A flux-corrected RBF-FD method for convection dominated problems in domains and on manifolds

Andriy Sokolov
Dmitri Kuzmin, Oleg Davydov and Stefan Turek

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

Workshop on
Variational Multiscale Methods and
Stabilized Finite Elements (VMS)

December 5-7, 2018

technische universität dortmund

fakultät für mathematik

Outline

1. **Motivation**
2. **The (meshless) RBF-FD method for surface-PDEs of the reaction-diffusion-convection type**
3. **FCT stabilization for convection dominated problems**
4. **Outlook**

Motivation: chemotaxis on a membrane

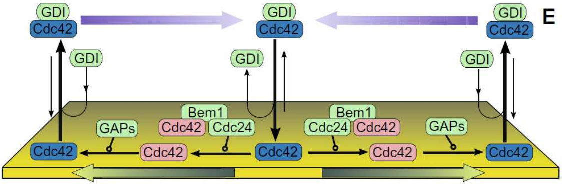


Figure: The membrane-cytoplastic shuttling of Cdc42 (inactive form, blue; active, pink). Taken from [1].

System of ODEs

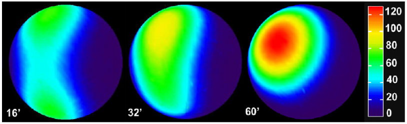


Figure: A 3D view on the surface of a yeast cell shows the distribution of the activated Cdc42. Taken from [1].

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

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

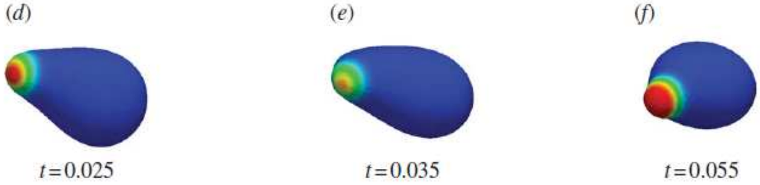
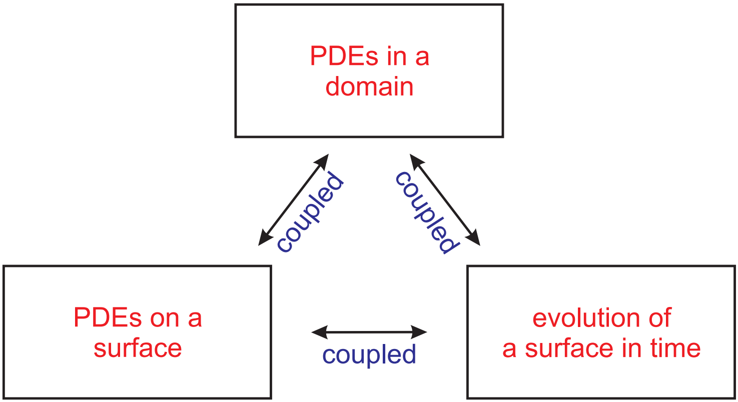


Figure: Migration of cells.

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

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

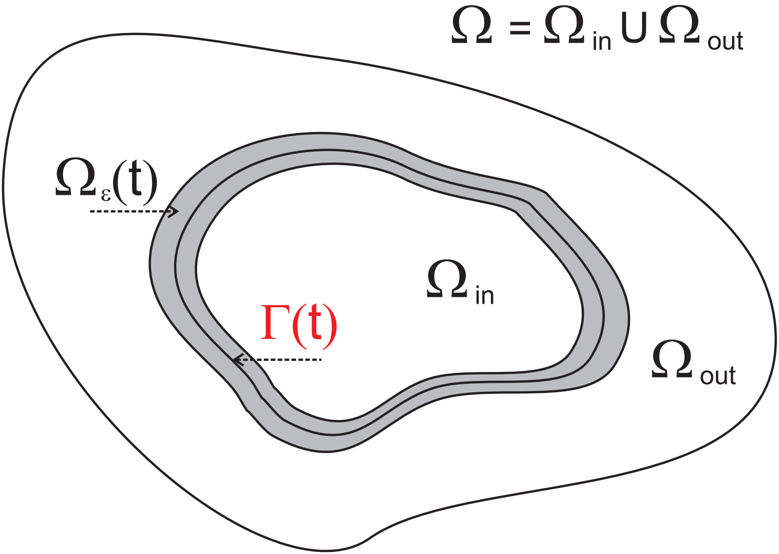
and

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

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 $|\nabla \phi| = 1$.

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

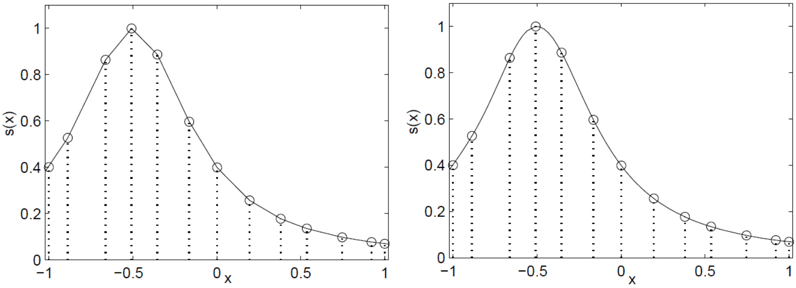
Radial Basis Functions (RBFs)

$$s(x) = \sum_{j=1}^{n} \lambda_j \varphi(\|x - x_j\|),$$

- dimension independent,
- sufficiently smooth,
- easy to find (multiple-) derivatives,
- the corresponding SLE is well-posed, i.e. $\det(\mathbf{A}) \neq 0$,
- meshfree,
- other advantages.

Linear vs Multiquadratic RBF:

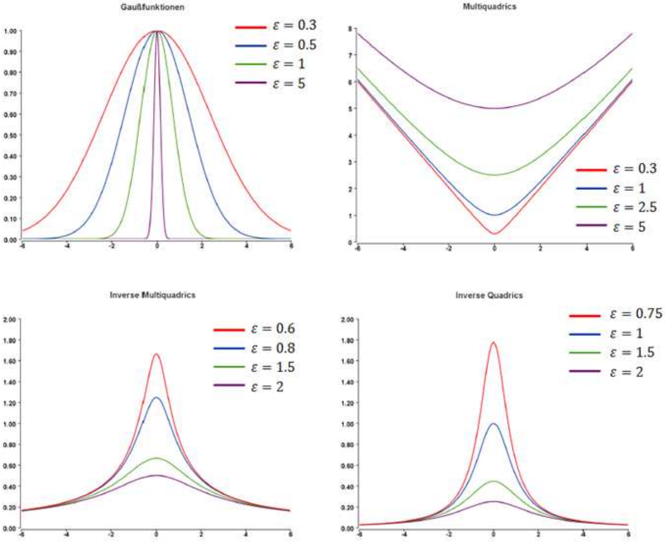
$$s(x) = \sum_{j=1}^{N} \lambda_j \|x - x_j\| \qquad s(x) = \sum_{j=1}^{N} \lambda_j \sqrt{c^2 + \|x - x_j\|^2}$$



Types of RBFs:

Infinitely smooth RBFs	$\varphi(\mathbf{r}) \ \ (\mathbf{r} \geq 0)$
Gaussian (GA)	$\mathbf{e}^{-(\varepsilon \mathbf{r})^2}$
Inverse quadratic (IQ)	$\dfrac{1}{1+(\varepsilon \mathbf{r})^2}$
Inverse multiquadratic (IMQ)	$\dfrac{1}{\sqrt{1+(\varepsilon \mathbf{r})^2}}$
Multiquadratic (MQ)	$\sqrt{1+(\varepsilon \mathbf{r})^2}$
Piecewise smooth RBFs	
Linear	$\mathbf{r}$
Cubic	$\mathbf{r}^3$
Thin plate spline (TPS)	$\mathbf{r}^2 \log \mathbf{r}$

Interpolation with RBFs



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 (Gaussian, polyharmonic).

Operator assembly: Laplace-Beltrami

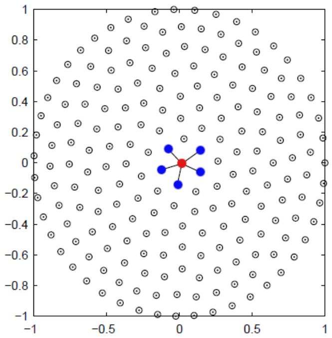
$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx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

(generalized) Finite Difference Radial Basis Function (FD-RBF)

Approximation of a (linear) differential operator:

$$\mathcal{L}u = f, \quad \Omega \subset \mathbb{R}^d$$

$$\mathcal{L}u|_{\zeta} \approx \sum_{\zeta_j \in \Xi_{\zeta}} \omega_{\zeta_j} u(\zeta_j), \quad \Xi_{\zeta} = \{\zeta = \zeta_0, \zeta_1, \zeta_2, \ldots, \zeta_K\}.$$



(generalized) Finite Difference Radial Basis Function (FD-RBF)

Approximation of a (linear) differential operator:

$$\mathcal{L}u = f, \quad \Omega \subset \mathbb{R}^d$$

$$\begin{aligned}
\mathcal{L}u|_{\zeta} &= \sum_{j=0}^{K} \omega_j u(\zeta_j), \\
\mathcal{L}\sum_{i=0}^{N} c_i \varphi(\|\zeta - \zeta_i\|) &= \sum_{j=0}^{K} \omega_j u(\zeta_j), \\
\sum_{i=0}^{N} c_i \mathcal{L}\varphi(\|\zeta - \zeta_i\|) &= \sum_{i=0}^{N} c_i \sum_{j=0}^{K} \omega_j \varphi(\|\zeta_j - \zeta_i\|) \\
\underbrace{\mathcal{L}\varphi(\|\zeta - \zeta_i\|)}_{rhs} &= \sum_{j=0}^{K} \underbrace{\omega_j}_{\omega} \underbrace{\varphi(\|\zeta_j - \zeta_i\|)}_{\boldsymbol{\Phi}} \\
\boldsymbol{\Phi}\,\omega &= rhs(\mathcal{L})
\end{aligned}$$

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

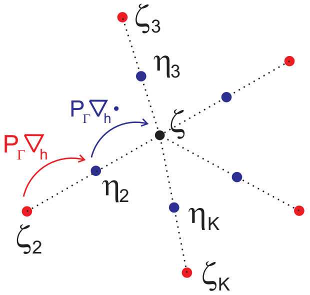
$$\partial_t \rho + \underbrace{\boldsymbol{v} \cdot \nabla \rho}_{\approx V(t,\Gamma(t))\rho_h} + \overbrace{\rho \nabla_{\Gamma(t)} \cdot \boldsymbol{v}}^{\approx G(t,\Gamma(t))\rho_h} + \underbrace{\nabla_{\Gamma(t)} \cdot (\boldsymbol{w}\rho)}_{\approx K(t,\boldsymbol{w},\Gamma(t))\rho_h} = \overbrace{D\,\Delta_{\Gamma(t)}\rho}^{\approx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

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

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

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

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

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

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

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

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

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

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}

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

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

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

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

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

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

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

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

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

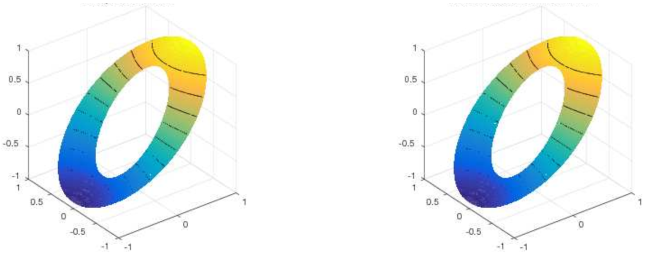
Analytical solution is

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

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

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.

Transport equation (the solid body rotation):

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

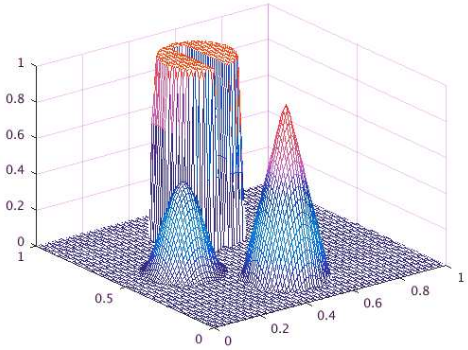


Figure: Initial condition.

Transport equation (the solid body rotation):

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

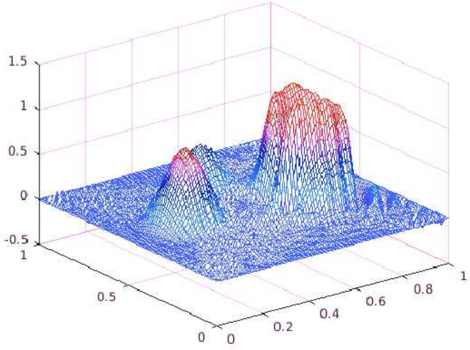


Figure: Non-stabilized solution, rotation angle $\alpha = \dfrac{2\pi}{3}$, $N = 100 \times 100$.

Transport equation (the solid body rotation):

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

Stabilization technique is required!!

Numerical tests: transport equation

Transport equation (the solid body rotation):

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

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).

Transport equation (the solid body rotation):

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

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

$$u_t + \mathbf{v} \cdot \nabla u - \alpha \Delta u = 0.$$

Numerical tests: transport equation

Transport equation (the solid body rotation):

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

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

$$u_t + \mathbf{v} \cdot \nabla u - \alpha \Delta u = 0.$$

... or???

FCT technique

Consider the general transport problem:

$$u_t \underbrace{-\nabla_{\mathcal{P}} \cdot (D \nabla_{\mathcal{P}} u)}_{\approx \boldsymbol{L}} + \overbrace{\boldsymbol{v} \cdot \nabla_{\mathcal{P}} u}^{\approx \boldsymbol{K}} = g, \qquad (1)$$

FCT technique

Consider the general transport problem:

$$u_t \underbrace{-\nabla_{\mathcal{P}} \cdot (D \nabla_{\mathcal{P}} u)}_{\approx \boldsymbol{L}} + \overbrace{\boldsymbol{v} \cdot \nabla_{\mathcal{P}} u}^{\approx \boldsymbol{K}} = g, \tag{1}$$

After the RBF-FD discretization one obtains the following semi-discrete problem:

$$\mathcal{M} \frac{d\,\boldsymbol{u}}{dt} = \mathcal{K} \boldsymbol{u} + g, \tag{2}$$

where $\mathcal{K} = -\mathcal{M}\left(\mathbf{L} + \mathbf{K}\right)$, and $\mathcal{M} = \mathrm{diag}(m_1, \ldots, m_N)$ is a diagonal mass matrix

FCT technique (implicit Euler)

$$[\mathcal{M} - \Delta t\, \mathcal{K}]\, u^{n+1} = \mathcal{M}\, u^n + \Delta t\, g^n.$$

1. Calculate the low-order (overdiffusive) solution u^{Low}:

$$[\mathcal{M} - \Delta t\, \mathcal{K}^*]\, u^{\mathrm{Low}} = \mathcal{M} u^n + \Delta t\, g^n,$$

where $\mathcal{K}^* = \mathcal{K} + \mathcal{D}$ is M-Matrix and $d_{ij} = \max\{-k_{ij}, 0, -k_{ji}\}, \;\; j \neq i.$

2. reconstruct the high-order solution u^{m+1} (i.e. to remove the artificial diffusion where it is possible):

$$\mathcal{M} u^{n+1} = \mathcal{M} u^{\mathrm{Low}} + \Delta t \bar{f}(u^{\mathrm{Low}}, u^n),$$

where $\bar{f}_i = \sum_{j \neq i} \alpha_{ij} f_{ij}$ with flux limiters $\alpha_{ij} \in [0, 1]$ and

$$f_{ij} = m_{ij}(\dot{u}_i^{\mathrm{Low}} - \dot{u}_j^{\mathrm{Low}}) + d_{ij}(u_i^{\mathrm{Low}} - u_j^{\mathrm{Low}}) \;\; - \text{antidiffusive fluxes.}$$

$$\dot{u}^{\mathrm{Low}} = \mathcal{M}^{-1}\, (\mathcal{K}^*)\, u^{\mathrm{Low}}$$

FCT technique (implicit Euler)

$$[\mathcal{M} - \Delta t\, \mathcal{K}]\, u^{n+1} = \mathcal{M}\, u^n + \Delta t\, g^n.$$

Flux-corrected transport algorithm:

1. Calculate the low-order (overdiffusive) solution u^{Low}:

$$[\mathcal{M} - \Delta t\, \mathcal{K}^*]\, u^{\text{Low}} = \mathcal{M} u^n + \Delta t\, g^n,$$

where $\mathcal{K}^* = \mathcal{K} + \mathcal{D}$ is M-Matrix and $d_{ij} = \max\{-k_{ij}, 0, -k_{ji}\}, \;\; j \neq i$.

2. reconstruct the high-order solution u^{m+1} (i.e. to remove the artificial diffusion where it is possible):

$$\mathcal{M} u^{n+1} = \mathcal{M} u^{\text{Low}} + \Delta t \bar{f}(u^{\text{Low}}, u^n),$$

where $\bar{f}_i = \sum_{j \neq i} \alpha_{ij} f_{ij}$ with flux limiters $\alpha_{ij} \in [0, 1]$ and

$$f_{ij} = m_{ij}(\dot{u}_i^{\text{Low}} - \dot{u}_j^{\text{Low}}) + d_{ij}(u_i^{\text{Low}} - u_j^{\text{Low}}) \;\; \text{– antidiffusive fluxes.}$$

$$\dot{u}^{\text{Low}} = \mathcal{M}^{-1} (\mathcal{K}^*)\, u^{\text{Low}}$$

FCT technique (implicit Euler)

$$[\,\mathcal{M} - \Delta t\, \mathcal{K}\,]\, u^{n+1} = \mathcal{M}\, u^n + \Delta t\, g^n.$$

Flux-corrected transport algorithm:

1. Calculate the low-order (overdiffusive) solution u^{Low}:

$$[\mathcal{M} - \Delta t\, \mathcal{K}^*]\, u^{\mathrm{Low}} = \mathcal{M} u^n + \Delta t\, g^n,$$

   where $\mathcal{K}^* = \mathcal{K} + \mathcal{D}$ is M-Matrix and $d_{ij} = \max\{-k_{ij}, 0, -k_{ji}\},\;\; j \neq i$.

2. reconstruct the high-order solution u^{m+1} (i.e. to remove the artificial diffusion where it is possible):

$$\mathcal{M} u^{n+1} = \mathcal{M} u^{\mathrm{Low}} + \Delta t \bar{f}(u^{\mathrm{Low}}, u^n),$$

   where $\bar{f}_i = \sum_{j \neq i} \alpha_{ij} f_{ij}$ with flux limiters $\alpha_{ij} \in [0, 1]$ and

$$f_{ij} = m_{ij}(\dot{u}_i^{\mathrm{Low}} - \dot{u}_j^{\mathrm{Low}}) + d_{ij}(u_i^{\mathrm{Low}} - u_j^{\mathrm{Low}}) \;\; - \text{antidiffusive fluxes.}$$

$$\dot{u}^{\mathrm{Low}} = \mathcal{M}^{-1}\, (\mathcal{K}^*)\, u^{\mathrm{Low}}$$

FCT: flux limiters for node i

Elimination of the artificial diffusion, which is due to $\mathcal{D} = \{d_{ij}\}$:

1. Compute the sums of positive/negative antidiffusive fluxes into node i

$$P_i^+ = \sum_{j \neq i} \max\{0, f_{ij}\}, \qquad P_i^- = \sum_{j \neq i} \min\{0, f_{ij}\}.$$

2. Compute the distance to a local extremum of the auxiliary solution u

$$Q_i^+ = \max\{0, \max_{j \neq i}(u_j^{n,\mathrm{Low}} - u_i^{n,\mathrm{Low}})\}, \qquad Q_i^- = \min\{0, \min_{j \neq i}(u_j^{n,\mathrm{Low}} - u_i^{n,\mathrm{Low}})\}.$$

3. Compute the nodal correction factors for the net increment to node i

$$R_i^+ = \begin{cases} \min\left\{1, \dfrac{m_i\, Q_i^+}{\Delta t P_i^+}\right\}, & P_i^+ \neq 0 \\ 0, & P_i^+ = 0 \end{cases} \qquad R_i^- = \begin{cases} \min\left\{1, \dfrac{m_i\, Q_i^-}{\Delta t P_i^-}\right\}, & P_i^- \neq 0 \\ 0, & P_i^- = 0 \end{cases}$$

4. Check the sign of the antidiffusive flux and apply the correction factor

$$\alpha_{ij} = \begin{cases} \min\{R_i^+, R_j^-\}, & \text{if } f_{ij} > 0, \\ \min\{R_i^-, R_j^+\}, & \text{otherwise.} \end{cases}$$

FCT technique (θ-scheme)

$$(\mathcal{M} - \theta \Delta t \mathcal{K})\, \boldsymbol{u}^{n+1} = (\mathcal{M} + (1-\theta)\Delta t \mathcal{K})\, \boldsymbol{u}^{n}.$$

1. Compute the high-order solution $\bar{\boldsymbol{u}}^{n+1}$ from the algebraic system

$$(\mathcal{M} - \theta \Delta t \mathcal{K})\, \bar{\boldsymbol{u}}^{n+1} = (\mathcal{M} + (1-\theta)\Delta t \mathcal{K})\, \boldsymbol{u}^{n}.$$

2. Compute the intermediate solution $\boldsymbol{u}^{n+\theta}$ by the low-order scheme

$$\mathcal{M} \boldsymbol{u}^{n+\theta} = \left(\mathcal{M} + (1-\theta)\Delta t \mathcal{K}^{L}\right) \boldsymbol{u}^{n}.$$

3. Compute antidiffusive fluxes $f_{ij}(\boldsymbol{u}^n, \boldsymbol{u}^{n+\theta})$ and correction factors $\alpha_{ij}^{n+\theta}$ to find the intermediate solution $\boldsymbol{u}^*$:

$$\mathcal{M} \boldsymbol{u}^* = \mathcal{M} \boldsymbol{u}^{n+\theta} + \Delta t \sum_{j \neq i} \alpha_{ij}^{n+\theta} f_{ij}(\boldsymbol{u}^n, \boldsymbol{u}^{n+\theta}).$$

4. Compute the stabilized high-order solution $\boldsymbol{u}^{n+1}$:

$$\left(\mathcal{M} - \theta \Delta t \mathcal{K}^{L}\right) \boldsymbol{u}^{n+1} = \mathcal{M} \boldsymbol{u}^*.$$

FCT technique (θ-scheme)

$$(\mathcal{M} - \theta\Delta t\mathcal{K})\, \boldsymbol{u}^{n+1} = (\mathcal{M} + (1-\theta)\Delta t\mathcal{K})\, \boldsymbol{u}^{n}.$$

Flux-corrected transport algorithm:

1. Compute the high-order solution $\bar{\boldsymbol{u}}^{n+1}$ from the algebraic system

$$(\mathcal{M} - \theta\Delta t\mathcal{K})\, \bar{\boldsymbol{u}}^{n+1} = (\mathcal{M} + (1-\theta)\Delta t\mathcal{K})\, \boldsymbol{u}^{n}.$$

2. Compute the intermediate solution $\boldsymbol{u}^{n+\theta}$ by the low-order scheme

$$\mathcal{M}\boldsymbol{u}^{n+\theta} = \left(\mathcal{M} + (1-\theta)\Delta t\mathcal{K}^{L}\right) \boldsymbol{u}^{n}.$$

3. Compute antidiffusive fluxes $f_{ij}(\boldsymbol{u}^{n}, \boldsymbol{u}^{n+\theta})$ and correction factors $\alpha_{ij}^{n+\theta}$ to find the intermediate solution $\boldsymbol{u}^{*}$:

$$\mathcal{M}\boldsymbol{u}^{*} = \mathcal{M}\boldsymbol{u}^{n+\theta} + \Delta t \sum_{j\neq i} \alpha_{ij}^{n+\theta} f_{ij}(\boldsymbol{u}^{n}, \boldsymbol{u}^{n+\theta}).$$

4. Compute the stabilized high-order solution $\boldsymbol{u}^{n+1}$:

$$\left(\mathcal{M} - \theta\Delta t\mathcal{K}^{L}\right) \boldsymbol{u}^{n+1} = \mathcal{M}\boldsymbol{u}^{*}.$$

FCT technique (θ-scheme)

$$(\mathcal{M} - \theta\Delta t\mathcal{K})\, \boldsymbol{u}^{n+1} = (\mathcal{M} + (1-\theta)\Delta t\mathcal{K})\, \boldsymbol{u}^{n}.$$

Flux-corrected transport algorithm:

1. Compute the high-order solution $\bar{\boldsymbol{u}}^{n+1}$ from the algebraic system

$$(\mathcal{M} - \theta\Delta t\mathcal{K})\, \bar{\boldsymbol{u}}^{n+1} = (\mathcal{M} + (1-\theta)\Delta t\mathcal{K})\, \boldsymbol{u}^{n}.$$

2. Compute the intermediate solution $\boldsymbol{u}^{n+\theta}$ by the low-order scheme

$$\mathcal{M}\boldsymbol{u}^{n+\theta} = \left(\mathcal{M} + (1-\theta)\Delta t\mathcal{K}^L\right) \boldsymbol{u}^{n}.$$

3. Compute antidiffusive fluxes $f_{ij}(\boldsymbol{u}^n, \boldsymbol{u}^{n+\theta})$ and correction factors $\alpha_{ij}^{n+\theta}$ to find the intermediate solution $\boldsymbol{u}^*$:

$$\mathcal{M}\boldsymbol{u}^* = \mathcal{M}\boldsymbol{u}^{n+\theta} + \Delta t \sum_{j\neq i} \alpha_{ij}^{n+\theta} f_{ij}(\boldsymbol{u}^n, \boldsymbol{u}^{n+\theta}).$$

4. Compute the stabilized high-order solution $\boldsymbol{u}^{n+1}$:

$$\left(\mathcal{M} - \theta\Delta t\mathcal{K}^L\right) \boldsymbol{u}^{n+1} = \mathcal{M}\boldsymbol{u}^*.$$

FCT technique (θ-scheme)

$$(\mathcal{M} - \theta \Delta t \mathcal{K})\, \boldsymbol{u}^{n+1} = (\mathcal{M} + (1-\theta)\Delta t \mathcal{K})\, \boldsymbol{u}^{n}.$$

Flux-corrected transport algorithm:

1. Compute the high-order solution $\bar{\boldsymbol{u}}^{n+1}$ from the algebraic system

$$(\mathcal{M} - \theta \Delta t \mathcal{K})\, \bar{\boldsymbol{u}}^{n+1} = (\mathcal{M} + (1-\theta)\Delta t \mathcal{K})\, \boldsymbol{u}^{n}.$$

2. Compute the intermediate solution $\boldsymbol{u}^{n+\theta}$ by the low-order scheme

$$\mathcal{M}\boldsymbol{u}^{n+\theta} = \left(\mathcal{M} + (1-\theta)\Delta t \mathcal{K}^{L}\right) \boldsymbol{u}^{n}.$$

3. Compute antidiffusive fluxes $f_{ij}(\boldsymbol{u}^{n}, \boldsymbol{u}^{n+\theta})$ and correction factors $\alpha_{ij}^{n+\theta}$ to find the intermediate solution $\boldsymbol{u}^{*}$:

$$\mathcal{M}\boldsymbol{u}^{*} = \mathcal{M}\boldsymbol{u}^{n+\theta} + \Delta t \sum_{j \neq i} \alpha_{ij}^{n+\theta} f_{ij}(\boldsymbol{u}^{n}, \boldsymbol{u}^{n+\theta}).$$

4. Compute the stabilized high-order solution $\boldsymbol{u}^{n+1}$:

$$\left(\mathcal{M} - \theta \Delta t \mathcal{K}^{L}\right) \boldsymbol{u}^{n+1} = \mathcal{M}\boldsymbol{u}^{*}.$$

FCT technique (θ-scheme)

$$(\mathcal{M} - \theta \Delta t \mathcal{K})\, \boldsymbol{u}^{n+1} = (\mathcal{M} + (1-\theta)\Delta t \mathcal{K})\, \boldsymbol{u}^{n}.$$

Flux-corrected transport algorithm:

1. Compute the high-order solution $\bar{\boldsymbol{u}}^{n+1}$ from the algebraic system

$$(\mathcal{M} - \theta \Delta t \mathcal{K})\, \bar{\boldsymbol{u}}^{n+1} = (\mathcal{M} + (1-\theta)\Delta t \mathcal{K})\, \boldsymbol{u}^{n}.$$

2. Compute the intermediate solution $\boldsymbol{u}^{n+\theta}$ by the low-order scheme

$$\mathcal{M}\boldsymbol{u}^{n+\theta} = \left(\mathcal{M} + (1-\theta)\Delta t \mathcal{K}^L\right) \boldsymbol{u}^{n}.$$

3. Compute antidiffusive fluxes $f_{ij}(\boldsymbol{u}^n, \boldsymbol{u}^{n+\theta})$ and correction factors $\alpha_{ij}^{n+\theta}$ to find the intermediate solution $\boldsymbol{u}^*$:

$$\mathcal{M}\boldsymbol{u}^* = \mathcal{M}\boldsymbol{u}^{n+\theta} + \Delta t \sum_{j \neq i} \alpha_{ij}^{n+\theta} f_{ij}(\boldsymbol{u}^n, \boldsymbol{u}^{n+\theta}).$$

4. Compute the stabilized high-order solution $\boldsymbol{u}^{n+1}$:

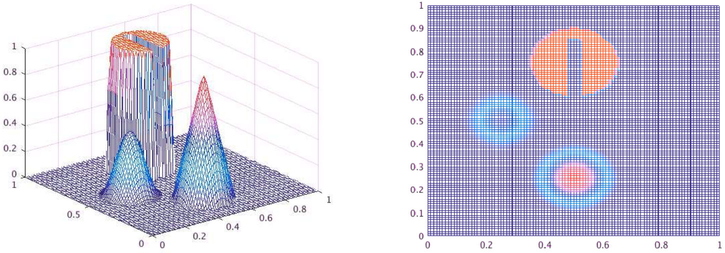
$$\left(\mathcal{M} - \theta \Delta t \mathcal{K}^L\right) \boldsymbol{u}^{n+1} = \mathcal{M}\boldsymbol{u}^*.$$

FCT technique: properties

1. **Positivity preservation**
2. **Mass conservation**
3. **High-order (cf., upwind-like schemes)**
4. **General purpose wrt. dimension of a problem and mesh**
5. **Does not require any artificial parameters**
6. **Applicable for convection dominated problems in domains and on surfaces**

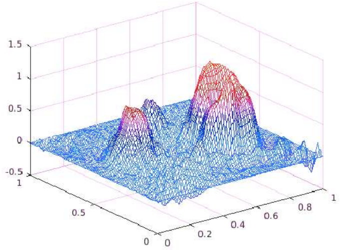
The solid body rotation:

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

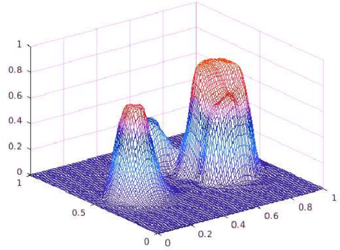


(a) front view (b) top view

Figure: Initial condition.



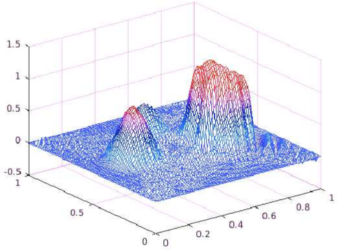
(a) non-stabilized, $N = 50 \times 50$



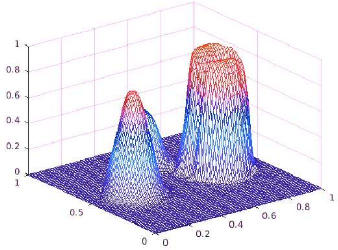
(b) FCT-stabilized, $N = 50 \times 50$

Figure: Numerical solution after rotation by the angle $\alpha = \dfrac{2\pi}{3}$.

FCT technique: transport equation

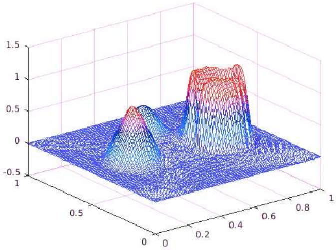


(a) non-stabilized, $N = 100 \times 100$

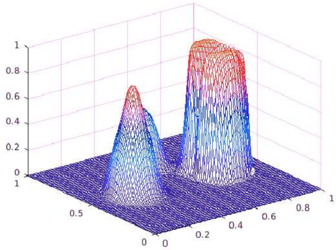


(b) FCT-stabilized, $N = 100 \times 100$

Figure: Numerical solution after rotation by the angle $\alpha = \dfrac{2\pi}{3}$.



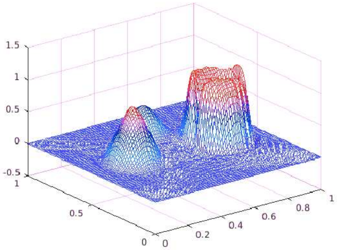
(a) non-stabilized, $N = 200 \times 200$



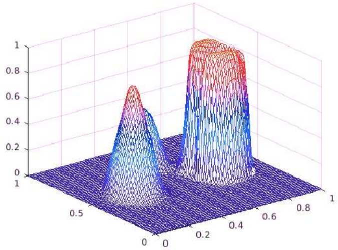
(b) FCT-stabilized, $N = 200 \times 200$

Figure: Numerical solution after rotation by the angle $\alpha = \dfrac{2\pi}{3}$.

FCT technique: transport equation



(a) non-stabilized, $N = 200 \times 200$



(b) FCT-stabilized, $N = 200 \times 200$

Figure: Numerical solution after rotation by the angle $\alpha = \frac{2\pi}{3}$.

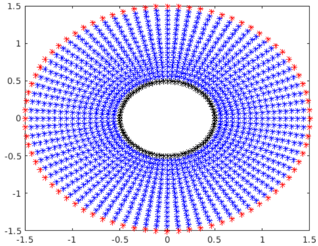
for details see Kuzmin, Sokolov, Davydov, Turek 2018

FCT technique: transport on manifold

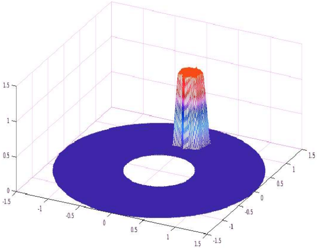
Configuration:

$$u_t + \mathbf{v} \cdot \nabla_\Gamma u = 0,$$

where $\phi(\boldsymbol{x}) = |\boldsymbol{x}| - 1.0$ and $\boldsymbol{v} = (-1, 0)^T$.



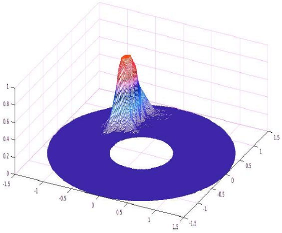
(a) placement of 1377 data points



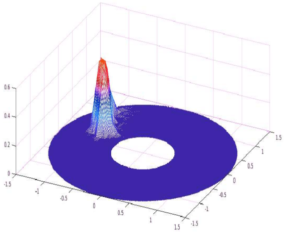
(b) initial condition

Figure: Initial setting.

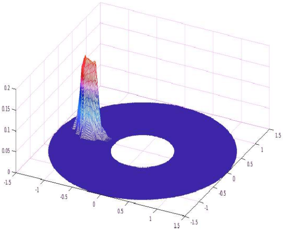
FCT technique: transport on manifold



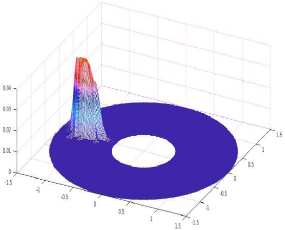
(a) $\alpha = \frac{\pi}{4}$



(b) $\alpha = \frac{\pi}{2}$

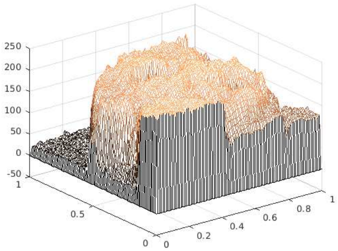


(c) $\alpha = \pi$

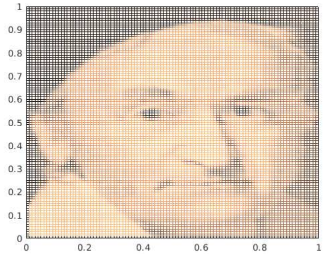


(d) $\alpha = \frac{3\pi}{2}$

Final example



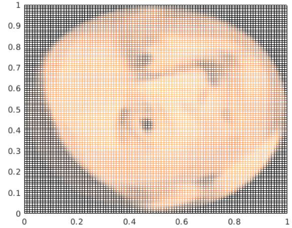
(e) front view



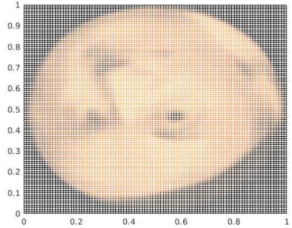
(f) top view

Figure: Initial condition, N = 200.

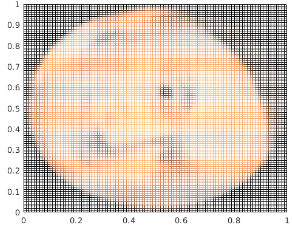
Final example



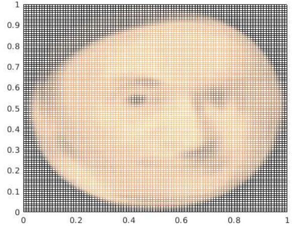
(a) $\alpha = \frac{\pi}{2}$



(b) $\alpha = \pi$



(c) $\alpha = \frac{3\pi}{2}$



(d) $\alpha = 2\pi$

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.**
4. **The incorporated FCT-stabilization technique make its possible to treat convection dominated problems in domains and on surfacess.**

Acknowledgements

- 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

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