Numerical Methods

- A **first order ODE initial value problem**:

$$\frac{dy(t)}{dt} = f\left(t, y(t)\right) \tag{1.1}$$

$$y(0) = y_0 \tag{1.2}$$

where f is some function of t and $y(t)$, (e.g. $f(t, y(t)) = t^2 + sin(y(t))$), and y(t) is the differentiable function that satisfies this equation.

This is called a **first order** (because the highest derivative is a first derivative, $\frac{dy(t)}{dt}$) **ordinary** (because the derivative is not a partial derivative) **Differential Equation** (obvious!) **Initial Value Problem** (because the initial value – the value of the function $y(t)$ at the starting point $t = 0$ – is specified)

Note: It may seem of limited value to only know how to solve a first order equation, but in fact this is pretty general. In the equation, $y(t)$ may be a vector, and then this is a system of first order ODEs. This is extremely general because every higher order ODE can be written as a system of first order ODEs.

But before we rush in and try to find a solution to this kind of equation, we need to make sure that there exists a unique solution.

- **Definitions**

1. **Lipschitz Continuity:** a function $f(t, y(t))$ is said to be Lipschitz continuous with respect to y if there exists a number $L > 0$ such that

$$||f(t, y_1) - f(t, y_2)|| \leq L||y_1 - y_2||$$

for all y_1 and y_2 in the relevant domain.

Lipschitz continuity is stronger than continuity but weaker than differentiability. This means that differentiability implies Lipschitz continuity, and Lipschitz continuity implies continuity.

2. **Stability:** *If two solutions start close together they will stay close together.*
For any $\epsilon > 0$ there exists $\delta > 0$ such that if $|y_0 - z_0| < \delta$ then the solution $y(t)$ of

$$\frac{dy(t)}{dt} = f\left(t, y(t)\right)$$

$$y(0) = y_0$$

and the solution $z(t)$ of

$$\frac{dz(t)}{dt} = f\left(t, z(t)\right)$$

$$z(0) = z_0$$

will satisfy $|y(t) - z(t)| < \epsilon$ for all $0 \leq t \leq T$.

- **Existance and uniqueness of a solution of (1.1):**

  Theorem If $f(t, y)$ is continuous with respect to t, and Lipschitz continuous with respect to y, then (1.1) has a unique, stable solution.

- **Numerical Solution of an ODE:**

  The idea behind numerical solutions of a *Differential Equation* is to replace differentiation by differencing. A computer cannot differentiate but it can easily do a difference. (Differentiation is a continuous process. Differencing is a discrete process.)

  Now we introduce the most important tool that will be used in this class. By the time this class ends, you'll be able to do **Taylor Expansions** in your sleep.

  Taylor Series Expansion: $y(t + \Delta t) = y(t) + y'(t)\Delta t + O(\Delta t^2)$.

  So:

$$y'(t) = \frac{y(t + \Delta t) - y(t)}{\Delta t} + O(\Delta t).$$

  And now we can attempt to solve (1.1) by replacing the derivative with a difference:

$$y((i + 1)\Delta t) \approx y(i\Delta t) + \Delta t f(i\Delta t, y(i\Delta t))$$

  Starting with $y(0)$ and stepping forward.

  What's good about this? If the O term is something nice looking, this quantity decays with Δt, so if we take Δt smaller and smaller, this gets closer and closer to the real value.

  What can go wrong? The O term may be ugly. The errors can accumulate as I step forward in time. Also, even though this may be a good approximation for $y'(t)$ it may not converge to the right solution.

  To answer these questions, let's look at this scheme in depth:

- **Euler Forward Method**

  From now on, we'll call y_i the numerical approximation to the solution $y(i\Delta t)$; $t_i = i\Delta t$.

$$y_{i+1} \quad = \quad y_i + \Delta t f(t_i, y_i) \qquad i = 1, ..., N - 1 \tag{1.3}$$

  This method is **explicit** because given y_0, everything on the right-hand-side is known and I can immediately get y_1 (and so on).

  Local Truncation Error: The LTE of the Euler Forward method is defined by $LTE(t) = \frac{y(t+\Delta t)-y(t)}{\Delta t} - f(t, y(t))$. *i.e.* it is the residue when the exact solution of the ODE (1.1) is plugged into the numerical scheme. If y_i is close to $y(t_i)$ then the LTE will be close to zero.

  If I don't know $y(t)$, what is the use of this definition? (and if I do know $y(t)$, what do I need the method for?!). It turns out that even without explicit knowledge of the

solution we can still calculate the LTE and use it as an estimate and control of the error, by placing certain smoothness assumptions on $y(t)$ and using Taylor Expansions.

For example, in our case:

$$\begin{aligned} LTE(t) &= \frac{y(t+\Delta t)-y(t)}{\Delta t} - f(t,y(t)) \\ &= y'(t) + \frac{1}{2}y''(\xi)\Delta t - f(t,y(t)) \\ &= \frac{1}{2}y''(\xi)\Delta t \end{aligned}$$

If we know that $|y''(t)| \leq M$, then:

$$|LTE(t)| \leq \frac{1}{2}M\Delta t$$

(*Quick Remark:* How do we get a bound for $y''(t)$? If f is smooth, then $y'(t) = f(t,y)$ and so $|y'(t)| \leq M$. The $y''(t) = \frac{d}{dt}f(t,y) = f_t + f_y y'(t)$ so it is bounded! In practice, we just assume that whatever derivative we need is bounded.

- **Global Error**

  How much does the LTE really mean? We are interested in the real error, not the one at one step.

  Global Error

  $$E(t_i) = y_i - y(t_i)$$

  or

  $$E(t_i) = y(t_i) - y_i.$$

  Now we assume a certain amount of smoothness, so

  $y(t_{i+1}) = y(t_i) + \Delta t f(t_i, y(t_i)) + \Delta t LTE(t_i)$

  and

  $y_{i+1} = y_i + \Delta t f(t_i, y_i)$

  Putting these together we get:

  $$E_{i+1} = E_i + \Delta t\left(f(t_i, y(t_i)) - f(t_i, y_i)\right) + \Delta t LTE(t_i).$$

  Now, let

  $$L_i = \frac{f(t_i, y(t_i)) - f(t_i, y_i)}{y(t_i) - y_i}.$$

  Since f is Lipschitz continuous with respect to y, this means:

  $$|L_i| = |\frac{f(t_i, y(t_i)) - f(t_i, y_i)}{y(t_i) - y_i}| \leq L$$

and so

$$e_{i+1} = e_i(1 + \Delta t L_i) + LTE(t_i)\Delta t$$
$$|e_{i+1}| \leq |e_i|(1 + \Delta L) + \frac{1}{2}M\Delta t^2 \qquad i = 1, ..., N-1$$

Lemma: If $z_{i+1} \leq z_i(1 + a\Delta t) + b$ Then $z_i \leq e^{ai\Delta t}(z_0 + \frac{b}{a\Delta t})$
Proof:

$$\begin{aligned}
z_{i+1} &\leq z_i \quad (1 + a\Delta t) + b \\
&\leq \quad (z_{i-1}(1 + a\Delta t) + b)(1 + a\Delta t) + b \\
&\leq \\
&\cdot \\
&\cdot \\
&\cdot \\
&\leq \quad z_0(1 + a\Delta t)^{i+1} + b(1 + (1 + a\Delta t)... + (1 + a\Delta t)^i) \\
&= \quad z_0(1 + a\Delta t)^{i+1} + b\frac{(1 + a\Delta t)^{i+1} - 1}{1 + a\Delta t - 1} \\
&\leq \quad z_0(1 + a\Delta t)^{i+1} + \frac{b}{a\Delta t}(1 + a\Delta t)^{i+1} \\
&\leq \quad (1 + a\Delta t)^{i+1}\left(z_0 + \frac{b}{a\Delta t}\right) \\
&\leq \quad e^{a\Delta t(i+1)}\left(z_0 + \frac{b}{a\Delta t}\right)
\end{aligned}$$

so

$$z_i \leq e^{ai\Delta t}(z_0 + \frac{b}{a\Delta t})$$

From this we obtain:

$$|E_i| \leq e^{Li\Delta t}(|E_0| + \frac{M}{2L}\Delta t)$$

Now, if $i\Delta t \leq T$ then

$$|E_i| \leq e^{LT}(|E_0| + \frac{M}{2L}\Delta t)$$

and since $|E_0| = 0$ we have:

Global Error of Euler Forward:

$$|E_i| \leq e^{LT}(\frac{M}{2L}\Delta t).$$

Compare this with the local error:

$$|LTE(t)| \leq \frac{1}{2}M\Delta t$$

we see that the global error has the same order as the local error with a different coefficient in the estimates. They are related by the Lipschitz constant L and the final time T.

- **Order of a Scheme**

  Definition The **Order** of a scheme r, is defined by $|E_i| = O(\Delta t^r)$.

  The higher the order of the scheme, the faster the error decays.

- Last time we examined the Euler Forward method:

$$y_{i+1} = y_i + \Delta t f(t_i, y_i) \quad i = 1, ..., N-1 \tag{1.4}$$

  We saw that this method has **Local Truncation Error:** $LTE(t) = O(\Delta t)$ and **Global Error** $|E_i| = O(\Delta t)$.

  The important thing to understand is that the **Local Truncation Error** is not always an indicator of what the **global error** will do. Schemes that have the same order of LTE and global error are good schemes.

- **Taylor Series Methods**

  To derive these methods we start with a Taylor Expansion:

$$y(t + \Delta t) \approx y(t) + \Delta t y'(t) + \frac{1}{2}\Delta t^2 y''(t) + ... + \frac{1}{r!}y^{(r)}(t)\Delta t^r.$$

  Now I take the equation and get:

$$\begin{aligned} y'(t) &= f(t, y(t)) \\ y''(t) &= f_t + f_y y'(t) \\ &= f_t + f_y f \\ y'''(t) &= f_{tt} + ... \end{aligned}$$

  The scheme is, then:

$$y_{i+1} = y_i + f_i \Delta t + \frac{f_{t_i} + f_{y_i} f_i}{2}\Delta t^2.$$

  The Taylor series method can be written as

$$y_{i+1} = y_i + \Delta t F(t_i, y_i, \Delta t)$$

  and we can easily show that for this scheme

$$LTE(t) = \frac{y^{(r+1)}(\xi)}{(r+1)!}\Delta t^r = O(\Delta t^r).$$

  (we can assume $y^{(r+1)}$ is bounded by M.)

  So the Local Truncation error is reasonable, but what about the global error? Just as in the Euler Forward case, we can show that

$$|E_i| \leq \frac{Mr}{(r+1)!}\frac{1}{L}e^{LT}\Delta t^r$$

  where L is the Lipschitz constant of F.

Advantages and Disadvantages of the Taylor Series Method

advantages	a) One step, explicit
	b) can be high order
	c) convergence proof easy
disadvantages	Needs the explicit form
	of derivatives of f.

- **Runge-Kutta Methods**

  These are still **one step** methods, but they are written out so that they don't look messy:

  Second Order Runge-Kutta Methods:

$$\begin{aligned} k_1 &= \Delta t f(t_i, y_i) \\ k_2 &= \Delta t f(t_i + \alpha\Delta t, y_i + \beta k_1) \\ y_{i+1} &= y_i + ak_1 + bk_2 \end{aligned}$$

  let's see how we can chose the parameters a,b, α, β so that this method has the highest order LTE possible.

$$\begin{aligned} k_1(t) &= \Delta t f(t, y(t)) \\ k_2(t) &= \Delta t f(t + \alpha\Delta t, y + \beta k_1(t) \\ &= \Delta t \left(f(t, y(t) + f_t(t, y(t))\alpha\Delta t + f_y(t, y(t))\beta k_1(t) + O(\Delta t^2) \right) \\ LTE(t) &= \frac{y(t+\Delta t) - y(t)}{\Delta t} - \frac{a}{\Delta t} f(t, y(t))\Delta t - \frac{b}{\Delta t} \left(f_t(t, y(t))\alpha\Delta t + f_y(t, y(t)\beta k_1(t) \right. \\ &+ \left. f(t, y(t))\,\Delta t + O(\Delta t^2) \right) \\ &= \frac{y(t+\Delta t) - y(t)}{\Delta t} - af(t, y(t)) - bf(t, y(t)) - bf_t(t, y(t))\alpha \\ &- bf_y(t, y(t)\beta f(t, y(t)) + O(\Delta t^2) \\ &= y'(t) + \frac{1}{2}\Delta t y''(t) - (a+b)f(t, y(t)) - \Delta t(b\alpha f_t(t, y(t)) + b\beta f(t, y(t)) f_y(t, y(t)) + O(\Delta t^2) \\ &= (1 - a - b)f + (\frac{1}{2} - b\alpha)\Delta t f_t + (\frac{1}{2} - b\beta)\Delta t f_y f + O(\Delta t^2) \end{aligned}$$

  So we want $a = 1 - b$, $\alpha = \beta = \frac{1}{2b}$.

  Fourth Order Runge-Kutta Methods:

$$\begin{aligned} k_1 &= \Delta t f(t_i, y_i) && (1.5) \\ k_2 &= \Delta t f(t_i + \frac{1}{2}\Delta t, y_i + \frac{1}{2}k_1) && (1.6) \\ k_3 &= \Delta t f(t_i + \frac{1}{2}\Delta t, y_i + \frac{1}{2}k_2) && (1.7) \\ k_4 &= \Delta t f(t_i + \Delta t, y_i + k_3) && (1.8) \\ y_{i+1} &= y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) && (1.9) \end{aligned}$$

The second order method requires 2 evaluations of f at every timestep, the fourth order method requires 4 evaluations of f at every timestep. In general: **For an rth order Runge-Kutta method we need S(r) evaluations of f for each timestep, where**

$$S(r) = \begin{cases} r & \text{for } r \leq 4 \\ r+1 & \text{for } r = 5 \text{ and } r = 6 \\ \geq r+2 & \text{for } r \geq 7 \end{cases}$$

Practically speaking, people stop at $r = 5$.

Advantages of Runge-Kutta Methods

1. One step method – same convergence proof as for Euler.
2. Don't need to know derivatives of f.

- **Multi Step Methods**

$$\frac{y_{i+1} + \alpha_1 y_i + \alpha_2 y_{i-1} + ... + \alpha_m y_{i-m+1}}{\Delta t} = \beta_0 f_{i+1} \beta_1 f_i + ... + \beta_m f_{i-m+1}$$

What have we got to gain? Accuracy with fewer function evaluations. But then the storage cost is high.

What's the drawback? We need starting values. You can get these by using a one step method!

Example: A two-step method:

$$\frac{y_{i+1} - y_{i-1}}{\Delta t} = 2 f_i.$$

The LTE is $O(\Delta t^2)$. Is the global error this good? We cannot do the same proof as the Euler, because this is a two step method.

We need a better way to check if the global error is the same order as the LTE. To be on the safe side, always take a method from the book!

- **Advanced topic: Stability**

First, we need some definitions:

1. **Stability:** A scheme is said to be stable if for fixed T and for every $\epsilon > 0$ there exists $\delta > 0$ such that if $0 \leq i\Delta t \leq T$ and $|u_0 - v_0| < \delta$ then $|u_i - v_i| < \epsilon$.
   Stability means that if initial conditions are close the numerical solutions must be close
2. **Absolute Stability** A scheme is said to be absolutely stably for $\lambda \delta t$ is, when the scheme is applied to
   $$y'(t) = \lambda y(t)$$
   with some initial condition $y(0) = y_0$, $|y_i$ is bounded as $i \to \infty$, Δt fixed.

3. **Characteristic Polynomial:** the characteristic polynomial of the multistep method

$$\frac{y_{i+1} + \alpha_1 y_i + \alpha_2 y_{i-1} + ... + \alpha_m y_{i-m+1}}{\Delta t} = \beta_0 f_{i+1} \beta_1 f_i + ... + \beta_m f_{i-m+1}$$

is

$$P(\lambda) = \lambda^m + \alpha_1 \lambda^{m-1} + ... + \alpha_m.$$

Stability is clearly a necessary condition for convergence. In our case, stability is also sufficient.

Theorem 1 If a scheme is stable and has $LTE(t) = O(\Delta t^r)$ Then it is convergent with $|E_i| = O(\Delta t^r)$ for $0 \leq i\Delta t \leq T$.

Theorem 2 A one-step method $y_{i+1} = y_i + F(t_i, y_i, \Delta t)$ is stable is F is Lipschitz continuous with respect to y.

Theorem 3 The multistep method is stable iff the following *root condition* is satisfied: If λ is a root of the characteristic polynomial $P(\lambda)$ then either $|\lambda| < 1$ or $|\lambda| = 1$ and is a simple root.

- **Implicit vs. Explicit methods and *stiff* problems:** Let's look at the simple problem $y' = ay$ where a is a negative number with a large magnitude, e.g. $a = -1000$. Clearly, the exact solution is then $y = Ce^{ay}$ which is a rapidly decaying function. Let's see what Euler's method does to this:

$$\begin{aligned} y^{n+1} &= y^n + \Delta t f(y^n) \\ y^{n+1} &= y^n + \Delta t a y^n \\ y^{n+1} &= (1 + a\Delta t)\, y^n \end{aligned}$$

If we start this up from y_0 we see that

$$\begin{aligned} y^1 &= (1 + a\Delta t)\, y_0 \\ y^2 &= (1 + a\Delta t)\, y_1 \\ &= (1 + a\Delta t)^2\, y_0 \\ y^3 &= (1 + a\Delta t)\, y_2 \\ &= (1 + a\Delta t)^3\, y_0 \end{aligned}$$

or, in general

$$y^n = (1 + a\Delta t)^n\, y_0.$$

Now, since the exact solution is rapidly decaying, we expect that the numerical solution should also decay – i.e. we need $|1 + a\Delta t| \leq 1$. Is this always true? Well, since $a < 0$ and $\Delta t > 0$ we always have

$$1 + a\Delta t \leq 1.$$

But we also need

$$1 + a\Delta t \geq -1$$

and this is only true if

$$\Delta t \leq \frac{2}{a} = \frac{1}{500}.$$

This is a very limiting time-step! In fact, if you don't observe this time-step your numerical solution will blow up! A problem that exhibits such rapid decay is called a *stiff* problem, and has a time-step restriction that is quite severe.

Can we avoid it? Yes, but for this you have to use an *implicit* method. In such a method you will need to solve a problem of the form $g(y^{n+1}) = 0$ for y^{n+1}, which may be unpleasant . . . (MATLAB will do it for you if necessary, and we will only do easy problems we can solve ourselves).

Instead of using Euler's method, we can use the implicit Euler's method:

$$\begin{aligned} y^{n+1} &= y^n + \Delta t f(y^{n+1}) \\ y^{n+1} &= y^n + \Delta t a y^{n+1} \\ (1 - a\Delta t) y^{n+1} &= y^n \\ y^{n+1} &= \frac{1}{1 - a\Delta t} y^n \end{aligned}$$

If we start this up from y_0 we see that

$$y^n = \left(\frac{1}{1 - a\Delta t}\right)^n y_0.$$

Now, is this quantity decaying?

$$\left|\frac{1}{1 - a\Delta t}\right| \leq 1$$

yes, because this quantity is always positive and $1 - a\Delta t > 1$.

So an implicit method can be used, with any size timestep that would be ok for accuracy.