Chapter 2
One Step Numerical Schemes

Abstract First the Euler scheme and its local and global discretisation errors are presented. Then several one step schemes such as Taylor schemes, Runge–Kutta schemes are introduced. Consistency and numerical instability are discussed.

Keywords Euler scheme · Local discretisation error · Global discretisation error · One step scheme · Numerical instability.

Consider the initial value problem (IVP)

$$\frac{\mathrm{d}x}{\mathrm{d}t} = f(t, x), \qquad x(t_0) = x_0 \tag{2.1}$$

and assume that the vector field f is at least continuously differentiable and that a unique solution $x(t) = x(t; t_0, x_0)$ exists on an interval $[t_0, T]$. In general, $x(t; t_0, x_0)$ is not explicitly known, so we want to find a numerical approximation of it. The simplest numerical scheme that produces such an approximation is the *Euler scheme*.

Consider a uniform partition of the time interval $[t_0, T]$ of constant stepsize $h = \frac{T-t_0}{N_h} > 0$, i.e., the discrete times $t_0, t_1, \ldots, t_{N_h}$, with

$$t_{n+1} = t_n + h \quad or \quad t_n = t_0 + nh \quad for \quad n = 0, 1, \ldots, N_h.$$

The *Euler scheme* for (2.1) is defined by the difference equation

$$x_{n+1} = x_n + h\, f(t_n, x_n), \qquad n = 0, 1, \ldots, N_h. \tag{2.2}$$

The scheme (2.2) can be derived heuristically by approximating the integral in the integral equation representation of the IVP (2.1). In fact, solutions to (2.1) satisfy the integral equation

$$x(t) = x(t_n) + \int_{t_n}^{t} f(s, x(s))\, \mathrm{d}s$$

on the subinterval $[t_n, t_{n+1}]$. In addition by continuity of f, $f(s, x(s)) \approx f(t_n, x(t_n))$ for all $s \in [t_n, t_{n+1}]$, provided $h > 0$ is small enough. Hence we can obtain the approximation

X. Han and P. Kloeden, *Attractors Under Discretisation*,
SpringerBriefs in Mathematics, DOI 10.1007/978-3-319-61934-7_2

$$x(t_{n+1}) = x(t_n) + \int_{t_n}^{t_{n+1}} f(s, x(s))\,\mathrm{d}s \approx x(t_n) + \int_{t_n}^{t_{n+1}} f(t_n, x(t_n))\,\mathrm{d}s$$
$$= x(t_n) + f(t_n, x(t_n)) \int_{t_n}^{t_{n+1}} \mathrm{d}s = x(t_n) + h\, f(t_n, x(t_n)).$$

2.1 Discretisation Error

A more geometrical derivation of (2.2) is to approximate the solution curve in the interval $[t_n, t_{n+1}]$ by the tangent to the integral curve at the point $(t_n, x(t_n))$.

Obviously we have an error (see Fig. 2.1),

$$\mathscr{E}_{n+1}^L = \|x(t_{n+1}) - x(t_n) - h\, f(t_n, x(t_n))\|\,.$$

It is called the *local discretisation error*.

Note that only in the first subinterval $[t_0, t_1]$ does the Euler scheme start at the same point x_0 as the differential equation. In the next subinterval $[t_1, t_2]$ it starts at $x_1 = x_0 + h\, f(t_0, x_0)$ and in general $x_1 \neq x(t_1)$. Then the local discretisation error

$$\mathscr{E}_2^L = \|x(t_2; t_1, x_1) - x_1 - h\, f(t_1, x_1)\| = \|x(t_2; t_1, x_1) - x_2\|$$

is, in general, not equal to the true error $\mathscr{E}_2 = \|x(t_2; t_0, x_0) - x_2\|$. Due to the continuous dependence of $x(t)$ in the initial conditions we may expect that $\mathscr{E}_2 \sim \mathscr{E}_2^L$ when $h > 0$ is small enough. This is, however, too heuristic. We need, in fact, the *global discretisation error*

$$\mathscr{E}_n := \|x(t_n; t_0, x_0) - x_n\|\,, \quad n = 0, 1, \ldots, N.$$

Clearly, $\mathscr{E}_0 = 0$ and $\mathscr{E}_1 = \mathscr{E}_1^L$, but in general $\mathscr{E}_n \neq \mathscr{E}_n^L$ for $n \geq 2$ (see Fig. 2.2).

Nevertheless, the local discretisation error is important for estimating the global discretisation error. It is easy to estimate the local discretisation error through a Taylor expansion. Let $x(t) = x(t; t_n, x_n)$. Then there exists a $\tau_n \in [t_n, t_{n+1}]$ such that

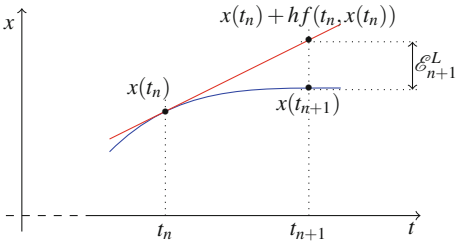

Fig. 2.1 Local discretisation error

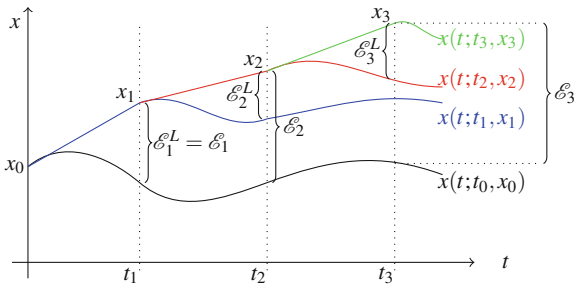

Fig. 2.2 Local versus global discretisation error

$$x(t_{n+1}) = x(t_n) + h\, x'(t_n) + \frac{1}{2!} h^2 \cdot x''(\tau_n)$$

where $x'(t) = \dfrac{\mathrm{d}}{\mathrm{d}t}\, x(t) = f(t, x(t))$ and

$$x''(t) = \frac{\mathrm{d}^2}{\mathrm{d}t^2}\, x(t) = \frac{\mathrm{d}}{\mathrm{d}t}\, f(t, x(t)) = \frac{\partial f}{\partial t}\,(t, x(t)) + \frac{\partial f}{\partial x}\,(t, x(t)) \cdot f(t, x(t)).$$

Note that $x''(t)$ is continuous, as f is continuously differentiable. Then the local discretisation error is

$$\mathscr{E}_{n+1}^L = \|x(t_{n+1}) - x(t_n) - h\, f(t_n, x(t_n))\| = \frac{1}{2!} h^2 |x''(\tau_n)| \le \frac{1}{2}\, h^2\, M, \qquad (2.3)$$

with

$$M := \max_{\substack{t_0 < t \le T \\ \|x\| \le R}} \left\{ \left\| \frac{\partial f}{\partial t}\,(t, x) \right\| + \left\| \frac{\partial f}{\partial x}\,(t, x) \right\| \cdot \| f(t, x) \| \right\} < \infty.$$

The estimate (2.3) is very coarse, but is useful for theoretical purposes. It shows that the local discretisation error is of second order, i.e., $\mathscr{E}_n^L \approx \mathscr{O}(h^2)$. More details on orders of the local and global discretisation error will be discussed in Sect. 2.3.

2.2 General One Step Schemes

The Euler scheme (2.2) is the simplest, nontrivial example of an explicit one step scheme, i.e., x_{n+1} depends explicitly on x_n and is determined once x_n is known. Such methods are ideal for a digital computer.

The general form of an explicit one step scheme with constant step size $h > 0$ is

$$x_{n+1} = x_n + h\, F(t_n, x_n; h),$$

where F is called the *increment function*. For the Euler scheme $F(t, x; h) \equiv f(t, x)$. But in general, F also depends on h, e.g., for the Heun scheme we have

$$F(t, x; h) = \frac{1}{2} \left[f(t, x) + f(t + h,\ x + h\, f(t, x)) \right] .$$

Obviously, to obtain a meaningful numerical scheme the increment function F needs to be related to the vector field function f. The question is then how can we find such an F? As in the Euler case we can use either

1. a Taylor expansion of a solution of the ODE, or
2. an approximation of the integral in the integral equation representation of a solution of the ODE.

Here we consider just the one dimensional case to keep things simple. For higher dimensional cases the reader is referred to [32].

2.2.1 Taylor Schemes

Let $f(t, x)$ be p-times continuously differentiable in both variables and let $x(t) = x(t; t_n, x_n)$ be the unique solution of the initial value problem

$$\frac{\mathrm{d}x}{\mathrm{d}t} = f(t, x), \quad x(t_n) = x_n \, . \tag{2.4}$$

Then we take the p-th order Taylor expansion of the function $x(t)$ about $(t_n, x(t_n))$ in $t = t_{n+1}$. There exists a $\tau_n \in [t_n, t_{n+1}]$ such that

$$x(t_{n+1}) = \sum_{j=0}^{p} \frac{h^j}{j!} x^{(j)}(t_n) + \underbrace{\frac{1}{(p+1)!} h^{p+1} x^{(p+1)}(\tau_n)}_{\text{remainder}}, \tag{2.5}$$

where the derivatives $x^{(j)}(t)$ are defined recursively by

$$\begin{aligned}
x^{(0)}(t) &= x(t), \qquad & x^{(1)}(t) &= \frac{\mathrm{d}}{\mathrm{d}t} x(t) = f(t, x(t)) \\
x^{(2)}(t) &= \frac{\mathrm{d}}{\mathrm{d}t} x^{(1)}(t) = \frac{\mathrm{d}}{\mathrm{d}t} f(t, x(t)) = \left(\frac{\partial f}{\partial t} + f \frac{\partial f}{\partial x} \right) (t, x(t)) =: Df(t, x(t)) \\
x^{(3)}(t) &= \frac{\mathrm{d}}{\mathrm{d}t} x^{(2)}(t) = \left(\frac{\partial}{\partial t} + f \frac{\partial}{\partial x} \right) Df(t, x(t)) := D^2 f(t, x(t)) \\
&= \frac{\partial^2 f}{\partial t^2} + \frac{\partial f}{\partial t} \frac{\partial f}{\partial x} + f \frac{\partial^2 f}{\partial t \partial x} + f \frac{\partial^2 f}{\partial x \partial t} + f \left(\frac{\partial f}{\partial x} \right)^2 + f^2 \frac{\partial^2 f}{\partial x^2},
\end{aligned}$$

and so on.

Here D is called the *total derivative* of f with respect to the ODE $\frac{\mathrm{d}x}{\mathrm{d}t} = f(t,x)$. In general, we write

$$x^{(j+1)}(t_n) = D^j f(t, x(t))\big|_{t=t_n} = D^j f(t_n, x(t_n))$$

for $j = 0, 1, \ldots, p$ with the convention $D^0 f \equiv f$. Then we discard the remainder term in (2.5) and obtain the approximation

$$x(t_{n+1}) \approx x(t_n) + \sum_{j=1}^{p} \frac{1}{j!} h^j D^{j-1} f(t_n, x(t_n)),$$

which motivates the iterative scheme

$$x_{n+1} = x_n + \sum_{j=1}^{p} \frac{h^j}{j!} D^{j-1} f(t_n, x_n),$$

or equivalently

$$x_{n+1} = x_n + h \sum_{i=0}^{p-1} \frac{h^i}{(i+1)!} D^i f(t_n, x_n). \tag{2.6}$$

Formula (2.6) is an explicit one step scheme with increment function

$$F(t,x;h) = \sum_{i=0}^{p-1} \frac{h^i}{(i+1)!} D^i f(t,x)$$

and is called the *Taylor scheme of order* p.

Example 2.1 The Taylor scheme of order 1 reads

$$x_{n+1} = x_n + h f(t_n, x_n),$$

which is just the Euler scheme with $F(t,x;h) = f(t,x)$.

The Taylor scheme of order 2 reads

$$\begin{aligned} x_{n+1} &= x_n + h f(t_n, x_n) + \frac{1}{2} h^2 \left(\frac{\partial f}{\partial t}(t_n, x_n) + f(t_n, x_n) \frac{\partial f}{\partial x}(t_n, x_n) \right) \\ &= x_n + hF(t,x;h) \end{aligned}$$

with

$$F(t,x;h) = f(t,x) + \frac{1}{2} h \left(\frac{\partial f}{\partial t}(t,x) + f(t,x) \cdot \frac{\partial f}{\partial x}(t,x) \right).$$

Remark 2.1 The local discretisation error for the Taylor scheme of order p has order $p+1$. In fact, it is given by the remainder in (2.5)

$$\mathscr{E}_{n+1}^{L} = \frac{1}{(p+1)!} h^{p+1} \left\| D^p f(\tau_n, x(\tau_n)) \right\| \leq \frac{1}{(p+1)!} M_p \cdot h^{p+1}$$

where the constant M_p is defined by

$$M_p := \max_{\substack{t_0 \leq t \leq T \\ |x| \leq R}} \left\| D^p f(t,x) \right\| \tag{2.7}$$

for a suitable R. Here R must be large enough so that $\|x(t)\| \leq R$ for all solutions on $[t_0, T]$ under consideration.

Taylor schemes have rarely been used in practice due to the need to derive higher order derivatives, although these days computer algebra software facilitates this task.[1] They are nevertheless very useful for theoretical reasons. For example, to determine the local discretisation error order of other one step schemes we compare them term by term with an appropriate Taylor scheme for which the local discretisation error order is known.

2.2.2 Schemes Derived by Integral Approximations

Other kinds of one step scheme can be derived by approximating the integrals in the integral equation representation of the solutions of the ODE. Such schemes involve only the values of the function f and not those of its derivatives.

In a subinterval $[t_n, t_{n+1}]$ the integral equation for the solution of the ODE reads

$$x(t) = x(t_n) + \int_{t_n}^{t} f(s, x(s))\, \mathrm{d}s.$$

The integrand function $F(t) = f(t, x(t))$ is continuous, hence integrable on $[t_n, t_{n+1}]$. We can apply different integral approximation rules for the integral

$$\int_{t_n}^{t_{n+1}} F(s)\, \mathrm{d}s, \tag{2.8}$$

such as the rectangle, trapezium and Simpson's rules.

[1] Coombes, Kocak and Palmer [31] used a 31st order Taylor scheme to investigate the 3-dimensional Lorenz system numerically.

2.2.2.1 Rectangle Rule

Using the left hand endpoint as the evaluation point of the integral (2.8) gives the rectangle rule

$$\int_{t_n}^{t_{n+1}} F(s)\,\mathrm{d}s \approx \int_{t_n}^{t_{n+1}} F(t_n)\,\mathrm{d}s = (t_{n+1} - t_n)\,F(t_n)$$

from which we obtain the approximation

$$x(t_{n+1}) \approx x(t_n) + (t_{n+1} - t_n)\,f(t_n, x(t_n))\,.$$

This motivates the *Euler scheme*

$$x_{n+1} = x_n + h\,f(t_n, x_n).$$

Alternatively, using the right hand endpoint as the evaluation point gives

$$\int_{t_n}^{t_{n+1}} F(s)\,\mathrm{d}s \approx \int_{t_n}^{t_{n+1}} F(t_{n+1})\,\mathrm{d}s = (t_{n+1} - t_n)\,F(t_{n+1})\,,$$

which leads to the implicit scheme

$$x_{n+1} = x_n + h\,f(t_{n+1}, x_{n+1}),$$

called the *implicit Euler scheme*.

Remark 2.2 Implicit schemes require additional work at each iteration step to solve an implicit equation for x_{n+1}, e.g., using Newton's method. All the same these schemes are often used on account of their better numerical stability properties and the possibility to use a larger step size. This will be discussed in Sect. 2.5.

2.2.2.2 Trapezium Rule

The trapezium rule uses the average of the left and right hand endpoints and gives

$$\int_{t_n}^{t_{n+1}} F(s)\,\mathrm{d}s \approx \frac{t_{n+1} - t_n}{2}\left[F(t_n) + F(t_{n+1})\right],$$

from which we again obtain an implicit scheme

$$x_{n+1} = x_n + \frac{h}{2}\left[f(t_n, x_n) + f(t_{n+1}, x_{n+1})\right],$$

which is called the *trapezoidal scheme*.

To avoid having to solve an implicit equation we could replace the x_{n+1} on the righthand side of the trapezoidal scheme by the x_{n+1} in the corresponding Euler scheme. Then we obtain an explicit scheme, which is called the *Heun scheme*:

$$x_{n+1} = x_n + \frac{h}{2}\left[f(t_n, x_n) + f(t_{n+1}, x_n + h\, f(f_n, x_n))\right] .$$

It is an explicit one step scheme with the increment function

$$F(t, x; h) = \frac{1}{2}\left[f(t, x) + f(t + h, x + h\, f(t, x))\right] .$$

Such heuristic modifications are typical. We need to take care that so the derived scheme is compatible or consistent with the ODE. A concept of consistency and a quick test for it will be given later Sect. 2.4.

2.2.2.3 Runge–Kutta Schemes

The Heun scheme is one of the simplest nontrivial examples from the family of Runge–Kutta schemes. It has two evaluations points of the function f for each iteration (i.e., each time subinterval). These are the intermediate steps or stages of the scheme. Typical Runge–Kutta schemes have $s \geq 2$ stages. In the case of two stages the increment function of the scheme has the general structure

$$F(t, x; h) = \alpha f(t, x) + \beta\, f(t + \gamma h, x + \gamma h\, f(t, x))$$

for appropriate constants α, β, γ. For the Heun scheme, $\alpha = \frac{1}{2}$, $\beta = \frac{1}{2}$ and $\gamma = 1$.

Runge–Kutta schemes belong to the class of *derivative free one step schemes*, in which a vector field function f is evaluated at several intermediate instants within the discretisation subinterval. More precisely, consider a partition

$$t_0 < t_1 < \cdots < t_n < \cdots < t_N = T, \qquad h_n := t_{n+1} - t_n$$

of the interval $[t_0, T]$ with positive step size $h_n > 0$. The solution $x(t)$ of the IVP (2.4) at t_{n+1} satisfies the integral equation (IE)

$$x(t_{n+1}) = x(t_n) + \int_{t_n}^{t_{n+1}} f(t, x(t))\, \mathrm{d}t. \tag{2.9}$$

For $t \in [t_n, t_{n+1}]$, write $f(t, x(t)) = g(t)$. There are many approximation formulae for an integral such as $\displaystyle\int_{t_n}^{t_{n+1}} g(t)\, \mathrm{d}t$, e.g., the Newton-Cotes and Gauß quadrature rules. Given the number of stages $s \geq 2$, the integral can be approximated by

$$\int_{t_n}^{t_{n+1}} g(t)\,dt \approx h_n \sum_{j=1}^{s} \alpha_j g(t_n + c_j h_n)$$

with evaluation instants

$$t_n \le t_n + c_1 h_n < \cdots < t_n + c_j h_n < \cdots < t_n + c_s h_n \le t_{n+1}.$$

This requires

$$0 \le c_1 < \cdots < c_j < \cdots < c_s \le 1.$$

When $\sum_j \alpha_j = 1$ we obtain the following approximation of the IE (2.9)

$$x(t_{n+1}) \approx x(t_n) + h_n \sum_{j=1}^{s} \alpha_j f\left(t_n + c_j h_n, x(t_n + c_j h_n)\right). \tag{2.10}$$

To derive a one step scheme we have to replace the term $x(t_n + c_j h_n)$ in (2.10) by

$$x(t_n + c_j h_n) = x(t_n) + \int_{t_n}^{t_n + c_j h_n} f(t, x(t))\,\mathrm{d}t, \qquad j = 1, \ldots, s, \tag{2.11}$$

which contains $x(t_n)$ and $x(t_{n+1})$ only.

Example 2.2 By the midpoint rectangle rule using the evaluation point $t_n + \frac{1}{2} h_n$,

$$\begin{aligned} x(t_{n+1}) &= x(t_n) + \int_{t_n}^{t_{n+1}} f(t, x(t))\,\mathrm{d}t \\ &\approx x(t_n) + (t_{n+1} - t_n)\, f\left(t_n + \frac{1}{2} h_n, x\left(t_n + \frac{1}{2} h_n\right)\right). \end{aligned}$$

We then approximate $x\left(t_n + \frac{1}{2} h_n\right)$ by the explicit Euler scheme on the subinterval $\left[t_n, t_n + \frac{1}{2} h_n\right]$,

$$x\left(t_n + \frac{1}{2} h_n\right) \approx x(t_n) + \frac{1}{2} h_n\, f(t_n, x(t_n)),$$

to obtain an expression which contains only $x(t_n)$ and $x(t_{n+1})$, i.e.,

$$x(t_{n+1}) \approx x(t_n) + h_n\, f\left(t_n + \frac{1}{2} h_n, x(t_n) + \frac{1}{2} h_n\, f(t_n, x(t_n))\right). \tag{2.12}$$

Formula (2.12) gives the *improved Euler scheme*

$$x_{n+1} = x_n + h_n\, f\left(t_n + \frac{1}{2} h_n, x_n + \frac{1}{2} h_n\, f(t_n, x_n)\right).$$

2.2.2.4 The General Form of Runge–Kutta Schemes

The summed integration formula (2.10) with (2.11) becomes quite complicated for large s, i.e., large number of evaluation points. Thus it is more convenient to do the intermediate evaluations (2.11) separately. For a scheme with s evaluation points, termed as *s stages*, we can write Runge–Kutta schemes in a compact form.

(i) *The explicit Euler scheme* ($s = 1$ with one evaluation point t_n):

$$x_{n+1} = x_n + h_n k_1^{(n)}, \qquad k_1^{(n)} = f(t_n, x_n).$$

(ii) *The improved Euler scheme* ($s = 2$ with two evaluation points t_n and $t_n + \frac{1}{2}h_n$):

$$x_{n+1} \;=\; x_n + h_n k_2^{(n)}, \qquad \begin{cases} k_1^{(n)} = f(t_n, x_n), \\ k_2^{(n)} = f(t_n + \frac{1}{2}\, h_n,\; x_n + \frac{1}{2}\, h_n k_1^{(n)}). \end{cases}$$

(iii) *The Heun scheme* ($s = 2$ with two evaluation points t_n and $t_n + h_n$):

$$x_{n+1} \;=\; x_n + \frac{1}{2}\, h_n k_1^{(n)} + \frac{1}{2}\, h_n k_2^{(n)}, \qquad \begin{cases} k_1^{(n)} = f(t_n, x_n), \\ k_2^{(n)} = f(t_n + h_n, x_n + h_n k_1^{(n)}). \end{cases}$$

We can also rewrite implicit schemes in this way.

(iv) *The implicit Euler scheme* ($s = 1$ with one evaluation point $t_n + h_n$):

$$x_{n+1} = x_n + h_n k_1^{(n)}, \qquad k_1^{(n)} = f(t_n + h_n, x_n + h_n k_1^{(n)}).$$

(v) *The Trapezoidal scheme* ($s = 2$ with two evaluation points t_n and $t_n + h_n$):

$$x_{n+1} = \; x_n + \frac{1}{2}\, h_n k_1^{(n)} + \frac{1}{2}\, h_n k_2^{(n)}, \quad \begin{cases} k_1^{(n)} = f(t_n, x_n) \\ k_2^{(n)} = f(t_n + h_n, x_n + \frac{1}{2}\, h_n k_1^{(n)} + \frac{1}{2}\, h_n k_2^{(n)}). \end{cases}$$

The examples above motivate the general form of Runge–Kutta schemes.

Definition 2.1 The general form of a *Runge–Kutta scheme with s stages* is

$$\begin{aligned} x_{n+1} &= x_n + h_n \sum_{i=1}^{s} b_i k_i^{(n)} \\ k_i^{(n)} &= f\left(t_n + c_i h_n, x_n + h_n \sum_{j=1}^{s} a_{i,j} k_j^{(n)}\right), \qquad i = 1, \ldots, s, \end{aligned}$$

where $0 \le c_1 < c_2 < \cdots < c_s \le 1$.

Such a scheme is uniquely determined by the column vector $\boldsymbol{c} = (c_1, \ldots, c_s)^\mathsf{T}$, the row vector $\boldsymbol{b} = (b_1, \ldots, b_s)$, and the $s \times s$ matrix $A = [a_{i,j}]$, that form the *Butcher Tableau*

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

Example 2.3 The Butcher tableaux for the explicit, implicit, and improved Euler scheme are, respectively,

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

The Butcher tableaux for the Heun and Trapezoidal scheme are, respectively,

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

2.3 Orders of Local and Global Convergence

Recall that the local discretisation error

$$\mathscr{E}_{n+1}^L = \|x(t_{n+1}; t_n, x_n) - x(t_n) - hf(t_n, x(t_n))\|$$

can be easily estimated through a Taylor expansion. But ultimately we need an estimate for the global discretisation error

$$\mathscr{E}_{n+1} := \|x(t_{n+1}; t_0, x_0) - x_{n+1}\|.$$

A general property of all "good" one step schemes is that the order of the global discretisation error is one power less than that of the local discretisation error. This is related to properties of the increment function F as stated in the following theorem.

Theorem 2.1 *Suppose that a one step scheme*

$$x_{n+1} = x_n + h\, F(t_n, x_n; h)$$

has local discretisation error of order $(p + 1)$ *and that the increment function* F *satisfies a Lipschitz condition in all three variables* (t, x, h). *Then the global discretisation error has order* p, *i.e.,* $\mathscr{E}_n \sim \mathscr{O}(h^p)$.

Proof The proof can be done by deriving a difference inequality. First we write the global discretisation error as

$$\begin{aligned}\mathscr{E}_{n+1} &= \|x(t_{n+1};t_0,x_0) - x_{n+1}\| \\ &= \|[x(t_n;t_0,x_0) - x_n] + h\,[F(t_n,x(t_n;t_0,x_0);h) - F(t_n,x_n;h)] \\ &\qquad + \left[x(t_{n+1};t_0,x_0) - x(t_n;t_0,x_0) - h\,F(t_n,x(t_n;t_0,x_0);h)\right]\| .\end{aligned}$$

Then by using the triangle inequality we obtain

$$\begin{aligned}\mathscr{E}_{n+1} \le \|x(t_n;t_0,x_0) - x_n\| + h\|F(t_n,x(t_n;t_0,x_0);h) - F(t_n,x_n;h)\| \\ + \|x(t_{n+1};t_0,x_0) - x(t_n;t_0,x_0) - h\,F(t_n,x(t_n;t_0,x_0);h)\|. \quad (2.13)\end{aligned}$$

The last term on the right hand side of (2.13) is exactly the local discretisation error of the one step scheme and by Remark 2.1,

$$\|x(t_{n+1};t_0,x_0) - x(t_n;t_0,x_0) - h\,F(t_n,x(t_n;t_0,x_0);h)\| \le \tilde{M}_p \cdot h^{p+1}, \quad (2.14)$$

e.g., with $\tilde{M}_p = \frac{1}{(p+1)!}M_p$ and M_p is as define in (2.7) for a Taylor scheme.

With $\mathscr{E}_n = \|x(t_n;t_0,x_0) - x_n\|$ and $\mathscr{E}_{n+1} = \|x(t_{n+1};t_0,x_0) - x_{n+1}\|$, from (2.13) and (2.14) we obtain

$$\begin{aligned}\mathscr{E}_{n+1} &\le \mathscr{E}_n + h\|F(t_n,x(t_n;t_0,x_0);h) - F(t_n,x_n;h)\| + \tilde{M}_p h^{p+1} \\ &\le \mathscr{E}_n + hL\|x(t_n;t_0,x_0) - x_0\| + \tilde{M}_p h^{p+1} = (1+Lh)\mathscr{E}_n + \tilde{M}_p h^{p+1},\end{aligned}$$

due to the Lipschitz condition on F where L is the Lipschitz constant.

Then by mathematical induction and $\mathscr{E}_0 = 0$, we can show that

$$\begin{aligned}\mathscr{E}_n &\le (1+Lh)^n\mathscr{E}_0 + \tilde{M}_p h^{p+1}\left(1 + (1+Lh) + \cdots + (1+Lh)^{n-1}\right) \\ &= \frac{(1+Lh)^n - 1}{(1+Lh) - 1} \cdot \tilde{M}_p h^{p+1} = \left((1+Lh)^n - 1\right)\frac{\tilde{M}_p}{L}h^p \\ &\le (e^{L(T-t_0)} - 1) \cdot \frac{\tilde{M}_p}{L}h^p := C_{T,p}h^p,\end{aligned}$$

which implies that the global discretisation error is $\mathscr{O}(h^p)$ and thus has order p. □

Example 2.4 The Heun scheme has local convergence order 3, i.e., $p+1 = 3$, so $p = 2$. Thus the order of its global covergence is $p = 2$.

Example 2.5 The Taylor scheme of order p has local convergence order of $p+1$ and global convergence order of p.

Note that in the proof of Theorem 2.1, we only used the Lipschitz condition for $F(t,x;h)$ in x and uniform continuity in t and h. The assumed Lipschitz condition

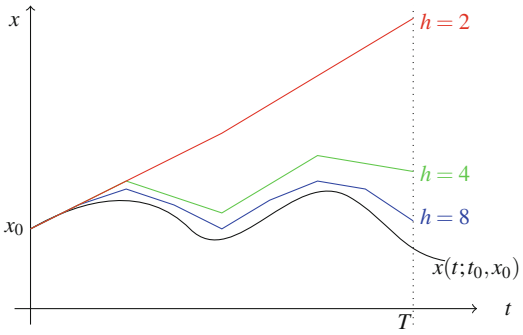

Fig. 2.3 Convergence on a finite time interval

follows from the smoothness of the vector field function $f(t,x)$ of the differential equation. For example for the pth order Taylor scheme the increment function

$$F(t,x,h) = \sum_{i=0}^{p-1} \frac{h^i}{(i+1)!} D^i f(t,x)$$

has total derivative of highest order $D^{p-1}f$, so f should be at least p-times continuously differentiable to obtain the global convergence order of p.

The global discretisation error $\mathscr{E}_n$ obviously depends on the step size h. We will write $\mathscr{E}_n(h)$ to emphasise this. According to Theorem 2.1, when $h \to 0$ we have the convergence

$$\lim_{h\to 0^+} \max_{0\le n\le N_h} \mathscr{E}_n(h) = 0.$$

In particular, the piecewise straight line curve joining the numerical iterates converges to the desired solution curve as $h \to 0$ (see Fig. 2.3). However, notice that the constant $C_{T,p}$ depends on the length of the time interval as well as the properties of the vector field function f. In fact, for every fixed p, we have $C_{T,p} \sim \mathscr{O}(e^T)$, which means that the error estimate is useful for a finite time T, but not for asymptotic behaviour, i.e., as $T \to \infty$.

2.4 Consistency

The derivation of many one step schemes is often heuristic, so we must ensure that they are compatible with the original differential equation. Naturally, we could prove directly that a numerical scheme is convergent, which may be a lot of work. The concept of *consistency* gives us a simple tool to see immediately if a scheme will converge, without having to prove this directly.

Let (t,x) be fixed. The solution $x(\cdot;t,x)$ of the ODE $\frac{\mathrm{d}x}{\mathrm{d}t} = f(t,x)$ satisfies the equation

$$x(t+h;t,x) = x + \int_t^{t+h} f(s, x(s;t,x))\,\mathrm{d}s.$$

A single iteration of the one step scheme with the starting point (t, x) and increment function F satisfies the equation

$$x(h) = x + h\,F(t, x; h). \tag{2.15}$$

Therefore the corresponding local discretisation error is given by

$$\mathscr{E}^{(h)} := \|x(t+h;t,x) - x(h)\| = h \left\| \frac{1}{h} \int_t^{t+h} f(s, x(s;t,x))\,\mathrm{d}s - F(t, x; h) \right\|.$$

From Theorem 2.1 we know that the order of the global discretisation error is always one power less than that of the local discretisation error. Thus to ensure global convergence we need

$$\lim_{h\to 0^+} \left\| \frac{1}{h} \int_t^{t+h} f(s, x(s;t,x))\,\mathrm{d}s - F(t, x; h) \right\| = 0$$

or, equivalently,

$$\lim_{h\to 0^+} F(t, x; h) = \lim_{h\to 0^+} \frac{1}{h} \int_t^{t+h} f(s, x(s;t,x))\,\mathrm{d}s = f(t, x).$$

In general, when F is at least continuous in all variables, the condition $F(t, x; 0) = f(t, x)$ for all (t, x) is necessary for the convergence of the one step scheme. Moreover, it is also a sufficient condition for the convergence of the one step scheme.

Definition 2.2 (*Consistency*) A one step scheme with increment function F is said to be consistent when

$$\lim_{h\to 0^+} F(t, x; h) = f(t, x), \qquad \forall\,(t, x).$$

Example 2.6 The increment function of the family of Runge–Kutta schemes with 2 stages is $F(t, x; h) = \alpha f(t, x) + \beta f\,(t + \gamma h, x + \gamma h\, f(t, x))$ with

$$\lim_{h\to 0^+} F(t, x; h) = (\alpha + \beta) f(t, x).$$

Hence such schemes are consistent if and only if $\alpha + \beta = 1$.

Theorem 2.2 *Suppose that the increment function F satisfies a Lipschitz condition in all three variables (t, x, h). Then the one step scheme (2.15) is convergent if and only if it is consistent.*

2.5 Numerical Instability

The global discretisation error of consistent one step numerical schemes suggests that the numerical solution will be a good approximation of the ODE solution provided the step size is small enough. The computer number field is, however, only finite. In particular, there exists an $\varepsilon_0 > 0$ (the machine epsilon) such that $\|x - y\| \geq \varepsilon_0$ for all $x \neq y$ in this computer number field. Hence the step size h cannot be taken too small. This may be problematic for *stiff* ODEs and lead to numerical instabilities.

Example 2.7 Consider the initial value problem

$$\frac{\mathrm{d}x}{\mathrm{d}t} = -10^N x, \quad x(0) = x_0, \tag{2.16}$$

which has the unique solution $x(t) = e^{-10^N t}\, x_0$ that decreases very rapidly and monotonically to $x = 0$ as $t \to \infty$ for $N \gg 1$.

The Euler scheme for the ODE (2.16) reads

$$x_{n+1} = x_n + h\left(-10^N x_n\right) = \left(1 - h10^N\right) x_n \tag{2.17}$$

which has the explicit solution

$$x_n = \left(1 - h10^N\right)^n x_0, \qquad n = 0, 1, 2, \ldots$$

Recall that $x_n = a^n x_0$ decreases monotonically to 0 if and only if $0 < a < 1$. Here we have $a = 1 - h\,10^N$ so $0 < 1 - h\,10^N < 1$ requires that the step size h should be smaller than 10^{-N}, i.e., the scheme (2.17) is only stable when $h < 10^{-N}$.

However, if $N \gg 1$, then $10^{-N} < \varepsilon_0$, the machine epsilon. Thus a step size $h > 10^{-N}$ must be used, which implies that $a = 1 - h\,10^N < 0$. If $-1 < a < 0$, or $10^{-N} < h < 2 \cdot 10^{-N}$, the numerical iterations still converge towards 0, but oscillating with alternating sign. This is obviously unrealistic. The situation is even worse for $h \geq 2 \cdot 10^{-N}$. Then there are increasing oscillations (see Fig. 2.4). For example, for $h = 100 \cdot 10^{-N}$, so $a = -99$, then $x_n = (-99)^n x_0$.

Example 2.8 Consider a 2-dimensional linear system

$$\frac{\mathrm{d}x}{\mathrm{d}t} = -10^N x, \quad \frac{\mathrm{d}y}{\mathrm{d}t} = x - y$$

or

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} x \\ y \end{pmatrix} = A \begin{pmatrix} x \\ y \end{pmatrix} \quad \text{with} \quad A = \begin{bmatrix} -10^N & 0 \\ 1 & -1 \end{bmatrix}.$$

The matrix A has eigenvalues $\lambda_1 = -10^N$, $\lambda_2 = -1$ with corresponding eigenvectors

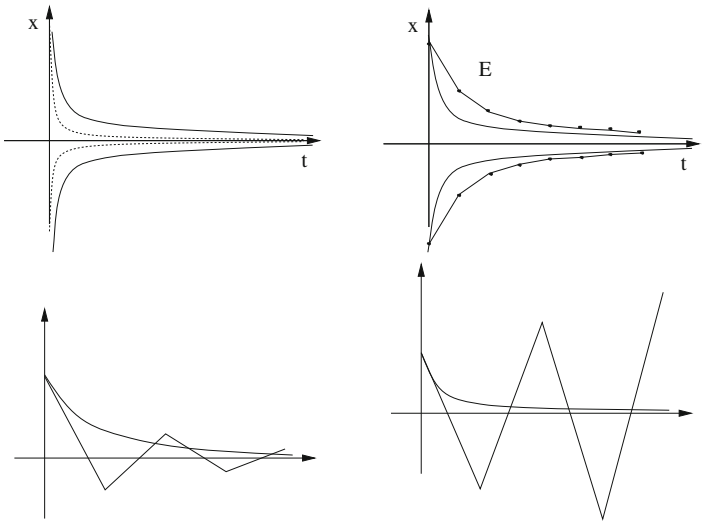

Fig. 2.4 Numerical instability: the same numerical scheme which is stable at h small becomes unstable when h increases

$$v_1 = \begin{pmatrix} 1 + 10^N \\ 1 \end{pmatrix}, \qquad v_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Then the general solution is

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = a \begin{pmatrix} 1 + 10^N \\ 1 \end{pmatrix} e^{-10^N t} + b \begin{pmatrix} 0 \\ 1 \end{pmatrix} e^{-t},$$

where a and b are arbitrary constants. For the initial value $(x(0), y(0)) = (0, 1)$ the solution is $(x(t), y(t)) = (0, e^{-t})$.

The corresponding Euler scheme is

$$x_{n+1} = x_n - h\, 10^N x_n, \quad y_{n+1} = y_n + h\, x_n - h y_n. \tag{2.18}$$

For $x_0 = 0$ we see that $x_n \equiv 0$, so

$$y_{n+1} = y_n - h y_n = (1 - h) y_n,$$

i.e., $x_n \equiv 0$ and $y_n = (1-h)^n y_0 \to 0$ monotonically as $n \to \infty$, provided $0 < h < 1$.

As an illustrative example, Let $h = 100 \cdot 10^{-N}$. Then the Euler scheme (2.18) becomes

$$x_{n+1} = -99 x_n, \qquad y_{n+1} = 100\, 10^{-N} x_n + \left(1 - 10^{-N+2}\right)\, y_n.$$

So with $x_0 = 0$, $x_n = (-99)^n x_0 \equiv 0$ for all $n \geq 0$ and as a result

$$y_{n+1} = \left(1 - 10^{-N+2}\right)\ y_n \to 0 \ \text{ as } \ n \to \infty.$$

If, however, there is a small error in x_0, so $x_0 \neq 0$, this will seriously affect the behaviour y_n, since now $x_n = (-99)^n x_0 \to \pm\infty$ as $n \to \infty$ and

$$\begin{aligned} y_n \ &= \ \left(1 - 10^{-N+2}\right)^n y_0 + \sum_{j=0}^{n-1} \left(1 - 10^{-N+2}\right)^{n-j-1} 10^{-N+2} (-99)^j x_0 \\ &\to \pm\infty \ \text{ as } \ n \to \infty. \end{aligned}$$

The above situation in Examples 2.7 and 2.8, referred to as *numerical instability*, is somewhat artificial, but the phenomenon can arise within a more complicated context. Implicit schemes are often used to avoid such numerical instabilities. For example, the implicit Euler scheme for the ODE (2.16) reads

$$x_{n+1} = x_n - h10^N x_{n+1},$$

which can be solved explicitly algebraically to give

$$\left(1 + h\ 10^N\right)\ x_{n+1} = x_n \Longrightarrow x_{n+1} = \frac{1}{1 + h\ 10^N}\ x_n.$$

As a result,

$$x_n = \left(\frac{1}{1 + h\ 10^N}\right)^n x_0 \to 0 \quad \text{for} \quad n \to \infty$$

for any $h > 0$. In this case step sizes $h \gg 10^{-N}$ can be used without affecting the behaviour of the numerical iterates.

Remark 2.3 In general an implicit scheme gives an algebraic equation at each step which may only be solved numerically. This requires additional work for every time step, but as a trade-off a much larger time step can be used, so the total amount of work needed could be much less.

2.6 Steady States of Numerical Schemes

Consider an autonomous ODE in $\mathbb{R}^d$

$$\frac{\mathrm{d}x}{\mathrm{d}t} = f(x) \tag{2.19}$$

and a consistent one step scheme with constant step size,

$$x_{n+1} = x_n + h\,F(h, x_n), \tag{2.20}$$

i.e., with $F(0, x) \equiv f(x)$, for all x.

Let x^* be a steady state of the ODE (2.19), i.e., $f(x^*) = 0$.

Example 2.9 Consider the Euler scheme

$$x_{n+1} = x_n + hf(x_n). \tag{2.21}$$

Since $f(x^*) = 0$, then $x_n \equiv x^*$ for all n and all $h > 0$ if $x_0 = x^*$, i.e., $x_h^* = x^*$ is also a steady state of the Euler scheme (2.21) for all step sizes $h > 0$.

Example 2.10 Consider the Heun scheme

$$x_{n+1} = x_n + hF(h, x) \quad \text{with} \quad F(h, x) = \frac{1}{2}\left(f(x) + f(x + hf(x))\right). \tag{2.22}$$

Then

$$F(h, x^*) = \frac{1}{2}\left(f(x^*) + f(x^* + hf(x^*))\right) = \frac{1}{2}\left(0 + f(x^* + 0)\right) = 0,$$

which implies that $x_h^* \equiv x^*$ is also a steady state of the Heun scheme (2.22) for all step sizes $h > 0$.

The proof of the following theorem is left to the reader.

Theorem 2.3 *Let x^* be a steady state of an autonomous ODE. Then $x_h^* \equiv x^*$ is a steady state of the corresponding Taylor and Runge–Kutta schemes for all $h > 0$ (possibly sufficiently small).*

In general, however, a steady state x^* of the ODE (2.19) needs not be a steady state of its numerical scheme (2.20), i.e., $F(h, x^*) = 0$ does not always hold for all (small enough) h even if $f(x^*) = 0$. This is illustrated by the following examples.

Example 2.11 Let $F(h, x) = f(x) + h$ in $\mathbb{R}^1$ and consider the numerical scheme

$$x_{n+1} = x_n + h(f(x_n) + h). \tag{2.23}$$

This increment function F is somewhat artificial, but is nevertheless consistent as $F(0, x) \equiv f(x)$ for all x. Let x^* be a steady state of the ODE (2.19), i.e., $f(x^*) = 0$. Then

$$F(h, x^*) = f(x^*) + h = 0 + h = h \neq 0, \quad \forall h > 0,$$

so x^* is not a steady state of this numerical scheme for any step size $h > 0$.

Does the numerical scheme (2.23) have any steady state nearby x^*? The answer is "not always". For example taking $f(x) = x^2$ in (2.23), then $F(h, x) = f(x) + h =$

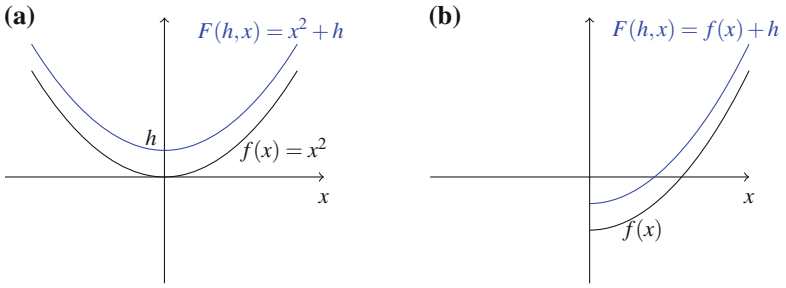

Fig. 2.5 $F(h, x) = 0$ has no solution when $f(x) = x^2$. It has a solution only when f crosses the $x-$axis

$x^2 + h$. But $x^2 + h \neq 0$ for all x and $h > 0$ (see Fig. 2.5a), and thus there is no steady state for the numerical scheme (2.23). In fact, for $F(h, x) = 0$ to have a solution, the f curve should cross the x-axis (see Fig. 2.5b). This holds if $f'(x^*) \neq 0$, i.e., if x^* is a hyperbolic steady state. Then by the Implicit Function Theorem, we know that the equation

$$F(h, x) = f(x) + h = 0$$

has a solution x_h^* in a neighbourhood of x^*, provided $h > 0$ is sufficiently small.

Example 2.12 Let $f(x) = ax$ with $a \neq 0$ in (2.19). Then $x^* = 0$ is the only steady state and it is hyperbolic since

$$f'(x^*) = a \neq 0$$

The equation $F(h, x) = ax + h = 0$ has the unique solution $x_h^* = -h/a$ which gives the numerical steady state. The approximation error of the steady state is

$$\left\| x_h^* - x^* \right\| = \frac{1}{|a|}\ h = \mathcal{O}(h),$$

which has the same order ($p = 1$ here) as the numerical scheme.

Remark 2.4 A general property of hyperbolic steady states is that a numerical steady state x_h^* exists with

$$\left\| x_h^* - x^* \right\| = \mathcal{O}(h^p),$$

where p is the order of convergence of the numerical scheme.

But a numerical scheme can also have other steady states which have no connection with the steady states of the corresponding ODE (see the example below).

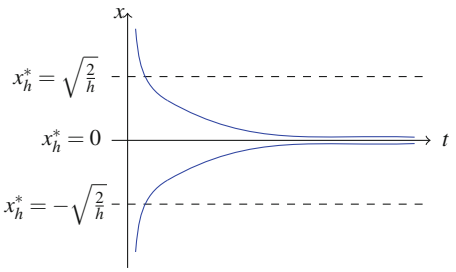

Fig. 2.6 The spurious solutions are irrelevant to true solutions of an ODE

Example 2.13 Let $f(x) = -x^3$ in $\mathbb{R}^1$. Then the only steady state for (2.19) is $x^* = 0$. Consider the Heun scheme:

$$x_{n+1} = x_n + \frac{h}{2}\left(f(x_n) + f(x_n + hf(x_n))\right),$$

where, in this case,

$$F(h,x) = \frac{1}{2}\left(f(x) + f(x + hf(x))\right) = -\frac{1}{2}x^3\left(1 + (1 - hx^2)^3\right).$$

Here the equation $F(h,x) = 0$ has solutions

$$x_h^* = 0, \quad x_h^* = \pm\sqrt{\frac{2}{h}}, \quad \forall h > 0.$$

The steady states $\pm\sqrt{\frac{2}{h}}$ have nothing to do with the ODE (see Fig. 2.6). They are called *spurious* or *ghost* solutions.

The above example is typical. Not only do we have $x_h^* \neq x^*$, but also

1. the spurious solutions diverge:

$$\left|x_h^*\right| = \sqrt{\frac{2}{h}} \to \infty \quad \text{for} \quad h \to 0^+.$$

2. The spurious solutions are *unstable*. To show this a linear stability analysis is carried out below. Linearising the scheme

$$x_{n+1} = x_n - \frac{h}{2}x_n^3\left\{1 + \left(1 - hx_n^2\right)^3\right\} := g(x_n)$$

about the steady state x_h^* results in

$$z_{n+1} = g'(x_h^*)\, z_n,$$

where

$$g'(x) = 1 - \frac{h}{2}\,3x^2\,\left\{1 + \left(1 - hx^2\right)^3\right\} + \frac{h}{2}\,x^3\,6hx\,\left(1 - hx^2\right)^2$$
$$= 1 - \frac{3h}{2}\,x^2\,\left\{1 + \left(1 - hx^2\right)^3\right\} + 3x^4h^2\left(1 - hx^2\right)^2.$$

Thus

$$g'\left(\pm\sqrt{\frac{2}{h}}\right) = 1 - \frac{3h}{2}\,\frac{2}{h}\,\left\{1 + 0^3\right\} + 3h^2\frac{4}{h^2} = 13$$

and we obtain the linearised system

$$z_{n+1} = 13z_n\,,$$

for which $z^* = 0$ is *unstable*.

3. The true steady state is *asymptotically stable*. The linearised system (about $x^* = 0$) is

$$z_{n+1} = g'(0)z_n \equiv z_n$$

because $g'(0) = 1$ here. It is actually only *neutrally stable* so nonlinear terms also need to be taken into account. To this end, define a Lyapunov function $V(x) := x^2$, for which

$$V(x_{n+1}) = x_{n+1}^2 = g^2(x_n) = x_n^2\,\left[1 - \frac{h}{2}\,x_n^2\left(1 + \left(1 - hx_n^2\right)^3\right)\right]^2 < x_n^2,$$

provided $x_n^2 < 2/h$.

For $x_0^2 < 2/h$ we have $2/h > x_0^2 > x_1^2 > \cdots$. Then $V(x_n)$ is strongly monotonically decreasing with $V(x_n) = x_n^2 \to 0$, which implies that $x^* = 0$ is *asymptotically stable* with the *basin of attraction*

$$B_h = \left\{x \in \mathbb{R}^1 :\, x^2 < \frac{2}{h}\right\}\,.$$

It obviously depends on the step size h.

Remark 2.5 The steady state $x^* = 0$ is globally asymptotically stable for the ODE in Example 2.13, but is only locally asymptotically stable for the Heun scheme with the bounded basin on attraction B_h that depends on the step size.

http://www.springer.com/978-3-319-61933-0

Attractors Under Discretisation

Han, X.; Kloeden, P.
2017, XI, 122 p. 23 illus., 14 illus. in color., Softcover
ISBN: 978-3-319-61933-0