Karlsruher Institut für Technologie

Department of Mathematics
Institute for Applied and
Numerical Mathematics

JProf. Dr. Katharina Schratz
Dr. Patrick Krämer
Dr. Simon Baumstark

Numerical Integrators for Nonlinear Dispersive Equations — Exercise Sheet 01 November 2, 2018

In the lecture "Comparison Numerical Integrators for Nonlinear Dispersive Equations", the nonlinear Schrödinger equation

$$i\partial_t u = -\Delta u + f(u), \quad u(0,x) = u^0(x) \in \mathbb{C}, \quad (t,x) \in [0,T] \times \mathbb{T}^d, \tag{NLS}$$

with polynomial nonlinearity $f(u)$, equipped with periodic boundary conditions on the d-dimensional torus $\mathbb{T}^d = [-\pi,\pi]^d$, serves as a model equation in order to construct, analyze and compare numerical schemes for the time integration of time-dependent partial differential equations (PDEs). We follow the *method of lines*, i.e. we firstly discretize the spatial differential operator $\Delta = \partial_{x_1}^2 + \cdots + \partial_{x_d}^2$ in space and then apply time integration schemes for the solution of the spatially discretized version of (NLS), which has the structure of an ordinary differential equation (ODE). The main challenge in the construction of numerical time integration schemes for equations of type (NLS) thereby lies in the fact, that for a given spatial mesh size $h = \frac{2\pi}{N_x}$ (according to a discretization of $\mathbb{T}$ with $N_x \in \mathbb{N}$ grid points) the discretized version of the spatial operator Δ behaves like $\frac{1}{h^2}$. Note that the mesh size $h \ll 1$ may be very small and thus the discretized version of (NLS) becomes a stiff ODE. Recall that ODEs are called stiff, if explicit integration schemes fail to so solve them for large step sizes.

In order to overcome these challenges, we construct and analyze splitting methods and exponential-type integrators for the solution of (NLS) throughout the lecture.

The following exercises discuss the construction and error analysis of splitting schemes for ODEs of type

$$iy'(t) = Ay(t) + f(y(t)), \quad y(0) = y^0 \in \mathbb{C}^N, \qquad \text{for all} \quad t \in [0,T], \quad T > 0 \tag{1}$$

where $A \in \mathbb{R}^{N\times N}$ is a regular, diagonalisable matrix with eigenvalues $0 < \omega_1 \le \omega_2 \le \cdots \le \omega_N \in \mathbb{R}$ and where $f : \mathbb{C}^N \to \mathbb{C}^N$ is smooth. The latter equation (1) can be seen as a spatially discretized version of (NLS).

Exercise 1: (Matrix Exponentials and Commutators)

(a) Let $A \in \mathbb{R}^{N\times N}$ be a diagonalisable matrix with $A = S^{-1}DS$, where $D = \operatorname{diag}(d_1,\ldots,d_N) \in \mathbb{R}^{N\times N}$ is a diagonal matrix containing the eigenvalues of A, and where $S \in \mathbb{R}^{N\times N}$ is regular. Show that

$$\exp(A) := \sum_{k=0}^{\infty} \frac{A^k}{k!} = S^{-1}\exp(D)S, \quad \text{where} \quad \exp(D) = \operatorname{diag}(e^{d_1},\ldots,e^{d_N}).$$

Hint: Taylor series expansion of the exponential function. *Remark: Note that the idea of writing*

$$g(A) = S^{-1}g(D)S, \quad \textit{where} \quad g(D) = \operatorname{diag}(g(d_1),\ldots,g(d_N)).$$

works for all functions g, for which its corresponding Taylor series expansion of g converges in d_j, $j = 1,\ldots,N$. Typical examples for g are any polynomial, cosine, sine, exponential, square root and many more functions.

(b) Explicitly compute $\exp(A)$ for

$$A = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$$

Hint: *The eigenvalues of A are 1 and 3.*

(c) Now consider another matrix

$$B = \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix}, \qquad a,b \in \mathbb{R}.$$

Show that the *commutator* of the matrix A from part b) and B satisfies

$$[A,B] := AB - BA \neq 0 \qquad \text{for } a \neq b.$$

next page →

In the following exercise we discuss the construction and the analysis of so-called Lie splitting schemes.

Exercise 2: (Lie splitting integration scheme)

Consider the linear ODE for $t \in [0, T]$

$$y'(t) = Ay(t) + By(t) = (A+B)y(t), \quad y(0) = y^0 \in \mathbb{C}^N, \qquad A, B \in \mathbb{C}^{N\times N}. \tag{2}$$

We split the latter equation into the subproblems

$$w'(t) = Aw(t), \quad w(0) = w^0, \qquad \text{and} \qquad z'(t) = Bz(t), \quad z(0) = z^0$$

with exact *flows*

$$\varphi_A^t(w^0) := e^{tA}w^0, \qquad \text{and} \qquad \varphi_B^t(z^0) = e^{tB}z^0,$$

respectively. The Lie splitting scheme for solving (2) with step size τ then reads

$$y_{\text{Lie}}^{n+1} = \Phi_{\text{Lie}}^{\tau}(y_{\text{Lie}}^n) = (\varphi_A^\tau \circ \varphi_B^\tau)(y_{\text{Lie}}^n) = e^{\tau A}e^{\tau B}y_{\text{Lie}}^n, \tag{Lie}$$

where y_{Lie}^n is an approximation to the exact solution of (2)

$$y(t^n) = e^{t_n(A+B)}y(0), \qquad t_n = n\tau,\ n = 0, 1, 2, \ldots.$$

(a) Show that the local error of Φ_{Lie}^{τ} satisfies

$$\left\| y(\tau) - y_{\text{Lie}}^1 \right\| \leq K\tau^2 \left\| [A, B]\, y_0 \right\|,$$

where K is a constant independent of $\tau,\ A,\ B$.
Hint: *Compare the terms $e^{\tau A}e^{\tau B}$ and $e^{\tau(A+B)}$ using Taylor series expansion.*

(b) Under the additional stability assumptions

$$\left\| e^{tM} \right\| \leq e^{t\omega_M}, \quad \omega_M \in \mathbb{R}, \quad M \in \{A,\ B,\ A+B\},$$

show that the global error of Φ_{Lie}^{τ} satisfies

$$\| y(t_n) - y_{\text{Lie}}^n \| \leq K\tau \left\| [A, B]\, y_0 \right\|,$$

where K is a constant independent of $\tau,\ A,\ B$.
Hint: *Lady Windermere's fan: local error + stability = convergence.*

(c) How does the global error change, if $[A, B] = 0$?

next page →

Programming Exercise 1: (Euler and Splitting schemes for ODEs)

In order to better understand the stiffness effect for explicit integration schemes, we focus on the linear ODE problem

$$iy'(t) = Ay(t) + f(y(t)), \quad y(0) = y^0 \in \mathbb{C}^N, \qquad \text{for all} \quad t \in [0,T], \quad T > 0 \tag{3}$$

where $A \in \mathbb{R}^{N\times N}$ is a regular, diagonalisable matrix with eigenvalues $0 < \omega_1 \le \omega_2 \le \cdots \le \omega_N \in \mathbb{R}$ and where $f : \mathbb{C}^N \to \mathbb{C}^N$ is a smooth function. In the following exercise, we implement the explicit Euler scheme

$$y^{n+1}_{\text{Eul}} = \Phi^\tau_{\text{Euler}}(y^n) = y^n_{\text{Eul}} + \tau(-i)\left(Ay^n_{\text{Eul}} + f(y^n_{\text{Eul}})\right).$$

(a) This exercise underlines the failure of explicit methods for stiff ODEs. Let $A = \omega = 32$ and $f \equiv 0$ and let $T = 1$. Furthermore let $y(0) = 1$. In MATLAB (or Python), write a script file which applies the explicit Euler scheme Φ^τ_{Euler} to (3) with step size $\tau = \frac{1}{50}$. Plot the real part of the numerical solution y^n_{Eul} and the real part of the exact solution $y(t_n) = e^{-it_n\omega}$ over all times $t_n = n\tau$ together in one figure. What happens with the numerical solution y^n_{Eul} over time?

(b) Now consider (3) in the linear matrix case with (cf. Exercise 1 b)c))

$$A = \omega\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}, \quad \omega = 5, \qquad f(y) = By, \qquad B = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}.$$

Furthermore consider $T = 2$, the time step size $\tau = \frac{1}{100}$ and the initial value $y(0) = y^0 = (1,0)^\top$.

α) Adapt the explicit Euler implementation from part a) such that it applies to the matrix case with the given matrices A, B. Note that the exact solution of the problem at time $t_n = n\tau$ now reads

$$y(t) = \exp(-it_n(A+B))y(0).$$

For the evaluation of the matrix exponential, you can use the MATLAB function `expm`.

β) Additionally, implement the Lie splitting scheme Φ^τ_{Lie} from (Lie) into the same program.

γ) Plot the real part of the first component of the exact solution $y(t_n)$, of the explicit Euler approximation y^n_{Eul} and of the Lie splitting approximation y^n_{Lie} over all times $t_n = n\tau$ into one figure.

δ) In another figure (or subplot) plot the evolution of the Eucledian error for both methods

$$\text{err}_\#(t_n) = |y(t_n) - y^n_\#| = \sqrt{|y_1(t_n) - y^n_{\#,1}|^2 + |y_2(t_n) - y^n_{\#,2}|^2}, \qquad \# \in \{\text{Eul}, \text{Lie}\}$$

over all times $t_n = n\tau$ into a `semilogy` plot. Discuss the behaviour of both errors over time.

(c) Now fix $T = 1$ and consider the matrix case with the matrices and initial values from part b). Extend your code from part b) and create double-logarithmic order plots for the schemes Φ^τ_{Eul} and Φ^τ_{Lie} applied to (3). Moreover choose time step sizes τ_m, $m = 1, 2, \ldots 10$ with

$$\tau_m = \frac{T}{N_T^m}, \quad \text{where} \qquad N_T = (N_T^1, \ldots, N_T^{M_\tau}) = T \cdot 2^m.$$

For creating the order plots proceed as follows:

Create an array `tau_array` containing all time step sizes τ_m, $m = 1, 2, \ldots, M_\tau$ and in additional arrays `errEul_array` and `errLIE_array` for the explicit Euler and Lie splitting time integration scheme, respectively, save for each τ_m the maximum error over all times t_n

$$\text{err}^{\max}_\# = \max_{t_n \in [0,T]} \text{err}_\#(t_n), \qquad \# \in \{\text{Eul}, \text{Lie}\}$$

where $y^n_\#$ denotes the numerical approximation to the exact solution $y(t_n)$ at time t_n. Plot the errors `err{#}_array` versus the time steps `tau_array` in a `loglog` plot and add a line of slope 1 corresponding to order 1 convergence, respectively. Why may it not be enough in our case to only consider the error at the end time $t_{N_{T,m}} \approx T$? ***Hint:*** *Note that for part c) you do not have to save the solutions for all times t_n.*

Discussion in the problem class monday 8:00 am, in room 3.061 in the Kollegiengebäude Mathematik 20.30.