# Scientific Computing 16/17, Homework #2, Sample Solution

Please return this assignment by Friday, Nov. 25. Please send a zip or tgz file by e-mail to juergen.fuhrmann@wias-berlin.de which contains the source code and a pdf describing your answer. Please prefix file names with your last names, e.g. `Müller-Nguyen-HA2.tgz`.

## 1. Problem description

Given:

- Domain $\Omega = (0,1)$
- outer normal $\mathbf{n}$
- Right hand side $f : \Omega \to \mathbb{R}$, $f = 1$
- "Conductivity" $\lambda = 1$
- Boundary value $v : \Gamma \to \mathbb{R}$, $v = 0$
- Transfer coefficient $\alpha = 1$

Search function $u : \Omega \to \mathbb{R}$ such that

$$
\begin{aligned}
-\nabla \cdot \lambda \nabla u &= f \quad \text{in } \Omega \\
+\lambda \nabla u \cdot \mathbf{n} + \alpha(u - v) &= 0 \quad \text{on } \Gamma
\end{aligned}
$$

minus sign would not yield coercive operator – mea culpa

## 2. Tasks

1. Calculate the exact solution of this problem
   - What is the limit of this solution for $\alpha \to \infty$?

**Answer outline:** - Calculation:

$$\text{Reformulation:} \quad \begin{cases} u'' & = -1,\ x \in (0,1) \\ -u' + \alpha u & = 0,\ x = 0 \\ +u' + \alpha u & = 0,\ x = 1 \quad \text{Careful with the different normal directions!} \\ & \qquad\qquad \text{In any case they need to be different.} \end{cases}$$

$$\text{from } u'' = -1: \quad u = -\frac{1}{2}x^2 + cx + d$$

$$\text{from bc at } x = 0: \quad c = \alpha d$$

$$\text{from bc at } x = 1: \quad d = \frac{1}{2\alpha}$$

$$\text{Finally:} \quad u_\alpha = -\frac{1}{2}x^2 + \frac{1}{2}x + \frac{1}{2\alpha}$$

$$u_\alpha \xrightarrow{\alpha \to \infty} u_\infty = -\frac{1}{2}x^2 + \frac{1}{2}x$$

- Interpretation of limit in the sense of the Dirichlet penalty method: for $\alpha \to \infty$, the solution $u_\alpha$ converges to the solution of the Dirichlet problem with $v = 0$:

$$\begin{cases} u'' & = -1,\ x \in (0,1) \\ \alpha u & = v,\ x = 0 \\ \alpha u & = v,\ x = 1 \end{cases}$$

2. Implement the finite volume discretization as a linear tridiagonal system on an equidistributed mesh with $N = 2^k + 1$ points with $k = 8 \dots 16$ Use the `numcxx` library or another equivalent tool for this purpose.
   - The library is now avilable via the course homepage.
   - Hint: have a look at the slides of lecture 06.
3. Use different solution strategies to solve the resulting linear system of equations:
   a) TDMA (Progonka)
   b) Dense matrix direct solver (e.g. LAPACK via numcxx)
   c) Sparse matrix direct solver (e.g. UMFPACK via numcxx)
   d) Iterative solver (e.g. Jacobi via numcxx)
   - Check the results against the exact solution. What happens if $N$ is increased ?
     - **Answer outline:** Unintendedly, the scheme shows some untypical but interesting behaviour in our case: it is **exact** for the given problem: for any given discretization point $x_i$, the discretized problem yields the exact solution of the continuous problem. We get this effect because of the constant right hand side, the purely second order equation, and the constant discretization step: For any $x \in \Omega, h > 0, \alpha$:

$$u(x) - u(x+h) = hx + \frac{h^2}{2} - \frac{h}{2}$$
$$u(x) - u(x-h) = -hx + \frac{h^2}{2} + \frac{h}{2}$$
$$\frac{-u(x-h) + 2u(x) - u(x+h)}{h} = h = hf(x)$$

As a consequence, the error between exact and discrete solution is in the order of magnitude of the roundoff error. For non-constant $f$ or non-constant $h$, we should see that the mean square error should decrease proportionally to $h^2$.

- Provide timings. Which method is the fastest ?
  - Hint: use e.g. `numcxx::cpu_clock()`
  - **Answer outline:** What we can expect is that of the exact methods, TDMA should be the fastest due to minimal overhead. LAPACK should quickly become the slowest, and for the larger problem sizes given it would even fail. As for the Jacobi method, its convergence is quite slow, so achieving the same accuracy as for the direct methods takes very long.
- What happens for values of the transfer coefficient $\alpha = 1, 10, 100, 1.0 \cdot 10^5, 1.0 \cdot 10^{10}, 1.0 \cdot 10^{20}$?
  - **Answer outline:** As one interpretation is that we implemented here the penalty method, for large $\alpha$ we should see a difference between $u_\alpha$ and $u_\infty$ in the region of round-off error, meaning that in effect, we solved the homogeneous Dirichlet problem.