ETH Lecture 401-0663-00L Numerical Methods for CSE

Numerical Methods for Computational Science and Engineering

Prof. R. Hiptmair, SAM, ETH Zurich

(with contributions from Prof. P. Arbenz and Dr. V. Gradinaru)

Autumn Term 2015

(C) Seminar für Angewandte Mathematik, ETH Zürich

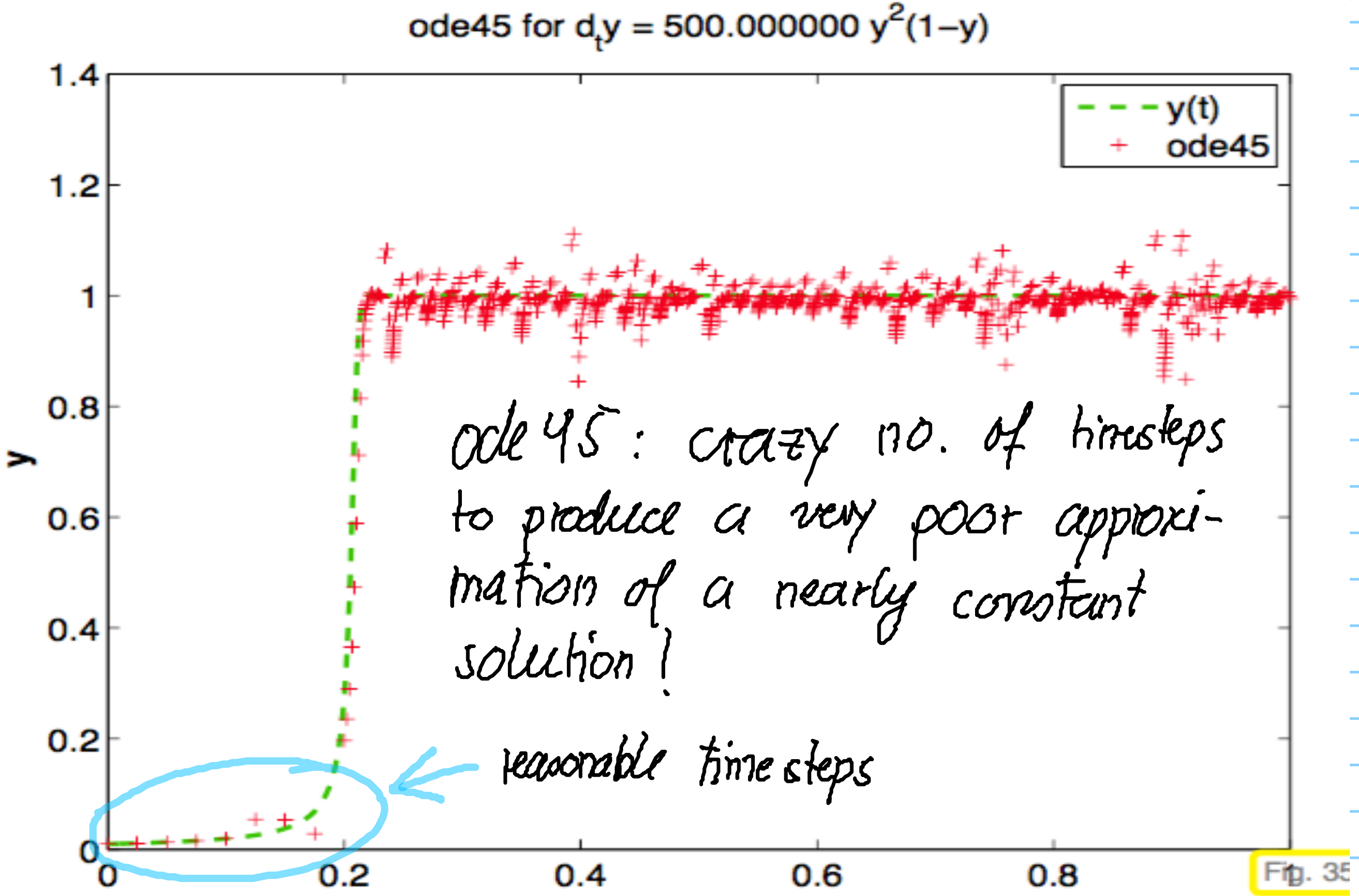
URL: http://www.sam.math.ethz.ch/~hiptmair/tmp/NumCSE/NumCSE15.pdf

XII. Single step methods for stiff IVPs

MATLAB-script 12.0.2: Use of MATLABintegrator ode45 for a stiff problem

```
fun = @(t,x) 500*x^2*(1-x);
options = odeset('reltol',0.1,'abstol',0.001,'stats','on');
[t,y] = ode45(fun,[0 1],y0,options);
```

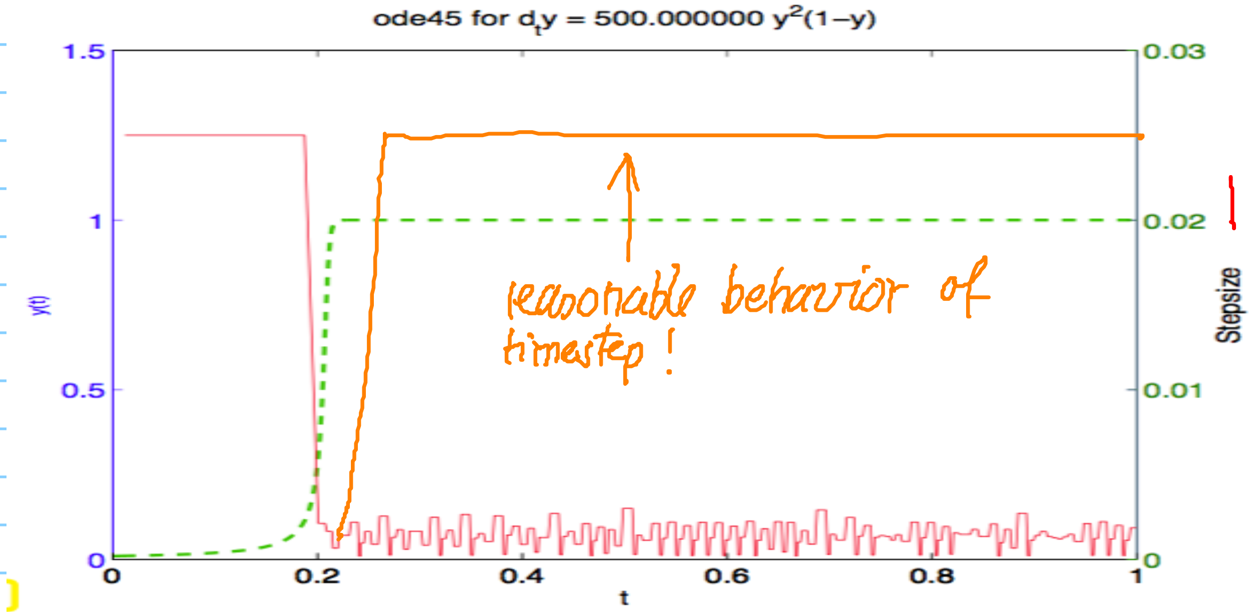
$$\dot{y} = \lambda y^2(1-y^2)$$



ode45: crazy no. of timesteps to produce a very poor approximation of a nearly constant solution!

reasonable timesteps

Fig. 35



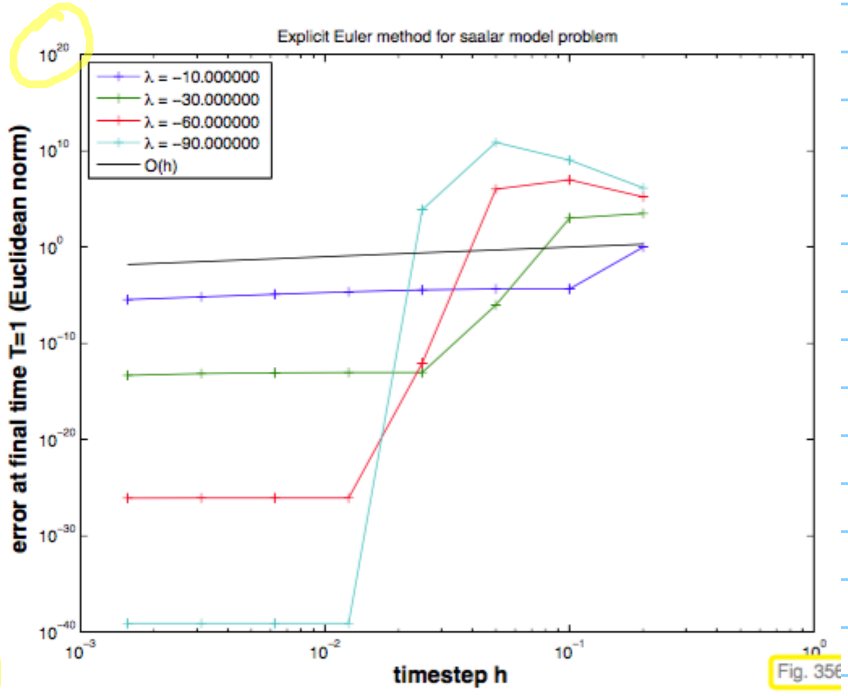
reasonable behavior of timestep!

②

What's wrong with RK-SSM underlying ODE45 ?

12.1 Model problem analysis

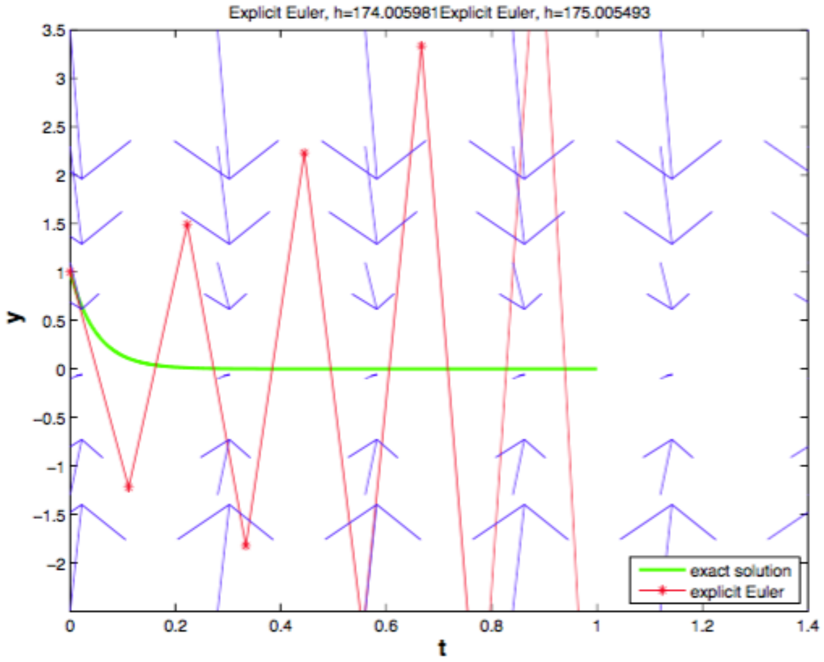
Example : Explicit Euler method for $\dot{y} = \lambda y$, uniform h
$[\, y(0) = 1 \,,\ \lambda < 0 \,]$ [decay equation]



55

Fig. 356

→ blow-up for large $|\lambda h|$



◁ Exponentially growing oscillations due to overshooting

Analysis : Expl. Eul. $\quad y_{k+1} = y_k + h\lambda y_k$

$$\Rightarrow \quad y_k = (1 + h\lambda)^k y_0 \quad , \; k = 1, \dots, N$$

$$\text{blow-up} \Leftrightarrow \quad |1 + h\lambda| > 1$$

For $\lambda < 0$ we want $y_k \to 0 \Leftrightarrow |1 + h\lambda| < 1$

If $\lambda < 0$, $|h\lambda| > 2 \Rightarrow$ blow-up

Necessary is $|h\lambda| \leq 2 \Rightarrow$ timestep constraint $h < \frac{2}{|\lambda|}$

Experiment : Adaptive timestepping (expl. Euler, expl. trap.)
for decay equation with $\lambda \ll 0$
$[\, \lambda = -100 \,]$

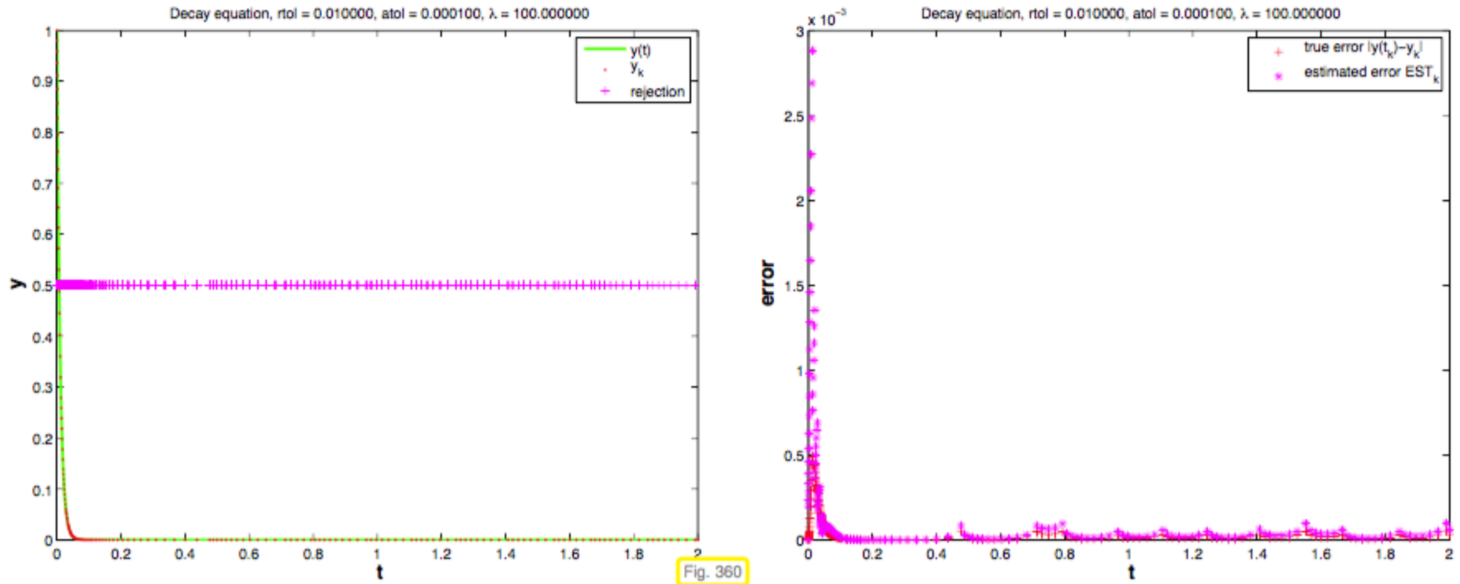


Fig. 360

→ adaptive timestepping detects time step constraints

→ small timesteps throughout (inefficient)

Is this a flaw of explicit Euler ?

Example: Explicit trapezoidal method [2 stage RK-SSM]

$$\dot{y} = \lambda y \quad \Rightarrow \quad y_1 = \underbrace{(1 + \lambda h + \tfrac{1}{2}(\lambda h)^2)}_{=:\, S(\lambda h)} y_0$$

$$\Rightarrow \quad y_k = S(\lambda h)^k y_0$$

No blow-up $\Leftrightarrow$ $|S(\lambda h)| \leq 1$

$[\lambda < 0]$ $\Leftrightarrow$ $-2 < \lambda h < 0$ → timestep constraint

For general explicit RK-SSM : $\begin{array}{c|c} \mathbf{c} & \mathfrak{A} \\ \hline & \mathbf{b}^T \end{array}$ [Butcher scheme]

applied $\dot{y} = \lambda y$, $\lambda \in \mathbb{R}$

Definition 11.4.9. Explicit Runge-Kutta method

For $b_i, a_{ij} \in \mathbb{R}$, $c_i := \sum_{j=1}^{i-1} a_{ij}$, $i,j = 1,\dots,s$, $s \in \mathbb{N}$, an s-stage explicit Runge-Kutta single step method (RK-SSM) for the IVP (11.1.20) is defined by

$$\mathbf{k}_i := \mathbf{f}(t_0 + c_i h, \mathbf{y}_0 + h \sum_{j=1}^{i-1} a_{ij} \mathbf{k}_j) \;, \quad i = 1,\dots,s \quad , \quad \mathbf{y}_1 := \mathbf{y}_0 + h \sum_{i=1}^{s} b_i \mathbf{k}_i \,.$$

The vectors $\mathbf{k}_i \in \mathbb{R}^d$, $i = 1,\dots,s$, are called increments, $h > 0$ is the size of the timestep.

$$f(t,y) = \lambda y \quad \Rightarrow \quad \begin{aligned} k_i &= \lambda (y_0 + h \sum_{j=1}^{i-1} a_{ij} k_j) \\ y_1 &= y_0 + h \sum_{i=1}^{s} b_i k_i \end{aligned} \qquad (*)$$

$$(*) \Rightarrow \begin{bmatrix} I - z\mathfrak{A} & 0 \\ -z\underline{b}^T & 1 \end{bmatrix} \begin{bmatrix} \underline{k} \\ y_1 \end{bmatrix} = y_0 \underbrace{\begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}}_{\mathbb{1}}$$

$$\underline{k} = \begin{bmatrix} k_1 \\ \vdots \\ k_s \end{bmatrix} \qquad z := \lambda h$$

$$\Rightarrow \quad y_1 = y_0 + z\underline{b}^T (I - z\mathfrak{A})^{-1} \mathbb{1}\, y_0$$

$$= \underbrace{S(z)}_{\in \mathbb{R}} y_0$$

Theorem 12.1.15. Stability function of explicit Runge-Kutta methods → [32, Thm. 77.2], [45, Sect. 11.8.4]

The discrete evolution Ψ^h_λ *of an explicit* s*-stage Runge-Kutta single step method (→ Def. 11.4.9) with Butcher scheme* $\begin{array}{c|c} \mathbf{c} & \mathfrak{A} \\ \hline & \mathbf{b}^T \end{array}$ *(see (11.4.11)) for the ODE* $\dot{y} = \lambda y$ *amounts to a multiplication with the number*

$$\Psi^h_\lambda = S(\lambda h) \quad \Leftrightarrow \quad y_1 = S(\lambda h) y_0$$

where S *is the* *stability function*

$$S(z) := 1 + z\mathbf{b}^T(\mathbf{I} - z\mathfrak{A})^{-1}\mathbf{1} = \det(\mathbf{I} - z\mathfrak{A} + z\mathbf{1}\mathbf{b}^T), \quad \mathbf{1} = [1,\dots,1]^T \in \mathbb{R}^s . \qquad (12.1.16)$$

↳ from Cramer's rule

▷ RK-sequence (uniform timestep $h>0$) : $y_k = S(z)^k y_0$

Examples :

- Explicit Euler method (11.2.7): $\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}$ ➤ $S(z) = 1 + z$
- Explicit trapezoidal method (11.4.6): $\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$ ➤ $S(z) = 1 + z + \frac{1}{2}z^2$
- Classical RK4 method: $\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} \end{array}$ ➤ $S(z) = 1 + z + \frac{1}{2}z^2 + \frac{1}{6}z^3 + \frac{1}{24}z^4$

$$\Updownarrow$$

$$e^z = 1 + z + \tfrac{1}{2}z^2 + \tfrac{1}{6}z^3 + \tfrac{1}{24}z^4 + \dots$$

Corollary 12.1.18. Polynomial stability function of explicit RK-SSM

For a consistent (→ Def. 11.3.10) s*-stage explicit Runge-Kutta single step method according to Def. 11.4.9 the stability function* S *defined by* (12.1.16) *is a non-constant polynomial of degree* $\leq s$*:* $S \in \mathcal{P}_s$.

$\Rightarrow \quad |S(z)| \longrightarrow \infty \quad$ as $\quad |z| \longrightarrow \infty$

$\Rightarrow$ Timestep constraint to avoid blow-up for $\dot{y} = \lambda y$

↳ $|\lambda| h$ sufficiently small $\underbrace{\text{to avoid blow-up}}$

$\Leftrightarrow \quad |S(\lambda h)| \leq 1$

Only if one ensures that $|\lambda h|$ is sufficiently small, one can avoid exponentially increasing approximations y_k (qualitatively wrong for $\lambda < 0$) when applying an explicit RK-SSM to the model problem (12.1.3) with uniform timestep $h > 0$,

Model problem analysis for linear systems of ODEs

$$\dot{\mathbf{y}} = \mathbf{M}\mathbf{y} \,, \quad \mathbf{M} \in \mathbb{R}^{d,d}$$

LA : Solved by diagonalization [Note: $\lambda_i \in \mathbb{C}$]

$$\boxed{\mathbf{V}^{-1}\mathbf{M}\mathbf{V} = \mathbf{D} = \mathrm{diag}(\underset{\uparrow}{\lambda_1}, \dots, \underset{\uparrow}{\lambda_d})}$$

eigenvalues

$\mathbf{z} = \mathbf{V}^{-1}\mathbf{y} \quad \Rightarrow \quad \dot{\mathbf{z}} = \mathbf{D}\mathbf{z}$

$\triangleq$ decoupled scalar linear ODEs

⑤ Explicit Euler method for $\dot{y} = My$: diagonalization

$$y_{k+1} = y_k + hMy_k$$

$$z_k = V^{-1}y_k \qquad z_{k+1} = z_k + hV^{-1}MVV^{-1}y_k = (I+hD)z_k$$

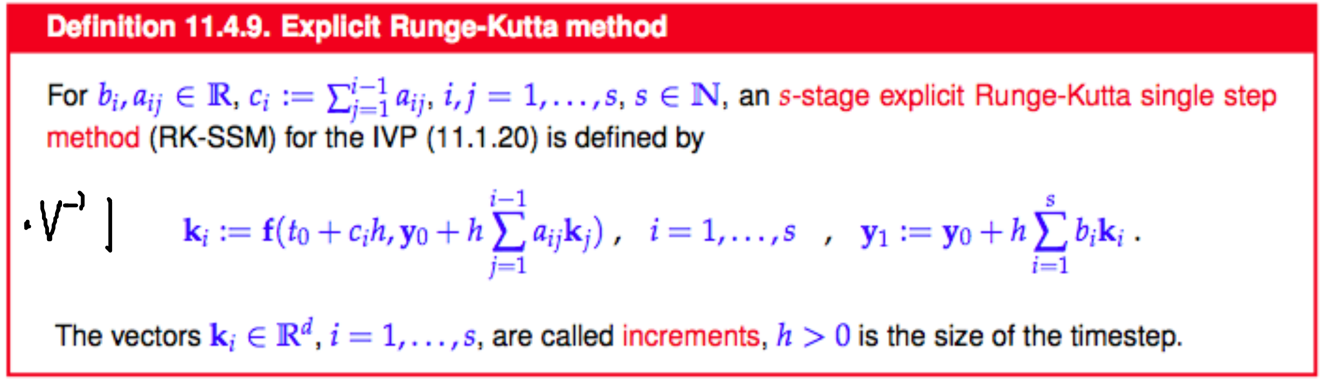
$$\Rightarrow (z_{k+1})_i = (1+h\lambda_i)(z_k)_i \quad [\text{decoupled}]$$

To avoid blow-up ensure $|1+h\lambda_i| \leq 1 \quad \forall i=1,\dots,d$

[also for $\lambda_i \in \mathbb{C}$] $\Leftrightarrow$ $h|\lambda_i| \leq 2$ timestep constraint

General RK-SSM applied to $\dot{y} = My$

$$[\quad V^{-1}MV = D = \text{diag}(\lambda_1,\dots,\lambda_d) \quad]$$



Definition 11.4.9. Explicit Runge-Kutta method

For $b_i, a_{ij} \in \mathbb{R}$, $c_i := \sum_{j=1}^{i-1} a_{ij}$, $i,j = 1,\dots,s$, $s \in \mathbb{N}$, an s-stage explicit Runge-Kutta single step method (RK-SSM) for the IVP (11.1.20) is defined by

$\cdot V^{-1}$]

$$\mathbf{k}_i := \mathbf{f}(t_0 + c_i h, \mathbf{y}_0 + h\sum_{j=1}^{i-1} a_{ij}\mathbf{k}_j)\,, \quad i = 1,\dots,s \quad , \quad \mathbf{y}_1 := \mathbf{y}_0 + h\sum_{i=1}^{s} b_i \mathbf{k}_i\,.$$

The vectors $\mathbf{k}_i \in \mathbb{R}^d$, $i = 1,\dots,s$, are called increments, $h > 0$ is the size of the timestep.

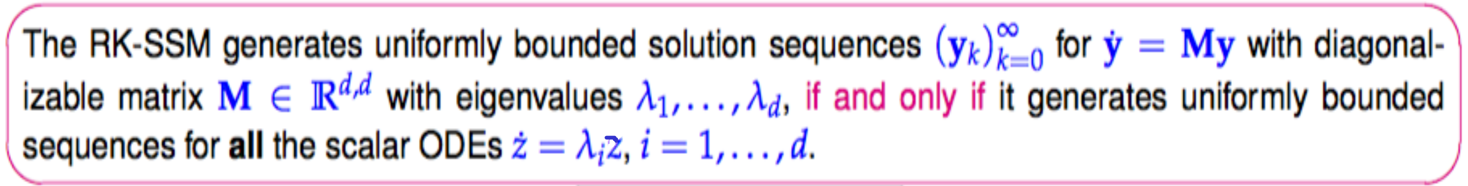
$f(y) = My$, transformed increments : $\hat{k}_i = V^{-1}k_i$

$$z_k = V^{-1}y_k$$

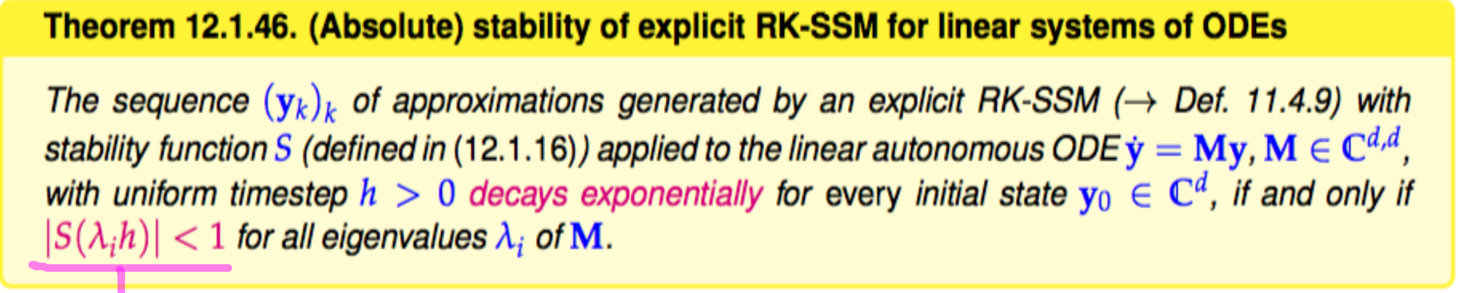
$$\Rightarrow \begin{aligned} \hat{k}_i &= D(z_0 + h\sum_j a_{ij}\hat{k}_j) \\ z_1 &= z_0 + h\sum_{i=1}^{s} b_i \hat{k}_i \end{aligned} \quad \Rightarrow \text{decoupled}$$

$$(\hat{k}_i)_\ell = \lambda_\ell[(z_0)_\ell + h\sum a_{ij}(\hat{k}_j)_\ell]$$

$\hookrightarrow$ Increment equation of RK-SSM applied to $\dot{z} = Dz$.

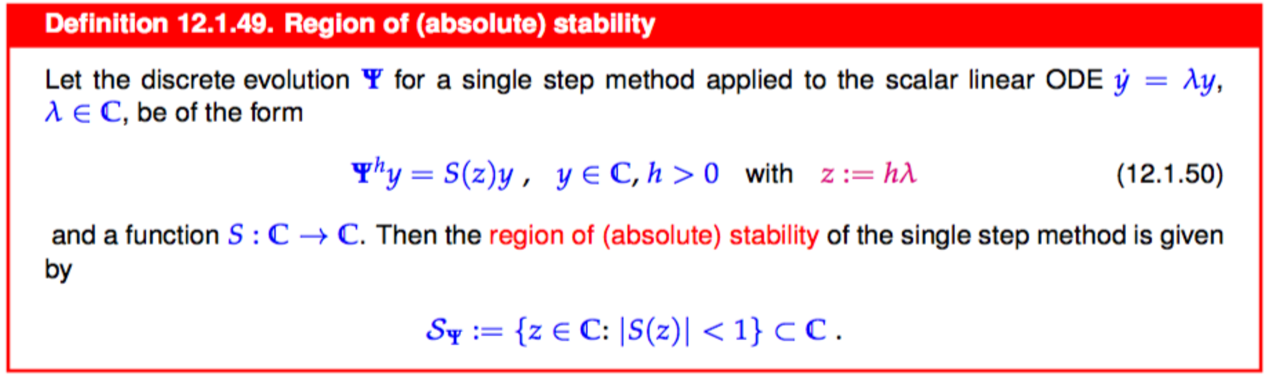


The RK-SSM generates uniformly bounded solution sequences $(\mathbf{y}_k)_{k=0}^{\infty}$ for $\dot{\mathbf{y}} = \mathbf{M}\mathbf{y}$ with diagonalizable matrix $\mathbf{M} \in \mathbb{R}^{d,d}$ with eigenvalues $\lambda_1,\dots,\lambda_d$, if and only if it generates uniformly bounded sequences for **all** the scalar ODEs $\dot{z} = \lambda_i z$, $i = 1,\dots,d$.



Theorem 12.1.46. (Absolute) stability of explicit RK-SSM for linear systems of ODEs

The sequence $(\mathbf{y}_k)_k$ of approximations generated by an explicit RK-SSM (→ Def. 11.4.9) with stability function S (defined in (12.1.16)) applied to the linear autonomous ODE $\dot{\mathbf{y}} = \mathbf{M}\mathbf{y}$, $\mathbf{M} \in \mathbb{C}^{d,d}$, with uniform timestep $h > 0$ decays exponentially for every initial state $\mathbf{y}_0 \in \mathbb{C}^d$, if and only if $|S(\lambda_i h)| < 1$ for all eigenvalues λ_i of $\mathbf{M}$.



Definition 12.1.49. Region of (absolute) stability

Let the discrete evolution Ψ for a single step method applied to the scalar linear ODE $\dot{y} = \lambda y$, $\lambda \in \mathbb{C}$, be of the form

$$\Psi^h y = S(z)y\,,\quad y \in \mathbb{C}, h > 0 \quad \text{with} \quad z := h\lambda \tag{12.1.50}$$

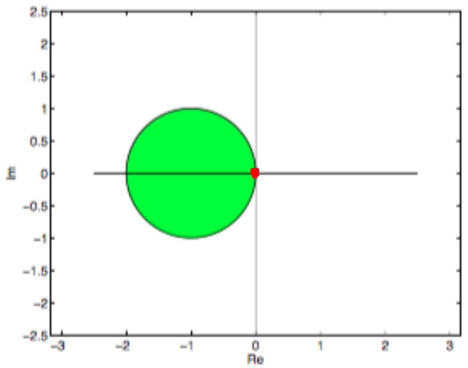
and a function $S : \mathbb{C} \to \mathbb{C}$. Then the region of (absolute) stability of the single step method is given by

$$\mathcal{S}_\Psi := \{z \in \mathbb{C} : |S(z)| < 1\} \subset \mathbb{C}\,.$$

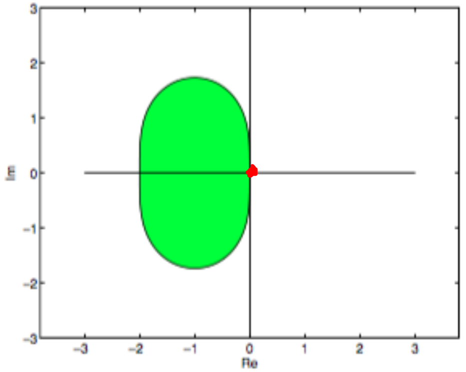
Why $\mathbb{C}$? → because of complex eigenvalues of M

If S_Ψ is bounded $\Rightarrow$ Timestep constraint to avoid blow-up.

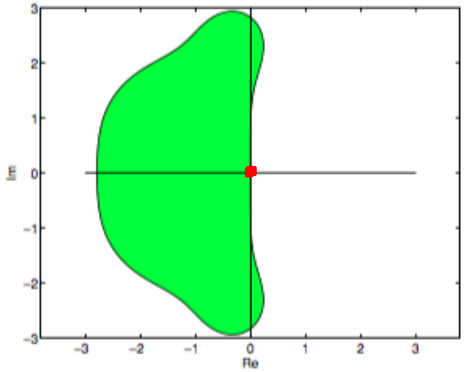
Examples :



$\mathcal{S}_\Psi$: explicit Euler (11.2.7)



$\mathcal{S}_\Psi$: explicit trapezoidal method



$\mathcal{S}_\Psi$: classical RK4 method

$|1+z| < 1$

Remark: $0 \in \partial S_\Psi$, because $S(z) = 1 + z + O(|z|^2)$ for consistent SSM.

boundary

12.2. Stiff IVPs

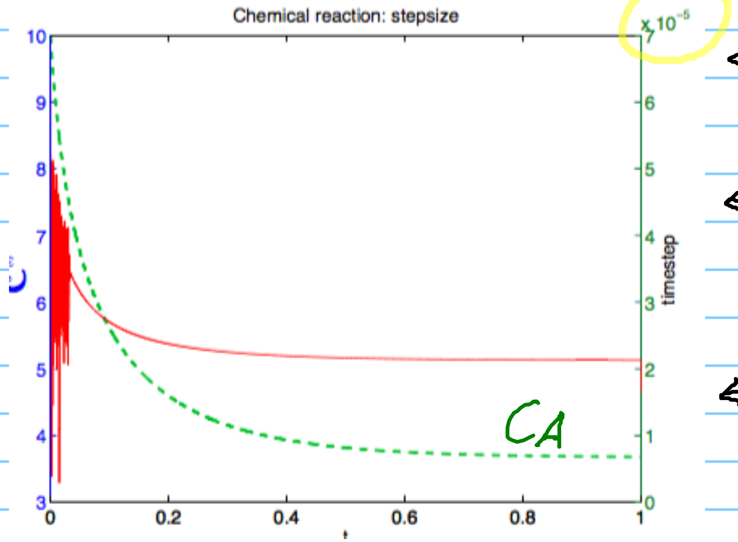
Example : Chemical reaction

reaction: $\underbrace{A + B \underset{k_1}{\overset{k_2}{\rightleftarrows}} C}_{\text{fast reaction}}$, $\underbrace{A + C \underset{k_3}{\overset{k_4}{\rightleftarrows}} D}_{\text{slow reaction}}$

reaction constants: $\boxed{k_1, k_2 \gg k_3, k_4}$

Mathematical model: non-linear ODE involving concentrations $\mathbf{y}(t) = (c_A(t), c_B(t), c_C(t), c_D(t))^T$

$$\dot{\mathbf{y}} := \frac{d}{dt}\begin{bmatrix} c_A \\ c_B \\ c_C \\ c_D \end{bmatrix} = \mathbf{f}(\mathbf{y}) := \begin{bmatrix} -k_1 c_A c_B + k_2 c_C - k_3 c_A c_C + k_4 c_D \\ -k_1 c_A c_B + k_2 c_C \\ k_1 c_A c_B - k_2 c_C - k_3 c_A c_C + k_4 c_D \\ k_3 c_A c_C - k_4 c_D \end{bmatrix} . \qquad (12.2.3)$$



◁ Explicit adaptive RK-SSM

← tiny timesteps though solution does not change much

← Looks like stability induced timestep constraint

Discussion : linear system : $\dot{y} = My$

EVs $\lambda_1, \dots, \lambda_d$:

- There is a λ_j : $\operatorname{Re}\lambda_j \gg 0$ → blow-up of exact solution

[Note: $\operatorname{Re}\lambda$ in $y(t) = e^{\lambda t}$ governs decay/growth :

$$|e^{\lambda t}| = e^{\operatorname{Re}(\lambda t)}$$]

⇒ Blow-up of (y_k) is even desirable

- $\underbrace{\mathrm{Re}\,\lambda_j \leq 0}\ \forall j$ and $|\mathrm{Re}\,\lambda_\ell| \gg 1$

→ Exact solution remains bounded

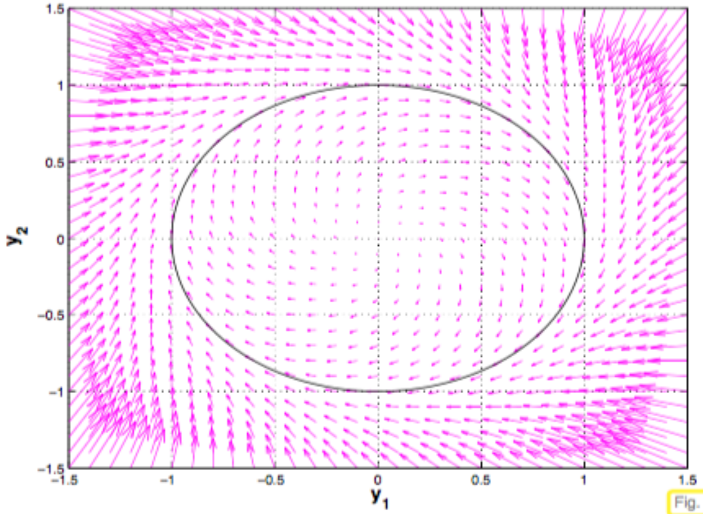
→ Blow-up of numerical solution disastrous !

→ Avoid blow-up ! → time step constraint ! $\lrcorner$

Example : Strongly attractive limit cycle

ODE : $\dot{\mathbf{y}} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mathbf{y} + \lambda(1-\|\mathbf{y}\|^2)\,\mathbf{y}$

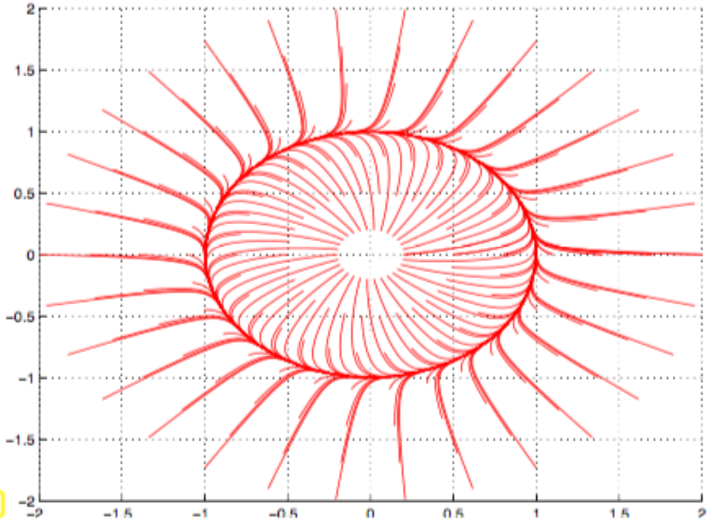
$\|\mathbf{y}(0)\|_2 = 1 \Rightarrow \|\mathbf{y}(t)\|_1 = 1 \quad \forall t$



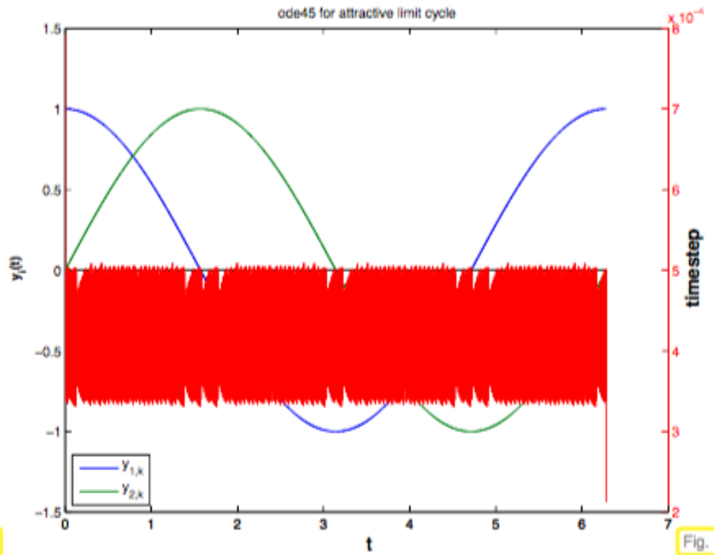
368

Fig. 369

vectorfield $\mathbf{f}$ ($\lambda = 1$)



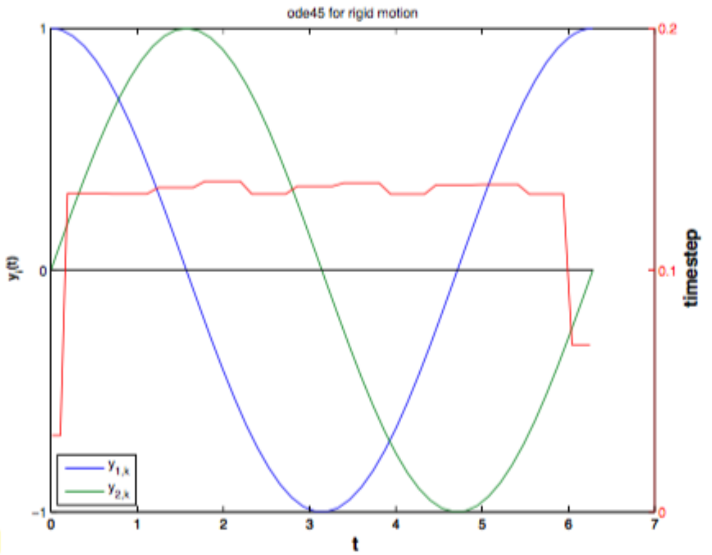
solution trajectories ($\lambda = 10$)



70

Fig. 371

many (3794) steps ($\lambda = 1000$)



accurate solution with few steps ($\lambda = 0$)

Notion 12.2.9. Stiff IVP

An initial value problem is called stiff, if stability imposes much tighter timestep constraints on *explicit single step methods* than the accuracy requirements.

Heuristic considerations for predicting stiffness :

- Linear ODEs $\dot{\mathbf{y}} = \mathbf{M}\mathbf{y}$: ✓
- General ODE $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$, stiffness state dependent

Stiff at state $\mathbf{y}^* \in \mathbf{y}(t)$?

Tool : Linearization $\mathbf{f}(\mathbf{y}) \approx \mathbf{f}(\mathbf{y}^*) + D\mathbf{f}(\mathbf{y}^*)(\mathbf{y}-\mathbf{y}^*)$

▷ "Close" to $\mathbf{y}^*$ solutions of $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$ will behave like solutions of the linearized ODE

$$\dot{\mathbf{z}} = \mathbf{f}(\mathbf{y}^*) + D\mathbf{f}(\mathbf{y}^*)(\mathbf{z}-\mathbf{y}^*) \qquad (L)$$

⑧ (L) ≙ affine linear ODE

Linearization of explicit RK-SSM

Definition 11.4.9. Explicit Runge-Kutta method

For $b_i, a_{ij} \in \mathbb{R}$, $c_i := \sum_{j=1}^{i-1} a_{ij}$, $i,j = 1,\dots,s$, $s \in \mathbb{N}$, an s-stage explicit Runge-Kutta single step method (RK-SSM) for the IVP (11.1.20) is defined by

$$\mathbf{k}_i := \mathbf{f}(\cancel{t_0 + c_i h}, \mathbf{y}_0 + h\sum_{j=1}^{i-1} a_{ij}\mathbf{k}_j)\ , \quad i = 1,\dots,s \quad , \quad \mathbf{y}_1 := \mathbf{y}_0 + h\sum_{i=1}^{s} b_i\mathbf{k}_i\ .$$

The vectors $\mathbf{k}_i \in \mathbb{R}^d$, $i = 1,\dots,s$, are called increments, $h > 0$ is the size of the timestep.

$y_0 := y^*$: $\quad k_i \doteq f(y^*) + Df(y^*)\, h \sum_{j=1}^{i-1} a_{ij} k_j$

≙ increment equ. for RK-SSM applied to (L) !

→ For small h RK-SSM at state y^* will behave like the same $\underbrace{\text{RK-SSM applied to (L)}}_{\text{amenable to linear model problem analysis}}$ at y^*.

for small timestep the behavior of an explicit RK-SSM applied to $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$ close to the state $\mathbf{y}^*$ is determined by the eigenvalues of the Jacobian $D\,\mathbf{f}(\mathbf{y}^*)$.

How to distinguish stiff initial value problems

An initial value problem for an autonomous ODE $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$ will probably be stiff, if, for substantial periods of time,

$$\min\{\operatorname{Re}\lambda :\ \lambda \in \sigma(D\,\mathbf{f}(\mathbf{y}(t)))\} \ll 0\,, \tag{12.2.15}$$

$$\max\{\operatorname{Re}\lambda, 0 :\lambda \in \sigma(D\,\mathbf{f}(\mathbf{y}(t)))\} \approx 0\,, \tag{12.2.16}$$

where $t \mapsto \mathbf{y}(t)$ is the solution trajectory and $\sigma(\mathbf{M})$ is the spectrum of the matrix $\mathbf{M}$, see Def. 7.1.1.

Examples :

- $\dot{y} = f(y) := \lambda y^2(1-y)\ , \quad \lambda \gg 1$

$$Df(y) = 2\lambda y(1-y) - \lambda y^2 \begin{cases} \ll 0 & \text{if } y \approx 1 \\ \approx 0 & \text{for } y \approx 0 \end{cases}$$

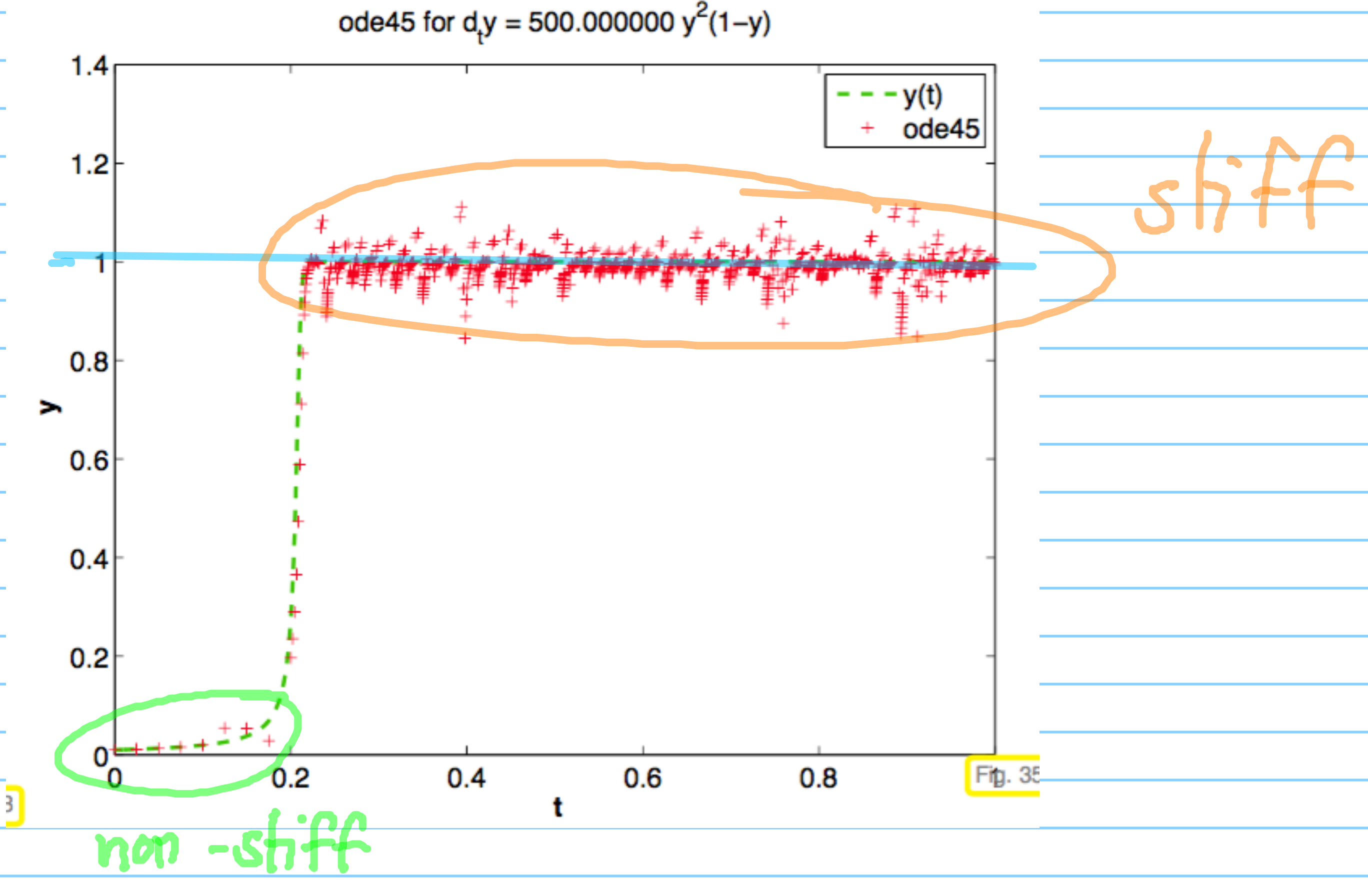


Fig. 35

- $\mathbf{f}(\mathbf{y}) := \begin{bmatrix}0 & -1\\ 1 & 0\end{bmatrix}\mathbf{y} + \lambda(1 - \|\mathbf{y}\|_2^2)\,\mathbf{y}\ , \quad \lambda \gg 1$

$$Df(y) = \begin{bmatrix}0 & -1\\ 1 & 0\end{bmatrix} + \lambda\{-2yy^T + (1 - \|y\|_2^2)I\}$$

$$\|y\|_2 = 1 \Rightarrow \sigma(Df(y)) = -\lambda \pm \sqrt{\lambda^2 - 1}$$

[$\dot{y} = \lambda y(1-y)$, stiffness conditional on y_0]
$\lambda \gg 1$

12.3. Implicit Runge-Kutta single step methods

12.3.1. The implicit Euler method for stiff IVPs

Imp. Eul.: $\quad y_{k+1} = y_k + hf(y_{k+1})$

Linear model problem analysis: apply to $\dot{y} = \lambda y$

$\rightarrow \quad y_{k+1} = \frac{1}{(1-h\lambda)} y_k \quad$ [uniform timestep h]

$\Rightarrow \quad y_{k+1} = \left[\frac{1}{1-h\lambda}\right]^k y_0$

If $\lambda < 0 \Rightarrow \left|\frac{1}{1-h\lambda}\right| < 1 \Rightarrow \lim\limits_{k\to 0} y_k = 0$ for all $h > 0$!

$\rightarrow$ No stability induced timestep constraint

Same result for linear ODEs: $\dot{y} = My$, $M \in \mathbb{R}^{d,d}$
[by diagonalization]

For any timestep, the implicit Euler method generates exponentially decaying solution sequences $(\mathbf{y}_k)_{k=0}^{\infty}$ for $\dot{\mathbf{y}} = \mathbf{M}\mathbf{y}$ with diagonalizable matrix $\mathbf{M} \in \mathbb{R}^{d,d}$ with eigenvalues $\lambda_1, \dots, \lambda_d$, if $\operatorname{Re}\lambda_i < 0$ for all $i = 1, \dots, d$.

Implicit Euler: order 1

12.3.2. Collocation SSM

First step of SSM for IVP: $\dot{y} = f(y)$, $y(0) = y_0$

$\rightarrow y_1$, stepsize h

(i) On $[0,h]$ approximate $y(t) \approx y_h(t) \in V$

$V \hat{=}$ finite-dim. space of functions $[0,h] \rightarrow \mathbb{R}^d$

Standard choice: $V = (\mathcal{P}_s)^d$, $\dim V = s+1$

(ii) Selection of y_h through collocation conditions

$$\left\{\begin{array}{l} y_h(0) = y_0 \;,\quad \dot{y}_h(\tau_j) = f(y_h(\tau_j)) \\ \text{for collocation points } 0 \leq \tau_1 \leq \ldots \leq \tau_s \leq h \end{array}\right.$$

$\rightarrow$ $s+1$ equations

(iii) $y_1 = y_h(h)$

Collocation points from reference interval $[0,1]$: $\tau_j = c_j h$

$$0 \leq c_1 \leq c_2 \leq \ldots \leq c_s \leq 1$$

$$\dot{y}_h \in (\mathcal{P}_{s-1})^d \Rightarrow \dot{y}_h(\tau h) = \sum_{j=1}^{s} \dot{y}_h(hc_j) L_j(\tau) \,,\quad 0 \leq \tau \leq 1$$

$\{L_j\}_{j=1}^{s} \hat{=}$ Lagrange polynomials of degree $s-1$ for node set $\{c_j\}_{j=1}^{s}$: $L_j(c_i) = \delta_{ij}$

from coll. cond.

$$\Rightarrow \quad \dot{y}_h(\tau h) = \sum_{j=1}^{s} f(y_h(c_j h)) L_j(\tau)$$

$\int_0^\xi \ldots d\tau$

$$\Rightarrow \quad y_h(\xi h) - \underbrace{y_h(0)}_{y_0} = h \sum_{j=1}^{s} \underbrace{f(y_h(c_j h))}_{=:\, k_j} \int_0^\xi L_j(\tau)\, d\tau$$

$\xi = c_i$ $\quad y_h(c_i h)$

$\xi = 1$

$$\mathbf{k}_i = f(t_0 + c_i h, \mathbf{y}_0 + h\sum_{j=1}^{s} a_{ij}\mathbf{k}_j) \,, \qquad a_{ij} := \int_0^{c_i} L_j(\tau)\,\mathrm{d}\tau \,,$$

$$\mathbf{y}_1 := \mathbf{y}_h(t_1) = \mathbf{y}_0 + h\sum_{i=1}^{s} b_i\mathbf{k}_i \,. \quad \text{where} \quad b_i := \int_0^1 L_i(\tau)\,\mathrm{d}\tau \,. \qquad (12.3.11)$$

Collocation single step method (CL-SSM)

$\rightarrow$ $s > 1$: k_i to be determined by solving a system of equations $\rightarrow$ implicit method

CL-SSM applied to $\dot{y} = f(t)$, $y(0) = 0$

$$y_1 = h \sum_{i=1}^{s} b_i f(c_i h) \quad , \quad b_i = \int_0 L_i(\tau)\,d\tau$$

$\hat{=}$ a polynomial quadrature formulas $\downarrow$ weights

Convergence of CL-SSM:

$\dot{y} = 10y(1-y)$, $y_0 = 0.01$, $T = 1$

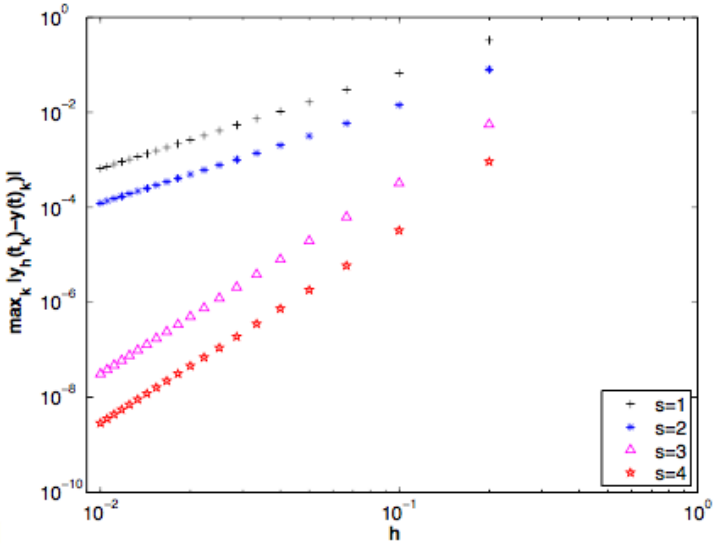
❶ Equidistant collocation points, $c_j = \frac{j}{s+1}$, $j = 1,\cdots,s$.

We observe algebraic convergence with the empiric rates

$s = 1 \;:\; p = 1.96$
$s = 2 \;:\; p = 2.03$
$s = 3 \;:\; p = 4.00$
$s = 4 \;:\; p = 4.04$

} same as orders of polynomial quad. rules with nodes c_j.

Fig. 374



❶ Gauss points in $[0,1]$ as normalized collocations points c_j , $j = 1,\ldots,s$.

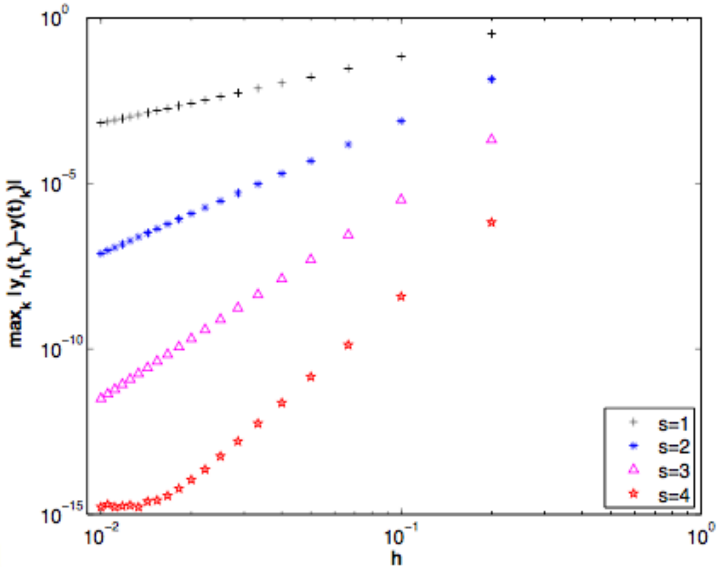
We observe algebraic convergence with the empiric rates

$s = 1 \;:\; p = 1.96$
$s = 2 \;:\; p = 4.01$
$s = 3 \;:\; p = 6.00$
$s = 4 \;:\; p = 8.02$

} order $2s$ = order of s-pt. Gauss quad.

Gauss collocation SSM

Fig. 375



Theorem 12.3.17. Order of collocation single step method [13, Satz .6.40]

Provided that $\mathbf{f} \in C^p(I \times D)$, *the order (→ Def. 11.3.21) of an* s*-stage collocation single step method according to* (12.3.11) *agrees with the order (→ Def. 5.3.1) of the quadrature formula on* $[0,1]$ *with nodes* c_j *and weights* b_j, $j = 1,\ldots,s$.

$$\mathbf{k}_i = f(t_0 + c_i h, \mathbf{y}_0 + h\sum_{j=1}^{s} a_{ij}\mathbf{k}_j)\,, \quad \text{where} \quad a_{ij} := \int_0^{c_i} L_j(\tau)\,\mathrm{d}\tau\,, \qquad (12.3.11)$$

$$\mathbf{y}_1 := \mathbf{y}_h(t_1) = \mathbf{y}_0 + h\sum_{i=1}^{s} b_i\mathbf{k}_i\,. \qquad b_i := \int_0^{1} L_i(\tau)\,\mathrm{d}\tau\,.$$

Definition 11.4.9. Explicit Runge-Kutta method

For $b_i, a_{ij} \in \mathbb{R}$, $c_i := \sum_{j=1}^{i-1} a_{ij}$, $i,j = 1,\dots,s$, $s \in \mathbb{N}$, an s-stage explicit Runge-Kutta single step method (RK-SSM) for the ODE $\dot{\mathbf{y}} = \mathbf{f}(t,\mathbf{y})$, $\mathbf{f}: \Omega \to \mathbb{R}^d$, is defined by ($\mathbf{y}_0 \in D$)

$$\mathbf{k}_i := \mathbf{f}(t_0 + c_i h, \mathbf{y}_0 + h\sum_{j=1}^{i-1} a_{ij}\mathbf{k}_j)\,, \quad i = 1,\dots,s \quad , \quad \mathbf{y}_1 := \mathbf{y}_0 + h\sum_{i=1}^{s} b_i\mathbf{k}_i\,.$$

The vectors $\mathbf{k}_i \in \mathbb{R}^d$, $i = 1,\dots,s$, are called increments, $h > 0$ is the size of the timestep.

Definition 12.3.18. General Runge-Kutta single step method (*cf.* Def. 11.4.9)

For $b_i, a_{ij} \in \mathbb{R}$, $c_i := \sum_{j=1}^{s} a_{ij}$, $i,j = 1,\dots,s$, $s \in \mathbb{N}$, an s-stage Runge-Kutta single step method (RK-SSM) for the IVP (11.1.20) is defined by

$$\mathbf{k}_i := \mathbf{f}(t_0 + c_i h, \mathbf{y}_0 + h\sum_{j=1}^{s} a_{ij}\mathbf{k}_j)\,, \quad i = 1,\dots,s \quad , \quad \mathbf{y}_1 := \mathbf{y}_0 + h\sum_{i=1}^{s} b_i\mathbf{k}_i\,.$$

As before, the $\mathbf{k}_i \in \mathbb{R}^d$ are called increments.

General Butcher scheme notation for RK-SSM

Shorthand notation for Runge-Kutta methods

Butcher scheme ▷

$$\begin{array}{c|c} \mathbf{c} & \mathfrak{A} \\ \hline & \mathbf{b}^T \end{array} := \begin{array}{c|ccc} c_1 & a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & & \vdots \\ c_s & a_{s1} & \cdots & a_{ss} \\ \hline & b_1 & \cdots & b_s \end{array}\,. \qquad (12.3.20)$$

Note: now $\mathfrak{A}$ can be a general $s \times s$-matrix.

K_i obtained by solving increment equations iteratively (simplified Newton method)

↳ freeze Jacobian

Stages : $g_i = h\sum_{j=1}^{s} a_{ij} k_j \Leftrightarrow k_i = f(y_0 + g_i)$

Focus on d=1: $= h\sum a_{ij} f(y_0 + g_j)$

$$\Updownarrow$$

$$\vec{g} = \begin{bmatrix} g_1 \\ \vdots \\ g_s \end{bmatrix} \in \mathbb{R}^s \; : \quad F(\vec{g}) = 0$$

$$F(\vec{g}) = \vec{g} - h\mathcal{A}\begin{bmatrix} f(y_0 + g_1) \\ \vdots \\ f(y_0 + g_s) \end{bmatrix}$$

Initial guess for simplified Newton: (h is small)

$$\vec{g}^{(0)} = 0$$

$$DF(0) = I - h\mathcal{A}\begin{bmatrix} \frac{\partial f}{\partial y}(y_0 + 0) & & \\ & \ddots & \\ & & \frac{\partial f}{\partial y}(y_0 + 0) \end{bmatrix}$$

$$= I - h\frac{\partial f}{\partial y}(y_0)\mathcal{A}$$

▷ Iteration: $$\vec{g}^{(\ell+1)} = \vec{g}^{(\ell)} - \underset{\substack{\uparrow \\ \text{invertible for } h \text{ small enough}}}{DF(\underline{0})^{-1}} F(\vec{g}^{(\ell)})$$

17.12.15 : Recap

- Stiffness
- Implicit Euler methods is unconditionally stable for $\dot{y} = \lambda y$, $\lambda < 0$
- Collocation RK-SSM (includes implicit Euler: $s = 1$, $c_s = 1$)

$$\mathbf{k}_i = f(t_0 + c_i h, \mathbf{y}_0 + h\sum_{j=1}^{s} a_{ij}\mathbf{k}_j)\,, \qquad \text{where} \quad a_{ij} := \int_0^{c_i} L_j(\tau)\,\mathrm{d}\tau\,,$$
$$\mathbf{y}_1 := \mathbf{y}_h(t_1) = \mathbf{y}_0 + h\sum_{i=1}^{s} b_i\mathbf{k}_i\,. \qquad b_i := \int_0^1 L_i(\tau)\,\mathrm{d}\tau\,. \tag{12.3.11}$$

$$s = 1,\ c_1 = 1 \Rightarrow a_{11} = b_1 = \int_0^1 L_1(\tau)d\tau = \int_0^1 1 d\tau = 1$$

12.3.4. Model problem analysis for implicit RK-SSM

Apply RK-SSM to $\dot{y} = \lambda y \Rightarrow y_1 = S(z)\, y_0$

$z := \lambda h$

Definition 12.3.18. General Runge-Kutta single step method (*cf.* Def. 11.4.9)

For $b_i, a_{ij} \in \mathbb{R}$, $c_i := \sum_{j=1}^{s} a_{ij}$, $i,j = 1,\dots,s$, $s \in \mathbb{N}$, an *s*-stage Runge-Kutta single step method (RK-SSM) for the IVP (11.1.20) is defined by

$$\mathbf{k}_i := \mathbf{f}(t_0 + c_i h, \mathbf{y}_0 + h\sum_{j=1}^{s} a_{ij}\mathbf{k}_j)\,,\quad i = 1,\dots,s \quad,\quad \mathbf{y}_1 := \mathbf{y}_0 + h\sum_{i=1}^{s} b_i\mathbf{k}_i\,.$$

As before, the $\mathbf{k}_i \in \mathbb{R}^d$ are called increments.

$f(t,y) := \lambda y$:

▶ $$k_i = \lambda(y_0 + h\sum_{j=1}^{i-1} a_{ij}k_j)\,, \qquad \Rightarrow \quad \begin{bmatrix} \mathbf{I} - z\mathfrak{A} & 0 \\ -z\mathbf{b}^\top & 1 \end{bmatrix} \begin{bmatrix} \mathbf{k} \\ y_1 \end{bmatrix} = y_0 \begin{bmatrix} \mathbf{1} \\ 1 \end{bmatrix}$$
$$y_1 = y_0 + h\sum_{i=1}^{s} b_i k_i$$

▷ $$S(z) := \underbrace{1 + z\mathbf{b}^T(\mathbf{I} - z\mathfrak{A})^{-1}\mathbf{1}}_{\text{stability function}} = \frac{\det(\mathbf{I} - z\mathfrak{A} + z\mathbf{1}\mathbf{b}^T)}{\det(\mathbf{I} - z\mathfrak{A})}\,, \quad z := \lambda h$$

$\uparrow$ $\Downarrow$

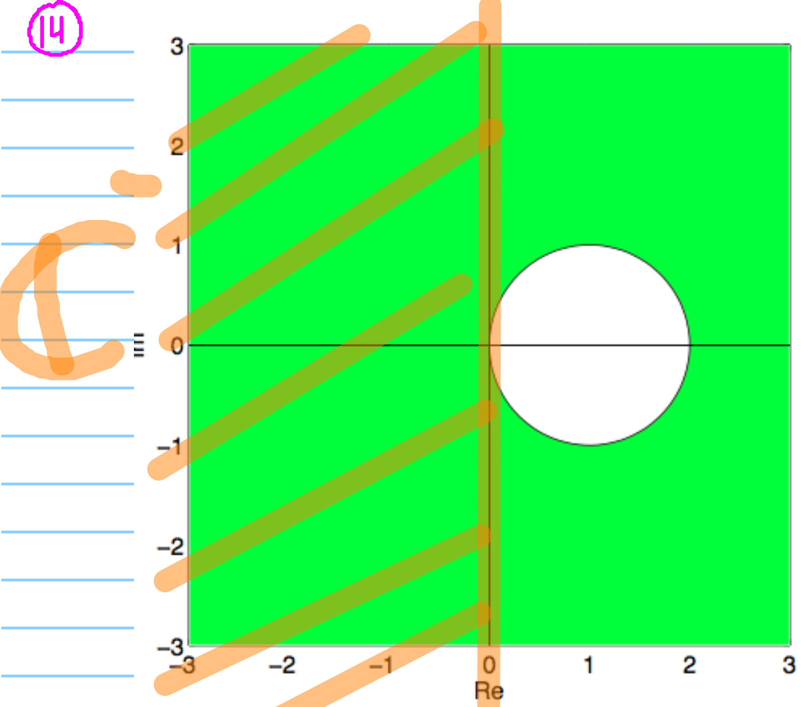
rational function : $S(z) = \dfrac{P(z)}{Q(z)}$, $P, Q \in \mathcal{P}_s$

Example:

- Impl. Euler: $\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array} \Rightarrow S(z) = \dfrac{1}{1-z}$

Region of stability: $\lim_{|z|\to\infty} S(z) = 0$

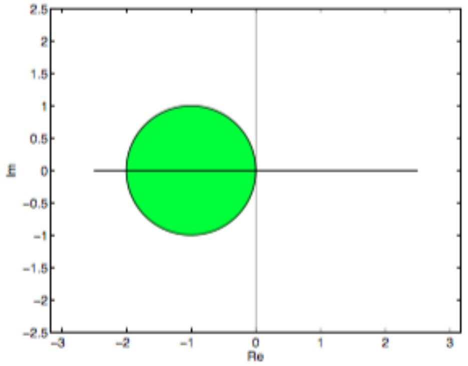
$$S_\Psi := \{z \in \mathbb{C} : |S(z)| < 1\} = \{z \in \mathbb{C} : |z-1| > 1\}$$



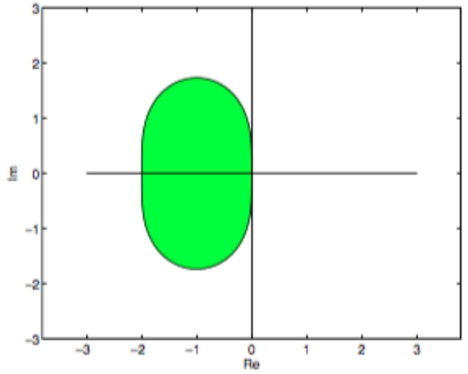
$\mathcal{S}_\Psi$: implicit Euler method (??)

▷ A-stable

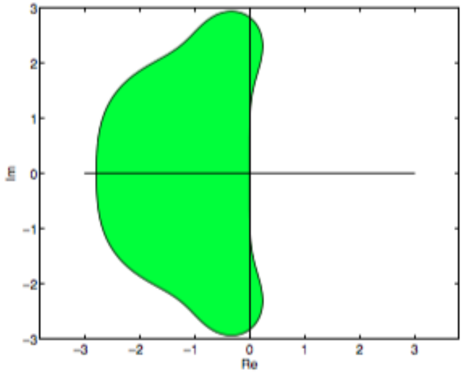
$\Updownarrow$



$\mathcal{S}_\Psi$: explicit Euler (11.2.7)



$\mathcal{S}_\Psi$: explicit trapezoidal method



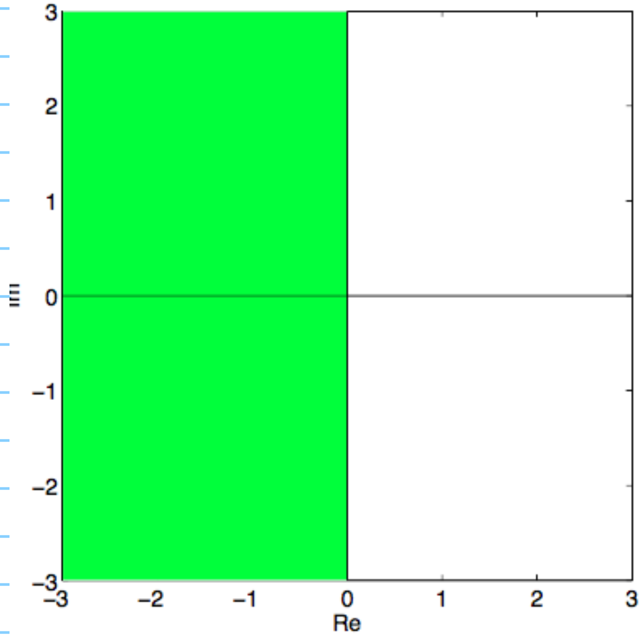
$\mathcal{S}_\Psi$: classical RK4 method

order 2s !

- Implicit midpoint rule: $[\, s = 1,\ c_1 = \frac12,\ b_1 = 1,\ a_{11} = \frac12 \,]$

= Gauss collocation RK-SSM

$$\begin{array}{c|c} \frac12 & \frac12 \\ \hline & 1 \end{array} \Rightarrow S(z) = \frac{1+\frac12 z}{1-\frac12 z}$$



$\mathcal{S}_\Psi$: implicit midpoint method (11.2.18)

$\mathcal{S}_\Psi = \mathbb{C}^-$

→ unconditional stability

→ ideal region of stability :

$\operatorname{Re}\lambda < 0 \Rightarrow h\lambda \in \mathcal{S}_\Psi$

$\Rightarrow$ decaying (y_k)

$\operatorname{Re}\lambda > 0 \Rightarrow |S(h\lambda)| > 1$

$\Rightarrow$ growing (y_k)

$\Updownarrow$

exact match for growth/decay of exact solution $y(t) = Ce^{\lambda t}$

Theorem 12.3.35. Region of stability of Gauss collocation single step methods [13, Satz 6.44]

s-stage Gauss collocation single step methods defined by (12.3.11) with the nodes c_s given by the s Gauss points on $[0,1]$, feature the "ideal" stability domain:

$$\mathcal{S}_\Psi = \mathbb{C}^- . \tag{12.3.34}$$

In particular, all Gauss collocation single step methods are A-stable.

⑮ A-stable = unconditionally stable for decay equ.

Definition 12.3.32. A-stability of a Runge-Kutta single step method

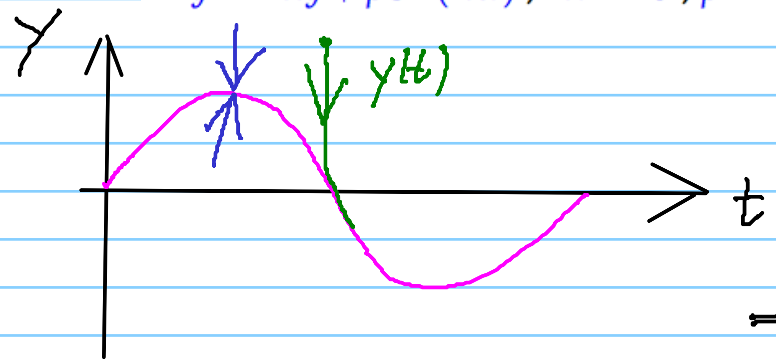
A Runge-Kutta single step method with stability function S is A-stable, if

$$\mathbb{C}^- := \{z \in \mathbb{C}\colon \mathrm{Re}\, z < 0\} \subset \mathcal{S}_\Psi . \quad (\mathcal{S}_\Psi \mathrel{\hat{=}} \text{region of stability Def. 12.1.49})$$

A-stable Runge-Kutta single step methods will not be affected by stability induced timestep constraints when applied to stiff IVP (→ Notion 12.2.9).

The catch :

Example : $\dot{y} = -\lambda y + \beta \sin(2\pi t)$, $\lambda = 10^6, \beta = 10^6$, $y(0) = 1$,



◁ $t \to \sin(2\pi t)$ will strongly attract solution

⇒ stiff problem

$$\dot{\underline{z}} = \begin{bmatrix} -\lambda z_1 + \beta \sin(2\pi z_2) \\ 1 \end{bmatrix} \underline{z} \quad , \underline{z}(t) = \begin{bmatrix} y(t) \\ t \end{bmatrix}$$

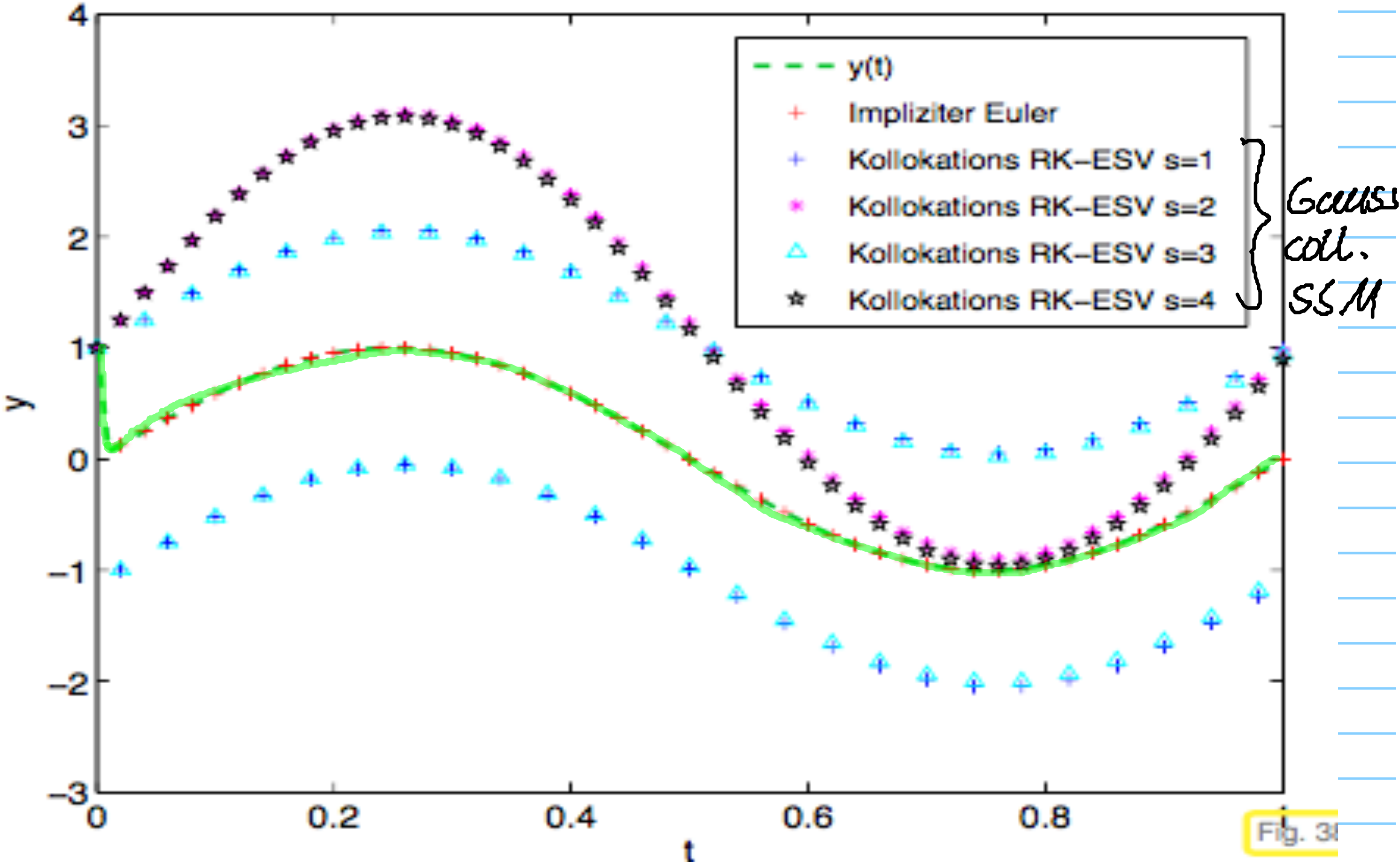


Fig. 3

For Gauss coll. SSM : $\lim_{|z| \to \infty} |S(z)| = 1$

$|\lambda h| \gg 1 \Rightarrow |S(\lambda h)| \approx 1$: $\underbrace{y_k = (S(z))^k y_0}_{\text{for } \dot{y} = \lambda y}$

⇒ Very slow decay of the SSM solution

⇕

Very fast decay of $t \to y(t)$

We want $S(``\infty") = 0$ → GC-SSM $|S(``\infty")| = 1$

↑ ↳ Satisfied for impl. Euler

desirable for very stiff problems

Definition 12.3.38. L-stable Runge-Kutta method → [32, Ch. 77]

A Runge-Kutta method (→ Def. 12.3.18) is L-stable/asymptotically stable, if its stability function (→ Thm. 12.3.27) satisfies

A-stability →

$$(i)\quad \operatorname{Re} z < 0 \;\Rightarrow\; |S(z)| < 1\,, \tag{12.3.39}$$

$$(ii)\quad \lim_{\operatorname{Re} z \to -\infty} S(z) = 0\,. \tag{12.3.40}$$

$$L\text{-stability} = A\text{-stability} + S(``\infty") = 0$$

Construction of L-stable methods:

$$S(z) = 1 + z\underline{b}^T(I - \mathcal{A}z)^{-1}\mathbf{1}$$

$$= 1 + \underline{b}^T(1/z \cdot I - \mathcal{A})^{-1}\mathbf{1}$$

$$\Rightarrow S(\infty) = 1 - \underline{b}^T\mathcal{A}^{-1}\mathbf{1} \overset{!}{=} 0$$

$$\underline{b} = (\mathcal{A})_{j,:} \quad [\,j\text{-th row of } \mathcal{A}\,]$$

$$\Rightarrow \underline{b}^T\mathcal{A}^{-1} = \underline{e}_j^T \Rightarrow \underline{b}^T\mathcal{A}^{-1}\mathbf{1} = 1$$

$S(\infty) = 0$ for all RK-SSM with Butcher scheme

$$\triangleright \quad \begin{array}{c|c} \mathbf{c} & \mathfrak{A} \\ \hline & \mathbf{b}^T \end{array} := \begin{array}{c|ccc} c_1 & a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & & \vdots \\ c_{s-1} & a_{s-1,1} & \cdots & a_{s-1,s} \\ 1 & b_1 & \cdots & b_s \\ \hline & b_1 & \cdots & b_s \end{array}$$

Satisfied for collocation RK-SSM with $c_s = 1$!

$$\mathbf{k}_i = f(t_0 + c_i h, \mathbf{y}_0 + h\sum_{j=1}^{s} a_{ij}\mathbf{k}_j)\,, \qquad \text{where} \quad a_{ij} := \int_0^{c_i} L_j(\tau)\,d\tau\,,$$

$$\mathbf{y}_1 := \mathbf{y}_h(t_1) = \mathbf{y}_0 + h\sum_{i=1}^{s} b_i\mathbf{k}_i\,. \qquad b_i := \int_0^1 L_i(\tau)\,d\tau\,. \tag{12.3.11}$$

= s-stage Gauss-Radau collocation SSM of order 2s-1, L-stable

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

Implicit Euler method

$$\begin{array}{c|cc} \frac{1}{3} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{3}{4} & \frac{1}{4} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$$

Radau RK-SSM, order 3

$$\begin{array}{c|ccc} \frac{4-\sqrt{6}}{10} & \frac{88-7\sqrt{6}}{360} & \frac{296-169\sqrt{6}}{1800} & \frac{-2+3\sqrt{6}}{225} \\ \frac{4+\sqrt{6}}{10} & \frac{296+169\sqrt{6}}{1800} & \frac{88+7\sqrt{6}}{360} & \frac{-2-3\sqrt{6}}{225} \\ 1 & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \\ \hline & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \end{array}$$

Radau RK-SSM, order 5

12.4. Semi-implicit RK-SSM

Idea: Fixed small number of Newton steps to compute increments

⇔ Linearize increment equation

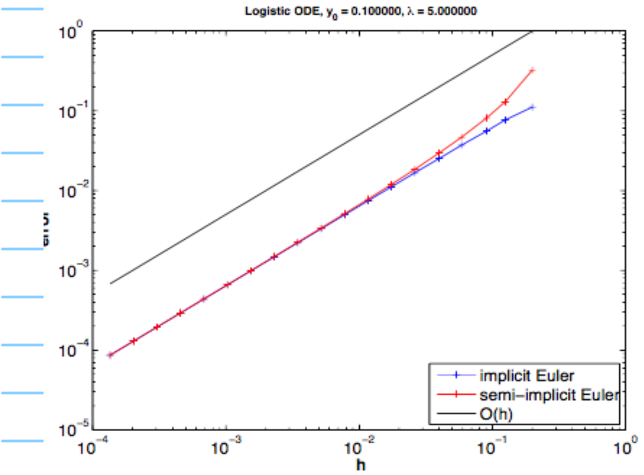
Example: Impl. Euler for $\dot{y} = f(y)$ $y_1 = y_0 + hf(y_1)$

Linearization (around y_0): $y_1 \doteq y_0 + h[f(y_0) + Df(y_0)(y_1 - y_0)]$

$$y_1 = y_0 + \underbrace{(I - hDf(y_0))}_{\text{invertible for sufficiently small } h}{}^{-1} hf(y_0) \qquad (*)$$

(*) = Semi-implicit Euler SSM

$[\ \sigma(I - hDf(y_0)) = 1 - h\sigma(Df(y_0))\ ]$



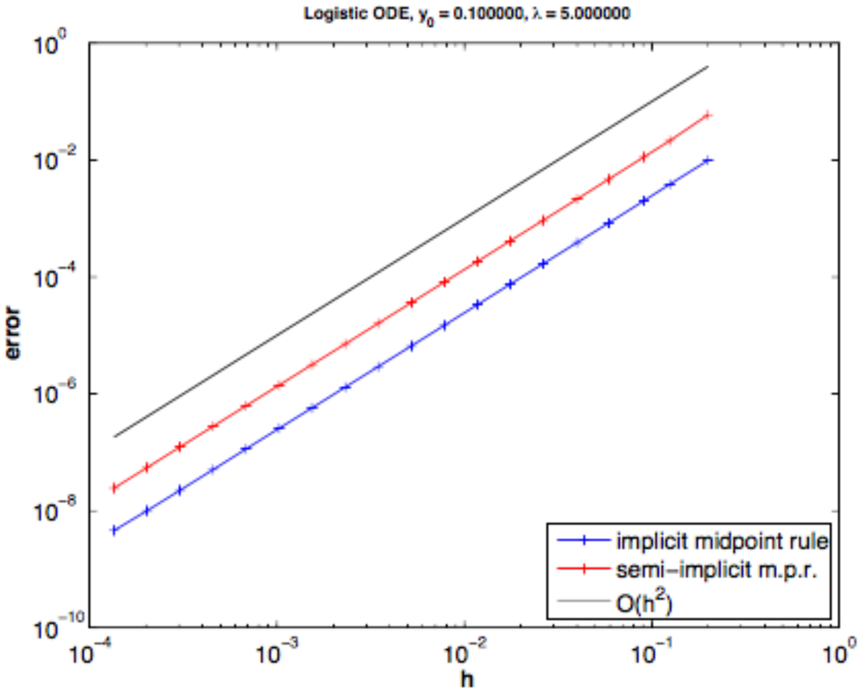
$\dot{y} = 5y(1-y)$

(*) $\hat{=}$ order 1

Semi-implicit midpoint method

Linearization $\left[\begin{array}{l} y_1 = y_0 + hf(\tfrac{1}{2}(y_0 + y_1)) \\ \hookrightarrow y_1 \doteq y_0 + h[f(y_0) + \tfrac{1}{2}Df(y_0)(y_1 - y_0)] \end{array}\right.$

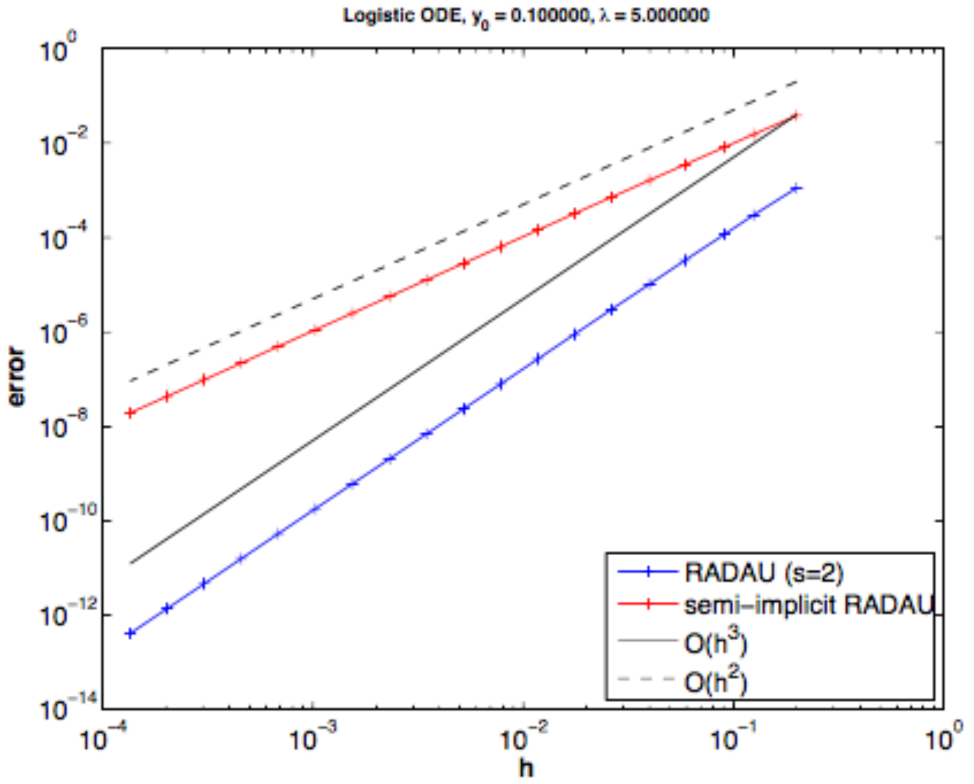
$$(1 - \tfrac{1}{2}hDf(y_0))(y_1 - y_0) = hf(y_0)$$



▷ Order 2 preserved

Is one Newton step enough for all implicit RK-SSM?

NO!



▷ Loss of order

Good news: Fixed number of Newton steps is enough

Class of s-stage semi-implicit (linearly implicit) Runge-Kutta methods (Rosenbrock-Wanner (ROW) methods):

$$(\mathbf{I} - ha_{ii}\mathbf{J})\mathbf{k}_i = \mathbf{f}(\mathbf{y}_0 + h\sum_{j=1}^{i-1}(a_{ij}+d_{ij})\mathbf{k}_j) - h\mathbf{J}\sum_{j=1}^{i-1} d_{ij}\mathbf{k}_j\,, \quad \mathbf{J} = D\,\mathbf{f}(\mathbf{y}_0)\,, \tag{12.4.6}$$

$$\mathbf{y}_1 := \mathbf{y}_0 + \sum_{j=1}^{s} b_j \mathbf{k}_j\,.$$

Practice: Determine a_{ij}, d_{ij}, b_j from order conditions

From a 2015 paper:

$$\dot{\mathbf{u}} = \mathbf{F}(t, \mathbf{u}), \quad \mathbf{u}(0) = \mathbf{u}_0.$$

A Rosenbrock–Wanner (ROW) method with s internal stages can be formulated by

$$\mathbf{k}_i = \mathbf{F}\left(t_m + \alpha_i\tau_m, \tilde{\mathbf{U}}_i\right) + \tau_m J\sum_{j=1}^{i}\gamma_{ij}\mathbf{k}_j + \tau_m\gamma_i\partial_t\mathbf{F}(t_m, \mathbf{u}_m),$$

$$\tilde{\mathbf{U}}_i = \mathbf{u}_m + \tau_m\sum_{j=1}^{i-1}\alpha_{ij}\mathbf{k}_j, \quad i = 1, \ldots, s,$$

$$\mathbf{u}_{m+1} = \mathbf{u}_m + \tau_m\sum_{i=1}^{s} b_i\mathbf{k}_i,$$

where $J := \partial_{\mathbf{u}}\mathbf{F}(t_m, \mathbf{u}_m)$ is the Jacobian of $\mathbf{F}$ w.r.t. $\mathbf{u}$, α_{ij}, γ_{ij}, b_i are the parameters of the method, and

$$\alpha_i := \sum_{j=1}^{i-1}\alpha_{ij}, \qquad \gamma_i := \sum_{j=1}^{i-1}\gamma_{ij}, \qquad \gamma := \gamma_{ii} > 0, \quad i = 1, \ldots, s.$$

$$\text{(A1)} \quad \sum_{i=1}^{s} b_i = 1$$

$$\text{(A2)} \quad \sum_{i=1}^{s} b_i\beta_i = \frac{1}{2} - \gamma$$

$$\text{(A3a)} \quad \sum_{i=1}^{s} b_i\alpha_i^2 = \frac{1}{3}$$

$$\text{(A3b)} \quad \sum_{i,j=1}^{s} b_i\beta_{ij}\beta_j = \frac{1}{6} - \gamma + \gamma^2$$

$$\text{(A4a)} \quad \sum_{i=1}^{s} b_i\alpha_i^3 = \frac{1}{4}$$

$$\text{(A4b)} \quad \sum_{i,j=1}^{s} b_i\alpha_i\alpha_{ij}\beta_j = \frac{1}{8} - \gamma/3$$

$$\text{(A4c)} \quad \sum_{i,j=1}^{s} b_i\beta_{ij}\alpha_j^2 = \frac{1}{12} - \gamma/3$$

$$\text{(A4d)} \quad \sum_{i,j,k=1}^{s} b_i\beta_{ij}\beta_{jk}\beta_k = \frac{1}{24} - \frac{1}{2}\gamma + \frac{3}{2}\gamma^2 - \gamma^3,$$

where we use the abbreviations $\beta_{ij} := \alpha_{ij} + \gamma_{ij}$ and $\beta_i := \sum_{j=1}^{i-1}\beta_{ij}$.

◁ Conditions for order 4

Table 3
Set of coefficients for ROS3PRL2 method.

γ = 4.3586652150845900e−01	
α_{21} = 1.3075995645253771e+00	γ_{21} = −1.3075995645253771e+00
α_{31} = 5.0000000000000000e−01	γ_{31} = −7.0988575860972170e−01
α_{32} = 5.0000000000000000e−01	γ_{32} = −5.5996735960277766e−01
α_{41} = 5.0000000000000000e−01	γ_{41} = −1.5550856807552085e−01
α_{42} = 5.0000000000000000e−01	γ_{42} = −9.5388516575112225e−01
α_{43} = 0.0000000000000000e+00	γ_{43} = 6.7352721231818413e−01
b_1 = 3.4449143192447917e−01	$\hat{b}_1$ = 5.0000000000000000e−01
b_2 = −4.5388516575112231e−01	$\hat{b}_2$ = −2.5738812086522078e−01
b_3 = 6.7352721231818413e−01	$\hat{b}_3$ = 4.3542008724775044e−01
b_4 = 4.3586652150845900e−01	$\hat{b}_4$ = 3.2196803361747034e−01

Order 3 ROW method, L-stable

How to explore L-stability of a ROW method ?

(i) Apply SSM to $\dot{y} = \lambda y$ [$f(y) = \lambda y$]

(ii) We get $y_1 = S(\lambda h) y_0$

[More general $Q(\lambda h) y_1 = P(\lambda h) y_0$, P, Q polynomials]

(iii) Verify $\lim_{z \to \infty} S(z) = 0$

(iv) Show $\mathrm{Re}\, z < 0 \Rightarrow |S(z)| < 1$

Use Thm : If S rational, defined on $\mathbb{C}^- \cup i\mathbb{R}$, $\lim_{z \to \infty} S(z) = 0$

$$\Rightarrow |S(z)| < \sup_{\tau \in \mathbb{R}} |S(i\tau)| \quad \text{for all } z \in \mathbb{C}^-$$

Remark:

"ideal stability function"

$$\dot{y} = \lambda y$$

$$y(h) = e^{\lambda h} y(0)$$

$$\Updownarrow \qquad y_0 = y(0)$$

$$y_1 = S(\lambda h) y_0$$

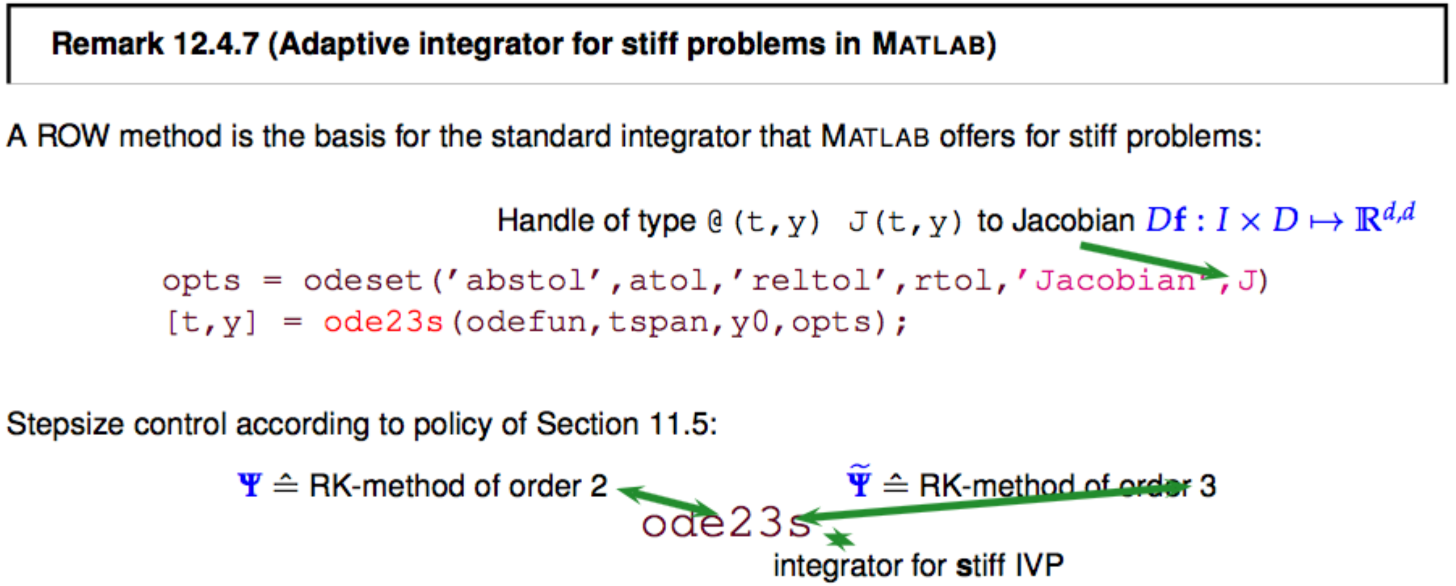
For SSM : $S(z) \sim e^z$ for $|z| \ll 1$

SSM of order p

$$\Rightarrow \quad S(z) - e^z = O(|z|^{p+1}) \quad \text{for } z \to 0$$

Summary :

- Stiff IVP
- Stability induced timestep constraint
- A-stability & L-stability
- Methods : (semi-) implicit RK-SSM usually in embedded form



Remark 12.4.7 (Adaptive integrator for stiff problems in MATLAB)

A ROW method is the basis for the standard integrator that MATLAB offers for stiff problems:

Handle of type `@(t,y) J(t,y)` to Jacobian $\mathbf{D}f : I \times D \mapsto \mathbb{R}^{d,d}$

```
opts = odeset('abstol',atol,'reltol',rtol,'Jacobian',J)
[t,y] = ode23s(odefun,tspan,y0,opts);
```

Stepsize control according to policy of Section 11.5:

$\Psi \hat{=}$ RK-method of order 2 $\qquad \widetilde{\Psi} \hat{=}$ RK-method of order 3

`ode23s`

integrator for stiff IVP