Numerical Methods for Differential Equations

Chapter 1: Initial value problems

Tony Stillfjord, Gustaf Söderlind

Numerical Analysis, Lund University

Contents

1. Initial value problems

Standard formulation of a system of ODEs

$$y' = f(t, y)\,; \qquad y(0) = y_0$$

with $f : \mathbb{R} \times \mathbb{R}^m \to \mathbb{R}^m$

Theorem *If $f(t, y)$ is continuous for $t \in [0, T]$ and satisfies the Lipschitz condition*

$$\|f(t, u) - f(t, v)\| \leq L[f] \cdot \|u - v\|$$

for all $u, v \in \mathbb{R}^m$ with Lipschitz constant $L[f] < \infty$, then there exists a unique solution to the initial value problem on $[0, T]$ for every initial value $y(0) = y_0$

Existence and uniqueness

Some problems *always* satisfy Lipschitz conditions on $\mathbb{R}^m$

Example A linear constant coefficient differential equation

$$\dot{y} = Ay; \qquad y(0) = y_0$$

has Lipschitz constant

$$L[f] = \max_{u \neq v} \frac{\|Au - Av\|}{\|u - v\|} = \max_{y \neq 0} \frac{\|Ay\|}{\|y\|} = \|A\|$$

The matrix norm $\|A\|$ is a Lipschitz constant for $f(y) = Ay$

Existence and uniqueness

Note *Most nonlinear problems do not satisfy a Lipschitz condition on all of $\mathbb{R}^m$*

Example The problem

$$\dot{y} = y^2; \qquad y(0) = y_0 > 0$$

has solution

$$y(t) = \frac{y_0}{1 - y_0 t}$$

The solution blows up at $t = 1/y_0$ ("Finite escape time")

Standard form $x' = F(t, x)$

Given $y'' = f(t, y, y')$ with $y(0) = y_0, \;\; y'(0) = y_0'$

Standard substitution Introduce *new variables*

$$\begin{aligned} x_1 &= y \\ x_2 &= y' \end{aligned}$$

to obtain a *system of first order equations*

$$\begin{aligned} x_1' &= x_2 \\ x_2' &= f(t, x_1, x_2) \end{aligned}$$

with $x_1(0) = y_0$ and $x_2(0) = y_0'$

2. The Explicit Euler method (1768)

Replace y' in $y' = f(t, y)$ by *finite difference approximation*

$$y'(t_n) \approx \frac{y(t_n + h) - y(t_n)}{h}$$

Let y_n denote the *numerical approximation* to $y(t_n)$ in

$$\frac{y_{n+1} - y_n}{h} = f(t_n, y_n), \quad y_0 = y(t_0)$$

Explicit Euler method Compute $\{y_n\}$ recursively from

$$y_{n+1} = y_n + hf(t_n, y_n)$$
$$t_{n+1} = t_n + h$$

Taylor series expansion

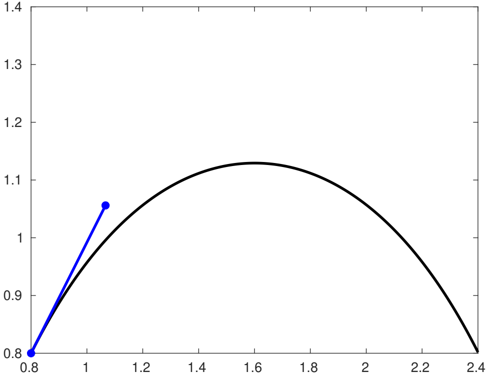
$$\begin{aligned} y(t+h) &= y(t) + hy'(t) + \frac{h^2}{2!}y''(\xi) \\ &= y(t) + hf(t, y(t)) + \mathcal{O}(h^2) \quad \Rightarrow \end{aligned}$$

$$y(t+h) \approx y(t) + hf(t, y(t))$$

Explicit Euler method obtained by dropping higher order terms

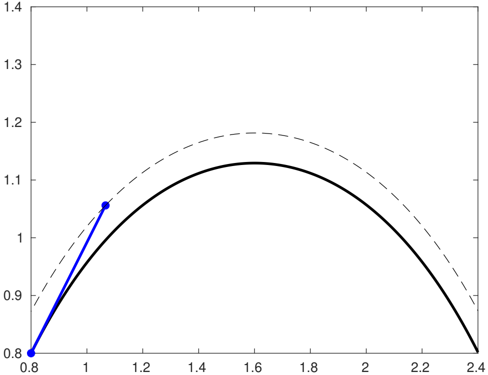
$$y_{n+1} = y_n + hf(t_n, y_n)$$

"Take a step of size h in the direction of the tangent"



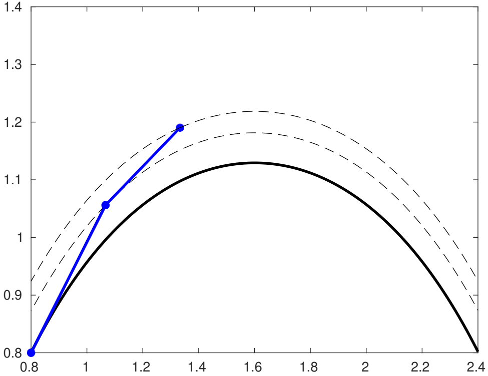
Ending up on a different solution trajectory (dashed curves), each step introduces a *local error*, eventually accumulating a large *global error*

"Take a step of size h in the direction of the tangent"



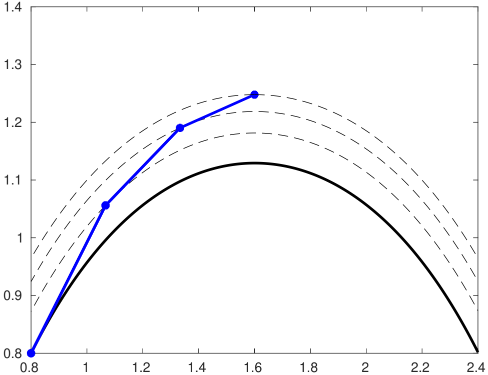
Ending up on a different solution trajectory (dashed curves), each step introduces a *local error*, eventually accumulating a large *global error*

"Take a step of size h in the direction of the tangent"



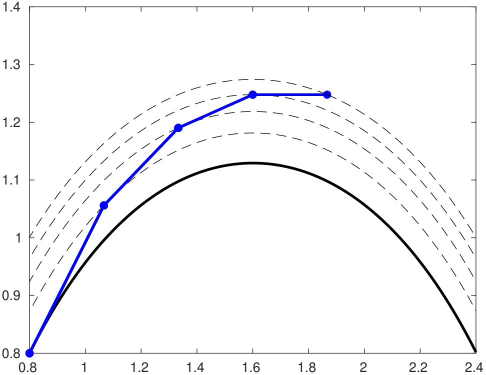
Ending up on a different solution trajectory (dashed curves), each step introduces a *local error*, eventually accumulating a large *global error*

"Take a step of size h in the direction of the tangent"



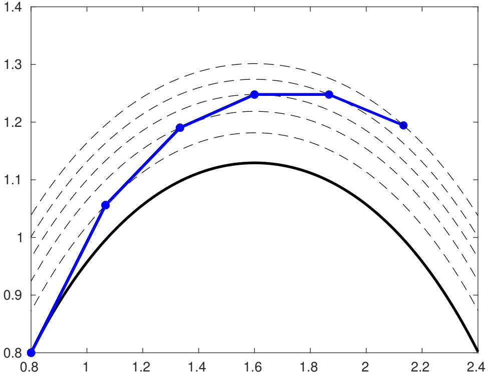
Ending up on a different solution trajectory (dashed curves), each step introduces a *local error*, eventually accumulating a large *global error*

"Take a step of size h in the direction of the tangent"



Ending up on a different solution trajectory (dashed curves), each step introduces a *local error*, eventually accumulating a large *global error*

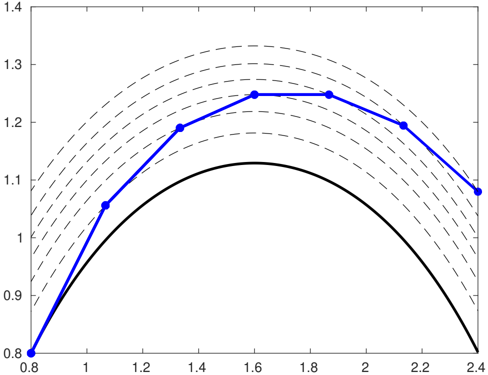
"Take a step of size h in the direction of the tangent"



Ending up on a different solution trajectory (dashed curves), each step introduces a *local error*, eventually accumulating a large *global error*

Geometric interpretation

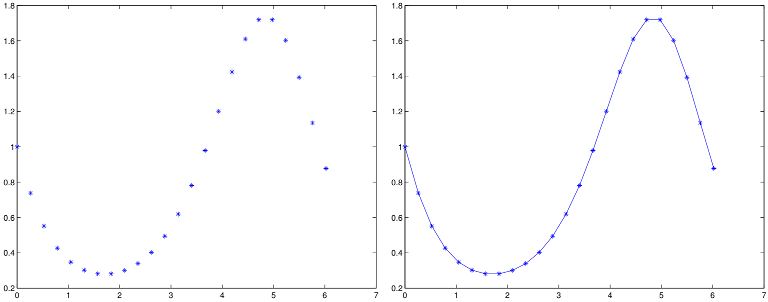
"Take a step of size h in the direction of the tangent"



Ending up on a different solution trajectory (dashed curves), each step introduces a *local error*, eventually accumulating a large *global error*

$$y' = -y \cos t; \quad y(0) = 1; \quad t \in [0, 2\pi]$$

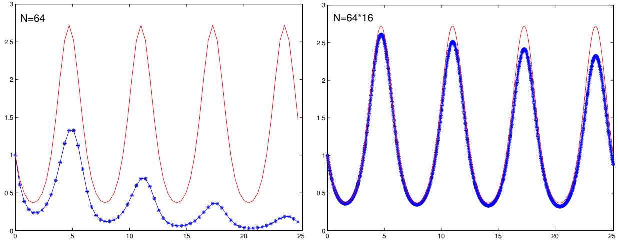
Stepsize $h = 2\pi/N$ with $N = 24$ gives $h = \pi/12$



The numerical solution is a *sequence of points* (t_n, y_n)

3. Convergence

Analytical and numerical solutions at $h = \pi/8$ and $h = \pi/128$

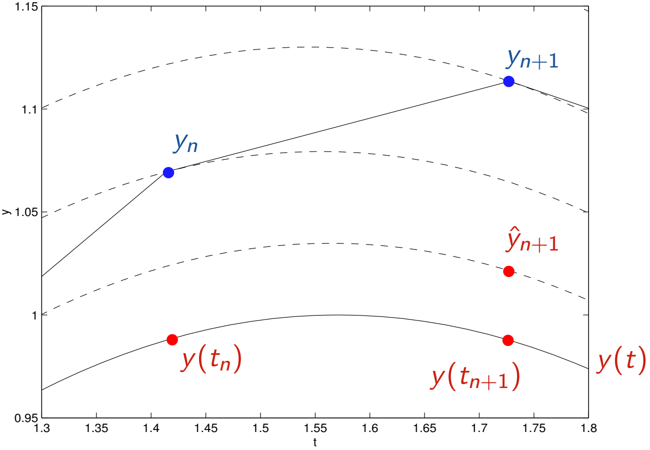


The *numerical solution* approaches the *exact solution* as $h \to 0$

Definition A method is *convergent* if, for every ODE with a Lipschitz vector field f, and every fixed $T = N \cdot h$, it holds that

$$\lim_{N\to\infty} \|y_{N,h} - y(T)\| = 0$$

Local and global errors



Global error $e_n = y_n - y(t_n)$ *and* $e_{n+1} = y_{n+1} - y(t_{n+1})$

Local error $l_{n+1} = \hat{y}_{n+1} - y(t_{n+1})$

Insert exact data $y(t_n)$ and $y(t_{n+1})$

$$y(t_{n+1}) = y(t_n) + hf(t_n, y(t_n)) - l_{n+1}$$

The residual is the *local error*

Taylor series $y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2} y''(t_n) + \dots \Rightarrow$

Explicit Euler local error $l_{n+1} \approx -\frac{h^2}{2} y''(t_n)$

The local error is evaluated along exact solution (solid curve)

Explicit Euler (numerical solution)

$$y_{n+1} = y_n + hf(t_n, y_n)$$

Subtract Taylor series expansion of exact solution

$$y(t_{n+1}) = y(t_n) + hf(t_n, y(t_n)) + \frac{h^2}{2}\, y''(t_n) + \dots$$

Global error recursion

$$e_{n+1} = e_n + hf(t_n, y(t_n) + e_n) - hf(t_n, y(t_n)) + l_{n+1}$$

shows how *local errors accumulate* into global error

Error propagation ...

$$e_{n+1} = e_n + hf(t_n, y(t_n) + e_n) - hf(t_n, y(t_n)) + l_{n+1}$$

Take norms and use Lipschitz condition

$$\|e_{n+1}\| \leq \|e_n\| + hL[f] \cdot \|e_n\| + \|l_{n+1}\|$$

Lemma *Assume that a non-negative sequence* $\{a_n\}$ *satisfies*

$$a_{n+1} \leq (1 + h\,\mu)a_n + ch^2$$

with $a_0 = 0$ *and* $\mu > 0$*. Then, for all* $n \geq 0$,

$$a_n \leq \frac{ch}{\mu}\left[(1 + h\,\mu)^n - 1\right] \leq ch\,\frac{\mathrm{e}^{\mu nh} - 1}{\mu}$$

Convergence of the Euler method

Theorem *The explicit Euler method is convergent*

Proof Suppose f is sufficiently differentiable. Given $h > 0$ and a fixed $T = Nh$, let $e_{n,h} = y_{n,h} - y(t_n)$

Apply the lemma to global error recursion, to get

$$\|e_{n,h}\| \leq \frac{c}{L[f]} h \left[(1 + h\,L[f])^n - 1\right] \leq ch \frac{\mathrm{e}^{TL[f]} - 1}{L[f]}$$

with

$$c = \max_n \|l_n\|/h^2 \approx \max_t \|y''\|/2$$

Convergence of the Euler method ...

$$\|e_{n,h}\| \leq \frac{c}{L[f]}\, h\,(e^{TL[f]} - 1) = C(T) \cdot h$$

implies the method is *convergent*, as $\lim_{h\to 0} \|e_{n,h}\| = 0$

Note

1) The global error can be made arbitrarily small
2) The error bound is way too large for practical purposes
3) Better error bounds can be obtained

Example

$$y' = -100y, \qquad y(0) = 1$$

Then $L[f] = 100$ and the exact solution is $y(t) = \mathrm{e}^{-100t}$ with $y''(t) = 100^2\mathrm{e}^{-100t}$, so $c = 100^2/2$, with bound

$$\|e_{n,h}\| \le \frac{100^2}{2 \cdot 100}\, h\,(\mathrm{e}^{100\,T} - 1)$$

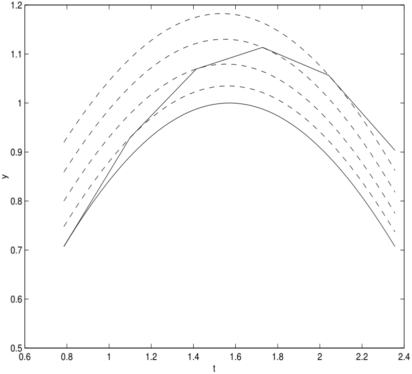
Error estimate at $T = 1$ is $\|e_{n,h}\| \le 50\, h\, \mathrm{e}^{100} \approx 1.4 \cdot 10^{45}\, h$!

Actual error As $y_n = (1 - 100h)^n$, the error at $T = 1$ for $h < 1/50$ is $\|e_{n,h}\| = |(1 - 100/N)^N - \mathrm{e}^{-100}| \le 3.7 \cdot 10^{-44}\, h$!

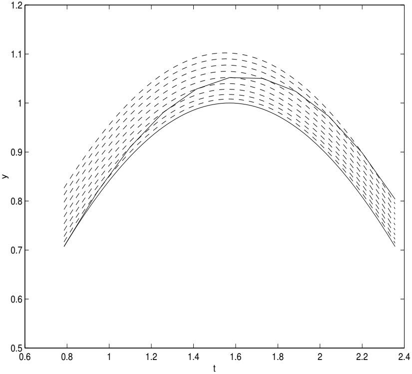
The error is overestimated by at least 89 orders of magnitude

Computational test $\quad \dot{y} = \lambda(y - \sin t) + \cos t$

$\lambda = 0.2$, with initial condition $y(\pi/4) = 1/\sqrt{2}$



$h = \pi/10$



$h = \pi/20$

Note Local error $\mathcal{O}(h^2)$, global error $\mathcal{O}(h)$!

4. Order of consistency

Always insert exact data and find the residual

If $y_{n+1} = \Phi_h(f, t_n, y_n, y_{n-1}, \dots)$, then the local error is

$$y(t_{n+1}) = \Phi_h(f, t_n, y(t_n), y(t_{n-1}), \dots) - l_{n+1}$$

Definition The *order of consistency* is p if

$$y(t_{n+1}) - \Phi_h(f, h, y(t_n), y(t_{n-1}), \dots) = \mathcal{O}(h^{p+1})$$

as $h \to 0$, for every analytic f. The *local error* is then $\mathcal{O}(h^{p+1})$

Alternative The order of consistency is p if the formula is *exact for all polynomials $y = P(t)$ of degree p or less*

Example Expanding in Taylor series,

$$y(t_{n+1}) - [y(t_n) + h\, f(t_n, y(t_n))] = \mathcal{O}(h^2)$$

so the method's consistency order is one

Alternatively, suppose $y(t) = 1$ with $f = y' = 0$. Then
$y(t_{n+1}) = y(t_n) + hf(t_n, y(t_n)) = 1 + h \cdot 0 = 1$, so exact

Next take 1st degree polynomial $y(t) = t$ with $f = y' = 1$. Then
$h = y(t_{n+1}) = y(t_n) + hf(t_n, y(t_n)) = 0 + h \cdot 1 = h$

For 2nd degree polynomial $y(t) = t^2$ with $f = y' = 2t$ we get
$h^2 = y(t_{n+1}) = y(t_n) + hf(t_n, y(t_n)) = 0 + h \cdot 0 = 0 \neq h^2$

5. The trapezoidal rule

Explicit Euler linearizes at t_n with slope $y'(t_n)$. Instead, take average of $y'(t_n)$ and $y'(t_{n+1})$ and approximate

$$y'(t_n) \approx \frac{1}{2}[f(t_n, y(t_n)) + f(t_{n+1}, y(t_{n+1}))]$$

This gives the *trapezoidal rule*

$$y_{n+1} = y_n + \frac{h}{2}[f(t_n, y_n) + f(t_{n+1}, y_{n+1})]$$

The method is *implicit*

Nonlinear equation solving is required on each step

Insert exact solution and expand in Taylor series

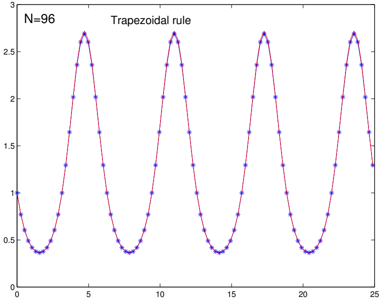
$$
\begin{aligned}
& y(t_{n+1}) - \{y(t_n) + \frac{h}{2}[f(t_n, y(t_n)) + f(t_{n+1}, y(t_{n+1}))]\} \\
&= y(t_n) + h\, y'(t_n) + \frac{h^2}{2} y''(t_n) + \frac{h^3}{6} y'''(t_n) + \mathcal{O}(h^4) \\
&\quad - \{y(t_n) + \frac{h}{2}\left(y'(t_n) + [y'(t_n) + h\, y''(t_n) + \frac{h^2}{2} y'''(t_n)]\right)\} \\
&= -\frac{h^3}{12} y'''(t_n) + \mathcal{O}(h^4)
\end{aligned}
$$

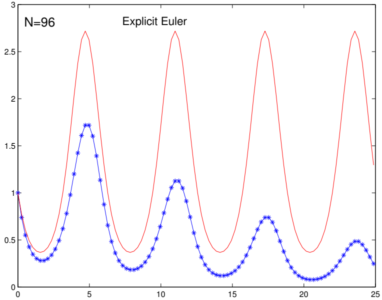
Theorem *The trapezoidal rule is convergent of order two*
(No proof given here)

The dramatic impact of 2nd order convergence

$$y' = -y \cos t; \quad y(0) = 1; \quad t \in [0, 8\pi]$$

Stepsize $h = 8\pi/N$ with $N = 96$ gives $h = \pi/12$





Numerical solutions with Trapezoidal rule and Explicit Euler

Implicit Euler $y_{n+1} = y_n + h\, f(t_{n+1}, y_{n+1})$

We need to solve a nonlinear equation to compute y_{n+1}

The extra cost is motivated if we can take larger steps

There are some problems where implicit methods can take enormously large time steps without losing accuracy!

We will return to how to solve nonlinear equations

We can also approximate the derivative $y'(t)$ by

$$y'(t) \approx f\left(\frac{t_n + t_{n+1}}{2}, \frac{y_n + y_{n+1}}{2}\right), \quad t \in [t_n, t_{n+1}]$$

resulting in the 2nd order *Implicit Midpoint Method*

$$y_{n+1} = y_n + h\, f\left(\frac{t_n + t_{n+1}}{2}, \frac{y_n + y_{n+1}}{2}\right)$$

6. Theta methods

Approximate y' by a *convex combination* of $y'(t_n)$ and $y'(t_{n+1})$

$$y_{n+1} = y_n + h\left[\theta f(t_{n+1}, y_{n+1}) + (1-\theta) f(t_n, y_n)\right], \qquad \theta \in [0,1]$$

- Explicit Euler $\theta = 0$
- Trapezoidal rule (implicit) $\theta = 1/2$
- Implicit Euler $\theta = 1 \quad \Rightarrow \quad y_{n+1} = y_n + h\, f(t_{n+1}, y_{n+1})$

Use Taylor series expansion to get

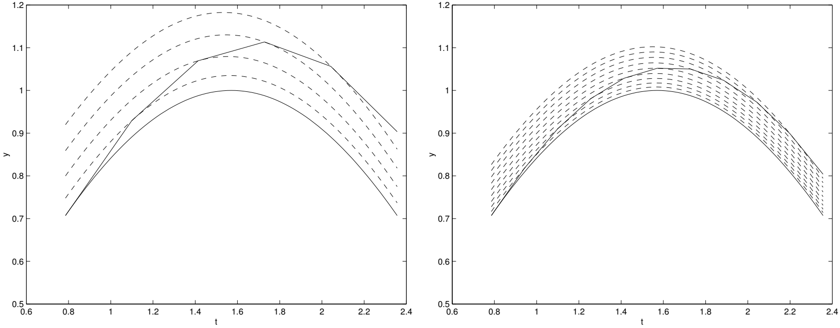
$$y(t_{n+1}) - y(t_n) - h\left[\theta f(t_{n+1}, y(t_{n+1})) + (1-\theta) f(t_n, y(t_n))\right]$$

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

If $\theta = 1/2$ the method is of order 2; otherwise it is of order 1

Theorem (without proof) *The Theta-methods are convergent*

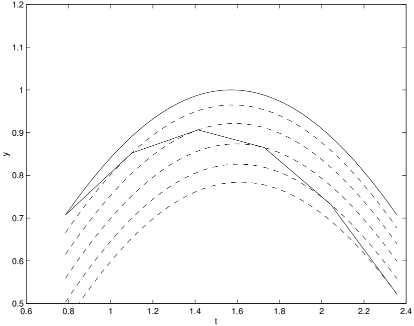
$$\dot{y} = \lambda(y - \sin t) + \cos t \quad \text{with} \quad \lambda = -0.2$$



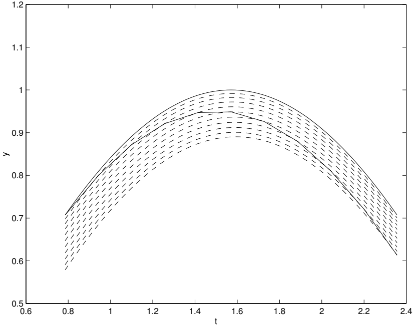
$h = \pi/10$ $\qquad$ $h = \pi/20$

Local error $\mathcal{O}(h^2)$, global error $\mathcal{O}(h)$

$$\dot{y} = \lambda(y - \sin t) + \cos t \quad \text{with} \quad \lambda = -0.2$$



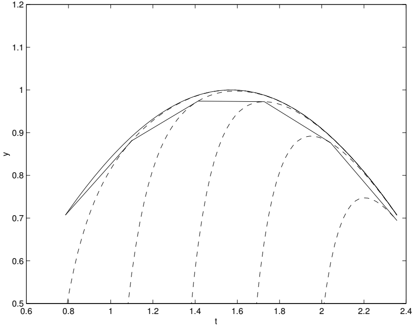
$h = \pi/10$



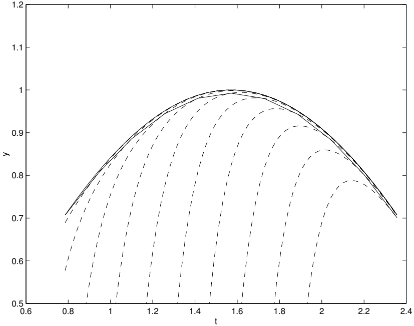
$h = \pi/20$

Local error $\mathcal{O}(h^2)$, global error $\mathcal{O}(h)$

$$\dot{y} = \lambda(y - \sin t) + \cos t \quad \text{with} \quad \lambda = -10$$



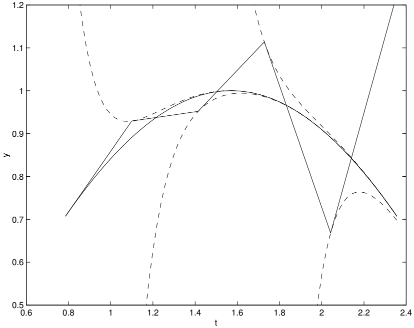
$h = \pi/10$



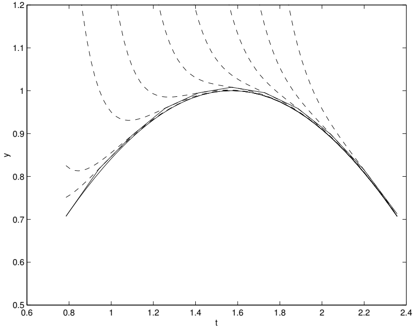
$h = \pi/20$

Local error $\mathcal{O}(h^2)$, global error $\mathcal{O}(h)$

$$\dot{y} = \lambda(y - \sin t) + \cos t \quad \text{with} \quad \lambda = -10$$



$h = \pi/10$



$h = \pi/20$

Numerical instability!

8. The linear test equation

Definition The *linear test equation* is

$$y' = \lambda y, \quad y(0) = 1, \quad t \geq 0, \quad \lambda \in \mathbb{C}$$

As $y(t) = \mathrm{e}^{\lambda t}$, we have

$$|y(t)| \leq K \quad \Leftrightarrow \quad \mathrm{Re}(\lambda) \leq 0$$

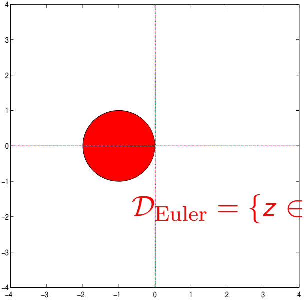
Mathematical stability Bounded solutions if $\mathrm{Re}(\lambda) \leq 0$

When does a numerical method have this property?

Does $\mathrm{Re}(\lambda) \leq 0$ imply **numerical stability**?

Definition The *stability region* $\mathcal{D}$ of a method is the set of all $h\lambda \in \mathbb{C}$ such that $|y_n| \leq \tilde{K}$ when the method is applied to the test equation

Example For Euler's method, $y_{n+1} = (1 + h\lambda)y_n$, implying that *$y_n$ remains bounded if and only if* $|1 + h\lambda| \leq 1$



$$\mathcal{D}_{\mathrm{Euler}} = \{z \in \mathbb{C} : |1 + z| \leq 1\}$$

A-stability

Definition A method is called *A-stable* if $\mathbb{C}^- \subset \mathcal{D}$

For the trapezoidal rule

$$\mathcal{D}_{\mathrm{TR}} = \left\{ z \in \mathbb{C} : \left| \frac{1 + \frac{1}{2}z}{1 - \frac{1}{2}z} \right| \leq 1 \right\} \equiv \mathbb{C}^-$$

so $\mathrm{Re}(z) \leq 0$ implies that y_n remains bounded *for any* $h > 0$

The explicit Euler method is *not A-stable* but the implicit Euler method and the trapezoidal rule are *A-stable*

Common idea – *"If the original problem is stable, then an A-stable method will replicate that behavior numerically"*

Relevance of linear test equation

Assume that $y' = Ay$ is diagonalizable, with $AT = T\Lambda$

Then $u = T^{-1}y$ satisfies $u' = \Lambda u$ (scalar equations)

If the explicit Euler method is applied to $y' = Ay$, we get

$$y_{n+1} = (I + hA)y_n$$

Putting $u_n = T^{-1}y_n$ leads to

$$u_{n+1} = (I + h\Lambda)u_n$$

T diagonalizes the differential equation *and* its discretization

Interpret λ in linear test equation as eigenvalue of A

9. Stiff ODEs

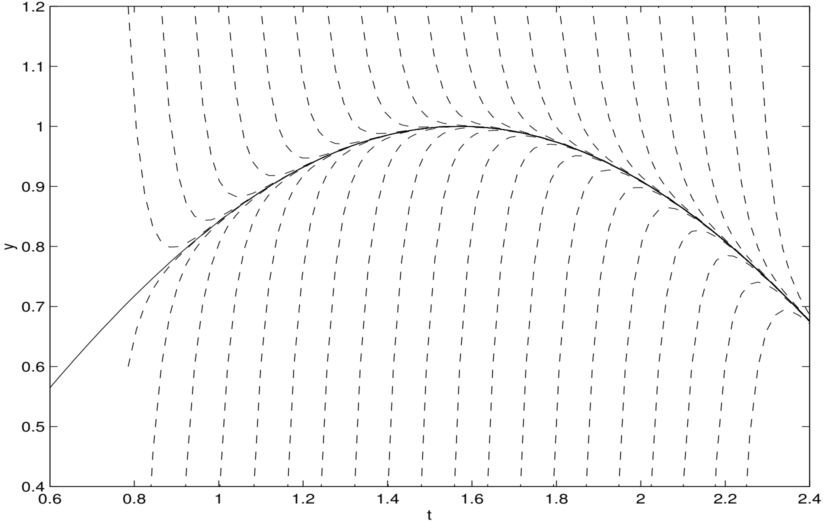
Example Solve $\dot{y} = \lambda(y - \sin t) + \cos t$ with $\lambda = -500$

Solution

$$
\begin{array}{ll}
\textit{Particular} & y_P(t) = \sin t \\
\textit{Homogeneous} & y_H(t) = \mathrm{e}^{\lambda t} \\
\textit{General} & y(t) = \mathrm{e}^{\lambda(t-t_0)}(y(t_0) - \sin t_0) + \sin t
\end{array}
$$

Study the flow of this equation and numerical solutions

Flow (solution trajectories) $\lambda = -50$



Stiffness

Stiff differential equations are characterized by homogeneous solutions being *strongly damped*

Example $\dot{y} = \lambda(y - \sin t) + \cos t$ with $\lambda \ll -1$

Explicit methods have *bounded stability regions* putting strong stability restrictions on the step size h

With *Explicit Euler* the solution approaches $\sin t$ as $t \to \infty$ if and *only if* $h < 1/250$

The step size h must be kept small, not to keep errors small, but *to maintain numerical stability*

The *Trapezoidal Rule* solution approaches $\sin t$ as $t \to \infty$ *for every* $h > 0$

The step size must only be kept small to keep errors small

For A-stable methods, there is no stability restriction, only an *accuracy restriction*

Implicit methods with *unbounded stability regions* put no stability restrictions on h

The *stepsize is only restricted by accuracy requirements*

Example The implicit Euler applied to the problem above only requires

$$l_n \approx \frac{h^2 y''}{2} \approx \frac{h^2}{2} \sin t_n$$

to be sufficiently small, *independently of* λ