Numerical Methods for Computational Science and Engineering

Autumn Semester 2018, Week 10

Prof. Rima Alaifari, SAM, ETH Zurich

7. Numerical solution of ODEs

ordinary diff. eqns

First order system of ODEs

$$\frac{dy_1}{dt} = f_1(t, y_1, \dots, y_d)$$

$$\frac{dy_2}{dt} = f_2(t, y_1, \dots, y_d)$$

$$\vdots \qquad \vdots$$

$$\frac{dy_d}{dt} = f_d(t, y_1, \dots, y_d)$$

or more compactly

$$\frac{d\underline{y}}{dt} = \underline{f}(t, \underline{y})$$

Unknowns: $y_1, \dots, y_d$: scalar functions of real variable t "time"

Given: RHS $f_1, \dots, f_d$ functions of $d+1$ variables $t, y_1, \dots, y_d$

continuous

$$\underline{f} : I \times D \to \mathbb{R}^d \quad \text{of time } t \text{ and state } \underline{y}$$

$D \subset \mathbb{R}^d$: "state space", $I \subset \mathbb{R}$ finite time interval

Notation: $\frac{d\underline{y}}{dt} = \dot{\underline{y}}$

Definition of solution:

A solution to ODE $\dot{\underline{y}} = \underline{f}(t, \underline{y})$, where $\underline{f}$ is continuous, is a function $\underline{y}: J \subset I \to D$, $\underline{y} \in C^1$ s.t. $\dot{\underline{y}} = \underline{f}(t, \underline{y})$ holds for all $t \in J$.

Smoothness of solutions:

$$\underline{f} \in C^m \Rightarrow \underline{y} \in C^{m+1}$$

Each solution $\underline{y}(t)$ parametrizes a curve $C \subset \mathbb{R}^d$
"trajectory/orbit of the system"

Example: predator – prey model

(simplified ecological model for 2 species, $d=2$)

$u(t)$ ... number of prey at time t

$v(t)$ ... number of predators at time t

Population growth model (Thomas Malthus, 1798)

population of species grows roughly proportional to its size:

$$\dot{u} = \varrho \cdot u$$
$$\dot{v} = \sigma \cdot v$$

growth rates ϱ, σ may depend on the other species

(more prey $\Rightarrow$ predators reproduce faster

more predators $\Rightarrow$ prey are consumed faster)

Simplifying assumption: unlimited resources of prey

Simplest model: Lotka–Volterra model

$\alpha, \beta, \delta, \mu > 0$

$$(*) \quad \begin{aligned} \dot{u} &= \alpha u - \beta u v = (\alpha - \beta v) u \\ \dot{v} &= -\mu v + \delta u v = (-\mu + \delta u) v \end{aligned}$$

growth rates: $\varrho = \alpha - \beta v$

$\sigma = -\mu + \delta u$

$$\underline{y} = \begin{bmatrix} u \\ v \end{bmatrix} \qquad \underline{f}(\underline{y}) = \begin{bmatrix} (\alpha - \beta v) u \\ (-\mu + \delta u) v \end{bmatrix}$$

$$\Leftrightarrow \quad \dot{\underline{y}} = \underline{f}(\underline{y})$$

Note:

① $v = 0 \Rightarrow \dot{u} = \alpha u \Rightarrow u(t) = u(0) e^{\alpha t}$

(no predators → prey increase exp. with growth rate α)

$u = 0 \Rightarrow \dot{v} = -\mu v \Rightarrow v(t) = v(0) e^{-\mu t}$

(no prey ⇒ predators decrease exp. with decrease rate μ)

② (*) is a system of ODEs for which the RHS $\underline{f}$ is independent of the time variable t

$$\dot{\underline{y}} = \underline{f}(\underline{y})$$

Such systems are called <u>autonomous</u> ODEs.

(3) Equilibrium points of this system?

i.e. when is the solution constant $\underline{y}(t) = \underline{y}^*$

$\Rightarrow \quad \underline{f}(\underline{y}^*) = 0$

$\Rightarrow \quad (\alpha - \beta v)\, u = 0$

$(-\mu + \delta u)\, v = 0$

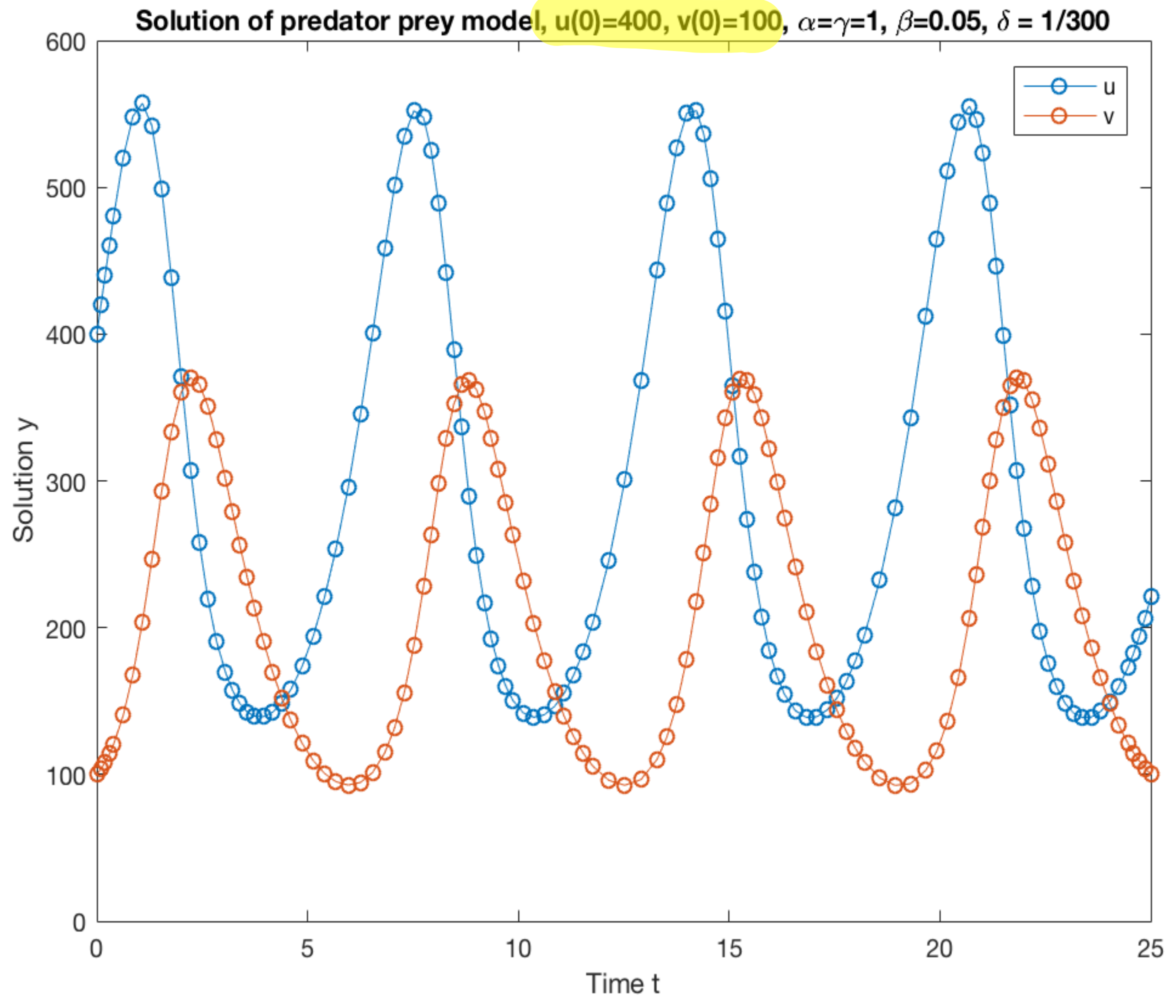
a.) $\alpha = \beta v, \ \mu = \delta u \quad \Leftrightarrow \quad v_2^* = \frac{\alpha}{\beta}, \quad u_2^* = \frac{\mu}{\delta}$

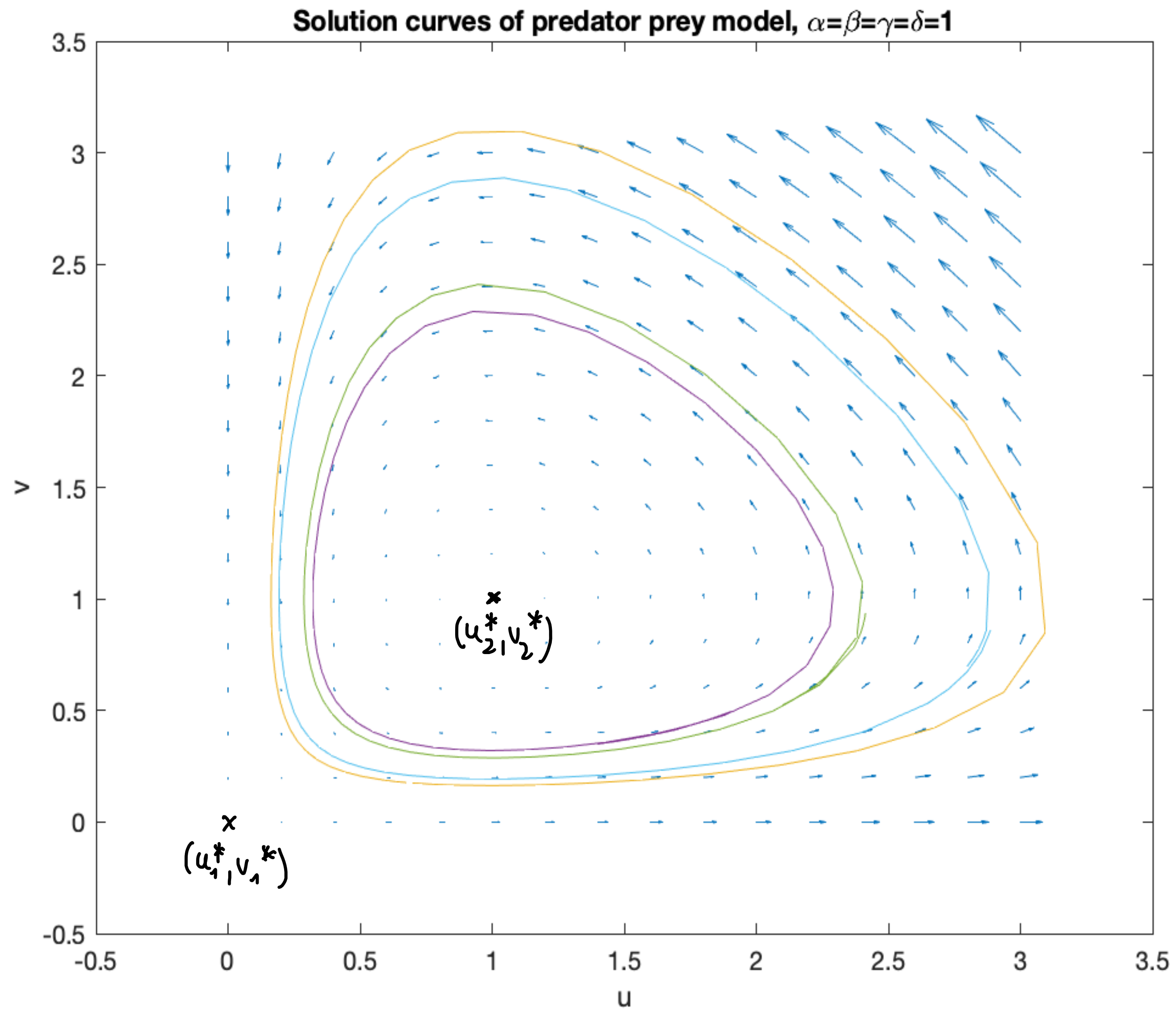
(non-trivial : birth rate of prey is precisely sufficient to continuously feed the predators)

b.) $u_1^* = 0, \quad v_1^* = 0$

(no animals at all)

2 initial values





$(u^*_{1/2}, v^*_{1,2})$ ... stationary points

(f describes movement of particles in this velocity field)

The solution depends on the initial size of the population $u(0), v(0)$

$\rightarrow$ to solve an ODE uniquely, we need additional conditions: Initial value problem (IVP)

i.e. ODE + initial conditions at start time $(t=0)$.

Higher order systems

Example: van der Pol equation

- modeling of electric circuits
- chemical reactions
- wind-induced motions of structures

$$\boxed{\ddot{u}(t) + (u^2 - 1)\dot{u}(t) + u = g(t)}$$

1 eqn
2^{nd} order

Convert this to a system of first orders?

$$y_1 = u$$

$$y_2 = \dot{u}$$

$$\Rightarrow \boxed{\begin{aligned} \dot{y}_1 &= y_2 \\ \dot{y}_2 &= g - (y_1^2 - 1)\, y_2 - y_1 \end{aligned}}$$

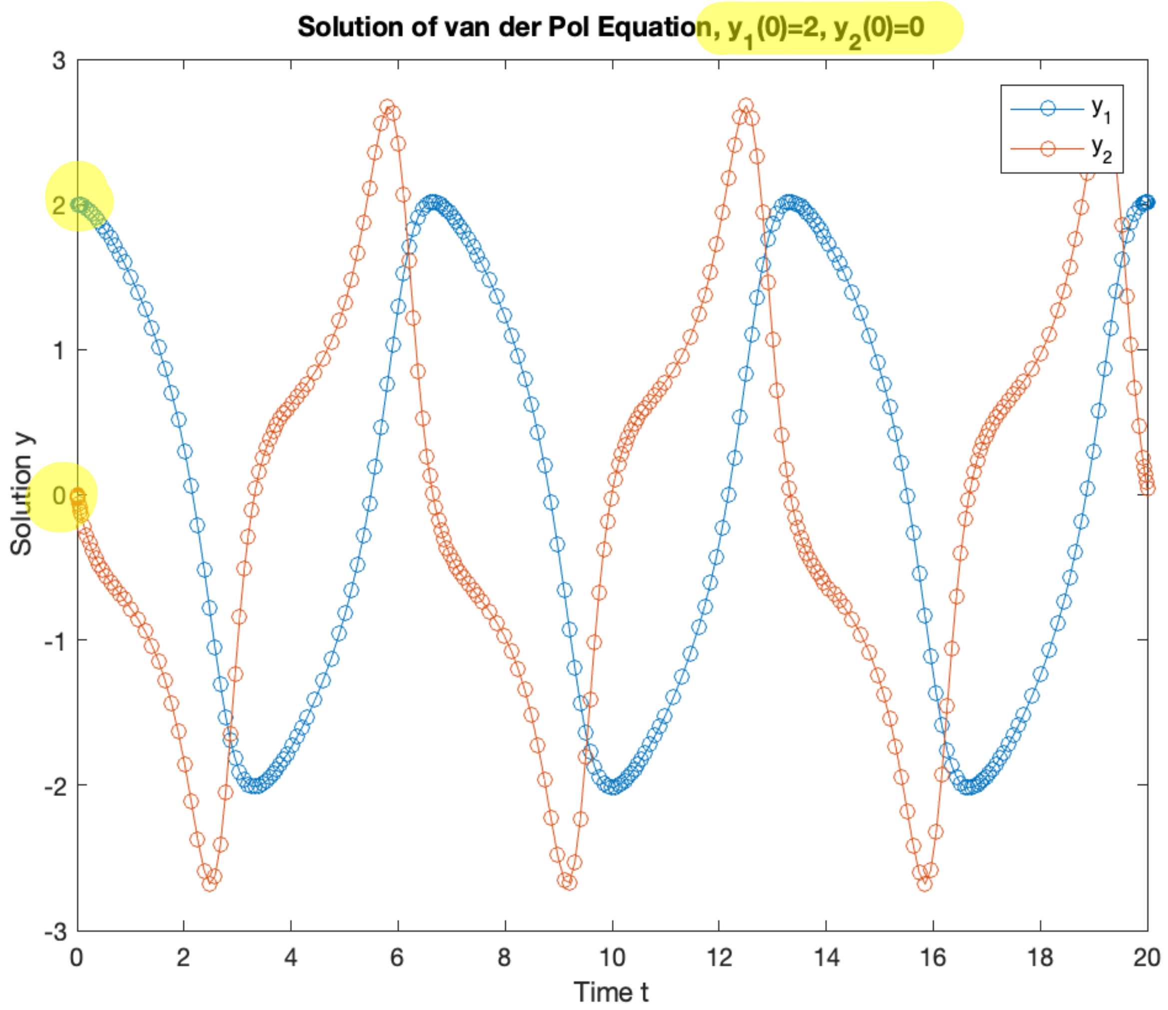
2 eqns
1^{st} order

and we can define $f_1(t, y_1, y_2) = y_2$

$$f_2(t, y_1, y_2) = g - (y_1^2 - 1)\, y_2 - y_1$$

If g is not constant, then this system is non-autonomous.

Solution example $(g(t) = 0)$ of an IVP



$\rightarrow$ 2 initial conditions!

Two general remarks:

① Any higher order equation/system can be transformed to a system of first order ODEs.

$$(**) \quad \underline{y}^{(n)} = \underline{f}(t, \underline{y}, \dot{\underline{y}}, \dots, \underline{y}^{(n-1)}) \qquad \underline{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix}$$

Extend the system by adding additional components: all the derivatives up to order $n-1$

$$\underline{z}(t) := \begin{bmatrix} \underline{y}(t) \\ \underline{y}^{(1)}(t) \\ \vdots \\ \underline{y}^{(n-1)}(t) \end{bmatrix} = \begin{bmatrix} \underline{z}_1 \\ \underline{z}_2 \\ \vdots \\ \underline{z}_n \end{bmatrix} \qquad \underline{z}(t) \in \mathbb{R}^{n \cdot d}$$

$$(**) \iff \dot{\underline{z}} = \underline{g}(\underline{z}) \quad \text{with } \underline{g}(\underline{z}) = \begin{bmatrix} \underline{z}_2 \\ \underline{z}_3 \\ \vdots \\ \underline{z}_n \\ \underline{f}(t, \underline{z}_1, \dots, \underline{z}_n) \end{bmatrix}$$

Note: This extended system requires that we specify initial conditions for the first $n-1$ derivatives:

$$\underline{y}(t_0), \dot{\underline{y}}(t_0), \dots, \underline{y}^{(n-1)}(t_0)$$

$\Rightarrow$ $n \cdot d$ initial values

② Autonomization

Recall: $\dot{\underline{y}} = \underline{f}(\underline{y})$ (as in predator-prey model)

is called an autonomous ODE

Any non-autonomous ODE can be transformed to an equivalent autonomous ODE.

How? By extending the system by one more variable

→ introduce an extra coordinate $y_0 = t$ to represent time

y_0 satisfies:
- $\dot{y}_0(t) \left(= \frac{dy_0}{dt} \right) = 1$
- initial condition $y_0(t_0) = t_0$

Thus we can write

$$\boxed{\begin{aligned} \dot{y}_1(t) &= f_1(t, y_1, \dots, y_d) \\ \dot{y}_2(t) &= f_2(t, y_1, \dots, y_d) \\ &\vdots \\ \dot{y}_d(t) &= f_d(t, y_1, \dots, y_d) \end{aligned}} \iff \boxed{\begin{aligned} \dot{y}_0(t) &= 1 \\ \dot{y}_1(t) &= f_1(y_0, y_1, \dots, y_d) \\ \dot{y}_2(t) &= f_2(y_0, y_1, \dots, y_d) \\ &\vdots \\ \dot{y}_d(t) &= f_d(y_0, y_1, \dots, y_d) \end{aligned}}$$

non-autonomous system — equivalent auton. system

Example: Autonomous form of the van der Pol equation:

original:
$$\dot{y}_1 = y_2$$
$$\dot{y}_2 = g(t) - (y_1^2 - 1)y_2 - y_1$$

autonomous form:

Introduce $y_0 = t$ (linear function in t)

$\dot{y}_0 = 1$

$\dot{y}_1 = y_2$

$\dot{y}_2 = \underbrace{g(y_0) - (y_1^2 - 1)y_2 - y_1}_{\text{function in } y_0, y_1, y_2}$

<u>Altogether</u>: It suffices to consider <u>autonomous first order</u> IVPs.

Remark: IVPs for autonomous ODEs:

initial time does not play a role

canonical choice $t = 0$.

Existence & uniqueness of solutions:

Recall from Analysis:

If $\underline{f}(t, \underline{y})$ is a differentiable function, then the IVP

$$\dot{\underline{y}} = \underline{f}(t, \underline{y}), \quad \underline{y}(t_0) = \underline{y}_0$$

admits a unique solution $\underline{y}$ defined on a <u>maximal</u> domain J, $t_0 \in J$.

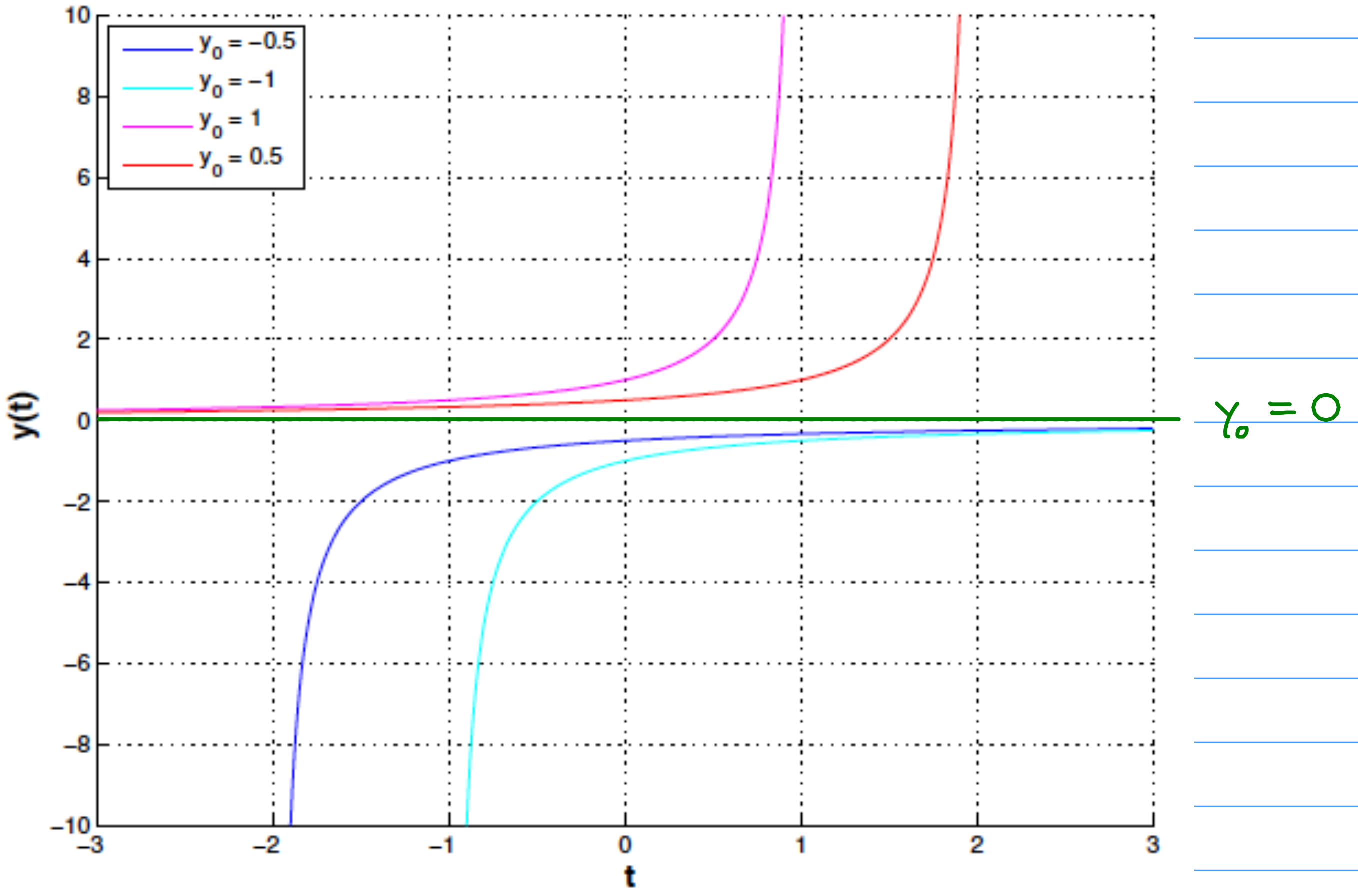
J depends on $(t_0, \underline{y}_0)$, i.e. $J = J(t_0, \underline{y}_0)$

Example: $\dot{y} = y^2 \qquad y(0) = y_0$

(autonomous)

has the solution

$$y(t) = \begin{cases} \dfrac{1}{y_0^{-1} - t} & \text{if } y_0 \neq 0 \\ 0 & \text{if } y_0 = 0 \end{cases}$$



$$J(0, Y_0) = J(Y_0) = \begin{cases} \mathbb{R} & \text{if } Y_0 = 0 \\ (-\infty, Y_0^{-1}) & \text{if } Y_0 > 0 \\ (Y_0^{-1}, \infty) & \text{if } Y_0 < 0 \end{cases}$$

(Maximal domain of definition of the solution depends on initial value Y_0)

Evolution operator

Consider $\dot{\underline{y}} = \underline{f}(\underline{y})$

and assume that $\forall\, \underline{y}_0 \in D$ the unique solution $\underline{y}$ is global, i.e. exists $\forall\, t \in \mathbb{R}$.

Definition 8.1.5 (Evolution operator/mapping). Under assumption 8.1.1 the mapping

$$\boldsymbol{\Phi} : \begin{cases} \mathbb{R} \times D & \mapsto & D \\ (t, \mathbf{y}_0) & \mapsto & \boldsymbol{\Phi}^t \mathbf{y}_0 := \mathbf{y}(t) \end{cases},$$

where $t \mapsto \mathbf{y}(t) \in C^1(\mathbb{R}, \mathbb{R}^d)$ is the unique (global) solution of the IVP $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$, $\mathbf{y}(0) = \mathbf{y}_0$, is the *evolution operator*/mapping for the autonomous ODE $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$.

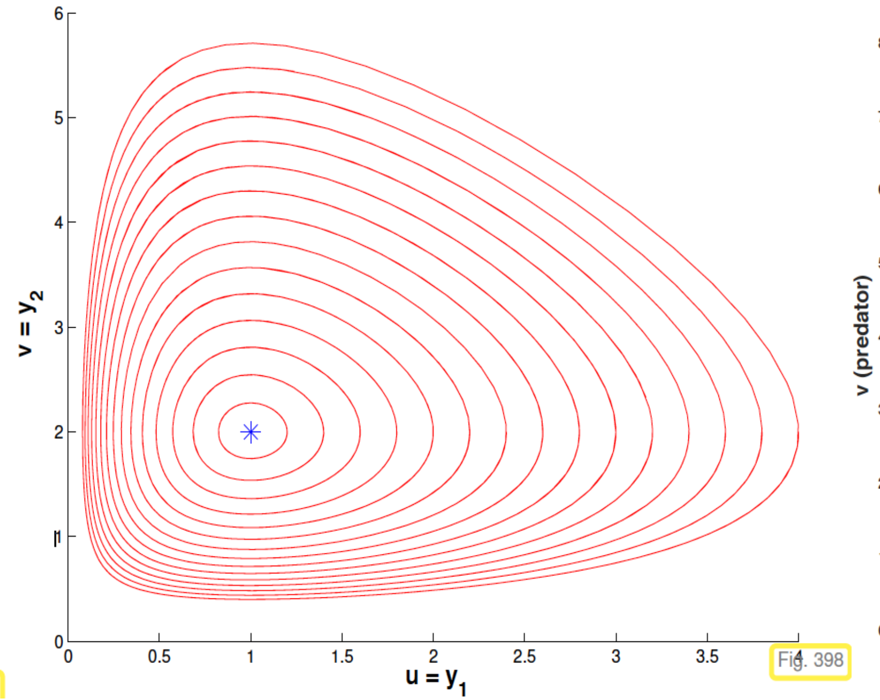
Note: • By definition $\frac{\partial \Phi}{\partial t}(t, \underline{y}_0) = \underline{f}(\Phi^t \underline{y}_0)$

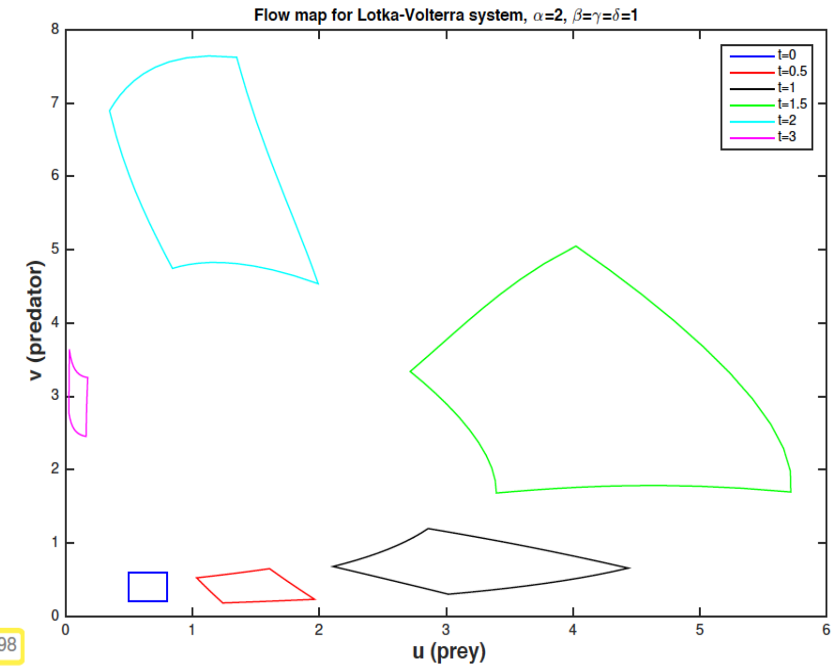
• Φ encodes the complete set of solutions of

$$\dot{\underline{y}} = \underline{f}(\underline{y})$$

$t \mapsto \Phi^t y_0$ trajectory

$y \mapsto \Phi^t y$ state mapping





trajectories

$t \mapsto \Phi^t y_0$

state mapping

$y \mapsto \Phi^t y$

Each curve corresponds to one choice y_0

In practice: ODEs very often do not have a closed form solution

Next: Find a numerical / approximate solution

Polygonal Approximation Methods

For the ODE $\dot{y} = f(y)$ we want an approximate model for Φ (evolution operator)

on a temporal mesh $\mathcal{M} = \{0 = t_0 < t_1 < t_2 < \ldots < t_N = T\}$

We start with a simple idea: Taylor approximation

$$y(t) \approx y(t_k) + (t - t_k)\,\dot{y}(t_k) = y(t_k) + (t - t_k)\, f(y(t_k))$$

$$\Rightarrow y(t_{k+1}) \approx y(t_k) + (t_{k+1} - t_k) f(y(t_k))$$

Starting from initial condition

$$y(0) = y_0$$

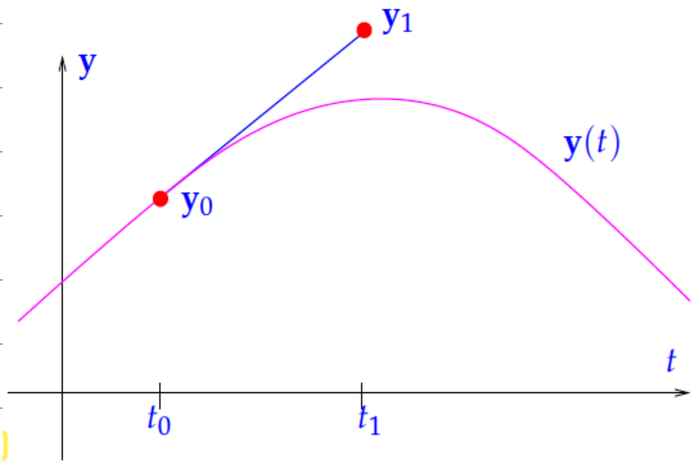
we can define y_k, the approximation of $y(t_k)$, iteratively as:

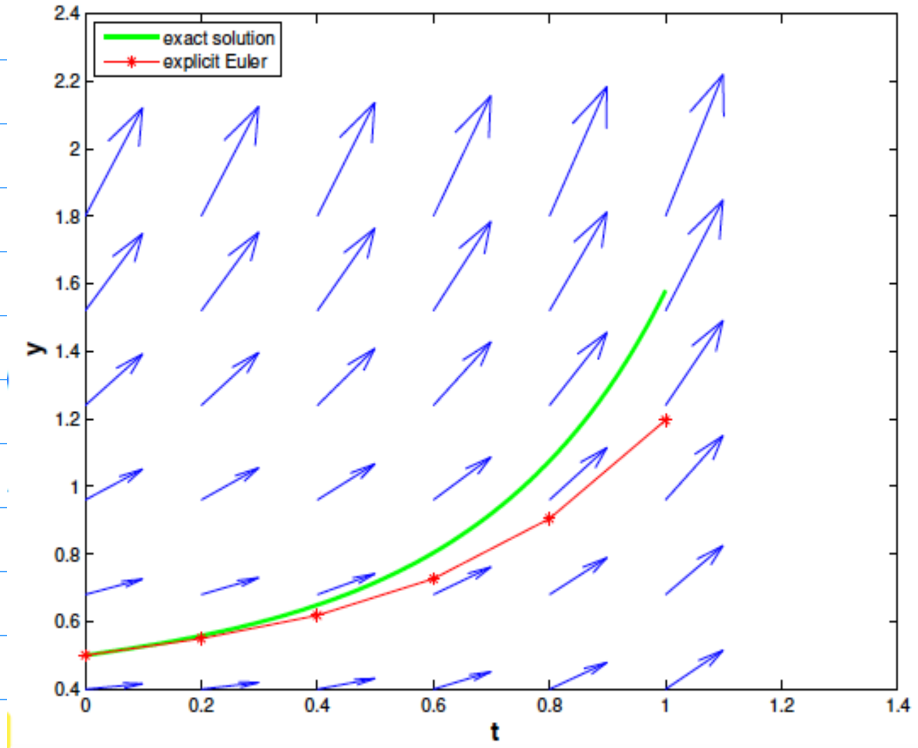
$$\boxed{y_{k+1} = y_k + \underbrace{(t_{k+1} - t_k)}_{=: h_k} f(y_k)}$$

Explicit Euler method

With equidistant mesh:

$$y_{k+1} = y_k + h f(y_k)$$





Example: $\dot{y} = y \qquad y(0) = 1 \qquad (*)$

(We know the explicit solution: $y(t) = \exp(t)$)

Explicit Euler with uniform step size h:

$$\begin{aligned} y_{k+1} &= y_k + h \cdot y_k = (1+h) y_k = (1+h)^2 y_{k-1} \\ &= \dots = (1+h)^{k+1} y(0) \\ &= (1+h)^{k+1} \end{aligned}$$

$\Rightarrow \quad Y_k = (1+h)^k$

$t_k = hk \qquad y(t_k) = \exp(t_k) = \exp(hk)$

$$\approx (1+h)^k = Y_k$$

$$(1+h)^{t_k/h} = Y_k$$

Formula from calculus:

$$\exp(t) = \lim_{h \to 0} (1+h)^{t/h}$$

So indeed: for small h, the explicit Euler scheme approximates the solution of (*).

Error $\quad e_{t_k} := Y_k - y(t_k) = (1+h)^{t_k/h} - \exp(t_k)$

h	$\lvert e_1\rvert$	$\lvert e_2\rvert$	$\lvert e_3\rvert$
0.1	0.125	0.662	2.636
0.01	0.0134	0.0730	0.297
0.001	0.00135	0.00738	0.0301
0.0001	0.000136	0.000739	0.00301

error at times $t = 1, 2, 3$

- for fixed h: the further we proceed in time, the larger the error
- the smaller h, the smaller the error at a fixed time t

BUT: smaller step size $\Rightarrow$ more steps are needed to get to time t as h decreases

$\rightsquigarrow$ increase in computational effort

- Error is proportional to the step size

Decreasing h by a factor $\frac{1}{10}$

(roughly) $\Rightarrow$ error decreases by factor $\frac{1}{10}$

(and complexity increases by factor 10)

$\rightarrow$ linear dependence of error on step size

$$|e_{t_k}| = |y_k - y(t_k)| \leq \underbrace{C(t_k)}_{\text{function of time \& initial condition, but independent of } h} \cdot h^1$$

Methods with such a linear dependence:

First-order methods

(Explicit Euler: First-order method)

Implicit Euler Method

Recall form of explicit Euler:

$$y_{k+1} = y_k + h\, f(y_k)$$

This can interpreted as a forward difference approximation:

$$y'(t_k) \approx \frac{y(t_{k+1}) - y(t_k)}{t_{k+1} - t_k} \qquad (h = t_{k+1} - t_k)$$

Alternatively, one can use backward difference:

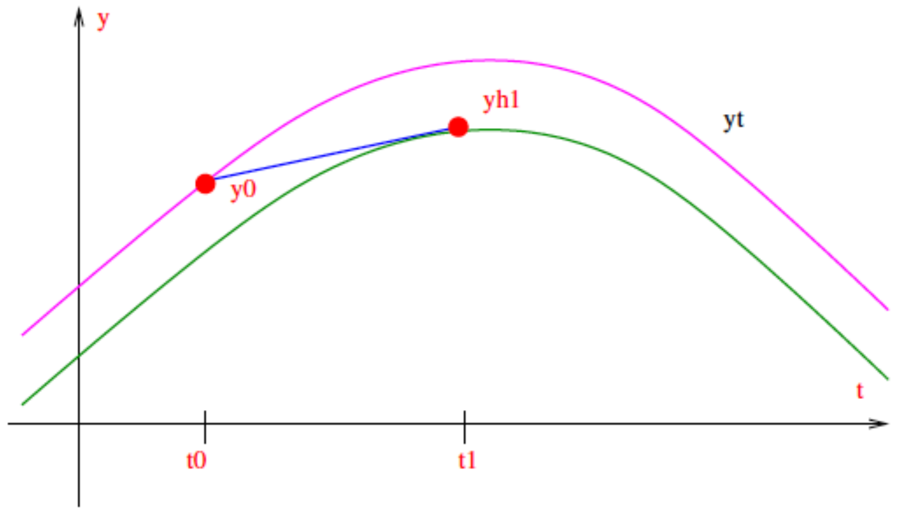
$$y'(t_{k+1}) \approx \frac{y(t_{k+1}) - y(t_k)}{t_{k+1} - t_k}$$

$$\left[\, y'(t_k) \approx \big(y(t_k) - y(t_{k-1})\big) \,/\, \big(t_k - t_{k-1}\big) \,\right]$$

$\Rightarrow$ $$\boxed{y_{k+1} = y_k + h f(y_{k+1})}$$

Implicit Euler Method

as a new scheme



Fundamental difference: RHS (for implicit Euler) depends on y_{k+1} which is not known at step k.

$\Rightarrow$ requires solution of a d-dimensional (possibly non-linear) system of equations! At each iteration step!

(Remark: solving non-linear systems will be discussed in Chapter 8 $\rightarrow$ e.g. Newton's method)

Benefits of such "implicit" methods? Better stability properties for stiff IVPs ($\rightsquigarrow$ will be discussed later)

Implicit midpoint method

Idea: central difference quotient as an approach

$$y'\left(\frac{1}{2}(t_k+t_{k+1})\right) \approx \frac{y(t_{k+1})-y(t_k)}{t_{k+1}-t_k}$$

$$\Rightarrow \; y(t_{k+1}) \approx y(t_k) + h \cdot \underline{f}\left(y\left(\frac{t_k+t_{k+1}}{2}\right)\right)$$

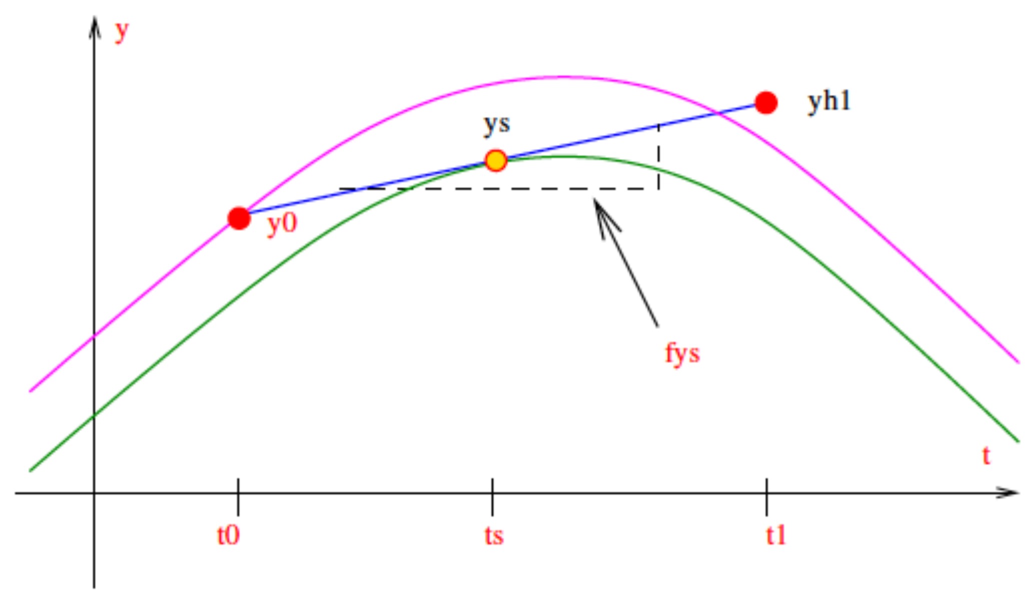
BUT: requires approximation of $y\left(\frac{t_k+t_{k+1}}{2}\right)$

$$\Rightarrow \; y\left(\frac{t_k+t_{k+1}}{2}\right) \approx \frac{1}{2}\left(y(t_k)+y(t_{k+1})\right)$$

$$\Rightarrow \; \boxed{y_{k+1} = y_k + h \cdot \underline{f}\left(\frac{1}{2}(y_k+y_{k+1})\right)}$$

Again: implicit method

Implicit midpoint method



Remark: Solutions of the non-linear systems in implicit Euler & implicit method exist, if h is sufficiently small.

Remark: expl. / impl. Euler & implicit midpoint method have in common that the new iterate y_{k+1} only involves the current state y_k.

such methods are called Single Step Methods

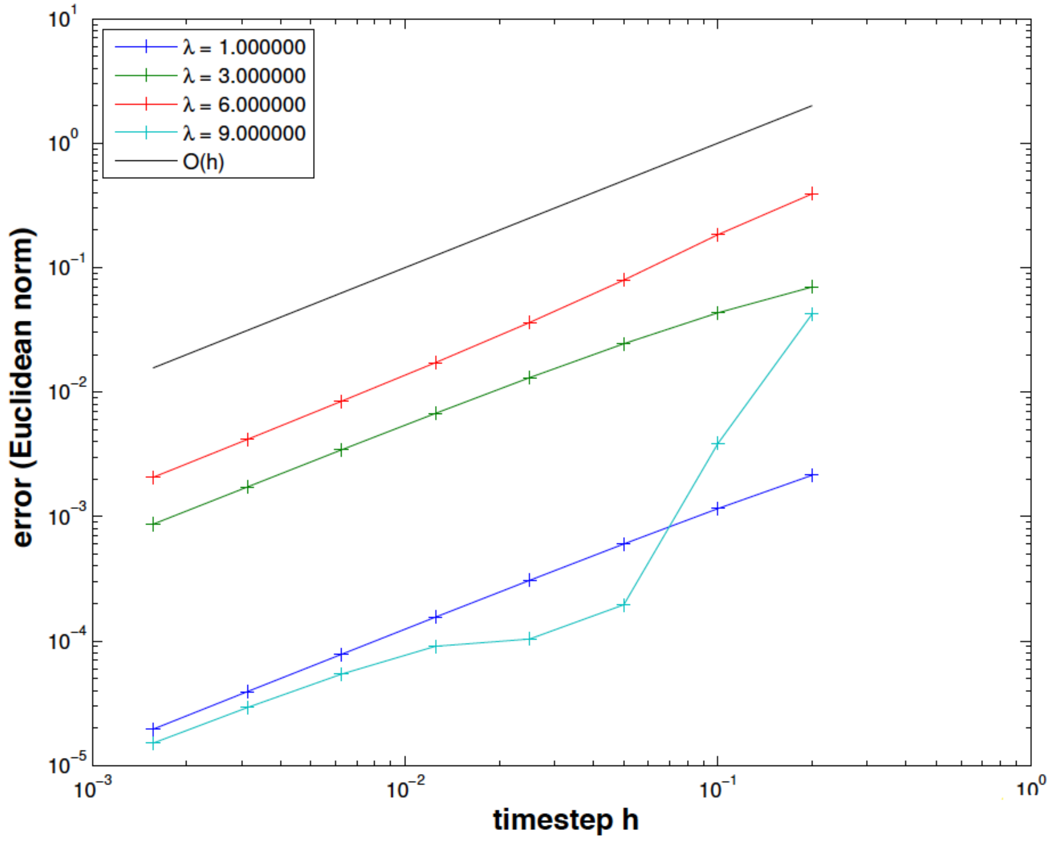
$[y_0, y_1, \dots, y_N]$ approximation of $[y(t_0), y(t_1), \dots, y(t_N)]$

Convergence rates of these 3 methods:

- all algebraic
- the two Euler methods are first-order
- the implicit midpoint method is second-order

Example: $\dot{y} = \lambda y(1-y)$, $y(0) = 0.01$

consider error $|e_1|$ at final time $t = 1$

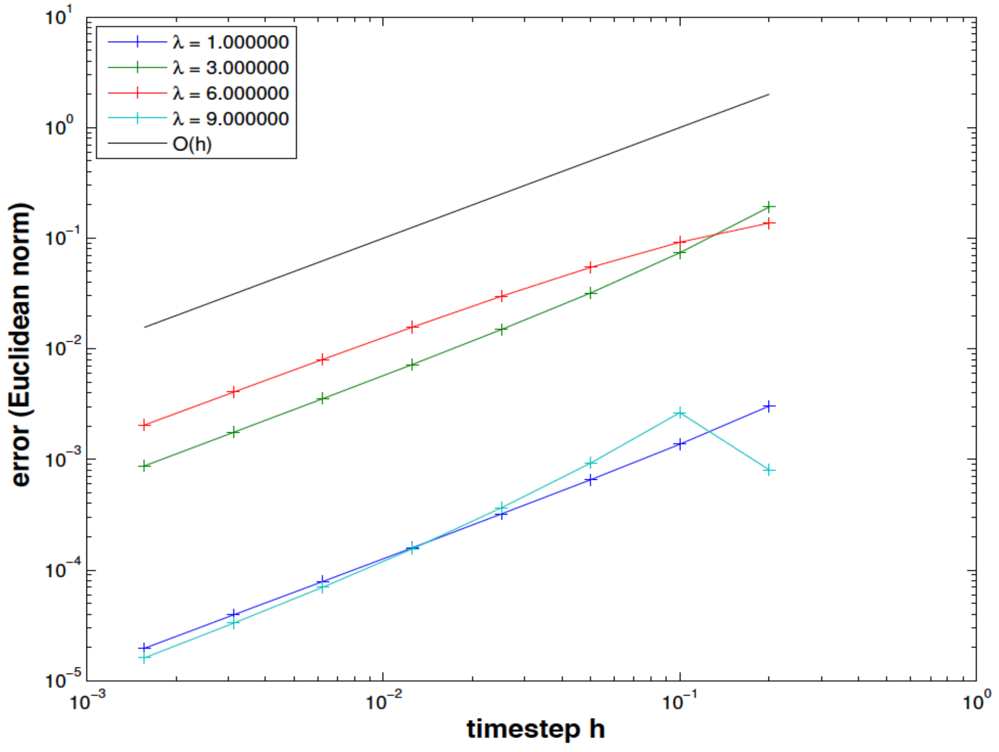


explicit Euler method

log-log-plot

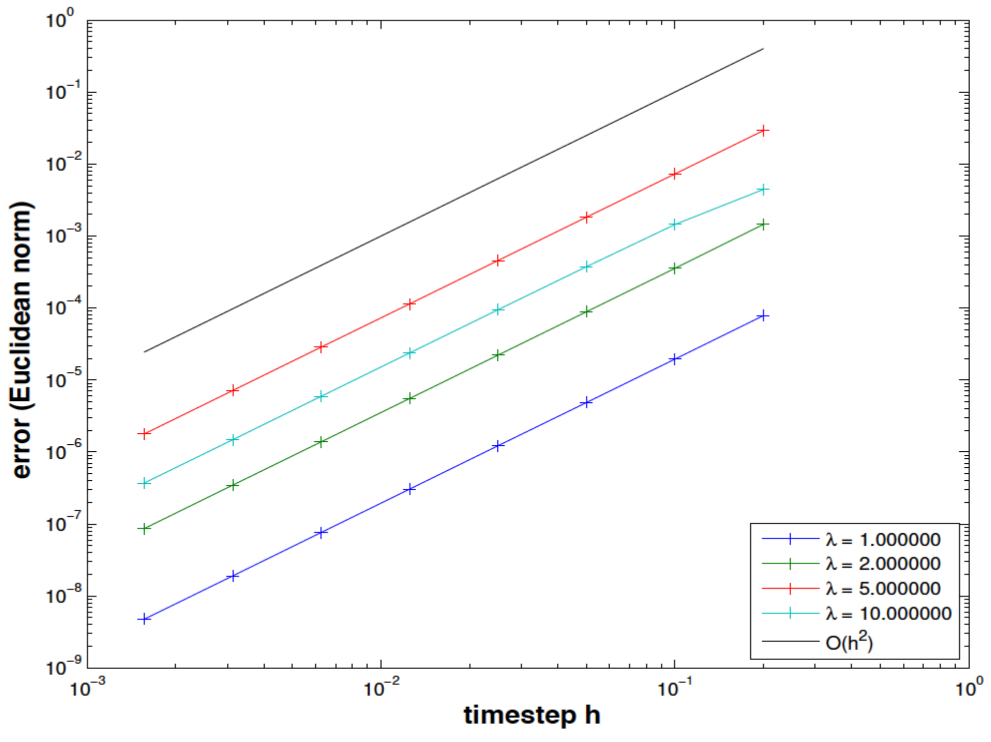
algebraic

$\mathcal{O}(h)$



implicit Euler method

$\mathcal{O}(h)$



implicit midpoint method

$|e_1| = \mathcal{O}(h^2)$

$\Rightarrow$ implicit midpoint method is a second order method

$$|e_{t_k}| \leq C(t_k)\, h^2$$

General Single Step Methods

So far:

$$y_{k+1} = \underbrace{y_k + h_k f(y_k)}_{\Psi(h_k, y_k)} \qquad \text{[Explicit Euler]}$$

$$y_{k+1} = y_k + h_k f(y_{k+1}) \qquad \text{[Implicit Euler]}$$

$$y_{k+1} = y_k + h_k f\left(\tfrac{1}{2}(y_k + y_{k+1})\right) \qquad \text{[Implicit midpoint]}$$

General SSM:

$$y_{k+1} = \Psi(h_k, y_k) = \Psi^{h_k} y_k$$

$\uparrow$ discrete evolution operator (approximates the true evolution operator Φ)

Single step: Ψ only takes in state y_k

Analytical setting:

$$y(t_1) = \Phi^{h_1} y_0$$

$$y(t_{k+1}) = \Phi^{t_{k+1}} y_0 = \Phi^{h_k}[y(t_k)]$$

$$\Phi : I \times D \to \mathbb{R}^d$$

$D \subset \mathbb{R}^d$

$\overset{?}{\approx}$

Numerical setting

$$y_1 = \Psi^{h_0} y_0$$

$$y_{k+1} = \Psi^{h_k} y_k$$

$$\Psi : I \times D \to \mathbb{R}^d$$

Explicit Euler: $y_{k+1} = \Psi^{h_k} y_k = y_k + h_k f(y_k)$

$$\Psi^h y := y + h f(y)$$

Recall: $\frac{\partial \Phi}{\partial t}(t, y) = f(y)$

Basic requirement on Ψ:

At $h=0$: $\frac{d}{dh}\Psi^h = f(y)$?

Consistent discrete evolution

The discrete evolution $\mathbf{\Psi}$ defining a single step method according to definition 8.3.1 and (8.33) for the autonomous ODE $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$ invariably is of the form

$$\boxed{\mathbf{\Psi}^h\mathbf{y} = \mathbf{y} + h\boldsymbol{\psi}(h,\mathbf{y})} \quad \text{with} \quad \begin{array}{l} \boldsymbol{\psi} : I \times D \to \mathbb{R}^d \text{ continuous,} \\ \boldsymbol{\psi}(0,\mathbf{y}) = \mathbf{f}(\mathbf{y}) . \end{array} \tag{8.34}$$

$f(y) =$

$$\frac{d}{dh}\Psi^h\Big|_{h=0} = 0 + \psi(0, y) + 0 \cdot \frac{d}{dh}\psi(0, y)$$

$$= \psi(0, y)$$

Note: Explicit Euler: $\psi(h, y) = f(y)$

General SSM: $\psi(0, y) = f(y)$

Example: Consistency of implicit Euler:

$$\Psi^h y = y + h \underbrace{f(\Psi^h y)}_{=\psi(h,y)}$$

$$\psi(0, y) = f(\underbrace{\Psi^0 y}_{=y}) = f(y) \quad \checkmark$$

Order of accuracy for SSMs

All SSMs converge algebraically

there is a $p \in \mathbb{N}$ such that the sequence $(\mathbf{y}_k)_k$ generated by the single step method for $\dot{\mathbf{y}} = \mathbf{f}(t, \mathbf{y})$ on a mesh $\mathcal{M} := \{t_0 < t_1 < \cdots < t_N = T\}$ satisfies

$$\boxed{\max_k \|\mathbf{y}_k - \mathbf{y}(t_k)\| \leq Ch^p} \quad \text{for} \quad h := \max_{k=1,\dots,N} |t_k - t_{k-1}| \to 0\,, \tag{8.37}$$

with $C > 0$ independent of $\mathcal{M}$

Definition 8.3.3 (Order of a single step method). The minimal integer $p \in \mathbb{N}$ for which (8.37) holds for a single step method when applied to an ODE with (sufficiently) smooth right hand side, is called the *order* of the method.