Karlsruher Institut für Technologie

Department of Mathematics
Institute for Applied and
Numerical Mathematics

JProf. Dr. Katharina Schratz
Dr. Patrick Krämer

Highly Oscillatory Problems — Exercise Sheet 05 July 3, 2018

Consider the nonlinear Klein–Gordon equation with a smooth nonlinearity $f(z)$ satisfying $f(\overline{z}) = \overline{f(z)}$

$$\partial_{tt} z(t,x) = -\underbrace{c^2(-\Delta + c^2)}_{=(c\langle\nabla\rangle_c)^2} z(t,x) + c^2 \lambda f(z(t,x)), \quad z(0,x) = \varphi(x), \quad \partial_t z(0,x) = c^2 \gamma(x), \qquad c \gg 1 \tag{KG}$$

with solution $z(t,x) \in \mathbb{C}$ for $t \in [0,T]$ and $x \in [-\pi,\pi] =: \mathbb{T}$, equipped with periodic boundary conditions. Furthermore, recall that by the ansatz of writing (we leave out the explicit dependence on x in the following)

$$\begin{cases} u(t) = z(t) - ic^{-1} \langle\nabla\rangle_c^{-1} \partial_t z(t), \\ v(t) = \overline{z}(t) - ic^{-1} \langle\nabla\rangle_c^{-1} \overline{\partial_t z}(t), \end{cases} \qquad z(t) = \frac{1}{2}\left(u(t) + \overline{v}(t)\right)$$

we derived the coupled first order in time equation for $u(t)$ and $v(t)$

$$\begin{cases} i\partial_t u(t) = -c\langle\nabla\rangle_c u(t) + \lambda c \langle\nabla\rangle_c^{-1} f\left(\frac{1}{2}\left(u(t) + \overline{v}(t)\right)\right), & u(0) = \varphi - ic\langle\nabla\rangle_c^{-1}\gamma, \\ i\partial_t v(t) = -c\langle\nabla\rangle_c v(t) + \lambda c \langle\nabla\rangle_c^{-1} f\left(\frac{1}{2}\left(\overline{u}(t) + v(t)\right)\right), & v(0) = \overline{\varphi} - ic\langle\nabla\rangle_c^{-1}\overline{\gamma}. \end{cases} \tag{1}$$

Construction of uniformly accurate exponential-type integrators:

Based on the following asymptotic approximation result for the solution z of (KG)

$$z(t) = z_0(t) + \mathcal{O}\left(c^{-2}\right), \qquad z_0(t) = \frac{1}{2}\left(e^{ic^2t} u_0(t) + e^{-ic^2t}\overline{v_0}(t)\right), \qquad \text{i.e.} \qquad z(t) \to z_0(t),\ c \to \infty,$$

we were able to construct efficient numerical schemes with time step size $0 < \tau < 1$ by numerically solving

the c-independent nonlinear Schrödinger equations for ansatz functions u_0, v_0 (cf. [Mas2002]),

which yield numerical approximations $z_0^n \approx z_0(t_n)$ at time t_n satisfying error bounds of order $\mathcal{O}\left(\tau^2 + c^{-2}\right)$. In particular, these error bounds imply that these schemes, applied to (KG) with a fixed time step size τ, yield

good results in the regime of large values $\quad c > \frac{1}{\tau}$ (why?).

However, in the

intermediate regime $\quad 1 \leq c < \frac{1}{\tau}$,

the $\mathcal{O}\left(c^{-2}\right)$ error term restricts the reachable accuracy for this scheme such that we have to develop a different strategy in order to approximate the exact solution. The ansatz of "twisting" the variables u, v satisfying (1) with the oscillatory phase e^{-ic^2t} thus yields the new "twisted" variables

$$u_*(t) = e^{-ic^2t} u(t) \quad \text{and} \quad v_*(t) = e^{-ic^2t} v(t) \quad \text{such that} \quad z(t) = \frac{1}{2}\left(e^{ic^2t} u_*(t) + e^{-ic^2t}\overline{v_*}(t)\right). \tag{2}$$

next page →

$\langle\nabla\rangle_c := \sqrt{-\Delta + c^2}$

[Mas2002] - N. Masmoudi and K. Nakanishi, *From nonlinear Klein–Gordon equation to a system of coupled nonlinear Schroedinger equations*, Math. Ann. 2002

Exercise 10: (Twisted First Order System and a Uniformly Accurate Integration Scheme)

Assume that the functions u, v satisfy system (1).

(a) Show that the "twisted" variables u_*, v_* defined in (2) satisfy the system

$$\begin{cases} i\partial_t u_*(t) = -\mathcal{A}_c u_*(t) + \lambda c \langle\nabla\rangle_c^{-1} e^{-ic^2t} f\left(\frac{1}{2}\left(e^{ic^2t}u_*(t) + e^{-ic^2t}\overline{v_*}(t)\right)\right), & u_*(0) = u(0), \\ i\partial_t v_*(t) = -\mathcal{A}_c v_*(t) + \lambda c \langle\nabla\rangle_c^{-1} e^{-ic^2t} f\left(\frac{1}{2}\left(e^{-ic^2t}\overline{u_*}(t) + e^{ic^2t}v_*(t)\right)\right), & v_*(0) = v(0), \end{cases} \quad (3)$$

where we define the operator $\mathcal{A}_c := c\langle\nabla\rangle_c - c^2$.

(b) Assume that you already know the solution $u_*(t_n), v_*(t_n)$ of the latter system at time t_n.
Apply Duhamel's formula to (3) in order to write down the exact solution $u_*(t_n+\tau), v_*(t_n+\tau)$ at time $t_n+\tau$.

(c) Let $r > 0$ and $w \in H^{r+2}(\mathbb{T})$ be arbitrary. Show that

$$\alpha)\quad \|\mathcal{A}_c w\|_{H^r} = \left\|(c\langle\nabla\rangle_c - c^2)w\right\|_{H^r} \le \frac{1}{2}\|w\|_{H^{r+2}} \quad \text{for all } c \in \mathbb{R}.$$

$$\beta)\quad \left\|(e^{-is\mathcal{A}_c} - 1)w\right\|_{H^r} \le |s|\frac{1}{2}\|w\|_{H^{r+2}}, \quad \text{for all } s \in \mathbb{R}.$$

$$\gamma)\quad \left\|e^{is\mathcal{A}_c}\right\|_{H^r\to H^r} = 1, \quad \text{for all } s \in \mathbb{R}.$$

Hint: Recall that for $w \in H^r$, we have $\|w\|_{H^r} = \sum_{k\in\mathbb{Z}}\left(1+|k|^2\right)^r|\widehat{w_k}|^2$ (see Exercise Sheet 3).

In the following, consider the cubic nonlinearity $f(z) = |z|^2 z$.

(d) Explicitly determine the coefficient functions $f^m_{u_*}(u_*, v_*),\ m \in \mathbb{Z}$, such that

$$e^{-ic^2t} f\left(\frac{1}{2}\left(e^{ic^2t}u_*(t) + e^{-ic^2t}\overline{v_*}(t)\right)\right) = \sum_{m\in\mathbb{Z}} e^{imc^2t} f^m_{u_*}(u_*, v_*).$$

Hint: Exploit the results of Exercise 8a) from Exercise Sheet 4.

(e) Let $r > d/2$. Show that the derivatives $\partial_t u_*, \partial_t v_*$ can be bounded uniformly with respect to c. More precisely, show that

$$\|u_*(t_n+s) - u_*(t_n)\|_{H^r} \le \frac{1}{2}|s|\,\|u_*(t_n)\|_{H^{r+2}} + \frac{1}{8}|s|\sup_{0\le\xi\le s}\left(\|u_*(t_n+\xi)\|_{H^{r+2}} + \|v_*(t_n+\xi)\|_{H^r}\right)^3.$$

and analogously for v_* interchanging $u_* \leftrightarrow v_*$.
Hint: Use part *b*) and *c*) and that for $r > d/2$ and $u, v \in H^r$ there holds $\|uv\|_{H^r} \le K\|u\|_{H^r}\|v\|_{H^r}$.

(f) Based on the Duhamel's formula (part b)) and the bounds from part c) and e), construct an exponential-type integration scheme Φ^τ_{uni} given through

$$\begin{pmatrix} u_*^{n+1} \\ v_*^{n+1} \end{pmatrix} = \Phi^\tau_{\text{uni}}\begin{pmatrix} u_*^{n+1} \\ v_*^{n+1} \end{pmatrix},$$

satisfying global first order in time convergence bounds towards the exact solution $(u_*(t_{n+1}), v_*(t_{n+1}))$ at time $t_n = n\tau,\ n = 0, 1, 2, \ldots, \lfloor T/\tau \rfloor$, which hold uniformly for all $c \ge 1$.

(g) Assume that $r > d/2$ and that $\sup_{0\le t\le T}\|u_*(t)\|_{H^r+2} + \|v_*(t)\|_{H^r+2} \le M_2$ unifomly in c. For the numerical approximation

$$z^n = \frac{1}{2}\left(e^{ic^2t_n}u_*^n + e^{-ic^2t_n}\overline{v_*^n}\right)$$

obtained via the scheme Φ^τ_{uni}, prove the following uniform in c first order in time global error bound

$$\|z(t_n) - z^n\|_{H^r} \le \tau\mathcal{M}^{\text{glob}},$$

where $\mathcal{M}^{\text{glob}}$ only depends on T, r, M_2 but not on the parameter c.

Programming Exercise 6: (Implementation of the Uniformly Accurate Scheme)

In this programming exercise we numerically underline the uniformly accurate in $c \geq 1$ first order in time convergence bound of the scheme Φ^{τ}_{uni} from Exercise 10 applied to the nonlinear Klein–Gordon equation (KG), where $f(z) = |z|^2 z$.

More precisely, we consider the Klein–Gordon equation (KG) with $\lambda = 0.5$ and the cubic nonlinearity $f(z) = |z|^2 z$ on the torus $\mathbb{T} = [-\pi, \pi]$ (i.e. periodic boundary conditions) and for $t \in [0, T]$. Furthermore we fix the (complex valued) initial data

$$\varphi(x) = \frac{\cos(x) + i\sin(2x)}{2 - \sin(x)}, \qquad \gamma(x) = \frac{\sin(2x) + i\cos(x)}{2 - \cos(x)}.$$

For practical implementation issues, we consider the discretization of space $x_j = jh,\ j = -\frac{N_x}{2} + 1, \ldots \frac{N_x}{2}$ with $N_x = 64$, i.e. $h = \frac{2\pi}{N_x}$, and of time $t_n = n\tau,\ n = 0, 1, 2, \ldots \lfloor T/\tau \rfloor =: N_T$ for time step sizes

$$\tau = \tau_m = \frac{T}{N_T^m}, \quad \text{where} \quad N_T^m, m = 1, 2, \ldots, m_{\max}$$

is the m-th divisor of the number $N_T^{m_{\max}} = 120$. All divisors of a number $N \in \mathbb{N}$ can be found in MATLAB (since version R2014b) via the function `divisors()`. Moreover, we consider the c values $c_\ell = 1.45^\ell,\ \ell = 0, 1, \ldots, M_c$, where at first we consider $M_c = 10$.

(a) Implement the scheme Φ^{τ}_{uni} (see Exercise 10) into the (final) code from Programming Exercise 5.

(b) Proceed as in Programming Exercise 5, in order to create order plots for the scheme Φ^{τ}_{uni} for the convergence in the time steps τ_m as well as in the c values c_ℓ. Thereby as a reference scheme (as in Programming Exercise 5), use the exponential integrator $\Phi^{\tau_{\text{ref}}}_{\exp}$ (see Exercise Sheet 4) with a very fine time step size $\tau_{\text{ref}} = \tau_{m_{\max}} / M_{\text{ref}}$ with $M_{\text{ref}} = 200$.

(c) For the choices c_ℓ with $\ell \in \{1, \lfloor \frac{M_c}{3} \rfloor, \lfloor \frac{2M_c}{3} \rfloor, M_c\}$, create efficiency plots for each of the methods, which are implemented in your code (recall that in addition to the scheme Φ^{τ}_{uni} in Programming Exercise 5, we implemented an exponential integrator as well as a Strang Splitting Limit Approximation).
Thereby proceed as follows:

- For each numerical scheme in the code from part b) measure the CPU time $T^{\ell,m}_{\text{elapsed}}$ which elapsed during the computation for **every** choice (c_ℓ, τ_m). For this purpose you can use the MATLAB routines `mycputime = tic()` (get current system time) and `toc(mycputime)` (compute the difference between the current system time and `mycputime`). Note that $T^{\ell,m}_{\text{elapsed}}$ is an element of an array of same dimension as the array corresponding to the measured errors $\text{err}^{\ell,m}_{\max}$ (see Exercise Sheet 3).
- Pick a particular $\ell \in \{1, \lfloor \frac{M_c}{3} \rfloor, \lfloor \frac{2M_c}{3} \rfloor, M_c\}$ and plot the pairs $(T^{\ell,m}_{\text{elapsed}}, \text{err}^{\ell,m}_{\max})$ (corresponding to c_ℓ) in a double-logarithmic `loglog` plot, where all three considered schemes share one common plot.
- Since we consider 4 different values of c_ℓ, $\ell \in \{1, \lfloor \frac{M_c}{3} \rfloor, \lfloor \frac{2M_c}{3} \rfloor, M_c\}$ in the efficiency plots, we can use `subplot(2,2,`*`index`*`)`, where *index* $= 1, 2, 3, 4$, in order to visualize the efficiency of the schemes for all chosen values c_ℓ at once.

Note that in our case, efficiency means that a scheme allows a small error at low CPU time, i.e. in the efficiency plot the corresponding values shall be in the lower left corner.

Because the error constant of our reference scheme $\Phi^{\tau_{\text{ref}}}_{\exp}$ still heavily depends on the large parameter c^2, even with a small reference time step τ_{ref}, the quality of the corresponding reference solution becomes very bad, as soon as c gets larger than a specific threshold.

(d) In order to cross-check our three schemes, implement in the above code the uniformly accurate scheme $\Phi^{\tau_{\text{ref}}}_{\text{uni}}$ with reference time step τ_{ref} as an additional reference scheme, such that you can easily switch between the two reference schemes $\Phi^{\tau_{\text{ref}}}_{\text{uni}}$ and $\Phi^{\tau_{\text{ref}}}_{\text{uni}}$.
Hint: You can use an `if / else` condition or also (if you use MATLAB) the `switch` command.

(e) How do your results change for the choice $\Phi_{\text{uni}}^{\tau_{\text{ref}}}$ as a reference scheme?

(f) Repeat your simulation for a larger number of c values, setting $M_c = 20$ and using $\Phi_{\text{uni}}^{\tau_{\text{ref}}}$ as a reference scheme.

Discussion in the problem class friday 11:30 am, in room 2.058 in the Kollegiengebäude Mathematik 20.30.