Runge-Kutta based local time-stepping methods for forced wave equations

Carina Santos

University of Basel

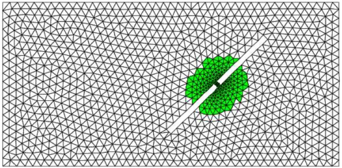
November 16, 2020

Overview

What is local time-stepping?

Local times-stepping, i.e advancing elements by their maximum locally defined time step, is used when the elements in the mesh differ greatly in size. This helps to reduce the computing time. But to ensure the stability of the method, one must then select the time step depending on the smallest local step.

What is local time-stepping?



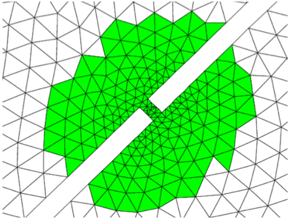


Figure: geometric features

The (Damped) Wave Equation

We now consider the time-dependent wave equation,

$$\begin{aligned} u_{tt} + \sigma u_t - \nabla \cdot (c^2 \nabla u) = f \quad & \text{in } \Omega \times (0,T) \\ u(\cdot,t) = 0 \quad & \text{on } \partial\Omega \times (0,T) \\ u(\cdot,0) = u_0, \; u_t(\cdot,0) = v_0 \quad & \text{in } \Omega. \end{aligned} \tag{1}$$

a standard model problem for wave phenomena.

- Ω bounded domain in $\mathbb{R}$, $c(x) > 0$, $\sigma(x) \geq 0$
- $f \in L^2\left(0,T; L^2(\Omega)\right)$ a (known) source term
- $u_0 \in H_0^1(\Omega)$, $v_0 \in L^2(\Omega)$ prescribed initial conditions

Weak formultaion

We now multiply our problem (1) with a test function $v \in C_0^\infty(\Omega)$:
Find $u : [0, T] \to H_0^1(\Omega)$ such that

$$\begin{aligned} (u_{tt}, v) + (\sigma u_t, v) + (c\nabla u, c\nabla v) &= (f, v) \quad \forall v \in H_0^1(\Omega),\ t \in (0.T), \\ u_{|_{t=0}} &= u_0 \quad \text{in } \Omega, \\ u_t\big|_{t=0} &= v_0 \quad \text{in } \Omega \end{aligned} \tag{2}$$

Semi-Discrete Galerkin FE Formulation

The discretization in space leads to a system of ODE's

$$\begin{aligned} \mathbf{M}\frac{d^2\mathbf{U}}{dt^2}(t) + \mathbf{M}_\sigma \frac{d\mathbf{U}}{dt}(t) + \mathbf{K}\mathbf{U}(t) = \mathbf{R}(t), \quad t \in (0, T), \\ \mathbf{M}\mathbf{U}(0) = u_0^h, \quad \mathbf{M}\frac{d\mathbf{U}}{dt}(0) = v_0^h, \end{aligned} \tag{3}$$

where u_0^h, v_0^h are suitable approximations to the initial conditions. The stiffness matrix **K** and the mass matrix **M** are sparse. Moreover, the mass matrix **M** is SPD and (block-)diagonal.

Advantages of RK method

- One-step method, no starting procedure
- Time adaptivity straightforward
- Larger stability regions (but more work per step)

To apply a RK method to (2), we first need to rewrite it as a first-order system.
Multiply (3) with $\mathbf{M}^{\frac{1}{2}}$ and $\mathbf{M}^{-\frac{1}{2}}$ to get:

$$\frac{d^2\mathbf{z}}{dt^2}(t) + \mathbf{D}\frac{d\mathbf{z}}{dt}(t) + \mathbf{A}\mathbf{z}(t) = \tilde{\mathbf{R}}(t) \tag{4}$$

where $\mathbf{z}(t) = \mathbf{M}^{\frac{1}{2}}\mathbf{U}(t)$, $\mathbf{D} = \mathbf{M}^{-\frac{1}{2}}\mathbf{M}_\sigma\mathbf{M}^{\frac{1}{2}}$, $\quad \mathbf{A} = \mathbf{M}^{-\frac{1}{2}}\mathbf{K}\mathbf{M}^{-\frac{1}{2}}$, $\tilde{\mathbf{R}}(t) = \mathbf{M}^{-\frac{1}{2}}\mathbf{R}(t)$ and

The matrix $\mathbf{A}$ is sparse, symmetric, and positive semidefinite. We now rewrite (4) as a first-order system

$$\begin{aligned} \frac{d\mathbf{y}}{dt}(t) &= \mathbf{B}\mathbf{y}(t) + \mathbf{F}(t), \quad t \in (0, T), \\ \mathbf{y}(0) &= \mathbf{y}_0, \end{aligned} \tag{5}$$

where we have introduced

$$\mathbf{y}(t) = \left(\mathbf{z}(t), \frac{d\mathbf{z}}{dt}(t)\right)^T, \quad \mathbf{B} = \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{A} & -\mathbf{D} \end{pmatrix}, \quad \mathbf{F}(t) = \begin{pmatrix} \mathbf{0} \\ \tilde{\mathbf{R}}(t) \end{pmatrix}.$$

RK-methods and numerical integration

$$\boxed{\mathbf{y}'(t) = \mathbf{B}\mathbf{y}(t) + \mathbf{F}(t)}$$

$$\begin{aligned}
\mathbf{k}_1 &= \mathbf{B}\mathbf{y}_n + \mathbf{F}(t_n),\\
\mathbf{k}_2 &= \mathbf{B}\left(\mathbf{y}_n + \Delta t\frac{\mathbf{k}_1}{2}\right) + \mathbf{F}\left(t_n + \frac{\Delta t}{2}\right),\\
\mathbf{k}_3 &= \mathbf{B}\left(\mathbf{y}_n + \Delta t\frac{\mathbf{k}_2}{2}\right) + \mathbf{F}\left(t_n + \frac{\Delta t}{2}\right),\\
\mathbf{k}_4 &= \mathbf{B}\left(\mathbf{y}_n + \Delta t\,\mathbf{k}_3\right) + \mathbf{F}\left(t_n + \Delta t\right),\\
\mathbf{y}_{n+1} &= \mathbf{y}_n + \frac{\Delta t}{6}\left(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right).
\end{aligned}$$

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	0	0	1	
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$

Figure: Coefficients of the classical RK4 (order 4)

Derivation

Let us now split $\mathbf{y}$ and $\mathbf{F}$ in two parts

$$\mathbf{y} = (\mathbf{I} - \mathbf{P})\mathbf{y} + \mathbf{P}\mathbf{y}, \ \mathbf{P}^2 = \mathbf{P}$$

$$\mathbf{F} = (\mathbf{I} - \mathbf{P})\mathbf{F} + \mathbf{P}\mathbf{F}, \ \mathbf{P}^2 = \mathbf{P}$$

Then, we have

$$\frac{d}{dt}\mathbf{y} = \mathbf{B}\mathbf{y} + \mathbf{F} = \mathbf{B}(\mathbf{I} - \mathbf{P})\mathbf{y} + \mathbf{B}\mathbf{P}\mathbf{y} + (\mathbf{I} - \mathbf{P})\mathbf{F} + \mathbf{P}\mathbf{F}$$

or

$$\mathbf{y}(t_n + \xi\Delta t) = \mathbf{y}(t_n) \ + \int_{t_n}^{t_n+\xi\Delta t} \mathbf{B}\mathbf{y}^{[c]}(t) + \mathbf{F}^{[c]}(t)\, dt$$
$$+ \int_{t_n}^{t_n+\xi\Delta t} \mathbf{B}\mathbf{y}^{[f]}(t) + \mathbf{F}^{[f]}(t)\, dt, \quad 0 \le \xi \le 1$$

Coarse part

$$\int_{t_n}^{t_n+\xi\Delta t} \mathbf{B}\mathbf{y}^{[c]}(t) + \mathbf{F}^{[c]}(t)\, dt$$

$$\begin{aligned}\int_{t_n}^{t_n+\xi\Delta t} \mathbf{B}\mathbf{y}^{[c]}(t) &\simeq \xi\Delta t\mathbf{B}\left(\sum_{i=1}^{4} b_i \mathbf{y}^{[c]}(t)(t_n + c_i\xi\Delta t)\right)\\ &\approx \xi\Delta t\mathbf{B}(\mathbf{I}-\mathbf{P})\left(\sum_{i=1}^{4} b_i \sum_{j=0}^{3} \frac{c_i^j(\xi\Delta t)^j}{j!}\mathbf{y}^{(j)}(t)(t_n)\right)\end{aligned}$$

Repeated use of (5) to evaluate the derivatives $\mathbf{y}^j$ of $\mathbf{y}$ above then leads to

$$\int_{t_n}^{t_n+\xi\Delta t} \mathbf{B}(\mathbf{I}-\mathbf{P})\mathbf{y}(t)$$

$$\simeq \xi\Delta t \mathbf{B}(\mathbf{I}-\mathbf{P}) \left(\sum_{i=1}^{4} b_i \sum_{j=0}^{3} \frac{c_i^j(\xi\Delta t)^j}{j!} \left(\mathbf{B}^j \mathbf{y}_n + \sum_{l=1}^{j} \mathbf{B}^{j-l} \mathbf{F}^{(l-1)}(t_n) \right) \right).$$

To avoid the derivatives of $\mathbf{F}(t)$, we also interpolate $\mathbf{F}(t)$ by a quadratic polynomial trough the points $(t_n, \mathbf{F}_n), (t_{n+\frac{1}{2}}, \mathbf{F}_{n+\frac{1}{2}})$, and $(t_{n+1}, \mathbf{F}_{n+1})$. Since the nodes c_i,i.e. $c_1 = 0, c_2 = c_3 = \frac{1}{2}$ and $c_4 = 1$ for RK4, the degree of $\mathbf{q}$ may be strictly less then $4 - 1 = 3$.

$$\begin{aligned}\mathbf{q}(t_n + \tau) = \mathbf{F}_n &+ \frac{\tau}{\Delta t}\left(-3\mathbf{F}_n + 4\mathbf{F}_{n+\frac{1}{2}} - \mathbf{F}_{n+1}\right) \\ &+ \frac{\tau^2}{2\Delta t^2}\left(4\mathbf{F}_n - 8\mathbf{F}_{n+\frac{1}{2}} + 4\mathbf{F}_{n+1}\right)\end{aligned} \tag{6}$$

Now, we replace the derivatives of $\mathbf{F}$ above by the corresponding derivatives of $\mathbf{q}$ to obtain

$$\mathbf{y}(t_n + \xi\Delta t)$$
$$\simeq \mathbf{y}_n + \mathbf{B}(\mathbf{I} - \mathbf{P}) \left(\sum_{i=1}^{4} b_i \sum_{j=0}^{3} \frac{c_i^j (\xi\Delta t)^j}{j!} \left(\mathbf{B}^j \mathbf{y}_n + \sum_{l=1}^{j} \mathbf{B}^{j-l} \mathbf{q}^{(l-1)}(t_n) \right) \right)$$
$$+ (\mathbf{I} - \mathbf{P})(\hat{\mathbf{q}}(t_n + \xi\Delta t) - \hat{\mathbf{q}}(t_n)) + \int_{t_n}^{t_n+\xi\Delta t} (\mathbf{B}\mathbf{P}\mathbf{y}(t) + \mathbf{P}\mathbf{F}(t))\, dt,$$

where $\hat{\mathbf{q}}'(t_n) = \mathbf{q}(t)$. Because, $\mathbf{F}$ is known, so are $\mathbf{q}$ and $\hat{\mathbf{q}}$, and thus all terms needed involving $\mathbf{F}^{[c]}(t)$, for advancing the solution are explicitly known.

Fine part

$$\int_{t_n}^{t_n+\xi\Delta t} (\mathbf{BPy}(t) + \mathbf{PF}(t))\, dt \simeq \int_0^{\xi\Delta t} (\mathbf{BP}\hat{\mathbf{y}}(\tau) + \mathbf{PF}(t_n + \tau))\, dt$$

where $\hat{\mathbf{y}}(\tau)$ solves the following differential equation for $0 < \tau \leq \Delta t$:

$$\begin{aligned}
\frac{d\tilde{y}}{d\tau}(\tau) = {} & \mathbf{B}(\mathbf{I} - \mathbf{P}) \Big[\mathbf{y}_n + \tau(\mathbf{B}\mathbf{y}_n + \mathbf{F}_n) \\
& + \frac{\tau^2}{2}\left(\mathbf{B}^2\mathbf{y}_n + \mathbf{BF}_n + \frac{-3\mathbf{F}_n + 4\mathbf{F}_{n+\frac{1}{2}} - \mathbf{F}_{n+1}}{\Delta t}\right) \\
& + \frac{\tau^3}{6}\left(\mathbf{B}^2\mathbf{y}_n + \mathbf{BF}_n + \mathbf{B}\frac{-3\mathbf{F}_n + 4\mathbf{F}_{n+\frac{1}{2}} - \mathbf{F}_{n+1}}{\Delta t}\right. \\
& \left.\left. + \frac{4\mathbf{F}_n - 8\mathbf{F}_{n+\frac{1}{2}} + 4\mathbf{F}_{n+1}}{\Delta t^2}\right)\right] \\
& + (\mathbf{I} - \mathbf{P})\left[\mathbf{F}_n + \tau\frac{-3\mathbf{F}_n + 4\mathbf{F}_{n+\frac{1}{2}} - \mathbf{F}_{n+1}}{\Delta t} + \frac{\tau^2}{2}\frac{4\mathbf{F}_n - 8\mathbf{F}_{n+\frac{1}{2}} + 4\mathbf{F}_{n+1}}{\Delta t^2}\right] \\
& + \mathbf{BP}\tilde{\mathbf{y}}(\tau) + \mathbf{PF}(t_n + \tau),
\end{aligned} \tag{7}$$

$$\tilde{\mathbf{y}}(0) = \mathbf{y}_n.$$

Hence, to adavance $\mathbf{y}$ from t_n to $t_n + \Delta t$, we shall solve (7) by using the RK4 scheme with a smaller time-step $\Delta \tau = \Delta t / p$. In summary, given $\mathbf{y}_n$, the LTS algorithm based on the classical explicit RK4 method for the solution of (5) computes $\mathbf{y}_{n+1} \simeq \mathbf{y}(t_n + \Delta t)$ as follows.

Algorithm: LTS-RK4(p)

1. Set $\tilde{\mathbf{y}}_0 := \mathbf{y}_n$.
2. Compute

$$\mathbf{w}_{n,0} := \mathbf{B}(\mathbf{I}-\mathbf{P})\mathbf{y}_n + (\mathbf{I}-\mathbf{P})\mathbf{F}_n,$$

$$\mathbf{w}_{n,1} := \mathbf{B}(\mathbf{I}-\mathbf{P})(\mathbf{B}\mathbf{y}_n + \mathbf{F}_n) + (\mathbf{I}-\mathbf{P})\frac{-3\mathbf{F}_n + 4\mathbf{F}_{n+\frac{1}{2}} - \mathbf{F}_{n+1}}{\Delta t},$$

$$\mathbf{w}_{n,2} := \mathbf{B}(\mathbf{I}-\mathbf{P})\left(\mathbf{B}^2\mathbf{y}_n + \mathbf{B}\mathbf{F}_n + \frac{-3\mathbf{F}_n + 4\mathbf{F}_{n+\frac{1}{2}} - \mathbf{F}_{n+1}}{\Delta t}\right) + (\mathbf{I}-\mathbf{P})\frac{4\mathbf{F}_n - 8\mathbf{F}_{n+\frac{1}{2}} + 4\mathbf{F}_{n+1}}{\Delta t^2},$$

$$\mathbf{w}_{n,3} := \mathbf{B}(\mathbf{I}-\mathbf{P})\left(\mathbf{B}^3\mathbf{y}_n + \mathbf{B}^2\mathbf{F}_n + \mathbf{B}\frac{-3\mathbf{F}_n + 4\mathbf{F}_{n+\frac{1}{2}} - \mathbf{F}_{n+1}}{\Delta t} + \frac{4\mathbf{F}_n - 8\mathbf{F}_{n+\frac{1}{2}} + 4\mathbf{F}_{n+1}}{\Delta t^2}\right).$$

Derivation

3. For m = 0,...p-1, compute

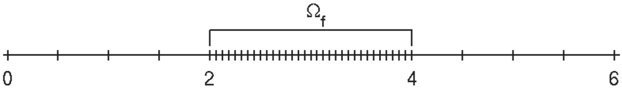
$$\begin{aligned}
\mathbf{k}_{1,\frac{m+1}{p}} &:= \mathbf{w}_{n,0} + m\Delta\tau\mathbf{w}_{n,1} + \frac{m^2}{2}\Delta\tau^2\mathbf{w}_{n,2} + \frac{m^3}{6}\Delta\tau^3\mathbf{w}_{n,3} \\
&+ \mathbf{BP}\tilde{\mathbf{y}}_{\frac{m}{p}} + \mathbf{PF}_{n,m}, \\
\mathbf{k}_{2,\frac{m+1}{p}} &:= \mathbf{w}_{n,0} + \left(m+\frac{1}{2}\right)\Delta\tau\mathbf{w}_{n,1} + \frac{1}{2}\left(m+\frac{1}{2}\right)^2\Delta\tau^2\mathbf{w}_{n,2} \\
&+ \frac{1}{6}\left(m+\frac{1}{2}\right)^3\Delta\tau^3\mathbf{w}_{n,3} + \mathbf{BP}\left(\tilde{\mathbf{y}}_{\frac{m}{p}} + \frac{\Delta\tau}{2}\mathbf{k}_{1,\frac{m+1}{p}}\right) + \mathbf{PF}_{n,m+\frac{1}{2}}, \\
\mathbf{k}_{3,\frac{m+1}{p}} &:= \mathbf{w}_{n,0} + \left(m+\frac{1}{2}\right)\Delta\tau\mathbf{w}_{n,1} + \frac{1}{2}\left(m+\frac{1}{2}\right)^2\Delta\tau^2\mathbf{w}_{n,2} \\
&+ \frac{1}{6}\left(m+\frac{1}{2}\right)^3\Delta\tau^3\mathbf{w}_{n,3} + \mathbf{BP}\left(\tilde{\mathbf{y}}_{\frac{m}{p}} + \frac{\Delta\tau}{2}\mathbf{k}_{2,\frac{m+1}{p}}\right) + \mathbf{PF}_{n,m+\frac{1}{2}}, \\
\mathbf{k}_{4,\frac{m+1}{p}} &:= \mathbf{w}_{n,0} + (m+1)\Delta\tau\mathbf{w}_{n,1} + \frac{1}{2}(m+1)^2\Delta\tau^2\mathbf{w}_{n,2} \\
&+ \frac{1}{6}(m+1)^3\Delta\tau^3\mathbf{w}_{n,3} + \mathbf{BP}\left(\tilde{\mathbf{y}}_{\frac{m}{p}} + \Delta\tau\mathbf{k}_{3,\frac{m+1}{p}}\right) + \mathbf{PF}_{n,m+1}, \\
\tilde{\mathbf{y}}_{\frac{m+1}{p}} &:= \tilde{\mathbf{y}}_{\frac{m}{p}} + \frac{1}{6}\Delta\tau\left(\mathbf{k}_{1,\frac{m+1}{p}} + 2\mathbf{k}_{2,\frac{m+1}{p}} + 2\mathbf{k}_{3,\frac{m+1}{p}} + \mathbf{k}_{4,\frac{m+1}{p}}\right)
\end{aligned}$$

4. Set $\mathbf{y}_{n+1} := \tilde{\mathbf{y}}_1$.

1D examples

We consider (1) with

- Source data
$$f(x,t) = \cos(\pi x)\left(\left(\pi^2 - 1\right)\sin(t) + \sigma\cos(t)\right)$$
- Exact solution: ($c \equiv 1$)
$$u(x,t) = \sin(t)\cos(\pi x).$$
- Computational domain:
$\Omega = [0,6],\ \Omega^{coarse} = [0,2] \cup [4,6],\ \Omega^{fine} = [2,4],\ p > 1$



- Damping coefficient: $\sigma \equiv 0.1$
- Homogeneous Neumann boundary condition

local refinement $p = 2, \Delta t = \Delta t_{RK4}$

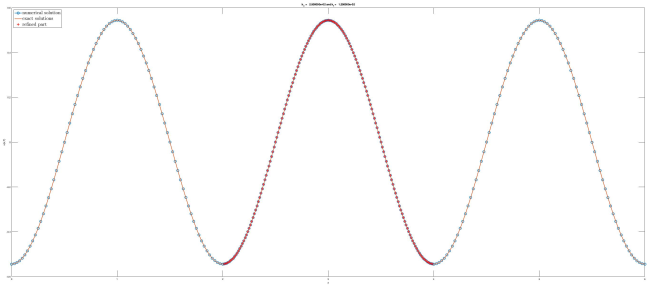
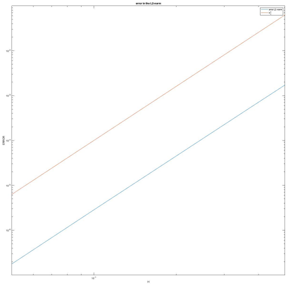


Figure: Numerical and exact solution for LTS-RK4 combined with $\mathcal{P}^3$ FE with $h^{coarse} = 0.025$ at time $T = 10$.



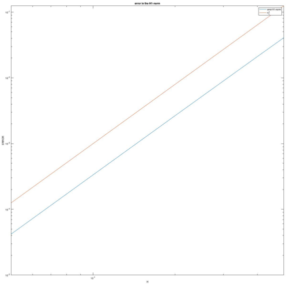


Figure: Errors of the LTS-RK4 schemes at time T = 10 as functions of $H = h^{coarse}$. In the left figure the L^2-norm and in the right figure the H^1-norm.

We now consider (1) with Neumann boundary, with $c = 1$ and $\sigma = 0$ on the interval $\Omega = [0, 1]$ with end time $T = 0.5$ and the initial conditions $u_0 = 0, v_0 = 0$. This yields the following two exact solutions

1. $u(x,t) = g(x) \cdot h\left(\frac{t-\tau}{t_0}\right)$, named quasistatic
2. $u(x,t) = h\left(\frac{t-\tau}{t_1}\right)$, named spatially constant

where

$$g(x) = \begin{cases} (x - \frac{1}{2})^2\, x^2, & \text{for } x < \frac{1}{2}, \\ (x - \frac{1}{2})^2\, (x-1)^2, & \text{for } x > \frac{1}{2}, \end{cases}$$

$$h(t) = \begin{cases} 0, & \text{for } t \leq 0, \\ \frac{1}{1 + \mathrm{e}^{\frac{1}{t} + \frac{1}{t-1}}}, & \text{for } t \in (0,1), \\ 1, & \text{for } t \geq 1, \end{cases} \tag{8}$$

with $\tau = 0.1,\ t_0 = 0.25$ and $t_1 = 0.8$.

Example 1:

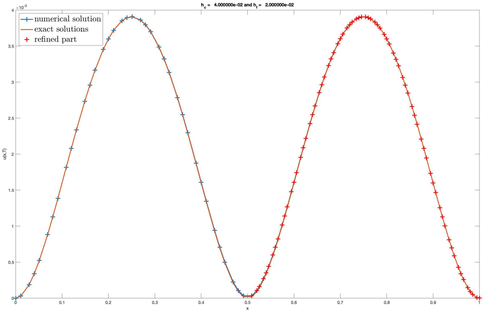


Figure: Numerical and exact solution of the quasistatic example for LTS-RK4 combined with $\mathcal{P}^3$ FE with $h^{coarse} = 0.04$, $\frac{h_c}{h_f} = 2$ at time $T = 0.5$ and $dt = 0.3 \cdot h_f$.

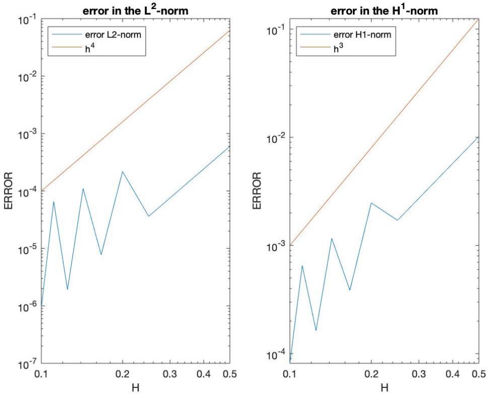


Figure: Errors of the LTS-RK4 schemes at time T $= 0.05$ as functions of $H = h^{coarse}$ for the quasistatic example. In the left figure the L^2-norm and in the right figure the H^1-norm.

Example 2:

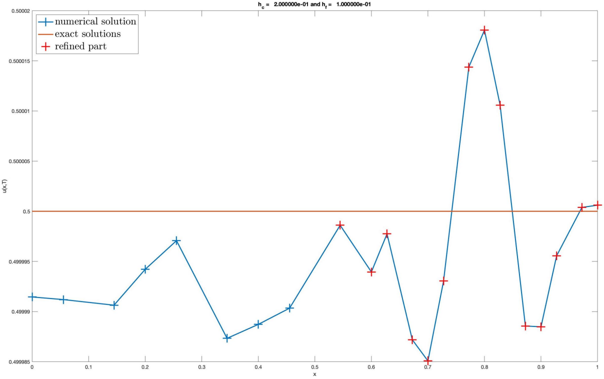


Figure: Numerical and exact solution of the spatially constant example for LTS- RK4 combined with $\mathcal{P}^3$ FE with $h^{coarse} = 0.2$, $\frac{h_c}{h_f} = 2$ at time $T = 0.5$ and $dt = 0.3 \cdot h_f$.

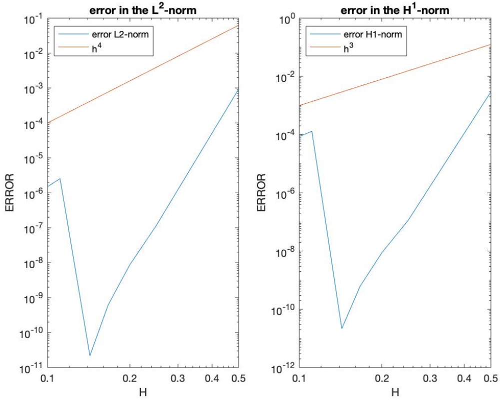


Figure: Errors of the LTS-RK4 schemes at time T = 0.05 as functions of $H = h^{coarse}$ for the spatially constant example. In the left figure the L^2-norm and in the right figure the H^1-norm.

Concluding remarks

- Wave equation with (or without) damping
- Mass-lumped $\Rightarrow$ block diagonal mass matrix $\Rightarrow$ explicit time integration
- support an arbitrary level and depth of refinement while maintaining the order of accuracy of the underlying Runge-Kutta method in the L^2 and H^1 norms for almost every example

THANK YOU FOR YOUR ATTENTION!