9. Initial Value Problems

A first order scalar *initial value problem* (IVP) has the general form:

$$\frac{dx}{dt} = f(t,x), \quad x(0) = x_0. \tag{39}$$

The unknown is the function $x = x(t)$. If we think of the independent variable t as time, we can interpret the ordinary differential equation (ODE) in (39) as an *evolution law*: the rate of change of x is a function of time and x itself. Under appropriate conditions, which will be clarified later, the initial value problem (39) has a unique solution. Our main concern is the numerical approximation of that solution on a prescribed time interval $[0, T]$.

We assume that the reader has already studied differential equations, ideally, in a specialized course. Still it may be helpful to review the basic components of ODE theory before delving into numerics.

9.1. **ODE and Calculus.** Conceptually, solving an ordinary differential equation of any kind can be thought of as reconstruction of a function from information about its derivative, which is to say, integration. In the simplest case, we may be given the derivative explicitly as a function of time:

$$\frac{dx}{dt} = f(t), \quad x(0) = x_0. \tag{40}$$

The solution is then the antiderivative (of f) that matches the initial condition:

$$x(t) = x_0 + \int_0^t f(s)\, ds. \tag{41}$$

Equation (41) contains a definite integral and for this reason is called *the solution in quadratures*. Note that the very term implies that there is a direct connection between quadrature and solving IVP's. Indeed, as we will see later, most of the quadrature methods that we now have at our disposal can be converted into *ODE solvers*. However, the conversion is not straightforward and requires some deep ideas which make the subject even more fascinating.

There are several classes of ODE which admit solutions in quadratures. Chief among them are *separable* equations, of which (40) is an instance. A separable IVP can be written in the general form:

$$\frac{dx}{dt} = \frac{f(t)}{g(x)}, \quad x(0) = x_0. \tag{42}$$

The solution is by separating variables—hence the name:

$$\int_{x_0}^{x} g(y)\,dy = \int_0^t f(s)\,ds. \tag{43}$$

Generally speaking, if an IVP is *not* separable, it is unlikely that it can be solved in quadratures. In fact, this is the primary reason why we are interested in numerical approaches in the first place. There is, however, one notable exception. The linear first order IVP

$$\frac{dx}{dt} = a(t)\,x + b(t), \quad x(0) = x_0, \tag{44}$$

can be solved in quadratures as follows:

$$x(t) = x_0\,e^{\int_0^t a(s)\,ds} + \int_0^t e^{\int_s^t a(u)\,du}\,b(s)\,ds. \tag{45}$$

In particular, if $a(t)$ is a constant function, say, $a(t) = \lambda$, then:

$$x(t) = x_0\,e^{\lambda\,t} + \int_0^t e^{\lambda\,(t-s)}\,b(s)\,ds. \tag{46}$$

An integral of the form

$$\int_0^t g(t-s)\,b(s)\,ds,$$

viewed as an operation on functions, is called *convolution* and abbreviated as $g \star b$. Any first order, linear, nonhomogeneous IVP with constant coefficients

$$\frac{dx}{dt} = \lambda\,x + f(t), \quad x(0) = x_0 \tag{47}$$

can be solved symbolically, using convolution:

$$x(t) = x_0\,e^{\lambda\,t} + e^{\lambda\,t} \star f. \tag{48}$$

Such IVP's arise in numerous applications and are of great practical importance. Furthermore, many nonlinear IVP's can be approximated with linear equations. For this reason, we will often use linear equations to test numerical methods. In fact, our general view will be that if a method does not do well with linear equations, it is not very useful.

9.2. **ODE and linear algebra.** In linear circuits and other applications one encounters (large) systems of linear equations with constant coefficients. These can be written in the form:

$$\frac{d\mathbf{x}}{dt} = A\,\mathbf{x} + \mathbf{f}(t), \quad \mathbf{x}(0) = \mathbf{x}_0. \tag{49}$$

Here $\mathbf{x}, \mathbf{f} : \mathbb{R} \to \mathbb{R}^N$ and A is a constant N-by-N matrix. If we define exponential of a matrix by

$$e^A = I + A + \frac{1}{2}A^2 + \frac{1}{6}A^3 + \ldots = \sum_{n=0}^{\infty} \frac{A^n}{n!} \tag{50}$$

we can write the solution of Equation (49) in the form analogous to (48):

$$\mathbf{x} = e^{t\,A}\,\mathbf{x}_0 + e^{t\,A} \star \mathbf{f}. \tag{51}$$

The vector valued convolution is defined by:

$$e^{t\,A} \star \mathbf{f} = \int_0^t e^{(t-s)\,A}\,\mathbf{f}(s)\,ds.$$

Notice that inside the integral is a matrix-vector product which produces a vector-valued function; that function is then integrated componentwise.

In practice, the matrix exponential is computed using the eigenvalue decomposition. Let

$$\Lambda = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_N \end{bmatrix}$$

be the diagonal matrix with eigenvalues of A on the main diagonal. Also, let

$$V = [\mathbf{v}_1 \cdots \mathbf{v}_N]$$

be the matrix whose columns are the corresponding eigenvectors. For simplicity, assume that the eigenvectors are linearly independent. Then $A = V\,\Lambda\,V^{-1}$ and, consequently, $e^{t\,A} = V\,e^{t\,\Lambda}\,V^{-1}$. If we substitute $\mathbf{x} = V\,\mathbf{u}$ and $\mathbf{f} = V\,\mathbf{g}$ in (51), we obtain

$$\mathbf{u} = e^{t\,\Lambda}\,\mathbf{u}_0 + e^{t\,\Lambda} \star \mathbf{g},$$

which is a system of N scalar equations of the form:

$$u_n(t) = e^{\lambda_n\,t}\,u_n(0) + e^{\lambda_n\,t} \star g_n.$$

Notice that these equations are the same in appearance as (48). The important conclusion that can be drawn here is that solving linear systems, conceptually, is not any more difficult than solving just one scalar linear IVP: if a numerical method can solve the scalar equation (47), it can be extended to (49).

Here we would also like to remind the reader that higher order IVP (both linear and nonlinear) can be converted to systems of first order

equations. For instance, the second order IVP, describing a forced mass-spring system

$$\frac{d^2x}{dt^2} = -k\,x + f(t), \quad x(0) = x_0, \quad \frac{dx}{dt}(0) = v_0$$

is equivalent to the first order vector system:

$$\frac{d}{dt}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k & 0 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ f(t) \end{bmatrix}, \quad \begin{bmatrix} x(0) \\ y(0) \end{bmatrix} = \begin{bmatrix} x_0 \\ v_0 \end{bmatrix}.$$

Henceforth we only consider first order IVP; higher order IVP will be converted to vector form:

$$\frac{d\mathbf{x}}{dt} = f(t, \mathbf{x}), \quad \mathbf{x}(0) = \mathbf{x}_0, \quad \mathbf{x} : \mathbb{R} \mapsto \mathbb{R}^N. \tag{52}$$

We now return to the task of numerically solving a scalar first order IVP (39).

9.3. **Numerical solution of IVP.** Let $T > 0$ be given and let $\{t_n\}_{n=0}^N$ be an increasing sequence in $[0, T]$ with $t_0 = 0$ and $t_N = T$: we call this sequence the *time grid*. By a numerical solution of IVP (39) on the interval $[0, T]$ we mean the sequence $\{(t_n, y_n)\}_{n=0}^N$ where y_n's are approximations of $x_n = x(t_n)$ for $n = 0, \ldots, N$. For the time being, we will restrict our attention to equispaced time grids: $t_n = h\,n$, where $n = 0, \ldots, N$. We will call the positive number h the *step size*.

All methods for solving IVP (39) that we are going to consider will be recursive in nature: that is, at the n-th step the value y_n will be computed using previously computed approximations y_{n-1}, y_{n-2}, $\ldots$, y_{n-N}. If $N = 1$ we will call the method 1-step; if $N = 2$ then the method is 2-step, and so on.

For any stepping method, we denote the error at the n-th step by $e_n = x_n - y_n$; this is the *local* error of the method. The *global* error is defined by

$$E = \max_{0 \le n \le N} |e_n|. \tag{53}$$

Note that for a given IVP on a given time interval, global error of a numerical method depends only on the step size h.

We say that the method converges if the global error goes to zero as the step size is decreased:

$$\lim_{h \to 0^+} E = \lim_{h \to 0^+} \left(\max_{0 \le n \le N} |e_n| \right) = 0 \quad \Rightarrow \quad \text{convergence.} \tag{54}$$

Needless to say, we are going to be interested only in convergent methods.

9.4. **One step methods.** We commence the discussion with one step methods for solving IVP (39). A number of such methods can be derived as follows. First, integrate both sides of IVP (39) from t_n to t_{n+1}. This leads to:

$$x_{n+1} = x_n + \int_{t_n}^{t_{n+1}} f(t, x(t))\, dt. \tag{55}$$

Now approximate the integral on the right-hand side of (55) using interpolation on the grid. Since the range of integration in (55) involves only two grid points, t_n and t_{n+1}, there are three immediate possibilities.

9.5. **Euler's Method.** Constant interpolation at the left endpoint leads to approximate equality:

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

This suggests the following method, which is named after Euler:

$$y_{n+1} = y_n + f(t_n, y_n)\,h, \quad y_0 = x_0. \tag{56}$$

Notice that Equation (56) closely resembles the equation of the tangent line. In fact, we can derive Euler's method using the tangent line approximation. Indeed, since the right-hand side of the IVP gives the slope of x, we can write:

$$x(t) \approx x(t_n) + x'(t_n)\,(t - t_n) = x_n + f(t_n, x_n)\,(t - t_n).$$

Setting $t = t_{n+1}$ leads to Equation (56). We also note that IVP (39) provides us with a natural starting value—the initial value. In fact, for any scheme we will always set $y_0 = x_0$, for obvious reasons.

9.6. **Backward Euler's method.** Using the right endpoint instead of the left, leads to the scheme:

$$y_{n+1} = y_n + f(t_{n+1}, y_{n+1})\,h, \quad y_0 = x_0. \tag{57}$$

Notice that y_{n+1} occurs on both sides of Equation (57) and must be solved for, either by hand or using some root-finding scheme. Since Equation (57) defines y_{n+1} implicitly, backward Euler's method is said to be *implicit*.

At this point you may wonder: Why bother with backward Euler's method when there is the regular Euler's method which is explicit and does not require root-finding? The short answer is *stability*. Backward Euler's method is much more stable than its explicit counterpart and can generally be used with much larger step size. As we will later see, in many cases, e.g., when solving the so-called *stiff* differential equations,

larger step size is more than an ample compensation for the overhead caused by root-finding.

9.7. **Trapezoid method.** If we interpolate the integrand in (55) using both endpoints, the result is the trapezoid method—the analogue of the trapezoid quadrature rule:

$$y_{n+1} = y_n + \frac{1}{2}\left(f(t_n, y_n) + f(t_{n+1}, y_{n+1})\right) h, \quad y_0 = x_0. \tag{58}$$

Like backward Euler's scheme, the trapezoid method is implicit. Since linear interpolation is more accurate than constant interpolation, we may also surmise that the trapezoid method is more accurate than either of the Euler's methods. As we will later confirm, this is indeed the case.

9.8. **Convergence of Euler's method.** Recall that convergence of stepping methods for solving IVP's is defined by condition (54). When put into words, Equation (54) simply says that the global error must go to zero as the step size is decreased. This, clearly, is a necessary requirement: if a method is not convergent it should not be used. We will now show that under certain conditions Equation (54) holds for Euler's scheme (56). The convergence of other methods can be proved similarly and is relegated to exercises.

Traditionally, any statement of convergence begins with careful enumeration of assumptions. This is necessary for the simple reason that assumptions are typically a big part of the proof. Our goal, however, is not to be as precise as possible but to maximize understanding. We therefore break away from the mathematical tradition and start with an imprecise statement of Theorem 8 below. We will make all of the necessary assumptions in the proof, as we go along; once it becomes clear what assumptions are needed, we will append them to Theorem 8, after the fact.

Theorem 8. *Euler's method converges.*

Proof. Recall that we use $e_n = x_n - y_n$ to denote local error and E to denote global error of a numerical scheme; the relationship between e_n and E is given by Equation (53). We are mainly interested in the global error. In fact, to prove that Euler's method converges, we must show that E goes to zero as the step size h is decreased (c.f., Equation (54)). However, in order to find E we do have to find e_n first. Accordingly, we begin with an investigation of the local error of Euler's method.

Since Euler's method (56) is recursive, it makes sense to seek a recursive relationship for its local error: that is, we would like to express

e_{n+1} in terms of e_n. This calls for a relationship between x_{n+1} and x_n. If we assume that $x \in C^2([0,T])$, as we proceed to do, we can obtain the latter through Taylor expansion (with the remainder term):

$$\begin{aligned} x_{n+1} = x(t_{n+1}) = x(t_n + h) &= x(t_n) + x'(t_n)\, h + \frac{x''(\tau_n)}{2}\, h^2 \\ &= x_n + f(t_n, x_n)\, h + \frac{x''(\tau_n)}{2}\, h^2, \quad \tau_n \in [t_n, t_{n+1}]. \end{aligned} \tag{59}$$

Notice that we used IVP (39) to replace the derivative $x'(t_n)$ with $f(t_n, x_n)$.

We now follow our plan and subtract Equation (56) from (59). This, after some rearrangements, results in a recursive relationship for the local error:

$$e_{n+1} = e_n + h\,(f(t_n, x_n) - f(t_n, y_n)) + \frac{x''(\tau_n)}{2}\, h^2, \quad \tau_n \in [t_n, t_{n+1}]. \tag{60}$$

Unfortunately, Equation (60) also contains x_n and y_n and is therefore not immediately useful. In order to proceed further, we need to relate the term in parentheses to e_n. To this end, assume that $f(t,x)$ is differentiable in the second variable. Then we can use MVT to write:

$$\begin{aligned} f(t_n, x_n) - f(t_n, y_n) &= \frac{\partial f}{\partial x}(t_n, \xi_n)\,(x_n - y_n) \\ &= \frac{\partial f}{\partial x}(t_n, \xi_n)\, e_n, \quad x_n \le \xi_n \le y_n. \end{aligned}$$

Accordingly, Equation (60) becomes

$$e_{n+1} = \left(1 + h\, \frac{\partial f}{\partial x}(t_n, \xi_n)\right)\, e_n + \frac{x''(\tau_n)}{2}\, h^2, \tag{61}$$

with $\tau_n \in [t_n, t_{n+1}]$ and $x_n \le \xi_n \le y_n$; this is the recursion we need.

We now turn our attention to the task of estimating E. Since the latter is defined as the maximum value of $|e_n|$ (where maximum is taken over the range of the subindex), we take the absolute values of both sides of Equation (61). This leads to:

$$\begin{aligned} |e_{n+1}| &= \left|\left(1 + h\, \frac{\partial f}{\partial x}(t_n, \xi_n)\right)\, e_n + \frac{x''(\tau_n)}{2}\, h^2\right| \\ &\le \left|1 + h\, \frac{\partial f}{\partial x}(t_n, \xi_n)\right|\, |e_n| + \frac{|x''(\tau_n)|}{2}\, h^2 \\ &\le \left(1 + h\, \left|\frac{\partial f}{\partial x}(t_n, \xi_n)\right|\right)\, |e_n| + \frac{|x''(\tau_n)|}{2}\, h^2. \end{aligned} \tag{62}$$

Notice that we used the *triangle inequality*

$$|a + b| \le |a| + |b|$$

to distribute the absolute value on the right-hand side. At first glance, it may seem that this forced us to relinquish some of the control since our equation became an inequality. However, the loss of the 'equals' sign is mostly superficial. Remember that our mission is not finding a symbolic formula for E. Rather we need to show that E goes to zero as h goes to zero and this can be done through the use of an inequality. We also remark that the use of the triangle inequality was actually forced: there is simply no other way to place absolute value around e_n on the left-hand side of Equation (61).

The *upper bound* for $|e_{n+1}|$ in terms of $|e_n|$ given by Equation (62) can be simplified further by replacing $|x''(\tau_n)|$ and $\left|\frac{\partial f}{\partial x}(t_n, \xi_n)\right|$ with appropriate estimates. Recall that we assumed $x \in C^2([0,T])$ in order to justify Taylor expansion (59). Therefore the second derivative $x''(t)$ is continuous on $[0,T]$. Consequently, the Extreme Value Theorem (EVT) guarantees that there exists a positive number, say M, such that $|x''(\tau_n)| \leq M$ for all $0 \leq \tau_n \leq T$. Accordingly, we can replace $|x''(\tau_n)|$ in (62) with constant M.

Although we assumed that $\frac{\partial f}{\partial x}$ exists (for all $t \in [0,T]$), we cannot expect it to be bounded. Adding C^1 requirement is insufficient in this case because the variable x is unbounded, so EVT does not apply. We therefore stipulate that there exists a positive number K such that

$$\left|\frac{\partial f}{\partial x}(t,x)\right| \leq K,$$

for all $t \in [0,T]$ and $x \in \mathbb{R}$. This allows us to replace $\left|\frac{\partial f}{\partial x}(t_n, \xi_n)\right|$ in (62) with the constant K. With these replacements, Equation (62) transforms into a simple inequality:

$$|e_{n+1}| \leq (1 + h\,K)\,|e_n| + \frac{M}{2}\,h^2. \tag{63}$$

We will now use (63) to derive a bound for the global error E. First, note that since $e_0 = 0$, Equation (63) immediately implies:

$$|e_1| \leq \frac{M}{2}\,h^2.$$

We can now use the estimate for $|e_1|$ to estimate $|e_2|$ as follows:

$$\begin{aligned} |e_2| &\leq (1 + h\,K)\,|e_1| + \frac{M}{2}\,h^2 \leq (1 + h\,K)\frac{M}{2}\,h^2 + \frac{M}{2}\,h^2 \\ &\leq \frac{M}{2}\,h^2\,\left(1 + (1 + h\,K)\right). \end{aligned}$$

The estimate for $|e_2|$ leads to the estimate of $|e_3|$

$$\begin{aligned} |e_3| &\leq (1 + h\,K)\,|e_2| + \frac{M}{2}\,h^2 \\ &\leq \frac{M}{2}\,h^2\,\left(1 + (1 + h\,K) + (1 + h\,K)^2\right), \end{aligned}$$

and so on. In general,

$$|e_n| \leq \frac{M}{2}\,h^2 \sum_{k=0}^{n-1} (1 + h\,K)^n$$

which, using the *geometric sum* formula,

$$1 + p + p^2 + \ldots + p^{n-1} = \frac{p^n - 1}{p - 1}$$

can be simplified to:

$$|e_n| \leq \frac{M\,h}{2\,K}\,\left((1 + h\,K)^n - 1\right). \tag{64}$$

The right-hand side of Equation (64) is a monotone function of n. Therefore, the global error $E = \max_{0 \leq n \leq N} |e_n|$ is bounded by:

$$E \leq \frac{M\,h}{2\,K}\,\left((1 + h\,K)^N - 1\right). \tag{65}$$

We can now show that, as $h \to 0^+$, the global error E goes to zero by taking the limit of the right-hand side of (65). Since $N = T/h$:

$$\lim_{h \to 0^+} (1 + h\,K)^N = \lim_{h \to 0^+} (1 + h\,K)^{\frac{T}{h}} = e^{K\,T}.$$

Therefore,

$$\begin{aligned} \lim_{h \to 0^+} E &\leq \frac{M}{2\,K} \lim_{h \to 0^+} h\,\left(\lim_{h \to 0^+} (1 + h\,K)^{\frac{T}{h}} - 1\right) \\ &= \frac{M}{2\,K}\,(e^{K\,T} - 1) \lim_{h \to 0^+} h = 0. \end{aligned}$$

This completes the proof. □

In order to establish convergence of Euler's method we were compelled to make the following assumptions:

(1) The solution of IVP (39) must be C^2 on $[0, T]$. This justifies Taylor expansion (59) and also gives us a bound M for use in the Taylor remainder.
(2) The derivative $\frac{\partial f}{\partial x}$ must exist and be bounded on the strip $[0, T] \times \mathbb{R}$. This allows the use of MVT and simplifies the bound for the magnitude of the local error.

It is safe to say that both of these assumptions hold for all IVP's of practical interest (e.g., linear IVP's with constant coefficients), with relatively few exceptions. In principle, we could simply preface the statement of convergence with something like "Assume that $x(t)$ is $C^2([0,T])$ and that $\frac{\partial f}{\partial x}$ is bounded..." However, it behooves us to put a little more thought into refining Theorem 8.

For instance, it would be best if we did not have to make any assumptions about the solution of IVP (39); after all, we approximate it because we do not know it. Can we replace $x \in C^2$ with another condition?

For answers, we turn to the initial value problem itself. Differentiating both sides of (39) with respect to time, we get:

$$\begin{aligned}\frac{d^2x}{dt^2} &= \frac{d}{dt}\left(f(t,x(t))\right) = \frac{\partial f}{\partial t}(t,x) + \frac{\partial f}{\partial x}(t,x)\,\frac{dx}{dt} \\ &= \frac{\partial f}{\partial t}(t,x) + \frac{\partial f}{\partial x}(t,x)\,f(t,x).\end{aligned}$$

Clearly, for this to be legitimate, we must require the existence of both partial derivatives of f. If, in addition to existence, we require that the partial derivatives are continuous, then x'' will be continuous as well. Hence, we can replace $x \in C^2([0,T])$ with $f \in C^1([0,T]\times\mathbb{R})$. As we remarked in the proof of Theorem 8, continuity of $\frac{\partial f}{\partial x}$ does not imply boundedness on the infinite strip $[0,T]\times\mathbb{R}$: this has to be a separate requirement. We conclude that a more precise statement of Theorem 8 is the following:

Theorem 9. *Let $T > 0$ and $S = [0,T]\times\mathbb{R}$. Suppose that Euler's method is applied to IVP (39). If the right-hand side $f(t,x)$ of the IVP has continuous partial derivatives on S, and if further $\frac{\partial f}{\partial x}$ is bounded on S, then Euler's method is convergent.*

As a Math 49 exercise, try to prove Theorem 9 without looking into the proof of Theorem 8. Keep in mind that the proof is in the proverbial pudding: the assumptions listed in the statement are there to allow Taylor expansion, the use of MVT, and so on.

Theorem 9 is neither the most exact nor the most widely used statement of its kind. There other convergence theorems where the assumptions imposed on f are milder. In fact, the most typical assumption is that of *Lipschitz continuity*[11]. For a function of one variable, the

[11]In order to be Lipschitz continuous a function has to be continuous, however, not every continuous function is Lipschitz continuous. For instance, $F(x) = x^2$ is not Lipschitz continuous on $(-\infty,\infty)$. The Lipschitz condition is thus a stronger requirement than continuity, yet it is far less stringent than differentiability.

Lipschitz condition can be stated as

$$|F(x) - F(y)| \le K\,|x - y|, \tag{66}$$

where $K \ge 0$ is a nonnegative Lipschitz constant. It is easy to show (exercise) that assuming the right-hand side of (39) to be Lipschitz continuous obviates the use of MVT and the requirement of boundedness of $\frac{\partial f}{\partial x}$. In fact, one arrives at Equation (47) much quicker if one has access to the Lipschitz constant.

9.9. **Multistep methods.** Any N-step method can be cast in the general form

$$\sum_{m=0}^{N} a_m\, y_{n+m} = h \sum_{m=0}^{N} b_m\, f(t_{n+m}, y_{n+m}), \quad n = 0, 1, 2, \ldots, \tag{67}$$

where the coefficients a_m and b_m are fixed numbers independent of h; the usual convention is to set $a_N = 1$. When $b_N = 0$ the method is explicit; otherwise it is implicit.

A vast number of multistep methods can be derived using interpolatory quadrature. To illustrate that, we turn our attention to two classical families of *Adams* methods.

9.10. **Adams methods.** Integrating IVP (39) from t_{n+N-1} to t_N leads to a straightforward generalization of Equation (55):

$$x_{n+N} = x_{n+N-1} + \int_{t_{n+N-1}}^{t_{n+N}} f(t, x(t))\, dt. \tag{68}$$

The idea of an Adams method is to replace the integrand in (68) with an interpolating polynomial. There are two main variations.

Let p_{N-1} denote the Lagrangian polynomial of order $(N-1)$ with nodes $(t_{n+m}, f(t_{n+m}, y_{n+m}))$, $m = 0, \ldots, N-1$. Setting

$$y_{n+N} = y_{n+N-1} + \int_{t_{n+N-1}}^{t_{n+N}} p_{N-1}(t)\, dt$$

results in an explicit method called N-step *Adams-Bashford* scheme. The following Maple code generates the first four Adams-Bashford methods.

```
for N from 1 to 4 do
     T := [seq((n+m)*h,m=0..N-1)]:
     F := [seq(f[n+m] ,m=0..N-1)]:
     p := interp(T,F,t):
     Q := int(p,t=(n+N-1)*h..(n+N)*h):
     print(y[n+N] = y[n+N-1] + Q);
end do:
```

Notice that for $N = 1$ we get Euler's method.

TABLE 1. Adams-Bashford schemes for $N = 1, 2, 3, 4$.

Adams-Bashford scheme ($f_k = f(t_k, y_k)$)
$y_{n+1} = y_n + h\, f_n$
$y_{n+2} = y_{n+1} + h\, \left(\frac{3}{2}\, f_{n+1} - \frac{1}{2}\, f_n\right)$
$y_{n+3} = y_{n+2} + h\, \left(\frac{23}{12}\, f_{n+2} - \frac{4}{3}\, f_{n+1} + \frac{5}{12}\, f_n\right)$
$y_{n+4} = y_{n+3} + h\, \left(\frac{55}{24}\, f_{n+3} - \frac{59}{24}\, f_{n+2} + \frac{37}{24}\, f_{n+1} - \frac{3}{8}\, f_n\right)$

If instead of p_{N-1} we use p_N, the Lagrangian interpolant of order N with nodes $(t_{n+m}, f(t_{n+m}, y_{n+m}))$, $m = 0, \ldots, N$, the result is the family of implicit N-step *Adams-Moulton* methods which starts with the trapezoid method.

TABLE 2. Adams-Moulton schemes for $N = 1, 2, 3, 4$.

Adams-Moulton scheme ($f_k = f(t_k, y_k)$)
$y_{n+1} = y_n + h\, \left(\frac{1}{2}\, f_{n+1} + \frac{1}{2}\, f_n\right)$
$y_{n+2} = y_{n+1} + h\, \left(\frac{5}{12}\, f_{n+2} + \frac{2}{3}\, f_{n+1} - \frac{1}{12}\, f_n\right)$
$y_{n+3} = y_{n+2} + h\, \left(\frac{3}{8}\, f_{n+3} + \frac{19}{24}\, f_{n+2} - \frac{5}{24}\, f_{n+1} + \frac{1}{24}\, f_n\right)$
$y_{n+4} = y_{n+3} + h\, \left(\frac{251}{720}\, f_{n+4} + \frac{323}{360}\, f_{n+3} - \frac{11}{30}\, f_{n+2} + \frac{53}{360}\, f_{n+1} - \frac{19}{720}\, f_n\right)$

9.11. **The order of accuracy.** The order of accuracy of ODE solvers can be defined in the same manner as for quadrature rules:

> A method has order k if it recovers exactly all polynomial solutions of order k or less.

While this definition has the advantage of simplicity, it is not convenient in practice. For multistep methods one typically uses an equivalent definition based on Taylor expansions, which we now present.

Suppose we are given an N-step method (67). Think of the terms y_{n+m} as approximations of $x(t + m\, h)$. Likewise, consider the terms $f(t_{n+m}, y_{n+m})$ as approximations of $x'(t + m\, h)$. We then have the following approximate equality:

$$\sum_{m=0}^{N} a_m\, x(t + m\, h) \approx h \sum_{m=0}^{N} b_m\, x'(t + m\, h).$$

The better the approximation, the more accurate the method. To quantify the accuracy, form the difference of the two sides and expand it into a Taylor series centered at $h = 0$. The order of the first nonzero term in that series is one more than the order of the method. In other words, we say that the method is of order k if:

$$\sum_{m=0}^{N} a_m \, x(t + m\,h) - h \sum_{m=0}^{N} b_m \, x'(t + m\,h) = \mathrm{O}\left(h^{k+1}\right). \tag{69}$$

As a simple example, consider Euler's method (56). Expanding

$$x(t + h) - x(t) - h\,x'(t)$$

into a Taylor series at $h = 0$ gives:

$$\frac{1}{2}\,x''(t)\,h^2 + \frac{1}{6}\,x'''(t)\,h^3 + \ldots$$

The first nonzero term in the series has order 2. Hence, Euler's method is of first order.

As another example, consider trapezoid method (58). Since

$$x(t + h) - x(t) - \frac{h}{2}\left(x'(t) + x'(t + h)\right) = -\frac{1}{12}\,x'''(t)\,h^3 + \ldots$$

the order of the method is two. As we suspected, the trapezoid method is more accurate than Euler's.

It may seem strange that the order of the method is one less than the order of the first nonzero term in the Taylor series. Yet there is a simple explanation. The first nonzero term in the Taylor expansion tells us how *local* errors depend on the step size h: for instance, for Euler's scheme the local error is quadratic in h. However, the convergence of the method is defined in terms of the *global* error. Making N steps of size h with local error $\mathrm{O}\left(h^{k+1}\right)$ will generally result in global error:

$$N \times \mathrm{O}\left(h^{k+1}\right) = \mathrm{O}\left(h^{k}\right).$$

That is why the order of a multistep method with local error $\mathrm{O}\left(h^{k+1}\right)$ is only k, rather than $(k + 1)$.

While a high order of accuracy is generally desirable, it should not be the only, or even the primary principle for constructing an ODE solver. The situation here is analogous to quadrature. Recall that one can construct Newton-Cotes rules with arbitrary degree of accuracy simply by taking enough (equispaced) nodes. Yet, only low degree Newton-Cotes rules are numerically stable and therefore useful in practice. Similarly, it is very easy to construct a numerical solver that has an arbitrary degree of accuracy but which is numerically unstable and

therefore useless. Therefore it is important to combine accuracy with stability requirements.

The stability of Adams methods, and multistep schemes in general, is even more subtle than that of quadrature rules. It will be discussed in the next handout.

9.12. **Numerical results.** As the first model problem, consider the *logistic equation*

$$\frac{dx}{dt} = x\,(1-x), \quad x(0) = \frac{1}{2}, \tag{70}$$

on the interval $[0, 5]$. Figure 12 shows the results of solving (70) using Euler's method with step sizes $h = 2^{-k}$, $k = 1, \dots, 10$. Higher dot density corresponds to smaller h.

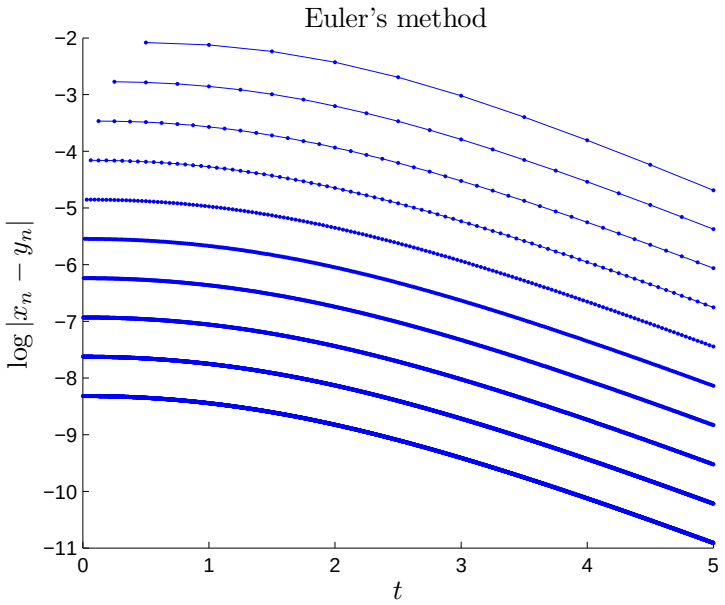

FIGURE 12. The error of Euler's method, as applied to $x' = x\,(1-x)$, $x(0) = \frac{1}{2}$ with step sizes $h = 2^{-k}$, for $k = 1, \dots, 10$

This is Euler's method on its best behavior: for each step size the local error decreases with each step. Notice also that the curves corresponding to different step sizes appear to be equispaced. For a method of order one halving the step size should reduce the logarithm of the global error by about $\log 2$: Figure 12 clearly confirms that. As we will soon see this does not always happen!

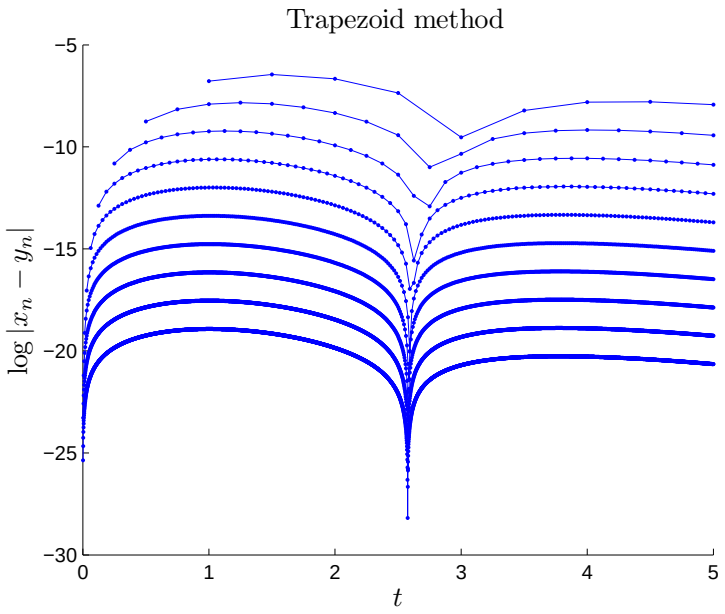

FIGURE 13. The error of the trapezoid method, as applied to $x' = x(1-x)$, $x(0) = \frac{1}{2}$, with $h = 2^{-k}$, $k = 1, \ldots, 10$.

Figure 13 shows the results of applying the trapezoid method to IVP (70). There is a peculiar dip in the middle, however, it is just noise coming from the IVP. Quite generally, the behavior of the local error can be very chaotic because it heavily depends on the underlying differential equation. In contrast, the behavior of the global error largely depends on the method and is far more predictable. The curves in Figure 13 are roughly spaced by $2\log(2)$ units apart. This shows that the global error of the trapezoid rule decreases quadratically with step size h, as we would expect from a second order method.

Of course, the best way to compare any two ODE solvers is to plot their global errors on the same log-log plot. If a method has order k, its global error is roughly proportional to h^k. Therefore:

$$\log E \sim k \log h.$$

For the Euler's scheme the log-log plot of E against h should be close to a line with slope one; for the trapezoid rule the slope should be approximately two.

Figure 14 shows that that is indeed the case. Remarkably, the plots are perfectly linear with slopes one and two, correct to five decimals! While the linear fit is not always so perfect, it is quite clear that our

definitions of global error and order are appropriate since they do not depend on the ODE.

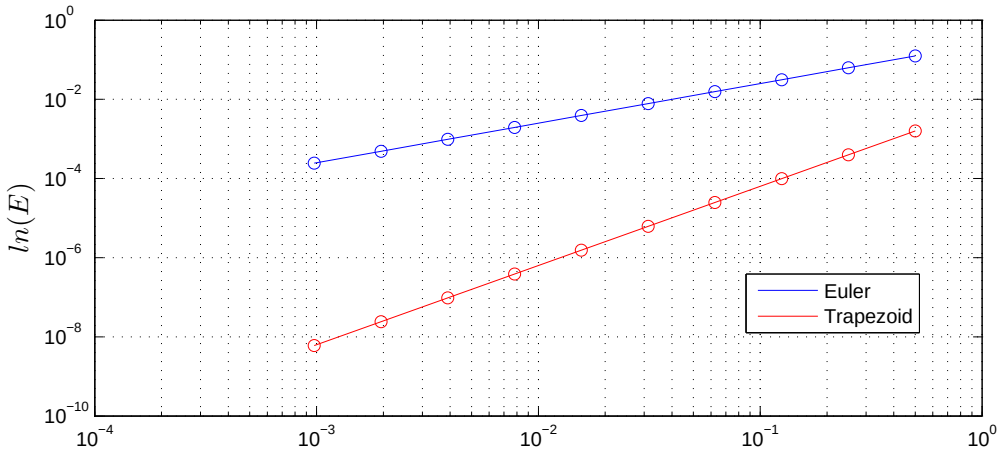

FIGURE 14. Log-log plots of the global errors produced by Euler's method (blue) and the trapezoid method (red) when applied to $x' = x\,(1 - x)$, $x(0) = \frac{1}{2}$ on $[0, 5]$ with step size $h = 2^{-k}$, $k = 1, \ldots, 10$

We conclude with a cautionary tale. Consider the following innocuous linear IVP

$$\frac{dx}{dt} = 20\,(\cos(2\,t) - x) - 2\,\sin(2\,t), \quad x(0) = 1, \tag{71}$$

whose exact solution is $x = \cos(2\,t)$. At first glance there is nothing in IVP (71) that suggests trouble. And yet, applying Euler's method to Equation (71) leads to an unpleasant surprise: the global error peaks at 10^{10}. Furthermore, at first the global error of Euler's method actually increases as the step size is decreased! Eventually, the error does begin to decrease, in accordance with Theorem 8. However, it is clear that Euler's method cannot be used with *stiff* Equations such as (71). In contrast, the global error of the trapezoid rule remains predictable. Clearly, the trapezoid rule is much more stable than Euler's method. Although even the trapezoid rule seems to be affected by stiffness: notice that the slope of the red curve is one rather than two.

The nature of the phenomenon shown in Figure 15 will become clear once we discuss linear stability theory. In the meantime, consider Figure 15 an illustration of the importance of stability.

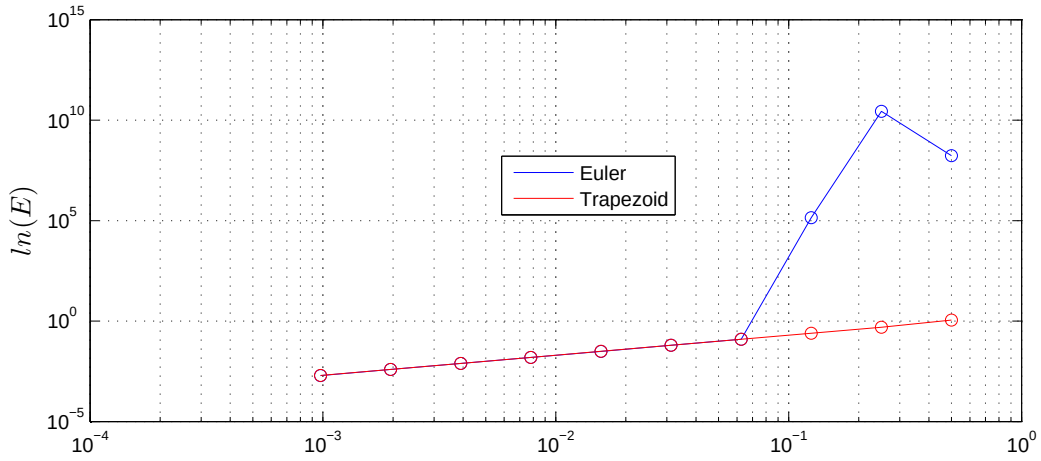

FIGURE 15. Log-log plots of the global errors produced by Euler's method (blue) and the trapezoid method (red) when applied to $x' = 20\,(\cos(2\,t) - 2\,\sin(2\,t))$, $x(0) = 1$ on $[0, 5]$ with step size $h = 2^{-k}$, $k = 1, \ldots, 10$

EXERCISES

(1) Prove that the trapezoid method (58) converges.
(2) Consider the following two-step implicit scheme:

$$y_{n+2} - 3\,y_{n+1} + 2\,y_n = h\left(\frac{13}{12}\,f_{n+2} - \frac{5}{3}\,f_{n+1} - \frac{5}{12}\,f_n\right)$$

Here $f_k = f(t_k, y_k)$.
   (a) Find the order of accuracy.
   (b) Test the stability of the scheme by applying it to the trivial IVP:

$$\frac{dx}{dt} = 0, \quad x(0) = 1,$$

   on $[0, 20]$ with $h = .1$, $h = .05$, and $h = .025$; for each step size initialize the method using exact values $y_0 = y_1 = 1$. Is the method stable?
(3) Find the orders of Adams-Bashford methods for $N = 2, 3, 4, 5$. What can you say about the general pattern?
(4) Find an implicit three-step method of order six (it is unique). Is it stable?