol} \) holds. When this is achieved, (*) is used to propose a (trial) stepsize for the next step \((n \rightarrow n + 1)\).

An elegant way to limit the computing effort is to compute \( y_n \) and \( \hat{y}_n \) using two RKM (of different orders) whose Butcher tableaus differ only in the vector \( b \) (same \( A \) and \( c \)). As a result, the slopes \( k_j \) of their stages are also the same. Such pairs of RKMs are called embedded RKMs and denoted

\[
\begin{array}{c|c}
\hat{c} & \hat{A} \\
\hline
\hat{b}^\top \\
\end{array}
\]

\[
\begin{array}{c|ccc}
0 & 0 & 0 \\
1 & 0 & 0 \\
1/2 & 1/2 \\
\end{array}
\]

Here the explicit Euler method \((p = 1)\) is embedded in the improved Euler method \((p = 2)\).
Another popular embedded pair is the **Fehlberg 4(5)-formula**: 

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1/4</td>
<td>1/4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3/8</td>
<td>3/32</td>
<td>9/32</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12/13</td>
<td>1932/2197</td>
<td>-7200/2197</td>
<td>7296/2197</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>439/216</td>
<td>-8</td>
<td>3680/513</td>
<td>-845/4104</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1/2</td>
<td>-8/27</td>
<td>2</td>
<td>-3544/2565</td>
<td>1859/4104</td>
<td>-11/40</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>25/216</td>
<td>0</td>
<td>1408/2565</td>
<td>2197/4104</td>
<td>-1/5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>16/135</td>
<td>0</td>
<td>6656/12825</td>
<td>28561/56430</td>
<td>-9/50</td>
<td>2</td>
<td>55</td>
</tr>
</tbody>
</table>

consisting of two six-stage RKMs of orders 4 and 5.
A further example is the Dormand-Prince 4(5)-formula:

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1/5</td>
<td>1/5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3/10</td>
<td>3/40</td>
<td>9/40</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4/5</td>
<td>44/45</td>
<td>−56/15</td>
<td>32/9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8/9</td>
<td>19372/6561</td>
<td>−25360/2187</td>
<td>64448/6561</td>
<td>−212/729</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>9017/3168</td>
<td>−355/33</td>
<td>46732/5247</td>
<td>49/176</td>
<td>−5103/18656</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>35/384</td>
<td>0</td>
<td>500/1113</td>
<td>125/192</td>
<td>−2187/6784</td>
<td>11/84</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>35/384</td>
<td>0</td>
<td>500/1113</td>
<td>125/192</td>
<td>−2187/6784</td>
<td>11/84</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>5179/57600</td>
<td>0</td>
<td>7571/16695</td>
<td>393/640</td>
<td>−92097/339200</td>
<td>187/2100</td>
<td>1/40</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Here a six-stage RKM of order 4 is embedded in a seven-stage method of order 5.