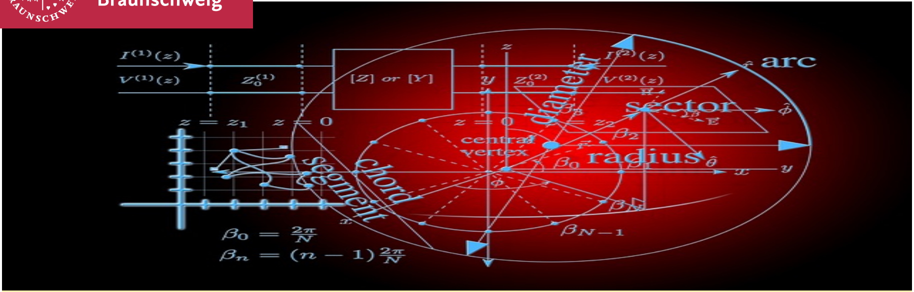
Technische Universität Braunschweig



Introduction to Scientific Computing

Multistep methods

Bojana Rosić, 27. Januar 2016

Multistep methods

Numerical methods for solving ODE

$$\dot{x} = f(x, t)$$

can be classified as

- **one step methods** : these methods use only information from one time point to compute the next

$$x_{n+1} = x_n + hf(t_n, x_n) \quad \text{requires knowledge on} \quad x_n$$

- **multistep methods**: these methods require knowledge on more than one time point

$$x_{n+1} = x_n + h\left(\frac{3}{2}f_n - \frac{1}{2}f_{n-1}\right) \quad \text{requires knowledge on} \quad x_n, x_{n-1}$$

Explicit multistep methods

Given ODE

$$\dot{x} = f(x, t)$$

we assume that the solution x is given at k time points

$$x_n = x(t_n), \quad x_{n-1} = x(t_{n-1}), \ldots, \quad x_{n-k+1} = x(t_{n-k+1})$$

The goal is to compute the solution at the next time point

$$x_{n+1} = x(t_{n+1})$$

Explicit multistep methods

As before

$$x(t_{n+1}) - x(t_{n-j}) = \int_{t_{n-j}}^{t_{n+1}} f(t, x(t))dt$$

with the only difference that the time derivative f is integrated on the time interval from t_{n-j} up to t_{n+1} where $j > 0$ and $j \leqslant k$. Going back to the previous lecture, we have learned that this integration can be performed by approximating **the time derivative** f by a polynomial P (polynomial interpolation) of degree p

$$P(t) = \sum_{k=0}^{p} f_{n-k}\ell_{n-k} = \sum_{k=0}^{p} f_{n-k} \prod_{\substack{l=0 \\ l \neq k}}^{p} \frac{t - t_{n-l}}{t_{n-k} - t_{n-l}}.$$

Explicit multistep methods

In such a case the previous integration reduces to the integration of polynomial P from t_{n-j} to t_{n+1}:

$$\int_{t_{n-j}}^{t_{n+1}} P(t)dt = \sum_{k=0}^{p} f_{n-k} \int_{t_{n-j}}^{t_{n+1}} \prod_{\substack{l=0 \\ l\neq k}}^{p} \frac{t - t_{n-l}}{t_{n-k} - t_{n-l}} dt = \sum_{k=0}^{p} w_{jknp} f_{n-k}.$$

Here, w_{jkip} are weights whose index depends on the polynomial degree p, current point k, begining of integration j and the end of integration n.

Explicit multistep methods

According to this, the explicit multistep method obtains the form

$$x(t_{n+1}) = x(t_{n-j}) + \sum_{k=0}^{p} w_{jknp} f_{n-k},$$

The time interval can be discretised with arbitrary stepsizes. In this case the weights w_{jknp} have to be computed in each time step during the simulation (see Shampine & Gordon 1975). If an equidistant grid is used, then the weights can be calculated before the simulation. They can be stored in a table.

Adams-Bashforth formulas

This type of method uses the formula

$$x(t_{n+1}) = x(t_{n-j}) + \sum_{k=0}^{p} w_{jkip} f_{n-k},$$

in which $j = 0$, i.e. the value of x_{n+1} is found by integrating from x_n

$$x_{n+1} = x_n + \int_{t_n}^{t_{n+1}} f dt$$

and approximating the function f by a polynomial P of degree p.

One example of Adams-Bashforth formulas

In other words, given points $x_n \Rightarrow f_n$ and $x_{n-1} \Rightarrow f_{n-1}$ one uses the interpolation polynomial $P = at + b$ to approximate the time derivative f such that

$$P(t_n) = at_n + b = f_n$$

and

$$P(t_{n-1}) = at_{n-1} + b = f_{n-1}$$

holds. From these two equations one may compute a and b and obtain $P(t)$ which is then substituted into

$$x_{n+1} = x_n + \int_{t_n}^{t_{n+1}} f dt = x_n + \int_{t_n}^{t_{n+1}} P(t) dt$$

to obtain the final formula $x_{n+1} = x_n + \frac{3}{2} hf_n - \frac{1}{2} hf_{n-1}$.

Adams-Bashforth formulas

Taking different number of points and $j = 0$ we get the Adams-Bashforth formulas (1883):

$$
\begin{aligned}
k &= 1: \quad & x_{n+1} &= x_n + hf_n, \\
k &= 2: \quad & x_{n+1} &= x_n + h\left(\tfrac{3}{2} f_n - \tfrac{1}{2} f_{n-1}\right), \\
k &= 3: \quad & x_{n+1} &= x_n + h\left(\tfrac{23}{12} f_n - \tfrac{16}{12} f_{n-1} + \tfrac{5}{12} f_{n-2}\right), \\
k &= 4: \quad & x_{n+1} &= x_n + h\left(\tfrac{55}{24} f_n - \tfrac{59}{24} f_{n-1} + \tfrac{37}{24} f_{n-2} - \tfrac{9}{24} f_{n-3}\right)
\end{aligned}
$$

Note: The first method is explicit Euler rule (one step method obtained by piecewise interpolation). Other methods are multistep as they need more than one starting value. In case when these values are not given, they can be computed with a one-step method.

Nyström formulas

For $j = 1$ we get the Nyström formulas

$$
\begin{aligned}
k &= 2: \quad & x_{n+1} &= x_{n-1} + 2hf_n, \\
k &= 3: \quad & x_{n+1} &= x_{n-1} + h\left(\tfrac{7}{3}f_n - \tfrac{2}{3}f_{n-1} + \tfrac{1}{3}f_{n-2}\right), \\
k &= 4: \quad & x_{n+1} &= x_n + h\left(\tfrac{8}{3}f_n - \tfrac{5}{3}f_{n-1} + \tfrac{4}{3}f_{n-2} - \tfrac{1}{3}f_{n-3}\right)
\end{aligned}
$$

Note: For $k = 2$ we get the explicit midpoint rule

Implicit multistep methods

Aim: Find a polynomial for f which is determined by the values

$$(t_{n+1}, f(t_{n+1}, x_{n+1})), \ldots, (t_{n-p}, f(t_{n-p}, x_{n-p})).$$

Ansatz:

$$P(t) = \sum_{k=-1}^{p} f_{n-k} \prod_{\substack{l=-1 \\ l \neq k}}^{p} \frac{t - t_{n-l}}{t_{n-i} - t_{n-l}}.$$

Integrate the polynomial

$$\int_{t_{n-j}}^{t_{n+1}} P(t)\,dt = \sum_{k=-1}^{p} f_{n-k} \int_{t_{n-j}}^{t_{n+1}} \prod_{\substack{l=-1 \\ l \neq k}}^{p} \frac{t - t_{n-l}}{t_{n-i} - t_{n-l}} dt.$$

Implicit multistep methods

Multistep method:

$$x(t_{n+1}) = x(t_{n-j}) + \sum_{k=-1}^{p} w_{jknp} f_{n-k},$$

with

$$w_{jknp} := \int_{t_{n-j}}^{t_{n+1}} \prod_{\substack{l=-1 \\ l \neq k}}^{p} \frac{t - t_{n-l}}{t_{n-k} - t_{n-l}} dt.$$

Adams-Moulton formulas

For $j = 0$ we get the Adams-Moulton methods:

$$
\begin{aligned}
k &= 1: \qquad & x_{n+1} &= x_n + \frac{h}{2}(f_{n+1} + f_n),\\
k &= 2: \qquad & x_{n+1} &= x_n + h\left(\tfrac{5}{12} f_{n+1} + \tfrac{8}{12} f_n - \tfrac{1}{12} f_{n-1}\right),\\
k &= 3: \qquad & x_{n+1} &= x_n + \frac{h}{24}\left(9 f_{n+1} + 19 f_n - 5 f_{n-1} + f_{n-2}\right)
\end{aligned}
$$

Backward differentiation formulas

This kind of method is also implicit, but difference to Adams like of methods is that the polynomial interpolation in a set of points

$$(t_{n+1}, x(t_{n+1})), \ldots, (t_{k-p}, x(t_{k-p}))$$

is used for the **approximation of the solution** $x(t)$ and not its derivative $f(t, x)$. For example, we may have two points x_n, x_{n+1} used for the interpolation of the solution x by a polynomial

$$P = at + b$$

such that

$$P(t_n) = at_n + b = x_n, \quad P(t_{n+1}) = at_{n+1} + b = x_{n+1}$$

holds.

Backward differentiation formulas

By susbtructing these two equations we have

$$a = \frac{x_{n+1} - x_n}{t_{n+1} - t_n} = \frac{x_{n+1} - x_n}{h}$$

Since

$$\dot{P}(t) = \frac{d}{dt}(at + b) = a, \quad \dot{x}|_t = f(t, x)$$

we require that

$$\dot{P}(t_{n+1}) = a = f(t_{n+1}, x_{n+1})$$

By susbtituting *a* into the last relation one obtains first order backward differentiation formula

$$x_{n+1} = x_n + hf(t_{n+1}, x_{n+1}).$$

Backward differentiation formulas

In a similar manner by taking higher number of known points and polynomial order one may derive the general formula

$$x_{n+1} = \sum_{l=0}^{p} a_{pl} x_{k-l} + h b_0 f(t_{n+1}, x_{n+1}).$$

These methods are called BDF(p)-schemes (backward differentiation formula) (see Gear, 1971).
BDF(2)-scheme:

$$k = 2: \qquad \frac{3}{2} x_n - 2 x_{n-1} + \frac{1}{2} x_{n-2} \quad = \quad h f_n \qquad \text{BDF(2)}$$

Accuracy and Consistency order

Methods we have just introduced are linear multistep methods of the form (only linear combination of x's and f's)

$$\sum_{l=0}^{k} a_l x_{n+l} = h \sum_{l=0}^{k} b_l f(t_{n+l}, x_{n+l})$$

Accuracy can be studied in a similar manner as in one step methods. Such a derivation then leads to the defintion of the consistency order p.

LMM: local error on example

The principle is similar as for one step method. Compute the exact solution of ODE $\dot{x} = x$ by

$$x_a(t_n) = 2\exp(t_n)$$

Then, compute the numerical solution in the next step t_{n+1} given exact solution in the previous steps

$$x_{n+2} = x_a(t_{n+1}) + \frac{h}{2}(-f(t_n, x_a(t_n)) + 3f(t_{n+1}, x_a(t_{n+1})))$$

Once the numerical solution is known, evaluate the local error

$$\epsilon_{loc} = x_a(t_{n+2}) - x_{n+2} = x_a(t_{n+2}) - x_a(t_{n+1})$$

$$-0.5h(-f(t_n, x_a(t_n)) + 3f(t_{n+1}, x_a(t_{n+1})))$$

LMM: local error on example

Furthermore, expand the exact solution into the Taylor series

$$x_a(t_{n+2}) = x_a(t_n) + x_a'(t_n)(t_{n+2} - t_n) + \frac{x_a''(t_n)}{2}(t_{n+2} - t_n)^2 + h.o.t.$$

as well as

$$x_a(t_{n+1}) = x_a(t_n) + x_a'(t_n)(t_{n+1} - t_n) + \frac{x_a''(t_n)}{2}(t_{n+1} - t_n)^2 + h.o.t.$$

Note that

$$t_{n+2} - t_n = (n+2)h - nh = 2h$$

and

$$t_{n+1} - t_n = (n+1)h - nh = h$$

.

LMM: local error on example

Also expand $f = x_a'$ into Taylor series

$$x_a'(t_{n+2}) = x_a'(t_n) + x_a''(t_n)(t_{n+2} - t_n) + \frac{x_a'''(t_n)}{2}(t_{n+2} - t_n)^2 + h.o.t.$$

as well as

$$x_a'(t_{n+1}) = x_a'(t_n) + x_a''(t_n)(t_{n+1} - t_n) + \frac{x_a'''(t_n)}{2}(t_{n+1} - t_n)^2 + h.o.t.$$

LMM: local error on example

Substruct $x_a(t_{n+1})$ from $x_a(t_{n+2})$

$$x_a(t_{n+2}) - x_a(t_{n+1}) = x_a' h + \frac{3}{2} x_a'' h^2.$$

and subtitute all previous formulas into the local error

$$\epsilon_{loc} = x_a(t_{n+2}) - x_{n+2} = x_a(t_{n+2}) - x_a(t_{n+1})$$

$$-0.5h(-f(t_n, x_a(t_n)) + 3f(t_{n+1}, x_a(t_{n+1})))$$

LMM: local error on example

Thus,

$$\epsilon_{loc} = x'_a h + \frac{3}{2} x''_a h^2$$

$$-0.5h(-x'_a + 3x'_a + 3x''_a h + 1.5x'''_a h^2)$$

which reduces to

$$\epsilon_{loc} = -\frac{3}{4} x'''_a h^3 = Ch^3$$

Consistency order follows from defition

$$\max \| \frac{\epsilon_{loc}}{h} \| \leqslant Ch^p$$

which gives $p = 2$.

LMM: local error on example

It can then be shown that Adams-Bashforth and BDF methods are of order k (where k is the number of steps), while Adams-Moulton methods are of order $k+1$ (with the exception of the case where the scheme is completed in a single step such as in Backward Euler, the order of which is $k=1$).

LMM: local error

For general ODE the exact solution reads

$$x_a'(t_n) = f(t_n, x_a(t_n))$$

and can be substituted into LMM

$$a_k x(t_{n+k}) + \sum_{j=0}^{k-1} a_j x_a(t_{n+j}) = h \sum_{j=0}^{k} b_j f(t_{n+j}, x_a(t_{n+j}))$$

such that the ***local error*** reads

$$\epsilon_{loc} = x_a(t_{n+k}) - x(t_{n+k})$$

$$\epsilon_{loc} = x_a(t_{n+k}) + a_k^{-1} \left(\sum_{j=0}^{k-1} a_j x_a(t_{n+j}) - h \sum_{j=0}^{k} b_j f(t_{n+j}, x_a(t_{n+j})) \right)$$

LMM: local error

Let us multiply the local error by a_k. Since this is constant it will not change the inequality for the consistency order

$$a_k \epsilon_{loc} = a_k x_a(t_{n+k}) + \sum_{j=0}^{k-1} a_j x_a(t_{n+j}) - h \sum_{j=0}^{k} b_j f(t_{n+j}, x_a(t_{n+j}))$$

which is then

$$a_k \epsilon_{loc} = \sum_{j=0}^{k} a_j x_a(t_{n+j}) - h \sum_{j=0}^{k} b_j f(t_{n+j}, x_a(t_{n+j}))$$

Linear multi-step methods: local error

The value $x_a(t_{n+j})$ can be obtained from the Taylor expansion around the point in time t_n such that

$$x_a(t_{n+j}) = x_a(t_n) + x_a'(t_n)(t_{n+j} - t_n) + \frac{x_a''(t_n)}{2}(t_{n+j} - t_n)^2 + h.o.t.$$

and

$$x_a'(t_{n+j}) = x_a'(t_n) + x_a''(t_n)(t_{n+j} - t_n) + \frac{x_a'''(t_n)}{2}(t_{n+j} - t_n)^2 + h.o.t.$$

Note that the difference

$$t_{n+j} - t_n = (n+j)h - nh = jh$$

Linear multi-step methods: local error

Hence,

$$
\begin{aligned}
a_k \epsilon_{loc} &= \sum_{j=0}^{k} a_j x_a(t_{n+j}) - h \sum_{j=0}^{k} b_j f(t_{n+j}, x_a(t_{n+j})) \\
&= \sum_{j=0}^{k} a_j x_a(t_{n+j}) - h \sum_{j=0}^{k} b_j x_a'(t_{n+j}) \\
&= \sum_{j=0}^{k} a_j \left[x_a(t_n) + x_a'(t_n)(jh) + \frac{x_a''(t_n)}{2}(jh)^2 + h.o.t. \right] \\
&\quad - h \sum_{j=0}^{k} b_j \left[x_a'(t_n) + x_a''(t_n)(jh) + \frac{x_a'''(t_n)}{2}(jh)^2 + h.o.t. \right]
\end{aligned}
$$

Linear multi-step methods: local error

$$
\begin{aligned}
a_k \epsilon_{loc} \quad = \quad & \left[\sum_{j=0}^{k} a_j\right] x_a(t_n) + h \left[\sum_{j=0}^{k} (j a_j - b_j)\right] x_a'(t_n) \\
& + h^2 \left[\sum_{j=0}^{k} (\frac{j^2}{2} a_j - j b_j)\right] x_a''(t_n) + \cdots + \\
& + \cdots + h^{q+1} \left[[\sum_{j=0}^{k} (\frac{j^{q+1}}{(q+1)!} a_j - \frac{j^q}{q!} b_j)\right] x_a^{(q+1)}(t_n) + \mathcal{O}(h^{q+2})
\end{aligned}
$$

Consistency

The linear multistep is consistent if

$$\lim_{h \to 0} \frac{\epsilon_{loc}}{h} = 0$$

$$\frac{\epsilon_{loc}}{h} = \frac{1}{h}\left[\sum_{j=0}^{k} a_j\right] x_a(t_n) + \left[\sum_{j=0}^{k} (ja_j - b_j)\right] x_a'(t_n)$$

$$+h\left[\sum_{j=0}^{k} (\frac{j^2}{2} a_j - jb_j)\right] x_a''(t_n) + \cdots +$$

$$+ \cdots + h^q \left[[\sum_{j=0}^{k} (\frac{j^{q+1}}{(q+1)!} a_j - \frac{j^q}{q!} b_j)\right] x_a^{(q+1)}(t_n) + \mathcal{O}(h^{q+1})$$

Consistency

Hence, the method is consistent if

$$\sum_{j=0}^{k} a_j = 0, \quad \sum_{j=0}^{k} (ja_j - b_j) = 0$$

and

$$\sum_{j=0}^{k} \left(\frac{j^q}{(q)!} a_j - \frac{j^{q-1}}{(q-1)!} b_j \right) = 0, \quad q = 2, \cdots p$$

Consistency of Adams-Bashforth method

Having

$$x_{n+2} = x_{n+1} + \frac{h}{2}(-f(t_n, x_n) + 3f(t_{n+1}, x_{n+1}))$$

where

$$a_0 = 0, \quad a_1 = -1, \quad a_2 = 1$$

and

$$b_0 = -0.5, \quad b_1 = 1.5, \quad b_2 = 0$$

one may check if the following conditions hold

$$\sum_{j=0}^{k} a_j = 0, \quad \sum_{j=0}^{k} (ja_j - b_j) = 0$$

Consistency of Adams-Bashforth method

$$\sum_{j=0}^{k} a_j = 0 - 1 + 1 \equiv 0$$

$$\sum_{j=0}^{k} (ja_j - b_j) = 0 \cdot a_0 - b_0 + a_1 - b_1 + 2a_2 - b_2$$

$$\sum_{j=0}^{k} (ja_j - b_j) = 0.5 - 1 - 1.5 + 2 - 0 \equiv 0$$

Hence, the method is consistent.

Consistency order of Adams-Bashforth method

Let us check now

$$\sum_{j=0}^{k}\left(\frac{j^q}{(q)!}a_j - \frac{j^{q-1}}{(q-1)!}b_j\right) = 0, \quad q = 2, \cdots p$$

$$\sum_{j=0}^{k}\left(\frac{j^2}{(2)!}a_j - \frac{j^{2-1}}{(2-1)!}b_j\right) = -0.5 - 1.5 + 2 - 0 \equiv 0$$

$$\sum_{j=0}^{k}\left(\frac{j^3}{(3)!}a_j - \frac{j^{3-1}}{(3-1)!}b_j\right) = 9/12 \neq 0$$

Hence, the order is $p = 2$.

Truncation error

There is another way of deriving consistency condition. From the defintion of truncation error (last lecture)

$$T_h(t) = \frac{1}{h}\sum_{i=0}^{k} a_i\, x(t+ih) - F(t, x(t), \ldots, x(t+kh), h),$$

we have

$$\begin{aligned} T_h(t) &= \frac{1}{h}\sum_{i=0}^{k} a_i\left(x(t) + ih\dot{x}(\xi)_i\right) - F(t, x_n, \ldots, x_{n+k}, h) \\ &= \frac{1}{h}x(t)\sum_{i=0}^{k} a_i + \sum_{i=0}^{k} ia_i\dot{x}(\xi)_i - F(t, x_n, \ldots, x_{n+k}, h). \end{aligned}$$

Truncation error

Note that

$$x(t+ih) = x(t) + ih\dot{x}(\xi)_i$$

according to the *mean value theorem*. The mean value theorem for functions of one variable implies that there exist a value $\xi_j \in [t, t+jh]$ such that

$$x(t+jh) - x(t) = \dot{x}(\xi)_{(j)} jh.$$

Truncation error

Also, note that

$$\dot{x}(\xi)_{(j)} \longrightarrow \dot{x} = f(t, x(t))$$

for $h \to 0$ (point at $t + ih$ and t match). Hence,

$$\sum_{i=0}^{k} i a_i \dot{x}(\xi)_i = \sum_{i=0}^{k} i a_i \dot{x} = \dot{x} \sum_{i=0}^{k} i a_i$$

Truncation error

Let us assume that $T_h(t) \longrightarrow 0$ for $h \longrightarrow 0$. Then, for $h \longrightarrow 0$ in

$$
\begin{aligned}
T_h(t) &= \frac{1}{h}\sum_{i=0}^{k} a_i \left(x(t) + ih\dot{x}(\xi)_i\right) - F(t, x_n, \ldots, x_{n+k}, h) \\
&\Rightarrow \frac{1}{h}\left(\sum_{i=0}^{k} a_i x + h\sum_{i=0}^{k} i a_i \dot{x}\right) - F(t, x, \ldots, x, h) \\
&= \frac{1}{h}\left(x\sum_{i=0}^{k} a_i + h\dot{x}\sum_{i=0}^{k} i a_i\right) - F(t, x, \ldots, x, h)
\end{aligned}
$$

Truncation error

Hence, in order that $T_h \to 0$ when $h \to 0$ it must be satisfied

$$\sum_{i=0}^{k} a_i = 0$$

as well as

$$\dot{x}(t) \sum_{i=0}^{k} a_i i = F(t, x(t), \dots, x(t), 0), \quad x(t) = x(t_n)$$

$$\Longrightarrow \quad f(t, x(t)) = \frac{F(t, x(t), \dots, x(t), 0)}{\rho'(1)},$$

Truncation error

Note that

$$\rho(\xi) = \sum_{i=0}^{k} a_i \xi^i$$

is the **first characteristic polynomial** of the difference equation

$$\sum_{i=0}^{k} a_i x_{n+i} = hF(t, x_n, \ldots, x_{n+k}, h),$$

when the right hand side is equal to zero

$$\sum_{i=0}^{k} a_i \, x(t + ih) = 0.$$

Consistency

Following previous derivation consistency can be also defined as

Definition

For $h \longrightarrow 0$ the truncation error goes to zero, $T_h(t) \longrightarrow 0$, if and only if both the following conditions hold:

$$\sum_{i=0}^{k} a_i = 0 \Leftrightarrow \rho(1) = 0 \text{ and } f(t, x(t)) = \frac{F(t, x(t), \dots, x(t), 0)}{\rho'(1)} \text{ for } t > 0.$$

Exercise

For the method $\frac{3}{2}x_n - 2x_{n-1} + \frac{1}{2}x_{n-2} = hf_n$ one has

$$\rho = \sum a_i \xi^i = \frac{1}{2} - 2\xi + \frac{3}{2}\xi^2$$

$$\rho' = -2 + 3\xi$$

$$F(t, x(t)) = f(t)$$

Hence,

$$\rho(1) = \frac{1}{2} - 2 + \frac{3}{2} = 0$$

$$\rho'(1) = -2 + 3 = 1$$

$$\frac{F(t, x(t), \ldots, x(t), 0)}{\rho'(1)} = \frac{f(t)}{1} = f(t)$$

Exercise

For the method

$$x_{n+1} = x_n + h\left(\frac{5}{12}f_{n+1} + \frac{8}{12}f_n - \frac{1}{12}f_{n-1}\right)$$

one has

$$\rho = \sum a_i \xi^i = \xi - 1$$

$$\rho' = 1$$

$$F(t, x(t)) = \frac{5}{12}f(t, x(t)) + \frac{8}{12}f(t, x(t)) - \frac{1}{12}f(t, x(t))$$

Hence,

$$\rho(1) = 1 - 1 = 0$$

$$\rho'(1) = 1$$

$$\frac{F(t, x(t), \dots, x(t), 0)}{\rho'(1)} = \frac{f(t)}{1} = f(t)$$

Proof

The approximation scheme has the form

$$\frac{1}{h}\sum_{i=0}^{k} a_i x_{n+i} = F(t, x_n, \dots, x_{n+k}, h).$$

If the method converges to a solution z of the ODE for $h \to 0$, we obtain

$$x_n = x(nh) = x(t) \to z(t), \dots, x_{n+k} = x(t+kh) \to z(t).$$

Proof

From this, it follows that for $h \to 0$

$$\sum_{i=0}^{k} a_i z(t) = \lim_{h \to 0} hF(t, x_n, \ldots, x_{n+k}, h) = 0,$$

and thus $\sum_{i=0}^{k} a_i = 0$. Also, one may conclude that

$$\sum_{i=0}^{k} a_i(x_{n+i} - x_n) = \sum_{i=0}^{k} a_i x_{n+i} - x_n \sum_{i=0}^{k} a_i = \sum_{i=0}^{k} a_i x_{n+i}$$

Hence, the original LMM can be rewritten as

$$\implies \quad \frac{1}{h} \sum_{i=0}^{k} \frac{i}{i} a_i \left(x(t+ih) - x(t)\right) = F(t, x_n, \ldots, x_{n+k}, h).$$

Proof

Putting *ih* into the substruction one has

$$\Longrightarrow \quad \sum_{i=0}^{k} i a_i \frac{(x(t+ih) - x(t))}{ih} = F(t, x_n, \dots, x_{n+k}, h).$$

Note that

$$\lim_{h \to 0} \frac{x(t+ih) - x(t)}{ih} = \dot{z}(t) = f(z, t).$$

Hence, for $h \to 0$ one obtains LMM

$$\sum_{i=0}^{k} a_i i \dot{z}(t) = \dot{z}(t) \sum_{i=0}^{k} a_i i = \dot{z}(t) \rho'(1) = F(t, z(t), \dots, z(t), 0)$$

Proof

Following this one obtains consistency conditions:

$$\dot{\mathbf{z}}(t) = \mathbf{f}(t, \mathbf{z}(t)) = \mathbf{F}(t, \mathbf{z}(t), \ldots, \mathbf{z}(t), 0)/\rho'(1).$$

$$\sum_{i=0}^{k} a_i = \rho(1) = 0$$

Hence, the convergent method is also consistent.

Remarks

Note that we have shown that **convergent** scheme is also consistent.

- The converse statement does not hold in general: The convergence of a scheme is not implied by its consistency alone.
- This fact may be visualised by contemplating that consistency considers the effect of one single step only. The smaller the step–size h becomes, the more steps are needed to integrate the solution up to a given point t, and this rise in the number of steps may result in a divergent scheme, even though the scheme is consistent.

Theorem

Theorem

An integration scheme is convergent if and only if it is consistent and zero stable, and in case it is convergent, the order of consistency and the order of convergence are equal.