# Advanced Discretization Methods (WS 19/20)
# Homework 12

**(P. Knabner, L. Wester)**

Deadline for submission (theory): January 28th, 2019, 12:15
Deadline for submission (programming): February 4th, 2019, 12:15

**Remark:** When you apply theorems, whether they were found in Knabner/Angermann or another source, make sure to **cite them**!

## Exercise 22: Fully discrete error estimate (2+2+2+2+2+2+3)

Consider the heat equation in 1D:

$$\begin{aligned} \partial_t u(t,x) - \partial_{xx} u(t,x) &= f(t,x) \qquad \text{for } t \in (0,T), x \in (0,1) \\ u(0,x) &= u_0(x) \qquad \text{for } x \in (0,1) \\ u(t,0) = u(t,1) &= 0 \qquad \text{for } t \in (0,T) \end{aligned}$$

The variational problem takes the form: Find $u$, s.t. for every $v \in V := H_0^1((0,1))$:

$$\int_0^1 \partial_t u(t) v \; dx + \underbrace{\int_0^1 \partial_x u(t) \partial_x v \; dx}_{=:a(u(t),v)} = \int_0^1 f(t) v \; dx \qquad 0 < t < T$$

We discretize in space, choosing $V_h \subset V$, $\dim V_h < \infty$ as the space of linear Lagrange functions and discretize in time using the Implicit Euler scheme to arrive at the discrete formulation: Find $U^n \in V_h$, s.t.

$$\left(\frac{U^n - U^{n-1}}{\tau_{n-1}}, v_h\right)_0 + a(U^n, v_h) = (f^n, v_h)_0 \qquad \forall n = 1, ..., N, \; v_h \in V_h$$

for $f^n := f(t_n)$, $t_n := t_{n-1} + \tau_{n-1}$, $t_0 = 0$. We further assume the input data is smooth enough to yield a solution $u \in H^2(0,T,L^2((0,1))) \cap H^1(0,T,H^2((0,1)))$.

a) We consider the Ritz-projection $R_h : H_0^1((0,1)) \to V_h$ defined as

$$a(R_h u(t), v_h) = a(u(t), v_h) =: l(v_h) \qquad \forall v_h \in V_h$$

Use Céa's lemma and Aubin/Nitsche to show that it satisfies:

$$\|R_h u(t) - u(t)\|_0 \le Ch^2 |u(t)|_2$$

b) Conclude with a) that for $n \geq 1$:

$$\|u(t_n) - R_h u(t_n)\|_0 \le Ch^2 \left( \|\partial_{xx} u_0\|_0 + \int_0^{t_n} \|\partial_{t,xx} u(t)\|_0 \; dt \right)$$

c) Show that $\theta_n := R_h u(t_n) - U^n$ satisfies:

$$\left(\frac{\theta_n - \theta_{n-1}}{\tau_{n-1}}, v_h\right)_0 + a(\theta_n, v_h) = \left(\frac{R_h u(t_n) - R_h u(t_{n-1})}{\tau_{n-1}}, v_h\right)_0 - (\partial_t u(t_n), v_h)_0$$

d) Use the fully discrete estimate from Exercise 21 to show that

$$\|\theta_n\|_0 \le \|\theta_0\|_0 + \tau \sum_{i=1}^n \|\omega_i^1\|_0 + \tau \sum_{i=1}^n \|\omega_i^2\|_0$$

where $\tau = \max_{i=0,\ldots,N-1} \tau_i$ and

$$\omega_i^1 := \left( \partial_t u(t_i) - \frac{u(t_i) - u(t_{i-1})}{\tau_{i-1}} \right)$$

$$\omega_i^2 := \left( \frac{u(t_i) - R_h u(t_i)}{\tau_{i-1}} - \frac{u(t_{i-1}) - R_h u(t_{i-1})}{\tau_{i-1}} \right)$$

e) Use Taylor expansion to prove

$$\tau \sum_{i=1}^n \|\omega_i^1\|_0 \le \tau \int_0^{t_n} \|\partial_{tt} u(t)\|_0 \; dt$$

f) Further, use the fundamental theorem of calculus to show that

$$\tau \sum_{i=1}^n \|\omega_i^2\|_0 \le Ch^2 \int_0^{t_n} \|\partial_{t,xx} u(t)\|_0 \; dt$$

g) With the use of all previous exercises and the choice $U^0 := R_h u(0)$, conclude the error estimate for the fully discrete problem:

$$\|u(t_n) - U^n\|_0 \le Ch^2 \left( \|\partial_{xx} u_0\|_0 + \int_0^{t_n} \|\partial_{t,xx} u(t)\|_0 \; dt \right) + \tau \int_0^{t_n} \|\partial_{tt} u(t)\|_0 \; dt$$

## Programming exercise 11: Convection-dominated problems with SUPG (20)

We continue to work on the code back from Programming exercise 2 (in particular, this exercise only has to be solved for linear finite elements, you do not need to implement the slightly more complicated case of quadratic elements).
We consider a transport equation of the form

$$\begin{aligned} \partial_t u - \nabla \cdot (a(x)\nabla u) + \boldsymbol{c}(x)\cdot\nabla u + r(x)u = f(t,x) \qquad & \text{for } (t,x) \in (0,T)\times\Omega \\ u(t,x) = u_D(t,x) \qquad & \text{for } (t,x) \in (0,T)\times\partial\Omega \\ u(0,x) = u_0(x) \qquad & \text{for } x \in \Omega \end{aligned}$$

Like before, $\Omega = [0,1]^2$ and $a : \Omega \to [0,\infty]$, $r : \Omega \to \mathbb{R}$ are scalar-valued, space-dependent parameters. $\boldsymbol{c} : \Omega \to \mathbb{R}^2$ is vector-valued, space-dependent and represents convection.

a) Expand your function `AssembleMatrices` by a functionality that constructs `B`, the convective matrix:

$$B_{ij} = \int_\Omega \boldsymbol{c}\cdot\nabla\varphi_j\varphi_i \; dx$$

Test your implementation using the real solution $u(t,x) := \exp(-t)\sin(\pi x)\sin(\pi y)$ and various parameter choices.

b) Now implement a new function `AssembleSUPG` which assembles

$$(A_{\text{SUPG}})_{ij} := a_{\text{SUPG}}(\varphi_j,\varphi_i) := \sum_{T\in\mathcal{T}_h}\delta_T\int_T(\boldsymbol{c}\cdot\nabla\varphi_j + r\varphi_j)\boldsymbol{c}\cdot\nabla\varphi_i \; dx$$

$$(F_{\text{SUPG}})_i := f_{\text{SUPG}}(\varphi_i) := \sum_{T\in\mathcal{T}_h}\delta_T\int_T f\boldsymbol{c}\cdot\nabla\varphi_i \; dx$$

given an element-dependent constant $\delta_T$. Use again the trapezoidal rule on the edge midpoints to evaluate the integrals.

c) Test your implementation by choosing $u_0(x) := \sin(5\pi x_1)\sin(5\pi x_2)\chi_{[0.2,0.4]^2}(x)$, $u_D(t,x) = 0$, $f(t,x) = 0$, $a(x) = 0.0001$, $T = 1$ once without SUPG:

$$(\partial_t u_h, v_h) + \underbrace{(a\nabla u_h,\nabla v_h) + (\boldsymbol{c}\cdot\nabla u_h, v_h) + (ru_h, v_h)}_{=:a(u_h,v_h)} = 0 \qquad \forall v_h \in V_h$$

and once with SUPG:

$$(\partial_t u_h, v_h) + a(u_h,v_h) + a_{\text{SUPG}}(u_h,v_h) = f_{\text{SUPG}}(v_h) \qquad \forall v_h \in V_h$$

for two different setups:

i) $\boldsymbol{c}(x) := [0.5, 0.5]^T$ and $r(x) = 0$

ii) $\boldsymbol{c}(x) := \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} x_1 - 0.5 \\ x_2 - 0.5 \end{pmatrix}$ and $r(x) = 1$

using the Crank-Nicolson method to discretize in time and $\delta_T = \text{size}(T)$. How does the solution behave for different values for $\delta_T$? Comment on your results.

**Note:** The Programming exercise this time is due in 2 weeks, on February 4th!