Numerical solution of surface PDEs with Radial Basis Functions

Andriy Sokolov, Oleg Davydov and Stefan Turek

Institut für Angewandte Mathematik (LS3)
TU Dortmund
andriy.sokolov@math.tu-dortmund.de

Localized Kernel-Based Meshless Methods for Partial Differential Equations

ICERM, RI, USA

August 7 - 11, 2017

Outline

1. **Motivation**
2. **RBF-FD for surface-PDEs of reaction-diffusion-convection type**
3. **Outlook**

Motivation: chemotaxis on a membrane

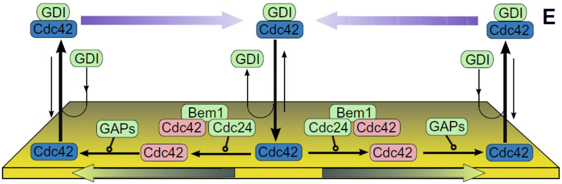


Figure: The membrane-cytoplastic shuttling of Cdc42 (inactive form, blue; active, pink). Taken from [1].

System of ODEs

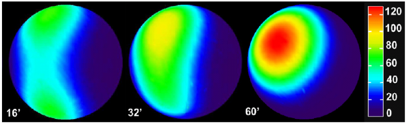


Figure: A 3D view on the surface of a yeast cell shows the distribution of the activated Cdc42. Taken from [1].

[1] *Dynamics of Cdc42 network embodies a Turing-type mechanism of yeast cell polarity* by A. B. Goryachev and A. V. Pokhilko, 2008.

Motivation: chemotaxis in a porous media

Christoph Landsberg, Florian Stenger, Andreas Deutsch, Michael Gelinsky, Angela Rösen-Wolff, Axel Voigt "Chemotaxis of mesenchymal stem cells within 3D biomimetic scaffolds – a modeling approach", Journal of Biomechanics **44**, pp. 359–364, 2011.

$$\begin{aligned} \partial_t u &= \nabla_\Gamma \cdot (D_u \nabla_\Gamma u - \chi u \nabla_\Gamma \nu) \text{ on } \Gamma, \\ \partial_t \nu &= \nabla \cdot (D_\nu \nabla \nu) + g \text{ in } \Omega_0, \end{aligned}$$

u is the density of the hematopoietic cells
ν is a stromal cell-derived factor 1α.

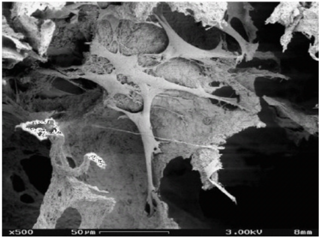


Figure: Scanning electron micrographs of longitudinal sections of a porous mineralized collagen scaffold, seeded with osteoblast-like cells.

Motivation: Γ-applications for chemotaxis models

Charles M. Elliott, Björn Stinner and Chandrasekhar Venkataraman
"Modelling cell motility and chemotaxis with evolving surface finite elements", J. R. Soc. Interface, published online, 2012.

parametric finite-elements method

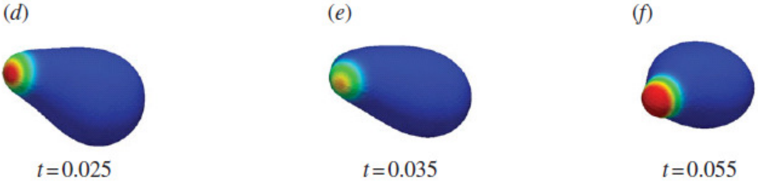
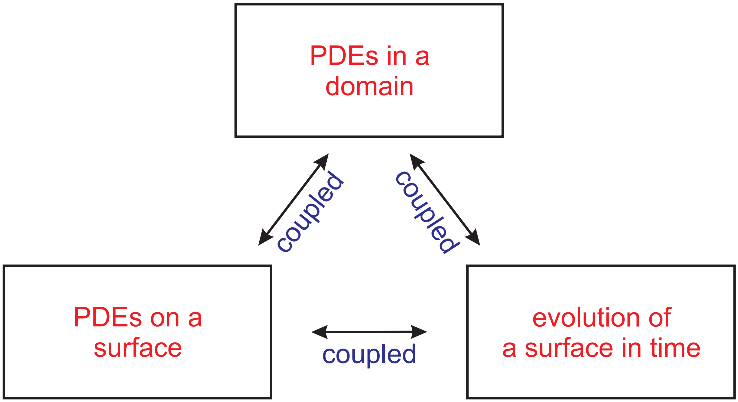


Figure: Migration of cells.

Framework



PDEs in a domain
coupled
coupled
PDEs on a surface
coupled
evolution of a surface in time

Framework

$$\frac{\partial^* \rho}{\partial t} + \nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho) \quad = \quad D\Delta_{\Gamma(t)}\rho + s(\cdot, \rho), \ \text{ on } \ \Gamma(t) \times T,$$

Framework

$$\frac{\partial^* \rho}{\partial t} + \nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho) \quad = \quad D\Delta_{\Gamma(t)}\rho + s(\cdot, \rho), \ \text{ on } \Gamma(t) \times T,$$

where

$$\partial_t^* \rho = \partial_t \rho + \boldsymbol{v} \cdot \nabla \rho + \rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}$$

and

$$\boldsymbol{v} = V\,\boldsymbol{n} + \boldsymbol{v}_S$$

is the velocity of the surface $\Gamma(t)$.

Framework

$$\frac{\partial^* \rho}{\partial t} + \nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho) \quad = \quad D\Delta_{\Gamma(t)}\rho + s(\cdot, \rho), \;\text{ on }\; \Gamma(t) \times T,$$

where

$$\partial_t^* \rho = \partial_t \rho + \boldsymbol{v} \cdot \nabla \rho + \rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}$$

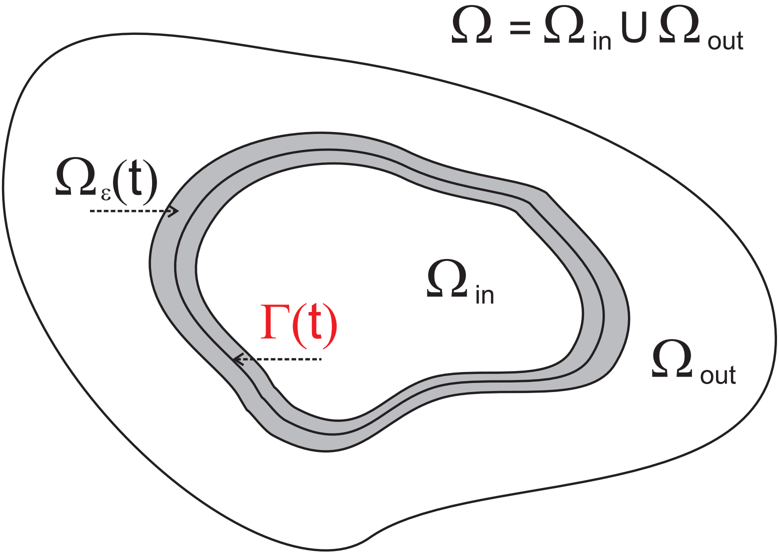
and

$$\boldsymbol{v} = V\,\boldsymbol{n} + \boldsymbol{v}_S$$

is the velocity of the surface $\Gamma(t)$.

$$\text{analytical prescription of } \Gamma = \Gamma(t).$$

Geometrical illustration



Surface PDE (evolving Γ)

$$\partial_t \rho + \boldsymbol{v} \cdot \nabla \rho + \rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v} + \nabla_{\Gamma(t)} \cdot (\boldsymbol{w} \rho) = D \Delta_{\Gamma(t)} \rho + s(\cdot, \rho)$$

Surface PDE (evolving Γ)

$$\partial_t \rho + \boldsymbol{v} \cdot \nabla \rho + \rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v} + \nabla_{\Gamma(t)} \cdot (\boldsymbol{w} \rho) = D \Delta_{\Gamma(t)} \rho + s(\cdot, \rho)$$

The level-set function:

$$\phi(\boldsymbol{x}) = \begin{cases} < 0 & \text{if } \boldsymbol{x} \text{ is inside } \Gamma \\ = 0 & \text{if } \boldsymbol{x} \in \Gamma \\ > 0 & \text{if } \boldsymbol{x} \text{ is outside } \Gamma \end{cases}$$

Surface PDE (evolving Γ)

$$\partial_t \rho + \boldsymbol{v} \cdot \nabla \rho + \rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v} + \nabla_{\Gamma(t)} \cdot (\boldsymbol{w} \rho) = D \Delta_{\Gamma(t)} \rho + s(\cdot, \rho)$$

The level-set function:

$$\phi(\boldsymbol{x}) = \begin{cases} < 0 & \text{if } \boldsymbol{x} \text{ is inside } \Gamma \\ = 0 & \text{if } \boldsymbol{x} \in \Gamma \\ > 0 & \text{if } \boldsymbol{x} \text{ is outside } \Gamma \end{cases}$$

Then

$$P_\Gamma = I - \frac{\nabla \phi}{\|\nabla \phi\|} \otimes \frac{\nabla \phi}{\|\nabla \phi\|} \quad \text{is a projection onto } \mathcal{T}_x \Gamma.$$

If ϕ is a signed distance, then $|\nabla \phi| = 1$.

Surface PDE (evolving Γ)

$$\partial_t \rho + \boldsymbol{v} \cdot \nabla \rho + \rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v} + \nabla_{\Gamma(t)} \cdot (\boldsymbol{w} \rho) = D \Delta_{\Gamma(t)} \rho + s(\cdot, \rho)$$

The level-set function:

$$\phi(\boldsymbol{x}) = \begin{cases} < 0 & \text{if } \boldsymbol{x} \text{ is inside } \Gamma \\ = 0 & \text{if } \boldsymbol{x} \in \Gamma \\ > 0 & \text{if } \boldsymbol{x} \text{ is outside } \Gamma \end{cases}$$

Then

$$P_\Gamma = I - \frac{\nabla \phi}{\|\nabla \phi\|} \otimes \frac{\nabla \phi}{\|\nabla \phi\|} \quad \text{is a projection onto } \mathcal{T}_x \Gamma.$$

If ϕ is a signed distance, then $|\nabla \phi| = 1$.

$$\Gamma_c(t) = \{\boldsymbol{x} : \phi(t, \boldsymbol{x}) = c\}.$$

RBF-FD for surface PDEs

Following Wright et al. 2012, 2014:

$$\nabla_{\Gamma(t)}\rho = \begin{pmatrix} (\mathbf{e}^x - n^x\mathbf{n})\cdot\nabla \\ (\mathbf{e}^y - n^y\mathbf{n})\cdot\nabla \\ (\mathbf{e}^z - n^z\mathbf{n})\cdot\nabla \end{pmatrix}\rho = \begin{pmatrix} \mathbf{p}^x\cdot\nabla \\ \mathbf{p}^y\cdot\nabla \\ \mathbf{p}^z\cdot\nabla \end{pmatrix}\rho = \begin{pmatrix} \mathcal{G}^x \\ \mathcal{G}^y \\ \mathcal{G}^z \end{pmatrix}\rho$$

$$(\mathcal{G}^x\, I_\varphi \rho(\mathbf{x}))\,|_{\mathbf{x}=\mathbf{x}_i} = \sum_{j=1}^{N} c_j\,(\mathcal{G}^x\,\varphi(\mathbf{r}_j(\mathbf{x})))\,|_{\mathbf{x}=\mathbf{x}_i}$$

$$= \sum_{j=1}^{N} c_j\,[\,((1 - n_i^x n_i^x)(x_i - x_j) - \\ - n_i^y n_i^y (y_i - y_j) \\ - n_i^z n_i^z (z_i - z_j))\,\frac{\varphi_{\mathbf{r}}^{'}(\mathbf{r}_j(\mathbf{x}_i))}{\mathbf{r}_j(\mathbf{x}_i)}\,]\,,$$

where φ is a radial basis function.

Operator assembly: Laplace-Beltrami

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\, \Delta_{\Gamma(t)} \rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot, \rho)$$

Operator assembly: Laplace-Beltrami

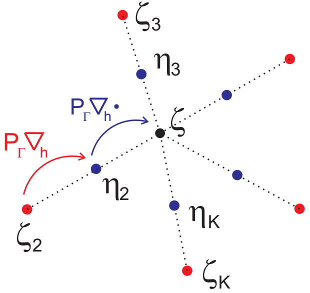
$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\, \Delta_{\Gamma(t)} \rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot, \rho)$$

$$L(t, \Gamma(t))\rho_h = (P_\Gamma \nabla_h \cdot)(P_\Gamma \nabla_h)\rho_h$$

Operator assembly: Laplace-Beltrami

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot, \rho)$$

$$L(t,\Gamma(t))\rho_h = (P_\Gamma \nabla_h \cdot)(P_\Gamma \nabla_h)\rho_h$$



where $\Xi = \{\varsigma, \varsigma_2, \varsigma_3, \ldots, \varsigma_K\}$ and $\Sigma = \{\varsigma, \eta_2, \eta_3, \ldots, \eta_K\}$.

Operator assembly: convection

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot,\rho)$$

$$V(t,\Gamma(t))\rho_h = (\boldsymbol{v} \cdot \nabla_h)\rho_h$$

Operator assembly: convection

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\, \Delta_{\Gamma(t)} \rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot, \rho)$$

$$V(t, \Gamma(t))\rho_h = (\boldsymbol{v} \cdot \nabla_h)\rho_h$$

$$(\boldsymbol{v} \cdot \nabla_h \rho_h)_i = \sum_{j \in \Xi} \sum_{p=1}^{d} v^p(\zeta_j)\, \omega_j^{\partial_p}\, \rho_h(\zeta_j).$$

Operator assembly: convection

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot, \rho)$$

$$V(t, \Gamma(t))\rho_h = (\boldsymbol{v} \cdot \nabla_h)\rho_h$$

$$(\boldsymbol{v} \cdot \nabla_h \rho_h)_i = \sum_{j \in \Xi} \sum_{p=1}^{d} v^p(\zeta_j)\, \omega_j^{\partial_p}\, \rho_h(\zeta_j).$$

The operator $K(t, \boldsymbol{w}, \Gamma(t)) \approx \boldsymbol{w} \cdot \nabla_{\Gamma(t)}\rho$ is assembled in a similar way.

Operator assembly: $\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}$

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot,\rho)$$

$$G(t,\Gamma(t))\rho_h = \rho_h(P_\Gamma \nabla_h \cdot \boldsymbol{v}_h)$$

Operator assembly: $\rho\nabla_{\Gamma(t)} \cdot \boldsymbol{v}$

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot,\rho)$$

$$G(t,\Gamma(t))\rho_h = \rho_h(P_\Gamma \nabla_h \cdot \boldsymbol{v}_h)$$

$$(\rho_h P_\Gamma \nabla_h \cdot \boldsymbol{v}_h)_i = (\rho_h)_i\,(P_\Gamma \nabla_h \cdot \boldsymbol{v}_h)_i\,.$$

Operator assembly: $\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}$

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot, \rho)$$

$$G(t, \Gamma(t))\rho_h = \rho_h (P_\Gamma \nabla_h \cdot \boldsymbol{v}_h)$$

$$(\rho_h P_\Gamma \nabla_h \cdot \boldsymbol{v}_h)_i = (\rho_h)_i \, (P_\Gamma \nabla_h \cdot \boldsymbol{v}_h)_i \, .$$

This discrete operator leads to a diagonal matrix.

Surface PDE (evolving Γ): scheme

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot,\rho)$$

Surface PDE (evolving Γ): scheme

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot,\rho)$$

Given ρ_h^n and $\Delta t = t_{n+1} - t_n$, solve for ρ_h^{n+1}

$$\begin{aligned} \frac{\rho_h^{n+1} - \rho_h^n}{\Delta t} \quad &+ \quad \theta \left(V^{n+1} + G^{n+1} + K^{n+1} - L^{n+1} \right) \rho_h^{n+1} \\ &= \quad -(1-\theta)\left(V^n + G^n + K^n - L^n \right) \rho_h^n \\ &+ \quad \theta\, s^{n+1} + (1-\theta)\, s^n. \end{aligned}$$

Surface PDE (evolving Γ): scheme

$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx L(t,\Gamma(t))\rho_h} + s(\cdot,\rho)$$

Given ρ_h^n and $\Delta t = t_{n+1} - t_n$, solve for ρ_h^{n+1}

$$\begin{aligned} \frac{\rho_h^{n+1} - \rho_h^n}{\Delta t} \quad &+ \quad \theta \left(V^{n+1} + G^{n+1} + K^{n+1} - L^{n+1}\right) \rho_h^{n+1} \\ &= \quad -(1-\theta)\left(V^n + G^n + K^n - L^n\right) \rho_h^n \\ &+ \quad \theta\, s^{n+1} + (1-\theta)\, s^n . \end{aligned}$$

$$\begin{aligned} \theta &= 1 \;-\; \text{Implicit} - \text{Euler} \\ \theta &= \frac{1}{2} \;-\; \text{Crank} - \text{Nicolson} \end{aligned}$$

Numerical tests: example 1

Solve

$$\frac{\partial^* \rho(\boldsymbol{x},t)}{\partial t} = D\Delta_{\Gamma(t)}\rho(\boldsymbol{x},t) + f(\boldsymbol{x},t) \qquad \text{on} \quad \Gamma(t),$$

where $\Gamma(t)$ is the zero level-set of

$$\phi(\boldsymbol{x},t) = |\boldsymbol{x}| - 1.0 + \sin(4\,t)(|\boldsymbol{x}| - 0.5)(1.5 - |\boldsymbol{x}|).$$

Analytical solution is

$$\rho(\boldsymbol{x},t) = e^{-t/|\boldsymbol{x}|^2}\,\frac{x_1}{|\boldsymbol{x}|}.$$

Numerical tests: example 1

Solve

$$\frac{\partial^* \rho(\boldsymbol{x},t)}{\partial t} = D\Delta_{\Gamma(t)}\rho(\boldsymbol{x},t) + f(\boldsymbol{x},t) \qquad \text{on} \quad \Gamma(t),$$

where $\Gamma(t)$ is the zero level-set of

$$\phi(\boldsymbol{x},t) = |\boldsymbol{x}| - 1.0 + \sin(4\,t)(|\boldsymbol{x}| - 0.5)(1.5 - |\boldsymbol{x}|).$$

Analytical solution is

$$\rho(\boldsymbol{x},t) = e^{-t/|\boldsymbol{x}|^2}\,\frac{x_1}{|\boldsymbol{x}|}.$$

$$\partial_t \rho + \underbrace{\boldsymbol{v}_S \cdot \nabla\rho + V\,\frac{\partial\rho}{\partial \boldsymbol{n}}}_{=\boldsymbol{v}\cdot\nabla\rho} \overbrace{-V\,H\rho + \rho\nabla_\Gamma \cdot \boldsymbol{v}_S}^{=\rho\nabla_{\Gamma(t)}\cdot\boldsymbol{v}} - D\,\Delta_\Gamma\rho = f$$

Numerical tests: example 1

Solve

$$\frac{\partial^* \rho(\boldsymbol{x},t)}{\partial t} = D\Delta_{\Gamma(t)}\rho(\boldsymbol{x},t) + f(\boldsymbol{x},t) \qquad \text{on} \quad \Gamma(t),$$

where $\Gamma(t)$ is a zero level-set of

$$\phi(\boldsymbol{x},t) = |\boldsymbol{x}| - 1.0 + \sin(4\,t)(|\boldsymbol{x}| - 0.5)(1.5 - |\boldsymbol{x}|).$$

Analytical solution is

$$\rho(\boldsymbol{x},t) = e^{-t/|\boldsymbol{x}|^2}\frac{x_1}{|\boldsymbol{x}|}.$$

Choose the time interval $[0, T = 0.1]$,
$\Delta t \approx h^2$ (for IE) and $\Delta t \approx h$ (for CN).

Numerical tests: example 1

Solve

$$\frac{\partial^* \rho(\boldsymbol{x},t)}{\partial t} = D\Delta_{\Gamma(t)}\rho(\boldsymbol{x},t) + f(\boldsymbol{x},t) \qquad \text{on} \quad \Gamma(t),$$

where $\Gamma(t)$ is a zero level-set of

$$\phi(\boldsymbol{x},t) = |\boldsymbol{x}| - 1.0 + \sin(4\,t)(|\boldsymbol{x}| - 0.5)(1.5 - |\boldsymbol{x}|).$$

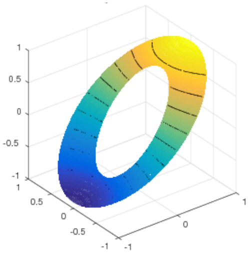
Analytical solution is

$$\rho(\boldsymbol{x},t) = e^{-t/|\boldsymbol{x}|^2}\,\frac{x_1}{|\boldsymbol{x}|}.$$

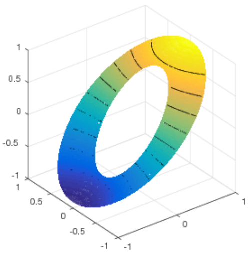
Choose the time interval $[0, T = 0.1]$,
$\Delta t \approx h^2$ (for IE) and $\Delta t \approx h$ (for CN).

$$l_2(\Omega)\text{-error} = \left(\frac{1}{N}\sum_{i=1}^{N}\left|u_{\text{analyt}}(\boldsymbol{x}_i,T) - u_{\text{num}}(\boldsymbol{x}_i,T)\right|^2\right)^{\frac{1}{2}}$$

Numerical tests: example 1



(a) analyt. solution at level lev 4



(b) num. solution at $T = 0.1$, level 4

Numerical tests: example 1

lev.	d.o.f	num. of time steps	$l_2(\Omega)$-error	order
Implicit-Euler scheme				
1	30	3	0.035854	–
2	100	10	0.009567	1.905
3	360	40	0.002602	1.878
4	1360	160	0.000748	1.798
5	5280	640	0.000213	1.812
Crank-Nicolson				
1	30	5	0.040218	–
2	100	10	0.09203	2.127
3	360	20	0.002367	1.959
4	1360	40	0.000673	1.814
5	5280	80	0.000192	1.809

Table: Convergence of the Implicit-Euler and Crank-Nicolson schemes.

Solve

$$\frac{\partial^* \rho(\boldsymbol{x},t)}{\partial t} = D\Delta_{\Gamma(t)}\rho(\boldsymbol{x},t) + f(\boldsymbol{x},t) \qquad \text{on} \quad \Gamma(t),$$

where $\Gamma(t)$ is a zero level-set of

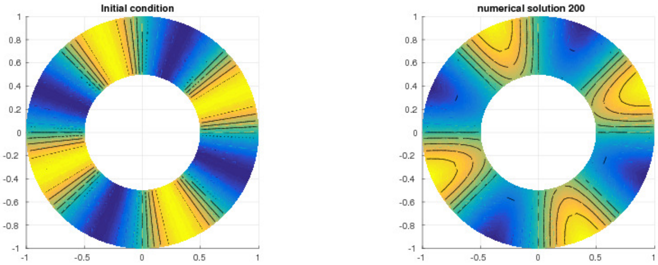
$$\phi(\boldsymbol{x},t) = |\boldsymbol{x}| - 1.0.$$

Initial condition is

$$\rho_0(\boldsymbol{x},t) = \sin(4\,\gamma), \quad \gamma \in [0, 2\pi].$$

$$atan2(x_2, x_1) = \begin{cases} \arctan(\frac{x_2}{x_1}) & ,\ x_1 > 0 \\ \arctan(\frac{x_2}{x_1}) + \pi & ,\ x_2 \geq 0,\ x_1 < 0 \\ \arctan(\frac{x_2}{x_1}) - \pi & ,\ x_2 < 0,\ x_1 < 0 \\ +\frac{\pi}{2} & ,\ x_2 > 0,\ x_1 = 0 \end{cases}$$

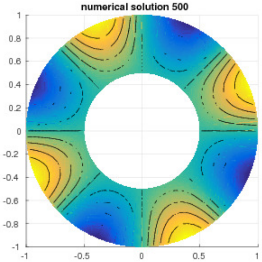
Numerical tests: example 2



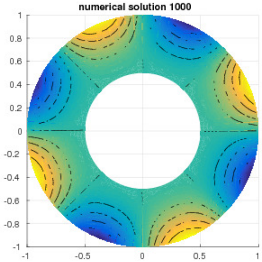
(c) at $t = 0.0$ (d) at $t = 0.002$

Figure: Solution at various time instances, $\Delta t = 0.0001$

Numerical tests: example 2



(a) at $t = 0.05$



(b) at $t = 0.1$

Figure: Solution at various time instances, $\Delta t = 0.0001$

Vanishing of ρ_0 occurs at a rate
which depends on the radius of the circle.

Solve

$$\frac{\partial^* \rho(\boldsymbol{x},t)}{\partial t} = D\Delta_{\Gamma(t)}\rho(\boldsymbol{x},t) + f(\boldsymbol{x},t) \qquad \text{on} \quad \Gamma(t),$$

where $\Gamma(t)$ is a zero level-set of

$$\phi(\boldsymbol{x},t) = |\boldsymbol{x}| - 1.0.$$

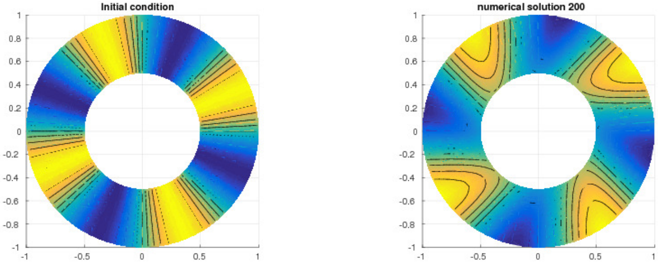
Initial condition is

$$\rho_0(\boldsymbol{x},t) = \sin(4\,\gamma), \quad \gamma \in [0, 2\pi],$$

and

$$\boldsymbol{v} := \boldsymbol{v}_S = \frac{(-\phi_{x_2}, \phi_{x_1})^T}{|\nabla\phi|}.$$

Numerical tests: example 3

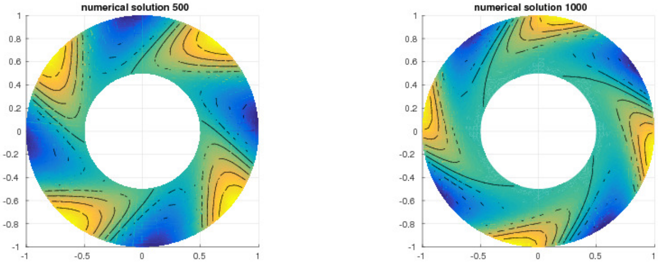


(a) at $t = 0.0$

(b) at $t = 0.002$

Figure: Solution at various time instances, $\Delta t = 0.0001$

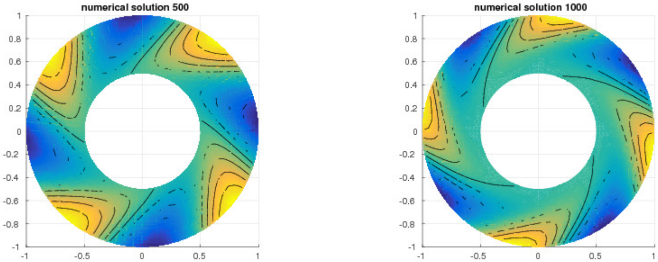
Numerical tests: example 3



(a) at $t = 0.05$ (b) at $t = 0.1$

Figure: Solution at various time instances, $\Delta t = 0.0001$

Numerical tests: example 3



(a) at $t = 0.05$ (b) at $t = 0.1$

Figure: Solution at various time instances, $\Delta t = 0.0001$

PDE on a surface which 'evolves' in the tangential direction.

Anisotropic diffusion:

$$-\nabla\left(\boldsymbol{A}\nabla u(\boldsymbol{x})\right) = f(\boldsymbol{x}) \qquad \text{in} \qquad \Omega = [0,1]^2,$$

where

$$\boldsymbol{A} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & \varepsilon \end{pmatrix} \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix}.$$

Anisotropic diffusion:

$$-\nabla\left(\boldsymbol{A}\nabla u(\boldsymbol{x})\right) = f(\boldsymbol{x}) \qquad \text{in} \qquad \Omega = [0,1]^2,$$

where

$$\boldsymbol{A} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & \varepsilon \end{pmatrix} \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix}.$$

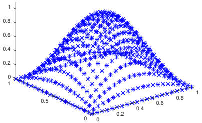


Figure: Numerical solution, h=1/20.

	E_{l_2}	EOC(l_2)	$E_{\max}$	EOC(max)
Stencil=5, $\varepsilon = 10^{-6}$, $\phi = \pi/6$				
h=1/5	58180	–	143297	–
h=1/10	71389	diverges	152334	diverges
h=1/20	76663	diverges	155059	diverges
Stencil=9, $\varepsilon = 10^{-6}$, $\phi = \pi/6$				
h=1/5	4895	–	10783	–
h=1/10	1882	1.379	4025	1.421
h=1/20	560	1.748	1150	1.807
Stencil=25, $\varepsilon = 10^{-6}$, $\phi = \pi/6$				
h=1/5	609	–	1529	–
h=1/10	47	3.695	85	4.168
h=1/20	3	3.969	12	2.824

Numerical tests: anisotropic diffusion

Anisotropic diffusion:

$$-\nabla\left(\boldsymbol{A}\nabla u(\boldsymbol{x})\right) = f(\boldsymbol{x}) \qquad \text{in} \qquad \Omega = [0,1]^2,$$

where

$$\boldsymbol{A} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & \varepsilon \end{pmatrix} \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix}.$$

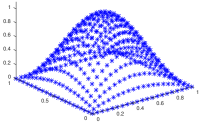


Figure: Numerical solution, h=1/20.

	E_{l_2}	EOC(l_2)	$E_{\max}$	EOC(max)
Stencil=9, $\varepsilon = 10^{-6}$, $\phi = \pi/6$				
h=1/5	4895	–	10783	–
h=1/10	1882	1.379	4025	1.421
h=1/20	560	1.748	1150	1.807
h↘	??	??	??	??
Stencil=25, $\varepsilon = 10^{-6}$, $\phi = \pi/6$				
h=1/5	609	–	1529	–
h=1/10	47	3.695	85	4.168
h=1/20	3	3.969	12	2.824
h↘	??	??	??	??

Numerical tests: anisotropic diffusion

Anisotropic diffusion:

$$-\nabla\left(\boldsymbol{A}\nabla u(\boldsymbol{x})\right) = f(\boldsymbol{x}) \qquad \text{in} \qquad \Omega = [0,1]^2,$$

where

$$\boldsymbol{A} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & \varepsilon \end{pmatrix} \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix}.$$

Special treatment of the RBF-FD stencil is required!!

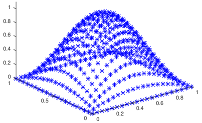


Figure: Numerical solution, h=1/20.

Numerical tests: anisotropic diffusion

Anisotropic diffusion:

$$-\nabla\left(\boldsymbol{A}\nabla u(\boldsymbol{x})\right) = f(\boldsymbol{x}) \qquad \text{in} \qquad \Omega = [0,1]^2,$$

where

$$\boldsymbol{A} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & \varepsilon \end{pmatrix} \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix}.$$

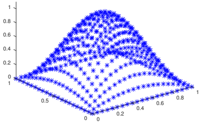


Figure: Numerical solution, h=1/20.

Special treatment of the RBF-FD stencil is required!!

Bengt Fornberg, Elisabeth Larsson, and Natasha Flyer, "Stable Computations with Gaussian Radial Basis Functions", *SIAM J. Sci. Comput.*, **33**(2), pp. 869-892, (2011).

Elisabeth Larsson, Erik Lehto, Alfa Heryudono, and Bengt Fornberg, "Stable Computation of Differentiation Matrices and Scattered Node Stencils Based on Gaussian Radial Basis Functions", *SIAM J. Sci. Comput.*, **35**(4), pp. A2096-A2119, (2013).

Natasha Flyer, Gregory A. Barnett, Louis J. Wicker, "Enhancing finite differences with radial basis functions: Experiments on the Navier-Stokes equations", *Journal of Computational Physics*, **316**, pp. 39-62, (2016).

The phase-field method:

$$\rho_t - \nabla \cdot (\nabla_\Gamma \rho(\boldsymbol{x})) = \rho(\boldsymbol{x}) + f \quad \text{on } \Gamma \subset \mathbb{R}^d,$$

The phase-field method:

$$B(\phi)\,\rho_t - \nabla\cdot\left(B(\phi)\nabla\rho(\boldsymbol{x})\right) = B(\phi)\,(\rho(\boldsymbol{x}) + f) \quad \text{in } \Omega_\varepsilon \subset \mathbb{R}^d,$$

where

$$\phi(\boldsymbol{x}) = \frac{1}{2}\left(1.0 - \tanh\left(\frac{2}{5h}(|\boldsymbol{x}| - 0.3)\right)\right),$$

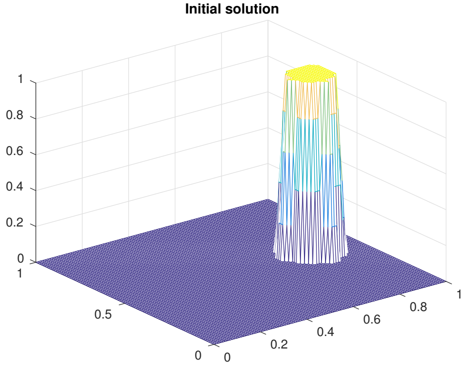
and

$$B(\phi) = 36\,\phi^2\,(1 - \phi^2).$$

$$\Gamma = \{\boldsymbol{x} : |\boldsymbol{x} - (0.5, 0.5)^T| = 0.3\}$$

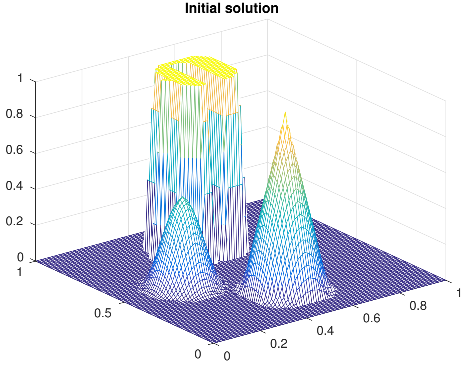
The phase-field method:

$$B(\phi)\,\rho_t - \nabla \cdot (B(\phi)\nabla\rho(\mathbf{x})) = B(\phi)\,(\rho(\mathbf{x}) + f) \quad \text{in } \Omega_\varepsilon \subset \mathbb{R}^d,$$



Transport equation (the solid body rotation):

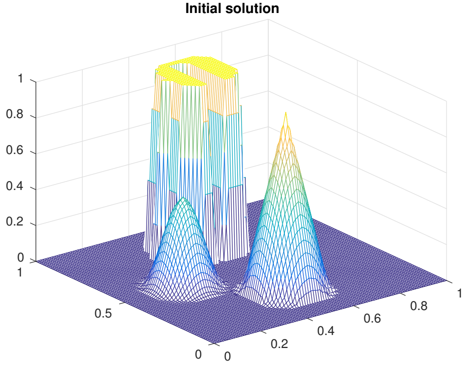
$$\rho_t + \mathbf{v} \cdot \nabla \rho = 0 \quad \text{in } \Omega = [0,1]^2.$$



Numerical tests: transport equation

Transport equation (the solid body rotation):

$$\rho_t + \mathbf{v} \cdot \nabla \rho - 0.0008\, \Delta\, \rho = 0 \quad \text{in } \Omega = [0,1]^2.$$



Numerical tests: transport equation

Transport equation (the solid body rotation):

$$\rho_t + \mathbf{v} \cdot \nabla \rho - 0.0008\, \Delta\, \rho = 0 \quad \text{in } \Omega = [0,1]^2.$$

Stabilization technique is required!!

Numerical tests: transport equation

Transport equation (the solid body rotation):

$$\rho_t + \mathbf{v} \cdot \nabla \rho \, - \, 0.0008 \, \Delta \, \rho = 0 \quad \text{in } \Omega = [0,1]^2.$$

Stabilization technique is required!!

Bengt Fornberg and Erik Lehto, "Stabilization of RBF-generated finite difference methods for convective PDEs", *Journal of Computational Physics*, **230**, pp. 2270–2285, (2011).

Numerical tests: transport equation

Transport equation (the solid body rotation):

$$\rho_t + \mathbf{v} \cdot \nabla \rho \; - \; 0.0008 \, \Delta \, \rho = 0 \quad \text{in } \; \Omega = [0,1]^2.$$

Stabilization technique is required!!

Bengt Fornberg and Erik Lehto, "Stabilization of RBF-generated finite difference methods for convective PDEs", *Journal of Computational Physics*, **230**, pp. 2270–2285, (2011).

... or ???

Numerical tests: evolution along a curve

Solve

$$\frac{\partial^* \rho}{\partial t} + \alpha\rho = D\Delta_{\Gamma(t)}\rho \qquad \text{on} \quad \Gamma(t) \subset \Omega = \{\boldsymbol{x} \in \mathbb{R}^2 : 0.5 \leq |\boldsymbol{x}| \leq 1.5\}.$$

Numerical tests: evolution along a curve

Solve

$$\frac{\partial^* \rho}{\partial t} + \alpha\rho = D\Delta_{\Gamma(t)}\rho \qquad \text{on} \quad \Gamma(t) \subset \Omega = \{\boldsymbol{x} \in \mathbb{R}^2 : 0.5 \le |\boldsymbol{x}| \le 1.5\}.$$

The level set function:

$$\phi(\boldsymbol{x}, t) = |\boldsymbol{x}| - (1.0 + b\,t\,\sin(5\gamma)).$$

The initial condition:

$$\rho_0 = \begin{cases} 0.75 & \text{if } 0.65 \le |\boldsymbol{x}| \le 0.85, \\ 0.0 & \text{otherwise.} \end{cases}$$

Solve

$$\frac{\partial^* \rho}{\partial t} + \alpha\rho = D\Delta_{\Gamma(t)}\rho \qquad \text{on} \quad \Gamma(t) \subset \Omega = \{\boldsymbol{x} \in \mathbb{R}^2 : 0.5 \leq |\boldsymbol{x}| \leq 1.5\}.$$

The level set function:

$$\phi(\boldsymbol{x}, t) = |\boldsymbol{x}| - (1.0 + b\, t\, \sin(5\gamma)).$$

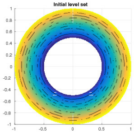
The initial condition:

$$\rho_0 = \begin{cases} 0.75 & \text{if } 0.65 \leq |\boldsymbol{x}| \leq 0.85, \\ 0.0 & \text{otherwise.} \end{cases}$$

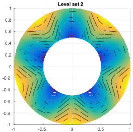
We set

$$\alpha = 0.2, \quad b = 10 \text{ and } \gamma = atan2(x_2, x_1).$$

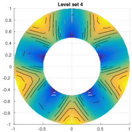
Numerical tests: evolution along a curve



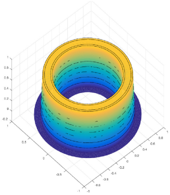
(a) level set $t = 0.0$



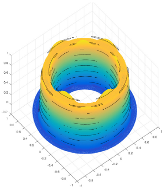
(b) level set, $t = 0.02$



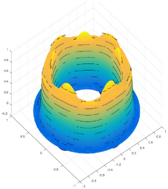
(c) level set, $t = 0.04$



(d) initial solution



(e) $t = 0.001$



(f) $t = 0.002$

Conclusions

1. **It is possible to treat PDEs of time-dependent surfaces which evolve both in normal and in tangential directions.**
2. **The method is accurate and robust.**
3. **The RBF-FD nature of the method allows sufficient flexibility while working with meshes.**

Conclusions

1. **A special treatment of RBF-FD generated stencils is required.**
2. **Stabilization of convection-like terms is necessary.**
3. **Coupling of surface PDEs with equation(s) which describe corresponding evolution of the surface is necessary.**
4. **The flexibility of our approach regarding 'meshes' should be further exploited.**
5. **The method should be implemented in an HPC-fashion code.**

Conclusions

1. **A special treatment of RBF-FD generated stencils is required.**
2. **Stabilization of convection-like terms is necessary.**
3. **Coupling of surface PDEs with equation(s) which describe corresponding evolution of the surface is necessary.**
4. **The flexibility of our approach regarding 'meshes' should be further exploited.**
5. **The method should be implemented in an HPC-fashion code.**

Conclusions

1. **A special treatment of RBF-FD generated stencils is required.**
2. **Stabilization of convection-like terms is necessary.**
3. **Coupling of surface PDEs with equation(s) which describe corresponding evolution of the surface is necessary.**
4. **The flexibility of our approach regarding 'meshes' should be further exploited.**
5. **The method should be implemented in an HPC-fashion code.**

Conclusions

1. **A special treatment of RBF-FD generated stencils is required.**
2. **Stabilization of convection-like terms is necessary.**
3. **Coupling of surface PDEs with equation(s) which describe corresponding evolution of the surface is necessary.**
4. **The flexibility of our approach regarding 'meshes' should be further exploited.**
5. **The method should be implemented in an HPC-fashion code.**

Conclusions

1. **A special treatment of RBF-FD generated stencils is required.**
2. **Stabilization of convection-like terms is necessary.**
3. **Coupling of surface PDEs with equation(s) which describe corresponding evolution of the surface is necessary.**
4. **The flexibility of our approach regarding 'meshes' should be further exploited.**
5. **The method should be implemented in an HPC-fashion code.**

- Prof. Dr. Oleg Davydov, University of Giessen
- Prof. Dr. Dmitri Kuzmin, TU Dortmund
- Prof. Dr. Stefan Turek, TU Dortmund

Thank you very much

for your attention!

Backup 1

Weights are combined in a form

$$\mathrm{trace}\,(A\,B) = \mathrm{sum}\left(\mathrm{sum}\left(A^T .* B\right)\right).$$