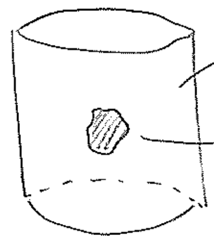
AMATH 342 - Computational Methods for Differential Equations

2016 01 04

Ordinary Differential Equations (ODEs)

Example



$A < T \to$ rock cools

$A > T \to$ rock heats up

Newton's Law of Cooling:

$$\underbrace{\frac{dT}{dt}}_{\text{time rate of change of temperature}} = -\underbrace{k}_{\text{proportionality constant } (k>0)} \underbrace{(T-A)}_{\text{difference between liquid and rock temperature}}$$

ODE: given k and A we can predict the future temperature of the rock

Example 2

the time rate of change of a population P(t) with constant birth and death rate is proportional to the size of the population:

$$\frac{dP}{dt} = kP$$

Example 3

Consider a particle with mass m. The motion of this particle along a straight line (x-axis) is described by its position function, $x = f(t)$.

The velocity of the particle is defined as

$$v(t) = f'(t) \longleftrightarrow v = \frac{dx}{dt}.$$

The acceleration of particle

$$a(t) = v'(t) = f''(t) \longleftrightarrow a = \frac{dv}{dt} = \frac{d^2x}{dt^2}$$

Newton's Second Law of Motion: if a force F(t) acts on a particle and is directed along its line of motion then

$$ma = F \longleftrightarrow m\frac{d^2x}{dt^2} = F$$

(Example 2)

$$\frac{dP}{dt} = kP$$

Every function of the form

$$P(t) = C\exp(kt)$$

is a solution of the ODE:

$$\frac{dP}{dt} = Ck\exp(kt) = k\left(C\exp(kt)\right) = kP(t).$$

Given P at time $t=0$ as $P(0)=P_0$ then the particular solution is

$$P(t) = P_0 \exp(kt).$$

What if we need to solve

$$\vec{y}' = \vec{f}(t,\vec{y}), \quad t \geq t_0, \quad \vec{y}(t_0) = \vec{y}_0 \,?$$

AMATH 342 goal is to approximate the solution of this problem using numerical methods.

- ~8 weeks: numerical methods for ODEs
- ~4 weeks: numerical methods for PDEs

ODE example:

$$\frac{dT}{dt}(t) = -k\left(T(t) - A\right)$$

PDE example: $$\frac{\partial T}{\partial t}(x,t) = k\,\frac{\partial^2 T}{\partial x^2}(x,t)$$

Solve

① $$\vec{y}' = \vec{f}(t,\vec{y}), \quad t \geq t_0, \quad \vec{y}(t_0) = \vec{y}_0$$

where

- $\vec{y}_0 \in \mathbb{R}^d$ a given vector
- notation: $\vec{y}$ is a vector, $\mathbb{R}^d$ is d-dimensional Euclidean space
- $\vec{f}(t,\vec{y}) : [t_0, \infty) \times \mathbb{R}^d \mapsto \mathbb{R}^d$ is a "well-behaved" function

What do we mean by "well-behaved" function $\vec{f}(t,\vec{y})$?

In this course we assume $\vec{f}(t,\vec{y})$ satisfies the <u>Lipschitz condition</u>:

$$\|\vec{f}(t,\vec{x}) - \vec{f}(t,\vec{y})\| \leq \lambda \|\vec{x} - \vec{y}\| \quad \forall \vec{x},\vec{y} \in \mathbb{R}^d,\ t \geq t_0$$

where $\lambda \in \mathbb{R}^+$ $(\lambda > 0)$ is the Lipschitz constant

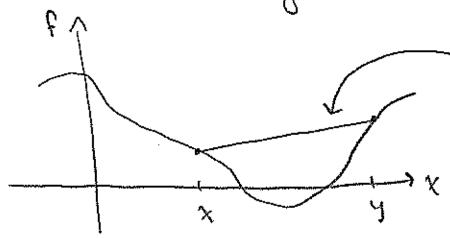
If f satisfies the Lipschitz condition then it is possible to prove that ① has a unique solution.

1-D Lipschitz condition:

- $|f(x) - f(y)| \leq \lambda |x-y|$

  if $\exists \lambda \in (0,\infty)$ then f is Lipschitz continuous:

  For every pair of points $(f(x), f(y))$, the absolute value of the slope of the line connecting them is not greater than λ.



slope $= \dfrac{|f(y)-f(x)|}{|y-x|} \leq \lambda$

Example 1: $f(x) = |x|$ is Lipschitz continuous but <u>not</u> continuously differentiable.

Proof: $|f(x)-f(y)| = ||x|-|y|| \leq |x-y|$ by the reverse triangle inequality, so $\lambda = 1$.

Example 2: $f(x) = \sqrt{x}$, $x \in [0,1]$ is not Lipschitz continuous, but is continuous.

Proof: $f'(x) = \frac{1}{2\sqrt{x}}$, as $x \downarrow 0 \rightarrow f'(x) = \infty$

there is no λ such that $|\sqrt{x} - \sqrt{y}| \leq \lambda |x-y|$ for $x, y \in [0,1]$

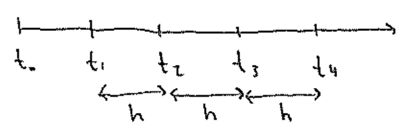
2016 01 05

Solve

$$\underbrace{\vec{y}' = \vec{f}(t, \vec{y})}_{\text{Given } \vec{y} \in \mathbb{R}^d \text{ and } t \geq t_0 \text{, we know the slope of } \vec{y}}, \quad t \geq t_0, \quad \underbrace{\vec{y}(t_0) = \vec{y}_0}_{\text{initial condition}}$$

Information:

Time axis, with grid:



Sequence t_0, $t_1 = t_0 + h$, $t_2 = t_0 + 2h$, ...

Given $\vec{y}_0$ at t_0 and the slope at t_0, can we then guess the value of $\vec{y}$ at $t=t_1$?

<u>Idea</u>: use linear interpolation $\Rightarrow$ we make the approximation that

$$\vec{f}(t,\vec{y}) \approx \vec{f}(t_0,\vec{y}_0) \text{ for } t\in[t_0,t_1]$$

$$\vec{y}\,'(t) = \vec{f}(t,\vec{y}) \approx \vec{f}(t_0,\vec{y}_0)$$

Integrate

$$\int_{t_0}^{t_1} \vec{y}\,'(t)\,dt \approx \int_{t_0}^{t_1} \vec{f}(t_0,\vec{y}_0)\,dt$$

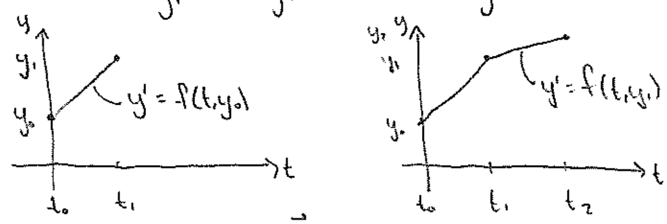
$$\vec{y}(t_1)-\vec{y}(t_0) \approx h\vec{f}(t_0,\vec{y}_0)$$

Notation: $\vec{y}_n$ is the numerical estimate of the exact solution $\vec{y}(t_n)$

$$\vec{y}(t_1) \approx \vec{y}(t_0) + h\vec{f}(t_0,\vec{y}(t_0))$$

$$\vec{y}_1 = \vec{y}_0 + h\vec{f}(t_0,\vec{y}_0)$$

1-D:



$$\vec{y}_2 = \vec{y}_1 + h\vec{f}(t_1,\vec{y}_1)$$

Repeat again to find $\vec{y}_3, \vec{y}_4, \vec{y}_5$, etc. We obtain the recursive scheme

$$\vec{y}_{n+1} = \vec{y}_n + h\vec{f}(t_n,\vec{y}_n), \quad n=0,1,2,3,\dots$$

This is called the <u>Euler Method</u>.

How good is the Euler Method at approximating (1)?

* Consider the time interval $[t_0, t_0+t^*]$
* Cover the time interval by an equidistant grid

* Every h results in a different grid and the smaller h the more time steps are in $[t_0, t_0+t^*]$
* The numerical solution $\vec{y}_n$ on a grid with time step h is denoted by $\vec{y}_{n,h}$, $n=0,1,2,\dots,\lfloor t^*/h \rfloor$

Important: a numerical method is convergent if for every ODE problem

$$\vec{y}\,' = \vec{f}(t,\vec{y}), \quad t\ge t_0, \quad \vec{y}(t_0)=\vec{y}_0$$

with $\vec{f}$ a Lipschitz function we have

$$\lim_{h\to 0} \max_{n=0,1,\dots,\lfloor t^*/h\rfloor} \| \vec{y}_{n,h} - \vec{y}(t_n) \| = 0$$

Is the Euler method convergent? Yes.

Proof: Assume that $\vec{f}$ and $\vec{y}$ are analytic. Define the numerical error as

$$\vec{e}_{n,h} = \underbrace{\vec{y}_{n,h}}_{\text{numerical sol.}} - \underbrace{\vec{y}(t_n)}_{\text{exact sol.}}$$

To prove: $\lim_{h\to 0} \max_{n} \|\vec{e}_{n,h}\| = 0$. Use Taylor's theorem to find that

$$\vec{y}(t_{n+1}) = \vec{y}(t_n) + h\vec{y}'(t_n) + \mathcal{O}(h^2)$$
$$= \vec{y}(t_n) + h\vec{f}(t, \vec{y}(t_n)) + \mathcal{O}(h^2)$$

Euler's method.

$$\vec{y}_{n+1,h} = \vec{y}_{n,h} + h\vec{f}(t, \vec{y}_n)$$

Subtract the two:

$$\vec{y}_{n+1,h} - \vec{y}(t_{n+1}) = \vec{y}_{n,h} - \vec{y}(t_n) + h\left(\vec{f}(t_n, y_n) - \vec{f}(t_n, \vec{y}(t_n))\right) + \mathcal{O}(h^2)$$
$$\rightarrow \quad \vec{e}_{n+1,h} = \vec{e}_{n,h} + h\left(\vec{f}(t_n, \vec{y}_n(t_n) + e_{n,h}) - \vec{f}(t_n, \vec{y}(t_n))\right) + \mathcal{O}(h^2)$$
$$\rightarrow \|\vec{e}_{n+1,h}\| = \|\vec{e}_{n,h} + h(\vec{f}(t_n, \vec{y}_n(t_n) + e_{n,h}) - \vec{f}(t_n, \vec{y}(t_n))) + \mathcal{O}(h^2)\|$$
$$\leq \|\vec{e}_{n,h}\| + h\|\vec{f}(t_n, \vec{y}_n(t_n) + \vec{e}_{n,h}) - \vec{f}(t_n, \vec{y}(t_n))\| + ch^2$$

$\vec{f}$ is Lipschitz, so

$$\|\vec{f}(t_n, \vec{y}(t_n) + \vec{e}_{n,h}) - \vec{f}(t_n, \vec{y}(t_n))\| \leq \lambda \|\vec{e}_{n,h}\|$$
$$\rightarrow \|\vec{e}_{n+1,h}\| \leq (1 + h\lambda)\|\vec{e}_{n,h}\| + ch^2, \quad n = 0, 1, 2, \ldots, \lfloor t^*/h \rfloor$$

Claim: $\|\vec{e}_{n,h}\| \leq \frac{c}{\lambda} h((1+h\lambda)^n - 1)$, $n = 0, 1, 2, \ldots$

Proof: Induction $\underline{n=0}$: $\|\vec{e}_{n,h}\| = 0$ ✓

$\underline{n>0}$: Assume claim is true up to n. True for $n+1$?

$$\|\vec{e}_{n+1,h}\| \leq (1+h\lambda)\|\vec{e}_{n,h}\| + ch^2$$
$$\leq (1+h\lambda)\frac{c}{\lambda} h((1+h\lambda)^n - 1) + ch^2$$
$$= \frac{c}{\lambda} h((1+h\lambda)^{n+1} - 1).$$

□

Now look at

$$\exp(x) = 1 + x + \tfrac{1}{2}x^2 + \tfrac{1}{6}x^3 + \ldots$$

so if $x > 0$ then $1 + x < \exp(x)$.

In our case, $h\lambda > 0$ so $1 + h\lambda < \exp(h\lambda)$. $\rightarrow (1+h\lambda) < \exp(nh\lambda)$, $n = 0, 1, 2, \ldots \lfloor t^*/h \rfloor$

$$\leq \exp(\lfloor t^*/h \rfloor h\lambda)$$
$$\leq \exp(t^*\lambda)$$

From

$$\|\vec{e}_{n,h}\| \leq \frac{c}{\lambda} h((1+\lambda h)^n - 1) \leq h\underbrace{(\exp(t^*\lambda) - 1)\frac{c}{\lambda}}_{\text{independent of } h}$$

So $\lim_{\substack{h\to 0 \\ 0 \leq nh \leq t^*}} \|\vec{e}_{n,h}\| = 0 \rightarrow$ Euler's method is convergent. ■

2016 01 07

Example 1: Solve the following ODE

$$\frac{dP}{dt} = kP, \quad P(0) = 1, \quad t \in [0,1]$$

where $k = 5$. <Matlab code/plots projected>

* Local error of Euler's method

$$\vec{y}_{n+1} = \vec{y}_n + h\vec{f}(t_n, \vec{y}_n)$$

Rewrite as

$$\vec{y}_{n+1} - \left(\vec{y}_n + h\vec{f}(t_n, \vec{y}_n)\right) = 0.$$

Replace $\vec{y}_n$ by the exact solution $\vec{y}(t_k)$, $k = 0, 1, 2, \ldots$

$$\vec{y}(t_{n+1}) - \left(\vec{y}(t_n) + h\vec{f}(t_n, \vec{y}(t_n))\right) \neq 0$$

$\Rightarrow$ use Taylor expansions

$$\vec{y}(t_{n+1}) = \vec{y}(t_n) + h\vec{y}'(t_n) + \mathcal{O}(h^2)$$

$$\rightarrow \vec{y}(t_{n+1}) - \left(\vec{y}(t_n) + h\vec{f}(t_n, \vec{y}(t_n))\right) = \left(\cancel{\vec{y}(t_n) + h\vec{f}(t_n, \vec{y}(t_n))} + \mathcal{O}(h^2)\right) - \left(\cancel{\vec{y}(t_n) + h\vec{f}(t_n, \vec{y}(t_n))}\right)$$

$$= \mathcal{O}(h^2)$$

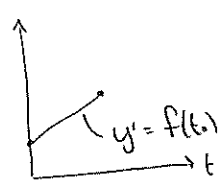
This is how well the numerical method approximates the ODE. To advance from t_n to t_{n+1} we are making an error of $\mathcal{O}(h^2)$

<u>Def</u>) Given some numerical method

$$\vec{y}_{n+1} = \vec{Y}_n(\vec{f}, h, \vec{y}_n, \ldots, \vec{y}_0). \qquad \left(= \vec{y}_n + h\vec{f}(t_n, \vec{y}_n) \text{ for Euler}\right)$$

it is of <u>order p</u> if

$$\vec{y}(t_{n+1}) - \vec{Y}_n(\vec{f}, h, \vec{y}(t_n), \vec{y}(t_{n-1}), \ldots, \vec{y}(t_0)) = \mathcal{O}(h^{p+1}).$$



Alternative: on $[t_0, t_1]$ approximate $\vec{f}(t, \vec{y})$ by

$$\frac{1}{2}\left(\vec{f}(t_0, \vec{y}_0) + \vec{f}(t_1, \vec{y}_1)\right).$$

$$\vec{y}'(t) \approx \frac{1}{2}\left(\vec{f}(t_0, \vec{y}_0) + \vec{f}(t_1, \vec{y}_1)\right)$$

$$\int_{t_0}^{t_1} \vec{y}'(t)\,dt \approx \int_{t_0}^{t_1} \frac{1}{2}\left(\vec{f}(t_0, \vec{y}_0) + \vec{f}(t_1, \vec{y}_1)\right)dt$$

$$\vec{y}(t_1) - \vec{y}(t_0) \approx \frac{1}{2}h\left(\vec{f}(t_0, \vec{y}_0) + \vec{f}(t_1, \vec{y}_1)\right)$$

$$\vec{y}_1 = \vec{y}_0 + \tfrac{1}{2}h\left(\vec{f}(t_0,\vec{y}_0) + \vec{f}(t_1,\vec{y}_1)\right)$$

Repeat over all time intervals $[t_n, t_{n+1}]$, $n = 0,1,2,\dots$ We obtain the recursive scheme

$$\vec{y}_{n+1} = \vec{y}_n + \tfrac{1}{2}h\left(\vec{f}(t_n,\vec{y}_n) + \vec{f}(t_{n+1},\vec{y}_{n+1})\right)$$

called the <u>Trapezium rule</u>.

$$\vec{y}_{n+1} - \left(\vec{y}_n + \tfrac{1}{2}h\left(\vec{f}(t_n,\vec{y}_n) - \vec{f}(t_{n+1},\vec{y}_{n+1})\right)\right)$$

Substitute the exact solution

① $$\vec{y}(t_{n+1}) - \left(\vec{y}(t_n) + \tfrac{1}{2}h\left(\vec{f}(t_n,\vec{y}(t_n)) - \vec{f}(t_{n+1},\vec{y}(t_{n+1}))\right)\right) = \delta$$

Taylor:

$$\vec{y}(t_{n+1}) = \vec{y}(t_n) + h\vec{y}\,'(t_n) + \tfrac{1}{2}h^2\vec{y}\,''(t_n) + O(h^3)$$
$$\vec{f}(t_{n+1},\vec{y}(t_{n+1})) = \vec{y}\,'(t_{n+1}) = \vec{y}\,'(t_n) + h\vec{y}\,''(t_n) + O(h^2)$$

Substitute in ① $\to$ $\delta = O(h^3)$ $\to$ Trapezoidal rule is <u>order 2</u>.

The trapezoidal converges.

Proof: num.: $\vec{y}_{n+1} = \vec{y}_n + \tfrac{1}{2}h\left(\vec{f}(t_n,\vec{y}_n) + \vec{f}(t_{n+1},\vec{y}_{n+1})\right)$

exact: $\vec{y}(t_{n+1}) = \vec{y}(t_n) + \tfrac{1}{2}h\left(\vec{f}(t_n,\vec{y}(t_n)) + \vec{f}(t_{n+1},\vec{y}(t_{n+1}))\right) + O(h^3)$

$$\vec{e}_{n+1,h} = \vec{e}_{n,h} + \tfrac{1}{2}h\left(\left(\vec{f}(t_n,\vec{y}_n) - \vec{f}(t_n,\vec{y}(t_n))\right) + \left(\vec{f}(t_{n+1},\vec{y}_{n+1}) - \vec{f}(t_{n+1},\vec{y}(t_{n+1}))\right)\right) + O(h^3)$$

Take the norm:

$$\|\vec{e}_{n+1,h}\| = \|RHS\| \le \|\vec{e}_{n,h}\| + \tfrac{1}{2}h\underbrace{\|\vec{f}(t_n,\vec{y}_n) - \vec{f}(t_n,\vec{y}(t_n))\|}_{(1)} + \tfrac{1}{2}h\underbrace{\|\vec{f}(t_{n+1},\vec{y}_{n+1}) - \vec{f}(t_{n+1},\vec{y}(t_{n+1}))\|}_{(2)} + ch^3$$

$\vec{f}$ is Lipschitz so

$$(1) \le \lambda_1\|\vec{y}_n - \vec{y}(t_n)\| = \lambda_1\|\vec{e}_{n,h}\|, \quad (2) \le \lambda_2\|\vec{y}_{n+1} - \vec{y}(t_{n+1})\| = \lambda_2\|\vec{e}_{n+1,h}\|$$

Take $\lambda = \max(\lambda_1,\lambda_2)$.

$$\|\vec{e}_{n+1,h}\| \le \|\vec{e}_{n,h}\| + \tfrac{1}{2}h\lambda\left(\|\vec{e}_{n,h}\| + \|\vec{e}_{n+1,h}\|\right) + ch^3$$
$$\left(1 - \tfrac{1}{2}h\lambda\right)\|\vec{e}_{n+1,h}\| \le \left(1 + \tfrac{1}{2}h\lambda\right)\|\vec{e}_{n,h}\| + ch^3$$

We are interested in $h \to 0$ so assume that $h\lambda < 2$.

$$\|\vec{e}_{n+1,h}\| \le \left(\frac{1+\frac{1}{2}h\lambda}{1-\frac{1}{2}h\lambda}\right)\|\vec{e}_{n,h}\| + \left(\frac{c}{1-\frac{1}{2}h\lambda}\right)h^3$$

Claim:

$$\|\vec{e}_{n,h}\| \le \frac{c}{\lambda}\left(\left(\frac{1+\frac{1}{2}h\lambda}{1-\frac{1}{2}h\lambda}\right)^n - 1\right)h^2$$

Proof: By induction on n. <u>$n=0$</u>: $\|\vec{e}_{0,h}\| = 0$ ✓

<u>$n>0$</u>: assume claim is true up to n. True for $n+1$?

$$\|\vec{e}_{n+1,h}\| \le \left(\frac{1+\frac{1}{2}h\lambda}{1-\frac{1}{2}h\lambda}\right)\|\vec{e}_{n,h}\| + \left(\frac{c}{1-\frac{1}{2}h\lambda}\right)h^3$$

2016 01 11

$$\leq \frac{c}{\lambda}\left(\left(\frac{1+\frac{1}{2}h\lambda}{1-\frac{1}{2}h\lambda}\right)^{n+1}\right)h^2 - \frac{c}{\lambda}\left(\frac{1+\frac{1}{2}h\lambda}{1-\frac{1}{2}h\lambda}\right)h^2 + \left(\frac{c}{1-\frac{1}{2}h\lambda}\right)h^3$$

$$= \frac{c}{\lambda}\left(\left(\frac{1+\frac{1}{2}h\lambda}{1-\frac{1}{2}h\lambda}\right)^{n+1} - 1\right)h^2 \qquad \square$$

We have

$$\|\vec{e}_{n,h}\| \leq \frac{c}{\lambda}\left(\left(\frac{1+\frac{1}{2}h\lambda}{1-\frac{1}{2}h\lambda}\right)^{n} - 1\right)h^2$$

where we have assumed that $h\lambda < 2$. So

$$\frac{1+\frac{1}{2}h\lambda}{1-\frac{1}{2}h\lambda} = 1 + \frac{h\lambda}{1-\frac{1}{2}h\lambda} \leq \underbrace{\sum_{\ell=0}^{\infty} \frac{1}{\ell!}\left(\frac{h\lambda}{1-\frac{1}{2}h\lambda}\right)^{\ell}}_{\text{Taylor series expansion of } \exp\left(\frac{h\lambda}{1-\frac{1}{2}h\lambda}\right)}$$

$$\Rightarrow \|\vec{e}_{n,h}\| \leq \frac{ch^2}{\lambda}\left(\left(\frac{1+\frac{1}{2}h\lambda}{1-\frac{1}{2}h\lambda}\right)^{n} - 1\right)$$

$$\leq \frac{ch^2}{\lambda}\left(\frac{1+\frac{1}{2}h\lambda}{1-\frac{1}{2}h\lambda}\right)^{n}$$

$$\leq \frac{ch^2}{\lambda}\exp\left(\frac{nh\lambda}{1-\frac{1}{2}h\lambda}\right) \qquad \forall n \geq 0 \text{ st } nh \leq t^*$$

$$\leq \frac{ch^2}{\lambda}\exp\left(\frac{t^*\lambda}{1-\frac{1}{2}h\lambda}\right).$$

$\lim_{\substack{h\to 0 \\ 0\leq nh\leq t^*}} \|\vec{e}_{n,h}\| = 0 \longrightarrow$ method is convergent $\blacksquare$

	Euler's Method	Trapezium Rule
scheme	$\vec{y}_{n+1} = \vec{y}_n + h\vec{f}(t_n, \vec{y}_n)$	$\vec{y}_{n+1} = \vec{y}_n + \frac{1}{2}h\left(\vec{f}(t_n, \vec{y}_n) + \vec{f}(t_{n+1}, \vec{y}_{n+1})\right)$
order	1	2
convergent	✓	✓
type	explicit	implicit

Explicit vs Implicit

Problem: $\vec{y}' = A\vec{y}$, $A \in \mathbb{R}^{d\times d}$ matrix, $\vec{y} \in \mathbb{R}^d$

Euler's Method: $\vec{y}_{n+1} = \vec{y}_n + hA\vec{y}_n = (I + hA)\vec{y}_n$

Trapezium rule: $\vec{y}_{n+1} = \vec{y}_n + \frac{1}{2}hA\vec{y}_n + \frac{1}{2}hA\vec{y}_{n+1}$

$\rightarrow (I - \frac{1}{2}hA)\vec{y}_{n+1} = (I + \frac{1}{2}hA)\vec{y}_n$

θ-methods

on $[t_n, t_{n+1}] \rightarrow \vec{y}' \approx \vec{f}(t_n, \vec{y}_n) \rightarrow$ Euler

$\vec{y}' \approx \frac{1}{2}(\vec{f}(t_n, \vec{y}_n) + \vec{f}(t_{n+1}, \vec{y}_{n+1})) \rightarrow$ Trap. rule

$\vec{y}' \approx \theta \vec{f}(t_n, \vec{y}_n) + (1-\theta)\vec{f}(t_{n+1}, \vec{y}_{n+1}), \quad \theta \in [0,1]$

θ-method:

$$\vec{y}_{n+1} = \vec{y}_n + h(\theta \vec{f}(t_n, \vec{y}_n) + (1-\theta)\vec{f}(t_{n+1}, \vec{y}_{n+1}))$$

if $\theta = 1 \rightarrow$ Euler method

$\theta = \frac{1}{2} \rightarrow$ Trap. rule

Order of θ-method:

$$\vec{y}(t_{n+1}) - \vec{y}(t_n) - h(\theta \vec{f}(t_n, \vec{y}(t_n)) + (1-\theta)\vec{f}(t_{n+1}, \vec{y}(t_{n+1})))$$ (Taylor series)

$$= (\theta - \tfrac{1}{2})h^2 \vec{y}''(t_n) + (\tfrac{1}{2}\theta - \tfrac{1}{3})h^3 \vec{y}'''(t_n) + O(h^4)$$

if $\theta = \frac{1}{2} \longrightarrow$ order 2

$\theta \neq \frac{1}{2} \longrightarrow$ order 1

So if $\theta = \frac{1}{2} \longrightarrow 2^{nd}$ order

$\theta = 1 \longrightarrow$ Explicit method

So what about other θ?

Also interesting: $\theta = 0$

$$\vec{y}_{n+1} = \vec{y}_n + h\vec{f}(t_{n+1}, \vec{y}_{n+1})$$

$\hookrightarrow$ Backward Euler (Implicit Euler)

- very good for stiff ODEs